

この度は、EEPROM エミュレーション・ライブラリ Pack01 パッケージ Ver.2.10 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、EEPROM エミュレーション・ライブラリ Pack01 パッケージ Ver.2.10 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
第7章	デバッグ方法.....	12
7.1	デバッグ時の注意事項	12
第8章	サンプルプログラム	13
8.1	サンプルプログラムの初期設定.....	13
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.....	14
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	14
8.4	内蔵RAM領域の定義	15

第1章 対象製品

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 EEPOM エミュレーション・ライブラリ Pack01	V1.16	RENESAS_RL78_EEL-FDL_T01_PACK01_2V10.exe	V2.10

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 EEPROM エミュレーション・ライブラリ Pack01 ユーザーズマニュアル	R01US0054JJ0103

第3章 バージョンアップ内容

本パッケージ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	V2.10	ライブラリ	EEL Pack01 の Pack バージョンは、V1.15 から V1.16 にバージョンアップしました。 DWP(データライトポインタ)の機能改善(内部処理)しました。ユーザインタフェースの変更はありません。
		ユーザーズマニュアル	ユーザーズマニュアル(R01US0054JJ0103)から変更はありません。

第4章 対応ツール

EEPROM エミュレーション・ライブラリ Pack01 Ver.1.16 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。

ツール名	バージョン
統合開発環境 : CubeSuite+ ^注	V1.00.00 以降
統合開発環境 : CS+ ^注	V3.00.00 以降

注. EEPROMエミュレーション・ライブラリPack01 Ver.1.16の対象とするCA78K0Rコンパイラバージョンは、V1.10以降です。

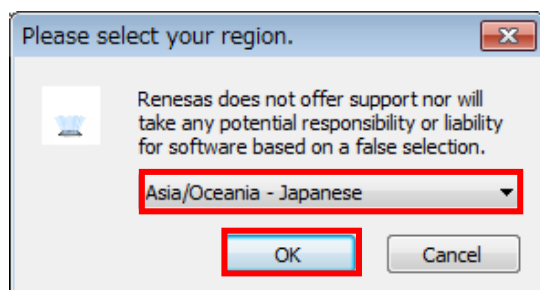
第5章 インストール

この章では、EEPROMエミュレーション・ライブラリ Pack01 パッケージ Ver.2.10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

EEPROMエミュレーション・ライブラリ Pack01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) EEPROMエミュレーション・ライブラリ Pack01 パッケージの圧縮ファイルを解凍、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。



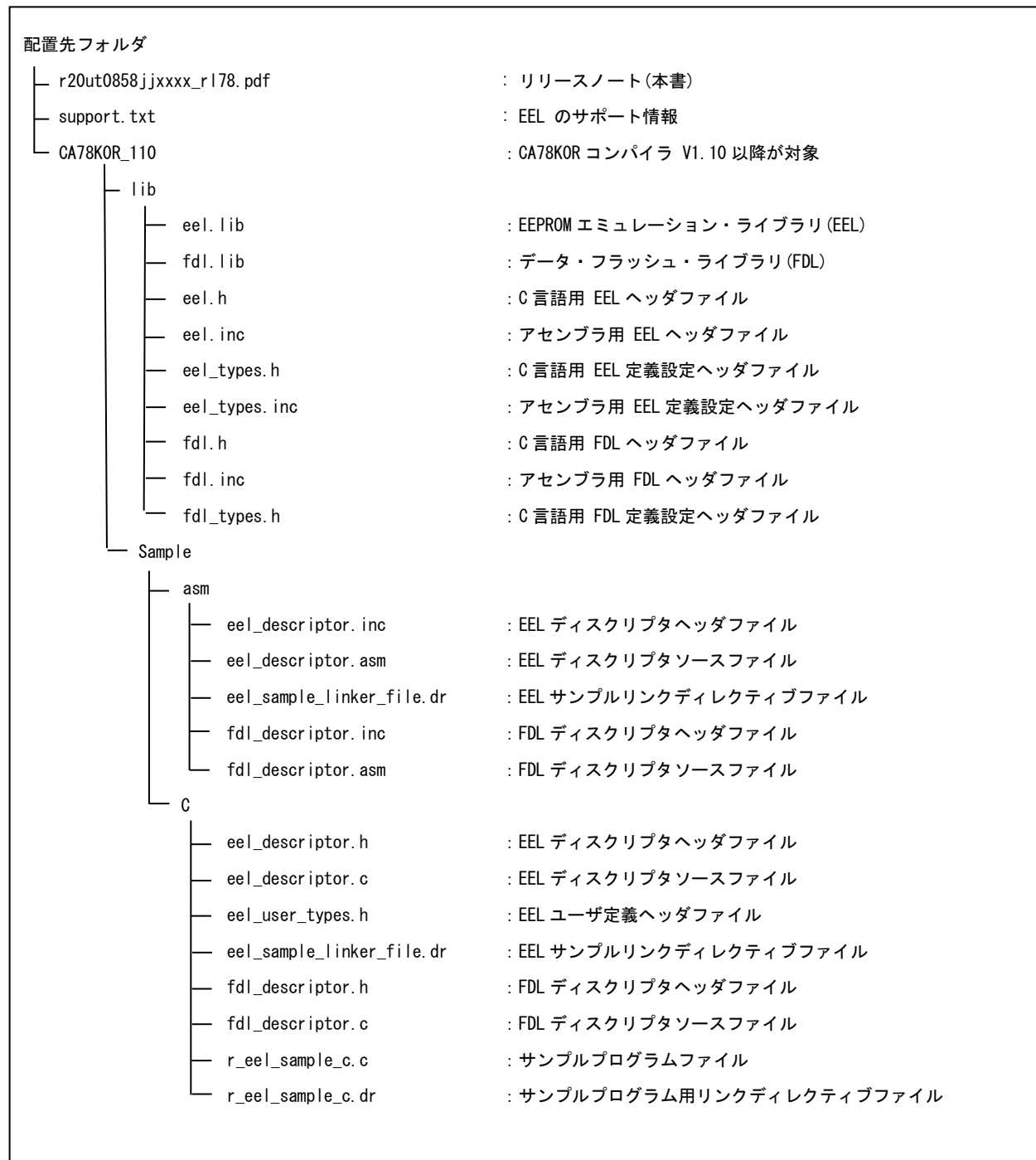
5.2 アンインストール方法

EEPROMエミュレーション・ライブラリ Pack01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EEPROMエミュレーション・ライブラリが入っているフォルダを削除します。

5.3 ファイル構成

EEPROMエミュレーション・ライブラリ Pack01 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. 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) を一緒に組み込んでください。

第6章 ビルド方法

この章では、EEPROMエミュレーション・ライブラリ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

EEPROM エミュレーション・ライブラリ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。

- ・統合開発環境 CS+ V3.00.00 以降、または CubeSuite+ V1.00.00 以降

6.2 CS+(旧 CubeSuite+)でのビルド方法

CS+を用いて EEPROM エミュレーション・ライブラリ Pack01 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。

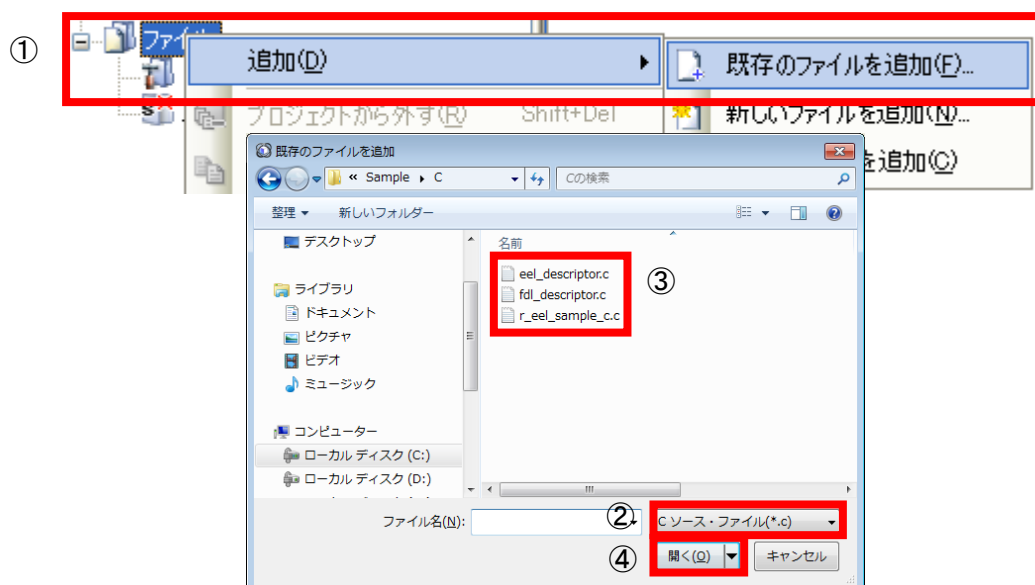
6.2.1 C言語の場合

(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(サンプルの場合は r_eel_sample.c.c)、及び EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリのディスクリプタファイル (eel_descriptor.c, fdl_descriptor.c) 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

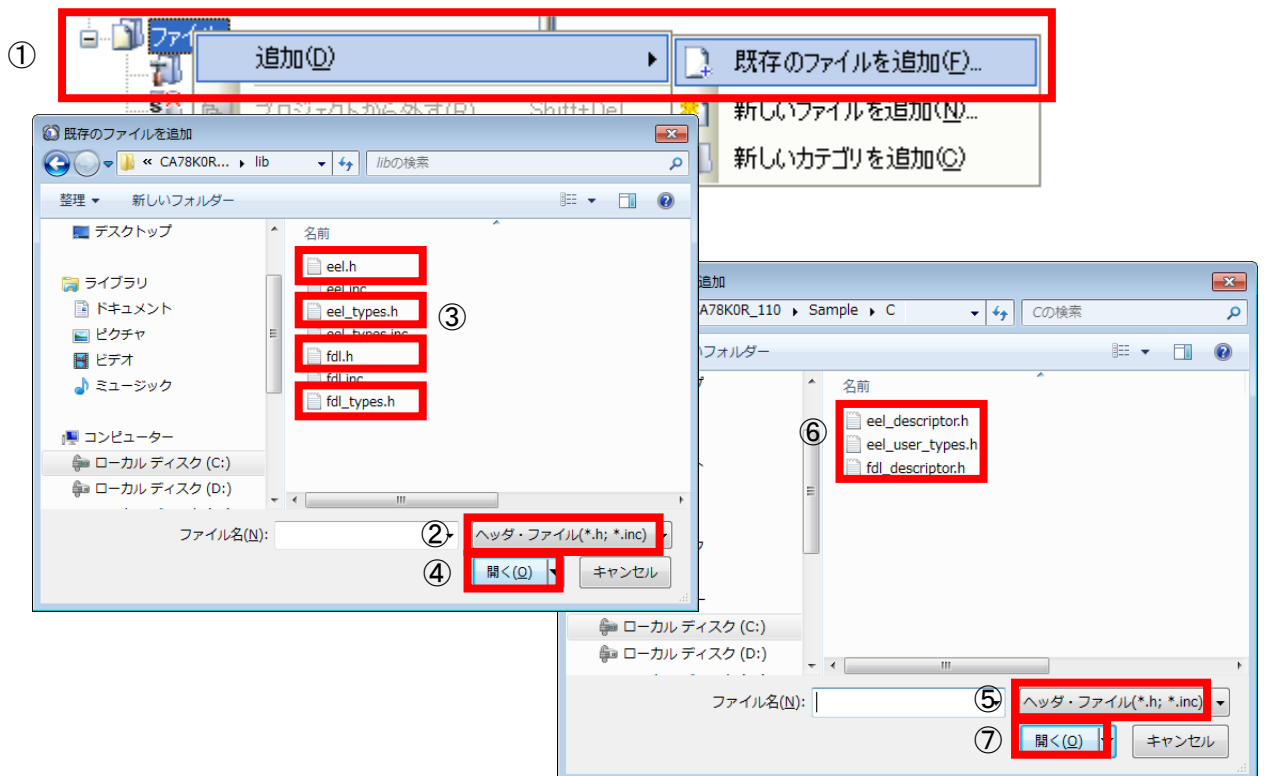


(2) インクルードファイルの設定

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、EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリのヘッダファイル、及びディスクリプタ用ヘッダファイル (eel.h, eel_types.h, fdl.h, fdl_types.h, eel_descriptor.h, fdl_descriptor.h, eel_user_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

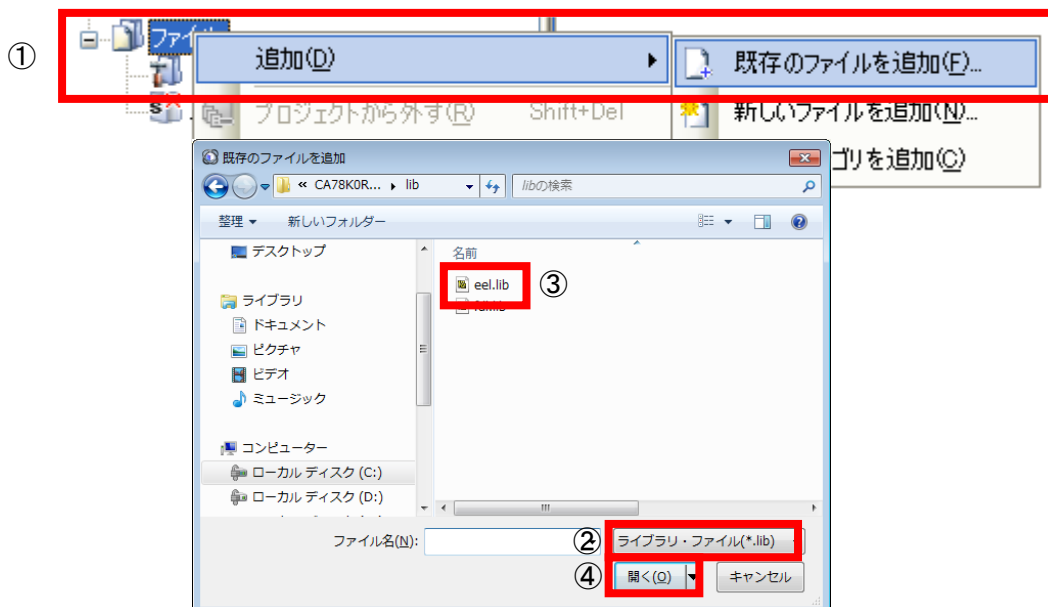


(3) ライブラリファイルの設定

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、EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリ (eel.lib, fdl.lib) のファイルを登録してください。

図 6-3 ライブラリファイルの登録

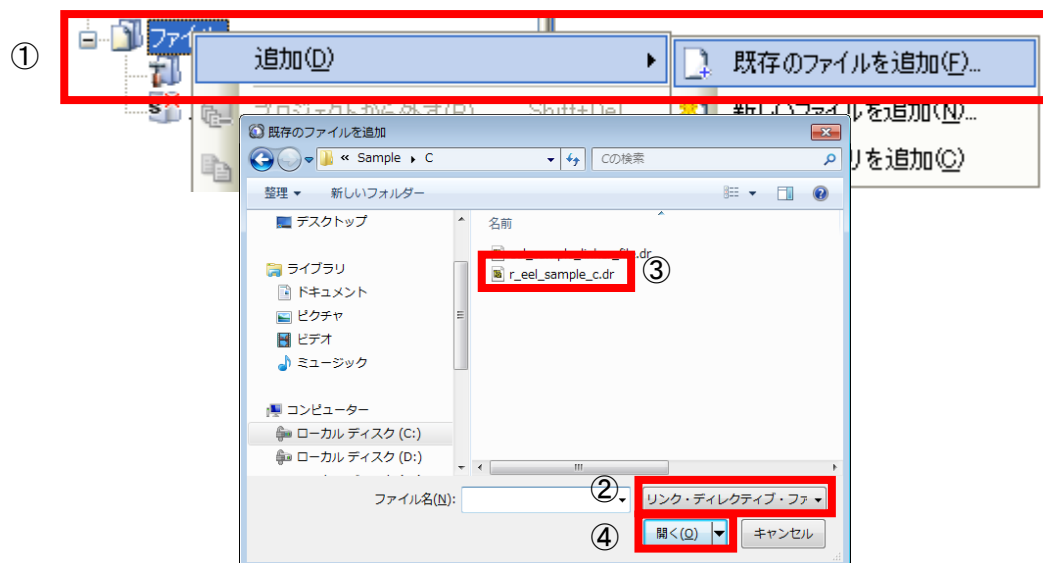


(4) リンクディレクティブファイルの設定

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(サンプルの場合は r_eel_sample_c.dr[※])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



注. 添付のサンプルディレクティブファイルを使用する場合は編集・修正が必要となる場合があります。

(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。

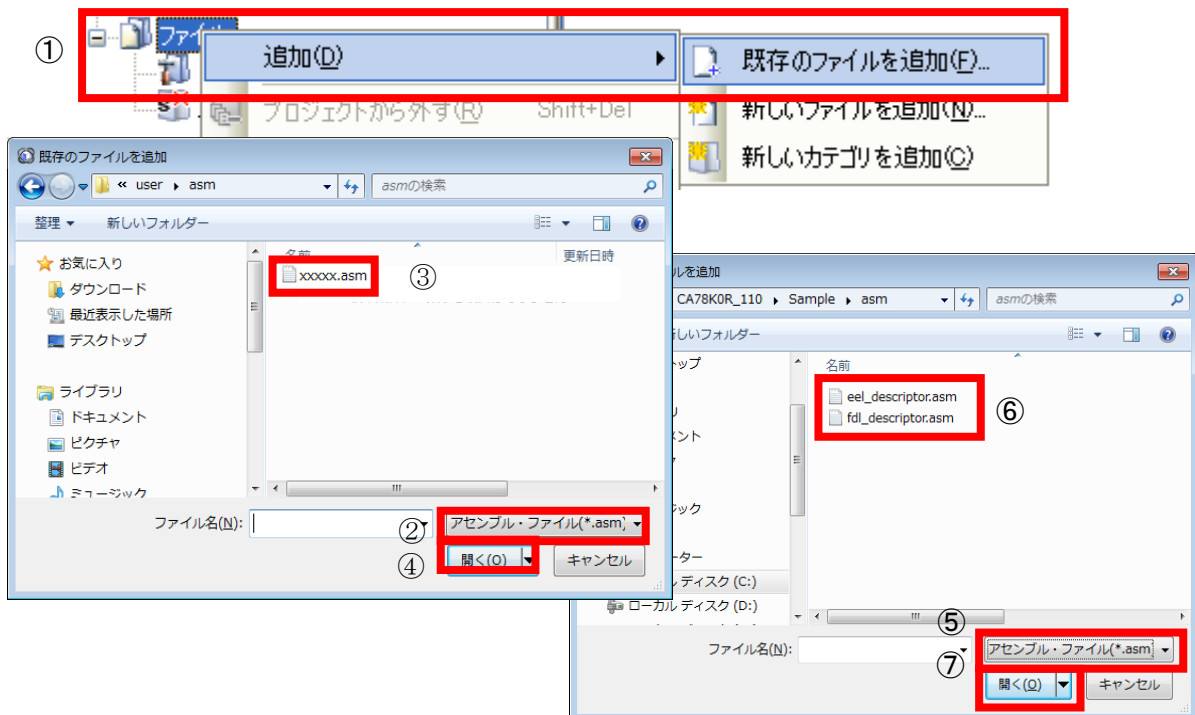
6.2.2 アセンブリ言語の場合

(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、ユーザ作成の[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、及び EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリのディスクリプタファイル (eel_descriptor.asm, fdl_descriptor.asm) 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

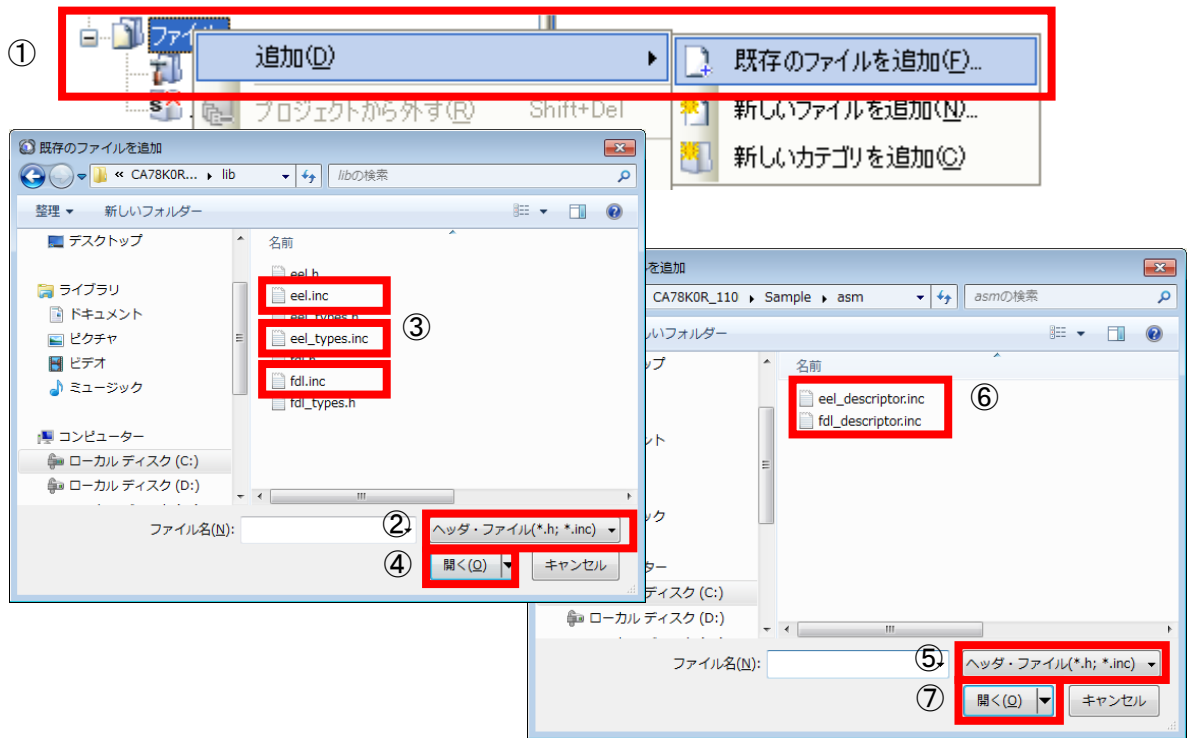


(2) インクルードファイルの設定

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、EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリのヘッダファイル、及びディスクリプタ用ヘッダファイル (eel.inc, eel_types.inc, fdl.inc, eel_descriptor.inc, fdl_descriptor.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

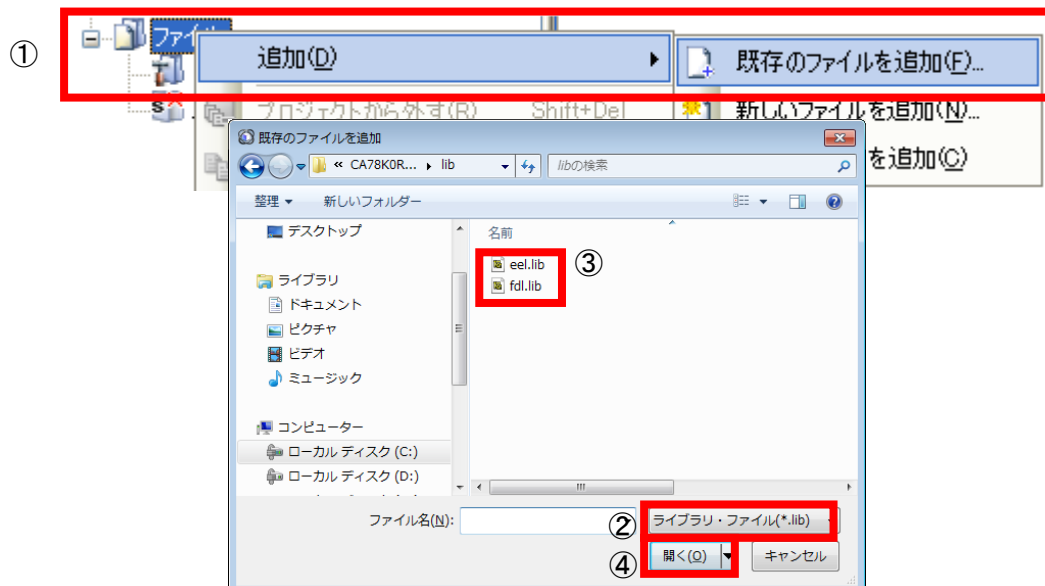


(3) ライブラリファイルの設定

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、EEPROM エミュレーション・ライブラリ、データ・フラッシュ・ライブラリ（eel.lib, fdl.lib）のファイルを登録してください。

図 6-7 ライブラリファイルの登録

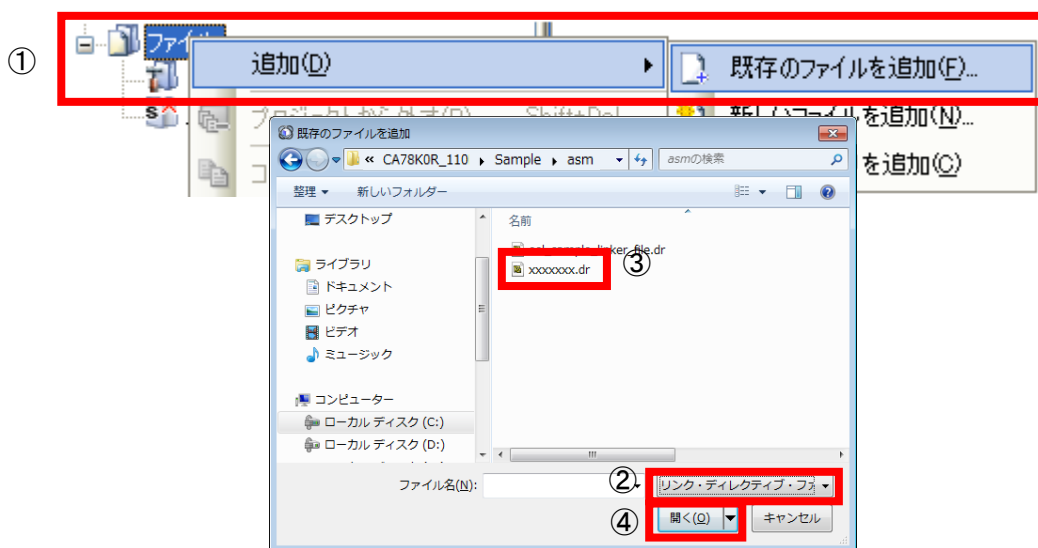


(4) リンクディレクティブの設定

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) 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

(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。

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、及び E20 を使用してデバッグを行う場合につきましては、以下の資料を参照してください。

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] ^注

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

EEPROM エミュレーション・ライブラリ Pack01 を使用してオンチップ・デバッグ・エミュレータ E1、もしくは E20 を使用する際の注意事項

- (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、もしくは E20 を使用して EEPROM エミュレーション・ライブラリ Pack01 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ください。シーケンサが正常に動作できなくなります。

注. 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。

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_eel_sample.c`)は、R5F100LEA(RL78/G13)を対象に QB-R5F100LE-TB ボードで EEPROM エミュレーション・ライブラリ Pack01 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、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_eel_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^{注1}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。^{注2}

- 注 1. 詳細については、ユーザーズマニュアルの「2.3 ソフトウェア・リソース」を参照してください。
2. 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 : フルスピード・モード

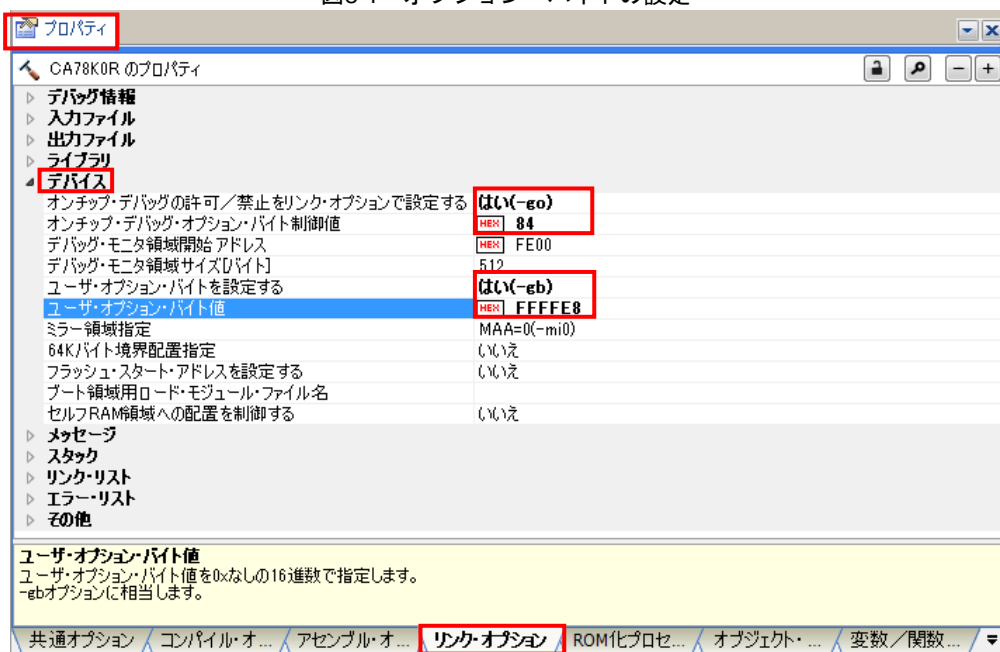
8.2 オプション・バイトとオンチップ・デバッグの設定について

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。

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。

ユーザ・オプション・バイトを設定するには、CS+の「リンク・オプションタブページ」から「ユーザ・オプション・バイトを設定する」を「はい」に設定した後、ユーザ・オプション・バイト値を「xxxxE8」に指定してください。

図8-1 オプション・バイトの設定



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(内蔵発振設定) */
#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

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特に指定しない場合、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、特に指定しない限り、スタック等がこの領域に配置^注される事になります。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-FFEDFH 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 : (0FF300H, 000B20H)
```

上記は、"RAM"名の領域を、FF300H から B20H バイトのサイズの領域 (FF300H-FFE1FH) ^注に再定義し、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: (0FFE20H, 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, 000400H)
```

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

```

; -----
; Define new memory entry for Self-RAM
; -----
MEMORY SELFRAM : ( 0FEF00H, 000400H ) ← セルフ RAM 領域の定義
; -----
; Redefined default data segment RAM
; -----
MEMORY RAM : ( 0FF300H, 000B20H ) ← 標準的に使用する RAM 領域の定義
; -----
; Define new memory entry for saddr area
; -----
MEMORY RAM_SADDR : ( 0FFE20H, 0000E0H ) ← 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。

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か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、その他の不適切に使用しないでください。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、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、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。また、当社製品および技術を、(1)核兵器、化学兵器、生物兵器等の大量破壊兵器およびこれらを運搬することができるミサイル（無人航空機を含みます。）の開発、設計、製造、使用もしくは貯蔵等の目的、(2)通常兵器の開発、設計、製造または使用の目的、または(3)その他の国際的な平和および安全の維持の妨げとなる目的で、自ら使用せず、かつ、第三者に使用、販売、譲渡、輸出、賃貸もしくは使用許諾しないでください。
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. お客様の転売、貸与等により、本書（本ご注意書きを含みます。）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、当社は一切その責任を負わず、お客様にかかる使用に基づく当社への請求につき当社を免責いただきます。
 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. 本資料に記載された情報または当社製品に関し、ご不明点がある場合には、当社営業にお問い合わせください。
- 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。

(Rev.3.0-1 2016.11)



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どうぞ。
総合お問合せ窓口：<https://www.renesas.com/contact/>