New Technical Notes

Macintosh

®

ME 8 - System Error 33, "zcbFree has gone negative"
Memory

Revised by:	March 1988
Written by: Bryan Stearns	July 1987

System 3.2 introduced a new system error, ID=33, generated by the Memory Manager when it notices that a heap had been corrupted in a certain way. This error is listed in the file "SysErr.a" as "`negZcbFreeErr`".

The Memory Manager will trigger an "ID=33" system error when, during some operation which scans the objects in the heap, it sees that its running count of free bytes (`zcbFree`, an internal value) has become negative (an impossible feat in normal operation). This is nearly always caused by writing zeros past the end of one of the Memory Manager's heap blocks (overwriting and corrupting the next block's header, making it appear to be a free block).

If you get this error, use a debugger (like Macsbug or TMON) when you attempt to reproduce the error, to check the consistency of the heap up to the point where the error occurs. You may need to do this repeatedly until you isolate the operation that is corrupting the heap.

Note that although the heap may become corrupted during a system call, this doesn't mean you've found a bug in the ROM; your code could be passing incorrect or invalid parameters to this or a previous system call, or could have corrupted a data structure used by a system call. More debugging is usually in order in this case; tools like Discipline (included in TMON; also available from users' groups and electronic services) can help detect invalid parameters in system calls. Also, there is a Macsbug command, AH, that can check the consistency of the heap on every system call. See the documentation that came with your debugger to see what special features it offers.

A note about "SysErr.a"

Technical Support is often asked for an up-to-date list of error codes. In general, this is provided in "SysErr.a", the file of error numbers shipped with the most current version of MPW. Admittedly, the documentation value of "SysErr.a" is sometimes low (as in the case of `negZCBFreeErr`), but it may give you a clue as to what the error might mean.

Further Reference:
- The Memory Manager