John Neil & Associates

P.O. Box 160699
Cupertino, CA 95016-0699

Software FPU™

Programmer Information

John Neil
February 25, 1993

What SoftwareFPU Does

SoftwareFPU is a control panel which allows programs that expect a Motorola 68881 or 68882 Floating Point Unit (FPU) to work correctly without one. It is intended to be used in any 68020 or greater Macintosh computer without an FPU. Since both SoftwareFPU and the FPU conform to the IEEE Standard for floating point arithmetic, the differences between the hardware FPU and SoftwareFPU are minimal. The current known incompatibilities with a hardware FPU are:

- FRESTORE does not support the busy state frame.
- FMOD produces the same result as FREM.
- Mid-instruction exceptions are reported as post-instruction exceptions.
- If an exception occurs in trace mode, two instructions will execute before control returns to the debugger.

P.O. Box 160699
Cupertino, CA 95016-0699

- Code which puts data below the stack pointer and then issues an FPU instruction will not work. This of course is an extreme no-no since data below the stack pointer can be clobbered by interrupt routines as well.
- Some emulated FPU instructions may produce more accurate results than on a hardware FPU.

The current list of known application incompatibilities are:

- Any program which patches the F-line exception vector itself will not work with SoftwareFPU. A development environment application would need to check for the presence of SoftwareFPU and patch a location inside the emulator to trap F-line exceptions. That location is 2 + the value of 68000 F-line exception vector.
- Programs using recursive algorithms may have problems. SoftwareFPU can require up to 600 bytes of stack space per FPU instruction. Recursive programs will probably require more stack space than usual to work properly, if they will work at all.
- A bug in the MPW 3.1 nan() function means that any program calling nan() will not work. Because of this, the MPW functions atan2(), asin(), and acos() will not work when they try to produce QNANs. Also, fscanf will not read in NANs correctly. The bug has been fixed in MPW 3.2.

P.O. Box 160699
Cupertino, CA 95016-0699

- Code which assumes that calls to SANE affect the FPU will not work. The MPW functions sinh() and cosh() may produce incorrect results in the exception status register, as the library code makes this assumption.

SoftwareFPU requires about 16K of system heap space to install correctly.

Performance

SoftwareFPU performance ranges from 30% to 85% of calling SANE directly, depending on the instruction type. However, applications typically do not slow down this much since they do not execute FPU instructions 100% of the time.

With SoftwareFPU installed, applications using direct FPU calls obtain a different performance tradeoff than using SANE. Take the performance of SANE on a non-FPU machine as a benchmark. SANE works about 10 times faster on machines with FPUs. Direct FPU calls run about 100 times faster on machines with FPUs, and 30-85% of SANE on non-FPU machines.

SysEnvirons/Gestalt Patch

The emulator patches Gestalt to indicate the presence of an FPU to applications. While this maximizes the number of

P.O. Box 160699
Cupertino, CA 95016-0699

programs that work, it would slow down the few programs that can perform either SANE or FPU calls because they will use the software FPU instead of SANE. The known applications of this type are DataDesk, Excel, Resolve, Lotus 1-2-3, and the Apple system software. The Gestalt patch checks to see if the current application is one of these two, and if so reports no FPU present.