

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。
5. 輸出に際しては、「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。本資料に記載されている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用途の目的で使用しないでください。また、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に使用することができません。
6. 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、誤りが無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。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されてい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生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
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  
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、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社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。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、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おりません。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、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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。
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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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。

=== 必ずお読みください ===

## 740 ファミリ用 C コンパイラ パッケージ V.1.01 Release 01

## リリースノート

## 第 4 版

株式会社ルネサス ソリューションズ

2006 年 10 月 16 日

## 概要

この度は 740 ファミリ用 C コンパイラ パッケージ V.1.01 Release 01 をお買い上げいただきまして誠に有難うございます。本資料は 740 ファミリ用 C コンパイラ パッケージ 電子マニュアルの補足等について説明します。電子マニュアルの該当項目をご覧になる場合は、併せてこのリリースノートをご覧いただきますようお願い申し上げます。

## 目次

ユーザ登録.....	3
最新情報のご案内.....	3
バージョンアップ内容.....	4
インストール.....	4
インストールをはじめる前に.....	4
ICC740 のインストーラ.....	4
インストール方法.....	4
インストール後に生成されるディレクトリとソフトウェア.....	5
環境設定.....	6
freeware に格納されているプログラムについて.....	6
注意事項.....	7
割り込み処理に関する注意事項.....	7
未定義関数に関する注意事項.....	8
統合開発環境 High-performance Embedded Workshop と組み合わせて使用する場合の注意事項.....	8
ソースファイルとヘッダ（インクルード）ファイルとの依存関係登録にについて.....	8
プロジェクトフォルダとリンク対象ファイル数について.....	9
IAR Systems 社ドキュメントについて.....	10
icc740.htm, Important information の第 2 項.....	10
manuaus.htm, IAR C Compiler Reference Guide の Support for interruptable ISRs 項.....	10
icc740.htm Important information の第 5 項.....	12
manuals.htm, IAR C Compiler Reference Guide の Inline Assembly 項.....	12
icc740.htm, Important information の第 7 項.....	12

icc740.htm, Important information の第 10 項 .....	12
icc740.htm, Important information の第 11 項 .....	12
icc740.htm, Important information の第 12 項 .....	13
icc740.htm, Important information の第 14 項 .....	13
icc740.htm, New features の第 1 項 .....	13
icc740.htm, Known problems の第 2 項.....	13
icc740.htm, Known problems の第 3 項.....	14
icc740.htm, Known problems の第 4 項.....	14
icc740.htm, Known problems の第 5 項.....	14
icc740.htm, Known problems の第 6 項.....	14
a740.htm, Important information の第 3 項 .....	14
a740.htm, Important information の第 4 項 .....	15
a740.htm, Important information の第 5 項 .....	15
a740.htm, Important information の第 6 項 .....	15
a740.htm, Important information の第 8 項 .....	15
a740.htm, Known problems の第 1 項.....	15
manuals.htm, IAR Assembler Reference Guide の第 2 項 ([740a0088]) .....	16
a740.htm, Known problems の第 2 項 ([740a0087]) .....	16
a740.htm, Known problems の第 4 項.....	16
a740.htm, Known problems の第 5 項.....	16
a740.htm, Known problems の第 6 項.....	16
xlink.htm, Known problems in Current Version の第 2 項.....	16
xlink.htm, Known problems in Current Version の第 3 項.....	16
xlink.htm, Known problems in Current Version の第 4 項.....	17
xlink.htm, Known problems in Current Version の第 5 項.....	17
xlink.htm, Known problems in Current Version の第 6 項.....	17
マニュアル正誤表.....	17
<b>統合開発環境の起動 .....</b>	<b>18</b>
High-performance Embedded Workshop の起動.....	18
Manual Navigator の起動 .....	18
<b>動作確認環境.....</b>	<b>18</b>
<b>ソフトウェアのバージョン一覧 .....</b>	<b>18</b>

- 
- Microsoft、MS- DOS、Windows および Windows NT は、米国 Microsoft Corporation の米国およびその他の国における登録商標です。
  - IBM および AT は、米国 International Business Machines Corporation の登録商標です。
  - Adobe および Acrobat は、Adobe Systems Incorporated (アドビシステムズ社) の登録商標です。
  - その他すべてのブランド名および製品名は個々の所有者の登録商標もしくは商標です。
-

## ユーザ登録

バージョンアップ情報や技術サポート等のサービスを受けるためにユーザ登録を行ってください。ユーザ登録をされていない場合は、これらのサービスを受けることができません。ご購入後 30 日以内にユーザ登録をしてくださるようお願いいたします。

### ユーザ登録方法

インストールすると以下のファイルが生成されます。

C:\Program Files\IAR Systems\ew23\740\support\74\regist.txt

C:\Program Files\IAR Systems\ew23 はデフォルトでインストールした場合のディレクトリ名です。regist.txt のファイル内容をすべてカット&ペーストして以下の電子メールアドレス宛に送付してください。

[regist\\_tool@renesas.com](mailto:regist_tool@renesas.com)

(株)ルネサス テクノロジーの個人情報保護方針につきましては、ルネサステクノロジーのホームページ「個人情報保護について」をご覧ください。

ユーザ登録でご提供頂きました個人情報は、お客様のサポート活動に活用させて頂き、そのために必要な範囲で(株)ルネサス テクノロジー、およびその関係会社、ならびに特約店に、電子データ、書面により提供させて頂きますので、ご了承の程お願い申し上げます。なお、提供を希望されない場合は、提供を停止させて頂きますので、お問い合わせ時にその旨ご連絡ください。その場合、サポート範囲が制約される場合がございます。

## 最新情報のご案内

以下の URL にホームページを公開しています。こちらでは、ルネサス開発環境に関する最新情報が掲載されていますのでご活用ください。

<http://japan.renesas.com/tools>

## バージョンアップ内容

- High-performance Embedded Workshop に対応しました。

## インストール

### インストールをはじめる前に

インストールをはじめる前に次の内容をご確認ください。

- 本製品の「使用権許諾契約書」、「リリースノート（本紙）」などをよくお読みください。製品をインストールした場合は、契約書の記載内容に同意されたものとみなします。
- 製品のインストールは専用のインストーラを使用します。
- インストールの途中でライセンス ID を入力する必要があります。インストールをはじめる前にライセンス ID を確認してください。
- ICC740 をバージョンアップする場合は、あらかじめ、インストールされている ICC740 をアンインストールしてから、新しいバージョンをインストールしてください。

ICC740 をアンインストールするには、「コントロールパネル」－「アプリケーションの追加と削除」を選択しアンインストールを実行してください。

### ICC740 のインストーラ

#### ■ 日本語環境

対応ホスト	対応 OS	インストーラ名	CD-ROM 上のディレクトリ
PC	Windows98 Windows2000 WindowsNT WindowsMe WindowsXP	SETUP.EXE	¥ICC740¥W95J

#### ■ 英語環境

対応ホスト	対応 OS	インストーラ名	CD-ROM 上のディレクトリ
PC	Windows98 Windows2000 WindowsNT WindowsMe WindowsXP	SETUP.EXE	¥ICC740¥W95E

## インストール方法

次の手順でインストールしてください。

- (1) CD-ROM 上の対象製品のインストーラが配置されているディレクトリに移動します。
- (2) インストーラを起動して表示されるメッセージにしたがってインストールを完了してください。

### ユーザ情報の入力について

インストールの途中、会社名や連絡先などのお客様の情報の入力がありますが、これはユーザ登録のために必要なファイルを作成するのに使用しています。

(株)ルネサス テクノロジーの個人情報保護方針につきましては、ルネサステクノロジーのホームページ「個人情報保護について」をご覧ください。

### インストール内容の選択について

ICC740 のみ、あるいは High-performance Embedded Workshop のみといった選択インストールはできません。個別にアンインストールは可能ですので、不要な場合はインストール後に、それぞれをアンインストールしてください。

### プログラムの削除について

インストールしたプログラムは以下の手順で削除します。

- (1) [スタート]→[コントロールパネル]をクリックします。
- (2) [プログラムの追加と削除]アイコンをダブルクリックします。
- (3) [プログラムの変更と削除]タブから削除したいプログラムをクリックします。
  - ルネサス 740 ファミリ用 C コンパイラパッケージ V.1.01 Release 01
  - IAR C Compiler Package for Renesas 740 V2.17A
  - High-performance Embedded Workshop

### AutoUpdater について

インストールが完了すると、AutoUpdater が自動的に起動し、常駐します。

AutoUpdater は、定期的にルネサス開発環境のホームページにアクセスし、各開発環境の更新状況を確認するツールです。

### スタートメニューの構成

インストール後は、[スタート] → [プログラム(P)] → [Renesas] に次のフォルダおよびショートカットが登録されます。

High-performance Embedded Workshop  
 740 Family C Compiler V.1.01 Release 01  
 ユーザ登録  
 Renesas AutoUpdate  
 ルネサス開発環境 HomePage

### インストール後に生成されるディレクトリとソフトウェア

インストールが完了するとインストーラのデフォルトのインストール先ディレクトリの下に、表に示すディレクトリが生成され、各ディレクトリに表のファイルがコピーされます。

- ICC740 インストール先ディレクトリ (デフォルト)  
 C:\Program Files\IAR Systems\ew23\740

表.インストール先ディレクトリとインストールファイル一覧

ディレクトリ	ファイル
bin	a740.exe (アセンブラ) Icc740.exe (C コンパイラ) xlink.exe (リンケージエディタ) xlib.exe (ライブラリアン)
inc	(ヘッダファイル類)
smp\Tw74	tutor3.c (サンプルプログラム)
lib	(ライブラリ類)
manual	rjj10j1409_icc740_u.pdf (C コンパイラパッケージ電子マニュアル)

## 環境設定

コマンドプロンプト上で ICC740 を利用する場合は、次の環境変数を設定してください。

### PC の環境設定

表の中の「自動」は、インストーラで AUTOEXEC.BAT を書きかえることができます。インストール時に環境変数の設定ダイアログボックスで「AUTOEXEC.BAT ファイルを変更する」をチェックしてください。

環境変数	設定例
C_INCLUDE	自動 (C_INCLUDE=C:\Program Files\IAR Systems\ew23\740\inc)
A740_INCLUDE	自動 (A740_INCLUDE=C:\Program Files\IAR Systems\ew23\740\inc)
XLINK_DFLTDIR	自動 (XLINK_DFLTDIR=C:\Program Files\IAR Systems\ew23\740\lib)
コマンドパス	自動 (“C:\Program Files\IAR Systems\ew23\740\bin”を追加)

### freeware に格納されているプログラムについて

製品 CD-ROM の freeware ディレクトリには、製品の機能を補足するプログラムが格納されています。これらのプログラムはインストーラではインストールしません。各プログラムを CD-ROM から直接インストールしてください。インストール方法および機能の詳細については各プログラムのドキュメントファイルを参照してください。

#### sc74

ルネサス製 740 ファミリ用 センブラパッケージで作成したソースファイルを ICC740 に付属のアセンブラ A740 のソースファイル形式に変換するソースファイルコンバータ

#### 注意事項

freeware は動作保証および保守は原則として行いません。本ソフトウェアを使用することにより、直接的、間接的に発生した損害については (株) ルネサステクノロジおよび (株) ルネサスソリューションズは一切の責任を負いません。

## 注意事項

### 割り込み処理に関する注意事項

割り込み処理実行中に呼び出す関数が、割り込み処理関数以外の関数からも呼び出される場合、リンク時に出力されるべき以下のウォーニング(\*)が出力されません。

V.1.00 Release 1 ~ V.1.00 Release 1B では出力されます。

```
Warning[w16]: Function "name" in module (file) is called from  
two function trees (with roots name1 and name2 )
```

\* M3T-ICC740 では、関数のローカル変数(関数引数および auto 変数)を静的に配置しています。そのため、ローカル変数を持つ関数を割り込み処理関数と、割り込み処理以外の関数の両方で使用するとローカル変数を破壊することがあります。リンカはこれに該当する関数呼び出しがある場合にこのウォーニングを出力する仕様ですが該当製品では出力しません。

発生例：

```
-----  
void func2( int );  
  
interrupt[2] void intr_1(void)  
{  
    func2( 2 );    /* func1() で、func2() を実行中に割り込みが発生 */  
                  /* すると、割り込みから復帰した func2() の      */  
                  /* ローカル変数が破壊される                      */  
}  
  
void func1( void )  
{  
    func2( 1 );  
}  
  
void main( void )  
{  
    func1();  
}  
-----
```

回避策：

割り込み処理関数と割り込み処理以外の関数で同じ関数を使用しないでください。

## 未定義関数に関する注意事項

未定義関数を呼び出した場合、リンク時に出力されるべき以下のエラーが出力されません。

```
Error[e46:] Undefined external "external" referred in module(file)
```

V.1.00 Release 1 ~ V.1.00 Release 1B では出力されます。

発生例：

```
-----  
void func3( int );  
  
void main( void )  
{  
    func3();      /* 未定義関数 */  
}  
-----
```

回避策：

未定義関数を呼び出していないかどうか確認してください。呼び出している場合はその関数を定義してください。

## 統合開発環境 High-performance Embedded Workshop と組み合わせて使用する場合の注意事項

### ソースファイルとヘッダ（インクルード）ファイルとの依存関係登録について

High-performance Embedded Workshop にて、ソースファイルとヘッダ（インクルード）ファイルとの依存関係を登録しているにも関わらず、登録内容が Build Options ダイアログボックス\*に表示されません。このため、登録内容の確認および修正ができません。また、既に登録されているファイルを 2 重に登録した場合には、依存関係のエラーが発生します。

\* High-performance Embedded Workshop の「ビルド」メニューから IAR ICC740 Toolchain」を選択するとオープンします。

回避策：

該当バージョンでは、依存関係の登録をしないでください。

ヘッダファイルのみを変更してリビルドする場合は「すべてをビルド」コマンドを実行してください。

---

## プロジェクトフォルダとリンク対象ファイル数について

リンク対象ファイル数が多くなるとリンク実行時に、High-performance Embedded Workshop が異常終了することがあります。

High-performance Embedded Workshop により、リンクが実行される際のコマンドライン文字数が 1024 文字を越える場合に発生します。

コマンドライン文字数とは、コマンド名、リンクオプション、およびリンク対象ファイル名の文字数を合計(スペースを含む)したものです。このとき、すべてのリンク対象ファイル名にはプロジェクトフォルダのディレクトリ名がフルパスで付加されます。

実際のコマンドライン文字数については、マップファイルに出力されたコマンドラインの情報を参照して確認してください。

発生例：

-----  
プロジェクトフォルダ

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop

リンク対象ファイル

cstartup.r31, main.r31, file1.r31, file2.r31, file3.r31, file4.r31, file5.r31,  
file6.r31, file7.r31, file8.r31, file9.r31, file10.r31, file11.r31

実際のコマンドライン

C:\¥Program Files¥IAR Systems¥ew23¥740¥bin¥xlink.exe

-f C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥lnk740.xcl

-l C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥develop.map

-x -Fieeee695 -ylmba -o

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥develop.695

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥cstartup.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥main.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file1.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file2.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file3.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file4.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file5.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file6.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file7.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file8.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file9.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file10.r31

C:\¥WorkSpace¥folder1¥folder2¥develop¥develop¥Debug¥file11.r31

-----

回避策：

以下の方法で回避してください。

(1) プロジェクトフォルダの名前を短く、かつ階層を浅くしてください。

例) C:\¥WorkSpace¥D1¥D1

(2) リンク対象のファイル数を少なくしてください。

## IAR Systems 社ドキュメントについて

CD-ROM の ICC740¥MANUAL フォルダに登録しています日本語マニュアルには、最新情報が記載されていません。最新情報については、英語版マニュアル等、下記ドキュメントを参照してください。ICC740 のインストール後生成される doc フォルダには、以下の英語ファイルが存在します。

a740. pdf	アセンブラ、リンカー、ライブラリアンのプログラミング・ガイド
cs740M. pdf	740 ファミリ用 C コンパイラパッケージでは参照不要です。
cw740. pdf	740 ファミリ用 C コンパイラパッケージでは参照不要です。
ew740. pdf	740 ファミリ用 C コンパイラパッケージでは参照不要です。
icc740. pdf	C コンパイラのプログラミング・ガイド 日本語版の C ライブラリリファレンス・ガイドとして jp¥clib_jp. pdf があります。
xlink. pdf	リンカー、ライブラリツールのリファレンス・ガイド 日本語版として jp¥xlink_jp. pdf があります。

a740. htm	A740 の最新情報（新機能、制限事項等）
CLibrary. htm l	C ライブラリの最新情報（新機能、制限事項等）
coding. htm	効率的なコーディング方法
cs740. htm	740 ファミリ用 C コンパイラパッケージでは参照不要です。
cs740m. htm	740 ファミリ用 C コンパイラパッケージでは参照不要です。
cwgui. htm	740 ファミリ用 C コンパイラパッケージでは参照不要です。
ewgui. htm	740 ファミリ用 C コンパイラパッケージでは参照不要です。
icc740. htm	ICC740 の最新情報（新機能、制限事項等）
manuals. htm	マニュアル補足
readme. htm	製品の最新情報（新機能、制限事項等）
wtd740. htm	740 ファミリ用 C コンパイラパッケージでは参照不要です。
xar. htm	XAR の最新情報（新機能、制限事項等）
xlink. htm	XLINK の最新情報（新機能、制限事項等）
xman. htm	XLINK のマニュアル補足

ICC740、A740、XLINK の最新情報の一部について和訳を提示します。

### icc740. htm, Important information の第 2 項

「割り込み可能な ISR (割り込み可能な割り込みサービスルーチン)」オプション (-h) を使用している場合は、C のランタイムライブラリを明示的に呼び出す C のソースコードでは、他の ISR からの割り込みが可能な ISR を使用しないでください! 詳細については、manuals. htm の -h オプションについての説明を参照してください。（次項に示します）

### manuaus. htm, IAR C Compiler Reference Guide の Support for interruptable ISRs 項

コード生成オプション「汎用割り込み可能サービス（コマンドラインオプション: -h) を使用すると、コンパイラは他の ISR からの割り込みが可能な ISR (割り込みサービスルーチン) を生成します。このオプションを使用してコードを生成するとき、ISR エントリコードは割り込みでないコンテキストからの場合のみ、式スタックを IES (割り込み式スタック) に切り替えます。つまり、ISR が他の ISR に対して割り込んだときも、IES が増加し続けるということです。

複数のレベルの割り込みを使用するには、割り込み先のサービスルーチン内での割り込みを可能にする必要があります。割り込みは、ヘッダファイル intr740. h で宣言する内蔵関数 enable\_interrupt() で有効にすることができます。enable\_interrupt() の呼出しは、ISR の中で割り込みを可能にする位置に配置してください。割り込み不可に戻す必要がある場合には、disable\_interrupt() を使用してください。

### 重要情報

他の ISR からの割込みが可能な ISR には、C のランタイムライブラリを明示的に呼び出す C ソースコードを含めないでください。これは、ランタイムライブラリは最適化技術を使用して構築されており、スタックポインタが常に最新のものとは限らないためです。ISR の割込みは IES を使用して、現在の IES のスタックポインタから一定のオフセットを置いた位置で開始されます。割込み先の ISR で、C ライブラリがスタックポインタより下位の IES を使用して動作中の場合、データが割込みを行う ISR によって上書きされることがあります。

割込み可能な ISR を使用すると、次のような動作が起こります。

コードが追加して生成されます。

各 ISR に対して 3 個の追加命令 (合計 6 バイト) が生成されます。これに加えて、IES に領域を割り当てるために、ネストされた各割込み (最初の割込みではなく) に対して 4 個の追加命令 (5 バイト) が生成されます。また変数 ?IES\_USAGE を初期化するために、出力される CSTARTUP には 2 個の追加命令 (合計 4 バイト) が追加されます。

追加変数が使用されます。

出力される CSTARTUP に変数 ?IES\_USAGE が追加され、RAM を 1 バイト使用します。

割込み可能な ISR を使用しないプロジェクトでは、追加変数 ?IES\_USAGE およびこれを初期化するコードを CSTARTUP から削除することができます。これによって、RAM 1 バイトおよび ROM 4 バイトを節約することができます。

## icc740.htm Important information の第 5 項

インラインアセンブラをコンパイラで使用することができます。このため、C のソースコード内のどの場所にもアセンブラ命令を挿入することができます。

ただし、インラインアセンブラ命令文を含むすべての関数の最適化は、非常に低いレベルでしか行われません。これは不正なコードが生成されることを防ぐためです。最適化する関数によっては、挿入されたコードが完全に異なる場所に配置されるようにコードの再配置が必要となる場合があるというのも理由の一つですが、最も重要な理由は、アセンブラ命令文によって発生する可能性がある副作用をコンパイラが識別することができないため、インラインアセンブラ命令文が含まれている場合には安全に最適化を行うことができないということです。詳細については、manuals.htm を参照してください。（次項に示します）

## manuals.htm, IAR C Compiler Reference Guide の Inline Assembly 項

インラインアセンブリを、コンパイラで使用できるようになりました。

### 構文

```
__asm("string with an assembler statement");
```

### 動作

アセンブラ命令文は、コンパイラが生成する他の任意のコードと同様に挿入されます。アセンブラ命令文は何らかのコードを生成する必要はなく、コメントを含むことはできません。

### 有効化

インラインアセンブラは、コンパイラがプロセッサ依存の言語拡張 (-e) を行うように設定されているときに有効になります。

### 例

```
__asm("LDA    #0");  
__asm("TAX   ;comments are allowed");  
__asm(" ; the statement does not have to do anything");
```

### 注意

インラインアセンブラを使用するときは、任意のインラインアセンブラ命令文を含む関数全体の最適化は、不正なコードを防ぐために非常に低いレベルでしか行われません。インラインアセンブラ命令文を挿入すると、一般的に副作用が発生します。コンパイラは挿入されたアセンブラ命令文で発生する可能性がある副作用を制御できないため、このコードを安全に最適化できません。

## icc740.htm, Important information の第 7 項

ゼロページポインタはサポートされています。

```
int zpage *ptr;
```

は、ゼロページ整数への 1 バイトのポインタを作成します。変数 ptr がデフォルトのメモリ領域に割り当てられています。

## icc740.htm, Important information の第 10 項

740 命令セットは、大きな整数型を使用した複合式を処理するときにはある程度の制限を受けます。可能であれば、符号なしの小さな整数型を使用してください。

## icc740.htm, Important information の第 11 項

ICC740 は、リエントラント関数をサポートしていません。再帰関数はサポートされていますが、実行時にオーバーヘッドが発生します。

**icc740.htm, Important information の第 12 項**

以下のコードを処理するときは、整数の格上げを確認してください。

```
char c, d, e, f;  
f = (c + d) / e;
```

ANSI C のリアルコンパイラ (ICC740 など) では、c および d を整数に格上げさせる必要があります。このため、コードが予測よりも大きくなる場合があります。キャストを挿入してコードを改良することができます (キャストを行っても正しいコードが作成される場合)。

```
f = (char)(c + d) / e;
```

ただし、「c + d」の結果が 255 を超える場合は同じ結果にはなりません。

**icc740.htm, Important information の第 14 項**

アプリケーションで D フラグ (BCD 演算) を使用している場合、C で記述された割込みルーチンは D フラグをクリアしないことに注意してください。BCD コードでは SEI/CLI のペアを使用するか、インライン関数 `cld_instruction` を最初の割込み時に使用します:

```
interrupt void my_interrupt (void)  
{  
    cld_instruction();  
    ...;  
}
```

**icc740.htm, New features の第 1 項**

新しい最適化により、コードのサイズが 1~3 パーセント小さくなります。

**icc740.htm, Known problems の第 2 項**

```
sfr P6 = 0x000014;  
unsigned char ubG_P26;  
  
void ie(void)  
{  
    P6.1 = ubG_P26;  
    ubG_P26 = 1 - ubG_P26;  
}
```

上記のコードでは、以下の内部エラーが発生します

```
In function: spill_reg_P20  
Diagnostic: nothing spilled
```

**icc740.htm, Known problems の第 3 項**

```
void test(void)
{
    __asm("NOP");
label:
    __asm("NOP");
    __asm("NOP");
    __asm("BRA *+2");
    goto label;
}
```

上記のコードでは、内部エラーが発生します:

```
In function: same_op_P51
Diagnostic: invalid EA
最適化レベルは -z8 以上。
```

**icc740.htm, Known problems の第 4 項**

アドレス 0x80B2 に関数 foo() があるものとみなします。次の記述で foo() を呼び出せます:

```
((void (*) (void)) 0x80B2)();
```

ただし、関数の呼出しはジャンプ先を見つけるのに間接アドレッシングを使用するため、

```
int address = 0x80B2;
((void(*) (void)) address)();
```

のような変数は、使用できません。

以下のように記述します:

```
int address = 0x80B2;
int *ap = &address;
((void(*) (void)) ap)();
```

**icc740.htm, Known problems の第 5 項**

アセンブラには行の長さに制限があるため、A740 を使用するためのアセンブリコードの生成によって問題が発生することがあります。DEFFN 文には呼び出されるすべての関数が記述されます。関数に多くの関数呼出しが含まれる場合は、アセンブラのソースコード行の最大長を超えることがあります。

この問題を解決するには、アセンブラの行の最大長を超えないように、C の関数に記述する関数の呼出しを少なくするか、短い関数名を使用してください。

**icc740.htm, Known problems の第 6 項**

再帰割込み関数を使用しないでください。実行時に初期化の問題が発生する場合があります。一般的に、必要な領域が多くなり実行時間も長くなるため、再帰割込み関数の使用は望ましくありません。

**a740.htm, Important information の第 3 項**

マクロ定義にファイルをインクルードすることはできません。

**a740. htm, Important information の第 4 項**

RSEG のセグメント型は、コードの生成に影響を与えず、リンクのチェックを行うだけです。このコードを動作させるには、新しい ZPAGE および NPAGE 命令を使用して、セグメント ラベルにデフォルトのサイズを設定します。

**a740. htm, Important information の第 5 項**

セグメント型では大文字と小文字が区別されます。したがって、「code」と「CODE」は違うものと認識され、アセンブラでエラーが発生します。

**a740. htm, Important information の第 6 項**

アセンブラ コードと C の形式によるプリプロセッサ コードを混用しないでください。C の言語とアセンブラの言語は異なります。ほとんどの C の形式によるプリプロセッサ命令に、対応するアセンブラ構成があります。コメントが C のコメントとは異なるため、特に#define 文の中でアセンブラコメントを使用するのは危険です。コメント全体が定義に含まれます。このため、警告なしに不正なコードが生成されることがあります。対応する C の形式のコメント「//」を代わりに使用してください。

```
#define aa 5 ; comment
```

SOURCE	PREPROCESSED	
lda #aa	lda #5 ; comment	Works as you expected
lda #aa+1	lda #5 ; comment+1	Load #5, wrong result!
lda #(aa+1)	lda #(5 ; comment+1)	Gives a syntax error

**a740. htm, Important information の第 8 項**

bit 命令を使用すると、不正な構文に対するエラーメッセージが表示されます。これは、740 命令セットに bit 命令があるためです。C コンパイラとアセンブラに共通するヘッダ ファイルで bit を使用するには、回避策として #ifndef を使用してください：

```
#ifndef __IAR_SYSTEMS_ASM
    bit declaration(s)
#endif
```

これによって C で bit 命令を参照することができますが、アセンブリ レベルでは参照できません。

**a740. htm, Known problems の第 1 項**

[740a0088]アセンブラは、次のコードに対して誤った出力を生成します。b\_start および b\_size はいずれも値 0 になります。

```
; Forward declare the 'SEG_EEP' segment.
RSEG SEG_EEP
```

```
RSEG CODE
PUBLIC b_start
PUBLIC b_size
```

```
b_start EQU SFB(SEG_EEP)
b_size EQU SIZEOF SEG_EEP
END
```

SFB または SIZEOF を使用できるようにするためにセグメントを事前に宣言する必要がありますが、このマニュアルでは、そのことについては説明していません (manuals. htm を参照のこと)。(次項に示します)

---

**manuals.htm, IAR Assembler Reference Guide の第 2 項 ([740a0088])**

RSEG または同様のセグメントが宣言された後、たとえば SFB または SIZEOF を使用することができるようになります。

**a740.htm, Known problems の第 2 項 ([740a0087])**

ラベルのアドレス計算が不正となり、「segment too long (max is ffff)」(「セグメントが長すぎます (最大長は ffff)」) というエラーが表示されることがあります。このエラーを発生させる範囲チェックは、安全のためにすべてのアドレスが通常のページアドレス内にあるときに実行されます。コードは正しく生成されます。

WA1: :zp 構文を使用して、次の例のように記号のサイズを宣言してください。

```
FOO DS 1
    LDA FOO:zp
    LABEL
```

WA2: 可能であれば、ラベルを削除してください。

WA3: モジュールをより小さな部分に分割してください。

**a740.htm, Known problems の第 4 項**

数行にまたがる DEFFN 宣言 (「¥」で連結) の場合、宣言内にエラーや警告がない場合は、最後の行だけがリストファイルに表示されます。これを防ぐためには、DEFFN 宣言全体を 1 行で記述してください。

**a740.htm, Known problems の第 5 項**

アセンブラのソース行の長さが制限されているため、コンパイラがアセンブラでは処理できないアセンブラコードを生成することがあります。その可能性がもっとも高いのは、C 関数で呼び出されるすべての関数を記述する DEFFN 文です。多数の関数がある (または関数名が長い) 場合、アセンブラが停止することがあります。解決策として、使用する関数の数を少なくしたり関数名が短くなるように C 関数を記述しなおし、全体の長さが行の最大長の制限以内になるようにしてください。

**a740.htm, Known problems の第 6 項**

関数名 (コンパイラで生成された関数も含む) には、アセンブラの予約語 (特にレジスタの名前) を使用することはできません。

**xlink.htm, Known problems in Current Version の第 2 項**

EW10554: [XLINK0119] 静的オーバーレイを行う場合、関数から間接的に、またはその他の場所から呼び出されたときにも XLINK は警告を出しませんが、割込み関数からの呼出しやメイン関数からの呼出しであった場合、危険です。

**xlink.htm, Known problems in Current Version の第 3 項**

EW10555: [XLINK0116] 特定の環境では、XLINK が空のセグメントを配置できないことがあります。特定の環境というのは、空のセグメントを配置するメモリ全体のアドレス範囲にすでに前のセグメント配置コマンドまたはアセンブラ モジュールによる絶対コードによってセグメントが配置されている場合を指します。ただし、その範囲より後ろのアドレス (バイトではありません) は空いています。

**xlink.htm, Known problems in Current Version の第 4 項**

EW10556: [XLINK0107] 前のセグメント配置コマンドによって配置された空のセグメントで、後のセグメント配置範囲が分割されることはありません。例: -ZEMPTY=400-4FF -ZBAR=0-FFFF。この記述によって、空のセグメント EMPTY がアドレス 400 に配置され、範囲 0-FFFF が 0-3FF と 400-FFFF に分割されます。

**xlink.htm, Known problems in Current Version の第 5 項**

EW10557: [XLINK0045] 静的オーバーレイ システム: 呼出し側に表れない関数 (警告 39) も、呼出し側の警告 16 (2 つの関数ツリーから呼び出される関数) の対象となるため、警告が 2 つ表示されます。

**xlink.htm, Known problems in Current Version の第 6 項**

EW12541: 参照されている関数および参照元のモジュールが常に正しい場合でも、リンカ リストファイル中の「参照元」情報が必ずしも正確であるとは限りません。リストファイルで宣言されている関数は、このような参照情報を持たない参照を行っている場合があります。正しい参照元は、通常は記述されている関数の (ソース ファイル内で) 近くに配置された関数です。

**マニュアル正誤表**

ページ		内容		
10	追加	表 1:コンフィグレーションとセッションの Configuration <table border="1" data-bbox="395 1025 1437 1070"> <tr> <td>Debug_740_E8_SYSYSTEM</td> <td>E8 システム用デバッグコンフィグレーション</td> </tr> </table>	Debug_740_E8_SYSYSTEM	E8 システム用デバッグコンフィグレーション
Debug_740_E8_SYSYSTEM	E8 システム用デバッグコンフィグレーション			
22	追加	表 4:コンフィグレーションとセッションの Configuration <table border="1" data-bbox="395 1126 1437 1171"> <tr> <td>Debug_740_E8_SYSYSTEM</td> <td>E8 システム用デバッグコンフィグレーション</td> </tr> </table>	Debug_740_E8_SYSYSTEM	E8 システム用デバッグコンフィグレーション
Debug_740_E8_SYSYSTEM	E8 システム用デバッグコンフィグレーション			

## 統合開発環境の起動

### High-performance Embedded Workshop の起動

Windows[スタート]→[プログラム]→[Renesas]→[High-performance Embedded Workshop]  
→[High-performance Embedded Workshop]メニューをクリックします。

### Manual Navigator の起動

Windows[スタート]→[プログラム]→[Renesas]→[High-performance Embedded Workshop]  
→[Manual Navigator]メニューをクリックします。

オンラインマニュアルおよび添付資料の参照ができます。

注意

Manual Navigator でマニュアルを表示するためには、Acrobat Reader が必要です。

Manual Navigator にマニュアルを登録した後、マニュアルのフォルダを移すとマニュアルを表示できなくなります。

## 動作確認環境

ICC740 の動作を実際に確認しているホストマシンおよび OS のバージョンについて以下に示します。

ホスト名	OS のバージョン	CD-ROM のディレクトリ
IBM PC/AT 互換機	Microsoft Windows 98 Microsoft Windows 2000 Microsoft Windows NT4.0 Microsoft Windows Me Microsoft Windows XP	W95J,W95E

## ソフトウェアのバージョン一覧

ICC740	V2.17A	IAR ANSI C Compiler
A740	V2.17A	IAR Assembler
CLIB	V3.34L	IAR C runtime library
XLINK	V4.59M	IAR Linker
XLIB	V3.29I	IAR Librarian

本製品には以下のソフトウェアが同梱されています。

SC74	V.1.00.00	ソースファイルコンバータ (技術サポート対象外ソフトウェア)
------	-----------	--------------------------------