

マイクロコントローラ技術情報

技術通知 78K0S アセンブラ・パッケージ RA78K0S 使用制限事項の件		発行番号	ZBG-CD-07-0048号	1 / 2	
		発行日	2007年 8月 30日		
		発行部門	NEC エレクトロニクス株式会社 マイクロコンピュータ事業本部 汎用マイコンシステム事業部 開発ツールソリューショングループ		
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	RA78K0S アセンブラ・パッケージ Ver.1.40 以上 操作編		資料番号: U16656JJ1V0UM00 (第1版)		
	RA78K0S アセンブラ・パッケージ Ver.1.30 以上 言語編		資料番号: U14877JJ1V0UM00 (第1版)		
	RA78K0S アセンブラ・パッケージ 構造化アセンブリ言語編 ST78K0S V1.00 以上		資料番号: U11623JJ1V1UMJ1 (第1版)		
	PM plus Ver.5.10		資料番号: U16569JJ1V0UM00 (第1版)		
	78K0S アセンブラ・パッケージ RA78K0S V1.40 使用上の留意点		資料番号: SUD-DT-03-0408		

1. 対象製品

RA78K0S V1.31/V1.40

2. 新たな制限事項

今回新たに制限事項 No.20 ~ No.23 を追加しました。詳細は、別紙1を参照してください。

- ・No. 20 RA78K0S で -Dsymbol オプションを指定すると A106 エラーとなる制限
- ・No. 21 C ソースから外部参照した関数名の定義側が 構造化アセンブラ・ソース、あるいはアセンブラ・ソースの PUBLIC シンボルである場合に、リスト・コンバータで A105 エラーになる制限
- ・No. 22 ORG 疑似命令のセグメント名を省略した場合に、リスト・コンバータで A106 エラーになる制限
- ・No. 23 複数の C ソースを登録しアセンブラ・ソース出力オプションを指定した場合に、アブソリュート・アSEMBル・リストが最初の1つ分しか出力しない制限

3. 回避策

今回追加、変更した制限事項の回避策です。詳細は、別紙1を参照してください。

- ・No.20 CC78K0S で「アセンブラ・ソース・モジュールの出力」オプションを指定する場合には、アセンブラ・オプションの”シンボル定義”オプションを指定しないでください。
- ・No.21 リスト・コンバータを起動しないでください。
- ・No.22 ORG 疑似命令のセグメント名を記述してください。
- ・No.23 コマンド・プロンプトから LC78K0S を用いて必要なアブソリュート・アセンブル・リストを生成してください。

4. 改善計画

今回新たに追加した No.20 ~ No.23 は、次バージョンで改善することを予定しています。

RA78K0S V2.00

2007年9月リリース予定

※ 本日程については予告なく変更する場合がございますので、改善品のリリース日程については、別途、弊社営業までお問い合わせください。

5. 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。

6. 発行文書履歴

78K0S アセンブラ・パッケージ RA78K0S 使用制限事項一覧 発行文書履歴

文書番号	発行日	記事
SBG-DT-04-0006	2004.1.23	初版
SBG-DT-04-0113	2004.3.12	制限事項追加 (No.17 ~ No.19)
ZBG-CD-07-0048	2007.8.30	新規制限事項追加 (No.20 ~ No.23)

以上

RA78K0S 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		
		V1.30	V1.31	V1.40
1	インクルード・ファイルの最後に“^Z”が無い場合、正常動作しない制限 (Windows 版のみ)	○	○	○
2	構造化アセンブラをコマンド・プロンプトから起動した時、デバイス・ファイルが無いとコマンド・プロンプトがエラーを出力する制限	○	○	○
3	アセンブラ・ソースに分岐先のない BR 疑似命令がアセンブラではエラーとならずに、リンカにてエラーとなる制限	×	○	○
4	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SEMB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	×	○	○
5	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	×	○	○
6	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	×	○	○
7	アセンブラ・ソースのアブソリュートなコード・セグメント中に構文解析パスで未解決な分岐疑似命令が存在する場合後方参照シンボルに対して HIGH/LOW 演算子付きの参照を行うと不正なコードを生成する制限	×	○	○
8	ディレクティブで”MERGE AT”指定したセグメントの配置優先順位が最優先されない制限	×	○	○
9	PUBLIC/EXTRN シンボルに、コンパイラから出力される補助シンボルテーブルが存在するとディバツガ、シミュレータでエラーになる制限	×	○	○
10	セグメント名が 8 文字の時、リンカのエラーメッセージ表示が不正になる場合がある制限	×	○	○
11	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	×	×	×
12	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	×	×	×
13	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。	×	×	×
14	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	×	×	×
15	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	×	×	×
16	プロジェクト・マネージャから ID78K0S-NS(V2.xx)、SM78K0S(V2.xx)を単体で起動して作成したプロジェクトファイルを開くとエラーになる制限	×	×	○
17	サイズが 0 のセグメントの配置が不正になる制限	×	×	×
18	アセンブラのネーム定義がオブジェクト・コンバータで A106 エラーになる制限	×	×	×
19	ソース・ファイル名が 14 文字より長い場合にリンカで A402 エラーになる制限	×	×	×

×:該当する
○:該当しない
-:対象外

No.	仕様変更・追加／不具合事項	バージョン		
		V1.30	V1.31	V1.40
20	RA78K0S で -Dsymbol オプションを指定すると A106 エラーとなる制限	×	×	×
21	C ソースから外部参照した関数名の定義側が 構造化アセンブラ・ソース、あるいはアセンブラ・ソースの PUBLIC シンボルである場合に、リスト・コンバータで A105 エラーになる制限	×	×	×
22	ORG 疑似命令のセグメント名を省略した場合に、リスト・コンバータで A106 エラーになる制限	×	×	×
23	複数の C ソースを登録しアセンブラ・ソース出力オプションを指定した場合に、アブソリュート・アセンブル・リストが最初の 1 つ分しか出力しない制限	×	×	×

×:該当する

○:該当しない

-:対象外

2. 使用制限事項の詳細

No. 1 インクルード・ファイルの最後に“^Z”が無い場合、正常動作しない制限(Windows 版のみ)

【内 容】 インクルード・ファイルの最後に“^Z”が無い場合、エラーあるいは不正なデバッグ情報を入力することがあります。

【回避策】 インクルード・ファイルの最後に“^Z”のみの行を入れてください。

【改善策】 V1.30 で修正済みです。

No. 2 構造化アセンブラをコマンド・プロンプトから起動した時、デバイス・ファイルが無いとコマンド・プロンプトがエラーを出力する制限

【内 容】 構造化アセンブラをコマンド・プロンプトから起動した時、デバイス・ファイルが無いと、コマンド・プロンプトがエラーを出力するか、または反応がなくなってしまう。

【回避策】 デバイス・ファイルがインストールされているか確認してください。また、構造化アセンブラまたはデバイス・ファイルのインストール・フォルダがデフォルトでない場合、-Y オプションでデバイス・ファイルのパスを指定してください。

【改善策】 V1.30 で修正済みです。

No. 3 アセンブラ・ソースに分岐先のない BR 疑似命令がアセンブラではエラーとならずに、リンクにてエラーとなる制限

【内 容】 アセンブラ・ソースに分岐先のない BR 疑似命令が書かれた場合、アセンブラではエラーとならずに、リンクにて「F304 Operand out of range (segment 'セグメント名', address xxxxH, type 'アドレッシング・タイプ') エラーとなります。

【回避策】 プロジェクト・マネージャをご使用の場合は、アセンブラオプションの設定で、[出力2] の「クロス・リファレンス・リストの出力」をオンにしてください。

コマンドラインで起動している場合は、RA78K0S に -kx オプションを追加してください。

ビルドを実行し、リンクが終了した後、リストファイル(*.prn) の後半を見ると、クロス・リファレンス・リストができています。このリスト中で、未定義シンボルには

「***** * ***** ****」

と表示されています。

(例)

クロス・リファレンス・リストが以下のように出力されていた場合、BBBBBBBB が未定義シンボルとなります。

AAAAAAAA	FE44H	ADDR	SHORT	174#	873	1601	
BBBBBBBB	*****	* *****	****	1250			
CCCCCCCC	306H	ADDR	PROG1	1008	1032	1046	1303#

【改善策】 V1.31 で修正済みです。

No. 4 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

MAC1 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				
6	6	----			CSEG
7	7	0000			SAMPLE1:
8	8				MAC1
9	9	0000	00	#1	NOP ; comment1 ハンカ
10	10				
10	11				END

【回避策】ありません。

【改善策】V1.31 で修正済みです。

No. 5 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(sym)/\$reset(sym)の記述があった場合 A109 エラーにならないかもしれませんが、\$set(sym)/\$reset(sym)が有効となり-D オプションは無視されます。

```
$if (sym)
```

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

```
$else
```

```
    ; (2) 「-Dsym=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 エラーにならないかもしれませんが、-D が無効のため、EQU が有効になっています。

```
$_if ( symb = OFFH )
```

```
    ; (3) 「-Dsymb=OFFH」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(sym)/\$reset(sym)/EQU を記述してください。

【改善策】V1.31 で修正済みです。

No. 6 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ください。

- 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】V1.31 で修正済みです。

No. 7 アセンブラ・ソースのアブソリュートなコード・セグメント中に構文解析パスで未解決な分岐疑似命令が存在する場合後方参照シンボルに対して HIGH/LOW 演算子付きの参照を行うと不正なコードを生成する制限

【内 容】アセンブラ・ソースのアブソリュートなコード・セグメント中に、構文解析パスで未解決な分岐疑似命令が存在する場合、後方参照シンボルに対して HIGH/LOW 演算子付きの参照を行うと出力されるコードが不正となります。

【回避策】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。

【改善策】V1.31 で修正済みです。

No. 8 ディレクティブで”MERGE AT”指定したセグメントの配置優先順位が最優先されない制限

【内 容】ディレクティブで”MERGE AT”指定したセグメントは、最優先で、AT で指定されたアドレスへの配置が行われなければなりません、最優先されないために、AT 指定のないセグメントが先に配置されてしまい、F206 または F304 エラーになることがあります。

【回避策】”MERGE AT”指定されたセグメントを配置する予定の領域に、既に配置されてしまったセグメントの配置場所を移動してください。

【改善策】V1.31 で修正済みです。

No. 9 PUBLIC/EXTRN シンボルに、コンパイラから出力される補助シンボルテーブルが存在するとディバグ、シミュレータでエラーになる制限

【内 容】リンカの制限により、PUBLIC/EXTRN シンボルに、コンパイラから出力される補助シンボルテーブルが存在すると、ディバグ、シミュレータが以下のエラーメッセージを出力します。

「b019(A):ファイルのシークに失敗しました。」

【回避策】ありません。

【改善策】V1.31 で修正済みです。

No. 10 セグメント名が 8 文字の時、リンカのエラーメッセージ表示が不正になる場合がある制限

【内 容】セグメント名が 8 文字の時、リンカのエラーメッセージ中のセグメント名表示が不正になる場合があります。

【回避策】ありません。

【改善策】V1.31 で修正済みです。

No. 11 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ください。

(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. 12 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. 13 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。

【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、“&”を使用しないでください。

【改善策】制限事項とします。

No. 14 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

【内 容】プロジェクト・マネージャでのメイク・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. 15 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

【内 容】プロジェクト・マネージャでのメイク・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)

```
$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。

No. 16 プロジェクト・マネージャから ID78K0S-NS(V2.xx)、SM78K0S(V2.xx)を単体で起動して作成したプロジェクトファイルを開くとエラーになる制限

【内 容】 ID78K0S-NS(V2.xx)、SM78K0S(V2.xx)を単体で起動して作成したプロジェクト・ファイル(プロジェクト・マネージャ以外のツールが作成したプロジェクト・ファイル)を、プロジェクト・マネージャで開こうとすると、以下のエラーになります。

「F150:PRJ ファイルの読み込みに失敗しました。」

【回避策】 ID78K0S-NS(V2.xx)、SM78K0S(V2.xx)を単体で起動して作成したプロジェクト・ファイルと同名のファイルを、プロジェクト・マネージャで新規プロジェクト・ファイルとして作成してください。新規作成後、プロジェクト・マネージャの[オプション]メニューから[デバッグの選択]を選択し、使用するデバッグ(ID または SM)を指定してください。

【改善策】 V1.40 で修正済みです。

No. 17 サイズが 0 のセグメントの配置が不正になる制限

【内 容】 次の条件をすべて満たす場合に、配置が不正になる場合があります。

- (1) サイズが 0 のセグメントが存在する
- (2) (1)のセグメントがアドレス指定されていない
- (3) (1)以外のセグメントにアドレス指定されたセグメントが存在する
リンク・リストのセグメントのアドレスが、アドレス昇順に出力されていないセグメントが不正に配置されたセグメントになります。

【回避策】 不正に配置されているセグメントをディレクティブ・ファイルでアドレス指定してください。

【改善策】 V2.00 で修正する予定です。

No. 18 アセンブラのネーム定義がオブジェクト・コンバータで A106 エラーになる制限

【内 容】 次の条件をすべて満たす場合に、シンボル情報が不正になる場合があり、オブジェクト・コンバータで、A106 エラーになります。また、デバッグでダウンロードすると「Ab019:ファイルの読み込みに失敗しました」とエラーになります。

ただし、不正コードにはなりません。

- (1) 構造化アセンブラ・ソース、または、アセンブラ・ソースでネームを定義している
- (2) (1)のネームを C ソースから関数名として外部参照をしている
- (3) リンクの順番が(2)の C ソースの後に(1)のファイルである

【回避策】 C ソースを、構造化アセンブラ・ソースやアセンブラ・ソースより後ろにリンクするようにしてください。

【改善策】 V2.00 で修正する予定です。

No. 19 ソース・ファイル名が 14 文字より長い場合にリンクで A402 エラーになる制限

【内 容】 次の条件をすべて満たす場合に、リンクにて A402 エラーになります。

- (1) ソース・ファイル名が 14 文字より長い
- (2) アセンブラでデバッグ情報を出力していない
デバッグ情報を出力しない指定は、次のいずれかの場合です。
 - PM または PM plus にて「Release Build」を選択する
 - アセンブラにて“-nga -ng”オプションを指定する

【回避策】 以下のいずれかの方法で回避してください。

- (1) ソース・ファイル名を 14 文字までに変更する。
- (2) PM または PM plus にて「Debug Build」を選択する、またはアセンブラにて“-ga -g”オプションを指定(デフォルトで指定)して、使用する。
アセンブラのデバッグ情報は、生成される HEX ファイルには影響を与えません。

【改善策】 V2.00 で修正する予定です。

No. 20 RA78K0S で -Dsymbol オプションを指定すると A106 エラーとなる制限

【内 容】 CC78K0S からデバッグ情報を付加したアセンブラ・ソース・ファイルに対し、RA78K0S で -Dsymbol オプションの symbol を指定すると、不正なシンボル情報(symbol 名が参照できない)を出力するため、CC78K0S でシンボルテーブルに対する A106 エラーになります。

【回避策】 CC78K0S で「アセンブラ・ソース・モジュールの出力」オプションを指定する場合には、アセンブラ・オプションの”シンボル定義”オプション指定をしないでください。
アセンブラ・ソース・ファイルに対して、個別アセンブラ・オプションの設定から”シンボル定義”オプション指定をしてください。

【改善策】 V2.00 で修正する予定です。

No. 21 C ソースから外部参照した関数名の定義側が 構造化アセンブラ・ソース、あるいはアセンブラ・ソースの PUBLIC シンボルである場合に、リスト・コンバータにて A105 エラーになる制限

【内 容】 C ソースから外部参照した関数名の定義側が 構造化アセンブラ・ソース、あるいはアセンブラ・ソースの PUBLIC シンボルである場合に、リスト・コンバータにて A105 エラーになります。

【回避策】 リスト・コンバータを起動しないでください。

【改善策】 V2.00 で修正する予定です。

No. 22 ORG 疑似命令のセグメント名を省略した場合に、リスト・コンバータで A106 エラーになる制限

【内 容】 ORG 疑似命令のセグメント名を省略した場合に、リスト・コンバータで A106 エラーになります。

【例】

```
-----アセンブラ・ソース-----  
          ORG 1000H  
          MOV  A, SADR2  
          MOVW AX, SADR3  
          :
```

【回避策】 ORG 疑似命令のセグメント名を記述してください。

【例】

```
-----アセンブラ・ソース-----  
MAIN1  ORG 1000H  
          MOV  A, SADR2  
          MOVW AX, SADR3  
          :
```

【改善策】 V2.00 で修正する予定です

No. 23 複数の C ソースを登録しアセンブラ・ソース出力オプションを指定した場合に、アブソリュート・アセンブル・リストが最初の 1 つ分しか出力しない制限

【内 容】 複数の C ソースを登録しアセンブラ・ソース出力オプションを指定した場合に、アブソリュート・アセンブル・リストが最初の 1 つ分しか出力しません。

【回避策】 コマンド・プロンプトから LC78K0S を用いて必要なアブソリュート・アセンブル・リストを生成してください

【改善策】 V2.00 で修正する予定です。

3. その他注意事項

特にありません。

以上