NOTE: This Technical Note has been underlined retired. Please see the Technical Notes page for current documentation.

Technical Note ME03
MaxApplZone and MoveHHi from Assembly Language

CONTENTS

Introduction

How to Do It

References

Downloadables

When calling `MaxApplZone` and `MoveHHi` from assembly language, be sure to get the correct code.

[Jan 01 1987]

Introduction

`MaxApplZone` and `MoveHHi` were marked [Not in ROM] in *Inside Macintosh, Volumes I-III* . They are ROM calls in the 128K ROM. Since they are not in the 64K ROM, if you want your program to work on 64K ROM routines it is necessary to call the routines by a `JSR` to a glue (library) routine instead of using the actual trap macro. The glue calls the ROM routines if they are available, or executes its copy of them (linked into your program) if not.

Back to top

How to do it:

Whenever you need to use these calls, just call the library routine. It will check `ROM85` to determine which ROMs are running, and do the appropriate thing.

For MDS, include the `Memory.Rel` library in your link file and use:

```
    XREF  MoveHHi     ; we need to use this 'ROM' routine
    ...
    JSR  MoveHHi      ; jump to the glue routine that will check ROM85 for us
```

For MPW link with `Interface.o` and use:

```
    IMPORT  MoveHHi   ; we need to use this
    ...
    JSR  MoveHHi      ; jump to the glue routine that will check ROM85 for us
```

Avoid calling `_MaxApplZone` or `_MoveHHi` directly if you want your software to work on the 64K ROMs, since that will assemble to an actual trap, not a `JSR` to the library.

If your program is going to be run **only** on machines with the 128K ROM or newer, you can call the traps directly. Be sure to check for the 64K ROMs, and report an error to the user. You can check for old ROMs using the `SysEnvirons` trap as described in Technical Note #129.

Back to top

References

Using Assembly Language

The Memory Manager

Technical Note M.FL.SysEnvirons -- Gestalt and SysEnvirons: a Never Ending Story

Back to top

Downloadables

 Acrobat version of this Note (28K)   Download

Back to top