

# ベンダ定義クラス機器用 簡単USBドライバ

中本 伸一

USBデバイスの中には、標準クラスではないデバイスも数多くあります。ベンダ定義クラス・デバイスと呼びます。

ベンダ定義クラス・デバイスは、ほとんどの場合は製造メーカからデバイス・ドライバが提供されます。USB機能付きマイコンから使いたい場合は、ドライバを自分で作成することになります。この章ではこうしたベンダ定義デバイスのドライバ作成方法のヒントを紹介します。

もちろん得られた情報を運用した結果は、あくまでも自己責任になりますので、ご注意ください。デバイスがうまく動いても動かなくても、他人に頼ることはできません。

## オリジナル! ベンダ定義クラスが多くなる理由

### ● ほとんどのUSBデバイスはベンダ定義クラス

世の中に出回っている、ほとんどのUSBデバイスは、ベンダ定義クラス・デバイスと呼ばれ、メーカが提供するデバイス・ドライバをインストールしなければ、認識されません。

こうしたデバイスを、自作USBマイコンで利用する際には、どうやって自作ドライバを作成すればよいのでしょうか?

実は多くの場合、こうしたベンダ定義クラス・デバイスは、全くオリジナルなUSB制御ではなく、既存のデバイス・クラスを、少しだけ拡張して作成されることが多いようです。それには以下の理由があります。

### ● 標準クラスのUSB仕様はよく考えられている

USBの公式サイト (<http://www.usb.org/>) からは、標準クラスに関するドキュメントが、豊富にダウンロード可能です。

標準クラス・デバイスのUSBプロトコルは、世界中の関連する機器メーカの現役エンジニアが、ワーキング・グループを作り、皆で長時間検討した上で、標準的なプロトコルとして自信を持って作成したもので

す。

要は現場の現役エンジニアの声を元に作成された仕様なので、ほとんどあらゆる利用方法を網羅した、柔軟性のあるプロトコルに仕上がっています。

またサブクラスという派生系の拡張仕様で、さらに広範囲なデバイスへの対応を可能にしています。

このように標準クラスのプロトコルには、世界中のエンジニアの知恵と経験が込められています。

どう考えてもゼロから仕様を考えるより、既存のプロトコルをベースにした方がよさそうです。

### ● しかし独自の機能を追加して差別化したい

そうはいつても、自社製品を他社と差別化して、より多くのシェアを獲得するのは、エンジニアの務めです。

他にはない機能を追求していくと、自然と標準クラスでは対応できない機能を、追加したくなるのが人情です。

またUSBの標準クラス仕様では、エラー処理に関する定義はあいまいです。エラーが発生した際の挙動は、あまり詳しく定義されていません。したがって基本的には、何らかのエラーが発生すると、単にデバイスは止まることとなります。

しかしメーカとしては、自動的にエラーから回復して、なるべくデバイスの動作を止めないようにしたり、どんなエラーが発生したかをアプリケーションに伝えたりしたいので、標準クラスを拡張して、独自のエラー処理機能を追加する必要に迫られます。

### ▶ 最も大きな理由…アップデート対応

多くのメーカでは、継続的に自社製品をサポートします。ユーザから何かの問題が寄せられれば、できる限りそれに対応して、ドライバをアップデートします。また製品の世代が進んで、新しい機能を実装した場合にも、ドライバをアップデートする必要があります。こうした製品サポートを考えると、OSに用意されている標準クラス・ドライバで認識されてしまうよりも、使用開始時に独自のドライバをインストールするチャンスが得られるベンダ定義クラスにしたいので