

リアルタイム OS 用マルチ・タスク・デバッガの実現方式

2E-6

橋本 一也* 金子 善則** 高倉 規彰* 中島 隆*

* 日本電気(株)

ULSI システム開発研究所

** 日本電気マイコンテクノロジー(株)

基本ソフトウェア部

1 はじめに

近年、マイコン組み込みシステムにおいて、プログラミングの容易さや並列処理による効率化のために、リアルタイム OS(以下 OS)を用いることが多くなってきている。これに伴い、OS 上のタスクを効率よくデバッグできるタスク・デバッガが開発されてきた。

従来、タスク・デバッガは、モニタやデバッグ用のタスクで実現されていたが、これらの方式では、ユーザによる組み込み作業が必要であることに加えて、メモリの制約を受けるという欠点があった。特にメモリ容量の小さなマイクロコントローラを利用するとき、この制約により、タスク・デバッガを実現するのは困難な場合が多い。そのため、一般的にデバッグ装置として利用されているインサーキット・エミュレータ(以下 IE)上でタスク・デバッガを実現することが望ましい。

一方、デバッグを行なうターゲット・システムが自動車のエンジン制御やモータ制御などのリアルタイム制御装置である場合に、ターゲット・システムのソフトウェアが停止してしまうと装置を破損してしまう恐れがあるが、リアルタイム応答性を保証することによって装置の破壊を回避することが可能になる。上記理由により、ターゲット・システムのソフトウェアのリアルタイム応答性を損なわずデバッグ可能とすることがタスク・デバッガの重要な機能として要求されている。

本論文では、このタスク・デバッガの実現にあたっての必要機能を洗い出し、その実現方式について考察する。

2 タスク・デバッガの機能

本デバッガのシステム構成を図1に示す。

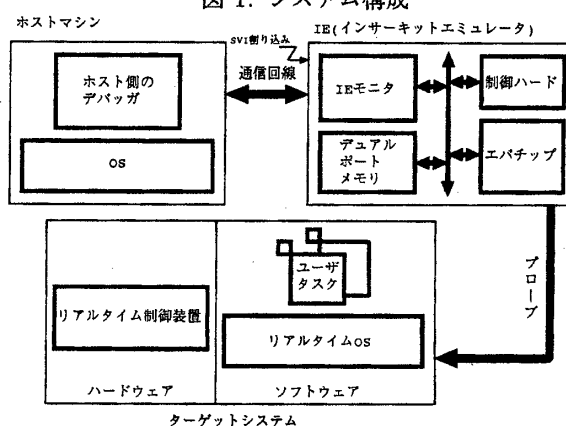
本デバッガの機能を次に示す。

- 任意タスクの停止 / 再開機能
- システムコールの発行機能
- 任意タスクのレジスタ表示 / 設定機能

本デバッガはデバッグ対象外のタスクや割り込み処理に影響を与えることなくタスク毎のデバッグ機能を提供

するものであり、デバッグ対象タスクのみを停止させることやデバッグ対象タスクのレジスタを参照することができる。

図 1: システム構成



3 問題点

本デバッガの機能を実現する上での問題点を次にまとめる。

3.1 リアルタイム応答性

ユーザがIEの操作を行なったときに、IEモニタは、ホストマシンからのコマンドを解釈して実行する。IEモニタの動作中はターゲット・システムのソフトウェアは停止しているが、本デバッガでは、ターゲット・システムがリアルタイム制御装置であることを想定しているため、リアルタイム応答性を損なってはならない。

現状のIEではIEモニタを動作させるためのスーパバイザ割り込み(以下SVI割り込み)の優先度は最高レベルであり、ユーザ割り込み処理中にSVI割り込みが発生すると、ユーザ割り込み処理が中断されてしまう。更に、IEモニタ動作中(SVI割り込み中)はユーザ割り込みが禁止状態になっているので、ユーザ割り込みが発生してもIEモニタ動作が終了するまで遅延されてしまう。このような点から、現状のIEの実装はリアルタイム応答性の保証が困難である。

3.2 IEモニタ動作時のデータ整合性

OS資源内容の変更や実行タスクの切替を行なうOSの処理はデータの整合性を維持するために、その処理を

Implementation of online debugger using In-circuit Emulator.

Kazuya Hashimoto*, Yoshinori Kaneko**,

Noriaki Takakura*, Takashi Nakajima*

*NEC Corporation.

**NEC Microcomputer technology.

開始してから終了するまでの区間を割り込み禁止状態にしている。そのような処理区間を OS クリティカル処理区間と呼ぶ。

2章で定義したタスク・デバッグ機能はこのデータの整合性が維持されているとき、つまり、ターゲット・システムが OS クリティカル処理区間外で停止しているときに行なわなければならない。

しかしながら、現状の IE の実装における SVI 割り込みはノンマスカブル割り込みであるため、SVI 割り込みが発生した時点で強制的にターゲット・システムのソフトウェアを停止して、IE モニタが動作する。このため、ターゲット・システムのソフトウェアが OS クリティカル処理区間を実行中に SVI 割り込みが発生した場合にデータの整合性を保証できない。

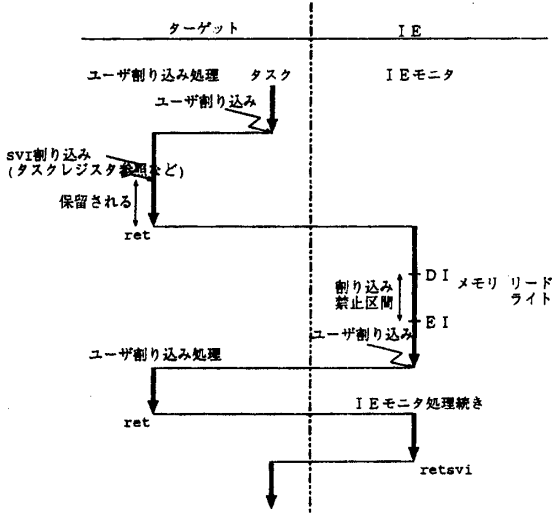
4 実現方法

前章であげた問題点に対し、IE の SVI 割り込みを以下のように機能改善することにより、ユーザ割り込みに対して、ほぼリアルタイムに応答可能なタスク・デバッグが実現できる。

4.1 リアルタイム応答性の維持方法

ユーザ割り込み処理を妨げずにデバッグ処理を行なうため、エバチップの SVI 割り込みの優先度をユーザ割り込みよりも低く設定可能にし、本デバッグ起動時に SVI 割り込みの優先度を最低に設定しておく。

図 2: デバッグ中のユーザ割り込み許可



これにより、ユーザ割り込み中に SVI 割り込みが発生した時には直ちに IE モニタが動作せず、ユーザ割り込みを優先して割り込み処理ルーチン終了後に IE モニタが実行されるようになる。

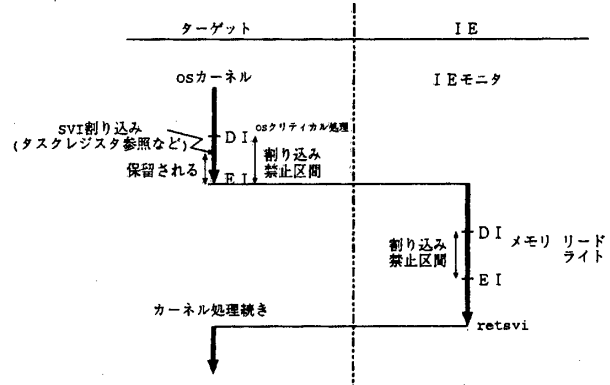
さらに、SVI 割り込み中にユーザ割り込みを受け付け可能とすることにより IE モニタ動作中にユーザ割り込みが発生しても、IE モニタ処理を中断し、直ちにユーザ割り込み処理ルーチンが実行可能になり、ユーザ割り込みが遅延される問題は解決される (図 2)。

4.2 データ整合性の維持方法

SVI 割り込みをマスカブル割り込みと同様に割り込み禁止/許可命令でマスク可能にする。

OS のクリティカル処理区間は割り込み禁止となっている。従って、OS クリティカル処理実行中に SVI 割り込みが発生しても OS のクリティカル処理区間の実行が終了するまで IE モニタの実行が保留され、(図 3) IE モニタが動作したときにはデータの整合性は維持されており問題は解決する。これにより、タスク・レジスタの参照やシステムコールが発行可能となる。

図 3: OS クリティカル処理中の IE モニタ実行回避



5 まとめ

本論文で、IE を利用したタスク・デバッグを提案し、その実現方式を検討することによって、ターゲット・システムのソフトウェアのリアルタイム応答性を損なわないタスク・デバッグが実現可能であることがわかった。

今後は、タスクが任意のシステムコールを発行したときに停止するシステムコール・ブレイクや任意の OS 資源をアクセスしたときに停止するオブジェクト・ブレイクなどの機能の追加を検討し IE を利用したタスク・デバッグの機能の充実を図りたい。

参考文献

[1] 橋本氏他, "インサーキット・エミュレータを利用した OS デバッグ機能の実現", 情処第 48 回 (平成 6 年前期) 全国大会講演論文集