Retrocomputing

How does the 80287 process infinity values?

Asked 4 years, 5 months ago Modified 4 years, 5 months ago Viewed 752 times

10

Modern x86 processors have an infinity control bit on the x87 FPU control register. This is bit 12 and it enables processing of infinity values in a manner that is compatible with the 80287 coprocessor. How does the 80827 process infinity values and do modern processors behave identically if the control bit is enabled?

80286

Share Improve this question Follow

asked Mar 23, 2020 at 21:50

Single Malt

1,869 ● 1 ● 8 ● 29

1 Answer

Sorted by: Highest score (default)

10

The 80287 (and 80187) were functionally identical to the original 8087 coprocessor, just with different external interfaces to match their companion CPUs. The 80387 (also produced as an 80287XL) was the first Intel x87-family FPU to introduce full IEEE-754 compatibility, and this involved some changes to the handling of infinities, Not-a-Numbers (NaNs) and denormals.

The IA32 architecture manual describes some of these changes thus:

> **8.2.2 Unsupported Double Extended-Precision Floating-Point Encodings and Pseudo-Denormals**
>
> The double extended-precision floating-point format permits many encodings that do not fall into any of the categories shown in Table 4-3. Table 8-3 shows these unsupported encodings. Some of these encodings were supported by the Intel 287 math coprocessor; however, most of them are not supported by the Intel 387 math coprocessor and later IA-32 processors. These encodings are no longer supported due to changes made in the final version of IEEE Standard 754 that eliminated these encodings.
>
> Specifically, the categories of encodings formerly known as pseudo-NaNs, pseudo-infinities, and un-normal numbers are not supported and should not be used as operand values. The Intel 387 math coprocessor and later IA-32 processors generate an invalid-operation exception when these encodings are encountered as operands. Beginning with the Intel 387 math coprocessor, the encodings formerly known as pseudo-denormal numbers are not generated by IA-32 processors. When encountered as operands, however, they are handled correctly; that is, they are treated as denormals and a denormal exception is generated. Pseudo-denormal numbers should not be used as operand values. They are supported by current IA-32 processors (as described here) to support legacy code.

It also states:

> **8.1.6 Infinity Control Flag**
>
> The infinity control flag (bit 12 of the x87 FPU control word) is provided for compatibility with the Intel 287 Math Coprocessor; it is not meaningful for later version x87 FPU coprocessors or IA-32 processors. See Section 4.8.3.3, "Signed Infinities," for information on how the x87 FPUs handle infinity values.

This states quite clearly that the Infinity Control bit is **not valid** on modern x87 FPUs, only on the 80287 and earlier. Modern x87 always performs infinity processing in accordance with IEEE-754. So, going back to the **80286 & 80287 Programmer's Reference Manual**...

> The infinity control bit (bit 12) determines the manner in which the 80287 treats the special values of infinity. Either affine closure (where positive infinity is distinct from negative infinity) or projective closure (infinity is treated as a single unsigned quantity) may be specified. These two alternative views of infinity are discussed in the section on Computation Fundamentals.

IEEE-754 specifies **affine closure**, with a distinct positive and negative infinity.

Share Improve this answer Follow

answered Mar 24, 2020 at 7:06

Chromatix
16.9k ● 1 ● 53 ● 70

2 The 80**1**87 was an 80**3**87 with a circuit that allowed connecting a 387 FPU to a 80186 CPU. (Source: Wikipedia) For this reason the 80187 will behave exactly as the 80387. – Martin Rosenau Mar 24, 2020 at 18:26