

第7章

DOS上で16ビットPCカードと32ビットCardBusカードを制御

CardBusブリッジの制御方法とPCカード・イネーブラの作成技法

大中 邦彦／熊谷 あき

16ビットPCICはISAバス・ベースでもあるため、今後は使われることはないだろうと思われる。しかし、32ビットPCICであるCardBusブリッジはPCIバス・ベースなので、組み込み機器で使われる場面が増えると思われる。またCardBusブリッジは16ビットPCICの機能も含んでいる。そこで、ここではCardBusブリッジに焦点を絞り、PCカード・イネーブラの具体的な作成事例について詳しく解説する。 (編集部)

● PCカード・サポート・ソフトウェアの形態

PCカードをサポートするソフトウェアの形態としては、大きく分けて重装備型と軽装備型という二つの考え方があります(図1)。

重装備型の代表は、Windowsなどのプラグ&プレイ対応OS上のPCカード・システムでしょう。PCカードが差し込まれると自動的に何のカードであるかを判定し、PCカードが必要とするリソースを調べてその環境で既に使用されているリソースとぶつからないようにアドレスなどを割り当てます。そして、PCカードに対応するドライバ・ソフトウェアを起動します。

もし、システム内にドライバが存在しない場合には、ドライバのインストールを促します。PCカードが抜き取られれば、ドライバ・ソフトウェアの動作を停止

させ、PCカードに割り当てたリソースを解放し、次にPCカードが差し込まれるのを待ちます。

このように、あらゆるPCカードに対応する本格的なプラグ&プレイを実現するためには、PCカードを正しく判別できる必要があります。しかし、PCカードの発展途上において、正しくないタプルが記述されたPCカードが存在することも事実です。また、CISに記述されている内容だけではPCカードの種類を特定できない場合もあります。そこで、Windowsなどでは、CISに記録されている製品名などの名前からPCカードを識別するためのデータベースを持っています。

もう一方の軽装備型としては、イネーブラと呼ばれるソフトウェアがあります。一般的にイネーブラは、

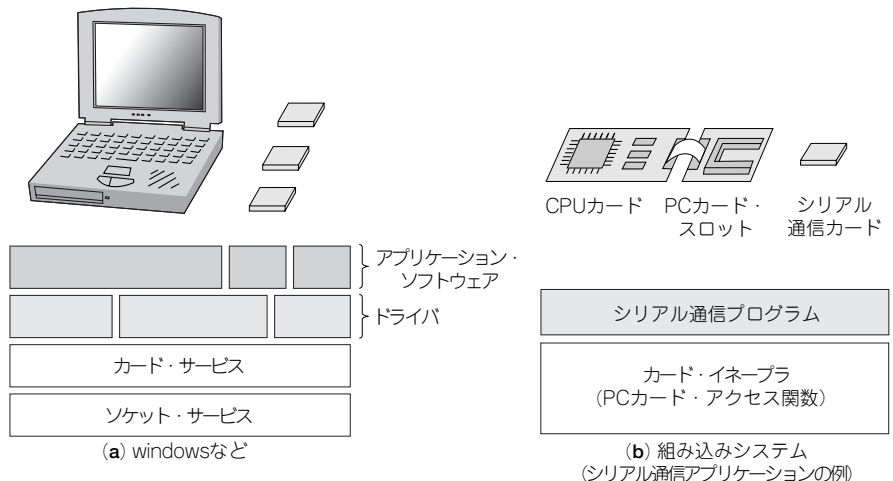


図1 PCカード・サポート・ソフトウェアの形態

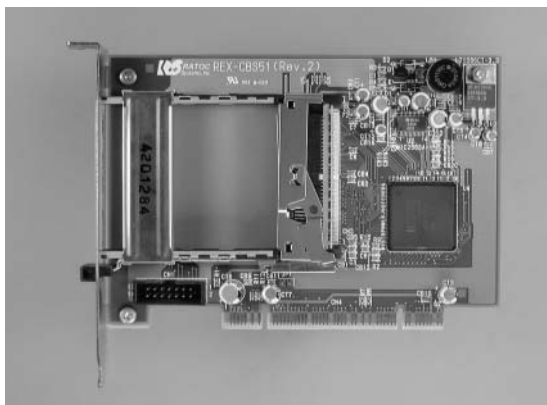


写真1 R5C478II搭載PCIバス増設タイプCardBusブリッジ・ボード

目的とするPCカードが既にソケットに挿入されている状態で実行され、カードのコンフィグレーションを行って適切にPCカードのリソースをシステムにマッピングしたのち、そのカード用のドライバやソフトウェアを実行するという形態をとります。つまり、PCカードの初期化ルーチンといえるでしょう。

イネーブラは、PCカードの必要とするリソースを割り当てるのが仕事で、差し込まれたPCカードのメーカー名や製品名を識別する必要は特にありません。

パソコンはあらゆる意味で汎用ですから、やりたいことに合わせてそのつどPCカードを差し込み、ソフトウェアを起動します。しかし、ここで想定している組み込み用途では、システムにモデム通信機能をもたせたり、LANに対応させるためといったように、初めから用途を想定して使用します。PCカードのデータベースをもったり、複雑に階層化されたドライバ構造を採用する必要はないわけです。

● 対象OSとハードウェア

OSとしてWindowsやLinuxを想定すると、それぞれのOSごとのデバイス・ドライバの知識などが必要になります。ここではCardBusブリッジの制御の具体的な事例を解説するのが主眼なので、C言語が分かればサンプル・プログラムが読めることが望めます。そこで、ここではDOS環境で動作するPCカード・イネーブラを作成します。使用するプログラミング言語にはDOS版のLSI C-86試食版を使用しました。

また、標準PCI-CardBusブリッジに準拠したすべてのCardBusブリッジに基本的には対応しています。リコー社製のCardBusブリッジではほぼすべてが問題なく動作しますが、Texas Instruments社(以降TI社)

製のCardBusブリッジでは、一部動作しないものがあります。

写真1にリコー社製CardBusブリッジR5C478IIを搭載したPCIバス増設タイプのCardBusブリッジ・ボードを示します。これ以外にも、リコー社製CardBusブリッジを搭載したノート・パソコンでも動作確認をしています。

1 プラグ&プレイを実現するには

● システムで使用中のリソースを把握する

プラグ&プレイを実現するには、今現在そのシステムでどのI/Oやメモリ、割り込みが使われているかを常時把握しておき、新たにリソースを割り当てる場合は使用中のリソースと衝突しないように未使用の領域を探して、その空間で動作できるようにコンフィグレーションする必要があります。

Windows上であればプラグ&プレイに完全対応しているので、OSがすべてのリソースを把握し管理しています。しかしDOSはプラグ&プレイではないため、OSに問い合わせてもリソース情報は得られません。

実はDOS環境では、BIOSと連携することで、リソース情報を取得します。486やPentium以降のPC/AT互換機にはプラグ&プレイBIOS(以降PnPBIOS)が搭載されており、このPnPBIOSがシステム・リソースを管理しています。

しかし、筆者はPC/AT互換機のBIOSには詳しくないので、PnPBIOSを直接呼び出してリソース情報を取得するプログラムを書くことができません。そこで、参考文献(3)でも紹介されているプラグ&プレイBIOS情報表示ツール(PnP.COM)を活用してみました。

● プラグ&プレイBIOS情報表示ツール

PnP.COMとは

プラグ&プレイBIOS情報表示ツールPnP.COMとは、PnPBIOSを呼び出してシステムで使用中のリソース情報を一覧表示するツールです。このツールの使い方そのものについては、参考文献(3)を参照してください。

リスト1にUSEDLISTコマンドで取得したリソース情報表示の様子を示します。上から、DMAチャネルの使用状況、IRQ割り当て、そして64Kバイト分のI/Oポートと、4Gバイト分のメモリ空間の使用状況です。表示中でたとえば「16550A Compatible COM Port」と表示のあるリソースは、シリアル・ポートに使われて