**HELPLINE**

Dilwyn Jones

Remember that you can send me your Helpline queries by email to helpline@quanta.org.uk , or by letter to the address inside the front cover. While we do our best to help, we obviously can't know everything about everything, so sometimes it takes a while to find someone who might have knowledge of what your query is about. Some queries may be published in the magazine and in the future listed on the Quanta website, although if you ask to keep the query private or anonymous, we will of course respect your wishes.

In the last issue, we looked at transferring files from a QL to emulators via serial cables. In this issue, I would like to discuss how to transfer files to the QLay-based emulators, using both QL floppy disks and transferring files between the QLay environment and the underlying Windows hard disk.

Most of what is printed here applies to all Windows versions of QLay (including QLay 2) and QL2K, which is based on the QLay emulator. None of this applies to other Windows emulators such as QemuLator, uQLx for Windows, MESS, or QPC.

QLay, QLay2 and QL2K in particular are great free QL emulators for Windows, which are slightly hampered by their lack of ability to read QL floppy disks. They can save files into a WIN drive on the PC hard disk and to something resembling a MDV cartridge but stored on the PC hard disk. They do not have floppy disk drivers, so to transfer files in and out of QLay, QLay2 and QL2K we need to use some specially written programs to assist with the task, referred to as QLTools and QlayT programs which are DOS programs, not QDOS ones. These programs greatly enhance what we can achieve with these free emulators, but they are not the easiest of programs to get to grips with and I am often asked about QLay file transfer, so I thought an article on this subject would be helpful.

**The QLay Environment**

QLay, QLay2 and QL2K all store their QL files in a special folder on the Windows hard drive. The emulator uses a directory structure known as "qlay.dir" which, as the name implies, means "a QLay directory". This stores not only the file itself, but crucially a copy of the QL file header which includes such things as executable file headers, dataspace and so on. Without these file headers, programs you normally execute with commands like EX, EXEC, EW, or EXEC_W couldn't work - vital information the QL system needs to be able to start that program would be missing and the attempt to execute the program would be doomed to fail with a "bad parameter" error because the QL simply didn't know how to start the program in view of the missing information.

So what these emulators do is place a copy of these file headers in a special structure called qlay.dir inside the folder QLay uses to store our files. QLay's "hard disk drives" are called WIN drives, with full names of WIN1_ WIN2_ and so on up to WIN8_

So, if you have written a BASIC program on QLay or QL2K, you can save your program to one of these drives with a command such as SAVE WIN1_MYPROGRAM_BAS pretty much like you'd save to FLP1_ or MDV1_ with a QL floppy disk drive or microdrive, for example.

Saving and loading files you created in QLay and QL2K is normally that easy, but the real fun and games comes when you want to copy files from a QL floppy disk to the emulator. You guessed it - QLay and QL2K can't read floppy disks!

So, what the authors of the emulators did was to make and supply some programs they referred to as "tools" to help you transfer files between QL floppy disks and the emulator (a DOS program called "qltoolsq"), and between the emulator's environment and the rest of the PC hard disk (a DOS program called "qlayt"). A copy of these two programs placed in QLay's WIN drive folders will help you transfer files to and from the emulator.

When you first set up QLay or QL2K you will have set up the emulator to have its various WIN files in separate Windows folders on your PC's hard drive. In the case of QLay2 for example, you may have stored the emulator itself in a folder called C:\QLAY\ and inside this folder you may have other folders called WIN1, WIN2, WIN3, WIN4 and so on. Calling them WIN just makes it a little easier to know what they are when you are browsing the Windows hard drive with Windows Explorer, or other Windows programs.

In the case of QL2K, the system is very similar with only a minor difference in that instead of calling the WIN folders WIN1, WIN2 etc., by default they have an underscore at the end of the name. QL2K normally installs in a folder called C:\Jadiam.org\QL2K\ (or may be in the "Program Files" folder on your hard disk). Wherever you store the emulator, inside the emulator's folder should be sub-folders called WIN1 or WIN1_ which hold the files your emulator saves to hard drive.

Using Windows Explorer, have a look inside these folders, but make sure you don't change anything yet. Locate the WIN folders and inside them you should see a file called qlay.dir and possibly a few small programs with names such as FT_BAS or QSBB_BAS. These are the programs the QLay emulator uses to help you set the timing values to make sure the emulator runs at the right speed. These may not be present in QL2K as its timing system is rather different. The main thing to look out for is the file called qlay.dir - if this is present, your emulator has set up its WIN drives.

By looking at the WIN folders like this you gain an idea of how QLay and QL2K store files on hard disk. They basically store the files as Windows files in these folders, but in addition they put a small QL file header in the qlay.dir file, so to properly use WIN drives the emulator must be able to find both the file itself in the folder and an entry inside the qlay.dir file. For example, if you save a BASIC program called myprog_bas it will (a) save it as a Windows file called myprog_bas and (b) create an entry in the qlay.dir. So the qlay.dir is both a list of files the emulator can "see" and the file header too.

There is a small restriction on these qlay.dir directory files. They can only handle about 160 files and if you try adding more than this, some versions of the emulator will fall over and some very strange things start happening which could cause you to encounter problems accessing these files. But since you can have up to 8 WIN drives, this allows you to use up to 8*160 or about 1200 files in total, so it need not be seen as a very big restriction.

Another restriction in QLay and QL2K is that you cannot use "hard" directories from the emulator. There is no MAKE_DIR command to create them. So the fact that you can only store up to about 160 files on each drive is actually a blessing in disguise - more than this and you'd be swamped and find it impossible to organise and find files!

Before we can start copying files in and out of the emulator's WIN drives, we need to make sure we are using the correct "tools" programs. These are called qlayt.exe and qltools.exe. These are supplied with the emulator and the easiest way to use them is to simply copy them into the WIN folders using Windows or DOS. Since these are called from DOS or Windows, you do not need to worry about their details being seen in the emulator's qlay.dir system because they cannot be used from the emulator itself.

First, decide if you are running on a Windows NT type of filing system (Windows 2000, XP, or Vista, for example) or on an earlier FAT-based filing system (Windows 95, or Windows 98). See the emulator's documentation on choosing the right version for your particular emulator and filing system. Later versions of the QL2K emulator are only really intended for more recent versions of Windows and the "NT" filing system programs are the only ones which will work.

So, copy the qltools.exe (the emulator documentation refers to this as qltoolsq since it is a version of an older program of the same name, modified for use with QLay and QL2K) and qlayt.exe programs to the Windows folders holding the WIN drives. I found it easiest to put copies in all the WIN folders, which is a bit wasteful of space but it makes it easier to use them while you are new to the system.

Qltools.exe is a program to handle QL-format floppy disks and transfer files between floppy disk and the emulators.

Qlayt.exe is a program to copy files between the QLay/QL2K filing system and the Windows hard disk drive.

**QLTools**

Assuming you now have your emulator set up ready, with all the WIN drives set up, make sure the emulator is NOT running, then we will try the qltools program to copy files from a QL floppy disk to the emulator.

Start a DOS box in Windows. For example, in Windows XP, go to the Start menu, All Programs, Accessories then Command Prompt. Alternatively, use the Run box to run something called "cmd" which is essentially the same thing. Now we need to change the directory so that we are in the emulator's WIN drive. Type in something like this into the DOS command prompt:

cd qlay\win1\

"cd" is a DOS command which stands for Change Directory. We use it to move to the folder where we placed the qltools.exe and qlayt.exe programs. You should now be able to use a DIR command from DOS to see the files present. The actual path name you type after the cd command will depend on where you have placed the emulator on your computer's hard drive.

Now we need to start the qltools program. Place a QL floppy disk with a few files on it in the PC's floppy disk drive. Like many DOS programs, the program is started by typing its filename and a few characters after it to indicate drive name and what to do.

A PC floppy disk drive is normally drive a: or drive b: so we can just type a space after qltools then the drive name and the colon symbol which the PC needs. First, we'll try getting a list of files from the QL floppy disk in drive A: with this command, typed into the DOS box's command line:

qltools a: -d

It is not usually case-sensitive. You can either type in qltools or qltools.exe, the PC should recognise either form. The a: is obviously the drive letter of the PC floppy drive, then the -d tells it that we want a 'directory' or a list of files. All being well, you should now see a list of files from the QL disk in the DOS box. The '-' symbol is called a "switch" followed by a letter which tells the program which action to take. -d lists files, lengths etc (standard long form)

while another version, using -s lists the short version of the directory, just the filenames, which may be more convenient when you just want a list of files:

Qltools a: -s

Note that although the command itself is not case sensitive, some of the switch commands may be case sensitive. Using upper or lower case letters may produce different results.

The next thing we'll do is to try to copy the files from the QL disk direct to the emulator's WIN drive. To do this, we'll enter a slightly different form of the command:

qltools a: -q

The program will now start to copy files from the QL disk to the emulator's WIN drive and update the qlay.dir system with the details of the new files, so that you can go into the emulator and do a DIR of the relevant WIN drive, e.g. DIR WIN1_ and you should be able to see and load these files you've just copied. And that's all there is to it - just remember to close down the DOS box before you start the emulator. It's that simple!

Qltools has its own manuals, but if ever you want a quick reminder of the commands available, just start the qltools.exe file with no parameters, i.e. in DOS just type the command 'qltools' and it'll show a list of the available commands on the screen. The -q option is the only one specific to QLay and QL2K. All the other commands are general Windows and QL file transfer ones.

For example, you can use qltools to copy a file from the Windows hard disk to a QL floppy using the -w or -W options (the upper case version does not query whether to overwrite files on the disk or not). For example, you could copy a text file README.TXT from C:\ on the hard drive to a QL floppy disk using this command:

qltools a: -w c:\memo.txt

You may find that executable QL programs copied from the PC's hard disk to a QL floppy disk fail to execute properly, often giving a "bad parameter" error. As mentioned above, this is because they have lost their executable file headers and dataspace information. QLTools can fix this, using an option to set the dataspace to a known value. If you don't know the correct value to use, a bit of trial and error may be needed. Giving a program too much dataspace is a waste of memory, while too little may mean the program won't run properly. The -x filename dataspace_size can allow QLTools to fix a damaged file of this nature from what seems to be a data file back into an executable program:

Qltools a: -x  filename 10240

The "filename" is the name of a file on the floppy disk. The number at the end is the number of bytes of dataspace (working memory) it is to be allowed when it starts. If you don't know the correct value, start with something like 10240 (10 kilobytes) and follow a bit of trial and error from there. This technique is very useful for fixing programs you've downloaded from the web and unzipped in Windows, which of course "breaks" QL executable programs. You should not need to do this with non-executables like BASIC programs, text files, databases, spreadsheets and other data files.

Remember that with QLTools, the "-q" option is the only one specific to QLay, QLay2, or QL2K. In all other respects, QLTools can be used as a general purpose Windows/QL disk file transfer program.

**QLayT**

We'll move on to Qlayt.exe now. This program is specific to QLay, QLay2 and QL2K and deals with copying files between the PC hard disk and the emulator environment. The emulator should not be running when you use this program, or confusion may occur.

QlayT helps you copy, delete and update files between the Windows hard disk and the QLay/QL2K part of the hard disk, basically anything where a file needs to have its entry created in qlay.dir or its entry updated. The options available are many and varied (some rather complex) but you'll find that you tend to use just a small number of the available options at first.

You can copy a file (or files) to the emulator's WIN folder using Windows Explorer or the DOS command line. What these won't do is update qlay.dir so that the emulator knows what's happening. This is where QlayT comes in. QlayT has a manual called qlay.man which can be loaded into Windows text file editors such as Notepad to read.

It can work in two ways. You can type in commands to work on one file at a time as we did with QLTools, or you if you wish to copy several files as one batch, you can create a list of filenames in a text file and QlayT can work from that list, saving you a lot of repeated typing.

Like QLTools, we enter the name qlayt in a filename with a list of options after the filename. If the emulator has no qlay.dir as yet, we can create one by using a -c switch in the command line after the word qlayt. If it already exists, we can use the -a switch instead, which appends the file's information to an existing qlay.dir rather than creating a new qlay.dir

For this example, we are going to copy a text file called memo.txt from the PC to use in the emulator. Using Windows Explorer, we have copied memo.txt into the WIN folder for the emulator, but the emulator doesn't yet know it's there until we introduce the information about the new file to the qlay.dir system. We'll copy a single file and update the information with this DOS command:

Qlayt -i memo.txt

Once it's done that, you can check that it was successful by using the -l (and that's a lower case letter l, not the number one) option to list files that the emulator knows about:

qlayt -l

This will list the names of files stored in the qlay.dir. It may report that there are more files than listed. For example, if QlayT and QLTools are in the same folder, these won't be included in the qlay.dir list.

If the file being transferred is supposed to be an executable program, you can fix the file header to make it executable by using -d and a number for the size of the dataspace as above. Alternatively, if you are not sure what the dataspace value should be, you might like to try the -X option which looks inside the program for something known as an "XTcc field" which holds the correct value. Not all programs have this "XTcc field" but it can be worth trying. It is used with a -i command like the -d variant mentioned above. -X finds the details and updates the entry in qlay.dir. Alternatively, if you just want to see the dataspace value for reference, use a lower case -x instead.

Obviously, if you are going to add a few files, it may be quicker to work with a list of files than repeatedly entering separate commands for each file to be added to qlay.dir. We can

create a suitable list in a text editor, or we can send the output of a DOS DIR command to a listing file, using the /b switch to list only filenames (no file lengths etc) and a > output qualifier to send the list to a file instead of to the screen as the DIR command in DOS normally does:

DIR /b >filelist.txt

This sends the list of filenames from the current Windows directory to a new DOS text file called filelist.txt, which we can use in a QlayT command using -c to create a new qlay.dir or use -a to append to an existing one (-a is the usual option once a qlay.dir has been created).

Entering this command into DOS will now make sure that qlay.dir "learns" of these new files introduced to its directory:

qlayt -a filelist.txt

What happens if qlay.dir already knows of a file like memo.txt, but you have played around with it in Windows and changed its size. QLay or QL2K may get a little confused at this point and may not be able to load it correctly, because it's a different size to the one they know about, so at best some data may go missing, at worst the file may be lost. Worry not, QlayT can "update" a file's details in qlay.dir to fix this. Make sure the emulator isn't running then run this dos command from the emulator's WIN directory to update the details:

Qlayt -u memo.txt

Suppose you want to remove the entry for a file from qlay.dir without deleting the file itself, e.g. you have finished with it in the emulator, now it needs to be processed in Windows. The -r switch in a qlayt command removes the entry from qlay.dir without deleting the file itself. The file stays in the WIN folder on the PC hard disk but the emulator no longer knows about it, so it is invisible to the emulator, but visible to Windows:

Qlayt -r memo.txt

Now the emulator has forgotten about memo.txt but you can still load it into a Windows program if you wish.

There are many options available with the QlayT program and this article lists the basics only, just enough to get you going until you have mastered the use of the program. The QLAY.MAN file is essential reading to find out all the available options.

**FLP? What's That?**

Here we have a bit of a catch-22 situation. If you transfer files from QL floppy disk to QLay or QL2K, the programs will most likely be designed to run from a disk drive such as FLP1_. But the emulators don't have any FLP drives and won't know what FLP1_ and FLP2_ and so on are. But the authors have thought of this and provided a WIN_USE command which lets you change the WIN device name to something like FLP so that the WIN drives can pretend to be floppy disks when required. WIN_USE "flp" changes the names of WIN1_, WIN2_ etc so that they now seem to be FLP1_, FLP2_ and so on, thus fooling the software.

**MDV**

QLay, QLay2 and QL2K also have drives called MDV. These are parts of the PC hard disk made to look like MDV files. So you can copy your old microdrive programs (well, unprotected ones anyway) to disk on your QL and then use QlayT to make them into little "MDV" drives on the PC hard disk. Copy the files onto the PC's hard disk and use the -C (note: capital C) option to copy the files to a qlay.mdv slot instead. The actual name of the mdv slot defaults to qlay.mdv but this can be changed with the -o option to another name.

**Conclusion**

The tools programs for these emulators are hard going at first because they work in a way which is a bit alien to us as QL users. Since these emulators can't handle QL floppy disks, we need to master these programs to allow us to copy files in and out of the emulator, so I hope this article encourages readers to explore the use of these programs to make it easier to use the free emulators brought to us by people like Jan Venema and Jimmy Montesinos.