

32ビットVシリーズ用ソフトウェア開発環境

橋本 一也 堂田 政志

日本電気(株) C&C共通ソフトウェア開発本部 基本システム開発部

当社では、オリジナル32ビットMPU(V60/V70/V80)と、その開発環境を提供している。本稿では、V60/V70/V80の新機能に対応したソフトウェア開発環境の現状と今後について述べる。現開発環境では、UNIXベースのクロス開発環境とリアルタイムOS(RX616/RX-UX)用サポートシステムを提供しているが、今後は、このリアルタイムOS用の開発環境を、特にシステムデバッグを中心に強化していく予定である。特徴として、リアルタイムOSに特化したデバッグ機能を高度なユーザインタフェースで利用可能とし、デバッグ環境の構築にあたりOS/デバッガ/通信の各インタフェースを共通化することで開発環境の拡張性/柔軟性を考慮している点である。

32-BIT V SERIES SOFTWARE DEVELOPMENT ENVIRONMENT

Kazuya Hashimoto Masashi Doda
Basic System Department

C&C Common Software Development Laboratory
NEC Corporation

11-5, Shibaura 2-Chome, Minatoku, Tokyo 108, JAPAN

We have provided 32-bit microprocessors, V60/V70/V80 and its software development environment. This paper describes an outline of the software development environment and several features considering new facilities of these chips.

Currently we have provided the cross software tools for these chips and the software tools for the realtime operating system, RX616 and RX-UX. For future, we will enrich the software environment for the realtime operating system, especially for the debugging phase at hardware/software integration time. There are two key features of next software environment; It provides several powerful facilities enough to debug application tasks on the realtime operating system with pleasant user interface. It also provides a distinct definion of interfaces among target operating system, debugger and their communications in order to construct a flexible/extensible software environment.

1. はじめに

ここ数年、マイコン応用分野も機器の多機能化、高性能化が強く求められるようになってきており、それにとりも高性能かつ大規模システムに対応可能な32ビットマイクロプロセッサ(以下MPUという)市場も、急激な立ち上がりを見せている。

一方、その開発環境はプロセッサの進展に充分対応しているとは言えない状況である。これは、32ビットMPU自身の高機能/高性能に伴い、その開発ツール自体の開発が困難になってきていることと、そのアプリケーションソフトウェアも大規模化/複雑化してきて、これまでの支援ツール群だけでは開発支援環境としては不十分であることがあげられる。

本稿では、当社の32ビットマイクロプロセッサであるVシリーズの紹介を行ない、特に上述のような状況に対して、当社がこれまで対応してきたソフトウェア開発環境について述べ、さらに今後の開発環境の方向について述べる。

2. 32ビットVシリーズMPUと周辺LSI

図-1に示すようにVシリーズMPUは、大きく16ビットMPU系と32ビットMPU系とに分かれている。

VシリーズMPUの製品展開

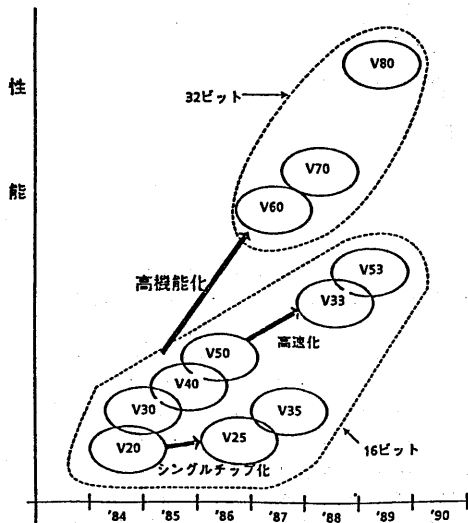


図-1 VシリーズMPU

ここでは、32ビットVシリーズMPUの μ PD70616(V60)/ μ PD70632(V70)/ μ PD70832(V80)の機能概要とそれらをサポートする周辺LSIについて述べる。32ビットVシリーズは、高級言語指向の命令体系、パイプライン方式、仮想記憶管理機構(MMU:Memory Management Unit)などの特徴を持つオリジナルMPUで

ある。V60/V70は、応用システムに必要な基本機能を1チップに集積することを目標としたのに対し、V80では、さらにメインフレームで採用されてきた高速化手法のオンチップキャッシュメモリや、分岐高速化技術等を取り入れることにより、V60/V70の基本機能を高速に処理することを目標としている。また、ソフトウェアの上位互換性を保証しており、言語処理系、デバッグ支援ツール、およびリアルタイムOS等はほぼそのまま共通に使用できる。

次表に、V60/V70/V80の諸元を示す。

	V60	V70	V80
仮想空間(実空間)	4GB(16MB)	4GB(4GB)	4GB(4GB)
レジスタ数	32	32	32
トランジスタ数	37.5万	38.5万	93万
パイプライン数	6段	6段	7段
キャッシュメモリ	なし	なし	内蔵
内部/外部バス	32/16ビット	32/32ビット	32/32ビット
最大性能	3.5MIPS	6.6MIPS	16.5MIPS
クロック周波数	16MHz	20MHz	33MHz

今後の方向としては、高速化、シングルチップ化、低価格化に向かって進んでいる。

MPUだけでなく、次のような周辺LSIも提供している。

- 浮動小数点演算コプロセッサ(μ PD72691)
- キャッシュメモリコントローラ(μ PD71641)
- クロックジェネレータ(μ PD71611, μ PD71621)
- システムバスコントローラ(μ PD71613)

上記のうち、IEEE754に準拠の μ PD72691は、三角関数などの初等関数演算命令に加えてベクトル演算や行列演算命令も有しており、ロボットやNC等のような、より精密な制御を必要とする分野では重要なチップである。

また、評価環境としては、PC-9800ベースSBCシステムやマルチバスI仕様等のSBCも提供し、ターゲットシステム構築/評価の便宜も図っている。

3. V60/V70/V80用ソフトウェア開発環境

チップの性能を最大限に引き出すようなアプリケーションソフトウェアを如何にして短期間で高品質に開発するかは、ハードウェア/ソフトウェアを含めた開発環境全体の問題である。

ここでは、当社が提供しているV60/V70/V80用ソフトウェア開発環境を説明する。本開発環境、MPUの機能を最大限に引き出すOSとUNIX*1)ベースのMPU開発支援ツールで構成されている。以下に開発環境の概要を述べる。

3.1 UNIXベース開発環境

広大な32ビットアドレス空間を利用できるMPU用の大規模ソフトウェア開発では、パソコンベースの環境は、そのCPUパワーやファイル容量の点から機能/性能不足である。最近の傾向であるワークステーション (WS)の低価格化も手伝って、今後はこのWS上のUNIXベース開発環境がいよいよ主流になると思われる。

我々は、このUNIX上にソフトウェア開発環境を整備している。それにより、UNIX上の基本ツールであるエディタを始めとしたソースファイル回りのユーティリティ等はそのまま利用することができる。

3.2 オペレーティングシステム

業界標準となっているUNIX SystemVとITRON仕様^{*)2)}に準拠したOSを提供している。

3.2.1 UNIX

PC-9800にV60アドオンボード上で動作するUNIX SystemV R2.2準拠のUNIX, PC-UX/V(V60)を提供している。主に小規模用のマイコンソフトウェア開発環境のひとつとして、また、UNIXベースのチップ評価環境として提供するために開発された。

3.2.2 リアルタイムOS

組込み用OSとしては、16ビットMPU以下では、独自でOSやモータまで開発することもあったが、32ビット用ともなると基本OS部分の複雑度も増しその開発にはかなりの工数が必要とされ、32ビット用基本OSとしては標準のOSを導入することが多くなると思われる。

VシリーズリアルタイムOSとしては、V20/V30用から、V60/V70/V80までITRON仕様に準拠のリアルタイムOS, RXシリーズで統一し、用途に応じて提供している。

V60/V70/V80用リアルタイムOS(RX616)は、32ビットMPU時代にふさわしいOSとして信頼性向上と、統合システムへの対応を目標にしている。高信頼性は、V60/V70/V80内蔵のプロテクションメカニズム、耐故障機能、及びランデブによるタスク間通信機能により実現している。統合システムには、3つの基本サブモジュールである、リアルタイムカーネル、UNIXスーパーバイザ、およびファイルサーバを統合したリアルタイムUNIX(RX-UX832)で対応し、リアルタイム処理にはITRON仕様に準拠した機能、データ処理にはUNIX SystemVに準拠した機能の同時使用を可能にしている。

また、最近では、応答性能だけでなく高スループット(CPUパワー)を要求するアプリケーションの出現や、宇宙ステーションのように耐故障性の強い要求に応じて、動的負荷分散機能、プロセッサ縮退運転機能を持ち、RX616に対してシステムコールレベルで上位互換の密結

合型/疎結合型マルチプロセッサ用リアルタイムOSを開発中である。

3.3 開発支援ツール

次に、ユーザがMPU組み込みソフトを開発する際に効率よくシステムを構築するための支援ツールについて述べる。開発支援ツールとしては、リアルタイムOS(RX616)に特化したツール群とリアルタイムOSに依存しない汎用のツール群とに分けられ、それぞれ、言語処理系とデバッガがある。

3.3.1 汎用サポートシステム

V60/V70/V80固有のクロス言語処理系を、C言語のコンパイラパッケージとして提供しており、次に示すものを用意している。

- (1) Cコンパイラ(cc70616)
- (2) アセンブラ(as70616)
- (3) リンカ(ld70616)
- (4) アーカイバ(ar70616)
- (5) コンフィギュレーター(cf70616)
- (6) ヘキサ変換(hx70616)
- (7) ネームリスト(nm70616)
- (8) サイズコマンド(size70616)
- (9) オブジェクトストリップ(strip70616)
- (10) オブジェクトダンプ(dump70616)

これらツールは、現在次に示すようなホスト環境で動作している。

マシン	OS
VAX-11ファミリ	UNIX4. [23]BSD, VMS
EWS-4800	EWS-UX/V(SystemVR2.2相当)
PC-9800シリーズ	PC-UX/V(V60)(同上)

これらツールの利用の流れを図-2に示す。ツールは一部のものを除いて、UNIX SystemV R2.2(VAX)をベースとしているが、オリジナルにはない幾つかの特徴を持っているので簡単に説明する。

● Cコンパイラ

コンパイラの開発にあたっては、コード生成部を自動生成するツール^[8]によって作成した。これにより、コンパイラ開発時の生産性、コンパイラ自身の保守性に効果があった。また、チップのアーキテクチャを活かした次の機能を有している。

1. バイブラインを意識した最適化
2. ユーザ指定可能なレジスタ割り当て
3. MPUの持つ豊富なアドレッシングモードから

*1) UNIX は米国AT&Tベル研究所で開発されたオペレーティングシステムの名前です。

*2) ITRON は、"Industrial - The Real Time Operating System Nucleus" の略称である

の最適選択

● アセンブラ

オリジナルへの拡張として次のことを行なっている。

1. アセンブルリスト出力
2. ユーザ定義マルチセクション
3. 組み込みマクロ機能

他のリンク以下のツール、オブジェクトファイル関連のユーティリティに関しては、UNIX SystemVのものほとんど同じである。オリジナルツールとして、以下の2つがある。

● ヘキサ変換

インテル/テクトロ/モトローラの各ヘキサ形式をサポートしている。

● コンフィギュレータ

仮想モードで動作するプログラム開発のために、特にその仮想アドレスと実アドレスのマッピングテーブル生成支援を行なうツールである。

リンクを通して作成されたロードモジュールは通常仮想アドレスモードのプログラムとなっている。これをどのように実メモリ上にロードするかを記述したマッピング情報ファイルとともに本ツールに入力することで、ロード可能なファイルとマッピングテーブルを出力する。

デバッグ環境であるが、インサーキットエミュレータ(以下エミュレータ)を中心に整備されている。

エミュレータは、2つのタイプを用意している。ひとつは、16ビットから32ビットMPUまで統一した操作環境を提供している高機能型であり、もうひとつは、パソコンと一体となったもので機能的には限定されるが、その

分安価となっている廉価版である。何れも高級言語デバッガ(CC70616-D)とのインタフェース(I/F)を持ち、C言語レベルのデバッグを可能としている。CC70616-Dは、以下の特徴を持っている。

- (1) チップの機能である(多重)仮想空間を利用したソフトウェアのデバッグをサポートしている
- (2) UNIXシェル(sh, csh, 等)をユーザI/Fに利用し、デバッグ履歴の保存やデバッグコマンドの再実行、実行結果をファイルへ保存/再利用する各機能を提供し効率のよいデバッグを可能にしている
- (3) エミュレータ、ターゲットに組み込まれたデバッグモニタ、およびホスト上のシミュレータ等種々のデバッグ環境とのI/Fを持ち、デバッグ段階に応じた構成が可能である

3.3.2 リアルタイムOSサポートシステム

リアルタイムOS用のサポートシステムとして、リアルタイムC、SGサポートツール、タスクリンクというツールを用意しており、これらは、汎用サポートシステムの言語処理系と連携し、OS組み込みソフトウェアを構築する。

リアルタイムCは、RX616の特徴のひとつである同期型タスク間通信機能を実現する機構であるランデブの記述を可能にするプリプロセッサである。

SGサポートツールは、ユーザのシステム情報(タスク仕様の定義、メモリバッファフル定義、各種初期値等)を定義したライブラリを作成するためのシステムジェネレータ(apsg)および、SG定義を容易に記述することが可能なC言語風のSG言語を提供している。

タスクリンク(vslld)は、システムジェネレータで作成

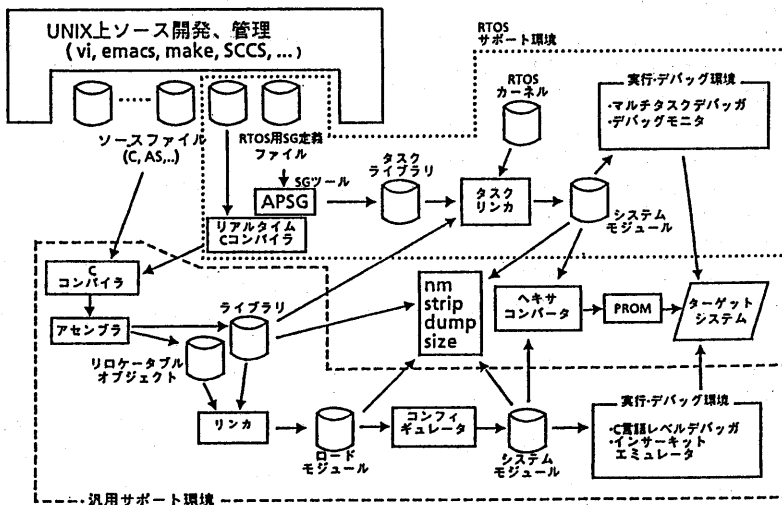


図-2 サポートシステム

したタスクライブラリのシステム情報により、カーネルとユーザ開発のアプリケーションをリンクし、ロードモジュールを作り上げる。

これらは、汎用サポートシステムと同様のホスト環境で稼働している。

デバッグ環境としては、カーネル組み込み型のデバッグモニタを提供しており、アプリケーション開発段階や、製品出荷後の障害を吸収できる。これは、メモリアクセスやブレークポイント関連機能の他にOSの資源表示機能を有している。また、OS上のタスクをシステム全体を停止させずにC言語レベルでデバッグ可能にするマルチタスクデバッグが試作完了している。

4. 今後の開発環境への展開

32ビットVシリーズの開発環境を今後どのように展開するかについて留意すべき問題点とともに述べる。

4.1 チップの新機能へのサポート

チップ内蔵のMMUとチップ自身の高速化への対応が、32ビットMPUの新機能に対するソフトウェア開発サポートにとっては大きな問題となる。

4.1.1 MMUと大規模ソフトウェアへの対応

32ビットMPUでは、マルチタスクシステム構築をサポートする機能(多重仮想空間)や、データの破壊を防ぐためのプロテクションメカニズムをサポートしたMMUを内蔵するものが増えてきている。従って、開発ツールでもこのMMUへの対応を考慮しなければならない。当社の提供する開発環境では、MMU対応として、コンフィギュレータやSGツールを提供し、デバッガでは、複数の仮想空間を名前で認識できる機能を実現した。

今後は、プログラムの大規模化に対応して以下の点についても充実を計る。

- ダウンロードのためのホストマシンとターゲットマシン間の高速リンク
- 複数人による分散開発への対応(ネットワークサポート)
- ソフトウェア複雑化への対応(上流工程支援)

4.1.2 チップ高速化への対応

チップの高速化に伴い、エミュレータ等のデバッグハードウェアの開発が困難になってきているため、エミュレータに代わるデバッグ手段が必要である。チップシミュレーションキータラゲットに組み込んで使われるデバッグモニタといったものが16ビットMPU以下の環境と同様に必要となる。

4.2 統合デバッガ

先にも述べたように32ビットMPU時代に入り、ますます市販のリアルタイムOSを購入する傾向が強くなると

考えられ、そのリアルタイムOS用開発環境の整備は急務となってきている。我々は、開発工程の中でも、特にシステムデバッグのフェーズが重要と考えおり、デバッガ及びその回りの支援環境を含めたデバッグ環境の整備を進めている。

従来より提供されてきたエミュレータやデバッグモニタ等の汎用のデバッグ環境は、チップI/Fに基づくツールが主体であったため、OSにより動的に生成されたタスク(プロセス)をデバッグするには、困難があった。今後は、このOS上のアプリケーションタスクを効率よくデバッグする環境が重要となる。

このため、前述したマルチタスクデバッガを強化し、C言語レベルの基本デバッグ機能と、リアルタイムOS(RX616等)に特化したタスクデバッグ機能とを包含した統合的なデバッガ(仮称VDB)を開発中である。VDBでは、この2つのデバッグ機能の融合を目指しているが、実現にあたっては、ホストマシンとターゲットマシン間の接続回線のインタフェースを規定し、種々の通信媒体(Etherや、RS-232C等)に対応する等、機能間のインタフェースを明確し、開発規模や用途に応じて柔軟な構成ができるよう配慮している(図-3参照)。さらに、VDBでは、デバッグ機能の提供だけでなくユーザインタフェースの改善、ネットワーク対応等デバッグフェーズでのトータルサポートを目的としている。実現予定の各機能の特徴を説明する。

(1) リアルタイムOS依存機能

アプリケーションタスクのデバッグ機能と、OS資源/システムコールに関わる以下のシステムデバッグ機能を提供する。

- 動的タスクのデバッグ機能
- タスク状態の表示機能
- タスク間通信トレース、システムコール発行履歴等のトレース機能
- イベントフラグ、セマフォ、メールボックス、メモリアル、各キュー等のOS資源表示機能
- システムコール発行機能
- デバッグタスク指定機能
1タスク1ウィンドでデバッグが可能
- システムレベルデバッグ機能
システムを止めることなくタスクのデバッグが可能モードとブレーク時にシステムを停止させて、システム全体のデバッグ可能とするモードを提供する。

(2) 基本機能

Cソースレベルの基本デバッグ機能とアセンブリ言語レベルの基本デバッグ機能を提供する。また、浮動小数点演算コプロセッサμPD72691にも対応し、逆アセンブル機能コプロセッサ内部のレジスタ表示/設定機能を提供する。

(3) 共通インタフェース

ターゲットデバッグ環境とホストデバッグ環境間

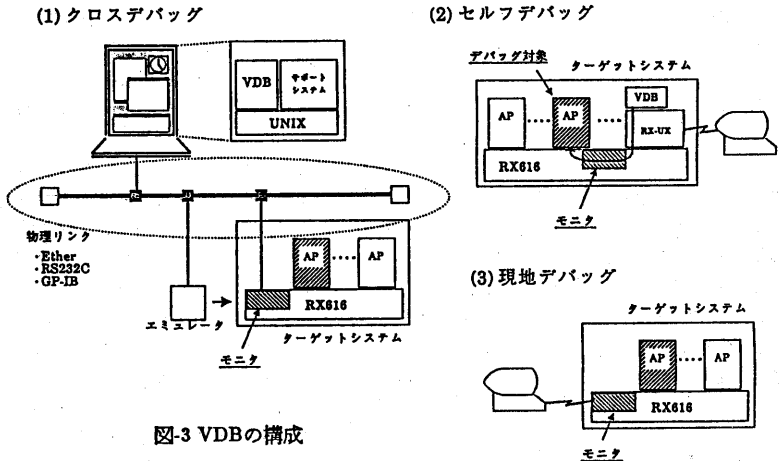


図-3 VDBの構成

のインタフェースを定義することにより、機種独立性を高め、デバッグモニタやエミュレータ等の種々の構成を可能としたり、また、この共通インタフェースに、OSとのインタフェースも規定し、RX616以外のリアルタイムOSにも適用可能とする。この共通インタフェースに対応したエミュレータが使用できると、モニタ版と同様の操作でエミュレータをコントロールすることが可能であり、エミュレータのデバッグ機能（リアルタイムトレースやROMデバッグ機能等）の使用で、デバッグの巾を広げることが可能である。

- (4) HMI(ユーザインタフェース)
HMIはデバッグ効率に大きく影響する。VDBでは、HMIとしてビットマップ/キャラクタ両ディスプレイ用のI/Fを提供する。
- (a) マルチウィンド版は、ユーザがデバッグ対象として指定したタスクに対してウィンドがオープンされる。そのウィンド中では、さらに幾つかのサブウィンドがオープンされ、メニューとキーボードによりコマンドを入力できる。また、OS資源のデータ構造をビジュアルに表示する機能も有している。
- (b) キャラクタディスプレイ版ではUNIXのシェルインタフェースを提供しており、シェルのalias, history, redirection, script等の機能をそのまま使用できる。
- (5) 今後の課題
さらに、以下の項目についても今後検討する。
- マルチプロセッサ用デバッグ機能
 - リアルタイムシステムの評価環境/ツール
 - 開発工程間の有機的な結合

5. おわりに

以上、当社が提供している32ビットMPUとその開発環境の概要を説明し、今後の展開の方向を示した。3節で述べたソフトウェア開発環境は、製品として出荷されていないものも一部含まれているが、社内では開発完了し、評価中である。4節の今後の展開で述べた機能の一部は、現在試作/評価中である。本稿では、Vシリーズについてのみ記述となっているが、述べてきた方向や問題点については、MPUには独立な議論である。今後は、本稿で述べた方向で進め、より良いマイコンソフト開発環境を構築していく予定である。

6. 参考文献

- [1]赤塚ほか：『Vシリーズの展開と開発支援ツールの詳細』、COMPUTER DESIGN 1月号(1989)
- [2]『Vシリーズのソフトウェア製品群』、NEC デバイステクノロジ、No.22(1988)
- [3]岩崎ほか：『リアルタイムUNIX RX-Ux832』、NEC 技報、No.41(1988)
- [4]平澤ほか：『Vシリーズマイクロプロセッサの開発環境』、NEC 技報、No.41(1988)
- [5]河本ほか：『キャッシュメモリと分岐予測機構を内蔵した32ビットマイクロプロセッサV80のアーキテクチャと応用』、COMPUTER DESIGN 5月号(1989)
- [6]中本ほか：『V80リアルタイムOSにおけるマルチタスクデバッグの実現』、情報学会ソフトウェア工学56-6(1987)
- [7]橋本ほか：『V60高級言語デバッグの実現法』、第34回情報全国大会論文集3S-4(1987)
- [8]三橋ほか：『コード生成部生成ツールCOO』、情報処理学会論文集、第28巻(1987)
- [9]藤林：『マイクロコンピュータソフトウェア開発環境』、標準化と品質管理、Vol.41(1988)
- [10]『32ビットマイクロプロセッサ-応用-開発-評価』；日経データプロ、(1988)
- [11]坂村健編：『ITRON入門』、岩波書店、(1988)
- [12]『RX616 ユーザーズ・マニュアル』、NEC、(1988)
- [13]中本ほか：『V60リアルタイムOSにおけるマルチタスクデバッグ』、第33回情報全国大会論文集3V-4(1986)