

**NAME**

diskboot – disk bootstrap programs

**DESCRIPTION**

There are two versions of the disk bootstrap program, *uboot*. One handles RP03 type drives and the other RP04 type drives. Otherwise, the versions are functionally equivalent.

The program must be located in block 0 of the disk pack. The space available for the program is thus only one block (256 words) which severely constrains the amount of error handling. Block 0 is unused by the UNIX file system, so this does not affect normal file system operation. To boot, the program must be read into memory starting at address 0 and started at address 0. This may be accomplished by standard DEC® ROM bootstraps, special ROM bootstraps, or manual procedures.

After initial load, the program relocates itself to high core as specified when assembled (typically 24K words, maximum of 28K). Next, memory below the program is cleared and the prompt '#' is typed on the console. A two digit field specifying the file system to use is expected. The first digit indicates which drive to use and the second digit which logical section of the pack to use. For example, 24 would correspond to /dev/rp24, or drive 2, section 4. By convention, section 0 always starts at cylinder 0 and is safest to use. No error checking is done on this field, invalid data will cause unpredictable results. Also, there is no error checking on disk reads.

After the file system select, the program prompts with a '='. The user must then enter the UNIX pathname of the desired file. The '#' character will erase the last character typed, the '@' character will kill the entire line, and 'A' through 'Z' is translated to 'a' through 'z'. Also, carriage return (CR) is mapped into line feed (LF) on input, and LF is output as CR-LF. The upper-case to lower-case conversion is used to handle upper-case-only terminals such as the TELETYPE® Model 33 or the DEC LA30. Therefore, a file name with upper case characters cannot be booted using this procedure.

After the name has been completely entered by typing CR or LF, the program searches the file system specified for the pathname. Note, the pathname may be any valid UNIX file system pathname. If the file does not exist, or if the file is a directory or special file, the bootstrap starts over and prompts '#'. Otherwise, the file is read into memory starting at address 0. If address 0 contains 000 407, a UNIX a.out program is assumed and the first 8 words are stripped off by relocating the loaded program toward address 0. Finally, a jump to address 0 is done by executing "jsr pc,\*\$0".

**FILES**

/usr/mdec/uboot - disk bootstrap  
/sys/source/util/rp03boot.s - RP03 source  
/sys/source/util/rp04boot.s - RP04 source

**SEE ALSO**

a.out(V), fs(V), unixboot(VIII)