
RC8000 ADP/LAN Device Processes:

CSP Terminal and Printer

Reference Manual

PN: 991 11001

RC Computer

Keywords:

RC8000, IFP, ADP, RC8402, RC8410, RC8411, RC8420, SW8740/1, RcLAN, CSP, terminal process, printer process.

Abstract:

This manual describes two of the external processes which may be used by RC8000 internal processes to communicate with LAN devices via an attached device processor (ADP): CSP terminal and CSP printer. The link management operations supported by the IFP/ADP main process are also described.

Date:

April 1988

**Copyright © 1988, Regnecentralen a.s/RC Computer a.s
Printed by Regnecentralen a.s, Copenhagen**

Users of this manual are cautioned that the specifications contained herein are subject to change by RC at any time without prior notice. RC is not responsible for typographical or arithmetic errors which may appear in this manual and shall not be responsible for any damages caused by reliance on any of the materials presented.

TABLE OF CONTENTS**PAGE**

1. INTRODUCTION	1
2. LINK MANAGEMENT	3
2.1 Connect Operation	3
2.2 Disconnect Operation	5
3. CSP TERMINAL PROCESS	7
3.1 General Description	7
3.2 Interrupt Character and Attention Message	11
3.3 Sense Operation	12
3.4 Set Attributes Operation	12
3.5 Lookup Attributes Operation	13
3.6 Output Operation	13
3.7 Input Operation	13
4. CSP PRINTER PROCESS	15
4.1 General Description	15
4.2 Output Operation	16
4.3 Sense Operation	17

APPENDICES

A. REFERENCES	19
B. MESSAGE FORMAT AND RESULT CONVENTIONS	21
C. CSP TERMINAL MESSAGE/ANSWER SUMMARY	23
D. CSP PRINTER MESSAGE/ANSWER SUMMARY	25

1. INTRODUCTION

This manual describes two of the external processes which may be used by RC8000 internal processes to communicate with LAN devices via an attached device processor (ADP) and the link management operations supported by the IFP/ADP main process.

The manual is intended for programmers who need to deal directly with these external processes. The reader is assumed to be familiar with the ADP and related concepts such as LAN device and device handler. A general description is given in (ref.1). Familiarity with RC8000 Monitor (ref.2) is also assumed.

The LAN devices are:

- character-oriented CSP terminals and printers,
- format-oriented 3270 terminals and printers,
- IMC ports.

CSP and 3270 terminals and printers are physical devices with which the RC8000 system communicates across RcLAN. IMC ports are not physical devices, but provide a similar programmatic interface which may be used for inter-process communication across RcLAN.

An internal process issues an operation request to one of these external processes by sending a message specifying the operation. When the operation has been executed the message is answered with result and status information.

The external processes for these devices work in cooperation with device handlers running on an ADP. Each external process has a link to the matching device handler. Normally the operation requests sent to one of these external processes are passed on (by Monitor) to the device handler via the link and also answered by the device handler. Links are normally created on the initiative of an internal process. CSP terminals are an exception to this rule, as explained below.

The existence of the device handler is, however, irrelevant to the internal process. It only sees the send message/wait answer interface to the external process.

The actual terminals and printers may be connected to any control unit which performs the CSP terminal or 3270 CU function and is attached to the same LAN as the ADP. As a special case the ADP may also serve as control unit. In order for RC8000 to use a device a connection between the device handler and the control unit, normally across the LAN, is required.

In the case of a CSP terminal the connection is established by the control unit on the initiative of the terminal user. For a CSP printer the connection is also established by the control unit which must be customized to know which host to connect to. When a CSP connection has been established it may be matched with an existing link, or, in the case of a terminal, it may cause a link to be created.

For the other device types the connection across the LAN is established by means of an operation request after the link has been created.

Chapter 2 deals with management of external processes and the associated links. Chapters 3 and 4 describe the external processes for CSP terminals and printers.

The IMC port process and the external processes for input and output from and to 3270 terminals and printers are not described in this manual, but currently have their own manuals (refs.3 and 4). A future edition of this manual will be revised to also cover these processes.

General message format and result conventions which are obeyed by all the external processes are described in Appendix B.

2. LINK MANAGEMENT

Often the application of a LAN device will be static in the sense that the external process and the associated link may be permanently established, and the user program does not need to control the establishment of the link. In such cases the appropriate way to create the link and the external process is to use the utility program crossload, which is described in (ref.1).

When, due to more dynamic requirements, this method is not sufficient, the user program may manage creation and removal of external process and links by issuing *Create Link* and *Remove Link* operations to the IFP/ADP main process.

Links may also be removed using crossload. Usually, however, it is not necessary to explicitly remove a link, as this is done by Monitor when the last internal process which is a user of the link (external process) is removed. This is true for all the LAN devices.

When a link is removed all queued operation messages are returned with result malfunction (cf. Appendix B) and the external process is also removed.

The main process and its name is preconfigured as part of the Monitor. Systems with multiple ADP's have a main process for each of these. By convention, these processes are named 'ifpmain1', 'ifpmain2' etc.

The number of external processes for ADP devices is also preconfigured in the Monitor. Initially these processes are unused, i.e. they have kind=68, indicating an external process which currently does not exist, but may be used for an ADP device. When a link is created to a device handler on an ADP, one of these external processes is always used. When a link is removed, the external process again becomes unused and may be reused for another link.

It should be noted that the *Create Link* and *Remove Link* operations do not create and remove the device handlers. The configuration of the ADP with respect to types and numbers of device handlers is static. How an ADP is customized with the appropriate device handlers is described in (ref.1). The device handlers survive link removal and may be relinked any number of times, but are, in general, reset when a link is removed, i.e. they lose any memory of past operations.

2.1 Create Link Operation

The operation creates an RC8000 external process with a link to a device handler on an ADP. The kind of the external process is changed to the appropriate value for the device in question. When this has been done, operations may be issued to the device via the external process.

The internal process must be user of the IFP/ADP main process.

Create Link message:	answer:
+0: 6<12 + users	+0: status word
+2: device type	+2: RC8000 device number
+4: device index	
+6: RC8000 device number	
+8:	
.. CSP device name	
+14:	

device type: Type of the device on the ADP, identifies the relevant device handler (some of the device handlers are not described in this manual, cf. ref.1):

- 1: CSP terminal handler (console)
- 2: IMC port handler
- 3: mailbox handlers
- 4: 3270 input handler
- 5: 3270 output handler
- 6: test device handler
- 7: floppy disk handler
- 8: CSP printer handler
- 9: streamer handler

device index: Number of the device handler incarnation to be used; 255 specifies the first free device handler of the given type.

RC8000 device number: Number of the external process to be used; -1 specifies the first unused external process with kind=68. In the latter case the answer message indicates the chosen device number.

CSP device name: This field is only used for CSP devices. It contains the name of the CSP device to be reached by the link. The link must be created prior to the connection between the device (control unit) and the device handler and can only be used to communicate with the specified device. Device names are explained in (ref.1). The name must consist of at most 10 ASCII characters, and the remaining part of the name field must contain nulls.

The status word is used as follows:

<u>Bit no.</u>	<u>Meaning when set</u>
10-15	Create Link result:
	0: OK
	1: Device trouble (see device status)
	3: No Monitor resources
	4: No IFP resources or no device handler
0- 7	Device status (when Create Link result = 1):
0	Device unknown
3	Device selftest fault
4	Resource problem

If users (the operation modifier in the first word of the *Create Link* message) is 0, only the sender process will become user of the external process. If users is 1, all users of the IFP/ADP main process will become users of the external process. Unless the intention is that the external process shall be generally and permanently available, users should be set to 0. Crossload creates links with users set to 1 to achieve general availability. If only the internal process which issues the *Create Link* operation really needs the link, it is normally desirable to have the link removed automatically along with the process. This is, however, impossible if there are other users.

Result 2, rejected, will occur if the internal process is not user of the IFP/ADP main process, or if another process has reserved the main process.

2.2 Remove Link Operation

The operation removes an RC8000 external process with a link to a device handler on the ADP and the link. The external process becomes unused (kind=68) and may be reused for another link.

The internal process must be user of the external process, or there must not be any users at all. It must also be user of the IFP/ADP main process. Otherwise the message will be rejected (result 2).

Remove Link message:	answer:
+0: 10<12	+0: 0
+2: RC8000 device number	

RC8000 device number: Number of the external process description.

Result 3, unintelligible, will occur if the device number does not identify an external process with a link to a device handler on the ADP associated with the IFP/ADP main process to which the message is sent.

3. CSP TERMINAL PROCESS

The first section of this chapter gives a general description of the way CSP terminals work. The remaining sections describe the format and meaning of the messages which are exchanged between an internal process and the CSP terminal process. The *Attention* message is sent unsolicited from the CSP terminal process. All the other messages are operation messages which are sent by an internal process and answered by the CSP terminal process.

3.1 General Description

A CSP terminal is represented in RC8000 as an external process with `kind=8`.

An internal process becomes user of a terminal process when it initializes a CSP terminal process, issues an operation or receives an attention message. Normally it receives an attention message when a connection has been made (see below).

A terminal process will accept operations from multiple internal processes, provided no internal process has reserved it.

The following operations are supported: *Sense*, *Set Attributes*, *Lookup Attributes*, *Output* and *Input*. In addition the external process may send *Attention* messages.

The *Set Attributes* operation may be used to change operational attributes of the terminal, i.e. operational mode (see below), echo, prompt and timeout.

The device handler performs one operation at a time and all operations strictly in the order they are issued. Operations sent in addition to the one being performed will be queued. In particular operations issued after after an *Input* operation are not performed until the *Input* operation completes.

Terminal connection

A link to a CSP terminal device handler may be created using the *Create Link* operation (cf. chapter 1). The operation must specify the device name of the CSP terminal, and the resulting link can only be used to communicate with the specified terminal. Operations will not be accepted until the terminal has been connected; see below. Normally only console links are created in this way, but the facility is general.

The connection between a terminal and the device handler is always established on the initiative of the terminal. The terminal will then identify itself - by device name - to the device handler. If the device name matches the one specified for an already created link, the connection and the link will be joined, and operations will then be accepted.

Normally, the device name will not match, and in this case a new CSP terminal process is created using a free (unused) IFP/ADP

external process (kind=68), and a link is created between the device handler and this process. The name of the CSP terminal process becomes the terminal's device name prefixed with a '0' character (this is to avoid confusion with names of files and other processes).

RC8000 Monitor provides a special feature used by TAS whereby all CSP terminals, which are not connected to already created links, are reserved by TAS when they are connected, and TAS is notified. If such a reservation is not made, i.e. on systems without TAS and for terminals connected to pre-created links, no internal process will be notified. An attention message must then be generated before any communication can take place between the terminal and an internal process (cf. section 3.2).

If a CSP terminal link is removed by an internal process or by Monitor when the last user process is removed, the terminal is disconnected.

If the control unit removes the connection - normally on the user's initiative - or the connection is removed because of network failure while the terminal process still has user(s), the link is removed if it was created on the terminal's initiative. A link created using *Create Link*, will not be removed by the device handler, but reverts to its original state before the terminal - was connected.

CSP terminal implementations

A CSP terminal can be implemented in several different ways:

- It can be a character terminal (RC45, RC851 or VT100-like) connected via a control unit attached to the LAN, e.g. RC890-30, or an RC8420 ADP. The terminal will be connected to the control unit by means of a V.24 or Circuit-II serial link.
- It can be a remote terminal connected via a packet switching (X.25) network. The terminal will be connected to a remote (triple X) PAD. In this case the unit attached to the LAN will be an X.25/X.29 gateway. The gateway and the PAD will communicate according to X.29 across the X.25 network.
- It can be a workstation or PC with a terminal emulator directly attached to the LAN.

The term *control unit* is used extensively in the following. In the case of a remote terminal connected via a PAD, it will refer to the gateway. In the case of a PC- or workstation-based CSP terminal the control unit shall be thought of as a part of the terminal emulator function integrated in the PC or workstation.

Note that the actual processing and buffering involved in performing the operations requested from the terminal process is carried out by Monitor, by the ADP, by the control unit, and possibly even by the PAD, in some combination. How the work is shared is irrelevant to the internal process. It only sees the

external process which is an abstraction for the whole terminal communication complex. However, some operational modes will not work efficiently for remote terminals connected via PAD's. This is discussed below.

Character set

The terminal character set is assumed to be 7-bit ASCII or an 8-bit extension thereof. However, except for a few control characters, all characters are passed unchanged between the internal process and the terminal. The cases where conversions occur are explicitly described below. Control characters are also recognized as input terminators.

Operational modes

The external process can operate in three major modes (note that the word mode as used in this document does not designate a modified form of an operation as in traditional RC8000 parlance):

- conversational,
- canonical, and
- file transfer.

The operational mode is changed by a *Set Attributes* operation. It affects the way in which input operations are performed.

Both conversational and canonical mode are line- (or field-) oriented. As characters are typed they are placed in a line buffer in the control unit or PAD. Normally they are echoed as they are placed in the buffer. An input message is terminated by a CR or other field-terminating character. The contents of the line buffer are subsequently transferred to the input area pointed out by an *Input* operation issued by the internal process.

Line editing is provided in canonical mode. Conversational mode can be specified with or without line editing. Line editing means:

- The CR or NL character terminates input. It is echoed as CR followed by NL and passed to the internal process as NL.
- The BS character when received in input causes the last character, if any, to be deleted from the line buffer. It is echoed as BS SP BS.
- The ENQ character when received in input causes the entire contents of the line buffer to be deleted. It is echoed as CR NL.

If the line buffer becomes full, its contents are passed to the RC8000 and can no longer be edited.

In conversational mode without line editing control characters terminate input. Control characters have the values 0..31 and 127. The terminating control character is passed to the internal process, but not echoed.

Conversational mode is the traditional method for RC8000 internal processes to interact with terminals. In this mode input characters are not accepted except when an *Input* operation has been issued. Depending on the control unit, a few characters may be stored in an intermediate buffer to allow the user to type ahead, but they will not be echoed as long as an *Input* operation has not been issued. Thus the operator does not get the impression that the characters have been accepted. The general pattern is that the internal process and the operator communicate by alternating turns.

This method was well suited in a system architecture where the host or front-end processor controlled the terminals directly. Alternating input and output turns can also be implemented across a LAN, but not across a packet switching network to a triple-X PAD. As a consequence, conversational mode works very inefficiently with a terminal connected by way of a PAD. In order to properly control input processing, the gateway must request the PAD to effectively transmit each character unechoed in a separate packet. This clearly makes poor use of the packet switching network.

Canonical mode provides line-oriented communication with no alternate turns enforced. Input can always be typed and will be accepted and normally echoed, whether or not the internal process has made an input area available by issuing an *Input* operation. If not, input lines will be buffered until an *Input* operation is issued. The amount of input that can be buffered depends on the buffering capacity of the device handler.

When the terminal operates in canonical mode the operator should not assume that input has been accepted by the internal process because it is echoed. However, if prompting is used, and the operator waits for the prompt before typing, she can be sure to be in step in with the internal process.

Canonical mode is natural for terminal control units which are not directly controlled by the host. In particular it fits the behaviour of a triple-X PAD.

It makes no sense for several internal processes to share a terminal process operating in canonical mode. Typically, the external process will therefore be reserved.

In file transfer mode input is accepted at all times (as in canonical mode) and buffered. There is no conversion of input and ETX is the only terminating character.

Buffered input which may be present in canonical or file transfer mode is flushed when an attention is generated (canonical only), when the operational mode is changed, and when the echo attribute of the terminal is changed. Note that this can be done with the operation modifier of an *Input* operation (blind, see section 3.7) as well as with a *Set Attributes* operation.

Status word

The status word is used as follows:

<u>Bit no.</u>	<u>Meaning when set</u>
2	Input timeout
7	Attention
11	Terminal not connected

Bit 11 will be set regardless of operation type, and is not discussed under the individual operations in the sections 3.2-3.6. It will occur when there is no terminal connected to the link.

3.2 Interrupt Character and Attention Message

The interrupt character is normally ESC. It can be changed by customizing the device handler (ref.1). Its occurrence in input is used to attract the attention of an internal process.

In file transfer mode, bit 7 will be set in the answer to the next *Set Attributes*, *Input* or *Output* operation which is returned after the interrupt character is received.

In the other modes, if the device handler is performing an *Input* operation when the interrupt character is received from the terminal, this operation is answered with status bit 7 set. More importantly, buffered input is flushed and an *Attention* message will be sent, as described below.

If the terminal process is reserved, the *Attention* message is sent to the reserving internal process. If the terminal process is not reserved, the text 'att' is displayed on a new line on the terminal and the subsequent input line is taken as the name of the internal process with which the operator wishes to communicate. If such a process is found, an *Attention* message is sent to it using the terminal's process description address as sender address. If no process is found the message 'unknown' will be displayed on the next line. If no name is typed (the line is empty), the message is sent to the same process which received the previous *Attention* message from the terminal process.

Attention message:

+0: 0

+2: status

status: 4 if an *Input* operation was returned with status *Attention*, otherwise 0. This information is only relevant if the terminal process is reserved.

The process which receives an *Attention* message must answer it. The contents of the answer is dummy. There will never be more than one unanswered *Attention* message from a CSP terminal process at a time. The action which occurs if an additional interrupt character is received from the terminal before the answer to the

previous Attention message has been received from the internal process depends on whether the CSP terminal process is reserved. If so, the interrupt character is ignored (the operator must be patient!). Otherwise, the previous Attention message is regretted and a new one is sent to the indicated process.

3.3 Sense Operation

The Sense operation is dummy.

Sense message:	answer:
+0: 0	+0: 0

3.4 Set Attributes Operation

The operation changes the attributes of the terminal according to the contents of the message.

Set Attributes message:	answer:
+0: 132<12	+0: status word
+2: echo<11 + mode	
+4: 0	
+6: prompt<8	
+8: 0	
+10: 0	
+12: input timeout<8	

echo: 1 = echo enabled (default)
0 = echo disabled

mode: 1 = conversational with line editing (default)
2 = conversational without line editing
3 = canonical
9 = file transfer

prompt: >0 = value of prompt character (default is 7=BEL)
0 means prompt disabled

input timeout: specified in seconds; meaning depends on mode:
conversational: operation timer
canonical: operation timer
file transfer: inter-character timer, max.value 25; a 0 value disables the timer.
Default timeout value is 60 seconds.

If a Set Attributes operation changes the echo or mode attribute, any buffered input is flushed.

In file transfer mode, status bit 7 will be set in the answer if an interrupt character was received in input. This is not an error, only informative.

3.5 Lookup Attributes Operation

The operation reads the current attributes from the terminal in the same format used for Set Attributes.

Lookup Attributes message:	answer:
+0: 134<12	+0: 0
	+2: echo<11 + mode
	+6: prompt<8
	+12: input timeout<8

3.6 Output Operation

The operation causes the characters in the output area, the text, to be written to the terminal. The characters must be stored three per word. Thus the number of characters will always be a multiple of three. Padding with NUL characters must be used if necessary.

Output message:	answer:
+0: 5<12 + transp	+0: status word
+2: first storage address	+2: number of halfwords
+4: last storage address	+4: number of characters

When an *Output* operation is performed in a mode other than file transfer mode, the name of the issuing process is first displayed on a separate line as 'from process name', but only if it differs from the name of the last process that communicated with the external process.

If transp is 0, output is converted as text, i.e. any occurrence of CR or NL in the data will be transmitted to the terminal as CR followed by NL. Otherwise (transp = 2 or 4) no conversion takes place.

The answer to an *Output* operation is sent to the user as soon as the text has been sent to the control unit. The halfword and character counts reported in answer messages concern the text that was sent to the control unit, and do not represent a guarantee that all the data has actually been transmitted to the terminal.

In file transfer mode, status bit 7 will be set in the answer if an interrupt character was received in input. This is not an error, only informative.

3.7 Input Operation

An input message makes an input area available for reading input characters.

Input message:	answer:
+0: 3<12 + trail<4 + blind	+0: status word
+2: first storage address	+2: number of halfwords
+4: last storage address	+4: number of characters

Trail is a 2-bit field. It specifies a number of character locations at the end of the input area which are not used for input characters. Thus it possible to control exactly the number of characters which can be received.

The modifier blind can be used to override the echo attribute of the terminal. If blind is 8, echo is effectively disabled while the *Input* operation is performed. If blind is 0 the echo attribute specified using *Set Attributes* has effect.

When an *Input* operation is performed in a mode other than file transfer mode, the name of the issuing process is first displayed on a separate line as '*to process name*', but only if it differs from the name of the last process that communicated with the external process. If prompting is enabled, the prompt character is then output. Subsequently, input characters from the terminal are accepted. The size (number of characters) of the line buffer into which characters are received is set equal to the size of the input area (as modified by trail).

In canonical and file transfer mode, if the effective echo attribute for the operation is different from what it was for the previous *Input* operation, any buffered input is flushed, to ensure that the data returned by the operation will have been echoed correctly. The operation then waits for the first input message to arrive from the terminal. If input messages are already queued (and not flushed), the operation completes immediately.

The line buffer size, in canonical mode, is independent of the size of the input area. It may therefore happen that a longer message is found in a line buffer than will fit in the corresponding input area. In this case the overflowing part is retained in the buffer to be read by the next *Input* operation. On the other hand, an *Input* operation cannot read beyond the end of a line.

An *Input* operation will be answered with status bit 7 set if an interrupt character was received during processing and with status bit 2 set if a timeout occurred.

4. CSP PRINTER PROCESS

The first section of this chapter gives a general description of the way CSP printers work. The remaining sections describe the format and meaning of the messages which are exchanged between an internal process and the CSP printer process. All these messages are operation messages which are sent by an internal process and answered by the CSP printer process.

4.1 General Description

A CSP printer is represented in RC8000 as an external process with kind=14.

Operations can be issued by an internal process that has reserved the CSP printer process. This may be done by initializing the CSP printer process. The printer process cannot be reserved or initialized before it has been created (see below). Messages from an internal process which has not reserved the CSP printer will be rejected (result 2).

The following operations are supported: *Sense* and *Output*.

Printer connection

A link to a CSP printer device handler must be created using the *Create Link* operation (cf. chapter 1). The operation must specify the device name of the CSP printer, and the resulting link can only be used to communicate with the specified printer. Operations will not be accepted until the printer has been connected; see below.

The connection between a printer and the device handler is always established on the initiative of the printer. The printer will then identify itself - by device name - to the device handler. If the device name matches the one specified for an already created link, the connection and the link will be joined, and operations will then be accepted. Otherwise the device handler will reject the connection.

If the link is removed on the initiative of an internal process, the connection to the printer (control unit) is also removed. If the connection is removed because of control unit or network failure, the link remains, but cannot be used.

Status word

The status word is used as follows:

<u>Bit no.</u>	<u>Meaning when set</u>
0	Intervention, i.e. printer faulted or deselected longer than a customized time
5	End of paper
11	No connection

Bit 2 will be set regardless of operation type, and is not discussed under the individual operations in the sections 3.2 and 3.3. It will only occur when there is no printer connected to the link.

4.2 Output Operation

The operation causes the characters in the output area, the text, to be printed. The characters must be stored three per word. Thus the number of characters will always be a multiple of three. Padding with NUL characters must be used if necessary.

Output message:	answer:
+0: 5<12 + conv<4 + statusrep	+0: status word
+2: first storage address	+2: number of halfwords
+4: last storage address	+4: number of characters

The text is sent to the control unit which performs the CSP printer function. The control unit disassembles the text message and outputs each character to the printer.

If conv is 1 the device handler will convert occurrences of CR or LF to CR followed by LF before sending the text to the control unit. If conv is 0 data will be passed unchanged. This is suitable if the printer performs a newline function when it receives one of these characters.

The control unit continually monitors the status of the printer in order to detect failures in the printout process. Such a failure will cause an *Output* operation to be answered with status bit 0 or 5 set.

The value of statusrep determines when the failure in the printout process is reported. If statusrep is 0, the device handler will wait for the control unit to report status at the end of each operation and pass the information in the status word of the answer message.

If statusrep is 2, the answer to the *Output* operation is sent to the internal process as soon as the text has been sent to the control unit, without waiting for a status report. Depending on the number of buffers used by the control unit, this may cause several operations to be buffered: answered, but not printed. When a failure is detected during the printout process it is then reported to the internal process in the status word of the next received *Output* operation.

End of paper status does not cause loss of data, but is merely informative. When an Intervention status is detected, the text transferred in *Output* operations up to the time the status is reported will be lost. If statusrep is set to 2 in all *Output* operations, it is, in general, not possible to know how much data has been lost in this situation.

Regardless of the value of statusrep the halfword and character counts reported in answer messages concern the text that was sent

to the control unit. When an Intervention status occurs, the count of characters actually printed is not reported.

4.3 Sense Operation

The Sense operation is answered immediately even if there are pending *Output* operations. The answer contains the last received status of the printer. This is either identical to the last status returned in the answer to an *Output* operation, or an updated status reported asynchronously from the control unit after a failure.

Sense message:

+0: 0

answer:

+0: status word

Any status that can occur in an answer to an *Output* operation can occur.

A. REFERENCES

- (1) PN 991 11003
RC8000 Attached Device Processors
System Administrator's Guide
- (2) RCSL No.31-D686
RC8000 MONITOR, PART1, System Design
- (3) PN 991 10631
RC8000/ADP3270 Input/Output Process, Reference Manual
- (4) PN 991 10631
RC8000/IMC Port Handler
Reference Manual

B. MESSAGE FORMAT AND RESULT CONVENTIONS

By convention, messages and answers to and from external processes have the following formats:

message:	answer:
+0: opcode<12 + modifier(s)	+0: status word
+2: first storage address	+2: number of halfwords
+4: last storage address	+4: number of characters/bytes
+6:	+6:
.. additional parameters	.. additional results
+14:	+14:

The opcode specifies the operation, possible subspecified by one or more modifiers in subfields of the least significant halfword. Operations which transfer data to or from a device or communication line have odd opcodes, control operations even opcodes.

The first and last storage address point out an RC8000 storage area which is used as source or destination for data transfers. This area is referred to as the input or output area.

The status word is coded in bits or small subfields and used to indicate device status which affected an operation. The exact usage is specified for each external process.

Number of halfwords and number of characters/bytes are only relevant for data transfer operation. Both fields indicate the amount of data which was actually transferred into or out of the input or output area. Halfwords means 12-bit entities, characters/bytes means 8-bit entities. Usually only of the numbers is relevant depending on the type of device, but by convention both are present.

Additional parameters and additional results depend on the external process and on the operation.

The result of an attempt to send an operation message to an external process indicates whether the message was accepted by the external process, and if not, why not. If the message was not accepted, i.e. if the result is not ok, the answer message is dummy.

<u>result</u>	<u>explanation/remarks</u>
1, ok	The operation was accepted, the answer contains status and possibly other information.
2, rejected	The external process was reserved by another process or requires a reservation that has not been made by the sender.
3, unintelligible	Illegal operation code or parameters.
4, malfunction	Device disconnected during operation.
5, no process	There is no the external process with the specified name.

C. CSP TERMINAL PROCESS: MESSAGE/ANSWER SUMMARY

<u>Operation</u>	<u>Message</u>	<u>Answer</u>
Sense	0	0
Set Attributes	132<12 echo<11+mode 0 prompt<8 0 0 input timeout<8	status
Lookup Attributes	134<12	0 echo<11+mode -- prompt<8 -- -- input timeout<8
Output	5<12+transp first address last address	status halfwords characters
Input	3<12+trail<4+blind first address last address	status halfwords characters

D. CSP PRINTER PROCESS: MESSAGE/ANSWER SUMMARY

<u>Operation</u>	<u>Message</u>	<u>Answer</u>
Output	5<12+statusrep first address last address	status halfwords characters
Sense	0	status



