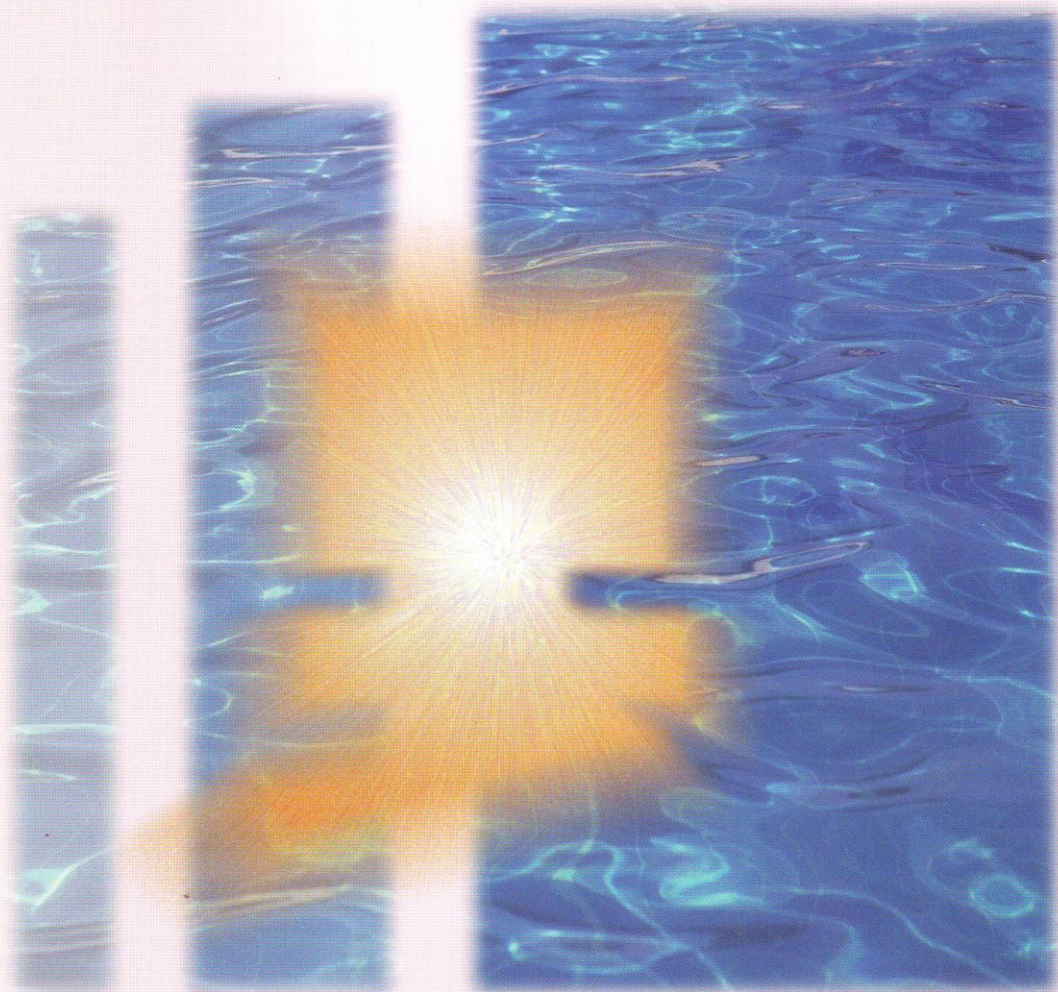


CQ ハンドブック・シリーズ

CQ Communication

# 無線データ 伝送と インターネット

ワイヤレス伝送システムの構築とハードウェア



CQ出版社



# 19200bps無線モデム内蔵 トランシーバを用いた IP 接続

430MHz でデータ伝送し Web を見る  
最大 1.0 ~ 1.6K バイト / 秒で転送

田中 卓史

## ある日の長女との会話

M: 高校の文化祭でインターネットの実演をしたいのだけど、学校が電話線を引いてくれないの。

T: そうだね。2日間の文化祭のために電話線を引いてはくれないだろうな。PHSを使ったら。

M: 物理部の部費はマック(Macintosh)を買うのが精一杯で全く余裕がないの。ハムのパケット無線でつなぐことできない。

T: 無手順ならつながると思うけど、IP接続はできるかな。一度やってみる価値はありそうだね。

と、もともと無線の好きなお父さんは、忘れていたハムの本能が目覚まし、高校生と一緒にIP接続の実験を開始したのであります。

## はじめの実験

娘のMacintosh(Performa588)は、モデムにより電話線経由でPPP接続されています。モデムをパケット無線のTNCに置き替えれば無線でつながる可能性があります。しかし、モデムとTNCとではコマンドも接続手順も通信方式も異なるので、どのような問

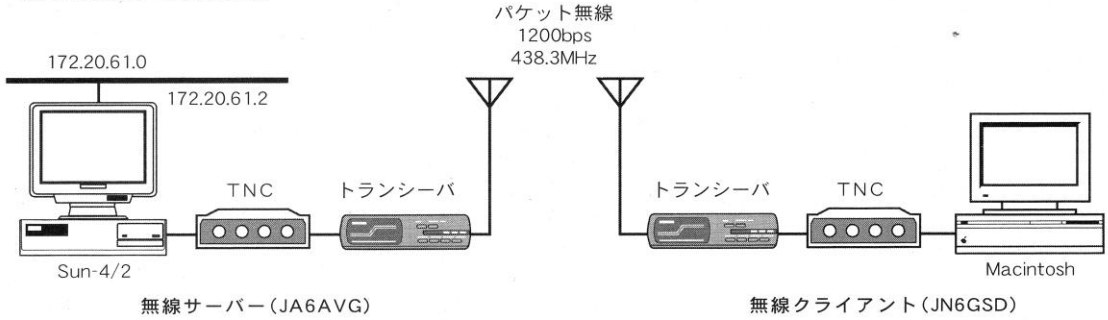
題が起こるか実験してみることにしました。

まず、UNIXワークステーション(Sun-4/2)にPPPをインストールし、シリアル・ポートとMacintoshのモデム・ポートをRS-232Cのクロス・ケーブルで直結しました。MacintoshのPPPにおいて電話番号の部分空白にしてアドレスなどを設定すると、問題なくIP接続ができ、NetscapeやNCSA-telnetなどのアプリケーションを使うことができます。

第1段階が成功したので、次に、UNIXワークステーションとMacintoshに、それぞれTNCと無線機をつなぎました(図1)。アマチュア無線のパケット無線では、TNCに自局のコールサインを設定し、CONNECTコマンドで相手局に接続して通信します。

相手局がUNIXマシンの場合は、制御コードを含むすべての信号を転送したいので、接続後にTNCが透過モードに移行するように設定しておきます。TNCは通常、改行コードが入力されるか、入力された文字が一定数になると、入力された文字列をまとめてAX.25のパケットにして送信しますが、UNIXでは1文字ごとでも送信し、エコーバックさせたいので、最後の文字がTNCに入力されてから100ミリ秒以内に次の文字が来ないときも送信するように設定します。UNIXマシンのほうは、モデムの接続に準じて、シリアル・ポートからログインできるように、また、ハードウェアで信号のフロー制御ができるように、デー

図1 無線による無手順接続の実験



タ・キャリアが検出できるように設定しておきます。ところが、これでもうまく行くとしたら、早速、問題が生じました。UNIXはログインを受け付けるため、シリアル・ポートにlogin:プロンプトを出します。一方、TNCもコマンドを受け付けるためにcmd:プロンプトを出します。UNIXはこれをログイン名として受け取り、Passwordを要求します。TNCはコマンドが入力されたと思い、そのようなコマンドは知らないと、“?EH”を出し、UNIXはおかしなパスワードがきたので login incorrectといい、TNCはこれをまた誤ったコマンドと思い、UNIXとTNCの間はループに陥ります。

この問題を解決するには、二つのTNC間にAX.25のリンクが成立し、TNCが透過モードに移行した後に初めて、サーバー側のワークステーションとTNCをつなぐことが必要になります。そこで、手元にあった部品でワークステーションとTNCを結ぶケーブルの途中に挿入するRS-232Cのコネクタ(アダプタ)を作りました(図2)。

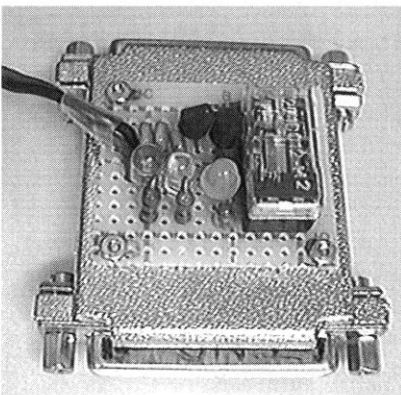
これはTNCが出力するDCD(キャリア検出)信号を利用して、AX.25のリンクが成立してから数100ミリ秒後にワークステーションとTNCの送受信データの回路を接続するモノです。

このコネクタによりループ問題が解決し、UNIXワークステーションが無線経由で遠隔利用できるよう

になりました。UNIXは、ふつう全2重通信で使うことが想定されています。端末のキーボードから打った1文字は、コールサインの付いたAX.25のパケットになり、無線リンクを往復して、戻ってきてからCRT画面に表示されます。これが数100ミリ秒かかるため、直接UNIXを使うのに比べるとキー操作が間延びした感じになります。

無手順接続が成功したので、PPP接続ができれば「めでたしめでたし」になります。そこで、MacintoshのPPPを起動すると、まず、UNIXへのloginが成功します。次にUNIX側のPPPが起動されると、回線の状態を調べてデータリンクを確立するためのLCP(Link Control Protocol)パケットが送られてきます。ところが、これにMacintosh側のPPPが応答せず、リンクが成立しないため、その後の認証の段階へと進みません。PPPに関するドキュメントなどを見てみると、PPPには全2重通信が想定されていることが書かれています。また、パケットの転送には同期式で用いられるHDLCフレームをRS-232Cのような非同期回線で送るように拡張した方式が用いられているようです。

フラグで囲まれた一まとまりのHDLCフレームがAX.25パケットを作るため、TNC側の都合で適当に切られて送られることになると、単にモデムをTNCに置き替えてPPP接続を行う方法は、たとえ無線を2バンド使って全2重にしたとしても、見通しが暗くなってきました。



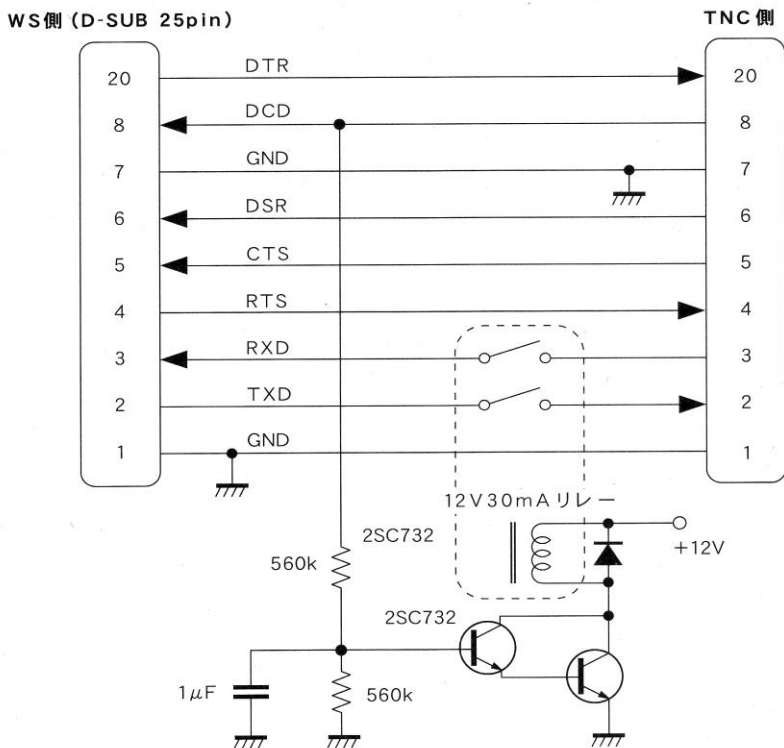
DCDでディレクションとなるRS-232Cコネクタ

## Linuxによる無線接続

インターネットの普及する以前、福岡にもTCP/IPによるアマチュア無線のパケット無線ネットワークができていて、KA9Qパッケージと呼ばれるソフトを用いて無線でIP接続を行ったことがあります。

KA9QではTNCをKISSというモードに設定します。KISSモードに設定すると、TNCは単にRS-232C上の非同期信号と無線上の同期HDLC信号との

図2 DCDでONになるコネクタの回路



変換を行うだけとなり、AX.25プロトコルはホスト・コンピュータ側において処理されます。

KA9Qマシンを無線側のゲートウェイにして、Macintoshをつなげないかなどと考えていたとき、小林直行氏の書かれた「Linux活用入門」<sup>(1)</sup>を見つけました。なんと、Linuxは初めからハムの無線IP接続が想定されています。さっそくDOS/VマシンにLinuxをインストールして実験してみると、うまく無線でIP接続ができます。

そこで、TNCをこれまで使ってきた1200bpsから東野電気の19200bpsTNC内蔵トランシーバ“PMT-192”に取り替えるてみると、リトライも少なく快適に使えます。実験を通して、無線IP接続の問題点や特徴などもわかってきたので、アマチュア無線PRのよい機会と、私の勤めている大学で開かれた第55回の情報処理学会全国大会に報告しました<sup>(2)</sup>。

無線でIP接続するためにLinuxをインストールしたのですが、すぐにLinux自体の素晴らしさにも気がつきました。今までワークステーションでしかできないと思っていたことが何でも可能です。世界中の多くの人により書かれたドキュメントも素晴らしく、すっかり気に入ってしまいました。

以下にLinuxによる無線接続の方法と実験を通して明らかになったことなどを紹介します。

## AX.25ドライバのインストール

DOS/VマシンにLinuxのインストールは行えるものと仮定します。私の場合、原稿を書いている時点での最新のものとしては雑誌Linux Japan (Vol.5)の付録のCD-ROM (Slackware 3.4)を使いました。これを用いるとLinux 2.0.30がインストールされますが、同じCD-ROMにLinux 2.0.31の圧縮されたソース・コードも入っているので、/usr/srcへ置きます。

なお、Linux (Slackware)をインストールする際に、ディスクNの選択でTCP/IPなどのネットワーク関係を、またディスクDの選択において無線ユーティリティのコンパイルの際に必要なncursesを選んでおきます。

無線でIP接続ができるようにするには、AX25-utilitiesと呼ばれるプログラムをインストールします。このプログラムの圧縮ファイルには、対応する開発版Linux (ver-2.1.\*)の番号が付されていて、安定版Linux (ver-2.0.\*)で使うには安定版のカーネルにパッチを当てる必要があります。

そこで、次の二つのプログラム (Linux 2.0.31用)をインターネットからダウンロードして、/usr/srcに置きます。

```
ax25-module-14e.tar.gz (187Kbyte)
ax25-utils-2.1.42a.tar.gz (346Kbyte)
```

私のホームページ

(<http://www.fit.ac.jp/~tanaka/ham-linux.html>)からもダウンロードできますので、インターネットに接続できる(Webの頁を見ることのできる)マシンで取り寄せ、フロッピーなどでインストールしたいマシンに移します。

準備ができれば、古いカーネル・ソースを取り除き、新しいカーネルソースを解凍してパッチを当てます。

```
$ cd /usr/src
$ rm -r linux
$ tar xvzf linux-2.0.31.tar.gz
$ tar xvzf ax25-module-14e.tar.gz
$ patch -p0 < ¥
> /usr/src/ax25-module-14/ax25-2.0.31-2.1.47-2.diff
```

次に、新しいLinuxのディレクトリに移って、カーネルの再構築の準備を行います。

```
$ cd /usr/src/linux
$ make menuconfig
```

ここで、AX.25に対応したカーネルとなるようにメニューを選びます。メニューでは、NETROMなどほかの無線ネットワークの仕組み組み込みますが、IP接続するだけなら不要です。必要に応じてPPPやイーサネット・ボードのドライバも組み込みます。メニューを選択する際、モジュール化しない方が設定が少し簡単になります。

```
Code maturity level options --->
[*] Prompt for development and/or
    incomplete
General setup --->
[*] Networking support
Networking options --->
[*] TCP/IP networking
[*] IP: forwarding/gatewaying
[*] Amateur Radio AX.25 Level 2
Network device support --->
[*] Network device support
[*] Radio network interfaces
[*] Serial port KISS driver for AX.25
```

これでカーネルの再構築の準備ができたので、次の

ようにメークを実行すると、新しいカーネルが生成され、古いものに置き替わります。

```
$ make dep
$ make clean
$ make zlilo bzImage bzlilo
    make zdisk
```

メニューでモジュールの選択を行った場合はモジュールもメークします。

```
$ make modules
$ make modules_install
    depmod -a
```

新しいカーネルで立ち上げ直してエラーがでる場合は、立ち上げ時に読まれる /etc/rc.d/rc.modulesなどの内容が元のままなので、エラーの内容を見て新しいカーネルにあうように修正します。

次にAX.25ユーティリティを解凍してインストールします。

```
$ cd /usr/src
$ tar xvzf ax25-utils-2.1.42a.tar.gz
$ cd ax25-utils-2.1.42a
$ make config
$ make
$ make install
$ make installconf
```

これで /etc/ax25/axports という名のファイルができるので、無線ポートのコールサインなどを設定します。私の場合、大学側の無線ゲートウェイは、次のように設定しています。パケット長とウインドウ・サイズは、コマンドでも変えることができます。

```
# name callsign speed packet window description
```



無線IPゲートウェイJA6AVG局(旧型ノートパソコンとPMT-192)

図3 実験で用いたIPアドレス(下2桁)

61.192	61.224	61.240	61.255
無線ネットワーク	ローカルネット	PPP 他	

radio JA6AVG-1 38400 512 8 438.3Mhz (19.2kbps)

## 無線ネットワークの構成

無線でIP接続を行うためにIPアドレスが必要になります。私の場合は、勤め先の福岡工業大学に割り当てられているアドレスの一部を無線接続の実験用に使わせてもらいました。

大学の幹線ネットワーク“172.20.1.0”<sup>(注)</sup>にはtiaraと名づけたマシン(Sun-4/5)がゲートウェイとなり研究室のサブネット“172.20.61.0”が接続されています。このサブネットには、既に多くのマシンが接続されていますが、まだ使用されていないアドレスの部分(192から上)を使わせてもらいました。

無線ネットワークの先にまた別のローカル・ネットをつなぐ実験なども行いたいので、利用できるアドレスの範囲を図3のように三つに分けて、192~223の32個のアドレスを無線ネットワーク用に、224~239の16個のアドレスを無線の先につなぐローカル・ネット用に、残りの240から上の16個をPPPな

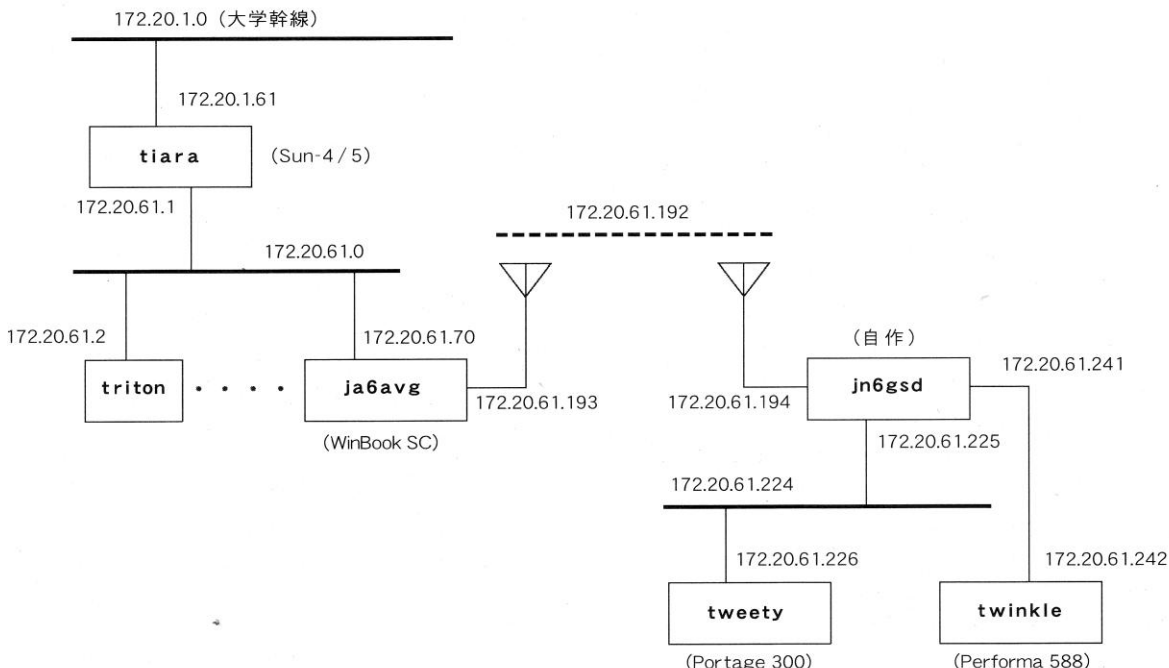
どに割り当てます。

小さなネットワークの作り方の書かれたものが見つからなかったので、こちらの思惑どおりにうまくネットワークが構成できるか、また、もとのサブネットのアドレスの範囲を172.20.61.0~172.20.61.191などと中途半端な範囲に変更できるのか疑問が残りますが、試してみるとところが実験の面白さです。

図4に、無線接続の実験に用いたネットワークの構成を示しています。サブネット172.20.61.0には無線へのゲートウェイとして“ja6avg”と名づけたノート・パソコン(486/33MHz, 8Mbyte)をPCMCIAのイーサネット・カードで接続して、アドレス172.20.61.70を与えます。無線部分も一つのネットワーク“172.20.61.192”をなし、“ja6avg”の無線ポートには“172.20.61.193”のIPアドレスを与えます。

型が古くなってメモリーもディスクも少なく、Windows95は重過ぎる、というノート・パソコンがLinuxをインストールすることで元気を取り戻し、ルーターとして活躍しています。無線ネットワーク“172.20.61.192”に対して、無線ポートのIPアド

図4 ネットワークの構成



(注) IPアドレスの上2桁“172.20”はクラスBのプライベート・ネットワーク(RFC1918)用のもので、実際とは異なります。



図5 ja6avgの /etc/rc.d/rc.inet1 ファイル

```

#!/bin/sh
/sbin/ifconfig lo127.0.0.1
/sbin/route add -net1 27.0.0.0 netmask 255.0.0.0 lo
IPADDR="172.20.61.70"           (ja6avgのイーサインターフェース側IPアドレス)
NETMASK="255.255.255.0"
NETWORK="172.20.61.0"           (tiaraをゲートウェイとするサブネット)
BROADCAST="172.20.61.255"
GATEWAY="172.20.61.1"           (tiaraのサブネット側IPアドレス)
/sbin/ifconfig eth0 ${IPADDR} broadcast ${BROADCAST} ¥
                        netmask ${NETMASK}           (インターフェースにアドレス等設定)
/sbin/route add -net ${NETWORK} netmask ${NETMASK} eth0
/sbin/route add default gw ${GATEWAY} metric 1       (ゲートウェイはtiara)
                                                (ここまでは netconfig コマンドで自動的に設定される)

# ax25-port
/usr/sbin/kissattach -i 172.20.61.193 -m 512/dev/ttyS0 radio
                                                (AX.25インターフェースを作ってアドレス61.193に)
/usr/sbin/kissparms -p radio -t 100 -s 10 -r 255 (AX パラメータを設定して)
/sbin/ifconfig ax0 netmask 255.255.255.224 broadcast 172.20.61.223
                                                (AXポートにネットマスクとブロードキャストを)
window 4096 ax0                               (無線ネット 61.192 は AXポートへ)
/sbin/route add -net 172.20.61.192 netmask 255.255.255.224 ¥
gw 172.20.61.194 window 4096 ax0             (ローカルネット 61.224 も AXポートへ)

```

レス“172.20.61.194”を持つ自作マシン“jn6gsd” (Cyrix p200+, 64Mbyte)が無線接続されます。このマシンには、PPP接続のMacintoshの“twinkle” (Performa588)と、ローカル・ネット“172.20.61.224”が接続され、ローカル・ネットにはLinuxをインストールしたノート・パソコン(東芝Portage 300)が接続されています。

## ネットワークの設定

Linuxをインストールしてnetconfigコマンドを実行すればイーサネットなどの通常のネットワーク設定を行うことができます。このコマンドでホスト名、IPアドレス、ドメインネーム、インターネットへのゲートウェイ、ドメインネーム・サーバーなどが設定されます。おもな設定は、ファイル /etc/rc.d/rc.inet1 に書かれています。元となるネットワーク“172.20.61.0”で使用するアドレスの範囲をネットマスクで0~191の範囲に設定できそうにないので、そのままにして、無線ポートの設定を追加します(図5)。

図5の“# ax25-port”より前の部分は、Linuxのインストール時か、netconfigコマンドを実行したときに自動的に作成されます。後の部分が無線ポートに関する設定です。

コマンド kissattachによりAX.25インターフェースが作成され、シリアル・デバイス /dev/ttyS0に接

続されてIPアドレス172.20.61.193が与えられます。パラメータの“-m 512”はAX.25パケットの最大データ長です。この行の最後の“radio”は /etc/ax25/axportsにおいて、行の最初に指定した文字列に対応します。

次のkissparmsコマンドの“-t 100”は、TNCがトランシーバを送信に切り替えてから、データをトランシーバに送出し始めるまでの時間TX delayを100msに設定しています。使っているTNC一体型トランシーバ(PMT-192)では、80msに設定するとパケットの往復が40msほど短縮されますが、数%~20%くらいのパケット・ロスが起きます。安定に使えるのは100msまでのようです。

スロット・タイムは一応10msに、パーシストは100%(255)に設定していますが、無線ポートのマルチアクセスや、同一チャネルを多数の局で使う場合には最適な値に設定し直すことが必要になります。

次のifconfigコマンドで無線ポートのインターフェースax0のネットマスクとブロードキャストを設定しています。また、routeコマンドでインターフェースax0の先に無線ネットワーク61.192があること、ホスト61.194をゲートウェイにしてネットワーク61.224があることを指定します。マシンを立ち上げ直して、コマンド“/sbin/ifconfig -a”や“/sbin/route -n”を実行してみると、インターフェースがうまくできているか、設定がルーティング・テーブルに反映されたかどうかわかります。

図6 jn6gsdの /etc/rc.d/rc.inet1 ファイル

```
#!/bin/sh
/sbin/ifconfig lo127.0.0.1
/sbin/route add -net127.0.0.0 netmask 255.0.0.0 lo
IPADDR="172.20.61.225"          (jn6gsdのイーサインターフェース側アドレス)
NETMASK="255.255.255.240"      (5bit以上はネットワークアドレス)
NETWORK="172.20.61.224"        (ローカルネットワーク)
BROADCAST="172.20.61.239"      (下位4bitはすべて1)
/sbin/ifconfig eth0 ${IPADDR} broadcast ${BROADCAST} ¥
                               netmask ${NETMASK}          (イーサインターフェースのアドレス等)
/sbin/route add -net ${NETWORK} netmask ${NETMASK} eth0
                               (ローカルネット61.224はイーサインターフェースへ)

# ax25-port
/usr/sbin/kissattach -i 172.20.61.194 -m 512/dev/ttyS0 radio
                               (AX.25インターフェースを作ってアドレス61.194に)
/usr/sbin/kissparms -p radio -t 100 -s 10 -r255 (AXパラメータをセット)
/sbin/ifconfig ax0 netmask 255.255.255.224 broadcast 172.20.61.223
                               (6bit以上はネットワークアドレス)
/sbin/route add -net 172.20.61.192 window 4096 ax0
                               (無線ネット61.192はAXポートへ)
/sbin/route add default gw 172.20.61.193 window 4096 ax0
                               (デフォルトのゲートウェイはja6avg)
```

図7 tweetyの /etc/rc.d/rc.inet1 ファイル

```
#!/bin/sh
/sbin/ifconfig lo127.0.0.1
/sbin/route add -net127.0.0.0 netmask 255.0.0.0 lo
IPADDR="172.20.61.226"          (ローカルネット上のアドレス)
NETMASK="255.255.255.240"      (5bit以上はネットワークアドレス)
NETWORK="172.20.61.224"        (ローカルネットワーク)
BROADCAST="172.20.61.239"      (下位4bitはすべて1)
GATEWAY="172.20.61.225"        (ゲートウェイはjn6gsd)
/sbin/ifconfig eth0 ${IPADDR} broadcast {BROADCAST} ¥
                               netmask ${NETMASK}          (イーサインターフェースを設定)
/sbin/route add -net ${NETWORK} netmask ${NETMASK} eth0
/sbin/route add default gw ${GATEWAY} metric 1          (ルーティング設定)
```

同じく、ローカル・ネットの無線ゲートウェイとなる“jn6gsd”もIPアドレスなどをあらかじめnetconfigコマンドで設定しておいて、多少モディファイして無線ポートの設定を追加します(図6)。

ローカル側の有線サブネットにつながるマシン

“tweety”は、無線ポートがないのでnetconfigコマンドによる設定だけで十分です(図7)。利用できるネームサーバーのアドレスなども正しく設定しておきます。

大学の幹線に接続されるマシン“tiara”(SunOS-



4.1.4)は、サブネット172.20.61.0を持ち、多くのマシンが接続されています。これら多くのマシンの設定は、できれば変更したくないなと思いながら、ネットマスクなどには触らずに、tiaraの“/etc/rc.local”に次の1行を追加して立ち上げてみると、61.192以上のアドレスを無線のゲートウェイja6avg(61.70)へとうまくルーティングしてくれます。

```
/usr/etc/route add net 172.20.61.192
172.20.61.70 1
```

しかし、この設定ではブロードキャストの問題があり、無線ゲートウェイja6avgの電源を切ると、tiaraも動かなくなる現象が生まれました。これではネットワーク“172.20.61.0”に迷惑をかけるので、次のように実際に無線ポートを介してつながるホストのみをtiaraの/etc/rc.localに書いて、個別にルーティングすることにしました。

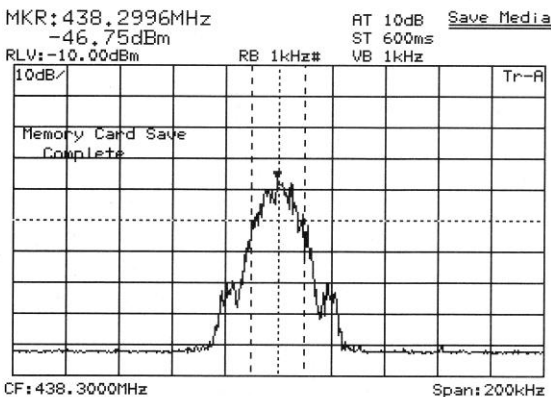
ネットワーク172.20.61.0上の他のホストで無線側とルーティングしたいマシンも同じように設定しておきます。

```
/usr/etc/route add host 172.20.61.194
172.20.61.70 1
/usr/etc/route add host 172.20.61.226
172.20.61.70 1
```

## TNCの設定

ここで用いたTNC内蔵トランシーバは、普通のトランシーバのようなツマミはなく、周波数の設定や転送速度の設定なども拡張された“TNCコマンド”により行います。普通のパケット通信に使う分にはすべての操作がキーボードから行えて便利ですが、IP接続の

図8 19200bpsの変調波のスペクトラム



ために一旦、KISSモードに設定してしまうと、コマンドモードに戻すまでは周波数の変更ができなくなります。私の場合は次のように設定しています。

```
cmd: ab 38400 (TNC コンピュータ間の転送
速度を38.4kbpsに)
cmd: hb 19200 (無線の転送速度を19.2kbpsに)
cmd: freq 3830 (無線の周波数を438.30MHzに)
cmd: kiss (TNCをKISSモードへ)
cmd: restart (設定を有効に)
```

変調方式はGMSKが使われています<sup>(3)</sup>。通信速度は19200 bpsで、従来用いられてきた9600 bpsの2倍と速くなっています。実際にどのくらいの帯域を占有しているか気になるので、スペクトラム・アナライザで分解能を1 kHzに設定して調べたところ、図8のように中心周波数のほぼ±20kHzに側波帯が広がっていることがわかりました。

## 無線IP接続の実験

以上の設定で、無線IP接続ができ、ローカル・ネット上の“tweety”からでも「ネットスケープ」を使って世界中にアクセスできます。もし、うまく接続ができないときは、“traceroute <相手アドレス>”コマンドを使えば、どこでルーティングに失敗しているか見当がつきます。

また、無線サーバー上で“listen -a”を実行しておけば、無線ポートを通るパケットがモニターできるので、正しくパケットが送られているかどうかわかります。ネットワークの設定を間違えて二つの無線ポート間でIPパケットのピンポンをさせると、ttlの値が1ずつ下がり、やがてパケットが失われる様子もわかります。

福岡市は、図9に示すように博多湾に面して広がっています。無線IP接続を行った福岡工業大学と修猷館高校との間にはダイエーのドーム球場があって見通しを遮るため、回折波か反射波を利用することになります。文化祭の会場にはビーム・アンテナを準備する時間がなく、とりあえず物置から探し出した真空管(6939)時代のタクシー無線のアンテナ(GP)を使いました。これを会場となった3階の窓際において、コマンド“ping 172.20.61.193”を実行すると、1秒ごとにja6avgあてのパケットが出ますので、位置を調整してパケットの通りのよい場所を見つけました。

回折波や反射波が複雑に干渉するためか、アンテナを置く最良の位置が不安定で、パケットの通りが悪くなる度に、位置の調整をやり直すことが必要になった

のですが、文化祭の会場は無線(無銭)でインターネットが使えるということで、テレビのニュースや、新聞に取り上げられるなど大盛況となりました。

文化祭の後、ほぼ大学から等距離にある草香江の自宅(中央区)と大学間でIP接続の実験を続けたのですが、自宅も同じビルに囲まれてロケーションが悪く、パケットの通り具合が変動します。それで音声通信用に使ってきた自作のGP(無指向性アンテナ)をやめて、5エレ八木に変えたところ、見違えるほどパケットの通りがよくなりました。

Sメーターでは電波が強く来ている、例えば、テレビを室内アンテナで見たときのように、マルチパスにより信号の波形が歪んでいたことが考えられます。TCPは再送が重なると送信の間隔をどんどん長くするため、pingコマンドを実行してパケットが2割以上失われるような状況では快適な通信は期待できません。パケットのロスが1割以下となるように、まず、電波の通路を整備する努力が必要なようです。

## 実験を通してわかったこと

TNCを利用した無線接続は、モデムによる有線接続と比べていくつか不利な点があります。

第1は、一組のトランシーバを用いる限り、必然的に半二重通信となる点です。二組のトランシーバを用いれば全二重通信も可能ですが、送受信電力の差が大きいため二つの周波数帯を使うことが必要になります。ただ、二組のトランシーバを用いて全二重にしても、電波を出しっぱなしにしない限り、TX delay(送受信切り替え時間)の問題が残ります。

先の「ネットワークの設定」で記した設定を行うと、pingコマンドでは400ms前後でパケットが無線回線を往復します。この時間遅れのため、エコーバックされる文字が重要な働きをするリモートのスクリーン・エディタは使い辛くなります。またX-Windowの



高校文化祭会場のJN6GSD局(田中美帆)

ターミナル・ウィンドウはリモートよりもローカルに開くほうがデータ転送量が少なく済みます。

第2は、転送速度の問題です。FTPでファイルの転送を行うと、回線の状況がよければ、先のネットワークの設定で、1.0~1.6kbyte/secの転送速度が得られます。kissattachコマンドのパケットのデータ長とrouteコマンドのウィンドウ・サイズ(確認を待たずに先送りできるデータ長)をいくつか取り替えてFTPの転送速度を調べたものを図10に示します。

転送速度は、データの内容によっても変わりますの

筆者JA6AVGのインターネット接続は、昨年RKBNEWS('97.6.6)でも取り上げられテレビでその様子が報じられた。

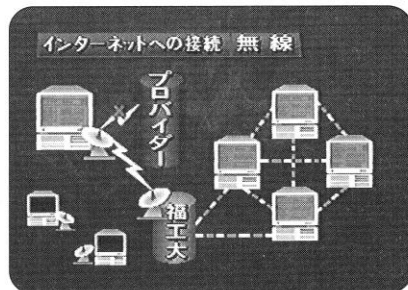
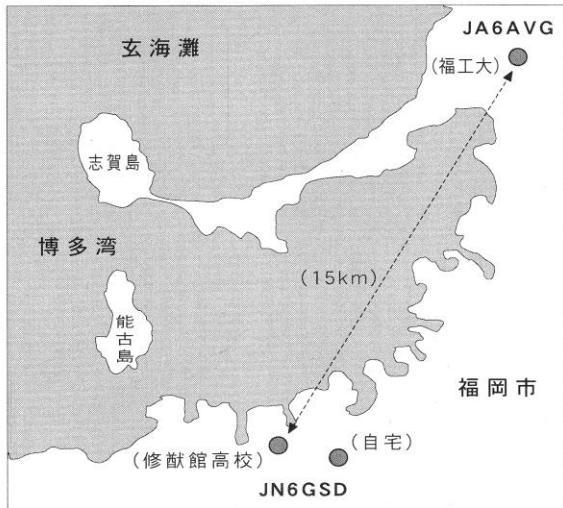


図9 無線IP接続の実験



で、精密な測定ではなく、およその傾向を調べたものです。FTPではデータフローが一方に偏るため、ウィンドウ・サイズを大きくすれば、TNCの送受信切り替え回数も少なくなり、19200 bpsの限界に近い転送速度が得られて、半二重通信の問題は、軽減されていることがわかります。この転送速度は、最近の有線モデムと比較すると、速いとは言えませんが、同じくデータフローが一方に偏るWWWのブラウジングにも有効です。Webを見るときはボトルネックが必ずしも直接にモデムをつなぐ電話線ではないことを考えれば十分使えます。また、10Mbyteを超えるNetscapeの圧縮ファイルでも、数時間ほっておけば電話代を気にすることなく転送してくれます。

第3は情報漏洩の問題です。一般の感覚からすると、「アマチュア無線の性格上、暗号が使えないのでメールもパスワードも丸見えとなり、プライベートな情報の転送には向かない」となります。しかし、自分でネットワークを使いながら、ほんとに秘密にしておきたい情報は少ないように思えます。メーリングリストのメールは皆に読んでもらおうと書き、メールボックス以外のUNIXファイルはデフォルトのまま「見ていいよ」にしています。また、ホームページは、できるだけ多くの人に見てもらおうと作ります。

アマチュア無線家なら自分のしゃべることは誰でも聞けることを知っており、ネットワークを情報放送の手段だと思えば、情報漏洩は問題ではなくて長所に変わる可能性さえあります。

表1にインターネットの主なアプリケーションと無線接続の問題点をまとめてみました。

例えば、telnetは、インタラクティブなコンピュータの遠隔利用の観点からみたもので、情報の転送量が少なく、TNCの通信速度は問題にならず○となる

が、送受信の切り替え速度が問題となり、スクリーン・エディターなど使い心地が悪く×となります。また、一般の感覚からするとパスワードが見えるので情報漏洩の点から×となります。

ftpは、大量のデータを転送するにはスピード不足で△であるが、送受信切り替え速度は問題にならず○となります。ftpは一応パスワードが要るが、anonymous ftpを利用する限りは不要なので△となり、WWWは、通信速度の点から△となるが、切り替え速度は問題にならず○、広告と同じように自ら広まることを望む情報なので情報漏洩の点は◎となります。

## Web, POPとも 実用になる!?

ここで説明に用いたIPアドレスは、実際の実験に用いたものではなく、上位2桁を自由に使えるプライベート・アドレス“172.20”に置き替えています。まだ試していないのですが、MasqueradeやNAT (Network Address Translator)と呼ばれて、プライベート・アドレスからアドレス変換によりインターネットにアクセスする方法もあるようです (RFC1631)。

無線ゲートウェイ“JA6AVG”にこの機能を持たせれば、無線ネットワークの構成はより自由に行えることとなります。

無線IP接続の実験を開始して、そろそろ1年になります。自宅は電話用に使ってきたトランシーバがTNC内蔵トランシーバ (TNCトランシーバ) に替わり、無指向性アンテナがビーム・アンテナに替わりました。無線接続ではNetscapeによるWebのブラウジングをいちばんよく利用しています。

自宅ネットは、常時電源を入れているわけではないので、直接的なメールの受信には向きません。そこ



JA6AVGのアンテナとロケーション。  
通信先はドーム球場など建物の向こう側、距離15kmで直視できない(見出し外伝搬)。

図10 FTPの転送速度(およその傾向を示す)

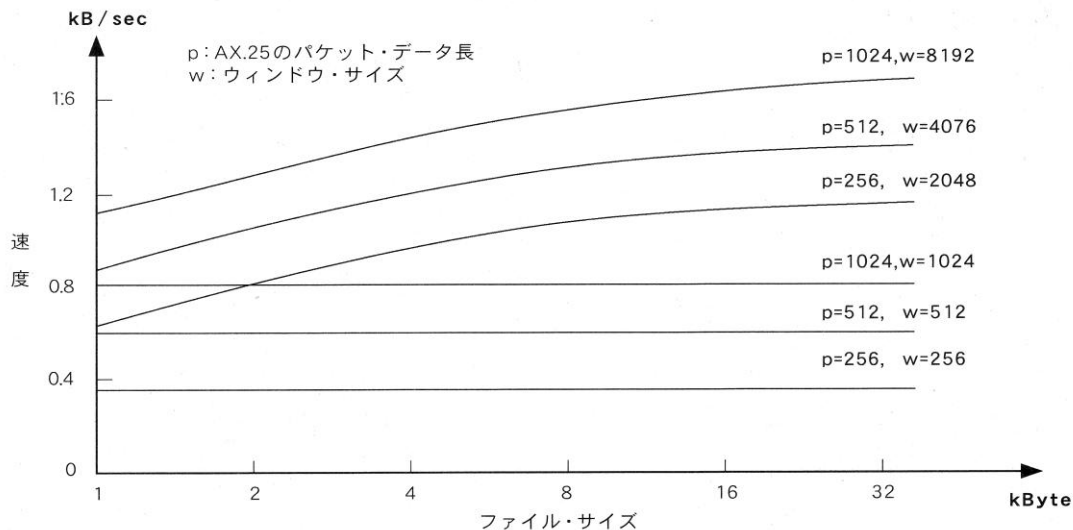


表1 アプリケーションと問題点

	通信速度	切替速度	情報漏洩
telnet	○	×	×
ftp	△	○	△
mail	○	○	×
WWW	△	○	◎

で、大学の幹線につながるマシン“tiara”でメールを送受信できるように設定して、tiaraにpopサーバーをインストールして配信しています。自宅からメールを送受信する場合は、Netscapeのメール機能を使ってこのpopサーバーにアクセスします。

使用しているTNC一体型トランシーバは、ネットワークの使用中でも、コンピュータが信号を出していなければ電話(FM音声通信)としても使えます。この機能はたいへん便利で、音声で連絡をとりながら、IP接続の実験を進めることができました。

一方、このトランシーバはTNCの部分にKISSモ-

ードに設定していることもあって、簡単に周波数を変えることができないので、通常の電話用アマチュア無線機としては使い難く感じます。ユーザーの希望として、次にデジタル専用の無線機器を開発される場合は、ぜひ1200MHz帯で19.2kbpsの4倍以上のスピードが出せ、TX delayを10msec以下に設定できるような機器を期待しています。

高速伝送の無線機器が手軽に利用できるようになれば、やがてレピータのように日本中いたるところにアマチュア無線によるインターネット・ゲートウェイができたりして、楽しいことになるかも知れません。

■参考文献■

- (1) 小林直行「Linux活用入門」, CQ出版, 1997.
- (2) 田中卓史「アマチュア無線を利用したインターネット接続」, 情報処理学会第55回全国大会講演論文集, pp.3-783-764, 1997.
- (3) 秦正人, 山内雪路(編)「バケット無線ハンドブック」, CQ出版, 1987.
- (4) Terry Dawson, "Linux AX25-HOWTO, Amature Radio", <http://sunsite.unc.edu/LDP/HOWTO/AX25-HOWTO.html>, 1997