

## Microfair report

The Winter ZX Microfair was held as usual in London's New Horticultural Halls, just before Christmas. This was the 28th Microfair, the main event for Sinclair users. We chronicled its history back in issue 6 of *Shopper*.

The Christmas fair was more crowded than the summer one, but turnout was lower than this time last year. Once again there were about 70 stalls: slightly fewer business stands this time, and more clubs and user groups.

Spectrum traders now occupy about half the space, but QL support is steady and there were signs of growth in Z88 coverage. There was a token showing of PC200s at the front of the hall, but these surrogate Sinclairs didn't attract much interest.

New computer prices were slightly higher than in the summer, with the original 48K Spectrum at £50, Spectrum Plus around £60 with bundled software, and Sinclair's 128 at £80. Current Amstrad models sold for within a few pounds of their normal list price. QL prices varied through the day, settling around £80 for 'JM' versions and £100 for the newer but equally-bugged 'JS' model.

### ZX-81

The ZX-81 has faded from the scene recently, but there were some real bargains for 81 users this year. 16K RAM packs sold for £2 – a snip for anyone with an unexpanded 81 mouldering in the attic. Classic programs like Bug Byte's *Adventure*, *Artic Chess 2*, Psion's *Vu-Calc* spreadsheet and many Sinclair titles sold for under 20p a copy.

These are just the thing if you're preparing a 'time capsule'. One day they will be chic again; a second-hand 16K ZX-81 can give as much insight about computing as a 'budget' PC at many times the price.

### SAM

The 'Super-Sinclairs' were notably absent – there was not one CST Thor to be seen, and only an 'artist's impression' of MGT's SAM. The ROM and case are still being developed, and at the moment there is only one prototype SAM in existence, made up as a 145 chip wire-wrapped test-bed.

Designer Bruce Gordon was

just back from Japan, overseeing Fujitsu's work on the custom gate array that will replace 138 chips used in his prototype. The first public showing of the 256K 'SAM Coupe' is scheduled for 29th January, at the National Amateur Radio Societies' Association Show in Blackpool, but production machines will not be available until the end of April.

### +2A fixers

MGT, (0792) 791100, did have the 'fixer', at £9.05, to make its Plus D disk and printer interface work on Amstrad's bodged Spectrum +2A. The Fixer lets you use MGT disks with a Plus Three, in 48K mode, tripling disk speed and quadrupling capacity compared with Amstrad's non-standard drive.

The Fixer is a short board that plugs into the back of a +3, giving a duplicate edge with an improved set of port signals. It should also work with some other peripherals normally incompatible with black Amstrads, but apparently it won't sort out the Opus Discovery or VTX modem. We'll try it with our collection of plug-in widgets and report our findings.

Micronet, the Prestel club for computer modem users, has just appointed new editors for its Spectrum pages. It has their own £10 'fixer' for the VTX-5000, which fits into the modem's three-way expansion cable and should allow other incompatible peripherals to be daisy-chained – it has the Multiface 1, ZX micro-drives and various joystick ports working on the +3, in 48K mode only.

Call Telemap – the Micronet system providers – on (01) 278 3143 for details and specific info, but bear in mind that this adaptor

is no use unless you've got a VTX cable or 'gender bender'.

Micronet's adaptor has an edge-connector on both sides, so you need a socket at the computer end. MGT's Fixer fits directly between the computer and peripheral, as it has a socket one side and an edge connector on the other.

Both may cause damage if plugged into a grey +2 or Sinclair Spectrum. They should only be used with late Amstrad models.

SixWord Ltd made one of its rare public appearances with the latest version of the Swift Disk interface. The company has almost finished a 'disk doctor' program for the system, which should fill a major gap in its armoury when it is released 'early next year'.

Programmers' publisher HiSoft had its 48K/128K tape range at the show, along with Plus Three bargains like Locomotive CP/M at £25, the ZipZap disk editor at £10.95 and compilers galore, including *Nevada Cobol* at £35 on a 3" disk. Kobrahsoft had *DICE* and other Spectrum utilities on show.

Multiface makers Romantic Robot had no new products at the show, but brought news of an 80-column card developed in Sweden for all models of Spectrum. Several other UK firms have considered distributing this, but have been put off by the high production cost – the card duplicates large amounts of the Spectrum hardware, and is likely to cost 'at least £50'.

You'll need a monitor display for serious use, and existing programs are unlikely to use the extra columns without modification. The hardware is finished but RoRo has made no firm decision about distribution at this stage – to influence them, call (01) 200

8870.

We picked up a copy of Michael Slater's *Investment Monitor* – a package for computerate yuppies seeking to survive taxing times and stock market hiccups. The package runs on the Spectrum and QL, and costs surprisingly little: £12 for 48K Spectrum, £14 for ZX micro-drive, £20 for unexpanded QL and £22 for QL 3.5" disk version, which we'll review in due course.

At the other end of the economy, DJW had new versions of its *Home Banker* program for both expanded and unexpanded QLs.

### QL bargains

Many QL programs are published through the monthly magazine *QL World*, which runs a 'micro-drive exchange' of readers' code, administered by Transform Ltd. Prices are usually pretty low, but fell to a remarkable £2 per title at the fair; some used two cartridges, which means they were sold at less than the trade price for blank cartridges in quantities of thousands!

*QL World* will vanish from the news-stands after February, leaving *Shopper* as the only magazine with proper QL coverage – although there's a token QL column in the weekly *New Computer Express*. If your QL friends haven't spotted *Shopper*, now's the time to get them reading!

Tony Tebby's firm QJump had its usual stand but sub-let parts of it to German utility and games house Jochen Merz Software, which bases all its programs on Tebby's Q-PTR interface. The rest of the stand was occupied by Q-View, run by former Sinclair and QJump employee Jonathan Oakley, now a HiTech consultant.

These Sinclair people never stop hacking, and Oakley has

Continued from page 127

dreamt up a new, low-cost QL interface that should fascinate QL hardware freaks. If you're not worried about soldering four extra parts on the back of your second processor and fitting an extra light in the keyboard top, read on.

Oakley's £5 'Caps LED' modification uses 128 bytes of code and an undocumented port to give optical indication of the setting of CAPS LOCK and CTRL-F5. For details of this, and QL Viewdata software, write to Q-View, 29 Carnaby Close, Godmanchester, Huntingdon, Cambs PE18 8EE, or call (0480) 412884.

PDQL, (021) 200 2313, had two major new programs. *XREF 200* analyses and indexes the structure, name-usage and dependencies of large SuperBASIC programs. Also from the prolific fingers of Chas Dillon comes the horribly-named *BASIC-C-Port*, which automatically translates SuperBASIC into Lattice C for compilation on an IBM PC. Shame!

#### QL hard disks

Miracle Systems, (0454) 317772, continues to tempt the QL and

Z88 with new add-ons - this time it had the folded metal box for the promised QL hard disk system on display, but still no works to go inside it and no news of its software.

When it's finished 'towards the beginning of 1989' this desirable peripheral is projected to cost £399 all inclusive, for a 30 megabyte drive, interface and controlling software. When pressed Miracle said that the launch is planned for 'January or February', but says all details are subject to change in the meantime.

The Miracle interface is designed to squirt data through the QL cartridge port, so that it is compatible with Miracle's Trump Card, which clogs all the addresses in the main QL expansion connector. Miracle plans to put a duplicate cartridge port at the back of the hard disk interface, just in case it thinks of anything to add later.

It seems the QL hard disk race is hotting up, after a false start last year when *Shopper* brought early news of Sandy's Megaboard. We saw a production prototype, but the package never reached the market, following the

collapse of the Italian manufacturing end of Sandy last year. Power Computing of Bedford has taken over some of Sandy's products, but doesn't intend to develop any new QL items.

Meanwhile Germany's ABC Electronik - not at the show - has licensed CST's SCSI interface and software, both of which already work on the Thor XVI and the original QL-based Thor I. The interface can handle several floppy or hard drives, and should cost around £460 for 20 Megabytes. It works very quickly through the main expansion port, and could well beat Miracle onto the market. For details, call ABC in West Germany on (0521) 890381.

CGH Services, (055) 934 574 was touting *Starplod* and *Valagon*, adventures by Alan Pemberton. CGH has also produced a corrected and compiled disk version of *D-Day*, Games Workshop's ambitious but bugged QL wargame.

A new QL graphics utility arrived from Belgium, filling the gap between 'paint' packages like *GraphiQL* or *Eye-Q* and so-called 'desktop publishing' programs. *The Painter* lets you

design up to 12 full screens at once, in four or eight colours, with 16 character designs to choose from. It works with Sandy and QJump mouse interfaces, and features 'stepless zooming' and 'fill picture in any tetragon' (!).

*The Painter* needs 256K of memory and one 3.5" disk drive. It costs £32 from Progs, PB 238, 3000 Leuven 1, Belgium.

There's no space to list all the other stands, but it was good to see famous names from the QL/Z88 scene like Athene, Care, D.P., Tony Firshman Services, Liberation, Schoen, Sector, Strong and Ultrasoft. User groups were there in abundance, including *Shopper*-recommended groups like Quanta, QL SUB, Adventurer's Forum, Z88 Users and the Spectrum Discovery Club.

It was a good show, with many bargains to be had. There should be another Microfair in the Spring - it's well worth visiting, particularly if you haven't been before.

Sid Martin and Timothy Green  
are Sinclair afficianadoes

# QL corner

by Sid Martin and Timothy Green

**Sid and Tim report on QL editors, PC emulation, and the 'slave blocks' that speed up disks and microdrives**

## Spy

*SPY* IS a new QL editor from ARK Distribution, a small firm that started out publishing applications built on Psion's *Archive* database. *Spy* is a 22K task that can edit any type of file. *Spy* multi-tasks and lets you edit or shuffle text between several files at once.

The QL comes with a 'complete suite' of bundled software, so why do users need anything as basic as a new editor? To understand why *Spy* is useful, we must consider its heritage.

### QL editors

The first QL editor was Psion's *Quill*, supplied free by Sinclair. It was friendly but bug-ridden at first; later versions work, but they're still slow and verbose. *Quill* uses its own document format, rather than plain text files, and won't multi-task unless you load special routines to 'tame' it.

Next came Metacomco's *ED*, aimed at programmers but inefficiently written in BCPL. *ED* was supplied with many QL compilers from GST and Metacomco. It multi-tasked, but screen handling was rudimentary and file-access, searches and block moves were stiothful.

## Editor is driven by a plethora of two and three letter commands rather than menus

Chas Dillon's frustration with *Quill* and *ED* led him to write *The Editor* in compiled SuperBASIC. *Editor* was much faster than *ED* and introduced the idea of the 'lazy screen'.

Earlier programs gave priority to the display, so that every key-press produced a flurry of screen activity. It was easy to type far ahead of the program.

'Lazy' editors concentrate on responding to the keyboard and updating the area immediately around the cursor. The rest of the screen is only updated when you stop typing for a moment.

The difference is a matter of personal taste, but there's no doubt that 'lazy' programs keep up with your typing better, especially if you're not using *Speedscreen*, which dramatically speeds up most editor displays.

*The Editor* was steadily expanded and revised to handle large documents and binary files as well as program text. A command language and printer post-processor were added, so new

versions can perform complex automatic changes and formatting operations. Eventually it expanded beyond the capacity of a 128K QL, and the souped-up 'Special Editor' was born.

*Editor* is driven by a plethora of two and three letter commands, rather than menus, making it closer in spirit to *ED* than to *Quill*. It's powerful, but not friendly.

HiSoft's *DevPac* editor is relatively simple. It was written in machine-code by Andy Pennell, author of several good QL and Spectrum books. It is a very fast programming editor, particularly when it comes to loading, saving, block moves and searches. However it expects you to press 'Enter' at the end of each line, so it's not much good for word-processing.

*DevPac* can't cope with control codes, and falls over if you try to use it to edit binary files. It has an integrated macro-assembler, and a monitor. MONQL, is included in the price. These make

it particularly suitable for machine-coders, although it's also well-suited to SuperBASIC and other block-structured programming.

*SPY* is modelled on *Editor*, but written as one small machine code file. It is simpler and cheaper than *Editor*, but loads faster and leaves more memory for text.

The table 'QL Editors compared' summarises the differences between the editors. Task sizes vary between software revisions, particularly in the case of *Editor*. The table is not definitive - there are lots of QL editors around. Two that we haven't used are *Text 87*, a word processor that concentrates on displaying exactly what will be printed, and *QD*, a mouse-driven programmer's editor.

### Spy

*Spy* is a general-purpose editor - it happily edits tasks and toolkits as well as text. You can use it to 'patch' data files, typing new data over the top of old stuff.

This is an easy way to change the names of toolkit commands that clash, alter task messages or re-configure programs. For instance, *Spy* could replace 'MDV' with 'RAM' or 'FLP' in any program, or change Psion's 'microdrive' messages.

*Spy* can customise itself - just

### QL Editors compared

	Psion Quill	Metacomco ED	D.P. Editor	Special Editor	HiSoft DevPac	ARK Spy	Master Spy
ALTkey control	N	N	N	N	Y	Y	Y
F3 control	Y	Y	Y	Y	N	Y	Y
Command files	N	N	Y	Y	N	N	N
Find and Replace	Y	Y	Y	Y	Y	N	Y
Undo command	N	Y	Y	Y	Y	N	?
MODE support	4/8	4/8	4/8	4/8	4	4	4
Config windows	N	Y	Y	Y	N	Y	Y
Movable windows	N	Y	Y	Y	Y	N	Y
Multiple windows	N	N	N	N	N	N	Y
Lazy screen	N	N	Y	Y	N	Y	Y
Help display	Y	N	Y	Y	Y	Y	Y
Screen formats	Y	N	N	Y	N	N	N
Multi-tasking	N	Y	Y	Y	Y	Y	Y
Edit codefiles	N	Y	Y	Y	N	Y	Y
Formatted print	Y	N	Y	Y	N	N	N
Edit from file	Y	N	N	N	N	N	N
Multiple files	N	N	N	N	N	Y	Y
Speed	Slow	Sluggish	Q.Fast	Q.Fast	Fast	Fast	Fast
Program size	51K	21K	60K	110K	26K	22K	35K
Price	Free	Bundled	£30	£50	£30	£15	£30



EXEC the task and use it to edit its own code file. It comes with a list of patch offsets so you can configure it to use particular files, devices, windows and prompts by editing certain locations. This is a very powerful technique, but it's easy to hit the wrong key by accident and corrupt the code. You should always work on a spare copy of the file, just in case.

#### Spy initiation

When you load a file into *Spy* it reads the header and suggests a size for the text memory 'buffer'. This buffer must have room for additions to the file, plus an 'index' of four bytes for each line, like *Editor*, so the suggestion is always greater than the file length.

*Spy* itself is a 22K file which uses 24K when running, plus buffer space. The default buffer size is 34K if you use *Spy* to edit itself, so you need about 60K free to re-configure *Spy*.

You can edit the default value, or configure *Spy* to allow more or less space for changes. Normally *Spy* moans if it hasn't got enough room for the index; you must specify a bigger buffer and re-load the text.

We found a small bug in these checks. If you try to cram *Spy*'s code into a 22K buffer it seems to load - but the file menu shows the size as 4,294,965,550 bytes, and the machine crashes soon after.

To edit several files you create multiple buffers and swap between them. Only one file appears on the screen at a time.

When you load a new file the old buffer space is not released until the new one is loaded and indexed. If you run out of space in a buffer on an un-expanded QL you may need to save, quit and re-load *Spy*, unless there's room for TWO text buffers, plus the *Spy* code. If you want to edit more than about 20K on a 128K QL it helps if you get the buffer size right from the start.

#### Editing

The usual arrow key combinations move the cursor and delete by line or character, but not by word. You can move a page at a

time to avoid slow scrolling, but this looks untidy unless you adjust the default width used for 'lazy' updates.

The tab key moves the cursor to the next tab column, but tab characters in a file are not expanded. Other commands are accessed by pressing ALT with a letter, or F3 - to bring up a menu bar - followed by the first letter of an option.

You can move the cursor straight to a given line number - ideal when scanning compilation listings. *Spy* can rapidly search forwards or backwards for specific text, which you have the option to edit before each scan. F1 displays a help file, F2 repeats a search, F4 re-draws the screen and F5 switches between overtyping and inserting.

*Spy* is no word processor; it will not paginate, centre or justify text for you. You can enter all possible character codes directly, so it's easy to make use of all the features of a printer if you can remember the code numbers and insert each one manually.

*Spy* can save a file or continuous 'block' of lines to any device, so you can print all or part of a file by saving to SER or PAR.

Blocks are normally 'local' to the file being displayed, and share its buffer space. To move text between files you use the 'global' block, which has its own memory area and can be read or written while editing any file. *Spy* only allows one global block, and one local block per file.

*Spy* is simple to use compared with most editors, but has a 33 page A5 manual, printed in photo-reduced daisywheel type. The text is a clear tutorial but there's no index. Extra sheets list configuration offsets, solicit bug reports and provide a 'Quick Reference', flimsily printed on paper rather than card.

*Spy* reached us on a 3.5" disk, holding versions configured for TV or monitor displays, a trivial boot file, help text and an 'update' document.

At £14.95 *Spy* is good value. If you don't already use D.P.'s *Editor*, or find the size and complexity of that program daunting, you should find plenty of uses for *Spy*.

#### Master Spy

ARK was working on a new version of *Spy* as we tested version 2.8. *Master Spy* will sell for twice the price but has some useful extra features. It lets you move or delete by word as well as character and line, and can replace text as well as search for it. Screen-handling is improved, so you can move or re-size windows at any time, and view several files at once without loading separate copies of the editor.

## Emulator news

AFTER rumours from various quarters, the long-awaited PC Emulator for QL systems has surfaced, care of Ant Computing in Cornwall. The £49.95 program is said to work on any QL with a disk drive and at least 192K of memory. It lets you run MsDos PC software on a QL, just as Sandy's CP/Mulator runs CP/M programs.

The emulator mimics IBM's MDA text display and can read, write and format disks in PC format. It lets you assign QL SER and PAR ports to simulate COM1, COM2 and LPT1.

The emulator uses a machine-code '8088 interpreter' to run MsDos programs. This is likely to be quite slow,

as it will need to perform several 68008 instructions for every 8088 instruction emulated.

If you want to perform heavy-duty data-processing, you're probably better off converting PC files to QL format and processing them with software designed to run on the 68008. Even so, the emulator will give you access to the vast library of PC software, including thousands of 'free' Public Domain titles, so it's a major event whatever the speed.

We've sent off for a review copy, and hope to report back soon. For further information send an SAE to 'Ant Computing', Meadow Lodge, Rame Cross, Penryn, Cornwall TR10 9EA.

#### Contact

Applied Research Kernel, Corve Farmhouse, Chale Green, Ventnor, PO38 2LA UK; (0983) 79 496.

## Slave Blocks

Last month we started to discuss the QL's operating system, QDos. We explained how the QL can run several programs at one time. This month we're going to explore 'slave blocks', which speed up file-handling and avoid device delays.

As most computers run there's usually some memory that's not being used. The QL takes advantage of every scrap of spare memory, using 'slave blocks' to speed up access to drives. QDos also uses multi-tasking so it doesn't have to wait for devices.

Microdrive cartridges contain a continuous tape loop, and can take up to seven seconds to find a given data-block. Floppy disks usually take less than a second, but it is still useful if the QL can get on with something else while the device is busy. Hundreds of thousands of instructions can be performed while the disk head winds from one edge of a disk to the other.

When a program wants to read or write data on disk or cartridge, it calls QDos to do the job. Data is stored in 512 byte lumps, or 'sectors'. These preset lumps are set up when you format a disk - they make it easy to divide the available space between files. Space on a drive is always allocated in units of whole sectors.

Most computers use a fixed area of memory, or 'buffer', for the 'next' or 'latest' sector from each file. When a program asks for the next line the system tries to get it out of the buffer. If the line happens to span the end of the sector the system copies out

the first part, then fetches the next sector into the buffer, and can extract the rest of the line.

The QL approach is slightly different. Rather than keep a fixed buffer for each file, the QL uses all spare memory for buffering and postpones or avoids using the drive whenever possible.

When you turn on the QL the first thing it does - after yawning, and coughing if you've got one of the early smoking models - is divide all but screen memory into 512 byte 'slave blocks'. You get 1760 of these chunks on a fully expanded QL with a 'Trump Card'.

Each Slave Block can be allocated to programs or data, or used to buffer information en route from a file. A block may be unused, unavailable, or marked to show the file, block, drive and sector it corresponds to. When a file buffer is needed, the QL just marks one of the unused chunks as 'ready to write', 'ready to read', or 'ready to verify'.

If writing, it transfers data into the block. When the device is ready it interrupts the processor so that the data can be moved from the block to the required drive.

If reading, the interrupting routine copies data into the hole and marks it 'ready to read' as soon as the relevant sector goes past the disk or tape head. Meanwhile the task is marked 'waiting for data' and suspended. This leaves the processor free to get on with other tasks until the data arrives.

Verifying is much like reading, except the system checks that the contents of the slave block match the recorded data. If they don't match the QL tries again several times before returning an error to the task.

You can see Slave Blocks in

**The QL takes advantage of every scrap of spare memory using slave blocks**

action when you SAVE a program of a few hundred lines to microdrive. The drive runs for a short while, and then the command cursor returns while the cartridge continues to rumble and rattle away. Everything goes a little slower until the drive stops, but you can still type in and run BASIC while the QL writes and verifies your program.

#### Quick directions

Slave Blocks speed up access to QL drives, particularly when you use the DIR command or open a file without changing the contents of the drive. PC users are used to delays when they open files or display directories. Several blocks must be re-read every time the PC needs to find your current place in the directory structure.

Likewise, the QL takes a long time to respond to DIR MDV1 when you first put a tape in the drive - it must scan the whole tape, which takes seven seconds,

to find the entire directory. But the next time you type DIR the QL drive only runs for a fraction of a second before the directory appears.

QDos only needs to run the tape for long enough to check that you haven't sneakily changed the cartridge. All the directory information is already in slave blocks, where it's instantly available. The sectors were copied via slave blocks as they were read, so it's easy to find them again.

Many programs repeatedly read files - help files, directories or configuration data, for example. Even in a large file, the same block may be read several times in the course of one session - say, if you're editing a document on disk. Slave Blocks make repeated reading much faster.

You don't need to change QL programs to take advantage of extra memory. The whole file need not be slaved - just the parts you're currently using, wherever

they may be in the file.

Slave Blocks are reclaimed when the system needs memory for something else, or re-cycled when a new buffer is needed and an unused block cannot be found.

If the slave blocks are all waiting for devices, QDos suspends the calling task and gets on with some other task until the device interrupt routine makes some space.

#### One bug ahead

Once you've got slave blocks on a system, lots of other good ideas suggest themselves. Some of them are dangerous! For instance, it's easy to see that disks and cartridges may work faster if the system reads blocks ahead of time. The QL does this; as you use one block it keeps an eye open for the next one, on the basis that it is likely to come in handy.

This speeds up file-reading encouragingly, but could cause early QLs to lock up when they

run out of memory. The problem comes when an early 'AH' or 'JM' version of the QL has two slave blocks left.

QDos had the nasty habit of reading the 'next' block over the top of the 'last' one, before it had finished with it. So it had to re-read the 'last' block, which (strangely) ended up on top of the 'next' one, and so ad infinitum.

This was obviously a serious bug. Sinclair 'fixed' it hilariously in the 'JS' ROM. This worked fine with two blocks of free memory, as it should have done all along. Unfortunately it wasn't much of a fix, as it STILL didn't disable the look-ahead when only one Slave Block was free.

The 'JS' system locked up almost as often as its predecessors. It's hardly surprising so many QL owners swear that the machine was unusable till they expanded the memory.

This bug didn't get fixed till the 'MG' version of the QL ROM, which was never released to the UK market. CST's Thor XVI does not have this bug.

#### Coming soon

Next month we shall investigate Sinclair's mysterious microdrives, explaining why they work perfectly for some people and drive others up the wall! It is not giving much away to say that quality control is the crucial issue.

At least four different types of microdrive cartridge have been manufactured; we'll explain how you can tell a good cartridge from a bad one. We'll also tell you how to adjust the speed of your drives to improve reliability or squeeze up to 15 per cent more data onto each tape. Bring a sharp screwdriver!

You can also look forward to a review of *Starplod*, a new space game from CGH Services, written by *Valagon* author Alan Pemberton. If conventional adventure games have not appealed because of their limited vocabulary and slow pace, you should find *Starplod* interesting.

You point at actions and objects displayed on the screen, rather than guess at words. *Starplod* is controlled with arrow keys or a joystick, but goes at your own pace. The displays include descriptive text, lists of objects, and colour pictures of the solar system, planets, spaceships and scenes on and under the planet's surface. Find out more next month.

Sid Martin and Timothy Green  
are Sinclair aficionados