

# 汎用PCにおけるリアルタイム制御機構の実現（その2）

5C-4

片山吉章      川上武      黒澤寿好

三菱電機（株） 情報技術総合研究所

## 1 はじめに

汎用パソコン上で、Windowsアプリケーションと制御プログラムを共存させることは、制御システムへの適用が最も有力視されているWindows NTでも、アプリケーションレベルでは十分な性能を達成していない。著者等は、この問題を解決するために、制御プログラムをWindows NTのドライバレベルで動作させる、リアルタイム制御機構を開発した。本稿では、このリアルタイム制御機構の概要と、1軸倒立振り子制御を用いた評価モデルによる、市販ソフトウェアと制御プログラムの共存性評価について述べる。

## 2 リアルタイム制御機構の概要

リアルタイム制御機構は、図1に示すような構造で、Windows NTの1デバイスドライバとして疑似的なリアルタイムOS環境を構築しており、制御プログラムはその内部で動作するリアルタイムタスク（RTタスク）として実現される。

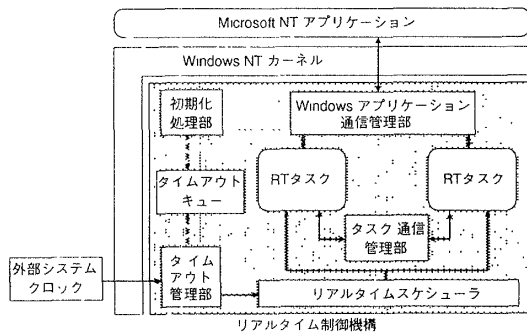


図1 リアルタイム制御機構の構造

An Implementation of Real-Time Control Mechanism for Personal Computers(2)  
 Yoshiaki Katayama, Isamu Kawakami and Hisayoshi Kurosawa  
 Mitsubishi Electric Corp.  
 Information Technology R&D Center  
 5-1-1 Ofuna, Kamakura, Kanagawa 247, Japan

## 2.1 リアルタイム制御機構の特徴

1. リアルタイムタスクは、リアルタイム制御機構のルーチンとして実装している。各タスクは、優先度・起動周期時間・名称の属性を持ち、複数のリアルタイムタスクの起動要求が生じた場合には、優先度に従ったスケジューリングを行なう。また、起動周期時間の設定値によって、周期起動/非周期起動に分けることもできる。
2. リアルタイム制御機構は、外部装置から発生するクロック割り込みを受付けるタイムアウト管理機能を持たせることにより、一般のリアルタイムOSとほとんど同様な周期起動、タイムアウト機構をリアルタイムタスクに提供することができている。周期起動のタスクは、タスク生成時にタイムアウトキューに登録されており、周期の到達したタスクは、タイムアウト管理部からリアルタイムスケジューラに起動要求が出されることになる。
3. Windowsアプリケーションとの通信は、Windowsアプリケーションへ特別なシステムコールを追加しているわけではなく、リアルタイム制御機構というWindows NTの論理デバイスへの入出力要求に対応付けしている。また、リアルタイムタスクへは、リアルタイムタスク間のメッセージ通信と等価なインタフェースを提供している。

## 2.2 Windows NT ドライバ構造との対応

Windows NTでの割り込み処理は、Interrupt Service Routine (ISR) とDeferred Procedure Call (DPC) ルーチンに分けて行われる。ISRは、割り込みを発生するデバイスのドライバに必要なルーチンであり、割り込みが発生したときに呼び出されるものである。ISRはデバイス割り込みと同じ優先度で実行されるため、通常長い時間を必要とする処理は実行されない。そのため、ISR

は DPC ルーチン呼び出し、そこで処理を継続する。DPC ルーチンは、ISR よりも優先度が低く、アプリケーションのスレッドよりも高い優先度で実行される。

本機構では、Windows NT のシステムタイマと独立したタイマをシステムクロックとして使用する。このタイマによって、ハードウェア割り込みを周期的に発生させ、本機構が用意した ISR (タイムアウト管理部) を呼び出す。ISR では、本機構のリアルタイムスケジューラを DPC ルーチンとして呼び出し、スケジューラがリアルタイムタスクのスケジューリング及びディスパッチを行なう。このため、DPC ルーチンよりも優先度の低いアプリケーションのスレッドの負荷にかかわらずリアルタイムタスクを起動することが可能となる。

### 3 評価

1 軸倒立振子の制御モデルを用いて、本機構の評価を行なった。評価を行なった環境は以下の通りである。

CPU: Pentium 120 MHz

OS: Microsoft Windows NT Version 3.51

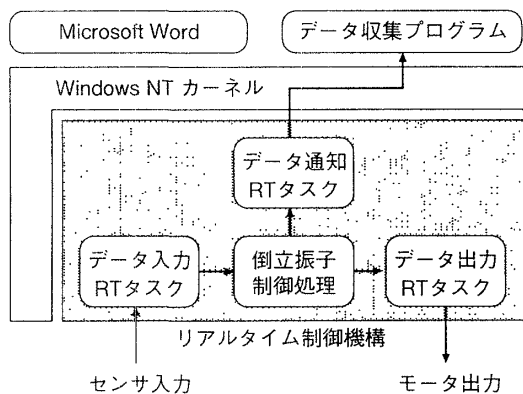


図2 1軸倒立振子の制御モデル

1 軸倒立振子とは、横方向のみ動作する台車の上に金属の棒を立て、その棒が常に立つように台車の制御を行なう装置である。

制御プログラムは図2に示すように、10msec 周期起動でセンサからデータを入力するタスク (データ入力 RT タスク) と、そのデータをもとに傾斜補正値を計算するタスク (倒立振子制御処理)、計算したデータをもとにモータを駆動するタスク (データ出力 RT タスク) で構成されてい

る。また、評価用としてデータ入力 RT タスクの起動時間を計測するために、データ通知 RT タスクとデータ収集プログラムを付加している。

Windows アプリケーションとしては、Microsoft Word を使用し、文章ファイルのオープン、セーブ等の操作を繰り返した。

その結果、アプリケーションレベルでは立たせることができなかった振子を、本機構を用いることによって立たせることが可能となった。

図3に、10msec 周期起動に設定した倒立振子の制御タスクが、周期に対してどのくらい遅延したかを示す。グラフの横軸は遅延時間であり、縦軸はその遅延時間を要した回数である。

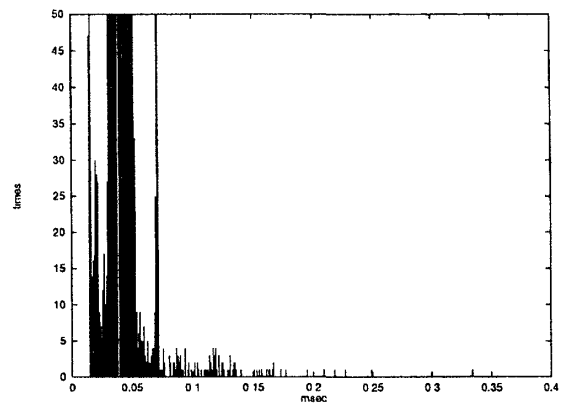


図3 周期タスク起動時の遅延時間

約 10 万回の測定データから、周期タスクの起動遅延時間は 0.35msec 以内であり、制御システムへの適用に十分なリアルタイム性を実現していることが確認できた。

### 4 おわりに

本稿では、Windows NT のデバイスドライバとしてリアルタイム制御機構を実現したこと、倒立振子をモデルにした制御プログラムと Windows アプリケーションの共存性評価について述べた。

本機構を用いることによって、Windows アプリケーションとリアルタイムタスクとの並行動作が実現可能となり、それらのアプリケーション間での協調作業が容易に実現可能となる。

### 参考文献

- [1] 川上, 片山, 黒澤: 汎用 PC におけるリアルタイム制御機構の実現 (その 1), 情報処理学会第 54 回全国大会, 1997