MM/1

User

Guide

Information in this manual is subject to change without notice.

Contents

How To Use This Manual

Congratulations on your purchase of an MM/1 computer system! This manual will help you start tapping the power of your new computer. It provides enough information to let you start using your MM/1 quickly.

This manual covers how to safely operate your new computer. You will also get an overview of the software included with the MM/1, and how to find help should you have questions. This User Guide gives a brief introduction into using the more common features of OS-9/68000 and the other software that comes with the MM/1. More detailed information about using the software is contained in the program-specific manuals shipped with the MM/1.

The best way to learn about your system is to use it, and to consult the manuals. Try what you are comfortable with and then move on.

Typographic Conventions

The following typographic conventions are used in this manual:

`commands you type at the keyboard`

`text that is displayed on your screen`

[keyname] press the key with this name. [ESC] means press the escape key.

MM/1 System Software

The first step in setting up your system is to ensure that you received all the necessary components. In addition to any special parts of your order, you should find

- The MM/1 CPU case
- Power cable
- Floppy disks (6)
- This manual
- OS-9/68000 manuals
- OddJob manual
- Warranty card

Using your MM/1 safely

The MM/1 has been designed and configured for safe operation. Many of its components resist power surges, helping to avoid damage to its state-of-the-art chips. For additional protection of your MM/1 and peripherals you should use a power strip which provides surge protection.

Do not plug in the MM/1 until it is correctly set up as specified in this manual.

Set up your MM/1 in a dry location and on a stable piece of furniture. This is especially important when you are using a hard disk drive. Any vibrations from the table during a hard disk read or write may cause problems. This is not the fault of the MM/1, it is merely the way hard drives work.

Do not open the MM/1 case. Opening the case voids your warranty and may expose you to risk of shock. The MM/1 should be unplugged any time the case is opened. Anyone who opens the MM/1 case should remove any watches or jewelry that could come in contact with any of the components of the MM/1.

You can further protect the data you store on your MM/1 with an optional Uninterruptible Power Supply (UPS). A UPS can help save your data by providing power to your system even

if the power in your home fails.

Do not spill liquids on or in your system. If the MM/1 becomes wet, leave it to dry for an extended time (at least one day). Call IMS for further instructions.

Power

Find a safe place to set up your MM/1 away from weather or excessive moisture. The power outlet should be a grounded connector (three pronged outlet).

Before plugging in the MM/1, make sure your hands are dry. Also make sure it is turned off. The power switch is on the front of the case, to the right of the Power light. When the switch is in (depressed), the power is on. When the switch is out (raised), the power is off.

The power connector for the MM/1 is shaped to make it easy to plug in. The "female" connector on the cord goes into the empty power socket on the back of the MM/1. The "male" end plugs into the wall.

If you will be using several powered components with your MM/1 (sound speakers, monitor, printer, and so on), you may wish to place them on a single power strip.

Do not turn on your MM/1 yet.

Monitor

With the MM/1 off and with the back of the MM/1 towards you, look for the video connector (a nine-pin female DIN connector at the bottom and close to the center of the MM/1 backplate -- see the backplate layout on page B-1).

For monitors supplied by IMS, simply connect the monitor cable to the monitor connector on the back of the MM/1. Then plug the monitor power cord into the wall.

For other monitors, contact IMS for adapter cables. IMS currently offers cables for several popular RGB-Analog monitors.

Adjust the MM/1 and monitor for easy viewing. Now turn the monitor on.

Keyboard

The MM/1 accepts keyboards usable on International Business Machines (IBM) XT or compatible computers. Please select a keyboard with the "SysReq" (system request) key.

Contact IMS if you would like to acquire the appropriate keyboard.

MM/1 compatible keyboards have a five-pin DIN connector. On some MM/1 models you will plug the keyboard connector into the front of the case, using the case's empty five-pin DIN connector. On other models, plug the keyboard into the back connector (a five-pin DIN connector on the lower right-hand side of back of the case). The back connector may not be exposed, indicating that you should use the keyboard connector on the front of the MM/1 case.

Booting the MM/1 system software

Before proceeding, determine if your MM/1 is configured to operate with one megabyte of memory, or with more than one megabyte.

If your MM/1 is configured for one megabyte operation, place disk #1 in your MM/1's default (top) floppy disk drive.

If your MM/1 is configured for more than one megabyte operation and has that memory installed, place disk #2 in your MM/1's default (top) floppy disk drive.

Now turn on the MM/1 , monitor, printer, and other peripherals. The floppy disk drive access light will turn on, indicating that the MM/1 is attempting to boot from the floppy disk drive. If your MM/1 has been configured with more than one drive, and if the light appears on the empty drive, put the boot disk in that drive.

The MM/1 will now boot directly from floppy disk. If the MM/1 fails to boot, please check that you have the correct disk in the drive.

If the system boot is successful, you will see a "_$" prompt.

Setting the system time

> IF YOU HAVE AN MM/1 PERSONAL, PLEASE SET THE SYSTEM TIME EVERY TIME YOU BOOT. THE MM/1 MUST KNOW THE CORRECT DATE AND TIME TO HANDLE MANY ESSENTIAL TASKS.

Please type the following command at the system prompt:

```
_$ setime [ENTER]
```

Carefully follow the prompt to set the time in your Real Time Clock.

To ensure that the time is correct, now type

```
_$ date [ENTER]
```

If you have an MM/1 Extended, this should be the last occasion you set the clock with Setime. You may need to reset it if you change time zones or from Daylight Savings Time to standard time or vice versa.

If you have an MM/1 Personal, type the following line:

```
_$ merge startup setime >startup.p [ENTER]
_$ copy -r startup.p startup [ENTER]
```

Personalizing your system

Interactive Media Systems, Inc. strongly suggests that you use the MM/1 as a multiuser computer system, even though for now you are a single user on your system.

By treating the MM/1 as a multiuser system, you can gain extra value from your MM/1. You can allow other people to log on and access specific programs and data. You can prevent users from accessing sensitive programs and data. In addition, Interactive Media Systems, Inc. plans a variety of additional products that will make extensive use of the multiuser capabilities of the MM/1.

Now type:

```
_$ mylogin [ENTER]
```

and follow the instructions.

Even if you are an experienced OS-9 user, please use the Mylogin instructions.

Trying out the system

To give you a sample of some of the capabilities of the MM/1, you may wish to try the following demonstrations.

For best results, acquire a cable for the MM/1's sound port. You will need a 5-pin DIN to dual-stereo jack cable. These are available from your local electronics store or from IMS.

Attach the cable to the sound port (see "MM/1 Backplate Layout" on page B-1). Now attach the stereo jacks to the input jacks of a stereo system. You can use an amplifier with separate speakers, or the combined powered speakers commonly used for portable personal stereo systems. These can also be acquired from your local electronics store.

If you have an MM/1 configured for one megabyte operation, now type

       `_$ demo1 [ENTER]`

If you have an MM/1 configured for more than one megabyte operation, and the memory is installed, type

       `_$ demo2 [ENTER]`

Enjoy!

Installing MM/1 Software on a Hard Drive

The MM/1 software provides a wealth of power, utility, and enjoyment. To take maximum advantage of this software, a hard disk drive is recommended. Hard disk drives access data much faster, and can store many times more information than floppy disks.

The MM/1 supports the SCSI (Small Computer Systems Interface) standard. This permits the use of up to seven SCSI devices on the MM/1. SCSI devices include hard disk drives, tape drives, CD-ROM drives, and others.

An MM/1 SCSI hard disk drive is easy to install. See the Appendix to this user guide called "Installing Hard Disk Drives on your MM/1" (beginning on page A-1). Interactive Media Systems, Inc. can also install the hard disk drive for you, preserving your warranty.

The MM/1 and Small Computer Systems Interface (SCSI)

Once you have installed a hard disk drive, you must format it before it can be used. Before you can format the SCSI drive, make sure the hard drive descriptor and driver are in memory. These are on Distribution Disk #6 in the /BOOTMODS/RBF directory. (See the ReadMe.First file in /BOOTMODS/RBF for an explanation of the modules.) To load the modules type:

```
chd /d0/bootmods/rbf
load -d h0
load -d rb33c93
```

(If these files are not found in the subdirectory, consult the ReadMe.SCSI file for the file names.)

The default device descriptor for hard disks is set to prevent accidentally formatting the hard disk. When you type :

```
dmode /h0
```

you will then see:

```
drv=00 stp=00 typ=80 dns=00 cyl=026C sid=04 vfy=01
sct=0011 tos=0011 ilv=03 sas=11 tfm=00 toffs=00
soffs=00 ssize=0200 cntl=0003 trys=07 lun=00
wpc=0000 rwr=0000 park=0000 lsnoffs=00000000
```

The cntl parameter must be set to 002 to enable formatting. To do this, type:

```
dmode /h0 cntl=002
```

Then you may proceed with formatting the hard disk as described in this manual. DO NOT save the changed hard drive descriptor. When you reboot the MM/1 the descriptor will default to cntl=003, preventing accidental hard disk formats.

Formatting SCSI hard drives is different with the MM/1 than with what you may be accustomed to with a Color Computer hard drive system or an IBM clone system. The biggest difference between a SCSI hard drive and other popular interfaces such as ST406 is that the drive itself is "intelligent". This means that you do not need to know all the many parameters of the SCSI hard disk drive to format it. All you need to know is the total number of sectors on the drive.

> IT IS IMPORTANT TO ALLOW YOUR HARD DISK TO WARM UP COMPLETELY BEFORE FORMATTING. BEFORE FORMATTING THE HARD DRIVE, LEAVE YOUR MM/1 SYSTEM TURNED ON FOR AT LEAST FIVE MINUTES.

Sectors, cylinders, heads, and dmode

For example, the following drive types and their respective number of sectors were taken from a Seagate handbook:

Drive Type	Sectors	Size
ST138N	63,139	32 MB
ST157N	95,015	48 MB
ST225N	41,720	21 MB
ST251N	84,254	43 MB
ST277N	126,790	64 MB

So, to correctly format a SCSI drive, all you need to tell the format utility is the number of

sectors on the drive. If you only know the number of sectors in decimal, you will need to convert this number to hexadecimal before proceeding.

The MM/1 format utility uses cylinders and heads as drive size parameters. How do you use these parameters to specify the number of sectors for the drive you have chosen?

Call the format utility with the following command line:

```
_$ format -nvnp -v=Test /h0 [ENTER]
```

The format utility will do a logical format of the hard drive creating the sector allocation table and write the information in logical sector zero (LSN0). It will return, among other things, the total number of sectors created on the drive. This is the number you are interested in.

If the number of sectors returned is different than the total number of sectors on your hard disk, then all you need to do is use the dmode utility. The dmode utility can change any parameter about the drive size (such as number of cylinders, number of heads, number of sectors per track). This way you can increase or decrease the number of sectors in the hard drive descriptor to make it equal to the number of actual sectors.

An example

For example, suppose you have an ST225N drive. If you run the format utility as shown above, it will return 41,680 sectors. You would not need to change anything since the parameters are set correctly. However, if you have an ST251N drive, you would need to DOUBLE the number of sectors that the format utility will create. Therefore, you could take the number of cylinders OR the number of heads and double that value. Using the dmode utility, you would try the following command line:

```
_$ dmode /h0 [ENTER]
```

you will then see:

```
drv=00 stp=00 typ=80 dns=00 cyl=026C sid=04 vfy=01
sct=0011 tos=0011 ilv=03 sas=11 tfm=00 toffs=00
soffs=00 ssize=0200 cntl=0003 trys=07  lun=00
wpc=0000 rwr=0000 park=0000 lsnoffs=00000000
```

Notice that "sid=04". That means that OS-9 thinks your drive has four sides (heads). You will want to double the number of sides (heads) to double the number of sectors. In this example, you would type:

```
_$ dmode /h0 sid=8 [ENTER]
```

Now if you type

```
_$ dmode /h0 [ENTER]
```

you would see:

```
drv=00 stp=00 typ=80 dns=00 cyl=026C sid=08 vfy=01
sct=0011 tos=0011 ilv=03 sas=11 tfm=00 toffs=00
soffs=00 ssize=0200 cntl=0003 trys=07  lun=00
wpc=0000 rwr=0000 park=0000 lsnoffs=00000000
```

Alternatively, you can double the number of cylinders:

```
_$ dmode /h0 cyl=04d8 [ENTER]
```

If you checked the current status of the /h0 descriptor with

```
_$ dmode /h0 [ENTER]
```

you would see

```
drv=00 stp=00 typ=80 dns=00 cyl=04D8 sid=04 vfy=01
sct=0011 tos=0011 ilv=03 sas=11 tfm=00 toffs=00
soffs=00 ssize=0200 cntl=0003 trys=07  lun=00
wpc=0000 rwr=0000 park=0000 lsnoffs=00000000
```

Since the dmode utility requires hexadecimal input, you will need to understand hexadecimal notation to correctly change the number of sectors. But since decimal numbers and hexadecimal numbers are the same up to the numeral 9, you can more easily do the arithmetic to double the number of sides (heads).

Once you set the number of cylinders to the total number of sectors actually on the hard drive, you can then format the drive without any command line options. This will then perform a physical format, ask you for a volume name, and then optionally perform a verify (which you should always do with a new drive).

Finally, you DO NOT need to save the changed hard drive descriptor to be able to access that drive in the future. As was mentioned before, SCSI hard drives are "intelligent" drives. The drive will keep track of the drive description you just established.

Booting from your hard drive

Now that your hard disk drive is correctly installed and formatted, you can make your MM/1

system bootable from your hard disk.

Boot up your MM/1. Type

```
_$ load format [ENTER]
_$ iniz r0 [ENTER]
_$ copy /d0/startup  /r0/startup [ENTER]
_$ chd /d0/sys [ENTER]
_$ makdir /r0/SYS [ENTER]
_$ dsave -e /r0/sys [ENTER]
```

Remove the boot disk from drive /d0. Now place an unformatted high-density floppy disk in drive /d0. Type

```
_$ format /d0 [ENTER]
```

and follow the prompts.

When the floppy disk has been completely formatted, put "MM/1 Disk Distribution Disk #6 -- Bootmods" in floppy drive /d0. Type

```
_$ chd /d0/bootmods [ENTER]
_$ chx /d0/cmds [ENTER]
```

Now, type

```
_$ dir [ENTER]
```

You should see a file named

```
makehdboot
```

This is a procedure file that will create an OS-9 boot file suitable for completing a boot from hard drive.

To execute it, type

```
_$ makehdboot [ENTER]
```

At the prompt, replace MM/1 Distribution Disk #6 with your newly formatted high density disk.

With the newly formatted disk in Drive /d0, now type:

```
_$ chd /r0 [ENTER]
_$ dsave -eb=64k /d0 [ENTER]
```

When Dsave has completed its work, at the system prompt, type

```
_$ os9gen /d0  -eq=/d0/os9boot [ENTER]
```

Now, press the reset button on the front of the MM/1. The boot should finish on the hard drive.

Transferring your distribution software to the hard drive

Once the MM/1 has successfully booted from the hard drive, label the disk in Drive /d0 "Hard Drive Boot" and put the current date on it.

Now put Distribution Disk #1 in Drive /d0. Type

```
_$ chd /d0; chx /d0/cmds [ENTER]
_$ move2h0 [ENTER]
```

When prompted, please remove Distribution Disk #1 from Drive /d0 and insert Distribution Disk #2 into the drive. Once again, type:

```
_$ chd /d0; chx /d0/cmds [ENTER]
_$ move2h0 [ENTER]
```

Repeat this process with the remaining Distribution Disks.

You're done!

Common OS-9/68000 Commands

The MM/1 system software is designed to run the OS-9/68000™ operating system.

The MM/1's operating system handles what you ask the MM/1 to do. Properly configured, it interprets your keyboard or mouse instructions. It also knows how to use your disk drives, monitors, terminals, memory, sound chips, and more.

The operating system thus handles a host of enormously complex and detailed tasks. However, the power of OS-9/68000 and MM/1 software is easy to use, due in large part to the Shell.

The OS-9/68000 Shell is software that lets you access and control much of the capabilities of the MM/1. Every time you see the "system prompt" (_$), you are being asked by the Shell to type in a command. When you press [ENTER], the Shell tries to interpret what you typed.

In this section, we will introduce you to some of the most widely used commands. To fully access the power of the MM/1 you should carefully study the complete documentation provided with your system, which provides detailed descriptions of OS-9/68000 and MM/1 commands, utilities, and programs.

Chd and Chx

OS-9/68000, like most modern operating systems, stores filenames on a disk in groups. Each group has a name. In addition to containing the names of files, each group can contain names of other groups. Groups of filenames are called directories.

The whole collection of software on a disk constitutes a group called the *root directory*. It is wise to have several directories in the root directory. These *subdirectories* provide a method of organizing your files in general groups, such as SOURCE, TEXT, PICS, and so on.

Often the directory structure is described as an inverted tree, with the root at the top, and subdirectories below:

```
                            \   (root)
           _____
          /                |                        \
       SOURCE            TEXT                      PICS
     _____|_              |                         |
    /       |             |                         |
  foo.c    bar.c        LETTERS                  demo.pic
```
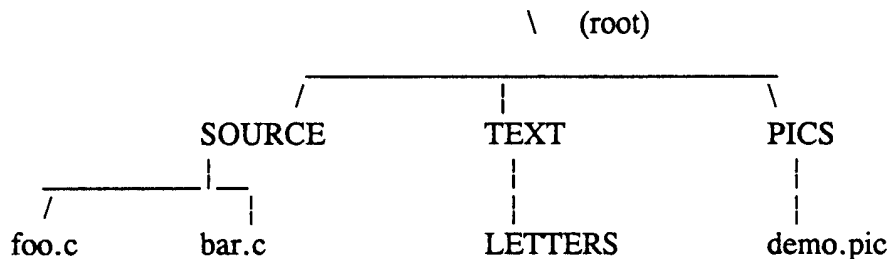
In this example, the root directory contains three subdirectories: SOURCE, TEXT, and PICS. The SOURCE subdirectory contains two files: foo.c and bar.c. The TEXT subdirectory contains one subdirectory: TEXT/LETTERS. The PICS subdirectory is empty.

There are two major kinds of files you will deal with on the MM/1. Most of the time, directories contain files that are filled with data, word processing files or telephone numbers. The other common kind of file contains not data but a *program*.

When these programs are small and do specific actions, they are called *commands* or *utilities*. More fully-featured, general purpose programs are often called *applications*. All of these kinds of files are instructions that the MM/1 can read and directly execute.

By tradition, this second type of file is called an *executable*, and are most often grouped together in their own directories, called execution directories. Execution directories generally do not contain data files (although there are some exceptions).

The Shell's functions include commands -- built into the Shell -- that help you select which directory of files you want to deal with.

To move from directory to directory, use the chd command. When you use this command, followed by the name of a directory of files, the MM/1 can keep track of which files and directories especially interest you.

To instruct the operating system which directory to search for executables, use the chx command.

Example 1

If the data files you're interested in are grouped into a directory named TEXT, tell the Shell that it should look there first by typing

```
_$ chd text [ENTER]
```

If the executables you need (for example, the MicroEmacs text editor) are in a directory called /h0/USR/CUST/CMDS, you may type

        _$ chx /h0/usr/cust/cmds [ENTER]

Example 2

If the files you want to deal with are in the root directory of /h0, you would inform OS-9 of this by typing:

        _$ chd /h0 [ENTER]

Now, OS-9 will look in this directory to find your files (unless you carefully specify otherwise by giving a file's full name).

If the commands, utilities, and programs you want to use are mostly or completely in /h0/CMDS, then let the MM/1 know this by typing:

        _$ chx /h0/cmds [ENTER]

dir

Suppose you type

        _$ chd /h0 [ENTER]

Now the MM/1 knows that /h0 is where it should look for your current files. You can list the files in this directory by using the dir command. This command is invoked by typing

        _$ dir [ENTER]

The dir command provides the following kind of output:

```
                    Directory of . 01:57:31
    AR          BASIC     BOOTMODS  C          CMDS
    COM         DEFS      DEFS.distr           DEMO.CMDS
    DEMOS       DOC       DRIVE     FIN        GAMES
    LIB         MAPS      NET       PCREF      PICS
    SCRIPTS     SNDS      SRN       SYS        TMP
    TODO        TOOLS     TXT       USR        VME.PCF
    adm         dem       osk.login printing.prices
    readme.dem            sample.login         startup
    tony.number           ved.new
```

By OS-9/68000 convention, those names all in capital letters represent directories. These directories contain the names of other files and perhaps of other directories as well.

Notice that one of the directories in the above listing is called SYS. Suppose you now type

```
_$ chd sys [ENTER]
```

followed by

```
_$ dir [ENTER]
```

You might see

```
                    Directory of . 02:02:34
    emacs.hlp  emacs.hlp.note      errmsg      errmsg.short
    moded.fields       motd        password    password.old
    srcdbg.hlp         stdfont     stdfonts    termcap
    termcap.old        umacs.hlp   ved_env.file
    ved_help.file
```

Copy

The copy command will make a duplicate of any file. All you have to tell the copy command is the name of the file to copy and where the duplicate should reside. If the file to copy appears in your current directory (that is, it appears in the directory listing when you type dir), then you can enter the file name as it appears in the listing. If you want to copy the file to another filename in the same directory, then just enter the new filename after the file's name. For example,

```
_$ copy original.file copy.of.file [ENTER]
```

will make a duplicate of *original.file* in the same directory, but under the new name *copy.of.file*. (Remember that two files cannot have the same name in the directory listing.)

If you want to copy any given file to any other file name, you can give the copy command a complete name for the source file and a complete name for the new file. For example, you can type

```
_$ copy /dd/sys/termcap /d0/txt/foo [ENTER]
```

This will copy the file named termcap (listed within the SYS directory on the /dd device), to a new file named foo. The file foo will new be listed in the TXT directory on the /d0 device.

Notice that device names all begin with a forward slash (/d0, /dd, /h0, /r0, /p, and so on).

Del

In addition to being able to create new files with commands like copy or applications like Emacs, you can delete them from directory listings. Once deleted, it is difficult or impossible to retrieve.

To delete a file named foo, type

```
_$ del foo [ENTER]
```

and to delete a file named /dd/usr/steve/cmds/sm.doc, type

```
_$ del /dd/usr/steve/cmds/sm.doc [ENTER]
```

In the previous example, you could also have redirected the MM/1's attention to the right directory with chd, and then given the del command the name of the file as it appears in that directory:

```
_$ chd /dd/usr/steve/cmds [ENTER]
_$ del smersh [ENTER]
```

List

With OS-9/68000, everything is treated as a file. Some files have special attributes and features that keep you from accessing all of them the same way. A directory is a file that lists other files. Use dir to list a directory's contents.

A file that is not a directory can be either a text file or a "binary" file. You can read a text file. If you try to read a binary file, your screen can behave strangely as these kinds of files contain codes that can confuse the OS-9 software managing your screen. (Listing a binary cannot damage your system.)

Text files are files such as correspondence, documentation, reference data, phone numbers -- anything you can handle with a text editor, for example.

You often want to see what a text file contains. OS-9/68000 permits you to read most text files with the simple command line:

```
_$ list filename [ENTER]
```

The filename can be the name of the file as it appears in your current directory listing, or it can be a complete filename that includes the device name and the names of the directories that contain the file in question.

For example, one can type

          _$ list /dd/txt/press/activities.out [ENTER]

or one can type

          _$ chd /dd/txt/press [ENTER]
          _$ list activities.out [ENTER]

In either case, your screen will now show:

     Recent IMS activities aid promotion of MM/1 computer

     Interactive Media Systems, Inc. would like to inform
     customers of this forum concerning its vigorous
     activities in the last two months. It is our firm belief
     that the time we spent has strongly advanced the MM/1 as
     a mainstream product.
                              ...

and so on.

Mfree

The mfree command will tell you how much memory is available for use by your programs and by MM/1 windows.

Under OS-9/68000, memory is carefully managed to provide speedy access to utilities while conserving memory resources. Normally when you use a command, utility, or program, OS-9 loads it from a disk drive into memory. Once in memory, OS-9 places it in a memory directory and gives the program a link count of one. If another user or process needs the program while it is in memory, OS-9 will increase its link count again. While the program has a non-zero link count, it will stay in memory. This makes your system responsive and conserves memory since all processes needing your program can find it right away in memory, and they all share a single copy.

When all processes are finished using the program in memory, one of two things can happen. If the program in memory has a link count of zero, it is normally removed from the memory directory and the memory it used is released to the operating system. But if the program in

memory was written by the programmer to be a "sticky" program, it will stay in memory even if its link count is zero. As long as memory is available for other programs, the sticky module will stay in memory.

mdir, procs, *and* kill

To keep track of which programs are in memory, you can look at the listing of the MM/1's memory directory. Simply type:

    _$ mdir [ENTER]

The MM/1 quickly presents you with a list such as:

```
        Module Directory at 21:31:01

    kernel    init      sysgo     tk68901   rtclock
    scf       null      nil       ms68901   windio
    stdfonts  ms        term      w         w1
    w2        w3        w4        w5        w6
    w7        w8        w9        rbf       rb37c65
    rb33c93   d0        d1        dd        h0
    math      cio       shell     setime    date
    mdir      mfree     procs     irqs      devs
    format    dir       copy      iniz      free
    load      syscall   break     WData     pipeman
    pipe      ram       r0        sc68070   sc68901
    t0        t1        p         sc68230   cls
    unlink    color     more      display   runb
    me        grep      sh        xmode     ls
    del       proff     tmode     pd        list
```

Not all modules in memory will be currently in use. You may wish to know which modules are involved in active processes. To find this information, use the procs command.

    _$ procs [ENTER]

The procs command tells you a variety of data about each active program (process) in your MM/1. Three important pieces of information are each program's "Id", "Prior" (priority), and "Module & I/O" data.

Id	PId	Grp.Usr	Prior	MemSiz	Sig	S	CPU Time	Age	Module & I/O
2	0	0.0	128	0.50k	0	w	0.22	10:07	sysgo <>>>term
3	2	0.0	128	3.75k	0	w	48.56	10:07	shell <>>>term
4	0	0.0	128	4.00k	0	w	0.06	10:07	shell <>>>w1
6	3	0.0	128	19.50k	0	w	2.85	8:11	sh <>>>term
7	5	0.0	128	19.50k	0	w	1:36.61	8:10	sh <>>>w1
9	10	0.0	128	19.75k	0	s	20.09	8:10	sh <>>>w2
10	0	0.0	128	4.00k	0	w	0.19	8:10	shell <>>>w2
12	6	0.0	128	59.25k	0	s	1:30.68	3:25	me <>>>term
13	7	0.0	128	15.50k	0	*	0.34	0:00	procs <>>w1>dd

In case any program you run needs to be stopped, you can move to another window, find the program's name under "Module & I/O" and its process id under "Id". With this knowledge, you can then Kill the program. If the program that needs to be stopped has a process id of 12, you would type:

```
_$ kill 12 [ENTER]
```

Generally this command is sufficient to stop the program that you wish to stop.

YOU SHOULD NEVER KILL A PROCESS IN USE BY OTHER USERS ON YOUR MM/1 SYSTEM UNLESS YOU KNOW EXACTLY WHAT YOU ARE DOING.

MM/1 Software Utilities

Introduction

The following sections provide an overview of some important MM/1 software. Additional documentation can be found on the distribution disks provided with the MM/1. Documentation on the disks can be found in files named ReadMe in many of the disks' directories. You can also find detailed information on Distribution Disk #4 in the directory called DOC. You can print this documentation with a continuous-feed printer by using the chd command to move to the right directory on your MM/1 disk. Then use the command

```
_$ proff filename.out   [ENTER]
```

on all files whose extensions are ".out". Make sure Proff is either in your current execution directory or in memory when you execute the command.

The MM/1 follows a long tradition of powerful computer systems, drawing its heritage from OS-9 and UNIX. These operating systems are known for the wide variety of commands offered to the user.

Some of the MM/1 tools include simple commands and enjoyable applications such as Paint. Others are flexible tools for power users. Others are specifically for programmers. As you learn more and more about your MM/1 system, the more you will polish your ability to choose the right tools. The power of the MM/1 will grow with you.

Paint

On the distribution boot disk #1 you will find a program called paint. To run paint, you must have a variety of font files in your current disk's SYS directory.

Add the following lines to your startup file.

```
merge sys/stdfont_01.fnt
merge sys/stdfont_02.fnt
merge sys/stdfont_03.fnt
```

```
merge sys/stdfont_04.fnt
merge sys/stdpats_16
merge sys/stdpats_256
```

These files are included in the SYS directory on Distribution Disk #2.

Make sure your Logitech™-compatible mouse is attached to /t2, the powered serial port on the MM/1 Extended board (see page B-4). Make sure that you have the following files in memory:

```
ms
ms901
```

Make sure you do NOT have the following file in memory:

```
t2
```

You can find out whether these files are in memory by using the mdir command. mdir gives you a directory listing of all modules in memory. Type

```
_$ mdir ms [ENTER]
```

If Ms is in memory, the mdir command will return its name. If ms is not in memory, mdir will return a blank line. Check for ms901 using the same method.

If the t2 descriptor is in memory, it probably was put there one of two ways. First, it may have been loaded into memory by your startup file. If it was, remove the line that loads it from the startup file. Second, t2 may be a part of your os9boot file. These are created by you with the os9gen command and a list of file names (called a bootlist). If you generated your own boot with os9gen and a bootlist, use MicroEmacs to remove the t2 name from the bootlist and regenerate your boot.

Reboot to ensure that your system now has the proper modules in memory. Now type

```
_$ loadbuffs [ENTER]
_$ paint [ENTER]
```

In a moment, the paint program will we awaiting your mouse-driven selections.

Many popular and powerful drawing commands are available in paint. Most every drawing tool can be used with colors and patterns can create vivid and subtle graphics.

All features will use the mouse as the primary input device. With the left button, you choose a function or use it. With the right button, you can change window screens.

When you hold down the left button while moving the cursor, you are "dragging" the cursor. You should drag the cursor down a menu list. You can drag the cursor when drawing a line, or when determining the size of a circle or bar.

Examining the top menu bar, you see three symbols or icons, followed by the words Files, Edit, Tools, Color, and Pattern.

The first icon is not used. The second icon, if clicked upon with the left mouse button, will start closing down paint by prompting you if you wish to quit. The third icon, shaped like a coffee mug, provides menu selections for common MM/1 system control functions. Some of these functions are implemented in the MM/1 version of paint; others will be implemented in the commercial version of paint to be available from HyperTech Software.

Selecting Shell from this menu will fork a Shell to your current paint screen. Use this Shell to list files or inspect directories, for example. You can also select Cal to see the current month. Use Calc to fork a calculator to a separate screen. Use your windowing hot-keys to select the calculator.

The first menu, Files, lets you create a New picture, lets you Abandon an old picture, and lets you Quit the program. Other options will be implemented in the commercial version of paint.

The second menu, Edit, will let you Undo the last action you took. You cannot Undo an undone action. Other options will be implemented in the commercial version of paint.

The third menu lists the available drawing and painting tools. These include Lines, Bars, Boxes, Circle, Pencil, Rays, Fill, and Text. Text will be implemented in the commercial version. All the other tools are available. Lines will create straight line segments when you drag your mouse cursor across the screen. Bars will create rectangles that are filled with the pattern chosen under the Pattern menu, and in the color chosen by the Color menu. Boxes create rectangles that are not filled, but whose outline is the color and pattern chosen in the Color and Pattern menus.

Circle creates unfilled circles whose outline is in the color and pattern that you select. Pencil lets you draw what you will in freehand fashion. Move the cursor slowly for accuracy and smooth action. If you move the cursor quickly, the lines you draw can be "step functions" -- attractively jagged. Experiment for pleasing effects.

Rays will create, in the color and pattern of your choice, a whole series of line segments whose starting point sits where your cursor is first clicked, and whose length and density are determined by the location of the cursor as you drag it, and how fast you drag it.

Fill will take the current background color and pattern and fill in the background until it reaches borders you create with other drawn entities. If the borders are completely drawn ("air-tight"),

the fill process will not pierce the border.

The Text option is not currently implemented.

The next menu lists some colors that are available. Other colors will be selectable in the commercial version of paint. If you need more colors, you can choose a pattern that suggests the color you like. If you want a very light blue, you can select the Light Blue color and a Small Dots pattern to give a lighter effect. Select a pattern and start drawing. The pixels set by your cursor will only be those that the chosen pattern allows. The last selection on the Colors menu is COlorSet. This allows you to adjust the color assigned to one of the sixteen color "slots" presented in the overlay window.

Using Alternative Fonts

The MM/1 supports up to 256 fonts in memory. Each font can contain 256 unique characters.

To use alternative fonts, you can load a new font into memory and then select it using the MM/1 windowing system.

To load a new font, you must first have a font file on your disk. These can be found on common information services such as CompuServe™ or Delphi™. You can also contact Interactive Media Systems, Inc. for information on alternative fonts for the MM/1.

Once a new font has been acquired, load it into memory using the Merge utility. You must then select the font group and buffer number assigned to the font using the Display command. Now your new font will appear on the screen.

To load the font, type

```
_$ merge fontfile.fnt
```

To find the font group and buffer number, use the Dump command to examine the font's header. Make sure the font is in your current directory. Now type:

```
_$ dump fontfile.fnt
```

You will see output such as

```
Addr    0 1  2 3  4 5  6 7  8 9  A B  C D  E F 0 2 4 6 8 A C E
----    ---- ---- ---- ---- ---- ---- ---- ---- ----------------
0000    1b2b c823 0000 0800 0808 0000 0000 0000 .................
0010    0000 0000 0000 0000 0000 0000 0000 0000 .................
            ***    15. duplicate lines ***
0110    0000 0010 1010 1000 0010 0024 2424 0000 ...........$$$..
```

This output starts with eight characters "1b2b c823". The second group of four characters is what interests us.

Use these four characters with the Display command as follows:

```
_$ display 1b3a c823
```

The Display command sends instructions directly to your window that govern the font it uses. "1b3a" tells the window that it must now use a new font, whose group and buffer codes follow. "c823" are the group and buffer number for the font.

Since the font has been Merged into your MM/1's memory, the windowing system can find the font you specified. Immediately after the Display command is executed, your new font is available.

Flicker

Flicker is a utility that provides MM/1 owners access to animations created by AutoDesk Animator™ or AutoDesk 3D Studio™. On MM/1 Distribution disk #4 is a directory called PICS/FLICKS that contains a sample animation. To run it, place the disk in your floppy drive, and type

```
_$ chd /d0/pics/flicks  [ENTER]
_$ chx /d0/cmds  [ENTER]
_$ flicker sample  [ENTER]
```

Playm/Plays

Also on Distribution disk #4 is a directory called SNDS. With that disk in Drive /d0, type

```
_$ chd /d0/snds  [ENTER]
_$ chx /d0/cmds  [ENTER]
_$ playm sample1.m  [ENTER]
```

Type [CTRL][E] simultaneously to exit Playm.

Playm replays a digitized sound file monaurally, while Plays replays a stereo sound file. Other

sample files are mono when they end in ".m" and stereo when they end on ".s". As you acquire more sound files, they will be labeled as mono or stereo. Now you have the tool to play them back!

Emacs

In addition to the Microware-distributed Umacs, its "big brother", Emacs, is included with your MM/1.

Emacs, like Umacs, is a text editor. A text editor allows you to move your edit cursor around the screen and edit a file through deletion, insertion, and formatting. Emacs also supports additional functions such as multiple buffers (usable for "cut-and-paste"), forward-searching for a string of characters, and keyboard personalization.

To round out your word-processing, the Proff text formatter is included. Proff and Emacs together make a complete system for creating letters, documentation, reports and dissertations, and so on.

Emacs, like much good OS-9 software, configures itself for the type of display you are using. To do this, it must know exactly what your display and keyboard are capable of doing; that is, it must know your "terminal capabilities".

IMS has provided boot disks that contain a file called SYS/termcap. This file must ALWAYS exist on your default drive (/dd). This file lists codes for a variety of terminals which summarize the cursor control and screen-manipulation capabilities of these devices.

Your MM/1's display and keyboard capabilities depend on the Signetics 66470 Video System Controller (VSC) chip. There is an entry in /dd/SYS/termcap for the VSC. To run Emacs, you must first acknowledge to your MM/1 that the VSC will be your terminal type. To do this, type

```
_$ setenv TERM vsc   [ENTER]
```

This command line designates the current window as a VSC terminal type.

Now, when you type

```
_$ me filename   [ENTER]
```

Emacs will start up on your screen. If you do not set your MM/1's window "environment" to any terminal setting, Emacs will return an error message.

Quick start with Emacs

Emacs is powerful. Some people in the Unix community will use nothing else to prepare documents or to program.

For most editing jobs, you really only need to know a few Emacs commands. Once you master these, you can carefully examine the Emacs documentation to try out more advanced features.

To start Emacs

Make sure that *me* is in your current execution directory. To do this, type

        _$ dir -x me   [ENTER]

The system should respond with "me" on a line by itself. That means OS-9 found the execution directory and that it also found *me*.

If *me* is not present, find the disk that contains it, set your current execution and data directories, and start again.

To use Emacs

Your screen will clear leaving only two lines at the bottom for "status" and "error" messages. The second-to-last line shows the program name and revision, the format mode in parentheses, and the buffer and file names.

The parentheses contain no word or space, indicating that your format mode is set to nothing. Modes available include Wrap, View, and Cmode.

Start typing. Type any text at all. Notice that your text continues off the screen to the right, and some of the left-most text disappears. The text is not gone; to see it, press the "Home" key on your keyboard. To back to the end of your line of text, press the "End" key.

Most often you want to have text wrap around as you type to the end of the screen. To do this, set the format mode to wrap. Type

        [CTRL][X] [M]

and at the prompt, type the word "wrap" followed by [ENTER].

Now the status bar has the word "WRAP" in parentheses.

Notice that the long line you typed still runs off the end of the screen (that's what the "_$" means). For now, just start typing again. Notice the text now wraps around when you approach the end of the line.

Try out the arrow keys. On the MM/1, these work fine (not all terminals that you may use remotely will support arrow keys as a default, but you won't need to worry about that for now).

If you try to move the cursor previous to the first character of the file, you won't be able to do it. If you try to move the cursor past the bottom of the file, you will also be frustrated. You may be able to move the cursor freely over what appears to be a blank screen and then suddenly find it cannot move farther along in the file. Actually, if you can move the cursor at all, you are moving it over a "character" in the file -- although it may be invisible. These include "spacebar" characters and "return" characters. They don't appear on the screen, but you can cursor over them.

Try deleting some text. Move your cursor with the arrow keys over some text. Press the "Del" key on your MM/1 keyboard. (You can also use [CTRL][D] if that is more convenient for your hands.)

Inserting text is easier -- you don't even need the MM/1 keyboard's "Ins" key! Just move the cursor over the text until you wish to start typing, and then type!

Now, go back to the line that you first typed that went off the screen to the right. Place the cursor at the beginning of the line with the "Home" key on your MM/1 keyboard. Press

```
[ESC][O]     (Escape and the letter "O")
```

The line should now wrap around.

Now go to the bottom of your text with the arrow keys. Type some characters. Now use the MM/1 keyboard's "BackSpace" key (NOT the left arrow key). This key will move the cursor back and erase the character over which it moves. Now try the left arrow key. Notice that it does not destroy the previous character.

Getting Help
Although Emacs has a built-in help facility, the easiest way to access the help file is to open another MM/1 window, then open up Emacs in the window with the help file as its file!

First, at a system prompt, type

```
_$ shell <>>>/wn&   [ENTER]
```

replacing the number "n" with the number of a window that has not yet been opened. (See the section on opening windows elsewhere for more details.)

Now, use the MM/1 hot keys (the Function keys in DWindows 1.0) to find the new window.

Now, type

```
_$ chd /dd/sys  [ENTER]
_$ me emacs.hlp  [ENTER]
```

and once inside Emacs, type

```
[CTRL][X]  [M]
```

responding with

```
view  [ENTER]
```

This loads the Emacs help file into Emacs in View mode -- that is, you can't inadvertently edit the Emacs help file.

How do you find specific help? Try the forward search mode. This handy Emacs feature allows you to type in a word, one letter at a time, whereupon Emacs will advance the cursor to the next word that matches the characters you have typed.

With the cursor at the top of the Emacs help file, type

```
[CTRL][X]  [CTRL][S]
```

In this mode, start typing characters of the word you seek -- say "deleting". Notice that you only need to type two characters to find the first occurrence!

If this is the section of the help file you want, press "Esc" and you are done.

If you want to find the next occurrence of your target string, continue the forward search by press

```
[CTRL][X]
```

The next occurrence of the string you typed will be found, and the cursor will mark that place. You can continue using [CTRL][X] until you have found the right spot in the file. If you go too far in the file, simply press the "Backspace" key.

Using Proff with Emacs

Now that you know some of the fundamentals of Emacs, you may want to know how to print a file that you created with it.

To do this, use Proff. Like its counterparts in the Unix world, nroff, mroff, troff, and others, it uses a system of "dot" commands to tell your printer what to do. Part of these commands is the ability to send special codes to your printer to take advantage of its unique features.

For most printing jobs, though, boldface (via double-strike), underline, centering, justification, headers, footers, and margins are about all you will need to know. Below, we discuss some of these features. Please refer to the Proff documentation on disk for more information.

Proff works by printing your file line by line until it reaches a line that begins with a period. These tell Proff that what follows on that line is a command it should pay attention to. In this case, by putting the ".left margin 5" command at the top of your file, all lines that follow will be offset from the left by five spaces, thanks to Proff.

To justify text

If you like the neat look of a justified right margin, you will be glad to know that Proff defaults to a justification mode. You can turn it off with

```
.nojustify
```

It is better to turn it off when doing chapter titles or tables. Place this command right above where you want justification to stop (also see below about spacing). To turn justification back on, type

```
.justify
```

Another feature that Proff uses as a default is "fill". Proff will fill a line of text with spaces to ensure that it can be gracefully justified. In some cases, then, when you do not want extra spaces inserted by Proff, just turn off fill mode with

```
.nofill
```

When you get to text where you want to fill it with spaces and justify it, you type

```
.fill
```

To center text

Centering your text is as intuitive as justification and filling. Right above the line where you want the text to be centered, type

```
.center on
```

and afterwards, type

```
.center off
```

A Proff example

Suppose you wanted to see the following:

```
                          ###

            Coming to the aid of the country
                      by V. Zimmer

Consider the citizen when confronted with the request

        Now is the time for all good men to come to the aid
of the country.

What a poor spirit he or she would be to rally to the
cause without thought! And what a poorer spirit they
would be to ignore the request.    Citizenry  is  as
begrudged by humans as mortality and taxes.   Yet it is
our lot, and so the citizen must consider the request
with  the  full  moral,  intellectual,  and  emotional
resources at hand.

                          ###
```

To print this kind of document, you must type the text and insert appropriate "dot" commands above each line where a formatting change is desired.

To center the title, you must turn centering on, and turn off fill and justification. You must then turn off centering and restore fill and justification. And when the quotation is referenced you must indent that paragraph or sentence a little more to the right. Afterwards, restore the indentation to normal.

Try it out! The commands you will use are

.in, .justify, .center, .fill, .lm

To get you started, the Emacs file for the title should read

```
.lm 5
.center on
.justify off
.fill off
Coming to the aid of the country
```

```
by V. Zimmer
.center off
.justify on
.fill on
```

To print with Proff, get your continuous-feed printer ready for printing, then type:

```
_$ proff filename >/p [ENTER]
```

If you don't have a continuous-feed printer but instead have a single-sheet printer, before you place any paper in the printer, type

```
_$ proff -s filename >/p [ENTER]
```

Now insert your paper and press [ENTER] for each page to be printed.

For more information, please be sure to read through or print out the documentation provided on Distribution Disk #5.

Giftoiff/iffs

The MM/1 uses the same graphics file format as Compact Disk-Interactive. This format, known as CD-I/IFF, is based on the IFF standard from Electronic Arts. CD-I/IFF is part of the Sony and Philips Green Book specification for CD-I and is a supported format by the ANSI committee on time-based hypermedia.

There are other formats for graphics, including GIF, TIFF, LZH, and many others. GIF is commonly used on the CompuServe™ Information Service (CIS), particularly by users of IBM-PCs.

You can download GIF files with Sterm, a terminal program included with the MM/1. Also included with the MM/1 is a "snap pack" courtesy of CIS that allows the first-time CIS user access to CIS with some free time. Sterm supports a CIS data transfer protocol called CompuServe B, a very fast and therefore cost-effective protocol, especially for large graphic files such as those used by the MM/1, Amiga™, IBM-PC, and Macintosh™ computers.

You may also be able to acquire GIF files from software vendors.

Once you have a GIF file, you can convert it to CD-I/IFF with the Giftoiff utility. The syntax for this utility is

```
_$ giftoiff file.gif >file.iff [ENTER]
```

Once you have converted the graphics file File.gif to the MM/1's CD-I/IFF format, you may delete the original file.

To display your new file, create a text file named file with the following line in it:

```
file.iff
```

then type:

```
_$ iffs file.iff
```

The IFF slide show program will present your picture.

Creating a Slide Show

If you have a series of CD-I/IFF files, you can present a slide-show by creating a text file with MicroEmacs of the names of each file, one filename per line, and then invoking Iffs. For example, if you have files `file1.iff`, `file2.iff`, `file3.iff`, and so on, first create a text file (named, say, `slide.show`) that contains

```
file1.iff
file2.iff
file3.iff
```

Make sure this `slide.show` file exists in the same directory as the graphics files themselves. (Alternatively, you can specify the full path name of each file, a handy method for displaying files that are stored in different directories from each other.)

You can now show the graphics files, one after the other, with the command

```
_$ iffs slide.show [ENTER]
```

The slide show will "rewind" itself and start over when it reaches the last file in the list.

Hdplaym/Playm/Plays

The MM/1 Extended permits the sampling and playback of digitized sound. This sound can be stereo or mono at a variety of sample rates. These sound files can also be large. For example, a three-minute version of a big-band jazz selection can be 700 KBytes.

To aid in the playback of these sound files, your MM/1 system comes with three simple utilities. Hdplaym will play large sound files directly from you hard disk drive, using a minimum of memory. Playm will play the same sound files, but entirely from memory. Plays will play stereo

sound files. Be sure you know whether the sound file you want to play is mono or stereo, or results can be not what you intended.

Plt/Display/Color

The Display command is included with the MM/1 to allow sequences of computer code to be sent to the display drivers. Using the appropriate codes, a programmer or user can tune a window to his or her needs. A discussion of these codes is included in the Appendix entitled, MM/1 Display Codes, beginning on page C-2.

If one of your programs leaves the colors of the MM/1 screen in an unusual state, you can restore the MM/1 to its original palette configuration with the Plt command. Simply type

```
_$ plt [ENTER]
```

To use the Plt command, it must be available in memory or in your current execution directory, along with Runb and the Display command.

If colors are still not to your liking, you can change them with the Color command. The Color command lets you set the foreground (text character) and background (screen) colors. Simply type

```
_$ color [ENTER]
```

for the Color help screen, or specify the colors you want as arguments:

```
_$ color yellow on grey [ENTER]
```

To execute, ensure that Color is in memory or in the execution directory along with Runb.

Smersh

The OS-9/68000 Shell provides several features that ease the typing of commands at the OS-9 prompt. First, it allows you to use the Backspace key to back up your cursor over the characters you typed. As the cursor moves backwards over the characters, they disappear. Simply backspace to the point in the command line where you made your mistake and retype to the end of the line.

If you don't catch your mistake in time, you will see results on your screen that were not what you intended. To try again, the OS-9 Shell allows you to strike [CTRL][A] (the [CTRL] key and the [A] key pressed simultaneously). The last command line executed by the OS-9 Shell re-appears before you. You can now use the Backspace key to edit your line.

For many users, these editing features are not enough. For this reason, the MM/1 comes bundled with a utility called Smersh. Smersh is a very simple line editor that follows the following conventions:

[INSERT]	The same as typing CTRL-C
[UP ARROW]	Display the last command entered. Press up arrow again for the command before that
[DWN ARROW]	Reverses the action of the up arrow key
[LFT ARROW]	Move the cursor to the left; non-destructive
[RT ARROW]	Move the cursor towards the end of the line; non-destructive
[DELETE]	Delete the character under the cursor

To execute Smersh, type at the OS-9 prompt:

```
_$ smersh [ENTER]
```

You will not be able to escape from Smersh as it traps the [ESC] key. When finished with Smersh, move to another window and type

```
_$ procs [ENTER]
```

Find Smersh in the module list. Look across to the far left column for the process id for Smersh. Use this number with the Kill command to stop Smersh.

```
_$ kill n [ENTER]
```

Here, "n" is the number of Smersh's id.

Sterm

One of the most popular uses for computers is telecommunicating with other computer users. This is often done via commercial information services. The MM/1 comes bundled with CompuServe™ Information Services (CIS) starter kits that offer some free time for first-time users.

You can access CIS with your MM/1 if you have a modem and serial cable that adapts from the MM/1's DB-9 to a DB-25 used by most modems.

Once your modem is set up and switched on, you will need to set the baud rate of the serial port

you use to match the lower of the following two baud rates:

- The maximum baud rate available from your information service
- The maximum baud rate of your modem

For example, if you have a 2400 baud modem and the information service permits access at 9600 baud, you will set the baud rate of your serial port to 2400 baud. If you modem permits 57.6 KBaud but you want to access a friend's computer system at 1200 baud, you will need to set the port to 1200 baud.

If the port you chose for your modem is /t0 (the lowest DB-9 serial port on the MM/1 backplane, next to the video connector), and if you wish the port to work at 2400 baud, type

```
_$ xmode /t0 baud=2400 [ENTER]
```

For more information on the location of your serial ports, please refer to the Appendix entitled "MM/1 Backplate Layout" on page B-4. Now, with the modem turned on, execute Sterm, the popular telecommunications package bundled with the MM/1:

```
_$ sterm -1 /t0 [ENTER] (note: 1 is a letter)
```

Sterm now tells you it is online. You can now have the modem dial your local access number for CIS or other information service or electronic bulletin board service (BBS). You may also be able to dial into computers at work, or, if using the MM/1 at work, to dial into your home computer from the MM/1.

You will note that Sterm currently supports XMODEM and CompuServe™ B protocols. XMODEM is widely used on information services. CompuServe™ B is used only on CompuServe. It is a high-speed, cost-saving protocol. Which method you use for uploading or downloading files, you will be prompted for the information you must provide Sterm to successfully complete the transfer. Consult the Sterm manual for more information.

Fildes

Fildes is a utility that provides the knowledgeable programmer and user the ability to manipulate the OS-9 descriptor of a file. **USE THIS UTILITY WITH GREAT CARE. IF YOU DO NOT UNDERSTAND HOW TO USE IT, YOU MAY DAMAGE YOUR FILES.**

You can change user and group id's, attributes, file creation and file modified dates, and even the file size. For more information, simply type

```
_$ fildes [ENTER]
```

for the Fildes help screen. Consult your OS-9 manuals for information on file descriptors.

Find

Find recursively wanders through a directory structure and processes the filenames it finds in various ways.

For example, you can "filter" through the files in a directory structure to find just those files you want by using wildcard searches. To do this, you simply use "find" with the name of the directory, the wildcard string, and the "-f" option:

```
_$ find IFF -f -name "*.iff" [ENTER]
```

will count the number of files contained in the IFF directory whose names include the ".iff" extension.

If you want to print the names of these files, just add "-print" to the command.

You can also specify actions to be taken on each file, including sorts or listing to a printer.

As usual, if you need more information, simply type the name of the utility with no arguments and press [ENTER] to get a help message.

Findstr

This command permits you to find an ASCII string within a text file. Much like the OS-9 Grep command, Findstr is often twice as fast or better. You can do searches that are case sensitive or case insensitive, and you can search for number or non-printable characters. Use Findstr by itself on a command line to access the help screen.

Lnlist

Lnlist is a simple print filter. It provides headers, permits printing of portions of documents, and supports some common printers for special printer features. Type the utility's name on the command line alone to access the help screen.

Ls

LS is a version of the popular UNIX directory listing utility, with extra features that are powerful and useful. For example,

```
_$ ls -Y [ENTER]
```

will tell you how many bytes are used by the files in the current directory. With other options you can select the directory listing to include file names along with specific data about each file, including creation date only, or complete pathname only, and so on.

For a complete listing of its features, type the command name alone on the command line and press [ENTER].

OddJob

This section provides an introduction to the powerful OddJob script processing language from TechnoTeacher, Inc. See the separate OddJob manual for a complete description of the language and its capabilities.

OddJob interprets scripts written according to the OddJob script specification. OddJob scripts greatly surpass the power of ordinary OS-9 scripts. OddJob resembles the UNIX C shell and wraps in features of awk and chat. These UNIX utilities and the UNIX C shell are among the most commonly used and most versatile software devised for microcomputers.

OddJob provides string processing, control flow, calculations, and input/output operations with familiar keywords:

String processing:	<set>, <arraytok>, <replace>, <alphameric>, <substring>, and <sort>
Control flow:	<if>, <else>, <endif>, <switch>, <case>, <default>, <break>, <loop>, and <continue>

Calculations can be made on variables with double-precision floating-point, n-dimensional arrays, and C-style operations.

Input/ouput:	<fprintf>, <openfile>, <closefile>, <readfile>, <writefile>, <seekfile>, and <tellfile>

More advanced OddJob capabilities

OddJob uses a process-interaction model that resembles Berkeley UNIX sockets. OddJob scripts can simulate the behavior of an interactive user. A script may be used to remove the necessity of human interaction in the completion of a program or process. An OddJob script can consult another process running remotely or locally.

The OddJob Debugger

Scripts can be debugged interactively with commands for setting and killing breakpoints, inspecting or setting variables, and so on.

Script timing can also be estimated with <julian>, <marktime>, or <difftime>.

Sample OddJob scripts

Distribution Disk #5 contains sample OddJob scripts.

Deciphering OS-9/68000 Error Messages

No one's fingers always type perfectly. When OS-9 notifies you that it doesn't understand what you want, it does so with error messages.

Microware's OS-9/68000 operating system is widely reputed for the quality of its error messages. There is a convenient method for finding the meaning of these message numbers. Try the Error command from IMS. At the prompt, type the error message number. (Error requires that the default drive /dd contain the file /dd/sys/errmsg and that Grep be available in memory or in your current execution directory.)

For example, if you type

        _$ em [ENTER]

and no command named "em" exists, OS-9 will tell you

        shell: can't execute "em" - Error #000:216

If you need more explanation, type

        _$ error [ENTER]

at which point you will see:

    error number?

Now type in the number on which you would like more information -- in this case, type

        error number? 216 [ENTER]

You will almost immediately see:

```
000:216 (E$PNNF)  File not found.
```

Now you know that OS-9 could not execute "em" because it couldn't find the file. For more information on how OS-9 finds files, see the discussion on directories elsewhere in this manual.

Getting Help With OS-9/68000 Commands

Using the Help command

At any time you can get more information on the usage of a command with the Help command.

Simply type

```
_$ help command.name [ENTER]
```

at the prompt.

Another method for finding help

The OS-9/68000 operating system also strongly suggests to programmers that help message be available to users of their programs when the user type a question mark after the program name. Almost all OS-9 programs conform to this suggestion.

You can therefore access help for a command without the Help utility by simply typing

```
command.name -? [ENTER]
```

at which point a help screen should appear.

MM/1 Windowing Software

Opening Windows

The MM/1 permits you to mix text and graphics on the same screen. Most of the time, you will be using a 640 x 210 display since this resolution is widely used for text applications, and because it can resolve 16-color graphics from a palette of 16.7 million colors.

If you wish to use another resolution mode for a window, see the section below, "Window Types".

To open a window, you can try one of two methods. Both methods involve starting the OS-9 Shell program on your new window. The first method actually tells Shell the name of the window. The second methods lets Shell choose the next available window.

The first method requires that you know which windows are already in use. Type

```
_$ procs [ENTER]
```

You will see a list of all processes that OS-9 is aware of. These processes include Shells or programs running in windows, so on the right hand side of the listing you will see the names of the programs and their respective windows.

Choose any window number not represented on that list. Say "/w4" is not listed in the `procs` output. Now you can type

```
_$ shell <>>>/w4& [ENTER]
```

This command line starts a Shell in window /w4. You generally will type the redirection symbols " < " and " > " exactly as you see here -- that is, one pointing left, then three pointing right. For more information, see the Microware manual set or other documentation concerning input/output redirection.

You don't need to create a Shell to run in /w4. It can be any command or program at all. You

could easily type

```
_$ me <>>>/w5& [ENTER]
```

to start up MicroEmacs in /w5.

Notice the ampersand ("&") symbol. This forces the program into the "background" in comparison to the current window. That is, the program you execute will do so not in the current window, but in another.

You also must use the slash symbol in front of the window. This is because /w4 and other windows are OS-9 devices, and all OS-9 device names begin with a slash.

Closing Windows

If you wish to close a window, you can kill the process or press [ESC] from that window.

First, you should ensure that the window is running no applications. Then you can type

```
_$ procs [ENTER]
```

to find the window's process id, located in the far left column under the heading "Id".

With this number, you can now type

```
_$ kill n [ENTER]
```

where "n" is the window's process id.

Alternatively, you can first close or remove any applications running on a window that you wish to delete, then press the [ESC] key. This generates an End of File condition for the window, which generally stops the window.

Making Windows Hard to Kill

If you wish to make your new window "escape proof" so that an accidental touching of the [ESC] doesn't kill your window, you can use the Shell's "-l" option. Using the lowercase L option when starting a Shell in a window forces the user to deliberately log out of the MM/1 system to remove the Shell. Thus, after you type

```
_$ shell -l <>>>/w6 [ENTER]
```

the Shell you created will be immune to the [ESC] key. It can still be killed by using the Kill command with the window's process id as an argument.

Window types

The MM/1 has several resolution modes. Not only are these handy for different types of graphics files, you can choose a text mode to suit your needs. Higher resolution screens can be used to preview documents, lower resolution screens can be used for text on games.

To set the type of window you would like, use the Xmode command with the Type option.

```
_$ xmode /wn type=xx [ENTER]
```

where /wn is some window that has not yet been opened ("n" is the number of the window). Replace "xx" with one of the type numbers listed below:

Type 00	16 colors	640 x 210	80 x 26	Border
Type 02	16 colors	640 x 420	80 x 52	Border
Type 03	256 colors	320 x 210	40 x 26	Border
Type 05	256 colors	320 x 420	40 x 52	Border
Type 06	16 colors	640 x 210	80 x 30	No Border
Type 07	16 colors	640 x 420	90 x 60	No Border
Type 08	256 colors	320 x 210	46 x 28	No Border
Type 09	256 colors	320 x 420	46 x 60	No Border

If you try to set a window type that is inappropriate for a program, your program may not function correctly. Please check a the application's documentation for details on how to establish the correct window type.

Expanding Your MM/1

The MM/1 was carefully designed to give you years of computing enjoyment. Part of this design is expandability. You can greatly enhance your MM/1's capabilities without the need for expensive add-on cards.

Sound

The MM/1 comes with a stereo sound port for both sampling and playback. To take advantage of this, you can purchase a cable that converts a 5-pin DIN connector (male) to 4 stereo jacks (male). Two of these play back sounds while two are used for sampling. The center pin on the 5-pin DIN is used to ground the signals.

The cable is all you need to begin sampling. Using a line-level device such as a stereo component tape deck or synthesizer at line level impedance, you can sample sounds on the MM/1. Call Interactive Media Systems, Inc. for information on software for sampling. To play back sounds, attach the two output cables to powered speakers or to a stereo amplifier with speakers.

Joystick

The MM/1 uses Tandy™ compatible joysticks made popular on the Color Computer™ and the Tandy 1000 series of computers.

For applications or entertainment software that requires a joystick, call IMS for information on the availability of a software driver if one is not supplied.

Inter-IC Network

The MM/1 has the built-in capability to transfer data and control information to other computers or to specialty circuits. Based on a Signetics protocol, the MM/1's network is a powerful, high-speed serial connection that can address up to 128 MM/1s.

To use the network as a data network between other MM/1s, contact IMS for cables and a

driver.

MIDI

The MM/1 CPU board has one serial port that can be configured for Musical Instrument Digital Interface (MIDI). MIDI is a standard digital format for specifying musical (and even non-musical) events. Pitch, note duration, note modulation, patch change information, and so on can be specified by a series of MIDI bytes.

MIDI is most commonly used to control and coordinate electronic music synthesizers. Much of the music you hear today on the radio and in the movies was created on computer with MIDI software controlling banks of synthesizers and drum machines.

MIDI can also be used to control recording studio equipment such as digital reverberation units, mixing consoles, and equalization. MIDI has even been used to control lights in a musical stage show environment.

By supporting MIDI in an inexpensive configuration, the MM/1 provides an extremely cost-effective method for creating state-of-the-art musical compositions.

The MM/1 with its windowing system is unique in its ability to improve the productivity of MIDI users. Contact IMS for the latest information on MIDI software and hardware availability.

Serial ports

In addition to the two serial ports provided with the MM/1 Personal and the third serial port provided with the MM/1 Extended, you can expand to two more serial ports for a total of five, with no expensive add-on boards required.

These serial ports attach to headers on the Extended I/O board and then connect to the backplate for secure operation. The ports can work at extremely high speeds and contain all necessary serial signals for extensive telecommunications.

SCSI

The MM/1's Small Computer Systems Interface (SCSI) allows you to expand beyond a single hard drive to two or three or more; you can also support CD-ROM drives that are SCSI compatible as well as tape drives for hard disk drive backups.

Available from third-party vendors for the MM/1 will be other devices for the SCSI bus. Please contact IMS for more information.

Monitors

The MM/1 uses video very similar to NTSC (American broadcast television) standards. Its Analog RGB video provides vivid color that can be used on monitors from the CM-8™ from Tandy up to Diamondscan and MultiSync monitors. Improved resolution can help you in certain critical applications. The MM/1 provided jumpers to allow you to work with most popular RGB-A monitors.

The MM/1 is also capable of using simple monochrome monitors, both composite and TTL. The extreme low cost of these monochrome monitors permits you to start using the MM/1 right away.

Appendix A: Installing Hard Disk Drives on the MM/1

Using an MM/1 with a hard disk drive is truly a pleasure. To help you understand how the MM/1 uses hard disk drives, below is a brief discussion of how the MM/1 hard disk drive interface was designed, how you use it, and why.

Background

The MM/1 Extended is the minimal configuration needed to support the use of a hard disk drive.

The MM/1 only uses Small Computer Systems Interface (SCSI) hard disk drives. This protocol, pronounced "scuzzy", has several advantages over competing protocols. First, SCSI is in wide use on personal computers, as well as on minicomputers, even mainframes. Second, there are many devices other than hard disk drives that use the SCSI protocol. These include tape backup drives, digitizers, Ethernet networking adapters, some floppy disk drives, scanners, and so on. Third, your MM/1 can support up to seven of these devices simultaneously on the MM/1 Extended's SCSI host adapter.

Installing a SCSI drive

To install a SCSI drive before the warranty period is over, contact IMS. Installing your own hard disk drive can void the warranty.

To install a SCSI drive after the warranty expires, first turn off the power to the MM/1, preferably by removing the power connector from the wall socket.

Make sure you have plenty of light and workspace. Remove all jewelry and make sure your hands are clean and dry.

Now remove the computer case cover, keeping track of which screws hold the case top down. You can use masking tape to tape the screws near their screw holes during installation.

Install the hard disk drive in a bay of your case according to the mounting instructions that accompany the drive, and find the SCSI cable that usually comes with SCSI hard disk drives. IMS can supply you with a cable if you need one.

Once the hard disk drive is physically installed, attach the hard disk drive cable from the header on the MM/1 Extended board to the hard disk drive. Usually SCSI cables have one edge colored differently than the rest of the cable. This color marks Line 1 of the cable. Line one should be oriented towards the back of the computer system towards the connects and backplate, and away from the front of the case. (Often SCSI cables are designed to ensure correct installation.)

Follow the hard disk drive's instructions on how to attach the cable to the hard disk drive.

Now you can follow the instructions "Installing Software On a Hard Drive" which begins on page 11 earlier in this manual.

Installing a non-SCSI drive

It is possible to purchase a non-SCSI hard disk drive, add to it a SCSI controller, and use it on the MM/1. If you have a non-SCSI hard disk drive, you may wish to consider this option.

IMS STRONGLY RECOMMENDS THAT YOU FIND COMPETENT TECHNICAL HELP IN ASSISTING THE USE OF NON-SCSI DRIVES WITH YOUR MM/1. Interactive Media Systems, Inc. will not be providing support for customers with non-SCSI drives since there are some many different problems that can occur.

In general, you should follow the following steps.

First turn off the power to the MM/1, preferably by removing the power connector from the wall socket.

Make sure you have plenty of light and workspace. Remove all jewelry and make sure your hands are clean and dry.

Now remove the computer case cover, keeping track of which screws hold the case top down. You can use masking tape to tape the screws near their screw holes during installation.

- Back up all the data from your non-SCSI drive. If you back up your data to 5.25" floppy disks in either PC format or Color Computer OS-9 or Atari ST OS-9 format, you may wish to also install a 5.25" floppy disk drive on the MM/1 while you have the case open. DO NOT TERMINATE THE 5.25" FLOPPY DISK DRIVE. Older 5.25" drives consume more power than the MM/1 typically permits. Termination of the drive can cause erratic results. Extremely old 5.25"

drives may not work.

●   Ensure that the MM/1 case can support your hard disk drive internally.

●   Obtain a SCSI controller for your drive. Please check with the manufacturer to ensure that it will work with your drive.

●   Attach the non-SCSI drive to the controller.

●   Install the disk drive in the MM/1, making sure that all connections are secure and that no metal touches any electronics.

●   Ensure that Line 1 of the cable is attached towards the back of the computer. The differently-colored edge of the ribbon cable will face the backplate and will be positioned away from the front of the case. Make sure it is secure.

●   With power attached to the drive, now plug in your MM/1 and turn it on.

●   WAIT AT LEAST FIVE MINUTES to let the hard disk drive warm up and settle in.

●   Now reformat the drive according to the instructions found earlier in this manual.

You can next either restore your data from floppy disks, or can use a null-modem cable, available at your local electronics store, to transfer the data to the MM/1 hard drive via a serial connection with your existing computer.

To use a serial connection, make sure both computers are set to the same baud rate. Then use terminal programs on both machines to facilitate the transfers. You can also use Kermit.
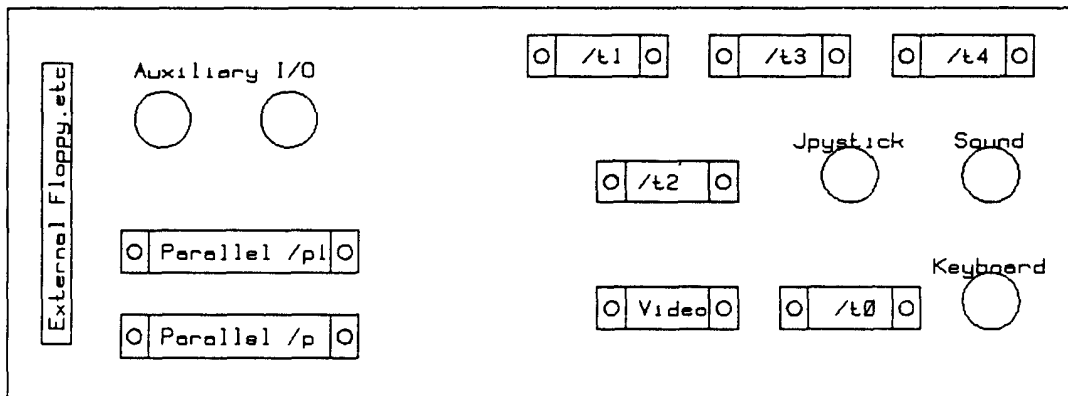
Appendix B: MM/1 Backplate Layout

Figure 1: MM/1 Backplate Layout

Appendix C: MM/1 Windows Display Codes

The company that supplies the windowing system has provided no documentation to date on the windowing software. That company recommends that MM/1 users use Tandy Color Computer windows display codes on the MM/1 to take advantage of the full power of the windowing system. The windowing software was designed to be compatible with the Tandy Color Computer windows Display Codes.

If you need further information on Tandy Color Computer windowing system, contact Interactive Media Systems, Inc. or your local Radio Shack dealer. Ask for information on the OS-9 Level 2 operating system with windows.

Appendix D: MM/1 Software

As of July 13, 1991, the contents of the MM/1 Distribution Disks are as follows:

MM/1 Distribution Disk 1: 1MB Boot & System Disk

Root Directory

CMDS	DEMO	FUN	OS9Boot	ReadMe	SYS
Startup	USR	color.note		demo1	
demo2	demo2.note		mergeit	mergeit.note	
move2h0	mylogin	os9boot.1meg.ext		startup.login	
startup.p					

CMDS

attr	backup	basic	cio	cls	cmp
color	copy	count	cpp	date	
deiniz	del	deldir	demo	dir	
diskcache	display	dmode	drive	dsave	dump
echo	error	flicker	format	format.2.3	free
giftoiff	grep	help	ident	iniz	link
list	load	loadbuffs	login	makdir	mdir
me	merge	mfree	moded	more	
os9gen	paint	pd	playm	plays	plt
printenv	procs	qsort	rename	runb	save
setime	shell	sleep	syscall	tee	
tetrix	tmode	touch	tsmon	unlink	xmode

DEMO

```
SNDS          color.fli            SNDS/ReadMe.snds
SNDS/phantom.m
```

FUN

```
ReadMe
```

SYS

```
emacs.hlp  errmsg     errmsg.short           moded.fields
motd       password   stdfont    stdfont_01.fnt
stdfont_02.fnt         stdfont_03.fnt         stdfont_04.fnt
stdfonts   stdpats_16            stdpats_256
termcap    umacs.hlp
```

NOTE: This directory also contains the invisible directory .BITMAPS, used by MM/1 applications to store fonts, brushes, and patterns. It currently contains dozens of public domain fonts for your use.

USR/CUST

NOTE: This directory is empty. You would use the USR directory for anyone that uses your MM/1 system in a multi-user setup. Consult IMS and Microware documentation for more information on how to set up your MM/1 for multiple simultaneous users.

MM/1 Distribution Disk 2: 3MB Boot & System Disk

Root Directory

```
CMDS       DEMO       PICS       SNDS       SYS
Startup    USR        mergeit    move2h0    os9boot
```

CMDS

```
anim       attr       backup     cio        cls        cmp
color      copy       date       deiniz     del
deldir     dir        diskcache  display    dmode
```

```
drive      dsave     dump      echo       error
format     free      grep      help       ident      iniz
link       list      load      loadbuffs  login
makdir     mdir      me        merge      mfree
moded      more      os9gen    pd         playm
plays      plt       printenv  procs      rename     runb
save       setime    shell     sleep      syscall
tmode      tsmon     unlink    xmode
```

DEMO

```
mouse
```

PICS/ANIMS/MOUSE

```
mouse.anm  mouse.ctl             mouse.iff  mouse.pic
```

NOTE: PICS will typically contain graphics files. The ANIM format is used by IBM-PCs and Amigas. The mouse animation, created on another computer system, is playable on the MM/1 with the MM/1 ANIM utility. Other files you can store in PICS might be GIF or MM/1 IFF files, available on CompuServe and Delphi.

SNDS

```
SNDS/SCREAMS          SNDS/SCREAMS/man
```

SYS

```
emacs.hlp  errmsg    errmsg.short          moded.fields
motd       password  stdfont    stdfont_01.fnt
stdfont_02.fnt       stdfont_03.fnt        stdfont_04.fnt
stdfonts   stdpats_16           stdpats_256
termcap
```

USR/CUST

NOTE: This directory is empty. Use this directory when you log into a multi-user version of the MM/1. Other directories can be placed in USR for other others as well. Consult the IMS and Microware documentation for more information on how to set up the MM/1 as a multi-user system.

MM/1 Distribution Disk 3: OS-9/8000 Other Commands

Root Directory

```
move2h0
```

CMDS

```
binex      break     build     c68       cc        cfp
chp        cio       ckermit   code      com
compress   cpp       dcheck    debug     demo      devs
diff       edit      edt       events    exbin
expand     fildes    find      findstr   fixlf
fixmod     frestore  fsave     gifshow   giftoiff
hdplaym    hdplays   iffs      iffshow   irqs    .
kermit     168       lnlist    ls        makdir    make
math       more      mtsmon    mv        ndir      nmon
nwatch     o68       oj        pr        proff     r68
rcp        romsplit  smersh    sort      sterm
syscall    tape      touch     tr    ·   umacs
```

MM/1 Distribution Disk 4: OS-9/68000 Tools

Root Directory

```
C          CGFX      CMDS      DEFS      LIB       PCF
PICS       SNDS      TOOLS     move2h0
```

C/SOURCE

```
README     cstart.a  example.c funcs.c   inittrap.a
makefile   mytraps.a           trapdefs.a
trapmain.a
```

CGFX

```
DOC         buffs.h   cgfx.l    mouse.h   winfo.h
DOC/cgfx.doc
```

CMDS

```
flicker    makdir    playm    plays
```

DEFS

```
MACHINE      RELS        cdfmdev.a cdfmstat.a
cdrvstat.a               ctype.h   dcmd.h      defs.copy
dir.h        direct.h    dma68070.d            drvstat.a
errno.h      events.h    float.h   funcs.a     io.a
iodev.a      ioglob.a    limits.h  makefile    math.h
memory.h     mk68901.d             modes.h     module.a
module.h     oskdefs.d             path.h      process.a
procid.h     rbf.h       rbfdev.a  rbfstat.a   sbf.h
sbfdesc.d    sbfdev.d    scc68070.d            scf.h
scfdev.a     scfstat.a             setjmp.h    setsys.h
sgcodes.h    sgstat.h    signal.h  stdio.h     strings.h
sysglob.a    sysio.a     sysio.h   systype.d   systype.d.asm
systype.d.cboot          systype.d.ind.68070
systype.d.pro.68070      tapehead.h            termcap.h
time.h       types.h     ucmdev.a  ucmstat.a   MACHINE/regs.h
```

LIB

```
cdidrvs1.l               cdidrvs2.l            cio.l
clib.l       clibn.l     cstart.r  drvs1.l     drvs12.l
drvs2.l      drvs20.l    drvs3.l   drvs4.l     drvs8.l
lib.save     math.l      sbf.l     sbfdrvtb.r
sbfstat.r    scfstat.l             sys.l       termlib.l
tickgeneric.r            ucstat.l  usr.l
```

PCF

```
IO           OBJS        read.me   read.me.first
```

PCF/IO

```
pcd0.a    pcd1.a    pcd2.a    pcd3.a    pcfdesc.a
```

PCF/OBJS

```
DESC.OMTI  DESC.VME320         pcf         DESC.OMTI/pcd0
DESC.OMTI/pcd1       DESC.VME320/pcd0
DESC.VME320/pcd1     DESC.VME320/pcd2
```

PICS

```
ReadMe     FLICKS/sample
```

SNDS

```
ReadMe     sample1.m
```

TOOLS

```
CMDS       KERMIT      MAIL        ReadMe      SPOOL       SYS
```

TOOLS/CMDS

```
com        grphfltr  kermit      mail        querymail  spl
splfltr    splman    splprt      umacs
```

TOOLS/KERMIT

```
KERMIT_DOC              KERMIT_SOURCE          read.me
```

TOOLS/KERMIT/KERMIT_DOC

```
os9ker.bwr              os9ker.doc             os9ker.hlp
```

TOOLS/KERMIT/KERMIT_SOURCE

```
date.list  makefile  os9con.c   os9get.c   os9inc.h
os9ker.c   os9qui.c  os9rec.c   os9snd.c   os9srv.c
os9utl.c
```

TOOLS/MAIL

ReadMe

TOOLS/MAIL/SPOOL

```
RELS        grphfltr.c           make.date makefile
queue.file             splfltr.c
```

TOOLS/SYS

```
mail.sys   termcap   umacs.hlp
```

MM/1 Distribution Disk 5

Root Directory

```
CMDS        EMACS.DOC            OJ.SCRIPTS
PROFF.DOC  ReadMe      STERM.DOC move2h0
```

CMDS

```
expand     CMDS/makdir
```

EMACS.DOC

```
ReadMe        amiga.rc_comp           azmap.cmd_comp
bdata.cmd_comp           block.cmd_comp
boxmac.cmd_comp          emacs.hlp_comp        emacs.rc_comp
emacs.tut_comp           emacs.txt_comp        emacsrc_comp
findcom.cmd_comp         func.cmd_comp
header.cug_comp          indent.cmd_comp       me110.rc_comp
me150.rc_comp            menu.cmd_comp         menu1_comp
readme_comp              submit.dsk_comp       trand.cmd_comp
```

OJ.SCRIPTS

```
ReadMe      fax.oj    ttienv.oj fax.oj    filetabs.oj
inizscf.oj            labp.old.oj         massed2.oj
massedit.oj           printertest.oj      rcxlabl.oj
rcy.oj    rcychgd.oj            rcyctrl.oj
rcydfmt.oj            rcydir.oj rcyexit.oj
rcyferr.oj            rcyinit.oj          rcymkdr.oj
rcyosem.oj            rcysatr.oj          rcyskpd.oj
rcystf.oj rcystop.oj            rcytatr.oj
rcytest.oj            rcyusg.oj rcyvoln.oj
rcz.oj    rczmkdr.oj            srnttilet.oj
toshmarg.oj           ttienv.oj ttilabel.oj
ttilc120.oj           ttilusag.oj         venturaindex.oj
```

PROFF.DOC

```
proff.doc_comp       proff.prf_comp
```

STERM.DOC

```
sterm.man_comp
```

MM/1 Distribution Disk 6

Root Directory

```
BOOTMODS    CMDS      DEMO      OS9Boot   OldBoot   SYS
demo2       demo2.note          move2h0
```

BOOTMODS

```
KERNEL    LISTS     NFM       PIPE      RBF       SCF
WIN       make.personal       makeboot  makehdboot
readme.first        readme.scsi
```

```
                    BOOTMODS/KERNEL
init.1mhd init3meg  init_base kernel    mk68901
readme.clock        rtcds1287 sysgo     sysgo.5
tk68901
```

```
                    BOOTMODS/LISTS
boot.1mfd  boot.1mhd            boot.3mhd boot.personal
bootlist.3mnohd
```

BOOTMODS/NFM

| n0 | n0nc | n6850 | n9026 | nc9026 | nfm |
| ns2 | ns3 | pipeman | | | |

BOOTMODS/PIPE

| null | pipe | pipeman |

BOOTMODS/RBF

| d0.hi | d0.st | d1.hi | dd.d0 | dd.h0 | h0h1 |
| r0 | ram | rb33c93 | rb37c65 | rbf | |

BOOTMODS/SCF

| nil | null | p | sc68070 | sc68230 | |
| sc68681 | sc68901 | scf | t0 | t1 | t2 |

BOOTMODS/WIN

keydrv	ms	ms68901	snddrv	stdfonts	term
w	w1	w2	w3	w4	w5w6
w7	w8	w9	windio		

CMDS

| cls | color | echo | flicker | makdir | me |
| os9gen | playm | runb | runflick | | |

DEMO

| membrane.fli | SNDS/nutcracker.m |

SYS

| emacs.hlp | termcap |

Index