

QUANTA HELPLINE 07 - Vol 26 Issue 2, Apr/May 09

Q. Is there any way of changing the case of a name in SuperBasic? I inadvertently entered a variable name in capitals and I want it to be in lower case.

A. SuperBasic (and Sbasic) use the case of the first instance of a name to determine the case of other (later) references to that name. The only way to change the case is to write a line which specifies the case, then merge in your original program, e.g. if your program defined the variable NUMBER but you wanted NUMBER to be in lower case, what you could do is NEW, then write a short one line program:

```
1 LET number=0
```

then merge the original program and you will find that all references to NUMBER have now become number. This not only works with variable names, you can use it to update procedure and function names too.

Q. My PC's mouse has a scroll wheel. Will it work with QPC?

A. Yes. Using the scroll wheel will scroll some menus in QL pointer driven programs, or scroll the editing screen in Toolkit 2's ED editor (very useful for scrolling through long basic listings!) along the lines of the "ALT-cursor up/down" keypresses. Clicking on the mouse wheel will pick the button frame, similar to an "ALT ." keypress, or like the third button on some mouse systems. This is only implemented in version 3 and later of QPC2, so it may not work on older versions. Also, it may not work on Windows 95 systems, as that version of Windoze implements the mouse wheel handler differently.

Q. Can I use a mouse on QemuLator? I tried, but couldn't get it to work.

A. Assuming we're talking about the Windows version of QemuLator, there is a small file called "macmouse" which needs to be loaded as well as pointer environment.

Q. I like to be able to put pointer driven programs into the QPac2 button frame, but not all programs have a Sleep icon. Is there any way of putting those programs in the button frame? My screen gets cluttered with so many programs open at the same time.

A. QPac2 has a useful little thing called Button_Sleep whose job is to do just this. The easiest way to call it is from QPac2's Exec menu, or you can set up a hotkey definition to just 'Zap' any program into the button frame. In this example, I use CTL-ALT-F1 (hold down the CTRL and ALT keys and tap the F1 key):

```
ERT HOT_WAKE (CHR$(233),'button_sleep')
```

CHR\$(233) is the key CODE for CTRL ALT F1.

Q. Is there a preferred order of which part of a boot program does what?

A. Although QL BASIC makes it quite easy to write a boot program, I suggest the following

order for the reasons explained below:

Lightning or Speedscreen if used

TK2_EXT or load Toolkit 2 is disk version

Set DATA_USE and PROG_USE if used when loading extensions

Install BASIC extensions (using LRESPR or RESPRLBYTES/CALL)

Hotkey or altkey definitions

HOT_GO command to activate hotkey job

Reset DATA_USE, PROG_USE etc as required

Execute any programs needed at startup

Chain load or merge a second basic program (e.g. a program which uses the extensions loaded by the boot program.

Q. My PC keyboard has extra function keys (F6 to F12) and various other extra keys like Home, Page Up, Page Down and End. Can I use these from Basic on my emulator?

A. Some of these extra keys are mapped onto equivalent QL keyboard keypresses, but it may vary a little from system to system. You can test if a PC keyboard key generates a valid QL key code with this little routine:

```
REPEAT loop:PRINT CODE(INKEY$(-1))
```

Here are some example results I get on my laptop, when using QPC2. The equivalent QL keyboard key presses are shown in brackets.

```
Home = 213 (SHIFT ALT UP)
Page Up = 212 (SHIFT UP)
Page Down = 220 (SHIFT DOWN)
End = 221 (SHIFT ALT DOWN)
F6 = 234 (SHIFT F1)
F7 = 238 (SHIFT F2)
F8 = 242 (SHIFT F3)
F9 = 246 (SHIFT F4)
F10 = 250 (SHIFT F5)
DEL = 202 (SHIFT RIGHT)
```

Q. Some time ago I created a program with an older version of Easyptr. The menus are still OK, but they are linked to an old version of the extensions called ptrmen_cde. I am now using a newer version of Easyptr. Do I have to redesign the menus, or is it possible to substitute the new extensions file somehow.

A. This can be done using the Appendix Manager program supplied with Easypttr. First of all, load the new base extensions file ptrmen_cde. This should not have any menu definitions appended to it at this stage. Next, load the original extensions file which has the menus appended to it. Appendix manager now takes the menu definitions from the original extensions file and appends them to the newer version you loaded first (in essence it strips the old extensions file out and adds or appends what's left to the new base extension file. Now just save the new version.

Q. When I try to access files on my PC with DIR DOS1_ from QPC2, some of the files are not shown in the list. Why is this?

A. I can think of two possible reasons. The first is that the file is simply flagged as "hidden" in Windows. More likely, it is possible that the filenames are too long for SMSQ/E which has the same limit on filename lengths as QDOS, 36 characters. This includes the directory name. So if the PC files are in a long directory path like

C:\Very long directory name\

This leaves only about 12 characters for a visible filename. There are two possible workarounds:

1. Move or copy the files to a shorter directory name, e.g. C:\JR\ and make sure the filenames are not too long for SMSQ/E.
2. Define a DOS drive to include all or part of the path name. This partially adds to the listable files, e.g. try seeing what's in C:\program files\internet explorer\ by using DIR "DOS1_Program Files_Internet Explorer_" and you won't have much success. Then, try DOS_DRIVE 8,"C:\Program Files\Internet Explorer\":DIR DOS8_ and you should see more files.

Q. I have bought a TFT flat screen monitor and can't get it to work with my Aurora system (part of the picture is missing). It works fine with my PC.

A. TFT screens tend to like one particular screen resolution and aspect ratio. For example, if it is a 1024x768 pixel 15 inch screen it will usually prefer resolution ratios of 4:3. In this case, you can try 512x384, or 640x480 or 768x576 on the Aurora. Some resolutions offered by the Aurora may be "scaled" by the flat screen monitor and look awful especially with text as it tries to make the video fit by repeating some pixels and not others. Experiment a little with DISP_SIZE commands on the Aurora until you find the best setting for your particular monitor.

Q. If I buy SMSQ/E for my computer, will it run my old SuperBasic programs? I have heard the Sbasic won't run all programs.

A. Yes, it will run almost all your old SuperBasic programs. You may find that the odd one does not work, but they are usually ones which use POKE commands to access the computer's data structures, such as POKEing directly into system variables, basic areas and the screen. The opposite is not always true – Sbasic has more commands and functions than SuperBasic. If a program written on an SMSQ/E system uses these new keywords, it may not

run on a SuperBasic machine. The SAVE and LOAD commands work in the same way on both types of system – Basic programs are saved as untokenised plain text files.

Q. I have some programs saved using Liberation Software's QSAVE utility. Does this work on an Sbasic system – can I run these programs?

A. You don't need the QLOAD/QSAVE software on an Sbasic system as compatible commands are built into Sbasic. There are also QMERGE and QMRUN commands. QLOAD and QLRUN will happily run programs saved using Liberation Software's QSAVE command.

Q. I have an old game which runs far too fast to be playable on my system.

A. If you are using SMSQ/E, try using the SLUG command to slow down the system. SLUG 1 will make it run at normal speed, while SLUG 5 should make it very slow. If you are using a fast QDOS based system which doesn't have a SLUG command, try Norman Dunbar's free Slowgold program to slow down the system. You can download Slowgold from:
www.dilwyn.uk6.net/utills/index.html

Q. I have a program which only runs from floppy disk. How can I make it work from hard disk?

A. If your system has the DEV device driver (most hard disk systems do) you can put the program into a directory on your hard disk and set the DEV device to point to this directory, then alter the DEV device name to masquerade as the floppy disk drive. For example, copy Quill into a directory called win1_pSION_ and then use the following DEV commands:

```
DEV_USE 1,win1_pSION_  
DEV_USE 2,win1_pSION_  
DEV_USE 'flp'  
FLP_USE 'fdk'
```

The first 2 commands set DEV1_ and DEV2_ to be equal to win1_pSION_ so that references to both flp1_ and flp2_ by the software can be handled. The third line DEV_USE "flp" changes the names of dev1_ and dev2_ to flp1_ and flp2_, so that when a program does DIR FLP1_ for example, it is really doing DIR win1_pSION_. Finally, FLP_USE "fdk" renames the floppy disk names to something else (in this case fdk1_ and fdk2_) so that they can still be used even though the name is now being used to access a hard disk directory.

Q. My QL has a Trump Card expansion. I have been offered a Gold Card. What is the difference and also what is the difference between Gold and Super Gold cards?

A. All were made by the same company, Miracle Systems Ltd. Your Trump Card gave your QL a total of 896K of RAM. A Gold Card gives your QL 2 MB of RAM and a faster 68000 processor, disk interface and battery backed clock. A Gold Card lets your programs run much faster than the QL and Trump Card. A Super Gold Card gives your QL 4MB RAM and an even faster 68020 processor, so programs ran even faster, a parallel printer interface, plus a

disk interface allowing connection of 2 pairs of floppy drives (4 drives total). Trump Card and Gold Card could handle up to 4 disk drives, but needed an external adaptor card at extra cost to run drives 3 and 4.

Q. My program needs to POKE values into system variables. I understand that the system variables may not be in the location in memory on SMSQ/E systems. Is there any way I can ensure that the POKEs work wherever the system variables lie in memory?

A. The SMSQ/E manual states that “the use of PEEK and POKE is best regarded as a form of terrorism.” Even so, it is relatively simple to locate the base of the system variables in memory using a special form of the Basic function VER\$. PRINT VER\$(-2) will tell you the base address of the system variables and you can simply add the offset of the value you want to change (e.g. LET addr=VER\$(-2):POKE addr+n,value), although it is not recommended practice to go fiddling around in this area.

There is also a special form of the POKE, POKE_W and POKE_L (and PEEK, PEEK_W and PEEK_L) commands which is designed to poke directly into system variable offsets, which reduces the risk of the system variables happening to change between the LET and POKE statements shown above:

```
POKE !offset,value
```

This version automatically adds the address of the base of the system variables to the offset value given, and places the value given into the resultant address.

VER\$(-2) and POKE !offset should work on both Sbasic and Minerva systems. If you need to make a program take advantage of facilities on Minerva or SMSQ/E systems, you can use the standard VER\$ to make a decision as to whether to use the extended facilities:

```
system_variables_base = 163840 : REMark fixed location on old Sinclair ROMs
version$ = VER$
IF version$ = 'JSL1' or version$ = 'HBA' THEN system_variables_base = VER$(-2)
REMark Minerva=JSL1, Sbasic = 'HBA'
```

I had intended to hold a Helpline Question And Answer forum on the Sunday morning of QL Is 25, and include questions raised in future Helpline columns.

However, with the benefit of hindsight, Sunday morning was the wrong time and between the small number of people there first thing and the fact that most questions to be asked had already been asked on the Saturday, it got cancelled and Steve Poole gave the (in my opinion) far more interesting talk on 3D Graphics instead.

I did make a note of questions I got asked over the weekend and I hope to include many of these in future Helpline columns once I have had a chance to write up comprehensive replies (the deadline for this issue is too close after QL Is 25).

It was actually quite amazing how many of the questions revolved around the same issues – emulators and sub-notebooks. Questions such as:

How do I get Toolkit 2 on an emulator?

How do I transfer QL software to an emulator?

How do the tools programs in QLAY and QL2K work?

Which sub-notebook is best for me?

Should I use a Linux or Windows emulator?

It did become quite clear to me that emulators are slowly but surely taking over as John Mason suggested in his Chairman's Notes recently. I've been quite accustomed to seeing PCs with QPC at workshops in recent years, and fewer and fewer traditional QL systems, including Auroras (Aurorae?) and Qx0 machines.