BASIC OPERATION 2

INTRODUCTION

This chapter is primarily for the person who will operate Desktop Cyber. Just like the real thing, many people may use Desktop Cyber at the same time, but the operator is the person who interacts with the system through the console, while users interact with the system through the timesharing system or through batch jobs.

CONSOLE KEYBOARD USAGE

Desktop Cyber's console window is a 6612 display console emulator. Since the 6612 didn't have a full keyboard, not all keys are supported. If you try to enter a key which isn't supported, the emulator just ignores it.

Most keys map obviously. Although the 0 on the 6612 was to the left of the 1, Desktop Cyber uses the 0 right where it's found on the conventional keyboard.

The CR key maps to ENTER.

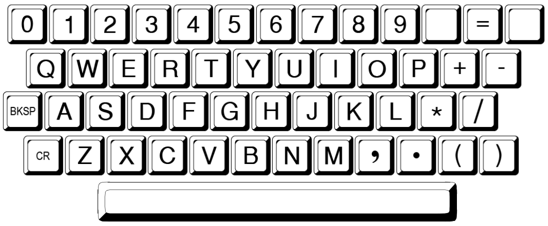
The only non-obvious mapping (for those who have not used a 6612 console) is for the two keys present on the 6612 but not seen on conventional keyboards—the left and right blank keys. The left blank maps to [and the right blank to].

NOTE

On a 6612, the left and right blank keys are on either side of the = key, as shown in figure 2-1 below. To a 6612 operator, pressing [and] for the left and right blank keys "feels" right.

The left blank clears the text you have entered on the command line (as a control-U in many UNIX shells). Except at dead start, in DIS, and O26, the right blank doesn't do anything. See the *NOS Version 1 Systems Programmer's Instant,* pub. 60449200, for details on DIS and O26 usage.

FIGURE 2-1. 6612 Keyboard



By default, Desktop Cyber's console emulator isn't pedantic about key rollover—on a true 6612 console, the operator couldn't type very fast; the console would only accept one key at a time, so the operator had to develop a somewhat awkward punch-and-wait keyboarding technique when sitting at the console. You don't have to worry about that—just type as you normally would. (If you are accustomed to the true rollover behaviour, you may set this option in the Desktop Cyber Operator Interface. See ***[Configuring Desktop Cyber Xref]*** for details.

DEAD START

The process of initializing a CDC Cyber/6000 series system is called *dead start.*[†]

On a real Cyber, a brief bootstrap program is toggled into a switch panel—called the dead start panel—behind a door on the central processing unit. Usually, this program is toggled in but once and only changed when necessary. Often, the switch configuration was written in longhand in octal on a punch card taped inside the door. This was done for two reasons:

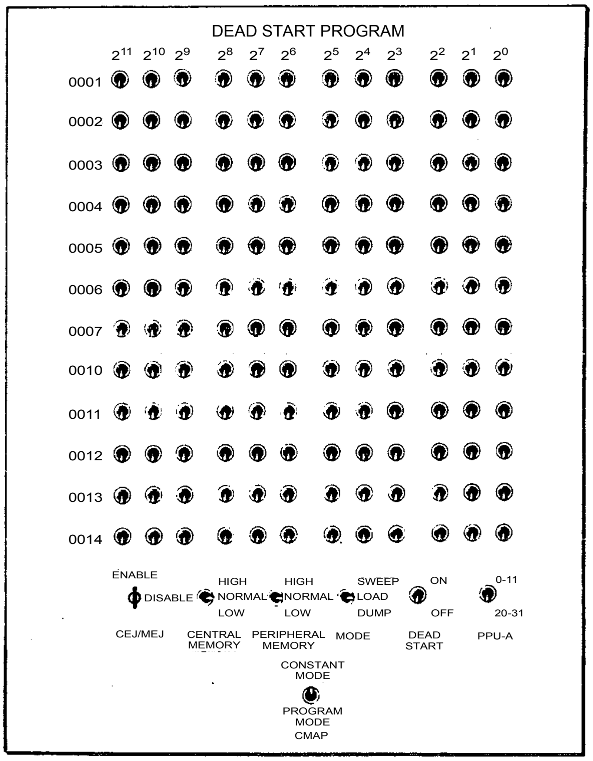
- Hardware engineers would often reconfigure the dead start panel during maintenance; a quick reference allowed the engineer to select easily from several configurations.

- In case the switches were accidentally jolted, the correct settings could be restored quickly and conveniently.

Dead start on real CDC hardware begins by actuating a momentary-contact switch either on the switch panel or on the operator's console. Although the operator could override various dead start panel parameters from the console, usually, once begun, the dead start proceeded automatically without operator intervention until the operator was solicited for the date and time.

A Cyber 70/6000 series dead start panel is shown below, adapted from page 2-3 of the *NOS Version 1 Operator's Guide*, pub. 60435600:

FIGURE 2-2. Dead Start Panel



Obviously, Desktop Cyber has no real dead start panel or dead start button. The dead start panel switches are configured in the *cyber.ini* file, and the dead start button is "pushed" by starting Desktop Cyber.

Here is how you normally dead start Desktop Cyber running NOS 1.4:

1. Log in to the host on its graphical console.

2. Start an xterm. You will keep this xterm open the entire time the system is up, since Desktop Cyber prints status messages in this window.

3. Verify that the DISPLAY environment variable is set correctly. It should be set to your display host name followed by ":0.0" as in *goldfinger.draughon.org:0.0*

NOTE

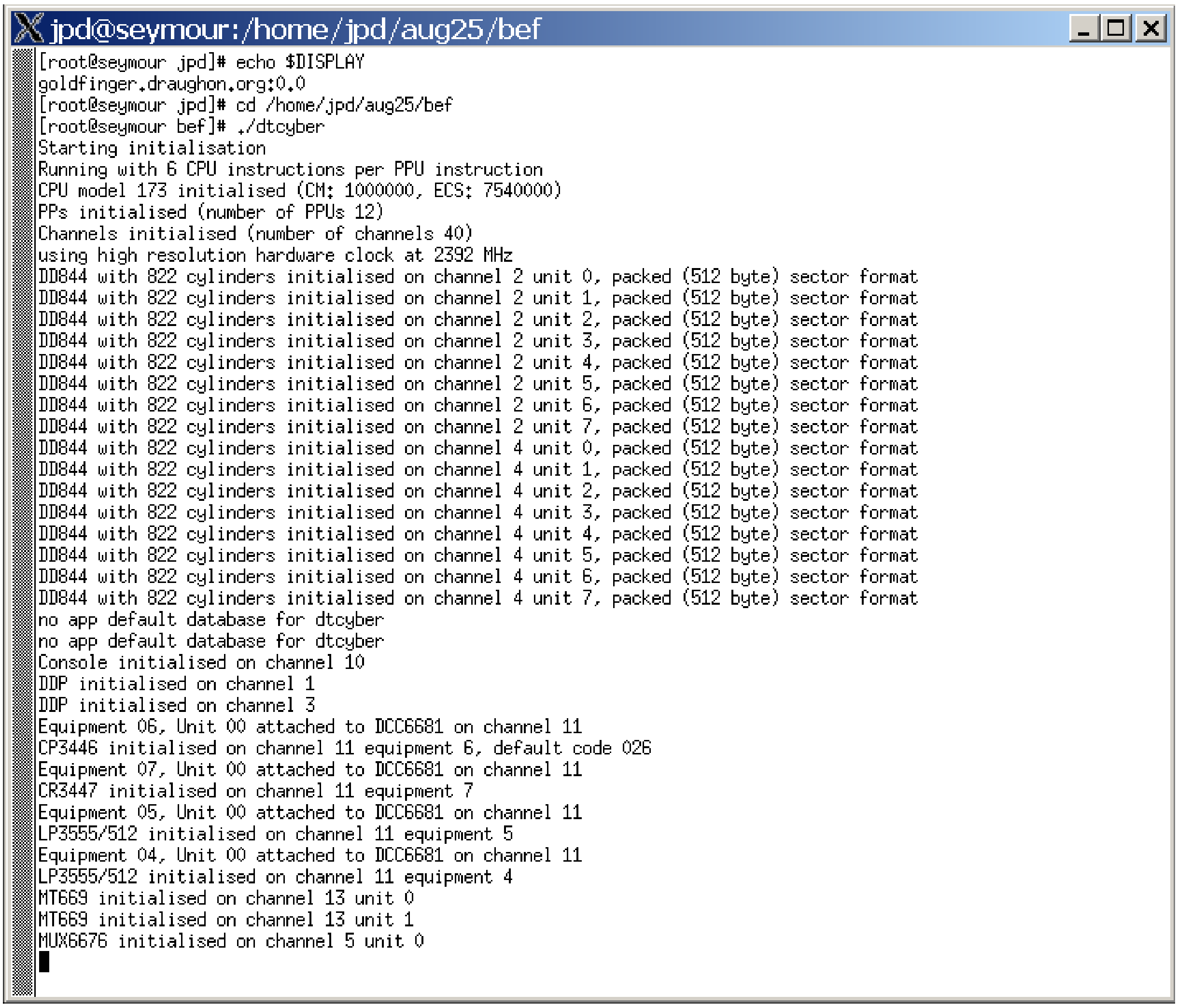
Desktop Cyber does not support the `-display` command line option. It must obtain the display server name from the environment.

4. Change to the Desktop Cyber home directory.

5. Type the command `./dtcyber` as shown below:

† A detailed but approachable discussion of dead start is discussed in the late James Thornton's *Design of a Computer: The Control Data 6600*. Mr. Thornton, with Seymour Cray, designed the CDC 6600 and therewith laid the foundation of supercomputing for decades to come. This book is available, with Mr. Thornton's permission, in PDF form at Al Kossow's *bitsavers.org* site in the directory */pdf/cdc/6x00/books* and is, even today, required reading for hardware design study. (There is also in the same directory a copy of the classic text on CDC assembly language by Ralph Grishman.)

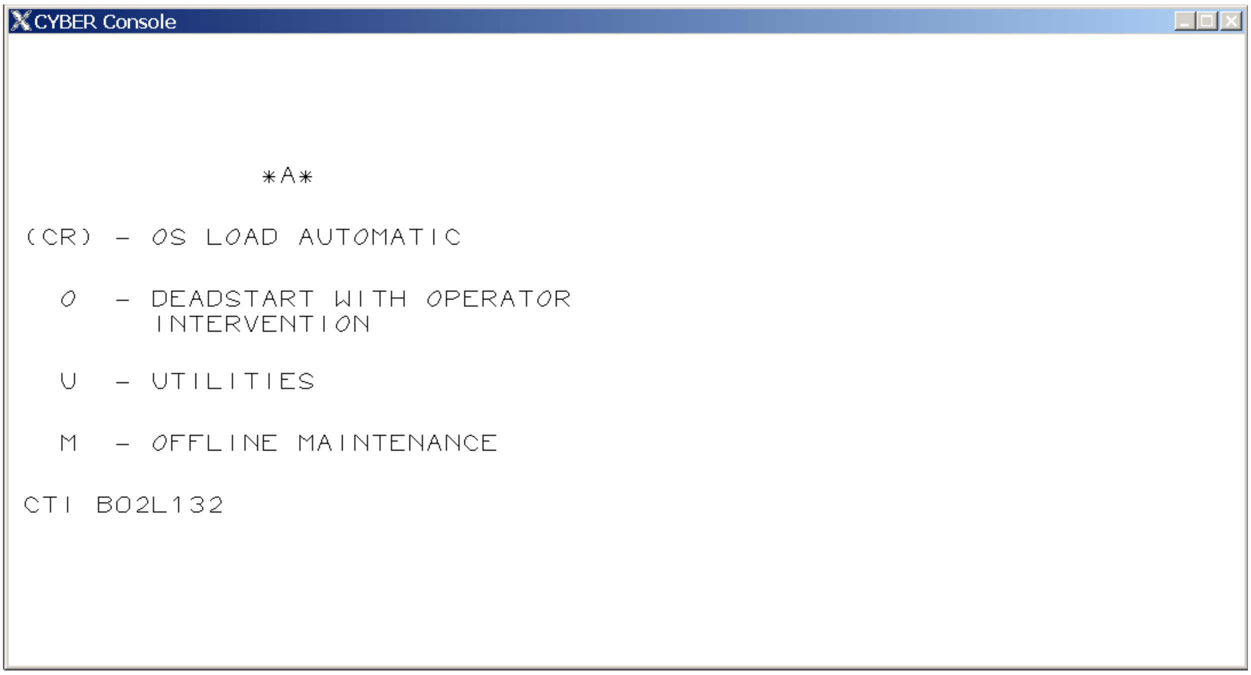
FIGURE 2-3. Starting Desktop Cyber



```
[root@seymour jpd]# echo $DISPLAY
goldfinger.draughon.org:0.0
[root@seymour jpd]# cd /home/jpd/aug25/bef
[root@seymour bef]# ./dtcyber
Starting initialisation
Running with 6 CPU instructions per PPU instruction
CPU model 173 initialised (CM: 1000000, ECS: 7540000)
PPs initialised (number of PPUs 12)
Channels initialised (number of channels 40)
using high resolution hardware clock at 2392 MHz
DD844 with 822 cylinders initialised on channel 2 unit 0, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 1, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 2, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 3, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 4, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 5, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 6, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 2 unit 7, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 0, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 1, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 2, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 3, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 4, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 5, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 6, packed (512 byte) sector format
DD844 with 822 cylinders initialised on channel 4 unit 7, packed (512 byte) sector format
no app default database for dtcyber
no app default database for dtcyber
Console initialised on channel 10
DDP initialised on channel 1
DDP initialised on channel 3
Equipment 06, Unit 00 attached to DCC6681 on channel 11
CP3446 initialised on channel 11 equipment 6, default code 026
Equipment 07, Unit 00 attached to DCC6681 on channel 11
CR3447 initialised on channel 11 equipment 7
Equipment 05, Unit 00 attached to DCC6681 on channel 11
LP3555/512 initialised on channel 11 equipment 5
Equipment 04, Unit 00 attached to DCC6681 on channel 11
LP3555/512 initialised on channel 11 equipment 4
MT669 initialised on channel 13 unit 0
MT669 initialised on channel 13 unit 1
MUX6676 initialised on channel 5 unit 0
█
```

6. The following X window will appear:

FIGURE 2-4. Initial Options Display



```
             *A*

(CR) - OS LOAD AUTOMATIC

  O  - DEADSTART WITH OPERATOR
       INTERVENTION

  U  - UTILITIES

  M  - OFFLINE MAINTENANCE

CTI B02L132
```

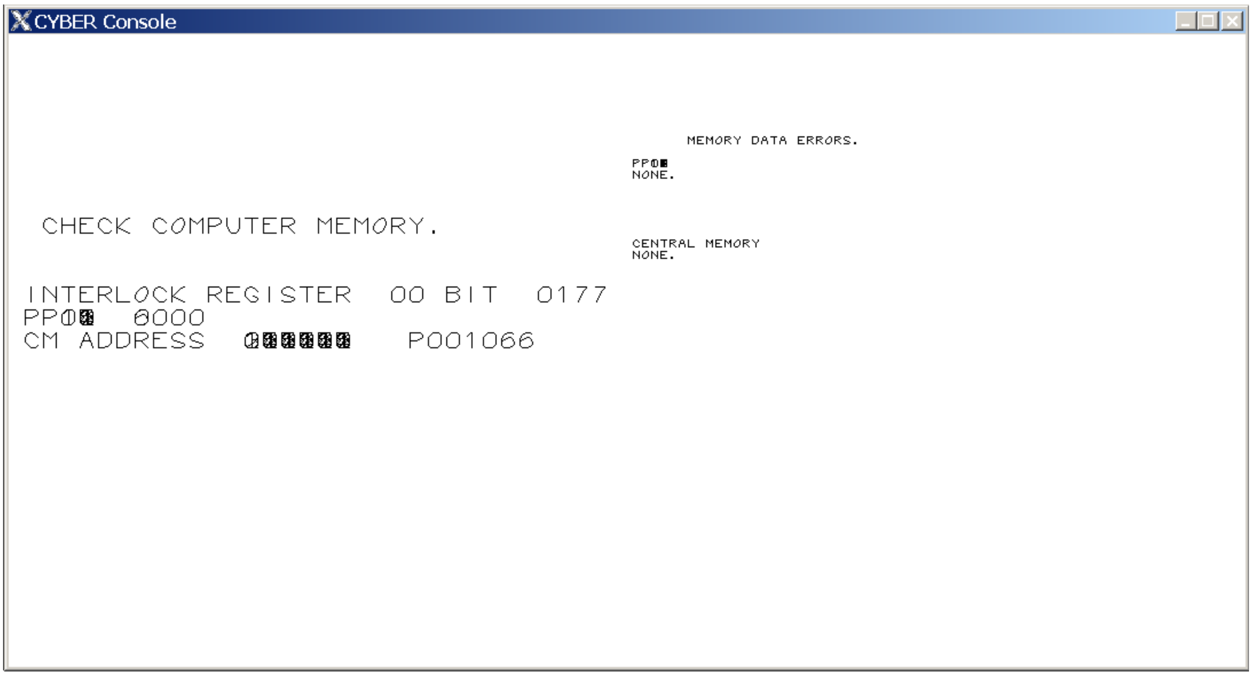
If you wish to accept the default set of dead start options, you may simply press ENTER and a level zero dead start will proceed without displaying the configuration decks.

NOTE

For a discussion of other dead start options, such as dead start level and configuration grammar display, see Appendix A.

7. The following will be briefly displayed as the system goes through its hardware verification sequence:

FIGURE 2-5. Initial Tests



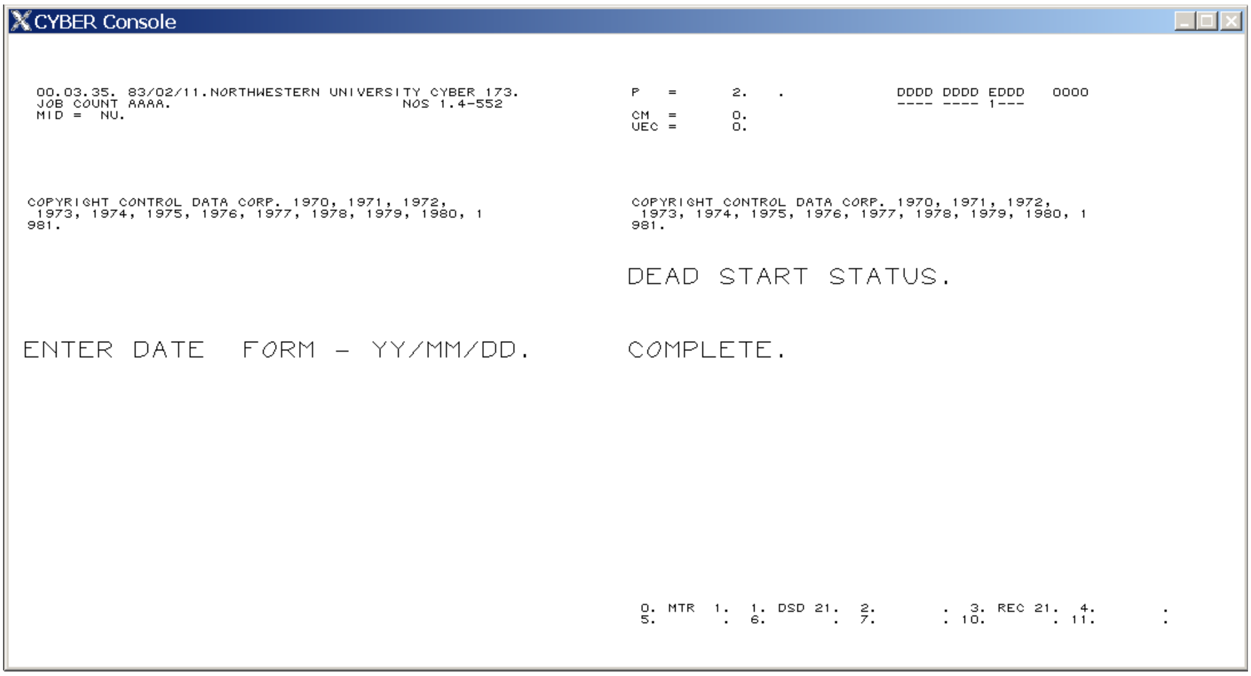
```
                                                          MEMORY DATA ERRORS.
                                                     PP0█
                                                     NONE.

 CHECK COMPUTER MEMORY.
                                                     CENTRAL MEMORY
                                                     NONE.

INTERLOCK REGISTER  00 BIT  0177
PP0█  6000
CM ADDRESS  ██████   P001066
```

After the hardware verification sequence, the operating system will load. You will see its progress on the right display, and you will be prompted to enter the date and time on the left display.

NOTE

If the dead start aborts, look up the message in Appendix B of the *NOS Version 1 Operator's Guide,* pub. 60435600.

FIGURE **2-6.** **Date/Time Entry**



```
00.03.35. 83/02/11.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AAAA.                             NOS 1.4-552
MID =  NU.


COPYRIGHT CONTROL DATA CORP. 1970, 1971, 1972,
 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1
981.



ENTER DATE  FORM - YY/MM/DD.
```

```
P    =      2.   .           DDDD DDDD EDDD   0000
                             ---- ---- 1---
CM   =      0.
UEC  =      0.


COPYRIGHT CONTROL DATA CORP. 1970, 1971, 1972,
 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1
981.

DEAD START STATUS.


COMPLETE.



 0. MTR  1.  1. DSD 21.  2.        .  3. REC 21.  4.       .
 5.       .  6.       .  7.        . 10.       . 11.       :
```

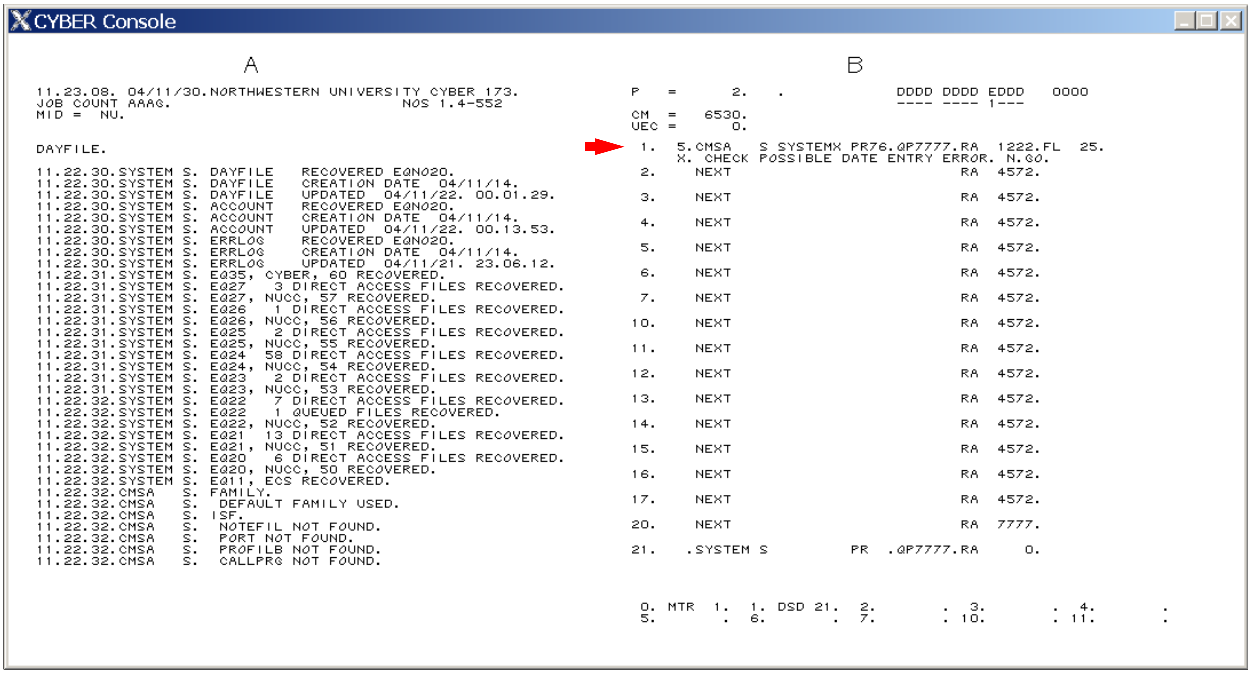
Enter the date and time when prompted, using the form shown, and pressing ENTER after each entry.

8. If the last dead start were more than one day ago (even if the system were up the entire time), you will see the following.

Note the message at control point 1 on the right-hand side.

The message is **CHECK POSSIBLE DATE ENTRY ERROR. N.GO.** If the date is correct, type `1.GO.` and press ENTER.

FIGURE 2-7. CMS Requesting Date Confirmation



```
                       A

11.23.08. 04/11/30.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AAAG.                               NOS 1.4-552
MID =  NU.


DAYFILE.

11.22.30.SYSTEM S.  DAYFILE   RECOVERED EQN020.
11.22.30.SYSTEM S.  DAYFILE   CREATION DATE  04/11/14.
11.22.30.SYSTEM S.  DAYFILE   UPDATED  04/11/22. 00.01.29.
11.22.30.SYSTEM S.  ACCOUNT   RECOVERED EQN020.
11.22.30.SYSTEM S.  ACCOUNT   CREATION DATE  04/11/14.
11.22.30.SYSTEM S.  ACCOUNT   UPDATED  04/11/22. 00.13.53.
11.22.30.SYSTEM S.  ERRLOG    RECOVERED EQN020.
11.22.30.SYSTEM S.  ERRLOG    CREATION DATE  04/11/14.
11.22.30.SYSTEM S.  ERRLOG    UPDATED  04/11/21. 23.06.12.
11.22.31.SYSTEM S.  EQ35, CYBER, 60 RECOVERED.
11.22.31.SYSTEM S.  EQ27   3 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S.  EQ27, NUCC, 57 RECOVERED.
11.22.31.SYSTEM S.  EQ26   1 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S.  EQ26, NUCC, 56 RECOVERED.
11.22.31.SYSTEM S.  EQ25   2 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S.  EQ25, NUCC, 55 RECOVERED.
11.22.31.SYSTEM S.  EQ24  58 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S.  EQ24, NUCC, 54 RECOVERED.
11.22.31.SYSTEM S.  EQ23   2 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S.  EQ23, NUCC, 53 RECOVERED.
11.22.32.SYSTEM S.  EQ22   7 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S.  EQ22   1 QUEUED FILES RECOVERED.
11.22.32.SYSTEM S.  EQ22, NUCC, 52 RECOVERED.
11.22.32.SYSTEM S.  EQ21  13 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S.  EQ21, NUCC, 51 RECOVERED.
11.22.32.SYSTEM S.  EQ20   6 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S.  EQ20, NUCC, 50 RECOVERED.
11.22.32.SYSTEM S.  EQ11, ECS RECOVERED.
11.22.32.CMSA   S.  FAMILY.
11.22.32.CMSA   S.   DEFAULT FAMILY USED.
11.22.32.CMSA   S.  ISF.
11.22.32.CMSA   S.   NOTEFIL NOT FOUND.
11.22.32.CMSA   S.   PORT NOT FOUND.
11.22.32.CMSA   S.   PROFILB NOT FOUND.
11.22.32.CMSA   S.   CALLPRG NOT FOUND.
```

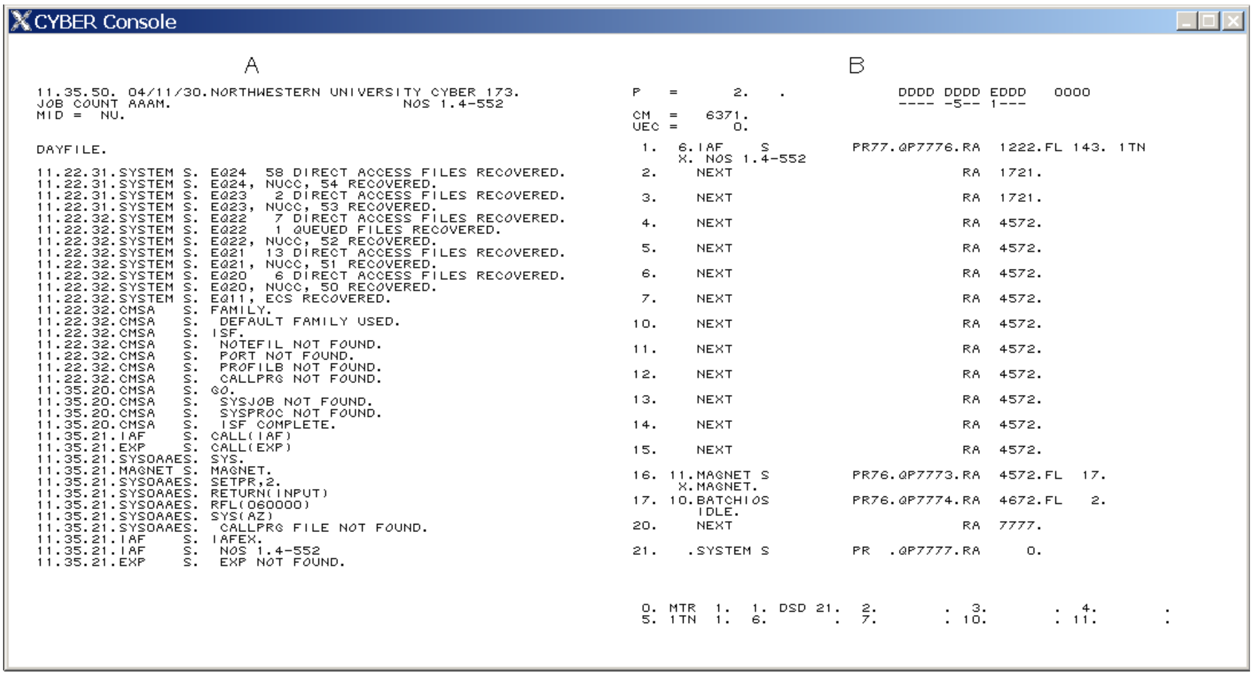
```
                                B

P    =      2.    .             DDDD DDDD EDDD   0000
                                ---- ---- 1---
CM   =   6530.
UEC  =      0.
 1.   5.CMSA   S SYSTEMX PR76.QP7777.RA  1222.FL  25.
      X. CHECK POSSIBLE DATE ENTRY ERROR. N.GO.
 2.     NEXT                           RA  4572.

 3.     NEXT                           RA  4572.

 4.     NEXT                           RA  4572.

 5.     NEXT                           RA  4572.

 6.     NEXT                           RA  4572.

 7.     NEXT                           RA  4572.

10.     NEXT                           RA  4572.

11.     NEXT                           RA  4572.

12.     NEXT                           RA  4572.

13.     NEXT                           RA  4572.

14.     NEXT                           RA  4572.

15.     NEXT                           RA  4572.

16.     NEXT                           RA  4572.

17.     NEXT                           RA  4572.

20.     NEXT                           RA  7777.

21.    .SYSTEM S         PR  .QP7777.RA     0.


 0. MTR  1.  1. DSD 21.  2.        .  3.        .  4.       :
 5.        .  6.        .  7.        : 10.        : 11.       :
```

9. The rest of the system will now come up, with the timesharing system (IAF), magnetic tape management system (MAGNET), and printer/card reader/card punch manager (BATCHIO) checking in.

NOTE

If you are an experienced NOS user, you may have noticed that the NAM (Network Access Method) subsystem is not running. At present, Desktop Cyber does not support the necessary hardware emulation (255x NPU, CDCNET device interfaces) which NAM requires. To accommodate this limitation, IAF has been customised so that it uses the 6676 multiplexer support code from the older TELEX subsystem. See ***[Interactive Use Xref]*** for differences between how one uses NAM IAF and the IAF running on Desktop Cyber with this release of NOS 1.4.

FIGURE 2-8. Display Following Successful Dead Start



```
                    A

11.35.50. 04/11/30.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AAAM.                           NOS 1.4-552
MID =  NU.


DAYFILE.

11.22.31.SYSTEM S. EQ24  58 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S. EQ24, NUCC, 54 RECOVERED.
11.22.31.SYSTEM S. EQ23   2 DIRECT ACCESS FILES RECOVERED.
11.22.31.SYSTEM S. EQ23, NUCC, 53 RECOVERED.
11.22.32.SYSTEM S. EQ22   7 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S. EQ22   1 QUEUED FILES RECOVERED.
11.22.32.SYSTEM S. EQ22, NUCC, 52 RECOVERED.
11.22.32.SYSTEM S. EQ21  13 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S. EQ21, NUCC, 51 RECOVERED.
11.22.32.SYSTEM S. EQ20   6 DIRECT ACCESS FILES RECOVERED.
11.22.32.SYSTEM S. EQ20, NUCC, 50 RECOVERED.
11.22.32.SYSTEM S. EQ11, ECS RECOVERED.
11.22.32.CMSA   S. FAMILY.
11.22.32.CMSA   S.  DEFAULT FAMILY USED.
11.22.32.CMSA   S. ISF.
11.22.32.CMSA   S.  NOTEFIL NOT FOUND.
11.22.32.CMSA   S.  PORT NOT FOUND.
11.22.32.CMSA   S.  PROFILB NOT FOUND.
11.22.32.CMSA   S.  CALLPRG NOT FOUND.
11.35.20.CMSA   S. GO.
11.35.20.CMSA   S.  SYSJOB NOT FOUND.
11.35.20.CMSA   S.  SYSPROC NOT FOUND.
11.35.20.CMSA   S.  ISF COMPLETE.
11.35.21.IAF    S. CALL(IAF)
11.35.21.EXP    S. CALL(EXP)
11.35.21.SYSOAAES. SYS.
11.35.21.MAGNET S. MAGNET.
11.35.21.SYSOAAES. SETPR,2.
11.35.21.SYSOAAES. RETURN(INPUT)
11.35.21.SYSOAAES. RFL(060000)
11.35.21.SYSOAAES. SYS(AZ)
11.35.21.SYSOAAES.  CALLPRG FILE NOT FOUND.
11.35.21.IAF    S. IAFEX.
11.35.21.IAF    S.  NOS 1.4-552
11.35.21.EXP    S.  EXP NOT FOUND.
```

```
                    B

P   =      2.   .             DDDD DDDD EDDD   0000
                              ---- -5-- 1---
CM  =   6371.
UEC =      0.

 1.  6.IAF    S       PR77.QP7776.RA  1222.FL 143.  1TN
     X. NOS 1.4-552
 2.    NEXT                       RA  1721.

 3.    NEXT                       RA  1721.

 4.    NEXT                       RA  4572.

 5.    NEXT                       RA  4572.

 6.    NEXT                       RA  4572.

 7.    NEXT                       RA  4572.

10.    NEXT                       RA  4572.

11.    NEXT                       RA  4572.

12.    NEXT                       RA  4572.

13.    NEXT                       RA  4572.

14.    NEXT                       RA  4572.

15.    NEXT                       RA  4572.

16. 11.MAGNET S       PR76.QP7773.RA  4572.FL  17.
     X.MAGNET.
17. 10.BATCHIOS       PR76.QP7774.RA  4672.FL   2.
       IDLE.
20.    NEXT                       RA  7777.

21.   .SYSTEM S       PR  .QP7777.RA     0.



 0. MTR  1.  1. DSD 21.  2.        .  3.        .  4.       :
 5. 1TN  1.  6.        :  7.        : 10.        : 11.       :
```

The system is now completely up and ready for interactive sessions and batch jobs.

SHUTDOWN

To shut down Desktop Cyber:

1. If users are logged on, you may wish to warn them. See the WARN command on page 2-17 and the DIAL command on page 2-14.

   If you wish to idle down the timesharing system by allowing existing users to complete their work without accepting new logins, you may prevent new logins by issuing the following command:

   ```
   SERVICE,TX,NJ0.
   ```

   This command sets the maximum number of jobs for timesharing at zero, but allows those currently running to finish.

   Instead of a FAMILY prompt, connecting users will receive the message SYSTEM CLOSED followed by your WARN message.

2. If the console keyboard is locked (see Figure 2-9 on page 2-12), issue the UNLOCK command.

NOTE

Some experienced analysts suggest stopping IAF with `1.OFFSW1.` and `1.IDLE.` before proceeding.

3. Issue the CHECK POINT SYSTEM command.

4. Issue the command STEP.

5. Type ALT + O and issue the Desktop Cyber command SHUTDOWN. Desktop Cyber will exit.

CONSOLE OPERATION

Most operator functions at the console are done through a PP program called DSD. DSD accepts commands from the keyboard and updates the various displays on the console.

DSD is interpretive—that is, when the operator is typing a command, DSD completes it as soon as it recognizes enough characters to establish the uniqueness of the command. As soon as the operator has entered a complete command, the echoed characters on the console “twinkle” to indicate that DSD is ready to process the command. DSD will not permit the entry of syntactically incorrect commands. If an entered command is syntactically correct but operationally impossible (such as trying to drop a job at an inactive control point), DSD responds with the message ***NO***. over the command line.

Except as noted, all DSD commands must end with a period.

DSD DISPLAYS

DSD provides data to the operator through several displays, rather than by printing reports in response to commands as is done with most command-line systems. Most of the displays are dynamic—as the data being reported on changes, the display is updated automatically.

The console shows two displays at once—one on each side. Typically, the operator keeps the A display on the left and B display on the right, since these are the two most general-purpose displays, but the operator will often bring up the other common displays—E,A; E,T; E,P; K; Q; and T—on the left side as needed, and usually returning to the A display when done.

The DSD displays may be selected two ways:

- With the command `xy.` where *x* is the left display and *y* is the right display. For example, to bring up the T display on the left side and the B display on the right, the operator would enter `TB.`
- With the command `x` where *x* is the left display. Note there is no period in the command. The right display will remain unchanged.

Some displays (A, C, D, E, F, G, K, and L) accept parameters.

- The A display accepts two types of parameters: a control point number, which returns the dayfile for the job at that control point; or a specific type of dayfile (A,ERROR LOG; A,ACCOUNT FILE; A,. See details for the A display on page 2-9).
- The E display accepts a parameter specified by a comma followed by the type of equipment display desired (E,A.; E,C.; E,M.; E,P.; E,T.). See details for the E display on page 2-9.
- The K and L displays accepts a parameter of the control point you wish to activate the display on. The control point is specified by `K,n.` where *n* is the control point number.
- The C, D, F, G, and M displays accept parameters following the letter and before and after a comma.

One doesn’t need to memorize what each of these displays does; the Z display contains a brief index of what the other displays do.

Here is a summary of the operator console displays:

TABLE 2-1. DSD Displays

Display	Purpose
A	Displays system dayfile, account file, and error log. `A,0.` Shows the system dayfile. `A,.` Same as A,0. `A,`*`n`*`.` Shows the dayfile for the job at control point *n*. `A,ACCOUNT FILE.` Shows the account dayfile. `A,ERROR LOG.` Shows the system error log.
B	Displays control point activity.
C, D	Displays central memory contents. Each display shows 40B (the B means octal, remember?) 60-bit words at a time; each word is shown in five groups of 12 bits along with the character equivalent of the word (if available). As there are two displays, you may display 100B locations of central memory by bringing up the C and D displays simultaneously with the command `CD.` By default, the C display begins at location 0B and D begins at 40B; each display, however, can begin at any desired location. The `+` and `-` keys page through the memory contents shown on the left display, and the `(` and `)` keys page through the contents on the right display. The C and D displays may also be used to show central memory relative to a control point. In the discussion below, C may be any of the memory displays (C, D, F, G, or M): `C,0.` Returns to the absolute address display. `C,.` Same as C,0. `C,`*`n`*`.` Shows central memory (or, for the M display, ECS) relative to control point *n*. A variety of other display options are available for these displays for showing memory at arbitrary locations; see *NOS Version 1 Operator's Guide* for details.
E	Equipment status tables. The E display is actually a group of five separate displays: `E,A.` Equipment status table. `E,C.` Mass storage configuration. `E,M.` Mass storage table. `E,P.` Pending tape requests. `E,T.` Tape drive status.

Display	Purpose
F, G	Displays central memory contents. Just like the corresponding C and D displays except each word is shown in four groups of 15 bits.
H	File name table (FNT). Lists the name, FNT ordinal, and type of each file currently in use by the system. The most common file types (by far) are— CM Central memory resident (common) file. FA Fast-attach file (a direct access permanent file with the disk address kept in central memory). IN Input file. LI Library file (locked central memory file). LO Local file. PM Direct access permanent file. PR Print file. PT Primary terminal file (a local file which has been designated as such by an interactive user for ease of use). PH Punch file. RO Rollout file. TE Timed/event rollout file. Consult the *NOS Version 1 Systems Programmer's Instant*, page 1-2 (included in your distribution) for a full listing of the file types. The H display may be limited to just the files at a specified control point by specifying `H,`*`n`*`.` where *n* is the control point number. `H,.` returns to the full display of all files in use across the system.
I	BATCHIO status. Shows the status of unit record equipment (card readers, punches, and printers).
J	Control point status. Shows the status of a specified control point. You specify the control point with `J,`*`n`*`.` where *n* is the control point number.

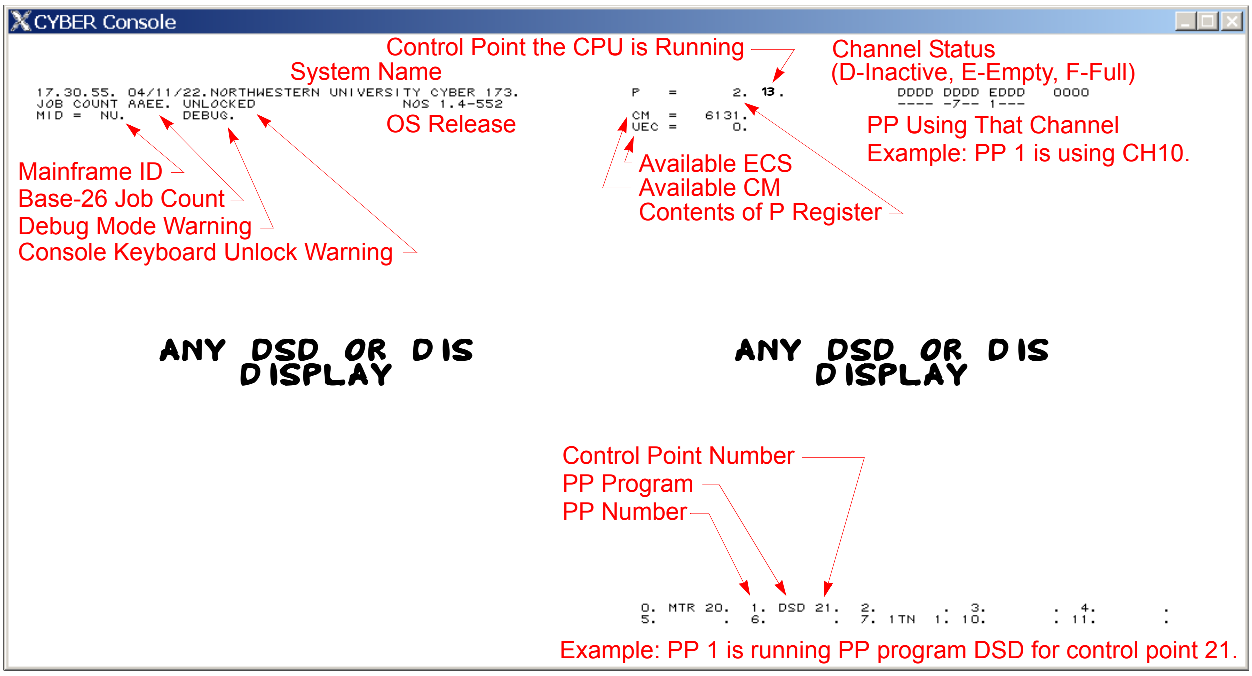
K, L	Programmable displays which a program may use to interact with the operator. A program requests operator assignment of the K or L displays by placing a message on the B display, for example:
	`20. S. MODVAGUS PR70.QP7761.RA 7537 FL 240.`
	`    X. REQUEST *K* DISPLAY.`
	The operator assigns the K display with `K,`*`n`*`.` where *n* is the control point number (20, in this case) as in `K,20.` The operator may now type in data for that program by preceding it with `K.` as in `K.C,JPD` (which is the MODVAL K display command to create a new user JPD; see *Account Management* on page 2-25).
	You may use the L display for the same thing as the K display. Anything that requests the K display may be assigned the L display. Its purpose is just a matter of convenience so that you may have two programs utilizing a programmable display up at the same time without having to tediously switch back and forth (as between `K,20.` and `K,17.`).

M	ECS display. Shows contents of extended core storage in the same manner as the C display shows for central memory.
N	File display. You may use this display to show the contents of any active file. Use the `DISPLAY,`*`fntord`*`.` command (where *fntord* is the FNT ordinal of the file from the H display) to show the specified file on the N display.
O	Transaction facility status. The transaction facility is not included in this distribution, so this display, while supported by NOS, isn't supported in this particular distribution.

Display	Purpose
P	PP communications area. For each peripheral processor, the following is listed: first line, input register, which contains the three-character name of the PP program being run; second line, output register, containing the four-character monitor request being processed; third line, first word of message buffer. The list is grouped 10 PPs per page; use the + and - keys to page through the list. A discussion of monitor requests begins on page 4-1 of the *NOS Version 1 Systems Programmer's Instant,* pub. 60449200.
Q	Queue status. Shows the status of input, output, and rollout queues, along with their corresponding FNT ordinals.
R	Export/Import status. Export/Import is the predecessor to RBF. It's not included in this distribution, so this display isn't supported.
S	System control information. Displays the priorities and scheduling parameters for the different queue types for the different job origin types.
T	Timesharing system status. Shows users currently logged in along with their terminal numbers and last operations.
W	MULTI status. Not installed.
Y	Monitor functions. Lists the several monitor mnemonics and codes.
Z	Directory. Gives an abbreviated listing of the various console displays (in other words, it's similar to the table you are reading).

The console also displays some system status at the top and bottom of the screen no matter what displays are selected, as shown in the diagram below:

FIGURE 2-9. Status on all DSD and DIS Displays



```
17.30.55. 04/11/22.NORTHWESTERN UNIVERSITY CYBER 173.          P   =     2. 13.        DDDD DDDD EDDD   0000
JOB COUNT AAEE. UNLOCKED                  NOS 1.4-552                                  ---- -7-- 1---
MID =  NU.      DEBUG.                                         CM  =  6131.
                                                               UEC =    0.

              ANY DSD OR DIS                                          ANY DSD OR DIS
                  DISPLAY                                                 DISPLAY

                                                               0. MTR 20.  1. DSD 21.  2.         .  3.        .  4.      :
                                                               5.        .  6.       .  7. 1TN  1. 10.         : 11.      :
```

DSD Special Keys

The following keys have special significance when entered as the first character on a line. These characters do not require a terminating ENTER to take effect.

TABLE 2-2. DSD Special Keys

Key	Significance
*	Temporarily releases the display console in favor of another program such as DIS, O26, BAT, SNK, etc.
=	Toggles memory references between absolute and relative.
+	If the C, D, F, G, M, T, or U displays are active on the left display, advances the display by 40B words.
-	If the C, D, F, G, M, T, or U displays are active on the left display, decrements the display by 40B words.
(	Advances the right display in the same manner that + does for the left.
)	Decrements the right display in the same manner than - does for the left.
/	Advances the left screen C, D, F, G, M, T, or U displays by the amount in the rightmost 18 bits (six octal digits) of the first word displayed.
ENTER	Sets repeat entry. The message REPEAT. appears above the command line. Subsequent commands are executed, but the command is not cleared from the command line so that the command may be easily edited and issued again by pressing ENTER.
LEFT BLANK	Clears the entered command and, if active, terminates repeat entry.

DSD COMMANDS

Many operator commands act on individual jobs executing at a control point. These commands are preceded by *n.* indicating that the operator needs to enter the control point number and a dot prior to entering the command. For example, `6.DROP.` will drop the job running at control point 6.

*

`n.*comment`

Places the specified *comment* in the dayfile of the job at control point *n.* Does the same thing as the *n.*COMMENT command. Although the *comment* does not need to terminate with a period, but conventionally *comment* begins with a single leading space and terminates with a period.

99

`99.`

Enables/disables automatic loading of DSD command syntax tables from disk. See the discussion on UNSTEP (page 2-17).

ASSIGN

`n.ASSIGN,dd.`

Assigns device *dd* to the job. The job must be requesting a device via either the REQUEST command or the REQUEST local file manager function. Typically, this will be a request to assign the operator display console to some program which has temporarily relinquished it (such as DIS or O26), or to request a tape mount. (This command is not used to satisfy requests for tape mounts made by the LABEL control statement.) For more information on tape requests made via REQUEST, see *How to Handle a Tape Request on the B Display* on page 2-42.

AUTO

`AUTO.`

Instructs the system to start the services which have been previously enabled (see the ENABLE command on page 2-15). This default system installation parameter deck causes this command to be executed automatically at dead start. AUTO is also used to resume job scheduling following an IDLE command (see the IDLE command on page 2-15).

BLITZ

`BLITZ.`

Drops every job running on the system except subsystems. Use `AUTO.` to start subsystems back up. The console must be unlocked. You don't need this dangerous command very often—it was dropped from NOS 2—but, as your author recalled one VIM member saying, "when you need it, you need it!"

CFO

`n.CFO.dd..d`

Places command string *dd..d* (no period at the end of *dd..d* unless the program is expecting one) starting in word 70B of the control point's field length as a command from the operator. The job running at the control point must be paused, expecting and know what to do with this data. Usually used for obtaining brief commands from operator (hence the name) without having to go through the K display.

CHECK POINT SYSTEM.

`CHECK POINT SYSTEM.`

Prepares the system for immediate shutdown. The system can be restored to its state before running this command by performing a level 1 dead start instead of a level 0 dead start; see Appendix A for details.

COMMENT

`n.COMMENT.comment`

Places *comment* in the job's dayfile. *n.*comment* is a shorthand way of doing the same thing. Although the *comment* does not need to terminate with a period, but conventionally *comment* begins with a single leading space and terminates with a period.

DATE

`DATE.yy/mm/dd.`

Changes the date without having to dead start. The console must be unlocked (see the UNLOCK command on page 2-17).

DEBUG

`DEBUG.`

Places the console in or out of debug mode. The word DEBUG appears at the top of the console over the left display to indicate when the system is in debug mode. The console must be unlocked to issue this command. In debug mode, users with system origin privileges can issue many system control commands from interactive sessions in addition to the system console.

DIAL

`DIAL,t,message`

Sends *message* to interactive terminal number *t*.

See also: Use the T display to determine which terminal a user is on.

DIS

`n.DIS.`

Takes the job at the control point into DIS. See *DIS Operation* on page 2-18.

NOTE

Take care not to type accidentally a comma in lieu of a period after *n*. If the console were unlocked and you typed `n,DIS.`, you would be inserting the display coded characters *IS.* at memory location *n*. See the *NOS Version 1 Operator's Guide*, pub. 60435600, page 3-27, for a discussion of direct memory entry.

`X.DIS.`

Starts a new system origin job at any available control point and starts DIS.

DISABLE

`DISABLE,feature.`

Stops a subsystem or disables an option. See ENABLE.

DROP

`n.DROP.`

Drops the job, but allows EXIT error processing. See the discussion for the OVERRIDE command (page 2-16) for the proper treatment on how to terminate jobs—use DROP before KILL, and KILL before OVERRIDE.

NOTE

Take care not to type accidentally a comma in lieu of a period after *n*. If the console were unlocked and you typed `n,DROP.`, you would be inserting the display coded characters *ROP.* at memory location *n*. See the *NOS Version 1 Operator's Guide*, pub. 60435600, page 3-27, for a discussion of direct memory entry.

ENABLE

`ENABLE,`*`feature`*`.`

Allows a subsystem to start or enables an option. Some options are:

BATCHIO Allows automatic startup of print, punch, and card reader system.

MAGNET Allows automatic startup of magnetic tape management system.

ROTARY CHECK

This feature remains from the MECC system and should always be disabled. (The default system installation parameter deck disables rotary check by default.)

SECONDARY USER CARDS

Allows users to issue additional USER cards once they've validated themselves initially to the system (akin to UNIX *su*).

TELEX Allows automatic startup of interactive access system.

FORM

`FORM`*`nn`*`,`*`fc`*`.`

Tells the system that the special forms indicated by *fc* are now mounted on printer with EST ordinal *nn*. Use `E,A.` to see a list of equipment and associated EST ordinals.

`FORM`*`nn`*`,.`

Clears the forms code on printer with EST ordinal *nn.*

GO

`n``.GO.`

Resumes execution of a paused job.

IDLE

`n``.IDLE.`

Asks the subsystem running at that control point to finish what it's doing and exit. See the discussion for the OVERRIDE command (page 2-16) for the proper treatment on how to terminate jobs.

`IDLE.`

Terminates job scheduling. Jobs which are at control points are permitted to remain until they either complete or are rolled out. To resume job scheduling, use the AUTO command (page 2-13).

KILL

`n``.KILL.`

Drops the job without EXIT error processing. See the discussion for the OVERRIDE command (page 2-16) for the proper treatment on how to terminate jobs.

LOCK

`LOCK.`

Locks the console to prevent accidentally entering dangerous commands.

LOGOFF

`LOGOFF,`*`t.`*

Logs off the interactive terminal number *t*.

See also: The T display shows the terminal number for each user.

MESSAGE

`MESSAGE,`*`message.`*

Changes the login banner to *message*. Users will see this message *before* they log in, and users already logged in will not see it at all. Although *message* need not contain a terminating period, the operator conventionally provides one.

See also: WARN to display a message *after* the user has logged in.

OFF

`OFF`*`nn`*`.`

Used to logically turn off equipment with EST ordinal *nn*.

OFFSW

`n``.OFFSW`*`x`*`.`

Turns off logical sense switch *x* ($1 \le x \le 6$) for reference by the job.

ON

`ON`*`nn`*`.`

Used to logically turn on equipment with EST ordinal *nn*.

ONSW

`n``.ONSW`*`x`*`.`

Turns on sense switch *x* ($1 \le x \le 6$) for reference by the job.

OVERRIDE

`n``.OVERRIDE.`

Removes the job from the executing job table. Should be used only as a last resort; use the DROP command first (page 2-14), which permits the job to do error processing before exiting. If the DROP were ineffective, use the KILL command (page 2-15), which does not permit error processing. Only if the KILL fails should you resort to OVERRIDE. Cannot be used to drop subsystems; see IDLE command (page 2-15).The console must be unlocked; see the UNLOCK command.

PURGE

`PURGE,`*`nnn`*`.`

Purges job with FNT ordinal *nnn* from the system. Often used to delete a massive print job before it starts, or to drop a job which isn't running at a control point (awaiting, for example, a tape mount request).

See also: DROP and KILL, which may be used to terminate jobs running at a control point; Q display to see the list of jobs and their FNT numbers.

PURGEALL

`PURGEALL,`*`t`*`.`

Purges all jobs of type *t* from the system.

t	Type
I	Input
O	Output
P	Punch
R	Rollout
T	Timed/Event Rollout

ROLLIN

`ROLLIN,`*`xxx`*`.`

Assigns job with FNT ordinal *xxx* maximum queue priority, making it immediately eligible for being scheduled to a control point.

See also: You may determine the FNT ordinal of a job by looking at the Q display.

ROLLOUT

`n``.ROLLOUT.`

Removes job currently assigned to control point *n* and places it in the rollout queue. The job will remain rolled out until explicitly rolled back in.

`n``.ROLLOUT,`*`xxxx`*`.`

Removes job currently assigned to control point *n* and places it in the rollout queue for *xxxx* seconds, at the end of which it is eligible for rescheduling to a control point.

STEP

`n``.STEP.`

Places the monitor in step mode for the job at that control point. Setting the monitor in step mode stops all central memory I/O operations and prevents the system from processing PP requests when the next monitor function is encountered. Pressing the space bar releases the present step and stops again for each subsequent monitor function. The console must be unlocked to issue this command.

`STEP.`

Places the monitor in single-step mode. The console must be unlocked to issue this command. After issuing this command, you may get the message `DISK BUSY` when you try to execute some commands; see the discussion of the UNSTEP command on page 2-17.

TIME

`TIME.`*`hh.mm.ss.`*

Changes the time without having to dead start. The console must be unlocked (see the UNLOCK command on page 2-17).

NOTE

NOS does not change the time automatically when going between standard and daylight savings/summer time.

UNLOAD

`UNLOAD,`*`xx.`*

Unloads the tape on unit *xx*. The tape must not be assigned to a job.

UNLOCK

`UNLOCK.`

Unlocks the console to permit the entry of potentially dangerous commands.

UNSTEP

`UNSTEP.`

Takes the monitor out of single-step mode.

NOTE

Did you accidentally STEP the system and now find that when you try to UNSTEP it, you get the message `DISK BUSY`? That's because DSD is trying to read DSD command syntax tables from off of disk, but the monitor has been stepped, which prevents the disk from being read. The solution is to use the 99 command, which tells DSD not to load syntax tables off of disk. Doing so, however, prevents you from using any of the DSD commands which are not built in to the resident part of DSD. Fortunately, UNSTEP is in the resident part. To take the system out of STEP mode, first issue the command `99.` Then issue the `UNSTEP.` command, and finally issue `99.` again to reenable the loading of syntax tables.

VSN

`VSN,`*`xx`*`,`*`vvvvvv.`*

Informs system that the operator has mounted unlabeled tape with VSN *vvvvvv* on unit *xx*. See *How to Handle a Tape Request on the E,P Display* on page 2-39. (This command is only needed when mounting unlabeled tapes; labeled tapes have the VSN encoded at the beginning of the tape.)

WARN

`WARN,`*`message`*`.`

Displays *message* to IAF users as they log in. For users already logged in, the message is displayed when they exit their next command. Although *message* need not contain a terminating period, the operator conventionally provides one.

`WARN,.`

Clears warning message.

See also: MESSAGE, which displays a message *before* the user logs in.

X

`X.`*`cmd`*`(`*`opts`*`)`
: Starts a system origin job to run the specified command *cmd* with the specified options *opts*. No terminating period is entered at the end of the command line.

`X.`*`cmd.`*
: Starts a system origin job to run the specified command *cmd*.

***n*=**

`n=cmd``(`*`opts`*`)`
: Starts a system origin job to run the specified command *cmd* with the specified options *opts* at control point *n*, if possible. No terminating period is entered at the end of the command line.

`n=cmd.`
: Starts a system origin job to run the specified command *cmd* at control point *n*, if possible.

DIS OPERATION

DIS is a job display package which:

- Allows the operator to interact with job control so that control statements may be entered conveniently without having to log in to a terminal or punching up a job deck. Jobs initiated through DIS are system origin (see *Job Origin Types* on page 2-37) and have privileges that non-system origin jobs do not have. For example, logging in with a user number and password is not required. DIS is started this way by the console command `X.DIS.`

- Allows the operator to interact with an already-executing job on-the-fly. This can be done to investigate what the job is doing, what files it has open, see what job steps are next, and even change what that job does. It is a powerful debugging tool in its own right. DIS is started in this way by the console command *`n`*`.DIS.` where *n* is the control point number of the executing job. A job of system origin or with system origin privileges (see the CSOJ permission on page 2-33) may bring itself into DIS by issuing the `DIS.` control statement.

NOTE

While DIS is certainly interactive, it isn't like a terminal where you type a command and view output on the screen. Job input is read from the mass storage file INPUT, and job output is written to mass storage file OUTPUT. Thus, to generate input or read output on the display, you must use the O26 editor and WRITE.INPUT. or READ.OUTPUT. accordingly.

Auto Mode

No matter how DIS is started, execution of the job is temporarily suspended at the end of each job step (that is, command) unless DIS is placed in *auto mode*. Auto mode causes DIS to read control statements and continue execution until either:

- DIS reaches the end of the job, whereupon job execution is again suspended and the operator may enter additional commands or directives.

- DIS encounters an error which would have aborted the job had it not been under DIS control.

- DIS is manually taken out of auto mode, whereupon execution is suspended at the conclusion of the current job step.

DIS is placed into auto mode by pressing the period key; it is taken out of auto mode manually by pressing the left blank key.

DIS Displays

Like DSD, DIS has several displays you may select, and you may change them at any time, even while the job is executing. You specify the displays the same way you do with DSD: `xy.` The left display, *x*, may be any of the displays listed below; the right display, *y*, must be one of B, C, D, N, T, or U.

TABLE 2-3. **DIS Displays**

Display	Purpose
A	Dayfile messages and files in use.
B	Job status, control statements, and register contents.
C, D	Central Memory displays in five groups of four octal digits and the display code translation.
F, G	Central Memory displays in four groups of five octal digits and the display code translation.
H	File name table entries.
M	ECS memory displays in five groups of four octal digits with display code translation.
N	Blank screen. Useful for the right display when the left display creeps data over to the right as in the T display.
T, U	Displays 240 (T) or 300 (U) words from central memory as coded lines of text.
V	Displays 512 words direct from central memory without interpretation.
Y	Displays function mnemonics and values.
Z	Display directory.

DIS Special Keys

The following keys have special significance when entered as the first character on a line. These characters do not require a terminating ENTER to take effect.

TABLE 2-4. **DIS Special Keys**

Key	Significance
*	Temporarily releases DIS control of the display console back to DSD. From DSD, press * to return to DIS.
=	Toggles memory references between absolute and relative.
+	If the C, D, F, G, M, T, or U displays are active, advances the display by 40B words.
-	If the C, D, F, G, M, T, or U displays are active, decrements the display by 40B words.
/	In DIS, advances the left screen C, D, F, G, M, T, or U displays by the amount in the rightmost 18 bits (six octal digits) of the first word displayed.
.	Causes DIS to automatically read control statements from the control statement buffer. The message AUTO MODE. appears above the command line.
ENTER	Sets repeat entry. The message REPEAT ENTRY. appears above the command line. Subsequent commands are executed, but the command is not cleared from the command line so that the command may be easily edited and issued again by pressing ENTER.
LEFT BLANK	Clears the entered command and, if active, terminates repeat entry.

DIS Commands

NOTE

Unlike DSD, DIS is not interpretive—DIS does not fill in parts of a command as it recognizes it. You must terminate each command with ENTER.

All DIS commands must contain a terminator (either a period or right parenthesis).

DIS is used primarily to enter control statements. In addition to control statements (see *[Batch Commands Xref]*), the following commands may also be used:

DCP

```
DCP.
```

Drops the central processor (that is, suspends execution of the program until either the command RCP. is entered (which causes resumption where the program left off) or a new command is entered, which causes the suspended program to stop before the new command is executed.

DROP

```
DROP.
```

Drops DIS. Does not drop the job if there are control statements remaining in the control statement buffer.

ELS

`ELS.`*`command.`*

Enter last statement. Enters control statement *command* in the control statement buffer after the last statement. This allows you to add on to an existing job.

ENA

`ENA`*`i,xxxxxx.`*

Sets register A*i* to *xxxxxx.*

ENB

`ENB`*`i,xxxxxx.`*

Sets register B*i* to *xxxxxx.*

ENFL

`ENFL,`*`xxxxxx.`*

Sets the central memory field length (the amount of central memory currently assigned to your job) to *xxxxxx*.

ENP

`ENP,`*`xxxxxx.`*

Sets program counter P to *xxxxxx.*

ENS

`ENS,`*`command.`*

Enter next statement. Clears the existing control statement buffer and enters *command* as the next statement.

ENX

`ENX`*`i,xxxxx xxxxx xxxxx xxxxx.`*

Sets register X*i* to *xxxxx xxxxx xxxxx.*

`ENX`*`i`*`,L`*`value.`*

Sets register X*i* left justified to *value.*

`ENX`*`i,b,xxxx`*`.`

Sets byte *b* of register X*i* to *xxxx*. Each word consists of five 12-bit bytes (four octal digits) numbers 0 through 4, left to right.

HOLD

```
HOLD.
```

Holds the job at its current status but drops the display console channel.

KIL

```
KIL.
```

Kills DIS by calling the KIL PP program. You might want to kill DIS when you have output files (OUTPUT, PUNCH, PUNCHB) which would otherwise be released to the appropriate output queue at job termination; using KIL prevents this release.

NOTE

KIL is not actually an internal DIS command; it's a PP program which is callable from system origin jobs.

O26

`O26.`

(Oh-two-six.) Calls the O26 full screen text editor. See *Using O26* on page 2-22.

RCP

`RCP.`

Request central processor. Used to resume execution following a *DCP* command.

RNS

`RNS.`

Read next statement. Reads and executes the next statement in the control statement buffer.

ROLLOUT

`ROLLOUT.`

Places the job in the rollout queue until the job scheduler rolls it back in. On Desktop Cyber, that's not likely to be very long.

`ROLLOUT,`*`xxxx.`*

Places the job in the rollout queue for approximately *xxxx* seconds.

SUI

`SUI,`*`index`*`.`

Sets the active user index to *index* without having to specify a user number and password.

SUN

`SUN,`*`usernumber.`*

Sets the active user number to *usernumber* without having to specify a password.

NOTE

To perform any permanent file operations, you must either specify a user index via SUI, a user number via SUN, or a user number and password via USER.

The SUI and SUN commands are actually control statements rather than internal DIS commands, but except for creating and editing system origin job decks, find little use outside of DIS.

Memory Entry

These commands permit the direct entry of data into memory. The console must be unlocked to enter data into an absolute address outside of the memory block assigned to the job running DIS. Memory addresses are specified as relative addresses unless the operator has used the = key to toggle relative/absolute address references (see the = entry in *DIS Special Keys* on page 2-19).

loc, val.	Changes contents of central memory word at address *loc* to *val.*
loc, b, val.	Changes contents of central memory address *loc*, byte *b*, to *val*. Each word consists of five 12-bit bytes (four octal digits), numbered 0 through 4, left to right.
loc,`D`*chars.*	Changes contents of central memory address *loc* to dis-

play code characters *chars*, left justified, zero filled.

USING O26

NOTE

Like DSD, but unlike DIS, O26 is interpretive; O26 fills in characters in a command as soon as you have entered enough of the command for it to be unique.

O26 is a full screen text editor which runs only on the operator's display console.[†] It gets its name from the punched card encoding popular at the time. Why name a text editor after a punch card encoding? Because the alternative to using O26 was to use a *real* O26 and punch in the text that way—interactive systems came much later.

O26 Special Keys

The following keys have special significance when entered as the first character on a line. These characters do not require a terminating ENTER to take effect.

TABLE 2-5. O26 Special Keys

Key	Significance
`0`	Sets insertion point at first line.

`1`
through
`7`
	Sets insertion point at $(n \times 4) + 1$.

`8`	Sets marker 8 at current line.
`9`	Sets marker 9 at current line.
`+`	Displays next page.
`-`	Displays previous page.
`*`	Holds display and returns control to DSD.
`/`	Toggles display scrolling.
`(`	Advances insertion point by one line.
`)`	Decrements insertion point by one line.
`=`	Clears insert flag.
`,`	Moves display to insertion point.
`.`	Deletes the line following the insertion point.
LEFT BLANK	Erases a partially-typed command.
ENTER	Sets repeat entry. The message REPEAT ENTRY. appears above the command line. Subsequent commands are executed, but the command is not cleared from the command line so that the command may be easily edited and issued again by pressing ENTER.
SPACE	Begins text insertion.

System Commands

DIS

`DIS.`

Writes the buffer, rewinds the output file, and returns control to DIS.

See also: XDIS, which permits returning to DIS without writing the file, and RDIS, which rewinds the output file, writes the buffer, rewinds the output file again, and returns control to DIS.

DROP

`DROP.`

Writes the buffer, rewinds the output file, and returns control to DSD.

See also: XDROP, which permits returning to DSD without writing the file.

[†] O26 is one of the first full screen text editors, and may actually be the first.

RDIS

`RDIS.`

Rewinds the output file, writes the buffer, rewinds the output file again, and returns control to DIS. This is the normal way of exiting O26.

XDIS

`XDIS.`

Transfers control back to DIS without writing or rewinding the file.

XDROP

`XDROP.`

Transfers control back to DSD without writing or rewinding the file.

File Commands

FILE

`FILE.`*`filename.`*

Sets the default file name to *filename.*

RC

`RC.`*`filename.`*

Reads MODIFY-compressed compile file *filename* (record type OPLC) into the O26 buffer.

READ

`READ.`*`filename.`*

Reads *filename* into the O26 buffer.

`READ.`

Reads the default file name into the O26 buffer or, if no default file has been specified by the FILE command, then it reads the most previously-specified file.

REWIND

`REWIND.`*`filename.`*

Rewinds *filename.*

`REWIND.`

Rewinds the default file or, if no default file has been specified, rewinds the most previously-specified file.

RI

`RI.`

Same as READ.INPUT.

`RI.`*`filename.`*

Same as READ.*filename.*

RNR

`RNR.`*`filename.`*

Clears the O26 buffer and reads the next record on *filename.*

`RNR.`

Clears the O26 buffer and reads the next record on the default file. If no such file has been specified, reads the next record on the most previously-specified file.

RO

`RO.`

Same as READ.OUTPUT.

`RO.`*`filename.`*

Same as READ.*filename*.

WRITE

`WRITE.`*`filename.`*

Writes *filename* from the O26 buffer.

`WRITE.`

Writes the default file or, if no default file has been specified, writes the most previously-specified file.

Editing Commands

COPY

`COPY.`

Copies lines of text between 8 and 9 markers and pastes at insertion point.

D

`D,*.`

Deletes text between 8 and 9 markers.

L

`L.`*`text`*

Replaces following line. Use for when you want to make small changes to the line. As soon as `L.` is entered, the line is listed on the command line. Use the space bar to position the pointer and enter the new text.

MOVE

`MOVE.`

Cuts lines of text between 8 and 9 markers and pastes at insertion point.

P

`P.`*`text`*

Inserts *text* beginning at insertion point. After entering the line, the **P.** prompt will remain, allowing the continued entry of text. When finished entering lines, press LEFT BLANK.

RS

`RS/`*`oldtext`*`/`*`newtext`*`/`

Replaces *oldtext* with *newtext* on following line. / may be any delimiting character.

Miscellaneous Command

UCC

`UCC=c.`

Sets the shift escape character to *c*. To enter a literal *c*, enter *cc*.

`UCC=.`

Clears the shift escape character.

Entering Characters Not on the Keyboard

You may use O26 to enter characters which aren't on the keyboard. Locate the character to be entered in table 2-6 on page 2-24 and enter the shift escape character followed by the listed character. For example, if the escape character is `/`, to enter a `$`, press `/` and then `S`. O26 will show the octal display code of the character. For example, $ will display on the input line as $\overset{S}{3}$.

TABLE 2-6. Character Escapes

To Enter:	Type the UCC Escape Character and Then:	Display Code
`$`	`S`	53
`#`	`0`	60
`[`	`1`	61
`]`	`2`	62
`%`	`3`	63
`"`	`4`	64
`_`	`5`	65
`!`	`6`	66
`&`	`7`	67
`'`	`Q`	70
`?`	`W`	71
`<`	`E`	72
`>`	`R`	73
`@`	`T`	74

To Enter:	Type the UCC Escape Character and Then:	Display Code
\	Y	75
~	U	76
;	,	77

To Create a Text File from DIS

1. Issue the command `O26.`
2. Type `P.` and the first line of text. After pressing ENTER, the **P.** will remain.
3. Continue entering text, pressing ENTER after each line.
4. When done entering text, press LEFT BLANK.
5. Enter the command `WRITE.`*`filename`*.
6. Enter the command `DIS.`

To Browse a File from DIS

1. Issue the command `O26.`
2. Once in O26, issue the command `READ.`*`filename.`*
3. You may use the `/` key to slowly scroll down the file, or you may use the `(` and `)` keys to move up and down a line at a time, or you may use `+` and `-` to move around a screenful at a time.
4. Issue the command `XDIS.` to return back to DIS.

ACCOUNT MANAGEMENT

Accounts are created, modified, and deleted through the validation file manager program MODVAL. MODVAL manages the system validation files VALIDUZ and VALINDZ.

USER INDICES

NOS keeps track of users through a user index, an integer ranging from 1 to 377777B. This index is used much like a user identifier (UID) in UNIX and related variants.

With certain exceptions noted below, jobs are identified by a two-part, seven-character job name: A four-character hash code followed by a three-character sequence number. The two parts are concatenated, with no typographical distinction separating them.

The hash code is a 17-bit, base-32 mapping of the user index utilizing the alphabet A to Z and 0 to 5. The sequence number is a monotonically-increasing base-26 quantity using the alphabet A to Z.

Exceptions:

- For system origin jobs (see *Job Origin Types* on page 2-37), the first four characters of the job name (left-justified, 0-filled) are used instead of the hash code. Thus, a system origin job started by `X.DIS.` will have a job name of DIS0 followed by a three-character sequence number, and a system origin job started by `X.MODVAL.` will have a job name of MODV followed by a three-character sequence number.
- For interactive origin jobs, the terminal number is used instead of the sequence number, expressed as a three-digit, right-justified, zero-filled octal number.

ACCOUNT CREATION

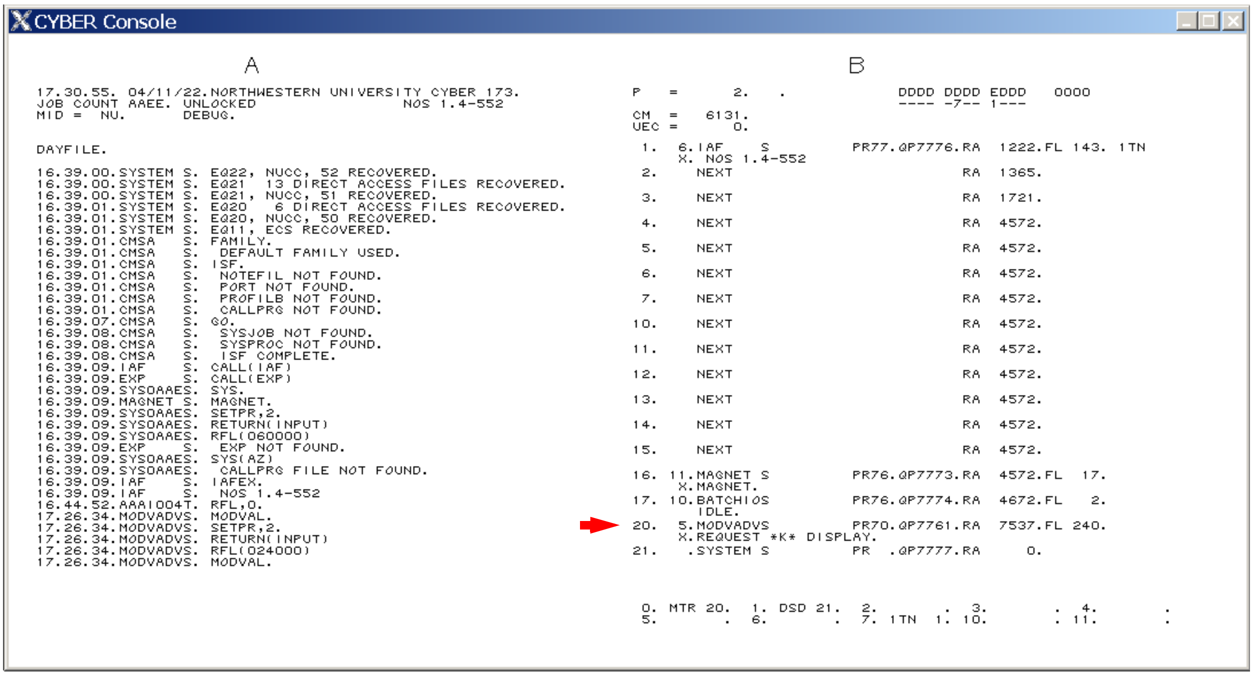
1. At the console, type `X.MODVAL.`

NOTE

Users will not be able to log in or out while you are in MODVAL, so don't dilly-dally.

2. Note the control point number requesting the K display. In the example below, MODVAL is running at control point 20.

FIGURE 2-10. Starting MODVAL



```
17.30.55. 04/11/22.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AAEE. UNLOCKED                   NOS 1.4-552
MID =  NU.       DEBUG.


DAYFILE.

16.39.00.SYSTEM S. EQ22, NUCC, 52 RECOVERED.
16.39.00.SYSTEM S. EQ21  13 DIRECT ACCESS FILES RECOVERED.
16.39.00.SYSTEM S. EQ21, NUCC, 51 RECOVERED.
16.39.01.SYSTEM S. EQ20   6 DIRECT ACCESS FILES RECOVERED.
16.39.01.SYSTEM S. EQ20, NUCC, 50 RECOVERED.
16.39.01.SYSTEM S. EQ11, ECS RECOVERED.
16.39.01.CMSA   S. FAMILY.
16.39.01.CMSA   S.  DEFAULT FAMILY USED.
16.39.01.CMSA   S. ISF.
16.39.01.CMSA   S.  NOTEFIL NOT FOUND.
16.39.01.CMSA   S.  PORT NOT FOUND.
16.39.01.CMSA   S.  PROFILB NOT FOUND.
16.39.01.CMSA   S.  CALLPRG NOT FOUND.
16.39.07.CMSA   S. GO.
16.39.08.CMSA   S.  SYSJOB NOT FOUND.
16.39.08.CMSA   S.  SYSPROC NOT FOUND.
16.39.08.CMSA   S.  ISF COMPLETE.
16.39.09.IAF    S. CALL(IAF)
16.39.09.EXP    S. CALL(EXP)
16.39.09.SYSOAAES. SYS.
16.39.09.MAGNET S. MAGNET.
16.39.09.SYSOAAES. SETPR,2.
16.39.09.SYSOAAES. RETURN(INPUT)
16.39.09.SYSOAAES. RFL(060000)
16.39.09.EXP    S.  EXP NOT FOUND.
16.39.09.SYSOAAES. SYS(AZ)
16.39.09.SYSOAAES.  CALLPRG FILE NOT FOUND.
16.39.09.IAF    S. IAFEX.
16.39.09.IAF    S.  NOS 1.4-552
16.44.52.AAAI004T. RFL,0.
17.26.34.MODVADVS. MODVAL.
17.26.34.MODVADVS. SETPR,2.
17.26.34.MODVADVS. RETURN(INPUT)
17.26.34.MODVADVS. RFL(024000)
17.26.34.MODVADVS. MODVAL.
```

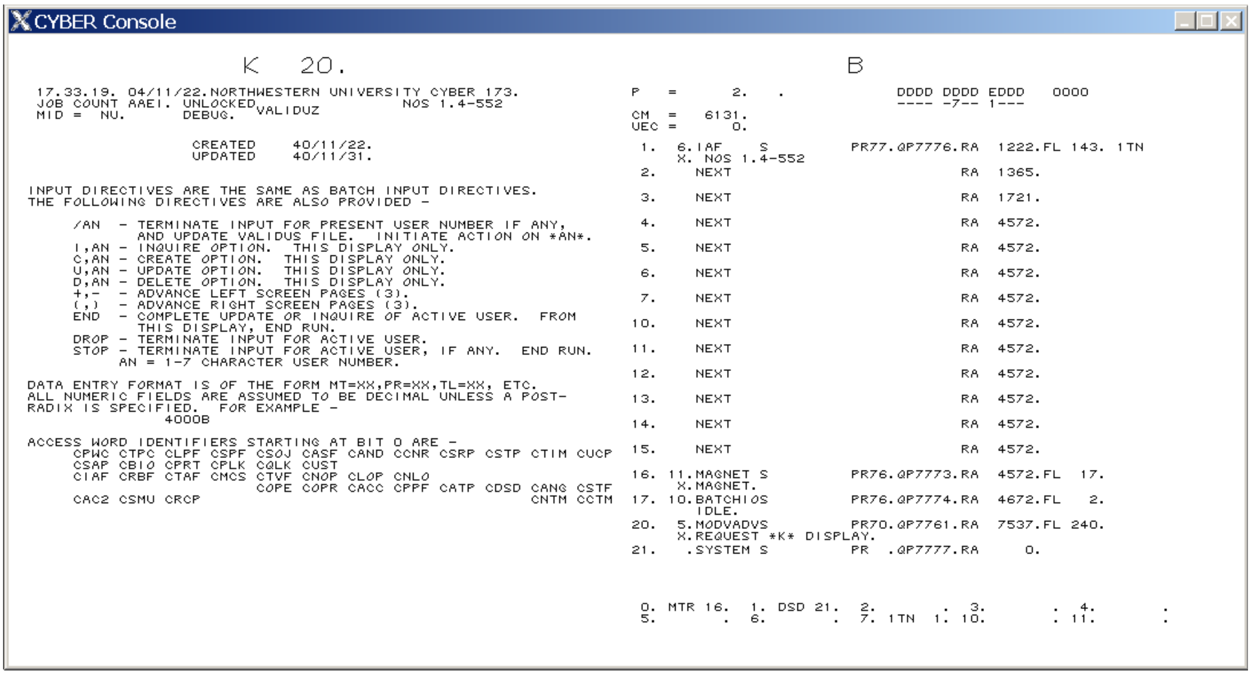
```
P   =      2.   .          DDDD DDDD EDDD   0000
                           ---- -7-- 1---
CM  =   6131.
UEC =      0.
 1.  6.IAF    S        PR77.QP7776.RA  1222.FL 143. 1TN
     X. NOS 1.4-552
 2.    NEXT                        RA  1365.

 3.    NEXT                        RA  1721.

 4.    NEXT                        RA  4572.

 5.    NEXT                        RA  4572.

 6.    NEXT                        RA  4572.

 7.    NEXT                        RA  4572.

10.    NEXT                        RA  4572.

11.    NEXT                        RA  4572.

12.    NEXT                        RA  4572.

13.    NEXT                        RA  4572.

14.    NEXT                        RA  4572.

15.    NEXT                        RA  4572.

16. 11.MAGNET S        PR76.QP7773.RA  4572.FL  17.
     X.MAGNET.
17. 10.BATCHIOS        PR76.QP7774.RA  4672.FL   2.
       IDLE.
20.  5.MODVADVS        PR70.QP7761.RA  7537.FL 240.
     X.REQUEST *K* DISPLAY.
21.   .SYSTEM S        PR  .QP7777.RA     0.


 0. MTR 20.  1. DSD 21.  2.       .  3.       .  4.       .
 5.       .  6.       .  7. 1TN  1. 10.       . 11.       .
```

3. Type `K,n.` where *n* is the control point number. In this example, you would type `K,20.` and the following would appear:

NOTE

Some experienced analysts suggest typing `KK.` after `K.n.` so the right display contains additional helpful information.

FIGURE 2-11. Initial MODVAL Display



```
                    K   20.

 17.33.19. 04/11/22.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AAEI. UNLOCKED                 NOS 1.4-552
 MID =  NU.      DEBUG.  VALIDUZ

                   CREATED    40/11/22.
                   UPDATED    40/11/31.

INPUT DIRECTIVES ARE THE SAME AS BATCH INPUT DIRECTIVES.
THE FOLLOWING DIRECTIVES ARE ALSO PROVIDED -

     /AN   - TERMINATE INPUT FOR PRESENT USER NUMBER IF ANY,
             AND UPDATE VALIDUS FILE.  INITIATE ACTION ON *AN*.
     I,AN  - INQUIRE OPTION.  THIS DISPLAY ONLY.
     C,AN  - CREATE OPTION.  THIS DISPLAY ONLY.
     U,AN  - UPDATE OPTION.  THIS DISPLAY ONLY.
     D,AN  - DELETE OPTION.  THIS DISPLAY ONLY.
     +,-   - ADVANCE LEFT SCREEN PAGES (3).
     (,)   - ADVANCE RIGHT SCREEN PAGES (3).
     END   - COMPLETE UPDATE OR INQUIRE OF ACTIVE USER.  FROM
             THIS DISPLAY, END RUN.
     DROP  - TERMINATE INPUT FOR ACTIVE USER.
     STOP  - TERMINATE INPUT FOR ACTIVE USER, IF ANY.  END RUN.
           AN = 1-7 CHARACTER USER NUMBER.

DATA ENTRY FORMAT IS OF THE FORM MT=XX,PR=XX,TL=XX, ETC.
ALL NUMERIC FIELDS ARE ASSUMED TO BE DECIMAL UNLESS A POST-
RADIX IS SPECIFIED.  FOR EXAMPLE -
               4000B

ACCESS WORD IDENTIFIERS STARTING AT BIT 0 ARE -
     CPWC CTPC CLPF CSPF CSOJ CASF CAND CCNR CSRP CSTP CTIM CUCP
     CSAP CBIO CPRT CPLK CQLK CUST
     CIAF CRBF CTAF CMCS CTVF CNOP CLOP CNLO
                         COPE COPR CACC CPPF CATP CDSD CANG CSTF
     CAC2 CSMU CRCP                                    CNTM CCTM
```

```
                          B

P   =      2.   .            DDDD DDDD EDDD   0000
                             ---- -7-- 1---
CM  =   6131.
UEC =      0.

 1.  6.IAF    S         PR77.QP7776.RA  1222.FL 143. 1TN
     X. NOS 1.4-552
 2.    NEXT                         RA  1365.

 3.    NEXT                         RA  1721.

 4.    NEXT                         RA  4572.

 5.    NEXT                         RA  4572.

 6.    NEXT                         RA  4572.

 7.    NEXT                         RA  4572.

10.    NEXT                         RA  4572.

11.    NEXT                         RA  4572.

12.    NEXT                         RA  4572.

13.    NEXT                         RA  4572.

14.    NEXT                         RA  4572.

15.    NEXT                         RA  4572.

16. 11.MAGNET S         PR76.QP7773.RA  4572.FL  17.
     X.MAGNET.
17. 10.BATCHIOS         PR76.QP7774.RA  4672.FL   2.
       IDLE.
20.  5.MODVADVS         PR70.QP7761.RA  7537.FL 240.
     X.REQUEST *K* DISPLAY.
21.   .SYSTEM S         PR  .QP7777.RA     0.



 0. MTR 16.  1. DSD 21.  2.      .  3.      .  4.      .
 5.       .  6.       .  7. 1TN  1. 10.     : 11.      :
```

4. Type `K.C,` *unum* where *unum* is the user number you wish to create. For this example, we'll create the account *PIB*.

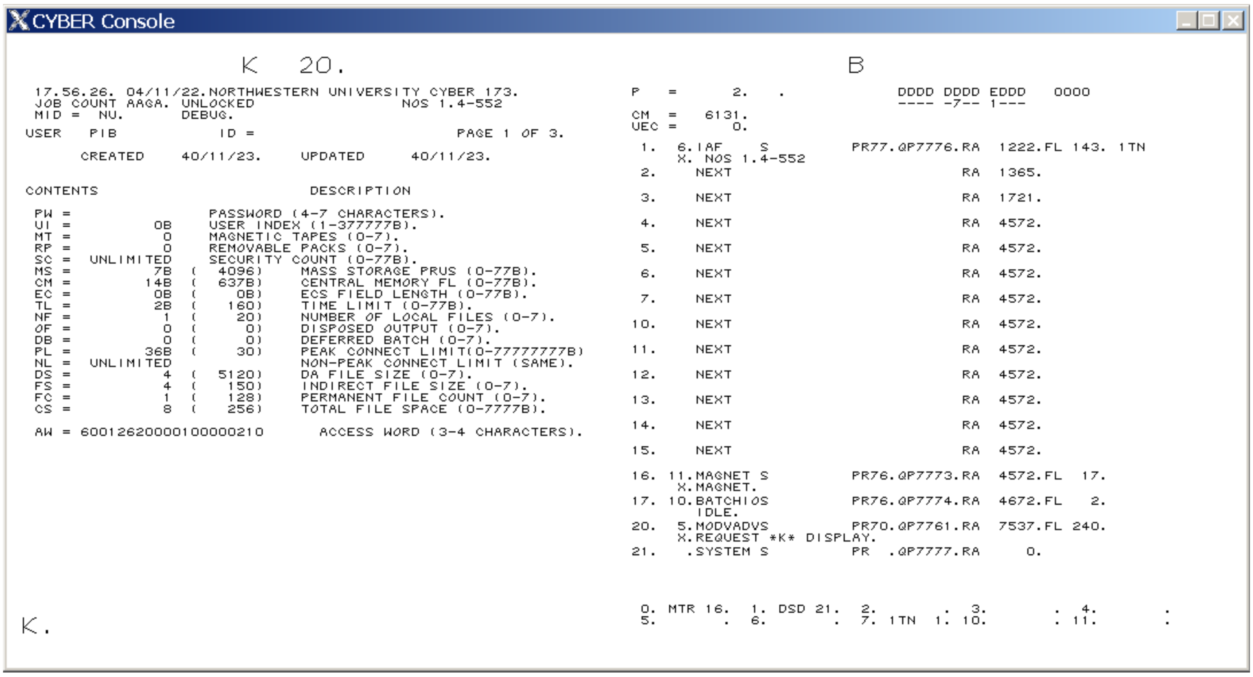
NOTE

User numbers consist of one to seven characters. The characters may be 0 through 9, A through Z, and the asterisk (*). The asterisk in a user number is somewhat like a wildcard; it allows the user implicit read access to the permanent file catalogues of matched user numbers. For example, user EESUM** has implicit read access to the permanent files belonging to EESUMPD and EESUMHA, but not to MESUMTS.

Passwords consist of four to seven characters. Like user numbers, the characters may be 0 through 9, A through Z, and the asterisk.

The following is displayed after typing `K.C,PIB`:

FIGURE 2-12. MODVAL User Creation



```
                    K   20.

 17.56.26. 04/11/22.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AAGA. UNLOCKED                    NOS 1.4-552
 MID =  NU.      DEBUG.
USER   PIB           ID =                         PAGE 1 OF 3.

     CREATED    40/11/23.    UPDATED     40/11/23.


CONTENTS                        DESCRIPTION

 PW =                  PASSWORD (4-7 CHARACTERS).
 UI =          0B      USER INDEX (1-377777B).
 MT =           0      MAGNETIC TAPES (0-7).
 RP =           0      REMOVABLE PACKS (0-7).
 SC =  UNLIMITED       SECURITY COUNT (0-77B).
 MS =          7B  (  4096)    MASS STORAGE PRUS (0-77B).
 CM =         14B  (  637B)    CENTRAL MEMORY FL (0-77B).
 EC =          0B  (    0B)    ECS FIELD LENGTH (0-77B).
 TL =          2B  (   160)    TIME LIMIT (0-77B).
 NF =           1  (    20)    NUMBER OF LOCAL FILES (0-7).
 OF =           0  (     0)    DISPOSED OUTPUT (0-7).
 DB =           0  (     0)    DEFERRED BATCH (0-7).
 PL =         36B  (    30)    PEAK CONNECT LIMIT(0-77777777B)
 NL =  UNLIMITED               NON-PEAK CONNECT LIMIT (SAME).
 DS =           4  (  5120)    DA FILE SIZE (0-7).
 FS =           4  (   150)    INDIRECT FILE SIZE (0-7).
 FC =           1  (   128)    PERMANENT FILE COUNT (0-7).
 CS =           8  (   256)    TOTAL FILE SPACE (0-7777B).

 AW = 60012620000100000210      ACCESS WORD (3-4 CHARACTERS).




 K.
```

```
                      B
P    =       2.   .           DDDD DDDD EDDD   0000
                              ---- -7-- 1---
CM   =    6131.
UEC  =       0.
 1.   6.IAF     S          PR77.QP7776.RA  1222.FL 143. 1TN
       X. NOS 1.4-552
 2.     NEXT                           RA  1365.

 3.     NEXT                           RA  1721.

 4.     NEXT                           RA  4572.

 5.     NEXT                           RA  4572.

 6.     NEXT                           RA  4572.

 7.     NEXT                           RA  4572.

10.     NEXT                           RA  4572.

11.     NEXT                           RA  4572.

12.     NEXT                           RA  4572.

13.     NEXT                           RA  4572.

14.     NEXT                           RA  4572.

15.     NEXT                           RA  4572.

16.  11.MAGNET S           PR76.QP7773.RA  4572.FL  17.
       X.MAGNET.
17.  10.BATCHIOS           PR76.QP7774.RA  4672.FL   2.
         IDLE.
20.   5.MODVADVS           PR70.QP7761.RA  7537.FL 240.
       X.REQUEST *K* DISPLAY.
21.    .SYSTEM S           PR  .QP7777.RA     0.


 0. MTR 16.  1. DSD 21.  2.       .  3.        .  4.       .
 5.       .  6.       .  7. 1TN  1. 10.        . 11.       .
```

5. Select a password for the user and enter it with `PW=`*`password`*. Here, we'll give PIB the password *BOZOS* by specifying `PW=BOZOS`.

NOTE

NOS dates from the days when passwords were kept in plaintext, so take care not to use a password which has value elsewhere—the NOS password can be easily recovered by anyone sitting at the console.

6. If you wish to set a vanity hash code for this user (see *Vanity Hash Codes* on page 2-36), enter the directive `FUI=`*`index`*`B` where *index* is the computed user index from the UNHASH program.

7. If you wish to accept the default permissions for this user (you probably *do not*—see note below), go to step 9 on page 2-36.

NOTE

The default permissions were appropriate 30 years ago, but for Desktop Cyber, they are probably inadequate. For example, the default permissions do not give the user access to tapes. Nevertheless, even today full permissions are most likely inappropriate, since NOS 1.4 provides powerful user monitoring features which allow a fully privileged user to monitor the individual keystrokes of others.

For a list of what to type to give reasonable permissions to a typical user, see the example at the end of the next step.

8. You may now change any of the following permissions. You may specify multiple permissions on the same line, delimited by commas, up to 50 characters. You may enter as many lines as you wish. Sugges-

tions on values are listed at the end of this step.

NOTE

Despite what the limits say, hardware resources are constrained by what is installed and supported by the system. For example, the MECC system can only have at most two tape drives, so specifying a value of MT greater than 2 doesn't give the user any additional resources than MT=2.

TABLE 2-7. MODVAL User Resource Use Permissions

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry
MT	Allowable magnetic tapes	0-7	MT=I If MT=7, Unlimited	MT=7
RP	Removable packs	0-7	RP=I If RP=7, Unlimited	RP=7
SC	Security count	0-77B	SC=I If SC=77B, Unlimited	SC=77B

MS	Mass storage PRUs	0-77B	MS=4096*I+512
			MS=10000B*I+1000B
			If MS=77B, Unlimited
				MS=77B

CM	Central memory field length/100B	0-77B	CM=60*I+17
			CM=74B*I+21B
			If CM=77B, Unlimited
				CM=77B

EC	ECS field length/1000B	0-77B	EC=16*I
			EC=20B*I
			If EC=77B, Unlimited
				EC=77B

TL	Time limit/10B, in seconds	0-77B	TL=64*I+8
			TL=100B*I+10B
			If TL=77B, Unlimited
				TL=77B

NF	Number of local files	0-7	NF=8*I+8
			NF=10B*I+10B,
			If NF=7, Unlimited
				NF=7

OF	Number of simultaneous output files	0-7	OF=4*I If OF=7, Unlimited	OF=7
DB	Number of simultaneous deferred batch jobs	0-7	DB=2*I If DB=7, Unlimited	DB=7
PL	Peak connect limit	0-77777777B	PL=I If PL=77777777B, Unlimited	PL=77777777B

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry
NL	Non-peak connect limit	0-77777777B	NL=I If NL=77777777B, Unlimited	NL=77777777B
DS	Direct access file size in PRUs (there are 64 CM words (640 characters) in a PRU)	0-7	Index Value 0 Job orig. default 1 512 2 1024 3 2560 4 4096 5 20480 6 32768 7 Unlimited	DS=7
FS	Indirect access file size in PRUs	0-7	Index Value 0 Job orig. default 1 8 2 24 3 40 4 64 5 104 6 504 7 Unlimited	FS=6[a]
FC	Number of permanent files	0-7	Index Value 0 Job orig. default 1 8 2 16 3 24 4 32 5 40 6 64 7 Unlimited	FC=7

CS	Cumulative file size, in PRUs	0-7777B	CS=32*I
			CS=40B*I
			If CS=0, Unlimited
				CS=0

LP	Lines printed per print job	0-77B	LP=1024*I+512
			LP=2000B*I+1000B
			If LP=77B, Unlimited
				LP=77B

CP	Cards punched per job—this must be non-zero if you wish to allow the user to mount their own tapes by issuing directives to the emulator[b]	0-77B	CP=64*I
			CP=100B*I
			If CP=77B, Unlimited
				CP=77B

CC	Control cards per job	0-77B	CC=16*I+64
			CC=20B+100B
			If CC=77B, Unlimited
				CC=77B

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry

DF	Dayfile messages per job	0-77B	DF=16*I+64
			DF=20B*I+100B
			If DF=77B, Unlimited
				DF=77B

SL	System Resource Unit (SRU) limit per job	0-77B	SR=64*I+8
			SR=100B*I+10B
			If SR=77B, Unlimited
				SL=77B

CN	Charge number			Leave blank
PN	Project number			Leave blank
AB	Terminal answer-back string			Leave blank
PA	Terminal parity	EVEN, ODD		EVEN
PO	Rubouts between carriage returns[c]	0-37B	PO=I If PO=37B, System default	PO=0
PX	Echoplexing	FULL, HALF	FULL means remote echo HALF means local echo	FULL
TT	Terminal type	TTY		TTY

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry
TC	Character set	ASCII, STANDAR (*sic*)		STANDAR
IS	Initial interactive subsystem	BATCH, BASIC, COBOL, EXECUTE, FORTRAN, FTNTS, MNFTS, M77TS, NULL, PASCAL, SNOBOL, ABASIC, FCOBOL, ACCESS		BATCH

AW
AP
ON
OFF
	Access control word specifying up to 60 permission bits.	See below	See below	*For users:*
				AP=ALL,
				AW=ALL,
				OFF=CSOJ,
				OFF=CTPC
				
				For systems programmers:
				AW=ALL,
				AP=ALL

a. We recommend against setting FS=7 because very large indirect access files may cause considerable performance degradation. Users needing such large files should use direct access files.

b. Users do not have to have CP nonzero to *use* tapes, but they must have CP nonzero to issue their own directives to the emulator to *mount* tapes without the operator's assistance.

c. This is a relic from a bygone day; rubouts were conventionally sent by the host after a carriage return on a Teletype terminal to give the printing carriage time to move to the beginning of the next line before additional data were sent.

The Access Control Word (AW) is a 60-quantity which specifies various user capabilities. They are toggled in MODVAL by specifying `AW=`*`cname`*. Using `ON=`*`cname`* turns on the permission, `OFF=`*`cname`* turns it off, and `AW=`*`cname`* toggles it. (The ON and OFF directives are undocumented features.)

Also supported, but not implemented here, are NAM application permissions. NAM is not running on this system, so MECC has redefined some of the NAM application bits. Application bits are specified with the `AP=`*`appl`* directive.

For both `AP=` and `AW=`, you may specify `ALL` to set all of the bits and `NUL` to clear all of the bits. Thus, `K.AW=ALL,AP=ALL` sets every possible bit and `K.AW=NUL,AP=NUL` clears every possible bit.

Here is a list of the capabilities supported by MODVAL, not all of which have applicability here:

NOTE

You may bring up a detailed description of these bits in MODVAL by issuing the DSD command `KK.` which will bring up the K display on both displays. Use the `K.(` and `K.)` commands to page through the right-hand display.

TABLE 2-8. MODVAL Capability Descriptions

Bit	MODVAL Capability Name	Significance
0	CPWC	Password change by the user permitted.[a]
1	CTPC	Telex privileged commands permitted.[b]
2	CLPF	Create large (direct access) permanent files.
3	CSPF	Create small (indirect access) permanent files.
4	CSOJ	System origin privileges. [c]
5	CASF	Create and access system library files—in other words, the user may use the COMMON command.[d]
6	CAND	Assign nonallocable devices—that is, request tapes.
7	CCNR	Charge not required. This bit should always be set.
8	CSRP	Access removable disk packs.
9	CSTP	Special transaction privileges. Not applicable here.
10	CTIM	User's terminal will not time out from inactivity.
11	CUCP	System Control Point facility use permitted.
12	CSAP	Special accounting privileges; this gives the user special access to charge and project number data. Not applicable here.
13	CBIO	Special BATCHIO privileges—specifically, the user may override the form control used on printers via the V carriage control character.
14	CPRT	User may use the PROTECT command to save ECS across job steps.
15	CPLK	Access link for permanent files. Not applicable here.
16	CQLK	Access link for queued files. Not applicable here.
17	CUST	Use ST on ROUTE and LINK statements. Not applicable here.
19	CSFC	Special forms code permitted on ROUTE statement.
24	CIAF	Interactive access permitted.
25	CRBF	Batch job access permitted.

Bit	MODVAL Capability Name	Significance
26	CTAF	Transaction facility access permitted. Not applicable here.
27	CMCS	Message control system access permitted. Not applicable here.
28	CTVF	Terminal verification facility access permitted. Not applicable here.
29	CNOP	Network operator access. Not applicable here.
30	CLOP	Local operator access. Not applicable here.
31	CNLO	Network/local operator access. Not applicable here.
40	COPE	Account is active. Should always be set.
41	COPR	Access alternate catalogs. When set, this permits the user to access other permanent file catalogs provided that the permissions on the alternate files allow such access.
42	CACC	Issue secondary user cards. The user may issue additional USER cards after logging in, usually to give access to that user's permanent files.
43	CPPF	Protected permanent file creation allowed. When set, allows the user to require a password to access files in that user's permanent file catalog. Used in environments where multiple users share the same account.
44	CATP	Access group catalogs. Should always be set. Deprecated.
45	CDSD	May use the DSD command—an interactive command which gives access to console displays.
46	CANG	Access non-group catalogs. Should always be set. Deprecated.
47	CSTF	Computer center staff access; provides access to the (not running) accounting system.
48	CAC2	Allow access via secondary user card. If this bit is not set, the account cannot be USERed to.
49	CSMU	Special master user. This is a feature of the accounting system allowing master users to control some permissions of their subordinate accounts.
50	CRCP	Remote console privileges. Not applicable here.
58	CNTM	Non-peak hours access. Should always be set.

Bit	MODVAL Capability Name	Significance
59	CCTM	Peak hours access. Should always be set.

a. Today, of course, it seems axiomatic that a user should be able to change the password, but at the time, users in remote locations often shared accounts, especially in academic environments.

b. This allows use of the ACCESS subsystem. Set this bit with great caution since under NOS 1, this gives far greater privileges than under NOS 2; an ACCESS user may monitor any user's keystrokes, send unsolicited (and anonymous) messages to any user's terminal, and may log any user off the system. Master user numbers—those containing one or more stars ("*") in the user number—have implicit access to these commands for their subordinate accounts whether or not this bit is set. See ***[Privileged Telex Commands Xref]*** for more details.

c. This means that the user may issue many commands which require system origin (that is, being run from the console) from any job origin (interactive or batch). Using this privilege requires the system be in debug mode (see the DSD DEBUG command, page 2-14). Set this bit with great caution since it permits the sufficiently-clever user to steal files belonging to other people. Whether the system is in debug mode or not, this bit also permits the user to assign equipment by EST ordinal.

d. In practice, what this means is that the user may obtain a local copy of the running operating system by issuing a COMMON,SYSTEM. command.

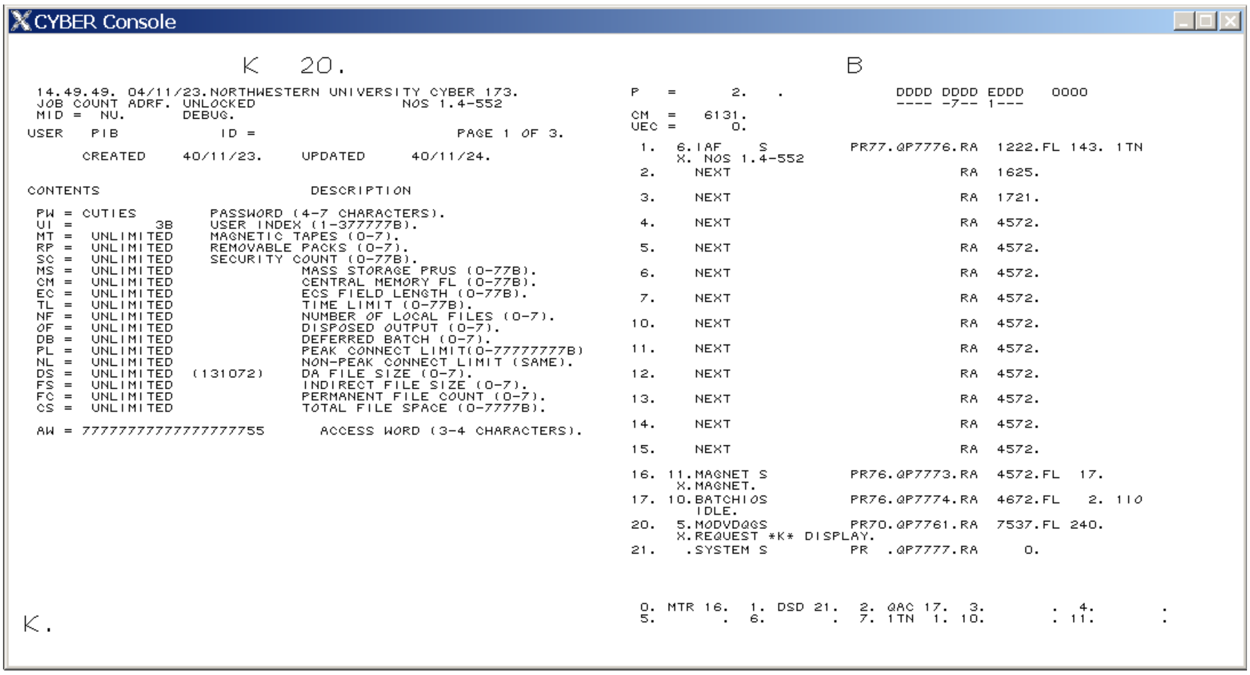
For our new user PIB (and these limits are appropriate for most users), we select:

FIGURE 2-13. Typical User Permissions

```
K.MT=7,RP=7,SC=77B,MS=77B,CM=77B,EC=77B,TL=77B
                    ┌──8───┐
K.NF=7,OF=7,DB=7,PL=77777777B,DS=7,FS=6,FC=7,CS=0
K.LP=77B,CP=77B,CC=77B,DF=77B,PX=FULL,PO=0,IS=BATCH
K.AP=ALL,AW=ALL,OFF=CSOJ,OFF=CTPC
```

The display now looks like:

FIGURE 2-14. MODVAL User Entry



```
                    K   20.

 14.49.49. 04/11/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT ADRF. UNLOCKED                    NOS 1.4-552
 MID =  NU.      DEBUG.
USER   PIB            ID =                        PAGE 1 OF 3.

     CREATED    40/11/23.    UPDATED    40/11/24.


CONTENTS                        DESCRIPTION

 PW = CUTIES           PASSWORD (4-7 CHARACTERS).
 UI =          3B      USER INDEX (1-377777B).
 MT =  UNLIMITED       MAGNETIC TAPES (0-7).
 RP =  UNLIMITED       REMOVABLE PACKS (0-7).
 SC =  UNLIMITED       SECURITY COUNT (0-77B).
 MS =  UNLIMITED                   MASS STORAGE PRUS (0-77B).
 CM =  UNLIMITED                   CENTRAL MEMORY FL (0-77B).
 EC =  UNLIMITED                   ECS FIELD LENGTH (0-77B).
 TL =  UNLIMITED                   TIME LIMIT (0-77B).
 NF =  UNLIMITED                   NUMBER OF LOCAL FILES (0-7).
 OF =  UNLIMITED                   DISPOSED OUTPUT (0-7).
 DB =  UNLIMITED                   DEFERRED BATCH (0-7).
 PL =  UNLIMITED                   PEAK CONNECT LIMIT(0-77777777B)
 NL =  UNLIMITED                   NON-PEAK CONNECT LIMIT (SAME).
 DS =  UNLIMITED  (131072)         DA FILE SIZE (0-7).
 FS =  UNLIMITED                   INDIRECT FILE SIZE (0-7).
 FC =  UNLIMITED                   PERMANENT FILE COUNT (0-7).
 CS =  UNLIMITED                   TOTAL FILE SPACE (0-7777B).

 AW = 77777777777777777755          ACCESS WORD (3-4 CHARACTERS).









K.
```

```
                    B

P   =     2.   .            DDDD DDDD EDDD   0000
                            ---- -7-- 1---
CM  =   6131.
UEC =      0.

 1.  6.IAF    S         PR77.QP7776.RA  1222.FL 143. 1TN
     X. NOS 1.4-552
 2.    NEXT                         RA  1625.

 3.    NEXT                         RA  1721.

 4.    NEXT                         RA  4572.

 5.    NEXT                         RA  4572.

 6.    NEXT                         RA  4572.

 7.    NEXT                         RA  4572.

10.    NEXT                         RA  4572.

11.    NEXT                         RA  4572.

12.    NEXT                         RA  4572.

13.    NEXT                         RA  4572.

14.    NEXT                         RA  4572.

15.    NEXT                         RA  4572.

16. 11.MAGNET S         PR76.QP7773.RA  4572.FL  17.
     X.MAGNET.
17. 10.BATCHIOS         PR76.QP7774.RA  4672.FL   2. 110
       IDLE.
20.  5.MODVDQGS         PR70.QP7761.RA  7537.FL 240.
     X.REQUEST *K* DISPLAY.
21.   .SYSTEM S         PR  .QP7777.RA     0.



 0. MTR 16.  1. DSD 21.  2. QAC 17.  3.       .  4.       .
 5.       .  6.       .  7. 1TN  1. 10.       . 11.       .
```

If you wish, you may page through the two other MODVAL pages by using the + and - keys followed by ENTER, but if you are following the example, you have already set everything which needs setting.

9. Type `K.END` to go back to the main MODVAL display.

10. If you want to add more users, go to step 4 on page 2-27.

11. Type `K.END` to exit MODVAL.

CLOSING AN ACCOUNT

NOTE

Despite what you read on the K display, you cannot use K.D,unum to delete an account. It doesn't work.

1. At the console, type `X.MODVAL.` Note the control point number requesting the K display.

2. Type `K,`*n.* where *n* is the control point number.

3. Type `K.U,`*unum* where *unum* is the user number you wish to delete.

4. Type `K.SC=0,OFF=COPE`

5. Type `K.END`

6. If you have more accounts to delete, go to step 3 on page 2-36.

7. Type `K.END`

VANITY HASH CODES

When creating an account, MODVAL will automatically assign a user index, and, consequently, a hash code. The automatic user index selection can be overridden, however. One reason for specifying a specific user index is that it permits a custom, vanity-style hash code.

All hash codes, because of the digest algorithm, are four characters long consisting of the characters A through Z and 0 through 5. The fourth character must be either A, I, Q or Y.

To compute the user index corresponding to a specified hash code:

1. Issue the command `X.DIS.`

2. Issue the command `SUN,JPD.`

3. Issue the command `GET,UNHASH.` This is a Fortran V source code file.

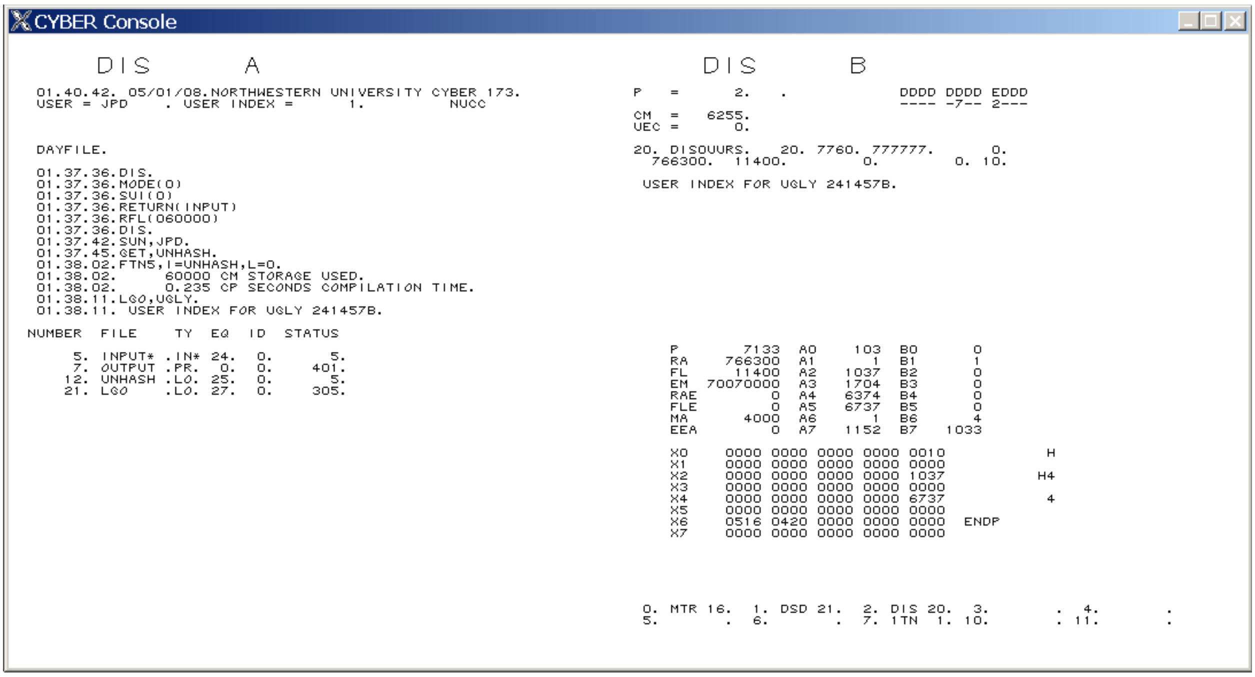
4. Compile the program with the command `FTN5,I=UNHASH,L=0.`

5. Execute the compiled program with `LGO,`*hash.* where *hash* is the desired hash code.

6. Read the corresponding user index on the DIS A or B display. This is the index you will specify for the `FUI=` directive when creating the account.

Example. The corresponding user index for hash code UGLY is 241457B as shown in the following figure.

FIGURE 2-15. Computing a Vanity Hash Code



```
      D I S         A

 01.40.42. 05/01/08.NORTHWESTERN UNIVERSITY CYBER 173.
 USER = JPD     . USER INDEX =       1.           NUCC


 DAYFILE.

 01.37.36.DIS.
 01.37.36.MODE(0)
 01.37.36.SUI(0)
 01.37.36.RETURN(INPUT)
 01.37.36.RFL(060000)
 01.37.36.DIS.
 01.37.42.SUN,JPD.
 01.37.45.GET,UNHASH.
 01.38.02.FTN5,I=UNHASH,L=0.
 01.38.02.     60000 CM STORAGE USED.
 01.38.02.     0.235 CP SECONDS COMPILATION TIME.
 01.38.11.LGO,UGLY.
 01.38.11. USER INDEX FOR UGLY 241457B.

NUMBER  FILE    TY  EQ  ID  STATUS

    5.  INPUT* .IN* 24.  0.      5.
    7.  OUTPUT .PR.  0.  0.    401.
   12.  UNHASH .LO. 25.  0.      5.
   21.  LGO    .LO. 27.  0.    305.
```

```
      D I S         B

P    =      2.   .             DDDD DDDD EDDD
                               ---- -7-- 2---
CM   =   6255.
UEC  =      0.

20. DISOUURS.   20. 7760. 777777.      0.
  766300.  11400.        0.         0. 10.

 USER INDEX FOR UGLY 241457B.



    P       7133  A0    103  B0      0
    RA    766300  A1      1  B1      1
    FL     11400  A2   1037  B2      0
    EM  70070000  A3   1704  B3      0
    RAE        0  A4   6374  B4      0
    FLE        0  A5   6737  B5      0
    MA      4000  A6      1  B6      4
    EEA        0  A7   1152  B7   1033

    X0    0000 0000 0000 0000 0010               H
    X1    0000 0000 0000 0000 0000
    X2    0000 0000 0000 0000 1037              H4
    X3    0000 0000 0000 0000 0000
    X4    0000 0000 0000 0000 6737               4
    X5    0000 0000 0000 0000 0000
    X6    0516 0420 0000 0000 0000  ENDP
    X7    0000 0000 0000 0000 0000


0. MTR 16.   1. DSD 21.   2. DIS 20.   3.         .  4.       :
5.       .   6.       .   7. 1TN  1.  10.         : 11.       :
```

JOB MANAGEMENT

JOB ORIGIN TYPES

NOS provides different scheduling priorities, limits, and, when applicable, billing rates to jobs according to how the job was initiated. For example, jobs submitted through the card reader for an overnight run would typically get low-priority access to the CPU and other resources but get charged a lower rate than interactive jobs, which typically get more preferential access to system resources but get billed at a higher rate for that preferential treatment.

Although there are many job origin types, there are but three commonly used:

- *Interactive, timesharing,* or *telex* jobs (all three terms refer to the same thing) are those run from interactive terminals.

- *Batch* jobs are not interactive. They run all of their job steps in sequence detached from the user's immediate control. Batch jobs come from three sources: In a bygone era, they were most commonly submitted by punched card decks; they may also be submitted by a control statement from any origin job (see ***[Submit and Route Xref]***); and they may be submitted by an external Desktop Cyber protocol which permits submission via email and host command language (see ***[External Interfaces Xref]***).

- *System* jobs are those submitted either as part of the dead start process or submitted from the console by either the X. command or through DIS. System origin jobs have special privileges that other jobs do not; for example, system origin jobs do not need to validate the user, and can get easy access to all permanent files on the system.

TAPE JOBS

NOTE

The system comes configured for two tape units—NT50 and NT51. This particular release does not support more than two tape drives, even if you configure more in the CMRDECK and *cyber.ini*.

A job may request a tape in several ways:

- The job may issue a LABEL control statement, which requests tape mounting and assignment through MAGNET, the resource executive. This is the most common approach. Tape requests go into a queue which the operator views on the E,P display. If the operator does not check E,P regularly, though, the requesting job may be in for a long wait.

- The job may issue a REQUEST control statement, which, depending upon supplied parameters, will either queue the request as if it had been made by LABEL, or flash a user-supplied message on the B display requesting the tape mount. In large installations, this last approach is considered rather rude, since it occupies a control point while the request is outstanding, and it does an end-run around the regular queue of tape requests. However, for such a lightly-loaded system (so one expects), REQUEST has the advantage of getting the operator's attention from across the room.

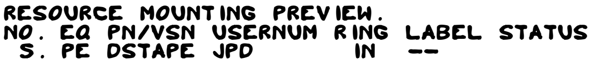
- If the job is of system origin or has system origin privileges (see the LIMITS command on page *[LIMITS Xref]* and MODVAL access control permissions in Table 2-8 on page 2-33), and the tape is already mounted but not assigned to a job, the job may use the ASSIGN command to directly assign the tape unit without going through the operator.

- The user's program may use an operating system interface to make a tape request without using a control statement. This is rare; it's typically done by system backup and restore software.

For Desktop Cyber, "tapes" are actually encapsulated disk files which typically have been read and converted offline from actual tape volumes. These files reside on the system hosting Desktop Cyber, not as NOS permanent files. We suggest that tape files have the same name as the original tape's VSN.

Like tapes, these files must be "mounted," or made available to Desktop Cyber. To NOS, such files have precisely the same behaviour as conventional tapes.

Tape mounts requested via LABEL (or REQUEST with enough parameters to identify the specific tape volume) will appear on the E,P display. Here is an example:



```
RESOURCE  MOUNTING  PREVIEW.
NO.  EQ  PN/VSN  USERNUM  RING  LABEL  STATUS
  S.  PE  DSTAPE  JPD         IN    --
```

The number refers to the job's FNT entry; you would use this number to PURGE the job if the VSN did not exist.

EQ refers to the specific equipment being requested, which you may ignore—it may be either *PE* (meaning phase encoding, or 1200 cpi density) or *GE* (group encoding, or 6250 cpi density).

VSN refers, of course, to the volume serial number of the tape. USERNUM is precisely that, so you can identify the requesting user if you have a question.

RING will be either *IN*, *OUT*, or “--.” This refers to a plastic ring which the operator can fit around the inner rim on the back of the tape to permits writing on the tape. *IN* means that the tape should be mounted writable; *OUT* means read-only, and “--” means that the user did not specify.

NOTE

Ring status is set with Desktop Cyber at the time the encapsulated tape file is loaded. Specifying the write option (W) at the end of the LOAD command causes the tape drive emulator to behave as if the tape were mounted with the ring in.

LABEL indicates whether the requesting user has specified if the tape has ANSI labels on it. ANSI labels permit the system to automatically identify the tape and assign it to the requesting job automatically; they also guard against the operator accidentally assigning the wrong tape to the job.

If a requested tape is labeled, the operator merely has to mount the tape. If the tape is not labeled, the operator must issue a VSN command so that the system knows which tape unit to assign. (See the VSN command, page 2-17.)

If the tape has been mounted incorrectly, the STATUS field will explain what’s wrong. The most common (by far) problem is a ring conflict—you mounted the tape read-write when read-only was requested, or you mounted it read-only and the ring in was requested.

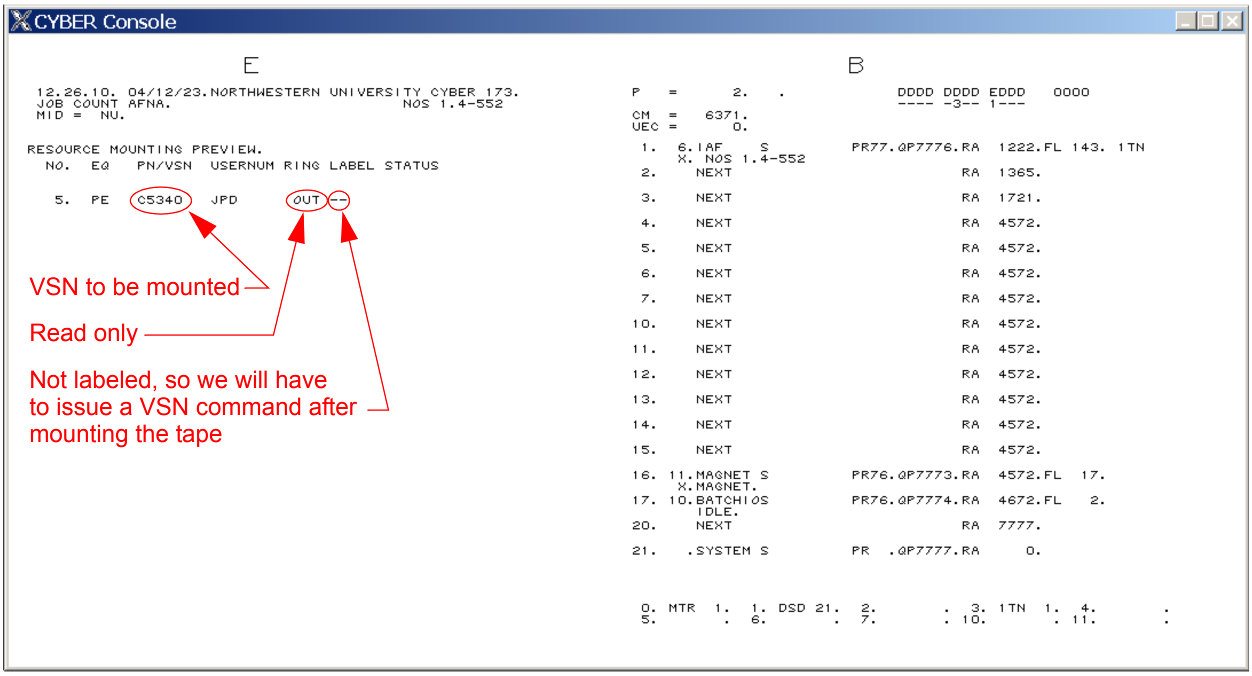
To mount a tape, you need not have a pending request. Thus, if there is a tape which you frequently use, you could just keep it mounted (by appropriate configuration grammar in the host *cyber.ini* file) at dead start and use it at your leisure.

How to Handle a Tape Request on the E,P Display

In this procedure, we will assume that you wish to mount a tape on NT51, which under the released configuration, is on channel 13, unit 1 (see the E,A display). To use NT50 instead of NT51, just substitute NT51 for NT50, and channel 13, unit 0 for channel 13, unit 1.

1. Check the E,P display and note the requested VSN and ring. See Figure 2-16 below.

FIGURE 2-16. Example E,P Display



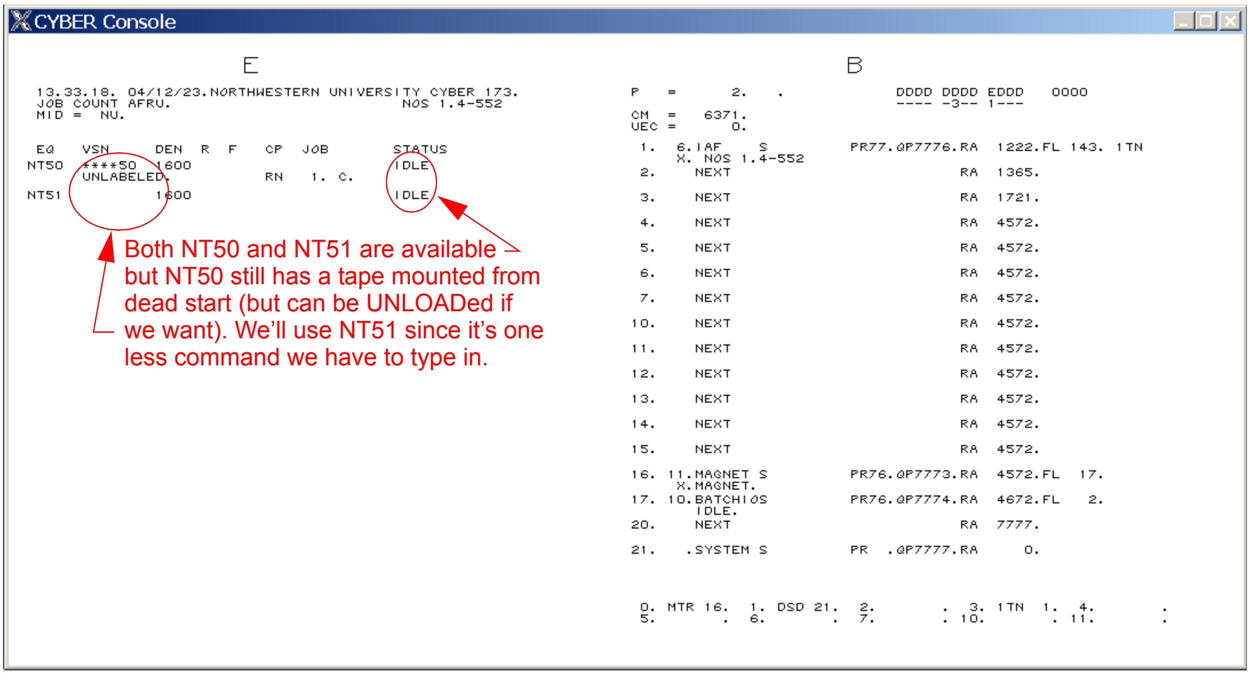
CYBER Console
E
B
VSN to be mounted
Read only
Not labeled, so we will have to issue a VSN command after mounting the tape

```
 12.26.10. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AFNA.                         NOS 1.4-552
 MID =  NU.

RESOURCE MOUNTING PREVIEW.
  NO.  EQ   PN/VSN  USERNUM RING LABEL STATUS

   5.  PE   C5340   JPD          OUT --
```

```
P    =       2.    .              DDDD DDDD EDDD   0000
                                  ---- -3-- 1---
CM   =    6371.
UEC  =       0.
 1.   6.IAF     S          PR77.QP7776.RA   1222.FL 143.  1TN
      X.  NOS 1.4-552
 2.     NEXT                           RA   1365.

 3.     NEXT                           RA   1721.

 4.     NEXT                           RA   4572.

 5.     NEXT                           RA   4572.

 6.     NEXT                           RA   4572.

 7.     NEXT                           RA   4572.

10.     NEXT                           RA   4572.

11.     NEXT                           RA   4572.

12.     NEXT                           RA   4572.

13.     NEXT                           RA   4572.

14.     NEXT                           RA   4572.

15.     NEXT                           RA   4572.

16.  11.MAGNET S           PR76.QP7773.RA   4572.FL  17.
      X.MAGNET.
17.  10.BATCHIOS           PR76.QP7774.RA   4672.FL   2.
        IDLE.
20.     NEXT                           RA   7777.

21.    .SYSTEM S           PR   .QP7777.RA      0.



 0. MTR  1.  1. DSD 21.  2.        .  3. 1TN  1.  4.       .
 5.        .  6.        .  7.        . 10.        . 11.       :
```

2. Check the E,T display and make sure that NT51 is not in use. See Figure 2-17 on page 2-40.

FIGURE 2-17. Example E,T Display



```
                     E

13.33.18. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AFRU.                            NOS 1.4-552
MID =  NU.

EQ    VSN      DEN  R  F   CP  JOB       STATUS
NT50  ****50   1600                      IDLE
      UNLABELED.               RN   1. C.
NT51           1600                      IDLE
```

```
                      B

P   =      2.    .          DDDD DDDD EDDD   0000
                            ---- -3-- 1---
CM  =   6371.
UEC =      0.
 1.  6.IAF    S        PR77.QP7776.RA  1222.FL 143. 1TN
     X. NOS 1.4-552
 2.    NEXT                        RA  1365.
 3.    NEXT                        RA  1721.
 4.    NEXT                        RA  4572.
 5.    NEXT                        RA  4572.
 6.    NEXT                        RA  4572.
 7.    NEXT                        RA  4572.
10.    NEXT                        RA  4572.
11.    NEXT                        RA  4572.
12.    NEXT                        RA  4572.
13.    NEXT                        RA  4572.
14.    NEXT                        RA  4572.
15.    NEXT                        RA  4572.
16. 11.MAGNET S        PR76.QP7773.RA  4572.FL  17.
     X.MAGNET.
17. 10.BATCHIOS        PR76.QP7774.RA  4672.FL   2.
       IDLE.
20.    NEXT                        RA  7777.
21.   .SYSTEM S        PR   .QP7777.RA     0.


 0. MTR 16.  1. DSD 21.  2.       .  3. 1TN  1.  4.       .
 5.       .  6.       .  7.       . 10.       . 11.       .
```

Both NT50 and NT51 are available but NT50 still has a tape mounted from dead start (but can be UNLOADed if we want). We'll use NT51 since it's one less command we have to type in.

3. At the console, type ALT + O to enter Desktop Cyber's operator mode.

4. If the tape is to be mounted with the write ring in, issue the command:

   ```
   LOAD,13,1,filename,W.
   ```

   If the tape is to be mounted ring out issue the command:

   ```
   LOAD,13,1,filename
   ```

   where *filename* is the name of the encapsulated tape file.

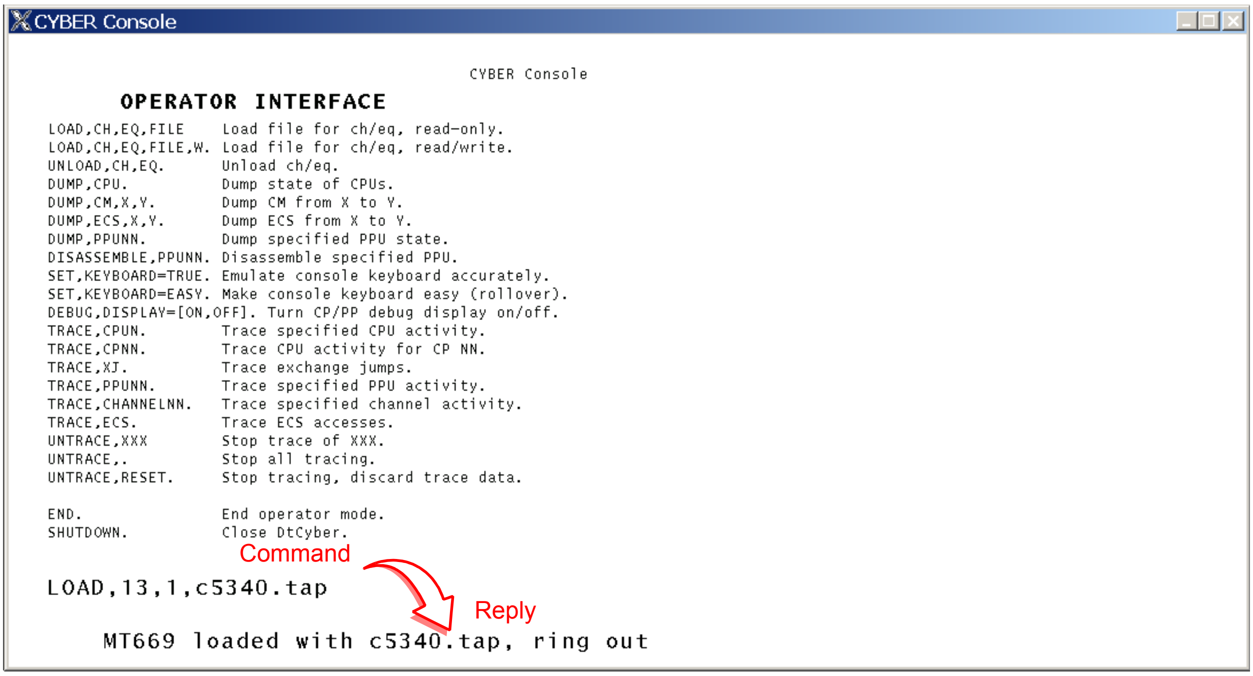
NOTE

Do not type a period at the end of *filename* unless the file name really does have a final period in its name.

If the file name itself contains a comma, rename it so that it does not.

See Figure 2-18 on page 2-41.

FIGURE 2-18. **Desktop Cyber Operator Interface**



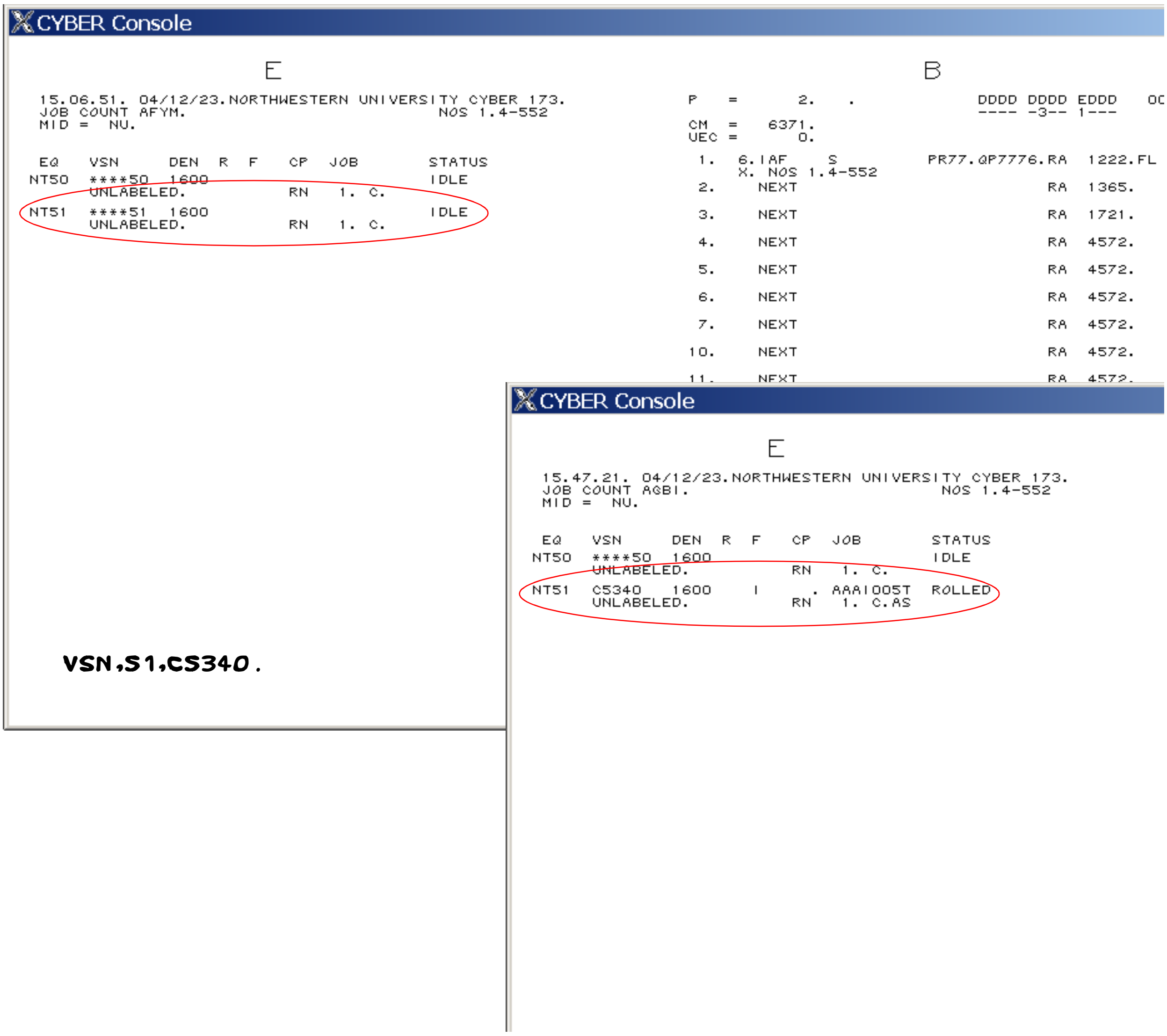
```
                                   CYBER Console
        OPERATOR  INTERFACE
LOAD,CH,EQ,FILE    Load file for ch/eq, read-only.
LOAD,CH,EQ,FILE,W. Load file for ch/eq, read/write.
UNLOAD,CH,EQ.      Unload ch/eq.
DUMP,CPU.          Dump state of CPUs.
DUMP,CM,X,Y.       Dump CM from X to Y.
DUMP,ECS,X,Y.      Dump ECS from X to Y.
DUMP,PPUNN.        Dump specified PPU state.
DISASSEMBLE,PPUNN. Disassemble specified PPU.
SET,KEYBOARD=TRUE. Emulate console keyboard accurately.
SET,KEYBOARD=EASY. Make console keyboard easy (rollover).
DEBUG,DISPLAY=[ON,OFF]. Turn CP/PP debug display on/off.
TRACE,CPUN.        Trace specified CPU activity.
TRACE,CPNN.        Trace CPU activity for CP NN.
TRACE,XJ.          Trace exchange jumps.
TRACE,PPUNN.       Trace specified PPU activity.
TRACE,CHANNELNN.   Trace specified channel activity.
TRACE,ECS.         Trace ECS accesses.
UNTRACE,XXX        Stop trace of XXX.
UNTRACE,.          Stop all tracing.
UNTRACE,RESET.     Stop tracing, discard trace data.

END.               End operator mode.
SHUTDOWN.          Close DtCyber.

LOAD,13,1,c5340.tap

      MT669 loaded with c5340.tap, ring out
```

5. Issue the command `END.` to return to Desktop Cyber’s emulation mode.

6. If the tape is unlabeled, issue the command `VSN,51,`*`vvvvvv`*`.` where *vvvvvv* is the VSN of the tape you just mounted. See Figure 2-19 on page 2-42.

If the tape is labeled, you do not need to issue a VSN command; the system will assign the tape to the job automatically, provided that the correct VSN were specified.

FIGURE 2-19. E,T Before and After VSN Command



```
                    E

 15.06.51. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AFYM.                        NOS 1.4-552
 MID =  NU.

 EQ    VSN      DEN  R  F   CP  JOB        STATUS
NT50   ****50   1600                       IDLE
       UNLABELED.           RN   1. C.
NT51   ****51   1600                       IDLE
       UNLABELED.           RN   1. C.



      VSN,51,CS340.
```

```
                         B

P   =     2.   .             DDDD DDDD EDDD   OC
                             ---- -3-- 1---
CM  =  6371.
UEC =     0.
 1.   6.IAF    S        PR77.QP7776.RA  1222.FL
      X. NOS 1.4-552
 2.    NEXT                         RA  1365.

 3.    NEXT                         RA  1721.

 4.    NEXT                         RA  4572.

 5.    NEXT                         RA  4572.

 6.    NEXT                         RA  4572.

 7.    NEXT                         RA  4572.

10.    NEXT                         RA  4572.

11.    NEXT                         RA  4572.
```

```
                    E

 15.47.21. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AGBI.                        NOS 1.4-552
 MID =  NU.

 EQ    VSN      DEN  R  F   CP  JOB        STATUS
NT50   ****50   1600                       IDLE
       UNLABELED.           RN   1. C.
NT51   C5340    1600    I    .  AAAI005T   ROLLED
       UNLABELED.           RN   1. C.AS
```

7. Verify that the tape request was satisfied by checking E,P and correcting any problems. (If the request were properly satisfied, the request will no longer appear on E,P.)

How to Handle a Tape Request on the B Display

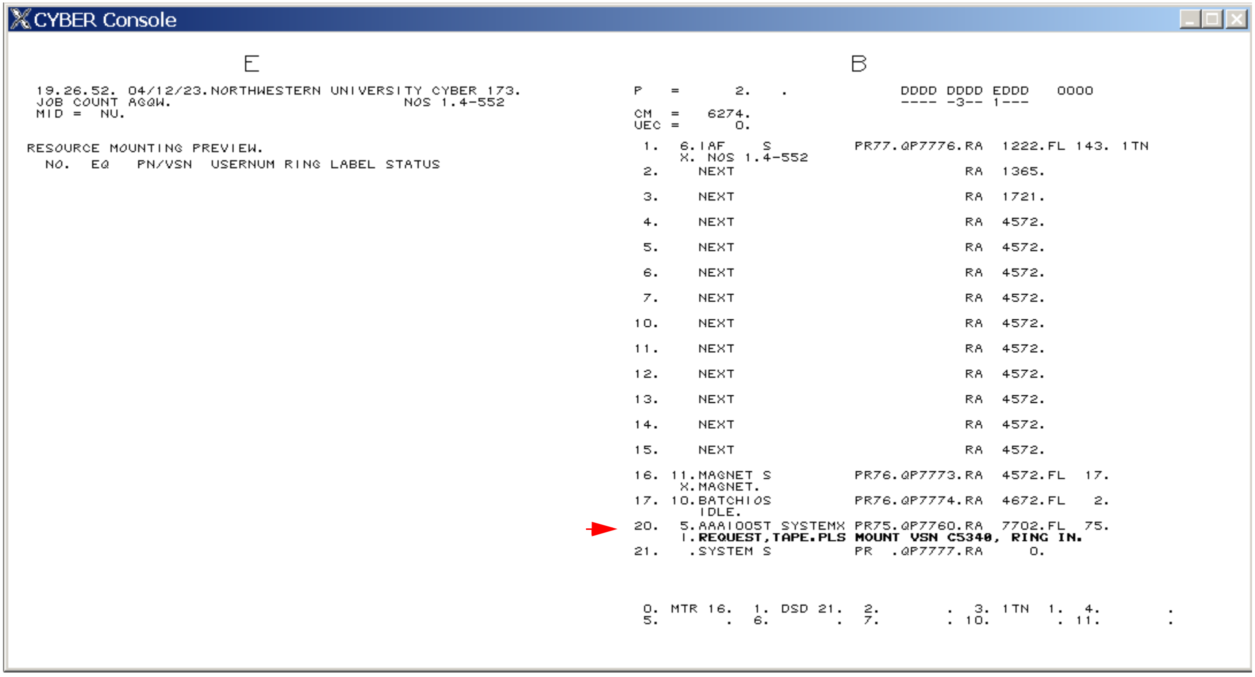
In this procedure, we will assume that you wish to mount a tape on NT51, which under the released configuration, is on channel 13, unit 1 (see the E,A display). To use NT50 instead of NT51, just substitute NT51 for NT50, and channel 13, unit 0 for channel 13, unit 1.

1. Whenever a tape is requested via the B display, a message will flash with the request as shown in the example below at control point 20. Note that there is no corresponding request on the E,P display.

   You will note that most of the request is in plain English; the message was passed to the B display as a comment on the REQUEST control statement. The message reads:

   ```
   REQUEST,TAPE.PLS MOUNT VSN
   CS340, RING IN.
   ```

FIGURE 2-20. Tape Request via the B Display



```
                    E

 19.26.52. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AGQW.                          NOS 1.4-552
 MID =  NU.

RESOURCE MOUNTING PREVIEW.
  NO.  EQ   PN/VSN  USERNUM RING LABEL STATUS
```

```
                          B

P    =       2.    .               DDDD DDDD EDDD   0000
                                   ---- -3-- 1---
CM   =    6274.
UEC  =       0.
 1.   6.IAF     S         PR77.QP7776.RA  1222.FL 143.  1TN
      X. NOS 1.4-552
 2.     NEXT                          RA  1365.

 3.     NEXT                          RA  1721.

 4.     NEXT                          RA  4572.

 5.     NEXT                          RA  4572.

 6.     NEXT                          RA  4572.

 7.     NEXT                          RA  4572.

10.     NEXT                          RA  4572.

11.     NEXT                          RA  4572.

12.     NEXT                          RA  4572.

13.     NEXT                          RA  4572.

14.     NEXT                          RA  4572.

15.     NEXT                          RA  4572.

16.  11.MAGNET S          PR76.QP7773.RA  4572.FL  17.
       X.MAGNET.
17.  10.BATCHIOS          PR76.QP7774.RA  4672.FL   2.
        IDLE.
20.   5.AAAIOO5T SYSTEMX  PR75.QP7760.RA  7702.FL  75.
      I.REQUEST,TAPE.PLS MOUNT VSN C5340, RING IN.
21.    .SYSTEM S          PR  .QP7777.RA     0.


 0. MTR 16.  1. DSD 21.  2.        .  3. 1TN  1.  4.       .
 5.        .  6.        .  7.        . 10.        . 11.       .
```

Mount the tape precisely as you would for a request made through the E,P display:

2. Check the E,T display and make sure that NT51 is not in use. See Figure 2-17 on page 2-40.
3. At the console, type ALT + O to enter Desktop Cyber's operator mode.
4. If the tape is to be mounted with the write ring in, issue the command:

   `LOAD,13,1,`*`filename`*`,W.`

If the tape is to be mounted ring out. issue the command:

`LOAD,13,1,`*`filename`*

where *filename* is the name of the encapsulated tape file.

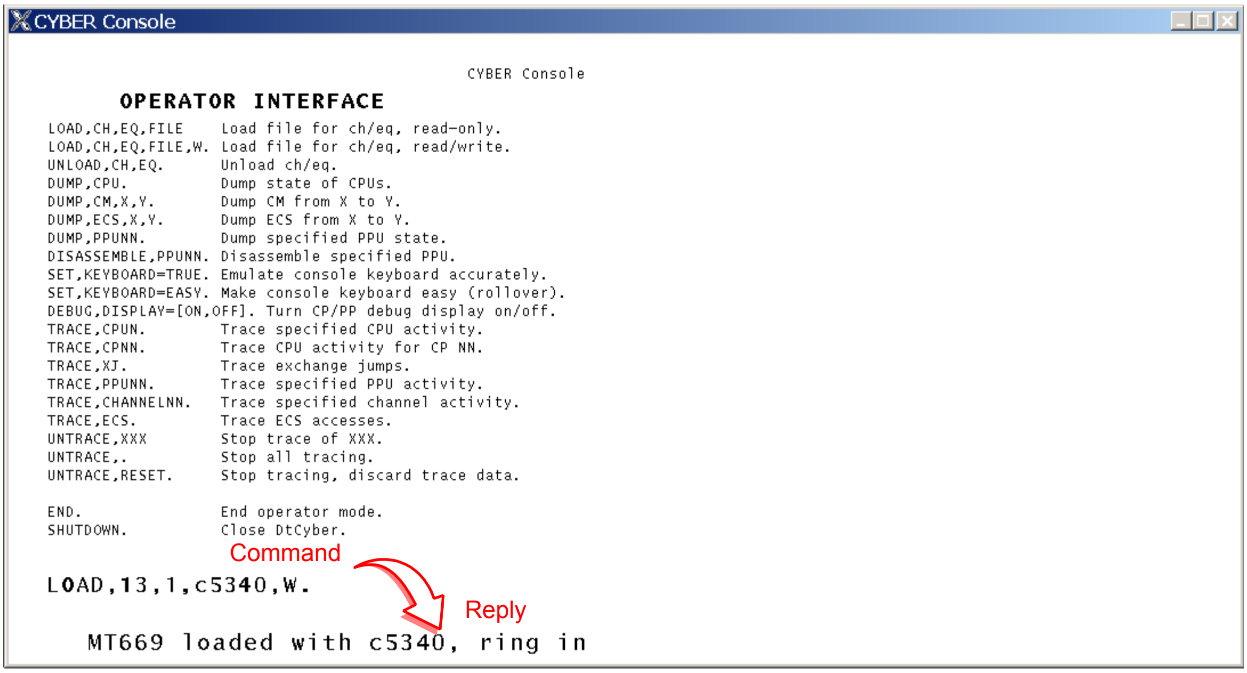
NOTE

Do not type a period at the end of *filename* unless the file name really does have a final period in its name.

If the file name itself contains a comma, rename it so that it does not.

See Figure 2-21 on page 2-44.

FIGURE 2-21. Desktop Cyber Operator Interface



```
                                          CYBER Console
        OPERATOR INTERFACE
LOAD,CH,EQ,FILE    Load file for ch/eq, read-only.
LOAD,CH,EQ,FILE,W. Load file for ch/eq, read/write.
UNLOAD,CH,EQ.      Unload ch/eq.
DUMP,CPU.          Dump state of CPUs.
DUMP,CM,X,Y.       Dump CM from X to Y.
DUMP,ECS,X,Y.      Dump ECS from X to Y.
DUMP,PPUNN.        Dump specified PPU state.
DISASSEMBLE,PPUNN. Disassemble specified PPU.
SET,KEYBOARD=TRUE. Emulate console keyboard accurately.
SET,KEYBOARD=EASY. Make console keyboard easy (rollover).
DEBUG,DISPLAY=[ON,OFF]. Turn CP/PP debug display on/off.
TRACE,CPUN.        Trace specified CPU activity.
TRACE,CPNN.        Trace CPU activity for CP NN.
TRACE,XJ.          Trace exchange jumps.
TRACE,PPUNN.       Trace specified PPU activity.
TRACE,CHANNELNN.   Trace specified channel activity.
TRACE,ECS.         Trace ECS accesses.
UNTRACE,XXX        Stop trace of XXX.
UNTRACE,.          Stop all tracing.
UNTRACE,RESET.     Stop tracing, discard trace data.

END.               End operator mode.
SHUTDOWN.          Close DtCyber.

LOAD,13,1,c5340,W.

    MT669 loaded with c5340, ring in
```

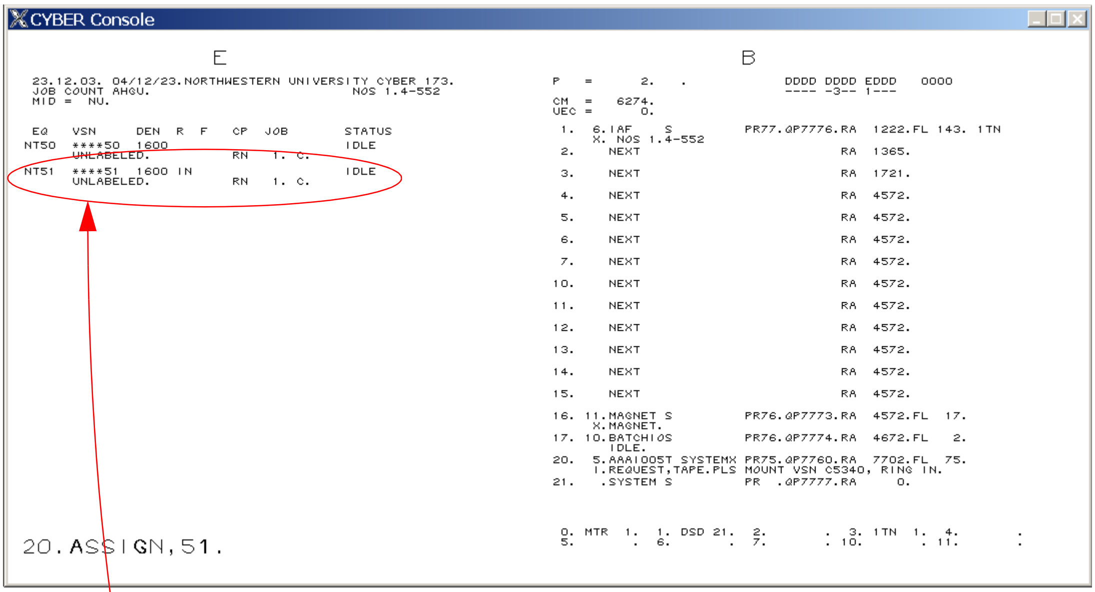
5. Issue the command `END.` to return to Desktop Cyber's emulation mode.

6. Now you must manually assign the tape to the job whether the tape is labeled or not. Assuming you mounted the tape on NT51, the command to do this is: *n*`.ASSIGN,51.` where *n* is the control point number where the tape request is flashing (in our example, 20). See *E,T and B Displays Before and After Assignment* on page 2-45.

NOTE

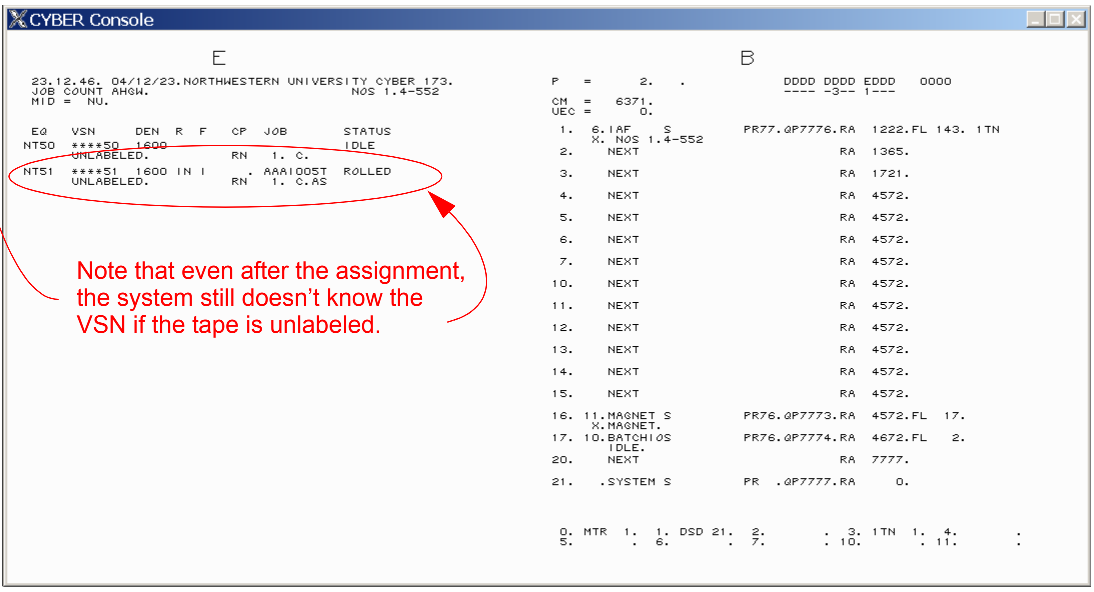
Users may also use REQUEST to make requests unrelated to tape devices. If you merely wish to acknowledge the request without assigning a device, you may assign null equipment (device 77) to satisfy the REQUEST.

FIGURE 2-22. E,T and B Displays Before and After Assignment



```
CYBER Console

                    E
 23.12.03. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AHGU.                          NOS 1.4-552
 MID =  NU.

 EQ    VSN      DEN  R  F    CP   JOB        STATUS
NT50   ****50   1600                         IDLE
       UNLABELED.            RN    1.  C.
NT51   ****51   1600 IN                      IDLE
       UNLABELED.            RN    1.  C.




20.ASSIGN,51.
```

```
                    B
P    =       2.    .            DDDD DDDD EDDD   0000
                                ---- -3-- 1---
CM   =    6274.
UEC  =       0.
 1.   6.IAF     S        PR77.QP7776.RA  1222.FL 143.  1TN
      X.  NOS 1.4-552
 2.     NEXT                         RA  1365.
 3.     NEXT                         RA  1721.
 4.     NEXT                         RA  4572.
 5.     NEXT                         RA  4572.
 6.     NEXT                         RA  4572.
 7.     NEXT                         RA  4572.
10.     NEXT                         RA  4572.
11.     NEXT                         RA  4572.
12.     NEXT                         RA  4572.
13.     NEXT                         RA  4572.
14.     NEXT                         RA  4572.
15.     NEXT                         RA  4572.
16.  11.MAGNET S         PR76.QP7773.RA  4572.FL  17.
      X.MAGNET.
17.  10.BATCHIOS         PR76.QP7774.RA  4672.FL   2.
        IDLE.
20.   5.AAAI005T SYSTEMX PR75.QP7760.RA  7702.FL  75.
      I.REQUEST,TAPE.PLS MOUNT VSN C5340, RING IN.
21.    .SYSTEM S         PR  .QP7777.RA     0.

 0. MTR  1.  1. DSD 21.  2.      .  3. 1TN  1.  4.      .
 5.       .  6.       .  7.      . 10.       . 11.      :
```



```
CYBER Console

                    E
 23.12.46. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AHGW.                          NOS 1.4-552
 MID =  NU.

 EQ    VSN      DEN  R  F    CP   JOB        STATUS
NT50   ****50   1600                         IDLE
       UNLABELED.            RN    1.  C.
NT51   ****51   1600 IN I     .  AAAI005T    ROLLED
       UNLABELED.            RN    1.  C.AS
```

```
                    B
P    =       2.    .            DDDD DDDD EDDD   0000
                                ---- -3-- 1---
CM   =    6371.
UEC  =       0.
 1.   6.IAF     S        PR77.QP7776.RA  1222.FL 143.  1TN
      X.  NOS 1.4-552
 2.     NEXT                         RA  1365.
 3.     NEXT                         RA  1721.
 4.     NEXT                         RA  4572.
 5.     NEXT                         RA  4572.
 6.     NEXT                         RA  4572.
 7.     NEXT                         RA  4572.
10.     NEXT                         RA  4572.
11.     NEXT                         RA  4572.
12.     NEXT                         RA  4572.
13.     NEXT                         RA  4572.
14.     NEXT                         RA  4572.
15.     NEXT                         RA  4572.
16.  11.MAGNET S         PR76.QP7773.RA  4572.FL  17.
      X.MAGNET.
17.  10.BATCHIOS         PR76.QP7774.RA  4672.FL   2.
        IDLE.
20.     NEXT                         RA  7777.
21.    .SYSTEM S         PR  .QP7777.RA     0.

 0. MTR  1.  1. DSD 21.  2.      .  3. 1TN  1.  4.      .
 5.       .  6.       .  7.      . 10.       . 11.      :
```

Note that even after the assignment, the system still doesn't know the VSN if the tape is unlabeled.

Unloading Tapes

Any time the tape unit status is listed as *idle* in the E,T display, the unit may be unloaded via the UNLOAD command (see the UNLOAD command, page 2-17). Usually this isn't necessary, though, since most jobs automatically unload the tape unit when they're done with the tape or at the end of the job.

If the job is of system origin or if the user requested via a control statement option to inhibit automatic unload, the tape will remain loaded but unassigned to a job, and if you need the unit later, you will have to explicitly UNLOAD the device.

Tapes cannot be unloaded in the Desktop Cyber Operator Interface; tapes can only be unloaded from DSD or by the job (either by command or at program termination).

BATCH JOBS

Users may submit batch input jobs through any of the following mechanisms:

- Building a text file (typically from an interactive session) which resembles a batch card deck, and then releasing the file into the input queue with the ROUTE control statement, specifying a disposition code (DC) of either IN or NO. Specifying DC=IN will cause the job output to be printed at job termination; DC=NO causes the job output to be discarded at the end of the job.

- Building a text file as in the previous bullet, but submitting it with SUBMIT instead of ROUTE. SUBMIT offers the advantage of allowing the user, with appropriate directives, to simplify somewhat the input file structure and to omit his or her password in the file.

- Building a text file offline using any familiar text editor (such as Microsoft Notepad) and submitting it via email from anywhere on the Internet and having the output returned by email.

- Building a text file on the host and submitting it via the *create_interstate_connection* command.

- Building a text file on the host and submitting it via the LOAD Desktop Cyber Operator Interface command.

Regardless of how the job enters the system, as soon as it enters the input queue, it is assigned a File Name Table (FNT) ordinal, which it keeps throughout the job's duration as it moves through the input, rollout, and output queues.

Batch jobs do not require operator intervention very often. Possibilities—rare though they be—include:

- The simplest thing a batch job can ask the operator to do is simply to give the job a GO (see page 2-15).

  The COBOL5 and FTN5 compilers, and the COMPASS assembler, offer easy ways for the programmer to place messages on the B display and then simply wait for the operator's permission to proceed. If it isn't a good time to run the job, the operator may, according to the user's instructions, either drop the job, roll it out (via the ROLLOUT command, page 2-16) indefinitely until the appropriate time, or, if the job load is light, just let it sit there at a control point until the time is right.

- A user may request the operator to set sense switches for the job to specify program options. These aren't real switches (although on computers from other manufacturers of the era, they actually were a literal bank of toggle switches on the front panel), but rather are virtual switches peculiar to each job rather than to the entire computer system. Users may set and query these switches via control statements and library routines as well, so operator intervention is usually not required.

- User programs can place short messages on the A and B displays and solicit simple operator input via the CFO command (COBOL5 in particular provides a native mechanism for this without having to call external routines, but programs written in other languages can do so by calling COMPASS routines).

- More complex interactions with the operator may be made through the K display. The program will alert the operator by placing the message `REQUEST *K* DISPLAY` on the B display.

The most common thing you might be expected to do for a batch job is to terminate it, though. Some reasons for terminating a job:

- The user asks you to. NOS 1 does not provide users with a means for terminating their own jobs, so they rely upon the operations staff to do it for them.

- The job is requesting a tape which doesn't exist.

- The job is giving some other part of the system fits. This isn't as common as you might think; the system does a pretty good job of compartmentalization.

How to Terminate a Batch Job

- If the job is executing (that is, you see it on the B display), use the DROP command (see page 2-14). This is a maskable signal and will allow the job to perform cleanup before exiting. If the job does not respond to a DROP, you may use the KILL command (see page 2-15), which is not maskable. To terminate subsystems (IAF, MAGNET, or BATCHIO), use the IDLE command (see page 2-15).

- If the job is rolled out, it must be rolled in before it can be dropped. This rollin can be done explicitly by the operator or implicitly through the use of the PURGE command.

  Unless the job is in an event wait (such as waiting for a tape mount), you may roll the job in with the ROLLIN command (see page 2-16). Once rolled in, you may issue a comment to the job's dayfile explaining why the job was dropped (see the COMMENT command, page 2-14). Finally, use the DROP command (page 2-14) and, if ineffective, the KILL command (page 2-15).

  If the job is in an event wait, or if you don't care to issue a diagnostic message to the job's dayfile, you may use the PURGE command. Consult the E,T (if a tape job) or Q displays to find the job's FNT ordinal and then use the PURGE command (see page 2-16) to drop the job. PURGE rolls the job in briefly just so it can be dropped.

INTERACTIVE JOBS

You will typically need to do only three things for or to an interactive job:

- Log the session off.

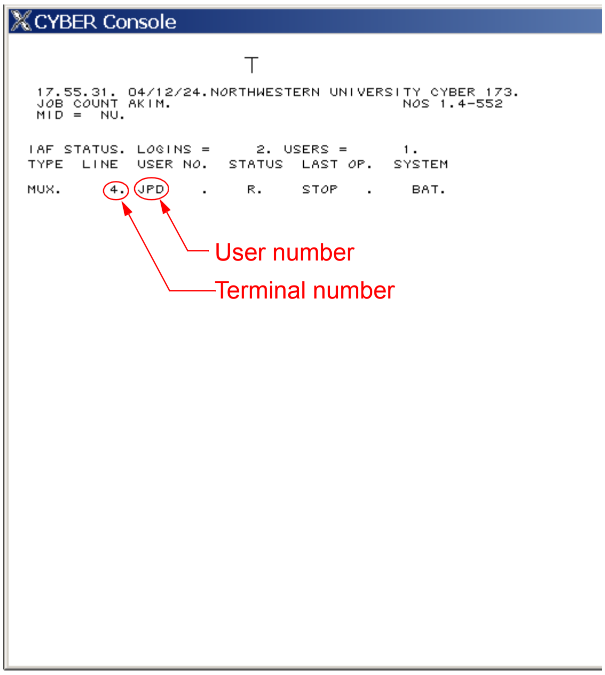
- Report the terminal number of a session so the user may recover the session following an accidental disconnection. (This information is provided to the user at login, but most users ignore it.)

- Place a message on one or more terminals.

How to Log a Session Off

1. Look at the T display and find the user number. The terminal number is listed in the *line* column. See 2-23 on page 2-48.

FIGURE 2-23. Using the T Display



```
                        T

 17.55.31. 04/12/24.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT AKIM.                          NOS 1.4-552
 MID =  NU.

IAF STATUS. LOGINS =     2. USERS =      1.
TYPE  LINE  USER NO.  STATUS  LAST OP.  SYSTEM

MUX.     4. JPD     .     R.    STOP   .    BAT.
```

2. Use the LOGOFF command with the terminal number to log that user off.

How to Determine the Terminal Number of an Interactive User

Simply consult the T display. See Figure 2-23.

How to Send a Message to One or More Terminals

- If you wish to send a message to a single terminal, use the T display to determine the terminal number, and use the DIAL command (page 2-14) to send the message. The message will be displayed immediately on the terminal.

- If you wish to send a message to all users and new users as they log in, use the WARN command (page 2-17). Users will receive the message as soon as their jobs become idle or right after they log in.

- If you wish to send a message to users which they will see *before* they log in, use the MESSAGE command (page 2-15). Users already logged in will not see the message.

OUTPUT JOBS

Desktop Cyber is concerned with two types of output jobs: print and punch.

Print Jobs

Desktop Cyber handles print jobs four ways in the order listed:

- Jobs routed with a forms code of EM (FC=EM) are emailed to the owner of the hash code registered in Desktop Cyber's print manager. Embedded output directives, if any, are ignored. See ***[Setting Up the Print System Xref]*** for details.

NOTE

To submit a job for automatic emailing, the user must be validated for special forms mounts (see CSFC, page 2-33).

- Output from batch jobs originating via email will be returned via email to the sender.

- Jobs with embedded output directives will be processed according to those directives. See ***[Setting Up the Print System Xref]*** for details.

- Jobs without any embedded output directives are printed on the host's default printer.

On true Cybers, there are all sorts of commands available to operators to control output by doing things such as reprinting parts of the job, terminating the current output job, and changing the form characteristics (such as lines per page) on the fly. Since Desktop Cyber isn't printing directly, but rather going through the host's native print services, most of these features don't have any relevance, since by the time you discover a problem, Desktop Cyber is already out of the loop. Thus, most printing issues will be addressed with the host's operating system rather than with BATCHIO on Desktop Cyber.

Handling special forms mounts must still be handled by the Desktop Cyber operator, though. Users with appropriate validation may route output jobs with a forms code specifying a specific type of paper to be mounted on the printer. If no printer has the requested form mounted, the job will wait in the output queue until the specified form is mounted or the job is purged (See the PURGE command, page 2-16).

We suggest that if your users need special forms mounted that they email the output back to themselves and print it themselves rather than request special forms mounts.

The FORM command (see page 2-15) is used to change the forms code mounted on a specific printer.

NOTE

Never change the forms code on LP63. That printer has its forms code set at EM automatically during dead start and if its forms code is changed, output will not be automatically emailed.

Punch Jobs

The Desktop Cyber punch queue is connected to a back end process on the host which is used for processing emulator directives. These directives are coded one per line in a local file which the user subsequently routes to the punch queue with a specified forms code. The directives are described in ***[Tapе Mounting for Users Xref]***. One purpose of this mechanism is to provide a means for allowing properly-validated users to mount their own tapes without the intervention of an operator. This is especially helpful when a large number of tapes have been encapsulated and the user wishes to experiment to recover the tape format.

To access this service, the user must be validated for punching cards (see the CP validation limit on page 2-30), disposing of files with special forms codes (see CSFC permission, page 2-33) and have permission to use the DSD command (see the CDSD permission, page 2-34) to ensure that the desired tape unit is not in use. You should take care that only responsible users are given this permission.

FUN AND GAMES

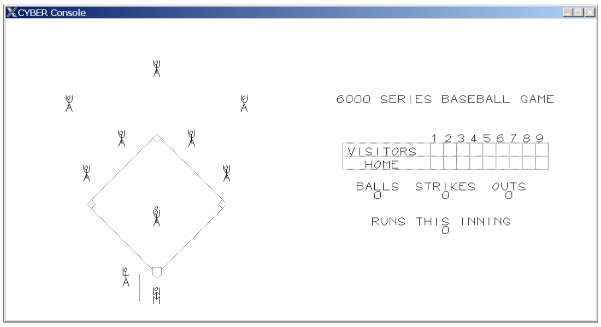
Try these diversions which can only be run from the console:

- 6000 Series Baseball. This was certainly one of the first video games ever. Start it with `X.BAT.` To pitch, press `P`; to swing press `S`. You want the bat to pass over the plate at the same time the ball does. You'll be able to figure out the rest. If the operations manager is hovering close by, * will suspend the display and you'll look hard at work. Pressing * again brings it back where you left off. If the ball moves preposterously fast, you need to increase the CPU ratio in *cyber.ini* so that the CPU gets stepped more often than the PPs. See ***[Adjusting the CPU Ratio Xref]*** for details.

NOTE

There is a 40-year-old bug in BAT which, under certain circumstances, causes the bat to lock up when swung twice in one pitch. While it makes sense to disallow multiple swings on one pitch, sometimes this bug disallows *all* subsequent swings. To avoid hitting this bug, just remember never to try to swing at a pitch more than once.

FIGURE 2-24. Baseball



```
6000 SERIES BASEBALL GAME

            1 2 3 4 5 6 7 8 9
VISITORS
   HOME

  BALLS   STRIKES   OUTS
    0        0        0

     RUNS THIS INNING
              0
```

- Andy Capp Display. `X.ADC.` `D` terminates the program.

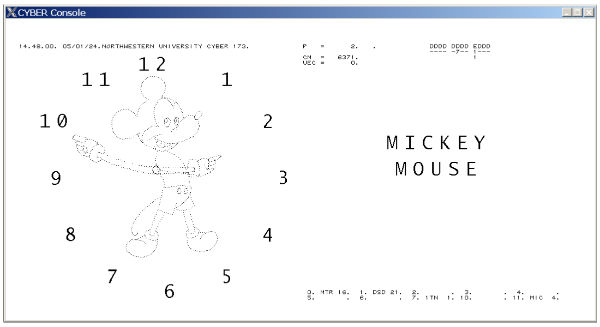
- Worms Display. `X.WRM.` You may enter a numeral from 1 to 6 to control the number of worms. `D` terminates the program.

- Snoopy. `X.DOG.` `D` terminates the program.

- Wink at Operators. `X.EYE.` If you'd like the winking to loop, use `X.EYE(1)` and use `D` to terminate the program.

- Mickey. `X.MIC.` A million-dollar analogue clock. `D` terminates the program.

FIGURE 2-25. Mickey

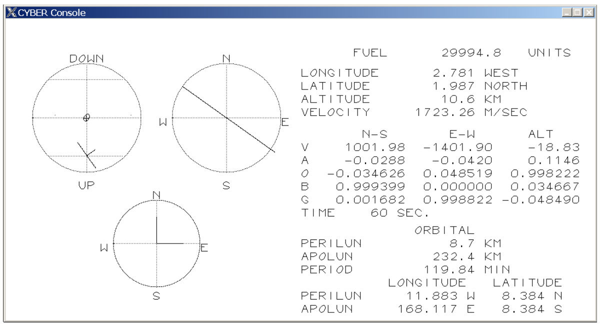


```
14.48.00. 05/01/24.NORTHWESTERN UNIVERSITY CYBER 173.      P   =      2.   .          DDDD DDDD EDDD
                                                                                   ---- -7-- 1---
                                                           CM  =   6371.                      1
                                                           UEC =      0.

                                                                MICKEY
                                                                MOUSE

0. MTR 16.   1. DSD 21.   2.       .  3.          .  4.       .
5.         . 6.         . 7. 1TN  1. 10.          : 11. MIC  4.
```

- Snake. `X.SNK.` Draws several walls on the left display and releases a snake which travels around the display and deflects at a 90º angle each time it hits a wall or edge of the display. Some walls (but not all of them) move a bit when hit. You may enter a numeral from 1 to 9 to control the number of snakes. Surprisingly addictive. `D` terminates the program. Not to be confused with the BSD UNIX game by the same name.

- Lunar. Written at Northwestern University, this is one of the first graphical flight simulators. It exists as indirect access file LUNAR in UN=CALLPRG. You'll want to start it from DIS. `K` terminates the program.

FIGURE 2-26. Lunar



```
          FUEL        29994.8   UNITS
LONGITUDE       2.781 WEST
LATITUDE        1.987 NORTH
ALTITUDE         10.6 KM
VELOCITY      1723.26 M/SEC
          N-S        E-W        ALT
V     1001.98   -1401.90     -18.83
A     -0.0288    -0.0420     0.1146
O   -0.034626   0.048519   0.998222
B    0.999399   0.000000   0.034667
G    0.001682   0.998822  -0.048490
TIME    60 SEC.
                ORBITAL
PERILUN           8.7 KM
APOLUN          232.4 KM
PERIOD         119.84 MIN
             LONGITUDE   LATITUDE
PERILUN       11.883 W   8.384 N
APOLUN       168.117 E   8.384 S
```

- 99 Bottles. Written by Control Freak Paul Koning, *NI1D*, this PP program provides step-by-step instructions on how to have a very bad morning. It was written because, well, it just had to be done. The source code exists as indirect access file BOTPP in UN=JPD. You'll want to assemble and SYSEDIT it onto the system from DIS, and then run it from the console with `X.BOT.`

  Try building it and running it yourself by following the instructions below.

BUILDING AND RUNNING 99 BOTTLES

1. From DSD, give the command

   ```
   X.DIS.
   ```

   This starts DIS at any available control point.

2. Issue the command

```
SUI,1.
```

since you may not issue any permanent file commands until the user index is set.

3. Issue the command

   ```
   GET,BOTPP/UN=JPD.
   ```

   This makes a local copy of indirect access file BOTPP from user number JPD.

   See ***[Indirect Access Files Xref]*** for more information on indirect access files.

4. Attach the system program library with

   ```
   ATTACH,OPL=WPL/UN=SYSLIB.
   ```

   This is the program library with the operating system source code, as modified by MECC. See ***[Direct Access Files Xref]*** for more information on direct access files.

5. Assemble BOTPP with the command

   ```
   COMPASS,I=BOTPP,S=PPTEXT.
   ```

   This causes COMPASS to read BOTPP as the input source, use PPTEXT as a macro source text, generate binary output on LGO (since B= was omitted, it defaults to LGO) and generate the assembler listing on file OUTPUT (since L= was omitted, it defaults to OUTPUT).

   See ***[Common Default Parameters Xref]*** for more information on the consistency across products of default parameter values.

6. If you would like to inspect the list output, go into O26 and issue the command `READ.OUTPUT.` Then use the `(` and `)` keys to scroll through the output listing. Use the `XDIS.` command to get back to DIS.

   See *DIS Operation* on page 2-18 for more information on how to use DIS and *Using O26* on page 2-22.

7. Now you are ready to SYSEDIT the new PP program onto the system. Rewind file LGO and SYSEDIT the program with these commands:

   ```
   REWIND,LGO.
    SYSEDIT.
   ```

   See ***[Creating a New Dead Start Tape Xref]*** for more information on SYSEDIT.

8. Now, cause DIS to drop the console display:

   ```
   HOLD.
   ```

   NOTE

   You must use HOLD instead of * because you need to drop the display channel to run the BOT program.

9. Start the BOT program with

   ```
   X.BOT.
   ```

   (The name is BOT instead of BOTPP because COMPASS uses the name defined in a pseudoinstruction instead of the file name, since a single file may contain multiple programs. The author chose the name BOT instead of BOTPP because PP programs must have precisely three-character names.)

10. Follow the scrolling instructions printed on the screen, if you dare.

11. When the program finishes, or when you get the idea, terminate the program with `D`.

12. After the program exits, you will see the following message flashing on the B display:

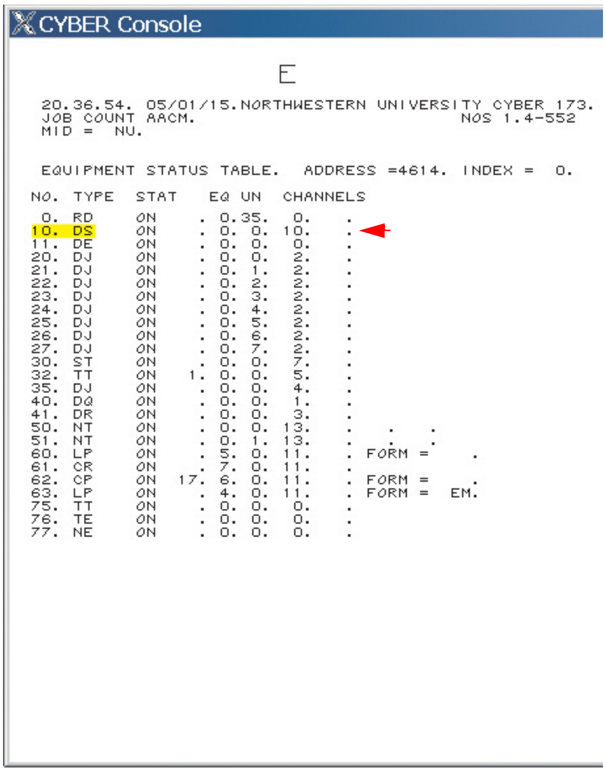
    ```
    REQUEST DISPLAY. (DIS)
    ```

13. Stop for a moment and think what you should do to get back to DIS. *Hint.* Check E,A for the EST ordinal of the display console (DS).

14. E,A shows that the EST ordinal for the DS (display console) equipment is 10. See Figure 2-27 on page 2-52.

    Manually assign the display with the command *n*.`ASSIGN,10.`

FIGURE 2-27. Finding the EST Ordinal of the Display Console



```
                              E

  20.36.54. 05/01/15.NORTHWESTERN UNIVERSITY CYBER 173.
  JOB COUNT AACM.                            NOS 1.4-552
  MID =  NU.

  EQUIPMENT STATUS TABLE.  ADDRESS =4614.  INDEX =  0.

 NO. TYPE  STAT   EQ UN  CHANNELS
  0. RD    ON    . 0.35.  0.   .
 10. DS    ON    . 0. 0. 10.   .
 11. DE    ON    . 0. 0.  0.   .
 20. DJ    ON    . 0. 0.  2.   .
 21. DJ    ON    . 0. 1.  2.   .
 22. DJ    ON    . 0. 2.  2.   .
 23. DJ    ON    . 0. 3.  2.   .
 24. DJ    ON    . 0. 4.  2.   .
 25. DJ    ON    . 0. 5.  2.   .
 26. DJ    ON    . 0. 6.  2.   .
 27. DJ    ON    . 0. 7.  2.   .
 30. ST    ON    . 0. 0.  7.   .
 32. TT    ON   1. 0. 0.  5.   .
 35. DJ    ON    . 0. 0.  4.   .
 40. DQ    ON    . 0. 0.  1.   .
 41. DR    ON    . 0. 0.  3.   .
 50. NT    ON    . 0. 0. 13.   .    .    .
 51. NT    ON    . 0. 1. 13.   .    .    .
 60. LP    ON    . 5. 0. 11.   . FORM =     .
 61. CR    ON    . 7. 0. 11.   .
 62. CP    ON  17. 6. 0. 11.   . FORM =     .
 63. LP    ON    . 4. 0. 11.   . FORM =  EM.
 75. TT    ON    . 0. 0.  0.   .
 76. TE    ON    . 0. 0.  0.   .
 77. NE    ON    . 0. 0.  0.   .
```

15. If you wish to make BOT a permanent part of the system, you will need to generate a new dead start tape. See ***[Creating a New Dead Start Tape Xref]*** for more information.

16. When you are through with DIS, drop it with the command

```
DROP.
```

NOTE

Whenever any job of any origin terminates, normally (unless the user has overridden this behaviour via the ROUTE control statement) the files OUTPUT, PUNCH, and PUNCHB are automatically released to the print, punch, and punch binary queues respectively, provided that those files are local and reside on mass storage. If any of these files are local and you do not wish them automatically disposed to the respective output queue at job termination, you should issue a CLEAR command before dropping DIS. The CLEAR command releases assignment of all files local to the job and thus prevents their automatic disposition at end of job.