

Q. I tend to have a large number of toolkits installed on my QL which take up a lot of memory. Is it possible in some way to extract just the extensions I use regularly from them? Also I would like to merge some of them into one file so that they load faster if possible.

A. A program called DEA, a disassembler program from Ergon Development, can in some cases extract keywords from toolkits. DEA used to be a commercial program but is now available as freeware from either of these websites:

<http://www.sinclairql.it/qlpage.htm>

or

<http://www.dilwyn.me.uk/asm/index.html>

The program comes with instructions in Italian and English.

Here's an example where I am extracting the MOVE\_MEMORY command from Turbo Toolkit.

First, start the DEA program itself. This brings up the menu shown in Figure 1



Figure 1

Move the highlighter over the "On File" command as shown in Figure 1 and press SPACE or ENTER. This will take you to the screen shown in Figure 2.

```

DEAssembler V5.303 Plus 3 - HIPOWER Disassembler
  © 1990 Ergon Development - Program by M. Ternelli

DDDS version 3.13
PTR GEN version 2.04
MMAH version 2.05

Filename? (devn_? for directory)  win1
Loading win1_dea_Turbo_TK_CODE ...
Loaded at 3876400
File length: 8170 bytes
ORIGin logical start address?  $10000

FILE win1_dea_Turbo_TK_CODE
Input code  Phys:3876400 Dec: 0% Bk:0 Lab:0 Mem: 79742 TK:0

```

Figure 2

It now asks for the "Origin logical start address". Press ENTER to accept the suggested default value, which will be correct. Press ENTER again to accept the default for the next entry, "Program execution start from code start offset".

This now brings up the Main Menu. Move the highlighter bar down to "Enter Keywords Manager" and press SPACE or ENTER.

```

DEAssembler V5.303 Plus 3 - HIPOWER Disassembler
  © 1990 Ergon Development - Program by M. Ternelli

DDDS version 3.13
PTR GEN version 2.04
MMAH version 2.05

Filename? (devn_? for directory)  win1
Loading win1_dea_Turbo_TK_CODE ...
Loaded at 3876400
File length: 8170 bytes
ORIGin logical start address?  $10000
Program execution start from code start offset?  $00000

Main menu
0  Create DEAssembler code
1  Multi-Level Parsing
2  Disassembly/Dump
3  Alter "examined label" status
4  User parsing options
5  Enter Keywords Manager
6  Empty internal work tables
7  Alter parsing settings
8  Working Storage Files
9  Sleep (shift caps to wake)

FILE win1_dea_Turbo_TK Len: 8170 Log: $10000 Last: $11FE9 Off:0 IP:16
Interactive Phys:3876400 Dec: 0% Bk:0 Lab:0 Mem: 79742 TK:0

```

Figure 3

Now ensure that the highlighter is on "Edit Keywords" (see Figure 4) and again press SPACE or ENTER.

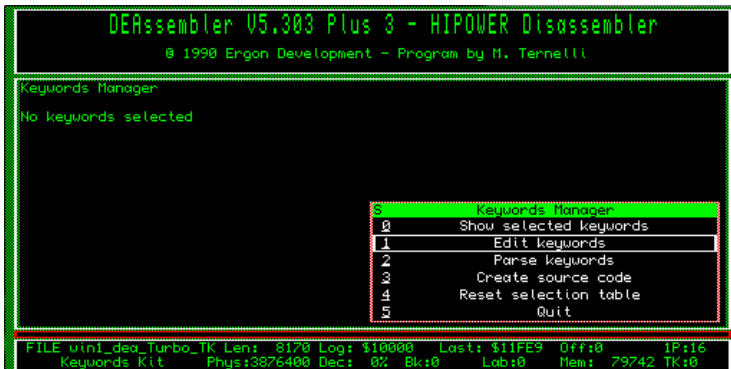


Figure 4

Now you need to enter a series of responses to questions the program asks you.

First, enter the name of the keyword you wish to extract. For the purposes of our example, this will be the MOVE\_MEMORY command, so enter "MOVE\_MEMORY" (without the quote marks). DEA looks for the keyword, then asks if you would like to "Keep track of MOVE\_MEMORY?". Press ENTER or Y for Yes.

Next, it asks if this is a procedure. In this case, MOVE\_MEMORY is a procedure, so press ENTER or Y for Yes (if it had been a function you would have replied with N for No).

Next, it asks for the next SB (SuperBASIC or SBASIC) keyword name, but we don't wish to do any more at this stage, so press ENTER to return to the keyword manager menu.

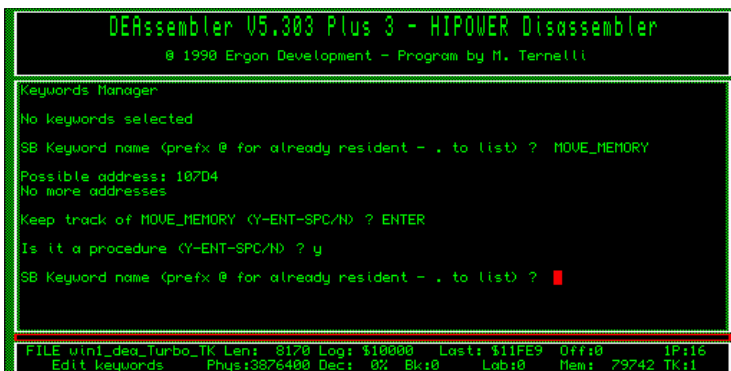


Figure 5

Now it asks "Parse keywords?". press ENTER and the menu in Figure 6 appears.

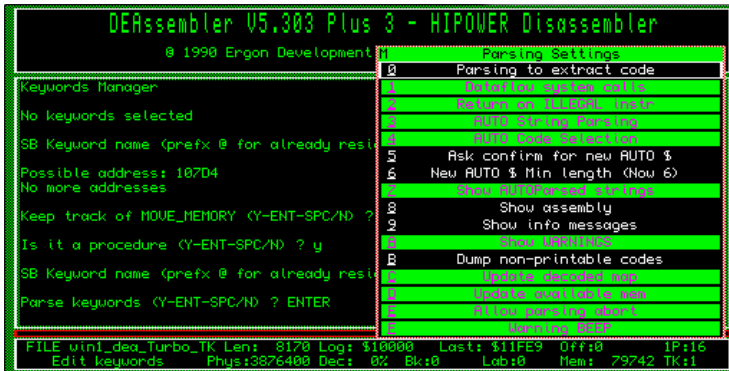


Figure 6

Press ENTER to accept the suggested defaults. When you see the message "Keywords Parsing Complete" in the DEA main window the Keywords Manager menu will reappear. See Figure 7

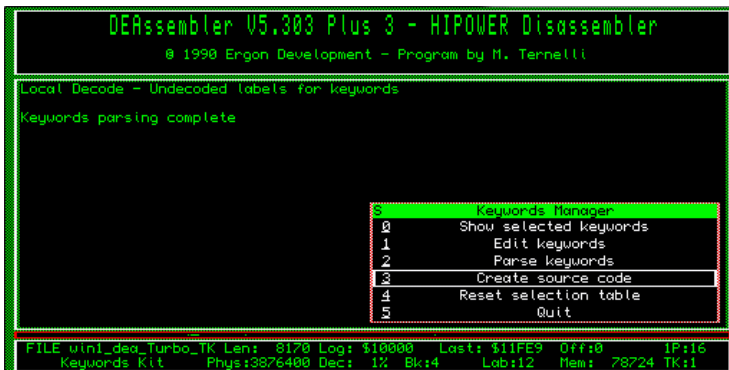


Figure 7

Choose item 3 in this menu, "Create source code" and press ENTER, which then brings up the "Output Stream Choice" menu, as shown in Figure 8.



Figure 8

This lets you specify where to send the output file to - this can be to screen, to a device file such as SER1, or (the option we want) which is to a file. Select the default of "1 On storage file (e.g. FLP1\_"

Next, it offers a default filename for the assembler file - which in our case may be something like FLP1\_Turbo\_TK\_asm or WIN1\_DEA\_Turbo\_TK\_asm. If you wish, edit this filename to something more suitable such as FLP1\_MOVE\_MEMORY\_asm or WIN1\_DEA\_MOVE\_MEMORY\_asm

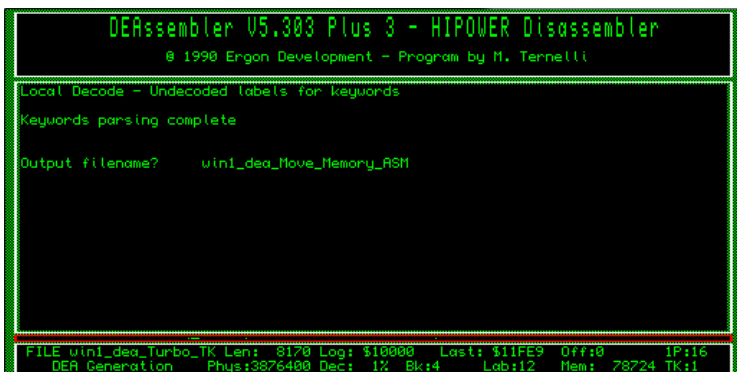


Figure 9

After this it shows the DEA Code Generation Menu - see Figure 10.

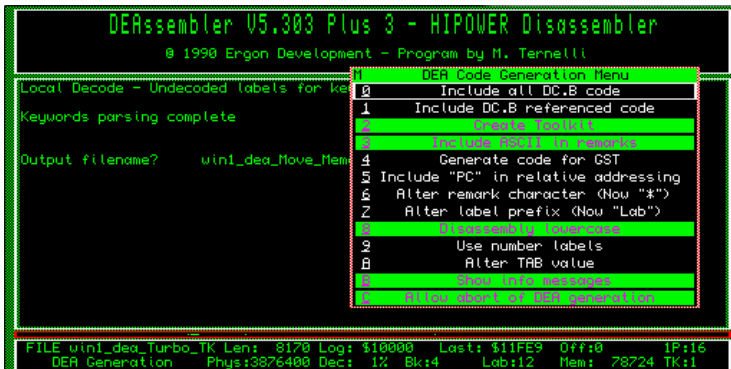


Figure 10

Assuming these defaults are OK (they will be fine for our example purposes) press ENTER. The DEA Code Remarking Menu now appears - see Figure 11

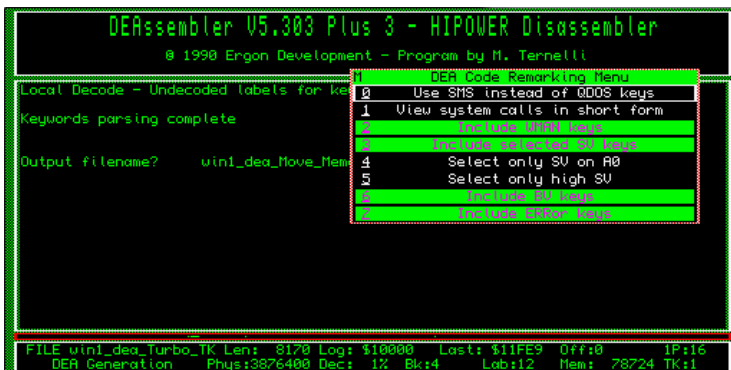


Figure 11

Press ENTER again if all these defaults are to your liking. The program now goes ahead and generates the disassembled output code with the filename you entered above.

Now you can load the assembler code generated into an editor to tidy up by hand, add comments, add more code, etc. then assemble it with your favourite assembler. Note that if you are using the GST QMAC assembler, you need to study the details in the DEA manual for how to set DEA to generate code suitable for use with this particular assembler.

DEA is not a miracle worker - it cannot be expected to handle each and every extension in each and every toolkit of course, but it will be fine for most.

**Copyright**

Note that there may be copyright implications in extracting keywords from some toolkits like this, plus of course you need to have a little knowledge of assembler on the QL to be able to use an Assembler program to reassemble the code you have extracted. Read any copyright notices or licences supplied with the original toolkit to see if you are OK to do this—generally it will be fine if the code generated is for your own use only, but it may be a different matter if you include the new code in a program which you intend to sell, for example.

## Combining Extensions Into One File

There are a few free programs available to combine extensions into one single file and I will list two such examples here.

The first is a utility program from Richard Kettlewell called P-Link. This is available from the usual PD QL software sources. Execute the program called PLINK and it will ask you to enter the filenames of the extensions files to be merged, until you make a blank entry and then it will ask you for a filename to save the combined set of extensions. It's that simple!

You can also call it with an input parameter filename of a list of files to be merged, followed by a blank line, followed by the output filename. As an example, we will create the following file, which we will call FLP1\_EXAMPLE\_TXT:

```
flp1_ptr_gen  
flp1_wman  
flp1_hot_rext  
  
flp1_mergedkit_cde
```

You would call this with something like:  
EX PLINK;'< FLP1\_EXAMBPLE\_TXT'

So what this does is to load and merge ptr\_gen, wman and hot\_rext and save all three in a single combined file called "mergedkit\_cde". This is an example only - you would not normally wish to merge these particular files.

If you wish to look at how it works, the package includes the source code, written in C.

A second program to do this is Emmanuel Verbeeck's Super Kit Merger, also available from the usual QL software sources. Figure 12 shows what the program looks like when it starts. Figure 13 shows what it expects you to enter.



Figure 12



Figure 13

In the top window, you'd enter the filenames of the extension files to be merged, pressing ENTER on a blank line to finish the list.

Next, in the second box down, you enter the filename of the merged file you wish to create.

In the third box down you can enter a filename for the program to create a short BASIC loader program, a kind of BOOT file if you wish.

Like P-Link, this program also comes with the source code, this time written in SuperBASIC, if you'd like to study how it works.

When using programs like these it goes without saying that you should keep backup copies of the original files in case you wish to go through the process



again at some date in the future!

P-Link and Super Kit Merger can be downloaded from the Assembler page on my website, at:

<http://www.dilwyn.me.uk/asm/index.html>