


# Norsk Data



## **SINTRAN III** **System Supervisor**

**ND-30.003.06A**





# **SINTRAN III**

## **System Supervisor**

**ND-30.003.06A**

## **NOTICE**

The information in this document is subject to change without notice. Norsk Data A.S assumes no responsibility for any errors that may appear in this document. Norsk Data A.S assumes no responsibility for the use or reliability of its software on equipment that is not furnished or supported by Norsk Data A.S.

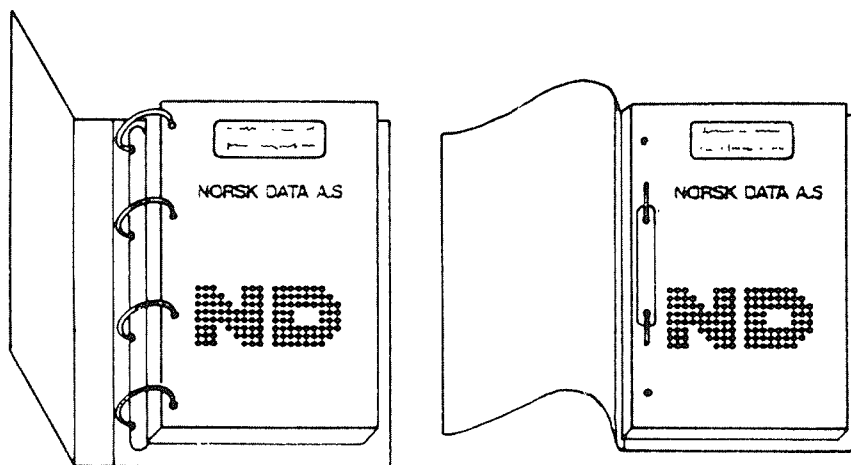
The information described in this document is protected by copyright. It may not be photocopied, reproduced or translated without the prior consent of Norsk Data A.S.

Copyright ©1985 by Norsk Data A.S.

This manual is in loose-leaf form for ease of updating. Old pages may be removed and new pages easily inserted if the manual is revised.

The loose-leaf form also allows you to place the manual in a ring binder (A) for greater protection and convenience of use. Ring binders with 4 rings corresponding to the holes in the manual may be ordered in two widths, 30 mm and 40 mm. Use the order form below.

The manual may also be placed in a plastic cover (B). This cover is more suitable for manuals of less than 100 pages than for large manuals. Plastic covers may also be ordered below.



A: Ring Binder

B: Plastic Cover

Please send your order to the local ND office or (in Norway) to:

**Norsk Data A.S**  
Graphic Center  
P.O. Box 25, Bogerud  
0621 Oslo 6, Norway

---

## ORDER FORM

I would like to order

..... Ring Binders, 30 mm, at nkr 20,- per binder

..... Ring Binders, 40 mm, at nkr 25,- per binder

..... Plastic Covers at nkr 10,- per cover

Name .....

Company .....

Address .....

.....

City .....



## PRINTING RECORD

[illegible]

SINTRAN III System Supervisor  
Publ. No. ND-30.003.06 Rev. A



**Norsk Data A.S**  
Graphic Center  
P.O.Box 25, Bogerud  
0621 Oslo 6, Norway

Manuals can be updated in two ways, new versions and revisions. New versions consist of a complete new manual which replaces the old manual. New versions incorporate all revisions since the previous version. Revisions consist of one or more single pages to be merged into the manual by the user, each revised page being listed on the new printing record sent out with the revision. The old printing record should be replaced by the new one.

New versions and revisions are announced in the Customer Support Information (CSI) and can be ordered as described below.

The reader's comments form at the back of this manual can be used both to report errors in the manual and to give an evaluation of the manual. Both detailed and general comments are welcome.

These forms and comments should be sent to:

Documentation Department  
Norsk Data A.S  
P.O. Box 25, Bogerud  
0621 Oslo 6, Norway

Requests for documentation should be sent to the local ND office or (in Norway) to:

Graphic Center  
Norsk Data A.S  
P.O. Box 25, Bogerud  
0621 Oslo 6, Norway

## **PREFACE**

---

### **THE PRODUCT**

SINTRAN III is a general purpose operating system for ND-100 and ND-500 computers. The operator and system supervisor functions on the following products are described in this manual:

SINTRAN III VSE, version J	ND-10174
SINTRAN III VSX, version J	ND-10575
SINTRAN III VSX-500, version J	ND-10576

The VSE and VSX variants are used on the ND-100 computers while the VSX-500 is used on ND-500 computers. VSE is an abbreviation of Virtual Storage Extended, and VSX is an abbreviation of Virtual Storage Extended for the CX CPUs.

---

### **THE READER**

There are two types of readers of this manual. The first type is the operators who control the daily operations of the system. The second type is the system supervisors who fulfill system programmer functions as well as operator functions.

---

### **PREREQUISITE KNOWLEDGE**

Operators do not have to read all the sections in this manual, and no prerequisite knowledge is required. System supervisors are supposed to possess a general knowledge of SINTRAN III.

---

### **THE MANUAL**

This manual provides the system supervisor with the knowledge to perform the tasks required. All users of the manual should read chapters 1, 2, 3 and 13. A short glossary is included in appendix A.

For those who supervise ND-500 installations, some tasks come in addition to those of an ND-100 system supervisor. The ND-500 exclusive operations will be described in separate sections.



---

## RELATED MANUALS

---

The system supervisor ought to know the following manuals:

SINTRAN III Timesharing/Batch Guide (ND-60.132)  
SINTRAN III Reference Manual (ND-60.128)  
SINTRAN III Real Time Guide (ND-60.133)

The manuals ND-100 Reference Manual (ND-06.014) and ND-500 Reference Manual (ND-05.009) contain hardware information. The manual ND-500 Loader/Monitor (ND-60.136) describes the extended facilities of an ND-500 computer system.

---

## NOTATIONS USED IN THIS MANUAL

---

SINTRAN III commands are written with the prompt character @. It is not keyed in by the user. Commands are written in UPPERCASE letters in this manual, but lower case letters are also accepted as input.

Command definitions show parameters in angular brackets. Parameters with a default value are also enclosed in parenthesis. For example, <file name> has no default value, but (<output file>) has. Optional parameters are enclosed in square brackets, eg., [<unit>].

Commands and parameters can be abbreviated as long as they do not become ambiguous. Omitted parameters will be asked for. User input is underlined in the examples. Each user input line must be terminated by carriage return (CR).

Parameters are separated by commas in this manual, but spaces may also be used. A slanted line in a parameter, eg., <input/output> indicates that one of the alternatives should be specified.

Octal numbers are written with a trailing B, eg., 31 is entered as 31B. Commands expecting decimal input also accept octal numbers. The term K, as used in 128 Kwords, means 1024. The term Mbyte indicates megabyte, ie., 1,000,000 bytes. A word is a 16-bit word unless otherwise specified.

---

## CHANGES FROM PREVIOUS VERSION

---

Major parts of the manual are rewritten. The structure is reorganized. The manual is updated for version J of SINTRAN III.

Changes from the previous version of the manual (ND-30.003.06) are indicated by a change bar.

## TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
<b>1    SYSTEM SUPERVISION OVERVIEW</b> . . . . .	<b>1</b>
1.1    User Categories . . . . .	1
1.2    Job Description for System Supervisor . . . . .	2
1.3    The Console . . . . .	3
1.4    System Supervisor Commands . . . . .	4
 <b>2    STARTING AND STOPPING THE INSTALLATION</b> . . . . .	 <b>5</b>
2.1    The ND-100 Panel . . . . .	5
2.2    Starting Procedure . . . . .	6
2.3    When to Stop the Installation . . . . .	7
2.4    The Different Disks . . . . .	7
2.4.1    The Panel of the SMD Disks . . . . .	8
2.4.2    The Panel of the HAWK Cartridge Disk . . . . .	9
2.4.3    The Panel of the Big Cartridge Disk . . . . .	10
2.4.4    The Panel of the MMD Disks . . . . .	11
2.5    Starting the Disks . . . . .	12
 <b>3    FILE SYSTEM SUPERVISION</b> . . . . .	 <b>13</b>
3.1    Files . . . . .	13
3.1.1    Creating Files . . . . .	15
3.1.2    Shrinking, Renaming and Deleting Files . . . . .	16
3.1.3    Scratch Files, Peripheral Files and Segment Files . . . . .	16
3.1.4    Retrieving Information about Files . . . . .	17
3.2    Directories . . . . .	19
3.2.1    Creating a Directory . . . . .	21
3.2.2    Enter, Rename, Set Default and Release Directory . . . . .	24
3.2.3    Retrieving Information about Directories . . . . .	25
3.2.4    Efficient Directory Structures . . . . .	27
3.2.5    Two Examples of Directory Lay Out . . . . .	29
3.2.6    Directory Maintenance and Correction . . . . .	34
3.3    Users . . . . .	34
3.3.1    Creating and Deleting Users . . . . .	34
3.3.2    Passwords . . . . .	36
 <b>4    SUBSYSTEMS</b> . . . . .	 <b>37</b>
4.1    The Batch System . . . . .	37
4.2    Data Communication and Network Systems . . . . .	38
4.2.1    Commands to Initiate and Terminate ND-NET . . . . .	38
4.2.2    COSMOS . . . . .	40
4.2.3    Remote Job Entry . . . . .	40

Section	Page
4.3 MAIL . . . . .	41
4.4 Reentrant Subsystems . . . . .	44
<b>5 ACCOUNTING . . . . .</b>	<b>47</b>
5.1 Files Used by the Accounting System . . . . .	48
5.2 Accounting Commands . . . . .	52
5.3 Dumping the Accounting Tables . . . . .	54
5.4 The ACCOUNTING SERVICE PROGRAM . . . . .	54
5.4.1 Command Summary . . . . .	55
5.4.2 Detailed Description of Commands . . . . .	56
<b>6 PERIPHERALS AND THE SPOOLING SYSTEM . . . . .</b>	<b>61</b>
6.1 Peripherals . . . . .	61
6.1.1 File Access For Peripheral Files . . . . .	62
6.1.2 Clearing Devices . . . . .	64
6.2 The Spooling System . . . . .	64
6.2.1 Preparing Spooling Files . . . . .	64
6.2.2 Spooling System Overview and Command Summary . . . . .	65
6.2.3 Spooling Commands for User SYSTEM . . . . .	68
6.2.3.1 Start and Stop Spooling Programs . . . . .	69
6.2.3.2 Give and Take Spooling Pages . . . . .	69
6.2.3.3 Define Spooling Conditions . . . . .	70
6.2.3.4 Spooling Form . . . . .	70
6.2.4 Spooling Commands for Public Users . . . . .	71
6.2.4.1 List Spooling Queue . . . . .	71
6.2.4.2 Stop, Start, Restart and Abort Print . . . . .	71
6.2.4.3 Backward and Forward Space Print . . . . .	72
6.2.4.4 Move Spooling Queue Entry . . . . .	72
6.2.4.5 Remove and Delete Spooling Files . . . . .	73
6.2.4.6 Append Spooling File . . . . .	73
6.2.4.7 Set Number of Print Copies . . . . .	74
6.2.4.8 List Spooling Form . . . . .	74
6.2.4.9 Spooling Pages Left . . . . .	74
<b>7 STOPPING, STARTING AND LOADING SINTRAN III . . . . .</b>	<b>75</b>
7.1 Stopping SINTRAN III . . . . .	76
7.1.1 Stopping the Activity in the System . . . . .	77
7.1.2 MOPC . . . . .	78
7.2 Warm Start . . . . .	79
7.2.1 SAVE AREA, IMAGE AREA, and SINTRAN III RESIDENT . . . . .	79
7.2.2 The Warm Start Procedure . . . . .	80
7.3 Cold Start . . . . .	83
7.3.1 Initial Commands . . . . .	84
7.3.2 Initializing Background Processes . . . . .	85
7.4 Initial Loading . . . . .	89
7.4.1 Formatting a Disk Pack . . . . .	90
7.4.2 Disk Space for User SYSTEM . . . . .	91



<u>Section</u>	<u>Page</u>
7.4.3	Disk Space Needed by User Scratch . . . . . 92
7.4.4	The Segment Files . . . . . 93
7.4.5	An Example of Initial Loading . . . . . 93
7.4.6	Explanation of the Example . . . . . 99
7.4.7	Loading Subsystems . . . . . 107
7.5	Loading a New SINTRAN III Version . . . . . 108
7.6	Additional Procedures for ND-500 Installations . . . . . 111
7.6.1	Loading ND-500 Software from Floppy Disks . . . . . 111
7.6.1.1	Loading to Segments . . . . . 112
7.6.1.2	The Swap Files . . . . . 112
7.6.1.3	Defining Memory Configuration and Swap Files . . . . . 112
7.6.2	Starting the ND-500 . . . . . 113
7.6.3	Stopping the ND-500 . . . . . 114
7.6.4	Loading Subsystems on the ND-500 . . . . . 115
7.6.4.1	The LINKAGE LOADER . . . . . 115
7.6.4.2	Example of Loading a Subsystem on the ND-500 . . . . . 116
<b>8</b>	<b>MEASURING AND IMPROVING THE SYSTEM PERFORMANCE . . . . . 117</b>
8.1	Measuring RT programs . . . . . 118
8.1.1	Preparation for Measuring RT Programs . . . . . 119
8.1.2	Results from Measuring RT Programs . . . . . 121
8.2	Relative Use of the CPU . . . . . 122
8.3	Measuring Use of an Address Space . . . . . 122
8.4	Measuring the Use of Physical Memory . . . . . 123
8.5	Listing CPU Time Used . . . . . 124
8.6	Diagnosis . . . . . 125
8.7	Solution . . . . . 126
<b>9</b>	<b>ERROR SUPERVISING . . . . . 127</b>
9.1	Error Logging . . . . . 128
9.2	Nonfatal Errors . . . . . 129
9.2.1	Error Messages from Central Part of SINTRAN III . . . . . 129
9.2.2	Summary of Run Time Error Messages . . . . . 130
9.2.3	Detailed Description of Run Time Errors . . . . . 133
9.3	Fatal Errors . . . . . 148
9.3.1	Main Groups of Error Symptoms . . . . . 148
9.3.1.1	The Machine has Stopped . . . . . 148
9.3.1.2	The Machine Hangs . . . . . 149
9.3.1.3	The Machine does not Function Properly . . . . . 149
9.3.2	Collecting Data for Error Identification . . . . . 149
9.3.2.1	What Should be Done before Errors Occur . . . . . 149
9.3.2.2	What to do when an Error has Occurred . . . . . 150
9.3.3	Use of the MEMTOF Dump . . . . . 152
9.3.3.1	How to Prepare and Use MEMTOF . . . . . 152
9.3.4	Some Easily Diagnosed Error Situations . . . . . 153
9.3.4.1	The ERRFATAL Routine . . . . . 153
9.3.4.2	SINTRAN III will not Start . . . . . 154
9.3.4.3	Memory Over Writing . . . . . 154
9.3.4.4	General Information about Hang Situations . . . . . 155

<u>Section</u>	<u>Page</u>
9.4	Terminals that do not Respond . . . . . 155
9.5	TELEFIX . . . . . 156
<b>10</b>	<b>PREVENTIVE MAINTENANCE AND TEST PROGRAMS . . . . . 159</b>
10.1	Maintenance to be Done by the Owner . . . . . 159
10.2	Loading Test Programs . . . . . 160
10.3	The FILE SYSTEM INVESTIGATOR . . . . . 161
10.3.1	Directory Consistency Check . . . . . 162
10.3.2	Correcting Errors . . . . . 163
10.3.3	FILE SYSTEM INVESTIGATOR Example . . . . . 165
10.3.4	Errors in a Directory Entry . . . . . 168
10.3.5	Errors in the User Files . . . . . 169
10.3.6	Errors in the Bit File . . . . . 169
10.3.7	More than One Version of a File or No Version Number 1 . 169
10.4	CPU Test Programs . . . . . 170
10.5	Memory Test Programs . . . . . 170
10.6	Input and Output Device Test Programs . . . . . 171
10.7	Test Programs to Run when the System has Stopped . . . . . 172
10.7.1	Memory Test Using the Microprogram . . . . . 172
10.7.2	Configuration Investigation . . . . . 173
<b>11</b>	<b>PATCHING . . . . . 175</b>
11.1	SINTRAN III Commands for Patching . . . . . 178
11.2	Patching with FMAC . . . . . 179
11.3	Patching with DMAC . . . . . 180
11.4	Patching with the SINTRAN SERVICE PROGRAM . . . . . 181
11.5	The System Variable ENDCOR . . . . . 182
11.6	Patching while SINTRAN III is Stopped . . . . . 183
<b>12</b>	<b>SINTRAN SERVICE PROGRAM REFERENCE . . . . . 185</b>
12.1	Command Summary . . . . . 186
12.2	Detailed Description of Commands . . . . . 189
<b>13</b>	<b>BACKUP . . . . . 225</b>
13.1	Stand-alone Programs for Taking Backup . . . . . 226
13.2	SINTRAN III Commands for Taking Backup . . . . . 228
13.3	The BACKUP SYSTEM . . . . . 229
13.3.1	Commands Used for Copying . . . . . 230
13.3.2	Handling Volumes on Magnetic Tape and Floppy Disks . . . 232
13.3.3	Copying a Users Files to a Volume . . . . . 233
13.3.4	Copying All Files in a Directory . . . . . 234
13.3.5	Advanced File Selection . . . . . 235
13.3.6	Modifying the Copying Mode . . . . . 236
13.3.7	Incremental Backup . . . . . 239
13.4	Personal Backup . . . . . 241

<u>Section</u>	<u>Page</u>
----------------	-------------

## **APPENDIXES**

A	Glossary . . . . .	244
B	Binary and Octal Numbers . . . . .	248
C	Device Names . . . . .	250
D	Example of a Disk Layout . . . . .	254
E	System Layout of Physical Memory . . . . .	256
F	File System Layout on Disk . . . . .	258
G	Physical Layout of Disks . . . . .	264
H	Mode Files Used to Start Up the System . . . . .	268
I	Example of a Software System Report . . . . .	276
J	Example of an ND Software Library Description . . . . .	278
K	ND-100 Switch Settings . . . . .	280
L	SINTRAN III Order Form . . . . .	284
	Index	288



LIST OF FIGURES

<u>Title</u>	<u>Page</u>
1. A Console . . . . .	3
2. The Panel of an ND-100 Computer . . . . .	5
3. The Panel of the SMD Disks . . . . .	8
4. The Panel of the HAWK Cartridge Disk . . . . .	9
5. The Panel of the Big Cartridge Disk . . . . .	10
6. The Panel of the MMD disks . . . . .	11
7. The Use of Spooling Forms . . . . .	66
8. User Files and Spooling Files in the Spooling Queue . . . . .	67
9. SAVE AREA, IMAGE AREA, and SINTRAN III RESIDENT . . . . .	79
10. A Warm Start Overview . . . . .	81
11. Overview of a Cold Start . . . . .	87
12. Initial Loading . . . . .	94
13. An Example of Incremental Backup . . . . .	239

LIST OF TABLES

<u>Title</u>	<u>Page</u>
1. User Categories . . . . .	1
2. Some Standard File Types . . . . .	14
3. The Organization of the Directories . . . . .	30
4. Organization of the Six Directories . . . . .	33
5. Remote Job Entry Manuals . . . . .	40
6. The file PROJNAM:DATA . . . . .	48
7. The file RTPROJ:DATA . . . . .	49
8. The Structure and Contents of ACCOUNTS:DATA . . . . .	50
9. Format of the Records in the file AACCOUNT:DATA . . . . .	52
10. File Access for Input Devices . . . . .	62
11. File Access for Output Devices . . . . .	63
12. File Access for Input and Output Devices . . . . .	63
13. File Access for Peripherals with Spooling . . . . .	63
14. The System Files . . . . .	91
15. Questions Related to System Performance . . . . .	117
16. Interrupts per Samples for Some Terminals . . . . .	120
17. The Interrupt Levels . . . . .	125
18. Commands in the FILE SYSTEM INVESTIGATOR . . . . .	162
19. Addresses in a Directory . . . . .	163
20. The SINTRAN SERVICE PROGRAM Commands . . . . .	186
21. Symbolic Names in Data Fields . . . . .	191
22. System Variables Available for Patching . . . . .	193
23. BACKUP SYSTEM Commands . . . . .	230
24. Subcommands in COPY-USERS-FILES and MULTIUSER-COPY . . . . .	231
25. The BACKUP SYSTEM SELECTION Commands . . . . .	236
26. Commands in the SERVICE PROGRAM CUF . . . . .	238
27. Binary, Octal, and Decimal Numbers . . . . .	249
28. Physical Characteristics of Various Disk Types . . . . .	268

---

## 1 SYSTEM SUPERVISION OVERVIEW

---

An ND-100 or ND-500 computer installation requires a manager. This person is referred to as the system supervisor. The system supervisor fills system programmer functions as well as operator functions. Operators control the daily operation of the system.

---

### 1.1 USER CATEGORIES

---

There are three different categories of persons who use the computer installation under SINTRAN III. These user categories are shown in table 1.

USER CATEGORY	USER NAME	AVAILABLE COMMANDS
Public users User RT User SYSTEM	Various RT SYSTEM	Public commands Public and real time commands All commands

Table 1. User Categories

Public users will access the computer system through a terminal or by submitting jobs for batch processing from files containing the necessary commands and data. These users are known to SINTRAN III by user names which must be input when logging in on a terminal or as the first command in a batch job. The user name may be protected by a password chosen by the user.

User RT is a privileged user, able to control real time programs. This user acts like any other public user to the system. The difference is the availability of an extended command set. The extra commands give the control functions necessary for the supervision of the RT programs. User RT should normally be protected by a password.

User SYSTEM has, in addition to the command sets of public users and user RT, a set of privileged commands capable of creating and deleting users, creating directories on disks, etc. This user should be protected by a password.



---

## 1.2 JOB DESCRIPTION FOR SYSTEM SUPERVISOR

---

The system supervisor should always have full knowledge of the computer installation with regard to the hardware configuration, the resource allocation, and the standard software in use. The work of the system supervisor is both to supervise the installation and to keep it running.

Supervising the installation implies:

### SUPERVISION

- Loading SINTRAN III.
- Loading subsystems.
- Implementing new patches.
- Creating new users and giving them disk space.
- Maintain the accounting system.
- Formatting disk packs.
- Taking backup of whole disks of files.
- Checking the file system, hardware and peripheral equipment with test programs at regular intervals.
- Assist users with problems.
- Install new subsystems when needed.

Keeping the installation running implies that the supervisor should:

### OPERATION

- Start, stop, and restart the computer system when necessary.
- Enter and remove disk packs.
- Initialize and print out error logs.
- Follow up error messages.
- Perform simple maintenance procedures.
- Know what action to take if a system failure occurs.
- Know the different test programs.
- Contact ND service when necessary.

Additionally, the system supervisor should keep a library of all documentation, and read the ND Bulletin which is published every month. The ND Bulletin contains information for ND's customers.

Of special interest to the system supervisor are the software system reports (SSRs). These inform the system supervisor of any new modifications and corrections to software. The system supervisor should notify other users. Newly registered programs and new program releases are also announced in the ND Bulletin.

The system supervisor logs in as user SYSTEM and the system supervisor should be the only one allowed to do so. A password should be created and used by the system supervisor. This system password should be changed regularly by the command @CHANGE-PASSWORD.

---

### 1.3 THE CONSOLE

---

The system supervisor will sometimes need to use the console to communicate with the system. This terminal outputs information on paper instead of on a video display screen. A console is shown in figure 1.

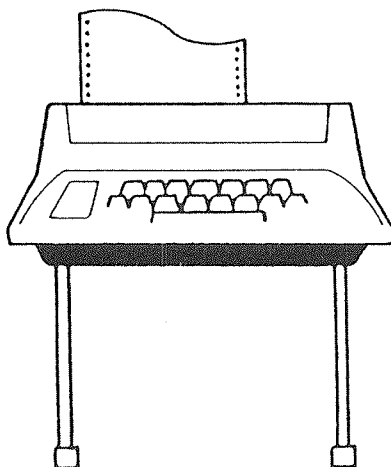


Figure 1. A Console

The console is different from other terminals in three respects:

- 1) Error messages from RT programs are normally printed on the console. Optionally, this may be altered to another terminal by the command @SET-ERROR-DEVICE. In this manual, the console is assumed to always be the error device.
- 2) The console is necessary to load and start SINTRAN III. It is generally used for operational control of the ND-100.
- 3) When the command @SET-UNAVAILABLE is given, it is impossible to log in from other terminals than the console. This condition may be reset by the command @SET-AVAILABLE. Both commands are restricted to user SYSTEM.

Note that subsystems which use the full terminal screen such as PED and NOTIS WP, must be used in line mode from the console.

---

#### 1.4 SYSTEM SUPERVISOR COMMANDS

---

The system supervisor uses the extended command set of user SYSTEM to control and maintain the computer system. In addition, user SYSTEM can have space for files and execute jobs in the same way as public users. User SYSTEM may log in from any terminal in the installation. He or she may, like other users, be logged in from more than one terminal at a time.

To obtain easy access to the shared resources of the installation, eg., printers, floppy disk units, text editors, and compilers, user SYSTEM is normally the owner of peripheral files and SINTRAN III subsystems. User SYSTEM's files are automatically searched when another user refers to files not found in his or her own user area.

Some powerfull commands are only available to the system supervisor. For this reason, only the system supervisor should have access to user SYSTEM. Only he or she should know how to log in as SYSTEM. If others are allowed to log in as SYSTEM, they should have a good knowledge of the installation and of SINTRAN III.

---

## 2 STARTING AND STOPPING THE INSTALLATION

---

This chapter describes how the computer, the disks, the printers and the terminals are started and stopped. See chapter 7 for a description of how SINTRAN III is loaded and initialized the first time.

---

### 2.1 THE ND-100 PANEL

---

The control panel of an ND-100 computer is shown in fig. 2. The display panel is optional while the control panel is standard on all machines. The buttons on the panel can only be used with the key in the ON position.

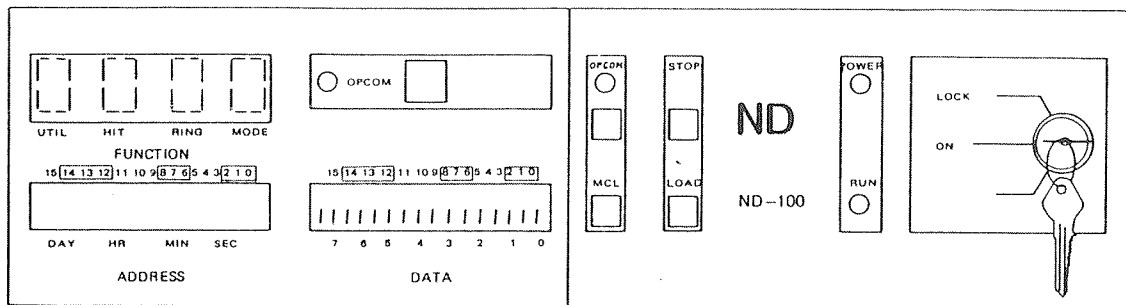


Figure 2. The Panel of an ND-100 Computer

The MCL button will initialize the computer. MCL is an abbreviation of MASTER CLEAR. The execution of programs stops, and MOPC is given control. MOPC is the microprogrammed operator's control that allows operational control of the computer. It may, for example, be used to load and execute test programs, examine registers, and modify the contents of main memory or your virtual memory.

MOPC can only be used from the console. The character # will be output when MOPC is ready to accept commands. An exclamation mark (!) will restart execution of the stopped programs. MOPC is described in more detail in section 7.1.2.

The STOP button is used to stop the executing programs and enter MOPC. The contents of main memory will not be destroyed and the stopped program can be restarted by typing an exclamation mark (!) on the console.

The LOAD button will load and start a preselected program. This program is normally SINTRAN III, but other programs can be selected. See appendix K.

The OPCOM button is used to start MOPC without stopping the computer. Ordinary programs will be able to continue while MOPC is used from the console. MOPC is terminated by pressing the ESCAPE-key.

---

## 2.2 STARTING PROCEDURE

---

The system supervisor is responsible for stopping and starting the system. If the system has been stopped and the power turned off, the starting procedure is as follows:

### NORMAL START

- 1) Switch the computer on. That is, turn the panel key from the LOCK to the ON position, switch the power on and press the MCL button. It is good practice to always press the STOP button first, to make sure that the system is stopped.
- 2) Start the disks. The control panels of the most commonly used disks are shown in the following sections.
- 3) Turn on the console, i.e., switch the power on. Make sure it has the correct speed according to the console speed switch setting on the CPU card. See appendix K. Check that the local/line switch on the console is set to line.
- 4) Load SINTRAN III by pressing first the MCL and then LOAD button. The green RUN indicator will light up. Wait some minutes for the message SINTRAN III IS RUNNING to be printed on the console.
- 5) Turn the key on the panel to the LOCK position.
- 6) Switch on the other terminals and the different peripherals, eg., printers, magnetic tape stations, and floppy disk units.

Below is an example of output from the computer to the console, after having pressed the MCL and LOAD buttons:

```
BATCH NUMBER =          1
SINTRAN III RUNNING      -

PAGES FOR SWAPPING(OCT.):  425

***01.43.18 TERMINAL      670
***SYSTEM AVAILABLE****
```

Only two of the lines of text here are standard. Those are SINTRAN III RUNNING and PAGES FOR SWAPPING (OCT.). The other lines are output from a batch job and from special commands which are automatically executed when LOAD is pressed. See section 7.2.2.

---

### **2.3 WHEN TO STOP THE INSTALLATION**

---

It is advisable that both the computer and the disks run day and night. The disks may be stopped, but the power should be left on. During longer periods of idleness, the installation may be stopped and turned off, but normally this should only take place for hardware maintenance.

If the installation is to be stopped, it is important to turn the disks off before the computer is turned off. Section 7.1 describes how to stop SINTRAN III.

---

### **2.4 THE DIFFERENT DISKS**

---

Various disks are connected to ND computers. How to start a particular disk depends on which type you have. The following sections shows the control panels of the most commonly used disks. Appendix G and to some extent appendix F describe disks in more details.

---

**2.4.1 THE PANEL OF THE SMD DISKS**

---

The SMD-33/37/66/75-MB and SMD-288-MB have the same panel. It is shown in fig. 3. SMD means storage module drive, and MB means megabyte. A megabyte is one million bytes.

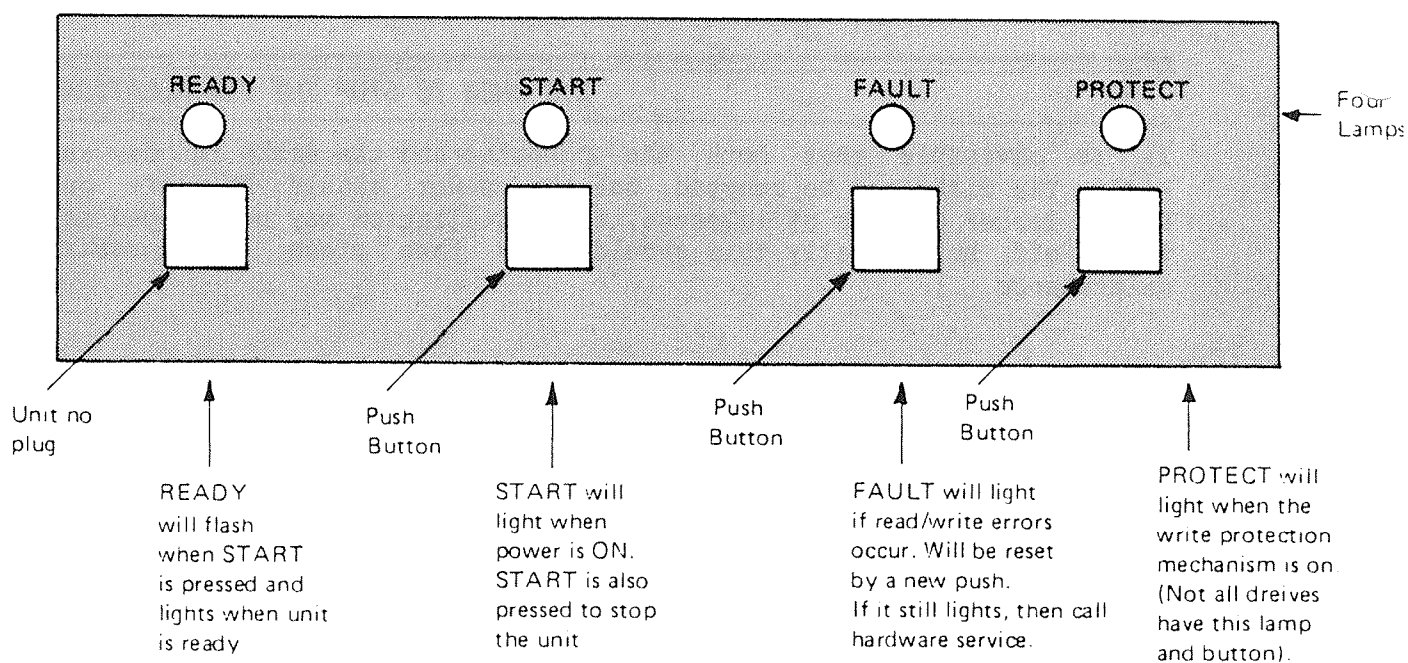


Figure 3. The Panel of the SMD Disks



---

### 2.4.2 THE PANEL OF THE HAWK CARTRIDGE DISK

---

The HAWK cartridge disk with 5 Mbyte removable and 5 Mbyte fixed disk packs are shown in fig. 4. When the disk unit is stopped, no buttons are illuminated.

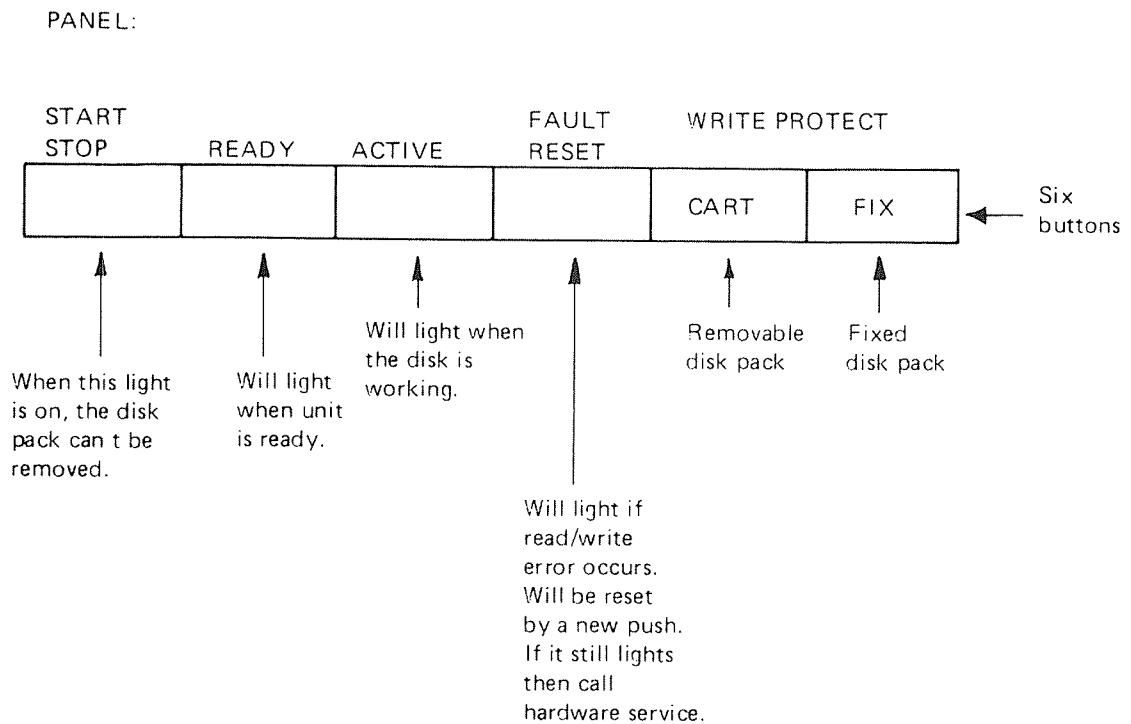


Figure 4. The Panel of the HAWK Cartridge Disk

### 2.4.3 THE PANEL OF THE BIG CARTRIDGE DISK

The Big cartridge disk (Phoenix disk) with 15 Mbyte removable and 15 Mbyte, 45 Mbyte or 75 Mbyte fixed disk packs have the panel shown in fig. 5.

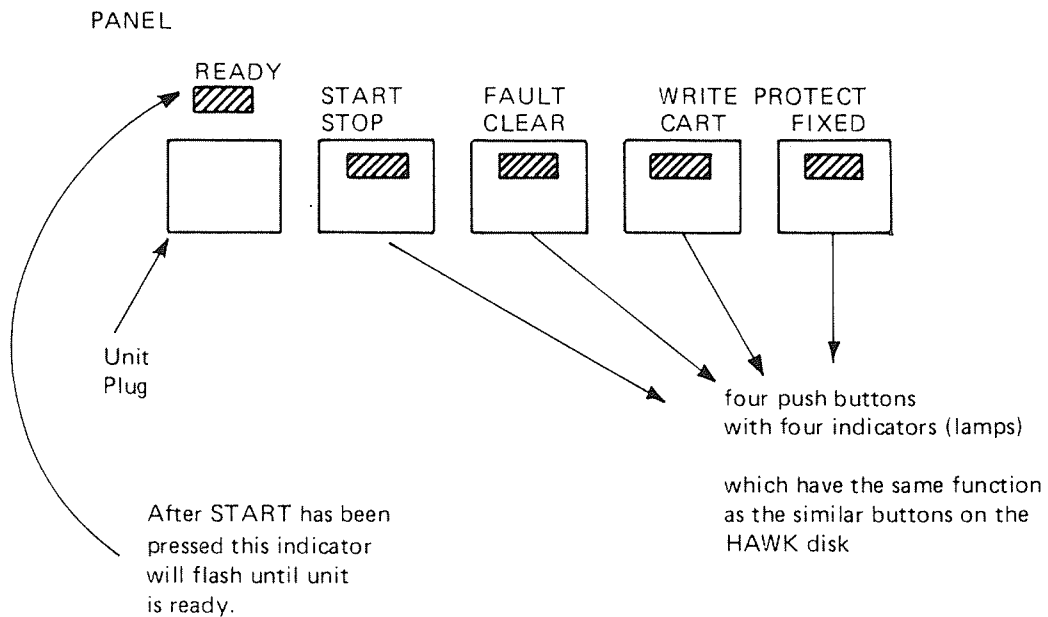


Figure 5. The Panel of the Big Cartridge Disk

---

#### 2.4.4 THE PANEL OF THE MMD DISKS

---

The panel of the MMD disks with 2 x 75 Mbyte disks is shown in fig. 6.

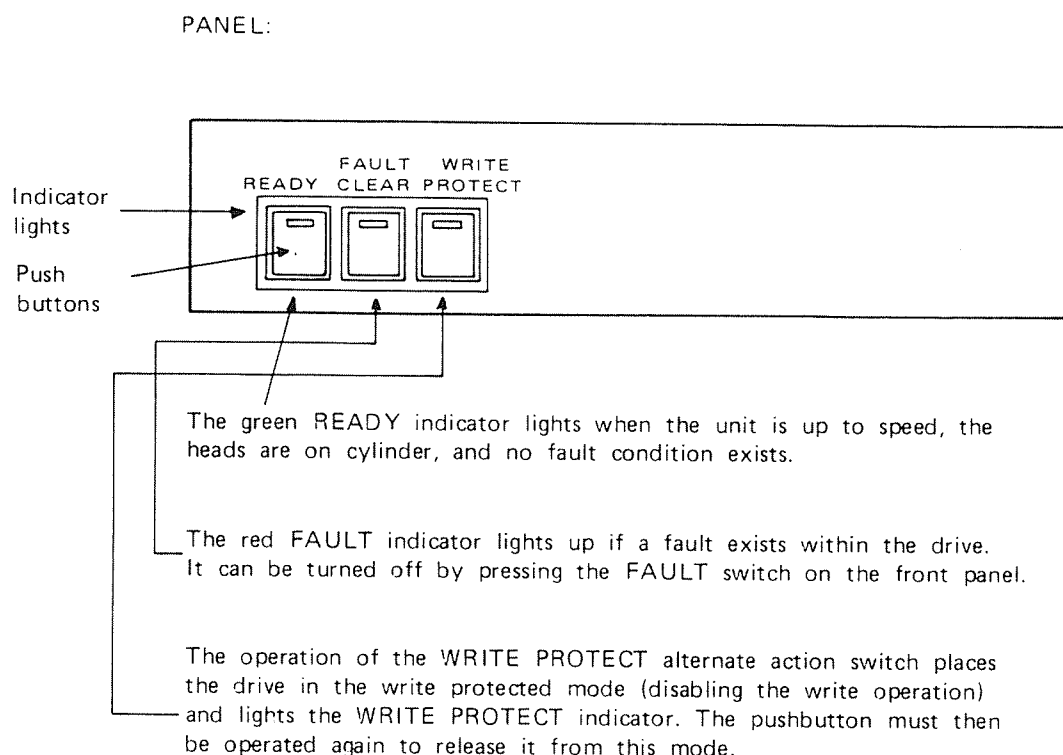


Figure 6. The Panel of the MMD disks

---

## 2.5 STARTING THE DISKS

---

SINTRAN III cannot work on a directory which is write protected, i.e., when the WRITE PROTECT button on the disk is pressed. The system will hang if you try to start it from a disk which is not ready. The same will happen if you try to enter a directory on a disk which is not ready. A light on the disk panel will normally indicate whether a disk is ready or not.

The disks are started by the following procedure:

### STARTING DISKS

- Make certain that a disk pack is mounted on the disk unit.
- Turn on the power switches on the back of the disk unit. There are two switches for SMD disks. The switches are normally located under the covers or behind a door.
- Press the START button on the disk panel.
- When the READY lamp lights up, the disk is ready to be accessed.

You mount a disk pack by the following procedure: Open the disk unit. Remove the bottom plastic cover of the disk pack and put the disk pack in. Twist the handle of the plastic cover til it can be removed. Do not mount or remove disk packs while SINTRAN III is running. How to stop SINTRAN III is described in section 7.1.

See appendix G for a detailed description of disks.

---

### 3 FILE SYSTEM SUPERVISION

---

This chapter describes system supervisor functions to be performed on files, users, and directories. How to test the directory structure is explained in section 10.3. The exact layout of the file system on a disk is described in appendix F.

---

#### 3.1 FILES

---

All kinds of information can be stored on mass storage files. File are allocated in pages of 1 Kwords. The pages of a file can be stored consecutively on the disk, or they may be scattered around with pointers to the pages used by the file. Some examples of information which can be stored in a file are:

- The text of a Fortran program.
- An executable program.
- Output from a program.
- A set of commands.
- A text document.
- A SINTRAN III subsystem.
- A routine library

A file is identified by a file name. This file name is used in commands to the file system. A complete file name has five parts as shown below, ie., the directory it is found on, the user who owns it, the actual name given to it, the file type and the version number. If the file name is abbreviated, it should be unambiguous.

(DIRECTORY:USER NAME)FILE NAME:FILE TYPE;VERSION

Files are organized into directories containing files for one or more users. A directory can be thought of as a catalogue. When a file is asked for, the system looks it up in the directory. When a new file is created, a new entry is made in a directory. Each disk or floppy disk in use has its own independent directory.

All users must be registered in a directory. Each file belongs to a user. The directory contains information about the disk space, ie., the size of the user area, available to each user.

A file type is used to designate the contents of the file. This file type may be set by the user as a maximum 4 character extension of the file name. It is separated by a colon. The file types shown in table 2 are commonly used as default in commands or subsystems.

Contents of the file	File type
Text Document	:TEXT
Formatted Text Documents	:OUT
High Level Language Program	:SYMB
Binary Relocatable format	:BRF
Executable Programs	:PROG
General Information	:DATA
Binary Programs	:BPUN
Form Definitions	:FORM
Macro File	:MCRO
ND Relocatable Format	:NRF

Table 2. Some Standard File Types

The manuals SINTRAN III Timesharing/Batch Guide (ND-60.132) and NRL - ND Relocating Loader (ND-60.066) explain the connection between subsystems and most default file types. It is recommended that the standard file types are used.

A file may exist in several versions. The version number is separated from the file type by a semicolon (;). The maximum number of file versions is 256. Version 1 is the newest version of the file.

Files may be created in four ways:

- ```
@CREATE-FILE <file name>, (<number of pages>)
```

Its pages are scattered around on the disk or floppy disk. This means that it has become an indexed file. However, if it is necessary to make it a contiguous file, that is, with its pages allocated on a contiguous area of mass storage, then the desired number of pages must given instead of 0 as second parameter of the command. The file type should be specified according to which subsystem it will be used under. If no directory or user name are given, the file will be created in your user area on your default directory. Default directories are explained in the next section.

- ```
@ALLOCATE-FILE <file name>, <page address>,  
                                     <number of pages>
```

4) The OPEN statement in application programs can be used to create a file. For example, this happens when formatting text files with NOTIS-TF. Then the :OUT file is created automatically.



---

### 3.1.2 SHRINKING, RENAMING AND DELETING FILES

---

Assume a user has a large file. After deleting parts of it in one of the editors NOTIS WP or PED the file will still occupy as many pages on mass storage as previously. The file might be effectively shrunk by renaming it, copying back to the original name and deleting the renamed file as shown in the following example. This will make more disk space available.

<pre> @RENAME-FILE FILE NAME: LETTER:TEXT NEW FILE NAME AND TYPE: X:TEMP @COPY TO DEVICE: "LETTER:TEXT" FROM DEVICE: X:TEMP @DELETE-FILE X:TEMP @ </pre>	<pre> Rename the old file.  Copy the renamed file to a new file with the old name. Delete the renamed file. </pre>
--	--

This will not work for files with holes in them. A hole is an unused, not allocated page in an indexed file. As shown in the example above, files may be deleted by the SINTRAN III command:

```
@DELETE-FILE <file name>
```

Another command, @DELETE-USERS-FILES, can be used to delete more than one file at a time. The command should be used with care.

---

### 3.1.3 SCRATCH FILES, PERIPHERAL FILES AND SEGMENT FILES

---

A scratch file is a working area belonging to a terminal or a batch processor. Each terminal and batch processor has its own scratch file. The scratch file will always be opened when logging in, and it is given file number 100. The scratch files belong to a user SCRATCH. See the command @SCRATCH-OPEN in the SINTRAN III Reference Manual (ND-60.128).

Peripherals such as magnetic tape units, printers, etc., may be treated by a user very much like sequential disk files. Peripherals are given file names by user SYSTEM, eg., LINE-PRINTER. A user may access the peripherals by using these names.

A SINTRAN III system contains from 1 - 4 segment files. These are contiguous mass storage files with file names SEGFILO:DATA to SEGFILE3:DATA. The segment files contain segments. Segments are contiguous areas on a disk used by SINTRAN III and RT programs.

---

### 3.1.4 RETRIEVING INFORMATION ABOUT FILES

---

To find out data about a specific file the command @FILE-STATISTICS may be useful. This command, which is available to all users, lists the following information:

- File index number.
- File name.
- Attribute (indexed, contiguous, peripheral, etc.).
- Access settings.
- Date of creation.
- Number of times opened.
- Last time opened for read and write.
- The size of the file in pages and in bytes.

For example, a file called REFMAN:TEXT is listed by the command:

```
@FILE-STATISTICS REFMAN,,  
FILE 11 : (PACK-TWO-685:SIJ)REFMAN:TEXT;1  
  (INDEXED FILE)  
  PUBLIC ACCESS : READ  
  FRIEND ACCESS : READ, WRITE, APPEND  
  OWN ACCESS : READ, WRITE, APPEND, COMMON, DIRECTORY  
  OPENED 25 TIMES  
  CREATED 11.22.28 JULY 4, 1983  
  OPENED FOR READ 09.04.53 OCTOBER 3, 1983  
  OPENED FOR WRITE 13.59.42 SEPTEMBER 30, 1983  
  34 PAGES , 68062 BYTES IN FILE
```

Another command @WHERE-IS-FILE, which also is available to all users, lists the following information:

- Any users that have opened the file, and from which terminal.
- Whether it has been reserved by an RT program.
- Whether it is free to use.
- Whether it has been opened by user RT.
- If it is a spooling file whether it is in use; ie., whether it has data in it waiting to be printed.

This command is most useful for collecting information about peripheral files. We want to list information about a particular PHILIPS printer. The printer is a peripheral file with 10 versions to hold files waiting to be output.

@WHERE-IS-FILE

FILE NAME: PHILIPS

PHILIPS;;1: RESERVED BY RT PROGRAM SPRT1

PHILIPS;;2: FREE TO USE

PHILIPS;;3: FREE TO USE

PHILIPS;;4: FREE TO USE

PHILIPS;;5: FREE TO USE

PHILIPS;;6: OPENED FOR RT PROGRAM USE

PHILIPS;;7: FREE TO USE

PHILIPS;;8: FREE TO USE

PHILIPS;;9: FREE TO USE

PHILIPS;;10: FREE TO USE

For user SYSTEM two other commands are useful to inspect file information. They are:

```
@DUMP-OBJECT-ENTRY [<directory name>:]<user name>,  
                  (<object number>),(<output file>)  
@CHANGE-OBJECT-ENTRY [<directory name>:]<user name>,  
                    (<object number>)
```

The first of these commands dumps, in octal, the object entry of a file on an output file. There is one object entry describing each file, eg., creation data, size, file name, disk address, access protections. To find out the exact meaning of the output, compare with the file system layout in Appendix F.

The second of these commands makes it possible to change one or more entries in the file object entry. The object number is the number output in front of each file name in the command @LIST-FILES.

---

### 3.2 DIRECTORIES

---

A directory is a logical concept by which the available space on a contiguous area of a disk or floppy disk is organized. It consists of a master block, a user file index and user files, an object file index block possibly with a subindex block, object files, a bit file, data file indexes and data files. The structure of a directory is explained in detail in appendix F.

Except for the data files, the manipulation of these entities is an overhead which should be reduced as much as possible. The worst effects of this have been eliminated in version I of SINTRAN III by incorporation of a disk cache for the file system. File system performance can be further improved by considering the directory structure described below:

The directory table is one of the critical tables in the file system. It is organized sequentially, and there is one entry for each directory in the system. Each entry in the table has a directory index. The first word of each entry marks the directory as entered or released, as main or default, or as reserved or not reserved for special use. The directory table is found on segment 6.

A main directory is where a user name can be defined and used by the file system, without requiring that user name to be defined on any other directory. Main directories should not be entered or released when the system is running. A main directories cannot be created on a floppy disk.

A related, but not identical, concept is a user's main directory. This is the main directory of the lowest directory index in which the user is defined. Therefore, if main directories are entered and released, a user's main directory may change.

In addition, when a user logs in, his or her main directory is determined at that time and stored on the terminal's system segment. Because of this, the file system may even determine two different main directories for the same user working on a specific terminal at a given time. This can be a source of problems in the file system.

A user's default directory is the default directory of lowest index in which a user has file space, ie., it is the directory to be searched for the user's files. A default directory is therefore a directory which is searched to determine a user's default directory. A main directory is always a default directory.

The search strategy used by the file system is as follows: If a user name is supplied to the file system and the user's main directory is to be determined, the file system starts with the lowest directory index. If it is a main directory, all user names are searched for a match. It then searches the next main directory and so on, until all main directories have been searched.

If an exact match, ie., identical names including length, is found between the supplied name and a user name on a directory, the search need go no further. When all directories have been searched, and if only one match is found, this one is used.

If no match is found, the user does not exist. If more than one match is found, the user name is ambiguous. Note that even after more than one match is found, the search continues, because there may be an exact match.

Note that a match is here considered to be where the input user name is a subset of the exact name. An exact match is where the input and the name as defined in the user table, are identical.

---

### 3.2.1 CREATING A DIRECTORY

---

A new directory on a disk or floppy disk is created by the command:

```
@CREATE-DIRECTORY <directory name>, <device name> [,<unit>]  
                  [,<'F' or 'R'>] [,<subunit>], (<bit file address>)
```

The parameters have the following meaning:

**DIRECTORY NAME:**

The name which will be written onto the new directory. A maximum of 16 alphanumeric characters and hyphens (-) is permitted.

**DEVICE NAME:**

The name of the mass storage device controller to which the new directory is connected. The device names of the controllers, eg., DISC-2-75MB-1, DISC-288MB-2 and FLOPPY-DISK-1, are found in appendix C.

**UNIT:**

The device unit number. It is only specified if more than one device unit is connected to the same mass storage device controller. If only one unit is installed, the unit number is neither necessary nor permitted in the command. Unit numbers start at 0.

**'F' OR 'R':**

F must be specified for a fixed cartridge disk pack and R for a removable cartridge disk pack. The parameter is only required for the 10/30/60/90 Mbytes cartridge disks.

**SUBUNIT:**

Applies only to fixed big cartridge disks, 2 x 75 Mbyte, and 3 x 75 Mbyte or 60/90 Mbyte drives.

**BIT FILE ADDRESS:**

This is the page address of the first page of the bit file. A bit file has one bit for each page on the disk to mark the page as free or used. One page of the bit file covers 16,384 pages on a directory. If a page address is not specified, the file system will place the bit file in the middle of the disk, or as near as possible to the middle, if bad pages occur in the middle. A bad page is a page that cannot be used. It will be marked used in the bit file.

For most systems the middle is the best place for the bit file. If a contiguous file of size greater than half the directory size is wanted, the position of the bit file can be moved. The size of the different disks is given in appendix G. You are given a choice of either a low or high page number. A low page number is best when data base files are to be created with the @CREATE-FILE command. A high page number is best for large SINTRAN III segment files.

If a disk is to contain large contiguous files together with indexed files, then careful consideration should be given to the position of the files. In particular, the contiguous areas should be positioned to restrict the indexed, spooling and temporary files to one area of the disk. The bit file should be in the middle of this area. For example, a 75 Mbyte directory has 36,945 pages:

1 page for the master block  
63 pages for SINTRAN:DATA  
64 pages for MACM-AREA:DATA  
6500 pages for SEGFILO:DATA

The files above must follow in this order on the disk starting from page 0. The three files are needed by all SINTRAN III systems. The installation is running a SIBAS database, which has 4 files. These follow SEGFILO:DATA in the specified order and with the specified sizes:

ORDER-DB:DATA	1600 pages
ORDER-SYS:DATA	6000 pages
ORDER-DA-1:DATA	8000 pages
ORDER-DA-2:DATA	8000 pages

The bit file, user files and other files then occupy the remaining 7,317 pages. Thus the bit file should be placed at disk address 33,286. Such placement of files on the disk will, in general, reduce the movement of the disk reader head and consequently the waiting time for the disk.

For example, to create a directory called FLOPPY-PACK on the floppy disk device controller 1, unit 1, you can type:

@CREATE-DIRECTORY  
DIRECTORY NAME: FLOPPY-PACK  
DEVICE NAME: FLOPPY-DISC-1  
DEVICE UNIT: 1  
BIT FILE ADDRESS:



To create a directory called PACK-FOUR on the fixed cartridge on disk unit 1, you can type:

@CREATE-DIRECTORY PACK-FOUR, DISC-10MB-1, 1, F, ,

When the command is executed, the type of disk is first checked by the file system. If it is a Winchester disk (DISC-14/21/23/45-MB-1/2 or an ECC disk 30/60/90/75/288, 2 x 75, 3 x 75) disk, then only the bit file area and the master block, ie., page 0 of the directory, will be tested for errors. Such disks should therefore be formatted with spare track reallocation by the test program DISC-TEMA before use. DISC-TEMA is described in the manual Test Program Description (ND-30.005).

On other types of disk the file system will test all pages on the disk, and mark any bad ones as in use in the bit file. After the bit file is written, the master block is written to page 0 and the operation is complete.

**WARNING!**

An old directory will be completely destroyed, if a new directory is created on the same disk or floppy disk. Pointers to pages used by different files in the old directory will be overwritten. Pages previously used will be marked as free in the new bit file, but the data still remains. Thus it is possible, by using the stand-alone program FILE SYSTEM INVESTIGATOR to investigate a destroyed directory. However, detailed knowledge of the file system is then necessary. See section 10.3. A stand-alone program is a program that can be executed when SINTRAN III is stopped or not loaded.

---

### 3.2.2 ENTER. RENAME. SET DEFAULT AND RELEASE DIRECTORY

---

When a disk pack or a floppy disk is put on a disk unit, the directory on it must be entered. The file system will then read certain data from the directory. This is done by the command:

```
@ENTER-DIRECTORY (<directory name>), <device name> [,<unit>]  
                  [,<'F' or 'R'>] [,<subunit>]
```

The parameters are the same as for @CREATE-DIRECTORY. The first entered directory automatically becomes a main directory. If the specified name matches the name found on the device, the directory is entered, otherwise an error message is given. The default directory name is the one found on the disk.

For example, to enter the floppy disk directory created in the previous example, you can type:

```
@ENTER-DIRECTORY FLOPPY-PACK,FLOPPY-DISC-1,1
```

Use the following command to set a directory as a default directory:

```
@SET-DEFAULT-DIRECTORY <directory name>
```

If a directory is not a default directory, the directory name must be given as a prefix, when accessing files on it. This is also the case if a user has space in more than one default directory.

A directory must be released before the disk or floppy disk is removed from the disk drive. The directory may only be released if no files are opened on it, and no users are logged in with this directory as their main or default directory. The command to use is:

```
@RELEASE-DIRECTORY <directory name>
```

After the directory is released, it may be entered again, or another disk or floppy disk may be mounted on the unit. The directory may also be entered on another ND computer installation.

It may happen that a user has forgotten to release a directory on a floppy disk, before taking it out. In such cases, mount any other floppy disk with a directory, release the old directory and then ignore the error message that is output.

An entered directory can be given a new name by the command:

```
@RENAME-DIRECTORY <old directory name>,<new directory name>,  
                <device name> [,<unit>] [,<'F' or 'R'>] [,<subunit>]
```

This first of the following commands defines a directory as a main directory. The next commands remove the effect of earlier commands to set default and main directories.

```
@SET-MAIN-DIRECTORY <directory name>  
@CLEAR-DEFAULT-DIRECTORY <directory name>  
@CLEAR-MAIN-DIRECTORY <directory name>
```

A system may have up to 16 main directories, each with a maximum of 256 user names. It is thus possible to have 4096 different user names on one system. All users must be defined in a main directory in order to access files on other directories where they also are defined. They need not have space in the main directory.

---

### **3.2.3 RETRIEVING INFORMATION ABOUT DIRECTORIES**

---

Two commands are used to retrieve information about directories. The first will list the names of the entered directories matching the directory name parameter. The second will list more information about the directory.

```
@LIST-DIRECTORIES-ENTERED (<directory name>), (<output file>)  
@DIRECTORY-STATISTICS (<directory name>), (<output file>)
```

The last command will list the unit number, the directory status, ie., default and main, and the unreserved and unused space on the specified output file. For example, the command below will list the information on the terminal which is the default output file.

@DIRECTORY-STATISTICS PACK-ONE.

```
DIR INDEX 1 : DISC-2-75MB-1 UNIT 0 SUBUNIT 0 : PACK-ONE
      (MAIN AND DEFAULT DIRECTORY)
      4137 PAGES UNRESERVED AND 13667 PAGES UNUSED OUT OF
                                         36945 PAGES
      MAXIMUM UNRESERVED CONTIGUOUS AREA ON DIRECTORY
                                         719 PAGES
```

The 4137 pages are unreserved, but not contiguous space. These commands may, like all statistical commands, be given by any user, but are of special interest to user SYSTEM.

Two commands affect the directory entry part of the master block. The directory entry contains the directory name, a pointer to the user table, a pointer to the file object table, and a pointer to the bit file. The commands are only permitted for user SYSTEM.

```
@DUMP-DIRECTORY-ENTRY <device name> [, <unit>] [, <'F' or 'R'>]
                        [, <subunit>], (<output file>)
@CHANGE-DIRECTORY-ENTRY <device name> [, <unit>]
                        [, <'F' or 'R'>] [, <subunit>] (subcommands)
```

The first of these commands dumps the directory entry in octal format. Appendix F shows the layout of the directory entry and the master block.

The second command makes it possible to change the contents of the directory entry of a directory which is not entered. The subcommands needed to change the directory entry are the same as for @LOOK-AT. See the SINTRAN III Reference Manual (ND-60.128). The command should be used with care.

---

### 3.2.4 EFFICIENT DIRECTORY STRUCTURES

---

This section contains advice on the definition of directories and users. Efficient use of the file system will improve the system performance. Appendix F explains the undelined terms in this section.

- 1) Do not have more main directories than you need. Pack the users on the main directories so that they occupy the minimum number of pages.
- 2) Do not have unused user names if this means that you must use an additional page in the user file index block. There are 32 user entries per page. It is worth combining users' files to achieve this.
- 3) Do not have more default directories than you need or can use.
- 4) A user should never be defined in more than one main directory, unless it is absolutely unavoidable.
- 5) Change the order of the directory entries in the directory table, so that directory index 0 is the main directory with users SYSTEM, user SCRATCH, user RT and others whose files are most frequently used. User entries with the lowest user indexes should be for users whose files are most used. The standard users BPUN-FILES and FLOPPY-USER should be as high as possible, eg., a third main directory if 3 main directories exist. The SINTRAN SERVICE PROGRAM command SWAP-DIRECTORY-ENTRIES can move directory entries. See chapter 12.
- 6) File system access during log in is reduced on a multi main directory system by specifying the main directory before the user name.

- 7) User's files should be organized so that those accessed most frequently have the lowest object number. The @FILE-STATISTICS command gives the number of times a file has been accessed. For example, there is no reason for SINTRAN:DATA, MACM-AREA:DATA and SEGFILO:DATA to have indexes 0, 1 and 2, but they almost always do. They can be moved by patching, but not by copying. Patching means to change the binary contents of files or memory locations.

First dump the contents of the file object entry by the command @DUMP-OBJECT-ENTRY. Patch the contents of this file object entry into the free object entry you choose. Use the previously described command @CHANGE-OBJECT-ENTRY, thereby also setting the used flag of this object entry. See appendix F for the organization of the file object entry, and section 10.3.2 on its manipulation. Remember to change the object index and the pointers to the next and previous versions, so that they agree with the new object file index. Finally, patch the original object entry and set word 0 to 0. Thus the entry used bit is reset, and the entry is free to use. Patching is described in chapter 11.

- 8) Organize the directory table carefully to avoid problems with changing user's main and default directory index. The directory with user SYSTEM and the segment file or files should be directory index 0. Any directories likely to be entered and released as main or default directories should have as high a directory index as possible. Special consideration should be given to which users are found there.
- 9) File access is further improved if file names are specified in full. This is inconvenient in interactive work, and that is why abbreviated file names were introduced. However, in mode and batch jobs full file names, user names and directory names ought to be used.

If a user is logged in and wishes to access files on his or her default directory, then no directory or user name should be given. Such information will slow down the access time. This applies to both batch and interactive work.

---

### 3.2.5 TWO EXAMPLES OF DIRECTORY LAY OUT

---

The first example is a system with a 60 Mbyte Phoenix disk. The users and their space requirements are listed below.

- 1 SYSTEM, 7000 pages (decimal).
- 2 SCRATCH, 1500 pages (approximately).
- 3 BASE, a user having the database files, 5000 pages.
- 4 CLERK-1,...,CLERK-5, five clerks dealing with the accounting ledgers. No pages, only the database is used.
- 5 WAGES-1,...,WAGES-5, five wages clerks. No pages
- 6 COSTING-1,...COSTING-10, ten clerks dealing with job costing. No pages.
- 7 STOCKCONTROL-1,...,STOCKCONTROL-3, PRODUCTION-1,..., PRODUCTION-3, six more clerks dealing with stock control and production. No pages.
- 8 SECRETARY-1,...,SECRETARY-8, eight secretaries, each using the NOTIS word processing system and some other subsystems. They use the file system facilities frequently throughout the day. Each has 1000 pages of file space.
- 9 The source code of the database applications is kept under user SOURCE. These files are accessed about once per week and occupy 2500 pages.
- 10 A further three users, APPLICS-1,...,APPLICS-3 have file space for developing application programs, documentation and general use of subsystems. They use the system nearly every day, but their use is not heavy. They each have 1000 pages of file space.
- 11 Five directors also have use of the system; DIREC-1,..., DIREC-5. They occasionally make use of systems like MERCUR, ACCESS, NOTIS-IR, but their use is occasional and not heavy. They each have 100 pages of file space.

Users 4 to 7 exist in order to control access to the computer system and the database and they have no file space. They make frequent use of file system facilities by their use of the machine. The four directories available, each with 7389 pages, should be organized as shown in table 3.

Dir. index	Disk surface	User index	User names	Number of pages	Notes
0	REMO- VABLE	0	SYSTEM	7000	Main and default directory. All users present
		1	SCRATCH	0	
		2	RT	0	
		3	BASE	0	
		4	CLERK-1	0	
		.	....	.	
		8	CLERK-5	0	
		9	WAGES-1	0	
		.	....	.	
		13	WAGES-5	0	
		14	COSTING-1	0	
		.	....	.	
		23	COSTING-10	0	
		24	STOCKCONTROL-1	0	
		.	....	.	
		26	STOCKCONTROL-3	0	
		27	PRODUCTION-1	0	
		.	....	.	
		29	PRODUCTION-3	0	
		30	SECRETARY-1	0	
		.	....	.	
		37	SECRETARY-8	0	
		38	SOURCE	0	
		39	APPLICS-1	0	
		.	....	.	
		41	APPLICS-3	0	
		42	DIREC-1	100	
		43	DIREC-2	100	
		44	DIREC-3	100	
		45	DIREC-4	0	
		46	DIREC-5	0	
		47	FLOPPY-USER	0	
1	FIXED UNIT 0	0	SCRATCH	1500	Default directory
		1	BASE	5000	
		2	DIREC-4	100	
		3	DIREC-5	100	
2	FIXED UNIT 1	0	SECRETARY-1	1000	Default directory
		.	....	.	
		6	SECRETARY-7	1000	
3	FIXED UNIT 2	0	SECRETARY-8	1000	Default directory
		1	APPLICS-1	1000	
		.	....	.	
		3	APPLICS-3	1000	
		4	SOURCE	2500	

Table 3. The Organization of the Directories



The second example concerns a university which has an ND-500 system with a directory structure consisting of six 75 Mbyte directories. The following users with the stated space requirements must be arranged on these directories.

- 1 SYSTEM, 10,000 pages.
- 2 SCRATCH, 4000 pages.
- 3 BPUN-FILES, 1000 pages for bpun and similar files.
- 4 ANALYST-1,...,ANALYST-5, five system analysts making constant use of the system, each requiring 500 pages.
- 5 POSTGRAD-1,...,POSTGRAD-200, two hundred post graduate students, each requiring 250 pages.
- 6 STUDENT-3-1,...,STUDENT-3-320, three hundred and twenty final year students, each requiring 150 pages.
- 7 STUDENT-1,...,STUDENT-680, six hundred and eighty first and second year students, each requiring 100 pages.

Apart from user SYSTEM, SCRATCH and ANALYST-1,...,ANALYST-5, the other users will have roughly equal use. The only important point to note is that students in a class will be making intense use of the machine simultaneously, and it is thus advisable to group student classes so that all user entries for students in a class lie on the same page.

The six directories available, each with 36945 pages, should be organized as shown in table. 4.

Directory index	User index	User names	Number of pages	Notes
0	0	SYSTEM	10,000	Main and default directory
	1	SCRATCH	4,000	
	2	RT	0	
	3	ANALYST-1	2,000	
		. . . . .		
		. . . . .		
	7	ANALYST-5	2,000	
	8	STUDENT-3-1	150	
		. . . . .		
		. . . . .		
	89	STUDENT-3-82	150	
	90	STUDENT-1	100	
		. . . . .		
		. . . . .		
	95	STUDENT-6	100	
Total:			36,900 Pages	
1	0	RESEARCH-1	500	
		. . . . .		
		. . . . .		
	49	RESEARCH-50	500	
	50	STUDENT-3-83	150	
		. . . . .		
		. . . . .		
	67	STUDENT-3-10	1150	
	68	STUDENT-7	100	
		. . . . .		
	159	STUDENT-98	100	
Total:			36,900 Pages	
2	0	POSTGRAD-1	250	
		. . . . .		
		. . . . .		
	89	POSTGRAD-90	250	
	90	STUDENT-236	100	
Total:			36,300 Pages	

TABLE CONTINUES...

Directory index	User index	User names	Number of pages	Notes
3	0	POSTGRAD-91	250	
		. . . . .		
	89	POSTGRAD-180	250	
	90	STUDENT-237	100	
		. . . . .		
	227	STUDENT-374	100	
		<u>Total: 36,300 Pages</u>		
4	0	STUDENT-375	100	
		. . . . .		
	169	STUDENT-544	100	
	170	STUDENT-3-102	150	
		. . . . .		
	249	STUDENT-3-181	150	
	250	POSTGTAD-181	250	
		. . . . .		
	255	POSTGRAD-186	250	
		<u>Total: 30,500 Pages</u>		
5	0	STUDENT-545	100	
		. . . . .		
	135	STUDENT-680	100	
	136	STUDENT-3182	150	
		. . . . .		
	184	STUDENT-3-320	150	
	185	POSTGRAD-187	250	
		. . . . .		
	198	POSTGRAD-200	250	
	199	BPUN-FILES	1,000	
	200	FLOPPY-USER	0	
		<u>Total: 25,450 Pages</u>		

Table 4. Organization of the Six Directories

---

### 3.2.6 DIRECTORY MAINTENANCE AND CORRECTION

---

The system supervisor is responsible for the maintenance of the directories. If errors occur, they should be corrected. Both test programs and SINTRAN III commands are used for this purpose. See section 10.3 which describes the FILE SYSTEM INVESTIGATOR and related SINTRAN III commands.

---

## 3.3 USERS

---

User SYSTEM is responsible for creating and deleting other users, and for the amount of space allocated to each on hard disks. A mass storage device contains a certain number of pages, eg., a 75 Mbyte disk pack contains 37000 pages, to be shared by the system and different users. Note that the file system will occupy some of these.

---

### 3.3.1 CREATING AND DELETING USERS

---

A new user is created and given disk space by the commands:

```
@CREATE-USER <[directory:] user name>
@GIVE-USER-SPACE <[directory:] user name>,<number of pages>
```

A user must exist in all directories where he or she is to be given space. In addition, he or she must also exist in a main directory. If the directory name is omitted in the above command, the executing user's main directory is assumed. When a user is created, he or she has no password.

The following example creates a new user named USER-ONE. The user is given a file space of 500 pages in directory PACK-TWO. PACK-TWO is not a main directory. The only main directory in the system is PACK-ONE.

```
@CREATE-USER
USER NAME: USER-ONE
@CREATE-USER
USER NAME: PACK-TWO:USER-ONE
@GIVE-USER-SPACE
USER NAME: PACK-TWO:USER-ONE
NUMBER OF PAGES: 500
```

Create the user in the  
main directory.  
Create the user in another  
directory.  
Give the user space in  
this directory.

A user can be removed from a directory. You cannot delete a user who has files in the specified directory. If you attempt this, an error message is output, and the user is not removed. The command to use is:

Assume user USER-ONE has created files in directory PACK-TWO and is moving to another installation, taking along the device PACK-TWO. He or she may then be deleted in the main directory by the command:

To use the files at another installation, PACK-TWO must be mounted and entered there. USER-ONE's files cannot be accessed before USER-ONE is created in a main directory of that installation. Alternatively, PACK-TWO can be made a main directory, or USER-ONE can be renamed to a user name defined in a main directory. The following examples show the possibilities:

@RENAME-USER PACK-TWO:USER-ONE. SYSTEM

ND-30.003.06

---

### 3.3.2 PASSWORDS

---

Users create and change their own passwords by the @CHANGE-PASSWORD command. The password of any user can be removed by user SYSTEM. Use the command below to clear a password:

@CLEAR-PASSWORD <user name>

There are various reasons why user SYSTEM would clear another user's password. For example,

- A user may have forgotten the password. User SYSTEM can clear this password for the user.
- User SYSTEM may need to enter a user's file space. He/she can do this by removing or clearing the user's password.

The number of unsuccessful attempts to give a correct user name and/or password when logging in is counted. This enter count is used to protect the system from unauthorized users. If the number of attempt to log in exceeds a defined value, the terminal will be disabled. The system supervisor uses the SINTRAN SERVICE PROGRAM to set maximum enter counts and to enable terminals after the maximum enter count is reached. See chapter 12.

---

## 4 SUBSYSTEMS

---

This chapter describes some SINTRAN III subsystems. Some of the subsystems are integrated into SINTRAN III, ie., they are part of the operating system.

---

### 4.1 THE BATCH SYSTEM

---

The batch system is described in detail in the manual SINTRAN III Timesharing/Batch Guide (ND-60.132). Two commands are restricted to user SYSTEM. They are the commands to activate and abort a batch process, ie., @BATCH and @ABORT-BATCH. The following is an example of starting a batch processor:

```
@BATCH  
BATCH NUMBER = 2
```

This command finds a passive batch processor and starts it. It then outputs the number of the batch processor as a decimal integer. The batch number may be used in future commands to identify the batch processor. If there are no passive batch processors in the system, the message NO BATCH AVAILABLE is printed.

After the batch processor is started, it immediately enters the waiting state, as the batch queue initially will be empty. It will automatically be restarted when a batch job is entered by the @APPEND-BATCH command.

This command below will abort a batch processor and release all resources reserved by the batch processor. Any job currently running will be aborted immediately, and the batch queue will be cleared.

```
@ABORT-BATCH <batch number>
```

The batch processor will then be in the passive state.

---

## 4.2 DATA COMMUNICATION AND NETWORK SYSTEMS

---

ND Computer Systems can be interconnected in network systems. The two network systems are ND-NET and COSMOS. ND computers can communicate with other manufacturers' computer systems through the remote job entry (RJE) subsystems.

---

### 4.2.1 COMMANDS TO INITIATE AND TERMINATE ND-NET

---

The system supervisor is responsible for starting and stopping the communication on a line. A communication line can be divided logically into a maximum of 16 communication channels. Each channel will be assigned a logical device number on either side. The command to initiate communication on a line is:

@START-COMMUNICATION [<line number>]

The <line number> may be omitted if there is only one possible remote connection. The line number will then be 1. A configuration table containing the correspondence between channel numbers and logical device numbers is sent to the remote computer.

If the remote computer answers with its own configuration table within 12 seconds, the message COMMUNICATION ESTABLISHED is printed. If no answer is received, NO REPLY is printed. The system will continue to retransmit the configuration table until @STOP-COMMUNICATION is given. Thus, there may be an arbitrary time delay between the @START-COMMUNICATION commands on the two computers are given.

Communication should be stopped from both computers connected to a line. The command to use is:

@STOP-COMMUNICATION [<line number>]

The <line number> may be omitted if there is only one possible remote connection. The commands to start and stop communication can only be given by user SYSTEM or user RT.

You can load a stand-alone program into a remote computer. The program should be in bpun format. The command to use is:

@REMOTE-LOAD <load file>,(<bootstrap address>),( <line number>)



The <load file> is the file to be loaded. The default file type is :BPUN. The <bootstrap address> is the starting address of the bootstrap in the remote computer. The bootstrap occupies approximately 400 words. The default address is 0. The <line number> is the line to the remote computer. The default value is 1 if there is only one remote computer.

The command is permitted for users RT and SYSTEM only. It is valid only when communication is stopped on the line. The bootstrap must reside on the file (SYSTEM)REMOTE-BOOTSTRAP:BPUN, in the local computer. It is usually created when the system is generated.

Before doing a remote load, the remote load must be initiated. This is done by pressing MCL and typing <octal logical device number>& on the console of the remote computer. The logical device number is that of the communication line. If automatic load is used, only MCL and LOAD are required.

In the following example, a file called REMFI:BPUN on the local computer will be loaded to the remote computer on line 1. The bootstrap will occupy the memory area 37400 - 37777 in the remote computer.

```
@REMOTE-LOAD REMFI, 37400, 1
```

Remote RT programs can access files through a communication line. You can create a password to protect your files from remote access. A file will only be opened if the password of user RT in the remote computer is equal to this password.

```
@REMOTE-PASSWORD (<line number>), (<password>)
```

The <line number> is the line to the remote computer. The default value is 1 if there is only one remote computer. The default password is no password. The command is permitted for user SYSTEM only. In the following example, files are only opened in the remote computer on line number 3 if user RT has the password DELTA.

```
@REMOTE-PASSWORD 3, DELTA
```

For further information about ND-NET and the facilities available, see the manuals SINTRAN III Communications Guide (ND-60.134) and ND-NET System Documentation (ND-60.081).

---

**4.2.2 COSMOS**

---

COSMOS is the newer network system for ND Computers. The operation and commands of the COSMOS network are described in the manual COSMOS Operator's Guide (ND-30.025).

---

**4.2.3 REMOTE JOB ENTRY**

---

A REMOTE JOB ENTRY (RJE) emulator is used to communicate with other manufacturers' computers. RJE emulators are delivered for emulating IBM 2780, IBM 3780, IBM HASP WORK STATION, CDC 200 USER, UNIVAC DCT 2000 and Honeywell Bull GERTS 115. An emulator is delivered on a floppy disk in BRF format. The emulators are run as RT programs and must be loaded by the RT LOADER. An RJE emulator started and stopped by user SYSTEM or user RT.

The procedure for each emulator is thoroughly documented in the appropriate manual, as well as the PD sheets accompanying the emulator floppy disks. The manuals describing remote job operations are shown in table 5.

MANUAL TITLE	MANUAL NUMBER
3270 Emulator User's Guide	ND-60.114
3780 User's Guide	ND-60.067
200 User Operator's Guide	ND-60.061
DCT-2000 Operator's Guide	ND-60.060
GERTS-115 Operator's Guide	ND-60.041
HASP Work Station	ND-60.069
HASP-II Work Station User's Guide	ND-60.143
Honeywell Bull Remote Job Entry Emulator User's Guide	ND-60.160
NTR User's Guide	ND-60.070
UTS-400 Emulator User's Guide	ND-60.159
7750 VIP Emulator User's Guide	ND-60.100
7750 VIP Emulator-II User's Guide	ND-60.140

Table 5. Remote Job Entry Manuals

---

### 4.3 MAIL

---

The MAIL system makes it possible for a user to send messages to any other user. User SYSTEM may send a broadcast, ie., a message sent to all users at the same time. MAIL is an integrated part of SINTRAN III.

Messages and broadcasts are normally put into a mailbox and the receiver is notified when logging in or logging out. The receiver must, however, ask for the waiting mail by entering @MAIL.

Messages and broadcasts may also be sent directly to all terminals. The message will be displayed immediately on all terminals whether they have having logged in users or not.

When MAIL is entered, a set of subcommands are available. Some of these commands are restricted to user SYSTEM. You enter MAIL as shown below. The <output file> will only be requested if you have mail in your mailbox.

@MAIL [<output file>]

Each time MAIL is entered, all waiting mail for this user will be printed on the specified <output file>. The default output file is your terminal.

An asterisk (\*) is used as the prompt in the MAIL system. The HELP command will list all available commands. You return to SINTRAN III by the EXIT command. The other available subcommands are:

<b>*INITIALIZE</b> <maximum number of messages>
---

This command must be given by user SYSTEM before the MAIL system can be used. It can also be used to reinitialize the system. The mailbox is the mass storage file (SYSTEM)MAILBOX:DATA. This command can only be given by user SYSTEM.

**\*BROADCAST**

The message is asked for by MESSAGE:. It must be terminated by CTRL L. The message will be put into the mailbox. The character \$ is converted to carriage return, ie., it starts a new line on output. Apostrophes (') are not permitted. The command can only be given by user SYSTEM.

**\*DIRECT-BROADCAST**

The message will be asked for by MESSAGE:. It must be terminated by CTRL L. The message will immediately be sent to all terminals. The character \$ and ' are handled as for BROADCAST. Only user SYSTEM is allowed to send direct broadcasts.

**\*SEND-MESSAGE <user name>**

The message is asked for by MESSAGE:. The text is terminated by CTRL L. The message will be put into the mailbox, addressed with the given <user name>. The characters \$ and ' are handled as for BROADCAST.

**\*SEND-DIRECT-MESSAGE <terminal number>**

The message is asked for by MESSAGE:. The text is terminated by CTRL L. The message will be sent immediately to the specified terminal. The characters \$ and ' are handled as for BROADCAST. The SINTRAN III command @WHO-IS-ON will list the terminal numbers of logged in users.

**\*LIST-MESSAGES (<output file>)**

All messages will be listed together with a message number. The message number is needed to delete a message. Default output file is your terminal.

**\*LIST-BROADCASTS (<output file>)**

All broadcasts will be listed together with a broadcast number. The broadcast number is used to delete broadcasts. Default output file is your terminal. The command is restricted to user SYSTEM.

**\*DELETE-MESSAGE <message number>**

The message will be deleted. The message number can be found in the output from LIST-MESSAGES. Only user SYSTEM is allowed to delete messages sent from other users.

**\*DELETE-BROADCAST <broadcast number>**

The broadcast will be deleted. The number can be found in the output from LIST-BROADCASTS. Only user SYSTEM is allowed to delete broadcasts.

**\*STOP-MAIL-SYSTEM**

The MAIL system will be unavailable. However, existing mail will not be lost. Only user SYSTEM is allowed to stop the MAIL system.

**\*RUN-MAIL-SYSTEM**

The MAIL system will be restarted, containing the same mail as before it was stopped by the command STOP-MAIL-SYSTEM. The command is restricted to user SYSTEM.

The MAIL system can only be entered by one user at a time. If MAIL commands are used in a batch file, the commands should be preceded by a @, eg., @BROADCAST.

---

#### 4.4 REENTRANT SUBSYSTEMS

---

A reentrant subsystem is a subsystem that can be used by one than one user at the same time. This will save space in main memory if the subsystem is used simultaneously. The following command will establish a subsystem as a reentrant subsystem:

```
@DUMP-REENTRANT <subsystem name>, (<start address>),
                    (<restart address>), <binary file>
```

The parameters are:

##### SUBSYSTEM NAME:

The name of the subsystem which should be dumped as reentrant.

##### START ADDRESS:

The initial start address. This can be found on the PD sheet accompanying the subsystem. See appendix J. The default address is 0.

##### RESTART ADDRESS:

The address where the program should be reentered by the @CONTINUE command. It can be found on the appropriate PD sheet. The default restart address is 0.

##### BINARY FILE:

The file must contain the subsystem in bpun format. The default file type is :BPUN.

The subsystem will be read from a binary file in bpun format. The command will call the RT LOADER, and the subsystem is then loaded to a segment. The subsystem name is inserted into the command table. This table is searched by the command @RECOVER, before it inspects the list of files for a file of type:PROG.

The following example shows how the BACKUP SYSTEM can be dumped reentrant. The subsystem is first copied from the floppy disk it is delived on.

```
@COPY-FILE
DESTINATION FILE: "(BPUN-FILES)BACKUP-SYSTEM:BPUN"
SOURCE FILE: (ND:FLOPPY-USER)BACKUP-SYSTEM:BPUN
@DUMP-REENTRANT
SUBSYSTEM NAME: BACKUP-SYSTEM
START ADDRESS: 0
RESTART ADDRESS: 0
BINARY FILE: (BPUN-FILES)BACKUP-SYSTEM
```

User BPUN-FILES is a user created to hold the bpun files. User SYSTEM is created as a friend of user BPUN-FILES with all kinds of access. The BACKUP SYSTEM will be loaded to a segment. Any user may now start it by giving:

@RECOVER BACKUP-SYSTEM or just @BACKUP-SYSTEM

The related command @DUMP-PROGRAM-REENTRANT can be used to dump prog files instead of bpun files reentrant.

The following command will delete a reentrant subsystem name from the command table:

@DELETE-REENTRANT <subsystem name>

Remember to delete old versions of subsystems before you load new ones. In the following example, the subsystem MAC will no longer be available as a reentrant subsystem.

@DELETE-REENTRANT  
SUBSYSTEM NAME: MAC

The commands @DUMP-PROGRAM-REENTRANT, @DUMP-REENTRANT and @DELETE-REENTRANT are only available to user SYSTEM. The command @LIST-REENTRANT is available to all users. For an example of the DUMP-REENTRANT:MODE file used to start SINTRAN III, see appendix H.





---

---

## 5 ACCOUNTING

---

This subsystem provides the facility to account for usage of the computer system's resources. Options can be chosen when your SINTRAN III is generated to permit accounting of:

### ACCOUNTING

- 1) CPU time used for ND-100 users, RT programs and ND-500 users.
- 2) Connect time for users accessing an ND-100 or ND-500.
- 3) Block input/output transfers through the file system, ie., 1 Kword pages written to or read from files on disk, floppy disk or magnetic tape.
- 4) Number of pages output on printers through the spooling system.

The accounting system consists of:

- 1) Pointers, flags and tables within SINTRAN III.
- 2) SINTRAN III commands.
- 3) Files belonging to user SYSTEM.
- 4) An RT program to provide RT accounting (option).
- 5) A service program to manage the accounting files and to produce reports from the accounting data collected.

It is the system supervisor's responsibility to create project names and project passwords. The ACCOUNTING SERVICE PROGRAM is used to create RT and background projects and later output accounting results. The supervisor should notify users of their project names and project passwords.

Accounting should be initialized with the SINTRAN III command @INIT-ACCOUNTING. After initialization, accounting should be controlled by @START-ACCOUNTING and @STOP-ACCOUNTING.

---

### 5.1 FILES USED BY THE ACCOUNTING SYSTEM

---

In the following information references to RT accounting can be ignored if your SINTRAN III has not been generated with RT accounting. The following files are created and maintained by the ACCOUNTING SERVICE PROGRAM:

ACCOUNTS:DATA.	(indexed file)
PROJNAM:DATA.	(indexed file)
RTPROJ:DATA.	(indexed file)

Once these files have been created, they should be given the access specified in table 6, 7 and 8. The accounting data accumulated in the file ACCOUNTS:DATA should be periodically dumped to the file AACCOUNTS:DATA. The ACCOUNTING SERVICE PROGRAM will create this file and the file AACCOUNTS:ISAM.

The file PROJNAM:DATA contains the legal combinations of project password and project name for background projects. It should have the format and access shown in table 6.

File access for PROJNAM:DATA:

PUBLIC: NO  
FRIEND: NO  
OWNER: READ,WRITE,APPEND,DIRECTORY

The format of the file:

Block size: 16 words.  
Block 0: Word 0 (first word) should contain the index of the last block in the file. If the file is empty, use 0.  
Block n: Words 0-7 should contain the project password, and words 8-15 the project name.

Project password and project name are both ASCII character strings with unfilled character positions containing spaces, ie., ASCII 40<sub>8</sub>. They are not terminated by apostrophes.

Table 6. The file PROJNAM:DATA

Each user RT program can be associated with a project name. The pairs of RT program names and project names are stored in the file RTPROJ:DATA. It should have the format and access shown in table 7.

File access for RTPROJ:DATA:

PUBLIC: NO  
FRIEND: READ  
OWNER: READ,WRITE,APPEND,DIRECTORY

The format of the file is:

Block size: 12 words  
Block 0: Word 0 (first word) contains the index  
of last block in the file.  
Block n: Words 0-3 RT program name in UPPER  
case letters, no paritybit set.  
Words 4-11 project name.

RT program name and project name are both ASCII character strings with unfilled character positions containing spaces, ie., ASCII 40<sub>8</sub>. They are not terminated by apostrophes.

Table 7. The file RTPROJ:DATA

The file ACCOUNTS:DATA contains accounting records written by @LOGOUT, the ACCRT RT program, the spooling programs, and the ND-500 monitor. It has the access and format shown in table 8.

## File access for ACCOUNTS:DATA:

PUBLIC: NO

FRIEND: READ,WRITE,APPEND

OWNER: READ,WRITE,APPEND,DIRECTORY

## The format of the file is:

Block size: 30 words (decimal)

Block 0: Word 0 contains the number of records in the file. Word 1 contains the number of accounts after which a warning is given each time an additional record is written. Word 2 contains the maximum number of accounts.

Block n: The format of each entry is:

Word	Public	RT	Spooling
0	USER NAME	RT PROGRAM	USER NAME
1	"	NAME	"
2	"	"	"
3	"	"	"
4	"	"	"
5	"	"	"
6	"	"	"
7	"	"	"
8	4 (BKD/ND-500)	1 (RT)	2 (SPOOLING)
9	LOGOUT TIME	DUMP TIME	PRINT TIME
10	"	"	"
11	TERMINAL TIME	0	0
12	"	0	0
13	TERMINAL No.	0	SPOOL DEV No.
14	TIME USED	TIME USED	No. OF PAGES
15	"	"	PRINTED.
16	PROJECT NAME	PROJECT NAME	PROJECT NAME
17	"	"	"
18	"	"	"
19	"	"	"
20	"	"	"
21	"	"	"
22	"	"	"
23	"	"	"
24	BLOCK IO	BLOCK IO	0
25	TRANSFERS	TRANSFERS	0
26	ND-500 CONNECT	0	0
27	TIME	0	0
28	ND-500 CPU	0	0
29	TIME	0	0

Table 8. The Structure and Contents of ACCOUNTS:DATA

The entries in table 8 have the following format:

BLOCK IO TRANS.	The number of pages read from or written to disk files in the file system. This is not necessarily the same as the number of pages read and written from a user's program.
DUMP TIME	The time and date when the ACCRT program dumped the RT system usage data to the file. They are given as year, month, day, hour, minute and second packed as 6,4,5,5,6 and 6 bits respectively.
LOGOUT TIME	The time and date when the user logged out.
NUMBER OF PAGES	Total number of pages printed including heading and trailer.
PRINT TIME	The time the printing of the file started.
RT PROGRAM NAME	The SINTRAN III name of the RT program.
SPOOL. DEV. NO.	The number of the device on which the printing took place.
TERMINAL NUMBER	The terminal at which the user was working.
TERMINAL TIME	This is a Fortran integer. See the ND Fortran Reference Manual (ND-60.145), appendix B, for definition of data types. The integer gives the period of time the terminal was logged in. It is given in basic time units. One such unit is normally 20 microseconds.
TIME USED	A Fortran integer giving the CPU time used in basic time units.
PROJECT NAME	An ASCII character string containing the project name. Unused character positions are filled with spaces, ie., ASCII 40 <sub>9</sub> , and there is no apostrophe at the end of the name.
USER NAME	An ASCII character string containing the user name. Unused character positions are filled with spaces and there is no apostrophe after the name.

The format of the records in the file AACCOUNTS:DATA is shown in table 9. All numbers are decimal. The basic time unit is normally 20 microseconds.

Word	Type	Explanations
1 -8	CHARACTER*16	User Name or RT Program Name
9 -16	CHARACTER*16	Project Name
17-18	DOUBLE INTEGER	Accumulated CPU Time (seconds)
19-20	DOUBLE INTEGER	Intermediate CPU Time (basic time units)
21-22	DOUBLE INTEGER	Accumulated Console Time (minutes)
23-24	DOUBLE INTEGER	Intermediate Console Time (seconds)
25-26	DOUBLE INTEGER	Accumulated Number of Page Printed
27-28	DOUBLE INTEGER	Intermediate Number of Page Printed
29-30	DOUBLE INTEGER	Accumulated Number of Block IO Transfers
31-32	DOUBLE INTEGER	Intermediate Number of Block IO Transfers
33-34	DOUBLE INTEGER	Accumulated ND-500 CPU Time (seconds)
35-36	DOUBLE INTEGER	Intermediate ND-500 CPU Time (b.t.u.)
37-38	DOUBLE INTEGER	Accumulated ND-500 Console Time (minutes)
39-40	DOUBLE INTEGER	Intermediate ND-500 Console Time (seconds)

Table 9. Format of the Records in the file AACCOUNT:DATA

---

## 5.2 ACCOUNTING COMMANDS

---

The following SINTRAN III commands affect accounting. They are restricted to user SYSTEM. The command to start accounting is:

```
@START-ACCOUNTING <background> [, <RT>,
                        <clear logged information>, <logging interval>]
                        [, <ND-500>] [, <spooling>]
```

The answers to <background>, <RT>, <ND-500> and <spooling> are either YES or NO. All public users' programs will be logged if YES is answered to <background>. The other questions will only be asked if relevant for SINTRAN III on your installation. If you answer YES to RT accounting, the questions <clear logged information> and <logging interval> will be asked.

The parameter <clear logged information> must be answered YES or NO. It refers to the RT accounting table in SINTRAN III in which CPU time used by RT programs is stored, and to that part of the IO accounting table used for accumulating file system IO for RT programs. When these tables, ie., ACCTAB and IOACCTAB, are reset, the entries for all user RT programs are set equal to zero.

The <logging interval> is the number of seconds between dumps of the RT accounting table on the file ACCOUNTS:DATA. To start the RT accounting system, the RT program ACCRT must have been loaded. The command will not initialise the accounting file, ie., ACCOUNTS:DATA.

You stop accounting by the command:

```
@STOP-ACCOUNTING <background> [, <RT>] [, <ND-500>]  
                                     [, <spooling>]
```

The answers are YES or NO. The accounting file is not affected. The command below initializes the file ACCOUNTS:DATA, creating it if it does not exist:

```
@INIT-ACCOUNTING (<desired number of accounts>),  
                  (<maximum number of accounts>, <background> [, <RT>],)  
                  <spooling>, <clearing logged information>,  
                  <logging interval>] [, <ND-500>]
```

The desired number of account records and the maximum number of account records may be specified. When the <desired number of accounts> has been reached, the message APPROACHING END OF ACCOUNTING FILE will be output on the terminal at every log out. The default values are 1000 and 1200 respectively.

The @INIT-ACCOUNTING command resets the ACCOUNTS:DATA file. It should only be used the first time the accounting is started. If is necessary to change the value of the two first parameters later, the command DUMP-AND-ACCUMULATE in the ACCOUNTING SERVICE PROGRAM should be used. Otherwise, the contents of the file ACCOUNTS:DATA will be lost.

When background accounting is selected, SINTRAN III will ask for the project password when a user tries to log in. The user input will not be echoed to the terminal. If the password is correct, the project name is printed on the terminal.

If it is incorrect, the question is repeated. After three unsuccessful attempts to give a valid project password, the user will automatically be logged out. If there are no entries in the PROJNAM:DATA file, log in will be as if background accounting were not started.

RT accounting is started by the command @START-ACCOUNTING. The CPU time used by those programs specified in the command @START-RT-ACCOUNTING is recorded.

```
@START-RT-ACCOUNTING <RT program name>
```

The parameter <RT program name> is checked against names in the file RTPROJ:DATA. If a match is found, the program is marked in the RT accounting table. RT accounting may be started for user defined RT programs only.

The RT description address is used as index in the tables ACCTAB and IOACCTAB. If the RT program is reloaded using another RT description, one needs to give @STOP-RT-ACCOUNTING for this program before reloading it. The command @START-RT-ACCOUNTING should be given for the reloaded program. The command to stop accounting is defined as:

@STOP-RT-ACCOUNTING <RT program name>

RT accounting is stopped for the specified RT program. The command @LIST-RT-ACCOUNTING displays the names of all RT programs being logged with their project names and the time used since the last dump of the RT accounting table. User RT is also allowed to use the start, stop, and list RT accounting commands.

---

### 5.3 DUMPING THE ACCOUNTING TABLES

---

The RT program ACCRT dumps accounting tables on the file ACCOUNTS:DATA. It is delivered on a floppy disk as ACCRT:BPUN and loaded onto segment 32 using the READ-BINARY command in the RT LOADER. This is shown in the following example:

```
@RT-LOADER
*READ-BINARY (ND:FLOPPY-USER)ACCRT:BPUN, 32
*YES
*END-LOAD
*EXIT
```

You start ACCRT by the commands @START-ACCOUNTING or @INIT-ACCOUNTING. The program is stopped by the command @STOP-ACCOUNTING.

---

### 5.4 THE ACCOUNTING SERVICE PROGRAM

---

This program should be loaded according to the PD sheets delivered with the floppy disks. The ACCOUNTING SERVICE PROGRAM makes use of the ISAM subsystem to store accounting records. ISAM is described in the manual ISAM Reference Manual (ND-60.108).



---

### 5.4.1 COMMAND SUMMARY

---

You enter the service program by giving @ACCOUNTING-SERVICE-PROGRAM. The EXIT command cause a return to SINTRAN III. The service program uses the prompt ACC:. The following commands are available:

DESCRIBE-ALL-COMMANDS	DUMP-AND-ACCUMULATE
LIST-RT-PROJECTS	PRINT-ACCUMULATED-ACCOUNTS
RESET-BACKGROUND-PROJECT-TABLE	ADJUST-ACCOUNTING-FILE
CREATE-BACKGROUND-PROJECT	RESET-ACCOUNTING-FILE
DELETE-BACKGROUND-PROJECT	NUMBER-OF-ACCOUNTS
LIST-BACKGROUND-PROJECTS	SET-INFORMATION-FILE
RESET-RT-PROJECT-TABLE	LOOK-ACCUMULATED
CREATE-RT-PROJECT	DELETE-ACCUMULATED-PROJECT
DELETE-RT-PROJECT	DELETE-ACCUMULATED-USER
EXIT	

HELP (<command name>) will list all commands matching the given command name string. If HELP is typed as parameter to HELP, the commands will be listed with their explanations.

A question mark (?) following an ambiguous command acts as HELP with the command name as parameter. If the command name is unique, it gives an explanation of this command. A (?) given in place of a parameter will explain this parameter.

ESCAPE can be used to abort parameter collecting in a command. If ESCAPE is answered to a question from an executing command, the command is aborted. If ESCAPE is given in other situations, you will return to SINTRAN III.

#### **WARNING:**

It is always possible that an error leaves the ISAM files, AACCOUNTS:DATA and AACCOUNTS:ISAM, in an inconsistent state. It is essential to take backup of these files before using the ACCOUNTING SERVICE PROGRAM commands:

DUMP-AND-ACCUMULATE
DELETE-ACCUMULATED-USER
DELETE-ACCUMULATED-PROJECT

---

**5.4.2 DETAILED DESCRIPTION OF COMMANDS**

---

The following is a detailed description of the commands and their parameters. Note that the command DUMP-AND-ACCUMULATE has subcommands.

**DESCRIBE-ALL-COMMANDS (<output file>)**

Describe all accounting system commands and their parameters. The default output file is your terminal.

**RESET-BACKGROUND-PROJECT-TABLE**

Clear all existing background project names and passwords from the file PROJNAM:DATA.

**CREATE-BACKGROUND-PROJECT <project password>, <project name>**

Create a new entry in PROJNAM:DATA containing the specified project name and password. The <project password> is the password to be specified by the user when logging in. The parameter <project name> is the name of the project to be charged for using the system.

**DELETE-BACKGROUND-PROJECT <project password>**

Delete the project password and project name specified from PROJNAM:DATA. The parameter <project password> is the password to be specified by the user when logging in.

**LIST-BACKGROUND-PROJECTS (<output file>)**

List the project passwords and their corresponding project names from entries on the PROJNAM:DATA file. The default <output file> is your terminal.

**RESET-RT-PROJECT-TABLE**

Clear all RT program names and associated project names from RTPROJ:DATA.

**CREATE-RT-PROJECT <rt program name>, <rt project name>**

Create an entry in RTPROJ:DATA giving the project name to be charged for system resources used by the specified RT program. The parameter <rt program name> must be a user defined RT program. The <project name> is the name of the project to be charged for system usage.

**DELETE-RT-PROJECT <rt program name>**

Delete an entry defining the project name for charging system usage by a specified RT program from PROJNAM:DATA. The <rt program name> must be the name of a user defined RT program.

**LIST-RT-PROJECT (<output file>)**

List all the RT projects defined in PROJNAM:DATA. The default <output file> is your terminal.

**DELETE-ACCUMULATED-USER <user name>**

Remove all details of this user from the accumulated accounts file.

**DUMP-AND-ACCUMULATE (RESET-ACCOUNTING-FILE: Subcommand)**  
**(DO YOU WANT A LOG: Subcommand)**  
**(DO YOU WANT ACCOUNTS: Subcommand)**

Process all the data written by SINTRAN III to ACCOUNTS:DATA. Produce a log and reset the file, if required. Transfer totals to AACCOUNTS:DATA and print a list of these totals. The command has subcommands which define the subsequent parameters.

S  
U  
B  
C  
O  
M  
M  
A  
N  
D

**YES <number of records before warning>,  
 <maximum number of records>**

After the specified <number of records before warning> have been written to the accounting file, a warning message is given every time a user logs out. The <maximum number of records> is the maximum number of accounting records that can be written to the accounting file.

**NO**

1

This subcommand has no parameters.

S  
U  
B  
C  
O  
M  
M

**YES <output file name for log>, <lines per page>**

The <output file name for log> is the mass storage or peripheral file name where information should be listed with the specified number of <lines per page>.

**NO**

2

This subcommand has no parameters.

S  
U  
B  
C  
O  
M  
M  
A  
N  
D

**YES <accounts by project name or user name or both>,  
 <do you want subtotalling>,<output file name for accounts>,  
 <lines per page>.**

These accounts can be sorted according to project name with one entry for every project name, or by user name with one entry for every user name, or both can be printed. The parameter <do you want subtotalling> is YES or NO. For every user name listed, subtotals are given for every project used by this user if YES is specified. If the output is sorted on projects, subtotals are given for every user. The <output file name for accounts> is a mass storage or peripheral file name where output is listed with the specified number of lines per page.

**NO**

3

This subcommand has no parameters.

**PRINT-ACCUMULATED-ACCOUNTS** <do you want accounts by user name>,  
<do you want accounts by project name>, <do you want subtotaling>,  
<output file name for accounts>, <lines per page>

Print accounts of total system usage for all user names and/or all project names. The totals are for the period of time since the accounting files were created on this system, ie., usually the first time the ACCOUNTING SERVICE PROGRAM was used. If you answer YES to <do you want accounts by user name>, the accounts will be output with one entry for each user.

If you answer YES to <do you want accounts by project name> the accounts are sorted according to project name with one entry for every project name. You may specify YES as answer to <do you want subtotaling> to have every user name listed with subtotals for each project. Each project will be listed with subtotals for each user. The last two parameters specify the output file and the number of lines on each page of listing.

**ADJUST-ACCOUNTING-FILE** <number of records before warning>,  
<maximum number of records>

Change the values of the maximum permitted number of accounts and the number of accounts before warning. The <number of records before warning> is the number of records that can be written to the accounting file before a warning message is given every time a user logs out. The <maximum number of records> is the maximum number of accounting records that can be written to the accounting file.

**RESET-ACCOUNTING-FILE** <number of records before warning>,  
<maximum number of records>

Change the word giving the number of records in ACCOUNTS:DATA. The <number of records before warning> is the number of records which can be written to the accounting file before a warning message is output every time a user logs out. The second parameter is the maximum number of accounting records that can be written to the accounting file.

**DELETE-ACCUMULATED-PROJECT <project name>**

Remove all details of this project from the accumulated accounts file. The project name must be defined in PROJNAM:DATA or RTPROJ:DATA.

**NUMBER-OF-ACCOUNTS**

List the number of accounts currently in the SINTRAN III accounting file, ie., ACCOUNTS:DATA, the maximum number of accounts permitted, and the number of accounting records before a warning is given.

**SET-INFORMATION-FILE <file name>**

Define the output file for information and for diagnostics. Diagnostics are also written to the terminal.

**LOOK-ACCUMULATED <user name>, <project name>**

Look at a specific entry in the file AACCOUNTS:DATA, ie., accumulated accounts for a given <user name> or <project name>. The project name must be defined in PROJNAM:DATA or RTPROJ:DATA.

---

## 6 PERIPHERALS AND THE SPOOLING SYSTEM

---

Peripherals, eg., line printers and floppy disk units, are treated as files. Most printers are controlled by a part of SINTRAN III called the spooling system.

---

### 6.1 PERIPHERALS

---

User SYSTEM must make peripheral files known to the file system. This is done by the command:

```
@SET-PERIPHERAL-FILE <file name>, <logical device number>
```

This command defines the specified <file name> as a peripheral file. The <logical device number> is an octal value identifying the device. A list of logical device numbers is found in the SINTRAN III Reference Manual (ND-60.128).

The <file name> may be chosen by the operator, but should preferably be the standard names given in that manual. If the file does not yet exist, it must be enclosed in quotes ("..."). For example, you introduce the line printer and give it the correct access by:

```
@SET-PERIPHERAL-FILE  
PERIPHERAL FILE: "LINE-PRINTER"  
LOGICAL DEVICE NUMBER: 5  
@SET-FILE-ACCESS  
FILE NAME: LINE-PRINTER  
PUBLIC ACCESS: WA  
FRIEND ACCESS: RWA  
OWN ACCESS: RWAD
```

The line printer has  
logical device number 5.

W means write access.  
A means append access.  
R means read access.  
D means directory access.

All printers and floppy disk units must be set as peripheral files. The following command will specify <file name> as the name used to access terminals:

```
@SET-TERMINAL-FILE <file name>
```

Public users may refer to their terminals by the specified <file name>. If the file does not exist, it must be enclosed in quotes. For example, you set the name used to access terminals to TERMINAL and the correct access by the command:

<pre>@SET-TERMINAL-FILE FILE NAME: "TERMINAL" @SET-FILE-ACCESS FILE NAME: <u>TERMINAL</u> PUBLIC ACCESS: <u>RW</u> FRIEND ACCESS: <u>RW</u> OWN ACCESS: <u>RWD</u></pre>	<pre>TERMINAL is the standard file name for terminals.  Read and write. Read and write. Read, write, and directory.</pre>
--	---

TERMINAL may now be used as an output or input file name. In the following example, FILE-ONE:DATA will be listed on your terminal. Some files created by NOTIS WP should not be inspected this way.

```
@COPY
TO DEVICE: TERMINAL
FROM DEVICE: FILE-ONE:DATA
END OF FILE
@
```

If a file is set as peripheral, peripheral is said to be an attribute of the file. Other file attributes are terminal, spooling, temporary, contiguous, allocated, and indexed.

---

### 6.1.1 FILE ACCESS FOR PERIPHERAL FILES

---

Peripherals are either input devices, output devices, or both. This section shows the file accesses that should be set. File access for input devices, eg., card readers, should be as shown in table 10.

PUBLIC:	READ
FRIEND:	READ
OWN:	READ, DIRECTORY

Table 10. File Access for Input Devices



File access for output devices, eg., printers, should be set as shown in table 11. This does not apply to printers that are controlled by the spooling system.

PUBLIC:	WRITE, APPEND
FRIEND:	WRITE, APPEND
OWN:	WRITE, APPEND, DIRECTORY

Table 11. File Access for Output Devices

File access for devices which handle both input and output, eg., magnetic tape, ND NET channels, and floppy disk units, should be set as shown in table 12.

PUBLIC:	READ, WRITE, APPEND
FRIEND:	READ, WRITE, APPEND
OWN:	READ, WRITE, APPEND, DIRECTORY

Table 12. File Access for Input and Output Devices

File access for peripherals with spooling, eg., line printers, should be set as shown in fig. 13.

PUBLIC:	WRITE, APPEND
FRIEND:	READ, WRITE, APPEND
OWN:	READ, WRITE, APPEND, DIRECTORY

Table 13. File Access for Peripherals with Spooling

---

### 6.1.2 CLEARING DEVICES

---

The following command clears a device buffer. The device is identified by its octal logical device number.

@CLEAR-DEVICE <logical device number>

This command may be used to stop the line printer if an attempt is made to print non alphanumeric information, or it may be used to stop a search for a non existent end-of-file mark on a magnetic tape to prevent the tape from winding off.

Note that @CLEAR-DEVICE on a magnetic tape operates on the magnetic tape controller. That is, if several magnetic tape stations are connected to the same controller, they will all be cleared by one @CLEAR-DEVICE command, except for STC (6250 bpi) - magnetic tape, where a @DEVICE-FUNCTION to clear one particular unit is available.

---

## 6.2 THE SPOOLING SYSTEM

---

The spooling system allows several users to address a printer "simultaneously". A spooling queue contains information about which files are to be printed, who entered them, how many copies will be made, and possibly messages to be output on the console prior to printing each file.

---

### 6.2.1 PREPARING SPOOLING FILES

---

If you want a spooling system, SINTRAN III must be generated with the required spooling programs. These are system included RT programs called SPRT1, SPRT2, etc. Before you start a spooling process it is necessary to:

- Create a peripheral file with the same logical device number number as on the SINTRAN III order form.
  - Set the correct access.
  - Create as many versions as you want places in the spooling queue. This is usually 10 versions known as spooling files.

The following shows an example for a LINE-PRINTER. Ten versions of the file are created. The first version is the peripheral file with logical device number 5. The remaining files are spooling files.

```
@SET-PERIPHERAL-FILE "LINE-PRINTER",5B  
@SET-FILE-ACCESS LINE-PRINTER,WA,RWA,RWAD  
@CREATE-NEW-VERSION LINE-PRINTER::10,0
```

Each spooling device has to be started by the command @START-SPOOLING. The spooling program then reserves the peripheral file. Once spooling is started, output to that peripheral goes via the spooling program.

The spooling program can be set to stop before printing each file by the command @DEFINE-SPOOLING-CONDITIONS. This allows the operator to receive messages about the job and change to a new paper type or to do other things.

Files to be printed are copied to spooling files. All the spooling files, even if they belong to different spooling programs, take their space from a "pool" of spooling pages under user SYSTEM. The default size of this "pool" is 500 pages. User SYSTEM can change the size by the commands @GIVE-SPOOLING-PAGES and @TAKE-SPOOLING-PAGES. Note that user SYSTEM must have at least as many free pages as the spooling system has spooling pages.

---

### 6.2.2 SPOOLING SYSTEM OVERVIEW AND COMMAND SUMMARY

---

This section contains a summary of the commands related to spooling. They will be described in detail in the next section. The following commands control the spooling program:

@DEFINE-SPOOLING-CONDITIONS	Commands used to
@GIVE-SPOOLING-PAGES	control the
@START-SPOOLING	spooling program
@STOP-SPOOLING	
@TAKE-SPOOLING-PAGES	

There are two ways to write to a peripheral. One uses the spooling files. The other appends the user's own files to the spooling queue. From an editor, you can specify the peripheral file as the output device. In a command, the peripheral file may be the output file. The file you want to print is then copied to a free spooling file, and a queue entry with the spooling file name is generated.

If the text you want to print has the desired format stored on a file, you can append a queue entry with the file name to the spooling queue. Text documents produced by NOTIS-TF should not be appended to a spooling queue.

The peripheral can have a spooling form identification defined by the command @SET-SPOOLING-FORM. The spooling form identification can be compared with the user text connected to the spooling queue entry. If a match is found, the file is printed.

This could be used to sort out which queue entries are printed, eg., those using the same paper format. If a spooling queue is shared by several peripherals, the spooling form can be used to print to the desired peripheral. This is shown in fig. 7.

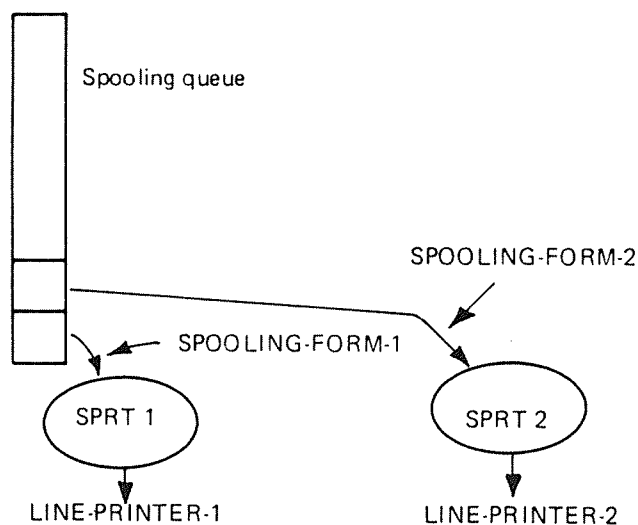


Figure 7. The Use of Spooling Forms

The following commands are used to control the spooling queue:

@APPEND-SPOOLING-FILE	
@DEFINE-SPOOLING-FILE-MESSAGE	Commands used to
@DELETE-SPOOLING-FILE	control the
@MOVE-SPOOLING-QUEUE-ENTRY	spooling queue.
@REMOVE-FROM-SPOOLING-QUEUE	
@SET-NUMBER-OF-PRINT-COPIES	
@SET-SPOOLING-FORM	

When a spooling file starts to get printed, it is removed from the spooling queue. There are a number of commands to control the file currently being printed. Such files are either spooling files or user files as shown in fig. 8.

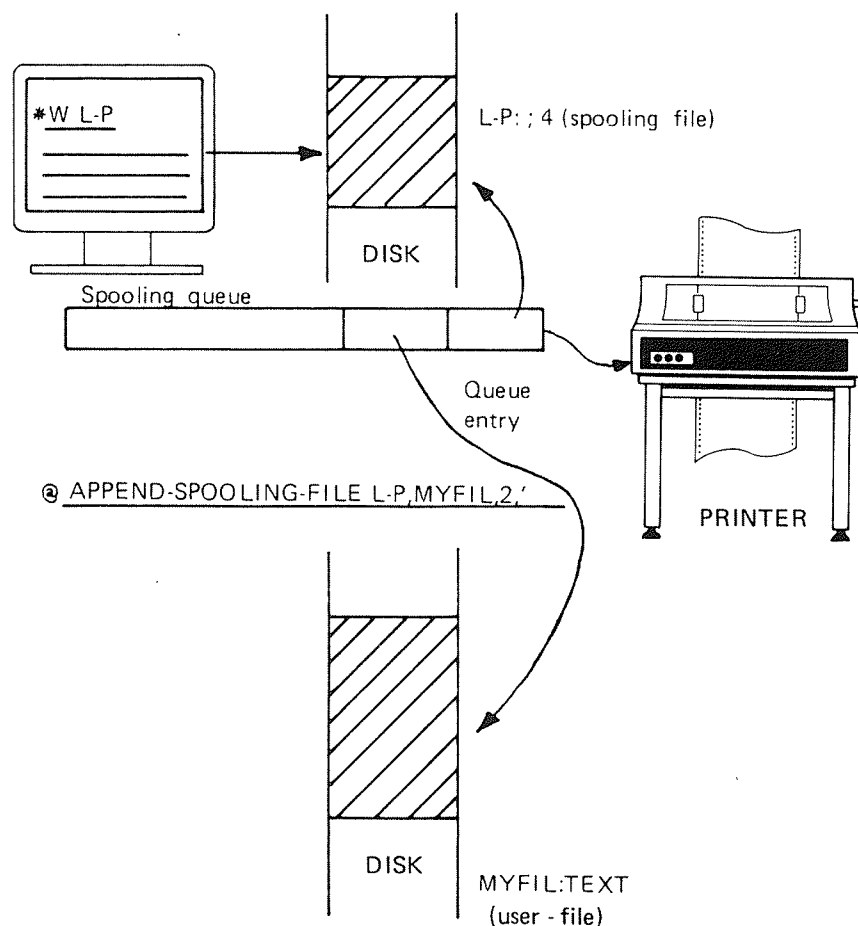


Figure 8. User Files and Spooling Files in the Spooling Queue

You can stop and start the printing, you can jump forwards or backwards in the file, or you can restart the printing. You may also terminate it even if it is not finished. The commands to control the file currently being printed are:

@ABORT-PRINT	Commands to control the file currently being printed.
@BACKSPACE-PRINT	
@FORWARD-SPACE-PRINT	
@RESTART-PRINT	
@START-PRINT	
@STOP-PRINT	

There are commands to provide information about the spooling queue, the spooling form and the number of free spooling pages. These commands are:

@LIST-SPOOLING-FORM	Commands to list the status of the system.
@LIST-SPOOLING-QUEUE	
@SPOOLING-PAGES-LEFT	

A spooling header containing various information is printed before each file. This spooling header can be removed by the SINTRAN SERVICE PROGRAM command REMOVE-SPOOLING-HEADER. The following spooling commands are available in the SINTRAN SERVICE PROGRAM. See chapter 12.

INSERT-SPOOLING-HEADER	Spooling commands in the SINTRAN SERVICE PROGRAM.
REMOVE-SPOOLING-HEADER	
SET-SPOOLING-DEVICE-NUMBER	

---

### 6.2.3 SPOOLING COMMANDS FOR USER SYSTEM

---

This section describes the spooling commands restricted to user SYSTEM. The SINTRAN SERVICE PROGRAM commands related to spooling are described in chapter 12.

---

### 6.2.3.1 START AND STOP SPOOLING PROGRAMS

---

The following command starts the spooling program for a specified printer:

`@START-SPOOLING <peripheral file name>`

The printer will be reserved for the spooling program, and the spooling program will print every file linked to the spooling queue for that device until the `@STOP-SPOOLING` command is used.

If more than one version of the file is a peripheral file, the spooling programs for all peripheral versions of the file are started. One specific peripheral device may be selected by including a version number in the file name.

An error message will appear if the specified file name is not the name of a peripheral, or if no spooling program exists for it. The command `@RTENTER` must have been given before spooling can be started.

The following command stops the spooling program for a specified peripheral:

`@STOP-SPOOLING <peripheral file name>`

Any file currently being printed will be completed before the spooling program is stopped. The spooling queue is unaffected by the command, and files may still be appended to the queue. The spooling program will resume printing the files in its queue when the `@START-SPOOLING` command is used.

---

### 6.2.3.2 GIVE AND TAKE SPOOLING PAGES

---

Initially, 500 pages are available to the spooling system. The number may be increased with the command:

`@GIVE-SPOOLING-PAGES <number of pages>`

Note that the number of spooling pages actually available never exceeds the number of free pages under user SYSTEM. If a file exceed the number of pages available for the spooling files, the terminal will be left hanging during print out of those pages which exceed the spooling pages. If the spooling space is filled up, the message NO

MORE SPOOLING PAGES AVAILABLE will appear on the terminal of those who try to put new files in the spooling queue.

You may take pages from the spooling system by the command:

```
@TAKE-SPOOLING-PAGES <number of pages>
```

The pages to be taken must be unused.

---

#### 6.2.3.3 DEFINE SPOOLING CONDITIONS

---

The following command defines the conditions for spooling on the specified peripheral:

```
@DEFINE-SPOOLING-CONDITIONS <peripheral file name>,  
    (<printing name of spooling files?>),  
    (<stop and wait for start-print before printing file?>),  
    (<number of lines per page>)
```

If the second parameter is YES, the file names of all files printed on the peripheral will be written on the console. The default answer is NO.

The third parameter may only be YES if only if the preceeding parameter was YES. This will stop the printer before each new file is output. The @START-PRINT command must then be used to start printing. The fourth parameter is the number of lines to be printed on each page. Default is 68.

---

#### 6.2.3.4 SPOOLING FORM

---

The following command defines a spooling form:

```
@SET-SPOOLING-FORM <peripheral file name>,  
    (<spooling form identification>)
```

The text specified as <spooling form identification> will be compared to the <text> specified in the command @APPEND-SPOOLING-FILE. If they match, the file is printed.



This is used to sort out all spooling entries with the same user text, eg., INVOICE. It is also used to reset the spooling form to an empty string, which is default. This will cause all files without a user text to be printed. The <spooling form identification> is terminated by an apostrophe (').

---

#### **6.2.4 SPOOLING COMMANDS FOR PUBLIC USERS**

---

The spooling commands described in this section are available to all users.

---

##### **6.2.4.1 LIST SPOOLING QUEUE**

---

The following command gives information on the entries in a spooling queue:

@LIST-SPOOLING-QUEUE <peripheral file name>, (<output file>)

The information includes the file's name, the number of bytes it contains, the name of the user who appended it, the number of copies to be printed and, if applicable, a user text to be written to the console when printing starts. Similar information is supplied for the file which is currently being printed as well as the number of bytes left to print.

---

##### **6.2.4.2 STOP, START, RESTART AND ABORT PRINT**

---

The commands in this section are only available to the user who appended the file to the printer and user SYSTEM. The following command stops printing on a specified printer:

@STOP-PRINT <peripheral file name>

The spooling system will await further commands. The following command resumes printing:

@START-PRINT <peripheral file name>

The following command restarts printing from the beginning of the file currently being printed:

@RESTART-PRINT <peripheral file name>

The command has no effect if no file is being printed. The command is useful when a paper crash has occurred in a printer.

The following command can be used to skip printing the remaining part of a file:

```
@ABORT-PRINT <peripheral file name>
```

The spooling program continues with the next file in the spooling queue. The command has no effect if no file is being printed.

---

#### 6.2.4.3 BACKWARD AND FORWARD SPACE PRINT

---

You may skip printing of some pages or lines of a file. The command to use is:

```
@FORWARD-SPACE-PRINT <peripheral file name>,  
                      (<number of pages>), (<number of lines>)
```

This command may only be used when the printing has been stopped. When printing resumes the given number of pages will be skipped. The number of lines per page depends on the printer. It may be changed by the command @DEFINE-SPOOLING-CONDITIONS. Only user SYSTEM and the user who appended the file may use this command. The two last parameters have 0 as default value.

If a paper crash occurs, it will be necessary to reprint some lines or pages. You can use the command:

```
@BACKSPACE-PRINT <peripheral file name>, (<number of pages>),  
                                           (<number of lines>)
```

Only user SYSTEM and the user who appended the file may use this command. The two last parameters have 0 as default value.

---

#### 6.2.4.4 MOVE SPOOLING QUEUE ENTRY

---

The spooling queue can be reorganized. Files in the queue of a particular printer can be moved by the command:

```
@MOVE-SPOOLING-QUEUE-ENTRY <peripheral file name>,<file name>,  
                           <insert or append?>, <before/after file name>
```

The file will be moved in front of or after the file specified as last parameter. If the third parameter is insert, the file will be placed in front. If the parameter is append, the file will be placed after. All users may move the entries they have appended further back in the spooling queue. User SYSTEM may move entries forward.

---

#### **6.2.4.5 REMOVE AND DELETE SPOOLING FILES**

---

The following command removes a file from the spooling queue:

```
@REMOVE-FROM-SPOOLING-QUEUE <peripheral file name>,  
                                <file name>
```

The contents of the file will be retained. Only user SYSTEM and the user who appended the file to the queue may use this command. Spooling files can be deleted by the command:

```
@DELETE-SPOOLING-FILE <peripheral file name>, <file name>
```

The file specified in the second parameter is removed from the spooling queue for the specified peripheral. If the file is a spooling file, its pages are released and returned to the pool of free spooling pages. Only user SYSTEM and the user who appended the file to the queue can delete the file from the queue.

---

#### **6.2.4.6 APPEND SPOOLING FILE**

---

The command to append one of your files to the spooling queue is:

```
@APPEND-SPOOLING-FILE <peripheral file name>,  
    <file name>, (<number of copies>), <text>  
    [, <printing message independent of spooling conditions?>]
```

The file specified in the second parameter is appended to the spooling queue of a printer. The specified number of copies of the file will be printed. Default value is one copy.

The fourth parameter defines a user text to be written to the console before the file is printed. The text should be terminated with an apostrophe (').

The last parameter is only requested if <text> is specified. It may take the values YES or NO. If YES, the text will be written, and the printer will stop before the file is output regardless of what was specified in the command @DEFINE-SPOOLING-CONDITIONS. If NO, the text will be suppressed if the command @DEFINE-SPOOLING-CONDITIONS has been given with NO as the second and third parameter. The default value for this optional parameter is NO.

---

#### 6.2.4.7 SET NUMBER OF PRINT COPIES

---

The following command can be used to determine the number of times a file in the spooling queue will be copied:

```
@SET-NUMBER-OF-PRINT-COPIES <peripheral file name>,  
                             <file name>, (<number of copies>)
```

Only user SYSTEM and the user who appended the file can do this. If several entries in the spooling queue match the <file name>, only the first will be affected.

---

#### 6.2.4.8 LIST SPOOLING FORM

---

The following command lists the current spooling form of a printer:

```
@LIST-SPOOLING-FORM <peripheral file name>
```

Spooling forms are defined by the command @SET-SPOOLING-FORM.

---

#### 6.2.4.9 SPOOLING PAGES LEFT

---

The following command lists the number of free pages that can be used by the spooling files:

```
@SPOOLING-PAGES-LEFT
```

Note that the number given is an upper limit. There may be less pages available if user SYSTEM has less free pages.

---

## 7 STOPPING, STARTING AND LOADING SINTRAN III

---

This chapter describes the different loading procedures for an ND-100 system. The additional procedures for an ND-500 system are explained at the end of the chapter.

Each SINTRAN III system is tailored for a particular installation. This system generation is normally done at ND, and the generated SINTRAN III system is delivered on five floppy disks. A person from ND will usually load the system the first time. It is called the initial loading.

As a system supervisor you will sometimes need to stop SINTRAN III, for example, to run a test program, take backup, or modify the system. When SINTRAN III is stopped, you can use MOPC to communicate with the computer from the console.

There are two ways of restarting the system:

- 1) A warm start will restart the system and initialize some minor parts of the system information, eg., which users that are logged in. Simple error situations may be handled by performing a warm start. This way of starting up SINTRAN III is usually done by first pressing the MCL and then the LOAD button on the computer's panel.
- 2) A cold start initializes the complete system before restarting SINTRAN III. You will need to load the stand-alone program MACM from a floppy disk to perform a cold start.

The warm or cold start initialization can be performed on a running computer by the commands @RESTART-SYSTEM or @COLD-START. Only user SYSTEM is allowed to use these commands.

New terminals, disk units, or printers with spooling may be connected to the computer. In that case, a new version of SINTRAN III must be generated. The procedure to load a new version of SINTRAN III is different from initial loading.

---

**7.1 STOPPING SINTRAN III**

---

Before you stop the system, you should make sure nobody is logged in and make sure that all active batch processes, RT program and printers are stopped first.

The stop procedure is normally:

```

@SET-UNAVAILABLE
TEXT: $$ MAINTENANCE$$
@MAIL
*DIRECT-BROADCAST
MESSAGE: $LOG OUT, THE SYSTEM
WILL BE STOPPED IN 5 MINUTES.
*EXIT
@WHO-IS-ON

==> 56 SYSTEM

@LIST-BATCH-PROCESS
      1 IDLE, NO USER LOGGED ON
@LIST-SPOOLING-QUEUE
PERIPHERAL FILE: LINE-PRINTER
QUEUE IS EMPTY
@STOP-SYSTEM

```

Make it impossible to log in except from the console. Inform logged-in users that the system will be stopped.

Check that no users are logged in.

Check the batch processors.

Check the printers.

Stop the system.

The command @STOP-SYSTEM cannot be abbreviated. The STOP button on the computer has the same effect. The character # is output on the console to show that MOPC is available. You may restart the system by entering an exclamation mark (!). The command @SET-AVAILABLE will allow users to log in from any terminal again.

**IMPORTANT!**

The system should normally be turned off for hardware maintenance only. It should be running at night and during week ends.

---

### 7.1.1 STOPPING THE ACTIVITY IN THE SYSTEM

---

If there are active users on the system, you ought to stop them before you stop the system. Do the following:

- Log out all users including TADs (terminal access devices) and batch processors.
- Stop XMSG if installed.
- Check that spooling devices are not printing.
- Stop user RT programs and RJE programs.

The command to log out users by force is:

@STOP-TERMINAL <terminal number>

The message ABORTED BY SYSTEM will be printed on that terminal. Only user SYSTEM is allowed to stop terminals in this way. The command @WHO-IS-ON will list all logged in users and their terminal numbers.

The command to stop a batch processor is

@ABORT-BATCH <batch number>

This command should be used for batch processors which are not passive. The commands @LIST-BATCH-PROCESS and @LIST-BATCH-QUEUE will display information about the processors. If user DATA-DICTIONARY is active, check that no one is using the ACCESS subsystem before you abort the batch.

The command to stop RT programs is:

@ABORT <rt program name>

This command should be used for user RT programs which are not passive or in RT WAIT. The command @LIST-RT-PROGRAMS will list all RT programs and their state. System-included RT programs are listed first, often ending with TIMRT.

The command to check if the printers are active is:

@LIST-SPOOLING-QUEUE <peripheral file>

The file currently being printed will be lost in a warm start. A cold start will destroy the spooling files in addition. The XMSG system is stopped by the SINTRAN SERVICE PROGRAM command STOP-XMSG.

---

### 7.1.2 MOPC

MOPC starts automatically when the system is stopped. It is a microprogram in read only memory that is used for operational control of the ND-100. You may also use MOPC while the computer is running. Just give the command @OPCOM or press the OPCOM button on the computer's panel.

All commands to MOPC must be given from the console. Below are some examples:

#!	Restarts the system.
# <u>2345!</u>	Starts the system from the octal address 2345.
# <u>1560&amp;</u>	Loads a stand-alone program from physical device number 1560.
# <u>000100&lt;000200</u>	Dumps the main memory locations 100:200 <sub>8</sub> .
# <u>0&lt;15RD</u>	Dumps the contents of the register blocks on all interrupt levels.
# <u>IRD</u>	Dumps the internal registers.
# <u>14#</u>	Performs a memory test for segment 14. A # is output if no errors are found.

Some of these commands will not be accepted if the computer is running. MOPC can also be used to modify registers, virtual memory, and physical memory. See the ND-100 Reference Manual (ND-06.014) for a detailed description of the available facilities.



---

## 7.2 WARM START

---

This section describes how a warm start can be performed on a stopped and on a running system. An overview of the involved system areas in physical memory and on the disk is given.

---

### 7.2.1 SAVE AREA, IMAGE AREA, AND SINTRAN III RESIDENT

---

SINTRAN III RESIDENT is the part of the operating system which is always kept in physical memory. Its pages are never swapped out. The most important system variables are kept in this area. Less important information is kept on the system-included segments and on ordinary mass storage files.

A copy of the initial version of SINTRAN III RESIDENT is stored on the disk as part of the IMAGE AREA, and another copy is part of the SAVE AREA. These areas play an important part in warm and colds starts.

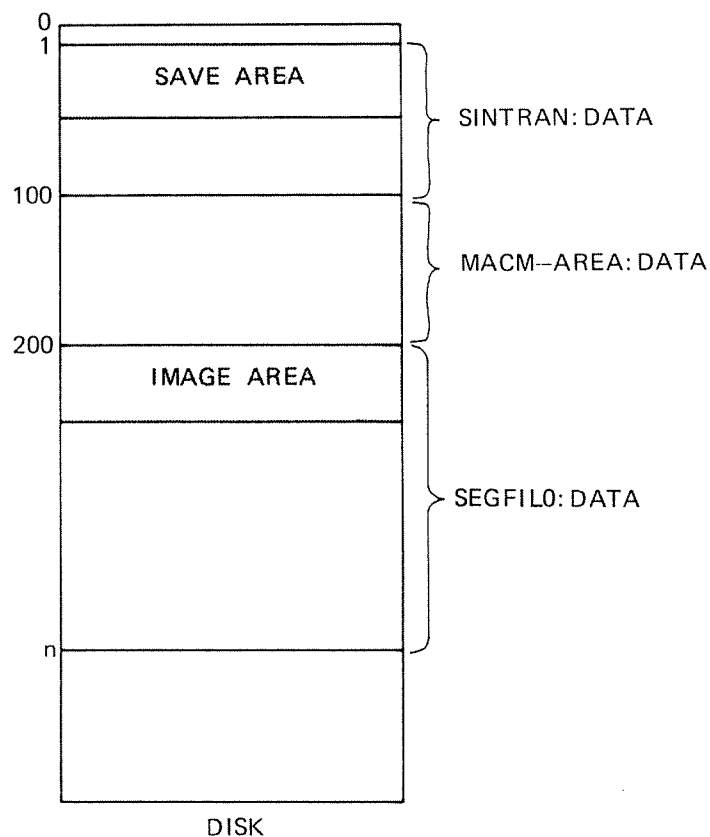


Figure 9. SAVE AREA, IMAGE AREA, and SINTRAN III RESIDENT

The SAVE AREA is on the allocated file SINTRAN:DATA. The IMAGE AREA is on a system-included segment on SEGFIL0:DATA. Both are system files placed on the first main directory. See fig. 10. The layout in physical memory and on the main directory is shown in appendix D and E.

---

### 7.2.2 THE WARM START PROCEDURE

---

A warm start is usually performed by using the buttons on the computer's panel. Turn the key to the ON position to make the buttons operational. The procedure is then:

WARM START

- 1) Press the MCL button.
- 2) Wait until the character # appears on the console, and press the LOAD button.
- 3) Wait for a SYSTEM IS AVAILABLE message. It should appear on all terminals after a couple of minutes.

The disk copy of SINTRAN III RESIDENT and the disk copy of the PAGING OFF AREA (POF) on the IMAGE AREA are fetched into physical memory. Figure 10 shows a warm start. The setting of the automatic load descriptor (ALD) must be correct. It describes where a program should be loaded from when the LOAD button is pressed. See appendix K.

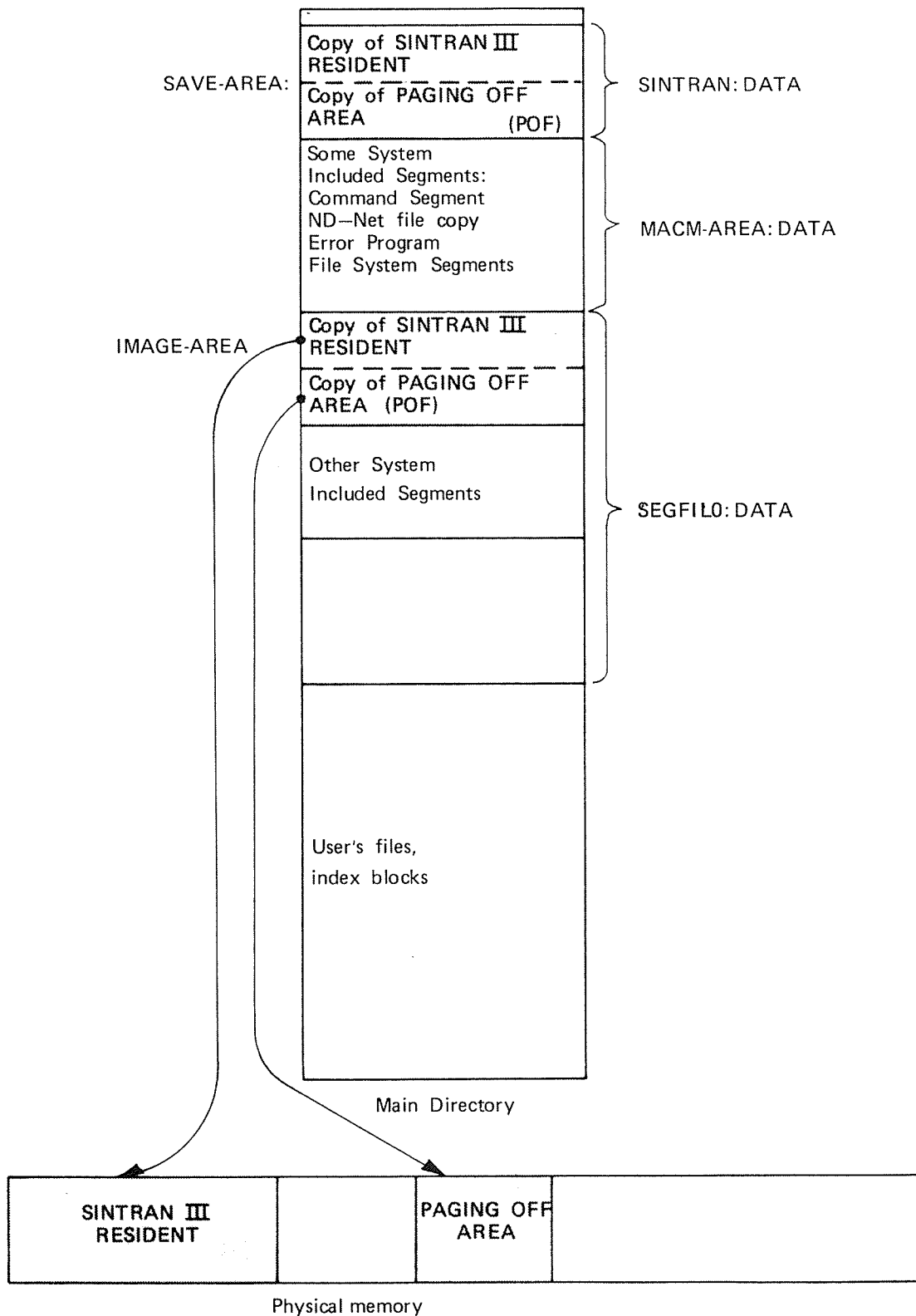


Figure 10. A Warm Start Overview

A warm start should append the batch file LOAD-MODE:SYMB to a batch processor. This depends on some initial commands that will be explained in section 7.3.1. The batch file normally performs the following tasks:

- Enters any directories not entered by the initial commands.
- Sets these directories default.
- Starts up any spooling processes.
- Starts communication.
- Starts XMSG if installed.
- Starts up any other batch processes.
- Starts up accounting.
- Sets the system available.

LOAD-MODE:SYMB

Appendix H shows an example of the file LOAD-MODE:SYMB. The file is tailored for one particular installation. If SINTRAN III is running, the same effect will result from executing the command:

@RESTART-SYSTEM

RT programs and segments loaded by the RT LOADER will not be changed. All files will be closed. The command will enter no directories. However, the main directory will be entered if this is specified as an initial command. See section 7.3.1.

---

### 7.3 COLD START

---

A cold start will often be done after a patch on the system files SINTRAN:DATA or MACM-AREA:DATA. In these cases you cannot load from the IMAGE AREA. See also appendix E and chapter 11. A cold start may also be performed if a warm start does not function. The cold start procedure for stopped systems is as follows:

#### COLD START

- 1) Press the MCL button on the computer's panel.
- 2) Place the first SINTRAN III floppy disk in unit 0 and type 1560& on the console. MACM is loaded from the floppy disk.
- 3) Type the command )HENT and give carriage return when MACM has been loaded. Use UPPER case letters. Line feed is given by the system.
- 4) Type 22!.
- 5) When SINTRAN III has started, press ESCAPE and give only carriage return when user name and password are asked for.
- 6) Update SINTRAN III's software clock if necessary.
- 7) Enter a main directory.
- 8) Log out.
- 9) Log in as user SYSTEM.
- 10) Run the batch file HENT-MODE:SYMB. This file contains a number of tasks, many of which are installation dependent. See the example in appendix H.

When the system is running, the cold start procedure can be executed by giving the command @COLD-START. Then you have to follow the procedure above from step 5. Make sure there is no activity in the system before you give the command.

---

### 7.3.1 INITIAL COMMANDS

---

The following command can be used to specify that an @ENTER-DIRECTORY command should be executed automatically at subsequent restarts of the system:

@INITIAL-COMMAND <command string>

The command string will be saved and executed when the system is restarted. The command @RTENTER will automatically be executed after the last initial command. Further directories can be entered by the LOAD-MODE:SYMB file.

All directories used by user RT and user SCRATCH should be entered and made default by initial commands. Failure to do this for user RT means that @RTOPEN or OPEN from an RT program will not access user RT's files by default.

The following is an example of how to have PACK-ONE entered automatically each time the system is restarted:

@INITIAL-COMMAND ENTER-DIRECTORY, PACK-ONE, DISC-10MB-1, 0, R

After the command @INITIAL-COMMAND has been given, the following command may be given one or more times:

@NEXT-INITIAL-COMMAND <command string>

The specified commands will be executed in sequence after the initial command when the system is restarted. The initial command buffer has room for 256 characters.

If a change in this command buffer is necessary, the buffer has to be filled up again starting with a new @INITIAL-COMMAND and one or more @NEXT-INITIAL-COMMANDs. A cold start clears the command buffer.

The contents of the initial command buffer can be listed by the command @LIST-INITIAL-COMMANDS. The following is an example of creating a commonly used initial command buffer:

```
@INITIAL-COMMAND ENTER-DIRECTORY PACK-ONE, DISC-66MB-1  
@NEXT-INITIAL-COMMAND BATCH  
@NEXT-INITIAL-COMMAND APPEND-BATCH, 1, LOAD-MODE:SYMB, SYS-OUT-1
```

---

### 7.3.2 INITIALIZING BACKGROUND PROCESSES

---

Each terminal and batch processor is controlled by a background process. The background processes are system-included RT programs. The following command must initialize RT programs before the terminals and batch processors can be used:

```
@INITIALIZE-BACKGROUND-PROGRAMS
```

The RT program controlling the console will always be initialized. This command must be given once after a cold start, and after the necessary segment files have been defined. See the SINTRAN SERVICE PROGRAM command DEFINE-SEGMENT-FILE.

This command will also allocate disk space for the background segments on the segment file or files. If an extended address space of 128 Kwords is needed to execute programs, this must be specified after using the @INITIALIZE-BACKGROUND-PROGRAMS command. The specification can be made through the command:

```
@CHANGE-BACKGROUND-SEGMENT-SIZE <logical device number>,  
                                <number of pages>
```

This command will set up the segment table entry for the RT program controlling the specified terminal or batch processor. The number of pages should be either 64 or 128. If you specify 128, the terminal may run two-bank programs. Note that @INITIALIZE-BACKGROUND-PROGRAMS must be executed before @CHANGE-BACKGROUND-SEGMENT-SIZE.

The command )HENT is a MACM command that will:

- Copy the SINTRAN III RESIDENT copy of the SAVE AREA to the SINTRAN III RESIDENT copy on the IMAGE AREA.
- Renew all system segments, ie., segments on SINTRAN:DATA and MACM-AREA:DATA will be copied to SEGFILO:DATA.

Note that any programs residing on segments such as user RT programs, reentrant subsystems, SIBAS, remote job entry systems, etc., will be lost when you give the )HENT command. The background segments of terminals and batch processors are initialized.

The initial command buffer is cleared, and all patches done only on the IMAGE AREA are lost. When 22! is typed, the "new" SINTRAN III RESIDENT and the PAGING OFF AREA is copied to physical memory. Fig. 11 shows an overview of a cold start.



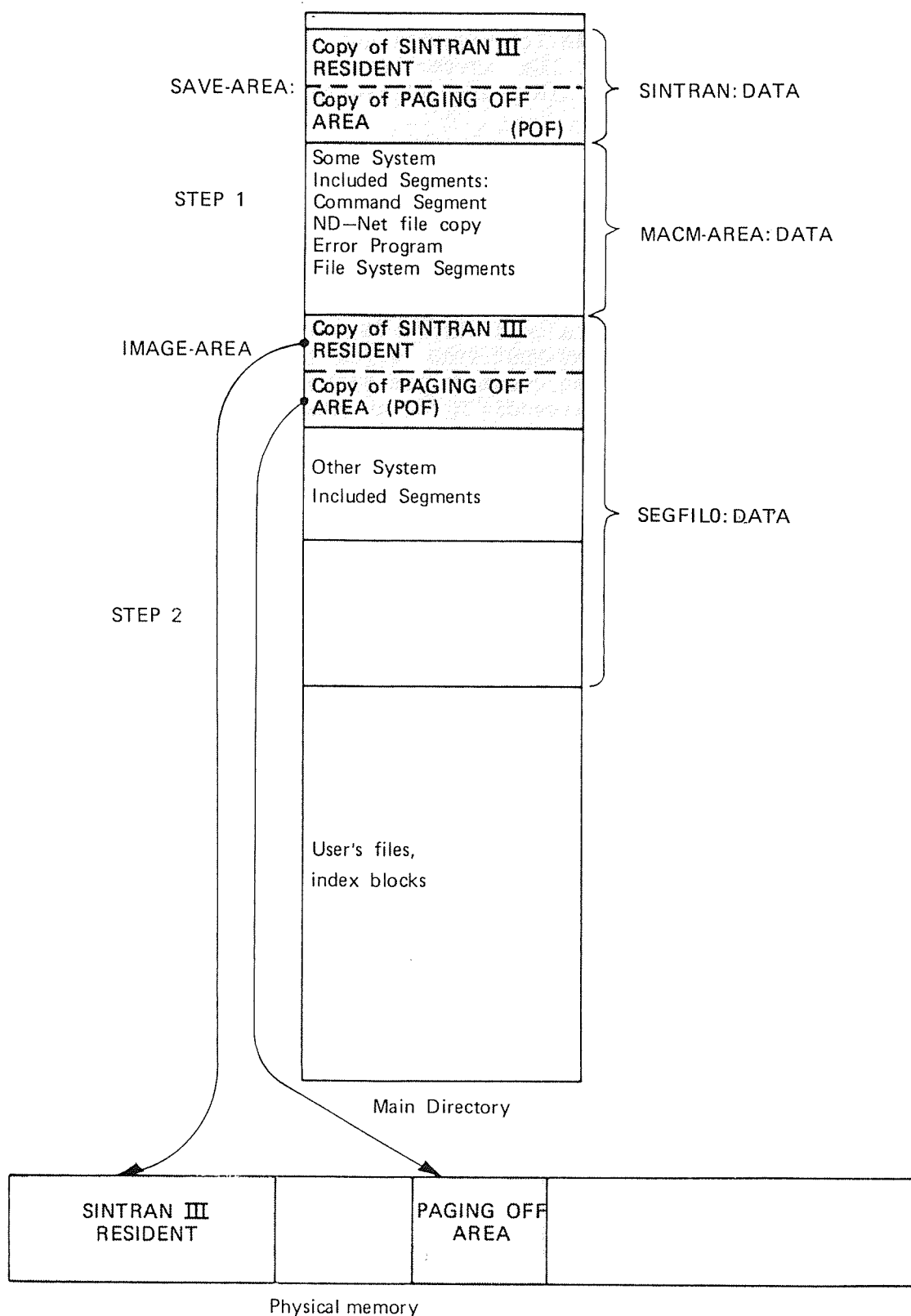


Figure 11. Overview of a Cold Start

The SAVE AREA copies of SINTRAN III RESIDENT and the PAGING OFF AREA are copied to the IMAGE AREA. The system segments on the files SINTRAN:DATA and MACM-AREA:DATA are copied to SEGFILO:DATA.

The SAVE AREAS of the RT LOADER, the spooling RT programs, the ND-NET program, the SINTRAN SERVICE PROGRAM, and MAIL are copied to other segments on SEGFILO:DATA. The copies of SINTRAN III RESIDENT and the PAGING OFF AREA on the IMAGE AREA are copied to physical memory.

A lot of information is lost in a cold start. It must be restored by executing the batch file HENT-MODE:SYMB which should contain all necessary commands. Some of these commands are system dependent, while others are always present. See appendix H for an example of a complete HENT-MODE:SYMB file.

Here is an example of commands in a HENT-MODE:SYMB file:

- 1) Set up the necessary initial commands to:
  - Enter the main directory.
  - Enter the directory on which user RT has space if this is not the main directory, and set this directory default.
  - Empty the temporary file used to contain output from the batch file LOAD-MODE:SYMB at each restart.
  - Start up a batch processor.
  - Put the file LOAD-MODE:SYMB in the batch queue.An example of this file is given in appendix H.
- 2) Execute @RTENTER to allow RT programs to access files.
- 3) Use the RT LOADER to define the name of all system-included RT Programs.
- 4) Initialize the background processes.
- 5) Start MAIL.
- 6) Initialize the batch processors.
- 7) Initialize all the spooling files.
- 8) Initialize the error log.
- 9) Load DMAC.
- 10) Load XMSG.
- 11) Put the file LOAD-MODE:SYMB in the batch queue.
- 12) Load any remote job entry systems (RJE) required.
- 13) Start a batch job to load the SIBAS database system.
- 14) Start a batch job to load all user RT programs.

HENT-MODE:SYMB

---

## 7.4 INITIAL LOADING

In most cases it is people from ND that load the system the very first time. They copy SINTRAN III from floppy disks to a disk, create different system files and users, and do the relevant patches.

The whole procedure will be explained in detail with an example in this chapter. Prerequisites for loading SINTRAN III from scratch are:

- 1) An ND computer with console, floppy disk and hard disk.
- 2) Five floppy disks containing SINTRAN III, MACM, and files called  
PATCHFILE:SYMB, SYMBOL-1-LIST:SYMB, SYMBOL-2-LIST:SYMB, DMAC:BPUN  
FMAC:PROG, LIBRARY-MARKS:SYMB, FILESYS-SYMBOLS:SYMB, and  
NEW-SYSTEM:PROG
- 3) Floppy disks with subsystems you need.
- 4) Necessary documentation, that is:
  - SOFTWARE SYSTEM REPORT sheets containing patches, see appendix I.
  - ND SOFTWARE LIBRARY PROGRAM DESCRIPTION sheets containing information about subsystems, see appendix J.
  - The manual SINTRAN III System Supervisor (ND-30.003).

---

#### 7.4.1 FORMATTING A DISK PACK

---

A new disk pack must be formatted before it can be used, ie., hardware addresses must be filled in and the disk must be checked for unusable pages. This is done by using stand-alone programs:

DIMS for cartridge disks (10 Mbyte).  
DISC-TEMA for all other disks.

The command to use is FORMAT. The programs are explained in detail in the manual Test Program Descriptions (ND-30.005). Section 10.2 describes how to run stand-alone programs. Note that the EEC disks should be formatted with spare track reallocation.

**WARNING!**

Any existing data on the disk are lost when the disk is formatted.

## 7.4.2 DISK SPACE FOR USER SYSTEM

Allocating disk space may be a problem when loading a new SINTRAN III system, especially with a small disk. The examples in the following sections will deal with loading to a small disk. The methods for solving these problems are valid for all disks.

A SINTRAN III operating system requires the system files shown in table 14. Those marked with an asterisk (\*), must be placed on the main swapping disk. These files must be taken into consideration when calculating how much space to give user SYSTEM.

*	SINTRAN:DATA	Allocated file - 63 pages.
*	MACM-AREA:DATA	Allocated file - 64 pages.
*	SEGFILO:DATA	Allocated file.
	(SEGFIL1:DATA)	Allocated file.
	(SEGFIL2:DATA)	Allocated file.
	(SEGFIL3:DATA)	Allocated file.
*	RTFIL:DATA	Indexed file.
*	MAILBOX:DATA	Indexed file.
*	Peripheral files, eg., LINE-PRINTER.	
*	(Spooling files, eg., LINE-PRINTER spooling files.)	
*	Subsystems in :PROG format, eg., ACCOUNTS.	
*	Subsystems in :BPUN format, eg., FORTRAN.	
*	Subsystem libraries in :BRF format, eg., FORTRAN-1BANK.	
*	Accounting file.	
*	LOAD-MODE:SYMB	See appendix H.
*	HENT-MODE:SYMB	See appendix H.
*	DUMP-REENTRANT:SYMB	See appendix H.
*	PATCHFILE:SYMB	See chapter 11.

Table 14. The System Files

The space required for the segment files depends upon:

- The number of terminals and batch processors (usually 133 pages each, or 69 pages on specially generated systems).
- The number and size of the reentrant subsystems.
- The number and size of remote job entry packages to be used.
- If SIBAS and/or TPS are to be run.
- The space required for user RT programs.

The space required for spooling files depend upon the use of the spooling system. The default number of spooling pages is 500. In addition, space is required for other users' files.

---

#### 7.4.3 DISK SPACE NEEDED BY USER SCRATCH

---

User SCRATCH owns the scratch files. Each terminal, batch processor, TAD, and ND-NET channels with background processes, needs a scratch file. The scratch files are indexed files. The RT LOADER, NOTIS WP, and many other subsystems use the scratch file.

You can refer to your scratch file by the file number 100. For example, when you compile a Fortran program, the BRF code can be copied to your scratch file by the command `COMPILE MYFILE, 1, 100`. The scratch file can later be used as input to the ND RELOCATING LOADER.

The number of background processes is printed when SINTRAN III starts. It is a good idea to create more scratch files than are currently required, since later installation of new terminals, batch processors or ND-NET channels with background processes will be simplified.

The space required for the scratch files depends on the number of background processes and the use of the scratch files. Each scratch file needs at least 32 pages. The subsystems' use of the scratch files is described in the PD sheets. See the PD sheet of NOTIS WP for an example.

#### **HINT**

Choosing a size of 50 pages multiplied by the number of background processes, will, in most cases, give sufficient space for user SCRATCH.

---

#### **7.4.4 THE SEGMENT FILES**

---

The segment files are contiguous areas on mass storage, where all segments in the SINTRAN III systems reside. Pages will be swapped between a segment and physical memory at run time. That is, when the code on a page on the segments is to be executed, the page will be fetched into physical memory.

It is possible to have four segment files. They should be named SEGFILO:DATA, SEGFI1:DATA, SEGFI2:DATA and SEGFI3:DATA. These files must be defined after they have been created, using the SINTRAN SERVICE PROGRAM command DEFINE-SEGMENT-FILE. See chapter 12. SEGFILO:DATA must always reside on the main swapping disk. The other segment files may be placed on any directory.

The space required by SINTRAN III itself, including all background segments, is printed out after a cold start or initial start. On a small disk it may happen that SEGFILO:DATA cannot be made big enough. In this case a file SEGFI1:DATA must be created for the background segments, except for the console. In addition, reentrant subsystems, remote job entry programs, user RT programs and SIBAS also need space on the segment files. See appendix E.

The maximum total space on the segment files is 16384 pages.

---

#### **7.4.5 AN EXAMPLE OF INITIAL LOADING**

---

This section describes how to copy SINTRAN III from floppy disks to mass storage the first time. Such initial loading is shown in fig. 12. When SINTRAN III is to be loaded again, the warm start procedure can be used.

# SINTRAN III System Supervisor Stopping, Starting and Loading SINTRAN III

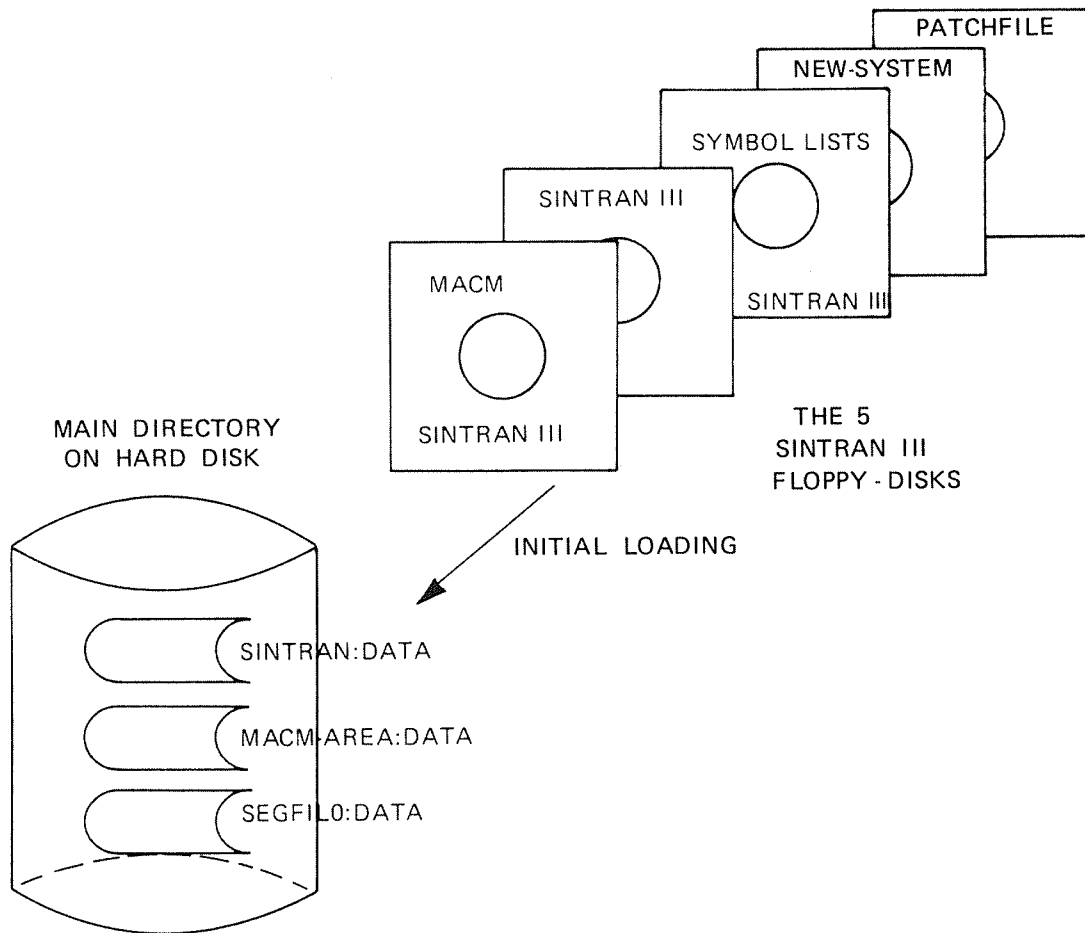


Figure 12. Initial Loading

The floppy disks should be inserted in the floppy disk unit in the order shown in figure 12. Then press the STOP and MCL buttons on the computers panel. The following is an example of initial loading from the console. The example is explained in the next section.



ND-30.003.06

SINTRAN III System Supervisor  
Stopping, Starting and Loading SINTRAN III

12.37.58          6 JANUARY      1984  
SINTRAN III - VSE - I REVISION:          6  
CPU (SYSTEM NUMBER):                  211  
GENERATED:    13.47.00                  5 DECEMBER      1983  
SINTRAN III RUNNING -

PAGES FOR SWAPPING(OCT.):          460

12.41.26          6 JANUARY      1984  
SINTRAN III - VSE I  
ENTER  
PASSWORD:

NO MAIN DIRECTORY  
OK  
@CREATE-DIRECTORY  
DIRECTORY NAME: PACK-ONE  
DEVICE NAME: DISC-75MB-1  
BIT FILE ADDRESS:  
@ENTER-DIRECTORY  
DIRECTORY NAME: PACK-ONE  
DEVICE NAME: DISC-75MB-1  
@CREATE-USER  
USER NAME: SYSTEM  
@LOG

12.43.51          6 JANUARY      1984  
--EXIT--

12.43.55          6 JANUARY      1984  
SINTRAN III - VSE I  
ENTER SYSTEM  
PASSWORD:

NO SUCH USER NAME IN MAIN DIRECTORY  
OK  
@GIVE-USER-SPACE  
USER NAME: SYSTEM  
NUMBER OF PAGES: 12000  
@ALLOCATE-FILE  
FILE NAME: SINTRAN  
PAGE ADDRESS: 1  
NUMBER OF PAGES: 77B  
@ALLOCATE-FILE  
FILE NAME: MACM-AREA  
PAGE ADDRESS: 100B  
NUMBER OF PAGES: 100B  
@ALLOCATE-FILE  
FILE NAME: SEGFILE0  
PAGE ADDRESS: 200B  
NUMBER OF PAGES: 2300  
@CREATE-USER  
USER NAME: SCRATCH  
@GIVE-USER-SPACE  
USER NAME: SCRATCH  
NUMBER OF PAGES: 1000

@CREATE-USER

USER NAME: RI

@CREATE-FRIEND

FRIEND NAME: RI

@CREATE-USER

USER NAME: FLOPPY-USER

@CREATE-USER

USER NAME: BPUN-FILES

@CREATE-FRIEND

FRIEND NAME: BPUN-FILES

@GIVE-USER-SPACE

USER NAME: BPUN-FILES

NUMBER OF PAGES: 1000

@LOG

12.59.28            6 JANUARY    1984

--EXIT--

12.59.35            6 JANUARY    1984

SINTRAN III - VSE I

ENTER SCRATCH

PASSWORD:

NO SUCH FILE NAME

OK

@SET-DEFAULT-FILE-ACCESS

PUBLIC ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWA

FRIEND ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWA

OWN ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWAD

@CREATE-FILE SCRATCH01,0

@CREATE-FILE SCRATCH02,0

@CREATE-FILE SCRATCH03,0

@CREATE-FILE SCRATCH04,0

@CREATE-FILE SCRATCH05,0

@CREATE-FILE SCRATCH06,0

@CREATE-FILE SCRATCH07,0

@CREATE-FILE SCRATCH08,0

@CREATE-FILE SCRATCH09,0

@CREATE-FILE SCRATCH10,0

@CREATE-FILE SCRATCH11,0

@CREATE-FILE SCRATCH12,0

@CREATE-FILE SCRATCH13,0

@CREATE-FILE SCRATCH14,0

@CREATE-FILE SCRATCH15,0

@CREATE-FILE SCRATCH16,0

@CREATE-FILE SCRATCH17,0

@CREATE-FILE SCRATCH18,0

@CREATE-FILE SCRATCH19,0

@CREATE-FILE SCRATCH20,0

@CREATE-FILE SCRATCH21,0

@CREATE-FILE SCRATCH22,0

@CREATE-FILE SCRATCH23,0

@CREATE-FILE SCRATCH24,0

@CREATE-FILE SCRATCH25,0

@LOG

13.04.01            6 JANUARY    1984

--EXIT--

SINTRAN III System Supervisor  
Stopping, Starting and Loading SINTRAN III

13.04.09      6 JANUARY    1984  
SINTRAN III - VSE I  
ENTER SYSTEM  
PASSWORD:  
OK  
@SINTRAN-SERVICE

\*DEFINE-SEGMENT-FILE  
MEMORY? Y  
SAVE-AREA? Y  
SEGMENT FILE NO: 0  
SEGMENT FILE NAME: SEGFILE0

\*EXIT

@INITIALIZE-BACKGROUND-PROGRAMS

@CREATE-FILE  
FILE NAME: RTFIL  
NUMBER OF PAGES: 0  
@SET-TEMPORARY-FILE  
FILE NAME: "SYSTEM-OUTPUT-1"  
@SET-TERMINAL-FILE  
FILE NAME: "TERMINAL"  
@SET-FILE-ACCESS  
FILE NAME: TERMINAL  
PUBLIC ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RW  
FRIEND ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RW  
OWN ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWD  
@SET-PERIPHERAL-FILE  
FILE NAME: "LINE-PRINTER"  
DEVICE NUMBER: 47B  
@SET-FILE-ACCESS  
FILE NAME: LINE-PRINTER  
PUBLIC ACCESS (R,W,A,C,D IN COMBINATIONS OR N): WA  
FRIEND ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWA  
OWN ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWD  
@CREATE-NEW-VERSION  
FILE NAME: LINE-PRINTER::10  
NUMBER OF PAGES: 0  
@SET-PERIPHERAL-FILE  
FILE NAME: "FLOPPY-0"  
DEVICE NUMBER: 1000  
@SET-FILE-ACCESS  
FILE NAME: FLOPPY-0  
PUBLIC ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWA  
FRIEND ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWA  
OWN ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWAD  
@CREATE-FILE  
FILE NAME: MAILBOX  
NUMBER OF PAGES: 0  
@SET-FILE-ACCESS  
FILE NAME: MAILBOX:DATA  
PUBLIC ACCESS (R,W,A,C,D IN COMBINATIONS OR N): N  
FRIEND ACCESS (R,W,A,C,D IN COMBINATIONS OR N): N  
OWN ACCESS (R,W,A,C,D IN COMBINATIONS OR N): RWAD

@MAIL

NOT INITIALIZED

\*INITIALIZE

MAX. NO. OF MESSAGES ( 512 WORDS EACH ): 10

\*EXIT

@LOG

13.19.12      6 JANUARY    1984

--EXIT--

---

---

#### **7.4.6 EXPLANATION OF THE EXAMPLE**

---

This section explains the output from the initial loading in the previous section. All numbers are octal.

---

#### **STEP 1**

---

Press ESCAPE to log in from the console. Answer with carriage return to ENTER and PASSWORD:. The message NO MAIN DIRECTORY will be output.

---

STEP 2

---

Update the SINTRAN III clock with the command @UPDAT if necessary.

---

STEP 3

---

Create a directory on unit 0 of the device specified when loading. This will take some time, for example, up to an hour on a 288 MByte disk if spare track reallocation is not used. Enter the directory and create user SYSTEM.

---

STEP 4

---

Log out and log in as user SYSTEM. The message NO SUCH USER NAME will be typed out, since an attempt is made to open a scratch file before user SCRATCH is created.

---

STEP 5

---

Give user SYSTEM sufficient user space. See Section 7.4.2.

---

STEP 6

---

Create the following system files by the command @ALLOCATE-FILE:

<b>SINTRAN:DATA</b> Start address: 1 Number of pages: 77 <sub>8</sub>	<b>MACM-AREA:DATA</b> Start address: 100 <sub>8</sub> Number of pages: 100 <sub>8</sub>	<b>SEGFILO:DATA</b> Start address: 200 <sub>8</sub> Number of pages: (See section 7.4.4)
---	---	---

If necessary, create the other segment files. Their sizes and start addresses depend upon the devices used. If more than one directory must be used for the segment files, the other directories must be created and entered, and space must be given to user SYSTEM on these directories.

---

STEP 7

---

Create user SCRATCH and give the user space. Each background process requires a scratch file. Decide how much space that is necessary for the scratch files and give the total number of pages to user SCRATCH. See section 7.4.3.

— STEP 8 —

Create user RT, FLOPPY-USER and BPUN-FILES. Make user RT a friend of user SYSTEM.

— STEP 9 —

Log out and log in as user SCRATCH. The message NO SUCH FILE NAME will be typed on the console since the scratch files have not yet been created.

— STEP 10 —

Set the default file access for user SCRATCH using the command:

@SET-DEFAULT-FILE-ACCESS RWA, RWA, RWAD

The scratch files will now automatically get the necessary file accesses at the time they are created.

— STEP 11 —

Create the scratch files SCRATCH01:DATA, SCRATCH02:DATA, etc. These are indexed files. Each background process requires one scratch file. One background process controls each:

- Terminal.
- Batch processor.
- Asynchronous modem with background.
- Terminal access device (TAD) used by COSMOS.
- ND-NET channel with background.

The number of background processes is printed out when SINTRAN III starts. However, a scratch file uses no space before it is actually used. It is advisable to create more scratch files than are currently required. Installation of new terminals, batch processors, ND-NET channels, etc., will then be simplified later.

— STEP 12 —

Log out and log in as user SYSTEM again. The message OK will be printed because the scratch file now exists. Only the console will be active. The background segment for this terminal has been placed on SEGFIL0:DATA. All other background processes must be initialized, but prior to this, the segment files must be defined.

---

STEP 13

---

Enter the SINTRAN SERVICE PROGRAM and define the segment files, including SEGFILO:DATA. The service program uses an asterisk (\*) as its prompt. An example is shown below:

```
@SINTRAN-SERVICE-PROGRAM
*DEFINE-SEGMENT-FILE
MEMORY? YES
SAVE AREA? YES
SEGMENT FILE NO: 0
SEGMENT FILE NAME: SEGFILO:DATA
*
```

(Similar for other segment files)

\*EXIT

---

STEP 14

---

The background processes may now be initialized by the command:

```
@INITIALIZE-BACKGROUND-PROGRAMS
```

If the segment file area can support the background segments, all the background processes will now be running, and the corresponding segments will be on the segment files. If the segment file area is too small, the following error message will be output:

```
NOT ENOUGH SPACE ON THE SEGMENT FILE(S)
```

Another segment file must be created, or the length of an existing segment file must be expanded. A new segment file must be defined in the SINTRAN service program and the @INITIALIZE-BACKGROUND-PROGRAMS command must be entered again. A file is expanded by the command @EXPAND-FILE or by redefining the file.



— STEP 15 —

Create the file RTFIL:DATA as an indexed file. File access for this file should be:

PUBLIC:	READ
FRIEND:	READ, WRITE, APPEND
OWN:	READ, WRITE, APPEND, COMMON, DIRECTORY

— STEP 16 —

Set the file name of the terminal by the command:

@SET-TERMINAL-FILE "TERMINAL"

File access for this file should be:

PUBLIC:	READ, WRITE
FRIEND:	READ, WRITE
OWN:	READ, WRITE, DIRECTORY

— STEP 17 —

Set peripheral files for all the peripherals. If the spooling system is to be used, the actual peripheral must be given spooling files. Details of the command are found in section 6.1 and the file access to be specified is found in section 6.1.1. Below is an example for a line printer:

@SET-PERIPHERAL-FILE "LINE-PRINTER", 5  
@CREATE-NEW-VERSION LINE-PRINTER::10, 0  
@SET-FILE-ACCESS LINE-PRINTER, WA, RWA, RWAD

---

STEP 18

---

Create the file MAILBOX:DATA as an indexed file. File access for this file should be:

PUBLIC:	NONE
FRIEND:	NONE
OWN:	READ, WRITE, APPEND, DIRECTORY

---

STEP 19

---

Enter the MAIL system by typing @MAIL. The MAIL system will respond with the message NOT INITIALIZED. An asteriks (\*) will be output as the prompt.

---

STEP 20

---

Initialize the size of the mailbox and return to SINTRAN III. In the following example, the mailbox may contain up to 10 messages.

```
*INITIALIZE 10  
*EXIT
```

The mailbox size is the maximum number of messages which the system will be able to handle. Each message requires 256 words on the file.

---

STEP 21

---

Give user SYSTEM and user RT secret passwords by the command @CHANGE-PASSWORD. The command will first ask for the old password. Respond with carriage return. The new password will not be printed on the terminal as you type it. This is done to keep it secret.

---

STEP 22

---

Copy all subsystems from floppy disks or magnetic tapes to hard disks under user SYSTEM. The easiest way is to use the COPY-USERS-FILES command in the BACKUP SYSTEM. See chapter 13.

---

— STEP 23 —

Load one of the editors QED, PED or NOTIS WP to write the batch files HENT-MODE:SYMB, LOAD-MODE:SYMB and DUMP-REENTRANT:SYMB. See appendix H for examples. Set their file access so that only user SYSTEM can read the files.

---

— STEP 24 —

Create the temporary file SYS-OUT-1:SYMB which will be used for output from the file LOAD-MODE:SYMB. If SIBAS is used, create the temporary file SYS-OUT-2:SYMB for output from the file SIBAS-LOAD:BATC. Temporary files are created by the command @SET-TEMPORARY-FILE.

---

— STEP 25 —

Create new users and give them appropriate space.

---

— STEP 26 —

Run the mode file SET-TERM-SPEED:SYMB to set terminal speeds. See appendix H for an example.

---

— STEP 27 —

Execute the program NEW-SYSTEM:PROG on the SINTRAN III floppy disk. See the procedure below.

@ENT-DIR

DIRECTORY NAME: N

DEVICE NAME: F-D-1

@(UN:SYS)NEW-SYSTEM

SINTRAN INITIATOR V.3.0 FOR SINTRAN-III VSE/VSX VER.I NOVEMBER 18, 1983

Computer : ND-100 with 48 bit floating point arithmetic

Instruction set : Standard

Operating system : SINTRAN-III VSE

Generated : 83.12. 5

Time according to SINTRAN: 13.41.45 6 JANUARY 1984

Do you want to update the clock (default=Y): NQ

Copying files from Sintran floppies.

Copying file: FILSYS-SYMBOLS:SYMB

Copying file: FMAC-1408D:PROG

Copying file: DMAC-1915E:BPUN

Replace the floppy in the FLOPPY-DISK-1, unit 0, with the 4th system floppy (Sintran-diskette-IV).

Ready to continue (default=Y): Y

Copying file: SYMBOL-1-LIST:SYMB

Copying file: SYMBOL-2-LIST:SYMB

Copying file: LIBRARY-MARKS:SYMB

Replace the floppy in the FLOPPY-DISK-1, unit 0, with the 5th system floppy (the patch floppy).

Ready to continue (default=Y):

General checking of the system will take approx. 30 seconds.

Starting patch-file.

During patching of Sintran the Patch-file system will produce output to the files:

--> (PATCH-SINTRAN:SYSTEM)PATCH-FILE and

--> (SYSTEM)PATCH-OUT:OUT

In case of abnormal termination of the patching (i.e. if you return to Sintran command mode without a message telling whether the patching is aborted or successfully completed) the files listed above should be investigated.

Messages will appear on the terminal...

--> Defining SEGFIL0 and loading DMAC.

--> Creating necessary files.

--> Checking the symbol lists.

--> Defining necessary symbols.

--> Start patching of SINTRAN-III.

--> Checking if patching went ok.

During checking of the patch output file a list of all patches done in your system, will be written to the file PATCH-OUT:OUT on the Patch-file. Do you want a copy to the terminal too (default=N):

Patching of Sintran-III successfully completed. Please do a cold start (ACOLD-START) of your system to get the patches into effect.

@

---

— STEP 28 —

---

Do a cold start, first running the batch file CONTROL-SEGMENTS:SYMB shown below, then the batch file HENT-MODE:SYMB. See section 7.3 for a description of cold starts and appendix H for an example of a HENT-MODE:SYMB file.

```
@CC =====
@CC = (SYSTEM)CONTROL-SEGMENTS:SYMB                      =
@CC = THIS FILE SHOULD BE RUN EACH TIME A NEW SINTRAN III  =
@CC = HAS BEEN INSTALLED AND BEFORE RUNNING THE HENT-MODE FILE. =
@CC = TAKE A LOOK AT FREE SEGMENTS AND SEGMENTS USED BY YOUR  =
@CC = SUBSYSTEM.                                             =
@CC =====
@CC =====
@CC = LOAD DMAC =
@CC =====
@RTENTER
@RT-LOADER
YES
READ-BINARY DMAC,7
YES
EXIT
@INITIALIZE-BACKGROUND-PROGRAMS
@RT-LOADER
LIST-FREE-SEGMENT,,,
EXIT
```

---

---

---

#### 7.4.7 LOADING SUBSYSTEMS

---

Subsystems are delivered on floppy disks in prog or bpun format, except for the routine libraries which are in BRF format. Software library object description sheets follow these floppy disks. See appendix J for an example.

All subsystems should be copied directly to disk files using the BACKUP SYSTEM's command COPY-USERS-FILES. The routine libraries should be copied to user SYSTEM's main directory, the other to optional directories.

The subsystems in bpun format may be dumped to ordinary prog files by the commands @PLACE-BINARY and @DUMP. They may also be dumped to segments as reentrant subsystems by the @DUMP-REENTRANT command. The possibilities are shown below:

```
@PLACE-BINARY FORTRAN-100-A:BPUN  
@DUMP "FORTRAN-100-A:PROG",0,1
```

or

```
@DUMP-REENTRANT FORTRAN-100-A,0,1,FORTRAN-100-A:BPUN
```

or, for subsystems in prog file format:

```
@DUMP-PROGRAM-REENTRANT FORTRAN-100-A,FORTRAN-100-A:PROG
```

When public users want to use the same subsystem simultaneously, the subsystem should be generated as a reentrant subsystem. This allowing many users to share it.

The batch file DUMP-REENTRANT:SYMB contains dump commands for the reentrant subsystems. Appendix H shows an example of this file. The commands in this file are executed after a cold start.

---

## 7.5 LOADING A NEW SINTRAN III VERSION

---

If you, for example, add terminals, a new line printer with spooling, or a new disk unit to the computer system, a new SINTRAN III has to be ordered. The system supervisor would then have to load the new SINTRAN III to an existing directory. Such loading should also be done for new SINTRAN III versions, when the SAVE AREA has been corrupted, or before running a new patch file.

It may happen that the existing segment file area is too small. This is no great problem, as long as contiguous space is available on some disk pack. It does not necessarily have to be on the same disk pack as SINTRAN III. This problem may wait until SINTRAN III has been started up.

If new peripherals are included, the necessary peripheral files must be created. If new background processes are added, corresponding scratch files must be created, etc. It may also happen that the files LOAD-MODE:SYMB and HENT-MODE:SYMB must be adjusted. Certain commands must also be executed.

The following procedure is used to load a new version of SINTRAN III:

— STEP 1 —

---

Perform the operations explained in section 7.4.5 up to 22! to load the new SINTRAN III. The following will be printed on the console:

```
OCTAL NO. OF PAGES THE SYSTEM WILL USE ON THE SEGMENT
FILE(S): <segment file area required for the system>
FIRST SYSTEM SEGMENT STARTS ON PAGE (OCT): <address>
NUMBER OF BACKGROUND PROCESSES (DEC.) : <number>
EACH BACKGROUND PROCESS NEEDS (OCT. PAGES): 205
```

SINTRAN III RUNNING

PAGES FOR SWAPPING (OCT.): <number of pages for swapping>

The output shows the size of the segment file area required for the system, the start of the first system segment, the number of pages for swapping including system segments, and the total number of background processes present in system. This information will indicate whether or not the existing segment file area is large enough.

Space must be added for reentrant subsystems, SIBAS, RT programs and remote job entry systems. Note also that a background segment cannot be divided, and that the segment area required for the system is constructed from modules of 133 pages. If the segment file area is too small, it must be increased after SINTRAN III has started.

— STEP 2 —

---

Enter the main directory and log in from the console as user SYSTEM as shown below:

- Press ESCAPE.
- After ENTER give carriage return.
- After PASSWORD give carriage return.
- The message NO USER ENTERED will be typed.
- Update the clock.
- Enter the main directory.
- Log out.
- Log in as user SYSTEM.

---

STEP 3

---

If any new peripherals have been added to the system, these must be created as peripheral files. See Section 7.4.6. If any new background processes have been added, log in as user SCRATCH and create corresponding scratch files.

---

STEP 4

---

SINTRAN III is now running with only the console active. The definition of segment files and the initialization of the other background processes proceeds as in section 7.4.6.

---

STEP 5

---

Load QED, PED or NOTIS WP, and update the files HENT-MODE:SYMB, LOAD-MODE:SYMB and DUMP-REENTRANT:SYMB if necessary. The page mode editors cannot be used from the console.

---

STEP 6

---

Run the mode file PATCHFILE:SYMB found on the fifth floppy disk with SINTRAN III and log out.

---

STEP 7

---

Do a cold start to run the HENT-MODE:SYMB file. See section 7.3.



---

## 7.6 ADDITIONAL PROCEDURES FOR ND-500 INSTALLATIONS

---

The rest of this chapter is only relevant to ND-500 computer systems. It shows how to install, start and stop the ND-500 part of an ND-500 system. The ND-100 part must have been started.

The commands and concepts related to the ND-500 are described in detail in the manual ND-500 LOADER/MONITOR (ND-60.136).

---

### 7.6.1 LOADING ND-500 SOFTWARE FROM FLOPPY DISKS

---

The ND-500 CPU is controlled by the ND-100, and part of the additional software is loaded to the ND-100. The software comes on the following three floppy disks:

- The ND-500 MONITOR (ND-10333C-PART1:FLOPPY-USER)
  - File 0: SEG30-MONITOR-C:BPUN
  - File 1: SEG31-MONITOR-C:BPUN
  - File 2: ND-500-MONITOR-C:BPUN
- The SWAPPER (ND-10333C-PART2:FLOPPY-USER)
  - File 0: SWAPPER:PSEG
  - File 1: SWAPPER:DSEG
- The ND-500 MICRO PROGRAM (ND-10332C:SYSTEM)
  - File 0: CONT-STORE-10500:DATA

The files containing the SWAPPER and the MICRO PROGRAM have to be copied to user SYSTEM. The other files may be copied, for example, to user BPUN-FILES, or loaded to their segments:

<u>@COPY-FILE</u>	
DESTINATION FILE: "SWAPPER:PSEG"	
SOURCE FILE: <u>(ND:FLOPPY-USER)SWAPPER:PSEG</u>	ND-500 programs use file types :PSEG, :DSEG, and :LINK.

It is important to notice that the files containing the SWAPPER have to be named SWAPPER:PSEG and SWAPPER:DSEG. The file containing the MICRO PROGRAM has to be named CONTROL-STORE:DATA. This is because the ND-500 MONITOR accepts default values in commands that have to do with the SWAPPER and the control store.

---

### 7.6.1.1 LOADING TO SEGMENTS

---

The ND-100 uses the subsystem ND-500 MONITOR to control the ND-500. When the ND-500 system files have been copied from the floppy disks, the ND-500 MONITOR may be loaded to segments:

```
@RT-LOADER
*READ-BINARY (BPUN-FILES)ND-500-SEG30-F,30
*YES
*READ-BINARY (BPUN-FILES)ND-500-SEG31-F,31
*YES
*EXIT
@DUMP-PROGRAM-REENTRANT ND-500-MONITOR,(BPUN-FILES)ND-500-MONIT-F
```

The commands are normally given from a batch file.

---

### 7.6.1.2 THE SWAP FILES

---

When you initialize the ND-500 MONITOR, you have to define which files are going to be used as swap files, ie., the swapping area for ND-500 segments. The following rules pertain to the establishment of swap files:

- A maximum of four swap files are allowed.
- Names should be SWAP-FILE-1:SWAP, etc., and they must be of type :SWAP.
- They should be contiguous files.
- They may belong to any user, but SYSTEM must have read and write access.
- They may exist on any directory.

---

### 7.6.1.3 DEFINING MEMORY CONFIGURATION AND SWAP FILES

---

Before you can start using the ND-500 MONITOR, you must define:

- Your memory configuration.
- Which files will be used as swap files.

How the memory configuration is defined at any one time may be investigated by the ND-500 MONITOR command below. See also the manual ND-500 Loader/Monitor (ND 60.136). N500 is the prompt used by the ND-500 MONITOR.

```
N500: MEMORY-CONFIGURATION
N100 PAGE NO. FOR N500 ADDREE ZERO: 000600
N500 PAGE NO. FOR THE N500 REGISTER BLOCK: 000001
```

This is one example of the output from this command. You should use the output as in the command sequence below:

```
@ND-500-MONITOR
N500: SET-ND-500-UNAVAILABLE
N500: DEFINE-MEMORY-CONFIGURATION
N100 PAGE NO FOR N500 PHYS. ADR. ZERO: 600
000001.MEMORY PART
SIZE (IN 2K BYTES): 1000
ND-100 ACCESS: YES
ND-500 PROGRAM ACCESS: YES
ND-500 DATA ACCESS: YES

LAST MEMORY PART DEFINED? YES

N500: DEFINE-SWAP-FILE SWAP-FILE-1:SWAP
N500: SET ND-500-AVAILABLE
N500: EXIT
```

The commands GIVE-ND-500-PAGES and TAKE-ND-500-PAGES will add and reduce the number of pages shared by the the two CPUs.

---

### 7.6.2 STARTING THE ND-500

---

To start up the ND-500, there are first three more tasks to accomplish:

- 1) Load the MICRO PROGRAM into the control store and start it.
  - 2) Load and start the SWAPPER. The SWAPPER always must run as process number 0.
  - 3) Give pages in the physical memory shared by the ND-500 and the ND-100 to the ND-500.

This can be done in two ways. When a process is started up, the monitor itself automatically performs the initialization:

```
@ND-500-MONITOR
N500: RECOVER.....
```

The alternative is to give the commands:

```
@ND-500-MONITOR
N500: LOAD-CONTROL-STORE, , , , ,
N500: MICRO-START, 0
N500: LOAD-SWAPPER, ,
N500: START-SWAPPER, ,
N500: GIVE-N500-PAGES, 299D
```

---

### 7.6.3 STOPPING THE ND-500

---

When the ND-500 must be stopped, check first that no processes are active, and let no new users enter the monitor. Below are some commands that can be used to check the activity on the ND-500, and also force a process out:

```
SET-ND-500-UNAVAILABLE
WHO-IS-ON
LIST-ACTIVE-PROCESSES
PROCESS-STATUS
LOGOUT-PROCESS <process no>
ABORT-PROCESS <process no>
.
.
SET-ND-500-AVAILABLE
```

There are three ways to stop the ND-500:

- 1) N500: MICRO-STOP  
The execution of the ND-500 MICRO PROGRAM is stopped, all registers are saved and execution may be resumed.
- 2) N500: STOP-ND-500  
The ND-500 CPU is stopped and execution cannot be resumed.
- 3) N500: MASTER-CLEAR  
Brings the ND-500 out of a state where it is hanging. It stops the ND-500 immediately, but the contents of the registers are unpredictable.

---

#### 7.6.4 LOADING SUBSYSTEMS ON THE ND-500

---

A subsystem on a floppy disk is a domain. The commands @COPY-FILE or @COPY cannot be used, nor the BACKUP SYSTEM. Such copying would destroy the DESCRIPTION-FILE:DESC of user SYSTEM, and all information about previously created domains would be lost. The rule is therefore to use the LINKAGE LOADER command COPY-DOMAIN.

---

##### 7.6.4.1 THE LINKAGE LOADER

---

The LINKAGE LOADER comes on a separate floppy disk containing the following files:

```
ND500 LINKAGE-LOADER (ND-10391E:FLOPPY-USER):  
      File 0: DESCRIPTION-FILE:DESC  
      File 1: LINKAGE-LOADER:LINK  
      File 2: LINKAGE-LOADER:DSEG  
      File 3: LINKAGE-LOADER:PSEG
```

The loading procedure for the LINKAGE LOADER is:

```
@ENTER-DIRECTORY,,FLOPPY-DISC-1,0  
@ND-500-MONITOR  
ND-500 MONITOR   VERSION F  
N500: RECOVER (ND:FLOPPY-USER)LINKAGE-LOADER  
ND LINKAGE LOADER   VERSION F  
N11: COPY-DOMAIN (ND:FL-US)LINK-LOAD "LINKAGE-LOADER-F"  
N11: EXIT  
N500: EXIT  
@RELEASE-DIRECTORY ND
```

The LINKAGE LOADER is used to load all subsystems to the ND-500. Each copied domain will produce several files with file type :PSEG, :DSEG and :LINK.

---

**7.6.4.2 EXAMPLE OF LOADING A SUBSYSTEM ON THE ND-500**

---

Below is an example of loading the FORTRAN-500 compiler from floppy disk:

```
@ENTER-DIRECTORY,,FLOPPY-DISC-1,0
@ND-500-MONITOR
ND-500 MONITOR   VERSION F
N500: LINKAGE-LOADER
ND LINKAGE LOADER   VERSION F
N11: DELETE-DOMAIN
DOMAIN NAME: FORTRAN-500
N11: COPY-DOMAIN
SOURCE DOMAIN: (ND--PART1:FL-US)FORTRAN-500-H
DESTINATION DOMAIN: "FORTRAN-500-H"
N11: EXIT
N500: DEFINE-STANDARD-DOMAIN FORTRAN-500-H,FORTRAN-500-H
N500: EXIT
@RELEASE-DIRECTORY ND--PART1
```

Commonly used domains, eg., compilers, editors and the LINKAGE LOADER, should be defined as standard domains. This will will reduce start up time.

---

## 8 MEASURING AND IMPROVING THE SYSTEM PERFORMANCE

---

This chapter describes how you can measure and improve the performance of SINTRAN III on your installation. If you have problems with the performance, a number of questions must be answered in order to clarify the problem. The most important of these are shown in table 15.

QUESTIONS RELATED TO SYSTEM PERFORMANCE
<ul style="list-style-type: none"><li>- How much memory is available for swapping ?</li><li>- How much memory is used by each application ?</li><li>- How many of the terminals are running each application ?</li><li>- How many batch processes are running ?</li><li>- What kind of tasks are being run under the batch processors, and what are their priorities ?</li><li>- Is the spooling system being used ? If so, what is the estimated print volume per time period ?</li><li>- Are any special RT programs being run ? If so, what are their priorities ?</li></ul>

Table 15. Questions Related to System Performance

The response time is the time it takes from the last user typed character is given until the first character is output on the screen. A number of facilities for measuring the use of resources are present in SINTRAN III. They are:

RT PROGRAM LOG  
PROGRAM LOG  
HISTOGRAM  
SYSTEM HISTOGRAM  
THE @TIME-USED COMMAND

In addition, the Fortran compilers have a logging facility called PROFILE-MAP. See the FORTRAN Reference Manual (ND-60.145).

---

### 8.1 MEASURING RT PROGRAMS

---

The RT PROGRAM LOG measures resource usage for a particular RT program and/or the system as a whole. It is also quite possible to measure the resource demands submitted from any terminal, as each terminal internally is controlled by an RT program.

Before running the RT PROGRAM LOG, the terminal from which the log is to be run should be removed from the time slicer. Its priority should be set higher than any other "ordinary" programs. The terminal used is normally the console. It is represented internally by the RT program BAK01. A priority of 60 is enough if no special RT programs with a higher priority are present in the system.

Increasing the priority is done to achieve measurements which are as accurate as possible. The procedure to prepare a terminal for using the RT PROGRAM LOG is:

- 1) Find the terminal number of your terminal. The command @WHO-IS-ON can be used.
- 2) Use the @LIST-DEVICE command to find out which RT program controls that terminal.
- 3) Call up the SINTRAN SERVICE PROGRAM and remove the RT program controlling the terminal from the time slicer.
- 4) Use the @PRIOR command to set the priority of the RT program.

The RT programs that controls terminals and batch processors are called background processes. The names of these RT programs are BAKnn and BKnnn for terminals, and BCHnn for batch processors, where n denotes a digit.

The time slicer is the part of the operating system that lets users share the CPU. The following is an example of preparing to use the RT PROGRAM LOG:



@WHO-IS-ON

```
===>      1  SYSTEM
           45  HANSEN-A
           560 PRICING-PBO
           561 SIB2
```

@LIST-DEVICE

```
LOGICAL DEVICE NUMBER: 1
INPUT/OUTPUT(0 OR 1): 0
RESERVED BY: BAK01
```

@SINTRAN-SERVICE-PROGRAM

```
*REMOVE-FROM-TIME-SLICE
LOGICAL DEVICE NUMBER: 1D
MEMORY? YES
IMAGE? NO
SAVE AREA? NO
*EXIT
```

@PRIOR

```
PR PROGRAM NAME: BAK01
PRIORITY: 64
```

The arrow indicates your terminal number. This is used as input to the next command.

This command will list the RT program which has reserved terminal number 1, ie., BAK01.

Then remove the terminal from the time slicer. It is necessary to be logged in as user SYSTEM to give these commands.

Set a new priority higher than 60 for the background process.

After the RT PROGRAM LOG the background process should be inserted in the time slicer again. Use the SINTRAN SERVICE PROGRAM command INSERT-IN-TIME-SLICE.

---

### 8.1.1 PREPARATION FOR MEASURING RT PROGRAMS

---

The database program SIBAS is measured in the following example. It has the RT program name SIBA. If you are interested only in measuring the overall system utilization, you can give carriage return instead of the name of an RT program.

@RT-PROGRAM-LOG

```
RT PROGRAM NAME: SIBA
REPORT INTERVAL(SEC): 10
INTERRUPTS/SAMPLE: 1
LOGICAL DEVICE NUMBER.: 160
INPUT/OUTPUT (0 OR 1): 0
LOGICAL DEVICE NUMBER: 161
INPUT/OUTPUT (0 OR 1): 1
OUTPUT FILE:
```

The parameter <report interval> is the number of seconds between each report. Each report will output one line with sampling results. Use a short interval, eg., 5-10 seconds, if you are interested in rapid fluctuations. If the overall load is of interest, use a longer time, eg., 30-120 seconds. The default value is 60 seconds.

The parameter <interrupts/sample> is the number of output interrupts of the user's terminal between each sample of the system state. Non-printable characters may be output to the terminal. That is why the cursor will disappear on screen terminals. This parameter has a default value of 8.

An example illustrates this: The number of output interrupts per second from a 9600 baud terminal is  $9600/11 = 873$ . Division by 11 is done because the terminal uses 11 bits to send one character with the necessary control information. In the same way, the number of interrupts per second from a 300 baud terminal, eg., a console, is given by  $300/11 = 27$ .

Assume an interval of 10 seconds is chosen. The number of samples taken during each 10 second period should be large enough to produce statistically significant results. If we use 1 interrupt/sample for a 300 baud terminal, we would be sampling at a rate of 27 samples/second. A 10 second period gives 270 samples, which is large enough to give us reliable results.

If we were using 2 interrupts per sample, the number of samples in the 10 second period would be half as many, ie., 135. This number is on the verge of being too small to produce reliable results. The calculation procedure is just the same for all other line speeds, only the numbers are different.

The overhead from a 300 baud, or about 27 characters per second terminal, eg., a console, at 1 interrupt per sample is negligible. More precisely, it is less than 1% of the CPU time. A 9600 baud terminal sampling at maximum rate, ie., 873 samples/second, will use about 14% of the CPU time.

For a 300 baud terminal, use 1 interrupts/sample
For a 1200 baud terminal, use 4 interrupts/sample
For a 4800 baud terminal, use 16 interrupts/sample
For a 9600 baud terminal, use 32 interrupts/sample

Table 16. Interrupts per Samples for Some Terminals

An easy-to-remember way of selecting the <interrupts/sample> parameter is given in table 16. Divide the baud rate of the terminal being used by 300 baud and use this result as the number of interrupts per sample. This means that the sampling rate will be about 27 samples per second.

The <logical device number> parameter may be used to identify one or two input/output devices for which the degree of utilization is to be logged. If a device is reserved, it is defined as being in use. The devices often logged are the SIBAS internal devices. These are reserved by a user program when a SIBAS call is made and released when the call has been executed. In this way the degree of utilization of SIBAS may be measured.

---

### 8.1.2 RESULTS FROM MEASURING RT PROGRAMS

---

The output from each report from the RT PROGRAM LOG command will look like:

CPU	SWAP	FILES	DISC	PASSIVE	IO-WAIT	UNIT 160	UNIT 161
25/38	02/05	00/05	10	00	75	00/32	00/31

The first figure of each pair for CPU, SWAP, FILES and the two logical device numbers (units) shows the RT program's percentage utilization of the corresponding resource. The figures for PASSIVE and IO-WAIT refer also to the RT program. The second figure in each pair and the DISC figure show the total percentage utilization of that resource.

In this example, the CPU was used 25% by the given RT program and 38% by the whole system, including the RT program. The figures for SWAP mean that out of 100 samples, the given program caused page swapping twice. Some other programs caused page swapping 5 times. Swapping means the transfer of pages between the disk and main memory performed by the paging system.

FILES means normal use of files, ie., user access. The DISC figure is approximately the sum of the total SWAP and FILES figures, as these two are the only sources of disk access.

All percentages are related to the given <report interval>. If the <report interval> is set to 10 seconds, and the DISC figure becomes 40%, then the disk system has been occupied 4 seconds out of 10. This includes time to set up disk transfers, seek time, and transfer time.

The CPU-, PASSIVE-, and IO-WAIT figures for a specific RT program add up to a certain percentage figure. In this example they are 25% + 0% + 75% = 100%. This percentage will be equal or close to 100% if the load on the system is small. It will drop below 100% when the work load increases.

The missing percentages give us an idea of the queue lengths in the system. When an RT program is waiting for the CPU or is in the queue waiting to reserve some input/output device, then it will not be classified in any of the groups defined by RT PROGRAM LOG. So when a sample is taken when the RT program is in some device queue, this will be a "missing" sample.

---

## 8.2 RELATIVE USE OF THE CPU

---

The PROGRAM LOG measures the relative amount of CPU time used by each RT program in a given time interval. The log is started by giving the command @START-PROGRAM-LOG which has <interrupts/sample> as its only parameter. This parameter is the same as that in the @RT-PROGRAM-LOG command described in the previous section.

The log is stopped by the command @STOP-PROGRAM-LOG which has an output file for the results as its only parameter. The following example shows a PROGRAM LOG:

```
@START-PROGRAM-LOG
INTERRUPTS/SAMPLE: 30
```

```
@STOP-PROGRAM-LOG
OUTPUT FILE: TERMINAL
```

	PERCENT	SAMPLES
DUMMY	82	997
STSIN	00	0
RTERR	00	0
RTSLI	01	11
.	.	.
BAK08	17	205
.	.	.

In the interval the log was operational, the program BAK08 was using the CPU in 17% and DUMMY in 82% of the samples. DUMMY is the RT program which executes when no other programs ask for execution, ie., when the system is idle. It has priority 0.

---

## 8.3 MEASURING USE OF AN ADDRESS SPACE

---

The CPU histogram measures the amount of CPU time spent in different parts of the virtual address space of an RT program. The parameters for the histogram are defined by the @DEFINE-HISTOGRAM command. They are the name of the RT program, the start address of the virtual address area to be logged and the address interval.

This will result in 64 equally sized intervals being logged. This means that the parameters you choose must satisfy the following equation. All figures and parameters are octal.

$$\text{START ADDRESS} + \text{INTERVAL} * 100 < 200000$$

The log may be started by the @START-HISTOGRAM command and stopped by @STOP-HISTOGRAM. The results will be printed by the command @PRINT-HISTOGRAM which has an output file as its only parameter.

@DEFINE-HISTOGRAM

RT NAME: GARP

START ADDRESS: 0

INTERVAL: 2000B

@START-HISTOGRAM

@STOP-HISTOGRAM

@PRINT-HISTOGRAM

OUTPUT FILE: TERMINAL

	PERCENT	SAMPLES
OUTSIDE	0	0 OUT OF 12571
SYSTEM:	1	126
0- 1777	0	0
2000- 3777	7	880
4000- 5777	9	1131
.	.	.
176000-177777	0	0

The term "system" here means CPU time spent on interrupt level 1, protection ring 1, 2 or 3. Level 1 means monitor call code is executing. For an explanation of interrupt levels and protection rings, see the manual SINTRAN III Real Time Guide (ND-60.133).

---

## 8.4 MEASURING THE USE OF PHYSICAL MEMORY

---

The SYSTEM HISTOGRAM measures the relative amount of CPU time used within sections of the physical memory on a specified interrupt level. The interrupt level, the start address, and the size of the increments of the memory area to be logged may be defined by the command @DEFINE-SYSTEM-HISTOGRAM.

This will log 64 equally-sized memory intervals. The log may be started by the @START-HISTOGRAM command and stopped by the command @STOP-HISTOGRAM. The results will be printed by @PRINT-HISTOGRAM which has one parameter, the name of the output file.

For example, if interest is centered around CPU use on interrupt level 1 in the physical memory address area 2000<sub>8</sub> to 3000<sub>8</sub>, then the following command sequence should be used:

@DEFINE-SYSTEM-HISTOGRAM

LEVEL: 1

START ADDRESS: 2000

INTERVAL: 10

@START-HISTOGRAM

@STOP-HISTOGRAM

@PRINT-HISTOGRAM

OUTPUT FILE: TERMINAL

	PERCENT	SAMPLES
OUTSIDE:	03	127 out of 3868
2000-2007	00	0
2010-2017	00	0
2740-2747	97	3740
.	.	.

---

### 8.5 LISTING CPU TIME USED

---

The @TIME-USED command returns the CPU time used since the terminal was logged in. The CPU time includes the time used by the inbyte and outbyte routines on interrupt level 4 and by the user program itself.

It is important to notice that this is never the total CPU time used, since time spent on levels 14, 13, 12, 11, 10, 3 and 1 (ring 1, 2 or 3) is not included. The operating system overhead, apart from monitor calls for certain character handling, is not included in the measured CPU time.

Table 17 shows a list of the activities on the 16 different interrupt levels.

Level	Activity
0	The system is in an idle loop.
1	All RT programs execute on this level.
2	Not used.
3	The kernel of SINTRAN III executes here.
4	Inbyte and outbyte monitor calls.
5	XMSG.
6	Not used.
7	Not used.
8	Not used.
9	Not used.
10	Driver routines for character device output.
11	Driver routines for mass storage devices.
12	Driver routines for character device input.
13	Updating of real time clock.
14	Internal interrupts, eg., page faults, power fail, etc.
15	Not used.

Table 17. The Interrupt Levels

The fraction of CPU time not included in @TIME-USED will vary from program to program. It is often between 1 - 30% of the total CPU time used. If we exclude certain monitor calls for character handling, the more monitor calls a program executes, the bigger the fraction of CPU time not included in @TIME-USED is.

---

## 8.6 DIAGNOSIS

---

The problem may be seen after just one run of the RT PROGRAM LOG. If for example the CPU load is approaching 100 % or page swapping is permanently higher than 10-20%, then the direct reason for bad response times may be obvious.

Theoretical calculations, however, should be made when possible, and compared with the measurements. Large discrepancies between the actual and the theoretical results should be questioned, since in some cases a fairly trivial misunderstanding may be the cause of the problem. For example, a bad choice of break strategy may result in the CPU load being up to 60% higher than is necessary.

A swap rate higher than 10-20% which means that too little physical memory is available, is often the most critical factor in an overloaded system. The swapping mechanism will consume both CPU and input/output resources, and the disk utilization may exceed 30-35%, which often is considered a critical limit.

A very high utilization of the CPU, eg., 70-100% does not necessarily mean that the situation is disastrous. If the different tasks in the system have priorities according to their importance, this can on the contrary be an ideal situation. Priorities are described in the next section.

---

### 8.7 SOLUTION

---

If the system is running badly due to lack of resources, one solution is to add more hardware. The other is to find out which programs use up the available resources. These can be optimized with respect to the resource, or you can lower their priority. In the latter case you must accept that they run more slowly.

#### SOLUTIONS TO RESOURCE SCARCITY

- 1) Optimize the program with respect to the resource.
- 2) Lower the relative priority of that program and accepting that it will run slower.
- 3) Add more hardware.

Similar performance tests can be performed on an ND-500 system. See the manual ND-500 LOADER/MONITOR (ND-60.136).



---

## 9 ERROR SUPERVISING

---

This chapter describes how system failures are detected, and how these failures are corrected. System failures are considered to include all types of errors and irregularities that cause the system to stop or run with lowered performance.

Whenever a system failure occurs, it is the system supervisor's responsibility to take charge of the installation. The system supervisor should identify the type of system failure, get all the necessary information to describe the state of the computer, and try to get the installation working again.

Error situations are usually detected through error messages from SINTRAN III. Error procedures are given for different situations. The state of the system is indicated by the lights on the computer's panel:

- The computer is stopped when the STOP light is on.
- The computer hangs when the STOP light is off, but the computer does not respond.
- The computer is dead when no lights are on.

System failures are either fatal or nonfatal:

Nonfatal errors are detected by SINTRAN III. Error message will appear on the user's terminal or on the console. Only error messages from the central parts of SINTRAN III will be covered in this chapter. These will be the most useful from the system supervisor's point of view. Some of them may indicate errors in the hardware or in the SINTRAN III operating system.

Fatal errors will almost all be detected by SINTRAN III, but no error message will be given. The system will go into a stop condition or it will hang.

---

**9.1 ERROR LOGGING**

---

Error messages from RT programs and system start up messages are written to the error device. This is normally the console, ie., the terminal with logical device number 1. However, it is possible to select another terminal as the error device by the command:

`@SET-ERROR-DEVICE <logical device number>`

Logical device numbers for logged in terminals are listed by the command @WHO-IS-ON. See appendix K for a list of logical device numbers. The following example shows how to route error messages to the terminal which have logical device number 9:

`@SET-ERROR-DEVICE 9`

In this manual the error device is always assumed to be the console. The error device will be set to the console by a cold start. The command below displays the logical device number of the current error device.

`@GET-ERROR-DEVICE`  
ERROR DEVICE 9

Before stopping a system, error messages should always be routed back to the console. Otherwise, there might be problems at the next warm start if the new error device is turned off.

All errors concerning hardware may be put on a segment. The command to initialize this segment is:

`@INITIALIZE-ERROR-LOG`

The contents of the error log is printed by the command:

`@PRINT-ERROR-LOG (<output file>)`

The default output file is your terminal. The above commands are only available to user SYSTEM.

---

## 9.2 NONFATAL ERRORS

---

This section describes the nonfatal error messages and what to do when they occur.

---

### 9.2.1 ERROR MESSAGES FROM CENTRAL PART OF SINTRAN III

---

At run time, errors may be detected by the system and written on the user's terminal or on the console. Most of the errors will cause the current RT program to be aborted. An error message as shown below will be displayed:

```
hh.mm.ss. ERROR nn IN rr AT ll ; tttt  
xx yy
```

If the error occurs in a background process, the error message will be written on the corresponding terminal. For other RT programs, the error message will be written to the console.

The parts of the error message have the following interpretation:

hh.mm.ss.	The time, ie., hours, minutes, seconds, when the error message was printed.
nn	Error number. For further explanation, refer to the list on the following pages.
rr	The RT program name or an octal address corresponding to an RT program name.
ll	Octal address where the error occurred.
tttt	Explanatory text.
xx, yy	Numbers carrying additional information about the error. One or both numbers may be omitted.

The following is an example of an error message from the RT program BAK03. BAK03 is the background process for the terminal with logical device number 34. Errors in system RT programs and in user RT programs give the same types of error message.

```
01.43.32 ERROR 14 IN BAK03 AT 114721; OUTSIDE SEGMENT BOUNDS
```

All system-included RT programs are automatically given names by the RT LOADER the first time the loader is called after a cold start. All error messages will then print the name of the RT program. See section 7.3 for an explanation of cold start.

User RT programs will usually not affect the SINTRAN III system. These error messages should only be reported to the owner of the RT program concerned.

---

### 9.2.2 SUMMARY OF RUN TIME ERROR MESSAGES

---

The following run time messages may occur. They are explained in detail in the next section. Note that some error codes are unused.

Error Code	Program Meaning	xx	yy	Aborted
00	Illegal monitor call			Yes
01	Bad RT program address			Yes
02	Wrong priority in PRIOR			Yes
03	Bad memory page	Page no.		
04	Internal interrupt on direct task level bit no.			No
06	Batch input error	Error no.		Yes
07	Batch output error	Error no.		
08	Batch system error	Error no.	L register	Yes
09	Illegal parameter in CLOCK			Yes
10	Illegal parameter in ABSET			Yes
11	Illegal parameter in UPDAT			Yes
12	Illegal time parameters			Yes
13	Page fault for non-demand			Yes
14	Outside segment bounds			Yes

Continues...

Error Code	Meaning	xx	yy	Program Aborted
15	Illegal segment number	Segment no.		Yes
16	Segment not loaded	Segment no.		Yes
17	Fixing demand	Segment no.		Yes
18	Too many fixed pages	Segment no.		Yes
19	Too big segment	Segment no.		Yes
20	Disk transfer error	Logical device no.	Unit °	No (abort if segm. transfer)
21	Disk transfer error	Last 16 bits of sector addr.	Hardware status	See ex- planation
22	False interrupt	Level		No
23	Device error	Hardware device no.	Hardware status	No
25	Already fixed	Segment no.		Yes
26	Mass storage time-out			No
27	Illegal parameter in CONCT			Yes
28	Space not available	Segment no.		Yes
29	MON 64 and MON 65	Error no.		Yes
30	Divide by zero			Yes
31	Permit violation			Yes
32	Ring violation			Yes
33	HDLC driver, fatal error			Yes
34	Illegal instruction			Yes
35	Reentr. Fortran stack error			Yes
36	Privileged instruction			Yes
37	IOX error	Address	Level	No

Continues...

Error Code	Meaning	xx	yy	Program Aborted
38	Memory Parity	PEA register	PES reg.	Yes
39	Memory out of range	PEA register	PES reg.	Yes
40	Power fail			No
41	Illegal error in ERMON			Yes
42	Overlapping segments	Segments		Yes
44	Corrected memory error	PEA register	PES reg.	No
45	Not demand segments			Yes
46	XMSG fatal error, internal error or inconsistency	XMSG error code	Physical address	Yes
47	XMSG user error	Calling level		Yes
48	False BEX interrupt			
49	Remote power fail interrupt			
50-69	User defined error (MON 142)	Error no.	Suberror no.	No
70	BEX parity error			
71	False MPM4 interrupt	Busc no.	Hardware status	No
72	MPM4 power fail interrupt	Busc no.		No
73	MPM4 memory out of range	Busc no.	Lower limit	No
74	MPM4 memory error	Local PES	Local PEA	No
75	MPM4 parity error	Busc no.	Lower lim.	No
76	MPM4 write parity error	Busc no.	Port code	No
90	Fortran run time error	Error no.		No
91	Fortran IO error	Error no.		No
100	Fortran library error			

---

### 9.2.3 DETAILED DESCRIPTION OF RUN TIME ERRORS

---

This section explains each run time error and the action to be taken by the operator. An output number is indicated by <nnn>. Blank lines in the messages are omitted in the description.

00	ILL. MONITOR CALL
----	-------------------

A monitor call, ie., a MON instruction, is incorrectly used, or has been given a wrong argument. In the case of RT programs which are background processes, error messages appear on the user's terminal.

System action: The specified RT program is aborted.

Operator's response: If it is a system RT program, list the RT description of the RT program and call ND for software service. If user RT program, report the error to the user.

01	ILL. RT PROG
----	--------------

An incorrect address of the RT description. Probably software error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

02	ILL. PRIORITY
----	---------------

An incorrect priority used in the monitor call PRIOR. The priority should be a number between 0 and 255. Probably software error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

04 INTERNAL INTERRUPT ON DIRECT TASK LEVEL  
LEVEL NO <nnn>  
BIT NO <nnn>

An interval internal interrupt from direct task level. Probably user error.

System action: Priority is given up.

06 BATCH INPUT ERROR  
FILE ERROR NO.: <nnn>  
<file system error message>

The batch processor has not been able to access the batch file for the current batch job. The user has possibly forgotten to include user name as a part of the batch file name, or the file access is illegal.

Operator's response: Report the error to the owner of the batch job.

07 BATCH OUTPUT ERROR  
FILE ERROR NO.: <nnn>

The batch processor has not been able to access the output file for a batch job. The file system error description explains the error. Otherwise as for error number 06, but for the output file.

Operator's response: Report the error to the owner of the batch job.

08 BATCH SYSTEM ERROR  
FILE ERROR NO.: <nnn>  
ADDRESS: <nnn>

An error has occurred in the batch system. It is probably a software error. It may possibly be a file system error, eg., a scratch file opened by another user, or the file may not exist.

Operator's response: Check if the batch system is properly initialized, and that a scratch file is available for the batch processor user. If it is, report the error to ND for software service.



09 ILL. PARAM. IN CLOCK

Incorrect array address used as argument in the monitor call CLOCK. Programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

10 ILL. PARAM. IN ABSET

Incorrect parameter used in the monitor call ABSET. Programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

11 ILL. PARAM. IN UPDAT

Incorrect parameter used in the monitor call UPDAT. Programming error or clock not updated after power off.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

12 ILL. TIME PARAM.

Incorrect parameter used in the monitor call TIME. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**13 PAGE FAULT FOR NON-DEMAND**

An RT program on a non-demand type segment tried to use an address outside the segment bounds. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

Example:

09.53.10 ERROR 13 IN PROGA AT 6157; PAGE FAULT FOR NON-DEMAND

**14 OUTSIDE SEGMENT BOUNDS**

An RT program on a demand segment tried to use an address outside the segment bounds. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**15 ILL. SEGMENT NUMBER  
SEGMENT NO. <nnn>**

Overlap of the address area or incorrect segment number used as parameters in the monitor calls MCALL or MEXIT. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

<b>16</b>	<b>SEGMENT NOT LOADED</b> <b>SEGMENT NO. &lt;nnn&gt;</b>
-----------	---

The segment in the monitor calls FIX or UNFIX is a demand segment, or the segment is not loaded by the RT LOADER. Probably user error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

<b>17</b>	<b>FIXING DEMAND</b> <b>SEGMENT NO. &lt;nnn&gt;</b>
-----------	--

Trying to fix a demand segment in physical memory. You may only fix non-demand segments.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

<b>18</b>	<b>TOO MANY FIXED PAGES</b> <b>SEGMENT NO. &lt;nnn&gt;</b>
-----------	---

Trying to fix more pages than allowed. Only a limited amount of physical memory may be used for fixed segments at the same time.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

<b>19</b>	<b>TOO BIG SEGMENT</b>
-----------	------------------------

A non-demand type segment is bigger than the SINTRAN III swapping area. Probably user error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

20	DISK TRANSFER ERROR HARDWARE DEVICE NO. <nnn> UNIT <nnn>
21	DISK TRANSFER ERROR DISK ADDRESS <nnn> STATUS <nnn>

These two error messages will always appear together and

indicate a mass storage transfer error, ie., a hardware error. Some common hardware device numbers are:

430	Line Printer 1	30	Terminal 33
500	Disk System 1	1540	Big disk system 1
510	Disk System 2	1560	Floppy Disk 1

The status code gives the type of error. See the SINTRAN III Reference Manual (ND-60.128), appendix G.

System action: In segment transport the RT program is aborted. No system action if it is an error return from a monitor call such as RFILE or WFILE.

Operator's response: An address mismatch has occurred if the status code is "nnn4nn". A parity error has occurred if the status is "nn1nnn". In that case you should first use the test program DIMS or DISC-TEMA to test for parity errors. If you find no errors, just continue. Otherwise, reformat the disk. Test for parity errors again. If errors still occur, contact ND for hardware service.

22	FALSE INTERRUPT LEVEL (DEC) <nnn>
----	--------------------------------------

Interrupt level 10, 11, 12, 13 or 14 was activated, but the IDENT instruction can not find a correct ident code for an input or output device. This is possibly noise or hardware error. The interrupt levels are used for:

- 10 Output interrupt.
- 11 Mass storage interrupt.
- 12 Input interrupt.
- 13 Real time clock.
- 14 Internal interrupt decoding.

Operator's response: Run the test program CONFIGURATION to check the state of the different device interfaces. If no errors are given, it is possibly due to noise. Call ND for hardware service if errors occur.

**23**     **DEVICE ERROR**  
         **HARDWARE DEVICE NO. <nnn>**  
         **STATUS <nnn>**

A device failure has occurred, probably a hardware error. The following hardware device numbers are used:

100- 157   Synchronous modem buffer card.  
200- 277   Asynchronous modem buffer card, terminal  
             buffer card.  
1200-1277   Asynchronous modem buffer card, terminal  
             buffer card.  
300- 377   Teletype buffer card, terminal buffer card.  
1300-1377   Teletype buffer card, terminal buffer card.  
400- 477   Paper tape devices, line printers, etc.  
500- 577   Mass storage devices.  
1500-1577   Mass storage devices.

The status word gives a more precise description of the error. See the status code description for the specified device.

Operator's response: If this error message occurs after a warm or cold start, and the printed status is equal to zero, then it is not necessarily an error. It may be due to:

- The device is connected to an RS-232 (V.24) interface, but not switched on.
- The device interface has been modified.

Report the error to ND. Call for hardware service only if the error occurs during normal running of SINTRAN III.

**25**     **ALREADY FIXED**  
         **SEGMENT NO. <nnn>**

Trying to fix a segment that is already fixed.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**26      MASS STORAGE TIME-OUT**

If the disk does not respond within a certain time interval, this message is printed. The time interval is 4 seconds.

Operator's response: Call ND for hardware service if the error recurs several times.

**27      ILL. PARAM. IN CONCT**

Incorrect parameters used in the monitor call CONCT. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**28      SPACE NOT AVAILABLE  
SEGMENT NO. <nnn>**

You try to load more information on a segment than is possible.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**29      FILE SYSTEM ERROR  
FILE ERROR NO. <nnn>  
<file system error description>**

The monitor calls ERMSG and QERMS are used to write a file system error message. An input parameter will hold the error code. These commands are normally used to write error messages in case of error exits from file system commands.

System action: ERMSG will give no system action, QERMS will abort the specified RT program.

Operator's response: Do the same as for error code 00.

**30      DIVIDE BY ZERO**

Division of a floating point number by zero is attempted.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**31      PERMIT VIOLATION**

The RT program attempted to access a page with an illegal reference type. This is probably programming error. Each individual segment may be protected against:

- Read access.
- Write access.
- Instructions fetch access.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**32      RING VIOLATION**

The RT program attempted to execute an instruction or access data on another page with a higher ring status. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**34      ILL. INSTR.**

The program attempted to execute an instruction that has not been implemented. Probably programming error.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**35 REENTRANT-FTN STACK ERROR**

A Fortran program attempted to use the stack outside the stack bounds.

Operator's response: Report the error to the user of the Fortran job. The user probably has to increase the stack size of the program.

**36 PRIV. INSTR.**

An RT program attempted to execute a privileged instruction.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

**37 IOX ERROR  
ADDRESS <nnn>  
LEVEL (DEC.) <nnn>**

This could mean either no answer from the external device, or an IDENT instruction is executed without an interrupt waiting. Probably a hardware error or user error, eg., a user trying to access a nonexistent device.

Operator's response: Run the test program CONFIGURATION INVESTIGATOR to check the state of the different devices' interfaces. Call ND for hardware service if the error recurs several times.

**38 MEMORY PARITY  
PEA <nnn>  
PES <nnn>**

A memory parity error has occurred. Probably hardware error. PEA gives the memory error address and PES gives the memory error status and bank number.

Operator's response: Report the error to ND and call for hardware service.



39      MEMORY OUT OF RANGE  
         PEA <nnn>  
         PES <nnn>

An attempt to address a nonexistent memory location has occurred. Probably hardware or system software error. PEA and PES are the same as for error code 38.

System action: The specified RT program is aborted.

Operator's response: Report the error to ND and call for hardware service.

40      POWER FAIL

This message will be given when SINTRAN III automatically restarts after a power failure or a @STOP-SYSTEM command.

Example:

14.21.32 ERROR 40 IN 23327 AT 110330: POWER FAIL

41      ILL. ERROR IN ERMON

Trying to use an error number outside the specified interval for ERMON.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

42      OVERLAPPING SEGMENTS  
         SEGMENTS <nnn> <nnn>

The two segments used by an RT program overlap in the virtual address space.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

44	<b>CORRECTED MEMORY ERROR</b> PEA <nnn> PES <nnn>
----	---

An error has occurred in memory and it has been corrected.

System action: None.

Operator's response: Note and call service if it occurs a second time.

45	<b>NOT DEMAND SEGMENTS</b>
----	----------------------------

A reentrant segment used by an RT program is overlapping a non-demand segment.

System action: The specified RT program is aborted.

Operator's response: Do the same as for error code 00.

46	<b>XMSG FATAL ERROR, INTERNAL ERROR OR INCONSISTENCY</b> XMSG ERROR CODE <nnn>
----	---

XMSG has crashed. A submessage will describe the error. See the SINTRAN III Communication Guide (ND-60.134).

System action: None.

Operator's response: Restart XMSG. Reload XMSG if the message occurs a second time.

47	<b>XMSG USER ERROR</b> CALLING LEVEL <nn>
----	--

Illegal use of XMSG. A submessage will describe the error. See the SINTRAN III Communication Guide (ND-60.134). The error may lead to a fatal XMSG error.

System action: None.

Operator's response: Make sure XMSG was generated with enough resources. Call ND for service if this is not the problem.

**48 FALSE BEX INTERRUPT**

Interrupt from the bus extender (BEX), but no error status set.

System action: None.

Operator's response: Note and call service if it occurs a second time.

**49 REMOTE POWER FAIL INTERRUPT**

Interrupt from the bus extender which has a special switch set.

System action: None.

Operator's response: Note and call service if it occurs a second time.

**70 BEX PARITY ERROR**

A bus extender parity error has occurred. Probably hardware error.

System action: None.

Operator's response: Report the error to ND and call for hardware service if the error occurs a second time.

**71 FALSE MPM4 INTERRUPT  
BUSE NO. <nnn>**

A false interrupt from one of the memory banks in MULTIPORT MEMORY 4.

System action: None.

Operator's response: Note and call service if it occurs a second time.

72	<b>MPM4 POWER FAIL INTERRUPT</b> <b>BUSC NO. &lt;nnn&gt;</b>
----	---

A false interrupt from one of the the memory banks in MULTIPORT MEMORY 4.

System action: None.

Operator's response: Note and call service if it occurs a second time.

73	<b>MPM4 MEMORY OUT OF RANGE</b> <b>BUSC NO. &lt;nnn&gt;</b>
----	--

An attempt to address a nonexistent memory location in MULTIPORT MEMORY 4 has occurred. Probably hardware or system software error.

System action: None.

Operator's response: Note and call service if it occurs a second time.

74	<b>MPM4 MEMORY ERROR</b> <b>LOCAL PES &lt;nnn&gt;</b>
----	--

Local MULTIPORT MEMORY 4 error. This error will be followed by error 73 or 75.

System action: None.

Operator's response: Call ND for hardware service if the error occurs a second time.

75	<b>MPM4 PARITY ERROR</b> <b>BUSC NO. &lt;nnn&gt;</b>
----	---

A MULTIPORT MEMORY 4 parity error has occurred. Probably hardware error.

System action: None.

Operator's response: Call ND for hardware service if the error occurs a second time.

76 MPM4 WRITE PARITY ERROR  
BUSC NO. <nnn>

A port discovered a parity error in MULTIPOINT MEMORY 4. Probably hardware error.

System action: None.

Operator's response: Call for hardware service if the error occurs a second time.

90 FORTRAN RUN TIME ERROR  
ERROR NO. <nnn>

A run time error has occurred in a Fortran program.

System action: None.

Operator's response: Report the error to the programmer.

91 FORTRAN IO ERROR  
ERROR NO. <nnn>

An input error has occurred in a Fortran program. The file system error description explains the error. This is a user program error.

System action: None.

Operator's response: Report the error to the programmer.

**TRANSF!!**

This special message is printed out when a transfer error occurs during a page swapping operation on the disk. Because the error message routine is on a segment, and the error has occurred when transferring a page to or from a segment, the simple message TRANSF!! is stored in memory and written out by a special routine which resides in memory.

System action: Background processes returns to command mode. Other RT programs are aborted.

Operator's response: This hardware error is probably due to either a disk error, disk controller error or memory error. Run the test program DIMS or DISC-TEMA to check for parity errors. Run MULTI to test the memory.

---

---

**9.3 FATAL ERRORS**

---

The system supervisor has the responsibility of starting the system if it has stopped because of a fatal error. In addition, he or she should try to identify the reason why the error occurred.

---

---

**9.3.1 MAIN GROUPS OF ERROR SYMPTOMS**

---

The machine has stopped if the yellow OPCOM lamp on the computer's panel is lit and the green RUN lamp is dark. If this is not the case, but there is no response from the machine, you have a hang situation. SINTRAN III may also function improperly.

---

---

**9.3.1.1 THE MACHINE HAS STOPPED**

---

SINTRAN III has a system routine called ERRFATAL. It may be called from many different routines if they discover fatal errors within internal data structures or other errors of such a serious nature that the system either cannot or should not continue to run.

The machine stops if it goes into ERRFATAL. To ascertain whether there is an ERRFATAL situation, the P register should be examined on the interrupt level which is active on the display panel. Use MOPC to dump the registers. See chapter 7. If the contents point to the ERRFATAL routine, ie., address 404 in SINTRAN III version I, this is the case.

The file system also has its own FATAL routine. If the system stops in this routine, the P register on interrupt level 1 will be 3573. Valuable information can then be obtained from the file system stack. This can be displayed as follows:

For SINTRAN III VSX systems:	For SINTRAN III VSE systems:
<u>OE</u> <u>26056&lt;62767</u>	<u>OE</u> <u>30056&lt;30767</u>

In both cases, the registers on all interrupt levels and the internal registers should be printed.

---

### 9.3.1.2 THE MACHINE HANGS

If the machine hangs, the RUN lamp's green light is on, but there is no response on the terminals. It may often be difficult to verify whether there is a real hang situation or perhaps just a heavily used machine.

There are many causes for a hang situation, and it is often difficult to find out what is wrong, because there are no error exits.

---

### 9.3.1.3 THE MACHINE DOES NOT FUNCTION PROPERLY

This includes all cases where SINTRAN III runs and commands may be executed, but the functions of commands or programs are not as expected.

---

## 9.3.2 COLLECTING DATA FOR ERROR IDENTIFICATION

When errors occur, the contents of registers and memory should be dumped on a printer or on a floppy disk.

---

### 9.3.2.1 WHAT SHOULD BE DONE BEFORE ERRORS OCCUR

SINTRAN III's own error log should be initialized, so that it will be possible to list hardware errors on a file. This is accomplished by the command @INITIALIZE-ERROR-LOG.

Take out a list on paper of all the RT programs. This is done through the command @LIST-RT-PROGRAMS. Always keep a MEMTOF floppy disk on hand before errors occur. MEMTOF will be explained in section 9.3.3.

---

### 9.3.2.2 WHAT TO DO WHEN AN ERROR HAS OCCURRED

---

If the machine has gone into a stop or a hang situation, the following should always be noted before pressing the buttons STOP or MCL:

- 1) Is the machine in a stop or a hang situation, ie., is the RUN lamp lit?
- 2) Are paging and interrupt turned on or off? Look at the computer's panel.
- 3) What interrupt levels are active? Look at the computer's panel.
- 4) Try to find out what was running on the machine when the error occurred. What programs were running on which terminals and which RT programs were running, eg., SIBAS, TPS or client programs. Also check whether COSMOS or ND-NET was in use. Include the peripheral programs that were in use. Especially important are DMA devices, such as magnetic tape, HDLC, MEGALINK, CAMAC, PIOC, GPIB, VERSATEC and ND-500.

When an overview of the situation has been obtained, one may start to dump the contents of registers and memory on the console as shown below.

- 
- 1) Press the STOP button on the computer's panel.
- 

- 2) Dump the CPU's internal registers by writing IRD on the console. Below is an example of such a register dump:

#### #IRD

```
000000 /163415 170041 156210 100005 153612 000000 020000 076033
000010 /000002 000001 021540 100000 000042 011775 177777 177777
```



The register names are:

PANS	STS	OPR	PGS	PVL	IIC	PID	PIE
CSE	ACTL	ALD	PES	PGC	PEA	Spare	Spare

The meaning of these names may be found in the ND-100 Reference Manual (ND-06.014).

---

3) Dump the register block for all interrupt levels by writing #Q<17RD. The order of the registers on each level are: Status, D, P, B, L, A, T and X. The main activities on the different interrupt levels are:

0	: Idle loop which only runs when the CPU is not used.
1	: Here all user programs and RT programs are executed, and also some SINTRAN III code.
2	: Not in use or direct tasks.
3	: Monitor level where most monitor calls are executed.
4	: INBT and OUTBT level.
5	: XMSG if in use.
6-9	: Not in use or direct tasks.
10	: Output drivers for terminals, line printers, etc.
11	: DMA drivers for disks and magnetic tape.
12	: Input drivers for terminals, PIOC, HDLC, etc.
13	: RT clock and HDLC level. Active every 20th millisecond.
14	: Internal interrupt decoding level.
15	: Not in use.

---

4) Dump areas of memory which might be of interest, for example, the first 400 locations of SINTRAN III RESIDENT. They contain many of the global parameters in SINTRAN III. This can be done by writing #Q<400.

---

5) Press MCL and dump the contents of physical memory to a floppy disk. Use the stand-alone program MEMTOF as described in the next section.

---

6) If you cannot find out more about the problem by yourself, and the machine must be operative rather quickly, you may try to start the machine again. If you suspect that a disk has been overwritten, run @TEST-DIRECTORY or the stand-alone program FILE SYSTEM INVESTIGATOR.

---

---

**9.3.3 USE OF THE MEMTOF DUMP**

---

MEMTOF is a stand-alone program to dump the contents of physical memory to floppy disks. Physical memory should be dumped if fatal errors occur. The dump is used to identify the cause of the error.

---

**9.3.3.1 HOW TO PREPARE AND USE MEMTOF**

---

Floppy disks with the MEMTOF program are delivered from ND. You may also prepare one yourself. Copy the files MEMTOF:BPUN and DUMPFL:BPUN from the SINTRAN III utility floppy disk to user SYSTEM. Then, log in as user SYSTEM and execute:

@LOAD-BINARY DUMPFL:BPUN

When you are asked for a peripheral file name for the floppy disk which should receive the dump, type FLOPPY-0. After the message "DUMPFL finished", take out the floppy disk and mark it MEMTOF.

How to take a MEMTOF memory dump after a fatal error:

- 1) Press the MCL button.
- 2) Put a MEMTOF floppy disk in floppy disk unit 0.
- 3) Write 1560& on the console. You will get the following output:

```
%%%% MEMTOF - dump MEMory TO Floppy V2.0 %%%%%
```

Replace the floppy disk with a new one. Then type any character to continue.

- 4) The program tells you what to do. You will get the following output when the program is finished:

```
%%%% MEMTOF dump finished %%%%%
```

- 5) You have now obtained a copy of memory banks 0 and 1 on the last floppy disk you put in. Mark the floppy disk MEMORY DUMP and include the date and the machine number.

Always include a description of the error symptoms and the output of the registers from before MCL was pressed. Also include a printout of the error log and the file SYMBOL-2-LIST:SYMB on a separate floppy disk if you can get the machine operative again. It is also a good idea to include the output from the @LIST-RT-PROGRAMS command and other possible error messages.

---

### 9.3.4 SOME EASILY DIAGNOSED ERROR SITUATIONS

---

This section shows some examples of error conditions which can be diagnosed easily.

---

#### 9.3.4.1 THE ERRFATAL ROUTINE

---

If the machine has gone into ERRFATAL on some interrupt level, you can always check the L register on that level, and see where the ERRFATAL routine was called. If you find that address in the SINTRAN III source listing part 1 or 2, you will often see why the ERRFATAL was called.

If the ERRFATAL occurred on level 14, it is often because there has been an internal interrupt on a level where this should not have occurred. For example, a PAGE FAULT on any other level than 1 or 4, or the errors MEMORY OUT OF RANGE or PROTECTION VIOLATION on the driver level.

To find out whether this has happened, you may check the following:

The contents of the B register on level 14 - 200 give an address called B14-200. This becomes a fixed address, which is 12016 in the I version of SINTRAN III. This and the next 4 locations in bank 0 are updated by SINTRAN III for each internal interrupt and given the following contents:

13650 : PERR	The P register on current active level when the internal interrupt occurred.
13651 : ACTLEV	The active interrupt level.
13652 : IBITNO	What kind of internal interrupt there was. See below.
13653 : PES	PES, if parity error or memory out of range.
13654 : PEA	PEA, if parity error or memory out of range.

IBITNO is the same as the bit number in the IIC register, ie.:

0 no internal interrupt.	6 PI : Privileged instruction.
1 MC : Monitor call.	7 IOX: IOX error.
2 PV : Protect violation.	10 PTY: Memory parity error.
3 PF : Page fault.	11 MOR: Memory out of range.
4 II : Illegal instruction.	12 POW: Power fail.
5 Z : Zero indicator set.	13 -15: Not in use.

---

#### 9.3.4.2 SINTRAN III WILL NOT START

---

If SINTRAN III goes into ERRFATAL on interrupt level 0 in the start-up phase, you may check in part two of the SINTRAN III source listing on the L register address on level 0. It may be because of patches done to SINTRAN III. Then the 7ENDC contents of location 141 is larger than 110000 (POFST).

The system will also stop if it does not have enough physical memory or it tries to put RT common outside physical memory. This will happen if DEFINE-RT-COMMON-SIZE in the SINTRAN SERVICE PROGRAM has been given to change the size of RT common, and then the memory size later has been reduced. There are also other possibilities during start-up. Be especially aware of parity errors on memory and disk.

If the character # does not appear on the console after LOAD has been pressed, it is possible that the disk is not ready or that there is another hardware error. Run test programs.

---

#### 9.3.4.3 MEMORY OVER WRITING

---

Memory being overwriting may cause a system to stop, hang or function improperly. It is often difficult to see if overwritings have occurred because it might have happened anywhere in memory, and one does not really know what should have been there. It may often be accidental that you find one or more locations that have been over written.

Some severe cases might be easy to recognize. If large parts of bank 0, that is, SINTRAN III RESIDENT, have values like 0 or 177777 or any other special pattern, this might be a reason to be alert. You may also check that locations 0 and 1 always contain 124000. The first 300 addresses of the SINTRAN III listing may also be checked. Here SINTRAN III has some pointers, flags and other variables one should be familiar with.

Support personell can locate overwritten memory easier after a MEMTOF dump. CPU, memory management, and all DMA devices such as disks should be checked with test programs when "unexplainable" errors arise.

---

#### **9.3.4.4 GENERAL INFORMATION ABOUT HANG SITUATIONS**

---

At every hang situations you can try to find out what takes place on the lowest active interrupt level. If the P register on any level contains 0 or 1, the machine will loop on this level. The 2 first locations contain instructions to jump to themselves.

At every hang situation, also check location 10 in virtual memory. It contains the address of the RT description of the active program. See the @LIST-RT-PROGRAMS list. Check that the high priority given to RT programs is not too high.

Otherwise situations similar to hanging might arise if one tries to access disks which are not ready, when the WRITE PROTECT button is pressed, or when there is a serious error on the disk. This can be verified by checking parity with the DISC-TEMA test program.

---

#### **9.4 TERMINALS THAT DO NOT RESPOND**

---

The system may function properly except for one terminal that does not respond. A user will, for example, experience this when she or he executes a program that runs in an endless loop with the escape function disabled.

You should handle the situation as shown below. First, you ought to give CTRL Q to make sure that the terminal is not stopped because CTRL S has been given. The line/local setting of the terminal should also be checked. Most terminals have a light to indicate that they are on-line.

**HANGING TERMINALS**

- 1) Log in as user SYSTEM.
- 2) Give the command @STOP-TERMINAL with the logical device number of the terminal as its parameter. This will log out the terminal. The command @WHO-IS-ON can be used find the logical device number.
- 3) Try to log in from the terminal. If it still does not respond, give the command @LIST-DEVICE with the logical device number of the terminal as the first parameter. Specify 0 as the second parameter. This will output the name of the background process that reserved the terminal, eg., BAK35.
- 4) Give the command @ABORT with the name of the background process as parameter, eg., @ABORT BAK36.
- 5) Give the command @RT with the name of the background process as parameter to restart the process, eg., RT BAK35.

Alternatively, you may restart the system.

The escape character is the character which is used before logging in and to stop executing programs and commands. The character which has this function can be defined for individual terminals by the command @DEFINE-ESCAPE-CHARACTER.

---

**9.5 TELEFIX**

---

ND has a product called TELEFIX, which provides remote diagnostics and maintenance services for its customers. TELEFIX offers the possibility of tracing and correcting faults in a machine some distance away via the existing telecommunication network.

A remote diagnosis system is especially suited to solve software problems. Using TELEFIX, diagnostic tests can be run remotely. With it, the system supervisor will be alleviated of performing many of the technical operations.

For additional security, TELEFIX provides the possibility of logging all the diagnosis and maintenance operations performed by the service center on the terminal at the customer site.

The TELEFIX system consists of several software packages which reside in the host computer. TELEFIX includes a floppy disk, the "TELEFIX floppy disk" and a device, the "TELEFIX adaptor". In addition, the customer who has TELEFIX installed must supply a telephone modem and a line to call the nearest TELEFIX center.

The TELEFIX floppy disk contains the necessary test programs and special file transfer programs. The TELEFIX adaptor is directly connected to the CPU board in the customer's computer. The TELEFIX adaptor is equipped with a switch which has three different positions:

- Normal
- Remote
- Remote listen

The normal position is used when the customer's computer operates normally. It is then impossible for the TELEFIX center to access that computer.

Setting the switch in remote position enables the TELEFIX center to access, operate and control the computer at the customer's site. There can be no communication in the opposite direction. All the instructions given by the center's computer and the output from the diagnosed computer will be written on the customer's console.

The third position, remote listen, enables both the service technician and the local system supervisor to input instructions during the diagnosis operation, eg., repeat the sequence of commands that lead to failure. In this case, proper communication is established between the customer and the TELEFIX center.

SUMMARY OF THE REMOTE DIAGNOSIS PROCEDURE PROVIDED BY TELEFIX
<ul style="list-style-type: none"><li>- In case of a failure, you may call the local ND service centre.</li><li>- The ND local support technician decides to use remote diagnosis to solve the problem.</li><li>- You are requested to set the TELEFIX adaptor switch on the remote position, insert the TELEFIX floppy disk into the floppy disk drive, and call the special TELEFIX center number.</li><li>- The TELEFIX operator starts to diagnose the problem.</li></ul>





---

## 10 PREVENTIVE MAINTENANCE AND TEST PROGRAMS

---

This chapter contains information about what kind of maintenance the system supervisor should do, and a summary of available test programs. The most commonly used test programs are explained. Some SINTRAN III commands related to error testing are described.

There are test and verification programs to test the file system, memory, CPU, mass storage and input/output devices. Detailed descriptions of these programs are found in the manuals Test Program Descriptions for ND-100 (ND-30.005) and Test Program Descriptions for ND-500 (ND-30.018).

---

### 10.1 MAINTENANCE TO BE DONE BY THE OWNER

---

The system supervisor should perform the following kinds of maintenance:

#### DAILY:

- Site should be kept clean and dustfree.
- Do not perform any kind of operation inside the computer without permission from ND.
- Never turn the power off during nights, weekends, etc.

#### MONTHLY:

- Clean air filters.

#### ANNUALLY:

- Clean the computer. You may clean the boards, gold contacts and memory modules with isopropanol. A vacuum cleaner may be of use.
- Check power supply output.

---

## 10.2 LOADING TEST PROGRAMS

---

The test programs are stand-alone programs delivered on floppy disks. The floppy disk contains a floppy disk monitor which offers commands to handle the test programs. The test programs are loaded and started as shown below:

### STARTING STAND-ALONE PROGRAMS

- Press the STOP and MCL buttons on the computer's panel.
- Put the floppy disk with the stand-alone program in the floppy disk unit 0.
- Type 1560& on the console to load the floppy disk monitor.
- When the floppy disk monitor is started up, it types an asterisk (\*). If only one stand-alone program exists on the floppy disk, it will be started.
- If more than one stand-alone program exists, type the name of the one you want to start. Default file type is :BPUN.

The following are the most important commands in the floppy disk monitor:

#### \*HELP

List the title and the version of the floppy disk monitor and the available commands.

#### \*LOAD-FILE <file name>

The floppy disk monitor will load the specified file as a stand-alone program. If only one bpun format file is present on the floppy disk, this will be automatically loaded after typing 1560&. This command is the default command, ie., typing only the file name will have the same effect.

**\*LIST-FILES <file name>**

List the files on the floppy disk, ie., the stand-alone programs.

**\*OPCOM**

Exit from the floppy disk monitor and enter MOPC. The character # will be output.

**\*PLACE-FILE <file name>**

The file will be read into memory. The floppy disk monitor types PLACED on the console. It then stops the computer and enters MOPC. This is useful if you want to patch the program before it starts. The program will start when "!" is typed.

---

### 10.3 THE FILE SYSTEM INVESTIGATOR

---

This stand-alone program can be used to check the consistency of a directory, ie., that its structure is correct, pointers to structures within the directory are correct, files do not conflict, the bit file is correct, the number of pages used is correct and user and object entries are correct.

The program only reads from the disk. It diagnoses errors, but does not attempt to correct them. The recommended procedure before testing a directory with the FILE SYSTEM INVESTIGATOR is:

- 1) Log in as user SYSTEM on one terminal. Then use the command @SET-UNAVAILABLE.
- 2) Check that no users are logged in.
- 3) Close the scratch file with the @CLOSE-FILE 100 command.
- 4) Stop SINTRAN III by pressing the MCL and STOP buttons.
- 5) Take backup of the directory.

Backup is described in chapter 13.

---

**10.3.1 DIRECTORY CONSISTENCY CHECK**

---

It is advisable to use the FILE SYSTEM INVESTIGATOR regularly to check directories. You should definitely use it before taking backup of a directory. The check takes only 5 minutes on a 75 Mbyte disk. To check that a directory is free from errors, three FILE SYSTEM INVESTIGATOR commands are needed:

```
#1560&
*FILESYS-INV:BPUN
...
DEVICE NAME: DISC-75MB-1
UNIT: Q
...
>DUMP-DIRECTORY-ENTRY
...
>LIST-USERS
...
>PAGE-LIST
...
```

Load the floppy disk monitor.  
Start FILE SYSTEM INVESTIGATOR.  
(output omitted in this example.)

---

The three commands  
needed to check the  
consistency of a directory.

---

If E is answered to the question output when using the command PAGE-LIST, only errors will be listed. The output will describe the type of errors and where they are located.

The output may be directed to a line printer by using the command SET-PRINTER-DEVICE-NUMBER. The physical device number for the line printer is usually 430. Table 18 shows the commands available in the FILE SYSTEM INVESTIGATOR.

DUMP-BIT-FILE	LIST-FILE-NAMES
DUMP-DIRECTORY-ENTRY	LIST-PAGE-NUMBERS
DUMP-FILE	LIST-USERS
DUMP-OBJECT-ENTRY	PAGE-LIST
DUMP-PAGE	REDEFINE-DISC-TYPE
DUMP-USER-ENTRY	SET-PRINTER-DEVICE-NUMBER
HELP	

Table 18. Commands in the FILE SYSTEM INVESTIGATOR

### 10.3.2 CORRECTING ERRORS

If a directory has serious errors, eg., page conflicts between files, you may attempt to correct them. Take a backup of the directory and try to correct the errors by using the following commands:

Dump commands	Change commands
@DUMP-DIRECTORY-ENTRY	@CHANGE-DIRECTORY-ENTRY
@DUMP-PAGE	@CHANGE-PAGE
@DUMP-USER-ENTRY	@CHANGE-USER-ENTRY
@DUMP-OBJECT-ENTRY	@CHANGE-OBJECT-ENTRY
@DUMP-BIT-FILE	@CHANGE-BIT-FILE

To look at the contents of a location within a page, entry or block, type its address relative to the start of the page, entry or block, followed by a /. The contents are then output.

The contents may then be changed by typing in a new value, followed by carriage return. Then the contents of the next location is output. If no new value is entered before carriage return, the old value is left unchanged. If an asterisk (\*) is typed, the current address is given. To terminate the command and return to SINTRAN III, type a period (.).

All addresses are relative to the start of the page, entry or block. They have the range shown in table 19. The contents of the locations in the pages, entries and blocks are described in appendix F.

DIRECTORY STRUCTURE	RANGE
User Entry:	0 < address < 37
Object Entry:	0 < address < 37 <sub>8</sub>
Directory Entry:	0 < address < 17 <sub>8</sub>
Bit file block:	0 < address < 17 <sub>8</sub>
Page:	0 < address < 1777 <sub>8</sub>

Table 19. Addresses in a Directory

Alternatively, by copying selected pages from a previous backup copy to the disk using the stand-alone program DISC-TEMA, pages containing erroneous pointers, user entries etc., can sometimes be corrected. This requires thorough knowledge of the file system.

Users with insufficient experience with the file system can use the SINTRAN III commands below when they get BIT FILE ERROR, PAGES MARKED AS OCCUPIED BUT THEY ARE NOT USED BY ANYONE or ERROR IN PAGES USED. Do not use them with other errors. Set the system unavailable, log out all users, and abort all batch jobs before using the commands.

```
@TEST-DIRECTORY <directory name>  
@REGENERATE-DIRECTORY <directory name>
```

The command @TEST-DIRECTORY rebuilds the bit file and resets the number of pages used, corrects open and current enter counts and other minor errors. It also reports major errors. Always execute @TEST-DIRECTORY before @REGENERATE-DIRECTORY.

The command @REGENERATE-DIRECTORY performs the same functions. It also deletes all, but one of the conflicting files and generally tidies up a directory so that its structure is correct. The two commands normally do not take more than 15 minutes on a 75 Mbyte disk. If possible, errors should be corrected by the CHANGE-commands mentioned above instead of by @REGENERATE-DIRECTORY.

**WARNINGS!**

- 1) A lot of information may be lost in attempts to correct a directory. It is essential to take backup first. If you use the command @REGENERATE-DIRECTORY, at least parts of any conflicting files can be recovered. Backup is described in chapter 13.
- 2) The commands @TEST-DIRECTORY and @REGENERATE-DIRECTORY should not be interrupted by pressing escape since they clear and rebuild the bit file.

---

### 10.3.3 FILE SYSTEM INVESTIGATOR EXAMPLE

---

This section shows an example of loading and using the FILE SYSTEM INVESTIGATOR. First you have to stop SINTRAN III. Use the MCL and STOP buttons. Then you should insert the floppy disk with the stand-alone program FILESYS-INV:BPUN in floppy disk unit 0. Type #1560 on the console as shown below:

#1560&

\*LOAD-FILE FILESYS-INV:BPUN

FILE SYSTEM INVESTIGATOR

SUT 2135K

ISSUED OCT 5, 1983

DEVICE NAME : : DISC-75MB-1  
DISC-75MB-1

DEVICE UNIT : 0

DIRECTORIES CREATED BY SINTRAN VERSION E OR LATER ARE  
A FEW PAGES SMALLER THAN THOSE CREATED BY VERSIONS A, B, C, D.  
IF YOU WANT THE SMALLER SIZE, TYPE Y, ELSE N: Y

TOTAL NO. OF PAGES IS 110121

THE COMMAND HELP GIVES YOU A LIST OF THE COMMANDS

>HELP

HELP

COMMAND :

MONITOR COMMANDS:

\*\*\*\*\*

EXPLAIN COMMANDS

LIST-SPECIAL-COMMANDS

HELP

TERMINAL-MODE

STOP-SYSTEM

SET-CONSOLE-DEVICE-NUMBER

OPCOM

MESSAGE

PROGRAM-PURPOSE

GO-TO-ADDRESS

PRINT-NOTE

FLOPPY-LOAD

SET-PRINTER-DEVICE-NUMBER

Comment:

Loads the FILE SYSTEM INVESTIGATOR from the floppy disk. The commands are given from the console while the system is stopped. The directory on unit 0 of the device DISC-75MB-1 is investigated.

Comment:

Lists all commands. The commands are divided in two classes. Only the program commands affects the directory investigation.

STOP-CLOCK  
 START-CLOCK  
 UPDAT  
 DATCL  
 DUMP-PRINTER-BUFFER  
 CLEAR-PRINTER-BUFFER  
 PROGRAM-STATUS  
 EXIT

PROGRAM-COMMANDS  
 \*\*\*\*\*

DUMP-BIT-FILE  
 DUMP-DIRECTORY-ENTRY  
 DUMP-FILE  
 DUMP-OBJECT-ENTRY  
 DUMP-PAGE  
 DUMP-USER-ENTRY  
 LIST-FILE-NAMES  
 LIST-USERS  
 LIST-PAGE-NUMBERS  
 PAGE-LIST  
 REDEFINE-DISC-TYPE

>DUMP-DIRECTORY-ENTRY  
 DUMP-DIRECTORY-ENTRY

Comment: Dumps the directory entry of PACK-ONE.
---

DIRECTORY ENTRY DUMP:

050101	041513	026517	047105	023400	000000	000000	000000	
								PACK-ONE
040000	044374							OBJECT FILE POINTER
040000	044376							USER FILE POINTER
000000	044050							BIT FILE POINTER
000000	063505							PAGES NOT RESERVED

>LIST-USERS  
 LIST-USERS

USER LIST:

000 SYSTEM  
 ENTER COUNT IS NOT ZERO

001 SCRATCH  
 002 RT  
 003 FLOPPY-USER

PAGES RESERVED ON THIS DIRECTORY (ACC. FROM USER ENTRIES):  
 024404



>PAGE-LIST  
PAGE-LIST

THE WHOLE PAGE LIST (W), PART OF IT (P), OR ONLY ERROR  
MESSAGES (E): E

FETCHING OF ALL PAGE NUMBERS IS STARTED

WARNING: CURRENT OPEN COUNT IS NOT ZERO. THE FILE IS OPEN FOR  
READ. USER AND FILE NOS.: 000000 000025

WARNING: CURRENT OPEN COUNT IS NOT ZERO. THE FILE IS OPEN FOR  
READ. USER AND FILE NOS.: 000001 000000

SORTING IS STARTED

PAGE LIST AND BIT FILE CHECK

CHECK FOR UNUSED, BUT OCCUPIED PAGES IS STARTED

BIT FILE ERROR (OR BAD PAGES, SET BY @CREATE-DIRECTORY).  
PAGES ARE MARKED AS OCCUPIED IN THE BIT FILE,  
BUT THEY ARE NOT USED BY ANYONE. PAGE NOS.:

075303

>DUMP-OBJECT-ENTRY  
DUMP-OBJECT-ENTRY

USER NO. (0-377): 0  
FILE NO. (0-377): 25

Comment:

Inspects file number 25 of  
user 0 further because of  
the listed irregularities.

USER 000 OBJECT ENTRY 025 DUMP:

110001	U/W/R/M TERM. NO OF RESERVING USER
044105 047124 026515 047504 042447 000000 000000 000000	HENT-MODE
051531 046502	TYPE ( SYMB )
000025 000025	POINTERS TO NEXT AND PREVIOUS VERSION
002377	ACCESS WORD
000010	TEMP/L/M/A/C/I/S/P/T
000000	DEVICE NUMBER
000000	MAIN DIR INDEX / USER INDEX OF RESERVING USER
000025	OBJECT INDEX OF THIS ENTRY
000001 000004	CURRENT AND TOTAL OPEN COUNT
100131 005514	DATE CREATED ( 1983.12.12 16.45.12 )
076133 000203	LAST DATE OPENED FOR READ ( 1983.12.13 16.02.03 )
100131 005514	LAST DATE OPENED FOR WRITE ( 1983.12.12 16.45.12 )
000000 000001	PAGES IN FILE
000000 003015	MAX. BYTE POINTER
040000 044765	FILE POINTER

&gt;DUMP-USER-ENTRY

DUMP-USER-ENTRY

USER NO. (0-0377): 0

USER 000 ENTRY DUMP:

```

100401          ENTER COUNT ETC.
051531 051524 042515 023400 000000 000000 000000 000000 SYSTEM
000000          PASSWORD
100130 175022    DATE CREATED ( 1983.11.12   15.40.18 )
076132 176727    LAST DATE ENTERED ( 1983.11.13   15.55.23 )
000000 023420    PAGES RESERVED
000000 012625    PAGES USED
000000          USER INDEX
000000          MAIL FLAG
002377          USER DEFAULT FILE ACCESS
000000 000000 000000    FREE SPACE
103402 000000 000000 000000 000000 000000 000000 000000
                                           FRIEND TABLE

```

Comment:

Inspects the user entry of  
user 0.

The above example demonstrates that for this case the enter count for SYSTEM is not 0. File number 25 for user 0 has a non-zero open count, so this file is examined closer by the command DUMP-OBJECT-ENTRY.

---

#### 10.3.4 ERRORS IN A DIRECTORY ENTRY

---

Most such errors can be corrected by using SINTRAN III commands. However, if the errors are on a main directory, it may be impossible to enter it. In this case one can use the command CHANGE in the stand-alone program DISC-TEMA. See section 10.6.

If the directory entry is destroyed, it may be restored if the user has a previous dump from the FILE SYSTEM INVESTIGATOR. Use the DISC-TEMA command TRANSLATE to translate the logical disk address (1 for most disks, 7 for HAWK disks), and then use the command CHANGE to restore the directory entry. The entry is located in addresses 760 to 777 (160 to 177 for HAWK disks) and consists of the 16 octal numbers dumped by the FILE SYSTEM INVESTIGATOR command DUMP-DIRECTORY-ENTRY.

The directory entry may also be restored by copying back 2 sectors (10 for HAWK disks) from a recent backup copy. Use logical address 0 and<sup>8</sup> set the directory mode off (important!) by the DISC-TEMA command MODE. Then start COPY with amount-2 (10 for HAWK disk).

If the error is not on the system disk pack, then an error in a directory entry can be corrected by the SINTRAN III command @CHANGE-DIRECTORY-ENTRY. In this case, the directory entry is always located in address 0-17<sup>8</sup>.

8

---

---

### **10.3.5 ERRORS IN THE USER FILES**

---

If a user has disappeared and apparently also that user's files, the best thing to do is to create that user again by the SINTRAN III command @CREATE-USER. It is important that the user gets the same number as before, ie., the user index as found from @DUMP-USER-ENTRY, since this number also identifies the files belonging to that user.

If the user name and probably the whole user entry is destroyed, use the command @CHANGE-USER-ENTRY, and put a zero in the first word of the user entry. This will remove the user. Then create that user again. Repeat this procedure if more user entries need correcting. Then give @REGENERATE-DIRECTORY.

---

---

### **10.3.6 ERRORS IN THE BIT FILE**

---

If a directory has a page conflict, one of the conflicting pages must be removed. This is best done by using @DELETE-FILE followed by @REGENERATE-DIRECTORY. If @DELETE-FILE gives an error message, the conflicting files may be copied individually to recover them.

If an error message tells you that there are pages in the bit file that are occupied, but not used by anyone, use @TEST-DIRECTORY. If the error in the bit file results from a destroyed index block, remove the file by the command @CHANGE-OBJECT-ENTRY. This is done by putting 0 in word 0 of the object entry. The command @TEST-DIRECTORY may be used afterwards. To be safe, run the FILE SYSTEM INVESTIGATOR once more.

---

---

### **10.3.7 MORE THAN ONE VERSION OF A FILE OR NO VERSION NUMBER 1**

---

If a non-spooling file has a version number other than 1, it means that the pointers to the previous or the next version are wrong. Use @CHANGE-OBJECT-ENTRY to correct them. The version contains the user number in the left byte and file number in the right byte.

---

**10.4 CPU TEST PROGRAMS**

---

The following test programs can be used to check the CPU:

INSTRUCTION	These programs test the instruction set of the ND-100 on different levels while taxing the microprogram MOPC.
PFAIL	The program tests that the power failure system will restart the computer.
RTC-12	The program tests the real time clock.

---

**10.5 MEMORY TEST PROGRAMS**

---

The following test programs can be used to test physical memory:

100-ERROR	Each memory location has 6 <u>parity bits</u> to check and correct bit errors in read, write or fetch operations. This program tests the error correction in the memory locations of the ND-100.
CACHE	Tests the <u>cache</u> system on the memory management card.
MEMORY	This program tests all available physical memory in ND-100 and the multiport memory.
MOVER	This program tests memory by moving itself to a new place in memory each time a test cycle is completed. This program can be run under SINTRAN III.
MULTI	Memory test program that also activates DMA devices to tax the system. DMA means direct memory access. DMA devices are used to transfer pages between physical memory and mass storage.
PAGING	Tests the paging part of the memory management system.

---

## 10.6 INPUT AND OUTPUT DEVICE TEST PROGRAMS

---

The following test programs can be used for input and output devices:

DIMS	A test and utility program for 10 Mbyte disks.
TECOD	Program that tests physical memory and 10 Mbyte disks.;
SMALL-RAND	Random test on 10 Mbyte disk. The program writes and reads randomly on the disk pack. This means that the contents of the pack will be destroyed and a scratch pack is needed.;
DISC-TEMA	This is the main test program for all disk types except 10 Mbyte.
SUPER-RAND	Random test for most disk types. The program writes and reads randomly on the disk pack. This means that the contents of the pack will be destroyed, and a scratch pack is needed.
WINCH-RAND-A	Random test for 8 and 5 inch disks. The program writes and reads randomly on the disk pack. This means that the contents of the disk pack will be destroyed, and a scratch pack is needed.
FLOPP-FORM	Used for stand-alone floppy disk formatting and for checking controller, drive and formatter functions during formatting.
FLOPPY-FU	Test floppy disk interface and functions.
FLOPPY-RAN	The program tests the floppy disk by writing and reading random data at random addresses.
STC-RUNNER	This program is used for manual operations of the STC tape subsystem.
STC-TEST	This program tests the STC magnetic tape controller.
TANB-MAG	Test program for Pertec and Tandberg magnetic tape controllers.
TET2200	Test of the Tandberg TDV-2200 for Nortext systems. The program runs under SINTRAN III.

TNCT	This program tests the Nordcom Color Terminal.
LP-TEST	Test of line printers and interfaces. The program runs under SINTRAN III.
PARAL-BYTE	Test of the parallel byte interface ND-653.
VERSATEST	Test for Versatec printer/plotter.
HDLIC-MEGALINK	Test program for HDLC and MEGALINK interfaces. It is also possible to test the connection between two machines.
SYNC-MODEM	Test of synchronous modem interfaces and communication between two machines.
UNIVERSAL-DMA	Test of the Universal DMA interface and channels to multiport memory by using Universal DMA.

---

### **10.7 TEST PROGRAMS TO RUN WHEN THE SYSTEM HAS STOPPED**

---

Some tests are performed when SINTRAN III has stopped. Physical memory and the configuration should be checked.

---

#### **10.7.1 MEMORY TEST USING THE MICROPROGRAM**

---

The computer has a microprogrammed memory test for physical memory. This is a useful feature as it may quickly be decided whether a given error is caused by the CPU or the memory. To use the test program, do the following:

- Press the STOP and MCL buttons to enter MOPC.
- Give lower memory address to be tested, B/
- Give upper memory address, X/ (32K = 077777, 64K = 177777)
- Start the memory test program by typing #.

If the test is successful, # is typed when finished. If the test is unsuccessful, a question mark will be printed on the console, and the test stops. The following registers on interrupt level 0 give information about the error:

P register: Failing address.  
L register: Test pattern.  
D register: Error pattern.  
T register: Failing bits.  
B register: Start address.  
X register: Stop address.

---

### 10.7.2 CONFIGURATION INVESTIGATION

---

Besides individual control of each device in a computer installation, it is necessary to know the connection between each device interface and the CPU. This can be done by using the stand-alone program CONFIGURATION.

SINTRAN III uses physical device numbers to access devices. Each device has a unique physical device number. When a device is ready for service, its interface issues an interrupt to SINTRAN III.

The ident code is used to identify the device that caused the interrupt. All input and output devices are connected to interrupt level 10, 11, 12 or 13. The physical and logical device numbers and ident codes are selected automatically when a terminal group is chosen. See Appendix K.

The output gives information about the machine's memory size, ALD setting, physical and logical device numbers, ident codes, and ident code priority sequence of levels 10, 11, 12 and 13, and whether cache memory is installed.





---

## 11 PATCHING

---

To patch is to change the contents of locations in physical memory or in mass storage. SINTRAN III can be modified and errors can be corrected by patching.

SINTRAN III should be patched to correct errors the very first time it is loaded. A mode file called PATCHFILE:SYMB is used for this purpose. It is delivered to customers on a floppy disk together with SINTRAN III.

The symbol list, ie., the file with a list of all system variables and their addresses in SINTRAN III, is also on the same floppy disk. Customers get this symbol list when they order SINTRAN III. Two additional, separate SINTRAN III source code listings are available. They can be ordered separately.

PATCHFILE:SYMB contains all the known error and modification patches. Before starting this mode file, make certain that the subsystem FMAC:PROG and DMAC:BPUN exist. List the patch file on a printer. Check if 7ENDC is less than an address specified in the file. See section 10.5. The execution of the mode file PATCHFILE:SYMB is done in a cold start.

Modification patches, eg., to set the speed of the terminals, are not included in PATCHFILE:SYMB. Such patches must be run separately. You ought to use another mode file, eg., MODIF-PATCHES:SYMB, for this purpose.

Each month, the ND Bulletin should be checked for changes and modifications that apply to the installed version of SINTRAN III. In addition to Software System Report information, the Error Information List is of special interest to the system supervisor.

The installed version of SINTRAN III should be promptly modified, and PATCHFILE:SYMB and MODIF-PATCHES:SYMB updated with the new patches listed in the Software System Report and the Error Information List. By listing PATCHFILE:SYMB and MODIF-PATCHES:SYMB, the system supervisor can see which patches have been added to the system.

Different patching methods will be explained in this chapter. The key terms that are used in these explanations are defined below.

**Patch**

A program modification made by changing the octal code, rather than by changing the source code before compilation.

**SINTRAN III RESIDENT**

The part of SINTRAN III which always remains in physical memory, ie., that part which is not swapped in and out of physical memory.

**PAGING OFF AREA**

A part of SINTRAN III which always remains in physical memory. This part is only accessed with the memory management system turned off, ie., with paging off. SINTRAN III RESIDENT consists of pages on page table 0. The PAGING OFF AREA pages, however, are not on any page table. The PAGING OFF AREA is abbreviated POF.

**IMAGE AREA**

Refers to the disk copy, on SEGFILO:DATA, of SINTRAN III RESIDENT and the PAGING OFF AREA. This copy is fetched to physical memory each time MCL and LOAD is pressed.

**Segment**

These are the swapped parts of SINTRAN III. For example, segment 3 is the command segment, segment 4 is the RT LOADER segment, segment 11 is the error log segment, segment 6 is the file system segment, segment 7 is the DMAC segment, segment 10 is the RTFIL segment, and segment 14 is the error program segment.

**SAVE AREA**

The general name given to those areas of the disk containing code and data for SINTRAN III, and from which the current IMAGE AREA and segments are created. The code here is never executed directly. The )HENT and @COLD-START commands copy the code and data from the SAVE AREA to the IMAGE AREA and the segments.

The system file SINTRAN:DATA is the SAVE AREA for SINTRAN III RESIDENT, the system segments, and the PAGING OFF AREA. The file MACM-AREA:DATA is the SAVE AREA for the command segment, the ND-NET file copy program, and the file system and error program segment.

The parts of SINTRAN III that are not on SINTRAN:DATA or MACM-AREA:DATA are saved on the segments. The RT LOADER is saved on segment 13, ie., the RT LOADER on segment 4 is initially a copy of segment 13.

From the above definition of SAVE AREA, it should be noted that any modification made to the SAVE AREA will remain until SINTRAN III is reloaded from floppy disks. Modifications to the IMAGE AREA of the segments will remain until a new cold start is performed. Any modification made to SINTRAN III RESIDENT or the PAGING OFF AREA lasts until a warm start is performed.

SINTRAN III can be patched in the following ways:

- The command @LOOK-AT.
- The subsystem FMAC.
- The command DMAC.
- The subcommands of the SINTRAN SERVICE PROGRAM.

SINTRAN III can also be modified off-line during a cold start procedure.

Note concerning addresses:

Addressing the different areas mentioned above is straightforward except for the patching of the PAGING OFF AREA on MACM-AREA:DATA. The code begins physically at an installation dependent address in excess of 100000. Hence, to convert from a physical address, eg., from a listing to a file address, one must subtract 100000<sub>8</sub> from the actual address as seen in the symbol list.

---

### 11.1 SINTRAN III COMMANDS FOR PATCHING

---

The command below can be used to examine and modify memory locations and registers on the ND-100:

@LOOK-AT <area>[,<additional parameters>]

The parameter <area> describes where the examination or modification should take place. The following alternatives are available:

MEMORY	A user's virtual memory space mapped via the normal page table. Maximum 64 Kwords.
ALT-MEMORY	A user's virtual memory space mapped via the alternative page table. Maximum 64 Kwords. Only available on two-bank background segments.
SEGMENT	A segment on mass storage and on memory if the segment is there at the time.
RTCOMMON	Locations of the common area for RT programs.
IMAGE	Locations on the IMAGE AREA containing a disk copy of SINTRAN III RESIDENT.
RESIDENT	SINTRAN III RESIDENT, ie., the locations in the lowest 64 Kwords of physical memory.
REGISTERS	Registers on each interrupt level.
ALT-SEGMENT	The part of background segment which can be accessed by the alternative page table.

The format of the alternative parameters are <octal address>/ <octal contents> <new contents>. Type a period to terminate the command. The SINTRAN III Reference Manual (ND-60.128) documents the use of the alternative parameters in detail.

## 11.2 PATCHING WITH FMAC

FMAC is a subsystem delivered to all customers on a floppy disk. The subsystem is a modified version of the MAC assembler. It should be copied to user SYSTEM as a file of type :PROG. FMAC can patch files of up to 64 Kwords with octal numbers or assembly instructions as input.

The first 64 Kwords of SEGFILO:DATA can be patched using FMAC. Patching SEGFILO:DATA will have the same effect as using the commands @LOOK-AT IMAGE and @LOOK-AT SEGMENT. The advantage of using FMAC is that the changes can be made with symbolic coding instead of octal machine code.

A patch implemented on MACM-AREA:DATA or SINTRAN:DATA is permanent, but will not become effective until the system is restarted using the )HENT command. Care must be taken when patching the IMAGE AREA. There is no protection against simultaneous updating by the RT LOADER or the file system. Such simultaneous updating will result in one of the two updates being lost. Use the command @SET-UNAVAILABLE before patching.

The following shows examples of patching with FMAC:

<pre> @FMAC - MACF - IMAGE FILE: <u>SINTRAN:DATA</u> &lt;address&gt;/&lt;old value&gt; &lt;new value&gt; )9EXIT @ </pre>	<pre> @FMAC - MACF - IMAGE FILE: <u>SINTRAN:DATA</u> 14052/045135 51751 51751/000000 150405 )9EXIT @ </pre>
--	---

---

### 11.3 PATCHING WITH DMAC

---

DMAC is a subsystem which is loaded onto a segment. It may be used to make changes to SINTRAN III RESIDENT and all segments. DMAC is a modified version of the MAC assembler. It is loaded as shown below:

```
@RT-LOADER
*READ-BINARY DMAC:BPUN, 7
CHANGING EXISTING SEGMENT? YES
*END-LOAD
*EXIT
@
```

The segment to be operated upon is then defined by the command:

```
)CLOAD <segment number>
```

The <segment numbers> 0, 1 and 2 have special meanings. 0 means SINTRAN III RESIDENT and the PAGING OFF AREA, 1 means RTCOMMON, and 2 means the IMAGE AREA. The following is an example of patching with DMAC:

```
@DMAC
)CLOAD 2
46540/124003 JMP I * 1
46541/131006 52320
46542/100170
)9EXIT
@
```

Either assembly instructions like `JMP I * 1` or octal numbers can be entered into the addresses specified before the slash (/). Carriage return only will leave the location unaltered.

---

#### 11.4 PATCHING WITH THE SINTRAN SERVICE PROGRAM

---

The SINTRAN SERVICE PROGRAM can be used to change system variables, system tables, etc. The program is only available for user SYSTEM, and it can only be activated from one terminal at a time. Chapter 12 describes the program in detail.

The program may operate on SINTRAN III RESIDENT, the IMAGE AREA, and the SAVE AREA. The relevant area to operate upon will be asked for each subcommand. The user only needs to answer YES or NO. Data fields and system variables can be patched with the two commands below. They will be described in detail in the next chapter.

```
CHANGE-DATAFIELD <logical device number>,<input>,<output>
CHANGE-VARIABLE <variable name> [,<index>] [,<new value>]
```

Below is an example of the use of the SINTRAN SERVICE PROGRAM to modify the data fields of SILENT and CENTRONIX terminals faster than 110 baud. A carriage return delay is necessary when using devices of this type. Terminals using the terminal interfaces may use a hardware facility on the interface by patching the input CNTREG for the relevant terminal. Bit 7 must be set.

@SINTRAN-SERVICE-PROGRAM

\*CHANGE-DATAFIELD

LOGICAL DEVICE NUMBER: 11

INPUT OR OUTPUT? INPUT

MEMORY? YES

IMAGE? YES

SAVE AREA? YES

IMAGE SAVE AREA

<u>CNTREG/44005</u>	<u>44005</u>	<u>44005</u>	<u>44205</u>
0	0	0	+

\*EXIT

@

The three columns of output represent SINTRAN III RESIDENT, the IMAGE AREA, and the SAVE AREA.

---

### 11.5 THE SYSTEM VARIABLE ENDCOR

---

ENDCOR is the name of a SINTRAN III variable, and 7ENDC is the symbol name of ENDCOR's contents. When extensions have been added to SINTRAN III, the contents of ENDCOR must be changed.

ENDCOR is the name of the location containing the first free address in the last page used by the resident part of SINTRAN III. Its contents may be inspected or changed using the CHANGE-VARIABLE command in the SINTRAN SERVICE PROGRAM or by looking at address 141 in SINTRAN III RESIDENT or the file SINTRAN:DATA. The initial value of 7ENDC is found in the symbol list.

The following example shows two alternative ways to find the value of 7ENDC, ie., the contents of ENDCOR:

```
@SINTRAN-SERVICE-PROGRAM
*CHANGE-VARIABLE
VARIABLE: ENDCOR
VALUE:
IMAGE? YES
SAVE AREA? YES
IMAGE: 51750 : <old contents of ENDCOR>
SAVE AREA: 51750
*EXIT
```

```
@LOOK-AT RESIDENT
*READY:
141/51750
- END -
@
```



---

### 11.6 PATCHING WHILE SINTRAN III IS STOPPED

---

This procedure should not be used unless absolutely necessary. There are patches which cannot be done on-line. The following procedure must then be performed:

- Press the STOP and MCL buttons on the computer's panel.
- Fetch MACM by using the first SINTRAN III floppy disk, and typing 1560&.
- Do a )HENT command.
- Do the patch <address>/<value> <new value>. The patch is now valid on the IMAGE AREA.
- Do a )GJEM command if the patch is to be made permanent, ie., on SINTRAN:DATA.
- Start SINTRAN III by typing 22!

A patch can be performed anywhere on a disk by means of the command:

)9ALTR,,,,,<sector address on the disk>

Sectors are explained in appendix G. If the mass storage address of a segment is found in the segment table, this address multiplied by the number of sectors per page can be used as a parameter to this command.



---

## 12 SINTRAN SERVICE PROGRAM REFERENCE

---

The SINTRAN SERVICE PROGRAM is a program that may change variables and tables used by the operating system, eg., the enter count for a terminal, the upper address of SINTRAN III RESIDENT, or the length of a time slice. This chapter contains a reference of the SINTRAN SERVICE PROGRAM commands. Sections 11.4 and 11.5 show examples of some commands.

The service program is entered by giving:

@SINTRAN-SERVICE-PROGRAM

The program is only available to user SYSTEM, and it can only be used from one terminal at a time. An asteriks (\*) is used to indicate that the program is ready to accept subcommands. You return to SINTRAN III by the EXIT command. The HELP command lists the available commands.

The SINTRAN SERVICE PROGRAM operates on SINTRAN III RESIDENT, the IMAGE AREA and the SAVE AREA. Each command asks which area is to be operated on. You should answer YES or NO in each case. Most of the commands can operate on all three areas simultaneously. Some can only operate on one area. They will ask which area to access.

Most numbers, both given as parameters to commands and output from the program, are octal by default. If the commands are used in a mode or batch file, they must be preceded by an @, ie., @CHANGE-VARIABLE. Some commands cannot be given from a batch job, but mode processing is possible.

**12.1 COMMAND SUMMARY**

The commands available in the SINTRAN SERVICE PROGRAM are shown in table 20. This section also explains key terms used in commands and parameters. The next section describes the commands in detail.

**SINTRAN SERVICE PROGRAM COMMANDS**

ASCII-DUMP	LAMU-INFORMATION
BACKGROUND-ALLOCATION-UTIL	LIST-ADDRESSES
CC	LIST-HDLC-BUFFER
CHANGE-BUFFER-SIZE	LIST-LAMU-CONSTANTS
CHANGE-DATAFIELD	LIST-SERVICE-COMMANDS
CHANGE-GPIB-BUFFERSIZE	LIST-TIME-SLICE-CLASS
CHANGE-TABLE	LIST-TIME-SLICE-PARAMETERS
CHANGE-VARIABLE	LIST-TIME-SLICED PROGRAMS
CLEAR-ENTER-COUNT	MONCALL-LOG
CPU-LOG	OCTAL-DUMP
CREATE-LAMU	PAGES-FROM-LAMU
DEFINE-BATCH-SUPERVISOR	PAGES-TO-LAMU
DEFINE-HDLC-BUFFER	PROTECT-LAMU
DEFINE-PROMPT-STRING	READ-BINARY
DEFINE-RTCOMMON-SIZE	REMOVE-FROM-BACKGROUND-TABLE
DEFINE-SEGMENT-FILE	REMOVE-FROM-EXTENDED-IDENT-TABLE
DEFINE-TIME-SLICE	REMOVE-FROM-IDENT-TABLE
DEFINE-TITLE	REMOVE-FROM-IOX-TABLE
DEFINE-USER-MONITOR-CALL	REMOVE-FROM-LOGICAL-UNIT-TABLE
DEFINE-USER-RESTART-PROGRAM	REMOVE-FROM-TIME-SLICE
DEFINE-USER-RESTART-SUBROUTINE	REMOVE-FROM-TIMER-TABLE
DEFINE-USER-START-SUBROUTINE	REMOVE-SINTRAN-COMMAND
DELETE-LAMU	REMOVE-SPOOLING-HEADER
DELETE-SEGMENT-FILE	RESET-COLDSTART-MODE-FILE
DISC-ACCESS-LOG	SEGMENT-WRITE-PERMIT
DUMP-RT-DESCRIPTION	SEGMENT-WRITE-PROTECT
DUMP-SEGMENT-TABLE-ENTRY	SET-CHANNEL-PRIORITY
EXIT	SET-CLOSED-SCRATCH-FILE-SIZE
FIND-CPULOOPTIME	SET-COLDSTART-MODE-FILE
HELP	SET-COMMAND-PROTECTION
INITIALIZE-SYSTEM-SEGMENT	SET-LAMU-CONSTANTS
INSERT-IN-BACKGROUND-TABLE	SET-MAX-ENTER-COUNT
INSERT-IN-EXTENDED-IDENT-TABLE	SET-SPOOLING-DEVICE-NUMBER
INSERT-IN-IDENT-TABLE	START-GPIB
INSERT-IN-IOX-TABLE	START-XMSG
INSERT-IN-LOGICAL-UNIT-TABLE	STOP-GPIB
INSERT-IN-TIME-SLICE	STOP-XMSG
INSERT-IN-TIMER-TABLE	SWAP-DIRECTORY-ENTRIES
INSERT-SPOOLING-HEADER	SWAPPING-LOG
LAMU-AREAS	

Table 20. The SINTRAN SERVICE PROGRAM Commands

The following is an explanation of terms used in the commands and parameters:

#### Background table

A table which contains the input data field for all terminals, batch processors, TADs and ND-NET channels connected to background processes.

#### Extended ident table

New devices may be added to a system after SINTRAN III has been generated. If there is no more space in the ident table, the extended ident table can be used. It usually contains ident codes with high values.

#### Data fields

Each device is described by a data field. All data fields have some information in common, eg., the state of the device, linking in various system queues, the device type, the ring protection, and identification of RT programs that reserve the device.

#### Ident table

An ident code is used to identify which device that caused an interrupt. The ident table connects the ident code to a logical device number. There is one ident table for each interrupt level.

#### IOX table

The IOX table contains the physical device numbers of all input and output devices that can be accessed through the machine instruction IOX.

#### Lamu

Abbreviation of logical addressed memory unit. A lamu is a limited physically and virtually contiguous address area in the SINTRAN III VSE and VSE-500 systems. Lamus are intended to be an extension to the segment structure. They make it possible for RT programs and background processes to address more space than covered by the available 3 segments. Lamus can be shared by several RT programs or CPUs.

#### RT description

Each program is described by an RT description which has an address in the RT description table. This address is sometimes used to identify the RT program.

**RTCOMMON**

An area in physical memory which can be used by all RT programs. It can be used for fast communication between RT programs. Some SINTRAN III systems are generated without RTCOMMON.

**Segment table**

Each segment is described by an entry in the segment table. The entry in the segment table describes where the segment is located, its size, how it is linked in various system queues, and it describes some switches.

**Spooling header**

A page that precedes each file output on a printer with spooling. The spooling header contains the name of the user who ordered the printout and other information.

**Spooling index**

The index number connected to each spooling program when the system was generated. SPRT1 has spooling index 1, SPRT2 has spooling index 2, etc. The spooling indices range from 1 and upward.

**Time slicer**

The part of the operating system which allows public users to share the CPU. Each user is allowed to use the CPU for short intervals called time slices.

**Timer table**

The table contains the addresses of all data fields which should be checked for device time-outs. A device time-out occurs when a device does not answer within a predefined time.

**XMSG**

XMSG is a system used for task-to-task communication, eg., between two RT programs on different interrupt levels or between a public user's program and an RT program.

---

## 12.2 DETAILED DESCRIPTION OF COMMANDS

---

This section describes the commands in the SINTRAN SERVICE PROGRAM in detail. The commands are ordered alphabetically.

### \*ASCII-DUMP

Dump each byte of an area as ASCII characters. See also OCTAL-DUMP. The command and parameters are defined as:

\*ASCII-DUMP <area> [, <segment number>], (<low address>),  
(<high address>), (<output file>)

AREA: Legal parameters are MEMORY, IMAGE, SAVE AREA or SEGMENT.

SEGMENT NUMBER: If SEGMENT is specified as <area>, the octal segment number should be entered.

LOW ADDRESS and HIGH ADDRESS: Address range to be dumped including limits. Both values are octal with 0 as default value.

OUTPUT FILE: The file where the information should be dumped. Default is your terminal.

#### RULES:

- 1) Each line of the dump consists of a maximum of 64 characters. Control characters such as LF and CR are output as printable characters, ie., CR in memory generates a carriage return on the terminal.

**\*CC**

This command has no effect. It is normally used to comment mode and batch files. The command and parameters are defined as:

**\*CC** <text>

TEXT:

The text can be any printable characters. It is terminated by carriage return.

**\*CHANGE-BUFFER-SIZE**

Change the length of the ring buffer of a device. The total buffer area is limited depending on system configuration. If a substantial increase in size is desired, calculations should be made. The command and parameters are defined as:

**\*CHANGE-BUFFER-SIZE** <logical device number>,<input/output>,  
<buffer size>,<(<image?>)>,<(<save area?>)>

LOGICAL DEVICE NUMBER:	The octal logical device number identifying the device.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.
BUFFER SIZE:	The octal length of the ring buffer in words or bytes.
IMAGE? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) For SINTRAN III/VSX, the following rule applies to terminals:  
The size of the datafields (input+output), plus the buffers (input+output), must fit inside one page. This gives the maximum size of the sum of input and output buffers to 3424<sub>8</sub> bytes. The default buffer size for terminals are 134<sub>8</sub> bytes in input and 270<sub>8</sub> bytes in output.



### \*BACKGROUND-ALLOCATION-UTILITIES

The background allocation system makes it possible to generate SINTRAN III with a larger number of terminals/TADs than the number of background processes. When a terminal/TAD requests a background process (user presses ESCAPE), the first free background process will be allocated if there is a free one.

\*BACKGROUND-ALLOCATION-UTILITIES has these subcommands:

SET-PERMANENT-CONNECTION <termno> <memory> <image> <save area>

A terminal will be permanently connected to a background process, if one is free. The connection is reset by:

RESET-PERMANENT-CONNECTION <termno> <memory> <image> <save area>

DISABLE-TIMEOUT <termno> <memory> <image> <save area>

The timeout function will not affect this terminal. The timeout function is restored by:

ENABLE-TIMEOUT <termno> <memory> <image> <save area>

CHANGE-LOGOUT-TIME <time> <memory> <image> <save area>

Changes the amount of time a terminal may be inactive before it is logged out (if the timeout function is on). Original value = 30 minutes.

CHANGE-WARNING-TIME <time> <memory> <image> <save area>

Changes the time a terminal may be inactive before the first logout warning is printed. Original value = 25 minutes.

TIMEOUT-OFF

Disables the timeout function for the entire system.

TIMEOUT-ON

Enables the timeout function for the entire system.

DISPLAY

Gives an overview of all the terminals and background processes.

LIST-PARAMETERS

Lists the current timeout parameters.

FREE-BACKGROUND-PROGRAMS

Lists those background programs presently not connected to a terminal/TAD.

HELP

EXIT

**RULES:**

- 1) The Background Allocation System is an option to be ordered on the SINTRAN III order form.

**\*CHANGE-GPIB-BUFFERSIZE**

Change size of buffers connected to a GPIB controller.

```
*CHANGE-GPIB-BUFFERSIZE <controller no.>,<user buffer size (oct)>,<DMA buffer size (oct)>
```

CONTROLLER NO.:	GPIB controller.
-----------------	------------------

USER BUFFER SIZE (OCT): User buffer size in (octal) words.

DMA BUFFER SIZE (OCT):	DMA buffer size in (octal) words.
------------------------	-----------------------------------

### \*CHANGE-DATAFIELD

Change the contents of variables in the data fields connected to devices. The displacements of items in each data field can be entered as symbolic names.

\*CHANGE-DATAFIELD <logical device number>,<input/output>,  
(<memory?>),(<image?>),(<save area?>)  
(subcommands for patching)

LOGICAL DEVICE NUMBER: The octal logical device number identifying the device.

INPUT/OUTPUT: Select the INPUT or OUTPUT part of the device.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

SUBCOMMANDS FOR PATCHING: These subcommands follow rules 3, 4 and 6 of the @LOOK-AT command. See the SINTRAN III Reference Manual (ND-60.128). The address specified must be a relative octal address within the data field or the symbolic name of an item within the data field.

The symbolic names available as displacements in data fields are shown in table 21. They cannot be abbreviated. For further information of data fields, see the SINTRAN III Real Time Guide (ND-60.133).

AERRB	CTRG	HDEV	MFUNC	TERM
ARG	CTTYP	HENTE	MINBHOLD	TMR
BHOLD	CXRG	HSTAT	MLINK	TMSUB
BLSZ	DBADR	IBLOAD	MTRANS	TRG
BRKTAB	DBPROG	ICORAD	RESLINK	TINFO
BSTATE	DERROR	IFUNC	ROUSP	TRLREG
BUFST	DFLAG	IMAXBHOLD	RTRES	TSPEED
BWLINK	DFOPP	IMAXW	SERRB	TSTATE
CARG	DRG	IOLOG	SETDV	TTMR
CESCP	DRIVER	IOTRANS	SSREF	TYPRING
CFREE	DRT	IRETW	STDEV	WERRB
CHARI	ECHOTAB	ISTATE	STDRIV	WFLAG
CNTRG	ERCNT	LAST	STRSEG	XRG
COMFL	FLAGB	MAX	TACNS	
CONVTAB	FYLLE	MAXBHOLD	TACOUNT	

Table 21. Symbolic Names in Data Fields

The following example shows how to change the speed for terminal 36 to 9600 baud. The variable TSPEED in the input data field of terminal 36 is modified from 177777 to 210. See appendix K for details of terminal speeds. The modification will not take effect until the system has been restarted.

```
@SINTRAN-SERVICE-PROGRAM
*CHANGE-DATAFIELD 36D,I,N,Y,Y
TSPEED/177777 177777 210
      0          0
*EXIT
@
```

#### RULES:

- 1) The command must not be used in batch jobs.

### \*CHANGE-VARIABLE

Examine and change system variables. These may be single variables or arrays. For some variables, the IMAGE AREA and/or SAVE AREA may be affected. Some variables may also affect SINTRAN III RESIDENT.

```
*CHANGE-VARIABLE <variable name>[,<index>],(<value>)[,(<memory?>)],
                  (<image?>),(<save area?>)
```

VARIABLE NAME:

The system variables shown in the following table can be examined and modified.

INDEX:

An octal index will be requested if the <variable name> is an array.

VALUE:

New octal value. The old value is default.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

VARIABLE	MEANING
BGLPAGE	Last logical page number legal for background programs.
BGFPAGE	First logical page number legal for background programs.
BUFERASE	Erase buffer option on/off
BYPINITC	Flag to bypass initial commands.
CACHLIM	Upper and lower limit of nocache area.
CCFPAGE	First legal logical page for RT common.
CCLPAGE	Last legal logical page for RT common.
CNVRT	Address of logical device tables. SINTRAN III RESIDENT is affected for index values from 0-24.
CPULOOPTIME	Number of microseconds in one run of the idle loop.
ENDCOR	Upper address of SINTRAN III RESIDENT.
EXSECURITY	Flags to show which security features to be used.
EXTDS	Address of the extended ident code tables. SINTRAN III RESIDENT is affected for index values from 0-3.
FIXMAX	Maximum number of pages which can be fixed in physical memory simultaneously. Affects SINTRAN III RESIDENT.
IDNTS	Address of the ident code tables. Affects SINTRAN III RESIDENT for index values from 0-3.
IMASK	Value is a mask to be used for enabling internal interrupts, ie., the TRR IIE instruction.
LCACHLIM	Change the lower limit of the CACHE-INHIBIT-LIMIT.
LOADI	Set to 0 if the RT LOADER shall initialize RTFIL. SINTRAN III RESIDENT is affected.
MAXP	Maximum number of pages in physical memory for a demand segment. SINTRAN III RESIDENT is affected.
NMATP	Number of RFA attempts before forced logout.
RTFPAGE	First legal logical page number for RT programs on page table 1.
RTLPAGE	Last legal logical page number for RT programs on page table 1.
SAFILNO	Segment file number for memory save.
SWPFLAG	Swapping and disk reservation as in the H-version.
TABLES	Address of timer, background, batch and RT common table (CCTAB). SINTRAN III RESIDENT is affected for index from 0-3.
TMCTAB	Monitor call types. See DEFINE-USER-MONITOR-CALL. The TMCTAB array is a byte array. The parameter <index> is a word index. Thus two bytes are changed by one command. SINTRAN III RESIDENT is affected for index values from 0-107.
UCACHLIM	Change the upper limit of the CACHE-INHIBIT-LIMIT.
UNAFLEG	Flag set if system is unavailable after the command @SET-UNAVAILABLE. SINTRAN III RESIDENT is affected.
USEGADR	Address of the first free entry in the segment table.

Table 22. System Variables Available for Patching

**\*CHANGE-TABLE**

Change values in one of four internal tables:

**USER-RESERVED-DEVICE-NUMBERS**

A table of hardware device numbers which should not be accessed at system start-up time. An element in this table is defined by the first and last device number of an interval.

**USER-RESERVED-MEMORY-AREA**

This table specifies memory pages which should not be used for system tables (memory map, device buffers, etc.). An element in this table is defined by the first and last physical page of an area not to be used for system tables.

**MEMORY-AREA-UNAVAILABLE-FOR-SWAPPING**

This table specifies memory areas which should not be used for swapping. An element in this table is defined by the first and last physical pages of an area not to be used for swapping.

**MEMORY-AREA-INVISIBLE-FOR-THIS-SYSTEM**

This table specifies memory areas which should not be accessed at all. This feature is used to handle the situations when several CPUs share the same multiport memory. If one CPU is using a particular area of memory, the start-up routine of another CPU may destroy the contents of this area if accessing it. An element in this table is defined by the first and last physical pages of an area to be invisible.

**\*CHANGE-TABLE <table>**

The CHANGE-TABLE command has the following subcommands:

**LIST-TABLE <image or save-area (default is image)> <output file>**  
Print the contents of the current table on the output file.  
Default output file is terminal.

**CHANGE-TABLE <table>**  
Change the current table.

**CHANGE-ELEMENT <image or save-area (default is image)>**  
    <element number>  
    <first devno. (oct)> <last devno. (oct)>  
    or: <first page (oct)> <last page (oct)>  
Change one element of the current table.

**DELETE-ELEMENT <image or save-area (default is image)>**  
    <element number>  
Clear one element of the current table.

**INSERT-ELEMENT <image or save-area (default is image)>**  
    <first devno. (oct)> <last devno. (oct)>  
    or: <first page (oct)> <last page (oct)>  
Insert a new element in the current table.

CLEAR-TABLE <image or save-area (default is image)>  
Clear all of the current table.

EXIT

HELP

**\*CPU-LOG**

Report CPU activity.

\*CPU-LOG (<interval in seconds>),( <output file>)

INTERVAL IN SECONDS:		Interval. The default is 30 seconds.
----------------------	--	--------------------------------------

OUTPUT FILE:		The default is terminal.
--------------	--	--------------------------

**\*DEFINE-HDLC-BUFFER**

Allocate buffer for specified HDLC interface.

\*DEFINE-HDLC-BUFFER <logical device number>,<buffer size>

LOGICAL DEVICE NUMBER:		Logical device number for HDLC interface.
------------------------	--	---

BUFFER SIZE:		Buffer size.
--------------	--	--------------

### **\*DEFINE-BATCH-SUPERVISOR**

Define an RT program to be started each time a batch job is terminated.

```
*DEFINE-BATCH-SUPERVISOR <RT program>, (<memory?>), (<image?>),  
                           (<save area?>)
```

RT PROGRAM NAME:	An RT description address or an RT program name.
------------------	--

MEMORY?, IMAGE? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
---------------------------------	--

### **\*DEFINE-PROMPT-STRING**

Define a prompt string to be printed instead of an @ in SINTRAN III remote mode. The system name may, for example, precede the @. The string may also be declared to replace the default prompt on the local system.

```
*DEFINE-PROMPT-STRING <string>, <local mode?>,<memory>),  
                      (<save area?>)
```

STRING:	The string to replace the old one. The text is terminated with an apostrophe (').
---------	---

LOCAL MODE?:	Answer YES or NO to whether the string also should be used in local mode.
--------------	---

MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
-------------------------	--



**\*DEFINE-RTCOMMON-SIZE**

Define the size of the RT common area in pages.

\*DEFINE-RTCOMMON-SIZE <number of pages>, <first physical page>,  
(<image?>), (<save area?>)

NUMBER OF PAGES:

The new octal size of the RT common area.

FIRST PHYSICAL PAGE:

The page in physical memory where RT common should start. In addition to the pages generated for the system, only 8 pages can be added. The default value is the upper end of physical memory.

IMAGE? and SAVE AREA?:

Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) The size can be increased by up to 10<sub>8</sub> pages in addition to the RT common size defined at system generation.

**\*DEFINE-SEGMENT-FILE**

Connect a segment file number to a segment file. This is needed for the RT LOADER to access the file.

```
*DEFINE-SEGMENT-FILE (<memory?>), (<save area?>),  
                      <segment file number>, <segment file name>  
                      [,(<redefine segment file?>)]
```

MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
SEGMENT FILE NUMBER:	Specify a segment file number in the range 0-3.
SEGMENT FILE NAME:	Any existing contiguous file.
REDEFINE SEGMENT FILE?:	Only requested if the <segment file number> is already defined. Specify YES or NO. The default answer is NO.

**RULES:**

- 1) User SYSTEM and user RT must have read and write access to the segment file.
- 2) The page address of all the pages in the segment file must be less than 7777<sup>8</sup>.
- 3) The command must not be used in batch jobs.

**\*DEFINE-TIME-SLICE**

Define the length of the time slices of the background processes which control the terminal and batch processors. The unit used in the parameters is 10 basic time units, ie., 20 milliseconds.

\*DEFINE-TIME-SLICE (<image?>),(<save area?>),  
(<change timeslice parameters (yes/no?)>)

IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

CHANGE TIMESLICE PARAMETERS (YES/NO?): If you answer yes, you will be asked for new values for the time slice parameters.

Example:

**@SINTRAN-SERVICE-PROGRAM****\*DEFINE-TIME-SLICE**

IMAGE? YES

SAVE-AREA? NO

CHANGE TIMESLICE PARAMETERS (YES/NO) (DEFAULT IS NO): YES

PRIORITY FOR OWNER OF SYSTEM RESOURCES WHICH ARE WAITED FOR BY OTHER PROGRAMS	(1B - 77B) /	67/:	IMAGE
NO. OF BASIC TIME UNITS IN ONE TIMESLICE UNIT	(1B - 400B) /	14/:	
LOWEST PRIORITY BEFORE GETTING RAISED ON BREAK	(1B - 70B) /	40/:	
LOWEST TIME COUNT BEFORE GETTING HASHED	(1B - 400B) /	22/:	
BIT MASK USED WHEN HASHING	(1B - 177B) /	17/:	
CHANGE TIMESLICE ELEMENTS (YES/NO) (DEFAULT IS NO)?	<u>YES</u>		
TIMESLICE CLASS	(0B - 7B) /	0/:	<u>6</u>

ESCAPE ELEMENT FOR THIS CLASS	(0B - 37B) /	30/:	<u>30</u>
BREAK ELEMENT FOR THIS CLASS	(0B - 37B) /	30/:	<u>30</u>
TIMESLICE ELEMENT TO CHANGE	(0B - 37B) /	30/:	<u>-</u>
PRIORITY FOR THIS ELEMENT	(1B - 77B) /	10/:	<u>10</u>
TIME COUNT FOR THIS ELEMENT	(1B - 400B) /	2/:	<u>2</u>
POINTER TO NEXT ELEMENT	(0B - 37B) /	31/:	<u>31</u>
CHANGE NEXT ELEMENT (YES/NO) (DEFAULT IS NO):	<u>YES</u>		
TIMESLICE ELEMENT NO.	31		
PRIORITY FOR THIS ELEMENT	(1B - 77B) /	6/:	<u>6</u>
TIME COUNT FOR THIS ELEMENT	(1B - 400B) /	4/:	<u>4</u>
POINTER TO NEXT ELEMENT	(0B - 37B) /	32/:	<u>32</u>
CHANGE NEXT ELEMENT (YES/NO) (DEFAULT IS NO):	<u>YES</u>		
TIMESLICE ELEMENT NO.	32		
PRIORITY FOR THIS ELEMENT	(1B - 77B) /	1/:	<u>1</u>
TIME COUNT FOR THIS ELEMENT	(1B - 400B) /	30/:	<u>30</u>
POINTER TO NEXT ELEMENT	(0B - 37B) /	31/:	<u>31</u>
CHANGE NEXT ELEMENT (YES/NO) (DEFAULT IS NO):			
MORE CLASSES (YES/NO) (DEFAULT IS NO):	<u>NO</u>		

**\*EXIT**

@

**\*DEFINE-TITLE**

Define the string to be output in addition to system version string when a user logs in on a terminal. The string is also output as part of the spooling header and as response to the command @LIST-TITLE.

\*DEFINE-TITLE <text>, (<memory?>), (<save area?>)

**TEXT:**

Any printable characters terminated by an apostrophe ('). The character \$ will start a new line on output. Carriage return on input is ignored. An apostrophe only specifies no text.

**MEMORY? and SAVE AREA?:**

Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) The maximum is 120 characters including the apostrophe.
- 2) The command must not be used in batch jobs.

The following is an example of the command and the generated output:

```
@SINTRAN-SERVICE-PROGRAM
*DEFINE-TITLE $ND-100.54 $SYSTEM NUMBER 485 $'
*EXIT
@LIST-TITLE
```

SINTRAN III - VSX I

ND-100.54  
SYSTEM NUMBER 485

@

**\*DEFINE-USER-MONITOR-CALL**

Define number, entry point address and type of user defined monitor calls. The code should be assembled using FMAC or DMAC.

```
*DEFINE-USER-MONITOR-CALL <monitor call number>, <start address>,
                           (<type>), (<memory?>), (<image?>),
                           (<save area?>)
```

MONITOR CALL NUMBER:	The octal number of the monitor call. User defined monitor calls may be in the range 170-177.
----------------------	---

START ADDRESS:	The octal entry point of the monitor routine.
----------------	---

TYPE:	Specify 1 if the monitor call should be available to RT programs only and 16 if it also should be available to background programs. The default value is 1.
-------	---

MEMORY?, IMAGE? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
---------------------------------	--

**\*DEFINE-USER-RESTART-PROGRAM**

Define the user restart RT program to be started by SINTRAN III after each power fail. The RT program may, for example, request the clock to be updated. See also DEFINE-USER-RESTART-SUBROUTINE.

```
*DEFINE-USER-RESTART-PROGRAM <program>, (<memory?>), (<image?>),
                              (<save area?>)
```

PROGRAM:	An octal RT description address or an RT program name.
----------	--

MEMORY?, IMAGE? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
---------------------------------	--

**\*DEFINE-USER-RESTART-SUBROUTINE**

Define the start address of a subroutine to be called when SINTRAN III is restarted after a power fail. See also the related command DEFINE-USER-RESTART-PROGRAM.

\*DEFINE-USER-RESTART-SUBROUTINE <restart address>, (<memory?>)  
(<image?>), (<save area?>)

RESTART ADDRESS:

The octal entry point of the subroutine.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) The subroutine must reside in SINTRAN III RESIDENT.

**\*DEFINE-USER-START-SUBROUTINE**

Define the address of a subroutine to be executed when the system is started.

\*DEFINE-USER-START-SUBROUTINE <start address>, (<image?>),  
(<save area?>)

START ADDRESS:

The octal entry point of the subroutine.

IMAGE? and SAVE AREA?:

Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) The subroutine must reside in SINTRAN III RESIDENT or on the operator communication segment, ie., segment 3.

**\*DELETE-LAMU**

Deletes a selected lamu from the lamu table. See also CREATE-LAMU and LAMU-INFORMATION.

**\*DELETE-LAMU** <lamu id>

LAMU ID:

The number identifying the lamu in the lamu table.

**RULES:**

- 1) The lamu cannot be in use by RT programs.
- 2) The physical pages of the lamu will remain in the lamu area.

**\*DELETE-SEGMENT-FILE**

Set a segment file unavailable for the RT LOADER. The file is not deleted from the directory. Use the @DELETE-FILE command for this purpose.

**\*DELETE-SEGMENT-FILE** (<memory?>),( <save area?>), <segment file number>

MEMORY? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

SEGMENT FILE NUMBER: The number of one of the four segment files, ie., a number from 0-3.

**RULES:**

- 1) A defined segment file must not be deleted without previously having been undefined with DELETE-SEGMENT-FILE.

**\*DISC-ACCESS-LOG**

Log all or a selection of disk accesses.

**\*DISC-ACCESS-LOG**

The command \*DISC-ACCESS-LOG has several subcommands:

```
DEFINE-DISC-ACCESS-LOG <disc access log file>
                        <small or big record size on disc log file
                          (default is big)>
                        <log all disc accesses (default is yes)>
                        [<count disc accesses to one controller only
                          (default is no)>]
                        [<logical device number of disc to log (octal)>]
                        [<count disc accesses to one disc unit only
                          (default is no)>]
                        [<drive number of drive to log>]
                        [<log only write accesses (default is no)>]
                        [<log only read accesses (default is no)>]
                        [<log only accesses to a limited part of the disc
                          (default is no)>]
                        [<first disc address in the disc part to log
                          (octal only)>]
                        [<last disc address in the disc part to log
                          (octal only)>]
```

Define the parameters for the disk log. Logfile must be a contiguous file. Record size is either 4 (small) or 8 (big) words.

**START-DISC-ACCESS-LOG**

Start logging of disk accesses if a log has been defined.

**STOP-DISC-ACCESS-LOG**

Stop logging of disk accesses.

**START-DISC-ACCESS-COUNTER <count all disc accesses (default is yes)>**

```
                        [<count disc accesses to one controller only
                          (default is no)>]
                        [<logical device number of disc to log (octal)>]
                        [<count disc accesses to one disc unit only
                          (default is no)>]
                        [<drive number of drive to log>]
```

Define and start a simple disk access log which only counts the number of read and write accesses.

**STOP-DISC-ACCESS-COUNTER**

Stop the simple disk access log.

**CLEAR-DISC-ACCESS-COUNTER**

Clear the counters used by the simple disk access log.

**DISC-ACCESS-COUNTER**

Report the values of the counters used by the simple disk access log.



DISC-DRIVER-ERROR-INFORMATION <logical device number>

Report the values of certain error variables in the disk driver.

DISC-ERROR-STATUS <logical device number> <unit number>

Report the error status in the disk datafield.

LOG-DISC-ACCESS-COUNTER <interval in seconds (default is 60 secs)>

Report the values of the counters used by the simple disk access log at specified intervals. This command must be terminated by pressing <escape>.

EXIT

HELP

**RULES:**

- 1) The Disk Access Log is an option to be ordered on the SINTRAN III order form.

### **\*DUMP-RT-DESCRIPTION**

Dump the symbolic names and the contents of locations in an RT description. See also the SINTRAN III command @LIST-RT-DESCRIPTION.

**\*DUMP-RT-DESCRIPTION** <program>, <area>, (<output file>)

PROGRAM:	An octal RT description address or an RT program name.
AREA:	Legal parameters are MEMORY, IMAGE, and SAVE AREA.
OUTPUT FILE:	The file where the information should be dumped. Default is your terminal.

### **\*DUMP-SEGMENT-TABLE-ENTRY**

Dump the symbolic name and contents of locations in a segment table entry. See also the SINTRAN III command @LIST-SEGMENT.

**\*DUMP-SEGMENT-TABLE-ENTRY** <segment number>, (<area>), (<output file>)

SEGMENT NUMBER:	The octal number of the segment to be dumped.
AREA:	Legal parameters are MEMORY, IMAGE or SAVE AREA.
OUTPUT FILE:	The file where the information should be dumped. Default is your terminal.

The following example shows the output from the command. The contents of the segment entries are described in the SINTRAN III Real Time Guide (ND-60.133).

```
*DUMP-SEGMENT-TABLE-ENTRY 30, MEMORY, ,  
SEGLINK:          0  
BPAGELINK:        0  
LOGADR:           1074  
MADR:              325  
FLAG:             162003
```

**\*EXIT**

Leave the SINTRAN SERVICE PROGRAM and return to SINTRAN III.

\*EXIT.

**\*HELP**

List all SINTRAN SERVICE PROGRAM commands. The command is identical to LIST-SERVICE-COMMANDS.

\*HELP (<output file>), [<command>]

OUTPUT FILE:	The file where the commands should be listed. Default is your terminal.
--------------	--

COMMAND:	List commands matching command name given.
----------	--

**\*INITIALIZE-SYSTEM-SEGMENT**

Fetch a new copy of the system segment for a terminal. The segment is transferred from the save area to the segment file.

\*INITIALIZE-SYSTEM-SEGMENT <segment (name or number(oct))>

L

SEGMENT (NAME OR NUMBER(OCT)):	The segment name or (octal) number of a system segment.
--------------------------------	--

**RULES:**

- 1) This command will only be executed if the corresponding background process is passive. Otherwise, the message BACKGROUND PROGRAM ACTIVE will be given.

**\*FIND-CPULOOPTIME**

Find number of "CPU loops" per second and store it in the SINTRAN III variable CPULOOPTIME.

**\*FIND-CPULOOPTIME**

**RULES:**

- 1) This command should only be issued when there is no activity on the system.

**\*INSERT-IN-BACKGROUND-TABLE**

Insert a device into the background table. See the related command REMOVE-FROM-BACKGROUND-TABLE.

**\*INSERT-IN-BACKGROUND-TABLE** <logical device number>, <input/output>,  
(<memory?>), (<image?>), (<save area?>)

LOGICAL DEVICE NUMBER: An octal logical device number.

INPUT/OUTPUT: Select the INPUT or OUTPUT part of the device.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) There must be a free entry in the background table.

**\*INSERT-IN-EXTENDED-IDENT-TABLE**

Insert an entry in the extended ident code table of a hardware interrupt level. See also REMOVE-FROM-EXTENDED-IDENT-TABLE.

**\*INSERT-IN-EXTENDED-IDENT-TABLE** <level>, <logical device number>,  
<input/output>, <ident code>, (<memory?>),  
(<image?>), (<save area?>)

LEVEL: The octal interrupt level.

LOGICAL DEVICE NUMBER: An octal logical device number in the range 1-77 or greater than 200.

INPUT/OUTPUT: Select the INPUT or OUTPUT part of the device.

IDENT: The new octal ident code.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*INSERT-IN-IDENT-TABLE**

Insert an entry in the ident code table of a hardware interrupt level.  
See also REMOVE-FROM-IDENT-TABLE and INSERT-IN-EXTENDED-IDENT-TABLE.

```
*INSERT-IN-IDENT-TABLE <level>, <logical device number>,
                        <input/output>, <ident code>, (<memory?>),
                        (<image?>), (<save area?>)
```

LEVEL:	The octal interrupt level.
LOGICAL DEVICE NUMBER:	An octal logical device number in the range 1-77 or greater than 200.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.
IDENT:	The new octal ident code.
MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.	

The following is an example of inserting and removing the output part  
of a device in the ident table:

```
@SINTRAN-SERVICE-PROGRAM
*INSERT-IN-IDENT-TABLE 12, 5, OUTPUT, 3
*REMOVE-FROM-IDENT-TABLE 12, 5, OUTPUT
```

**\*INSERT-IN-IOX-TABLE**

Make a hardware device number available for @EXECUTE-IOX and EXIOX  
(MON 31). See also REMOVE-FROM-IOX-TABLE.

```
*INSERT-IN-IOX-TABLE <hardware device number>, (<memory?>),
                    (<image?>), (<save area?>)
```

HARDWARE DEVICE NUMBER:	An octal physical device number.
MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.	

Insert a logical device number in the logical device table. See also REMOVE-FROM-LOGICAL-UNIT-TABLE.

LOGICAL DEVICE NUMBER:	An octal logical device number in the range 1-77 or greater than 200.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.
DATA FIELD:	The octal address of the data field of the device.

- 1) There must be a free entry in the logical device table for the logical device number.

```
@SINTRAN-SERVICE-PROGRAM
*INSERT-IN-LOGICAL-UNIT-TABLE 5, INPUT, 14341
*REMOVE-FROM-LOGICAL-UNIT-TABLE 6, OUTPUT
```

**\*INSERT-IN-TIME-SLICE**

Cause the background process priority to be changed dynamically. If a background process is not in the time slicer, it will run at a fixed priority. See also REMOVE-FROM-TIME-SLICE and @PRIOR.

\*INSERT-IN-TIME-SLICE <logical device number>,<time slice class>,  
(<memory?>),(<image?>),(<save area?>)

LOGICAL DEVICE NUMBER: The octal logical device number of a terminal, batch processor or communication device.

TIME SLICE CLASS: The number of time slice class to be used.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) A background process not in the time slicer will run on a fixed priority. This priority is defined by @PRIOR.



**\*INSERT-IN-TIMER-TABLE**

Insert a logical device number in the timer table. See also REMOVE-FROM-TIMER-TABLE

\*INSERT-IN-TIMER-TABLE <logical device number>, <input/output>

LOGICAL DEVICE NUMBER:	An octal logical device number in the range 1-77 or greater than 200.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.

**RULES:**

- 1) There must be a free entry in the timer table.

The following is an example of inserting and removing devices from the timer table:

```
@SINTRAN-SERVICE-PROGRAM  
*INSERT-IN-TIMER-TABLE 5,OUTPUT  
*REMOVE-FROM-TIMER-TABLE 6,INPUT
```

**\*INSERT-SPOOLING-HEADER**

Cause the spooling program to print the spooling header and trailer between each file output by the spooling system. See also the command REMOVE-SPOOLING-HEADER.

**\*INSERT-SPOOLING-HEADER** <spooling index>,( <memory?> ),( <save area?> )

**SPOOLING INDEX:** The index of the spooling device as defined by the command SET-SPOOLING-DEVICE-NUMBER.

**MEMORY? and SAVE AREA?:** Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*LAMU-AREAS**

List the memory areas reserved for lamus. See also LAMU-INFORMATION.

**\*LAMU-AREAS** ( <output file> )

**OUTPUT FILE:** The file where the information should be dumped. Default is your terminal.

**\*LAMU-INFORMATION**

List all the relevant information of one or more lamus. See also LAMU-AREAS.

**\*LAMU-INFORMATION** <lamu id>,( <output file> )

**LAMU ID:** The number that identifies the lamu in the lamu table. If 0 is specified, information about all lamus will be listed.

**OUTPUT FILE:** The file where the information should be dumped. Default is your terminal.

**\*LIST-HDLC-BUFFER**

List buffer for specified HDLC interface.

**\*LIST-HDLC-BUFFER** <logical device number>

LOGICAL DEVICE NUMBER:		Logical device number for HDLC interface.
------------------------	--	---

**\*LIST-TIME-SLICE-CLASS**

List information about a specific time slice class.

**\*LIST-TIME-SLICE-CLASS** <timeslice class>,<image or save-area>

TIMESLICE CLASS:		Time slice class.
------------------	--	-------------------

IMAGE OR SAVE-AREA:		Select area.
---------------------	--	--------------

**\*LIST-TIME-SLICE-PARAMETERS**

List time slicer information.

**\*LIST-TIME-SLICE-PARAMETERS** (<image?>),(<save-area?>)

IMAGE? or SAVE-AREA?:		Select the areas to be modified by answering each question with YES or NO. The default answer is NO.
-----------------------	--	--

**\*LIST-TIME-SLICED-PROGRAMS**

List information about time sliced programs.

**\*LIST-TIME-SLICED-PROGRAMS**

### **\*LIST-ADDRESSES**

List the addresses of the most commonly used global variables and entry points within SINTRAN III RESIDENT.

\*LIST-ADDRESSES (<output file>)

OUTPUT FILE:		The file where the information should be dumped. Default is your terminal.
--------------	--	---

The following shows an example of the output:

```
*LIST-ADDRESSES,
RET:              1
GET0:             1
GET1:             1
GET2:             1
GET3:             1
GET4:             1
GET5:             1

*
```

### **\*LIST-LAMU-CONSTANTS**

List the total number of lamus and the number of lamus per RT program.

\*LIST-LAMU-CONSTANTS

The command has no parameters.

### **\*LIST-SERVICE-COMMANDS**

List all SINTRAN SERVICE PROGRAM commands. The command is identical to HELP.

\*LIST-SERVICE-COMMANDS (<output file>)

OUTPUT FILE:		The file where the commands should be listed. Default is your terminal.
--------------	--	--

**\*OCTAL-DUMP**

Dump each byte of an area as octal numbers. Use ASCII-DUMP dump to have the area dumped as characters.

\*OCTAL-DUMP <area> [,<segment number>],(<low address>),  
(<high address>),(<output file>)

AREA: Legal parameters are MEMORY, IMAGE, SAVE AREA or SEGMENT.

SEGMENT NUMBER: If SEGMENT is specified as <area>, the octal segment number should be entered.

LOW ADDRESS and HIGH ADDRESS: Address range to be dumped including limits. Both values are octal with 0 as default value.

OUTPUT FILE: The file where the information should be dumped. Default is your terminal.

**RULES:**

- 1) Each line of the dump consists of 8 words. A separate column lists the address of the first word.

The following example shows how the memory area from 1000-1777<sub>8</sub> can be dumped on the terminal:

\*OCTAL-DUMP MEMORY,,1000,1777,,

Inform on monitor call activity for the system.

## EXIT

**\*PAGES-FROM-LAMU**

Return pages from the lamu area to the swapping area. See also PAGES-TO-LAMU and DELETE-LAMU.

**\*PAGES-FROM-LAMU** <first physical page>,( <memory?> ),( <image?> )

FIRST PHYSICAL PAGE:	The first page to be removed.
MEMORY? and IMAGE?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*PROTECT-LAMU**

Change lamu protection.

**\*PROTECT-LAMU** <lamu id>,<ring>,<protection bits (rfw)>

LAMU ID:	Lamu id of lamu to be protected.
RING:	Protection ring.
PROTECTION BITS (RFW):	Protection bits (read/fetch/write).

**\*PAGES-TO-LAMU**

Move pages from the swapping area to the lamu area. See also LAMU-AREA and PAGES-FROM-LAMU.

\*PAGES-TO-LAMU <first physical page>, <number of pages>,  
(<memory?>), (<image?>)

FIRST PHYSICAL PAGE:	The physical page that starts the lamu area.
NUMBER OF PAGES:	The octal number of pages to be moved from the swapping area to the lamu area.
MEMORY? and IMAGE?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*READ-BINARY**

Read binary information from a file in bpun format to a system area.

\*READ-BINARY <area>, <file name>

AREA:	Legal answers are MEMORY, IMAGE AREA or SAVE AREA.
FILE NAME:	The name of the binary file to be read into the selected area. Default file type is :BPUN.



**\*REMOVE-FROM-BACKGROUND-TABLE**

Remove a device from the background table. See the related command INSERT-IN-BACKGROUND-TABLE.

**\*REMOVE-FROM-BACKGROUND-TABLE** <logical device number>, <input/output>,  
(<memory?>), (<image?>), (<save area?>)

LOGICAL DEVICE NUMBER: | An octal logical device number.

INPUT/OUTPUT: | Select the INPUT or OUTPUT part of the  
device.

MEMORY?, IMAGE? and SAVE AREA?: | Select the areas to be modified by  
answering each question with YES or NO. The  
default answer is NO.

**\*REMOVE-FROM-EXTENDED-IDENT-TABLE**

Insert an entry in the extended ident code table of a hardware interrupt level. See also INSERT-IN-EXTENDED-IDENT-TABLE.

**\*REMOVE-FROM-EXTENDED-IDENT-TABLE** <level>, <logical device number>,  
<input/output>, <ident code>, (<memory?>),  
(<image?>), (<save area?>)

LEVEL: | The octal interrupt level.

LOGICAL DEVICE NUMBER: | An octal logical device number in the range  
1-77 or greater than 200.

INPUT/OUTPUT: | Select the INPUT or OUTPUT part of the  
device.

IDENT: | The octal ident code.

MEMORY?, IMAGE? and SAVE AREA?: | Select the areas to be modified by  
answering each question with YES or NO. The  
default answer is NO.

Remove an entry from the ident code table of a hardware interrupt level. See also the related commands INSERT-IN-IDENT-TABLE and REMOVE-FROM-EXTENDED-IDENT-TABLE.

LEVEL:	The octal interrupt level.
LOGICAL DEVICE NUMBER:	An octal logical device number in the range 1-77 or greater than 200.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.
IDENT:	The octal ident code.
MEMORY?, IMAGE? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

Make a hardware device number unavailable for @EXECUTE-IOX and EXIOX (MON 31). See also INSERT-IN-IOX-TABLE.

```

HARDWARE DEVICE NUMBER: [ An octal physical device number.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by
                             [ answering each question with YES or NO. The
                             [ default answer is NO.

```

**\*REMOVE-FROM-LOGICAL-UNIT-TABLE**

Remove a logical device number from the logical device table. See also INSERT-IN-LOGICAL-UNIT-TABLE.

```
*REMOVE-FROM-LOGICAL-UNIT-TABLE <logical device number>,
                                   <input/output>,( <memory?>),( <image?>),
                                   ( <save area?>)
```

LOGICAL DEVICE NUMBER: An octal logical device number in the range 1-77 or greater than 200.

INPUT/OUTPUT: Select the INPUT or OUTPUT part of the device.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*REMOVE-FROM-TIME-SLICE**

Cause the background process to run on a fixed priority. If a background process is in the time slicer, its priority will be modified dynamically. See also INSERT-IN-TIME-SLICE and @PRIOR.

```
*REMOVE-FROM-TIME-SLICE <logical device number>,( <memory?>),
                        ( <image?>),( <save area?>)
```

LOGICAL DEVICE NUMBER: The octal logical device number of a terminal, batch processor or communication device.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) A background process not in the time slicer will run on a fixed priority. This priority is defined by @PRIOR.

**\*REMOVE-FROM-TIMER-TABLE**

Remove a logical device number from the timer table. See also INSERT-IN-TIMER-TABLE.

\*REMOVE-FROM-TIMER-TABLE <logical device number>, <input/output>

LOGICAL DEVICE NUMBER:	An octal logical device number in the range 1-77 or greater than 200.
INPUT/OUTPUT:	Select the INPUT or OUTPUT part of the device.

**\*REMOVE-SINTRAN-COMMAND**

Remove a SINTRAN III command. File system commands cannot be removed. See also SET-COMMAND-PROTECTION.

\*REMOVE-SINTRAN-COMMAND <command>, (<memory?>), (<save area?>)

COMMAND:	The command to be disabled.
MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) A cold start is necessary to recover the command if it is removed in MEMORY. If it is removed on the SAVE AREA, SINTRAN III must be reloaded to recover it.

**\*REMOVE-SPOOLING-HEADER**

Cause the spooling program to stop printing the spooling header and trailer between each file. See also INSERT-SPOOLING-HEADER.

**\*REMOVE-SPOOLING-HEADER** <spooling index>,(**<form feed before?>**),  
(**<form feed after?>**),(**<memory?>**),  
(**<save area?>**)

SPOOLING INDEX:

The index of the spooling device as defined by the command SET-SPOOLING-DEVICE-NUMBER.

FORM FEED BEFORE? and FORM FEED AFTER?: Answer YES or NO to whether you want an empty page before and after each file is printed.

MEMORY? and SAVE AREA?:

Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*SEGMENT-WRITE-PERMIT**

Allows write access to a segment. See also SEGMENT-WRITE-PROTECT.

**\*SEGMENT-WRITE-PERMIT** <segment number>

SEGMENT NUMBER:

The octal segment number of the segment to be write permitted.

**RULES:**

- 1) The segment must have been loaded to.

**\*RESET-COLDSTART-MODE-FILE**

Remove definition of parameters for commands to be executed at cold start.

\*RESET-COLDSTART-MODE-FILE

**\*SET-COLDSTART-MODE-FILE**

Set parameters for commands to be executed at cold start.

\*SET-COLDSTART-MODE-FILE <parameters to the enter-directory command  
when entering main directory>,  
<coldstart input file>,  
<coldstart output file>

L

PARAMETERS TO THE ENTER-DIRECTORY COMMAND WHEN ENTERING MAIN DIRECTORY

This parameter contains directory name, disk name and unit and/or subunit numbers.

COLDSTART INPUT FILE: Name of mode file to be run at cold start.

COLDSTART INPUT FILE: Name of output file for mode file to be run at cold start.

**\*SEGMENT-WRITE-PROTECT**

Protect a segment against writing. See also SEGMENT-WRITE-PERMIT.

**\*SEGMENT-WRITE-PROTECT** <segment number>

SEGMENT NUMBER:

The octal segment number of the segment to be write protected.

**RULES:**

- 1) The segment must have been loaded to.

**\*SET-CHANNEL-PRIORITY**

Set the priority of an ND-NET channel.

**\*SET-CHANNEL-PRIORITY** <channel name>, <priority>, (<memory?>),  
(<image?>), (<save area?>)

CHANNEL NAME:

The peripheral file name of a communication channel.

PRIORITY:

The new octal priority in the range 0-377<sub>8</sub>.

MEMORY?, IMAGE? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*SET-CLOSED-SCRATCH-FILE-SIZE**

Set the maximum number of pages to remain in a scratch file when it is closed.

**\*SET-CLOSED-SCRATCH-FILE-SIZE** <logical device number>, <number of pages>,  
(<memory?>), (<save area?>)

LOGICAL DEVICE NUMBER:	The octal logical device number of the scratch file.
NUMBER OF PAGES:	The octal maximum number of pages to remain in the scratch file.
MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**RULES:**

- 1) The size for all scratch files is changed when the save area is modified.
- 2) The size must not be set to zero.

**\*SET-COMMAND-PROTECTION**

Change the category of users who are allowed to use a SINTRAN III command. See also REMOVE-SINTRAN-COMMAND.

**\*SET-COMMAND-PROTECTION** <command>, <protection>, (<memory?>),  
(<save area?>)

COMMAND:	Any command except file system commands.
PROTECTION:	Select a user category. PUBLIC permits the command for all users. RT permits the command for user RT and user SYSTEM. SYSTEM permits the command for user SYSTEM only.
MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.



\*SET-LAMU-CONSTANTS

Set the system constants of the lamu system, ie., the number of lamus per RT program and the total number of lamus.

```
*SET-LAMU--CONSTANTS <number of lamus per RT program>,
                    <total number of lamus>
```

NUMBER OF LAMUS PER RT PROGRAM: The number of lamus available for each  
RT program.

TOTAL NUMBER OF LAMUS:	The maximum number of lamus in the system.
------------------------	--

**RULES:**

- 1) The system needs to be restarted for the command to take effect.

\*SET-MAX-ENTER-COUNT

Set the maximum number of consecutive unsuccessful attempts a user can make to log in on a terminal. If the number is exceeded, the terminal will be disabled until the command CLEAR-ENTER-COUNT is given.

```
*SET-MAX-ENTER-COUNT <logical device number>, <enter count wanted>
                        [, <max enter count>], (<memory?>),
                        (<save area?>)
```

LOGICAL DEVICE NUMBER: The logical device number of the terminal, batch processor or communication device.

ENTER COUNT WANTED?: Answer YES or NO to whether you want to use the enter count facility.

MAX ENTER COUNT:	The maximum number of consecutive unsuccessful attempts to enter that can be made on the specified terminal.
------------------	--

MEMORY? and SAVE AREA?: Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*SET-SPOOLING-DEVICE-NUMBER**

Connect a spooling index to a logical device number. See also the command INSERT-SPOOLING-HEADER.

**\*SET-SPOOLING-DEVICE-NUMBER** <spooling index>,<logical device number>,  
(<memory?>),(<save area?>)

SPOOLING INDEX:	The octal index of the spooling device in the system. The range is determined at system generation time.
LOGICAL DEVICE NUMBER:	The octal logical device number of the printer to be used for spooling.
MEMORY? and SAVE AREA?:	Select the areas to be modified by answering each question with YES or NO. The default answer is NO.

**\*START-XMSG**

Fix the XMSG PAGING OFF and BUFFER AREA segments in physical memory and start the routine XROUT. See the SINTRAN III Communication Guide (ND-60.134). See also STOP-XMSG.

**\*START-XMSG****RULES:**

- 1) The command should be performed before starting ND-NET, preferably in the batch file LOAD-MODE:SYMB.

**\*START-GPIB**

Start a GIB controller.

\*START-GPIB-BUFFERSIZE <controller no.>

CONTROLLER NO.:		GPIB controller.
-----------------	--	------------------

**\*STOP-GPIB**

Stop a GIB controller.

\*STOP-GPIB-BUFFERSIZE <controller no.>

CONTROLLER NO.:		GPIB controller.
-----------------	--	------------------

**\*STOP-XMSG**

Disable the XMSG monitor call and release the physical memory space.

**\*STOP-XMSG**

**RULES:**

- 1) XMSG can be restarted again, provided ND-NET has not been started since XMSG was stopped.

**\*SWAP-DIRECTORY-ENTRIES**

Exchange two directories in the directory table.

\*SWAP-DIRECTORY-ENTRIES <directory index 1>, <directory index 2>,  
(<save area?>)

L  
DIRECTORY INDEX 1 and DIRECTORY INDEX 2: The index of the directories  
to be exchanged. Directory indices are listed  
by the command @LIST-DIRECTORIES-ENTERED.

SAVE AREA?: Answer YES or NO to whether the SAVE AREA  
should be affected. Default is NO.

**\*SWAPPING-LOG**

Inform on swapping activity for the system.

\*SWAPPING-LOG

Subcommands:

```
START-SWAPPING-LOG <log swapping for a specific program (default is yes)>
                   [<rt name>]
```

Start swapping-log for one specific program, or for all programs.

STOP-SWAPPING-LOG

Stop the current logging. The log may be restarted by:

RESTART-SWAPPING-LOG

READ-SWAPPING-LOG

Print the current contents of the log. This command may be given both during the log (before STOP-SWAPPING-LOG), and after.

```
SWAPPING-LOG <log swapping for a specific program (default is yes)>
[<rt name>]
```

```
<interval in seconds (default is 60 secs)>
```

Print swapping log information at specified intervals, default 60 seconds. This command must be terminated by pressing <escape>.

If either of the commands START-SWAPPING-LOG or SWAPPING-LOG has been used, they cannot be used again before the command STOP-SWAPPING-LOG is given.

EXIT

HELP

**RULES:**

- 1) The Swapping Log is an option to be ordered on the SINTRAN III order form.

---

## 13 BACKUP

---

Files can be destroyed due to software or hardware errors. User errors may also destroy files, eg., giving a @DELETE-FILE command with the wrong file name as parameter. Backup means a copy which can be used to restore lost files. The backup copy can be stored on disk, floppy disk, or magnetic tape.

How often you should take backups depends on the importance of the files. It can vary from several times a day to once a month. The system supervisor has the responsibility of taking backups of all files in the system. In addition, each user may take their own personal backups on floppy disks.

There are three different ways of performing backup. What method to use depends on your needs and the system configuration.

### BACKUP METHODS

- Using stand-alone programs.
- Using file system commands.
- Using the BACKUP SYSTEM.

Before taking a backup of the complete system, be sure your directory is not corrupted. Use the FILE SYSTEM INVESTIGATOR and execute the commands DUMP-DIRECTORY, LIST-USERS, and PAGE-LIST. See section 10.3.

A complete backup of the disk packs is called one backup generation. The number of backup generations to keep before recycling the disk packs is system dependent. You may also have different backup cycles, eg., one generation for every day, two generations for every week, and two generations for every month.

---

### 13.1 STAND-ALONE PROGRAMS FOR TAKING BACKUP

---

Stand-alone programs provide the fastest method of taking backup of hard disks. The different stand-alone programs below are available for taking backups. Use the HELP command to list the commands in these programs.

- DIMS (Disk Maintenance System) - for 5/10 Mbyte disks

DIMS has a COPY command that may be used for copying one complete cartridge disk pack to another. The command FORMAT can be used for disk formatting. The program is delivered with a user's guide.

- DISC-TEMA - for all disks except 10 Mbyte cartridge disks.
- WTOF
- FTOW

Winchester to floppy (WTOF) and Floppy to Winchester (FOTW) are for use with the ND Satellite computers for the Winchester disks.

- MCOPY
- MCOPY-TANB

MCOPY is a program for copying between disks and magnetic tapes. For example, MCOPY can copy a complete disk pack to a magnetic tape and copy it back when necessary. The program is delivered with a user's guide.

Logged in users should be informed in due time before the stand-alone program is started. Use the MAIL system or some other sort of announcement. Some customers have a fixed time of day, eg., before 8 o'clock in the morning or after 8 o'clock in the evening when backup can take place and no users are allowed to log in.

SINTRAN III must be stopped before the stand-alone backup procedure is started. Before the STOP and MCL buttons are pressed, make sure that:

- No users are logged in, also remote users.
- The batch processors are idle.
- Printers are idle.
- User RT programs are passive.

See  
section 7.1

Use the COPY command in DISK-TEMA to create the backup. Remember to test your backup for validity, using the commands VERIFY or COMPARE. VERIFY compares the contents of two disk areas by reading them both from the disk, and then comparing them word-by-word in memory.

COMPARE reads the first disk area and compares it with the second disk area. The first area is read into memory and then back to the disk interface and the second area is read to the interface. The comparison is then done on the interface, not in the memory.

Obviously, VERIFY takes more time than COMPARE. It is a good idea to do COMPARE until an error is found. Then use VERIFY on these areas. Continue with COMPARE when it is again possible.

DISC-TEMA has, in addition, the command MATCH. This means that COMPARE will be performed as long as no error is found. VERIFY takes over whenever an error is encountered, returning again to COMPARE afterwards.



---

### 13.2 SINTRAN III COMMANDS FOR TAKING BACKUP

---

Two SINTRAN III commands can be used for taking on-line backups of the complete system. The following command copies all files in the source directory onto the destination directory:

```
@COPY-DIRECTORY <destination directory name>,  
                  <source directory name>
```

The users and the file names will be the same in the destination directory as in the source directory. The destination directory should be empty when the command is given, ie., the directory should be created but no users or files should exist. The destination files will be reorganized, thus giving more contiguous space. Allocated files will be created, but not copied.

The following command copies all pages on the source device onto the destination device. It applies to devices such as disk packs, magnetic tapes and floppy disks. The destination device cannot be an entered directory.

```
@COPY-DEVICE <destinaton device name>, <source device name>
```

These two commands are only available to user SYSTEM. When taking backup of a disk pack or magnetic tape, the users of these devices should be informed, eg., using the MAIL system, so that these devices will not be used. If it is a disk pack with a main directory, then no one should use the system at all.

Make sure there is no activity on the system before starting the backup. When these commands are used on floppy disks, no specific action needs to take place.

### 13.3 THE BACKUP SYSTEM

The BACKUP SYSTEM offers a variety of facilities for copying files to and from disks, floppy disks, and magnetic tape. Backup may also be taken on remote systems when COSMOS is installed.

Some of the BACKUP SYSTEM commands have subcommands. An extensive description of all functions, commands, subcommands, and parameters are available interactively. The following example shows how you enter the BACKUP SYSTEM and how the HELP command and the question mark (?) can be used to retrieve information at all levels of the dialogue. ESCAPE is used to restart parameter collection in a command.

	@ <u>BACKUP-SYSTEM</u>
	BA-SY: ?
E	... short description of the BACKUP SYSTEM ...
X	BA-SY: <u>HELP</u>
A	COMMAND NAME:
M	... list of commands and parameters in the BACKUP SYSTEM ...
P	BA-SY: <u>COPY-USERS-FILES?</u>
L	... description of the command COPY-USERS-FILES ...
E	BA-SY: <u>COPY-USERS-FILES</u>
S	DESTINATION TYPE: ?
	... description of the needed subcommand ...
O	DESTINATION TYPE: <u>HELP</u>
F	... list of the available subcommands with parameters ...
	DESTINATION TYPE: <u>DIRECTORY?</u>
O	... description of subcommand DIRECTORY ...
N	DESTINATION TYPE: <u>DIRECTORY</u>
L	DESTINATION DIRECTORY NAME: ?
I	... description of the needed parameter ...
N	DESTINATION DIRECTORY NAME: <u>PACK-TWO</u>
E	DESTINATION USER NAME: <u>&lt;ESCAPE&gt;</u>
	LAST COMMAND IGNORED
H	DESTINATION TYPE: <u>&lt;ESCAPE&gt;</u>
E	BA-SY: <u>DESCRIBE-ALL-COMMANDS</u>
L	OUTPUT FILE: <u>LINE-PRINTER</u>
P	... output of all documentation on the line printer ...
	BA-SY: <u>EXIT</u>
	@

This manual is intended to give an overview of the commands available in the BACKUP SYSTEM. The available commands are shown in table 23. Detailed documentation is found in the SINTRAN III UTILITIES MANUAL (ND-60.151).

COMMANDS	SUBCOMMANDS	USER SYSTEM ONLY
DESCRIBE-ALL-COMMANDS	NO	NO
COPY-USERS-FILES	YES	NO
DEVICE-COPY	YES	YES
MULTIUSER-COPY	YES	COPYING TO DIRECTORY
RECREATE-FILES-AND-USERS	NO	NO
CREATE-VOLUME	NO	NO
LIST-VOLUME	NO	NO
DELETE-VOLUME-FILES	NO	NO
SERVICE-PROGRAM-CUF	YES	SOME SUBCOMMANDS
HELP	NO	NO
EXIT	NO	NO

Table 23. BACKUP SYSTEM Commands

---

### 13.3.1 COMMANDS USED FOR COPYING

---

You may copy one or more files belonging to one user by the command COPY-USERS-FILES. Files belonging to several users can be copied with the command MULTIUSER-COPY. Only user SYSTEM may use MULTIUSER-COPY to copy files to a directory.

Both commands have subcommands to describe the source and destination. The subcommands to specify the destination are DIRECTORY and VOLUME. These have their individual set of parameters, eg., the name of the files and users to be copied. The source type may be specified as DIRECTORY, VOLUME or PARAMETER-FILE.

A parameter file is a file containing the names of the files or users to be copied. Output from the commands @LIST-FILES and @LIST-USERS is accepted. Information in the parameter file not recognized as file or user names is ignored.

DESTINATION TYPE:	SOURCE TYPE:	MANUAL SELECTION:
DIRECTORY VOLUME	DIRECTORY VOLUME PARAMETER-FILE	YES NO LIST SELECT (one further level of commands)

Table 24. Subcommands in COPY-USERS-FILES and MULTIUSER-COPY

Table 24 shows the subcommands. The third subcommand is used to make a further selection of files to be copied. You answer the prompt MANUAL SELECTION: with YES, NO, LIST or SELECT. YES will cause the BACKUP SYSTEM to output the name of each file before copying. You have to confirm or skip copying by answering YES or NO for each file.

LIST and NO will copy all files without manual confirmation. LIST will output the name of all files copied. SELECT is used to enter a new subsystem to make advanced selection of files. For example, you may select all indexed files modified since the last backup was taken.

The BACKUP SYSTEM sets maximum access rights to the destination files before copying unless the destination is remote. That is, public users cannot protect destination files against themselves or against user SYSTEM by setting no write access.

The COPY-USERS-FILES and MULTIUSER-COPY commands will also copy the contents of the directory information FILE ACCESS, LAST DATE OPENED FOR READ, LAST DATE OPENED FOR WRITE, CREATION DATE and MAX BYTE POINTER from the source file to the destination file. For user SYSTEM and for those with directory access to the source, the last date OPENED FOR READ and number of times OPENED will not be updated on the source.

The BACKUP SYSTEM also has a service program mainly used to modify the function of the COPY-USERS-FILES and MULTIUSER-COPY commands. It is started from the BACKUP SYSTEM by giving SERVICE-PROGRAM-CUF. See the SINTRAN III UTILITIES MANUAL (ND-60.151) for further documentation of the subcommands available in SELECT and the SERVICE PROGRAM CUF.

---

### 13.3.2 HANDLING VOLUMES ON MAGNETIC TAPE AND FLOPPY DISKS

---

A volume has the same function for sequential storage media as a directory has for a disk. If magnetic tape is to be used as destination, the CREATE-VOLUME command should be used first. The user executing this command becomes the owner of the volume. A volume can contain files from many users. The first file on a volume may extend over several volumes.

A volume may also be created on a floppy disk. Volumes will be written in the BACKUP SYSTEM's default format. All available magnetic tape formats produced by the BACKUP SYSTEM or the old SINTRAN III command @COPY-USERS-FILES, are automatically detected when read.

Public users can only copy to and from their own volumes, or volumes owned by FLOPPY-USER. User SYSTEM has read and write access to all volumes. It is possible for user SYSTEM to modify the BACKUP SYSTEM to give all users access to any other user's volumes. The SERVICE PROGRAM CUF is used for this purpose. To enable communication with other manufacturers' computers, ANSI standard label format is available for magnetic tapes.

The device MAG-TAPE-1 unit 0 must have the peripheral file name MAG-TAPE-0, unit 1 must have the name MAG-TAPE-1, etc. To use the volume format on a floppy disk, the FLOPPY-DISC-1, unit 0 must have the name FLOPPY-1, unit 1 the name FLOPPY-2, etc. This can be set using the command @SET-PERIPHERAL-FILE.

---

### 13.3.3 COPYING A USERS FILES TO A VOLUME

---

The following is an example of how files can be copied from a disk to a volume on a floppy disk. The two files CHAPTER-ONE:TEXT and CHAPTER-TWO:TEXT are copied to the volume EXVOL. User P-HANSEN has default directory PACK-TWO. Default values are shown between slanting lines (/...../).

```
@BACKUP-SYSTEM
BA-SY: CREATE-VOLUME
VOLUME NAME: EXVOL
DEVICE NAME: FLOPPY-DISC-1
DEVICE UNIT: 0
BA-SY: COPY-USERS-FILES
DESTINATION TYPE: VOLUME
DESTINATION VOLUME NAME: EXVOL
DESTINATION DEVICE NAME: FLOPPY-DISC-1
DESTINATION DEVICE UNIT: 0
DESTINATION FILE GENERATION: 1
SOURCE TYPE: DIRECTORY
SOURCE DIRECTORY NAME /PACK-TWO/:
SOURCE USER NAME /P-HANSEN/:
SOURCE FILE NAME / /: :TEXT
MANUAL SELECTION: YES
FILE 4: (PACK-TWO:P-HANSEN)CHAPTER-ONE:TEXT;1
          INDEXED FILE 3 PAGE. MODIFIED 29/08-83 (YES/NO?) YES
FILE 5: (PACK-TWO:P-HANSEN)CHAPTER-TWO:TEXT;1
          INDEXED FILE 7 PAGE. MODIFIED 14/10-83 (YES/NO?) YES
FILE 9: (PACK-TWO:P-HANSEN)MEMO:TEXT;1
          INDEXED FILE 1 PAGE. MODIFIED 17/03-83 (YES/NO?) NO
BA-SY: EXIT
```

The copied files will have the same names on the volume as on the source directory. The copied files can be copied back to the disk by using EXVOL as source and PACK-TWO as destination.

If new backup copies of the same files will be stored on the volume later, you can use the <destination file generation> parameter. For example, the file generations can be numbered consecutively. This will help you distinguish between different generations of the same file later on. Alternatively, you could use a date identification, eg., 1125 meaning November 25th.

If the source medium is a volume, and the user given by destination user name differs from the original owner of the file, there will be a question as to which user should receive the copy, or if a new user name should be given. Answer with YES or NO.

---

### 13.3.4 COPYING ALL FILES IN A DIRECTORY

---

The following example shows how all files on a directory can be copied to a directory on another disk. The destination directory BACKUP49 has been created and entered, but is otherwise empty. Only user SYSTEM is allowed to specify directory as destination type.

```
@BACKUP-SYSTEM
BA-SY: MULTIUSER-COPY
DESTINATION TYPE: DIRECTORY
DESTINATION DIRECTORY NAME: BACKUP49
SOURCE TYPE: DIRECTORY
SOURCE DIRECTORY NAME: PACK-TWO
SOURCE USER NAME:
MANUAL USER CHECK?: NO
MANUAL SELECTION: NO
BA-SY: EXIT
```

All files will be copied since the empty user string matches all user names. Files opened for write on the source directory will not be copied. You must ensure that enough space is available for all files to be copied. The BACKUP SYSTEM will create all the necessary file names.

You may set an automatic expansion mode before copying files. In that case the destination users will be given enough disk space automatically. Nonexisting destination users will be created. See the command DESTINATION-EXPANSION in the SERVICE PROGRAM CUF.

If you are copying between directories where the destination files already exist, the source and destination date for last opened for write are checked. If the destination is written to later than the source, you will be asked whether the copying is taking place in the right direction.

Here is another example of how a complete directory can be copied:

```
@BACKUP-SYSTEM
BA-SY: DEVICE-COPY
  Destination device name: DISC-70-1
  Destination device unit: 2
  Source device name: DISC-4-70-1-F
  Destination device unit: 1.3
  Function: COPY
  Pages to copy: 34765
  Copying
  Comparing
    - OK
BA-SY: EXIT
```

The command DEVICE-COPY is restricted to user SYSTEM only. It is used to copy complete disk directories to backup disks or streamer tapes. If source and destination are both disks, at most one of them should be an entered directory. If source and destination are both disks and of the same size, destination will be identical (unless destination is the system disk). If sizes are different, one of them will (or should) be a volume. If a disk is used as volume, the other disk should be an entered directory.

Note that if subdivided disk units are used, specification of subunit is separated from unit by a period. In the example above, 1.3 means unit 1, subunit 3.



---

### 13.3.5 ADVANCED FILE SELECTION

---

The following example shows the use of a parameter file and advanced selection of files to be copied. Only user SYSTEM is allowed to give this command sequence. Version 1 of files with file type :SYMB will be copied if they have been modified since the last backup was taken. Contiguous files are specified as an exception.

```
@LIST-USERS
USER NAME:
OUTPUT FILE: "USER-LIST:SYMB"
@CREATE-FILE
FILE NAME: BACKUP-LOG:SYMB
NUMBER OF PAGES: 0
@BACKUP-SYSTEM
BA-SY: SERVICE-PROGRAM-CUF
CUF-SERV: MASTER-LOG-MODE
MASTER LOG FILE//: BACKUP-LOG:SYMB
APPEND ACCESS / YES / :
CUF-SERV: EXIT
BA-SY: MULTIUSER-COPY
DESTINATION TYPE: DIRECTORY
DESTINATION DIRECTORY NAME: BACKUP-010284
SOURCE TYPE: PARAMETER-FILE
PARAMETER FILE NAME: USER-LIST:SYMB
MANUAL USER CHECK: NO
MANUAL SELECTION: SELECT
SELECTION: FILE-NAME
FILE NAME: :SYMB;1
SELECTION: NOT FILE-ATTRIBUTE
FILE ATTRIBUTE: CONTIGUOUS
SELECTION: MODIFIED-SINCE-LAST-BACKUP
SELECTION: LIST-FILES-SELECTED
OUTPUT FILE: "BACKUP-LIST:SYMB"
SELECTION: EXECUTE
MANUAL FILE CHECK: LIST
BA-SY: EXIT
```

Different selections are specified in addition to version 1 of files with file type :SYMB. The file BACKUP-LIST:SYMB containing the names of the copied files may later be used as a parameter file for restoring the files. EXECUTE is the command to leave the SELECTION system and start copying.

The commands available in the SELECTION system are shown in table 25. The different selections can be combined by use of the logical operators AND, OR, NOT, and parentheses. Use the question mark to list explanations of the commands and operators on the terminal.

COMMANDS	COMMANDS	OPERATORS
FILE-NAME FILE-ATTRIBUTE DELETE-LAST-KEY FILE-INDEX-INTERVAL EXECUTE HELP	MODIFIED-SINCE-LAST-BACKUP WRITTEN-DATE-INTERVAL GENERATION-INTERVAL DELETE-CURRENT-SELECTION DESTINATION-FILES-EXIST LIST-SELECTION LIST-FILES-SELECTED	AND OR NOT ( )

Table 25. The BACKUP SYSTEM SELECTION Commands

### 13.3.6 MODIFYING THE COPYING MODE

The SERVICE PROGRAM CUF commands can be used to modify the function of the copying command. The following example shows some possible modifications:

```

@BACKUP-SYSTEM
BA-SY: SERVICE-PROGRAM-CUF
CUF-SERV: SET-ALLOCATE-CREATE-DEFAULT
DEFAULT ANSWER: YES
CUF-SERV: SET-VOLUME-ACCESS
GENERAL PUBLIC ACCESS: YES
CUF-SERV: MASTER-LOG-MODE
MASTER LOG FILE: BACKUP-LOG:SYMB
APPEND ACCESS: YES
CUF-SERV: DESTINATION-EXPANSION-MODE
AUTOMATIC EXPANSION: YES
CUF-SERV: DUMP-BACKUP-SYSTEM
PROG user name 'SYSTEM' : SYSTEM
CUF-SERV: EXIT
BA-SY: EXIT
@DUMP-PROGRAM-REENTRANT
NAME: BACKUP-SYSTEM-G
FILE NAME: (SYSTEMES)BACKUP-G:PROG

```

Allocated and contiguous files cannot always be created as they were described on the source directory or volume. In this case you will be asked if the files should be copied. If you answer YES, the following rules apply:

- 1) Allocated source files will be created as contiguous files if possible. Otherwise, they will be created as indexed files.
- 2) Contiguous files will be created as indexed files.

If NO is answered, such files will not be copied. The command SET-ALLOCATE-CREATE-DEFAULT can be used to give a default answer to the question. This is useful in batch and mode jobs.

The command SET-VOLUME-ACCESS lets users other than the owner access a volume. The destination user area on a directory can be too small for the source files. The command DESTINATION-EXPANSION-MODE will select a copying mode where the destination user area is expanded automatically if user SYSTEM copies to that area.

The name of the copied files, the date of copying, the destination and the source can be stored in a log file. The command MASTER-LOG-MODE is used for selecting this copying mode. The log information can either overwrite the old information in the log file or be appended to it. The log file can later be used as a parameter file. A related command, USER-COPY-LOG-MODE, is available for public users.

These modifications will not exist the next time you enter the BACKUP SYSTEM unless you give the command DUMP-BACKUP-SYSTEM. This will dump the BACKUP SYSTEM on the file BACKUP-SYSTEM:BPUN. Remember to execute the command @DUMP-REENTRANT afterwards. Only user SYSTEM may dump the system.

All commands available in the SERVICE PROGRAM CUF are listed in table 26. The commands available to user SYSTEM only are marked with an asterisk. Further information about the commands is available interactively by use of the question mark. For example, a description of the command MASTER-LOG-MODE is output by entering MASTER-LOG-MODE?.

COMMAND	SYSTEM	COMMAND	SYSTEM
MASTER-LOG-MODE	*	SET-SINGLE-SEARCH	
USER-COPY-LOG-MODE		RESET-SINGLE-SEARCH	
MODE-STANDARD-VOLUME		SET-MATCHING-MODE	
MANUAL-STANDARD-VOLUME		SHRINKING-MODE	
MODE-BACKUP-SYSTEM-VOLUME		DUMP-BACKUP-SYSTEM	*
DESTINATION-EXPANSION-MODE	*	HELP	
SET-ALLOCATE-CREATE-DEFAULT		EXIT	
SET-VOLUME-ACCESS	*		

Table 26. Commands in the SERVICE PROGRAM CUF

### 13.3.7 INCREMENTAL BACKUP

The following is an example of incremental backup. A complete backup of all files is taken each week. In addition, a backup of all modified files is taken daily. The backups are taken on disks which are kept for two weeks. See fig. 13.

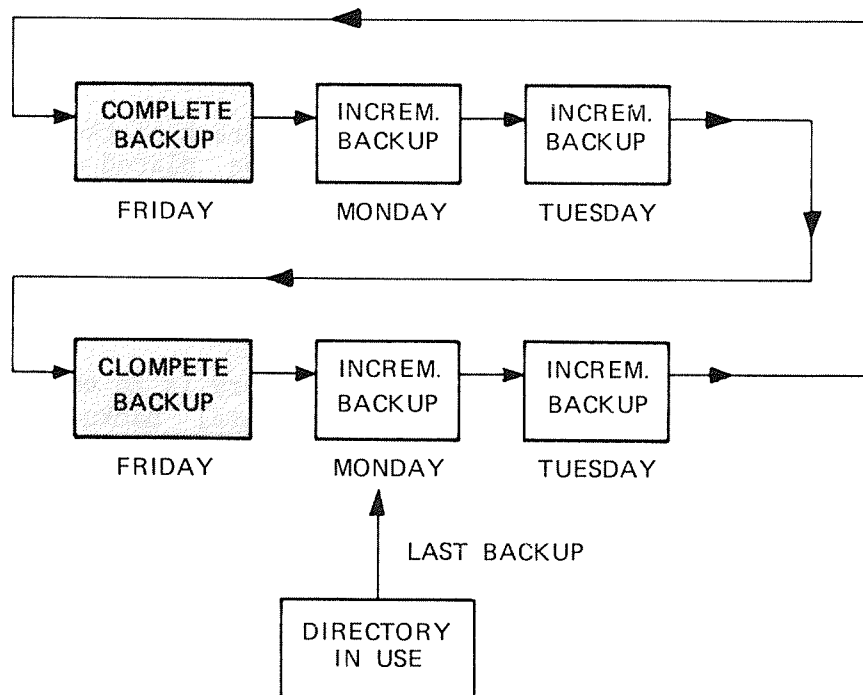


Figure 13. An Example of Incremental Backup

New directories are created both for the complete directories and each incremental directory. A log file should exist. The complete backup should use the redundant selection command:

MODIFIED-SINCE-LAST-BACKUP OR NOT MODIFIED-SINCE-LAST-BACKUP

All files will be copied. The log can be output on the terminal. The purpose of the command is to mark the files copied to the backup. The mark will be removed when the file is written to. The BACKUP SYSTEM can use the mark to select modified files for later incremental backups.

The complete backup takes some hours, eg., 2-3 hours for a 75 Mbyte directory pack. It may run simultaneously with other use of the computer. Files opened for write in the mean time will not be copied. Backup can be taken much faster by the stand-alone program DISC-TEMA. In any case, this program should not be used the first time because it will not mark the files as copied.

A new directory should be created for each incremental backup. The following selection command should be used to take a backup of the modified files:

#### MODIFIED-SINCE-LAST-BACKUP

A log file for each week must be used to keep track of the most recent backup of a file. The log file should be placed on the source directory. It should be opened by append access when taking incremental backup to accumulate the information during a week. However, the first incremental backup after a complete backup should initialize one of the log files.

The easiest way of restoring a complete directory is to use the last complete backup and copy all the incremental backups to it. The incremental backups may be copied back in reversed order. The BACKUP SYSTEM will then ask whether the copying is taking place in the right direction the first time it reads a file already restored. If the direction is confirmed, the system will ask if such files should be skipped. Thus, it is possible to restore the last version of each file only.

The previous example showed one example of incremental backup. You may save space on the incremental backups by using them several times between each complete backup. The command MODIFIED-SINCE-LAST-BACKUP creates new versions for existing destination files. Version 1 is the most recent copy of the file. Older file versions can easily be restored by a WRITTEN-DATE-INTERVAL selection.

You may also use the same incremental backups for several directories, eg., all the directories on one computer. Each user should then not have files on more than one of these directories. Files with the same names and users on different directories will be copied to different versions on the backup, and thus they will be difficult to recover.

If you are taking backup on magnetic tape, you can use the selection command LIST-SELECTED-FILES to create a list of the copied files and users. When restoring these files on an empty directory, you may use the BACKUP-SYSTEM command RECREATE-FILES-AND-USERS with the list as parameter file. This will create empty user areas and files on the directory. The selection command DESTINATION-EXPANSION should be used to give each user the appropriate space to hold the files.

---

### 13.4 PERSONAL BACKUP

---

Personal backups are usually taken on floppy disks. Directories on floppy disks can be created and entered by public users. The BACKUP SYSTEM allows public users to create volumes instead of directories on floppy disks.

SINTRAN III can handle a maximum of two floppy disk controllers with up to three units each. Before a floppy disk can be used, it must be formatted as shown in the following example:

```
@DEVICE-FUNCTION FLOPPY-1, FORMAT-FLOPPY
```

FLOPPY-1 is the peripheral file name of the floppy disk unit where the floppy disk is inserted. All ND floppy disks are formatted prior to delivery. Only one user is normally created on a floppy disk. The number of pages available on the floppy disk depends on the floppy disk system in use. The most common sizes are 148 and 616 pages.

The following example shows how a floppy disk can be used to store a personal backup. The formatted floppy disk are inserted in unit 0 of the device FLOPPY-DISK-1. The directory BACKUP-DIR and the user HANSEN is created. User HANSEN must exist in a main directory.

```
@CREATE-DIRECTORY BACKUP-DIR, FLOPPY-DISK-1, 0,,  
@ENTER-DIRECTORY BACKUP-DIR, FLOPPY-DISK-1, 0  
@CREATE-USER BACKUP-DIR:HANSEN  
@GIVE-USER-SPACE BACKUP-DIR:HANSEN, 616  
@COPY-FILE "(BACKUP-DIR:HANSEN)BOOK1:TEXT", BOOK1:TEXT  
@RELEASE-DIRECTORY BACKUP-DIR
```

A new file BOOK1:TEXT is created on the floppy disk. Remember to release the directory before you remove the floppy disk. If you want to restore the copy of BOOK1:TEST, the following command sequence should be used:

```
@ENTER-DIRECTORY BACKUP-DIR, FLOPPY-DISK-1, 0  
@COPY-FILE BOOK1:TEXT, (BACKUP-DIR:HANSEN)BOOK1:TEXT  
@RELEASE-DIRECTORY BACKUP-DIR
```

If you create a directory on a floppy disk, all previous information on the disk will be lost. The same applies to formatting. If some one has forgotten to release a directory, the floppy disk unit will be occupied. Insert an empty floppy disk, enter @RELEASE-DIRECTORY, and ignore the error message. The unit will then be free to use.

You will not be allowed to access user names on a floppy disk if they do not exist in a main directory. Use the command @RENAME-USER to change the user name on the floppy disk, eg., to your own user name or FLOPPY-USER. FLOPPY-USER was created for this purpose.







---

---

**APPENDIX A GLOSSARY**

---

This appendix will explain some important concepts and terms. An understanding of it is assumed in the other parts of this manual.

**ASCII**

Abbreviaition of American Standard Code for Information Interchange. A code connecting characters to numbers. An ASCII table is found in the SINTRAN III REFERENCE MANUAL (ND-60.128)

**Assembler**

This is a program that translates assembly language statements into machine code.

**Basic time unit**

A basic time unit is normally 20 milliseconds. The real time clock will give an interrupt every basic time unit, ie., 50 times per second.

**Batch job**

A batch job is the execution of a file which contains SINTRAN III commands and other input otherwise given from the terminal, eg., parameters, subsystem commands, and input to programs. Batch jobs are executed by a batch processor independently of any terminal.

**Baud**

The number of bits transmitted in one second. In most transmissions some of these bits are control bits.

**Block**

A block is a set of data treated as a logical unit. The size of a block may vary.

**Bootstrap program**

A program which runs in an otherwise empty computer and controls the loading of another program.

**Character-oriented device**

This is a common name for terminals, paper tape readers, card readers, line printers, etc. It signifies a device which transfers bytes to and from the computer via the A register.

**Compiler**

A computer program that translates high-level language programs into relocatable machine code, eg., BRF code.

**Data formats for ND-100**

The standard instruction set provides instructions for the following six different data formats on the ND-100:

- |                              |            |
|------------------------------|------------|
| - single bit                 |            |
| - 8-bit byte                 |            |
| - 16-bit word                |            |
| - 32-bit double word         |            |
| - 48-bit floating point word | (mutually  |
| - 32-bit floating point word | exclusive) |

**DMAC**

A special version of the MAC assembler used to debug RT programs and to perform assembly and patching on segments.

**FMAC**

A special version of MAC for assembly of code onto files and patching of machine code on files.

**High-level language**

This is a language that allows programmers to specify problem solving procedures in a notation more familiar than the computer's machine code. Some examples are Fortran, Algol, Basic, Pascal, Apl and Simula.

**Ident code**

All input/output external devices have an ident code on one or two of the interrupt levels 10, 11, 12 or 13. The ident code is used to identify a device causing an interrupt.

**Logical device number**

Each device has a logical device number which SINTRAN III uses when accessing the device.

**Logical unit number**

Same as logical device number.

**Mass storage device**

A common name for disks, magnetic tapes and floppy disks.

**MAC**

An assembler for ND-100 computers.

**MACM**

A modified MAC assembler used for system loading, i.e., mass storage assembly. When running, the stand-alone program MACM has control of the CPU and external devices.

**MACM-AREA:DATA**

A system file containing the SAVE AREA for the file system and command segments. It is an allocated file of 64 pages starting on page 100<sub>8</sub>.

**Operating system**

The operating system is the program which users need to communicate with the computer efficiently. It supports the sharing of resources. A resource may be memory, processors or different devices. The operating system for all ND computers is called SINTRAN III.

**Page**

A page is a unit of information. One page is 1 Kword = 1024 words or 2048 bytes. Both physical and virtual memory are divided into pages.

**Patch**

A program modification performed by changing the octal code or the object code, rather than changing the source code.

**Physical device number**

Each input/output external device has a physical device number. It is used in the IOX control instruction when accessing the device.

**Relocatable code**

Compiler output which must be loaded by a loader to create machine executable code. The relocatable format for ND-100 is called binary relocatable format (BRF).

**RT program**

RT programs means real time programs. These programs are terminal independent and have user-assigned priorities. They are controlled by special commands in SINTRAN III. There are both system RT programs and user RT programs. System RT programs are programs included with a SINTRAN III system. The user-written RT programs are loaded on segments by the RT LOADER.

**Segment**

A segment is a physically or virtually contiguous address area residing on mass storage. The address area contains an image of the memory. Segments in the ND-100 are placed on the segment files, eg., SEGFILO:DATA. Segments consist of an arbitrary number of contiguous 1 Kword pages. The size is limited to a maximum of 128 pages.

**Segment files**

The segment files are contiguous files on mass storage. All segments reside on the segment files. A system can have 4 segment files called SEGFILO:DATA, SEGFI1:DATA, SEGFI2:DATA and SEGFI3:DATA respectively. SINTRAN III itself will use a certain amount of space on SEGFILO:DATA; how much it uses is printed out after a cold start or initial loading.

**SINTRAN:DATA**

A system file containing the SAVE AREA for SINTRAN III RESIDENT, and the PAGING OFF AREA. It is an allocated file of 63 pages starting at page 1.

**Software**

Software is programs, compilers and procedures for a computer system, as contrasted with the hardware make-up of the machine. It includes operating systems, assemblers, compilers, subroutine libraries and user programs.

**Source program**

A source program is a set of user written instructions which can solve a problem. The source program must be compiled or assembled into machine language object code.

**Spooling**

The technique by which output to low-speed devices like printers is placed into queues on faster devices, eg., disks, while awaiting transmission to the slower devices.

**Stand-alone program**

This is a program that controls the CPU and runs the machine all by itself. A bootstrap is a few instructions that can be used to load the program without needing an operating system.

**System generation time**

System generation time is the time the operating system is assembled for a particular machine configuration. The operating system is tailored to fit a particular computer installation, for example, the number of spooling programs are defined.

**Word**

The ND-100 has a 16-bit word format. The bits are conventionally numbered 0 to 15 from right to left. The most significant bit is number 15. The ND-500 has a 32-bit format. A 16-bit word consists of two bytes of 8 bits each. The contents of an ND-100 word is conventionally represented by a 6-digit octal number. The representation of a negative number is in the 2's complement range  $-32768 \geq x \geq 32767$ . Thus, the word with all 16 bits set to zero is 000000, while the word with all bits set to one, ie., -1, is represented as 177777.

---

**APPENDIX B BINARY AND OCTAL NUMBERS**

---

This appendix explains the binary, octal and decimal number systems. Table 27 shows how to count with binary, decimal and octal numbers.

DECIMAL	BINARY		OCTAL
0	000	000	00
1	000	001	01
2	000	010	02
3	000	011	03
4	000	100	04
5	000	101	05
6	000	110	06
7	000	111	07
8	001	000	10
9	001	001	11
10	001	010	12
11	001	011	13
12	001	100	14
13	001	101	15
14	001	110	16
15	001	111	17
16	010	000	20
17	010	001	21

Table 27. Binary, Octal, and Decimal Numbers

From the above table and basic knowledge of numbers you see that:

DECIMAL NUMBER:  $135 = (1 \times 10^2) + (3 \times 10^1) + (5 \times 10^0)$

BINARY NUMBER:  $010\ 000\ 111 = 2^7 + 2^2 + 2^1 + 2^0$

OCTAL NUMBER:  $207 = (2 \times 8^2) + (0 \times 8^1) + (7 \times 8^0)$

All the above numbers represent 135 decimal.

Numbers can be converted from one number system to another as shown below:

DECIMAL TO OCTAL CONVERSION		
Problem: $135_{10} = ? \text{ octal}$		
Dividing by 8 gives the following quotients (Q) and remainder (R):		
OPERATION	Q	R
$135/8$ gives	16	7
$16/8$ gives	2	0
$2/8$ gives	0	2
The result is found from the rightmost column:		
$135_{10} = 207_8 = 010\ 000\ 111$		

OCTAL TO DECIMAL CONVERSION		
Problem: $2167_8 = ? \text{ decimal}$		
Multiplying each digit:		
$7 \times 8^0 = 7 \times 1 =$	7	
$6 \times 8^1 = 6 \times 8 =$	48	
$1 \times 8^2 = 1 \times 64 =$	64	
$2 \times 8^3 = 2 \times 512 =$	1024	
	<u>1143</u>	
The result is the sum of the multiplications.		
$2167_8 = 1143_{10}$		

The editors NOTIS WP and PED have a built-in facility to convert octal and decimal numbers. Use the home command V followed by the number to be converted. Octal numbers should be given with a trailing B, eg., 437B. Give carriage return to have the decimal and octal value output.



## APPENDIX C DEVICE NAMES

The disks, floppy disks and magnetic tape stations have the following device names and device controllers. The R ending some disk names means removable, the F means fixed with reallocated tracks, and the S means reallocated sectors.

DEVICE NAMES	MASS STORAGE DEVICE CONTROLLER
DISC-10MB-1	Cartridge disk controller 1
DISC-10MB-2	Cartridge disk controller 2
DISC-14MB-1	14-Mbyte disk controller 1
DISC-14MB-2	14-Mbyte disk controller 2
DISC-21MB-1	21-Mbyte disk controller 1
DISC-21MB-2	21-Mbyte disk controller 2
DISC-23MB-1	23-Mbyte disk controller 1
DISC-23MB-2	23-Mbyte disk controller 2
DISC-30MB-1	30-Mbyte big cartridge disk controller 1
DISC-30MB-2	30-Mbyte big cartridge disk controller 2
DISC-33MB-1	33-Mbyte disk controller 1
DISC-33MB-2	33-Mbyte disk controller 2
DISC-38MB-1	38-Mbyte disk controller 1
DISC-38MB-2	38-Mbyte disk controller 2
DISC-45MB-1	45-Mbyte disk controller 1
DISC-45MB-2	45-Mbyte disk controller 2
DISC-60MB-1	60-Mbyte disk controller 1
DISC-60MB-2	60-Mbyte disk controller 2
DISC-66MB-1	66-Mbyte disk controller 1
DISC-66MB-2	66-Mbyte disk controller 2
DISC-70MB-1	70-Mbyte disk controller 1
DISC-70MB-2	70-Mbyte disk controller 2
DISC-70MB-3	70-Mbyte disk controller 3
DISC-70MB-4	70-Mbyte disk controller 4
DISC-2-70MB-1-F	Subdivided 140-Mbyte disk controller 1
DISC-2-70MB-2-F	Subdivided 140-Mbyte disk controller 2
DISC-2-70MB-3-F	Subdivided 140-Mbyte disk controller 3
DISC-2-70MB-4-F	Subdivided 140-Mbyte disk controller 4
DISC-4-70MB-1-S	Subdivided 288-Mbyte disk controller 1
DISC-4-70MB-2-S	Subdivided 288-Mbyte disk controller 2
DISC-4-70MB-3-S	Subdivided 288-Mbyte disk controller 3
DISC-4-70MB-4-S	Subdivided 288-Mbyte disk controller 4
DISC-4-70MB-1-R	Subdivided 288-Mbyte disk controller 1
DISC-4-70MB-2-R	Subdivided 288-Mbyte disk controller 2
DISC-4-70MB-3-R	Subdivided 288-Mbyte disk controller 3
DISC-4-70MB-4-R	Subdivided 288-Mbyte disk controller 4

Continues...

DEVICE NAMES	MASS STORAGE DEVICE CONTROLLER
DISC-4-70MB-1-F	Subdivided 288-Mbyte disk controller 1
DISC-4-70MB-2-F	Subdivided 288-Mbyte disk controller 2
DISC-4-70MB-3-F	Subdivided 288-Mbyte disk controller 3
DISC-4-70MB-4-F	Subdivided 288-Mbyte disk controller 4
DISC-6-70MB-1-S	Subdivided 450-Mbyte disk controller 1
DISC-6-70MB-2-S	Subdivided 450-Mbyte disk controller 2
DISC-6-70MB-3-S	Subdivided 450-Mbyte disk controller 3
DISC-6-70MB-4-S	Subdivided 450-Mbyte disk controller 4
DISC-6-70MB-1-F	Subdivided 450-Mbyte disk controller 1
DISC-6-70MB-2-F	Subdivided 450-Mbyte disk controller 2
DISC-6-70MB-3-F	Subdivided 450-Mbyte disk controller 3
DISC-6-70MB-4-F	Subdivided 450-Mbyte disk controller 4
DISC-75MB-1	75-Mbyte disk controller 1
DISC-75MB-2	75-Mbyte disk controller 2
DISC-2-75MB-1	Subdivided 150-Mbyte disk controller 1
DISC-2-75MB-2	Subdivided 150-Mbyte disk controller 2
DISC-3-75MB-1	Subdivided 288-Mbyte disk controller 1
DISC-3-75MB-2	Subdivided 288-Mbyte disk controller 2
DISC-90MB-1	90-Mbyte disk controller 1
DISC-90MB-2	90-Mbyte disk controller 2
DISC-140MB-1-S	140-Mbyte disk controller 1
DISC-140MB-2-S	140-Mbyte disk controller 2
DISC-140MB-3-S	140-Mbyte disk controller 3
DISC-140MB-4-S	140-Mbyte disk controller 4
DISC-140MB-1-F	140-Mbyte disk controller 1
DISC-140MB-2-F	140-Mbyte disk controller 2
DISC-140MB-3-F	140-Mbyte disk controller 3
DISC-140MB-4-F	140-Mbyte disk controller 4
DISC-2-225MB-1-F	Subdivided 450-Mbyte disk controller 1
DISC-2-225MB-2-F	Subdivided 450-Mbyte disk controller 2
DISC-2-225MB-3-F	Subdivided 450-Mbyte disk controller 3
DISC-2-225MB-4-F	Subdivided 450-Mbyte disk controller 4
DISC-2-225MB-1-S	Subdivided 450-Mbyte disk controller 1
DISC-2-225MB-2-S	Subdivided 450-Mbyte disk controller 2
DISC-2-225MB-3-S	Subdivided 450-Mbyte disk controller 3
DISC-2-225MB-4-S	Subdivided 450-Mbyte disk controller 4
DISC-288MB-1-R	288-Mbyte disk controller 1
DISC-288MB-2-R	288-Mbyte disk controller 2
DISC-288MB-3-R	288-Mbyte disk controller 3
DISC-288MB-4-R	288-Mbyte disk controller 4
DISC-288MB-1-S	288-Mbyte disk controller 1
DISC-288MB-2-S	288-Mbyte disk controller 2
DISC-288MB-3-S	288-Mbyte disk controller 3
DISC-288MB-4-S	288-Mbyte disk controller 4
DISC-288MB-1-F	288-Mbyte disk controller 1
DISC-288MB-2-F	288-Mbyte disk controller 2
DISC-288MB-3-F	288-Mbyte disk controller 3
DISC-288MB-4-F	288-Mbyte disk controller 4

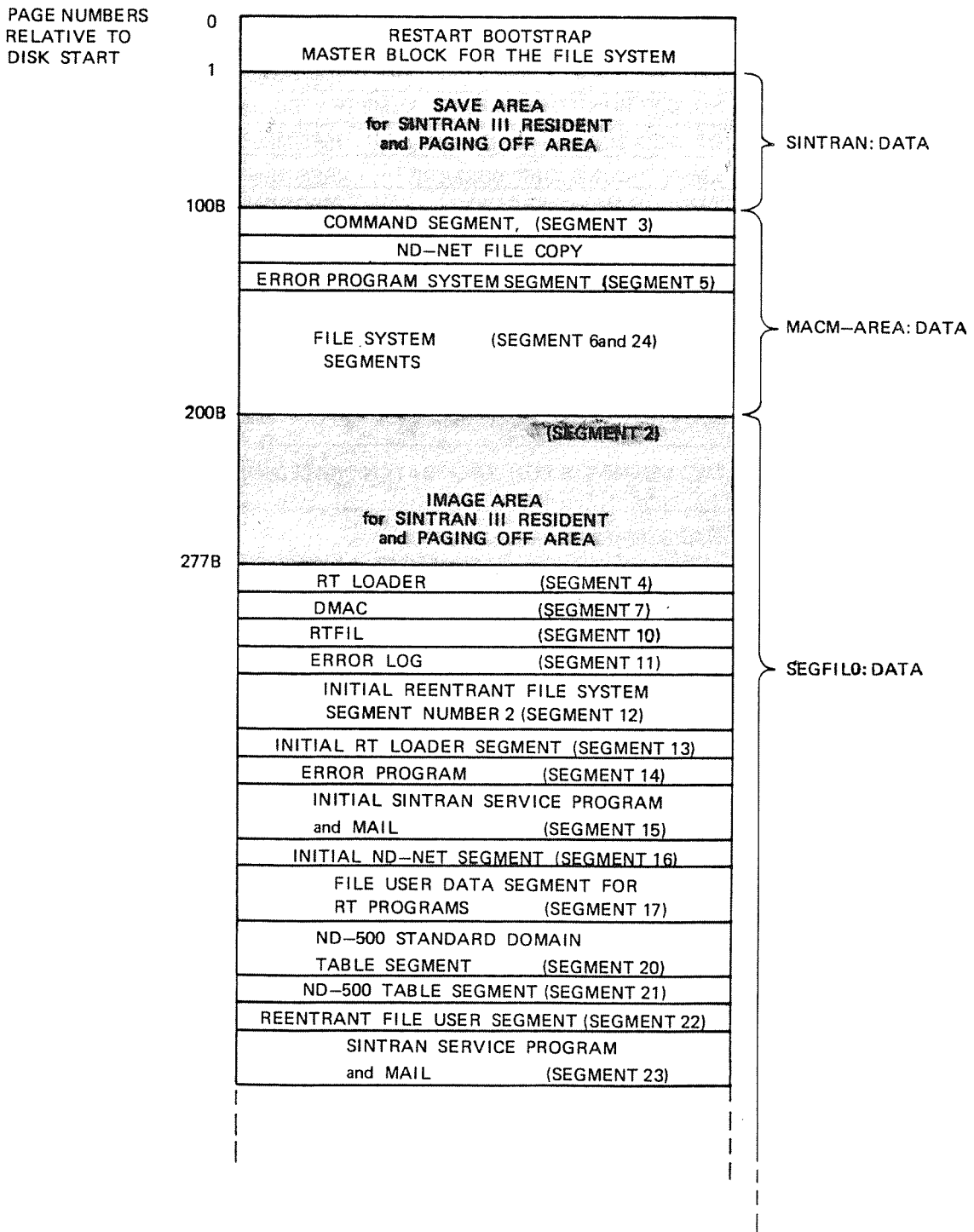
Continues

DEVICE NAMES	MASS STORAGE DEVICE CONTROLLER
DISC-45OMB-1	450-Mbyte disk controller 1
DISC-45OMB-2	450-Mbyte disk controller 2
DISC-45OMB-3	450-Mbyte disk controller 3
DISC-45OMB-4	450-Mbyte disk controller 4
MAG-TAPE-1	Magnetic tape controller 1
MAG-TAPE-2	Magnetic tape controller 2
MAG-TAPE-3	Magnetic tape controller 3
MAG-TAPE-4	Magnetic tape controller 4
FLOPPY-DISC-1	Floppy disk controller 1
FLOPPY-DISC-2	Floppy disk controller 2

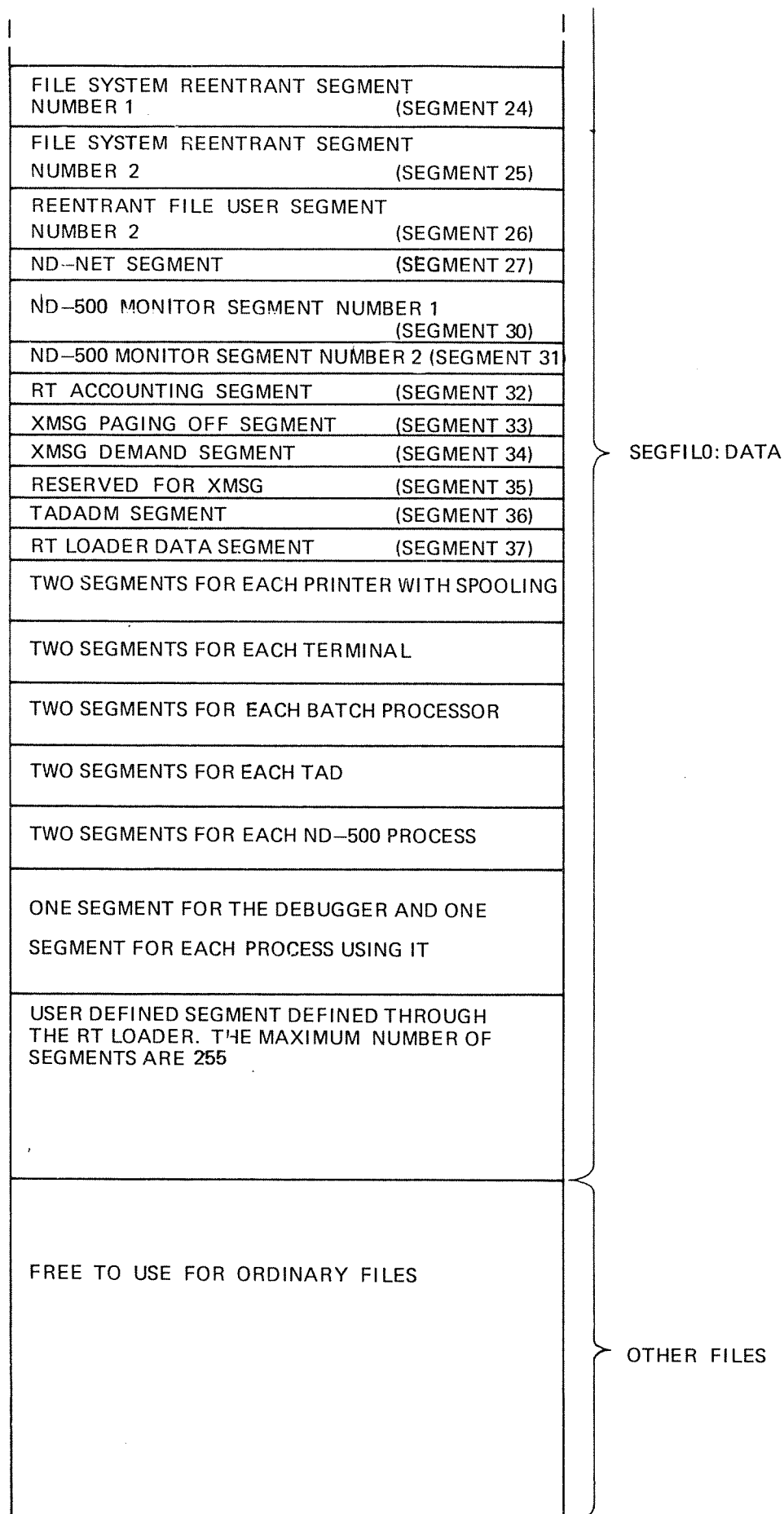


## APPENDIX D EXAMPLE OF A DISK LAYOUT

This appendix shows how the system areas and system segments are placed on a disk.

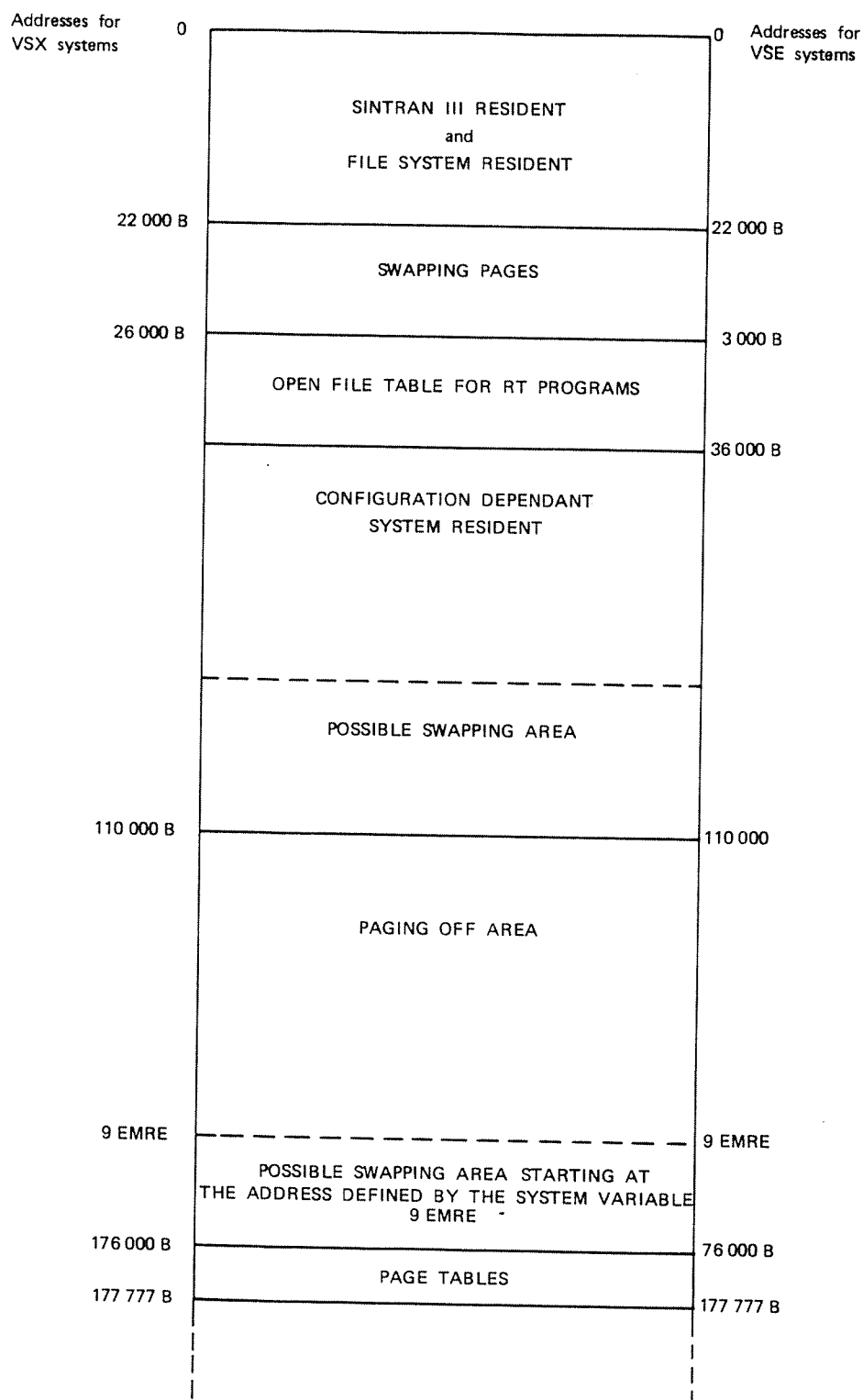


Continued on next page



## APPENDIX E SYSTEM LAYOUT OF PHYSICAL MEMORY

This appendix shows how SINTRAN III uses physical memory.

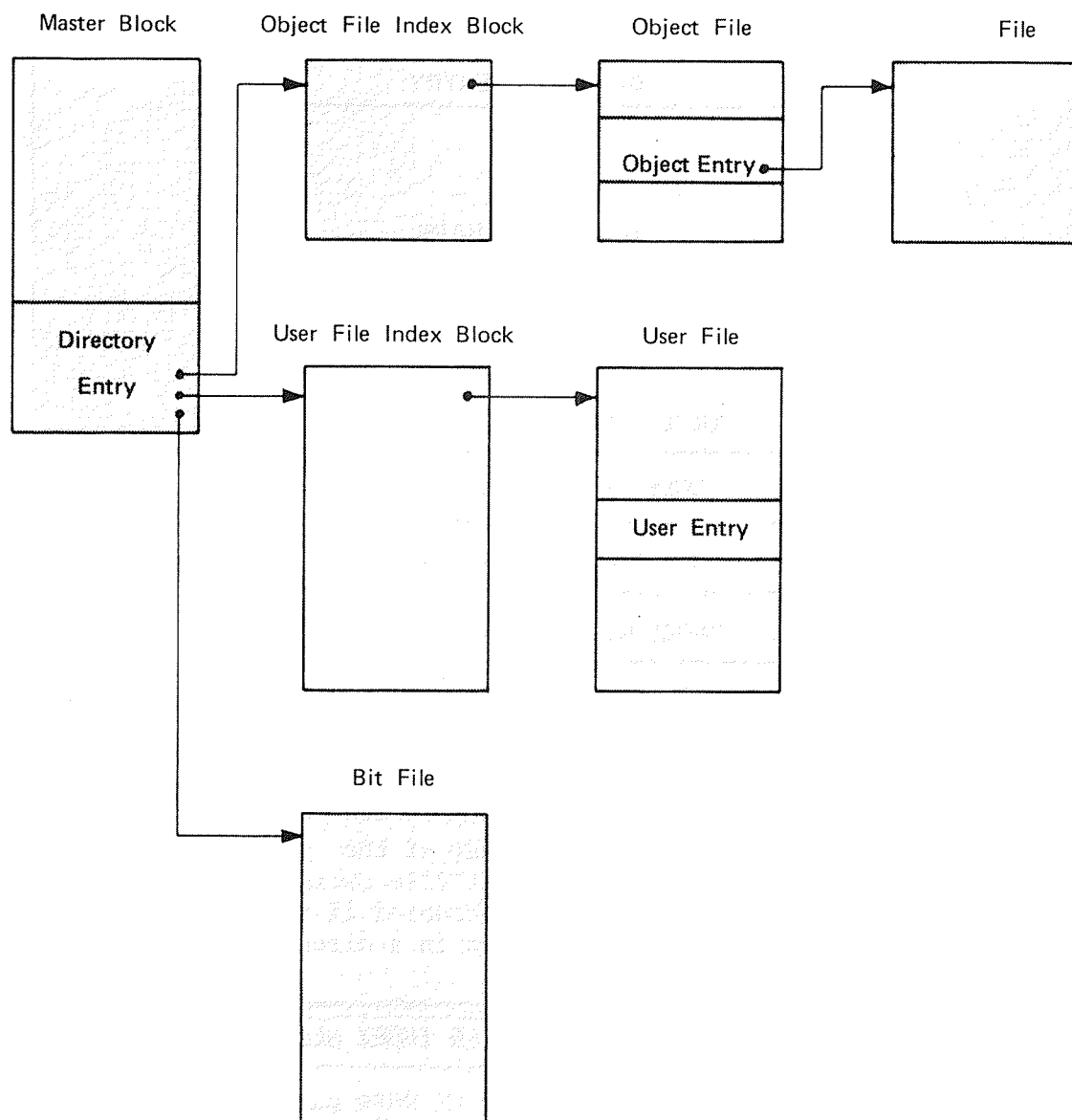






## APPENDIX F FILE SYSTEM LAYOUT ON DISK

A directory on a disk is organized by the file system as shown below. The object file index block contains pointers to object files. Each object file contains up to 32 object entries. Each object entry describes one file, eg., the file name, the owner, the size, and the file access. The user file index block contains pointers to user files which have one user entry for each user.



The system will expand itself by using subindex blocks if an index block requires more than 512 pointers. Indexed files will use one page for pointers to the pages containing data.

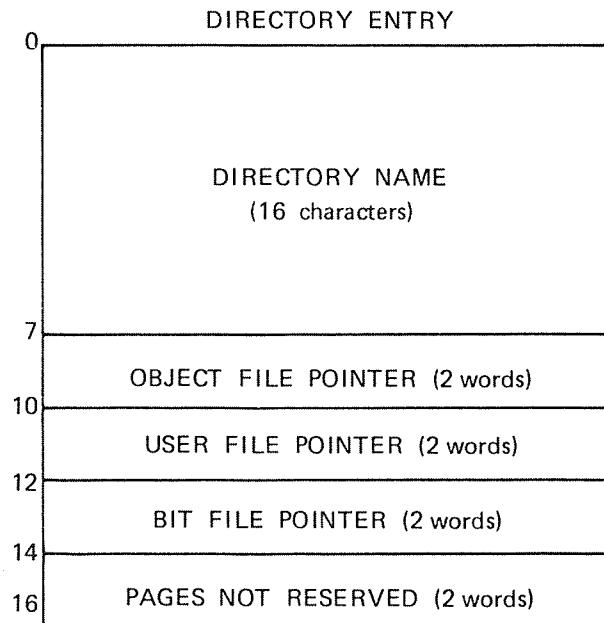
---

**F.1 MASTER BLOCK**

---

The master block is a 1 Kword block located at page 0 on a directory. Only the last part of the master block is used by the file system. The remaining part may, for example, be used for a restart program.

The last part of the master block is called the directory entry. The directory entry contains the directory name, a pointer to the bit file, a pointer to the user file index block, a pointer to the object file index block, and the number of pages not reserved for any users.



---

**F.2 BIT FILE**

---

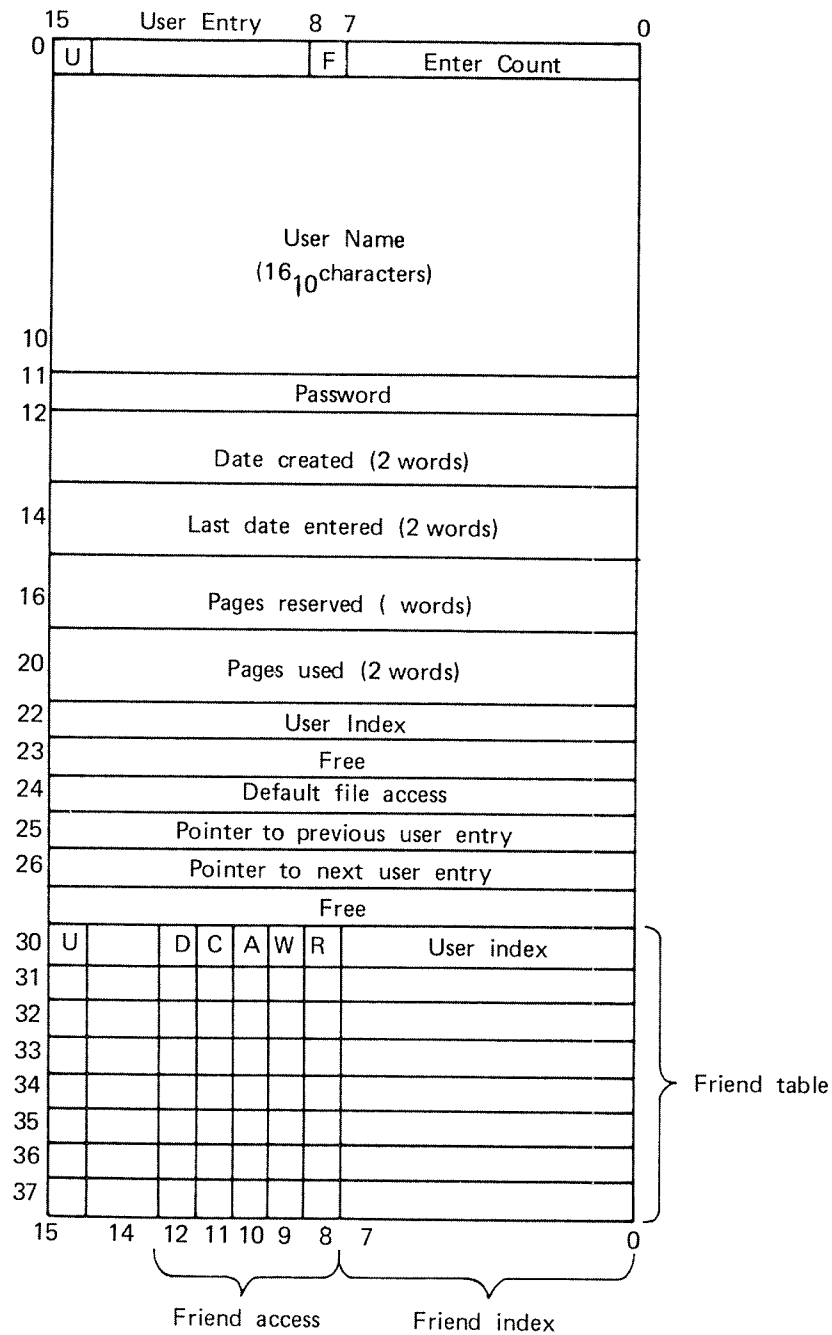
The bit file contains a free/reserved map of the pages on the disk pack or floppy disk. Each bit in the bit file corresponds to one page. The page is free if the bit is 0, and occupied if the bit is 1. There is 1 bit file page for each 16,384 pages in a directory.

---

**F.3 USER FILE INDEX BLOCK AND USER FILES**

---

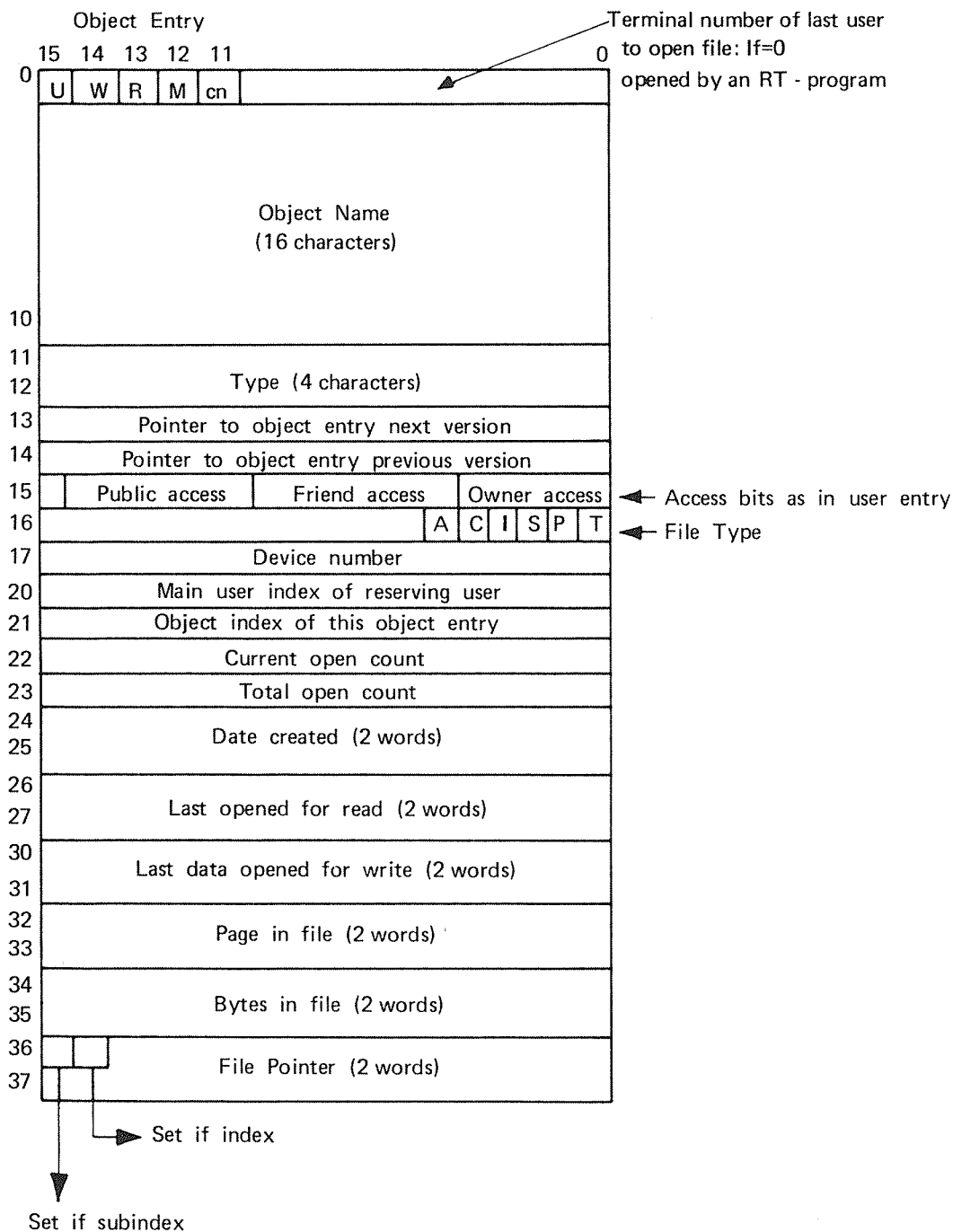
The user file index block points to one or more user files. The user files contain information about all users of the disk pack or floppy disk. Each user has one user entry. This entry contains the name of the user, a table of friends, and the different access restrictions each friend may have to the user's files. Each directory may have 256 users.



U means ENTRY USED  
D means DIRECTORY ACCESS  
C means COMMON ACCESS  
A means APPEND ACCESS  
W means WRITE ACCESS  
R means READ ACCESS  
F means USER/OBJECT ENTRY FLAG (1 FOR USER ENTRY)

**F.4 OBJECT FILE INDEX BLOCK**

The object file index block points to the object file. The object file contains information about all user's files on the disk or floppy disk. Each file has one object entry in the file. This entry contains the name and type of the object, access restrictions, file length, version pointers, and a pointer to the file. Each disk pack may have 65,536 files, divided into 256 files for each user. User 0 has object entries 0-255, user 1 object entries 256-511, etc.



U	means	ENTRY USED
W	means	CURRENTLY OPENED FOR WRITE
R	means	FILE RESERVED
M	means	FILE MODIFIED (OPENED FOR WRITE OR SPOOLING FILE IN USE)
F	means	USER/OBJECT ENTRY FLAG (0 FOR OBJECT ENTRY)
A	means	ALLOCATED FILE
C	means	CONTIGUOUS FILE
I	means	INDEXED FILE
S	means	SPOOLING FILE
P	means	PERIPHERAL FILE
T	means	TERMINAL FILE

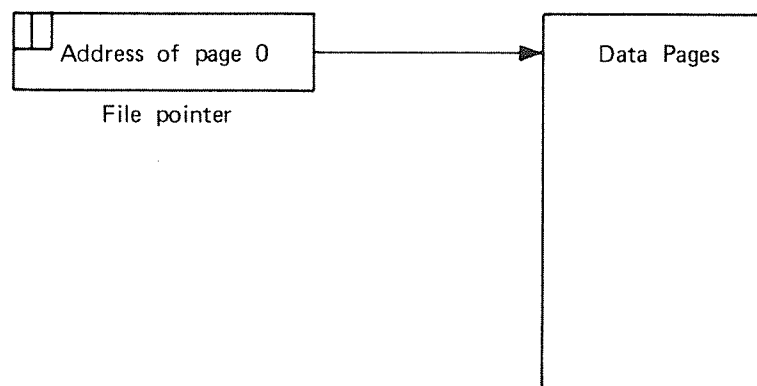
---

### F.5 USER'S FILES AND FILE LAYOUT

---

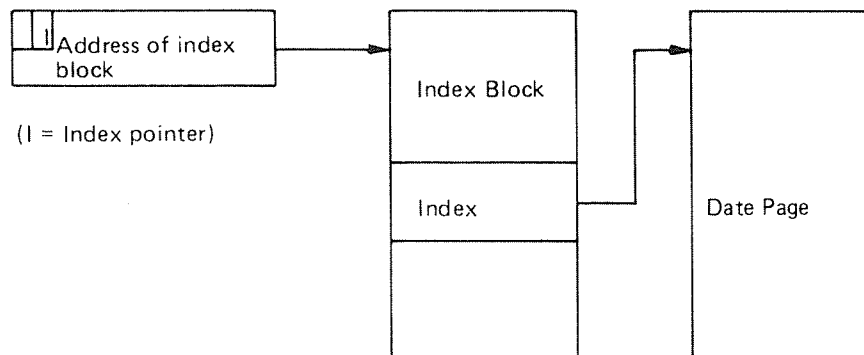
Each object entry contains information about one of a user's files. Users may access the data in the file sequentially or randomly by different monitor calls. Indexed files will be expanded automatically in write operations.

A file on disk is either organized as a contiguous or as an indexed file. A contiguous file may be of any length. It is defined by its start address and length.



Contiguous File

The pages of an indexed file are defined by a 1 Kword index block, which contains pointers to the pages of the user's file. The pages are scattered around on the disk.



Indexed File

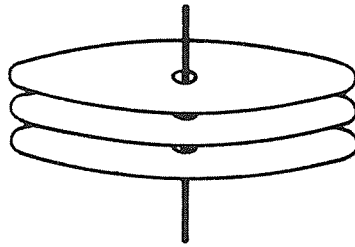
---

## APPENDIX G PHYSICAL LAYOUT OF DISKS

---

The file system supports a number of different disk types with different physical layout. The general structures, however, are common to all of them.

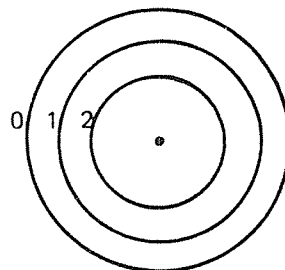
All disk packs consist of one or more disks with magnetic surfaces for recording. The disks are attached to a common axis as shown below. The disk packs are mounted on a disk unit. Disk packs are either fixed or removable.



Disk Pack

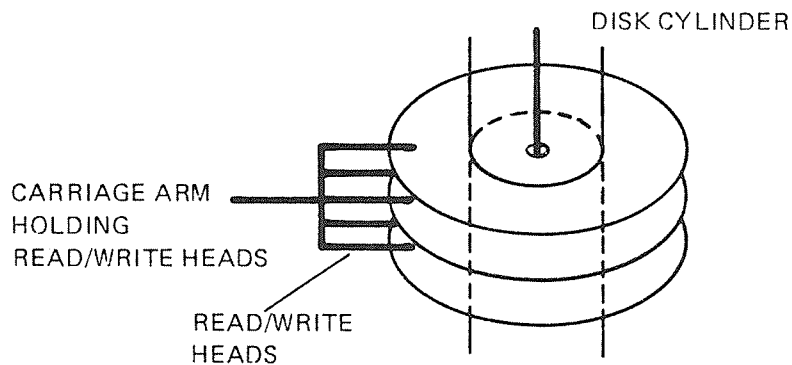
For a given disk pack, some of the surfaces are used for alignment purposes. The rest are available for data. The available surfaces are numbered from 0 and upwards. The numbering method is disk dependent.

Each surface has a number of concentric circles, called tracks. The number of tracks is disk dependent, varying between 400 and 823 for disk types used with ND computers. The tracks are numbered from 0 and upwards, starting at the outer track.



Surface with Tracks

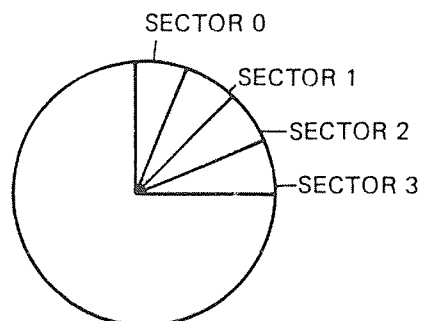
Each surface has a track number 0, a track number 1, etc. All tracks with the same number are referred to as a cylinder. Thus, all the track 0's are called cylinder 0.



Disk Cylinder

Physical disk addresses are organized by disk cylinders. The lowest disk addresses are in cylinder 0, the next in cylinder 1, etc. This reduces carriage arm movements when accessing data at subsequent disk addresses. In each cylinder, the lowest disk addresses are on surface 0, the next on surface 1, etc.

Each track is divided into sectors. The number of sectors per track is 11, 18 or 24 for ND disk systems. The sectors are numbered from 0 and upwards.



Sectors



Each sector consists of a number of 8-bit bytes. In ND disk systems this number is either 256 or 1024. The number of bytes per sector is the same for all tracks on a disk. Therefore, the tracks closer to the center of the disk have a higher density than those at the edge.

The file system operates in units of pages, ie., 1024 words. Table 28 gives the physical characteristics of the disk types supported by SINTRAN III.

Disk type	Size designation	Exact capacity in bytes	No. of data surfs.	No. of tracks/ surface	No. of sectors/ track	No. of bytes/ sector	No. of pages/ cylinder	Total capacity (pages)
HAWK	5MB	4,988,928	2/pack	406	24	256	6	2,436
SMD	33MB	33,751,040	5	412	16	1024	40	16,480
SMD	66MB	67,420,160	5	823	16	1024	40	32,920
SMD	38MB	37,969,920	5	412	18	1024	45	18,540
SMD	75MB	75,847,680	5	823	18	1024	45	37,035
CMD	30MB	30,339,072	2	823	18	1024	18	14,814
CMD	60MB	60,678,144	4	823	18	1024	36	29,628
CMD	90MB	91,017,216	6	823	18	1024	54	44,442
SMD	288MB	288,221,184	19	823	18	1024	171	140,733
RSD	80MB	75,847,680	6	823	18	1024	45	37,035
Winch. Micr.	45MB	45,121,536	6	823	9	1024	27	22,032
Winch. Fuj.	2x75MB	151,326,720	10	816	18	1024	90	73,890
FINCH	22MB	21,970,944	3	596	12	1024	18	10,728
MMD	2x75MB	151,436,720	10	821	18	1024	90	73,890

Table 28. Physical Characteristics of Various Disk Types

Each disk unit is controlled by a mass storage device controller. A mass storage device controller may control from 1 to 4 disk units. These units are numbered from 0 to 3. Only unit 0 is used unless SINTRAN III is generated with multiple units. The device name of the mass storage devices are shown in appendix C.

---

## APPENDIX H MODE FILES USED TO START UP THE SYSTEM

---

This appendix shows examples of the batch or mode files used to start up SINTRAN III. The files are system dependent.

---

### H.1 EXAMPLE OF BATCH FILE THE LOAD-MODE:SYMB

---

The batch file LOAD-MODE:SYMB is used in a warm start. This example shows the file for one particular configuration:

```
@ENTER SYSTEM PASSWORD,PROJECT PASSWORD,10
@CC *****
@CC * (SYSTEM)LOAD-MODE:SYMB    SYSTEM NUMBER 485. UPDATED 84.02.31. *
@CC *****
@CC =====
@CC = ENTER ALL DIRECTORIES AND SET THEM DEFAULT                      =
@CC =====
@ENTER-DIRECTORY,,DISC-75-1, 1
@SET-DEFAULT-DIRECTORY PACK-TWO
@CC @ENTER-DIRECTORY,,DISC-75-1, 2
@CC @SET-DEFAULT-DIRECTORY PACK-THREE
@CC =====
@CC = START XMSG AND TADADM                                           =
@CC =====
@CC @MODE (UTILITY)XMSG-START:SYMB,,
@CC @HOLD 1, 1
@CC @HOLD 5, 2
@CC @START-TADADM
@CC =====
@CC = START SPOOLING PROCESSES                                       =
@CC =====
@START-SPOOLING LINE-PRINTER
@CC @START-SPOOLING PHILIPS-1
@GIVE-SPOOLING-PAGES LINE-PRINTER, 500
@CC =====
@CC = START ACCOUNTING                                              =
@CC =====
@CC START-ACCOUNTING YES, YES
@CC =====
@CC = START BATCH NUMBER 2 AND MORE IF CONFIGURED                  =
@CC =====
@BATCH
@CC @BATCH
@CC =====
@CC = SET TERMINAL TYPES ON ACTUAL TERMINALS                        =
@CC = PHILIPS-1 ON 44B AND DIABLO ON 63B                            =
@CC =====
```

```

@CC @SET-TERMINAL-TYPE 45B,3
@CC @SET-TERMINAL-TYPE 46B,3
@CC @SET-TERMINAL-TYPE 47B,53
@CC @SET-TERMINAL-TYPE 60B,3
@CC @SET-TERMINAL-TYPE 61B,53
@CC @SET-TERMINAL-TYPE 62B,3
@CC @SET-TERMINAL-TYPE 64B,3
@CC @SET-TERMINAL-TYPE 65B,53
@CC @SET-TERMINAL-TYPE 66B,3
@CC @SET-TERMINAL-TYPE 7B,3
@CC @SET-TERMINAL-TYPE 17B,53
@CC @SET-TERMINAL-TYPE 52B,3
@CC @SET-TERMINAL-TYPE 53B,3
@CC =====
@CC = SET SYSTEM AVAILABLE FOR USERS =
@CC =====
@SET-AVAILABLE
@CC @MODE (UTILITY)COS-START-SERVER:SYMB,,,,,
@OPERATOR *** PLEASE CHECK THE CLOCK ***
@OPERATOR *** SYSTEM IS AVAILABLE ***
&&

```

---

### ***H.1.1 ADDITIONS TO THE LOAD-MODE:SYMB FILE FOR ND-500 SYSTEMS***

---

This example shows additional commands needed in the LOAD-MODE:SYMB file for ND-500 systems.

```

@ND-500-MONITOR
START-SWAPPER
EXIT

```

It is not absolutely necessary to include these commands in the LOAD-MODE:SYMB file, but it will speed up the process of starting the first user.

---

## H.2 EXAMPLE OF THE BATCH FILE HENT-MODE:SYMB

---

The batch file HENT-MODE:SYMB is used in a cold start.

```
@ENTER SYSTEM, PASSWORD, PROJECT PASSWORD, 10
@CC *****
@CC * (SYSTEM)HENT-MODE:SYMB   SYSTEM NUMBER 485. UPDATED 84.02.31. *
@CC *****
@CC =====
@CC = INITIAL COMMANDS                                     =
@CC =====
@INITIAL-COMMAND ENTER-DIRECTORY, , DISC-75MB-1,0
@NEXT-INITIAL-COMMAND CONNECT-FILE SYSTEM-OUTPUT-1, 105, W
@NEXT-INITIAL-COMMAND CLOSE-FILE, 105
@NEXT-INITIAL-COMMAND SET-ERROR-DEVICE, 1
@NEXT-INITIAL-COMMAND BATCH
@NEXT-INITIAL-COMMAND APPEND-BATCH 1, LOAD-MODE:SYMB, SYS-OUT-1
@CC =====
@CC = LOAD DMAC                                           =
@CC =====
@RTENTER
@RT-LOADER
YES
READ-BINARY DMAC, 7
YES
EXIT
@CC =====
@CC = LOAD COSMOS TERMINAL ACCESS DEVICE ADMINISTRATOR   =
@CC =====
@RT-LOADER
READ-BINARY (UTILITY)COS-TADADM, 36
YES
EXIT
@CC =====
@CC = INITIALIZE BATCH OUTPUT FILES                       =
@CC =====
@CONNECT-FILE SYSTEM-OUTPUT-1, 105, W
@CLOSE-FILE 105
@CONNECT-FILE SYSTEM-OUTPUT-2, 105, W
@CLOSE-FILE 105
@CC =====
@CC = INITIALIZE THE TERMINAL SPEED AND MAIL              =
@CC =====
@APPEND-BATCH 1, SET-TERM-SPEED, SYSTEM-OUTPUT-2
@MAIL
@RUN-MAIL
@EXIT
```

```

@CC =====
@CC = INITIALIZE ERROR LOG                                     =
@CC =====
@SET-ERROR-DEVICE 1
@INITIALIZE-ERROR-LOG
@CC =====
@CC = LOAD OF COSMOS                                           =
@CC =====
@APPEND-BATCH 1, (UTILITY)COS-FA-USER-LOAD:SYMB, SYSTEM-OUTPUT-2
@APPEND-BATCH 1, (UTILITY)COS-FSART-LOAD:SYMB, SYSTEM-OUTPUT-2
@CC =====
@CC = INITIALIZE SPOOLING FILES AND ENSURE THAT THEY ARE EMPTY. =
@CC = THIS SHOULD BE DONE FOR EACH VERSION FROM 2 AND UPWARDS  =
@CC = ON EACH PRINTER WITH SPOOLING                             =
@CC =====
@OPEN-FILE LINE-PRINTER;2, R
@OPEN-FILE LINE-PRINTER;3, R
@OPEN-FILE LINE-PRINTER;4, R
@OPEN-FILE LINE-PRINTER;5, R
@OPEN-FILE LINE-PRINTER;6, R
@OPEN-FILE LINE-PRINTER;7, R
@OPEN-FILE LINE-PRINTER;8, R
@OPEN-FILE LINE-PRINTER;9, R
@OPEN-FILE LINE-PRINTER;10, R
@CLOSE-FILE -1
@CC OPEN-FILE PHILIPS-1;2, R
@CC OPEN-FILE PHILIPS-1;3, R
@CC OPEN-FILE PHILIPS-1;4, R
@CC OPEN-FILE PHILIPS-1;5, R
@CC OPEN-FILE PHILIPS-1;6, R
@CC OPEN-FILE PHILIPS-1;7, R
@CC OPEN-FILE PHILIPS-1;8, R
@CC OPEN-FILE PHILIPS-1;9, R
@CC OPEN-FILE PHILIPS-1;10, R
@CC CLOSE-FILE -1
@CC =====
@CC = LOAD OF XMSG                                             =
@CC =====
@CC @APPEND-BATCH 1, (UTILITY)XMSG-LOAD:SYMB, SYSTEM-OUTPUT-2
@CC =====
@CC = LOAD OF REENTRANT SUBSYSTEM                               =
@CC =====
@APPEND-BATCH 1, DUMP-REENTRANT:SYMB, SYSTEM-OUTPUT-2
@CC =====
@CC = LOAD OF NOTIS WP                                         =
@CC =====
@CC APPEND-BATCH 1, WP-DUMP-K:SYMB, SYSTEM-OUTPUT-2
@CC =====
@CC = LOAD OF SIBAS                                           =
@CC =====
@CC @APPEND-BATCH 1, (SIBAS)SIB2-LOAD:BATC, SYSTEM-OUTPUT-2

```

```

@CC =====
@CC = LOAD OF ANY OTHER SUBSYSTEM =
@CC =====
@CC @APPEND-BATCH 1,<WANTED SYBSYSTEM>,SYSTEM-OUTPUT-2
@CC =====
@CC = LOAD OF SOME COMMUNICATION EMULATORS =
@CC =====
@CC @APPEND-BATCH 1,HASP-LOAD:SYMB,SYSTEM-OUTPUT-2
@CC @APPEND-BATCH 1,RLOAD-3270:SYMB,SYSTEM-OUTPUT-2
@CC @APPEND-BATCH 1,VIP-LOAD:SYMB,SYSTEM-OUTPUT-2
@CC @APPEND-BATCH 1,GRTS-LOAD:SYMB,SYSTEM-OUTPUT-2
@CC =====
@CC = EXECUTION OF LOAD-MODE:SYMB FILEN =
@CC =====
@APPEND-BATCH 1,LOAD-MODE:SYMB,SYSTEM-OUTPUT-1
@CC
@OPERATOR ***** WAIT FOR SYSTEM AVAILABLE MESSAGE *****

```

---

### **H.2.1 ADDITIONS TO THE HENT-MODE:SYMB FILE FOR ND-500 SYSTEMS**

---

This example shows the additional commands in the HENT-MODE:SYMB file of ND-500 systems.

```

@RT-LOADER
READ-BINARY (BPUN)ND-500-SEG30-F,30
YES
READ-BINARY (BPUN)ND-500-SEG31-F,31
YES
EXIT
DUMP-PROGRAM-REENTRANT ND-500-MONITOR,(BPUN)ND-500-MONITOR-F
@ND-500-MONITOR
DEFINE-MEMORY-CONFIGURATION 600
1000,YES,YES,YES
YES
DEFINE-SWAP-FILE SWAP-FILE-1:SWAP
DEFINE-STANDARD-DOMAIN LINKAGE-LOADER,LINKAGE-LOADER-F
DEFINE-STANDARD-DOMAIN FORTRAN-500,FORTAN-500-H
EXIT

```

---

**H.3 EXAMPLE OF THE BATCH FILE DUMP-REENTRANT:SYMB**

---

The batch file DUMP-REENTRANT:SYMB is used to dump subsystems reentrant after a cold start.

```

@ENTER SYSTEM,PASSWORD,PROJECT PASSWORD,10
@CC *****
@CC *(SYSTEM)DUMP-REENTRANT:SYMB SYSTEM NUMBER 658. UPDATED 84.3.10 *
@CC *****
@CC =====
@CC = DUMP OF REENTRANT SUBSYSTEMS =
@CC =====
@DUMP-REENTRANT PERFORM, 0, 0, (BPUN-FILES)PERFORM
@DUMP-REENTRANT MAC, 177777, 177775, (BPUN-FILES)MAC-1415C
@DUMP-REENTRANT NPL, 0, 1, (BPUN-FILES)NPL
@CC @DUMP-REENTRANT BASIC, 0, 1, (BPUN-FILES)BASIC
@CC @DUMP-REENTRANT FTN-2090I, 0, 1, (BPUN-FILES)FTN-2090I
@CC @DUMP-REENTRANT FORTRAN-100, 11, 11, (BPUN-FILES)FORTRAN-100
@CC @DUMP-REENTRANT DEBUGGER, 0, 0, (BPUN-FILES)DEBUGGER
@CC @DUMP-REENTRANT PLANC-100, 0, 1, (BPUN-FILES)PLANC-100
@DUMP-REENTRANT NRL, 0, 1, (BPUN-FILES)NRL
@DUMP-REENTRANT QED, 0, 1, (BPUN-FILES)QED
@DUMP-REENTRANT PED-ENG, 0, 1, (BPUN-FILES)PED-ENG
@DUMP-REENTRANT SORT-MERGE, 32252, 32252, (BPUN-FILES)SORT-MERGE
@DUMP-REENTRANT BACKUP-SYSTEM, 0, 1, (BPUN-FILES)BACKUP-SYSTEM
@DUMP-REENTRANT LOOK-FILE, 0, 0, (BPUN-FILES)LOOK-FILE
@CC =====
@CC = COSMOS PROGRAMS =
@CC =====
@CC @DUMP-PROG-REENTRANT CONNECT-TO, (UTILITY)COS-CONNECT-TO
@CC @DUMP-PROG-REENTRANT FS-ADMINISTRATOR, (UTILITY)COS-FS-ADMIN
@CC @DUMP-PROG-REENTRANT FA-SERVER-TAD, (UTILITY)COS-FA-SERV-100
@CC @DUMP-PROG-REENTRANT FILE-TRANSFER, (UTILITY)COS-FILE-TRANS
@CC =====
@CC = SIBAS-PROGRAMS =
@CC =====
@CC @DUMP-REENTRANT SIBAS-SERVICE-MH, 0, 0, (SIBAS-2D)SIBAS-SERV-MH
@CC @DUMP-REENTRANT SIBINTER-MH, 0, 0, (SIBAS-2D)SIBINTER-MH
@OPERATOR *** DUMP-REENTRANT IS FINISHED ***
&&

```



---

#### H.4 EXAMPLE OF THE MODE FILE SET-TERM-SPEED:SYMB

---

This mode file is used to set the terminal speed after a cold start.

```
@CC *****
@CC * (SYSTEM)SET-TERM-SPEED:SYMB SYSTEM NUMBER 658. UPDATED 84.3.10 *
@CC * MODE FILE TO SET THE TERMINAL SPEEDS. *
@CC * THIS FILE SHOULD BE RUN AFTER EACH TIME A NEW SINTRAN III *
@CC * HAS BEEN INSTALLED AND AFTER NEW-SYSTEM HAS BEEN RUN. *
@CC *****
@SINTRAN-SERVICE-PROGRAM
@CHANGE-DATAFIELD 45B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 46B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 47B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 60B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 61B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 62B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 64B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 65B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 66B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 7B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 17B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 52B,I,Y,Y,Y
TSPEED/210
.
@CHANGE-DATAFIELD 53B,I,Y,Y,Y
TSPEED/210
.
@REMOVE-SPOOLING-HEADER 1,N,N,Y,Y,Y
@REMOVE-SPOOLING-HEADER 2,N,N,Y,Y,Y
@EXIT
```



## APPENDIX I EXAMPLE OF A SOFTWARE SYSTEM REPORT

The following is an example of a software system report regarding the installation of NOTIS WP.

```
! NORSK DATA A.S    SOFTWARE SYSTEM REPORT    !
!
!
! DATE: 831110 ! PRODUCT NAME: NOTIS-WP version J.
!           ! PRODUCT CODE: 10079J    REPORT NO: 1 B
!
! REASON (I/M/E/F/N): E    PROGRAM: INSTALL-WP
! SUBJECT:
! Error in the installation procedure when installing NOTIS-WP
! under Sintran III version I.
!
! SYMPTOM:
!
!
!
! DESCRIPTION:
! The installation procedure is changed so that the new
! command 'NEW-BACKGROUND-SEGMENT' replaces 'NEW-SEGMENT' when
! using the RT-LOADER under Sintran III version I.
!
```

Enter the floppy labelled ND-10079J-1 and enter an editor with the file '(ND-10079J-1)INSTALL-WP:COM' and do the following changes (in the correct order!).

- 1) Change line 622 to:  
^WRITE WPD NEW<BACKGR>-SEGMENT <SEG2>,,DM,,
- 2) Change line 616 to:  
^WRITE WPD NEW<BACKGR>-SEGMENT <SEG1>,,DM,,
- 3) Insert this line above line 51 (above '^I\_VERSION:'):  
^BACKGR:==''
- 4) Insert this line above line 48(above '^GOTO I\_VERSION:')  
^BACKGR:=='-BACKGROUND'

After doing these changes, write the file back to the floppy and perform a complete installation by answering I to the question :

> What do you want to do , Install or Change (I or C) ?  
after starting the installation program.



## APPENDIX J EXAMPLE OF AN ND SOFTWARE LIBRARY DESCRIPTION

The following is a software library description for the subsystem FILE HANDLER.

NORSK DATA A/S		NORD SOFTWARE LIBRARY PROGRAM DESCRIPTION		PAGE 1 OF 2	
PRODUCT	NAME FILE HANDLER			ND-NUMBER 10516A	CATEGORY STPR
				ND-NUMBER FOR SOURCE 10517A	
ISSUED	DATE 83.06.29			BY (INITIALS) PÅK	
COMPUTERS	X 10	. 12	. 50	X 100	. 500 . ...
INSTR.SET	48 BIT FL.		32 BIT FL.		(X)COMMERCIAL
OP.SYSTEM	X SIN III VS		. SIN III RT		. ALONE . . . . .
DOCUMENTATION	TITLE: FILE HANDLER USER'S MANUAL				NUMBER: 60.175.01
PURPOSE	UTILITY PROGRAM				
PROGRAMS {FILES}	PROG.NUMB.	NAME	TYPE CONTAINING		
	204196A	FILE-HANDLER	BPUN FILE-HANDLER		
	204197A	FILE-HANDLER	PROG FILE-HANDLER		
	204198A	FILE-HANDLER	HELP DOCUMENTATION		
	204199A	EMPLOYEES	DATA EXAMPLE		
	204200A	IN-DES	DATA EXAMPLE		
	204201A	SALES	DATA EXAMPLE		

FILE-HANDLER can be used to retrieve and reformat data records from large data file(s), to generate test files, to produce a report or a smaller file that can be used in further data processing.

The user can specify extraction criteria and various output records and report layout by interacting with the program or by running a mode-file (which may be generated by FILE-HANDLER).

A split file option exists whereby data records not satisfying the selection criteria can be placed on a secondary output file.

The program can do simultaneous sorting and output selection and reporting.

The FILE-HANDLER system consist of 6 files.

- FILE-HANDLER:BPUN
- FILE-HANDLER:PROG
- FILE-HANDLER:HELP
 

This file must be copied to user SYSTEM with the same name as above. FILE HANDLER:HELP contains the on-line help information.
- EMPLOYEES:DATA

NORSK DATA A/S		NORD SOFTWARE LIBRARY PROGRAM DESCRIPTION		PAGE 2 OF 2	
PRODUCT	NAME FILE HANDLER	ND-NUMBER 10516A	CATEGORY STPR		
		ND-NUMBER FOR SOURCE 10517A			

- SALES:DATA  
These files contain test data which make it possible to run the examples in the File Handler user's manual (60.175.01).

- IN-DESC:DATA  
This file contains the input record description for the file SALES:DATA. (Ref. File Handler User's Manual, page 18).

1 LOADING

Dump the FILE-HANDLER reentrant ( NB! The FILE-HANDLER:PROG must be copied to user SYSTEM, even if the program is dumped reentrant).

@DUMP-REENTRANT FILE-HANDLER 47474 47474 (ND:FL)FILE-HANDLER:BPUN

Then you have to use the following SINTRAN-III command for all the terminals from which you want to run FILE-HANDLER.

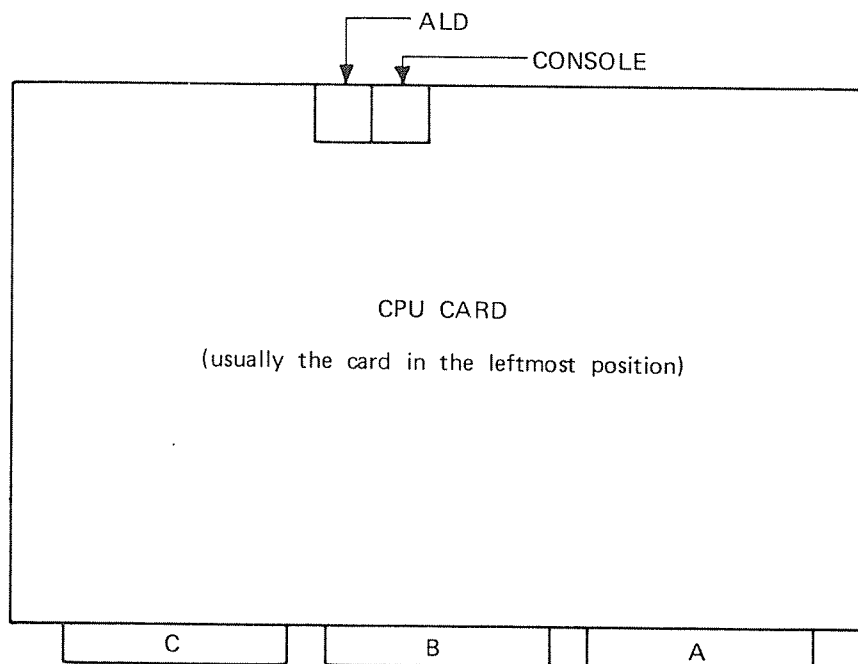
@CHANGE-BACKGROUND-SEGMENT-SIZE <terminal-nr> 128

2 NOTES

- You must use version H or later of SINTRAN-III.
- ON a NORD-10/NORD-10/S the commercial-instruction-set must be installed in order to run the program.
- If possible you should use terminal type 53 or 55 when running File Handler.
- FILE HANDLER will produce 'COMPLETION-CODE' for later testing by JEC (job executing control) which is to be released with the I-version of SINTRAN-III.

## APPENDIX K ND-100 SWITCH SETTINGS

The automatic load descriptor (ALD) switch is found on the ND-100 CPU card as shown below. You will find the CPU card in the leftmost crate, ie., crate number 1. The setting of this thumbwheel switch defines the function of the LOAD button on the computer's panel.

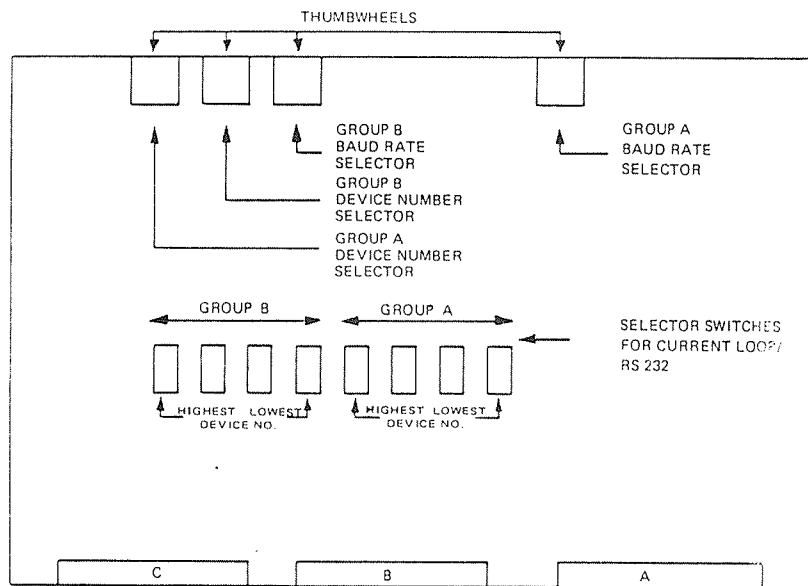


### K.1 THE AUTOMATIC LOAD DESCRIPTOR SETTINGS

The function of the automatic load descriptor settings:

ALD	I12	LOCK and Standby Power OK	LOCK and UNLOCK Standby Power not OK and Load
2	101600	Binary load from 1600	Binary load from 1600
3	100400	Binary load from 400	Binary load from 400
4	121540	Mass load from 1540	Mass load from 1540
5	120500	Mass load from 500	Mass load from 500
6	101560	Binary load from 1560	Binary load from 1560
7	100000	STOP	STOP
8-9			
10	1600	Start in address 20	Binary load from 1600
11	400	Start in address 20	Binary load from 400
12	21540	Start in address 20	Mass load from 1540
13	20500	Start in address 20	Mass load from 500
14	1560	Start in address 20	Mass load from 1560
15	0	Start in address 20	STOP

The thumbwheel switches on 8-terminals interface cards (3013) are shown below. The switches are used to set the baud rate and the physical device numbers of the terminals.



## K.2 PHYSICAL AND LOGICAL DEVICE NUMBERS FOR TERMINALS

Terminals are divided into groups of four terminals. An 8-terminal card includes two groups, A and B. Each of these groups consists of four terminals according to the table below:

Terminal	Terminal group	Physical device no.	Logical device numbers		Background process
			octal	decimal	
1	0	300	1	1	BAK01
2		310	11	9	BAK02
3		320	42	34	BAK03
4		330	43	35	BAK04
5	1	340	44	36	BAK05
6		350	45	37	BAK06
7		360	46	38	BAK07
8		370	47	39	BAK08
9	2	1300	60	48	BAK09
10		1310	61	49	BAK10
11		1320	62	50	BAK11
12		1330	63	51	BAK12
13	3	1340	64	52	BAK13
14		1350	65	53	BAK14
15		1360	66	54	BAK15
16		1370	67	55	BAK16



Terminal	Terminal group	Physical device no.	Logical device numbers		Background process
			octal	decimal	
17	12	200	7	7	BAK17
18		210	17	15	BAK18
19		220	52	42	BAK19
20		230	53	43	BAK20
21	13	240	54	44	BAK21
22		250	55	45	BAK22
23		260	56	46	BAK23
24		270	57	47	BAK24
25	14	1200	70	56	BAK25
26		1210	71	57	BAK26
27		1220	72	58	BAK27
28		1230	73	59	BAK28
29	15	1240	74	60	BAK29
30		1250	75	61	BAK30
31		1260	76	62	BAK31
32		1270	77	63	BAK32
33	4	640	1040	544	BAK33
34		650	1041	545	BAK34
35		660	1042	546	BAK35
36		670	1043	547	BAK36
37	5	1100	1044	548	BAK37
38		1110	1045	549	BAK38
39		1120	1046	550	BAK39
40		1130	1047	551	BAK40
41	6	1140	1050	552	BAK41
42		1150	1051	553	BAK42
43		1160	1052	554	BAK43
44		1170	1053	555	BAK44
45	7	1400	1054	556	BAK45
46		1410	1055	557	BAK46
47		1420	1056	558	BAK47
48		1430	1057	559	BAK48
49	8	1500	1060	560	BAK49
50		1510	1061	561	BAK50
51		1520	1062	562	BAK51
52		1530	1063	563	BAK52
53	9	1640	1064	564	BAK53
54		1650	1065	565	BAK54
55		1660	1066	566	BAK55
56		1670	1067	567	BAK56

Terminal	Terminal group	Physical device no.	Logical device numbers		Background process
			octal	decimal	
57	10	1700	1070	568	BAK57
58		1710	1071	569	BAK58
59		1720	1072	570	BAK59
60		1730	1073	571	BAK60
61	11	1740	1074	572	BAK61
62		1750	1075	573	BAK62
63		1760	1076	574	BAK63
64		1770	1077	575	BAK64

---

### ***K.3 INITIAL BAUD RATE FOR TERMINALS***

---

The thumbwheel positions for the baud rate have the following meaning:

0 = 110 baud	8 = 2400 baud
1 = 150 baud	9 = 600 baud
2 = 300 baud	10 = 200 baud
3 = 2400 baud	11 = 134.5 baud
4 = 1200 baud	12 = 75 baud
5 = 1800 baud	13 = 50 baud
6 = 4800 baud	14 = unused
7 = 9600 baud	15 = unused

Selector switches for current loop/RS232-C (V.24) are:

Switch set to 0 selects current loop.  
Switch set to 1 selects RS232-C (V.24).

See the ND-100 Reference Manual (ND-06.014) for further information about the ND-100 switch settings.

APPENDIX L SINTRAN III ORDER FORM

This appendix shows the two pages of the SINTRAN III order form. The form is needed to generate a SINTRAN III system for a particular installation.

SINTRAN III ORDER												PAGE 1	
CUSTOMER:						CPU NO:			DATE:				
CUSTOMER REP. :						PHONE:							
CUSTOMER NOTES:									MAINT. :				
VS:	VSE:	VSX:	VSE/500:	VSX/500:	ISSUED BY:	DUMPED:	(1)	LISTING:	(2)				

MASS STORAGE															
CONT\UNIT	10MB	33MB	66MB	38MB	75MB	288MB	30MB	60MB	90MB	3X75MB	2X75MB	14MB	21MB		
1\															
2\															
	MAG-TAPES			FLOPPY DISK											
	HP	PERT	STC	SMALL	BIG										
1\						SYSTEM UNIT :								23MB	45MB
2\															

TERMINALS (3)																							
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
B																							
25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
121	122	123	124	125	126	127	128																

INPUT/OUTPUT DEVICES											
TAPE READER	TAPE PUNCH	CARD READER	CARD PUNCH	CALCOMP STANDARD	CALCOMP BUFFERED	TEKTRO 613	XON/XOFF	SYNC. (4 MODEM)	CAMAC	ACH	UNI.DMA. ORI. F16
DR11C INPUT	DR11C OUTPUT	NO OF UNIV. DMA INTERF.		HDLC DMA (4)	ARIS. DIG.	HDLC (4 SYN.MOD)	X21 DRIV.				

(1) B: ON BIG FLOPPY S: ON SMALL FLOPPY (2) C: COMPLETE LISTING F: ON FLOPPY S: SYMBOL LISTING P: ON PAPER N: NO LISTING (3) B: TERM. WITH BACKGROUND R: TERM. WITHOUT BACKGROUND S: PRINTER TERM. WITH SPOOLING P: PRINTER TERM. WITHOUT SPOOLING T: TEXT EDITING (TERM. 16-2) F: PHOTOCSETTER (TERM. 29-31) H: HARRIS PHOTOCSETTER (TERM. 29-30) G: GRAF CASSETTE (TERM. 27-28) X: RESERVED FOR NO-NET (4) EXCLUSIVE OF NO-NET CONNECTIONS (5) GIVE OCTAL NUMBERS	DELIVERY	MAIL :	LOCAL PICKUP :
	SOFTWARE NOTES		VERS GEN BY DATE

SINTRAN III ORDER															PAGE 2						
<u>LINE-PRINTERS AND OTHER I/O</u>																					
LINE- PRINTER	VERSATEC DMA I/O	COSMOS SPOOLING	DEVICE NUMBER FOR SPOOLING (5)																		
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15				
DIGITAL IN	OUT	ANALOG INPUT	PROCESS I/O DRIVER																		
			ANALOG DIGITAL																		
<u>SYSTEM PARAMETERS</u>																					
RT PROG. -24-	SEGM. -40-	SEMA. -5-	INTERN. CHAR. -2-	DEVICE BLOCK	BATCH PROC. -1-	FIXC MON- 160*	DIR. TASK	CONN. DATAF.	NORD 50	SIBAS 1-3 1-6 1-12			TPS	HASP DMA							
IBM	REMOTE JOB HONEYWELL	ENTRIES UNIVAC	CDC	MEAS	CORE COM.	DEV. BUF. 2	MAG-TAPE FROM D-T	DIR. TRAN. ON MAG-TAPE	NO. OF RT FROM D-T			NOT 128K BACKGR.	DRIVE ND023								
TELIN/ TELENT	INSTR/* OUTSTR. MON-161	READ* SEG. MON-53	ND-500	ND-500 PROC.	FIXC5 -5- * MON-61	ACCOUNTING RT I/O	BUS EXT.	SREENT MON-212	SYMBOL. DEBUG. MON-204	XMSG	FILACC SEGM.	CONNECT TO									
TAD	NO ECHO BREAK 7	EXT. OP.FI	MPM-4	PIOC	CX CPU	LOG IN	EXT. ADD.														
<u>ND-NET</u>																					
LINE NO	ASYN	SYNC MODEM FULL DPX. HALF DPX.		H D L C DMA I/O		CHANNELS															
						1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1																					
2																					
3																					
4																					
5																					
6																					
7																					
8																					
9																					
					POOL :																
* : DEFAULT ON ND-500 SYSTEMS 2 : DEFAULT 20 DEVICE BUFFERS																					



Index

100-ERROR test program . . . . .	170.
7ENDC . . . . .	154, 182.
9EXIT . . . . .	180.
AACCOUNTS:DATA . . . . .	48.
AACCOUNTS:ISAM . . . . .	48.
ABORT . . . . .	77, 155.
ABORT-BATCH . . . . .	37, 77.
ABORT-PRINT . . . . .	72.
ABORT-PROCESS . . . . .	114.
ABORTED BY SYSTEM . . . . .	77.
accounting . . . . .	47.
tables . . . . .	54.
ACCOUNTING SERVICE PROGRAM . . . . .	48, 54.
ACCOUNTS:DATA . . . . .	48.
ACCRT:BPUN . . . . .	54.
ACCTAB . . . . .	52.
ACTLEV . . . . .	153.
ADJUST-ACCOUNTING-FILE . . . . .	59.
ALD . . . . .	80, 281.
ALLOCATE-FILE . . . . .	15, 100.
allocated file . . . . .	15.
ALREADY FIXED . . . . .	139.
ALT-MEMORY . . . . .	178.
ALT-SEGMENT . . . . .	178.
APPEND-BATCH . . . . .	37.
APPEND-SPOOLING-FILE . . . . .	73.
ASCII . . . . .	245.
ASCII-DUMP . . . . .	189.
assembler . . . . .	245.
automatic load . . . . .	
decription . . . . .	281.
descriptor . . . . .	80.
B14-200 . . . . .	153.
background . . . . .	
process . . . . .	85, 92, 101, 118.
segment . . . . .	85, 102.
segment size . . . . .	85.
table . . . . .	187, 205, 214.
BACKSPACE-PRINT . . . . .	72.
BACKUP . . . . .	225.
generation . . . . .	225.
SYSTEM . . . . .	115, 229.
bad page . . . . .	21.
BAK01 . . . . .	118.
BAKnn . . . . .	118, 129.
basic time unit . . . . .	245.
batch . . . . .	37, 76.
job . . . . .	245.
process . . . . .	118.
processor . . . . .	37, 77, 85.
stop . . . . .	77.

supervisor . . . . .	195.
system . . . . .	37.
BATCH INPUT ERROR . . . . .	134.
BATCH OUTPUT ERROR . . . . .	134.
BATCH SYSTEM ERROR . . . . .	134.
baud . . . . .	245.
BCHnn . . . . .	118.
BEX PARITY ERROR . . . . .	145.
BIG CARTRIDGE DISK . . . . .	10.
binary number . . . . .	249.
bit file . . . . .	21, 259, 260.
bit file error . . . . .	169.
BKnnn . . . . .	118.
block . . . . .	245.
bootstrap . . . . .	39.
program . . . . .	245.
bpun format . . . . .	107.
BPUN-FILES . . . . .	31, 44, 111.
bpun format . . . . .	160.
broadcast . . . . .	41, 42.
example . . . . .	76.
number . . . . .	43.
buffer size . . . . .	190.
Bulletin . . . . .	2.
CACHE test program . . . . .	170.
CC . . . . .	190.
CHANGE-BACKGROUND-SEGMENT-SIZE . . . . .	85.
CHANGE-BIT-FILE . . . . .	163.
CHANGE-BUFFER-SIZE . . . . .	190.
CHANGE-DATAFIELD . . . . .	181, 191.
CHANGE-DIRECTORY-ENTRY . . . . .	26, 163.
CHANGE-OBJECT-ENTRY . . . . .	19, 163.
CHANGE-PAGE . . . . .	163.
CHANGE-PASSWORD . . . . .	3, 36, 104.
CHANGE-USER-ENTRY . . . . .	163.
CHANGE-VARIABLE . . . . .	182, 192.
channel . . . . .	38.
character-oriented device . . . . .	245.
clear device . . . . .	64.
CLEAR-DEFAULT-DIRECTORY . . . . .	25.
CLEAR-DEVICE . . . . .	64.
CLEAR-ENTER-COUNT . . . . .	194.
CLEAR-MAIN-DIRECTORY . . . . .	25.
CLEAR-PASSWORD . . . . .	36.
CLOAD . . . . .	180.
clock . . . . .	100.
test . . . . .	170.
cold	
start . . . . .	75, 83, 107.
start overview . . . . .	87.
COLD-START . . . . .	75, 83.
command	
category . . . . .	220.

protection . . . . .	220.
remove . . . . .	217.
table . . . . .	44.
comment . . . . .	190.
communication . . . . .	38.
COMPARE . . . . .	227.
compiler . . . . .	246.
complete	
backup . . . . .	239.
file name . . . . .	13.
CONFIGURATION INVESTIGATOR . . . . .	173.
console . . . . .	3.
contiguous file . . . . .	15, 263.
CONTINUE . . . . .	44.
control	
panel . . . . .	5.
store . . . . .	114.
CONTROL-SEGMENTS:SYMB . . . . .	107.
CONTROL-STORE:DATA . . . . .	111.
converting octal numbers . . . . .	250.
COPY . . . . .	16, 62, 115.
COPY-DEVICE . . . . .	228.
COPY-DIRECTORY . . . . .	228.
COPY-DOMAIN . . . . .	115.
COPY-FILE . . . . .	44, 111, 115.
CORRECTED MEMORY ERROR . . . . .	144.
COSMOS . . . . .	40.
CPU test program . . . . .	170.
create user . . . . .	34.
CREATE-BACKGROUND-PROJECT . . . . .	56.
CREATE-DIRECTORY . . . . .	21, 22, 241.
CREATE-FILE . . . . .	15.
CREATE-LAMU . . . . .	194.
CREATE-NEW-VERSION . . . . .	65, 103.
CREATE-RT-PROJECT . . . . .	57.
CREATE-USER . . . . .	34, 241.
CREATE-VOLUME . . . . .	233.
creating files . . . . .	15.
CTRL Q . . . . .	155.
CTRL S . . . . .	155.
cylinder . . . . .	266.
data	
field . . . . .	191.
fields . . . . .	187.
formats . . . . .	246.
data communication . . . . .	38.
DATE-INTERVAL . . . . .	236.
decimal number . . . . .	249.
default directory . . . . .	20, 24, 25.
DEFINE-BATCH-SUPERVISOR . . . . .	195.
DEFINE-ESCAPE-CHARACTER . . . . .	156.
DEFINE-HISTOGRAM . . . . .	123.



DEFINE-MEMORY-CONFIGURATION . . . . .	113.
DEFINE-PROMPT-STRING . . . . .	195.
DEFINE-RTCOMMON-SIZE . . . . .	196.
DEFINE-SEGMENT-FILE . . . . .	93, 102, 197.
DEFINE-SPOOLING-CONDITIONS . . . . .	65, 70.
DEFINE-SWAP-FILES . . . . .	113.
DEFINE-SYSTEM-HISTOGRAM . . . . .	124.
DEFINE-TIME-SLICE . . . . .	198.
DEFINE-TITLE . . . . .	199.
DEFINE-USER-MONITOR-CALL . . . . .	200.
DEFINE-USER-RESTART-PROGRAM . . . . .	200.
DEFINE-USER-RESTART-SUBROUTINE . . . . .	201.
DEFINE-USER-START-SUBROUTINE . . . . .	201.
delete	
file . . . . .	16.
user . . . . .	35.
DELETE-ACCUMULATED-PROJECT . . . . .	60.
DELETE-ACCUMULATED-USER . . . . .	57.
DELETE-BACKGROUND-PROJECT . . . . .	56.
DELETE-BROADCAST . . . . .	43.
DELETE-CURRENT-SELECTION . . . . .	236.
DELETE-DOMAIN . . . . .	116.
DELETE-FILE . . . . .	16, 225.
DELETE-LAMU . . . . .	202.
DELETE-LAST-KEY . . . . .	236.
DELETE-MESSAGE . . . . .	43.
DELETE-REENTRANT . . . . .	45.
DELETE-RT-PROJECT . . . . .	57.
DELETE-SEGMENT-FILE . . . . .	202.
DELETE-SPOOLING-FILE . . . . .	73.
DELETE-USER . . . . .	35.
DELETE-USERS-FILES . . . . .	16.
DESCRIBE-ALL-COMMANDS . . . . .	56.
DESCRIPTION-FILE:DESC . . . . .	115.
DESTINATION-EXPANSION-MODE . . . . .	238.
DESTINATION-FILES-EXIST . . . . .	236.
device	
buffer . . . . .	190.
controller . . . . .	268.
name . . . . .	21, 251, 268.
test . . . . .	171.
unit . . . . .	21.
DEVICE-FUNCTION . . . . .	241.
DEVICE ERROR . . . . .	139.
device number table . . . . .	282.
diagnosis . . . . .	125.
DIMS . . . . .	90, 226.
test program . . . . .	171.
DIRECT-BROADCAST . . . . .	42, 76.
directory . . . . .	13, 19, 228.
addresses . . . . .	163.
consistency . . . . .	162.

entry . . . . .	26, 260.
error . . . . .	168.
index . . . . .	19, 223.
layout . . . . .	259.
name . . . . .	21.
swap . . . . .	223.
table . . . . .	19.
DIRECTORY-STATISTICS . . . . .	25, 26.
DISC-TEMA . . . . .	23, 90, 226.
test program . . . . .	171.
disk	
BIG CARTRIDGE . . . . .	10.
cylinder . . . . .	266.
definitions of terms . . . . .	265.
formatting . . . . .	90.
handling . . . . .	7.
HAWK . . . . .	9.
maintenance system . . . . .	226.
MMD . . . . .	11.
names . . . . .	251.
organization . . . . .	259.
pack . . . . .	265.
PHOENIX . . . . .	10.
sector . . . . .	266.
space . . . . .	34, 91.
starting . . . . .	12.
track . . . . .	265.
unit . . . . .	265, 268.
write protection . . . . .	12.
disks Winchester . . . . .	226.
DIVIDE BY ZERO . . . . .	141.
DMA . . . . .	170.
DMAC . . . . .	180, 246.
domain . . . . .	115.
DUMMY . . . . .	122.
DUMP . . . . .	108.
memory . . . . .	152.
system area . . . . .	212.
DUMP-AND-ACCUMULATE . . . . .	58.
DUMP-BACKUP-SYSTEM . . . . .	238.
DUMP-BIT-FILE . . . . .	163.
DUMP-DIRECTORY-ENTRY . . . . .	26, 163.
DUMP-OBJECT-ENTRY . . . . .	19, 163.
DUMP-PAGE . . . . .	163.
DUMP-PROGRAM-REENTRANT . . . . .	45.
DUMP-REENTRANT . . . . .	44, 108, 236.
DUMP-REENTRANT:SYMB . . . . .	105.
example . . . . .	274.
DUMP-RT-DESCRIPTION . . . . .	203.
DUMP-SEGMENT-TABLE-ENTRY . . . . .	203.
DUMP-USER-ENTRY . . . . .	163.
DUMPFLL:BPUN . . . . .	152.

emulator . . . . .	40.
ENDCOR . . . . .	182.
enter count . . . . .	194, 221.
ENTER-DIRECTORY . . . . .	24, 35, 115, 241.
initially . . . . .	84.
enter count . . . . .	36, 168.
entry point . . . . .	211.
ERRFATAL . . . . .	148, 153.
error . . . . .	127.
device . . . . .	4, 128.
information list . . . . .	175.
log . . . . .	128, 149.
logging . . . . .	128.
messages . . . . .	130.
symptoms . . . . .	148.
with terminal . . . . .	155.
escape character . . . . .	156.
exact match . . . . .	20.
EXECUTE . . . . .	236.
EXIT . . . . .	204, 238.
EXPAND-FILE . . . . .	102.
extended ident table . . . . .	187, 205, 214.
failure . . . . .	127.
FALSE BEX INTERRUPT . . . . .	145.
FALSE INTERRUPT . . . . .	138.
FALSE MPM4 INTERRUPT . . . . .	145.
fatal error . . . . .	127, 148, 153.
file . . . . .	13.
allocated . . . . .	15.
attribute . . . . .	17, 62, 235.
error . . . . .	169.
index number . . . . .	17.
layout . . . . .	263.
mass storage . . . . .	13.
system . . . . .	13.
type . . . . .	14.
version . . . . .	14.
version error . . . . .	169.
FILE-ATTRIBUTE . . . . .	236.
FILE-INDEX-INTERVAL . . . . .	236.
FILE-NAME . . . . .	236.
FILE-STATISTICS . . . . .	17, 28.
FILESYS-INV . . . . .	161.
file name complete . . . . .	13.
file system layout . . . . .	259.
FILE SYSTEM ERROR . . . . .	140.
FILE SYSTEM INVESTIGATOR . . . . .	161, 162.
FIXING DEMAND . . . . .	137.
FLOPP-FORM test program . . . . .	171.
FLOPPY-1 . . . . .	232.
FLOPPY-FU test program . . . . .	171.

FLOPPY-RAN test program . . . . .	171.
FLOPPY-USER . . . . .	242.
floppy disk	
formatting . . . . .	241.
monitor . . . . .	160.
test . . . . .	171.
floppy monitor . . . . .	160.
FMAC . . . . .	179, 246.
form . . . . .	70.
FORMAT . . . . .	90, 226.
FORMAT-FLOPPY . . . . .	241.
formatting	
disks . . . . .	90.
floppy disks . . . . .	241.
FORTRAN RUN TIME ERROR . . . . .	147.
FORWARD-SPACE-PRINT . . . . .	72.
FTOW . . . . .	226.
generation	
backup . . . . .	225.
system . . . . .	75.
GENERATION-INTERVAL . . . . .	236.
GET-ERROR-DEVICE . . . . .	128.
GIVE-N500-PAGES . . . . .	114.
GIVE-SPOOLING-PAGES . . . . .	65, 69.
GIVE-USER-SPACE . . . . .	34.
hang situation . . . . .	148, 149, 155.
hanging	
computer . . . . .	127.
terminal . . . . .	155.
HAWK disks . . . . .	9.
HDLC-MEGALINK test program . . . . .	172.
HELP . . . . .	204, 236, 238.
HENT . . . . .	83, 86.
HENT-MODE:SYMB . . . . .	83, 89.
example . . . . .	271.
ND-500 . . . . .	273.
high-level language . . . . .	246.
histogram . . . . .	122.
commands . . . . .	123.
hole . . . . .	16.
IBITNO . . . . .	153.
ident	
code . . . . .	173, 246.
table . . . . .	187, 206, 215.
II . . . . .	154.
ILL. ERROR IN ERMON . . . . .	143.
ILL. INSTR . . . . .	141.
ILL. MONITOR CALL . . . . .	133.
ILL. PARAM. IN ABST . . . . .	135.
ILL. PARAM. IN CLOCK . . . . .	135.
ILL. PARAM. IN CONCT . . . . .	140.
ILL. PARAM IN UPDAT . . . . .	135.

ILL. PRIORITY . . . . .	133.
ILL. RT PROG . . . . .	133.
ILL. SEGMENT NUMBER . . . . .	136.
ILL. TIME PARAM. . . . .	135.
IMAGE AREA . . . . .	80, 87, 176.
incremental backup . . . . .	239.
index block . . . . .	259, 264.
indexed file . . . . .	15, 264.
INIT-ACCOUNTING . . . . .	47, 53.
initial	
command . . . . .	84, 89.
command buffer . . . . .	84.
loading . . . . .	75, 89, 94.
INITIAL-COMMAND . . . . .	84, 85.
initialize MAIL . . . . .	41, 104.
INITIALIZE-BACKGROUND-PROGRAMS . . . . .	85, 102.
INITIALIZE-ERROR-LOG . . . . .	128.
INITIALIZE-SYSTEM-SEGMENT . . . . .	204.
input device . . . . .	62.
INSERT-IN-BACKGROUND-TABLE . . . . .	205.
INSERT-IN-EXTENDED-IDENT-TABLE . . . . .	205.
INSERT-IN-IDENT-TABLE . . . . .	206.
INSERT-IN-IOX-TABLE . . . . .	206.
INSERT-IN-LOGICAL-UNIT-TABLE . . . . .	207.
INSERT-IN-TIME-SLICE . . . . .	119, 208.
INSERT-IN-TIMER-TABLE . . . . .	209.
INSERT-SPOOLING-HEADER . . . . .	210.
INSTRUCTION test program . . . . .	170.
INTERNAL INTERRUPT ON DIRECT TASK LEVEL . . . . .	134.
interrupt level . . . . .	124, 151.
interrupt level active . . . . .	153.
IOACCTAB . . . . .	52.
IOX . . . . .	154.
instruction . . . . .	64.
table . . . . .	187, 206, 215.
ISAM . . . . .	54.
lamu . . . . .	187, 194, 202, 210-213.
LAMU-AREAS . . . . .	210.
LAMU-INFORMATION . . . . .	210.
level interrupt . . . . .	124, 151.
line switch . . . . .	6.
LINE-PRINTER . . . . .	65, 103.
line printer test program . . . . .	172.
LINKAGE LOADER . . . . .	115.
LIST-ACTIVE-PROCESSES . . . . .	114.
LIST-ADDRESSES . . . . .	211.
LIST-BACKGROUND-PROJECTS . . . . .	57.
LIST-BATCH-PROCESS . . . . .	76.
LIST-BROADCASTS . . . . .	43.
LIST-CURRENT-SELECTION . . . . .	236.
LIST-DEVICE . . . . .	118, 155.

LIST-DIRECTORIES-ENTERED . . . . .	25.
LIST-FILES floppy disk monitor . . . . .	161.
LIST-INITIAL-COMMANDS . . . . .	85.
LIST-LAMU-CONSTANTS . . . . .	211.
LIST-MESSAGES . . . . .	42.
LIST-REENTRANT . . . . .	45.
LIST-RT-ACCOUNTING . . . . .	54.
LIST-RT-PROGRAM . . . . .	77.
LIST-RT-PROJECT . . . . .	57.
LIST-SELECTED-FILES . . . . .	236.
LIST-SERVICE-COMMANDS . . . . .	211.
LIST-SPOOLING-FORM . . . . .	74.
LIST-SPOOLING-QUEUE . . . . .	71, 76.
list logged in users . . . . .	118.
LOAD button . . . . .	6, 281.
LOAD-CONTROL-STORE . . . . .	114.
LOAD-FILE . . . . .	160.
LOAD-MODE:SYMB . . . . .	82, 89, 222, 269.
ND-500 . . . . .	270.
LOAD-SWAPPER . . . . .	114.
loading	
ND-500 subsystems . . . . .	115.
SINTRAN III . . . . .	75, 93.
subsystems . . . . .	107.
loading to segments . . . . .	112.
log	
file . . . . .	237.
out . . . . .	77.
logged in users . . . . .	118.
logging errors . . . . .	128.
logical addresssed memory unit . . . . .	187.
logical device number . . . . .	246.
logical unit numbers . . . . .	246.
logical unit table . . . . .	207, 216.
LOGOUT-PROCESS . . . . .	114.
LOOK-ACCUMULATED . . . . .	60.
LOOK-AT . . . . .	178.
LP-TEST test program . . . . .	172.
MAC . . . . .	246.
MACM . . . . .	83, 86, 246.
MACM-AREA:DATA . . . . .	83, 86, 176, 247.
create . . . . .	100.
MAG-TAPE-1 . . . . .	232.
magnetic tape . . . . .	232.
magnetic tape backup . . . . .	226.
MAIL . . . . .	41, 76, 104.
from batch . . . . .	43.
mailbox . . . . .	41.
size . . . . .	104.
MAILBOX:DATA . . . . .	41.

create . . . . .	104.
main directory . . . . .	19, 25.
maintenance . . . . .	159.
MANUAL SELECTION . . . . .	236.
MANUAL-STANDARD-VOLUME . . . . .	238.
mass storage	
device . . . . .	246.
file . . . . .	13.
MASS STORAGE TIME-OUT . . . . .	140.
mass storage device controller . . . . .	268.
master block . . . . .	19, 22, 23, 259, 260.
MASTER-CLEAR ND-500 . . . . .	114.
MASTER-LOG-MODE . . . . .	238.
MASTER CLEAR . . . . .	5, 6.
maximum enter count . . . . .	36.
MB . . . . .	8.
MC . . . . .	154.
MCL . . . . .	6.
button . . . . .	5.
MCOPY . . . . .	226.
MCOPY-TANB . . . . .	226.
measure system performance . . . . .	117.
measuring address space . . . . .	122.
measuring RT programs . . . . .	118.
megabyte . . . . .	8.
memory	
configuration . . . . .	112.
dump . . . . .	152.
test program . . . . .	170.
MEMORY-CONFIGURATION . . . . .	113.
MEMORY OUT OF RANGE . . . . .	143.
MEMORY PARITY . . . . .	142.
MEMTOF . . . . .	150, 152.
MEMTOF:BPUN . . . . .	152.
message	
number . . . . .	43.
send . . . . .	41.
MICRO-START . . . . .	114.
MICRO-STOP . . . . .	114.
microprogram . . . . .	5, 172.
MMD disks . . . . .	11.
MODE-BACKUP-SYSTEM-VOLUME . . . . .	238.
MODE-STANDARD-VOLUME . . . . .	238.
MODIF-PATCHES:SYMB . . . . .	175.
MODIFIED-SINCE-LAST-BACKUP . . . . .	236, 239.
modify SINTRAN III . . . . .	175.
MOPC . . . . .	5, 76, 78, 172.
MOR . . . . .	154.
MOVE-SPOOLING-QUEUE-ENTRY . . . . .	72.
MOVER test program . . . . .	170.
MPM4 MEMORY ERROR . . . . .	146.

MPM4 MEMORY OUT OF RANGE . . . . .	146.
MPM4 PARITY ERROR . . . . .	146.
MPM4 POWER FAIL INTERRUPT . . . . .	146.
MPM4 WRITE PARITY ERROR . . . . .	147.
MULTI test program . . . . .	170.
MULTIUSER-COPY . . . . .	234, 235.
N500: . . . . .	113.
ND Bulletin . . . . .	2.
ND-100 Panel . . . . .	5.
ND-500 . . . . .	111.
HENT-MODE:SYMB . . . . .	273.
LOAD-MODE:SYMB . . . . .	270.
MICRO PROGRAM . . . . .	111.
MONITOR . . . . .	111.
swapping area . . . . .	112.
ND-500-MONITOR . . . . .	113.
ND-NET . . . . .	38.
ND LINKAGE LOADER . . . . .	115.
NEW-SYSTEM . . . . .	105.
NEXT-INITIAL-COMMAND . . . . .	84.
NLL . . . . .	115.
nonfatal error . . . . .	127, 129.
NON DEMAND SEGMENTS . . . . .	144.
normal start . . . . .	6.
NOTIS WP . . . . .	16.
NUMBER-OF-ACCOUNTS . . . . .	60.
numbers octal binary decimal . . . . .	249.
object	
entry . . . . .	19, 259, 262.
file . . . . .	259.
index . . . . .	19.
object file index block . . . . .	262.
octal number . . . . .	249.
OCTAL-DUMP . . . . .	212.
OPCOM . . . . .	78.
button . . . . .	6, 78.
floppy disk monitor . . . . .	161.
operating system . . . . .	247.
operator . . . . .	1.
output device . . . . .	63.
OUTSIDE SEGMENT BOUND . . . . .	136.
OVERLAPPING SEGMENTS . . . . .	143.
page . . . . .	13, 247.
conflict . . . . .	169.
swapping . . . . .	93.
PAGES-FROM-LAMU . . . . .	212.
PAGES-TO-LAMU . . . . .	213.
PAGE FAULT FOR NON-DEMAND . . . . .	136.
PAGING test program . . . . .	170.
PAGING OFF AREA . . . . .	87, 176.
panel . . . . .	5.
PARAL-BYTE test program . . . . .	172.



parameter file . . . . .	231, 235.
parity bits . . . . .	170.
password . . . . .	36.
patch . . . . .	176, 247.
file . . . . .	108.
PATCHFILE:SYMB . . . . .	110, 175.
patching . . . . .	175, 193.
PED . . . . .	16, 105.
performace diagnosis . . . . .	125.
performance improvement . . . . .	117.
peripheral . . . . .	6, 61.
file . . . . .	16, 61, 103.
file access . . . . .	62.
PERMIT VIOLATION . . . . .	141.
PERR . . . . .	153.
PF . . . . .	154.
PFAIL test program . . . . .	170.
PHILIPS . . . . .	18.
PHOENIX disk . . . . .	10.
physical device number . . . . .	173.
physical device number . . . . .	247.
PI . . . . .	154.
PLACE-BINARY . . . . .	108.
PLACE-FILE . . . . .	161.
POF . . . . .	80, 87.
POFST . . . . .	154.
POW . . . . .	154.
POWER FAIL . . . . .	143.
test . . . . .	170.
print header . . . . .	210, 218.
PRINT-ACCUMULATED-ACCOUNTS . . . . .	59.
PRINT-ERROR-LOG . . . . .	128.
PRINT-HISTOGRAM . . . . .	123.
printer . . . . .	71, 103.
printing . . . . .	73.
PRIOR . . . . .	118.
priority . . . . .	118, 119, 216, 219.
PRIV. INSTR. . . . .	142.
PROCESS-STATUS . . . . .	114.
PROGRAM LOG . . . . .	122.
project . . . . .	
name . . . . .	47.
password . . . . .	47, 53.
PROJNAM:DATA . . . . .	48.
prompt string . . . . .	195.
PTY . . . . .	154.
public user . . . . .	1.
PV . . . . .	154.
QED . . . . .	105.
READ-BINARY . . . . .	213.
real time program . . . . .	247.

RECOVER . . . . .	44, 45.
reentrant subsystems . . . . .	44, 86, 108.
REENTRANT-FTN STACK ERROR . . . . .	142.
REGENERATE-DIRECTORY . . . . .	164.
REGISTER . . . . .	178.
release directory . . . . .	24.
RELEASE-DIRECTORY . . . . .	115, 241.
relocatable code . . . . .	247.
remote password . . . . .	39.
REMOTE-LOAD . . . . .	38.
REMOTE-PASSWORD . . . . .	39.
remote diagnostics . . . . .	156.
remote job emulator . . . . .	40.
REMOTE POWER FAIL INTERRUPT . . . . .	145.
REMOVE-FROM-BACKGROUND-TABLE . . . . .	214.
REMOVE-FROM-EXTENDED-IDENT-TABLE . . . . .	214.
REMOVE-FROM-IDENT-TABLE . . . . .	215.
REMOVE-FROM-IOX-TABLE . . . . .	215.
REMOVE-FROM-LOGICAL-UNIT-TABLE . . . . .	216.
REMOVE-FROM-SPOOLING-QUEUE . . . . .	73.
REMOVE-FROM-TIME-SLICE . . . . .	118, 216.
REMOVE-FROM-TIMER-TABLE . . . . .	217.
REMOVE-SINTRAN-COMMAND . . . . .	217.
REMOVE-SPOOLING-HEADER . . . . .	218.
rename file . . . . .	16.
RENAME-DIRECTORY . . . . .	25.
RENAME-USER . . . . .	35, 242.
RESET-ACCOUNTING-FILE . . . . .	59.
RESET-BACKGROUND-PROJECT-TABLE . . . . .	56.
RESET-RT-PROJECT-TABLE . . . . .	57.
RESET-SINGLE-SEARCH . . . . .	238.
RESIDENT . . . . .	79.
resource . . . . .	4.
response time . . . . .	117.
RESTART-PRINT . . . . .	71.
RESTART-SYSTEM . . . . .	75, 82.
RING VIOLATION . . . . .	141.
RJE . . . . .	40.
RT . . . . .	1.
command . . . . .	155.
description . . . . .	187.
program . . . . .	247.
RT-PROGRAM-LOG . . . . .	118, 119.
RTC-12 test program . . . . .	170.
RTCOMMON . . . . .	178, 188.
size definition . . . . .	196.
RTENTER . . . . .	84, 89.
RTFIL:DATA . . . . .	103.
RTOPEN . . . . .	84.
RTPROJ:DATA . . . . .	48.
RT description dump . . . . .	203.
RT LOADER . . . . .	89, 112, 130, 180.

RT program system included . . . . .	77.
RUN-MAIL-SYSTEM . . . . .	43.
run time errors . . . . .	129.
SAVE AREA . . . . .	79, 87, 176.
scratch . . . . .	16.
file . . . . .	16, 92, 220.
OPEN . . . . .	16.
user . . . . .	31.
user access . . . . .	101.
SCRATCH01:DATA . . . . .	101.
sector . . . . .	183, 266.
segment file definition . . . . .	202.
SEGFIL0:DATA . . . . .	80, 86, 93, 247.
create . . . . .	100.
segment . . . . .	17, 176, 247.
background . . . . .	102.
file . . . . .	17, 93, 247.
file area . . . . .	109.
file definition . . . . .	197.
file size . . . . .	102.
number . . . . .	180.
protection . . . . .	218, 219.
system . . . . .	204.
table . . . . .	188, 203.
SEGMENT-WRITE-PERMIT . . . . .	218.
SEGMENT-WRITE-PROTECT . . . . .	219.
SEGMENT NOT LOADED . . . . .	137.
SELECTION commands . . . . .	236.
send message . . . . .	41.
SEND-DIRECT-MESSAGE . . . . .	42.
SEND-MESSAGE . . . . .	42.
set priority . . . . .	118.
SET-ALLOCATE-CREATE-DEFAULT . . . . .	238.
SET-AVAILABLE . . . . .	4.
SET-CHANNEL-PRIORITY . . . . .	219.
SET-CLOSED-SCRATCH-FILE-SIZE . . . . .	220.
SET-COMMAND-PROTECTION . . . . .	220.
SET-DEFAULT-DIRECTORY . . . . .	24.
SET-DEFAULT-FILE-ACCESS . . . . .	101.
SET-ERROR-DEVICE . . . . .	4, 128.
SET-FILE-ACCESS . . . . .	61.
SET-INFORMATION-FILE . . . . .	60.
SET-LAMU-CONSTANTS . . . . .	221.
SET-MAIN-DIRECTORY . . . . .	25, 35.
SET-MATCHING-MODE . . . . .	238.
SET-MAX-ENTER-COUNT . . . . .	221.
SET-ND-500-AVAILABLE . . . . .	114.
SET-ND-500-UNAVAILABLE . . . . .	113, 114.
SET-NUMBER-OF-PRINT-COPIES . . . . .	74.
SET-PERIPHERAL-FILE . . . . .	61, 103.
SET-SINGLE-SEARCH . . . . .	238.
SET-SPOOLING-DEVICE-NUMBER . . . . .	222.

SET-SPOOLING-FORM . . . . .	70.
SET-TEMPORARY-FILE . . . . .	105.
SET-TERM-SPEED:SYMB . . . . .	105.
example . . . . .	275.
SET-TERMINAL-FILE . . . . .	61, 103.
SET-UNAVAILABLE . . . . .	4, 76, 161.
SET-VOLUME-ACCESS . . . . .	238.
SEVICE PROGRAM CUF . . . . .	236.
shrink file . . . . .	16.
SHRINKING-MODE . . . . .	238.
SIBAS-LOAD:BATC . . . . .	105.
single search . . . . .	238.
SINTRAN:DATA . . . . .	80, 83, 86, 176, 248.
create . . . . .	100.
SINTRAN III . . . . .	
clock . . . . .	100.
initial loading . . . . .	89.
listing . . . . .	175.
modify . . . . .	175.
prompt . . . . .	195.
starting . . . . .	6, 75, 79.
stopping . . . . .	7, 76.
versions . . . . .	108.
SINTRAN III RESIDENT . . . . .	79, 87, 176.
SINTRAN SERVICE PROGRAM . . . . .	181, 185.
SMALL-RAND test program . . . . .	171.
SMD . . . . .	8.
software . . . . .	248.
system report . . . . .	3.
source program . . . . .	248.
SPACE NOT AVAILABLE . . . . .	140.
spoolig header . . . . .	188.
spooling . . . . .	248.
conditions . . . . .	70.
device number . . . . .	222.
file . . . . .	64.
form . . . . .	70.
header . . . . .	210, 218.
index . . . . .	188, 222.
pages . . . . .	69.
program . . . . .	64.
queue . . . . .	71.
system . . . . .	64.
SPOOLING-PAGES-LEFT . . . . .	74.
spooling form . . . . .	66, 74.
SPRT1 . . . . .	18, 64.
SSR . . . . .	3.
stand-alone . . . . .	
program . . . . .	23, 248.
program loading . . . . .	160.
standard file type . . . . .	14.

start	
cold . . . . .	75, 83.
normal . . . . .	6.
warm . . . . .	75, 80.
START-ACCOUNTING . . . . .	47, 52.
START-COMMUNICATION . . . . .	38.
START-HISTOGRAM . . . . .	123.
START-PRINT . . . . .	71.
START-PROGRAM-LOG . . . . .	122.
START-RT-ACCOUNTING . . . . .	53.
START-SPOOLING . . . . .	69.
START-XMSG . . . . .	222.
starting	
disks . . . . .	12.
ND-500 . . . . .	113.
SINTRAN III . . . . .	6, 75, 79.
STC-RUNNER test program . . . . .	171.
STC-TEST test program . . . . .	171.
STOP button . . . . .	6.
STOP-ACCOUNTING . . . . .	47, 53.
STOP-COMMUNICATION . . . . .	38.
STOP-HISTOGRAM . . . . .	123.
STOP-MAIL-SYSTEM . . . . .	43.
STOP-ND-500 . . . . .	114.
STOP-PRINT . . . . .	71.
STOP-PROGRAM-LOG . . . . .	122.
STOP-RT-ACCOUNTING . . . . .	54.
STOP-SPOOLING . . . . .	69.
STOP-SYSTEM . . . . .	76.
STOP-TERMINAL . . . . .	77, 155.
STOP-XMSG . . . . .	223.
stopped computer . . . . .	127, 148.
Stopping	
ND-500 . . . . .	114.
SINTRAN III . . . . .	7, 76.
STOP button . . . . .	6.
subsystem loading . . . . .	104.
subunit . . . . .	21.
SUPER-RAND test program . . . . .	171.
swap	
files . . . . .	112, 113.
rate . . . . .	125.
SWAP-DIRECTORY-ENTRIES . . . . .	223.
SWAP-FILE-1:SWAP . . . . .	113.
SWAPPER . . . . .	111.
swapping . . . . .	93, 121.
symbol	
list . . . . .	175.
SINTRAN III . . . . .	211.
SYMBOL-2-LIST:SYMB . . . . .	152.
SYNC-MODEM testprogram . . . . .	172.
SYS-OUT-1:SYMB . . . . .	105.

SYS-OUT-2:SYMB . . . . .	105.
system . . . . .	1.
commands . . . . .	4.
failure . . . . .	127.
files . . . . .	80, 91.
file addresses . . . . .	100.
generation . . . . .	75.
generation time . . . . .	248.
HISTOGRAM . . . . .	123.
password . . . . .	3, 104.
performance . . . . .	117.
segment . . . . .	204.
supervisor . . . . .	1.
title . . . . .	199.
variable . . . . .	193, 211.
system area dump . . . . .	189.
SYSTEM UNAVAILABLE . . . . .	76.
TAD . . . . .	101.
TAKE-SPOOLING-PAGES . . . . .	65, 70.
TAKE-USER-SPACE . . . . .	35.
TANB-MAG test program . . . . .	171.
TECOD test program . . . . .	171.
TELEFIX . . . . .	156.
adaptor . . . . .	157.
temporary file . . . . .	105.
terminal . . . . .	61, 77, 85, 103.
error . . . . .	155.
file . . . . .	61.
hanging . . . . .	155.
number . . . . .	118, 282.
test program . . . . .	159, 160.
TEST-DIRECTORY . . . . .	164.
TET2200 test program . . . . .	172.
time . . . . .	
slice . . . . .	208.
slicer . . . . .	118, 188, 216.
TIME-USED . . . . .	124.
timer table . . . . .	188, 209, 217.
time slice definition . . . . .	198.
title definition . . . . .	199.
TNCT test program . . . . .	172.
TOO BIG SEGMENT . . . . .	137.
TOO MANY FIXED PAGES . . . . .	137.
track . . . . .	265.
TRANSF . . . . .	148.
unit . . . . .	21, 268.
UNIVERSAL-DMA testprogram . . . . .	172.
UPDAT . . . . .	100.
user . . . . .	
area . . . . .	13, 34.
categories . . . . .	1.
category . . . . .	220.

entry . . . . .	260, 261.
file . . . . .	260.
monitor call . . . . .	200.
restart program . . . . .	200.
restart subroutine . . . . .	201.
RT . . . . .	1.
SCRATCH . . . . .	16, 92.
space . . . . .	34.
SYSTEM . . . . .	1, 31.
USER-COPY-LOG-MODE . . . . .	238.
USER-STATISTICS . . . . .	35.
user file index block . . . . .	260.
verification program . . . . .	159.
VERIFY . . . . .	227.
VERSATEST test program . . . . .	172.
version file . . . . .	14.
volume . . . . .	232, 233.
warm start . . . . .	75, 80.
WHERE-IS-FILE . . . . .	18.
WHO-IS-ON . . . . .	42, 76, 77, 118.
ND-500 . . . . .	114.
WINCH-RAND-A test program . . . . .	171.
Winchester disks . . . . .	226.
word format . . . . .	248.
WTOF . . . . .	226.
XMSG . . . . .	78, 188, 222, 223.
XMSG FATAL ERROR . . . . .	144.
XMSG USER ERROR . . . . .	144.
Z . . . . .	154.



\*\*\*\*\*

**SEND US YOUR COMMENTS!!!**

\*\*\*\*\*



Are you frustrated because of unclear information in this manual? Do you have trouble finding things? Why don't you join the Reader's Club and send us a note? You will receive a membership card — and an answer to your comments.

Please let us know if you

- \* find errors
- \* cannot understand information
- \* cannot find information
- \* find needless information

Do you think we could improve the manual by rearranging the contents? You could also tell us if you like the manual!



\*\*\*\*\*

**HELP YOURSELF BY HELPING US!!**

\*\*\*\*\*

Manual name: SINTRAN III System Supervisor

Manual number: ND-30.003.06 Rev. A

What problems do you have? (use extra pages if needed)

---



---



---



---

Do you have suggestions for improving this manual ?

---



---



---



---



---

Your name:

Date:

Company:

Position:

Address:

What are you using this manual for ?

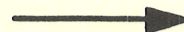
**NOTE!**

This form is primarily for documentation errors. Software and system errors should be reported on Customer System Reports.

**Send to:**

Norsk Data A.S  
Documentation Department  
P.O. Box 25, Bogerud  
0621 Oslo 6, Norway

Norsk Data's answer will be found on reverse side





SEND US YOUR COMMENTS!!!

Answer from Norsk Data

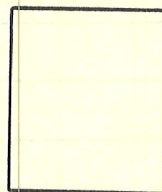
Are you having trouble with this manual? Do you have trouble finding things? Why don't you join the Feedback Club and receive a note? You will receive a membership card and an answer to your comments.

Please let us know if you find errors. We cannot understand information without your information. If you think we could improve the manual by rearranging the contents, you could also tell us if you like the manual.

Manual name: SINTAN II/2 Supervisor Manual number: ND-3003.08 Rev. A

Answered by

Date



**Norsk Data A.S**

Documentation Department

P.O. Box 25, Bogerud

0621 Oslo6, Norway



**Systems that put people first**

NORSK DATA A.S OLAF HELSETHS VEI 5 P.O. BOX 25 BOGERUD 0621 OSLO 6 NORWAY  
TEL.: 02 - 62 60 00 - TELEX: 18284 NDN