

RealView[®] Development Suite

バージョン 3.0

スタートガイド

ARM[®]

RealView Development Suite

スタートガイド

Copyright © 2003-2006 ARM Limited. All rights reserved.

リリース情報

本書には以下の変更が追加されています。

変更履歴

日付	発行	変更
2003年9月	A	RVDS v2.0 リリース
2004年1月	B	RVDS v2.1 リリース
2004年12月	C	RVDS v2.2 リリース
2005年5月	D	RVDS v2.2 SP1 リリース
2006年3月	E	RVDS v3.0 リリース

著作権

® または ™ のマークが付いた言葉およびロゴは、ARM Limited が所有する登録商標または商標です。本書に記載されている他の製品名は、各社の所有する商標です。

本書に記載されている情報の全部または一部、ならびに本書で紹介する製品は、著作権所有者の文書による事前の許可を得ない限り、転用・複製することを禁じます。

本書に記載されている製品は、今後も継続的に開発・改良の対象となります。本書に含まれる製品およびその利用方法についての情報は、ARM が利用者の利益のために提供するものです。したがって当社では、製品の市販性または利用の適切性を含め、暗示的・明示的に関係なく一切の責任を負いません。

本書は、本製品の利用者をサポートすることだけを目的としています。本書に記載されている情報の使用、情報の誤りまたは省略、あるいは本製品の誤使用によって発生したいかなる損失・損傷についても、ARM Limited は一切責任を負いません。

機密保持ステータス

本書は非機密扱いであり、本書を使用、複製、および開示する権利は、ARM および ARM が本書を提供した当事者との間で締結した契約の条項に基づいたライセンスの制限により異なります。

製品ステータス

本書の情報は最終版であり、開発済み製品に対応しています。

Web アドレス

<http://www.arm.com>

目次

RealView Development Suite スタートガイド

	序章	
	本書について	vi
	フィードバック	x
第 1 章	はじめに	
1.1	RealView Development Suite のコンポーネント	1-2
1.2	RealView Development Suite のライセンス	1-10
1.3	RealView Development Suite のマニュアル	1-11
1.4	RealView Development Suite のサンプル	1-13
1.5	ARM Developer Suite	1-15
1.6	ターゲットアクセスのサポート	1-16
1.7	RVDS 環境で問題を修正する方法	1-19
第 2 章	RVDS v3.0 の機能	
2.1	RVDS v3.0 の変更点	2-2
2.2	RealView Debugger の変更点	2-3
2.3	RealView Compilation Tools の変更点	2-4
2.4	シミュレータのサポート	2-5
2.5	CodeWarrior for RVDS の変更点	2-6
2.6	マニュアルの変更点	2-7

第 3 章

RealView Development Suite の使い方

3.1	ビルドタスクとデバッグタスクの概要	3-2
3.2	サンプルプロジェクトの使用	3-5

付録 A

armenv ツールの使用

A.1	armenv ツールについて	A-2
A.2	armenv ツールの使用	A-3

付録 B

以前のリリースについて

B.1	RVDS v2.2 SP1 と RVDS v2.2 との間の変更点	B-2
B.2	RVDS v2.2 と RVDS v2.1 との間の変更点	B-3
B.3	RVDS v2.1 と RVDS v2.0 との間の変更点	B-6
B.4	RVDS v2.2 と ADS v1.2.1 との間の変更点	B-8

序章

本章では、*RealView® Development Suite v3.0* スタートガイドについて概説します。このガイドでは、*RealView Development Suite* (RVDS) を使用してソフトウェアプロジェクトを管理し、アプリケーションプログラムをデバッグする方法を紹介します。本章は以下のセクションから構成されています。

- 本書について (P. vi)
- フィードバック (P. x)

本書について

RealView Development Suite には、ARM® アーキテクチャベースのプロセッサをターゲットとしたソフトウェア開発プロジェクトをビルド、デバッグ、および管理するためのツールが収録されています。本書の内容は以下のとおりです。

- RealView Development Suite を構成するソフトウェアコンポーネントの説明
- RVDS v3.0、以前のバージョンの RVDS、および *ARM Developer Suite™* (ADS) v1.2 の違いの概要
- RealView Development Suite を初めて使用するユーザのための用語集

対象読者

本書は、ARM アーキテクチャベースのプロセッサをターゲットとする開発プロジェクトの管理に RVDS を使用している開発者を対象としています。本書の内容は、ソフトウェア開発の経験が豊富で、ARM 開発ツールを熟知していないソフトウェア開発者を想定して記述されています。

本書の構成

本書は以下の章から構成されています。

第1章 はじめに

RVDS v3.0 のコンポーネント、ライセンス、およびマニュアルの概要を説明します。

第2章 RVDS v3.0 の機能

RVDS v3.0 の新機能について説明します。

第3章 RealView Development Suite の使い方

RVDS ツールで実行できる主なタスクについて説明します。また、RVDS 付属のサンプルプロジェクトについても説明します。

付録 A armenv ツールの使用

armenv ツールの使用方法について説明します。

付録 B 以前のリリースについて

前のバージョンの RVDS および ADS v1.2 について説明します。

表記規則

本書では以下の表記規則を使用しています。

<i>italic</i>	重要事項、重要用語、相互参照、引用箇所を斜体で記載しています。
bold	メニュー名などのユーザインタフェース要素を太字で記載しています。また、必要に応じて記述リスト内の重要箇所、ARM プロセッサの信号名、重要用語、および専門用語にも太字を使用しています。
monospace	コマンド、ファイル名、プログラム名、ソースコードなど、キーボードから入力可能なテキストを示しています。
<u>monospace</u>	コマンドまたはオプションに使用可能な略語を示します。コマンド名またはオプション名をすべて入力する代わりに、下線部分の文字だけを入力することができます。
<i>monospace italic</i>	コマンドまたは関数の引数で、特定の値に置き換えることが可能なものを示しています。
monospace bold	サンプルコード以外に使用される言語キーワードを示しています。
...	パス名の後に ... がある場合、該当ディレクトリが最後に指定したディレクトリ名の下にあることを示しています。通常、オペレーティングシステムによってパス名が異なる場合、その部分が省略されます。以下に例を示します。 <pre>install_directory\ARM\RVD\Examples\...</pre> パス名の途中に ... がある場合、指定したディレクトリ名の中にディレクトリが存在することを示します。通常、このようなパスでは、バージョン番号やビルド番号およびプラットフォーム固有のディレクトリ名が省略されます。以下に例を示します。 <pre>install_directory\ARM\RVD\Core\...\etc</pre>

参考資料

ここでは、補足情報を記載した ARM Limited および各社の出版物を紹介します。

ARM Limited は自社出版物の定期的な更新・修正を行っています。最新の正誤表、追補表、および FAQ については、<http://www.arm.com> をご覧下さい。

ARM の出版物

ARM アプリケーションの使用を制御する GLOBEtrouter Inc. 製の FLEXlm ライセンス管理システムの詳細については、以下のマニュアルを参照して下さい。

- *ARM FLEXlm ライセンス管理ガイド v4.0* (ARM DUI 0209J-00)

RVDS v3.0 でのライセンス管理の詳細については、必ずこのマニュアルのバージョン 4.0 を参照するようにして下さい。

本書は RVDS の付属マニュアルの 1 冊です。その他の付属マニュアルを以下に示します。

- *RealView デバッガ v3.0 エssenシャルガイド* (ARM DUI 0181J-00)
- *RealView デバッガ v3.0 ユーザガイド* (ARM DUI 0153J-00)
- *RealView デバッガ v3.0 ターゲットコンフィグレーションガイド* (ARM DUI 0182J-00)
- *RealView Debugger v3.0 Trace User Guide* (ARM DUI 0322)
- *RealView Debugger v3.0 RTOS Guide* (ARM DUI 0323)
- *RealView Debugger v3.0 Command Line Reference Guide* (ARM DUI 0175)
- *RealView Compilation Tools v3.0 基本操作ガイド* (ARM DUI 0202J)
- *RealView Compilation Tools v3.0 デベロッパガイド* (ARM DUI 0203J)
- *RealView Compilation Tools v3.0 アセンブラガイド* (ARM DUI 0204J)
- *RealView Compilation Tools v3.0 コンパイラ/ライブラリガイド* (ARM DUI 0205J)
- *RealView Compilation Tools v3.0 リンカ/ユーティリティガイド* (ARM DUI 0206)
- *RealView ARMulator ISS v1.4 ユーザガイド* (ARM DUI 0207J-00)
- *RealView Development Suite AXD/armstd デバッガガイド* (ARM DUI 0066J-00)
- *RealView Development Suite CodeWarrior IDE Guide* (ARM DUI 0065)
- *RealView Development Suite 用語集* (ARM DUI 0324J)

以下のマニュアルでは、ARM のアーキテクチャ、プロセッサ、関連デバイス、およびソフトウェアインタフェースに関する一般的な情報を提供しています。

- *ARM Reference Peripheral Specification* (ARM DDI 0062)
- お使いのハードウェアデバイスの ARM データシートまたはテクニカルリファレンスマニュアル

ソフトウェアインタフェースと ARM でサポートされている標準に関する一般的な情報については、`install_directory\Documentation\Specifications\` を参照して下さい。

RealView Development Suite での使用に適した ARM デバッグインタフェースに関する情報については、以下のマニュアルを参照して下さい。

- *RealView ICE and RealView Trace User Guide* (ARM DUI 0155)
- *Multi-ICE[®] ユーザガイド* (ARM DUI 0048J-00)
- *ARM MultiTrace[™] User Guide* (ARM DUI 0150)
- *ARM Agilent Debug Interface User Guide* (ARM DUI 0158)

他の出版物

ARM アーキテクチャに関する一般的な情報については、以下の出版物を参照して下さい。

Steve Furber, *ARM system-on-chip architecture* (2nd edition, 2000). Addison Wesley, ISBN 0-201-67519-6.

CEVA, Inc の CEVA-Oak、CEVA-TeakLite、および CEVA-Teak プロセッサの詳細については、<http://www.dspg.com> をご覧下さい。

LSI Logic の ZSP 事業部の CEVA-Oak、CEVA-TeakLite、および CEVA-Teak プロセッサの詳細については、<http://www.zsp.com> をご覧下さい。

フィードバック

ARM Limited では、RVDS および本書に関するフィードバックをお待ちしております。

RealView Development Suite に関するフィードバック

RVDS に関して問題がある場合は、購入元にお問い合わせ下さい。このとき、迅速かつ適切な対応をさせて頂くために、以下の情報をご用意下さい。

- お名前と会社名
- 製品のシリアル番号
- 製品のリリース情報
- プラットフォームの詳細（ハードウェアプラットフォーム、オペレーティングシステムの種類とバージョンなど）
- 問題を再現するサイズの小さな独立したサンプルコード
- 操作の目的と実際の動作に関する詳しい説明
- 使用したコマンド（コマンドラインオプションを含む）
- 問題を例示するサンプル出力
- ツールのバージョン情報（バージョン番号、日付を含む）

——— 注 ———

RealView Debugger に関して問題がある場合は、[Help] → [Send a Problem Report...] メニューを使用してソフトウェア障害レポートを作成できます。詳細については、RealView Debugger のマニュアルを参照して下さい。

本書に関するフィードバック

本書に関するご意見につきましては、以下の内容を記載した電子メールを errata@arm.com までお送り下さい。

- マニュアル名
- 文書番号
- 問題のあるページ番号
- 問題点の簡潔な説明

補足すべき点や改善すべき点についてのご提案もお待ちしております。

第 1 章

はじめに

本章では、*RealView® Development Suite* (RVDS) v3.0 について概説します。また、コンポーネントアプリケーション、RVDS v3.0 の機能を拡張するために購入できる追加ライセンス、および付属マニュアルの概要について説明します。

本章は以下のセクションから構成されています。

- *RealView Development Suite* のコンポーネント (P. 1-2)
- *RealView Development Suite* のライセンス (P. 1-10)
- *RealView Development Suite* のマニュアル (P. 1-11)
- *RealView Development Suite* のサンプル (P. 1-13)
- *ARM Developer Suite* (P. 1-15)
- ターゲットアクセスのサポート (P. 1-16)
- RVDS 環境で問題を修正する方法 (P. 1-19)

1.1 RealView Development Suite のコンポーネント

RVDS には、ARM[®] ファミリの RISC プロセッサで実行される組み込みシステムアプリケーション用の調整された開発環境が用意されています。この開発環境は、サポートドキュメントとサンプルを備えたツールスイートで構成されています。このツールを使用すると、ターゲットハードウェアまたはソフトウェアシミュレータで、アプリケーションを記述、ビルド、およびデバッグできます。

このセクションは以下のサブセクションから構成されています。

- *RealView Development Suite* のインストール、サンプル、およびマニュアルのディレクトリ
- *CodeWarrior for RVDS* (P. 1-3)
- *RealView Compilation Tools* (P. 1-4)
- *RealView Debugger* (P. 1-5)
- *RealView ARMulator 命令セットシミュレータ* (P. 1-7)
- *Instruction Set System Model* (P. 1-8)
- *RVDS* のサンプルプロジェクト (P. 1-8)
- *ARM eXtended Debugger* (P. 1-9)
- *ARM* シンボリックデバッガ (P. 1-9)

1.1.1 RealView Development Suite のインストール、サンプル、およびマニュアルのディレクトリ

システムにインストールされているさまざまな RVDS のディレクトリには、役に立つファイルがあります。RVDS のマニュアルでは、必要に応じてこれらのディレクトリを参照しています。

すべてのディレクトリは、主なインストールディレクトリの下にあります。また、マニュアルで使用されるサンプルの多くは、1つのサンプルディレクトリ内にあります。例外については、参照先のディレクトリが明記されます。

インストール、サンプル、およびマニュアルの主なディレクトリを、表 1-1 に示します。表に示した *install_directory* は、デフォルトのインストールディレクトリです。別のインストールディレクトリを指定した場合、パス名は選択したディレクトリに対応します。

表 1-1 RealView Development Suite のディレクトリ

ディレクトリ	Windows 環境のデフォルトのパス	Sun Solaris および Red Hat Linux 環境のデフォルトのパス
<code>install_directory</code>	C:\Program Files\ARM	~/arm
サンプル ^a	<code>install_directory\RVDS\Examples\...</code>	<code>install_directory/RVDS/Examples/...</code>
マニュアル ^b	<code>install_directory\Documentation\...</code>	<code>install_directory/Documentation/...</code>

- RealView Development Suite 付属のサンプルの概要については、「RealView Development Suite のサンプル」(P. 1-13) を参照して下さい。
- マニュアルの表示方法の詳細については、「RealView Development Suite のマニュアル」(P. 1-11) を参照して下さい。

1.1.2 CodeWarrior for RVDS

CodeWarrior for RVDS は、Metrowerks CodeWarrior IDE バージョン 5.7 をベースにしています。RVDS v3.0 の CodeWarrior for RVDS は、Windows XP システムおよび Windows 2000 システムのみでサポートされています。Sun Solaris および Red Hat Linux では、RVDS と共に提供されません。

注

ブレークポイントの設定など、CodeWarrior の組み込みデバッグ機能は使用しないで下さい。アプリケーションをコンパイルした後、**[Project]** → **[Debug]** メニューオプションを使用して RealView Debugger を起動し、すべてのデバッグ操作を RealView Debugger 内で実行します。

CodeWarrior for RVDS の概要

- ソフトウェア開発プロジェクトを管理するためのグラフィカルユーザインタフェースを提供します。CodeWarrior for RVDS を使用すると、ARM プロセッサや Thumb[®] プロセッサをターゲットとする C、C++、および ARM アセンブリ言語のコードを開発できます。以下の機能を提供することにより、ビルドサイクル時間が短縮されます。
 - 幅広いプロジェクト管理機能
 - ルーチンを簡単に検索できるようにするコードナビゲーションルーチン
- プロジェクトコードをコンパイル、アセンブル、およびリンクするように ARM ツールを構成できます。

- ソースコードファイル、ライブラリファイル、およびコンフィグレーション設定をプロジェクトにまとめることができます。また、各プロジェクトでは、ビルドターゲット設定の複数のコンフィグレーションを作成および管理できます。例えば、デバッグ用のビルドターゲットとリリース用のビルドターゲットをコンパイルし、ARM7TDMI[®] ベースのハードウェアでコードをターゲットにできます。ビルドターゲットでは、独自の設定を使用すると同時に、同じプロジェクト内でファイルを共有できます。
- 以下の機能が備わっています。
 - 構文を色分けするソースコードエディタ。このエディタは、CodeWarrior for RVDS ブラウザと統合されています。
 - コードで定義されているシンボルのデータベースを保持するソースコードブラウザ。このブラウザを使用すると、ソースコードを迅速かつ簡単にナビゲートできます。
 - グレップ形式の正規表現を使用できるようにする検索および置換機能。この機能では、複数のファイル内の一括検索も実行できます。
 - ファイル比較機能。この機能を使用すると、あるテキストファイルと別のテキストファイルの相異点を検索（必要に応じてマージ）したり、ディレクトリの内容を比較したりできます。

CodeWarrior for RVDS の使い方の詳細については、*RealView Development Suite CodeWarrior IDE Guide* を参照して下さい。

1.1.3 RealView Compilation Tools

RealView Compilation Tools (RVCT) を使用すると、C、C++、または ARM のアセンブリ言語プログラムをビルドできます。RVCT は次のツールで構成されています。

- ARM と Thumb の C および C++ コンパイラ `armcc`
- ARM と Thumb のアセンブラ `armasm`
- ARM リンカ `armlink`
- ARM ライブラリアン `armar`
- ARM イメージ変換ユーティリティ `fromelf`
- サポートライブラリ

RVCT で使用可能な機能の詳細については、*RealView Compilation Tools 基本操作ガイド* を参照して下さい。

RVCT のツールとユーティリティおよびそれらの使用方法の詳細については、RVCT のマニュアルを参照して下さい。マニュアルは、「*RealView Development Suite のマニュアル*」(P. 1-11) にリストされています。また、RVCT ツールのアップデートやパッチについては、使用できるようになったときに ARM Web サイトを参照して下さい。

1.1.4 RealView Debugger

RealView Debugger とサポートされているデバッグターゲット（「ターゲットアクセスのサポート」(P. 1-16) 参照）により、アプリケーションプログラムのデバッグが可能になります。また、プログラムの実行フローを完全に制御することによって、エラーをすばやく検出して修正できます。

注

Sun Solaris または Red Hat Linux 環境での RealView Debugger の使用に関する固有の情報については、Sun Solaris や Red Hat Linux の RealView Debugger に関する付録を参照して下さい。この付録は、*RealView デバッガユーザガイド*にあります。

RealView Debugger では、以下の項目がサポートされています。

- マルチプロセッサデバッグ（「マルチプロセッサデバッグ」(P. 1-5) 参照）
- *Digital Signal Processor* (DSP) デバッグ（「DSP デバッグ」参照）
- トレース、解析、およびプロファイリング（「トレース、分析、およびプロファイリング」(P. 1-6) 参照）
- ベンダ固有のプラグインのダウンロードによるオペレーティングシステム (OS) の認識機能（「OS の認識機能」(P. 1-7) 参照）

RealView Debugger のデフォルトのライセンスを使用すると、1 つの ARM アーキテクチャベースのプロセッサで実行されるアプリケーションをデバッグできます。ただし、追加のライセンスを購入すると、RealView Debugger の機能を拡張して、複数のプロセッサで実行されているアプリケーションをデバッグしたり、DSP でのデバッグをサポートしたりすることができます。詳細については、「*RealView Development Suite のライセンス*」(P. 1-10) を参照して下さい。

RealView Debugger で使用可能な機能の詳細については、*RealView デバッガエッセンシャルガイド*を参照して下さい。

RealView Debugger とその使用方法の詳細については、RealView Debugger のマニュアルを参照して下さい。マニュアルは、「*RealView Development Suite のマニュアル*」(P. 1-11) にリストされています。

マルチプロセッサデバッグ

マルチプロセッサデバッグを使用すると、複数のプロセッサで実行しているソフトウェアシステムをデバッグできます。プロセッサは、単一の開発ボードまたは複数の開発ボードに設置できます。どちらの場合も、RealView Debugger では、プロセッサごとに異なる接続を使用します。

マルチプロセッサデバッグを使用すると、複合コアシステムをデバッグして、プロセッサのオペレーションを同期化できます。

複数のボードの接続で同じプロセッサを使用する場合は、新しいターゲット記述の作成が必要になることがあります。カスタムターゲットの作成の詳細については、*RealView デバッガターゲットコンフィグレーションガイド*を参照して下さい。

マルチプロセッサデバッグには、個別に購入したライセンスが必要です。詳細については、「*RealView Development Suite のライセンス*」(P. 1-10)を参照して下さい。

マルチプロセッサデバッグの詳細については、*RealView デバッガユーザガイド*の複数のターゲットのデバッグに関する章を参照して下さい。

DSP デバッグ

RealView Debugger では、以下の DSP のデバッグをサポートしています。

- CEVA-Oak、CEVA-Teaklite (リビジョン B と C)、および CEVA-Teak (リビジョン A と B) DSP
- Samsung Scorpio II の CEVA-Teak
- LSI Logic ZSP400 および ZSP500 DSP

DSP デバッグには、個別に購入したライセンスが必要です。詳細については、「*RealView Development Suite のライセンス*」(P. 1-10)を参照して下さい。

DSP のサポートに関する詳細については、*RealView デバッガユーザガイド*の DSP サポートに関する章を参照して下さい。

トレース、分析、およびプロファイリング

RealView Debugger では、トレース用ハードウェアまたはハードウェアシミュレータのいずれかを使用したトレースをサポートしています。トレース用ハードウェアには、以下のいずれかを使用できます。

- *Embedded Trace Macrocell™* (ETM) 搭載のプロセッサ
- ARM *Embedded Trace Buffer™* (ETB™) などのオンチップトレースバッファ
- RealView ICE などの Joint Test Action Group (JTAG) インタフェースユニット

ETM 以外の基本的なトレースは、RealView ARMulator® ISS ハードウェアシミュレータによってサポートされます。

RealView Debugger で使用可能なトレース機能の詳細については、*RealView デバッガ拡張機能ユーザガイド*のトレースに関する章を参照して下さい。

OS の認識機能

OS の認識機能とは、RealView Debugger に組み込まれている拡張機能です。この RealView Debugger の拡張機能を使用するには、ご使用の OS 用のプラグインを入手する必要があります。OS のプラグインを使用すると、RealView Debugger は OS のスレッドやリソース（クエリ、メールボックス、セマフォなど）に加え、C またはアセンブラのソースレベルのシンボリックデバッグ情報も参照します。ご使用の OS 用のプラグインの入手方法の詳細については、「*RealView Debugger のダウンロード*」(P. 1-7) を参照して下さい。

RealView Debugger の OS の認識機能はベンダに依存していません。OS のプラグインは、複数のベンダからダウンロードして使用できます。そのため、さまざまな OS プラットフォーム用のアプリケーションを開発できます。複数のベンダの OS 用のプラグインを同じデバッグセッションで使用する場合は、新しいターゲット記述の作成が必要になることがあります。カスタムターゲットの作成の詳細については、*RealView デバッグターゲットコンフィグレーションガイド*を参照して下さい。

OS のサポートの詳細については、*RealView Debugger RTOS Guide* の OS のサポートに関する章を参照して下さい。また、OS のサポートに関する情報が公開されたときには ARM Web サイトも参照して下さい。

RealView Debugger のダウンロード

RealView Debugger の [Help] メニューからさまざまな RealView Debugger のダウンロードにアクセスできます。

[Help] → [ARM on the Web] → [Goto RTOS Awareness Downloads]

ARM Web サイトの「*OS Aware Debugger*」Web ページが表示されます。任意の OS プラグインをここで探してダウンロードできます。

[Help] → [ARM on the Web] → [Goto Update and Utility Downloads]

ARM Web サイトの「*RealView Debugger - Updates & Utilities*」Web ページが表示されます。ソフトウェアのアップデートおよびユーティリティをここで探してダウンロードできます。

1.1.5 RealView ARMulator 命令セットシミュレータ

RealView ARMulator 命令セットシミュレータ (RVISS) は、メモリシステムおよびペリフェラルを使用して、ARM プロセッサの命令セットとアーキテクチャをシミュレートします。

RVISS を使用すると、ターゲットハードウェアを使用せずに組み込みアプリケーションの開発およびデバッグを開始できます。これは、ハードウェアがまだ開発中の場合や使用可能な開発ボードの数に制限がある場合に役立ちます。

表 1-2 では、Windows、Sun Solaris、および Red Hat Linux 環境で ARM デバッガから使用できる RVISS のインタフェース接続について示します。RealView Connection Broker 経由で RVISS に接続すると、RVISS の複数のインスタンスに接続して、マルチプロセッサシステムをシミュレートできます。

表 1-2 各プラットフォームでサポートされている RealView ARMulator ISS 接続

ARM デバッガ	リモートデバッグインタフェース接続	RealView Connection Broker 接続
RealView Debugger	Windows ^a	Windows、Sun Solaris、および Red Hat Linux
AXD	Windows	該当なし
armsd	Windows、Sun Solaris、および Red Hat Linux	該当なし

a. RealView Debugger での RVISS へのリモートデバッグインタフェース接続は本リリースで廃止されます。

RVISS で使用可能な機能の詳細については、*RealView ARMulator ISS ユーザガイド*を参照して下さい。

1.1.6 Instruction Set System Model

Instruction Set System Model (ISSM) は、Cortex™-A8 プロセッサおよび Cortex-M3 プロセッサの命令セットとアーキテクチャをシミュレートします。詳細については、*RealView デバッガターゲットコンフィグレーションガイド*を参照して下さい。

1.1.7 RVDS のサンプルプロジェクト

サンプルプロジェクトは、RVDS に付属しています（*RealView Development Suite* の「サンプル」(P. 1-13) 参照）。サンプルプロジェクトは次のディレクトリにあります。

```
install_directory\RVDS\Examples\...
```

Windows の [スタート] メニューから直接サンプルプロジェクトを探すには、次の項目を選択します。

[プログラム] → [ARM] → [ARM RealView Development Suite v3.0] → [Examples]

1.1.8 ARM eXtended Debugger

ARM eXtended Debugger (AXD) は、シングルプロセッサのデバッガで、Windows のみに対応しています。AXD とサポートされているデバッグターゲット (「ターゲットアクセスのサポート」 (P. 1-16) 参照) により、アプリケーションプログラムのデバッグが可能になります。また、プログラムの実行フローを制御することによって、エラーをすばやく検出して修正できます。

—— 注 ——

AXD は、古い ARM7™ および ARM9™ をサポートするために提供されているため、RealView ICE JTAG ユニットでは動作しません。

AXD の使用方法の詳細については、*RealView Development Suite AXD and armsd Debuggers Guide* を参照して下さい。

1.1.9 ARM シンボリックデバッガ

ARM シンボリックデバッガ (armsd) は、コマンドラインインタフェースから実行するシングルプロセッサのデバッガです。Windows、Sun Solaris、および Red Hat Linux で使用できます。

armsd を使用すると、RVISS のリモートデバッグインタフェース (RDI) を使用してアプリケーションプログラムをデバッグできます (「*RealView ARMulator* 命令セットシミュレータ」 (P. 1-7) 参照)。

—— 注 ——

armsd と RVISS を併用した場合にサポートされるのは、ARM7 ターゲットと ARM9 ターゲットだけです。

armsd の使用方法の詳細については、*RealView Development Suite AXD and armsd Debuggers Guide* を参照して下さい。

1.2 RealView Development Suite のライセンス

RVDS のすべてのライセンスは、FLEXlm ライセンス管理システムによって制御されています。RVDS ライセンスを追跡および制御するには、FLEXlm サーバソフトウェアを使用します。現在、ARM Licensing Web ページ (<http://license.arm.com>) を使用してライセンスを請求できます。詳細については、*ARM FLEXlm ライセンス管理ガイド*を参照して下さい。

注

ライセンスを請求するために直接 ARM Limited に問い合わせる必要はありません。

RealView Debugger の機能で個別に使用できる RealView Development Suite のライセンスについては、以下のセクションで説明します。

- マルチプロセッサデバッグのライセンス
- CEVA-Oak および CEVA-Teaklite DSP デバッグのライセンス
- CEVA-Teak DSP デバッグのライセンス
- LSI Logic ZSP DSP デバッグのライセンス

1.2.1 マルチプロセッサデバッグのライセンス

マルチプロセッサデバッグのライセンスを使用すると、複数のプロセッサで実行しているソフトウェアシステムをデバッグできます（「マルチプロセッサデバッグ」(P. 1-5) 参照）。

1.2.2 CEVA-Oak および CEVA-Teaklite DSP デバッグのライセンス

CEVA-Oak および CEVA-Teaklite *Digital Signal Processor* (DSP) のサポートライセンスを使用すると、CEVA-Oak DSP や CEVA-Teaklite DSP で実行しているアプリケーションをデバッグできます（「DSP デバッグ」(P. 1-6) 参照）。

1.2.3 CEVA-Teak DSP デバッグのライセンス

CEVA-Teak *Digital Signal Processor* (DSP) のサポートライセンスを使用すると、CEVA-Teak DSP で実行しているアプリケーションをデバッグできます（「DSP デバッグ」(P. 1-6) 参照）。

1.2.4 LSI Logic ZSP DSP デバッグのライセンス

LSI Logic ZSP *Digital Signal Processor* (DSP) のサポートライセンスを使用すると、ZSP400 DSP や ZSP500 DSP で実行しているアプリケーションをデバッグできます（「DSP デバッグ」(P. 1-6) 参照）。

1.3 RealView Development Suite のマニュアル

このセクションでは、RVDS 付属のマニュアルについて説明します。以下のセクションから構成されています。

- マニュアルのリスト
- オンラインでの詳細情報の取得

1.3.1 マニュアルのリスト

RVDS のマニュアルは以下のガイドで構成されています。

- *RealView Development Suite* スタートガイド (本書)
- *ARM FLEXlm* ライセンス管理ガイド
- *RealView* デバッガエッセンシャルガイド
- *RealView* デバッガユーザガイド
- *RealView* デバッガターゲットコンフィグレーションガイド
- *RealView Debugger Trace User Guide*
- *RealView Debugger RTOS Guide*
- *RealView* デバッガコマンドラインリファレンスガイド
- *RealView Compilation Tools* 基本操作ガイド
- *RealView Compilation Tools* デベロッパガイド
- *RealView Compilation Tools* アセンブラガイド
- *RealView Compilation Tools* コンパイラ/ライブラリガイド
- *RealView Compilation Tools* リンカ/ユーティリティガイド
- *RealView ARMulator ISS* ユーザガイド
- *RealView Development Suite AXD and armsd Debuggers Guide*
- *RealView Development Suite CodeWarrior IDE Guide*
- *RealView Development Suite* 用語集

ARM Limited および各社の関連する出版物については、各マニュアルの「参考資料」を参照して下さい。

1.3.2 オンラインでの詳細情報の取得

インストールのタイプに応じて、付属マニュアルはすべてオンラインで入手できます。ファイル形式は、Windows および Sun Solaris 向けのマニュアルは PDF および DynaText、Red Hat Linux 向けのマニュアルは PDF で提供しています。PDF ファイルと DynaText ファイルには、同じ情報が記載されています。このマニュアルは、P. 1-3 表 1-1 で説明したマニュアル用のディレクトリにインストールされます。

RVDS v3.0 には、新しい PDF 形式の付属マニュアルが用意されています。この一連のマニュアルには、Collection.pdf という 1 つの PDF からアクセスできます。付属マニュアルをすべてインストールすると、すべての PDF ファイルのテキスト検索が可能になります。

詳細については、以下のセクションを参照して下さい。

- Windows 環境に RVDS をインストールした場合
- Sun Solaris および Red Hat Linux 環境に RVDS をインストールした場合

Windows 環境に RVDS をインストールした場合

Windows システムおよび Sun Solaris システムでは、マニュアルをオンラインで DynaText 形式の電子ブックとしても入手できます。DynaText 形式のマニュアルの内容は、PDF 形式のマニュアルの内容と同じです。

Windows の [スタート] メニューから [プログラム] → [ARM] を選択します。ここで以下のいずれかを選択します。

- PDF 形式のファイルを表示するには、[RealView Development Suite v3.0] → [RVDS v3.0 Documentation Suite] を選択します。
- DynaText 形式のファイルを表示するには、[DynaText Documentation] を選択します。

Sun Solaris および Red Hat Linux 環境に RVDS をインストールした場合

すべての付属マニュアルは、Sun Solaris の場合は PDF 形式および DynaText 形式、Red Hat Linux の場合は PDF 形式で入手できます。デスクトップリンクを設定した場合、これらのリンクを使用して必要なマニュアルにアクセスします。

注

Red Hat Linux 環境では、DynaText 形式のファイルはインストールされません。

1.4 RealView Development Suite のサンプル

RVDS のマニュアルにあるサンプルの多くのコードは、主なサンプルディレクトリ（「*RealView Development Suite* のインストール、サンプル、およびマニュアルのディレクトリ」 (P.1-2) 参照) にあります。

また、このディレクトリには、マニュアルで説明されていないサンプルコードも含まれています。詳細については、各サンプルディレクトリ内の `readme.txt` を参照して下さい。サンプルは以下のサブディレクトリにインストールされます。

<code>asm</code>	ARM アセンブリ言語プログラミングの一部のサンプル。これらのサンプルは、 <i>RealView Compilation Tools</i> アセンブラガイドで使用されています。
<code>cached_dhry</code>	Dhrystone サンプルをベースにビルドされた、キャッシュと TCM を初期化するルーチンのサンプル。
<code>Cortex-M3</code>	ARM Cortex™-M3 プロセッサ用の Hello World サンプル。このサンプルには、サンプルスキッタファイルとビルドスクリプトが含まれています。
<code>cpp</code>	基本的な C++ サンプルの一部。
<code>databort</code>	標準的なデータアポートハンドラの設計マニュアルとサンプルコード。
<code>dcc</code>	デバッグ通信チャネルの使用法を示すサンプルコード。このサンプルについては、 <i>RealView Compilation Tools</i> デベロッパガイドで説明されています。
<code>dhrystone</code>	Dhrystone ベンチマーク。このサンプルは <i>RealView Debugger</i> のマニュアルで使用されています。
<code>dsp</code>	ヘッダファイル <code>dspfns.h</code> を十分に活用するのに必要な小さいソースファイル。このファイルでは、DSP タイプの一連のプリミティブ動作を定義し、ARM コンパイラのインラインアセンブリ機能の使用法を示します。
<code>emb_sw_dev</code>	<i>RealView Compilation Tools</i> デベロッパガイドの組み込みソフトウェア開発に関する章で参照されているサンプルプロジェクト。このサンプルには、次のサブディレクトリが含まれています。
<code>buildn</code>	サンプルプロジェクトをビルドするためのバッチファイルおよびメークファイル。各プロジェクトの説明については、関連する <code>readme.txt</code> ファイルを参照して下さい。
<code>dhry</code>	Dhrystone ベンチマークプログラム用のソースファイル。このプログラムでは、各 <code>buildn</code> ディレクトリ内のサンプルプロジェクトのコードベースを提供しています。
<code>include</code>	ユーザ定義ヘッダファイル。

<code>scatter</code>	サンプルプロジェクトのビルドに使用されるスキヤッタファイル。
<code>source</code>	サンプルプロジェクトのビルドに必要なその他すべてのソースファイル。
<code>fft_v5te</code>	ARM アーキテクチャ v5TE (ARMv5TE) 用に最適化された Fast Fourier Transform コード。
<code>inline</code>	ARM の C コードおよび C++ コードをコンパイルする際のインラインアセンブラの使用法を示すサンプル。詳細については、 <i>RealView Compilation Tools</i> デベロッパガイドの C、C++、およびアセンブリ言語の混在に関する章を参照して下さい。
<code>interwork</code>	ARM コードと Thumb コード間のインターワークを示すサンプル。詳細については、 <i>RealView Compilation Tools</i> デベロッパガイドの ARM と Thumb のインターワークに関する章を参照して下さい。
<code>mmugen</code>	MMUgen ユーティリティのソースとマニュアル。このユーティリティでは、必要とされる仮想アドレスから物理アドレスへの変換について記述するルールファイルから MMU ページテーブルのデータを生成できます。
<code>picpid</code>	位置非依存コードの記述方法のサンプル。
<code>sorts</code>	ARM の C ライブラリで使用される挿入ソート、シェルソート、およびクイックソートを比較するサンプルコード。
<code>svc</code>	サンプル <i>Supervisor Call</i> (SVC) ハンドラ。
<code>unicode</code>	マルチバイト文字のサポートを評価できるようにするサンプルコード。
<code>vfpsupport</code>	ベクタ浮動小数点 (VFP) の操作を有効にして実行するためのサンプルコード。また、VFP の使用時にデバッグシステムを設定するためのさまざまなユーティリティファイル、および PDF 形式の <i>Application Note 133 Using VFP in RVDS</i> も含まれています。

1.5 ARM Developer Suite

また、RVDS には、*ARM Developer Suite*[™] (ADS) v1.2.1 の完全バージョンも含まれています。ADS v1.2.1 は、RVDS のインストール時にはインストールされません。ADS v1.2.1 を使用する必要がある場合は、個別にインストールして下さい。

1.5.1 ADS と RVDS の両方をインストールする際の考慮事項

ADS と RVDS の両方をインストールする場合は以下の点に注意して下さい。

- RVDS に加えて ADS もインストールできますが、ADS と共にインストールされる ADS ARMulator と RealView ARMulator ISS の両方を使用する場合には注意が必要です。詳細については、*RealView Development Suite* のリリースノートを参照して下さい。ADS をインストールしている場合は、ADS デバッガを使用して RealView ARMulator ISS に接続できます (*RealView ARMulator 命令セットシミュレータ* (P. 1-7) 参照)。
- Windows マシンに ADS と RVDS の両方をインストールする場合は、ARM SuiteSwitcher ユーティリティを使用する必要があります。ただし、SuiteSwitcher ユーティリティを使用して、インストールされた異なるバージョンの ADS を切り替えることはできません。SuiteSwitcher の無償ダウンロードについては、ARM Web サイトを参照して下さい。

1.5.2 古い ADS コード生成ツールのインストール

古いバージョンの ADS を使用する必要がある場合、ADS v1.2.1 と同じ CD-ROM に古いバージョンの ADS が収録されています。

Windows 用には、ADS v1.1 および v1.0.1 のインストーラが提供されています。

Sun Solaris および Red Hat Linux 用には、ADS v1.1 および v1.0.1 のコード生成ツールだけが提供されています。インストーラは用意されていないため、ファイルを手動でコピーする必要があります。ファイルは、CD-ROM の `ads_1_1` ディレクトリおよび `ads_1_0_1` ディレクトリにあります。以下の手順を実行して下さい。

1. プラットフォームに必要なコンポーネント、およびそのコンポーネントのコピー先を決定します。
2. 該当プラットフォームのディレクトリから適切な実行可能ファイルを選択します。
3. 必要なライブラリをコピーします。
4. パスに実行可能ファイルの場所が含まれていること、および以下の環境変数を設定していることを確認します。
 - ARMINC - ARM のインクルードファイルが格納されているディレクトリ
 - ARMLIB - ARM のライブラリファイルが格納されているディレクトリ

1.6 ターゲットアクセスのサポート

RVDS v3.0 および ADS v1.2.1 の ARM デバッガでサポートされているターゲットアクセスの概要については、以下のセクションで説明します。

- *Windows 環境での RVDS ターゲットアクセスのサポート*
- *Sun Solaris および Red Hat Linux 環境での RVDS ターゲットアクセスのサポート (P. 1-17)*
- *ADS v1.2.1 ターゲットアクセスのサポート (P. 1-18)*

1.6.1 Windows 環境での RVDS ターゲットアクセスのサポート

Windows 環境で RVDS の ARM デバッガによってサポートされているターゲットアクセスを、表 1-3 に示します。

表 1-3 Windows 環境でサポートされているターゲットアクセス

ターゲットアクセス	RealView Debugger v3.0 以降	AXD v1.3.1	armsd
Agilent Debug Interface (ADI)	○	○	
Angel デバッグモニタ (Remote_A)	○	○	○
Instruction Set System Model (ISSM)	○		
Multi-ICE	○	○	
MultiTrace™	○		
RealMonitor	○	○	
RealView ARMulator ISS	RealView Simulator Broker (localhost) のみ	RDI のみ	RDI のみ
RealView ICE	○		
RealView Trace	○		
Versatile プラットフォーム上の組み込み RealView ICE Micro Edition (USB 接続)	○		

以下の点に注意して下さい。

- Versatile プラットフォームに対して USB 接続を使用するには、カスタムインストールを実行し、RealView ICE Micro Edition の USB デバッグサポート用のオプションを選択する必要があります。
- RealView Trace ソフトウェアは、RealView Debugger と共に自動的にインストールされます。
- RealView Debugger では、RealMonitor と RealView ICE を併用できるようになりました。RealMonitor と RealView ICE を併用する方法の詳細については、*RealView デバッガターゲットコンフィグレーションガイド*を参照して下さい。
- MultiTrace をサポートするには、ARM トレースデバッグツール (TDT) をインストールする必要があります。
- インストール後には、RealView Debugger で、ADI DLL (gateway.dll) を ARM-A-RR ターゲットリストに追加する必要があります。詳細については、*RealView デバッガターゲットコンフィグレーションガイド*の RDI ターゲットの操作の説明を参照して下さい。

1.6.2 Sun Solaris および Red Hat Linux 環境での RVDS ターゲットアクセスのサポート

Sun Solaris および Red Hat Linux 環境で RVDS の ARM デバッガによってサポートされているターゲットアクセスを、表 1-4 に示します。

表 1-4 Sun Solaris および Red Hat Linux 環境でサポートされているターゲットアクセス

ターゲットアクセス	RealView Debugger v3.0 以降	armsd
ISSM (MxDI)	○	
RealView ARMulator ISS	RealView Simulator Broker (localhost) のみ	RDI のみ
RealMonitor	○	
RealView ICE	○	

RealView Debugger では、RealMonitor と RealView ICE を併用できるようになりました。RealMonitor と RealView ICE を併用する方法の詳細については、*RealView デバッガターゲットコンフィグレーションガイド*を参照して下さい。

1.6.3 ADS v1.2.1 ターゲットアクセスのサポート

ADS v1.2.1 の AXD デバッガでは、以下のターゲットアクセスをサポートしています。

- RDI インタフェースのみを使用した RealView ARMulator ISS
- RealMonitor
- Multi-ICE
- MultiTrace (TDT をインストールする必要があります)
- Agilent Debug Interface (Gateway)
- Angel デバッグモニタ (Remote_A)

1.7 RVDS 環境で問題を修正する方法

RVDS のコンポーネントアプリケーションを実行中に問題が発生した場合は、RVDS 環境が正しく設定されていることを確認します。

- Sun Solaris や Red Hat Linux 環境では、シェルの種類に応じて、RVDS30env.sh スクリプトまたは RVDS30env.csh スクリプトのいずれかを実行します。Sun Solaris または Red Hat Linux 環境で RVDS 環境を設定する際には、この方法を使用することをお勧めします。このスクリプトの実行方法の詳細については、*RealView Development Suite* インストールガイドを参照して下さい。
- Windows 環境に RVDS をインストールすると、インストール中に詳細セットアップオプションの **[Update environment variables in registry]** オプションの選択を解除した場合を除いて、armenv ユーティリティが自動的に実行されます。ただし、インストールした後、armenv ユーティリティを使用して RVDS 環境を変更できます。armenv ユーティリティの使用方法の詳細については、「付録 A armenv ツールの使用」を参照して下さい。このユーティリティは、Sun Solaris システムおよび Red Hat Linux システムでも使用できます。

注

本リリースでは、armenv ユーティリティをカスタムインストールで使用できません。Windows システムでカスタムインストールを実行した場合は、手動で環境変数を設定する必要があります（「主な RVDS 環境変数」(P. 1-20) 参照）。Sun Solaris システムや Red Hat Linux システムでは、RVDS30env.sh スクリプトまたは RVDS30env.csh スクリプトのいずれかを使用します。

1.7.1 主な RVDS 環境変数

表 1-5 では、設定する必要がある主な RVDS 環境変数を示しています。... はインストール先のパス要素と置き換えて下さい。可能な場合は、「RVDS 環境で問題を修正する方法」(P. 1-19) で説明した推奨方法を使用してこれらの変数を設定します。また、PATH 環境変数にさまざまな RVDS コンポーネントアプリケーション実行可能ファイルの場所が含まれていることを確認します。

表 1-5 Windows 環境の主な RVDS 環境変数

環境変数	設定
ARMROOT	インストールディレクトリのルート (<i>install_directory</i>)。Windows 環境でのデフォルトの場所は次のとおりです。 C:\Program Files\ARM
ARMCONF	RVISS およびさまざまな RDI ターゲットコンフィグレーションファイルの検索に使用されます。 <i>install_directory</i> \RDI\armperip\...\... <i>install_directory</i> \RVARMulator\v6ARMulator\...\...\win_32-pentium、 <i>install_directory</i> \RVARMulator\ARMulator\...\...\win_32-pentium
ARMDLL	RVISS およびさまざまな RDI ターゲット DLL ファイルの検索に使用されます。 <i>install_directory</i> \RVARMulator\v6ARMulator\...\...\win_32-pentium、 <i>install_directory</i> \RVARMulator\ARMulator\...\...\win_32-pentium、 <i>install_directory</i> \ARM\RDI\Targets\Remote_A\...\...\win_32-pentium、 <i>install_directory</i> \RDI\rdimsvr\...\...\win_32-pentium
ARMLMD_LICENSE_FILE	RVDS ライセンスファイルの場所。この環境変数の詳細については、 <i>ARM FLEXlm</i> ライセンス管理ガイドを参照して下さい。
RVCT30BIN	RVCT プログラム実行可能ファイル。 <i>install_directory</i> \RVCT\Programs\...\...\win_32-pentium
RVCT30INC	RVCT コンパイラのインクルードファイル。 <i>install_directory</i> \RVCT\Data\...\...\include\windows
RVCT30LIB	RVCT コンパイラのインクルードファイル。 <i>install_directory</i> \RVCT\Data\...\...\lib
RVDEBUG_HLPPATH	RealView Debugger のオンラインヘルプファイル。 <i>install_directory</i> \Documentation\RVD\...\release\windows\onlinehelp
RVDEBUG_INSTALL	RealView Debugger の実行可能ファイル。 <i>install_directory</i> \RVD\Core\...\...\win_32-pentium

第 2 章

RVDS v3.0 の機能

本章では、*RealView® Development Suite* (RVDS) v3.0 の新機能と、RVD v1.8 からの主な変更点について説明します。本章には以下の情報が含まれます。

- *RVDS v3.0* の変更点 (P. 2-2)
- *RealView Debugger* の変更点 (P. 2-3)
- *RealView Compilation Tools* の変更点 (P. 2-4)
- シミュレータのサポート (P. 2-5)
- *CodeWarrior for RVDS* の変更点 (P. 2-6)
- マニュアルの変更点 (P. 2-7)

2.1 RVDS v3.0 の変更点

RVDS v3.0 の主な変更点については、以下のセクションで説明します。

- RVDS v3.0 の新機能
- 今後廃止される機能と廃止された機能

2.1.1 RVDS v3.0 の新機能

RVDS v3.0 の新機能を以下に示します。

- TrustZone® アーキテクチャがサポートされます。
- ARMv7 用の *Thumb®-2 Execution Environment* (Thumb-2EE) がサポートされます。
- 以下の ARM Cortex™ プロセッサファミリがサポートされます。
 - Cortex-A8
 - Cortex-M3
- Cortex-A8 プロセッサと Cortex-M3 プロセッサ用のシミュレータモデルが使用できるようになりました。これらのモデルには、*RealView Debugger* の新しい *Instruction Set System Model* (ISSM) ターゲットアクセスを經由してアクセスできます。
- Linux アプリケーションのデバッグに対するサポートが追加されました。

2.1.2 今後廃止される機能と廃止された機能

今後廃止される機能と RVDS v3.0 で廃止された機能を以下に示します。

- *ARM eXtended Debugger* (AXD) と *ARM シンボリックデバッガ* (armsd) のサポートは廃止される予定です。
- *CodeWarrior* のメイクファイルインポータ機能と *Batch File Runner* 機能は廃止される予定です。
- *Multi-ICE®* ダイレクト接続を經由した *RealView Debugger* のリモート接続に対するサポートは廃止されました。つまり、DSP プロセッサに接続するには、*RealView ICE* を使用する必要があります (*RealView ICE* は、個別に購入する必要があります)。
- *RealView Debugger* のプロジェクトマネージャとその関連機能は廃止されました。

その他の廃止予定の機能については、以下を参照して下さい。

- *RealView Compilation Tools 基本操作ガイド*
- *RealView デバッガエッセンシャルガイド*

2.2 RealView Debugger の変更点

RealView Debugger v3.0 の主な変更点を以下に示します。

- RealView Debugger がシングルプロセスとして実行されるようになりました。Target Vehicle Server (TVS) が別のエンティティとして存在することはなくなります。
- [Connection Control] ウィンドウが再設計されました。詳細については、*RealView デバッガユーザガイド*を参照して下さい。
- [Synch] タブの機能が [Synchronization Control] ウィンドウで使用できるようになりました。詳細については、*RealView デバッガユーザガイド*を参照して下さい。
- レジスタ枠が再設計されました。選択したレジスタを [User] タブにコピーして、ユーザー固有のビューを作成できるようになりました。詳細については、*RealView デバッガユーザガイド*を参照して下さい。
- Linux アプリケーションのデバッグに対するサポートが追加されました。詳細については、*RealView Debugger RTOS Guide* を参照して下さい。
- RealView Debugger のプロジェクトマネージャとその関連機能が廃止されたため、RealView Debugger 内ではプロジェクトの作成やイメージのビルドを実行できなくなりました。ただし、ソースコードの編集および検索機能は引き続き使用できます。

—— 注 ——

RVDS v3.0 でプロジェクトを作成してビルドするには、CodeWarrior for RVDS を使用します（「CodeWarrior for RVDS」(P. 1-3) 参照）。

- シミュレータのサポートが変更されました。詳細については、「シミュレータのサポート」(P. 2-5) を参照して下さい。
- RealView Broker (RVBroker) が再設計されました。RealView Debugger は、ローカルホスト (RVISS) 接続に対しては引き続き自動的に RVBroker を実行しますが、リモートシミュレータの接続に対する RVBroker の起動方法が変更されました。リモートワークステーションで RVBroker を起動するときは、ユーザー名の指定が必要になりました。詳細については、*RealView デバッガターゲットコンフィグレーションガイド*を参照して下さい。

—— 注 ——

Multi-ICE ダイレクト接続のサポートは、RVDS v3.0 で廃止されました。

RealView Debugger の変更点の詳細については、*RealView デバッガエッセンシャルガイド*を参照して下さい。

2.3 RealView Compilation Tools の変更点

RealView Compilation Tools (RVCT) v3.0 の主な変更点を以下に示します。

- RVCT v3.0 では、ARMv7 用の Thumb-2EE がサポートされます。
- ARM アセンブラを使用することにより、インテルワイヤレス MMX テクノロジーの命令をアセンブルし、PXA270 プロセッサ用のコードを開発できます。
- RVCT v3.0 では、*ABI for the ARM Architecture (base standard)* [BSABI] に記載されているように、DWARF 3 (Draft Standard 9.6) のデバッグテーブルが完全にサポートされます。
- ARM コンパイラおよびリンカでスレッドローカル記憶域 (TLS) をサポートすることにより、プログラムで複数のスレッドを使用できます。
- ARM コンパイラで、強化されたループの最適化がサポートされます。

RVCT の変更点の詳細については、*RealView Compilation Tools 基本操作ガイド*を参照して下さい。

2.4 シミュレータのサポート

RVDS では、以下のシミュレータのサポートが提供されるようになりました。

- Cortex-A8 プロセッサと Cortex-M3 プロセッサをシミュレートする ISSM がサポートされます。
- RealView ARMulator[®] ISS (RVISS) で、MPCore[™] をシミュレーションのターゲットとして使用できるようになりました。ただし、これによって複数のプロセッサがモデル化されるわけではないため、このモデルに接続しても 1 つのプロセッサにしか接続されません。

RDI ARMulator はシミュレーションのターゲットとして使用できなくなりました。このため、以下のいずれかを使用します。

- localhost ターゲットアクセスで new_arm 接続を使用し、RVISS を使用してシミュレートする ARM[®] プロセッサに接続します。
- ISSM ターゲットアクセスを使用し、いずれかの Cortex モデルに接続します。

これらは RealView Debugger と共にインストールされます。

2.5 CodeWarrior for RVDS の変更点

CodeWarrior for RVDS の主な変更点を以下に示します。

- External Build Wizard がサポートされるようになりました。このウィザードは、廃止予定のメイクファイルインポータ機能と Batch File Runner 機能に置き換わるものです。
 - ファイル拡張子 .cc のサポートが追加されました。
 - 認識されないソースファイルの拡張子 (.cmd など) を使用すると警告メッセージが表示されるようになりました。
 - [New] ダイアログボックスでサポートされないエントリを以下に示します。
 - [Project] タブの [PowerParts Application Wizard] エントリと [PowerParts Component Wizard] エントリ
 - [File] タブの [Component Catalog File] エントリと [PowerParts Palette Wizard] エントリ
 - [Object] タブの [PowerParts Form Wizard] エントリ
 - Compilation Tools の変更に応じて、パネルの設定が追加または削除されました。詳細については、*RealView Compilation Tools 基本操作ガイド*を参照して下さい。
- 詳細については、*RealView Development Suite CodeWarrior IDE Guide* を参照して下さい。

2.6 マニュアルの変更点

RVDS の新機能に関する説明を除いて、RVDS のマニュアルで主に変更された点は、RealView Debugger のマニュアルです。RealView Debugger のマニュアルは、以下のように再編成されました。

- *RealView デバッガ拡張機能ユーザガイド*の情報は、次のマニュアルに移動しました。
 - DSP のサポートに関する説明が記載された章は、*RealView デバッガユーザガイド*に移動しました。
 - 複数ターゲットのデバッグに関する説明が記載された章は、*RealView デバッガユーザガイド*に移動しました。
 - RealView Debugger のトレースに関する説明が記載された章は、*RealView Debugger Trace User Guide*に移動しました。
 - OS のサポートに関する説明が記載された章は、*RealView Debugger RTOS Guide*に移動しました。
- *RealView デバッガターゲットコンフィグレーションガイド*のターゲットへの接続に関する説明が記載された章は、*RealView デバッガユーザガイド*に移動しました。
- *RealView デバッガユーザガイド*の構造が、より作業に基づいたものになるように変更されました。
- RealView Debugger のプロジェクトマネージャが廃止されたため、*RealView Debugger Project Management Guide* は提供されません。

RVDS のマニュアルに加えられたその他の詳細な変更点については、以下を参照して下さい。

- *RealView デバッガエッセンシャルガイド*
- *RealView Compilation Tools 基本操作ガイド*

第 3 章

RealView Development Suite の使い方

RealView® Development Suite (RVDS) 付属のコンポーネント製品を使用すると、アプリケーションを構成する複数のイメージをビルドしてデバッグすることができます。本章では、RVDS ツールを使用してイメージをビルドおよびデバッグするための基本タスクについて説明します。本章は以下のセクションから構成されています。

- *ビルドタスクとデバッグタスクの概要* (P. 3-2)
- *サンプルプロジェクトの使用* (P. 3-5)

3.1 ビルドタスクとデバッグタスクの概要

表 3-1 は、RVDS ツールを使用してアプリケーションをビルドおよびデバッグするための主なタスクの手順概要と詳細な手順が記載されているマニュアルを示しています。

参照マニュアルに記載されているタスクは、表 3-1 と異なる順序で記載されている場合があります。初めて RVDS ツールを使用する場合は、参照マニュアルに記載されている順序でタスクを実行することをお勧めします。表 3-1 の順序は、タスクの一般的な実行順序を反映したものです。

表 3-1 主なビルドタスクとデバッグタスク

手順	説明	参照
1	プロジェクトの管理およびビルドに使用する RVDS アプリケーションを選択します。 <ul style="list-style-type: none"> CodeWarrior for RVDS を使用する場合は、手順 3 に進みます。 RVCT を使用してコマンドラインからビルドする場合は、手順 2 に進みます。 	
2	RVCT ビルドツールを直接使用する場合は、必要なビルドコマンドを含むメークファイルまたは Windows コマンドファイルを作成します。手順 9 に進み、RealView Debugger にイメージをロードしてデバッグします。	<i>RealView Compilation Tools 基本操作ガイド</i>
3	CodeWarrior for RVDS を起動します。	<i>RealView Development Suite CodeWarrior IDE Guide</i>
4	既存の CodeWarrior for RVDS プロジェクトがある場合は、手順 6 に進みます。 既存のプロジェクトがない場合は、アプリケーション用に CodeWarrior for RVDS プロジェクトを作成します。	<i>RealView Development Suite CodeWarrior IDE Guide</i>
5	アプリケーションのイメージをビルドするのに必要なビルドターゲット設定を指定します。手順 7 に進みます。	<i>RealView Development Suite CodeWarrior IDE Guide</i>
6	既存の CodeWarrior for RVDS プロジェクトを開きます。	<i>RealView Development Suite CodeWarrior IDE Guide</i>
7	CodeWarrior for RVDS プロジェクトのイメージをビルドします。	<i>RealView Development Suite CodeWarrior IDE Guide</i>

表 3-1 主なビルドタスクとデバッグタスク (続き)

手順	説明	参照
8	使用するイメージを選択します。 <ul style="list-style-type: none"> プリビルドされたサンプルイメージなど、既存のイメージを使用する場合は、手順 9 に進みます。 新しいイメージをビルドする場合は、手順 1 に戻ります。 	サンプルプロジェクトの使用 (P. 3-5)
9	RealView Debugger を起動します。	RealView デバッグエッセンシャルガイド
10	必要に応じて、デバッグターゲットおよび接続を設定します。	RealView デバッグターゲットコンフィグレーションガイド
11	デバッグターゲットに接続します。	<ul style="list-style-type: none"> RealView デバッグエッセンシャルガイド RealView デバッグユーザガイド
12	デバッグ可能なイメージをロードします。	<ul style="list-style-type: none"> RealView デバッグエッセンシャルガイド RealView デバッグユーザガイド
13	ブレークポイント、トレースポイントなど、デバッグ機能を準備します。	<ul style="list-style-type: none"> RealView デバッグエッセンシャルガイド RealView デバッグユーザガイド RealView Debugger Trace User Guide RealView Debugger RTOS Guide
14	イメージを実行します。	<ul style="list-style-type: none"> RealView デバッグエッセンシャルガイド RealView デバッグユーザガイド
15	ステップング、変数やメモリの内容を表示するなど、必要なデバッグタスクと監視タスクを実行します。トレースポイントを使用している場合は、RealView Debugger のトレース解析機能を使用して、トレース出力を解析します。	<ul style="list-style-type: none"> RealView デバッグエッセンシャルガイド RealView デバッグユーザガイド RealView Debugger Trace User Guide RealView Debugger RTOS Guide
16	デバッグセッションの結果によって、次に進む手順が異なります。 <ul style="list-style-type: none"> 問題がある場合は、手順 17 に進みます。 問題がない場合は、最終的なりリリース用のイメージを再ビルドします。 	<ul style="list-style-type: none"> RealView Development Suite CodeWarrior IDE Guide RealView デバッグエッセンシャルガイド RealView Compilation Tools 基本操作ガイド

表 3-1 主なビルドタスクとデバッグタスク (続き)

手順	説明	参照
17	ソースコードにある問題の修正に使用するツールを選択します。 <ul style="list-style-type: none">CodeWarrior for RVDS を使用します。任意のソースエディタを使用します。	<ul style="list-style-type: none"><i>RealView Development Suite CodeWarrior IDE Guide</i>
18	問題を修正したら、イメージを再ビルド、リロード、およびデバッグする必要があります。 <ul style="list-style-type: none">CodeWarrior for RVDS を使用している場合は、手順 6 に戻ります。RVCT を直接使用している場合は、手順 2 に戻ります。	

3.2 サンプルプロジェクトの使用

独自のアプリケーションイメージをビルドしてデバッグする必要がありますが、RVDSのマニュアルに記載されているタスクでは、RVSD 付属のサンプルプロジェクトを使用しています（「RealView Development Suite のサンプル」(P. 1-13) 参照）。

RVDS コンポーネントの機能を熟知するまでは、記載されている指示に従うことをお勧めします。ただし、ユーザマニュアルに記載されている多くのタスクでは、サンプルプロジェクトのファイルを変更する必要があるため、サンプルプロジェクトのファイルを変更する前に、サンプルプロジェクトのファイルおよびディレクトリのバックアップコピーを作成して下さい。

付録 A

armenv ツールの使用

この付録では、ARM® RealView® 製品のインストールの管理に使用できる *armenv* ツールについて説明します。この付録は以下のセクションから構成されています。

- *armenv* ツールについて (P. A-2)
- *armenv* ツールの使用 (P. A-3)

A.1 armenv ツールについて

armenv ツールを使用すると、以下の操作を実行できます。

- ARM RealView 製品の環境変数を設定および削除します。
- インストールした ARM RealView 製品間で競合が発生していないかどうかを確認します。
- 同じ製品のバリエーションを設定します。

注

本リリースの RVDS では、armenv ツールをカスタムインストールで使用できません。

armenv ツールは、以下の場所にあります。

`install_directory/bin/platform`

A.2 armenv ツールの使用

このセクションでは、armenv コマンドの構文について説明し、その使用例を示します。説明する内容は以下のとおりです。

- armenv コマンドの構文
- armenv のコマンドライン引数
- サンプル (P. A-5)

A.2.1 armenv コマンドの構文

armenv ツールのコマンド構文は、以下のとおりです。

```
armenv [-r root] [-u] -p product [[--and] -p product]...[--user|--sys|--proc]
[--bat|--sh|--csh|--posh|--exec program [args]]
```

引数については、「armenv のコマンドライン引数」で説明します。

A.2.2 armenv のコマンドライン引数

表 A-1 に、すべてのプラットフォームで使用できるコマンドライン引数を示します。

表 A-1 汎用的な armenv の引数

引数	説明
--help	コマンドライン引数のヘルプを表示します。
-r root	製品のインストールのルート (<i>install_directory</i>) の絶対パスです。例えば Windows の場合、デフォルトのルートは以下のようになります。 C:\Program Files\ARM
-p product	ARM RealView 製品です。詳細については、「製品の構文」(P. A-5) を参照して下さい。
--and	この引数より前に指定されたすべての製品の設定を計算してから、この引数より後に指定されたすべての製品の設定を計算します。後のグループの設定で前のグループの設定がオーバーライドされます。

表 A-1 汎用的な armenv の引数 (続き)

引数	説明
--proc	現在のプロセスの環境のみを変更します。 Windows 環境では、この引数を --system または --user と共に使用することはできません。
--exec	新しい環境でプログラムを実行します。 Windows 環境では、この引数を --bat と共に使用することはできません。また、Sun Solaris と Red Hat Linux 環境では、この引数を --sh、--csh、または --posh と共に使用することはできません。
-u	製品の設定時に環境に加えた変更を元に戻す操作を試行します。

表 A-2 に、Windows システム固有のコマンドライン引数を示します。

表 A-2 Windows 固有の armenv の引数

引数	説明
--system	Windows システムのレジストリの SYSTEM 領域を更新します。これがデフォルトです。
--user	Windows システムのレジストリの USER 領域を更新します。
--bat	現在のコマンドプロンプトウィンドウの環境を変更します。これがデフォルトです。

表 A-3 に、Sun Solaris および Red Hat Linux システム固有のコマンドライン引数を示します。指定できるのは、これらのうち 1 つのみです。

表 A-3 Sun Solaris と Red Hat Linux 固有の armenv 引数

引数	説明
--csh	csh 構文のシェルスクリプトを生成します。
--sh	sh 構文のシェルスクリプトを生成します。
--posh	移植可能なシェルスクリプトを生成します。 これがデフォルトです。

製品の構文

製品を指定する構文を以下に示します。

```
-p category [name] [version [rev]] [-v name value]...
```

各引数には以下の意味があります。

category 製品の識別子です。例えば、RVDS や RVCT などがあります。

name この引数は使用しないで下さい (armenv ツールでは、デフォルト名の Contents が使用されます)。

version 製品のバージョン番号です。例えば、3.0 などがあります。バージョンを指定しなかった場合、インストールされている製品の最新バージョンが使用されます。

rev 製品の特定のビルド番号です。ビルド番号を指定しなかった場合、インストールされている製品の最新ビルドが使用されます。

```
-v name value
```

同じ製品のバリエントを識別します。

name バリエントの種類です。例えば、platform などがあります。この引数では、platform バリエント以外は使用しないことをお勧めします。

value 特定のバリエントです。例えば、solaris-sparc などがあります。この引数は、RVDS v3.0 の Sun Solaris および Red Hat Linux バリエントの両方がインストールされている場合などに使用します。この引数の使用例については、「サンプル」を参照して下さい。

A.2.3 サンプル

csh シェルと RVDS v3.0 の最新ビルドの Sun Solaris 環境変数を設定するには、以下のように入力します。

```
armenv -r ~/ -p RVDS 3.0 -v platform solaris-sparc --csh
```

RVCT v3.0 と RVCT v2.2 との間で競合が発生していないかどうかを確認するには、以下のように入力します。

```
armenv -p RVCT 3.0 -p RVCT 2.2
```

RVCT v3.0 の設定で RVCT v2.2 の設定をオーバーライドするには、以下のように入力します。

```
armenv -p RVCT 2.2 --and -p RVCT 2.0
```


付録 B

以前のリリースについて

本章では、*RealView® Developer Suite* (RVDS) の以前のリリース (v2.2 SP1、v2.2、v2.1、v2.0、および *ARM Developer Suite™* (ADS) v1.2.1) 間の主な違いについて概説します。以下のセクションで、各リリース間の変更点について説明します。

- *RVDS v2.2 SP1* と *RVDS v2.2* との間の変更点 (P. B-2)
- *RVDS v2.2* と *RVDS v2.1* との間の変更点 (P. B-3)
- *RVDS v2.1* と *RVDS v2.0* との間の変更点 (P. B-6)
- *RVDS v2.2* と *ADS v1.2.1* との間の変更点 (P. B-8)

B.1 RVDS v2.2 SP1 と RVDS v2.2 との間の変更点

このセクションでは、RVDS v2.2 SP1 と RVDS v2.2 との間の変更点について説明します。説明する内容は以下のとおりです。

- マニュアルの変更点
- デバッガのサポート
- ビルドツールのサポート

B.1.1 マニュアルの変更点

マニュアルの変更点を以下に示します。

- CodeWarrior の ARM® 機能を使用する方法が記載された *RealView Developer Suite CodeWarrior IDE Guide* が追加されています。
- CodeWarrior の導入部分に関する説明が記載された章が *RealView Developer Suite スタートガイド* から削除され、*RealView Developer Suite CodeWarrior IDE Guide* に組み込まれています。
- RealView Debugger のマニュアルの、サポートされる DSP に関する説明が変更されています。

B.1.2 デバッガのサポート

RVDS v2.2 SP1 と RVDS v2.2 のデバッグツールに関する主な違いは、RealView Debugger で CEVA-Oak、CEVA-TeakLite、CEVA-Teak、ZSP400、および ZSP500 DSP がサポートされるようになったことです。

B.1.3 ビルドツールのサポート

RVDS v2.2 SP1 と RVDS v2.2 のビルドツールの違いはわずかです。詳細については、*RealView Compilation Tools 基本操作ガイド*を参照して下さい。

B.2 RVDS v2.2 と RVDS v2.1 との間の変更点

このセクションでは、RVDS v2.2 と RVDS v2.1 との間の変更点について説明します。説明する内容は以下のとおりです。

- IDE のサポート
- デバッガツールのサポート
- ビルドツールのサポート (P. B-4)
- Agilent プローブのサポート (P. B-5)

B.2.1 IDE のサポート

RealView Debugger IDE の代わりに CodeWarrior IDE が提供されます。RVDS v2.2 の CodeWarrior IDE は、Metrowerks CodeWarrior v5.6 をベースにしています。

————— 注 —————

RVDS v2.2 の CodeWarrior for RVDS は、Windows XP および Windows 2000 システムでのみサポートされ、Sun Solaris と Red Hat Linux ではサポートされません。

CodeWarrior for RVDS の詳細については、「*CodeWarrior for RVDS*」(P. 1-3) を参照して下さい。

————— 注 —————

CodeWarrior for ADS を使用している場合は、「*CodeWarrior IDE の変更点*」(P. B-8) を参照して下さい。

B.2.2 デバッガツールのサポート

RVDS v2.2 と RVDS v2.1 のデバッグツールに関する主な違いは、RealView Debugger に以下の機能が追加されたことです。

- 強化されたメニューの構造
- 強化された枠処理メカニズム
- データナビゲータ枠により強化されたデータナビゲーション
- 国際化のサポート
- 強化されたソースコードの色分け
- トレース、解析、およびプロファイリングの拡張機能
- 強化された RTOS のサポート
- gcc でビルドされたイメージのサポート
- 新しい CLI コマンドの PRINTDSM と TRACEEXTCOND

また、RealView Debugger のスタンドアロンエディタと Vi 編集モードに対するサポートが廃止されています。

詳細な変更点の一覧については、*RealView デバッグエッセンシャルガイド*を参照して下さい。

B.2.3 ビルドツールのサポート

RVDS v2.2 と RVDS v2.1 のビルドツールに関する主な違いを以下に示します。

- RVCT v2.2 では、新しい ARMv6 コアがサポートされます。例えば、ARM TrustZone® テクノロジー向けに最適化したソフトウェアが組み込まれた ARM1176JZF-S™ や、ARM968EJ-S、ARM1156T2F-S™、ARM MPCore™ などがあります。
- RVCT v2.2 で使用できる新しい Thumb®-2 命令セットにより、多くの新しい 32 ビット命令と、いくつかの新しい 16 ビット命令が導入されます。
Thumb-2 命令セットには、以前の 16 ビット Thumb 命令がサブセットとして含まれています。
- RVCT v2.2 は、*Base Platform ABI for the ARM Architecture* [BPABI] (未公開のドラフト) を完全に順守しています。
- RVCT v2.2 では、*ABI for the ARM Architecture (base standard)* [BSABI] に記載されているように、DWARF 3 (Draft Standard 9) デバッグテーブルの初期サポートが提供されます。
- コマンドラインオプション `-g` を指定すると、現在のコンパイル用のデバッグテーブルが生成されます。最適化オプションを指定する場合は、`-O` を使用します。デフォルトでは、`-g` オプションを使用しても、最適化の設定には影響しません。
これは、RVCT v2.2 で変更された動作です。
- RVCT v2.2 では、コマンドラインオプション `--apcs /fpic` がサポートされており、System V の共有ライブラリと互換性があるコードをコンパイルできます。
- ARM リンカで、共有ライブラリのビルドと、共有ライブラリに対するリンクがサポートされます。また、SVr4 の実行可能ファイルと共有オブジェクトをビルドしたり、コードの生成方法を指定したりするための新しいコマンドラインオプションも使用できます。
- ARM リンカで、GNU 拡張シンボルバージョン管理モデルがサポートされます。
- 浮動小数点演算に関する ARM の実装が、C99 関数のサポートを強化するように変更されています。これによって大きく動作が変更された場合、開発者は新しく導入された互換性モードを使用して、容易にコードを移行して新機能を使用できます。
- RVCT v2.2 では、Linux アプリケーションと共有ライブラリのビルドがサポートされます。

詳細な変更点の一覧については、*RealView Compilation Tools 基本操作ガイド*を参照して下さい。

B.2.4 Agilent プローブのサポート

RVDS v2.2 では、Agilent プローブがカスタムインストールオプションとしてサポートされます。

B.3 RVDS v2.1 と RVDS v2.0 との間の変更点

このセクションでは、RVDS v2.1 と RVDS v2.0 との間の変更点について説明します。説明する内容は以下のとおりです。

- デバッガツールのサポート
- ビルドツールのサポート

B.3.1 デバッガツールのサポート

RVDS v2.1 と RVDS v2.0 のデバッグツールに関する主な違いを以下に示します。

- *ARM eXtended Debugger* (AXD) が含まれています (「*ARM eXtended Debugger*」(P. 1-9) 参照)。
- *ARM* シンボリックデバッグ (armsd) が含まれています (「*ARM* シンボリックデバッグ」(P. 1-9) 参照)。
- RealView Debugger に以下の機能が追加されています。
 - トレースとプロファイリングの拡張機能
 - 強化された RTOS のサポート
 - よく使用する機能にすばやくアクセスできる新しいツールバーボタンと変更されたメニュー

B.3.2 ビルドツールのサポート

RVDS v2.1 と RVDS v2.0 のビルドツールに関する主な違いを以下に示します。

- *Application Binary Interface for the ARM Architecture (Base Standard)* (ABI for the ARM Architecture (Base Standard)) への準拠が強化されています。 <http://www.arm.com/> の「ABI for the ARM Architecture」ページを参照して下さい。
- C++ の例外処理がサポートされます。このため、エクスポートテンプレートを除く、*ISO/IEC 14822 :1998 International Standard for C++* で定義されている ISO C++ がすべてサポートされます。
- 複数ファイルのコンパイルやリンカのフィードバックなど、より多くの最適化機能が含まれています。
- イメージのサイズをさらに小さくするために、読み取り / 書き込みデータ領域が圧縮されます。
- 一部の GNU C および C++ の拡張機能がサポートされます。
- 多くの新しいコマンドラインオプションがビルドツールに追加されています。
- 単一ダッシュキーワードと一部のコマンドラインオプションが廃止予定になっています。

注

これらのツールでは、8 バイトのアライメントを保持しているかどうかの確認が、より厳密に行われるようになっていきます。コンパイラでは、**PRESERVE8** と **REQUIRE8** を使用してコードを生成します。リンカでは、8 バイトのアライメントを必要とするコードが、8 バイトのアライメントを保持するコードだけを呼び出しているかどうかを確認します。このため、この変更は、古いアセンブラコード、オブジェクトファイル、およびライブラリに影響があります。既存のアセンブリファイル、オブジェクトファイル、またはライブラリが 8 バイトのアライメントを保持しているかどうかを確認し、必要に応じて修正する必要があります。詳細については、*RealView Compilation Tools* アセンブラガイドと *RealView Compilation Tools* リンカ/ユーティリティガイドを参照して下さい。

B.4 RVDS v2.2 と ADS v1.2.1 との間の変更点

このセクションでは、RVDS v2.2 と ADS v1.2.1 との間の変更点について説明します。説明する内容は以下のとおりです。

- *CodeWarrior IDE* の変更点
- デバッガの変更点 (P. B-9)
- ビルドツールの変更点 (P. B-9)
- ARM シミュレータの変更点 (P. B-11)

B.4.1 CodeWarrior IDE の変更点

CodeWarrior for RVDS と CodeWarrior for ADS との間の変更点を以下に示します。

- CodeWarrior for ADS は CodeWarrior v4.2 をベースにしていたましたが、CodeWarrior for RVDS は、Metrowerks CodeWarrior v5.6 をベースにしています。
- CodeWarrior v4.2 で Perl スクリプトを処理するためのサポートを提供していた CodeWarrior Perl プラグインの MWPerl は、CodeWarrior v5.6 では廃止されています。今後 Metrowerks ではこのプラグインはサポートされません。
- ARM ツール固有のコンフィグレーションパネルが RVDS v2.2 に合わせて変更されています。
- RVDS v2.2 では、個別の ARM コンパイラが 1 つのコンパイラに統合されたため、コンパイラのコンフィグレーションパネルが 1 つになっています。
- AXD や armsd だけでなく、RealView Debugger を使用してイメージを実行およびデバッグすることができます。
- ライブラリを連結できます。
- CodeWarrior for ADS プロジェクトを CodeWarrior for RVDS にインポートできます。
- CodeWarrior for RVDS のデフォルトの ARM ステーションナリには、DebugRel ビルドターゲットは含まれていません。ただし、CodeWarrior for ADS プロジェクトをインポートすると、DebugRel ビルドターゲットが作成され、ビルドターゲット用に行った設定が維持されます。
- ADS コンパイラとは異なり、RVCT コンパイラではブラウザ情報が生成されません。この機能は、CodeWarrior の組み込み言語パーサによって提供されます。
- コードのフォーマットが強化されます。
- C++ テンプレートクラスのコードの完了を含むコードの完了が提供されます。
- 次へ / 戻る機能が提供されます。
- 印刷時のテキストの折り返し機能が提供されます。

- ソースに関連する `#includes` がサポートされます。
- 検索対象にコメント部分の参照 / 非参照を選択できます。
- 言語パーサの速度とフィードバックが強化されます。
- 新しいエディタのバインドが提供されます。
- プロジェクトウィンドウの [Code] 列と [Data] 列の表示 / 非表示を切り替えることができます。
- ワークスペースがサポートされます。

— 注 —

CodeWarrior IDE のすべてのターゲット接続とデバッグ機能は、CodeWarrior for RVDS では使用できません。これらの機能を実行するには、いずれかの ARM デバッガを実行する必要があります。

B.4.2 デバッガの変更点

RVDS v2.2 と ADS v1.2.1 のデバッグツールに関する主な違いを以下に示します。

- 最新の ARM デバッガである RealView Debugger を使用すると、以下のような高度なデバッグ機能を実行できます。
 - マルチプロセッサデバッグ
 - OS 認識デバッグ
 - 強化されたターゲットの可視性
 - トレース、分析、およびプロファイリング
 - Ethernet と USB を経由した RealView ICE JTAG 制御ユニットへのアクセス
- AXD が拡張され、RVDS v2.2 付属の新しい RealView Compilation Tools を使用してビルドされた C および C++ プログラムをデバッグできるようになります。

B.4.3 ビルドツールの変更点

RVDS v2.2 と ADS v1.2.1 のビルドツールに関する主な違いを以下に示します。

- 新しい ABI for the ARM Architecture (Base Standard) を順守しています。
<http://www.arm.com/> の「ABI for the ARM Architecture」ページを参照して下さい。この仕様は古い ADS ABI とは異なり、`--apcs /adsabi` コマンドラインオプションと一部互換性があります。

- EDG (Edison Design Group) のフロントエンドを使用することにより、*ISO/IEC 14822 :1998 International Standard for C++* で定義されている ISO C++ が完全にサポートされます。これには例外、ネームスペース、テンプレート、およびランタイム型情報 (RTTI) のインテリジェント実装が含まれますが、テンプレートのエクスポートは含まれません。
- 一部の GNU 言語の拡張機能がサポートされます。
- ARM と Thumb のコンパイルを関数単位で実行します。
- インラインアセンブラが再設計されています。また、新しい組み込みアセンブラを使用すると、アウトオブラインアセンブリコードを含めることが可能になります。
- 使用されていない関数を削除するようにリンカからフィードバックが提供されます。
- ARM アーキテクチャ v6 命令の完全なサポートが追加されています。
- 読み取り / 書き込みデータ圧縮により、ROM サイズを最適化できます。
- 使用されていない C++ 仮想関数が削除されます。
- 複数のコンパイルユニットにまたがって最適化を実行する、複数ファイルのコンパイルが提供されます。
- ライブラリ検索パスを指定し、ユーザーライブラリの検索先を指定することができます。
- RO コードとデータを別の実行領域に分離できます。
- 新しい分散ロード属性が提供されます。
- Unicode 文字とマルチバイト文字がサポートされます。
- 関数の復帰アドレス、現在のスタックポインタの値、および現在のプログラムカウンタの値にアクセスするためのコンパイラ組み込み関数を使用できます。追加の組み込み関数を使用すると、C または C++ コードに BKPT 命令を挿入できます。
- 値を返さない関数を識別し、コンパイラからより効率的なコードが生成されるようにすることができます。
- C++ の名前マングルスキームが変更されています。
- ARM プロファイラ (armprof) が RVCT 付属のツールではなくなりました。
- ARM アプリケーションライブラリが RVCT 付属ツールではなくなりました。
- ADS コンパイラとは異なり、RVCT コンパイラではブラウザ情報が生成されません。

- アセンブラ、コンパイラ、およびリンカのコマンドラインオプションが変更されています。

二重ダッシュ (--) を使用したコマンドラインキーワードの指定 (--cpp など) と、単一ダッシュ (-) を使用した 1 文字のコマンドラインオプションの指定 (-S など、引数の有無は問いません) がサポートされます。

— 注 —

ADS と RVCT の以前のバージョンで使用されていた単一ダッシュコマンドラインオプションは、以前のバージョンとの互換性を維持するために引き続きサポートされます。

- fromelf オプション -ihf が廃止されています。

— 注 —

ツールでは、8 バイトのアライメントを保持しているかどうかの確認が、より厳密に行われるようになっていきます。コンパイラでは、PRESERVE8 と REQUIRE8 を使用してコードを生成します。リンカでは、8 バイトのアライメントを必要とするコードが、8 バイトのアライメントを保持するコードだけを呼び出しているかどうかを確認します。このため、この変更は、古いアセンブラコード、オブジェクトファイル、およびライブラリに影響があります。既存のアセンブリファイル、オブジェクトファイル、またはライブラリが 8 バイトのアライメントを保持しているかどうかを確認し、必要に応じて修正する必要があります。詳細については、*RealView Compilation Tools アセンブラガイド* と *RealView Compilation Tools リンカ/ユーティリティガイド* を参照して下さい。

B.4.4 ARM シミュレータの変更点

RealView ARMulator® ISS は、ARM シミュレータの最新バージョンです。このシミュレータでは、RealView Connection Broker と RDI を経由した接続がサポートされています。RealView Debugger で RealView Connection Broker を経由してシミュレータに接続すると、シミュレータに対して複数の接続を確立できます。RealView Debugger、AXD v1.3、および armsd を使用すると、RealView ARMulator ISS の RDI インタフェースに接続できます。

— 注 —

RealView Development Suite v2.2 に加えて ADS をインストールすることもできますが、RealView ARMulator ISS と ADS ARMulator の両方を使用する場合は注意が必要です。詳細については、*RealView Developer Suite v2.2 Release Notes* を参照して下さい。

以前のリリースについて