

SetTrapAddress Install custom code to replace a system routine

#include <OSUtils.h>

Operating System Utilities

void **SetTrapAddress**(*trapAddr*, *trapNum*);
long *trapAddr* ; address of custom code
short *trapNum* ; the trap to intercept. See TrapWords.

SetTrapAddress changes an element of the trap dispatch table so that subsequent invocations of that trap will cause execution to go to a specified address. Use **NSetTrapAddress**, **SetToolTrapAddress** or **SetOSTrapAddress** if you know you are running on a 128K ROM or later.

trapAddr is the address of code to handle execution of an Operating System or Toolbox function. For the 64K ROMs, this value must be less than 64K (you can allocate a 6-byte record in the system heap and fill it with a jump to your code).

trapNum identifies the ROM routine you wish to replace. See TrapWords for a list.

Returns: none

Notes: Be sure not to depend on register A5 within your patch. This is because you may have intercepted a trap used by a desk accessory. Use **SetCurrentA5** and **SetA5** to be sure you do it right.

Note: Be sure to change all traps back to their original addresses before your application exits!

The trap dispatcher changed between the 64K and 128K ROMs. For more information, see **About Compatibility**.