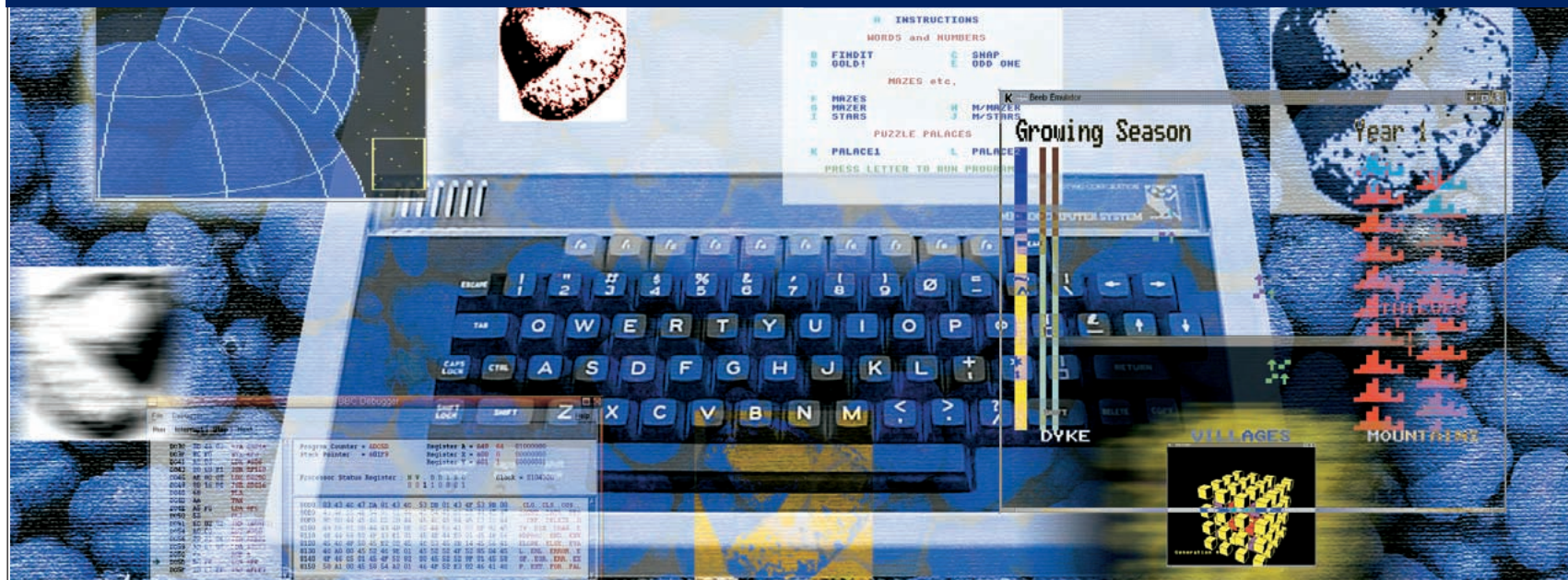


Emulators



Acorn emulators

Simon Goodwin tests Linux Acorn emulators from Atom to Archimedes.



Your cover disc includes emulator sources and many of the utilities mentioned in the article.

Acorn's BBC Micro model B became a standard fixture in British schools.

Acorn was a major player in the UK micro industry for 20 years, first selling a bare-board 6502 development system, then home computers of steadily increasing sophistication from the Atom, via the BBC Micro, to the 32 bit Archimedes powered by their own ARM chip, capable of running Linux in its own right.

Few home micro manufacturers develop a microprocessor architecture as well as computers and operating systems. This rare legacy means ARM emulation is still very relevant. A dozen Linux programs allow you to emulate features of all the Acorn desktop systems, from the Atom, through various models of Acorn BBC Micro, to the Archimedes.

Acorn stopped making computers in 1999, but rights to their machines passed to Pace Microelectronics, work continues on their 32 bit operating system RiscOS, and the spin-off company ARM Holdings, formerly Acorn RISC Machines, is a leading processor design company, responsible for millions of chips at the heart of PDAs, Set Top Boxes and other appliances. So Acorn emulation can give an insight into one of the most pervasive RISC architectures, as well as access to much classic British educational software.

Acorn's roots

Acorn-developed BBC Micros were standard in UK schools for more than a decade. These machines, known as Beebs, had a generous range of interfaces, thanks to stipulations of the contract for the BBC's computer literacy series that brought the broadcaster's endorsement.

The token-ring Econet gave effective networking for classroom use — and fully

expanded machines had an MFM floppy controller, RS423 serial, Centronics parallel and four analogue inputs, a user port, 1 M/s parallel expansion bus and a 'tube' port for fast parallel connections to memory shared with coprocessors, through FIFO buffers. Later this was used to augment the 2 MHz 6502 with faster versions, CP/M-compatible Z80s, and other CPUs.

The BBC Micro project was born in 1980 but it took two years and much wrangling before Acorn got the BBC Micro into production, with help from ICL, and there was a six month waiting list for much of 1982. Acorn's previous machines, the Acorn 1 and Atom, were far simpler machines, typically sold as kits. The Acorn 1 was a bare-board 6502 programmed in hex. The Atom added a case, keyboard and flickery TV display, but with only 2KB of RAM, expandable to 12KB, and an 8KB Operating System.

Later Acorn pitched a small version of the BBC Micro, the Electron, at home users, arguing that it would give owners a head start at school. The Electron was a disappointment, slower than the real thing and not entirely compatible, and lacking the games software range of its home computer contemporaries like the Commodore 64 and Spectrum 128.

Commodore responded to the educational dominance of the BBC Micro by bundling a Beeb emulator with their *Class of the Nineties* UK Amiga pack. This made emulation a major selling point of a new computer bundle, and highlights the relevance of the BBC Micro a decade after its introduction. Acorn themselves eased the introduction of the 32 bit Archimedes by producing a competent emulator for old 8 bit BBC micro software.

The Archimedes shrugged off Acorn's original OS and DFS and ADFS filing software in favour of RiscOS, a modern multi-tasking operating system with a neat GUI. A stop-gap, called Arthur, was written mainly in BASIC and looked better than it worked. The best Acorn game is *Elite*, a hybrid of true 3D shooting with space economics in the *Galactic Trader* mould, and *Frak* was the best-looking original platform title. But educational software remains the Beeb's greatest asset.



Eight Acorn emulators

There are eight Linux emulators for eight bit Acorn systems, including one for the Atom, a couple capable of running Electron software, and several for the BBC Micro range, from Model A to Master Compact.

AtomEm

Version 0.3 of *AtomEm* comes as C source, with scant documentation, though the author, Frans Faase, has a helpful web page. *AtomEm* tries to load three 4KB ROM files – ‘akernel.rom’, ‘abasic.rom’ and ‘afloat.rom’, though the last was optional on the original machine. It emulates the Atom’s display, CPU and keyboard. Press shift to reverse the direction of the two supported arrow keys.

A useful monitor and disassembler is built in. HELP lists most of the instructions, but not BREAK – to set a breakpoint – or LOAD and SAVE. *AtomEm* is capable of loading binary programs, HTML or text listings, but not disk images.

AtomEm supports lowres text and 256x192 mono bitmaps (without authentic snow!) on any X display depth. It builds OK on SuSE Linux 7.2 after **make depend** to tailor it for the distro, and adding ‘-L/usr/X11/lib’ to LDFLAGS.

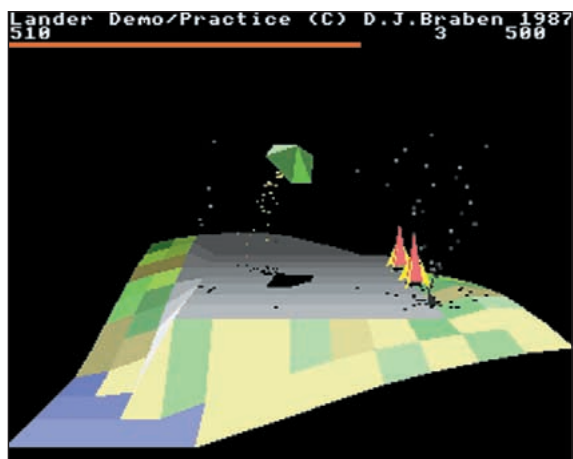
Beeb 6502s

A couple of Beeb emulators focus mainly on the 6502 processor. Version 0.2 of *bbcmicro* comes precompiled for x86 and fails to load *libXm.so.2* when launched. Linking *libXm.so.1* there in ‘/usr/lib’ got it up and running – even without ROMs, though it needs BASIC and OS 1.2 to be useful. It has a good set of monitor commands, and clear resizable displays of the CPU state, memory and 6502 disassembly, but not a lot else.

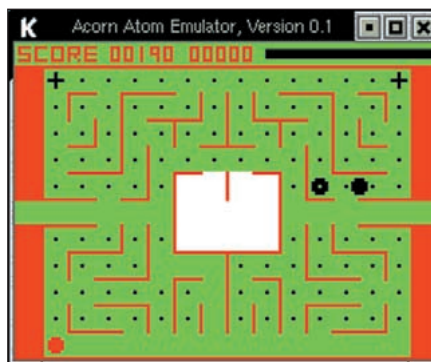
em6502 is a 6502 emulator capable of running BBC BASIC and Acorn’s operating system, written in 1996 as a Bristol University project. To compile and run this on a recent Linux box takes just four commands:

```
cp Makefile.linux Makefile
make depend
make
./em6502
```

This gets the nice text GUI up – but unfortunately text is all you get. *em6502* concentrates on CPU emulation and does not



Lander, later *Zarch*, was an impressive demo of the Archimedes 32 bit BASIC.



The chunky atom *Snapper* game is unplayably fast on a modern PC.

simulate other aspects of the BBC micro, like graphics, sound or peripheral devices. Even so it’s capable of running BBC BASIC, in the teletext mode 7.

em6502 needs a patched version of Acorn’s OS ROM; the documentation outlines the patching but doesn’t automate it. Like *bbcmicro*, *em6502* appeals most if you need to write and debug BBC Micro machine code.

beebem

Beebem comes from the author of *Arcem*, and is a front runner despite minor build, display and keyboard foibles. The *beebem* binary comes with a script to load 8 bit X but fails on the *libstdc++-libc6.2-2.so.3* dependency. Source uses **.configure linux**, but cannot find *iostream.h* and moans a lot about it. But **make** compiles to a runnable binary which only falls over for want of ROMs and 8 bit video. It needs 8 bit X with MITSHM.

Graphics are doubled vertically to keep the PAL aspect ratio, and sound is supported, plus disk images in single or double-sided formats with 2.5K per track. Once I added the OS1.2 and BASIC2 ROMs *beebem* worked well with PD disk images I downloaded, accessing them from the default disk path ‘/var/lib/beebem/diskimg/test.ssd’.

Keys are mapped like a real Beeb so ‘’ is next to ENTER and END is the copy key used for two-cursor editing – one picks up characters and the other shows where they will be inserted when END is pressed.

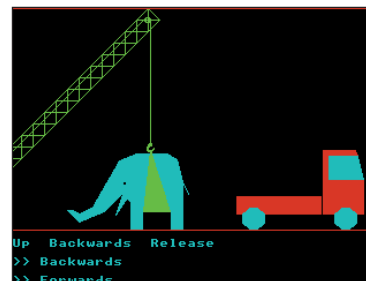
Xbeeb

James Fidell’s *Xbeeb* is sporadically updated but LXF interest prompted a major upgrade earlier this year, when it gained access to most of the formats used for Beeb files online, authentic though CPU-hungry sound, and support for modern graphics depths and resolutions. This long awaited update is on your cover disc, with substantial documentation.

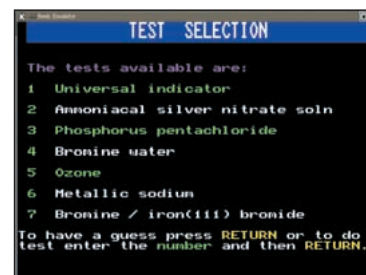
Xbeeb supports access to the Linux file system, with a CATALOG file to hold the information needed by Acorn’s *CAT and *INFO commands. As most of the work is done in Linux not



The 4K BASIC on the Atom was usable but eccentric.



BBC Education – you’ll believe a turtle can control an elephant.



Now you can revise your carbon chemistry with a Beeb.

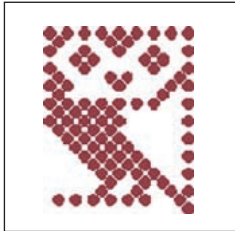


Not *SimChancellor*, but *Great Britain Limited* – they don’t make economies like that any more.

Emulators

DFS, filenames are case sensitive; the keypad enter key allows the simulated disk path to be changed. An alternative XDFS using a new emulator specific ROM image is more compatible with a real beeb.

As in *beebem*, the key layout mimics a real BBC micro; F10 is equivalent to F0 on a Beeb, and Insert on a PC keyboard mimics Acorn's copy key. Nine neat X fonts are used for the MODE 7 teletext display, and must be converted from bdf to pcf format before installation. By default Xbeeb runs at the same speed as a real BBC Micro, but the speed limiter is removed if you edit src/Config.h to undefine FASTHOST, though this will mess up the audio emulation.



This dotted owl symbolised the BBC Computer literacy project.

beeb

The source for *beeb* has not been touched since 1995 but compiles without problems bar a warning about the old, dangerous and deprecated `gets` function. The emulator needs

'os.bbc' and 'basic.bbc' 15K and 16K ROM files from a real Beeb, saved with something like this:

```
*SAVE OS C000+3C00 C000
FOR I%=84000 TO &7FFF STEP
4:1%:4000=1%:&8000:NEXT
*SAVE BASIC 4000+4000 8000
```

After running these commands in BASIC on a real BBC Micro, put the results with lower-case names and a .bbc suffix in data, and fire up the emulator with `.beeb`. *Beeb* can use cursors rather than X if text is all you want, but support for graphics modes is limited and it has no sound.

TBE

Tom's Beeb Emulator, by Tom Lees, is written in C++. It can emulate BBC Micro Models A and B or Electrons, with ADFS floppy and SASI – early SCSI – drives. It comes with a cross-disassembler for 6502 code and CPU emulation routines. It offers limited sound support, for tones but not random noise.

The version of *TBE* I found lacked serial port emulation and disk formatting code. Tom's TO DO list mentions Electron Plus,

Arc access

Archimedes emulation is a tricky business, given the speed and sophistication of the original machines, but if you're willing to settle for a mid-range ARM 6 with some restrictions it's quite viable on a recent Linux system.

There are four variations on the theme of ARC emulation – *ArcEm*, *ARMphetamine*, *Riscose* and *ROX* – but *ArcEm* is the only one ready for use in classic emulation. The others focus on specific layers of the Archimedes environment, from the ARM processor up to Acorn's RISC OS (Reduced Instruction Set Computer Operating System).

ARMphetamine is a fast emulation project setting out to cross-compile ARM RISC machine code on the fly. Links to its former home on Dynarec appear broken and I've not been able to find anything to try, but if *ARMphetamine* should resurface it could come closer to true ARM performance on alien CPUs than existing interpreting emulators. ARM make their own conventional emulator freely available. It's a good start if you want to run ARM machine code but fairly slow and not specific to the Archimedes hardware.

Linux RiscOS

Riscose takes another tack, aiming to provide a RiscOS environment in which applications can be statically compiled to run on Linux. Developer

Matthew Bloch told LXF, "I'm flat out on other contracts to worry about *Riscose* at the moment, but it has run *grep* for a while, and Norcroft *cc* has been nearly there for a few months. There are a couple of people who chip in occasionally but it's fallen by the wayside for the moment. The implication of running *cc* is that you could cross-compile RISC OS applications on Posix systems."

If you have a lingering enthusiasm for the RiscOS way of doing things try *ROX*. It's not an emulator, but it brings a RiscOS look and feel to Linux. *ROX* is a desktop environment for X which should be instantly familiar to Archimedes users, and quite easy for others to pick up. Once you teach it some file associations it's a capable rival to *GMC* or *KDesktop*, with excellent file matching, searching, localisation and support for drag-and-drop via *XDND*.

ARCEM 0.50

ArcEm is an ARM6 emulator based on code from Acorn, written by David Gilbert, one of the developers of the experimental Amulet ARM processors. *ArcEm* emulates an Acorn A300 or A400 with floppy and hard drives, MEMC memory management mouse and keyboard, authentic video and battery-backed configuration RAM. It has been tested on a 16 and 32 bit X display – earlier

versions required 256 colour palette mapping – but might not work on a packed 24 bit X desktop.

ArcEm 0.50 looks good but lacks sound, Econet, serial and parallel port access. Keyboard handling is rather sluggish and lock keys must be pressed twice to take effect from Linux. Floppies can be in 720KB MSDOS format or Acorn's 800K E format which uses five 1K sectors rather than 9 half K ones, as determined at compile-time. Four files in the launch directory correspond to the floppy drives. These can be links to '/dev/fd0', for example. You must send a signal to indicate a **FloppyImage0** disk change, in order to swap disks without rebooting the emulator.

You need to define hard drives in a 'arcemrc' file in your home directory. The example in the emulator archive allocates a couple of 20MB hardfiles:

```
cp armul-arc/arch/dot.arcemrc ~/.arcemrc
```

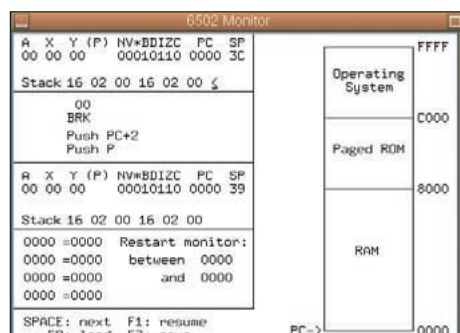
Assign 'ENDIAN=BIGEND' in the Makefile if compiling on a big-endian CPU such as a 68K or PPC. The ARM, like the 6502, the x86 and the author's DEC Alpha, is little-endian.

The makefile looks for the *Gnu C compiler* in '/home/dg/egcs/bin/gcc'; the line 'CC = /usr/bin/gcc' is more likely to work on typical Linux distros. *GCC 2.95* throws up warnings about dodgy definitions and initialisation, mostly in Acorn's ARM emulation code, but it generates a working executable.

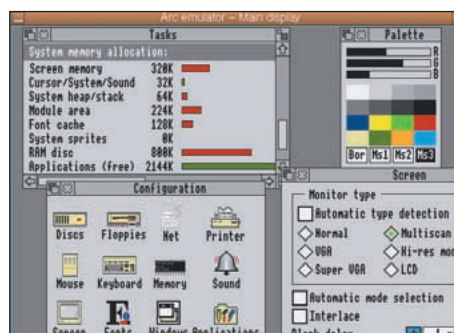
The RiscOS31 ZIP file online contains four 512KB images, and these need to be merged into one 2MB file for the emulator, like this:

```
unzip RiscOS31.ZIP
cat IC24.ROM IC25.ROM IC26.ROM IC27.ROM
>ROM
```

A link to '/dev/fd0' failed to read an Archimedes World disk at the first sector; **CAT** gives **error &41c - disc not understood**. My Amigas can read it with Frank Swift's *XFS* so I guess it's just not in the correct format for the emulator. Acorn switched from D format to E format, which is supported by *ArcEm*, during the nineties, so Arch disk formats vary. MSDOS floppies may be the simplest option.



Em6502 gives a text-mode window into the Beeb's CPU and memory map.



Graphical configuration programs are built into the Archimedes ROM.

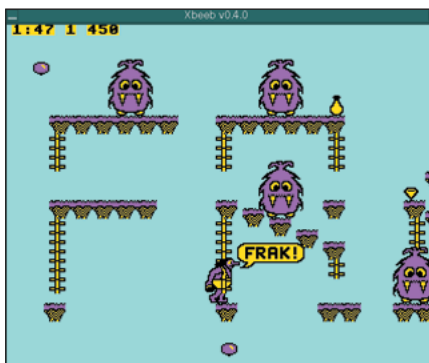
BBC Master 128 and Compact emulation, plus support for tapes, Econet and co-processors — none of which are presently implemented. You'll need *GTK* and thread support to compile the source on Linux.

ElectrEm

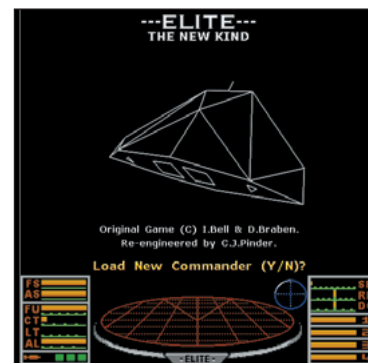
ElectrEm is the only Linux emulator specifically for the Acorn Electron. It is available as ready-made binaries with the necessary ROMs, with the knowledge of Pace Micro Electronics who own the rights after the breakup of Acorn.

Unlike the basic Electron emulation in *TBE*, *ElectrEm* supports tape files and +1 and +3 expansion units, and has a menu system. You have a choice of logical or physical keyboard maps, reflecting Electron layout or your host keyboard, or can set your own by editing an ASCII config file. The documentation is in RTF (Rich Text format), with a 19 page generic file and one page about each specific version.

One Linux version uses *GGI* for input and output; the other expects *SVGAlib 1.3*, gaining joystick input but suffering messy ASCII entry for filenames. Sound is a bit erratic on both. Printer

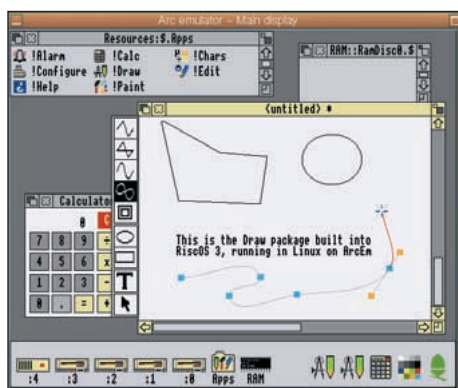


Frak is authentically slow but good-looking in the new Xbeeb.



A souped-up Elite now runs natively in Linux via Allegro, and there's even a version that runs within Emacs!

output is text only. If programs are Electron-specific this is the Acorn emulator for you, but if they would also run on a BBC Micro, *Xbeeb* or *BeebEm* are more attractive.



The Archimedes has a resident vector graphics package with support for smooth splines.

Once you start the emulator with a RiscOS ROM you will find yourself in a simple shell with a * prompt. The **help** command is a good start; there's a lot of good stuff in that 2MB ROM. *BASIC* puts you into BBC BASIC 5, where the supervisor commands remain available if prefixed with an asterisk. Try ***Help Commands**, and note that, unlike Acorn BASIC keywords, you don't have to type these in capitals. If it freezes, press Scroll Lock once or twice.

The desktop environment is accessed with ***desktop**. Drive icons appear along the bottom, and others as programs are launched. At first you see an Acorn logo, which calls up status information, and a coloured panel to adjust the palette. A folder icon labelled 'Apps' calls up a window with built-in program icons. As you launch these or others from drives they add icons to the toolbar.

You must press + on your numeric pad to lock the Linux pointer in the middle of the emulator window and transfer control to the emulator's pointer, before you can make selections on the Archimedes screen. Press + again to release the pointer for Linux. The middle mouse button calls up RiscOS menus, reminiscent of *KDE2* or *GNOME*'s toolbar, though RiscOS could do it long before.

The main screen background is called the pinboard, and windows appear there with conventional gadgets to move, scroll, pan, raise, lower, maximise, minimise, resize and close these. Pointer tracking is a bit slow but usable; though I found I was not able to make selections on the extreme left of the pinboard, window dragging meant this was not a serious problem.

The ROM includes a vector drawing package, with support for B-splines, a pixel painting application for making sprites and larger images, and the usual desktop calculator, toolbar alarm clock and text editor. The ROM also has context-sensitive help which updates messages in a fixed window depending on what you are pointing at, like Apple's bubble help, but without the delayed floating overlays. The *Characters* application lets you preview the impressive anti-aliased fonts.


Configure calls up a dozen application icons that set up peripherals and GUI preferences. Real Arcs are very good at changing mode on the fly, but *ArcEm* on X has a more difficult job as it must potentially emulate all sorts of display on a single X desktop. I was able to select the colour 800x600 pixel SuperVGA, rather than the 640x480 Multiscan default, and various others by adjusting the monitor

type and flipping though default mode numbers in the screen configuration window. But selecting a HiRes monitor confuses *ArcEm*, which redraws a strip of the screen but then loses track of the mouse.

ROM resources

Your cover disc includes emulator sources, *ROX* Desktop and Filer packages for Debian, rewrites of the classic Acorn space game *Elite* for Linux and, amazingly, *Emacs*, connection and conversion utilities, cross assemblers and disassemblers.

Epic's £5 Flash ROM 2 includes the Atom emulator with software, the Beeb DFS ROM image, and vintage BBC emulators for Unix including three versions of *BeebEm* and *beeb*, *Xbeeb* archives and update patches, *em6502* and *bbcim*, a file converter.

It also holds utilities like *Kermit* and other comms software for the beeb, hosted ROM, file and graphic converters, with helpful overview files in the utility directories. Overall Flash ROM 2 contains a relatively well-organised collection of Beeb-related files, though this is far from the sole focus of the CD, which particularly excels in support for another British 6502 system, the Oric. But we'll have to look at those files another month... 

Acorn links

Emulators and ROMs on the web

ArcEm: <http://ftp.linux.org.uk/pub/linux/arcem>

BBC Micro emulator:

<http://www.qualsar.co.uk/bbcmicro>

Beeb emulator:

<ftp://ftp.mrc-lmb.cam.ac.uk/pub/jkb/beeb>

BeebEm home:

<http://titan.abdelion.com/kaell/beebem>

Beeb software: <http://www.nvg.ntnu.no/bbc>

Elite for Emacs:

<http://members.fortunecity.com/salkosuo/elite-for-emacs>

Elite originals:

<http://www.users.waitrose.com/~elitemarc/elite/archive>

Electron emulator: <http://electrem.emuunlim.com>

Epic (Flash ROM 2 CD): <http://www.epiconline.co.uk>

Portable Elite in C:

<http://home.clara.net/cjpinder/elite.html>

RiscOS on X: <http://www.janw.easynet.be/rox>

Tom's Beeb emulator:

<http://www.lpsg.demon.co.uk/files/beeb>

Xbeeb:

<http://www.cloud9.co.uk/james/BBCMicro/Xbeeb>