

この度は、統合開発環境 CubeSuite+をご使用いただきまして、誠にありがとうございます。

この添付資料では、本製品をお使いいただく上での制限事項および注意事項等を記載しております。ご使用の前に、必ずお読みくださいますようお願い申し上げます。

## 目次

第 1 章	対象デバイスについて .....	2
第 2 章	ユーザーズ・マニュアルについて.....	3
第 3 章	アンインストール時の選択キーワード.....	4
第 4 章	変更点 .....	5
4.1	CA78K0 の変更点 .....	5
4.1.1	翻訳限界値の改善 .....	5
4.1.2	#asm文の改善 .....	5
4.1.3	制限事項の解除 .....	5
第 5 章	注意事項.....	6
5.1	メモリ・バンク配置支援ツールに関する注意事項 .....	6
第 6 章	制限事項.....	7
6.1	CA78K0 の制限事項 .....	7
6.1.1	CA78K0 の制限事項一覧.....	7
6.1.2	CA78K0 の制限事項詳細.....	8
6.2	メモリ・バンク配置支援ツールの制限事項.....	14
6.2.1	メモリ・バンク配置支援ツールの制限事項一覧.....	14
6.2.2	メモリ・バンク配置支援ツールの制限事項詳細.....	15
第 7 章	ドキュメント訂正 .....	16

## 第1章 対象デバイスについて

CA78K0 がサポートする対象デバイスに関しては、WEB サイトに掲載しています。

こちらをご覧ください。

CubeSuite+製品ページ：

<http://japan.renesas.com/cubesuite+>

## 第2章 ユーザーズ・マニュアルについて

本製品に対応したユーザーズ・マニュアルは、次のようになります。本文書と合わせてお読みください。

マニュアル名	資料番号
CubeSuite+ V1.03.00 78K0 コーディング編	R20UT2141JJ0100
CubeSuite+ V1.01.00 78K0 ビルド編	R20UT0783JJ0100
CubeSuite+ V1.03.00 メッセージ編	R20UT2147JJ0100

## 第3章 アンインストール時の選択キーワード

本製品をアンインストールする場合は、2つの方法があります。

- ・ 統合アンインストーラを使用する(CubeSuite+自体をアンインストールする)
- ・ 個別にアンインストールする(本製品のみをアンインストールする)

個別にアンインストールを行なう場合、コントロールパネルの

- ・ 「プログラムの追加と削除」(WindowsXP の場合)
- ・ 「プログラムと機能」(Windows Vista, Windows 7 の場合)

から、「CubeSuite+ CA78K0 V1.30」を選択してください。

## 第4章 変更点

本章では、CA78K0 コンパイラの変更点について説明します。

なお、以下の変更によってバージョンアップでコードが変わる可能性があります。

### 4.1 CA78K0の変更点

CA78K0 V1.21 から V1.30 への変更点について説明します。

#### 4.1.1 翻訳限界値の改善

- switch 文の case ラベル数 257 → 1024
- 1 つの論理ソース行の文字数 2048 → 32767
- 1 つの翻訳単位内で同時に定義可能なマクロ識別子の数 32767 → 60000
- 1 つの構造体、または共用体のメンバ数 256 → 1024
- マクロのネスト 200 → 10000
- カッコで囲まれた式の入れ子のレベル数 32 → 1024

#### 4.1.2 #asm文の改善

以下の#asm 文に関して改善しました。

- (1) C ソースにて「#asm」によりコンパイラ管理外のシンボルの外部定義/外部参照宣言を行い、
- (2) その外部定義/外部参照宣言したシンボル名長が 9 文字以上の場合に、リンク時に E3405 エラーになります。

#### 4.1.3 制限事項の解除

- 以下の制限事項を解除しました。

##### No.77 条件演算子に関する制限

【内容】 条件演算子の第 2 および第 3 オペランドが boolean 型の時にコード不正となります。

(例)

```
__boolean b1;
unsigned char uc1;
void func()
{
    b1 = (uc1 & 0x80) ? (__boolean)1 : (__boolean)0
}
```

## 第5章 注意事項

本章では、CA78K0 V1.30 の注意事項について説明します。

### 5.1 メモリ・バンク配置支援ツールに関する注意事項

#### メモリ・バンク配置支援ツールの起動でエラーとなる注意事項

【内容】再リンク機能で、フラッシュ領域に対してメモリ・バンク配置支援ツールを起動するとエラーになる場合があります。

## 第6章 制限事項

本章では、CA78K0 の制限事項について説明します。

### 6.1 CA78K0の制限事項

CA78K0 V1.30 における制限事項一覧を以下に示します。

#### 6.1.1 CA78K0の制限事項一覧

##### (1) アセンブラ部の制限事項一覧

No.	制限事項
1	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 (PM+版のみ)
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

##### (2) コンパイラ部の制限事項一覧

No.	制限事項
16	signed 型のビット・フィールドを符号無しのビット・フィールドとして処理する制限
43	入出力関数の標準ライブラリの出力変換で、動作が不正となる制限
44	int/short 型の最小値-32768 のサイズが 4 となる制限
47	関数定義の識別子の型に対して、仮引数型と関数定義の識別子の型とが合わずに、エラーとなる制限
48	関数定義の識別子並びで、宣言していない仮引数を int 型とせずエラーとなる制限
49	#演算子が正しく展開できない制限

## 6.1.2 CA78K0の制限事項詳細

CA78K0 V1.30 における制限事項の詳細は以下のとおりです。

## (1) アセンブラ部の制限事項詳細

## No. 1 構造化アセンブラ言語記述で範囲が交差する制御文がエラーとなる制限 (PM+版のみ)

【内容】 制御文を#ifdef~#endif 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



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。

以下のケースで、不正な処理をします。

(1) **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。

(2) **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。

## (2) コンパイラ部の制限事項詳細

No. 16 signed 型のビット・フィールドを符号無しビット・フィールドとして処理する制限

【内容】 signed 型のビット・フィールドを、符号無しビット・フィールドとして処理します。

【回避策】 ありません。

No. 43 入出力関数の標準ライブラリの出力変換で、動作が不正となる制限

【内容】 関数 printf, sprintf, vprintf, vsprintf の出力変換で、以下のような場合に動作が不正となります。

変換指定子 "d, i, o, u, x および X" に対して、".2" のように精度を指定した場合に、0 フラグを無視しません。

(例)

```
#include <stdio.h>

void func()
{
    printf("%04.2d\n", 77);
}
```

※(不正な動作) "0077" となります

(正しい動作) " 77" となります

変換指定子 "g,G" に対して、指定した精度+1 を精度とします。

(例)

```
#include <stdio.h>

void func()
{
    printf("%.2g", 12.3456789);
}
```

※ (不正な動作) "12.3" となります

(正しい動作) "12" となります

【回避策】 ありません。

No. 44 int/short 型の最小値-32768 のサイズが 4 となる制限

【内 容】 int/short 型の最小値-32768 のサイズが 4 となります。

```
(例)
int x;
void func()
{
    x = sizeof(-32768);
}
```

※(不正な動作) x の値が 4 となります

(正しい動作) x の値が 2 となります

【回避策】 (-32767-1) と記述してください。

No. 47 関数定義の識別子の型に対して、仮引数型と関数定義の識別子の型と合わずに、エラーとなる制限

【内 容】 関数定義の識別子の型に対して実引数拡張をしていないため、仮引数型と関数定義の識別子の型と適合せずに、E0747 エラーとなります。

```
(例)
int fn_char(int);
int fn_char(c)
char c;
{
    return 98;
}
```

【回避策】 仮引数型と関数定義の識別子の型を合わせてください。

**No. 48** 関数定義の識別子並びで、宣言していない仮引数を int 型とせずエラーとなる制限

【内 容】 関数定義の識別子並びで、宣言していない仮引数を int 型とせず E0706 エラーとなります。

(例)

```
void func(x1, x2, f, x3, lp, fp)

int (*fp)( );

long *lp;

float f;

{
    :
}
```

【回避策】 関数定義の仮引数はすべて宣言してください。

## No. 49 #演算子が正しく展開できない制限

【内 容】 以下のいずれかの条件で正しく展開できません。

(条件1) #演算子で「'''」を正しく展開できずに、コンパイル時にエラーとなります。

(条件1の例)

```
#include <string.h>
#define str( a) (# a)
int x;
void func()
{
    if (strcmp(str(''), "¥''") == 0) x++;
}
```

※(不正な動作) コンパイル時にエラーとなります

(正しい動作) if (strcmp( ("¥''"), "¥''") == 0) x++; となります

(条件2) #演算子とネストを含むマクロを正しく展開できません。

(条件2の例)

```
#define str(a) #a
#define xstr(a) str(a)
#define EXP 1
char *p;
void func()
{
    p = xstr(12EXP);
}
```

※(不正な動作) “p = ("12E1");”となります。

(正しい動作) “p = ("12EEXP");”となります。

【回避策】 ありません。

## 6.2 メモリ・バンク配置支援ツールの制限事項

メモリ・バンク配置支援ツールにおける制限事項一覧を以下に示します。

### 6.2.1 メモリ・バンク配置支援ツールの制限事項一覧

No.	制限事項
1	__flashf 関数に関する制限
2	オプション指定に関する制限
3	リンク・ディレクティブに関する制限
4	callt 領域に関する制限

## 6.2.2 メモリ・バンク配置支援ツールの制限事項詳細

メモリ・バンク配置支援ツールの制限事項の詳細は以下の通りです。

### No. 1 `__flashf` 関数に関する制限

【内容】 `__flashf` 関数は共通領域に配置しなければなりません。メモリ・バンク配置支援ツールは `__flashf` 関数を判別できないため、`__flashf` 関数が共通領域以外に配置され、ビルド時にエラーとなる可能性があります。

【回避策】 `__flashf` 関数を含むファイルは、C ソースファイルのプロパティを開き、[ビルド設定]タブ の [メモリ・バンク]カテゴリの [配置する領域] で、"共通領域"を指定してください。

### No. 2 オプション指定に関する制限

【内容】 メモリ・バンク配置支援ツールはコンパイル・オプション、アセンブル・オプション、リンク・オプションの「その他の追加オプション」で指定できるオプションを解析していません。

【回避策】 ビルド・ツール・プロパティで指定できるオプションは、「その他の追加オプション」の欄に記述しないようにしてください。

### No. 3 リンク・ディレクティブに関する制限

【内容】 リンク・ディレクティブ・ファイルには対応していません。メモリ・バンク配置支援ツールは、デバイス・ファイルで定義されているメモリ領域にもとづいて配置を行います。（例えば、`const` データや初期値データなど関数以外をバンク領域に配置した場合にも対応していません。）

【回避策】 [メモリ・バンク配置オプション]タブ の [マージン]カテゴリで、各領域の空き領域を調整してください。

### No. 4 `callt` 領域に関する制限

【内容】 メモリ・バンク配置支援ツールでは、`#pragma section` 指定、または疑似命令の発行により、`@@CALT` のセクション名が変更された場合、参照情報ファイルに出力する“`callt` 領域を利用して呼び出された関数の参照回数”が不正値となります。

【回避策】 `@@CALT` のセクション名を変更しないでください。

## 第7章 ドキュメント訂正

なし

すべての商標および登録商標は、それぞれの所有者に帰属します。



## ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。  
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、  
家電、工作機械、パーソナル機器、産業用ロボット等  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、  
防災・防犯装置、各種安全装置等  
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っていません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事情報に使用しないてください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/contact/>