ABOUT YOUR ORDER

Before you open the diskette package, examine the documentation to confirm you have received what you ordered. You should also determine from the documentation if the software you ordered will meet your needs. If you have received the wrong item, return the unopened disk package with all documentation and a note stating the problem and we will send you the correct item. If after looking at the documentation you feel there was a misunderstanding as to the function of the software, and it won't meet your needs you may return the UNOPENED disk package and documentation for a refund. No refunds will be given after the diskette package is opened (except for media that is defective according to the terms on the disk package).

ABOUT DOCUMENTATION

No amount of documentation will do you any good if you don't read it. The documentation included with this software assumes you have a basic knowledge of using your system and does not explain in depth information that is covered elsewhere (in your system manuals). We have tried to use terminology consistent with that used in the OS-9 system documentation. If you have not at least read through your OS-9 documentation that was included with your system we strongly urge you to do so. If you do not understand something about our documentation, first see if there is some word you skipped over that you did not understand (like "pipes", "device descriptor", "I/O redirection", etc.) that is explained in the OS-9 COMMANDS or OS-9 TECHNICAL MANUAL, study that manual then reread our documentation, if it still does not make any sense then try giving us a call.

DISKETTES

If you have difficulty reading the diskette supplied try it in more than one drive if you have more than one. If that doesn't work use the test program supplied with the RS OS-9 BOOT diskette to check your drives rotational speed. If the diskette has been exposed to temperature and humidity extremes it may need to sit for a day in your environment after you receive it to achieve dimensional stability. Another factor affecting diskette compatability is the track alignment on the disk drives (yours and ours), if they are off a disk becomes unreadable. If after these attempts you can still not read the disk return it for a replacement.

<div align="center">

D. P. Johnson
7655 S.W. Cedarcrest St.
Portland, OR 97223
(503) 244-8152

</div>

The diskette for **SDISK3** contains the following:

```
    directory of SDISK3 diskette  09:27:35
CMDS              MODS             PATCHES          SRC

    directory of CMDS  09:27:56
install           disktype         fls              sformat
modbuster         descgenL2

    directory of MODS  09:28:08
sdisk3.slow       sdisk3

    directory of PATCHES  09:28:21
slowboot.pat      d0.40ds.pat      fastbootstep.pat
w7term.pat        d1.40ds.pat      d1.80ds.pat      interleave3.pat
pcread.patch.b    pcread.patch.a

    directory of SRC  09:28:33
d1.3              dd.3             d0.3
```

The CMDS directory contains executable programs, refer to the documentation with this package for their use. You will probable want to copy SFORMAT and DISKTYPE to your working CMDS directory on your system disk.

If a directory named "DOC" is present, it will contain additional documentation test files with information that is not in the printed documentation. Read these files (use the LIST command) for more information about using programs on this diskette.

The MODS directory contains the SDISK3 driver modules. The sdisk3.slow file is a version of the driver that switches to the slow cpu clock rate when accessing the disk controller, this is needed for the older disk controllers using the 1791 controller chip (or other) that is not spec'ed for the hi clock speed. (Also I have found some "compatible" disk controllers have inexcplicable problems and it is probably better to use these with the .slow driver, the disk throughput is the same as for the fast clocking version since this driver only slows the cpu during the disk access which is limited by the speed of the data transfer from the disk anyway.)

The PATCHES directory contains patch files that are used with the MODPATCH utility. Most of these are used automatically the the INSTALL command.

The SRC directory contains some sample device descriptor sources. These are for reference only since all necessary descriptors are generated by the DescgenL2 command and the INSTALL command.

USE OF SDISK3 DRIVER

SDISK3 is a disk driver module that will replace the CC3Disk module in the OS9Boot file of your Level II OS-9 system disk. (COCO3 level 2 only.. NOTE: this is not the same as "version" 2.00 on the COCO2 which is "Level 1" OS-9). The device descriptors in the OS9Boot file, D0, DD, Dl ..etc. will be replaced also. To simplify the creation of a new system disk an automated installation macro (created by the Ll Utility Pak macgen command) is provided, see instructions under INSTALLATION INSTRUCTIONS.

INSTALLATION INSTRUCTIONS

The following installation procedure makes these assumptions:

- You have at least 2 disk drives (since the save utility is not included in the level 2 system disk a single drive installation is impractical.)

- Drive 0 is some type of 48 tpi drive (i.e. 35 or 40 tracks), the other drives may be any type.. if you already have drive 0 setup as and 80 track drive.. you will need to do a manual installation.

- The commands TMODE, COPY, DEL, LOAD, UNLINK are loaded in memory.. on the original level 2 system disk these are merge in the same file with the SHELL command so these are automatically loaded on startup unless you have altered this file.

PREPARATION:

(1) You will need 2 blank diskettes (these will be reformatted by the INSTALL macro).. label these "A" and "B", the INSTALL macro will refer to these as "DISK-A" and "DISK-B".

Boot your system from a backup of the original level 2 system disk, (the install macro will refer to this disk as the "SYSTEM" disk). Do NOT start any windows until the install macro is finished.

The Sdisk3 diskette will be referred to as the "SDISK3" disk by the install macro.

(2) After the system has booted (enter the date/time for the setime called by the startup file), then enter the following command:

       load modpatch

(3) Remove the SYSTEM disk from /d0 and insert the "SDISK3" disk, then enter the command:

       /d0/cmds/install -d=0

(NOTE: The "-d=0" will cause the device descriptor for "DD"
to be a copy of the descriptor for /d0... if you use -d=1
then DD will be a copy of the descriptor for /d1, etc.
Omitting the "-d=n" entirely will cause the DD descriptor to
be unchanged from the disk you booted.)

From this point on follow all instructions shown on the
screen. Answer all yes/no questions on the screen with a
single keystroke of "Y" or "N"..(don't be too heavy handed
on the keyboard so that you cause the key to repeat or you
will have "Pre-answered" the next question and foul things
up). When it is time to change a diskette the instructions
will be on the screen and the macro will wait for the
"READY? (Y or N)" question to be answered...type "Y" after
you change the disk.

When the above procedure is complete, DISK-B will be a new
bootable system disk with SDISK3 and new descriptors in place of
the CC3Disk drive. When operating with SDISK3 drive use the
SFORMAT command from the SDISK3 disk instead of the FORMAT
command that came on the original OS-9 level 2 disk. SFORMAT will
format all OS-9 5" disk formats.. in order to format an OS-9
format other than the CoCo format you must first use the DISKTYPE
command also on the SDISK3 disk to set up the desired device
descriptor for the format you wish.
    For easy re-configuration of you boot file, retain DISK-A
from the above procedure. Delete and unwanted modules from the
BOOTMODS directory, and copy any additional modules desired to be
in the new OS9Boot file you want to build to this directory. To
create this new bootfile on a newly formatted diskette, simply do
a chd to the BOOTMODS directory and then type:

        fls ! os9gen /d0   (or /d1 etc..)

NOTE: The fls command was on the SDISK3 diskette in the CMDS
directory, you could use "ls" from the L1 Utility pak in its
place. This sequence will include all of the modules in the
BOOTMODS directory in the new bootfile.

With SDISK3 you can automatically read a single sided diskette in
a double sided drive, or a 40 track disk in an 80 track drive.
Just put the diskette in the drive and access, no special action
is required. The determining factor in how a disk will be read is
how it was formatted. The format program (or Sformat)
writes information to sector zero of the diskette which tells
the system how many tracks, sides etc. the diskette has.

The following I$GETSTT functions are implemented in the sdisk3 driver:

SS.DREAD (Function code $80) Direct Read function reads specified sector into user buffer. Double density sectors up to 512 bytes may be read, single density sectors must be 128 or 256 bytes long.

Entry Conditions:

```
A = path number
B = $80
U = track (msb) / sector number (lsb)
X = buffer address to read data into
Y = sector size / FMT
```

:	Y-REGISTER CONTENTS	:
:Y15:Y14:Y13:Y12:Y11:Y10:Y09:Y08:Y07:Y06:Y05:Y04:Y03:Y02:Y01:Y00:
: lsb sector size : ms bits size : x : : : :
:S07:S06:S05:S04:S03:S02:S01:S00:S11:S10:S09:S08: - :tpi:dns:sid:

(Y contains following:)
bits 8-15 = least significant 8 bits of 12 bit sector size in bytes.
bits 4-7 = most significant 4 bits of 12 bit sector size with bit y-7 being the most significant bit of the count
bit 0 = side (0 or 1)
bit 1 = density (0=single, 1=double)
bit 2 = TPI   (0=48 TPI, 1=96 TPI)
bit 3 = not used (reserved for Hi-density)

Exit Conditions:
Buffer pointed to by X-reg contains data read from sector.

```
If error:
CC = C bit set
B  = error code
```

SS.SDRD (Function code $84) System Direct Read function.. same register entry and function as SS.DREAD except buffer address in (X) is in system memory map instead of user memory map.

The following special I$SETSTT functions are implemented in the sdisk3 driver:

SS.RST (Function code $03) Restore head to track 0.
SS.WTK (Function code $04) Format a track.
SS.FRZ (Function code $0A) Freeze DD. information.

The above function as described in the OS-9 Technical Information manual under the I$SETSTT function.

SS.DWRIT (Funtion code #80) Direct sector write. Writes data from user buffer to specified track, sector, and side of drive. Any sector size may be written in double density, and either 128 byte or 256 byte sectors for single density may be written.

Entry Conditions:

```
A = path number
B = $80
U = track (msb) / sector number (lsb)
X = buffer address to read data into
Y = sector size / FMT
```

:	Y-REGISTER CONTENTS	:
:Y15:Y14:Y13:Y12:Y11:Y10:Y09:Y08:Y07:Y06:Y05:Y04:Y03:Y02:Y01:Y00:
:   lsb sector size   : ms bits size : x :   :   :
:S07:S06:S05:S04:S03:S02:S01:S00:S11:S10:S09:S08: - :tpi:dns:sid:

```
(Y contains following:)
bits 8-15 = least significant 8 bits of 12 bit sector
                size in bytes.
bits 4-7  = most significant 4 bits of 12 bit sector
                size with bit y-7 being the most
                significant bit of the count
bit 0 = side (0 or 1)
bit 1 = density (0=single, 1=double)
bit 2 = TPI  (0=48 TPI, 1=96 TPI)
bit 3 = not used  (reserved)
```

Exit Conditions:
    Data from buffer at X-reg is written to disk.

```
If error:
CC = C bit set
B  = error code
```

NOTE: The verify function will work for all sector sizes up to and including 512 bytes. The SS.FRZ call is not necessary before using the direct read/write calls since the LSN0 information will remain unchanged during any direct read/write call.

SS.SDWRT (Function code $84) System Direct Write. Same function and register assignment as SS.DWRIT except the buffer address (X) is in the system memory map instead of the user memory map.

SS.UNFRZ (Function code **$81**) Unfreeze DD. information. (Reactivates the reading of LSN 0 to DD.**xxx** variables after SS.FRZ call has shut it off.

Entry Conditions:

    A = path number
    B = $81

Exit Conditions:
    none

NOTE: The SS.FRZ call is only in effect for 1 read of LSN0 after each SS.FRZ call.. for this reason the SS.UNFRZ call is rarely necessary.

SS.MOTIM (Function **$83**) Set drive motor on time constant. Allows setting the motor on time constant for the disk drives. This constant determines how long the motors will run after and access before shutting off. When the motors are off there is a half second delay for startup before the next access can take place..if the motor time constant is just a little too short to keep the motors running between operations (say a copy or compile) then the motor startup delay each time will add to the overall operating delay. Keeping the motors running constantly however adds needlessly to the diskette wear.

Entry conditions:

    A = path number
    B = $83
    X = time constant in clock ticks (1/60 second) 1-$FFFF

Exit conditions:
    CC = C bit set if error
    B = error code

DescgenL2

Syntax: DescgenL2 [-d=n]

DescgenL2 is a device descriptor generator program used to create
device descriptors for use with the SDISK3 driver module.
DescgenL2 will ask the number of drives and the characteristics
of each and will produce a device descriptor for each drive
0..(n-1). The descriptor will be written to the Current work
directory to a file named D0, D1 etc.. If the optional -D=n
parameter was given on the command line the descriptor "DD" will
be generated also which will be equivalent to drive "n", i.e. for
-d=0, descriptor DD will point to drive D0.

DISKTYPE

SYNTAX: Disktype [-opts] /device [/device ..]

Used to modify a disk descriptor for various OS-9 formats. If no
options are given then the current device setting is displayed
only.

Options: -c   Set for Color Computer OS-9 format.
           -j   Set for Japanese OS-9 formats.
           -m   Set for Mizar type OS-9 formats.
           -s   Set for Standard OS-9 formats.

Examples: Disktype /d0 /d1
                (identifies the current settings of d0 and d1).
            Disktype -s d1
                (sets /d1 to access any Standard OS-9 format).

Once the descriptor is set for a particular OS-9 type, you may
format that type with SFORMAT. (If you are using a Color
Computer, you should use the Color Computer OS-9 format for
all of your disks except when you need to transfer information to
another system that uses one of the other disk types.)

Example: If you need to read/write a Standard OS-9 disk format,
e.g. single density on /d1,

        disktype -s /d1

                        would set the drive for this type, you
could then do a dir, copy etc. to this disk, you can not use a
Color Computer format disk in this drive again until you do

        disktype -c /d1
                        to restore the drive to the COCO
format.

SYNTAX: MODBUSTER pathname

Performs the inverse function of the MERGE command, i.e. breaks a file containing several modules into separate files each containing one module.

Modbuster will create a file for each module contained in pathname in the current work directory with the file name being the same as the module name. If a file already exsists with the same name as the one being created, operation will be aborted.

example:
```
        makdir MODS
        chd mods
        modbuster /d0/os9boot
```

This sequence separates each module contained in os9boot into a separate file in the directory MODS. This is useful for rebuilding the boot. If you have the LS command which is part of Filter kit #1 an easy way to build a new boot file containing all of the modules in the mod directory is:

```
        chd mods
        ls ! os9gen /d1
```

By first making sure the mods directory contains all modules you desire to be in the new bootfile, this sequence will install a new bootfile on /d1 (provided you have a blank disk there). If you don't have ls you can use fls which is on the sdisk distribution version, fls is a stripped form of ls which lists all files in the current directory.

If you don't have the LS command then you would need to type in the module names for os9boot at the keyboard, or better yet since it is less error prone, use build or edit to create a file with these names then redirect the os9gen input to this file.

```
e.g.        chd MODS
            build bootlist
            ? sdisk
            ? d0
            ? rbf    (etc.)


            os9gen /d1 <bootlist
```

NOTE: Modubuster returns error #102 if there is insufficient memory to process the merged module, use the shell # parameter to allot more memory.

NOTE: The FLS command included in the SDISK3 cmds directory can be used instead of LS, e.g.   fls ! os9gen /d0

SYNTAX: SFORMAT /devname [opts]

Sformat when used with SDISK intalled allows formatting COCO OS-9 format diskettes with one or two sides and any number of cylinders, up to the capacity of the drive. When the device name ("/devname") is for a standard OS-9 format it will also allow formatting of the standard single and double density formats. If the "R" option is not given SFORMAT will display a table of format parameters for the given drive "/devname" and wait for operator response to quit the program, continue formatting, or change the parameters.

```
opts:  S = Single density   (valid for OS-9 standard formats only)
       D = Double density
       R = Ready            (proceed immediately with formatting)
       1 = 1 side
       2 = 2 sides
       4 = 48 TPI           (to format 48 TPI on 96 TPI drive)
       "disk name"
       ´no. of cylinders´
       :Interleave:
```

The above options will overide the default parameters taken from the device descriptor module for the format operation. If the "R" option is used, formatting will begin at once without waiting for the operator to enter (Y, Q, or N).

EXAMPLE:
 OS9: SFORMAT /D1

*** STANDARD DISK FORMAT ***
(C) Copyright 1983 D.P. Johnson
ALL RIGHTS RESERVED

FORMAT PARAMETERS:

```
    Double Density
 80 Cylinders
  2 sides
    Color Computer format
 18 Trk 0 Sectors
 18 Sectors/Track
```

Formatting drive /D1
y (yes), n (no), or q (quit)
Ready?N

Change to 48 TPI?Y
Double Sided?N
No. of Cylinders=35

FORMAT PARAMETERS:

```
    Double Density
 35 Cylinders
  1 sides
    Color Computer format
 18 Trk 0 Sectors
 18 Sectors/Track

Formatting drive /D1
y (yes), n (no), or q (quit)
Ready?Q
```

The "Q" entered above quits (exits) the program without
formatting the disk on /D1, by entering "Y" the formatting
operation would proceed. The parameters in the original table
are determined by the drive capabilites defined in the drive
device descriptor. These parameters can also be changed by
options included on the command line when sformat was invoked.
To format an OS-9 Standard Format disk, one of the device
descriptors must be confirgured for the standard format with the
DISKTYPE utility before SFORMAT is used.