

New Technical Notes

Macintosh

®

Developer Support

OS 525 - System Error Handler Q&As OS Utilities

Revised by: Developer Support Center
Written by: Developer Support Center

September 1993
December 1992

This Technical Note contains a collection of Q&As relating to a specific topic—questions you've sent the Developer Support Center (DSC) along with answers from the DSC engineers. While DSC engineers have checked the Q&A content for accuracy, the Q&A Technical Notes don't have the editing and organization of other Technical Notes. The Q&A function is to get new technical information and updates to you quickly, saving the polish for when the information migrates into reference manuals.

Q&As are now included with Technical Notes to make access to technical updates easier for you. If you have comments or suggestions about Q&A content or distribution, please let us know by sending an AppleLink to DEVFEEDBACK. Apple Partners may send technical questions about Q&A content to DEVSUPPORT for resolution.

New Q&As this month:
Disabling Macintosh force quit option

Disabling Macintosh force quit option

Date written: 9/15/92

Last reviewed: 6/14/93

Is there a way to disable the force quit option from System 7?

The force quit is handled by calling SysErr with an error code (or selector) of 20002 (0x4E22). The error code is passed in register D0. A simple patch to this trap, and a check for that selector will allow you to abort its function:

```
***** DontForceQuit *****
        INCLUDE      'traps.a'
        INCLUDE      'SysEqu.a'
        INCLUDE      'SysErr.a'

kSysError EQU $A9C9

PatchIt   PROC                EXPORT      ;Apps invoke this by calling PATCHIT() in C;
        IMPORT      MySysError      ;Trap patch code
        IMPORT      OldSysError     ;Pointer to original Trap Code
```

; This trap patcher leaves the patch in the app heap
; it will be unpatched when the app quits. For a persistent
; patch, the patch must be installed at INIT time in the system heap

```
    lea        OldSysError,a1 ;get the address of the pointer saver
    move.w    #kSysError,d0 ;now get address of SysError routine
    _GetTrapAddress ,NEWTOL
    move.l    a0,(a1) ;save it for later
    move.w    #kSysError,d0 ;set up trap num
    lea        MySysError,a0 ;set up new trap address
    _SetTrapAddress ,NEWTOL ;install the patch
    rts ;exit
```

ENDP

MySysError PROC EXPORT

IMPORT OldSysError

```
    cmpi.w    #20002,d0 ;first, check for the right selector
    bne.s    PassItOn ;if not ours, let the real SysError
have it
    rts ;if selector matches, return to sender
```

PassItOn

```
    move.l    OldSysError,-(sp) ;leave all registers unmolested
    rts
```

ENDP

OldSysError PROC EXPORT

DC.L 0

ENDP

END