

## 第5章

### 外部からの入出力をマスタする

# シリアル・ポートで遊ぼう

パソコンに、9ピンのコネクタがついているのを見たことはないでしょうか？ 最近のパソコンでは省略されているものもあるので、見たことのない人もいるかもしれません。玄箱PROの場合には、内蔵シリアル・ポートもありますが、シリアル・コンソールに使われています。本章ではUSBシリアル・ケーブルを利用しています(写真5-1)。

「シリアル(serial)」というのは直列・順次という意味で、データを1ビットずつ順に送っていることから名づけられています。同時にデータを送る方法もあり、その場合は「パラレル」(並列)と呼ばれます。パラレルの場合は4ビット、8ビット、16ビットなどのデータを一度に送ります。最近ではUSBが普及したので、シリアル・ポートの影が薄れてしまいましたが、まだ自作のデバイスを作るにはUSBよりも簡単です。また信号の流用によって、簡単に外部機器を制御することも可能です。そこでUSBシリアルを玄箱PROにつないで遊んでみます。

### 5-1 シリアル・ポートを知ろう

#### ● シリアル・ポートの規格

シリアル方式の通信にはいくつもの規格があります。USBやEthernetもシリアル通信です。ここで取り上げるのはD-sub 9ピン・コネクタを使う、EIA-574規格です。

この規格は、RS-232C(EIA-232)規格(25ピン)の、ピン数が少ないものと見なせるので、歴史的経緯

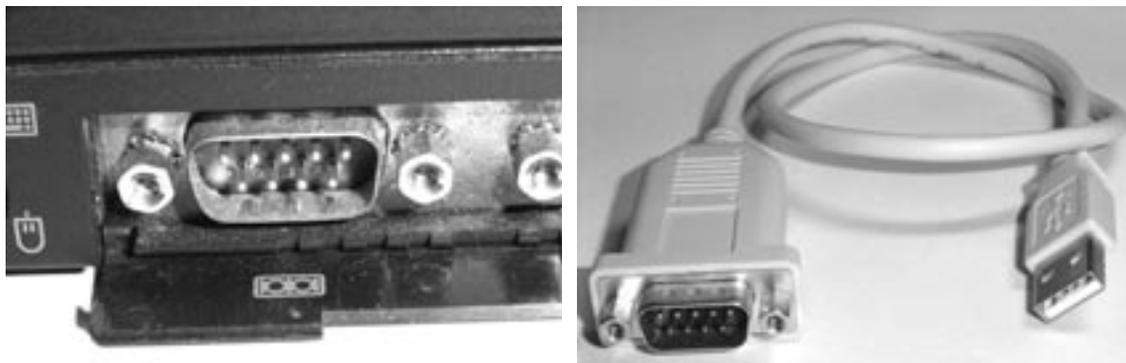


写真5-1 左：ノート・パソコンのシリアル・ポート、右：USBシリアル・ケーブル

表5-1 EIA-574(RS-232C)のD-sub 9ピン・コネクタのピン配置

ピン番号	略称	意味	通信の向き
①	DCD	キャリア検出	入力
②	RXD	受信データ	入力
③	TXD	送信データ	出力
④	DTR	データ端末レディ	出力
⑤	GND	グラウンド	なし
⑥	DSR	データ・セット・レディ	入力
⑦	RTS	送信要求	出力
⑧	CTS	送信可能	入力
⑨	RI	被呼表示	入力

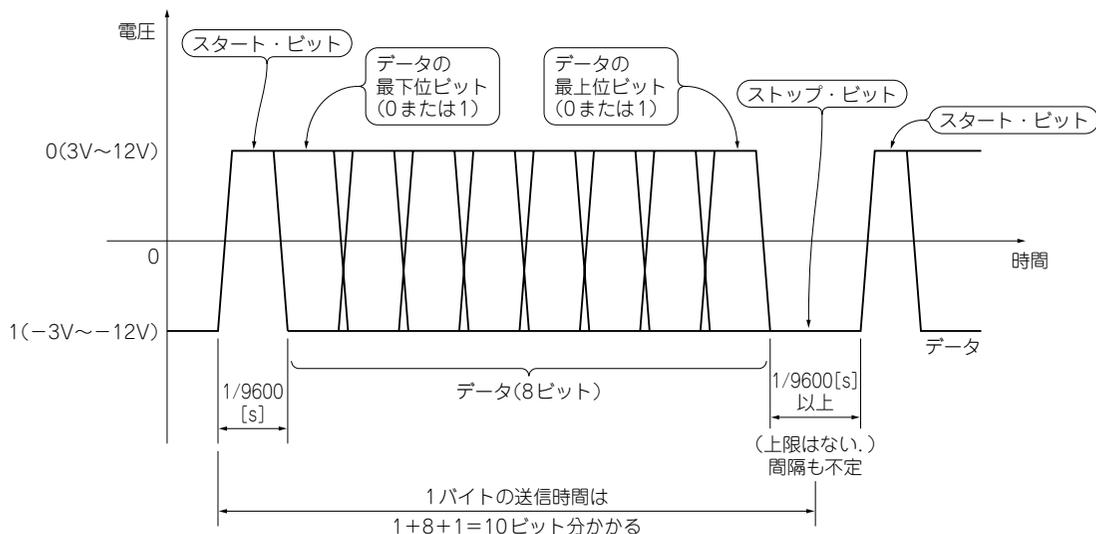


図5-1 9600bps, 8ビット, パリティなし, ストップ・ビット1のときのデータの送り方

からRS-232C(規格, 形式)とも呼ばれることがあります。ここでも厳密な区別は必要ないので, 以下区別はしません。

### ● EIA-574(RS-232C)のピンの意味

各ピンの意味は表5-1のようになっています。グラウンド(5ピン)に対して, 信号ピンの電圧が3[V] ~ 15[V]のとき0を, -3[V] ~ -15[V]のとき1を意味します。通信の向きはパソコン/USBシリアルなど, オス側のコネクタのついている機器から見たものです。

最低限, 通信に必要なのは, ②, ③, ⑤ピンの3本です。玄箱PROのシリアル・コンソールでは, この3本のみを使っています。このときには⑦, ⑧ピンをコネクタ内でショートしておく(ループバックしておく)と, 通信できないトラブルを防げます。

パソコン同士をつなぐときには, クロス・ケーブルを使って②, ③を相手の③, ②に, ⑦, ⑧を⑧, ⑦につなぐことで通信できます(ソフトウェアにより指定がある場合はそのようにつなぐ)。⑤はそのまま⑤へつなぎます。

## ● データの送受信(TXDとRXD)

データは、7または8ビット単位で送受されます(図5-1)。TXDで送信し、RXDで受信します。最初にデータの開始を表すスタート・ビット(0)、データ(最下位ビットから順に送る、0または1が7個または8個続く)、(必要な場合に)パリティ・ビットの0または1、ストップ・ビット(1)が1単位です。1ビットを送る時間(タイミング)を送受する機器間で合わせることで、1本の線で1方向のデータのやり取りを実現します。送受で2本と、電圧の基準となるグラウンドを合わせて3本の通信線でデータのやり取りが実現されます。

送受間で、ボーレート(1ビットを送る時間を決める)、データの単位(7または8ビット)、パリティ(データ・エラーを検出する方式、偶数・奇数・なし)、ストップ・ビットの長さ(1ビットまたは2ビット)、フロー制御の方式を一致させます。ボーレートの単位はbps(bit per second, 1秒あたりに送るビット数)です。

9600bpsの場合は、1ビットの長さは1/9600[sec]です。データのほかに、スタート・ビットとストップ・ビットがつくので、データが8ビット(1バイト)でパリティ・ビットなしの場合でも、8+2=10ビットを送る必要があります。そのため、9600bpsでは、1秒間に送れるのは960バイトまでです。

## ● フロー制御(RTSとCTS)

機器がデータを受け取る準備ができていないときに、相手からデータを送られると受け取りそこなってしまう(取りこぼし)。そこで、相手に送信を待たせることでデータの取りこぼしを防ぐのがフロー(flow, 流れ)制御です。

フロー制御には、データの中の2文字(XON/XOFF)を送信中断と再開に使うソフトウェア・フロー制御、RTS/CTS線を使うハードウェア・フロー制御があります(RTSが+のときに相手に送信許可を示し、CTSが+のとき相手から送信許可を受けたとする)。ソフトウェア・フローでは、通信に必要な線が省略できる反面、2文字が特別な意味に使われてしまいます。ハードウェア・フローでは、データに制約はありませんが、専用に2本の線が必要になります。

お互いが十分に速く、取りこぼしがないことを仮定できるか、取りこぼしをほかの方法でカバーできる場合は、フロー制御をしない(フロー制御なし)場合もあります。玄箱PROのシリアル・コンソールが、その例です。⑦、⑧ピンをコネクタ内につないでおくのは、フロー制御なしで通信するときにも、通信用LSIの都合でCTSが+でないと送信しないようになっているものがある(過去にあった)ための対策です。自身のRTS(+をCTSにつなぐことで、つねに+と見なしてしまいます。

## ● モデム制御用記号(DCD, DSR, DTR, RI)

DCD, DSR, DTR, RIはモデム制御用信号です。最近ではあまり利用されなくなりましたが、電話回線を通してコンピュータが通信するのにモデムが長い間利用されてきました。DCDは、モデムが相手との通信を確立したことをパソコンに示します。RIは、モデムがパソコンに電話がかかっていることを知らせます。

DTRはモデムに、パソコンが準備ができたことを知らせます。モデムは、これをDSRにより受け取ります。これにより、電話線を通した通信にモデムを使うときに、自動でダイヤルしたり、着信時にパソコンに知らせるといったことが想定されています。

## ● 信号の転用

RTS, CTS, DCD, DTR, DSR, RI線は, フロー制御やモデム制御が不要な場合には別の用途に転用することもできます. 転用することで最大2本のデジタル(ON/OFF)出力と, 最大4本のデジタル入力が使えることになります.

デジタル出力の電圧は, パソコン(USBシリアル)によって $\pm 3 \sim \pm 15V$ の範囲で差があることに気をつけてください. デジタル入力の電圧は+ (0) のとき $3 \sim 15V$ , - (1) のとき $-3 \sim -15V$  ( $0 \sim -15V$ でもたいていはOK)の範囲となるようにすることが必要です.

## 5-2 シリアル・ポートを使ってみる(シリアル・チェッカの製作)

### ● シリアル・ポートの信号を見る

シリアル・ポートにどのような信号が流れているのでしょうか, ここではシリアル・チェッカを作成し, LEDを点灯させ確かめてみます.

### ● シリアル・チェッカの準備

シリアル・ポートの信号を, 信号の0/1に合わせてLEDが点灯することで目に見えるようにします. いくつか市販もされているのですが, 自作することにしました.

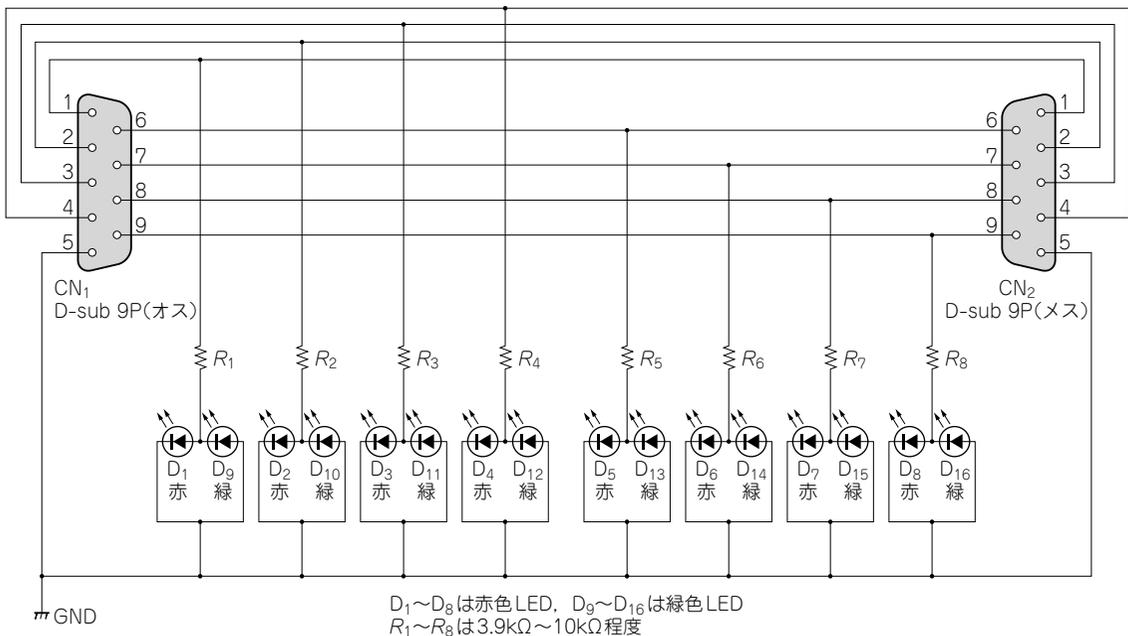


図5-2 シリアル・チェッカの回路図