

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 ====

M32C/90, M32C/80, M16C/80, M16C/70 シリーズ用 C コンパイラパッケージ  
**V.5.20 Release 02**  
 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  
 2005 年 6 月 1 日

### 概要

この度は M32C/90, M32C/80, M16C/80, M16C/70 シリーズ用 C コンパイラパッケージ V.5.20 Release 02 をお買い上げいただきましてありがとうございます。

本資料は C コンパイラパッケージの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。

|   |    |
|---|----|
| 1. 注意事項.....  | 3  |
| 1.1. コンパイラに関する注意事項.....   | 3  |
| 1.1.1. if-else 文に関する注意事項.....   | 3  |
| 1.1.2. 関数の実引数に const 修飾型オブジェクトのアドレスを渡す場合の注意事項.....                    | 4  |
| 1.1.3. inline 関数のネストに関する注意事項.....                                     | 4  |
| 1.1.4. コマンドオプション-I に関する注意事項.....                                      | 5  |
| 1.1.5. インクルードファイルの検索に関する注意事項.....                                     | 5  |
| 1.1.6. インラインアセンブル機能(#pragma ASM ~ #pragma ENDASM、asm 関数)に関する注意事項..... | 5  |
| 1.1.7. _Bool 型を使用しているプログラムのデバッグに関する注意事項.....                          | 5  |
| 1.1.8. 前処理命令#define に関する注意事項.....                                     | 5  |
| 1.2. アセンブラに関する注意事項.....   | 5  |
| 1.2.1. アセンブラオプション"-mode60"および"-mode60p"に関する注意事項.....                  | 5  |
| 1.3. 機種依存に関する注意事項.....  | 6  |
| 1.3.1. M16C の割り込み制御レジスタに関する注意事項.....                                  | 6  |
| 1.3.2. SFR 領域のアクセスに関する注意事項.....                                       | 6  |
| 1.4. High-performance Embedded Workshop に関する注意事項.....                 | 7  |
| 1.4.1. デバッグ機能について.....  | 7  |
| 1.4.2. MAP ビューワの起動方法について.....   | 7  |
| 1.5. TM に関する注意事項.....   | 7  |
| 1.6. MS-Windows に関する注意事項.....   | 7  |
| 1.6.1. 動作環境に関する注意事項.....  | 7  |
| 1.6.2. ファイル名に関する注意事項.....   | 7  |
| 1.6.3. ウィルスチェックプログラムに関する注意事項.....                                     | 8  |
| 1.6.4. バージョンアップするときの注意事項.....   | 8  |
| 2. C コンパイラパッケージのインストール.....   | 9  |
| 2.1. インストールを始める前に.....  | 9  |
| 2.2. インストール時の注意事項.....  | 9  |
| 2.3. インストーラ.....  | 9  |
| 2.4. インストール手順.....  | 10 |
| 2.5. インストール後の環境設定.....  | 10 |
| 3. ユーザ登録.....   | 10 |

|        |   |    |
|--------|---|----|
| 3.1.   | ユーザー登録方法.....   | 10 |
| 4.     | バージョンアップ内容.....   | 11 |
| 4.1.   | コンパイラ機能追加・改定.....   | 11 |
| 4.1.1. | エラー処理への変更.....  | 11 |
| 4.2.   | コンパイラ問題点修正.....   | 11 |
| 4.3.   | アセンブラ機能改定.....  | 11 |
| 4.3.1. | ID コードおよびROM プロテクトコード設定機能の改定.....                           | 11 |
| 4.4.   | アセンブラ問題点修正.....   | 11 |
| 5.     | ソフトウェアのバージョン一覧.....   | 12 |
| 6.     | M32C/90, M32C/80, M16C/80, M16C/70 シリーズ用リアルタイムの対応バージョン..... | 12 |
| 7.     | 標準関数ライブラリの MISRA C ルール適合に関して.....                           | 13 |
| 7.1.   | ルール違反の要因.....   | 13 |
| 7.2.   | ルール違反となった検査番号.....  | 13 |
| 7.3.   | 評価環境.....   | 13 |
| 8.     | TM から High-performance Embedded Workshop V.4 への移行の手引き.....  | 14 |
| 8.1.   | 概要.....   | 14 |
| 8.2.   | 変換手順.....   | 14 |
| 8.3.   | 注意事項.....   | 15 |
| 8.3.1. | 移行できる情報、できない情報.....   | 15 |
| 8.3.2. | クロスツール.....   | 16 |
| 8.3.3. | High-performance Embedded Workshop のバージョン.....              | 16 |
| 8.3.4. | ロードモジュールコンバータ.....  | 16 |
| 8.3.5. | 外部ツール.....  | 17 |
| 8.3.6. | リンク順序.....  | 20 |
| 8.3.7. | スタートアッププログラムの先頭リンク.....                                     | 20 |
| 9.     | 使用権許諾契約書.....   | 21 |

## 1. 注意事項

製品をご使用になる際は、以下の内容に従ってくださるようお願いいたします。

### 1.1. コンパイラに関する注意事項

#### 1.1.1. if-else 文に関する注意事項

- 内容
  - if文の制御式の結果が真または偽いずれの場合にもビットフィールドに定数を代入する文を記述していると誤ったコードを生成します。
- 発生条件
  - 以下の条件をすべて満たす場合に発生します。
  - (1) else を持つ if 文がある。
  - (2) (1)の制御式の結果が真または偽いずれの場合も同じ変数名を持つ構造体のビットフィールドに定数を代入している。
  - (3) (2)のビットフィールドのビット幅は、真または偽の場合に関わらずいずれも 1 ビットである。
  - (4) (2)のビットフィールドのビット位置は真または偽の場合で異なる。
  - (5) (2)の定数は真または偽の場合で異なる。
- 発生例

```

char    c;

struct S
{
    int    b0:1;        /* 発生条件(3) および (4) */
    int    b1:1;        /* 発生条件(3) および (4) */
    int    b2:1;
    int    b3:1;
    int    b4:1;
    int    b5:1;
    int    b6:1;
    int    b7:1;
    int    b8:8;
};

void    func(void)
{
    if (c == 1){        /* 発生条件(1) */
        s.b0 = 0;      /* 発生条件(2), (3), (4), および(5) */
    }
    else{               /* 発生条件(1) */
        s.b1 = 1;      /* 発生条件(2), (3), (4), および(5) */
    }
}

```

- 回避策
  - else のブロック中にダミーの asm()関数を挿入してください。

```

void    func( void )
{
    if ( c == 1 ){
        s.b0 = 0;
    }
    else{
        s.b1 = 1;
        asm();          /* ダミーの asm()関数を挿入 */
    }
}

```

### 1.1.2. 関数の実引数に const 修飾型オブジェクトのアドレスを渡す場合の注意事項

- 内容

実引数が const 修飾型オブジェクトのアドレスである関数呼び出しの記述に対し、正しい型の実引数を記述しても誤った警告メッセージが表示されます。なお、誤った警告が表示されても、生成されたコードに問題はありませ

- 発生条件

以下の条件をすべて満たす場合に発生します。

- (1) 呼び出される関数の仮引数が const 修飾型へのポインタ型である。
- (2) (1)の仮引数に対応する実引数に、以下のいずれかの値を記述している。
  - const 修飾型オブジェクトにアドレス演算子(&)を適用した結果
  - const 修飾型の配列の名称
- (3) (2)の const 修飾型オブジェクト、または const 修飾型の配列要素は、(1)の仮引数のポインタが指す型と同じ型を持つ。

- 発生例

```

const int  ten = 10;          /* 発生条件(3) */
const int  arr[2] = { 1, 2 }; /* 発生条件(3) */

void subr(const int *);      /* 発生条件(1)および(3) */

void mainr(void)
{
    subr(&ten);              /* 発生条件(2) */
    subr(arr);              /* 発生条件(2) */
}

メッセージ表示例
[Warning(ccom):example.c,line 8] assignment from const pointer to non-const pointer
====>    subr(&ten);
[Warning(ccom):example.c,line 9] assignment from const pointer to non-const pointer
====>    subr(arr);

```

- 回避策

誤って警告される実引数の前に、仮引数の型へ型変換するキャスト演算子を挿入してください。

```

void mainr(void)
{
    subr((const int *)&ten); /* キャスト演算子を挿入 */
    subr((const int *)arr);  /* キャスト演算子を挿入 */
}

```

### 1.1.3. inline 関数のネストに関する注意事項

- 内容

仮引数を持つ inline 関数をネストすると、誤った実引数(実引数と異なる変数)を参照する場合があります。

- 発生条件

以下の条件をすべて満たす場合に発生します。

- (1) inline 関数をネストしている。
- (2) 呼び出し元 inline 関数 A と呼び出し先 inline 関数 B の仮引数名が同一である。

● 発生例

```

inline      B(int aaa, char ccc)                /* 発生条件(2) */
{
    .....
}

inline      A(int c, int aaa, char *ccc)        /* 発生条件(2) */
{
    int      i;
    char     c;

    B(i,c);                                     /* 発生条件(1) */
}

```

● 回避策

次のいずれかの方法で回避してください。

- (1) 呼び出し先の関数（発生例では、inline 関数 B）の引数名を変更する。
- (2) inline 関数のネストをしない。
- (3) コンパイルオプション-Oforward\_function\_to\_inline(-OFFTI)を使用する。

#### 1.1.4. コマンドオプション-Iに関する注意事項

コマンドオプション"-I"で指定できるディレクトリの個数は最大 50 個です。

#### 1.1.5. インクルードファイルの検索に関する注意事項

#include の記述において、ドライブ名付きで記述しコンパイル対象となるファイルが存在するディレクトリとは異なったディレクトリからコンパイルした場合、インクルードファイルを検索できない場合があります。

#### 1.1.6. インラインアセンブル機能(#pragma ASM ~ #pragma ENDASM、asm 関数)に関する注意事項

- #pragam ASM ~ #pragma ENDASM 内の記述に対して、アセンブル及びリンク時のエラーメッセージの行数、デバッグ情報の行情報等が正常に出力されない場合があります。
- コンパイラは、レジスタや変数の有効範囲について、プログラムフローを解析して処理を行っているため、インラインアセンブル機能(#pragma ASM ~ #pragma ENDASM または asm 関数)でフローに影響を与えるようなブランチ（条件ブランチ含む）を記述しないようにしてください。
- インラインアセンブル機能を使用してレジスタの値を変更する記述をする場合、有効範囲中でレジスタの値を変更した情報を得ることができません。必ずレジスタを退避・復帰してください。

#### 1.1.7. \_Bool 型を使用しているプログラムのデバッグに関する注意事項

\_Bool 型を使用したプログラムをデバッグする場合、デバッガが \_Bool 型に対応しているかご確認ください。\_Bool 型に対応していないデバッガをご使用になる場合は、コンパイル時にデバッグオプション "-gBTC(-gbool\_to\_char)" をご使用ください。

#### 1.1.8. 前処理命令#define に関する注意事項

マクロ ULONG\_MAX と同一値になるマクロを定義する場合は、必ず接尾語 UL を付けてください。

## 1.2. アセンブラに関する注意事項

### 1.2.1. アセンブラオプション"-mode60"および"-mode60p"に関する注意事項

M16C/60,30,Tiny,20,10 および R8C/Tiny シリーズと機能の異なる下記命令については、命令の置き換えがされず、ワーニングメッセージも出力されません。

したがって、ソースプログラムにこれらの命令が記述されている場合、各命令の機能を確認してください。

---

 対象となる命令
 

---

SMOVB

SMOVF

RMPA

### 1.3. 機種依存に関する注意事項

#### 1.3.1. M16C の割り込み制御レジスタに関する注意事項

最適化オプション"-O5"を指定すると、ビット操作命令(BTSTC、BTSTS)を生成する可能性があります。BTSTC、BTSTS 命令は、M16C の割り込み制御レジスタを書きかえる命令として使用できません。

本オプションを使用する場合は、必ず生成されたコードに問題がない事をご確認ください。

- 発生例

以下のプログラムに対して最適化オプション"-O5"を指定した場合、最適化により BTSTC 命令を生成します。このため、割り込み要求ビットの判定が正しく行われず、意図しない動作をおこします。

```
#pragma ADDRESS TA0IC 006ch /* M16C/80 タイマ A0 割り込み制御レジスタ */
struct {
    char    ILVL : 3;
    char    IR   : 1; /* 割り込み要求ビット */
    char    dmy : 4;
} TA0IC;

void WaitUntillRisON(void)
{
    while( TA0IC.IR == 0) /* 1 になるまで待つ */
    {
        ;
    } /* 1 になったら 0 に戻す */
}
```

- 回避策

- (1) 該当する最適化オプションに加えてオプション"-Ono\_asmopt[-ONA]"を指定することにより、BTSTC、BTSTS 命令を生成する最適化を抑制してください。
- (2) 以下のように"asm 関数"を挿入することにより、最適化を抑制してください。

```
void WaitUntillRisON(void)
{
    while( TA0IC.IR == 0)
    {
        asm();
    }
}
```

- 注意

オプション"-Ono\_asmopt[-ONA]"、または asm 関数の使用による対策後は、必ず BTSTC、BTSTS 命令が生成されていない事を確認してください。

#### 1.3.2. SFR 領域のアクセスに関する注意事項

SFR 領域のレジスタをアクセスする場合には特定の命令を使用しなければならないことがあります。

この特定の命令は機種毎に異なりますので、詳しくは各機種のユーザーズマニュアルなどを参照してください。この注意事項に関わる命令は、asm 関数等のインラインアセンブル機能を使用して、プログラム中に命令を直接記述してください。



## 1.4. High-performance Embedded Workshop に関する注意事項

### 1.4.1. デバッグ機能について

C コンパイラパッケージ V.5.20 Release 02 用 High-performance Embedded Workshop は、シミュレータデバッグ機能、エミュレータデバッグ機能は使用できません。デバッグを行うときは、M32C/90, M32C/80, M16C/80, M16C/70 シリーズ用エミュレータデバッガやシミュレータデバッガをご用意ください。

### 1.4.2. MAP ビューワの起動方法について

M16C、R8C、M32C では、High-performance Embedded Workshop のメニューから MAP ビューワを起動することができません。MAP ビューワは次の手順で実行してください。

- (1) メニューの[基本設定] [カスタマイズ]をクリックしてください。
- (2) カスタマイズダイアログボックスのメニュータブから、追加ボタンをクリックしてください。
- (3) ツールの追加ダイアログボックスの各項目を指定します。

|          |   |
|----------|---|
| 名前       | MAPViewer (任意の名称)   |
| コマンド     | C:\¥Renesas¥NC308WA¥V520R02¥BIN¥MapViewer.exe<br>(コンパイルインストールディレクトリにある mapviewer.exe を指定してください) |
| 引数       | \$(CONFIGDIR)¥\$(PROJECTNAME).x30   |
| 初期ディレクトリ | \$(CONFIGDIR)   |

- (4) OK ボタンをクリックします。

[ツール]メニューに(2)で指定した名称が追加されます。この名称をクリックすることにより、MAP ビューワを起動してください。

## 1.5. TM に関する注意事項

- 統合化開発環境 TM を使用する場合、V.3.00 またはそれ以降のバージョンをご使用ください。C コンパイラパッケージ V.5.20 Release 02 は、V.2.01 以前の TM ではご使用になれませんので、ご注意ください。
- TM V.3.00 以降の使用において、TM V.2.01 以前のバージョンで作成したプロジェクトを流用する場合に、コンパイルオプション、アセンブルオプションの“-finfo”が有効になりません。個別に“-finfo”オプションを追加してください。詳しくは、TM のリリースノートを参照してください。

## 1.6. MS-Windows に関する注意事項

### 1.6.1. 動作環境に関する注意事項

- (1) C コンパイラパッケージは、Windows 98、Windows NT 4.0 以降の環境で動作します。Windows 3.1 および Windows NT 3.5x 以前のバージョンでは動作しません。
- (2) 日本語 Windows NT 環境でコマンドプロンプトのサイズが「80 x 25」以外に設定されている場合、製品を起動するとコマンドプロンプトのサイズが頻繁に切り替わります。コマンドプロンプトのサイズは「80 x 25」に設定してください。

### 1.6.2. ファイル名に関する注意事項

ソースプログラムファイルの名前や作業を行うディレクトリ名は、次の注意事項に従ってください。

- 漢字を含むディレクトリ名、ファイル名は使用できません。
- ファイル名に使用するピリオド(.)は一つのみ使用可能です。
- ネットワークパス名は使用できません。ドライブ名に割り当ててご使用ください。
- 「ショートカット」は使用できません。
- 空白文字を含むディレクトリ名、ファイル名は使用できません。
- "..."表記を用いて2つ以上のディレクトリを指定することはできません。
- パス指定を含めたファイル名の長さが128文字以上になるものは使用できません。

### 1.6.3. ウィルスチェックプログラムに関する注意事項

ウィルスチェックプログラムが常駐した状態で C コンパイラパッケージを起動すると正常に起動しない場合があります。その場合は、ウィルスチェックプログラムの常駐を解除してから C コンパイラパッケージを起動しなおしてください。

### 1.6.4. バージョンアップするときの注意事項

C コンパイラパッケージ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 C コンパイラパッケージ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。

- C コンパイラパッケージのアンインストール手順

C コンパイラパッケージ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。

## 2. C コンパイラパッケージのインストール

### 2.1. インストールを始める前に

インストールを始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」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。
- C コンパイラパッケージを快適に使用するには、32M バイト以上のメモリと 20M バイト以上の空きハードディスク領域が必要です。
- 製品のインストールは専用のインストーラを使用してください。
- 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。インストールを始める前にライセンス ID を確認してください。

### 2.2. インストール時の注意事項

- 本製品をインストールする場合、コンパイラと一緒に統合開発環境 High-performance Embedded Workshop も必ずインストールされます。
- インストール中に下記のメッセージが記載されたダイアログが出力された場合は、「すべてはい」を選択してください。

[ダイアログメッセージ]  
 次のファイルは既にシステムにインストールされています。  
 C:¥WINDOWS¥TEMP¥...  
 ファイルを上書きしてもよろしいですか?  
 :

### 2.3. インストーラ

インストーラは次に示す環境毎に用意されています。ご購入になった製品を確認の上、該当するインストーラを使用してください。

- 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 |
|----------------------------|---|-----------|-----------------|
| IBM <sup>1</sup> PC/AT 互換機 | Microsoft Windows <sup>2</sup> 98<br>Microsoft Windows Me<br>Microsoft Windows NT<br>Microsoft Windows 2000<br>Microsoft Windows XP | SETUP.EXE | ¥NC308WA¥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 |
|---------------|--|-----------|-----------------|
| IBM PC/AT 互換機 | Microsoft Windows 98<br>Microsoft Windows Me<br>Microsoft Windows NT<br>Microsoft Windows 2000<br>Microsoft Windows XP | SETUP.EXE | ¥NC308WA¥W95E   |

<sup>1</sup> IBM および AT は、米国 International Business Machines Corporation の登録商標です。

<sup>2</sup> Microsoft、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。

## 2.4. インストール手順

C コンパイラパッケージは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。

## 2.5. インストール後の環境設定

インストールが完了した後、次の環境変数を設定してください。

表中の「自動」は、インストーラが AUTOEXEC.BAT を書きかえます。したがって、デフォルトでインストールを実行した場合は、AUTOEXEC.BAT を書きかえる必要はありません。

| 環境変数   | 設定例  |
|--------|--|
| BIN308 | 自動 (SET BIN308=C:\¥RENESAS¥NC308WA¥V520R02¥BIN)    |
| INC308 | 自動 (SET INC308=C:\¥RENESAS¥NC308WA¥V520R02¥INC308) |
| LIB308 | 自動 (SET LIB308=C:\¥RENESAS¥NC308WA¥V520R02¥LIB308) |
| TMP308 | 自動 (SET TMP308=C:\¥RENESAS¥NC308WA¥V520R02¥TMP)    |
| NCKIN  | SET NCKIN=SJIS                                     |
| NCKOUT | SET NCKOUT=SJIS                                    |
| コマンドパス | 自動 (C:\¥RENESAS¥NC308WA¥V520R02¥BIN を追加)           |

## 3. ユーザ登録

バージョンアップ情報や技術サポートなどのサービスを受けるためにユーザー登録を行ってください。ユーザー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。

また、ユーザー登録は**ご購入後 30 日以内**に行ってくださいようお願い申し上げます。

### 3.1. ユーザー登録方法

C コンパイラパッケージをインストールすると以下のファイルが生成されます。

```
¥ renesas¥nc308wa¥v520r02¥ support ¥ nc308wa ¥ regist.txt
```

¥ renesas¥nc308wa¥v520r02 はデフォルトでインストールした場合のディレクトリです。

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
```

## 4. バージョンアップ内容

### 4.1. コンパイラ機能追加・改定

#### 4.1.1. エラー処理への変更

“mismatch prototype parameter type.” のワーニングをエラーに変更しました。

### 4.2. コンパイラ問題点修正

ツールニュースでお知らせしました以下の問題点を修正しました。

- long long 型変数または long long 型定数の除算および剰余算に関する注意事項
- デバッグ情報に関する注意事項
- if 文の条件が真の場合と偽の場合とで同一の関数を呼び出す場合の注意事項
- ?を 3 個以上続けて書く場合の注意事項
- MISRA C ルールチェッカ SQMlint 用オプション使用に関する注意事項
- 配列に含まれるビットフィールドメンバの読み出しを行う場合の注意事項
- near および far 修飾子に関する注意事項
- long long 型変数の乗算に関する注意事項
- \_Bool 型の戻り値を持つ inline 関数に関する注意事項
- inline 関数に関する注意事項
- コンパイルオプション-Oforward\_function\_to\_inline (-OFFTI) に関する注意事項
- 依存関係の更新機能強化ツール使用時の注意事項
- typedef を使用して構造体を返す関数を使用する場合の注意事項
- 浮動小数点型変数と定数値との乗算文、加算文、または減算文が連続している場合の注意事項

### 4.3. アセンブラ機能改定

#### 4.3.1. ID コードおよびROM プロテクトコード設定機能の改定

- アセンブラ指示命令".ID"で設定された ID コードはアブソリュートモジュールファイルに出力されます。
- アセンブラ指示命令(.ID, .PROTECT) と lmc308 のコマンドオプション(-ID, -protect1, -protect2, -protectx) の優先順位を変更しました。 lmc308 のコマンドオプションが優先されます。

### 4.4. アセンブラ問題点修正

ツールニュースでお知らせしました以下の問題点を修正しました。

- リンクオプション"-JOPT"に関する注意事項
- アセンブラ指示命令".ID"および".PROTECT"に関する注意事項

## 5. ソフトウェアのバージョン一覧

C コンパイラパッケージ V.5.20 Release 02 に含まれているソフトウェアの各バージョンは以下のとおりです。

- nc308 V.1.05.06.000
- cpp308 V.4.05.04
- ccom308 V.5.02.11.000
- aopt308 V.1.02.00
- as308 V.4.01.00.000
- mac308 V.2.21.00.000
- pre30 V.1.10.12
- asp308 V.4.00.00
- ln308 V.4.02.00.000
- lb308 V.2.01.00
- lmc308 V.3.00.00
- xrf308 V.2.01.00
- abs308 V.1.02.00
- stk V.1.00.04
- utl308 V.1.00.10
- Stk Viewer V.1.00.01
- MapViewer V.3.00.00

## 6. M32C/90, M32C/80, M16C/80, M16C/70 シリーズ用リアルタイムの対応バージョン

本製品は、リアルタイム OS V.1.10 Release 1A に対応しています。本製品と組合せてリアルタイム OS をご使用になる場合には、リアルタイム OS V.1.10 Release 1A 以降にリリースされたバージョンをご使用ください。

## 7. 標準関数ライブラリの MISRA C ルール適合に関して

C コンパイラパッケージの標準関数ライブラリの C ソースコードは、MISRA C ルールに対して 52 のルール違反<sup>3</sup>が認められますが、これらの違反は動作に支障がありません。

### 7.1. ルール違反の要因

C コンパイラパッケージの標準関数ライブラリの C ソースコードにおいて、ルール違反となった主な要因は次の通りです。

- C コンパイラの仕様（near/far 修飾、asm()関数、#pragma）
- ANSI 規格に基づく関数の宣言
- 条件文における評価順序をカッコ()により明示的に記述していない
- 暗黙の型変換

### 7.2. ルール違反となった検査番号

ルール違反になった検査番号は次のとおりです。

|     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1   | 12  | 13  | 14  | 18  | 21  | 22  | 28  | 34  | 35  |
| 36  | 37  | 38  | 39  | 43  | 44  | 45  | 46  | 48  | 49  |
| 50  | 54  | 55  | 56  | 57  | 58  | 59  | 60  | 61  | 62  |
| 65  | 69  | 70  | 71  | 72  | 76  | 77  | 82  | 83  | 85  |
| 99  | 101 | 103 | 104 | 105 | 110 | 111 | 115 | 118 | 119 |
| 121 | 124 |     |     |     |     |     |     |     |     |

### 7.3. 評価環境

|                 |   |
|-----------------|---|
| コンパイラ           | C コンパイラパッケージ V.5.20 Release 1                                     |
| コンパイルオプション      | -O -c -as308 "-DOPTI=0" -gnone -finfo -fNII -misra_all -r \$*.csv |
| MISRA C ルールチェッカ | MISRA C ルールチェッカ V.1.00 Release 1A                                 |

<sup>3</sup> MISRA C ルールチェッカ SQMLint による検査結果値です。

## 8. TM から High-performance Embedded Workshop V.4 への移行の手引き

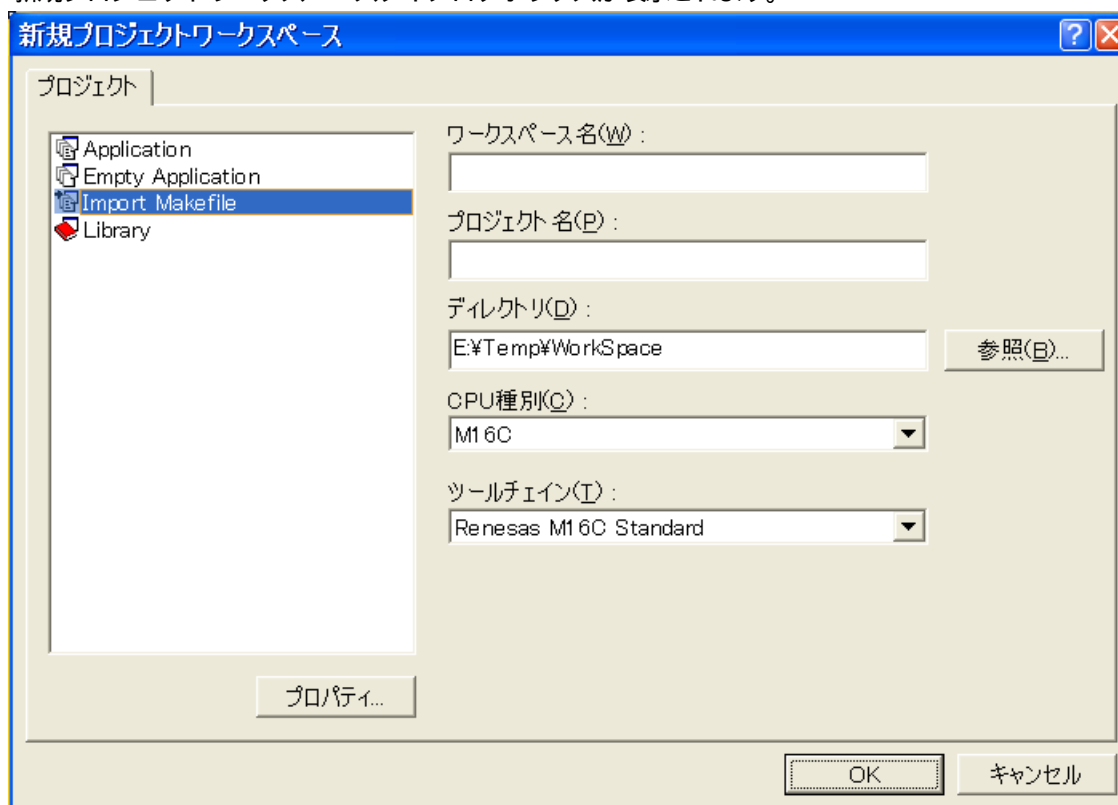
### 8.1. 概要

統合化開発環境 TM V.2.xx、V.3.xx で作成したプロジェクトを High-performance Embedded Workshop V.4 環境へ移行するには、High-performance Embedded Workshop の Import Makefile 機能を使用します。「Import Makefile」は、指定された makefile に書かれているソースファイルやオプション情報からプロジェクトを作成する機能です。TM のプロジェクトファイルは、GNU make から実行可能である makefile フォーマットで作成されています。「Import Makefile」にて、TM のプロジェクトファイルを makefile として指定することで、TM のプロジェクトを High-performance Embedded Workshop のプロジェクトへ変換することができます。「Import Makefile」では、TM のプロジェクトファイル以外に hmake、nmake、gmake 用の makefile フォーマットのファイルを High-performance Embedded Workshop のプロジェクトに変換することができます。

### 8.2. 変換手順

以下に、TM のプロジェクトを High-performance Embedded Workshop のプロジェクトへ移行する手順を示します。

- (1) メニュー [ファイル] [新規ワークスペース] をクリックします。
- (2) 新規プロジェクトワークスペースダイアログボックスが表示されます。



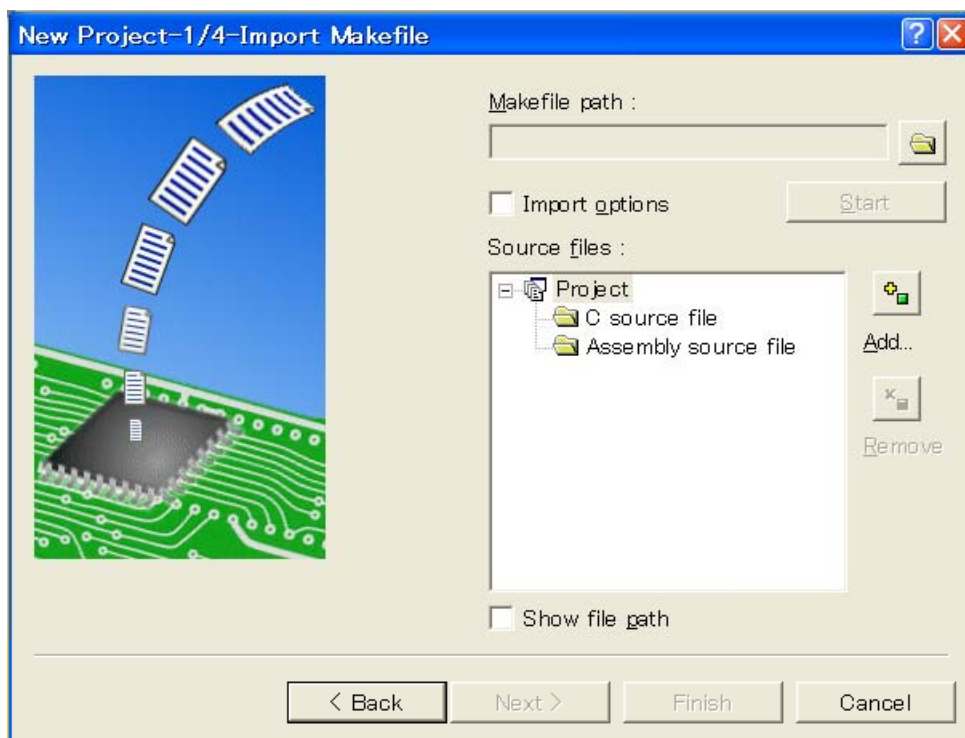
- (1) CPU 種別を選択します。TM のプロジェクトで使用していた CPU 種別を選択してください。
- (2) ツールチェーンを選択します。ツールチェーン名とクロスツール名の対応は以下の通りです。TM のプロジェクトで使用していたツールチェーン (クロスツール) を選択してください。

| ツールチェーン名              | クロスツール名 |
|-----------------------|---------|
| Renesas M16C Standard | NC30WA  |
| Renesas R8C Standard  | NC8C    |
| Renesas M32C Standard | NC308WA |
| Renesas M32R Standard | CC32R   |

- (3) プロジェクトタイプから Import Makefile を選択します。



- (4) ディレクトリを指定します。
  - (5) ワークスペース名を指定します。ワークスペース名を指定すると自動的に（ワークスペースと同名の）プロジェクト名が指定されます。
  - (6) OK ボタンをクリックします。
- (3) New Project-1/4-Import Makefile ウィザードダイアログボックスが表示されます。



- (1) 「Import options」をチェックします。  
この項目をチェックすると、ビルド（コンパイラ、アセンブラ etc）オプション情報が High-performance Embedded Workshop プロジェクトへ移行されます。チェックをはずすと、オプション情報は無視されません（High-performance Embedded Workshop プロジェクトへ移行されません）。
  - (2) Makefile path にファイルが指定されるとすぐに指定ファイルの解析作業が行われます。解析が終了すると解析したソースファイルが Source files ツリーに表示されます。  
「Start」ボタンをクリックすると、再度、指定ファイルの解析作業が行われます。
  - (3) 解析結果（Source files ツリー）に誤りがある場合は、Add...、Remove ボタンから Source files ツリーを編集してください。
  - (4) Next ボタンをクリックします。
- (4) 以降はダイアログボックスの指示に従って作業を進めてください。

### 8.3. 注意事項

#### 8.3.1. 移行できる情報、できない情報

TM のプロジェクトを High-performance Embedded Workshop 環境へ移行する場合、TM のプロジェクトの全構成を移行できるわけではありません。移行できる情報は、以下の通りです。

- (1) アセンブラソースファイルパス
- (2) C 言語ソースファイルパス
- (3) アセンブラオプション
- (4) C コンパイラオプション
- (5) リンカオプション（リンク順序を除く）

その他の情報は High-performance Embedded Workshop 環境へ移行することができません。移行できない情報は、「Import Makefile」の処理終了後、これ以降に示す注意事項の通りに High-performance Embedded Workshop プロジェクトを編集してください。

### 8.3.2. クロスツール

「Import Makefile」では、クロスツールのバージョンを High-performance Embedded Workshop プロジェクトへ移行することができません。よって、TM のプロジェクトで使用していたクロスツールのバージョンにかかわらず、High-performance Embedded Workshop プロジェクト移行後に使用可能なクロスツールのバージョンは以下のものになります。

|         |   |
|---------|---|
| NC30WA  | V.5.20 Release1、V.5.30 Release1、V.5.30 Release 02 |
| NC8C    | V.5.30 Release1                                   |
| NC308WA | V.5.20 Release1                                   |

### 8.3.3. High-performance Embedded Workshop のバージョン

TM のプロジェクトを High-performance Embedded Workshop 環境へ移行する場合、移行先の High-performance Embedded Workshop のバージョンにより移行できる情報が異なります。High-performance Embedded Workshop のバージョンによる移行可能情報は以下の通りです。

|         |                   | High-performance Embedded Workshop |           |           |           |        |
|---------|-------------------|------------------------------------|-----------|-----------|-----------|--------|
|         |                   | ~V.3.01.02                         | V.3.01.04 | V.3.01.05 | V.3.01.06 | V.4.00 |
| NC30WA  | V.5.20 Release 1  |                                    |           |           |           |        |
|         | V.5.30 Release 1  |                                    |           |           |           |        |
|         | V.5.30 Release 02 | --                                 | --        | --        | --        |        |
| NC8C    | V.5.30 Release 1  |                                    |           |           |           |        |
| NC308WA | V.5.20 Release 1  |                                    |           |           |           |        |

：アセンブラソースファイル、C 言語ソースファイルおよびアセンブラ、C コンパイラ、リンカのオプション移行可能

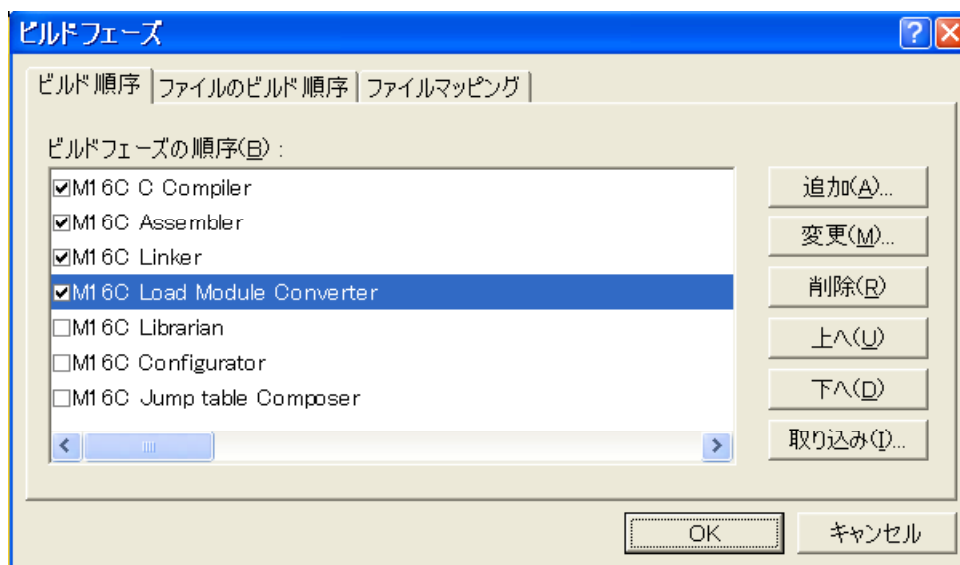
：アセンブラソースファイル、C 言語ソースファイルのみ移行可能

High-performance Embedded Workshop4 がバンドルされたコンパイラ製品から順次 になります。

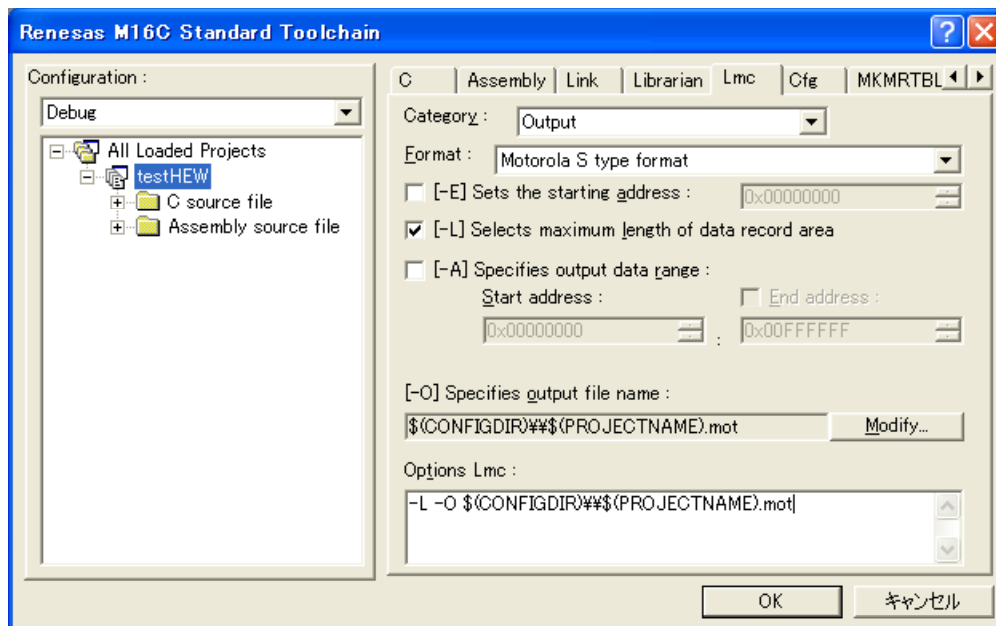
### 8.3.4. ロードモジュールコンバータ

「Import Makefile」では、ロードモジュールコンバータの情報(コマンド実行、オプション情報)を High-performance Embedded Workshop プロジェクトへ移行することができません。TM のプロジェクトでロードモジュールコンバータを使用していた場合は、「Import Makefile」の処理終了後、以下の手順でロードモジュールコンバータの設定を行ってください。

- (1) メニュー [ビルド] [ビルドフェーズ] をクリックします。
- (2) ビルドフェーズダイアログボックスが表示されます。



- (1) Mxxx Load Module Converter をチェックします。
- (2) OK ボタンをクリックします。
- (3) メニュー [ビルド] [Renesas Mxxx Standard Toolchain...] をクリックします。
- (4) Renesas Mxxx Standard Toolchain ダイアログボックスが表示されます。



- (1) Lmc タブをクリックします。
- (2) Category を変更してオプションを指定します。
- (3) OK ボタンをクリックします。

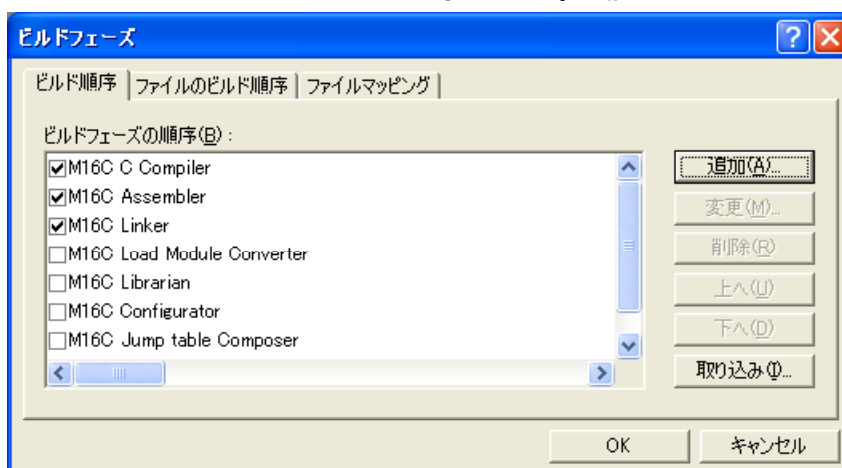
### 8.3.5. 外部ツール

「Import Makefile」では、アセンブラ、C コンパイラ、リンカ以外のツールの情報（コマンド実行、オプション情報、依存関係）を High-performance Embedded Workshop プロジェクトへ移行することができません。TM のプロジェクトで、アセンブラ、C コンパイラ、リンカおよびロードモジュールコンバータ以外のツールを使用していた場合は、High-performance Embedded Workshop のカスタムビルドフェーズを作成していただく必要があります。カスタムビルドフェーズは、標準のビルド実行（アセンブラ、C コンパイラ、リンカ）の前後または途中で外部ツールを実行するための独自のビルドフェーズです。

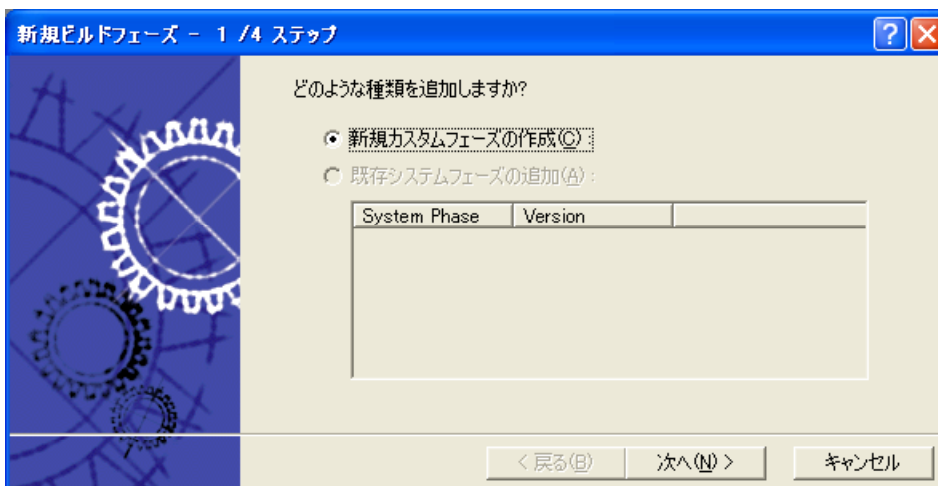
カスタムビルドフェーズ作成手順についての詳細は、High-performance Embedded Workshop4 ユーザーズマニュアル「3.2 カスタムビルドフェーズを作成する」をご覧ください。

ここでは、例としてクロスツールにバンドルされている xrf30 を登録する方法を示します。

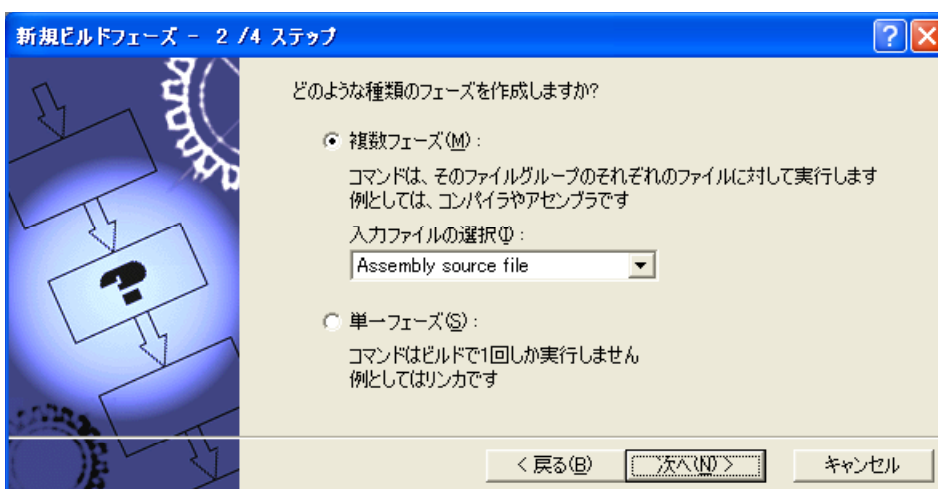
- (1) メニュー [ビルド] [ビルドフェーズ] をクリックします。
- (2) ビルドフェーズダイアログボックスが表示されます。追加ボタンをクリックします。



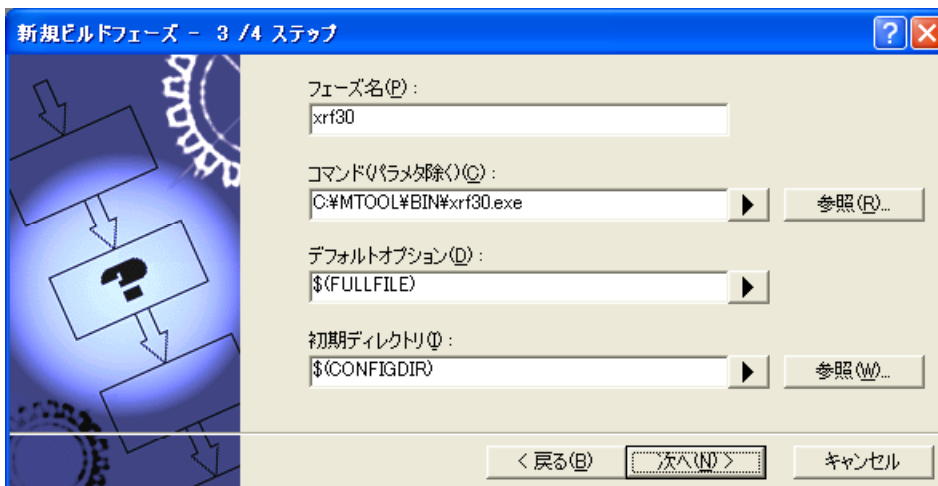
- (3) 新規ビルドフェーズダイアログボックスが表示されます。ウィザードに従ってツールを登録します。



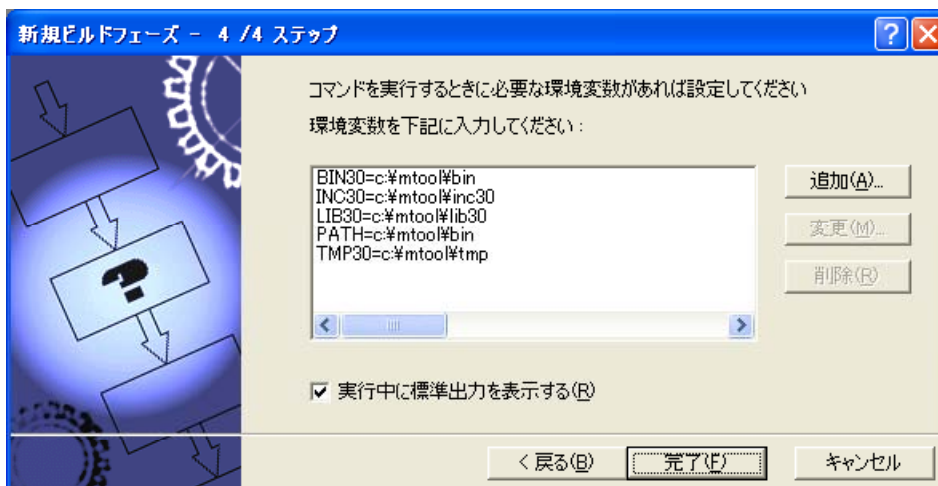
- (1) [1/4 ステップ] 次へボタンをクリックします。



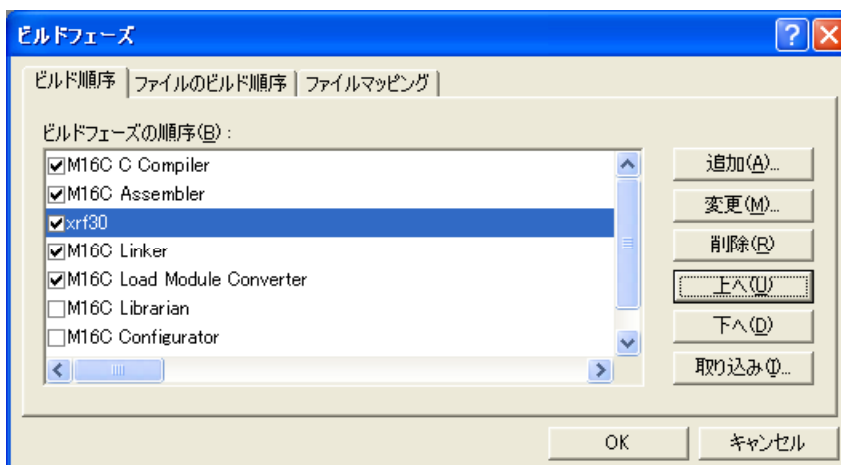
- (2) [2/4 ステップ] 複数フェーズを選択します。  
 (3) [2/4 ステップ] 入力ファイルの選択から Assembly source file を選択します。  
 (4) [2/4 ステップ] 次へボタンをクリックします。



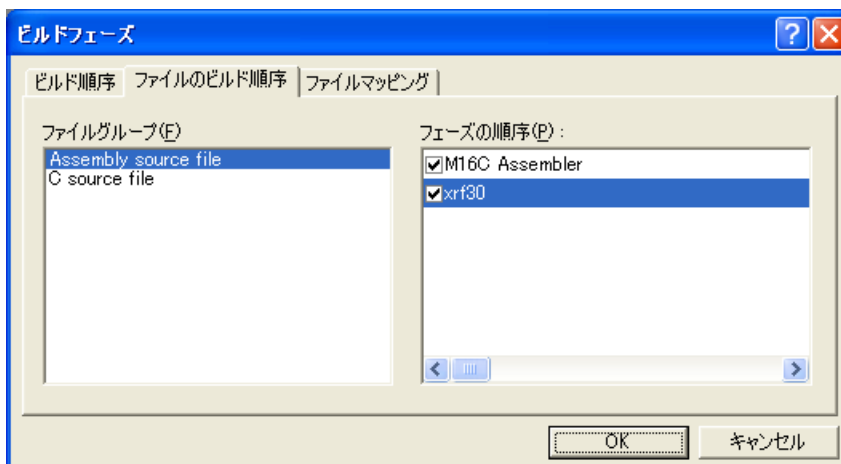
- (5) [3/4 ステップ] フェーズ名に xrf30 を、コマンドに xrf30 のフルパスを指定します。  
 (6) [3/4 ステップ] 次へボタンをクリックします。



- (7) [ 4/4 ステップ ] 環境変数を指定します。
- (8) [ 4/4 ステップ ] 実行中に標準出力を表示するをチェックします。
- (9) [ 4/4 ステップ ] 完了ボタンをクリックします。
- (4) ビルドフェーズダイアログボックスに戻ります。  
登録したビルドフェーズ ( xrf30 ) がリストの最後に追加されます。



- (1) xrf30 を選択します。
- (2) 上へボタンをクリックします。Assembler の下まで xrf30 を移動します。
- (3) ファイルのビルド順序タブをクリックします。



- (4) xrf30 をチェックします。
- (5) OK ボタンをクリックします。
- (5) メニュー [ ビルド ] [ xrf30 ] をクリックします。
- (6) xrf30 Options ダイアログボックスが表示されますので、必要に応じてオプションなどを設定してください。  
この設定を行うと、ビルド時のアセンブラ実行後（リンカの実行前）に xrf30 がすべてのアセンブラソースファ

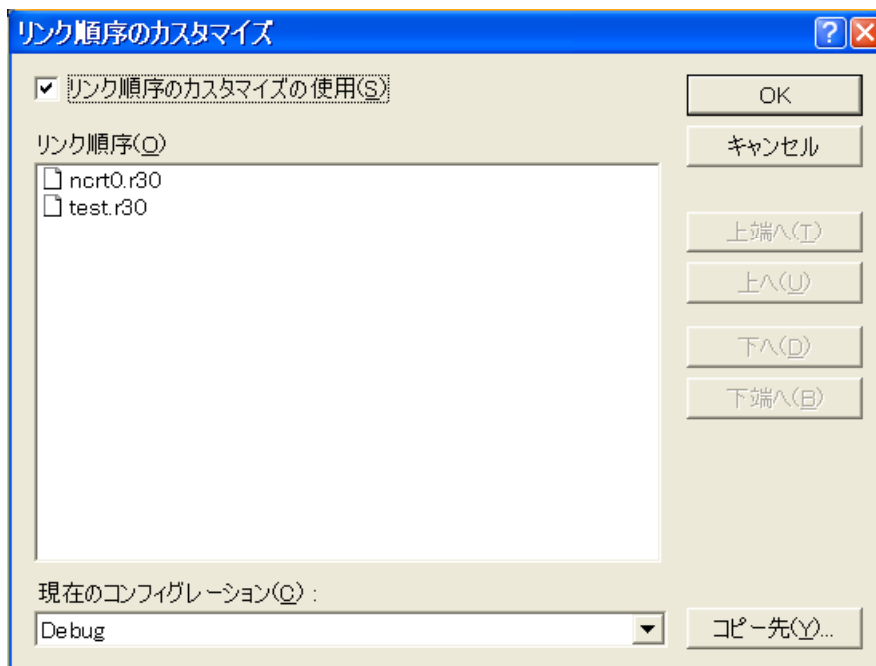
イルに対し実行されるようになります。

### 8.3.6. リンク順序

「Import Makefile」では、リンク順序の情報を High-performance Embedded Workshop プロジェクトへ移行することができません。リンク順序は、アルファベット順となります。リンク順序を変更する場合は、以下の手順で設定してください。

本機能は、High-performance Embedded Workshop4 がバンドルされたコンパイラ製品からの対応となります。

- (1) メニュー [ビルド] [リンク順の指定] をクリックします。
- (2) リンク順序のカスタマイズダイアログボックスが表示されます。



- (1) リンク順序のカスタマイズの使用をチェックします。
- (2) リンク順序リストからファイルを選択して、上へまたは下へボタンをクリックします。
- (3) OK ボタンをクリックします。

### 8.3.7. スタートアッププログラムの先頭リンク

「Import Makefile」では、リンク順序の情報を High-performance Embedded Workshop プロジェクトへ移行することができません。リンク順序は、アルファベット順となります。そのためスタートアップが先頭にリンクされないことがあります。スタートアッププログラムを先頭にリンクするには、「リンク順序」の項の設定を行ってください。

「3.6 リンク順序」に対応していないバージョンのコンパイラは、以下の手順で設定してください。

- (1) メニュー「ビルド」「Renesas XXX Standard Toolchain...」をクリックします。("Renesas XXX Standard Toolchain"ダイアログボックスが開きます)
- (2) 「link」タブをクリックします。
- (3) 「Category:」から、「Input」を選択します。
- (4) 「Show entries for:」から、「Relocatable files」を選択します。
- (5) 「Add」ボタンをクリックします。("Add Relocatable files"ダイアログボックスが開きます)
- (6) 「Relative to:」から、「Configuration directory」を選択します。
- (7) 「File path:」にスタートアッププログラム名(例: NC30WA の場合、ncrt0.r30)を入力します。
- (8) 「Add Relocatable files」ダイアログボックスの[OK]ボタンをクリックします。
- (9) 「Renesas XXX Standard Toolchain」ダイアログの[OK]ボタンをクリックします。

## 9. 使用権許諾契約書

許諾複製部数： 1

### 重要 - 以下のソフトウェア使用権許諾契約書を注意してお読みください。

本ソフトウェアは、お客様がこの「ソフトウェア使用権許諾契約書」にご同意頂いた場合にのみご使用頂けます。本契約書にご同意頂けない場合は、マスターディスクの梱包シールを解かずに販売店にご返却くだされば、代金をお返しいたします。

## 「ソフトウェア使用権許諾契約書」

お客様（以下、「甲」といいます）と株式会社ルネサス テクノロジ（以下、「乙」といいます）とは、この「ソフトウェア使用権許諾契約書」（以下、「本契約」といいます）とともに提供されるソフトウェア及びそのマニュアルにつき、以下の通り契約するものとします。

### 1. 用語の定義

- (1) 「許諾ソフトウェア」とは、本製品に含まれるコンパイラ、アセンブラおよび関連する実行プログラム及び「ライブラリ」並びにそのマニュアルをいいます。
- (2) 「指定システム」とは、甲が許諾ソフトウェアをインストールし、使用するコンピュータシステムをいいます。ネットワークを使用している場合には、許諾ソフトウェアに含まれる実行プログラムを実行する中央処理装置を持つコンピュータシステムをいいます。
- (3) 「許諾複製部数」とは、本契約書上部に記載された部数であって、乙から甲に対してマスターディスクの複製および「指定システム」へのインストールを許諾する範囲をいいます。
- (4) 「ライブラリ」とは、基本入出力機能又は文字判定機能等を有した関数プログラム群であって、リロケータブルオブジェクトプログラムで提供されるものをいいます。
- (5) 「開発ソフトウェア」とは、甲が作成するソフトウェアであって、ライブラリの全部若しくは一部が組み込まれたものをいいます。

### 2. ライセンスの許諾

乙は甲に対し、以下の譲渡不可の非独占的権利を許諾します。

- (1) 許諾ソフトウェアを許諾複製部数の範囲内の指定システムで使用し、開発ソフトウェアを作成し、株式会社ルネサス テクノロジが開発したマイクロコンピュータを搭載したシステム（以下「甲システム」といいます）に組み込むこと
- (2) 本契約第2条(1)及び(4),(5)のために指定システムにインストーラを用いて許諾複製部数の範囲内で許諾ソフトウェアの複製を持つこと
- (3) 許諾複製部数の範囲内でマスターディスクの複製を持つこと
- (4) 開発ソフトウェアとライブラリを結合すること
- (5) 開発ソフトウェアを複製し、甲システムに組み込み製造および販売すること
- (6) 本契約第2条(1)のために許諾ソフトウェアに含まれる電子マニュアルをプリンタ装置で出力すること

### 3. 制限

- (1) 甲は、本契約による使用権を譲渡したり、その他第三者に許諾ソフトウェアを使用させることはできません。

- (2) 甲は、許諾ソフトウェアおよびその複製物に含まれている著作権表示を取り除いてはなりません。
- (3) 甲は許諾ソフトウェアをレンタルまたはリースすることはできません。
- (4) 甲は、許諾ソフトウェアをリバースエンジニアリング、逆コンパイル、逆アセンブルすることはできません。

#### 4. 許諾ソフトウェアの権利

- (1) 許諾ソフトウェアの著作権はすべて乙に帰属するものであり、本契約のいかなる条項も、乙にかかる著作権の全部又は一部を譲渡するものではありません。
- (2) 許諾ソフトウェアの二次的著作物である開発ソフトウェアの著作権は、甲に帰属します。ただし、開発ソフトウェアに含まれるライブラリの著作権は乙に留保されます。

#### 5. 秘密保持

- (1) 甲は、許諾ソフトウェアを秘密として保持し、その全部または一部を第三者に開示してはなりません。
- (2) 前項の義務は、許諾ソフトウェアに含まれる情報のうち、次の各号のいずれかに該当するものには及ばないものとし、
  - 甲が許諾ソフトウェアを受領したときに既に所有していた情報
  - 甲が許諾ソフトウェアを受領したときに既に公知であった情報
  - 甲が許諾ソフトウェアを受領後、甲の責によらず公知となった情報
  - 甲が許諾ソフトウェアにふれることなく独自に開発した情報
  - 行政庁または裁判所から開示を求められた情報。ただし、この場合、開示に先立ち甲は書面により乙に通知し、乙に当該開示に異議を申し立てる機会を与えなければならないものとし、

#### 6. 契約期間と終了

- (1) 本契約は、甲が許諾ソフトウェアの梱包シールを解いた時から発効し、下記の各号により終了するまで有効に存続するものとし、
- (2) 甲は乙に対し1ヶ月前の書面通知を出すことにより、何時でも本契約を終了させることができます。
- (3) 乙は、甲が本契約のいずれかの条項に違反したときは甲に書面通知を出すことにより何時でも無条件に本契約を終了させることができます。

#### 7. 本契約終了後の義務

- (1) 甲は、本契約が終了した場合には、乙から受領した許諾ソフトウェア及び本契約に基づいて作成した許諾ソフトウェア複製物のすべてを破棄し、また、開発ソフトウェアに含まれるライブラリ（ただし、甲システムに組み込んで販売したものを除きます）を開発ソフトウェアから完全に排除するものとし、但し、乙から書面による承諾を得た場合には、甲は保存用として許諾ソフトウェアの複製物を1部保存することができます。
- (2) (甲は、本契約終了の日から前項の処置を行い、その旨を証明する文書を1ヶ月以内に乙に提供するものとし、

#### 8. 紛争処理

- (1) 許諾ソフトウェア又は開発ソフトウェアのうち組み込まれた許諾ソフトウェアの部分に関し、甲と第三者との間に著作権、工業所有権その他の権利侵害などの紛争が生じた場合には、乙は自己の責任及び費用でこれを解決するものとし、なお、当該第三者との紛争が開発ソフトウェアのうち組み込まれた許諾ソフトウェア以外の部分に関するもの、開発ソフトウェアを甲システムに組み込んだことによって生じたもの及び甲の仕様もしくは指示に基づくものは、甲の責任と費用でこれを解決するものとし、
- (2) 前項において責任の分担又は費用の分担につき疑義が生じた場合、甲乙協議の上それらを解決し、また、相手方当事者から解決のための協力を要請された場合には、自己の可能な範囲で情報提供などの協力を行なうものとし、



## 9. 乙の免責

- (1) 乙は、許諾ソフトウェアに関するいかなる保証および賠償を行いません。従って許諾ソフトウェアに関して発生した問題は甲の責任および費用負担によって処理されるものとします。但し、乙は、甲が許諾ソフトウェアに関するユーザー登録をした場合に限り、許諾ソフトウェアの納入日から1年間、許諾ソフトウェアに使用上の不具合が発生した場合の乙の適切と判断する援助及び修正を行うこととし、これをもって唯一の責任とします。また本条項により甲に乙が提供したソフトウェアは許諾ソフトウェアとみなします。
- (2) 乙は前項に基づく援助及び修正ですべての不具合が修正されることを保証するものではありません。
- (3) なお乙は本契約第9条(1)に基づく援助および修正にかかる費用を甲に請求することができるものとします。

## 10. その他

- (1) 本契約に関わる紛争は、乙が指定する裁判所を管轄裁判所として解決するものとします。
- (2) 本契約に規定のない事項及び甲と乙との間に生じた疑義については、甲乙協議のうえ解決するものとします。

株式会社ルネサス テクノロジ