09/23/96

Howto Setup Hayes Compatible Modem on RISC System/6000

Special Notices

Information in this document is correct to the best of our knowledge at the time of this writing. Please send feedback by fax to "AIXServ Information" at (512) 823-4009.

Please use this information with care. IBM will not be responsible for damages of any kind resulting from its use. The use of this information is the sole responsibility of the customer and depends on the customer's ability to evaluate and integrate this information into the customer's operational environment.

About This Document . . . 1
An Overview of Setting Up a Hayes Compatible Modem . . . 1
Cabling . . . 1
Configuring the TTY . . . 6
Modem Configuration . . . 8
  Setting up Using cu . . . 9
Setup Through motalk.c . . . 12
About motalk.c . . . 12
Description . . . 12
Prerequisites . . . 12
Troubleshooting Tips . . . 16
Appendix . . . 16
  Program 1 . . . 16
  Modem Programs: . . . 17
  USRobotics: . . . 18
  IBM: . . . 18
  Hayes: . . . 18
  UDS: . . . 18
  Telebit: . . . 18
  Other Modems: . . . 18
Reader's Comments . . . 20

About This Document

The following document provides an overview of setting up a Hayes Compatible Modem for the RISC System/6000 and is applicable to AIX versions 3.2 and 4.1.

An Overview of Setting Up a Hayes Compatible Modem

To setup a modem for dialing in/out of the RISC can be divided into three basic steps:

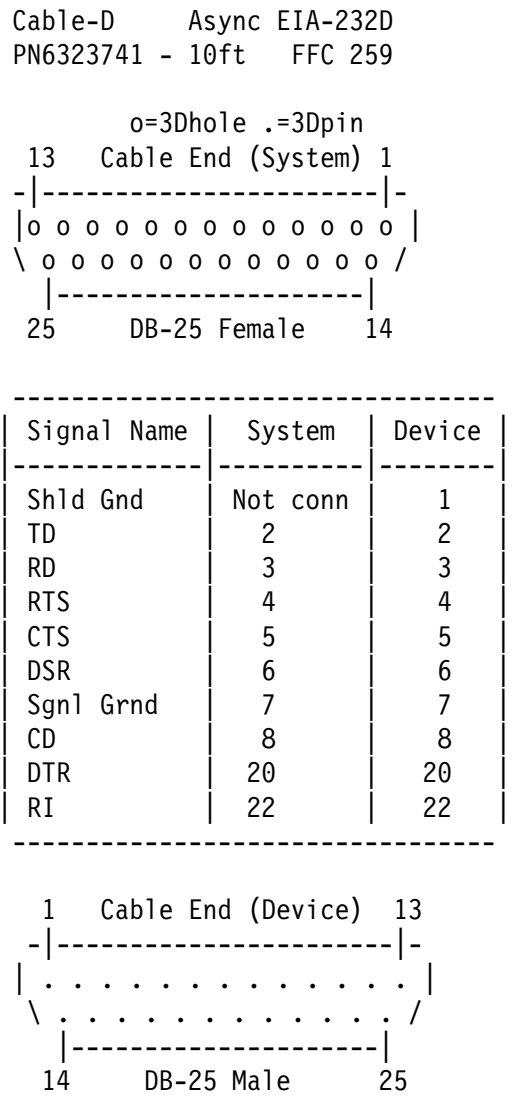
- The installation of the hardware on the RS/6000. This includes such things as proper cabling,etc.
- The configuration of the TTY device.
- The configuration of the modem itself.

These steps are expanded into detail. Once these steps have been followed, there should not be a problem with your modem connection.

Cabling

Connecting your modem to the rs/6000 should not be a problem. Most modems use the EIA-232 standard for their connection to the system. Here is a list with specs for the different adapters found in the AIX environment.

Use IBM cable D (DB25 to DB25) or equivalent to connect your modem to a native serial port, 8 or 16 port async adapter. Its pinouts are:


Cable-D Async EIA-232D
PN6323741 - 10ft FFC 259
o=3Dhole .=3Dpin
13 Cable End (System) 1
25 DB-25 Female 14
Signal Name | System | Device
Shld Gnd | Not conn | 1
TD | 2 | 2
RD | 3 | 3
RTS | 4 | 4
CTS | 5 | 5
DSR | 6 | 6
Sgnl Grnd | 7 | 7
CD | 8 | 8
DTR | 20 | 20
RI | 22 | 22
1 Cable End (Device) 13
14 DB-25 Male 25

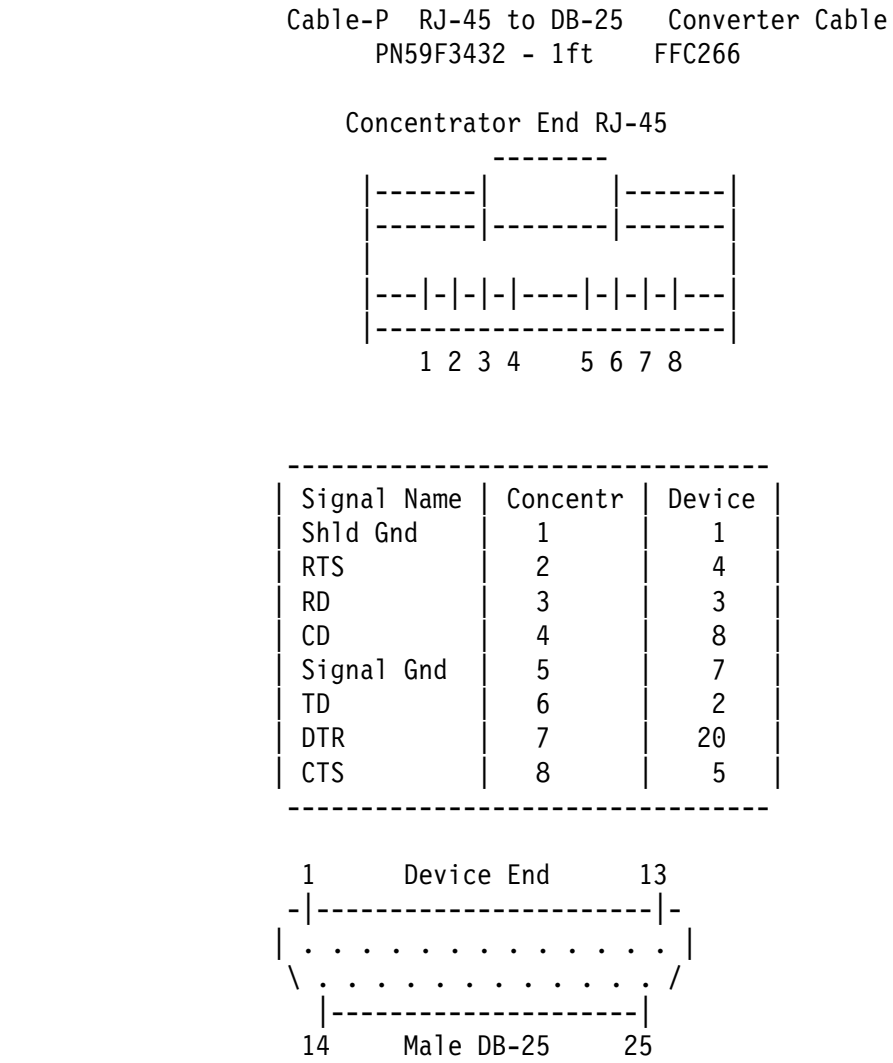
To attach a modem to a native serial port on a RS/6000 with a 10-pin modulo adapter, use IBM cable B or equivalant:

```
Cable-B   Adapter Modu-to-DB25 S1/S2
      PN00G0943 - .33ft   FFC 252

      Cable End (System)
          5      1
       --|-------|--
      |  # # # # #  |
       | # # # # # |
        -|-------|-
        10       6
```

Signal Name	System	Device
Shld Gnd	Not conn	1
TD	1	2
RD	6	3
RTS	3	4
CTS	8	5
DSR	7	6
Reserved	5	7
DTR	2	20
RI	4	22
DCD	9	8
Signal gnd	10	7

To attach a modem to a 64 port adapter:

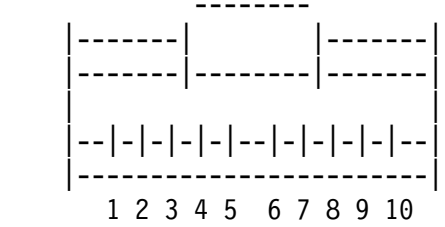


If you are attaching the modem to a 128 port adapter, the following cable spec can be used. It is Cable NK, and its 8 wire equivalent.

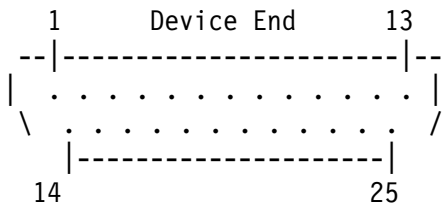
Cable NK 10-pin RJ45-to-DB25 Modem Cable

Remote Async Node End

RJ45 10-pin



8pin RJ45	10pin rj45	Signal Name	Device DB25
	1	RI	22
1 --	2	DSR*	6
2 --	3	RTS	4
3 --	4	Chassis Gnd	Shell
4 --	5	TD	2
5 --	6	RD	3
6 --	7	Signal gnd	7
7 --	8	CTS	5
8 --	9	DTR	20
	10	CD*	8



Note: If you are using the 8 wire connection, you need to be aware that the DSR signal will not be used. With this in mind, you will connect pin 1 to pin 8. Also, you need to enable altpin through smit, or by executing

```
#/usr/lbin/tty/stty-cxma -a altpin tty#
```

Configuring the TTY

Log into the system as root and issue the command

```
#smit tty
```

You will see a screen somewhat like the following:

```
        TTY

Move cursor to desired item and press Enter.

 List All Defined TTYs
 Add a TTY
 Move a TTY to Another Port
 Change / Show Characteristics of a TTY
 Remove a TTY
 Configure a Defined TTY
 Generate Error Report
 Trace a TTY
```

Select 'Add a TTY' to add a new tty device. If you already have a tty defined, then check the entries following with your tty definition.

Since the Parent adapters will differ from system to system, we will concentrate on some of the common characteristics. Only the changes that need to be made from the defaults will be shown here.

```
   Add a TTY

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                    [Entry Fields]
*PORT number                        []      +
 BAUD rate                          [9600]  +
 PARITY                             [none]  +
 BITS per character                 [8]     +
 Number of STOP BITS                [1]     +
 XON-XOFF handshaking               no     +
 RTS-CTS handshaking                yes    +
 TERMINAL type                      []
 Enable LOGIN                       disable +
```

The PORT number is important at this point. By viewing the list(through the F4 key or hitting the list button), select the appropriate port where the modem is physically connected at.

The BAUD rate should be the highest DTE baud rate which the modem can support. You will have to consult your vendor or modem manual for this info. As a rule of thumb, if the modem is advertised as 14.4 or 28.8, you might be safe in selecting the 38400 baud rate.

If your modem does not support auto baud negotiation, you can still setup a port to allow callers to dial in at multiple baud rates by setting up the port for multiple baud rates. Note that the baud rates entered MUST be in decreasing order from left to right and the number of entries is the baud rate field MUST match the number of entries in the fields PARITY, BITS per character, and number of STOP BITS.

```
BAUD rate                [9600,2400,1200]
PARITY                   [none,none,none]
BITS per character       [8,8,8]
Number of STOP BITS      [1,1,1]
```

Unless your specific use of the modem requires otherwise, the PARITY, BITS per second, and STOP BITS should be left to the defaults.

Handshaking with modems should be Hardware(RTS/CTS), set to yes, and XON/XOFF set to no. This is recommended for file transfers and uucp connections. Do note that for these options to be displayed through smit, you need to make sure that you have APAR IX40410 installed in your system, otherwise refer to Program 1 at the Appendix of this document.

If you will be using XON/XOFF flow control, then you need to be aware of the Transmit Buffer Count entry.

```
| Transmit buffer count             16                  +
```

This value indicates how many bytes the RS/6000 will send out before checking for an Xoff character. Tweaking this value by decreasing/increasing might be to your advantage when experiencing data loss.

The TERMINAL type can be user defined. This entry is used when users dial into this tty, and the environmental variable TERM is set to this value.

The Enable LOGIN entry has four options:

- enable: A getty process is started on the port on bootup or after the device is created. When the carrier detect signal is raised high by the modem, a login herald is sent to the modem. This option is used if users are dialing into the RS/6000 only. It is not possible to dial out if this option is selected.
- disable: No getty process is started on the port. If the modem is set to auto answer, it most likely will not answer because the RISC is not asserting DTR. If the modem or cabling has DTR forced high, the modem will answer, but no login banner will be displayed. This option is selected for dialing out from the RS/6000 only.

- share: A getty process is started on the port; however, no lock is grabbed on the port until carrier detect is raised high. With share, users can dial into the RS/6000 and login, and when no one is logged in, users can dial out of the port without making any changes to the TTY. Note that this option may not function properly at levels of AIX earlier than 3.2.3e unless PTF U412752 is installed.
- delay: Similar to share except that not only must carrier detect be present before a lock is grabbed on the port, but a carriage return must also be received on the tty before a login herald is displayed. This option is useful in cases where carrier detect is strapped high by cabling, or the modem itself. If you are having difficulty with share, try delay. To the caller, the only noticeable difference is the need to press enter before seeing the login herald.

For a 128 port Adapter, note the following specific entries:

```
Force Carrier                          disable      +
Perform Cooked Processing in Adapter   enable       +
Use Alternate RJ-45 Pinouts            disable      +
```

These should be left as is, unless:

- An 8 wire cable is being used, then set Alternate RJ-45 to enable
- You are experiencing problems when connecting to this tty, set Cooked Processing to disable

This completes the changes to the TTY settings in SMIT for the configuration of the modem. The next step involves configuring the modem itself. The settings are generally available on most Hayes compatible modems, however, some modems have variations of these command. Also, some have dip swithes that control the same configurations as through software setup, which overrides any software commands issued.

Modem Configuration

A modem should be programmed with the following minimal settings for a successful connection to the rs/6000:

- The DTE (modem to RS/6000) baud rate should remain fixed on the last AT command sent.
- The Carrier Detect signal (modem to RS/6000) should be set to be on when a modem to modem connection is established.
- The Data Terminal Ready signal (RS/6000 to modem) should be set for the modem to accept DTR from the RS/6000.
- The DCE (modem to modem) baud rate should be set to a variable state.

You might have to refer to your modem manual or vendor for the specific command strings or dip switch settings for the above descriptions.

For programming modems, there are two methods available through AIX:

- The first is to use the BNU utility cu. You must have bosext1.uucp.obj installed on your machine to have the cu command.
- The second way is to use a C program, motalk.c.

Setting up Using cu

In order to use cu for the configuration of your modem, you need to update one configuration file. In /etc/uucp, there is a file called Devices. In this file, add the following entries with vi or your favorite text editor:

```
ACU tty# - <Baud> hayes
Direct tty# - <Baud> direct
```

where tty# is the tty device added through smit, and <Baud> is the BAUD rate defined through smit for this tty.

Now, at your command prompt execute:

```
#cu -dml tty#
```

Note If you get a usage error, you have an older version of cu, pre 3.2.4 and you need to install APAR IX24051. This is for the -m option flag.

Other than a successful message, you might get two other errors

- NO DEVICES AVAILABLE Check the entries in the /etc/uucp/Devices file. If something is wrong with the entries, this error message will be displayed.
- CAN'T ACCESS DEVICE Check that nothing is running on the port. Make sure you have the getty disabled by using 'pdisable tty#' command.

You will get a succesful 'Connected' message after executing the command. The output should look something like this:

```
   altconn called
   Device Type Direct is wanted
   ttylock tty1 succeeded
   Attempting to open /dev/tty1
   fixline(7, fixline - direct
   Changing tty c_cflags from 03206275
   Changing tty c_cflags to   03206275
   gdial(direct) called
   Changing tty c_cflags from 03206275
   Changing tty c_cflags to   03206275
   getto ret 7
   The status of device fd is 7.
F_GETFL=2,iflag= 3005',oflag= 0',cflag= 3206275',
lflag= 0',cc[0]= 3',[1]= 34'
[2]= 10' , [3]= 25' , [4]= 1' ,
[5]= 0' , [6]= 0' ,
[7]= 21' , [8]= 23' , [9]= 3
2' , [10]= 31' , [11]= 22' ,
[12]= 17' , [13]= 27' ,
[14]= 26' , [15]= 0' ,
call _mode(1)
Connected
_receive started
transmission started
```

At this point, you can program your modem directly. The simplest AT Command string to send should be used to verify the connection:

```
AT
```

You should see the transmit/receive data indicators on the modem flicker and it should respond with "OK". If it does not, verify that your modem is on and is connected to the port that you have the tty defined at. Make sure that you are using a cable matching IBM's cable D and not a null modem cable (no interposer).

If you do receive OK, you are connected to your modem and can set it up as required by AIX. Send the modem the setup string:

Note: As mentioned above, you might want to check with your specific modem vendor/support group for the correct setup string to send to your modem. The following is and example which has worked with most AT-Compatible modems.

```
ATE0Q2&D2&C1S0=3&W
```

- AT Informs the modem that the rest of the line is a command to it.
- E0 Turns echo off. After this command takes effect, you will not b e able to see what you are typing, but the modem will still be accepting your input. Note that this option is not necessary if &C1 is set, but is encouraged to be set.

- Q2 Turns result codes off for incoming calls but leaves them active for outgoing calls. If you dial into the RISC and receive a message that your login was entered in all capitals, this change should fix the problem.
- &D2 Instructs the modem to hang up when the RISC drops DTR.
- &C1 Instructs the modem only to assert the carrier detect (DCD) signal when the modem is connected with another modem. Common signs that this setting is incorrect include dialing into the RISC, connecting, and never receiving a login. Another common sign of this is to see a "Init: Command is respawning too rapidly" error on the console, although the respawning error is also commonly caused by faulty cabling.
- S0=1 S0 (zero not the letter o) is a register containing the ring number to answer the phone on. To prevent a modem from answering, set S0=0.
- &W Writes the settings to the modems NVRAM, so that the above changes are not lost when the modem is powered off. Note that if you try cu -ml tty# a second time and the above changes are no longer in place, there is a good chance that your problem is with the modem hardware (ie dead battery).

After making these changes, end the session with

```
~.
```

Note: In the Appendix, there is sample AT strings for other modems that are being used. Check the appendix to see if your particular modem is listed. Otherwise, contact your vendor/support center for assistance in the modem commands to follow the prescribed settings above.

You will see "connection has ended" and be back to the command prompt. If you temporarily had to pdisable tty# to use cu, then you should set it back to the setting you wanted it on (enable, delay, share, etc) either via SMIT, or with the commands (as appropriate):

```
penable tty#
pdelay  tty#
pshare  tty#
```

If you have completed the steps above, you should be able to receive incoming calls on an enabled/delayed/shared TTY. If the port is set to disable/share/delay you should be able to dial out. To dial out with cu use the syntax:

```
cu <phone_number>
```

Setup Through motalk.c

If you are unable to program your modem via cu, then an alternate method is to use a simple C program to program your modem or dial out with. The program has very limited functionality, but if you need a simple way to program your modem to properly handle incoming calls, this program will work. Note that you need the C compiler installed in order to compile the below program. To see if you have the C com- piler installed, try the command "cc" to see if you receive a usage message (indicating it is installed) or a "not found" message indicating that the compiler is not installed on your machine.

Note: Please use this information with care. IBM will not be responsible for damages of any kind resulting from its use.

About motalk.c

The enclosed C language source code has been tested on AIX versions 3.2 and 4.1.. Please be aware that it contains page breaks and other document formatting. These will need to be edited from the document before compiling.

Description

The motalk program allows the user to easily connect to a modem port for programming and does not require any customization to the "/usr/lib/uucp/ Devices" file as with the cu utility.

This C program is a modification of the original motalk program available for AIX version 3.2.

Prerequisites

The modem should be powered "on" and physically attached to the tty port using appropriate cabling.

```
*/
/*********************************************************/
/*  motalk_41_32.c     Modem setup program                */
/*                                                        */
/*     Compile:  cc -o motalk motalk_41_32.c              */
/*       Usage:  motalk /dev/tty? [speed]        */
/*********************************************************/
#include <errno.h>
#include <stdio.h>
#include <signal.h>
#include <fcntl.h>
#include <termio.h>
```

```
FILE   *fdr, *fdw;
int    fd;
struct termio term_save, stdin_save;
void Exit(int sig)
{
        if (fdr) fclose(fdr);
        if (fdw) fclose(fdw);
        ioctl(fd, TCSETA, &term_save);
        close(fd);
        ioctl(fileno(stdin), TCSETA, &stdin_save);
        exit(sig);
}
main(int argc, char *argv[ ])
{
        char   *b, buffer[80];
        int    baud = 0, num;
        struct termio term, tstdin;
```

```
        if (argc < 2 || !strcmp(argv[1], "-?")) {
                fprintf(stderr,
                "USAGE: motalk /dev/tty? [speed]\n");
                exit(1);
        }
        if ((fd = open(argv[1], O_RDWR|O_NDELAY)) < 0 ) {
               perror(argv[1]);
               exit(errno);
        }
        if (argc > 2) {
              switch(atoi(argv[2])) {
                      case 300:
                              baud = B300;
                              break;
                      case 1200:
                              baud = B1200;
                              break;
                      case 2400:
                              baud = B2400;
                              break;
                      case 4800:
                              baud = B4800;
                              break;
                              baud = B4800;
                              break;
                      case 9600:
                              baud = B9600;
                              break;
                      case 19200:
                              baud = B19200;
                              break;
                      case 38400:
                              baud = B38400;
                              break;
                      default:
                              baud = 0;
                              fprintf(stderr,
                                "%s: %s is an unsupported baud\n" ,
                              argv[0], argv[2]);
                              exit(1);
                        }
               }
        /*      SAVE STDIN AND TTY STATE. ALSO TRAP SIGNALS   */
               ioctl(fd, TCGETA, &term_save);
               ioctl(fileno(stdin), TCGETA, &stdin_save);
               signal(SIGHUP,  Exit);
               signal(SIGINT,  Exit);
               signal(SIGQUIT, Exit);
               signal(SIGTERM, Exit);
```

```
/* SET STDOM TP RAW MODE, NO ECHO    */
   ioctl(fileno(stdin), TCGETA, &tstdin);
   tstdin.c_iflag = 0;
   tstdin.c_lflag &=   (ICANON | ECHO);
   tstdin.c_cc[VMIN] = 0;
   tstdin.c_cc[VTIME] = 0;
   ioctl(fileno(stdin), TCSETA, &tstdin);

/* SET TTY STATE   */
   ioctl(fd, TCGETA, &term);
   term.c_cflag |= CLOCAL|HUPCL;
   term.c_iflag &=  ICRNL; /*
   TO AVOID UN-NEEDED BLANK LINES */
     if (baud > 0) {
              term.c_cflag &=  CBAUD;
              term.c_cflag |= baud;
         }
   term.c_lflag &=  (ICANON | ECHO); /* TO FORCE RAW MODE */
   term.c_cc[VMIN] = 0;
   term.c_cc[VTIME] = 10;
   ioctl(fd, TCSETA, &term);
   fcntl(fd, F_SETFL, fcntl(fd, F_GETFL, 0) &  O_NDELAY);

/* OPEN TTY FOR READ AND WRITE  */
     if ((fdr = fopen(argv[1], "r")) == NULL ) {
              perror(argv[1]);
              exit(errno);
     }
     if ((fdw = fopen(argv[1], "w")) == NULL ) {
              perror(argv[1]);
              exit(errno);
       }

/* ALL I/O IS UNBUFFERED */
   setbuf(stdin, 0);
   setbuf(stdout, 0);
   setbuf(fdr, 0);
   setbuf(fdw, 0);

/* TALK TO THE MODEM  */
   puts("Ready... ¬C to exit");
   while (1) {
           if ((num = read(fileno(stdin), buffer, 80)) > 0)
                   write(fileno(fdw), buffer, num);
           if ((num = read(fileno(fdr), buffer, 80)) > 0)
                   write(fileno(stdout), buffer, num);
    }
    Exit(0);
}     /*       E N D    O F    P R O G R A M       */

/*
```

As a last resort, if you do not have a C compiler installed in your system, and uucp is not installed in your system either, then you will have to program your modem through a PC or another AIX box.

Keep in mind that what will be important is the Baud rate at which you program your modem at and the command string which you send to the modem.

Troubleshooting Tips

If the above steps do not succeed in configuring your modem:

- Ensure you have the proper cabling. The cable for a modem is NOT the same cable used for a terminal or printer. If you dial into the RISC and do not receive a login, it is a sign that cabling may be a problem. If you use cu -ml tty#, connect to the port and hitting any key does not cause any activity on the RxD/TxD indicators on the modem, cabling is a likely cause.
- If your modem does not answer at all: Verify that auto answer is set on (usually indicated by an AA indicator light). Also, even with AA on, the modem generally will not answer if DTR is not high. If your TR light is not on, either the port is disabled, the cabling is faulty, or your modem has a hardware problem. Try placing the modem on a different port to see if the serial port is bad. Try swapping cables and modems if possible.
- Double check setups in SMIT. Do the baud rate, parity, data bits and stop bits match what users are connecting at? Is enable LOGIN set properly?
- If you receive the getty respawning too rapidly error, it is usually caused by cabling or a modem holding the carrier detect signal high. Be sure that your modem dip switches and/or AT&C1 is set. With this, you might want check the /etc/security/failedlogin file and see if it is increasing when a tty is enabled. This points to CD signal either strapped high or faulty cabling.
- If you are unable to connect to the port with cu, is -m a recognized flag? If not, you may not be able to configure the modem with this method. Is the device busy? If so, pdisable tty# before issuing the cu command. If you receive a message "no devices available" ensure that the /etc/uucp/Devices file is in a correct format.

Appendix

Program 1

This program is supplied AS IS. If it is not possible to install the recommend APAR, then use this at your own risk.

```
/*          C Program to add RTS discipline to tty port(s).

     Tips:

To make the program more permanent, insert the the file
name of the compiled version (complete with path) at the
end of your "/etc/rc" file and the changes will take
effect again at next reboot. You can modify this program
to include a variable in place of /dev/tty0 which would
change in a loop to be any tty numbers for which rts is
requested.  Program is setup to change only tty0.

NOTE: This program is supplied "as is" and is NOT
supported by IBM.  It is intended as an aid to
administrators only.
_____________________________________________________________


     To create:     vi addrts.c <enter>
     To compile:    cc -o addrts  addrts.c
     ____________________________________________________

/*                    Program starts now                 */

#include <stdio.h>
#include <fcntl.h>
#include <termios.h>
#include <sys/tty.h>

main()
{
   int fd;
   printf("opening\n");
   fd =3D open("/dev/tty0", O_NDELAY|O_RDWR);=20
   printf("running ioctl\n");
   ioctl(fd, TXADDCD, "rts");   /*  adds rts to tty0       */
   printf("closing"\n");
   close(fd);=20
}
```

Modem Programs:

These command strings are supplied AS IS, and they are not guaranteed to work with every modem.

It is strongly recommended that your modem settings are set in the manner that follows:

- The DTE (modem to RS/6000) baud rate should remain fixed on the last AT command sent.
- The Carrier Detect signal (modem to RS/6000) should be asserted (high) when a modem to modem connection is established.

- The Data Terminal Ready signal (RS/6000 to modem) should be set for the modem to accept (follow) DTR from the RS/6000.
- The Flow control used on the modem should be set for RTS/CTS (Hardware) flow control for both TxD(Transmit) and RxD(Receive). Xon/Xoff(Software) flow control is strongly discouraged.
- The Data compression used on the modem should be set to a selective mode. This is true for modems with V.42bis, V.42, V.32, and V.34. MNP compression should be disabled.
- The DCE (modem to modem) baud rate should be set to a variable state.

USRobotics:

On USRobotics Sportster modems, the following string has been verified to work:

```
AT&F&B1&C1&D2&H1&I0&R2&K3&N0Q2E0&W
```

IBM:

On IBM 7855 modems, the following strings have been verified to work:

```
AT&F&D2&C1&B8E0L0N1R2Q2S0=3D1&W
```

For compression, use the previous string along with:

```
ATN3%C1A3&W
```

Hayes:

On Hayes Smartmodems, the general AT command string used in this doc will work. An adendum would be:

```
AT&F&D2&C1&K3E0Q2S0=3D1&W
```

UDS:

On UDS Modems, the following string has been verified:

```
AT&FQ2E0Y0&C1&D2&S1%B5%E0*LC&W
```

Telebit:

On telebit modems, the following string can be used:

```
AT&F&C1&D2&Q0&R3&S1&T5E0Q2X12&W
```

Other Modems:

You should be able to use the general AT command setup string stated earlier in this doc. Should you need assistance, contact your modem vendor/support personnel.

If errors are received when sending the AT command strings, try sending them in pieces. For example,

```
AT&F&D2&C1&B8E0L0N1R2Q2S0=3D1&W
```

can be split into the commands.

```
AT&F
AT&D2
AT&C1
AT&B8
ATE0
ATL0
ATN1
ATR2
ATQ2
ATS0=3
ATD1
AT&W
```

This will allow you to identify which AT command is not understood by your modem. There may be an equivalent command string for your particular modem. Please consult your modem manufacturer for such specifics.

Reader's Comments

Please fax this form to (512) 823-4009, attention "AIXServ Information". You may also e-mail comments to: elizabet@austin.ibm.com. These comments should include the same customer information requested below.

Use this form to tell us what you think about this document. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

If you need technical assistance, contact your local branch office, point of sale, or 1-800-CALL-AIX (for information about support offerings). These services may be billable. Faxes on a variety of subjects may be ordered free of charge from 1-800-IBM-4FAX. Outside the U.S. call 415-855-4329 using a fax machine phone.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Note: If you have a problem report or item number, supplying that number may help us determine why a procedure did or did not work in your specific situation.

Problem Report or Item #: Branch Office or Customer #:

Be sure to print your name and fax number below if you would like a reply:

Name: Fax Number:

END OF DOCUMENT (modem.howto.zap)