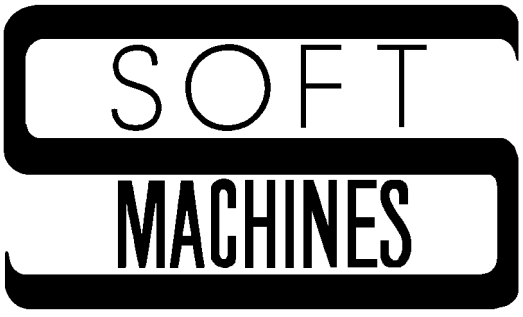
Autolog

Installation and Platform Guide

Linux



Copyright © 2001 Robert P. Rubendunst. All rights reserved.
Printed in the United States of America

Soft Machines
PO Box 14
Sidney, IL 61877

(217) 688-3317
support@softmach.com

This manual is protected by U.S. copyright statutes and may not be reproduced or transmitted in any form or by any means without the express written permission of the copyright holder.

Use of the programs described herein is governed by an End User License Agreement. Refer to the End User License Agreement for terms, conditions, and limited warranty information.

The trademarks and servicemarks referred to in this manual are the trademarks and servicemarks of their respective holders. Soft Machines has made every attempt to indicate such trademarks and servicemarks with initial capital letters.

Contents

Installing Autolog or Slave on Linux 1
- **Installing with `rpm` 1**

Using Autfix to Install the PIC and Configure Ports 2

Configuring the Modem 4

Autfix: Changing Autolog's Defaults 5
- **Comm Port Settings 6**
- **Modem Dialing Settings 7**
- **Special Keys 8**
- **File Transfer Settings 8**
- **Color Settings 10**
- **Phone Log File Settings 10**
- **Restore All Default Settings 11**
- **Quit Without Saving Changes 11**
- **Save All Changes and End Session 11**

The Autolog Initialization File 11

Installing the Dialer 12

Installing Slave Only 12
- **Installing Slave by Uploading 14**

Killing Slave on the Local System 15

Linux- and UNIX-Specific Behavior of Autolog Commands 15
- **Entering and Exiting Autolog 15**
- **Cancel Key 15**
- **Switches 16**
- **Date and Time Comparisons 16**
- **Text Conversion 17**
- **Baud Rate 17**
- **The `debug` Command 17**

File Naming and Directory Conventions 18
- **Files That Autolog Creates 18**

Autolog Installation and Platform Guide

The Linux operating system comes in many flavors. Presently, just Red Hat 5.2 through 7.1 is oficially supported, but don't be too surprised if Autolog runs fine on other Linux platforms.
In this platform guide, we use Linux/UNIX to refer to the varieties of Linux, and UNIX, under which Autolog runs. In the ***Autolog User's Guide,*** references to UNIX should be understood to apply to all varieties of Linux, and UNIX, for which Autolog is available.

Installing Autolog or Slave on Linux

Autolog runs on version 7.0 or later of Red Hat Linux. Installation starts with using `rpm`. If you are installing the full Autolog package, you also need to install the Product Installation Code (PIC) using the Autfix program *after* you use rpm. Slave is the remote file-serving program for SMT files transfers.

When Autolog writes to the terminal screen, it expects that you have declared your **`TERM`** setting to match your terminal device. If the **`top`** command works on your Linux terminal, then Autolog probably will, too.

Installing with `rpm`

To install Autolog or Slave, you must be logged as the superuser, `root`. Then follow these steps:

1. Make sure you are logged on as the `root` superuser.
2. Copy the Autolog rpm file {Autolog-file} to a handy place, such as the /tmp directory. To install, use the command rpm -I {autolog-file}. For example, if the file is named autlogrh71.rpm, then use **`rpm –i /tmp/autlogrh71.rpm`** to install the package.
3. If you installed Slave only, this completes the installation. If you are installing Autolog, you must also enter the product installation code (PIC). Skip ahead to the section "Using Autfix to Install the PIC and Configure Ports" for instructions on entering the PIC.

Using Autfix to Install the PIC and Configure Ports

Your copy of Autolog requires a unique Product Installation Code, or PIC. You use the Autfix program to enter the PIC. First, you must be logged as the superuser, `root`. Then follow these steps:

1. Make sure you are logged on as the `root` superuser.
2. You use the Autfix program to install the Autolog PIC. To run Autfix, enter the command `autfix -i`.
3. A screen like the one in Figure 1 will be displayed. Email Soft Machines (sales@softmach.com) or call Soft Machines at (217) 688-3317. Email or tell our support staff that you would like to install Autolog for Linux. Read or send us the System Signature from the screen. We will reply with a Product Installation Code. Type it carefully at the prompt, and also *write it on a piece of paper* and keep it in a safe place.
4. The next step is to configure your modem ports. A screen like the one in Figure 2 will be displayed.
   a. At the first prompt, enter the names of your modem ports. When an Autolog user types the `link` command without an argument, these are the ports that will be displayed. You may enter a wildcard list (e.g., **`ttyS[01]`**) or a space-separated list (e.g., **`ttyS0 ttyS2 ttyS3`**). Then press **Enter** (the key labeled "Return," "Enter," or "↵").
   b. The second prompt, the default baud rate, indicates the baud rate to use when an Autolog user `link`s to a port. After entering the default baud rate, press **Enter** to proceed to the next prompt.

```
AUTFIX ver 1.0.5          Use arrow keys to navigate

                          Phonetic Alphabet
        Alpha    Echo     India    Mike     Quebec   Uniform  Yankee
        Bravo    Foxtrot  Juliet   November Romeo    Victor   Zulu
        Charlie  Golf     Kilo     Oscar    Sierra   Whiskey
        Delta    Hotel    Lima     Papa     Tango    X-ray

        Please include dashes when reading unique code

Please enter your name or company name: Selena's Systems

System Signature is GP7N-TI55-GGAA-F

Enter installation code: _
```

Figure 1 Entering the Product Installation Code.

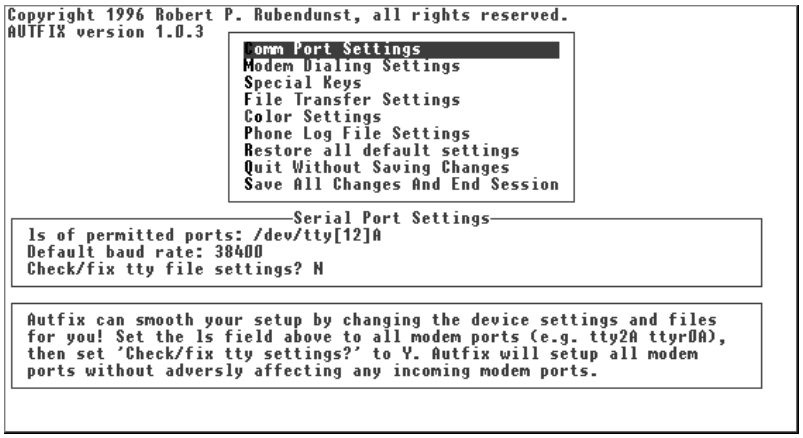


Figure 2 Configuring ports with Autfix.

c. The third prompt asks if you want these ports to be configured automatically. Be sure that the ports you listed in step *a* are serial communication ports (ports for modems or other devices that you'll be communicating with using Autolog). Answer **Y** to configure the ports. The mode, owner, and group of the ports will be modified to allow Autolog to have access to the ports. If you're not sure what type of ports are listed, answer **N**. You can use the Autfix program again later to configure the modem ports if you have trouble with them.

d. Press **ESC** (the Escape key). Type **S**, or use the down arrow cursor key, to select "Save All Changes and End Session," then press **Enter** to exit Autfix and resume the installation.

5. The Autolog installation is complete. Although `usr/lib/autolog` contains most of the Autolog files, the machine-specific PIC is stored in `/etc/autolog.unq`.

The Autolog installation directory, `usr/lib/autolog`, should be used for most Autolog-related files, such as script files (`*.ago`), macro files (`*.amf`), files for the dialer menu system, and so on. This directory is where Autolog will look for its resources unless a different path is specified when indicating resource files.

To start Autolog, from the Linux shell prompt, enter the command `autolog`. (You may also use the command `autlog`, which is the command used on systems that allow only six-letter file names.) The ***Autolog User's Guide*** gives detailed instructions for using the Autolog program. The next section of this platform guide gives detailed setup information.

Configuring the Modem

In order to take advantage of Autolog's modem-control features, your modem cable should include connections for hanging up (DTR), carrier detect (DCD), and hardware flow control (RTS and CTS). We recommend that you buy a good-quality, full-featured modem cable. Don't use a three wire modem cable! Be sure that your modem is configured to use the hardware signals available in your modem cable. Refer to your modem manual for instructions on changing your modem's settings.

If your modem will be used for both incoming and outgoing calls, we recommend that you use the **mgetty** program to handle incoming callers. Some of the other getty programs (like getty or mingetty) that come with Linux don't share the port properly, or don't work properly with modems. The mgetty rpm is on most distrubutions of Linux. (The getty program is controlled by the /etc/inittab file. Please don't touch the inittab file unless you know what you are doing!)

If you want to use a modem for incoming and outgoing calls, then mgetty is the only getty that we have had any success with. The port's entry in /dev must be owned by uucp, and belong to the group uucp. The port's privileges must be rw- rw- r--, or 664 in octal.

An example of an inittab line for the 2nd native serial communications port mgetty is **tty1:2345:respawn:/sbin/mgetty -s 9600 -r /dev/ttyS**1.

Some programs change the modem ports owner, group, and privileges. If this happens, Autolog often cannot use the port, or can only use the port when run as root.

Our experience is that fax software does not play well with others and considers faxmodems non-shareable.

If you are using the modem for incoming and outgoing calls, it should be set to report carrier detect changes. Most modems use the command `AT&C1` to accomplish this. You can use Autolog to give your modem this command: `link` to the modem port, press the **change key** to enter talk mode, then type the modem commands that tell your modem to report carrier detect changes and to save this setting as its power-up default.

To use Autolog's `hangup` command, your modem cable must include a connection for DTR and your modem must be set to hang up when the DTR signal is pulsed. To use Autolog's `carrier` command, your modem cable must include a connection for DCD. To use the `flow` command for hardware flow control, your modem cable must include CTS and RTS connections and your modem must be configure to use these signals for flow control.

If you will be connecting to a device other than a modem (such as another computer or a bar code scanner), you should use a null modem cable that includes hardware flow control connections (RTS and CTS) for best results.

The most important part of setting up your modem is making sure that your modem passes all 8-bit characters transparently. If at all possible, you should disable any "software flow control" features so that characters such as XON/XOFF and ENQ/ACK are passed intact. Make sure that the modem does not discard any characters. This precaution will prevent frustration if you plan to use any error-correcting file transfers! You should also configure your modem to start up with no echoing and in quiet mode.

Note that function keys may present a problem for some communications devices. Not all modems or other devices can send multibyte sequences (such as those generated by a function keypress) fast enough for them to be detected by the remote system as a single multibyte sequence rather than a series of individual characters. If function keys present a problem, contact the equipment manufacturer to see whether the problem can be resolved.

Autfix: Changing Autolog's Defaults

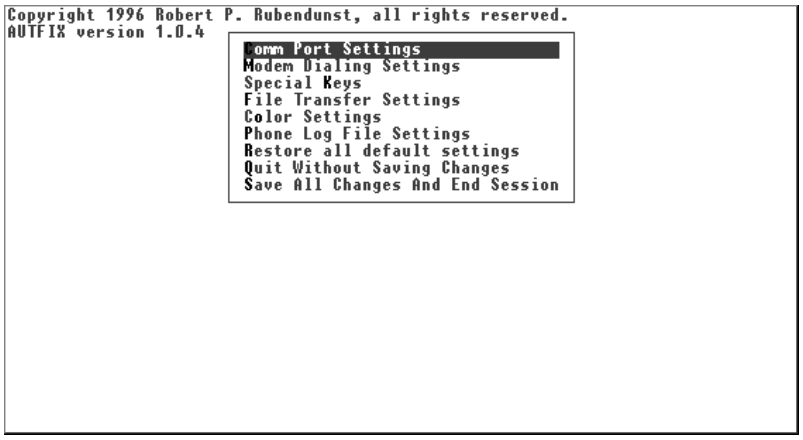


Figure 3 The main Autfix menu.

The Autfix program is used to set Autolog's defaults. To run the Autfix program, you must be logged on as the `root` superuser. Then enter the command `autfix`. A screen similar to the one in Figure 3 will be displayed. You can use the up and down arrow keys of your keyboard or type the highlighted letter of a selection to select the setting you want to change. Then press **Enter** to display a pop-up screen for adjusting that setting. Within each pop-up screen, you can press the up and down arrow keys (or **Enter**) to move among the fields, and you can press **ESC** to exit that screen and return to the main Autfix menu. Except as noted in the following sections, the changes you make are not stored until you select

Save All Changes And End Session. Except as noted, you can abandon your changes by selecting *Quit Without Saving Changes*. Each of the main menu selections is discussed in the following sections.

Comm Port Settings

You may have already visited the *Comm Port Settings* screen when you installed Autolog. This selection allows you to configure the communications ports that Autolog users will `link` to. When you select this entry, a screen like the one in Figure 4 will be displayed.

The first prompt is for entering the names of the communications ports on your system. You can enter a wildcard list (e.g., **`/dev/ttyS[0-1]`**) or a space-separated list (e.g., **`/dev/ttyS0 /dev/ttyS1 /dev/ttyS2`**) of port names, using the same conventions as Linux's `ls` command. The ports you list here will be the ports displayed when an Autolog user types `link` without an argument to locate a port. Autolog users can link to other ports not specified in this list, although this is not recommended. You should enter the names of ports that will be used for serial communications, such as modem ports or ports to which other devices (such as handheld computers or bar code scanners) are connected by a null modem cable.

The second prompt is the initial baud rate used when a user first links to a port. You can change this baud rate with the `baud` command after each `link` command.

The third prompt causes Autfix to configure your communications port correctly for Autolog use. Enter **`Y`** to have Autfix modify the mode, owner, group, and login setting of the ports for trouble-free use with Autolog. (Ports whose login attribute is set to `enable` are changed to `delay`, which works better for modem sharing.) Only the ports you listed in the first prompt will configured. If you specify ports other than modem or serial communications ports (e.g., ports to which terminals or printers are connected), *do not* have Autfix configure these ports—enter **`N`** at this prompt in this case.

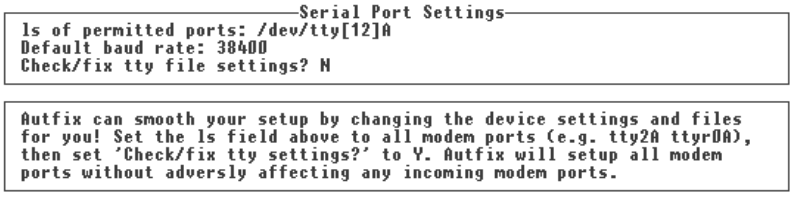


Figure 4 Communication port settings.

☞ Use caution: If you configure the ports, you *cannot undo* this change automatically, not even with the Autfix "Quit" or "Restore" selections. You must manually reconfigure the ports to their previous settings.

Press **ESC** to exit the communications port screen and return to the Autfix main menu.

Modem Dialing Settings

The *Modem Dialing Settings* screen (see Figure 5) is used to create special dialing macros that can be used in Autolog's `dial` command. You can store up to four dialing macros, $L0 through $L3. The dialing macros can be password protected, if you wish. The dialing macros can be used, for example, for a phone card access number, for the prefix necessary to get an outside line, or simply for frequently called numbers.

At the first prompt, enter a password if you want to password-protect the dialing macros, or press **Enter** to leave the password empty. If you enter a password, you will need to enter the same password if you want to change or display the dialing macros. Note that a password is not required to *use* the dialing macros in a `dial` command, only to display or change them in Autfix.

In the next four fields, enter the numbers or dialing characters to store in each macro, $L0 through $L3. After defining a dialing macro, any Autolog user can use that macro by including `$L`*n* in an Autolog `dial` command, where *n* is the number 0 to 3. For instance, if you enter **`9W`** (to get an outside line and wait for a second dial tone) in the $L1 field, an Autolog user can dial an outside phone number by using the command **`dial $L1 555-1212`**. If you enter **`1-217-688-2947`** (Soft Machines' update system) in the $L3 field, an Autolog user can use the command **`dial $L3`** to call Soft Machines.

Press **ESC** to exit the dialing macros screen and return to the Autfix main menu.

```
       Dialing Macros and Password
Password:_
$L0
$L1
$L2
$L3
  Press ESCape to return to main menu
```

Figure 5 Modem dialing macros.

Special Keys

The *Special Keys* selection (see Figure 6) is used to set the default characters for Autolog's special "hot keys." See Section 2.6 in Chapter 2 of the ***Autolog User's Guide*** for more information about these special keys.

In each field, you can enter the character that you want to assign to each of Autolog's special keys. The character you type will be the character used for each key each time a user starts Autolog. You can enter a control character by preceding it with the **^** character (or the "meta" character if it's not currently the ^ character). For instance, enter **^B** for control-B (the ASCII STX character, decimal value 2). If you don't want a key to have a default, enter **^@** (the ASCII null character, decimal value 0).

Press **ESC** to exit the special keys screen and return to the Autfix main menu.

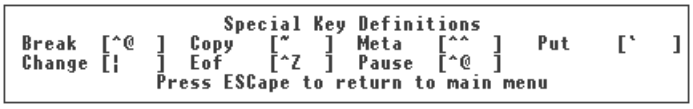


Figure 6 Autolog's special key defaults.

File Transfer Settings

The *File Transfer Settings* screen (see Figure 7) is used to select default settings that affect Autolog's file transfer commands.

The first two fields, *Packet timeout* and *Re-trys before abort*, affect all error-correcting protocols. The packet timeout is the number of seconds that Autolog will wait for an acknowledgment of a file transfer packet before attempting to retransfer the packet. A timeout value of 10 seconds will work well for SMT and ZMODEM file transfers. A longer timeout may be required for YMODEM and XMODEM file transfers (we recommend 30 seconds for these protocols). The timeout value you specify here is the default value. Autolog users can adjust the timeout value at any time during an Autolog session with the `timeout` command (see Chapter 3 in the ***Autolog User's Manual***).

The *Re-trys before abort* field sets the maximum number of times a packet is retransferred before Autolog aborts an error-correcting file transfer.

The *SMT Transfer* fields affect the SMT file transfer commands `transmit` and `receive`. These two fields allow you to select the smallest and largest packet Autolog will use for SMT file transfers. SMT protocol dynamically adjusts the packet size for the most efficient transfers. The "factory default" settings of 64 to 4,096 bytes are the

absolute range for the SMT packet size and, in general, allow the greatest efficiency for SMT file transfers. However, if you use a pseudo-duplex modem (a half-duplex modem that simulates full-duplex operation, such as the DataRace Race I, Race II, or BMX), a fixed packet size works best. You can enter the same number for the smallest and largest packet sizes to use a fixed packet size with pseudo-duplex modems.

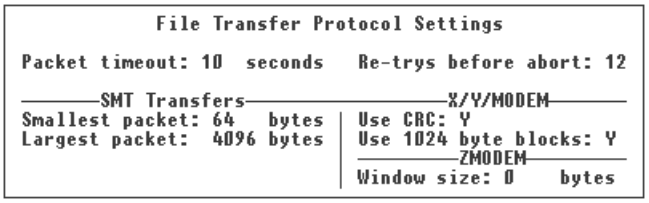


Figure 7 File transfer default settings.

The *X/Y/MODEM* fields affect XMODEM and YMODEM file transfers. The *Use CRC* field determines whether CRC (cyclic redundancy check) or checksum error checking is used by default. CRC error checking is more reliable than checksum but is not supported by all XMODEM and YMODEM programs you may encounter. Enter **Y** at this prompt to use CRC error checking (the `xcrc` option is turned on by default—see Chapter 3 in the ***Autolog User's Manual***). Enter **N** at this prompt to use checksum error checking (`xcrc` is off by default).

The *Use 1024 byte blocks* field determines whether the `x1k` option (see Chapter 3 in the ***Autolog User's Manual***) is on or off by default. In other words, it determines whether X- and YMODEM transfers use standard 128-byte packets or larger 1-kilobyte (1,024-byte) packets. The larger packet size may make file transfers go faster, but it is not supported by all remote XMODEM and YMODEM programs. Enter **Y** to use the larger 1,024-byte packets, or enter **N** to use the smaller, standard 128-byte packets.

The final field, *Window size*, affects only ZMODEM file transfers. This field determines the ZMODEM transfer "streaming window" size, the number of bytes that can be sent but unacknowledged at any given time during a ZMODEM transfer. The "factory default" is 0, which disables an upper limit on the ZMODEM window for maximum speed. In other words, an entire file may be sent before any acknowledgment is received. When data frequently needs to be retransferred (e.g., when the remote system is very busy, with an unreliable modem connection, or with some bursty types of line noise), you should set the window size to be smaller. A smaller window means less efficiency on good connections, but better efficiency on bad connections. An Autolog user can override

the default setting entered here by using the `window` ZMODEM transfer switch (see Chapter 3 in the ***Autolog User's Guide***).

Press **ESC** to exit the file transfer settings screen and return to the Autfix main menu.

Color Settings

The *`Color Settings`* screen (see Figure 8) lets you select the colors Autolog uses for foreground (text) and background. Autolog also uses other colors for certain displays (such as lines or the bar chart for showing file transfer progress), but these are not user definable; they are selected automatically for the best contrast with the selected foreground and background colors.

The first prompt, *`Allow Autolog to set colors?`*, determines whether Autolog uses the foreground and background colors that are currently used by your terminal when you start up Autolog. Answer **N** to this prompt to ignore the Autfix settings and use the terminal's current colors when you start Autolog. You will not see the other fields if you answer **N**. Answer **Y** to make Autolog change the terminal's colors to those you select in the next fields.

In the next two fields, enter the foreground and background colors you want Autolog to use. Enter the number that matches the color you want in the "Color key." The small sample display shows what your color selections will look like in Autolog.

Press **ESC** to exit the color settings screen and return to the Autfix main menu.

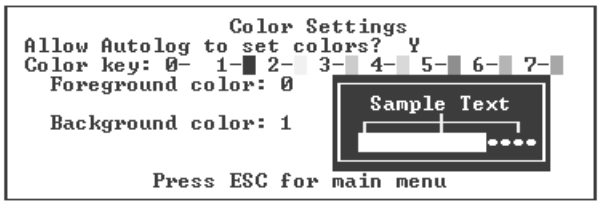


Figure 8 Autolog's color settings.

Phone Log File Settings

The *`Phone Log File Settings`* screen (see Figure 9) controls whether Autolog keeps a log of phone calls placed with the `dial` command and lets you specify the file. For a sample phone log file, see Figure 1.3 in Chapter 1 of the ***Autolog User's Manual.***

At the first prompt, answer **Y** to have Autolog keep a phone log, or answer **N** if you don't want Autolog to keep a phone log.

If you answered **Y** at the first prompt, at the second prompt enter the name you wish to give the phone log file. If no path is specified, the specified file will be created in the Autolog installation directory, `usr/lib/autolog`.

Press **ESC** to exit the phone log screen and return to the Autfix main menu.

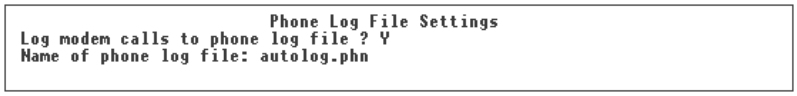


Figure 9 Phone log file settings.

Restore All Default Settings

The *`Restore All Default Settings`* selection will restore all Autolog's "factory default" settings. Note that *it will not undo any port configuration* that was done in the *`Comm Port Settings`* screen. If you change your mind after restoring the factory defaults, select *`Quit Without Saving Changes`* to return the settings to what they were when you started Autfix.

Quit Without Saving Changes

The *`Quit Without Saving Changes`* selection will undo any changes you made during this Autfix session. The settings will be returned as they were before you started Autfix, and you will exit the Autfix program.

Save All Changes and End Session

The *`Save All Changes And End Session`* selection will save the Autfix settings and end the Autfix program. Any changes you made to the settings are saved and will be used the next time Autolog is started up.

The Autolog Initialization File

Autolog can use an initialization script each time it is started. This script file must be called `autolog.ini` and must be located in the Autolog directory, `usr/lib/autolog/autolog.ini`.

The Autolog initialization file is simply an Autolog script, or `go`, file that is used when a user enters the `autolog` command to start up Autolog. See Chapter 4 in the ***Autolog User's Guide*** for more information about script files. You may wish to create an `autolog.ini` that contains the commands users will enter each and every time they use Autolog, for example, a `link` command to select the modem port, a

`modem at` command if using AT-compatible modem, frequently used options and settings, and so on. The Autfix program previously described can also be used to select the default value of some settings.

If you wish, you can create a symbolic link to this file using a different name. For compatibility with the AMOS release of Autolog, you could create a link named `autolog.ini` and then refer to the file as `autlog.ini`. Here is the symbolic link command that would do this:

```
ln -s /usr/lib/autolog.ini /usr/lib/autlog.ini
```

Ashell users may find this symbolic link handy, so that Ashell's VUE program can be used to edit the Autolog initialization file.

Installing the Dialer

The files needed for the dialer menu system (see Chapter 5 of the ***Autolog User's Guide***) are installed when you install the complete Autolog package. The files used by the dialer are `dialer.dat`, `dialer.ago`, and `autset.ago`. Some of these files will need to be customized for your system. Instructions for customizing `dialer.dat`, the dialer database, can be found in Chapter 5 of the ***Autolog User's Guide.*** You will also need to customize the `autset.ago` script file. The `autset.ago` file is used to establish a communications channel (i.e., select the modem, port, and baud rate) when you place a call using the dialer.

You can edit `autset.ago` with your usual text editor or word processor, but be sure to save it as a text-only file. This script contains four Autolog commands:

```
link port
baud number
flow in rts out cts
modem at
```

You need to replace the argument that appears in the `link` command with the name of the communications port you want to use with the dialer on your system. Replace the argument of the `baud` command with the baud rate you want to use on this port. The last two commands do not need to be changed. The dialer uses Autolog's `dial` command, which requires that your modem honor the AT command set as specified. If your modem does not use the AT command set, the dialer will not be able to place calls.

Installing Slave Only

The Slave program is the remote file-serving program for SMT file transfers. As a licensed Autolog user, you may distribute the Slave program to as many remote sites as you wish. If you have installed

Autolog locally, Slave is already installed and ready to use. To install Slave on a remote system, follow the instructions below.

The easiest way to install Slave on a remote system is from a diskette. To install Slave on an LINUX system that doesn't have Autolog, see first section of this guide, "Installing Autolog or Slave on LINUX." Follow the instructions for installing just Slave on an LINUX system. You can also upload Slave using YMODEM or ZMODEM protocol as explained below. But when uploading Slave to another UNIX or LINUX system, you will need to modify the mode, owner, and group of the Slave program file; we recommend that only experienced LINUX/UNIX system operators attempt to install Slave on a Linux/UNIX system by uploading it.

Slave is also available for the SCO UNIX, MS-DOS/Windows, and AMOS operating environments. (**We don't supply one for Linux - just install the Autolog rpm file remotely!**) Slave diskettes (or other appropriate media) for other platforms are available from Soft Machines. These versions of Slave for other platforms are also in your Autolog installation directory, so that you can upload Slave to a remote system if necessary.

It's always a good idea to make sure that the remote system doesn't already have a newer version of Slave before you attempt to install Slave. At the system or shell prompt (or from a DOS command prompt in Windows), enter the command `slave`. If the remote system already has Slave installed, you will see a display similar to the one in Figure 10 that tells you the version number of the remote system's Slave program. A response such as *`Bad command or file name`* or *`not found`* indicates that Slave is not installed (or not installed properly).

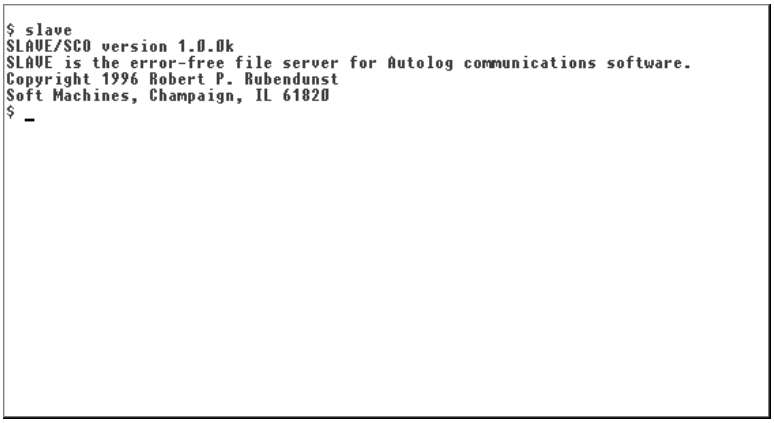


Figure 10 Entering the Slave command manually.

Installing Slave by Uploading

1. Call the remote system using Autolog. Verify that the remote system has software that supports YMODEM or ZMODEM file transfers.
2. The Slave program must be placed in the correct directory: `usr/bin` for Linux/UNIX, C:\AUTOLOG for DOS/Windows, or `SYS:` for AMOS. Make sure that the correct directory exists and that you can transfer a file there (or move the file there after transferring it).
3. Start the YMODEM or ZMODEM software on the remote system and prepare it to receive a file. Press Autolog's **change key** to enter command mode. Enter the appropriate `ztransmit` or `ytransmit` command. (See Chapter 3 in the ***Autolog User's Guide*** for more information about using the `ztransmit` and `ytransmit` commands.) When transferring Slave between unlike operating systems, use the == form of the file transfer command, for example:

   `ztransmit C:\AUTOLOG\SLAVE.EXE == usr/lib/autolog/slave.exe`

   The file you need to transfer is in your Autolog installation directory, usually `usr/lib/autolog`. There are several versions of Slave in this directory:

   `slave.aix` is for AIX
   `slave.sco` is for SCO UNIX
   `slave.lit` is for AMOS
   `slave.exe`, `slvdos.exe`, and `slvw32.exe` are for DOS and Windows. Only `slvdos.exe` is needed for DOS or Windows 3.x systems. Only `slvw32.exe` is needed for Windows 95 or Windows NT systems. When in doubt, send all three `sl*.exe` files to DOS/Windows systems.

   Make sure you transfer the Slave file(s) appropriate for the remote operating system.
4. For DOS/Windows or AMOS, after uploading Slave to the correct directory (or moving it there after uploading; see step 2), the installation is complete. For Linux/UNIX, proceed to the next step.
5. On Linux/UNIX, you must rename the Slave file and change its mode, owner, and group. To do this, you must become the `root` or superuser. Make sure `slave` is in the `/usr/bin` directory. Then enter the following commands:

   `mv slave.sco slave` *or* **`mv slave.aix slave`**
   `chmod 111 /usr/bin/slave` *(three number* 1*'s, not letter* l*'s)*
   `chown bin /usr/bin/slave`
   `chgrp bin /usr/bin/slave`

Killing Slave on the Local System

When a remote Autolog user is done transferring files using SMT protocol with your system, the Slave program is normally exited automatically. If an unforeseen mishap, such a power outage or phone disconnection occurs, UNIX or LINUX typically terminates the Slave program. In the rare event that Slave continues to run after a remote user has disconnected from your system, you can use `kill` to end Slave.

Linux- and UNIX-Specific Behavior of Autolog Commands

Some Autolog commands behave somewhat differently under different operating systems. The following sections describe the commands that behave differently under Linux/UNIX from other platforms.

Entering and Exiting Autolog

The command used to start Autolog is **`autolog`** under Linux. However, the command **`autlog`** may also be used (for compatibility with operating systems that use only six-letter file names).

When you use Autolog's `system` command to temporarily access the Linux/UNIX shell, the shell used may not be your usual shell (i.e., the shell you used to start Autolog). The shell used will be the same shell that the Linux/UNIX `system( )` command uses. The method used to exit the shell and return to your Autolog session depends on the shell in use. Control-D or the command `exit` are commonly used, but this depends on the shell.

Autolog's `finish` command normally restores the communications port to its pre-Autolog state. The `quit` command normally leaves the port as it was configured during the Autolog session. However, under Linux/UNIX, port parameters (such as the baud rate) may revert to some other state after using `quit`.

Cancel Key

The **cancel key** is used to stop the execution of an Autolog command. For instance, if you use the `dial` command to place a call, you can press the **cancel key** to terminate dialing before the remote modem answers. Or you can press the **cancel key** to abort a file transfer that is in progress. The **cancel key** you use is the same as that used by your Linux/UNIX shell to terminate command execution. Control-C or the delete key are commonly used, but this depends on your shell. LINUX generally uses control-C.

Switches

You can use the same switch format for Autolog commands as you do for Linux/UNIX commands: a space plus a hyphen plus the letter(s) that specify the option. The switch should come immediately after a command, before any arguments:

command **–***switch1* **–***switch2 . . . arguments*

Here's an example:

```
ztransmit –hash –q myfile.*
```

However, for compatibility among platforms, you can also use this "universal" switch format: slash (with no intervening space) plus letter(s):

*command***/***switch1***/***switch2 arguments*

Here's an example of the "universal" switch format:

```
ztransmit/hash/q myfile.*
```

☞ An exception to this rule is the `fixed` switch for the `packetsize` command. The `fixed` switch must always use this exact format: **`packetsize`** *number***`/f`**. (See Chapter 3 of the ***Autolog User's Guide*** for details about the `packetsize` command.)

Date and Time Comparisons

Many file transfer switches (such as ZMODEM's `newer` and `longernewer` switches) allow comparisons of file dates and times. Because Linux/UNIX stores more than one date-and-time stamp for files, the date and time used for comparisons is the last-modification date and time.

Some operating systems (including Linux/UNIX and DOS/Windows) take the local time zone into account when comparing dates and times. When transferring files with a remote system that is capable of doing so, date and time comparisons take place as if both local and remote time had been converted to Universal Coordinated Time (UTC). For example, a file on a system in the United States' Central time zone created at 10:00 A.M. is one minute newer than one created the same day at 8:01 A.M. on a system in the Pacific time zone. Your system's time must be set correctly for comparisons to be accurate.

Other operating systems (such as AMOS) do not account for the local time zone. Time stamps for files on these systems are compared as if they were in UTC, even though they probably are in local time rather than UTC. This means that date and time comparisons are rather meaningless when transferring files with such systems.

Text Conversion

SMT and ZMODEM file transfers support the `text` switch, which converts the line endings of a file into the line ending appropriate for a text file on the destination system.

Linux/UNIX normally uses a linefeed character (control-J, ASCII decimal value 10) at the end of lines in text files. Other operating systems, however, may use other character(s). When you use the text switch in an SMT or ZMODEM file transfer command, *outgoing* files' line endings will be changed to carriage return (control-M, ASCII decimal value 13) plus linefeed. The destination system should convert the carriage return–linefeed into the native line ending for that system, if necessary.

When the `text` switch is used, carriage return–linefeed in an *incoming* file will be changed to linefeed alone, the native Linux/UNIX line ending. Incoming files using linefeeds alone will be left as is.

Note that different operating systems may use many other types of line endings. However, the assumption by SMT and ZMODEM protocols is that the remote software uses the `text` option as well so that whatever line ending is native to the remote system will be translated into a "generic" carriage return–linefeed line ending when you download files and that the "generic" line ending will be translated into the destination system's native line ending.

Baud Rate

Autolog sets the communications port speed to the default baud rate you entered in Autfix (the factory default is 38400, unless you change it with Autfix). If you need to use another baud rate, you can use the `baud` command after you have used the `link` command.

The `debug` Command

The `debug` command is discussed in Section 2.9 of Chapter 2 of the ***Autolog User's Guide.*** As noted in that chapter, the `debug uart` command does not work under Linux/UNIX. Instead, the `debug port` command is used to display diagnostic information about the communications port under Linux/UNIX.

There is a another special `debug` option only for Linux/UNIX: `debug nolock`. This command lets Autolog bypass UUCP port locking.

UUCP port locking is a method to prevent two users from trying to use the same modem at the same time. The first user or process to gain control of the port locks out others from using that port. Normally, if you try to `link` to a port that is locked, Autolog will indicate that the port is busy and will not allow you to `link` to it. The `debug nolock` command makes Autolog bypass the UUCP port-locking scheme used to prevent

modem collisions, so Autolog can try to `link` to a "busy" modem port. Note that a port that is locked may be in use by another process. Autolog's attempts to get control of the port may or may not be successful, depending on who or what is locking the port.

To allow Autolog to try to link to busy ports, enter the command:

```
debug nolock true
```

To resume normal, recommended operation (Autolog does not allow `link`ing to busy ports), enter

```
debug nolock false
```

File Naming and Directory Conventions

Unless you specify otherwise, Autolog will look for its initialization and other script files (e.g., script files, `autolog.ini`, etc.) in the Autolog installation directory (`usr/lib/autolog`). When transferring files, unless you specify otherwise, Autolog will look for files to send and will place received files in the current directory.

Linux/UNIX file names can be longer than 14 characters long in general. Unlike many other operating systems, the dot (`.`) does not indicate a traditional file extension: It can appear anywhere in a file name and can appear more than once. File names are case sensitive. Although traditional file extensions are not required by Linux/UNIX, you must include a file extension for Autolog's resource files (script files, macro files, and key translation files). If no file extension is given for these files, Autolog will automatically look for a file with the appropriate default extension, so it will not be able to find a file that has no extension.

The names of files you receive from other operating systems with different naming conventions will be altered to adhere to the Linux/UNIX naming convention if necessary, although Linux/UNIX is generally more permissive about file names than other operating systems. However, be forewarned that using spaces as part of a filename may lead to problems in specifying such a file, as most system commands consider a space to denote the end of one filename and the start of another filename. In some cases, you may be able to use single quote marks around filenames with odd characters, but this may not work with every command.

To specify files in a file transfer command, you can use the wildcard symbols `*`,`?`, and `[ ]`. You can specify a list of files by separating the names with spaces, for example:

```
transmit myfile.txt yourfile.txt ourfile.txt
```

Files That Autolog Creates

When you use the `key save` command to save key translations (see Section 2.3 in Chapter 2 of the ***Autolog User's Guide***), Autolog will

create a file in the Autolog installation directory with the same name as your TERM environment variable, unless you specify a different file name. If a file name that matches your TERM variable can't be found the next time you start up Autolog, Autolog will look for a file called `keys`, which is therefore a good name to use if you want to create key translation files for all users to use, regardless of their terminal characteristics. The file extension depends on whether terminal emulation was active when you save the key translations. If no terminal emulation is active, the file extension is `.ago`. If one of the VT100 emulations is active, the file extension is `.an1`. If the "generic" Autgen terminal emulation is active, the file extension is `.am1`. If the WYSE60 terminal emulation is active, the file extension is `.wy1`.

As described in Section 3.2 in Chapter 3 of the ***Autolog User's Guide,*** the `rename` option prevents overwriting files with `zreceive` when the file names must be truncated or otherwise altered and when this alteration would result in creating files of the same name. The first 26 files that must be renamed will be called `au`*xnnnnn,* where *x* is a letter from a to z and *nnnnn* is a five-digit number that corresponds to Autolog's job process number. The next 26 files will be called `bu`*xnnnnn,* and so on. For example, the first 27 files will be renamed `aua00950`, `aub00950`, . . . , `auz00950`, `bua00950`, ...

Index

Autfix, 2–3, 5–11
autolog.ini, 11–12
autset.ago, 12

Baud rate, 2, 6, 17
Break key, 7–8

Cancel key, 16
carrier, 4
Change key, 7–8
Color settings, 10
Communications port, 2–3, 5–6
Copy key, 7–8

Date and time comparison, 16–17
debug
  nolock, 18
  port, 18
Default settings, 5–11
dialer, 12
Dialing macro, 6–7
Directory, 18
  Autolog installation, 3

ENQ/ACK, 4
EOF character, 7–8
Error correction
  for XMODEM and YMODEM, 9

File name, 18–19
  extension, 18
File transfer settings, 8–10
finish, 15
Function keys, 4

hangup, 4
Hardware flow control, 4

Initialization file, 11–12
Installation
  of Autolog, 1–3
  of dialer, 12
  of Slave, 13–15
Interrupt key, 16

key save, 19

Line ending, 17
Log file of phone calls, 10–11

Macro, dialing, 6–7
Meta character, 7–8
Modem, 4–5

Naming convention, 18–19

Packet size
  for SMT protocol, 8
  for XMODEM and YMODEM, 9
  for ZMODEM, 9
Path, 18
  Autolog installation, 3
Pause key, 7–8
Phone call log, 10–11
Product Installation Code (PIC), 2
Put key, 7–8

quit, 15

rename, 19
Retries, 8

Screen colors, 10

Settings, default, 5–11
Shell, 15
Slave
  ending, 15
  installation of, 13–15
SMT protocol, 8
Special keys, 7–8
Starting Autolog, 3–4, 15
Start-up script, 11–12
Switch, 16
system, 15

Terminal emulation
  and key translation, 19
text switch, 17
Time zone, 16–17
timeout, 8

Wildcarding, 19
window, 9

x1k, 9
xcrc, 9
XMODEM, 9
XON/XOFF, 4

YMODEM, 9

ZMODEM, 9