

マイコン・プログラマのための  
Linux 超入門

第3回

高性能処理のキモ！プログラム実行順序決定のメカニズム

宗像 尚郎

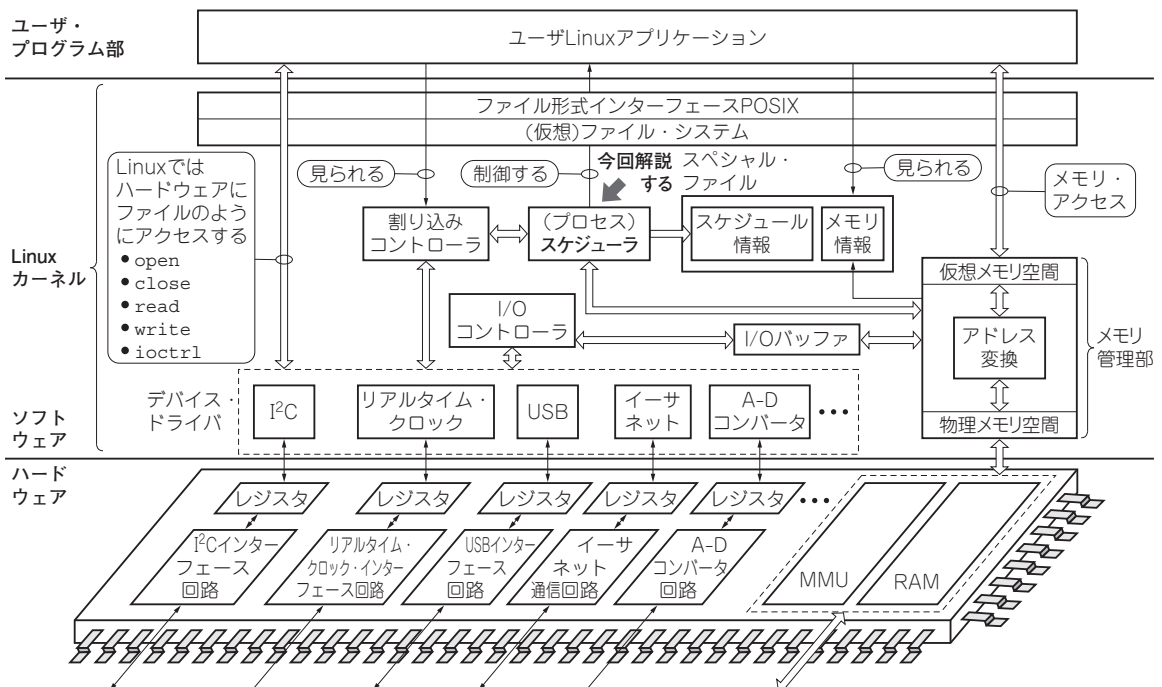


図1 今回解説すること…Linuxカーネルが大事な処理を優先して行うためのプログラム実行順序決定のスケジューラのメカニズム

今回は、Linuxカーネルが大事な処理を優先しつつ高性能処理を行うためのキー・テクノロジーであるプログラム実行順序決定(タスク・スケジューリング)のメカニズムについて解説します(図1)。

(編集部)

## 今回の内容

## ●OSが複数プログラムを並行して動かすために…時分割で実行する

Linuxカーネルは各アプリケーションに対して時分割(タイム・スライス)で順番にCPU資源を割り当てることで、複数のアプリケーションを同時実行しているように見せています。このしくみをプリエンティブ・マルチタスク・スケジューリング機構といいます。

一つの負荷の重いプログラムがCPU資源を長時間握ってしまっても他の処理が実行できなくなることを避けられます。組み込み機器で使われてきたリアルタイムOSであるITRONもタイム・スライスによるタスク切り替えを行います。

## ●Linuxの時分割のしくみはかなりインテリ

ITRONのタスク切り替え間隔は固定時間です。次に実行するタスクの選択は静的優先度ベースのラウンドロビンなどの直感的に挙動が予測しやすい単純なポリシーが採用されていました。

一方Linuxは本格的なマルチユーザ、マルチタスクのサーバOSとして進歩してきたので、マルチプロセス環境の最適利用なども考慮された高度なタスク切り替えのアルゴリズムが導入されています。