

# NORD

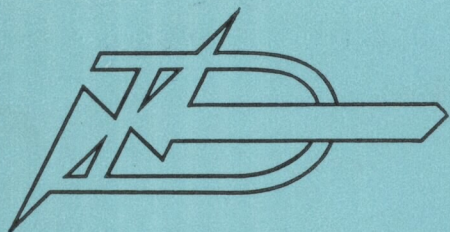
COMPUTER SYSTEMS

REFERENCE MANUAL FOR THE  
NORD TIMESHARING SYSTEM

by

Bo Lewendal

16 February 1973



A/S NORSK DATA-ELEKTRONIKK

Økernveien 145, Oslo 5

REFERENCE MANUAL FOR THE  
NORD TIMESHARING SYSTEM

by

Bo Lewendal

16 February 1973

### NORD Document Preparation

This reference manual was typed and edited on a NORD Timesharing System. Formatting was done by a NORD typesetting program called RUNOFF.

The material in this manual is for information purposes only and is subject to change without notice.

# TABLE OF CONTENTS

--oo0oo--

		Page
1	INTRODUCTION. . . . .	1
2	ACCESS TO NORD-TSS. . . . .	2
3	UTILITY COMMAND RECOGNITION . . . . .	3
4	FILES AND FILE NAMING . . . . .	4
5	THE UTILITY COMMAND LANGUAGE. . . . .	6
5.1	Logging onto NORD-TSS . . . . .	6
5.1.1	HELP. . . . .	7
5.1.2	LOGOUT. . . . .	7
5.2	Manipulation of the user machine. . . . .	8
5.2.1	Description of the user machine . . . . .	8
5.2.2	Manipulating the user's virtual memory. . . . .	9
5.2.2.1	RESET . . . . .	9
5.2.2.2	MEMORY. . . . .	9
5.2.2.3	DELETE-MEMORY . . . . .	9
5.2.2.4	LOAD-BINARY . . . . .	9
5.2.2.5	PLACE-BINARY. . . . .	9
5.2.2.6	GOTO-USER . . . . .	9
5.2.2.7	"escape". . . . .	9
5.2.2.8	DUMP. . . . .	10
5.2.2.9	RECOVER . . . . .	10
5.2.2.10	CONTINUE. . . . .	10
5.2.3	The file system . . . . .	11
5.2.3.1	OPEN-FILE . . . . .	11
5.2.3.2	CLOSE-FILE. . . . .	11
5.2.3.3	DELETE-FILE . . . . .	11
5.2.3.4	LIST-FILE . . . . .	11
5.2.3.5	RENAME. . . . .	12
5.2.3.6	ALLOCATE. . . . .	12
5.2.3.7	LIST-OBJECTS. . . . .	12
5.2.3.8	CREATE-FRIEND . . . . .	12
5.2.3.9	DELETE-FRIEND . . . . .	12
5.2.3.10	LIST-FRIENDS. . . . .	12
5.2.3.11	DEFINE-UIB-ACCESS . . . . .	12



5.2.3.12	DEFINE-FILE-ACCESS. . . . .	13
5.2.3.13	MAKE-REENTRANT. . . . .	13
5.3	Miscellaneous Commands. . . . .	14
5.3.1	STATUS. . . . .	14
5.3.2	SET-REGISTER. . . . .	14
5.3.3	EXAMINE . . . . .	14
5.3.4	RESERVE . . . . .	14
5.3.5	RELEASE . . . . .	14
5.3.6	WHERE-IS. . . . .	15
5.3.7	CLOCK-OFF . . . . .	15
5.3.8	CLOCK-ON. . . . .	15
5.3.9	LINK-TO . . . . .	15
5.3.10	BREAK-LINKS . . . . .	15
5.3.11	PASSWORD. . . . .	16
5.3.12	CLEAR-PASSWORD. . . . .	16
5.3.13	WHO-IS-ON . . . . .	16
5.3.14	INIT-ACCOUNTING . . . . .	16
5.3.15	LIST-ACCOUNTS . . . . .	16
5.3.16	PAUSE . . . . .	16
5.3.17	CREATE-USER . . . . .	17
5.3.18	DELETE-USER . . . . .	17
5.3.19	LIST-USERS. . . . .	17
5.3.20	TIME-USED . . . . .	17
5.3.21	RESPONSE-TIME . . . . .	17
5.3.22	DISK-SPACE. . . . .	17
5.3.23	MODE. . . . .	17
6	SUBSYSTEMS. . . . .	18
6.1	QED . . . . .	18
6.2	MAC . . . . .	18
6.3	FTN4. . . . .	18
6.4	FLDR. . . . .	18
6.5	BASIC . . . . .	19
6.6	DESCRIBE. . . . .	19
6.7	COPY. . . . .	19
6.8	PRINT-FILE. . . . .	19
6.9	KRYSSREF. . . . .	19
6.10	MAIL. . . . .	20
6.11	CHESS . . . . .	20
6.12	BONDESJAKK. . . . .	20
7	MONITOR CALL SYSTEM . . . . .	21
7.1	Description of Monitor Calls. . . . .	22
Appendix A	Line Edit Commands. . . . .	39
Appendix B	Generating NORD-TSS . . . . .	41

Index . . . . . 43

## 1 INTRODUCTION

The Norsk Data-Elektronikk Timesharing System (NORD-TSS) is a medium scale multi-access computational system which implements a user machine which is an extended subset of the NORD-1 computer. From the user's point of view NORD-TSS may be considered to consist of two parts:

1. The Utility Command Processor which provides the user with some tools to control and modify his user machine.
2. The Monitor Call System which gives access to the user program to extensions of the NORD-1 virtual machine.

As will be seen, NORD-TSS not only offers the resources of the NORD-1 computer to more than one simultaneous user, but also offers services not normally provided by the NORD-1.

NORD-TSS is intended for installations where many people require immediate access to a computing system. In such a situation NORD-TSS provides more services to each of many users than a stand-alone computer can provide for one user.

Reliability, power and ease of use have been the major design criteria for NORD-TSS.

## 2 ACCESS TO NORD-TSS

The primary medium by which computer and user communicate is the teletype console. Paper tape and punched cards are also available but are more cumbersome and unsuitable for primary input. A line printer is available for auxiliary output.

To establish a connection between teletype and computer the user must press the "escape" key on the teletype. The computer responds with a printed message indicating that it is ready to accept further input from the teletype in the form of commands. Through these commands the Utility Command Processor will provide sufficient services to get the user into contact with all other NORD-TSS facilities. The Utility is primarily a medium for specifying machine requirements and for handling users' files.



### 3 UTILITY COMMAND RECOGNITION

When typing in commands to NORD-TSS it is only necessary to type sufficient characters to distinguish the intended command from all other permissible commands. A special character, "-", exists in order to separate a command name into two or more distinct parts. Any and all parts of a command name may be abbreviated. Consider as an example the commands LOAD-BINARY and LIST-FILE. The first command may be typed as LOAD, LOAD-B, L-BINARY, L-B or LO or in quite a few other ways. The second command may be typed as LIST, LIST-F, L-FILE, L-F, LI-F or LI. However, if only L is typed NORD-TSS will indicate that the command is ambiguous.

The abbreviation lookup function just described is a standard NORD-TSS function which is almost always available to the user when typing in names. Examples of things which may be abbreviated are: command names, file names and user names.

As well as being able to abbreviate commands the user may also use the QED line edit feature while typing in a line. For a thorough description of the line edit feature refer to the QED manual. A brief description of the line edit commands will be found in the appendix.

It should be mentioned that all lines typed to NORD-TSS should be terminated by carriage return.

The collection of parameters for NORD-TSS commands is done in a standardized way as follows. The parameters to a command may be separated by either a comma or any number of spaces. If the user does not know what parameters a command expects or in which order he should type them in, he may simply omit any or all parameters. In this case NORD-TSS will ask for the required parameters.



#### 4 FILES AND FILE NAMING

A major component of NORD-TSS is a general purpose file system. This file system allows the user to uniformly access mass storage files and peripheral devices.

The general format of a file name is:

(<user name>)<file name>:<type name>

Any or all parts of a file name may be abbreviated.

Each user in NORD-TSS has his own file directory. The <user name> in the above format specifies in which file directory the file exists. The <user name> may consist of up to 14 alphanumeric characters.

The <file name> may consist of up to 16 alphanumeric characters.

The <type name>, consisting of up to 4 alphanumeric characters, specifies for what purpose the file is to be used. The <type name> is usually one of the following:

SYMB	Symbolic file
BIN	Binary file
BRF	Binary Relocatable Format file
PROG	Program file
LIBR	Library file
DATA	Data file

Normally, one need only specify the <file name> portion of a complete file name. The <type name> is usually obvious from context. The <user name> portion of a file name is necessary only when referring to another user's file.

When accessing a file from a program it is usually cumbersome and time consuming to refer to it by its file name. For this reason there is an OPEN-FILE command that returns a file number which uniquely identifies a file as long as it is open. A user may have a maximum of five files open at one time.

As has been mentioned, peripheral devices are accessed in the same way as mass storage files. The file names of peripheral devices are fixed and consist only of the <file name> part. Commands exist for reserving specific peripheral devices for just one user since it makes no sense for several users to access a device such as a card reader concurrently. Peripheral devices have fixed file numbers and need not be opened. Following is a list of the file names of the peripheral devices including their octal file numbers:

1	TELETYPE
2	TAPE-READER
3	FAST-PUNCH
4	CARD-READER
5	LINE-PRINTER
10	SLOW-PUNCH

The mass storage files may be accessed in sequential or random mode, and each file may be individually protected in any combination of DELETE, WRITE and READ access on three levels: Owner, Friend and Public. Owner is defined as the user in whose file directory the file exists. Friend is defined as a user whom the Owner has specified as someone who may have more access to the Owner's files than the general Public has. Public is defined as all users other than the Owner or his Friends. The default access mode for a file is such that Public users have only READ access to the Owner's files. Friends have both READ and WRITE access while the Owner also has DELETE access, i.e., he may delete the specified file or change its access mode. The access mode for a file may, however, be changed to anything else. For example, one may specify that Public users have no access, in which case the specified file is completely invisible to the Public.

## 5 THE UTILITY COMMAND LANGUAGE

### 5.1 Logging onto NORD-TSS

NORD-TSS's response on receiving the first "escape" from a hitherto dead teletype is to produce the message NORD TSS 2.5 IS UP followed by a carriage return line feed and the message @ENTER.

NORD-TSS now expects the user to type his name followed by a carriage return. This name must be one of which the system is aware, i.e., the name of an authorized user. If a password is associated with the declared user name a carriage return is output followed by the word PASSWORD.

The correct password must now be typed. The teletype echo will be suppressed so that the secrecy of the password will be preserved. The password is terminated by typing carriage return, and, if it has been correctly delivered, "OK, carriage return, line feed" is output. A user may set or change his password with the command PASSWORD which will be described later.

NORD-TSS will now type "PROJECT NUMBER P-". The user is now expected to type in an integer representing a project number. This project number is only used for accounting purposes. After the user types carriage return, NORD-TSS will respond with the herald character @. Commands may now be given to the Utility Command Processor.



#### 5.1.1 HELP

The HELP command merely lists on the teletype the commands available to the user. It exists only for the welfare of those users not completely familiar with NORD-TSS. The DESCRIBE program may be used to obtain further information about specific commands. The DESCRIBE program is described in section 6 of this manual.

#### 5.1.2 LOGOUT

The LOGOUT command causes the connection between the teletype and NORD-TSS to be terminated. The teletype is then in a dead state and "escape" must be typed in order to restore the connection with NORD-TSS. In addition, the LOGOUT command indicates how much compute time and console time has been used by the user while he was logged on.

## 5.2 Manipulation of the User Machine

### 5.2.1 Description of the User Machine

The user machine consists of a subset of core from 30000 octal to 57777 for a 24K NORD-1 or 77777 for a 32K NORD-1. This we will call the user's virtual memory since several different users may share the same region of real core.

The instruction set of the user machine is a subset of the NORD-1 instruction set. In fact, this subset corresponds exactly with the set of instructions which may be executed while in restricted mode in the NORD-1. That is, all instructions except IOT, TRR, MCL, MST, ION, INTEN, INTDS and those instructions which store into memory other than the user's virtual memory.

It was previously mentioned that NORD-TSS is an extended subset of the NORD-1. It is extended in three ways:

1. By the existence of the Utility Command Processor which provides a large number of user services.
2. By the existence of a file system which greatly extends the I/O system.
3. By the existence of a Monitor Call instruction which greatly extends the power of the instruction repertoire.



## 5.2.2 Manipulating the User's Virtual Memory

### 5.2.2.1 RESET

The RESET command releases all memory held by the user.

### 5.2.2.2 MEMORY start address, end address

The MEMORY command, when supplied with a null argument, lists on the teletype which portions of the user's virtual memory are allocated. If the arguments are supplied, then the user's virtual memory is redefined accordingly.

### 5.2.2.3 DELETE-MEMORY address of block

The DELETE-MEMORY command is used to release individual 2K blocks of virtual memory. The parameter specifies the start address of a block to be released.

### 5.2.2.4 LOAD-BINARY file

The LOAD-BINARY command simulates the action of pressing MASTER CLEAR and LOAD on the NORD-1. Input is taken from the specified file. This command also appropriately modifies the extent of the user's virtual memory.

### 5.2.2.5 PLACE-BINARY file

Same as LOAD-BINARY except that the loaded program is not started up.

### 5.2.2.6 GOTO-USER address

The GOTO-USER command transfers control to the user program at the specified address.

### 5.2.2.7 "Escape"

If the "escape" key is pressed while a user program is running, control will return to the Utility Command Processor with a message indicating where the program was interrupted being typed out. The same thing happens when the user program executes a WAIT instruction. All registers are saved. Therefore, the user program may be restarted by supplying the GOTO-USER command with the address at which the program was interrupted. All open files are closed when control returns to the Utility Command Processor.

#### 5.2.2.8 DUMP file, start address, restart address

The DUMP command saves the contents of the user's virtual memory plus the central registers on the specified file. The start address parameter indicates where the program should be started when it is later retrieved with the RECOVER command. The restart address parameter indicates where the program should be started when restarted with the CONTINUE command.

#### 5.2.2.9 RECOVER file

The RECOVER command retrieves a program from the specified file and starts it up at its main start address. As in the LOAD-BINARY command, the extent of the user's virtual memory is appropriately modified. There is an alternate form of the RECOVER command which is provided for the convenience of the user, whereby one may leave out the name RECOVER completely. In other words, instead of typing RECOVER MAC one may simply type MAC.

#### 5.2.2.10 CONTINUE

The CONTINUE command is used to restart a program which has previously been started with the RECOVER command. The program is started up at the address specified by the third parameter of the DUMP command.

### 5.2.3 The File System

#### 5.2.3.1 OPEN-FILE file, R or W or RX or WX

The OPEN-FILE command is used to open an existing file or to create and open a new file. As was mentioned before, a file name may consist of up to sixteen characters plus user and type information, and may also be abbreviated. Furthermore, if the file is a new file, then the filename must be enclosed by double quotes (""). The second argument specifies the type of access required. This may be R for read sequential, W for write sequential, RX for read random, or WX for read/write random. The OPEN-FILE command responds by typing FILE NUMBER = n where n is the octal number to be supplied to INBT or OUTBT when communicating with the file. At the present time a user may have up to five files open simultaneously. The OPEN-FILE command may also be called from a user program via a Monitor Call instruction.

#### 5.2.3.2 CLOSE-FILE file number or -1

The CLOSE-FILE command is used to close a file which has been opened with OPEN-FILE. The argument specifies which file to close. If -1 is supplied as the argument, then all currently open files are closed. The CLOSE-FILE command may also be called from a user program. Files are automatically closed when returning from a user program to the Utility Command Processor.

#### 5.2.3.3 DELETE-FILE file

The DELETE-FILE command deletes the specified file from the user's file directory and frees its disk space.

#### 5.2.3.4 LIST-FILE argument

The LIST-FILE command lists the names of all those files for which the given argument string is a subset of the complete file name. If the argument is null, then all file names are listed since the null string is always a subset of any other string. If LIST-FILE TS is typed, then all file names beginning with TS are listed. LIST-BL results in all file names where the second part of the name begins with BL being listed.



#### 5.2.3.5 RENAME old-name, new-name

The RENAME command is used to change the name of a file. The contents of the file remains unchanged.

#### 5.2.3.6 ALLOCATE file, track address, number of tracks

The ALLOCATE command creates a file which occupies a specific area of the disk. This command may only be executed by user SYSTEM.

#### 5.2.3.7 LIST-OBJECTS

The LIST-OBJECTS command lists on the teletype which files are currently open and some data about each of these files.

#### 5.2.3.8 CREATE-FRIEND user-name

The CREATE-FRIEND command defines the the specified user to be a friend of the currently logged on user.

#### 5.2.3.9 DELETE-FRIEND user-name

The DELETE-FRIEND command deletes the specified user from the currently logged on user's list of friends.

#### 5.2.3.10 LIST-FRIENDS

The LIST-FRIENDS command lists on the teletype which friends the currently logged on user has in his friend table.

#### 5.2.3.11 DEFINE-UIB-ACCESS access-word

The DEFINE-UIB-ACCESS command causes the user index block access of the currently logged on user to be set to that specified by access-word.

5.2.3.12 DEFINE-FILE-ACCESS file, access-word

The DEFINE-FILE-ACCESS command causes the access mode of the specified file to be set to that specified by access-word.

5.2.3.13 MAKE-REENTRANT file, list of block addresses

The MAKE-REENTRANT command makes the specified 2K blocks of the specified program file read only.



### 5.3 Miscellaneous Commands

#### 5.3.1 STATUS

The STATUS command lists on the teletype the contents of the user program's central registers.

#### 5.3.2 SET-REGISTER register, value

The SET-REGISTER command allows the user to change the contents of any register accessible by a user program. The first argument may be P, X, T, A, D, L, B or an octal address in which case the contents of a core location is to be modified. The second argument specifies an octal value which is to be stored into the specified register. If the number is followed immediately by the letter D then the number is taken to be decimal. For example, 100D is 144 octal.

#### 5.3.3 EXAMINE addr1, addr2

The EXAMINE command allows the user to examine the contents of any core location(s). If only the first argument is given, then the contents of that location is printed on the teletype. If both arguments are given, then the contents of locations addr1 to addr2 are printed on the teletype.

#### 5.3.4 RESERVE device-name

Since confusion would arise if a peripheral device could be used by more than one user at a time, a RESERVE command has been made available to permit a user to reserve a specific device while he uses it. The device-name may be TAPE-READER, FAST-PUNCH, SLOW-PUNCH, CARD-READER, LINE-PRINTER or an abbreviation of any of these names.

#### 5.3.5 RELEASE device-name

The RELEASE command releases the specified device so that some other user may use it.

5.3.6 WHERE-IS device-name

The WHERE-IS command indicates who, if anyone, is using the specified device, i.e., who has it reserved.

5.3.7 CLOCK-OFF

The CLOCK-OFF command turns off the real-time clock if the calling user is user SYSTEM.

5.3.8 CLOCK-ON

The CLOCK-ON command turns on the real-time clock.

5.3.9 LINK-TO user-name or TTY number

The LINK-TO command sets up a communication link between the calling user and the specified user. User may be either a user name or a teletype number. Upon completion of the command, teletype output generated by one of the users will also appear on the other user's teletype. In other words, the LINK-TO command permits two users to converse with each other or observe what the other user is doing.

5.3.10 BREAK-LINKS

The BREAK-LINKS command terminates any communication link set up by LINK-TO.

5.3.11 PASSWORD

The PASSWORD command allows a user to change his password. When called, the PASSWORD command elicits the message OLD PASSWORD IS at which time the user is expected to type in his old password. This is done so that no unauthorized person may change a user's password. If the old password is accepted, the system elicits the message NEW PASSWORD IS at which time the user is expected to type in a new password consisting of any sequence of characters terminated by carriage return. If only carriage return is typed, no password will be

required when logging onto the system.

5.3.12 CLEAR-PASSWORD user-name

The CLEAR-PASSWORD command is used to clear the password for the specified user. Only user SYSTEM may execute this command. This command is useful in cases where a user has forgotten his password.

5.3.13 WHO-IS-ON

The WHO-IS-ON command lists on the teletype which users are currently logged on and on which teletype the user is running.

5.3.14 INIT-ACCOUNTING

Initializes the accounting system. This command may only be executed by user SYSTEM.

5.3.15 LIST-ACCOUNTS title-string

Lists on the line printer the current contents of the accounting file. The title-string may be any sequence of characters and is output at the top of the account list.

5.3.16 PAUSE

The PAUSE command elicits the message PASSWORD IS . For all practical purposes the teletype is now considered by the system to be dead until the password of the currently logged on user is typed in. The PAUSE command may therefore be used to lock a teletype for a period of time.

5.3.17 CREATE-USER user-name

The CREATE-USER command, which may only be executed by user SYSTEM, enters a new user name into the system user directory. The new user is allocated an empty file directory and a null password.



5.3.18 DELETE-USER user-name

Deletes the specified user along with all of his mass storage files. This command may only be executed by user SYSTEM.

5.3.19 LIST-USERS

Lists the names of those users who are authorized to use the NORD Timesharing System.

5.3.20 TIME-USED

Lists the amount of compute time and console time that the user has used since he logged on.

5.3.21 RESPONSE-TIME

Lists the average response time for the whole system from the time that it was started.

5.3.22 DISK-SPACE

Lists the amount of file space which is left in the system.

5.3.23 MODE input file, output file

The MODE command implements a form of batch processing. Normally, command input and output is from and to the teletype. The MODE command allows the user to change the command input and output files to any general files. For example, one may take command input from the card reader and have command output go to the line printer. This feature is useful when one wants to perform a large number of predefined operations such as assembling a large programming system.

## 6 SUBSYSTEMS

Much of the software in NORD-TSS consists of subsystems. A subsystem is actually just a user program which performs some function for the user. Compilers, assemblers, editors and utility programs are examples of subsystems. A subsystem is started with the RECOVER command. The user may generate his own subsystems with the DUMP command. Following is a short description of each standard subsystem of NORD-TSS.

### 6.1 QED

QED is a program for editing symbolic text. It has extensive facilities for inserting, deleting and changing lines of text, a line edit feature, a powerful symbolic search feature, automatic tabs which may be set by the user, and a substitute command which permits all occurrences of a specified string of characters to be replaced with another string. Text may be read from any file and written onto any file.

### 6.2 MAC

MAC is the assembler and interactive debugging system for the NORD-1 computer.

### 6.3 FTN4

FTN4 is the FORTRAN IV compiler system for the NORD-1 computer.

### 6.4 FLDR

FLDR is the FORTRAN IV loader and runtime system.

### 6.5 BASIC

BASIC is an interactive programming system which is easy to learn and to use. BASIC is especially suitable for novice programmers and also for programming something very quickly.



#### 6.6 DESCRIBE

DESCRIBE is a utility program which gives a description of any or all commands available to the NORD-TSS user. To get a description of the OPEN-FILE command one need only type DESCRIBE OPEN. If DESCRIBE is called without a parameter it will describe all commands.

#### 6.7 COPY

COPY is a program which is used to copy one sequential file onto another sequential file. The files may be mass storage files or peripheral devices. The format of the COPY command is

COPY <destination> <source>

#### 6.8 PRINT-FILE

PRINT-FILE is a program which is used to list symbolic files on the line printer. The format of the PRINT-FILE command is

PRINT-FILE <file name> [, <tab specification>]

The optional tab specification consists of a series of column numbers separated by comma.

#### 6.9 KRYSSREF

KRYSSREF is a program which lists a symbolic assembly program on the line printer. Line numbers and page headings are added to the listing. The program listing is followed by a cross reference list over all defined and referenced symbols.

#### 6.10 MAIL

MAIL is a subsystem which allows a NORD-TSS user to send a message to any specified user or all users. The message is delivered when the addressed user types MAIL to the Utility Command Processor. The addressed user is also notified that he has mail when he logs on or off NORD-TSS.

6.11 CHESS

CHESS is a program which plays chess with the NORD-TSS user. Standard chess notation is used to designate moves.

6.12 BONDESJAKK

BONDESJAKK is a program which plays GO-MOKU (or "N in a row") with the NORD-TSS user.

## 7 THE MONITOR CALL SYSTEM

Up to now we have been describing services which are available through the Utility Command Processor. It is obvious that there must also be a mechanism for accessing various services from within a user program. Otherwise, there would be no way, for instance, to do input/output operations since the IOT instruction is illegal for user programs. The monitor call instruction, MCALL=161000, has therefore been defined as the medium through which the user program may access service routines in the monitor.

In general, arguments to monitor calls are put into the central registers and MCALL n is executed where n specifies which service routine is wanted. Most monitor calls skip when successful. In the following descriptions of monitor calls we will assume that if no failure return is specified then the monitor call returns without skipping.

Whenever a monitor call returns an error message number, the user may execute MCALL 64 in order to get an error message written on the teletype.

## 7.1 Description of Monitor Calls

0 LEAVE Exit from user program

No arguments

Return control to the Utility Command Processor

1 INBT Input a byte

T = file number

Return A

A = byte

Failure Return A

A = 2 if bad file number

A = 3 if end of file detected

A = 4 if card reader error

A = 5 if illegal device (device not reserved)

A = 6 if error in co-routine linkage (file no. 6)

A = 12 if end of device (timeout)

A = error number otherwise

2 OUTBT Output a byte

T = file number

A = byte

Return

Failure Return A

A = 2 if bad file number

A = 3 if end of file detected

A = 5 if illegal device (device not reserved)

A = 6 if error in co-routine linkage (file no. 6)

A = error number otherwise



3 ECHOM Define echo mode

A = 0 means always echo  
A = 1 means echo everything but control characters  
A = 2 means special MAC echo strategy  
A = 3 means use specified echo table pointed to by X  
X = address of echo table if A = 3

Return

The echo table consists of an 8 word bit table where each individual bit specifies whether or not a corresponding character should be echoed. Bit 15 of word 0 corresponds to the ASCII character with value 0 and bit 0 of word 7 corresponds to the ASCII character with value 177. A one bit specifies that the corresponding character should not be echoed while a zero bit specifies that it should be echoed.

4 BRKM Define break mode

A = 0 means always break  
A = 1 means break only on control characters  
A = 2 means special MAC break strategy  
A = 3 means use specified break table pointed to by X  
X = address of break table if A = 3

Return

The break table has the same format as the echo table in MCALL 3 except that a one bit specifies that the corresponding character should be a break character while a zero bit specifies that it should not be a break character. A break character is a character which causes the user program to be swapped in and started up. Normally, break characters are those characters which will cause the user program to supply some output.

5 RDISK Read page from scratch file

X = core address of page  
T = page number on file

Return

Failure Return A

A = error number

6 WDISK Write page on scratch file

X = core address of page

T = page number on file

Return

Failure Return A

A = error number

Each user has available to him a scratch file. This file need not be opened with the OPEN-FILE command. The user manipulates the scratch file via the RDISK and WDISK monitor calls. A page consists of 256 words. Pages are addressed consecutively on the file by page numbers ranging from zero and up over.

7 RPAGE Read page from random file

T = file number

X = core address of page

A = page number on file

Return

Failure Return A

A = error number

10 WPAGE Write page onto random file

T = file number

X = core address of page

A = page number on file

Return

Failure Return A

A = error number

The RPAGE and WPAGE monitor calls function exactly like RDISK and WDISK except that one must specify which file to operate on.

11 RCTIM Read compute time clock

No arguments

Return AD

AD = elapsed compute time

The AD register, upon return from the monitor call, contains a double precision integer which represents the compute time as a multiple of 20 ms.

12 EDIT Edit a line (QED line edit function)

X = string descriptor of old line

T = string descriptor of new line

Return

Failure Return if terminated with &L

The EDIT monitor call may be used to read in a line of text from the teletype. While the user types in his text he may correct mistakes using any of a variety of control characters. These control characters are described in Appendix A. The EDIT function also permits the user to copy selected text from an old line. For a description of string descriptors refer to monitor call 15 (SETUP).

13 CIBUF Clear input buffer

T = file number

Return

Failure Return A

A = error number

Deletes the contents of the input buffer for the specified device.

14 **COBUF** Clear output buffer

T = file number

Return

Failure Return A

A = error number

Deletes the contents of the output buffer for the specified device. If the user specifies a disk file for either CIBUF or COBUF, then the disk file is closed as with the CLOSE monitor call.

15 **SETUP** Generate a string descriptor

X = address of string descriptor (5 word object)

T = location of string

AD = maximum size of string in bytes

Return

A string descriptor consists of a 5 word object containing four character pointers: beginning pointer, end pointer, reader pointer and writer pointer. The contents of a string is defined to be the characters between the reader pointer and writer pointer. The reader pointer may not be greater than the writer pointer and both the reader and writer pointers must be within the area bracketed by the beginning and end pointers. The SETUP monitor call generates a string descriptor which describes a string of maximum size AD and beginning at T. The reader and writer pointers are initially set to point to the beginning of the string. All monitor calls which manipulate strings, except for the MSG monitor call, use string descriptors of this form.



16 GCI Read a byte from a string

X = address of string descriptor

Return A

A = byte

Failure Return if string is empty

GCI is used to read a byte from the front of a string.  
The reader pointer is incremented after the byte is read.

17 GCD Read a byte from the end of a string

X = address of string descriptor

Return A

Failure Return if string is empty

GCD is used to read the last byte of a string. The writer  
pointer is decremented after the byte is read.

20 WCI Write byte onto end of string

X = address of string descriptor

A = byte

Return

Failure Return if no more room in string

WCI is used to write a byte onto the end of a string. The  
writer pointer is incremented after the byte is written.

21 WCD Write byte onto the front of a string

X = address of string descriptor

A = byte

Return

Failure Return if no more room in string

WCD is used to write a byte onto the beginning of a string. The reader pointer is decremented after the byte is written.

22 GC LOOK AT NEXT BYTE IN STRING

X = address of string descriptor

Return A

A = byte

GC is used when one wishes to merely look at the next byte of a string. The string is in no way affected by this monitor call.

23 LEN Read length of string

X = address of string descriptor

Return AD

AD = length of string

LEN is used to find out how many bytes are in a string. The number of bytes is returned in the AD register as a double precision integer.

24 SETR Set reader pointer of string

X = address of string descriptor

AD = byte displacement

Return

SETR is used to move the reader pointer of a string. The parameter in AD may not be less than zero nor greater than the writer pointer. SETR is used most often to reset the reader pointer to the beginning of a string after one has read the whole string.

25     SETW            Set writer pointer of string

X = address of string descriptor  
AD = byte displacement

Return

SETW is used to move the writer pointer of a string. The parameter in AD may not be less than zero nor less than the reader pointer. SETW is used most often after SETUP to set the writer pointer to the end of a string.

26     not used

27     STEQL           Compare contents of two strings

X = address of string descriptor of first string  
T = address of string descriptor of second string

Return if both strings have the same content

Failure Return if the strings are not the same

30     STARR           Get element of string array

X = address of string descriptor  
T = pointer to string array  
A = index in string array (indexed from one)

Return

Failure Return if index out of range

A string array consists of an array where the first word indicates the number of elements in the array. The remaining words of the string array consist of pointers to strings, one for each entry. The strings consist of a sequence of characters terminated with the character \. STARR sets up the string descriptor pointed to by X to point to the string specified by the array index in A.

31 GCMD Read command from string

X = string descriptor of command string  
T = string descriptor of string to receive command

Return

Failure Return if no command in string

GCMD is used to read a command from the string specified by X. A command is defined as any sequence of characters terminated by space, comma or end of string. Leading blanks are skipped. The command is copied into the string specified by T. Both strings are updated appropriately.

32 MSG Print a message

X = pointer to beginning of string

Return

MSG prints on the teletype the string pointed to by X. \$ is interpreted as carriage return line feed and \ is interpreted as the end of the string.

33 DIV Integer divide

A = dividend  
T = divisor

Return A,D

A = result  
D = remainder

DIV implements integer divide which does not exist as a machine instruction on the NORD-1. A is divided by T with the result being placed in A and the remainder in D.

34 NORM Double precision normalize

AD = double precision integer

Return TAD

TAD = floating point number corresponding to AD



NORM converts a double precision integer into a standard floating point format.

35 IOUT            Output integer on teletype

A = integer  
T = radix

Return

IOUT prints the integer A on the teletype using the radix specified in T.

36 CNS            Convert number to string

A = integer  
T = radix  
X = address of string descriptor

Return

CNS performs the same function as IOUT except that the integer is written into the string specified by X.

37 CSN            Convert string to number

X = address of string descriptor  
T = radix

Return A

A = integer read from string

Failure Return if no integer found

CSN converts numeric characters found on the front of string X to an integer using the radix specified in T.

40 TRIM           Delete leading spaces in string

T = addresss of string descriptor

Return

TRIM deletes all leading spaces in the string specified by T.

41 BCOPY Copy block of words

X = source address  
T = destination address  
A = number of words

Return

BCOPY is used to move a block of words from one place in core to some other area. The number of words to be moved is specified by A. The block of A words starting at X is moved to the area of core specified by T.

42 OPENF Open a file

X = string descriptor of file name or 0  
T = 0 for write sequential  
T = 1 for read sequential  
T = 2 for read/write random  
T = 3 for read random  
A = pointer to default file type

Return A

A = file number

Failure Return A

A = error number

OPENF is used to open an existing file or to create and open a new file. If the file is a new file, then the file name must be enclosed by double quotes (""). The argument in T specifies in which way the user wants to access the file. The file number returned in A is the number to be supplied to INBT, OUTBT and other monitor calls when communicating with the file. If X = 0 when the monitor call is executed then NORD-TSS expects the user to type the file name on the teletype terminated by carriage return. Line edit commands may be called while typing in the file name.

setOP(15)

43 CLOSE Close a file or all files

T = file number or -1 for all files

Return

Failure Return A

A = error number

CLOSE is used to close a file which has been opened with the OPENF monitor call. The argument in T specifies which file to close. If -1 is supplied as the argument, then all currently open files are closed except for the user's standard scratch file. CLOSE may be used in place of CIBUF and COBUF or vice versa.

44 GCLIN Copy Utility command line to user

X = address of string descriptor

Return

GCLIN is used to get a hold of the parameter string following a utility command to start up a user subsystem. For example, if the user program PROG is called by:

@PROG AAA BBB CCC

then the program PROG may call GCLIN and have returned the string AAA BBB CCC.

45 DBRK Define breakpoint routine

T = address of register block

X = address of breakpoint routine

Return

Failure Return if bad arguments

This and the following two monitor calls implement a breakpoint function which is used by MAC to allow the user to define locations in his program where he wants execution to be interrupted in order that he may examine registers and core locations.



The DBRK monitor call is used to define where NORD-TSS should save the user program register block and to which location in MAC control should be transferred when the user program reaches a breakpoint. X specifies the address of the routine in MAC. T specifies where the register block should be saved. A register block consists of a nine word block containing the following registers: P, X, T, A, D, L, STS, B and MPR.

46    GBRK            Breakpoint instruction

No arguments

When this monitor call is encountered in a user program, the user's central registers are copied into the register block defined by DBRK and control is transferred to the routine which is also defined by DBRK.

47    SBRK            Restart user program

T = address of register block

Failure Return if bad argument

After MAC has processed a breakpoint, it restarts the user program with SBRK. T specifies a register block where the user program's central registers are loaded from. The MPR register is not loaded since the user is not allowed access to this register. The user program is restarted at the address given by the P register of the register block.

50    not used

51    not used

52    GUNUM            Convert user name to user number

X = string descriptor of user name

Return A

A = user number

Failure Return if no such user name



GUNUM is used to determine which user number a specified user name corresponds to. The monitor call does a failure return if the user name is not found in the user directory.

53 GUNAM Convert user number to user name

A = user number  
X = string descriptor

Return

Failure Return if no such user number

GUNAM is used to determine the user name which corresponds to the user number specified in A. The user name is appended to the string specified by X. If no user corresponds to the number in A, then the monitor call does a failure return.

54 PMSG Put message in mailbox

A = user number of addressee  
X = core address of message

Return

Failure Return if mailbox is full

PMSG is used solely by the MAIL subsystem to send a message to a specified user.

55 IMAIL Initialize mailbox

No arguments

Return

IMAIL is used by the MAIL subsystem to initialize the mailbox. Only user SYSTEM may execute this monitor call.

56 GMAIL Get user's mail

X = core address of message buffer

Return

GMAIL is used solely by the MAIL subsystem to retrieve a user's mail. The mail is copied into the message buffer specified by X.

57 WHUSR Which user am I?

No arguments

Return A

A = user number

WHUSR is used to determine what the user number is of the user who calls this monitor call. The user number is returned in A.

60 ABLKP Abbreviation lookup

X = string descriptor of given string

A = address of string array

D = index in array where search should start

Return A

A = index in string array of match

Failure Return A

A = -1 if no match

A = -2 if ambiguous

ABLKP is used to find the closest match between the given string and the strings in the string array. The search is started at the string specified by the index in D. A points to a string array of the form described in the explanation for monitor call STARR. When a match is found, the index of the matched string in the string array is returned in A. An error indication results if there is no match or if the given string is ambiguous, i.e. if the given string is a subset of more than one of the strings in the string array. For more information on abbreviation lookup refer to section 3, called Utility Command Recognition.

61 not used

62 not used

63 SOUT Output a string

X = string descriptor

Return

SOUT is used to output onto the teletype a string. The string is specified by X.

64 ERMSG Output error message

A = error number

Return

ERMSG is used to generate an error message corresponding to the error number in A. This error number is returned by most of the monitor calls upon failure return.

65 QERMS Output error message and leave user program

A = error number

QERMS performs the same function as ERMSG except that it will return to the Utility Command Processor after printing the error message.

66 not used

67 not used

70 EXEC Run program in monitor mode

Skip Return A

A = -1 if program allowed to run in monitor mode  
A = 0 otherwise

EXEC is used by special system programs to set the MPR register such that they may execute any instructions.

71 GNXTR Get next track of file

T = file number  
A = track number

Return A

A = next track number or -1 if no more

Failure Return A

A = error number

GNXTR is used to find out where the next allocated track in a file is. A specifies at which track one should start searching for the next track. If there are no more tracks, then -1 is returned in A. A track consists of 2048 words or of 8 pages. Track addresses range from zero and up over.

72 DLTRK Delete track

T = file number  
A = track number  
X = number of tracks to be deleted

Return

Failure Return A,T

A = error number  
T = last track deleted or -1 if none

DLTRK is used to delete a specified track of a file. The area freed on the disk may now be used by other files. The track address is specified in A and the number of tracks to be deleted is specified in X.



APPENDIX A -- Line Edit Commands

&A    Backspace one character (types UPARROW).  
&C    Copy one character from old line.  
&D    Copy rest of old line and terminate edit.  
&E    Change insert/replace mode (types < or >).  
&F    Copy rest of old line (without typing) and terminate  
      edit.  
&H    Copy rest of old line without terminating edit.  
&L    Terminate edit.  
&M    Terminate edit (carriage return).  
&OQ   Copy old line up to but not including character Q.  
&PQ   Skip characters in old line up to but not including  
      character Q (% is typed for each character skipped).  
&Q    Backspace to the beginning of the new and old lines  
      (types LEFTARROW CR).  
&R    Retype fast.  
&S    Skip one character in old line (types %)  
&T    Retype aligned.  
&VQ   Take character Q literally.  
&W    Backspace one "word" (types \).  
&XQ   Skip characters in old line up to and including the  
      character Q (% is typed for each character skipped).  
&Y    Append rest of old line to new line and edit the  
      result.

&ZC Copy old line up to and including character C.

The following is the procedure for generating a NORD-TSS installation. The necessary tapes are as follows:

1. MINIT (one of PD-842,843,844,845,909,910)
2. TSS 2.5 Binary Tape (special for each installation)
3. Subsystem Tape 1 (PD-943)
4. Subsystem Tape 2 (PD-944 or PD-945)

If one only requires to update the operating system code, only the TSS 2.5 Binary Tape need be loaded (with MASTER-CLEAR and LOAD).

1. Load the appropriate MINIT tape (i.e. correct disk type, device number, etc.) and use it to initialize the disk.
2. Load the TSS 2.5 Binary Tape using the MASTER-CLEAR and LOAD pushbuttons. The switch register should be set to 131313 while the tape is being loaded. (This will cause user SYSTEM to be automatically created when the system is started for the first time.)
3. When the system starts up, press MEMORY-PROTECT and log on as follows:

NORD TSS 2.5 IS UP

@ENTER SYSTEM  
PROJECT NUMBER P-753  
@

4. Type in the following. (User typed text is underlined).

@INIT-ACCOUNT  
@ALLOCATE "SYSTEM:DATA" 0\_342  
@DEFINE-UIB-ACCESS (SYSTEM) 733  
@OPEN "SCRATCH1:DATA" WX  
FILE NUMBER = 100  
@OPEN "SCRATCH2:DATA" WX  
FILE NUMBER = 101  
@CLOSE -1

continue for each teletype in the system:

@OPEN "SCRATCH12:DATA" WX  
FILE NUMBER = 101  
@CLOSE -1

```
@DEFINE-UIB-ACCESS_(SYSTEM)_731  
@RESERVE_T-R
```

put Subsystem Tape 1 in tape reader

```
@MODE_T-R.TEL
```

typeout from automatic loading of subsystems

```
@RESET
```

put Subsystem Tape 2 in tape reader

```
@MODE_T-R.TEL
```

typeout from automatic loading of subsystems

```
@RESET
```

5. NORD-TSS is now completely loaded and is ready for use. The first thing which should be done is to create authorized users.



## INDEX

abbreviation lookup . . . . .	36
abbreviation of commands. . . . .	3
ABLKP . . . . .	36
access. . . . .	11
access mode . . . . .	5
Access to NORD-TSS. . . . .	2
access word . . . . .	12
ALLOCATE. . . . .	12
assemblers. . . . .	18
authorized user . . . . .	6
BASIC . . . . .	18
batch processing. . . . .	17
BCOPY . . . . .	32
block copy. . . . .	32
BONDESJAKK. . . . .	20
break mode. . . . .	23
BREAK-LINKS . . . . .	15
breakpoint. . . . .	33
BRKM. . . . .	23
CARD-READER . . . . .	14
central registers . . . . .	10, 14
CHESS . . . . .	20
CIBUF . . . . .	25, 33
CLEAR-PASSWORD. . . . .	16
CLOCK-OFF . . . . .	15
CLOCK-ON. . . . .	15
CLOSE . . . . .	26, 33
CLOSE-FILE. . . . .	11
CNS . . . . .	31
COBUF . . . . .	26, 33
collection of parameters. . . . .	3
commands. . . . .	3, 19
compare strings . . . . .	29
compilers . . . . .	18
compute time. . . . .	7, 17, 25
console time. . . . .	7, 17
CONTINUE. . . . .	10
COPY. . . . .	19
CREATE-FRIEND . . . . .	12
CREATE-USER . . . . .	16
cross reference . . . . .	19
CSN . . . . .	31
DBRK. . . . .	33

debugging system. . . . .	18
default access mode . . . . .	5
DEFINE-FILE-ACCESS. . . . .	13
DEFINE-UIB-ACCESS . . . . .	12
DELETE access . . . . .	5
DELETE-FILE . . . . .	11
DELETE-FRIEND . . . . .	12
DELETE-MEMORY . . . . .	9
DELETE-USER . . . . .	17
DESCRIBE. . . . .	7, 19
Description of the User Machine . . . . .	8
design criteria . . . . .	1
disk space. . . . .	11
DISK-SPACE. . . . .	17
DIV . . . . .	30
DLTRK . . . . .	38
DUMP. . . . .	10, 18
echo mode . . . . .	23
ECHOM . . . . .	23
EDIT. . . . .	25
edit commands . . . . .	3
editing . . . . .	18
editors . . . . .	18
ERMSG . . . . .	37
error message . . . . .	37
escape key. . . . .	9
EXAMINE . . . . .	14
EXEC. . . . .	37
FAST-PUNCH. . . . .	14
file directory. . . . .	4, 11
file name . . . . .	4, 11
file names. . . . .	3, 11
file number . . . . .	4
file space. . . . .	17
file system . . . . .	4
File System . . . . .	11
Files and File Naming . . . . .	4
FLDR. . . . .	18
FORTTRAN IV. . . . .	18
Friend. . . . .	5
FTN4. . . . .	18
GBRK. . . . .	34
GC. . . . .	28
GCD . . . . .	27
GCI . . . . .	27
GCLIN . . . . .	33
GCMD. . . . .	30
GMAIL . . . . .	36

GNXTR . . . . .	38
GO-MOKU . . . . .	20
GOTO-USER . . . . .	9
GUNAM . . . . .	35
GUNUM . . . . .	34
HELP . . . . .	7
herald character . . . . .	6
I/O system . . . . .	8
IMAIL . . . . .	35
INBT . . . . .	11, 22
INIT-ACCOUNTING . . . . .	16
input a byte . . . . .	22
input buffer . . . . .	25
instruction set . . . . .	8
integer divide . . . . .	30
interactive debugging . . . . .	18
Introduction . . . . .	1
IOT instructions . . . . .	21
IOUT . . . . .	31
KRYSSREF . . . . .	19
LEAVE . . . . .	22
LEN . . . . .	28
line edit . . . . .	3
line edit commands . . . . .	32, 39
line edit function . . . . .	25
LINE-PRINTER . . . . .	14
LINK-TO . . . . .	15
LIST-ACCOUNTS . . . . .	16
LIST-FILE . . . . .	11
LIST-FRIENDS . . . . .	12
LIST-OBJECTS . . . . .	12
LIST-USERS . . . . .	17
LOAD . . . . .	9
LOAD-BINARY . . . . .	9, 10
loader . . . . .	18
logging on . . . . .	6
LOGOUT . . . . .	7
MAC . . . . .	10, 18, 34
MAIL . . . . .	19
MAIL subsystem . . . . .	35
mailbox . . . . .	35
MAKE-REENTRANT . . . . .	13
Manipulating Virtual Memory . . . . .	9
Manipulation of the User Machine . . . . .	8
mass storage . . . . .	4
MASTER CLEAR . . . . .	9



MCALL . . . . .	21
memory. . . . .	9
MEMORY. . . . .	9
message . . . . .	19, 35
MODE. . . . .	17
Monitor Call. . . . .	8, 11
Monitor Call System . . . . .	1, 21
Monitor Calls . . . . .	22
monitor mode. . . . .	37
MSG . . . . .	26, 30
N in a row. . . . .	20
new file. . . . .	11
NORM. . . . .	30
normalize. . . . .	30
novice programmers. . . . .	18
open <del>file</del> . . . . .	32, 39
OPEN-FILE . . . . .	4, 11, 19, 24
OPENF . . . . .	32
OUTBT . . . . .	11, 22
output a byte . . . . .	22
output buffer . . . . .	26
Owner . . . . .	5
page number . . . . .	24
parameter string. . . . .	33
password. . . . .	6
PASSWORD. . . . .	15
PAUSE . . . . .	16
peripheral device . . . . .	14
peripheral devices. . . . .	4
PLACE-BINARY. . . . .	9
PMSG. . . . .	35
PRINT-FILE. . . . .	19
PROJECT NUMBER. . . . .	6
Public. . . . .	5
QED . . . . .	3, 18
QERMS . . . . .	37
random file . . . . .	5, 11, 24
RCTIM . . . . .	25
RDISK . . . . .	23
READ access . . . . .	5
read byte . . . . .	27
real core . . . . .	8
real-time clock . . . . .	15
RECOVER . . . . .	10, 18
RELEASE . . . . .	14



RENAME. . . . .	12
RESERVE . . . . .	14
reserving devices . . . . .	4
RESET . . . . .	9
RESPONSE-TIME . . . . .	17
restart address . . . . .	10
restarting a program . . . . .	10
restricted mode . . . . .	8
RPAGE . . . . .	24
runtime system. . . . .	18
SBRK. . . . .	34
scratch file. . . . .	23, 24
sequential file . . . . .	5, 11, 19
service routines. . . . .	21
SET-REGISTER. . . . .	14
SETR. . . . .	28
SETUP . . . . .	25, 26
SETW. . . . .	29
SLOW-PUNCH. . . . .	14
SOUT. . . . .	37
STARR . . . . .	29, 36
start address . . . . .	10
STATUS. . . . .	14
STEQL . . . . .	29
string array. . . . .	29
string descriptor . . . . .	26
Subsystems. . . . .	18
symbolic files. . . . .	19
TAPE-READER . . . . .	14
TIME-USED . . . . .	17
track . . . . .	38
TRIM. . . . .	31
type information. . . . .	11
type name . . . . .	4
user machine. . . . .	8
user name . . . . .	4, 34
user names. . . . .	3
user number . . . . .	34
user program. . . . .	18, 21
user SYSTEM . . . . .	12, 16, 16, 16
Utility Command Language. . . . .	6
Utility Command Processor . . . . .	1, 8, 9, 11, 22
Utility Command Recognition . . . . .	3, 36
utility programs. . . . .	18
virtual machine . . . . .	1
virtual memory. . . . .	8, 9, 10

WAIT instruction. . . . .	9
WCD . . . . .	27
WCI . . . . .	27
WDISK . . . . .	24
WHERE-IS. . . . .	15
WHO-IS-ON . . . . .	16
WHUSR . . . . .	36
WPAGE . . . . .	24
WRITE access. . . . .	5
write byte. . . . .	27



## GENERERING AV TSS 3.0

MINIT kjøres først. Den starter automatisk med et spørsmål hvor man svarer med I. Dette starter initialiseringen av disken, og etter en tid skriver programmet ut FINISHED når initialiseringen er ferdig.

Nå leses systemtapen (TSS 3.0) inn og håndswitchregisteret må nå være 131313.

NORD TSS 3.0 IS UP

Dette blir skrevet automatisk ut på TT1 hvis tapen er lest riktig inn

@ENTER SYS

PROJECT NUMBER P-1

(Her kommer en feilmelding første gang)

@DISK

nn \* TRACKS LEFT OUT OF 2048 TRACKS (4096K)

@TRANS SYS SYS nn \*

@INIT - ACC

@ALLOCATE "SYSTEM:DATA" 0 342 (Dette tar litt tid)

@DEF-UIB-ACC (SYSTEM) 733

@OPEN "SCRATCH1: DATA", WX

FILE NUMBER = 100

@OPEN "SCRATCH2: DATA", WX

FILE NUMBER = 101

@CLOSE -1

} Dette må gjøres for hver Teletype i systemet

*101, 1, file 2*  
*PROVE*  
*MIKRO*

(Det må foretas CLOSE -1 etter at 6 filer er åpnet grunnet systemets begrensning på 6 åpne filer samtidig.)

@DEF-UIB-ACC (SYSTEM) 731

@RESERVE T-R

@MODE T-R TEL

Leser subsystem tape 1

@RESET

@MODE T-R TEL

Leser subsystem tape 2

@RESET

@MODE T-R TEL

Leser subsystem tape 3

@RESET

@MODE T-R TEL

Leser subsystem tape 4

@CREATE-USER <user name>

@TRANSF <user name>SYST <no of tracks>

} Dette gjøres for hver ny bruker som genereres

\* Det tallet som nn indikerer blir skrevet ut ved hjelp av DISK kommandoen, og må brukes som input til TRANS kommandoen.