

H8C/C++コンパイラパッケージ V6 使用上のご注意

H8C/C++コンパイラパッケージV6には以下の不具合3件(コンパイラ1件、アセンブラ2件)があります。ご使用の際にはご注意くださいようお願い申し上げます。

なお、本不具合につきましては、次期リビジョンで改修する予定です。

1. 該当製品

H8C/C++コンパイラパッケージ

2. コンパイラ (V.6.00.02) 注意事項

【該当製品】

R0C40008XSW06R (V6.00 Release02)

R0C40008XSS06R (V6.00 Release02)

R0C40008XSH06R (V6.00 Release02)

2.1 ビットフィールドメンバの設定/参照に関する注意事項(H8C-0002)

【現象】

ビットフィールドメンバをアドレスへ設定又は、アドレスから参照した場合に、オブジェクト不正となる問題が発生する場合があります。

【発生条件】

以下のすべての条件を満たした場合に発生する場合があります。

- (1) CPUオプションにH8SXAまたはH8SXXを選択している
- (2) 構造体に7ビット以下のビットフィールドメンバが宣言されている
- (3) (2)の構造体変数は、レジスタに割りついていない
- (4) (3)の変数のビットフィールドメンバに値の設定、または、参照をする式がある
- (5) (4)の対象となるメンバの構造体の先頭からのオフセットが以下の場合
アドレスサイズ :20 0x0-0x7fff 又は 0x000f8000- 0x000fffff

アドレスサイズ :24 0x0-0x7fff 又は 0x00ff8000- 0x00ffffff

アドレスサイズ :28 0x0-0x7fff 又は 0x0fff8000- 0x0fffffff

アドレスサイズ :32 0x0-0x7fff 又は 0xffff8000- 0xffffffff

【回避策】

参照するビットフィールドのメンバを持つ構造体変数のアドレスを、その構造体のポインタ型変数へ代入し、ポインタ変数を介して参照する。

```
(修正前) -----
typedef struct { unsigned char a:2;          // 発生条件(2)(3)(5) unsigned char
b:2; unsigned char c:4;}st;extern st str;          // 発生条件(3)void
func(unsigned char);char ff(char var01){ func(str.b);          // 発生条件(4)}-
----- (修正後) -----
-----typedef struct {
unsigned char a:2; unsigned char b:2; unsigned char c:4;}st;extern st str;volatile
st *tmp;void func(unsigned char);char ff(char var01){ tmp = &str; //ポインタ変数に
構造体変数のアドレスを代入 func(tmp->b);//ポインタを介してビットフィールドメンバを
参照}------
```

【恒久対策】

次期リビジョンで改修する予定です。(2004年10月リリース予定)

3. アセンブラ (V.6.0.01, V.6.0.02) 注意事項

【該当製品】

R0C40008XSW06R (V6.00 Release00, Release01, Release02)

R0C40008XSS06R (V6.00 Release00, Release01, Release02)

R0C40008XSH06R (V6.00 Release00, Release01, Release02)

3.1 構造化アセンブリ機能の制御文の分岐サイズ記述時のエラーに関する注意事項

【現象】

構造化アセンブリ機能の制御文にサイズ記述なしで分岐サイズを指定した場合(E)300 ILLEGAL MNEMONICが出力される問題が発生します。

【発生条件】

構造化アセンブリ機能の制御文をサイズ記述なしで分岐サイズを記述した場合に発生します。

【回避策】

以下のいずれかの方法で回避することができます。

- (1) 分岐サイズ記述を削除し、DISPSIZE制御命令で分岐サイズを記述する。かつ、BR_RELATIVEオプションおよびOPTIMIZEオプション選択しない。

```

(修正前) -----
.IF:16 (R0L<EQ>R1L) ;(E)300 ILLEGAL MNEMONIC add.w #1,r0
.ELSE:8 ;(E)300 ILLEGAL MNEMONIC add.w #2,r0 .ENDI
;(E)631 END DIRECTIVE MISMATCH-----
(修正後) -----
.DISPSIZE FBR=16 ;追加 .IF (R0L<EQ>R1L)
;分岐サイズ記述削除 add.w #1,r0 .DISPSIZE FBR=8 ;追加 .ELSE
;分岐サイズ記述削除 add.w #2,r0 .ENDI-----

```

(2) サイズ記述が可能な場合は、サイズ記述する。

```

(修正前) -----
.IF:16 (R0L<EQ>R1L) ;(E)300 ILLEGAL MNEMONIC add.w #1,r0
.ENDI----- (修正後) -----
.IF.B:16 (R0L<EQ>R1L) ;サイズ記述の追加 add.w #1,r0 .ENDI-----

```

ただし、サイズ記述のない、.ELSE, .CASE, .BREAK, .CONTINUEは、本回避策で回避できません。回避策(1)で回避してください。

【恒久対策】

次期リビジョンで改修する予定です。(2004年10月リリース予定)

3.2 不当遅延スロット命令に対するエラー未出力に関する注意事項

【現象】

遅延スロットの命令に、1ワード命令以外を記述した場合に、(E)150 INVALID DELAY SLOT INSTRUCTIONを出力されない問題が発生します。

【発生条件】

以下のすべての条件を満たした場合に発生します。

【発生条件】

以下のすべての条件を満たした場合に発生します。

- (1) CPUオプションにH8SXN,H8SXM,H8SXAまたはH8SXXを選択している
- (2) 遅延スロットの命令に1ワード命令以外を記述している

```

(発生例) .import _a .import _b .SECTION P,CODE,ALIGN=2 L1:
BRA/S L2 ;遅延分岐命令 MOV.L @_a,@_b ;遅延スロット命令(8バイト
長命令) L2:

```

【対応策】

アセンブラプログラム内の遅延スロット命令が1ワード命令以外かを確認してください。

【恒久対策】

次期リビ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。

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