

LockMemory

Make part of the address space immovable

#include <Memory.h>

Memory Manager**Debugger Support Under Virtual Memory**OSErr**LockMemory**(*address*, *count*);

void

**address*

is the start address of the memory range

unsigned long*count*

is the size of the range

returnsError Code; 0=no error

The **LockMemory** function makes a portion of the address space immovable in physical memory and ineligible for paging.

address is the start address of the range that is to be locked in RAM

count is the size in bytes of that range.

Returns: an operating system Error Code.

noErr (0) No error

paramErr (-50) Error in parameter list

notEnoughMemoryErr (-620) Insufficient physical memory

interruptsMaskedErr (-624) Called with interrupts masked

Notes: If the starting address parameter supplied to the **LockMemory** function is not on a page boundary, then it is rounded down to the nearest page boundary. Similarly, if the specified range does not end on a page boundary, the count parameter is rounded up so that the entire range of memory is locked. Locked pages are marked noncacheable by the CPU.