

Win32 APIを利用して開発できる組み込みOS

# Windows Embedded CE用 USBモデム・ドライバの製作(前編)

江島 午郎  
清水 浩行

2006年11月、Microsoft社は組み込みシステム向けのOSである「Windows Embedded CE 6.0」を正式にリリースした。ここではUSBモデム・ドライバを例に、Windows Embedded CEを使ったデバイス・ドライバ開発の実際を紹介する。前編の今回は、USBクラス仕様やドライバの作成方法などについて説明する。

(編集部)

本稿では、Microsoft社のOSである「Windows Embedded CE(以下、Windows CE)」で利用できるUSBモデム・ドライバの製作事例を紹介します。

## 1. リアルタイム OS “Windows CE”

Windows CEはPDA(携帯型情報端末)のためのOSと誤解されがちですが、そうではありません。Windows CEは、本来は組み込みシステムのために設計されたリアルタイムOSです。

Windows CEのもつリアルタイム性

「リアルタイムOSとは何か」という定義については、多くの文献や記事などで述べられています。それらによると、リアルタイムOSの要件はその名の通り、まず「リアルタイム性」であるとされています。この「リアルタイム性」という言葉を「素早く応答する」という意味だと思っている人が多いようですが、正確には、「入力となる事象に対する応答が実時間で、その最大時間が設計段階において予測できる」という意味です。

リアルタイム性の実現に必要なOSの機能は、

- イベント・ドリブン
- プリエンプティブ・マルチタスク
- 固定優先度レベルに基づくスケジューリングと十分な優先度レベル

などです。

これらの要件で考えると、例えば「Windows NT/2000/XP(以下、Windows NT系)」やそれらの組み込み版と位置付けられる「Embeddedシリーズ」、最近、組み込みシステムでも注目されている「Linux」を始めとするUNIX系OSなどはリアルタイムOSではありません。これに対してLynuxWorks社の「LynxOS」やQNX Software Systems

社の「QNX Neutrino」、RadiSys社の「Microware OS-9」、TenAsys社の「iRMX」、Wind River Systems社の「VxWorks」、そして国産OSである「μITRON」などはリアルタイムOSです。

もちろんWindows CEもリアルタイムOSであり、イベント・ドリブンでプリエンプティブ・マルチタスクをサポートしています。また、マルチプロセス/マルチスレッド・モデルですが、スケジューリングの基本単位はスレッドです。個々のスレッドは256段階の固定優先度レベルを持ち、その優先度レベルに基づいてスレッドのスケジューリングが行われます。

しかし、Windows CEはリアルタイムOSの仲間として紹介されることが少ないようです。それには、いくつかの理由があります。例えば、Windows CEがPDA向けのOSであると誤解されていたり、Windows NT系と同じようなOSであると思われる、といったこともあるでしょう。

そしてこれは筆者個人の意見ですが、「リアルタイム性」という言葉に対して、組み込みエンジニアの間にある種の独特なニュアンスが付け加わっているためではないかと考えています。必要以上に応答性能にこだわり、「必要にして十分とは何か」ということを追求する姿勢が、忘れられがちのように思います。これは、ITRON登場以前に、リアルタイムOSが「リアルタイム・モニタ」と呼ばれていたころから組み込みシステムの開発に携わってきた筆者の率直な感想です。

リアルタイム・システムには、ご存じの通り「ハード・リアルタイム」と「ソフト・リアルタイム」の二つのシステムがあります。ハード・リアルタイム・システムと呼ばれるものは、例えば航空機の操縦ピットや自動車のブレーキ・システムなど、特定のイベントに対してアプリケーションが100%の確率で指定された時間内に応答しなければ



ばならないようなシステムであると定義できます。一般に、この時間は数 $\mu$ s(マイクロ秒)とされています。用途としては、人命にかかわるシステムなど、意外と限られたもので、それ以外の大多数の組み込みシステムはソフト・リアルタイム・システムに分類されます。

Windows CEは、先に定義したハード・リアルタイムの要件こそ満たしていませんが、実際にカー・ナビゲーション・システムをはじめとする多くの機器に使用されています。すなわち、ソフト・リアルタイム・システムには必要にして十分なリアルタイムOSです(筆者の意見とは異なり、Windows CEをハード・リアルタイムのOSと位置付ける考え方もある。これは、リアルタイム・システムの定義の違いによるものと考えている)。

ブラウザやメディア・プレーヤが用意されている

Windows CEの特徴は、幅広い機器に対応できるスケラビリティを備えていることでしょう。例えばGUI(Graphical User Interface)を持たないヘッドレス・システム(ゲートウェイのようにディスプレイやキーボード、マウスなどを備えないシステム)にも、カー・ナビゲーション・システムのように本格的なGUIを備えたシステムにも対応可能です。Webブラウザやメディア・プレーヤなど、最初から豊富なコンポーネントが用意されているのも、魅力の一つです。

組み込みシステム用OSでは、ヘッドレス・システムを構成したときのフットプリント(プログラムが必要とする記憶領域の大きさ)が重要になります。Windows CEの場合、最低限の機能に絞れば、500Kバイトのフットプリントを実現できます。応答性能については、筆者の認識する範囲では、最悪値で100 $\mu$ s ~ 500 $\mu$ s程度です。

また、リアルタイムOSとして必須のタスク/スレッド間同期機構として、イベントやセマフォ、ミューテックス、クリティカル・セクションなどに対応しています。これらの機能は、アプリケーション・プログラミング・インターフェースの標準であるWin32 APIとして実装されています。その使い方は、広く一般に利用されているWindows NT系と高い互換性を持っています。

昨今は組み込みシステム開発にもグローバル化の波が押し寄せてきています。日本国内に限定すればITRON OSでも問題ありませんが、Win32 APIであれば世界中のどの国の開発者にも、ほとんど説明なしで通じます。つまり、

Windows CEは、開発者の集めやすさや再利用性の面でも、有利であると言えます。

## 2. USBドライバの構成

組み込みシステムのOSとしてWindows CEの導入を検討する場合、Windows NT系のデバイス・ドライバを想像して、「複雑で大変なのではないか」と警戒する方も多いようです。しかし、Windows CEのデバイス・ドライバはWDM(Windows Driver Model)ではなく、簡素でシンプルなものになっています。

4個以下のエントリ関数を書くだけ

Windows CEには多種多様なデバイス・ドライバが用意されていますが、最も基本的なデバイス・ドライバの構造は「ストリーム・インターフェース・ドライバ」と呼ばれています。

例えば、LEDを点灯させる単純なLEDドライバを考えてみましょう。仮にそのソース・ファイルの名前をled.cとします。ストリーム・インターフェース・ドライバに最低限必要な10個のエントリ関数(表1)を準備する必要があります。それぞれの関数名の最初のxxxの部分は、3文字で表現したデバイス・プリフィックスと呼ばれるデバイス名で、任意の名前を付けることができます。例えばLEDドライバのデバイス・プリフィックスを“LED”とすると、エントリ関数名はLED\_Init, LED\_Deinit, LED\_Open,...となります。10個のエントリのうち2個は電源管理に関するもので、4個は初期化に関する形式的なものです。実質的に書く必要のあるエントリ関数は4個以下です。

Windows CEのデバイス・ドライバは、DLL(Dynamic

表1 ストリーム・インターフェース・ドライバのエントリ関数

エントリ関数	説明
XXX_Init	デバイスを初期化する
XXX_Deinit	デバイスを終了する
XXX_Open	デバイスを開く
XXX_Close	デバイスを閉じる
XXX_Read	デバイスからデータを読む
XXX_Write	デバイスへデータを書く
XXX_IOControl	デバイスへコマンドを送る
XXX_Seek	デバイスのデータ・ポインタを移動する
XXX_PowerDown	デバイスへの電源をサスペンドする
XXX_PowerUp	デバイスへ電源を再投入する