

リアルタイムOSの基礎

名古屋大学 大学院情報科学研究科
附属組込みシステム研究センター
人材育成プログラム (NEP)

最終更新 : 2015/03/27

リアルタイムシステムとは?

- JISによる「実時間処理」の定義
 - 計算機外部の処理に関係を持ちながら、かつ外部の処理によって定められる時間要件にしたがって、計算機が行なうデータ処理
- リアルタイムシステム研究者の間で一般的な定義
 - 処理結果の正しさが、出力される結果値の正しさに加えて、結果を出す時刻にも依存するようなシステム
- **! 単に速い応答を求められるシステムをリアルタイムシステムと呼ぶわけではない**
- 多くの組込みシステムはリアルタイムシステム
 - 組込みシステムは、機械・機器を制御するシステムであり、制御対象の時間要件にしたがうことが必要
 - 一部の処理のみにリアルタイム性が求められる場合も

予測可能性 (predictability)

- システムを動作させる前に、システムの（時間的な）振舞いが予測できること … リアルタイムシステムが満たすべき要件（必要条件）
 - RTOSの場合には、各機能の最悪実行時間や最悪レスポンス時間が予測できること
- 最近のプロセッサには、キャッシュや分岐予測など、予測を困難にする各種の機構が採用されており、いろいろなレベルの予測可能性を考えざるをえない
 - 100%保証するためには、悲観的な予測値を使うしかない
 - 確率的な予測も研究されている（確立されていない）
 - どういう条件で実行時間が長くなるかがわかっていることが最低条件（ストレステストが可能）

リアルタイムOS (RTOS) とは?

- 文字通り, リアルタイムシステム構築のためのOS
- 具体的には, 次のような特徴を持つOS (これらの特徴をすべて持っているとは限らない)
 - (1) リアルタイムシステム向けの機能を持つ
 - プリエンプティブな優先度ベーススケジューリング
 - 優先度継承や優先度上限プロトコルのサポートなど
 - (2) 予測可能性を持つ
 - OSの各サービス時間があらかじめわかっている
 - ただし, 予測可能性にもいろいろなレベルがある
 - (3) 時間制約を管理 ← 一部の研究ベースのRTOS
 - OSが各処理の時間制約を考慮してスケジューリング
 - (4) 高速に応答 (制御対象に対して十分に)

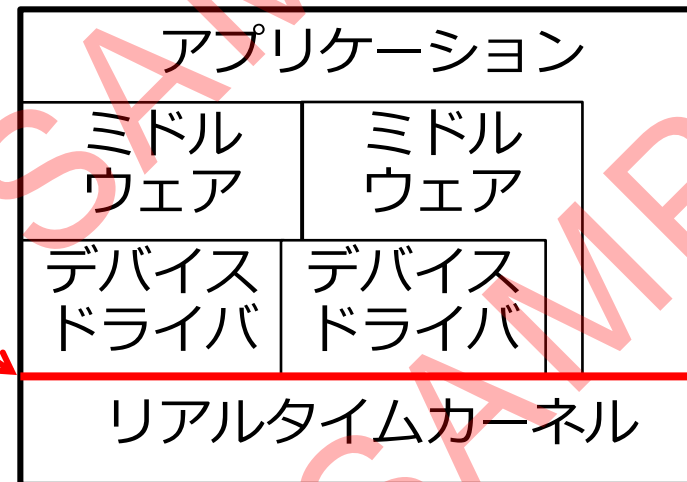
RTOSの構造とリアルタイムカーネル[1/5]

- アーキテクチャによるRTOSの分類（中間的なものもある）

- 汎用OS型



- リアルタイムカーネル型



API

- OSの機能は豊富
- サイズは比較的大きい
- 一般に応答時間は遅い
- デバイスドライバは別のAPIで作成

- カーネルの機能は限定
- カーネルサイズは小さい
- 一般に応答時間は早い
- デバイスドライバとアプリケーションは同じAPI

RTOSの構造とリアルタイムカーネル[2/5]

- リアルタイムカーネル
 - (元々は) RTOSの中心になるモジュールの意味
 - どのようなコンピュータシステムにも共通する資源 (プロセッサ, メモリ, タイマ, ...) を扱う
 - 汎用OSのカーネルとは意味 (定義) がやや異なる
 - 保護機能を持っているとは限らない
 - リアルタイムカーネルは, 構造上/役割上はOSに他ならないが, 汎用OSとは機能的に大きく異なるため, 区別したい場合にはリアルタイムカーネルと呼ぶ
 - リアルタイムカーネル相当の機能しか必要としない場合もある. その場合は, 「リアルタイムカーネル=RTOS」
 - リアルタイムモニタ, リアルタイムエグゼクティブと呼ばれることもある
- ! この講義では, 主にリアルタイムカーネル型を想定

RTOSの構造とリアルタイムカーネル[3/5]

- 確認：オペレーティングシステム（OS）の役割
 - ! どの範囲のソフトウェアをOSと呼ぶかは、明確な定義がない
 - 例：ウィンドウシステムはOSの一部か？
 - ハードウェアの仮想化
 - ハードウェアとしてのコンピュータの機能を拡張し使いやすく柔軟なコンピュータを提供すること
 - そのために、ハードウェアを抽象化・多重化
 - ハードウェア資源の管理機能
 - アプリケーションにハードウェア資源の共同利用を許し、資源が効率よく使用されるように管理すること
- 例) ファイルシステム
- ストレージデバイスというハードウェア資源を抽象化・多重化するとともに、複数のアプリケーションからのアクセスを許す

RTOSの構造とリアルタイムカーネル[4/5]

- リアルタイムカーネル型が出てきた理由
 - OSが必ずサポートしなければならないI/O装置はない
 - 多くの組込みシステムが共通に持つI/O装置が無い
 - 例) ストレージデバイスを持たない組込みシステムも多い
 - 複数のアプリケーションで同一のI/O装置を共有する状況は少ない
 - I/O装置の制御は組込みシステムの目的そのもの
- 保護のための機能は必須ではない
 - 組込みソフトウェアは、**組込み対象の機器を制御することのみ**を目的に設計され、機器に固定されている
 - デバッグが終われば、アプリケーションソフトウェアは信頼できるという前提が成り立つ
- ! 最近、汎用システムと近い性質を持つ組込みシステムも多くなってきた（典型例は携帯電話やカーナビ）