Sinclair specials

Simon N. Goodwin peers beyond the Spectrum to discover some other Z80 pretenders.

On the CD and DVD

The ZX Spectrum may be the most celebrated of Sinclair's computers, as we noted in LXF14, but millions of other Sinclair machines were sold before and after the heyday of the Spectrum. This column looks at those machines, and Linux emulators for them – the Sinclair QL, the Sinclair ZX80 and ZX81, and close relation the Jupiter ACE.

These are interesting machines, uncommonly well-documented, as you might expect of systems developed in the UK. Often the original manuals and technical documents are available online, as are the vital ROM files, and even operating system source code.

Qdos story

There are a couple of QL emulators for Linux, or more properly for its Qdos operating system. Qdos influenced the early design of Linux – the QL, or Quantum Leap, was Linus Torvald's choice of computer when he outgrew his grandfather's Commodore VIC20. He picked the QL because of its Motorola processor and the multitasking capabilities of Qdos, unrivalled in any home computer at the time.

Linus only adopted Intel after the 386 had plastered over the worst misfeatures of x86 architecture, and QL development stalled following Amstrad's takeover of Sinclair. Since then Qdos has continued to grow, through new, much faster Motorola-based hardware – some of which can run Linux – and emulators.

Most of Qdos was written by Tony Tebby in the second half of 1983, as a back-up in case the official QL operating system Sinclair commissioned from GST in Cambridge missed the launch deadline. In fact, Qdos was also late – it was not fit to be shipped until May 1984, yet the QL was announced in January in a rush to pre-empt Apple's Macintosh, which used the same 68K and bitmap screen, though without multitasking or colour.

GST-68K/OS arrived eventually, but by then Sinclair had plumped for Qdos, and few QL owners adopted GST's resource-hungry alternative. A third system, CP/M-68K, arrived a few months later, but that had little to recommend it other than a superficial similarity to CP/M and MSDOS; it was nonetheless picked up by Atari, renamed TOS, and went on to hobble their ST range of home computers.

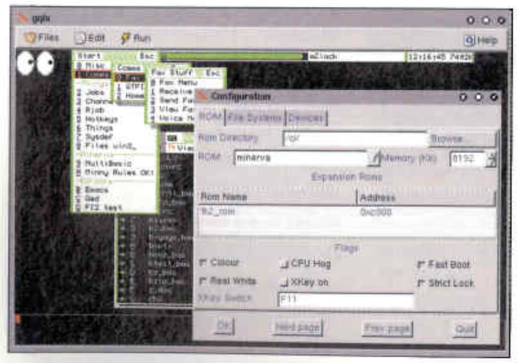
Qdos has much in common with Unix – it is a device-independent, multitasking system, with a priority scheme identical to the original Linux one. All free memory is used to minimise the re-reading of drives, but background I/O is handled by the scheduler, rather than processes for each device. And rather than a shell it uses the block-structured Sinclair SuperBASIC as its command language.

SuperBASIC was designed for Sinclair by Jan Jones, rather than bought in from Cambridge mathematicians Nine Tiles like earlier ZX ROMs. It's a far cry from ZX BASIC or MBASIC, with 32 bit addressing and more than a thousand extension keywords freely available, in the form of re-entrant 'toolkit' files.

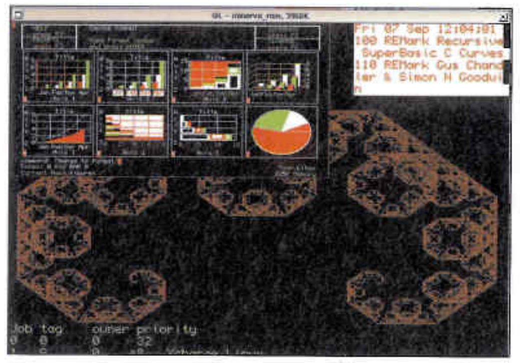
Program structures are well-supported, but data structures are scant beyond local and global arrays, 32K strings, 16 bit integers and six-byte real values. It's more than adequate as a command language, and great for prototyping, but you wouldn't want to use it for a group project.

The interpreter is not re-entrant but SuperBASIC can be compiled into standalone concurrent tasks, with multi-threading, sharing code and data. The latest compiler, Turbo, is now open source, still in active development, and on your LXF CD. The standard Qdos C compiler is C68, but GCC is also available. You can download many versions of the 48K QL ROM, and the emulators also run *Minerva*, a third-party Qdos-compatible ROM with many extensions.

Re-entrant 'toolkit' extension commands and functions augment the inbuilt set. Most of these are free, but some remain proprietary. Tebby's commercial *Toolkit 2*, originally sold as an expansion ROM, implements wildcards, default paths and command history. We can't covermount this as it's not freely-distributable, but our CD includes a PD toolkit with the key commands needed by compiled programs.

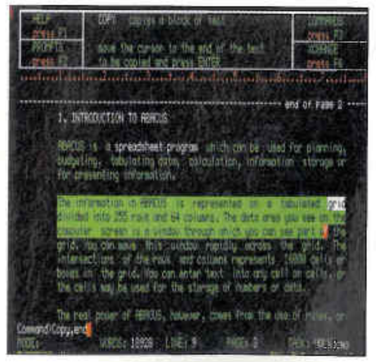


***GQLX*, the optional VDK frontend for *UQLX*, and a *QPTR* Qdos desktop.**

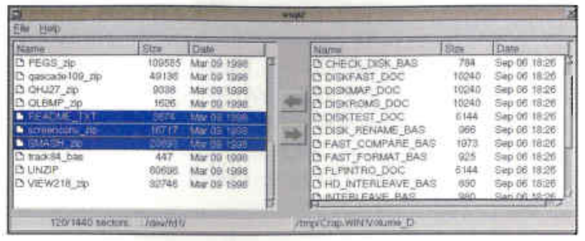


```
Fri 07 Sep 12:04:01
100 REMark Recursive
SuperBasic C Curves
110 REMark Gus Chand
ler & Simon N Goodwi
n
```

Psion's *Easel* in a corner of a big *UQLX* Minerva desktop.



The QL standard wordprocessor *Quill*, from Psion *Xchange*.



Name	Size	Date
FEGS_zip	109565	Mar 09 1998
qascade109_zip	49136	Mar 09 1998
QHJ27_zip	9038	Mar 09 1998
QLBMP_zip	1626	Mar 09 1998
README_TXT	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]
SMASH_zip	[illegible]	[illegible]
track84_bas	447	Mar 09 1998
UNZIP	60696	Mar 09 1998
VIEW218_zip	32746	Mar 09 1998

Name	Size	Date
CHECK_DISK_BAS	784	Sep 06 18:26
DISKFAST_DOC	10240	Sep 06 18:26
DISKMAP_DOC	10240	Sep 06 18:26
DISKROMS_DOC	10240	Sep 06 18:26
DISKTEST_DOC	6144	Sep 06 18:26
DISK_RENAME_BAS	966	Sep 06 18:26
FAST_COMPARE_BAS	1973	Sep 06 18:26
FAST_FORMAT_BAS	925	Sep 06 18:26
FLPINTRO_DOC	5144	Sep 06 18:26
HD_INTERLEAVE_BAS	690	Sep 06 18:26
INTERLEAVE_BAS	990	Sep 06 18:26

120/1440 sectors /dev/fd1/ /tmp/Crap.WIN/Volume_D

***WQXT* is a friendly twin-list GUI for Qdos floppy, QXL.WIN and Linux file-system transfers.**

Contacts

There's much QL and ZX material on the LXF CD, including five emulators and related materials, but to show their true versatility you'll need to trawl the web. The following links should help you augment your collection:

Sinclair QL: http://www.soft.net.uk/dj
UQLX:
http://www.geocities.com/SiliconValley/Bay/2602/uqlx_main.html
QDOS:
http://wwwusers.imaginet.fr/~godefroy/english/QDOS.html
QLAY: http://web.inter.nl.net/hcc/A.Jaw.Venema
XACE: http://hem.passagen.se/tiletech/ace.htm
ZX80 DIY: http://www.home-micros.freeserve.co.uk/zx80/zx80.html
Z81: http://rus.members.beeb.net/z81.html
ZX81: ftp://ftp.zx81.nl/emulators/unix/

QL hardware

The original QL hardware was designed down to a £400 price, and it showed. Backing storage was 100K Sinclair microdrive tapes, with a reasonable 16K/second transfer rate but a tardy 3.5 second average access time. The 68008 processor was 32 bit internally, but hampered by an 8 bit bus, slowed further by contention from the custom display chip. A cheap Intel co-processor handled input from two slow RS-232 ports, and square-wave sound effects.

The first few columns of the 512 by 256 pixel display fell off the left edge of a PAL TV due to a belated timing bodge, and there were only eight TTL colours even in the 256 by 256 pixel mode. Hardware flashing used serial attributes, like Teletext but in a bitmap. Emulators don't bother with this. The RGB colours were augmented by providing 256 stipple patterns that could be used in place of solid colours.

Screen windows offered floating-point scaling and clipping of vector and turtle graphics, and eight preset text sizes. CSIZE 2,1 makes displays a lot easier to read on a big X display. That command affects the default window used by PRINT. CSIZE #0,2,1 scales up command entry, and CSIZE #2,2,1 does the same for listings.

Simon's much-expanded QL barely fits the original case.

The silent, slimline Qdos and Linux-compatible Q60.

With 128K RAM, a quarter needed for the screen, and two microdrives, the basic QL did not match up to the potential of the CPU. Even so, Psion managed to bundle a four-part business suite onto it, using tape overlays. This was later sold as Psion Xchange for PCs and QL-derived 'One Per Desk' systems sold by BT and ICL. The language used to program the Archive relational database became Psion's OPL organiser language.

Xchange is freely available, and vital to read the Quill _DOC files which are a Qdos standard. It also includes *Easel* for business graphics, the *Archive* database, and the spreadsheet *Abacus*. These are competent packages, though limited by the original QL screen-size.

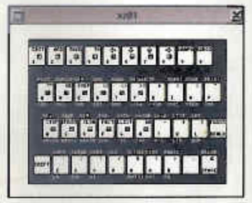
Sinclair and Thorn EMI had provided the basic hardware, helped by Samsung who made the neater US variant, in Korea. Amstrad showed little interest, apart from restarting production for a batch of machines sold through Dixons, but third parties piled in with extra memory, floppy and hard drives, faster processors and full colour graphics.

They're still at it, though hardly any of the original hardware is used now. The QL has a big international user group, Quanta, with sub-groups and workshops in the UK, Germany, Holland, France and North America. The credit for this longevity owes more to Motorola and the Qdos operating system, than to David Karlin's shoehorned hardware. These days Qdos is as likely to run on a PC, Amiga, ST or Mac as on custom-made hardware.

Linux QLones

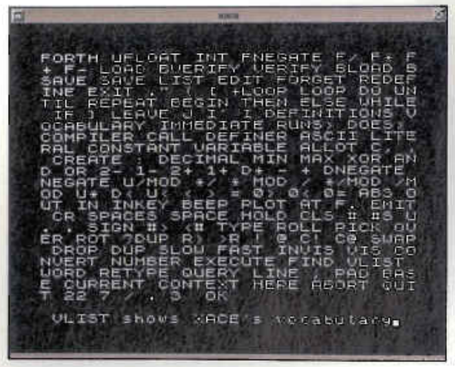
The state of the art on the Qdos scene now are the Q40 and Q60 boards which can run Linux, the free 'Classic

»



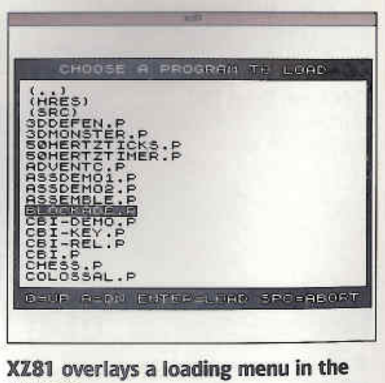
The ZX-81 piled four or five functions on each (so called) key.

Software Farm's *Forty Niner* pushed the ZX81 hardware beyond Sinclair's expectations.



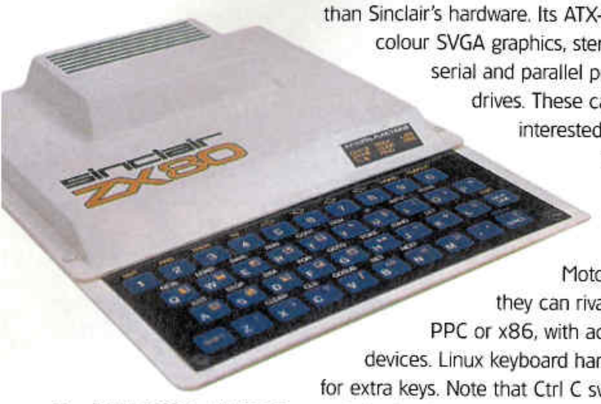
```
FORTH UFLOAT INT FNEGATE F/ F* F
+ F- LOAD BVERIFY VERIFY BLOAD B
SAVE SAVE LIST EDIT FORGET REDEF
INE EXIT ." ( [ +LOOP LOOP DO UN
TIL REPEAT BEGIN THEN ELSE WHILE
 IF ] LEAVE J I' I DEFINITIONS V
OCABULARY IMMEDIATE RUNS> DOES>
COMPILER CALL DEFINER ASCII LITE
RAL CONSTANT VARIABLE ALLOT C, ,
 CREATE : DECIMAL MIN MAX XOR AN
D OR 2- 1- 2+ 1+ D+ - + DNEGATE
NEGATE U/MOD */ * MOD / */MOD /M
OD U* D< U< < > = 0> 0< 0= ABS O
UT IN INKEY BEEP PLOT AT F. EMIT
 CR SPACES SPACE HOLD CLS # #S U
. . SIGN #> <# TYPE ROLL PICK OV
ER ROT ?DUP R> >R ! @ C! C@ SWAP
 DROP DUP SLOW FAST INVIS VIS CO
NVERT NUMBER EXECUTE FIND VLIST
WORD RETYPE QUERY LINE . PAD BAS
E CURRENT CONTEXT HERE ABORT QUI
T 22 7 / . 3  OK

 VLIST shows XACE's vocabulary.
```

***XACE* has a comprehensive FORTH vocabulary.**



```
CHOOSE A PROGRAM TO LOAD
(..)
(HRES)
(SRC)
3DDEFEN.P
3DMONSTER.P
50HERTZTICKS.P
50HERTZTIMER.P
ADVENTC.P
ASSDEMO1.P
ASSDEMO2.P
ASSEMBLE.P
BLOCKADE.P
CBI-DEMO.P
CBI-KEY.P
CBI-REL.P
CBI.P
CHESS.P
COLOSSAL.P
Q=UP A=DN ENTER=LOAD SPC=ABORT
```

XZ81 overlays a loading menu in the ZX font.

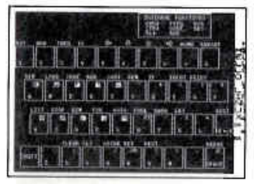


Sinclair's ZX80: 4K ROM and 1K RAM for a bargain £99.

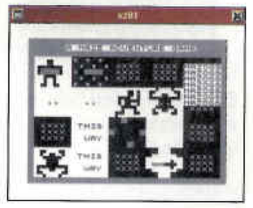
Qdos', or Tebby's commercial follow-up SMSQ on their 68040 or 68060 processors. The 80MHz Q60 is the fastest 68K Linux platform around, and runs QL code more than 100 times faster than Sinclair's hardware. Its ATX-format motherboard has high-colour SVGA graphics, stereo sound, and slots for buffered serial and parallel ports, HD floppies and IDE hard drives. These cards appeal to Motorola devotees interested in Linux as well as Qdos.

The Linux Qdos emulators, *QLAY* and *UQLX*, cannot match the speed of the Q60, because they have to interpret the Motorola 68K instruction set, but they can rival the Q40's 68040/40 on a fast PPC or x86, with access to all the Linux ports and devices. Linux keyboard handling is good, with full support for extra keys. Note that Ctrl C swaps between input contexts, and the break sequence is Ctrl SPACE.

Emulator graphics are rendered via X in *UQLX*, and SVGAlib in *QLAY*, but Unix implementations have some catching up to do. *QLAY* only supports original QL resolutions; while it's rival allows much larger windows but no more colours. So far, Tebby's true colour drivers only work on SMSQ, – via the Aurora QL graphics card, QXL hardware and the Windows QPC2 emulator – not yet the original Qdos run by the Linux emulators.



Keyword help on the Sinclair ZX-80 emulator.



Don Priestley's scrolling maze adventure *Mazogs*.

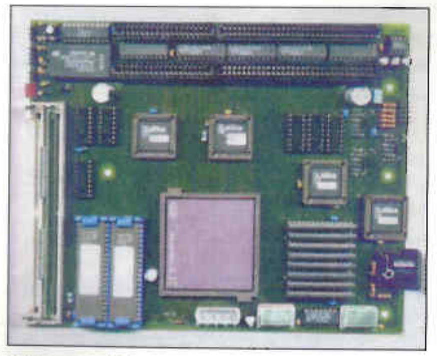
Qdos filing

Linux tools transfer files to and from Qdos floppies and QXL.WIN files. The latter are hard disk images, originally designed for a QXL, a 68040 co-processor for PCs, and since used on Qdos PD CDs and emulators.

Executable file transfers are complicated by the need to preserve the 'dataspace' of Qdos tasks. This magic number determines the amount of space initially allocated for data when the task is executed. Qdos puts this in the disk directory, but there's no equivalent place for it on a CD or in a Unix directory, so emulators have to store it separately.

The documentation for *qltools* (for floppies) and *qxltool* (HD/CD) explain ways to keep Qdos files intact. These are shell utilities; Jonathan Hudson's *WXQT* graphical interface does similar things using the WXwindows toolkit. It's a boon if you have lots of files to transfer.

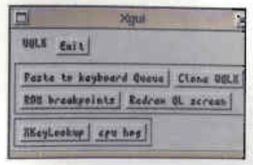
Several equivalent tools run within Qdos, much as .hqx and .sit files pack the forks of a Mac file into one stream of bytes. InfoZIP standards support this Qdos extension, so the Qdos unzip task fully extracts tasks compressed on a real QL or emulator.



Q40s run Qdos 100 times faster than QLs, and Linux too.

The standard path separator in Qdos is an underscore, rather than a slash or a dot. Thus the SuperBASIC startup file is read from mdv1_BOOT, and other files have names like win1_Codegen_Task or FLP1_DEMO_BAS. BASIC programs are launched with LOAD or LRUN, and tasks with EXEC.

Qdos is case-independent and names are limited to 36 characters after the device prefix, normally three letters, a drive number and the underscore. *UQLX* supports much longer names in the local file-system, but at the price of case specificity.



A simple X panel boosts control over *UQLX*.

QLAY

QLAY was the first QL emulator for Linux. *QLAY 0.83* is only available as an x86 binary, compiled in 1997. It's showing its age, but is an authentic emulation if your system can run it. Source is available for the later version 0.90, but this only includes makefiles for Windows and MSDOS. *QLAY* for Linux uses SVGAlib, and must be started as 'root' from a console.

QLAY emulates original QL hardware, including microdrives, graphics and sound. It does not extend Qdos for add-on drives or bigger screens, though it can access the host filesystem, and supports 8MB RAM and system extensions through ROM images.

QLAY is potentially much faster than an original QL, thanks to a 68K emulation core derived from UAE. Most users will need to tune it by running a short program that compares emulation speed with the host's real-time clock.

FT_BAS supplies a command line parameter which set the sound and cursor update rates. This parameter only affects the emulated system's timing interrupts: **-f 2900** is about right on a 200MHz Pentium that emulates Qdos ten times faster than Sinclair's original. Another option, **-w**, can slow down the whole emulator, for programs that become unusable beyond the authentic QL speed.

Microdrives are emulated using 170K files with the original QL directory and data inside. An accessory program, *QLAYT*, can move files into and out of these tape images, but without the latency and 'bad or changed medium' reports! Microdrive images are a lot faster and more reliable than the real thing but rather confining, so *QLAY* includes a dummy ROM device driver, NFA, which gives 'Native File Access.'

The Qdos device 'win1_' can see files on the Linux file system, but only if they are logged in the file 'qlay.dir', which also stores extra information QL directories need to identify and run tasks correctly. *QLAYT* is again used to manipulate this metadirectory file.

UQLX, QL++

QLAY does a good job of running old QL software, especially if it relies on microdrives and original QL timing, but *UQLX* is better if you want to manipulate lots of files or pixels, and essential if you need access to serial ports or host TCP/IP. *QLAY* is a straightforward emulator, while *UQLX* aims to bring the strengths of Qdos and Unix together in a single productive environment.

The QLAY README file is plain text that tells you just enough to get those old programs running. *UQLX* comes with thousands of lines of HTML. Its documentation is far more comprehensive, but inevitably more complicated.

UQLX has excellent integration with Unix, on any system that can run GCC – not just Linux, but BSD, HPUX, Irix, Solaris and SunOS, on Alpha, PPC, 68K, SPARC, MIPS and HP-PA as well as x86 platforms. The emulated *Qdosx* can launch and control Unix programs and be called from Unix scripts with I/O redirection.

UQLX offers direct access to QL floppies, hard disk files and images on CD, without depending on *QLAYT*. It can support megapixel displays, running a free Minerva ROM. Yet *UQLX* could do with updates to match multimedia capabilities recently grafted onto Qdos, such as the sampled sound system QLSSS and true colour drivers for SMSQ.

These two emulators are complementary. In different ways they're both good, and as their sources are freely available they are likely to improve.

ZX81

The ZX81, sold in the USA as the TS1000, was Sinclair's first big hit, thanks to a deal with Timex to do the manufacturing and US distribution, and some impressive production engineering that reduced a complete computer into four or five chips – the ROM, the CPU, one or two static RAM chips, and a custom chip that replaced a couple of dozen TTL parts in ZX80.

The Ferranti logic array, later reprogrammed for Spectrum and QL, cut costs and defeated clone-makers. It also gave a continuous TV display, whereas the ZX80 screen was blank unless the machine was waiting for a key-press. The Z80 processor was tied up feeding it with character patterns except during vertical blanking, so the cost of this flicker-free screen was a reduction to 25 per cent of full speed.

This made the ZX81 the slowest home computer ever – with just 1K RAM, expandible to 16K with a wobbly-add-on unit – but it was also by far the cheapest in 1981, at £70 built or £50 in simple kit form. Personal Computer World called it "amazing value for money" and chain stores that had never dreamed of selling computers shifted hundreds of thousands, despite a sometimes hilarious lack of staff training.

The TS1000 was so cheap to make that it ended up as a $10 impulse buy, shrink-wrapped at American supermarket checkouts. Sinclair and Timex uncovered a vast market that no one had foreseen, and fuelled demand for more capable micros. A ZX-81, plus a cheap black and white telly and cassette deck, gave millions of people who had never dreamed of owning a computer their first taste of programming.

Linux ZX81

There is only one ZX81 emulator for Linux, though one of the oldest ZX81 emulators, *Xtricator*, runs on Qdos if you don't mind a German keyboard layout. The Linux native *Z81* comes in three different versions.

Z81 uses SVGAlib, so it requires PC clone hardware. It runs most ZX-81 programs, including those that reprogram the font to get pseudo high-resolution graphics. It loads and saves in the download format popularised by *Xtricator* and *Xtender*, but can also load files from the Atari ST ZX-81 emulator if their suffix is changed from '.81' to '.P'.

XZ81 is a version of the program for X, and is quite portable. *Z81txt* is a text-mode variant for Linux consoles. This version is not included in the latest release, 2.0, so old and new versions are on our CD, though most users will want the graphics.

By default the speed is regulated down to match the original hardware. This was about 0.8MHz, effectively, in SLOW mode, when the Z80A processor spent most of its time generating the display, in a tight software loop, and 3.2MHz in FAST mode.

Like the Spectrum, this emulator uses keyword entry where the effect of each key depends on context. ZX80 and ZX81 machines used a custom character code, so I've collected and compiled tools to convert saved files to ASCII, in the zx8utils archive on the *LXF* coverdisc. Russell Marks's *ZX81GET*, bundled with his emulator, can even read original cassette files via '/dev/dsp', with luck and a following wind.

Key points

Only the original 40 keys are emulated, in their original layout, so most of a modern PC keyboard will be ignored, outside the alphanumeric group, Enter, space, shift and full stop. Thus you have to press SHIFT P for quotes and SHIFT for a comma. *XZ81* allows Ctrl or Alt to be used in place of SHIFT, and this is recommended to avoid X keycode confusion.

To get the loading menu, type **LOAD "" (J Ctrl P Ctrl P)**. This shows all .P files in a list and allows them to be picked with Enter after scrolling with Q and A. It's a·pity more host keys – in particular the directional arrows, emulated with shifted digits – are not usable. F1 gives keyboard help – press Shift Enter to get the functions, printed under each key on the help screen. ESC reset and F10 quits the emulator.

Jupiter Ace

Not *really* from Jupiter, and not really ace...

The Jupiter Ace is an honourary Sinclair – it was not actually sold by Sinclair Research, but had an ex-Sinclair design team and was built very much like a hybrid of the ZX systems. The emulator is derived from an early version of Z81, emphasising the family connections.

When they'd finished designing the ZX Spectrum Richard Altwasser and Stephen Vickers left Sinclair and set up Jupiter Cantab to make their vision of a home computer for programmers rather than gamers.

The Ace was in the Sinclair spirit, with a flimsy ZX80 style case, Z80A, 3K RAM, Spectrum-like keys and beeper.

It's unique selling point was a good implementation of the powerful but eccentric Forth programming language, rather than the usual BASIC. Forth is a threaded compiled language, invented by Charles Moore to control telescopes, based on Reverse Polish Notation.

Jupiter's ACE looked like a Spectrum in a ZX80 case.

This makes it fast but rather a minority interest – indeed, our new CD compiler Neil Bothwick cut his teeth on one!

The limited RAM (3K, expandable to 49K), black and white TV display and dependence on cassettes didn't help, either. The Linux emulator is called Xace, currently at version 0.4, and based on Z81 version 0.2. It works, but unless you program in Forth yourself it's rather impenetrable.

Another Sinclair budget innovation was the £50 ZX printer, a tiny gadget that printed in black on narrow coils of silver thermal paper. *Z81* emulates this by generating a PBM file, authentic but for the smudging and wobbly alignment, as each line is 'printed' by the emulator. You can load the PBM (portable bit map) into Unix tools like *GIMP* for editing or conversion.

Erudite QL adventure, *Voyage of the Beano*.

ZX80

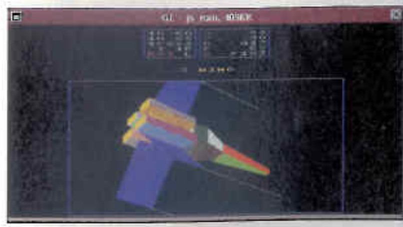
The letters in TS1000 stand for Timex Sinclair. The ZX prefix simply comes from the bottom left corner of the QWERTY keyboard, and the 81 comes from the year when the ZX81 arrived. *Z81* can also emulate the previous year's Sinclair ZX80. With half the ROM space and no facility to compute and display at the same time, the ZX80 was a much more limited system, with integer-only arithmetic and minimal string handling.

The ZX80 emulation has no file menu. The ZX80 LOAD command took no parameters, simply loading the next thing played from tape, so the emulator requires you to rename each file to 'zx80prog.p' before loading, and back again after it's been saved.

Unlike the Tardis, a ZX81 was diminutive on the inside too.

Sinclairs on CD

The freely-copyable *Qdos Emulators* CD includes *QLAY* and *UQLX* emulators for Linux, utilities, original and updated system ROMs, OCRed QL manuals and other documentation on a Joliet-format CD, plus over 300MB of Qdos software in a QXL.WIN file. It's available from Q Celt (Eire) 00353 404 45319, http://www.soft.net.uk/dj.



Dave Barker's QL *Elite* demo made good use of screen RAM paging, stippled colours and the Motorola CPU maths unit.