

*****制限事項通知統合版*****
 *
 * 本文書は、マイクロコンピュータ技術文書閲覧サービスの開始にあたり、
 * 既に通知させて戴いておりました制限事項を1つの文書にまとめたものです。
 * 2004年8月18日時点のすべての制限事項が掲載されています。
 *

マイクロコンピュータ技術情報

技術通知 78K4 シリーズ アセンブラ・パッケージ RA78K4 使用制限事項の件		発行番号	ZBG-CD-04-0053号	1 / 1
		発行日	2004年 8月 18日	
		発行部門	NEC エレクトロニクス株式会社 第三システム事業本部 汎用マイコンシステム事業部 開発ツールグループ	
文書分類	○ 使用制限事項	バージョン・アップ	ドキュメント誤記訂正 (正誤表)	その他
関連資料	RA78K4 アセンブラ・パッケージ Ver.1.60 以上 操作編		資料番号:U16708JJ1V0UM00(第1版)	
	RA78K4 アセンブラ・パッケージ Ver.1.50 以上 言語編		資料番号:U15255JJ1V1UM00(第1版)	
	RA78K4 アセンブラ・パッケージ 構造化アセンブラ・プリプロセッサ ST78K4 V1.10 以上		資料番号:U11743JJ1V1UMJ1(第1版)	
	PM plus Ver.5.10		資料番号:U16569JJ1V0UM00(第1版)	
	78K4 シリーズ アセンブラ・パッケージ RA78K4 V1.60 使用上の留意点		資料番号:SUD-DT-03-0440	

CP(K),O

1. 対象製品

RA78K4 Ver.1.60

2. 制限事項一覧

制限事項の履歴とその詳細情報が含まれました使用制限事項一覧を別紙1に記載します。

3. 発行文書履歴

78K4 シリーズ アセンブラ・パッケージ RA78K4 使用制限事項一覧 発行文書履歴

文書番号	発行日	記事
ZBG-CD-04-0053	2004年8月18日	統合版初版

以上

RA78K4の使用制限事項一覧

1. 製品履歴

No.	仕様変更・追加／不具合事項	バージョン		
		V1.50	V1.51	V1.60
1	アセンブラ・ソースのマクロ定義内の全角文字のコメントが、アセンブル・リスト・ファイル中のマクロ展開部分で表示されない制限	×	○	○
2	アセンブラの-D オプションでシンボルに数値を指定しても、マクロは真偽値としてしか利用できない制限	×	○	○
3	アセンブラ・ソース中の最終セグメントにあるBR 疑似命令を最適化した後にコード数の計算を誤り、出力されるコードが不正になる場合がある制限	×	○	○
4	構造化アセンブリ言語記述で範囲が交差する制御文がエラーとなる制限	×	×	×
5	saddr.bit の値を持つビット・シンボルを EQU 定義する際、saddr の部分に最適化の影響を受けるレーベルが記述されるとエラー等が不正になる制限	×	×	×
6	マクロ疑似命令 IRP をネストさせるとコンカティネート(&)が結合されない制限	×	×	×
7	プロジェクト・マネージャから構造化アセンブラを起動する場合のインクルード・ファイルの依存関係の制限	×	×	×
8	プロジェクト・マネージャからアセンブラを起動する場合のインクルード・ファイルの依存関係の制限	×	×	×

×:該当する

○:該当しない

-:対象外

2. 使用制限事項の詳細

No. 1 アセンブラソースのマクロ定義内の全角文字のコメントが、アセンブル・リスト・ファイル中のマクロ展開部分で表示されない制限

【内容】マクロの定義部分のコメントに全角文字を記述すると、アセンブル後に作成されるアセンブル・リスト・ファイル中の、マクロ展開された部分のコメントが、全角文字だけ表示できません。

(例)

sample.asm

```
NAME      sample
MACRO     MACRO
NOP      ; comment1 全角コメント ハンカ
ENDM
```

```
CSEG
SAMPLE1:
MAC1
END
```

sample.prn

Assemble list

ALNO	STNO	ADRS	OBJECT	M	I	SOURCE STATEMENT
1	1					NAME SAMPLE
2	2			M		MAC1 MACRO
3	3			M		NOP ; comment1 全角コメント ハンカ
4	4				M	ENDM
5	5					CSEG
6	6	----				SAMPLE1:
7	7	000000				MAC1
8	8					NOP ; comment1 ハンカ
9	10	000000	00		#1	
10	11					END

【回避策】ありません。

【改善策】Ver.1.51で修正済みです。

No. 2 アセンブラの-D オプションでシンボルに数値を指定しても、マクロは真偽値としてしか利用できない制限

【内 容】アセンブラ・ソース中で\$set/\$reset/EQU を記述する代わりに、-D オプションでシンボル値を指定しても、-D で指定したシンボル値が有効になりません。

(例1)

「-Dsyms=0 以外」でも(2)が有効になってしまうことがあります。

また、-D 指定時に\$set(syms)/\$reset(syms)の記述があった場合 A109 エラーにならない限りませんが、\$set(syms)/\$reset(syms)が有効となり-D オプションは無視されます。

```
$if (syms)
```

```
    ; (1) 「-Dsyms」または「-Dsyms=0 以外」が指定された時
```

```
$else
```

```
    ; (2) 「-Dsyms=0」が指定された時
```

```
$endif
```

(例2)

-D オプションが無視され、F303、F407 エラーになってしまいます。

また、-D 指定時に EQU の記述があった場合、A109 エラーにならない限りませんが、-D が無効のため、EQU が有効になっています。

```
$_if ( symb = 0FFH )
```

```
    ; (3) 「-Dsymb=0FFH」が指定された時
```

```
$_elseif ( symb = 01H )
```

```
    ; (4) 「-Dsymb」または「-Dsymb=1」が指定された時
```

```
$else
```

```
    ; (5) symb に(3),(4)以外の値が指定された時
```

```
$endif
```

【回避策】-D オプションを使用せず、\$set(syms)/\$reset(syms)/EQU を記述してください。

【改善策】Ver.1.51 で修正済みです。

No. 3 アセンブラ・ソース中の最終セグメントにある BR 疑似命令を最適化した後にコード数の計算を誤り、出力されるコードが不正になる場合がある制限

【内 容】アセンブラ・ソース中の最終セグメントにある BR 疑似命令を最適化した後にコード数の計算を誤り、出力されるコードが不正になる場合があります。

アセンブル・リスト・ファイル(*.prn)、リンク・リスト・ファイル(*.map)の情報は正しいですが、リロケータブル・オブジェクト・ファイル(*.rel)の情報が不正になり、結果、ロード・モジュール・ファイル(*.lmf)、HEX 形式オブジェクト・モジュール・ファイル(*.hex)の情報も不正になります。

【回避策】以下のいずれかの方法で回避して下さい。

- BR 疑似命令ではなく、命令長の明確な BR 命令を記述してください。
- コードが欠落しているセグメントの最後に、欠落しているサイズ分の NOP を挿入してください。

【改善策】Ver.1.51 で修正済みです。

No. 4 構造化アセンブリ言語記述で範囲が交差する制御文がエラーとなる制限

【内 容】制御文を#ifdef~#endif で分割または交差するように囲んだ場合、#ifdef が真の時に制御文がエラーになります。

(例)

```
switch(mode)
#ifdef stsw
case 1:
break
#endif
default:
break
ends
```

← #ifdef~#else/#endif の範囲

← case~次の case/default/ends までの範囲

【回避策】ネスト(入れ子)の場合には、正しく処理されます。制御文の範囲を交差させないようにソースを書き直して下さい。

(例)

```
#ifdef stsw
switch(mode)
case 1:
break
default:
break
ends
#else
switch(mode)
default:
break
ends
#endif
```

← #ifdef~#else/#endif の範囲

← case~次の case/default/ends までの範囲

【改善策】制限事項とします。

No. 5 saddr.bit の値を持つビット・シンボルを EQU 定義する際、saddr の部分に最適化の影響を受けるレーベルが記述されるとエラー等が不正になる制限

【内 容】saddr.bit の値を持つビット・シンボルを EQU 定義する際に、saddr の部分に最適化の影響を受けるレーベルが記述されると、不正な処理を行います。

【現 象】以下のケースで、不正な処理をします。

- saddr.bit が 0FD20H の時、レーベルがパス 1 で領域外、パス 2 で領域内となる場合、EQU 定義行に対し、パス 1 ではエラーを出力しますが、パス 2 では出力されず、不正なオブジェクトを生成します。
- saddr が 0FF1FH の時、レーベルがパス 1 で領域内、パス 2 で領域外となる場合、EQU 定義行に対し、パス 1 ではエラーになりませんが、パス 2 ではエラーを出力します。この EQU シンボルを参照した後に定義するレーベルに対し、以下のアセンブラエラーが出力されます。

「F410 Phase error」

また、このレーベルを参照するとオブジェクトが不正になります。

【回避策】ありません。

【改善策】制限事項とします。

No. 6 マクロ疑似命令 IRP をネストさせるとコンカティネート(&)が結合されない制限

【内 容】文字列の結合記号“&”を含む IRP をネストさせると、パラメータが置換されず、マクロの展開結果が不正になります。

(例)

```
IRP ZZZ,<1,2,3>
IRP XXX,<4,5,6>
    LABEL&ZZZ&XXX:    ; 展開結果が不正になります
ENDM
ENDM
```

【回避策】IRP をネストさせた場合は、“&”を使用しないで下さい。

【改善策】制限事項とします。

No. 7 PM plus から構造化アセンブラを起動する場合のインクルード・ファイルの依存関係の制限

【内 容】PM plus でのメイク・ファイル作成時において、構造化アセンブラ言語で記述されたインクルードファイルの依存関係のチェックは、コメントと文字列削除処理のみ行い、#ifdef などの条件は無視します。

【現 象】下記の例において、#ifdef、#else、#endif 行は無視されます。そのため#include に指定されたファイル名は参照の有無にかかわらず実際にファイルが存在するかのチェックが行われます。#include に指定されたファイルが存在しないとビルド時にエラーになります。

(例)

```
#ifdef SYM
#include "func1.inc"
#else
#include "func2.inc"
#endif
```

【回避策】ありません。

【改善策】制限事項とします。

No. 8 PM plus からアセンブラを起動する場合のインクルード・ファイルの依存関係の制限

【内 容】PM plus でのメイク・ファイル作成時において、アセンブラ言語で記述されたインクルードファイルの依存関係のチェックは、コメントと文字列削除処理のみ行い、\$if、\$if など条件は無視します。

【現 象】下記の例において、\$ifdef、\$else、\$endif 行は無視されます。そのため#include に指定されたファイル名は参照の有無にかかわらず実際にファイルが存在するかのチェックが行われます。\$include に指定されたファイルが存在しないとビルド時にエラーになります。

(例)

```
$if (SYM)
#include(func1.h)
$else
#include(func2.h)
$endif
```

【回避策】ありません。

【改善策】制限事項とします。

3. その他注意事項

特にありません。

以上