

メモリやI/Oなどの資源をリアルタイムOSで活用して管理する方法

杉本 明加

組み込み機器はコストを低減したいという要求から、使用できる資源に制限がある。限りある資源をどう管理して機能を実現するかが組み込み開発のポイント一つである。本章では、メモリやI/Oなどの資源をリアルタイムOSで管理する方法を解説する。 (筆者)

本章ではリアルタイムOS (RTOS) の活用方法の最後として、資源管理を取りあげます。資源 (リソースとも呼ぶ) という言葉は非常にあいまいですが、ここではメモリやI/Oなどを指します。

資源管理のミスはしばしば致命的な問題を引き起こします。通常、資源は有限なものであり、「資源がどれくらい使われてどれだけ残っているか」、「誰が使っているか」といった管理を適切に行う必要があります。

典型的な例はメモリで、プログラミングの経験者であればmallocやnew演算子によるメモリ確保を経験したことがあるでしょう。そして、確保したメモリを開放し忘れてメモリ・リークが発生したり、使用済みのメモリにアクセスしてエラーを起こしてしまったことがあると思います (もちろん筆者にも経験がある)。

RTOSの多くは資源を管理するための機能を提供しており、TOPPERS/ASPも資源管理向けの機能を直接的、および間接的な方法で用意しています。

1. プログラミングにおける資源

資源管理の方法を紹介する前に、「資源」という言葉の指すところをもう少し考えておくことにします。ここまでの章でOSが扱うものとして出てきた言葉 (タスク、通信、排他など) はそれなりにイメージしやすいものだったと思いますが、資源という言葉はいまいち具体性に欠けます。

一般的な定義を、フリーの百科事典であるWikipediaで調べてみましょう。「計算資源」の説明を見てみると、

『計算機 (具体的なコンピュータ、そこで動くプロ

セスやジョブ、あるいは抽象的な計算模型) が計算のために使う「資源」である。計算リソースとも訳す。計算機資源、コンピュータ資源、コンピュータリソース、計算機リソース (computer resource) とも。単に資源、リソース (resource) とも。』

となっており、あいまいですね。さらに読み進めると、

『計算時間 (CPU 時間) とメモリ使用量が代表的な資源である。そのほか、2次記憶、入出力装置など、情報処理のためのあらゆる機器が資源となる。物理機器に限らず、ファイルやネットワーク接続、メモリ空間なども仮想資源である。』

と記載があり、プログラムが使用するものは基本的に資源と呼べるのが分かります。実際のところこの定義で間違いはないのですが、話をしやすくするためにもう少し具体的に資源の種類を挙げることにします。

- CPU
- メモリ (RAM, ROM)
- 入出力装置 (マウス、キーボード、タッチ・パネルなど)
- 2次記憶 (フラッシュROMやメモリ・カードなど)
- 通信装置 (LAN, USB など)
- 画像・音声装置 (LCD, スピーカ, カメラなど)

思いつくところではこれくらいでしょうか (もちろん資源と数えるものはもっと無数にある)。

2. 資源の管理方法

次に、どのような資源管理が必要か考えてみましょう。管理方法は資源のタイプによって異なりますが、ここでは