

Volume 3
Issue 6

RISC USER

MAY 1990
Price £1.60

Bank Switching

Arc to Psion

Desktop File Loader

A Desktop CMOS RAM Manager



Next Technology's CD ROM
for the Archimedes

The magazine and
support group
exclusively for users
of the Archimedes



CONTENTS

FEATURES

	4
News	11
Bank Switching	19
Using Draw Files in Basic (Part 2)	23
Bits and PCs (Part 2)	27
Assembler Workshop: Quick Sort	39
Into the ARC: The Palette and the Task Manager	49
Mastering the Wimp (Part 8): Icons cont'd	52
Points Arising	56
Hints & Tips	59
Postbag	61

UTILITIES AND APPLICATIONS

	15
Arc to Psion	35
Desktop File Loader	43
A Desktop CMOS RAM Manager	47
Desktop Wibble	

REVIEWS

	7
Next Technology's CD ROM for the Archimedes	32
Improving the Picture: Three monitors from Taxan	44
CAD from Linear Graphics	53
Archimedes and RISC OS: Book review	

RISC
USERTHE ARCHIMEDES
MAGAZINE AND
SUPPORT GROUP

RISC User is published by BEEBUG Ltd.

Co-Editors:

Dr Lee Calcraft, Mike Williams

Assistant Editor: Kristina Lucas

Technical Editor: Alan Wrigley

Technical Assistant: Glyn Clements

Production Assistant: Sheila Stoneman

Advertising: Sarah Shrive

All rights reserved. No part of this publication may be reproduced without prior written permission of the Publisher.

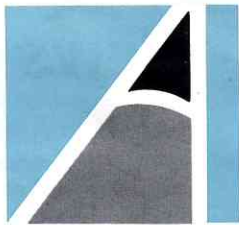
The Publisher cannot accept any responsibility whatsoever for errors in articles, programs, or advertisements published. The opinions expressed on the pages of this journal are those of the authors and do not necessarily represent those of the Publisher, BEEBUG Limited.

BEEBUG Ltd © 1990

BEEBUG, 117 Hatfield Road, St. Albans, Herts AL1 4JS.

Tel: St. Albans (0727) 40303.

Fax: (0727) 60263



FEEDBACK PLEASE

In this issue of the magazine we have an article on using the Archimedes in conjunction with the Psion Organiser, and next month we will be covering the Cambridge Computer Z88 from a similar point of view. If you would be interested in follow-up articles on either of these topics, please let us know.

BEEBUG PRODUCTS FOR THE ARCHIMEDES

While work at BEEBUG continues to put the finishing touches to the new DTP package, recently christened **Ovation**, two other products are also about to see the light of day:

Font Pack One

This contains four complete outline fonts for the Archimedes:

A Palatino equivalent

An Avant Garde equivalent

A Helvetica (or Homerton) equivalent

and A Symbol set of Greek and maths characters.

This will be available to members by the end of April at £49.50 inc. VAT.

ISO C

This is a brand new C language development pack. It runs from the RISC OS Desktop, and contains its own multi-tasking editor (which operates similarly to Edit). All the processes involved in C development have been fully integrated in

this package, so that all you need to do is to click on the editor, type in your C program, then click on **Compile**, and your program will be compiled, linked and run - all in a single sequence controlled from the Desktop.

This compiler also includes a built-in ARM code assembler, and will run from floppies in a 1Mbyte machine. It costs £79.00 inc. VAT to members, and is available now in a pre-release version (with free upgrades to the release version).

PRICE REDUCTION ON ACORN COMPUTERS

Latest news from Acorn is that prices on all machines are being reduced as from 1st April 1990. New prices are: **Master 128** £399 (was £439), **A3000** £599 (was £649), **A410/1** £1099 (was £1199), **A420/1** £1499 (from £1699) and the **A440/1** £2099 (from £2499) - all prices exclude VAT. This will hopefully give a welcome boost to the Acorn market.

SPRING OPEN DAY - 22 APRIL

If there are no hitches with this issue, you should receive it in time for a final reminder about our Spring Open Day. Come along and talk to magazine staff, bring your technical queries, and take a closer look at what is new in the Archimedes world. There will be stands manned by Acorn, Computer Concepts, Colton Software, EMR and many more.

L.G.C.

THE LEARNING CURVE

On April 2nd, Acorn launched a major new initiative aimed at increasing its share of the domestic market. *The Learning Curve Project* is targeted at parents, in an effort to encourage them to provide their children with the same computing environment at home as at school. The heart of the project is a package comprising an A3000 computer bundled with First Word Plus, Genesis (which was reviewed in last month's RISC User), and a PC Emulator package containing DR-DOS - Digital Research's version of MS-DOS. Also included are a demonstration video and a parents' guide to the National Curriculum. All this costs only £50 (plus VAT) more than the price of the basic A3000, and as such represents excellent value.

The project will be supported by a major marketing effort from dealers, an advertising campaign in the national press, television advertising in the TVS region and a Sunday Times video which offers parents a guide to educational computing. Schools will also be able to participate by staging a parents' evening to promote the project. Schools agreeing to stage one or more evenings will be able to purchase one A3000 at a discount of 33%.

The Learning Curve costs £803.85 inc. VAT for a base system, or £1091.35 inc. VAT with a colour monitor, and is available now directly from BEEBUG. Information leaflets are also available from BEEBUG, tel. (0727) 40303.

Acorn Computers is at Fulbourn Road, Cherry Hinton, Cambridge CB1 4JN, tel. (0223) 245200.

MORE MUSIC

Arriving soon from Clares will be a new music package, which aims to expand and improve on the facilities offered by Acorn's *Maestro*. Currently being developed under the working title of *MidiScribe*, the package will offer better editing facilities than

Maestro, and will also accept input from a MIDI keyboard. It will be possible to play a tune on the keyboard, transcribe it onto staves in *MidiScribe* and produce hardcopy on a printer. The release date is expected to be early summer. For further details contact Clares Micro Supplies, 98 Middlewich Road, Northwich, Cheshire, tel. (0606) 48511.

AWAY WITH DONGLES

In response to feedback from schools, many of whom are worried that dongles might be stolen, Computer Concepts is releasing a non-dongled, network protected version of *Impression* which will be licensed to a particular school, and will only run on that school's network. Further information may be obtained from Computer Concepts, Gaddesden Place, Hemel Hempstead, Herts HP2 6EX, tel. (0442) 63933.

ACCOUNTING FOR PERSONAL TASTES

Apricote Studios, whose *Account Book* and *Invoice Program* packages have proved popular with small businesses, has now turned its attention to the domestic market. *Personal Accounts* is claimed to be the most user-friendly home and club accounts program yet, and features multiple bank accounts, direct debits and powerful search and edit facilities. Running on a 1 Mbyte machine, there is sufficient memory for over 20,000 entries. *Personal Accounts* costs £14.95 inc. VAT, and is available from Apricote Studios, 2 Purls Bridge Farm, Manea, Cambridgeshire PE15 0ND, tel. (035 478) 432. Versions for the BBC model B and Master are also available.

GRAPHICS PACKAGES

Higher education, engineering and research form the focus for two new graphics packages, FAM and GINO, announced by Acorn. FAM is a Unix-based package which runs in R140 computers under RISC iX. It is designed to interface with all major CAD systems, and provides powerful finite

element modelling facilities. Currently available are FAMbuild, for modelling and analysis, and FAMresult for display and graphical presentation of results.

GINO is a well-known FORTRAN graphics subroutine library. This new implementation comprises 3 modules and runs on Archimedes or R140 computers under RISC OS. GINO-F 3D provides a library of FORTRAN subroutines containing a variety of drawing and graphic functions, GINOGRAF has facilities for producing graphs and charts, and GINSURF displays 3D data as contour maps or surface views with hidden line removal.

Acorn Computers can be contacted at the address given earlier.

MORE MERGERS

Hot on the heels of its recent merger, Longman Logotron is now to merge with BBC Soft, the software arm of BBC Enterprises. The combined operation will be managed by Longman Logotron and will be one of the largest companies in the educational software field. Existing BBC Soft packages, such as the popular comms program *ArcComm*, will continue to be marketed by the new company, and further titles will be developed shortly. For further information, contact Longman Logotron, Dales Brewery, Cambridge CB1 2LJ, tel. (0223) 323656.

MAPPING THE WORLD

Micro Studio is releasing a set of discs containing maps of the world in Draw file format, covering well over 100 countries. The maps are drawn in black outline, with major cities marked, and large areas of water shown in blue. Further colouring can be added using Draw. The complete set costs £29.95 inc. VAT and is available from April. Micro Studio can be contacted at 22 Churchgate Street, Soham, Ely, Cambridgeshire CB7 5DS, tel. (0353) 720433.

ALL CHANGE AT THE TOP

Acorn has a new managing director following the promotion of Harvey Coleman to a senior position with Olivetti in Canada. The new man is Sam Wauchope, formerly Acorn's sales and marketing director. In announcing the change, the company took the opportunity to stress its confidence in the future, which is good news for all users of Acorn computers.

EXPAND YOUR HORIZONS

The *Dual User Port Expansion Card*, from Science Frontiers, is a podule for A300 and A400 series machines, which uses the same 65C22 VIA chip as the BBC micro to provide the Archimedes with two BBC-compatible 8-bit bi-directional input/output ports and two hardware timers. Thus the podule can be used to control a variety of external devices, in exactly the same way as the BBC. It can be accessed from Basic, C, or by using SWI calls. Control routines are supplied in ROM on the board itself, and interrupts (IRQ or FIQ) are supported. The expansion card costs £97.69 inc. VAT, and is available from Science Frontiers, 7 Porthill Court, Aberdeen AB1 1DU, tel. (0224) 642765.

LONG LIVE THE 310?

There has been much speculation and rumour recently concerning the future of the Archimedes 310, and in particular whether the next version of RISC OS, when it arrives, will render the machine obsolete because of the 310's inability to address more than 1/2 Mbyte of ROM. While there is no concrete news on this subject as yet, it does seem likely that third party developers are working on upgrades to the 310 to allow it to handle larger ROMs in future, and if this is the case it will be good news for 310 owners who have been worried about being left behind despite having invested only recently in the very latest technology.

RU

Archimedes Software



Disc 7 - A New Chess Program for The Archimedes.

At last, a Chess program that takes full advantage of RISC OS. Chess is now a true RISC OS application, installing itself on the icon bar and running on the desktop in a window at the same time as other applications. This means you can carry on working with other programs whilst the Chess program is thinking of its next move.



Features: load and save games, edit board, step backwards and forwards through game, computer 'v' computer mode, play black or white, reverse board, save game setup. multiple levels, print game. Moves displayed in algebraic notation. Full mouse control over game, on screen clocks. Hint. Choice of piece when promoting pawn. New better design of board and pieces. Undo move. Complete with full Acorn ANSI C V3.00 source code - a model C application. A good chess game that fits beautifully into RISC OS.



Price £5.99 inclusive.



Disc 16 - Spark Super RISC OS Archive Program.

Spark, allows you to store files and directories in a highly compressed form in archive files. Fully RISC OS compatible, multitasking, completely mouse driven, the contents of archives are shown in disc type viewers. Perfect for sending Impression documents by electronic mail. Files can be dragged from archives into applications or from an application into an archive. It is possible to move around the directories in an archive and add or delete files. Spark is compatible with IBM '.arc' and pkarc files as well as with the older Archimedes 'arc' files. The Spark disc also contains a number of pd applications, IBackDrop sticky desktop and almost 1Mbyte of colour pictures.



Price £5.99 inclusive.

Also available:

- Disc 1 EMACS. Classic multi document editor with macro language. Includes full C source.
- Disc 2 MicroSpell. 43,000 word real time spelling checker module works with !Edit etc..
- Disc 4 XLISP. Object oriented version of LISP with C source and progs. New 2.00 version.
- Disc 5 C Toolkit. 20+progs for C programmers with sources. grep, awk, make, sed, more...
- Disc 6 Kermit. Archimedes, BBC and MSDOS versions of file transfer/comms. prog.
- Disc 8 CrossStar. Wimp based crossword solver with huge dictionary.
- Disc 10 More C Tools. yacc and lex with examples and C source. diff, ctags, valspeak...
- Disc 11 Little SmallTalk. original object oriented language with C source, docs, programs.
- Disc 14 GNU Tools#1. Quality tools with source. grep, egrep, sed, awk, diff. New versions.
- Disc 15 Clip Art#1 50 black and white pictures, also 65 Epson LQ printer fonts, !Sparkplug
- Disc 18 MTV Raytracer. database driven 3D raytracing package with C source code. Lots of fun making pretty raytraced pictures. Animated IMTVDemo, !SparkPlug, examples.
- Disc 19 Starchart. RISC OS program to make Draw file starmaps. Catalog of 9000 stars.
- Disc 21 More U Tools. Patch. Suite of programs to handle shar files, Rh, like 'find', executes commands on files, Csplit, GnuDbm - GNU database manager. All with C sources.

Each disc is £5.99 inc. Buy four claim another one free!

David Pilling, P.O. Box 22, Thornton Cleveleys, Blackpool. FY5 1LR.

Free Air Mail delivery on overseas orders. Full list available on request.

Next Technology's CD ROM for the Archimedes

Mike Williams previews the latest innovation for the Archimedes in the form of CD-ROM from Next Technology.

The latest technology to impact on the computer scene is in the area of multimedia systems, and the Archimedes with its sheer power and high performance graphics is ideally placed to exploit such systems. Acorn has already made one foray into this market with interactive video (the Doomsday project), which was perhaps ahead of its time as well as being too expensive to be widely affordable. The latest venture is a collaboration with Cambridge based Next Technology which has developed a CD-ROM unit for the Archimedes.



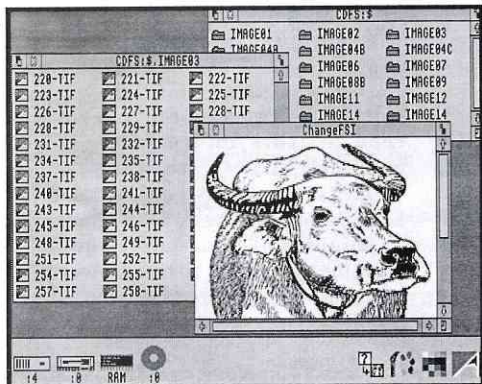
The Next Technology CD-ROM unit should be available by the time you read this. Our preview is based on experiences with a development system running on pre-release software. But first of all, what is CD-ROM?

Everyone is more or less familiar with audio CD (standing for Compact Disc), small digitally recorded discs read by laser, and claimed to give greater clarity in hi-fi sound than conventional vinyl records. CD-ROM uses the same basic technology to record data (which can replace or supplement audio) on CD. With

suitable software, a computer can then read all this information. CD-ROM currently has a capacity of some 600MBytes - that's equivalent to some 250,000 pages of text, 5000 photographs, 72 minutes of high quality audio, or a combination of all three. However, CDs are read-only, hence the term CD-ROM (CD Read Only Memory).

Next Technology has developed a CD-ROM unit, capable of reading CD-ROMs and feeding the data into the Archimedes. Any audio is output separately, and can be directed to the Archimedes monitor's built in speaker(s), or to any hi-fi amplifier and speaker system. Sensibly, Next has adopted the standard ISO 9660 format (itself derived from the earlier High Sierra standard). This means that the Archimedes can access all other CD-ROMs produced to this standard, and several hundred already exist in the PC world. Next has also produced the software to access CD-ROM in the form of the CDFS (CD Filing System). This is fully compatible with all other Archimedes filing systems.

The Next CD-ROM unit consists of a flat box which sits easily on top of a 400/1 series machine and under the monitor (though it can be used with any Archimedes including an A3000), and is connected to the host system via a SCSI interface (supplied as part of the system). This means that as many as 28 CD-ROM units may be connected via the one SCSI interface providing some 17 gigabytes of information! The data transfer rate is of the order of 150K bytes per second with an average access time of 0.4 seconds. CDs must be enclosed in a protective case or caddy (one is supplied with a sample CD-ROM) for insertion into the CD-ROM unit.



Using CDFS to access and convert a TIFF format picture

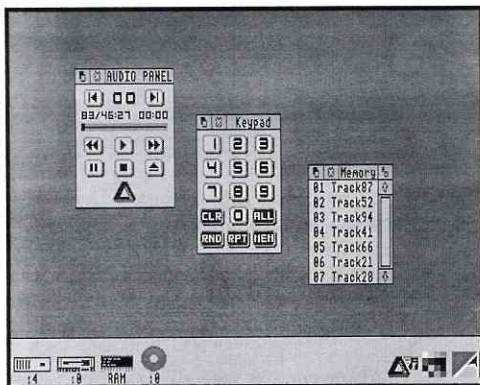
The CDFS will appear on the icon bar as a filing system (though this had not been fully implemented on the development system). Clicking on the drive icon, as with any filing system, will open a window to show the contents of the CD-ROM. Using a clip-art CD produced for PCs, opening any of the top level directories revealed hundreds of picture files (in TIFF format). Using the program ChangeFSI (see RISC User Volume 2 Issue 10) it was a trivial matter to select any picture and convert it into an Archimedes sprite (it is likely that the ChangeSFI program will be supplied with the CD-ROM unit, as it can convert a wide variety of screen formats into equivalent Archimedes sprites).

This illustrates one application of CD-ROM, with its huge storage capacity, as a resource system. An Archimedes with CD-ROM unit can be readily connected to an Econet network, allowing the CD-ROM unit to be accessed by all network users, including those with a BBC model B or Master series computer! Because of the way it is implemented, it would be very easy to set up CD-ROM applications accessed using Genesis (see review in RISC User Volume 3 Issue 5).

The data on a CD can be anything (text, pictures, sound, animations etc). It

is also possible to include programs, but most suppliers of CD-ROMs will more likely supply any software on a standard computer disc. That way, a CD-ROM can be used on any system; only the software needs customising. Hopefully, that will mean that the Archimedes, with its proprietary operating system RISC OS, will still be able to benefit from most of the CD-ROMs produced.

The CD-ROM unit can also play standard audio CDs, and thus perform as a very high quality CD player. As part of the system, users will be provided with the Audio Panel, window-based software for controlling the playing of audio CDs. This controls all normal functions, such as play, forward, reverse, next track, random play, continuous play etc, and you can also program the sequence in which tracks will be played, although there is no facility at present to save a programmed sequence. Thus Archimedes users buying a CD-ROM unit are also getting a fully fledged audio CD player, and one which can be playing as you use your Archimedes for other work.



The Audio Panel in operation

Operation of all features is simplicity itself, relying as it does on standard Wimp and filing system conventions. Indeed, it is almost a disappointment that cataloguing a CD-ROM produces a directory viewer

Next Technology's CD ROM for the Archimedes

much like any other, but that in itself is the benefit of a standardised approach. The results, whether using data, or audio, are certainly impressive.

If schools or any other group want to create their own CD-ROM, then Next have the facilities to do this at a cost of between £200 and £300 per CD. Normally CDs are pressed from a high precision master (which costs £2000 to £3000). So-called recordable CD (also known as Write Once, Read Only) uses a layer of three metals which can be fused to create the CD data bits directly using a laser. The only problem in creating a full capacity CD-ROM is that the data will occupy some 800 floppy discs! However, Next also promises for the future a development system with a high capacity hard disc drive and nine track tape recorder (at a combined cost of the order of £8000).

At the moment, both data and audio can be placed on the same CD, but the controlling software must know the type of signal it expects to receive. Future

developments include CD-ROM XA (extended architecture). With this, the decoding hardware can detect whether it is reading data or audio, separate the two and present two signals concurrently, offering the potential for simultaneous sound and data).

The CD-ROM unit will be marketed and supported by Next Technology, but discussions are taking place with Acorn and other software houses for the development of more CD-ROM based information. For the time being the combination of Archimedes and CD-ROM offers a power and versatility which few other systems are likely to match, certainly not at this price.

Product	CD-ROM Unit
Supplier	Next Technology Corporation Ltd St John's Innovation Centre, Cambridge CB4 4WS. Tel. (0223) 420222
Price	£695 plus VAT (A3000) including SCSI interface, all connecting leads, sample CD and documentation (price for Archimedes A300/400 version to be announced - about £50 more). RU



ArcScan II Contains the Following Databases:

- RISC User Volumes 1 - 2
- BEEBUG Volumes 1 - 8
- The Indexes to the following Archimedes Acorn manuals:
 - Basic User Guide (Arthur)
 - Programmer's Reference Manual (Arthur)
 - RISC OS User Guide
 - RISC OS Basic Manual
 - RISC OS Programmer's Reference Manual

Fully Customisable

A user-defined custom file allows easy customisation for other data.

For example:

1. If you need to know about Obey Files, enter the words 'obey', 'files' and click on the AND logic. All finds in RISC User will be instantly displayed. You can now call up the Acorn Manuals indexes and repeat the search to get further references.
2. To find the SWI &400D4, select the new Programmers Reference Manual from the menu, enter "SWI &400D4", and the answer appears, complete with page references.

ArcScan II



ArcScan Features

- Fast ARM Code searching
- Mouse controlled
- Runs from the Desktop
- Keyword entry with selectable logic
- Fully customisable
- Extensive on-screen help
- Hard copy options

ArcScan II Disc + manual £14.95 members (£18.95 non-members) Code 0109A ArcScan Upgrade £7.95 members (£10.95 non-members) Code 0110A
Please add 60pp&p. For upgrade, please state month of purchase of RU Special Disc.....

Phone your order now on (0727) 40303 or send your cheque/postal order to the address below. Please quote your name and membership number. When ordering by Access, Visa or Connect, please quote your card number and the expiry date.

BEEBUG Ltd, 117 Hatfield Road, St Albans, Herts AL1 4JS. Telephone (0727) 40303.

MORE NEW RELEASES FROM DABS PRESS

Z88 PipeDream: A Dabhand Guide

Z88

from John Allen - expert Z88 journalist

PipeDream, the built in word processor, spreadsheet and database built into the Z88 is an extremely powerful tool, and John Allen, author of many articles of the Z88, and experienced user has produced what must be the ultimate tutorial and reference work on the software. With many example documents and templates, this book takes you from first principles to a thorough understanding of the program. Linking to PipeDream on other machines such as the BBC and Archimedes is also covered.

Only £14.95 all inclusive

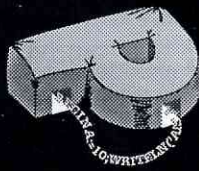
AVAILABLE 24th APRIL

Also still available is our first book on the famous Cambridge portable marvel, Z88: A Dabhand Guide, written by Trinity Concepts, authors of the Z88 operating system. Also priced at £14.95

THE LATEST DABHAND GUIDE

CAMBRIDGE PASCAL

Full featured Pascal Compiler for the Acorn
Archimedes



**DABS
PRESS**

Cambridge Pascal is a comprehensive version of the Pascal language and includes many extra enhancing features over the standard specification including full dynamic string handling, local error handling, random access files, direct access to 'star' commands and RISC OS SWI calls. It is 100% Desktop compatible but can

also be run from the command line. All programs compile into fast, efficient machine code.

Cambridge Pascal requires any Archimedes running RISC OS with at least 1Mb of RAM. Hard discs and extra memory are useful, but the system is perfectly workable with this minimum configuration.

Price £79.95 (£69.52+VAT). Site licences available. Free demonstration disc and full specification sheet available on request.

DEMO DISC FREE ON REQUEST

We have moved! Please note our new address and telephone number for all correspondence.

**DABS PRESS (RU5)
22 WARWICK STREET
PRESTWICH
MANCHESTER
M25 7HN
TEL: 061-773 8632
FAX: 061-773 8290**

For personal callers Warwick St. is in the centre of Prestwich Village, just off Bury New Road.

Dabs Press books and software are available from all good bookshops and Acorn dealers, or in case of difficulty direct from the above address. Access/Visa/official orders accepted. Free catalogues on request. Prices include 15% VAT (on software) and UK postage. Foreign deduct VAT but add £2.50 Europe/surface, £12 airmail.

061-773 8632

ARCHIMEDES FIRST STEPS

Beginners Guide to the
Archimedes & BBC A3000

This book is the perfect introductory guide to the Archimedes, to guide you through the first few months of ownership, acting as an easy-to-read supplement to the User Guide.

The book describes in detail how to put the RISC OS Desktop to best use, and also documents the programs on the Applications Discs provided with the machine.

But the book also goes beyond this and describes the sort of software and hardware additions available to the Archimedes owner, and how to choose and install them.

The many features of this book include:

- Applicable to all Archimedes with RISC OS
- Using the Desktop, RAMdisc and ADFS
- Edit, Paint, Draw and Maestro
- The Task Manager
- The BBC and PC emulators
- Hardware and software additions
- Illustrated throughout

AVAILABLE NOW!

PRICE £9.95

ONE OF OUR BEST SELLERS

Bank Switching

Alan Wrigley and Glynn Clements show you how to use dual screens in your programs.

Bank switching is the term for a useful technique which can produce results not easily obtainable by other methods. It refers to the capability of the Archimedes to switch instantaneously between two completely separate screen displays, or banks of screen memory, hence the term. This technique is often exploited to reduce screen flicker in animations. While this article does not intend to delve too deeply into animation techniques, we will show an example of how bank switching might be used for this purpose, and also suggest some other ideas which you may wish to develop further for your own programs.

If you are familiar with the way a VDU screen works, you will know that the display is refreshed, line by line, at a frequency of 50Hz. Plotting a moving object involves removing it from the screen and then re-drawing it in its new position. As a result of this, it is possible that some of the screen lines will be refreshed part-way through the process of re-plotting. This may mean, for example, that half the object is on the screen at that instant, and half is not, giving rise to flicker. The larger the area re-plotted, the worse the effect may be.

Bank switching can overcome this source of flicker. The screen which is not currently being displayed is updated, and then the bank switching mechanism is invoked to switch to the new screen instantaneously. This gives a very smooth transition, and the technique has been used to great effect in a good many games. Note, however, that there is a distinction between flicker and jerkiness. The latter will arise if the time taken to re-draw the screen is significantly long, and bank switching will not help to produce smoother movement if you are animating frames at a faster rate than they can be re-drawn.

There may be circumstances, however, where you wish to simply display complex graphics screens, without seeing them slowly re-drawn in front of your eyes. Using bank switching, you can draw on the undisplayed screen, and switch it in when the task is completed, thus giving a much neater effect to your graphic displays.

Yet another extremely useful application of bank switching is for debugging programs which rely heavily on screen output. If you wish to print out variables while a program is running, the resulting display will spoil the program's own screen output to some extent. However, using bank switching, it is possible to print out the values of variables on the alternative screen while the program is running, and switch screens when you wish to read them, for example by including an error handler which automatically switches in bank 2.

USING BANK SWITCHING

To use bank switching, you must first make sure your machine is configured with enough screen memory. If you are using an 80K screen mode, such as mode 12, you will need to have 160K of screen memory configured for two banks. There is, in fact, no reason why you should not have more than two banks, but you must configure enough screen memory for them all. To allocate more screen memory, click Menu on the Task Manager icon and select Task Display. Drag the red bar opposite Screen memory to the required size. You must do this each time you wish to run a bank switching program (assuming it needs more memory than is normally configured in your machine), since each time you return to the Desktop, the Task Manager will shrink the allocation back to the maximum size required for the current mode. Note that this method of memory allocation only works if you run the program directly from the Desktop, or by pressing F12 and then entering Basic. If you quit the Desktop completely using Ctrl-Shift-F12, screen memory will be allocated on the basis of the currently configured mode (*not* Wimpmode).

Listing 1 moves 16 coloured circles around the screen, removing and re-plotting each one in turn. Type in the listing and save it. Before it can be run, you will need to configure 160K of screen memory, as described above. Also, the program uses the standard Wimp 16-colour palette for the circles, so you must save a palette file, called *!Palette*, in the same directory as the program. This is done by clicking Menu over the palette icon,

selecting *Save*, and dragging the file icon to the directory viewer.

Initially the program does not use bank switching, and you will see the flicker clearly on some of the circles. Now alter line 140 to read:

```
B%=2
```

and run it again. This time the flicker has gone, thanks to bank switching. To see the effect of updating a large screen area without bank switching, restore line 140 to its original value, delete line 270, and replace line 240 with:

```
240 J%=0
```

Then insert the following line:

```
225 CLS
```

This clears the whole screen before re-drawing the circles, and the flicker is now very noticeable. If you now invoke bank switching by changing line 140 again, you will see a marked difference.

HOW IT WORKS

The technique makes use of two `OS_Byte` calls. `OS_Byte 113` determines which bank is actually displayed by the computer's hardware, while `OS_Byte 112` selects the bank which is written to by the VDU drivers. Each call takes a further parameter which specifies the bank number (1 for the first bank, 2 for the second and so on). You will see in listing 1 that `B1%` and `B2%` represent the two banks used. Lines 180-190 switch each variable alternately between a value of 1 and 2, and lines 210-220 use the `OS_Byte` calls described above. This results in one bank being written to while the other is displayed, and vice versa. The purpose of the `WAIT` statement in line 200 is to wait until the end of the current display frame before updating the screen. This maximises the time available while the electron beam is inactive and results in a smoother effect.

Note the error handler in line 150, and in particular the two `OS_Byte` calls. These are very important, as they ensure that the main screen bank is switched back in, for both output and display, if the program should crash for any reason. If this were not done, and the program terminated while output was being directed to a different bank from the one being displayed, you would not see the error, nor any subsequent commands that you typed in. Under these circumstances, effecting a mode change, by

typing `MODE12` for example, would restore the default screen bank.

PRE-DRAWN SCREENS

Listing 2 is a short program which uses bank switching to swap between complex screens without having to re-draw them. Four banks are used, so you must allocate 320K of screen memory to run the program. Screen 1 is used to display a menu containing names of geometric shapes. The other three screens, containing a selection of random squares, circles and triangles respectively, are drawn once only at the start of the program. When an item is selected from the menu, the appropriate screen is switched in instantly. Using only one screen bank, it would take approximately a quarter of a second to re-draw each screen when selected from the menu.

DEBUGGING

Listing 3 shows how you can use bank switching as a debugging aid. Each time the value of a variable is to be printed on the alternative screen, the program calls `PROCvar`, with a single parameter containing the expression whose value is to be printed. `PROCvar` then switches screen banks, prints the value, and switches them back again. Be careful not to send too much output to screen 2, since if it scrolls you will have lost some of the information by the time this screen is switched in. An error handler is set up in line 100, which calls `PROCerror` to switch to the alternative screen if the program terminates with an error. You can force an error to occur at a specific point in your program by including the statement:

```
ERROR 255,<errstring>
```

where *errstring* can be any string you choose, such as "Program terminated".

It is also possible to display the values while the program is running. If you insert the following lines into listing 3, then each time `PROCvar` is called it will temporarily halt the program and display the alternative screen until the space bar is pressed:

```
1005 *FX113,2
1023 PRINT"Press space bar"
1026 REPEAT UNTIL GET=32
1035 *FX113,1
```

You will no doubt be able to think of other uses for bank switching. If you

discover any interesting uses for the technique, let us know and we may be able to publish them in our Hints and Tips columns.

Listing 1

```
10 REM >Circles
20 REM Using bank switching
30 REM for simple animation
40 :
100 MODE15
110 MODE12
120 *Print !Palette
130 B1%=1
140 B2%=1:REM Set to 2 for bank switch
ing
150 ON ERROR SYS "OS_Byte",112,1:SYS "
OS_Byte",113,1:REPORT:PRINT" at line ";E
RL:END
160 REPEAT
170 FOR A%=0 TO 355 STEP 5
180 B1%=B1% EOR 3
190 B2%=B2% EOR 3
200 WAIT
210 SYS "OS_Byte",113,B1%
220 SYS "OS_Byte",112,B2%
230 FOR I%=0 TO 15
240 FOR J%=-10 TO 0 STEP 10
250 GCOL 0,I%*-(J%=0)
260 CIRCLE FILL (I%MOD4)*250+265+1
00*SIN(RAD(A%+J%)),(I%DIV4)*250+137+100*
COS(RAD(A%+J%)),100
270 NEXT J%
280 NEXT I%
290 NEXT A%
300 UNTILFALSE
```

Listing 2

```
10 REM >Shapes
20 REM Using bank switching to switch
30 REM between pre-drawn screens
40 :
100 MODE 12
110 *Print !Palette
120 ON ERROR SYS "OS_Byte",112,1:SYS "
OS_Byte",113,1:REPORT:PRINT" at line ";E
RL:END
130 PROCdrawscreen
140 VDU28,33,22,57,9
150 PRINT"SHAPE MENU"'"1. Squares"'
"2. Circles"'"3. Triangles"'"4. Quit"'
'"Select number"
160 REPEAT A%=GET-48
170 IF A%>0 AND A%<4 THEN
180 SYS "OS_Byte",113,A%+1
```

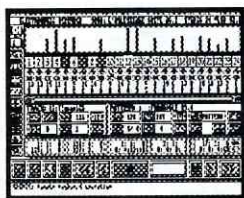
```
190 REPEAT UNTIL GET=32
200 *FX 113,1
210 ENDIF
220 UNTIL A%=4
230 VDU26,12:END
240 :
1000 DEF PROCdrawscreen
1010 FOR I%=2 TO 4
1020 SYS "OS_Byte",112,I%
1030 CLS
1040 FOR J%=1 TO 50
1050 GCOL RND(15)
1060 CASE I% OF
1070 WHEN 2:RECTANGLE FILL RND(1279),RN
D(1023),RND(500)
1080 WHEN 3:CIRCLE FILL RND(1279),RND(1
023),RND(300)
1090 WHEN 4:X%=RND(1279):Y%=RND(1023)
1100 MOVE X%,Y%:x%=RND(400):y%=RND(400)
1110 MOVE X%-x%,Y%-y%
1120 PLOT 85,X%+x%,Y%-y%
1130 ENDCASE
1140 NEXT J%
1150 PRINTTAB(25,15)"Press space to ret
urn to menu"
1160 NEXT I%
1170 *FX 112,1
1180 ENDPROC
```

Listing 3

```
10 REM >Debug
20 REM Using bank switching
30 REM to debug programs
40 :
100 ON ERROR PROCerror
110 REM Main program
120 REM To print variable values,
130 REM call PROCvar(variable)
140 REM END
150 :
1000 DEF PROCvar(x%)
1010 *FX112,2
1020 PRINT x%
1030 *FX112,1
1040 ENDPROC
1050 :
1060 DEF PROCerror
1070 *FX 113,2
1071 *FX 112,2
1080 REPORT:PRINT" at line ";ERL
1090 PRINT"Press space bar"
1100 REPEAT UNTIL GET=32
1110 *FX 113,1
1111 *FX 112,1
1120 END
```

RU

ARCHIMEDES MUSIC SOFTWARE



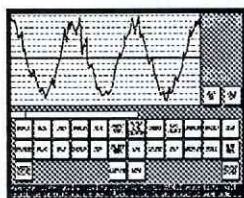
★ **STUDIO 24 PLUS Version 2 - The acclaimed software!**
★ **Acorn User Best Music Software Award 1989/90**

The ultimate recording studio for Midi instruments & ARC internal sounds! Innovative features include score, grid, event & drum edit, remote control, music video production + SMPTE control built-in, realtime notation & graphic displays for record and playback, auto lyrics, copy, quantise & sysex windows, master keyboard, multi-tasking. 64 channel Midi control. and much more! Price only £169



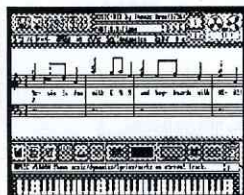
★ **SCOREWRITER PMS - Now for any dot-matrix printer!**

Incredible quality printout from this professional typesetting program using text input, with notation on-screen before printing. Also takes Studio24Plus V2 music for ultra-fast conversion to full score. Produces virtually any kind of orchestral scoring, with vocal lyrics, different fonts and a vast set of music shapes to complete the full music page. All for £149! (Postscript publ. version also available).



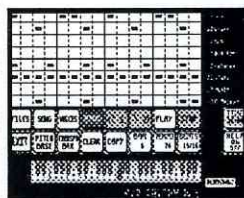
★ **SOUNDSYNTH - The best sounds for ears in years!**

The only sampling system for EMR's programs and also usable in your own programs or Maestro - lets you record ANY sound, ANY musical instrument, ANY voice, for adding up to 32 real instruments and fabulous effects to your music. Create sounds by harmonics, equations, random, mouse and sampling, reverse, echo, slapback, overlay, append, looping, interpolate etc. Complete system with top quality EMR Sampler 8 hardware for mic/tape input at only £126.50



★ **MICROSTUDIO - Learn to play any piece of music!**

A NEW kind of music software with a great micro studio for recording Midi instruments and a remarkable teaching system in one program. Use with ANY Midi keyboard for record/play plus Arc sounds - ideal for beginners. Get the family to learn your recorded music or try EMR's music from Peer Gynt to Ghostbusters! Watch the music, keys, add lyrics & printout parts - addictive learning! £79



★ **RHYTHMBOX - There's nothing like this music player!**

Fast, powerful program for playing its own ARC internal sounds, others created with SoundSynth or ready-made from EMR's Creations discs. Choose 8 new sounds with stereo pan and volume for ANY bar, click on note boxes in the simple on-screen bar grid, set pitch from screen keyboard, and arrange bars into a sequence with unlimited repeat loops. The easy way to compose for £29.95!

Other software: **Midi Analyser** - give your Midi keyboard a professional checkout! £29.95, **Creations Sample Discs** - huge library of orchestral inst, animals, voices, percussion and effects for Arc internal playback - 2 for £19.90. **VuMusic** graphics via Midi only £14.95. **Symphony** music classics & demo discs £9.95 each. **Daton Star** modern music library sets £29.95. **Assoc. Board** Piano Grades on disc + accomp. for inst. grades £14.95. Plus all **Midi, Video, SMPTE Hardware** - Acorn A3000 Int.Midi/User 16ch., Midi 16ch. 300/400 series, EMR Midi 2 32ch, Midi 4 64ch, instruments, cables, Arc sound connectors & more! Full details from EMR including **training courses**. All prices inc VAT. Add £2.50 Postage.

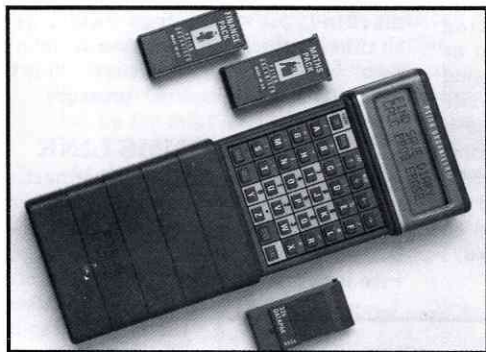
EMR LTD 14 Mount Close, Wickford, Essex SS11 8HG. Tel. Sales: 0702 335747

Enjoy Music Creativity with EMR!

Arc to Psion

In the first of two articles on using the Arc in conjunction with small portable computers, Lee Calcraft connects his Archimedes up to a Psion Organiser.

Most readers will have come across the Psion Organiser II in one guise or another. It is sold in chain stores such as Dixons for use as a computerised pocket Filofax; and is used in large multiples such as Marks and Spencers for obtaining the prices of goods, given a product code number, entered either at the keyboard or with a bar code reader.



In spite of its small size (it measures just 5.5 x 3 x 1 inches, and weighs 8.8 oz without battery), the Organiser is a complete computer in its own right, driven by a Hitachi HD6303X CMOS processor at 1MHz. It is powered by a PP3 9 volt battery giving between 2 and 6 months service, and comes in four different models, which range in price from £79 to £189 inc. VAT. The two models at the lower end of the scale (the CM and XP) have a two-line LCD display of 16 characters per line, while the more recent LZ models have a four-line display of 20 characters per line. Both displays are clear and easy to read (though neither has descenders for lower case letters such as p, q etc. - and these are displayed raised above the base line). Another variable is the amount of on-board RAM. This ranges from 8K in the cheapest model to 64K in

the most expensive. For the purposes of this review, I used a 64K LZ model; though most facilities discussed here are available on the XP and both LZ models. The CM offers a reduced spec, but can still be used with the important Comms Link - see later).

All models have two small recessed sockets for add-on memory. This can take the form of Rampaks (supplying up to 64K of RAM each), or Datapaks (of up to 128K each). You can also plug ready-programmed applications into these two sockets, choosing from titles such as the Finance Pack, Maths Pack, Spelling Checker and so on. There is also a Pocket Spreadsheet pack which is file-compatible with Lotus 123.

BUILT-IN APPLICATIONS

When you switch on the Organiser you are presented with a menu of applications. These include a database, a notebook (LZ models only), diary with optional alarms for each entry, calendar, clock, alarms and a sophisticated calculator. Each is easy to use, and where necessary provides further sub-menus.

There are three further items on the main menu which will be of interest to the serious user: *Xfiles*, *Utils*, and *Prog*. The first allows you to use the built-in database with selectable files, rather than using the *Main* default file. *Utils* contains a number of utilities for cataloguing a RAM drive, customising the sound output, copying and deleting files etc.

The *Prog* menu entry opens up the world of OPL programming, using the OPL language compiler. OPL looks much like a dialect of Basic, and contains a number of very useful functions for dealing with data files. It is strongly

procedural, and the built-in compiler is extremely easy to use. The main inconvenience of OPL is that each procedure or function in a program must be compiled separately. Thus you type in a main procedure, compile and save it, repeating the process for sub-procedures. Once you have done this, the program may be run from within the compiler front end, or installed on the Organiser's main menu (at whatever position you choose), and will run just by moving the cursor to it and pressing the *Execute* key.

Of course, while the keyboard is ok to use for small amounts of data entry (its non-qwerty layout takes some getting used to), for any lengthy entry, such as writing an extensive program or creating a large database, it is much better to write a text file on the Arc, and then transfer it using the Organiser's (optional) Comms Link package. And for serious program development Psion can supply a cross compiler which will run on a PC (and on the Arc with the PC Emulator installed, I understand).

programs. More interestingly, it allows the Organiser to be used as a portable data-gathering terminal for the Arc. At one level, notes and diary entries typed into the Organiser can be uploaded into the Arc, while at another, data capture devices such as bar code readers or temperature sensing devices can be used on the Organiser, and the data bulk-uploaded at a later date.

The Organiser has a large user base, and there are many companies supplying software and hardware add-ons for it. Digitron Ltd., for example, produce a data logger unit which plugs into the top of the machine, providing four data logging channels which can be used to monitor raw voltage, temperature, relative humidity and atmospheric pressure.

USING THE COMMS LINK

The Organiser achieves connectivity through the use of the optional Comms Link unit. This comprises a small plastic box about two inches square which plugs into a recessed socket at the top of the

Organiser, conferring the RS232 standard. The lead emerging from the Comms Link unit terminates in an IBM PC XT compatible 25 pin D connector. Adaptors are available to convert this for use with the PC AT and the Mac, but Archimedes users must either wire up their own adaptor (see figure 1), or have one made up by a dealer (see end of article).

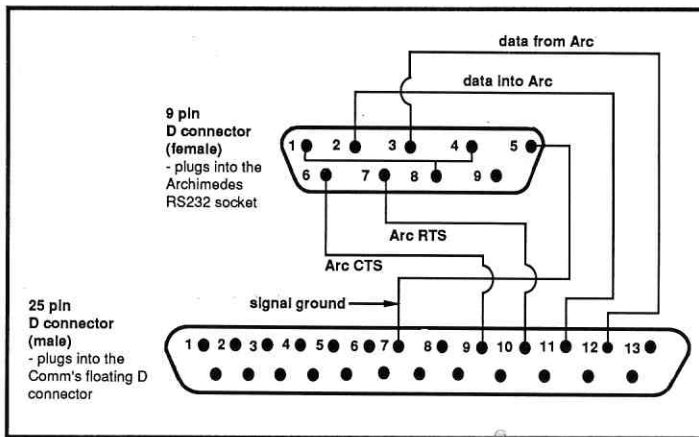


Figure 1. Wiring of adaptor for use with the Comms Link on the Archimedes. Both plugs viewed from the rear

The Comms Link can of course also be used to send data from the Organiser back to the Arc. This is useful for making permanent backups of both data and

The Comms Link unit contains extensive ROM-based comms software which is automatically downloaded into the Organiser at power up. Once this is installed, a new entry

Comms appears on the Organiser's main menu, and selecting this gives a 7-choice comms menu. This includes options to enter the terminal emulator, to transmit a file, receive a file, set up comms parameters, and something called *Auto*. This is a neat routine for discovering the comms settings of a distant computer. It cycles through all the possible combinations of baud rate and data bits, sending a brief message at each setting. All you need to do then is to check the screen of the distant terminal for a readable message, and you will know what setting to use.

In any use of the serial port on the Archimedes or any other computer, it is essential to set the communications protocol to exactly match the distant computer. On the Organiser, this can be performed using the *Setup* option from the *Comms* menu. For communicating with the Arc, I used the Arc's default of 1200 baud (though both Arc and Organiser will happily operate at 9600 baud), 8 bits and 2 stop bits with parity disabled. Finally, I set handshaking to *RTS*, and Protocol to *None* (Xmodem, and a special Psion protocol are the other two options). A special option allows these settings to be saved for future use.

The Organiser is now set up to both send and receive files, and to operate in terminal emulation mode. But it is not so simple with the Arc. You need to install some appropriate software at the Arc end of things. You can for example use BEEBUG's Hearsay, or one of the other comms packages for both file transfer and for keyboard to keyboard communication in terminal emulation mode. But it is also possible to write a short program on the Arc to achieve the same ends.

One of the Organiser's many strengths is its built-in database system. This has full searching, sorting, and editing options, and to test the machine I wrote a short program on the Arc to download a file containing the whole index to the *Archimedes Programmer's Reference Manual* (currently supplied with ArcScan II). The downloading software (which is supplied on this month's magazine disc) adjusts field terminators to the ASCII 9

character preferred by the Organiser, replaces the Arc's underline character with a minus sign (since although the Organiser displays it correctly, you cannot enter the underscore character in a search statement from the Organiser's keyboard), and removes the word *Page(s)* from each reference, to trim the length of the database.

```
REM OPL EXAMPLE PROGRAM
LOCAL Z
DO
  CLS
  PRINT "Multiplies by 2"
  INPUT Z
  PRINT "Result=";2*Z
  PRINT "Space, or Q to Quit"
UNTIL GET$="Q"
```

Figure 2. A short example of an OPL program. The DO statement is the exact equivalent of Basic's REPEAT

Downloading the 55K file took more than half an hour to a 128K EPROM Datapak in the Organiser (downloading it to a RAM file would have been very much quicker). It was then ready for use. So to find the SWI number for SWI "Wimp_CreateIcon", just select *Find* from the main menu, enter "WIMP-CRE" (case ignored), and press *Execute* - you only need the first few characters of a target string. Within two or three seconds, the Organiser had found the entry, displaying both the SWI number, and the page reference clearly on the screen. Pressing *Execute* again finds the next match: "Wimp-CreateMenu", and so on. On another occasion some time ago, I downloaded the BEEBUG retail catalogue into an Organiser model XP in a similar way. It proved to be an excellent way of looking up the price and product code of any named item, giving results in a matter of seconds.

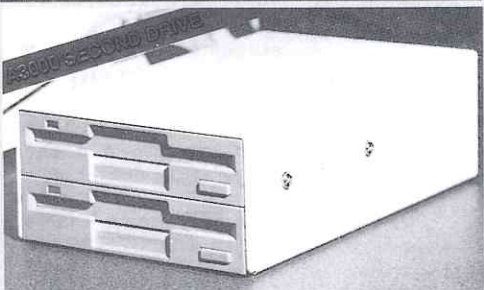
Continued on page 26



THE WORLDS MOST POWERFUL MICRO

A3000	£649	410/1	£1199
310	£899	420/1	£1699
310M	£959	440/1	£2499

**SPECIAL OFFER UNTIL JANUARY 1990
BUY AN A3000 OR ACORN'S DESKTOP
PUBLISHER AND SAVE 20% OFF THE
COST OF A MORLEY RAM BOARD**



THE A3000 OPTION ACORN FORGOT

- Easy to install simply plug in and run
- Includes full fitting kit and instructions
- High quality Japanese Drive
- Dual Drive Kit** £125

LEADS & MISC. HARDWARE ETC.

Arch Monitor Leads:		Printer Stand 80 col	£15
SCART	£8	Printer Stand 135 col	£19
9 Way 'D' (Msync)	£8	Tilt & Swivel Monitor Stand	£11
Arch Serial Lead	£8	Monitor Stand/plinth	
Arch Printer Lead	£6	BBC	£13
BBC Printer Lead	£6	Master	£14
2 to 1 Printer switch	£20	BBC/Master SCSI	
3 to 1 Printer switch	£30	Hard Disc Interface	£50
4 to 1 Printer switch	£36	Eprom Eraser	£18
64k Printer Buffer	£115	Eraser with timer	£23
256k Printer buffer	£199		

(NOTE switching units & print buffers come with printer lead)

The Morley TELETEXT ADAPTER is now available for the Archimedes. The specification including screen dumps etc., is the same as the BBC version but connection is via a Podule socket. £125

EXISTING USERS OF OUR BBC TELETEXT ADAPTER WE CAN UPGRADE THE UNIT FOR THE ARCHIMEDES FOR £35

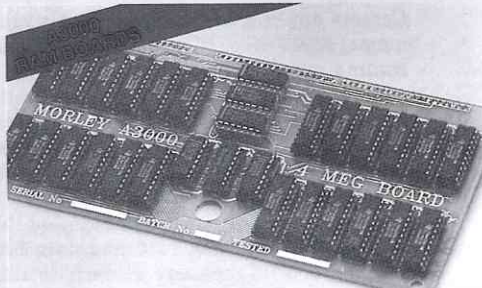
RING FOR DETAILS

We are now an Acorn dealer and service centre. If you would like to have a look at the A3000 or any other Acorn/Morley products please feel free to call in and discuss your requirements with our staff.

COMING SOON

Morley Module Podule

Allows frequently required 'Modules' to be loaded into a non-volatile storage area on the podule. Thus saving the tedium of searching through your discs for the modules required.

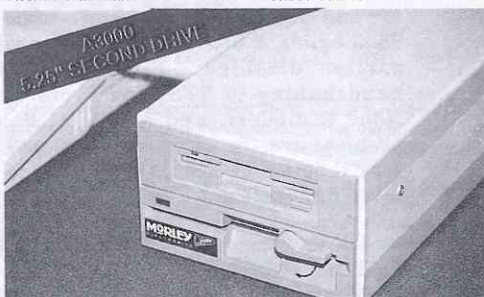


THE BEST VALUE MEMORY UPGRADES

- Easy to install simply plug in and run
 - Fully compatible with all Acorn software including desktop publisher
 - A must for all serious Archimedes owners
 - Make full use of Acorn's powerful Desk-Top Publishing package
 - 1 Meg upgrade gives you a total of 2 Meg on-board
 - 1 Meg board upgradeable to 3 Meg
- | | |
|------------------|------|
| 1 Meg Upgrade | £150 |
| 3 Meg Upgrade | £415 |
| 1 Meg to 3 Meg | £298 |
| 305 1/2 Meg | £105 |
| 400 Series 1 Meg | £145 |
| 400 Series 3 Meg | £395 |

ARCHIMEDES PERIPHERALS

Colour Monitor	£220	300/400 series	
Mono Monitor	£73	Acorn Rom Podule	£56
Printer Lead	£9	2 Slot Backplane	£37
A3000		4 Slot Backplane	£59
Monitor stand	£29	I/O Podule	£80
User port & MIDI Podule	£49	Midi Podule	£65
Serial Upgrade	£19	20 Meg Hard Disc & Controller	£465
Econet Upgrade	£49	Acorn DeskTop Pub.	£149
Pipedream 3	£147	Clares Pro Artisan	£145
Risc Os Companion	£49	Clares Artisan	£29
1st Word +	£69	Clares Toolkit	£34
Autodesk CAD	£79	Clares Toolkit +	£43
Acorn PC Emulator	£99		



Kit includes: 5.25" Teac 80 track drive; case; power supply; 3.5" drive cradle; interface and leads £169

A3000 SCSI PODULE £149

A3000 WINCHESTERS

Including SCSI Podule, drive, case and leads

20 meg	£465	30 meg	£495
40 meg	£575	60-200 meg POA	

TO ORDER . . .

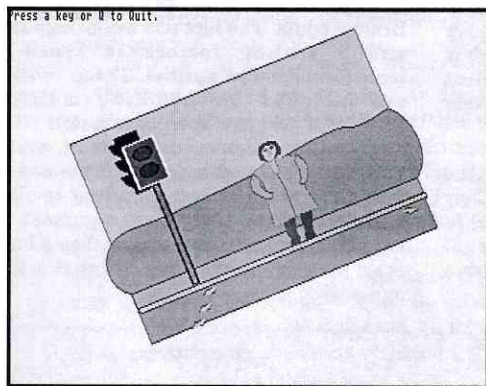
CARRIAGE: £1 on all orders. £3 on larger items (solitaires).
VAT: UK customers please add 15% to cost incl. carriage.
BY POST: Enclose cheque made payable to MORLEY ELECTRONICS LTD.

BY PHONE: By quoting your ACCESS or VISA card number.
Goods usually despatched by return.
Government and Education orders welcome.

Using Draw Files in Basic (Part 2)

Mike Summerton extends his Draw library to cover sprites, pictures and matrices.

The DrawLib library, started last month, allows objects created with the Draw application to be displayed and manipulated from within Basic programs. This month we shall add routines to the library for displaying Draw files containing sprites. Routines will be added which allow both Path and Sprite objects to be moved, scaled and rotated. A short demo program will illustrate this by displaying the contents of a Draw file, which can then be moved, scaled and rotated as specified by the user.



Sample Draw file scaled and rotated

All the routines described below need to be added to the DrawLib library listed last month. Make sure you keep exactly to the line numbers as shown, and the two parts should then match perfectly. The additions to DrawLib are contained in Listing 1.

The program *Transform* given in Listing 2 shows you how to use these routines. As last month it assumes that you have saved your palette as *Palette* and that it can load the ColourTrans module from Applications disc 1 if not already loaded. It also assumes the existence of a Draw file called *drawfile*, and displays the

contents of this when run. It then requests information about how the picture is to be manipulated (translation, scaling and rotation) and then displays the transformed picture.

Note that sprites as such will not be rotated by this version of DrawLib, only pictures including any sprites which may form part of the picture.

The complete DrawLib library is contained on this month's magazine disc together with the Transform demonstration program and a sample Draw file. The procedure PROCDrawLib_Help has also been updated here to provide information on the new features.

That completes the DrawLib library, though more routines to cope with further Draw objects could be added. Nevertheless, much can be achieved with the library as it is, and it should have given you some insight into what can be achieved by using Draw files from within Basic programs.

PROGRAM NOTES

Last month we developed a routine PROCObj_Display to display a Draw object on screen, but it only displayed Path type objects. Table 1 shows all the types of objects currently implemented in Draw files. This month's additions deal particularly with Sprite objects (type 5).

Object Type	Object Name	Description
0	Font Table	List of Font names and numbers
1	Text Object	Text (system or anti-aliased)
2	Path Object	As described last month
5	Sprite Object	Sprite in its usual form
6	Group Object	An object which contains other objects
7	Tag Object	An identifier to another object
9	Text Area Object	As created using Edit (see User Guide)
10	Text Column Object	Again as created using Edit

Table 1. Draw Objects

In fact, we have added routines for handling three new types of object, *Sprites*,

Pictures and *Matrices*, together with new procedures for handling Draw Objects, Areas, and Paths.

We need to add some code to PROCObjMod_Setup in order to set up some more workspace areas for handling sprites and matrices (lines 421 to 429). Note that the line in last month's listing:

```
330 DIM ObjMod_mat%PC% 24
```

is changed to:

```
330 DIM ObjMod_mat%PC% 32
```

to make more space for the matrix.

The PROCObj_Display routine (line 1040) must be changed to allow sprites to be displayed, by including additional lines inside the CASE statement (lines 1081 to 1083). A new routine, PROCObj_TransDisplay, is also required (lines 1113 to 1125) which is like PROCObj_Display, but which first transforms the object using the matrix which is the extra parameter.

One additional routine is added to the OBJECT handling section, PROCObj_Copy (source object, destination object, length) in lines 1127 to 1132. This is an internal routine which simply copies an object from a source buffer to a destination buffer.

Two further routines for handling Areas are given in lines 1503 to 1520. An Area object is a 4 word block which holds the co-ordinates of the bottom left and top right of an area of the screen (in Draw units). This is primarily used in the Wimp environment to decide whether a particular object is visible within the current window. FNArea_TransOverlap transforms a set of co-ordinates before deciding whether an object will be visible after the transformation. PROCArea_Change sets up an Area object to contain the 4 specified OS co-ordinates. Two additional Path handling routines, to display a transformed Path object, follow at line 1960.

We also require a new collection of procedures for handling sprites. These include PROCsprite_Display to display a sprite in its normal size and position, and PROCsprite_TransDisplay to display the sprite after transforming it using a matrix.

The first argument to both of these routines is a sprite. This is actually a pointer to a normal sprite block as used to store all sprites. The second argument is the *position* which is a pointer to a two word block holding the co-ordinates where the bottom left hand corner of the sprite is to appear (in normal OS units). The third argument in PROCsprite_TransDisplay is a *Transformation Matrix*.

A *Matrix* object is a block of memory holding information describing how to transform an object. The structure of a matrix is shown in Table 2. The first eight words are in exactly the same format as that required by the Draw_Fill and Draw_Stroke primitives in the RISC OS Draw module. The last two words hold the x and y scaling factors to speed up transformation of sprites. Three routines are included for handling matrices. PROCMat_Change accepts a matrix, the x and y scaling factors, the angle of rotation (anticlockwise) and x and y offsets and the matrix is set up corresponding to these transformations. If the first argument is a variable having the value zero, then a buffer is set up to hold the matrix before it is filled and passed back.

Word	Description
1	x scaling factor *COS angle to rotate (<<16).
2	x scaling factor *SIN angle to rotate (<<16).
3	y scaling factor *-SIN angle to rotate (<<16).
4	y scaling factor *COS angle to rotate (<<16).
5	x translation (<<16).
6	y translation (<<16).
7	x scaling factor (<<8).
8	y scaling factor (<<8).

Note that <<16 indicates that the two most significant bytes hold the integer part and the least two significant bytes hold the fractional part.

<<8 means that three bytes are used for the integer part and one for the fractional part.

Table 2. The Transformation matrix

Finally, there are two routines for handling a new object called a *Picture*. A picture is a collection of Draw objects which relate to each other. The contents of a Draw file become a picture once the file header is removed. The two routines PROCpic_Display

and PROCPic_TransDisplay display the contents of a picture in one go. The arguments are exactly analogous to PROCObj_Display and PROC_ObjTransDisplay.

Listing 1

```

10 REM >DrawLib2
20 REM Part 2 May 1990
30 :
330 DIM ObjMod_mat% 32
421 REM Set up sprite scale factors bl
ock
422 DIM ObjMod_scales% 16
423 !ObjMod_scales%=1
424 ObjMod_scales%!4=1
425 ObjMod_scales%!8=1
426 ObjMod_scales%!12=1
427 REM Set up pixel translation table
for colours of sprites
428 DIM ObjMod_ptt% 1024
429 DIM ObjMod_pa% 64
960 b%!4=1%:OSCLI("LOAD "+n$+" "+STR$(
(b%+8)):!(b%+1%+8)=0
1081 WHEN 5 : REM a Sprite
1082 IF p%=0 THEN p%=b%+8
1083 PROCSprite_Display(b%+24,p%)
1111 ENDPROC
1112 :
1113 DEF PROCObj_TransDisplay(b%,a%,p%,
m%)
1114 LOCAL t%
1115 IF FNArea_TransOverlap(a%,b%+8,m%)
THEN
1116 CASE !b% OF
1117 WHEN 2 : REM a Path
1118 PROCPath_TransDisplay(b%+24,p%
,m%)
1119 WHEN 5 : REM a Sprite
1120 IF p%=0 THEN p%=b%+8
1121 PROCSprite_TransDisplay(b%+24,
p%,m%)
1122 OTHERWISE : REM Do nothing
1123 ENDCASE
1124 ENDIF
1125 ENDPROC
1126 :
1127 DEF PROCObj_Copy(s%,d%,l%)
1128 LOCAL i%
1129 FOR i%=0 TO l%-1
1130 d%?i%=s%i
1131 NEXT
1132 ENDPROC
1133 :
1501 =TRUE
1502 :
1503 DEF FNArea_TransOverlap(a%,b%,m%)
1504 LOCAL v%,u%,sx%,sy%,bx%,by%
1505 v%=FNPath_TransPoint(b%,m%)
1506 u%=FNPath_TransPoint(b%+8,m%)
1507 sx%=v%!0:bx%=u%!0:sy%=v%!4:by%=u%!
4
1508 IF sx%>bx% THEN SWAP sx%,bx%
1509 IF sy%>by% THEN SWAP sy%,by%
1510 IF sy%>a%!12 OR by%<a%!4 THEN = FA
LSE
1511 IF sx%>a%!8 OR bx%<a% THEN = FALS
E
1512 =TRUE
1513 :
1514 DEF PROCArea_Change(RETURN a%,sx%,
sy%,bx%,by%)
1515 IF a%=0 THEN a%=FNBuf_Create(20)+4
1516 a%!0=sx%<<8
1517 a%!4=sy%<<8
1518 a%!8=bx%<<8
1519 a%!12=by%<<8
1520 ENDPROC
1950 :
1960 DEF PROCPath_TransDisplay(b%,p%,m%
)
1970 PROCObj_Copy(m%,ObjMod_mat%,24)
1980 PROCPath_Display(b%,p%)
1990 ENDPROC
2000 :
2010 DEF FNPath_TransPoint(p%,m%)
2020 LOCAL c%,d%
2030 DIM d% 8
2040 c%=FNBuf_Create(24)+4
2050 !c%=2:c%!4=!p%:c%!8=p%!4:c%!12=0:c
%!16=0
2060 SYS"Draw_TransformPath",c%,0,m%,0
2070 !d%=c%!4:d%!4=c%!8
2080 PROCBuf_Scrap(c%-4)
2090 =d%
2100 :
2120 REM *Object SPRITE*
2140 :
2150 DEF PROCSprite_Display(b%,p%)
2160 LOCAL i%
2170 FOR i%=0TO15:ObjMod_pa%!(i%*4)=!(b
%+4+i%*8):NEXT
2180 SYS"ColourTrans_SelectTable",b%!40
,ObjMod_pa%,-1,-1,ObjMod_ptt%
2190 SYS"OS_SpriteOp",52+512,b%,b%,(!p%
)>>8,(p%!4)>>8,0,0,ObjMod_ptt%
2200 ENDPROC
2210 :
2220 DEF PROCSprite_TransDisplay(b%,p%,
m%)
2230 LOCAL i%,d%

```

```

2240 FOR i%=0TO15:ObjMod_pa%!(i%*4)=!(b
%+44+i%*8):NEXT
2250 SYS"ColourTrans_SelectTable",b%!40
,ObjMod_pa%,-1,-1,ObjMod_ptt%
2260 d%=FNPath_TransPoint(p%,m%)
2270 PROCMat_Convert(m%,b%,sm%)
2280 SYS"OS_SpriteOp",52+512,b%,b%,(!d%
)>>8,(d%!4)>>8,0,sm%,ObjMod_ptt%
2290 ENDPROC
2300 :
2320 REM *Object MATRIX*
2340 :
2350 DEF PROCMat_Change(RETURN m%,sx,sy
,a,x%,y%)
2360 IF m%=0 THEN m%=FNBuf_Create(36)+4
2370 m%!0=FNMat_trunits(COS(RAD(a))*sx)
2380 m%!4=FNMat_trunits(SIN(RAD(a))*sx)
2390 m%!8=FNMat_trunits(-SIN(RAD(a))*sy
)
2400 m%!12=FNMat_trunits(COS(RAD(a))*sy
)
2410 m%!16=x%<<8:m%!20=y%<<8
2420 m%!24=FNMat_trunits(sx)
2430 m%!28=FNMat_trunits(sy)
2440 ENDPROC
2450 :
2460 DEF PROCMat_Convert(m%,b%,RETURN m
l%)
2470 LOCALsf%,pt%
2480 DIM sf% 16, pt% 16
2490 IF m1%=0 THEN m1%=FNBuf_Create(36)
+4
2500 REM Convert matrix for sprite scal
ing
2510 SYS"Wimp_ReadPixTrans",512,b%,b%,,
,,sf%,pt%
2520 m1%!0=(m%!24)*sf%!0
2530 m1%!4=(m%!28)*sf%!4
2540 m1%!8=410000*sf%!8
2550 m1%!12=410000*sf%!12
2560 m1%!16=m%!16
2570 m1%!20=m%!20
2580 ENDPROC
2590 :
2600 DEF FNMat_trunits(n)
2610 LOCAL t%
2620 t%=(INT(n))<<16
2630 n=(n-INT(n))*2^16
2640 t%=t% OR n
2650 =t%
2660 :
2680 REM *Object PICTURE*
2700 :
2710 DEF PROCPic_TransDisplay(b%,a%,p%,
m%)
2720 LOCAL Obj%

```

```

2730 Obj%=b%+44
2740 WHILE Obj%<0
2750 PROCObj_TransDisplay(Obj%,a%,p%,
m%)
2760 Obj%=FNObj_Next(b%,Obj%)
2770 ENDWHILE
2780 ENDPROC
2790 :
2800 DEF PROCPic_Display(b%,a%,p%)
2810 LOCAL Obj%
2820 IF a%=0 THEN PROCArea_Change(a%,0,
0,1280,1024)
2830 Obj%=b%+44
2840 WHILE Obj%<0
2850 PROCObj_Display(Obj%,a%,p%)
2860 Obj%=FNObj_Next(b%,Obj%)
2870 ENDWHILE
2880 ENDPROC

```

Listing 2

```

10 REM > RUDraw.TRANSFORM
20 *SET RUDraw$Dir $.RUDraw
30 LIBRARY "<RUDraw$Dir>.DrawLib"
40 MODE 12
50 PROCObjMod_Setup
60 wa%=0
61 PROCArea_Change(wa%,0,0,1280,1024)
80 file%=FNFile_Load("<RUDraw$Dir>.dr
awfile",b%)
90 IF file% THEN
100 mat%=0
110 PROCPic_Display(file%,wa%,0)
120 REPEAT
130 INPUT"Enter x and y coordinate
s for picture eg 400,100)",x%,y%
140 INPUT"Enter angle of rotation
eg 45",a
150 INPUT"Enter x scaling factor e
g 0.8",sx
160 INPUT"Enter y scaling factor e
g 0.8",sy
170 IF sy=0 THEN sx=1 :REM a sensi
ble default
171 IF sy=0 THEN sy=1
180 CLS
190 PROCMat_Change(mat%,sx,sy,a,x%
,y%)
200 PROCPic_TransDisplay(file%,wa%
,0,mat%)
210 PRINT "Press a key or Q to Qui
t.":A=GET
220 CLS
230 UNTIL A=ASC("Q") OR A=ASC("q")
240 PROCBuf_Scrap(b%)
250 ENDIF
260 END

```

RU

Andrew Benham supplies some more advice for users of the PC Emulator.

EDITING TEXT FILES

In the last *Bits and PCs* (Volume 3 Issue 4), mention was made of using a text editor to create certain customisation files. Unfortunately the only text editor supplied with the PC Emulator is the EDLIN line editor. For those unfamiliar with line editors, they tend to be far more unfriendly than screen editors (such as Twin and Edit), and EDLIN is no exception. The editor is invoked by typing EDLIN followed by a filename. This will produce a prompt of either:

```
New file
```

```
*
```

or:

```
End of input file
```

```
*
```

depending on whether EDLIN needs to create a new file, or is editing a file which already exists.

Editing commands may now be entered at the star prompt. All files are given line numbers by EDLIN, but these numbers are merely for the convenience of the editor and refer to the relative position of the line within the file. They are not intrinsic to the file, as line numbers are to a Basic program. All the time you are using the editor, it keeps a note of the "current line", which is the line that editing commands will work on unless other lines are explicitly specified in the command. The current line is marked by an asterisk after the line number, and to start with it will always be line 1.

The first command to use is L (list lines); up to 23 lines at a time can be listed, with the line numbers displayed. If the command is given without a line number, EDLIN will start listing at the current line, and will, if appropriate, list 11 lines on either side. You can specify a line number placed in front of the command, e.g. 25L, and EDLIN will list from there to the end of the file, or 23 lines

in total, whichever is less. Alternatively you can specify a range of line numbers, e.g. 10,20L. When the lines have been listed, the star prompt will be redisplayed.

To insert additional lines (and this applies also to starting an empty file), the insert command I is used. If no line number is given, the insert is made before the current line. If a line number is placed before the I, lines are inserted before the given number, which then becomes the current line. To insert text at the end of the file, replace the line number with a hash symbol (#). Having entered the insert command, the prompt will change to an indented

```
<line number>:*
```

New lines may now be entered, and will be inserted at the specified point in the file. To leave insert mode, type Ctrl-C at the start of the next line. The prompt will change to the previous *. The L command may be used again to list lines from the file.

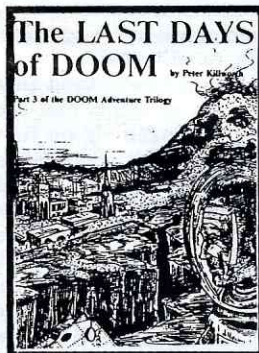
Finally, use the E command to end the edit, which automatically saves the modified file. If the editing has not gone to plan, use the Q command to quit the edit without saving the modified file (by responding with Y at the *Abort edit (Y/N)?* prompt).

EDLIN has other features: you will find these listed in any good MS-DOS guide. However, there are many screen editors available which are far more suitable than EDLIN as a general purpose editor, so it is seldom necessary to learn the full set of EDLIN commands.

HARD DISC NOTES

For those fortunate enough to have an Archimedes equipped with a hard disc, the PC Emulator may be used with a section of the hard disc (a partition) reserved for MS-DOS usage. A program

Classic Adventures, Games for Tinies - and Other Things...



LAST DAYS OF DOOM is now available.

Over 150K of text
and 218 rooms
await you! With
your three-man
crew and trusty
robot dog by your
side you'll have to

deal with squashed spaceships, renegade robots, slow beings - a whole host of original Killworth puzzles - in your quest to save Doom. In this final instalment of the Doom trilogy, the planet itself is doomed, and your mission is to save it! Pack includes Robot Dog Construction Kit and an enormous free game: *Hazarin*. Only £19.95 (inc VAT) plus £1.00 postage.

COMPILATIONS

AVON/MURDAC (Partington) £19.95. *COUNT-DOWN & RETURN TO DOOM, PHILOSOPHER'S QUEST* (Killworth) £19.95 *ACHETON, KINGDOM OF HAMIL* (Partington/Thackray) £19.95. Titles which were originally released by Acornsoft have been expanded and enhanced for these RISC versions! Add £1.00 postage..

TINY LOGO

TINY LOGO and TINY DRAW are mouse-driven A3000 enhancements of Northamptonshire's popular *Nimbus* control and drawing packages - ideal infant/lower junior introductions to these vital N.C. skills. Price: around £20, as is FREDDY TEDDY (colours, sequencing and matching) and FREDDY'S ADVENTURE - full details on request.

GIANT KILLER EXTRA!

Our MATHS adventure *Giant Killer* is available for the A3000/310/400 series, price £17.50. NEW by popular demand is our *Giant Killer Support Disc* - a graphic introduction to the more complex puzzles in this best-selling adventure game, price £14.95 (inc VAT) plus £1.00 p & p. Both suit ages 9 to 14+.

STOCK PACK

A3000/310/400 series stock control and accounts for your small business or GCSE business or computer studies class. Includes disc and two manuals. £29.95 (inc. VAT) plus £1.00 postage.

ACCESS/VISA

All products are available direct from:

TOPOLOGIKA, PO Box 39, Stilton, PE7 3RL (tel) 0733 244682

(If your local store doesn't stock them, ask them why not!)



called HARDDISC is supplied on the ADFS disc of the PC Emulator, and this may be used to set up the partition on the hard disc. Two programs must then be run using the Emulator to set up the partition for MS-DOS.

The ADFS program HARDDISC creates a 10 Megabyte partition of the hard disc as standard; on a 20 Megabyte disc this may be regarded as too large a partition. The program has been written, complete with comments, in such a way as to allow this size to be easily modified, with certain restrictions. Examination of the program will show that, while the size of the file which HARDDISC creates can be modified, it must be an exact multiple of &8800 (34816) bytes.

The partition is installed by creating an area on the hard disc containing the entire space which has been specified for the DOS partition, and this area is seen by the ADFS as one file. The first MS-DOS program (FDISK) sets up this disc space with an MS-DOS filing system structure, and then the second program HDINSTAL copies files onto the new disc from the MS-DOS boot floppy disc. The FDISK program can cause confusion during the installation process. It asks the question *Do you wish to use the entire fixed disc for DOS?*, to which you should answer Y although it seems wrong (and inviting disaster!). However, FDISK only knows about the space previously reserved for the PC partition on the disc, and you will want MS-DOS to use all of this space. The formatting performed by HDINSTAL when it runs is also confined to the reserved space!

Assuming that you are running the latest version (1.33) of the Emulator (the version number is displayed when you boot up), then once the PC partition has been created, the hard disc structure for RISC OS can be tidied up by placing the file Drive_C inside the !PC application directory. The PC Emulator must be advised of where the partition is, and this

is done by using Edit to alter the final line of the file IPC.!Run2.

Once MS-DOS is operational on a hard disc, an irritating feature becomes apparent. Because of the need to run the Emulator ADFS disc first, the prompt *Insert MSDOS boot disk and press any key* is displayed. As MS-DOS is already present on the hard disc, this prompt is superfluous. It is, however, possible to modify the Emulator to remove it. With the Archimedes in RISC OS mode (i.e. not DOS), and the current directory set to the PC Emulator, using for example:

```
*DIR adfs::HardDisc4.$PC
```

enter the following Basic program. This will load the original file called Rom, make a safety copy of it called OldRom, modify the file, and save the modified file back as Rom.

```
10 REM >RomPatcher
20 DIM buffer &2000
50 SYS "OS File",&FF,"Rom",buffer,0
80 FOR loop%=&15D TO &174
90 buffer?loop%=&90
100 NEXT
130 SYS "OS_FSControl",25,"Rom",
"Oldrom"
160 SYS "OS File",&0A,"Rom",&FFD,,
buffer,buffer+&2000
180 END
```

From the next MS-DOS restart, the PC Emulator will start up and enter MS-DOS without interruption.

The program HDINSTAL copies all the files from the original MS-DOS disc to the root directory of the hard disc. This tends to make the root directory appear cluttered. It is generally easier to find your way around a hard disc if there are a few clearly-named directories, each containing perhaps a small number of related files. A common way of installing MS-DOS on a hard disc is to create a directory (using MKDIR) called DOS in the root directory. Most of the original files from the system floppy can be moved into

this directory, except the following which should remain in the root directory:

```
CONFIG.SYS
AUTOEXEC.BAT
COMMAND.COM
```

There are also two hidden files which must be in the root directory, but as the name suggests they are not normally visible to the user. These files are the MS-DOS system files, called IO.SYS and MSDOS.SYS, which will have been installed in the root directory by HDINSTAL. Once all the remaining files (the so-called *external commands*) have been moved into the DOS directory, they will not be found when the system searches for them unless the system is told where they are! This is achieved by using another very useful configuration command, PATH. The PATH command supplies a list of directories to search, in the supplied order, if a user command is not recognised by MS-DOS and is not the

name of a file in the current directory. Therefore if the files have been moved into the DOS directory on drive C, the command:

```
PATH C:\DOS
```

should either be entered at the keyboard or included in the AUTOEXEC.BAT file. More than one directory may be searched; additional directories can be added to the list, with each complete pathname separated by a semi-colon, for example:

```
PATH C:\DOS;C:\UTILS;C:\TOOLS
```

The current setting of the path list may be found by entering PATH with no parameters.

By keeping the hard disc structure tidy and using meaningful names, the task of finding particular files and directories becomes very much easier. This also applies to the RISC OS hard disc area - after 6 months, it is easier to remember what an obvious filename is for than a filename such as TEMP or FRED! RU

Arc to Psion (continued from page 17)

CONCLUSION

The Psion Organiser II complements the Arc in many ways, and the two machines work well together. The Organiser is extremely well made, and appears to be a highly reliable device. It is a powerful machine for its size, and contains a host of built-in applications. Its database handling is excellent, and it is easy to install databases generated on the Arc into a Datapak for rapid recall of a wide range of facts. It can also act as a useful hand-held data gathering unit, with the ability to upload its data to the Arc for analysis and permanent storage. The powerful OPL programming language gives added flexibility, allowing the user complete freedom to fully customise the machine to suit his particular needs. The only real drawback to this excellent little machine is in my view its lack of a qwerty keyboard.

Product Supplier	Organiser II Psion UK PLC, Alexander House, 85 Frampton Street, London NW8 8NQ. Tel. (071) 262 5580
Tech Help	Tel. (071) 262 8188

Prices (inc. VAT)	Organiser II £79.95-£189.95 Comms Link £59.95
--------------------------	--

For a useful book on the Organiser, see: Mike Shaw's *Using and Programming the Psion Organiser*, published by Kuma Computers at £9.95.

Other contacts:

Digitron (for the Organiser data logger) are on (0992) 587441.

Y2 Computing (for Organiser connectors) are on (0923) 50161.

Next month in this spot we will be taking a look at the use of the Cambridge Computer Z88 with the Arc. RU

Assembler Workshop - Quick Sort

Fred Hartley presents an ARM code routine to sort Basic arrays.

One of the most common operations in data processing is that of sorting a number of data items into some order, for example numeric or alphabetic. Indeed, RISC OS itself has a very flexible sort routine built into it, and made available to the user via a SWI call. However, this cannot be easily used to sort Basic arrays, and that is where the program presented here comes in, allowing all types of Basic arrays to be sorted, but using ARM assembler.

ENTERING THE PROGRAM

There are two programs given below. The main one, listing 1, is the source code for the ARM code sorting routine. This should be typed in and saved. When run, the sort routine is assembled and saved with the filename 'QsortAmc', though this can be changed by altering line 90.

The second program, given in listing 2, demonstrates the sorting routine. This uses the routine to sort random real, string and integer arrays of a chosen size. You can specify either a single run, or a number of runs, in which case the program prints out the average time for each type of array. The sorted arrays are also printed.

USING THE PROGRAM

To use the sorting routine in your own programs you must first load it. This can be done using the statements:

```
DIM sort 800
OSCLI "LOAD QsortAmc "+STR$~sort
These should appear in the initialisation routines of your program.
```

Any Basic array can now be sorted using a statement of the form:

```
CALL sort, name$( )
CALL sort, name( ) or
CALL sort, name%( )
depending on the type of the array. The empty brackets are essential to show that an array is involved.
```

QUICK SORT

There have been many sorting algorithms devised over the years, and it is not always an easy choice as to which to use. One factor is the nature of the data being sorted - many routines that perform well on average fall over when the data is already partly sorted, with perhaps only a few items out of order. Another factor is that some routines require a large amount of additional memory as workspace while sorting.

On the whole, the Quick Sort algorithm used here is one of the best, with an average speed proportional to the number of elements times the log to the base two of the number of elements. This is normally written as $O(N\log_2N)$, where the O reads 'is of the order of'. The log term makes Quick Sort particularly useful when a large number of elements are being sorted.

HOW IT WORKS

Next month's *Assembler Workshop* will explain in some detail how the sorting routine works, concentrating in particular on how the Basic arrays are accessed from machine code. In the meantime, this month's magazine disc contains an unabridged version of the program which has a large number of comments which help explain its operation.

Listing 1

```
10 REM >Qsort
20 REM Program Quick Sort
30 REM Version A1.0
40 REM Author F.W.Hartley
50 REM RISC User May 1990
60 REM Program subject to copyright
70 :
80 DIM sort 1024:PROcAss
90 OSCLI "SAVE QsortAmc "+STR$~sort+"
"+STR$~addr
```

```
100 END
110 :
120 DEF PROCass
130 array=0:size=1
140 I=2:J=3:shift=3
150 K=4:lhs=4:Lo=4
160 rhs=5:Hi=5:M=6:X=6
170 Iaddr=7:Ival=7:Xaddr=7
180 Xmant=7:Xword=7
190 Jaddr=8:Jval=8:Yword=8
200 Maddr=9:Mval=9:Mmant=9
210 Xlen=10:Xexp=10:Xbyte=10
220 Mlen=11:Mexp=11:type=12
230 sp=13:link=14:ctr=14
240 inx=14:pc=15
250 :
260 FOR pass%=0 TO 2 STEP 2
270 P%=sort
280 [OPT pass%
290 ; CALL sort,array_name()
300 ; sorts array elements
310 ; in alphabetical order
320 .sort STMFD (sp!),{link}
330 LDR type,[9,#4]
340 AND rhs,type,#&100
350 BEQ exit:LDR array,[9]
360 .sort1
370 LDR inx,[array],#4
380 CMP inx,#0:BNE sort1
390 LDR size,[array],#4
400 SUB size,size,#1
410 TEQ type,#&180
420 MOVEQ type,#3:BEQ begin
430 AND type,type,#7
440 TEQ type,#4:BEQ begin
450 TEQ type,#5:BEQ begin
460 B exit
470 ;-----
480 .begin
490 MOV Lo,#0:MOV Hi,size
500 BL main
510 .exit LDMFD (sp!),{pc}^
520 ;-----
530 .main
540 STMFD (sp!),{link}
550 MOV I,Lo:MOV J,Hi
560 ADD M,I,J:MOV M,M,ASR#1
570 TEQ type,#3:BEQ ssorta
580 TEQ type,#4:BEQ isorta
590 B rsorta
600 ;-----
610 .ssorta
620 MOV X,M:BL fetchX
630 MOV Maddr,Xword
640 MOV Mlen,Xbyte
650 ;-----
660 .ssortb SUB I,I,#1
670 .ssortc ADD I,I,#1
680 MOV X,I:BL fetchX
690 BL compareS:BLT ssortc
700 ADD J,J,#1
710 .ssorted SUB J,J,#1
720 MOV X,J:BL fetchX
730 BL compareS:BGT ssorted
740 CMP I,J:BGT ssorte:BL swop
750 ADD I,I,#1:SUB J,J,#1
760 .ssorte CMP I,J:BLE ssortb
770 B main3
780 ;-----
790 .isorta
800 LDR Mval,[array,M,ASL#2]
810 .isortb SUB I,I,#1
820 .isortc ADD I,I,#1
830 LDR Ival,[array,I,ASL#2]
840 CMP Ival,Mval:BLT isortc
850 ADD J,J,#1
860 .isortd SUB J,J,#1
870 LDR Jval,[array,J,ASL#2]
880 CMP Jval,Mval:BGT isortd
890 CMP I,J:BGT isorte
900 STR Ival,[array,J,ASL#2]
910 STR Jval,[array,I,ASL#2]
920 ADD I,I,#1:SUB J,J,#1
930 .isorte
940 CMP I,J:BLE isortb
950 B main3
960 ;-----
970 .rsorta
980 MOV X,M:BL fetchX
990 MOV Mmant,Xmant
1000 MOV Mexp,Xexp
1010 .rsortb SUB I,I,#1
1020 .rsortc ADD I,I,#1
1030 MOV X,I:BL fetchX
1040 BL compareR:BLT rsortc
1050 ADD J,J,#1
1060 .rsorted SUB J,J,#1
1070 MOV X,J:BL fetchX
1080 BL compareR:BGT rsorted
1090 CMP I,J:BGT rsorte
```

Assembler Workshop - Quick Sort

```
1100 BL swop
1110 ADD I,I,#1:SUB J,J,#1
1120 .rsorte CMP I,J
1130 BLE rsortb:B main3
1140 ;-----
1150 .main3
1160 ; the interval (Lo,Hi) has
1170 ; now been partitioned into
1180 ; two segments (Lo,J),
1190 ; (I,Hi) such that
1200 ; all of the array elements
1210 ; in the first are less than
1220 ; all of the array elements
1230 ; in the second
1240 ;
1250 CMP J,Lo:BLE main5
1260 STMFd (sp)!,{2-6}
1270 MOV Hi,J:BL main
1280 LDMFD (sp)!,{2-6}
1290 .main5
1300 CMP I,Hi:BGE main6
1310 STMFd (sp)!,{2-6}
1320 MOV Lo,I:BL main
1330 LDMFD (sp)!,{2-6}
1340 .main6
1350 LDMFD (sp)!,{pc}^
1360 ;-----
1370 .compareR
1380 STMFd (sp)!,{lhs,rhs,link}
1390 CMP Xexp,#0
1400 RSBEQ rhs,Mmant,#0
1410 BEQ cR1
1420 CMP Mexp,#0
1430 MOVEQ rhs,Xmant:BEQ cR1
1440 EORS rhs,Xmant,Mmant
1450 MOVLt rhs,Xmant
1460 BLT cR1
1470 SUB lhs,Xmant,Mmant
1480 SUB rhs,Xexp,Mexp
1490 CMP Xmant,#0
1500 RSBLT lhs,lhs,#0
1510 RSBLT rhs,rhs,#0
1520 CMP rhs,#0
1530 MOVEQ rhs,lhs
1540 .cR1 CMP rhs,#0
1550 .xx
1560 LDMFD (sp)!,{lhs,rhs,pc}
1570 ;-----
1580 .fetchX
1590 STMFd (sp)!,{shift,Yword,link}

1600 MOV inx,array
1610 ADD inx,inx,X,ASL#2
1620 ADD inx,inx,X
1630 LDRB Xbyte,[inx,#4]
1640 AND shift,inx,#3
1650 BIC inx,inx,#3
1660 LDMFD inx,{Xword,Yword}
1670 MOVS shift,shift,LSL#3
1680 MOVNE Xword,Xword,LSR shift
1690 RSBNE shift,shift,#32
1700 ORRNE Xword,Xword,Yword,LSL shift
1710 LDMFD (sp)!,{shift,Yword,pc}^
1720 ;-----
1730 .compareS
1740 STMFd (sp)!,{4,5,Maddr,link}
1750 TEQ Xlen,#0:BEQ cS2
1760 TEQ Mlen,#0:BEQ cS2
1770 CMP Xlen,Mlen
1780 MOVLE ctr,Xlen
1790 MOVGT ctr,Mlen
1800 .cS1 LDRB lhs,[Xaddr],#1
1810 LDRB rhs,[Maddr],#1
1820 CMP lhs,rhs:BNE cS3
1830 SUBS ctr,ctr,#1:BNE cS1
1840 .cS2 SUBS ctr,Xlen,Mlen
1850 .cS3
1860 LDMFD (sp)!,{4,5,Maddr,pc}
1870 ;-----
1880 .swop
1890 STMFd (sp)!,{4,5,link}
1900 MOV Iaddr,array
1910 ADD Iaddr,Iaddr,I,ASL#2
1920 ADD Iaddr,Iaddr,I
1930 MOV Jaddr,array
1940 ADD Jaddr,Jaddr,J,ASL#2
1950 ADD Jaddr,Jaddr,J
1960 MOV ctr,#4
1970 .s1
1980 LDRB lhs,[Iaddr,ctr]
1990 LDRB rhs,[Jaddr,ctr]
2000 STRB rhs,[Iaddr,ctr]
2010 STRB lhs,[Jaddr,ctr]
2020 SUBS ctr,ctr,#1:BGE s1
2030 .s2 LDMFD (sp)!,{4,5,pc}
2040 ;-----
2050 .addr
2060 ]
2070 NEXT
2080 ENDPROC
```

New Outline Fonts for the Archimedes



- The new outline fonts are suitable for every Archimedes application incorporating the Font Manager: DTP packages, etc.
- Each font contains 215 letters and symbols: Latin 1 alphabet with DPT extensions
- All fonts are hinted
- A comprehensive manual supplied with each delivery
- Prices from £3.99 to £9.50 per font
- *Special discounts for schools*
- Site licences for business use

We also supply printers, computers and disc drives at extremely low prices. Probably the best printer at its price is Canon BJ-130 Bubble Jet used for this ad:

- *It has a resolution of 360 by 360 dots per inch*
- *It takes paper up to A3 size*
- *And we sell it for only £529.00!*

Latest: Aleph One ARM-3 available at £585.00

Prices are exclusive of VAT and delivery

Write for a **free 32 page Font Catalogue** to:

The Electronic Font Foundry

18 Brockenhurst Road • ASCOT SL5 9DL

Listing 2

```

10 REM                >QsortEx
20 REM Program      Quick Sort Demo
30 REM Version      A1.0
40 REM Author       F.W.Hartley
50 REM RISC User    May 1990
60 REM Program      subject to copyright
70 :
80 LOMEM=&C000:MODE 0
90 DIM sort 1024
100 SYS "OS_File",255,"QsortAmc",sort,
0
110
120 INPUT "Enter DIM of array: "Q%
130 DIM A$(Q%),A%(Q%),A(Q%)
140 Max%=1E9
150 INPUT "Number of runs: "NR%
160 IF NR%>1 PROCavg:END
170 PROCfill
180 PROCexec
190 PROCptim
200 INPUT "Print sorted arrays y/n: "y
es$
210 IF INSTR("Yy",yes$) PROCparr
220 END
230 :
240 DEF PROCfill
250 FOR I%=0 TO Q%
260 A(I%)=RND(1)-0.5
270 A%(I%)=RND(Max%)-Max%/2
280 L%=RND(7)
290 C$=CHR$(40+RND(26))
300 FOR J%=1 TO L%:C$=C$+CHR$(60+RND(
26)):NEXT
310 A$(I%)=C$
320 NEXT
330 ENDPROC
340 :
350 DEF PROCexec
360 TIME=0:CALL sort,A():TR=TIME
370 TIME=0:CALL sort,A$():TS=TIME
380 TIME=0:CALL sort,A%():TI=TIME
390 ENDPROC
400 :
410 DEF PROCavg
420 TSSum=0:TRsum=0:TISum=0
430 PRINT "      Run      Real
String      Integer"
440 FOR Z%=1 TO NR%
450 PROCfill:PROCexec
460 TRsum+=TR:TSSum+=TS:TISum+=TI
470 @%=&A:PRINT Z%,TR,TS,TI
480 NEXT
490 TRav=TRsum/Z%:TSav=TSSum/Z%
500 TIav=TISum/Z%
510 @%=&2020A:PRINT'TAB(10),TRav,TSav,
TIav
520 ENDPROC
530 :
540 DEF PROCparr
550 PRINT:FOR I%=0 TO Q%
560 @%=8:PRINT I%;TAB(10);
570 @%=&20F09:PRINT A(I%);TAB(25);
580 @%=15:PRINT A$(I%);TAB(35);
590 PRINT A%(I%):NEXT
600 ENDPROC
610 :
620 DEF PROCptim
630 @%=15
640 PRINT "Time in centisecs to sort
array of dimension ";Q%
650 @%=&2020A
660 PRINT "Integer: ";TI:TAB(20);
670 PRINT "Real: ";TR:TAB(40);
680 PRINT "String: ";TS
690 ENDPROC

```

RU

Napoleonic Battle Simulator

Refight famous historical battles of the Napoleonic era. Become the commander in the field and direct vast armies of infantry, cavalry and artillery.

- Multi-tasks in the Desktop environment.
- Two players or one player against the computer.
- Two historical scenarios - Waterloo and Maida. Each scenario includes the regiments involved on a scale map of the battlefield.
- Features include play-by-mail option, zoom in / out map, status details, four complexity levels, timer.
- Comprehensive rule booklet covering movement, combat, visibility and morale.

Send a cheque for £15-95 payable to Spearpoint or SAE for further information to

Spearpoint
3, Lawrence Crescent,
Edgware,
Middx, HA8 5PB.

Improving the Picture

Three monitors from Taxan reviewed by David Spencer

There must be a large number of Archimedes users who purchased their systems complete with the Acorn colour monitor (or even the original mono monitor), and who are now considering upgrading to a multi-sync unit to provide extra screen modes. In this article I will look at two popular multi-syncs, and a special high resolution mono monitor.



Taxan 870

Before looking at the individual monitors it is useful to explain the difference between a multi-sync monitor, and the standard type, such as the Acorn colour monitor and the Phillips CM8833. In nearly all monitors and TVs, the picture is formed by scanning an electron beam (or three in fact for a colour tube) across the screen in rows, moving it down in the process so as to cover the whole visible area. In a normal monitor, the number of times a second that the beam scans across, and the number of times it covers the entire screen per second are fixed at 15625 and 50 respectively. These are the same rates used in a normal television - a fact that was very important when computers could be used with either a proper monitor, or a normal TV. Unfortunately, the fixed scanning rate, and the hardware of the Archimedes, impose a restriction on the screen modes available.

With a multi-sync monitor, the fixed scanning rates no longer apply. Instead, the monitor can work with a wide range of scanning rates, and this allows higher resolution modes to be implemented. Other features often found on multi-syncs include better quality picture tubes and a higher video bandwidth, both of which give a sharper picture, particularly with fine detail.

THE TAXAN 770 AND 870

Probably the most common multi-sync monitor among Archimedes users is the 14 inch Taxan 770. There is also a 16 inch version of this - the Taxan 870.

The 770 and 870 utilise a ground glass screen, rather than the more normal glossy finish found in televisions and standard monitors. This offers the advantage that the ambient light falling on the screen is distributed, rather than forming a single patch. However, a disadvantage of the ground glass screen is that at close range, it appears that the picture is out of focus.

Both the 770 and 870 have plastic cases which closely match the Archimedes in colour. Both monitors come complete with a swivel and tilt stand which means that they can be placed on top of the Archimedes, and then the viewing position adjusted. A generous length monitor lead is also supplied together with a power cable which plugs into the IEC output on the back of the Archimedes. The instructions for using the monitors are minimal and tend to concentrate on PCs fitted with Taxan's own graphics cards. However, there is enough information to install and use the monitor properly.

Taxan also offer the 770 in a low radiation version. With any monitor, the electron beams inside the tube produce X-rays when they hit the inside of the screen (or to be precise the aperture grill just behind the screen). These X-rays stream out of the front of the monitor, and could be considered a health hazard, particular

when the user sits only a couple of feet from the screen. The long-term effect of this type of X-ray exposure has yet to be determined. However recent research (published in the February 1990 issue of *Wireless World* magazine) suggests that the electric and magnetic fields produced by the scanning coils in a monitor can pose a serious threat to health, and these fields are also reduced in the low-radiation monitor. One useful side effect of the low radiation version of the 770 is that the build up of dust on the screen is greatly reduced.

THE 770 AND 870 IN USE

As soon as you switch on either of these monitors you notice an immediate improvement over a standard monitor. Even in a non-multi-sync mode, the whole picture is much sharper. One disadvantage of this, however, is that you see a lining effect, with thin black lines between each row of pixels. However, you soon get used to this. If you now go into the Desktop and select mode 20 (the multi-sync version of mode 12) you will wonder how you ever managed without a multi-sync. The entire picture appears much stronger and smoother. The picture is also larger as the border area around the outside is reduced to only a thin line. From outside the Desktop, mode 20 offers 64 lines of text per screen rather than the 32 of mode 12. This can be very useful when using a text editor such as *Twin*, or the *Basic Editor*, as you can view twice as much at a time.

The 770 and 870 do however have some drawbacks. Firstly, unlike an ordinary monitor, they need to be carefully set up. This is done using a number of controls on the back panel (or the front in the case of the 870). Although the instruction manual doesn't mention the *Archimedes* at all, it does contain enough information to make the set-up process relatively straightforward. One important point though is that the monitor should be turned on for at least half an hour before any adjustments are made, otherwise you can end up with a flickering picture.

Another problem, which appears to affect all multi-syncs, not just the 770 and 870, is that after several hours of viewing a multi-sync mode, your eyes are very tired and cannot focus properly. It is almost as if the multi-sync modes offer too high a resolution and the eye cannot cope. This problem is compounded if fluorescent lighting is used, as this causes a high frequency flicker on the screen. It is possible at this may be due to the ground glass screen.

Another common complaint about the 770 and 870 is that the picture changes in size when the display changes from predominantly black (for example outside the Desktop) to white or grey (in the Desktop for example). This is a failing of the power supply in the monitor. Luckily though, such sudden changes of display are rare, and hence this is more of a minor inconvenience than a serious problem. It must also be said that most monitors suffer from this problem to some extent.



Taxan 775

THE NEW TAXAN 775

By the time you read this review, the current 770 range will have been superseded by the newer 775. Unfortunately, one of these new monitors was not available for review. The 775 will look the same as the current 770, and should be priced the same. It will however feature an improved picture tube offering a reduced dot pitch, and also a better power supply to prevent the breathing

effect described above. Table 1 gives the basic specification for the Taxan 770, 870 and 775.

Monitor	770	870	775
Screen size (inch)	14	16	14
Dot pitch (mm)	0.31	0.31	0.28
Bandwidth (MHz)	>30	>30	>30
H. Scan rate (KHz)	15-35	15-37	15-37
V. Scan rate (Hz)	50-90	50-90	50-90
Width (mm)	356	385	356
Height (mm)	349	350	349
Depth (mm)	398	420	398
Weight (Kg)	14.7	18	14.7

Table 1. Monitor specifications

THE VIKING TWO

This beast really is a monitor and a half. Unlike the other monitors in this review, the Viking 2, also from Taxan, is not a multi-sync monitor. It is instead a special high resolution black and white monitor. The Viking 2 also differs from other monitors in that rather than increasing the range of available modes, it reduces it to just one - mode 23.

One very important point is that the Viking 2 is a black and white, and not a monochrome, monitor. Whereas a monochrome monitor can display black and white and all the intermediate shades of grey, the Viking 2 can only display black or white.

The Viking 2 can only be used on Archimedes that are equipped with special circuitry to allow mode 23 to be used. This is fitted as standard to the original A440 and to the new 400/1 series. However, the extra circuitry is not available on the A300 series, nor on the A3000. Neither can it be added to these machines at a later date.

THE VIKING 2 IN USE

The Viking 2 offers a display resolution of 1152 by 896 pixels. This is excellent for DTP work where almost two full A4 pages can be displayed at a time. Similarly, CAD applications benefit from the increased display size. Furthermore, being a black and white monitor, the display is very sharp and individual pixels can easily be distinguished.

Unfortunately, the Viking 2 does have some drawbacks. An immediate and insurmountable problem is that there are a large number of software packages around that will only work in chosen modes, and mode 23 generally isn't one of them. Typical of this class of programs are games, which normally run in mode 12 or 15. Also high on the list of non-starters are some of the most popular art packages. For example, Artisan is restricted to mode 12, while ProArtisan and Atelier will only work in mode 15. On the plus side, any Desktop applications should work properly in mode 23.



Viking 2

A related problem lies in RISC OS itself. As only black and white are available, dot patterns must be used to represent grey shades, just as in mode 0 in the Desktop. Unfortunately, RISC OS falls down sometimes on this, with effects such as thin lines vanishing. Hopefully any new version of RISC OS will address this problem and improve the situation.

PRICES

All the monitors featured in this article are available from Beebug, at the prices given below. All prices include VAT, and those in brackets are for non-members of RISC User or Beebug.

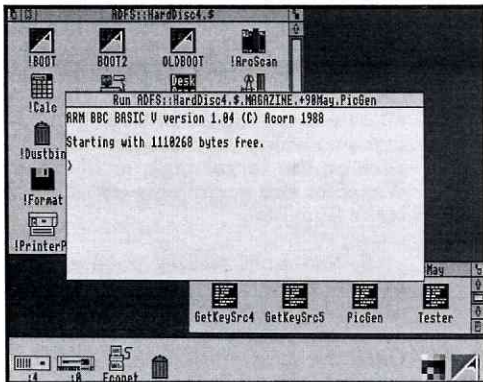
Taxan 775	£516.35 (£543.53)
Taxan 775 LR	£535.00 (£561.75)
Taxan 870	£659.95 (£693.95)
Taxan Viking 2	£862.50 (£905.63) <i>RU</i>

Desktop File Loader

by Lee Calcraft

WHAT IT DOES

One of the problems faced by the programmer when using the Desktop is that there is no simple way to *load* rather than *run* Basic programs. In previous issues we have suggested a number of ways around the problem, such as the Basic-to-text converter (Volume 2 Issue 8). Here a Basic program file could be dragged to the converter, which would convert it into a text file so that it could be dragged into Edit for editing purposes. The method adopted in the present article works in a different way, and offers a number of advantages.



The effect of using Ctrl-double-click on an application's icon is to load the application into Basic, and open a command window, allowing you to examine and edit the program

Essentially, once the accompanying code is installed, double-clicking on an application or on a Basic program file from the Desktop filer with the Ctrl key held down, causes the chosen program to be loaded into memory rather than run. The memory allocated for the purpose is exactly the same as that which would have been allocated to the program for normal multi-tasking purposes, and is determined by the WimpSlot size and the Task Manager.

As a result, you can list your program on screen, use the Basic editor to edit your program (simply type *EDIT* once Basic

Edit has been loaded), save the program back to disc if you wish using the normal SAVE command, or in fact perform any other function that you would normally be able to perform from within the Basic environment.

When you are ready, you can just type *RUN*, and your program will run as a Desktop multi-tasking program with the whole of the Desktop set up exactly as it was before you used Ctrl-double-click, and without the necessity of saving it back to disc first. To examine the program again, all you have to do is to stop the task (either by clicking on its close icon, or by clicking *Menu* on its icon on the icon bar, and selecting *Quit*). This will take you straight back into Basic as before.

Moreover, all the program's variables will be intact. So if you are checking out an error, you can test any variable that you wish. Compare this to normal Wimp error reporting. If your program hits an error, you will get an error message, but you cannot interrogate the states of your variables, because they are all lost (along with the program itself) when you quit the Desktop to enter Basic.

Finally, if you want to remove the task under test, just type *QUIT*, then press space as instructed. The application will disappear, and the Desktop will continue from where it left off.

INSTALLING THE LOADER

To install this system you need to load a short relocatable module (whose purpose is simply to check the state of the Ctrl and Alt keys), and to set up a RunType alias for Basic files, as we will see in a moment. First of all, type in the accompanying program, and save it away before running it. When it is run it will create a relocatable module named *RMGetKey*, and automatically save it to the directory *\$.MODULES*". If you do not have such a directory on your work disc, you should create one by typing:

```
*CDIR $.MODULES
```

Next you will need to install the following two lines on your boot file (i.e. an Obey file - not an Exec file - named !BOOT located in the root directory of your disc which is booted once at the start of a session):

```
*RMLoad $.MODULES.RMGetKey
*Set Alias$@RunType_FFB GetKey|M IF |<
Get$Key><>1 THEN Basic -quit |"%%0|" %%*
1 ELSE Basic -Load |"%%0|" %%*1
```

The first line simply loads the module into the RMA. The second line is a bit more complex. Essentially, whenever the operating system encounters the command *RUN in conjunction with a Basic file (i.e. a file of type &FFB) it executes this line. This first calls *GetKey, a command recognised by the module, and in response the module checks the Ctrl and Alt keys, and sets the system variable Get\$Key accordingly. If Ctrl is pressed it sets it to 1, if Alt is pressed it sets it to 2; otherwise it sets it to 0.

Having made the call to GetKey, the line referred to above checks the state of the variable Get\$Key, and either loads or runs the program accordingly. Thus if Ctrl is pressed the load operation will take place, otherwise the Basic program will be run as normal.

Although the module tests for both the Ctrl and Alt key, we have not made use of the latter. This is included for experimental purposes. You could for example set up an alias for Obey and Exec files which loads the files into Edit if the Alt key is pressed. You cannot use the Ctrl key for this, because the first file to be read when you double-click on an application is its !Run Obey file, and this would prevent Ctrl-double-click from loading a Basic !RunImage file in the same directory.

HINTS ON USING THE LOADER

1. If you want to get directly into the Basic Editor, without having to type *Edit* each time, you could set up a function key to engage Edit - e.g.:

```
*KEY1 EDIT|M
```

and insert the following immediately after the word *ELSE* in the RunType alias

above:

```
FX 138,0,129|M
```

You will need a space before the *FX*. This "presses" function key F1 if the load option is selected.

2. To increase the size of window available for listing a program etc. once the Ctrl-double-click is invoked, simply issue a mode change - e.g. type MODE 12.

3. Be careful when saving programs from within Basic using this software. Unless you give a full path name your program will (as always) be saved to the currently selected directory. Issuing *CAT will catalogue this, and so confirm its whereabouts.

4. If you want to run Twin once you have loaded a Basic program using this system, you will have to make sure (as always with Twin) that there is enough RAM available. Use the Task Manager to allocate enough RAM for Twin to the *Next* task just before you issue the Ctrl-double-click on the target task, or increase its WimpSlot size accordingly (by editing the task's !Run file).

5. Non-multi-tasking programs which use the SYS "Wimp_CommandWindow" call on exit to remove the "Press space..." message, behave oddly on termination. Once the program has terminated, just use Ctrl-L to get you back to Basic.

6. Debugging

If you want to check independently that the module is working properly, type in the following procedure and add it to the end of the main program. Once the main program has been run, typing *PROCTest* will allow you to check it out. Press the Space bar as instructed, then immediately (within one second), hold down the Ctrl key. You should see the number 1 printed out (or 2 for the Alt key), or zero if neither Alt nor Ctrl were pressed.

If this is ok, but the system is not working as a whole, then the RunType alias should be checked. The best way to do this is to type:

*SHOW Alias\$@RunType_FFB
 The underline character before the "F" is created by pressing the Shift key together with minus (i.e. the key to the right of zero).

Module testing procedure

```

880 REM >Tester
890 DEFPROCtest
900 REPEAT
910 PRINT"Press space, then Ctrl or
Alt"
920 A=GET
930 A=INKEY(100)
940 *GetKey
950 *ECHO <Get$Key>
960 UNTIL FALSE
970 ENDPROC
  
```

RMGetKey module generator

```

10 REM >GetKeySrc5
20 REM Program Test Ctrl & Alt Keys
30 REM Version A 1.05
40 REM Author Lee Calcrafft
50 REM RISC User May 1990
60 REM Program Subject to copyright
70 :
80 REM Generates a relocatable
90 REM module called RMgetKey
100 REM Sets <Get$Key> to 1 if Ctrl
110 REM pressed, or 2 for Alt
120 DIM R% &1000
130 FOR i%=4 TO 7 STEP 3
140 P%=0:O%=R% :REM start code at &O
150 [OPT i%
160 ;-----
170 ; Module Header
180 ;-----
190 EQU D 0 ;startcode
200 EQU D 0 ;initialise
210 EQU D 0 ;finalise
220 EQU D 0 ;service
230 ;
240 EQU D title ;Title string offset
250 EQU D help ;Help string offset
260 EQU D command ;Command table offset
270 ;-----
280 ; Title & Help Strings
290 ;-----
300 .title
310 EQU S "GetKeyModule"
320 EQU B 0:ALIGN
330 .help
340 EQU S "Get Key Module"+CHR$9+"1.05
(+MID$(TIME$,5,11)+)"
  
```

```

350 EQU B 0:ALIGN
360 ;-----
370 ; Table of commands
380 ;-----
390 .command
400 EQU S "GetKey" ;The command name
410 EQU B 0 ;Zero terminator
420 ALIGN
430 EQU D keycommand ;Address of code
440 EQU D &00000100 ;Response flags
450 EQU D keysyntax ;Invalid syntax
460 EQU D keyhelp ;Help text
470 EQU D 0 ;Table terminator
480 ;-----
490 ; Help and Syntax messages
500 ;-----
510 .keysyntax
520 EQU S "Syntax: *GetKey (no params)"
530 EQU B 0
540 .keyhelp
550 EQU S "*GetKey sets <Get$Key> to 0,
1 or 2"
560 EQU B 0
570 ALIGN
580 ;-----
590 ; Code for *GetKey
600 ;-----
610 .keycommand
620 STMFD R13!,(R14}
630 MOV R0,#&79
640 MOV R1,#&81
650 SWI "OS_Byte"
660 CMP R1,#0
670 MOVNE R1,#1
680 BNE ok
690 MOV R1,#&82
700 SWI "OS_Byte"
710 CMP R1,#0
720 MOVNE R1,#2
730 .ok
740 ADD R1,R1,#48
750 STRB R1,kvalue
760 ADR R0,key
770 SWI "OS_CLI"
780 LDMFD R13!,{PC}
790 .key
800 EQU S "Set Get$Key "
810 .kvalue
820 EQU W 0
830 ;-----
840 }:NEXT
850 SYS "OS File",10,"$.MODULES.RMgetK
ey",&FFA,,R%,O%
860 *RLOAD $.MODULES.RMgetKey
870 END
  
```

16 bit SCSI



Oak's high speed 16 bit SCSI provide offers a new level of performance for the entire Archimedes range (including the A3000) with data transfer rates of up to 1.4Mb per second, a considerable increase over both ST506 and 8 bit SCSI controllers. Up to 2 Gigabytes of winchester storage may be fitted per machine, and seven SCSI devices including 4 winchesters may be attached to the card.

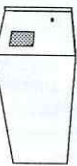
The card provides a new filing system SCIFS, and an icon and filter for the Risc OS desktop. It can work in tandem with ADRS winchesters, and is compatible with the PC emulator. Low level (SVU) support is provided for other SCSI devices (e.g. tape streamers, CD ROMs etc.)



Internal winchester bits are supplied as plug in and go units, with all necessary cabling and mounting hardware, and external drives are supplied in sturdy metal cases, colour matched to the Archimedes, and have their own power supply and fan. External drives are also suitable for the A3000.

A comprehensive manual with easy to follow, fitting instructions is provided with each system, along with a versatile formatting and utility program. Oak SCSI drives may be write protected for security.

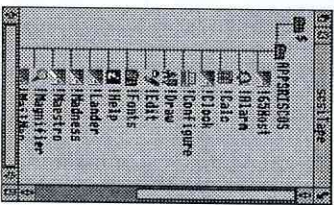
All Oak SCSI winchester drives are subjected to rigorous quality control procedures, and each drive comes with its own test certificate.



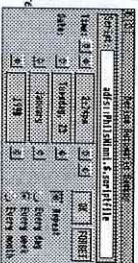
Internal Drives (for SCSI card)	External Drives (for SCSI card)		
20Mb (HDNRAC30C)	£275.00	20Mb (HDNAC30C)	£335.00
45Mb (HDNRAC35C)	£495.00	45Mb (HDNAC35C)	£655.00
80Mb (HDNRAC38C)	£795.00	80Mb (HDNAC38C)	£955.00
100Mb (HDNRAC40C)	£1114.00	100Mb (HDNAC40C)	£1274.00
200Mb (HDNRAC48C)	£1458.00	200Mb (HDNAC48C)	£1618.00
SCSI Card (SC3A4C)	£199.00	300Mb (HDNAC330C)	£2845.00
SCSI Card (SC3B00)	£199.00	P&P external drives	£15.00
P&P internal drives	£10.00		

16 bit SCSI Controller Card and High Speed Winchester Drives

Tape Backup



Unattended backups may be triggered on a regular basis using the built automatic backup facility. Prompts to prevent overwriting of existing files may be given during a restore as required. Note: An Oak SCSI controller card is required. Drives are supplied with 1 free tape cartridge.

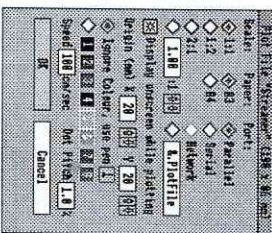


Tape Streamers	£999.95
60Mb without SCSI card (TS60SCA)	£1099.95
60 Mb with SCSI card (TS60SC)	£1294.00
150Mb without SCSI card (TS150SCA)	£1354.00
150Mb with SCSI card (TS150SC)	£1515.00
P&P on Tape Streamers	£24.95
60Mb Tape Cartridge	£27.95
150Mb Tape Cartridge	£27.95
P&P on Tape Cartridges	£0.75

SCSI Tape Streamer

This Risc OS utility allows users of IDraw (and other software than produces drawings) to output drawings to HPGL compatible plotters.

By simply dragging a drawing onto the Worra Plotter icon on the icon bar, it will be output as HPGL commands to the selected destination.



Pen speed, plotter size and pen configuration may be selected from within Worra Plotter. Outputs can be scaled and objects and then sent to either of the ports, the network or to file. Output may be mimicked on screen at the plot proceeds.

Worra Plotter will render all CAD type objects and will render text objects in the plotter's own font. Symbols, line thicknesses, fills and text column objects are ignored.



Worra Plotter costs only

£29.95 P&P £4.50

WorraPlotter



Oak Computers
Cross Park House
Low Green
Rawdon
Leeds
LS19 6HA
Tel: 0532 502615
Fax: 0532 506868

Accepted
Orders
Government
VAT. All prices exclude

Plotter Driver

Into the Arc

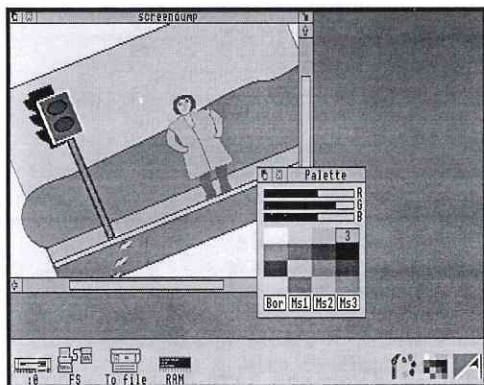
The Palette and the Task Manager

by Mike Williams

This month in our series Into the Arc I thought we would take a look at two of the most obvious tools on the RISC OS Desktop, the Palette and the Task Manager. The Task Manager in particular is very useful, and if you are going to make the most of your Archimedes then you will be using this quite a lot. In any case it is quite informative, and fun to play around with. To start with, though, we will take a look at the palette.

THE PALETTE UTILITY

The main function of the palette utility should be obvious: the control of the 16 colours used by the Desktop. Click with the Select button on the palette icon and an enlarged palette appears with three slider bars above, and four cryptic icons below. The slider bars show the amount of red, green and blue mixed in each colour.



Using the palette utility to adjust colours

Clicking on any of the sixteen colours selects that as the current palette colour. Subsequently adjusting the slider bars will alter that colour, and every occurrence of that colour on the Desktop. The corresponding logical colour number is also shown. The palette therefore provides a way of mixing a set of colours which you might wish to use in your own programs.

To do this you need to know how to save the palette settings. This is very simple. If you click with the Menu button, either over the palette icon on the icon bar, or over the palette window, the resulting menu will include a Save option. Follow the option arrow sideways, specify the name by which the palette settings are to be saved (MyColours for example), and drag the palette file icon to the appropriate directory viewer. The file created will have a palette icon, so you can recognise it from the Desktop for what it is.

If you want to re-use a previously saved palette setting, then from the Desktop, either double click on the palette file's icon, or drag the file icon to the palette icon on the icon bar. If you press F12 for the command line prompt, you will find that typing:

```
*MyColours
```

will also load the previously saved palette settings, provided that these have been saved to the current directory, the root directory, or the library directory (and the library is correctly assigned). This means that any program you might write could include the same (or a similar line), and the colour settings from the palette file will then be immediately available by using the corresponding logical colour numbers. However, you should note that this use of the palette is only valid for programs run from the Desktop. This will also mean that the Desktop colours themselves will subsequently be affected.

The boxes at the foot of the palette window allow you to control separately the colour of the screen border, and the colours used for the mouse pointer. Again, saving the palette is a way of saving alternative border or pointer colours for use in your own programs.

The palette provides two other menu options related to palette settings. You can invert all the grey scale colours (colours 0

to 7). Provided you have not altered these, black goes to white, light shades to dark and so on. If you have changed the colour mix for any of these shades, then the results of inverting can be more unexpected. The other palette option allows you to recover the default colour settings at any time.

Although experimenting with the palette settings can produce bizarre or interesting colour effects, most users are quite happy to settle for the default colour settings - after all, no doubt Acorn's programmers also experimented before deciding on the defaults as being the best. However, the Archimedes and RISC OS give you complete freedom to decide for yourself.

MODE CHANGES

Although not so obvious, the palette icon on the icon bar also provides the means for you to experiment with different screen modes for your Desktop. For convenience I have summarised some of the more useful modes you might like to consider in Table 1. Modes 12 and 15 appear as menu options when you click with the Menu button on a palette icon. The other modes can be obtained by moving the pointer to the blank area at the foot of the mode table whereupon the red caret appears. Type in the desired mode number from the keyboard and press Return.

As well as choosing the mode which gives you the most pleasing display, bear in mind the amount of memory required.

The more memory that is used for this purpose, the less there is left for other tasks and applications. Modes which use more memory also demand more processor time to maintain the display and it is

possible to notice a noticeable slowing down of screen operations in these circumstances. Table 1 gives details of the modes as tested on a standard colour monitor. Remember that modes 18 to 21 are designated for use with multi-sync monitors, and you will probably want

to use either mode 20 (16 colours) or mode 21 (256 colours) if you use a monitor of that type.



Using the palette utility to select mode

Mode	Memory	Comments
8	40K	Interesting monochrome Desktop
12	80K	The default 16 colour Desktop
14	80K	Similar to mode 12 but 25 lines
15	160K	The default 256 colour mode
16	132K	132 column screen gives 'larger' Desktop
17	132K	Similar to mode 16 but 25 lines
19	80K	Gives reasonable monochrome Desktop
20	160K	For multi-sync; otherwise defaults to mode 12
21	320K	For multi-sync; otherwise defaults to mode 15
24	264K	Good resolution 'wide' screen with 256 colours
26	75K	Defaults to mode 8
27	150K	Defaults to mode 12
28	300K	Defaults to mode 15

Table 1. Useful Desktop modes for standard monitors

Once you have decided upon your choice of screen mode for the Desktop, you will probably find it convenient to save this as part of the configuration for your system. However, you may well find different modes suit different purposes. Mode 16 is good for DTP work, for example, because of the amount of information which it is possible to display on the screen at the same time. However, this mode also uses a massive 264K and you can readily see the effect of this on

performance if you open the directory viewer for either of the two Applications discs - the screen will appear to flash with the inability of the processor to update it quickly enough as processor time is devoted to other tasks.

Use the Configure application from Applications disc 1. Double click to install this on the icon bar, and then click on that icon with Select to open the Configure window. Click on the current Desktop screen mode and type in your preferred mode. The configuration setting is saved immediately, but the effect is only apparent the next time the Desktop is re-drawn (by pressing Ctrl-Break for example).

THE TASK MANAGER

The Archimedes 'A' icon at the bottom right of the screen represents the Task Manager. This allows you to monitor the state of currently active (multi-tasking) tasks, and to control the allocation of memory. Unlike the palette utility, the Task Manager responds only to the use of the Menu button, and provides four options.

The first, New task, provides an opportunity from within the Desktop to specify any star command. In that respect it provides an alternative to the use of F12, which most users seem to find both quicker and simpler. Should the command entered need to display information (like *CAT for example) a window will be opened for this purpose and the necessary information displayed followed by the message "Press SPACE or click mouse to continue". When you respond accordingly, the window disappears.

However, the main function of this menu option (New task) is to initiate a new task. For example, you could install Edit from Applications disc1 by entering:

```
:Appsl.$.!Edit
```

where Appsl is the name of the disc. This is equivalent to double clicking on the Edit icon in the directory viewer for this disc. You can try this out to satisfy yourself that it works, but most newcomers to the Archimedes are unlikely to need this option.

Another menu option is exactly equivalent to pressing F12. The last option in this menu is marked exit and provides an exit from the Desktop to the command line interpreter (CLI). This has a keyboard equivalent too, in the form of Ctrl-Shift-F12. The CLI is the part of RISC OS which attempts to execute any so-called star commands.

The most useful option from this menu is the Task Manager itself. Although you may well find this initially daunting, it is worth getting to know what it can do for you. While you are experimenting you will probably find it convenient to click on the top right-hand corner of its window to resize this to full screen size. This reduces the amount of scrolling necessary to see all that is going on.

The Task Manager display is divided into three sections, top to bottom. Firstly, there is a display of current applications, showing the memory allocated, and the amount of free memory. Secondly, there is a similar display for module tasks. Lastly there is a complete table of memory allocations. You can change all memory allocations shown in red by using the Select button to adjust the length of the bar.

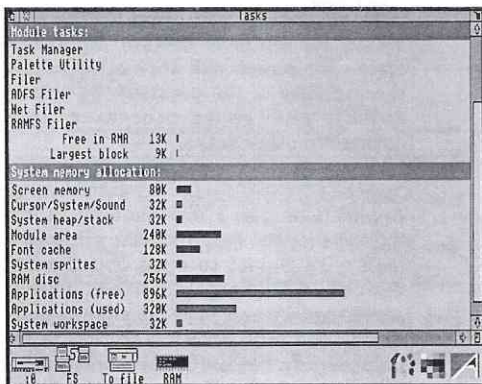
Any applications which you have loaded (by double clicking on their icons) will be listed in the first section. More importantly, memory left over for applications is divided into two categories labelled Next and Free.

By default, most of the available free memory is put in the Next category, but this is not always best. The amount of memory allocated to an application is to some extent determined by limits set within the application itself. If available, an application will grab the maximum memory specified for it. Reducing the Next category restricts the amount of memory the application can take, leaving more memory for any other applications which you subsequently install. However, the run-time parameters for an application may result in this limit being over-ridden, provided there is enough Free memory.

There are no hard and fast rules here - and of course, with applications like DTP or word processing, restricting the memory allocated to an application reduces the amount of information which that application can handle. In addition, you might not leave enough memory to run an application at all. If so, just increase the Next allocation and try again.

The Module tasks section shows those modules which are currently multi-tasking, and the amount of memory allocated for this purpose, and that still free. You can't control this allocation, but modules normally take what memory they need if it is available.

If you click the Menu button over either the Application tasks or Module tasks areas a menu appears with four choices. Three of these are identical in appearance and function to the menu options already described above. The fourth provides a means of deleting any application or module from the list. In fact, there are some applications for which



Task Manager display showing module and system memory allocation (on a 2Mbyte system)

this is the only way of getting rid of them, but they are few and far between, so don't worry if you haven't come across one yet. Just remember the option is there if you need it. Some applications terminated in this way will warn you if there is new information which you have not yet saved to a file; others give no warning at all; so take care with this.

To be continued next month.

RU

"FIRST"

ADVANCED STATISTICAL SYSTEM

For Acorn Archimedes; BBC Models B, B+ and Master; "Industry Standard" machines

A powerful and practical tool for Research, Industry, Business, and Teaching

- Integrated, interactive, robust, fast, accurate, modular. Designed to facilitate data critique.
- Many data entry options. Full data management and formatted display. Extensive data transforms.
- Handles missing values and data subsets. Scatterplots, regression plots, function plots.
- Univariate statistics. Paired and unpaired t tests. Chi-square tests. Nonparametric methods.
- Unrivalled REGRESSION facilities eg weighted, through the origin, standardised. Full statistical information.
- Residuals, fitted and predicted values. Much more. Correlations. ANOVA. Contour plots of regressions.
- Regression DIAGNOSTICS - VIFs, influential points etc. Automatic warnings.
- ROBUST regression - many influence functions. Powerful NONLINEAR least squares
- Time Series, eg moving averages, exponential smoothing, causal models.
- One- Two- and Three-way ANOVA subsystems. Distributions generator. Histograms. Linear Calibration.
- Cluster analysis. Many other powerful analytical and descriptive features. Full utilities.

Price £120-£180 (machine and version dependent). Special RISC USER discount available.

There's far too much to describe here. Get full information now from:

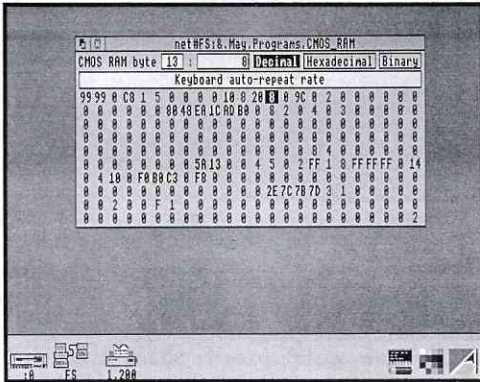
Serious Statistical Software, Lynwood, Benty Heath Lane, Willaston, South Wirral L64 1SD Tel. 051 327 4268

A Desktop CMOS RAM Manager

by Mike Ironmonger

The program described here appears only on the magazine disc.

CMOS RAM is the small area of battery-backed RAM in which various pieces of information, mostly configuring the Archimedes, are stored. This article describes a CMOS RAM Editor, which gives access to all 240 bytes of information in CMOS RAM, allowing all values to be changed as required. The edited version may be used to update the CMOS RAM itself, or saved in a file. Previously saved files may be reloaded for further editing, or for updating CMOS RAM to an alternative configuration.



CMOS RAM edit in action

The application "iCMOS_Edit" is an easy to use Desktop CMOS RAM editor. When the program is run (by double clicking on its icon) there is a short delay while the CMOS RAM is read, and then the icon appears on the icon bar. Clicking on this icon (with Select) opens the program's window.

The lower part of the window shows the value of all 240 bytes of CMOS RAM, with the *current* byte inverted. The top section of the window gives details about the current byte as follows:

The first value shows the location of the byte. You can select a new byte either by clicking on it in the lower section, or by typing the number of the required byte into this box and pressing Return.

The second number is the value of the current byte. If you type a new number in here and press Return, the new value will appear in the main window. Note that this doesn't affect the actual value stored in the CMOS RAM. To change the values stored in the RAM you must use the *Update CMOS* option described below.

The next three boxes determine whether the value of the byte is to be displayed and/or entered in decimal, hexadecimal, or binary. The current base is highlighted, and a new base is chosen simply by clicking on the corresponding name. The full width box is used to display a brief description of the current byte's function.

Clicking the Menu button over the window produces a menu with three options. Choosing *Update CMOS* will write all 240 values displayed in the window into the CMOS RAM. *Read CMOS* performs the reverse operation and reads the values held in the CMOS RAM and reflects them in the window. The final option *Save* leads to a standard RISC OS save window. The files are saved as 240 bytes (0-239) and given the filetype &7FE.

An existing saved file may be reloaded into the CMOS RAM Editor by dragging its icon to the CMOS RAM application on the icon bar.

For general information on configuration settings for the Archimedes refer to the RISC OS User Guide. For details of the 240 byte settings refer to volume 2 of the *Programmer's Reference Manual*. RU

CAD from Linear Graphics

Alan Wrigley looks at a powerful new draughting and design package.

The speed and power of the Archimedes would seem to make it ideal for CAD applications, and there are in fact a number of CAD packages already on the market. They have now been joined by LinCAD, from Linear Graphics.

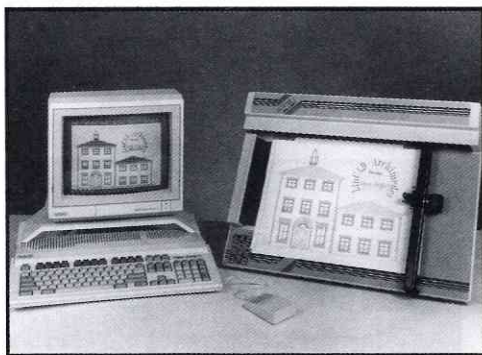
THE PACKAGE

LinCAD comes as a single disc together with a comprehensive 260 page manual. There is also a folder containing teachers' notes and a selection of work sheets, evidence that Linear Graphics is targeting the package strongly towards the educational market. The manual is a model of clarity, and all the functions of the software are clearly explained, with liberal use of diagrams showing the effects on screen of the actions taken. Instructions are given on making a back-up copy of your LinCAD disc (as usual there is a copyright warning), and installing LinCAD on a hard disc. Rather frustratingly, the manual does not have an index, although the contents list at the front is very comprehensive.

When you start up the application, it is immediately apparent that LinCAD takes over the whole machine and is not multi-tasking. Furthermore, although it is fully mouse-driven and uses pull-down menus, the conventions do not follow the standard RISC OS guidelines which Acorn has gone to great lengths to specify. This is a topic which is undergoing heated debate in the Acorn world at present, and which many believe has serious ramifications for the future of the Archimedes. Linear Graphics says it has opted for its own approach in order to keep the software functionally compatible with Nimbus and IBM versions.

Once you have used LinCAD for a few minutes, however, you will find it second

nature. Designs are drawn on a full screen, with a grid of dots to guide you if you require them. Across the top of the screen are the menu headings, and menus are activated simply by placing the pointer over the heading, whereupon the menu pops up. The Select button is used to select, both for items from a menu, and indeed throughout the software, while the Menu button is used to cancel any activity.



LinCAD in use on an A3000, and the resulting output on a plotter

Unfortunately this approach to menus has two drawbacks. Firstly, it is very annoying, when trying to select an item quickly near the bottom of the menu, to find that the menu disappears if the pointer overshoots the menu area, and has to be re-displayed. Secondly, when sub-menus are required, they do not appear to one side of the main menu, as with RISC OS; instead, the main menu is closed and the sub-menu displayed separately on the screen. Furthermore, because of the wealth of features offered by LinCAD, there is simply not enough room at the top of the screen for all the necessary facilities, and a further set of sub-menus is available via the function keys. All this could have been avoided by following the RISC OS conventions.

SOFTWARE FUNCTIONS

The wealth of facilities provided by this package is far too great to be covered adequately in a review such as this. As you would expect, drawings can be built up from all the usual elements, such as lines, arcs, splines, circles, rectangles and triangles. The last three items may be outlined or filled. Rubber-banding is possible using the *track* facility, and you can also draw regular-sided polygons and ellipses. Tracks may be thick, thin, outlined or filled, and corners may be filleted (rounded) or arced (a more pronounced curve). The style of all lines drawn by the program is customisable, with a choice of solid, dashed, dot-and-dash, etc.

Having created an element or series of elements, these can be edited, either as individual items, or by specifying an area of the screen as an editing block. You can move, erase, copy, rotate, scale, distort and so on. You can also load elements from a file into your drawings, and save them back again. Completed drawings can of course be saved, either as sprite files, Draw files, or DXF files. A number of sample drawings are supplied on the program disc. Unfortunately the Draw file conversion routine is not very efficient as it appears not to group objects, but treats each small element in a drawing on the same level, to the extent that each text character, for example, is split into a large number of separate paths at the top level. This introduces an unnecessary degree of complexity and makes converted object difficult and time-consuming to edit in Draw.

Any one of sixteen colours can be selected from a menu, and all drawing operations are executed in the current colour. Areas of the drawing can be hatched, and both the angle and pitch of the hatching can be varied within a wide range. You can also zoom in on an area of the drawing, at up to x32 magnification.

TEXT

One of the most attractive features of LinCAD is its text handling. If you select the *Text* option from the appropriate menu, you can then choose between two different fonts. As supplied, these are a wire-frame and a Roman font, but they can be replaced by any fonts from the LinSIGN/LinFONT package. You can alter the height and width of the text. But the most useful feature is that text does not have to remain horizontal on the screen, as it does when using Draw for example. You can rotate text while positioning it on the screen, and you can even choose to have the text follow an arc or line. Thus you can have text printed around the perimeter of a circle, or around the sides of a complex shape.

HELP!

A comprehensive help facility is built into the package. Pressing a function key when any menu item is highlighted by the pointer will display a box on the screen giving details of that item.

PRINTER OUTPUT

LinCAD has a hard copy output facility which supports a range of plotters, including of course Linear Graphics' own Plotmate, as well as various matrix printers, including Epson FX and LQ, and H-P Laserjet, Deskjet and Paintjet. I tested the package with the FX option, but the quality of output is not as good as that produced by the RISC OS PrinterDM driver. Neither the Integrex, nor colour printing on the Epson, is presently supported. Since the package produces drawings in colour, and many schools have colour printers, this seems a serious omission. However, Linear Graphics assured me that an Integrex driver will be included in the package as soon as possible. Once again, a higher level of RISC OS compatibility, allowing the package to use the RISC OS printer drivers, would have seemed a more sensible course to take.

One point that needs watching when using LinCAD is its extremely healthy appetite for memory. On a 1 Mbyte machine, you may well need to clear all other applications, and any unnecessary modules, out of memory before starting. Even when the program runs, you may still, if you are not careful, run out of memory when adding an element to a design, or loading a drawing from a file. The program does not crash under these circumstances, however, but merely informs you that you cannot carry out that action.

CONCLUSIONS

There are a great many other facilities provided by LinCAD, all of which add up to a most comprehensive drawing package. It just seems a pity that, for reasons which are possibly quite valid in the circumstances, the software has not taken advantage of the RISC OS

framework that Acorn has so painstakingly provided. There are, of course, a number of other major items of software which plough their own furrow, and not everyone is convinced that conformity is obligatory. If you can forgive LinCAD for its individuality then it is a worthwhile piece of software, and one which I am sure will be widely used. It does, however, represent a substantial investment and I would advise potential purchasers to check out the facilities very carefully to ensure that it meets their requirements.

Product Supplier LinCAD Archimedes
Linear Graphics Ltd.
Unit 39,
Mochdre Industrial Estate,
Newtown, Powys SY16 4LE.
Tel. (0686) 629292

Price £228.85 inc. VAT
£172.50 inc. VAT to education

RU

INVOICING & ACCOUNTS

New: V3 of The Account Book now available. Comprehensive small business accounts to trial balance. VAT approved. Absolutely the easiest program to use, with neat final books and hundreds of reports. No entry limits. "The Account Book gets first prize for both price and performance"- comparison in Micro User-July '89. A true user-friendly program. It succeeds admirably "-Beebug - Oct '88. And that was Version 2, V3 has many new features. £27.95.

New: V2 The Invoice Program. Database, Invoices (unpaid and paid), Statements (individual and automatic), Stock presets, Debtor lists, Linking with The Account Book and loads more £27.95. You will not be disappointed!!-See review BEEBUG Dec'89.

Special Offer: £49.95 if purchased together.

Apricote Studios
2 Purls Bridge Farm
Manea
Cambs
PE15 0ND

Tel: 035 478 432 for information, help or to order.

NOAH
PROFESSIONAL

Archimedes Software

Tools#1	£ 10	FastROM BEdit	HardReset Shell	BigMode
Tools#2	£ 10	DiscFree FileFinder	MouseModule DisAssembler	
Tools#3	£ 10	BootFileGenerator AutoMount	Setup AssRoutines	
Paint 16	£ 10	WaffordDigitiser Magnifier	3D - Part ColourStatistic	
Paint 256	£ 129	RiscOsApp Digitisers (Arvis - Wafford)	Scanners advanced KeyCommands	
Macro Assembler	£ 59	coming soon ask for it!		
DemoDisc	£ 0	please send us a SAE		



G.M.A.mbH Wandsbeker Chaussee 58
2000 Hamburg 76 West Germany
Tel.: 01049 - 40 - 2512415, Fax.: 01049 - 40 - 2502860

Desktop Wibble

Nick Smith presents a program that will put your sanity to the test!

Desktop Wibble provides a short and simple demonstration of how the VIDC support chip can be programmed from assembler. Type in the listing, save it, then run the program. This will create a module called WibbleC and save it to the current directory. Double clicking on the module's icon in the directory viewer will install it, whereupon the entire display will start to move around the screen in a circular motion. As you can imagine, this can be very infuriating, especially when trying to use the mouse to select or drag files at the same time!

The program can easily be modified to produce other motion patterns. For example, in order to move the screen in a figure-of-eight pattern, the following lines should be altered to read:

```
520 FOR theta=0 TO (4*PI) STEP (2*PI/45)
530 x%=(COS(theta/2)*23)+123
```

The program will run in any 640x256 screen mode (e.g. 12 or 15). It works by writing directly to the VIDC chip registers. The register number is specified in the top 13 bits, and the data written to address &3400000. We should point out, however, that we do not normally advise writing to logical RAM in this way, as it is clearly machine dependent.

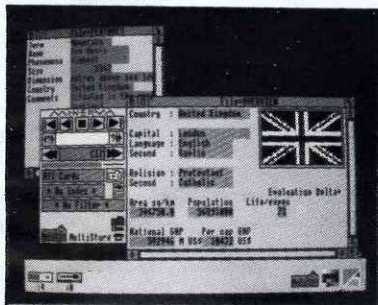
Wibble can be stopped by pressing function key F12 and then typing *FX13,4. To re-start it, use *FX14,4. Typing *RMKILL Wibble will also stop the motion and remove the module from memory.

If you really want to drive yourself crazy, try running Wibble and Madness (from Applications Disc 2) at the same time!

```
10 REM >WibSrc
20 REM Program Wibble module source
30 REM Version A1.00
40 REM Author Nick Smith
50 REM RISC User May 1990
60 REM Program Subject to Copyright
70 :
100 DIM code% 2000
110 FOR pass%=4 TO 7 STEP 3
120 P%+0:0%=code%
```

```
130 [OPT pass%
140 EQU0 0:EQU0 init
150 EQU0 fini:EQU0 0
160 EQU0 name:EQU0 0:EQU0 0
170 .name
180 EQU0 "Wibble":EQU0 0:ALIGN
190 .init
200 STMF0 R13!,{R0-R2,R14}
210 MOV R0,#16:ADR R1,VIDC
220 MOV R2,#0:SWI "XOS_Claim"
230 MOV R0,#14:MOV R1,#4
240 SWI "XOS_Byte"
250 LDMFD R13!,{R0-R2,PC}^
260 .fini
270 STMF0 R13!,{R0-R2,R14}
280 MOV R0,#13:MOV R1,#4
290 SWI "XOS_Byte"
300 MOV R0,#16:ADR R1,VIDC
310 MOV R2,#0:SWI "XOS_Release"
320 LDMFD R13!,{R0-R2,PC}^
330 .VIDC
340 STMF0 R13!,{R0-R6,R14}
350 CMP R0,#4:BNE exit
360 LDR R0,point:ADR R2,table
370 ADD R0,R0,R2:LDMFD R0!,{R3-R6}
380 MOV R1,#&3400000
390 STR R3,[R1]:STR R4,[R1]
400 STR R5,[R1]:STR R6,[R1]
410 SUB R0,R0,R2:CMR R0,#1456
420 MOVGE R0,#0:STR R0,point
430 .exit
440 LDMFD R13!,{R0-R6,PC}^
450 .point EQU0 0
460 .table
470 ]
480 P%=P%+1456:0%=0%+1456
490 [OPT pass%
500 ]:NEXT
510 p%=table+code%
520 FOR theta=0 TO (2*PI) STEP (PI/45)
530 x%=(COS(theta)*23)+123
540 y%=(SIN(theta)*18)+36
550 p%!0=(&8C<<24)+(x%<<14)
560 p%!4=(&90<<24)+(x%+320)<<14)
570 p%!8=(&AC<<24)+(y%<<14)
580 p%!12=(&B0<<24)+(y%+256<<14)
590 p%+=16:NEXT
600 OSCLI "SAVE WibbleC "+STR$-code%+"
"+STR$~(P%)
610 OSCLI "SETTYPE WibbleC MODULE"
620 PRINT "'Module saved as 'WibbleC'"
630 END
```

RU



MULTISTORE

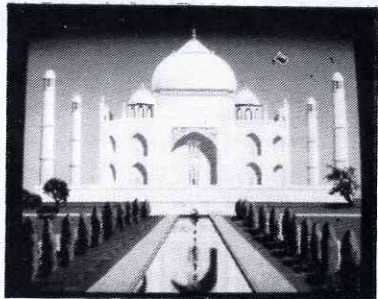
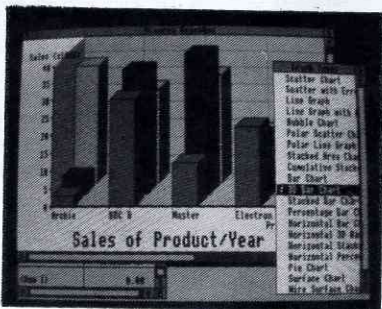
- THE DATABASE

A multi-tasking, relational database allowing many database files to be opened at one time. Files may be linked so that relationships between files are automatically tracked. Rapid searches by any criteria, including 'sounds like' facility. Key fields for fast search, indices to allow file browsing in more than one order without sorting. Fast sort facility by single field or complex expression. Comprehensive reporting facilities allow any report, including totalled lists of fields, labels with multi-font capability, line trapping, justification, paging, multi-set stationery and even completely relational reports.

GRAPHBOX

- THE GRAPHICS

With twenty graph types and its multi-tasking capabilities GraphBox is one of the most flexible packages for the Archimedes. Enter data through Graph Edit or drag data in from any other legal multi-tasking software and produce a graph which can then be exported to a wordprocessor or DTP package. GraphBox gives you a professional graph in a matter of minutes, whether you require a simple scatter or a more complex polar chart. These features plus the ability to enter formulae on the X, Y or Z axis and save it to the Acorn Draw or Edit applications make GraphBox an impressive data presentation package.



ATELIER

- THE STATE OF THE ART

An art package in a class of its own, Atelier exploits the 256 colour capabilities of the Archimedes. Stunning graphics can be created easily and professionally. Atelier is multi-tasking which means your pictures or artwork can be exported to a DTP package by simply dragging them across. Comprehensive sprite editor, undo brush, patterns and definable brushes. Extremely smooth fill routines, ability to import digitised images and advanced features which allow you to wrap images onto any definable 3D shape coupled with the Sequencer to animate your images - only your imagination can hold you back.

PROFESSIONAL SOFTWARE FOR THE ARCHIMEDES & A3000



INERVA

SOFTWARE

69 SIDWELL STREET, EXETER EX4 6PH. TEL: (0392) 437756 FAX: (0392) 421762

Mastering the Wimp

Part 8 - Icons cont'd

by Lee Calcraft

At the end of the last article, we were experimenting with an icon placed in the window of our original test program from part one of the series. Clicking on this icon alternately highlights and de-highlights it - implying selection and subsequent deselection. It is obviously important that a task should be able to find out about the state of the icons within its window(s). This is achieved in one of two ways. Firstly the Wimp Poll automatically reports events associated with the interaction between the pointer and all icons (providing that the icon flags have been set up for this), and secondly the task can use the call `Wimp_GetIconState` to find out the current state of a nominated icon. We will take these two approaches in turn.

USING WIMP POLL

The value of the Wimp flags used for the test icon was `&800B43D`, and within this the value of the icon button type was 11. This tells the Wimp that clicking over the icon should alternately select and deselect it (the highlighting is automatic), and that the button state should be returned under such circumstances. Dragging also causes an event, but we are not concerned with this here.

The Wimp reports button presses through reason code 6. This was discussed at some length in connection with button presses over windows in part 4 of the series, and you are referred to this for background information. The parameter

block returned by the Wimp with reason code 6 was given in figure 3 of that article. It is reproduced here as figure 1.

R1+0	Mouse x coord (screen coords)
R1+4	Mouse y coord
R1+8	Button type
R1+12	Window handle (-1 for background, -2 for icon bar)
R1+16	Icon handle (-1 for work area background)

Figure 1. Parameter block returned by Wimp Poll with a reason code of 6

To try this out on the present icon, add the following line to the test program with the icon added as described last month:

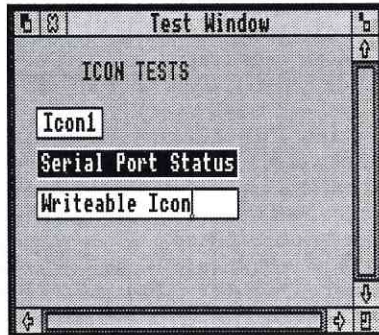
```
465 WHEN 6:IF block%!16=i1handle% THEN  
SOUND 1,-15,70,4
```

In case you have renumbered the program, this should fall somewhere within the CASE statement in PROCpoll. The result of this line should be to make a sound whenever the mouse

is clicked over the icon. Clicking the mouse elsewhere in the window has no effect. All we have done here is to look at the contents of R1+16 (the handle of the icon over which the click was made), and if it is the same as the known handle (i1handle%) of our test icon (originally returned by the Wimp in response to

`Wimp_CreateIcon`) then a sound is made.

Currently no check is made on which button is clicked, but we could easily test `block%!8` for the button number. Note also that both the Select and Adjust buttons



The test program with the three experimental icons

toggle the highlighting, but the Menu button has no effect on it. This is determined by the Wimp, and is not customisable. It powerfully reinforces the notion that selections should not be made with the Menu key.

SYS Wimp_GetIconState

The other way in which a task can obtain information about its icons is through `Wimp_GetIconState` (SWI &400CE). This is called with R1 pointing to a block of memory which contains the window handle at `block%`, and the icon handle at `block%+4`. On exit from the call, these 8 bytes are preserved, and from `block%+8` the 32-byte icon block associated with `Wimp_CreateIcon` is inserted (see figure 1 from last month). Thus the minimum x co-ordinate of the icon's bounding box appears at `block%+8`, the y co-ordinate at `block%+12`, and so on. The icon flags (returned at `block%+24`) can be used to check whether the icon is in a selected or deselected state (by checking bit 21 of the icon flags), etc.

Of course, this call must be made by the task itself, and has nothing to do with the Wimp Poll sequence - except insofar as the task can only perform any activity after the Wimp Poll has returned. To take a real example, in the File Find program published last month `Wimp_GetIconState` is used to determine whether the *Files* and *Directories* icons have been selected by the user, immediately before performing a search.

INDIRECTION

As we said last month, an icon information block can contain up to 12 characters of text (held at `R1+24` - see figure 1 from last month). In our example icon a string of 5 characters was supplied ("Icon1"), and constituted the text within the icon, but this could equally well have

been the name of a sprite used in the icon. If this text string needs to be more than 12 characters in length, then it must be handled differently. We must use so-called indirection. All this means is that in place of the text string in the icon information block we put an address of an area of RAM which will contain the data.

In fact the 12 bytes used by the text string permit up to 3 data words to be supplied in their stead, and Acorn use these variously to give the address of a text buffer, the length of the buffer, pointers to a sprite name, to sprite control blocks, and to a so-called validation string. This latter is used for things like specifying which keyboard characters should be accepted in a writeable icon, and which not - as we shall see next month.

Sprite	Text	Use of the 3 data words
0	1	+24 Pointer to text buffer +28 Ptr to validation string +32 Text buffer length
1	0	+24 Ptr to sprite or sprite name +28 Ptr to sprite control block (+1 for Wimp sprite area) +32 0 if +24 is a sprite pointer, length if it's a name pointer
1	1	+24 Pointer to text buffer +28 Ptr to validation string, which can contain sprite name +32 Text buffer length

Figure 2. The meaning of the 3 data words depends on the state of the sprite and text bits in the icon flags. The address of the 3 words is given as an offset from R1, the start of the parameter block

The exact use to which the 3 data words is put depends on the state of three bits in the icon flags: bit 8 (indirection), bit 1 (sprite) and bit 0 (text). Bit 8 must be set, otherwise indirection is not used. The effect of the other two bits on an indirected icon is given in figure 2.

To help to clarify this we will look at a simple example. Suppose we wish to create an icon containing the text string "Serial Port Status". Because this is more than 12 characters in length, we must use an indirection icon. We must first derive a value for the icon flag for our new icon. If we take as basis the flag used in our earlier example, (&800B43D), the only change we need to make is to set the indirection flag (at bit 8). We can do this by adding 2^8 (= &100) to the original value. This gives a new flag value of &800B53D. If you are competent at ANDing and ORing binary numbers you can use Mike Ironmonger's excellent Desktop Calculator (Volume 3 Issue 2 page 13) to alter flag values. Alternatively you can use the Icon Flag Generator program supplied last month, entering the original flag value, and pressing Return on each entry except bit 8.

This new icon is still text only, so the sprite bit is zero, so that the three data words take the form given in the first of the three entries in figure 2 (i.e. `sprite=0, text=1`). This tells us that the first word must point to the text string in memory, the second to a validation string (we will set this to -1 because no validation string is used in this example), while the third simply gives the text buffer length.

If we dimension a small text buffer at the start of our test program using:

```
82 DIM text1% &20
```

and place our text string into it with:

```
84 $text1%="Serial Port Status"
```

our call to FNicon for the new icon would take the following form:

```
226 i2handle%=FNicon (whandle%, 32, -210,  
300, 48, &800B53D, "", text1%, -1, &20)
```

This places the icon at 32,-210 within the work area, and gives it a length of 300 graphics units and a height of 48. Now if you run the program you will see the new icon. Clicking on it will again alternately select and deselect it.

WRITEABLE ICONS

A writeable icon is exactly what the word suggests: an icon into which text

may be typed by the user. You will see writeable icons in use on the Desktop whenever you perform a *save* operation for example. This form of icon makes life very easy for the programmer, because all the text input and editing performed by the user is handled entirely by the Wimp.

To see how this works, we will create a simple writeable icon along the lines of the icon used in the test above. We will need to dimension another small buffer. This time we will put an empty string into the buffer to start with:

```
86 DIM text2% &20:$text2%=""
```

though it could contain anything you wish up to a maximum length of the buffer size (&20) less 1 (for the terminator). Next we need to generate a suitable set of icon flags. We can base these on those used in the previous example. The only change that we need to make is in the icon button type. This was previously type 11, but we now need type 15. If you use the Icon Flag Generator from last month, and enter the previous value (&800B53D) as a starting point, you will get a new value of &800F53D. If you now add the following line to the test program, the job is done:

```
226 i3handle%=FNicon (whandle%, 32, -270,  
300, 48, &800F53D, "", text2%, -1, &20)
```

When you run the program, you will see the new icon. This will be empty to start with. But as soon as you click the mouse over the icon, instead of the icon being highlighted, the red caret (a vertical cursor) appears, and the window surround turns cream in colour to indicate that this window, and this window alone, has the so-called input focus. Any text typed in by the user will be written into this icon. The user may edit his text using the left and right cursor keys, and use Delete to delete characters to the left of the caret, and Copy to delete characters to its right. Ctrl-U will delete the whole line.

You may notice that when it first appears, the caret is centred horizontally within the icon, and as you type in text or

delete it, it remains centred. This is because we have set bit 3 of the icon flags for this icon. To right justify the text, we need to unset bit 3 and set bit 9 (Right justify). Probably the most common state is to use left justified text in writeable icons. This is achieved by clearing both bits 3 and 9. To do this with the present icon, we need to change the icon flags value to &800F535.

Of course, there is not much point in letting the user type in text if the task doesn't know what has been written. But this is easy: it is stored in the indirected text buffer set up at the start of the program. The string held in \$text2% will always reflect what the user has typed, and may be read at any time. To test this, get out of the Desktop, and run the test

program, entering some text. Then click on the *Quit* icon, and type:

```
PRINT $text2%
```

Or better still, use the Desktop load utility described elsewhere in this issue to perform this test from within the Desktop.

This is probably a good point to experiment further with some of the features of icons that we have covered so far. Next month we will finish off the topic with a look at exclusive selection groups, validation strings, and the use of the icon bar.

This month's magazine disc contains the test application with the three experimental icons, plus a copy of the Icon Flag Generator program first published last month.

RU

• POINTS ARISING • POINTS ARISING • POINTS ARISING •

THE ALLEGED BUG IN PRM OS_SerialOp

by Lee Calcraft

Last month's hints carried an item on a bug in the *Programmer's Reference Manual* page 194 on the SWI "OS_SerialOp" (reason code 1). Unfortunately this was incorrect.

To explain: the PRM gives information about OS_SerialOp1 which conflicts with the settings displayed on all the RISC OS printer drivers. We brought the anomaly to the attention of Acorn's technical departments, and after some deliberation they told us that the fault was in the PRM not the drivers. So we reported the fact in the magazine, and altered the Serial Port Status program (RISC User Volume 3 Issue 4) accordingly.

We have subsequently discovered that the PRM is in fact correct, and Acorn have now confirmed this. The upshot is that the

hint was wrong, the PRM is correct, the printer drivers are all incorrect in this regard, and so is the Serial Port Status Utility.

As far as the latter is concerned, a new version appears on this month's magazine disc. If you are using the version listed in the magazine, simply alter lines 270 and 280 to read as follows:

```
270 WHEN stop%=1 AND bits%=8 AND v%=  
1:truestop=1  
280 WHEN stop%=1 AND bits%=5 AND v%=  
0:truestop=1.5
```

As for the drivers, these will be corrected by Acorn in due course. But for the moment (assuming you are not interested in 5 bit operation), the only error occurs when using the drivers in serial mode, with 8 bits and the maximum number of stop bits. In this case the driver menus tell you that you are using 1 stop bit if parity is not set, and 2 otherwise. The reverse is the case.

RU

Archimedes and RISC OS

Reviewed by Mike Williams

'ARCHIMEDES AND RISC OS'

by Zvonimir Racic, published by Prentice Hall at £18.95.

Given the few books which have so far been published for the Archimedes it is disappointing to have to report so negatively on the latest offering. Indeed this is a curious book in many ways. The style of English used is often odd and some of the mis-spellings (or misprints) are unusual to say the least ("a splitted cursor" for example, or "seriell" printer); the author refers consistently to *diskettes*, thereby revealing a greater familiarity with the PC world (no disadvantage there), but a curious disregard for the conventions of the Acorn market; and lastly we have a book which was apparently written 12 months ago with little regard for the then newly emerging RISC OS operating system.

Prior to RISC OS, the Archimedes range was supplied with an operating system called *Arthur* and what now seems like a rudimentary desktop which most users rapidly abandoned. The book's first chapter is devoted to this! Much of the information in my view is confused or misleading, but as hardly anyone is now likely to be using the Arthur desktop this hardly matters. However, after six pages the author states that "Compared to the desktop the CLI provides much better functions and you will soon be working at this level almost exclusively". The CLI (command line interpreter) interprets star commands typed in by the user. The bulk of the book which then follows (150 pages out of 280) is devoted to a systematic reference and explanation of star commands. This is fine, but hardly what the new owner of an Archimedes wants (or needs) today, and anyway they are, to a great extent, covered in the RISC OS User Guide supplied with every Archimedes.

Given the obvious PC background of the author, there is, not surprisingly, a chapter

on the PC emulator for the Archimedes. There is much useful information here concerning performance and compatibility of software compared to a genuine MS/DOS system, particularly relevant to those who have previously used or are already familiar with this environment.

There is then a twenty page chapter on the RISC OS Desktop and the contents of the two Applications discs now supplied with the Archimedes. Even here, the book contains gaping inaccuracies - "Note that Archimedes differentiates between large and small letters" in reference to file names (which are also stated to be limited to a maximum of *eight* characters!) Other confusions abound.

Chapter five contains various appendices, including details of screen modes (called *modi*), FX calls and VDU calls, all of which are included anyway in the User Guide supplied with the Archimedes.

Whether author or publisher is more to blame for this book I know not. It is lavishly produced, being spiral bound within attractive hard covers, designed to be turned through 90 degrees and laid flat for reading. However, it is very pricey for the kind of information it contains (*Archimedes First Steps* published by Dabs Press, and reviewed in RISC User Volume 3 Issue 4, is half the price and much more useful). In contrast, Racic's book is badly written, confusing and inaccurate; and out of date in its approach to the Archimedes. This is a shame - its approach to the Archimedes from a PC background (and with references to the Amiga and Atari ST) would have done much to endear it to those migrating to the Archimedes from such machines. As it is I do not recommend anyone to buy this book. *RU*

- Full feature word processing • Full capability DTP • Multiple documents in memory • **Multiple windows on each document** • Any scale view from 1%-900% • Automatic hyphenation
- Built-in spelling checker • Spell check as-you-type • **Full 24-bit RGB, CMYK & HSV colour control** • Fast character mode printing • 300K free workspace on a 1Mbyte machine • Supplied with 13 outline fonts
- Flowing graphics with text • Automatic index generation • **Automatic contents generation** • Automatic abbreviation expansion • Automatic file conversion from other WPs • **Virtual memory system - unlimited document size**
 - Simple to use - only 5 main menus
 - Word count • Scaled printing • portrait & landscape pages in one document • **active master pages**
 - thumbnail printing • rotated printing • left/right only printing
 - pamphlet printing • collated printing • pause between pages • cut/copy/paste between documents
 - multiple master pages • **fully customisable start-up**
 - programmable key short cuts
 - overlaying styles • styles can be applied to any part of a paragraph
 - dynamic memory usage • **search & replace for styles and codes**
 - search & replace with wildcards
 - case-sensitive search & replace
 - auto-page numbering • **roman numbering option**
 - page numbering start at any value
 - multi-line headers & footers
 - graphics in headers and footers
 - graphics on/off options for screen & print
 - style merge from other documents
 - style template documents
 - **condensed & expanded text** • left, centre, right and decimal tabs • each style can have its own ruler
 - single & double underline options • text strikeout
 - underline colour control • **rule-offs** • hyphenation control on individual words • hyphenation exception dictionaries
 - programmable tab leader characters
 - internal accuracy to 1/72,000ths inch
 - **horizontal and vertical kerning to 1/1000ths em** • text flow around frame control (both sides, either side or neither side) • guide frames • controllable frame snap
 - embedded graphics frames in text • **embedded graphics on the line** • go to page & chapter
 - two guess-misspell-word options • word anagram solutions • crossword solving facilities
 - **8 user dictionaries** • auto-load user dictionaries on start-up • user dictionaries compatible with Spellmaster • all measurements can be specified as inches, millimetres, centimetres, meters, points, feet, yards or picas • abbreviation expansion as you type • **word substitution as you type** • insert current date & time options • controllable preferences for measurement units & default scale views • character mode printer drivers for Epson & LaserJet printers

IMPRESSION

*If other programs can offer all these features,
then they might be as good as Impression.*



Computer Concepts Ltd

Gaddesden Place
Hemel Hempstead
Herts HP2 6EX
Tel. 0442 63933

Hints & Tips... Hints & Tips

SINGLE KEY SAVE

Lee Calcraff

The following function key definition sets up function key F1 to save a Basic program using its in-core filename (see note below).

```
*KEY 1 VDU21|M Z|M SAVE|M IF ERR=4 THEN
VDU6:P."File saved":SOUND 1,-10,70,1 ELS
E VDU6:SOUND 1,-15,200,6:P.REPORT$'"File
not saved"|M
```

But the key performs somewhat more than is achieved by just typing *SAVE*. It generates different sounds for successful and unsuccessful saves (particularly useful for hard disc users), and it uses customised reporting, giving the message:

```
File saved
if it was successful
or Protected disc (or whatever)
File not saved
otherwise.
```

Normally if you just issue the command *SAVE* to save a program using its in-core name, you get no message on screen to inform you that the save took place, and there is no audible confirmation.

NOTE: An in-core filename is a filename held in a REM in the first line of a Basic program; thus:

```
10 REM >MyProg
```

Typing *SAVE* will save the program with the name *MyProg* to the current directory.

WINDOW SCROLLING

David Spencer

When repeatedly scrolling a window by holding down Select or Adjust over one of the arrow icons next to the scroll bars, the speed of scrolling depends on the keyboard auto-repeat rate. Similarly, the delay for repetition start is determined by the auto-repeat delay. These can be set up using **CONFIGURE REPEAT* and **CONFIGURE DELAY*, or temporarily using **FX12* and **FX11* respectively. The same in fact applies to any icons that use an auto-repeat button type (see the article

Mastering the Wimp, in the previous issue of RISC User).

While on the subject, the auto-repeat rate also determines the speed of text scrolling when Ctrl key is pressed. RISC OS waits between line feeds for a length of time corresponding to the auto-repeat rate.

THE RIGHT CASE

Alan Wrigley

Newcomers to the Archimedes are sometimes confused about when to use upper or lower case, and when it doesn't matter. Filenames, and all operating system commands (i.e. those preceded by "*"), may be in either upper or lower case. When choosing filenames, it is good practice to use case in a meaningful way, e.g. *TextFile* or *ScreenDump*. The file will be saved with the name exactly as specified, but accessing it later does not require the case to match. For example:

```
*Delete SCREENDUMP
would delete the file ScreenDump.
```

Within Basic, however, the right case is imperative. Basic keywords *must* be in upper case, and variables in Basic programs are case sensitive, e.g. *handle%* is *not* the same variable as *Handle%*. This is easy to overlook when typing in listings, particularly in the case of variables.

ERRORS AND THE HOURGLASS

David Spencer

When the hourglass is displayed on the screen, it will normally be removed automatically if an error occurs. However, because of the way in which Basic generates its own errors, this will not happen in response to a 'Basic' error. Therefore, if you are using the hourglass from a Basic program, you should include the statement:

```
SYS "Hourglass_Smash"
```

in the error handling code to prevent the hourglass being left on. For full details of using the hourglass refer to RISC User Volume 2 Issue 7.

Hints & Tips... Hints & Tips

DEBUGGING HALT

Lee Calcraft

When debugging, it can be useful to stop a program at a particular point - either to find out how far it has got before a crash or whatever, or to test the values of variables etc.

You can of course insert *END* or *STOP* at a particular point, but this may well leave you at an awkward point in your program, because neither of these keywords causes a program's own error handler to be invoked. For example, the keyboard might be disabled, you might be writing to the wrong screen bank, function and cursor keys could well be out of action, and files may be left open.

To avoid this, use something like the following:

```
ERROR 255, "HALT in PROCedit"
```

When your program hits this line, it will stop with the message *HALT in PROCedit* telling you where the halt occurred (assuming that your error handler uses *REPORT* or *REPORT\$*), together with the line number (assuming that *ERL* is printed), and all the resetting performed by your error handler will come into effect.

ONE BIG STEP FOR A MOUSE

David Spencer

RISC OS allows the resolution of the mouse pointer to be changed via **CONFIGURE MOUSESTEP* and the Basic statement *MOUSE STEP* (or the corresponding *OSWORD* call). The figure specified is the number of OS co-ordinates by which the pointer position changes for each step of mouse movement, and in turn this affects the 'speed' of the pointer. However, you need to be careful in using a mouse step smaller than the number of OS co-ordinates per screen pixel (see below), as this can result in what appear as straight lines, (drawn using *!Draw* etc.), being a pixel out when viewed in a higher resolution mode.

The number of OS co-ordinates per pixel in the X direction is 8, 4 and 2 for 20, 40 and 80/132 column modes respectively, while in the Y direction it is 4 for normal modes and 2 for multi-sync ones. **Configure Mousestep* only allows a single step, which is used for both the X and Y directions, but *MOUSE STEP* from Basic allows separate steps for each direction.

CLAIMING MEMORY

Alan Christie

If you dimension a block of memory, for example:

```
DIM fred% 99
```

then 100 bytes of memory is reserved, starting at *fred%*. If you then dimension *fred%* again in the same program, a further block of memory is reserved but the original block is not reclaimed. This situation might well occur where the block is dimensioned inside a procedure. Each time the procedure is called, a further 100 bytes are reserved. If this is allowed to continue unchecked, then at some point the program will stop and the error message "No room for..." will be issued.

RESTORE DATA BUG

David Spencer

Owing to a bug in Basic, the *RESTORE DATA* statement causes problems if it is used in a function or procedure (Basic crashes), or within a *FOR-NEXT*, *REPEAT-UNTIL* or *WHILE-ENDWHILE* loop. This is not normally a problem, because *LOCAL DATA* and *RESTORE DATA* will normally be used in a function or procedure, and in this case you can omit the *RESTORE DATA* part, and it will be performed automatically (and correctly).

Another quirk of *LOCAL DATA* and *RESTORE DATA* is that because of the way the Basic tokeniser works, these must always be the last statements on a line.

RU

ARE YOU GOOD ENOUGH?

As the leaders in software for the Archimedes range of computers, CLARES MICRO SUPPLIES are looking to extend our range even further. We are looking for people who are as excited by the Archimedes as we are.

If you have written any programs, completed or not, then we would like to hear from you.

If you have any ideas for programs and have the ability to execute the ideas then we want to hear from you.

If you have the ability to program the Archimedes but not the ideas to program then we want to hear from you.

Programs can be written in any language as long as they perform their stated task. Many of our programs contain large chunks of BASIC with ARM code in the areas that it is needed. BASIC on the Archimedes is a very powerful language and we do not attach any snob value to its use. If your program does what is meant to do then that's all we are interested in. Why not join the top team on the Archimedes. You get the support of our in-house team, privileged access through us to Acorn and invitations to our informal programmers seminars.

The most important point is that you will be earning top royalty rates of if you prefer we will purchase your program outright.

Please write, in confidence, to Mr. D. Clare at:

**Clares Micro Supplies,
98 Middlewich Road,
Northwich,
CHESHIRE CW9 7DA**

If you have a program either complete or in development then please enclose a copy for our evaluation.

To protect yourself we advise that you lodge a copy of the program with your bank or solicitor BEFORE you send us a copy. You can then prove that your program pre-dates anything that we have.

Act today and become part of the leading software team producing software for the worlds fastest micro.



IMPROVED BEDIT FRONT END

I have already found the Wimp front-end for Basic Edit to be very useful (see RISC User Volume 3 Issue 4). However, one problem with loading programs from the Desktop is that you remain in the current directory. When using the save option from Bedit the edited program is saved there rather than in the directory from which it was loaded. I have therefore added a function `FNfind_directory($)` to the program, and call this as the first line of `PROCbasic()` to set the directory for subsequent saves as the one from which the program originates. I think other readers may find this useful:

```
1511 IF file$<>" THEN OSCLI("DIR "+
FNfind_directory(file$))
1680 DEFFNfind_directory(file$)
1690 pos=LEN(file$)-1
1700 WHILE INSTR(file$,".",pos)=0
1710   pos-=1
1720 ENDWHILE
1730 =LEFT$(file$,pos-1)
```

David Allen

APPRENTICE ARCHIMEDIANS

I would like to take issue with Mr Rainthorpe over his letter about articles for beginners (Postbag RISC User Volume 3 Issue 4). I have upgraded to an A3000 after six years with a BBC B, and I feel rather as a driver must who has upgraded from a three wheeler to a Porsche. I look to RISC User to provide information I can't find elsewhere. The manuals, although good, don't always answer those "but how do I?" questions.

I use the computer mostly for word processing, but I like to potter. I have written several programs which work and do what I want. I was delighted to find the article referred to (*Into the Arc* Volume 2 Issue 10), and was elated when I managed to get my own specially designed icon to appear in the directory viewer. So maybe it wasn't a proper multi-tasking application, but I can run it from the Desktop, and it gives me great satisfaction.

I think the success of the magazine depends upon creating and maintaining a versatility

that will appeal to all its readers at whatever stage they happen to be. Please keep sparing a few pages for the *Apprentice Archimedeans*.

Maureen Whitaker

DRAWING CIRCULAR ARCS (1)

With reference to Mr. Colman's letter (in Postbag Volume 3 Issue 4) on drawing circular arcs, I use the following technique. First draw a circle, and then draw two rectangles with line and fill colour set the same as the background to obscure the unwanted part of the circle.

D.A.Freeder

DRAWING CIRCULAR ARCS (2)

With reference to the item about using Draw to trace circular arcs (Volume 3 Issue 4) I would say there is more to this than meets the eye. Although it is true to say that Draw is object oriented, perhaps the crucial question is "What is the most primitive object?"

If after drawing a circle, *Path Edit* mode is selected, it will be seen that the circle consists of four segments, each with start, end and control points. Logically it should then be possible to delete one or more segments, and it is.

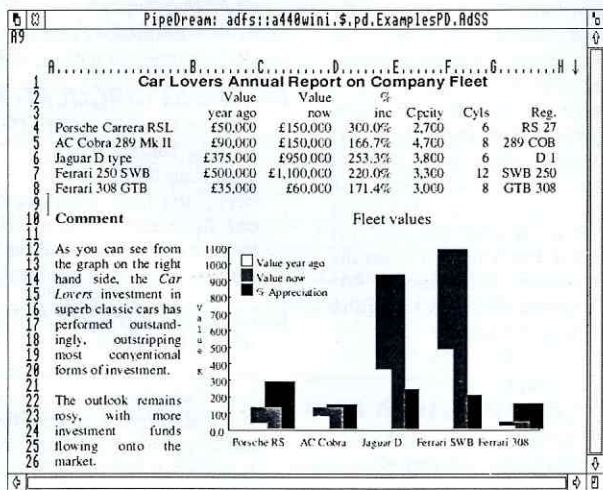
Deleting a segment is achieved by first clicking *Adjust* on one of the blue end points, or yellow control points. This will highlight the selected segment in red. Now from the *Edit* sub-menu select *Open Path* and a gap will appear between two segments. The segment on the clockwise side of the gap can then be deleted by clicking *Select* on one of its points, and then using *Delete Segment* in the *Edit* sub-menu. Further segments may be deleted, provided that each segment to be deleted is to the clockwise side of the gap.

Clearly this procedure will only yield quarter, half, and three quarter circles, but other fractions can be achieved by superimposing a second such arc over the first but rotated.

Denham Boardman

I was aware of this at the time of the original letter, but I felt the advice I gave best suited the basic requirement, and Mr Freeder's suggestion also provides a simple solution. However, Mr Boardman is quite right, and shows that there is indeed more to Draw than meets the eye. *RU*

PIPEDREAM 3



PipeDream 3 breaks down the barriers between word processor, spreadsheet and database. You can include numerical tables in your letters and reports, add paragraphs to your spreadsheets, and perform calculations within your databases.

Based on PipeDream 2, the best-selling integrated package for the Archimedes, PipeDream 3 has been completely re-written to take full advantage of RISC OS - if you can use RISC OS, you can use PipeDream 3. It is fully multi-tasking and multi-windowing, so you can work on many documents at once and instantly move information between them. And since PipeDream 3 can automatically load and save most popular file formats, including VIEW and First Word Plus, switching to it from other programs has never been easier.

Power, flexibility, speed, ease of use. PipeDream 3. Breaking down the barriers.

For a free brochure, see your Archimedes dealer, or phone us on 0954 211472 or return the coupon.

PipeDream 3 is for all Archimedes computers with RISC OS and 1Mbyte of RAM.

PipeDream 3 costs £147.00 +VAT.

Major features include:

- many documents loaded at once
- intuitive RISC OS user interface
- displaying and printing of pictures within text
- built-in 93,003 word spelling checker
- file compatibility with PC & Z88 PipeDream and BBC View Professional
- background recalculation
- keystroke compatibility with Z88 & PC PipeDream
- Z88 filing system
- automatic loading of VIEW, ViewSheet, Lotus, First Word Plus, Tab and CSV files
- automatic saving of VIEW, Lotus, Acorn DTP format, Tab and CSV files
- multi-field sorting
- use of all available fonts
- 62 spreadsheet functions
- external references for 3-D modelling
- macro file recorder
- slot protection

For a free brochure, complete and return this coupon

PipeDream 3 View Professional

Name _____

Address _____

Post code _____

Colton Software, Broadway House, 149-151 St. Neots Road, Hardwick, Cambridge CB3 7QJ, England.
 Fax: 0954 211607 Tel: 0954 211472

All trademarks acknowledged. The chart in the screen shown above was produced by sending numbers from PipeDream 3 to Lingenuity's Presenter 2 and then loading the resulting graph back into PipeDream 3.

Colton Software, Broadway House, 149-151 St. Neots Road, Hardwick, Cambridge, CB3 7QJ, England.

Fax. 0954 211607 Tel. 0954 211472

Technical Queries

ENVELOPE IN BASIC V

The HELP command within Basic includes ENVELOPE in its list of keywords. Typing HELP ENVELOPE gives a statement saying that this takes 14 numeric parameters separated by commas. This command is not mentioned in the old or new User Guide, and trying to guess 14 parameters is very difficult. Can you help?

Paolo Stra

The ENVELOPE command in BBC Basic is left over from the days of the BBC model B and Master series using Basic IV or earlier. The Envelope command provided the means to 'shape' a sound when the SOUND command was used. On the Archimedes the ENVELOPE command has no effect whatsoever, although Basic V still accepts the command to provide compatibility with programs written for earlier versions of Basic.

Although it has no effect in Basic V, its function in earlier Basics was to determine the characteristics of both pitch and amplitude. The pitch envelope consisted of three sections, and the amplitude four sections (called attack, decay, sustain and release). The parameters specified the rates of change in these parts of each envelope. Up to four envelopes could be created, for use with the SOUND command where the second parameter specifies the envelope number.

CREATING A PRINTER DRIVER

A problem I would appreciate some advice on concerns the Computermate CP80 printer. Whilst it is largely Epson compatible, it does feed up slightly further than an Epson on graphics dumps, and thus printouts from Draw etc. have gaps between each scanned line, which makes text almost unreadable and totally ruins an otherwise very impressive printout. I am sure there is a simple amendment to the dump program which would solve this, and I would appreciate your advice on a solution.

John Heath

The printer driver specifications for different printers are held in a text file created with Edit. If you have a printer which is not catered for by the printer drivers supplied, then it is perfectly feasible to create your own.

To do this load Edit from Applications disc 1, and then on the same disc open the application directory for PrinterDM by double-clicking while holding down Shift. This will reveal a file

called PrData. Open this by double clicking on its icon or by dragging it to the Edit icon on the icon bar.

To create a new printer driver, go to the end of the last printer driver entry in the file and copy any existing entry for a specific printer to this position (note that the very end of the file contains an outline description of the format of a printer driver). Now edit the copied entry to create the new driver, making sure you give it the next printer number in sequence (probably 13). Follow the pattern of the other drivers, and the outline referred to above. Once complete, resave PrData. When you next install the printer driver on the icon bar, use the Select key to cycle through the entries until you find the one you have created. You may need to proceed by trial and error before you get it completely right.

DISTINGUISHING KEYS ON THE KEYBOARD

Using Basic, the keys labelled Page Up and Page Down return the same ASCII values as the Cursor Up and Cursor Down keys. Can you provide any help on a way of telling the difference?

Valerie Oliver

There appears to be no solution to this problem as far as ASCII codes are concerned. Once the command *FX4 1 has been issued, then the cursor keys and others on the keyboard which normally perform a function return an ASCII code instead. For example, Cursor-Up returns &18F and Shift-Cursor-Up returns &19F. Unfortunately, Page-Up returns &19F and Shift-Page-Up returns &18F, the same codes as the cursor key, though reversed (but see next issue). Thus this method cannot be used to distinguish between the four keys.

The alternative is to use the negative INKEY function which detects individual key presses. This takes the form INKEY-n where the value of 'n' determines the key to be detected. Cursor-Up has a value of 58 while Page-Up has a value of 63, Cursor-Down has a value of 41 while Page-Down uses 78. In every case, negative INKEY returns a value of TRUE if the relevant key was pressed at the time of execution, and False otherwise. Thus:

```
REPEAT
IF INKEY-63 THEN PRINT"Page Up pressed"
IF INKEY-57 THEN PRINT"Cursor Up pressed"
UNTIL FALSE
```

is a simple loop which distinguishes between the keys indicated.

RU

RISC USER MEMBERSHIP

RISC User is available only on subscription at the rates shown below. Full subscribers to RISC User may also take out a reduced rate subscription to BEEBUG (the magazine for the BBC micro and Master series).

Subscription Rates	RISC User	RISC User & BEEBUG
UK, BFPO, Ch.Is	£16.90	£25.00
Rest of Europe & Eire	£24.00	£36.00
Middle East	£29.00	£43.00
Americas & Africa	£31.00	£46.00
Elsewhere	£34.00	£51.00

All subscriptions, including overseas, should be in pounds sterling. We will also accept payment by Connect, Access and Visa, and official UK orders are welcome.

Back Issues

We intend to maintain stocks of back issues. New subscribers can therefore obtain earlier copies to provide a complete set from Vol.1 Issue 1. Back issues cost £1.20 each (£1.60 for Volume 3.) You should also include postage as shown:

Post and Packing	UK, BFPO, Ch.Is	Europe plus Eire	Elsewhere
First Issue	60p	£1.00	£2.00
Each subsequent Issue	30p	50p	£1.00

MAGAZINE DISC

The programs from each issue of RISC User, together with a number of additional items, are included on a monthly 3.5" disc. This will be available to order, or you may take out a subscription to ensure that the disc arrives at the same time as the magazine.

Disc Subscription Rates

Single issue discs
Six months subscription
Twelve months subscription

UK	Overseas
£4.75	£4.75
£25.00	£30.00
£50.00	£56.00

Disc subscriptions include postage, but you should add the appropriate rate (see above) per disc for individual orders.

RISC User & BEEBUG, 117 Hatfield Road, St Albans, Herts AL1 4JS.
Tel: St Albans (0727) 40303 (24hrs answerphone for Connect/Access/Visa orders and subscriptions)

RISC USER MAGAZINE DISC CONTENTS

PROGRAMS LISTED IN RISC USER MAY 1990

Bank Switching

Three programs demonstrating different uses of bank switching - animation, switching screens between alternative displays and debugging.

Using Draw Files in Basic

The updated library of routines to display and manipulate Draw Files, and a program to demonstrate the translation, scaling and rotation of pictures (including sprites).

Assembler Workshop - Quick Sort

A routine for sorting Basic arrays, using ARM assembler.

Desktop File Loader

A very useful utility which allows you to load a program from the Desktop, rather than run it.

Desktop Wibble

A short routine which demonstrates how to program the VIDC and brings some fun to the Desktop.

ADDITIONAL PROGRAMS REFERRED TO IN THE MAGAZINE

A Desktop CMOS RAM Manager

An easy to use Desktop CMOS RAM editor, published only on the disc.

Mastering the Wimp (Part 8)

A test program, which offers three experimental icons, and a copy of last month's Icon Flag Generator program.

A Serial Port Status Utility

An upgraded version of this program, published in Vol.3 No.4.

BONUS ITEMS

Circuit Drawing

This user friendly application allows you to design your own circuit diagrams, by selecting and dragging symbols from a comprehensive library to the screen.

New Tunes for Maestro

Additional music pieces for classic music lovers.

Ogre's Lair

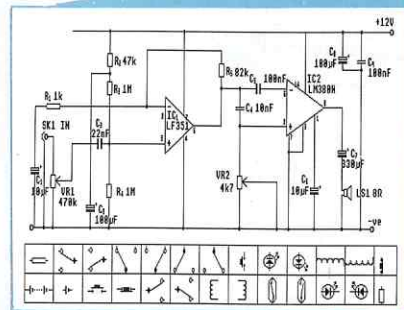
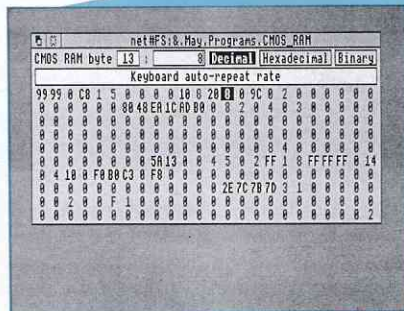
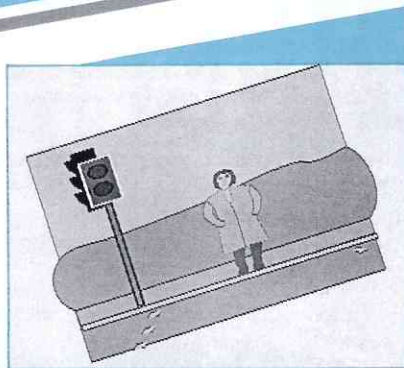
An arcade type of game to provide hours of fun.

Arc to Psion

Software for communicating with the Psion Organiser - including options to download ArcScan files and OPL programs to the Organiser.

ArcScan Data

ArcScan data files for this issue of RISC User and BEEBUG Vol.8 No.10.



BEEBUG

The Archimedes Specialists

Beebug Hardware Upgrades for the A3000 and Archimedes

BEEBUG Hard Drives for the 400/1 series

The BEEBUG internal hard disc upgrades use only high quality mechanisms from respected manufacturer's such as NEC, Rodime and Seagate. All our drives are easy to fit.

- All drives have an access time of less than 29ms
- 40 & 53Mb drives feature autoparking
- Full 12 months warranty
- 53Mb drive uses the top of the range Rodime drive as used in the Archimedes 440/1
- All drives fully tested and formatted prior to despatch
- Free fitting (by appointment) if brought to our showroom
- Backed by the resources of Beebug
- Full fitting instructions included

0120G Beebug A400 20Mb Drive	£199.95
0121G Beebug A400 40Mb Drive	£326.95
0122G Beebug A400 50Mb Drive	£499.95

BEEBUG A3000 Memory Upgrades

The BEEBUG A3000 memory upgrades are available in either 1 or 3 Mb versions and are easy to fit by the user. This will give a total of either 2Mb or a massive 4Mb of memory.

By using the latest surface mount technology we have achieved a board size of a mere 256x46 mm.

The 1 Mb board can be easily upgraded to 3 Mb at any time by the user, unlike some other boards on the market.

- Easy to install with fitting instructions provided
- Full 12 months warranty
- 1 Mb expandable to 3Mb by the user
- Uses latest surface mount technology
- Free fitting (by appointment) if required

0130D Beebug A3000 Memory 1Mb	£164.95
0131D Beebug A3000 Memory 3Mb	£399.95
0132C 1Mb to 3Mb upgrade	£250.95

BEEBUG A3000 Disc Buffer

An exciting new release for the A3000 allowing the connection of 5.25" drives to the A3000. Some of the features include:

- Connector for BBC-type 5.25" & 3.5" drives
- On board switches allow drive mapping to be changed. eg. The internal drive could be made drive 1, and an external drive as drive 0 - ideal for the PC emulator
- Whole interface contained on an internal card
- Easy to fit, supplied with full instructions
- Will work with most drives that have an internal Power Supply Unit
- Can read DFS files with optional DFS Reader
- Internal and external drives can be double stepped to read 40 track discs
- Full 12 months warranty

0135C A3000 buffer	£39.95
0102B DFS reader	£ 9.90

BEEBUG Archimedes 5.25" Disc Buffer MK II

A high quality buffer mounted neatly and simply in the back a 300 or 400 series Archimedes.

- Almost any 5.25" drive with a PSU may be used
- Up to four drives (two external, two internal)
- Full buffering of signal from computer
- DIP switches allow re-mapping of drives

0795C 5.25" Buffer (305/310/440)	£33.25
0784C 5.25" Buffer (400/1 series)	£33.25
0102B DFS reader	£ 9.90

BEEBUG Serial Link Kit

A fast, convenient method of transferring data between the BBC Micro/Master and the Arch. A3000 users will require the A3000 serial chips.

- Transfer in both directions
- WIMP driven, easy to use software
- Compatible with both DFS and ADFS files
- Includes software, 3m cable and instructions

0796C Serial Link Kit	£17.25
-----------------------	--------

117 Hatfield Road, St Albans, Herts AL1 4JS Telephone: (0727) 40303 FAX: (0727) 60263

Prices quoted are for BEEBUG members and include VAT. See Retail Catalogue for details on postal charges.

Prices and specifications subject to change without notice.