

Q-DOCK

The Q-Dock program and documentation are Copyright (C) 2014 Dilwyn Jones. As of 19/08/17 the Q-Dock software is freeware.

The sprite graphics supplied are public domain and may be freely copied or used in your own programs if you wish.

Q-Dock is an example of a program dock system. A dock is a kind of "tray" in which programs are parked and represented by icons. You can just click on these icons to start the program. It makes life so much easier than having to manually type in EXEC commands!

The dock panel itself is a simple graphic file, which looks like a shelf, a tray, or similar item. For Q-Dock, the graphic is a normal sprite graphic of a given size. A few are supplied, many more can be downloaded from the Q-Dock page on my website. Or, you can design your own in a QL sprite editor able to design QL sprites of size 512x64 pixel, or create them in a graphics program and convert the graphics to QL sprites using a program such as Marcel Kilgus's PNGconv program.

The program icons are also standard QL sprite graphics. Several are supplied, many more can be downloaded from the Q-Dock page on my website. Or, you can design your own using any QL sprite editor - the current version of Q-Dock uses 32x32 pixel sprites.

To distinguish between the two types of sprites, dock sprites are given the filename suffix "_dck" (although "_spr" can also be used), while program icons are given the filename suffix "_spr". Both types are standard QL or GD2 format sprites, despite the different name conventions, the only major difference being size.

Requirements

Expanded memory (minimum 512K).

Pointer environment and Window Manager version 2.00 or later on QDOS, or SMSQ/E operating system v3.00 or later.

Toolkit 2 (which is built into SMSQ/E and some floppy disk systems).

Desirable (but not absolutely essential)

High colour system - 8-bit colours or better (e.g. Aurora, QPC2).

Hard disk system.

Features

Up to 14 program icons per dock, up to 9 docks.

Auto-hide option - program wakes up when you place the pointer at the edge of the screen (distance from edge configurable). Q-Dock will hide itself if the pointer is held outside its window area for a given length of time.

Icon shadow and reflection options for enhanced graphical effects.

Dock may be moved around to preferred screen location.

Optional pop-up bubble helps.

Store programs, or data files (if File Info 2 used), or command entries - many options.

Auto-add option, allowing program to scan your hard disk and add all executables found (space permitting).

Auto-add icons associated by name, e.g. if myprog_exe and myprog_spr are present, myprog_spr is used by default (can be changed) as the icon for myprog_exe.

Jobs screen allows programs to be picked, stopped, etc.

Exec screen allows programs to be started manually if not already added to Q-Dock (e.g. rarely used programs).

Choice of dock and icon graphics supplied, more available from my website.

Optional transparent program window, subject to operating system limitations, allowing transparent (alpha-mask) dock sprites to be used.

Uses Window Manager 2 colour themes for its menus.

Hopefully very, very simple to use!

Getting Started

If you received Q-Dock on floppy disk, just copy all of the files into a directory on your hard disk system or onto an empty floppy disk. NEVER run Q-Dock from the master disk supplied - it may work, but risks damaging your master copy.

If you received Q-Dock as a zipped file, whether downloaded or as an email attachment, transfer the zip file to your QL or emulator and unzip it there to a directory of your choice. NEVER unzip in Windows using programs like Winzip or 7-zip, Windows doesn't understand QL executable file headers and will damage the executables, which will subsequently fail to execute properly, giving errors such as "bad parameter" due to the loss of file headers.

If you are unfamiliar with using Unzip on a QDOS or SMSQ/E system, there are articles on the subject of zip/unzip and general PC/QL file transfer on my website, at www.dilwyn.me.uk/gen/pcqlxfer/index.html

Once installed to the directory of your choice, you need to configure the program to tell it where its files are located. Systems which include the Home Directory thing allow Q-Dock to check where it was executed from, so if Q-Dock detects that the Home Directory thing is present, it will assume all its setup files are in the same directory as it was started from. Only two settings

need to be configured, the directory and whether or not to make use of the Home Directory thing to override the configured default drive. All other settings are changed from within Q-Dock itself.

Use the Config program to configure Q-Dock - just EXEC the Config program and tell it to load qdock_obj, change the two items if required and save the altered settings.

That's it, Q-Dock is now ready to use.

Starting Q-Dock

Simply execute the program file called QDOCK_obj.

```
EXEC WIN1_QDOCK_OBJ
```

One of two things will happen...

1. You see absolutely nothing. Assuming there was no error message, it is likely that the auto-hide feature is turned on, so Q-Dock hides itself until the pointer is moved to the bottom edge of the screen. Move the pointer to the bottom of the screen and Q-Dock should pop up automatically. Unless you now move the pointer to Q-Dock within a couple of seconds, it assumes you selected it by accident and hides itself again, out of the way.
2. The Q-Dock display appears.

Auto-Hide Limitations

The auto-hide and window transparency features are not problem free. Pointer environment is not fully geared up to advanced graphical things like this, so you may find funny effects happening from time to time, depending on how Q-Dock windows are partially hidden or buried and whether or not background redrawing is turned on (on systems supporting that). Occasionally, even if Q-Dock is hidden you may have to manually CTRL-C back to it to get it to reappear when auto-hide is on. You might like to set up a hotkey to pick Q-Dock if it has got buried under something and can't pop up by itself - ERT HOT_PICK('d','qdock') is an example of such a hotkey. You'll soon get used to these little limitations as you get used to Q-Dock.

Stopping Q-Dock

You can stop Q-Dock in one of two ways:

1. Press the Esc key and a confirmation (Yes or No) menu appears in case you pressed Esc by mistake.
2. Bring up the commands menu by pressing the F3 key and select the Quit command.

If Q-Dock has become unresponsive for some reason, you can go to BASIC and remove the Q-Dock job with the command RJOB 'qdock'

Using Q-Dock

The Q-Dock display is fairly minimalist in that it occupies an area of only 512 pixels wide and 64 pixels high. Within that, you have:

1. the dock area itself (the part in which the icons sit)
2. the dock switching area consisting of 9 coloured or numbered small buttons which let you select any of the 9 docks. The nine docks can be given names to help identify them if you have reserved some docks for particular groups of applications. The names can be shown in the bubble help (assuming it's enabled in the Settings menu) simply by hovering the pointer over one of the dock switch icons. These icons may have their appearance changed in the Settings menu, while the dock names can be changed using the Docks command in the Settings menu.
3. four command buttons to the bottom left, which let you move the dock around the screen (the usual MOVE icon), a green refresh button to redraw the display if it becomes corrupted, an 'F3' button to bring up a Commands menu, and a small icon which looks like the red, white and black mode 4 windows on a QL - this is used to 'pick' or go to the BASIC command line. These four buttons may be hidden if the command view has been turned off in the Settings menu.
4. the rest of the Q-Dock display - the significance of this is that if you click with the mouse in this area, it brings up the commands menu.

Terminology: A Hit means left click mouse button or press space on the keyboard while the pointer is over an item. A Do means right mouse click or press ENTER on the keyboard while the pointer is over an item.

A Hit on a program icon in the dock simply executes a copy of that program. Multiple copies can usually be started, so be careful how many times you click on an icon if the program is slow to start!

If you Hit on where an icon might be situated, but none is there (the slot is empty) you get the Add item menu - see below.

A Do on an icon brings up a context or actions menu, providing actions relevant to the current program icon:

Add	Delete	Edit	Exec
Move	Pick	Print	Reset
Rjob	Set	Set & Pick	Switch

Add:

Add a new program icon to the dock if the icon slot is empty, otherwise you are invited to edit the existing entry. The screen layout is the same for both add and edit, just that the entry form is filled in with the details of the existing program entry if editing.

The Add (or Q-Dock Program Settings) form looks complex since there are so many options. However, most of these entries are not used for the majority of programs, so you will find that

most programs only need about four entries - the name of the program, filename, icon filename and job name, and sometimes the program is intelligent enough to provide some of these for you, e.g. when you enter the program filename, it will try to extract the job name automatically and look for a matching icon too (e.g. if the program filename is myprog_exe, Q-Dock looks in myprog_exe to see if it can extract the job name, then looks for an icon called myprog_spr either in either the Q-Dock directory or the directory containing myprog_exe - you can amend these later if the automatic names are not the ones you really want).

The full list of everything you can enter for a particular program is as follows:

Program name	This appears in bubble help, for example and serves to identify the icon
Filename	Filename of the program, including drive and directory
Job name	The job name, used to PICK or RJOB a program
Command line	Passed to a program as an option command line if a filename was given or sent to the BASIC command line as a BASIC command if no filename specified
Icon filename	Filename of icon used for the program in a dock
DATA_USE	DATA_USE setting for this program (if required)
PROG_USE	PROG_USE setting for this program (if required)
DEV_USE 1	DEV_USE string for DEV drive 1 (if required)
DEV_USE 2	DEV_USE string for DEV drive 2 (if required)
Psion K	Number of kilobytes to allow memory-hogging programs like Quill - usually a value from 128 to 512 kilobytes is adequate
FLP_USE	three character string to rename the FLP disk driver device name, e.g. FLP_USE mdv
RAM_USE	three character string to rename the RAM disk driver device name, e.g. RAM_USE flp
WIN_USE	three character string to rename the WIN disk driver device name, e.g. WIN_USE flp
DEV_USE	three character string to rename the DEV driver name, e.g. DEV_USE flp
PAR_USE	three character string to rename the PAR parallel port driver name, e.g. PAR_USE ser
NFA_USE	three character string to rename the NFA (Native File Access) driver name in SMSQmulator, e.g. NFA_USE win
SFA_USE	three character string to rename the SFA driver name in SMSQmulator, e.g. SFA_USE win
QXL_USE	three character string to rename the QXL driver name in SMSQmulator, e.g. QXL_USE win
SDC_USE	three character string to rename the QL-SD device driver name from SDC to something else, e.g. SDC_USE win

Some programs cannot be configured to run from anything other than their hard coded device name, e.g. before QL hard disks became commonplace, it was quite common for programs to be sold which were only able to run from microdrive or floppy disk. In this case, they can only run from device names MDV or FLP. Suppose, for example, that we have copied a program from microdrive cartridge to floppy disk, but not converted the MDV device references to FLP, one way around this in BASIC is to use a command such as FLP_USE MDV to change the FLP

device driver name to the microdrive equivalent MDV in order to fool the program into running from floppy disk. Q-Dock can do this for you - in the FLP_USE box, type in the three letters MDV.

There are programs around which will search for and replace those names with another three character string, such as WIN to help convert them to run from hard disk. However, if you wish to place these programs in directories (hard disks can quickly become cluttered if you place everything in the root directory without using directories), you will need to be imaginative and somehow fool these programs into running from hard disk. One solution, if your system includes the DEV device driver, is to use something like this in BASIC - I'll assume for this example we are trying to use a program called MYPROG_EXE from a directory called WIN1_FOLDER_, and that it is set up to run only from FLP1_ and FLP2_:

DEV_USE 1,WIN1_FOLDER_	make it access win1_folder_ when it thinks it's accessing flp1_
DEV_USE 2,WIN1_FOLDER_	ditto for flp2_
DEV_USE FLP	make references to FLP1_ or FLP2_ actually access DEV1_ and DEV2_
FLP_USE FDK	rename "real" flp device to respond to a name like fdk1_ and fdk2_ just so we can still access them if we need to!

Once you understand how the BASIC commands work, you will see how to enter the details in Q-Dock's program settings screen. Obviously, if your system has no DEV driver, you can't use this, BUT, however, the DEV device driver is available as a stand-alone extension from my website at www.dilwyn.me.uk/tk/index.html

If you have told Q-Dock to rename the device drivers in this way, it affects all running programs (just like the BASIC command equivalents do). So after you have finished with the program concerned, you will need to reset the settings by using the **Reset** command in Q-Dock.

If this all sounds too complex, remember that the vast majority of programs only need the program name, filename, job name (which is usually extracted automatically) and an icon filename.

Once you've finished entering the details in the program settings window, click on the OK button to create the icon in the dock.

If you select the Add command by mistake, just click on the X icon in the top right to return to the dock.

There are special and advanced forms of the Add command you can use when the need arises.

If you are a QPC2 user, one form allows you to use the QPC_EXEC command facility to start a Windows program from within the emulator. Using this allows you to start a Windows executable program, a Windows data file, or even a website, all from within Q-Dock. Don't worry if you don't know what this means, you can safely ignore this part if you wish.

By starting the command line entry for a program with the four characters "QPC:" (without the quotes) the Filename and Command Line fields are passed to a QPC_EXEC command to call

the PC files.

There are three separate ways of calling this:

1. Just a filename, in which case the Windows program is called with no arguments:

Filename: notepad.exe

This is equivalent to QPC_EXEC "notepad.exe" in BASIC on QPC2 and simply starts the Notepad text editor in Windows. Since we have not specified a drive or path, it assumes that the program can be found in the default Windows path mechanism. If the filename given is a data file such as a Word document, the Windows file association mechanism will try to find the relevant default program for these files, e.g. for Word document files, it would try to start Microsoft Word or Wordpad, for example.

Special case: if the filename is that of a SuperBASIC or SBASIC program with a name ending in _bas or _sav, upon execution of that program's icon, aq menu comes up asking how you'd like it to be executed:

1.1 If in SBASIC, the BASIC program can be executed as a daughter SBASIC job, like starting a BASIC program with an EX command in SBASIC (this is not available if running in SuperBASIC on a QDOS system).

1.2 Via FileInfo 2 if in use on this system

1.3 Sent as an LRUN command to BASIC by picking BASIC and sending the LRUN command to it. Note that timing limitations prevent the PTOF command of Easyptr working properly on some systems - this is the most likely cause of failure (this happens on my own QPC system!)

2. Just a command line, in which case the Windows file is called with no arguments:

Filename:

Command line: myfile.txt

This is equivalent to QPC_EXEC "", "myfile.txt" which may not be very useful like that (it will probably just pop up the QPC2 folder in Windows Explorer!), so gets converted to QPC_EXEC "myfile.txt" to ensure Windows searches for the program associated with .txt files and so loads the file into the default text editor, normally Notepad.

3. Both a filename and command line, in which case the Windows program is called with the command line argument:

Filename: mspaint.exe

Command line: qlgraphics.bmp

Here, a converted QL screen copy which has been converted to a .bmp Windows graphics file is loaded into the Paint program in Windows.

4. Some Windows programs support command line switches, allowing you to set certain options in the Windows program as well as actions like loading a file. A useful one sometimes is the "/p" switch which many programs use to print a file.

Filename: mspaint.exe

Command line: qlgraphics.bmp /p

This loads the qlgraphics.bmp into Windows Paint and the /p switch makes it start printing if

the version of Paint supports that. One way of printing a QL screen or other QL graphic as long as you have the means to convert them to a file format which Paint can handle, e.g. use my BMP program to convert the QL screen or PIC file to a .BMP file first. I'm sure you can come up with some very imaginative ways of calling Windows programs in this way!

Other special options for the Filename and Command line fields are:

1. Put a directory name in the Filename field. Q-Dock will open that directory using the file handler specified in the Settings menu, such as the QPAC2 Files menu. A quick way of getting a list of files.

2. Specify a non-executable file in the Filename field. This is processed via the FileInfo 2 thing if it is installed on your system. If you have never used FileInfo 2 before, it is a system whereby filename endings are associated with specified programs, so that the right program is used to "execute" a non-executable file, e.g. _doc document files should be started with Quill. FileInfo 2 is available to download free from most sources of free QL software.

3. Leave the Filename field blank and put a command in the Command line field plus an optional job name in the job name field. This is interpreted as a command string to send to a specified job name. Leave the job name blank and the command gets sent to the main BASIC interpreter command window. A backslash '

t the end of the command gets converted to a linefeed, equivalent to pressing ENTER in BASIC:As an example, here's a version that ends up doing a DIR FLP1_ in BASIC:

Filename:

Job name:

Command line: DIR FLP1_

Note that the command is sent using the Easyptr PTOp command system. Timing issues when the PTOp command is used to send a string to a job mean it doesn't work correctly on some systems in the current version (including my QPC2 system unfortunately for me).

As I said above, these extended options can get pretty tricky to understand, so ignore the advanced options if you wish.

Delete:

Deletes the program icon. You will be asked to confirm the delete. There is no way to get it back afterward, so select with care!

Edit:

Same as **Add**.

Exec:

Start the program clicked on. Exactly the same as a simple Hit on the icon, but this is provided so that if you accidentally Do instead you can launch the program from the menu by clicking on Exec to avoid having to press Esc to cancel the menu first. It's also useful if you want to start a second copy of a program, since a normal Hit on an icon a second time while a program is

running picks that program instead of starting another copy (deliberate, so as to minimise the risk of starting two copies by mistake, plus the convenience of picking a program with a simple Hit on an icon if it is already running, which some docks on other computer systems are able to do).

Move:

Move the icon to another slot. Useful if you want to rearrange the order of the icons in a dock. First, it asks which dock to move the icon to - click on the dock number required, or click on OK to change position in the current dock, or click on the ESC icon to quit without moving the icon.

Pick:

If there is a copy of the program running already, clicking on the Pick command for that icon will make the QL switch to that program.

Print:

Prints a list of the programs set up in Q-Dock together with their settings. Useful reference if you can't remember which dock you placed an icon in, or wish to check if any of the program icons changes any of the xxx_USE settings, for example.

Reset:

Resets the various xxx_USE to default states if changed by any of the programs. For example, Q-Dock does the equivalent of FLP_USE FLP, WIN_USE WIN, DEV_USE DEV and so on. Remember that all of these take global effect for all programs, just like the equivalent BASIC commands would.

After you have issued a Reset command to change the settings back to normal, you may have left a program running which would need the settings restored next time you Pick that program. This can be done with the Set or Set & Pick commands which will restore settings for the program concerned without starting another copy.

Rjob:

Removes a running copy of the program associated with the icon you clicked on. Uses the equivalent of the BASIC command RJOB "Job name" - this is why the Job name entry is required when you add or edit a program for Q-Dock. Be careful if you have more than one program with the same job name though, it is all too easy to remove the wrong program and lose unsaved work!

Set:

Apply the settings specified for the icon you clicked on, e.g. the FLP_USE, WIN_USE, DATA_USE etc settings.

Set & Pick:

Apply the settings specified for the icon you clicked on, then Pick that program.

Switch:

Switch to one of the other docks - choose the one you want from the menu which appears.

Commands Menu

This menu can be shown by pressing the F3 key, or by clicking in a part of the dock window not occupied by an icon and not by an empty icon slot.

My preferred place to click is just to the left or right of the dock switching icons when I don't feel like pressing F3!

The Commands menu looks very similar to the icon context menu described above, and like that one the commands can be selected either by clicking on the command (Hit or Do) or pressing the underlined key for the command you require.

Add	Docks	Move	Refresh	Sleep
Auto-add	Exec	Print	Reset	Switch
Copyright	Jobs	Quit	Settings	Zap items

Add:

Lets you add a new program icon to the dock. Broadly the same as clicking on an empty icon slot to add an icon there, the only difference is that this command locates the first free icon slot in the current dock. If the current dock number is full of icons, an error message is shown and you'll need to switch to one of the docks which has a free icon slot. With up to 14 icons per dock and a total of 9 docks, it is unlikely that you will have used all 126 program icon slots!

Auto-add:

This is an automated version of the Add command, which looks through the drive/directory you specify for all the executable program files it can find and add them to any free slots it finds in the docks.

First, it asks for the name of the drive/directory it should scan to search for executable program files. This can be an entire drive (which may take a long time if you have a full hard drive) or if like me you are in the habit of putting copies of your most used programs together in a single directory on your hard disk you can speed up the process by just scanning this directory. The search routine is recursive, which means it will also scan sub-directories if there are any. This is most useful when you tell Q-Dock to scan the entire hard disk drive for program files. If you make a blank entry, the command aborts and returns to Q-Dock (useful if you selected the command in error).

Auto-Add checks to see if there are any existing icons in any of the docks. If any are present, it asks if you wish to delete these, to help avoid duplication. Equally, if all docks are full of icons, you will get a message saying there is no room to add any more program icons.

Automating the search should be thought of as a time-saving first step, since the search is not particularly intelligent in terms of creating settings for the programs it finds. What it will do is

add icons if it finds suitably named sprite graphics. It only uses 32x32 pixel QL sprites. They can be mode 4 (QL), mode 16 (Aurora 256 colour), mode 32 (QXL/QPC/SMSQmulator), mode 33 (Q40/Q60) or even mode 64 sprites in theory, although those haven't been tested as I don't have any to test!

The convention used for names is to take the filename of the program, strip off any extension such as `_exe`, `_obj`, or `_task` and create a new name with `_spr` added, for example `myprog_exe` would become `myprog_spr`. Programs with names which have no extension, such as `Archive`, will simply have `_spr` added, so for `Archive`, Q-Dock will look for an `archive_spr` sprite file.

Q-Dock then looks in the same directory as that which contains the program for a sprite with this name and uses that for the icon if found. If not found, Q-Dock will also look in its own system directory for a suitably named sprite and if found there, that one is used. If neither is found, the icon graphic is left blank and Q-Dock will use one of its own primitive inbuilt mode 4 sprites (which should work in any screen mode).

Q-Dock will try to fill in the Jobname field if it can find the jobname within the executable program file.

If Q-Dock finds programs called `Quill`, `Archive`, `Abacus`, or `Easel` it will also fill in the "Psion K" field with a value of 128 by default - you can manually edit this later to a different value if required.

Once Auto-add has finished, the next thing you should do is go through the icons to do any hand-editing required. Q-Dock is not very intelligent in terms of settings required for older QL programs in particular, so you will need to check if any of the `xxx_USE` settings need editing, change sprites used for the icons perhaps, amend settings for the Psion programs if you wish to use those, plus of course amend the item names, which are shown as a bubble help window when you hover the pointer over a program icon in one of the docks. Q-Dock will usually just use a name derived from the filename of the program, possibly without any filename extension, possibly truncated to fit. In other words, if it finds a program called `MyProgram_Task`, it might create a name such as `MyProg` - you will most likely need to change Q-Dock's choice of program names!

Copyright:

Brings up a small copyright notice screen, also showing a program version number and how to contact me for help should you need it. Click on OK to return to Q-Dock.

Docks:

Lets you assign names to any of the docks. By default the names are simply 'Dock 1' to 'Dock 9'. Just click on any of the existing names or press the underlined key shown alongside to change them to something more meaningful if you wish, e.g. Dock 1 could be changed to Games, Dock 2 to Utilities and so on.

Exec:

Lets you enter a filename of a program to execute manually. Useful for programs which are not used often enough to justify parking them in a dock. Just make a blank entry to quit if you select this command in error.

The Exec command can work in one of two ways, depending on how you selected it. A Hit on

the command brings up a simple type-in menu - this may be useful if you are selecting and stuffing the filename from another program. A Hit on the Exec command brings up a File Selection menu in the usual style for my programs - select one of the three letter drive names, a drive number, click on a filename preceded by a '>' symbol to enter that directory and so on. Click on the name of the file execute and Q-Dock will try to start it. Non-executable file types can be executed via FileInfo2 if that is in use on your system.

Jobs:

This command brings up a job control menu which lets you Pick (switch to) a running program, or Rjob a program (stop a running program, like the RJOB command in BASIC). The menu shows a list of running programs together with job names, priorities, approximate size and so on. When you enter the menu, Pick is pre-selected as that is safer than Rjob which has to be manually selected to prevent accidental closure of running programs which may cause loss of unsaved files. So, select the Pick or Rjob items at the top first, depending on which you need, then click on the name of the file you want to pick or remove. Be very very careful when removing jobs, especially if there's more than one instance of a program with the same name, it may be hard to identify which will be removed. Of course, you don't have to use either Pick or Rjob, this menu is quite useful in identifying programs running and roughly how much memory each takes and what's running at which level of priority so you can get an idea of which programs use what system resources if you think you may be running low on memory or there seems to be one program running very fast at the expense of others. You can refresh the list by clicking on the refresh icon between the JOBS heading top centre and the Escape icon top right.

Move:

This brings up the normal move icon to allow you to move the Q-Dock position around the screen to the preferred location. The new position is saved in the settings file and remembered for the next time you use Q-Dock. The reason for its inclusion here is that the Move icon on the left of the Q-Dock display is quite small and tricky to hit accurately if you don't have a mouse, or have an unsteady hand.

Print:

This command will make a printout of the programs and settings used in Q-Dock. It can be a handy reference document if things are not working quite as you'd like them to since it is quicker to scan through a few pages of program settings than go through them one by one manually in Q-Dock with the Edit command. Can also be useful if you wish to copy settings for one program to another, for example. The listing is organised by dock number, then within that by icon number. The command asks you where to print to (e.g. PAR or SER1), although you can enter a filename if you'd like to print to a file to view it later in an editor to save paper. Make a blank entry for the 'Print to' name if you wish to re-run to the menu. Note that the heading includes a printout date for reference, so make sure your QL clock is showing the correct time and date if you think this might be useful.

Quit:

Unsurprisingly, this quits from Q-Dock, shutting down the program. Unlike pressing ESC from the main dock, there is no confirmation option so select with care!

Refresh:

Redraws the display if it's been corrupted. Deficiencies in the transparent window routines mean you can pick up stray background colours from time to time when using a theoretically transparent Q-Dock window. This lets you redraw the display tidily. This can also be done with

a double click on the Move icon on the left of the dock, or clicking on the green down arrow alongside it.

Reset:

If some of your Q-Dock programs set things like DATA_USE, PROG_USE, FLP_USE, DEV settings and so on, Reset is a quick way of restoring normality - it goes through the list of settings, applying normality commands like FLP_USE FLP, DEV_USE DEV and so on, cancelling these settings. As with the same commands in BASIC, the effect is generally global, so it will cancel the current settings for all currently running programs. You can of course restore the settings for a particular running program by right clicking on its icon in one of the docks and choosing the Set or Set&Pick commands.

Settings:

This does most of the program configuration, such as choosing dock graphics, switching bubble help off and on and toggling autohide off/on. There are some many options, it has its own section in this manual further on.

Sleep:

Puts Q-Dock to sleep in the QPAC2 button frame. I'm not sure why you might want to do this, the only reason I can think of is to prevent it popping up needlessly if you have autohide switched on and you need to use a full-screen program like a graphics program without Q-Dock popping up every time you draw near a screen edge. Sleep requires that the QPAC2 button frame is installed. If it isn't, the only thing Sleep can do is try to minimise the Q-Dock display, but this doesn't work on all systems.

Switch:

Select a different dock to switch to - there are 9 to choose from, each with up to 14 icons. Just click on the dock name shown to select one of the nine, or press the key indicated by the underlined character to the left of the name. Normally, you could switch more quickly by clicking on one of the nine little icons at the bottom of the docks, but since these are quite small it can be fiddly if you have to use cursor keys, have a clumsy mouse, or an unsteady hand, so this menu is provided as a secondary option. Hint: probably the quickest and easiest ways to switch docks without having to bring up this menu is to press the keys 1 to 9 to select the relevant dock number 1 to 9 in the main Q-Dock screen.

Zap Items:

Used to clear out icons from any one or all docks when they get cluttered or full of programs you no longer need. Use with care! The ALL option is particularly risky - it basically clears out your entire Q-Dock array of program icons. There is a Are You Sure Yes Or No confirmation option, but even so, please use this command with extreme care!

Q-Dock Configuration - Settings Menu

The Settings menu, accessed by pressing F3 or clicking on the F3 icon in the Q-Dock home screen, is used to set configurable options for Q-Dock. The information is saved in a file called qdock_dat in your Q-Dock directory. This is a plain text file - you can load it into a text editor to see what information it contains.

The settings menu lets you alter the following Q-Dock settings:

Skin Sprite	: A 512x64 pixel mode 4 or GD2 QL sprite
Transparent background	: Off or on
Printer port	: Ser1, Ser2, Par, or even a filename
Sorted file lists	: Off or on, sorting of filenames in menus
Sound off/on	: Error beeps can be turned off
Bubble help off/on	: Off or on
Delay	: time before Q-Dock autohides when pointer is out of its main window
System palette number	: 0 to 3 - colour scheme
Autohide off/on	: Off or on
Autohide distance	: How close to screen edge pointer wakes up Q-Dock when it's auto-hiding
Dock indicator colours	: Colour number of the dock switch buttons
File handler name	: thing or filename
Commands view	: shaped or text dock switch buttons
Shadow type	: shadow or reflection type selection

Skin sprite:

The skin sprites are used to represent the docks themselves. They may be QL mode 4 or GD2 sprites of 512x64 pixels in size. A selection is supplied to choose from, more are available from my website's Q-Dock page.

It's quite easy to create your own either using a sprite editor able to edit sprites of this size, or using a program such as Marcel Kilgus's PNGconv program for Windows, which converts .png graphics files to QL sprite format. Pointer environment is fairly forgiving about mode numbers, although it's best to stick to traditional mode 4 sprites if you are using Q-Dock on a QDOS system with Window Manager 2.

Alpha-blend transparency in GD2 sprites is supported where the version of Window Manager you are using supports it. Alpha-blend means transparency - you can see some of the background through the dock sprite which makes for a very nice effect if you are using a glass-style dock combined with the transparent background option being selected. Some of the sample dock graphics supplied and some on my website have transparency masks included, which can be quite effective, but it does depend to a degree on the background behind the dock graphic - very transparent blue dock over a blue background might not look very effective, for instance.

To select a sprite, you click on the field to the right of the words 'Skin sprite' in the Settings window, then use the file selection menu which has a sprite viewing option to preview the sprites. Or, of course, you can use a separate sprite viewer such as my Sprv program or Per Witte's Msprv program. If you press ESC without actually clicking on a sprite filename, Q-Dock uses a plain simple solid colour dock, which may be a random colour.

The sprite file selection menu is just like the file selection menus in most of my programs, with the only difference that if you Hit (left click) on the filename of a sprite, it is displayed first so you can see what it looks like before committing to it. Small sprites are shown just to the right of the list of files. A new window opens up to display the larger dock graphics. This window includes a couple of lines of text at the bottom showing the size and display mode of the sprite

to make sure it's suitable. You can press F1 or space or enter to check what it looks like against various background colours, which might be useful if it is a sprite with a high degree of transparency. Press Esc to return to the file selection menu. Once you have decided on a sprites to use, Do on its filename in the list to select it. Or press Esc just to use a plain random solid colour dock.

The file selection menu normally expects sprite graphics files to have filenames ending with `_dck` (for dock sprites) or `_spr` (for programicon sprites). It is possible to change this in the file selection menu - near the top right there is an item called "F3-Ext". Click on this to amend the extension - Hit on it to type in a new extension, or Do to select from a few preset options in a menu, such as `_spr`, `_dck`, or `_ico`. The significance is of course that to keep file lists short, the file selection menu only shows files with the suffix indicated - useful when your Q-Dock directory has a large number of files of various types.

The dock sprites supplied have the filename extension `_dck` (standing for 'dock') to distinguish them from the smaller 32x32 program icon sprites, which have the filename extension `_spr`. This is merely for convenience, as apart from size the two types of graphics are identical in format and this naming convention is used only to differentiate between the two types when they are all mixed up in the same directory.

Dock sprites may have a bitmapped mask or an alpha mask pattern. Of those supplied, alpha-mask transparency dock sprites have filenames which include `_a` in the filename, while bitmapped (non-see through) dock sprites have `_t` in the filename. No file format reason for this, it is just to distinguish between them.

Transparent background:

Q-Dock supports the use of transparent program windows, where the usual paper colour is not drawn when a window is opened. While the mechanism is supported by the operating system, the window save and restore routines of the pointer environment do not handle very well - what you may find happening is that as you CTRL C in and out of programs, the area saved for Q-Dock may include background programs or wallpaper or screen background colour. Where programs don't overlap too much, this is not an issue, but from time to time you might find the Q-Dock display "catches" some of the display behind it, and if either Q-Dock or the background program is moved, bits of each other's displays might get caught up and dragged around. While messy and inconvenient, it's easily cured by clicking on the program's Refresh icon. Pointer driven programs usually have an icon to redraw the display, while older programs normally use keys such as F4 or SHIFT-F5 to redraw their own display. Q-Dock usually tries to stay ahead of this by redrawing its own display where it thinks such display corruption might occur, e.g. when dragging the Q-Dock display around the screen. This is the reason why Q-Dock appears to redraw its own display more often than might otherwise be thought necessary.

The transparent window display is handy if you can put up with the occasional corruption, since Q-Dock can be a different shape to the normal rectangular QL programs.

Printer port:

The default printer port to be used for printing out the list of program settings within Q-Dock. This can also be temporarily amended during printing if required.

Sorted file lists:

Lets you choose whether or not file selection menus present the list of files in alphabetically sorted order. Yes is the usual option, but on slow systems sorting a long list of files can take a long time (for example, when you have built up a large collection of Q-Dock icon sprites!!!).

This is the default setting. You can temporarily change the setting between sorted and unsorted by clicking on the "Sort" item in the file selection menu.

Sound off/on:

Specifies whether the error "grunt" sound is turned off or on. While the sound can be useful in drawing your attention to certain errors, you may find it gets on your nerves or those around you, so it can be turned off if you prefer.

Bubble help off/on:

Bubble help is a system whereby if you hover the on-screen pointer over certain items in the program, it opens up a very small window with a short text either describing what that button or icon does, or shows the name of a dock when you hover over the 9 dock selector buttons, for example, or the name of a program associated with a given icon. Or even, if your choice of sprites don't make it too easy to see if that gap between two icons is just a gap or indeed an empty icon slot you can add another icon to, it will helpfully say "empty icon slot" or similar if it is an unused icon slot. While bubble help can be useful at first while you get used to the program, you may prefer to turn it off later when it is no longer useful.

Delay:

If Q-Dock is set to auto-hide itself, the way it determines whether to go into hiding is to wait for a set amount of time once the pointer is detected as held outside of its window area. Indeed, if Q-Dock is in the way of something else, just move the pointer away from Q-Dock for a few seconds (the default is 3 seconds) if autohide is selected. It works in units of whole seconds.

It's not infallible, things might get too complex to work reliably if lots of programs overlapped and Q-Dock is part buried under another program's windows. You'll quickly get used to when it does and doesn't work.

System palette number:

The Window Manager keeps a selection of (normally) four preset colour themes. Sometimes, these are also known by the term "colour themes". The idea is that programs which understand and use these system palettes can take on a broadly similar look in terms of colours and borders and so on. Indeed, packs of such colour theme files are available from websites such as my own to vary these colour schemes, although you'll need to read up a bit on how to use them. You can design your own - Wolfgang Uhlig wrote a program called Q-CoCo to create colour theme files. Q-CoCo is available to download from websites including my own and that of Bob Spelten jr. Most of us use the default system palette number 0, but if you want a program to look different, or wish to check how Q-Dock might look in a different colour scheme, try system palettes number 1,2 and 3.

Autohide off/on:

Like many dock utilities, Q-Dock has a wonderful feature whereby it can minimise itself to get out of the way when it is not in use (defined as when pointer is outside the Q-Dock window for a few seconds). It doesn't always get it right - it's actually a bit tricky to program under QDOS and SMSQ especially when Q-Dock is partially buried by other program windows - but it can be very useful. If you prefer Q-Dock to work like normal QL programs, just using CTRL C to

switch between programs for instance, set this to off. There are three settings - off, autohide 1 (activated by pointer at bottom of screen) and autohide 2 (activated by pointer at top of screen). To toggle between the three states, just click on this item and it cycles round the three possible settings.

Autohide distance:

When Q-Dock has gone into autohide, you need a way to get it to pop back up when you want to use it. Most dock programs live at the bottom of the screen, so the best way is for it to check the pointer position when hidden and if it spots that the pointer is being held near the edge of the screen, take action to pop itself up for use. The edge used is usually the bottom of the screen, partially because that's where docks normally live, partially because the QPAC2 button frame and most QL programs use the top of the screen, but Q-Dock can live at the top of the screen if you prefer.

We need to specify how close to the edge of the screen the pointer needs to be held for Q-Dock to wake up. We do this by specifying how many pixels from the top or bottom should be considered "wake up territory". Normally about 16 to 20 pixels will suffice with the standard on screen pointer arrow. So enter this value.

Dock indicator colour:

When the little buttons you can click on to select docks are shown, you can set the colour to use for the current dock. This should be an ink colour number from 0 to 7, although it is best to avoid 0 since this will be the same as the other dock indicators (black). This is also the colour value used for the text-style dock numbers.

File handler name:

When you create an icon for looking in a given directory, this is the Thing or program used as the file manager, to view the list of files. It may be a Thing or menu like the QPAC2 Files menu, or a file manager program like my Q-Trans program. Just specify its filename or thing name here - for the QPAC2 files menu you can use the Thing name "Files" (without the quotes). The file handler is started with an EXEP command, which allows for Thing or file names.

Commands view:

This lets you select the appearance of the commands menu at the bottom of the dock. The commands can be turned off if you find they get in the way and are just not required. Some docks look a lot better with these turned off, if you can remember the keystrokes needed for commands (F3), quit (ESC), Move (ctrl-F4), Refresh (Ctrl-F2), and dock switching (1 to 9).

Shadow type:

Icons you have set up in a dock can have shadows or reflections under them. There is a choice of up to 9 shadow strengths, and two reflection types for different surfaces. Reflections take a lot of time to calculate at each redraw, so are best not used on slower systems. To select a shadow, a reflection, or none at all, just select the appropriate one in the menu shown when you click on this item.

Sprites

Sprites used as program icons in a dock are 32x32 pixel QL sprites, which can be in QL mode 4 or on suitable systems may be mode 16 (256 colours, Aurora style), mode 32

(QPC/QXL/SMSQmulator), mode 33 (Q40 or Q60) and possibly - untested yet - 24-bit colour sprites. The system is quite good at adapting sprites for different colour resolution modes, although it is best to stick to traditional mode 4 sprites if running on a machine or emulator with only mode 4. The sprites should be no larger than 3000 bytes each.

Sprites used as dock graphics should be 512x64 pixels in size with the same choice of colour resolutions as above for the 32x32 icon graphics.

Icon and dock graphics can be a bit smaller than the sizes stated, although this is unlikely to be useful.

Dock and icon sprites may have regular bitmask transparency, or alpha-mask transparency (i.e. partly see-through) on systems supporting alpha-mask sprite blending. Dock sprites having alpha-mask transparencies can be identified among those supplied through having filenames containing the characters '64al', whereas dock sprites with bitmask transparency have '64tr' in the filename. It is normal for the area outside the dock shape itself to be transparent, although this is not compulsory.

Docks and icon sprites may be uncompressed or RLE compressed. Uncompressed ones are a bit bigger (may require more memory), while compressed sprites are a little slower to process.

For convenience, dock sprites have names ending with "_dck" to distinguish them from smaller program icon sprites which have names ending with "_spr". Apart from size, obviously, both are otherwise identical in terms of file format. If you create your own 512x64 dock graphics which have filenames ending with _spr, they will work fine like that but you may find it more convenient to rename them so that they have filenames ending with _dck instead of _spr so that you can tell them apart from ordinary program icon sprites.

To download more icon and dock sprites (mostly free public domain ones), go to the Q-Dock page on my website at www.dilwyn.me.uk/qdock/index.html

On that page you can also download my freeware Sprv sprite viewer program, which has a 'tiled' viewing mode so you can view a whole page of sprites at a time.