

ARMCTv4.1 における変更点

本ドキュメントは ARM 社の提供する ARM Compiler toolchain Version4.1 Migration and Compatibility (ARM DUI 0530A) 内 Chapter3 Migrating from RVCT v4.0 to ARM Compiler v4.1 の内容に基づき作成されたものです。内容につきましては全て上記ドキュメントをマスターといたしておりますので、ご使用の際には必ず上記ドキュメントを参照の上、本ドキュメントは参考資料として用いる形をお取りくださいますようお願い申し上げます。

本ドキュメントでは ARM Compiler toolchain (ARMCT) v4.1 と以前のリリースである RealView Compilation Tools (RVCT) v4.0 での移行と互換性に影響のある変更点について記載しています。

1. RVCTv4.0 と ARM Compiler v4.1 での一般的な変更点	3
1.1 参考資料	3
2. RVCTv4.0 と ARM Compiler v4.1 でのコンパイラに対する変更点	3
2.1 参考資料	3
3. RVCTv4.0 と ARM Compiler v4.1 でのリンカに対する変更点	3
3.1 参考資料	3
4. RVCTv4.0 と ARM Compiler v4.1 でのアセンブラに対する変更点	4
4.1 参考資料	4
5. RVCTv4.0 と ARM Compiler v4.1 での C および C++ ライブラリに対する変更点	5
5.1 参考資料	5

1. RVCTv4.0 と ARM Compiler v4.1 での一般的な変更点

デフォルトのヘッダやライブラリディレクトリ設定用の環境変数の慣習的なネーミングが変更されました。これらは RVCT ではなく、ARMCC をプレフィックスとするようになりました。例えば、RVCT40INC ではなく ARMCC41INC のようになります。

1.1 参考資料

リファレンス

Introducing the ARM Compiler toolchain :

- 2-12 ページ *Toolchain environment variables*

2. RVCTv4.0 と ARM Compiler v4.1 でのコンパイラに対する変更点

列挙型の符号ルールが慣習に沿って変更されました。列挙型のコンテナは負の定数が定義されていなければ unsigned になります。RVCT v4.0 10Q1 パッチでは GCC モードでのみこの変更が加えられています。

-O3 は--multifile を暗黙的に指定しなくなりました。--multifile オプションは別オプションとして常に別個に存在しており、ビルドにはこのオプションを指定することを推奨されます。

2.1 参考資料

リファレンス

Compiler Reference :

- 3-109 ページ *--multifile, --no_multifile*
- 3-114 ページ *-Onum*

3. RVCTv4.0 と ARM Compiler v4.1 でのリンカに対する変更点

armlink v4.1 は GNU linker 制御スクリプトのサブセットをサポートします。GNU ld とより緊密に合致した振る舞いのために、armlink は--sysv コマンドラインオプション指定時には内部リンカ制御スクリプトを使用します。以前のバージョンの armlink は内部スキッタファイルを使用していました。

制御スクリプトの使用は論理的には RVCTv4.0 と同等ですが、物理的には異なったレイアウトを生成します。以前のデフォルトスキッタファイルでのレイアウトに戻りたい場合は、--no_use_sysv_default_script コマンドラインオプションを使用してください。

-T オプションを使用することにより、内部制御スクリプトをユーザ定義の制御スクリプトに置き換えることができます。

3.1 参考資料

リファレンス

Linker Reference :

- 2-159 ページ *--use_sysv_default_script, --no_use_sysv_default_script*
- 2-153 ページ *--sysv*

4. RVCTv4.0 と ARM Compiler v4.1 でのアセンブラに対する変更点

以下の変更がアセンブラに加えられました：

アセンブラによるファイルの読み込みと処理の方法が変更されました

過去のアセンブラはアセンブルされるソースファイルがアセンブラに対する 2 つのパスの間でいずれかを取る可能性がありました。次の例では、シンボル `foo` が最初のパスで定義されていないために、シンボル `num` は 2 番目のパスで定義されます。

```
AREA x, CODE
[ :DEF: foo
num EQU 42
]
foo DCD num
END
```

アセンブラによるファイルの読み込みと処理の方法は変更され、より厳密になりました。上記のようなコードは両方のパスにおいてファイルを通じて同一で有るように書き直す必要があります。

アセンブラによるメッセージ出力の変更

一般的に、全てのメッセージにおいて問題となっている部分を caret 文字が示す形でソース行の参照場所を表示します。

例：

```
"foo.s", line 3 (column 19): Warning: A1865W: '#' not seen before constant expression
 3 00000000 ADD r0,r1,1
      ^
```

診断メッセージの変更

多くの ARM (SP を使用する) 命令は、32-bit Thumb 命令が紹介された時非推奨とされました。これらの命令は 32-bit Thumb 命令を持っている CPU に対してのアセンブルをおこなわない場合、非推奨として診断されなくなりました。過去の CPU に対してこれらのワーニングを有効にするには、`--diag_warning=1745,1786,1788,1789,1892` のオプションを使用します。この変更は RVCT v4.0 09Q4 パッチにておこなわれました。

廃止されたコマンドラインオプション

`-o` コマンドラインオプションは廃止されました。代わりに `-o` を使用してください。

4.1 参考資料

リファレンス

Assembler Reference :

- 2-14 ページ `--diag_warning=tag{, tag}`
- 2-24 ページ `-o filename`

5. RVCTv4.0 と ARM Compiler v4.1 での C および C++ライブラリに対する変更点

ライブラリは Thumb2 をサポートするターゲット上で、より多くの Thumb2 コードを使用するようになりました。これによりパフォーマンスに影響することなく、コードサイズを削減する効果が期待されます。必要であれば、`--no_thumb2_library` リンカオプションによって、古いスタイルのライブラリに戻すことができます。

一部のまれなケースにおける算術演算関数の戻り値が POSIX/C99 要求に従うようになりました。過去の振る舞いに戻すには以下のオプションを使用します：

```
#pragma import __use_rvct_matherr
```

RVCTv4.0 09Q4 パッチ以降、以下のオプションで新しい振る舞いを有効にすることができます：

```
#pragma import __use_c99_matherr
```

5.1 参考資料

コンセプト

Using ARM C and C++ Libraries and Floating-Point Support :

- 2-127 ページ *How the ARM C library fulfills ISO C specification requirements*

— 以上 —