

LCDproc User's Guide

The Ultimate Guide to LCDproc 0.4.4

Rene Wagner

`reenoo@gmx.de`

Guillaume Filion

`gfk@logidac.com`

William W. Ferrell

`wwf@splatwerks.org`

LCDproc User's Guide: The Ultimate Guide to LCDproc 0.4.4

by Rene Wagner, Guillaume Filion, and William W. Ferrell

Copyright © 2003 Rene Wagner

Copyright © 2002 Guillaume Filion

Copyright © 1999 William W. Ferrell

This document is a guide to LCDproc written for users. It covers LCDproc 0.4.4 ONLY. Some things may be valid for 0.5 as well, but a lot has changed since 0.4.1 and a lot will change for 0.5.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover texts, and with no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Table of Contents

1. Introduction.....	1
1.1. About this Document	1
1.2. What is LCDproc?	1
1.3. The LCDproc Server - LCDd.....	2
1.4. The LCDproc "Main" Client - lcdproc.....	2
1.5. Other LCDproc Clients	2
2. How to Obtain LCDproc.....	3
2.1. Versioning	3
2.2. Download Last Day's CVS Version of LCDproc as a Tarball	4
2.3. Download The Latest Version of LCDproc from CVS	4
2.4. Debian GNU/Linux	5
2.5. Gentoo Linux	6
3. Installation.....	7
3.1. Build LCDproc.....	7
3.2. Install LCDproc Directly From The Sources	7
3.3. Generate And Install Packages of LCDproc	8
4. LCDproc Configuration	10
4.1. Configure LCDd.....	10
4.1.1. LCDd.conf: The [server] Section	11
4.1.2. LCDd.conf: The [input] Section	12
4.1.3. LCDd.conf: The Drivers Section	13
4.2. The LCDproc Init Scripts.....	13
4.2.1. init-LCDd	14
4.2.2. init-lcdproc	14
5. LCDproc Drivers	15
5.1. The HD44780 Driver	15
5.1.1. Connections	15
5.1.2. Compiling	28
5.1.3. Configuration.....	28
5.1.4. Running	28
5.1.5. Questions & Answers	29
5.1.6. Miscellania	29
5.2. The Toshiba T6963 Driver	30
5.2.1. Connections	30
5.2.2. Compiling	30
5.2.3. Configuration.....	30
5.2.4. Running	31
5.3. Parallel Port Troubleshooting.....	31
5.3.1. Check The Wiring	31
5.3.2. Power Source Unregulated / Noisy	31
5.3.3. Ground Lift.....	32
5.3.4. Latchup	32
5.3.5. Contrast.....	32
5.3.6. Parallel Port Voltage	32

5.3.7. Enable Signal Rise Time	32
5.3.8. EMI.....	33
5.3.9. One or Two Black Lines.....	33
5.3.10. Software Too Fast	33
5.3.11. LED Backlight.....	33
5.3.12. HD44780 Compatible.....	33
5.3.13. Miscellania	34
5.4. The Matrix Orbital Driver (MtxOrb)	34
5.4.1. Matrix Orbital LCD Modules	34
5.4.2. Matrix Orbital Hardware Installation	35
5.4.3. Copyright	37
5.5. The lircin Driver.....	38
5.5.1. Checking Your LIRC Setup.....	38
5.5.2. Build LCDd with the lircin Driver	38
5.5.3. Configure LCDd to Use the lircin Driver	38
5.5.4. Modify Your ~/.lircrc.....	39
6. Running LCDproc	41
6.1. Running LCDd.....	41
6.1.1. Running LCDd from the command line.....	41
6.1.2. The Command Line Options of LCDd.....	41
6.2. Running lcdproc	42
6.2.1. The Command Line Options of lcdproc	42
7. Contact Us	43
7.1. Errata.....	43
7.2. The LCDproc Mailing List	43
7.3. Reporting Bugs	43
A. GNU Free Documentation License.....	45
A.1. PREAMBLE	45
A.2. APPLICABILITY AND DEFINITIONS	45
A.3. VERBATIM COPYING.....	46
A.4. COPYING IN QUANTITY	46
A.5. MODIFICATIONS.....	47
A.6. COMBINING DOCUMENTS.....	48
A.7. COLLECTIONS OF DOCUMENTS	49
A.8. AGGREGATION WITH INDEPENDENT WORKS.....	49
A.9. TRANSLATION	50
A.10. TERMINATION.....	50
A.11. FUTURE REVISIONS OF THIS LICENSE.....	50
A.12. How to use this License for your documents	50

List of Tables

4-1. The Standard LCDd Input Keys	12
5-1. HD44780: Power Connections	15
5-2. HD44780: Default Keystrokes	17
5-3. HD44780: 4-bit Pinouts (1).....	20
5-4. HD44780: 4-bit Pinouts (2).....	20
5-5. HD44780: 4-bit Keypad Pinouts	21
5-6. HD44780: Connection Types	28
5-7. T6963 wiring schematic	30
5-8. Arguments	30

Chapter 1. Introduction

1.1. About this Document

This document was meant as a tutorial for LCDproc users. It tries to introduce you into the world of LCDproc giving you an overview of the project. After reading this document you will be able to set up your own LCDproc'ed system.

Note: Please note that this document is still "under construction". We hope to finish it until the final release of LCDproc, which will probably take place in May or June 2002. If you run into any trouble feel free to write to the LCDproc mailing list. See <http://lcdproc.omnipotent.net/mail.php3> for details on how to subscribe to the list.

Therefore you might want to have a look at <http://lcdproc.sourceforge.net/docs/>, to get the latest version of this document (, unless you want to generate it yourself from the docbook files in the CVS).

This document was initially written for LCDproc 0.4.3. At the time of writing there had already been the "LCDproc User's Guide" written by William W. Ferrel in 1999. His version covered an early version of LCDproc and therefore concentrated on Matrix Orbital displays.

William's document was "recycled" for the description of the Matrix Orbital display driver and for other parts of this document.

In several other places e-mails and other documents have been included in this document. The authors of those are listed below every such document.

1.2. What is LCDproc?

LCDproc is a client/server suite including drivers for all kinds of nifty LCD displays.

The server 'LCDd' makes it possible to display text and other data on an LCD display. As well LCDd can handle certain input devices.

Support for devices is added by drivers. We distinguish between output and input drivers. LCDd currently supports only one single output driver, which may at the same time handle input. Nevertheless several input (only) drivers are supported.

Currently there are drivers for several serial devices: Matrix Orbital, Crystal Fontz, Bayrad, LB216, LCDM001 (kernelconcepts.de), Wirz-SLI and PIC-an-LCD; and some devices connected to the LPT

port: HD44780, STV5730, T6963, SED1520 and SED1330. There are input (only) drivers for LIRC and joysticks.

Clients can connect to LCDd through common sockets.

Various clients are available. The "main" client lcdproc, which is shipped with the LCDproc distribution, can display things like CPU load, system load, memory usage, uptime, and a lot more.

1.3. The LCDproc Server - LCDd

LCDd is one of those well known *NIX daemons. BUT it's not just *one* daemon. It's the one that is supposed to drive your LCD ;)

LCDd can either be run from the command line or automatically by the init scripts shipped with the distribution.

As other daemons, LCDd has to be configured. In this respect a lot has changed since LCDproc 0.4.1. While LCDd retrieved all its configuration settings from the command line in 0.4.1, it now has a configuration file, which is normally `/etc/LCDd.conf`.

1.4. The LCDproc "Main" Client - lcdproc

While LCDd only offers the functionality of displaying text on a display, lcdproc actually retrieves data worth displaying.

lcdproc gets its information from the `/proc` filesystem.

lcdproc can connect to an LCDproc server either on the local system or on a remote system as long as it is reachable. It extracts the same statistics regardless of where it sends this information. The statistics it gathers include CPU utilization, memory utilization, disk utilization, network utilization, system uptime, time, and date, and so on. It displays this information in assorted ways, and can be tailored to taste.

1.5. Other LCDproc Clients

As it is rather simple to write an LCDproc client, you can find various clients on the Internet.

Unfortunately we cannot provide a list of LCDproc clients (yet). So, have fun searching google (<http://www.google.com>) or freshmeat (<http://freshmeat.net>). Simply type 'lcdproc' for the keyword.

Chapter 2. How to Obtain LCDproc

2.1. Versioning

At the time of writing there were four versions of LCDproc floating around on the Internet.

LCDproc 0.4.1

LCDproc 0.4.1 was the last "stable" release of LCDproc. It still uses the old command line configuration.

Warning

There are known security problems with LCDproc 0.4.1. A remote exploit is possible. An attacker can make use of some buffer-overflows in the client communication code of LCDd, in order to get root access to your system. Therefore running LCDproc 0.4.1 is NOT RECOMMENDED!

LCDproc 0.4.2

LCDproc 0.4.2 was meant to be the next stable release of LCDproc. As 0.4.1 it uses the old command line configuration.

Note: The known issues about 0.4.1 had been fixed. Unfortunately the guy who had the last pending patches on his box kind of "vanished" from the LCDproc mailing list. Of course you can get 0.4.2 from CVS. BUT its current state is unknown.

LCDproc 0.4.3

LCDproc 0.4.3 is the current stable release of LCDproc. It no longer uses the old command line configuration, but introduces the configuration file `/etc/LCDd.conf`. As well the driver API has slightly changed, which should not make older drivers incompatible, but might cause some trouble. As well the way LCDd reports messages (including error messages) has changed.

Note: Even though LCDproc 0.4.3 is the most stable version we can offer, it is still not perfect.

LCDproc 0.4.4

LCDproc 0.4.4 will be the next stable release of LCDproc.

Note: The 0.4.x branch will only be slightly updated. E.g. we plan better support of lcdproc (the client) for *BSD. Yet, the active development will move to 0.5 now.

As 0.5 will not be stable in the near future, we may also work on porting drivers from 0.4.1 to 0.4.4 on demand. We do not have all the hardware LCDd supports. So, we only work on drivers that can be tested by YOU.

As well, the development and inclusion of new drivers is possible.

LCDproc 0.5

LCDproc 0.5 is the development version of LCDproc. Everything is possible ;) Drivers can already be loaded at runtime. We will also work on client supplied menus and other nifty stuff.

Warning

From time to time LCDproc 0.5 might not even compile due to drastic changes. You have been warned!

Furthermore older drivers will NOT work with LCDproc 0.5 AT ALL! Feel free to port a driver that has not been ported to 0.5 yet ;)

2.2. Download Last Day's CVS Version of LCDproc as a Tarball

There are nightly distributions of the CVS branches of LCDproc. You can download them from <http://lcdproc.sourceforge.net/nightly/>

To extract the files run either

```
$ tar xvfz lcdproc-CVS-*.tar.gz
```

or

```
$ bunzip2 -c lcdproc-CVS-*.tar.bz2 | tar xv
```

2.3. Download The Latest Version of LCDproc from CVS

Of course you can download the latest stuff from CVS via anonymous login.

Create a "cvs" directory somewhere on your machine (not really necessary but useful):

```
$ mkdir ~/cvs
$ cd ~/cvs
```

Login to CVS:

```
$ cvs -d:pserver:anonymous@cvs.lcdproc.sourceforge.net:/cvsroot/lcdproc login
```

(Hit enter when prompted for a password.)

Get the files from CVS:

```
$ cvs -d:pserver:anonymous@cvs.lcdproc.sourceforge.net:/cvsroot/lcdproc checkout -r stable-0-4-3 lcdproc
```

Once you've done that and want to update the downloaded files to the latest stuff you can use the "update" command of CVS (Make sure you have logged in to CVS first.):

```
$ cvs -d:pserver:anonymous@cvs.lcdproc.sourceforge.net:/cvsroot/lcdproc update -r stable-0-4-3 lcdproc
```

Now that you have downloaded the files you can prepare them for compiling, but first you should (you don't have to) copy them to another place on your machine:

```
$ mkdir ~/lcdproc-cvs
$ cp -f -R ~/cvs/lcdproc ~/lcdproc-cvs/`date +%Y%m%d`
$ cd ~/lcdproc-cvs/`date +%Y%m%d`
```

2.4. Debian GNU/Linux

Debian GNU/Linux users can get the debian package of LCDproc. Please note that there may be different versions in stable, testing, unstable and experimental.

Provided apt is configured properly (/etc/apt/sources.list etc.). You should be able to install the package running:

```
$ su
Password: top secret
# apt-get install lcdproc
# dpkg -l lcdproc
```

The last command will print the version of the package that has been installed.

2.5. Gentoo Linux

Gentoo Linux users can build lcdproc from the portage tree.

```
$ su
Password: top secret
# emerge lcdproc
```

Chapter 3. Installation

3.1. Build LCDproc

Now that you have downloaded the LCDproc distribution you can start building it.

Note: If you have installed the debian package with apt-get (or another debian tool) or built lcdproc using gentoo portage, you can skip this chapter.

If you're building this version from CVS, you'll need autoconf, automake, aclocal & autoheader installed.

If you have autoconf and friends, run:

```
$ sh autogen.sh
```

This produces the configure script and supporting files. It has already been run if you using the tarball distribution.

Once the above command has run, the rest is pretty standard:

```
$ ./configure --help
```

Read about the options, figure out what to use.

```
$ ./configure --prefix=/usr/local --enable-drivers=ncurses,cfontz
```

Be sure to replace /usr/local with the prefixdir you want (e.g. /usr for RedHat) and ncurses,cfontz with a comma-separated list of drivers you want.

```
$ make
```

Congratulations: You have just compiled your version of LCDproc ;)

3.2. Install LCDproc Directly From The Sources

If you want to install LCDproc more or less permanently you can run:

```
$ su
```

```
Password: top secret
# make install
```

Note: make install is absolutely OPTIONAL. You can also run LCDproc directly from the source directory. See below for details.

3.3. Generate And Install Packages of LCDproc

As an alternative (which is actually better ;) to installing directly from the sources you can generate packages using the packaging tool EPM.

First of all you may need to download EPM from <http://www.easysw.com/epm/> and install it according to the instructions that are included in its source distribution.

Debian users (who do not want to download the official debs via apt-get) can of course use the debian package of epm:

```
# apt-get install epm
```

Note: There are of course other and maybe better ways to generate packages for your system. The reason for us to choose EPM was that it provides the developers with a tool that makes it possible to write one list file for all platforms defining what the resulting package is meant to look like. This way we do not have to learn all the package managing tools of the different platforms that are supported by LCDproc.

To generate an LCDproc package follow these instructions:

Tip: It is of certain importance that you have run `./configure` with the correct pathname settings for your system. Otherwise the resulting package will install the files in the wrong directories.

```
$ epm -v -f native LCDproc
```

Note: Generating an RPM package as a non-root user will fail, RPM wants to generate the files from the tree under `/usr/src/RPM`, which you do not have write access to as a non-root user. If you want to generate the package as a non-root user anyway, you may want to follow these instructions.

A workaround for the described problem is creating a file named `~/ .rpmmacros` which contains:

```
%_topdir ~/rpm
```

Important: `~/rpm` must contain the same tree usually found under `/usr/src/RPM`

Unfortunately `epm` does not read `~/rpmmacros` and of course returns warnings. Don't worry! That's OK ;)

In order to actually install the generated package follow the instructions in your system's manual.

Chapter 4. LCDproc Configuration

4.1. Configure LCDd

As mentioned in the introduction LCDd, the LCDproc server, now (with version 0.4.3 of LCDproc) has its own configuration file, which is normally `/etc/LCDd.conf`.

Note: If you have not installed LCDproc from the sources the configuration file might have a different location. You should find it when making your system's package manager list all the files in the LCDproc package.

The format of the `/etc/LCDd.conf` is ini-file-like.

It is divided into sections that start at markers that look like `[section]`. Comments are all line-based comments, and are lines that start with `'#'` or `';`.

The server has a 'central' section named `[server]`. Further each driver has a section which defines how the driver acts. Those sections start with `[drivername]`.

The drivers are activated by specifying them in a `driver=` line in the server section, like:

Example 4-1. `LCDd.conf`: Specify which driver to use

```
Driver=curses
```

This tells LCDd to use the curses driver. The first driver specified here that is capable of output functionality will be used as 'the' output driver. All extra drivers can only serve as input. The default driver to use is curses.

Warning

If LCDd is started automatically by an init-script using the curses driver will lock `/dev/tty1`! So, be careful about what you are doing here.

The drivers can read their own options from the config file, but most of them don't do this yet. They expect 'command-line'-format parameters that were previously placed on the command line. These parameters can be given to the driver in the following way:

Example 4-2. LCDd.conf: Compatibility mode for drivers written for 0.4.1

```
Arguments="place arguments here"
```

The arguments between the quotes are passed to the driver. As said before these are the same arguments that would have been passed to the driver using the old command line format of

LCDd -d driver "driverargs"

Note: The -d option still works, but does not allow driverargs any more.

If -d is specified on the command line, the Driver= options in the config file are ignored.

4.1.1. LCDd.conf: The [server] Section

The [server] section of the LCDd.conf contains the settings for the LCDproc server LCDd.

Driver=

Tells the server which driver(s) to use. See above for details

Note: The default setting is Driver=none which makes the server exit right after the start. This is necessary to avoid trouble with package installations.

Bind=

Tells the server to bind to the given interface. Defaults to Bind=127.0.0.1 which is actually the safest variant.

Port=

Tells the server to listen to this specified port. Defaults to 13666.

ReportLevel=

Sets the reporting level. Defaults to 2 (warnings and errors only).

ReportToSyslog=

Should we report to syslog instead of stderr ? Defaults to no.

WaitTime=

Sets the default time in seconds to display a screen.

User=

User to run as. LCDd will drop its root privileges, if any, and run as this user instead. Defaults to User=nobody.

Note: If you want to use the server menu, to shutdown or reboot your system, you will have to set this to root. Otherwise LCDd does not have the privileges to run commands like **init 6**.

ServerScreen=

Enables the server screen even when other screens are active. Defaults to no.

Foreground=

The server will stay in the foreground if set to true. Otherwise the server will fork to background and report to syslog. Defaults to yes.

4.1.2. LCDd.conf: The [input] Section

The [input] section enables you to set some general ("global") options related to the way LCDd handles input "events".

Table 4-1. The Standard LCDd Input Keys

Keyname	Function	
	Normal context	Menu context
PauseKey	Pause/Continue	Enter/select
BackKey	Back (Go to previous screen)	Up/Left
ForwardKey	Forward (Go to next screen)	Down/Right
MainMenuKey	Open main menu	Exit/Cancel

Note: The way input keys are handled will change completely in LCDproc 0.5.

4.1.2.1. LCDd.conf: [input] Section: The Free*Key Options

The Free*Key Options are interesting for those users whose keypad (or other input source) has got only four (or less) keys. In this case LCDd would normally use all those keys to enable you to control the server menu and the other server functions (like switching to the next screen). If these keys are used by a client to do anything but noticing that the user has e.g. entered the menu or switched to another screen, this will probably cause trouble, as one keypress has several effects at the same time.

The Free*Key options enable you to influence this behaviour.

FreePauseKey=

If set to yes, this prevents LCDd from handling the PauseKey itself. Then a client can request the key 'A' and handle it.

FreeBackKey=

If set to yes, this prevents LCDd from handling the BackKey itself. Then a client can request the key 'B' and handle it.

FreeForwardKey=

If set to yes, this prevents LCDd from handling the ForwardKey itself. Then a client can request the key 'C' and handle it.

FreeMainMenuKey=

If set to yes, this prevents LCDd from handling the MainMenuKey itself. Then a client can request the key 'D' and handle it.

Tip: Even if you "free" the above keys, you can still use the menu provided you set FreeMainMenuKey to no.

4.1.3. LCDd.conf: The Drivers Section

As mentioned earlier, each driver has its own section in the `LCDd.conf`.

The settings are more or less self-explanatory. So, read through the section of your driver and change everything necessary.

4.2. The LCDproc Init Scripts

The LCDproc distribution contains init scripts for RedHat- and Debian-based GNU/Linux distributions. You can find them in the `docs/` directory of the LCDproc sources.

Note: The init scripts are generated using autoconf. So, again it is important that you have run `./configure` with the correct options for your system.

Refer to your system's manual on how to install the scripts.

4.2.1. init-LCDd

The file `scripts/init-LCDd.*` is the init script for the LCDproc server LCDd. It does not require modification.

4.2.2. init-lcdproc

The file `scripts/init-lcdproc.*` is the init script for the LCDproc "main" client lcdproc. As lcdproc itself does not (yet) have a configuration file you may want to modify the options the init script passes to lcdproc.

Note: You can retrieve a listing of all options of lcdproc running `lcdproc --help`.

The lcdproc init script retrieves the options it will pass to lcdproc from the file `/etc/lcdproc.conf` (`scripts/lcdproc.conf` in the sources distribution).

Example 4-3. `lcdproc.conf`: Modify the option passed to lcdproc

```
# /etc/lcdproc.conf
#
# Configuration file of the main LCDproc client "lcdproc"
# NOTE: The configuration file of LCDd is /etc/LCDd.conf

# Set SCREENS to the screens you want lcdproc to send to LCDd
# lcdproc --help will give you a list of the screens available

SCREENS="C X"
```

In this example lcdproc will only send information on the CPU usage [C] and system load [X] to the server.

Chapter 5. LCDproc Drivers

This chapter contains the documentation of each LCDproc driver, which may include the installation process of the hardware as well as the configuration of LCDd.

5.1. The HD44780 Driver

There are several ways of wiring up the HD44780 devices. Your choice will probably be governed largely by your ability to wire up each one and/or a desire to use the device with other programs.

The LCDproc HD44780 driver supports the following connections on a parallel port:

- 4-bit
- 8-bit (winamp style)
- extended 8-bit (LCD + LED bargraph)
- serial LPT

It also supports a PIC-an-LCD connected to a serial port and the USBLCD interface.

The driver also lets you use multiple displays as a single virtual display. For example, a 4, 2 and 1 line display can be used to form a 7 line display. The number of displays is limited by the individual HD44780 driver.

5.1.1. Connections

5.1.1.1. Common connections for all connectiontypes

No matter what connectiontype you choose, you will always need some connections. They are explained here.

5.1.1.1.1. Power

All variants use the same method of obtaining power. i.e., for each LCD:

Table 5-1. HD44780: Power Connections

LCD	pin	signal
1	GND	(connect to any of pins 18 - 25 of you parallel port)

LCD	pin	signal
2	+5V	
3	Vadj	(contrast)

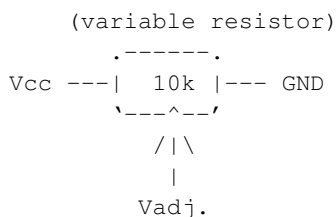
Warning

Always double check your power connection, your display will probably NOT survive a reversely connected supply !

There are several ways to get 5V:

- Connect to a 5V line intended for disk drives (the red wire is 5V, black is GND).
- Get it from a joystick port (pin 1 and 9 are 5V, 4, 5 and 12 are GND). It seems that some soundcards can use these lines for communication, so if you want to use this first check whether it really gives a 'clean' 5V.
- If you don't have a backlight, you can sometimes get the needed mA's from the LPT port itself. Connect a few diodes from the data pins to a capacitor and you have the 5V. If it's strong enough is another question...
- Get it from the keyboard connector. I do not recommend to use this with a backlight, as the keyboard connector is often protected with a fuse of 100mA or 200mA.

Figure 5-1. HD44780: Connecting the contrast adjusting pin (Vadj.)



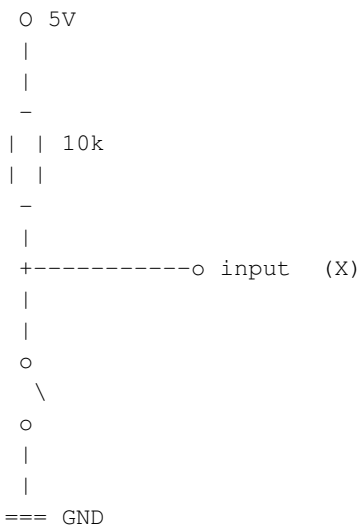
5.1.1.1.2. Keypad

You can connect a keypad with all connection types. The maximum supported number of keys differs per type. There are several ways to connect the keys to the input pins.

5.1.1.1.2.1. Direct Keys

If you connect a key like sketched below, then you can only connect one key per input pin. It is a simple solution if you need only few keys.

Figure 5-2. HD44780: Direct Keys



By default, the following keystrokes are generated by the different keys:

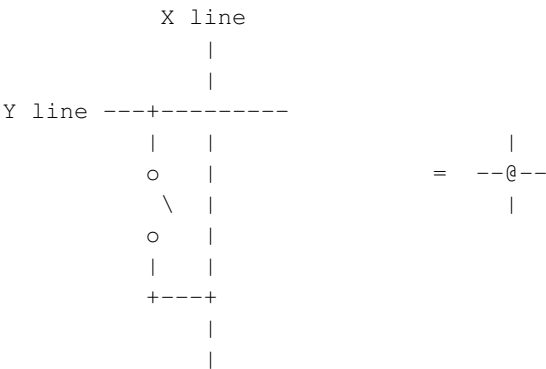
Table 5-2. HD44780: Default Keystrokes

X0	A
X1	B
X2	C
X3	D

5.1.1.1.2.2. Matrix Keys

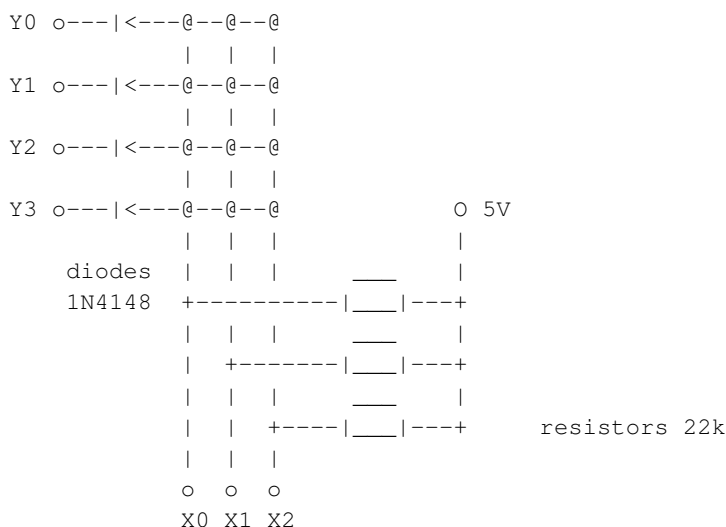
Using a matrix, we can connect much more keys. To simplify the drawing here, we replace all switches with an @ symbol:

Figure 5-3. HD44780: Single Matrix Key



We connect the matrix of keys like this:

Figure 5-4. HD44780: Complete Key Matrix



As you can see, you need 1 resistor per X line, and 1 diode per Y line. Lcdproc will presume that you have a keypad with a layout like a telephone connected, with X and Y lines connected as show. To be more precise, it assumes this:

Figure 5-5. HD44780: Keypad Layout

	X0	X1	X2	X3
Y0	1	2	3	A
Y1	4	5	6	B
Y2	7	8	9	C
Y3	*	0	#	D

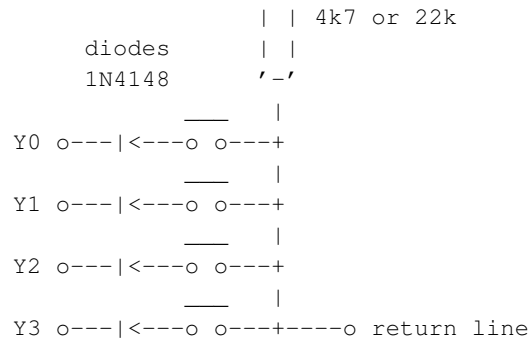
If you only need 10 keys, leave the rest away. However, the lcdproc menu is controlled by the keystrokes A to D. You should modify and recompile the driver to get an other keypad layout.

You can buy arrays of keys that are connected like this in the electronics shop. They usually call it a matrix keypad. To hook it to lcdproc, you would only need to add the resistors and diodes.

If you want to use just one return line, for example with the serialLpt wiring, it looks (completely drawn) like this:

Figure 5-6. HD44780: One Return Line



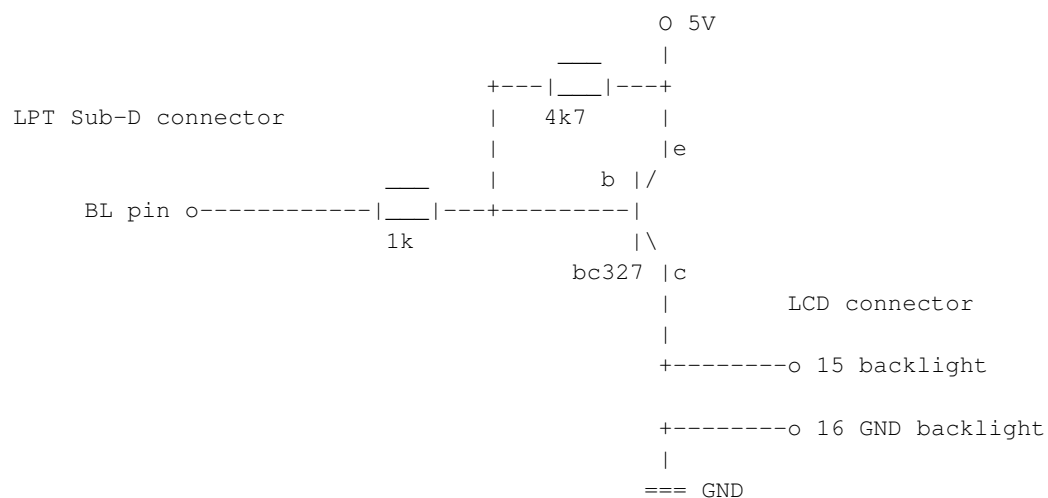


Tip: If the driver generates keypresses without that you actually press a key, it might be that the unconnected input lines are picking up electromagnetic waves from the air. In that case connect the unconnected input lines (pin 10, 11, 12, 13 and 15 of the LPT) to VCC = 5V.

5.1.1.1.3. Backlight

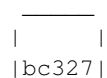
A small extension allows you to switch the backlight of the display on and off. At the moment only the 4bit and winamp connection types support this. The extension uses one output pin, you cannot use that pin for other functions anymore. The wiring looks like this:

Figure 5-7. HD44780: Backlight Wiring



Note: 4k7 means 4,7 kohm.

The BC327 transistor has the following connections:





Caution

Sometimes the backlight connections are not on the 'main' connector, but on the side. If that is the case, there is usually NO RESISTOR present to limit the current through the LEDs. Therefor you should then add a resistor after the transistor of about 10 ohm (see display documentation).

Tip: If you want the backlight to light a bit while it's switched 'off', you can add a resistor bypassing the transistor from e to c, with a value of, say 47ohm or 22ohm. (My 4x20 has an internal resistor of 6ohm, so with 47 ohm extra it lights at only 1/9th. I like this. Joris.)

5.1.1.2. 4-bit

This is originally based on "lcdtext" (by Matthias Prinke).

Table 5-3. HD44780: 4-bit Pinouts (1)

printer port	LCD
D0 (2)	D4 (11)
D1 (3)	D5 (12)
D2 (4)	D6 (13)
D3 (5)	D7 (14)
D4 (6)	RS (4)
D5 (7)	RW (5) (LCD3 - 6) (optional - pull all LCD RW low)
D6 (8)	EN (6)
D7 (9)	EN2 (LCD2 - 6) (optional)

If you want to connect more than two displays to the parallel port then wire D5 (pin 7) to the enable line (pin 6) of the third LCD. Then for displays four to seven use:

Table 5-4. HD44780: 4-bit Pinouts (2)

printer port	LCD
--------------	-----

printer port	LCD
STR (1)	EN4
LF (14)	EN5
INIT (16)	EN6
SEL (17)	EN7

The optional keypad can be connected as follows:

Table 5-5. HD44780: 4-bit Keypad Pinouts

printer port	LCD
D0 (2)	Y0
D1 (3)	Y1
D2 (4)	Y2
D3 (5)	Y3
D4 (6)	Y4
D5 (7)	Y5
nSTRB (1)	Y6
nLF (14)	Y7
INIT (16)	Y8
nSEL (17)	Y9
nACK (10)	X0
BUSY (11)	X1
PAPEREND (12)	X2
SELIN (13)	X3
nFAULT (15)	X4

The optional backlight wiring should be connected to D5, pin 7.

5.1.1.3. 8-bit "Winamp"

This type of connection should work with winamp.

Figure 5-8. HD44780: "Winamp" wiring

printer port	LCD
D0 (2)	D0 (7)
D1 (3)	D1 (8)
D2 (4)	D2 (9)
D3 (5)	D3 (10)

D4 (6)	D4 (11)
D5 (7)	D5 (12)
D6 (8)	D6 (13)
D7 (9)	D7 (14)
nSTRB (1)	EN (6)
nLF (14)	nRW (5) (EN3 6 - LCD 3) (optional)
INIT (16)	RS (4)
nSEL (17)	EN2 (6 - LCD 2) (optional)

If you want the display to work with the Winamp plugin, wire nLF (pin 14) to nRW of your LCD. You can then use the plugin in bidirectional mode (which is much faster). With 3 connected LCDs this is not possible. Note from Benjamin: I haven't tried using winamp while having the third LCD connected to this line.

The optional keypad can be connected as follows:

Figure 5-9. HD44780: "Winamp" wiring - Keypad

printer port	keypad
D0 (2)	Y0
D1 (3)	Y1
D2 (4)	Y2
D3 (5)	Y3
D4 (6)	Y4
D5 (7)	Y5
D6 (8)	Y6
D7 (9)	Y7
nLF (14)	Y8
INIT (16)	Y9
nACK (10)	X0
BUSY (11)	X1
PAPEREND (12)	X2
SELIN (13)	X3
nFAULT (15)	X4

The optional backlight wiring should be connected to nSEL, pin 17.

5.1.1.4. 8-bit "lcdtime"

This is originally based on "lcdtime" (by Benjamin Tse <blt@ComPorts.com>) and allows you to combine the LCD with a LED bargraph. The LCD is driven by LCDproc and the LEDs by another program such as portato. Further details can be obtained from:

<http://metalab.unc.edu/pub/linux/system/status/lcdtime-0.2.tar.gz>

<http://metalab.unc.edu/pub/linux/system/status/meter-0.2.tar.gz>

<http://metalab.unc.edu/pub/linux/system/status/portato-1.2.tar.gz>

Note: Theoretically this wiring sends the data over twice as slow as the winamp or ext8bit wirings, because it only sends 4 bits at a time.

The LCD connections are:

Figure 5-10. HD44780: "lcdtime" wiring

printer port	LCD
D0 (2)	D0 (7)
D1 (3)	D1 (8)
D2 (4)	D2 (9)
D3 (5)	D3 (10)
D4 (6)	D4 (11)
D5 (7)	D5 (12)
D6 (8)	D6 (13)
D7 (9)	D7 (14)
nSEL (17)	-
nSTRB (1)	RS (4)
nLF (14)	RW (5) (LCD2 - 6) (optional - pull all LCD RW low)
INIT (16)	EN (6)

See the lcdtime tar-ball (above) for full details of the bargraph connections.

The optional keypad can be connected as follows:

Figure 5-11. HD44780: "lcdtime" wiring - keypad

printer port	keypad
D0 (2)	Y0
D1 (3)	Y1
D2 (4)	Y2
D3 (5)	Y3
D4 (6)	Y4
D5 (7)	Y5
D6 (8)	Y6
D7 (9)	Y7
nSTRB (1)	Y8
nSEL (17)	Y9
nACK (10)	X0
BUSY (11)	X1
PAPEREND (12)	X2
SELIN (13)	X3

nFAULT (15) X4

The backlight wiring should be attached to nSEL, pin 17. Because the portato program (mentioned above) also uses this pin to control the bargraph, you cannot use the backlight control together with the bargraph.

5.1.1.5. Serial LPT

This interface uses a handful of wires to interface to the HD44780. Suitable for high noise, long connections. Designed by Andrew McMeikan <andrewm@engineer.com>. The original wiring and driver can be found at:

<http://members.xoom.com/andrewmuck>

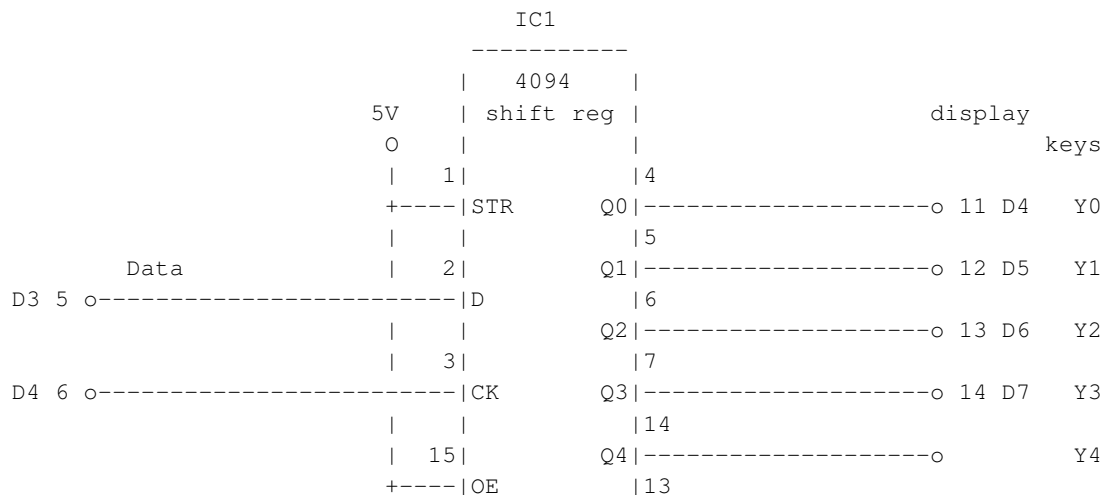
I (Joris) have extended this driver and the wiring a bit. It now supports keys again (it had earlier supported keys, but some time did not).

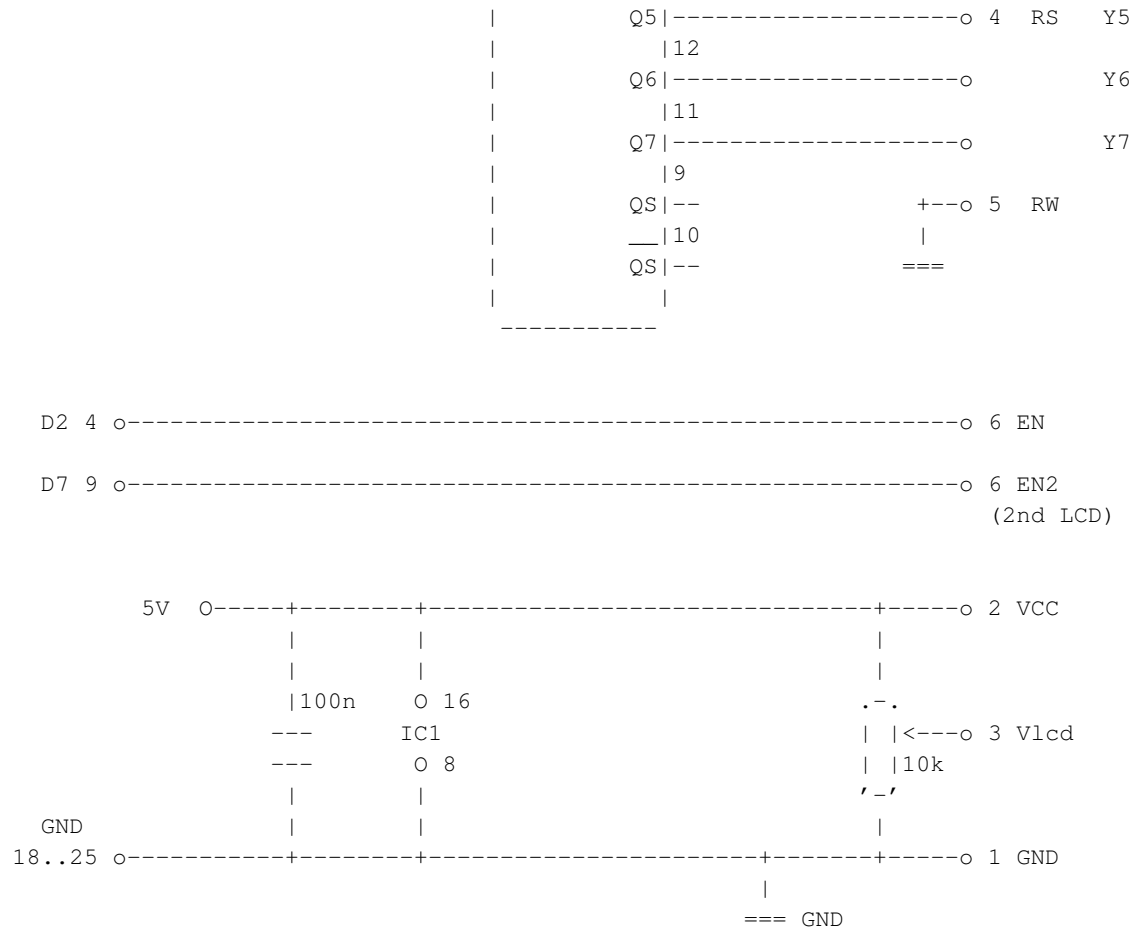
Further I have extended the driver and the wiring to be able to run using 2 instead of 3 output pins. That's even one less pin ! :)

Of course the use of fewer lines than the other wirings can not stay without drawbacks. In this case the simplicity of the long feeding wires is compensated by some intelligence in the decoding of the data. If you have no experience with the soldering iron, I do not recommend to build this wiring.

OK, so here is the wiring. First of the 'simple' 3 wires version. IC1 is the shift register, a 4094. Do not forget to connect the 5V to pin 16 and GND to pin 8 of the IC.

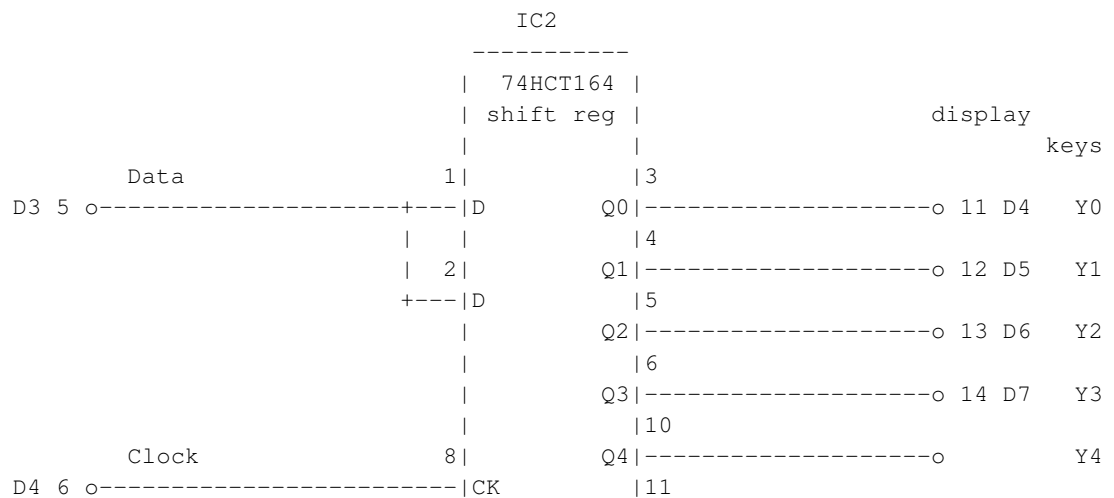
Figure 5-12. HD44780: Serial LPT wiring ('simple')

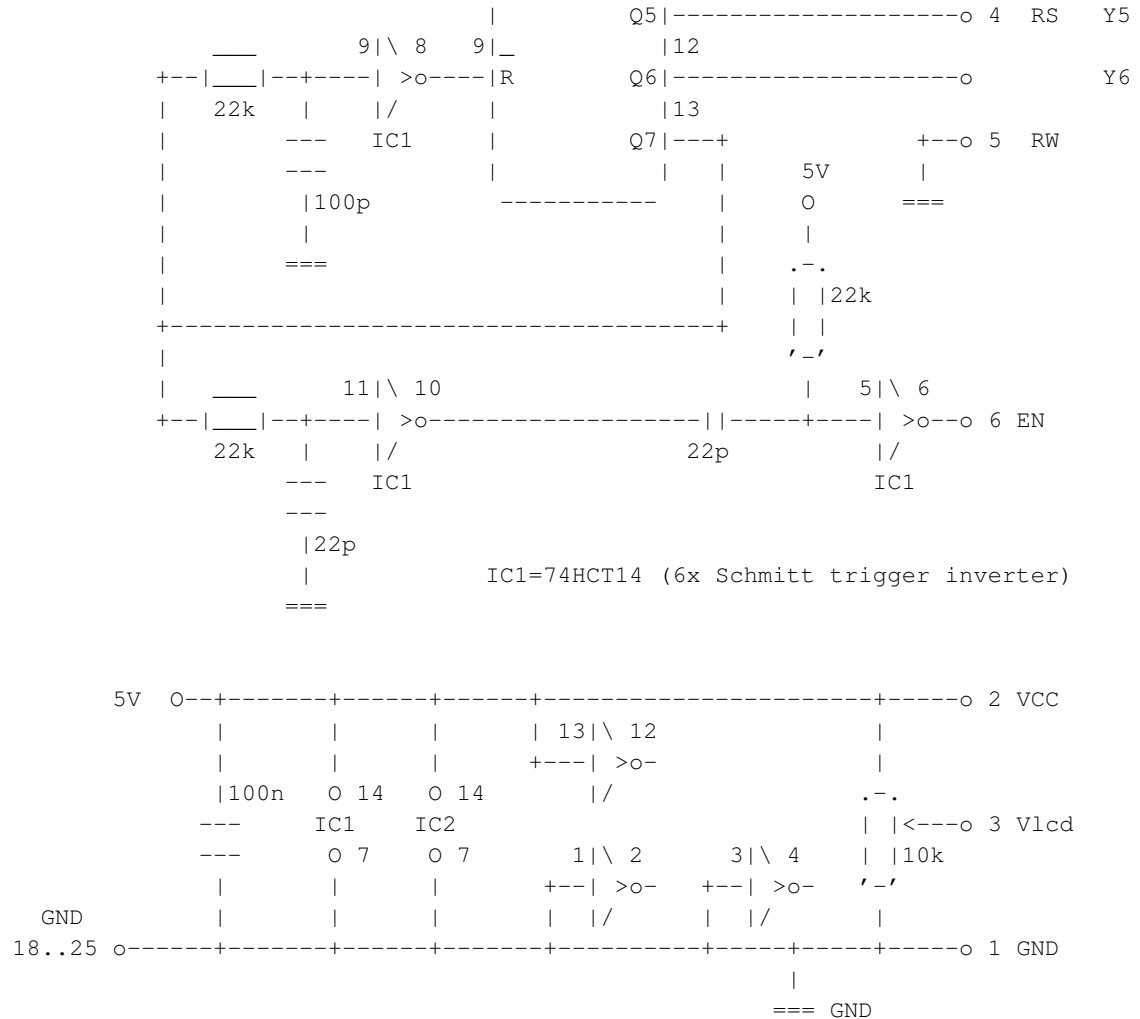




The second possible wiring is with 2 output lines. This one is a bit more complex. If you do not understand the schematic, do not build it.

Figure 5-13. HD44780: Serial LPT wiring ('complex')





5.1.1.5.1. Serial LPT Keypad

Note: To understand this part of the serialLpt documentation, you also need to read the keypad section in this document.

serialLpt wiring supports a keypad. The 3 wires version supports 8 keys, or if you use multiple return lines up to $8 \times 5 = 40$ lines. The 2 wires version supports 7 keys, or with multiple return lines $7 \times 5 = 35$ keys.

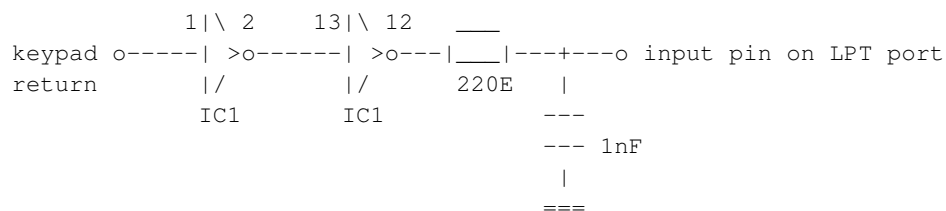
Figure 5-14. HD44780: Serial LPT - Keypad return lines

nACK (10)	X0
BUSY (11)	X1
PAPEREND (12)	X2
SELIN (13)	X3

nFAULT (15) X4

On lines longer than, say a meter, you should buffer the return line(s). If you only have 1 return line, you can buffer it with two remaining buffers from the 74HCT14:

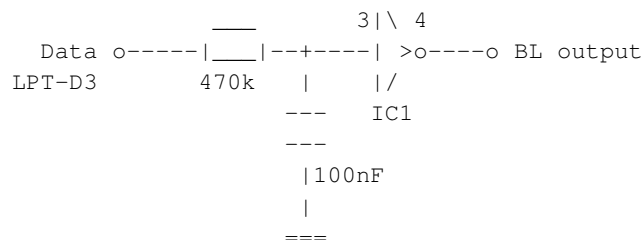
Figure 5-15. HD44780: Serial LPT - Keypad return lines buffered



5.1.1.5.2. Serial LPT Backlight

Also a backlight is supported. You will also need a port from the 74HCT14 for that. The BL output below should be connected to the BL input in the backlight section

Figure 5-16. HD44780: Serial LPT - Backlight extra circuit



5.1.1.6. PIC-an-LCD serial device "picanlcd"

The PIC-an-LCD module is also supported. It is not connected to the LPT port but to a serial port, which saves you from a lot of potential problems. To use it, specify the device to which you have connected the module in the config file with the Device= setting. The default is /dev/lcd. It does not support a keypad nor backlight switching.

5.1.1.7. USBLCD Interface

The USBLCD Interface lets you use HD44780 compatible LCDs via the USB port. Make sure you have the USBLCD kernel module installed and loaded, since 2.4.20 it is part of the Linux Kernel or can be downloaded from www.usblcd.de (<http://www.usblcd.de>). If you want to use another device file then the

default `/dev/usb/lcd` you can specify this with the `"Device="` option in the config file. It is also recommended to set `"Backlight=yes"` and `"Keypad=no"`.

5.1.2. Compiling

Make sure that the HD44780 files are built when you run configure. This can be done by specifying `"--enable-drivers=all"` or by `"--enable-drivers=hd44780"`.

5.1.3. Configuration

Since LCDproc 0.4.3 the HD44780 driver can be configured from the configfile (LCDd.conf).

The [HD44780] section is more or less self-explanatory. Yet, a list of the possible connection types has proved useful. The following table should make it clear, how the ConnectionType is to be set.

Table 5-6. HD44780: Connection Types

Wiring / Display Type	ConnectionType=
HD44780 4bit Wiring	4bit
HD44780 8bit Wiring ("lcdtime")	8bit
HD44780 Serial LPT Wiring	serialLpt
HD44780 8bit Wiring "winamp" Style	winamp
PIC-an-LCD serial device "picanlcd"	picanlcd
USBLCD Interface	usblcd

Tip: The above table might be outdated, when you read this. If you suspect this is the case, you might want to have a look at `server/drivers/hd44780-drivers.h` which contains the actual translation code.

5.1.4. Running

Modify the LCDd.conf file before you run LCDd. In this config file are detailed instructions on how to configure the HD44780 driver.

Then as usual, start LCDd with the correct config file:

E.g. `LCDd -c ./LCDd.conf`

If you want to override the driver selection in LCDd.conf then use:

LCDd -c ./LCDd.conf -d HD44780

If you use this, the HD44780 driver will read the options from the config file anyway.

5.1.5. Questions & Answers

1. What should a properly powered HD44780 LCD show?

It will show lines filled with blocks. The position of these lines will depend on the LCD size. A 2x20 will fill the first line, a 4x20 will fill the first and third lines and a 2x40 will fill the first line on both halves on the LCD.

2. What should a properly connected (but not initialised) HD44780 LCD show?

The fact of being properly connected won't affect the content of the display. So it should be lines filled with blocks like in the preceding question.

3. What should a properly initialised HD44780 LCD show?

Once initialised, the HD44780 will go blank, then show what it receives from LCDproc. Generally it will be something like "Clients: 0 Screens: 0".

4. What should be the BIOS LPT port type?

They should all work. We are only using the features of the SPP (normal) type, so set it to this if in doubt.

5. What is the difference between numerous connection variants and which one is the best, simplest or most compatible.

There's no real answer to that, if there was a real better connection type we would only use this one. Connections using 8 bits are twice as fast as the one using 4 bits. The winamp connection is compatible with the WinAmp plugin.

5.1.6. Miscellania

This text has originally been taken from a message by Bill Farrow
<bffarrow@arrow.bsee.swin.edu.au>.

Updated February 2000, Benjamin Tse <blt@ComPorts.com>

Updated October 2001, Joris Robijn <joris@robijn.net>

Converted to docbook March 2002, Rene Wagner <reenoo@gmx.de>

Updated April 2002, Rene Wagner <reenoo@gmx.de>

Added the Q&A September 2002, Guillaume Fillion <gfk@logidac.com>

5.2. The Toshiba T6963 Driver

This section talks about using LCDproc with LCD displays that use the T6963 chipset. Usually, this chipset is used on big character LCD displays that can often act as a screen.

5.2.1. Connections

Table 5-7. T6963 wiring schematic

Parallel port	LCD
1 (Strobe)	5 (WR)
2-9 (Data)	11-18 (Data)
14 (Autofeed)	7 (CE)
16 (Init)	8 (C/D)
17 (Slct)	6 (RD)
18 (GND)	3 (GND)
+5V	3 (LCD +)

5.2.2. Compiling

Make sure that the T6963 files are built when you run configure. This can be done by specifying "--enable-drivers=all" or by "--enable-drivers=t6963".

5.2.3. Configuration

Since LCDproc 0.4.3 the T6963 driver must be configured from the configfile (LCDd.conf).

At the time of writing, the T6369 driver uses the Arguments option for its configuration. This will be parsed as if it was passed as argument to a program.

Table 5-8. Arguments

-p --port	Select the output port to use [0x378]
------------------	--

-p --port	Select the output port to use [0x378]
-t --type	Select the LCD type (size) [20x6]

Tip: An example configuration could look like this: Arguments="--port 0x378 --type=20x6"

5.2.4. Running

Modify the LCDd.conf file before you run LCDd. In this config file are detailed instructions on how to configure the T6963 driver.

Then as usual, start LCDd with the correct config file:

E.g. **LCDd -c ./LCDd.conf**

If you want to override the driver selection in LCDd.conf then use:

LCDd -c ./LCDd.conf -d T6963

If you use this, the T6963 driver will read the options from the config file anyway.

5.3. Parallel Port Troubleshooting

Unfortunately attaching an LCD module to a parallel port is not trivial.

In most cases it requires soldering abilities and basic knowledge of electronics.

The following hints might be helpful:

5.3.1. Check The Wiring

Wiring errors can easily be made. If you are unexperienced with the soldering iron better have someone solder it for you. Display modules are sensitive to electro static discharges, so touch a grounded (earthed) surface (metal computer case, water pipes...) before you handle these. You may also want to buy a static ground bracelet that you connect to your wrist to a ground. They are usually sold for a couple dollars (or the price of a couple beers in case the dollar is not a valid currency where you live 8) in electronics stores.

5.3.2. Power Source Unregulated / Noisy

Make sure your power supply delivers steady 5 Volts without noise or interruptions. The bare wall plug in transformer is not suitable, though you can make it stabilized by adding an 7805 and a few capacitors. Some noise induced in the supply lines may be tricky to track, even if you have an oscilloscope.

5.3.3. Ground Lift

The power supply wires and especially the GND wires should be a little thicker than the other wires. If GND is not thick enough (or not existent, see 1) the resistance of the wire may cause differing GND potentials in the circuit. This may lead to strange display behaviour. It may also be wise to solder a 100nF capacitor directly to the GND and VDD pins of the display.

5.3.4. Latchup

Never let the supply voltage get much below the IO signal voltage. It may lead to a latchup condition (the LCD will try to get current from the IO ports) which will destroy the controller chip on the display.

5.3.5. Contrast

If you don't see anything on your display it may be that your contrast voltage is set wrong. Turn your contrast potentiometer all the way to the end connected to GND. Contrast is highest then.

Beware: The module you got so ultra cheap may be an enhanced temperature model which needs a negative contrast voltage for sufficient contrast - see chapter 99 on how to make negative voltage.

5.3.6. Parallel Port Voltage

Many modern mainboards and especially notebooks will not nearly output 5V for a logic H as the older parallel ports did, because the operating voltage of computers is lower than 5V these days. I have measured voltages between 2.5V and 4V for logic H, which is barely within specification of the HD44780. If you account RCL of your cable, this may not be enough and can cause unreliable operation.

5.3.7. Enable Signal Rise Time

If you ever read the HD44780 datasheet you will notice that somewhere in the 'signal timing' table is written: 'Enable Signal Rise Time max. 20nS'. That means the Voltage on the HD44780 pin called

'Enable' has to rise from 0 Volts to 5 Volts within 20 Nanoseconds and the other way round. They should better print that in big fat red letters, because most HD44780s are really picky about the enable signal rise time.

That is a Problem: If you count together the bad driving characteristics of the parallel port combined with the capacitance of flat ribbon cable you may easily get an order of magnitude slower rise time. Therefore you should only use really short cable (shorter than 50cm) for connecting the display to the parallel port. It may also be useful to use pull-up resistors on the display module or a schmitt-trigger.

Note: The rise time of a digital output can (usually) not be altered by Software.

5.3.8. EMI

The cable from the parallel port to the display may be sensible to electromagnetic interference and may emit electromagnetic radiation. If you place your cellphone near the cable, you may get unexpected display readings, on the other hand your house neighbour may not be able to listen to his/her favourite radio station any more - so better use shielded cable and put the display in a metallic case, perhaps a computer case.

5.3.9. One or Two Black Lines

If you see one or two black lines on the display it means nothing more than that the display is powered and contrast voltage is present. If one or two black lines appear the controller has not been reset properly by the on chip power on reset generator. No need to worry - it will be reset by the LCDd software. But if the black line will not disappear although the wiring is working, the controller on the display may be defective.

5.3.10. Software Too Fast

If you have a super GHz computer it may happen that the signal timing generated by LCDd is too fast. Adjust DELAYMULT in the source file to a bigger value. Parallel port wirings usually don't permit to read back the busy flag of the controller chip, so timing must be adjust so that the controller never is busy.

5.3.11. LED Backlight

Check whether you need a resistor for your LED Backlight and which value it should have. If you forget the required resistor the backlighting LEDs might become hot and draw excessive current.

5.3.12. HD44780 Compatible

The original HD44780 controller that we advertise to support has become the industry standard for alphanumeric character displays. The original HD44780 is out of production. It has many successors from many manufactures, which sometimes won't tell you that their chips are 'compatible'.

To name a few: KS 0066, KS 0070, KS 0076, LC 7985, NT 3881, SED 1278, ST 7066 ...

5.3.13. Miscellania

This text has originally been taken from a message
(<http://lists.omnipotent.net/pipermail/lcdproc/2002-May/005832.html>) by Robin Adams
<robin@adams-online.de>

Converted to docbook and slightly modified May 2002, Rene Wagner <reenoo@gmx.de>

5.4. The Matrix Orbital Driver (MtxOrb)

This section covers the installation process for the Matrix Orbital LCD module intended for use with LCDproc.

We will examine the installation process of the hardware in small steps, as it is vitally important to pay close attention to detail during hardware installation to avoid damaging equipment.

5.4.1. Matrix Orbital LCD Modules

LCDproc was born out of original tinkering by William Ferrell with one of these LCD modules. Their ease of installation and use (as well as the amazing amount of patience demonstrated by the folks at Matrix Orbital whilst William figured things out) meant one less thing to worry about during the early stages of LCDproc's life.

These 20x4 alphanumeric modules are connected via standard DB-9 cabling and connectors. They draw either 5V or 12V, depending on the module purchased, and are attached with a standard floppy cable connector (with a slightly modified wire configuration).

Once connected, using them is a breeze. They can operate at any number of different baud rates and serial configurations, but normally they run at 19,200 baud, 8-N-1, making them quite quick. Sending ASCII to the module will make it simply display that text at its current cursor position. The module has a built-in

BIOS that recognizes commands (sent by transmitting a single-byte "marker" signifying that a command is on the way, followed by the single-byte command character itself along with any parameters, if needed) allowing the programmer to clear the screen, position the cursor anywhere, define custom characters (up to 8 at a time), draw bar graphs and large numbers, change the LCD's contrast, and so on.

The BIOS included also implements line-wrapping (i.e. writing past the twentieth character on the first row will automatically move the cursor to the first character on the second row), and screen scrolling (i.e. writing past the twentieth character on the fourth row causes the whole screen to scroll up one row, clearing the fourth line and positioning the cursor at the first character on that line).

These modules are fast. Using the auto-line-wrap feature and disabling the auto-scrolling feature, the screen can be updated thirty times per second if **every** character on the screen is changed. If updating less than the whole screen, the LCD can update faster than can be seen by the human eye. This, of course, more than meets LCDproc's needs.

5.4.2. Matrix Orbital Hardware Installation

Regardless of what specific type of hardware you intend to use with LCDproc, installation is usually straightforward, and requires only a few steps. Regardless, you must use caution while working inside your computer system or with any hardware attachments.

Warning

Installing new hardware inside a computer system can be dangerous to both system components and the installer. Use caution whenever adding a component to the inside of your system, altering a power cable, or physically mounting a device inside a computer system.

When installing hardware inside a computer, make sure it's turned off and that its power is disconnected. This is especially important when making changes to power cables (as some LCD modules require).

5.4.2.1. Matrix Orbital LCD/VFD Module Installation

The LCD and VFD modules from Matrix Orbital are relatively straightforward to install. With a small, regular (flat-head) screwdriver, a spare floppy drive power cable, and a bit of luck, installation will take less than an hour.

These installation instructions assume that you are installing the module into a PC or PC-style system (one with AT- or ATX-compliant power cabling) and that you have some idea of where you intend to permanently mount the module. For mounting ideas and tips, refer to the section "Mounting" below.

Before you start: Your Matrix Orbital LCD or VFD module should be clearly marked with an indication of the module's power requirements. It should be either a 5 volt or 12 volt unit. You should have this information available before proceeding.

5.4.2.1.1. Power Cable Modification

The first step in installing the module is making the necessary modifications to a floppy drive power cable in order to provide power to the module. The modifications must be made based on the module's power requirements -- either 5V or 12V -- depending on which module you purchased.

A standard floppy drive power cable has a smaller connection than a "normal" PC power connector. However, like a "normal" power connector, it has four wires: one yellow, one red, and two black. The red wire provides +5V power, and is "hot" or live when the system is powered up. The yellow wire provides +12V power, and is also hot when the system is powered up. Both black wires are ground. [TODO: INCLUDE A FIGURE HERE SHOWING A "STANDARD" FLOPPY CONNECTOR]

One of the hot wires and one of the black wires will not be needed for your module's power connection; they will be completely removed when the power cable modification is complete.

Warning

Do NOT make this modification to a power cable attached to a running system!
Electrocution resulting in personal injury and/or damage to the system can result.

Using a regular screwdriver, press down the small metal locking flap of one of the two black wires on the small end of the cable, and pull the black wire from the connector. Using a pair of needle-nose pliers, squeeze the other end of the same black wire, and pull it out of the large end of the cable. This black wire can be set aside; it will not be used for the module's power connection. Either wire can be safely removed; you may safely remove either wire. [TODO: INCLUDE A FIGURE HERE SHOWING THIS PROCESS]

Next, using the same procedure, remove the unneeded hot wire. If your module is 5V, you do not need the yellow (+12V) wire. Conversely, if your module is 12V, you do not need the red (+5V) wire. The removed wire can be set aside; it will not be used for the module's power connection. [TODO: INCLUDE A FIGURE HERE]

The floppy power connector should now have only two wires attached to it. Leave the larger end alone from now on; these connections are correct (the larger end connects to your system's power mains). Move the two remaining wires to the outside connectors on the small end of the cable. Orientation does not particularly matter here; the connector will fit on the module's receptacle in either orientation. [TODO: A FIGURE HERE]

You should now have a properly modified power connector. When physically attaching this connector to the module, the black (ground) lead should be connected to the pin labelled GND, while the colored (+5V/+12V) lead should be connected to the pin labelled +5V/+12V.

Test the power connection before connecting the data line or mounting the module. Connect the module to the power connector, and the connector to your system's power mains. Turn the system on.

Caution

If the module does not immediately display its initial BIOS screen and light up its backlight (or light up the screen if a VFD module is being used), *immediately* power down the system, disconnect the module and connector, and double-check the modification before trying again. Do NOT leave the system on if the module does not immediately respond; module or system damage could result.

When the LCD powers up and displays its initial BIOS screen, you've gotten the power connection wired properly and can now properly mount the module and make its final connections. Matrix Orbital Corporation sells a PC bay insert mount for the 20x4 and 20x2 modules (LCDproc, however, only supports the 20x4 at present). The inserts provide an easy means of mounting the LCD modules inside a PC using one (for the 20x2) or two (for the 20x4) 5 1/4" bays.

Note: Describing how to physically mount the module in a PC case is beyond the scope of this document; LCDproc's website contains more detailed mounting information and examples.

5.4.2.1.2. Serial Connection

The LCD module uses a standard DB9 serial connector. You can attach the module to your system using a direct cable to the motherboard, or by removing one of your system's serial ports from the back of the case, then connecting it to a standard serial cable to the module.

While connecting the serial cable to the module, be sure to configure the module's serial interface settings. Typically, setting the module to its fastest setting (19,200 baud, 8-N-1) is recommended. The speed settings can be configured from the config file `/etc/LCDd.conf`. If not specified in the config file, the Matrix Orbital module driver in LCDproc default to use these settings.

5.4.3. Copyright

This section was originally part of the `lcdproc.sgml` file by William W. Ferrell <wwf@splatwerks.org>

Slightly modified in order to include it in this document March 2002, Rene Wagner <reenoo@gmx.de>

5.5. The lircin Driver

The lircin driver enables you to use any IR remote control that works with LIRC to control the LCDproc server (LCDd) and/or clients that can handle input.

Of course you need a working LIRC setup. Refer to <http://www.lirc.org> for more information on LIRC itself.

5.5.1. Checking Your LIRC Setup

Basically all you need is a running lircd. And of course you have to start lircd as root.

Also, make sure that the permission of `/dev/lircd` are correct.

5.5.2. Build LCDd with the lircin Driver

You need to add lircin to the `--enable-drivers=...` list.

Then simply run make.

5.5.3. Configure LCDd to Use the lircin Driver

First of all you need to activate the driver by adding a `Driver=lircin` line to your `LCDd.conf`

Example 5-1. LCDd.conf: Activate the lircin driver

```
Driver=mtxorb
Driver=lircin
```

This activates the mtxorb driver as the output driver and the lirc driver as the input driver.

Then you have to modify the `[lircin]` section of your `LCDd.conf`.

5.5.3.1. LCDd.conf: The [lircin] Section

The [lircin] section of the `LCDd.conf` contains the settings for the lircin LCDproc driver.

`lircrc=`

Normally all LIRC clients scan the file `~/ .lircrc`. However, you might want to have a separate file to configure the LCDproc lircin driver only.

This option enables you to specify the file you want the lircin driver to scan.

`prog=`

All LIRC keys are assigned to a program using the `prog=...` option in the `~/ .lircrc` (or the file you have specified with `lircrc=...`).

The `prog=...` line must be the same as in your `~/ .lircrc` (or the file you have specified with `lircrc=...`).

5.5.4. Modify Your `~/ .lircrc`

As mentioned above you can either modify the `~/ .lircrc` or use a separate file for the lircin LCDproc driver (See above for details).

No matter which file you use, you have to add at least the following lines to the file:

Example 5-2. `~/ .lircrc`: Specify the keys for the lircin driver

```
begin
    prog = lcdd
    button = 2
    config = A
end

begin
    prog = lcdd
    button = 4
    config = B
end

begin
    prog = lcdd
    button = 6
    config = C
end
```

```
end

begin
    prog = lcdd
    button = 8
    config = D
end
```

Which buttons you specify here depends on your RC and your LIRC configuration. Anyways, config=A/B/C/D is necessary to control the server menu of LCDd. Of course you can define other keys. Those keys will not be handled by the server but sent to a client. Refer to the documentation of the client you want to use, to find out which keys are necessary for that client.

Chapter 6. Running LCDproc

6.1. Running LCDd

If you have installed the init-scripts you can simply start, stop and restart LCDd with the init-script.

6.1.1. Running LCDd from the command line

There are several reasons for running LCDd from the command line

- You don't want to install LCDd but run it from the source directory.
- You want to do some debugging.
- You want to get the output directly on stderr.
- ...

Note: If you run LCDd as a "normal" user, it will not change to the user specified in the config file. For parallel port devices you will need root privileges anyway.

The simplest command that will run LCDd is the following. It is useful for running LCDd from the source directory, e.g. after building.

```
$ server/LCDd -c LCDd.conf
```

6.1.2. The Command Line Options of LCDd

Running **LCDd -h** gives you an overview of the currently available command line options, including a list of the compiled in drivers.

Example 6-1. LCDd -h

```
LCDd Server Daemon (part of lcdproc), 0.4.3dev
Copyright (c) 1999 Scott Scriven, William Ferrell, and misc contributors
This program is freely redistributable under the terms of the GNU Public License
Usage: LCDd [ -hfiws ] [ -c <config> ] [ -d <driver> ] [ -a <addr> ] \
[ -p <port> ] [ -u <user> ] [ -w <time> ] [ -r <level> ]
```

Available options are:

```
-h Display this help screen
-c <config> Use a configuration file other than /etc/LCDd.conf
-d <driver> Add a driver to use (output only to first)
```

```

-f Run in the foreground
-i Disable showing of the main LCDproc server screen
-w <waittime> Time to pause at each screen (in seconds)
-a <addr> Network (IP) address to bind to
-p <port> Network port to listen for connections on
-u <user> User to run as
-s Output messages to syslog
-r <level> Report level (default=2)

```

Currently available drivers:

```

lcdm001, LCDM001, MtxOrb, MatrixOrbital, CFontz, CrystalFontz, LB216, text,
curses, ncurses, BayRAD, glk, glc

```

6.2. Running lcdproc

You will probably more often run **lcdproc** from the command line than you will run LCDd.

6.2.1. The Command Line Options of lcdproc

Running **lcdproc -h** gives you an overview of the currently available command line options.

Example 6-2. lcdproc -h

```

LCDproc, 0.4.3dev
Usage: lcdproc [-s server] [-p port] [modelist]
Options in []'s are optional.
modelist is "mode [mode mode ...]"
Mode letters:  [C]pu [G]raph [T]ime [M]emory [X]load [D]isk [B]attery
proc_[S]izes [O]ld_time big_cloc[K] [U]ptime CPU_SM[P]
[A]bout

Use "man lcdproc" for more info.
Example:
lcdproc -s my.lcdproc.server.com C M X -p 13666

```

Note: You will not be able to connect to a remote server, unless it listens to the correct interface and port! See `LCDd.conf`: The [server] Section for details on the server setup.

Chapter 7. Contact Us

7.1. Errata

Of course LCDproc is not perfect (yet). We do our very best to improve it, but in some cases we are very much restricted in our efforts.

The main reason for that is the fact that we do NOT have all the hardware people have written drivers for. Unfortunately some developers have kind of vanished and don't react to mails from the mailing list any more.

So, as far as drivers are concerned we rely on YOU as testers. We have developed elaborate "coding in the dark" skills over the time. E.g. the CFontz driver has been updated and ported to 0.4.3 without the developers having the hardware.

7.2. The LCDproc Mailing List

We are a bit lazy about the bug-tracking and whatever stuff on SourceForge. So, please contact us directly through the mailing list.

For details on how to subscribe to the list see <http://lcdproc.omnipotent.net/mail.php3>.

We like people to subscribe to the list with their real names. Of course we cannot and do not want to force you to do so. Anyways, we need to know your name, if you want to contribute code to LCDproc (legal issues of copyrights).

7.3. Reporting Bugs

Possibly you will find a bug in the LCDproc distribution. Before reporting this to the mailing list, please respect the following:

- Your system should be more or less up to date. This does not mean that you have to update from GNU/Linux kernel 2.0.x to 2.2.x or from 2.2.x to 2.4.x. But we would like to make sure that your problem is not related to a known bug in the kernel or maybe your compiler.
- Especially LCDd might need certain privileges to execute a command. Make sure LCDd HAS the rights to do so.
- When experiencing problems with LCDd, make sure that your hardware is OK. E.g. you should make sure that the wiring for your (in this case most likely parallel) device is correct.

- Make sure that you use the correct speed settings for your device. Incorrect speed settings (baud rate) are most likely to produce garbage scrolling on your display. Refer to the specifications of your device. If your device needs a speed setting that is not supported by LCDd send a message to the list.
- Make sure that you have modified the configuration file according to your needs and that LCDd actually uses the configuration file. I.e. you might have to run LCDd with the -c option

Anyways, no question is too stupid to ask ;) Feel free to ask whatever you want. Unfortunately replying to mails takes time as well (a damn lot of time). So, if you want LCDproc to develop faster, please try to solve a problem yourself first.

BUT if you have actually FOUND A BUG we will be quite happy if you let us know. We NEED YOU as testers and appreciate any feedback.

Appendix A. GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

A.1. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

A.2. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A.3. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

A.4. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all

these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

A.5. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

A.6. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

A.7. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

A.8. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on

covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

A.9. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

A.10. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

A.11. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.12. How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.