

## ■ハイブリッドOS (HybridOS)

### ●いいとこどり

「ハイブリッド」という言葉が一般的に使われるようになった代表としては「ハイブリッドカー」(Hybrid Car)が挙げられます。

電気で動作する「モーター」とガソリンなどの燃料を使った「エンジン」(内燃機関)の複数の動力源を使い分けるなど、複数の技術を応用した車のことです。

エンジンとモータを駆動源として切り替えるタイプとエンジンを単なるモータ駆動のための発電機として使うタイプに大きく分けられます。

前者を「パラレル方式」、後者を「シリーズ方式」と呼びます。

ハイブリッドカーはエンジンと電気モータの「いいとこどり」をした車ということができます。

同じように開発効率および移植性の高さリアルタイム性能の追求という二つの「いいとこどり」を狙った技術が「ハイブリッドOS」です。

### ●ハイブリッド化の意義

ハイブリッドOSはひとつのプロセッサで $\mu$ ITRONとLinuxを共存させたり、 $\mu$ ITRONとWindowsCEを共存させるというようなことが行われます。

つまり、リアルタイム性能が高いとされる、 $\mu$ ITRONと他のOSとの組み合わせで語られることが多いようです。

この場合のLinuxまたはWindowsCEなどのOSをここでは便宜的に「ゲストOS」と呼びましょう。

このようなことを行う視点は二つあります。

#### (1) $\mu$ ITRON資産重視

これは $\mu$ ITRONで開発した資産をそのまま使い、 $\mu$ ITRON上で対応していないプロトコルスタックなどの追加開発を行うのが大変という場合。

この大変な開発がゲストOSでは既に解決済みであることが重要です。TCP/IP関連のプロトコルスタック、ファイルシステムとの連携などについて目を向けられることが多くなります。

#### (2) リアルタイム性能補完重視

ゲストOSでの豊富な開発資産があるが、製品性能的にリアルタイム性能が要求されるイベント処理を追加実装したい場合。

### ●ハイブリッド化が必要なシステム

(1)の視点でハイブリッド化を行う場合、既にリアルタイム性能が要求されるデバイス制御を伴った実装が行われているということが重要です。つまり、ベースシステムがそれほどリアルタイム性能を追求していない $\mu$ ITRONシステムなのであれば、ハイブリッド化するまでもなく、ゲストOS環境に乗り換えれば済む話です。

#### (2)の視点でも同様の側面があります

リアルタイム性能を引き上げたいというのは既存の機能の性能改善という意味ではなく、独立したハードウェアイベントの処理について語る必要があるというところが重要です。

つまり、例えばTCP/IPの通信性能を上げたいとか、WWWサーバの応答性を上げたいという意味のリアルタイム性能補完ではなく、特定の追加デバイスの割り込み応答性能がLinuxのデバイスドライバでは満たせない、というような場合にのみ有効だということです。

そのような特殊デバイスや特殊イベント処理の対応予定もないのにハイブリッドOSを使う必要はありません。

### ●ハイブリッドOSの実装の基本は「パラレル方式」

以上のようにハイブリッドOSの利用目的のポイントを考えるとハイブリッドカーでの「シリーズ方式」は選択肢からは外れます。 $\mu$ ITRON上でイベント検知してその応答処理をLinux側のプロセスで処理するというような構成はあまり意味がないからです。

リアルタイム性能が要求されるデバイスに対する処理は全て $\mu$ ITRONで完結させて、WWWサーバ経由での表示処理などの少しくらい遅れても構わない処理のためにその情報を $\mu$ ITRONからLinuxのプロセスに通知するというような使い方になります。このような目的を踏まえてハイブリッドOSには主に以下の機能が実装されます。

- (a) ゲストOS起動機能
- (b) 割り込み調停機能
- (c) OS間通信機能

割り込み調停機能部分でデバイス単位の割り込み処理対象OSを確定するようにパラレル制御します。