[Notification]

R20TS0378EJ0100
Rev.1.00
Dec. 16, 2018

Dynamic Checking for Corruption in Stack Area for Quality and Security Enhancement!

Introducing Detection of Stack Smashing Feature of Renesas Compiler Professional Edition

Outline

This document introduces one of the features of the Renesas Compiler (CC-RL, CC-RX, and CC-RH) professional edition; Detection of Stack Smashing.

This feature enables you to check for any corruption in the stack area while running the program. It prevents program runaway or malfunction that might occur when the stack area is corrupted by a program bug or security attack.

1. Features

1.1 Improves Quality and Security of Programs

Detection of stack smashing feature executes error processing when the stack area is smashed. This prevents the program from entering runaway or malfunctioning, enabling **the development of programs with enhanced quality and safety.**

This function detects any corruption in the stack area by embedding an arbitrary value in the stack area before executing a function and checking this value after the execution of the function. In this way program runaway or malfunction is prevented.

The processing without the detection of stack smashing feature and the processing with this feature are shown in Figure 1 and Figure 2.

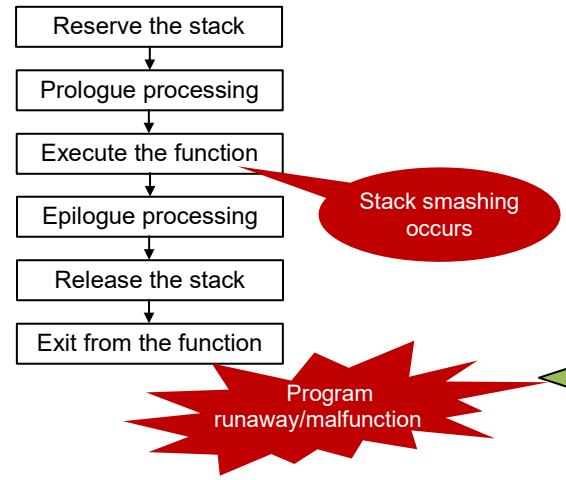

Figure 1 Standard Processing

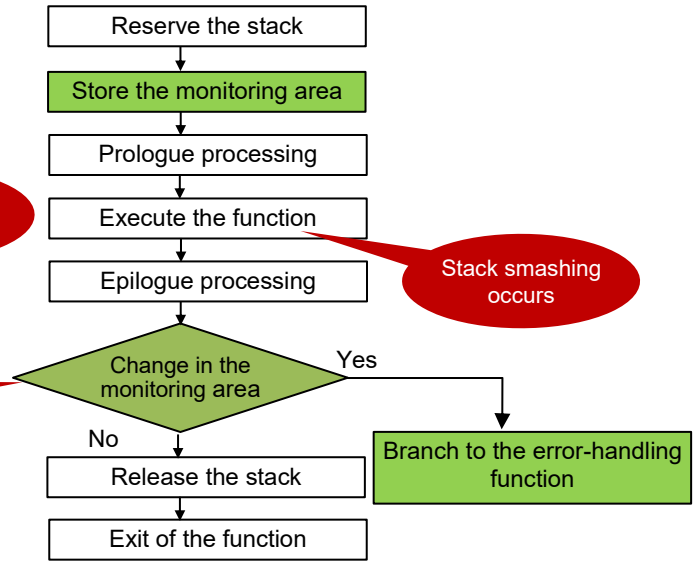

Figure 2 Processing with Stack Smashing Detection Function

The stack area used by the function is reserved at the entry of the function. An area called the monitoring area (2-bytes in CC-RL, 4-bytes in CC-RX/CC-RH) is allocated immediately before the local variable area and stores an arbitrary value (Figure 3). The monitoring area does not change when executing standard processing, however, the value in the monitoring area changes if the stack area is smashed by some means (Figure 4).

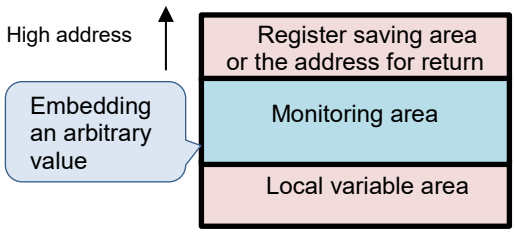

Figure 3 [Before executing the function] Monitoring area

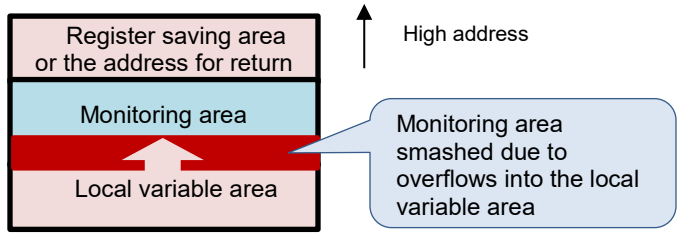

Figure 4 [When processing the function] Stack Smashing

After executing the function, the compiler checks that the value stored in the monitoring area is still the same. If the value has been overwritten, it is determined as stack smashing and error processing is executed (Figure 5).
The error-handling function can be defined by users.

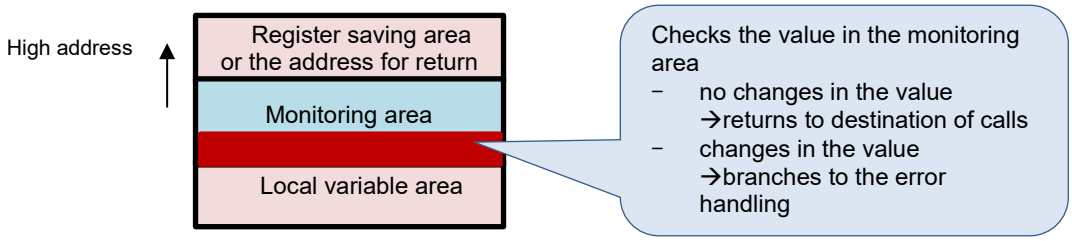

Figure 5 [After executing function] Checking Monitoring Area

Specifying the error-handling function when the stack is corrupted can prevent program runway or malfunction, allowing you to develop programs with improved quality and safety.

1.2 Application Example

The detection of stack smashing function can be used in either one of the following ways.

(1) Enable the detecting stack smashing function in the property of IDE (CS+ or e^2 studio).

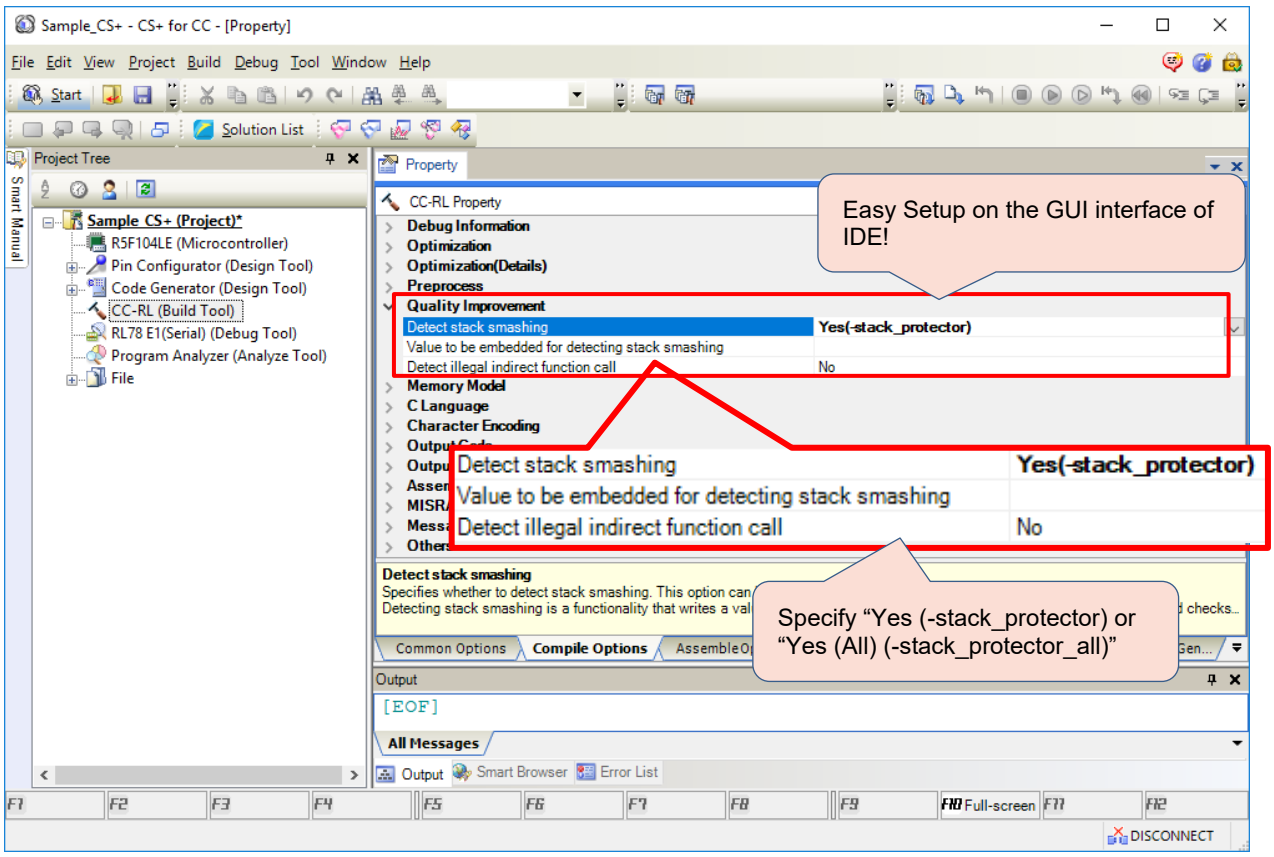

Figure 6 CS+ Settings (for CC-RL)

(2) Specify the target function by using the #pragma extended language directive.

"#pragma stack_protector" specifies the target function for detection of stack smashing.

```
#pragma stack_protector function name (num=specified value)
```

Embeds a specified value in the monitoring area. If the specification of "(num=specified value)" is omitted, the compiler automatically selects the number to be embedded in the monitoring system.

Next, define the error-handling function.

When stack smashing is detected, the __stack_chk_fail function is called for error processing.
The processing of __stack_chk_fail function can be defined by users.

```
void __stack_chk_fail(void) {
// Define Error Processing
}
```

2. Other Features of Professional Edition

- MISRA-C Rule Checking Feature

  This function is featured in the TOOLNEWS below.

  Learn more about this feature from the URL below:

  https://www.renesas.com/search/keyword-search.html#genre=document&q= r20ts0342

  [Notification]

  Perform MISRA-C Rule Check During Compilation to Reduce Man-hours and Improve Quality for Program Development!

  MISRA-C Rule Checking Feature of Renesas Compiler Professional Edition

- Synchronization Features in the Updating of Control Registers

  This function is featured in the TOOLNEWS below.

  Learn more about this feature from the URL below:

  https://www.renesas.com/search/keyword-search.html#genre=document&q= r20ts0347

  [Notification]

  Automatic Insertion of Synchronization Processing to Reduce Man-hours for Development of RH850 Family!

  Synchronization Features in the Updating of Control Registers of Renesas Compiler Professional Edition

- Other Useful Features

  Renesas Compiler professional edition provides additional features such as:

  Detection of Illicit Indirect Function Calls Enhanced Security for Dynamic Memory Management Functions, Half-precision Floating Point

  For details, refer to the following leaflet:

  https://www.renesas.com/search/keyword-search.html#genre=document&q=r20pf0024

  Renesas Compiler Professional Edition

For details about features of the Renesas Compiler Professional edition, refer to the Application Note below. It introduces features to help improve the quality of your programs and reduce the development time. There is also an example of the C source code that can be copied and pasted to try out right away.

https://www.renesas.com/search/keyword-search.html#genre=document&q=r20ut4026
Renesas Compilers Professional Editions

3. How to Purchase a Product

To order a product, contact your local Renesas Electronics sales office or distributor.

If you have a node-locked license for a standard edition, you can upgrade your compiler from the standard to professional edition by additionally purchasing an upgrade (edition) license. For orderable part number, refer to the following web page for the compiler packages.

CC-RL: https://www.renesas.com/rl78_c
CC-RX: https://www.renesas.com/rx_c
CC-RH: https://www.renesas.com/rh850_c

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Dec. 16, 2018	-	First edition issued

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061 Japan
Renesas Electronics Corporation

■Inquiry
https://www.renesas.com/contact/