

汎用 PC におけるリアルタイム制御機構エミュレータの実現

3Q-2

片山吉章 竹並春佳 川上武 二村祐地

三菱電機（株）情報技術総合研究所

1 はじめに

我々は、制御システムにおけるパーソナルコンピュータ（PC）の適用領域拡大を目的に、Windows NT 上でミリ秒単位の周期処理を実現するリアルタイム制御機構の開発を行っている [1][2]。リアルタイム制御機構を用いることにより、単一プロセス上でリアルタイム制御プログラムと Windows アプリケーションの共存が可能となる。

今回、リアルタイム制御機構のプログラム開発環境を充実させるため、C 言語による制御プログラム開発を支援するリアルタイム制御機構エミュレータを実現した。本エミュレータは、Windows NT のデバイスドライバとして動作するリアルタイム制御機構を Windows プロセス上でエミュレートするものである。本稿では、その実現方法について述べる。

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、Windows NT のデバイスドライバの 1 つとして実装しており、リアルタイム OS 環境を提供している。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として実現される。以下に、本機構が提供する主な機能を示す。

- マルチタスク実行
- 周期／非周期タスク実行
- 固定優先度スケジューリング
- タスク間メッセージ通信
- Windows アプリケーションとのメッセージ通信

3 問題点と解決策

リアルタイム制御機構では、リアルタイムタスクは Windows NT のデバイスドライバ内部に組み

An Implementation of Emulator for Real-time Control Mechanism on Personal Computers
Yoshiaki Katayama, Haruka Takenami,
Isamu Kawakami and Yuji Nimura
Mitsubishi Electric Corp.
Information Technology R&D Center
5-1-1 Ofuna, Kamakura, Kanagawa 247-8501, Japan

込まれる。そのため、制御プログラム開発は、デバイスドライバ開発と同一の環境を必要とする。また、制御プログラムにバグが存在した場合、Windows NT の再起動を避けることはできないため、そのデバッグは困難なものとなっている。

これらの問題点を解決するために、我々は以下に示す 2 つの方法を検討した。

- ソフト PLC 開発／実行環境の実現
FA 分野において、すでに主流となっている PLC と同様の開発／実行環境をリアルタイム制御機構上で実現する。
- リアルタイム制御機構エミュレータの実現
リアルタイム制御機構を Windows プロセス上でエミュレートし、Windows アプリケーションと同様のデバッグ環境を提供する。

前者は、すでに実現しているが [3]、今回新たに後者についてもその実現を図ることとした。これは、きめ細かい制御の記述や開発者の使いやすさ向上のため、C 言語での開発環境の充実も必要と判断したことによる。以下に、開発したリアルタイム制御機構エミュレータおよび開発支援ツールについて、それぞれその実現方法を述べる。

4 リアルタイム制御機構エミュレータの実現

デバイスドライバとして動作するリアルタイム制御機構の構造を図 1 に、今回実現した Windows プロセスとして動作するリアルタイム制御機構エミュレータの構造を図 2 に示す。リアルタイム制御機構と同等の動作を Windows プロセスとして実現するため、次の 3 つの機能を変更／追加した。

- 1) 周期起動用タイマの変更
- 2) API（ドライバサポートルーチンおよび RTPI）のエミュレーション機能の追加
- 3) 代理ドライバによるハードウェアアクセスおよび Windows アプリケーション通信の中継機能の追加

4.1 周期起動用タイマ

リアルタイム制御機構では、周期起動用として高精度な外部タイマも使用できるようにしている

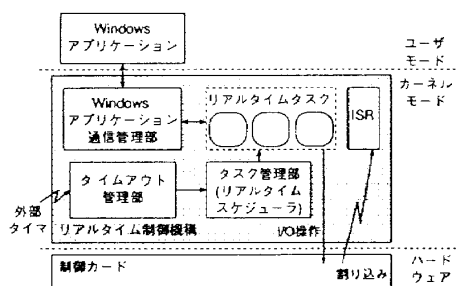


図1 リアルタイム制御機構の構造

が、制御プログラム開発時は外部タイマが無くても開発が進められるように、本エミュレータでは周期起動用に Windows NT のマルチメディアタイマ機能を用いることとした。

4.2 APIのエミュレーション機能

リアルタイムタスクのプログラムが用いるドライバサポートルーチンおよび RTPI (Real-time Task Programming Interface: リアルタイム制御機構が提供) の各機能実現のため、これらのエミュレーションを行うライブラリ (サポートルーチンエミュレーション部, RTPIエミュレーション部) を作成した。

4.3 代理ドライバ

I/O 操作や割り込みサービスルーチン (ISR) の登録などハードウェアとのやりとりは、Windows プロセス上で実行できないため、代理ドライバにその処理を依頼することで実現した。

ISR については、本エミュレータで動作する場合、リアルタイムタスク内の関数を ISR として登録できないため、代わりに代理ドライバ側で用意した ISR を登録するようにした。割り込みが発生すると代理ドライバ側の ISR を経由して、エミュレータ内のイベント受信部により、対応するリアルタイムタスク内の関数が呼び出される。

また、リアルタイムタスクと通信する Windows アプリケーションはドライバに対して通信を行うため、Windows アプリケーションとの通信も代理ドライバを経由して行うようにした。

5 開発支援ツールの実現

開発支援ツールは、Microsoft Visual C++ Version 4.2/5.0 (VC) のプラグインであり、表1に示す機能を実現している。以下に本ツールを使ったリアルタイムタスク開発手順を示す。

1. VCの開発環境である Microsoft Developer Studio を起動し、リアルタイムタスク開発用プロジェクトファイルを開く。
2. リアルタイムタスクの雛型を生成する。

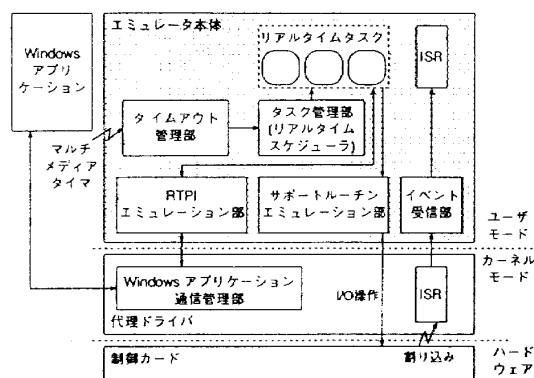


図2 リアルタイム制御機構エミュレータの構造

表1 開発支援ツールの機能

リアルタイムタスクの雛型生成
リアルタイムタスクのコンフィグレーション
リアルタイム制御機構のコンフィグレーション
リアルタイム制御機構のビルド

3. 雛型をもとにリアルタイムタスクのコーディングを行う。
4. リアルタイム制御機構エミュレータを用いて、VCの開発環境下でデバッグを行う。
5. デバッグを終えたリアルタイムタスクを組み込み、リアルタイム制御機構 (ドライバ) のビルドを行う。
6. ドライバをインストールし動作確認を行う。

本手順により、アプリケーションレベルで十分なデバッグを行うことができるため、多くの場合カーネルモードでのデバッグが不要となる。そのため、単一マシンでの開発が可能となった。

6 おわりに

リアルタイム制御機構上での制御プログラム開発環境を充実させるため、リアルタイム制御機構エミュレータを実現した。

今後は、実現したリアルタイム制御機構エミュレータを実際の計装制御システム開発に適用していく予定である。

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
- [2] 片山, 川上, 黒澤: “汎用 PC におけるリアルタイム制御機構の実現 (その2)”, 情報処理学会第54回全国大会, 1997
- [3] 片山, 竹並, 川上, 黒澤: “汎用 PC におけるソフト PLC の開発/実行環境の実現”, 情報処理学会第56回全国大会, 1998