

4V-1 プロセッサアーキテクチャの仮想化について  
— 割り込み要因論理化の一手法 —

小野 亮治 花沢 満 大類 駿 大久保 利一

NTT 電気通信研究所

1. はじめに

ソフトウェアの規模、複雑さが一層増大してきた現在、その開発期間、コスト削減のためには、異種プロセッサ間でのソフトウェア流通性の向上が重要な課題である。このためには、ソフトウェアをプロセッサアーキテクチャに依存しない構成とすることが必要である。本論文では特に割り込み処理について、プロセッサアーキテクチャに依存した物理的割り込み要因を論理的割り込み要因として上位ソフトに見せるメカニズムを提供することで、その流通性を向上させる手法について述べる。

2. 割り込み処理仮想化の位置付けと狙い

これまでにプロセッサのCPU資源、メモリ資源の仮想化についてはそれぞれ、プロセス、プロセス空間といった仮想化技術により実現して来たもののに対し、割り込みについては有効な手法が提案されていない<sup>2)</sup>。従って、割り込み処理についても仮想的な割り込みインタフェースを設けることで上位ソフトの流通性を向上させる必要がある。図1に割り込み処理仮想化の位置付けについて示す。

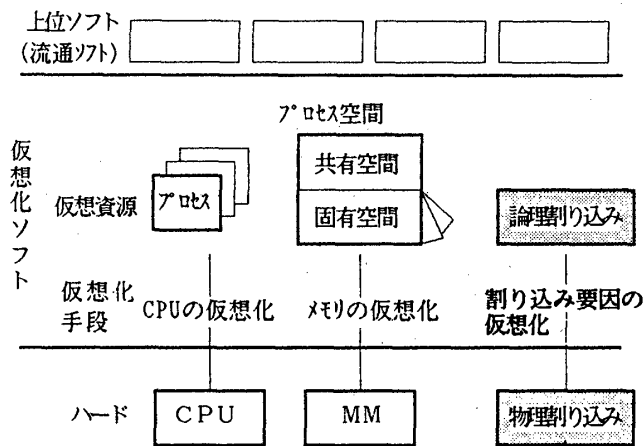


図1. プロセッサアーキテクチャの仮想化

具体的には次の点を狙いとする。

- (1) 直接的にマスク処理を意識する上位ソフト

の流通性向上

- (2) 汎用バス対応の標準的入出力装置パッケージや同一アーキテクチャの系列プロセッサ対応の割り込みハンドラの流通性向上

3. 物理レベル割り込み処理方式の問題点

物理レベルでの割り込み処理方式の問題点を割り込みマスク機能を例に考える。

プロセッサの多くが割り込みのマスク機能を2階層で提供するため、以下2階層の場合について述べる。ここでいう2階層とは、図2に示すように群マスク、個別マスクビットにより制御する場合や、割り込みレベルとデバイス対応のフラグで制御する場合などを包括的に言っている。

下位のマスク機能は、割り込み要因ごとにマスクする機能であることから個別マスク、上位の機能は複数の割り込み要因を一括してマスクする機能であることから群マスクと呼ぶこととする。一般に、割り込み要因を何群に分割しているかはプロセッサごとに異なる。従って、例えば障害割り込みとそれ以外の2群に分割して管理するプロセッサ上で割り込みマスク機能を利用して作成したプログラムを、他プロセッサへ流通させようとした場合、そこでさらに細分化した群マスク機能を提供している場合は、流通は困難となる(図2)。

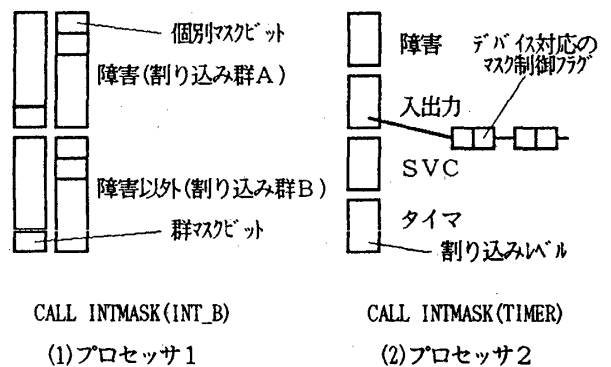


図2. プロセッサのマスク機構の相違とコーディングの相違例

4. 割り込み要因の論理化手法

3章に示したように流通ソフトに見える割り込

み要因の相違がその流通を妨げる原因の一つとなっている。従って、割り込み要因を論理化（共通化）し、流通ソフトにはプロセッサ種別によらず同一の要因で見せることが可能であれば、その流通性の向上が期待できる。以下では、流通ソフトに見せる割り込み要因の番号がプロセッサによらないという意味で、特に論理割り込み要因と呼ぶこととする。論理割り込み要因に対して、物理的なハードウェアの割り込み要因を物理割り込み要因と呼んで区別する。論理割り込み要因は、類似割り込み要因（例えば入出力割り込み）を群（クラス）として管理し、この群単位にマスク制御を行う利用形態を考えると、割り込みクラスとその下で定義される割り込み要因で構成するのが有効である。

割り込みの仮想化ソフトは以下の機能を提供する。

(1) 物理割り込み要因と論理割り込み要因の変換機能

(2) クラスによるマスク制御機能

割り込み要因論理化のイメージを図3に示す。

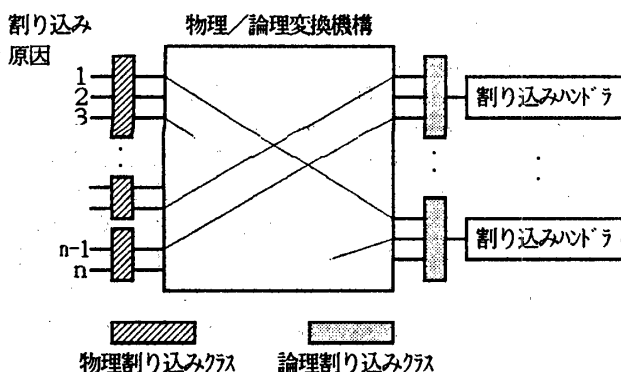


図3 割り込み要因論理化のイメージ

5. 割り込みクラスの共通化（論理化）の可能性

4章で割り込み要因の論理化手法について述べたが、実際のプロセッサへの実現の可能性について考える。

割り込み論理クラスは、具体的には以下の観点で設定した。

- (1) プログラムの走行に伴って発生する同期型割り込み（内部割り込み）とプログラムの走行とは無関係に発生する非同期型割り込み（外部割り込み）はクラスを分離する。
- (2) 割り込みマスク単位として実際の利用形態に適合していること。（同時にマスクする必要のある割り込み要因は同一クラスにすること）

以上の観点から割り込みを分類すると表1の8クラスが適当である。ただし障害については完全にプロセッサ対応となることを考えるとプロセッサによってはさらに細かく分類することが必要である。

なお、表1のその他のクラスについては、特にプロセッサ依存性が強く汎用の7クラスにあてはまらない割り込みを分類する。

実際に数種のプロセッサに適用してみた結果、7クラスでほぼ規定できる見通しを得た。

表1 割り込み標準クラス

クラス	割り込み種別
障害	CPU障害等 (*1)
プログラム	オーバフロー、0除算等 (*2)
デバッグ	トラップ、トレース等 (*2)
SVC	スーパーバイザコール割り込み (*2)
タイマ	インターバルタイマ、周期割り込みタイマ等 (*3)
入出力	DCH割り込み等 (*3)
アロケーション	他プロセッサからの割り込み (*1)
その他	その他特殊な割り込み (*1)

- (注) \*1 アロケーションキック、システム構成依存の割り込み
- \*2 アロケーションキック依存の割り込み
- \*3 汎用的な割り込み

6. 効果

論理割り込み要因の概念を導入し、割り込み要因の仮想化を行うことで次のような効果が期待できる。

- (1) ハードウェアの提供するマスク機構を意識しない汎用的なプログラム設計ができる
- (2) 系列プロセッサ間での割り込みハンドラの共通化

7. おわりに

プログラムの流通性向上のため、プロセッサ間で異なる割り込み要因を統一的な論理要因で記述する方式について述べた。今回は標準の割り込みクラスを設定することによる流通性の向上について述べたが、個別要因の標準化を特に行わないことによる影響について今後検討する。また、論理化に伴うオーバヘッドの定量的な評価についても検討する予定である。

[参考文献]

- 1) 久保田：“CHILL並列処理機能を用いた交換プログラムの実行制御手法”，昭和58年度電子通信学会全国大会 1983
- 2) 榎木：“マイクロプロセッサ用オペレーティングシステムインタフェースMOSI (IEEE 855)とITRONの国際標準化について”，計算機アーキテクチャ研究会報告No. 61 1986