NAME

fpgetround, fpsetround, fpgetmask, fpsetmask, fpgetsticky, fpsetsticky – IEEE floating-point environment control

SYNOPSIS

#include <ieeefp.h>

fp_rnd fpgetround(void);

fp_rnd fpsetround(fp_rnd *rnd_dir***);**

fp_except fpgetmask(void);

fp_except fpsetmask(fp_except *mask***);**

fp_except fpgetsticky(void);

fp_except fpsetsticky(fp_except *sticky***);**

MT-LEVEL

MT-Safe

DESCRIPTION

There are five floating-point exceptions: divide-by-zero, overflow, underflow, imprecise (inexact) result, and invalid operation. When a floating-point exception occurs, the corresponding sticky bit is set (1), and if the mask bit is enabled (1), the trap takes place. These routines let the user change the behavior on occurrence of any of these exceptions, as well as change the rounding mode for floating-point operations.

The following floating-point exception masks are OR-ed together to form *mask.*

```
FP_X_INV    /* invalid operation exception */
FP_X_OFL    /* overflow exception */
FP_X_UFL    /* underflow exception */
FP_X_DZ     /* divide-by-zero exception */
FP_X_IMP    /* imprecise (loss of precision) */
```

The following floating-point rounding modes are passed to **fpsetround ()** and returned by **fpgetround()**.

```
FP_RN       /* round to nearest representative number */
FP_RP       /* round to plus infinity */
FP_RM       /* round to minus infinity */
FP_RZ       /* round to zero (truncate) */
```

The default environment is rounding mode set to nearest (**FP_RN**) and all traps disabled.

Individual bits may be examined using the constants defined in **<ieeefp.h>**.

RETURN VALUES

fpgetround() returns the current rounding mode.

fpsetround() sets the rounding mode and returns the previous rounding mode.

fpgetmask() returns the current exception masks.

fpsetmask() sets the exception masks and returns the previous setting.

fpgetsticky() returns the current exception sticky flags.

fpsetsticky( ) sets (clears) the exception sticky flags and returns the previous setting.

SEE ALSO

isnan(3C)

NOTES

fpsetsticky( ) modifies all sticky flags. **fpsetmask( )** changes all mask bits. **fpsetmask( )** clears the sticky bit corresponding to any exception being enabled.

C requires truncation (round to zero) for floating point to integral conversions. The current rounding mode has no effect on these conversions.

One must clear the sticky bit to recover from the trap and to proceed. If the sticky bit is not cleared before the next trap occurs, a wrong exception type may be signaled.