telser.doc

COLLABORATORS			
	TITLE : telser.doc		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		December 8, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

1 telser.doc **1**

1.1 telser.doc . . . 1
1.2 Table Of Contents . . . 1
1.3 Legal Stuff . . . 2
1.4 Copyright . . . 2
1.5 License Agreement . . . 3
1.6 Disclaimer . . . 3
1.7 Introduction . . . 3
1.8 Demo Restrictions . . . 4
1.9 Requirements . . . 4
1.10 Installation . . . 4
1.11 Configuration . . . 5
1.12 Device Unit Configuration . . . 5
1.13 Telnet Hosts Configuration . . . 7
1.14 Terminal Emulation Configuration . . . 9
1.15 Modem Profile Configuration . . . 10
1.16 Server Mode Configuration . . . 10
1.17 Using telserd Server on Multiple TCP Ports . . . 11
1.18 Using telserd in rlogind mode . . . 12
1.19 How telser works . . . 13
1.20 Using telser.device . . . 13
1.21 Using tstelnet . . . 14
1.22 TelserMon . . . 15
1.23 Gadtools Graphical User Interface . . . 16
1.24 Telser Main Window . . . 17
1.25 Telser Options Window . . . 19
1.26 Telnet Send Commands Window . . . 20
1.27 Logging . . . 21
1.28 Modem Commands . . . 22
1.29 Connection Result Codes . . . 24

1.30 serial.device compatibility 25
1.31 Login Scripts 25
1.32 Troubleshooting 26
1.33 History 29
1.34 Acknowledgements 30
1.35 Registration Form 31
1.36 Contact Address 32

Chapter 1

telser.doc

1.1 telser.doc

```
    ============================================================
    telser 1.20 -- serial telnet(d) device for TCP/IP and AS225r2
                           Documentation
                         - March 6, 1995 -
    ============================================================
```

Copyright (c) 1994-1995 by Sam Yee. All rights reserved.

Table Of Contents

Legal Stuff
Introduction
Demo Restrictions
Requirements
Installation
Configuration
How telser works
Using telser.device
Using tstelnet
TelserMon
Gadtools Graphical User Interface
Logging
Modem Commands
serial.device compatibility
Login Scripts
Troubleshooting
History
Acknowledgements
Registration Form
Contact Address

1.2 Table Of Contents

MAIN telser.doc

1. Legal Stuff
  1.1. Copyright
  1.2. License Agreement
  1.3. Disclaimer
2. Introduction
3. Demo Restrictions
4. Requirements
5. Installation
6. Configuration
  6.1. Device Unit Configuration
  6.2. Telnet Hosts Configuration
  6.3. Terminal Emulation Configuration
  6.4. Modem Profile Configuration
  6.5. Server Mode Configuration
    6.5.1. Using telserd Server on Multiple TCP Ports
    6.5.2. Using telserd in rlogind mode
7. How telser works
8. Using telser.device
9. Using tstelnet
10. TelserMon
11. Gadtools Graphical User Interface
  11.1. Telser Main Window
  11.2. Telser Options Window
  11.3. Telnet Send Commands Window
12. Logging
13. Modem Commands
  13.1. Connection Result Codes
14. serial.device compatibility
15. Login Scripts
16. Troubleshooting
17. History
18. Acknowledgements
19. Registration Form
20. Contact Address

1.3 Legal Stuff

Copyright
License Agreement
Disclaimer

1.4 Copyright

The program, telser and its associated files are written and copyrighted by Sam Yee. telser is SHAREWARE and the UNREGISTERED version of telser may be distributed freely providing the following conditions hold:

o Distributors may not charge more than the cost of a diskette used in the distribution of this program.

o Distributors may only distribute the unmodified copy of the original

program, along with its documentation, and copyright notices intact.

o Commercial distribution is only possible with written permission from the author.

1.5 License Agreement

The REGISTERED versions of telser may not be redistributed. Redistribution is illegal, immoral, and strictly prohibited. The licensed keyfile may be used on more than one machine if you own those machines and that they are not for business use. Otherwise, different keyfiles are required for each machine.

1.6 Disclaimer

This program and its documentation are provided "as is". No warranties are made with respect to the accuracy, reliability, performance or operation of this software and information. You are using this program at your own risk. The author is not liable for any damages that may have been caused by using this software.

1.7 Introduction

telser.device ("telser") is a modem simulator over a TCP telnet/rlogin connection. It simulates basic modem commands, so you can use telecommunication ("comm") programs over a network connection. To your comm program, it thinks it's talking to a modem. Terminal emulation, file transfers, scripting, etc. are all handled by your comm program. Connecting to a host is as simple as typing "ATDT abc.edu,23" or even simpler if you add the host to your comm's phonebook.

telser supports most of the telnet/rlogin negotiation commands and options. For example, you may elect to notify the remote host on changes to your terminal emulation type, and window size. An intuitive "gadtools" graphical user interface is supplied to control your connections. You also have the option to automatically connect to a host after starting up your comm program. Instant logins at the click of a button!

telser can also run in host mode, which means it would accept incoming telnet/rlogin calls. Calls may be tracked by telser's caller ID feature. You can easily set up a "multi-line" BBS over the internet!

telser is capable of unlimited device units, which means you can have unlimited incoming and outgoing connections. It supports both TCP/IP packages on the Amiga, namely AmiTCP (tested with 3.0b2 and 4.0/4.1) and AS225r2.

1.8 Demo Restrictions

The demo version of telser allows one device unit. Opening a second unit causes a device busy result code. If you use telser often you are recommended to register. When you register you get unlimited device units and help keep this project alive...

1.9 Requirements

telser runs on any Amiga(tm) with version 2.04 of the OS, running AmiTCP or AS225r2. AmiTCP 3.0b2 and 4.0 demo has been tested, and it does not work with 2.3. telser uses very little memory (perhaps 90K). To make telser even more memory efficient, make it resident. Then every device unit you open will use about 20K.

1.10 Installation

The telser archive contains the following files:

```
telser_install      - Installer script
telser_install.info - icon for above
bin/telser          - telser main program for AmiTCP
bin/tstelnet        - comm program that uses telser.device
TelserMon           - telser monitor commoditiy
TelserMon.info      - icon for TelserMon
c/telser            - telser main program for AS225r2
db/telser.conf      - configuration file for each device unit
db/telser.hosts     - list of host aliases
db/telser.mdm       - stores modem configuration
db/telser.terms     - list of terminal types
db/telserd.conf     - configuration file for serv/telserd
devs/telser.device_amitcp  -
                       device driver that talks to bin/telser
devs/telser.device_as225r2 -
                       device driver that talks to c/telser
doc/telser-Axsh-setup.doc - Axsh setup tutorial in text format
doc/telser-Axsh-setup.doc.info -- icon for above
doc/telser-Axsh-setup.guide - Axsh setup tutorial in AmigaGuide(r)
                              format
doc/telser-Axsh-setup.guide.info - icon for above
doc/telser.doc      - telser documentation in text format
doc/telser.guide    - telser documentation in AmigaGuide(r) format
doc/telser.guide.info - icon for the above
doc.info            - icon for doc/ directory
libs/rawin.library  - linemode editing handling library
serv/telserd_amitcp - telser daemon ("server") for AmiTCP
serv/telserd_as225r2 - telser daemon for AS225r2
src/tstelnet.c      - sample source that uses telser.device
src/tstelnet.makefile - makefile of course
s:telser.key        - registration keyfile for telser
```

```
Once unpacked, the above files reside in the telser/ directory.

AmiTCP Installation
-------------------

To install telser for AmiTCP, unpack the archive into AMITCP:
 1> lha x DH0:telser100.lha AMITCP:

Copy and rename the device driver telser.device_amitcp into devs:
 1> copy AMITCP:telser/devs/telser.device_amitcp
         devs:telser.device clone

Rename the executables:
 1> rename AMITCP:telser/serv/telserd_amitcp AMITCP:serv/telserd

AS225r2 Installation
--------------------

To install telser for AS225r2, unpack the archive into INET:
 1> lha x DH0:telser100.lha INET:

Copy and rename the device driver telser.device_as225r2 into devs:
 1> copy INET:telser/devs/telser.device_as225r2
          devs:telser.device clone

Rename the executables:
 1> rename INET:telser/bin/tstelnet INET:c/
 1> rename INET:telser/serv/telserd_as225r2 INET:serv/telserd

Installer Script
----------------

A  Installer  script ("telser/telser_install") is included to automate
the above procedures.  Just click on the install icon and way you go.
```

1.11 Configuration

```
telser  is  a  very  configurable  program.  Most of the configuration
files  can  be  manipulated  by  the  save  buttons in section
Telser Options Window.  The rest must be editted with a text editor.
In  most  files,  a  "#" or ";" ignores ("comment out") lines.

 Device Unit Configuration
 Telnet Hosts Configuration
 Terminal Emulation Configuration
 Modem Profile Configuration
 Server Mode Configuration
```

1.12 Device Unit Configuration

db/telser.conf allows you to configure each unit of telser.device.

The follow is how db/telser.conf looks like:

```
-------------------------------------------------------------------------------------------

#unit default host (alias)  map serial break to   OPENWIN    LINGER    DEBUG     ↩
   logfile
#                          Nothing               or         or        or
#                          OpenWindow            NOOPENWIN  NOLINGER  NODEBUG
#                          AbortOutput
#                          AreYouThere
#                          Break
#                          EraseChar
#                          EraseLine
#                          GoAhead
#                          InterruptProcess
#                          NoOperation
#                          SynchOperation
 0    ""                   OpenWindow            OPENWIN    NOLINGER  DEBUG    T: ↩
    telser.log
 2    home                 OpenWindow            OPENWIN    NOLINGER  DEBUG    T: ↩
    telser.log
 3    home                 OpenWindow            NOOPENWIN  NOLINGER  DEBUG    T: ↩
    telser.log
-------------------------------------------------------------------------------------------
```

Unit field:
    The unit number the comm program will use.

Host (Alias) field:
    This field tells telser which host should a connection be attempted once the comm program is started. Enter "" if automatic connection is not desired at telser.device opening time. You may use an IP address, host alias, or a real hostname for this field.

Map Serial Break To field:
    Most comm programs allow the user to send a physical break signal to the modem. This feature is inapplicable to a telnet connection. Instead, telser allows the user to do various things when he/she tries to send a "virtual" break. When such a break is sent, telser can be directed to do the following:

    Nothing: ignore the virtual break signal
    OpenWindow: Open telser's main window if it's not already opened.
                If already opened, bring the window to front.
    AbortOutput: Send a telnet Abort Output command to host.
    AreYouThere: Send a telnet Are You There command to host.
    Break: Send a telnet Break command to host.
           Note that this break is nothing like a modem break.
    EraseChar: Send a telnet Erase Character command to host.
    EraseLine: Send a telnet Erase Line command to host.
    GoAhead: Send a telnet Go Ahead command to host.
    Interrupt Process: Send a telnet Interrupt Process command to host.
                       Often this command kills the foreground process
                       on Unix machines.

```
    NoOperation: Send a telnet No Operation command to host.
    SynchOperation: Send a telnet Synchronize Operation to host.

    Consult  Telnet  RFC's  (e.g.,  RFC854) for more information about
    Telnet commands and options.

OpenWin/NoOpenWin field:
    This  flag  tells telser to open its window when your comm program
    starts up.  If OpenWin is specified, the window will be opened, or
    if NoOpeWin no window will be opened.

Linger/NoLinger field:
    This  flag  allows  you  to  terminate  your  comm program without
    closing  the  telnet  connection.  This is useful when you need to
    switch  comm  programs.  When Linger is specified, telser will not
    quit  when  the  comm  program  exits.   Otherwise, if NoLinger is
    specified,  the  connection  will  be closed and telser terminates
    when  the  comm program quits.  This flag sets the Linger checkbox
    in  section  Telser Options Window.  Some BBS packages need this
    option on.

Debug/NoDebug field:
    When  Debug  is  specified,  all  telnet  negotiation  options and
    commands will be logged.

Logfile field:
    Specify  the  file  name  where  all  logs should go.  Enter "" to
    disable  logging  to  file.   The  logfile is written after telser
    exits.   For  an  explanation  on  the  format  of log entries see
    section Logging.

This  file  can  be  manipulated  by the "Save Unit Options" button in
section  Telser Options Window.  This  file is not necessary for the
operation  of  telser.  If a unit or the file is missing defaults will
be used.
```

1.13 Telnet Hosts Configuration

db/telser.hosts looks something like this:

```
-------------------------------------------------------------------------
#                                       telnet options
#                                       bitmap (if set)
#                                       0 - initiate negotiation
#                                       1 - do binary
#                                       2 - do echo
#                                       3 - do sga
#                                       4 - will ttype
#                                       5 - will naws
#                                       6 - will linemode
#                                       7 - will tspeed
#                                       8 - will lineflow
#                                       9 - force linemode
#                                       10 - ignore neg.
#alias   hostname or IP      port term  bits            rlogin? login id    login- ↩
   script
```

```
#                                       01234567890
home     localhost           23   ansi  11111111010
archie   archie.rutgers.edu  23   ansi  11111111010
school   fraser.sfu.ca       23   vt100 11111100010     y       samy
work     incognito.com       23   vt220 11111111010
fun      res.com             0    vt100 11111111010     n       root        amitcp ↩
   :db/res.login
-------------------------------------------------------------------------------------
```

Alias field:
    An alias for the host you want to connect to.

Hostname or IP field:
    The real name or IP address of the host you want to connect to.

Port field:
    The port you want to telnet into. Normally this is 23, but for MUD (Multi-User Dungeon) sites, this can be something like 6667. Specify 0 if you don't care which port you are connecting to.

Term field:
    The terminal emulation you will be using for the connection. Note that telser does not handle terminal emulation per se. It is up to your comm program to do so. Term is provided so that the host you are connected to will automatically know what term you are using. You must enable automatic terminal emulation if you want to notify the host about the terminal you are using. Telnet options bit #4 is used for this purpose.

Telnet Options Bitmap field:
    This bitmap allows you to specify what is to be performed during telnet negotiation. Putting a "1" at a bit position sets it. The bits are defined as follows.

    Bit 0: We initiate telnet negotiation with the host.
           If 0, telser will not talk first during Telnet negotiation.
    Bit 1: We ask the host to go into binary mode.
    Bit 2: We ask the host to echo what we typed. We will not do any echoing.
    Bit 3: We ask the host to Suppress Go Ahead.
    Bit 4: We tell the host we will be sending our terminal type.
    Bit 5: We tell the host we will be sending our window size.
    Bit 6: We tell the host we will be going into linemode.
           In linemode, editing is done locally. This speeds up command line editing immensely.
    Bit 7: We tell the host we will be sending our terminal speed.
    Bit 8: We tell the host we can do local flow control.
           If host grants permission, when user presses ^S (XOFF), incoming TCP data is stopped until user presses ^Q (XON). When incoming flow is stopped, outgoing flow is still possible.
    Bit 9: We do linemode whether host allows it or not. In linemode, command line editing is performed locally. This bit does not result in any negotiation with the host.
    Bit 10: If set, the telnet protocol is not used. Telnet control

```
            codes (if any) are passed directly to the comm program.
            This is required to connect to some TCP ports, such as
            UUCP and MUD servers.

    Of course the host can refuse any of our requests.
    To  understand what these Telnet options and commands do, refer to
    their RFCs (Request-For-Comments).  The following RFCs deal with
    the telnet protocol: 652, 653, 727, 854, 855, 856, 857, 858, 859,
    860, 885, 930, 1073, 1079, 1143, and 1372,

rlogin field:
    This field indicates whether the host should be connected with the
    rlogin  protocol.   Enter "Y" for yes, or "N" for no.  If "N" then
    the login id field is ignored.  The rlogin protocol typically uses
    TCP port 513.  See Telser Main Window.

login: field:
    This  field  specifies  the  login  ID  to  be used for the rlogin
    protocol.  See Telser Main Window

login-script field:
    The script file used for auto-login.  See section Login Scripts
    on the script language.

This  file  can  be manipulated by the "Save Telnet Options" button in
section Telser Options Window.
```

1.14 Terminal Emulation Configuration

```
db/telser.terms  allows you to specify the list of terminal emulations
your  comm  programs can support.  They can be used to notify the host
on what emulation you are using.

db/telser.terms looks something like this:
-----------------------------------------------------------------------
#term   cols rows
ansi    80   24
dumb    80   24
vt52    80   24
vt100   80   24
vt102   80   24
vt200   80   24
vt220   80   24
vt240   80   24
vt300   80   24
vt320   80   24
vt340   80   24
vt420   80   24
rip     80   48
unknown 80   24
xterm   160  128
-----------------------------------------------------------------------

Term field:
    The terminal type supported by your comm program.
```

```
Cols field:
    The number of columns of your comm window.

Rows field:
    The number of rows of your comm window.

This  file is what gets displayed in the terminal list box in
Telser Main Window.   Telnet negotiation on terminal type and window
size will use these definitions.
```

1.15 Modem Profile Configuration

```
db/telser.mdm  stores  the  modem profile for each telser.device unit.
The  profile  is  loaded  every  time  telser.device  is opened with a
particular unit.

db/telser.mdm looks something like this:
-----------------------------------------------------------------------
[0]
B0
E1
F1
M1
...
-----------------------------------------------------------------------

The  number  in  [] is the unit number of the modem.  What follows are
the  registers, etc.  for that unit.  In this case, telser.device unit
 0 will  use  the  above.   You  normally don't need to edit this file
directly.   Do  your stuff with the "AT" commands in your comm program
and type "AT&W" to save it to this file.
```

1.16 Server Mode Configuration

```
db/telserd.conf  stores  the  units that telserd should attempt to use
when an incoming call is detected.

db/telserd.conf looks something like this:
-----------------------------------------------------------------------
#unit ring-interval max-rings   startup-command cleanup-command logfile
#     (seconds)                 "" for none     "" for none     "" for none
 0    2             2           ""              ""              t:telserd.log
 2    2             2           ""              ""              t:telserd.log
...
-----------------------------------------------------------------------

Unit field:
    The  telser.device  unit  to check for the possibility of making a
    connection.   If this unit is not already opened or busy, the next
    one will be attempted.
```

```
Ring-Interval field:
    The number of seconds between "RING" messages are sent to the unit.

Max-Rings field:
    The maximum number of "RING"s to send to the unit before giving up.
    After giving up, the next unit is tried.

Startup-Command field:
    The command to execute before actually trying to talk to the unit.
    If the command fails (errorcode non-zero), this unit is ignored.

Cleanup-Command field:
    The command to execute after the telnet connection is closed.

Logfile field:
    The file where all logs are written to. If logging is
    undesirable, enter "" for this field. The logfile is written
    after telserd exits.

In addition to editing the above file you must also edit db/inetd.conf
and db/services.

Your db/inetd.conf file should look like this:

telnet    stream  tcp nowait root amitcp:serv/telserd -telserd

If you are using the current version of AS225 or newer, you must omit
the "root" word from the above line!

Your db/services file should look like this:

telnet          23/tcp

Usually, the services file is already set up by your TCP package.

To get your BBS running over telser, you must make sure
db/telserd.conf contains the unit number that is opened by the BBS
software. Also, db/telser.conf should have an entry for the unit
used. All units to be used by telserd should not have auto-login
enabled!!! i.e., make sure the "host" field is blank (""). You
wouldn't want your BBS automatically dial out everytime the device is
openend. If you are using AXsh with telserd, make sure you have
LINGER on or AT&D0!!! It may apply to other BBS packages as well.


 Using telserd Server on Multiple TCP Ports
 Using telserd in rlogind mode
```

1.17 Using telserd Server on Multiple TCP Ports

```
If you need telserd to launch from different TCP ports, you may do so
by passing the telserd program a different telserd.conf file.
telserd.conf specifies which telser.device units to connect the remote
user to. The command line options are:
```

```
    serv/telserd -telserd -c db/telserd2.conf

    (-telserd is the name of the newly created process.
     telserd2.conf is your alternative telserd config file.
     The filename is arbitrary.)

In  addition,  you  must also edit your db/services and db/inetd.conf
files:

db/services:
    telnet2  1234/tcp

    (port 1234 is just an example.)

db/inetd.conf:
    telnet2 stream tcp nowait root serv/telserd -telserd -c db/telserd2.conf

Note  that  telnet2  is  used  in  both  files.  This allows inetd to
determine which daemon to launch when service is requested at a port.

This  setup  is  useful  if  you  want  to run different programs from
different  ports.  For example, your AXsh runs from port 23, BBS runs
from port 3000, and AmigaMUD runs from 6667.
```

1.18 Using telserd in rlogind mode

```
By default, telserd acts as a telnetd server, but it can also act as a
rlogind  server.  This is accomplished by passing the "-r" or "RLOGIN"
parameter  to  telserd.  db/services  must  have  the "login" service
specified  and  db/inetd.conf  must determine which daemon to load for
"login".

Example:

db/services:
    login 513/tcp

db/inetd.conf:
    login stream tcp nowait root serv/telserd -telserd -r

The standard rlogin port is 513, but you may change it.

When the user logs on, the environment variables are set:  USER<unit>,
<TERM><unit>,  and BAUD<unit>.  <unit> is the telser.device unit.  For
example,  when  I  rlogin  from  my  school the variable contents are:
USER2=samy,TERM2=vt100,BAUD2=9600.  If you are writing an application
that uses telser, you may take advantage of these variables.

While  the  user  is logged in, the standard way to forcibly terminate
the connection is by pressing "<CR>~." (carriage-return tilde period).

When  the  user disconnects, the above three environment variables are
cleared.
```

1.19 How telser works

```
In client mode:

<comm-prog> <--> <telser.device> <--> [telser] <-..->  [telnetd]
                (((  LOCAL HOST)))             (TCP/IP)  ((( REMOTE HOST )))
```

Your <comm-prog> opens telser.device and talks to it. telser.device relays all your serial commands to [telser]. [telser] also handles the telnet/telnetd protocol. [telser] communicates with telnetd (or other daemons) on the remote host.

```
In server mode:

<BBS> <--> <telser.device> <--> [telser] <--> [telserd] <-..-> [telnet]
                (((  LOCAL HOST)))                      (TCP/IP) ((  REMOTE ))
```

Your <BBS> opens <telser.device> and waits for an incoming call. The remote user uses the [telnet] program to connect to [telserd]. ([telserd] is launched by inetd.) [telserd] simply moderates messages between [telser] and [telnet]. Once [telserd] makes a connection, it sends a "RING" message to [telser], which is then passed back to the <BBS>. If [telser] is set up to auto-answer it would do an auto-matic connection with [telserd]. Otherwise, the <BBS> must explicitly send an "ATA" command to [telser] to "answer" the call.

1.20 Using telser.device

To use telser.device, run your comm program the usual way. Select the option to change the serial device and unit. Change from "serial.device" to "telser.device". And if you need to have automatic connections, define a unit in db/telser.conf. See
Device Unit Configuration for details.

If telser.device is opened successfully you should be able to send modem commands to it.

To make a connection with a host either type "ATDT <host>,[<port>]" from your comm program or enter the host information and click on the Connect button in the Telser Main Window. To disconnect from host, either log off from the host, or type "+++ATH", or press the Disconnect button from the Telser Main Window. After disconnection, a "NO CARRIER" message is returned.

When you are done, just exit your comm program. telser will stay in memory and/or close any connection depending on the following condition. If LINGER on or AT&D0, telser will not exit. If LINGER on and AT&D2, telser will disconnect if connected, but will not exit. If LINGER off and AT&D2, telser will exit and close any connection.

Under any other condition, telser stays running.

Controlling telser from the Shell

If a telser client is still running (when running with the linger option on) but it is not servicing a comm program, it may be terminated by sending a ^C break signal to it with the "break" command.

For example:
 1> status
 2> break <telser-process> C

To open a telser client's window, send it an E signal. e.g., "break <telser-process> E". To close the window, "break <telser-process> D". A telser process can also be controlled by TelserMon.

1.21 Using tstelnet

tstelnet is a really simple comm program that opens telser.device and allows you to connect to a host. tstelnet starts from device unit 0 and goes through to 20 until telser.device is opened successfully. You may of course specify a specific unit to attempt. By trying each unit you can use tstelnet to act like the "telnet" program.

Usage: tstelnet [-ddevice,unit] host-name [port]

Examples: tstelnet archie.rutgers.edu
          tstelnet mud.com 6667
          tstelnet -dtelser.device,100 amiga.rules.com

tstelnet essentially sends "ATDT <host>,[port]" to the telser.device. If the host name is not given, the user is prompted for the host and port. The following prompts are displayed:

Telnet Host: abc.edu
Port [23]:

tstelnet is particularly useful for telnetting out of a BBS because it is 8 bit clean, so file transfers will work.

Note that tstelnet is not limited to the usage of telser.device. You can use it as a comm program for a real serial port. e.g., "tstelnet -dserial.device,0 1-604-434-3665" would open serial.device unit 0 and tone-dials 1-604-434-3665. To exit tstelnet just hit ecape three times. (Three ^C break signals would also work.) For your enjoyment, the C source code to tstelnet is included in the src/ directory.

1.22 TelserMon

TelserMon is a commodity program that allows you to monitor the status of each telser unit. The program should go into your SYS:WBstartup/ drawer, so it is run every time your system boots into Wokrbench. In the TelserMon.info icon, the following tooltypes are supported:

DONOTWAIT
    If specified, this is a standard tooltype which tells Workbench not to wait for the program to return before it (WB) continues to execute.

CX_PRIORITY=<priority>
    Specifies the priority the commodity should run at.

    Default: CX_PRIORITY=0

CX_POPUP=<YES|NO>
    Specifies whether the TelserMon window should open when it starts.

    Default: CX_POPUP=YES

CX_POPKEY=<hotkey>
    Specifies the key to press (or "hotkey") to bring up the monitor window.

    Some of the standard qualifier keys are:

    lshift - left shift
    rshift - right shift
    capslock - Caps Lock
    control - Ctrl
    lalt - left Alt
    ralt - right Alt
    lcommand - left Amiga
    rcommand - right Amiga

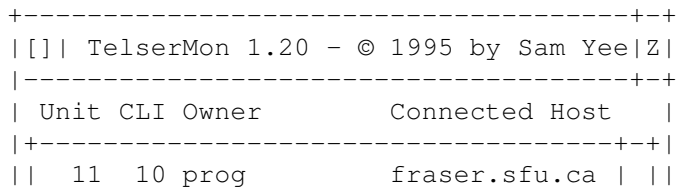
    Default: CX_POPKEY=control alt t

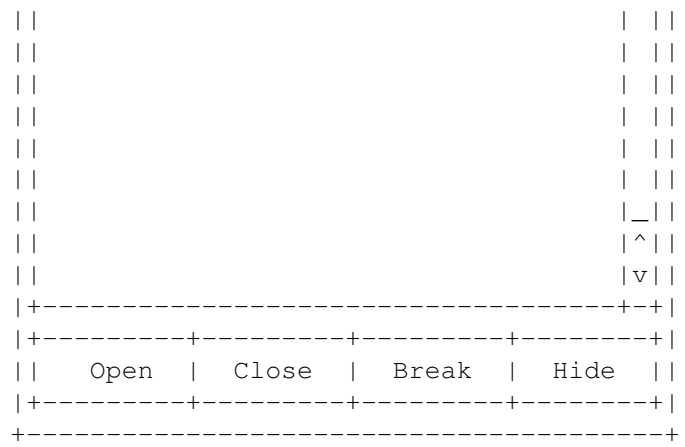
TCPDIR=<directory>
    This specifies the directory where db/telser.conf is stored. telser.conf lets TelserMon know which units it should check.

    Default: TCPDIR=AmiTCP:

TelserMon can also be run from the command line, and it also uses the same arguments as the icon tooltypes.

When the TelserMon window opens, it looks like the following:





Unit column:
    The unit number being monitored.

CLI column:
    The CLI number of the telser process.

Owner column:
    The name of the program that opened the unit. If this column is empty, this unit has no owner.

Connected Host column:
    The host the unit is connected to. If the host is empty, no connection is made.

Open button:
    When pressed, the window associated with the selected unit is opened. Essentially, an E signal is sent to the CLI. Double clicking on a unit in the units list box is effectively the same as clicking on the unit and then Open.

Close button:
    When pressed, the window of the unit is closed. A D signal is sent to the CLI.

Break button:
    Terminate the selected telser unit if it has no owner.

Hide button:
    This button causes the TelserMon window to close. The window can again be opened by pressing the hotkey.

TelserMon can be terminated by either hitting the close gadget on the window or by sending the program a ^C break signal.

1.23 Gadtools Graphical User Interface

As mentioned before, your telnet connections can be managed with a GUI. The GUI may be opened on any public screen by setting the environment variable "TELSERPUBSCREEN" to point to the name of the

screen. For example:

```
 1> setenv TELSERPUBSCREEN TERM
 1> copy ENV:TELSERPUBSCREEN envarc: ; permanently saves it
```

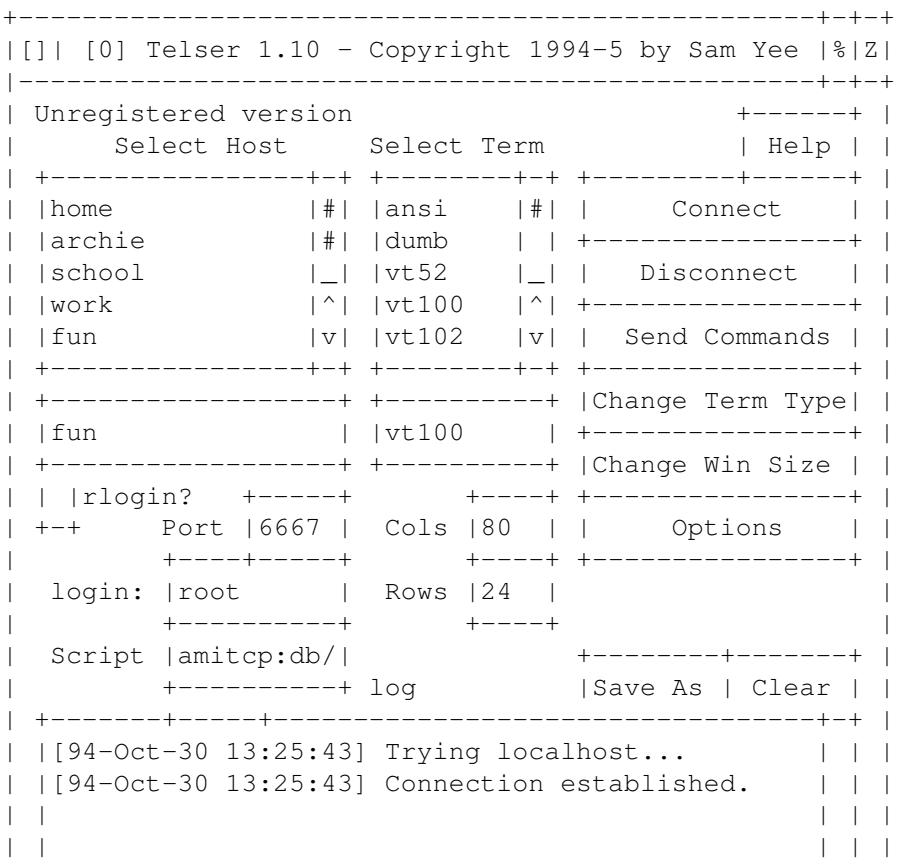
If the public screen cannot be opened, the Workbench screen is used instead. Note that some comm programs, such as Term, opens the serial device first before opening its screen. In this case, if telser was told to open its GUI at start up time, it will not appear in the comm's screen. To solve this problem, simply close the GUI and re-open it. Note that the environment variable can be changed at any time and will take effect the next time the GUI is re-opened.

Telser has 3 windows in which you can change things.

```
 Telser Main Window
 Telser Options Window
 Telnet Send Commands Window
```

1.24 Telser Main Window

If you specify it, when telser.device is opened a window that looks like the below opens. The OpenWin/NoOpenWin option in section Device Unit Configuration can determine whether this window will be opened at device opening time. Alternatively, you may send a "virtual" break signal to telser, as described in section Device Unit Configuration, or typing "AT[" in command mode.



The title bar displays the unit number (e.g., "[0]") of telser.device. You can safely close the window if you don't need it anymore. All other windows associated with this window will be closed as well. Hitting the Escape key also closes the window.

The second line of text displays who telser is registered to.

The gadgets are defines as follows.

Select Host:
    Select a host you want to connect to. To attempt a connection, click once on a host and click on the Connect gadget. Otherwise double-click on the host. If you want to connect to a host not in the list, simply enter it in the edit box below the list box and click on the Connect button.

Port:
    This edit box enables you to establish a connection with a host at particular port number. In this example, the familar port 6667 is used.

rlogin?:
    The rlogin protocol allows you to connect to a site automatically, without enter a login ID or even a password. You should see the Unix man pages on rlogin(d) before trying this feature. Rlogin posts a security risk, so use it with caution.

    Rlogin requires both the local username and the remote username to be sent to the host. The local user name is retrieved from the $USER environment variable. The remote name (or login id) can be specified in the db/telser.hosts file or manually entered from the username field below.

    If this checkbox is checked, use the rlogin protocol to do the connection. After it's checked, the login: gadget is activated and the port number is cleared if it is set at 23 (telnet). For the port number, you may enter 513 (rlogin) or just leave it at 0 to use the default. If the rlogin checkbox is unchecked, the login id gadget is ghosted out.

    Note that rlogin is a simple protocol, unlike telnet, so the only thing you can do while connected (if host permits) is changing the window size. File transfers may not work properly because rlogin is not 100% 8-bit clean. However, on some sites, file transfers work better with rlogin!

login:
    The remote login id to use for the rlogin protocol. It is ghosted out if rlogin is not used.

Script:

The script file to execute after connecting to the host.
See section Login Scripts

Select Term:
Select the terminal type you will be using for this connection. Once a term has been selected, the Cols and Rows edit boxes reflect the type of terminal. However, Cols and Rows can still be manually changed. Note that telser will only notify the host which terminal you are using if you tell it to.
See Telser Options Window

Cols:
This edit box allows you to specify the number of columns your comm program supports.

Rows:
The number of rows your comm program supports.

Help:
Brings up the AmigaGuide(r) help file.

Connect:
Attempt a connection with the specified host. Once a connection has been established, this buttons is ghosted out.

Disconnect:
Disconnect from the host you are connected to. If you are not connected to a host, this button is ghosted out.

Send Commands:
When a connection is made, you may control the telnet connection by clicking on this button. It will then bring up a window with many buttons you can push. See Telnet Send Commands Window

Change Term Type:
If you are already connected, you may tell the host what your new terminal type is.

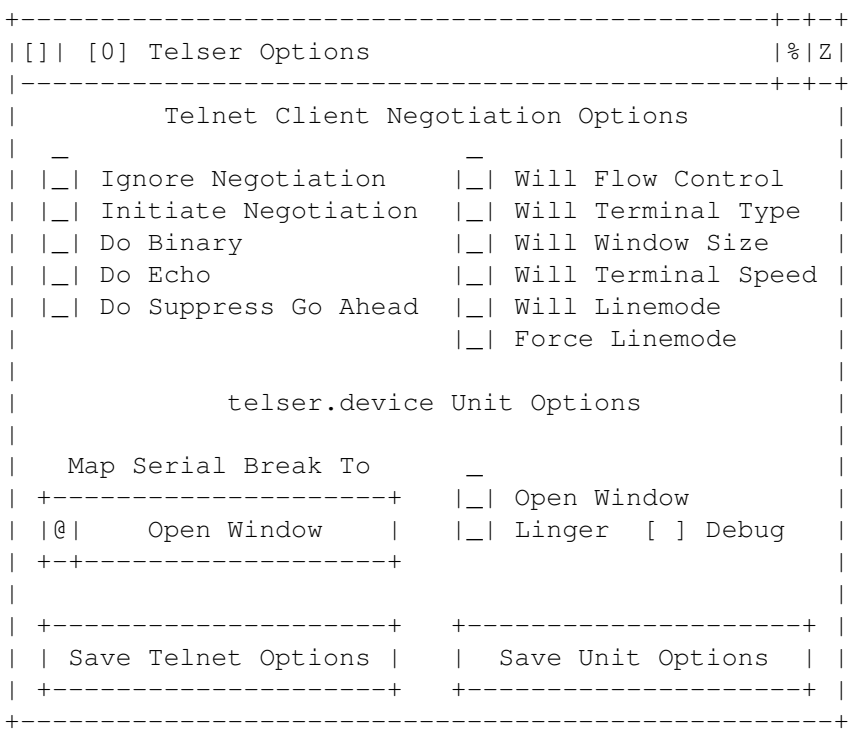
Change Win Size:
If you are already connected, you may tell the host how big your comm program's window is. Programs like "vi" on Unix will auto-adjust itself to reflect the change in window size.

Save As:
Save the logs to a particular file. A file requester will open up to request for a filename.

Clear:
Clear the logs. No warning is given prior to the clear.

1.25 Telser Options Window

This window allows you to manipulate options for a telnet connection or a telser unit.



The "[0]" above indicates that these options are for telser.device unit 0.

Telnet Host Options:
    These options are defined in Telnet Hosts Configuration

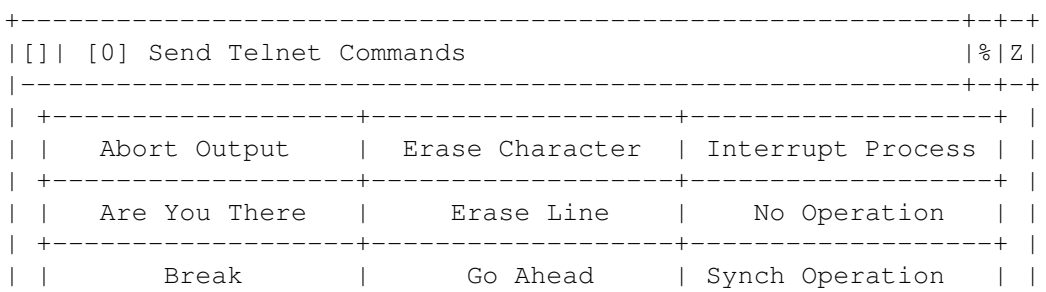
Telser Unit Options:
    These options are defined in Device Unit Configuration

Save Telnet Options:
    Save these telnet options to the db/telser.hosts file.

Save Unit Options:
    Save these telser.device unit options to the db/telser.conf file.

1.26 Telnet Send Commands Window

When a telnet connection has been established, sometimes it is useful to send telnet commands to the host. This window will allow you to do just that.



```
| +------------------+------------------+------------------+ |
+-------------------------------------------------------------+
```

The "[0]" above indicates that these options are for telser.device unit 0.

All these commands are documented in "Map Serial Break To field" of section Device Unit Configuration.

1.27 Logging

All log entries have a time/date stamp. The logs are written at the time telser.device closes, unless of course you click on the "Save Log" button in the Telser Main Window. Most of the log entry types are defined as follows.

Connection closed by foreign host.
    telnetd on the other end has closed our connection

Connection closed.
    you forcibly close the connection

connect() error
    Unable to connect to host at a particular port number.
    Check your host name and port number and try again.
    Port number 0 is the default.

Connection established.
    A successful outgoing connection has been made with the host.

local: <local-name>; remote: <remote-name>; term: <term>; baud: <baud>
    While handling incoming rlogin connections, this log entry displays the local name the user wants to log in with, the remote name of the user, terminal type used, and baud rate he/she is at.

Received ^C break signal
    You sent a break signal to the telser process. If you are connected, you will no longer.

received TCP urgent data

rlogind connection established.
    A successful incoming rlogind connection has been made.

select() returns <result>: error <code>
    Internal error <code> caused by select()

socket() error
    Unable to open a socket

tcp/rlogin: unknown service
    The ermote host does not have rlogind running.

tcp/telnet: unknown service

```
    The remote host does not have telnetd running.

telnetd connection established.
    A successful incoming telnetd connection has been made.

Trying <host>...
    Client is trying to open a telnet connection with <host>
    Note that this may take a while if the host is unreachable.
    Hitting a key on your keyboard won't abort it.

unknown host: <host>
    <host> is not in your hosts file or your name server.
    Try again with an IP address instead.

Refer to your TCP/IP package for other network error codes.
```

1.28 Modem Commands

The basic Hayes(r) compatible modem commands and options are supported by telser.device. Also, several new commands are used by telser. Note that many commands and options are there for compatibility reasons and are ignored. The effective commands and options are defined as follows.

Command/ Options	Function
A	Manually answer an incoming call.
A/	Re-execute the last command. Useful for redialing. It does not take AT or <Enter>.
AT	Attention: lets the telser know that commands are being issued to it.
DP or DT	Attempt to connect to a host. E.g., "ATDT archie.rutgers.edu,23". The 23 following the host name is the port number.
DSn	Connect to the host stored at host list position n. See &Zn. E.g., "ATDS3"
En	Command mode local echo. E0 Echo OFF E1 telser.device displays keyboard commands
Fn	Online local echo. F0 Echo ON F1 Echo OFF (default)
H0	Disconnect from host.
In	Information display. I0 Three digit version number. I4 Display current modem settings. I5 Display nonvolatile memory (NVRAM) (ie., from file) settings. I7 Return version information
O0	Return online.
Qn	Result Codes displayed/suppressed. Q0 Display result codes Q1 Quiet mode; no result codes.

```
Sr=n        Set register r to n.
Sr?         Show contents of S-Register r.
V0          Verbal/numeric result codes.
            V0  Numeric codes
            V1  Verbal codes
Xn          Result code set.

            Result Code     X0  X1  X2  X3  X4
            ---------------------------------
            0/OK            .   .   .   .   .
            1/CONNECT       .   .   .   .   .
            2/RING          .   .   .   .   .
            3/NO CARRIER    .   .   .   .   .
            4/ERROR         .   .   .   .   .
            5/CONNECT 1200      .   .   .   .
            6/NO DIALTONE           .       .
            7/BUSY                      .   .
            8/NO ANSWER                 .   .
            10/CONNECT 2400     .   .   .   .
            13/CONNECT 9600     .   .   .   .
            18/CONNECT 4800     .   .   .   .
            20/CONNECT 7200     .   .   .   .
            21/CONNECT 12000    .   .   .   .
            25/CONNECT 14400    .   .   .   .
            26/CONNECT 19200    .   .   .   .
            27/CONNECT 38400    .   .   .   .
            28/CONNECT 57600    .   .   .   .
            29/CONNECT 115200   .   .   .   .
Z           Reset to software defaults.
&Cn         Carrier Detect (CD) signal.
            &C0 CD override
            &C1 Normal CD operations
&Dn         Data Terminal Ready operations.
            &D0 DTR override; so DTR drop would not cause a
                disconnection from host.
            &D2 Normal DTR operations; telser disconnects when DTR
                drops.
&V          Same as "ATI4"
&W          Write current configuration to NVRAM (ie., db/telser.mdm).
&Zn=s       Write host name string s to NVRAM at position n.
+++         Escape to Online-command mode, or if S14=1, disconnect
            from host and escape to command mode.
[           Open the telser window. E.g., "AT[".
]           Close the telser window. E.g., "AT]".
+CID=n      Caller ID
            +CID=0 disable caller ID.
            +CID=1 enable caller ID: display IP address of caller
            +CID=2 enable caller ID: display hostname of caller
            +CID=3 enable caller ID: display both address and name

            When caller ID is on, incoming calls will cause the
            following lines to be displayed:

            RING
            CID = IP address and/or hostname

*cmd        Special modem commands for sending telnet commands.
```

```
            *AO     Abort Output
            *AYT    Are You There?
            *BRK    Break
            *EC     Erase Character
            *EL     Erase Line
            *GA     Go Ahead
            *IP     Interrupt Process
            *NOP    No Operation
            *SYNCH  Synch Operation

$           Display quick command summary.  E.g., "AT$"


S-Registers

Register    Default     Function
--------    -------     ------------------------------------------
S0          0           Sets the number of rings on which to answer in
                        Auto Answer mode.
S1          0           Counts and stores the number of rings from an
                        incoming call.
S2          43          Stores the ASCII decimal code for the escape
                        character.  The default is '+' (43).    A
                        value of 128-255 disables the escape code.
S3          13          Stores the ASCII decimal code for the
                        Carriage Return Character.
S4          10          Stores the ASCII decimal code for the Line
                        Feed Character.
S5          8           Stores the ASCII decimal code for the Back-
                        space character.
S14         0           If set to 1, the connection will be closed up
                        on receipt of the escape code, returns to
                        command mode and sends the NO CARRIER result
                        code.
S19         0           Sets the duration, in minutes, for the
                        Inactivity Timer.  To disable it set it to 0.
                        When the timer goes off, any connection will
                        be closed.
S50         0           When in host mode, the number of seconds to
                        wait before returning the CONNECT string.

 Connection Result Codes
```

1.29 Connection Result Codes

```
Telnet  connection  result  codes  are  mapped  to the following modem
result codes.

NO CARRIER
    unknown host or disconnection closed

NO ANSWER
    unable to connect to host because host is down

NO DIALTONE
```

```
    unable to open a socket

BUSY
    unknown service

CONNECT ...
    connection established.
```

1.30 serial.device compatibility

The basic serial.device commands are supported. The ones that are not supported are simply ignored and a successful result code is returned to the comm program. Most comm programs are supported. Some examples are: Term 4.1/4.2, VLT 5.867, NComm 3.0, Terminus 2.0c/d, Termite 1.00 demo, Platinum Works! Comm, JPTC, TinyTerminal, and X-Comm. Note that when using telser with Terminus, the "Ignore ODU" option in settings/port must be turned off. When switching to telser.device in X-Comm, make sure you activate the port (Device/Active) and then flush it (Device/Flush Port). AXSH 1.3, DLG 1.0 BBS/OS, AmigaMud, and CNet BBS has also been tested to work. The comm programs that don't work include hft, and Baud Bandit. If you are a comm/BBS program author and telser doesn't work with your program let me know.

1.31 Login Scripts

telser supports a limited scripting language which enables you to auto-login into a host. Lines in the script file can be commented out by preceding the lines with ";" or "#" characters. Commands are not case sensitive. Command arguments that contain spaces must be quoted. The script commands are described as follows.

```
DELAY <seconds>
    Delay some seconds before executing the next line in the script.

    Example: DELAY 5
             delays 5 seconds

DTENTHS <10th-seconds>
    Delays some 10th seconds before executing the next line.

    Example: DELAY 5
             delays 1/2 seconds

END
    Ends the executing of the script. This is optional and required
    only if early termination of the script is desired.

MESSAGE <string>
    Sends a string to the comm program. The string is the same format
    as SEND.
```

```
    Example: MESSAGE "About to auto-login...\r\n"

SEND <string>
    Sends a string to the host.  The string may contain special
    control codes, which are defined as follows:

    \b - backspace (control-H)
    \e - escape (control-[)
    \f - form-feed (control-L)
    \n - newline (control-J)
    \r - carriage-return (control-M)
    \t - tab (control-I)

    Example: SEND "samy\r"
             sends samy<CR> to the host.  Usually sent when the host
             prompts "login: "

WAIT <string>
    Waits until a string is received before executing the next line.

    Example: WAIT "login: "
             waits until the "login:" string is received before moving
             on.

The  script file may reside anywhere, and its name is specified in the
telser.hosts file.  (See section Telnet Hosts Configuration).  Hosts
may  share  the  same  script files.  This is particular useful if you
connect to several hosts using the same user id and password.

Example script file:

    ; this script auto logins into my U
    message "Auto logging into school...\r\n"
    wait "ogin: "
    send "samy\r"
    wait "assword:"
    send "nicetry\r"

If  telser's scripting capability is primitive for your needs, you can
always  use  the  script language built in your comm program.

You  are not recommended to use login scripts if your Amiga is used by
someone you don't really trust.  They can steal your passwords!
```

1.32 Troubleshooting

```
------------------------------------------------------------------------
Problem:
    Nothing happens after I see the "CONNECT" string.

Cause:
    Telnet negotiation failed.
```

Solution:
    Turn BINARY mode off from the Telser Options Window. On some Suns, use the terminal "UNKNOWN" when logging in. After you logged in, issue "setenv TERM <your-term>". (The Suns I have access to don't suffer this problem.)

Problem:
    You can't enter your user id and password.

Cause:
    The host expects a different end-of-line character that terminates a line of input.

Solution:
    On some Suns, use ^J (linefeed) instead of carriage-return. If possible, remap your carriage-return key to linefeed. It may also help if you uncheck the "Do Binary" checkbox in the Telser Options Window.

Problem:
    You can't log into Amiga NetBSD systems.

Cause:
    Telnet negotiation failed.

Solution (maybe):
    Telser's LINEMODE options are not 100% implemented. Hopefully, it'll be completed in the future releases. Meanwhile, try the host again with LINEMODE set off in the Telser Options Window

---

Problem:
    You can log into a BSDI BSD/386 or BSD 4.4 system, but it won't respond to your keystrokes.

Cause:
    Telnet negotiation failed.

Solution:
    telser doesn't support ("real" or "kludge") LINEMODE completely, so turn off LINEMODE in the Telser Options Window and try again.

Problem:
    When you use DTR drop to hang up telser you don't see the full "NO CARRIER" string.

Cause:
    DTR drop involves closing and re-opening the device. Some comm programs flush the device after it opens it, but before the first read. Therefore, the "NO CARRIER" can be partly flushed out.

Solution:
    Do not use DTR drop to hang up. Instead use +++ATH\r.

Problem:
When telser.device is opened the second time, the comm program reports busy.

Cause:
The unregistered version of telser allows only one unit at a time. You are recommended to register. See Registration Form
You chose the same unit as the first comm program.

Solution:
If you are already a registered user, make sure the telser.key file is in S:
Chose a different unit and try again.

Problem:
You can't do file transfers.

Cause:
Intolerable time delays affecting the underlying transfer protocol.

Solution:
Play with the protocol parameters such as relaxed timing, packet sizes, etc. If all else fails, you can always "uuencode" the file and ascii capture the output from the "cat" command.

Try to log on with the rlogin protocol and do file transfers. It works on some systems, like IRIX.

Problem:
You cannot abort dialing.

Cause:
TCP/IP is taking a long time finding a route to the host.

Solution:
Issue a "status" command and then use "break" to send a Ctrl-C break signal to the telser process responsible for the telser.device unit you are using.

Problem:
You see garbage after you connect.

Cause:
Wrong TCP port

Solution:
Make sure you telnetted to the correct port (typically 23), and rlogin to the right port also. Telnetting to a rlogin port or rloginning to a telnet port will display garbage and possibly get disconnected.

Problem:

```
    You cannot get telser to work with CNet BBS.

Cause:
    modem setup incorrectly.

Solution: (written by Dan Fraser (IRC nick: Optic)):
    There are some simple changes that must be made to allow telser to
    work properly with CNet BBS.

    First,  load  CONFIG  and configure a new CNet port as if you were
    installing a new modem.  Completely remove the "Init#1", "Init#2",
    "Terminal", and "Term link" modem strings.  Set the "Idle baud" to
    whatever  speed  you  would  like to use with telser (ie:  19200).
    Remember  to  change the serial device to "telser.device".  Ensure
    that the "Ans.  pause" field is set to at least 10.

    Now,  load  the port in CNet's control panel, open its screen, and
    enter  terminal  mode.   Type  "ATS0=0&D2&W" to make the necessary
    changes  to  telser's config.  Exit terminal mode, and your telser
    port is all ready to go.

-------------------------------------------------------------------------
Problem:
    You cannot get telser to work with TSL 2.0e (Terminus) from
    Workbench

Causes:
    unknown

Solution:
    As reported by Dan Zerkle:

    If  I  start  Terminus  directly from the workbench (without TSL),
    it's fine.  It's also fine to run TSL from the shell.

    To summarize:

                    Shell   Workbench
    Terminus        OK      OK
    TSL             OK      HANGS

    I  now  suspect that is is due to a bug or incompatibility in TSL,
    not telser.
```

1.33 History

```
 Ver. YY/MM/DD Changes
 ---- -------- -----------------------------------------------------------
 1.20 95/03/06 - save telnet options now saves the rlogin preferences
                 and login ID.
               - at startup, if window is opened, it will not be
                 become active.  (Suggested by Vincent Hodges)
               - telserd can now take config filename from command
                 line.  Therefore, it can now service different ports.
```

```
                (Suggested by Vincent Hodges)
              - now works with Platinum Works! comm program
              - can disable telnet negotiation.  This is needed for
                some MUD servers and UUCP.
              - added modem register S50 to allow CONNECT string
                report to be delayed while in hostmode
              - added simple script capabilities
                (Suggested by Dennis Lee Bieber)
              - added TelserMon to monitor telser units
              - can now handle incoming rlogin connections
              - file transfers much improved
 1.10 95/01/29 - Checkbox gadgets are now scaled.
              - ZModem upload/downloading much improved.
              - Spurious characters problem fixed.
              - Caller ID can now report hostname as well as IP
                address.
              - RLogin protocol now supported, as suggested by
                Eddy Carroll.
              - serial EOF mode supported, but not fully tested.
              - asynchronous connect, but stuff like address
                resolution is still synchronous.  You can abort
                host "dialing" some of the time, but not always.
              - known enforcer hits removed
              - added telnet flow control negotiation
              - linemode can now be forced

 1.00 95/01/01 - Original release
```

1.34 Acknowledgements

```
Thank-you's go to...

 - The AmiTCP group at Helsinki University for their great work,

 - Stephan Sürken for Text2Guide, which made creating AmigaGuide
   documents easy,

 - All the beta-testers (TOO many to list.), especially those who
   made reports.

   Special thanks to:

   David Zvekic (IRC nick: Ensoniq) for major testing with the AS225
        version;

   NJ Verenini (IRC nick: Spumoni) for testing the AmiTCP version and
        writing the AXsh installation guide;

   Robert Reiswig (IRC nick: RobR) for writing the Installer script;
        and

   Dan Fraser (IRC nick: Optic) for writing the CNet BBS installation
        tip in the Troubleshooting section,
```

- David Swasbrook (author of SwazBlanker, IRC nick: Swaz) for helping debug TelserMon's popup key, and

- All registered users. Without you, there probably wouldn't have been as many enhancements since 1.0.

1.35 Registration Form

```
---------------------- telser Registration --------------------------

Surname_________________________________ Given Name_________________________________

Company Name____________________ Type of Business_____________________________

Street Address_____________________________________________________________________

City____________________________________ Prov./State_____________________________

Postal/ZipCode_____________________ Country_____________________________________

E-Mail_____________________________________________ uuencoded files OK?___

Networking software and hardware in use_____________________________________

_____________________________________________________________________________

Hostname of BBS telserd is used on__________________________________________

Where did you obtain telser?_____________________________ Version Number____

Suggestions for future releases or new programs_____________________________

_____________________________________________________________________________

Registration fee: $15US for unlimited units.
                  Canadian users may send $20CAN.

Method of payment:
[ ] Money order
[ ] Personal check, please allow 3-4 weeks for clearance.
    No personal checks from outside of Canada and US.
[ ] Cash (wrapped with sufficient paper).
    I hold no responsibility for missing cash.

I have read the section Legal Stuff and agree with it.

_____________________________  _____________________________________________
       (Date)                              (Signature)
-----------------------------------------------------------------------------
```

1.36 Contact Address

Registration fees, questions, ideas, comments, bug reports, etc.
should go to:

Snail Mail: Sam Yee
4595 Nanaimo St.
Vancouver, B.C.
Canada V5N 5J5

Internet: samy@sfu.ca (IRC nick: Encoder in channel #amiga)
FidoNet: 1:153/765 (Terra Firma BBS (604) 434-3665)