RC8000/RC9000-10

SW8740/1, SW9890I

LAN Device Processes Reference Manual

Keywords:
RC8000, RC9000-10, IFP, ADP, RC8410, RC8411, RC8420, SW8740/1, SW9890, RcLAN, CSP, terminal process, printer process, 3270 input process, 3270 output process

Abstract:
This manual describes fout of the external processes which may be used by RC8000/RC9000-10 internal processes to communicate with LAN devices via an attached device processor (ADP) or LAN Controller: CSP terminal, CSP printer, and the 3270 input and output processes. The link management operations supported by the IFP/ADP (RC8000) or DLC (RC9000-10) main process are also described.

SW Release:
This manual edition is issued as part of SW8740/1 Rel. 6.0 and SW9890I.

Date:
December 1988.

PN: 991 11262

Table Of Contents

1. Introduction

This manual describes four of the external processes which may be used by internal processes to communicate with LAN devices via an attached device processor (ADP/RC8000) or LAN controller (RC9000-10) and the link management operations supported by the ADP or LAN controller.

The manual is intended for programmers who need to deal directly with these external processes. The reader is assumed to be familiar with concepts such as LAN device and device handler. A general description is given in the ADP system administrator's guide/RC9000-10 System Administrator' Guide. Familiarity with RC8000/RC9000-10 Monitor is also assumed.

The LAN devices, with which the system communicates across RcLAN, are:

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

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 or LAN controller. 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 LAN. As a special case an ADP may also serve as control unit. In order for the system 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. Chapter 5 describes the external process for input and output to 3270 clustered devices.

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

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 `makelink`.

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 ADP or LAN controller main process.

Links may also be removed using the utility program `deletelink`. 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 and the external process is also removed.

The main process and its name is preconfigured as part of the Monitor. Systems with multiple ADPs or LAN controllers have a main process for each of these. These processes are named `lanmain1` for the first ADP/LAN controller, `lanmain2` for number 2, etc.

The number of external processes that can be used 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 a LAN or I/O channel device. When a link is created to a device handler on an ADP or a LAN controller, one of these external processes is allocated and its kind changed according to the device type. 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 or LAN controller with respect to types and numbers of device handlers is static. How an ADP or LAN controller is customized with the appropriate device handlers is described in the system administrator's guide. 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.

This chapter also describes the Sense operation, which can be issued to the main process in order to retrieve the host name given to the ADP or LAN controller.

In general, if an operation performed by Monitor as a result of a message to an ADP main process times out, the message will be answered with result malfunction and all links to the ADP will be removed.

2.1 Create Link Operation

The operation creates an external process with a link to a device handler on an ADP or LAN controller. 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 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: device number
 +8:                          +8: device index
 ..  LAN device name
+14:
```

device type:
Type of the device on the ADP or LAN controller, identifies the relevant device handler (some of the device handlers are not described in this manual, cf. the System Administrator's Guide):

1: CSP terminal handler (console)
2: IMC port handler
4: 3270 input handler
5: 3270 output handler
6: lanstat device handler
7: floppy disk handler (RC8000 ADP only)
8: CSP printer handler

device index:
Irrelevant in the message unless device type is 4 or 5 (3270 input or output handler) in which case it specifies the index of the device handler (starting from 0 among device handlers of the same type) to which a link is requested. The value 255 is non-specific, i.e. indicates the first free device handler. These device handlers must be used pairwise. A matching pair of input and output device handlers have equal indices.

Also irrelevant in the answer unless device type is 4 or 5 (3270 input or output handler) in which case it is the index of the device handler to which a link was made. The recommended procedure to set up links to a 3270 device handler pair is: first create a link to a non-specific input handler, then a link to the output handler with the same index.

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 device number in the answer message indicates the chosen device number.

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 the system administrator's guide. 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:

Bit no.	Meaning when set
10-15	Create Link result: 0: ok, link created 3: No Monitor resources, i.e. no unused external process with kind=68 4: No free device handler (or other resource problem on the IFP/ADP) 5: Link already exists

No link is created unless the Create Link result is 0.

Link already exists means:

- If a link to a specific 3270 device handler was requested, the specified device handler is linked already.

- If a link to a CSP device was requested, a link already exists with the specified device name. The device may or may not be connected to the link.

If users (the operation modifier in the first word of the Create Link message) is 0, only the sender proces will become user of the external process. If users is 1, all users of the 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. 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 main process, or if another process has reserved the main process.

2.2 Remove Link Operation

The operation removes an external process with a link to a device handler 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 main process. Otherwise the message will be rejected (result 2).

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

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 or LAN controller associated with the main process to which the message is sent.

2.3 Sense Operation

The Sense operation can be used to check that an ADP or LAN controller is responsive. If not, it will time out. The answer contains the host name which was given to the ADP or LAN controller when it was loaded and which it uses to identify itself on the LAN.

The internal process must be user of the main process. Otherwise the message will be rejected (result 2).

```
Sense message:          answer:
+0: 0<12 + responder     +0: 0
                         +8: host name (bytes 1-3)
                        +10:   -     -  (bytes 4-6)
                        +12    -     -  (bytes 7-8)
```

If the value of responder is 0, the message traverses the full path across the IFP to the ETC in the Multibus subsystem. Thus the responsiveness of the ETC is checked. If the value is 1, the IFP will answer without involving the ETC.

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/RC9000-10 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, which must be of type nologin (CSP customization option used for consoles and communication lines to be controlled by a specific host via a terminal interface). Operations will not be accepted until the terminal has been connected; see below.

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 terminal is of type nologin and the device name matches one specified for an already created link, the connection and the link will be joined, and operations will then be accepted.

In the case of ordinary terminal (CSP terminal which can connect to any host) a new CSP terminal process is created using a free (unused) 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/RC9000-10 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. RC9310, 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, e.g. RC9330,

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 or LAN controller, 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 PADs. 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 (transmitted across the LAN and) passed to an internal process in response to an Input operation. This will happen even if the input area is not filled. In conversational mode this also happens when an input operation times out, and the line buffer is non-empty.

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 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 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:

Bit no.	Meaning when set
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 (cf. the system administrator's guide). 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 last process which communicated with the terminal.

```
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 + immediate<6 +          +0: status word
    trail<4 + blind
+2: first storage address         +2: number of halfwords
+4: last storage address          +4: number of characters
```

Immediate is a single-bit field. If it is set, the operation will be answered immediately with any data that has already been received, including characters that have been typed ahead blindly. When this bit is set, the norml input timeout is disregarded.

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 is 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, regardless of mode, an `Input` operation cannot read beyond the end of a line or line buffer, even if the input area is not filled.

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/RC9000-10 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:

```
Bit no.  Meaning when set

 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 dis cussed 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 print out 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:	**answer:**
+0: 0	+0: status word

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

5. 3270 Input And Output Processes

The first section of this chapter gives a general description of the way the 3270 input and output processes work. Sections 5.2 and 5.3 describe the format and meaning of the operation messages which can be sent from an internal process to the output and input processes, respectively, and the resulting answers. The final section describes the 3270 data stream which is transferred as input or output data in areas pointed out by operation messages.

5.1 General Description

The 3270 input and output processes are external processes with kind = 28.

An internal process becomes user of one of these processes when it it created (cf. chapter 2), or when included as user by its parent.

The 3270 input and output processes must always be used pairwise. A pair consisting of an input process and an output process, which are linked to a corresponding device handler pair on an ADP or LAN controller, lets an internal process communicate with all 3270 devices that are reachable through control units, which are attached to the LAN and perform a 3270 CU function. The control units may either control local devices directly, or they may be front-end processors communicating with remote cluster controllers either via direct lines or as gateways to public or private data networks.

Input data streams from all devices are multiplexed on the link between the input process and the 3270 input handler, and similarly output data streams to all devices are multiplexed on the link to the 3270 output handler. This is suitable for an application design where one internal process communicates with a large number of terminals in terms of transactions, such that the dialog per transaction consists of a single input message from a terminal and a resulting output message from the application. The next input message will typically come from another terminal, and will initiate a new transaction.

Only one internal process can meaningfully use a pair of 3270 input and output processes. If multiple 3270-oriented application processes are

present, they must use separate input/output process and device handler pairs.

Device addressing

Up to 100 3270 CUs may be active on the LAN, each one identified by a CU number in the range 0..99. Each CU, in turn, may provide access to up to 128 devices, each identified by a device number in the range 0..127.

In message format descriptions throughout this chapter, the names `cu` and `dev` refer to CU and device number, respectively. These numbers are always passed in single bytes, binary coded. In input data the high order bit will also be set, but is not part of the number. In output data and in messages, the high order bit may be set, but is disregarded.

The number of CUs that the 3270 device handler pairs on an ADP will communicate with, up to the maximum of 100, is determined by an ADP or LAN controller customization parameter (see the system administrator's guide). If the customized number is *n*, `cu` values 0.*n*-1 are considered legal and greater values illegal, when they occur in operation messages and ouput data. An illegal value will cause the operation to be rejected with result 3, unintelligible.

There are two types of 3270 devices, terminals and printers. Device numbers give no information about device type.

The CU and device numbers occur in the beginning of each 3270 data stream message, transferred by means of `Input` and `Output` operations, allowing the internal process to identify the source of each received message and address output messages to the proper destinations.

One physical control unit attached to the LAN may occupy several CU numbers and thus appear to the internal process as more than one CU. In particular this is true of front-end processors, each of which may provide access to a large number of remote devices. The CU and device numbers used on the LAN between front-end processors and RC8000 internal processes need not be identical to the numbers whereby the actual remote 3270 control units are addressed. The addresses may be mapped in the front-end processor, typically so that a large number of remote CUs, each with few devices, appear on the LAN as a smaller number of CUs, each with a large number of devices, thus making better use of the available address space.

For local control units and the terminals and workstations which perform the device functions, the CU and device numbers are typically specified as part of the customization of the equipment.

Activation and deactivation

A 3270 device handler pair must be activated by means of an `Activate` operation before it begins to communicate with CUs across the LAN. The operation specifies a name whereby the device handler will make itself, i.e. the application provided by the internal process, known to all CUs. Once it has been activated, the device handler will automatically

establish and maintain connections to all the CUs it is customized to communicate with. Attempts to connect are made repeatedly, so that as soon as a CU becomes active on the LAN, a connection will be made. If a connection is lost due to some failure, e.g. if the CU is temporarily out of operation, it will be reestablished as soon as possible.

When a connection has been established between a device handler pair and a CU, the pair, i.e. the application, can be selected from terminals by means of the application name. Typically a CU shows the names of all the applications to which it is connected in a menu.

There is also a `Deactivate` operation which is used to break all connections to CUs and delete the name given to the device handler pair without removing the process pair. The connections are also broken when the 3270 input and output processes are removed. In fact it suffices to remove the output process.

3270 data stream, commands and responses

The data exchange between an application and a 3270 device is message-oriented. The messages sent in the outbound data stream, i.e. from application to device, are called *commands*; and the messages sent in the inbound data stream are called *responses*.

A command can hold a complete display image including field and attribute information. Typically commands are several hundred characters long. Responses in some cases are very short, but typically contain input data from a number of fields on a display form.

The application sends 3270 commands by means of `Output` operations to the 3270 output process. The actual command (outbound data stream) must be passed in the output area pointed out by the operation message. When a command is longer than the output area, it may be split over several successive operations. The end of a command is indicated by an `ETX` character.

Similarly 3270 responses are read by `Input` operations to the 3270 input process, and a response which is too long to fit in the input area must be read by several successive operations, with end of response message indicated by `ETX`.

A read-type command may be sent to a 3270 device causing it to transmit the contents of its device buffer as a response, but normally the read commands are implicitly generated by the cluster controllers, when an attention key is hit on the keyboard of a terminal device. Responses consequently arrive unsolicited.

When data is transferred by a write-type command to a device, it will be stored in the device buffer. If the device is a terminal, it will also be displayed. If the device is a printer, a printout operation will start, provided the print bit is set in the Write or Copy Control Character (`WCC/CCC`) of the command.

Device status reporting

When an application is selected from a terminal, so that the terminal becomes attached to that application, a special short message, `connect attention`, is sent from the device to the application. This message always marks the beginning of the dialog between terminal user and application. An application cannot address a terminal from which it has not received a `connect attention` message; the terminal will be reported unavailable.

Printers do not send similar messages. A printer device must be reserved by the application, before write-type commands will be accepted. The reservation may fail or be queued, if the printer is busy.

Status information from devices is passed in separate messages in the inbound data stream and must be read just like responses, by means of `Input` operations. The data returned in the input area must be inspected to determine if the message is a response or a status message.

The following events cause status messages to be sent:

- A printout operation initiated by a write-type command completes.
- The status of reserved printer changes to unavailable or from offline to ready.
- A previously attached terminal becomes unavailable, possibly because it selects or is transferred to another application.

Display device transfer

As a special feature, the 3270 input and output processes support the transfer of a device from one application to another, on request of the application. This feature can be used among cooperating applications, even on separate hosts on a LAN.

The request takes the form of a special command, passed in the same way as regular 3270 commands, i.e. in the output area pointed out by an `Output` operation.

5.2 Output Process

This section describes the operations which can be sent to the 3270 output process. The operation status word is always 0 in answers to these operations.

5.2.1 General Control Operations

Operations in this group affect the overall state of the 3270 input and output process pair.

Activate
The Activate operation assigns an application name to the process pair whereby it will be known on the LAN and causes the process pair to enter the active state, where connections with all visible CUs are established and maintained. If the pair is already active, all existing connections are broken and new ones established with the new application name.

```
Activate message:              answer:
 +0: 4<12 + 4                   +0: 0
 +4: application name
 +6:        -
 +8:        -
+10:        -
```

If the application name is shorter than 12 characters, it must be terminated with null.

Deactivate
The Deactivate operation causes all connections with CUs to be broken and the current application name to be deleted. The process pair enters the passive state.

```
Deactivate message:            answer:
+0: 4<12 + 8                   +0: 0
```

Sense
The Sense operation returns the application name currently in effect. If the process pair is passive, the name will be all nulls.

```
Sense message:                 answer:
 +0: 0                          +0: 0
                                +4: application name
                                +6:        -
                                +8:        -
                               +10:        -
```

5.2.2 Device Control Operations

Operations in this group are concerned with individual devices. The answer is obtained by communication with the CU in question and contains a CU result field, which stems from the CU.

Lookup device
The Lookup device operation is an enquiry as to the type, status and most recently generated attention of the specified device.

```
Lookup device message:         answer:
 +0: 4<12 + 12                  +0: 0
                                +2: CU result
 +4: cu<8 + dev                 +4: unchanged
                                +6: AID
                                +8: device status byte
```

```
CU result:
0 ok
6 unavailable
7 device number out of range
```

The AID and device status byte contain valid information only when the CU result is ok. See section 5.4 for AID values, and section 5.3, Status byte, for a description of the device status byte (when a status byte is read by a Lookup device operation, the MS bit will be 0).

Reserve device
The Reserve device operation is a request to reserve a printer device. It is answered immediately by the CU.

```
Reserve device message:          answer:
+0: 4<12 + 16                    +0: 0
                                 +2: CU result
+4: cu<8 + dev                   +4: unchanged

CU result:
0 ok
1 not processed
3 no resources
6 unavailable
7 device number out of range
8 device not a printer
10 printer reserved
```

Unless the CU result is ok, the printer was not reserved.

Release device
The Release device operation cancels a printer reservation. If the printer is shared with other applications it shall be sent when the application no longer requires the printer.

```
Release device message:          answer:
+0: 4<12 + 20                    +0: 0
                                 +2: CU result
+4: cu<8 + dev                   +4: unchanged

CU result:
0 ok
7 device number out of range
8 device not a printer
11 device not reserved
```

Wait ready
The Wait ready operation, like Reserve device, is a request to reserve a printer device. If the printer is busy, reserved or unavailable, this request is not answered immediately, but queued.

```
Wait ready message:              answer:
+0: 4<12 + 32                    +0: 0
                                 +2: CU result
+4: cu<8 + dev                   +4: unchanged


CU result:
0 ok
1 not processed
3 no resources
7 device number out of range
8 device not a printer
```

Unless the CU result is ok, the printer was not reserved.

5.2.3 Address Lookup Operations

These operations are almost completely dummy and are retained only for compatibility with similar processes implemented using older front-ends. They do return information as to whether a specified CU is connected or not. The operations are answered by the 3270 output handler without any communication with the CU.

Lookup physical

```
Lookup physical message:         answer:
+0: 2<12 + 14                    +0: 0
                                 +2: lookup result
+4: line<16 + cu<8 + dev         +4: unchanged
                                 +6: equal to word +4

lookup result:
0 The CU is connected, and l=0.
1 line <> 0.
2 The CU is disconnected, and l=0.
```

Lookup logical

```
Lookup logical message:          answer:
+0: 2<12 + 12                    +0: 0
                                 +2: lookup result
                                 +4: equal to word +6
+6: line<16 + cu<8 + dev         +6: unchanged

lookup result:
same as for Lookup physical
```

5.2.4 Output Operation

The `Output operation` is used to transfer a 3270 data stream command or the special `Transfer display device` command to the output process.

```
Output message:                 answer:
+0: 5<12                        +0: 0
+2: first storage address       +2: number of halfword
+4: last storage address        +4: number of character
```

The complete character string which must be passed as output data to effect the transfer of a 3270 command to a device has the form:

cu	dev	ESC	command	ETX

where `cu`, `dev`, `ESC` and `ETX` are single bytes, and the `command` may have any length. The `ETX` character must occur in the last word of the output area, otherwise it is treated as part of the command. A command may be segmented, i.e. transferred by means of several successive `Output` operations. This will be assumed, if there is no `ETX` character in the last word of the output area. In this case the next operation passed to the 3270 output process must be another `Output` operation, containing the continuation of the command.

In general the 3270 output handler checks only that the header (`cu`, `dev`, `ESC`) and the first character of the command (the command character) is legal and then sends the command uninspected to the CU. If anything is illegal in the header, result 3, unintelligible, is given. See subsection 5.4.1 for a description of IBM 3270-compatible commands.

If a command is addressed to a CU which is not connected, result 4, malfunction, is given. However, if the command is segmented, this result is only given for to the last segment.

Transfer display device command

A special case, which is an extension of the IBM 3270-compatible command set, is the `Transfer display device` command, which has the following form:

56	128	application name	connect text
1	1	12	0-43 (no. of bytes)

The addressed device, which must be a terminal, will be detached from the sending application, which will receive a status message, and then become attached to the application identified by the application name specified in the command, as if the terminal operator had selected that application. The connect text is passed to the target application in the `connect attention` message, which is read like a 3270 response (see section 5.3).

A `Transfer display device` command must not be segmented.

5.3 Input Process

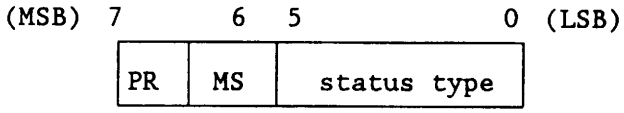
This section describes the operations which can be sent to the 3270 input process. These operations are only used to read input. All control operations must be sent to the output process.

The input process uses a sense ready protocol, i.e. `Input` operations are always answered immediately, regardless of whether any input data is available for transfer to the application. To know when there is input data, the application must send a `Sense ready` operation, which is answered when input data is available, or at the latest after 5 seconds.

When an input message arrives from a CU, the pending `Sense ready` operation is answered, if there is one, and the message is queued by the 3270 input handler until an `Input` operation is received from the application. The input queue is controlled by a timer, which can be customized. If the timer expires, indicating that the application has failed to read its input for an unexpectedly long time, the queued messages are discarded, and on those terminals where it is possible a message is displayed in the status line.

Status byte

The device status byte, which occurs as input data in status messages and also occurs in the answer to the `Lookup device` operation, is coded as shown below:



When the PR bit is 1, the device is a printer. When the MS (more status) bit is 1, a more recent status may be obtained by means of a `Lookup device` operation.

The following values of status type are possible:

0 device end, normal completion of printout operation, or printer ready after being offline;

1 not ready, printout operation completed abnormally because of time out on printer busy signal;

3 offline, printout operation completed abnormally because the printer went offline;

4 unavailable, the device became unavailable;

6 transferred, terminal became unavailable by executing a `Transfer display device` command.

10 Unexpected error result received from CU. If this status occurs, it may represent an error in the device handler and should be reported as an error.

Sense ready operation

```
Sense ready message:          answer:
+0: 0<12 + 2                  +0: status
                              +2: 0
                              +4: 0

status: 0       input ready
        bit 2   timeout
```

Input operation

The Input operation is used to transfer a 3270 data stream response, connect attention or status message to the input area of the application process.

```
Input message:                answer:
+0: 3<12                      +0: 0
+2: first storage address     +2: number of halfwords
+4: last storage address      +4: number of characters
```

The complete character string which is be passed as input data has the form:

cu	dev	AID	response/status byte/connect/text	ETX

where cu, dev, AID and ETX are single bytes, and the field between AID and ETX may have any length, including 0. When an input message is longer than the input area, the overflowing part of the message will be read by the next Input operation. ETX is always written as the last byte in an input area, in the position indicated by the character count. When a message overflows, ETX is not written in the input area.

AID stands for Attention IDentifier. In 3270 responses, the value identifies the attention key which was hit. See subsection 5.4.2 for a description of IBM3270-compatible responses. Status messages and connect attention messages are identified by special AID values.

The AID value identifying a status message is 156. It is followed by a device status byte.

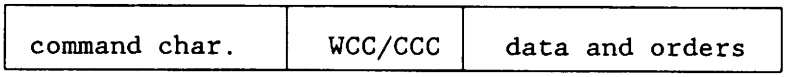
The AID value for a connect attention message is 157. It may be followed by up to 43 characters of connect text, cf. subsection 5.2.4, Transfer display device command. In the normal case of manual selection, the connect text field is empty.

5.4 IBM 3270-Compatible Data Stream

The 3270 commands which are transmitted from the 3270 output handler to CUs and the 3270 responses which the 3270 input handler receives from CUs are compatible with the IBM 3270 Information Display System. Detailed information on the meaning of the commands and responses with respect to terminal and printer functions may be found in IBM documentation.

5.4.1 3270 Commands

The general format of a command is as shown below:



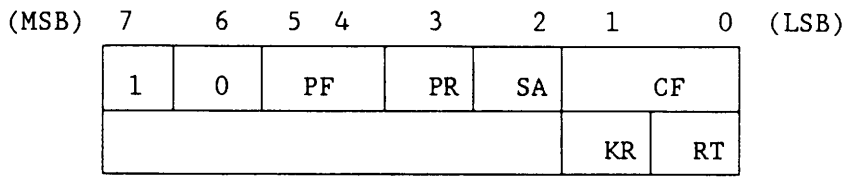
The command character identifies the command in question. The commands fall in three groups:

- read-type (R) commands, which explicitly request that a response be generated and sent in the inbound data stream,
- write-type (W) commands, which specify updating operations to be performed on the device buffer and possibly initiate other operations as well,
- control-type (C) commands, which initiate operations that do not require the transfer of data between the host/application and the device buffer.

The commands which are allowed by the 3270 output process are listed below:

Command name	Type	Graphic	Dec.	Hex
Write	W	1	49	31
Read Buffer	R	2	50	32
Erase/Write	W	5	53	35
Read Modified	R	6	54	36
Copy	C	7	55	37
Erase/Write Alternate	W	=	61	3D
Read Modified All	R	>	62	3E
Erase All Unprotected	C	?	63	3F

The Write/Copy Control Character (WCC/CCC) is coded as follows:



CF copy format (only in CCC):
- 00: only atributes
- 01: attributes and unprotected fields
- 10: attributes and protected fields
- 11: entire device buffer

RT reset MDT bits (only in WCC)

KR keyboard restore (only in WCC)

SA sound alarm

PR print (initiate printout operation)

PF printout format:
- 00: transparent
- 01: 40 characters/line
- 10: 64 characters/line
- 11: 80 charcters/line

The data which occurs in write-type commands following the WCC consists of displayable characters and printer control characters interspersed with device buffer orders. The following printer control characters are defined:

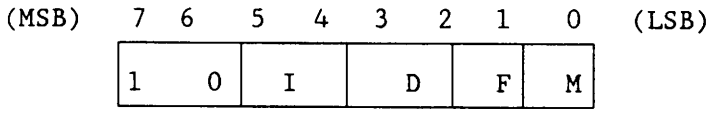
Abbreviation	Control character	Decimal code
NL	Newline	10
VT	Vertical Tab	11
FF	Form Feed	12
CR	Carriage Return	13
EM	End Medium	25

Device buffer orders occupy from 1 to 4 bytes as shown in the table below. ATR means attribute, CHAR means any displayable character, and ADDR1, ADDR2 is a buffer address coded as two 6 bit quantities (high order bits followed by low order bits, cf. also subsection 5.4.3).

Abb.	Order name	Byte 1	Byte 2	Byte 3	Byte 4
SF	Start Field	29	ATR		
SBA	Set Buffer Address	17	ADDR1	ADDR2	
IC	Insert Cursor	19			
RA	Repeat to Address	20	ADDR1	ADDR2	CHAR
PT	Program Tabulation	9			
EUA	Erase Unprotected to Address	18	ADDR1	ADDR2	
USM	UnSolicited Message	31			

Attribute character

The attribute character (ATR) is coded as follows:




```
M       modified field (MDT)

F       flashing

D       display
              00: normal display
              01: cursor selectable
              10: intensified and cursor selectable
              11: nondisplay

I       input:
              00: alphanumeric
              01: numeric
              10: protected
              11: automatic skip
```

5.4.2 3270 Responses

A short response message, which is sent when a PA key, CLEAR or USM is hit, consists of the AID character only. Otherwise the AID character is followed first by the cursor address coded as two 6 bit quantities (high order bits followed by low order bits, cf. also subsection 5.4.3) and then data and orders.

If the response is to a Read Modified (explicit or implicit) or Read Modified All command, the data and orders part will contain for each modified field an SBA order (cf. subsection 5.4.1) followed by the field contents with nulls suppressed. The field contents are left out if the AID is CURSEL and the command is Read Modified. If the display is unformatted there are no SBA orders.

If the response is to a Read Buffer command the data and orders part will contain the entire device buffer contents including nulls with each attribute character represented as an SF order (cf. subsection 5.4.1).

AID codes

The following codes are used for **AID** (Attention IDentifiers) characters:

Name	Decimal code	Description
NO_AT	160	No AID pending (display)
NO_AP	161	- (printer)
SEND	129	Send/Enter (Cursor Select)
PF1	130	Program Function 1
PF2	131	- 2
PF3	132	- 3
PF4	133	- 4
PF5	134	- 5
PF6	135	- 6
PF7	136	- 7
PF8	137	- 8
PF9	138	- 9
PF10	139	- 10
PF11	140	- 11
PF12	141	- 12
PF13	147	- 13
PF14	148	- 14
PF15	173	- 15
PF16	174	- 16
PF17	175	- 17
PF18	176	- 18
PF19	177	- 19
PF20	178	- 20
PF21	179	- 21
PF22	180	- 22
PF23	181	- 23
PF24	182	- 24
CLEAR	145	Clear Screen
USM	152	UnSolicited Message
PA1	142	Program Access 1
PA2	143	- 2
PA3	144	- 3
PA4	150	- 4
PA5	151	- 5
PA6	162	- 6
PA7	163	- 7
PA8	164	- 8
PA9	165	- 9
PA10	166	- 10
MSR	170	Magnetic Slot Reader, 3277/Datasaab style
MSR3278	171	- 3278 style
CURSEL	149	CURsor SELect

5.4.3 Character Codes

This section summarizes the character codes which are used in the data stream. Notice that the character codes are not necessarily unique, i.e. the meaning of a particular code in some cases depends on the context.

ISO 7-bit codes

All displayable characters are coded according to the appropriate national character code derived from ISO 646. Control characters which fall within the set defined in ISO 646 also have ISO codes, viz. in the range 0..31. When a character (8 bits) shall be interpreted to 7-bit ISO code, the high order bit is always 0.

The following control characters are used:

Abb.	Dec	Type of use
NULL	0	Device buffer data (empty position)
PT	9	Device buffer order
NL	10	Printer control character
VT	11	Printer control character
FF	12	Printer control character
CR	13	Printer control character
SBA (DC1)	17	Device buffer order
EUA (DC2)	18	Device buffer order
IC (DC3)	19	Device buffer order
RA (DC4)	20	Device buffer order
EM	25	Printer control character
DUP	28	Device buffer data (DUP character)
SF (GS) FM	29	Device buffer order
USM	30	Device buffer data (FM character)
	31	Device buffer order

Note that the command character specifying the type of command present in the outbound data stream is coded as a displayable character.

Extensions to 7-bit code

Codes for purposes not satisfied by the repertoire of ISO codes are always greater than 128. There are two kinds of extensions to the ISO code set:

- AID codes, cf. Appendix D
- binary or subfield encoded 6-bit quantities with high order bits equal to 10, viz.:
  - attribute (ATR) characters
  - Write/Copy Control Characters (WCC/CCC)
  - device buffer addresses (ADDR1, ADDR2)

A. Message Format And Result Conventions

By convention, messages and answers to and from external proces ses 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/byte
 +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/RC9000-10 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 operations. 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. Number of halfwords is always even. Usually only one 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.

Character data is always passed 3 characters per word using big-endian byte ordering, in messages and answers as well as in input and output areas. Unused character positions contain nulls.

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.

result	**explanation/remarks**
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.
5, no process	There is no the external process with the specified name.

B. CSP Terminal Process: Message/Answer Summary

Operation	Message	Answer
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+immediate<6+ trail<4+blind first address last address	status halfwords characters

Status word:	Bit no.	Meaning when set
	2	Input timeout
	7	Attention
	11	Terminal not connected

C. CSP Printer Process: Message/Answer Summary

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

Status word:	Bit no.	Meaning when set
	0	Intervention
	5	End of paper
	11	No connection

D. 3270 Input Process: Message/Answer Summary

Operation	Message	Answer
Sense ready	0<12+2	status 0 0
Input	3<12 first address last address	0 halfwords characters

Status word:	Bit no.	Meaning when set
	2	Input timeout

E. 3270 Output Process: Message/Answer Summary

Operation	Message	Answer
Sense	0	0 -- application name "" "" ""
Activate	4<12+4 0 application name "" "" ""	0
Deactivate	4<12+8	0
Lookup device	4<12+12 -- cu<8+dev	0 CU result -- AID device status
Reserve device	4<12+16 -- cu<8+dev	0 CU result
Release device	4<12+20 -- cu<8+dev	0 CU result

Wait ready	4<12+32 -- cu<8+dev	0 CU result
Lookup physical	2<12+14 -- line<16+cu<8+dev	0 lookup result -- line<16+cu<8+dev
Lookup logical	2<12+12 -- line<16+cu<8+dev	0 lookup result line<16+cu<8+dev --
Output	5<12 first address last address	0 halfwords characters