

組み込み機器に USBホスト機能を実装するには

桑野 雅彦

USBは標準インターフェース

USBはPCにおける標準インターフェースの地位を確立したといっても過言ではない状況になっています。キーボードやマウスを筆頭に、外付けHDDやCD/DVD-ROMドライブ、デジタルカメラのメモリ・カード・リーダー、プリンタ、各種ネットワーク・アダプタ、最近ではデュアル・ディスプレイ用のVGAアダプタまで登場しています。もはやたいの用途の周辺機器は、USBだけで接続できるようになっています。

このように、豊富にそろった周辺機器を、PCだけに使わせておくのはもったいないと考えるのも自然の成り行きです。こうして組み込み機器にも、さまざまなUSB周辺機器を接続したいという要求が強くなってきました。

USBの設計思想

図1にUSBのシステム構成を示します。たとえるならUSBの周辺機器というのは、上司(ホスト)からの指示がなければ動かず、言われたことしかやらない、怠け者の部下のようなものです。上司はそれぞれの部下がどのような仕事をしているかを考えて各ターゲットのエンドポイントをアクセスするスケジュールを立てて処理しなくてはならないというわけです。

ホストの側はそれぞれのターゲット機器へのコマンド送出手続きや状態チェック、データの読み取りなどのスケジュールリング、着脱のチェックなどをすべて行わなくてはならないので非常にたいへんなのですが、PCなどホストのCPUは十分に高い性能をもっていますし、コントローラ・チップのゲート数もマザーボード上のコントローラの総ゲート数からすれば大したことはなく、コスト上の問題はありません。

逆にターゲットの側はハードウェア/ソフトウェアともかなり単純な構造になるので、安価な機器にも使用できるということになります。おおざっぱな計算ですが、ホスト・コントローラは1万ゲート程度、ターゲット側は1500ゲート程度で実現できるということです。

このように、ホストとターゲットをあえて非対称とすることで、ターゲット側を極力単純で低価格なものにしながら、さまざまな周辺機器を一つのバスに接続するという目標を達成したのがUSBであると言えるでしょう。実際にキーボードやマウスといった、低価格な周辺機器にもUSBインターフェースを搭載することができたのは、この設計思想のおかげと言えます。

USB標準クラスとは

Windowsでは、USBキーボードやUSB接続の外付けHDDをつないでも、外部からドライブを読み込むことなく、OSが標準でもっているドライブが組み込まれて、すぐにこれらを使うことができます。USBではあらかじめ一般的な周辺機器がカテゴリ化され、その制御方法が統一されているためです。これを標準クラスと呼びます。

組み込み機器にUSBでキーボードやHDDを接続したいという場合は、USBホスト・コントローラやそのデバイス・ドライバを実装すればよいだけではありません。一般的には図2のような構造で各種ドライバやソフトウェアが必要です。実際にはUSBホスト・コントローラにもいくつかの種類があるので、ドライバの移植性を考慮すると、USBホスト・コントローラのデバイス・ドライバも、複数の階層に分けられます。

本特集の第1部では、USB周辺機器としてもっとも一般的に使われているキーボードやマウスなどHID(ヒューマン・インターフェース・デバイス)と、メモリ・カード・リーダーやHDDなどのストレージ・デバイス、そしてポートを増設するときに必須となるUSBハブについて解説します。

USBホストの実装形態いろいろ

USBホストと言っても、その実装形態にはいくつかの種類があります。

まず一つは、PCをそのまま組み込み機器に置き換えたイメージで、常時USBホストとして動作する組み込み機器です。もう一つは、普段はUSBターゲット機器でありながら、必要に応じてUSBホストに化けるUSB機器です(USB On-The-Go仕様対応機器)。本特集では、前者の常時ホストとして動作する組み込み機器の設計事例について重点を置いて解説します。

USB On-The-Go仕様対応機器の設計事例については、参考

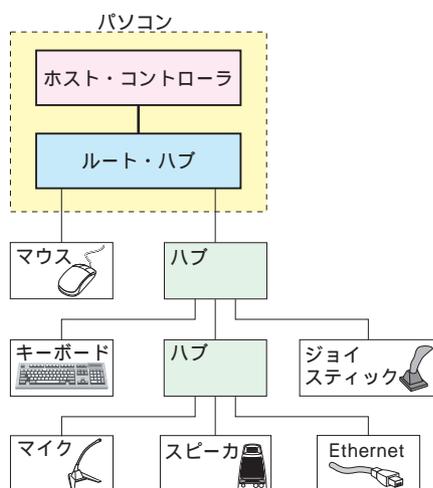


図1 USBの接続トポロジ



図2
ホスト機能の実現にはUSBホスト・コントローラ/ドライバと各USBクラス・ドライバが必要

文献(3)などを参照してください。

USBホスト機能実現のために要求されるリソース

すでに説明したUSBの設計思想は、比較的リッチなリソースを持つことができるPCをホストとした場合には合理的です。しかし、PCと比較してメモリ容量やCPU処理能力が劣る組み込み機器では、ホスト機能の実装はかなりの負担となります。これがUSBホスト機能を組み込み機器に実装する場合の難しいところです。

そのため、Mバイト単位のメモリを載せ、動作周波数が100MHz以上の32ビットRISCマイコンを搭載した組み込みシステムなど、一昔前のPCレベルのリソースを用意するか、またはある程度仕様を限定し、USBハブ未対応としたり、接続可能なUSB周辺機器も一部に特定して運用するなどします。

もっとも後者の場合であっても、組み込み機器はもともと用途を限定したシステムであるため、あまり問題となることはありません。接続可能な周辺機器がキーボードだけ、あるいはHDDだけとなっても、まったく接続できなかった場合と比較すれば、使い勝手が大きく違います。

本特集では、第2部でH8に組み込み向けUSBホスト・コントローラを接続したCPUカードを使い、USBホスト機能を実装する事例を、第3部ではより本格的なUSBホスト・コントローラの制御事例として、UHCI(Universal Host Controller Interface)の制御プログラムの作成事例を紹介します。

USBのデバイス・ステート管理

第2部や第3部で解説する、USBホストで実行するUSBターゲットの初期化処理では、図3に示すUSBのデバイス・ステート図を頭に入れて読んでください。USBに電源を投入し、バス・リセットを発行後は、USBターゲットはデフォルト状態となります。USBホストは、ここでSET_ADDRESSデバイス・リクエストを発行することで、USBターゲットに対してUSBアドレスを割り当てます。さらにSET_CONFIGURATIONデバイス・リクエストを発行することで、USBターゲットが動作可能状態となります。これはすべてのUSBターゲットで共通の処理となります。

USBの基本プロトコルの理解

本特集では誌面のつごうで、USBの各種プロトコルについて

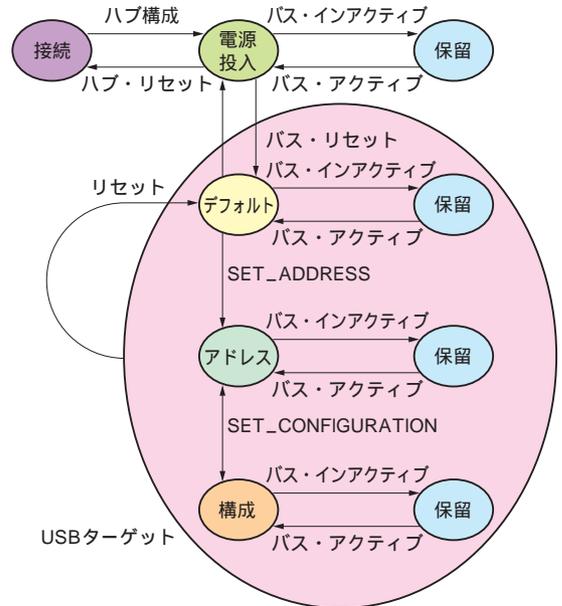


図3 USBのデバイス・ステート図

の詳しい解説はできませんでした。実際のパケットの組み立てや、受け取ったパケットに対するACKなどの送出処理などは、ある程度USBコントローラが自動的に行ってくれます。したがって、すべてのプロトコルの詳細を詳しく理解しておく必要はありません。しかしUSBホスト機能の実装やデバッグ作業をスムーズに進めるためには、USB通信プロトコルについて、ある程度の理解も必要です。最後に、USBシステムを設計する場合に有効な、各種参考文献を示しておきます。

参考文献

- (1) USBターゲット機器設計のすべて, TECH I Vol.27, CQ出版社.
- (2) 改訂新版 USBハード&ソフト開発のすべて, TECH I Vol.30, CQ出版社.
- (3) USBホスト&ターゲット・システム設計技法, Interface 2004年10月号, CQ出版社.

くわの・まさひこ パステルマジック