

# 混沌とした中から

## 各種インターフェース (9)

各種インターフェースの紹介も9回目となり、最後に紹介するのがUSBです。USB (Universal Serial Bus)は、キーボードやマウス、モデム、ジョイスティックなどの周辺機器とパソコンを結ぶシリアルインターフェース規格です。もともとは本体と周辺機器との間であまり大容量のデータをやり取りしない機器を接続するための規格として策定されたもので、始めはUSB1.1という規格名でした。ここでこの後のUSBに影響するのがIEEE1394の前回まで説明したシリアルSCSIインターフェースです。というのもこの当時高速シリアルインターフェースとして予定されていたのがIEEE1394だったのですが、立ち上げが遅れてしまっているうちにその間にUSB1.1が爆発的に普及してしまったということがあります。それまでPS/2(その前はRS232-Cのシリアルインターフェースでしたか)であったマウスとキーボードのインターフェースもノートを中心にこのごろはUSBになっています。

ところでちょっとここでPS/2インターフェースについてです。PS/2というのはIBMのパソコンの名前です。PS/2は1987年に発表されたPC/AT互換機で、いろいろ新しい技術を採用したパソコンで、それまでのPC/ATのパソコンからの脱皮を図ったものでした。新しい技術としては、データバス規格としてそれまでISAであったものを32ビット対応 (ISAバスは16ビット対応で、拡張したEISAで32ビット対応になったが、EISAはIBMは採用していない)したMCAバスであったり、OSとしてもOS/2を採用しました。MCAバスは独自バスでこれまでのISAバスは自由に使えたのですが、このときIBMは互換機メーカーからライセンス料を徴収しようとしたためメーカーの各社の反発を受け普及しなかった(三菱はとかイギリスのapricotは採用した)ということがあります。このPS/2でキーボードやマウスを接続するためのコネクタ規格を作りました。その頃のハードでは重すぎて普及しなかったOS/2が共同開発していたマイクロソフトのWindowsに押されて普及しなかったため、MCAバスなどと同様にPS/2というパソコン自体が忘れ去られたようになっていますが、このPS/2インターフェースだけが残って名前の由来が忘れ去られています。

USBに戻ります。初めのUSB1.1は転送速度がLS (Low Speed) とFS (Full Speed) モードの2種類あり、それぞれ1.5Mbpsと12Mbpsとなります。キーボードやマウスなどはLSモードを使用することになり、大容量の通信を行うスキャナやプリンタなどはFSモードを使用することになります。それとUSB最初のバージョンである1.0の場合はあいまいでしたが、1.1となって周辺機器の消費電力の使用などが規定され、USBケーブルだけで電源供給を含め周辺機器を使用することができるようになりました。コネクタ形状は4ピンのコネクタで、初期は横に並べた断面が長方形のType Aと2つずつ上下に配置した断面が正方形に近いものの2種類でしたが、それではデジタルカメラなどではコネクタの取り付け場所に問題があるということでType Bを小型にしたMini-Bというコネクタの仕様もあります。この低速のUSB1.1によりノートを中心にマウスがUSB対応が中心となり、このごろでは1台に3~4ポートあるものもありますし、電源供給できるということからパソコンとは直接関係のない周辺グッズであるPCライトや保温用ヒータなどまであります。

USBの特徴として自動認識があります。つまり一昔前の表現で言えばPlug&playですし、電源を入れたまま抜き差しのできるホットプラグにも対応しています。この点からもそれまでのものより使いやすくなっています。(次回へ続く)

(今週の情報誌から)

○日経パソコン 2月27日号

特集 姿を見せ始めたWindowsVista

→2001年のWindowsXPの発売から早5年、次のWindowsであるWindowsVistaの姿が見え始めた。どこがどう変わるのか。その特徴は、ユーザインターフェースの刷新と検索機能の充実、セキュリティ強化3点。ユーザインターフェースでは、半透明のウィンドウと3D表示、よく使うアプリケーションをすぐに使えるようにする機能など。検索機能は、HDDが大容量化したことに対応するための内部データの検索(タイトルだけでなく全文検索も可能)、検索結果は仮想フォルダに収集することもできる。セキュリティ機能強化は、管理者といえどもオールマイティではなくなる。

○日経システム構築 3月号

特集 システムの引継ぎ

→このシステムはあの人しかわからないということはよくある。しかし、その人がいなくなったらどうする。頭の中のノウハウをどう引き継げばよいのか。問題点を明確にし、原因を基にして現状を打破し、引継ぎ問題を撲滅するにはどうするか。文書化はもちろん必要だが、あまりに詳細なドキュメントは必要ない。開発段階から将来を見据え、変更があった点、気づいた点は文書にフィードバックすることに努力する。ドキュメントも生きている。手をかけないと死んでしまう。