

# Operating System

(CC-BY 4.0) 2024 Taniii.com

2024-08-03

---

01-4-1.pdf

## 1. プロセッサの実行モードは 特権モード と 非特権モード の 2 つに分類できる

## 2. プロセッサの特権モードが必要な理由を説明せよ

プロセッサの特権モードが必要な理由は以下の通りです：

- **システムの安定性確保**：特権モードではシステム全体の制御を行うための特権命令が実行可能であり、これによりシステムの安定性を維持します。一般のユーザプログラムがこれらの命令を実行すると、システムが不安定になる可能性があるため、特権モードでのみ実行できるように制限しています。
- **資源の保護**：特権モードを使用することで、重要なシステム資源（メモリ、CPU など）や他のプログラムのデータを保護することができます。特権命令により、これらの資源の管理と保護を行います。
- **マルチプログラミングの実現**：特権モードを利用することで、複数のプログラムを安全に並行実行させることができます。これにより、効率的な資源利用と高いシステムパフォーマンスを実現します。

## 3. 実行しているプログラムとは非同期に発生する割り込みを 外部割り込みと呼び、入出力 割り込みや タイマ 割り込みなどがある。

## 4. 実行しているプログラムの状態を契機に発生する割り込みを 内部割り込みと呼び、カーネル呼出し 割り込みや アドレス変換例外 割り込み などがある。

## 5. 割り込みが発生した際に、呼び出される特別なプログラムの事を カーネル と呼ぶ。

## 6. カーネルが動作するときのプロセッサの実行モードは 特権モードであり、割り込みは 禁止モードとなる。

## 7. 割り込みが発生した際にプロセッサのレジスタの値を退避しないとどのような問題が生じると考えられるか

プロセッサのレジスタの値を退避しないと、以下のような問題が生じます：

- **データの破壊**：割り込み処理中にレジスタの値が変更されるため、割り込み前のプログラムに戻った際に正しい動作ができなくなります。これにより、データの破壊や計算結果の誤りが発生します。
- **プログラムの異常終了**：レジスタの値が正しく保持されないと、プログラムの実行が不正確となり、異常終了やクラッシュが発生する可能性があります。
- **システムの不安定化**：レジスタの値が退避されないことで、複数のプログラムが並行して実行される環境においてシステムの不安定化を招きます。これにより、全体のシステムパフォーマンスや信頼性が低下します。

## 8. マルチプログラミングシステムと割り込みの関係を説明せよ

マルチプログラミングシステムと割り込みは以下のような関係があります：

- **割り込みによる効率的な資源利用**：マルチプログラミングシステムでは、複数のプログラムが並行して実行されます。割り込み機構を利用することで、あるプログラムが入出力待ちなどで停止している間に他のプログラムを実行することができ、資源の利用効率が向上します。
- **割り込みによる制御の切り替え**：割り込みが発生すると、プロセッサは現在実行中のプログラムを一時停止し、割り込み処理を行います。この際、カーネルが動作し、適切な処理が行われた後、元のプログラムまたは他のプログラムに制御を戻します。これにより、スムーズなプログラムの切り替えが実現されます。
- **割り込みによる応答性の向上**：外部割り込み（入出力割り込みやタイマ割り込みなど）により、システムが外部イベントに迅速に対応できるようになります。これにより、システム全体の応答性が向上し、ユーザや他のプログラムからの要求に迅速に対応することが可能となります。

## 9. アプリケーションがシステムコール関数を呼び出す際のカーネルの処理の流れをまとめよ

アプリケーションがシステムコール関数を呼び出す際のカーネルの処理の流れは以下の通りです：

1. アプリケーションがシステムコール関数（例：`read()`）を呼び出す。
2. システムコール関数内で、カーネル呼び出し命令が実行される。
3. カーネル呼び出し割り込みが発生し、割り込みコードが判定される。
4. 割り込みコードに応じて、カーネル内の対応する処理部が選択される（例：入出力処理）。
5. 選択された処理部が、入出力装置への指令を発行し、装置が動作を開始する。
6. カーネルは次に実行するプロセスを選択し、プロセススケジューリングを行う（例：プログラム B を選択）。
7. 選択されたプロセス（例：プログラム B）の実行が開始される。
8. 入出力装置が処理を終了し、割り込みを発生させる。
9. 入出力割り込みが発生し、カーネルが入出力処理を行う。
10. 再び次に実行するプロセスを選択し、プロセスの実行が開始される（例：元のプログラム A に戻る）。

## 10. カーネルに含まれる処理にはどのようなものがあるか挙げよ

カーネルに含まれる処理には以下のものがあります：

- **プロセスの実行管理**：プロセスの生成、終了、スケジューリングなどを行う。
- **入出力装置の制御**：入出力装置へのデータの入出力や制御を行う。
- **ファイル管理**：ファイルシステムの管理やファイル操作を行う。
- **メモリ管理**：メモリの割り当てや解放、仮想メモリの管理を行う。
- **仮想メモリの制御**：仮想メモリの管理やページング、スワッピングを行う。
- **プロセスの同期制御**：プロセス間の同期や通信を行う。
- **タイマの管理**：タイマの設定や管理を行う。
- **割り込み処理**：割り込みの処理や割り込みハンドラの実行を行う。

## 11. \*割り込みが存在しないシステムで TSS を実行するにはどのような手法が考えられるか

割り込みが存在しないシステムで TSS（Time-Sharing System）を実行するためには、以下のような手法が考えられます：

- **ポーリング**：割り込みの代わりに、プロセッサが定期的に各入出力装置の状態をチェックする。これにより、入出力の完了や他のイベントを検出する。ただし、効率が悪くなる可能性がある。

- **タイムスライスの強制終了**：各プログラムに一定のタイムスライスを与え、そのタイムスライスが終了すると強制的に次のプログラムに制御を移す。タイマーを利用して時間を測定し、タイムスライスの終了を検出する。
- **コールバック機能の利用**：入出力操作が完了した際に、入出力装置が特定のコールバック関数を呼び出す仕組みを導入する。これにより、入出力操作の完了を検出する。

(CC-BY 4.0) 2024 Taniii.com