

## μITRON仕様に準拠したG<sub>MICRO</sub>シリーズ用リアルタイムOS:MR3200の開発

坪田 秀夫 中田 清 斉藤 和則

三菱電機(株)LSI研究所

TRONチップ仕様32ビットマイクロプロセッサ“G<sub>MICRO</sub>シリーズ”用リアルタイムOSとして、機器組み込み型制御をターゲットとするμITRON仕様に準拠した“MR3200”を開発した。本OSはG<sub>MICRO</sub>シリーズの特長を活かしてタスク数や優先度数に依存しない応答性を持ち、サーボ系やセンサ系の制御に要求される十数μ秒以下の割り込み禁止期間と約18μ秒という高速なタスク切り替え時間を実現している。

また、本OSは既に製品化されている当社16ビットシングルチップマイコンMELPS7700シリーズ用μITRON仕様OS“MR7700”とC言語レベルで完全に互換性を持っており、8ビットから32ビットまでのリアルタイムOSの応用分野を広げることが可能となった。

### Implementation of MICRO-ITRON Specification on G<sub>MICRO</sub> Series 32bit Microprocessors

Hideo Tsubota Kiyoshi Nakata Kazunori Saitoh

LSI R&D Laboratory, MITSUBISHI ELECTRIC CORPORATION

4-1 Mizuhara, Itami City, Hyogo 664, Japan

The MICRO-ITRON specification has been designed for the critical realtime applications. MR3200 was developed as the realtime OS based on MICRO-ITRON specification with G<sub>micro</sub>/100, which was based on TRONCHIP specification and one of G<sub>micro</sub> series. MR3200 could be realized interrupt inhibit time less than 15 micro sec. and quick response time (task switching time was 18 micro sec.) independent of number of tasks and priorities.

Mitsubishi had already implemented MICRO-ITRON specification on original 16-bit single-chip microcontroller called MELPS7700, and two OSs have perfect compatibility with application programs written in C language. Combination with MR3200 and MR7700 makes the realtime OS' world more wide.

### 1. はじめに

当社ではTRONプロジェクトの一環としてTRONチップ仕様準拠32ビット・マイクロプロセッサ：G<sub>MICRO</sub>/100の開発と共にμITRON仕様リアルタイムオペレーティングシステム：MR3200の開発を完了した。本リアルタイムオペレーティングシステム（OSと称す）はG<sub>MICRO</sub>/100の特長である高性能と機器組み込み型制御の用途を活かし、高速なタスク切り替えと応答性を実現した。本OSは十数μ秒の応答性を必要とする機器制御にも対応でき、リアルタイムOSの応用分野の裾野を広げることが可能となった。本論文ではMR3200の設計指針、機能概要及び評価結果について述べる。

### 2. G<sub>MICRO</sub>/100の特長

G<sub>MICRO</sub>/100はG/200、G/300とともにシリーズ化されたTRONチップ仕様に準拠した32ビットマイクロプロセッサであり、16本の汎用レジスタと以下のようなシリーズ共通の特長を備えている。

- 豊富で対称性のよいアドレッシングモード（多段間接モード）
- 高性能（G<sub>MICRO</sub>/100：6～10MIPS）
- OS関連命令（LDCTX/STCTX、~~キ~~操作命令、DI機能、JRNG命令）
- ビットマップディスプレイ操作（任意長ビットフィールド命令）
- 4レベルのリングブザー

また、G<sub>MICRO</sub>シリーズの中でG<sub>MICRO</sub>/100は以下の特長をもつ。

- MMU、FPUサポートを外した廉価版
- 機器組み込み型コントローラ
- ASICへの展開が可能（G<sub>MICRO</sub>/100のチップサイズは10mm<sup>2</sup>）

### 3. μITRON仕様

TRONプロジェクトではリアルタイムOS仕様として、μITRON仕様とITRON仕様が策定されている。μITRON仕様は機器組み込み型制御を狙いとした高速、コンパクトな仕様である。一方、ITRON仕様は豊富な機能と強力な例外管理やオブジェクトの動的生成機能を備え、機器組み込み型からディスクベースシステムまで構築できる仕様となっている。

μITRON仕様は図3.1に示すようにITRON仕様のサブセットであり、システムコールのインプリメンテーションは1～5にレベル分けされ、チップ仕様やアプリケーション分野に最適なシステムを構築できる。また、各システムコールは図3.2の様にオプションによってシステムコールをプリミティブ化して高速処理とコードサイズの最適化を可能としている。

従って、我々は機器組み込み型制御分野をターゲットとするG<sub>MICRO</sub>/100用のリアルタイムOS仕様に、高速性とコンパクト性に優れたμITRON仕様を採用した。

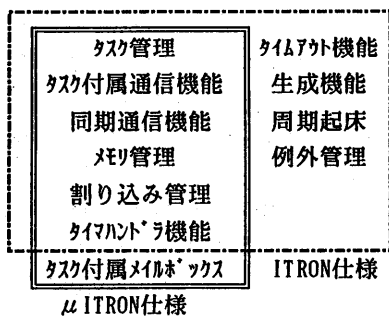


図3.1 ITRON仕様の機能構成

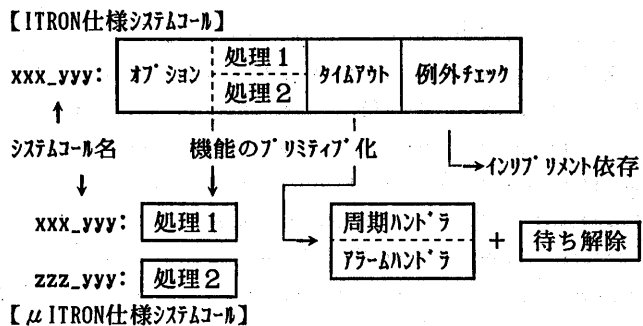


図3.2 μITRON仕様の最適化



## 2) ディスパッチング

タスクディスパッチングでは実行可能な最高優先度のタスクを検索し、コンテキストをスイッチングする。MR3200では図4.1のように優先度の検出に任意長ビットフィールドのBVSC命令を用いて比較、分岐処理を伴わない一定処理時間を実現した。ビットフィールドの終端はアイドルタスク（常にREADY/RUN状態）の優先度の255である。ここで、第0ビットと最終の第255ビットまでサーチする処理時間差はほぼ0.4μ秒以下であり、リアルタイムOSの処理時間からみれば一定とみなしてもよい。

## 3) スケジューリング

システムコールの実行によってタスク状態が遷移するとタスクをレディーキューにエンキュー、デキューする。この時、キュー状態の変化に伴うレディーキュービットのセット/クリアにGMICROシリーズのビット操作（BSET、BCLR）命令を用いた。これらはビットオフセットを任意の値に指定できるので優先度値に依存することなく、また1命令で完了するので処理時間が一定となる。

### (2) コードサイズのコンパクト化

32ビットCPUではオペランドにメモリ間接や相対アドレッシングを用いると32ビットを必要とするのでコードサイズが大きくなり、また低速メモリを使用するとメモリウェイトが入って処理時間が低下する。そこで、MR3200ではOSのローカル変数を全てレジスタ上に置き、処理の高速化とタスクのコンテキスト領域以外はスタックも使用しないようにしてメモリの節約を計った。また、システムコールは4.3に示すようにサブルーチン形式をとっているため、システムコールレベルでコードサイズの最適化が可能である。

## 4.3 機能の実現

μITRON仕様に基づいてインプリメントする機能、及びシステムの実行環境を以下の様に設定した。

### (1) インプリメンテーションレベル

広範囲のアプリケーションに対応できるITRON仕様相当の機能を実現するために5個のタスク状態（RUN, READY, WAIT, SUSPEND, DORMANT）とレベル3までの全機能、更に要求待ち状態のタイムアウト機能を提供するためにレベル4（周期/アラームハンドラ）及びレベル5（強制待ち解除）の機能をインプリメントした。また、イベントフラグは多様な用途に対応できるように32ビット仕様をサポートした。図4.2はMR3200の機能構成を、図4.3はインプリメントした機能レベルを示す。尚、ハンドラの定義はシステム立ち上げ時に定義する方法をとっている。

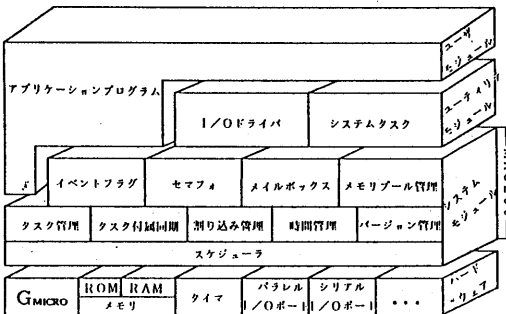


図4.2 MR3200の機能構成

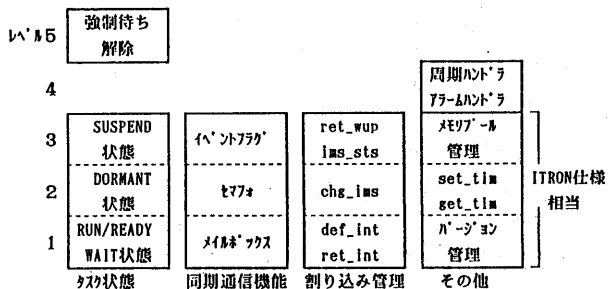


図4.3 インプリメンテーションレベル

### (2) 実行環境

μITRON仕様ではOSの動作環境やシステムコールのアセンブラインタフェースは規定されてい

ない。また、Gmicroシリーズは5~6段のパイプライン処理を行っているので、TRAPA命令やDI機構を用いるとパイプラインがキャンセルされ、それらのオーバヘッドだけでも数μ秒かかる。本OSの開発目的は機器組み込み型（静的なシステム）をターゲットとする高速性、応答性を実現することなので、リング保護機構は使用していない。つまり、タスクはOSと同じリング0で動作し、システムコールの起動はサブルーチン形式とした。また、パラメータは極力、レジスタに格納してバケット構造はとらないようにした。

表4.1 動作モード

	RNG	スタック
OS	0	SPO
タスク	0	SPO
割り込みハンドラ	0	SPI

パラメータ: R1~R3  
 システムコール起動: BSR \_sys\_XXX\_yyy  
 リターンパラメータ: R0←戻りコード  
 R1~R3

図4.4 アセンブリ言語ファームウェア

### 5. 評価結果

MR3200の基本性能を表5.1に示す。MR3200はGmicroシリーズ全てで動作可能である。性能評価はGmicro/100 (20MHz、No-Wait)で行った。その結果、タスク切り替え時間は優先度に依存することなく、待ち状態のタスクを起床させるwup\_tsk()システムコールで約18μ秒（その間の割り込み禁止期間が約9μ秒）、最も時間のかかるタスク優先度を変更するchg\_pri()システムコールで24μ秒（同、15μ秒）という高速な応答性を得た。コードサイズはリアルタイム、マルチタスクシステムで必要不可欠な4個のシステムコールで約2.1KBとコンパクトである。また、表5.2はシステムで使用するRAM容量であり、システムが使用するスタックはタスクのコンテキストセーブ領域（72B）のみとなっている。

表5.1 MR3200の性能仕様

項目	仕様
ターゲットロケッサ	Gmicroシリーズ
最大タスク数	65535個
優先度数	1~255
システムコール数	53個
OSコードサイズ	約2.1KB <sup>*0</sup> ~約11.6KB
OS核データサイズ	181B <sup>*1</sup> ~
OS核記述言語	アセンブリ言語
タスク切り替え時間	約18μ秒 (タスク起床) ~ 24μ秒 (優先度変更)
最大割り込み禁止時間	約9μ秒 (タスク起床) ~ 15μ秒 (優先度変更)

表5.2 RAM容量

データ領域	RAM容量
システム	72+8×優先度数+割り込みスタック
タスク	29/タスク+72(スタック)+α <sup>*2</sup>
イベントフラグ	13/フラグ
ヒープ	10/ヒープ
メールボックス	20/メールボックス+バッファサイズ
メモリプール	14/メモリプール+プールサイズ

\*2: アリケーションスタックサイズ

\*0: slp\_tsk(), iwup\_tsk(), ret\_int(), sta\_tsk()

\*1: タスク (ID=1, 優先度=1) を1つ含む

表5.3は当社のμITRON仕様OSの性能比較である。チップの仕様や動作周波数などは異なるが、MR7700の約1.5倍の性能を得ている。

表5.3 μITRON仕様の性能比較

タスク切り替え処理	MR3200 Gmicro/100 20MHz, No-Wait	MR7700 M37700M2 16MHz, No-Wait
タスク起床 (割り込み禁止期間)	18μ秒 (9μ秒)	29μ秒 (20μ秒)
イベントフラグ	26μ秒	38μ秒
ヒープ	21μ秒	30μ秒
メールボックス	21μ秒	35μ秒

## 6. システム構成とアプリケーション開発手順

アプリケーションの開発手順ではシステム生成とアプリケーションプログラムの作成に分けられる。システム生成はMR3200のシステムプログラムをライブラリ化する。この作業はシステム導入時に最初に一度行うだけでよい。MR3200を用いたアプリケーション開発フローを図6.1に示す。

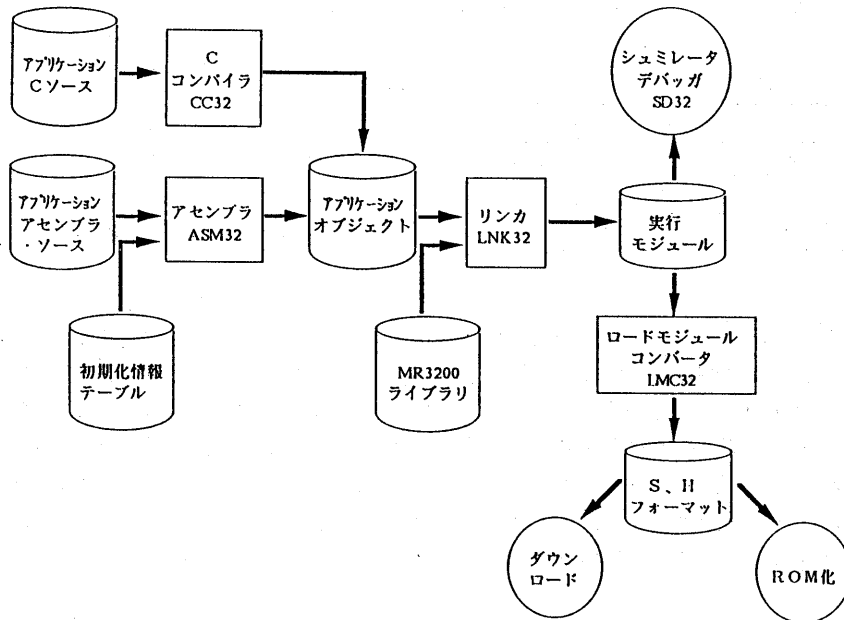


図6.1 MR3200のアプリケーション開発フロー

## 7. まとめ

今回我々はTRONチップ仕様32ビットマイクロプロセッサGMICROシリーズ上に $\mu$ ITRON仕様準拠のリアルタイムOS:MR3200を開発した。その結果、タスク数や優先度に依存することなくタスク切り替え時間が最大で約24 $\mu$ 秒、最大割り込み禁止期間が約15 $\mu$ 秒と非常に高速で応答性の優れた性能を得た。これはGMICROシリーズのもつ高性能と $\mu$ ITRON仕様の軽快さの融合であり、リアルタイムOSの応用分野の裾野の拡大が可能となった。

GMICROシリーズは32ビットの応用分野に対してチップ仕様をシリーズ化している。それに対応してリアルタイムOSのシリーズ化も必要であり、 $\mu$ ITRON仕様でカバーできない動的なシステム運用やプロテクションを必要とする分野などをターゲットとしてITRON仕様OSの開発も行っていく予定である。

最後に、MR3200の開発に関して御指導頂いた東京大学の坂村助教授、並びにLSI研究所の関係各位に深く感謝致します。

### 【参考文献】

- [1]坂村健；“リアルタイムオペレーティングシステム—ITRON”，日本計算機学会誌Vol.3, No.5, 1985.
- [2]坂村健；“Micro-ITRONの設計方針”，トロン技術研究会, Vol.1, No.1, p.1-17, トロン協会, Jun. 1988
- [3]坂村健他；“リアルタイムOS, Micro-ITRONを1チップマイコンの内蔵ROMに搭載”，日経エレクトロニクス, 1988.8.22, p.193-200