

この度は、データ・フラッシュ・ライブラリ Type04 パッケージ Ver.2.00 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、データ・フラッシュ・ライブラリ Type04 パッケージ Ver.2.00 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	対象製品	2
第2章	ユーザーズマニュアル	2
第3章	バージョンアップ内容	2
第4章	対応ツール	3
第5章	インストール	3
5.1	インストール方法	3
5.2	アンインストール方法	3
5.3	ファイル構成	4
第6章	ビルド方法	5
6.1	使用するソフトウェア	5
6.2	CS+(旧CubeSuite+)でのビルド方法	5
6.2.1	C言語の場合	5
6.2.2	アセンブリ言語の場合	8
6.3	ビルド時の注意事項	11
6.3.1	CA78K0Rコンパイラ使用時	11
6.3.2	CC-RLコンパイラ使用時	12
第7章	デバッグ方法	13
7.1	デバッグ時の注意事項	13
第8章	サンプルプログラム	14
8.1	サンプルプログラムの初期設定	14
8.2	オプション・バイトとオンチップ・デバッグの設定について	15
8.3	C言語用サンプルプログラムのコンパイルスイッチ	16
8.4	内蔵RAM領域の定義	17
8.4.1	CA78K0Rコンパイラ使用時	17
8.4.2	CC-RLコンパイラ使用時	18

## 第1章 対象製品

データ・フラッシュ・ライブラリ Type04 パッケージ Ver.2.00 では、「RL78 ファミリ CA78K0R コンパイラ用データ・フラッシュ・ライブラリ Type04 Ver.1.05B (JP\_R\_FDL\_RL78\_T04\_V1.05\_B\_J.zip)」と「RL78 ファミリ CC-RL コンパイラ用データ・フラッシュ・ライブラリ Type04 Ver.1.05A (JP\_R\_FDL\_RL78\_T04\_V1.05\_CCRL\_A\_J.zip)」を統合し、インストーラを使用してインストールする方法に変更になりました。

本リリースノートの対象製品を示します。

製品名	Ver.	インストーラファイル名	Ver.
RL78 ファミリ CA78K0R コンパイラ用 データ・フラッシュ・ライブラリ Type04	V1.05	RENESAS_RL78_FDL_T04_2V00.exe	V2.00
RL78 ファミリ CC-RL コンパイラ用 データ・フラッシュ・ライブラリ Type04	V1.05		

## 第2章 ユーザーズマニュアル

本バージョンは下記のユーザーズマニュアルに対応しています。

マニュアル名	資料番号
RL78 ファミリ データ・フラッシュ・ライブラリ Type04 ユーザーズマニュアル	R01US0049JJ0106

## 第3章 バージョンアップ内容

本バージョンのバージョンアップ内容を示します。

No.	パッケージ Ver.	対象	内容
1	V2.00	CA78K0R コンパイラ用 ライブラリ V1.05	ZIP Ver.( JP_R_FDL_RL78_T04_V1.05_B_J.zip)からライブラリ本体の変更はありません。
		CC-RL コンパイラ用 ライブラリ V1.05	ZIP Ver.( JP_R_FDL_RL78_T04_V1.05_CCRL_A_J.zip)からライブラリ本体の変更はありません。
		ユーザーズマニュアル	Rev1.05 から Rev1.06 に改版 詳細な改版内容につきましては、ユーザーズマニュアルの改版履歴をご参照ください。

## 第4章 対応ツール

データ・フラッシュ・ライブラリ Type04 Ver.1.05 を使用する場合には、下記のバージョンを使用してください。

対象ライブラリ	ツール名	バージョン
CA78K0R コンパイラ用 ライブラリ	統合開発環境 : CubeSuite+	V1.00.00 以降
	統合開発環境 : CS+	V3.00.00 以降
CC-RL コンパイラ用 ライブラリ	統合開発環境 : CS+	V3.01.00 以降

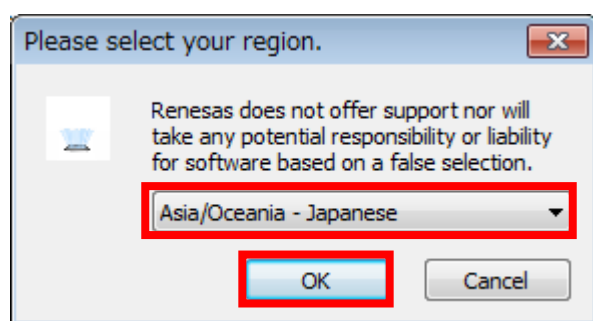
## 第5章 インストール

この章では、データ・フラッシュ・ライブラリ Type04 パッケージ Ver.2.00 のインストールとアンインストールの手順について説明します。

### 5.1 インストール方法

データ・フラッシュ・ライブラリ Type04のインストールは次の手順で行います。

- (1) Windowsを起動します。
- (2) データ・フラッシュ・ライブラリ Type04 パッケージの圧縮ファイルを解凍し、インストーラを実行します。
- (3) プルダウンメニューから"Asia/Oceania – Japanese" を選択します。
- (4) "OK"ボタンを押下し、以降、インストーラの指示に従い、実行します。



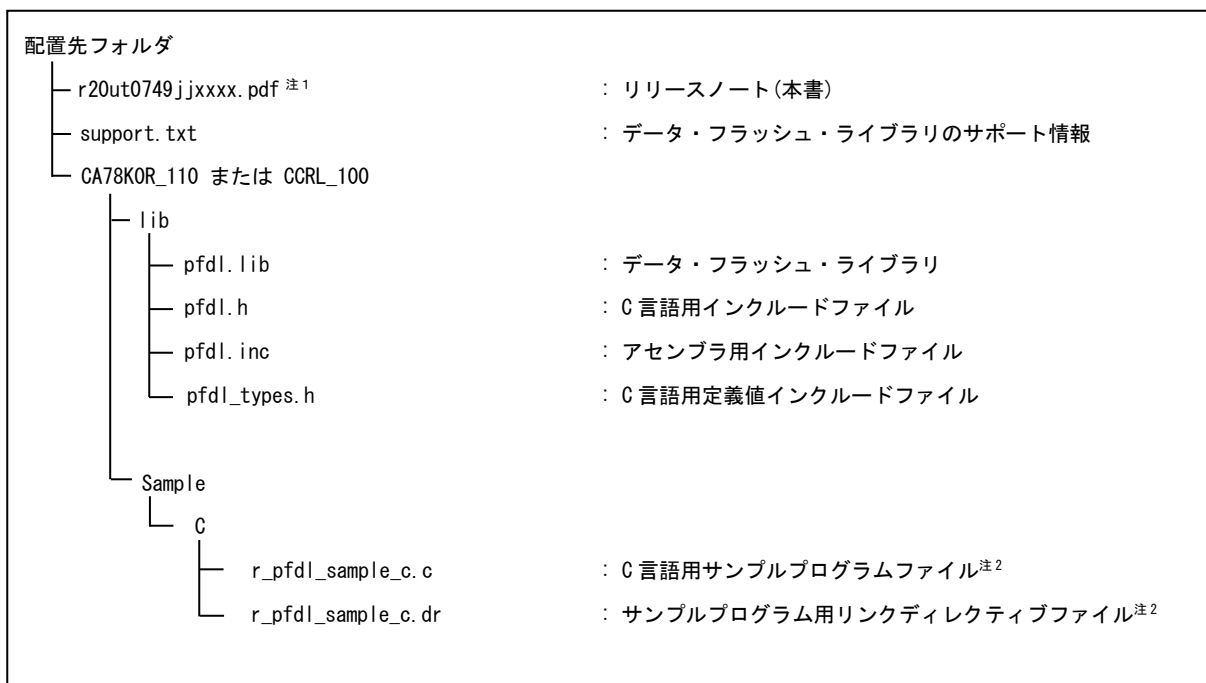
### 5.2 アンインストール方法

データ・フラッシュ・ライブラリ Type04のアンインストールは次の手順で行います。

- (1) Windowsを起動します。
- (2) ユーザ任意の場所に配置したデータ・フラッシュ・ライブラリ Type04が入っているフォルダを削除します。

## 5.3 ファイル構成

データ・フラッシュ・ライブラリ Type04パッケージのインストーラを実行することによって、展開されるファイルの構成は、次のとおりです。



注1. xはバージョン番号、またはRev番号の為、省略しています。

注2. CA78K0R用でサンプルプログラムを使用する場合は、プログラムファイル(\*.c)とリンクディレクティブファイル(\*.dr)と一緒に組み込んでください。

CC-RL用では、リンクディレクティブファイル(\*.dr)[リンク情報の設定ファイル]は添付されていません。

CC-RL用サンプルプログラムのリンク情報は、CS+のリンクの設定画面で指定してください。

## 第6章 ビルド方法

この章では、データ・フラッシュ・ライブラリ Type04を用いたプログラムのビルドの手順を説明します。

### 6.1 使用するソフトウェア

データ・フラッシュ・ライブラリ Type04 を用いたプログラムをビルドするには、次の統合開発環境が必要です。

- ・ CA78K0R コンパイラ用：統合開発環境 CS+ V3.00.00 以降、または CubeSuite+ V1.00.00 以降
- ・ CC-RL コンパイラ用：統合開発環境 CS+ V3.01.00 以降

### 6.2 CS+(旧CubeSuite+)でのビルド方法

CS+を用いてデータ・フラッシュ・ライブラリ Type04 をユーザプログラムに組み込んで、ビルドする手順を説明します。

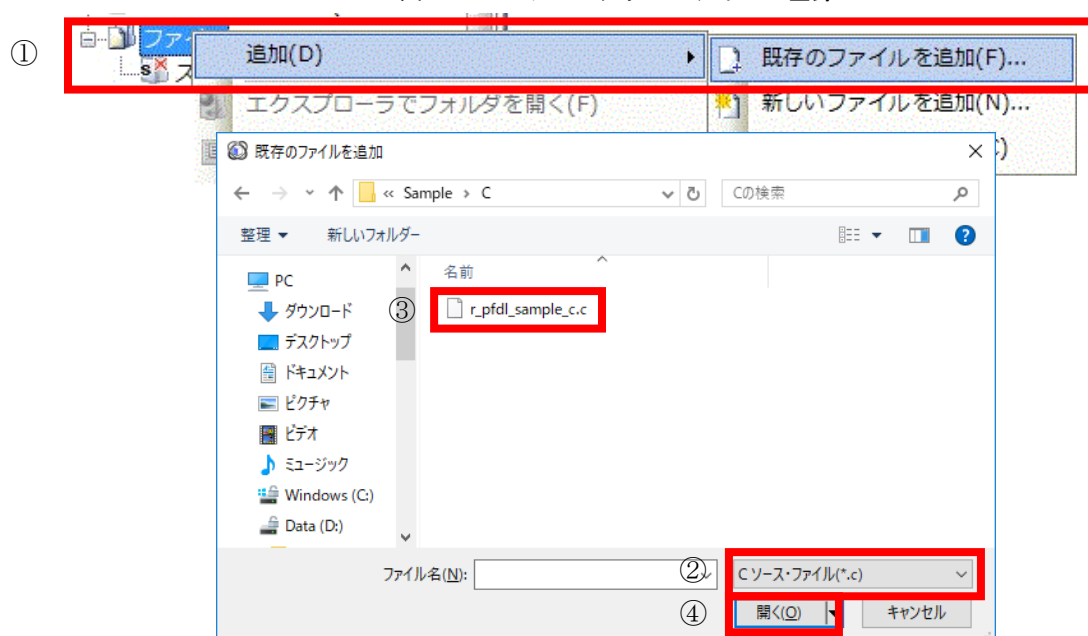
#### 6.2.1 C言語の場合

##### (1) プロジェクトの作成とソースファイルの設定

CS+でプロジェクトを作成し、表示された画面の左側にある[ プロジェクト・ツリー ]から、[ ファイル ]を右クリックして表示されるリストの[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-1）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ C ソースファイル(\*.c) ]を選択し、ソースファイルとしてユーザプログラムのファイルを登録してください。

図 6-1 ユーザプログラムファイルの登録

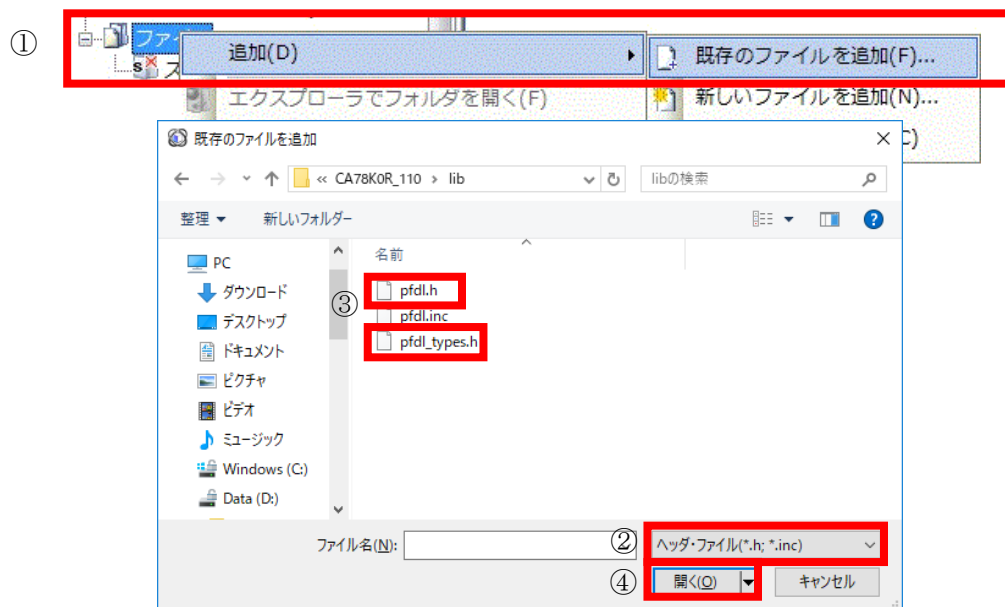


## (2) インクルードファイルの設定

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-2）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ ヘッダファイル(\*.h; \*.inc) ]を選択し、データ・フラッシュ・ライブラリのヘッダファイル（pfdl.h, pfdl\_types.h）を登録してください。

図 6-2 インクルードファイルの登録

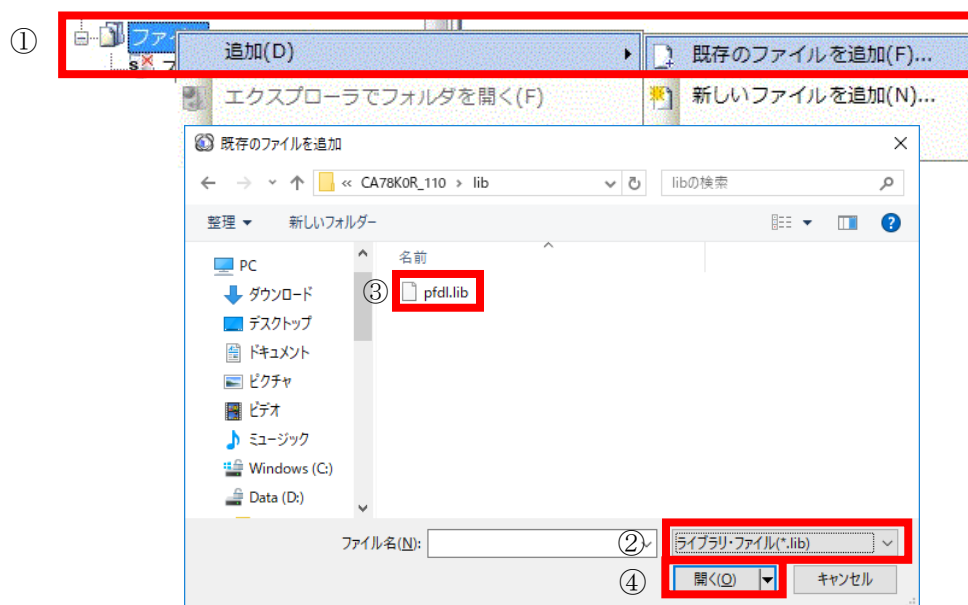


## (3) ライブラリファイルの設定

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-3）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ ライブラリファイル(\*.lib) ]を選択し、データ・フラッシュ・ライブラリのファイル（pfdl.lib）を登録してください。

図 6-3 ライブラリファイルの登録

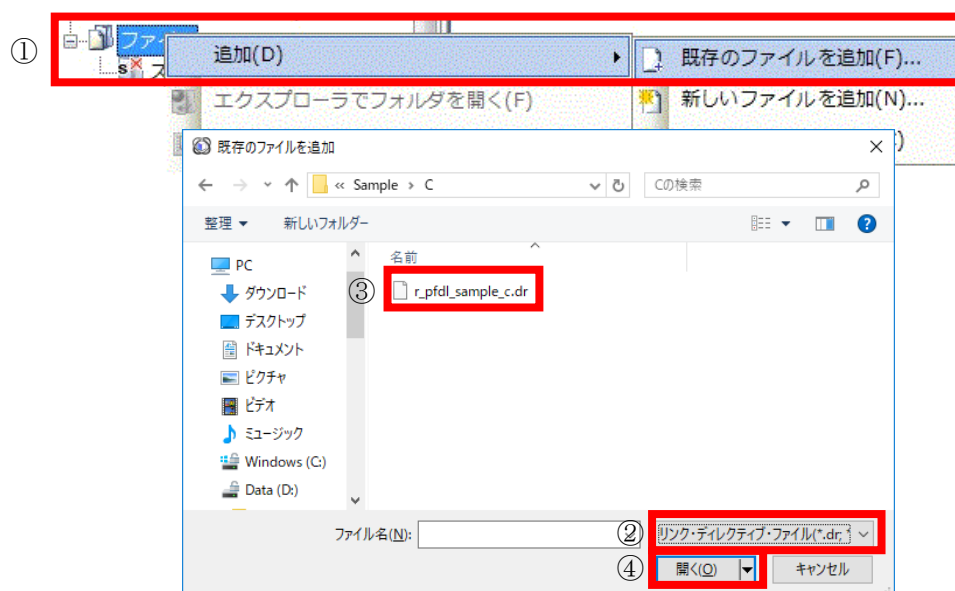


#### (4) リンクディレクティブファイルの設定 (CA78K0R コンパイラ使用時のみ)

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-4）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ リンクディレクティブファイル(\*.dr; \*.dir) ]を選択し、リンクディレクティブファイル名のユーザファイルを登録してください。

図 6-4 リンクディレクティブファイルの登録



#### (5) ビルド

CS+のメニュー [ ビルド ] → [ ビルド・プロジェクト ] をクリックして、ビルドを実行します。

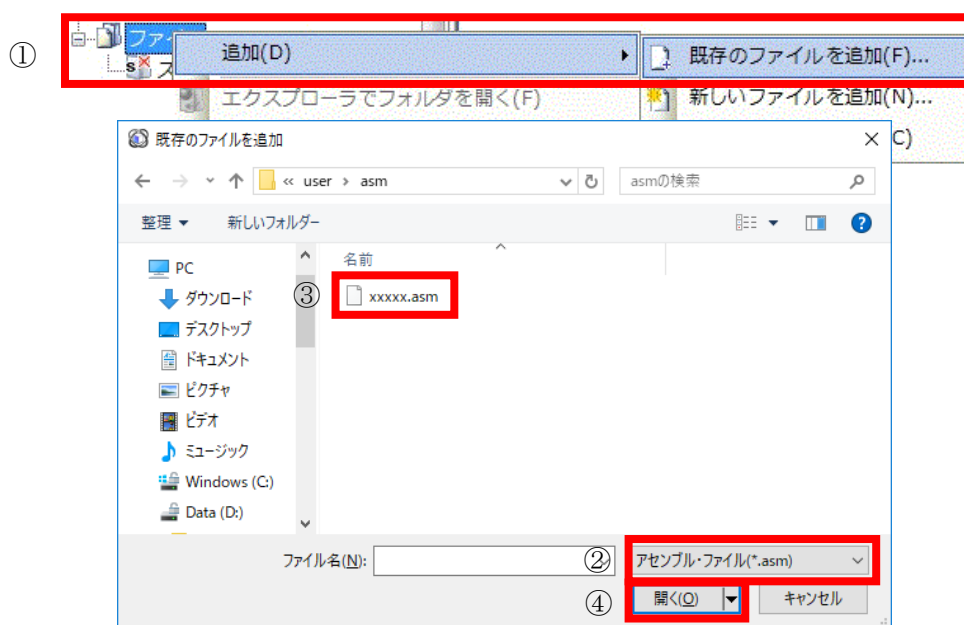
## 6.2.2 アセンブリ言語の場合

### (1) プロジェクトの作成とソースファイルの設定

CS+でプロジェクトを作成し、表示された画面の左側にある[ プロジェクト・ツリー ]から、[ ファイル ]を右クリックして表示されるリストの[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-5）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ アセンブルファイル(\*.asm) ]を選択し、ソースファイルとしてユーザプログラムのファイルを登録してください。

図 6-5 ユーザプログラムファイルの登録



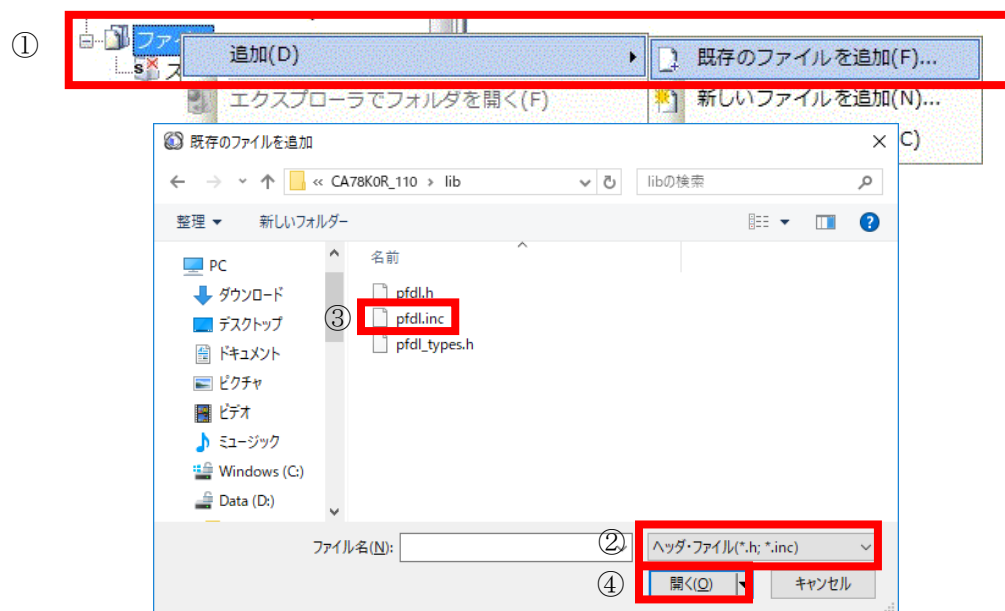
### (2) インクルードファイルの設定

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-6）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ ヘッダファイル(\*.h; \*.inc) ]を選択し、データ・フラッシュ・ライブラリのヘッダファイル (pfdl.inc) を登録してください。



図 6-6 インクルードファイルの登録

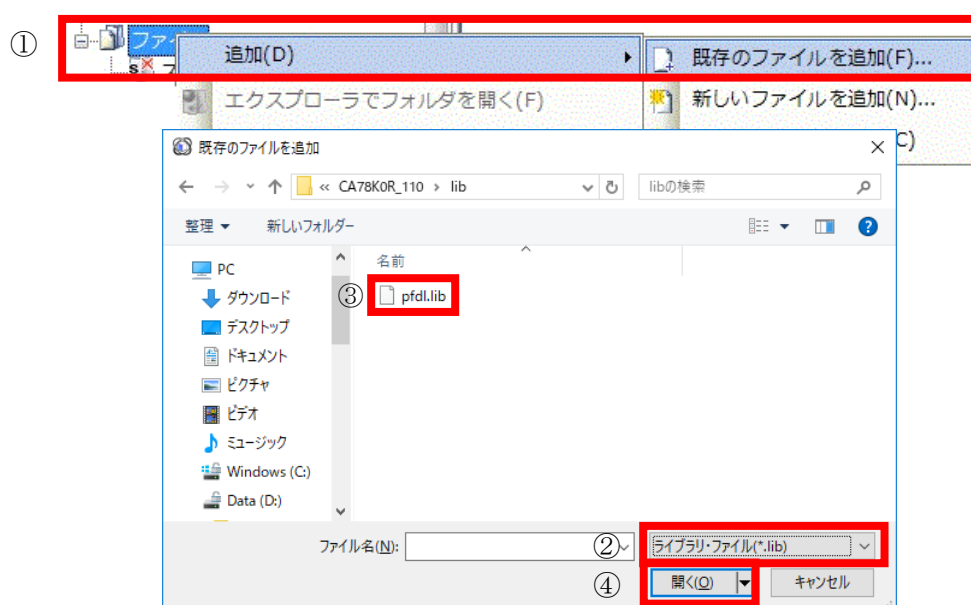


### (3) ライブラリファイルの設定

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます（図 6-7）。

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ ライブラリファイル(\*.lib) ]を選択し、データ・フラッシュ・ライブラリのファイル (pfdl.lib) を登録してください。

図 6-7 ライブラリファイルの登録

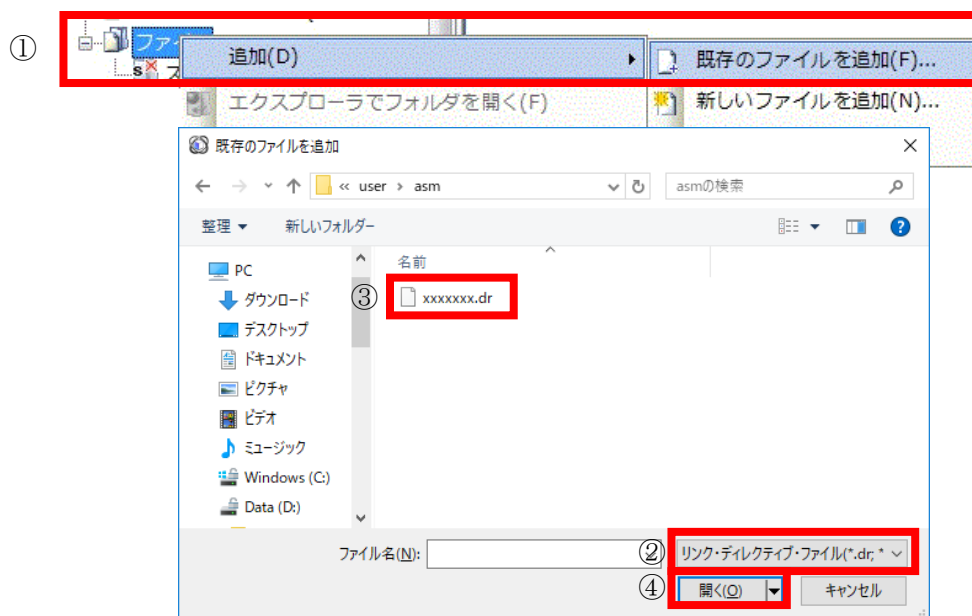


#### (4) リンクディレクティブの設定 (CA78K0R コンパイラ使用時のみ)

CS+の[ プロジェクト・ツリー ]にある、[ ファイル ]を右クリックして表示されるリストから[ 追加 ]を選択し、[ 既存のファイルを追加 ]をクリックすると、[ 既存のファイルを追加 ]画面が表示されます。(図 6-8)

次に画面中にある[ ファイルの種類 ]のプルダウンメニューをクリックすると、ファイルの種類の一覧が表示されるので、その中にある[ リンクディレクティブファイル(\*.dr; \*.dir) ]を選択し、リンクディレクティブファイル名のユーザファイルを登録してください。

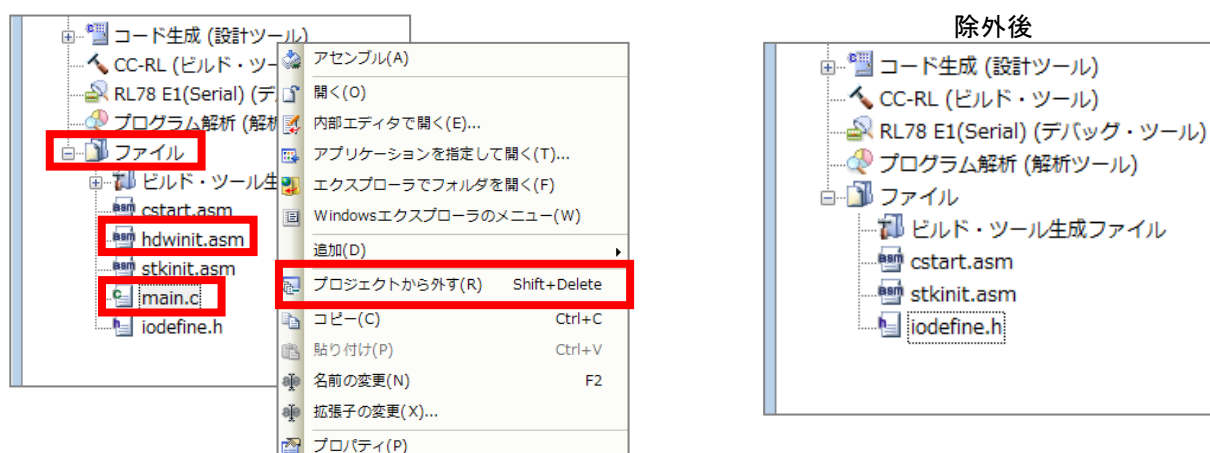
図 6-8 リンクディレクティブファイルの登録



#### (5) 自動生成されたファイルの除外(CC-RL コンパイラ使用時のみ)

CC-RL コンパイラ用 CS+では、[ プロジェクト・ツリー ]の[ ファイル ]にいくつかの自動生成されるファイルがあります。この中で、"main.c"と"hdwinit.asm"の処理は、既にデータ・フラッシュ・ライブラリのサンプルプログラムに含まれています。その為、この2つのファイルをプロジェクトから除外する必要があります。

図 6-9 自動生成ファイルの除外



## (6) ビルド

CS+のメニュー [ ビルド ] → [ ビルド・プロジェクト ] をクリックして、ビルドを実行します。

## 6.3 ビルド時の注意事項

### 6.3.1 CA78K0R コンパイラ使用時

#### (1) オンチップ・デバッグ機能使用時

CS+でオンチップ・デバッグ機能を有効にした後、プログラムのビルドを行うと、以下のようなエラーが発生する場合があります。

RA78K0R error E3212: Default segment can't allocate to memory - ignored  
Segment '??OCDROM' at xxxxxH-200H

このエラーは、オンチップ・デバッグ機能で使用するモニター領域(OCDROM)用のセグメント配置ができないために発生していますので、エラーを回避するためには、プロジェクトに組み込んでいるリンクディレクティブファイル(\*.dr)に、以下のような内容を追記し、セグメント配置用の領域を別途用意してください。

MEMORY OCD\_ROM : ( 0xxxxxH, 00200H )

備考 1. xxxxx : エラーが発生した箇所の先頭アドレス

2. 領域名「OCD\_ROM」は参考例としての表記となります。

## 6.3.2 CC-RL コンパイラ使用時

### (1) オンチップ・デバッグ機能使用時

CS+でオンチップ・デバッグ機能を有効にした後、プログラムのビルドを行うと、以下のようなエラーが発生する場合があります。

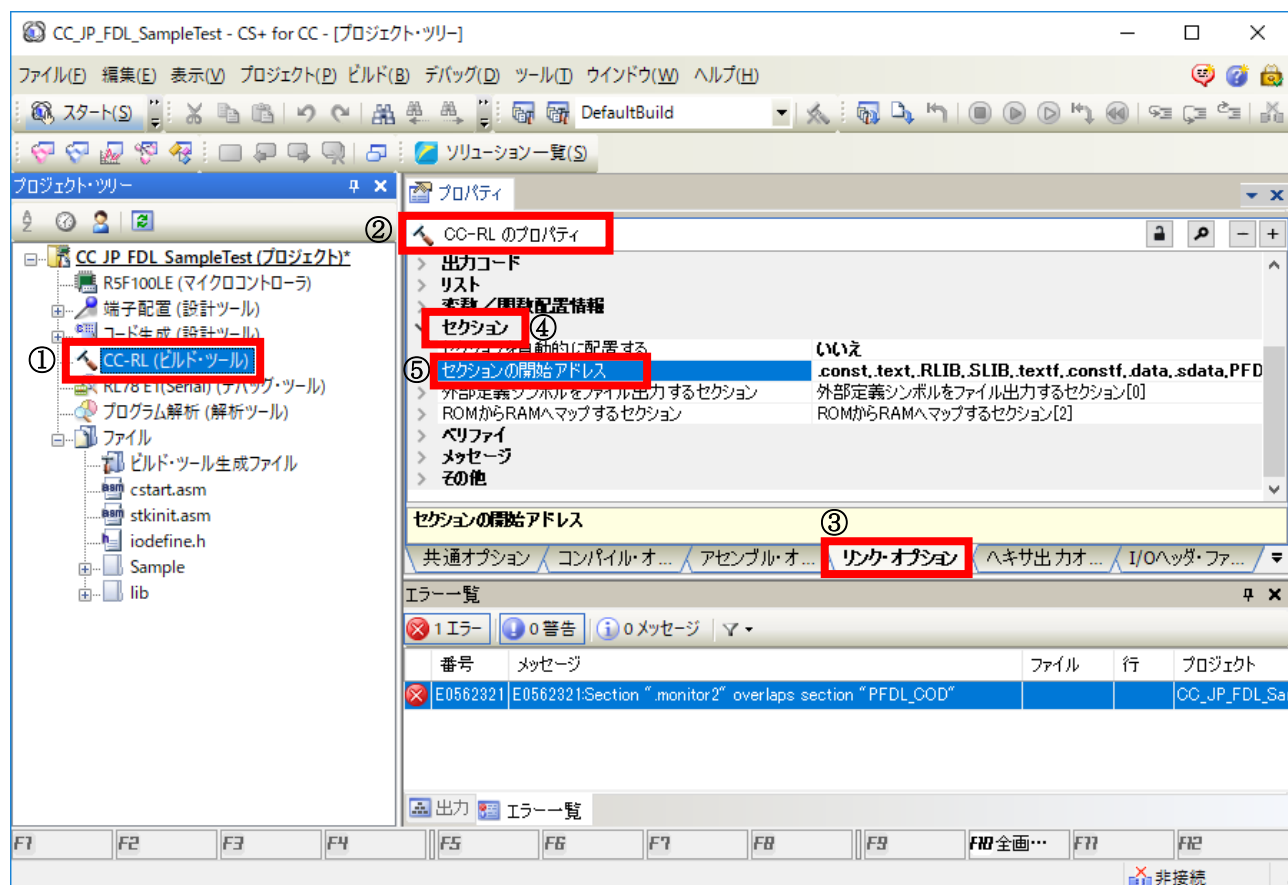
E0562321:Section ".monitor2" overlaps section "xxxxx"

このエラーは、オンチップ・デバッグ機能で使用するモニター領域(OCDFROM)用のセクション配置ができないために発生しています。エラーを回避するためには、CS+の[プロジェクト・ツリー]から①CC-RL(ビルド・ツール)]を選択、右クリックで表示されるプロパティを選択し、表示された②"CC-RLのプロパティ"の③"リンク・オプション"タブを選択します。④"セクション"項目内の⑤"セクションの開始アドレス"で、オンチップ・デバッグモニター用のセクション配置領域(monitor2 : R5F100LE では、初期アドレス 0xFE00-0xFFFF)に、他のセクションが重複しないよう配置を修正してください。(図 6-10)

セクションの設定の詳細については、CC-RL コンパイラ ユーザーズマニュアルをご確認ください。

備考 1. xxxxx : セクション名

図 6-10 セクション配置の変更



## 第7章 デバッグ方法

IECUBE、またはオンチップ・デバッグ・エミュレータ E1、E2、E2 エミュレータ Lite、及び E20 を使用してデバッグを行う場合につきましては、以下の資料の参照してください。

タイトル
CubeSuite+ 統合開発環境 ユーザーズマニュアル RL78 デバッグ編[CS+ for CA,CX] <sup>注</sup>
CS+ 統合開発環境 ユーザーズマニュアル RL78 デバッグ・ツール編[CS+ for CC] <sup>注</sup>

注. このドキュメントは、弊社HPの「統合開発環境 CS+ (旧 CubeSuite+)」のページからダウンロードしてください。

### 7.1 デバッグ時の注意事項

- (1) CubeSuite+ Ver.1.01 より前のバージョンで、オンチップ・デバッグ・エミュレータ E1、E2、E2 エミュレータ Lite、もしくは E20 を使用してデータ・フラッシュ・ライブラリ Type04 Ver.1.05 のコマンドを実行した場合、シーケンサが完了した事を確認するまでブレイクを実行しないで下さい。シーケンサが正常に動作できなくなります
- (2) フラッシュ・ライブラリのデバッグはシミュレータでは実行できません。デバッグを行う場合は、RL78 マイクロコントローラのオンチップ・デバッグ機能を使用するか、もしくは IECUBE をご用意ください。

## 第8章 サンプルプログラム

添付のサンプルプログラム( r\_pfdl\_sample.c.c )は、R5F100LEA(RL78/G13)を対象に QB-R5F100LE-TB ボードでデータ・フラッシュ・ライブラリ Type04 の使用方法を簡単に確認することが可能なように用意しているプログラムです。あくまで参考例となりますので、必ずサンプルプログラム通りに作成する必要があるわけではありません。簡易的な動作確認用のプログラムとしてご使用ください。

また、CA78K0R コンパイラ用サンプルプログラムのリンクディレクティブファイル( r\_pfdl\_sample.c.dr )は、サンプルプログラムで使用するスタックやデータバッファ等を配置禁止領域<sup>注1</sup>に配置しないように指定する事を目的としています。サンプルプログラムを使用する場合は、こちらのファイルも一緒に組み込んでください。<sup>注2,3</sup>

CC-RL コンパイラ用サンプルプログラムには、リンクディレクティブファイル( r\_pfdl\_sample.c.dr )は不要ですが、CS+の画面上で、サンプルプログラムで使用するスタックやデータバッファ等を配置禁止領域<sup>注1</sup>に配置しないように[リンク・オプション]で、[セクション]配置を設定する必要があります。

- 注 1. 詳細については、ユーザーズマニュアルの「2.2 ソフトウェア環境」を参照してください。
2. 添付のリンクディレクティブファイルは RAM 領域のサイズ指定が 2K バイトとなっていますが、RAM が 2K バイトより大きい製品でも、サンプルプログラム( r\_pfdl\_sample.c.c )を使用する場合に、定義されている領域の変更をする必要はありません。そのままビルド等を行う事ができます。
  3. ご使用の環境やプログラムの変更によっては使用中のデータが意図しない領域へ配置される場合があります。実行モジュール作成後はマップファイルを確認し、プログラムやデータの配置状態を必ず確認してください。また、各コードやデータの定義方法や配置条件等については、CS+のユーザーズマニュアルを参照してください。

### 8.1 サンプルプログラムの初期設定

サンプルプログラムは以下の初期設定で動作します。変更が必要な場合は、サンプルプログラムを修正してください。

- ・ CPU の動作周波数 : 高速オンチップ・オシレータ 32MHz
- ・ 電圧モード : ハイスピード・モード

## 8.2 オプション・バイトとオンチップ・デバッグの設定について

オンチップ・デバッグを行う場合は「オンチップ・デバッグの許可／禁止をリンク・オプションで設定する」を「はい」に設定し、「オンチップ・デバッグ・オプション・バイト制御値」を「84」に指定してください。CC-RLコンパイラ用では、「デバッグ・モニタ領域を設定する」を「はい」に指定してください。また、サンプルプログラムは高速オンチップ・オシレータを32MHzに設定する事で正常に動作します。リンク・オプションから、「ユーザ・オプション・バイトを設定する」を「はい」に指定後、ユーザ・オプション・バイト値を「xxxxE8」に指定し、高速オンチップ・オシレータを32MHzに設定してください。

図8-1(a) オプション・バイトの設定(CA78K0Rコンパイラ)

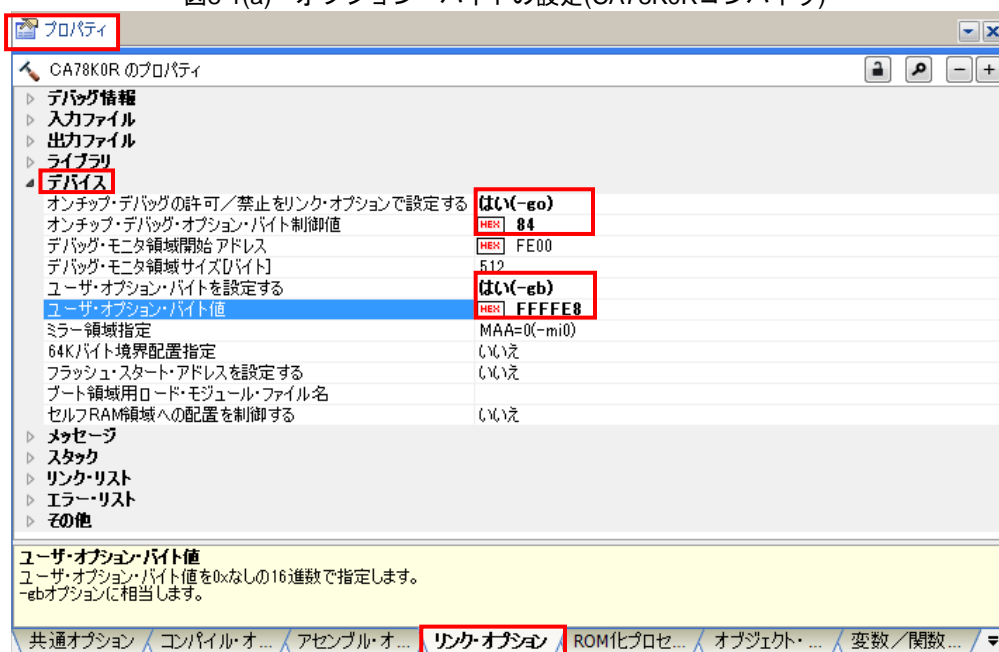
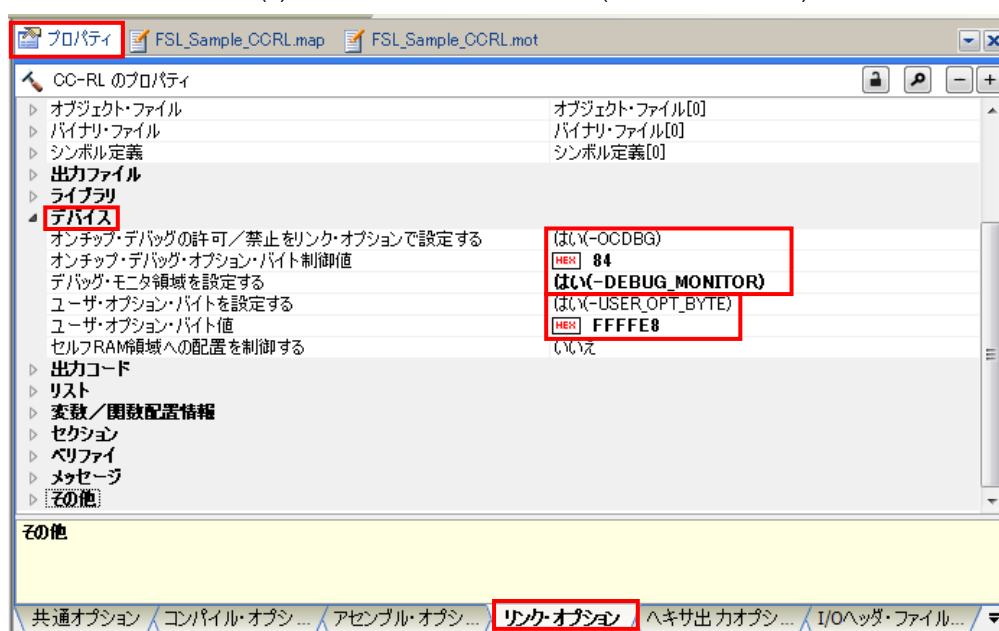


図8-1(b) オプション・バイトの設定(CC-RLコンパイラ)



### 8.3 C言語用サンプルプログラムのコンパイルスイッチ

サンプルプログラムには以下のようなコンパイルスイッチが用意されています。コンパイルスイッチは QB-R5F100LE-TB ボードで動作確認用の LED を点灯させるために使用しています。使用する場合は、対象となる CPU ボードの「#define」宣言が有効になるように、[#if 0]を[#if 1]に変更してください。

```
/* **** */
/* サンプル・プログラム・プログラムスイッチ用シンボル */
/* **** */
/* QB-R5F100LE-TB 単体使用時 ON 可能 */
#if 0
#define __QB_R5F100LE_TB__

/* その他ボード */
#else
#define __NON_TARGET__
#endif
```

QB-R5F100LE-TB の場合、#if 1 に設定可能



## 8.4 内蔵RAM領域の定義

### 8.4.1 CA78K0R コンパイラ使用時

CA78K0R コンパイラ使用時の初期状態では、内蔵 RAM 領域については全域が"RAM"という名称の領域として自動的に定義されています。特にリンクディレクティブファイル等で指定しない限り、スタック等がこの領域に配置<sup>注</sup>される事になりますが、この場合、フラッシュ・ライブラリの使用禁止領域(セルフ RAM、FFE20H-FFEFFH の領域)にスタックやデータバッファ等が配置されてしまい、プログラムが正常に動作できなくなることがあります。

添付されているサンプルプログラム用リンクディレクティブファイルでは、解決策の一つとして"RAM"名の領域をフラッシュ・ライブラリの使用禁止領域を含めないように再定義し、スタック等が使用禁止領域に配置されないようにしています。

MEMORY RAM : (0FEF88H, 000E98H)
---------------------------------

上記は、"RAM"名の領域を、FEF88H から E98H バイトのサイズの領域 (FEF88H-FFE1FH) <sup>注</sup>に再定義し、フラッシュ・ライブラリの使用禁止領域が"RAM"名の領域に含まれないようにしています。

ただし、この設定だけでは「FFE20H-FFEFFH」の領域が他の用途でも使用できなくなるため、別途以下の定義を追加する必要があります。この領域の名称に関しては特に制限等はありません。

MEMORY SADDR_RAM: (OFFE20H, 0000E0H)
--------------------------------------

また、セルフ RAM 領域がある場合は、その範囲を以下のように"SELFRAM"名の領域として定義する事により、この領域に対し、自動的に変数等が配置されないように制限する事が可能です。

MEMORY SELFRAM : (0FEF00H, 000088H)
-------------------------------------

以下に、RL78/G13(RAM 4KB/ROM 64KB 製品)の場合の設定例を記載します。

<pre>; ----- ; Define new memory entry for Self-RAM ; MEMORY SELFRAM : ( 0FEF00H, 000088H )</pre>	セルフ RAM 領域の定義
<pre>; ----- ; Redefined default data segment RAM ; MEMORY RAM : ( 0FEF88H, 000E98H )</pre>	標準的に使用する RAM 領域の定義
<pre>; ----- ; Define new memory entry for saddr area ; MEMORY RAM_SADDR : ( OFFE20H, 0000E0H )</pre>	FFE20H-FFEFFH 領域の定義

注. CA78K0Rのリンカでは、配置先が指定されないデータ（セグメント・タイプDSEGおよびBSEG）はそのデータの再配置属性に従い、内蔵RAM領域に配置されます。そのため、状況によっては特定のデータが"RAM"名の領域に配置されない場合があります。

各データの定義や配置方法等の詳細についてはCS+のユーザーズマニュアルを参照してください。

また、ビルド時に生成されるマップファイル(\*.map)を必ず参照し、各データの配置状態を確認してください。

## 8.4.2 CC-RL コンパイラ使用時

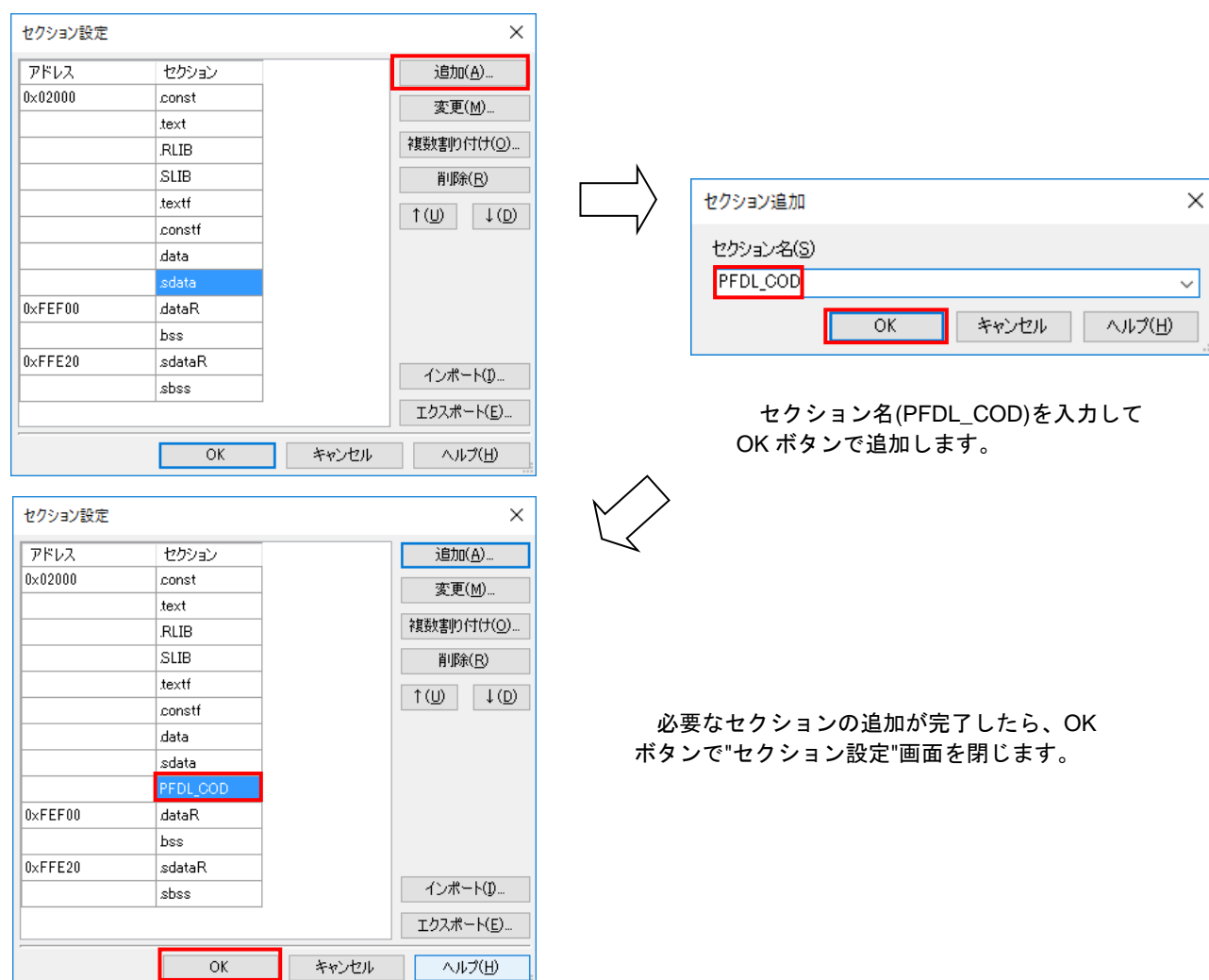
### (1) インクルード・パスの追加

CC-RL コンパイラ用 CS+は、初期状態ではインクルード・パスは設定されていません。データ・フラッシュ・ライブラリで使用するヘッダファイルのインクルード・パスを追加する必要があります。データ・フラッシュ・ライブラリで使用するヘッダファイルは、"pfdl.h"、"pfdl\_types.h"、及び CS+が自動生成する"iodefine.h"です。[ コンパイル・オプション ]-[ プリプロセス ]-[ 追加のインクルード・パス ]で、各ファイルが存在するインクルード・パスを追加してください。

### (2) セクションの定義

CC-RL コンパイラ用 CS+ 使用時は、ROM 領域、RAM 領域で使用する各セクションを定義する必要があります。セクションの定義は、CS+の画面の[ リンク・オプション ]-[ セクション ]で指定することができます。[セクションを自動的に配置する]を"いいえ"に設定し、[セクションの開始アドレス]の設定画面を開き、ROM 領域にデータ・フラッシュ・ライブラリのセクション(ここでは、本サンプルプログラム動作に必要なセクション、PFDL\_COD)を追加します。(図 8-2)

図8-2 データ・フラッシュ・ライブラリのセクション設定例(ROM領域)



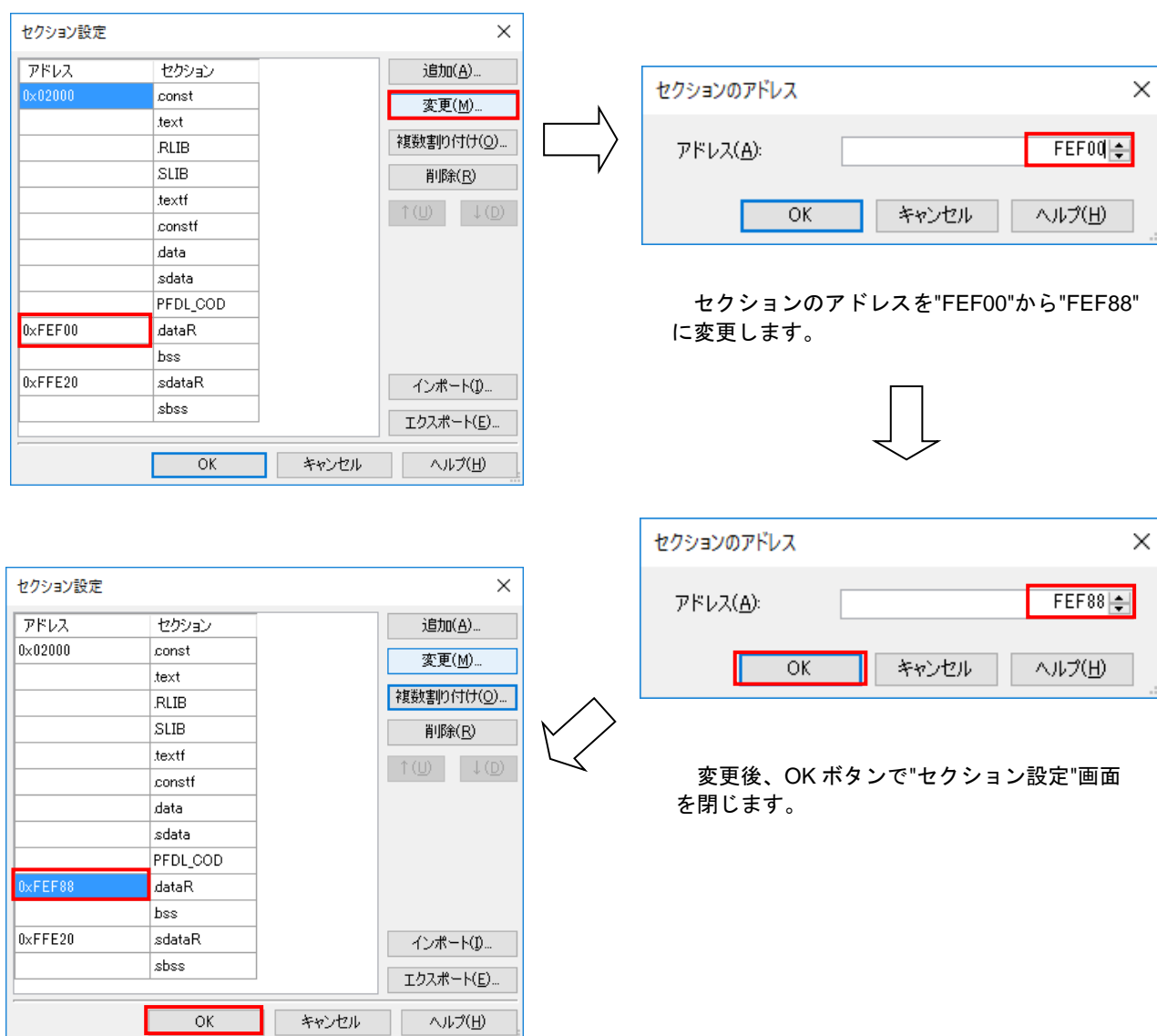
セクション名(PFDL\_COD)を入力して  
OK ボタンで追加します。

必要なセクションの追加が完了したら、OK  
ボタンで"セクション設定"画面を閉じます。

### (3) セルフ RAM 領域の確保

CC-RL コンパイラ用 CS+ でセクション設定の初期状態では、ユーザ RAM 領域が内蔵 RAM 領域の先頭(サンプルプログラム対象の R5F100LEA では 0xFE00)から確保されています。しかし、R5F100LEA では、データ・フラッシュ・ライブラリがセルフ RAM 領域として 0xFE00-0xFE87 を使用する為、この領域を避けてユーザ RAM 領域を設定する必要があります。ここでは、アドレス 0xFE00 から設定されているユーザデータの先頭アドレスを 0xFE88 に変更する例を示します。

図8-3 ユーザRAM領域範囲の変更例(RAM領域)



注意) セクションの設定で[セクションを自動的に配置する]を一度"いいえ"にして、ユーザ RAM 領域範囲の変更をした後に、"はい"に設定することで、ユーザが定義したセクションも含め、再自動配置されますが、自動配置では、ユーザが定義していない領域に対してもセクションを配置する可能性があり、意図していなかった領域にデータが配置される可能性があります。必ず、データ・フラッシュ・ライブラリで使用するソフトウェアリソース(特に RAM 上データ)が、配置可能領域に配置されていることをマップファイル(\*.map)等で確認してください。

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生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
  2. 当社製品、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またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
  3. 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、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング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）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等  
当社製品は、データシート等により高信頼性、Harsh environment向け製品と定義しているものを除き、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（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
  6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合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、データシート等において高信頼性、Harsh environment向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他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。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
  9. 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。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
  10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものいたします。
  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。
  12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 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が直接的、間接的に支配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。

(Rev.4.0-1 2017.11)



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 エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■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：<https://www.renesas.com/contact/>