

## H8SX, H8S およびH8ファミリ C/C++コンパイラパッケージ V.6 ご使用上のお願い

H8SX, H8S およびH8用 C/C++コンパイラパッケージ V.6 の使用上の注意事項を連絡します。

- 構造体型配列メンバを持つ入れ子の構造体の場合、間違ったメモリ領域を参照する (H8C-0026)

### 1. 該当製品

H8, H8S, およびH8SXファミリC/C++コンパイラパッケージ

V.6.00 Release 00 ~ V.6.00 Release 03, V.6.01 Release 00 および V.6.01 Release 01

製品型名 :

Windows版	R0C40008XSW06R
Solaris版	R0C40008XSS06R
HP-UX版	R0C40008XSH06R

### 2. 内容

構造体の入れ子として構造体を宣言し、入れ子の構造体が配列型であった場合、間違ったアドレスにアクセスすることがあります。

#### 2.1 発生条件

以下の条件をすべて満たす場合に発生します。

- (1) cpu=h8sxn, h8sxm, h8sxa, h8sxxまたはae5オプションを選択している。

V.6.01 Release 00以降のコンパイラパッケージで legacy=v4オプションを選択しなかった場合は、前述したオプションに加え、cpu=2000n, 2600n, 2000a または2600aオプションを選択している場合も該当します。

- (2) 構造体型変数を宣言している。
- (3) (2)の構造体に、2段以上の構造体の入れ子を宣言し、もっとも深い入れ子の構造体を配列型で宣言している。
- (4) (2)の構造体の先頭のメンバに、(3)の入れ子の構造体のメンバを宣言していない。
- (5) 構造体型変数の参照または設定に、ドット演算子を用いている。

## 2.2 発生例

```
-----  
-  
struct {  
    int data;        // 発生条件(4)  
    struct {        // 発生条件(3) 1段目の入れ子の構造体  
        struct {    // 発生条件(3) 2段目の入れ子の構造体  
            int a;  
            int b;  
        }x[2];      // 発生条件(3) 最も深い入れ子が構造体型の  
        配列  
    }y;  
}z;                // 発生条件(2)  
int v;  
  
void func(int offset){  
    v = z.y.x[offset].a; // 発生条件(5)  
}  
-----  
-
```

## 3. 回避策

以下のいずれかの方法で回避してください。

- (1) 構造体の先頭メンバに、発生条件(3)の入れ子の構造体を配置する。

例)

```
-----  
-  
struct {  
    struct {        // 先頭メンバに構造体型を配置する  
        struct {  
            int a;  
        }  
    }  
}
```

```

        int b;
        }x[2];
    }y;
    int data;
}z;
int v;

void func(int offset){
    v = z.y.x[offset].a;
}

```

-----

- (2) 構造体型変数へのアクセス方法をポインタに変更する。

例)

-----

```

struct str{
    int data;
    struct {
        struct {
            int a;
            int b;
        }x[2];
    }y;
}z;
int v;

void func(int offset){
    struct str *p = &z; // ポインタ型変数を宣言する

    v = p->y.x[offset].a; // ポインタでアクセスする
}

```

-----

#### 4. 恒久対策

次期リビジョンアップで改修する予定です。(2006年1Q予定)

---

#### [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.