

4

MINIX ON THE COMMODORE AMIGA

This chapter tells you how to install and run MINIX on a Commodore Amiga. Four sections are present in this chapter. The first section discusses the kind of hardware you need to run MINIX. The second section gives an overview of how to get MINIX running. The third one goes into more details. The fourth one is about troubleshooting. If during the installation you have problems, please check the troubleshooting section. You may have run into a common problem whose solution is well known and described there. When you have finished reading this chapter and have successfully installed MINIX, please skip to Chap. 6 to learn about using your newly installed MINIX system.

4.1. MINIX HARDWARE REQUIREMENTS

MINIX should run on any Amiga 500 or Amiga 2000 that has at least 1M of memory. The most common peripherals are supported, except for the hard disk. Extra memory or a second drive makes programming much more pleasant. While it is possible to boot MINIX with a 512K 1 drive system, it is difficult to do anything serious, certainly not recompiling the operating system. To do that, you really should buy an additional 512K.

4.2. HOW TO START MINIX

Throughout the discussion below, lines printed in the Helvetica typeface are either commands you should type on the keyboard, or are lines that the computer will display for you.

Before running MINIX for the first time, make a backup of all the diskettes, to prevent disaster if one of them should be subsequently damaged. They are not copy protected. However, all of them, except the first one (called: "BOOT"), are not AmigaDOS disks, so do not use any of the usual AmigaDOS disk copy programs. Instead use either a copy program that is able to copy IBM-PC disks, or use the supplied *diskcopy* utility. Since MINIX does not come with a format program, use the *transfer* utility, which can be found in the C directory of the boot disk and can be invoked by typing:

```
BOOT:C/transfer -f
```

under AmigaDOS. If you do not have a program to copy IBM-PC disks under AmigaDOS you can not yet backup your original disks. Please remember to do so once you have got MINIX working.

To boot MINIX, proceed as follows.

1. Turn on the Amiga, and insert the boot diskette (BOOT) in any drive. If the Amiga was already powered on, you may also press both Amiga keys while holding down the control key to reset the Amiga.
2. Because the AmigaDOS diskette does not contain any of the usual utilities such as *setmap*, *rename*, etc., you have to copy them from your original Workbench disk onto the AmigaDOS disk yourself. When you boot MINIX for the very first time, a little program will show you exactly how to do so. When you have successfully copied the required utilities onto the AmigaDOS disk you should re-boot the Amiga. From now on the Amiga will automatically load MINIX whenever your boot from the BOOT disk.
3. About 15 seconds later, MINIX will ask you to either specify a root device or press return. Insert the root disk (ROOT) and press return. The RAM-disk will now be loaded into memory.
4. Another 10 seconds later MINIX will display a line telling how much memory the machine has, how large the operating system (including all its tables and buffers) is, how large the RAM disk is, and how much memory is available for user programs (the first number minus the next two). Check to see that the available memory is at least positive. MINIX will not run with negative memory. To do anything useful, however, at least 200K of available memory is needed.

5. When the RAM disk has been loaded, the system initialization file, */etc/rc*, is executed. It asks you to remove the root file system and then insert the */usr* file system ("USR") in drive 0 and type a carriage return. Do so.
6. After */usr* has been mounted, you will next be requested to enter the date (and time). Enter a 12-digit number in the form MMDDYYhhmmss, followed by a carriage return. For example, 3:35 p.m. on July 4, 1976 was 070476153500.
7. You will now get the message:

login:

on the screen. Type:

ast

and wait for the system to ask for your password. Then type:

Wachtwoord

being careful to type the first letter in upper case. Lower and upper case letters are always distinct in MINIX. Do not use an upper case letter when a lower case one is called for or vice versa. Like UNIX, MINIX regards "a" and "A" as two distinct characters. Please do not type "a" when you mean "A". It matters.

8. If you have successfully logged in, the shell will display a prompt (dollar sign) on the screen. Try typing:

```
ls -l /bin
```

to see what is in the */bin* directory on the root device. After that, try:

```
ls -l /usr/bin
```

to see what is on the drive 0 diskette. To stop the display from scrolling out of view, type CTRL-S; to restart it, type CTRL-Q. (Note that CTRL-S means depress the "control" key on the keyboard and then hit the S key while "control" is still depressed.)

9. If you have more than one diskette drive, you can mount one of the other diskettes by inserting it into drive 1 and typing:

```
/etc/mount /dev/dd1 /user
```

If you want to use drive 2 or 3, replace */dev/dd1* by */dev/dd2* or */dev/dd3* respectively. Use *ls* to inspect it. You can now try out other commands.

10. When you are finished and want to log out, type: CTRL-D. The

login:

message will appear, and you or another user can log in again.

11. When you want to shut the computer down, make sure all processes have finished, if need be, by killing them with *kill*. Then type:

sync

or just log out. When the disk light goes out, you can turn the computer power off. *Never, ever* turn the system off without first running *sync* or logging out (which does an implied *sync*). Failure to obey this rule might result in a garbled file system and lost data. If you forget and just turn off the computer, next time you boot, be sure to run *fsck* to repair the file system.

4.3. A MORE DETAILED LOOK

In this chapter we will describe some of the details of MINIX. Note: some programming examples will be presented in the rest of this chapter. You can recognize them by the prompts: The `1>` prompt indicates that you should type the command in an AmigaDOS CLI window, the `$` indicates a normal MINIX commando and the `#` indicates commands that should be run by the superuser (logged in as *root*). You *must not* type the prompts themselves, just type the commands following them.

The MINIX distribution consists of one disk in the normal 880K AmigaDOS format (which contains some tools and a binary of the operating system and is used for booting MINIX) plus a number of double-sided 720K MINIX disks. We will refer to these diskettes in the rest of this manual by their name in the first column of the following table. Here is the list of diskettes:

Name	Size	File system	Description
01 BOOT	880K	AmigaDOS	Used for booting MINIX
02 ROOT	720K	MINIX	160K Root file system copied to RAM disk
03 USR1	720K	MINIX	System Binaries 1 (/usr)
04 USR2	720K	MINIX	System Binaries 2
05 USR3	720K	MINIX	System Binaries 3
06 ACK	720K	MINIX	C compiler
07 SRC1	720K	MINIX	Operating System Sources
08 SRC2	720K	MINIX	Commands Sources 1
09 SRC3	720K	MINIX	Commands Sources 2

If you have not already made backups, now is the time to do so. You can use the normal AmigaDOS procedure to copy BOOT, as is described in the AmigaDOS

manuals, or you can use any of the available disk copiers. To copy the MINIX disks you will have to use MINIX itself. Be sure you have 8 formatted (see section 4.2) disks ready, to copy the original onto. Be sure to follow *diskcopy's* instructions and repeat this 7 more times. You can also use other means, e.g., *dos2dos's* format command or a real PC or Atari-ST to format the disks. We will refer to the copies as BOOT, ROOT, USR, ACK and SRC. Keep the original disks write protected under all circumstances to prevent accidental loss of the original source.

4.3.1. Keyboards

The Amiga comes with different keyboards in different countries. MINIX solves this in the normal Amiga-way: keymaps. This section describes how to set up your keyboard for MINIX.

If you have one of the European keyboards, you must first install a keymap for your particular version of the keyboard (unless you are willing to live with the US key bindings, meaning that the character engraved on the keytop will not always correctly describe which key it is). Life would have been a lot simpler if typewriter manufacturers had devised an international standard keyboard 100 years ago.

There are several methods for installing a keymap, increasing in complexity. If the one you use fails, please try one of the other methods. For all of the methods, we assume that you are a bit familiar with the CLI. If you are not, please read that part of the manual that came with your Amiga. Start up your Amiga and boot from your favorite Workbench disk. Now put BOOT: in a drive.

Using One of the Prefab Keymap Files.

Find out which one you normally use by typing:

```
1> type S:startup-sequence
```

You should see a line like:

```
setmap nl
```

which means you normally use the Dutch (nl) map:

```
1> cd BOOT:
1> dir devs/keymaps
```

You will see several files, all starting with *m_*, such as *m_usa0*. These are keymap files. You should specify your keymap by editing *BOOT:S/startup-sequence*. To specify the *nl* keymap, change *BOOT:c/setmap m_usa1* to *BOOT:c/setmap m_nl*.

If you cannot find your favorite map among the *m_** files, or it fails for some other reason proceed with step 2.

Converting Your Keymap.

Patch a keymap using one of the keymap editors available. We'll assume that you are using *KeyMapEd* because it is public domain and quite good. If you have another keymap editor it will probably do just fine.

The only changes necessary are: help, up, left, right, down to \x00, del to \x7f, f1 to \x1bOP, f2 to \x1bOQ, f3 to \x1bOR, f4 to \x1bOS, f5 to \x1bOT, f6 to \x1bOU, f7 to \x1bOV, f8 to \x1bOW, f9 to \x1bOX, f10 to \x1bOY.

You might change the (shifted) function key definitions: these are the only ones where it makes sense to select "string." If you do so you can map any of them to an arbitrary string. Do not to exceed an average length of 20 characters per key because in the kernel there are only 400 bytes to store their definitions. When redefining the keys, do not change the definitions of the (unshifted) function keys, since they are used by the *mined* editor.

Give up

If you think that this is all quite complicated or you are not so sure about really doing any of it, you can skip it for now and find out how the default map (*m_usal*) works for you. If worst comes to worst, experimentally determine what all the keys do, and paste paper stickers on the key tops giving their new functions.

4.3.2. The Preferences

When MINIX boots it copies all sorts of information from AmigaDOS such as the mouse pointer, which will be used as a cursor under MINIX, border and character colors, the keymap, the memory map, etc. To change the default settings you can boot from the BOOT disk, hit CTRL-D before MINIX has actually booted and then run *preferences* from your Workbench.

4.3.3. Exchanging Files between AmigaDOS and MINIX

Just as in the other versions of MINIX you can exchange files between AmigaDOS and MINIX. One problem, however, is that AmigaDOS uses a nonstandard diskette format; not just a different file system, but also a different encoding scheme for the data. To overcome this problem, we have provided a *transfer* utility to read, write and format MINIX diskettes under AmigaDOS. For more information on *transfer* consult the manual pages in Chap. 8.

The 720K diskettes used by MINIX on the Amiga conform to the industry standard for 3.5 inch diskettes, and can be read on the Atari ST using MINIX there. Thus you can make MINIX file systems on your Amiga and then use them on an Atari ST and vice versa. In fact, binary programs compiled on any of these systems can be run on any of the others without modification. This makes it easier for you to share

software with other MINIX users. People who do not believe in standardization are requested to read Sec. 4.3.1 again.

4.3.4. Making Backups of MINIX

Ok, how about *your first good deed as a MINIX user*? Boot MINIX login, and type:

```
# cd /  
# diskcopy
```

diskcopy asks you to insert the source disk, insert ROOT, the first of the disks, do so and hit return. After a while *diskcopy* will ask you for the (formatted) destination disk, insert one of the disks you've just formatted. repeat this process for the others and then store the original disks together with the original BOOT in a safe place. You will not need them again unless you accidentally damage one of the new copies. *Diskcopy* unmounts the */usr* disk so you'll have to remount it when it is done. First insert the the USR diskette, then type:

```
# /etc/mount /dev/dd0 /usr
```

You can check if MINIX is working 100% as follows. Type:

```
# cd /usr/test  
# run
```

These elaborate tests take over 15 minutes. If no error messages appear, the system is working properly. Be sure the diskette is write enabled.

You can now edit files, compile programs, or do many other things. The reference manuals given in Chap 8. and the extended ones in Chap 9. tell you about the programs available and what they do. The descriptions are for reference purposes, however. They are not tutorials. If you are unfamiliar with UNIX, it is suggested that you first read one of the many books available on this subject. Any good computer bookstore will have a wide selection of them.

4.3.5. Boot Procedure Options

While the default boot sequence will probably be just fine most of the time, you can change the behavior of the MINIX kernel with some useful options. The MINIX kernel, fs, mm and init, packed together in *BOOT:minix.img*, contain the MINIX code. This data file is read in by a little utility program called *minix*. This is an ordinary AmigaDOS program, it finds out some things about your Amiga (how much RAM you have, what keymap you use, if you have a NTSC 60Hz or PAL 50Hz machine etc.), then it loads *minix.img*, passes the information to it and starts it off. For the exact usage of *minix*, see the manual pages.

4.3.6. Even More Details

If you want to know more about the exact differences between the Amiga and the IBM versions of MINIX, you might also read the Atari specific chapter, since the Amiga version was derived from the Atari version, which was derived from the IBM version. The IBM version was not derived from anything. Details about exactly what devices are available, how the tty driver works, etc. can be found there.

4.4. TROUBLESHOOTING

Sometimes things can go wrong. If you are having trouble getting started, you should try to find a friend's machine and try MINIX there. If it works, then the problem is clearly due to incompatible hardware. To verify that this is indeed the problem, remove (or at least disconnect) all optional equipment from your Amiga and try again. If this works, insert the optional equipment one device at a time, rebooting MINIX after each one is installed until the guilty party is located.

If problems arise after you have gained enough experience to recompile the kernel, you might compile (parts of) the kernel with the `-DDEBUG` flag as to allow extra debugging output to appear when MINIX is running. This option is not very useful for inexperienced users, however.