

QL Today

Volume 9
Issue 5
Jan./February
2005

ISSN 1432-5454

The Magazine about QL, QDOS,
Sinclair Computers, SMSQ...

EPSON, Canon, PDF,
HP, Lexmark, Fax
Kyocera...

NEW!

Printing problem solved!

QPC can now print to
any PC printer, even
faxes or PDF printer
drivers!

The solution:

QPCPrint

Contents

- 3 Editorial
- 4 News
- 9 QL Today adapts to Reality *Jochen Merz*
- 9 Small Ads
- 10 QPCPrint - a new Virtual Printer by
Marcel Kilgus *Rich Mellor*
- 12 Start Here - Essential Information For
21st Century QLers *Roy Wood*
- 18 QDT v1.00 - A Preliminary Review
Per Witte
- 22 Another Addition to GWASS
George Gwilt
- 29 Curve Smoothing *Stephen Poole*
- 30 Keyboards and Perfection
George Gwilt
- 36 Changes at J-M-S ... the Move part 2
Jochen Merz
- 40 QL Software Sources - Part 2
Dilwyn Jones
- 44 Too Complex? *Dilwyn Jones*
- 46 Programming in Assembler - Part 13
Norman Dunbar
- 51 A Sense of History *Geoff Wicks*
- 53 Byts of Wood *Roy Wood*

Advertisers

in alphabetical order

Jochen Merz Software	21, 37
QBranch	38, 39
Quanta	11
RWAP	25
TF Services	31
Geoff Wicks	53

QL Today

ISSN 1432-5454

German office & Publisher:

Jochen Merz Software	Tel. +49 203 502011
Kaiser-Wilhelm-Str. 302	Fax +49 203 502012
47169 Duisburg	email: JMerz@j-m-s.com
Germany	email: QLToday@j-m-s.com

English office:

Q Branch	Tel. +44 1273 386030
20 Locks Hill	Mobile +44 7836 745501
Portslade	Fax +44 1273 381577
BN41 2LB	email: qbranch@qbranch.demon.co.uk
United Kingdom	email: QLToday@j-m-s.com

Editor:

Dilwyn Jones	email: dilwyn.jones@tesco.net
41 Bro Emrys	email: QLToday@J-M-S.com
Tal-Y-Bont, Bangor	
Gwynedd	
United Kingdom LL57 3YT	

Co-Editor:

Bruce Nicholls	Tel +44 1708 510764
38 Derham Gardens	Fax +44 870 0568755
Upminster	email: qltoday@q-v-d.demon.co.uk
Essex RM14 3HA	email: QLToday@j-m-s.com
United Kingdom	

QL Today is published bi-monthly, our volume begins on beginning of June. Subscriptions begin with the current issue at the time of sign up. Please contact the German or English office for current subscription rates.

We welcome your comments, suggestions and articles. YOU make **QL Today** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email. We prefer ASCII, Quill or text87 format. Pictures may be in .SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

**The deadline for Issue 6 is the
20th of March!**

QL Today reserves the right to publish or not publish any material submitted. Under no circumstances will **QL Today** be held liable for any direct, indirect or consequential damage or loss arising out of the use and/or inability to use any of the material published in **QL Today**. The opinions expressed herein are those of the authors and are not necessarily those of the publisher.

This magazine and all material within is ©copyright 2004 Jochen Merz Software unless otherwise stated. Written permission is required from the publisher before the reproduction and distribution of any/all material published herein. All copyrights and trademarks are hereby acknowledged.

OK, what hit me?

I'm not referring to some of the verbal punch-ups that sometimes happen on the ql-users email mailing list, but rather I turned my back for a little while in January 2005 and all of a sudden a whole new QL scene has appeared out of nowhere.

QPC2 gets TCP/IP access and I've been browsing the web with Lynx. Jon Dent and Marcel Kilgus swap SBASIC programs for reading emails on the mailing list. Yes, SBASIC. Suddenly, QPC users can print to supposedly Windows-only printers with the new QPCprint utility. QDT sneaks out. So does the long awaited Q-Word game, a feast of colour and sound on modern QL systems. Balled, a colourful new game from Wolfgang Lenerz is quietly released. New versions of SMSQ/E come out. Pointer Environment and Toolkit 2 can now be distributed as freeware.

I rather thought that it was taking a bit longer than usual to write the news columns this month!

It all started when I found myself on the receiving end of a QPC2 v3.30 beta test copy. At the time, I didn't think too much of it, as I was used to regular updates of the emulator. It soon dawned on me that this was a major update as I saw the magic term (for me anyway): TCP/IP.

What Marcel has done is bolted on a system quite similar to that in the uQLx emulator, where TCP/IP access is done via the underlying operating system, Windows in the case of QPC2, Unix or Linux in the case of uQLx. It is well documented for uQLx users, so since Marcel's implementation in QPC2 is broadly compatible, so QPC2 hit the ground running. Then I came unstuck trying to use Lynx, currently the only QL browser, which turned out to be a bit harder to set up than I'd thought. I've got it going, and if I've fully sorted it out hope to write about my experiences in the next issue. Next thing, I saw Jon Dent (author of soql, another internet access software system for ordinary QLs) and Marcel Kilgus trade SBASIC listings showing how to access emails. This was getting scary - email from SBASIC? Not tackled

that yet, but you never know, Lynx was a major achievement for me.

Then Marcel pops up again with something called QPCprint, which allows QPC2 users to print to supposedly Windows-only printers. Hooray, I can finally print to that nice Samsung laser printer in my office at work! More from Rich Mellor about that in this issue. Not to be outdone, Q-Word and QDT both finally made their public debut. Both major works of QL software. It is so encouraging to see such new programs using all the latest bells and whistles (probably literally in the case of Q-Word!). We also got some games from Wolfgang Lenerz and Ken Tait, just to keep us amused!

Then Roy Wood tells me that as a result of discussions with Tony Tebby (and what a debt of gratitude we owe to him over the years!) the pointer environment files ptr_gen, wman and hot_rext, along with Toolkit 2, can now be released as freeware and may be included with QL emulators.

So much good news all in one month. Yet Sir Clive stopped making QLs roughly 20 years ago. So what went wrong? Nothing. It's called enthusiasm and belief. QLers are a determined and independent lot. We've learned to help ourselves and each other. We stay in touch via a combination of the old and new - magazines and the internet. A loyal core of traders and committed software authors and hardware designers along with a strong (if sometimes undeservedly criticised) user group means we're in a pretty healthy state after 20 years and will be around for a while yet.

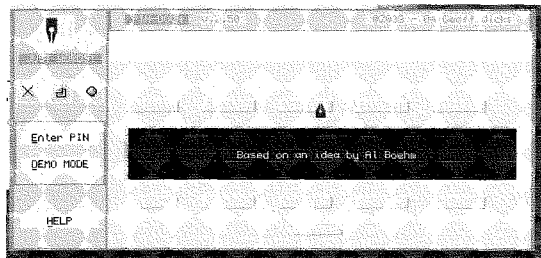
Speaking of digs at Quanta, try this one. I know we in Wales now have our own national Assembly Government. But honestly Quanta, it's not so bad that you have to put airmail stickers on the envelope containing my copies of the newsletter!!!

Even though it'll be February by the time you read this, I'll wish you all a very happy new year, assuming you've not already abandoned the editor's page in search of all this great news further on in the magazine!

NEWS

PIN-DOWN UPDATE

Just Words! has now released a GD2 version of PIN-DOWN. A demo version of the program, restricted to randomly generated 4 digit PIN-codes, can be downloaded from the Just Words! webpage.



Next in line for updating, but behind schedule is QL-RHYMES. AUTO-GRAPH will take even longer. <http://members.lycos.co.uk/geoffwicks/justwords.htm>

RWAP Software NEWS

Unfortunately we have had to move back to the Midlands temporarily and until further notice can be contacted at:

26 Oak Road, Shelfield, Walsall, West Midlands WS4 1RQ

Tel: 01922 691607 or via our website: www.rwapsoftware.co.uk

The best news is that QWord was released just before Christmas and is available now - see our new advert and full colour flyer for some great screenshots and prices. Please note that there is a separate version for QPC2/QXL, Q40/Q60 and Aurora. All versions require the high colour version of SMSQ/e except for the Aurora version which will run in full colour under QDOS and Minerva.

The SBASIC/SuperBASIC Reference Manual is again once available (thanks to Marcel's QPCPrint utility - see our review of this elsewhere in QL Today). We have however, decided not to produce any more printed copies of the Reference Manual for the time being and this version is now available on CD-ROM as Adobe Acrobat files for £20 plus post and packing.

If you have a printed version of the manual, now is the time to upgrade it to the latest release version (REL 4) and offer it on the second hand market for those who do not have the means to read CD-ROMs on their QL.

SMSQ/E News

Wolfgang Lenerz writes:

SMSQE version 3.09 is out now.

This introduces a few bugfixes. Notably, the cachemode module for the Qx0 is now implemented correctly and configuring the cache for SERIALIZED, WRITETHROUGH or COPYBACK will turn it on/off according to your wish.

Please note that in most cases SMSQ/E for the Qx0 will be shipped with the caches OFF (SERIALIZED). You can configure this easily, of course.

SMSQ/E will no longer enable you to use RAM9... - only RAM1 to 8 are allowed.

Marcel Kilgus also implemented a HOT_GETSTUFF\$ extension (across all versions) which allows you to get at the stuffer buffer from Basic.

Finally, Bruno Coativy implemented another treat for the QXL version: You can now again get back at SMSQ/E from DOS by typing "SMSQE /".

Where do we go from here?

Version 3.10 of SMSQ/E should see the implementation of a "home directory" which can be obtained by the programs aware of this feature.

QBranch News

We have a reasonable amount of second user stock at the moment so, if you want to upgrade to a Super Gold Card or buy a Gold Card/Trump Card as a spare now is the time. We have Brand New Auroras and One second user Aurora in stock. We also have a couple of Qubides, two Cub Monitors and several standard QLs - all tested and working.

We are currently shipping SMSQ/E v3.09. The Q40 version of this has an extra config item which will allow you to configure the cache modes. It is important to note that, in order to blow this as a ROM you will need to have all the caches switched off. In this mode it is very slow. We will be shipping it with the WRITETHROUGH cache enabled which will bring it back to its normal speed. Enabling the COPYBACK cache will make it faster but will cause problems with some QLiberated programs.

Printer Control Program

Wolfgang Lenerz reports that there is a new beta version of the Printer Control Program (formerly known as Proforma Printer) on his "website".

Thanks to Per Witte's extensive testing, this is now close to release (at the time of writing this), once Wolfgang has implemented one or two

more features.

This program uses the Progs Proforma engine and all its associated printer drivers (as supplied with Prowess) to implement a powerful file printing system, and may be downloaded from this address:

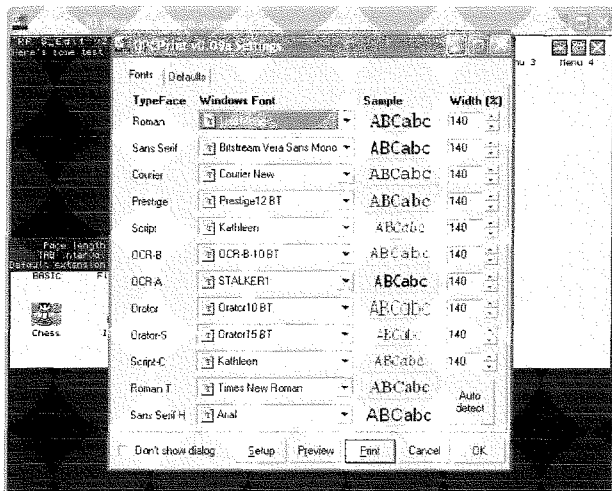
<http://www.scp-paulet-lenerz.com/14mljkl24/wolf/download/>

QPCprint

Thanks to a new program by **Marcel Kilgus**, QPC2 users can now print to printers which could not previously be accessed, so-called Windows-only printers. These printers have had their intelligence stripped from them as operating systems and computer systems became powerful enough to drive the printers themselves directly without the printers knowing about control codes in the traditional sense.

QPC2 has always had a facility to link in a printer filter in the printers dialogue part of QPC2 configuration, but until QPCprint came about, nobody had built a complete working filter.

This is a program which sits between QPC2's printer ports and the Windows printing system and printer drivers. The program running on QPC2 prints Epson-style text or graphics and this is converted to a format suitable for printing via the current Windows printer driver, in other words it's a filter program which accepts Epson ESC/P2 printer codes in and converts to a format which can be handled by the Windows printer driver. End result? A printer like the one I have in my office at work (A Samsung ML-1210) can now be printed to by QL software, where previously only Windows could use it. So those cheap little inkjet printers devoid of any type of control codes suddenly becomes accessible to us QLers. In theory at least, even Marcel Kilgus couldn't possibly test every single one of those around!



QPC2 v3.30

Not being content with solving our printing problems, Marcel Kilgus has also been busy with his QPC2 emulator. The latest v3.30 now brings us fully working TCP/IP facilities. This is touted as being very compatible with the implementation in uQLx, only this is for Windows, not Unix/Linux.

With the use of Jonathan Hudson's Lynx browser and Email programs, QPC2 users can now access the web and email from their QPC2 emulators. Sadly, like QPCprint, this is for use with QPC2 only at the moment anyway, as it uses the underlying Windows facilities.

While the TCP/IP news is probably the best part of the news, I'm pleased to report that as well as being somewhat faster than previous QPC2s (apparently QPC2 v3.30 can be up to 40% faster than v3.20), this version can handle up to 8 parallel ports (not quite sure why unless it's for people like Jochen Merz with printers and modems and all sorts of gadgets hanging off every port imaginable on their computers), and there are a couple of new SBASIC extensions for dealing with the printer filter names.

SMSQ/E VERSIONS

Wolfgang Lenerz, the SMSQ/E registrar, has again been busy with updates to the SMSQ/E operating systems. With several versions concurrent for several systems (QXL, Qx0, QPC2, Gold Cards etc) this must be a nightmare to keep track of, so I'll print a list of documented changes to each recent version here to help you to decide if it's worth updating.

SMSQ/E Version 3.07

New keyword EX_M use like EX, but the job created is owned by the calling job (wl).

Q40 flp density is REALLY preset to High (1.44 MB disks). If you want to... ..format other density disks, set the density to D or E. (wl).

QXL version allows you to determine which PC drive is win1_ etc... (BC).

QPC SMSQ/E sampled sound system (SSSS) support (MK).

QPC small fix in FORMAT of removeable WIN drives (MK).

SMSQ/E Version 3.08

QPC internal version only: new basic commands.

SMSQ/E Version 3.09

HOT_GETSTUFF\$ added (MK).

Qx0 code is ROMmable again (wl).

Qx0 cachecontrol module revamped (wl).

No directory drive nbr > 8 allowed (wl).

QXL version bugfixes and restart possibility (BC).
 QPC new basic commands (QPC_WINDOWSIZE, PAR_GETFILTER, PAR_SETFILTER) (MK).
 Outline move spurious outline problem fixed (wl).
 The letters in brackets are the initials of the developer concerned - WL is Wolfgang Lenerz, MK is Marcel Kilgus and so on.
 SMSQ/E sources are available from Wolfgang's website, along with the definitive list of changes, at

www.scp-paulet-lenerz.com/smsqe

LYNX Browser

While taking my first tentative steps to getting online from QPC2 while testing Marcel Kilgus's TCP/IP system recently added to QPC2, I became a little bit frustrated with trying to get going with the QL version of Lynx.

The full package includes source files, documentation, runtimes, help files and is a pretty large package all round (rather like me, you might say). I ran into some difficulty decompressing the archive I downloaded from Jonathan Hudson's website, since it seems to have been put together on either a non-QDOS system or at least a filing system which allows very long path names, as it seems to have a habit of truncating and renaming and overwriting some files because of the QDOS filename length issues.

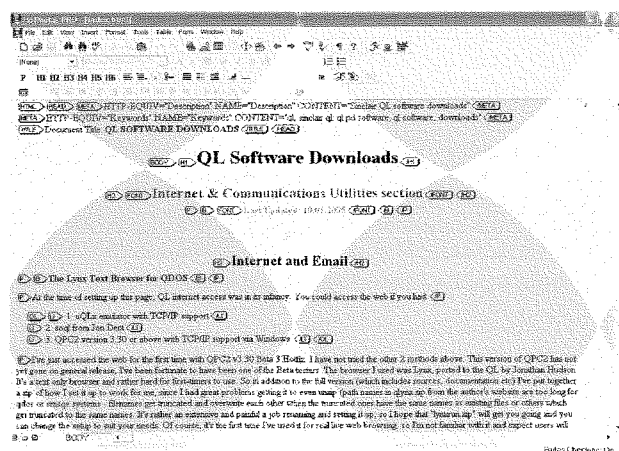
With help from a few others (what a great lot QLers are, where else would you get so many people all diving in to help a fellow Qler in trouble!) I put together a cut down package of the bare minimum set of files required to get online with the new QPC2, the only part which I haven't yet fully sorted out is the Help files, only part of which are working at the moment until I figure out where I went wrong.

I've put this package onto my website at

www.dilwyn.uk6.net/internet/index.html

Here you can download both the full 2.4MB archive with everything (sources and all) or my minimal runtime system configured for use with QPC2. That version is about 480KB of download. While configured in this way, it seems to work on my system, I cannot guarantee it will work on yours, but having been pared down to just about minimum it should make it easier to adapt to get it running on other systems if need be. The configuration files etc are actually fairly well commented (some are plain text files so you can load them into your favourite QL editor to see what's going on and edit them as required). Whether it works 'out of the box' or needs slight reconfiguration for your system, I hope it does help you to take advantage of Marcel's work in providing

TCP/IP on QPC2. Lynx is a text browser, but handles non-text pages such as my website pretty well. It is quite possible to navigate my website and download programs with Lynx, for example. The basic way in which you get online is to establish a Windows connection to the internet and QPC2's software (Lynx in this case) uses that connection to access the web.



FDISK for Q40/Q60

Jérôme Grimbart wrote to the mailing list:

"I have now put online my PE version of Fdisk for the Q40/Q60. It is available at

<http://jgrimbert.free.fr/ql/download.html>

Beware: it is missing a tutorial, and it can mess up your whole disk if you do not know what you are doing. (It does not make a backup of the partition table: once you have requested to write it, it is written. Until then, you are pretty safe)

It will edit only the drive whose mounted in win1_, (usually the primary drive) so if you want to edit a slave drive, use win_drive 1,1,0 beforehand.

Sources are also provided.

If someone happens to write a tutorial for it, I would be glad to insert it into the zip file (as well as making a web-page of it)."

Q-WORD

Just a quick note to let everyone know that v1.02 of QWord for QPC2, Q40/Q60 and Aurora has finally been released.

Cost is £20 for each version (although you can get another version for a different machine for only £5 more).

The QPC2 version is supplied on CD. Q40/Q60 version and Aurora versions are supplied on HD disk - Aurora version will even install to HD disk!!

TURBO TOOLKIT Manual PDFs

Phoebus Dokos reports that he has finished two editions of the Turbo Toolkit manuals prepared as Adobe Acrobat PDF files for those able to read such files.

The first one is in a booklet format (ready to print and be bound) here:

<http://uqlx.dokos-gr.net/turbo335-booklet.zip>

and the other in single page (A5) format here

<http://uqlx.dokos-gr.net/turbo335-single.zip>

He says that it was originally done for his own use as he hated the plain text format manuals you can download, but then decided to unleash these on the QL community. Thanks Phoebus. I think.

QUANTA Show Dates

Geoff Wicks has taken on the role of show co-ordinator for Quanta and is attempting to co-ordinate the QL show dates for next year. This is to avoid unfortunate clashes similar to that between the Quanta AGM and the USA show last year. It is hoped that in this way Quanta can again become a co-ordinating body for show dates.

Geoff has prepared a document in the form of a planner. The intention is that the planner will be updated constantly and that it will be a "rolling" planner. In other words at the end of March the first quarter of 2005 will be deleted and the first quarter of 2006 added.

He hopes to post this document on the internet so that the information is freely available to all QL-users and show organisers. Please get in touch with Geoff if you are organising a show and wish to have your show and dates considered. Contact him at the address in his Just Words adverts.

QL Graphics Conversion

I've updated the Graphics page on my software download site to include the BMP and QL high colour screen conversion and other graphical utilities, some of which have appeared in QL Today articles. The upload includes a copy of the article from Vol 8 Issue 3 of QL Today and the SBASIC listings showing how to convert between 24 bit BMP files and QL mode 32 and mode 33 screens in both directions.

I hope this helps anyone trying to get their QL to handle PC graphics, or to transfer QL graphics to PC-style files.

Point your browser at this URL, then steer to the Graphics page.

www.dilwyn.uk6.net

Pointer Environment and Toolkit 2

Roy Wood writes:

"After a discussion with Tony Tebby, Jochen Merz and I have his permission to release the three Pointer Environment extensions HOT_REXT, PTR_GEN and WMAN as freeware. They will now

be freely distributable. Copyright will remain with Tony Tebby. I have some documentation with explanations of the Keywords etc. which should be bundled with them.

We have also got Tony's agreement to do the same for Toolkit II. At the moment I do not have a copy of this which can be LRESPR'd or a text file to accompany it but I hope to have both soon. OK

This means that these files can now be included with emulators free of charge.

It should be noted that the Menu Extensions remain commercial as they are still being supported and developed by Jochen Merz Software. Anyone wishing to distribute these with their software should contact him for a licence.

I hope you will all agree this is good news."

After receipt of this email, I placed the most recent versions I have onto the Toolkits page of my website at

www.dilwyn.uk6.net/tk/index.html

At the time of writing, these were not necessarily the latest versions but were quite adequate for everyday use. I hope to add more recent versions as they are made available to me.

Clavier Update

Wolfgang Lenerz has released a new version of the "clavier" program via his website. Clavier is a program for creating and editing keyboard tables for SMSQ/E systems. If you have a keyboard designed for a language or country not currently supported in SMSQ/E you may be able to use Clavier to create your own. Files created can be used via the KBD_TABLE extension in SMSQ/E.

www.scp-paulet-lenerz.com/14mljkl24/wolf/download/

DISKMATE 5 now Freeware

P I Monstad, author of the disk and file handling program DiskMate 5, has now kindly released the program as freeware. The package includes an English and German manual, and may be downloaded from

www.dilwyn.uk6.net/files/index.html

as a 254KB zipped file.

Also available on the same page is a slightly hacked version from Marcel Kilgus, allowing use of GD2 colours.

QDT is Here!

After all the publicity and hype and the cover disk demo, the long awaited full release of QDT (QL DeskTop) is finally here!

This is available from Jochen Merz and Q Branch

<http://smsq.j-m-s.com>

and

<http://www.qbranch.demon.co.uk/News.htm>

and further information can be obtained from Jim Hunkins web site:

<http://www.jdh-stech.com/>

A new demo version is also available from these sites and is freely downloadable.

J-M-S News

Balled

A new game "Balled" from Wolfgang Lernerz is now available from **Jochen Merz Software** - it works both in hi-colour and "normal" mode 8 and takes full advantage of the hi-colours, of course. And if you have a recent version of QPC or Q40/Q60 then you get better sound as well.

Please have a look at

smsq.j-m-s.com

for details and a screenshot.

The BBS CD

As written in earlier issues of QL Today, the planned BBS CD which contains all the public contents of the JMS BBS has become reality. It contains one QXLWIN file with all the files from the BBS. Price: 5 EUR incl. postage. The BBS is still online (due to the major delay of my move) but will be shut down as soon as the office moves as announced.

New internet update features are under development and already seem to be working very well. As soon as it goes online to the public, there will also be some updates waiting for you to be downloaded.

QL OTHELLO

After 15 years away from the QL scene, Ken Tait has re-released his Othello game for the QL. At the moment, it does not work too well on Minerva versions around 1.97, probably due to a compiler issue, but seems to work on most other systems including QemuLator with a QDOS ROM image and QPC2. It does not require pointer environment. It plays a pretty strong game with 5 available playing levels, with the option to change sides during a game if you are not doing very well(!) and even to set up any board position and playing from there if you want to be certain of beating the computer or wish to study the possibilities of given board situations. You can step back and forward between all moves made so far and even replay the game played so far so that you can try to see where you went wrong when you realise you are suddenly on a hiding to nothing. Othello is quite easy to use, but quite an addictive game.

The game is freeware and may be downloaded from

www.dilwyn.uk6.net/games/index.html

from where you can also click on a link to see instructions both for this game and how to play Othello. In common with all programs on the website, it will also be available from the PD library services on disk for those who do not have access to the internet.

The same author has hinted he is looking at releasing another of his early QL software, a large chess database which may be of interest to QLers who use any of the 3 or 4 QL chess packages I am aware of (Psion Chess, Black Knight, GNU Chess or QL Chess). This package is apparently not tied to any one chess program, it is a database of opening moves, strategies and the like and a derivative was released for Atari computers at one point. While this isn't yet 100% confirmed, the author has indicated he is willing to consider this now that QemuLator has whetted his appetite for QL software once more!

Per Witte Website Update

Per Witte's QL website (Knoware) has been updated at

<http://knoware.mysite.freemove.com/index.html>

A.. 2005/02/06 Update

a.. Qwirc V0.66 - update and more bug fixes

b.. LX2 V0.03 - tiny but important change!

See Readme for details.

c.. Msprv V0.07 - bug fix and small update

GWASS Updated

The latest version of Gwass assembler is now available on the SQLUG website. This is the version that George Gwilt has been using to assemble the modified sources of SMSQE & Minerva. It includes expressions handling.

www.jms1.supanet.com

(Do not confuse the 'JMS' in the URL with JMS of Jochen Merz Software, it refers in fact to the initials of the website owner, John Sadler of the Scottish QL User Group.)

OBITUARY

It is with regret that we have to announce the recent death of Bill Richardson's wife, Dorothy. We at QL Today would like to extend our condolences to Bill and his family at this sad time.

QL Today adapts to Reality

Jochen Merz

Dear readers - the time for a change has come ("Oh dear!"). And I want to be absolutely honest with you ("That means really bad news!").

No, not bad news at all! QL Today will soon be starting its 10th volume. Wow!

I never thought it would last that long when I started it. And with the fireworks of good news for us QLers - just look at the releases of new software in January! - it seems we have more ideas in the scene than in the whole of 2004.

Roy and I are faced with several problems to consider, but the majority of this consideration affects me.

Here are the facts: QL Today's price has been stable for two years now. Postage has gone up, UPS transport costs have gone up and are likely to be increased again. We don't, however, want to have to raise the subscription price. In the past nine years QL Today always appeared in time - unless it was deliberately moved (announced in advance) to be released at a QL show. You cannot imagine what kind of pressure that put on me: whatever amount of material I had, a full issue had to be delivered. No matter what was happening in my personal life I put every effort in getting the magazine out in time... illness etc. actually counted less.

I discovered the hard way that I need to adjust priorities and make the release more flexible. I am talking about a corridor of, say +/- two weeks. If

there is enough material, the issue will be released earlier, if there's not enough, I just wait a bit - with less pressure. If something important such as my health gets in the way I'll delay it a week - and if Roy and I find we can meet up anyway at a show then we will release it at that show.

It has been a real pleasure producing the last two issues, for example. I can recall very well, however, that the summer issues in the last two years were extremely hard - there was a general shortage of material. This is understandable, as, even for the most dedicated QL enthusiast, the computer takes a back seat in the summertime everywhere. But why shouldn't we take account of this fact? The computing "season" is in the "dark months" (October-March), so we will stick to bi-monthly issues here, but for the summer month we decided we should drop one issue. This way we can not only adjust the subscription price DOWN although we take the cost increase into consideration, provide you with a magazine with the same quality (and probably content) as before with a better timing for events and also ensure that the producers of QL Today can actually continue to produce it for you (and get a summer break too). Looking forward to Volume 10 of QL Today and hope that you join us!

The following table reflects the price changes for a full year's subscription:

Destination	Old price	New price
UK	36 EUR	31 EUR
NL + D	39 EUR	33.50 EUR
EUROPE	42 EUR	36 EUR
WORLD	48 EUR	41 EUR

Small Ads

For Sale

- * Sinclair QL in its the original complete box (grey cover and polystyrene). It runs pretty well. Has no problems. Membrane is OK.
- * 12 mdvs cartridges formatted december 2004.
- * The four original PSION programs (in 4 mdvs): EASEL,QUILL,ABACUS and ARCHIVE (v 2.21, spanish).
- * Monitor (12" monocrome, green), in original box (thick polystyrene + air-bubbles-paper inside).
- * Sinclair QL user and programmer manuals (spanish)
- * Two covers each for 4 mdvs
- * Technical drawings (circuitry, repairs,ecc)
- * Digital Photos available

Manuel Barbadillo

c/ Valparaiso, 23-30B

E-41013 Sevilla

SPAIN

EMail: rankus@us.es

QPCPrint – a new Virtual Printer

by Marcel Kilgus

Rich Mellor

As many people will have spotted in the over the past few months, the problem of QLs printing to modern printers has been a thorn in our side for a while. As printer manufacturers strive to cut costs by making their printers less intelligent, the QL community is in danger of losing its ability to print.

The problem is two fold – most modern printers no longer accept plain text sent direct to them (lasers tend to be an exception, as do some of the Canon printers). – Epson for example, only make one inkjet which can handle plain text – the A3 wide 1520.

To some extent, this has already been resolved by Wolfgang Lerner's PFF device (see QL Today Vol 9 Iss 4) which captures plain text and processes it using Proforma so that graphics output is sent to the printer. So far as I know, no-one has taken the plunge to try this on a modern low cost Epson, so as yet it is unknown whether this does solve this issue. The other issue is how to connect the printer – most come with a USB connection rather than a parallel port. This is unlikely to be resolved easily for a standard QL, although possibly it will be for Q40/Q60 users.

However, Marcel has taken the bull by the horns to develop a solution which was originally designed to resolve these issues for QPC2 users. He developed QPCPrint.

The easiest way of using QPCPrint with QPC2 is to use the configuration control panel in QPC2 to attach QPCPrint as

a filter on PAR1. Now any information sent to PAR by the programs running in QPC2 will be captured by QPCPrint.

You need to ensure that your programs are configured to output to PAR (or use PAR_USE SER if you like) and use an Epson printer driver.

QPCPrint then captures this data and converts all of the standard Epson control codes to produce a document which can then be printed via any installed Windows printer driver in one easy step. Therefore if you can attach a printer to your PC, all your problems are solved.

I have tried QPCPrint with a whole host of QL programs, including LineDesign, Text 87 and Xchange and all of their output (including colour pictures) is converted successfully by QPCPrint. Marcel has gone to great lengths to ensure that the layout of the final printed document appears the same as that which would have been

produced by my printer connected to the QL natively, except in one area, the typeface used – this is because this can vary greatly from printer to printer and Marcel has had to try

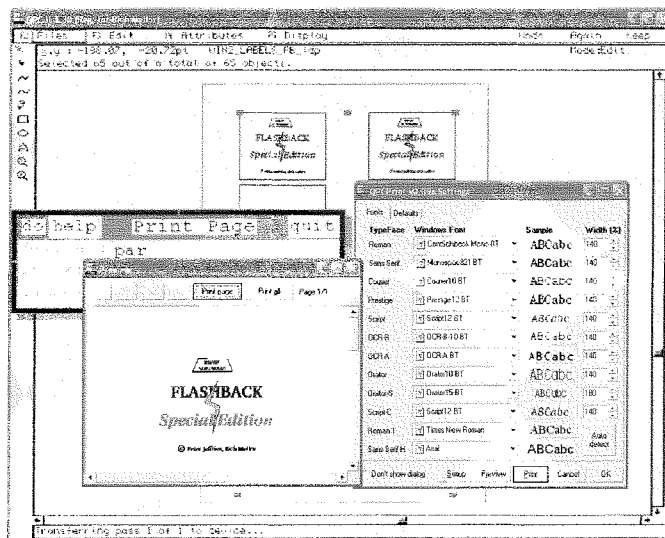
to match Windows fonts to those used by standard Epson inkjets. You can, however, con-

figure the fonts used by QPCPrint if they are not to your liking.

When the QL prints plain text, it expects the printer to have various inbuilt fonts. The appearance of those fonts may depend on your actual printer, so QPCPrint comes together with a few standard fonts (there are bound to be many more already installed on your PC) which replace the standard Epson ones. You can easily select which font you think is best to use to replace the Epson ones (although the defaults are adequate for our purposes), which allows you to have much more control over the output that you would have from within a native QL program.

However, all this does not have to concern the average QL user – all you see is that when you print from within a QL program, a configuration box comes up on screen, which allows you to select the device to print to (Setup), Preview the output on screen (Preview) or send it to the printer (Print).

You can also Cancel if you decide that it is not quite what you want.



You can even decide not to show the configuration dialog each time that you print from a

QUANTA



Independent QL Users Group

World-wide Membership is by subscription only,
offering the following benefits:

- Bimonthly Newsletter – up to 40 pages
- Massive Software Library – All Free !
- Free Helpline and Workshops
- Regional Sub-Groups. One near you?
- Advice on Software and Hardware problems
- Subscription just £14 for UK members
- Overseas subscription £17

Barclaycard: Visa: Access: MasterCard: Accepted

Now in our Twenty Second Year

Further details from the Membership Secretary

**John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH (UK).
Tel. +44 (0) 161 865 2872**

or

**Visit the Quanta Web Site
<http://www.quanta.org.uk>
E-mail: membership@quanta.org.uk**

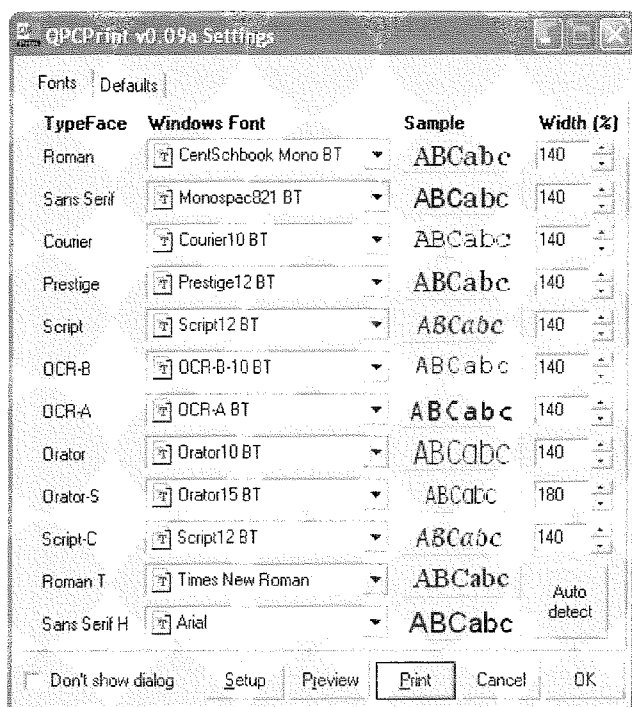
Next QUANTA Sponsored Event

**SUSSEX QL USER GROUP
PORTSLADE WORKSHOP & QUANTA AGM
Portslade Town Hall,
Mile Oak Community Centre,
HOVE,
SUSSEX.**

Sunday 17th April 2005

From 10.00 am to 4 p.m. AGM at 2.00 p.m.

Ring Roy Wood on 01273 – 386030 (or 07836-745501) for full details.



QL program, in which case, the output will be sent direct to the printer, without you realising that you are using QPCPrint.

One way of proving just how QPCPrint adds a whole new range of flexibility to the QL, I decided to try putting it through its paces to see if I could find a way of converting the SBASIC/SuperBASIC Reference Manual from its original Text87 format into Adobe Acrobat.

This proved quite easy – I needed to download Adobe's Postscript printer and install that on my PC first of all, as well as get hold of a copy of Ghostscript which can read this format (and convert it to Adobe Acrobat). All I then had to do,

convert the 1000 page SBASIC/SuperBASIC Reference Manual sources.

What About other Emulators?

One of the best features of QPCPrint, is that you can set it up to capture any output sent to the LPT1 port on your computer (for example, by DOS programs or other QL emulators) and convert this. There are other similar products available, but we have not found any which can cope with all of the standard Epson control codes and they are mainly designed to convert plain text and certainly not graphics.

Setting up QPCPrint to do this is fairly straightforward provi-

ded that you follow the instructions provided carefully, although it does require that you download and install some additional freeware from the internet. I have also only tested this on a Windows XP computer, although I understand that it should also work under Windows 95 and Windows 98.

The only slight problem with this is that if you have an actual printer connected to the parallel port on your PC (LPT1), then you will not be able to use QPCPrint in this way, unless you set it to capture LPT2 instead. If you do this, then you will need to ensure that output from your favourite QL and DOS programs is sent to the LPT2 port instead (for example, use

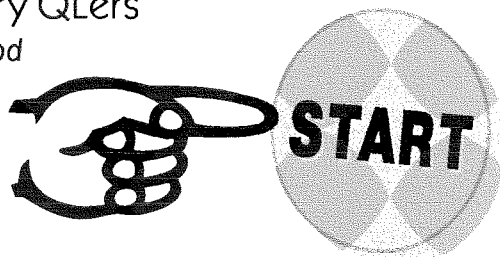
OPEN #3,"PAR2" within Qemulator rather than OPEN #3,"PAR"). This makes QPCPrint an extremely useful utility to have on your PC and will enable QL users to use any modern printer from within their programs without having to worry about someone writing a new printer driver or whether they have the right hardware to connect a printer.

QPCPrint will soon be available from J-M-S - watch Jochen's homepage SMSQ.J-M-S.com and, of course, Marcel's homepage www.KILGUS.net and the newsgroup! - Jochen

Start Here

Essential Information For 21st Century QLers

Roy Wood



Introduction. (Stop me If You've Heard This Before)

Some of the QL Today subscribers may find this rather basic stuff and consider it to be a waste of magazine space but we have always tried to cater for a range of users with a range of abilities. It is obvious from the phone calls, emails, comments we get at workshops and from David McCann's letter in the last issue that some people have missed out sections of the learning

curve needed to access the newer hardware and emulators. It is equally obvious that some of the documentation for these innovations is maybe not as explanatory as it could be. I hope that we can fill in a few gaps. If there is anything you would like us to cover please get in touch and we will try to add it.

The purpose of this series of articles is to help users to make the transition between a standard QL system with memory expansion and disk drive(s) to a system which has a hard disk and possibly extended resolution and colour drivers. Increasing numbers of QL users have bought Aurora, QXL, Q60 or QPC2 systems and some of these people have either never attempted to write a boot file of any description or have failed to get to grips with the new facilities available to them. Other, newer, hardware, such as the Rom-Disq, SuperHermes and the use of CD readers and ZIP drives on the Qubide could also do with some explanation. If we fail to convey this information to the users it is a waste of a very innovative body of work by a lot of dedicated people. I will limit the first articles of this series to SMSQ/E and make the assumption that the user has version 3 or later since upgrades from older versions of SMSQ/E to the newer versions are essentially free. Maybe other authors will contribute addendum to cover other operating systems such as Minerva, UQLX and QDOS Classic. This article will be wholly SMSQ/E based since, as far as I am aware, SMSQ/E is the only system to offer extended resolutions.

In this series of articles we will cover a multitude of things which people seem to have problems relating to.

These would seem to include:

- Screen Resolution
- Hard Drives and Partitioning
- BOOT files including the loading of extensions
- SMSQ/E Configuration Settings
- Uses of the Colour Drivers
- Printing
- Networking

The list will be added to by request.

Part I - If What you See is not What You Want.

The big problem is where to start. In order to tackle some screen resolution problems we would first have to have the means to use Menuconfig, The level 2 configuration program, or to have a BOOT file which changes the resolution. If

we take the first path and use menuconfig we would have to have the menu extensions loaded and that would require a BOOT file. You can see where this is going - there is no logical starting point.

With that in mind I will start with the screen resolution.

**** WARNING **** Before I start this let me issue one word of warning. If you start from a higher resolution and try to scale down to a lower resolution you can run into problems on all versions of SMSQ/E. The end result will be, at best, unpredictable, depending on which programs are running at the time and what size their windows are, and in the worst case fatal to the O/S. Always start from a lower resolution and increase from there. It is generally the best idea to start from the position of having nothing loaded except SMSQ/E and any keyboard drivers needed to run the system. Do not re-size any BASIC windows. Use the following descriptions as a guide and play with the screen resolutions from there. You will find this is usually safe on all systems. In a later article we will discuss how to resize and recolour the QL's BASIC windows. For now we will content ourselves with size alone.

It is, on the whole better to try to find a screen resolution which suits you for all programs that you are running and then stick to that. The best thing about QL systems is that no changes are permanent and, unless you write new values into a BOOT file or configure SMSQ/E, a reboot will always bring you back to where you were.

New, Clear Resolutions

Most variants of SMSQ/E will allow the user to choose the resolution at which the system runs. What do we mean by resolution? We are referring here to the number of pixels in the X and Y axis - that is horizontally and vertically. The standard QL resolution that we are all familiar with is 512 x 256 - this being 512 pixels horizontally by 256 pixels vertically. This arrangement acts like a kind of grid and will allow the programmer to position something exactly on the screen. Each item displayed on the screen is drawn in pixels and the programs plot these pixels to the screen. This is where newcomers to the higher resolution screens run into their first problems.

Each character of text is made of a set of pixels and, when the QL was first designed, all computers did exactly what QDOS did which was to plot out the characters in a fixed size or a series of fixed sizes. It did this by drawing each individu-

al character at fixed height and width. QDOS does allow some leeway by use of the CSIZE command the parameters of which are expressed as two numbers (i.e. CSIZE 1,1 will give a character 8 pixels wide by 20 pixels tall. The standard character is CSIZE 0,0 which is 6 pixels wide by 10 pixels tall).

You may wonder why the PC does not have this problem to the same degree. This is partly because of the way that the PC writes to the screen. If you increase the resolution on the PC the text does get smaller but you can change one variable to make all screen text change size. This is because most of the programs on a PC use vector graphics. Most modern systems have an optimum resolution of 800 x 600. The QL will write direct to the screen and, because there is no direct integration of all the programs there is no direct command to change the fonts or the font sizes. Worse still most QL programs have no way of changing their font or font size. This was not an option when most of them were written.

Many of the applications on the PC will look little different at 800 x 600 then they do at 1600 x 1280 (so long as the user changes the overall screen font size) for this reason. ProWesS was the only program to attempt to create such a system for the QL.

This is all very well at the standard resolution but what happens when you increase it. The answer is that the characters still use the same number of pixels but those pixels have now become much smaller and so the character displayed is also smaller.

Some of our older users find the text size at the higher resolutions is too small to read. The flip-side of the coin is that, at higher resolutions, you can get a lot more data on the screen. At 1024 x 512, for instance, you can open two QDs side by side. It is, therefore, important to decide which factors most affect your decision, hence the need to find a resolution which suits the individual. This, then is the first task facing us.

Most of the systems on which SMSQ/E runs, with the exception of the standard QL itself, will allow the user to change the resolution. Most will also accommodate the standard 512 x 256 QL screen as the lowest level and give the option of increasing that resolution either by command line/software at run time or by configuring the actual SMSQ/E file itself.

The resolutions available will depend on the hardware on which the system is being run. It is worth noting at this point that the standard QL display has a ratio of 2:1 (the 256 Y resolution is exactly

half the 512 X resolution) but the standard PC screen is 4:3. This means that circles on a QL style screen will become ellipses on a PC style display and squares will become rectangles. Some hardware will also not allow some resolutions to be displayed. There is a list at the end of the Aurora manual of the resolutions of which it is capable and some of these need changes to the jumpers on the board. Bear this in mind when reading the following explanations of how to change the screen resolution. If you give SMSQ/E a screen size in which it is not capable of producing a display it will revert to the nearest lower one. This may not be true of the monitor you are using. Some monitors, especially older ones and the modern TFT screens have a fixed range of signals and you may well find yourself locked out. It is best, then, to adopt the command line option first because a reboot will bring you back to the original display.

Command Line

The instructions in this section do not apply to the Q40/Q60 which has only four available screen resolutions. This will be dealt with separately.

The simplest way of changing the resolution is to go to the command line and type:

```
DISP_SIZE xx,xx
```

where 'xx,xx' represents the number of pixels in the X and Y axis respectively. A command like

```
DISP_SIZE 640,480
```

will produce a screen 640 pixels horizontally and 480 pixels vertically - or at least it will if the hardware supports it.

If you have started off with a display size in standard QL Mode (512 x 256) then you will now have a rectangular three colour QL display in the top right hand corner with a black surround to the right and directly below it. The black area is not 'dead space'. programs can use this area for display but it is outside of the 512 x 256 'QL Rectangle'

I will divert now to explain why we have that rectangle. When the extended resolutions were first designed there were a lot of programs that would only write to that area of the screen. It was, therefore important that it was there for these programs. That situation changed a lot as more and more programs became able to 'write outside the box' but that 512 x 256 display is built into SMSQ/E. You can change this yourself and you can change the character size too but we will come to that when we move into the BOOT file section.

If your system/monitor will support 800 x 600 (PC) or 1024 x 512 (QL) try entering those values and see how the display looks.

If you already have a BOOT file you can enter this line into it at the start:

```
DISP_SIZE 1024,512
```

and now your system will start in that resolution all the time.

One caveat here. If your system starts in that resolution already (you may have a system set up by someone else for instance) and they have reset the sizes of channels #0, #1 and #2, typing DISP_SIZE and inserting lower values to the ones you are currently using will reduce the screen size. If the QL windows have been increased to accommodate the new bigger screen you will find that you lose the command line window at the bottom of the QL display because the display always has its point of origin at 0,0 or the first pixel in each axis which is at the top left hand corner. This will mean you can no longer get to it to type a command to revert to the previous size. This will be discussed in the BOOT file section of this series but the best rule of thumb here is to test the resolution using the DISP_SIZE command before committing yourself to a configuration in either the BOOT file or by configuring the actual SMSQ/E. If you find yourself in the position of being unable to access the command line hit the CONTROL/ALT/SHIFT/TAB keys together and SMSQ/E will do a soft reset.

Configure the File

Once you have a display size you are happy with the next task is to make this permanent. You can do this in two ways. You can either add a line to the start of the BOOT file to set a resolution or you can use Menuconfig to configure the actual SMSQ/E file to run in that resolution. We will tackle the BOOT file option later but, for now we will look at the Menuconfig method.

Menuconfig requires a few things in order to operate. To start with you will need the Pointer Environment loaded but, since we are configuring a copy of the high colour version of SMSQ/E that should be no problem. You will also need the menu extensions (menu_rext). Both of these programs are commercial and copyright to Jochen Merz Software. There is a free to use CONFIG program but this will not configure level 2 config blocks. There is no reason why a free version of Menuconfig should not be written and distributed but no-one has done this. Menuconfig and menu_rext are always supplied with the programs that need them.

IMPORTANT NOTE: NEVER CONFIGURE AN ORIGINAL FILE. KEEP THE MASTER COPY SAFE AND ALWAYS CONFIGURE A BACKUP COPY

I will detour here to explain level 2 configuration to you. Some time ago it was realised that having to reconfigure each program all over again every time an upgrade came out was very tedious. Level 2 configuration relies on each program having a defined 'Config ID', a list of which was kept by Jochen Merz and is now maintained by Wolfgang Lernerz. Anyone wanting to use level two config blocks should first get an ID from Wolfgang. The purpose of this is simple. When you finish configuring a program menuconfig will offer to save the setting to a special file. This file lists the settings of that program under its unique ID. The next time you need to configure a version of that program you can just tell it to learn the settings from that last time.

Ok so back to the action. First you will need to load the Menu Extensions. You can do this by typing:

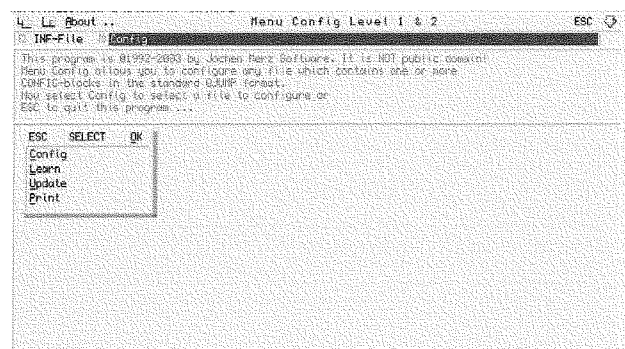
```
IRESPR DEV_MENU_REXT
```

(DEV is the device which holds the menu_rext file, e.g. flp1)

Then you fire up Menuconfig with

```
EX DEV_MENUCONFIG
```

(some of these files may have an extra extension such as _ENG or _ENGLISH etc. You can safely delete these extensions because they are just there to identify the program language.)



When Menuconfig appears you will be presented with the window as shown above. I am not going to launch into a big explanation of Menuconfig here. I hope it will be provided in a later article because it is an important tool in the setting up of many programs. The pressing part of this is how to use it to configure SMSQ/E to give a specific resolution. You will see, in the top section a window with two commands in it.

F2 INF-File

This will give you access to the INF files which store the information on the programs you have configured already. More on this in the full Menuconfig description.

F3 Config

This is the one which concerns us here. Click on the space just in front of the F3 or press the F3 key on the keyboard and you will be presented with a file menu. Locate the file you wish to configure and right click it. This will then give you a menu with four options.

Config: This is the one we will use here if we have not previously configured the file or if we want to make a change to the file.

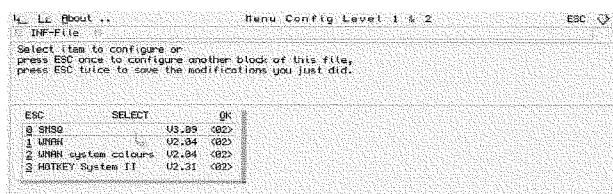
Learn: This will read the config block settings into the INF-File. Useful if you want to keep the current settings safe whilst experimenting or to make sure that INF-File is up to date with your current settings before upgrading.

Update: This will write the values in the INF-File to the program being configured. (Nothing is changed if there are no values in the INF-File for this program)

Print: This will print the contents of the config block in text format so you can look over them or keep them as a backup.

OK we are ready configure so click on Config.

The Real Work Begins



Most versions of SMSQ/E have four configurable sections each of which contains its own list of items. QPC2 is a special case and will be dealt with separately.

SMSQ: This is the one we are concerned with here since it contains the display settings but I will list the others for completeness. It also contains the Default Keyboard and Message Language, Stuffer Buffer Key Selection, and items specific to that version of SMSQ/E. In the SuperGold Card version this will be whether to use the QIMI mouse and keyboard type. In the QXL version it will have options to set boot device, Processor clock speed etc.

WMAN: This contains the options on how the

window should be moved, by Sprite, Whole Window, or Outline. The Standard Pointer Environment method is to use the Sprite.

WMAN system colours: These will set the actual colours the Pointer Environment will use for its system menus and windows.

HOTKEY System II: This will change the way that the Stuffer Buffer is set up.

The QXL version of SMSQ/E has an extra option:

HOSTS: This will allow you to change the name and location of the 8 drives seen as WIN1_ to WIN8_ more on this later.

So, go to the SMSQ section and click on that. The two options that concern us here are :

Initial Colour Depth: This can be set to QL, which will give us a standard four colour display in mode 4 or High Colour 16 bit which will use the High Colour drivers. The Aurora version has only QL and 256 colour options because the Aurora is a special case. It should be noted that selecting 256 colours on the Aurora will limit the resolution of the display.

Display size: This will choose the actual resolution for the screen. Each version of SMSQ/E has its own list of settings which are dependant on the hardware. The Atari version has two settings one for X res and the other for Y res and these can be set directly by entering the values for the Atari system you are using.

The QXL offers 512x256, 640x480, 800x600 and 1024x768-

Q40/Q60 offers QL Mode 4, Large 16 Bit and Small 16 Bit.

The SuperGoldCard offers 512x256, 512x320, 512x384, 512x480, 640x320, 640x480, 768x384, 768x480, 768x576, 1024x480, 1024x512, 1024x768.

Once this is set press OK to return to the previous menu and OK again to exit the configuration. If you have made changes you will be prompted to save the file either by overwriting the original or with a new file name. When you exit Menuconfig you will be asked if you want to write these values to the INF-File. If you are not sure about the changes you have made do not write the values at this point. Test the file and, if it is to your liking go back to Menuconfig and use the 'Learn' Option.

QPC2 Configuration

QPC2 is a different case because you can configure the resolution of QPC2 in two different ways. You are offered a configuration menu when

you first start QPC2. This will allow you to set almost all of its options in one, easy to use, way. You can also use Menuconfig as described in the previous section. In the end you are better off using a combination of both. One thing which daunts many of the new users and puts them off using Menuconfig is that the program is not actually on the SMSQ/E section of the hard drive. QPC2 can, however, access the Windows side of the hard drive and this makes it very easy to configure in the normal way.

The QPC2 settings are dictated by the graphics card and settings of the version of Windows it is run under and these are what will appear in the config box at the start. You cannot set any options that are not allowed by the hardware. Much of the setup of QPC2 is covered in an article in QL Today (vol7, iss3, p28). I will, however, go ahead and cover the resolution settings again.

Using the QPC2 Configuration Box

When you fire up QPC2_ you are first presented with a Configuration Box. This will give you access to many of QPC2's settings but the ones which concern us here are those related to its resolution and colour depth. On the left hand side you this box you will see a Box labelled Display Modes. Beneath this is a list of the resolutions supported by the device you are running it on.

To the right of this window you will see a Box to select the Colour Depth where you have a choice of High Colour / 8 bit / QL Colours. Below this is the Resolution window. It will display its choice of resolution for you and to change this all you have to do is to select one of the available resolutions from the box to its left. If you have selected QL Colours in the Colour Depth box you will also see an option to use 512 x 256 the standard QL resolution. It is not easy mapping a 2:1 resolution ratio to a 4:3 one making it look good and still filling the screen but Marcel has done a very good job of this.

Below this is a check box which will allow you to select whether you want QPC2 to run full screen (unchecked) or in a Window (checked). Both options have their advantages and uses but if you run at full resolution in Window mode the text displayed will be much smaller, depending on the size of the window you are running in.

There are many more settings available from this Window but these are covered by the article mentioned above.

Don't forget that, if you want the program start up in the same way next time. You will have to hit the save button before continuing.

Using Menuconfig

When you install your first copy of QPC2 onto the hard drive it will, if you use the install program, create a subdirectory called C:\QPC\. The install will then copy the appropriate files to this subdirectory. If you have made no changes to the configuration of QPC2 you will, therefore, find the SMSQE.BIN file in DOS1_QPC_ when you look from the SMSQ/E emulation.

Start Menuconfig as described in the previous section and go to the DOS1_QPC_ directory and select the SMSQE.BIN file. you can then follow the same procedure as before for the configuration of the file.

Select the display option. The settings here are exactly the same as those for the Configuration Box so you do not have any extra options to pursue. The configuration of the screen size is by typing in a value and you have more of a chance of error here than with the method described above. There are some settings whose access is only available from Menuconfig but none of these are related to screen resolution and so beyond the scope of this article.

Even if you have not used Menuconfig to configure QPC2 it is worth loading the SMSQ.BIN file into the program so you can use the 'Learn' option and write the values to the INF-File. This will save you a lot of time when you come to update QPC2 because all you have to do is to load the new version into Menuconfig and hit 'Update'.

Q40/Q60 Screen Resolutions

The Q40/Q60 version of SMSQ/E handles the screen resolution issue differently. The reason for this is hardware related and I will not go into all that now. You have four screen resolutions and colour depths available and the screen sizes are tied directly to the colour depths. These are :

Mode 8 (256x256)

Mode 4 (512x256)

Mode 33 Small (512x256) (High Colour)

Mode 33 Large (1024x512)(High Colour)

Switching from 1024x512 to 512 x 256 (any colour depth) will only work if you use a command line before issuing the DISP_SIZE Command i.e.

WTV 0,0

or

WMON 0,0

Failure to do this will cause a crash. Even if you do issue these commands you will have pro-

blems if any part of window 0 is out of the 512 x 256 display area. In actual fact any attempt to reduce the display size at runtime will cause problems to any programs running at the time and while some will still run they may be rather erratic.

If you are starting from 512 x 256 you can issue some display commands.

DISP_SIZE 1024 will switch the display to mode 33 (high Colour) and 1024 x 512

DISP_COLOUR 3,1024 will have a similar effect.

You can use Menuconfig to change the startup resolution of SMSQ/E on the Q40 / Q60 as described above.

Conclusion

There you have it. You should now know how to set up any SMSQ/E screen. One last thing to mention is how to find out how large your screen is. If you want to know what resolution you are running under go to the command line and type:

PRINT SCR_XLIM

for the horizontal resolution

and

PRINT SCR_YLIM

for the vertical one.

You may find some of these articles in previous magazines helpful;

subject	Volume	Issue	Page
Config level 2	1	3	37
Qmenu pt 1	1	3	24
Qmenu pt2	1	4	18
QMenu pt 3	1	5	40
Abbreviations	2	6	27
ZIP & UNZIP	4	1	42
FileInfo 2	4	5	48
ACP	4	6	20
SERNET	5	1	31
SERNET	5	3	26
Adding to the QL	5	5	18
QPAC2	5	6	46
QPAC2 pt2	6	1	15
QPAC2 Pt3	6	2	32
QTyp	6	2	26
Extensions	6	3	12
Serial Cables	7	1	28
QPC2	7	3	28
CD ROMon Q40	9	1	31
Network	9	4	24

Next Issue - Part II 'Drive Time'

QDT v1.00

A Preliminary Review

Per Witte

Introduction

QDT is a Desktop program for SMSQE systems with extended colour capabilities. It will run on high-end platforms such as Aurora with Super Gold Card, Q40, Q60 or QPC2. QDT is visually pleasing and displays a high degree of programming proficiency in every detail. This is perhaps not surprising when you consider that Jim Hunkins, QDT's author, is a professional programmer doing a stint of busman's holiday. He has degrees in Electrical Engineering and Computer Science and has worked as a programmer in a wide variety of contexts for a number of years, most recently for Apple Corp. He is a long-term Sinclair aficionado, though I'm happy to be able to reassure readers that there are NO indications that Jim has ever owned a C5. Any confusion must have arisen over his (and he admits this quite openly) ownership of an Apple dual G5 tower, host to most of his work on QDT(!) Living in sunny California as he does, it may seem unfair to some of us that Jim isn't doing QDT for free,

but we understand that there are costs involved in living, even in sunny California, so I guess we have to make allowances. (In fact I think it a healthy sign that commercial software is being produced as it helps to oil the machinery of software production and energise the market.)

Last month QL Today readers were treated to a demo of QDT and an accompanying article by Jim. Despite Roy Wood's assertion in the same issue that he was not writing a review, between him and Jim we did get a review of sorts. Considering that, and the plentiful detailed information available on the QDT website at

www.jdh-stech.com

I am bound to take a different approach to avoid simply re-hashing it all again here.

Why QDT?

In essence, we already have a "desktop" on all SMSQE systems, ie many different programs piled on top of each other on the background screen. Many of us will have improved desktops for point and click access to programs and files, such as Qpac2 or Cueshell and/or Cascade, so what else could we possibly want in that connection? After pondering this question for a while I came up with the following answer:

QDT provides, or will provide, three things that can improve productivity:

- 1) The Desktop paradigm.
- 2) Drag and drop [DnD]
- 3) Raising standards

1) Regarding the first, it is a matter of taste whether you find this idea helps or hinders you. However, WIMPS, GUIs or Desktops are here to stay, as even dyed-in-the-wool Unix fans now appear to have acknowledged – and for a good reason: If properly implemented they can simplify many operations and ease the drudgery of common housekeeping tasks considerably.

2) DnD seems so simple: You point to an icon, engage it, and move it to some location, eg into a folder or on top of another icon, where you disengage – or drop – it. In the first case you would expect to transfer or copy the icon (and perhaps whatever object the icon represented) to the location specified, while in the second you might expect the program attached to the icon pointed to, to pop up and load the data file you just dropped on it. However, this is only genuinely useful if it is done in a way that allows external programs to take advantage of it too.

3) One reason why it is difficult to produce some of the software people using other systems seem to take for granted, is that there is no typical QL system; every "QL environment" is unique. A number of system capabilities depend on various external bits and bobs being installed from a variety of different sources. Also, important file locations are not fixed; and different platforms impose various restrictions which cannot easily be circumvented. QDT bumps up the minimum standard by including extra free and commercial utilities, by cleverly trying to cope with the awkwardness of dispersed files with its Installer program, and by rejecting systems below a certain minimal capability, such as lack of colour, or limited memory or grunt.

If QDT is successful in convincing a significant proportion of Qlers that this is the Front-end they have been waiting for, it could generate a "virtuous cycle" of new users attracting programmers to write programs for the QDT environment, which again will encourage more users to upgrade.

A brief peek under the bonnet

There is the difficulty that many of the system facilities required to produce the illusion of a

desktop are not presently available in the SMSQE Pointer Environment [PE]. The SMSQE Background is not a "real" desktop, there is no native facility for icons (as we now know them), and objects are not second nature to the system as you will find in many contemporary Operating Systems [OS]. Jim has had to use his imagination to find alternative ways to bend PE to his will. He succeeds surprisingly well in this. The icons all line neatly up in their designated positions on the desktop (Background) and it is possible to move them, interactively, from one place to another. If this doesn't happen on your system you may need to set WM_MOVEMODE to 2. And that gives the game away: The icons are simply program windows, and each icon is its own job (similar to the Qpac2 buttons).

But windows must be square - yet icons can be any shape, and therefore the illusion sometimes shatters as, in certain circumstances, you may see the debris of other job's windows surrounding the icons. There is not much Jim can do about this from within QDT. If he sticks with this design decision, it will probably require some change and addition to SMSQE. This may come in due course, or Jim may find other ways of setting about it. Icons already inside a QDT folder have a much easier time of it, as moving them about is just a matter of using the icon to be dragged as the pointer sprite inside the folder window. But this is how far it goes at the present time; you cannot drag and drop objects between QDT folders and the Desktop, nor between folders.

According to Jim's QDT roadmap, the first item on the list of To Dos is the implementation of DnD. If I understand correctly, this will at first only be implemented locally, ie within the QDT system, but the hope and intention is to make this universally available.

What works and what doesn't?

At present QDT v1.00 very basic. It is merely a sketch of what it can or will be if it gets the necessary support from the QLing public and, of course, its author. An internal implementation of DnD would be a first, essential step. A second is a QDT file manager: Now every object has to be individually created on the Desktop or in the folders, which involves a lot of work to set up, and considerable effort to maintain.

The icon drawing utility Icon Draw is a great little tool. Although there are still a few snags, it looks great and works well, and one can perhaps assume that it represents the standard of future programs and utilities from the same stable. The tabbed notebooks still require some work, but it

is satisfying to see the hoops Jim makes PE go through to get the effects he wants to achieve. For example tabs are not native to PE, but Jim manages to achieve a good likeness in look and functionality. However, at the moment these tools don't really add anything, as if you didn't have QDT in the first place you wouldn't need them. They will come into their own when QDT becomes a fully fledged Desktop.

I'm not happy with some of the design and implementation decisions Jim has made, but some of my reservations are merely my own and won't bother others. For example QDT leaves its log file open as long as QDT is up and running. This means you cannot change the partition on that device or, for instance, run a second copy of QPC2 off the same disk set while QDT is running. Another issue I have is with the large number of same-named jobs QDT creates; two for each object on the desktop. With a dozen or so objects they soon crowd out the other job information and also soak up quite some resources, including processing power and memory (each icon job presently takes up at least 30k), not to mention the time it takes to load all of this at boot up. However, these criticisms are provisional, as I believe that much may still change, and improvements can be expected as the product develops and matures.

What else is included?

I would like to say something about the software environment that QDT expects. SMSQE, the all-inclusive multi-platform successor to Qdos, mainly designed and written by Tony Tebby, author of Qdos, already comes with most of what is required, but if you were starting from Qdos (which QDT won't run under because it requires high colour) you would need:

Pointer Interface [PI]	} Pointer Environ- ment [PE]	} SMSQ/E
Window Manager [Wman]		
Hotkey system [HK]		
Colour drivers (an optional extra on Aurora)		

PI above gives you non-destructible windows, mouse pointer, sprites and basic windowing functions. **Wman** does all the menu stuff, including window buttons and menu scroll bars. All this now comes in glorious colour on the hardware mentioned, thanks especially to Marcel Kilgus. The **hotkey system** manages shortcut keys but more importantly adds Things. Things are just bits of data, code or jobs in memory that can quickly, easily and safely be accessed, shared and removed.

The remaining software comes supplied with the QDT package:

Qmenu is a residually loaded toolkit by Jochen Merz, consisting of small, runtime-configurable dialogue boxes that present a common interface built on PE. They can be used by anyone's programs in various circumstances (eg to locate a file to load, or to display a warning message), dispensing with the need for everyone to invent their own version of the wheel, and get on with more productive work. It is included with QDT as it, and many other PE programs, use it.

Env_bin is a toolkit created by David Nash and others to implement the concept of environmental variables as found on many other operating systems. This can be used for setting default values for individual programs or system-wide values such as default paths. Many programs, PE or not, require environmental variables, particularly if they derive from the Unix world.

FileInfo2 [FI2] or equivalent is a must-have on any modern system. The concept is widely used and integrated on other OSes, but on the QL it can be obtained as a freeware toolkit. It was originally started by Wolfgang Lernerz, but later considerably extended by Thierry Godefroy. If you think about it, you don't really work with programs at all, but with data. FI2 links a data file, such as a Quill document or a Sbasic source file, to one or more associated programs, so that by "EXECuting" the data file you actually start its associated program with the data file loaded, ready to be viewed or altered. Many programs, such as file managers (Qpac2 and Cueshell) are FI2-aware, so just clicking on the name of the data file you want to work with loads the program with the data file in it. Very simple, no mistyping or remembering of difficult names, just point and click. FI2 is fiddly to configure, although it has a reasonably good program to do so with. It's well worth the effort, and really, if you are going to use QDT it is a must.

Who is QDT for?

We have among our ranks those who even today, after up to a respectable 21 years, still claim to be beginners. Some of those may find the idea of a WIMPS attractive. But despite its simplicity in use, setting up an environment capable of running QDT would be no mean feat if you currently had anything less than an Aurora and were not familiar with PE. It may be an idea to package a complete system upgrade for new users and eternal beginners, to include simple documentation and all the software that is needed for a modern, WIMPS-driven front-end.

Kaiser-Wilh.-Str. 302 D-47169 Duisburg
Tel. 0203 502011 Fax 0203 502012
<http://smsq.j-m-s.com> smsq@j-m-s.com

QDT - The QL DeskTop

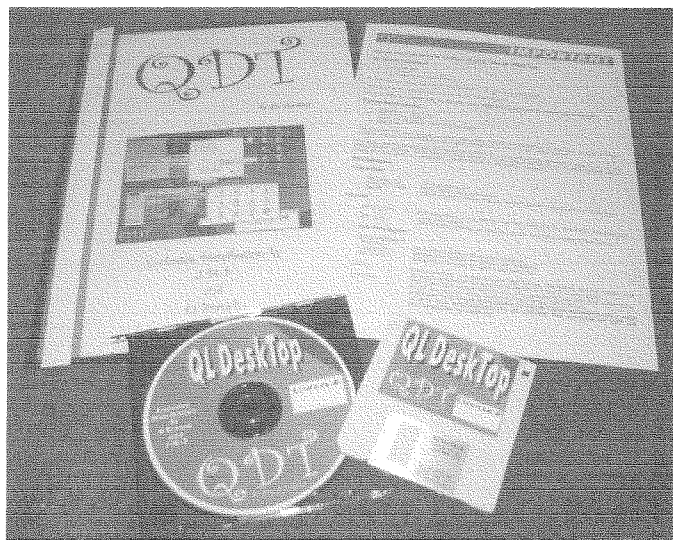
Yes, it has taken a long time to develop. But it is extremely well tested and has grown much over the initial specification. And now it is available! QDT brings the QL a modern, easy- and intuitive-to-use graphical user interface. Arrange your

programs, games, applications in folders, start your favourite applications with a single click. Fully configurable!

QDT comes with an automatic installer – like a guided tour. Installing it is as easy as never before!

System requirements:

- SMSQ/E Version 3.06 or later
- Graphics Driver 2 Colour support
- HARD DRIVE with at least 3MB free space.
- 4MB RAM



You would like to test QDT first? No problem, download a Demo version first.

What do you get when you order QDT?

- A floppy disk for Aurora or quick-start users.
- A CD ROM containing a QXL.win file, backgrounds, sprite editor, and many other extras.
- A printed quick-start guide
- 12 months free updates!

QDT costs EUR 43,- or £30 plus postage (EUR 3,-/£2 for Europe, EUR 6,-/£4 for the rest of the world).

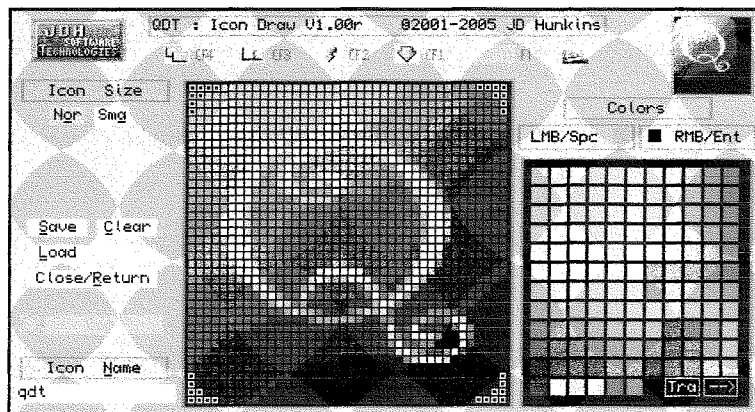
**If you wish to place your order via internet, please do not send your credit card details in an email!
Please use the secure contact form on the SMSQ homepage: SMSQ.J-M-S.COM**

We accept VISA, MasterCard and Diners Club!

One school of professional users aren't into SMSQE at all, so won't currently be able to use QDT. Another group will always shun GUIs and other things that distract them from their verbal intimacy with their computers, so they will be a lost cause too, unless they break a few fingers or get their heads around the fact they don't have to give up one for the other. Another school will have gone the way of PE and Qpac2, probably with the system extensions mentioned earlier, and possibly with Jonathan Hudson's Cascade and other PE tools that constitute a sort of patchwork front-end. QDT would have to provide some exceptional capability to beat that combination, to my mind. A real drag and drop facility plus a file manager that can work with directories would be quite convincing.

The future

QDT is here now. Jim has set down a marker for the quality of top QL software, in particular with regard to Look and Feel. He also pushes the envelope of the possible with the current stan-



The Icon draw program

dard of his software. For now though, you will have to buy QDT on the strength of future developments, as it very much on the way, but not quite yet there. A chunk of that development is included in the price: "Through 2005 all updates will be at no charge as additional functionality is added. The final functionality level at the end of 2005 will depend on the actual interest in QDT. Note that there is a minimum set that is expected to be implemented, regardless," according to Jim. By buying into QDT now, you will be voting with your money – the only true democracy there is! Would I buy QDT? Yes I would.

Another Addition to GWASS

George Gwilt

The Problem

During the QL2004 show at Eindhoven recently I was persuaded by both Wolfgang Lernerz and Marcel Kilgus that it would be easier to use GWASS to assemble SMSQE source if it could understand Qmac macro parameters.

I should explain that SMSQE is written to be assembled by Qmac. Although most of the code is acceptable to both assemblers, there are differences. The main difference is in the syntax of macros. The differences here are so great that almost all the Qmac macros have to be re-written for GWASS. As it happens I have already produced a new set of all these macros so that, provided the source code can be written to include whichever set of macros is required, only the coding of the parameters themselves need be altered. If GWASS could do this alteration during assembly the changes to SMSQE source for Qmac/GWASS assembly would become dramatically less.

The Solution in Outline

When dealing with a macro call, GWASS reads and stores the parameters before bringing in each line of the macro for assembling. The parameters are read in from a line of characters in a buffer to which register A1 points. I realised that if needed GWASS could be made at this stage in the program to translate Qmac style parameters to GWASS style and alter A1 to point to the new set.

Of course, this meant that GWASS would have to have another option, whether to use GWASS parameters or Qmac ones. Luckily the entire system of options has already been set up so that adding the new option was easy. You can now alter the Qmac/GWASS switch permanently by using Menuconfig. This can be overridden at run time by setting an appropriate value in the parameter list when GWASS is EXecuted. Finally, and most importantly, the switch can be set by a command in the source code. QM_PAR will set the switch to use Qmac parameters. GW_PAR will set it to use GWASS parameters.

This last facility is crucial. Suppose that the switch is set to 'Qmac' to assemble SMSQE. Suppose further that one of the GWASS style macros itself calls another macro. If this other

macro had parameters they would undoubtedly be in GWASS style. So disaster would occur unless GWASS were told to use Qmac parameters for the SMSQE source code and GWASS style parameters for the internally called macros. Well, this is easy. Each of the GWASS macros which calls another GWASS macro using parameters now includes initial code which detects the current value of the switch and alters it to 'GWASS' if needed. Final code in the macro ensures that the initial value of the switch is restored.

The Translation Subroutine

The operation of translating is essentially simple.

- A. Set a pointer to a buffer
- B. Read a character from A1
- C. Translate this to the altered character
- D. Put the new character to the buffer
- E. Go back to B.

The process ends when there are no more relevant characters.

Step C is, of course, of most interest. This depends on the differences between Qmac and GWASS. The parameters are presented in broadly similar manner. They appear on the line of the macro call and possibly on continuation lines. The end of a parameter is signalled in the case of Qmac by one of SPACE or TAB (which constitutes "white space"), comma, LF or back slash (\). However, GWASS does not recognise back slash. Moreover white space signals the end of all parameters whereas in Qmac there may be more to come. This means that the characters SPACE, TAB, LF, back slash and comma all require special treatment.

Continuation lines are signalled in Qmac by the back slash. GWASS requires a comma followed by white space or LF and an ampersand starting the continuation line.

Sometimes a parameter is required to contain SPACE. In order for the SPACE not to be taken as the termination of the parameter the parameter is enclosed in curly brackets, " { } ", for Qmac. GWASS uses "<" ">" ("less than" and "greater than") for the same purpose.

To avoid confusion with a parameter, Qmac will accept a comment on a line of parameters provided it starts with semi-colon.

All the characters mentioned so far have to be identified for step C. In addition there is the problem of "value of". Qmac signals "value of" by square brackets. Thus, if "var" has been equated to, say, 4, then Qmac replaces [var] by 4. The same feature exists in GWASS but the expres-

sion to be evaluated lies between a vertical line and a tilde "~" and " ". The inclusion of either of these in a parameter will cause trouble when GWASS prepares for assembly a macro line containing that parameter. To avoid this trouble GWASS requires substitute characters for vertical line and tilde to be put in any macro parameter. Thus vertical line, tilde, alpha and delta must all be added to the list of characters needing special treatment.

Finite State Machine

We have now 13 different characters which may need special treatment at step C. The treatment also depends on where we are in the process. For example a comma signals the end of a parameter if we are reading characters inside a parameter. But if the parameter has been enclosed inside curly brackets the comma must be treated as just another character in the parameter.

I had recently read Hugh Rooms' recent article in QL Today Vol 9 Issue 2 where he describes his version of a program to disassemble a register list. He uses the technique of Finite State Automaton (or Machine). It seemed obvious to me that the Qmac parameter translation problem would lend itself very well to this method. At any rate the description of step C would be most easily set out using the method.

The number of different states is 6

0. Looking for a parameter, comma not needed
1. Inside a parameter not starting with either { or {
2. Inside curly brackets
3. Just after a right curly bracket
4. Looking for a parameter, no comma yet
5. Inside a parameter starting with {

For each state the treatment of each of the 13 special characters and of all the remaining characters has to be defined.

This can be set out in two tables, the first giving the change of state for any character and the second giving the translated characters to be sent to the GWASS buffer.

I have not described in detail all the changes that must be made to Qmac parameters to make them acceptable to GWASS. However, the tables below, assuming that they are complete, should allow an investigator to find what happens in any particular case. For example Qmac signals a null parameter by two commas with nothing but white space between them. What does GWASS do? Let us assume that a comma appears just after a parameter. The tables show that we go to

state 0 and copy the comma, preceded by "greater than" if we had been in state 5 and not the more common state 1. We can see that white space is ignored in state 0; it has no effect. So

when another comma appears, since we are still in state 0, the characters "less than", "greater than" and a comma are all issued. So we see that GWASS needs positive proof of a null parameter.

State Alteration Table

Character	Current State					
	0	1	2	3	4	5
TAB/SPACE	.	4	.	4	.	4
LF	END	END	ERR	END	END	END
,	.	0	.	0	0	0
;	END	.	.	END	END	.
<	5	.	.	ERR	5	.
>	1	.	.	ERR	1	.
\	END	END	.	END	END	END
{	2	.	.	ERR	2	.
}	1	.	3	.	1	.
& ~	1	.	.	ERR	1	.
Others	1	.	.	ERR	1	.

In addition to the continuing states 0 to 5 there are two forms of exit. END is a valid exit but ERR indicates an exit with an error.

Action Table

Character	Current State					
	0	1	2	3	4	5
TAB/SPACE	.	.	CPY	.	.	>
LF	>
,	<>	CPY	CPY	CPY	CPY	>,>
;	.	CPY	CPY	.	.	CPY
<	<<	CPY	CPY	.	,<<	CPY
>	CPY	CPY	>>	.	,>	>>
\	.	.	CPY	,	,	>,>
{	<	CPY	CPY	.	,<	CPY
}	CPY	CPY	>	.	,}	CPY
& ~	SCPY	SCPY	SCPY	.	,SCPY	SCPY
Others	CPY	CPY	CPY	.	,CPY	CPY

In the above table CPY means "copy" and SCPY means "copy the substitute character currently standing for | or ~". Also a full stop (or period) indicates "no action".

I have not included in these tables entries for the characters substituting for | and ~. The presence of either of these in a parameter leads to an error in all cases.

Implementation

I decided to start coding this Machine by assigning a program number to each of the 256 character values. Then for each of the states I thought I would set up a table of word relative pointers to code. There would one pointer for each program number. The first word in each table would point to the code for program number 0, the second for program number 1 and so on. The program numbers would range from 0 to 13.

The program numbers would be translated to actual code by means of a set of State Tables,

each containing word relative pointers to the required code.

If I had carried out that plan there would have been six Stat Tables. This seemed to me to be slightly ungainly and not very interesting. Instead I decided to limit the State Tables to four, for states 0 to 3, and to indicate the remaining states by a value in D6. Thus the new state 0 is the old state 0 with D6 = 0. State 4 becomes equal to state 0 with D6 = 2. Similarly state 2 with D6 = 1 replaces state 5. The old state 2 becomes the new state 2 with zero D6.

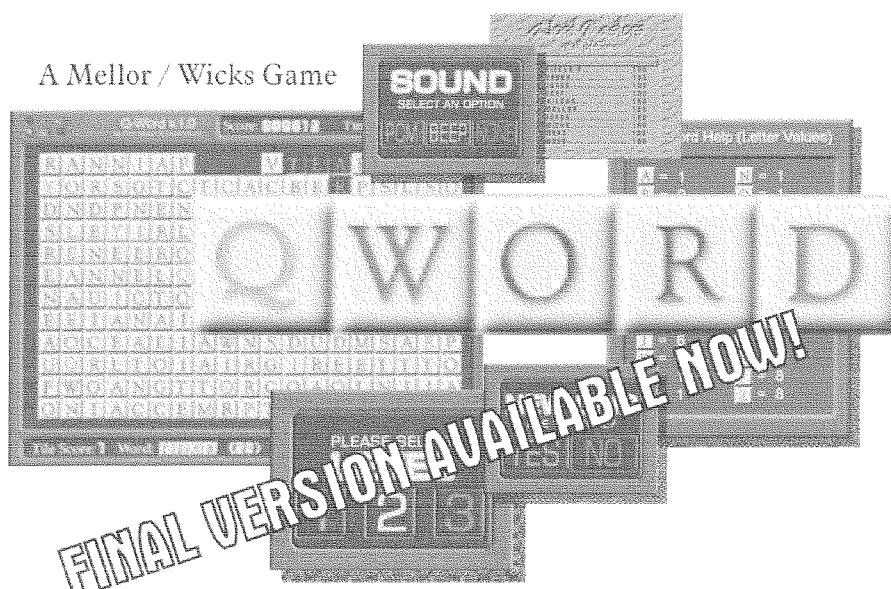
RWAP Software

in cooperation with

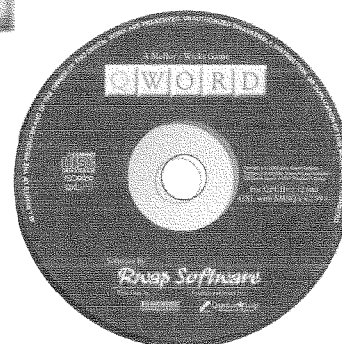


PRESENTS...

A Mellor / Wicks Game



for the Q40 & Q60
(And Aurora/SGC)



for QPCII & QXL

Scrabble® Meets Tetris® Meets the clock!
In glorious colour and digital sound! (Aurora too!)

For the first time in the history of the QL, a truly high colour, digital sound *really* addictive game, in superb 1024 x 512 (640 x 480 on QPC/QXL and 512 x 480 on the Aurora) high colour. SMSQ/e is NOT needed for the Aurora (and soon for the Q40/60) but it is recommended. The innovations are too many to mention but let's just say that the surprises start with its user-friendly installer which is unlike any other!

The object of the game is to clear letter tiles from the grid by using them to form words. Points are based on the length of the word and the letters used. Simple, yet addictive!

PRICES

QWord for QPC2/QXL (CDROM)	£20
QWord for Q40/Q60 (HD FDD)	£20
QWord for Aurora/SGC (HD FDD)	£20
Add Pword (500,000 word UK dictionary)	£10

LANGUAGES

Each version is supplied with one dictionary (normally the OSPD UK English Scrabble Dictionary). However, QWord can also be configured to use the following alternative language grids:

Dutch, English (USA), English (UK), French & German

If you want more than one version of QWord (eg. to play on Aurora as well) - add £5 for each additional version.

Specialist QTyp dictionaries for each language have been produced, which removes the need for accented characters in the foreign languages.

You can choose ANY one dictionary to be supplied with QWord.

Alternatively, order additional dictionaries for £2 per dictionary.

SPECIAL OFFER! FOR AURORA/SGC USERS
BUY Q-WORD WITH Colour Drivers version of SMSQ/e for the SGC
!!ONLY £50!!

RWAP Services

26 Oak Road, Sheffield, Walsall, West Midlands WS4 1RQ, United Kingdom. Website: <http://www.rwapsoftware.co.uk>

We accept:



(PayPal/Nochex Payments to RWAPSoftware@aol.com. Cheques in £ sterling made payable to R. Mellor)

I set up a table of bytes at QTAB(A6) giving for each of the first 192 characters its program number. The program number for all characters with value greater than 191 is zero. By using the program number as an index into the current state table I was able to determine the address of the appropriate code to branch to.

Much as in Hugh Rooms' example I set the pointer to the current State Table in an address register. I use A5.

The central part of the Machine appears at PR31_NXT in the program shown below. The next character is put into D0.W and its program number put in D1.W taken either from QTAB(A6) or set to zero if the character value exceeds 191.

The word index to the required code is extracted from the State Table to which A5 points and is added to A5 to give the destination of the code itself.

Each bit of code will usually end with a branch back to PR31_NXT to deal with the next character.

The code below shows in detail how all this works.

The subroutine is entered either at PR31_Q1 or at PR31_Q. The first entry occurs when the first line of parameters has appeared. The second entry is made just before any continuation lines are processed.

```
; Translates Qmac style parameters from A1 to GWASS style in MAC_BUF
; If ampersand is needed for the next line, D6 is set to 1
; The 1st entry is at PR31_Q1, which clears D6
;
PR31_QR    REG        D1/D2/A2/A5
;
PR31_Q1    MOVEQ       #0,D6                mark 1st line of parameters
          BRA          PR31_Q2
PR31_Q     MOVEQ       #1,D6                mark subsequent lines (& needed)
PR31_Q2    MOVEM.L     PR31_QR,-(A7)
          LEA          PR31_ST0,A5          start at state 0
          LEA          MAC_BUF(A6),A2       set a suitable destination buffer
          MOVEQ        #0,D0                see that D0.W is 0
          TST.L        D6
          BEQ          PR31_NXT             no "&" needed
          MOVE.W       #"& ",(A2)+
          MOVEQ        #0,D6                clear "&" and "," marker
;
; This deals with the "next" character
; Bits 8 to 15 of D0 are zero
; A5 -> the current state (PR31_STx)
;
PR31_NXT   MOVE.B      (A1)+,D0             next character
          MOVEQ        #0,D1
          CMPI.B       #$BF,D0             The QTAB table is from 0 to $BF
          BHI          PR31_NXT1           Take the "number" as 0
          MOVE.B       QTAB(A6,D0.W),D1    Find the "number" from the table
          MOVE.W       (A5,D1.W*2),D1      Address of code relative to A5
PR31_NXT1  JMP         (A5,D1.W)           Go to the code
;
; QTAB(A6) is a table of numbers for certain characters.
; All are zero except:
;
; Char    ASCII    Number
;
; $        $        1
; TAB      9        2
; LF       A        1
; SPACE    20       3
; ,        2C       D
; ;        3B       C
; <        3C       7
; >        3E       6
; \        5C       4
; {        7B       8
; |        7C       5
; }        7D       9
; ~        7E
; "        AC       A Note " and ; stand . .
; '        AD       B . . for the current defaults
; All others      0
```

```

;
; State 0
; Looking for a parameter
;
PR31_ST0  DC.W      PR31_SA-PR31_ST0  Prog 0   other
          DC.W      PR31_SB-PR31_ST0  1       TAB/SPACE
          DC.W      PR31_SC-PR31_ST0  2       LF
          DC.W      PR31_SD-PR31_ST0  3       ,
          DC.W      PR31_SE-PR31_ST0  4       {
          DC.W      PR31_SA-PR31_ST0  5       }
          DC.W      PR31_SF-PR31_ST0  6       \
          DC.W      PR31_SA-PR31_ST0  7       >
          DC.W      PR31_SK-PR31_ST0  8       |
          DC.W      PR31_SL-PR31_ST0  9       ~
          DC.W      PR31_ER6-PR31_ST0 10      'u'
          DC.W      PR31_ER6-PR31_ST0 11      'i'
          DC.W      PR31_SS-PR31_ST0 12      <
          DC.W      PR31_SC-PR31_ST0 13      ;

```

```

;
; State 1
; Inside a parameter
;
PR31_ST1  DC.W      PR31_SH-PR31_ST1  Prog 0   other
          DC.W      PR31_SM-PR31_ST1  1       TAB/SPACE
          DC.W      PR31_ST-PR31_ST1  2       LF
          DC.W      PR31_SI-PR31_ST1  3       ,
          DC.W      PR31_SH-PR31_ST1  4       {
          DC.W      PR31_SH-PR31_ST1  5       }
          DC.W      PR31_SF-PR31_ST1  6       \
          DC.W      PR31_SJ-PR31_ST1  7       >
          DC.W      PR31_SQ-PR31_ST1  8       |
          DC.W      PR31_SR-PR31_ST1  9       ~
          DC.W      PR31_ER6-PR31_ST1 10      'u'
          DC.W      PR31_ER6-PR31_ST1 11      'i'
          DC.W      PR31_SH-PR31_ST1 12      <
          DC.W      PR31_SH-PR31_ST1 13      ;

```

```

;
; State 2
; Inside {}
;
PR31_ST2  DC.W      PR31_SH-PR31_ST2  Prog 0   other
          DC.W      PR31_SH-PR31_ST2  1       TAB/SPACE
          DC.W      PR31_ER5-PR31_ST2  2       LF
          DC.W      PR31_SH-PR31_ST2  3       ,
          DC.W      PR31_SH-PR31_ST2  4       {
          DC.W      PR31_SO-PR31_ST2  5       }
          DC.W      PR31_SH-PR31_ST2  6       \
          DC.W      PR31_SU-PR31_ST2  7       >
          DC.W      PR31_SQ-PR31_ST2  8       |
          DC.W      PR31_SR-PR31_ST2  9       ~
          DC.W      PR31_ER6-PR31_ST2 10      'u'
          DC.W      PR31_ER6-PR31_ST2 11      'i'
          DC.W      PR31_SH-PR31_ST2 12      <
          DC.W      PR31_SH-PR31_ST2 13      ;

```

```

;
; State 3
; After }
;
PR31_ST3  DC.W      PR31_ER7-PR31_ST3 Prog 0   other
          DC.W      PR31_SM1-PR31_ST3 1       TAB/SPACE
          DC.W      PR31_SC-PR31_ST3  2       LF
          DC.W      PR31_SI-PR31_ST3  3       ,
          DC.W      PR31_ER7-PR31_ST3  4       {
          DC.W      PR31_ER7-PR31_ST3  5       }
          DC.W      PR31_SF1-PR31_ST3  6       \
          DC.W      PR31_ER7-PR31_ST3  7       >
          DC.W      PR31_ER7-PR31_ST3  8       |
          DC.W      PR31_ER7-PR31_ST3  9       ~
          DC.W      PR31_ER7-PR31_ST3 10      'u'

```

	DC.W	PR31_ER7-PR31_ST3	11	';
	DC.W	PR31_ER7-PR31_ST3	12	<
	DC.W	PR31_SC-PR31_ST3	13	;
PR31_SK	MOVE.B	SBT1(A6),D0	get the substitute for	
PR31_SA	SUBQ.B	#2,D6	do we need a comma? . .	
	BNE	PR31_SA1	. . no	
	MOVE.B	#',',(A2)+		
PR31_SA1	MOVE.B	D0,(A2)+	copy character	
	MOVEQ	#0,D6		
PR31_SA2	LEA	PR31_ST1,A5	-> state 1	
PR31_SB	BRA	PR31_NXT		
PR31_SF	SUBQ.B	#2,D6	do we need a comma? . .	
	BNE	PR31_SC	. . no	
PR31_SF1	MOVE.B	#',',(A2)+		
PR31_SC	MOVEQ	#0,D6	reset marker	
	MOVE.B	#10,(A2)+	mark end (LF)	
	MOVEM.L	(A7)+,PR31_QR	restore registers	
	LEA	MAC_BUF(A6),A1	set A1 for parameter input	
	RTS		Exit - translation done	
PR31_SD	SUBQ.L	#2,D6	comma been? . .	
	BEQ	PR31_SD1	. . no	
	MOVE.W	#"<>",(A2)+	[Do not try this on 680200!!]	
PR31_SD1	MOVE.B	D0,(A2)+		
	MOVEQ	#0,D6	Clear the marker	
	BRA	PR31_NXT		
PR31_SE	SUBQ.L	#2,D6	comma been? . .	
	BNE	PR31_SE1	. . yes	
	MOVE.B	#',',(A2)+		
PR31_SE1	MOVE.B	#"<",(A2)+		
	LEA	PR31_ST2,A5	-> state 2	
	BRA	PR31_NXT		
PR31_SI	SUBQ.L	#1,D6	special <>? . .	
	BNE	PR31_SI2	. . no	
	MOVE.B	#">",(A2)+	end the <...> string	
PR31_SI2	MOVEQ	#0,D6	clear markers	
	MOVE.B	D0,(A2)+	put comma	
PR31_SI1	LEA	PR31_ST0,A5	-> state 0	
	BRA	PR31_NXT		
PR31_SL	MOVE.B	SBT2(A6),D0	get the substitute for ~	
	BRA	PR31_SA		
PR31_SO	MOVE.B	#',',(A2)+	to replace }	
	LEA	PR31_ST3,A5	-> state 3	
	MOVEQ	#0,D6	clear the marker	
	BRA	PR31_NXT		
PR31_SP	SUBQ.L	#1,D6	special <>? . .	
	BNE	PR31_SF1	. . no	
	MOVE.B	#">",(A2)+	end the <...> string	
	BRA	PR31_SF1		
PR31_SM	SUBQ.L	#1,D6	special <>? . .	
	BNE	PR31_SM1	. . no	
	MOVE.B	#">",(A2)+	end the <...> string	
PR31_SM1	MOVEQ	#2,D6	comma may be needed	
	BRA	PR31_SI1	-> state 0	
PR31_SJ	CMPI.B	#1,D6	special <>? . .	
	BNE	PR31_SH	. . no	
PR31_SU	MOVE.B	D0,(A2)+	copy twice	
PR31_SH	MOVE.B	D0,(A2)+		
	BRA	PR31_NXT		


```

PR31_SQ  MOVE.B    SBT1(A6),D0      get the substitute for |
        BRA       PR31_SH

PR31_SR  MOVE.B    SBT2(A6),D0      get the substitute for ~
        BRA       PR31_SH

PR31_SS  SUBQ.L    #2,D6             comma needed? . .
        BNE       PR31_SS1         . . no
        MOVE.B    #'',(A2)+
PR31_SS1 MOVE.W    #"<",(A2)+
        MOVEQ     #1,D6             mark special <>
        BRA       PR31_SA2         -> state 1

PR31_ST  SUBQ.L    #1,D6             inside special <>? . .
        BNE       PR31_SC         . . no
        MOVE.B    #">",(A2)+       end the <...> string
        BRA       PR31_SC

```

```

; The following all produce error messages in the listing.
; The parameters are then all set to null for the macro

```

```

PR31_ERX  BSR      DO_ERR            insert the error
        BRA       PR31_F            return with null parameters

PR31_ER5  MOVE.L    #86,(A7) set D1
PR31_ER5A MOVEM.L   (A7)+,PR31_QR    restore registers and set D1
        ADDQ.L    #4,A7             jump over RTS
        BRA       PR31_ERX

PR31_ER6  MOVE.L    #87,(A7) set D1
        BRA       PR31_ER5A

PR31_ER7  MOVE.L    #33,(A7) set D1
        BRA       PR31_ER5A

```

Curve Smoothing

Stephen Poole

In a forthcoming article I will explain how to join up dots with smooth curves, an exercise which turned out to be far more difficult than at first envisaged.

But this time this article will demonstrate a simple method of smoothing jagged lines using 'Bezier Curves'. These were invented by a Renault Engineer of the same name, to allow technical drawings of prototypes to be easily digitised, and to help calculate trajectories for numeric machine tools and robots.

Bezier Curves are calculated by finding the mid-points of successive lines and then joining those mid-points one to another, then repeating the procedure from the first to the last point until the connecting lines are close enough together to approximate a curve. The advantage of this method is that it is straightforward and therefor

fast. The inconvenience is that these curves miss all the initial points, and so are unsuitable for specific functions such as Contour drawing.

Whilst writing the Bezier program, I doodled around with it whilst debugging, which revealed by animation the evolution of the calculations, and hence all the intermediate stages. At the time I had just read one of Stephen Hawkings books, and especially liked his descriptions of Einsein's Theory of Relativity. This explains that the only universal dimension is Space-Time which amongst other properties is warped by gravity. So I drew a space-time graph with the horizontal time-axis bending back onto itself, (which is taking time-warping beyond its limits, but as I said I was just experimenting with the program). The result was the following animated routine: Warp_bas. Try experimenting if you wish, a good starting-point being the variable 'n'. Take care if you multitask programs having different MODEs, as the QL gets a little temperamental. Who said the QL couldn't access the Web?

```

100 ::
110 REMark Warp_bas. by S.Poole, v15mar92, v27dec2004.
120 REMark (Hit ESC key to interrupt whilst drawing).
130 REMark (Hit ESC key to QUIT when drawing done).
140 :
150 CLEAR: OPEN#1,con_16: MODE 8: PAPER 0: z=256: SCALE z,25,25

```

```

160 WINDOW 512,z,0,0: BORDER 0: OVER 0: FILL 0: INK 1: CLS
170 dx=150: dy=dx*.75: n=100: dv=n DIV 8: i=1: x=1: y=2: c$=CHR$(27)
180 DIM t(n,y): t(1,x)=230: t(1,y)=150: t(n,x)=230: t(n,y)=150
190 :
200 REPEAT loop
210   FOR f=2 TO n-1
220     tfx=t(f-1,x)+RND(-dx TO dx): IF tfx>430 OR tfx<-30: GO TO 220
230     tfy=t(f-1,y)+RND(-dy TO dy): IF tfy>z+60 OR tfy<-30: GO TO 230
240     t(f,x)=tfx: t(f,y)=tfy: POINT tfx,tfy
250   END FOR f: f=1
260   :
270   FOR j=n TO 2 STEP -1
280     i$=INKEY$(#1): IF i$=c$: i=1: CLS: NEXT loop
290     ox=t(f,x): oy=t(f,y): INK i
300     t(f,x)=(ox+t(f+1,x))/2: t(f,y)=(oy+t(f+1,y))/2
310     LINE ox,oy TO t(f,x),t(f,y)
320     :
330     FOR f= 2 TO j-1
340       t(f,x)=(t(f,x)+t(f+1,x))/2: t(f,y)=(t(f,y)+t(f+1,y))/2
350       LINE TO t(f,x),t(f,y)
360     END FOR f
370     :
380     LINE TO t(f+1,x),t(f+1,y): f=1
390     IF (j MOD dv)=1: i=i+1: IF i>7: i=1
400   END FOR j
410   :
420   i$=INKEY$(#1,500): IF i$=c$: EXIT loop: ELSE i=1: INK i: CLS
430 END REPEAT loop: MODE 4
440 ::

```

Keyboards and Perfection

George Gwilt

I use The Editor, Perfection and QD to examine, produce and edit files of all sorts. Probably QD is the program I use most frequently, but the other two are indispensable. In particular I use Perfection when I want to print a file. Occasionally I want to reformat the file before printing. Perfection allows you to reformat part of a paragraph by pressing ALT/CAPS and to reformat to the end of the document with SHIFT/ALT/CAPS. I was thus annoyed when I found that I could not do that with my Q40 and Q60. Nor can you do it with QPC2. All seven of the combinations of CAPS with SHIFT CONTROL and ALT have absolutely no effect.

They do on the original unexpanded QL of course.

That was some time ago now. In fact it was only recently that I found time to solve this problem. Before I tell you how this was done, I must first explain how the Q40 and the Q60 deal with the keyboard.

Hardware

On the Q40/60 the keyboard is controlled by the "master chip". This intercepts information from the keyboard and sets it in the "keycode register" at address \$FF00001C. This register is regularly polled by a program in SMSQE which processes the information and sets it in partially digested form into an initial buffer. A second polling program, smsq_kbd_poll.asm, extracts the information from this buffer, decodes it, and puts the result into the current keyboard queue.

A user has the power to determine just what characters are sent to the current keyboard queue by this second program by setting his own array of private keyboard tables using the machine procedure KBD_TABLE. Because I was not able to find definitive documentation detailing how this was to be done I had to experiment on my own and this is what I found.

The information from a keyboard gives a "scan code" for each key and this value is passed to SMSQE which places this value in the initial keyboard buffer. It is these "scan codes" that keyboard tables translate.

The keyboard tables in SMSQE have entries for each of the "scan codes" from 0 to \$7F inclusive.

TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets. Can be used with lights/hifi/monitors—ie a QL monitor can be used as a switch control.

Cost £24

superHermes

A major hardware upgrade for the QL. All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/keyclick) IBM AT kbd I/F // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software)£90 (£92/£93)
IBM AT UK layout Keyboard.....£11 (£13/£15)
Serial mouse.....£8 (£8.50/£9)
Capslock/scrolllock LED£1 (£1.50/£1.50)
Keyboard or mouse lead£3 (£3.50/£3.50)
High speed serial (ser3) lead.....£4 (£4.50/£4.50)

Hermes available for £25 (£26/£27) Working ser1/2 and independent input, debounced keyboard.

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.

Cost (incl keyboard lead)£53 (£54/£55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thom-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating system/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+ Hermes) & built in Multibasic.

First upgrade free. Otherwise send £3 (+£5 for manual if reqd).

Send disk plus SAE or two IRCs

MK1...£40 (£41/£43) MK11...£65 (£66/£67)

MINERVA RTC (MK11) + battery for 256 bytes ram. CRASHPROOF clock & I²C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL. A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage - it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second. Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off RomDisq at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq.....£39 (£40/£41)
4mbytes RomDisq.....£65 (£66/£67)
8 mbytes RomDisq.....£98 (£99/£100)
Aurora adaptor.....£3 (£3.50/£4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost£34 (£35/£36)

I2C INTERFACES

Connects to Minerva MK11 and any Philips I²C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)

2 amp (for 8 relays, small motors).....£40 (£43/£44)

4 amp total (for motors etc).....£45 (£48/£50)

Relays (8 3a 12v 2-way mains relays (needs 2a power driver).....£25 (£28/£29)

Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required.....£25 (£27/£28)

Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue outputs (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting.....£30 (£31/£32)

Temp probe (-40°C to +125°C).....£10 (£10.50/£11)

Connector for four temp probes.....£10 (£10.50/£11)

Data sheets.....£2 (£2.50/£3)

Control software & manual (for all I/F).....£2 (£2.50/£3)

QL SPARES

Keyboard membrane no longer on sale
1377 PAL£3 (£3.50/£4)
Circuit diagrams£3 (£3.50/£4)
68008 cpu or 8049 IPC.....£8 (£8.50/£9)
8301/8302 or JM ROM or serial lead.....£10 (£10.50/£11)
Power supply (sea mail overseas).....£12 (£19/£23)
Other components (sockets etc) also available

Prices include postage and packing (Airmail where applicable) Prices are: UK (Europe /Rest of world). Payment by cheque drawn on bank with UK address/postal order or CASH! I can no longer accept card payments as UK only does PDQ transaction. SAE or IRC for full list and details

22 Feb 03

29 Longfield Road, TRING, Herts, HP23 4DG

Tel: +44 (0) 1442-828254

Fax/BBS: +44 (0) 1442-828255

tony@firshman.co.uk

http://www.firshman.co.uk

There are five of these tables. The first is for codes without SHIFT or CONTROL. The second is with SHIFT, the third is with CONTROL, the fourth is with both SHIFT and CONTROL and the fifth is with Alt Gr. The action of ALT is different. For the set of keys which produce a single ASCII code (keys such as ALT/CTRL/UP) the value obtained from one of the first four tables has one added to it. Thus the code for CTRL/UP found in the third table, is 210 and for ALT/CTRL/UP is 211.

Setting Private Tables

To produce your own private keyboard tables you have to know how these relate to the keyboard itself. As has been described earlier, each key has a scan code and this gives the position in the tables. Thus for scan code 62, which is the scan code for the key D on an English keyboard, if you wanted to have the same result as for an ordinary QL you would put 100 in position 62 in the first table, 68 in position 62 of the second table, 4 in position 62 of the third table and 164 in position 62 of the fourth table.

On my computers I have keyboards which contain at least the 102 standard keys. Some keyboards have extra keys, but these are not recognised by the Q40/60 master chip and so have no effect.

The relationship between keys and scan codes is given in three tables below.

Table A gives a schematic layout of a keyboard with a number for each key. This number is purely arbitrary and is there merely to identify a key and, having been assigned by myself, will not relate to anyone else's numbering.

Table B shows for each scan code the number of the key it represents.

Table C gives, for each key number, both the name of the key itself and also its scan code.

These three tables enable you to find easily the scan code for any particular key and also the key for any particular scan code.

This information is enough to enable anyone to make up his own private keyboard tables.

Table E below shows what I did for my own use.

Loading the Tables

Once the tables are prepared they have to be stored in RAM and then linked to the operating

system by KBD_TABLE. There are two ways of doing this. Either a complete language module can be set up or just a private keyboard table. The first of these will have, in addition to a set of keyboard tables, information about error messages and so on which can be invoked by LANG_USE. It is the second way that I will now describe.

In addition to the five tables already dealt with there must be a table of "non-spacing indents". These concern (I believe) the ability to type accents without the cursor moving to the next position. At any rate this table is completely blank for an English keyboard. It consists of 256 zero bytes.

The set of all these tables must be preceded by two words giving the relative positions of the start of the set of five tables and then the start of the non-spacing indents.

This can be set in a file either by assembling a set of DC instructions or by S*BASIC. I used the latter method and my program, adapted from that given in the source code for SMSQE which was written to set up a whole language module, is shown below as **Table F**. One reason for using my own program was that the SMSQE program does not allow for the Alt Gr table.

Table D gives more information about loading the private keyboard table.

Back to Perfection

I started out by indicating that I had solved the problem of using ALT/CAPS and SHIFT/ALT/CAPS with Perfection. Readers may have noticed that I have not yet explained how I did this. True, the entries in second to fourth keyboard tables at \$3A, which is the scan code for CAPS, give the ASCII codes needed, but unfortunately the polling program, smsq_kbd_poll_asm, does not allow CAPS with any other combination of keys to produce any entry in a keyboard queue.

What I did was to add two instructions to the program at "kbp_caps".

The instructions were:

```
tst.w    kb_stat(a3)    ; If any of SHIFT,
                        ; CONTROL, ALT ..
bne      kbp_norm      ; .. set a normal
                        ; code
```

When I compiled the new SMSQE (using GWASS), I called it, in anticipation, v4.00.

Table A

Keyboard Layout (with key numbers)

Each of the 102 keys in the IBM keyboard is given a number as shown:

ESC 1	F1 2	F2 3	F3 4	F4 5	F5 6	F6 7	F7 8	F8 9	F9 10	F10 11	F11 12	F12 13
----------	---------	---------	---------	---------	---------	---------	---------	---------	----------	-----------	-----------	-----------

f 17	1 18	2 19	3 20	4 21	5 22	6 23	7 24	8 25	9 26	0 27	- 28	= 29	<-- 30
---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	-----------

TAB 38	Q 39	W 40	E 41	R 42	T 43	Y 44	U 45	I 46	O 47	P 48	[49] 50	ENTER 71
-----------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	-------------

CAPS 59	A 60	S 61	D 62	F 63	G 64	H 65	J 66	K 67	L 68	; 69	' 70	# 51
------------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------

SHIFT 76	\ 102	Z 77	X 78	C 79	V 80	B 81	N 82	M 83	, 84	. 85	/ 86	SHIFT 87
-------------	----------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	-------------

CTRL 92	ALT 93	SPACE 94				ALT GR 95	CTRL 96
------------	-----------	-------------	--	--	--	--------------	------------

Prt Sc 14	Scroll 15	Pause 16
--------------	--------------	-------------

Insert 31	Home 32	Pge Up 33	Num 34	/ 35	* 36	- 37
--------------	------------	--------------	-----------	---------	---------	---------

Delete 52	End 53	Pge Dn 54	7 55	8 56	9 57	+ 58
--------------	-----------	--------------	---------	---------	---------	---------

4 72	5 73	6 74
---------	---------	---------

@ 88	1 89	2 90	3 91	ENTER 75
---------	---------	---------	---------	-------------

<- 97	98	-> 99	0 100	. 101
----------	----	----------	----------	----------

Each of the keyboard tables contains values for most of these keys. The keys not given a value in the tables include the SHIFT, CONTROL and ALT keys as well as CAPS (on its own) and SCROLL, although these keys, of course, have a scan code.

Table B
Scan Code to Key Number

The following table gives, for each scan code, the number of the key it represents. Thus the entry at row 3 column 5, which is 86, shows that key 86 has scan code \$35.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	-	1	18	19	20	21	22	23	24	25	26	27	28	29	30	38
1	39	40	41	42	43	44	45	46	47	48	49	50	71	92	60	61
2	62	63	64	65	66	67	68	69	70	17	76	51	77	78	79	80
3	81	82	83	84	85	86	87	36	93	94	59	2	3	4	5	6
4	7	8	9	10	11	34	15	55	56	57	37	72	73	74	58	89
5	90	91	100	101	14	35	102		95				75	96		
6						12	13	32	88	33		97		99		53
7	98	54	31	52		16										

When a key is pressed, the initial keyboard routine translates that to a "scan code". This code is used by the polling routine to send an actual ASCII value to a keyboard queue. The "scan code" is used as the address into one of the five keyboard tables from which the actual ASCII code is drawn.

Table C
Key Numbers to Key and to Scan Code

Table A assigns a number to each key. To find the corresponding scan code you can either search Table B for the position of the key number, or, more easily, use this table to find the scan code.

Given a scan code you can use Table B to determine the key number and then you can use this Table to find the key.

Key No	Key	Scan	Key No	Key	Scan
1	ESC	01	2	F1	3B
3	F2	3C	4	F3	3D
5	F4	3E	6	F5	3F
7	F6	40	8	F7	41
9	F8	42	10	F9	42
11	F10	44	12	F11	65
13	F12	66	14	PRT SC	54
15	SCROLL	46	16	PAUSE	75
17	'1'	29	18	'1'	02
19	'2'	03	20	'3'	04
21	'4'	05	22	'5'	06
23	'6'	07	24	'7'	08
25	'8'	09	26	'9'	0A
27	'0'	0B	28	-	0C
29	=	0D	30	<-- (Delete)	0E
31	Insert	72	32	Home	67
33	Page Up	69	34	Num Lock	45
35	Num /	55	36	Num *	37
37	Num -	4A	38	TAB	0F
39	Q	10	40	W	11
41	E	12	42	R	13
43	T	14	44	Y	15
45	U	16	46	I	17
47	O	18	48	P	19

49	[1A	50]	1B
51	#	2B	52	Delete	73
53	End	6F	54	Page Down	71
55	Num 7	47	56	Num 8	48
57	Num 9	49	58	Num +	4E
59	Caps Lock	3A	60	A	1E
61	S	1F	62	D	20
63	F	21	64	G	22
65	H	23	66	J	24
67	K	25	68	L	26
69	;	27	70	'	28
71	ENTER	1C	72	Num 4	4B
73	Num 5	4C	74	Num 6	4D
75	Num Enter	5C	76	Left SHIFT	2A
77	Z	2C	78	X	2D
79	C	2E	80	V	2F
81	B	30	82	N	31
83	M	32	84	,	33
85	.	34	86	/	35
87	Right SHIFT	36	88	@	68
89	Num 1	4F	90	Num 2	50
91	Num 3	51	92	Left Ctrl	1D
93	Alt	38	94	SPACE	39
95	Alt Gr	58	96	Right Ctrl	5D
97	<-	6B	98		70
99	->	6D	100	Num 0	52
101	Num .	53	102	\	56

Table D

How to Load a Keyboard Table

The manual for Q40/Q60 indicates that you can load a private keyboard table by the following

```
i = RESPR(512): LBYTES "kt",i: KBD_TABLE i
```

The manual goes on to warn that a "... keyboard table loaded in this way should not be prefaced by a flag word".

Unfortunately there appears to be no mention of the format of the table. The program for the keyword KBD_TABLE requires the following for a "private" table such as kt.

```
H1 word offset to the start of the main table
H2 word offset from bytes 2 to the start of the non-spacing idents
T1 main table
T2 non-spacing idents
```

The tables T1 and T2 can be in either order. T1 consists of the five blocks mentioned above each \$80 bytes long. Table T2 is 256 bytes long. This makes the file 900 bytes long.

If T1 is set first, then the private table file should start with two words, 4 then 642 followed by the five blocks of main table and finally 256 zero bytes.

Table E

My Main Table

I have set my main table slightly differently from the standard one supplied.

Key	Effect	Standard
F11	CTRL/F1	(nothing)
F12	CTRL/F2	"
End	Alt ->	Shift/Alt/Down
Home	Alt <-	Shift/Alt_Up
Page Up	Shift/Alt /Up	Shift/Up
Page Down	Shift/Alt/Down	Shift/Down
Alt Gr	This, when held down, counteracts Num Lock	

Unfortunately, we cannot present Table F - the listing - as it seemed to have arrived incomplete. We tried to reach George for a complete listing but did not get it in time for the printer... you will have to wait for the next issue or check the QL Today homepage, sorry.

Changes at J-M-S ... the Move part 2

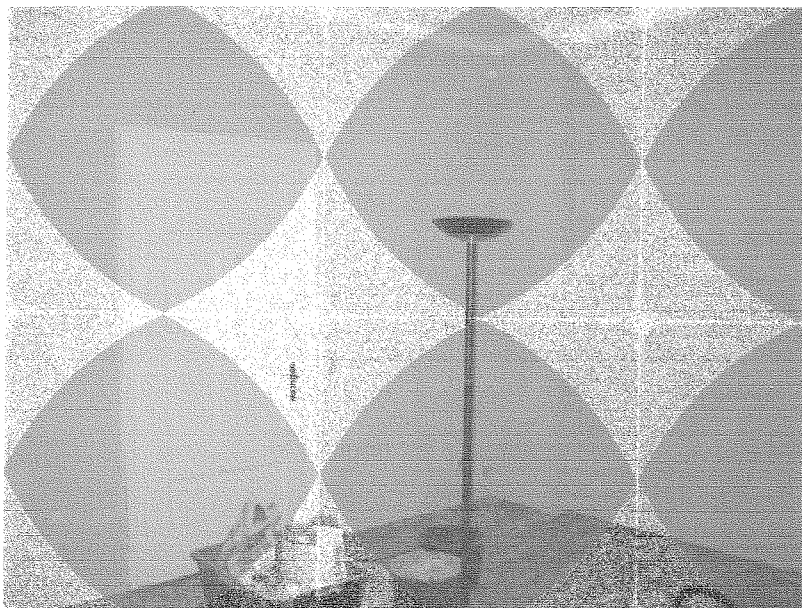
Jochen Merz

Last issue I gave you a look at the office-move situation at J-M-S.

You probably guessed it - the office is still at the old address.

Not surprising, as again several problems were forced onto me. Also, much time was spent on more enjoyable things like QL matters: QDT had to be released, Balled had to be tested and released, QPCPrint had to be tested, QL Today has to be prepared - and the main shareware job required some attention as well - some money needs to come in as well.

Still, progress can be reported: the "server" room is nearly finished. And the office: the electrics



and half the walls are also done (left picture) - and even the new safe is installed (see picture above)! But this wall still needs to get dry so we are a bit stuck there. Also, we discovered today that the window sill we ordered was 6cm wider than our specification... well, that did not surprise us very much as everything we touch seems to turn into a problem. Still smiling :-)

The floor is still in a very bad condition after some days work. Still far from "ready" - and a move in February is obviously again out of the question.

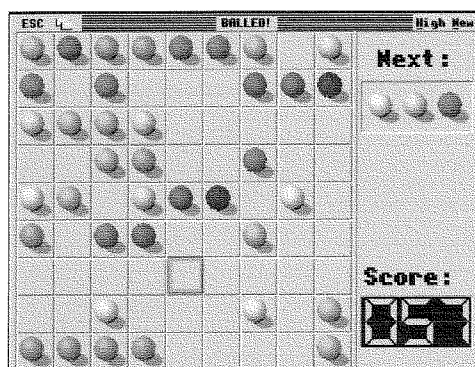
I'll keep you informed.

JOCHEN MERZ SOFTWARE

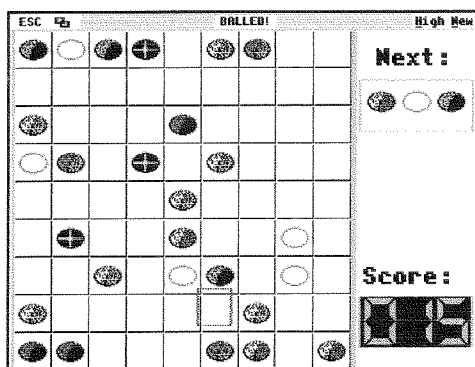
Kaiser-Wilh.-Str. 302 D-47169 Duisburg
Tel. 0203 502011 Fax 0203 502012
<http://smsq.j-m-s.com> smsq@j-m-s.com

The new game is here: **BALLED!**

Balled is the new Game from Wolfgang Lernerz for QDOS and SMSQ/E. It runs in high-colour mode (looks great) and can also be played in standard QL Mode 4. Have a look at the following screenshots (Balled in Hi-Colour mode and Balled in MODE 4):



Also, Balled uses the digital sound system if you run it on a recent version of SMSQ/E and QPC, Q40 and Q60! The aim of the game is to build lines by moving coloured balls around, following some rules, of course. After every move, more balls appear, so you have to be careful not to fill up the board – then the game is over! Different levels of difficulty and wildcard balls are also featured. Find out how to get bonus and higher scores to fill up the high-score table!



Balled costs only EUR 11,90 plus postage (EUR 2,- Europe, EUR 4,- for the rest of the world).

If you wish to place your order via internet, please do not send your credit card details in an email!
Please use the secure contact form on the SMSQ homepage: SMSQ.J-M-S.COM

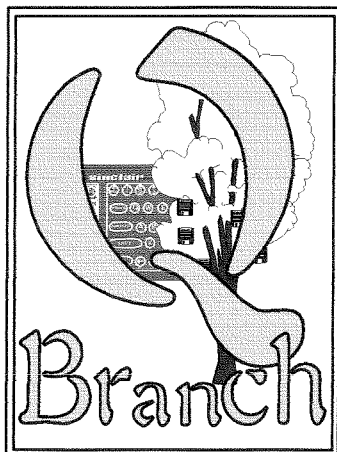
We accept VISA, MasterCard and Diners Club!

NEXT ISSUE WILL CONTAIN SOMETHING MANY READERS ARE WAITING FOR:

INTERNET ON THE QL

... WE WILL START WITH QL EMULATORS - WHICH SHOULD PLEASE THE MAJORITY OF OUR READERSHIP!

WE PLAN TO HAVE THE NEXT ISSUE AVAILABLE AT THE HOVE SHOW!



Text 87

£ 79.00

Typset 94 £ 29.00

Fountext 94 £ 39.00

2488 drivers £ 29.00

Epson ESC/P2 drivers

£ 26.00

Text 87

High Colour

Patch

Available Now

£9.00

PROGRAMMING

QD 2003 £ 49.00

QD + QBasic £ 63.00

QD + Qliberator + QBasic £ 104.00

Qliberator £ 50.00

Master Spy v 3.3 £ 30.00

QPTR £ 32.00

Easyptr pt 1 & 2 (together) £ 30.00

Easyptr pt 3 (C library) £ 14.00

QMake £ 18.00

QMon / JMon £ 22.00

Basic Linker £ 22.00

DISA 3 £ 34.00

QMenu £ 16.00

!!! New Programs !!!

What a good start to the New year! Not only do we have the release of QDT, a program which will revolutionise the way in which the QL works, but we also have a new game BALLED! by Wolfgang Lenerz. Truly a bumper start to the QL 21st anniversary.

To go with this we also have a new release of QPC2 (v3.30) which has TCP/IP support and is substantially faster than the previous version. Users should check the Q Branch website for details on the new features and added keywords.

Here at the QBranch HQ we have also been Beta testing QPCPRINT. This is a new program which will allow QPC2 users to print to any printer attached to the PC that QPC2 is running on. This is not available quite yet but call for details.

What a Happy New Year !

QDT

£ 30.00

+ P&P

£2.00 (UK) £2.50 (EU) £3.00 (ROR)

Requires SMSQ/E to run

Buy QDT and any copy of SMSQ/E and get £10.00 discount

Balled! £ 9.50

SMSQ/E

Gold Card / Atari / QXL Version

Version 3.09 out now! £ 32.00

Aurora High Colour

version available now

£ 56.00 upgrade £ 21.00

Get Cueshell for only £15.00 with any copy of SMSQ/E

UTILITIES

FiFi 2 £ 21.00

QSup £ 30.00

QSpread 2003 £ 51.00

Cueshell 2 £ 15.00

Qload / Qref £ 15.00

Disk Mate 5 £ 16.50

QPAC 1 £ 22.00

QPAC 2 £ 42.00

QTYP 2 £ 31.00

QLQ £ 30.00

QL Keyboard Membranes

Available Now !

£ 17.50 ea + P&P

Q Branch

Feeling out on a limb?
Reach out for Q Branch.
Suppliers of Quality QDOS/SMSQ products
Hardware and Software.

Tel +44 (0) 1273-386030 fax +44 (0) 1273-430501

Mobile +44 (0) 7836-745501

email qbranch@qbranch.demon.co.uk web : http://www.qbranch.demon.co.uk

20 LOCKS HILL, PORTSLADE,
E. SUSSEX. BN41 2LB. UK.

Q Branch

Hardware in Stock Now!

Second User Qubides, Gold and Super Gold
Cards: In stock!
Second User Black Box QLs, tested and
working: In stock!
New Aurora Boards : In Stock !
Many more Items. Call for details.

QPC 2 v3.30

Now Only

£ 65.00

New Version !!

QPCPrint Coming Soon

These programs are now
High Colour

QMAke

WinEd

QPAC 1

QPAC 2

QSUP

<<<<<<NEW>>>>>>

SuQcess

Disk Mate 5

Please send old master
disks and £1.60 to cover
return postage and copying
fees.

Hardware

Second user items are sometimes available
from us.

Call for details.

QXL II £ 40.00*

Recycled superHermes £ 65.00 *

R'cycld SuperGold Card £ 110.00 *

Recycled Gold Card £ 45.00 *

Recycled Aurora £ 65.00*

Qubide £ 45.00*

Qplane £ 5.00

Aurora cables £ 3.00

Aurora rom adaptor £ 3.00

'Arfa Braquet' £ 8.00

MC plate £ 6.50

Qubide upgrades
to version 2.01 £ 8.00

* when available.

Call for details

These Programs are paid Upgrades:

QSpread 2001 to 2003 £ 10.50

QD 98 to 2003 £ 10.50

QPC2 -

update from v3.xx Free

update from v2 £ 13.90

update from v1 £ 34.00

Return Master Disk
With Order

Q Branch Programs

The Knight Safe 3

£ 35.00

Q - Route v1.08C

£ 25.00

upgrades from previous versions

£ 5.00

Route finding programme

Q - Count

£ 25.00

The Fractal Collection

£ 35.00

Pointer driven home accounting

Post and Packing is included with Software Prices
Unless Otherwise Stated.



We can accept payment by MasterCard, Visa, Switch, and Maestro. We can also
accept sterling cheques drawn on a UK bank or payment direct to our UK Bank.
Barclays Bank Account# 50543497 sort Code 20-12-75 Acc name QBranch



QL Software Sources - Part 2

Dilwyn Jones

Here is the second part of the QL software sources article. We apologise that due to a misunderstanding between Quanta and ourselves (we forgot to tell Roy Brereton that the second part of the article in QL Today had been postponed by one issue) Quanta members got to read it before QL Today readers.

Last issue, we stopped at Italy. Another Italian software download site is operated by **Giorgio Garabello**, perhaps better known for starting the "linked list" of QL websites, known as the Quantum Ring. Giorgio hosts an extensive QL software download site at

www.sinclair-ql.it

and the download index page is at:

<http://www.sinclair-ql.it/down/index.html>

This site also presents you with a choice of English or Italian by clicking on the appropriate flag as you enter the home page.

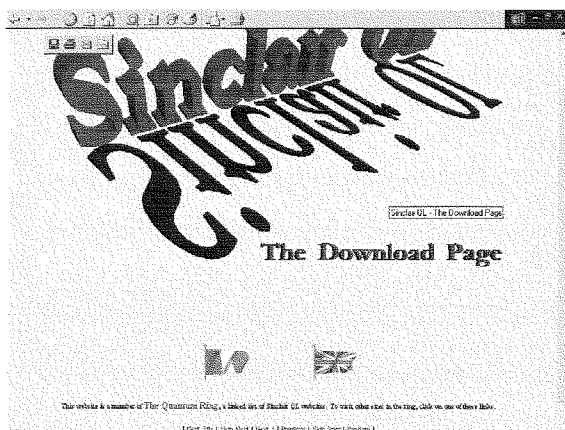


Figure 5 - Giorgio Garabello's Sinclair QL Downloads Site

Spanish Qlers have three websites I would like to refer to. The first is operated by **Miguel Angel Hernandez**. This site includes something called the International Freeware Exchange, a vast collection of free QL software. The site is in Spanish only, but is quite easy to navigate for those who do not speak Spanish. The IFE was formed by pooling free software available from several countries with a strong history of use of the QL such as Sweden, Germany, France, The Netherlands, Italy and Britain. Miguel's site includes material for other Sinclair computers as well, and is at:

<http://homepage.mac.com/miguelahernandez1/Espanyol/Pages/Inicio.html>

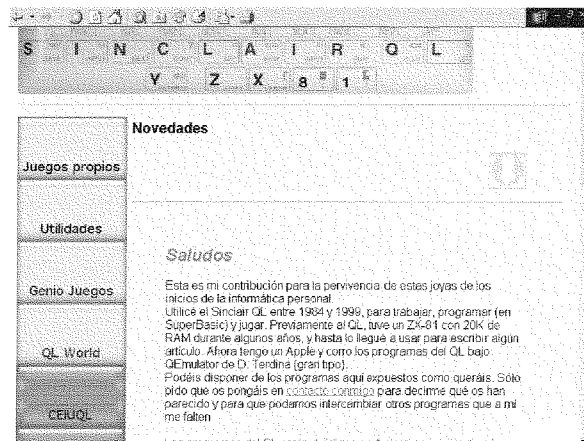


Figure 6 - Miguel Angel Hernandez's website

Recursos en Castellano (or Spanish QL Resources) is a site operated by **Javier Guerra**. Spanish Qlers can download an extensive library of QL magazines or ezines in the Spanish language from this site, at:

<http://sinclairql.info/>



Figure 7 - Javier Guerra's website

Another Spanish website is run by **Pedro Reina**. Pedro has written quite a lot of QL software over the years and he's made it freely available from his site at:

<http://pedroreina.org/>

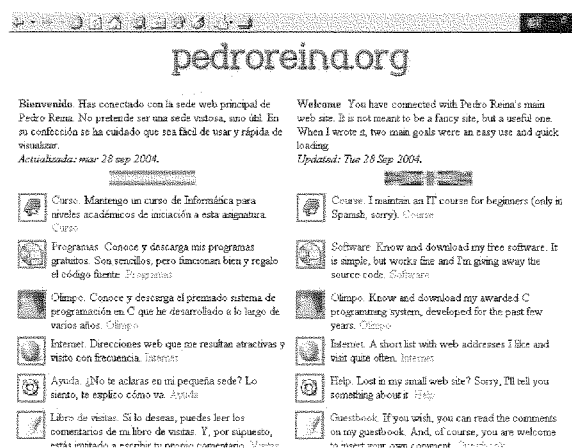


Figure 8 - Pedro Reina's site

Back to Italy, and the Beginners Club is operated for Italian Qlers by **Andrea and Paulo Carpi**. Their site

<http://begclub.bbk.com/ql/>

is available in English and Italian and includes several programs written originally by them for club members.



Figure 9 - Beginners Club of Vercelli, Italy

Dave Walker is another prolific author of QL software, and probably best known for maintaining the C68 C language compilers for the QL. Dave has written other software over the years, including the Discover and Textidy file transfer programs, and the less well known OPD transfer programs for transferring programs between QL and ICL OPD microdrives (anyone still use those?). This is another site largely restricted to an author's own software, but again full of high quality software for the QL. Visit:

<http://homepage.ntlworld.com/itimpi/index.htm>

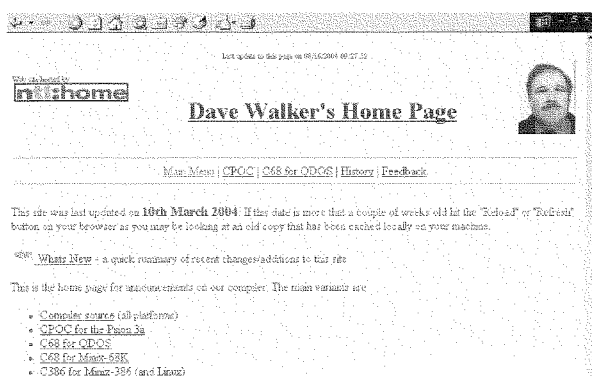


Figure 10 - Dave Walker's website

John Sadler runs the website of SQLUG, the Scottish QL and Linux User Group. This site is the primary source of software written by or maintained by **George Gwilt**, including assembler programs and the Turbo Compiler and associated utilities. There is a wealth of stuff here for BASIC, assembler and C programmers. The SQLUG site is at

<http://www.jms1.supanet.com>

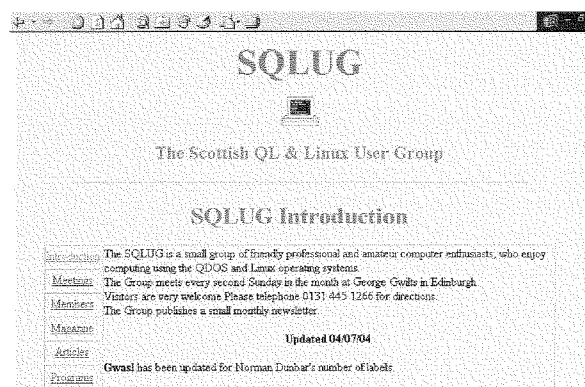


Figure 11 - The SQLUG website

If you are after some GD2 utilities to make use of the high colour modes on modern SMSQ/E systems, you could do worse than visit the sites of **Per Witte** and **Wolfgang Uhlig**. Here you'll find colour selection utilities, games, database demonstration, sprite and menu viewers and a few other general utilities. **Wolfgang Uhlig's** site is at:

<http://www.uhlich.nl/ql/>

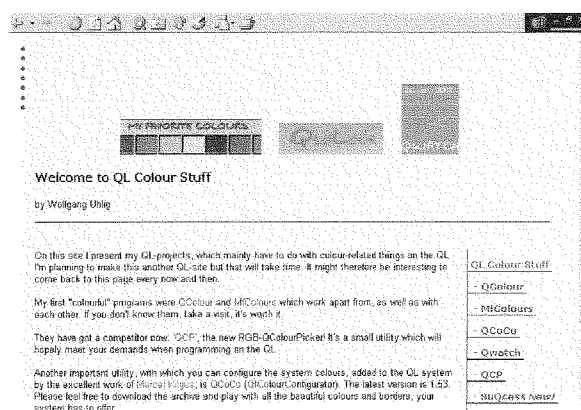


Figure 12 - Wolfgang Uhlig's website

While **Per Witte's** site is 'nowhere to be found', err, sorry, that should have read Knoware, to be found at:

<http://knoware.mysite.freemove.com/>

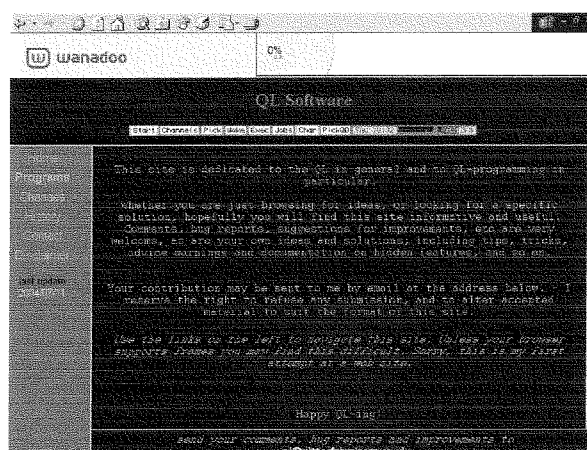


Figure 13 - Per Witte's Knoware site

Over in America, **Tim Swenson** maintains a site with programs and information to download. The programs are mostly Tim's own work – including the Structured SuperBASIC programming aid, FileCfg for creating configuration blocks for BASIC programs. Tim is also editor of the long running ezine, QL Hackers Journal. To visit Tim's site, go to:

<http://www.geocities.com/svenqhj/>

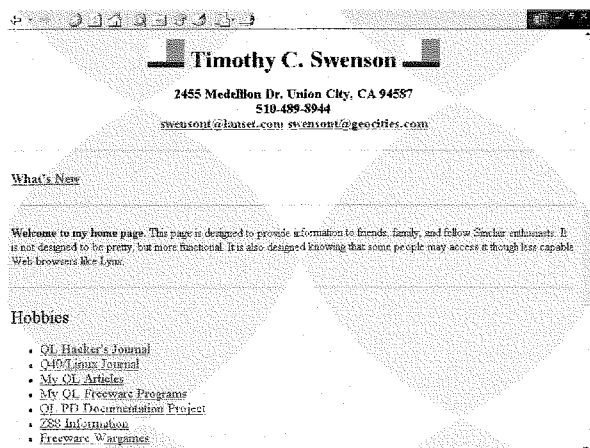


Figure 14 - Tim Swenson's website

Jérôme Grimbert is another prolific author of QL software. His work includes a GD2 sprite editor, games, programming and graphics utilities. His software is available for free download at his website:

<http://jgrimbert.free.fr/>

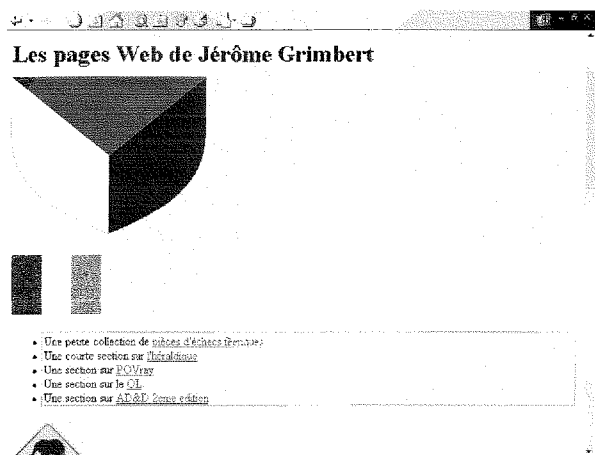


Figure 15 - Jérôme Grimbert's website

Hans-Peter Recktenwald runs a site in Germany dedicated to QL software and Forth programming. It includes a number of pointer interface utilities such as the PEX enhanced windowing utility, several programming utilities, circuit diagrams and something called QVFS, the QDOS Virtual Filing System which sets out to give an internet style file system for the QL. Hans-Peter's site may be found at:

<http://members.tripod.com/phpr/index-ql.html>

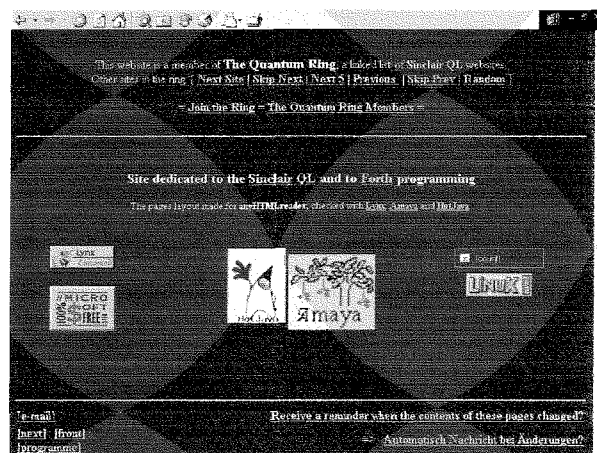


Figure 16 - Hans-Peter Recktenwald's website

Peta's Sinclair QL Support Site includes a huge number of downloadable files, including the Quasar Mailbox Software Download. This is a huge site, based in Germany, with a seemingly never ending list of files to download. The site is split into categories and available at either:

<http://sinclairql.emuunlim.com/>

Or:

<http://www.sinclairql.de>



Figure17 - Peta's website, now where have I seen that opening screen before!

Thomas Much's site includes several games from the 1980s (e.g. MissionX, Diamonds, Q-Ball and Parallax) along with a graphics program (Paintshop) and Graphic Application Manager, a QL front end. These programs date from the 1980s, but you may like to check them out at Thomas Much's website:

<http://www.uni-karlsruhe.de/~Thomas.Much/QL/>
(see figure 18 next page)

Mark Swift is best known to the QL community for his work on the Amiga QDOS Emulator and QDOS Classic for Amiga and Q40/Q60. He has also produced a lot of software which is directly related to these emulators and computers, and

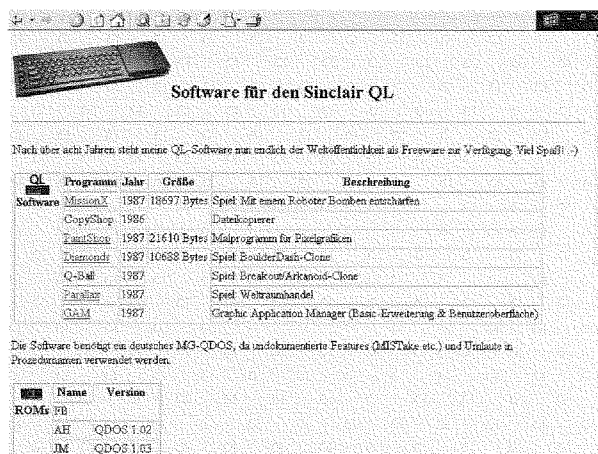


Figure 18 - Thomas Much's website

some more general software such as the PDTK. All his current QL software is on this website. His website is at:
<http://www.mswift.unisonplus.net/ql/index.html>

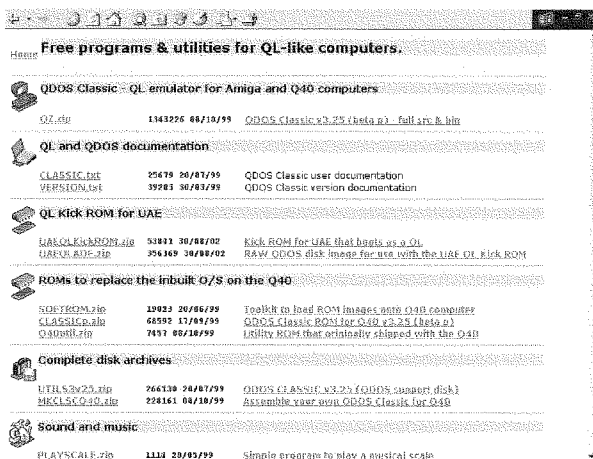


Figure 19 - Mark Swift's website

Joachim van der Auwera of PROGS has made the Prowess windowing system freely distributable. This and all its sources and documentation are available from the PROGS website at:
<http://www.progs.be/>

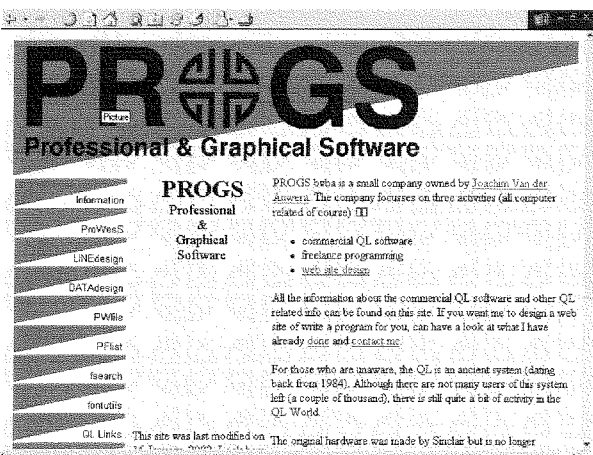


Figure 20 - PROGS website

Interested in the sources for SMSQ/E? The registrar, **Wolfgang Lenerz**, makes these available for those interest in studying the sources and contributing to SMSQ/E development. The source files can be downloaded either complete (a very large file!) or in sections from his site at:
<http://www.scp-paulet-lenerz.com/smsqe/>



Figure 21 - Wolfgang Lenerz's SMSQ/E website

Geoff Wicks's Just Words website contains downloadable demo versions of some of his software range along with downloadable copies of the formerly commercial programs which Geoff made into freeware as time passed. Visit his site at:

<http://members.lycos.co.uk/geoffwicks/justwords.htm>

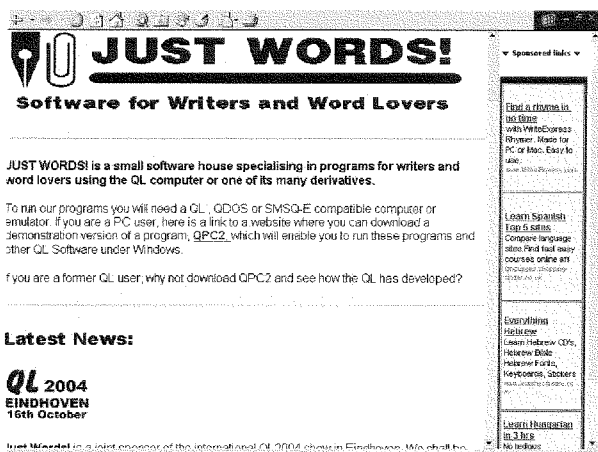


Figure 22 - Geoff Wicks's website

Looking for QDOS or HTML information? Our very own assembler, **Norman Dunbar**, has his website full of QDOS information and a nifty HTML tutorial. Take a PEEK at:
<http://www.bountiful.demon.co.uk/qdos/index.html>
 (please see figure 23 on next page)

Marco Vacquier has allegedly not maintained his Dutch QL pages since 1997 according to the home page. This offers links to the Dutch or En-

success record is not very good, especially Aurora ones. And my ham-fistedness with PCs is probably legendary - a recent posting on the QL-Users Email Mailing List had Roy Wood quickly shouting out "No... don't let Dilwyn near the registry". To some extent, this shows that it is perfectly possible for you to be considered quite knowledgeable in one field but the exact opposite in another. I'm sure this must be true in Roy Brereton's case - he would know everything about his day job, I would probably know nothing at all about it (I'm being cautious because I don't know what it is!)

There is an element of compromise with QL Today content as well. To some extent, we can only publish what we receive and many people tell us they subscribe to QL Today largely for the news side of it, to keep in touch with developments and so on. Authors like Marcel Kilgus, George Gwilt and Wolfgang Lernerz have been very good at writing about their projects, which are often fairly technical in nature (emulators, compilers, programming tools and operating system development respectively).

We also have people like David Denham who tend to write based on what they have been doing of late. David's Clocking In series was one of our most popular series recently, based on feedback received. David's articles are usually about learning some aspect of the QL, such as getting to grips with the QL network in a recent issue (actually despite our long association with the QL both Jochen and I found that article both interesting and instructive), and these may come more under the banner of either beginners or moderate experience levels.

We often find that experienced users (programmers, traders and hardware developers included here) tell us that too much of the stuff is too simplistic and has

been covered too many times in the past (how much more can you teach people about Quill or Archive?).

Whereas less experienced users tell us the magazine is full of assembler, window manager and colour driver stuff and other subjects that simply go over their heads and they can't hope to follow.

What's hard is gauging how many want the advanced stuff and how many want more everyday easier to follow articles. It's tempting to think that as Quanta is a user group with self-help in mind and operates a helpline, that Quanta should cater more for less experienced users and QL Today should concentrate more on information and new, more advanced stuff because that's what we seem to do best. This would be too simplistic an answer of this, though, because a degree of competition is good - one organisation helps to keep the other on their toes and it encourages us both to understand and provide what the readership wants.

The content of QL Today reflects what we are sent for publication fairly accurately. The authors mentioned above provide helpful articles and information on the latest stuff, Norman Dunbar and Herb Schaaf provide ongoing "learning" articles in their respective fields and people like David Denham and myself supply articles which we hope are interesting, relevant and hopefully not too technical. Finally, we can always rely on Geoff Wicks and Roy Wood to provide the views and controversy!

I've written an article about Xchange which depending on space available will appear either in this issue or the next one. Xchange is the next step of the Psion Quill, Archive, Abacus and Easel programs supplied with the original QL, but a version with more integration, with all of the programs working

more as one program. The Xchange versions of these programs are very similar to the originals, but with some improvements and slight differences here and there which may confuse the less experienced user. This article is written as a series of Questions And Answers based on real questions asked to me about Xchange.

At this point I'd like to invite feedback, both on the Xchange article if we manage to squeeze it into this issue and on this issue in general.

1. Do you feel that the Xchange article is useful and readable, or do you think it's too simplistic given the number of years that the QL and Quill have been around? Is anyone interested in articles about Quill or Xchange any more? Do you think people would like to see this taken one step further and provide a guide to where to get hold of Xchange, how to set it up and start using it on a QL? Are there other programs you'd like us to cover in this way?

2. Do you think this issue includes too many articles which are too technical in nature? Or do you think that there is a reasonable mix of advanced and less advanced articles? Is there someone you'd like to write an article on a particular subject you think they know more about?

Please do write or send an email to us to let us know how you feel.

Even better, write a short article for us on a subject you feel would be about right for the average readership level and we'll look at what we are sent and try to spot gaps in what we are catering for. We are always on the lookout for new articles to publish, especially ones from people who have never or rarely written for us before. We are not able to pay for articles, but you do get to see your work in print and you may even make new friends and contacts because of it!

Programming in Assembler - Part 13

Norman Dunbar

Recursion in Assembly Language

After the recent musings on single and double linked lists, this time I'm going to delve into the murky depths of a subject I've never seen before discussed for QDOSMSQ assembly language. The subject is recursion.

Recursion is a very simple concept, but for some people, it can be quite difficult to get their head around it, and it never comes clear. I suspect for those people, trying to do it in assembly language is equally difficult. Lets hope I can explain it in simple enough terms for even me to understand!

A recursive routine is simply a routine which calls itself until a certain exit condition is detected. The exit condition is very important, if you miss it out, your programs will loop around until such time as the stack fills up and the program crashes, or eats itself.

Here's a very simple example of a program that will recurse 'forever' until it dies.

Start	<code>bsr.s</code> <code>rts</code>	Recurse
Recurse	<code>bsr.s</code> <code>rts</code>	Recurse

None of the RTS instructions will ever get executed, because all the program does is calls Recurse over and over again, but each call is nested inside the previous call, so the (A7) stack pointer keeps going down by 4 each time it is called as the BSR instruction stacks the return address and then branches off to the next iteration of Recurse.

Recursion educators are very fond of certain examples when teaching recursion. The towers of Hanoi, Factorials, Exponentiation, Fibonacci numbers etc. I'm no different, so here's a few small explanations and examples.

Factorials

The factorial of a number is that number, multiplied by the factorial of the number before it. So, 4 factorial = 4 * 3 Factorial. There is no concept of a negative factorial, so -3 factorial, doesn't exist. The smallest factorial number is 1 factorial, which has the value of 1.

Factorial numbers are usually written as $n!$ where $n! = n * (n-1)!$ and so on. This implies recursion and we have the following simple code.

Just before we delve into the code be aware that factorials get very big very quickly, 12! (\$1C8CFC00) is the largest that can fit into a single 32 bit (unsigned) register and 13! (\$17328CC00) cannot fit. The largest factorial we can fit into a 16 bit unsigned word is only 8! (\$9D80) and as the 68008 processor can only multiply 16 bit numbers, this means that 9! will be the biggest that the following routine can calculate without overflowing.

Note, other processors in the MC68000 family can multiply 32 bit numbers.

On entry to the code, D0W is the number to calculate the factorial of and on exit, D0L is the result. D0W must in range 1 to 9 only but this routine does not perform error trapping - on your own head be it!


```

Start      bra.s      Start2      ; Skip over result area
Result     dc.l      0            ; Result area
FirstNumber dc.w      9            ; Assume 9! by default

Start2     lea        Result,a3    ; Where to shove the answer
           moveq     #0,d0         ; Clear all 32 bits of D0.L
           move.w    4(a3),d0      ; Fetch FirstNumber
           bsr.s     Factorial     ; Do it
FRET_1     move.l     d0,(a3)      ; Shove it !
           moveq     #0,d0         ; No errors
           rts                ; Back to SuperBasic

Factorial  move.w     d0,-(a7)      ; Stack current number
           subq.w    #1,d0         ; Calculate previous number
           bne.s     Not_zero      ; Not done unless next number is zero
FPOP       move.w     (a7)+,d0      ; Retrieve the value of 1 from stack
           rts                ; Back to FRET_1 if FirstNumber was 1 else
FRET_2

Not_zero   bsr.s     Factorial     ; Lets go round again, and again
FRET_2     mulu      (a7)+,d0      ; Do the multiplication
           rts                ; Exit

```

For the simplest case, assume we start with a value of \$0001. The stack will look like this at label 'Start':

Return address to SuperBasic.

As we drop through the code beginning at 'Start2' and execute our first branch to subroutine 'Factorial' the stack now looks like this:

Return address to SuperBasic.
Return address to FRET_1

Tracing through the 'Factorial' code, we stack the current value of D0W (which is 1) so the stack now looks like this:

Return address to SuperBasic.
Return address to FRET_1
 \$0001

After the subtraction, D0 has become zero, so we exit out of the 'Factorial' code from label FPOP by popping the value \$0001 off the stack into D0W leaving the stack like this:

Return address to SuperBasic.
Return address to FRET_1

Then we execute the RTS instruction to return us to the label FRET_1 where we store the result of \$00000001 from D0.L into the result area set aside for this very purpose.

So far so good, we haven't actually done any recursion yet, but read on. If we start with the value of \$0002 in 'FirstNumber' then the process is slightly different. We start, as ever with the SuperBasic return address on the stack when we are at label 'Start'.

Dropping into 'Start2' and executing our first BSR to 'Factorial', the stack is as above at the same point. Nothing much has changed. However, we then stack the current value in D0W to give a stack as follows:

Return address to SuperBasic.
Return address to FRET_1
 \$0002

This is slightly different. When we calculate the next number, we do not set D0W to zero, so we skip out of the 'Factorial' code block to the code at 'Not_zero' which immediately causes another BSR to 'Factorial' leaving the stack as follows:

```
Return address to SuperBasic.  
Return address to FRET_1  
$0002  
Return address to FRET_2
```

Once again, we stack D0W and subtract one to find that we have now reached zero. The stack looks like this:

```
Return address to SuperBasic.  
Return address to FRET_1  
$0002  
Return address to FRET_2  
$0001
```

Once more, we pop the value \$0001 off the stack back into D0W and then execute the RTS instruction. This time, however, we do not return to FRET_1 but to FRET_2 where we end up with the following stack arrangement:

```
Return address to SuperBasic.  
Return address to FRET_1  
$0002
```

The instruction at 'FRET_2' causes the top word on the stack to be multiplied by D0W the result store in D0L. This leaves D0L equal to \$00000002 which just happens to be the correct value for 2! and we exit the code by returning to 'FRET_1' where we store the result again.

The process is similar for all the other numbers, so 5! will have a stack looking like this when we reach, but just before we execute the code at 'FPOP':

```
Return address to SuperBasic.  
Return address to FRET_1  
$0005  
Return address to FRET_2  
$0004  
Return address to FRET_2  
$0003  
Return address to FRET_2  
$0002  
Return address to FRET_2  
$0001
```

The stack will begin to unwind as we do the sequence of MULU and RTS instructions at FRET_2 as we first calculate 1!, then 2!, then 3!, then 4! and finally 5! which is the result we return to SuperBasic.

To run the above code, do this, or something like it:

```
1000 Start = ALCHP(128)  
1010 LBYTES win1_source_factorial_bin, Start  
1020 :  
1030 DEFine PROCedure Fact(n)  
1040   IF n < 1 OR n > 9 THEN  
1050     PRINT n; ' is slightly out of range, 0 to 9 only please.'  
1060   END IF  
1070   POKE_W Start + 6, n  
1080   CALL Start  
1090   PRINT n; '! = '; PEEK_L(Start + 2)  
1100 END DEFine Fact
```

Now, at the SuperBasic prompt, run the above to load the code, you only need to do this once, then just type Fact(n) where 'n' is a value between 1 and 9 as described above in the text. The results will be 'interesting' if you use values outside of this range.

Actually, in the interests of experiment, I tried it out. Using values above 9 is fine, up to a point, but zero will trash SuperBasic as the stack wanders down through memory and corrupts data that it should be anywhere near. Larger values will no doubt have the same effect, but anything over 9 gives incorrect results as the 16 bit MULU instruction isn't using the additional bits of the number. Anyone got a good 32 bit MULU and/or MULS routine they want to share?

I don't have the numeric skills to write one, and while there are plenty on the Web, they are, of course, someone else's work and subject to copyright etc.

⌈YOU ARE HERE - THE ABOVE IS TESTED OK⌋

The Fibonacci Series

The Fibonacci series looks like this:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55 ...

Apart from the first two numbers, each number in the series is the sum of the previous two numbers. This is written as

$$\text{Fibonacci}(N) = \text{Fibonacci}(N-1) + \text{Fibonacci}(N-2)$$

The very explanation cries out for recursion, you can see it in the statement above. We need to cater for the first two terms in the series and test for a Fibonacci(0) or Fibonacci(1) and return the value 1 for both of these values. The slight difference between Fibonacci and Factorial is that we need to recurse twice for each number, once for (N-1) and once for (N-2). This makes the code slightly more interesting and the stack too.

Here's how it looks in plain and simple SuperBasic:

```
1000 DEFine FuNction Fibonacci(n)
1010   IF n = 0 OR n = 1 THEN RETurn 1
1020   RETurn Fibonacci(n-1) + Fibonacci(n-2)
1030 END DEFine
```

So how difficult could it be to convert the above (two) lines of working code into assembly language? It all depends on how easily you get your head around the recursion, I had to sit and stare at the screen for a while until I finally came up with the following code:

```
Start      bra.s    Start2          ; Skip over result storage
Result     dc.l     0                ; Space for our result, address = Start + 2
FirstNumber dc.l     9                ; Assume Fibonacci 9 by default = Start + 6

Start2     lea      Result,a3        ; Where to shove the answer
           move.l   4(a3),d0         ; Fetch FirstNumber
           bsr.s    Fibonacci        ; Do it
FRET_1     move.l   d0,(a3)          ; Shove it !
           moveq    #0,d0            ; No errors
           rts                     ; Back to SuperBasic

Fibonacci  cmpi.l   #2,d0            ; Special cases 0 or 1 ?
           bcc.s    Fib_2            ; No, D0 is 2 or more. (Unsigned !)
           moveq    #1,d0            ; Return 1 for Fibonacci(0) or Fibonacci(1)
           rts                     ; That's our exit clause !
```

```

Fib_2      subq.l   #1,d0          ; Calculate N-1
           move.l   d0,-(a7)       ; Stack our 'N-1' value
           bsr.s    Fibonacci      ; Work out Fibonacci(N-1)
FRET_2     move.l   d0,-(a7)       ; Save the result of Fibonacci(N-1)
           move.l   4(a7),d0       ; Retrieve N-1
           subq.l   #1,d0          ; Calculate N-2
           bsr.s    Fibonacci      ; Work out Fibonacci(N-2)
FRET_3     add.l    (a7)+,D0        ; Add Fibonacci(N-1) to Fibonacci(N-2)
           adda.l   #4,a7          ; Tidy original N-1 off of stack
           rts                    ; And return

```

To run the above code, do this, or something like it:

```

1000 Start = ALCHP(128)
1010 LBYTES win1_source_fibonacci_bin, Start
1020 :
1030 DEFINE PROCEDURE Fib(n)
1040   IF n < 0 THEN
1050     PRINT n; ' is slightly out of range, 0 to 9 only please.'
1060   END IF
1070   POKE_L Start + 6, n
1080   CALL Start
1090   PRINT n; '!' = '; PEEK_L(Start + 2)
1100 END DEFINE Fib

```

This time we can use numbers larger than 9 as we are adding 32 bit values in the code, not multiplying. Of course, you can still pick a number big enough to trash the stack. Interestingly enough, Fib (30) executes in 1 second on my QPC setup, but the original SuperBasic version ran and ran and ran I CTRL-SPACE'd it after a while.

As an exercise, try to work out what the stack looks like for different values of N - it's an interesting lesson in mind numbing loops. Once you figure it out though, it gets easier.

When you are writing recursive code like the two examples above, you must remember two golden rules:

you must always have a 'get out' clause to stop recursion; and
never ever try to use other registers as storage - it just doesn't work !

In the above, we just stacked our working values and this is fine, but in other code, you might need to have a lot more values to stack, so how best to do this? The answer is quite simple, use the LINK and UNLK instructions which are designed to build stack frames that you can access using Address Register Indirect With Displacement - for example 4(a5) - addressing mode instructions.

Out of interest, has anyone spotted a potential problem with the above code?

The calculation of Fib(N-2) duplicates most of the work done by Fib(N-1). One solution to this 'problem' is to have an array of values in memory and when calculating a new value, store it in the table if it has not been stored already, if it has been stored already, simply extract it from the table.

The last two paragraphs should have given you an inkling of some 'homework' - which will not be marked - feel free to try out the implied exercises for yourself. The only problem with the array of values is that you never know how big to make the table and you need some method of determining if the table has been initialised (to all zeros) GWASL doesn't fill buffers with zeros, just with assorted random characters, unlike an array in SuperBasic.

The array could be set up as follows:

```
Answers    dc.l    0          ; Fib(0), zero indicates uninitialised answer table
           ds.l    1000       ; Fib(1) through Fib(1000)
```

Because GWASL won't initialise the entries for 2 to 1000 you have to do it yourself, as follows:

```
Init       lea      Answers,a3    ; Start of answer array
           cmpi.l   #0,(a3)       ; Has table been initialised yet ?
           bne.s    Done          ; Yes, exit
           move.w   1000,d0       ; 1000 = 1001 entries
I_Loop     clr.l    (a3)          ; Zeroise an entry
           adda.l   #4,a3         ; Pointer to next entry
           dbra     d0,I_Loop     ; Do the rest
           lea      Answers,a3    ; Answer(0)
           move.l   #1,(a3)+      ; Initialised to Fib(0)
           move.l   #1,(a3)       ; An initialise Fib(1) as well.
```

Code like the above should be called at the start of the file so that the initialisation is only ever performed once per session. Making multiple calls to the Fibonacci code will only require the table be initialised once.

When the code has calculated Fib(N-1) then it can store the result in D0.L into the table. As N-1 is on the stack, it can be retrieved into a spare register - say D1 - and converted to an offset by shifting it right twice (LSL.L #2,D1) and using that as an offset into the answer table.

Now, when asked to calculate a value, check the offset into the table and if it is not zero, return that as your answer - no recursion and much faster. You'll have to remember to limit the number of allowable values if using a table - you could end up corrupting some random bits of memory and the amount of space you need to ALCHP will go up as a result of the table - check it after assembly to see how big it is.

Have fun and if you feel brave, Dilwyn wrote a SuperBasic version of 'The Towers Of Hanoi' some time back, why not convert that to Assembly language :o)

See you next time.

A Sense of History

Geoff Wicks

Is the QL just a load of old rubbish? Perhaps not today, but what about in 10 to 20 years time? Will all that remains of the QL be a scavenger on a landfill site, wondering what that odd bit of black plastic used to do? Almost from the day of its launch the QL was consigned, with some justification, to the rubbish dump by most computer specialists. The QL Sinclair gave us was a shoddy hybrid unsuitable for either business or home use. It was not Sir Clive but the peripherals indus-

try that turned the QL into an efficient computer for both business and personal use.

Yet for a computer that has had such a troubled birth and an equally troubled history, it has proved to be remarkably resilient. In spite of scare after scare about its imminent death the QL has survived for over 20 years. Some dot com companies have not lived for that number of weeks!

Initially my attitude to the QL was ambivalent. I bought my first machine when it became

affordable after the price had been halved. Within three months I was regretting my investment, because the price had halved again. Nevertheless, I was stuck with it, and when I had kitted it out with extra memory, disk drives and a decent monitor I was able to do serious word processing. Paradoxically the commercial failure of the QL became an advantage. Backing up a system with a spare was financially possible, and peripherals could be bought one at a time as finances permitted. Little did I realise in those early days that almost 20 years later I would still be using a QL, although

now in the form of PC software. How important is the history of the QL to us? It has played a fairly important part in almost a third of my life. After 20 years of development in which many users have invested huge amounts of energy, would we really be happy to see all the QLs lying forgotten in some landfill site? Or has the time come when we should be setting up a QL archive telling the story of our machine?

Dilwyn Jones has often suggested there should be a "History of the QL" mini exhibition at major events such as last year's QL2004 and this year's QL is 21 show. As the programme organiser for QL is 21 I would like to take up this suggestion, but is there anyone who would be prepared to do the work?

Perhaps we should be more ambitious and consider a permanent QL archive. Does anyone know of a museum interested in building up a collection of computer history? If so, how we would like the QL to be represented? How could we contribute to the collection?

The QL cannot claim the same fame in computing history as the Spectrum and ZX81, but are we saying that nothing about the QL is worth saving? When I attended the ORSAM 2004 show in Norwich, I realised for the first time that the QL has something that sets it apart from other retro-computers. We QL users have continued to develop our machine, both in hardware and in the operating system, whereas the other machines have largely stood still. This is something of which we should be proud.

A related project, which periodically arises, is a QL software archive, although it is not always clear exactly what people envisage by this. I suspect

even the protagonists do not agree among themselves what they mean by a software archive.

If the task were to be done thoroughly, it would be a formidable undertaking. I have spoken to many people with experience of chasing up old software with a view to preservation and improvement, and all have suggested the idea of a complete software archive is a non-starter. In practice they encountered too many problems and did not make the progress they had expected.

Even the initial task of compiling a list of all commercial and public domain programs is formidable, and would take months of research. Then the logistical problems of tracing and contacting the authors, ex traders and copyright holders would be immense.

To give some idea of the problems, there is in existence a large database of current and former QL users, but this is not subdivided into categories. To extract the addresses of ex-traders and authors you would first need a list of their names. Even then the contact details would be up 15 years out of date. Recently some of us have been scanning the UK electoral register in attempt to trace such people, and have had only limited success.

Even if we were successful in tracing the people, we would often find a grey area of copyright between author and trader. Digital Precision have generously allowed some of their programs to be freely distributed and developed, but there have been restrictions on the disclosure of the code. Digital Precision was a well run firm, but even with their software there are uncertainties over the copyright to all or parts of their programs.

Subscribers to the QL-users email group will know this has caused complications in the release of Perfection, and confusion over which parts of the package can be freely distributed and which not.

Yet a further complication is that the source code for many programs has been lost. If my memory is correct Psion were prepared in principle to release the code of their QL programs, but could not do it in practice as the machines containing the code were either obsolete or inaccessible. For other reasons we are unlikely to gain access, even if someone were prepared to buy the rights, to the code of Text87 and QLiberator. In short, were we to attempt to set up a software archive, we may have to accept there would be severe shortcomings in the collection. However, perhaps we should think about the programs that deserve a place in the official history of the QL. I started to write this article about 6 months ago, but other matters intervened and it remained unfinished. When I picked it up again to complete it, I discovered an unexpected bonus. During those 6 months the mood within the QL community has changed.

In the first half of 2004 many QL users were in a pessimistic mood and believed the QL was slowly dying by attrition. At the end of the year - I like to think partly as a result of QL2004 - there was a new enthusiasm. Two major software projects, QWord and QDT, were at or near release; the printer problem was being actively tackled; and the QL-users email group was bursting with ideas and controversy. People are again looking into the future, but should that prevent us from respecting and maintaining our past?



JUST WORDS!

A little behind...

No, this is not a reference to a corny joke about sitting on bacon slicers, but an apology because we are behind schedule in updating our software to GD2.

The good news is that there is now a GD2 version of PIN-DOWN, and you can download a demo version from our webpage.

The bad news is that you will have to wait a little longer for GD2 versions of QL-RHYMES and AUTO-GRAPH.

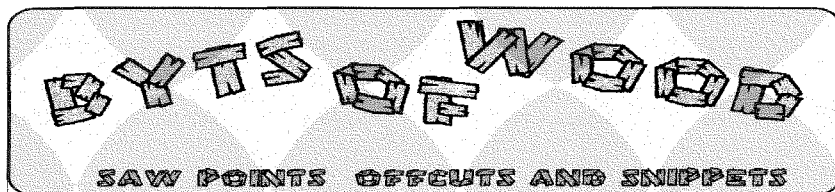
We have other plans for 2005, including a revamped webpage with more download possibilities.

Geoff Wicks, 56 Peveril Crescent, West Hallam, Derbyshire DE7 6ND, U.K.

Tel: +44 (0)115 - 930 3713

email: gwicks@beeb.net

Web: <http://members.lycos.co.uk/geoffwicks/justwords.htm>



Here we are in 2005 and I have been beta testing one new piece of software and one update for release this year. Not only that but I am also expecting to take delivery of QWord - another one of those addictive games that will stop me from getting this column finished on time. Things are looking up.

Start of a New Year -

Back to an Old Complaint

I have a lot of sympathy for David McCann's problems expressed in the last issue. It is something I have raised in this column a few times before. The root of the problem is that we, as a community of QDOS/SMSQ Users, have decided

that there are, and will be, no new entrants to the club. Anyone who cannot write a few lines of BASIC, configure a program or compose a workable boot file is on the outside. They may be looking in in envy but they are also invisible to us. No one expects someone who has an interest in computing to be lured from the shiny world of the PC or the Open Source Technoverse of LINUX to suddenly take up our system as a hobby but what we all miss are the large number of people who still use a black box QL with a Gold or Super Gold Card and have done this for the last 15 years. When they want to make the jump to a Q60 or QPC2 they are under-suppor-

ted by the documentation and, to an extent, by the people who provide the hardware and software.

I will admit that there are obstacles in the assimilation of these people into a more technically challenging system and there is often a resistance from them to being guided. It is often very frustrating dealing with people who ask the same question over and over again - especially when the answer to the question is in the manual provided and you have done your duty in pointing them at that answer on the first occasion. This said, we need these people to get up to speed with the rest of us and to appreciate the mountain of work done by those who do support the system - often on a free or extremely cheap basis. To this end I am proposing a 'Start Here' section for this magazine

and I hope to be able to contribute at least the first two articles in it for the next two issues. Most of the information given here will be simple and basic but it may well help people like David McCann, who have not followed the progress the system has made over the last 10 years. Many of you reading this will think this information is not needed and there is an argument that many of the 'old school' QL users do not buy this magazine anyway because they feel it is 'too technical' but that is something we should strive to change. We should be welcoming the less technically apt members and helping them to understand because that understanding will make them continue to use the system and may well make them more a part of the community.

You Can Help!!

This may well be a pointless request, given the level of response we get from these things, but we would appreciate some assistance in making this series of articles comprehensive and informative.

1. We need to know which areas people find the most baffling. In the first section I want to cover Screen Resolution and, in the second, Hard Drives. The third and fourth will cover BOOT files and Extensions but what else do people want to know and what areas cause them the biggest problems when moving from a long black box to a Q60, QPC2 etc.

2. We need some people to write some articles. If you can provide a section on a topic which might help someone get a bit more from his system get in touch with us and volunteer. It could be the most basic

piece of writing but it could be that 'Eureka' moment for someone who has struggled with a new O/S or piece of hardware.

Most of my writing will be based on SMSQ/E but we would welcome contributions from emulator users about non-SMSQ/E subjects.

As I said, response to these sort of requests has, in the past, not been great, but we really need to be able to help those with no technical support and this would seem to be a good way to do it.

If you are one of these people let us know what you want to find out how to do or what you find hard to use. You could also come along to a workshop and ask us. This is, after all, what the workshops were originally for and what they should be about.

Back to the Problems

I would like to address some of David McCann's problems here and now.

I would agree that the documentation for SMSQ/E on all systems is somewhat lacking in 'first time user' friendliness. One of the biggest problems people have is dealing with both the screen resolution and the hard drive settings. This is why I hope to be able to cover these in the first two sections of 'Start Here'. I believe that improvements were made to the partitioning of SMSQ/E which do allow larger partitions to be stable. Larger memory allocation in QPC2 and other systems using SMSQ/E is now available. Most of this was, I think, because of the slave block problem which was resolved by Marcel Kilgus. The manuals should reflect these changes.

I cannot really comment too much on LINUX and I am not

sure that this magazine is the place to really look deeply into it but I would like to comment that any expectation that LINUX will offer scanning on a Q60 is probably a little optimistic. The problem here is a hardware one in that most scanners of a reasonable price are USB and, as far as I am aware, there is no way to connect a USB device to a Q60. I can see no real reason why he should not have Internet access but, on the other hand, the Internet has become very complex with many large sites using Flash animation and all sorts of tricks to make the content more exciting. Older browsers, without the extensions needed for these things, would find it hard to cope with some of these sites and would fall over completely when faced with some of the 'secure' sites. Much of the LINUX documentation I have read is even more impatient with users who are not of the required level of education in these things and is very 'jargon intensive'. I can see why he would have problems there. With regard to the mouse problem, Q60 SMSQ/E will look for a mouse on the highest serial port available first and then next the next until it gets to SER1. If it does find one on any port it will initialise it.

I will address the screen resolution issues separately but I am not sure what there is to unimpressed with in SMSQ/E apart from the examples he listed. I agree with both of these, in fact. Minerva did have some very neat features in it and I have long suggested that they be added to SMSQ/E - especially now that Minerva code is available to all.

SMSQ/E is modular so adding a 'Compose Characters' module should be easy and even adding an extra keyword to

SBASIC should not be beyond us. It is in our interest, now that Minerva is no longer supported, to merge these two O/S's and make them as compatible as possible. It is also in our interest to have a single platform to run programs on and a single, managed, development path for that platform.

The difference between the hard drive formats is also something I would like to see eradicated. This is hard to do because there are many Qubide users and their hard drives are set up in the older 'Rebel' format. The QXLWIN files, introduced for the QXL seem to be the best way of doing things and Nasta is looking into using these for the Qubide replacement - the SuperIDE. Mark Swift was given the rights to use the Qubide format for his QDOS Classic system when SMSQ/E and its code were still not open source. Maybe he would consider a version of QDOS Classic that uses QXLWIN files.

I had problems with ProWesS running on a Q40 and this was one of my many arguments at the time when QBranch was selling them. I was told that this had now been resolved. ProWesS has long been an open source program and available to anyone who wants to make changes.

At the time of its release it was a massive undertaking to produce a fully vectored system for the QL but I have always found it rather slow - even on a fast system. I am sure that, given some work, it could be speeded up and optimised. It would then be finally recognised as the the innovation it should have been. I believe the problems with ProWesS on the Q40 and Q60 lie with the cache controls which were introduced to the Q40/Q60

SMSQ/E to speed it up. ProWesS does run very well on my Q40 here but then I turn the caches off. Maybe someone would like to write something on that.

QPAC 1 and QPAC 2 both have some vital and very useful utilities in them and can prove to be, in my view, essential to any any system. I wrote a full explanation on using QPAC 2 in a previous issue of QL Today and that is available from me if anyone should want it. I used to provide it, along with a disk with a simple boot file which would set up a basic QPAC2 system with very copy of QPAC2 I sold and that seemed much more helpful than the manual and tutorial supplied. I can, again, provide this to anyone who would want one (call or email me).

One last word on SMSQ/E versus QDOS in the area of BASIC. SMSQ/E is much more strict in the way it enforces the rules of SuperBASIC. The older QDOS would let a lot of violations through and these loopholes were either used by programmers as short cuts or were not fully understood and never picked up because the programs worked. Under SMSQ/E, if any programs break these rules it will stop them working and issue an error. Many people have criticised SMSQ/E for this but It does mean we have a tighter, more predictable, system.

I Saw It On The Radio

Strangely enough, the one point on which I would take issue with Mr. McCann is a point I made quite strongly in a letter to the Quanta magazine many years ago. At the time I expressed no desire to own a PC. I made this argument on the grounds that the QL was cap-

able, in itself, of achieving most of what a PC could do in a more efficient way. Although that is still, in some areas true, the PC continued to evolve when we stood still and it had a vast weight of money and power behind it so the statement no longer holds water. I see no reason now why I should not own a PC (In fact I have two) and you all know the high esteem I have for QPC2.

In short, if you want Internet Access that is more than just text based, scanning, digital photography and a whole host of modern facilities you simply have no choice. You have to have either PC or MAC hardware. Whether you run Windows, LINUX or MAC software on it is up to you. What he is saying is I have had a radio for many years and I like it but now I want to watch TV and I don't want to buy a TV - I want to buy a better radio that shows moving pictures.

Of course, once you have the hardware you need to have QPC2 to run on it and then you have the best of both worlds.

What Do You Want?

Roy Brereton sent me copy of a letter to Quanta in which the user was saying "Pointer environments, endless colours all leave me cold". Roy was saying this is directly related to many of the things I have written in this column. He broached the question 'are the improvements in the system just there for the movers are shakers of the QL world and do they have no relation to the grass roots users?' in a way this echoes a comment from David McCann's letter 'I suspect that, for most of them, SMSQ/E is a sort of a game; after using it they go back to Windows'. While I can respect those people for

whom the more modern interfaces and extensions to the QL system are just pointless eye candy I cannot help but wonder what these people would like the Quanta magazine to be writing about. If you do not discuss the development of the system what else is there?

On the older systems, where memory was tight and clock cycles are of the penny farthing variety I can understand why you might want not to have the system slowed down by any un-needed code but, even on a Gold Card, the overhead from the Pointer Environment is slight and by the time we got to running QPC2 on 3GHz PCs and systems like the Q60 why not make use of modern short-cuts? It has to be faster and more accurate to copy files using QPAC2 than by typing them in on the command line, especially when you are copying some but not all of the files from one destination to another. It also has to be easier to use Fileinfo 2 than to open a program and tell it to load a file. And so we come full circle to the QL luddite community.

Beyond the Fridge

I had a spate of corrupted or unreadable disks both sent to me recently and, it would seem sent out by me. I could not work out why this was happening all of a sudden. I then received a very attractive gift from British Telecom. A sheet of words, printed on a laminated plastic base and designed to be broken apart and used as fridge magnets! Now I know that not many people use floppy disks and audio cassettes any more but there are a lot of video tapes being used and having all these things sitting in your postie's sack

together with a clutch of fridge magnets does a lot of damage. Quite how you stop this kind of irresponsible behaviour I don't know.

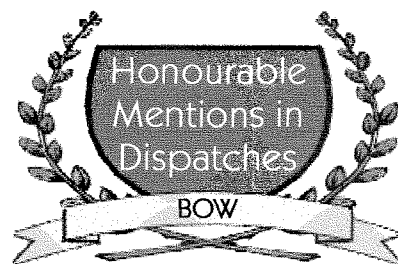
GAlling

I suppose I should not end this column without having another whinge about things in the post. There are two things I would like to see improve in the new year so I am urging you to partake of a couple more New Year's Resolutions. I am doing this with my QBranch hat on.

When we took over the service of upgrading Qubide ROMs from Qubbesoft I tried to emphasise to those who bought the upgrades that we were just charging for the service, the floppy disk with the new code on it and the upgrade sheets. We did expect the grateful users to return the old GALS and EPROMS for reprogramming. We could have asked you to send them in before we did the work but that would have deprived you of your machines for a while. Many people did return theirs but a significant number have not and we are now running out of chips. If you have these lying around in your home please send them back to me so others can benefit.

The second whinge is about disks. A floppy disk is made of plastic. I can break, bend or be otherwise destroyed in the post. It is, therefore not a good idea to stick it in a little brown envelope and put it in the post. A padded envelope and a bit of cardboard will help it survive the journey better. I would also like to point out that, whilst some upgrades are free, postage is not. If I have to pay the postage, replace the disk and provide a proper means of getting it back to the user whilst

not charging I am losing out. OK that is my whinging over and done with - for this issue at least.



Not Just a QWord

I wrote most of this article over the Christmas period between reading and replying to the many emails that filled the QL Users list. Amongst these there were many people replying to the questions I posed about screen resolution and a few other referring to how to find a way to detect if the system a program is being run on has the high colour drivers. This is by no means easy because, in order to ensure that programs written for the high colour drivers do not just crash on systems that do not have it, the KEYWORDS which are used to access the colour components have been added to the systems that do not have the actual drivers. While this is, in many ways, as good thing for compatibility it is a complete pain when the programmer has to access the drivers and so has to test to see if they are there. Wolfgang Lernerz and Marcel Kilgus are looking into ways of resolving this problem. Interesting, though, that every problem we throw up makes us take steps towards a more complete system.

The reason for this problem being so far to the fore recently is the release of QWord. This has to be the most advanced looking piece of software I have ever seen running on a QL system and I can promote it

in this column with a clear conscience because it is not something we will be selling at QBranch. Some of you may snort with derision at the concept of playing a game on a QDOS/SMSQ system but this is worth buying just to marvel at the way it looks. I can also say that it is very addictive. If my copy had arrived before I started to write this the column would be very late. An excellent piece of work all round from Geoff Wicks, and Rich Mellor with notable contributions from Phoebus Dokus who provided a lot of the graphics and sound. I believe that Marcel deserves a lot of thanks for testing (along with Peter Graf) and making some special changes to QPC2 too. The program uses the QLSSS sound system by Mark Swift and Simon Goodwin and extra sound based extensions by Wolfgang Lernerz. George Gwilt was also involved in changing some of the Turbo routines. So that is the first Honourable Mention of 2005 and it takes in most of the active community!

I'm Home Dear

Two other things that were reverberating around the user group in the weeks before I put this column to bed were a new concept for QDOS/ SMSMQ a return to another, older, complaint. The new concept, which has long been a standard thing for many other, more mainstream, systems is the idea of a 'Home' directory. This would mean that a program would 'know' where it was executing from and, therefore, not need to be told where it's ancillary files were. This would be a good thing for programmers and users alike because many of the problems people have when new programs come out

are because they have no idea how to configure them to use any extensions or data files that they may need. This is a good idea that does need some development and it seems to be getting that attention.

The old complaint (and indeed request) was, of course, that the old QDOS file name length restriction should be changed and a better directory system introduced. When this was aired some years ago in Quanta I was very much against the idea of changing the system because so many of the older programs would not be able to cope. Over the years I have changed my mind somewhat and I do feel that, by making the system overly compliant to older software, we hold ourselves back. Again this is that argument about the old guard versus the new upstarts but developments in the way the QL systems operate in the last few years have shown that radical ideas can be incorporated without losing compatibility with older systems. I am sure that this change can, if approached with the right degree of thought, allow older programs to run as normal and newer ones to take advantage of the extended file system. I fully expect that I will be writing more about this in my next column.

Casting a Spell

I would like to make a request for a program to be looked at in the coming year. I use QTYP spelling checker a lot - not because I cannot spell but because my fingers can't. Most of the time I come up with the industry standard typos - 'teh' for 'the' and 'taht' for 'that' so, when I finish writing I do a search and replace on these

words before running the spell checker. This mops up a lot of the errors. I do, however, come up with the odd word where my brain cannot work out what I have done wrong and then I resort to Geoff Wicks' excellent 'Spell Crib'. The standard QTYP window comes up with two options when it finds a word it does not recognise. You can either ignore it or add it to the temporary word list. I would like a further option, and that would be to call Spell Crib and allow me to find the word there. How about it Geoff?

A Good Start - Keep it up!!

This has definitely been a promising start to the new year with this and QDT debuting. There are some other good software projects on the go too but I was, at the time of writing this, sworn to secrecy on one of these projects and I have not, at this moment, had the test copy that Marcel said he was sending to me so full comments on that may have to wait for next issue. Marcel has, just as I finished this, raised the veil somewhat and announced that he is close to completing a printer device for QPC2 which will allow users to print to any printer attached to PC.

This is a solution many of us have been waiting for a long time to see. It will, unfortunately, not solve any problems for standard QLs, Auroras or Q.xx users because it does need the manufacturers printer device drivers to be running on the underlying system.

I have yet to see the results but, if they are as good as Jochen says, the next Honourable mention has been reserved and is well deserved. Intrigued? Just you wait!

Hope your New Year has started well.



The QL Show Agenda



Regular QL Meeting - (NL) Eindhoven

Saturday, 26th of March, 10:00 to 16:00

Pleincollege St. Joris, Roostenlaan 296

Jochen Merz Software will be at the show,
bringing all the new programs, updates and special offers along!

Hove Workshop - (UK)

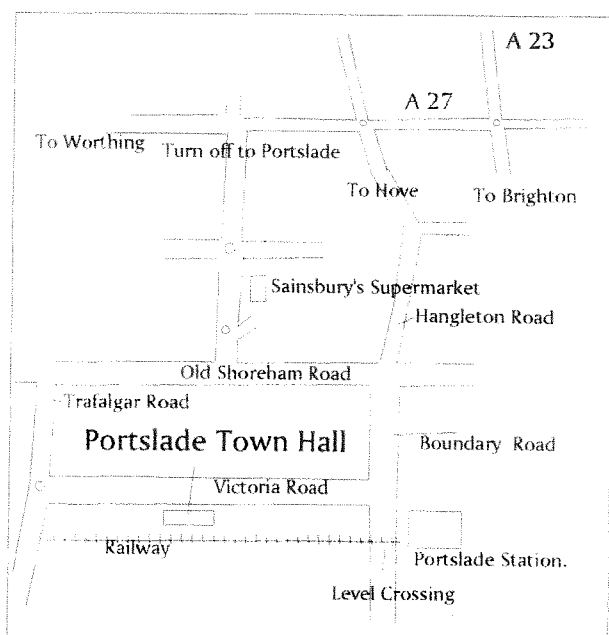
Quanta AGM and Workshop

Portslade Town Hall

Hove, Sussex

April 17th 2005

This years Hove Workshop will be held at, was has become our usual venue, Portslade Town Hall, just a few minutes walk from the Portslade Railway Station. It will also be the Quanta AGM so make a note in your diaries now. It is too early to say who will be attending the show but we expect a good turn out. If the weather is fine you can even take in a day by the sea the day before. Catering, as usual, will be by a bevy of lovely local ladies and Roy Woods hyperactive daughter.



International QL Meeting - (NL) Eindhoven

Saturday, 18th of June, 10:00 to 16:00

Pleincollege St. Joris, Roostenlaan 296

We hope to turn this QL show into another major event!

All major QL dealers will be there!

Reserve this date in your agenda and come to the show -
we would really like to repeat the success of QL 2004!