



**DESKTOP CYBER 2.1
OPERATOR'S AND USER'S GUIDE**

**CDC® OPERATING SYSTEMS:
MECC NOS 1.4 PSR 552**

INDEX OF CONSOLE COMMANDS

Command	Mode	Page	Command	Mode	Page
*	DSD	2-15	LOCK	DSD	2-13
■ Account Management	DSD	2-22	LOGOFF	DSD	2-13
ASSIGN	DSD	2-12	MESSAGE	DSD	2-13
AUTO	DSD	2-12	MOVE	O26	2-21
BLITZ	DSD	2-12	O26	DIS	2-18
CFO	DSD	2-12	O26 Special Keys	O26	2-19
CHECK POINT SYSTEM	DSD	2-12	OFF	DSD	2-14
CKP	DSD	2-12	OFFSW	DSD	2-14
COMMENT	DSD	2-12	ON	DSD	2-14
COPY	O26	2-21	ONSW	DSD	2-14
D	O26	2-21	OUT	O26	2-21
DCP	DIS	2-18	OVERRIDE	DSD	2-14
DEBUG	DSD	2-12	P	O26	2-21
DIAL	DSD	2-13	PURGE	DSD	2-14
DIS	DSD	2-13	RCP	DIS	2-18
DIS	O26	2-20	READ	O26	2-20
DIS Displays	DIS	2-16	REWIND	O26	2-20
DISABLE	DSD	2-13	RNR	O26	2-21
DROP	DSD	2-13	RNS	DIS	2-18
DROP	DIS	2-18	ROLLIN	DSD	2-14
DROP	O26	2-20	ROLLOUT	DSD	2-14
DSD Displays	DSD	2-9	ROLLOUT	DIS	2-19
ELS	DIS	2-18	RS	O26	2-21
ENA	DIS	2-18	STEP	DSD	2-14
ENABLE	DSD	2-13	SUI	DIS	2-19
ENB	DIS	2-18	SUN	DIS	2-19
ENP	DIS	2-18	UCC	O26	2-21
ENS	DIS	2-18	UNLOAD	DSD	2-15
ENX	DIS	2-18	UNLOCK	DSD	2-15
FILE	O26	2-20	UNSTEP	DSD	2-15
FORM	DSD	2-13	VSN	DSD	2-15
GO	DSD	2-13	WARN	DSD	2-15
HOLD	DIS	2-18	WRITE	O26	2-21
IDLE	DSD	2-13	X	DSD	2-15
KILL	DSD	2-13	XDIS	O26	2-20
L	O26	2-21	XDROP	O26	2-20
LOAD	Operator Interface	2-41			

PREFACE

INTRODUCTION

Desktop Cyber is an emulator which brings back to life the revolutionary design of Control Data Corporation (CDC) 6000 series and Cyber series mainframes. The first CDC Cybers were designed and built by a small team headed by Seymour Cray at CDC's Chippewa Falls labs in the early 1960s. Desktop Cyber emulates this series of mainframes and peripherals in software.

The software provides emulation of typical 6000 and Cyber 7x and 17x systems including common peripherals such as operator console, tape and disk drives, central memory, extended core storage, card readers and punches, printers, and terminal multiplexers.

Desktop Cyber successfully dead starts and runs the following CDC operating systems: Chippewa OS (COS), SMM, KRONOS 2.1, NOS 1.2, NOS 1.3, NOS 1.4, NOS 2.2, NOS 2.8.1, and NOS/BE 1.5.

This manual contains information necessary to install, configure, operate, and use Desktop Cyber running the Minnesota Educational Computing Consortium (MECC) version of the Control Data Corporation's Network Operating System (NOS) 1.4.

NOTE

While Desktop Cyber runs many CDC operating systems on several hardware platforms, this manual assumes the emulator will be running MECC NOS 1.4 on a Linux host.

AUDIENCE AND ORGANISATION

This document assumes that you are familiar with the host on which Desktop Cyber is being run. No familiarity with the emulator hardware or operating system is assumed.

This manual is organised as follows:

- Section 1 Installation and configuration. This section covers both NOS and host configuration matters. This section is intended primary for the individuals who will be installing the system and maintaining the host/NOS interface, but may not necessarily be involved in running the system or using it on a day-by-day basis.
- Section 2 Operations. This section covers the day-to-day maintenance of the system including setting up user accounts, job management, and managing the host/system interface. The intended audience is individuals who will keep the system running on a daily basis.
- Section 3 Usage. This section covers how to use NOS from the perspective of the end user running interactive and batch origin jobs.

CONVENTIONS

Literal messages displayed on the console (as opposed to a terminal) are shown in Paul Koning's **SEYMOUR** font, which closely resembles the font used on early Control Data display consoles. Terminal sessions are shown in upper case `LIKE SO` with terminal input set from the corresponding lower case.

Command syntax descriptions and keystrokes are shown in upper case `LI KE TH I S` with command parameters (control point, user number, etc.) in italics. Keys with multicharacter names are shown in small caps as in `ENTER`, `ESC`, and `ALT`.

RELATED PUBLICATIONS

This manual is not intended to be exhaustive. We know that setting up and running a mainframe—even when that mainframe is being emulated on a palmtop computer—is daunting if you don't have extensive previous experience with the computer or its operating system, so the purpose of this document is to get you over the initial hump. Other Control Data manuals cover the topics in this manual in greater depth.

The Controlfreaks maintain an extensive online library of publications with the generous permission of CDC's successor, British Telecom Consulting. Here are some of the manuals in the Controlfreaks inventory which you may find useful. Some, such as the

NOS Version 2 Installation Handbook, may not appear to be directly useful since it addresses a later release of NOS, but many of the concepts still apply.

Control Data Publication	Publication Number and Year
NOS Version 1 Reference Manual, vol. 1	60435400J 1979
NOS Version 1 Reference Manual, vol. 2	60445300E 1977
NOS Version 1 Applications Programmers Instant	60436000H 1980
NOS Version 1 System Programmers Instant	60449200G 1981
NOS Version 1 Operator's Guide	60435600M 1980
NOS Version 1 System Maintenance Reference Manual	60455380H 1981
NOS Version 2 Installation Handbook	60459320B 1983
ALGOL 60 Version 3 Reference Manual	60329000D 1973
BASIC 2.1 Reference Manual	19980300B 1974
COBOL Version 5 Reference Manual	60497100H 1980
COMPASS Version 3 Reference Manual	60492600G 1979
FORTRAN 4 Reference Manual	60305600E 1974
Fortran 5 Reference Manual	60481300H 1985
FORTRAN Common Library Mathematical Routines Reference Manual	60498200 1978
Sympl Version 1 Reference Manual	60496400F 1980
Modify Reference Manual	60450100D 1978
Text Editor Reference Manual	60436100 1979
Update Reference Manual	60342500H 1978
Xedit Version 3 Reference Manual	60455730 1978

ACKNOWLEDGEMENTS

Tom Hunter expresses his thanks to the following people who have encouraged him and helped in various ways:

Wanee Hunter	John Gibbins
Clare Johnstone	Steve Peltz
David Webb	Dave Mausner
Douglas Quebbeman	Dennis Henriksen
Jeff Woolsey	Jeffrey Katcher
Walter Spector	Gerard van der Grinten
Tony Epton	Mike Arrington
Paul Repacholi	Joe Cychosz
John Zabolitzky	Tim Smart (BTC)
Freddy Meerwaldt	Barry Murphy (BTC)
James Wiley	Peter Bartsch
John Laird	Phillip Draughon
Kent Olsen	Ken Hunter
Paul Koning	

A special thank you goes to the man who was responsible for the design of Control Data Corporation's most successful large-scale computer, the CDC 6600 system.



Seymour Cray

Seymour Cray was born in 1925 in Chippewa Falls, Wisconsin; he died in 1996 in Colorado Springs, Colorado, from injuries suffered in a car accident.

COLOPHON

The particular and peculiar variation of typefaces and the use of whitespace cannot be preserved by any existing OCR technology. It forms part of the aesthetic that made CDC a thing of interest to me. It's not just about retrocomputing, it's a gestalt.

—Douglas Hurst Quebbeman,
Founder of the Controlfreaks

The look and feel of the Desktop Cyber 2.1 Operator's and User's Guide is based upon the layout of manuals produced by CDC in the late 1970s and early 1980s. Because of the consistency in CDC documentation, experienced users could pick up an unfamiliar CDC manual and quickly locate needed information. CDC documentation appeared in one- and two-column formats depending upon the subject matter. We have chosen to use the two-column format since it looks better with the many short paragraphs in the reference sections.

Most CDC manuals of the time were set with section titles and headings in Linotype Helvetica and body text in Times Roman. Owing to licensing restrictions on the PDF-embedded use of these fonts, we have substituted Arial for Helvetica—the differences between Arial and Helvetica are subtle—and Times New Roman for Times Roman.

Most body text in original CDC documentation was set with six percent letter spacing (like *this* instead of *this*), apparently because adding the letterspacing made it easier, if need be, to later compress the copy and add new material without changing the pagination. However,

typographers today deprecate letterspacing mixed-case copy, and in light of how this document is distributed and reproduced, its original purpose is now an anachronism, so letterspacing is one attribute we have not reproduced.

We've also made the point size a bit larger since the Controlfreaks are a bit older. In doing so, we made the right margin ragged, since space-filling a large point, narrow column looks rather sparse and is difficult to read.

As opposed to American convention, we have adopted international English spelling in this document as a tip of the hat to Desktop Cyber's designer and principal developer, Tom Hunter, who also made substantial text contributions to this document.

This publication was composed and finished in Adobe FrameMaker 7.0 using scanned overlays of actual CDC manuals for page and paragraph layout. Screen captures were done with Nestegg Software Screen Capture 1.4.6 and enhanced for readability with Adobe Photoshop CS and Adobe Illustrator CS.

CONTENTS

Index of Console Commands	Front Cover	Entering Characters not on the Keyboard	2-21
PREFACE	iii	To Create a Text File from DIS	2-22
Introduction	iii	To Browse a File from DIS	2-22
audience and Organisation	iii	Account Management	2-22
Conventions	iv	User Indices	2-22
Related Publications	iv	Vanity Hash Codes	2-23
Acknowledgements	v	Account Creation	2-24
Colophon	v	Closing an Account	2-34
2. BASIC OPERATION	2-1	Job Management	2-34
Introduction	2-1	Job Origin Types	2-34
Console Keyboard Usage	2-1	Tape Jobs	2-35
Dead Start Panel	2-1	How to Handle a Tape	
Dead Start	2-1	Request on the E,P Display	2-37
Shutdown	2-7	How to Handle a Tape	
Console Operation	2-7	Request on the B Display	2-40
DSD Displays	2-8	Unloading Tapes	2-43
DSD Commands	2-12	Batch Jobs	2-44
DIS Operation	2-15	Interactive Jobs	2-45
Auto Mode	2-16	To log a session off	2-45
DIS Displays	2-16	To determine the terminal	
DIS Special Keys	2-17	number of an interactive	
DIS Commands	2-18	user	2-46
Memory Entry	2-19	To send a message to one or	
Using O26	2-19	more terminals	2-46
O26 Special Keys	2-19	Output Jobs	2-46
System Commands	2-20	Print Jobs	2-46
File Commands	2-20	Punch Jobs	2-47
Editing Commands	2-21	Fun and Games	2-48
Miscellaneous Commands	2-21	Bulding and Running 99 Bottles	2-48
		Index of Control Statements	Back Cover

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 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

The left blank clears the text you have entered on the command line (as a control-U in many UNIX shells), and the right blank tabs through several console displays.

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.

DEAD START PANEL

Obviously, there is 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.

DEAD START

The process of initializing Desktop Cyber and its hosted operating system is called *dead start*. Usually, the process is initiated by an operator, but then proceeds without intervention.

Here is how you normally dead start your Desktop Cyber:

1. Log in to the Linux 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

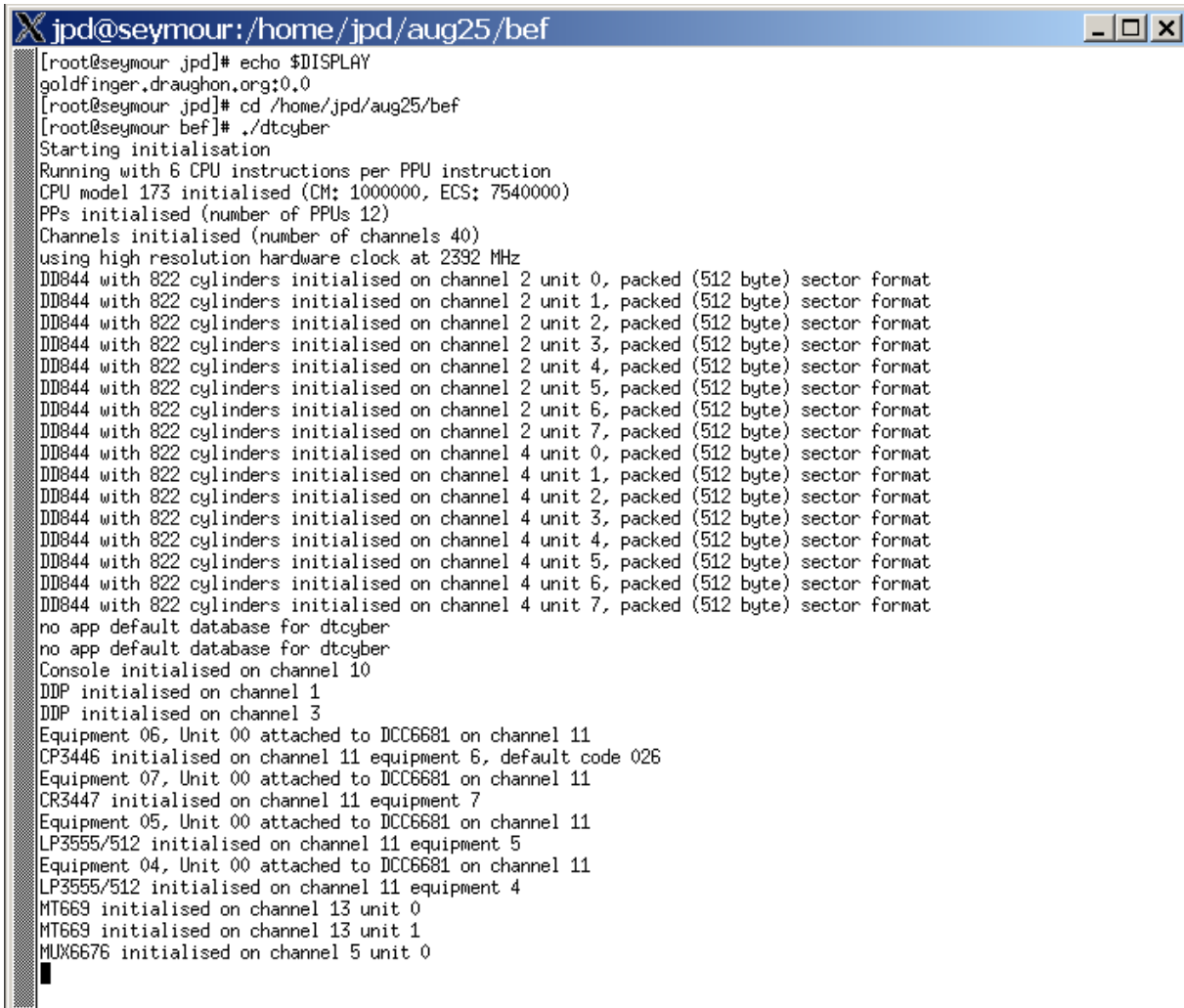
your Linux host name followed by “:0.0” as in *goldfinger.draughon.org:0.0*

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:

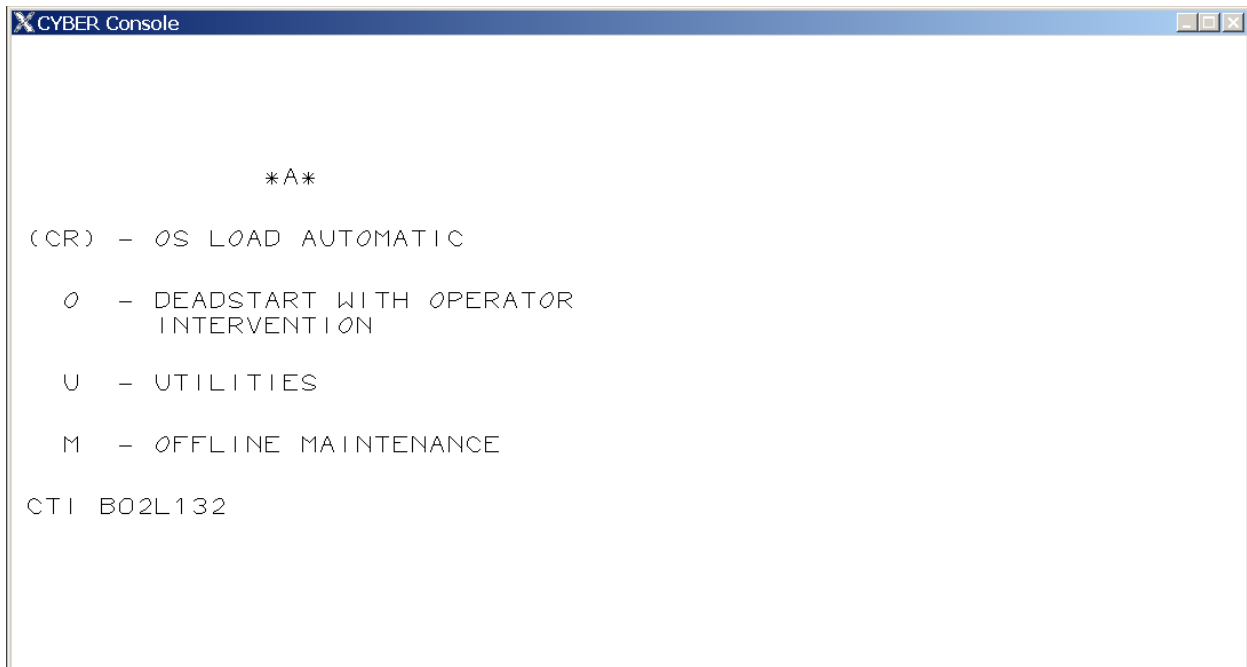
FIGURE 2-1. Starting Desktop Cyber



```
X jpd@seymour: /home/jpd/aug25/bef
[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 PPU's 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 2-2. Initial Options Display



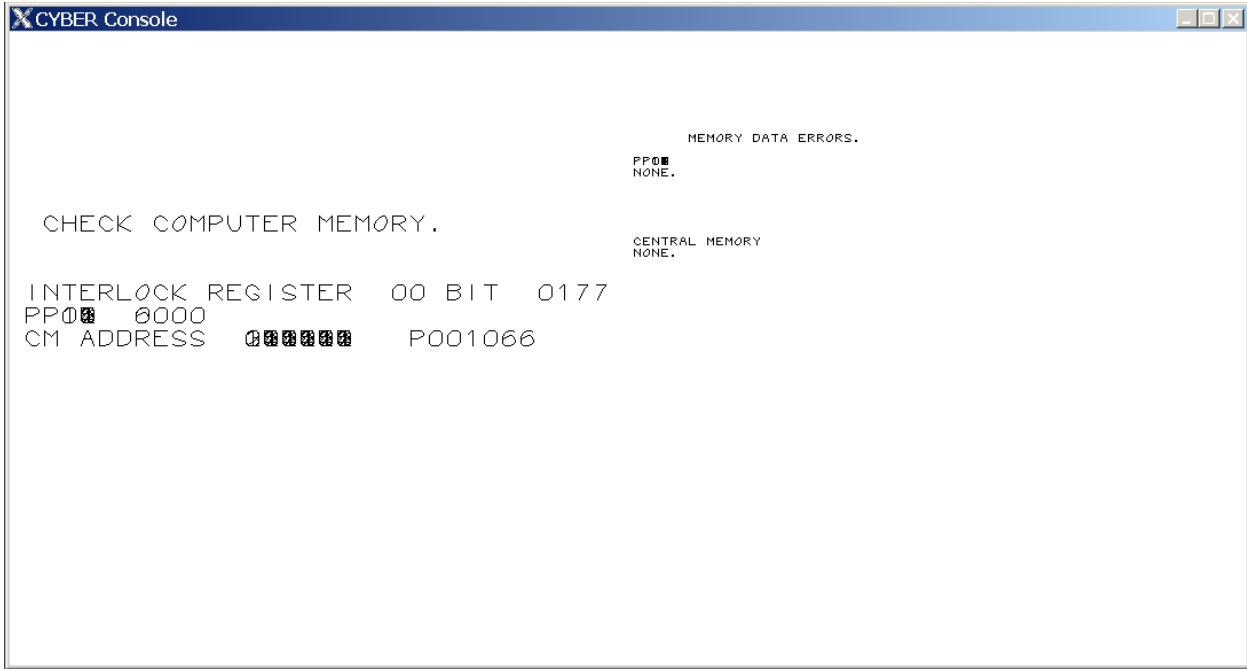
If you wish to accept the default set of dead start options, you may simply press **ENTER** and a level zero dead start will proceed without displaying the configuration decks.

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

NOTE

If you wish to perform a dead start other than level zero, display the configuration decks, or override the dead start panel options, instead of pressing **ENTER**, press **O**, then **P**, and enter your desired options. Once done, press **ENTER** to proceed with the dead start.

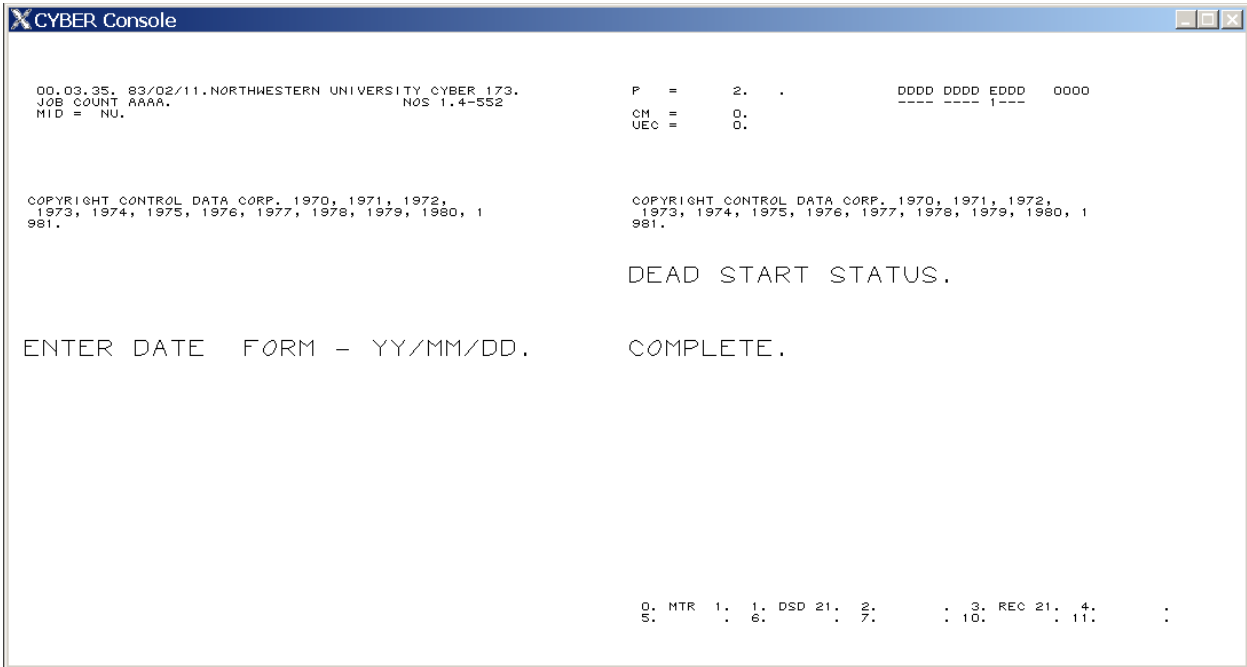
FIGURE 2-3. Initial Tests



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.

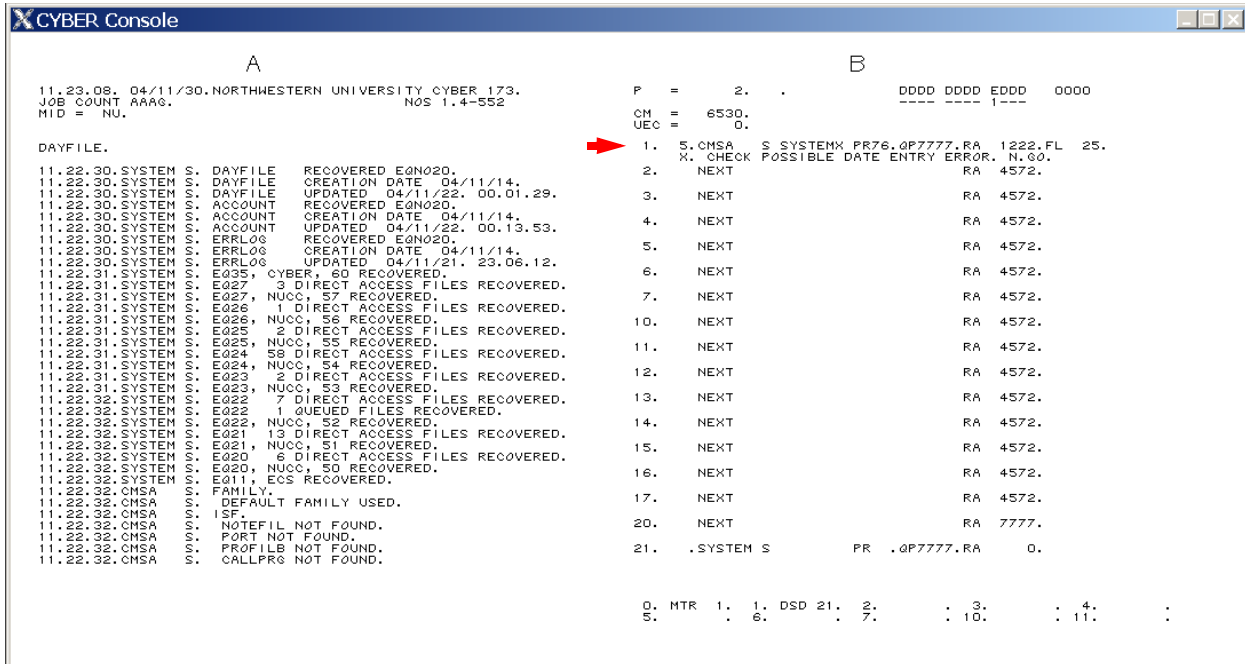
FIGURE 2-4. Date/Time Entry



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

- If the last dead start were more than a few days 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.

FIGURE 2-5. CMS Requesting Date Confirmation



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

9. The rest of the system will now come up, with IAF, MAGNET, and BATCHIO checking in.

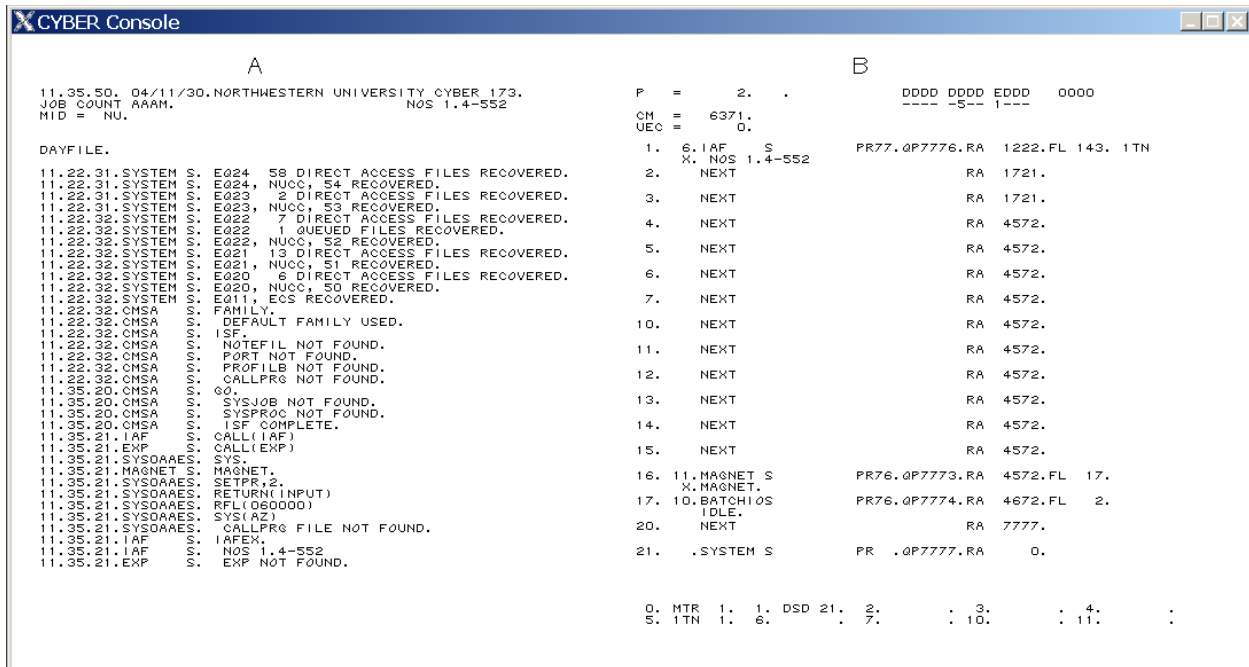
NOTE

It really isn't the IAF you may recall. Actually, it's more like a tricked-up TELEX, so if you are accustomed to NAM IAF, you are in for a gentle re-education. See [*Interactive Use Xref*] use for details. Here, though, is a brief summary of what you need to know about the subtle, yet significant, differences:

- To terminate a running job, type STOP ENTER.
- To terminate a running job which is issuing output, just type S while the output is being printed.
- To interrupt a running job which is issuing output, just type I while the output is being printed. Press ENTER to continue.
- To cancel a line of input you have entered, press ESC.
- To exit text mode, type CONTROL + C.

To obtain a job status report at any time other than when the job is requesting input, type E ENTER.

FIGURE 2-6. Display Following Successful Dead Start



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

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

SHUTDOWN

To shut down Desktop Cyber:

1. If users are logged on, you may wish to warn them. See the WARN command on page 2-15 and the DIAL command on page 2-13.
2. If the console keyboard is locked (see Figure 2-7 on page 2-11), issue the UNLOCK command.
3. Issue the CHECK POINT SYSTEM command.
4. When the message **SYSTEM CHECK-POINT COMPLETE.** appears at the SYS control point, issue the command STEP.

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.

Most DSD commands must end with a period. Commands which place messages somewhere (such as in a dayfile or on a user's terminal) do not have to terminate with periods.

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 are selected 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.

Some displays (A, C, D, E, F, G, and K) have subdisplays which can be activated on the left side with the command x, s , where x is the display and s the subdisplay. For example, E, T. brings up the tape status display on the left side.

Here is a review of the operator console displays:

TABLE 2-1. DSD Displays

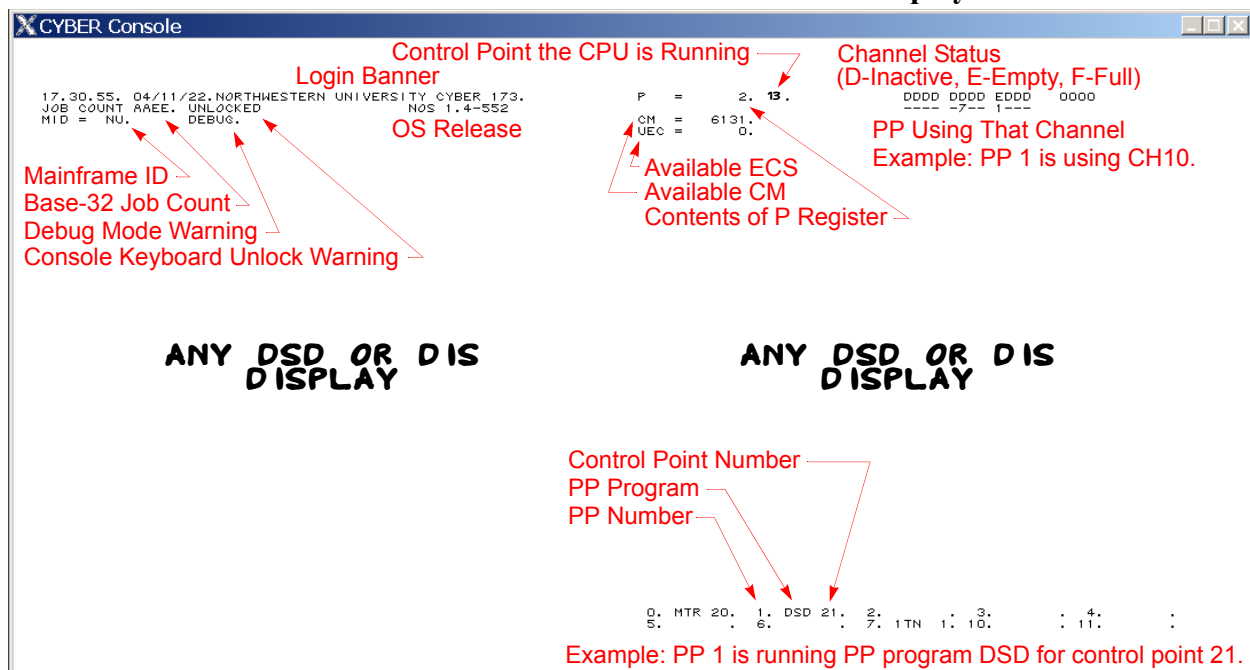
Display	Purpose
A	<p>Displays system dayfile, account file, and error log.</p> <p>A, O. Shows the system dayfile.</p> <p>A, <i>n</i>. Shows the dayfile for the job at control point <i>n</i>.</p> <p>A, ACCOUNT FILE. Shows the account dayfile.</p> <p>A, ERROR LOG. Shows the system error log.</p>
B	<p>Displays control point activity.</p>
C, D	<p>Displays central memory contents. Each display shows 40B (the B means octal, remember?) 60-bit words at a time which you can page through with the + and - keys if you have the display on the left, or the (and) keys if on the right. Each word is shown in five groups of 12 bits. The only difference between the C and D displays is that the C display begins at location 0B and D begins at 40B. Thus, you may display the first 100B locations of central memory by bringing up the C and D displays simultaneously with the command CD.</p> <p>The C and D displays may also be used to show central memory relative to a control point. In the discussion below, C may also be any of the central memory displays (C, D, F, or G):</p> <p>C, O. Returns to the absolute address display.</p> <p>C, <i>n</i>. Shows central memory relative to control point <i>n</i>.</p>
E	<p>Equipment status tables. The E display is actually a group of five separate displays:</p> <p>E, A. Equipment status table.</p> <p>E, C. Mass storage configuration.</p> <p>E, M. Mass storage table.</p> <p>E, P. Pending tape requests.</p> <p>E, T. Tape drive status.</p>
F, G	<p>Displays central memory contents. Just like the corresponding C and D displays except each word is shown in four groups of 15 bits.</p>

Display	Purpose
H	<p>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—</p> <p>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.</p>
	<p>Consult the <i>NOS Version 1 Systems Programmer's Instant</i>, page 1-2 (included in your distribution) for a full listing of the file types.</p>
I	<p>BATCHIO status. Shows the status of unit record equipment (card readers, punches, and printers).</p>
J	<p>Control point status. Shows the status of a specified control point. You specify the control point with <i>J, n</i>, where <i>n</i> is the control point number.</p>
K, L	<p>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:</p>
	<pre>20. S. MODVAGUS PR70.QP7761.RA 7S37 FL 240. X. REQUEST *K* DISPLAY.</pre> <p>The operator assigns the K display with <i>K, n</i>, where <i>n</i> is the control point number (20, in this case) as in <i>K, 20</i>. The operator may now type in data for that program by preceding it with <i>K</i>, as in <i>K, C, JPD</i> (which is the MODVAL K display command to create a new user JPD).</p>
M	<p>ECS display. Shows contents of extended core storage in the same manner as the C display shows for central memory.</p>
N	<p>File display. You may use this display to show the contents of any active file. Use the <i>DI SPLAY, fntord</i>. command (where <i>fntord</i> is the FNT ordinal of the file from the H display) to show the specified file on the N display.</p>
O	<p>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.</p>

Display	Purpose
P	PP communications area. For each peripheral processor, the following is listed: first line, input register; second line, output register; third line, first word of message buffer.
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.
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 2-7. Status on all DSD and DIS Displays



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.

ASSIGN *n*. ASSI GN, *dd*.
 Assigns device *dd* to the job. The job must be requesting a device (such as a tape mount) via the REQUEST command for this to do anything; this command is not used to satisfy requests for tape mounts requested by the LABEL control statement; see *How to Handle a Tape Request on the B Display* on page 2-40.

AUTO AUTO.
 Instructs the system to start the services which have been previously enabled (see the ENABLE command on page 2-13).

BLITZ BLI TZ.
 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* 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 POI NT SYSTEM.
 Prepares the system for immediate shutdown. The console must be unlocked.

CKP *n*. CKP.
 Checkpoints the job and starts it over again.

COMMENT *n*. COMMENT. *comment*
 Places *comment* in the job's dayfile.

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	DI AL, <i>t</i> , <i>message</i> Sends <i>message</i> to terminal number <i>t</i> . <i>See also</i> : Use the T display to determine which terminal a user is on.		VALIDATION Requires verification of mass storage tables.
DIS	<i>n</i> . DI S. Takes the job at the control point into DIS. <i>See DIS Operation</i> on page 2-15. <i>X</i> . DI S. Starts a new system origin job at any available control point and starts DIS.	FORM	FORM <i>nn</i> , <i>fc</i> . Tells the system that the special forms indicated by <i>fc</i> are now mounted on printer with EST ordinal <i>nn</i> . Use E, A. to see a list of equipment and associated EST ordinals. FORM <i>nn</i> ,. Clears the forms code on printer with EST ordinal <i>nn</i> .
DISABLE	DI SABLE, <i>something</i> . Stops a subsystem or disables an option. <i>See ENABLE</i> .	GO	<i>n</i> . GO. Resumes execution of a paused job.
DROP	<i>n</i> . DROP. Drops the job, but allows EXIT error processing.	IDLE	<i>n</i> . I DLE. Asks the subsystem running at that control point to finish what it's doing and exit.
ENABLE	ENABLE, <i>something</i> . 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. SECONDARY USER CARDS Allows users to issue additional USER cards once they've validated themselves initially to the system (akin to UNIX <i>su</i>). TELEX Allows automatic startup of interactive access system.	KILL	<i>n</i> . KI LL. Drops the job without EXIT error processing.
		LOCK	LOCK. Locks the console to prevent accidentally entering dangerous commands.
		LOGOFF	LOGOFF, <i>t</i> . Logs off the interactive terminal number <i>t</i> . <i>See also</i> : The T display shows the terminal number for each user.
		MESSAGE	MESSAGE, <i>message</i> . Changes the login banner to <i>message</i> . Users will see this message <i>before</i> they log in, and users already logged in will not see it at all.

	<i>See also:</i> WARN to display a message <i>after</i> the user has logged in.	ROLLIN	ROLLIN, <i>xxx</i> . Assigns job with FNT ordinal <i>xxx</i> maximum queue priority, making it immediately eligible for being scheduled to a control point.
OFF	OFF <i>nn</i> . Used to logically turn off equipment with EST ordinal <i>nn</i> .		<i>See also:</i> You may determine the FNT ordinal of a job by looking at the Q display.
OFFSW	<i>n</i> .OFFSW, <i>x</i> . Turns off logical sense switch <i>x</i> ($1 \leq x \leq 6$) for reference by the job.	ROLLOUT	<i>n</i> . ROLLOUT. Removes job currently assigned to control point <i>n</i> and places it in the rollout queue. The job will remain rolled out until explicitly rolled back in.
ON	ON <i>nn</i> . Used to logically turn on equipment with EST ordinal <i>nn</i> .		<i>n</i> . ROLLOUT, <i>xxxx</i> . Removes job currently assigned to control point <i>n</i> and places it in the rollout queue for <i>xxxx</i> seconds, at the end of which it is eligible for rescheduling to a control point.
ONSW	<i>n</i> .ONSW, <i>x</i> . Turns on sense switch <i>x</i> ($1 \leq x \leq 6$) for reference by the job.		
OVERRIDE	<i>n</i> . OVERRI DE. Removes the job from the executing job table. The console must be unlocked; see the UNLOCK command.	STEP	<i>n</i> . 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.
PURGE	PURGE, <i>nnn</i> . Purges job with FNT ordinal <i>nnn</i> 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). <i>See also:</i> 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.		

	STEP.	X	X. <i>cmd(opts)</i> Starts a system origin job to run the specified command <i>cmd</i> with the specified options <i>opts</i> .
UNLOAD	UNLOAD, <i>xx</i> . Unloads the tape on unit <i>xx</i> . The tape must not be assigned to a job.		X. <i>cmd</i> . Starts a system origin job to run the specified command <i>cmd</i> .
UNLOCK	UNLOCK. Unlocks the console to permit the entry of potentially dangerous commands.	*	*
UNSTEP	<i>n</i> .UNSTEP. Takes the CPU out of step mode for that job. Only has an effect if the job had been previously STEPPed.		When in DIS, relinquishes display back to the command console interface (DSD). When in DSD, returns control back to DIS.
	UNSTEP. Takes the CPU out of single-step mode.		
VSN	VSN, <i>xx</i> , <i>vvvvvv</i> . Informs system that the operator has mounted tape with VSN <i>vvvvvv</i> on unit <i>xx</i> . See <i>How to Handle a Tape Request on the E,P Display</i> on page 2-37.		
WARN	WARN, <i>message</i> . Displays <i>message</i> to TELEX users as they log in. For users already logged in, the message is displayed when they exit their next command.		
	WARN, . Clears warning message. <i>See also:</i> MESSAGE, which displays a message <i>before</i> the user logs in.		

DIS OPERATION

DIS is a job display package which:

- 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 2-31) may bring itself into DIS by issuing the DIS control statement.
- Allows the operator to interact with job control so that control statements may be entered conveniently without having to log in to a terminal. Jobs initiated through DIS are system origin (see *Job Origin Types* on page 2-34) 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`.

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 2-2. 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	Central Memory displays in four groups of five octal digits and the display code translation.
G	Central Memory displays in four groups of five octal digits with COMPASS mnemonic translation.
H	File name table entries.

Display	Purpose
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 2-3. 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 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 primary 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.	ENB	ENB <i>i</i> , <i>xxxxxx</i> . Sets register <i>Bi</i> to <i>xxxxxx</i> .
DROP	DROP. Drops DIS. Does not drop the job if there are control statements remaining in the control statement buffer.	ENP	ENP, <i>xxxxxx</i> . Sets program counter <i>P</i> to <i>xxxxxx</i> .
ELS	ELS. <i>command</i> . Enter last statement. Enters control statement <i>command</i> in the control statement buffer after the last statement. This allows you to add on to an existing job.	ENS	ENS, <i>command</i> . Enter next statement. Clears the existing control statement buffer and enters <i>command</i> as the next statement.
ENA	ENA <i>i</i> , <i>xxxxxx</i> . Sets register <i>Ai</i> to <i>xxxxxx</i> .	ENX	ENX <i>i</i> , <i>xxxxxx</i> <i>xxxxxx</i> <i>xxxxxx</i> <i>xxxxxx</i> . Sets register <i>Xi</i> to <i>xxxxxx</i> <i>xxxxxx</i> <i>xxxxxx</i> . ENX <i>i</i> , <i>L value</i> . Sets register <i>Xi</i> left justified to <i>value</i> . ENX <i>i</i> , <i>b</i> , <i>xxxx</i> . Sets byte <i>b</i> of register <i>Xi</i> to <i>xxxx</i> . 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.
		O26	O26. (Oh-two-six.) Calls the O26 full screen text editor. See <i>Using O26</i> on page 2-19.
		RCP	RCP. Request central processor. Used to resume execution following a <i>DCP</i> 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.

Memory Entry

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, Dchars.
Changes contents of central memory address *loc* to display 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 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. O26 Special Keys

Key	Significance
0	Sets insertion point at first line.
8	Sets marker 8 at current line.
9	Sets marker 9 at current line.
+	Displays next page.

Key	Significance
-	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.
ENTER	Sets repeat entry mode.
SPACE	Begins text insertion.

System Commands

DIS	DIS. Writes the buffer, rewinds the output file, and returns control to DIS. This is the normal way to exit O26. <i>See also:</i> XDIS, which permits returning to DIS without writing the file.
DROP	DROP. Writes the buffer, rewinds the output file, and returns control to DSD. <i>See also:</i> XDROP, which permits returning to DSD without writing the file.
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. <i>filename</i> . Sets the default file name to <i>filename</i> .
READ	READ. <i>filename</i> . Reads <i>filename</i> 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. <i>filename</i> . Rewinds <i>filename</i> .

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

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.

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 Commands

OUT

OUT.
Transfers the file OUTPUT to the print queue, and PUNCH to the punch queue.

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 2-5 on page 2-22 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 53.

TABLE 2-5. 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
\	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 WRI TE. *file name*.
6. Enter the command DI S.

To Browse a File from DIS

1. Issue the command O26.
2. Once in O26, issue the command READ. *file name*.
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 XDI S. 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 Linux.

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 digest 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 2-34), the first four characters of the job name (left-justified, A-filled) are used instead of the hash code. Thus, a system origin job started by X. DI S. will have a job name of DISA 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.

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. DI S.
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 FUJ = directive when creating the account.

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

FIGURE 2-8. Computing a Vanish Hash Code

```

CYBER Console
DIS      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.FTNS,I=UNHASH,L=0.
01.38.02.      80000 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 .L0. 25. 0. 5.
21. LGO .L0. 27. 0. 305.

DIS      B
P =      2.      .      DDDD DDDD EDDD
CM = 6255.
UEC =      0.
20. DISDUURS. 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
FILE      0 A5      6737 B5      0
NA      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. :
S. : 6. : 7. 1TN 1. 10. : 11. :

```

ACCOUNT CREATION

1. At the console, type X. MODVAL.

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

NOTE

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

FIGURE 2-9. Starting MODVAL

```

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

DAYFILE.
16.39.00.SYSTEM S. EQ22, NUCC, S2 RECOVERED.
16.39.00.SYSTEM S. EQ21 13 DIRECT ACCESS FILES RECOVERED.
16.39.00.SYSTEM S. EQ21, NUCC, S1 RECOVERED.
16.39.01.SYSTEM S. EQ20 6 DIRECT ACCESS FILES RECOVERED.
16.39.01.SYSTEM S. EQ20, NUCC, S0 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.SYSQAES. SYS.
16.39.09.MAGNET S. MAGNET.
16.39.09.SYSQAES. SETPR,2.
16.39.09.SYSQAES. RETURN(INPUT)
16.39.09.SYSQAES. RFL(060000)
16.39.09.EXP S. EXP NOT FOUND.
16.39.09.SYSQAES. SYS(AZ)
16.39.09.SYSQAES. CALLPRG FILE NOT FOUND.
16.39.09.IAF S. IAFEX.
16.39.09.IAF S. NOS 1.4-552
16.44.52.AAA1004T. 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
CM = 6131.
UEC = 0.
-----7---1---
1. 6.IAF S PR77.0P7776.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 PR76.0P7773.RA 4572.FL 17.
X.MAGNET.
17. 10.BATCHI0S PR76.0P7774.RA 4672.FL 2.
IDLE.
20. 5.MODVADVS PR70.0P7761.RA 7537.FL 240.
X.REQUEST *K* DISPLAY.
21. .SYSTEM S PR .0P7777.RA 0.

0. MTR 20. 1. DSD 21. 2. 3. 4.
S. 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:

FIGURE 2-10. Initial MODVAL Display

```

CYBER Console
K 20.
17.33.19. 04/11/22.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AAREI. UNLOCKED VALIDUZ NOS 1.4-552
MID = NU. DEBUG.

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 -
CPHC CTPC CLPF CSPF CSOJ CASF CAND CCNR CSRP CSTP CTIM CUQC
CSAP CBIO CPRT CPLK CQLK CUST
CIAF CRBF CTAF CMCS CTVF CNOP CLOP ONLO
CAC2 CSMU CRCP COPE COPR CACC CPFF CATP CDSJ CANG CSTF
CNTM CCTM

P = 2. . DDDD DDDD EDDD 0000
CM = 6131.
UEC = 0.
-----7---1---
1. 6.IAF S PR77.0P7776.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 PR76.0P7773.RA 4572.FL 17.
X.MAGNET.
17. 10.BATCHI0S PR76.0P7774.RA 4672.FL 2.
IDLE.
20. 5.MODVADVS PR70.0P7761.RA 7537.FL 240.
X.REQUEST *K* DISPLAY.
21. .SYSTEM S PR .0P7777.RA 0.

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

```

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

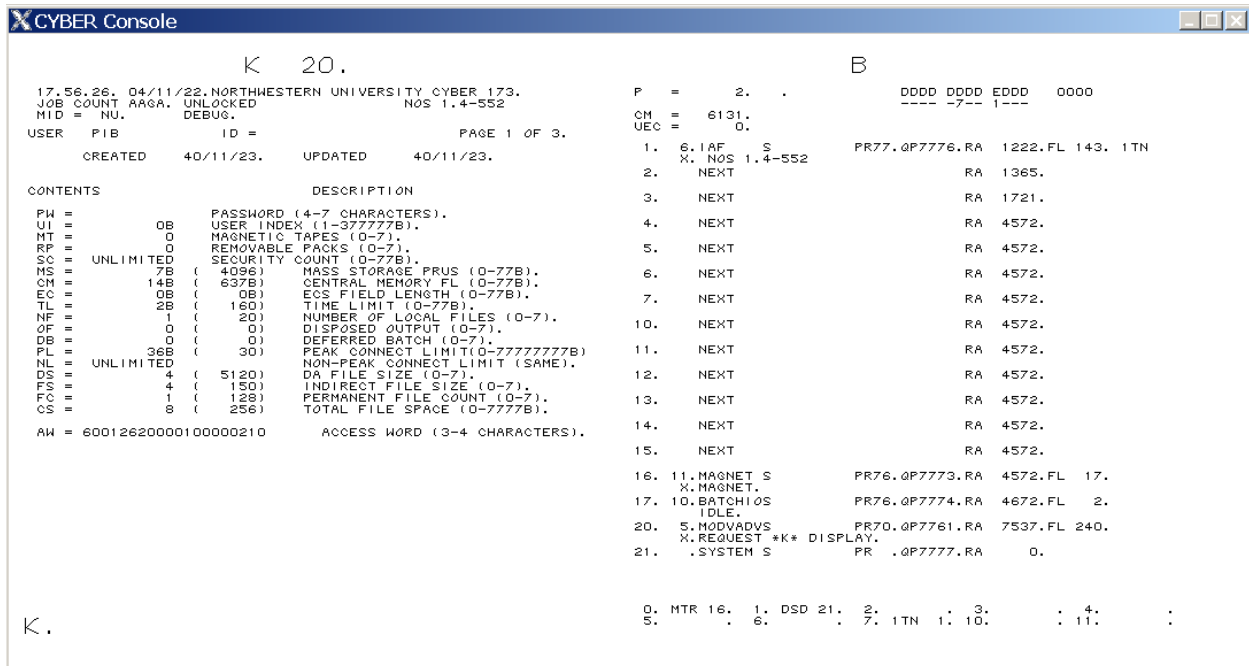
The following is displayed after typing K. C, PI B:

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 catalogs 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.

FIGURE 2-11. MODVAL User Creation



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

- If you wish to set a vanity hash code for this user (see *Vanity Hash Codes* on page 2-23), enter the directive *FUI = indexB* 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 2-34.

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

TABLE 2-6. 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

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry																		
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																		
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	<table border="1"> <thead> <tr> <th>Index</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>0</td><td>Job orig. default</td></tr> <tr><td>1</td><td>512</td></tr> <tr><td>2</td><td>1024</td></tr> <tr><td>3</td><td>2560</td></tr> <tr><td>4</td><td>4096</td></tr> <tr><td>5</td><td>20480</td></tr> <tr><td>6</td><td>32768</td></tr> <tr><td>7</td><td>Unlimited</td></tr> </tbody> </table>	Index	Value	0	Job orig. default	1	512	2	1024	3	2560	4	4096	5	20480	6	32768	7	Unlimited	DS=7
Index	Value																					
0	Job orig. default																					
1	512																					
2	1024																					
3	2560																					
4	4096																					
5	20480																					
6	32768																					
7	Unlimited																					
FS	Indirect access file size	0-7	<table border="1"> <thead> <tr> <th>Index</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>0</td><td>Job orig. default</td></tr> <tr><td>1</td><td>8</td></tr> <tr><td>2</td><td>24</td></tr> <tr><td>3</td><td>40</td></tr> <tr><td>4</td><td>64</td></tr> <tr><td>5</td><td>104</td></tr> <tr><td>6</td><td>504</td></tr> <tr><td>7</td><td>Unlimited</td></tr> </tbody> </table>	Index	Value	0	Job orig. default	1	8	2	24	3	40	4	64	5	104	6	504	7	Unlimited	FS=7
Index	Value																					
0	Job orig. default																					
1	8																					
2	24																					
3	40																					
4	64																					
5	104																					
6	504																					
7	Unlimited																					
FC	Number of permanent files	0-7	<table border="1"> <thead> <tr> <th>Index</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>0</td><td>Job orig. default</td></tr> <tr><td>1</td><td>8</td></tr> <tr><td>2</td><td>16</td></tr> <tr><td>3</td><td>24</td></tr> <tr><td>4</td><td>32</td></tr> <tr><td>5</td><td>40</td></tr> <tr><td>6</td><td>64</td></tr> <tr><td>7</td><td>Unlimited</td></tr> </tbody> </table>	Index	Value	0	Job orig. default	1	8	2	16	3	24	4	32	5	40	6	64	7	Unlimited	FC=7
Index	Value																					
0	Job orig. default																					
1	8																					
2	16																					
3	24																					
4	32																					
5	40																					
6	64																					
7	Unlimited																					

Permission Keyword	Meaning	Allowable Values You May Set in MODVAL (I)	How the Value is Interpreted to Give the Actual Limit	Suggested MODVAL Entry
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	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
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	0-37B	PO=I If PO=37B, System default	37B
PX	Echoplexing	FULL, HALF		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	<i>For users:</i> AP=ALL, AW=ALL, OFF=CSOJ, OFF=CTPC <i>For systems programmers:</i> AW=ALL, AP=ALL

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 TO REVIEWERS

Does anyone know what some of these permission bits mean? I got these from the MODVAL source. Most of them are obvious, but the mystery bits I have so marked.

TABLE 2-7. MODVAL Capability Descriptions

Bit	MODVAL Capability Name	Significance
0	CPWC	Password change permitted.
1	CTPC	Telex privileged commands permitted. (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.)
2	CLPF	Create large (direct access) permanent files.
3	CSPF	Create small (indirect access) permanent files.
4	CSOJ	System origin privileges. 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). 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.
5	CASF	Create and access system library files—in other words, the user may use the COMMON command.
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.

Bit	MODVAL Capability Name	Significance
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.
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. The effect of this bit is a mystery to me.
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. The effect of this bit is a mystery to me.
44	CATP	Access group catalogs. The effect of this bit is a mystery to me.
45	CDS D	May use the DSD command—an interactive command which gives access to console displays.
46	CANG	Access non-group catalogs. The effect of this bit is a mystery to me.

Bit	MODVAL Capability Name	Significance
47	CSTF	Computer center staff access, presumably provides access to the 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.
59	CCTM	Peak hours access. Should always be set.

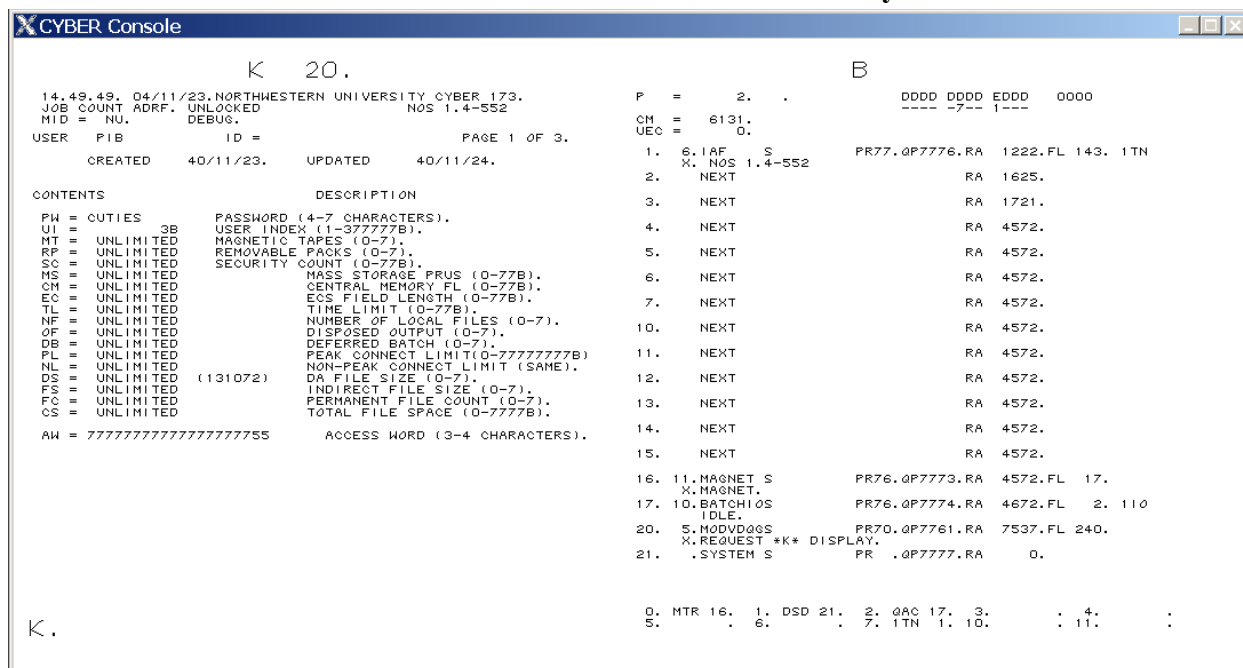
For our new user PIB (and these limits are appropriate for most users), we select:

FIGURE 2-12. Typical User Permissions

K. MT=7, RP=7, SC=77B, MS=77B, CM=77B, EC=77B, TL=77B
 K. NF=7, OF=7, DB=7, PL=⁸77777777B, DS=7, FS=7, FC=7, CS=0
 K. LP=77B, CP=77B, CC=77B, DF=77B, PX=FULL, IS=BATCH
 K. AP=ALL, AW=ALL, OFF=CSOJ, OFF=CTPC

The display now looks like:

FIGURE 2-13. MODVAL User Entry



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 2-26.
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 2-34.
7. Type K. END

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 2-7 on page 2-31), 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.

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 labelled, the operator merely has to mount the tape. If the tape is not labelled, the operator must issue a VSN command so that the system knows which tape unit to assign. (See the VSN command, page 2-15.)

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.

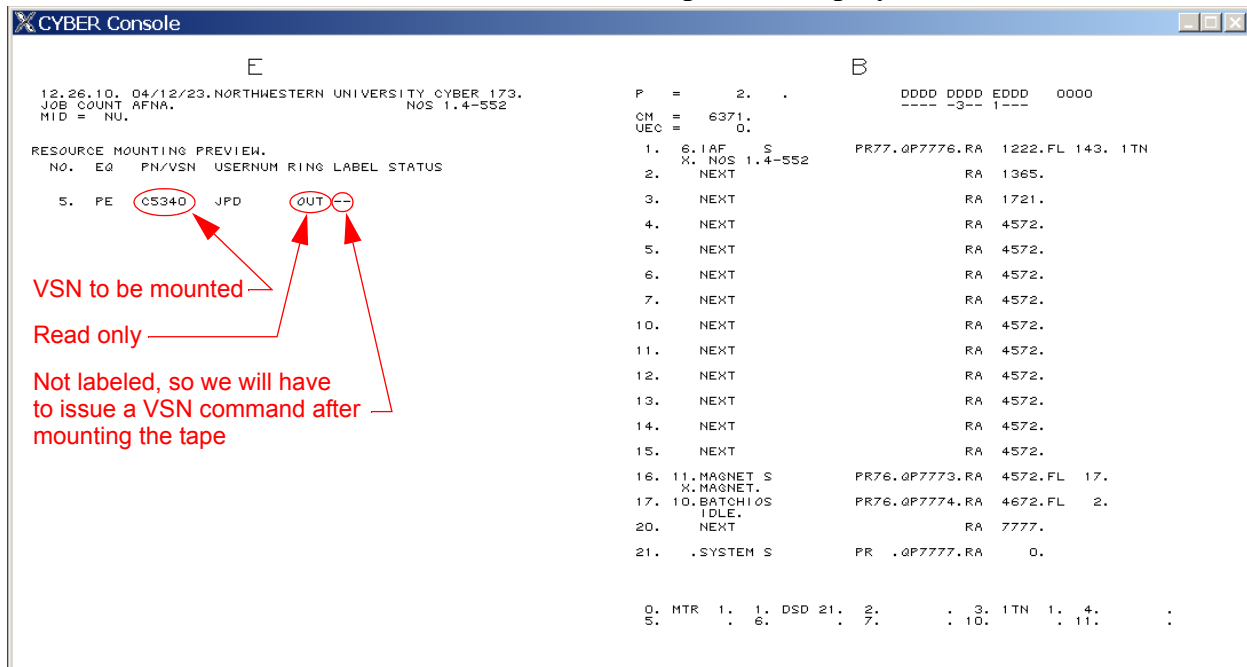
To mount a tape, you need not have a pending request. Thus, if there is a tape which you frequently use, you could just keep it mounted (by appropriate configuration grammar in the host cyber.ini file) at dead start and use it at your leisure.

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.

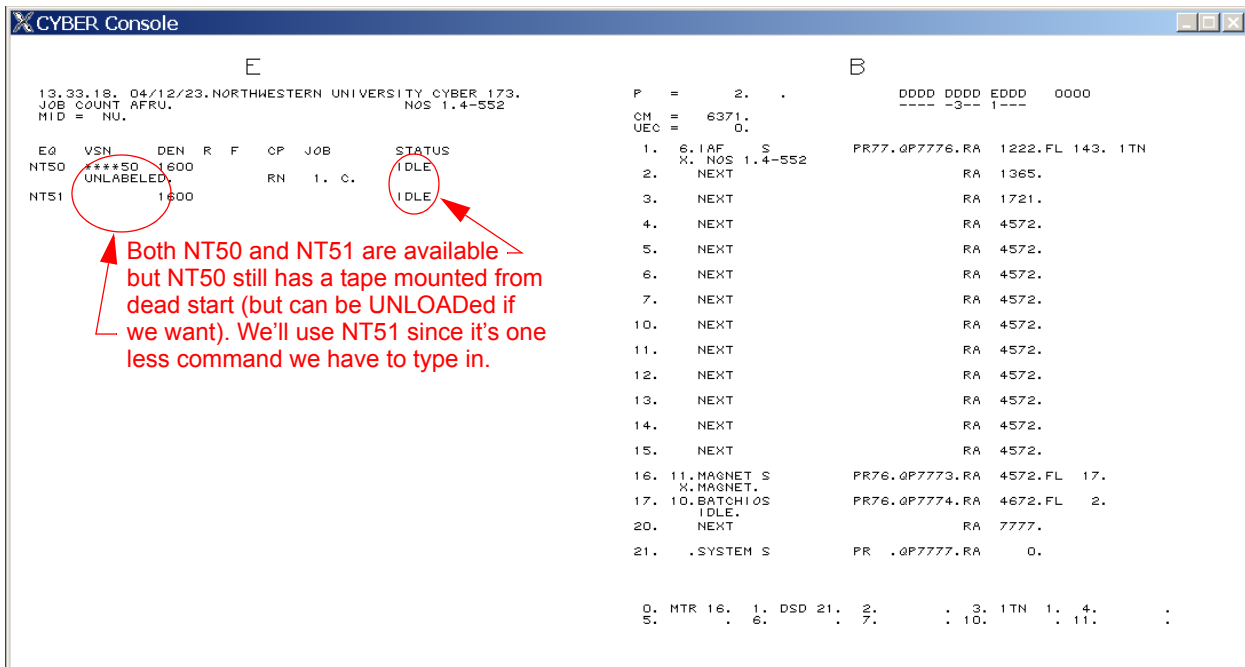
1. Check the E,P display and note the requested VSN and ring. See Figure 2-14 below.

FIGURE 2-14. Example E,P Display



2. Check the E,T display and make sure that NT51 is not in use. See Figure 2-15 on page 2-38.

FIGURE 2-15. Example E,T Display



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*

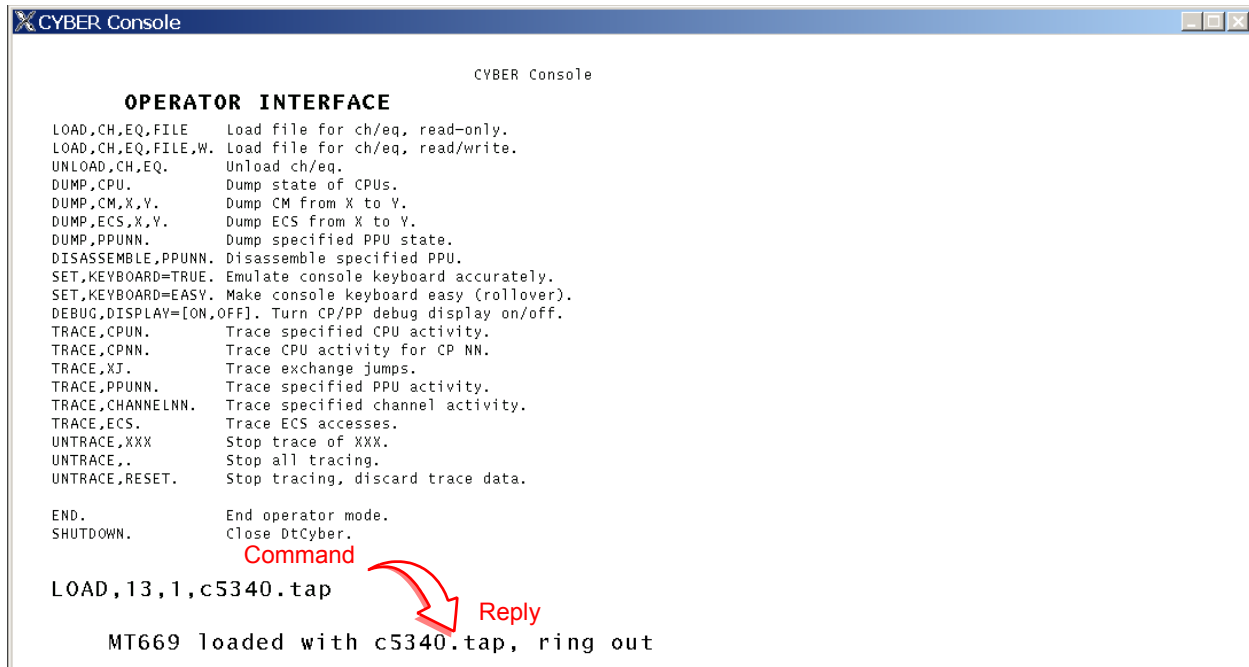
where *filename* is the name of the encapsulated tape file.

NOTE

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

See Figure 2-16 on page 2-39.

FIGURE 2-16. Desktop Cyber Operator Interface



5. Issue the command `END.` to return to Desktop Cyber's emulation mode.
6. If the tape is unlabelled, issue the command `VSN, 51, vvvvvv.` where `vvvvvv` is the VSN of the tape you just mounted. See Figure 2-17 on page 2-40.

If the tape is labelled, 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 2-17. E,T Before and After VSN Command

CYBER Console

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
NTS0	****50	1600					IDLE
	UNLABELED.				RN	1. C.	
NTS1	****51	1600					IDLE
	UNLABELED.				RN	1. C.	

B

P = 2. . DDDD DDDD EDDD OC
 ---- -3-- 1----

CM = 6371.
 UEC = 0.

1. 6.1AF 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.

CYBER Console

E

15.47.21. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
 JOB COUNT A&B1. NOS 1.4-552
 MID = NU.

EQ	VSN	DEN	R	F	CP	JOB	STATUS
NTS0	****50	1600					IDLE
	UNLABELED.				RN	1. C.	
NTS1	C5340	1600				. AAA100ST	ROLLED
	UNLABELED.				RN	1. C.AS	

VSN.S1.CS340.

- 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.

- Whenever a tape is requested via the B display, a message will flash with the request

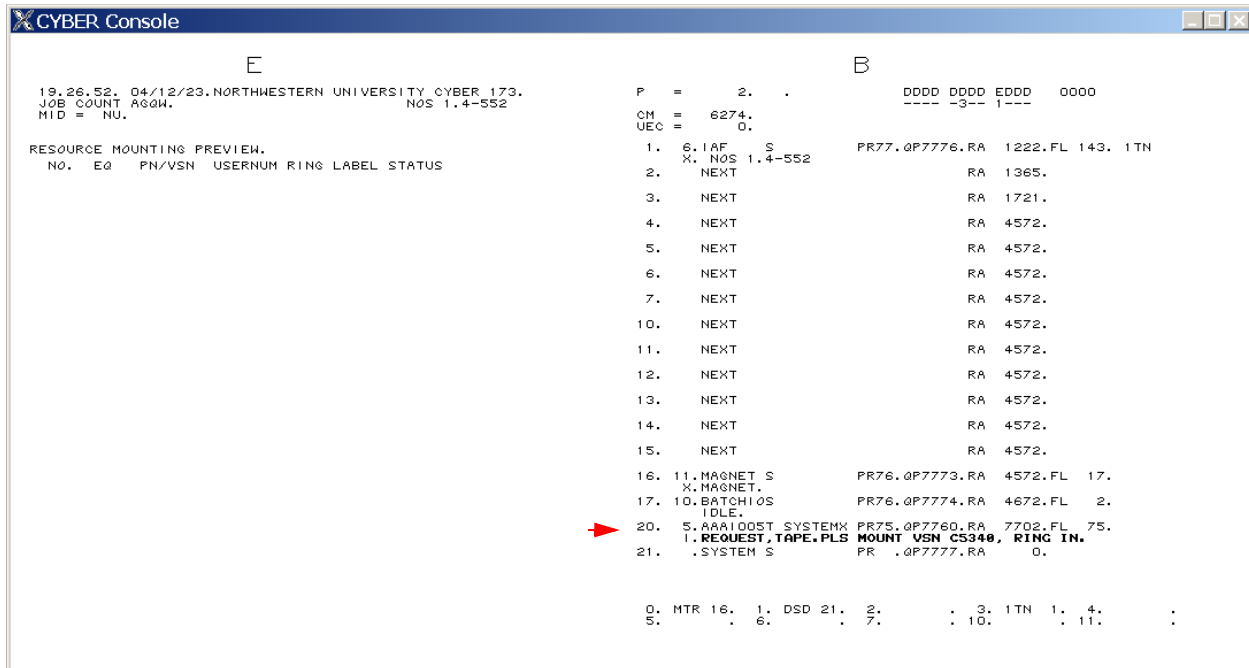
as shown in the example below at control point 20. Note that there is no corresponding request on the E,P display.

the B display as a comment on the REQUEST control statement. The message reads:

You will note that most of the request is in plain English; the message was passed to

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

FIGURE 2-18. Tape Request via the B Display



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

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

2. Check the E,T display and make sure that NT51 is not in use. See Figure 2-15 on page 2-38.
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.

LOAD, 13, 1, *filename*

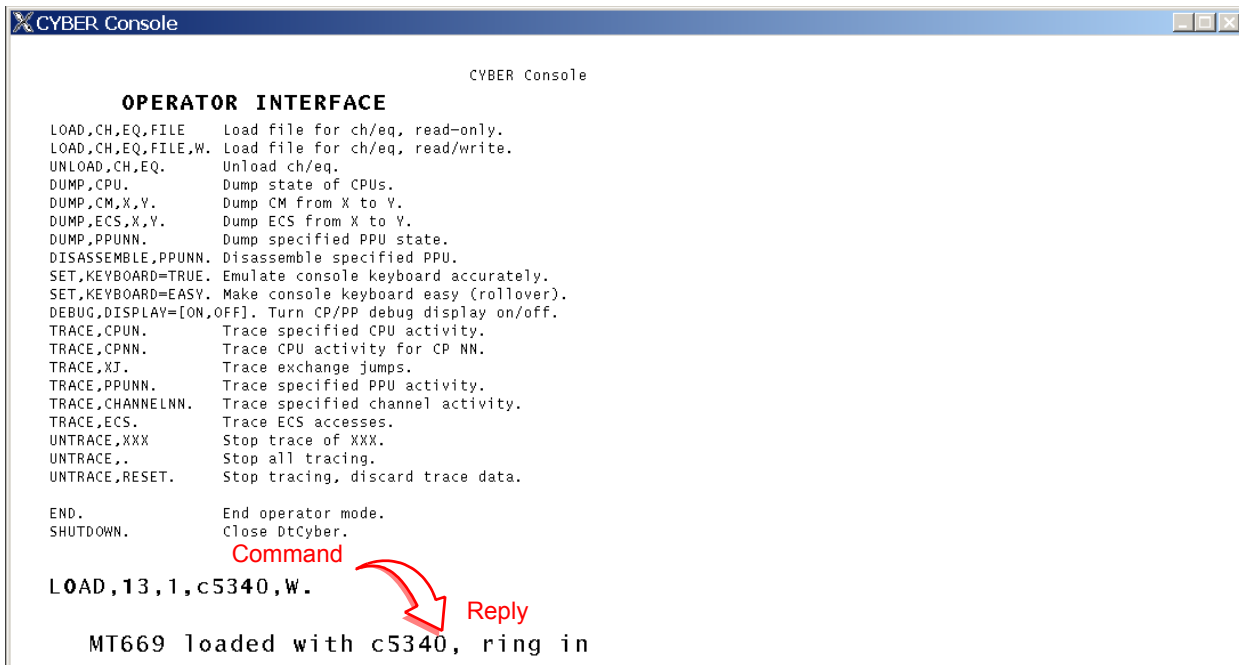
where *filename* is the name of the encapsulated tape file.

NOTE

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

See Figure 2-19 on page 2-42.

FIGURE 2-19. Desktop Cyber Operator Interface



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 labelled or not. Assuming you mounted the tape on NT51, the command to do this is:
n.ASSI GN, 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 2-43.

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 2-20. E,T and B Displays Before and After Assignment

```

CYBER Console
E
23.12.03. 04/12/23.NORTHWESTERN UNIVERSITY CYBER 173.
JOB COUNT AHQU. NOS 1.4-552
MID = NU.
E0 VSN DEN R F CP JOB STATUS
NTS0 ****50 1600 UNLABELED. RN 1. C. IDLE
NTS1 ****51 1600 IN UNLABELED. RN 1. C. IDLE

P = 2.
OH = 6274.
UEC = 0.
1. 6.1AF S PR77.0P7776.RA 1222.FL 143. 1TN
X. NOS 1.4-552 RA 1365.
2. NEXT RA 1721.
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. 11.MAGNET S PR76.0P7773.RA 4572.FL 17.
X.MAGNET.
17. 10.BATCHIOS PR76.0P7774.RA 4672.FL 2.
IDLE.
20. 5.AAA100ST.SYSTEMX PR75.0P7760.RA 7702.FL 75.
1.REQUEST.TAPE.PLS MOUNT VSN C5340. RING IN.
21. .SYSTEM S PR .0P7777.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 AHQU. NOS 1.4-552
MID = NU.
E0 VSN DEN R F CP JOB STATUS
NTS0 ****50 1600 UNLABELED. RN 1. C. IDLE
NTS1 ****51 1600 IN I , AAA100ST ROLLED UNLABELED. RN 1. C. AS

P = 2.
OH = 6371.
UEC = 0.
1. 6.1AF S PR77.0P7776.RA 1222.FL 143. 1TN
X. NOS 1.4-552 RA 1365.
2. NEXT RA 1721.
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. 11.MAGNET S PR76.0P7773.RA 4572.FL 17.
X.MAGNET.
17. 10.BATCHIOS PR76.0P7774.RA 4672.FL 2.
IDLE.
20. NEXT RA 7777.
21. .SYSTEM S PR .0P7777.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 unlabelled.

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 2-15). 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.

Unloaded tapes, whether explicitly by the operator or implicitly by the system, do not need to be unloaded in the Desktop Cyber Operator Interface.

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.

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 2-13). 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 2-14) 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.

To terminate a job:

- If the job is executing (that is, you see it on the B display), use the DROP command (see page 2-13). 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 2-13), which is not maskable. To terminate

subsystems (IAF, MAGNET, or BATCHIO), use the IDLE command (see page 2-13).

- If the job is rolled out, it must be rolled in before it can be dropped. Unless the job is in an event wait (such as waiting for a tape mount), you can roll the job in with the ROLLIN command (see page 2-14), and then DROP the job. If the job is in an event wait, use the E, T or Q displays to find the job's FNT ordinal and then use the PURGE command (see page 2-14) to drop the job. PURGE rolls the job in briefly just so it can be dropped.

INTERACTIVE JOBS

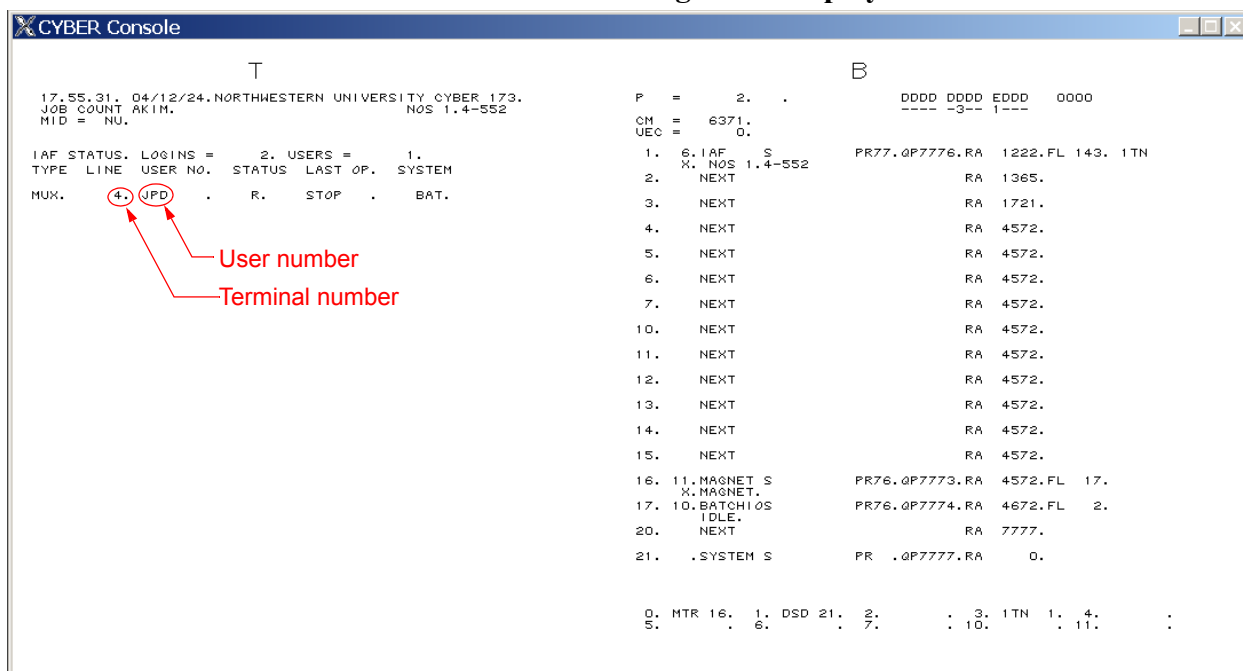
You will typically need to do only three things for or to an interactive job:

- 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.

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 2-21 on page 2-46.

FIGURE 2-21. Using the T Display



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

message as soon as their jobs become idle or right after they log in.

To determine the terminal number of an interactive user:

Simply consult the T display. See Figure 2-21.

- If you wish to send a message to users which they will see *before* they log in, use the MESSAGE command (page 2-13). Users already logged in will not see the message.

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 2-13) 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 2-15). Users will receive the

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 2-32).

- 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 2-14).

NOTE TO REVIEWERS

Does anyone know how the operator can get a list of output jobs showing the requested forms code? One would think this is in the Q display, but it's not.

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 2-13) 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 backend 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. The directives are described in [\[Tape Mounting for Users Xref \]](#). 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 2-29), disposing of files with special forms codes (see CSFC permission, page 2-32) and have permission to use the DSD command (see the CSDS permission, page 2-32) 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.
- Andy Capp Display. X. ADC. D terminates the program.
- Worms Display. X. WRM. 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.
- 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.
- 99 Bottles. Written by Control Freak Paul Koning, *NIID*, 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.

See [\[Direct Access Files Xref \]](#) for more information on indirect access files.

5. Assemble BOTPP with the command

COMPASS, I=BOTPP, S=PPTXT.

This causes COMPASS to read BOTPP as the input source, use PPTXT 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 2-15 for more information on how to use DIS and *Using O26* on page 2-19.

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

REWIND, LGO.
SYSEDT.

See [\[Creating a New Dead Start Tape Xref \]](#) for more information on SYSEDT.

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.)

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

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

REQUEST DISPLAY. (DIS)

12. 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).

13. Manually assign the display with the command *n*. ASSIGN, 20.

(E,A. shows that the EST ordinal for the DS (display console) equipment is 20.)

14. 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.

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

DROP.

INDEX

- Numerics
99 bottles 2-48
- A
access control word 2-30
account creation 2-24
account file 2-9
account management 2-22
andy capp display 2-48
application permission bits 2-30
ASSIGN 2-12
AUTO 2-12
auto mode 2-16
- B
baseball 2-48
batch jobs 2-44
batch origin 2-35
BATCHIO 2-6, 2-10, 2-13
BLITZ 2-12
- C
capability permissions 2-31
central memory contents 2-9
CFO 2-12, 2-44
CHECK POINT SYSTEM 2-12
CKP 2-12
CMS 2-5
COBOL5 2-44
command from operator 2-44
COMMENT 2-12
COMPASS 2-45
console keyboard 2-1
control point activity 2-9
control point status 2-10
create_interstate_connection 2-44
- D
date entry error 2-5
dayfile 2-16
dayfile, system 2-9
DCP 2-18
dead start 2-1
 date and time entry 2-4
dead start panel 2-1
DEBUG 2-12
DEBUG mode 2-12
DIAL 2-13
DIS 2-13, 2-20
DIS commands 2-18
DIS displays 2-16
DIS Operation 2-15
DIS special keys 2-17
DISABLE 2-13
 -display option 2-2
display, X window 2-2
DROP 2-13, 2-18, 2-20
DSD 2-7
 displays 2-8
- E
ELS 2-18
emailed job output 2-46
ENABLE 2-13
ENAi 2-18
ENBi 2-18
ENP 2-18
ENS 2-18
enter last statement 2-18
enter next statement 2-18
ENXi 2-18
equipment status table 2-9
error log 2-9
es 2-48
EYE 2-48
- F
file browsing in DIS 2-22
file creation in DIS 2-22
file name table 2-10, 2-44

FNT 2-44
FORM 2-13
forms codes 2-46
FTN5 2-44

G
games 2-48
GO 2-13, 2-44

H
HOLD 2-18

I
IAF 2-6
IAF jobs 2-45
IAF origin 2-35
IDLE 2-13
interactive jobs 2-45
interactive origin 2-35

J
job management 2-34
job origin types 2-34
job output, emailed 2-46
job termination 2-45
jobs, IAF 2-45
jobs, interactive 2-45
jobs, print 2-46
jobs, punch 2-47
jobs, telex 2-45

K
keyboard, console 2-1
KILL 2-13

L
LABEL 2-35
LOCK 2-13
LOGOFF 2-13
logoff 2-45
LUNAR 2-48

M
MAGNET 2-6, 2-13
MESSAGE 2-13

messages, sending to users 2-46
MODVAL 2-24

N
NAM 2-6

O
O26 2-18
O26, using 2-19
OFF 2-14
OFFSW 2-14
ON 2-14
ONSW 2-14
operator interface 2-39
operator, winking 2-48
output jobs 2-46
OVERRIDE 2-14

P
print jobs 2-46
punch jobs 2-47

Q
queue status 2-11

R
RCP 2-18
REQUEST 2-35
request central processor 2-18
resource use limits 2-27
ring 2-36
RNS 2-18
ROLLOUT 2-19
ROUTE 2-44

S
SECONDARY USER CARDS 2-13
sense switches, setting 2-44
set user index 2-19
set user number 2-19
shutdown 2-7
Snoopy 2-48
STEP 2-14
SUBMIT 2-44
SUI 2-19

SUN 2-19
system origin 2-35

T

tape job management 2-35
tape requests, processing 2-37
tape ring 2-36
tapes, requesting 2-36
tapes, unloading 2-43
TELEX 2-6, 2-13
telex jobs 2-45
telex origin 2-35
text mode, exiting 2-6
timesharing origin 2-35
timesharing status 2-11
transaction facility 2-10

U

UNLOAD 2-15

UNLOCK 2-15

UNSTEP 2-15

V

VALIDATION 2-13

VSN 2-15, 2-36

W

WARN 2-15

winking at operator 2-48

worms display 2-48

write ring 2-36

X

X 2-15

X window -display 2-2

XDIS 2-20

XDROP 2-20