

システム LSI ソフトウェアの開発環境と 実行環境

Development and Run-time Environments for System LSI Software

あらまし

システム LSI 用のソフトウェアは、開発量が多く、プログラム構造も複雑になってきており、また種類も多岐に渡る。このような状況のなかで、プログラム開発者は限られた期間で確実にソフトウェア開発を行わなければならない。しかし、従来のコンパイラ、デバッガなどの基本ツールだけではこれらの状況に対応できなくなってきた。これに応えるため、富士通では統合開発環境“SOFTUNE”を開発した。SOFTUNE は、ソースファイルをプロジェクトという単位にまとめて管理し、エディタ、コンパイラ、デバッガなどのツールを連携させることで開発効率を上げることができる。また、ビジネスパートナー各社との協調により、言語ツール、デバッガ環境、ICE や、リアルタイム OS、CO デザイン ツールに至るまでの製品を幅広く取り揃えたことで、顧客が目的に応じた開発ツールを選択できるようにした。

一方、近年はインターネットに代表されるように、ネットワーク、マルチメディア、リアルタイム処理をキーワードとするソフトウェアの開発が多くなってきている。このため、出来上がったソフトウェアを実際のシステムに近い環境で動作させ検証する「実行環境」が必要となっているため、リアルタイム OS を搭載し、ネットワーク、マルチメディアに対応したリファレンスボードとマルチメディアライブラリを提供した。

Abstract

Software for system LSI is becoming increasingly diversified and complex. Because of this, program developers are finding it difficult to meet deadlines with just a compiler and debugger. Therefore, Fujitsu has developed an integrated development environment called SOFTUNE. SOFTUNE raises development efficiency by linking tools such as the editor, compiler, and debugger. Fujitsu also provides various other products, for example, language tools, debugger environments, ICEs, realtime operating systems, and CO-design tools, which customers can select according to their development needs.

To help developers prepare software for today's environment, Fujitsu has developed reference boards and multimedia libraries with realtime operating systems for run-time verification of network and multimedia software.

This article introduces the development of such system LSI software environments.



五十嵐 純 (いがし じゅん)

1979年東海大学工学部光学工学科卒。
1983年富士通入社。当初はパソコン
の企画に従事。以後マイコン開発支
援システムの企画、開発に従事。
システム LSI ソフトウェア部基盤ソ
フトウェア開発部



根岸康之 (ねぎし やすゆき)

1978年東京工業大学総合理工学研究
科生命化学専攻修士課程了。1983年
富士通入社。マイコン開発支援シ
ステムの企画、開発、サポートに従事。
システム LSI ソフトウェア部基盤ソ
フトウェア開発部

ま え が き

半導体の高集積度化，高性能化に伴い，ハードウェアのソフトウェア化が可能になってきた。とくに32ビット・マイクロプロセッサは，従来の制御中心の用途から情報通信処理，さらにマルチメディアデータ処理へと適用が広がっている。ハードウェア技術のソフトウェア化はマルチメディア処理，とくに携帯機器のネットワーク対応でますます不可欠の要素となってきている。これらの分野には，マイクロプロセッサ・コアに各種周辺マクロを取り込んだシステムLSIが使用される。

このシステムLSIのソフトウェアを開発する環境として，コンパイラ，アセンブラ，およびICEを主な開発ツールとしていた従来型の開発環境に加え，これらの機能を統合して使い易くした統合開発環境や，さらにその上流の設計を支援するCASEツール，またハードウェアとソフトウェアのCOデザインツールなどが充実した開発環境が要望されている。さらに開発環境だけでなく，実際のソフトウェアが実行される環境とほぼ同じハードウェア機能を備えたりファレンスボードの上に，OSやドライバ，マルチメディアライブラリなどのソフトウェアを搭載した実行環境を用意することもシステムLSIのソフトウェアの評価・デバッグには重要である。このようにシステムLSIのソフトウェアの開発環境としては，開発環境と実行環境のそれぞれに幅広く各種開発ツールを提供する必要がある(図-1)。

本稿では，システムLSIのソフトウェアの開発環境に対する従来の問題点と，それに対する富士通のソリューションについて紹介する。⁽¹⁾

従来の開発環境の問題点

富士通では，従来ワークステーションやパソコン上に

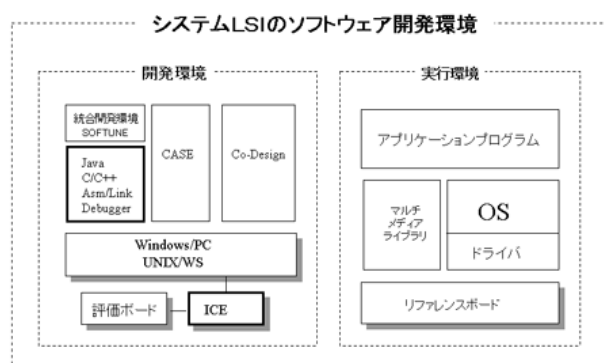


図-1 開発環境と実行環境

Fig.1-Cross and run-time environments for system LSIs.

コンパイラ，アセンブラ，リンカ，デバッガなど一連のクロス開発環境を提供してきた。しかし，これらをプログラム開発者が使用する際には，使用するマイクロプロセッサに最適なコードを生成するためにコンパイラやリンカに対して複雑なオプションを指定する必要がある。また，プログラム開発者は「コーディング」，「ビルド」，「デバッグ」という作業工程を何度も行き来することになる。しかし，従来の開発環境ではこれらの作業はコンソールのコマンドラインからの指示により一つひとつ独立した作業として行わねばならず効率が悪かった。

組込みシステムを開発するシステムベンダは，市場にタイムリに製品を投入しなければならない。そのためプログラム開発者は効率よくソフトウェアを開発しなければならないという使命がある。また，近年ではマイクロプロセッサの性能が向上した分，プログラム開発者が作成するプログラムの規模も大きく複雑になる傾向があり，多人数で開発する状況となってきている。このような状況においては開発環境はネットワークシステムに対応していること，また増え続けるソースプログラム資産を確実に管理できること，そしてこれらソフトウェア開発環境自体が高性能でかつ安価で取り揃えられること，などが要望されている。

富士通の統合開発環境 “SOFTUNE”

これらの要望へのソリューションとして，富士通では統合開発環境 “SOFTUNE” ^(注)(ソフチューンと読む)というワークベンチを開発し提供している。SOFTUNEはWindows上で動作し，コーディング，コンパイル，リンク，デバッグの繰返しを簡単な操作でスムーズに行えるよう「プロジェクト」というダイアログにソースファイルやオプションなどの情報をまとめた。また，従来はコンパイラ，リンカなどのツールをプログラム開発者が直に起動していたが，これを「ビルド」という操作にまとめた。これにより，プログラム開発者はコンパイラ，リンクの操作に余計な注意を払わなくて済むようになり，その分，コーディングとデバッグの作業に集中できるようになった。また，これらの機能を実現するためにSOFTUNEの開発においては，専用のテキストエディタとデバッガを開発した(図-2)。

以下，SOFTUNEに統合されている主な機能について述べる。

(注) 富士通の登録商標。今後，富士通製のマイクロプロセッサの開発ツールの総称名として使用する。

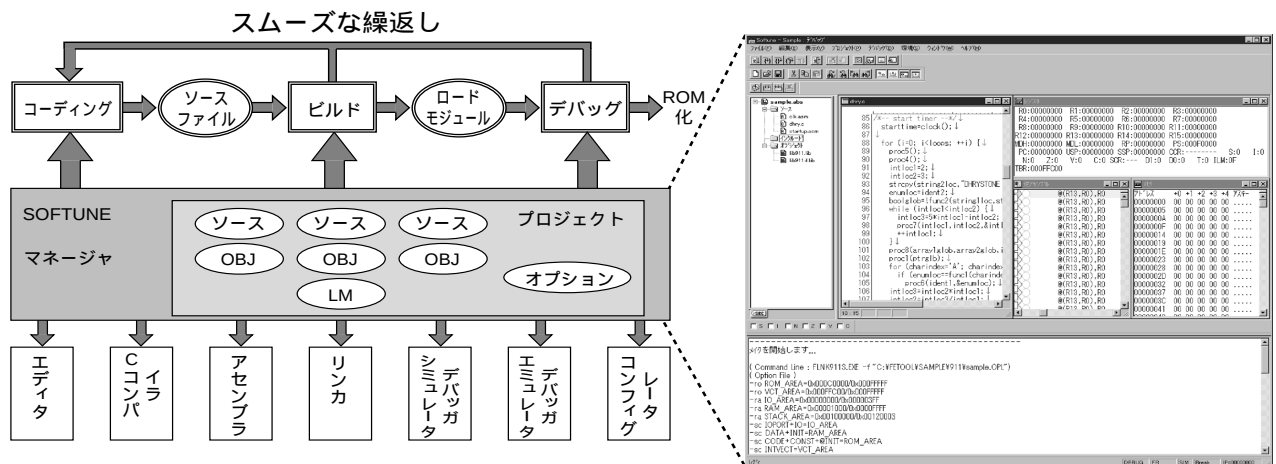


図-2 SOFTUNEのコンセプトと外観イメージ
Fig.2-Concept and external image of SOFTUNE.

● 各ツールを統合するSOFTUNEマネージャ

SOFTUNEマネージャは、ソースファイル名やツールへのオプションの情報をプロジェクトという単位にまとめて管理し、SOFTUNEワークベンチの中枢に位置する。SOFTUNEマネージャは、プログラム開発者の生産性と安全性を高めることを第一の目的としている。この考えに基づき次の点に考慮して設計している。

- (1) シンプルで統一されたダイアログ設計
- (2) ラップトップの画面サイズに納まるダイアログ設計
- (3) 従来の言語ツールはそのまま使い互換を維持
- (4) 従来ツールにオプション設定ダイアログを新規追加
- (5) ビルド、エラージャンプなどによりツール連携
- (6) エディタ内蔵
- (7) カスタマイズ機能
 - ・市販のエディタが組み込める
 - ・市販のソース世代管理ツールなどを組み込める
 - ・日本語/英語、両方の表示を持ち、切り換え可能
- (8) 言語マニュアルをオンラインヘルプ化して内蔵
- (9) 複数人でネットワークを使う環境を考慮

SOFTUNEマネージャの開発に当たっては、内蔵エディタの機能を簡易化するなどの仕様の割り切りを行なったものを早期に開発し市場投入した。これにより現在のバージョンV03への機能強化に当たっては、市場のプログラム開発者からの数々の要求を盛り込んで開発した。

● コンパイラ最適化技術

組込みシステムを開発する場合、コストを考慮してROMおよびRAMサイズを極力小さく設計することが要求される。このため、マイクロプロセッサ用コンパイラは、いかに効率のよい小さなコードを出力するかが重要なポイントとなっている。SOFTUNEのコンパイラ開発者

は、コンパイラが最適化しやすい命令セットとなるよう、マイクロプロセッサのアーキテクチャの設計にも関与している。また、コンパイラ本体に関してはメインフレームやワークステーションを開発している富士通の情報処理部門で長年培われた技術を導入している。これらの効果が相まって、コンパイラの最適化技術に関しては他社マイクロプロセッサ向けの各社コンパイラに比較して優秀な値を実現している(図-3)。

● GUIを備えた新しい言語支援ツール：Cチェッカ・Cアナライザ

コンパイラはソースプログラム内の文法的な構文エラー箇所に対して診断メッセージを出力する。しかし、この診断メッセージはCの言語仕様をよく理解していないと分かりにくいものであり、またコンパイル単位である一つのソースプログラムだけで検出できる情報しか出力できない。

SOFTUNE Cチェッカは、プロジェクトに登録されたCソースファイルに対して静的に詳細な検査を行い、コンパイラでは検出されないコーディングのミスを検出し、WindowsのGUIを利用して修正案を表示する。プログラマの使用目的に合わせたオプションが設定でき、例えば、富士通製マイクロプロセッサへ移植する場合の注意すべきコーディング箇所も指摘できるようになっている。Cチェッカは内部的には前述のSOFTUNE Cコンパイラをエンジンとして使用している。

また、SOFTUNE Cアナライザは、主にプログラム開発の終盤のチューニングや保守のフェーズで使用するツールであり、関数間の構造や参照情報、統計情報をツリー図型に視覚的に表示し、それを印刷することができる。また、組込みシステムのプログラマはRAMメモリの搭載

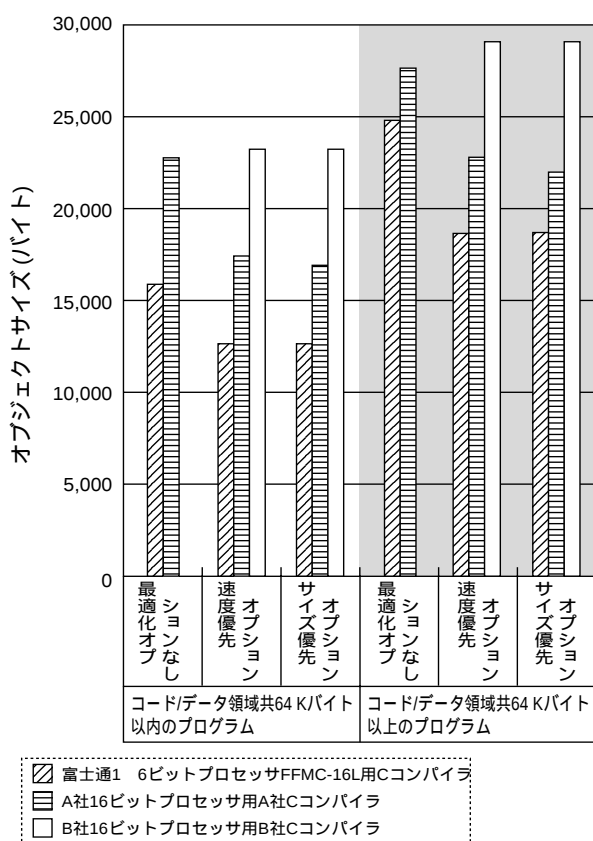


図-3 Cコンパイラの性能比較結果
Fig.3-Performance comparison result of C compiler.

量の関係で各関数が使用するスタック量が気になるが、Cアナライザは、各ソースプログラム毎のコンパイル時に検出したスタック使用量を情報ファイルとして出力することにより、それらの情報の集計をとり、その結果を各関数ツリー図内に表示させることができる。このためプログラムは、関数の構造やスタックの使用量を視覚的に把握できるので、プログラムの性能および品質について再チェックし保守することが可能である。

- リアルタイムOSを構築：REALOSコンフィグレータ
富士通のITRON仕様のリアルタイムOSはREALOSという商品名である。REALOSはOSの各機能をモジュールとして提供しているため、プログラム開発者は必要とするモジュールだけを組み込めばよい。しかし、実際にリアルタイムOSを構築するためには、必要なモジュールに対しタスクの数やシステムコールの種類など、多岐に渡るパラメータを設定してリンクしなければならない。富士通では、このパラメータ設定を補助するツールとして従来よりREALOSコンフィグレータを提供していたが、SOFTUNE V03では、SOFTUNEマネージャのなかにウィザード機能として統合した。このウィザード機能を使うことによりプログラム開発者は、つぎつぎと表示される質問ダイアログに従ってパラメータを設定していくだけ

で目的の機能を持つOSを的確に構築することができる。

- リアルタイムでデバッグする道具：ICE

プログラム開発者の実機ボードでのプログラムをリアルタイムで検証するためには、ICE(In Circuit Emulator)が必要である。ICEは実機ボード上のマイクロプロセッサ・ソケットにマイクロプロセッサの代わりにボッドと呼ぶコネクタを差込み、マイクロプロセッサの動作をリアルタイムでエミュレーションするものである。このため、プログラム動作時のレジスタやメモリ内容、実行命令のトレースなどをリアルタイムでデバッグすることができる。

- 3種類のデバッガを統合：SOFTUNEデバッガ

デバッガには、ICEと接続してデバッグを行うエミュレータデバッガ、ICEを使わずにソフトウェアでシミュレーションするシミュレータデバッガ、実機ボード上のROMにモニタプログラムを置きホストと通信しながらデバッグするモニタデバッガの3種類がある。従来はこれらのデバッガを個々に製品化していたが、SOFTUNEデバッガでは操作画面の共通化、メンテナンス性などの観点からこれらを一気に統合した。また、この3種類のどのデバッガでもREALOSの各タスク単位でのデバッグができる。

SOFTUNE V03では、SOFTUNEデバッガは前述のSOFTUNEマネージャとプログラムの内部構造的にも一体化した仕様で開発を行った。このことにより、例えばプログラム開発者がデバッグ中にソースプログラム上にバグを発見した際に、即座にSOFTUNEマネージャのテキストエディタを呼び出すことができるようになった。これにより、プログラム開発者は、デバッグ中に、すぐにソース修正、ビルドを行い、出来上がったロードモジュールを再ロードし再びデバッガモードに戻る、ということが簡単な操作で確実に行えるため、開発効率の向上に貢献している。

- 評価ボード

マイクロプロセッサの性能評価を行うためのボードとして評価ボードを用意している。このボードには、富士通のマイクロプロセッサ、およびROMやRAMメモリや基本的な周辺I/Oデバイスなどを搭載しているため、この上でプログラム開発者が作成した評価用のプログラムを動作させてマイクロプロセッサの性能やプログラムの検証などが行える。

後述の実行環境に関する章で紹介する「リファレンスボード」は、より顧客の最終製品の実機ボードに近い構成のボードであり、ここで述べた「評価ボード」とは区別して呼んでいる。

各社ビジネスパートナーとの協調

マイクロプロセッサのソフトウェア開発環境としては富士通製だけでなく、プログラム開発者に広く受け入れられているツールベンダ(ビジネスパートナー)の開発環境も豊富に用意している。ツールベンダの製品を揃えておくことは、いままで他社のマイクロプロセッサを使用していたプログラム開発者が開発ツールの操作性は変えずに富士通のマイクロプロセッサに切り換えたい、という要求に対して有効な役目を果たす。以下にビジネスパートナーから提供されている主なツールについて述べる。

- 言語ツール・デバッグ環境

コンパイラの最適化性能に特徴をもつ言語系ツールおよびデバッガとして、Green Hills Software社の“Multi”を提供している。また、マイコン開発ツールとして国内で広く採用されているガイオテクノロジー社の“XASS-V”も提供している。さらに、フリーのソフトウェアとして“GNU”も提供している。サポートについてはCygnus Solutions社が行っている。

- ICE

他社のマイクロプロセッサも含めて利用者の多いICEとして、横河デジタルコンピュータ社の“ADVICE”を提供している。

- リアルタイムOS

32ビットクラスの組み込みシステムではリアルタイムOSは必要不可欠となる。国際的なリアルタイムOSとして次のものを取り揃えている。

- (1) Wind River Systems社 “VxWorks”
- (2) Integrated Systems社 “pSOS”
- (3) Mentor Graphics社 “VRTX”

そのほか、OS-9/DAVIDや、CHORUSなどもサポートしていく予定である。

- COデザインツール

ASIC設計を行う顧客が、実チップを試作する前にファームウェアの動作試験やパフォーマンスの評価試験などでハード・ソフトの同時設計、同時検証を行うためのツールとして、COデザインツールを提供している。横河電機の「仮想ICE」は、Verilog-HDL CPUモデルや仮想ロジックアナライザ機能、キャッシュモニタ機能を備え、主にハードウェア設計者が使用するツールである。

これに対し、ガイオテクノロジー社の“Asim-G”は、主にソフトウェア設計者が実チップ試作前に使用することを考慮してシミュレーション時間の高速化を念頭に開発されたツールである。

ソフトウェアの実行環境への要求

近年のインターネットに代表されるネットワークにおけるデータは、テキストのほかに画像、音声など様々で、これらはデジタル化され圧縮されたものである。またネットワーク環境では、プラットフォームに依存しないミドルウェアやコンテンツが要求され、実行環境は通信とJava、MPEGなどのデータの処理を中心としたものとなる。ネットワークでは1対1の通信のほか、放送に近い不特定多数との通信もあり、コンテンツ処理の時間的制約がある。リアルタイムOSは時間的制約のある処理には必須である。

したがって、実行環境はネットワークとマルチメディアのリアルタイム処理がポイントである。このため、リアルタイムOSを搭載し、ネットワーク、マルチメディアに対応したリファレンスボードとマルチメディアライブラリを提供した。

- ネットワーク環境でのOS

組み込み用のOSとして、ITRON仕様のREALOSを8ビット、16ビット、32ビットのマイクロコントローラ用に提供してきたが、近年リアルタイムOSのネットワーク対応は組み込み機器においても必要となってきた。

ネットワーク環境では特定のハードウェアとソフトウェアを前提にすると、つぎつぎ開発される技術に対応が困難となる。Java言語およびその関連技術はこの問題を解決するキーである。JavaVM(Virtual Machine)の命令コードであるJavaバイトコードは中間コードであり、特定のハードウェアに依存せず、ネットワーク経由で配布可能である。

JavaOSはJavaバイトコード実行に特化しており、そのサイズは現在普及しているパソコンOSのサイズの約1/10であり、ネットワーク機能、ウインドウシステムを持つ。JavaOS自身をネットワーク経由でダウンロードし起動することが可能で、維持管理がし易い。JavaOSは今後の高速なネットワークの普及に伴い、HMI(Human Machine Interface)を必要とする機器における標準OSとしての普及が期待される(図-4)。

- リファレンスボード

ネットワークコンピュータ用のJavaOSのようにその用途がほぼ決まっているものがある。その場合には要求されるすべての機能を搭載したボードが必要になる。JavaOS用ボードに搭載される機能はパソコンにおける搭載機能と同等である。

マルチメディアライブラリ用では、各マルチメディア

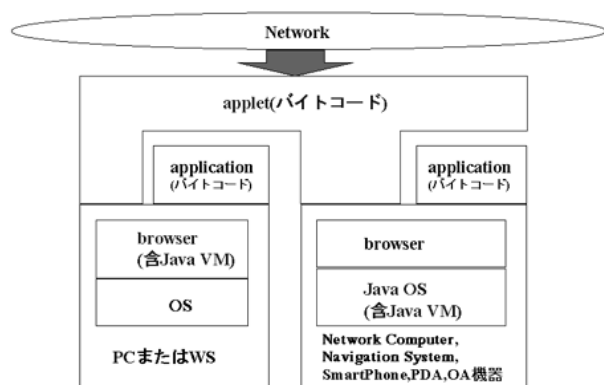


図-4 Java実行環境
Fig.4-Java run-time environment.

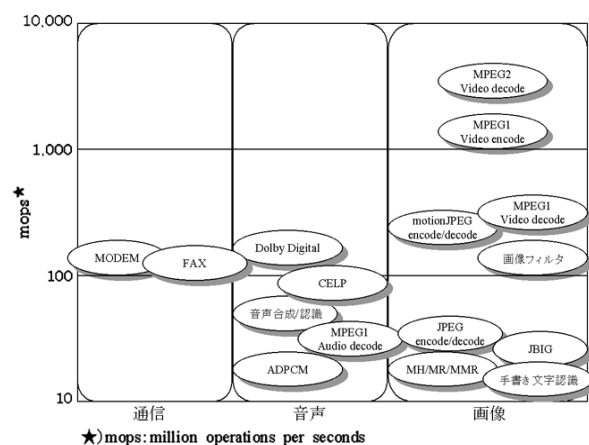


図-5 メディア処理と必要な性能
Fig.5-Performance for processing media data.

表-1 リファレンスボード

		Java用 リファレンスボード	メディアライブラリ用 リファレンスボード	
市場		ネットワークコンピュータ	FAX	デジタルカメラ
搭載機能	graphics			
	audio			
	network			
	serial			
	parallel			
	keyboard			
	mouse			
	file系			

データを対象としており、画像や音声の各データに応じたシステム構成をサポートしたボードが用意される(表-1)。

● マルチメディアライブラリ

画像、音声のマルチメディアデータの種類の多数あり、マルチメディアデータ処理に要する性能を図-5に示す。それらの処理に要求される演算性能は様々である。

マルチメディアデータの処理は従来は専用チップに任せていたが、マイクロプロセッサの性能向上により専用チップに任せていた分野をマイクロプロセッサで処理するようになってきた。性能によってはマイクロプロセッサ単独で処理できるものもあるが、性能を要求するものはマルチメディアアシスト(MMA)という専用チップも使

用する。MMAとマイクロプロセッサの併用またはマイクロプロセッサ単独による処理では、ソフトウェアによる機能アップや性能向上が可能であるほか、スペースの限られた機器では物理的な要求にも応えることができる。

む す び

富士通はマイクロプロセッサを幅広く取り揃えており、それに対するソフトウェア開発環境として、使い易く高機能な富士通製統合開発環境“SOFTUNE”を始め、ビジネスパートナー各社からも種々の開発ツールを取り揃えた。また、ミドルウェアライブラリや、評価ボード、リファレンスボード、リアルタイムOSなどの実行環境も充実させている。

今後は、開発環境をC++、Javaに対応させ、マルチメディアライブラリや、リファレンスボード、ドライバソフトなど、実行環境をさらに充実させていく。

参考文献

- (1) - : 日経エレクトロニクス, 1997. 9. 22付録, 日経BP企画部編集. 富士通特別広報特集: マルチメディア機器の高機能・高性能化を支える「SPARClite」.