RC8000

SW8740/1

Attached Device Processors
System Administrator's Guide

Keywords:
RC8000, ADP, RC8410, RC8411, RC8420, RC8532, F861, F862, F863, F864, ITC, RcCircuit, SW8740/1, SW8741/1, CSP, adpload, FTS, lanstat, terminal download.

Abstract:
System Administrator's Guide for RC8000 Attached Device Processors RC8410/11 and RC8420. This guide gives a general description of the product and covers customization, initialization and some device dependent utilities. It is addressed primarily to the system administrator with a short chapter for users of terminals connected via an ADP.

SW Release:
This manual edition is issued as part of SW8740/1 Rel. 6.0.

Date:
December 1988.

PN: 991 11330

TABLE OF CONTENTS PAGE

1. INTRODUCTION

An Attached Device Processor (ADP) is a subsystem for RC8000. It is used to connect I/O devices to the main processor system. There are two different ADPs with different product names: the RC8420 Multibus I/O system and the RC8410/11 LAN adapter. This guide covers both types of ADP.

The ADPs are microprocessor-based and require software to operate. This guide covers the basic ADP software package (SW8740/1) which provides most of the ADP functions. It is used for all the ADPs. One additional SW package, SW8703/1 IBM Communication, which builds on top of the basic package, is briefly mentioned, but has its own documentation.

The ADP functions fall in three categories:

LAN connection

The ADP attaches the RC8000 to a local area network (LAN). This enables RC8000 to communicate - via device control units - with terminal and printer devices across the LAN and with other computer systems that are also attached to the LAN.

A wide range of RC products may be used as device control units. These include RC891, RC890-30, RC900 with a LAN controller and terminal multiplexer, RC Partner running the CSP terminal program, and finally the ADP itself or another RC8000 ADP.

Terminals and printers may operate as character-oriented (CSP) devices or as 3270 device clusters. CSP is the standard for communication with character-oriented terminals and printers on RcLAN.

The functions which the ADP performs when it communicates with control units across the LAN on behalf of the RC8000 host computer system are called the *CSP host* and the *3270 host* function, respectively.

Other computer systems with which the RC8000 may communicate as a peer are RC39, RC900, RC8000 and RC Partner.

Controlling a floppy disk drive

The ADP (RC8420 only) may be used to control the RC8352 5 1/4" floppy disk drive which can only be accessed by the local RC8000.

Controlling LAN devices

The ADP may operate as a control unit for terminals and printers connected via RcCircuit and V.24 interfaces. When operating as a control unit, the ADP may connect the devices to any host computer attached to the LAN, including the local RC8000. The devices may work as character-oriented (CSP) terminals and printers, or as 3270 terminals and printers.

These ADP functions are referred to as the *CSP terminal*, *CSP printer* and *3270 CU* functions.

Note: The 3270 CU function, which also enables the devices to communicate with a remote IBM or IBM-compatible host computer via a V.24 or X.21 interface controlled by the ADP, is not included in SW8740/1, but in SW8703/1.

Another function of the ADP as a control unit is to download terminal programs to RC45 and RC855 multifunction terminals, i.e. to fetch these programs from file storage and transmit them to the terminals.

An ADP consists of an RC8000 board, referred to as the IFP, for InterFace Processor, and a subsystem built from Multibus boards separate from the main RC8000 system. The Multibus subsystem contains a main processor board and an interface board called MBA, for MultiBus Adapter. The main processor board has an Intel 80186 microprocessor and also controls the LAN interface. It is therefore called the ETC, for EThernet Controller.

All intercommunication between the main RC8000 system and the Multibus subsystem passes via the IFP and the MBA which are interconnected by a cable.

Additional controllers, some of them optional, may be present in the Multibus subsystem. The controllers and the devices they control are described in chapter 2.

This guide is primarily addressed to the system administrator for the RC8000 system. Section 2.7 gives an overview of system administrator tasks.

Chapter 8 contains material aimed at general users of terminals and printers connected via an ADP. The system administrator should make sure that this information is made available to the users.

Chapter 2 is a general description of the ADPs. It identifies and briefly describes all the functions provided in the ADP SW package. It also describes each type of ADP, how it can be configured with optional units, and the functions it can perform. The chapter concludes with the above-mentioned overview of system administrator tasks.

Chapter 3 describes how an ADP can be customized, i.e. adapted to the system configuration and usage. Customization parameters for the various ADP functions are specified in a number of text files which can be changed by ordinary text editing. Chapter 4 describes the utility program crossload which is used to load the IFP and in most cases also the ADP. Chapter 5 deals with FTS, file transfer service. It provides information which is needed by the system administrator in order to provide this service on the system. Chapter 6 describes the utility program lanstat which can be used to obtain status information about the ADP and the LAN to which it attaches the RC8000 system. Chapter 7 describes special utilities for the RC8352 floppy disk drive.

Chapters 3 through 8 provide information about features and functions which for some types and configurations of ADPs are irrelevant. Chapter 2 provides a background which should enable the system administrator to easily identify what information is relevant for his/her system.

The following conventions are used in syntax descriptions in this guide. Metasymbols are written in italics. Slash (/) is used to delimit alternative sub-expressions. Parentheses delimit sub-expressions. A parenthesized sub-expression followed by *, i.e. '(sub-expression)*', means 0 or more occurrences of the sub-expression. In some cases parentheses without the * are used to indicate optional sub-expression; this will be stated in the context.

2. PRODUCT DESCRIPTION

The purpose of this chapter is to give a general understanding of the ADP products and specifically to enable the reader to understand the remainder of the guide and to choose the information which is of relevance.

The first two sections describe RcCircuit and provide some necessary information about the naming schemes used with RcLAN. The third section describes all the functions provided in the ADP SW package (SW8740/1). Sections 2.4 and 2.5 describe the two different types of ADPs. Finally the last section gives an overview of those system administrator tasks that have to do with ADPs.

2.1 RcCircuit

RcCircuit is a simple and inexpensive terminal network. It consists of two twisted wire pairs in a common cable. The two pairs are used independently. They are called Circuit-I and Circuit-II. Some installations will use one, some the other, and some both. Separate controllers are used in the control unit to control each of the Circuits.

RcCircuit is multidropped with up to 32 devices addressable on one cable connected to a control unit. Each of the two Circuits has its own set of addresses. RC terminals, RC45 and RC855, can be attached to both Circuits with an address on each one. Normally a terminal only communicates on one Circuit at a time, depending on the emulator program which is active.

Circuit-I is used for block-mode communication where typically 2000 characters are transmitted in a block that takes approximately 60 msecs. to transfer. 3270 device emulation and terminal download use Circuit-I. Only RC terminals can be attached to Circuit-I. Printers can only be attached indirectly via terminals.

Circuit-II is used for character-mode communication with terminals and printers. In this mode the control unit can transmit or receive characters to or from several devices within a single millisecond. Printers with a parallel Centronix interface and terminals or printers with a serial asynchronous V.24 interface may be attached to Circuit-II via an RC3934 or RC3935 adapter. RC terminals may also be attached directly. Devices attached to Circuit-II are gated onto RcLAN through a CSP terminal function in a control unit.

2.2 Naming Of RcLAN Nodes and Devices

Every host computer, control unit or PC attached to an RcLAN is called a node of the network. An RC8000 with an ADP is both a host computer and a control unit, but it is only one node. Each node must have a node name if it announces a general service on the LAN. The node name is used to identify the service provider. For example, if a LAN has several host computers that provide a

CSP host or FTS server function, terminal users will distinguish between them by means of the node names. A node name is a character string consisting of at most 8 characters which must be letters or digits. It must begin with a letter.

The node name of an RC8000 computer attached to a LAN by means of an ADP is called the *host name*.

Node names are also used to identify devices connected to a control unit. Locally, the control unit will identify the devices by means of a channel number. A global *device name* is then formed by concatenating the node name of the control unit with the channel number of the device in the form of two decimal digits. Thus a device name is at most 10 characters long (refer to section 2.4, ITC terminal controller, for an example).

2.3 ADP Software Functions

FTS, File Transfer Service, allows the user to transfer files between pairs of RC computer systems attached to the LAN, regardless of type, for example between two RC8000 systems or between an RC8000 system and a Partner. This service is provided by a set of programs which run on the main processor of the RC8000 system, making use of the ADP for access to the LAN. A general description of FTS is given in (ref.1). Chapter 5 contains the information which the system administrator needs in order to make FTS available on the RC8000 system.

The ADP SW package also contains some utility programs supporting the special features of the RC8352 floppy disk drive. Chapter 7 describes these programs.

The IFP which provides the interface between the RC8000 system and the Multibus subsystem has a microprocessor whose program must be loaded. The IFP program as well as the RC8000 utility program adpload used to load it are part of the ADP SW package.

The remaining part of the ADP SW package consists of programs to be executed in the Multibus subsystem. Programs are required both for the main processor board and for additional controllers. The Multibus subsystem is also loaded from the main RC8000 system by adpload. The following two subsections describe these programs.

2.3.1 Device Handlers

ADP devices appear to RC8000 software as external processes, i.e. the standard type of device interface on RC8000. For each external process there is a matching *device handler* on the ADP. The external processes are described in separate manuals aimed primarily at systems programmers.

For a LAN device, terminal or printer, the device handler communicates with a control unit across the LAN. As a special case, if the terminal or printer is connected to the same ADP, the communication between device handler and control unit will be local.

The floppy disk device (if present) is controlled directly by its device handler.

Device handlers are activated when the ADP is loaded and started. Before a device handler can be accessed by an RC8000 process it must have a *link* to an external process. The utility programs makelink and deletelink are available for link maintenance (ref.4).

Links may also be created by the programs which use them by means of requests to the ADP main process (cf. ref.2). The main process is a special external process which provides the basic handle to the ADP. The Monitor must be configured with an ADP main process for each ADP in a system. These processes are always called 'lanmain1' for the first ADP, 'lanmain2' for ADP number 2, etc.

The possibilities for customizing the device handlers are described in section 3.4.

The following device handlers are provided:

CSP terminal and printer handlers

This is the CSP host function which makes it possible for RC8000 to communicate with character-oriented terminals and printers via control units attached to the LAN. The control unit must be configured and customized to perform the CSP terminal function.

The CSP host function announces the availability of the host on the LAN using the host name. The announcement enables CSP terminals and printers to connect to the host. When a connection has been established the CSP terminal or printer transmits its device name to the device handler.

Links to CSP terminal or printer device handlers may be created by internal processes, e.g. using makelink. When this is done, the device name of the desired terminal or printer is specified. The device handler will then wait for the device with the specified name to be connected. At this time the link and the device connection are joined, and the link may be used for data transfer between an internal process and the device.

For a CSP terminal the method described above is not the normal way to create a link. Instead, the link and the external process are created automatically when the CSP terminal connects to the CSP host, i.e. the device handler. Within the RC8000 system, the name of the CSP terminal process thus created is derived from the terminal device name by prefixing the digit '0'.

Link creation by an internal process is appropriate for consoles and requires that the CSP terminal in question is customized as type nologin, cf. section 2.3.2.

Connections from CSP printers or nologin terminals will be rejected by the device handler if a link with the device name has not been created.

CSP terminals are compatible with RC8000 conventions for interaction with terminals, so that except for some special cases programs written to communicate with terminals connected by means of older front-ends continue to work with CSP terminals.

Similarly CSP printers are compatible with previous methods of attaching printers to RC8000 systems. In particular CSP printers can be used with PRIMO and BOSS. Section 8.4 describes how to create the catalog entry which is needed by PRIMO or BOSS to access a CSP printer.

The RC8000 CSP terminal and printer processes are described in (ref.2).

3270 input and output handlers

This is the 3270 host function which enables RC8000 to communicate via the LAN with control units which perform a 3270 CU function. The device handlers (input and output) provide the interface required by the FORMAT8000 terminal access procedures. A device handler pair expects to be reserved by an RC8000 application process which uses these procedures and to be informed of the name whereby the application shall be known to CUs. The device handler will then make connections to all CUs.

The number of CUs to which connections are made is defined by customization of the device handler. The control units on the other hand must be customized to provide the 3270 CU LAN Interface (CULANI) function.

The 3270 input and output processes are described in (ref.2).

IMC port handler

This is a pseudo-device which supports process-to-process communication enabling programs running on the RC8000 to communicate with programs running on other computer systems attached to the LAN using the general IMC (Inter Module Communication) functions. The port handler is used by FTS and can also be used by user-written programs to communicate with programs on other RC computer systems.

Floppy disk handler

The RC8352 floppy disk drive is a local device which requires an RC8420 ADP, cf. section 2.4. The flexible disc process is described in (ref.3).

Lanstat device handler

The lanstat device is a pseudo-device used by the lanstat utility program to obtain status information about the ADP and the LAN.

2.3.2 Control Unit Functions

The following device control unit functions are provided:

Terminal download

RC45 and RC855 terminals are microprocessor-based. In their multifunction versions these terminals can execute emulator programs that are loaded via Circuit-I. Terminal programs are not part of the ADP SW package. They are distributed in separate SW packages.

Only RC8420 supports terminal download. Terminal programs must be installed as RC8000 disk files. The terminal downloader accesses these files via FTS when they are needed.

The downloader program which runs on the ADP normally presents the terminal user with a menu of available emulator programs and loads the one which the user selects.

The downloader can be customized very flexibly. The possibilities are described in section 3.5.

CSP terminal and printer functions

Only the RC8420 ADP can perform the CSP terminal and printer functions. Terminals and printers are connected to the ADP via Circuit-II or V.24 interfaces supported by the ITC terminal controller (see section 2.4).

The CSP terminal function allows a terminal to communicate with any host computer that is attached to the LAN and provides the CSP host function.

A CSP printer will automatically connect itself to a CSP host on the LAN. Normally this will be the RC8000 system to which the ADP belongs, but another host may be selected by customization.

A CSP terminal may be customized as type nologin in which case it will connect automatically to the specified host. Printers and nologin terminals will only be accepted by the CSP device handlers on RC8000 ADPs - including the same ADP that the device is attached to - if a link has been created specifying the appropriate device name. Such links are normally created by crossload. A terminal to be used as system console must be specified as type nologin.

Once a connection between a device and a host has been established, the main task of the ADP is to receive blocks of data from the host, disassemble the messages and transmit them character by character to the terminal or printer, and conversely to receive typed characters from terminals, assemble them into appropriate messages, and transmit them as blocks of data to the host.

The programs which perform the CSP terminal and printer functions are the same as in SW8934, CSP Communication for RC890-30. More detailed information about this function may be found in the user's guide for that package (ref.6).

Customization for ITC devices is described in section 3.6.

2.4 RC8420 Multibus I/O System

All ADP functions, as described in section 2.3, can be performed by an RC8420 ADP provided the necessary options are installed.

The Multibus subsystem of the basic RC8420 contains the MBA, the ETC and a terminal/printer controller called the ITC, for Intelligent Terminal Controller. In addition the following options may be installed:

- F861 Micronet adapter
- F862 Ethernet adapter
- F863/F864 Communication controller
- RC8352 Floppy disk drive

ITC terminal controller

The ITC is used exclusively for the CSP terminal function. The terminals and printers connected to the ITC can communicate across the LAN or with the local RC8000 as CSP devices.

The ITC supports 4 V.24 serial asynchronous ports and Circuit-II with 32 addresses. This allows a maximum of 36 devices, terminals or printers, to be connected. Each device is identified by a *channel number*. The V.24 ports have channel numbers 0..3, while the channel number for the device with address *i* on Circuit-II is *i*+4. As described in section 2.2 the channel number - in the form of two decimal digits - is used in combination with the node name (RC8000 host name) to form a device name which is unique on the LAN. For example, if the host name is 'othello', the device with address 4 on Circuit-II has the device name 'othello08'.

F861 micronet/F862 ethernet adapter

One of these options is required in order to attach the ADP to a LAN. The adapter is mounted on the ETC.

F863/F864 communication controller

Both of these controllers support Circuit-I with 32 addresses and one V.24 or X.21 serial synchronous port. The latter is used by the 3270 CU function to communicate with a remote IBM or IBM-compatible host computer.

The only function in the basic ADP SW package which requires one of these controllers is terminal download.

RC8352 floppy disk drive

RC8352 includes an adapter which is mounted on the ETC as well as the actual 5 1/4" floppy disk drive. The drive supports two rotation speeds, high and low.

With high speed, which is normally used, single/double sided diskette usage and single/double density are supported. There are 77 tracks on each side, and the number of sectors per track can be selected as 8, 15 or 26. Using double density the number of

bytes per sector are 1024, 512 and 256 in the three cases. The maximum formatted capacity is 1.2 MBytes.

Only one format is supported with low speed. This is the format used for RC890 diskettes: 80 tracks per side, dual sided, 10 sectors/track, 512 bytes/sector.

The floppy disk drive is supported by standard RC8000 floppy disk utilities as well as some special utilities which are provided in the ADP SW package and described in chapter 7.

2.5 RC8410/11 LAN Adapter

The Multibus subsystem of an RC8410/11 only contains the MBA and the ETC. RC8410 includes a Micronet adapter (same as F861) and RC8411 includes an Ethernet adapter (same as F862). No options can be installed. This ADP serves as a LAN interface only. No devices can be attached. The device handlers for LAN devices including pseudo-devices are supported.

2.6 System Administration

This section provides an introductory overview to system administration by describing a number of tasks which must be handled by the system administrator. The tools to be used are described in detail in the following chapters.

System generation and customization

The system administrator must keep track of the types and numbers of device handlers which are required for the system and also of the associated external processes. The device handlers must be specified by customization of the ADP (cf. chapter 3), whereas the number of external processes can only be changed by generating a new Monitor. The external processes to be used for links to ADP devices must be generated with kind=68 (unused ADP device process).

Initialization

The system start job should perform the following functions:

- load the IFP and the ADP (adpload, cf. chapter 4),
- start the ftsserver and ftsuser programs in dedicated internal processes (cf. chapter 5),
- create external processes for which public/permanent links are appropriate (makelink).

Permanent links may be needed for the following devices: system console, CSP printers, 3270 clusters via the input and output device handlers, and floppy disk. Links for CSP terminals are created automatically when these devices become connected to the ADP. The ftsserver and ftsuser programs (cf. chapter 5) make their own links. So does lanstat (cf. chapter 6).

Printer descriptions

PRIMO and BOSS can use CSP printers. PRIMO can also pass printer output via FTS to a spooler on another RC computer attached to the LAN. Appropriate entries in the main catalog must be created which describe such printers. See section 8.4.

Assistance to FTS users

Users of other systems attached to the LAN who wish to transfer files to and from the RC8000 system must be given resources as users of the RC8000, i.e. in the s user catalog, and must also be registered as FTS users, i.e. in the FTS user catalog (cf. chapter 5).

The system administrator should inform such users of the values needed for the *server*, *user* and *printer* parameters of the FTS utilities (cf. chapter 5 and ref.1) in order to access the RC8000 system. If there is more than one ADP (more than one LAN), FTS users on the RC8000 system must also be informed about the numbers assigned to the LANs (i.e. to the ADPs: the *adpno* parameter of the FTS utilities).

Supervising the LAN, diagnosing and reporting errors

The utility program lanstat (cf. chapter 6) can be used to monitor events on the LAN as seen from the ADP. Such information as well as dumps of the ADP memory which may also be taken by means of lanstat may be useful in the event that an error or malfunction must be reported to RC Computer.

It is possible to attach a diagnostic console to the ADP. During load and start of the ADP information will be written on the console about the parameter files which are read and the programs which are loaded. Also error messages may be written on the console in the event of ADP failures. The information written on the console during load and start is also written to an RC8000 file called 'adpstartlog'. When setting up parameter files it may be useful to scan this file to see if any errors are found in the files. Do not be disturbed that most of the information appears unreadable. Do not try to read the file until the ADP has finished loading. The file is written via FTS with name base determined by the ADP's entry in the FTS user catalog, cf. section 5.2.

3. ADP CUSTOMIZATION

An ADP can be customized in a flexible way by means of parameters which are specified textually in five parameter files. The parameter files are the *ADP configuration* file, the *system* file, the *device handler* file, the *ITC* file and the *terminal download* file.

Whenever the ADP is loaded, an initial program is executed which reads the parameter files to determine which software modules shall be activated and how they shall operate. The appropriate modules are then loaded and activated with the specified parameters. Default values for all the parameters are built into the initial program, and these values remain in effect in all cases where no modification is read from the parameter files. Consequently, the parameter files need only contain specifications of deviations from the default parameter values. However, very few functions are enabled by default (as they would often not be desired), so it is, in general, necessary for the system administrator to set up suitable parameter files.

When reading its parameter files (and program files for software modules to be loaded) the ADP acts as an FTS user accessing the files via the FTS server which must therefore be running on the RC8000 main CPU. The FTS user catalog must contain an entry for this special FTS user, otherwise the ADP cannot read its files. For systems with more than one ADP the system administrator must set up these entries; see section 3.1. For systems with one ADP the distributed catalog contains an appropriate entry, which assumes that the parameter files are named with system base.

The ADP package includes files which show examples of how to specify parameters, the *edit source files*. These files are not read by the ADP. They are intended to serve as starting points when the system administrator customizes the system. The idea is to create the actual parameter files by editing the edit source files. To change the edit source files themselves has no effect.

The following table shows the RC8000 file names which must be used for the parameter files and the file names for the corresponding edit source files.

	file name	edit source file
ADP configuration file	configcst	configed
system file	syscst	sysed
device handler file	adpcst	adped
ITC file	itccst	itced
terminal download file	menudlcst	menudled

To create, for example, the system parameter file, the editor may be called as follows (in a process with catalog base equal to the system base except in the case of multiple ADPs):

```
syscst=edit sysed
```

For each parameter file there is a section below describing the parameters which can be specified in the file. To improve readability some parameters which are normally not modified by

the system administrator are not mentioned in these sections, but described in Appendix B.

The following paragraphs explain syntax rules that apply to the parameter files.

Semicolon (;) may be used to introduce comments; i.e. the part of a text line which follows a semicolon is ignored when the file is read by the ADP.

In the system file, the device handler file, and the terminal download file, each line contains the specification of a parameter value (or just a comment if the line begins with a semicolon). In these files a parameter specification line always begins with the name of the parameter, which is a string of letters (small or capital letters are equivalent). Then follows an equals sign and the value to be used for the parameter in question.

Sections 3.3-3.5 describe the syntax for each of these parameter specification lines. The syntax is shown to the right of the underlined part of the line which introduces the description of the parameter. Parentheses are used to delimit optional parts.

In these files, whenever a parameter specification includes a text (character string), each character following the preceding delimiter (comma or equals sign) is significant, i.e. a character string may include leading blanks. On the other hand, a numeric parameter may be preceded by blanks, which in this case have no significance.

3.1 Multiple ADPs

In a system with multiple ADPs the FTS user catalog must contain an entry for each of these to allow them to read their parameter and program files, cf. section 5.2.

The standard search rule for file names applies when files are read via FTS. The ADPs should therefore be given different bases so that they can read different parameter files. The bases -8741..-8741 for ADP number 1, -8742..-8742 for ADP number 2, etc., are recommended.

When parameter files are created for each ADP they should be named with base equal to the one assigned to the ADP in question, for example by using the base command before the editor is invoked, if this is done under s.

3.2 The ADP Configuration File

The ADP configuration file contains a list of keywords, called configuration switches. The switches determine which functions are activated when the ADP is started. There is no configuration switch for device handlers; their activation is controlled by parameters in the device handler file. However, the following

switches must appear in order to activate the associated functions:

conf.switch	function
CSPTERM	CSP terminal function
CSPPRINT	CSP printer function
IBMCOM	3270 CU function (requires software from SW8703/1 and additional customization as described in the documentation for SW8703/1)
CULANI	Interface to RC host locally or via LAN for 3270 CU function; this switch must be set in addition to IBMCOM if emulated 3270 terminals and printers are to communicate with RC8000 hosts across the LAN (or locally).

The edit source file configed contains more information.

3.3 The System File

The system file contains specification of parameters for general, i.e. device handler independent, aspects of Circuit-I and LAN communication. Most of these parameters are obscure and the system administrator should normally not take any interest in them, as the default values are suitable for most configurations. There is one exception:

CU identification: CUIDENT=*no*

CUIDENT specifies the number whereby the 3270 CU function will identify itself toward Partners performing 3270 terminal emulation. This is significant if multiple 3270 CUs are accessible on the LAN. The default value of *no* is 1. Values may be specified in the range 2..99. The parameter is only relevant if SW8703/1 is used.

3.4 The Device Handler File

This section covers the parameters that can be specified in the device handler file (adpcst). For each kind of device the number of device handlers can be customized. If the number is 0 (the general default), the device handler program is not loaded.

Parameters for CSP terminal and printer handlers

Number of consoles: CONSOLES=*no*

CONSOLES specifies the number of CSP terminal device handlers for which the link will be created by an internal process, typically for use as a console. These device handlers must match CSP terminals specified as type nologin. *no* must be in the range 0..20, and the default value is 0.

Number of devices: CSPNODHS=*no*

CSPNODHS specifies the number of device handlers for ordinary CSP terminals. For these device handlers the link is created when the terminal is connected across the LAN. The number must be sufficient to allow for the maximum number of terminals which need to be connected simultaneously. *no* must be in the range 0..100, and the default value is 2.

Interrupt character: INTCHAR=*no*

INTCHAR specifies the interrupt character which shall be used to generate RC8000 attentions from CSP terminals attached via the device handler. *no* is the decimal value of the character, 0..255. The default is 27 (ESC).

Number of shared input buffers: CSPNBUFS=*no*

CSPNBUFS specifies the number of buffers to be shared among CSP terminals and consoles operating in canonical mode. These buffers are used for queued input buffers containing data that has been typed ahead of prompting. *no* must be in the range 0..400, and the default value is 0.

A single terminal is only allowed to occupy half the buffers.

Number of printers: PRINTERS=*no*

PRINTERS specifies the number of device handlers for printers. *no* must be in the range 0..20, and the default value is 0.

Deselection timer: DESELTIME=*no*

DESELTIME specifies the length of time during which a CSP printer may be deselected (local), e.g. to change paper, before the condition is treated as a fault. *no* is the number of seconds and must be in the range 1..600. The default value is 120.

Parameters for 3270 input and output handlers

Number of devices: 3270NODHS=*no*

3270NODHS specifies the number of input/output device handler pairs. Each application in RC8000 using 3270 devices must have its own device handler pair. *no* must be in the range 0..50, and the default value is 0.

Number of control units: 3270NOCUS=*no*

3270NOCUS specifies the number of 3270 CUs each device handler pair can communicate with. The CUs are numbered from 0 and up, corresponding to port names 'CU00', 'CU01', ... etc. These names must be customized in the respective control units (for example RC890-30 or SW8703/1 running on an ADP). *no* must be in the range 0..100, and has the default value 0.

3270 input timeout: 3270TOUT=*no*

3270TOUT specifies how long the device handler will hold an input data message from a terminal while waiting for the RC8000 internal process to issue an Input operation, i.e. read the data, before it drops the data and shows the message 'Host timeout, data lost', in the status line of the terminal. *no* is the timeout value in seconds; its value must be in the interval 10..120. The default is 20.

Parameters for IMC port handler

Unless a system has user programs which use the IMC port handler, only the ftsserver and ftsuser processes will need port handlers. The appropriate values for the port handler parameters will depend on the parameters given to these processes, cf. sections 5.1 and 5.3.

Number of ports: NOPORTS=*no*

NOPORTS specifies the number of port handlers to be activated. *no* must be in the range 1..100, and the default value is 3.

Number of connection end-points: MAXCONN=*no*

MAXCONN specifies the maximum number of connection end-points per port. The number should be at least equal to the number of incarnations of the FTS server, cf. section 5.1. *no* must be in the range 5..100, and the default value is 5.

Parameters for floppy disk handler

Floppy present: FLOPPY=0/1

FLOPPY specifies if the device handler for the RC8352 floppy disk drive shall be activated. The default value is 0, i.e. the floppy disk handler is not activated.

Example:

```
CONSOLES= 1          ;system console
CSPNODHS= 15         ;number of CSP terminals admitted
PRINTERS= 3          ;number of CSP printers which can connect
FLOPPY= 1            ;enable floppy disk
```

3.5 The Terminal Download File

This section covers the parameters that can be specified in the terminal download file (menudlcst). Terminal download uses Circuit-I

The menu-based terminal downloader identifies terminals by their addresses on Circuit-I. The default menus are identical for all terminals. The default menu is shown below as it appears on a terminal.

```
              TERMINAL FUNCTION MENU

              Key  Description

              PF1  ANSI X3.64 Terminal
              PF2  IBM 3270 Emulator
              PF3  Terminal Configurator




   Press function key to select application
```

Each line beginning with a key name represents a terminal program which can be selected for download. The downloader must know three things, each represented by a character string, about a terminal program:

1. A file name whereby the file containing the program to be downloaded will be retrieved. These names are automatically extended with '855' or 's45' for an RC855 or RC45 terminal, respectively, to obtain the actual file name. For example, if the file name is specified as 'confi' the actual name of the program file which is downloaded to an RC45 terminal is 'confis45'. Because of the automatic extension, the same terminal program specifications may be used for both RC855 and RC45 terminals.

2. A description to appear in the menu line representing the terminal program, e.g. 'IBM 3270 Emulator'.

3. (optional) A parameter to be supplied to the terminal program subsequent to download.

Terminal program: TPRG,*tpno*=*fname*,*desc*(,*param*)

TPRG specifies a terminal program which may be referred to in terminal menu specifications (TMENU, see below) by its number, given as *tpno*, which must be in the range 0..25. Thus there can be at most 26 terminal programs. *fname* is the file name, at most 8 characters; *desc* is the description, at most 25 characters; the optional part *param* is the character string which, if present, will be passed to the downloaded program, at most 89 characters. For information on the use of this feature, refer to the documentation for the terminal emulator program.

The same file name may be used in several terminal program specifications. This can be meaningful if the *desc* and *param* parts are different.

Terminal menu: TMENU,*addr=tpnolist*

TMENU specifies the menu for the terminal with address *addr* on Circuit-I. *addr* must be in the range 0..31. *tpnolist* is a list of terminal program numbers. It must consist of numbers which have appeared in TPRG parameter specification lines prior to the TMENU line. The numbers must be separated by commas.

A number may be immediately preceded by the letter 'D' to indicate a default terminal program. If a terminal is configured for default download and its menu contains a default terminal program, then no menu is shown when the terminal is powered on or reset. Instead, the default program is loaded without requiring selection by the operator.

Example:

The following lines could be used to specify the default menu (distributed as the file menudled):

```
TPRG,0=CONFI,Terminal Configurator
TPRG,1=3270,IBM 3270 Emulator
TPRG,2=ANSI,ANSI X3.64 Terminal
TMENU,0=D2,1,0
TMENU,1=D2,1,0
TMENU,2=D2,1,0
TMENU,3=D2,1,0
...
TMENU,31=D2,1,0
```

3.6 The ITC File

The ITC file contains parameters that specify the use of ITC channels. If a channel is not mentioned in the file it is disabled. Every channel which is enabled must be described by a group of parameter specification lines, beginning with a CHANNEL parameter specification. The next CHANNEL parameter specification marks the beginning of the specification of the next channel, and so on.

In this file every parameter specification has the form of a keyword specifying the parameter followed by another keyword or number specifying the value.

The parameters and associated values are described in the following table:

parameter keyword	values and explanation
CHANNEL	specifies a channel: J1, J2, J3, or J4 are keywords specifying the V.24 channels with channel numbers 0..3 corresponding to the connector engravings. C0, C1,.., C31 are keywords specifying the Circuit-II channels with channel numbers 4..35. Default is disabled.

TYPE

specifies the use of the channel:
TERMINAL: terminal with general host selection
NOLOGIN: terminal or communication line which will automatically and exclusively be connected to the specified host (HOSTSEL)
PRINTERS or PRINTERP: printer which will automatically connect to the specified host; PRINTERS is for a serial printer, PRINTERP for a printer with a parallel Centronix interface (via RC3934).
A channel to be used for an RC8000 console, or, in general, to be connected to a CSP terminal link created by an internal process, must be specified as NOLOGIN.
Default is TERMINAL.

HOSTSEL

specifies the node name of the CSP host to which the device will connect automatically. Max. 8 characters. Default for NOLOGIN or PRINTER is the local RC8000 system, i.e. the host name. Default for TERMINAL is no host selection, i.e. choice from menu. The name must be enclosed in single quotes.

CHARSET

specifies the character set supported by the device attached to the channel: 7BIT, TTY, VT100, 8BIT, VT200. 7BIT and TTY are equivalent; 8BIT and VT200 are also equivalent. Default is 8BIT/VT200.

7BIT/TTY and VT100 are used for devices with a 7-bit character set. On a V.24 interface the number of data bits will be 7. On a Circuit-II channel the high order data bit transmitted to the device will always be zero. If VT100 is specified, locally generated menus will be written using VT100 control sequences. For a terminal which does not support these sequences, TTY shall be specified.

8BIT/VT200 is used for a device with an 8-bit character set. If the device is a terminal, VT200 control sequences will occur in locally generated menus.

The appropriate specification for an RC45 terminal is VT200. However, the terminal itself should be set up as VT100. If such a terminal is connected to an RC900 system, the RC900 will send the control sequence which sets the terminal in VT200 mode.

Note that for a printer or nologin terminal the CHARSET parameter only determines the character size, 7 or 8 bits.

LINESPEED specifies the transmission rate in bits/second on a V.24 interface (J1-J4):
300, 600, 1200, 1800, 2400, 4800, 9600, 19200
Default is 4800. J1 does not support 19200.

STOPBITS specifies the number of stop bits transmitted per character on a V.24 interface (J1-J4):
1, 2. Default is 1.

PARITY specifies the type of parity generated and checked on a V.24 interface (J1-J4):
NONE, EVEN, ODD. Default is NONE.

The parity bit is in addition to the number of bits determined by the CHARSET parameter.

COMT specifies the time to wait for an XON character after XOFF has been received:
1S, 5S, 20S, INF. Default is INF.

MODEMT specifies the length of time during which the data carrier detected signal may be inactive before a channel is considered offline (J1-J4). A numeric parameter indicates milliseconds:
1 - 60000, INF. Default is 200.

Example:

```
CHANNEL         J1          ; console, note HOSTSEL omitted
    TYPE        NOLOGIN
    LINESPEED   9600
    STOPBITS    2
    MODEMT      INF

CHANNEL         J2
    TYPE        PRINTERS
    HOSTSEL     'othello' ; could be another host
    LINESPEED   9600
    COMT        20S
    MODEMT      1000

CHANNEL         C12
    TYPE        TERMINAL
    CHARSET     VT200
    PARITY      EVEN
```

4. ADPLOAD

The utility program adpload is used to reset and load an ADP. It loads the program for the IFP and the initial program for the ETC in the Multibus subsystem. The initial program is then started and proceeds to read the parameter files for the ADP and load the necessary software modules in the Multibus subsystem. Two of the parameter files, adpcst and syscst, are also read by adpload. When the ADP reads its files, it uses FTS. The FTS server must therefore be started immediately after adpload is called, if it is not running already.

The program call is (the parts in parentheses are optional):

(*outfile* =) adpload (adpno.*no*) *adptype* (hostname.*namespec*)

adptype ::= rc8410 / rc8411 / rc8420

If an *outfile* is specified, messages from the program are written on the specified file, otherwise on current output.

The adpno parameter may be used to specify a particular ADP in an installation which has more than one; *no* is the number of the ADP: 1, 2, etc. Default is 1.

The specification of ADP type is used to select the correct initial program for the ETC.

The name given as *namespec*, which must be a string of at most 8 characters, is used by the ADP as its host name. If the hostname specification is omitted, the host name is taken from Monitor, i.e. the installation name which is defined when the Monitor is generated.

If there are multiple ADPs they usually attach the RC8000 system to different LANs. In this case it is recommended that the ADPs be given the same host name, so that the RC8000 system is seen with the same name on all the LANs. On the other hand, if an RC8000 system is attached to the same LAN via multiple ADPs, they must be given different host names so they can be distinguished.

Error messages written by adpload are self-explanatory.

If the IFP has gone badly astray, adpload may not be able to establish the necessary communication with it. In this case adpload will write a message indicating a timeout. When this has occurred, the following special form of the adpload command should be given, before the failing command is repeated:

adpload (adpno.*no*) panicreset

5. FILE TRANSFER SERVICE

FTS allows users to transfer files between any two RC computer systems attached to the same LAN. A general description of FTS is given in (ref.1). The description introduces a number of concepts, including FTS server and FTS user, and explains the parameters which must be supplied to FTS to specify file transfers. To understand the remaining part of this chapter the reader must be familiar with the general description.

On RC8000 systems the FTS server function is provided by the ftsserver program which must run permanently in an internal process. In order to access files in RC8000 file storage the ADP acts as an FTS user, accessing its files via the FTS server. Therefore ftsserver must be running to support the ADP, even if FTS is otherwise not used.

The FTS utility programs wr and rr work in conjunction with a program called ftsuser, which, just like ftsserver, must run permanently in an internal process. This process will accept requests for file transfers to or from FTS servers on other computers attached to the LAN in the form of messages from other internal processes. The programs wr and rr generate such messages, but they may also be generated by other processes. This is the mechanism whereby PRIMO may serve requests to print files on printers attached to other systems.

LAN/ADP number

If a system is attached to multiple ADPs and/or LANs, the ADP number to be used to reach other servers must be made known to users of FTS. This information is required both when wr and rr are used and when a main catalog entry is created describing a printer to be reached by PRIMO via FTS (cf. section 8.4).

Server name

Users accessing the RC8000 FTS server function via the LAN need to know the server name whereby ftsserver announces itself on the LAN. The server name of the RC8000 system is identical to the host name, cf. section 2.2 and chapter 4.

Receiving files to be printed

The FTS server can receive files to be printed on a printer accessible to PRIMO on the RC8000 system. The FTS server passes the printer name as well as the file contents to be printed to PRIMO which controls the actual printing process. Thus PRIMO must be installed and running if this feature is to be used, and the printer name to be specified in the call of wr is the name of the catalog entry which describes the printer to PRIMO.

For example, if a printer is described by a catalog entry named 'lp5' on the RC8000 system with name 'ophelia', then the command

```
wr s.ophelia p.lp5 letter
```

may be given on another RC8000 system attached to the LAN to transfer the file named 'letter' to ophelia to be printed on printer lp5.

5.1 Ftsserver

The ftsserver program must run permanently in an internal process. The program can run even if the ADP is not active. If the link to the ADP is lost, ftsserver will try to set it up again every 30 seconds. It may therefore be started even before the ADP has been loaded.

Ftsserver can run in several incarnations within its process. The number of incarnations determines the number of FTS users (on other systems) that can be serviced simultaneously. The recommended name for the process is ftsserver. The name of the program file may change with new releases and will be given in the package description.

If the RC8000 system has multiple ADPs, an ftsserver process must be started for each one. In this case, append the ADP/LAN number to the process name (ftsserver1, ftsserver2 etc.).

Function bits 4 and 5 must be set to allow the process to create and remove links to IMC port handlers.

See the package description for other process requirements.

Disk resource requirements

When ftsserver receives a file which does not overwrite an existing file, it must have disk resources to store the file. The ftsserver process should therefore be started with a reasonable amount of disk resources. How much is up to the system administrator.

Resources used for a print file are reused after PRIMO reports back that the file has been printed, but resources for user files are consumed. This is also the case when an existing file is overwritten with a longer received file.

Program call

Assuming the program file name is bftsserver the program call is as follows:

bftsserver (*param*)*

where each *param* may be one of the following:

1.*adpno*	*adpno* is the number of the ADP (and the LAN). Default is 1.

bufsize.*size* — *size* is the size of the data buffers which are used to move file data. The default is 4096 bytes. If less than 128 is specified, the buffer size will be set to 128.

incno.*no* — *no* is the number of incarnations and must be at least 2. The default is 5.

ftscat.*catname* — *catname* is the name of the file which will be used as FTS user catalog. The default name is 'ftsusercat'.

Starting ftsserver, an example

The ftsserver process may be created and the program started by means of the following commands, which should be contained in a file named startserver:

```
*new ftsserver
*base -8388607 8388605
*buf 9
*area 9
*function 4 5
*size 100000
*perm disc 1000 10
*perm disc1 1000 10
*i 4 startserver
*run
*unstack
bftsserver incno.6
finis
```

To execute the commands in the startserver file, the command 'read startserver' may be given from the system console or included in the general system startup file.

Note that this is only an example. The process could be started in several other ways.

Status commands and error messages

The ftsserver process will display status information in response to commands received from a terminal. The following commands are supported:

disp — gives information about transfers in progress.

last — gives information about the last events that have occurred, from a buffer which is overwritten cyclically. Events are file transfers and errors experienced by the ftsserver process. One line is displayed for each file transfer with the following information: transfer start time, R/W for read or write, file name, file base, modification time for a file which is read, and transfer end time (or error message in case of a disk resource problem or other abnormal termination).

stop — stops the process after current transfers.

Error messages are not written on the system console, but only in the buffer, from which they can be retrieved by means of the *last* command. All messages, including logged file transfers are time stamped. The following situations are reported:

- sense error

  The ADP does not respond. This message is written periodically when the ADP is not operational.

- link rejected *rno*

  An attempt by the ftsserver to create a link to a port handler on the ADP failed. The reason indicated by *rno* may be:

  1 No unused external process
  2 No free port handler

- reducing incs *rno*

  The number of incarnations is reduced because of lack of resources. The kind of resources is indicated by *rno* and is most likely to be:

  1 Number of end-points per port handler

- link created *n* incs

  A link to a port handler has been successfully set up and initialized supporting *n* incarnations of the server.

- link removed

  The link to the port handler was abnormally removed because the ADP went out of operation.

- no buffer

  An incarnation of the ftsserver was stopped because the number of RC8000 buffers given to the process was too small.

5.2 FTS User Catalog

Before starting a file transfer, the FTS server will look up the user in the FTS user catalog to establish the proper disk, base for file name search, and access rights. The FTS user catalog is a file containing an entry per user. A user is defined by a user name and a user number, so there may be several entries with the same user name, but different user numbers. The user number may thus be used as a password. By default the name of the FTS user catalog file is 'ftsusercat'. A different name may be used if desirable. An existing ACP user catalog may be used as FTS user catalog, but not vice versa.

A catalog containing a few predefined entries is included in the distribution. One of these entries with the user name 'fts' which

allows printing, but no reading and writing of files, is used by the FTS server, when the FTS *user* parameter is missing in a file transfer request from an FTS user. This means files can be spooled for printing by users who are not registered on the RC8000 system.

The other entries are for ADPs. They allow each ADP to access RC8000 files as an FTS user: to read parameter files and program files for the ADP itself as well as for download to terminals, and to write the file 'adpstartlog'. These entries have user name 'adp' and user number equal to the ADP number. Consult the package description for further details. If an installation has more ADPs than the catalog has entries for, the system administrator must make the entry using ftscatup, as described below. The bases specified must be different for the different ADPs, cf. section 3.1.

The FTS user catalog is maintained by means of the utility program ftscatup. The program call is:

ftscatup (*command*)*

The commands are executed in the order they appear. There are five legal forms of *command*. These are explained below. In all cases *name* and *no* refer to user name and user number, respectively.

command	explanation
l	Lists the entries in the FTS user catalog with base matching the caller's user base; if caller has system base, all entries are shown.
i.*name.no.l.u.d.w.r.p*	Inserts an entry for the specified user. *l* and *u* are the lower and upper base, respectively. *d* is the name of the disk on which the user may write. *r*, *w* and *p* are optional and indicate access rights: read, write and print, respectively.
d.*name.no*	Deletes the specified entry.
n.*catname.size*	Creates a new FTS user catalog in a file named *catname* consisting of *size* segments. Each segment can hold 16 entries.
c.*catname*	Defines the name (*catname*) of the file holding the FTS user catalog to be accessed by subsequent commands. If no c command is present, the default name 'ftsusercat' is used.

Messages from ftscatup are self-explanatory.

5.3 Ftsuser

Like the ftsserver, the ftsuser program must run permanently in an internal process. It can also run in several incarnations within the process. The number of incarnations determines the number of FTS users, i.e. processes on the local RC8000 system, that can be serviced simultaneously. The name of the process must be ftsuser. The name of the program file may change with new releases and will be given in the package description.

One ftsuser process may be used for communication on multiple LANs, if required, as the link to the ADP is established for each transfer.

Function bits 4 and 5 must be set to allow the process to create and remove links to IMC port handlers.

See the package description for other process requirements.

Program call

Assuming the program file name is bftsuser the program call is as follows:

bftsuser (*param*)*

where each *param* may be one of the following:

bufsize.*size*	*size* is the size of the data buffers which are used to move file data. The default is 4096 bytes. If less than 128 is specified, the buffer size will be set to 128.
incno.*no*	*no* is the number of incarnations. The default is 2.

All information needed by the ftsuser program is obtained per file transfer from the requesting internal process.

Starting ftsuser

The ftsuser process is created and started in a fashion similar to the ftsserver process, cf. section 5.1. However, ftsuser does not require disk resources, only entries for current input and current output.

6. LANSTAT

The utility program lanstat may be used to obtain information about the activities of the ADP and the traffic on the LAN.

The form of the program call is (parentheses without a * enclose optional parts):

(*outfile* =) lanstat ((1.*adpno*) (*command*)*)*

outfile is the name of a file which is used as destination if a dump command is given.

The 1 parameter specifies which ADP/LAN the following commands concern; *adpno* is the number of the ADP. Default is 1.

A number of commands may be given in one call. Each command must be one of the following:

command	information displayed
lan	LAN statistics: traffic and retransmission counters etc.
ports(.*sub*)	IMC port names visible on the LAN. In addition to the port name, the local/global scope of each port is displayed, and a link indication which makes it possible to see which ports are on the same node on the LAN. Local ports are ports which are not visible on other nodes. *sub*, if present, must be a character string, shorter than 12 characters. If it is present, only those port names in which *sub* occurs as a substring will be shown. When a CSP host announces itself on the LAN it does so by opening an IMC port whose name ends with 'CSP'. Consequently, all visible CSP hosts may be displayed by the call: lanstat ports."CSP" Similarly, FTS servers announce themselves by opening a port whose name ends with 'FTS'.
dev	Survey of device handlers and links to external processes.
csp	Status information for CSP console and printer device handlers. For each device handler which is linked to an external process it is shown whether a device is connected to the device handler. The (expected) name of the device is also shown.
d3270	Status information for 3270 input and output device handlers. For each pair its state and

the name of the application, if available, is shown. For active device handler pairs also a list of connections to CUs and their states with information on the number of times the connection has been established.

dump *seg offs len* — ADP memory contents. The dump is binary, so an *outfile* is necessary. *seg* and *offs* are segment (paragraph) and offset addresses of the memory area to be dumped; *len* is the number of bytes. The parameters may be given as decimal or hexadecimal numbers. Example:

```
lanstat dump 0 0 16:80000
```

mem — Free memory on the ADP, listed as a number of segments with start address and size.

7. FLOPPY DISK UTILITIES

This chapter describes some floppy disk utilities which can only be used with the RC8352 floppy disk drive and are distributed as part of the ADP SW package. The utilities support the diskette formats that are used with other RC products: RC890/RC891, RC890-30 and RC750 (Partner), in particular three formats: the RC890 format, the CP/M format and the CCP/M format.

The primary intended use of the utilities is copying of files between diskettes and RC8000 backing storage, in particular installation of SW packages which are distributed in the RC890 format. These packages include terminal emulator programs to be downloaded to RC45 and RC855 terminals.

7.1 Fdformat

This utility program formats a diskette. It supports a number of different ways of doing so as described in section 2.4, RC8352 floppy disk drive.

7.1.1 Program Call and Function

The program call is:

fdformat *param*

param ::= help / (*fdproc* (*format-spec*)*)

If the help parameter is given, help information is output. Otherwise the diskette inserted in the drive is formatted as specified.

fdproc ::= Name of the flexible disc process, cf. section 4.1, connect command.

format-spec ::= (dside.(yes/no)) / (ddens.(yes/no))
/ (trackf.(0/1/2/3)) / (std.(yes/no))
/ (speed.(low/high)) / (check.(yes/no))

The dside parameter specifies a double sided (yes, default) or single sided (no) diskette. The ddens parameter specifies double (yes, default) or single (no) density. The trackf parameter specifies the number of bytes per sector, 0: 128, 1: 256, 2: 512 (default), or 3: 1024.

The std parameter specifies IBM standard format for cylinder 0 (yes, default) or the same format as the remaining cylinders, i.e. as specified by the ddens and trackf parameters (no). The IBM standard is: 26 tracks per sector, track 0 always single density. Thus the sector size is 128 bytes on track 0 and either 128 or 256 bytes on track 1 (if double sided) depending on the ddens parameter.

The speed parameter specifies high (default) or low speed. With high speed the format can be specified freely. Note, however, that 1024 bytes/sector cannot be obtained in single density, and 128 bytes/sector not in double density.

With low speed, only the RC890 format is supported (see subsection 7.1.3 below).

The check parameter specifies whether to perform a read check after formatting (yes) or not (no, default).

The informative messages are self-explanatory.

7.1.2 Error Messages

***fdformat syntax

Will appear after a syntactical error and will be followed by the help information.

***fdformat unknown param

The parameter specified is unknown.

***fdformat *fdproc device-status* (cylinder *cyl-no*)

A device status of some sort was encountered. Depending on the error a cylinder number may be specified.

***fdformat incompatible values of trackformat and density

An illegal format combination was attempted.

7.1.3 Floppy Disk Format Usage

Diskettes to be used with the cpm, cpmsys and cpmbak utility programs must be in the CP/M format, i.e. double sided, double density, 512 bytes/sector, IBM standard, high speed (all default values).

Diskettes to be used with the ccpm utility program must be in the CCP/M format, i.e. double sided, double density, 1024 bytes/sector, all cylinders equal, high speed, which may be specified as 'trackf.3 std.no'.

Diskettes to be used with the cpm890 utility program must be in the RC890 format, i.e. double sided, double density, 512 bytes/sector, all cylinders equal, low speed, which may be specified as 'speed.low std.no'.

Diskettes to be used with the fdinit, fdload and fdsave utility programs must not be formatted with 'std.no'.

7.2 Cpm, ccpm, cpm890

These three utility programs are intended for transfer of files between a diskette and RC8000 file storage. See subsection 7.1.3 for a specification of the diskette formats supported by the three programs. In all cases cylinders 0 and 1 are reserved for systems purposes, and the file organization is CP/M-compatible.

The file naming follows CP/M convention, i.e. *filename.ext*, where *filename* is up to 8 characters and *ext* is up to 3 characters. Upper and lower case letters are not distinguished, letters normally displayed in upper case.

The three programs work almost identically. In the following description, the term 'the program' therefore applies to any one of them.

7.2.1 Conversational Call and Function

The programs are intended for conversational use. They therefore prompt for commands and parameters.

The program call is '*prog fdproc*', where *prog* is the program name and *fdproc* is the name of the flexible disc process, cf. section 4.1, connect command.

Upon call, the program reads the catalog part of the diskette and verifies it for consistency. The catalog is written to the diskette after any command that modifies it.

For most commands two files must be specified, a diskette file and an RC8000 file. The program prompts for the file names to be typed.

A command to write to an RC8000 file will overwrite the file if it exists already, otherwise attempt to create it. A command to write to a diskette file that exists already will first delete it, causing the catalog to be written to the diskette, then create it as if it had not existed.

Before a transfer to an RC8000 file is started, the shortclock field of the entry tail is set to the current time. After a successful transfer the bytecount is set in word 10 of the entry tail.

There are 11 commands in all: user, rtxt, rbin, wtxt, wbin, read, readq, del, dir, help and end. Each of these is described below.

user

This command applies only to the ccpm program. It sets current userid to the specified value. Only diskette files with the specified userid are visible when subsequent commands are executed. The default userid is 0.

rtxt

A diskette file is copied to an RC8000 file with text conversion, i.e. CR characters (13) are discarded, and the SUB character (26) is taken as end-of-file indication and converted to EM (25).

rbin

A diskette file is copied to an RC8000 file as a binary file, i.e. no conversion. The bytecount will be a multiple of 128 as this is the logical record size of the CP/M file system.

wtxt

An RC8000 file is copied to a diskette file with text conversion, i.e. NL character (10) are written as the sequence CR (13) followed by LF (10), and the EM character (25) is taken as end-of-file indication and converted to SUB (26).

wbin

An RC8000 file is copied to a diskette file as a binary file, i.e. no conversion. If word 10 of the entry tail indicates a bytecount value (word 9 must be 0), the transfer terminates when the bytecount is exhausted, if this happens before the end of the file.

read

This command applies only to cpm890, not to cpm or ccpm. It works as rbin, but applies to all files on the diskette without prompting for file names. The RC8000 file names are generated from the CP/M file names by converting upper case letters to lower case and omitting the dots between *filename* and *ext*.

readq

This command applies only to cpm890, not to cpm or ccpm. For each file on the diskette it will ask whether the file should be copied to an RC8000 file, and if so, ask whether the file is text or binary and what the RC8000 file name should be and then proceed to copy the file.

del

This command prompts for the name of a diskette file, which will be deleted.

dir

This command reads and verifies the diskette catalog and displays its contents.

help

This command outputs help information.

end

This command terminates execution of the program.

The informative messages are self-explanatory. The prompt for a command is 'select function ?'.

7.2.2 Call From a File

The programs may also be called from a file. Each command is given on a separate line following the program call. All commands

except readq can be used. The syntax is as follows:

command ::= rtxt *files* / rbin *file* / wtxt *file* / wbin *file*
/ del *fd-file* / read / dir / help / end

files ::= *RC8000-file* *fd-file*

RC8000-file is the name of the RC8000 file
fd-file is the name of the diskette file.

The last command must be end.

Note that when the program is called from a file, command prompts will appear in the output among explanatory messages.

7.2.3 Error Messages

In the error messages below *prog* stands for *cpm*, *ccpm* or *cpm890*.

***prog* fp syntax

The program was not called with the correct fp syntax '*prog fdproc*'. The program is terminated.

***prog* syntax

A syntax error was encountered. The first time this happens the help command is called implicitly. The program will prompt for a command to resume operation.

***prog file-name device-status*

A device status of some sort was encountered. File name will be *RC8000-file* or *fdproc*. The program call is terminated.

***prog* connect *RC8000-file error-indication*

The attempt to connect the specified file did not succeed. The reason is given in *error-indication*.

***prog* catalog full

An attempt to create or extend a diskette file went wrong because no free entries could be found in the catalog.

***prog* disc full

An attempt to extend a diskette file went wrong because no free data blocks could be found.

***prog* directory inconsistence blockno *block-no*

The verification of the diskette catalog went wrong. Either the referenced blockno was rubbish, or the block is referred to by more than one file in the catalog. The diskette may only be used for read access.

***prog* directory inconsistent - only rtxt, rbin, dir and end are allowed

Attempt to use wtxt, wbin or del function after a directory inconsistence.

7.2.4 Installing Files From an RC890 Diskette

This subsection gives a brief self-contained description of how to install files, e.g. terminal programs, from a diskette in the RC890 format.

1. Make sure the flexible disc process is created and linked to the floppy disk handler and note its name (*fdproc*).

2. Note the CP/M file names of the files to be installed and the file names they should be given on the RC8000.

3. Insert the diskette in the drive and give the command 'cpm890 *fdproc*'.

4. When cpm890 prompts for function, type 'readq', and then as the program asks about each file on the diskette, specify whether to read it, whether it is text or binary, and whether the name should be as suggested.

5. Give the end command to terminate cpm890.

7.3 Cpmsys

The program cpmsys copies the part of a CP/M diskette that is not included in the file system (cylinders 0 and 1) to or from an RC8000 disk file.

7.3.1 Conversational Call and Function

The cpmsys program is intended for conversational use. It therefore prompts for commands and parameters.

The program call is 'cpmsys *fdproc*', where *fdproc* is the name of the flexible disc process, cf. section 4.1., connect command.

There are 4 commands in all: read, write, help and end. Each of these is described below.

read

If the RC8000 file does not exist cpmsys attempts to create it. Before the transfer shortclock is inserted in the entry tail. The contents of diskette cylinders 0 and 1 are copied to the RC8000 file. Upon successful completion the number of bytes transferred is written in word 10 of the catalog entry tail and word 9 is set to 0.

write

The contents of the RC8000 file are copied to the diskette starting at the first sector of cylinder 0 and ending at the last sector of cylinder 1, or when the file has been exhausted, whichever happens first. If word 10 of the catalog entry tail holds a bytecount (word 9 of the entry tail must be 0) it will be taken to indicate the length of the file.

help

This command outputs help information.

end

This command terminates execution of the cpmsys program.

The informative messages are self-explanatory. The prompt for a command is 'select function ?'.

7.3.2 Error Messages

***cpmsys fp syntax

The program was not called with the correct fp syntax ('cpmsys *fdproc*'). The program is terminated.

***cpmsys syntax

A syntactical error was encountered. The first time this happens the help command is called implicitly. The program will prompt for a command to resume operation.

***cpmsys *file-name device-status*

A device status of some sort was encountered. The file name may refer to the RC8000 file or to the flexible disc process. The program call is terminated.

***cpmsys connect *RC8000-file error-indication*

The attempt to connect *RC8000-file* did not succeed. The reason is given as *error-indication*.

7.4 Cpmbak

This program may be used to maintain backup copies of diskettes in the CP/M format.

7.4.1 Conversational Call and Function

The cpmbak program is intended for conversational use. It therefore prompts for commands and parameters.

The program call is 'cpmbak *fdproc*', where *fdproc* is the name of the flexible disc process, cf. section 4.1., connect command.

There are 4 commands in all: read, write, help and end. Each of these is described below.

read

If the RC8000 file does not exist cpmbak attempts to create it. Before the transfer, shortclock is inserted in the entry tail. The entire contents of the diskette are copied to the RC8000 file. Upon successful completion the number of sectors transferred is written in word 10 of the catalog entry tail, and word 9 is set to 0.

write

The contents of the RC8000 file, which is assumed to be the result of a previous call of cpmbak using the read function, is copied to the diskette.

help

This command outputs help information.

end

This command terminates execution of the cpmbak program.

The informative messages are self-explanatory. The prompt for a command is 'select function ?'.

7.4.2 Error Messages

***cpmbak fp syntax

The program was not called with the correct fp syntax ('cpmsys *fdproc*'). The program is terminated.

***cpmbak syntax

A syntactical error was encountered. The first time this happens the help command is called implicitly. The program will prompt for a command to resume operation.

***cpmbak *file-name device-status*

A device status of some sort was encountered. The file name may refer to the RC8000 file or to the flexible disc process. The program call is terminated.

***cpmbak connect *RC8000-file error-indication*

The attempt to connect *RC8000-file* did not succeed. The reason is given as *error-indication*.

8. ADP/LAN TERMINALS AND PRINTERS

This chapter contains information for users of terminals and printers which:

- are controlled by an ADP as control unit, i.e. directly attached via Circuit or V.24, and/or
- communicate with an RC8000 host via RcLAN or directly via an ADP.

Section 8.1 describes the use of the terminal download function. This section applies to a multifuntion terminal with an ADP as control unit. Section 8.2 describes how a terminal with an ADP as control unit works as CSP terminal. Section 8.3 gives information about CSP printers controlled by an ADP. The use of 3270 devices controlled by an ADP is not described in this guide, but in the documentation for SW8703/1.

Section 8.4 describes how an RC8000 host communicates with a CSP terminal. This description is also relevant for CSP terminals which are not attached to an ADP, for example Partners. Section 8.5 describes how to make entries in the main catalog which let PRIMO or BOSS use a CSP printer or let PRIMO use a printer spooler on another system attached to the LAN. Section 8.6 describes how an RC8000 host works with 3270 terminals.

8.1 Terminal Download

The terminal may be configured to show a menu from which the user may select a program to be downloaded. The menu is self-explanatory. The terminal may also be configured to bypass the menu, provided it contains a default program. Terminal configuration is described in the user's guide for the terminal.

Certain error situations may occur during download. A message will then be shown in the status line (bottom line) of the terminal. The messages that are shown may be changed by the system administrator. The explanations that follow use the default (unchanged) texts:

- Disk access error

  The ADP failed to establish a connection with the FTS server. The error could be due to a temporary lack of resources, so retry a couple of times.

- Program not found

  The program file was not found. Most likely it has not been installed. The reason could also be a customization error, e.g. the file name was misspelled in the terminal program specification.

- CU disconnected

  The connection between the ADP and the terminal was broken. The problem may be temporary, and a couple of retries should be attempted.

Each data block transmitted on Circuit-I is protected by a redundancy check and retransmitted in case of error. An excessive number of retransmissions, which may occur if the cable is in poor condition, causes the connection to be broken.

The connection is also broken, as seen from the terminal, if the ADP ceases to operate.

- Checksum error

  An error was detected when a checksum was computed on the complete program after all blocks had been transferred to the terminal. This message therefore does not indicate a transmission problem. The reason may be that a bad program file was read, or possibly a RAM error within the terminal.

8.2 CSP Terminals Controlled By an ADP

The ADP must be RC8420 and the ITC channels used for terminals must be customized in file itccst.

A character-oriented terminal is required, either a fixed-function terminal connected to the ADP via a V.24 port or Circuit-II or a multifunction terminal connected via both Circuits with an ANSI X3.64 or RC851 terminal program. In the latter case Circuit-I is used for downloading the terminal and Circuit-II for the CSP terminal function.

The CSP terminal function is based on a program in the ADP which connects the terminal to a host computer. When the terminal is ready, i.e. its program has been loaded, the user should strike a key to alert this program. It will then display a menu of the available hosts, i.e. those that have announced themselves on the LAN. The menu is self-explanatory. When the user has selected a host, the ADP will make the connection. The ADP may also be customized to connect to a specified host automatically without showing the menu.

When a connection has been made, the host will usually issue a prompt. The prompt depends on the host. From this point on, communication is with the host and the CSP function is transparent.

Normally the user logs in to an access system such as RC8000 TAS or operating system on the host. When the user logs out of this system, the connection between terminal and host is usually broken by the host. When this happens, communication is again with the CSP terminal program in the ADP, and a new connection may be established.

8.3 CSP Printers Controlled By an ADP

The ADP must be RC8420 and the ITC channels used for terminals must be customized in file itccst. A printer with a serial interface may be attached to one of the V.24 ports. A printer

with a parallel interface may be attached via an RC3934 Adapter to Circuit-II. In the latter case, note that if the RC3934 is not operative, the ADP will react as if the printer is powered off, i.e. report Intervention status.

8.4 Communicating With an RC8000 Host From a CSP Terminal

When a CSP terminal is connected to an RC8000 host, and the host has resources to create a terminal process, it will display the message 'connected to *hostname*'. Otherwise a message which indicates lack of resources will be shown.

If the host system uses TAS, the terminal will then automatically begin communication with TAS. Otherwise the user must generate an attention by typing an interrupt character (normally ESC) to begin interaction with an internal process. The interrupt character is echoed as 'att', and the operator must then type the name of the desired internal process terminated by RETURN. If the terminal is reserved by a process, this dialog is omitted, and an attention is just sent to the reserving process.

The CSP host function keeps track of the internal process which is currently interacting with the terminal. The relevant interactions are: output from process to terminal, input or attention from terminal to process, or reservation of the terminal. When the current process changes, i.e. before output from a new process is displayed or input to a new process is accepted, the CSP host function writes 'from *process*' or 'to *process*' on a separate line, so that the operator will know the source or destination of the data.

To send an attention to the current process it is unnecessary to type the process name, i.e. if the 'att' prompt is answered with an empty line, an attention is sent to the process which most recently interacted with the terminal.

The interrupt character may be customized on the ADP. It will be the same for all CSP terminals communicating with the RC8000 host via the same ADP.

The RC8000 CSP host function supports three commands which were also found on older front-ends:

CTRL-ø st	show *hostname*
CTRL-ø re	remove terminal connection
CTRL-ø di	display available hosts

These commands are intercepted by the ADP, and there is no interaction with internal processes.

When one of these special commands or the name of a process to receive an attention message is expected, the terminal is in a special mode, where normal processing of input/output to/from internal processes is suspended. In order to obey standard timeout rules towards internal processes, the CSP host function imposes a time limit of 15 seconds on the typing of the command following ctrl-ø or the process name.

The internal process may set the terminal mode to be conversational or canonical. In particular, TAS may be instructed to set the terminal mode.

When the mode or some other terminal attribute, such as echo, is being changed on the instruction of an internal process, input already typed will be flushed, even if it has been echoed (canonical mode). The interrupt character may also be temporarily ignored at such times.

In conversational mode, which is the traditional RC8000 terminal mode, input is only accepted when it is specifically requested by an internal process. In canonical mode, which is provided because it will work better for terminals connected via a PAD and a packet switched network, input is always accepted and buffered by the ADP. However, if the input buffers of the ADP are overwhelmed, input may be discarded even when it has already been echoed. When this happens a message will be produced by the ADP to inform the user. If the user always waits for the prompt from the program (usually the bell) before typing input, the situation will not arise.

Backspace may be used to edit current input line, and the whole line may be regretted by ENQ (ctrl-E).

The RC8000 host breaks the connection when the last internal process which is a user of the CSP terminal process is removed or dissociates itself from the terminal. Normally this is when the user logs out of TAS.

8.5 Using CSP and FTS Printers With PRIMO and BOSS

CSP printers

PRIMO and BOSS can both use CSP printers. The printer shall be described in the main catalog by an entry created as follows:

ename = set *lpmode pname* d.0 0 0 0 0

ename is the entry name which is used to refer to the printer.

lpmode is one of the following:

2048.14	slow, no newline conversion	(mode 0)
2050.14	fast, no newline conversion	(mode 2)
2064.14	slow, newline conversion	(mode 16)
2066.14	fast, newline conversion	(mode 18)

Fast means the CSP printer only sends status reports when there is an error status. Slow means a status is sent for every data block which is transmitted across the LAN and allows tighter recovery control. Fast is generally recommended. If the actual printer has a large buffer, tight control is illusory anyway.

If the printer is customized to perform a newline function (both CR and LF) when it receives an LF character, no newline conversion should be specified. If the printer cannot perform this conversion, select newline conversion. The ADP will then insert the extra CR characters.

pname is the name of the CSP printer process. Normally the link is created and the name assigned using makelink.

Example:

```
lp7 = set 2066.14 printer7 d.0 0 0 0 0
```

When this entry exists the printer may be referred to as lp7, e.g.:

```
filexfer pip lp7
```

The example above uses PRIMO. If BOSS is used the name of the main catalog entry may be used to refer to the CSP printer as a remote printer (note that BOSS requires the name to be at most six characters long).

BOSS can also use a CSP printer as a standard printer. In this case the device number of the printer must be specified in the customization of BOSS, and the printer process must therefore always be created with the same device number (makelink). The printer mode for BOSS standard printers can also be customized. The effect of printer modes for CSP printers is indicated in parentheses above under *lpmode*.

FTS printer via server on another system

By making use of FTS, PRIMO can also use printers which are attached to other systems on the LAN. These systems must have FTS servers interworking with local spoolers. When PRIMO submits a file for printing by a spooler on another system, PRIMO does not have control of the spool queue and printing process. The spool queue must be monitored on the system where the spooler runs. This may for example be a Partner or RC900 system, or it may be PRIMO on another RC8000.

The feature requires a main catalog entry to describe the server plus an additional entry for each printer attached to the server. The server entry is necessary to hold information which PRIMO needs, but which will not fit into the printer entry.

The server entry is created as follows:

sname = set ip *server* d.0 *adpno* 0 0 0

and each printer entry as follows:

ename = set lp *printer* d.0 0 0 12.*cc* 0

server is the FTS server name.

adpno is the number of the ADP which provides access to the appropriate LAN, normally 1.

printer is the printer name known to the FTS server.

sname and *pname* are catalog entry names. *cc* must be a number. *sname* must be *cc* characters long. *ename* must be longer, but with the first *cc* characters equal to *sname*.

Example:

```
romeo = set ip romeo d.0 1 0 0 0

romeop7 = set lp lp5 d.0 0 0 12.5 0
```

When these entries exist the printer known to the server romeo as lp5 may be referred to as romeop7, e.g.:

```
filexfer pip romeop7
```

8.6 Communicating With an RC8000 Host From a 3270 Terminal

The terminal must be connected via a control unit providing the 3270 CU function. The control unit maintains a menu of communication lines and/or applications in the LAN environment to which each terminal device may be connected.

An RC8000 application process may, via links to a 3270 device handler pair on the ADP, make itself reachable from 3270 CUs attached to the LAN. The application will be shown in the menus and may be selected by terminal users. Communication will then be transparent between the application and the terminal device.

When the operator has pressed an attention key, thereby causing the 3270 CU to send an input message to the host, this message will be buffered on the ADP and delivered to the application when it issues an input operation to read the message. In this situation the ADP activates a timer, and if the timer expires before the message has been read, the ADP will discard the input message and show the message "Host timeout, data lost" in the terminal status line. The length of the timer may be customized. Normally it should be set so that the message will only be shown when a serious error has occurred on the RC8000. The best value for the timer depends on application response times.

To remove the timeout message, generate a Reset attention from the keyboard. If the application is still running, resend the input message by means of the same attention key which was used in the first place. Note, that a relatively short timeout may be used to detect that an application is running slowly.

A. REFERENCES

Part numbers in references to RC documentation are subject to change as new editions are issued and are listed as an identification aid only. To order, use package number.

References (1)-(3) are all part of SW8740/1-D, together with this guide.

(1) PN: 991 11384
RC8000/RC9000-10 FTS User's Guide

Provides a general description of FTS, file transfer service, and describes the user interface on RC8000.

(2) PN: 991 11262
LAN Device Processes, Reference Manual

Describes the external processes for character-oriented (CSP) terminals and printers and the 3270 input and output processes as well as the link management operations supported by the ADP main process.

(3) PN: 991 10543
RC8000/ADP Flexible Disc Process, Reference Manual

Describes the external process for the RC8352 floppy disk drive.

(4) PN: 991 11265
Package number: SW8010/1 or SW8010I-D
System Utility, Maintenance Programs

Describes a large number of utility programs for system administration, including makelink and deletelink.

(5) PN: 991 10244
Package number: SW8900-D
RC890/RC891 Control Unit, User's Guide

(6) PN: 991 11099
Package number: SW8934-D
RC890 CSP Communication, User's Guide

B. EXOTIC CUSTOMIZATION PARAMETERS

This appendix has one section for each of the relevant parameter files:

B.1 Parameters In the Device Handler File (adpcst)

Parameters for ADP3270 input and output handlers

Number of buffers: 3270NBUFS=no

3270NBUFS specifies the number of buffers each input device handler uses to receive input from terminals. No must be in the range 0..5, and has the default value 3.

Input message size: 3270RSIZE=size

3270RSIZE specifies the length in bytes of the buffers used for input from the terminals. Size must be in the range 0..3860, and has the default value 256. Note that after initialization the length of the data units exchanged between the CU and the device handler is negotiated, and the actual buffer size will be determined by the smallest buffer. Memory is therefore best utilized if both buffer sizes are the same.

Output message size: 3270TSIZE=size

3270TSIZE specifies the length in bytes of the buffers used for output to the terminals. Size must be in the range 0..3860, and has the default value 1024. Like the input buffers the size of data units exchanged between the CU and the device handler is negotiated, and the buffers should be of the same size.

B.2 Parameters In the Terminal Download File (menudlcst)

In the terminal download menu (cf. the illustration in section 3.6), three texts frame the menu:

1. TERMINAL FUNCTION MENU
2. Key Description
3. Press function key to select application

These "frame texts" appear in fixed positions on the display, but may be modified using the FTEXT parameter specification. They will appear identically on all the terminals. The key names (PF1, PF2, PF3) cannot be modified; they correspond to engravings on the keys.

Frame text: FTEXT,*ftno=text*

FTEXT specifies one of the frame texts of the download menu. *ftno* is a number: 1, 2 or 3, which identifies the particular text. The numbering is as indicated above for the default texts. *text* may be at most 45 characters long.

Error text: ETEXT,*etno=text*

ETEXT specifies a text which may occur as an error message in the status (bottom) line of the terminal in case of an error during download. *etno* must be a number in the range 1..4, and the *text* may be at most 25 characters long. The error situations and the default texts are explained in section 8.1.