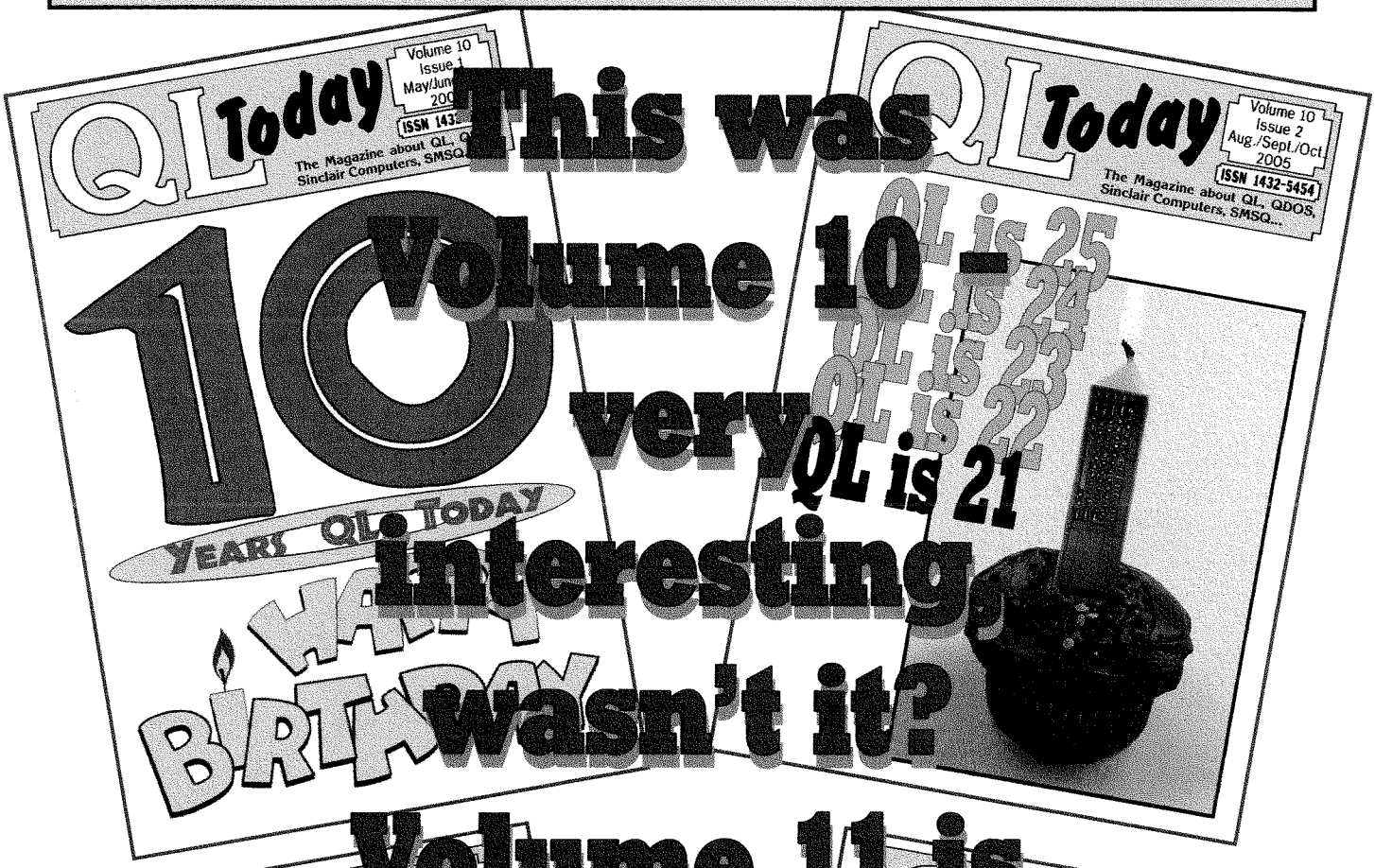
QL Today

Volume 10
Issue 5
April-June
2006

ISSN 1432-5454

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



This was Volume 10 - very interesting wasn't it?

Volume 11 is coming up please make sure you won't miss it!

www.QLToday.com

Contents

Advertisers

in alphabetical order

QL Today

ISSN 1432-5454

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

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

Editor:
Geoff Wicks
56 Peveril Crescent
West Hallam
Derbyshire DE7 6ND
United Kingdom
Tel. +44 115 9303713
email: gwicks@beeb.net
email: QLToday@j-m-s.com

Co-Editor:
Bruce Nicholls
38 Derham Gardens
Upminster
Essex RM14 3HA
United Kingdom
Tel +44 20 71930539
Fax +44 870 0568755
email: qltoday@q-v-d.demon.co.uk
email: QLToday@j-m-s.com

QL *Today* is published five times a year, our volume begins on beginning of June. Please contact the German or English office for current subscription rates or visit our homepage **www.QLTODAY.com**.

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.

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.

The deadline for the next issue is the 15th of July 2006

Editorial

by Geoff Wicks

This year we didn't get a mention, which, I must confess, disappointed me. But last year, in his annual report to members, Quanta's chairman referred to QL Today not by name, but rather quaintly as "another QL related newsletter".

Do I detect a nuance here? Quanta members are blessed with a "proper magazine", but QL Today readers have to make do with a mere "newsletter".

So let's take a look at our achievements. In 2005 we produced 294 A4 pages of QL news, shows information, adverts and articles. Just over 11% was advertising and most of the rest, about 165,000 words, editorial content. We produced 18 pages of QL news and 6 pages of shows information. We published articles from 21 different authors and letters from another 4 contributors. Finally there was the popular QL documentation insider CD.

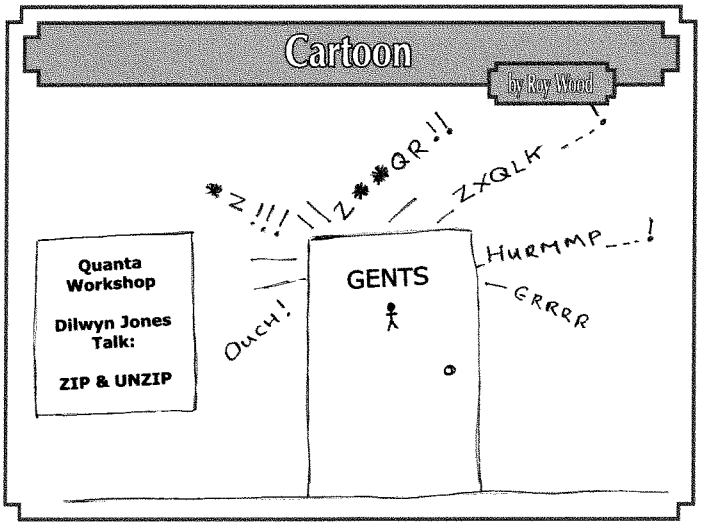
Not a bad record for a mere newsletter!

One of the reasons QL Today is a successful publication is because we are an enthusiastic team who believe in what we are doing. Jochen and I may be the visible front men, but we are dependant on two others. Co-Editor is Bruce Nicholls who proof-reads every page of every issue, even though our publishing schedule gives him only a few days to do so. He also maintains the QL Today website. Roy Wood runs the English Office and does much of the work in sending out the magazine and collecting subscriptions. He also has his ear close to the ground and feeds back much reader opinion.

There was little work for me as editor in the production of the documentation CD. The hard work was done by Dilwyn who compiled it and Jochen and Roy who ensured there was a copy for each reader.

We are also lucky to have a group of faithful and expert writers who contribute regularly. In the last few months we have received a good volume and variety of copy, but we are now approaching the summer months when things go quiet. We would welcome your contribution. But what of Quanta's "proper magazine". Quanta can proudly boast that it is the only QL publication that has survived as long as the QL, but for almost ten years now it has been struggling, and has become a poor shadow of what it should be. In the last few months the Quanta committee have put a lot of effort into restoring its reliability and readability, and are having some measure of success. We at QL Today wish them well. A strong Quanta Magazine would be good not only for Quanta but also for the whole QL community.

And, believe it or not, also for QL Today.



QPC2 to go 68020?

At the recent Quanta show in Manchester, George Gwilt made a surprise announcement that he was working with Marcel Kilgus to add 68020 instructions to QPC2. The Q60/Q40, QXL and Gold Cards use a 68020 processor, but until now QPC2 emulates the 68000.

So far George has written only a subset of 68020 instructions. In his own words:

"For various reasons only a subset of 68020+ instructions have been coded so far for QPC2.
The new instructions are:
The 8 bit field instructions (eg BFTST)
PACK
UNPK
CMP2
RTD MOVE from CCR

Enhanced instructions are:
DIVU/S long form
MULU/S long form
LINK long form
CHK long form
EXTB.L byte to long word
CMPI more ‹ea›s TST more ‹ea›s

Instructions not emulated include
MOVE16 (68040) and TRAPcc."

Marcel is now adding more instructions:

"I didn't implement MOVE16 because it's in the LINE-F segment and I don't want to break it up for just one command. But CAS, CAS2, TRAPcc, CHK2 are all implemented now, along with new exception format and pretty much everything the 68020 supports, except MMU and FPU commands (which are basically optional on a 68020, too)."

He also gave an indication of the quality control that remains to be done. Writing about the trial version of QPC2 that George demonstrated at Manchester, he comments:

"That worked fine from the get go. But still I've spend the last few days almost full-time getting everything to release quality and certainly have to spend more time on it (reading the documentation, reviewing and understanding every single line of code, extending the test-suites, constant regression tests and comparisons to a real QXL, solving the found problems, doing some refactoring of decade old code along the way, adding new features, software compatibility tests, working on the manual... there is a lot of work involved in my commercial releases!).
So all I'm saying is, I guess all this will be part of the next version, but I'm always cautious with promises. An extensive beta phase is certainly needed first. My usual testers will get it anyway, if any more want to be part of it, drop me a line. And when it does come out, people can thank George, without him it certainly wouldn't have happened!"

SMSQ-E v3.12

Wolfgang Lenerz has announced the release of SMSQ-E v.3.12. He writes:

"What's new in this version:
Background drawing: Even when a window is partially covered, printing into this window continues. This goes with two new Basic commands: PE_BGON to turn this feature on, PE_BGOFF to turn this feature off. By default, this feature is TURNED OFF, so use the PE_BGON command in your boot file if you want to keep it on.
Basic has a new command line history: if you press the up/down arrows, you get the latest commands entered.
New Basic command JOBID, by Per Witte: result = JOBID [(lnr, tag) | ‹name›)] named job id or my id (no parameters)
Thierry Godefroy improved Qx0 cache handling.
Jochen Merz donated the source code for the "uti" and "util_menus" libraries to the SMSQE source code. Hence, the libraries are no longer used, we use the direct files instead.
New commands POKE_F, POKES_F, PEEK_F, PEEKS_F for floating point poking/peeking by Marcel Kilgus."

More Marcel

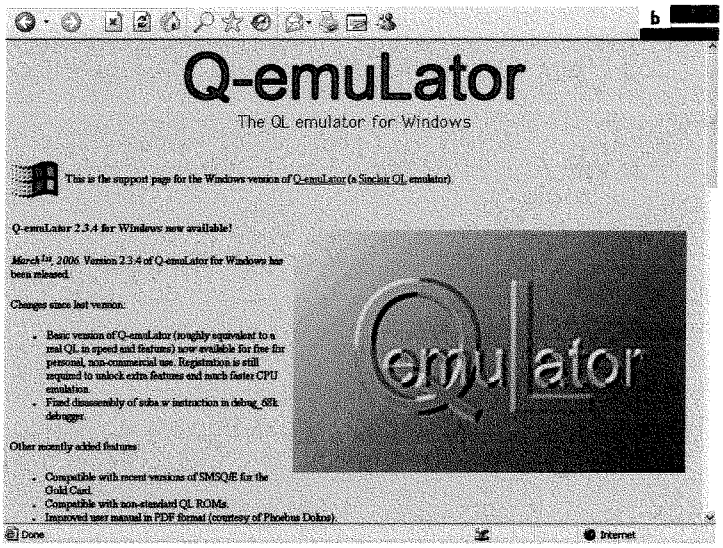
Marcel Kilgus has posted some new files on his website:

"I've finally managed to put some bits and pieces online. Mainly there are 3 new SBasic toolkits published:
- One that is mostly a wrapper around SMSQ/E I/O traps (people looking for a INPUT replacement could do worse than to check this out!)
- One that includes routines to convert to/from the HSV colour space (the routines that power Wolfgang Uhlig's QCP colour picker)
- One that can encode/decode base64 encoded data.

Also the version histories of both QPAC2 and EasyPtr are now online."
He has also added a new version of Qascade together with the source code.
http://www.kilgus.net

Q-Emulator Update

Daniele Terdina has released a new version of Q-Emulator:
"Q-emuLator version 2.3.4 for Windows has been released and can be downloaded at
http://users.infoconex.com/~daniele/winql.html
The basic version is now available for FREE. It emulates the features of an unexpanded QL (but can also access QL floppy disks and the Windows file system) and its speed is limited to be roughly equivalent to that of a real QL. Registration is still required to access all of the extra emulator features and for much faster emulation speed.
QemuFast (a prototype of a version of Q-emuLator that runs four times faster) is also available again, for registered users only."
In a subsequent emailing Dilwyn Jones gave more details about the features of the program:
"I downloaded and installed this version. What you get is basically a 128K QL with QemuLator's known levels of compatibility and ability to use any QDOS or Minerva ROM, and the possibility of using non-standard 48K ROMs and 16K ROM images like Toolkit 2. The earlier trial version only gave you 30% of the speed of an unexpanded 68008 QL, but I think it gave you the option of a small memory expansion. This trial version does not allow ramdisks, parallel port, TCP/IP access, level 2 file system or QXL.WIN file access, or expanded memory, so it is well worth stumping up for the expanded version, but this free trial version does run at the same speed as an original QL with 128K of memory and supports up to 8 drives (which can be floppy drives or PC hard disk directories) and you can use your preferred ROM version (it comes with a version of Minerva). This unregistered 128K version is quite OK for testing software on the emulator, or for running 128K software like games or Superbasic programs.
If you opt to register for the expanded version, you get the missing features and more. Examples of what you'll get with the expanded registered version:
More RAM available TCP/IP access (use Jonathan Hudson's Lynx, QL-FTP and email programs) Level 2 filing system (i.e. directories) Read and (new to this version) write to QXL.WIN, like QPC2 or QXL. Ramdisks Parallel printer port access Much faster than the unregistered version (depends to some extent on the speed of your PC). Ability to run Gold Card SMSQ/E
And a little bonus: once registered you have access to QemuFast, a much faster version, although not quite as compatible with some QL software as the standard QemuLator (bearing in mind that a registered QemuLator is already much faster than most people assume).
I guess QemuLator has lived in the shadow of QPC2 to some extent over the years, which is a pity. I can see that QPC2 is good for those like me who write programs to use the latest facilities, whereas QemuLator might be a good choice for those who just want to write the occasional SuperBasic program and run their existing QL software without being too bothered about high colour and such things (e.g. people who return to the QL after a period of absence)."

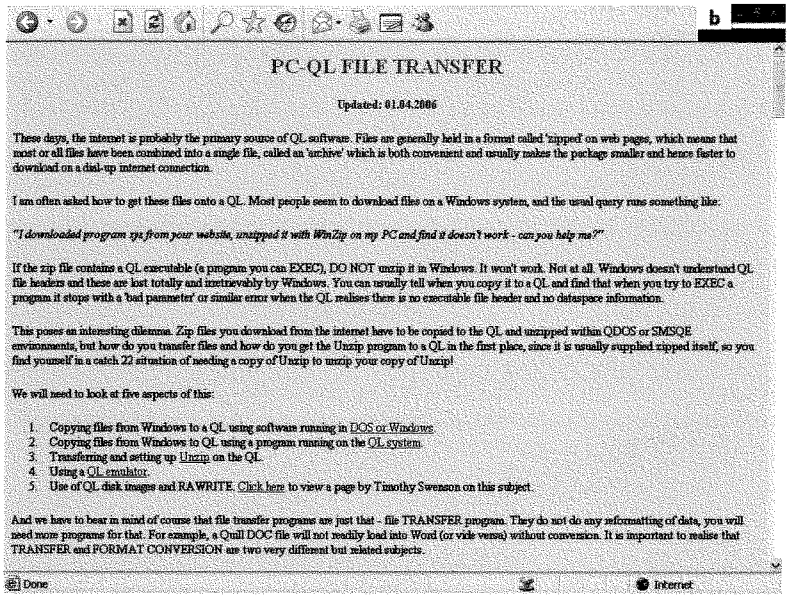


UNZIP Solutions

Much QL software can now be downloaded from the internet, but for the newcomer there are many pitfalls, the most serious being that, for various technical reasons, it is impossible to use the PC unzip program to expand QL software. Unzipping is especially difficult for users whose QL system cannot read and wrote to PC formatted disks.
Dilwyn Jones has set up an information page on his website to help first time unzippers:
http://www.dilwyn.uk6.net/gen/pcqlxfer/index.html

One possibility is the use of a DOS program RAWREAD.EXE and this program can be down-

loaded from Dilwyn's site. On the download page there is help documentation by Tim Swenson.
http://www.dilwyn.uk6.net/gen/pcqlxfer/raw.html


PC-QL FILE TRANSFER

Updated: 01.04.2006

These days, the internet is probably the primary source of QL software. Files are generally held in a format called 'zipped' on web pages, which means that most or all files have been combined into a single file, called an 'archive' which is both convenient and usually makes the package smaller and hence faster to download on a dial-up internet connection.

I am often asked how to get these files onto a QL. Most people seem to download files on a Windows system, and the usual query runs something like:

"I downloaded program xyz from your website, unzipped it with WinZip on my PC and find it doesn't work - can you help me?"

If the zip file contains a QL executable (a program you can EXEC), DO NOT unzip it in Windows. It won't work. Not at all. Windows doesn't understand QL file headers and these are lost totally and irretrievably by Windows. You can usually tell when you copy it to a QL and find that when you try to EXEC a program it stops with a 'bad parameter' or similar error when the QL realises there is no executable file header and no dataspace information.

This poses an interesting dilemma. Zip files you download from the internet have to be copied to the QL and unzipped within QDOS or SMSQE environments, but how do you transfer files and how do you get the Unzip program to a QL in the first place, since it is usually supplied zipped itself, so you find yourself in a catch 22 situation of needing a copy of Unzip to unzip your copy of Unzip!

We will need to look at five aspects of this:

1. Copying files from Windows to a QL using software running in DOS or Windows
2. Copying files from Windows to QL using a program running on the QL system
3. Transferring and setting up Unzip on the QL
4. Using a QL emulator
5. Use of QL disk images and RAWRITE. Click here to view a page by Timothy Swenson on this subject.

And we have to bear in mind of course that file transfer programs are just that - file TRANSFER program. They do not do any reformatting of data, you will need more programs for that. For example, a Quill DOC file will not readily load into Word (or vide versa) without conversion. It is important to realise that TRANSFER and FORMAT CONVERSION are two very different but related subjects.

Done

Internet

Is COLDFIRE Vapourware?

In response to this question on the QL-users email list Arnoud Nazarian replied "I don't think so, but it seems to be a very long term project"
He then described contacts he had had with Nasta:
"In a few words:
I was there, in Croatia, in May 2004 and Summer 2005. I intend to go again this summer. My project is: I want Nasta to make a very low cost "QL". At this moment I am not really interested to see a "QL" based on the very latest fastest possible Coldfire, but this would be a beginning. And I am ready to invest more than 1000 Euro. And I believe other people would also invest some money in a well prepared project.
The fact is that Nasta is certainly our man to do the hardware, but right now he must live and he has already too much to do for his employer. Because he currently develops electronics not based on his knowledge of the QL and he regrets it. However his employer is also quite interested in a low cost "QL" for his own use, this is sure.
The second problem that he explained to me is not design of a board but industrialisation. As a matter of fact there is a mechanical problem: modern surface mounted chips require toolings and machines that is readily available in the far east, but there they are not interested by low production batches. So he wants to find a supplier interested and able to manufacture small batches in Europe. And even possibly in Croatia. Last summer this seemed to be his major problem even if he had an idea...
In the meantime Nasta had told me that we could try to find some sort of development board, pay it and lend it to Marcel to try to adapt SMSQ/E to the Coldfire (Coldfire emulator of the 68k). But IRC he did not go very far into this direction as a board made by Freescale (I believe) would be usable from a software point of view, if only the connectors were bigger to be able to adapt monitor, keyboard, serial connection etc.. Mechanical problem again."

New Shows Policy

The Quanta Committee has recently discussed the future of QL shows and has launched a new initiative to encourage attendance.
In principle the committee want to continue alternating the venue for the AGM between the north and south and also hold further workshops each year. They have a preference for two day workshops with the Saturday being mainly a trading day and Sunday a lecture/demonstration day. To stimulate the lecture programs they are prepared to finance bed and breakfast for 2 or 3 speakers.
At QL is 21 Quanta financed the attendance of a few people who had made significant contributions to the QL community and some of these people contributed to the lecture program. The policy was also implemented at the recent Manchester show.
Quanta is continuing its experiments to produce its magazine in downloadable form. One recent suggestion for reducing the file size is to replace the traders advertisements with links to an electronic copy and to their websites. Quanta recently updated its members guide with, as a sign of gratitude, a free advertisement to its regular advertisers.

In the Pipeline

Dilwyn Jones will shortly be launching a new QL CD containing a collection of royalty free images that can be used as wallpaper. These will be in a choice of screen resolutions from 512 x 256 to 1024 x 768.
Also supplied will be a conversion program to convert between Windows BMP files and QL screen formats, including _scr, _pic and _psa in 4,8,256 and 16 bit modes. The conversion will

work in either direction and it will also be possible to convert Page Designer 2 and 3 files.

George Gwilt is currently updating his SVSCR suite that was reviewed in the last QL-Today. He is making this both _pic and _psa compatible and is improving the onscreen information.

SUDOKU

Ian Pine has updated his Sudoku program that was published in QL-Today volume 10 issue 3. This now makes additional checks before entering the recursive phase and in his own words now eats the impossible puzzle mentioned in the last QL Today for breakfast.
The new version can be downloaded from the QL-Today website:
www.qltoday.com

Above Our Heads?

The QL-users email group recently received a message from outer space. We reproduce it in full and without further comment:
"Greetings Earthlings,
I'm playing around with Trolltech's QT4 on Windows and Linux at the moment, and I've taken my old 'baby' the Quill Stripper (aka WinStripper) and I'm rewriting it using QT - this will, hopefully, let me see how easy it is to use QT, how easy it is to write cross-platform applications and convert WinStripper to Linux.
I am toying with the options to convert a quill doc into the following :
- text format, as at present
- html format, complete with bold, underline, sub and superscript.
- DocBook XML format
- PDF format.
I'm actually quite excited about the last option as PDF files are 'taking over the world' and I actually have it working in a rudimetary fashion. Is this of any interest to anyone?
The project is still in the development stages, I'm still learning QT4 and I have a new compiler set up to use as well. I'm hoping to build the application using a 'plugin' format so that I can 'easily' add new WordProcessor formats as and when required. I'm experimenting with too many things all at once though :o)
Hopefully, I'll have plugins working, plus quill and Perfection plugins developed 'soon'.
As the version of QT4 I'm using is Open Source, I am obliged to supply the full source code to anyone who wants it. This is a pleasure to do and I've got no qualms whatsoever about doing it.
Watch this space. Cheers, Norman."

QSudoku

by Norman Dunbar

As I'm off on holiday (again!) this week, I'm not writing an article on assembler this time. Instead I'm rising to the challenge thrown down by Geoff in the last issue and I have written a Soduku generator.

I did have a little help from a couple of articles in PC Plus magazine and in the Christmas edition of New Scientist, however, all the code in the following program is my own.

I have to confess straight away, I have not met the exact challenge as defined by Geoff, I don't build a grid and then take away numbers etc. In this program, I start with a grid that is in the 'usual' Soduku form at (with zeros replacing blanks) and simply shuffle it about. Because of the way that this shuffling is done, we end up with a new board ready to be solved.

New Scientist advises me that there are over 12 trillion (12,000,000,000,000) different ways to rearrange a single board. I suspect that this program should generate enough games to keep even the most ardent fan occupied.

The program is quite modular and should be easy to follow, so here we go.

```
1000 REMark Sudoku Generator
1005 REMark By Norman Dunbar
1010 REMark ------------------------------------------------------------------------
1015 REMark Information and 'stuff' in this program are taken from the public
1020 REMark domain plus various articles and discussions in the following
1025 REMark magazines :
1030 :
1035 REMark PC Plus - assorted issues between 2005 and 2006.
1040 REMark New Scientist - Christmas 2005 issue.
1045 REMark ------------------------------------------------------------------------
1050 REMark NOTE : There is no copyrighted material in this program.
1055 REMark ------------------------------------------------------------------------
1060 :
1065 REPeat MainLoop
1070   Initialise
1075   ChooseLevel
1080   IF ChosenLevel = 0 THEN EXIT MainLoop: END IF
1085   Rows
1090   Columns
1095   SegmentRows
1100   SegmentColumns
1105   Rotate
1110   Scramble
1115   CLS
1120   DisplayGrid #1
1125   PrintGrid
1130   IF Debugging THEN CLOSE #3: END IF
1135 END REPeat MainLoop
1140 :
1145 :
1150 STOP
1155 :
1160 :
```

The code above simply introduces the program and shows the main loop which sets everything up and calls the various routines to create (sort of) a new challenge.

```
1165 DEFine PROCedure Initialise
1170   RANDOMISE DATE
1175   RESTORE 1210
1180   :
1185   REMark Define how many templates for *Each* difficulty level we have
1190   REMark in the data statements at lines 20000 onwards. There must be
1195   REMark the same number of templates in each level.
1200   :
1205   READ NumTemplates
1210   DATA 3
1215   :
1220   REMark The Main grid
1225   DIM Cells(9, 9)
1230   :
1235   REMark a working copy - if rotation etc required
1240   DIM WorkCells(9, 9)
1245   :
1250   REMark Used to randomise the cells in a single column
1255   DIM MixerCells(9)
1260   FOR x = 1 TO 9: MixerCells(x) = x: END FOR x
1265   :
1270   REMark used to randomise cols, rows, and 3 by 3 grids
1275   DIM WeeMixer(3)
1280   FOR x = 1 TO 3: WeeMixer(x) = x: END FOR x
1285   :
1290   REMark Constants for the chosen difficulty level etc
1295   Done = 0
1300   Debugging = 0
1305   :
1310   REMark Printer device
1315   Printer$ = 'par' : REMark CHANGE THIS TO SUIT YOUR SYSTEM
1320   :
1325   REMark Assume user has chosen 'done' initially.
1330   ChosenLevel = 0
1335 END DEFine Initialise
1340 :
1345 :
```

The initialisation code sets up the game by first seeding the random number generator and reading the data line which defines how many templates I have built for each difficulty level.

In this version of the program there are three. A couple of 9 by 9 grids are then defined, followed by a 9 cell 'mixer' grid which is used later to scramble the 9 digits in each row. A smaller three cell mixer is used to mix up the columns and rows within a single segment and also to shuffle segments around.

The 'printer' device is then initialised to 'par' on my system - but you should change this to suit wherever you print to.

```
1350 DEFine PROCedure ChooseLevel
1355   REMark The user chooses a level and from that we choose one of that
1360   REMark level's templates to use when generating a puzzle. If the user
1365   REMark chooses the 'impossible' level, then we simply show the 'bad'
1370   REMark grid.
1375   REPeat GetLevel
1380     CLS
1385     PRINT 'QSudoku Generator 0.1'
1390     PRINT 'By Norman Dunbar'
1395     AT 4, 0: PRINT 'Please choose from the following difficulty levels :'
1400     AT 6, 10: PRINT 'Level 1 - Easy'
```

```
1405      AT 7, 10: PRINT 'Level 2 - Medium'
1410      AT 8, 10: PRINT 'Level 3 - Hard'
1415      AT 9, 10: PRINT 'Level 4 - Difficult'
1420      AT 10, 10: PRINT 'Enter 0 to quit'
1425      AT 12,0: INPUT 'Please enter your selection, 0 to 5 only : ', Choice$
1430      :
1435      IF Choice$ = '' THEN NEXT GetLevel: END IF
1440      IF NOT Choice$ INSTR '12340' THEN NEXT GetLevel: END IF
1445      ChosenLevel = '0' & Choice$
1450      EXIT GetLevel
1455    END REPeat GetLevel
1460    :
1465    IF ChosenLevel = Done THEN RETurn : END IF
1470    REPeat GetDebugging
1475      AT 13,0: INPUT 'Please type 1 for debugging, 0 for not : ';Choice$
1480      IF Choice$ = '' THEN NEXT GetDebugging: END IF :
1485      IF NOT Choice$ INSTR '01' THEN NEXT GetDebugging: END IF
1490      Debugging = '0' & Choice$
1495      EXIT GetDebugging
1500    END REPeat GetDebugging
1505    :
1510    IF Debugging THEN DebugChannel: END IF
1515    :
1520    REMark Data for each level starts at line 20000. There are 'n' templates
1525    REMark for each level. This is set at line 2040 above. Each template has
1530    REMark 9 lines of data. To Determine which template we use, we generate
1535    REMark a random number between 1 and NumTemplates.
1540    :
1545    RandomTemplate = RND(1 TO NumTemplates)
1550    RestoreLine = 20000 + (1000 * (ChosenLevel -1)) + 100 * (RandomTemplate -1)
1555    RESTORE RestoreLine
1560    :
1565    FOR Row = 1 TO 9
1570      FOR Col = 1 TO 9
1575        READ Cells(Row, Col)
1580      END FOR Col
1585    END FOR Row
1590    :
1595    IF Debugging THEN
1600      PRINT #3, 'Initial Grid, Level ' & ChosenLevel & ', Template ';
1605      PRINT #3, RandomTemplate & ', Data line ' & RestoreLine
1610      DisplayGrid 3
1615      PRINT #3
1620    END IF
1625 END DEFine ChooseLevel
1630 :
1635 :
```

The code in the ChooseLevel allows the user to define the level of difficulty desired, and, for any give level, initialises the board with a randomly chosen template for that level.
Should the user have chosen to show what is happening (or debugging) then the initial grid and some information is written to the chosen debugging file or device. I did find this very useful when debugging the code.

```
1640 DEFine PROCedure MixUp
1645   REMark Scramble the three entries in WeeMixer
1650   LOCal x, y, SaveMe
1655   FOR x = 1 TO 3: WeeMixer(x) = x: END FOR x
1660   FOR x = 3 TO 2 STEP -1
1665      y = RND(1 TO x)
```

```
1670       SaveMe = WeeMixer(y)
1675       WeeMixer(y) = WeeMixer(x)
1680       WeeMixer(x) = SaveMe
1685    END FOR x
1690 END DEFine MixUp
1700 :
```

MixUp simply takes the three cell mixer and randomises it. This sets the new order for rows and columns within a segment. It works on the principle that if you have three columns (say) then they can be arranged in one of 6 different ways :

ABC ACB BAC BCA CAB CBA

So scrambling the digits 123 in a similar manner decides the new order of columns 1, 2 and 3 and so on.

```
1705 DEFine PROCedure Rows
1710   REMark Scramble each set of three rows amongst themselves. This
1715   REMark maintains the Sudoku-ness of the set of three rows.
1720   REMark Do this three times each for each set of three rows.
1725   LOCal Row, mix, Offset
1730   IF Debugging : PRINT #3, 'Rows': END IF
1735   FOR mix = 1 TO 3
1740     MixUp
1745     Offset = (mix -1 ) * 3
1750     IF Debugging: PRINT #3, 'Mix ' & mix: END IF
1755     FOR Row = 1 TO 3
1760       SwapRows Offset + Row, Offset + WeeMixer(Row)
1765       IF Debugging THEN
1770          PRINT #3, 'Swapping rows ' & (Offset + Row) & ' and ' & (Offset +
WeeMixer(Row))
1775          DisplayGrid 3
1780       END IF
1785     END FOR Row
1790   END FOR mix
1795 END DEFine Rows
1800 :
```

The Rows (and Columns below) work in a similar manner. Given a starting grid, three rows (or columns) are rearranged using the three cell mixer explained above. This preserves the 'Soduku-ness' of the board so that the rules still apply. We only mix up cells within a segment's row or column. For example, rows 1, 2 and 3 are shuffled, then 4, 5 and 6 and so on. We do this three times - which should cover all eventualities.

```
1805 DEFine PROCedure SwapRows(FromRow, ToRow)
1810   REMark Swap the cells from one row to another row.
1815   :
1820   LOCal ThisCell, SaveCell
1825   FOR ThisCell = 1 TO 9
1830     SaveCell = Cells(FromRow, ThisCell)
1835     Cells(FromRow, ThisCell) = Cells(ToRow, ThisCell)
1840     Cells(ToRow, ThisCell) = SaveCell
1845   END FOR ThisCell
1850 END DEFine SwapRows
1855 :
1860 :
```

SwapRows is called to exchange the contents of a pair of rows. It does this by saving the current contents of the 'From' cells before overwriting them with the 'To' cells and finally putting the saved values into the 'To' cells. Quite simple.

The following code to shuffle columns is almost exactly as described above for shuffling rows and will not be explained further.

```
1865 DEFine PROCedure Columns
1870   REMark Scramble each set of three rows amongst themselves. This
1875   REMark maintains the Sudoku-ness of the set of three rows.
1880   LOCal Col, mix, Offset
1885   IF Debugging : PRINT #3, 'Columns': END IF
1890   FOR mix = 1 TO 3
1895     MixUp
1900     Offset = (mix -1) * 3
1905     IF Debugging : PRINT #3, 'Mix ' & mix: END IF
1910     FOR Col = 1 TO 3
1915       SwapColumns Offset + Col, Offset + WeeMixer(Col)
1920       IF Debugging THEN
1925         PRINT #3, 'Swapping columns ' & (Offset + Col) & ' and ' & (Offset +
             WeeMixer(Col))
1930         DisplayGrid 3
1935       END IF
1940     END FOR Col
1945   END FOR mix
1950 END DEFine Columns
1955 :
1960 :
1965 DEFine PROCedure SwapColumns(FromCol, ToCol)
1970   REMark Swap the cells from one column to another column.
1975   :
1980   LOCal ThisRow, SaveCell
1985   FOR ThisRow = 1 TO 9
1990     SaveCell = Cells(ThisRow, FromCol)
1995     Cells(ThisRow, FromCol) = Cells(ThisRow, ToCol)
2000     Cells(ThisRow, ToCol) = SaveCell
2005   END FOR ThisRow
2010 END DEFine SwapRows
2015 :
2020 :
2025 DEFine PROCedure SegmentRows
2030   REMark Swap segments (3 * 3 smallgrid) vertically
2035   LOCal SegRow
2040   MixUp
2045   IF Debugging : PRINT #3, 'Segment Rows': END IF
2050   FOR SegRow = 1 TO 3
2055     SwapSegmentRow (SegRow), (WeeMixer(SegRow))
2060     IF Debugging THEN
2065       PRINT #3, 'Swapping segment rows ' & SegRow & ' and ' & WeeMixer(SegRow)
2070       DisplayGrid 3
2075      END IF
2080   END FOR SegRow
2085 END DEFine SegmentRows
2090 :
2095 :
```

The code above in SegmentRows is very similar to the code in the Rows procedure but differs in that it actually swaps groups of three rows at a time, not just a single row. The code might swap the top three rows of the grid with the middle three rows. Again, this preserves the grid and allows the rules of Soduku to be valid.

```
2100 DEFine PROCedure SwapSegmentRow(FromGrid, ToGrid)
2105   LOCal SegRow, FromSeg, ToSeg
2110   FromSeg = (3 * FromGrid) -3
```

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 Third 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

```
2115   ToSeg = (3 * ToGrid) -3
2120   FOR SegRow = 1 TO 3
2125     SwapRows FromSeg + SegRow, ToSeg + SegRow
2130   END FOR SegRow
2135 END DEFine SwapSegmentRow
2140 :
2145 :
```

The code above does the hard work of swapping over groups of three rows and simply calls the SwapRows code described above to swap each of the three rows affected.

```
2150 DEFine PROCedure SegmentColumns
2155   REMark Swap segments (3 * 3 smallgrid) horozontally
2160   LOCal SegCol
2165   MixUp
2170   IF Debugging : PRINT #3, 'Swapping Segment Columns': END IF
2175   FOR SegCol = 1 TO 3
2180     SwapSegmentColumn (SegCol), (WeeMixer(SegCol))
2185     IF Debugging THEN
2190        PRINT #3, 'Swapping segment columns ' & SegCol & ' and ' &
WeeMixer(SegCol)
2195        DisplayGrid 3
2200     END IF
2205   END FOR SegCol
2210 END DEFine SegmentColumns
2215 :
2220 :
2225 DEFine PROCedure SwapSegmentColumn(FromGrid, ToGrid)
2230   LOCal SegCol, FromSeg, ToSeg
2235   FromSeg = (3 * FromGrid) -3
2240   ToSeg = (3 * ToGrid) -3
2245   FOR SegCol = 1 TO 3
2250     SwapColumns  FromSeg + SegCol, ToSeg + SegCol
2255   END FOR SegCol
2260 END DEFine SwapSegmentColumn
2265 :
2270 :
```

The code above works in a similar manner to that described for swapping segment rows and will not be discussed further, other than to say that it obviously works on groups of columns rather than rows.

```
2275 DEFine PROCedure Rotate
2280   REMark Rotate the grid clockwise by 90 degrees. This is the
2285   REMark only rotation that cannot be carried out by some other
2290   REMark manipulation.
2295   REMark We only rotate if RND(1 to 2) generates a 2.
2300   :
2305   LOCal Row, Col
2310   IF RND(1 TO 2) <> 2 THEN RETurn : END IF
2315   FOR Row = 1 TO 9
2320     FOR Col = 1 TO 9
2325       WorkCells(Col, 10 - Row) = Cells(Row, Col)
2330     END FOR Col
2335   END FOR Row
2340   :
2345   FOR Row = 1 TO 9
2350     FOR Col = 1 TO 9
2355       Cells(Row, Col) = WorkCells(Row, Col)
2360     END FOR Col
2365   END FOR Row
```

```
2370    IF Debugging THEN
2375       PRINT #3, 'Rotating 90 degrees clockwise'
2380       DisplayGrid 3
2385    END IF
2390 END DEFine Rotate
2395 :
2400 :
```

According the PC Plus, there is only a need to rotate a grid by 90 degrees clockwise because all other rotations are able to be produced by other means. I am not good enough to dispute (or even confirm) this fact so the code remains here and only rotates as described.

Not all grids are rotated - there is a 50% chance that any given grid will get rotated, so don't be surprised if the debugging output doesn't always show it.

```
2405 DEFine PROCedure Scramble
2410    REMark Scramble each of the Rows at random. Each row is
2415    REMark obviously scrambled in exactly the same order or
2420    REMark we no longer have a Sudoku !
2425    LOCal Row, Col, SaveMe
2430    FOR Row = 1 TO 9
2435      MixerCells(Row) = Row
2440    END FOR Row
2445    IF Debugging : PRINT #3, 'MixerCells = '; MixerCells!: PRINT #3: END IF
2450    :
2455    FOR Row = 9 TO 2 STEP -1
2460      Col = RND(1 TO Row)
2465      SaveMe = MixerCells(Col)
2470      MixerCells(Col) = MixerCells(Row)
2475      MixerCells(Row) = SaveMe
2480    END FOR Row
2485    IF Debugging : PRINT #3, 'MixerCells = '; MixerCells!: PRINT #3: END IF
2490    :
2495    FOR Row = 1 TO 9
2500      FOR Col = 1 TO 9
2505        Cells(Row, Col) = MixerCells(Cells(Row, Col))
2510      END FOR Col
2515    END FOR Row
2520    IF Debugging THEN
2525       PRINT #3, 'Scrambling'
2530       DisplayGrid 3
2535    END IF
2540 END DEFine Scramble
2545 :
2550 :
```

Finally, we have the Scrambling code. What this does - and it took me ages to really understand it, PC Plus again - is to take each cell in each row and 'randomise' it into a different order. The digits are still 1 to 9 and unique in a row so there's no problems there. It re-arranges every row into the same order so that the rules are preserved. It is this final scrambling which produces the biggest number of permutations for any given grid.

```
2555 DEFine PROCedure DisplayGrid(Channel)
2560    REMark Display the grid
2565    LOCal Row, Col, Value
2570    FOR Row = 1 TO 9
2575      FOR Col = 1 TO 9
2580        Value = Cells(Row, Col)
2585        IF Channel <> 4 : REMark PRINTER CHANNEL
```

```
2590          PRINT #Channel, Value;
2595        ELSE
2600          IF Value = 0
2605            PRINT '_';
2610          ELSE
2615            PRINT #Channel, Value;
2620          END IF
2625        END IF
2630        IF Col/3 = INT( Col/3) THEN PRINT #Channel, ' '; : END IF
2635      END FOR Col
2640        IF Col/3 = INT( Col/3) THEN PRINT #Channel, ' '; : END IF
2645      END FOR Col
2650      PRINT #Channel
2655      IF Row/3 = INT(Row/3) THEN PRINT #Channel: END IF
2660    END FOR Row
2665    PRINT #Channel
2670 END DEFine DisplayGrid
2675 :
2680 :
```

The code above allows the grid to be displayed. The format is not pretty by any means, but is adequate for me (I think). Assuming we are not sending output to the printer, zeros are printed in place of gaps which makes for better readability. I imagine that anyone using this program will improve the above code to make things more Soduku-like.

```
2685 DEFine PROCedure PrintGrid
2690  REMark Let the user choose to print out the grid, and if so chosen,
2695  REMark simply open a printer channel and call DisplayGrid above with
2700  REMark the chosen channel id.
2705  LOCal Choice$
2710  REPeat PrintLoop
2715   AT 12,0;: INPUT "Do you want to print this puzzle (Y/N) ";Choice$
2720   IF Choice$ INSTR 'YyNn' THEN EXIT PrintLoop: END IF
2725  END REPeat PrintLoop
2730  IF Choice$ == 'n' THEN RETurn : END IF
2735  OPEN_OVER #4, Printer$
2740  DisplayGrid #4
2745  CLOSE #4
2750 END DEFine PrintGrid
2755 :
2760 :
```

The above code prints the grid to the chosen printer device. This calls DisplayGrid above to do the hard work - again, the output isn't very pretty.

```
2765 DEFine PROCedure DebugChannel
2770   REPeat DebugLoop
2775     INPUT 'Please enter a file name for the debug output : ';DebugFile$
2780     IF DebugFile$ <> '' THEN EXIT DebugLoop: END IF
2785   END REPeat DebugLoop
2790   :
2795   OPEN_OVER #3,DebugFile$
2800 END DEFine DebugChannel
2805 :
2810 :
```

This code handles the user's chosen debug parameters and files. There's nothing much to explain here!

Finally, the code below must start at the line numbers give - do not renumber these lines. Each level of difficulty has the same number of templates (in this case there are three).

```
2815 REMark Data for the templates for
     each level. The data starts at
2820 REMark line 20000 plus
2825 REMark       1000 * (ChosenLevel -
     1) plus
2830 REMark        100 * (RandomTemplate
     - 1)
2835 :
2840 :
20000 REMark Easy Template 1
20001 DATA 4,0,0,0,0,0,0,0,7
20002 DATA 0,6,9,0,0,0,8,2,0
20003 DATA 0,0,0,2,5,9,0,0,0

20004 DATA 0,3,0,0,2,0,0,7,0
20005 DATA 0,0,0,4,6,3,0,0,0
20006 DATA 0,1,0,0,7,0,0,9,0
20007 DATA 0,0,0,5,3,8,0,0,0
20008 DATA 0,4,7,0,0,0,3,1,0
20009 DATA 3,0,0,0,0,0,0,0,2
20019 :
20100 REMark Easy Template 2
20101 DATA 0,0,6,0,2,0,5,0,0
20102 DATA 0,0,3,6,0,7,8,0,0
20103 DATA 2,0,0,0,0,0,0,0,7
20104 DATA 0,2,0,0,6,0,0,3,0
20105 DATA 0,0,0,2,0,4,0,0,0
20106 DATA 0,9,0,0,8,0,0,4,0
20107 DATA 4,0,0,0,0,0,0,0,8
20108 DATA 0,0,2,7,0,6,9,0,0
20109 DATA 0,0,5,0,9,0,2,0,0
20119 :
20200 REMark Easy Template 3
20201 DATA 6,0,9,0,0,0,5,0,1
20202 DATA 0,0,0,6,0,5,0,0,0
20203 DATA 4,0,0,0,0,0,0,0,9
20204 DATA 0,3,0,4,0,1,0,7,0
20205 DATA 0,2,0,0,0,0,0,8,0
20206 DATA 0,9,0,3,0,6,0,1,0
20207 DATA 9,0,0,0,0,0,0,0,8
20208 DATA 0,0,0,8,0,2,0,0,0
20209 DATA 5,0,7,0,0,0,6,0,3
20219 :
21000 REMark Medium Template 1
21001 DATA 0,0,0,5,6,2,0,0,0
21002 DATA 0,2,3,1,0,8,6,9,0
21003 DATA 5,0,0,0,0,0,0,0,8
21004 DATA 0,0,4,0,0,0,1,0,0
21005 DATA 6,3,0,0,0,0,0,8,2
21006 DATA 0,0,7,0,0,0,5,0,0
21007 DATA 8,0,0,0,0,0,0,0,3
21008 DATA 0,5,6,8,0,9,4,2,0
21009 DATA 0,0,0,6,2,7,0,0,0
21019 :
21100 REMark Medium Template 2
21101 DATA 0,0,8,4,0,5,6,0,0
21102 DATA 0,4,0,7,0,9,0,3,0
21103 DATA 3,0,0,0,0,0,0,0,4
21104 DATA 1,6,0,0,0,0,0,8,7
21105 DATA 0,0,5,9,0,4,3,0,0
21106 DATA 4,7,0,0,0,0,0,2,5
21107 DATA 8,0,0,0,0,0,0,0,3
21108 DATA 0,3,0,8,0,1,0,4,0
21109 DATA 0,0,1,3,0,7,2,0,0
21119 :
21200 REMark Medium Template 3
21201 DATA 9,0,7,0,0,0,8,0,3
21202 DATA 0,0,4,0,0,0,7,0,0
21203 DATA 8,3,0,0,7,0,0,9,2
21204 DATA 7,0,0,2,0,5,0,0,9
21205 DATA 0,0,2,0,0,0,5,0,0
21206 DATA 4,0,0,7,0,9,0,0,1
21207 DATA 5,4,0,0,8,0,0,2,7
21208 DATA 0,0,6,0,0,0,9,0,0
21209 DATA 2,0,3,0,0,0,1,0,6
21219 :
22000 REMark Hard Template 1
22001 DATA 0,1,0,4,0,8,0,3,0
22002 DATA 2,4,0,0,3,0,0,6,5
22003 DATA 0,0,0,0,0,0,0,0,0
22004 DATA 8,2,0,0,5,0,0,7,6
22005 DATA 0,0,0,0,0,0,0,0,0
22006 DATA 3,9,0,0,8,0,0,2,1
22007 DATA 0,0,0,0,0,0,0,0,0
22008 DATA 4,5,0,0,6,0,0,1,7
22009 DATA 0,7,0,9,0,4,0,8,0
22019 :
22100 REMark Hard Template 2
22101 DATA 0,0,0,9,1,3,0,0,0
22102 DATA 0,0,5,0,0,0,4,0,0
22103 DATA 9,6,0,0,0,0,0,3,8
22104 DATA 8,2,0,5,0,7,0,1,6
22105 DATA 1,0,3,0,0,0,9,0,4
22106 DATA 7,4,0,1,0,9,0,8,5
22107 DATA 6,8,0,0,0,0,0,4,9
22108 DATA 0,0,7,0,0,0,8,0,0
22109 DATA 0,0,0,6,9,8,0,0,0
22119 :
22200 REMark Hard Template 3
22201 DATA 0,0,9,1,0,4,3,0,0
22202 DATA 0,0,0,0,0,0,0,0,0
22203 DATA 3,0,0,7,9,5,0,0,6
22204 DATA 2,4,8,0,0,0,9,1,7
22205 DATA 7,0,0,0,0,0,0,0,4
22206 DATA 5,9,6,0,0,0,8,3,2
22207 DATA 8,0,0,2,5,7,0,0,9
22208 DATA 0,0,0,0,0,0,0,0,0
22209 DATA 0,0,5,8,0,6,7,0,0
22219 :
23000 REMark Difficult Template 1
23001 DATA 5,0,0,0,0,0,0,0,8
23002 DATA 0,4,0,0,0,0,0,1,0
23003 DATA 0,0,7,1,6,4,2,0,0
23004 DATA 1,0,0,4,0,6,0,0,9
23005 DATA 0,0,0,0,1,0,0,0,0
```

```
23006 DATA 6,0,0,2,0,8,0,0,3
23007 DATA 0,0,6,8,7,5,9,0,0
23008 DATA 0,1,0,0,0,0,0,5,0
23009 DATA 8,0,0,0,0,0,0,0,4
23019 :
23100 REMark Difficult Template 2
23101 DATA 0,1,0,2,0,6,0,5,0
23102 DATA 0,0,0,0,0,0,0,0,0
23103 DATA 0,0,9,5,0,8,1,0,0
23104 DATA 1,9,0,0,4,0,0,2,7
23105 DATA 0,0,0,0,3,0,0,0,0
23106 DATA 3,7,0,0,2,0,0,1,8
23107 DATA 0,0,2,3,0,4,8,0,0
23108 DATA 0,0,0,0,0,0,0,0,0
23109 DATA 0,5,0,1,0,7,0,6,0
23119 :
23200 REMark Difficult Template 3
23201 DATA 0,0,3,9,0,8,7,0,0
23202 DATA 0,0,0,0,0,0,0,0,0
23203 DATA 2,0,0,0,0,0,0,0,6
23204 DATA 5,0,7,2,0,1,6,0,8
23205 DATA 0,1,0,0,8,0,0,7,0
23206 DATA 6,0,9,7,0,3,4,0,2
23207 DATA 8,0,0,0,0,0,0,0,1
23208 DATA 0,0,0,0,0,0,0,0,0
23209 DATA 0,0,4,3,0,6,9,0,0
23219 :
24039 :
30000 DEFine PROCedure sa
30010   SAVE_0 win1_source_QSoduku_bas
30020 END DEFine
```

That's all for now. By the time you read this I shall hopefully have returned from a couple of weeks in sunny South Africa - my wife needs another holiday in the sun it appears. I'm like a fine wine, myself, I do not travel well !

See you next time and enjoy creating your own Soduku games.

Start Here - Part 5

Essential Information For 21st Century QLers

by Roy Wood

So now it is time to get your hands dirty with some code for the BOOT file. We have done the groundwork and, just in case you missed some of the earlier stuff I will repeat it a little. I am going to make very few assumptions about the programs you are going to load but I will assume a system based on the Pointer Environment either from the individual files, HOT_REXT, WMAN, PTR_GEN (which are now freeware) or by my own preference, SMSQ/E. I am also going to assume that you will be using QPAC 2 and give a few tips about using this to launch other programs. If you are using the original Pointer Environment files on a Gold or Super Gold Card system you will need to have the Toolkit II extensions loaded to run some of the commands in this file. you can do this by starting with the following line:

```
10 TK2_EXT
```

You can get the extensions loaded automatically from the Gold/Super Gold Card by setting it from the command line. You can just type:

```
AUTO_TK2F1
```

This will then set the Card to Load the Toolkit II extensions and start the QL as if you pressed the F1 key. (Substitute the F1 for F2 if you are used to running the QL in that mode) You can disable this by typing:

```
AUTO_DIS
```

You will not need to load this if you are using SMSQ/E because it is built into the SMSQ/E code, but it is, however, needed to load the SMSQ/E code itself because it does provide the LRESPR keyword. You could use the older RESPR command in the format:

```
x=RESPR(SIZE) : LBYTES flp1_SMSQ/E,x : Call x
```

modified to substitute the size in bytes of the SMSQ/E code for the value SIZE but this is complex and it is much simpler to load the TKII extension and then use the simpler LRESPR command. I will be using that throughout this BOOT file.

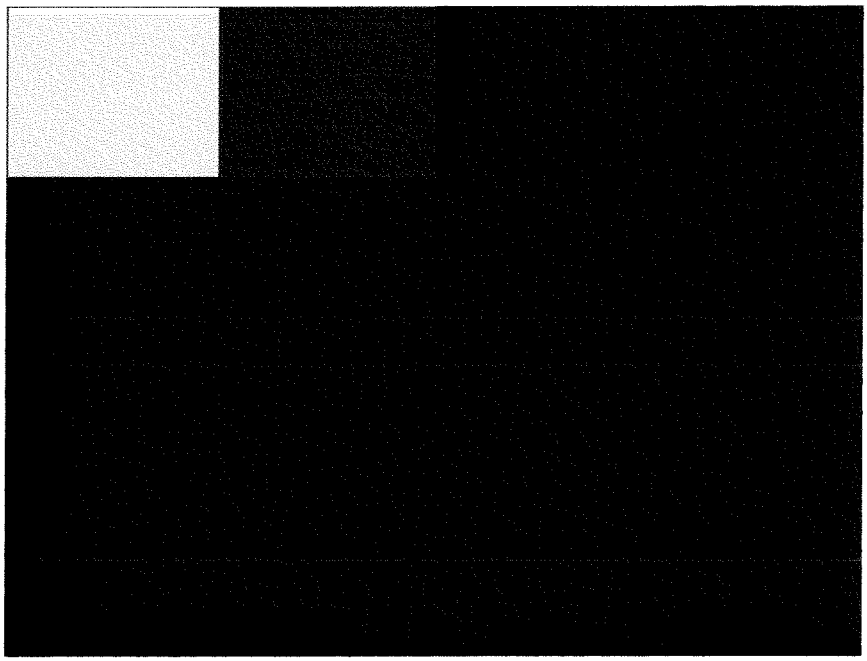
««IF YOU ARE NOT GOING TO USE SMSQ/E GOTO 'Before We Head Off'»»
(Slap Wrist!)

The next thing to do is to decide how we want it to look.

On To Your Own Screens

When you fire up one of the newer systems with a display size larger than the old 512 x 256 display the first screen will appear as a small box somewhere on the screen. This has often led people to assume that that is all the screen that

is available to them. The problem here is the way that most QL systems write to the screen. Most programs write directly in pixels so the letters will always occupy the same number of pixels on every system. If you move from a QL with a 14" monitor displaying 512 x 256 to an Aurora with a 14" SVGA monitor and a resolution of 1024 x 512 you can see that this is exactly four times smaller than the original display. Of course, if you plug that into a 20" monitor it will look a lot better.



How the screen appears when you start QPC or any of the versions of SMSQ/E is high resolution this one is 1024 x 768. Note although the three windows occupy the upper left all of the screen is usable.

A program like Quill (Roger Godley has done a lot of work on this and you can now have larger PSION display sizes) will occupy only one quarter of the screen because its window sizes are hard coded and it cannot be changed. Most of the later, P.E. programs, however, have a re-size option which allow them to be expanded to occupy as much screen as you wish them to. The wording on these programs will, however, remain the same size. There are some programs which do allow you to set larger fonts for use in the programs themselves. The two main programs are Text 87 and all of the programs which use the ProWesS vector graphics system.
For the sake of this explanation, however, we will concern ourselves with setting the main display size and three windows. personal preference comes into play here so you can do a lot with these three windows to set up the starting screen. The QL's basic windows are something like:

```
WINDOW#0,512,50,0,204 : WINDOW#1,256,
204,256,0 : WINDOW#2,256,204,0,0
```

If you break this down the commands are

`WINDOW` This is the SuperBasic command to define a window

`#0` This the channel which is linked to that window

`512` This is the width of the window in pixels.

`50` This is the height of the window in pixels.

`0` This is the horizontal origin of the window.

`204` This is the vertical origin of the window.

The vertical and horizontal origins of the window are always taken from the top left hand corner.
Imagine the screen to be a sheet of graph paper with X axis running horizontally and the Y axis vertically. In contrast to the normal way a graph is set out this one has its point of origin as the top left hand corner. if you count the pixels along the X axis you will get 512 and along the Y axis you will get 256. All of the Window calculations are taken from there.
So the above lines will draw the standard windows but we have not set the INK and PAPER colours. These can be set quite simply with the following line:

```
INK#0,4 : INK#1,7 :INK#2,0 and
PAPER#0,0 : PAPER#1,2 : PAPER#3,7
```

Again these commands are made up in the following way:

`INK` The SuperBasic command to set the ink colour

`#0` The channel which in which the ink is applied

`0` The actual colour used

The same setup is used for the 'PAPER' command. You can find the numbers for the colours in the QL Users Guide and many other publications such as Jan Jones SuperBasic book. Modern SMSQ/E High Colour drivers give you access to many more colours but I will deal with how to get different colours in your basic three windows later. For the time being we will stick

with Red, Black and White.
To clear the old windows and display only these ones you will need to issue commands to clear the screens so:

```
CLS#0 : CLS#1 :CLS#2
```

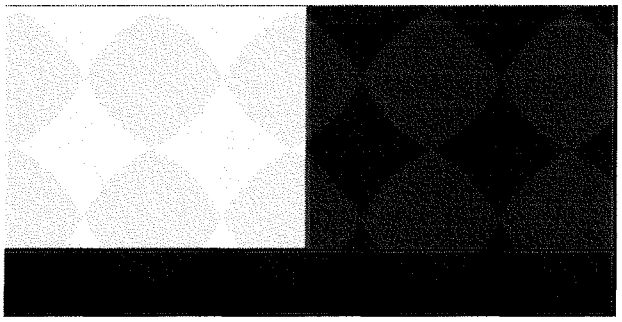
So far so good but the old QL windows had borders around them so we should do the same. To do this we need to use the 'BORDER' command.

```
BORDER#0,2,7 :BORDER#1,2,4 : BORDER#2,2,4
```

This will give you a white border, 2 pixels wide around the console channel (#0) and a green border, 2 pixels wide, around the other two channels (#1 & #2)
The BORDER command will accept a further parameter allowing you to create stipple effects so

```
BORDER#0,2,7,0 : BORDER#1,2,4,0 :
BORDER#2,2,4,0
```

will give you a black and white stipple effect on the border for the console channel and a green and black stipple effect on the other two.
The same extra parameter can be added to the 'PAPER' and 'INK' commands but the effect can often make text hard to read.



The familiar QL screen we all know and love

The First Step

So now we have the first few lines of our new boot file. If you have followed things up till now you will have a BOOT file which looks like this:

```
100 TK2_EXT : REM You only need this for
    Gold/SuperGold Card Systems
110 WINDOW#0,512,50,0,204
120 WINDOW#1,256,204,256,0
130 WINDOW#2,256,204,0,0
140 INK#0,4 : INK#1,7 : INK#2,0
150 PAPER#0,0 : PAPER#1,2 : PAPER#2,7
160 BORDER#0,2,7,0 : BORDER#1,2,4,0 :
    BORDER#2,2,4,0
170 CLS#1 : CLS#2 : CLS#0
180 PRINT#0,'DONE'
```

I have added the last line to print a message in the console channel to tell you that the process has got to the end. Try experimenting with the colour commands to get windows, borders and inks of varying shades. I, personally, did not like the original colour setup so I made all of my windows black and give them solid green borders but this is a personal preference and you may have other ideas. This was before the Colour drivers came along and now they are completely different.

Get a Bigger Picture

All of the above relates to the standard QL screen but SMSQ/E users on all platforms except the standard QL or an Aurora plugged into a standard QL monitor have access to larger screen areas and that has lead to some confusion when first running their systems.
First let us get one thing absolutely clear. You have access to the entire area seen on the monitor provided the monitor is set correctly to cope with the computer it is connected to. The small version of the QL's opening screen is a 512 x 256 representation displayed on a larger screen. The reason for this small screen QL is, as I said before, that some people never opened windows in their own programs and just used those that were available when the computer was started up. Even some older commercial programs did this and if any of these was run on a system with larger windows the display would be muddled. It was therefore decided to continue with the current 512 x 256 screen as a startup and allow users to modify them for themselves.
If you are using any of the systems that use SMSQ/E you will have access to many different screen resolutions and the one that you choose to use will be decided by the quality of your monitor, your eyesight and your own preference. I tend to use 1024 x 512 on my Aurora system because that gives me a screen area four times the size of the standard QL and allows me to have several active programs at the same time. It is possible in this resolution to have two copies of QD side by side for instance. the Aurora will support resolutions from 512 x 256 to 1024 x 768 with various monitor limitations. Many of these are in the standard QL pixel ratio of 2:1 but some are in the PC ration of 4:3.
Other resolutions are available on different machines but platforms such as the QXL will only be able to display in the PC 4:3 pixel ratio and this will lead to some distortion on the screen. Running a QXL at 800 x 600 is not too bad a compromise however and that is the way used to run QPC on my laptop. QPC2 has made giant strides in screen resolution and can map the

original 512 x 256 resolution to a PC screen. This will look perfect if you boot straight to it but can look a little odd if you shift to it during a session. Personally speaking I prefer the higher resolution but the low one is useful if you are running programs such as the original Quill which are fixed in the old format.

So now we have to make a decision about the resolution that you want to use. The first consideration here is the programs you are going to run on the system. If you are using standard PSION programs they will only use 512 x 256 anyway and there was no way to change this until Roger Godley worked his magic. Running the Aurora at the lower resolution on an SVGA monitor will give a crisp, clear, and very snappy display and you can always change this from the command line later if you want to. For more information on this refer back to the earlier section of 'Start Here' which dealt with displays (Vol. 9 Issue 5). If, on the other hand, you use a selection of Pointer Environment (P.E.) programs you may want to use a larger display size and take advantage of the extended screen area. Most P.E. programs have a resize icon and can be made to expand to fit a larger screen. QSpread, for instance, will automatically open to fill the available screen (if the user sets the config block to do it) and can be resized to have two sheets on the screen at the same time if needed. At 1024 x 768 the number of cells visible is staggering!

There is a drawback to this, however, and that, as I mentioned above, is the character size. Most QL programs draw the characters on the screen directly. If you make the resolution larger the character size is smaller and, if your eyesight is challenged you may find the screen hard to read. You will, therefore, have to trade off increased viewing area against legibility.

ProWesS, on the other hand, gains from the increased pixel count because it uses vector graphics to write to the screen. this is the same method used by the PC and, although it is slower, it always displays the characters in the same size. Larger resolution just mean smoother graphics. Text 87 will also adapt itself to higher resolutions (Aurora users will have to patch the program by running the patch file on the SMSQ/E disk) but you will have to configure the program to get the best on-screen display.

One last consideration here is your monitor. Some monitors will refuse to display some resolutions with some dramatic results. I had a batch of monitors that would not run at 640 x 320 which, unfortunately, was the resolution that SMSQ/E for the Aurora is supplied in. On BOOTing up the monitor produced many strange noises and sparkling displays before the safety circuit kicked in to close it down. Some of the cheaper SVGA monitors will display characters with dull patches in the higher resolutions and make it hard to read. This was a very old type of monitor and most modern CRT ones will give a very good display. TFT's are more tricky because of the strict pixel layout and many will not work at all with the Aurora.

Most of this you will have to decide for yourself when it comes to writing your file. I will assume for the sake of this BOOT file that we are using QPC on an 800 x 600 PC and write the boot file to cope with that resolution.

First we will set the SMSQ/E display size with the line;

```
DISP_SIZE 800,600
```

and then we will alter our window size appropriately. The new BOOT file will look like this:

```
100 TK2_EXT
110 DISP_SIZE 800,600
120 WINDOW#0,800,100,0,500
130 WINDOW#1,400,498,400,0
140 WINDOW#2,400,498,0,0
150 INK#0,4 : INK#1,7 : INK#2,0
160 PAPER#0,0 :PAPER#1,2 :PAPER#2,7
170 BORDER#0,2,7,0 : BORDER#1,2,4,0 :
    BORDER#2,2,0
180 CLS#1: CLS#2: CLS#0
190 PRINT#0,'DONE'
```

Note that I have slightly reduced the height of windows 1 & 2 which allows a small gap between these and the console channel. This is not necessary but looks better. There is one command here that is not available from the standard QDOS/ SMSQ/E/ Tk2 command set and that is DISP_SIZE. This command can only be used if you have SMSQ/E running at the time. QPC has an SMSQE.BIN file which is used to provide the working environment. QXL users were provided with SMSQ.EXE which was used to fire up the the QXL board itself and run as a QDOS replacement. SMSQ was 99% compatible with standard QDOS and requires the use of the WMAN, PTR_GEN and HOT_REXT files to provide the P.E. This version does not include the DISP_SIZE command. Only SMSQ/E allows the user to change the resolution from the command line in this way.

Aurora users also have this problem. If you have an Aurora and have only used the standard ROMs or a Minerva you only have access to the

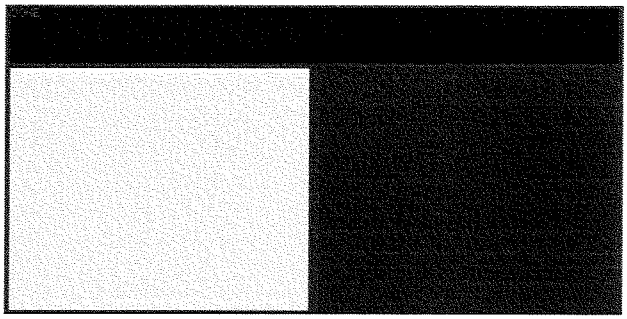
512 x 256 screen. Only when the SMSQ_GOLD operating system is run do you have access to higher resolutions and this contains the DISP_SIZE keyword needed for the BOOT file above. Assuming that you have a hard disk and a copy of SMSQ/E the above BOOT file should invoke SMSQ/E right at the start if you are an Aurora User. You should, therefore add the line:

```
IF VER$<>'HBA : LRESPR WIN1_SMSQ_GOLD
```

This test is used so that you do not run into a constant loop because, when SMSQ/E is started it automatically resets the machine and re-runs the BOOT file. There should be a test built into the SMSQ/E program to prevent a loop happening but some versions did not have this so it is worth considering. The variable VER$ will return the ROM which is running at the time. PRINT VER$ will give 'JS', 'JM', or 'AH' on standard QL ROMS or 'HBA' for SMSQ/E. To find the version number of SMSQ/E try `PRINT VER$(1)`
The start of the AURORA BOOT file should look like this:

```
100 TK2_EXT
110 IF VER$<>"HBA":LRESPR WIN1_SMSQ_GOLD
120 DISP_SIZE 768,384
130 WINDOW#0,768,50,0,334
140 WINDOW#1,384,334,384,0
150 WINDOW#2,384,334,0,0
160 INK#0,4 : INK#1,7 : INK#2,0
170 PAPER#0,0 :PAPER#1,2 :PAPER#2,7
180 BORDER#0,2,7,0 : BORDER#1,2,4,0 :
    BORDER#2,2,0
190 CLS#1: CLS#2: CLS#0
200 PRINT#0,'DONE'
```

If you are not running this on an Aurora but on QPC or the QXL omit line 110 which LRESPR's SMSQ/E and change the window sizes to the ones we discussed before. You should also change the window and display sizes to those discussed above. The lines quoted above will give the user access to the standard screen setup and it is really up to them which display size to set and how their screens will look.
One or two useful things should be mentioned before we move on. If you are running an SMSQ/E version and want to change the display size you should realise that the screen's top left hand corner is always 0,0. This means that if you open up the windows as suggested above and then change the display size the console window will probably be off-screen. Changing back to the original size will have to be typed blind. You could make the console window (#0) appear at the top of the screen instead of at the bottom and have access to it at any display size.



This is the screen turned upside down with channel# 0 at the top. This means you can access the command line in any resolution. Very useful. If you reduce the screen resolution but have opened the windows to larger sizes.

There is another way to change the display size and I will discuss that later.

Another thing to be considered is the QPAC 2 button frame. For many users the button frame overwrites the SuperBasic Windows thus making SuperBasic inactive. Some badly written BOOT files that I have seen start the buttons and then try to do something which needs the console window to be active. The BOOT will stop at this point until you click on or otherwise call the basic windows. This can be prevented by making the windows smaller so that the buttons do not overwrite the Basic area. This has the added advantage of leaving you with both the button frame and the Basic windows active at the same time. I will discuss QPAC 2 a bit later in this article.

One last thing before we leave the display area. Text. I like to have some display text on the screen when I boot up and, because I am selling software and hardware when I am at a QL show I usually put the system configuration in a text display in WINDOW#2. This is a simple process all you have to do is to add a few print statements to the BOOT file such as:

```
PRINT#2,"Hello, Dave."
```

(see 2001 - the movie not the line number)

I use these print statements to give me the version numbers of the superHermes and SMSQ/E these lines are as follows:

```
PRINT#2,"SMSQ/E v.";VER$(1)
PRINT#2,"superHermes v.";IPCVER$
```

These are useful indications of which versions you are running at the time. They will also return an error if the extensions or operating system has not been loaded.

If you are running a different display size you may like to change the size of the letters before you

THANK YOU!

I would like to thank YOU, the readers of QL Today, for your great support! Without you, QL Today would not exist for so long! Your support keeps us, the QL Today team, and the QL "dealers" going. 10 years of QL Today are a long time, and have been a lot of work too. We don't know what is going to happen in the next 10 years, but one thing is for sure: Volume 11 is coming up next!

print. you can change the size or the lettering in any window by using the following command:

```
CSIZE#2 2,1
```

Let us look again at the boot file as we now have it.

```
100 TK2_EXT
120 DISP_SIZE 768,384
130 WINDOW#0,768,50,0,334
140 WINDOW#1,384,332,384,0
150 WINDOW#2,384,332,0,0
160 INK#0,4 : INK#1,7 : INK#2,0
170 PAPER#0,0 :PAPER#1,2 :PAPER#2,7
180 BORDER#0,2,7,0 : BORDER#1,2,4,0 :
    BORDER#2,2,4,0
190 CLS#1: CLS#2: CLS#0
200 CSIZE#2, 2,1
200 PRINT#2,"Hello Dave"
210 PRINT#2,CHR$(10)
220 PRINT#2,"SMSQ/E v.";VER$(1)
230 PRINT#0,'DONE'
```

I have only invoked the SMSQ/E version number since many of you will not be using super-Hermes. You can of course change the text displayed on line 200 to anything you wish. The CHR$(10) that is printed in line 210 is the code for 'ENTER' and just gives you a blank line between the two statements (just PRINT #2 would do the same).

From now on The BOOT file is the same for all systems

I will write the boot file as if it was being done for an Aurora based machine running on an SVGA monitor at 768 x 384. Users who want to adapt it for QPC or a QXL should use the window definitions described above or experiment with their own definitions. The way the screen looks is up to the end user. One of the members of the Sussex User Group had a narrow Window #1 on the right side of the screen and Window #2 on the left with Window #0 as a box at the top left side. You can, of course just leave it as it is.

Apart from the screen capabilities the differences between the platforms are few. I am writing the following assuming a minimum memory of 2Mb (Gold Card) and a small hard drive. Users with ROMDisqs should substitute WIN1_ for ROM1_ and those running from ED disks should use FLP1_

If you are not going to use SMSQ/E you will have to load the Pointer Environment files if you want to use QPAC2 and have the Button Frame.

III Setting Defaults

We now come to the default settings side of the BOOT file. Strictly speaking you do not have to set these but they do sometimes come in handy and certain programs need to have them set up in order to work. The main two used by most users are PROG_USE and DATA_USE. the syntax for setting these is pretty simple. All you do is type:

```
PROG_USE WIN1_PROGS_
```

and

```
DATA_USE WIN1_DATA_
```

Their function is varied and depends very much on the user and the programs used. At its simplest once you have set the PROG_USE default you can start programs with shorter command lines. For example, if you have Quill stored on the hard disk as WIN1_PROGS_Quill, you can start the program with a simple

```
EX Quill
```

This makes writing the BOOT file easier because you can set the PROG_USE default and then use the above command to start the programs. It does, however, mean that you cannot separate programs into other subdirectories and still use the simplified command.

Another, more useful, side to the PROG_USE default is found in many modern programs which configure or otherwise need to access other programs. Once the default is set programs such as MENUCONFIG (Jochen Merz's level 2 configuration program) can be pointed directly to the directory containing most of the programs. On my Hard disk I have a subdirectory called 'PROGS' which contains most of the programs in normal use. I then subdivide the directory into PROGS_UTILS_, PROGS_ARCHIVERS_, PROGS_GIF_, PROGS_GAMES etc. The setting of the PROG_USE directory starts me in the right part of the hard disk to look for the program I wish to configure.

Once the PROG_USE default has been set it initiates a variable in the system which will stay until either the system is reset or a different PROG_USE is set. This variable is PROGD$ and it can be accessed by any program so you can use it in your own BASIC programs if you wish. ie. if you want to start a program from one of your own programs you could use a line such as:

```
EX PROGD$&'MYPROG'
```

to start a program in the specified directory or

```
PRINT PROGD$
```

to display the current set directory.

Many programmers have used this in their programs and give instructions that certain files should be stored in the PROG_USE directory in order for the program to be able to call on them. When copying a program to the hard disk and setting up the BOOT file it is wise to look at the BOOT provided with the Master disk of that program. This is a good indication of the programmers way of thinking. If it contains a PROG_USE or DATA_USE statement be very careful where you copy the essential files.

The advent of the 'config block' and environment variables, a concept foundered in the 'C' and UNIX community have alleviated the situation somewhat here although you do still come across some programs whose authors have decided on this as the best way to keep track of the files they need.

DATA_USE has a similar function in that it is the directory where some programs expect to find their operating data. It is also a default target directory for a variety of programs and, as such, could be considered to be a good primary branch from your main directory. In my directory there is a subdirectory called DATA which then branches out to encompass the various types of data needed by various programs. It also has a variable associated with it and this is called DATAD$. This variable can be used in your own programs in a similar way the PROGD$.

One anomaly here is that extensions are regarded as DATA and not as PROGrams. If the file has to be LRESPR'd then it will take the DATA default and not the PROG default. If you are trying to load the menu extensions for instance and use the line

```
xx LRESPR menu_rext
```

menu_rext must be in the DATA default directory and not the PROG default.

These two default device variables can be very useful when using things like Jochen Merz's Menu_rext extensions to call the correct directories for loading files and programs. More of this in a later section. There are, however, drawbacks to this device too. Again, the real fault lies in the way the system has developed over the years. Some programs have to have their DATA stored in the DATA default device and, even worse, some reset the DATA default during their operation. Early versions of Disk Mate 5 used to do the latter and this often led to problems after it had been run. This problem is now solved because the program stores the settings when it starts and restores them before quitting. There are not many programs that still do this and most are now configurable in one way or another. If you are using a program that needs to have files in pre-defined directories then it is always worth checking to see if it has been upgraded.

In addition to this, if you are using QPAC 2, typing EXEP 'Files' at the command line will always give you a directory of the DATA default and that, in itself, is exceptionally useful.

There are several other device defaults that can be set in the BOOT file. These other defaults are useful in certain circumstances but do beware because they can lead you into very murky waters indeed.

Toolkit II, as supplied on ROM or on the Gold and Super Gold Cards have a couple of devices which can be used to change system defaults. The first is designed to be used when you have to run programs which are hard coded to only run from microdrive.

```
FLP_USE MDV
```

for instance will allow you to copy the Quill program onto flp1_ and the DATA files, such as the Quill '_doc' files onto flp2_ and run the program from there with no configuration. It is, however, much better to configure the program to use flp1_ and flp2_ than to add this line to a permanent BOOT file because they effectively remove the calls to the actual floppy drives.

Another command is the

```
PAR_USE SER
```

Found in the same toolkit. This is meant for exactly the same sort of problem - allowing programs which are hard coded to use SER1 as the printer output device to use the parallel port provided by the Super Gold Card and some other expansions. Again the same drawback applies. The use of the command removes the PAR device from the table and calls to Par are, therefore, ignored.

In my opinion the most confusing of these commands is DEV_USE. This will allow you to redirect calls from one device to another. This means, in fact, you can rename a drive or directory and then call the files or programs in that directory with a short name. For example:

```
DEV_USE 1, WIN1_PROGS_
```

will set DEV1_ to be WIN1_PROGS_

IF you then use the line:

```
EX DEV1_QD
```

it will in fact execute the QD that is in WIN1_PROGS_. This is all very well but can be

very confusing because it is calling a file or program from a variable directory. If you are not very careful, you may forget you did this weeks or months down the line and then re-assign DEV1_ to some other directory. In that case all calls to DEV1_ go there and the BOOT file falls over.
Other authors have added other commands. Phil Borman has added PATH and SUBDIR (these can be found on the Qubide support disk) and, although they can be useful to trick Quill and other older programs into believing a long sub-directory tree is a short filename they can add confusion and should be used with care.
Full descriptions of all these and where to find the toolkits with them in can be found in Rich Mellor's excellent SuperBASIC Reference Manual.
I think that is about all for now and we will start loading resident extensions and programs in the next instalment.

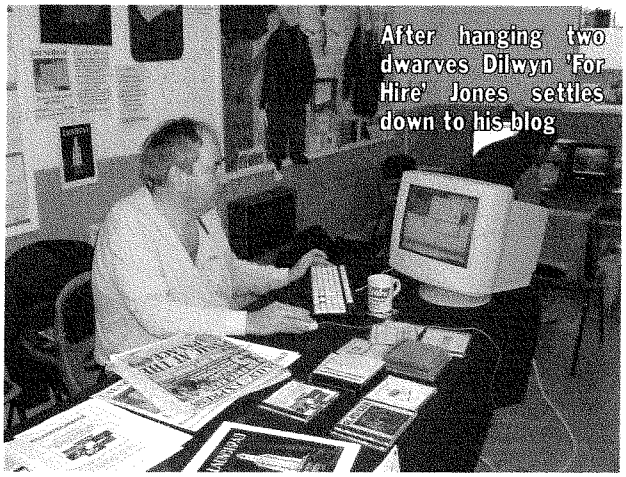
As I mentioned at the beginning of this series in QL Today I do not want, and am not qualified to, write all of it. I do need help for future articles. The next issue will feature the second part of my BOOT file section and that may run into the issue after that if space is tight. Tony Firshman has promised an article on Hardware detailing all the bits you could/can plug into a standard QL and I would like the next article in the series to deal with emulators.
Now I have only really used the QXL and QPC so I need some input. What I would like is for uses of the other emulators to write a shortish piece on how to install, configure, run, and manage the other emulators. Volunteers first and then I will assign the tasks. I would also like very short input from users about what they liked and disliked about the emulators they have tried.
This would be good for people new to emulators and may help them make their choice on what to use. It would also be good for the authors of the software so they can see what the users want and how well they are doing in providing it.
So here is you mission - if you choose to accept it. Let me know and I will collate and organise the final article(s).
It is important that we do this so that, if people decide they should move onto other platforms, they can take their QLs with them and stay part of the community.
In addition to this, if anyone would like to write on a subject for the 'Start Here' series get in touch and we will discuss it.

Roy Wood

The Manchester Show and Quanta AGM 2006

by Geoff Wicks

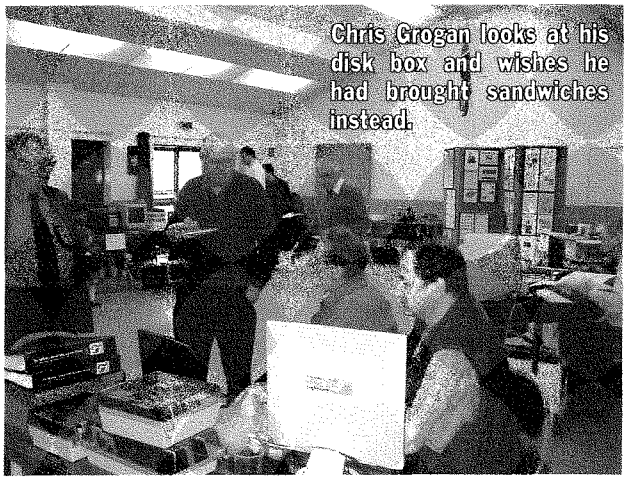
It has become a firm northern tradition. Once every two years when Quanta hits town, the hotels in Manchester sit chock-a-block full. Unfortunately not with QL-ers, but with football supporters. Come to think of it that means that Manchester must have some football team or other.
This year the number of QL-ers was fewer than normal. Perhaps the lowest there has ever been at a Quanta AGM. Numbers never rose above the low twenties on the Saturday, there were 18 at the show dinner, about a dozen at the Sunday lectures and just 14 at the AGM. Only one trader was present, but this was compensated by the presence of a pseudo trader and QL Senior Statesman. Dilwyn Jones, Custodian Pursuivant Extraordinary of the QL Archive, sat majestically at a table with his ceremonial range of CDs laid out before him. And I can exclusively reveal that before long he will have more interesting goodies to release to the QL World.


After hanging two dwarves Dilwyn 'For Hire' Jones settles down to his blog

Manchester is always a difficult location for a show. Many people attending have an overnight

stay, and the proximity of the show's location to the football stadium means local hotels are rapidly filled when there is a home game. Added to that there is the prejudice of the south against the north. Before one show, a few years back, a QL-er seriously questioned whether it would be possible to get to the venue without getting mugged. But then Sarah Gilpin tells a story that does little to dispel the myths about the north. Happen when she were nowt but a wee slip of a lass she had digs where they kept t'coal in t'bath. We may have been small in numbers, but this was a good show and no one appeared to be wandering around with nothing to do. Indeed the Sunday morning lectures contained two QL "You heard it or saw it here first" moments.

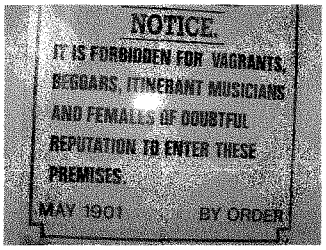
Manchester, or more accurately NEMQLUG, is probably the largest of Quanta's subgroups and certainly one of the most active. Some even see it as a threat. During the two years I was on the committee two Quanta worthies from down south independently warned me to watch out for those people in Manchester as they were planning to take Quanta over.



Chris Grogan looks at his disk box and wishes he had brought sandwiches instead.

NEMQLUG had made sure that its own members were working on their machines like an old style show, had asked Dilwyn to bring his QL history exhibition, had a display of robots and arranged a program of talks.

The show was held in the traditional location of the Endeavour Scout Hut, where most people miss the notice at the entrance:



"It is forbidden for vagrants, beggars, itinerant musicians and females of doubtful reputation to enter these premises."
This incidentally was not the reason the traders did not attend.
Both Tony Firshman and Roy Wood could qualify as "itinerant musicians" but Tony was singing innocently in his church choir and Roy was having gearbox problems. I assume this was a problem with his car although when an ancient rocker blames his gearbox it could mean he had nothing to wear to the gig.
As I walked around on the Saturday looking at the screens and talking to various people I realised there was an unofficial theme to the show. GD2 graphics are rapidly moving into the mainstream and people are talking about the image software they want to see. This theme became more structured on the Sunday, but more of that later.
Saturday evening was the traditional Chinese banquet at the Pond Quay restaurant. In the past impecunious QL-ers have found the price of the meal on the high side, and preferred to sit at separate tables in a Brewers Fayre. The Pond Quay is about a fiver more than QL-ers like to pay, but it would be hard to find greater value for money. Such is the popularity of this restaurant than in the four years Quanta has been going there it has doubled in size and has gained a branch in the prestigious setting of the Lowry Centre.

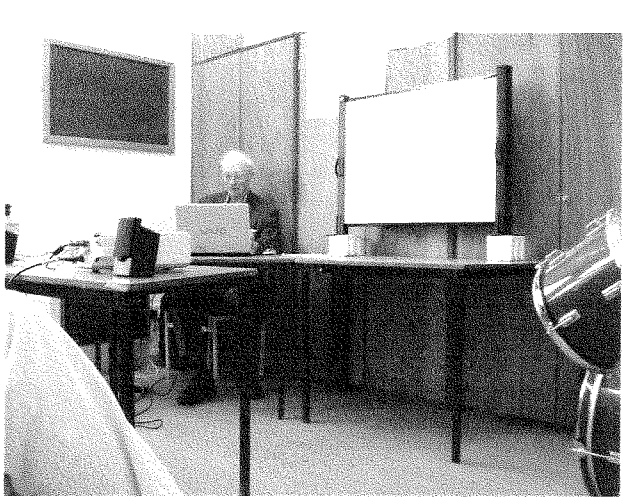
Northern Trains had managed to delay a fifteen minute train journey by fifteen minutes without any explanation, and I was the last to arrive at the dinner. I had been allocated a seat between John and Sarah Gilpin. The message could not have been clearer. Quanta's naughty boy had been placed between mummy and daddy to ensure his good behaviour. (This strategy didn't work!).

The Pond Quay is an ideal setting for a communal meal. The banquet takes about 3 hours and the courses are placed on rotating disks each of which serves 6 guests. (If, of course, FV or SH had been present they would have needed a disk for themselves.) We all managed to resist the temptation to see how hard we could spin the disks to see at which point centrifugal force would become visible.

Towards the end of the meal Veronica, the owner of the restaurant, joined us. We mischievously tipped her off that the chairman of Quanta has a history of enthusiastically smashing glasses at celebratory dinners. It was a joy to see his face as she crept up behind him and confiscated his drink.

After the culinary highlights of the Saturday evening I knew Sunday morning would be a letdown as the restaurant at my hotel was closed for renovations. I went to bed with a fantasy that NEMQLUG would take pity on me and on arrival at the scout hut I would be greeted with a full English served by topless waitresses. By morning reality had kicked in and I had to make do with a cup of coffee and a bacon butty on Manchester Airport Station.

Sunday morning was the lecture day. George Gwilt started off by demonstrating his Sudoku solving program. This has had some improvements made to it since he demonstrated it at QL is 21 and he has now written a 4 x 4 grid version.



After George, John Gregory and I presented a "Psion as never before" item, largely based on the article that appears in this issue of QL Today. After giving a brief history of the Psion programs, I first gave a demonstration of Marcel Kilgus' improvements to Xchange. We then moved on to Roger Godley's work, and this was the first time his updated Psion programs have been publicly demonstrated.

Roger is a QL-er who lives in Spain and he is slowly modifying the Psion suite for use on high resolution GD2 colour screens. He sees this as being very much a Quanta project and his work is a major acquisition for the Quanta library. What had been thought impossible for 20 years has become a reality in the last six months. We first of all showed the Quill versions that are already in the library and then went on to Abacus and Archive versions that John and I had privately received in our capacities of Quanta Librarian and QL Today Editor respectively.

People were amazed to see the difference in the working area between the standard Psion screen and a widescreen version of Abacus which fills a 1024 x 768 screen. Roger Godley had asked John Gregory to sound out opinion if we would

like the Psion programs to be moveable and have resizeable windows. We QL-ers are demanding folk and said yes.
After the Psion presentation we had another QL first. Things got really technical as George Gwilt described some changes that may come to QPC2. He has pointed out for some time that most QL systems (Q60, Q40, QXL and the Gold Cards) use a 68020 processor whereas QPC2 only emulates the 68000. He has been working with Marcel to add 68020 instructions to QPC2.
Although the technical details were only understood by two or three people at the show, the central message was clear. The changes will allow programmers to use more compact code with fewer commands and this will increase programming possibilities and speed. We can expect more Turbo improvements and more graphical possibilities, plus a QPC2 compatible version of GWASS.
In the ensuing discussion several points came up. As QL graphics improve, we are being increasingly confronted with the limitations of the Aurora card and native hardware. Are we moving to an emulator based, software only, QL? As yet we have no method of dumping GD2 screens to a printer. Who would be able to write this? Could ProWeSs be written to make it fully compatible with GD2 systems and would the extra speed allow us to make more use of vector fonts?


The Gang of Four

The last item of the Manchester weekend was the Quanta AGM. This has been a good year for Quanta and the chairman was able to make a positive report to members. Highlight of the year had been the successful "QL is 21" show in Portsmouth, but also there have been improvements to the technical quality of the Quanta Magazine and the first experiments to produce this in downloadable form. On the negative side it has proved impossible to make major improvements to the website.
The committee have also been looking at shows and hope to stimulate interest in these by paying for bed and breakfast costs for 2 or 3 lecturers and demonstrators. This led to some discussion about the traders. Four of the last six Quanta sponsored shows have seen a poor attendance of traders and the question was raised if we should think of holding traderless shows in future. The chairman clearly wanted to keep all options open and it is was generally agreed that the traders did have good reason to feel a little neglected at QL is 21.
The treasurer reported that Quanta had made a loss of £2,890 during the year, but this was largely because of the £3,167 cost of QL is 21. This had been financed from the Accumulated Fund. Without QL is 21 Quanta would more or less have broken even. No detailed breakdown of the costs of QL is 21 was given, and there was some discussion over the cost of the AA sign posting.
There were only four nominations for the Quanta Committee and all were elected unopposed. John Mason, Sarah Gilpin and John Gilpin all remain in their present offices of chairman, secretary and treasurer for a further 3 year period. Roy Brereton remains on the committee, but in a new capacity as librarian. An amendment to the constitution to formalise the discontinuance of the library account was approved.
Quanta's working practices do not allow for any other business, but members raised three matters. Membership figures published in the magazine show a 40% decline in membership from 272 to 161 in 13 months. In practice these are not equivalent figures as all 2006 subscriptions are not yet in, and, in fact, membership has held up well over the past year. The committee agreed to publish the membership as at 31st December 2005 in the magazine so that accurate comparisons can be made.
There was a further query about whether members had been sent an accurate copy of the constitution in particular with regard to article 9.1. The chairman confirmed that the signatures of 100 members and not 5% (about 13) were required to call a Special General Meeting, but gave an undertaking that this is being reviewed by the committee. Given that officers are now elected for a three year period; that about 40% of members have to sign a request for a Special General Meeting and then deposit £300; and that the chairman ruled that there could be no formal item of "Any Other Business", serious questions can now be asked about how far Quanta remains structurally a democratic organisation.

Finally Dilwyn Jones asked whether Quanta would be prepared to contribute financially to stimulate a programmer to write a printer driver to dump GD2 screens. As in the discussion during the lecture period, people had difficulty in understanding what was being requested. It is not a matter of QL compatible printers, but the fact that mode 4 and mode 8 screens can be dumped to a printer whereas mode 16, mode 32 and mode 33 cannot.

The chairman gave a noncommittal answer, but this writer says that if Quanta can afford £600 for AA sign posting, it should start spending similar sums on projects that give more direct benefit to more QL-ers.

That was Manchester. Next stop Hove. But first an explanation and then a warning.

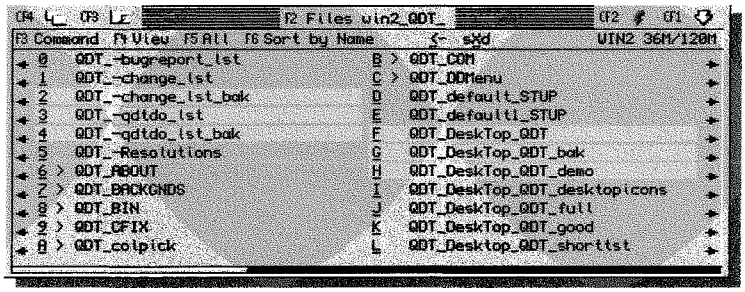
The explanation: Dilwyn's title of Custodian Pursuivant Extraordinary of the QL Archives is just a posh way of saying that he occasionally has to barricade himself in his attic to stop his wife throwing away all the computer junk.

The warning: You may read or hear elsewhere that six years ago the author of this piece was evicted from the Pond Quay Restaurant and after this year could face a complete ban following Grand Larceny of the Table Linen. These are scurrilous rumours that should be totally disregarded.

QDT Progress - FileManager

by Jim Hunkins

QDT: File Manager Feature Description



Many QDT users and potential buyers have been anxiously awaiting the File Manager component to arrive. Unfortunately for everyone, this is taking much longer to finish than had been hoped. The good news is that progress continues, despite the usual time conflicts, a few pointer environment menu structure surprises, and unexpected complexities of the File Manager design. On the other hand, the complexities are an indication of the expected usefulness when it finally gets here sometime this summer.

The work is far enough along to share a preview of the functionality that will be available. This article is actually a condensation of parts of the user manual that will accompany the module.

What is a File Manager

Basically, a file manager is a program that lets the user manipulate files on different devices such as hard disks and floppies. This includes things such as copying, moving, and deleting files, along with executing programs directly from files.

Today in the pointer environment, there are a couple of common file managers that bring this basic functionality. For example, illustrated here is the File Manager from QPAC II. This program allows most of the required basic functionality, including executing files through the integration of the Thierry Godefroy's FileInfo.

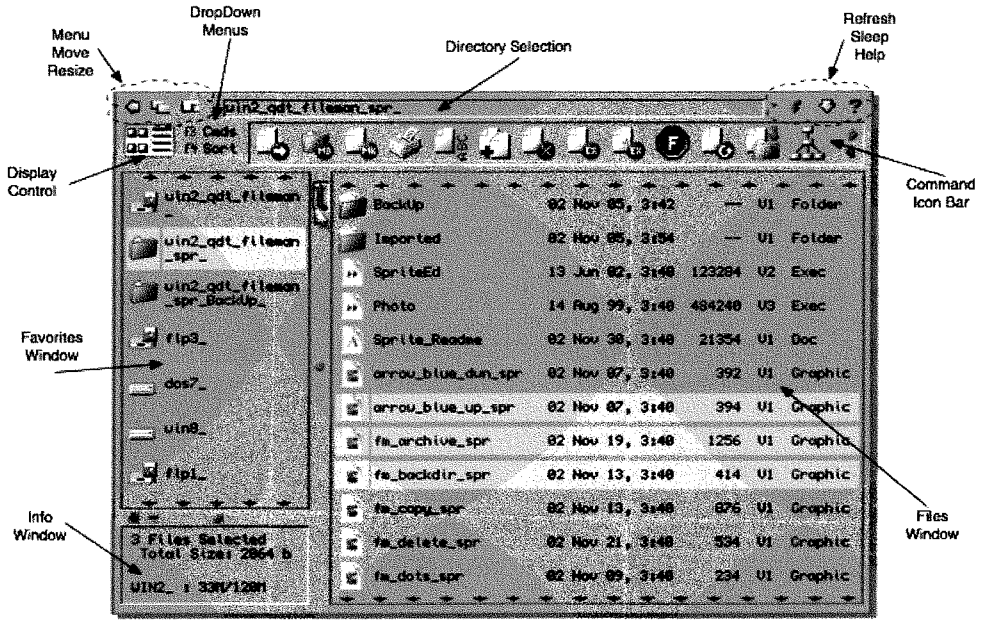
With the introduction of QDT, a modern user front end has been brought to modern day QL users. This includes drag and drop, desktop folders and icons, along with many other conveniences.

An important part of QDT is its own upcoming File Manager Object. This new File Manager will at first glance have the same functionality of the standard file managers available today but with a more modern look and feel. In addition, the new File Manager object will be fully integrated with QDT's drag and drop and folder system and add some additional QDT specific along with other general capabilities.

QDT's File Manager

Work is on going today which means some of the information and images here will likely change in the final release. The following discussion describes the major functionality and displays several screen captures of the current functioning code. The QDT File Manager will probably be released in different increments starting with what I consider a reasonable functionality (to be determined) all the way up to the full featured version.

This article should give a good picture as to what to expect when the File Manager does finally arrive. Of course, user input is always welcome, before and after the File Manager shows up.



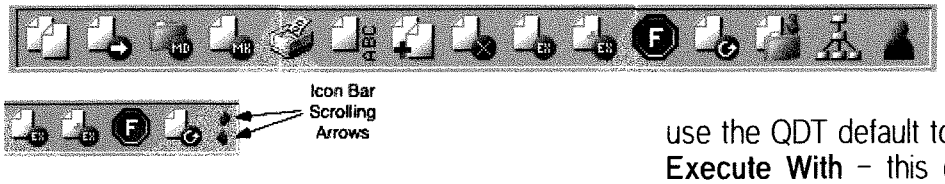
The File Manager window as shown above is broken into several sections. The top area is the control and directory section. It consists of the standard loose items for moving, resizing, refresh, sleep and the QDT specific menu and help, the directory selection window, display controls, drop down menu loose items, and the command icon bar.

The large window on the right side is the Files Window. It shows all the files in the current directory and can be setup to display the information in a wide variety of ways (detailed later).

In the left middle the Favorites Window holds directories/drives that the user will want to access on a regular basis.

At the bottom left, the Info Window shows information on a directory, drive, individual or groups of files.

Command Icon Bar



The Command Icon Bar holds 15 different icons for the most commonly used functions, as shown above. If the File Manager window is shrunk too far to show all the icons, control arrows will appear (see the smaller view of the Bar) on the right side of this Bar. One or both arrows will show depending on if there are more icons available to the left or right of the visible icons. Clicking on the left arrows scrolls the icons to the left, the right arrow scrolls them to the right, giving access to all the Command Icons in the bar, regardless of the window size. The following is a short description of each function. The descriptions follow the icons shown from left to right.

Copy - this copies the selected file(s) and/or directory to another location.

Move - this moves the selected file(s) and/or directory to another location.

Make Folder Object - this will add a folder object to the QDT Desktop containing objects for each file within the selected folder or containing all the selected files in the Files Window.

Make Executable Object - this makes an executable object on the QDT Desktop for the executable file(s) selected. If the selected file is not actually executable, QDT will try to add an object for that selection that will execute by the FileInfo or QDT default according to the file type.

Print - this prints the file(s) selected.

Edit Name - this allows for editing the name of the selected file.

Duplicate - this makes a copy of the selected file and places them in the same location but with a modified name

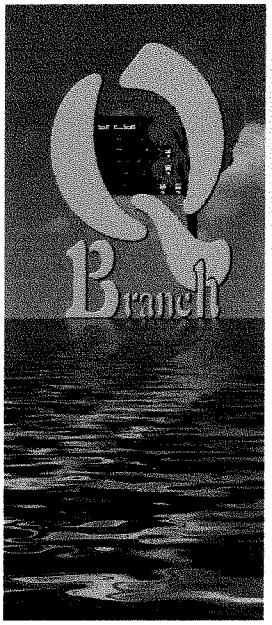
Delete - this deletes the selected file(s) and/or directories

Execute - this executes the selected file. If the file is not an executable, the File Manager will check FileInfo for the default. If not found, it will use the QDT default to execute the file with.

Execute With - this executes the selected file with the user's choice of executables - either FileInfo list or file menu selected.

Format - this formats the disk/drive.

Update - this effectively "updates" the file, changing the QL's version flag.



Feeling out on a Limb?
Reach out for QBranch
Suppliers of Quality QDOS/SMSQ products
Hardware and Software

20 Locks Hill
Portslade
Sussex
BN41 2LB
UK
Tel: +44(0)1273 486030
Fax: +44(0)1273 430501
Mobile: +44(0)7836 745501
email : sales@qbranch.demon.co.uk
www.qbranch.demon.co.uk

We can accept payment by Visa, Mastercard, Maestro and Switch. We also accept sterling cheques drawn on a UK bank. or payment direct to our bank:
Natwest
Sort code 60-17-01
Acc. 84534826
BIC : NWBKGB 2L
IBAN :
GB34 NWBK 6017 0184 5348 26

VISA MasterCard Maestro

QDT

The Ultimate QL Desktop
v1.02
£41.50 + p&p
£2.00 UK / 2.50 EU / £ 3.00 RoW
Needs SMSQ/E & extended colours to run

EasyPTR v4

Pts 1&2
Major Rewrite!
£42.00
Produce High Colour pointer-driven menus from SuperBASIC
Also on CD Rom

Text 87 £ 79.00

High Colour Patch (all systems) £9.00
ESC/P2 drivers £ 26.00 / Fountext £39.00
Typset94 / 2488 drivers £29.00 ea.

SMSQ/E and QPC 2

SMSQ/E for Gold Card / Atari / QXL / Q40
Aurora High Colour Driver included with Gold Card Version
£ 27.00

QPC 2 v 3.32
Now only £ 42.00

Upgrade Prices:
from v 3.xx Free
from v 2.xx £13.90
from v 1.xx £34.00

Delivered on CD ROM with Manual as Adobe Acrobat File

QPCPrint

Print to any Windows Installed Printer from Your QL Programs !
£27.00

Programs upgraded to high colour version

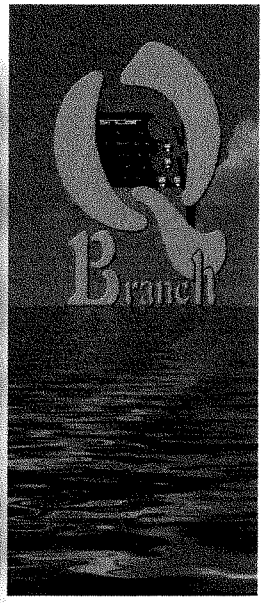
QMake / WinEd / QPAC 1 / QPAC 2 / QSUP / SuQcess / Disk Mate 5
Send old master disk and £1.60 per disk to cover costs

Free !

Paid Upgrades	
QD 98 to QD 2003	£10.50
QSpread 2001 to QSpread 2003	£10.50

Return Master Disk with order

This is the last issue of volume 10 so please remember to renew your subscription. At the moment, in the hardware box, we have a single Qubide hard disk interface for sale and three Aurora motherboards. If you call is quickly you might just catch the summer madness and get a good deal by buying an Aurora & SMSQ/E at the same time.
New Hardware is coming in all the time so call us if you need something special.



Programs

Utilities

Fifi2	File Finder	£21.00
QSup	Utilies	£30.00
QSpread	Spreadsheet	£51.00
Cueshell 2	File Manager	£15.00
QPAC 2	File Manager & Utilities Package	£42.00
QPAC 1	Calendar, Clock, Calculator, Sysmon	£22.00
QLoad/Qref	Fast load for QDOS	£15.00
QTYP2	Spell Checker	£31.00
QLQ	Printer Utility	£30.00
SuQcess	Database	£28.00
Q-Route	Route Finder	£25.00
Knight Safe3	Backup Program	£35.00
Fractal Collection	Fractals	£35.00
QCount	Accounting	£25.00

Programming

QD 2003	Text Editor & More	£ 49.00
QBASIC	QLiberator to QD Link	£ 15.00
QLiberator	Basic Compiler	£ 50.00
QD + QBASIC		£ 63.00
QD + QBASIC + QLiberator		£104.00
QPTR	Pointer Toolkit	£ 32.00
MasterSpy	Fast Text Editor	£ 30.00
QMake	Assembler Tools	£ 18.00
QMon/Jmon	Monitor - Upgrade only	£ 22.00
BASIC Linker	Basic Library Linker	£ 22.00
Disa 3	Dissassembler	£ 34.00
QMenu	Menu Extensions & tutorial	£ 16.00
Easyptr v4	Toolkits & Programming Extns	£ 41.50
Easyptr v4	Part 3 C extensions	£ 14.00

HARDWARE

We have a rotating stock of both new and second user hardware. It is best to call or email us for details of what is available.

Recycled Items
(when available)

Super Gold Card	£110.00
Gold Card	£ 45.00
Aurora	£ 65.00
QXL	£ 35.00
superHermes	£ 65.00
DiRen Keyboard Interface	£ 15.00
Qplane	£ 5.00

New Items

Aurora	£70.00
Aurora w/8301 & Minerva	£80.00
8mb Rom Disq	£98.00
4mb Rom Disq	£65.00
2mb RomDisq	£39.00
mPlane	£34.00
MCplate	£ 6.50
Various braQuets	£ 8.00
K'board Membrane	£17.50
Gold /Super Gold Card Batteries	£10,00

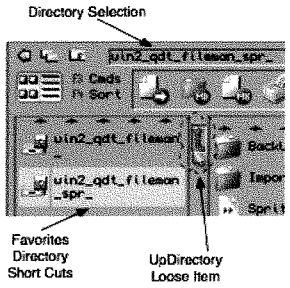
We also have a collection of standard QLs, QL Power supplies and some QL books.

Cables for the Aurora, Qubide and Super Gold Card ROMs and other QL accessories are also available from us.

Call for details

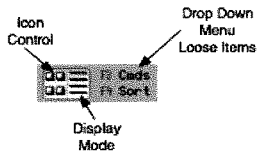
Archive – this archives the selected files into a single zip (or otherwise) archive file.
Tree – this selects between listing files only in the immediate directory or including files in all sub-directories too.
User Defined – this can have a user defined function applied to it.

Directory Controls



Setting the directory from which to display the files is done primarily with this set of controls. The normal method is the Directory Selection box which also displays the current directory.
An F2 or right click on the box brings up the standard directory selection menu. A left click on the box allows for direct user editing of the location.
If there is at least one directory level above the disk drive shown (ie: win1_firstlevel_secondlevel_), the the UpDirectory Loose Item is made selectable. Other wise UpDirectory is faded out. By clicking on the UpDirectory loose item, the selected directory will loose its outer directory level, reducing for each click until it is down to the drive only level, at which time the option will not be selectable.
Clicking on any of the listed drives/directories in the Favorites Window to the left sets the chosen favorite as the selected directory.
An option will be made to set the displayed directory (Source) or the destination directory. There will also be an option to momentarily display the destination directory in the Directory Selection box. It was decided to not show both at the same time due to the extra screen space that would be required. The final implementation details are still being worked on.

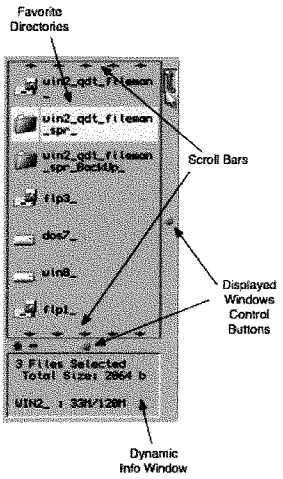
Display Controls / Drop Down Menus



The Display Controls on the left side control how the Files Window displays its information. Clicking on the Icon Control alternates between no icon, tiny icon and small icon. The Display Mode button chooses between grid, column, column with details, and line with details. See the File Windows section for additional information.

The Drop Down Menu loose items give access to sort orders and additional commands that are not in the icon bar, such as Make Directory. A modified Drop Down menu will also be available directly from the Files Window. An alternate menu is available from the Menu loose item (upper left hand corner) which is related to the File Manager itself versus files and directories.

Favorites and Info Windows



The Favorites Window is a place to keep the most commonly accessed directories. By clicking on the '+' below this window, the directory shown in the Directory Selection window is copied into the Favorites Window. When a favorite is highlighted, clicking on the '–' will remove that Favorite from the window. Right clicking on a Favorite will copy it to the Directory Selection window and update the Files Window to reflect the new directory choice.
The Favorites Window is also scroll enabled to allow showing of up to 10 Favorites even if the File Manager window is too small to show them all at one time.
The Info Window will display information on an individual file, a group of files, or directory/disk info.
The Favorites and Info Windows can optionally be hidden. By clicking on the Displayed Windows Control Button on the right side of these two windows, both windows will be hidden and the Files Window will cover the entire File Manager area. Clicking on the same button will bring them back.
The Info Window by itself can be hidden with the Displayed Window Control Button just above it, giving more area to display the Favorites.

Files Window

Type	Columns	Icons	Text Info
Grid	yes	required	file name only
Columns	yes	optional	file name only
Columns – Detailed	yes	optional	file name, minimal date info, size
Lines – Detailed	no	optional	file name, date/time, size, version, type

The Files Window shows all the files and directories contained in the Directory Selection. It can display in different formats which are selected by

the previously mentioned Display Controls. The table gives the different combinations that can be used. The icon (optional for all but the grid) is used as a quick visual reference as to file type or directory.
Left clicking on an entry will highlight it for further action and update the info in the info window, the contents of which will depend on whether more than one item is highlighted. There are two types of right clicking actions, one that is by pure right mouse click and the other uses the ALT key with the click. One will automatically either execute or open the file, depending on the FileInfo and/or QDT option and the other option will bring up a menu of actions. The choice of which activity goes with the single right click is user configurable.

QDT Integration and Other Functionality

All of the above functionality would make the File Manager object very useful just by itself. But in the tradition of going one step further, additional functionality is being built in.

Drag and Drop: fully integrated into QDT's Drag and Drop capabilities. This means that files and folders can be copied or moved between different File Manager Objects (each being a different directory) by simply dragging and dropping the selections. Future revs will also allow the File Manager window to split to support two directories simultaneously.

Building QDT Folders: a simple menu/icon bar selection on highlighted directories and/or files will automatically setup a new QDT folder and add objects for every selected file in that folder. Very useful for building document or project folders for example.

Making QDT Objects: another will make an object that can be executed if the file is executable or executed with its related FileInfo or QDT default option.

File Info: of course the latest version of FileInfo will be integrated. A future version will also include a new QDT Notebook type interface into the FileInfo configuration.

Archiving: Archiving using zip/unzip by default (user selectable) is included.

Split or multiple Windows: different windows of the File Manager can be opened at the same time and will work essentially the same way different folders work, each being its own entity but running within the QDT communication/control system (ie: drag and drop and some other upcoming capabilities).

Preset Directory File Manager Objects: File Manager object icons can be added to the desktop or any folder with Directory, Favorites, size, and file display format all preserved. An option will be included to lock these settings or update them every time the Object is closed.

Properties Notebook: all configuration settings will be done through the individual File Manager's Properties Notebook. Global default settings will be added to the QDT primary Properties Notebook.

Other options are being being considered but are still in the early concept stage.

Watch for Progress Reports

Be sure to keep an eye on the official QDT website (**http://jdh-stech.com/QDT/qdt.html**) for progress and release announcements. Since this update adds functionality to the entire QDT data structure, the update will require a full refresh of all QDT executables. Therefore direct download updates will not be available. Instead an email system for delivery will be setup on the web site along with your normal distributor update methods. This will be a free update for registered owners of QDT.

GD2 Stipples

by Geoff Wicks

The last issue of QL-Today contained a tutorial on using of GD2 colours in your own programs. This time we are going to look in more detail at GD2 stipples.

But first a short recap. To use the new colours you replace the normal keywords PAPER, INK, STRIP, BLOCK and BORDER with the new keywords WM_PAPER, WM_INK, WM_STRIP, WM_BLOCK and WM_BORDER.

The new colours are in groups:

0 - 255	QL Colours
256 - 511	Palette Colours
512 - 767	System Colours (for the experts!)
768 - 1023	Grey shades
1024 - 1279	3D effects
16384 - 32767	Stipples
32768 - 65535	RGB colours

Last time we looked at these colours using a short SuperBasic program:

```
10 WTV
20 INK 7 : PAPER 0 : CLS
40 WINDOW #2, 300,154,50,50
45 PAPER #2, 0 : CLS #2
50 WM_BORDER #2,2,1025
60 AT 2,3 : INPUT "colour ";colour
70 WM_BLOCK #2, 292,50,0,0,colour
80 WM_BLOCK #2, 292,50,0,50,colour+5
90 WM_BLOCK #2, 292,50,0,100,colour+10
100 PAUSE
110 GO TO 60
```

If you still have this program, you might like load it into your machine. If not, there is no need to worry. However, we shall be using a modified version of the program to learn about stipples.

To follow this tutorial you will need a QL system using a version of SMSQ/E above 3.00, and Aurora card users should remember the card does not have the memory to display all of the new colours.

So let's begin our GD2 stipple tutorial.

This is the magical formula the QL uses to display GD2 stipples:

`%01ssxxxxxxyyyyyy`

This represents two bytes and it is a little easier to follow if we split it into the separate bytes:

`01ssxxxx xxyyyyyy`

If the first of these two bytes were 01000000 we would have the number 64. If it were 10000000 we would have the number 128. As this is the higher of the two bytes we have to multiply these numbers by 256 to give 16384 and 32768 respectively. In other words the start of the first byte is telling the QL we have a number between 16384 and 32767 which is the stipple range.

The two letters ss tell the computer the sort of stipple:

0 = dot
1 = horizontal stripe
2 = vertical stripe
3 = checkers

In simple language dots will be in the range 16384 to 20479; horizontal stripes in the range 20480 to 24575; vertical stripes in the range 24576 to 28671; and checkers in the range 28672 to 32767. If you still have the short basic program from last time you can experiment with these.

We can begin to build up a formula more simple for humans:

stipple = 16384 + type*4096
+ colour information.

We now come to the xxxxxx. This is the first colour and there are no prizes for guessing that yyyyyy is the second colour.

There are six x's so the maximum number of colours is 2 x 2 x 2 x 2 x 2 x 2 = 64. Similarly there are six y's and the maximum number of colours in 64.

We can now complete the formula that humans can understand:

stipple = 16384 + type*4096
+ colour2*64 + colour1

Type can only be 0, 1, 2 or 3 and each colour can have a value between 0 and 63.

To experiment with stipple colours our short basic program is very similar to the program we used to look at GD2 colours in general:

```
10 WTV
20 INK 7 : PAPER 0 : CLS
30 WINDOW #2, 300,124,50,50
40 PAPER #2, 0 : CLS #2
50 WM_BORDER #2,2,1025
60 AT 1,3 : PRINT "              "
70 AT 2,3 : PRINT "              "
80 AT 1,3 : INPUT "1st colour ";colour1
90 AT 2,3 : INPUT "2nd colour ";colour2
100 colour = 16384 + 64*colour1 +
    colour2
110 WM_BLOCK #2, 292,30,0,0,colour
120 AT 5,5 : PAPER 0 : INK 7 : PRINT
    colour
130 WM_BLOCK #2, 292,30,0,30,colour+4096
140 AT 8,5 : PAPER 0 : INK 7 : PRINT
    colour+4096
150 WM_BLOCK #2, 292,30,0,60,colour+8192
160 AT 11,5 : PAPER 0 : INK 7 : PRINT
    colour+8192
170 WM_BLOCK #2, 292,30,0,90,colour
    +12288
180 AT 14,5 : PAPER 0 : INK 7 : PRINT
    colour+12288
190 PAUSE
200 GO TO 60
```

If you already have the original program loaded into your machine, you can save time by modifying it.

First of all there is a slight change to the size of the window defined in line 30. As we are now working with two colours there are some additional input lines (60 to 90). After the input information there is the formula for calculating stipple colours (100). Then there is an additional WM_BLOCK command and also changes to the size of the existing blocks. Finally after each block command there is a line to print the stipple number.

We now have a program for viewing stipples, but there is still a practical problem of knowing which colours are actually used in the stipples. They are the first 64 colours in the 256 colour palette. Fortunately they have been given names and you can look these up in the table to the right.

So if we want a red and white stipple, we could say colour1 = 1 and colour2 = 3. Try inserting various values in the short program you have typed in. You will find you can get some interesting backgrounds for your programs by forming stipples from two of the pastel shades.

0 Black	22 Orange	44 Steel blue
1 White	23 Lime green	45 Dull pink
2 Red	24 Apple green	46 Brown
3 Green	25 Bright blue	47 Khaki
4 Blue	26 Mauve	48 Dusky green
5 Magenta	27 Peach	49 Dusky blue
6 Yellow	28 Light yellow	50 Midnight blue
7 Cyan	29 Light blue	51 Plum
8 Dark slate	30 Sky blue	52 Dusky pink
9 Slate grey	31 Rose pink	53 Buff
10 Dark grey	32 Pink	54 Avocado
11 Grey	33 Beige	55 Dull turquoise
12 Light grey	34 Pastel pink	56 Dull blue
13 Ash grey	35 Pastel yellow	57 Faded purple
14 Dark red	36 Pastel green	58 Cerise
15 Light green	37 Pastel cyan	59 Tan
16 Mustard	38 Pastel blue	60 Grass green
17 Dark green	39 Pastel rose	61 Sea green
18 Sea blue	40 Brick	62 Ultramarine
19 Dark blue	41 Light khaki	63 Deep purple
20 Purple	42 Dull green	
21 Shocking pink	43 Dull cyan	

Brian Kemmet writes:

Hello Geoff!

Just a quick line on the reply you gave to Robert Hartung on setting up QPCPrint:

It would be worth mentioning that on the QPC SER/PAR dialogue page, you have to make sure that the Radio button marked "Printer" is selected, otherwise the "Filter" box remains greyed out, and cannot be ticked. It took me a while to realise this. I had the same problem as Robert. I left the tick in the box marked "PAR 1" - and couldn't figure out why I could not use the "Use Filter" box, consequently QPCPrint either printed garbage, but mostly did nothing!

Hope you don't mind me pointing this out, and I hope you can pass this on to Robert, who is probably sat in front of his monitor scratching his head in bewilderment!

Steven Poole writes:

SPRING CLEANING

Dear Geoff!

Now its nostalgia time again, requiring a little meditation and some chucking out of rubbish. Where shall I start?

Back in the sixties, whilst studying metrology, (Quality control for English Readers), men in black came to the Plymouth Polytechnic asking if anybody wanted to study pneumatic logic circuits. One look at the binary coding gave me immediate intellectual indigestion, so I declined to take up the offer. Then at work (for a large American electro-engineering concern), they started computerising everything. You may think you have never worked with punch-cards? Think again what are clocking-in cards? Everything was analysed, timed and Taylorised, and what was a skilled job became a hole-punching race totally inadapted to the subtleties of real conditions. But it pleased shareholders who believed the hype that computers were intelligent entities that would solve all their problems. But if anything went wrong, it was always the fault of the computer. The machine was of course accompanied by a whole host of computer engineers who were already promising to robotise everything, saying that job losses would result in the creation of abundant leisure for everybody. In fact, most of the time numeric machine tools were inoperant because the programs were so difficult to understand and adapt. Of course the mainframe company made huge amounts of money and it was the workforce that got all the inconvenience down the back of its neck. So I quit, and there being no other quality control work in Plymouth, went and got a job in a commercial bank.

But no luck there either, as they had just swallowed the hook and installed a giant computer. The accounts side was not too bad, but they had bought a word-processing system for the typing pool. So instead of writing perfectly-simple well adapted letters to clients, we had to choose between one of 30 or so pro-forma models and try to customise a few words in the gaps to fit. Needless to say, the results were miserable, but the shareholders were happy as they believed they had acquired the latest competitive technology, and the workload on typists went up as the rate of correspondence increased to make up for the inefficiency. And so it went on, job after job, Taylorisation. Yet I went to evening courses to study Taylorisation, and learned that Taylor came to his conclusions with Phillips, about whom we never here one word spoken. Why do I mention Phillips? Because without his contribution to work study, computers lead to the reduction of people into numbers.

Most readers will be sufficiently old as to remember Doctor J.Bronowski's excellent BBC documentary 'The Ascent of Man' in the sixties, in which he kneels in a bog in the wastelands of Auchwitz Concentration Camp holding the cinders of his ancestors in his bare hands and says: "I beseech you in the bowels of Christ not to teach people to use computers". What he was referring to was main-frame mentality: Just before the war, a leading American digital-processing company sold a complete nationwide census system to the Nazis, which enabled them to keep tracks on people, by numerising all their personal records. People became numbers, and numbers were written off at a stroke, millions at a time.

In 1981 IBM engineers started looking at the handful of Apple computers that such people as radio hams were breaking their piggy-banks to do their quasi-mystic logic calculations on. They came up with the IBM Personal Computer prototype which their commercial staff dismissed as being only suitable for electronics engineers and

suchlike, and which it was thought would never be compatible with IBM mainframe methods. So IBM put out tenders for an operating system, which should logically have been CP/M but was snapped up by Bill Gate's start-up 'Microsoft', for which he bought up the coincidentally-named QDOS operating system. Bill Gates was a good businessman, he produced his DOS before the IBM deadline! So the PC was an IBM system, profiting from the huge reputation of Big Blue's quasi-world wide dominance of main-frame computing.

Around 1984 a large handful of companies had developed their own operating systems and started to produce home computers, especially bearing in mind the phenomenal success of Sinclair's ZX series of computers, which proved that with a little logic anybody could write efficient computer programs, and did, and thousands of programs were published in a whole range of specialist magazines. So why didn't these home computers prosper? The simple answer is lack of flexibility. Whereas IBM PC's forced the competition to make clones if they hoped to compete, again because of the mainframe reputation of corporative interactivity. So PC's could be built by buying standard subcontracted parts and competition forced prices down. And a good thing too, as PC's were very expensive, only affordable by medium-sized companies, costing several month's wages for the average man.

About this time I was using programmable hand-held calculators in my surveying job and reading up about computers which could possibly reduce my work-load. I went to a professional computer fair at Paris and examined the plethora of machines available. One appeared suitable: A Husky Hunter sold by the British Army, being shock and water-proof and costing 3000 pounds. Would they sell me software? NO! Military secrets! Would professional Surveying software run on it? Yes! But don't expect it to adapt to your exact needs as a) it will cost you an arm and a leg, and b) you will have to pay an engineer to modify programs for you, which will take a couple of months and an extortionate salary!

So what use is a Husky Hunter? Thus came the surprise: The Husky contains an excellent Basic which will allow you to easily write your own programs at little cost!

Now I had always read that the only valid reason for choosing a computer was that the right software was available! Yet now I was instructed to learn programming!

At that time Sinclair announced his plans for a Quantum Leap in Programming Power: The QL. Lord Clive wanted a machine for the small entrepreneur, and that is why it was offered with the four Psion programs which would cover any conceivable business need, plus a very powerful basic language (with the neat structure of Pascal yet the flexibility to permit interactive testing via an interpreter!) Now, say you have a green-grocery shop: You will have about an hour and a half a day to do computing, so Sinclair gave you a tutorial course so you could teach yourself coding and software operation in your spare time at no cost. Why bother to Compute at all? Because you can use computer power to do what computers do best: Forecasting. Once you have entered your Data, you can change values anywhere and rapidly see how your finances can be improved. Sinclair produced a machine affordable for the small entrepreneur. It had microdrives because these were impressively faster than the tapes on rival machines and much cheaper than floppy disk units which at that time cost more than the QL itself! In fact the machine had so many advantages over the competition that not only small businessmen but also all manner of computer enthusiasts immediately ordered it: An avalanche that Sinclair was not ready to satisfy. Even the French Government was ready to order millions for their 'Informatique pour tous' schools program, but non-respect of delivery dates ruled that out. Sinclair too had a reputation of success, as did IBM, and also conceived the QL to allow competition from sub-contractors to bring down the price of extensions. But the Press had no QLs to test: It is evident that most of the press reviews were written by journalists who had never got their hands onto one. After buying the unique test-machine from the French Importer, I spent many months examining press statements about the QL: They were all obviously false and based on hearsay. The one problem I did find with my machine was that the rubber rollers would ride up the drive shafts leaving tapes out of contact from the read-heads, but a letter to the suppliers soon overcame this.

Anyway, the Press had collectively decided that the QL was not a 'Professional' machine, quoting the example that the keys were not adequate for typists. But as a greengrocer, are you able to type at 120 words per minute? The QL was able to perform everything I demanded of it, but my company boss was under the charm of IBM mainframe policy (and the directives of tax authority constraints) and so bought an expensive PC,

with the imaginable hassle that that brought, and was finally unable to produce the output of the QL.

PCs have come a long way since then, and as my brother points out, you can do just about anything with windows if you know how but he is an electronics engineer who has followed the development of PCs since their beginnings and spends all his time on them. He is the very example of what IBM commercial people were saying originally: PCs are for very educated people and not for Joe Bloggs. Yet the myth of computers 'solving all your problems' has meant that they have become a status symbol and that everybody who can reasonably afford one won't be seen dead without one, and the very latest model at that. So PCs have sold all over the world in millions, but how many people really know how to use them? Most (educated) people I know have to get their PC set up by a technician, who has to be called regularly to see why it is not working as expected. This is of course mainframe policy: You need specialists to write, adapt and teach you how to use software. PCs have been sold to about half the population of the developed world: (The above-average IQ part). Now the lower-IQ part also want computers: To run CDs, DVDs, do slideshows and to send emails to their family. But they can't understand all the arborescent complications of windows, and want simple front-ends that let you do anything with just two clicks of the mouse. Old people want computers for looking-up topics of interest on the internet, and children want computers to play games on. Yet computers, for all their potential power are under-used: On the internet the most frequently searched word is sex. The most money is made by companies from computer-games that reduce people to numbers: Exterminate without hesitation. Strangely, there are very few games which include a police force to ensure the morality of user behaviour. So Doctor Bronowski was right: Mass-produced computers are anti-social. This is because of the corporate main-frame mentality that has reduced people to numbers in games.

All that is pretty depressing stuff. Most computers spend ninety-five percent of their time generating wait-states. You sit down, you turn your computer on and it waits for input. Yet Science cruelly lacks computing power: If you have broadband you will probably leave your computer connected 24 hours a day. So why not donate your wait-states to scientific research organisations that will use you computer whenever you are not multitasking?

Now 50% of the population are inadapted to mainframe PC policy, and hundreds of millions of poor people in the world can't afford a months (European) salary to buy one anyway.

It just so happens that the MIT, (Massechusetts Institute of Technology), one of the world's major research players, has revealed plans for a 100-dollar portable computer that is simple to operate. They plan to sell 100 million a year, including to third-world countries, which probably means no window's bloatware, just a plain clockwork power-unit, a RW-DVD, flash memory, a simplified Web protocol, keyboard and small screen. So what will be the Operating system? Just think about it SMSQ/E & QDT would do fine But you will say, where's the PC compatibility? Why, with QPC your QL is totally compatible! Just take your DVD and QPC to your local library and away you go! Now critics will tell you that third world peasants can't afford 100 dollars. Yes, but average villagers could pay one dollar each, (which is a day's wages), and each village could immediately have access to a quantum leap in technological progress. So a 100-euro portable computer is perfectly feasible with current technology, but MIT haven't decided who could manufacture it.

Finally, and if you are still reading this, here comes the point of this article: There has been some discussion of building a 100-euro portable QL, with specifications similar to those of the MIT machine. But plans are not settled, neither by MIT nor the QL Community. More on that elsewhere in the magazine. If you have any suggestions, why not write to the Editor who will forward your comments to the appropriate instances.

Personally, I consider that such a machine would be economically viable. All we need is to get a good highly rational design and get it right first time. A suitable manufacturer has already been suggested, so the only thing we now need is determination, and of course a financial backer...If only Alan Sugar would support the QL other than in allowing us to use his patents!

2006 is likely to be a very interesting year for the future of the QL.

Editor's note: To avoid misunderstandings: a QL cannot run QPC - and does not turn it into a PC compatible system. The 100 EUR machine would have to be based on a native SMSQ/E implementation, as just the license price for Windows would make it impossible to produce and sell at this price.

Comments are most welcome!

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/scrollock 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 Thorn-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 requd). Send disk plus SAE or two IRCs

MKI..£40 (£41/£43) **MKII..£65** (£66/£67)

MINERVA RTC (MKII) + 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 MKII 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

PSION as never before

by Geoff Wicks

The QL and the Psion programs grew up almost hand in hand, but whereas the QL quickly developed into an adult, the Psion programs scarcely reached adolescence. Show an original black box user the present day QL and he would be in unfamiliar territory. Load a copy of Quill and he would feel at home. Or would have done until recently. In the last six months the Psion programs have changed almost beyond recognition.

From the start the QL was promoted as a business computer. For this reason about 18 months before the QL was launched, what was then a small company called Psion was commissioned to write the business software for the computer.

Psion was already well known to Sinclair as they had produced several programs for the Spectrum including Vu-3D, Vu-file, Vu-calc and some games, but the QL Psion suite was probably the biggest job they tackled for Sinclair.

The Psion team were chairman David Potter, technical director Charles Davies and software director Colly Miles. Two other workers were Martin Stamp and Martin Brown.

Psion had the problem from the start that they were developing software for a machine that did not exist. Most development work was done on a VAX minicomputer with a concept that it could then be ported to various operating systems. Easel was a problem because the VAX lacked high graphics capabilities and in this case some development work was done in monochrome on a BBC computer.

The Psion software was written in less than a year, but there was a scare at Christmas 1983 when the Psion team received their first QL. It took them 24 hours to get the software operating on the machine. Nevertheless they were able to demonstrate versions of the program, albeit bug ridden, when the QL was launched about three weeks later.

More than 20 years on, if last year's Quanta survey is accurate, 65% of QL users are still using Quill and up to 71% Abacus. No figures are available for Archive or Easel use, but there is anecdotal evidence that many QL-ers regard Archive as being better that PC database programs because of its powerful programming language.

With the exception of Archive, the Psion programs set high standards of user friendliness. In particular Quill performed well as a WYSIWYG word processor. At the time the state of the art PC word processor was WordPerfect whose commands could only be accessed through numerous combinations of the Function, Shift and Control keys. However, once WordPerfect was superseded by more modern word processors using vector fonts, the limitations of Quill became more apparent.

Perhaps the greatest tribute to Quill comes from the QL world where the two alternative word processors, Perfection and Text87, were clearly modelled on it. Both of these word processors have more facilities than Quill, but neither had the soft-hyphening and decimal tabs.

When the Thor computer was released in 1987 it came with the Psion programs bundled together in the Xchange suite. This had slight improvements to the original programs. Originally the Xchange suite could not be run on black box QLs, but some years later its was modified to allow this, and has since become the preferred version of the Psion programs for many QL users.

And there, apart from some minor tweaking of Xchange, the development of the Psion programs stopped. Psion was happy for the QL community to continuing developing the suite, but unfortunately could not actively co-operate. The tapes containing the source codes were still available, but the computer to run them on was not. While the QL went on to develop high resolution screens and more colours, the Psion software remained much the same as it was in when the QL was launched in 1983. We were told there was no possibility of further improvements to the Psion suite.

But the QL has a history of the impossible suddenly becoming possible. During the last six months two people, Marcel Kilgus and Roger Godley, have been independently working on major changes to the Psion programs.

Marcel has been updating the Xchange suite and has introduced 4 major changes. The most welcome is increasing the working area from 512 x 256 to 512 x 512, a facility long wanted by users. He has also made changes to the colour drawing routines and given Quill's status area a slightly different font colour to differentiate it from the text. The programs now accept the DOS as well

as the QL file separator and there have been improvements to Archive to facilitate printing.

Roger Godley's work is less well known, partly because it is still at the beta development stage and partly because his work is only available via Quanta. He is concentrating on the individual Psion programs and is modifying them for high resolution screens and GD2 colours. He started work on Quill, which Marcel tells me is the most difficult of the Psion programs to modify. Patching the new colours is fairly simple, but locating all the places to patch them is very complicated especially when you have also modified the screen size.

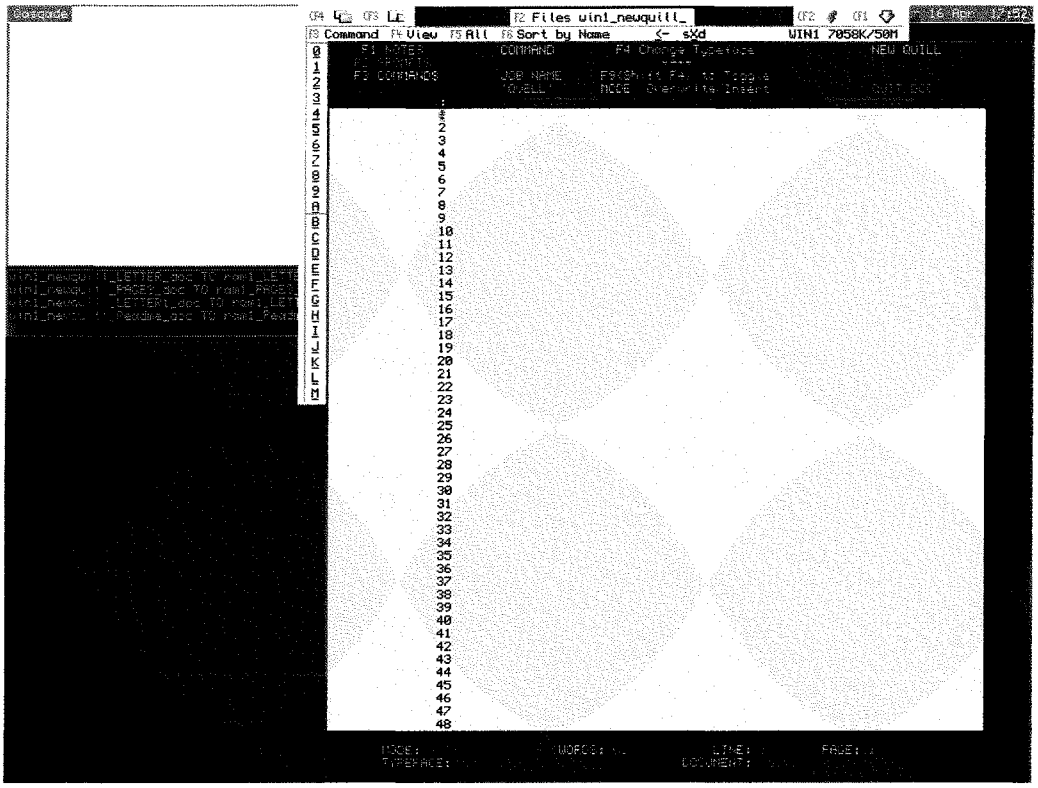
Roger's Quill is unlike any you have ever seen. It comes in two shades. Vellum has a light yellow background with a choice of blue or black text, and Velvet has a dark black background with a choice of yellow or light blue text. In both versions the text area stands out well from the status area. It is possible to run several copies of the new Quill at any one time and each new version has a slightly different colour for the working area.

The screens are not resizable, but the programs come in two versions for different screen sizes. Up to 64 lines of text can be fitted into a page. Because they use huge quantities of memory Aurora users may find them disappointing, and as they are still beta versions, you have to be prepared for some minor bugs and a lack of documentation. However any Quill user will soon adapt to the new version and the more advanced and technical users could even try modifying the colours.

Two people who have tried the program are Malcolm Cadman and John Gregory. Malcolm reports that he was able to run 11 copies of the new Quill in a 16Mb version of QPC2. John comments, "*Although ... recognisably Quill it has a pleasing modern feel to it*". I am no longer a Quill user, but I found the colours of Vellum version much pleasanter to use than the traditional Quill ones.

At first sight Roger's versions of Abacus and Archive seem slightly less familiar than the original versions, but this is probably the psychological effect of seeing their size on the screen as much as anything else. The "standard" version of Abacus supplied to QL Today defaults to a display 66 rows by 10 columns. The "widescreen" version has a massive default of 70 rows by 20 columns. In one of the illustrations to this article you can compare the size of the widescreen version with that of traditional Psion.

The standard version has a blue frame on a black background and uses a cyan ink. The widescreen version has a light green frame on a silver grey background and uses black ink.

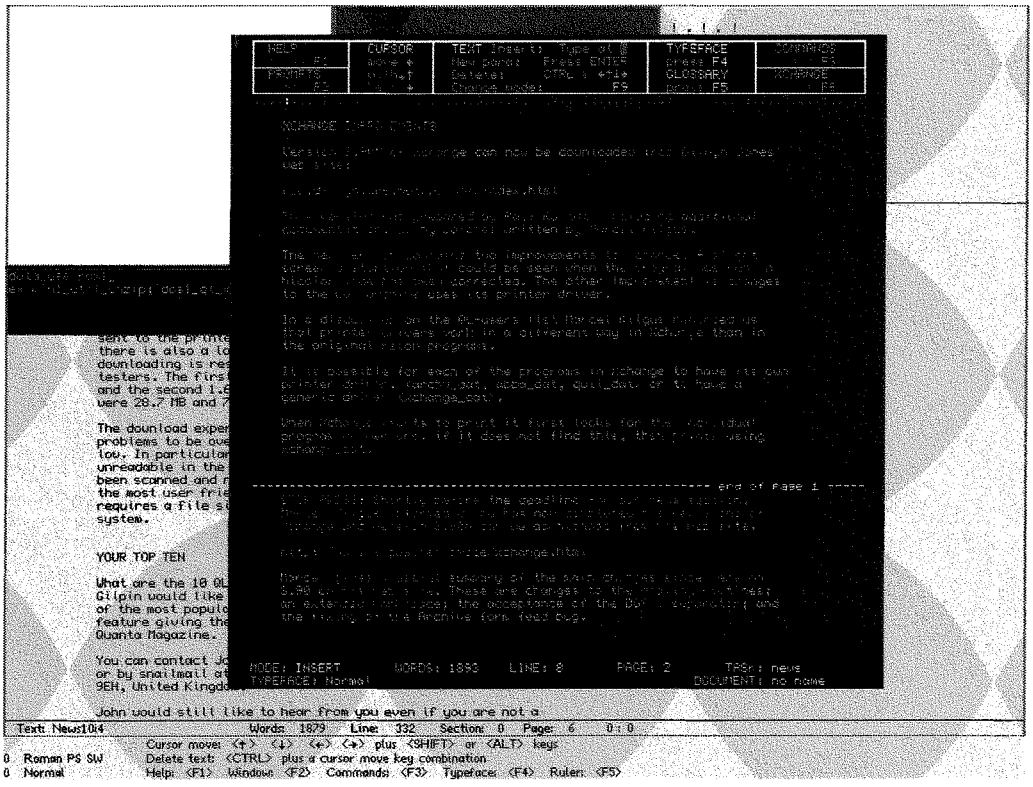


Archive is a more difficult program to describe as so much depends on how a user programs it for his own use. I tested it using the "look" and "display" commands on one of my data bases. The 800 screen version displayed the commands in a blue ink on a dark green background - not a good colour combination on my monitor, although it looked better on the computer projector at the Manchester show. The data base, itself, is displayed with blue text on a light yellow background.

The 1024 screen version has a much better display of the commands with black letters on a dull gold background. In this case the background of the data base is a dark green with black letters. An interesting innovation is that the contents of the database are printed on a dull gold background, which highlights them in comparison to the field names.

The one Psion program I have not mentioned is Easel. This has not been forgotten and Roger Godley is currently working on it.

At the moment the programs are only beta versions and only available to Quanta members. Roger Godley clearly sees his work as being a Quanta project and thus they are unlikely to be available via the internet or other software libraries. This may disappoint some QL-ers who are not Quanta members, but then a year's subscription to Quanta would be a small price to pay for software improvements of this quality.



DIY Hardware Add-ons for your Sinclair Computer - Part I

by Phoebus R. Dokos, B.Sc

Video Out for the ZX Spectrum/ ZX-81

DISCLAIMER

These projects have been tested and found to work properly. However neither I nor QL Today can be held liable for any damage, direct or indirect, that is inflicted upon your machine whether by faulty installation or omission in these articles.

This issue we start a new series that looks at add-ons and improvements that a regular user with some soldering experience can perform at a relative small amount of time and at a manageable cost. Although the first couple of projects concern the ZX81, ZX Spectrum/ 128K/+2/+2A and +3 I should take a moment here and remind you to look on the cover of QL Today. It is after all "The magazine about QL, QDOS, Sinclair Computers, SMSQ"!
We will look at two small and one larger project for the ZX-81 and ZX Spectrum. Here I should want to thank Sami Veehma for his valuable input and hardware that made these possible.

A. Projects 1 & 2

DIFFICULTY LEVEL: EASY

Composite Video Output for the ZX-81 and ZX Spectrum / Spectrum +

IMPORTANT:

THESE MODIFICATIONS DO NOT APPLY TO VHF VERSIONS OF THE ZX-81 OR TIMEX SINCLAIR 1000 AND 1500! IF YOU ARE NOT SURE WHAT YOU HAVE, LOOK FOR THE PRESENCE OF A "3/4 CHANNEL" SWITCH. IF THIS EXISTS, THEN YOU HAVE A VHF VERSION. IF NOT, THEN PROCEED.

Introduction

Most of us that have at least opened our ZX Spectrum, ZX-81, or our QL, will have surely seen ASTEC's modulators sitting proudly behind the aerial socket and producing the output we all see on our television sets (if we still use these that is!). We have also seen how annoying and fuzzy the picture is. The simple answer to improvement (simple because there are more complex ones for sure) is to bypass the modulator and feed our telly (or VCR) with the Composite Video signal that feeds into the modulator (figure 1). Right? Wrong!

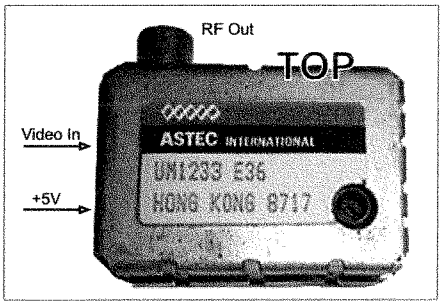


Figure 1 - ASTEC RF Modulator

For reasons only known to Sinclair engineers, the video signal is anything BUT standard and so it has to be brought within acceptable ranges as to:

a. Protect our Video Display equipment and
b. Provide the best possible quality picture, as to reduce our eyestrain
c. Make it compatible with "quirky" television sets/VCR's and monitors.

Now it is possible that many of you have sought the quick fix of just soldering a female RCA connector to the video signal input and ground (in this case most likely the metal cover of the ASTEC modulator) and fed it to your television or monitor. However this solution is not only dangerous to both your Spectrum/ZX 81 but also to your display equipment and image quality.
Note here that I do not separate the machines as it is NOT needed so far. You will be informed later on which procedure suits which machine.

There are two ways to perform this task. Both involve a complete removal of the modulator but one proposes its replacement with only the parts needed (not recommended as other procedures will be needed to secure the video connector) and the other the use of its shell (removal of the circuit board therein and replacement with three components) so that everything looks as if it were made by the factory (neater!).

Tools needed

- Low power soldering iron (such as Antex 12W or similar) and solder of course
- De-soldering braid and/or de-soldering pump (If you have one of course, use a de-soldering iron)
- Small Flat screwdriver (to be used to uncover the top of the modulator)
- Small cutter -or even scissors if you don't mind the edges becoming dull-
- Small Philips screwdriver
- Static-Safe IC foam -or- polystyrene covered with aluminium foil.

Parts Needed

ZX-81:

- Transistor: BC547 or equivalent
- Resistor: 100O
- Capacitor: Electrolytic 220 F 16V(Radial preferred as polarity is easily identified but axial would do too)
- Heat shrinking insulation (if available)

ZX Spectrum:

- Transistor: BC547 or equivalent
- Resistor: 75O
- Capacitor: Electrolytic 100 F 16V (Radial preferred as polarity is easily identified but axial would do too)
- Heat shrinking insulation (if available)

Installation

First we need to make sure that we as well as our workspace are static-free. To loosely quote from the ZX-81 KIT Assembly Instructions: "if you do not have a Static discharge strap, remove your shoes and avoid carpets!". Then we can set to remove the modulator from our Speccy or Zeddy (sic!).

In order to do that, first locate and unscrew the screws at the bottom of your Spectrum and ZX-81. Please note that ZX-81s have hidden screws underneath their rubber feet. Take special care removing these as, since they are really old by now, they can crack and break. Once the cover is off, remove the tails of the membrane (and be extra careful with the ZX-81 as replacement membranes are almost impossible to procure now) and set the cover aside together with the screws. If you are modifying a Spectrum+, you can also choose to remove the plastic feet and their plastic "retaining springs" as they tend to fall of and get lost! Also if you have a ZX Spectrum+, slowly remove the reset button from the left hand side (it slides in and out very easily). It will be hanging there so make sure it doesn't get in your way. Both ZX-81 and Spectrum have two securing screws that fasten the PCB to the bottom of the case. Remove these as well and set aside. Make sure you do not mix them with the case screws as they are of different lengths and a wrong screw can crack the Spectrum's or ZX-81 cheap case (Trust me.. Once bitten, twice shy!). Now it is also a good time to remove the ULA from its socket and place it on the static safe IC foam (or the polystyrene makeshift IC carrier). You can save yourself a lot of grief that way. The ULA is the only component of both the Spectrum and ZX-81 for which new replacements are almost impossible to find.

Next, it is time to fire up our iron and flip the computer's PCB so that the component side faces down. With PCB flipped, the ZX Spectrum's Modulator is located on the top right hand side and the ZX-81s on the right hand side towards the top.

Next we should de-solder at four points:

a. The +5V input to the modulator (First wire going into the modulator from the bottom left -bottom right if the PCB is reversed-, the top considered to be the side with RCA connector - see figure 1) - The insulation at the point of entry is usually brown.
b. The video input to the modulator (Second wire going in the modulator) - The insulation at the point of entry is usually white.
c. Ground and left retaining tab of the modulator and
d. Ground and right retaining tab of the modulator

These last two are easily recognizable by the larger blobs of solder on them and the obvious mesh track pattern around them.

WARNING

MAKE SURE THAT YOU DO NOT OVERHEAT THE PCB DURING DE-SOLDERING. IF YOU DO SO, YOU MAY DAMAGE THE ULA OR OTHER COMPONENTS.

Once the solder has been completely removed, carefully straighten the two modulator retaining tabs with your long-nose pliers and pull the modulator gently out of the PCB.

Move the PCB out of the way and concentrate on the modulator. Although it is possible to retain both the modulator PCB and modify its output to Composite Video only, I would not recommend it. There are several reasons for that:

- There will be little "wiggle" room for you to add the components.

- With the presence of other components (and especially if you do not insulate the components to be installed, the possibility of a short circuit increases dramatically
- Interference may be introduced if you attempt to provide both Comp. Video AND RF from within the modulator

Next, we need to remove the top of the modulator. This is accomplished by using the flat screwdriver and gently loosening the locking tabs of the top cover. Once inside you will notice 6 areas where de-soldering (and cutting) is needed. Looking from above there are two areas on the bottom, two on the top and the two input lines coming from the left and the RCA connector. The Video line can be completely removed (as it can be replaced by feeding the base of the transistor through the video in hole and use that, however the power line is pretty much impossible to remove without damaging either the insulation or the line itself, so I recommend cutting that as close to the modulator's PCB as possible. De-solder the points where the ground lines connect to the modulator's case as well as the line to the RCA connector's centre socket and remove the PCB completely. If you have taken care when doing that, you could potentially replace it in the future (but why you would want to do that is something that I wouldn't know!).
Before you continue, make sure that you check with the transistors' manufacturer so that you can be sure which pins are the collector, base and emitter. Sometimes (although not too often), these change between manufacturers and you can end up with a non working project (and Spectrum or ZX-81).

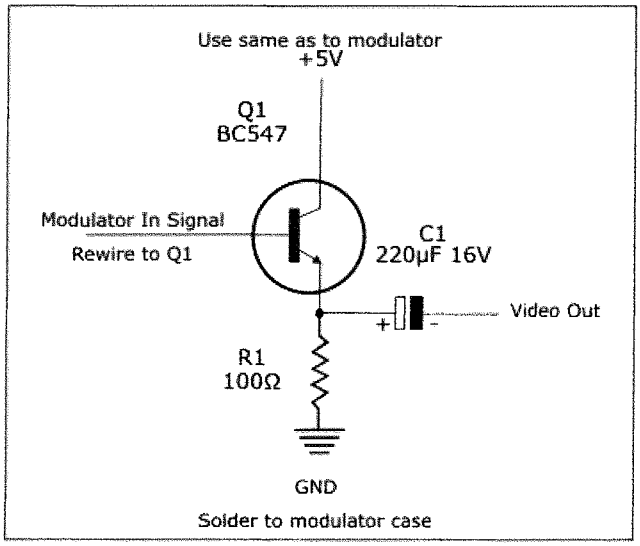


Figure 2 - ZX-81 Video Fix
© 2002 Sami Veehma
Modified for QL Today - © 2005 Phoebus Dokos

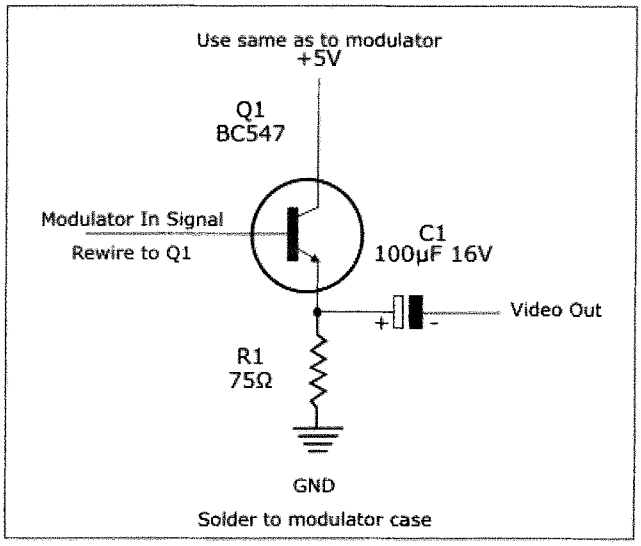


Figure 3 - ZX Spectrum 16K/48K/+ Video Fix
© 2002 Sami Veehma
Modified for QL Today - © 2005 Phoebus Dokos

Next using figures 2 and 3 as a reference do the following:

1. Cut the transistor's collector pin, short enough, pass the heat-shrinking insulation over the already retained +5V line and solder the collector pin with the +5V line. Then pass the heat shrinking insulator over it and heat it up to shrink and provide both insulation and mechanical stability
2. Cut the negative pole of the capacitor and solder it to the RCA connector's centre pin (the one insulated from the case)
3. Solder the resistor to the transistor's emitter AND the positive pole of the capacitor (thus making a "T" joint)
4. With care not to break any connections already made, solder the free side of the resistor to the ground (the case of the modulator) - Allow for some movement of the whole assembly as the last step will require some flexibility
5. Finally slowly feed the base of the transistor through the video in hole (you previously removed the line from). It should be long enough to solder back to the original Spectrum or ZX-81 PCB hole when the modulator is placed back. Make sure that nothing except the resistor's one end touches the modulator's case and that all joints are in order.

That is it for the modulator assembly. Replace the cover as it acts like shielding and get ready to solder the modulator assembly back to the computer's PCB. First, put the modulator retaining tabs back through their PCB mounting holes and bent them slightly. Next, while maintaining slight

pressure on the modulator assembly to make sure that there is no clearance between the bottom of the modulator and computer PCB, solder them both back on the PCB. Inspect the assembly for a "cold" solder joint and for any mechanical "play". If everything seems fine, flip the PCB over, pass the +5V and Video In lines (the latter being now the base of the transistor as mentioned above) through their respective holes on the component side, flip the PCB back and solder it in place. Then replace the ULA from its carrier to its socket - Please make sure that both notches on the ULA and the socket point to the same direction! Before we continue putting everything back together, it is time to...

Test the results

And we do that by connecting our Composite Video Cable to our display and to the modulator (which of course isn't a modulator any more) and supply power to the computer. If you see the - 1982 Sinclair Research message -or- the inverted block K cursor of the ZX-81, you are ready to put back the computer together! If not well go back and inspect the connections you made as well as the positioning of the ULA (Particularly check if pins are bent or out of the socket and if the orientation is correct)

Next retrieve the screws for the mainboard and screw it on the bottom of the case, refit the feet, springs and reset button (in the case of a ZX Spectrum+), gently but firmly reinsert the keyboard membrane tails back in their respective sockets and replace the top cover.

This project does NOT fix the annoying dot crawl effect on the ZX Spectrum, but does a lot to improve image stability and reduce eyestrain for both Spectrum and '81.

Eindhoven QL Meeting - March '06

by Jochen Merz

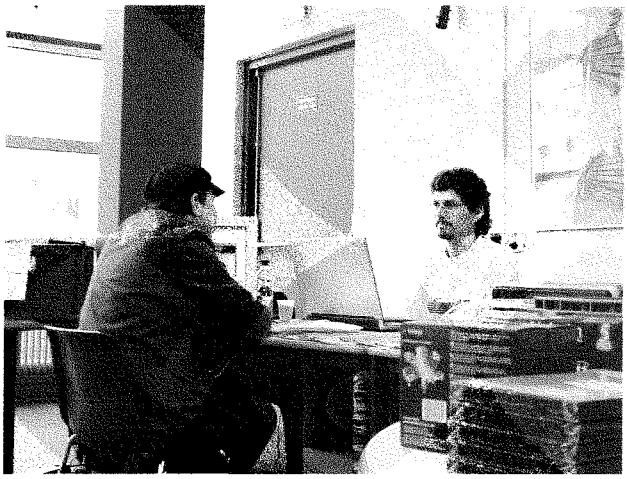
The usual bunch of QL dealers were present at just another international Eindhoven QL meeting: TF Services, QBranch, Geoff Wicks, together with more visitors from the UK.

During the day, the usual customers and friends from surrounding countries arrived ... but it was a rather quiet meeting. The queue at the coffee corner was larger than the queue at the JMS desk, as you can see below.

On the other hand, there was plenty of time for chats, as you can see on the picture in the right-hand column.

Although this was supposed to be one of the "bigger" meetings, it was not advertised like the success QL 2004 afterglow last year - also, talks etc. were missing.

Jens Wildgruber demonstrated his own program to solve Sudoku puzzles - I tried one myself ... very good! Maybe something for another cover disk, I wonder...?



Stephen Poole plays a game of Battleships with Jochen but finds it rather one sided.

Well, it was a pleasant meeting, as usual. I am glad we have the well-known, very reliable venue. Pity some shows are not so well visited. What do you feel is missing? Please let us know!

Do you forget about the meetings without a mailshot in advance (which are very expensive nowadays, as the postage gets more expensive every year too). Do you prefer fewer shows?

RWAP Services NEWS!

New Websites!

We are proud to present our new websites! Starting September, we have separated our business in three separate websites.

You can find them at:

<http://www.rwapadventures.com/> (Adventure Games)
<http://www.rwapservices.co.uk/> (General Site) and
<Http://www.rwapsoftware.co.uk/> (Software)

New Products!

Q|W|O|R|D 1.0 NOW WITH DIGITAL SOUND ON QPC2!

The wait is now over! Q-Word version 1 is finally available!

Platforms:
QPC/QXL, Q40/Q60, Aurora (with SGC)

Prices:

All versions without P-Word	£20.00p
All versions with P-Word	£30.00p

Notes:
Q-Word DOES NOT require SMSQ/E with GD2 support -OR- SMSQ/E at all on the Aurora or Qx0 machines. It works on the highest colour depth everywhere regardless of Operating System.
The Aurora version is available on either HD or ED disk. For the latter add £1.00 to the price. ED version is uncompressed and can be run directly from the floppy. All other Floppy versions are compressed. QPC/QXL version comes on CD. Non CD versions DO NOW support digital sound on QPC2

Quantum Leap ED Drives

After many years of unavailability, here they are again! These are high quality Mitsubishi constructed/IBM badged drives with full warranty.
Unlike previously sold ED drives, these do not require FLP_JIGGLE and have no problems formatting 720K disks. However for the latter a switch is included with the cable.

ED Bare unit (no cable)	£ 29.99p
ED Boxed unit (complete with cable/ PSU)	£ 98.99p
Single unit Cable (with switch)	£ 4.99p
Dual unit Cable (with two switches)	£ 5.99p

(More options available, contact us for details. Also available mass quantities of ED and DD disks!)

Talent COMPUTER SYSTEMS for Windows

For QLers that run Windows or with incompatible hardware for Talent Games, we now have re-released these adventures so that they can run on your Windows-equipped PC. No Emulator, floppies, microdrive backups etc. required, just a one-click install! Of course the full QL line is still available! (See side column)

Talent Games for Windows	ea. £ 10.00p

(Each Game includes a runtime installation of QLAY-2 by Jimmy Montesinos)

Old Favourites!

Utilities

Sidewriter v1.08	£ 10.00p
Landscape Printing (EPSON printers)	
ImageD v1.03	£ 10.00p
3D object generator	
Q-Help v1.06	£ 10.00p
Superbasic On-Screen help system	
Q-Index v1.05	£ 5.00p
Keyword-to-topic finder	
ProForma ESC/P2 Drivers v1.04 for ProWeSs	£ 8.00p
Printer Driver	

Applications

Flashback SE v2.03 (upgrade only)	£ 5.00p
Database	
QL Cash Trader v3.7	£ 5.00p
Accounting/Finance	
QL Payroll v3.5	£ 5.00p
Accounting/Finance	
QL Genealogist v3.26	£ 20.00p
Genealogy	
Genealogy for Windows	£ 50.00p
QL Genealogist to Windows version upgrade	£ 25.00p
QL Cosmos v2.04	£ 5.00p
Planetarium	
Q-Route v2.00	£ 25.00p
Route Finding	
Upgrade from v1.xx	£ 5.00p
Britain map v1.11	£ 2.00p
BIG Britain map (needs 2Mb) v2.03	£ 5.00p
Various Britain Area maps (ask for details)	ea. £ 2.00p
Ireland map v1.00	£ 5.00p
Belgium map v1.01	£ 2.00p
Catalonia map v1.03	£ 2.00p
P-Word UK English Dictionary (500.000 words!)	£ 15.00p
Dictionary	

Leisure

Return to Eden v3.08	£ 10.00p
Adventure	
Nemesis MkII v2.03	£ 8.00p
Adventure	
The Prawn v2.01	£ 8.00p
Adventure	
Horrorday v3.1	£ 8.00p
Adventure	
West v2.00	£ 5.00p
Adventure	
The Lost Kingdom of Zkul v2.01	£ 5.00p
Adventure	
All 6 games above	£ 25.00p
D-Day MkII v3.04	£ 10.00p
Strategy/War Simulation	
Grey Wolf v1.08	£ 8.00p
Graphical Submarine Simulation	
War in the East MkII v1.24 (upgrade only)	£ 5.00p
Strategy/War Simulation	
Open Golf v5.20	£ 8.00p
Sports Simulation	
QuizMaster II v2.07	£ 5.00p
Quiz	
Stone Raider II v2.00	£ 5.00p
Arcade Game	
Hoverzone v1.2	£ 5.00p
Arcade Game	
Deathstrike v1.5	£ 5.00p
Arcade Game	
Flightdeck v1.0	£ 10.00p
Flight Simulation	
All 6 games above (Open Golf, QuizMaster II,Stone Raider II, Hoverzone, Deathstrike and Flightdeck)	£ 28.00p

Notes on Software requirements
The following programs have a minimum SGC card requirement: P-Word, Qword, Big Britain MAP for Q-Route

RWAP Services
178 Newtown Road, Bedworth, Warwickshire, CV12 8QN, United Kingdom. Tel: +44 2476 490616 (From the UK dial: 02476 490616)
Website: http://www.rwapsoftware.co.uk
We accept:

PayPal VERIFIED NOCHEX E-MAIL MONEY PREFERRED PAYMENT

(For PayPal please add 4% to the total price. Cheques in £ sterling made payable to R. Mellor)

Tony Firshman was a little light headed

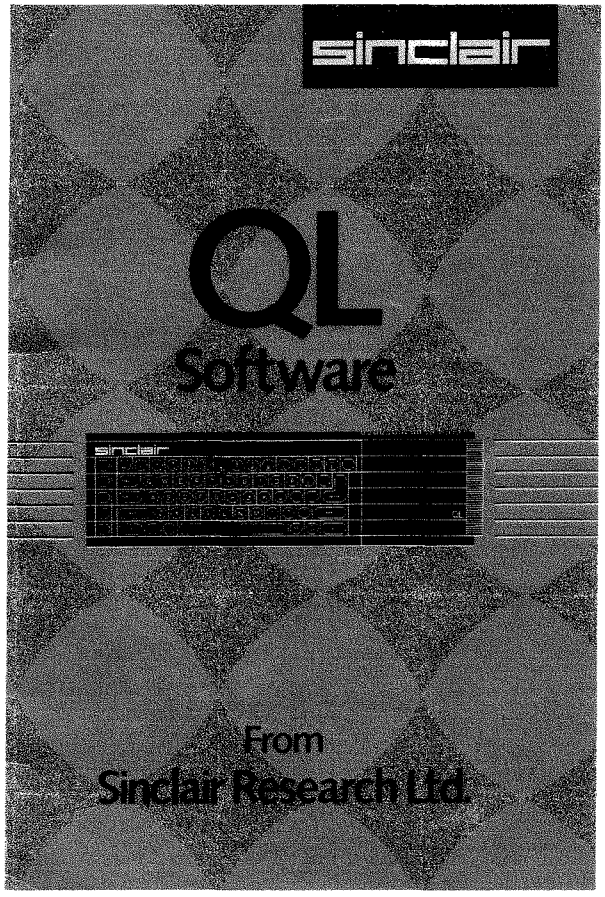
After the show, we headed off to "the chinese" (as we do every time), to discover it was closed. Not forever, but for four weeks, if I remember correctly ... so we may find it will be open again next time. We drove to the centre of Eindhoven to find the restaurant Tony wanted to go to was crowded ... well, it was early Saturday evening. There are plenty of restaurants in the centre of Eindhoven, so the group rushed into another one. The menu was not too complex: the choice was vegatarian or not vegetarian... but that was all you had to choose between.
We chatted for about an hour or one and a half, the QL meeting continued as Alf Kendall got his laptop out, and I fixed one of his Hotkey problems. Then we had to drive back to Germany.
Looking forward to the next Eindhoven meeting, hoping to see more visitors who tried to make it but didn't.

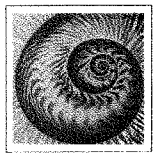
Do you remember...? – Part 3

by Ralf Rekönd

Unfortunately for us readers, Ralf was busy (offline) with private matters and had no time to produce the promised information about the SPK yet. A few days ago (way after this issue's deadline), he reported back to be active QLing again, so we hope to have the article ready for you from Ralf next issue.
Meanwhile, two more reminders from the past:

The first assembler, together with an Editor to edit source files. A better, later version was known as the "GST Assembler" which became the standard assembler in the QL scene for a long time. It is now available in an improved, extended and debugged version from Quanta.





QL Assembler

incorporating

Screen Editor

For anyone who wants to learn machine code or write simple assembly language programs, this program opens up the power of the Motorola 68000 chip.

QL Assembler includes a powerful screen editor and 68000 assembler. Both exploit the full windowing and multi-tasking facilities of the QL, so you can pass from the editor to the assembler to SuperBASIC at will.

- The full screen editor makes it easy to compose your source file in 68000 assembly language and has command features to Search and Replace or Repeat sets of commands.
- The assembler will then take the source file directly from the editor and convert it to 68000 machine code. Output can be saved or directed to a printer.
- Both the editor and assembler are economical on memory (21K and 18K). This means that they will load completely into memory and the assembler runs extremely quickly.
- Detailed documentation is provided for both programs. The assembler also contains a full list of equalities for Qdos system calls.

Price:
£39.95
(includes detailed manual)

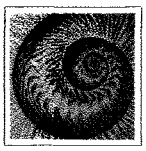
Software written by:
GST

5501

sinclair

The following is something **EVERY** QL user should know - as it comes for free built into SMSQ/E for a long time (and GoldCard, SuperGoldCard, ATARI QL emulators etc. Have a look - there was EX_DEF. What exactly did it do? Why was it removed in future versions?

QL Toolkit



An invaluable aid to anyone interested in writing their own programs, the QL Toolkit contains a host of useful routines and extensions to SuperBASIC.

The QL Toolkit offers a large number of features which will be of interest to almost any QL user – such as a SuperBASIC screen editor to allow a program to be altered by typing anywhere on the screen, a spooler to allow a file to be copied or printed out while other tasks are undertaken, a SuperBASIC screen editor, and extra Microdrive commands.

■ These are supplemented by sets of commands which apply to specific functions of the machine, such as control of multi-tasking, file handling, user-defined graphics, and memory control.

■ QL Toolkit comes in two sections. The extensions to SuperBASIC can be loaded into memory (usually at the start of a working session) and the new commands are then available directly from SuperBASIC. The other section is in the form of stand-alone programs written either in machine code or SuperBASIC which can be loaded from Microdrive as and when they are needed.

■ For a summary of the powerful facilities offered by QL Toolkit, see the other side of this sheet.

Price:
£24.95
(includes detailed manual)

Software written by:
QJUMP

sinclair

5502

QL-Toolkit summary

SuperBASIC Extensions

AJOB	Activate job
ALCHP	Allocate memory in common heap
BGET	Get bytes
BIN	Convert from binary
BIN$	Convert to binary
BPUT	Put bytes
CDEC$	Convert to fixed format decimal
CHAR_INC	Set character increments
CHAR_USE	Set/reset character founts
CLCHP	Clear all memory allocated by ALCHP
CURDIS	Disable the cursor
CURSEN	Enable the cursor
DATA_USE	Set default data file directory
ED	SuperBASIC program editor
EX_DEF	Set defaults for EX/EW/ET
EX/EW/ET	Run machine code programs
EXTRAS	List all extra M/C routines
FDAT	Get program data space from header
FDEC$	Convert to fixed format decimal
FLEN	Get file length from header
FOP_DIR	Open directory (returns error code)
FOPEN	Open file (returns error code)
FOP_IN	Open file for input only (returns error code)
FOP_NEW	Open new file (returns error code)
FOP_OVER	Open, overwrite if exists (returns error code)
FPOS	Get current file position
FREE_MEM	Get the free memory
FTYP	Get file type from header
GET	Get data in internal format
HEX	Convert from hexadecimal
HEX$	Convert to hexadecimal
IDEC$	Convert to fixed format decimal
JOBS	List current jobs
JOB$	Get job's name
NXJOB	Get next job in tree
OJOB	Get job's owner
PARTYP	Get type of parameter
PARUSE	Get usage of parameter
PJOB	Get job's priority
PROG_USE	Set default program directory
PUT	Put data in internal format
RECHP	Release memory to common heap
RENAME	Rename a file
RJOB	Remove job
SPJOB	Set priority of job
SPL	Spool a file
SPL_USE	Set spool output/directory
STAT	Print drive statistics
TRUNCATE	Truncate to current position
VIEW	View a file
WDEL	Delete file(s)
WDEL_F	Delete file(s) (forced)
WDIR	List directory
WMON mode	Reset windows monitor defaults
WSTAT	List file statistics
WTV mode	Reset windows TV defaults

Filters

CCL	Concatenates the line based files
CCT	Concatenates files
CPY	Copies a file
LNO	Adds line numbers to a file
MORE	Copies to a window in bits
PAGE	Splits files on to pages
UC	Converts to upper case

Machine-code Programs

ALARM	Alarm clock
CLOCK	Digital clock in window 0
CLOCKH	Clock with hands (redrawn)
CLOCKN	Clock with hands (not redrawn)
CLOCKS	Digital clock in fixed position
ELLIPSES	Demonstration graphics
LOGO	Demonstration graphics

SuperBASIC Programs

FBACK	Fast selective backup
MBACK	Multiple selective backup
PATCH	File patch utility
SBACK	Fast selective backup
UDC	User defined character generator

Founts

FOUNT0	(875 bytes) The default primary fount
FOUNT1	(687 bytes) The default secondary fount
SUBSO	(875 bytes) Subscript fount (31 to 95)
SUPER0	(875 bytes) Superscript fount (31 to 95)

Extensions to Microdrives

FS_RENAM	Rename file (name at (A1))
FS_TRUNC	Truncate file to current position
IO_CLOSE	Close now datestamps the update location ($38) in the file header
IO_OVERW	Open overwrite now overwrites an existing file

Sources

PAGE_ASM	Source for PAGE filter
UC_ASM	Source for UC filter

Minimalist Games - Part 1
"Membrane"

by Stephen Poole

First, why Membrane_bas?..
.Well, it's because if you play this game on a standard QL keyboard, it will probably not stand up to the bashing you will give it for long!
Now what does the program do? It is in fact very much like the old Pub 'test your reflexes' slot-machines: Run it, then, when the warning bips stop, hit any key as soon as you see a random BLOCK flash up at any random place on the screen, (for a random length of time). If you hit it before it disappears you score points. After 10 blocks have flashed up, your score is given as a percentage, which is a polite way of saying 'Sorry mate, you're either sloshed or dead tired!'. I have tried to make the game sufficiently hard as to ensure that you are on your wit's edge if you hope to get 100%. Remember the delay it takes to depress a key its full distance...

Lines 230 and 260 make sure you have lifted your finger from the key, before allowing the program to continue. Lines 160, 250 & 290 just clear the keyboard buffer, whilst pausing the (blank) screen for yet another (random) period of time. So there is no hope of anticipating a flash...
Change 'tries' if you want more than 10 blocks. This listing was written for a SuperGoldCard with SMSQ/E. You may have to adapt the following variables if you run the program on other types of QL: Modify 'min' and 'max' to adapt the length of time the block flashes on and off if necessary, (but first be sure to change the 'speed' variable, as this regulates the bip-speed and wait delay). It took me some time to work out how to get these reaction-times coded efficiently, as this was a problem I had never encountered previously.

Suffice it to say that with min=25 and max=35, you have to react within .5 to .7 seconds, which is a reflex-rate well adapted to driving conditions. Personally I prefer min=10 and max=20 as this makes for harder concentration. Remember, every time you touch INKEY$, (like PAUSE), the program then continues immediately, which would mess up timing if special measures were not taken.

Although this program is minimalist in function, it did need a minimum of features to make it operate correctly. But at least I did manage to keep it on one page of WINDOW#2,420,206,0,0 for easy reading and editing.

```
100 REMark Membrane_bas. by S.Poole, v11jun2005
110 REMark Hit any key when Block appears, or 'Q(q)' to Quit.
120 OPEN#1,con_16: WINDOW 512,256,0,0: PAPER 0: INK 7
130 REPeat loop
140  ct=0: min=25: max=35: speed=5: tries=10
150  OVER 0: CLS: AT 1,1: PRINT'Get Ready...': OVER -1
160  FOR wait=1 TO max: BEEP 1234,56: i$=INKEY$(#1,speed)
170  CLS: FOR over_draw=1 TO tries
180   ac=RND(511): dn=RND(255)
190   wd=RND(511-ac): hg=RND(255-dn): cl=RND(2 TO 7)
200   BLOCK wd,hg,ac,dn,cl: rd=RND(min TO max): i$=INKEY$(#1,rd)
220   IF i$<>'':ct=ct+1: BEEP 1234,rd: IF i$=='q': EXIT loop
230   REPeat loop2: i$=INKEY$(#1,5): IF i$='': EXIT loop2
240   BLOCK wd,hg,ac,dn,cl: rd=RND(min TO max)
250   FOR wait=1 TO rd: i$=INKEY$(#1,speed)
260   REPeat loop2: i$=INKEY$(#1,speed): IF i$='': EXIT loop2
270  END FOR over_draw: score=(ct*100)/tries
280  AT 1,1: CLS 3: INK 7: PRINT' Score:'!score!'%'
290  FOR wait=1 TO max: i$=INKEY$(#1,speed)
300 END REPeat loop: WINDOW 512,206,0,0: OVER 0: CLS: STOP
```

More Sudoku

by Geoff Wicks

We have used Norman's program to generate two puzzles for you to solve. The left one is at the easiest level and the right one the hardest.

Readers may like to try other methods of generating sudoku puzzles. In the last issue Herb Schaaf gave some sudoku websites. One of these by Hirofumi Fujiwara gives a 10 part course in how to write sudoku puzzles by hand. Is there a QL-er who would like to try writing a sudoku generating program based on this course?

www.pro.or.jp/~fuji/sudoku/index-eng.html

			6					3
				8	4	9	2	
2	4	7						
7	9	1						
				6	3	1	5	
			1					2
		2		1			3	
6	1	8						
		3		5			4	

	9		6		8		2	
		1	2		5	9		
9	1			4			6	7
				3				
3	7			6			9	5
		6	3		4	5		
	2		9		7		8	

Byts of Wood

by Roy Wood

In the crystal ball department I have done quite well over the last few columns. In the last one I was talking about Tarquin Mills' fight with Microsoft and the Office of Fair Trading and I mentioned that, after a vast legal battle (fought largely with the money we pour into the EU Black Hole), Microsoft had backed down and made a version of Windows without its Media Player. I also mentioned that I could see no evidence of this version being sold. Well, when the fight returned to court in April, sales figures were released which show that none of the major PC manufacturers had shipped PCs with this version installed and sales for the boxed product were only around 13,000 which is a tiny fraction of the number of copies of Windows sold. Are they wasting their time and our money or what.

I also compared this to the software patents legislation that is proceeding through the committees of both the US and the EU. In the US, however, it would seem it is now a reality and actually forced Microsoft to downgrade (I know some of you will not think that is possible) some of the core routines in the system. This downgrade was released as part of the recent security patch. The reason for it was the result of yet another lawsuit. A company called Eolas was granted one of these software patents for 'the embedding of small interactive programs such as plugins, applets, scriptlets or Active X Controls into World Wide Web documents'. Microsoft was, therefore, forced to remove some of the code that makes this possible.

Now, neither of these things are particularly QL related I hear you mumble, reaching for the dispepsia tablets, but, apart from smugly smugly polishing my 'told you so' badge, there is a a bearing here. In the first case the point is simple. Stop shooting catapults at the castle and find a better way to undermine it. So many people shout about how their software/OS of choice is better that the big M$ edifice but all that happens is pointless lawsuits which achieve very little. Why not develop your software/OS until it is so much better, so much more user friendly and so much more useful that people will want to choose it over the opposition? It does not take money to do the development, in fact, sometimes, the more teams of programmers you sling at a project the more error and holes creep in.

In the second case the problem is more insidious. Today it is Eolas fighting for money from Microsoft. Now they don't really want that patent so they can use it themselves, they want it so they can screw money out of anyone who uses the same technology. The words I used to describe the patent are those actually used in the document and you can see how vague and all encompassing they are. What if we write a browser for the QL and some poor QL author comes to their attention and gets the full corporate lawyer treatment?

In this way they are able to patent an idea and not a physical object or process. That is a dangerous situation and not one which should be taken lightly. We were denied a GIF driver for LINEdesign many years ago because there was a charge levied on the use of software to create a GIF file. This was not using the software created by the patent holder but software written by anyone to converted or rendered a file into the GIF format. When the format was first released it was meant to aid the swapping of data between programs and systems. Its author made it freely available but, because it was written by an employee of Compuserve and his terms of employment did not allow him to write software for a third party, his employers were able to appropriate the code as soon as it became popular.

In this case, a piece of software designed to free up the exchange of graphics files became a revenue maker for a giant corporation. The patent mentioned above is a similar situation and even the intervention in court of Tim Berners-Lee, the internet pioneer, was unable to stop them getting the patent. This could backfire on us should anyone do something similar.

ROHS

The latest bit of EU legislation to wash up on our detritus laden shores is the ROHS directive. This, like the EMC legislation which sought to protect us from electrical interference (can we have one which protects us from political interference next please), is being put in place 'for our own good'. I have not done too much research on this but I am not alone in that because it put many of the larger electronic component manufactures on the back foot. In essence it states that 'producers of electronic goods which are put on the market

from 1st July this year must ensure that they do not contain hazardous substances'. This, some think, applies only to sales within the industry - meaning a company which assembles PCs cannot buy non ROHS compliant goods but it can use up its stock of lead or other hazardous substance packed hard drives and motherboards and sell them to the public. I cannot recall the last time I sucked a hard drive - pause to go 'Oo-er missus' - but I cannot see how the amount of toxins in the computer in front of me is going to affect me and the WE directive that came into force last year made it compulsory that waste electronic goods did not go into landfill but were disposed of in a correct, recycled, way so it cannot be the toxins leaking into the ground that they worry about. I wonder what it is all about. Some of the resellers are unaware of this new directive and, a quick trawl through the components on sale at the moment reveals that many of the ROHS compliant goods are not even going to be available until the beginning of June.

Once again, in the UK at least, the burden of dealing with this legislation falls squarely on the shoulders of the Department of Trade and Industry's Weights and Measures department and we all saw how well they dealt with the previous legislation.

I wonder how this will affect the few pieces of hardware still current for the QL. A few months ago, as we drove home from the last Eindhoven meeting, Tony Firshman was telling me that he had enormous problems with a batch of chips for the superHermes. I was unable to program the chips for the Super Gold Card when I wanted to build a third batch of 50 and Tony was suffering the same problems with the main chip for the superHermes. Luckily he has now found a batch of chips he can use but, just as he breathes a sigh of relief, is he now going to fall foul of the new ROHS legislation?

You are Going?

I received an interesting disk through the post the other day. It came from Hugh Rooms who was one of the members of the Sussex QL group. He has been involved with Amateur Radio for a while and somehow diverted his attention to GPS systems. He linked a GPS receiver to his PC and used a program written in SBASIC on QPC2 to output the data to the screen. He is not using this to plot a route or even give his position but is, instead, using it to track the positions of the satellites in the sky. I ran the program here but there seemed to be something not quite right in my setup and a got a few error messages. It is, however, a work in progress and in its early stages. He is going to be at the Hove Show and will be giving a demonstration of the program and then a talk about the ideas behind it.

This is, in many ways, the kind of innovative use that the QL is good for. SBASIC, as we all know, is a very flexible language which works well for a simple boot file and can be used for much more complex and interesting tasks. It does not take a genius to write some of the more simple applications and yet, in the hands of a deft programmer, it can do remarkable things.

Sackville Hotel

Talking of the Hove Show, which is where, I hope, many of you are reading this, you may realise that this is the 10th Sussex QL Workshop. Some of you may recall the first of these workshops which was held in the Sackville Hotel on the Hove seafront. This may bring back some memories, especially to those unfortunate enough to have to stay at the hotel itself. I recall that the accommodation offered was slightly erratic. Hot and cold running water meant it ran from room to room and some of the guests had better lodgings than others.

I was drawn to this hotel for the first show because it looked so charismatic from the outside but we did not go back there for the second year. The hall was too small and the rooms were not good enough. Those who did stay there may not be surprised to know that it closed recently for renovation. The plan was to redecorate and re-appoint all of the hotel rooms refurbish he dining room and other communal spaces and put four luxury flats on the roof. It was during this latter process that the roof collapsed and the whole middle of the building caved in. Whether this was due to a miscalculation on the part of the architects who were doing the redesigning or a weakness in the building that the surveyors missed is still being debated but it is currently a ruin. The QL community has outlived it!

Zip and Unzip

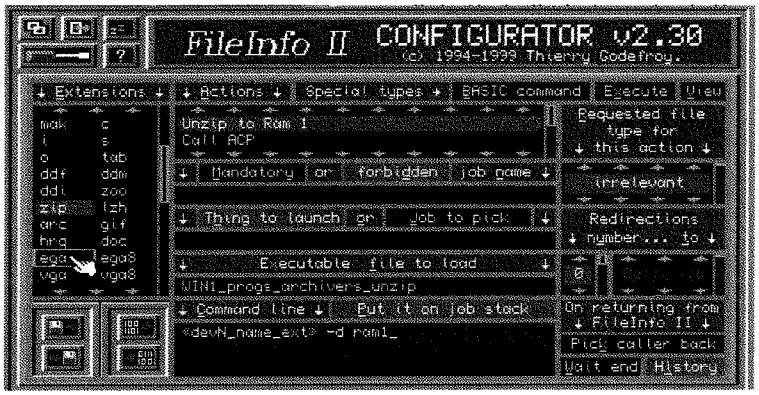
Dilwyn Jones has been talking about ZIP files on the user group list recently. The subject was raised because people do sometimes have problems extracting the many QL programs that are available on the Internet. In the course of the discussion I mentioned using Theirry Godefroy's wonderful FileInfo 2. This is the method I use the most for extracting files from a ZIP archive and I find it very simple to use. The problem of course lies, not in its use but in the setup. Dilwyn seemed to think that it would confuse many people but I will attempt to explain it here and you can let me know if you think it is too hard.

The first problem you have, of course, is to get a copy of UNZIP. Now you can download this from the internet but it does come in a zipped archive so it may be a problem to extract UNZIP from a ZIP archive without having UNZIP which is why you downloaded the archive in the first place and.....is your head hurting yet?

QL Today readers will, however, have a copy of UNZIP on almost every disk we have provided with the magazine so, hooray! We are off to a good start. On one of the early disks we even gave a way a copy of FileInfo 2. FileInfo 2 is a program that sits in the background and allows you to execute a data file from QPAC 2, Cueshell Disk Mate 5 etc. It will then, if you have it set up correctly, load the program that you have chosen to use that DATA file. This is a simplistic view of its function and you can read about it in greater detail if you have QL Today Issue 5 Volume 4.

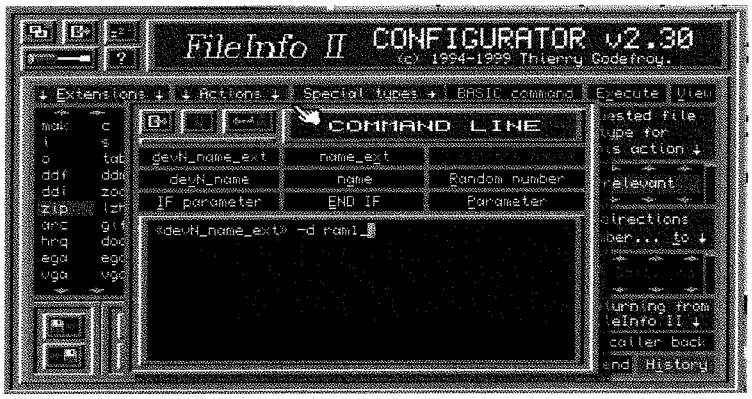
Setting up FileInfo2 is not hard but it can be frustrating. The configuration program uses a similar procedure to that used when recording macro on most systems. The frustration comes in when you make an error and hit the 'delete' key. Of course what you then get is «DELETE» or something similar, in your macro. That said, once you have the hang of it it is quite easy to go through the keypresses you need to do a task and add them to the FileInfo2 configuration database.

There are two pictures here to help you.



In the first is the front end of the configurator. ZIP is high-lit in the left hand 'Extensions' window. If it is not there on your copy go to the bottom of the list and click in an empty space and add it. 'Actions' is the window which will allow you to make a list of things you can do with files of this extension 'UNzip to RAM 1 is high-lit. Again, if it is not there go to an empty space and add it. 'Executable File to Load' should show the location of your UNZIP program. 'Command line' should contain the command line shown and the box beside it 'Put it on job stack' should be high-lit.

In the second you will see, in the foreground, the command line editor that you can get to by clicking on the command Line box. The nine boxes above the command line editor will add the text that is in them to the command line. Just click on «devN_name_ext» and it will appear in the box. Clicking on the second icon from the



top left (three vertical bars) will set the cursor flashing so you can type and just add a space and the '-d ram1_'. The third icon at the top left hand corner will delete if you have made a mistake and the first icon will take you back to the main window with the command line intact.

All you have to do is to click on the icon at the bottom left to save your work and then click on the icon next to it to make the database current.

I'm sure you can all set this up.

Once you have done this all you have to do is to execute a ZIP archive from a files menu that is FileInfo2 compatible and it will be extracted to RAM1_

PDF Conversions

After my deriding Quill in the last column Norman Dunbar announced on the User Group mailing list he was looking into a pdf converter based on his 'Stripper' program. Now I am all in favour of getting the QL to produce or accept more mainstream formats and I think that it is this lack of inter-connectivity that has fuelled many of our users departure from the QL fold. There was a lot of interest and no shortage of suggestions about how this could be achieved or what course it could take. maybe, if he achieves it, it should be combined with Geoff Wicks (can I still give him a plug now he is editor?) wonderful QL to PC convertor.

In many ways we often in this column wondered why people leave the QL for other systems but maybe we should be thinking about this in a different way. Not why are they leaving but what is drawing them away. If we understood more about what attracts them to a different system we might stand more of a chance of improving our system enough to get them interested again.

Other Readers

I was collecting the renewals for the next issue of this magazine over the last month and someone called asked if the magazine was going ahead for another year. He said that 'we are a dwindling group of people' so he wanted to make sure we were going to have another year at least. I am pleased to be able to report that, on the basis of the renewals I have received so far the readership of this magazine has remained pretty stable. We did lose a couple of our number who, sadly, passed on and a couple more who, for one reason or another, have said they will not be renewing but we gained almost a reciprocal amount who subscribed for the first time. One of whom is very keen and has all the back issues of the magazine.

Other systems gain followers because people want to play the old retro games they played when they first had a computer but the QL had few of these and I suppose it is interest in the system that draws them in.

So we will be around for a while yet. If you are one of the small number who have not renewed yet you can do so without worrying that we are going to fold.

And now a little humour to get you into the summer:

Modern Technology Decoded

Nanotechnology - Electronic equipment with controls too minute for your grandmother to use

DVD-RAM - Electronic equipment acquired by driving a car into a retailer's shop front.

Mini-Disc - That part of your back you damaged carrying a large monitor into a QL workshop **Digital Camera** - A camera so small that all you get is photos of your fingers

Talking Book - That person you accidently stand next to at a workshop who then goes on to detail every Trap and Call in the 68xxx instruction set and is marginally less interesting than the St Petersburg Telephone Book.

Mobile Phone - Telephone that is never where you last left it

Text Message - Legitimised Dyslexia

System Overhead - Exclaimation often heard when standing in Dilwyn Jones' Garden

Barcode - Coded message used to arrange to meet your friends at the pub without letting on to your wife.

Barcode reader - A wife who has cracked the code

Terrabyte - Amount of space needed on a hard drive to store all your Zombie Movies

Surround Sound - Policeman with a megaphone standing outside your house asking you to surrender

5.1 Stereo - Stereo system with the volume controls marked in decimals. Usually owned by a talking book (see above)

Icon - Cheap MP3 player you bought at the market which does not work

CCTV - Spanish Television Station

PDA (Personal Digital Assistant) - Private Manicurist

Tablet PC - Laptop used by Moses

Email - Exclamation heard in the north of England when the postman arrives

This is (nearly) the last page of Volume 10 of QL Today.
We hope you enjoyed reading the issues. You can be sure that we will try very hard to produced a Volume 11 for you which will be as interesting as the Volumes before.

Please help keeping the costs down. If you have not renewed yet, please fill in the renewal form which is enclosed with this issue and return it - or send an e-mail to your QL Today distributor (i.e. QBranch or JMS).

Thank you very much - and "see you" in Volume 11!

The QL Show Agenda

Regular QL Meeting - (NL) Eindhoven

Saturday, 17th of June, 10:00 to 16:00
Pleincollege St. Joris, Roostenlaan 296
JMS will be there - with his usual range of QL/SMSQ programs, special offers, CD-Rs, DVDs etc.

The Canadian US Show

We plan to have the 2006 North American QL Show in the Niagara Falls area of the US/Canada on September 30th 2006 with the exact location on US or Canadian side yet to be decided.
We will have a dinner on Friday evening and an all day show on Saturday. Jim Hunkins will be able to demonstrate the latest in QDT development. Contact Bill Cable cable@cyberportal.net or phone 1-603-675-2218 for more information. More details will be announced soon.

Quanta and The QL Today Ad

Some of you, who are Quanta members, may have noticed that our ad in the last issue was woefully out of date. We would like to point out that this was not our fault. We put a new advert together and sent it to the Quanta editor to be included in the current issue. We also received an acknowledgement that it would be changed but the ad they used was the one from back in January. The only reason that we point this out to you is that we do not want out readers to feel that we do not care about these things. We are remonstrating with Quanta about this since it is not the first time that this has happened.

A quick note regarding QL Shows from JMS:

JMS has been to the last Eindhoven shows, but unfortunately not to the last two QL Shows in the UK. It looks as if I have to miss Hove as well, although I hope to be able to deliver the magazine for the show. I am very sorry about this, and this does not demonstrate lack of interest - otherwise we would not start Volume 11 of QL Today. Major problems with both our cars stopped me from travelling long distances, and so did the lack of company (no borrowed car, no chance to share costs and driving). With my current health situation, I prefer not to drive long distances on my own anyway. Nothing serious!
I will be at Eindhoven in June again, that's for sure. And hopefully at the next UK show too!

The Next Issue

We plan to have the next issue ready at the End of July/Middle of August. It depends on how quickly we will get reviews, articles etc. Summer is usually fairly quiet, that's why we cannot provide a precise date this time.
Final reminder: please do not forget to renew if you haven't done it yet!