

CS+ CXコンパイラ ご使用上のお願い

CS+ CXコンパイラ (V850E2Mコア用) の使用上の注意事項を連絡します。

- 64Kバイト境界に配置された8バイト型変数の注意事項 (No.16)

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

CX V1.20~V1.31

2. 内容

8バイト型変数が64Kバイト(0x10000)境界に配置された場合に、正しくアクセスできない場合があります。

3. 発生条件

以下(1)~(2)を全て満たす場合に、(1)の8バイト型変数に正しくアクセスできません。

(1) 8バイト型(long long型 または double型)の外部変数 または 静的変数 (構造体メンバ または 共用体メンバ、および配列を含む)にアクセスしている。

(2) (1)の変数が以下のいずれかの条件を満たす。

(2-1) data属性セクション(data または .bssなど)の変数であり、かつ、以下のアドレスに配置される。

$$\text{GPレジスタの値} - (32\text{K} + 4) + (64\text{K} \times N)$$

$N=1,2,3,\dots$ または $N=-1,-2,-3,\dots$

(2-2) const属性セクション(.constなど)のconst変数であり、かつ、以下のアドレスに配置される。

$$(32K - 4) + (64K \times N)$$

$$N=0,1,2,3,\dots$$

(2-3) 96Kバイト以上の構造体 または 構造体配列のメンバであり、
かつ、ポインタを用いて以下のオフセットでアクセスされる。

$$(96K - 4) + (64K \times N)$$

$$N=0,1,2,3,\dots$$

発生条件例: 発生条件(2-3)

```
-----  
struct ST {  
    char array[0x17ffc];    // 0x17ffc = 96K - 4  
    long long ll;  
} ss, *p;  
  
...  
p->ll = 0;                // 発生条件(2-3)でN=0の場合  
-----
```

4. 回避策

ダミーの変数を代入して、
該当する8バイト型変数のアドレスを変更してください。

回避策の適用例: 発生条件(2-3)の場合

```
-----  
struct ST {  
    char array[0x17ffc];  
    char dummy;           //ダミー変数を定義  
    long long ll;  
} ss, *p;  
  
...  
p->ll = 0;  
-----
```

5. 恒久対策

今後のバージョンで改修予定です。(次のリリース時期は未定です)

[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。

© 2010-2016 Renesas Electronics Corporation. All rights reserved.