

プログラム

Java

<http://www.02.246.ne.jp/~torutk/javahow2/javaxcomm.html>

RXTX ライブラリ を使う。

バ イ ナ リ
ソ ー ス
API ド キ ュ メ ン ト
サンプル

インストール方法は、解凍してテキストウな場所にコピー。
ライブラリへのパスは、

java のオプションの `java.library.path`

Windows の場合

環境変数の PATH

Linux の場合

環境変数の LD_LIBRARY_PATH

でも可。

Java (古いかも)

<http://www.okakogi.go.jp/People/miwa/program/how2com/how2com.html>

<http://www.moriwaki.net/wiki/index.php?%5B%5BCommunicationsAPI%5D%5D>

サ ン プ ル プ ロ グ ラ ム
必要なライブラリ

VB

MSCOMM を使えば OK

プログラムで気をつけること

- ・ (Data Terminal Ready(データ端末レディ)
- ・ (Request To Send)

を送ればほとんどの場合は問題ない。

開発環境

RS232C のポートをクロスケーブルで繋ぐ

または

VirtualPC で COM ポートの設定で名前付きパイプを使う

例
COM1 ¥¥.¥pipe¥com
COM2 ¥¥.¥pipe¥com

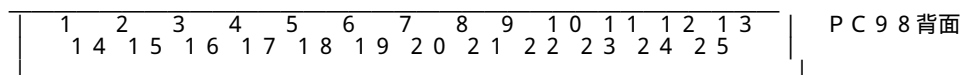
とすると VirtualPC 上の COM1 と COM2 がクロスされる。

RS232Cプロジェクトより

<http://kasai86.ld.infoseek.co.jp/old/rs232c.html>

ピンの配置

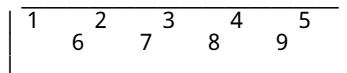
DB 25 ピンのシリアルポート (PC9801 シリーズ) のピンの配置



信号	ピン	機能
GND	1	アース
TXD	2	送信データ
RXD	3	受信データ
RTS	4	モデムへの送信要求
CTS	5	モデムからの送信可能信号
DSR	6	モデムからの動作可能信号
DCD	8	キャリア検出 (モデムが相手方のデータ受信)
RXC	17	受信クロック
DTR	20	ターミナル (パソコン) の動作可能信号
TXC	24	送信クロック出力

上記のピン配置は PC9801 シリーズのものです。

DB 9 ピン (DOS/V 機等) 背面からみた図



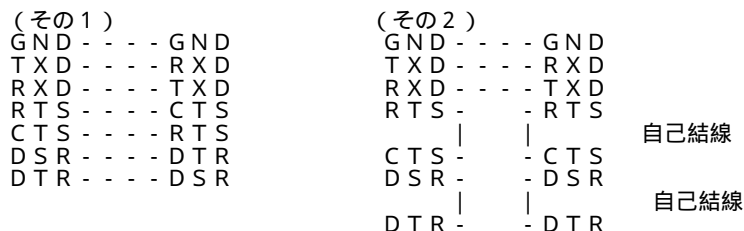
信号	ピン
DCD	1
RXD	2
TXD	3
DTR	4
GND	5
DSR	6
RTS	7
CTS	8
RI	9

RS232C では、,,, が全て ON (モデムもパソコンも動作可能状態で、データ送信要求がなされた) 時、TXD よりデータが送信されます。(モデムが接続されている場合、パソコンからモデムにデータが転送され、モデムから相手の機器にデータが送出される。)

パソコン同士の結線

パソコン間のデータ転送では、7 ピン (7 つの信号) のみ使われます。他のピンは使われません。7 つの回線 (信号) の機能は、TXD (送信)、RXD (受信)、RTS (送信要求)、CTS (送信可)、DSR (データセットレディ)、GND (接地)、DTR (データ末端レディ) です。

昔の BASIC では RS232C をオープンすると (OPEN 文) RTS (送信要求) DTR (データ末端レディ) が ON になります。従って、下記のように結線すれば、RTS, CTS, DSR, DTR の全てが ON となりデータの送信ができるようになります。



TXD, RXD は必ずクロスして結線します。(送信側のデータ送信と受信側のデータ受信を結ぶ。当たり前ですが)

その1の結線では、RTSとCTS、DTRとDSRをクロスさせ、それぞれのRTS, DTR

のONが相手のCTS, DSRをONするようにします。

その2では、自分のRTSとCTS, DTRとDSRを自己結線し、自己のRTS, CTSのON

が自己のCTS, DSRをONするようにします。ただし、この場合は相手のパソコンの状況にかかわらず、データ送信可能となるので、あいてのパソコンの準備ができていなければ、データは無駄な垂れ流しとなってしまいます。

さて、以上はパソコン間の結線でも比較的厳密な場合で、古いパソコンの BASIC の場合もっとルーズなケースが多いようです。古い BASIC では、DTRとDSRもまったくチェックしてないケースがほとんどです。従って、DTR, DSRを無視して、5本の信号線だけで、送受信可となります。

更には、GND - - - GND, TXD - - - RXD, RXD - - - TXDの3本だけでO.K. という場合も結構あります。

通信プロトコル

パソコン同士をつなぐ場合であっても、お互いに通信条件について一致させておかないと、データのやりとりができません。設定すべき条件として一般的には以下のものがあります。

ボーレート (データ転送スピード)

データ転送のスピードです。1秒間に転送するビット数をあらわします。BASICでは、300、600、1200、2400、4800、9600ボーの中から選択することが多いと思います。

ビット長

データの1文字をあらわすのに、7ビットで表すか、8ビットで表すかの選択です。

ストップビット

データの区切りとして、データの最後につけるのがストップビットです。これを、1ビットにするか2ビットにするか選択します。

パリティチェック

送られてきた1文字分のデータに誤りがないか、確認するために付加するビットです。偶パリ、奇パリ、パリティ無しから選択します。

Xパラメータ

データが受信能力を超えて送られてくると、受信バッファからデータがあふれて、データの取りこぼしをしてしまいます。この時、送信側に一時的にデータ送信の停止を要求することができます。この機能のON(XON)、OFF(XOFF)をきめます。受信側で受け取ったデータをプリンターに出力するなど、時間のかかる処理をする時にはXONにします。ただし、昔のBASICシステムでは、送信側が停止の合図を受け取っても、自動的に、送信停止にならないことが多いようです。BASICプログラムで停止の合図をうけたら、一時停止になるよう、プログラムする必要があります。

Sパラメータ

データビット長で7ビットを選択すると、(JISコード)カナ文字が送信できません。SパラメータをONにすると、SOコード(以下のデータをカナとして扱う)、SIコード(以下のデータを英数文字として扱う)の送信がなされ、カナの送信も可能になります。

これ以外にも、各機種固有の設定(キャリッジリターンの取り扱いなど)がありますが、ここでは割愛します。