

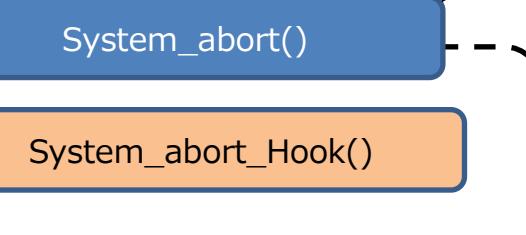
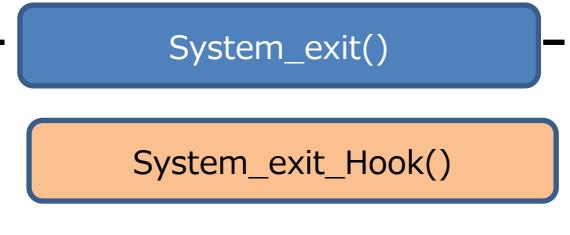
■ Application



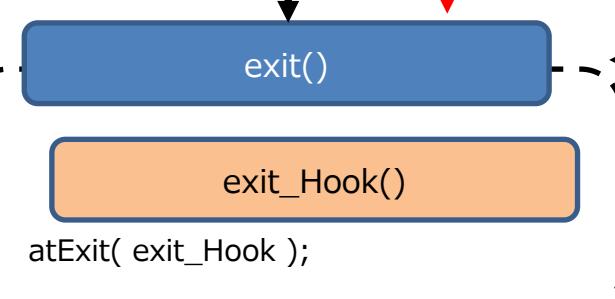
■ Sysbios/PDK



■ XDC TOOL Class



■ GCC newlib



1 Cases where Sysbios / PDK sets NG for parameter check

Assert Macro

xdc_runtime Assert_raise()

Error_raise_Hook()

Various HOOK handler calls

2 Case of calling System_abort

3 Case of calling System_exit()

4 Case of calling abort

5 Case of calling exit()

Our implement code.

TI implement code.

→ Where you want to make sure there are no calls.

Comment

Cases 1 to 3 can be detected with System_exit_Hook() / System_abort_Hook(). ... No problem
Case 5 is atExit () and can be detected by registering the HOOK function. ... No problem.
In the case of 4, the HOOK function registered with atExit () is not called from exit () . (Only infinite loop) Undetectable

In the future, even if there is no use point at the present time
If a third-party library is installed, the same confirmation will be required.

I think that it is better to present the procedure of implementing exit () / abort () with symbol overriding as the official procedure of TI.

Because it is not limited to our environment.

for example

```
myOverride_abort_exit.c
void abort(void) // gcc override symbol
{
    // ANY Proc
    // Turn on the Industrial Red LED on my device!!
    _exit(999); // primitive exit() function.
}
void exit(int stat) // gcc override symbol
{
    // ANY Proc
    // Turn on the Industrial Red LED on my device!!
    _exit(stat); // primitive exit() function.
}
```