6-2 パソコンと通信する

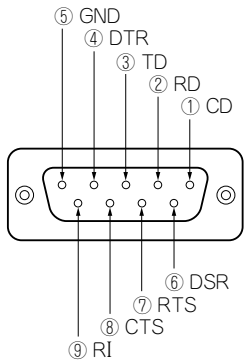
AVRどうし，あるいはほかのマイクロコントローラのUARTポートのような正論理のTTLレベルのシリアル・ポートと通信を行うのであれば，TXDを相手のRXDに，RXDを相手のTXDに直接接続するだけで，あとはプログラムを準備すれば通信が可能になります．ところが，パソコンのシリアル・ポートの場合はRS-232Cあるいは EIA-574と呼ばれ，電圧レベルや論理が異なります．

また，シリアル・ポートをもたないノート・パソコンなどでも市販のUSB-RS（シリアル）アダプタを使用すれば，簡単にシリアル・ポートを追加することができます．この場合は，必ずUSB-RSアダプタに仮想COMポート・ドライバが付属しているので，ソフト的にもCOMポートが増設されたことになり，Windows APIからのシリアル通信が可能になります．ただし，この場合はあくまでもCOMポートが増設されたことになりますから，USBの機能や性能がフルに発揮されるわけではありません．

● RS-232C（EIA-574）

RS-232Cは，パソコンが世に出た当初から使われているシリアル通信の規格です．当初はパソコンとモデムを接続するためのもので，コントロール線の名称にその名残を見ることができます．現在ではEIA-232F，あるいはEIA-574（9ピン版のEIA-232F）と呼ぶのが正式のようです．しかし，パソコンのカタログなどを見ると「シリアル・ポート：RS-232C D-SUB 9ピン，16550A互換」といったEIA準拠という意味の書き方をしていることが多いようです．また，データ通信を行うTXD，RXD以外に数本のコントロール線があり，Windowsパソコンの場合，Windows APIから制御可能です．このコントロール線を使ってフロー制御を行うこともできますが，TXD，RXD以外の線はあまり使われないようです．

RS-232Cの信号線は，パソコンなどでよく使われる5 Vや3.3 Vのロジック・レベルでなく±15 Vが利用されます．したがって，通常のロジックICやCPUなどと接続する際には，必ずレベル変換が必要です．レベル変換が必要ですが，それ以外は単純な非同期通信が行われるだけなので，受け手の設計が簡単にすみ，現在でも簡易な通信方式として根強く利用されています．**図6-6**に，PC/AT互換機で使



ピン番号	I/O	論理	略号	原　文	機能（もともとの意味）
①	I	正	CD	Carrier Detect	相手のモデムが送信の準備ができている
②	I	負	RD	Receive Data	受信したデータ
③	O	負	TD	Transmission Data	送信するデータ
④	O	正	DTR	Data Terminal Ready	端末（パソコン）の通信準備ができている
⑤			GND	Ground	グラウンド
⑥	I	正	DSR	Data Set Ready	モデムの通信準備ができている
⑦	O	正	RTS	Request To Send	送信要求． 全二重ではつねにON
⑧	I	正	CTS	Clear To Send	送信可（送信を許可する）
⑨	I	正	RI	Ring Indication	電話回線に呼び出しを受けている

図6-6　PC/AT互換機で使われる9ピンRS-232Cのコネクタと信号線
EIA-574規格，通常パソコン側にDサブのオス・コネクタが使われる，データ線は負論理（−5〜−15 Vが‘1’），コントロール線は正論理．

リスト6-1　簡単な通信プログラム(prg0601.c)

```
//
// WinAVR シリアル通信テスト・プログラム prg0601.c
//     [Current Configration Options]に mega48 を設定すること
#include <avr/io.h>

int main( void )
{
    unsigned char x;

    DDRB  = 0b11111111;
    DDRC  = 0b00001111;
    DDRD  = 0b11110000;

    PORTB = 0b00000111;

    UBRR0 = 25;

    UCSR0B = 0b00011000;

    UCSR0C = 0b00000110;

    while(1) {

        while( !(UCSR0A & 0b10000000 ) )

        x = UDR0;

        PORTC = 0x0f & x;
        PORTD = 0xf0 & x;

        while( !(UCSR0A & 0b00100000 ) )

        UDR0 = x;

    }

}
```

われる9ピン・コネクタと各信号線を示します．

ATmega48ボードでは，レベル変換ICを使ってAVRの信号線のV_{cc}レベル，本ボードでは乾電池3本を使用するので4.5 V，をRS-232Cのレベルに変換しています．また，基板に設置している9ピン・コネクタは，パソコンとストレート・ケーブルで接続することを想定し，TXDとRXDとを**図6-6**とは逆に配置しています．

● 通信実験

それでは，**図6-5**のポーリングを順に3回並べたテスト・プログラムを試します．テスト・プログラムを**リスト6-1**に示します．**リスト6-1**では，初めにチップの初期化，シリアル通信関連の設定を行い，そのあと送受信のループに入ります．

シリアル通信関連の設定では，初めにボーレート・クロックの設定を行います．システム・クロックを，内蔵1 MHzを使うことにして，16×26分周で2400ボー規格に合うシリアル通信クロックを生成します．分周モードはデフォルト設定の標準モードを使用し，分周値はUBRRHに0，UBRRLに25を設定します．続いて，USART制御ステータス・レジスタB，Cに送信，受信の動作仕様を設定します．USART制御ステータス・レジスタBには，割り込み未使用，送信・受信イネーブル，キャラクタ・サイズ8ビットを設定しています．USART制御ステータス・レジスタCには，非同期モード，パリティ：なし，ストップ・ビット：1ビット，キャラクタ・サイズ：8ビットを設定しています．

ループでは，初めに1バイト受信を行い，その後，そのデータをLEDに出力の後，パソコンにエコーバック送信します．**リスト6-1**を動作させるためには，パソコン側で2400ボー，8ビット，パリティ：なし，フロー制御：なしに設定したターミナル・ソフトを用意します．この設定のターミナル・ソフトとともに**リスト6-1**を動作させたときのターミナル・ソフトの画面，Atmega48ボードの7セグメントLEDの様子を**図6-7**と**写真6-1**に示します．**図6-7**では，キーボードから順にabcedfgと入力し，そのデータがエコーバックしていることがわかります．**写真6-1**では，入力された最後の文字gのアスキ・コード0x67に対応したセルa，b，c，f，gが点灯していることがわかります．

図6-7　リスト6-1の動作の様子
パソコンから「abcdefg」と入力すると，エコーバックされたキーボードの文字が受信できている．

リスト6-2　PC側の簡単な通信プログラム（prg0602.cpp）

```
//
// prg0602.c  COMポート送受信プログラム
//
#include <windows.h>
#include <conio.h>
#include <stdio.h>

void main()
{
    HANDLE h;                                   ← ファイル・ハンドル

    h = CreateFile( "COM1",
                    GENERIC_READ | GENERIC_WRITE,
                    0,                          ← 非共有
                    0,                          ← セキュリティ属性，使用しない
                    OPEN_EXISTING,              ← 既存ファイルのオープン
                    0, 0 );                     ← 属性，テンプレート
    if( h == INVALID_HANDLE_VALUE ){
            printf("オープン・エラー¥n");
            return;
    }

    DCB dcb;
    GetCommState( h , &dcb );                   ← 現在の設定をベースにして設定
    dcb.ByteSize = 8;                           ← 8ビット
    dcb.BaudRate = 2400;                        ← 2400ボー
    dcb.fParity  = 0;                           ← パリティoff
    dcb.fOutxCtsFlow = 0;                       ← CTCフロー制御off
    if( ! SetCommState( h , &dcb )){
```

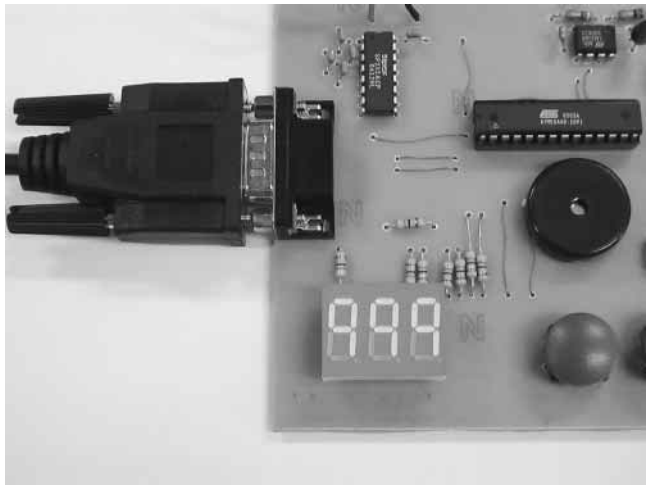


写真6-1　リスト6-1の動作の様子
入力された最後の文字gのアスキ・コード0x67に対応したセルa，b，c，f，gが点灯している．