

## JOB2BAS

by Dilwyn Jones

With the increasing use of QL emulators prompting a need to transfer ever more files between a QL and an emulator, one solution has been to make use of the ZIP compression utility to protect the QL executable job header as a QL program gets transferred through a non-QL environment.

If you copy a QL program such as Quill into a Windows folder, for example, Windows does not know how to handle the QL executable's header, strips it off, and to all intents and purposes converts what was a program you could EXEC on the QL into a simple data file. Try to execute that in QDOS or SMSQ/E and you might well get an error message such as "bad parameter".

Using QL ZIP to hide away the header does work, but of course you need to know how to use QL ZIP and UNZIP as well as know how to transfer the file through another operating system so that the QL emulator receives the file intact. Plus there is another little snag - if the QL emulator has no Unzip program, you will need a copy of Unzip to Unzip the Zipped copy of Unzip - catch 22.

So I thought I'd attack this problem from another angle and try to see if I could think of a way of allowing QL executable programs to pass through another operating system without losing the file header. This method had to ensure that apart from Toolkit 2 (which most emulators either have built in or can load a simple Toolkit 2 ROM image).

So, the rather radical solution I came up with was to convert the QL program to a SuperBASIC program. Yes, that's right, a BASIC program.

SuperBASIC and SBASIC programs can pass through other operating systems as simple data files as long as you don't change the end of line character by doing something like loading the QL BASIC program into a wordprocessor or something like that, which might add carriage return characters to the end of the lines of BASIC and so prevent it from running.

What JOB2BAS does is to store the executable program as a (long) set of DATA statements in a BASIC program, and adds a few lines of BASIC to those DATA statements to allow the original program to be reconstituted just by running the resultant BASIC program on the target computer.

So the process is:

1. Run the JOB2BAS\_bas program
2. Tell it which executable program to encode as a BASIC program
3. The output BASIC program is written to a file

4. Transfer this new BASIC program (we might call it OUTPUT\_BAS for example) to the target operating system
5. Copy the BASIC program into the target QL emulator
6. Run it on there and tell it what filename to use to save the reconstituted executable program. It remembers the original filename, but you can choose a new filename if you wish.
7. Having saved the executable program file, you can then move any other files needed over as well and proceed to test the software on the emulator.

It is a difficult concept to visualise, but having used the process once or twice (e.g. to transfer a copy of QL UNZIP to the emulator!), you should find it's an easy enough program to use.

There are three options you can change at the start of the program:

1. First line number of the created BASIC program
2. Line increment step from one line to the next
3. The number of items on each line of DATA values (higher values allow longer program to be encoded, but make the output program more difficult to read if you wish to study it)