

RH850ファミリ用Cコンパイラパッケージのご使用上のお願い

RH850ファミリ用Cコンパイラパッケージ CC-RHの使用上の注意事項を連絡します。

- 変数に一致と大小比較判定を両方行う場合の注意事項 (No.1)

注: 注意事項の後ろの番号は、注意事項の識別番号です。

1. 該当製品

CC-RH V1.00.00～V1.01.00

2. 内容

同一関数内で、if文 または ループの制御式として、"!=" または "==" による比較式と、大小比較判定式を組み合わせて記述している場合に、そのif文 または ループ の判定を誤る場合があります。

3. 発生条件

以下を全て満たす場合に発生することがあります。

- (1) -Odefaultオプション、-Osizeオプション または -Ospeedオプション のいずれかが有効である。
- (2) 定数(注1)と変数(注2)を "!=" または "==" により比較する式が存在する。
注1: 静的に定数と分かる式を含む。
注2: 配列変数、構造体メンバ、および共用体メンバを含む。
- (3) (2)の比較式を含む関数内に、定数と(2)の変数を "<", ">", "<=", または ">=" により比較する式が存在する。
- (4) (2)の変数はvolatile修飾されていない。
- (5) (2)および(3)の比較式は以下のいずれかを満たす。
 - (5-1) (2)と(3)の比較式が "||" または "&&" で結合されている
 - (5-2) (2)と(3)の比較式がそれぞれ "if文" または "?:式" の条件式にありかつそれぞれの "if文" または "?:式" が連続して実行される
- (6) (2)と(3)の比較式間に別の式および文がない。

発生条件例: -Odefaultオプションが有効な場合 発生条件(1)

```
-----  
int g(void);  
int f(void)  
{  
    int x = g(); // 発生条件(4)  
    if (x == -2 || // 発生条件(2)(5-1)  
        x > 1200) { // 発生条件(3)(6)  
        return 1;  
    }  
    return -1;  
}  
-----
```

4. 回避策

以下のいずれかで回避可能です。

- (1) -Onothingオプションを指定する。
- (2) 発生条件(2)の変数をvolatile修飾する。
- (3) 発生条件(2)および(3)の比較のうち後に実行される方の直前に、
ダミーのvolatile変数を参照する。

回避策の適用例: 回避策(3)の場合

```
-----  
int g(void);  
volatile int dummy; // ダミーのvolatile変数の宣言  
int f(void)  
{  
    int x = g();  
    if (x == -2 ||  
        (dummy, x > 1200)) { // 回避策(3)  
        return 1;  
    }  
    return -1;  
}  
-----
```

5. 恒久対策

次期バージョンで改修する予定です。

[免責事項]

過去のニュース内容は発行当時の情報をもとにしており、現時点では変更された情報や無効な情報が含まれている場合があります。ニュース本文中のURLを予告なしに変更または中止することがありますので、あらかじめご承知ください。

