

## GCC3. 4. 5 FAQ

### ◆ Questions

- How to avoid FPU error?
  - How to round a denormalized number to zero?
- 

### ◆ Answers

2 empirical solutions are given below.

- (1) Force to change DN bit into 1 during application execution.

The compiler outputs the mask data for rewriting FPSCR as a table format. Rewritten this data forces to change DN bit into 1 at any time. On the actual procedure, add the following 3 lines to initial processing of the application.

```
extern int __fpscr_values[];
__fpscr_values[0]=0x00040000;
__fpscr_values[1]=0x000c0000;
```

However, in this procedure, every demormalized number is rounded to zero. The accuracy of the operation with generating denormalized number is lower.

- (2) Remodel the execution environment initial setting runtime routine.

Take glibc-2.3.3 as an example, the source of the runtime routine exists in (glibc source directory)/sysdeps/sh/elf/start.S. Rewrite this file to adjust the DN bit. The patch is given below.

```
--- sysdeps/sh/elf/start.S      2004-08-16 13:50:55.000000000 +0900
+++ sysdeps/sh/elf/start.S.new  2006-04-25 10:42:32.000000000 +0900
@@ -108,5 +108,5 @@
     data_start = __data_start
     .global __fpscr_values
 __fpscr_values:
-     .long    0
-     .long    0x80000
+     .long    0x00040000
+     .long    0x000C0000
```

Note: On using math function, there may be insufficient cases.

## Appendix : FPU exception

FPU exceptions are as follows:

- (1) Inexact exception : When overflow, underflow, or rounding occurs
- (2) Underflow : When the operation result underflows
- (3) Overflow : When the operation result overflows
- (4) Division by zero : Division with a zero divisor
- (5) Invalid operation : In case of an invalid operation, such as NaN input
- (6) FPU error : When FPSCR.DN = 0 (zero) and a denormalized number is input

In (1)-(5) on the FPU exception, a bit corresponding in FPSCR Enable field indicates an exception cause to process. When the bit is 1, FPU exception occurs to call routine for exceptional process. When the bit is 0 (zero), FPU exception is not called, FPU runs exception cause process as follows.

FPU exception cause	processing
(1) Inexact exception	An inexact result is generated.
(2) Underflow	When FPSCR.DN = 0 (zero), a denormalized number with the same sign before rounded value or (0.0) is generated.
	When FPSCR.DN = 1, (0.0) with the same sign before rounded value is generated.
(3) Overflow	In round to zero mode, the maximum normalized number with the same sign before rounded value is generated.
	In round to the closet mode, infinity with the same sign before rounded value is generated.
(4) Division by zero	Infinity with the same sign before rounded value is generated.
(5) Invalid operation	qNaN is generated as the result.

- (6) FPU error, in this error there is no corresponding bit in the FPU exception enable field. The FPU error is controlled by switching FPSCR.DN.

DN bit is the bit to switch processing method when demormalized number is input. When FPSCR.DN = 0 (zero), the FPU processes denormalized number without changing, and generate FPU exception.

When FPSCR.DN = 1, the demormalized number is processed as (0.0), and FPU exception does not occur.