H8S,H8/300 シリーズ C/C++コンパイラ V.6.00.02 で対策した制限事項

1. 構造体の配列型メンバのアドレス取得不正

現象

構造体に配列型メンバを先頭以外に宣言し、配列型データのアドレスを参照する式を記述した場合、正しいアドレスを取得できない。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。
(2) 構造体の先頭以外のメンバに配列型データを宣言している。
(3) 配列型メンバの先頭でない要素のアドレスを取得する式を記述している。

2. プレ/ポスト インクリメント/デクリメントの不正な最適化

現象

speed オプションを指定し、プレインクリメント/プレデクリメント、ポストインクリメント/ポストデクリメントのいずれかを記述した場合、正しい値が得られないことがある。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。
(2) speed オプションを指定している。
(3) ソース内にて、以下のいずれかを記述している。
- プレインクリメント
- プレデクリメント
- ポストインクリメント
- ポストデクリメント

3. table 展開方式 switch 文でレジスタ割り付け不正

現象

table 展開方式で switch 文を展開した場合、レジスタの退避/回復が正しく行われない。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。

(2) case=table オプションを指定している。
　　または case=auto オプションを指定し、table 方式で switch 文が展開されている。
(3) switch 文の直前に、レジスタ退避回復命令が出力されている。

4. 0 クリアコード不正削除

現象

分岐ごとに 0 を設定するコード(SUB 命令)を記述した場合、0 を設定するコードを不当に削除してオブジェクト不正となる。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8/300, H8/300L のいずれかを指定している。
(2) 最適化あり(optimize=1:デフォルト)を指定している。
(3) 分岐毎に 0 クリアを行うコードを記述している。

5. 4byte 以下の構造体の代入式不正削除

現象

4 バイト以下の構造体変数を使用した場合に、代入式が不当に削除する。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。
(2) 最適化あり(optimize=1:デフォルト)を指定している。
(3) 宣言した構造体が次のいずれかの条件を満たしている。
　(a) 構造体の先頭から 2byte を 1byte でアクセスするメンバ、もしくは構造体の先頭から 2byte 目を 2byte でアクセスするメンバが存在する。
　(b) サイズが 2,3,4 バイトのいずれかの構造体で、2 つ以上のメンバを持ち、少なくともそのひとつのメンバがビットフィールドメンバである。
(4) (3)のメンバをアクセスしている。

6. 構造体メンバの参照方法混在時不正

現象

同一メモリ領域をポインタを使用した式と使用しない式で連続してアクセスした場合、設定コードを不当に削除する。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。
(2) 最適化あり(optimize=1:デフォルト)を指定している。
(3) 同じメモリ領域への値の設定/参照の式が連続して存在する。
(4) (3)の式はそれぞれポインタを使用した式とポインタを使用しない式で構成されている。

7. ループ内帰納変数の 2 次式の計算で結果不正

現象

ループ内で、m*(i*i+b*i)という形のループ変数 i の 2 次式がある場合、最適化により結果不正となる。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN, H8SXM, H8SXA, H8SXX のいずれかを指定している。
(2) 最適化あり(optimize=1：デフォルト)を指定している。
(3) 下記の条件を全て満たすループが存在する。
  (a) long/unsigned long 型のループ変数がある。
  (b) (a)のループ変数 2 次式がそのループ内に存在する。
  (c) (b)のループ変数を x とすると、(b)の 2 次式が「m * (x * x + b * x)」という形をしている。

8. 引数の構造体/共用体メンバのアクセス不正

現象

structreg オプションを指定し、４バイト以下の引数の構造体/共用体メンバにアクセスした場合、正しく値が反映されない。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として H8SXN/H8SXM/H8SXA/H8SXX のいずれかを指定している。
(2) structreg オプションを指定している。
(3) 4 バイト以下の構造体または共用体を引数として使用している。
(4) 引数がアドレス参照されている場合、もしくは関数内部でアドレス参照される場合。

9. ポインタ比較式不正

現象

ポインタにキャストした定数同士を比較した場合、比較結果が異なる。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) 比較式がある。
(2) (1)の両辺が共にポインタ型にキャストされた定数である。
(3) (2)の定数のうち、少なくとも一方が signed long の範囲外(2147483648～4294967295)の値である。

10. ビットフィールドの設定・参照不正

現象

ビットフィールドへの設定および参照を行った場合、値の設定や参照が正しく行えない。

発生条件

次の(a),(b)いずれかの条件を全て満たす場合、発生することがある。

(a)
  (1) CPU 種別として 300HN, 300HA のいずれかを指定している。
  (2) 最適化あり(optimize=1:デフォルト)でコンパイルしている。
  (3) 引数または局所変数の構造体を宣言している。
  (4) (3)の構造体が[unsigned] char 型で、サイズが 8bit のビットフィールドメンバを持つ。
  (5) (3)の構造体はレジスタ上に割りつき、(4)のビットフィールドメンバは En 上に割りついている。

(b)
  (1) CPU 種別として 300HN, 300HA, 2000N, 2000A, 2600N, 2600A のいずれかを指定している。
  (2) 最適化あり(optimize=1:デフォルト)でコンパイルしている。
  (3) 引数または局所変数の構造体を宣言している。
  (4) (3)の構造体が[unsigned] short または[unsigned] int で、サイズが 16bit のビットフィールドメンバを持つ。
  (5) (3)の構造体は、境界調整数が 1 である(pack=1 オプション、#pragma pack 1 が指定されている)。
  (6) (3)の構造体はレジスタ上に割りつき、(4)のビットフィールドメンバは En の下位 8bit と RnH に割りついている。

11. &構造体.配列[0] 等構造体メンバのアドレス参照時エラー

現象

&構造体.配列[0](&構造体->配列[0])形式で先頭アドレスを参照した場合、正しいアドレスが求められない、および内部エラーが出力される場合がある。

発生条件

次の(a),(b)いずれかの条件を全て満たす場合、発生することがある。

(a)

(1) CPU 種別として 300HN, 300HA, 2000N, 2000A, 2600N, 2600A のいずれかを指定している。
(2) 構造体メンバが配列で定義されている。
(3) (2)の先頭のアドレス値を用いて 2 回以上の加減算を実施する。
(4) 先頭アドレスは&構造体.配列[0]、または&構造体->配列[0]形式で求めている。

(b)

(1) CPU 種別として 300HN, 300HA, 2000N, 2000A, 2600N, 2600A のいずれかを指定している。
(2) 最適化あり(optimize=1:デフォルト)でコンパイルする。
(3) 変数の配列を定義している。
(4) (3)の先頭のアドレス値を用いて、2 回以上の加減算を実施する。
(5) 先頭アドレスを&配列[0]形式で求めている。

12. &=0、|=0xFFFF の不正アドレスアクセス

現象

[unsigned]short/int 型の変数に対して複合論理演算を実施した場合、不正なアドレス(本来のアドレス+2)に値を設定するコードが生成される。

発生条件

次の条件を全て満たす場合、発生することがある。

(1) CPU 種別として 300, 300HN, 300HA, 2000N, 2000A, 2600N, 2600A のいずれかを指定している。
(2) 変数を[unsigned] short/int 型で宣言している。
(3) 変数は volatile 宣言されたポインタである。
(4) 下記の複合代入演算を記述している。
・変数 &=0;
・変数 |= 0xffff;

株式会社 日立超LSIシステムズ

(c) Hitachi ULSI Systems Co., Ltd. 1995,2014. All rights reserved.