

V60用 アルタイムOSの検討

- 32ビットI-TRONに向けて -

門田 浩 *、古城 隆 **、寺本 雅則 **

(日本電気(株) * マイクロコンピュータ技術本部、** マイクロコンピュータソフトウェア開発本部)

0 はじめに

半導体技術の進歩により、CAD用ワークステーションやプロセス制御機器等の分野にも32ビットマイクロプロセッサが使用可能になってきた。このようなマイクロプロセッサ応用システム規模の増大により、搭載されるOSの重要性も増している。例えば16ビットマイクロプロセッサの時代には組込みシステム専用の機能を中心に設計されていたリアルタイムOSも、システム規模とプロセッサの処理能力の増大により、データ処理あるいはマンマシンインタフェースにも比重がかかってきている。さらには広義のソフトウェア生産性向上が求められている。本報告では、このたび日本電気が開発した32ビットマイクロプロセッサ V60 (製品名μPD70616)用のリアルタイムOSを開発するに際しての検討事項を中心に述べる。

1 16ビットマイクロプロセッサ用I-TRON [1]

実時間制御を主目的とする工業分野のソフトウェアが普及する鍵は第一に処理性能で、次に機能の豊富さと使い易さである。I-TRONでは、性能を重視するため極端な仮想マシン・アプローチは採らず、一部の機能はインプリメント対象のCPU毎に仕様を定める事を許している。I-TRONは将来多くの機能がシリコン化されるべき基本機能を提供するチップ核と、基本入出力やデバッグ機能を提供するチップ核周辺とからなる。更に、チップ核周辺は核機能のシステム・コールによる拡張を主とする外核と、デバッグなどのシステム・タスクからなるユーティリティタスクに分れる。I-TRONではチップ核の外部仕様を詳細に規定しているが、チップ核周辺は厳密には規定しておらず例えば入出力タスクの設計法をガイドするに止っている。その理由は入出力関係のドライバは余りにもデバイスに依存し過ぎるため標準化が困難だからである。

I-TRONは先ず16ビットマイクロプロセッサ用(シングルプロセッサ)に設計された。現在、I-TRONのインプリメント対象となっているCPUは68000、8086系等の16ビットCPUである。それぞれ、I-TRON/68K、I-TRON/86等と呼ぶ。日本電気の16ビットマイクロプロセッサV20/V30用にインプリメントされたRX116 [2]はI-TRON仕様で設計開発された最初の商用製品でI-TRON/86に分類される。

2 32ビットマイクロプロセッサ V60 [3] [4]

リアルタイムOSの検討をする前に、先ずV60のアーキテクチャについて概説する。V60は内部データ処理構造が32ビット、外部とのインタフェースが16ビットのいわゆる32/16構造を持つマイクロプロセッサである。最新のCMOSプロセス技術を駆使して37万5千素子が1チップに集積されている。その設計目標は、高性能でありかつソフトウェア指向のアーキテクチャを提供することである。

高性能の実現のためにV60は6段のパイプライン構造と16MHzの高速クロックを採用している。パイプライン構造が理想的に動作すると最大4命令が同時に処理される。この時最大3.5MIPSの処理能力が期待できる。また、仮想記憶管理機構や浮動小数点演算機能などの基本的な機能は出来るだけオンチップに収めチップ外へのアクセスによる性能低下を回避している。

2.1 アーキテクチャ概要

V60アーキテクチャはソフトウェアの生産性や信頼性向上のために考慮が払われている。設計初期の発想としては、いかにしてコンパイラを作りやすくし、かつ生成されたオブジェクトコードの性能を上げるかにあった。これを追及した結果、ソフトウェア指向のアーキテクチャとなった。

(1) レジスタ構成

図1にV60の持つレジスタを示す。プログラマビブルなレジスタと特権レジスタに大別される。プログラマビブルなレジスタとして、32本の32ビット汎用レジスタとプログラムカウンタおよびPSWが提供される。この汎用レジスタはおのおのの浮動小数点データを含む全てのデータを扱うことができ、かつスタックあるいはアキュムレータとして使用できる。汎用レジスタアーキテクチャの主な利点を以下に挙げる。

- a) 専用レジスタ方式のプログラムにおいてプログラムのコンテキストが変わる毎に該当する専用レジスタを退避/復帰するオーバーヘッドを除くことが出来る。
- b) プログラマによるレジスタ管理が不要となる。
- c) コンパイラ設計者にとってはコード生成時のレジスタ管理が不要となる。

さらに実行時に於いては、多数のレジスタをデータキャッシュとして用い、メモリアクセスを減らすことにより性能向上を図れる。

(2) アドレス空間

大規模なソフトウェアを制約なく記述できるように、V60は仮想アドレスモードで4GB、実アドレスモードで16MBのリニアな空間を提供する。仮想空間はセクション(1GB)、エリア(1MB)およびページ(4KB)の3階層で構成される。エリアとページのマッピングのために2種類のマッピングテーブルを使用するが、この変換オーバーヘッドを除くために16エントリのTLBを内蔵している。図2にマッピングメカニズムを示す。

(3) データタイプ

通常の8/16/32/64ビットデータ以外に、32ビット及び64ビットのIEEE754準拠の浮動小数点データ、31桁までの10進データ、8/16ビットのストリングデータが扱える。後者は関連する命令とともに応用指向の機能を提供する。16ビットのストリングデータは漢字などの非英語系の文字列を扱うことを想定している。

(4) アドレッシングモード

21種類のアドレッシングモードが提供される。特に高級言語指向のモードとしてスケールドインデックシング、二重間接等がある。

(5) 命令形式

V60の命令形式は2オペランド形式であるが、各々のオペランドは基本的には独立であり、殆どどのようなデータタイプ/アドレッシングモードでも自由に記述できる。mem-reg, mem-memなどが制約なく選べるのでコンパイラのコード生成時あるいはソフトウェアとしてのコンパイラそのものの開発が容易になる。

(6) 高機能命令

高機能命令はソフトウェア作成時によく洗われるバタンを一まとめにしたものである。ソフトウェアの生産性向上の他にメモリアクセス時の競合による性能低下軽減にも寄与する。一般的なものとして、手続きの呼び出し関連命令と、ループ制御命令がある。それ以外にV60にはOSを意識した高機能命令がある。

a) コンテキストスイッチ命令

b) 非同期トラップ命令

図3はV60がサポートするコンテキストである。大きさは可変となっている。

2. 2 保護機能

V60には0-3の4種類の実行レベルがある。レベル0が一番高い実行権を持つ。実アドレスモードでは特権/非特権の2モードに縮退しているが、仮想アドレスモードではエリアおよびページによる2レベルの保護が可能である。われわれはこの方式をレベル保護方式と呼んでいる。

(1) エリアによる保護

仮想空間にマッピングされる各エリアには、アクセス時の条件を指定できる。指定されたアクセス権とプログラムの実行レベルを比較し、不当なアクセスからエリアの内容を保護する。所謂リング風な保護が可能である。

(2) ページによる保護

エリアによる保護は論理的な保護機構を提供することを目的としているのに対し、ページによる保護はページそのもののデータ属性に因る保護を目的としている。

2. 3 メモリマップト I/O

V60は、OS作成時によく用いられるメモリマップト I/Oをサポートする。I/Oアドレスは仮想空間上にマッピングされ、該当するページにI/Oである旨を指定すると、特権入出力命令を用いずとも通常の MOVE 等命令で入出力が出来る。

2. 4 デバッグ機能

通常のシングルステップ、ブレークポイント以外に、オンチップのアドレストラップ機能がサポートされる。アドレストラップ用に2組のアドレスレジスタとマスクレジスタ及びアクセスモードを指定するレジスタが提供される。

2. 5 FRMモード

高信頼性システムを実現可能とするために、バスサイクル毎に自己（FRMモード）の生成したアドレスおよびデータと他のV60（実行中）の出力するアドレスおよびデータを比較し、不一致が検出されるとトラップ信号を発生する機能を備えている。

2. 7 V20/30エミュレーションモード

16ビットマイクロプロセッサとのソフトウェア継承性を保つために、V20/30エミュレーションモードを備えている。エミュレーションモードのプログラムは実アドレスモードでも仮想アドレスモードでも実行できる。

3 V60用リアルタイムOS [5] [6]

はじめに32ビットの時代になるとリアルタイムOSの役割も変化することを述べた。われわれの設計方針を以下に示す。

- (1) 単なるリアルタイム制御のためのOSではなく、リアルタイム制御+データ処理、インタラクティブな処理が可能なOSを実現する。
- (2) 応用ソフトウェア作成上の制約を可能な限り排除する。
- (3) 高信頼システムの実現を可能とし、広義のソフトウェア生産性を高める。
- (4) 障害発生後の回復処理、再開処理機能等を強化し、フォールトトレラントな対応が可能なOSを目指す。

この上でI-TRONとの融合を図って行く。

3. 1 OSのシステム構成

図4に本OSの基本構造を示す。リアルタイムOSとしての基本機能を提供するリアルタイムカーネル、ファイルサーバや入出力ドライバ等のシステムタスク群、非タスクハンドラおよびTSSスーパーバイザが主な構成要素である。

(1) リアルタイムカーネル

リアルタイムOSとしての基本機能を提供する部分である。タスク管理、タスク間通信、同期、相互排除、タイマ管理および割込管理機能等を提供する。I-TRONの機能はリアルタイムカーネルで実現できる。更に本リアルタイムOSではタスク間通信機能として、Adaのランデブ機能をを提供する。ランデブ機能はカーネルに直接実現されるのではなく、リアルタイムCと呼んでいるC言語のプリプロセッサで処理されカーネルに接続される。[7]

(2) TSSスーパーバイザ

本システムの特徴を示すのがTSSスーパーバイザである。TSSスーパーバイザ自身はリアルタイムカーネルのタスクの集まりである。データ処理指向のタスクはTSSタスクとしてこの上で走行する。例えば、UNIXインタフェースをこの上を実現するにはTSSスーパーバイザとしてスケジューラやページデーモンを実現し、メモリ管理やディスパッチャなどはリアルタイムカーネルで実現する。ファイルシステムとしてUNIXのものを採用するとリアルタイムOS上にUNIXを実現できる。

3.2 リアルタイムカーネル

我々はカーネルを実現するに際し、応用ソフトウェア作成上の制約を可能な限り排除する、高信頼システムの実現を可能とし、広義のソフトウェア生産性を高めると言う要求事項を満たすためにV60のもつ仮想記憶機構と、関連する保護機構を積極的に利用している。

(1) リアルタイムカーネルの保護

V60の実行レベルとエリアでの保護機構を利用し、リング保護システムを形成している。リアルタイムカーネルを最上位に、TSSスーパーバイザおよび通常のタスクを中間に、そしてTSSタスクを最下位の保護レベルに割りつける。

(2) タスク間アクセス保護

V60の仮想記憶機構を利用すると、多重仮想空間を形成することができる。これを図5に示す。本OSでは多重仮想空間のもとで各タスクを各々の仮想空間面に割りつける。これにより各タスクはおのおの4GBまでの空間を利用できる。また異なった仮想空間間のアクセスはV60がチェックするのでタスク間のアクセス保護が実現できる。

(3) I-TRON機能の実現

本OSに限らず、I-TRON機能の実現に於いてで問題になるのはターゲットのマイクロプロセッサ依存の部分である。この問題はI-TRONが仮想マシンアプローチを捨てた事による。I-TRONの基本思想によれば、ターゲットのマイクロプロセッサが異なれば互換性がなくなるわけであるが、この差は最小に留める必要がある。

・割込関連システム・コール

V60は基本的に既存の8/16ビット用周辺LSIを利用する。割込のメカニズムもμPD71059(8259同等)をもちいるベクタ方式である。従って、I-TRONの仕様を殆ど継承できる。但し、fet_dat システム・コールは86系アーキテクチャ特有の問題を回避するために設けられたので、本OSでは不要となる。割込専用スタックポインタISPの利用により、割り込まれたタスクとは全く無関係なデータ領域を確保出来る。

・アクセスアドレス

16ビットI-TRONでは実行時の性能を上げるために、アクセスアドレスと称して各種コントロールブロックのベースアドレスをユーザに見せている。仮想記憶システム下ではこの情報の持つ物理的意味は異って来る。もしこれに相当するものがあるとするれば、仮想空間番号と仮想アドレスである。本OSではアクセスアドレスを単なるキーあるいはトークンとしか扱っておらず、仮に16ビットI-TRON(RX116)のユーザがアクセスアドレスを積極的に利用していた場合上位互換性が失われる。特にメールボックスの利用時に問題が顕著である。

・メールボックスにおけるメッセージの渡し方

本OSではポインタの受渡しによるメッセージのコピーなどは行わず、共有空間を用いてメッセージの交換をしている。

・ cre_tsk 時のパラメータ等

16ビットI-TRONにはスタックサイズなどの資源をユーザが直接指定する部分がある。本OSでは、あるいは32ビットリアルタイムOSとしては適当なパラメータ指定とは考えにくい。また、タスクの生成ごとに度々パラメータを指定するのも煩わしい。本OSではシステムジェネレーション時に指定した主なパラメータを継承する方式を検討している。

(4) V60 提供機能の利用

次にV60の持つ機能の利用について述べる。V60は高級言語およびオペレーティングシステム指向の命令を各種サポートしている。以下、その利用法を簡単に述べる。

- ・ コンテキストスイッチ命令
タスクのスイッチ
- ・ 非同期システムトラップ命令
ディスパッチ時の事象保留 (図6)
- ・ 先行1検出命令
レディキュー操作

参考文献

- [1] 坂村、"I-TRON" 情報処理学会オペレーティングシステム研究会資料、Sept. 1984
- [2] 門田、"V20/V30用リアルタイムOS I-TRON", bit, July 1985
- [3] Y.Yano et al "A 32b CMOS VLSI Microprocessor with On-Chip Virtual Memory Management", Proceeding of ISSCC, pp 36-37, 297, Feb. 1986
- [4] 佐藤他 "仮想記憶管理機構と浮動小数点演算機構を内蔵したマイクロプロセッサV60" 日経エレクトロニクス (TO BE PUBLISHED)
- [5] 古城他 "32ビットマイコン用リアルタイムOSの試作 -設計方針-"、情報処理学会第31回全国大会予稿、pp 245-246, 1985
- [6] 下島他 "32ビットマイコン用リアルタイムOSの試作 -カーネルの実現-"、情報処理学会第31回全国大会予稿、pp 247-248, 1985
- [7] 世良他 "通信処理用汎用リアルタイムOSの交換処理への適用" 電子通信学会交換研究会資料、1986

32ビットマイクロプロセッサ向けリアルタイムOSの構成

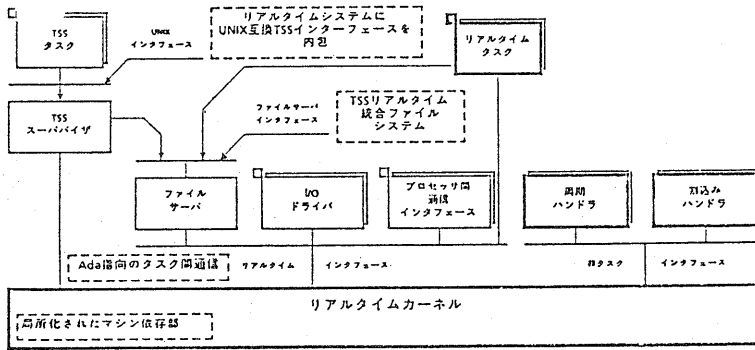


図 5

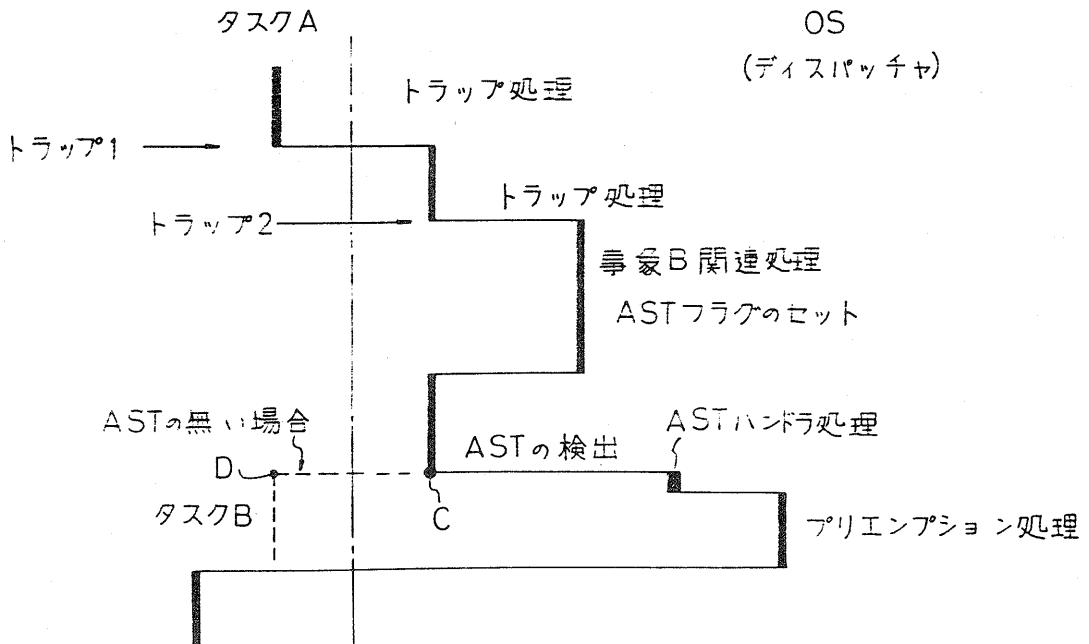


図 6