

# 変数群の同アドレスへの配置方法

RL78用 Cコンパイラ CA78K0R

株式会社ルネサス ソリューションズ  
ツールビジネス本部 ツール技術部

2014/6/20 Rev. 1.00

R20UT3044JJ0100

- **変数を同じアドレスに配置するには**
  
- **変数のアドレスへの配置イメージ**
  
- **変数群を同じアドレスに配置するには**
  - **Cソース上での変数のセクション名の変更**
    - Cコンパイラで出力されるセクション名と属性
    - Cソース上でのセクション指定の注意
  - **リンク・ディレクティブ・ファイルでセクションの配置を指定**
    - リンク・ディレクティブの記述形式
  - **リンク・ディレクティブ・ファイルのCubeSuite+への登録**

# 変数を同じアドレスに配置するには

## ■ 数個の変数の場合

- 数バイトの変数であれば、共用体と構造体を使用することにより、同じアドレスに違う名前を指定することが可能です。
  - データ転送する変数群にて、転送する際にアクセスする形と、使用する際にアクセスする形が違う場合など

## ■ 多数の変数の場合

- 数Kバイトの変数群の場合には、共用体、構造体を使用することは、現実的に難しくなります。
  - お客様のシステム上で、モードにより処理が異なり、その際にモードごとに使用する変数を別のアドレスに配置するのではなく、共通のアドレスに配置する場合など

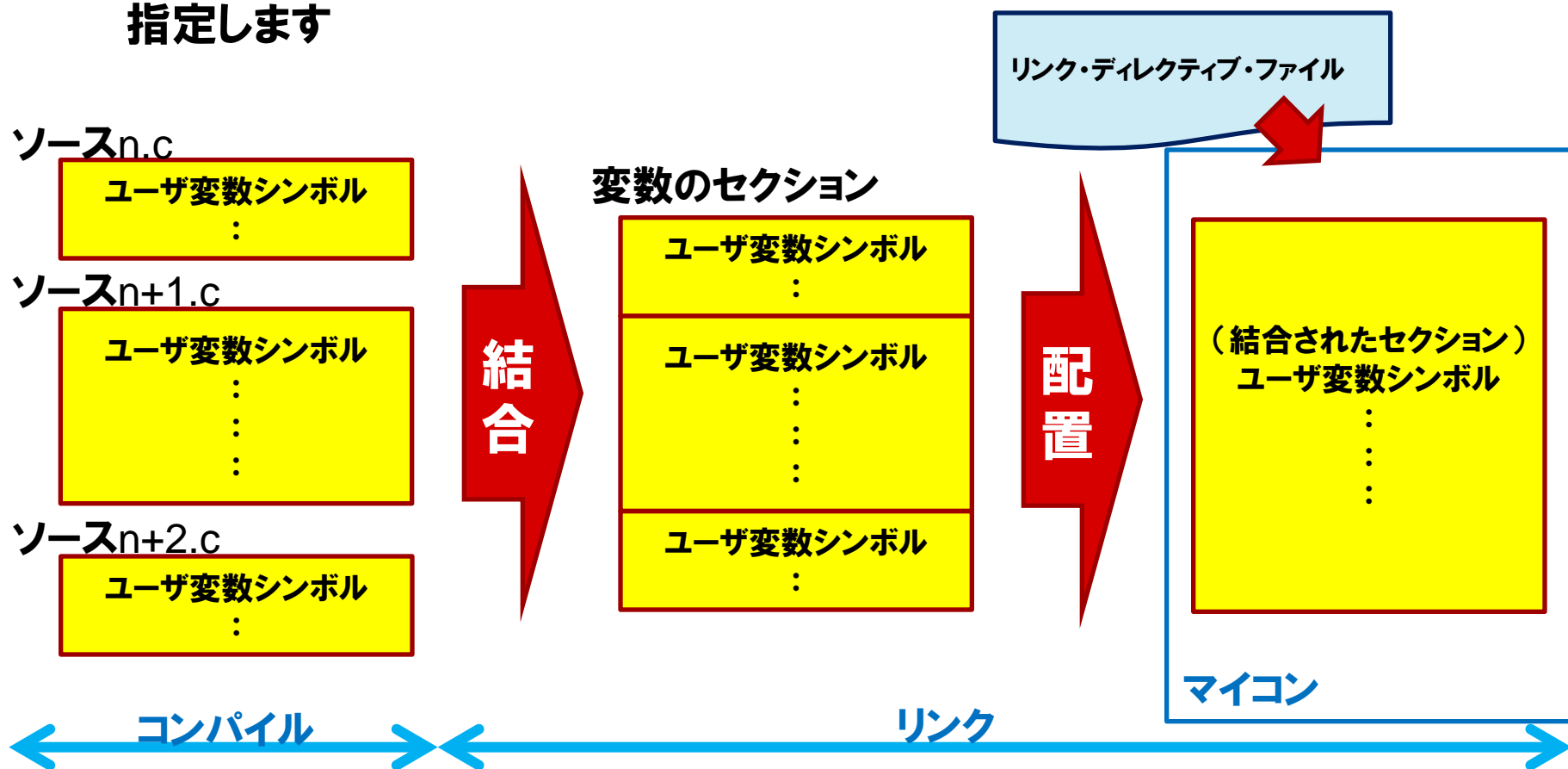
## ■ 本資料では、「多数の変数の場合」についての説明を行います

## ■ セクション、セグメントの表現の注意

- CA78K0Rでは、C言語ではセクションと呼びます。ただし、セクション＝セグメントで同じものを示しています。

# 変数のアドレスへの配置イメージ

- 変数は、コンパイル時にセクションとして出力されます
- リンク時に、同じ属性の同じ名前のセクションを結合し、メモリに配置します
- メモリの配置方法は、リンク時に入力するリンク・ディレクティブ・ファイルで指定します



# 変数群を同じアドレスに配置するには

- Cソース上で変数のセクション名を変更
  - リンク時に配置を指定するためにセクション名を変更する
  - #pragma section を使用する
  - 注意:初期値あり変数は使用しないでください
- リンク・ディレクティブ・ファイルでセクションの配置を指定
  - 同じアドレスにセクションを配置するために指定する
  - リンク・ディレクティブのメモリ空間の機能を使用する
- リンク・ディレクティブ・ファイルをCubeSuite+のプロジェクトに登録する
  - リンク時にセクション配置等を指示するために指定する

# Cソース上での変数のセクション名の変更

## ■ #pragma section を使用

- デフォルトで出力されるセクションの名前を変更する

- 記述形式

- #pragma section **コンパイラ出力セクション名** **変更セクション名**
- **コンパイラ出力セクション名**は次ページ参照

(例)

```
#pragma section @@DATA UDATA
```

```
int a;  
int b;
```

```
#pragma section @@DATA @@DATA
```

**ユーザセクション名に変更**

**デフォルトのセクション名に戻す**

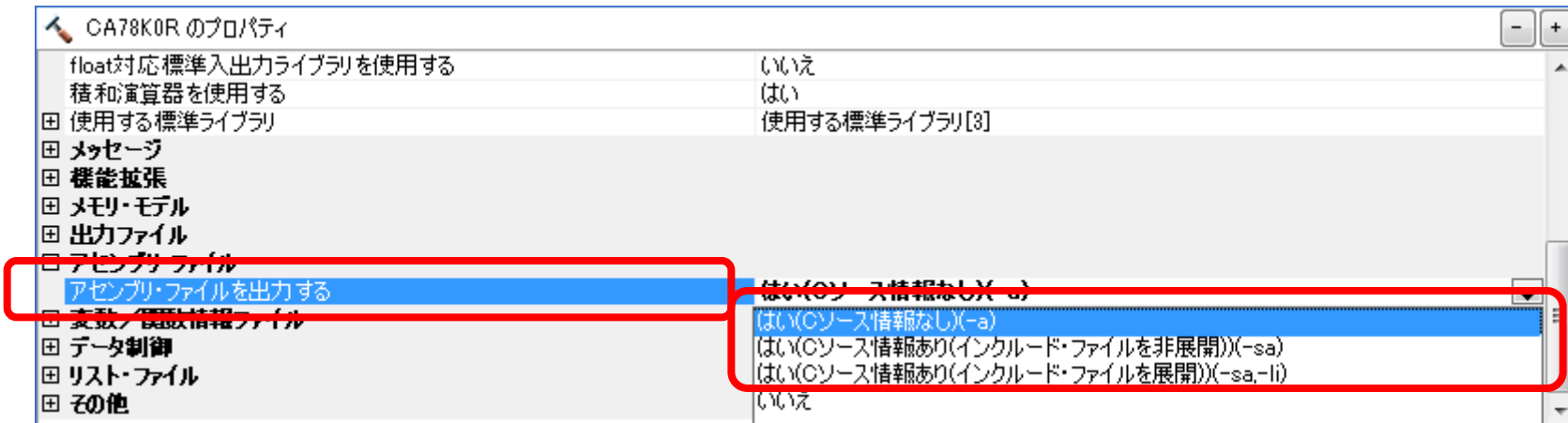
# Cコンパイラで出力されるセクション名と属性

## ■ CA78K0Rでデフォルトで出力されるデータセグメント(セクション)名と属性

セクション名	セグメント・タイプ	再配置属性	説明
@@INIT	DSEG	BASEP	データ領域用セグメント(初期値あり, near領域配置)
@@INITL	DSEG	UNIT64KP	データ領域用セグメント(初期値あり, far領域配置)
@@DATA	DSEG	BASEP	データ領域用セグメント(初期値なし, near領域配置)
@@DATAL	DSEG	UNIT64KP	データ領域用セグメント(初期値なし, far領域配置)
@@INIS	DSEG	SADDRP	データ領域用セグメント(初期値ありsreg変数)
@@DATS	DSEG	SADDRP	データ領域用セグメント(初期値なしsreg変数)
@@BITS	DSEG		boolean型変数, bit型変数用セグメント

## Cソース上でのセクション指定の注意

- コンパイラで直接オブジェクト(\*.rel)を生成できない場合には、次のワーニングが出力されます。
  - CC78K0R warning W0837: Output assembler source file , not object file
- コンパイラのオプションで、アセンブリ・ファイルを出力するように指定してください。
  - CubeSuite+ の CA78K0R(ビルド・ツール)のプロパティ
    - [コンパイル・オプション]タブの「アセンブリ・ファイル」
    - 「アセンブリ・ファイルを出力」で「はい」を含むものを選択





# リンク・ディレクティブ・ファイルでセクションの配置を指定(1/2)

- 通常のメモリ空間(REGULAR)ではない、メモリ空間(EX1、...、EX15)にメモリ領域を作成し、そのメモリ領域に、セクションを配置する

(例)

- メモリ領域の一つ目
  - 名前ROM、開始アドレス 0x0. サイズ 0x10000、REGULAR空間 (デフォルト)
- メモリ領域の二つ目
  - 名前RAM、開始アドレス 0xfef00. サイズ 0x1100、REGULAR空間 (デフォルト)
- メモリ領域の三つ目
  - 名前RAM2、開始アドレス 0xfef00. サイズ) 0xc00、EX1空間

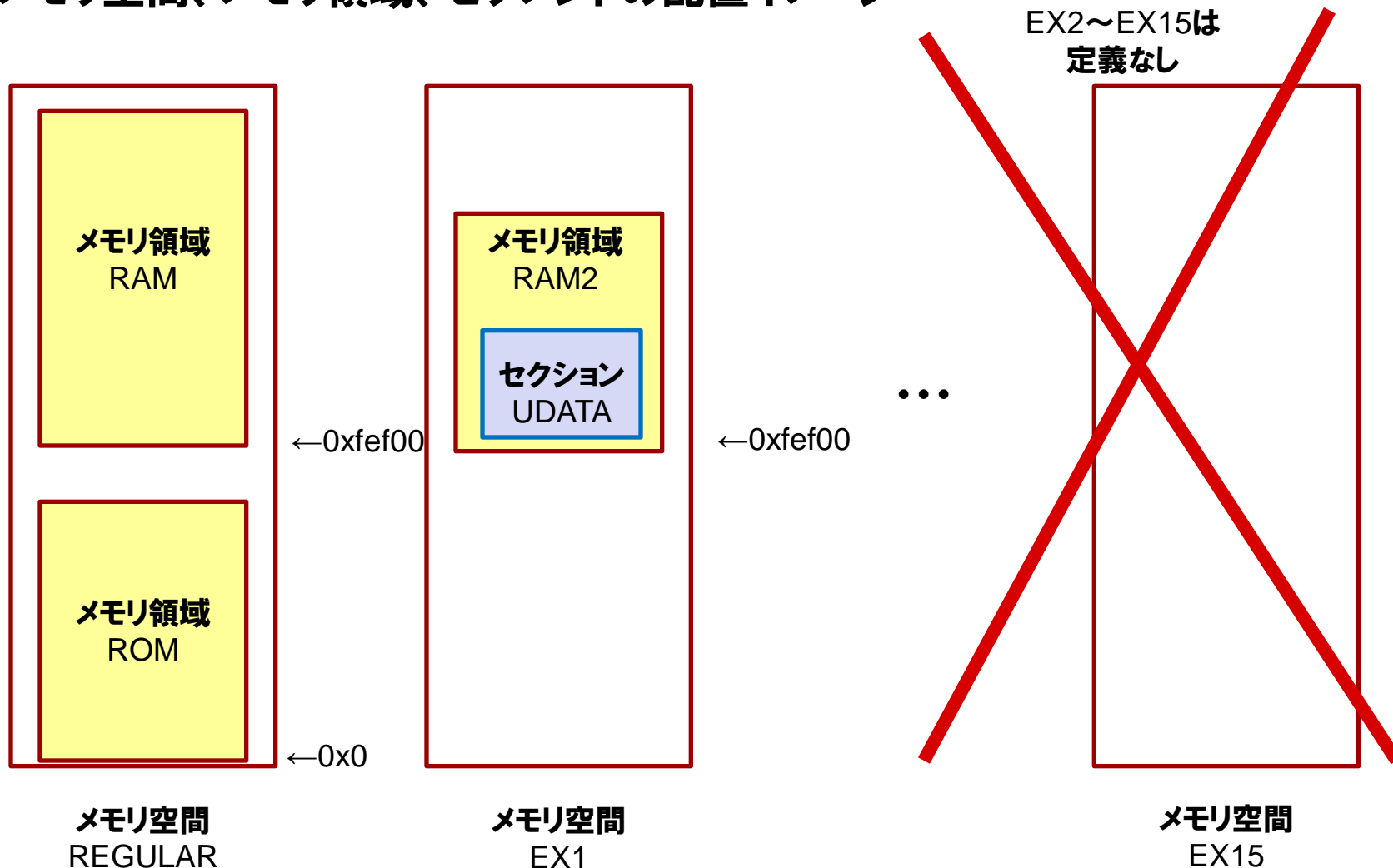
```
MEMORY ROM : (00000H , 010000H )  
MEMORY RAM : (0FEF00H , 01100H )  
MEMORY RAM2 : (0FEF00H , 00C00H ) /EX1  
  
MERGE UDATA := RAM2 / EX1
```

EX1にメモリ領域定義

EX1のメモリ領域に  
セクションを配置

# リンク・ディレクティブ・ファイルでセクションの配置を指定(2/2)

## ■ メモリ空間、メモリ領域、セグメントの配置イメージ



# リンク・ディレクティブの記述形式(1/2)

## ■ メモリ・ディレクティブ

- **MEMORY** **メモリ領域名** : ( **スタート・アドレス** , **サイズ** ) [ / **メモリ空間名** ]

- **メモリ領域名**

- 作成する**メモリ領域**の名前を指定します

- **スタート・アドレス**

- 作成する**メモリ領域**の開始アドレスを指定します

- **サイズ**

- 作成する**メモリ領域**のサイズを指定します

- 終了アドレスではないので注意して下さい

- **メモリ空間名**

- 作成する**メモリ領域**の配置する**メモリ空間**を指定します

- REGULAR, EX1, ..., EX15

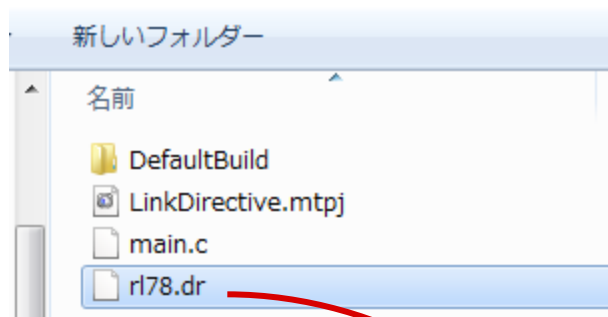
# リンク・ディレクティブの記述形式(2/2)

## ■ セグメント配置ディレクティブ

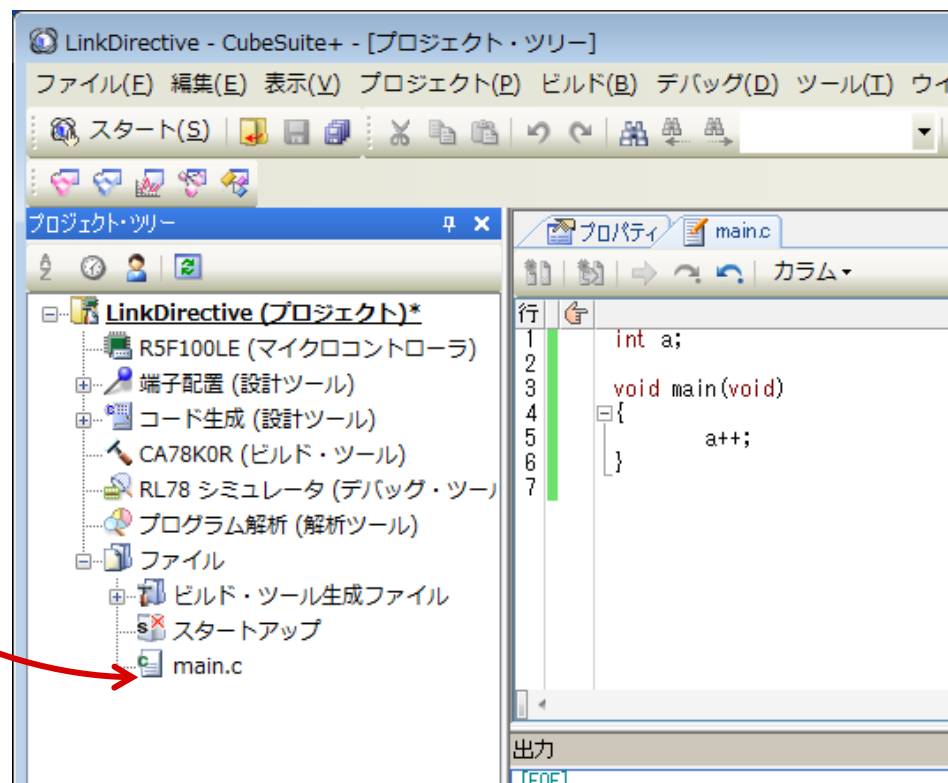
- **MERGE** セグメント名 : [AT ( スタート・アドレス )][ = メモリ領域名 ] [ / メモリ空間名 ]
- **MERGE** セグメント名 : [結合属性][ = メモリ領域名 ][ / メモリ空間名 ]
  - セグメント名
    - 配置を指定する**セグメント**の名前を指定します
  - スタート・アドレス
    - 配置を指定する**セグメント**の開始アドレスを指定します(省略可)
  - メモリ領域名
    - 配置を指定する**セグメント**の格納する**メモリ領域**を指定します(省略可)
  - メモリ空間名
    - 配置を指定する**セグメント**の**メモリ空間**を指定します
      - REGULAR, EX1, ..., EX15
  - 結合属性
    - 配置を指定する**セグメント**の結合属性を指定します
      - SEQUENT(デフォルト)  
セグメントを出現順に、順次空きを作らないようにマージします。BSEGはビット単位で出現順にマージします。
      - COMPLETE
      - 同名のセグメントが複数存在する場合はエラーとします

# リンク・ディレクティブ・ファイルのCubeSuite+への登録

- CubeSuite+ のプロジェクトツリーパネルに、リンク・ディレクティブ・ファイルを、ドラッグ&ドロップして登録する



**登録されると  
リストに表示されます**



**RENEASAS**

**ルネサス ソリューションズ株式会社**

© 2014 Renesas Solutions Corp.