

Q-emuLator for Windows

Version 2.3.3 - March 9th, 2005

Users' Manual

Edited by Phoebus Dokos

Q-emuLator is © 1994-2005 by Daniele Terdina
The Sinclair block logo, SuperBASIC and QL are © 1983 - 2004 Amstrad plc
QDOS is © 1983 - 2004 Amstrad plc except North America where it is © Paul Holmgren - Frank Davis
SMSQ/e, The Pointer Environment, SBASIC and Toolkit II are © 1983-2005 by Tony Tebby
Minerva and MultiBasic are © 1985-2005 by Q-View/Laurence Reeves
Microsoft® Windows™ is © 1981 - 2005 Microsoft Corporation
All other brands mentioned within this manual are © of their respective owners
This manual is © 1994-2005 by Daniele Terdina

Table Of Contents

1. Getting Started	3
1.1 Introduction	3
1.2 Requirements	3
1.3 Special Considerations for XP SP2	3
1.4 Installation	3
1.5 Running the emulator.	4
1.6 Using QL ROMs	4
1.7 Running SMSQ/e for the Gold Card.	5
2. Emulator Features	6
2.1 Main Features	6
2.2 "Expanded QL" mode Features	6
2.3 Overview	7
2.4 Full screen modes	8
3. File Handling	9
3.1 QDOS files	9
3.2 Access to QDOS formatted floppy disks	9
3.3 Format QL floppy disks	9
3.4 Access to QXL.WIN files	10
4. Supported Devices	11
4.1 Keyboard	11
4.2 Ram disk.	11
4.3 Sound	12
4.4 Mouse support	12
4.5 Joystick	13
4.6 Real time clock	13
4.7 Serial ports	14
4.8 Parallel ports.	15
4.9 TCP/IP.	15
5. Configuring Q-emuLator	16
5.1 The QL Configuration window	16
5.2 Configuration files	17
5.3 Using 16KB QL expansion ROMs	18
Appendix I - Format of .QCF files	19
Appendix II -Format of QL files stored in the PC file system	22
Appendix III -Q-emuLator support site	23
Appendix IV -E-mail address for feedback	24
Appendix V -QL ROMs supported by Q-emuLator	25
Appendix VI -Debug_68k Users' manual	26
1. Installation	26
2. The debug_68k window	26
3. The disassembly window	27
4. The register display	27
5. The stack display	27
6. Debugger commands	27
7. Stepping through code	28
8. Using debug_68k.dll in emulators	28

1. Getting Started

1.1 Introduction

Q-emuLator is a software emulator of the Sinclair QL home computer.

Originally written in 1994/95 for the Mac OS, Q-emuLator was ported to Windows three years later, and is still being improved.

Q-emuLator emulates the QL processor (the Motorola MC68008) and redirects the QL I/O (video, keyboard, mouse, mass storage, sound, joystick, parallel and serial ports) to the PC hardware. Putting a QL ROM in this virtual environment makes it possible to run QDOS without its noticing that it's not running in a QL black case.

Q-emuLator runs like any other applications in the Windows environment, without taking over all the machine. You can switch to other active applications, and Q-emuLator can also run (more slowly) in the background.

Q-emuLator (Expanded registration) includes a RAM disk and a TCP/IP driver so that you can use most QDOS-compatible Internet applications.

1.2 Requirements

A 486 or better system with Windows 95, Windows 98, Windows ME, Windows NT 4, Windows 2000, Windows XP or later.

Sound emulation and full screen graphics are available if DirectX is installed on your system. (Recent Windows installations already include DirectX).

Formatting of QL disks is available only when running on Windows NT 4 or later (including Windows XP and Windows 2000).

1.3 Special Considerations for XP SP2

No special considerations exist apart from adjustment of your firewall settings to use with QDOS-compatible Internet applications.

1.4 Installation

To install Q-emuLator, run the setup.exe program that you downloaded from the Q-emuLator web site. The full installation (recommended) requires about 2 MB of free disk space.

To uninstall Q-emuLator, open the 'Add or Remove Programs' Windows control panel, select 'Q-emuLator' from the list of installed programs and click the 'Remove' button.

To purchase a registration code, follow the link from the Q-emuLator's web site.

1.5 Running the emulator

After installing Q-emuLator, you can start it by using the shortcut in the Windows Start menu (or by clicking on the desktop shortcut icon or the QuickLaunch icon, if you chose to create them during the installation process).

Before starting the emulated QL, you may want to click on one of the microdrive slots (the gray rectangles in the lower part of the Q-emuLator window) to attach it to a location containing some QL software (a QDOS floppy disk, a windows directory or a QXL.WIN file). For example, attach it to the QL Software\QL demo subdirectory found in the Q-emuLator installation folder.

Start emulation by selecting 'Start' from the 'QL' menu, or simply by clicking on the picture of the QL keyboard in the main window.

If the first microdrive slot is attached to a mass storage location containing a SuperBASIC program named 'boot', that program will automatically start after you press F1 or F2 at the initial QL prompt.

1.6 Using QL ROMs

The emulator needs a file with a supported¹ QL ROM image to work, but such a file is already provided with Q-emuLator. The provided ROM is a Minerva 1.97 ROM, compatible with the original Sinclair ROMs, but faster and with more features. Minerva ROMs are public domain. Tony Firshman made version 1.89 and earlier public domain a few years ago, and more recently Laurence Reeves (who wrote Minerva) made all versions public and released the latest sources under the GNU license. (Download Minerva sources from <http://bergbland.info>).

If you prefer to use Sinclair ROM images, you can find them on the Internet (follow the link from the 'Q-emuLator for Windows' www page), or you can create an image on your QL with the "**SBYTES FLP1_QL_ROM,0,49152**" command.

This command doesn't work on Gold Card and Super Gold Card systems, as you get a modified ROM image instead of the original one. Therefore, if you have a (Super) Gold Card, you must disconnect it before using the SBYTES SuperBASIC command and copy the ROM to a microdrive, then from there to a floppy after reconnecting the card.

Once you have the ROM file on a QDOS floppy disk you can read it from Q-emuLator on your PC. Follow these steps:

1. Launch Q-emuLator.
2. Insert the QDOS floppy disk with the ROM file image in the PC drive (A: or B:).
3. Click on the gray rectangle representing the left microdrive slot (below the window with the QL picture); choose 'QDOS Floppy Disk' from the pop-up menu that appears, and choose either 'A:' or 'B:' from the sub-menu, depending on where you inserted the floppy disk. Q-emuLator looks at the disk and displays its name on the microdrive slot.
4. Choose the 'Install QL ROM from disk...' command from the 'File' menu. This instructs Q-emuLator to copy the ROM image in its directory and use it for emulation.
(Warning: this command requires Q-emuLator to reside in a writeable directory and not on CD-ROM or other read-only media).

What if you would like to install a new ROM image that is already on your PC, instead? (For example a ROM that you downloaded from Internet.) In this case you need to point Q-emuLator to the ROM image: go to the 'QL Configuration' window (Choose 'QL Configuration...' from the 'QL' menu), click

¹ See appendix V for supported QL Roms.

the uppermost 'Select...' button and find the ROM image file in the file selection window that appears. Click the 'OK' button to close the 'QL Configuration' window. Done!

Q-emuLator also includes a Toolkit II ROM containing many SuperBASIC extensions that are commonly used by QL software. The provided Toolkit II ROM image (version 2.10) is © 1985 by Tony Tebby. Tony Tebby changed its distribution from commercial to freeware in January 2005 (when he also changed to freeware the HOT_REXT, PTR_GEN and WMAN Pointer Environment extensions).

1.7 Running SMSQ/e for the Gold Card

(available only with the "Expanded QL" registration)

If you have a Gold Card or Super Gold Card QL expansion and you are running a recent copy of the SMSQ/e operating system on it, you can use your copy of SMSQ/e with Q-emuLator, too. Just configure the emulated QL to start with any Sinclair or Minerva ROM and plenty of RAM, and load and execute the SMSQ_GOLD² file as you usually do on your (Super) Gold Card system. (E.g. `LRESPR win1_SMSQ_GOLD`, assuming you have the Toolkit II installed in the emulator and you put the SMSQ_GOLD file in a PC directory associated with win1_).

Q-emuLator now emulates some of the Gold Card's hardware. Not all of it, but just enough to support SMSQ/e. The emulator also detects SMSQ/e being loaded and automatically adds its own device drivers to SMSQ/e, allowing for example access to the Windows file system. A mouse driver is also installed when you load SMSQ_GOLD, allowing you to use the PC mouse right away.

Note: Compatibility with SMSQ/e for the Gold Card is a new and still experimental feature. In particular, some problems may be expected with the file system and I/O in general, as the Q-emuLator's device drivers are not completely compatible with the ones in SMSQ/e. Some preliminary testing has been done using SMSQ/e for the Gold Card versions from 2.97 to 3.07, and Q-emuLator is not expected to run with old versions of SMSQ/e. Please report any problems you find running SMSQ/E on Q-emuLator.

²

Newer SMSQ/e edition executables are usually named: GoldCard_bin.

2. Emulator Features

2.1 Main Features

- Original QL with up to 384KB of RAM
- Supports all Sinclair and Minerva ROMs as well as “Custom” (hacked) ROMs and experimental ROM editions
- Up to two 16KB expansion ROMs
- Up to 8 virtual microdrives, each mapped to a Windows directory or a floppy disk drive
- Access to QDOS formatted floppy disks
- Format QL floppy disks (Windows XP/2000/NT only)
- Original QL screen (mode 4, mode 8) and second screen (used by some games and by the Minerva ROM)
- Windowed or Full Screen modes of operation
- Compatible with all PC keyboard layouts with or without dead keys³
- Sound (BEEP) emulation
- Joystick (CTRL) emulation
- Fast Serial ports
- Ability to store QL and Q-emuLator settings for easy retrieval and fast changes of emulation mode
- Easily editable QL and Q-emuLator settings through a user-friendly interface
- Ability to paste text from the Windows clipboard
- Speed comparable to that of a real QL.

2.2 “Expanded QL” mode Features

The “Expanded QL” registration has some more features:

- Full emulation speed (exact speed depends on your PC speed), many times faster than a real QL.
- Up to 16MB of QL RAM
- Support for “Level-2” subdirectories
- Map microdrive slots to QXL.WIN files
- Compatibility with recent versions of SMSQ/e for the Gold/Super Gold Cards⁴
- Built-in ram disk
- Mouse support (when the Pointer Environment is installed⁵)
- parallel ports
- acceleration of some QDOS graphics routines
- fast screen mode (Shift+F12)

³ All international keyboard layouts are supported, including US International.

⁴ Currently all recent (v.3.07 at the time of writing) SMSQ/e versions **without** colour drivers are supported.

⁵ Mouse support is enabled automatically when SMSQ/e (that includes the Pointer Environment) is installed.

2.3 Overview

The program main window has two parts: the upper part shows the QL display, and the lower part the virtual microdrives (Figure 1).



Figure 1 - The Q-emuLator main window (QL emulation stopped)



Figure 2 - Slot assignment pop-up menu

You can attach a Windows directory, a disk drive or a QXL.WIN file to a microdrive slot by clicking in it (in the microdrive slot gray rectangle) and selecting a command from a pop-up menu (Figure 2).

Regardless of the type of mass storage attached to a particular slot, you can access it with any of the MDV, FLP or WIN device names. In other words, all of MDV1_, FLP1_ and WIN1_ refer to slot number 1. This way, you can for example copy your microdrive programs to a Windows directory on your hard disk, and it will automatically work despite the program trying to access itself by using MDV1_ instead of WIN1_ (WIN is the device name traditionally used to access the hard disk in QDOS systems expanded with a hard disk interface).

To transfer files from your QL system to the PC you can use QDOS formatted floppy disks and read them from Q-emuLator. If your PC has problems reading your old QL floppy disks and you still have your QL, try using Q-emuLator to format an empty floppy disk and copy your old floppy disks to the new one by using your QL.

2.4 Full screen modes

When the emulation is running, you can enter Q-emuLator's full screen mode(s). Full screen modes are available on most systems, but may not be available on old systems if DirectX™ is not installed or if the graphics card driver doesn't support DirectX™.

Press the F12 keys to enter the full screen mode (Figure 3). This mode can be slow, because each QL pixel is mapped to multiple PC pixels, but the QL display takes the whole screen. This is the only mode that emulates flashing in the QL 8 colours display mode. To return to window mode, press F12.

In full screen mode, menus and microdrive slots are not visible. To access them, return to window mode by pressing F12.

If you have the "Expanded QL" registration, press the Shift+F12 key to enter the fast screen mode, where the QL display occupies most of the screen. This is the fastest graphics mode. Press F12 again to return to window mode.

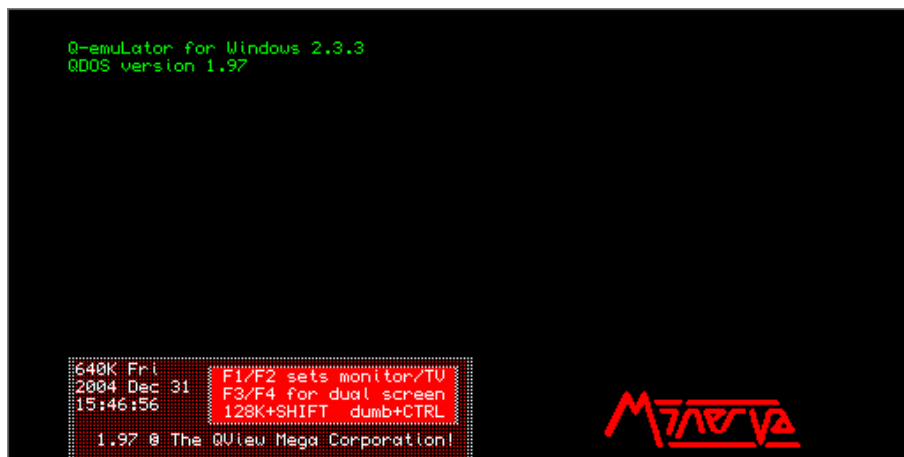


Figure 3 - Full Screen mode Q-emuLator booting Minerva 1.97

3. File Handling

3.1 QDOS files

QL executable files (as well as some other kinds of files) are stored on the PC in a special format containing QDOS-specific information.

The "QL Software\FileUtil" folder contains some utilities that can be used on a QL, should you ever need to translate files on a QL between QDOS and Q-emuLator for Windows formats. The translation is automatically carried out for you on your PC when you use Q-emuLator to copy files between a Windows directory and a QDOS formatted disk. To translate (on a QL) QDOS files to the Windows format you can use the "QLToWin_exe" utility. The archive contains two other utilities: "MdvToWin_exe" translates to the Windows format files stored on copy protected microdrives, while "WinToQL_exe" translates back from the Windows format to the QL's.

Most non-executable files do not need to store extra QDOS information; therefore they have the same format both in the QDOS and in the Windows environment.

3.2 Access to QDOS formatted floppy disks

You can attach QDOS formatted floppy disks to the virtual microdrive slots. When you click on a slot you can choose to attach it to a Windows directory, to a QXL.WIN file ("Expanded QL" registration only) or to a floppy disk drive with a QDOS-formatted disk in it, or you can empty the slot if it was already filled. Note that you access each of the eight virtual MDV slots with all of the MDV, FLP and WIN device names, regardless of what type of medium is attached to a slot.

Note: FLP1 and FLP2 refer to slot 1 and slot 2, NOT to drive A: and B:.. Of course, if you like you can attach drive A: to slot 1 and drive B: to slot 2, but you are not required to do so.

3.3 Format QL floppy disks

(not available on Windows 95/98/ME)

The **FORMAT** command is available only on Windows NT4, Windows 2000 or Windows XP or later.

For example, if you want to format a floppy disk in QL format you could proceed as follows:

1. Start Q-emuLator, and start emulation by clicking on the QL picture and then pressing F1 at the QL F1/F2 prompt;
2. Click on the first microdrive slot and select QDOS Floppy Disk and then A: from the menu;
3. Insert the disk in drive A: and finally enter the SuperBASIC command **FORMAT FLP1_**.

When you **FORMAT** a disk, Q-emuLator automatically detects the disk density and formats the disk with the appropriate number of sectors, resulting in 720KBytes (1440 sectors) of free space for a DD (double density) disk and 1.44 MBytes (2880 sectors) of free space for a HD (high density) disk.

A window pops up asking you to choose between Full Format and Fast Format modes, or you can cancel the operation. Full Format completely erases the disk, while Fast Format just empties an already formatted disk, but is extremely fast.

IMPORTANT: Use the Fast Format option only to quickly erase all the files from an already formatted disk (either in PC or QL format). If the disk is unformatted and you use Fast Format, you may lose all the data that you

subsequently store on the disk. With both the Full and Fast types of format, any data stored on the disk is completely erased and there's no way to recover it after the format operation!

You can use the Fast Format option to quickly erase a formatted PC disk and convert it to the QL disk format.

During a Full Format operation, a window shows its progress and offers the option to abort it by pressing the Cancel button. However, even if you press Cancel at this point, any data previously present on the disk are lost.

3.4 Access to QXL.WIN files

(available only with the "Expanded QL" registration)

In this manual the term 'QXL.WIN file' refers to the Windows/DOS files used by the QXL⁶ (a hardware QL emulator for DOS) to store the emulated QL filesystem. These files are similar to images of SMSQ/e hard disks stored in a single DOS file.

Q-emuLator allows you to attach QXL.WIN files to microdrive slots to access their content: right-click on a microdrive slot and select QXL Hard Disk File...) from the menu, then choose a QXL.WIN file to open.

You can read and modify existing QXL.WIN files, but creating new ones is not supported, yet. However, some free tools (wxqt2 and qxltools) are available on the Internet (<http://www.daria.co.uk/>) that will provide you with that capability.

⁶ Also used by uQLx - the "sister" emulator of Q-emuLator for UN*X systems-, its derivatives and the QPC II SMSQ/e emulator.

4. Supported Devices

4.1 Keyboard

Q-emuLator is compatible with all types of PC keyboards and as of version 2.3.3 with various keyboard layouts including those with “dead keys”.

The keyboard type-ahead buffer has been expanded from the QL's 7 characters to over 1000 characters.

Some PC keys not available originally on the QL are used as a shortcut for common combinations of QL keys (Table 1):


































PC (Q-emuLator) Key	QL Key combination	Notes
	 	-
	 	-
 ... 	  ...  	-
	 	-
	 	-
		-
		-
	 	Both  and   can be used to freeze/unfreeze the QL display output. The Scroll Lock Led on the keyboard is on when the QL screen is frozen.

Table 1 - Q-emuLator - QL Key equivalents

Three other PC keys have special functions :

-  and  : Toggle between window and full screen modes.
- : Paste up to 1KB of text from the Windows clipboard.
- : Invoke the debug_68k debugger (if installed).

4.2 Ram disk

(available only with the “Expanded QL” registration)

Q-emuLator provides a built-in ram disk device driver. If you use it instead of loading a QL software ram disk driver (like ‘ramprt’) you will gain in speed.

The device name is ‘ram’ (but you can change it in the ‘QL Configuration’ window) and it has eight drives, ‘ram1_’ through ‘ram8_’.

You can use the ram disk both in dynamic and static mode. In dynamic mode you can freely copy files to the ram disk and more ram will be allocated for it every time it is needed, up to filling the

whole QL's memory. In static mode you assign a fixed memory amount to each drive, by formatting the drive and specifying the requested number of blocks as the medium name: For example, the command `'FORMAT ram1_400'` assigns a fixed amount of 200KB (as each block is 512 bytes long) to the first ram drive.

To erase a whole ram disk drive at once use the `'FORMAT'` command without specifying a file name: For example `'FORMAT ram4_'`.

In static mode all the memory needed by the driver is allocated when you format the drive. This means that you are sure that no other job can take away that memory before the ram driver needs it, but also that the ram driver is not allowed to use more memory than the amount specified.

While dynamic mode is easier to use, you are encouraged to use static mode whenever it is possible, because it is a lot faster: in dynamic mode the driver has to call emulated QDOS routines every time that it wants to allocate or release a block of memory, whereas in static mode all the memory is allocated at the beginning and QDOS is no longer called.

The ram disk device driver is not level 2 (ie. it doesn't support subdirectories), and there are no SuperBASIC extensions to modify its behaviour (for example there is no `RAM_USE` keyword).

If you prefer to use another ram disk driver (although it will be slower, as it has to be emulated), just disable the built-in device driver in the 'QL Configuration' window, and install your driver by installing an add-on rom containing it or by loading it from your boot file.

Note: For a ram disk drive used in dynamic mode, the DIR command reports a negative number of free sectors and zero total sectors. Zero is the number of sectors statically allocated to the ram drive (none when in dynamic mode). The number of free sectors is calculated as the total number of statically allocated sectors (zero for a dynamic ram drive) minus the number of used sectors, and it is therefore negative for a dynamic ram drive. Consider for example a ram drive with 100 sectors full of data; if in dynamic mode, DIR reports -100/0 free sectors; if the ram drive was statically formatted with 400 sectors, DIR reports 300/400 free sectors.

4.3 Sound

QL `'BEEP'` emulation is available if DirectSound™ (a DirectX™ component) is installed on your system.

Sound is on by default. You can turn it off through a switch in the 'QL' menu.

There is still a minor problem with sound emulation: a click is sometimes heard if a `BEEP` command is issued when a previous `BEEP` command is still playing.

4.4 Mouse support

(available only with the "Expanded QL" registration)

Mouse emulation is provided under the Pointer Environment through a special mouse driver, named 'MacMouse'.

To take advantage of it, install the 'MacMouse' mouse driver (included in the Q-emuLator package) JUST AFTER having installed the Pointer Environment (that is, the 'PTR_GEN' extension, not included in the Q-emuLator package).

To install the mouse driver use the following SuperBASIC command:

```
a=RESPR(1024):LBYTES mdv1_MacMousexy,a:CALL a
```

or, if Toolkit II is installed:

```
LRESPR mdv1_MacMousexy
```

The command assumes that the 'MacMousexy' file (replace x and y with the version number: for example 'MacMouse11' for the mouse driver version 1.1) is in the first Microdrive slot.

Exception: **You don't need to follow these steps to install the mouse driver when running SMSQ/e for the Gold Card. SMSQ/e includes the Pointer Environment in it, and Q-emuLator automatically installs a mouse driver when SMSQ/e is loaded.**

Once the mouse driver is installed, any active QL mouse cursor will be moved instead of the Windows pointer when it is over the QL display window (provided that the QL display window is the frontmost window).

Note that as the QL cursor follows the Windows' one, its speed is independent from any settings in the QL environment. On the PC, the mouse speed can be adjusted in the Mouse Control Panel.

Unlike the left and right mouse buttons, the central button is not emulated.

4.5 Joystick

Select 'Joystick' in the 'QL' menu to enable or disable joystick emulation. When enabled, the PC joystick will be seen by the QL as connected to the CTL1 port.

4.6 Real time clock

The QL had a real time clock, but you had to set it every time you started the computer, because there was no battery to keep it working while the computer was powered off. On Q-emuLator there's no need to set the QL real time clock, as the current date and time is read from the PC clock every time the program starts.

Note: Changing the QL clock settings doesn't affect the PC clock, and the change is limited to the duration of the current emulation session.

4.7 Serial ports

SER1 and SER2 are linked to COM1: and COM2:, although you can redirect them to any COM port. With the "Expanded QL" registration you can also use SER3 and SER4, linked by default to COM3: and COM4:.

To open a serial port I/O channel you specify the device name and options with the following syntax:

SERnpft

n, *p*, *f* and *t* represent optional characters that you can add to the name to specify the port number and other parameters. The following table (Table 2) shows the possible values for these parameters:

Parameter	Parameter's meaning	Possible Values	Values' meaning	Default Value
<i>n</i>	Serial Port number	1-4	Port 1 - Port 4	1
<i>p</i>	Parity Checking	E O M S	Even Odd Mark Space	None (no parity checking)
<i>f</i>	Flow Control	H I	Hardware Handshake No Handshake	H
<i>t</i>	CR, EOF translation	R Z F C	No translation Use Ctrl-Z as End-Of-File Use Form Feed as End-Of-File Use Ctrl-Z as End-Of-File and Carriage Return as End-Of-Line	R

Table 2 - SER device settings

You can omit any of the *n*, *p*, *f* and/or *t* parameters (the driver assumes default values), but if you specify them you can't invert their order. For example you can write **SERHZ**, but not **SERZH**.

The baud rate can be set to most of the usual QDOS values, plus many PC values not originally supported by the QL, up to 256000 baud, if your serial port supports it. For the higher transmission rates, however, it is recommended to use hardware handshake, as otherwise the emulated QL could be too slow to handle the data without losses. A speed of 31250 baud is also available for MIDI applications.

The **TRA** command, present in JS and later ROM's versions, is not implemented. This command allows to specify a character translation table, but usually communication programs don't use this feature.

The Minerva and SMSQ/e extensions to the serial driver are not currently supported.

4.8 Parallel ports

(available only with the “Expanded QL” registration)

You can access to the LPT1: and LPT2: parallel ports of your PC through the PAR1 and PAR2 QL devices.

The driver supports bi-directional ports.

To open a parallel port I/O channel you specify the device name and options with the following syntax:

PARnt

n and *t* represent optional characters that you can add to the name to specify the port number and other parameters. The following table (Table 3) shows the possible values for these parameters:

Parameter	Parameter's meaning	Possible Values	Values' meaning	Default Value
<i>n</i>	Parallel Port number	1-2	Port 1 -2	1
<i>t</i>	CR, EOF translation	R Z F C	No translation Use Ctrl-Z as End-Of-File Use Form Feed as End-Of-File Use Ctrl-Z as End-Of-File and Carriage Return as End-Of-Line	R

Table 3 - PAR device settings

You can omit the *n* and/or *t* parameter (the driver assumes default values), but if you specify them you can't invert their order. For example you can write PAR2Z, but not PARZ2.

You can disable the QL parallel device or change its name in the 'QL Configuration' window.

4.9 TCP/IP

(available only with the “Expanded QL” registration)

Q-emuLator implements **SCK_** and **TCP_** device drivers to allow use of the Windows TCP/IP stack. Only a subset of the socket functions are implemented, but there is enough functionality to run the QDOS port of the Lynx Internet browser, FTP, email and other programs.

TCP/IP is available by default. You can disable or reenble the QL TCP/IP device driver in the 'QL Configuration' window.

5. Configuring Q-emuLator

5.1 The QL Configuration window

The term *configuration* denotes a set of Q-emuLator's preferences and of emulated QL's settings. Only one configuration can be active at any time. You can change some configuration settings through menus. For example you can change the QL's RAM amount through the 'RAM' menu (Figure 4). All the other settings that can't be changed through menus are editable in the 'QL Configuration' window.

To open the QL Configuration window, choose the 'QL Configuration' command from the 'QL' menu (Fig. 5).



Figure 4 - RAM menu

The QL Configuration window has five pages (tabs), named 'ROMs' (Fig. 6), 'Devices' (Fig. 7), 'Serial ports' (Fig. 8), 'Graphics' (Fig. 9) and 'Start-up' (Fig. 10). You access each page by clicking on its label in the top part of the window. Some of the options in the window may be disabled if you have the "Original QL" registration.



Figure 5 - QL menu

The 'ROMs' page allows setting the names and locations of the QL ROM images to be used by the emulator. File paths can be absolute or relative. In the latter case they are relative to the directory containing the Q-emuLator application.

The 'Devices' page allows to enable or disable devices not present in the original QL (for example, the ram disk) and to change their names. Device names can be 1 to 9

characters long. You can also enable or disable support for QL subdirectories.

The 'Serial ports' page defines which PC COM port to attach to each of the four QL SER devices available (SER1 to SER4). A QL SER device can be disabled by setting its output to 'None'. You can

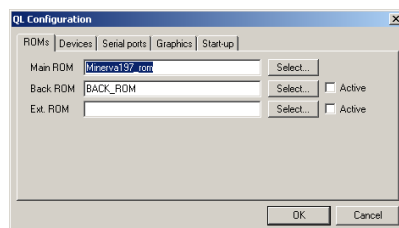


Figure 6 - ROMs

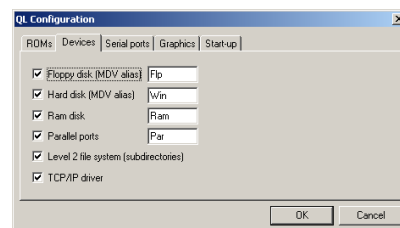


Figure 7 - Devices

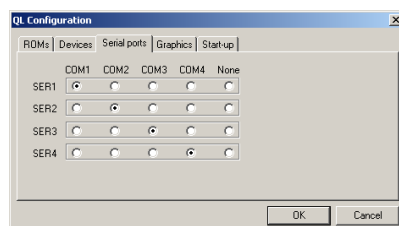


Figure 8 - Serial Ports

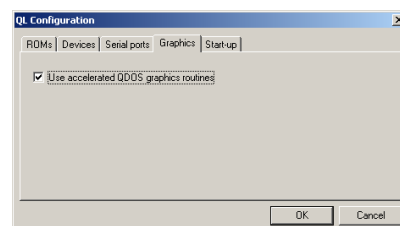


Figure 9 - Graphics

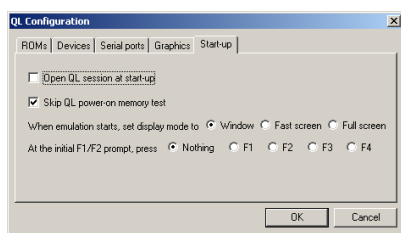


Figure 10 - Startup

route more than one QL SER device to the same COM port, provided you don't use all of them at the same time.

The 'Graphics' page allows to enable or disable acceleration of some QDOS graphics commands (**CLS**, **SCROLL**, **PAN**, **BLOCK**, **MODE**). Enable graphics acceleration for maximum speed, disable it when you run emulation at original QL speed to play games or in case graphics acceleration turns out to be incompatible with your QL software.

The 'Start-up' page allows to specify an optional key to be automatically pressed at the initial QL F1/F2 prompt, which display mode to use when starting emulation and to enable or disable two more Q-emuLator preferences:

- If the '*Skip QL power-on memory test*' option is checked, the QL will not execute the memory test at start-up. The memory test is the first thing that you see when you power on or reset a real QL and is responsible for the coloured dots that fill the screen before the F1/F2 prompt. Checking this option skips this test, so you go directly to the F1/F2 prompt. As a result, the QL's start-up is quite faster, especially with large amounts of QL RAM.
- The '*Open QL session at start-up*' option instructs Q-emuLator to immediately start a QL emulation session when starting or when reading the configuration from file.

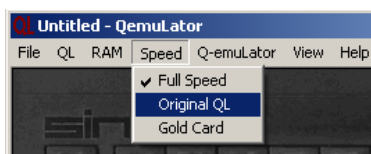


Figure 11 - Speed Menu

Q-emuLator's emulation speed is determined by the 'Speed' menu (Fig. 11). There you can choose between "Full Speed", "Original QL" and "Gold Card". The default is "Full Speed" however this as well as the "Gold Card" options will "demote" themselves to "Original QL" if your registration mode is not "Expanded QL".

5.2 Configuration files

Configuration files store the emulated QL's configuration and some other Q-emuLator's settings.

Configuration files have a ".QCF" suffix (extension).

To start Q-emuLator with the settings specified in a particular configuration file, double-click the icon of the .QCF file, or drag it onto the Q-emuLator.exe icon. If you start Q-emuLator without opening any configuration file, the program uses a default configuration. If you want to change the default configuration, you can do this by saving the desired settings into a configuration file named "default.qcf" and placing it in the Q-emuLator's directory.

Additionally you can create or change configuration settings by creating a file or edit an existing one in a normal text editor (e.g. Notepad). The items and format of the ".QCF" files is described in Appendix I.

You load and save configuration files through commands in the Q-emuLator's 'File' menu (Fig. 12).

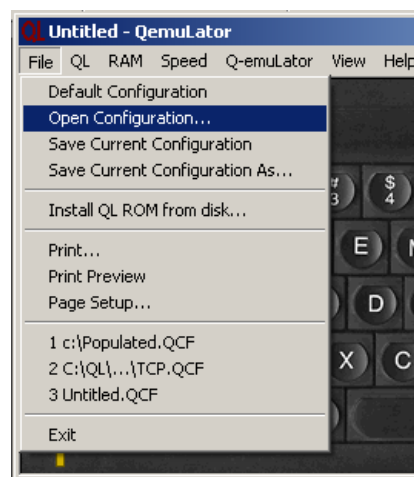


Figure 12 - File Menu

5.3 Using 16KB QL expansion ROMs

A 16KB expansion ROM can also be optionally used by the emulator. If a 16KB expansion ROM is installed on your QL, you can get an image of it with the **"SBYTES FLP1_BACK_ROM,49152,16384"** command. To tell Q-emuLator to use the ROM image, copy it to your hard disk, go to the '*QL Configuration*' window (Fig. 6), press the second '*Select...*' button (the one on the '*Back ROM*' line) and select the ROM image on your hard disk.

Appendix I

Format of .QCF files

A configuration file is a text file used to store the emulated QL's configuration and some other Q-emuLator's settings.

Configuration files are quite similar to Windows' ".INI" files⁷: each line of text describes a different preference or setting, identified by a keyword. The format of each line is:

Keyword = Value

You can, for instance, set the QL's ram amount to 640K with the following line in a configuration file:

Ram = 640K

Configuration files have the ".QCF" suffix. Unlike ".INI" files, you can have multiple configuration files, and you can store them anywhere you like.

The following table (Table 4) describes the keywords you can find and use in configuration files. For each keyword the following table shows the possible values, their meaning, and the default value. The last column shows how you can alter the setting from within Q-emuLator.

Note: Some of these keywords may be ignored by Q-emuLator if you have the "Original QL" registration mode .

Keyword	Value	Meaning	Default	Q-emuLator
RAM	<i>N</i> <i>N K</i> <i>N M</i>	QL ram amount: <i>N</i> bytes <i>N</i> Kilobytes <i>N</i> Megabytes	128K	RAM menu
MainRom	<i>name/path</i>	Name/Path of main (48KB or greater, at address \$0) ROM Image	Minerva	QL Config. Window
BackRom	<i>name/path</i>	Name/Path of back (16KB or greater at address \$C000) ROM Image	TK2	QL Config. Window
BackRomActive	Yes/No or On/Off	Enable back 16KB ROM?	Yes	QL Config. Window
ExpRom	<i>name/path</i>	Name/Path of external (16K at address \$C0000) ROM Image	(none)	QL Config. Window
ExpRomActive	Yes/No or On/Off	Enable external 16KB ROM?	No	QL Config. Window
UseFloppyName	Yes/No or On/Off	Access MDV slots via floppy disk alias?	Yes	QL Config. Window
FloppyName	<i>name</i>	Floppy disk alias/device name	FLP	QL Config. Window
UseHardDiskName	Yes/No or On/Off	Access MDV slots via hard disk alias?	Yes	QL Config. Window
HardDiskName	<i>name</i>	Hard Disk alias/device name	WIN	QL Config. Window
HasRamDisk	Yes/No or On/Off	Enable built-in Ram Disk?	Yes	QL Config. Window
RamDiskName	<i>name</i>	Name of Ram Disk Device	RAM	QL Config. Window
HasParPort	Yes/No or On/Off	Enable Parallel Port Device?	Yes	QL Config. Window

⁷ An ".INI" file is a text file that stores configuration information and/or startup values for a particular program.

Keyword	Value	Meaning	Default	Q-emuLator
ParPortName	<i>name</i>	Name of Parallel Port Device	PAR	QL Config. Window
TCP	Yes/No or On/Off	Enable TCP/IP drivers?	Yes	QL Config. Window
Subdirs	Yes/No or On/Off	Enable Level2 file drivers?	On	QL Config. Window
FastStartup	Yes/No or On/Off	Skip initial QL memory test?	No	QL Config. Window
AutoStartSession	Yes/No or On/Off	Launch emulation session when Q-emuLator starts?	No	QL Config. Window
FirstKey	None F1 F2 F3 F4	No special action on F1/F2 prompt Simulate a F1 keypress at startup Simulate a F2 keypress at startup Simulate a F3 keypress at startup Simulate a F4 keypress at startup	None	QL Config. Window
Sound	Yes/No or On/Off	Sound Emulation State	On	Sound command in the QL menu
Joystick1	Default or None	CTL1 Joystick Emulation State	None	Joystick command in the QL menu
SER1	None COM1 COM2 COM3 COM4	Disable QL serial port 1 (Ser1_) Assign Ser1_ to PC COM1: Assign Ser1_ to PC COM2: Assign Ser1_ to PC COM3: Assign Ser1_ to PC COM4:	COM1	QL Config. Window
SER2	as above	As above but replace Ser1_ with Ser2_	COM2	As above
SER3	as above	As above but replace Ser1_ with Ser3_	COM3	As above
SER4	as above	As above but replace Ser1_ with Ser4_	COM4	As above
SLOT1	Empty DISK_A DISK_B QXL:full_path path	The slot (device) is empty QDOS Floppy in Drive A: QDOS Floppy in Drive B: QXL.WIN file at full_path Windows Filesystem directory identified by path. Path should be an absolute path (ie. c:\ql\)	Empty	Popup menu accessible by clicking on each microdrive slot
SLOT2	as above	As above	Empty	As above
SLOT3	as above	As above	Empty	As above
SLOT4	as above	As above	Empty	As above
SLOT5	as above	As above	Empty	As above
SLOT6	as above	As above	Empty	As above
SLOT7	as above	As above	Empty	As above
SLOT8	as above	As above	Empty	As above
Speed	QL GoldCard Full	Original QL Speed QL with Gold Card Speed Full Emulation Speed	Full	Speed Menu
DisplayMode	Window 640x480 1024x768	Q-emuLator Window Fast Screen Mode Full Screen Mode	Window	QL Config. Window
AcceleratedGraphics	Yes/No or On/Off	Enable QDOS Graphics acceleration?	Yes	QL Config. Window
WindowHeight	Height	Height of Q-emuLator's window in pixels	376	Mouse or maximise icon

Table 4 - .QCF file options

Notes:

1. Keywords are not case sensitive and can contain spaces. For example, you can write 'FloppyName', but also 'FloppyNAME', 'FLOPPYNAME' or 'floppy name'.
2. Paths can be absolute or relative. In the latter case the path is relative to the working directory or to Q-emuLator's directory

As an example, the following is the content of a configuration file that specifies the default values:

```
Ram=640K
MainRom=QL_ROMs\Minerva189_rom
BackRom=QL_ROMs\TK2_rom
BackRomActive=Yes
ExpRom=
ExpRomActive=No
UseFloppyName=Yes
FloppyName=Flp
UseHardDiskName=Yes
HardDiskName=Win
HasRamDisk=Yes
RamDiskName=Ram
HasParPort=Yes
ParPortName=Par
TCP=On
Subdirs=On
Speed=Full
FastStartup=Yes
AutoStartSession=No
FirstKey=None
DisplayMode=Window
AcceleratedGraphics=Yes
Sound=On
Joystick1=None
SER1=COM1
SER2=COM2
SER3=COM3
SER4=COM4
Slot1=Empty
Slot2=Empty
Slot3=Empty
Slot4=Empty
Slot5=Empty
Slot6=Empty
Slot7=Empty
Slot8=Empty
WindowHeight=376
```

You are encouraged to create a personalised default configuration file that Q-emuLator will load each time it starts: customize your settings through the Q-emuLator menu commands (most settings are in the QL Configuration window), choose the *Save Configuration* command from the 'File' menu, and save the file in the Q-emuLator's directory, naming it '**default.qcf**'.

Appendix II

Format of QL files stored in the PC file system

QL files have a special piece of information associated with them, called the 'QDOS file header'. The header stores such information as the file name and whether the file is an executable program.

Q-emuLator for Windows stores part of the header at the beginning of files. The header is present only when it is useful, ie. only if it contains non-default information.

The header has the following format:

OFFSET	LENGTH(bytes)	CONTENT
0	18	"J!QDOS File Header"
18	1	0 (reserved)
19	1	total length_of_header, in 16 bit words
20	length_of_header*2-20	QDOS INFO

The first 18 bytes are there to detect whether the header is present (ID string).

The headers Q-emuLator supports can be 30 bytes or 44 bytes long (the value of the corresponding byte at offset 19 is either 15 or 22). In the first case, there are 10 bytes with the values present in bytes 4 to 13 of the 64 bytes QDOS header. In the second case the same piece of information is followed by 14 bytes containing a microdrive sector header, useful for emulating microdrive protection schemes. Additional header information (file length, name, dates) is obtained directly from the file through the host file system.

Some QL programs to translate between QDOS and Q-emuLator for Windows file formats are included in the Q-emuLator package.

The translation is transparently performed when you move files between QDOS floppy disks and Windows directories through Q-emuLator.

There is no need for you to use the provided conversion utilities, except when you want to copy files from a copy-protected microdrive to a floppy disk.

Note: The Mac version of Q-emuLator uses a different scheme: the QDOS information is stored in the file's resource fork. (On Macintosh all files have both a data fork - corresponding to a Windows, DOS or UNIX sequential file - and a resource fork, containing structured information called 'resources'). The latest version of Q-emuLator for Mac OS can read files created by Q-emuLator for Windows, but not vice versa.

Appendix III

Q-emuLator support site

You can find news and information about Q-emuLator (both Mac and Windows versions) at the following World Wide Web address:

<http://users.infoconex.com/daniele/q-emulator.html>

Appendix IV

E-mail address for feedback

Please send feedback about Q-emuLator (bug reports, questions, comments, suggestions, etc.) to the following e-mail address:

qemulator@kagi.com

Appendix V

QL ROMs supported by Q-emuLator

The following table (Table 5) shows the compatibility status of the current (v.2.3.3) Q-emuLator with available QL-compatible ROM images.

ROM Name	QDOS Version	SuperBasic Version	Author	Compatible	Size	Notes
AH	1.02	AH	Sinclair	●	48KB	
TB	1.03	TB	Sinclair	●	48KB	Reduced keyboard functionality
JM	1.03	JM	Sinclair	●	48KB	
JS	1.10	JS	Sinclair	●	48KB	
JS-4M	1.10	JS	?	●	48KB	Better to use the regular JS instead
JSU	1U10	JSU	Sinclair	●	48KB	
MG	1.13	MG	Sinclair	●	64KB	
MGUK	1Σ13	MGUK	John Alexander	●	48KB	
ULTRAMG	?	?	Ultrasoft	○	48KB	Crashes
MF	1.14	MF	?	●	48KB	RAM must be set to 1920KB or less
MGE	1E14	MGE	Sinclair	●	48KB	
MGG	1G13	MGG	Sinclair	●	48KB	
MGR	1R13	MGR	Sinclair	●	48KB	
MGI	1I13	MGI	Sinclair	●	48KB	
Minerva	1.6x to 1.98	JSL1	Q-View	●	48KB	
M89-T6	1.89	JSL1	H.P. Reckenwalt	●	80KB	
Tyche	2.05	TY05	?	●	64KB	RAM must be set to 1920KB or less

Table 5 - QL ROMS supported

Appendix VI

Debug_68k Users' manual

Debug_68k.dll is a component that can be used by emulators to add a debugger for an emulated Motorola 68000 CPU.

1. Installation

Put the debug_68k.dll in the directory containing the emulator executable file.

2. The debug_68k window

When you invoke the debugger (press F11 in Q-emuLator when QL software is running), or when execution hits a breakpoint, the debug_68k window pops up (Figure 13).

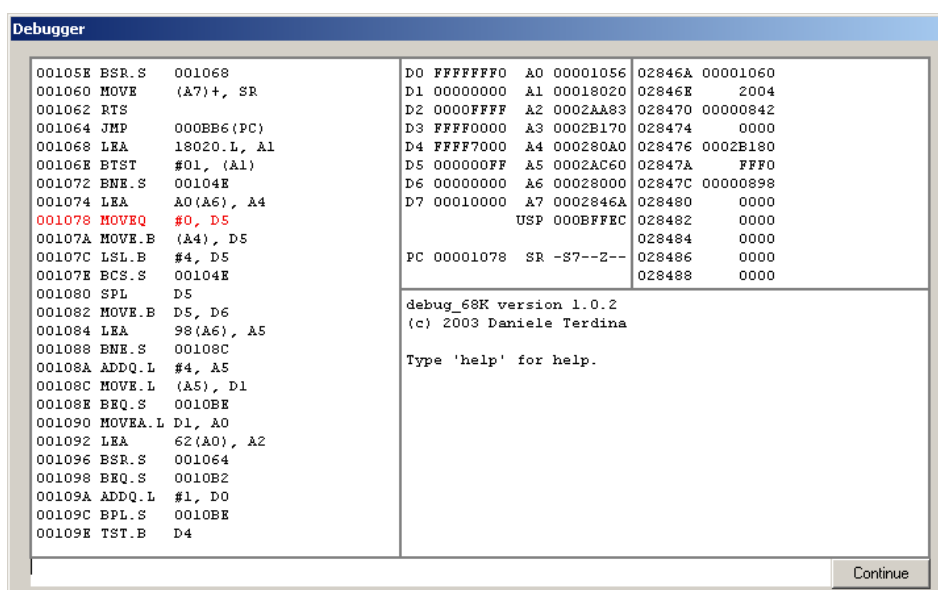



Figure 13 - Debug_68K Window

At the bottom of the window there is a text input box (it looks like an horizontal white bar), with a "Continue" button to its right. Pressing the button closes the window and resumes execution.

The upper part of the window is divided in four parts. From left to right they are the disassembly area, the register display and the stack display, and the output area which is under the register and stack displays.

When you digit a debugger command on your keyboard, it appears in the input area, and it gets executed when you press the  key. When the command gets executed, the command and possibly its results are displayed in the output area.

All numeric values displayed in the debug_68k window are always hexadecimal.

Addresses are displayed with six hexadecimal digits, as this covers the 16MB address space of the 68000 processor.

3. The disassembly window

This window shows the disassembly for the 68000 code corresponding to a small area of memory. There's one machine code instruction per line (or a `DC.W` followed by a 2-byte value for codes that don't represent a valid Motorola 68000 instruction), preceded by its starting address in memory.

A line with red text in the disassembly area shows the machine code instruction that will be executed next.

After execution of one or more 68000 instructions, the range of disassembled addresses will change to show the code at the next `PC` (Program Counter), and some code before and after the next `PC`.

You can scroll the code with the up arrow, down arrow, page up and page down keys.

To display the code around an address `<addr>`, you can use the command `"u <addr>".` ("U" stands for "Unassemble").

If you hover the mouse over some of the operands that are specified through addressing modes, a pop-up will show the address value. The address value is computed according to the current register values, which may be different than the value it will have when the instruction will get executed. The exception is the `PC` register: when used in addressing modes, it takes the value of the instruction's address instead of the current value of the `PC` register.

4. The register display

Red text in the register display area highlights values that have changed after execution of the last instruction or group of instructions.

The bottom line in the register window may either be blank or display emulator-specific status. (For example, in Q-emuLator I use it to show whether special system code is executing, like system calls or interrupt handlers.)

To change a register value, type in `"<reg>=<value>"` (e.g. `"D0=0"`). `<value>` can be any valid expression (type `"expr"` for expression help).

5. The stack display

The upper right part of the debugger window shows the content of the stack, starting with the current stack pointer address on top. The content is shown in words, or longs when the values look like valid addresses.

6. Debugger commands

Type `"help"` for a list of available commands.

Note that there is currently no "clear breakpoint" command. However, you can use `"bp <bpNumber> 0"` to clear a breakpoint.

7. Stepping through code

Use the **F1** key to step to the next instruction. **F1** steps into function calls.

Use **F2** to step to the next instruction, but step over function calls as if they were a single instruction.

F3 works like **F2**, with the additional rule that if the instruction is a conditional branch, then it executes instructions until the next PC is reached (i.e. 1 instruction if the branch is not taken, or any number if it is taken). It is similar to setting a breakpoint to the next instruction and then restart emulation. This is especially useful for the branch instruction that ends a loop, when you want to go to the first instruction after the loop is completely executed.

Shift F3 is complementary to **F3**: For conditional branch instructions, it runs until the branch target address is reached.

To run unconditionally (or until execution hits the next breakpoint), use **F11**, or type 'g', or press the 'Continue' button.

8. Using debug_68k.dll in emulators

It's easy to add debugging capabilities to an emulator of a M68000 based system by using debug_68k.dll. If you are an emulator author, write to qemulator@kagi.com to receive instructions, the header file with the interface, and sample glue source code. Use is free provided credit for it is given in the emulator's documentation, and the debug_68k.dll and this document are distributed in their original form with the emulator.