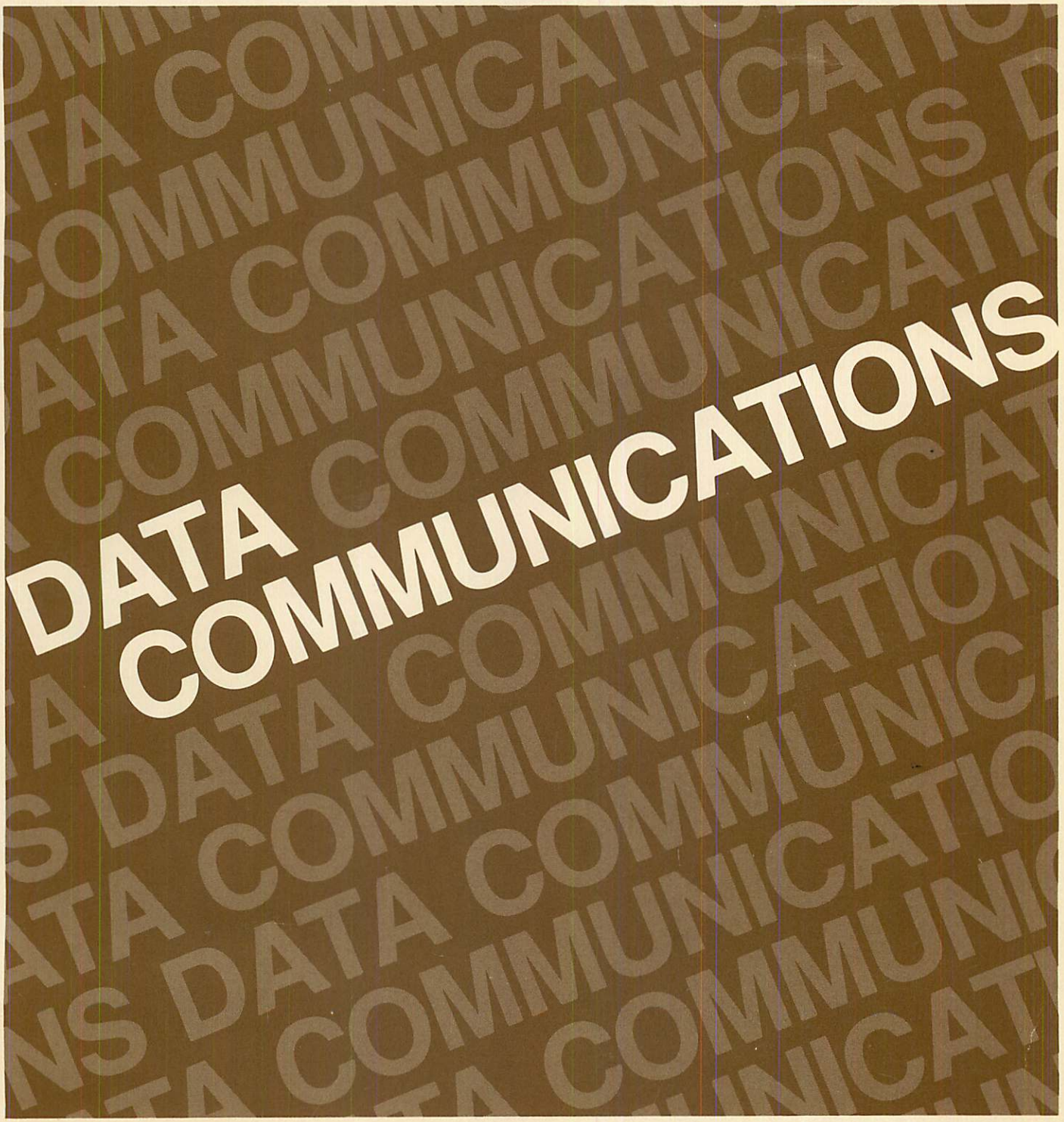
# HP 3000 Computer Systems

HEWLETT hp PACKARD

2780/3780 Emulator
reference manual



# HP 3000 Computer Systems

# 2780/3780 Emulator

## Reference Manual

HEWLETT PACKARD

5303 STEVENS CREEK BLVD., SANTA CLARA, CALIFORNIA, 95050

Part No. 30000-90047
Product No. 30130C
Software Version No. C.00.03

Printed in U.S.A. 6/77

## NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

# LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the date of the current edition and of any pages changed in updates to that edition.

Second Edition ..... Jun 1977

# PRINTING HISTORY

New editions incorporate all update material since the previous edition. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The date on the title page and back cover changes only when a new edition is published. If minor corrections and updates are incorporated, the manual is reprinted but neither the date on the title page and back cover nor the edition change.

First Edition. . . . . Jun 1976
Update Package No. 1 . . . . . Jan 1977
Second Edition . . . . . Jun 1977

The second edition documents two new parameters for the RJLINE command (LOCK= and PRI=). It also includes a new example in Section VII (Example 7). Additionally, some areas of the original manual were rewritten for clarity.

# PREFACE

This manual describes the Hewlett-Packard 2780/3780 Emulator which is designed to run under the control of the HP 3000 Multiprogramming Executive (MPE/3000) Operating System. The Emulator makes it possible for you to exchange data between your HP 3000 Computer System and a variety of remote processors over the public telephone (switched) network or private leased lines.

This manual assumes that you have a working knowledge of MPE/3000 and that you have some familiarity with either the IBM 2780 Data Transmission Terminal or the IBM 3780 Data Communication Terminal. Knowledge of the IBM Binary Synchronous Communications (BSC or Bisynch) line protocol conventions would be somewhat helpful but is not required.

Throughout this manual the term "data set" is consistently used to mean a set of data (a group of related input or output records) and the term "MODEM" is consistently used to mean the piece of equipment that converts digital signals to tones (suitable for transmission over telephone lines) and tones to digital signals.

This manual is arranged in five sections with ten appendices:

- Section I summarizes the IBM 2780 and 3780 terminals and the Emulator and its environment.
- Section II tells you how to initiate an Emulator run.
- Section III describes the ten Emulator commands.
- Section IV presents six annotated examples illustrating how you can use the Emulator in various situations.
- Section V describes the informational and error messages that can occur during an Emulator run.
- Appendix A presents a comparison of the IBM 2780 and 3780 terminals.
- Appendix B summarizes the options available for some typical MODEMs and indicates those options that are recommended (or required) for use with the Emulator.
- Appendix C contains a tabulation of the ASCII, EBCDIC, and Hollerith character sets.
- Appendix D summarizes the ten Emulator commands.
- Appendix E tells you how to configure the Emulator.
- Appendix F summarizes the error messages.
- Appendix G shows an example of a procedure used with output commands.
- Appendix H describes the CS/3000 Trace Facility that provides a record of line activities during Emulator operation.
- Appendix I summarizes Binary Synchronous Communications line protocol control characters.
- Appendix J explains the characteristics of Emulator files and how and when they are closed.

This manual references the following manuals:

- *System Manager/System Supervisor Reference Manual*, 30000-90014.
- *HP 30055A Synchronous Single-Line Controller*, 30055-90001.

# CONTENTS

# CONTENTS (continued)



# ILLUSTRATIONS



# TABLES

# INTRODUCING THE 2780/3780 EMULATOR

SECTION I

The Hewlett-Packard 2780/3780 Emulator lets you transfer data between your HP 3000 Computer System and a variety of remote processors in a full multiprogramming environment over the public telephone (switched) network or a private leased line. You can transfer data at rates of up to 9600 bits-per-second, depending upon your choice of MODEM. Higher transmission rates can be supported for certain applications. However, such determination must be made on an individual basis.

The Emulator, which runs under the control of the HP 3000 Multiprogramming Executive Operating System (MPE/3000), makes your HP 3000 Computer System appear to the remote processor as either an IBM 2780 or 3780 Data Transmission Terminal. The Emulator is more flexible than the IBM terminals in that it allows you to use a greater variety of input/output devices, including disc and magnetic tape.

The remote processor can be any of the following:

- Any computer which supports the IBM 2780 or 3780 Data Transmission Terminals.
- An IBM 2780 or 3780.
- Another HP 3000 Computer System which is also using an HP 2780/3780 Emulator.

You invoke the Emulator with the MPE :RJE command as described in section II. You can operate the Emulator in either the job (batch) or session (interactive) mode. Ten Emulator commands allow you to control the sequence of input/output processing. These commands are summarized in Table 1-1.

Provided that your HP 3000 Computer System has more than one Synchronous Single-Line Controller, several people may use the Emulator concurrently. The number of concurrent users is limited by the number of Synchronous Single-Line Controllers which are available. Prior to invoking the Emulator, you can specify which Synchronous Single-Line Controller you wish to use through the use of an MPE file equation (this is described in Section II of this manual).

## 1-1. IBM 2780 DATA TRANSMISSION TERMINAL

The IBM 2780 Data Transmission Terminal was IBM's original low-cost remote batch processing terminal and was the first data processor to use IBM Binary Synchronous Communications telecommunications line protocol (commonly referred to as BSC or bisynch). Physically, the IBM 2780 consists of the following hardware elements:

- A control unit, a bisynch adapter, and buffers.
- A card reader capable of reading up to 400 cards-per-minute.
- A card punch (optional) capable of punching up to 355 cards-per-minute.
- A line printer (optional) capable of printing up to 300 lines-per-minute.

Functionally, an IBM 2780 is capable of the following:

- Transmitting or receiving in ASCII, EBCDIC, or six-bit transcode.
- Accepting horizontal tabulation codes, thereby making it possible to process line printer output more efficiently (this capability is optional on the IBM 2780).

Table 1-1. HP 2780/3780 Emulator Commands

| COMMAND | FUNCTION |
|---|---|
| #RJLINE | Defines the characteristics of the communications line. |
| #RJIN | Transmits input data from your HP 3000 Computer System to the remote processor. |
| #RJOUT | Requests and processes routed output from the remote processor. |
| #RJIO | Initiates transmission of a one-line message to the remote processor. |
| #RJINFO | Initiates a file display printing of the communications line. |
| #RJDEBUG | Sets the Emulator into the debug mode. |
| #RJLIST | Requests and processes unrouted list output from the remote processor. |
| #RJPUNCH | Requests and processes unrouted punched output from the remote processor. |
| #RJEOD | Transmits an EOT control character (end-of-file) to the remote processor. |
| #RJEND | Terminates the HP 2780/3780 Emulator subsystem. |

Note: The terms "routed" and "unrouted" refer to the presence or absence of component select codes in conjunction with output transmitted from the remote processor. This concept is discussed in detail under the descriptions of the #RJOUT, #RJLIST, and #RJPUNCH commands in Section III, of this manual.

- Accepting vertical forms control codes, thereby facilitating the formatting of line printer output.
- Transmitting or receiving data in the transparent mode, thereby making it possible to transmit or receive binary data (this capability is optional on the IBM 2780 and applies only to the EBCDIC mode). Transparent mode is the transmission of binary data with no recognition of most control characters.
- Performing short-record truncation. This makes it possible to suppress trailing blanks from card input, thereby increasing the transmission efficiency. However, the user must indicate the start of each trailing blank field by including an EM control character in the data.
- Transmitting or receiving blocked data, thereby increasing the transmission efficiency. The maximum number of records allowed per block is 2 (as an option, this maximum can be raised to 7).

For a comparison of the IBM 2780 and 3780, see Appendix A of this manual. For a detailed description of the IBM 2780, refer to **Component Description: IBM 2780 Data Transmission Terminal** (IBM SRL GA27-3005).

## 1-2. IBM 3780 DATA COMMUNICATION TERMINAL

Like the IBM 2780, the IBM 3780 Data Communication Terminal is a control unit with a set of input/output devices. Physically, the IBM 3780 consists of the following hardware elements:

- A control unit, a bisynch adapter, and buffers.
- A card reader capable of reading up to 600 cards-per-minute.
- A card punch (optional) capable of punching up to 359 cards-per-minute.
- A line printer capable of printing up to 425 lines-per-minute.

Functionally, an IBM 3780 is capable of the following:

- Transmitting or receiving in ASCII or EBCDIC.
- Accepting horizontal tabulation codes, thereby making it possible to process line printer output more efficiently.
- Accepting vertical forms control codes, thereby facilitating the formatting of line printer output.
- Transmitting or receiving data in the transparent mode, thereby making it possible to transmit or receive binary data (this capability is optional on the IBM 3780 and applies only to the EBCDIC mode).
- Performing short-record truncation. This makes it possible to suppress trailing blanks from card input, thereby increasing the transmission efficiency. This function is performed automatically; the user does **not** need to include EM or IRS control characters in the data.
- Compressing blank fields within the data, thereby increasing the transmission efficiency. The IBM 3780 does this automatically by replacing blank fields with a two-byte control sequence.
- Transmitting or receiving blocked data, thereby increasing the transmission efficiency. The maximum number of records allowed per block is 216 in the non-transparent mode and 1 (optionally 6) in the transparent mode.
- Interleaving the data passed between the communications line and the input/output devices through the use of two 512-byte buffers, thereby increasing the overall transmission efficiency.
- Generating WACK (Wait-before-transmit ACKnowledgement) and TTD (Temporary Text Delay) control character sequences when it is temporarily unable to transmit or receive data.

For a comparison of the IBM 2780 and 3780, see Appendix A of this manual. For a detailed description of the IBM 3780, refer to **Component Information for the IBM 3780 Data Communication Terminal** (IBM SRL GA27-3063).

## 1-3. HP 2780/3780 EMULATOR CAPABILITIES

The HP 2780/3780 Emulator provides the capabilities stated in paragraphs 1-1 and 1-2 plus the following:

- When emulating an IBM 2780, it performs short-record truncation **without** the user having to supply EM control characters in the data.
- When emulating an IBM 2780, it can perform blank field compression.
- When emulating either an IBM 2780 or 3780, it provides input/output access to a greater variety of peripheral devices including disc and magnetic tape.
- When emulating an IBM 2780, it can block more than 7 records.

## 1-4. SYSTEM ENVIRONMENT

The HP 2780/3780 Emulator is designed to run under the control of the HP 3000 Multiprogramming Executive (MPE/3000) Operating System. The Emulator requires the minimum HP 3000 Computer System configuration plus one of the following MODEMs or its equivalent:

- A Bell System Type 201A3 Data Set for 2000 bit-per-second (bps) transmission via the public telephone (switched) network.
- A Bell System Type 201B3 Data Set for 2400 bps transmission via a private leased line.
- A Bell System Type 201C Data Set for 2400 bps transmission via either the public telephone (switched) network or a private leased line.
- A Bell System Type 208A Data Set for 4800 bps transmission via a private leased line.
- A Bell System Type 208B Data Set for 4800 bps transmission via the public telephone (switched) network.
- A Bell System Type 209A Data Set for 9600 bps transmission via a private leased line.

The MODEM options recommended for use with the HP 2780/3780 Emulator are presented in Appendix B of this manual.

# OPERATING THE EMULATOR

SECTION II

To operate the HP 2780/3780 Emulator, you must:

1. Define any formal file designators which you will use during the job or session.
2. Invoke the Emulator.
3. Define the communications line.

## 2-1. DEFINING YOUR FORMAL FILE DESIGNATORS

If you will be using the form **formal file designator* in the :RJE command (described in paragraph 2-2) or in any of the Emulator commands, you must equate each formal file designator to an actual file designator, a logical device number, or a device class name by executing MPE :FILE commands (one for each formal file designator). Refer to the description of the :FILE command in the MPE Commands reference manual.

## 2-2. INVOKING THE EMULATOR

After you have defined the formal file designators, invoke the Emulator by issuing an MPE :RJE command. The format of the :RJE command is as follows:

:RJE [*command file*]

$$\left[,[\textit{input file}]\left[,[\textit{list file}]\left[,[\textit{punch file}]\right]\right]\right]$$

The parameters are defined in table 2-1. You must supply the parameters in the order shown above. If you wish to omit any of the first three parameters, you must include the associated comma(s). If you wish to omit all of the parameters, you can omit the commas altogether. If you wish to omit the last one, two, or three parameters, you can omit the associated comma(s). Some valid format examples are as follows:

:RJE
:RJE *command file*
:RJE *,input file*
:RJE *,,list file*
:RJE *,,,punch file*
:RJE *command file,,,punch file*
:RJE *,input file, list file*

You can, either explicitly or by default, designate a single device or file as being both the command and input files. In such a case, the Emulator will switch back and forth between the command and input modes when reading from the particular device or file. Initially the Emulator is in the command mode. In the *command mode*, all data read from the command/input file is interpreted as Emulator commands. When the Emulator encounters an #RJIN command, it switches to the input mode. In the input mode, all data read from the command/input file is transmitted to the remote processor. When the Emulator encounters another command, it reverts back to the command mode, and so forth.

As you will learn from Section III of this manual, you can also specify the input, list, and punch files in some of the Emulator commands. If you do, the file specification in the Emulator command overrides the corresponding one in the :RJE command. In other words, the input, list, and punch file specifications in the :RJE command apply only when the corresponding specification is omitted from an Emulator command. For example, if you omit the *output file reference* from an #RJOUT command, the list and punch files which you specified in the :RJE command will be used.

## 2-3. DEFINING THE COMMUNICATIONS LINE

The Emulator assumes that the communications line has the device class name RJLINE. If your HP 3000 Computer System does include the device class name RJLINE, then the communications line is already defined and you are not required to perform any further actions. If it does not include RJLINE, then you must define the communications line by specifying the DEV parameter in the RJLINE command. This is performed when the RJLINE command parameters are used to define the communications link between the Emulator and remote processor, as described in Section III. The user and account must have Communication System (CS) capabilities to use the RJLINE command.

Table 2-1. :RJE Command Parameter Definitions

| FILE OR DEVICE | FUNCTION | ALLOWABLE VALUES | DEFAULT VALUE | FORMAL FILE DESIGNATOR USED BY EMULATOR |
|---|---|---|---|---|
| command file | File or device from which the Emulator reads HP 2780/3780 Emulator directives. The directives are described in Section III of this manual. | actual file designator<br>*formal file designator<br>$STDIN<br>$STDINX | $STDINX | RJECOM |
| input file | File or device from which the Emulator reads input data to be transmitted to the remote computer. | actual file designator<br>*formal file designator<br>$STDIN<br>$STDINX | $STDINX | RJEIN |
| list file | File or device to which list output received from the remote computer is to be routed. | actual file designator<br>*formal file designator<br>$STDLIST<br>$OLDPASS<br>$NEWPASS | $STDLIST | RJELIST |
| punch file | File or device to which punched output received from the remote computer is to be routed. | actual file designator<br>*formal file designator<br>$OLDPASS<br>$NEWPASS | $OLDPASS (if it exists; see Note) or $NEWPASS | RJEPUNCH |

Note: Emulator punch files are generally expected to have a file code of 1060 (decimal). If $OLDPASS exists, but has a file code other than 1060, then $NEWPASS is opened with a file code of 1060 and used. If you specify a punch file in the :RJE or #RJPUNCH commands, it may have any file code if it is an OLD file; if it is not an OLD file, the Emulator will open a NEW file with the specified name and with a file code of 1060.

Note: See Appendix J for input file characteristics.

## 2-4. :RJE COMMAND EXAMPLES

All of the examples below show you how to invoke the HP 2780/3780 Emulator. They differ from one another only in how the various files are defined.

Remember, you can redefine the input, list, or punch file for a specific input/output operation by supplying a file reference parameter in the particular Emulator command.

### 2-5. USING DEFAULT VALUES

To define all files by default, use the following :RJE command:

```
:RJE
```

The files are defined as follows:

- Command file: $STDINX
- Input file: $STDINX
- List file: $STDLIST
- Punch file: $OLDPASS (if it exists) or $NEWPASS (if $OLDPASS does not exist)

### 2-6. USING DEVICE CLASS NAMES

To define one or more of the files as device classes you must use MPE :FILE commands in addition to the :RJE command.

Assume that LP and CP were defined at system generation time as the device class names for all line printers and card punches (respectively) in the system. To use LP and CP as file specifications in an :RJE command, you must first equate them to formal file designators as follows:

```
:FILE LIST; DEV=LP
:FILE PNCH; DEV=CP
```

Note: You can use any formal file designators you wish. LIST and PNCH are used only for the purposes of this example.

To define the command and input files by default and the list and punch files as the device classes LP and CP (respectively), use the following :RJE command:

```
:RJE ,,*LIST,*PNCH
```

The files are defined as follows:

- Command file: $STDINX
- Input file: $STDINX
- List file: The first available line printer in the system.
- Punch file: The first available card punch in the system

## 2-7. USING ACTUAL FILE DESIGNATORS

Assume that ABC is the actual file designator for an existing file on disc. To define the command and input files as a single file (the file ABC) and the list and punch files by default, use the following :RJE command:

```
:RJE ABC,ABC
```

The files are defined as follows:

- Command file: ABC
- Input file: ABC
- List file: $STDLIST
- Punch file: $OLDPASS (if it exists) or $NEWPASS (if $OLDPASS does not exist)

The file ABC contains the Emulator commands and the input data sets. Each #RJIN command must be immediately followed by the associated input data set.

## 2-8. USING LOGICAL DEVICE NUMBERS

To define one or more of the files as specific devices, you must use MPE :FILE commands in addition to the :RJE command.

Assume that, at system generation time, 21 was defined as the logical device number of an interactive terminal, 5 was defined as the logical device number of a card reader, and LP and CP were defined as the device class names for all line printers and card punches (respectively) in the system. To use 21, 5, LP, and CP as file specifications in an :RJE command, you must first equate them to formal file designators as follows:

```
:FILE CMND; DEV=21
:FILE INPT;  DEV=5
:FILE LIST;  DEV=LP
:FILE PNCH; DEV=CP
```

Note: You can use any formal file designators you wish. CMND, INPT, LIST, and PNCH are used only for the purposes of this example.

To define the command and input files as logical devices #21 and #5 (respectively) and the list and punch files as device classes LP and CP (respectively), use the following :RJE command:

```
:RJE *CMND,*INPT,*LIST,*PNCH
```

The files are defined as follows:

- Command file: Logical device #21
- Input file: Logical device #5
- List file: The first available line printer in the system.
- Punch file: The first available card punch in the system.

# WRITING HP 2780/3780 EMULATOR COMMANDS

SECTION III

The 10 HP 2780/3780 Emulator commands are summarized in Table 3-1 and are described in detail in this section. Emulator commands are used for two purposes:

1. To define the necessary parameters for input and output operations.
2. To control the sequence of HP 2780/3780 Emulator input and output activities.

An Emulator command is signified by a # character in the first position of the command record (batch job) or is prompted by a # character if the command file is your terminal (interactive session). In the latter case, when entering an Emulator command do **not** enter another # character; the prompt character serves as the first character of the Emulator command.

This section was designed with the hope that it could be used for both learning and reference. With this in mind, the following general format was chosen for presenting each Emulator command:

| | |
|---|---|
| FUNCTION | — describes briefly what the command can do. |
| FORMAT | — shows the format of the command and briefly defines the variables that you can supply. |
| DEFAULT VALUES | — describes the default values for each parameter. |
| COMMENTS | — describes in detail the operation of the command and the implications of the various parameters. |
| EXAMPLES | — presents annotated examples of the command. |

The first three topics are intended primarily as reference material. If, while perusing them, you are confused about some aspect of the command, refer to the "COMMENTS" and "EXAMPLES" paragraphs.

Table 3-1. HP 2780/3780 Emulator Commands

| COMMAND MNEMONIC | MNEMONIC DEFINITION | FUNCTION |
|---|---|---|
| #RJLINE | Remote Job LINE | Defines the communications link between the Emulator and the remote processor. |
| #RJIN | Remote Job INput | Defines the necessary parameters for a particular input operation and initiates the transmission of one input data set to the remote processor. |
| #RJOUT | Remote Job OUTput | Defines the necessary parameters for a particular output operation (list and/or punch) and initiates the receipt of routed output from the remote processor. |
| #RJIO | Remote Job Input Output | Initiates transmission of a one-line message to the remote processor and initiates receive mode for the receipt of routed data from the remote processor. |
| #RJINFO | Remote Job Line INFOrmation | Initiates a file display printing of the communications line. |
| #RJDEBUG | Remote Job DEBUG | Sets the emulator into the debug mode, allowing the user access to the debugging facilities. |
| #RJLIST | Remote Job LIST | Defines the necessary parameters for a particular list output operation and initiates the receipt of unrouted list output from the remote processor. |
| #RJPUNCH | Remote Job PUNCH | Defines the necessary parameters for a particular punched output operation and initiates the receipt of unrouted punched output from the remote processor. |
| #RJEOD | Remote Job End of Data | Transmits an EOT (End Of Transmission) control character to the remote processor. |
| #RJEND | Remote Job END | Terminates the HP 2780/3780 Emulator subsystem. |

## 3-1. #RJLINE

## 3-2. FUNCTION

The #RJLINE command defines the communications link between the Emulator and the remote processor.

## 3-3. FORMAT

#RJLINE *emulator type*

[;LINECODE= {ASCII / EBCDIC}]

[;CONNECT= {DIAL [,"*phone number*"] / ANSWER}]

[;MAXRPB=*blocking factor*]

[;ID=*terminal identification*]

[;RIN=*rin* [,*password*]]

[;CHNL3=*x*]

[;XEND]

[;TRACE=[[ALL][,[*mask*][,[*numentries*][,WRAP]]]]]

[;DEV = *n*] [;LOCK = {YES / NO}] [;PRI = {HIGH / NORMAL}]

#RJLINE *emulator type*
*emulator type* specifies what type of data transmission terminal is to be emulated. It may be either of the following:

2780
3780

Note: Whenever you have the choice of emulating either an IBM 2780 or 3780, you should be aware that the Emulator operates more efficiently in the 3780 mode.

;CONNECT = DIAL, *phone number*
*phone number* is an ASCII character string consisting of numeric characters (0 through 9) and dash characters (-) enclosed within quotes not greater than 20 characters long. The phone number entered is included as part of the dial message to notify the computer operator to dial the remote processor. In response to the dial message, the operator dials the remote processor and then enters YES or NO through the operator's console, indicating whether or not the remote processor answered. The entered phone number may include an area code. If so, the area code is followed by a dash and then by the telephone number.

If phone number is omitted from the :CONNECT= DIAL parameter, the default phone number entered during Emulator configuration (see Appendix E) is used. Entering a phone number automatically overrides the default phone number.

Note: Phone numbers, identification, and forms messages do not require quotes or delimiters, and the quotes may be omitted.

;MAXRPB=*blocking factor*
*blocking factor* is a decimal integer specifying the maximum number of records per transmission block.

;ID=*terminal identification*
*terminal identification* is an ASCII character string to be used as your terminal ID when the communications link is being established. This parameter should always be an ASCII character string. If the line code is specified as being EBCDIC(LINECODE=EBCDIC), the Emulator automatically converts the terminal identification from ASCII to EBCDIC.

Note: This parameter should be used only when the remote processor requires a terminal ID. The Emulator transmits the terminal ID only if the Emulator initiates transmission (i.e., only if the first operation is #RJIN rather than #RJOUT, #RJLIST, or #RJPUNCH). The Emulator always ignores a terminal ID sent by a remote processor.

;RIN=*rin* [,*password*]
*rin* is a valid global resource identiification number (RIN).

*password* is a valid RIN password.

Note: The use of global resource identification numbers is described under "Managing Resources through Identification Numbers (RINS)" in the MPE Commands System reference manual.

;CHNL3=*x*
*x* is a digit specifying a channel on a line printer carriage control tape. Under MPE, channel 3 is always used for single spacing with a skip from bottom-of-form to top-of-form. The CHNL3=*x* parameter causes the Emulator to convert channel 3 codes received from the remote processor to whatever channel code (*x*) you specify. Of course it is assumed that the specified channel on the line printer carriage control tape contains the sequence of punches expected by the remote processor. This responsibility is left up to you.

;XEND
is a parameter for suppressing the CS ERR 158 printout. It is used when communicating with GE High-Speed Service to avoid an error printout when no error exists.

Note: See Appendix H, CS/3000 Trace Facility, for a description of the following variables.

;TRACE=ALL
will generate trace records for all CREAD, CWRITE, CCONTROL and CCLOSE requests. If ALL is not specified, then trace records will be written only when a CREAD, CWRITE, CCONTROL or CCLOSE procedure is completed with a transmission error.

;TRACE=,*mask*
*mask* is an octal integer preceded by a % sign (%nn) or an equivalent decimal number. It consists of a six-bit field representing the two-digit (nn) mask field. From bits 0 to 5 below (relative to the six-bit field), select the type of trace entries required and enter a % sign and the two octal numbers derived from the bit values or an equivalent decimal number:

Bit 0 = Generate STN entries (octal 40)
Bit 1 = Generate OPR and EDT entries (octal 20)
Bit 2 = Generate RCT entries (octal 10)
} First octal digit

Bit 3 = Generate RTX entries (octal 4)
Bit 4 = Generate SCT entries (octal 2)
Bit 5 = Generate STX entries (octal 1)
} Second octal digit

Note: CMP entries are automatically generated. Setting bits 0 through 5 to 0 provides the default trace mask described below.

;TRACE=,,*numentries*
*numentries* is a decimal integer for the maximum number of trace entries in a trace record, but not greater than 255. The CS trace file must be purged in order to allow a change in the number of entries (*numentries*).

;TRACE=,,,WRAP
causes trace entries that overflow the trace area (greater than *numentries*) to overlay the prior trace entries.

;DEV=*n*
*n* is either the logical device number assigned to the communications line at system generation time or the device class name (other than RJLINE) assigned to the communications line at system generation time.

Note: If several Synchronous Single Line Controllers share the same device class name, the Emulator will use the first available one.

## 3-4. DEFAULT VALUES

If the LINECODE= parameter is omitted from the RJLINE command, the default is the transmission code entered during configuration of the Emulator (see Appendix E).

If the CONNECT= parameter is omitted, the default is DIAL and no dial message is issued.

If the MAXRPB= parameter is omitted, the default is as follows:

| | |
|---|---|
| 2780, non-transparent: | 7 |
| 2780, transparent: | 4 |
| 3780, non-transparent: | 255 |
| 3780, transparent: | 6 |

Note: Transparency or non-transparency is declared in the #RJIN command which is described in paragraphs 3-7 through 3-17 of this manual.

If the ID= parameter is omitted, the default is no ID transmission.

If the RIN= parameter is omitted, the default is no RIN usage.

If the CHNL3= parameter is omitted, the default is 8.

If the XEND parameter is omitted, the default is an error condition when DLE EOT is received.

If ALL is omitted from the TRACE= parameter, the default is that only I/O errors are traced. If *mask* is omitted, the default is generation of all entry types except STN. If *numentries* is omitted, the default is a maximum of 25 trace entries per trace record. If WRAP is omitted, the default is to discard overflow trace entries.

If DEV= is omitted, the default is the opening of a device class called RJLINE.

If the LOCK= parameter is omitted (or LOCK=YES), RJE locks the stack during activity. Refer to the Comments paragraph for information about the LOCK feature.

If the PRI= parameter is omitted (or PRI=HIGH), RJE priority is established by a call to GETPRIORITY. Consequently, RJE will have higher priority than the CS subqueue. Refer to the Comments paragraph for information about the PRI parameter.

## 3-5. COMMENTS

The #RJLINE command is used only when a communications link is to be established between the Emulator and a

remote processor. If you are merely using the Emulator for printing or punching output data sets from disc or magnetic tape, the #RJLINE command is not necessary and should be omitted. In this context, RJLINE refers to a communications "line" and not to a line of code. The user must have Communication System (CS) capabilities to use the RJLINE command.

The only required parameter is emulator type which specifies whether an IBM 2780 or 3780 is being emulated.

The telephone connection can be established by dialing the remote processor via a MODEM or by receiving a call from the remote processor (either manually via a MODEM or automatically via an automatic answer capability). With a leased line or a switched line configured as a non-switched line, the connection is not broken after each call and redialing is not required.

If a CONNECT=DIAL parameter includes a phone number, the Emulator prints a dial message on the operator's console immediately after execution of the first #RJIN, #RJOUT, #RJLIST, or #RJPUNCH command. The computer operator must then dial the number, complete the connection, and respond YES or NO to let the Emulator know whether or not the telephone connection was established successfully. In response to YES, the Emulator reads the next command (batch job) or displays a # on your terminal to prompt you for the next command (interactive session). In response to NO, the Emulator issues a CSERR1, 57 or 2,57 (see Section V for error codes). If the Emulator is ready to actually transmit or receive data and the line has not yet been established, the Emulator waits until the communications link is present. In such a case, the Emulator will wait up to 15 minutes (transmit or receive) for the "Data Set Ready" (CC) signal from the MODEM to become "set." If "Data Set Ready" is not detected within the allowed time, the Emulator issues a CS ERR 1, 151 or 2, 151 error message.

If CONNECT= ANSWER is specified, the Emulator continues processing. In session mode, if the Emulator is ready to actually receive or transmit data and the line has not been established, there is no timeout and the wait is infinite. In batch mode, the connect timeout is operational; ANSWER is enabled when the Emulator executes the first #RJIN, #RJOUT, #RJLIST, or #RJPUNCH command. If the connection is not established when the Emulator is ready to transmit or receive data, an abort is necessary to terminate.

If the CONNECT= parameter is omitted, the Emulator continues processing. If the Emulator is ready to actually receive or transmit data and the line has not been established, the Emulator waits up to 15 minutes (receive or transmit) for "Data Set Ready" to become "set."

Note: CONNECT=DIAL implies that your HP 3000 Computer System is the primary station and CONNECT=ANSWER implies that it is the secondary station.

Although the CONNECT= parameter is optional, its use eliminates the problem of determining the appropriate time to dial the host system. With this parameter, the operator dials and makes the connection immediately after he receives the dial prompt (that is, immediately after the first #RJIN, #RJOUT, #RJLIST, or #RJPUNCH command executes). Without this parameter, dialing is performed at the same time but no prompt appears on the console. For this reason, always provide the CONNECT= parameter if an operator other than yourself must physically dial the telephone number. In addition, when using the #RJIN command to satisfy the requirements of the CONNECT= parameter, a complete buffer must be sent from the Emulator subsystem to the driver. For a message or file which does not fill the buffer, an #RJEOD (or #RJIN, #RJOUT, #RJLIST, or #RJPUNCH) command must be specified before you dial the host system.

If the ID= parameter is present, the Emulator transmits a terminal ID using the supplied terminal identification character string. If the ID= parameter is omitted, no terminal ID is transmitted. The Emulator only transmits the terminal ID if the first input/output operation is #RJIN (no terminal ID is transmitted if the first input/output operation is #RJOUT, #RJLIST, or #RJPUNCH). The Emulator always ignores a terminal ID received from the remote processor.

If the RIN= parameter is present, the Emulator attempts to perform a global lock (LOCKGLORIN) on the specified resource identification number (RIN). If the RIN can be locked, processing continues. If the RIN is already locked, the Emulator waits for the RIN to become available. If you specify an erroneous RIN or password, the Emulator issues a RIN ERROR:0 error message (refer to Section V of this manual). The use of resource identification numbers is described under "Managing Resources through Resource Identification Numbers (RINS)" in the MPE Commands reference manual.

Note: The RIN capability should be used whenever the possibility exists that more than one job or session will try to use the same Synchronous Single-Line Controller at the same time. This could occur if a batch Emulator job enters the system while an Emulator session is in progress or when several batch Emulator jobs enter a system which has spooling capability. If the RIN capability is not used in such situations, some of the Emulator jobs could be terminated by a CS ERR 0, 13 error condition. It should also be noted that the proper operation of the RIN capability requires the cooperation of all jobs and sessions which wish to use the particular communications interface; all must know the proper RIN to use and all must use it.

If only I/O errors are to be traced, and ALL is not specified in the TRACE parameter, the trace facility deposits in the trace file only those records in which a line I/O error occurred.

If WRAP is specified in the TRACE parameter, the trace entries are deposited in a trace record in a circular pattern. For example, if *numentries* specifies a maximum of 35 trace entries per trace record, trace entries beyond the 35th will ovarlay the first, second (and so on) trace entries in the record. When this happens, the overlayed trace entries will be missing from the listing, a warning message will appear at the start of the record in the listing stating that the records are missing.

If LOCK=YES or the LOCK=parameter is omitted, the RJE system locks its stack during activity. The LOCK=NO parameter allows the RJE stack to be managed like any other portion of memory (that is, Memory Management can swap the stack). The benefit of this is most evident in small memory configurations where there is contention between subsystems and programs with large stack requirements. The LOCK= parameter has no effect on the allocation of buffer space by the CS driver. When RJE is invoked, the driver allocates space as usual.

If PRI=HIGH or the PRI=parameter is omitted, the priority for RJE activity is established with a call to GETPRIORITY. The priority is 150 absolute in a linear subqueue (bottom of the BS and top of the CS queue). Because CS priority tends to drift down to the lower limit of the CS range, RJE tends to have higher effective priority than CS. The PRI=NORMAL parameter omits the call to GETPRIORITY and allows RJE to compete for priority in either session or batch mode like any other normal process.

## 3-6. EXAMPLES

```
#RJLINE 2780;LINECODE=ASCII;TRACE=ALL,,128
```

This command states that an IBM 2780 is to be emulated, the communications link is to be established by the computer operator dialing the remote processor via a MODEM, no terminal ID is to be transmitted, the transmission code is ASCII, the maximum number of records per transmission block is either 7 (for non-transparent mode) or 4 (for transparent mode), all events (not just errors) will be traced with an %35 mask, and 128 entries per request can be made in the trace record (additional entries will be lost).

```
#RJLINE 2780;ID="MM";LINECODE=ASCII
```

Same as above except that the Emulator is to transmit the terminal ID MM when the communications link is being established. Remember that the ID will be transmitted only if the first input/output command in the job or session is an #RJIN command.

```
#RJLINE 3780;LINECODE= EBCDIC
```

This command states that an IBM 3780 is to be emulated, the communications link is to be established by the computer operator dialing the remote processor via a MODEM, no terminal ID is to be transmitted, the transmission code is EBCDIC, and the maximum number of records per transmission block is either 255 (for non-transparent mode) or 6 (for transparent mode).

```
#RJLINE 2780;CONNECT=ANSWER;
LINECODE=ASCII
```

This command states that an IBM 2780 is to be emulated, the Emulator is to monitor the MODEM for an incoming call from the remote processor, no terminal ID is to be transmitted, the transmission code is ASCII, and the maximum number of records per transmission block is either 7 (for non-transparent mode) or 4 (for transparent mode).

```
#RJLINE 3780;MAXRPB=100;LINECODE=ASCII
```

This command states that an IBM 3780 is to be emulated, the communications link is to be established by the computer operator dialing the remote processor via a MODEM, no terminal ID is to be transmitted, the transmission code is ASCII, and the maximum number of records per transmission block is either 100 (for non-transparent mode) or 6 (for transparent mode).

# 3-7. #RJIN

## 3-8. FUNCTION

The #RJIN command initiates the transmission of one input data set from your HP 3000 Computer System to the remote processor. The term "input data set" as used in this manual means the equivalent of an input card deck in an IBM 2780/3780 environment. The input data set can be read from an interactive terminal, punched cards, punched tape, a disc file, or a magnetic tape file.

## 3-9. FORMAT

#RJIN [ *input file reference*]

[;REC=[ *start rec*] [ ,*end rec*]]

$$\left[;\text{COMPRESS}=\left\{\begin{matrix}\text{YES}\\ \text{NO}\end{matrix}\right\}\right]$$

$$\left[;\text{XPARENT}=\left\{\begin{matrix}\text{YES}\\ \underline{\text{NO}}\end{matrix}\right\}\right]$$

[;TRUNCATE={YES / NO}]

[;INCODE={ASCII / EBCDIC / BINARY}]

[;MAXSIZE=*nnn*]

[;XEOF]

Note: In all commands, Y and N can be used for YES and NO.

#RJIN *input file reference*
*input file reference* is any of the following:

logical device number
"device class name"
actual file designator
*formal file designator
} See Appendix J for input file characteristics.

@procedurename[({G / P / S})]

Note: When G is specified, the libraries are searched in the following order: Group, Account Public, System. When P is specified, the libraries are searched in the following order: Account Public, System. When S is specified or the parameter omitted, the System library is searched.

If a user-supplied procedure is specified, it must adhere to the following format:

INTEGER PROCEDURE *procname(target);*
ARRAY *target;*

The Emulator loads the procedure by issuing an MPE LOADPROC intrinsic call. The G, P, and S parameters are the equivalent of the 2, 1, and 0 *lib* parameters (respectively) of the LOADPROC intrinsic. The Emulator then calls the procedure. The procedure should move an input record into *target* and then return control to the calling program (i.e., the Emulator) passing a count (+ = words; − = bytes) and CCE. The Emulator transmits the record from *target* to the remote processor and then calls the procedure again. This continues until all input records have been transmitted, at which time the procedure should pass CCG to the Emulator. The maximum allowable size of *target* is 400 bytes for 2780 emulation and 512 bytes for 3780 emulation.

Note: The Emulator runs in the privileged mode. If you are supplying a procedure, it must be privileged. For a procedure to be privileged, you must have privileged capability and prepare (:PREP) the procedure with the privileged mode.

**CAUTION**

**If your procedure uses a significant amount of stack space (as an example, for procedure-local arrays), it may be necessary to increase the value of the STACK= parameter of the :PREP command in the installation file I00I130C.HP30130.SUPPORT and install RJE with a larger stack. If the Emulator executes with its stack locked (that is, LOCK= NO is not specified in the #RJLINE command) and the stack overflows, a SYSTEM FAILURE 150 occurs.**

;REC= [*start rec* [,*end rec*]]
*start rec* and *end rec* are MPE File System logical record numbers (not the values in positions 73-80 of the input records). These numbers must be positive integers. If both are present, *start rec* must be less than or equal to *end rec.* These parameters are used for transmitting a part of an MPE file. The first record in the file is record zero.

;MAXSIZE= *nnn*
is a parameter for extending the 80 byte limit on logical record size sent by the emulator, and is necessary to transfer USL files and program files between two HP 3000's. *nnn* is positive for length in words, with a maximum of 128, and negative for bytes, with a maximum of -256.

;XEOF
causes the Emulator to not terminate after an input file read error. This allows inputs from "foreign" magnetic tapes and other media, where the end-of-file conventions do not conform to MPE standards.

## 3-10. DEFAULT VALUES

If the input file reference parameter is omitted from the #RJIN command, the one specified in the :RJE command is used.

If *end rec* is omitted, all records beginning with *start rec* through the end of the input file are transmitted. If *start rec* is omitted, all records from the start of the file through and including *end rec* are transmitted. If the REC= parameter is omitted, the entire input file is transmitted.

If the COMPRESS= parameter is omitted, the default is NO for IBM 2780 emulation and YES for IBM 3780 emulation.

If the XPARENT= parameter is omitted, the default is NO (not transparent text).

If the TRUNCATE= parameter is omitted, the default is YES.

If the INCODE= parameter is omitted, the default is ASCII.

If the MAXSIZE= parameter is omitted, the default is an 80 byte limit on logical record size.

If XEOF is omitted, the default is Emulator termination after an input file read error.

## 3-11. COMMENTS

**3-12. GENERAL OPERATION.** The #RJIN command causes the immediate transmission of one input data set from your HP 3000 Computer System to the remote processor. The input data set may be read from an interactive terminal, a card reader, a punched tape reader, a disc file, or a magnetic tape file. If it is being read from a disc, you can specify starting and/or ending logical record numbers to transmit a part of that file.

A single #RJIN command immediately followed by an #RJOUT, #RJLIST, #RJPUNCH, #RJEOD, or #RJEND command is the equivalent of reading one input card deck through an IBM 2780 or 3780 with the end-of-file key on.

Two or more successive #RJIN command are the equivalent of reading that number of input card decks through an IBM 2780 or 3780 with the end-of-file key off. The first #RJOUT, #RJLIST, #RJPUNCH, #RJEOD, or #RJEND command following such a series of #RJIN commands implicitly performs the same function as turning the end-of-file key on during the reading of the final deck in the series.

The transmission code (ASCII or EBCDIC) is specified in the #RJLINE command. The INCODE= parameter specifies what code the input data set is in prior to transmission (ASCII or EBCDIC). If the specified input code differs from the transmission code, the necessary conversion is done automatically by the Emulator. INCODE=BINARY specifies that no conversion is to be performed; the input data is to be transmitted bit-for-bit as it appears in the input file.

Note: INCODE=BINARY implies **no** short-record truncation and **no** data compression regardless of what is specified by the TRUNCATE= and COMPRESS= parameters. INCODE=BINARY is ordinarily used when the input process is being controlled by a user-supplied procedure and you do not want the Emulator to do anything to the data other than transmit it.

Entering a CONTROL-Y subsystem break signal from the terminal during an interactive session transfers control to the RJE command mode. The signal is transmitted by holding down the CTRL key and pressing the Y key. (See the MPE Intrinsics reference manual, section IV, under CONTROL-Y Traps.) When used with an output operation such as #RJOUT, #RJPUNCH, and #RJLIST, entering CONTROL-Y before output is received terminates the output command and returns to RJE command mode; entering after output is received sends EOT to the system transmitting (host system) and returns to RJE command mode. CONTROL-Y may also be entered when performing an #RJIN operation from the session input device. This will terminate the input operation and return to command mode, but will not cause buffered data or an EOT to be transmitted, as would be the case if #RJEOD were used to terminate input.

**3-13. DATA COMPRESSION.** When data compression is being performed, the input records are compressed by having consecutive blanks transmitted as a two- byte sequence: a GS or IGS control character followed by a count specifying how many blanks are being compressed. If you want data compression performed and an IBM 2780 is being emulated, you must specify COMPRESS= YES. Conversely, if you do **not** want data compression and an IBM 3780 is being emulated, you must specify COMPRESS=NO.

Note: If INCODE=BINARY or XPARENT= YES is specified, no data compression will be performed regardless of what is specified by the COMPRESS= parameter.

**3-14. TRANSPARENT AND NON-TRANSPARENT MODES.** The Binary Synchronous Communications (BSC) conventions require the use of certain control characters. (See Appendix I.) When ASCII or EBCDIC data is being transmitted, the user's data ordinarily will not conflict with the BSC control characters. However, when binary data is being transmitted, certain bit sequences in the data could duplicate some of the BSC control characters. If this were to happen, the transmission could be accidentally fouled up. To avoid this, the transparent mode was devised. In the non-transparent mode (the normal mode of transmission), BSC control characters and data are intermixed: the data being supplied by the user and the BSC control characters being supplied by the communications interface. In the transparent mode, specified by XPARENT= YES, BSC control characters are sent as part of a two-byte DLE sequence: a DLE control

character followed by the particular BSC control character. In this case, the user's data may contain any bit pattern (if it contains the bit pattern for a DLE control character, the Emulator transmits two successive DLEs). The transparent mode is used for sending binary data or for sending ASCII or EBCDIC data which must include BSC control characters as part of the data.

**3-15. SHORT-RECORD TRUNCATION.** When short-record truncation is being performed, trailing blanks are automatically suppressed from input records. You need not include EM or IRS control characters in the data. However, EM and IRS control characters, if present in the data, will perform their intended function correctly (the Emulator will not alter such records).

If TRUNCATE= NO is included in the #RJIN command, input records which are shorter than 80 bytes are padded with blanks to make 80-byte records.

Note: If INCODE= BINARY or XPARENT= YES is specified, **no** short-record truncation will be performed regardless of whether or not TRUNCATE= YES is specified. In the transparent mode, if a record is shorter than 80 bytes, it is padded with blanks to make an 80-byte record. In the non-transparent mode with INCODE= BINARY, the data is not altered at all (it is neither truncated nor expanded).

## 3-16. EXAMPLES

```
#RJIN 6
```

Assuming that 6 was defined at system generation time as the logical device number of a card reader, this command states that one input data set is to be read through the particular card reader and be transmitted to the remote processor. The following parameters are assumed by default:

- Transmit the entire input data set.
- Perform data compression if an IBM 3780 is being emulated; do **not** if an IBM 2780 is being emulated.
- Use the non-transparent mode.
- Perform short-record truncation.
- The input records are in ASCII format.

```
#RJIN FILE6;REC= 0,99;INCODE= EBCDIC
```

Assuming that FILE6 is the actual file designator of an existing file on disc, this command states that the first 100 logical records of FILE6 are to be transmitted to the remote processor and that the input records are in EBCDIC format. The following parameters are assumed by default:

- Perform data compression if an IBM 3780 is being emulated; do not if an IBM 2780 is being emulated.
- Use the non-transparent mode.
- Perform short-record truncation.

```
#RJIN *INFILE;REC= 26,50
```

Assuming the INFILE has been equated to an existing file on disc via an MPE:FILE equation, this command states that logical records 26 through 50 of the particular file are to be transmitted to the remote processor. The following parameters are assumed by default:

- Perform data compression if an IBM 3780 is being emulated; do not if an IBM 2780 is being emulated.
- Use the non-transparent mode.
- Perform short-record truncation.
- The input records are in ASCII format.

```
#RJIN @MYPROC(G);INCODE= BINARY
```

This command states that the procedure MYPROC is to be executed and that no data conversion is to be performed. The Group Library, the Account Public Library, and the System Library (in that order) will be searched for the procedure MYPROC. The following parameters are assumed by default:

- Do not perform data compression.
- Use the non-transparent mode.
- Do not perform short-record truncation.

```
#RJIN USLFILE;XPARENT= YES;MAXSIZE= 128
```

This command states that an existing disc file name USLFILE is to be transmitted in transparent mode. The transmitted record size will be the record size read from the file or 128 words, whichever is the smaller. The following parameters are assumed by default:

- Transmit the entire data set.

- Do not perform data compression.
- Do not perform short-record truncation.

#RJIN *IBMTAPE;XPARENT=YES;INCODE=BINARY;XEOF

Assuming the IBMTAPE has been equated to a magnetic tape via an MPE:FILE equation, this command states that no data conversion is to be performed, data is to be transmitted in transparent mode and the Emulator is not to terminate on an input file error. The following parameters are assumed by default:

- Do not perform data compression.
- Do not perform short-record truncation.
- Transmit the entire data set.

#RJIN ;XPARENT=YES;INCODE=BINARY

This command states that one input data set is to be read from the input file specified in the :RJE command and be transmitted to the remote processor in the transparent mode. No data conversion will take place. The above command would most likely appear in a :JOB deck and be immediately followed by a binary data deck. The following parameters are assumed by default:

- Transmit the entire input data set.
- Do not perform data compression.
- Do not perform short-record truncation.

## 3-17. #RJOUT

### 3-18. FUNCTION

The #RJOUT command allows you to initiate the receipt of routed output data sets from the remote processor. The term "routed" means that each output record (for 2780 emulation) or output data set (for 3780 emulation) is preceded by an IBM 2780/3780 point-to-point component select code. Most remote job processing systems, such as IBM HASP/360, transmit routed output. The term "output data set" as used in this manual means a collection of records that are to be punched on cards, punched on paper tape, or printed on a line printer. The output data sets can be printed or punched directly as they are received or they can be stored in a disc or magnetic tape file for subsequent printing/punching off-line. The off-line processing must be performed by the #RJLIST and #RJPUNCH commands.

If the remote processor transmits routed output, you should use the #RJOUT command. If the remote processor transmits unrouted output or if you are printing/punching off-line from a disc or magnetic tape file, you should use the #RJLIST and #RJPUNCH commands.

Note: The #RJOUT command will accept unrouted output but generally treats it as list output.

### 3-19 FORMAT

#RJOUT [*output file reference*]
[,*count*]
[;WAIT=[*m*] [,*s*]]
[;OUTCODE={ASCII | EBCDIC | BINARY}]
[;OUTSIZE=*nnn*]

#RJOUT *output file reference*
*output file reference* is any of the following:

logical device number | "device class name" } See Appendix J for input file characteristics.

actual file designator | *formal file designator } See Appendix J for input file characteristics.

«« procedurename [( {G | P | S} )]

Note: When G is specified, the libraries are searched in the following order: Group, Account Public, System. When P is specified, the libraries are searched in the following order: Account Public, System. When S is specified or the parameter omitted, the System library is searched.

Note that if *output file reference* is present, all output received will be passed (as list output) to that file or device. To have list output passed to one file or device and punched output passed to another, omit the output file reference parameter. In such a case, the *list file* and *punch file* specified, explicitly or by default, in the :RJE command will be used.

If a user-supplied procedure is specified, it must adhere to the following format:

```
PROCEDURE procname(target,count);
  VALUE count; INTEGER count;
  ARRAY target;
```

The Emulator loads the procedure by issuing an MPE LOADPROC intrinsic call. The G, P, and S parameters are the equivalent of the 2, 1, and 0 *lib* parameters (respectively) of the LOADPROC intrinsic. The Emulator moves an output record into the array *target* and then calls the procedure. The size of the array *target* (expressed as a negative byte count) is passed as a value (count) in the procedure call. The procedure does what it wishes with the output record and then returns control to the calling program (i.e., the Emulator). This continues until all output records have been processed, at which time the Emulator ceases to call the procedure. (Appendix G contains an example of an output procedure.)

Note: The Emulator runs in the privileged mode. If you are supplying a procedure, it must be privileged. For a procedure to be privileged, you must have privileged capability and prepare (:PREP) the procedure with the privileged mode.

**CAUTION**

If your procedure uses a significant amount of stack space (as an example, for procedure-local arrays), it may be necessary to increase the value of the STACK= parameter of the :PREP command in the installation file I00I130C.HP30130.SUPPORT and install RJE with a larger stack. If the Emulator executes with its stack locked (that is, LOCK= NO is not specified in the #RJLINE command) and the stack overflows, a SYSTEM FAILURE 150 occurs.

#RJOUT, *count*
*count* is a decimal integer specifying how many output data sets are to be processed by this command.

;WAIT= *m*
*m* is a decimal integer specifying the maximum number of minutes the Emulator is to wait for each output data set.

;WAIT= ,*s*
*s* is a decimal integer specifying the maximum number of seconds the Emulator is to wait for each output data set.

Note: The *m* and *s* parameters, whether used individually or in combination, must not specify more than a 32767 second interval. In either case, the total "wait time" applies to any period of inactivity during the execution of the #RJOUT command. If, during the receipt of a data set or between data sets, a period of inactivity occurs that exceeds the specified "wait time," the Emulator issues a CS ERR 1, 209 error message and then terminates the #RJOUT command.

A WAIT=0 parameter disables the Timeout.

;OUTSIZE= *nnn*
is a parameter for altering the division of received transparent 3780 blocks into logical records; the default for these logical records is 80 bytes. *nnn* is positive for length in words, with a maximum of 128, and negative for bytes, with a maximum of -256.

## 3-20 DEFAULT VALUES

If the *output file reference* parameter is omitted from the #RJOUT command, the *list file* and *punch file* are those set by the :RJE command.

If the *count* parameter is omitted, one output data set is processed.

If the *m* parameter is omitted, the Emulator will wait the number of seconds specified in the *s* parameter for each output data set.

If the *s* parameter is omitted, the Emulator will wait the number of minutes specified in the *m* parameter for each output data set.

If both the *m* and *s* parameters are omitted, the Emulator will wait three minutes for each output data set.

If the OUTCODE= parameter is omitted, the default is ASCII.

If the OUTSIZE= parameter is omitted, the default is an 80 byte limit on logical record size. This is only significant when the Emulator is receiving transparent text in 3780 mode, with the default causing the text to be arbitrarily divided into 80-byte records.

## 3-21. COMMENTS

**3-22. GENERAL OPERATION.** The #RJOUT command defines the necessary parameters for an output operation (list and/or punch) and initiates the operation immediately. As explained in paragraph 3-23 below, the output is expected to be routed. The #RJOUT command will accept unrouted output but generally treats it as list output.

Normally you should omit *output file reference* from the #RJOUT command. If it is included in the command, all output is treated as list output to be passed to that file or device. If it is omitted, list output will be passed to the *list file* specified in the :RJE command and punched output will be passed to the punch file specified in the :RJE command.

The transmission code (ASCII or EBCDIC) is specified in the #RJLINE command. The OUTCODE= parameter specifies what code the output data sets are to be in when they are passed to the specified file or device (ASCII or EBCDIC). If the transmission code differs from the output code, the necessary conversion is done automatically by the Emulator. OUTCODE=BINARY specifies that no conversion is to be performed; the output data is to be passed to the appropriate file or device bit-for-bit as it was received (including the component select codes, which would otherwise be stripped off). OUTCODE=BINARY is ordinarily used when you are employing your own component select codes and the output is to be processed off-line at some later time by #RJLIST or #RJPUNCH under the control of a user-supplied procedure.

Entering a CONTROL-Y subsystem break signal from the terminal during an iteractive session transfers control to the RJE command mode. The signal is transmitted by holding down the CTRL key and pressing the Y key. (See the MPE Intrinsics reference manual, section IV, under CONTROL-Y Traps.) When used with an output operation such as #RJOUT, #RJPUNCH, and #RJLIST, entering CONTROL-Y before output is received terminates the output command and returns to RJE command mode; entering after output is received sends EOT to the system transmitting (host system) and returns to RJE command mode. CONTROL-Y may also be entered when performing an #RJIN operation from the session input device. This will terminate the input operation and return to command mode, but will not cause buffered data or an EOT to be transmitted, as would be the case if #RJEOD were used to terminate input.

**3-23. ROUTED AND UNROUTED OUTPUT.** Output data sets transmitted from the remote processor can be either routed or unrouted. The term "routed" means that each output record (IBM 2780) or each output data set (IBM 3780) is preceded by a component select code specifying whether the output is to be printed or punched. Most remote job processing systems, such as IBM HASP/360, transmit routed output. The #RJOUT command will correctly process routed output that employs the IBM 2780 or 3780 point-to-point component select conventions.

If two HP 3000 Computer Systems are communicating with each other using HP 2780/3780 Emulators, you and the remote user can establish your own output routing capability. This is done by devising your own procedures for formatting and transmitting input and for processing output. Use the appropriate procedure names (@ procedure name) as the input and output file references in the #RJIN and #RJOUT commands.

If you are communicating with a remote processor which transmits unrouted output, you must know the order in which the output data sets are transmitted. Use the #RJLIST and #RJPUNCH commands, in the proper sequence, to receive unrouted output.

**CAUTION**

**If the** #RJOUT, #RJLIST, and #RJPUNCH commands are misued, data can be lost.

The #RJOUT command treats all unrouted data sets as list output. Thus if an unrouted punched output data set is processed by #RJOUT, the data is passed to the list file or device and is no longer available from the remote processor.

## 3-24. EXAMPLES

```
#RJOUT ,6
```

This command states that six output data sets are to be received from the remote processor. List output data sets (routed or unrouted) and unrouted punched output data sets will be passed to the *list file* specified in the :RJE command; routed punched output data sets will be passed to the *punch file* specified in the :RJE command. After they are received, and before they are passed to the appropriate file or device, the data sets will be converted to ASCII if they are not already in ASCII. By default, the maximum "wait time" is 3 minutes.

```
#RJOUT
```

This command is the same as the above example except that one output data set is expected.

```
#RJOUT @OUTPROC;OUTCODE=BINARY
```

This command states that the user-supplied procedure OUTPROC is to be executed, and no data conversion is to be performed. The System Library will be searched for the procedure OUTPROC. By default, the maximum "wait time" is three minutes.

```
#RJOUT OFILE,10;WAIT=1,30
```

This command states that ten output data sets are to be received from the remote processor, converted to ASCII if they are not already in ASCII, and stored in the file OFILE. Note that all output received (printed and

punched, routed and unrouted) will be treated as list output and be passed to the file OFILE. If the file OFILE does not already exist, the Emulator will open a new file by that name and use it. The maximum "wait time" is 1 minute, 30 seconds.

## 3-25. #RJLIST

### 3-26. FUNCTION

The #RJLIST command allows you to do the following:

1. Initiate the receipt of unrouted list output data sets from the remote processor. The data sets can be printed directly on a line printer as they are received or they can be stored in a disc or magnetic tape file for subsequent printing off-line.

2. Receive data sets routed to the line printer.

3. Initiate the off-line printing of list output data sets which were received from a remote processor at some prior time and were stored in a disc or magnetic tape file.

The #RJLIST command defines the necessary parameters for a particular list output operation and initiates the operation immediately.

### 3-27. FORMAT

#RJLIST [*list file reference*]

[,*count*]

[;WAIT= [*m*] [,*s*]]

[;SOURCE=*source file reference*]

[;OUTCODE= {ASCII / EBCDIC / BINARY}]

[;FORMSMSG=*message*]

[;OUTSIZE=*nnn*]

[;AUTOPAGE= {YES / NO}]

#RJLIST *list file reference*
*list file reference* is any of the following:

logical device number
"device class name"
actual file designator
*formal file designator

} See Appendix J for input file characteristics.

@procedurename [( {G / P / S} )]

Note: When G is specified, the libraries are searched in the following order: Group, Account Public, System. When P is specified, the libraries are searched in the following order: Account Public, System. When S is specified or the parameter omitted, the search is in the System library.

If a user-supplied procedure is specified, it must adhere to the following format:

PROCEDURE *procname(target,count);*
VALUE *count;* INTEGER *count;*
ARRAY *target;*

The Emulator loads the procedure by issuing an MPE LOADPROC intrinsic call. The G, P, and S parameters are the equivalent of the 2, 1, and 0 *lib* parameters (respectively) of the LOADPROC intrinsic. The Emulator moves an output record into the array target and then calls the procedure. The size of the array *target* (expressed as a negative byte count) is passed as a value (*count*) in the procedure call. The procedure does what it wishes with the output record and then returns control to the calling program (i.e., the Emulator). This continues until all output records have been processed, at which time the Emulator ceases to call the procedure.

Note: The Emulator runs in the privileged mode. If you are supplying a procedure, it must be privileged. For a procedure to be privileged, you must have privileged capability and prepare (:PREP) the procedure with the privileged mode.

**CAUTION**

**If your procedure uses a significant amount of stack space (as an example, for procedure-local arrays), it may be necessary to increase the value of the STACK= parameter of the :PREP command in the installation file I00I130C.HP30130.SUPPORT and install RJE with a larger stack. If the Emulator executes with its stack locked (that is, LOCK= NO is not specified in the #RJLINE command) and the stack overflows, a SYSTEM FAILURE 150 occurs.**

#RJLIST, *count*
*count* is a decimal integer specifying how many output data sets are to be processed by this command.

;WAIT= *m*
*m* is a decimal integer specifying the maximum number of minutes the Emulator is to wait for each output data set.

;WAIT= ,*s*
*s* is a decimal integer specifying the maximum number of seconds the Emulator is to wait for each output data set.

Note: The *m* and *s* parameters, whether used individually or in combination, must not specify more than a 32767 second interval. In either case, the total "wait time" applies to any period of inactivity during the execution of the #RJLIST command. If, during the receipt of a data set or between data sets, a period of inactivity occurs that exceeds the specified "wait time," the Emulator issues a CS ERR 1, 209 error message to the operator's console and then terminates the #RJLIST command.

A WAIT=0 parameter disables the Timeout.

;SOURCE= *source file reference*
is any of the following:

logical device number
"device class name"
actual file designator
*formal file designator

;FORMSMSG= *message*
initiates the MPE File System forms message (see MPE Intrinsics reference manual, section II, under FOPEN) when printing the received file. The message terminates with an optional period.

;OUTSIZE= *nnn*
is a parameter for altering the division of received transparent 3780 blocks into logical records; the default for these logical records is 80 bytes. *nnn* is positive for length in words, with a maximum of 128, and negative for bytes, with a maximum of -256.

;AUTOPAGE= YES
specifies that the FWRITE carriage controls of %40 (single space), %60 (double space) or %304 (triple space) will be used where automatic page ejection is desired, but is not provided by the host system application.

;AUTOPAGE= NO
specifies that FWRITE carriage controls of %201 (single space), %202 (double space) or %203 (triple space) where automatic page ejection is provided by the host system application.

## 3-28. DEFAULT VALUES

If the *list file reference* parameter is omitted from the #RJLIST command, the one specified in the :RJE command is used.

If the *count* parameter is omitted, one list output data set is processed.

If the *m* parameter is omitted, the Emulator will wait the number of seconds specified in the *s* parameter for each output data set.

If the *s* parameter is omitted, the Emulator will wait the number of minutes specified in the *m* parameter for each output data set.

If both the *m* and *s* parameters are omitted, the Emulator will wait three minutes for each output data set.

The *source file reference* parameter is used only for printing data sets off-line which were received from a remote processor at some prior time and were stored in a disc or magnetic tape file. There is no default value.

If the OUTCODE= parameter is omitted, the default is ASCII.

If the FORMSMSG= parameter is omitted, forms message is not used.

If the OUTSIZE= parameter is omitted, the default is an 80 byte limit on logical record size when the Emulator is receiving transparent text as a 3780; in which case the default causes the text to be arbitrarily divided into 80-byte records. In all other cases, output record size is determined by the received data record size.

If the AUTOPAGE= parameter is omitted, the default is no automatic page ejection when printing.

## 3-29. COMMENTS

**3-30. ON-LINE OPERATION.** The #RJLIST command defines the necessary parameters for a particular list output operation and initiates the operation immediately. As explained in paragraph 3-32 below, the output is expected to be unrouted. The output can be printed directly as it is received or it can be stored in a disc or magnetic tape file for subsequent printing off-line.

The transmission code (ASCII or EBCDIC) is specified in the #RJLINE command. The OUTCODE= parameter specifies what code the output data sets are to be in after they are received (ASCII or EBCDIC). If the transmission code differs from the output code, the necessary conversion is done automatically by the Emulator.

OUTCODE= BINARY specifies that **no** conversion is to be performed; the output data is to be passed to the appropriate file or device bit-for-bit as it was received.

Entering a CONTROL-Y subsystem break signal from the terminal during an interactive session transfers control to the RJE command mode. The signal is transmitted by holding down the CTRL key and pressing the Y key. (See the MPE Intrinsics reference manual, section IV, under CONTROL-Y Traps.) When used with an output operation such as #RJOUT, #RJPUNCH, and #RJLIST, entering CONTROL-Y before output is received terminates the output command and returns to RJE command mode; entering after output is received sends EOT to the system transmitting (host system) and returns to RJE command mode. CONTROL-Y may also be entered when performing an #RJIN operation from the session input device. This will terminate the input operation and return to command mode, but will not cause buffered data or an EOT to be transmitted, as would be the case if #RJEOD were used to terminate input.

**3-31. OFF-LINE OPERATION.** The #RJLIST command defines the necessary parameters for a particular list output operation and initiates the operation immediately. The SOURCE= parameter specifies the file or device from which the output is to be read. The L, *count*, WAIT=, AUTOPAGE=, and OUTCODE= parameters have no meaning and should not be used.

**3-32. ROUTED AND UNROUTED OUTPUT.** Output data sets transmitted from the remote processor can be either routed or unrouted. The term "routed" means that each output record (IBM 2780) or each output data set (IBM 3780) is preceded by a component select code specifying whether the output is to be printed or punched. Most remote job processing systems, such as IBM HASP/360, transmit routed output. The #RJOUT command will correctly process routed output that employs the IBM 2780 or 3780 point-to-point component select conventions.

If two HP 3000 Computer Systems are communicating with each other using HP 2780/3780 Emulators, you and the remote user can establish your own output routing capability. This is done by devising your own procedures for formatting and transmitting input and for processing output. Use the appropriate procedure names (#@ procedurename) as the input and output file references in the #RJIN and #RJOUT commands.

If you are communicating with a remote processor which transmits unrouted output, you must know the order in which the output data sets are transmitted. Use the #RJLIST and #RJPUNCH commands, in the proper sequence, to receive unrouted output.

If you know that your host system will always return all list output sets followed by all punch output sets but are not sure of the number of each to be returned, enter a sequence similar to the following to ensure retrieving all output data sets:

```
#RJLIST "LP",10
#RJPUNCH PUNCHFILE,10
```

If you now received not more than 10 list output sets followed by 10 punch output sets, they will have been correctly processed.

**CAUTION**

If the #RJOUT, #RJLIST, and #RJPUNCH commands are misused, data can be lost.

The #RJOUT command treats all unrouted data sets as list output. Thus if an unrouted punched output data set is processed by #RJOUT, the data is passed to the list file or device and is no longer available from the remote processor.

If you are using the #RJLIST or #RJPUNCH command, and receive routed output that is inconsistent with the command, the Emulator will issue a routing error message and:

1. If you are running the Emulator from an interactive session, you may now issue the appropriate #RJLIST or #RJPUNCH command.

2. If you are running the Emulator from a batch job, the emulator will read the command file until a command different from the erroneous #RJLIST or #RJPUNCH command is found. If the new command is now an appropriate #RJLIST, #RJPUNCH or #RJOUT command, the output will be correctly processed. If the appropriate new command cannot be found, the Emulator will terminate and the output may be lost.

## 3-33. EXAMPLES

```
#RJLIST 4;OUTCODE=ASCII
```

Assuming that 4 was defined at system generation time as the logical device number of a line printer, this command states that one list output data set is to be printed in ASCII on the particular line printer as it is received from the remote processor. By default, the maximum "wait time" is 3 minutes.

```
#RJLIST "LP",3;OUTCODE=EBCDIC;WAIT=,90
```

Assuming that LP was defined at system generation time as the device class name of all line printers in the system, this command states that three list output data sets are to be printed in EBCDIC on the first available line printer in the system. The maximum "wait time" is 90 seconds.

```
#RJLIST "LP";SOURCE=LIST
```

Assuming that LP was defined at system generaton time as the device class name of all line printers in the system and LIST is the actual file designator of an existing file on disc, this command states that output data sets are to be read from the file LIST and be printed in ASCII on the first available line printer in the system.

## 3-34. #RJPUNCH

### 3-35. FUNCTION

The #RJPUNCH command allows you to do the following:

1. Initiate the receipt of unrouted punched output data sets from the remote processor. The data sets can be punched directly on a card punch as they are received or they can be stored in a disc or magnetic tape file for subsequent punching off-line.
2. Receive data sets routed to the punch.
3. Initiate the off-line punching of punched output data sets which were received from a remote processor at some prior time and were stored in a disc or magnetic tape file.

The #RJPUNCH command defines the necessary parameters for a particular punched output operation and initiates the operation immediately.

### 3-36. FORMAT

#RJPUNCH [*punch file reference*]

[,*count*]

[;WAIT=[*m*] [,*s*]]

[;SOURCE=*source file reference*]

[;OUTCODE={ASCII / EBCDIC / BINARY}]

[;OUTSIZE=*nnn*]

#RJPUNCH *punch file reference*
*punch file reference* is any of the following:

logical device number
"device class name"
actual file designator
*formal file designator

} See Appendix J for input file characteristics.

(« procedurename [( {G / P / S} )]

Note: When G is specified, the libraries are searched in the following order: Group, Account Public, System. When P is specified, the libraries are searched in the following order: Account Public, System. When S is specified or the parameter omitted, the search is in the System library.

If a user-supplied procedure is specified, it must adhere to the following format:

```
PROCEDURE procname(target,count);
  VALUE count; INTEGER count;
  ARRAY target;
```

The Emulator loads the procedure by issuing an MPE LOADPROC intrinsic call. The G, P, and S parameters are the equivalent of 2, 1, and 0 *lib* parameters (respectively) of the LOADPROC intrinsic. The Emulator moves an output record into the array target and then calls the procedure. The size of the array *target* (expressed as a negative byte count) is passed as a value (*count*) in the procedure call. The procedure does what it wishes with the output record and then returns control to the calling program (i.e., the Emulator). This continues until all output records have been processed, at which time the Emulator ceases to call the procedure.

Note: The Emulator runs in the privileged mode. If you are supplying a procedure, it must be privileged. For a procedure to be privileged, you must have privileged capability and prepare (:PREP) the procedure with the privileged mode.

**CAUTION**

If your procedure uses a significant amount of stack space (as an example, for procedure-local arrays), it may be necessary to increase the value of the STACK= parameter of the :PREP command in the installation file I00I130C.HP30130.SUPPORT and install RJE with a larger stack. If the Emulator executes with its stack

locked (that is, LOCK= NO is not specified in the #RJLINE command) and the stack overflows, a SYSTEM FAILURE 150 occurs.

#RJPUNCH, *count*
*count* is a decimal integer specifying how many output data sets are to be processed by this command.

;OUTSIZE=*nnn*
is a parameter for altering the division of received transparent 3780 blocks into logical records; the default for these logical records is 80 bytes. *nnn* is positive for length in words, with a maximum of 128, and negative for bytes, with a maximum of -256.

;WAIT= *m*
*m* is a decimal integer specifying the maximum number of minutes the Emulator is to wait for each output data set.

;WAIT= ,*s*
*s* is a decimal integer specifying the maximum number of seconds the Emulator is to wait for each output data set.

Note: The *m* and *s* parameters, whether used individually or in combination, must not specify more than a 32766-second interval. In either case, the total "wait time" applies to any period of inactivity during the execution of the #RJPUNCH command. If, during the receipt of a data set or between data sets, a period of inactivity occurs that exceeds the specified "wait time," the Emulator issues a CS ERR 1, 209 error message and then terminates the #RJPUNCH command.

A WAIT=0 parameter disables the Timeout.

;SOURCE= *source file reference*
is any of the following:

logical device number
"device class name"
actual file designator
*formal file designator

## 3-37. DEFAULT VALUES

If the *punch file reference* parameter is omitted from the #RJPUNCH command, the one specified in the :RJE command is used.

If the *count* parameter is omitted, one punched output data set is processed.

If the *m* parameter is omitted, the Emulator will wait the number of seconds specified by the *s* parameter for each output data set.

If the *s* parameter is omitted, the Emulator will wait the number of minutes specified by the *m* parameter for each output data set.

If both the *m* and *s* parameters are omitted, the Emulator will wait three minutes for each output data set.

The *source file reference* parameter is used only for punching data sets off-line which were received from a remote processor at some prior time and were stored in a disc or magnetic tape file. There is no default value.

If the OUTCODE= parameter is omitted, the default is ASCII.

If the OUTSIZE= parameter is omitted, the default is an 80 byte limit on logical record size when the Emulator is receiving transparent text as a 3780; in which case the default causes the text to be arbitrarily divided into 80-byte records. In all other cases, output record size is determined by the received data record size.

## 3-38. COMMENTS

**3-39. ON-LINE OPERATION.** The #RJPUNCH command defines the necessary parameters for a particular punched output operation and initiates the operation immediately. As explained in paragraph 3-41 below, the output is expected to be unrouted. The output can be punched directly as it is received or it can be stored in a disc or magnetic tape file for subsequent punching off-line.

The transmission code (ASCII or EBCDIC) is specified in the #RJLINE command. The OUTCODE= parameter specifies what code the output data sets are to be in after they have been received (ASCII or EBCDIC). If the transmission code differs from the output code, the necessary conversion is done automatically by the Emulator. OUTCODE=BINARY specifies hat no conversion is to be done; the output data is to be passed to the appropriate file or device bit-for-bit as it was received.

Entering a CONTROL-Y subsystem break signal from the terminal during an interactive session transfers control to the RJE command mode. The signal is transmitted by holding down the CTRL key and pressing the Y key. (See the MPE Intrinsics reference manual, section IV, uder CONTROL-Y Traps.) When used with an output oper tion such as #RJOUT, #RJPUNCH, and #RJLIST, entering CONTLCY before output is received terminates the output command and returns to RJE command mode; entering after output is received sends EOT to the system transmitting (host system) and returns to RJE command mode. CONTROL-Y may also be entered when performing

an #RJIN operation from the session input device. This will terminate the input operation and return to command mode, but will not cause buffered data or an EOT to be transmitted, as would be the case if #RJEOD were used to terminate input.

**3-48. OFF-LINE OPERATION.** The #RJPUNCH command defines the necessary parameters for a particular punched output operation and initiates the operation immediately. The SOURCE= parameter specifies the file or device from which the output is to be read. The *count*, WAIT=, and OUTCODE= parameters have no meaning and should not be used.

**3-41. ROUTED AND UNROUTED OUTPUT.** Output data sets transmitted from the remote processor can be either routed or unrouted. The term "routed" means that each output record (IBM 2780) or each output data set (IBM 3780) is preceded by a component select code specifying whether the output is to be printed or punched. Most remote job processing systems, such as IBM HASP/360, transmit routed output. Use the #RJOUT command to correctly process routed output that employs the IBM 2780 or 3780 point-to-point component select conventions.

If two HP 3000 Computer Systems are communicating with each other using HP 2780/3780 Emulators, you and the remote user can establish your own output routing capability. This is done by devising your own procedures for formatting and transmitting input and for processing output. Use the appropriate procedure names (*@procedurename*) as the input and output file references in the #RJIN and #RJOUT commands.

If you are communicating with a remote processor which transmits unrouted output, you must know the order in which the output data sets are transmitted. Use the #RJLIST and #RJPUNCH commands, in the proper sequence, to receive unrouted output.

If you know that your host system always returns all list output sets followed by all punch output sets but are not sure of the number of each to be returned, enter a sequence similar to the following sequence in a batch job to ensure retrieving all output sets:

```
#RJLIST "LP",10
#RJPUNCH PUNCHFILE,10
```

If you receive not more than 10 list output sets followed by 10 punch output sets, they will have been correctly processed.

**CAUTION**

If the #RJOUT, #RJLIST, and #RJPUNCH commands are misused, data can be lost.

The #RJOUT command treats all unrouted data sets as list output. Thus if an unrouted punched output data set is processed by #RJOUT, the data is passed to the list file or device and is no longer available from the remote processor.

If you are using the #RJLIST or #RJPUNCH command, and receive routed output that is inconsistent with the command, the Emulator will issue a routing error message and:

1. If you are running the Emulator from an interactive session, you may now issue the appropriate #RJLIST or #RJPUNCH command.

2. If you are running the Emulator from a batch job, the Emulator will read the command file until a command different from the erroneous #RJLIST or #RJPUNCH command is found. If the new command is now an appropriate #RJLIST, #RJPUNCH or #RJOUT command, the output will be correctly processed. If the appropriate new command cannot be found, the Emulator will terminate and the output may be lost.

## 3-42. EXAMPLES

```
#RJPUNCH 6;OUTCODE=BINARY
```

Assuming that 6 was defined at system generation time as the logical device number of a card punch, this command states that one punched output data set is to be punched on the particular card punch as it is received from the remote processor and that no data conversion is to be performed. By default, the maximum "wait time" is 3 minutes.

```
#RJPUNCH "CP",2;OUTCODE=EBCDIC;WAIT=2
```

Assuming that CP was defined at system generation time as the device class name for all card punches in the system, this command states that two punched output data sets are to be punched in EBCDIC on the first available card punch in the system as they are received from the remote processor. The maximum "wait time" is 2 minutes.

```
#RJPUNCH ;SOURCE=PNCH
```

Assuming that PNCH is the actual file designator of an existing file on disc, this command states that output data sets are to be read from the file PNCH and be passed to the file or device specified by the punch file parameter in the :RJE command.

```
#RJPUNCH PNCH;WAIT=,90
```

This command states that one punched output data set is to be passed to the file PNCH on disc as it is received from the remote processor. If the file PNCH does not already exist, the Emulator will open a new file by that name and use it. The maximum "wait time" is 90 seconds.

```
#RJPUNCH USLFILE;OUTSIZE=128
```

This command states that one output data set will be passed to the disc file USLFILE. Assuming that the Emulator is in 3780 mode and that the transmitting station is sending transparent records with MAXSIZE=128, the use of this parameter will allow the transfer of 128-word records.

## 3-43. #RJEOD

### 3-44. FUNCTION

The #RJEOD command transmits any remaining data buffered by the Emulator and an EOT (End Of Transmission) control character to the remote processor.

### 3-45. FORMAT

The format of the #RJEOD command is as follows:

```
#RJEOD
```

### 3-46. COMMENTS

Normally you will not use the #RJEOD command because the Emulator automatically transmits an EOT whenever an #RJOUT, #RJLIST, #RJPUNCH, or #RJEND command is encountered following an #RJIN command.

If you issue a series of #RJIN commands, the input data sets associated with the commands are sent together as a single transmission with no intervening EOT control characters. When an #RJOUT, #RJLIST, #RJPUNCH, or #RJEND command is subsequently encountered, the Emulator then sends an EOT.

You will normally use an #RJEOD command only when the operating characteristics of the remote processor require that an EOT be transmitted and the Emulator would not otherwise transmit one. Two examples are as follows:

1. Some remote job processing systems require that the "log-on" card be immediately followed by an EOT control character. To accomplish this, use the following command sequence:

   ```
   #RJIN ;TRUNCATE=NO
   /*SIGNON     REMOTEnn
   #RJEOD
   ```

2. Some remote job processing systems require that each input data set be terminated by an EOT control character. If you are issuing successive #RJIN commands and wish to terminate each input data set with an EOT, use the following command sequence:

   ```
   #RJIN (parameters)
   #RJEOD
   #RJIN (parameters)
   #RJEOD
   #RJIN (parameters)
   #RJEOD
   ```

Sometimes when an error condition occurs, you may be in doubt as to what the current state of the communications line is. In such a case, you can return the line to a known state (the "control" state) by transmitting an EOT control character.

## 3-47. #RJEND

### 3-48. FUNCTION

The #RJEND command terminates the HP 2780/3780 Emulator subsystem. The messages that are issued in conjunction with this command are described in Section V of this manual.

### 3-49. FORMAT

The format of the #RJEND command is as follows:

```
#RJEND
```

## 3-50. #RJIO

### 3-51. FUNCTION

The #RJIO command applies to special situations where a one-line message is sent and a variable number of lines are received in response to the message. An example is a /*$DA sent to an IBM HASP/370 system.

An #RJIO message is equivalent to the following sequence:

```
#RJIN
message
#RJEOD
#RJOUT ;WAIT=3
```

The following two examples are equivalent to the above sequence:

#RJIO *message*

*#message* (only in an interactive session, where the message starts with a special character)

The above sequence is equivalent to a #RJIO message but the RJIO may be omitted for convenience. The message in this case must have a special character, not a letter or digit, in column one.

### 3-52. FORMAT

#RJIO *message* (Batch)
#[RJIO] *message* (Session, where the first character of message is a special character.)

Thus, to send a /*$DA message to a HASP system and receive the response, the following four formats are equivalent:

1. #RJIN
   /*$DA
   #RJOUT;WAIT=3

2. #RJIN
   /*$DA
   #RJEOD
   #RJOUT;WAIT=3

3. #RJIO /*$DA

4. #/*$DA (only in an interactive session)

## 3-53. #RJINFO

### 3-54. FUNCTION

The #RJINFO command initiates a line display printing of the communications line. A display box prints on $STDLIST as an aid in diagnosing problems.

### 3-55. FORMAT

#RJINFO

Appendix H contains an example of a line display printing and an explanation of the entries.

## 3-56 #RJDEBUG

### 3-57. FUNCTION

The #RJDEBUG command sets the Emulator into the debug mode, allowing the user access to the debugging facilities as a diagnostic aid. (See the MPE DEBUG/Stack Dump reference manual.) This command should be used only in debugging user-written input and output procedures.

### 3-58. FORMAT

The format of the #RJDEBUG command is as follows:

#RJDEBUG

# PROGRAMMING AND OPERATING EXAMPLES

SECTION IV

The examples in this section show you how to write and execute Emulator command sequences that allow your HP 3000 Computer System to communicate with the following remote processors:

1. A remote job processing system such as IBM HASP/360.
2. An IBM 2780 Data Transmission Terminal.
3. Another HP 3000 Computer System that is also being controlled by an HP 2780/3780 Emulator.

Some of the examples show job (batch) mode usage and others show session usage. Each example is thoroughly annotated so you can easily follow what is happening.

Note: In the examples that communicate with a remote job processing system, the remote system is assumed to be similar to IBM HASP/360. For example, it is assumed that:

1. You do **not** have to transmit a terminal ID when establishing the communications link;
2. You must transmit a "log-on" card image immediately followed by an EOT control character after the communications link has been established and before any other input/output activity is performed; and
3. You must transmit a "log-off" card image just before terminating the communications link.

By making minor alterations to the command sequences, you can adapt these examples to communicate with most any remote job processing system.

In the example that communicates with an IBM 2780, you can adapt the example to communicate with an IBM 3780 or another similar communications terminal by making minor alterations to the #RJLINE and #RJIN commands.

The scope of each example is as follows:

Example 1

A batch job that communicates with a remote job processing system over the public telephone network. The job transmits one input data set to the remote processor and receives one list output data set in return.

Example 2

Two interactive sessions that communicate with a remote job processing system over the public telephone network. The first session transmits one input data set to the remote processor and then terminates. The second session receives the list output data set generated as a result of the first session's input being processed.

Example 3

A batch job that communicates with a remote location over a private leased line. Data is received from a remote terminal or system, processed in a user output procedure and a result data set returned.

Example 4

A batch job that communicates with a remote job processing system over the public telephone network. The job deck is a combination of card decks supplied by several users and the computer operator. The job transmits three input data sets to the remote processor and receives three list output data sets and one punched output data set in return.

Example 5

An interactive session that communicates with an IBM 2780 Data Transmission Terminal. The Emulator command sequence is permanently stored in a file on disc and is executed periodically by the computer operator by means of a three-command MPE session. The Emulator command sequence receives a variable number of punched output data sets that, when received, are stored in a disc file.

Example 6

An interactive session that transfers a file from one HP 3000 Computer System to another. The command sequence for both computer systems is shown in parallel.

Example 7

An interactive session that communicates with an IBM host which supports HASP, emulating a 3780 Data Transmission Terminal. A FORTRAN program, stored in an HP 3000 disc file, is sent to the host where it executes to generate output for one standard and two special forms.

## 4-1. EXAMPLE 1: RJE BATCH JOB (ONE INPUT DATA SET, IMMEDIATE OUTPUT)

This example shows you how to put together a complete, self-contained remote job entry (RJE) batch job which a computer operator could then run on your HP 3000 Computer System without adding anything to the job. The entire job is in the form of a punched card deck. It contains one input data set. One list output data set is expected as a result of the input being processed.

The example is based on the following assumptions:

**Mode of Operation:** Batch job.

**Device Emulated:** IBM 2780.

**Remote Processor:** A remote job processing system that requires a /*$LOGON card immediately followed by an EOT control character at the beginning of the run and a /*$LOGOFF card at the end of the run.

**Communications Link:** Public telephone network. The computer operator of your HP 3000 Computer System must manually make the telephone connection with the remote processor.

**Transmission Code:** EBCDIC.

**Input Code:** ASCII. The Emulator automatically converts the input data from ASCII to EBCDIC at transmission time.

**Command File:** Card reader ($STDINX, the standard job input device).

**Input File:** Card reader ($STDINX, the standard job input device).

**List File:** Line printer ($STDLIST, the standard job listing device).

**Punch File:** Not used.

The job deck is as follows:

```
    :JOB username.acctname
1   :RJE
2   #RJLINE 2780;LINECODE=EBCDIC;CONNECT=DIAL,"409-247-7000"
3   #RJIN ;TRUNCATE=NO
    /*SIGNON     REMOTE37
4   #RJEOD
5   #RJIN
    Input
    Data Set
6   #RJOUT ;WAIT=10
7   #RJIN ;TRUNCATE=NO
    /*SIGNOFF
8   #RJEND
    :EOD
    :EOJ
```

The annotations for the job deck are as follows:

1. This command invokes the Emulator and defines the four Emulator files by default as follows:

| | |
|---|---|
| Command File: | $STDINX |
| Input File: | $STDINX |
| List File: | $STDLIST |
| Punch File: | $OLDPASS (if it exists) or $NEWPASS (if $OLDPASS does not exist) |

2. This command states that an IBM 2780 is to be emulated, the transmission code is to be EBCDIC, the communications link is to be established by your computer operator manually dialing the remote processor (409-247-7000) via a MODEM, the maximum number of records per transmission block is to be 7 (non-transparent mode) or 4 (transparent mode), no terminal ID is to be transmitted, and no RIN is to be used.

3. This pair of cards specifies that an 80-column card image is to be transmitted to the remote processor. The /*SIGNON card satisfies the log-on requirements of the host computer.

4. This command completes transmission of the above #RJIN sequence, then sends an EOT control character to the remote processor. The Emulator now issues a dial message to the operator's console. In response, the computer operator dials 409-247-7000 on the MODEM and then enters YES or NO through the operator's console to let the Emulator know whether or not the remote processor answered.

5. This command transmits one input data set to the remote processor. The input data set immediately follows the #RJIN command in the job deck. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the non-transparent mode is to be used, short-record truncation is to be performed, and the input data set is in ASCII format. The Emulator will automatically convert the input to EBCDIC at transmission time.

6. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, and then causes the Emulator to turn the line around and wait for output from the remote processor. One output data set will be processed. When received, it will be passed to the file named $STDLIST (the standard job listing device). If transmission of the expected output does not begin within 10 minutes, the Emulator will terminate the #RJOUT command and read the next Emulator command.

7. This pair of cards transmits an 80-column card image to the remote processor. The /*SIGNOFF card terminates the RJE session.

8. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

## 4-2. EXAMPLE 2: RJE SESSION (ONE INPUT DATA SET, DEFERRED OUTPUT)

This example shows you how to transmit a remote job entry (RJE) input data set to a remote processor by means of a session and then retrieve the associated output by means of a subsequent session. In the first session, the Emulator commands are entered through an interactive terminal and the input data set is read from a disc file. In the second session, the Emulator commands are again entered through an interactive terminal while the output data set is passed to a line printer.

The example is based on the following assumptions:

**Mode of Operation;** Interactive session.

**Device Emulated:** IBM 2780.

**Remote Processor:** A remote job processing system that requires a /*$LOGON card immediately followed by an EOT control character at the beginning of the run and a /*$LOGOFF card at the end of the run.

**Communications Link:** Public telephone network. The computer operator of your HP 3000 Computer System must manually make the telephone connection with the remote processor.

**Transmission Code:** EBCDIC.

**Input Code:** ASCII. The Emulator automatically converts the input data from ASCII to EBCDIC at transmission time.

**Command File:** Interactive terminal ($STDINX, the terminal through which the session was initiated).

**Input File:** $STDINX for /*SIGNON and /*SIGNOFF card images and disc file (INFIL) for input data set.

**List File:** Line printer (logical device number 6).

**Punch File:** Not used.

The first session is as follows:

```
:HELLO username.acctname
:RJE                                          (1)
#RJLINE 2780;LINECODE=EBCDIC                  (2)
;CONNECT=DIAL,"247-7000"
#RJIN ;TRUNCATE=NO                            (3)
/*SIGNON     REMOTE72 password
#RJEOD                                        (4)
#RJIN INFIL                                   (5)
#RJEOD                                        (6)
#RJIN ;TRUNCATE=NO                            (7)
/*SIGNOFF
#RJEND                                        (8)
:BYE
```

The second session is as follows:

```
:HELLO username.acctname
:RJE
#RJLINE 2780;LINECODE=EBCDIC
;CONNECT=DIAL,"247-7000"
#RJIN ;TRUNCATE=NO
/*SIGNON     REMOTE72 password
#RJOUT 6;WAIT=5                               (9)
#RJIN ;TRUNCATE=NO
/*SIGNOFF
#RJEND
:BYE
```

The annotations for the above sessions are as follows:

Note: Annotations 1, 2, 3, 7, and 8 for the first session also apply to the corresponding commands in the second session.

1 This command invokes the Emulator and defines the four Emulator files by default as follows:

| | |
|---|---|
| Command File: | $STDINX (the interactive terminal through which the session was initiated) |
| Input File: | $STDINX |
| List File: | $STDLIST (the interactive terminal through which the session was initiated) |
| Punch File: | $OLDPASS (if it exists) or $NEWPASS (if $OLDPASS does not exist) |

2 This command states that an IBM 2780 is to be emulated, the transmission code is to be EBCDIC, the communications link is to be established by your computer operator manually dialing the remote processor (247-7000) via a MODEM, the maximum number of records per transmission block is to be 7 (non-transparent mode) or 4 (transparent mode), no terminal ID is to be transmitted, and no RIN is to be used.

3 This pair of entries specifies an 80-column card image is to be transmitted to the remote processor. The /*SIGNON card satisfies the log-on requirements of the host system.

4 This command completes transmission of the above #RJIN sequence, then sends an EOT control character to the remote processor. The Emulator now issues a dial message to the operator's console. In response, the computer operator dials 247-7000 on the MODEM and then enters YES or NO through the operator's console to let the Emulator know whether or not the remote processor answered.

Note: Since the Command File and the Input File are the same and the Emulator is currently in the input mode, the # prompt character is not issued by the Emulator and must be entered by the user as part of the RJEOD command.

5 This command transmits one input data set to the remote processor. The input data set is read from the disc file INFIL. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the non-transparent mode is to be used, short-record truncation is to be performed, and the input data set is in ASCII format. The Emulator will automatically convert the input to EBCDIC at transmission time.

6 This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command.

7 This pair of entries transmits an 80-column card image to the remote processor. The /*SIGNOFF card terminates the RJE session.

8 This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

Note: Since the Command File and the Input File are the same and the Emulator is currently in the input mode, the # prompt character is **not** issued by the Emulator and must be entered by the user as part of the RJEND command.

9 This command completes transmission of the above #RJIN sequence, sends an EOT control character to the remote processor, and causes the Emulator to turn the line around and wait for output from the remote processor. One output data set will be processed. When received, it will be passed to logical device 6 (assumed to be a line printer in this example). If transmission of the expected output does not begin within 5 minutes, the Emulator will terminate the #RJOUT command and prompt you for the next command.

Note: Since the Command File and the Input File are the same and the Emulator is currently in the input mode, the # prompt character is **not** issued by the Emulator and must be entered by the user as part of the RJOUT command.

## 4-3. EXAMPLE 3: RJE BATCH JOB (PUNCH OUTPUT TO A USER PROCEDURE)

This example shows you how to receive data, process it in a user-written procedure and return result data.

The example is based on the following assumptions:

**Mode of Operation:** Batch job.

**Device Emulated:** IBM 3780.

**Remote Processor:** A remote job processing system that requires a /*SIGNON card immediately followed by an EOT control character at the start of the run and a /*SIGNOFF card at the end of the run.

**Communications Link:** Private leased line. The communications link is automatically established when an #RJLINE command is processed.

**Transmission Code:** ASCII.

**Input Code:** Binary.

**Command File:** Card reader ($STDINX), the standard job input device.

**Input File:** RTNDATA

**List File:** Line printer ($STDLIST, the standard job listing device).

**Punch File:** User-supplied procedure PUNCHPROC in the user's Group Library.

The job deck is as follows:

```
:JOB username.acctname
1 :RJE
2 #RJLINE 3780;LINECODE=ASCII
3 #RJPUNCH @PUNCHPROC(G);OUTCODE=BINARY;OUTSIZE=128;WAIT=180
4 #RJIN RTNDATA;XPARENT=YES;INCODE=BINARY;MAXSIZE=128
5 #RJEND
:EOD
:EOJ
```

The annotations for the job deck are as follows:

1. This command invokes the Emulator and defines the four Emulator files **by default** as follows:

   Command File: $STDINX

   Input File: $STDINX

   List File: $STDLIST

   Punch File: $OLDPASS (if it exists) or $NEWPASS (if $OLDPASS does not exist)

2. This command states that an IBM 3780 is to be Emulated, the transmission code is ASCII, the maximum number of records per transmission block is to be 255 (non-transparent mode) or 6 (transparent mode), no terminal ID is to be transmitted and no RIN is to be used. Since the communications link is a private leased line, the connection with the remote processor is established automatically when the #RJLINE command is executed.

3. This command states that there will be a waiting period of up to three hours to receive one output data set, received data will be processed by the user-supplied procedure PUNCHPROC, no code conversion will be performed by the Emulator, output record size will be either the received record size, if the data is non-transparent, or 128 words, if the data is transparent. See Appendix G for an example of a user output procedure.

4. This command transmits the contents of the file RTNDATA in transparent mode, no data compression, truncation or code conversion will be performed, and the input record size will be the record size of the file RTNDATA or 128 words, whichever is less. This file might have been created and written by the user procedure, based upon received data.

5. This command transmits an EOT control character to the remote processor, thus completing the transmission indicated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

## 4-4. EXAMPLE 4: RJE BATCH JOB (MULTIPLE USER INPUT, IMMEDIATE OUTPUT)

This example shows how you, other users, and a computer operator can together build a complete remote job entry (RJE) batch job which transmits your input and the other users' input to a remote processor and retrieves the output immediately. The entire job is in the form of a punched card deck. It contains three input data sets. Three list output data sets and one punched output data set are expected as a result of the input being processed.

The example is based on the following assumptions:

| | |
|---|---|
| **Mode of Operation:** | Batch job. |
| **Device Emulated:** | IBM 2780. |
| **Remote Processor:** | A remote job processing system that requires a /*SIGNON card immediately followed by an EOT control character at the start of the run and a /(SIGNOFF card at the end of the run. |
| **Communications Link:** | Public telephone network. The computer operator of your HP 3000 Computer System must manually make the telephone connection with the remote processor. |
| **Transmission Code:** | ASCII. |
| **Input Code:** | ASCII. |
| **Command File:** | Card reader (\$STDINX, the standard job input device). |
| **Input File:** | Card reader (\$STDINX, the standard job input device). |
| **List File:** | Line printer (\$STDLIST, the standard job listing device). |
| **Punch File:** | Any card punch in the system (in this example CP is assumed to be the device class name for all card punches in the system). |

The job deck is as follows:

```
    :JOB username.acctname
1   :FILE PNCH;DEV=CP
2   :RJE ,,,*PNCH
3   #RJLINE 2780;CONNECT=DIAL,"247-7000";LINECODE=ASCII
4   #RJIN ;TRUNCATE=NO
    /*SIGNON     REMOTE18
5   #RJEOD
```

Supplied by the computer operator to the batch deck.

```
6   #RJIN
      Input Data
      Set #1
7   #RJEOD
```

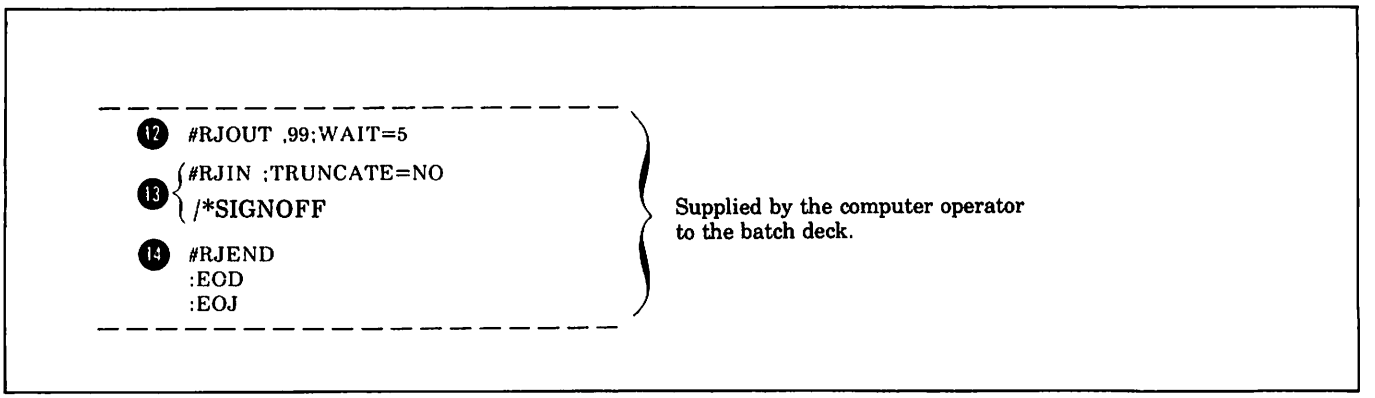
Supplied by you.

```
8   #RJIN
      Input Data
      Set #2
9   #RJEOD
```

Supplied by user "A."

```
10  #RJIN
      Input Data
      Set #3
11  #RJEOD
```

Supplied by user "B."




The annotations for the job deck are as follows:

1. This command equates the device class name CP to the formal file designator PNCH.

2. This comand invokes the Emulator and defines the four Emulator files as follows:

| | |
|---|---|
| Command File: | $STDINX (the standard job input device). |
| Input File: | $STDINX. |
| List File: | $STDLIST (the standard job listing device). |
| Punch File: | Any card punch in the system. |

3. This command states that an IBM 2780 is to be emulated, the transmission code is to be ASCII, the communications link is to be established by your computer operator manually dialing the remote processor (247-7000) via a MODEM, the maximum number of records per transmission block is to be 7 (non-transparent mode) or 2 (transparent mode), no terminal ID is to be transmitted, and no RIN is to be used.

4. This pair of cards transmits an 80-column card image to the remote processor. The /*SIGNON card satisfies the log-on requirements of the host system.

5. This card transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command.

6. This command transmits the first input data set to the remote processor. The input data set immediately follows the #RJIN command in the job deck. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the non-transparent mode is to be used, short-record truncation is to be performed, and the input data set is in ASCII format.

7. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command.

8. This command transmits the second input data set to the remote processor. The input data set immediately follows the #RJIN command in the job deck. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the non-transparent mode is to be used, short-record truncation is to be performed, and the input data set is in ASCII format.

9. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command.

10. This command transmits the third input data set to the remote processor. The input data set immediately follows the #RJIN command in the job deck. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the non-transparent mode is to be used, short-record truncation is to be performed, and the input data set is in ASCII format.

11. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command.

12. This command causes the Emulator to turn the line around and wait for output from the remote processor. The #RJOUT command states that 99 output data sets are expected. This number was chosen as a convenience so that the same #RJOUT card could be used in many different runs without having to be altered. The #RJOUT command will be terminated by the Emulator whenever 5 minutes elapses without anything being received from the remote processor.

13. This pair of cards transmits an 80-column card image to the remote processor. The /*SIGNOFF card terminates the RJE session.

14. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

## 4-5. EXAMPLE 5: SESSION EXECUTING A DISC-RESIDENT EMULATOR COMMAND SEQUENCE

This example shows you how to execute a disc-resident Emulator command sequence by means of a three-command MPE session. Such a situation arises when a commonly-used Emulator command sequence must be executed on a regular basis.

The case shown in this example involves an installation that must, several times a day, establish a connection with a remote IBM 2780, accept a variable number of punched output data sets, and then terminate the connection.

This example is based on the following assumptions:

**Mode of Operation:** Interactive session.

**Device Emulated:** IBM 2780.

**Remote Processor:** An IBM 2780 which transmits order processing information in the form of 80-character records, one record per order. The orders are grouped into data sets, one data set per product line.

**Communications Link:** Public telephone network. The operator of your HP 3000 Computer System must manually make the telephone connection with the remote processor.

**Transmission Code:** ASCII.

**Output Code:** ASCII.

**Command File:** A disc file named RJCMND.

**Input File:** Not used.

**List File:** $STDLIST (the interactive terminal through which the session is initiated).

**Punch File:** A disc file named ORDFIL.

Assume that the following Emulator command sequence is stored in the disc file named RJCMND:

```
1  #RJLINE 2780;CONNECT=DIAL,"247-7000";LINECODE=ASCII
2  #RJPUNCH ,99;WAIT=10
3  #RJEND
```

When the Emulator command sequence is to be executed, the computer operator executes the following MPE commands through an interactive terminal:

```
4  :HELLO username.acctname
   :RJE RJCMND,,,ORDFIL
   :BYE
```

The annotations for the example are as follows:

1. This command states that an IBM 2780 is to be emulated, the transmission code is ASCII, the communications link is to be established by the computer operator manually dialing the remote processor (247-7000) via a MODEM, the maximum number or records per transmission block is 7 (non-transparent mode) or 2 (transparent mode), no terminal ID is to be transmitted and no RIN is to be used.

2. The Emulator now issues a dial message to the operator's console. The computer operator dials 247-7000 on the MODEM and then enters YES or NO through the operator's console to let the Emulator know whether or not the remote processor answered. The Emulator will now wait for output from the remote processor. The #RJPUNCH command states that 99 unrouted punched output data sets are expected. This number was chosen so that the Emulator command sequence could be used repeatedly without the #RJPUNCH command having to be altered. The #RJPUNCH command is terminated automatically by the Emulator whenever 10 minutes elapse without anything being received from the remote processor. When received, the output data sets will be passed to the disc file named ORDFIL.

3. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

4. This command invokes the Emulator and defines the four Emulator files as follows:

| | |
|---|---|
| Command File: | A disc file named RJCMND. |
| Input File: | Not used. |
| List File: | $STDLIST (the interactive terminal through which the session is initiated). |
| Punch File: | A disc file named ORDFIL. |

## 4-6. EXAMPLE 6: HP 3000 TO HP 3000 FILE TRANSFER

This example shows you how to pass a file from one HP 3000 Computer System to another using the Emulator. The Emulator command sequence for **both** computer systems is shown in parallel. Since the receiving system (System B in this example) must start executing Emulator commands **before** the telephone connection is established, the overall operation must be coordinated either by a schedule or by verbal communication between the operators of the two computer systems. In this example it is assumed that the file transfer takes place at 4:00 PM everyday.

The example is based on the following assumptions:

| | SYSTEM A | SYSTEM B |
|---|---|---|
| **Mode of Operation:** | Interactive session. | Interactive session. |
| **Device Emulated:** | IBM 3780 | IBM 3780 |
| **Communications Link:** | Public telephone network. The computer operator must manually make the telephone connection with System B via a MODEM. | Public telephone network. The telephone connection is automatically established when an incoming call is detected by the MODEM after the #RJLINE command has been executed. |
| **Transmission Code:** | ASCII. | ASCII. |
| **Input Code:** | ASCII. | Not applicable. |
| **Output Code:** | Not applicable. | ASCII. |
| **Command File:** | $STDINX (the interactive terminal through which the session was initiated). | $STDINX (the interactive terminal through which the session was initiated). |
| **Input File:** | A disc file named XFILE. | Not applicable. |
| **Punch File:** | Not applicable. | A disc file named HPFILE. |

The command sequences are as follows:

| SYSTEM A | SYSTEM B |
|---|---|
| 3:55 PM | :HELLO *username.acctname* |
| | :RJE |
| | ❶ #RJLINE 3780;CONNECT=ANSWER ;LINECODE=ASCII |
| | ❷ #RJPUNCH HPFILE;OUTSIZE=128 |
| 4:00 PM:HELLO *username.acctname* | |
| :RJE | |
| ❸ #RJLINE 3780;CONNECT=DIAL,"123-4567" ;LINECODE=ASCII | |
| ❹ #RJIN XFILE;XPARENT=Y;MAXSIZE=128 | |
| ❺ #RJEND | ❻ #RJEND |
| :BYE | :BYE |

The annotations for the command sequences are as follows:

1. This command states that an IBM 3780 is to be Emulated, the transmission code is to be ASCII, the MODEM is to start monitoring the line for an incoming call, no terminal ID is to be transmitted and no RIN is to be used.

2. As soon as the communications link has been established, this command causes the Emulator to accept an output data set from System A. When received, the output data set will be passed to the file named HPFILE on disc. Since the received data is transparent, it will be processed as 128-word records rather than 80-byte records as would be the case if the OUTSIZE parameter were not specified. At this point in the run the Emulator merely waits (pauses) until the communications link is established and data transmission is begun.

3. This command states that an IBM 3780 is to be emulated, the transmission code is to be ASCII, the communications link is to be established by the computer operator manually dialing System B (123-4567) via a MODEM, no terminal ID is to be transmitted and no RIN is to be used.

4. This command transmits one input data set to System B. The input data set is read from the file named XFILE on disc. The #RJIN command states that the entire input data set is to be transmitted, no data compression is to be performed, the transparent mode is to be used, short record truncation is not to be performed, the input data set is in ASCII format and the maximum input record size is 128 words.

5. This command transmits an EOT control character to System B, thus completing the transmission initiated by the preceding #RJIN command, and then terminates the HP 2780/3780 Emulator.

6. This command transmits an EOT control character to the remote processor, thus completing the transmission initiated by the preceding #RJIN command, terminates the HP 2780/3780 Emulator subsystem, and performs a line disconnect operation.

## 4-7. EXAMPLE 7: A FORTRAN JOB SENDS OUTPUT TO MULTIPLE FORMS

At times, persons using the RJE Emulator would like to vary the forms for the output of different data sets. This example sends output to one standard and two special forms from the FORTRAN program shown in Figure 4-1. It illustrates the commands submitted at the terminal, the prompts printed on the system console, the dialing procedure, and the replies required of the console operator in order to retrieve the output.

The example uses the standard device "Operator-controlled Mode" that is supported by HASP. For more information about this mode, refer to pages 45 through 49 of the OS/VS2, HASP II Version 2, Operator's Guide (IBM part number GC27-6993-0). On the HP 3000, the example uses the normal spooling capabilities to the line printer.

The example is based on the following assumptions:

| | |
|---|---|
| **Mode of Operation:** | Interactive session. |
| **Device Emulated:** | IBM 3780. |
| **Remote Processor:** | IBM host computer supporting HASP. |
| **Communications Link:** | Half-duplex modem operating with a dial network. |
| **Transmission Code:** | EBCDIC. |
| **Input Code:** | ASCII. |
| **Command File:** | Interactive session terminal. |
| **Input File:** | IBMF3. |
| **List File:** | Line printer (logical device 6). |
| **Punch File:** | Not used. |

Legend of Terminal and Console Prompts:

| | |
|---|---|
| : | MPE prompt for input from a user |
| # | Emulator prompt for input from a user |
| /* | IBM job control command |
| /*$ | HASP subsystem command |
| $ | HASP response to a user |
| = | Console operator prompt for MPE attention (control-A) |
| ? | MPE prompt to the console operator for a reply |

NOTE: In the following dialogue, all input to the system is underlined.

```
:HELLO username.acctname
1 :FILE LP;DEV=LP
2 :RJE
  HP30130C.00.02 RJE/3000 (CS A.00.06)
    WED, APR 20, 1977, 7:19 PM
3 #RJLINE 3780;DEV=14;LINECODE=EBCDIC;CONNECT=DIAL, 123-4567
4 #RJIN
  /*SIGNON          REMOTE30
  #RJEOD
                5 (at the console)
                  ?ST/19:32/28/LDEV#14  DIAL 123-4567. ANSWER? (Y/N)
                  =REPLY 28,Y
6 #RJIN IBMF3
  #RJEOD
7 #RJOUT;WAIT=2
  $19.33.04  JOB  116 ON RM30.RD1 - GSDX2769  BRAWN
```

```
8  #/*$DJ116
   $19.39.00 JOB 116 GSDX2769 EXECUTING G PRIO 4

9  #/*$DF
   $19.50.19 OUT 30 F=C333 C=6   T=**** CLS C= 1
   $19.50.19 OUT 30 F=C777 C=6   T=**** CLS C= 1
   $19.50.19 OUT 30 F=STD. C=6   T=**** CLS A= 2

10 #/*ST RM30.PR1,F=C333,Q=C
   $20.09.22 OK

11 #RJLIST *LP;FORMSMSG=C345

        12 (at the console)
           IO/20:11/14/FORMS: C345
           ?IO/20:11/14/SP#6/ IS #S101;LP ON LDEV#6? (Y/N)
           =REPLY 14,Y
           ?IO/20:16/14/LDEV#6 FORMS ALIGNED OK? (Y/N)
           =REPLY 14,Y

13 #/*ST RM30.PR1,F=STD.,Q=A
   $20.11.04 OK

14 #RJLIST *LP;WAIT=2

        15 (at the console)
           ?IO/20:17/14/SP#6/ IS #S101;LP ON LDEV#6 (Y/N)
           =REPLY 14,N
           ST/20:18/SP#6/#0195 DEFERRED

16 #/*ST RM30.PR1,F=C777,Q=C
   $20.12.23 OK

17 #RJLIST *LP;FORMSMSG=C778

        18 (at the console)
           IO/20:18/14/FORMS: C778
           ?IO/20:18/14/SP#6/ IS #S101;LP ON LDEV#6 (Y/N)
           =REPLY 14,Y
           ?IO/20:23/14/LDEV #S6 FORMS ALIGNED OK? (Y/N)
           =REPLY 14,Y

   #/*SIGNOFF
   #RJEND
    END OF SUBSYSTEM

   :BYE

        19 (at the console)
           =ALTFILE #0195;OUTPRI=2
           IO/20:26/14/STANDARD FORMS
           ?IO/20:26/14/SP#6/ IS #S101;LP ON LDEV#6 (Y/N)
           =REPLY 14,Y
```

The annotations for the terminal session and console activity follow:

1. This file definition for the line printer will be used from the Emulator subsystem for #RJLIST output.

2. This command invokes the Emulator and defines the Emulator files by default.

3. This command states that the IBM 3780 is to be emulated. It assigns logical device 14 to the communications line and specifies EBCDIC as the linecode. The console operator will be prompted to dial 123-4567 at the appropriate time.

4. The #RJIN command provides the signon sequence for the host computer. The #RJEOD command completes the buffer and signals the Emulator to prompt the console operator with the phone dial message.

5. The dial message is printed on the system console. Before responding, the operator places the call, receives the carrier tone, and presses the DATA button. Then he types YES, thus completing the dialing sequence.

6. This command transmits the FORTRAN program to the remote processor from an HP 3000 disc file named IBMF3. The input is in ASCII format but the Emu-

lator automatically converts it to EBCDIC for transmission.

The program uses the FORTGCLG IBM subsystem. It also uses the standard JCL provided by the host except that logical devices 3 and 7 are respecified for special forms. The STP3 file equations, shown in the program listing in Figure 4-1, replace the standard JCL. However, because the JCL provided by the host is installation dependent, the STP3 file equations may differ for each IBM host system.

7 The #RJOUT allows the HASP system to transmit the number it has assigned to this job, J116 on the host system. At this point, the job has been submitted through the HASP subsystem, it has been spooled on the host input spooling device, the JCL library procedure has been added to the job, and the standard JCL definitions for devices 3 and 7 have been replaced by the //STP3 file equations in the FORTRAN program.

8 This command requests the status of J116. The host response indicates the job is active, awaiting execution. This means that all the required system resources are available for its execution and that the JCL is satisfactory. The job is queued by priority to await CPU execution.

9 This command requests that the host system display the status of files associated with the job. The response indicates the job has executed, its output has been spooled, and the data sets are now available to HASP for output to the physical device. The response shows that output is assigned to the special forms specified in the FORTRAN STP3 file equations, as well as to the standard printer output device (device 6).

10 This command configures the HASP printer for special form, C333. HASP can now transmit any pending data sets that require this special form (in this case, one data set).

11 This command signals HASP to transmit the first data set. It also specifies a FORMSMSG for the HP 3000 spooler. The form assigned on the HP 3000, C345, is independent of the form designated on the IBM host.

12 At this point the first data set has been transferred to a 3000 spool file. The first message to the system console states which form is required for the output. It is followed by a prompt asking if the form is available. The second prompt asks whether the form is properly aligned on the line printer. After the system operator responds Y to the questions, the output is printed in the format: HASP banner, output data, HASP banner. Note that the actual printing of spooled output is handled by the MPE Spooler subsystem, not by the Emulator. The two are independent.

13 This command reconfigures the HASP printer for standard output file(s).

14 This command signals HASP to transmit the second data set.

15 The second data set has been transferred to a 3000 spool file. However, in this case, the operator defers printing until a later time by responding No to the request for output device assignment. This changes the priority of the spool file to 0.

16 This command configures the HASP printer for special form, C777. Now HASP can transmit any data sets that require this form (in this case, one data set).

17 This command signals HASP to transmit the third data set. It also specifies a FORMSMSG for the HP 3000 spooler. The form assigned on the HP 3000, C778, is independent of the form designated on the IBM host.

18 At this point, the third, and last, data set has been transferred to a 3000 spool file. The 3000 system prompts the console operator for the special form, C778, and asks whether it is aligned on the line printer. After the operator responds Y, the output is printed.

19 Finally, in order to retrieve the output destined for the standard form (the output that was deferred earlier), the console operator alters the file's priority and replies to the subsequent prompt for device assignment.

```
//GSDX2769  JOB '6468 474420','BRAWN',CLASS=R
//* MEL BRAWN, BLDG 53U, EXT 2767
//      EXEC FORTGCLG
//STP1.SYSIN DD *
C
C OUTPUT TO THE FIRST FORM
C
      DO 50 I=1,5
   50 WRITE (6,900)I,I,I,I
  900 FORMAT (2I8,10X,'OUTPUT TO FIRST FORM',10X,2I8)
C
      DO 100 I=1,5
      DO 100 J=111111,1111110,111111
      K=J-111111
  100 WRITE (6,1000) K,K,K,K,K,K,K,K,K,K,K,K,K,K,K,K,K,K,K,K
 1000 FORMAT (1H,20I6)
C
      DO 150 I=1,5
  150 WRITE (6,900)I,I,I,I
C
C OUTPUT TO A SECOND FORM
C
      DO 200 I=1,5
  200 WRITE (7,1010)I,I,I,I
 1010 FORMAT (2I8,10X,'OUTPUT TO FORM NUMBER TWO',10X,2I8)
C
      DO 300 I=1,100
  300 WRITE (7,1030)I,I,I,I,I,I,I,I,I,I
 1030 FORMAT (10I10)
C
      DO 400 I=1,5
  400 WRITE (7,1010)
C
C OUTPUT TO A THIRD FORM
C
      DO 500 I=1,5
  500 WRITE (3,1040)I,I,I,I
 1040 FORMAT (2I8,10X,'OUTPUT TO THIRD FORM',10X,2I8)
C
      DO 600 I=1,100
  600 WRITE (3,1030)I,I,I,I,I,I,I,I,I,I
C
      DO 700 I=1,5
  700 WRITE (3,1040)I,I,I,I
C
      STOP
      END
/*
//STP3.FT03F001 DD SYSOUT=(C,,C333),
//         DCB=(RECFM=UA,BLKSIZE=133,LRECL=133)
//STP3.FT07F001 DD SYSOUT=(C,,C777),
//         DCB=(RECFM=UA,BLKSIZE=133,LRECL=133)
//
```

*These STP3 file equations specify special forms C333 and C777.*

Figure 4-1. A Sample FORTRAN Program That Uses Multiple Forms

# INTERPRETING MESSAGES

SECTION V

The HP 2780/3780 Emulator issues two types of messages:

- Informational messages.
- Error messages.

The informational messages are those which are normally issued at the start and end of an Emulator job or session (for example, END OF SUBSYSTEM). The error messages are issued only when something goes wrong (for example, **** LIST FILE ERROR: 2,34).

The Emulator displays the messages either on the job listing device (batch job) or on your terminal (interactive session).

This section presents all of the messages and describes their formats, their probable causes, and what you should do in response to them.

## 5-1 START MESSAGE

When an :RJE command is executed, the Emulator issues the following message:

HP30130x.yy.zz(CSx.yy.zz)RJE/3000 date, time

where

x
is an alphabetic character specifying what versions of the HP 2780/3780 Emulator and CS are being used.

yy
is a pair of digits specifying the version's update level.

zz
is a pair of digits specifying the version's fix level.

date
is an alphanumeric character string specifying the day of the week and the date of the particular job/session (for example: THU, MAY 27, 1976).

time
is an alphanumeric character string specifying the starting time of the particular job/session (for example: 7:45 AM).

After the start message is issued, the Emulator reads the first Emulator command (batch job) or displays a # on your terminal to prompt you for the first Emulator command (interactive session).

## 5-2. FILE SYSTEM ERRORS

Messages issued as the result of file system errors indicate that the Emulator could not open, close, read from, or write into a file using the MPE File System. A file system error always causes a batch job to terminate. In an interactive session, after issuing the error message the Emulator displays a # on your terminal to prompt you for the next Emulator command.

### 5-3. ERROR MESSAGES

The formats of the file system error messages are as follows:

**** COMMAND FILE ERROR: x,yy

**** INPUT FILE ERROR: x,yy

**** LIST FILE ERROR: x,yy

**** PUNCH FILE ERROR: x,yy

**** OUT FILE ERROR: x,yy

**** SOURCE FILE ERROR: x,yy

where

x
is a digit specifying the type of error, as follows:

0 = FOPEN error (file could not be opened)

1 = FCLOSE error (file could not be closed)

2 = FREAD or FWRITE error (file could not be read from or written into)

3 = ran out of file space

yy
is a two-digit error code returned by the MPE FCHECK procedure.

### 5-4. PROBABLE CAUSES

Error codes 0, 1, and 2 are caused by a variety of conditions, of which some typical examples are:

- The specified file (or the device containing the file) is currently being used by someone else or is not connected on-line to the computer.

- You violated the security restrictions associated with the file.
- You accidentally specified the name of a non-existent file.

Error code 3 is associated only with the LIST, PUNCH, and OUT files. It indicates that the specified file was not large enough to accommodate all of the output data set.

### 5-5. RECOVERY PROCEDURES

For error codes 0, 1, and 2, look up the two-digit error code (yy) in the table of error codes presented in connection with the FCHECK procedure in the MPE Intrinsics reference manual. This gives you a specific diagnosis of the problem. Correct the situation and then resubmit the job or proceed with the session.

For error code 3, redefine the file with larger limits and then resubmit the job or proceed with the session. Depending upon the remote computer system with which you are communicating, the output data set may or may not be recoverable.

## 5-6. LINE ERRORS

Messages issued as the result of line errors indicate that the communications link could not be established or you tried to perform on-line input/output without having established the communications link. A line error always causes a batch job to terminate. In an interactive session, after issuing the error message the Emulator displays a # on your terminal to prompt you for the next Emulator command.

### 5-7. ERROR MESSAGE

The format of the line error message is as follows:

**** LINE ERROR: x

where

x
is a digit specifying the type of error, as follows:

0= Line not yet established.

Table 5-1 shows the most frequent user line errors, their probable causes and recovery actions.

### 5-8. COMMUNICATIONS SYSTEM (CS) ERROR CODES

The format of the input/output error message is as follows:

**** CS ERROR: x, yyy, zz

where

x
is a digit specifying the type of error, as follows:

0 = Line open error

1 = Line read error

2 = Line write error

yyy
is a number specifying the CS irrecoverable error code, as shown in Table 5-2.

zz
is a number specifying the CS recoverable error code returned from the SSLC driver, as shown in Table 5-3 (zz is a function of the yyy error code and may not appear).

Table 5-2 contains the CS irrecoverable error codes while table 5-3 contains the CS recoverable error codes. An error is irrecoverable when the system is unable to recover from the error and it is up to the user to take corrective action.

Table 5-1. Common User Line Errors

| Message | Probable Cause | Recovery Procedure |
|---|---|---|
| **** CS ERR 0, 12 | You used an erroneous logical device number or device class name. | Job: Correct the #RJLINE command and resubmit the job.<br>Session: Enter a correct #RJLINE command. |
| **** CS ERR 0, 11<br>or<br>*** CS ERR 0, 13 | Another user has the line, or the line is unavailable for some other reason | Job: Resubmit when the line becomes available.<br>Session: Terminate and then restart the Emulator when the line becomes available. |
| **** LINE ERROR 0 | You tried to perform on-line I/O (#RJIN, #RJOUT, #RJLIST, or #RJPUNCH) before issuing an #RJLINE command. | Job: Add an #RJLINE command to your deck and resubmit the job.<br>Session: Issue an #RJLINE command and then proceed with the session. |

The system can recover from a recoverable error and no action is required by the user; a recoverable error code may follow an irrecoverable error code to inform the user of the error. The more frequent I/O errors, their causes and recovery actions are described in Table 5-4. Frequent deck set-up errors are listed in Table 5-5.

## 5-9. PROCEDURE ERROR

Messages issued as the result of a procedure error indicate that something was wrong with a procedure that you specified in an Emulator command. A procedure error always causes a batch job to be terminated. In an interactive session, after issuing the error message the Emulator displays # on your terminal to prompt you for the next Emulator command.

## 5-10 ERROR MESSAGE

The format of a procedure error message is:

**** PROCEDURE ERROR: xx

where

xx
is a two-digit error code. This code is described in the Loader Errors table under MPE Diagnostic Messages in the MPE Intrinsics Reference Manual.

Table 5-2. CS Irrecoverable Error Codes Used With The Emulator

| Code (Decimal) | Meaning | Code (Decimal) | Meaning |
|---|---|---|---|
| 0 | Request completed successfully. | 156 | An internal error was detected by the CS driver. |
| 6 | Invalid driver name. | | |
| 7 | Driver not in system. | 157 | Remote protocol error occurred. |
| 8 | Driver not compatible with line's attributes. | 158 | Remote station sent shutdown sequence and disconnected. |
| 9 | This line not configured to change drivers. | | |
| 10 | Undefined line. | 159 | Remote sent shutdown sequence and disconnected before the last I/O request was issued. |
| 11 | Line not available. | | |
| 12 | Not CS line. | | |
| 13 | CS line in use. | 160 | An internal error was detected by MPE. |
| 14 | ID sequence length greater than 16 characters. | 201 | Current I/O operation aborted. |
| 15 | Buffer size greater than 4096 words. | 202 | Invalid Emulator request was detected by CS driver. |
| 17 | Phone number length greater than 20 characters. | 203 | Remote not ready to receive (remote sent a NAK sequence in response to local line bid). |
| 18 | Character in phone number was not numeric or a hyphen. | 204 | Remote rejected the line bid. |
| 19 | Local mode not compatible with line type. | 205 | Remote primary station bid for the line while local station was also bidding. |
| 24 | Could not open trace file. | 206 | Remote has requested to send (an RVI sequence was received). |
| 25 | Trace file record size was too small. | | |
| 26 | User does not have CS capability. | 207 | Retry count exhausted. |
| 28 | No line specified. | 208 | Unexpected text was received. |
| 31 | Insufficient memory space. | 209 | Receive timeout occurred while waiting for text from the remote station. |
| 48 | No virtual memory available for trace and/or buffering. | 210 | Remote sent end-of-tansmission. |
| 53 | I/O error on trace file. | 211 | Remote sent end-of-transmission sequence and disconnected before the last I/O request was issued. |
| 57 | Operator responded "no" to a dial message. | | |
| 58 | No phonelist for dial attempt. | 212 | During execution of a conversational write with the output buffer specified or the input buffer, the remote station requested a resend of the output buffer but its contents had been modified while receiving from the remote. |
| 101 | Nonresponding Synchronous Single Line Controller (SSLC). | | |
| 102 | SIO transfer error. | | |
| 103 | Modem became not ready. | 213 | Remote sent an ACK sequence in response to local acknowledgment. |
| 104 | Carrier loss occurred on modem. | | |
| 105 | SIO timing error (data overrun). | 214 | Remote sent a NAK sequence in response to local acknowledgement. |
| 151 | Connect timeout occurred. | | |
| 153 | Remote station rejected the connection. | 215 | Remote sent an RVI sequence in response to local acknowledgment. |
| 154 | Power failure occurred. | | |
| 155 | Local timeout occurred. | 217 | Line bid timeout occurred while waiting for the remote system's line bid. |

Table 5-3. CS Recoverable Error Codes Used With The Emulator

| Code (Decimal) | Meaning | Code (Decimal) | Meaning |
|---|---|---|---|
| 0 | No recoverable error occurred. | 9 | Received enquiry character after sending text. |
| 1 | Invalid ID sequence received. | 10 | Remote requested a resend of local's last response. |
| 2 | Received unintelligible sequence. | | |
| 3 | Block check character error. | 11 | Remote requested a resend of last text. |
| 4 | Remote did not respond to text. | 12 | Received end-of-transmission character while line was in control state. |
| 5 | Received incorrect acknowledgment. | | |
| 6 | Remote attempted to bid for the line. | 13 | Received text overflow. |
| 7 | No response to line bid. | 14 | Data overrun occurred on SIO multiplexer. |
| 8 | Received unintelligible sequence after sending text. | 15 | Data transfer error occurred on SIO multiplexer. |

Table 5-4. Common CS Line Errors

| Message | What Happened | Probable Cause | Recovery Procedure |
|---|---|---|---|
| **** CS ERR 1, 57<br>or<br>**** CS ERR 2, 57 | You (or the computer operator) entered NO in response to the DIAL REQD message. | The remote site's telephone was busy or did not answer. | Terminate the session and try again later. |
| **** CS ERR 1, 101<br>or<br>CS ERR 2, 101 | The Synchronous Single Line Controller failed to respond to an I/O instruction. | Something may be wrong with the Synchronous Single Line Controller. | Reissue the particular Emulator command. If the same error occurs, call the nearest HP Sales and Service Office. |
| **** CS ERR 1, 102<br>or<br>**** CS ERR 2, 102<br>or<br>**** CS ERR 1, 105<br>or<br>**** CS ERR 2, 105 | A timing or parity error occurred in the HP 3000 SIO bus. | Something may be wrong in the SIO bus. | Reissue the particular Emulator command. If the same error occurs, call the nearest HP Sales and Service Office. |
| **** CS ERR 1, 103<br>or<br>**** CS ERR 2, 103 | The "Data Set Ready" (CC) signal from the MODEM has changed from "set" to "clear" (i.e., the communications link has been terminated). | The remote processor may have terminated the communications link because you accidntally specified the wrong transmission code in the #RJLINE command. | Correct the LINECODE= parameter in the # RJLINE command and reissue the command. |
| | | The remote processor may have terminated the communications link because you did not adhere to its log-on conventions. | Correct the log-on card (or the #RJIN command that sends it) and restart the session. |

Table 5-4. Common CS Line Errors (Continued)

| Message | What Happened | Probable Cause | Recovery Procedure |
|---|---|---|---|
| | | If you were entering an input data set through an interactive terminal, the remote processor may have timed-out (and terminated the communications link) because you were not entering the data fast enough. | Restart the session and enter the data faster this time. |
| | | The remote processor may have shut down in the middle of your job. | Call the operator at the remote site and find out what their problem is. |
| | | The telephone connection may be bad. | **Switched:**<br>Restart the session and re-dial.<br>**Private:**<br>Run a job that has worked okay in the past. If it fails also, call the nearest HP Sales and Service Office.<br><br>If it runs okay, something is wrong with the data your most recent job was transmitting. |
| **** CS ERR 1, 153<br>or<br>**** CS ERR 2, 153<br>or<br>**** CS ERR 2, 203<br>or<br>**** CS ERR 2, 204 | A NAK or EOT or DLE/EOT was received in response to your line bid (i.e., in response to the ENQ generated by an #RJIN command). | The remote processor requires a terminal ID and you either supplied an erroneous one or none at all. | Correct the ID= parameter in the #RJLINE command and then re-issue the #RJLINE command. |
| | | The remote processor currently does not want to receive input. | Terminate the session and try again at a later time.<br><br>or<br><br>Terminate the session and call the operator at the remote site to find out what their problem is. |
| **** CS ERR 1, 158<br>or<br>**** CS ERR 2, 158 | A DLE/EOT sequence was received. | A DLE/EOT sequence always indicates that the remote processor is terminating the communications link.<br><br>You may have set up your input deck incorrectly. Some common errors of this sort are shown in Table 5-5. | Examine your input deck for the errors shown in Table 5-5.<br><br>If you find errors, correct them and restart the session. |

Table 5-4. Common CS Line Errors (Continued)

| Message | What Happened | Probable Cause | Recovery Procedure |
|---|---|---|---|
| | | | If your deck seems to be correct, call the operator at the remote site and find out what their problem is.<br><br>Note: Some RJE systems use DLE/EOT as a job terminator after transmitting the last available output. If you were receiving output when the message was issued and if this is a characteristic of the RJE system with which you are communicating, ignore the message. This message may be inhibited by supplying the XEND parameter of #RJLINE. |
| **** CS ERR 1, 201<br>or<br>**** CS ERR 2, 201 | The operator of your HP 3000 Computer System entered an ABORTIO command for your Synchronous Single Line Controller through the operator's console. | Only your computer operator knows for sure. | Ask your computer operator. |
| **** CS ERR 1, 209 | Receive Continue Time-Out. After the remote processor began transmitting output, a period of inactivity occurred in which no data was received within the allowed time (20 seconds). | Some RJE systems terminate the final output transmission by letting the other end of the line time-out. | Check the output that you received. If it seems to be all there, ignore the error the error message. |
| | | The EOT terminating the final output data set may have been garbled. | Check the output that you received. If it seems to be all there, ignore the error message. |
| | | Something may have happened to interrupt the transmission in the middle of a data set. | Check the output that you received. If the time-out occurred in the middle of a data set, issue an #RJEOD command and then re-issue the particular Emulator output command. |
| **** CS ERR 1, 217 | Line Bid Time-Out. The remote processor did not bid for the line within the allowed time (3 minutes or the specified WAIT= time). | The remote processor currently has no output ready to send. | Try again at a later time. |

Table 5-4. Common CS Line Errors (Continued)

| Message | What Happened | Probable Cause | Recovery Procedure |
|---|---|---|---|
| **** CS ERR 2, 205 | If your HP 3000 is a secondary station, this means that an ENQ was received in response to your line bid (i.e., in response to the ENQ generated by a #RJIN command). | The remote processor want to transmit something. | Issue an #RJOUT command to find out what the remote processor wants to transmit. |
| **** CS ERR 2, 206 | An RVI was received while transmitting data. | The remote processor has a high priority message to transmit. | Issue an #RJOUT command to find out what the remote processor wants to transmit. |
| **** CS ERR 2, 207, 2 | A NAK was received in response to a transmission block on 16 successive tries. | You may accidentally be generating bisynch control codes in your data (this could result from mispunched data or from transmitting binary data in the non-transparent mode). | If you are transmitting binary data, make sure you specified XPARENT=YES in your #RJIN command.<br><br>If you are transmitting ASCII or EBCDIC data, check the data for mispunched characters (to the extent that it is practical to do so). |
| | | The communications line may be bad, causing irrecoverable parity errors. | **Switched:**<br>Restart the session and redial.<br><br>**Private:**<br>Run a job that has worked in the past. If it fails, call the nearest HP Sales and Service Office.<br><br>If it runs, something is wrong with the data your most recent job was transmitting. |
| | | There may be a hardware or software malfunction at either end of the line. | Call the nearest HP Sales and Service Office. |
| **** CS ERR 2, 207, 4 | Response Time-Out. The remote processor did not respond within 16 tries (48 seconds) when the Emulator transmitted an enquiry or data block. | You may have accidentally specified the wrong transmission code. | Correct the LINECODE= parameter in your #RJLINE command and re-issue the #RJLINE command. |
| | | You and the remote processor may both be primary stations and both be trying to transmit data as the first i/O operation. | Call the operator at the remote site and get coordinated. |

Table 5-4. Common CS Line Errors (Continued)

| Message | What Happened | Probable Cause | Recovery Procedure |
|---|---|---|---|
| | | The communications line may be bad or there may be a hardware or software malfunction at either end of the line. | **Switched:**<br>Restart the session and redial.<br>If the same error occurs, call the remote site and find out if they are having problems.<br><br>If all is okay at the remote site, call the nearest HP Sales and Service Office.<br><br>**Private:**<br>Call the remote site and find out if they are having problems. If all is okay at the remote site, call the nearest HP Sales and Service Office. |
| **** CS ERR 2, 210 | An EOT was received while transmitting data. | The remote processor for some reason is not able to continue. This may, but does not necessarily, indicate that the communications link has been terminated. | If the communications link is still present, retry the #RJIN command.<br><br>If the communications link was terminated, try to re-establish it. If it can successfully be re-established, retry the #RJIN command.<br><br>If the communications link cannot be re-established or if the #RJIN command continues to fail, call the operator at the remote site and find out what their problem is. |

Table 5-5. Common Deck Set-Up Errors

| Error | Probable Message (CS Error Message #) |
|---|---|
| Wrong LINECODE= specified in #RJJLINE command. | 103 or 207, 4 |
| Wrong emulator type specified in #RJLINE command. | 207, 2 or 207, 4 (transmitting)<br>158 (receiving)<br>It is also possible that no message will be issued but that the output will be garbled. |
| COMPRESS=NO (3780) or TRUNCATE=NO not specified in the #RJIN command when the remote processor require uncompressed or untruncated input (e.g., during log-on). | 158 or 103 |
| XPARENT=YES not specified in the #RJIN command when transmitting binary data. The same error condition can arise from mispunched ASCII or EBCDIC data. | 207, 4 or 207, 2 |
| Forgot to log-on to the remote system when log-on is required. | 158 or 103 |
| More than one user trying to log-on with the same ID (i.e., the same ID= parameter in the #RJLINE command or the same ID in the remote system's log-on procedure). The applicability of this condition is dependent upon the characteristics of the remote system with which you are communicating. | 103 |

### 5-11. PROBABLE CAUSES

Either the Emulator could not find the specified procedure or something was wrong with the procedure.

### 5-12. RECOVERY PROCEDURES

Make sure that the procedure name which you used is correct, that you are referencing the proper library, and that the procedure actually exists. If that does not solve the problem, examine the procedure itself and correct it.

## 5-13. ROUTING ERRORS

Messages issued as the result of routine errors indicate that a routed punched output data set was received when it was not expected (i.e., when a #RJLIST command was being executed) or a routed list output data set was received when it was not expected (i.e., when an #RJPUNCH command was being executed). The command which was being executed when the error occurred is terminated. In an interactive session, after issuing the error message the Emulator displays a # on your terminal to prompt you for the next Emulator command. In a batch job, processing continues only if the next Emulator command is a different type of output command than the one which was just terminated. If the next command is #RJIN, #RJLINE, #RJEOD, or #RJEND the job is terminated; if the next command is the same type of output command which was just terminated, then the routing error still exists and the next command beyond that will be examined before deciding whether or not the job should be terminated.

### 5-14. ERROR MESSAGE

The format of a routing error message is:

**** ROUTING ERROR: x

where

x
is a digit specifying the type of error, as follows:

0 = Routed list output data set was received when it was not expected.

1 = Routed punched output data set was received when it was not expected.

### 5-15. PROBABLE CAUSES

You used an #RJLIST or #RJPUNCH command when you should have used #RJOUT.

### 5-16. RECOVERY PROCEDURES

In an interactive session, issue an #RJOUT command.

In a batch job, if the job was terminated add the appropriate output command(s) to your deck and resubmit the job. Depending upon the remote computer system with which you are communicating, the output data set may or may not be recoverable. If, due to the structure of your job deck, the job was **not** terminated, make sure that you received all expected output. If you did **not** receive all expected output, be sure to include the appropriate output command(s) in your next job deck to retrieve the outstanding output.

## 5-17. COMMAND ERRORS

Messages issued as the result of command errors indicate that something was wrong with an Emulator command (for example, the command was totally unrecognizable or it contained syntax errors). A command error always terminates a batch job. In an interactive session, after issuing the error message the Emulator displays a # on your terminal to prompt you for the next Emulator command.

### 5-18. ERROR MESSAGES

The formats of the command error messages are:

**** COMMAND ERROR: x

**** SYNTAX ERROR: y,z

where

x
is a digit specifying the type of error, as follows:

0 = Invalid command (the command was totally unrecognizable).

1 = Command too long.

2 = Too many parameters.

y
is a digit specifying the type of error, as follows:

0 = Invalid keyword.

1 = Duplicate keyword.

2 = Invalid command construction.

3 = Invalid parameter.

4 = Invalid numeric parameter.

5 = Required parameter missing.

z
is a digit specifying the number of the erroneous parameter, such as 3 meaning that the third parameter is in error.

Note: When issuing the **** SYNTAX ERROR: y, z error message, the Emulator always counts leading, but not trailing, non-keyword parameters regardless of whether they were supplied or omitted. This means that if one or more of the non-keyword parameters are omitted, the parameter pointer (z) includes the omitted non-keyword parameters and you must adjust the pointer mentally in order to locate the erroneous parameter. For example, if you supplied an #RJOUT command which does not contain the *output file reference* and *count* parameters, the parameter pointer in a **** SYNTAX ERROR:y, z error message is two greater than what you might otherwise expect. In such a case, if the second supplied keyword parameter is in error, z would be four instead of the expected two.

### 5-19. PROBABLE CAUSES

You made an error in coding, keypunching, or entering the particular command.

### 5-20. RECOVERY PROCEDURES

For a batch job, correct the erroneous command and resubmit the job.

In an interactive session, correct the erroneous command and re-enter it.

## 5-21. RIN ERRORS

Messages issued as the result of a RIN error indicate that an invalid resource identification number (RIN) or an invalid RIN password was used. A RIN error always causes a batch job to be terminated. In an interactive session, after issuing the error message the Emulator displays a # on your terminal to prompt you for the next Emulator command.

### 5-22. ERROR MESSAGE

The format of a RIN error message is:

**** RIN ERROR: 0

### 5-23. PROBABLE CAUSES

You used an illegal RIN or RIN password.

### 5-24. RECOVERY PROCEDURES

For a batch job, correct the RIN or RIN password and then resubmit the job.

In an interactive session, correct the RIN or RIN password and then re-enter the command.

# IBM 2780/3780 COMPARISON

APPENDIX A

Table A-1 summarizes the features of the IBM 2780 Data Transmission Terminal and the IBM 3780 Data Communication Terminal. The information is presented in such a way that you can either reference specific information for one of the two terminals or compare the capabilities of one terminal against the other.

Table A-1. IBM 2780/3780 Comparison

| FEATURE | IBM 2780 | IBM 3780 |
|---|---|---|
| Peripheral Devices | 400 cpm card reader.<br>355 cpm card punch (optional).<br>300 lpm line printer (optional). | 600 cpm card reader.<br>359 cpm card punch (optional).<br>425 lpm line printer. |
| Half/Full Duplex Protocol | Half duplex only. | Half duplex only. |
| Permissible Communications Networks | Point-to-Point (leased or switched).<br>Multipoint (2- or 4-wire). | Point-to-Point (leased or switched).<br>Multipoint (2- or 4-wire). |
| Permissible Transmission Codes | ASCII (non-transparent mode only).<br>EBCDIC (transparent or non-transparent modes).<br>Six-bit transcode. | ASCII (non-transparent mode only).<br>EBCDIC (transparent or non-transparent modes). |
| Error Checking | ASCII: Odd parity VRC and 8-bit LRC.<br>EBCDIC: 16-bit CRC.<br>Six-bit transcode: 12-bit CRC. | ASCII: Odd parity VRC and 8-bit LRC.<br>EBCDIC: 16-bit CRC. |
| Permissible Transmission Rates | 1200, 2000, 2400, 4800 bps. | 1200, 1800, 2400, 3600, 4800, 7200 bps. |
| Buffer(s) | One 400-byte buffer used for either input or output. | Two 512-byte buffers used simultaneously for input and output. |
| Accept Horizontal Tabulation Codes? | YES (optional). | YES. |
| Accept Vertical Forms Control Codes? | YES. | YES. |
| Transmit and Receive in Transparent Mode? | YES (optional; applies only to EBCDIC mode). | YES (optional; applies only to EBCDIC mode). |
| Short-Record Truncation? | YES (user must supply EM control characters in the data). | YES (done automatically; user does not need to supply EM or IRS control characters in the data). |
| Blank Compression? | NO. | YES (done automatically; user does not need to supply any special characters in the data). |
| Maximum Blocking Factor | 2 (optionally 7). | Non-Transparent Mode: 255<br>Transparent Mode: 1 (optionally 6). |
| Generate WACK and TTD Sequences When Temporarily Unable to Transmit or Receive? | NO. | YES. |
| Intermediate Block Terminator Character | ASCII: US BCC<br>EBCDIC: IUS BCC<br>Six-bit transcode: ITB BCC | ASCII: RS (no BCC)<br>EBCDIC: IRS (no BCC)<br>Receive Mode: Will accept IUS BCC. |
| Recognize WACK? | YES (receive mode only). | YES. |
| Component Select Codes (Point-to-Point Only) | ESC followed by any carriage control code specifies printer.<br>ESC 4 specifies punch. | DC1 specifies printer.<br>DC2 or DC3 specifies punch. |
| SOH at Start of Received Text | Treated as an STX. | Not recognized. |

# MODEM OPTIONS

APPENDIX B

Tables B-1 through B-6 present the Bell System options for MODEMs which can be used with the HP 2780/3780 Emulator. Wherever possible, recommendations for which option to choose are also shown.

Modems normally used for full-duplex operation are models 201B, 208A, and 209A.
Models normally used for half-duplex operation are 201A, 201C, and 208B.

When ordering a modem, it is customary to specify that it provide continuous carrier and continuous request-to-send.

**Type of MODEM:** Bell System Type 201A3 Data Set.

**Type of Line:** Public Telephone Network (Switched).

**Transmission Rate:** 2000 bits-per-second.

Table B-1. 201A3 Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | EIA interface.<br>Contact interface. | A1 (required) |
| B3<br>B4 | With alternate voice.<br>Without alternate voice. | B3* |
| C5<br>C6 | With new sync.<br>Without new sync. | C6 (required) |
| D7<br>D8 | Half duplex (2-wire).<br>Full duplex (4-wire). | D7 |
| E9<br>E10 | 4-wire continuous carrier.<br>4-wire carrier controlled by REQUEST TO SEND. | ** |

*If option B3 is selected and automatic answering is to be used, the automatic answering capability is normally provided as a key-controlled function. If you want the automatic answer to be permanently wired, then state so in the Remarks column on the Bell System order form.

**If option D7 is selected, the E options have no meaning and should be ignored.

**Type of MODEM:** Bell System Type 201B3 Data Set.

**Type of Line:** Private Leased Line.

**Transmission Rate:** 2400 bits-per-second.

Table B-2. 201B3 Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | EIA interface.<br>Contact interface. | A1 (required) |
| B3<br>B4 | With alternate voice.<br>Without alternate voice. | B3* |
| C5<br>C6 | With new sync.<br>Without new sync. | C6 (required) |
| D7<br>D8 | Half duplex (2-wire).<br>Full duplex (4-wire). | D8 |
| E9<br>E10 | 4-wire continuous carrier.<br>4-wire carrier controlled by REQUEST TO SEND. | E9** |

*If option B3 is selected and automatic answering is to be used, the automatic answering capability is normally provided as a key-controlled function. If you want the automatic answer to be permanently wired, then state so in the Remarks column on the Bell System order form.

**If option D7 is selected, the E options have no meaning and should be ignored.

**Type of MODEM:** Bell System Type 201C Data Set (also called DATAPHONE 2400).

**Type of Line:** Public Telephone Network (Switched) or Private Leased Line.

**Transmission Rate:** 2400 bits-per-second.

Table B-3. 201C Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | Transmitter internally timed.<br>Transmitter externally timed. | A1 (required) |
| B3<br>B4 | Without 801 Automatic Calling Unit.<br>With 801 Automatic Calling Unit. | B3 |
| C5<br>C6 | EIA interface.<br>Contact interface. | C5 (required) |
| D7<br>D8 | Without automatic answer.<br>With automatic answer. | D8 |
| E9<br>E10 | Automatic answer permanently wired.<br>Automatic answer key-controlled. | Either * |

*If option D7 is selected, the E options have no meaning and should be ignored.

**Type of MODEM:** Bell System Type 208A Data Set (also called DATAPHONE 4800).

**Type of Line:** Private Leased Line.

**Transmission Rate:** 4800 bits-per-second.

Table B-4. 208A Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | Transmitter internally timed.<br>Transmitter externally timed. | A1 (required) |
| B3<br>B4 | Continuous carrier.<br>Switched carrier. | B3 |
| C5<br>C6 | Switched REQUEST TO SEND.<br>Continuous REQUEST TO SEND. | C6 |
| D7<br>D8 | One second holdover used.<br>One second holdover not used. | D7 |
| E9<br>E10 | With new sync.<br>Without new sync. | E10 (required) |
| F11<br>F12 | CC ON when analog loop is present.<br>CC OFF when analog loop is present. | F11 |

**Type of MODEM:** Bell System Type 208B Data Set (also called DATAPHONE 4800).

**Type of Line:** Public Telephone Network (Switched).

**Transmission Rate:** 4800 bits-per-second.

Table B-5. 208B Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | Transmitter internally timed.<br>Transmitter externally timed. | A1 (required) |
| B3<br>B4 | Without 801 Automatic Calling Unit.<br>With 801 Automatic Calling Unit | B3 |
| C5<br>C6 | CC OFF when analog loop is present.<br>CC ON when analog loop is present. | C6 |
| D7<br>D8 | Without automatic answer.<br>With automatic answer. | D8 |
| E9<br>E10 | Desk mounting.<br>Rack or cabinet mounting. | Either |

**Type of MODEM:** Bell System Type 209A Data Set (also called DATAPHONE 9600).

**Type of Line:** Private Leased Line (3002 type four-wire) with D1-type conditioning (no C-type conditioning).

**Transmission Rate:** 9600 bits-per-second.

**Description of Application:** Point-to-Point 9600 bps operation.

Table B-6. 209A Options and Recommendations

| OPTION NUMBER | DESCRIPTION | RECOMMENDATION |
|---|---|---|
| A1<br>A2 | Transmitter Timing supplied by Data Set.<br>Transmitter Timing supplied by Computer/Terminal. | A1 (required) |
| B3<br>B4 | Data Set Ready interface lead ON for Analog Loopback Mode.<br>Data Set Ready interface lead OFF for Analog Loopback Mode. | B4 |
| C5<br>C6 | Transmitter Timing slaved by Receiver.<br>Transmitter Timing not slaved by Receiver. | C6 |
| D7<br>D8 | Elastic Store option enabled (IN).<br>Elastic Store option disabled (OUT). | D8 |
| E9<br>E10 | Continuous Carrier operation.<br>Switched Carrier operation. | E9 |
| F11<br>F12 | Switched Request-to-Send operation.<br>Continuous Request-to-Send operation. | F12* |
| | Grounding:<br>Protective Ground to Signal Ground. | AA to AB |
| | With Alternate Voice<br>Without Alternate Voice<br>The Data Set normally is supplied without a hand set. | Either |

Notes: *F11 and F12 are significant only if E10 is selected.

If the 209A Modem will be used in multiplexing applications, the options shown do not apply. In this case, refer to *Bell System Data Communications Technical Reference, Data Set 209A Interface Specifications* (Publication 41213) for additional information.

# CHARACTER SET

APPENDIX

C

Table C-1 presents all of the ASCII, EBCDIC, and Hollerith codes which the HP 2780/3780 Emulator can handle. Note that if INCODE=BINARY or OUTCODE=BINARY is specified in an Emulator #RJIN, #RJOUT, #RJLIST, or #RJPUNCH command, no conversion occurs; the Emulator merely passes along whatever bit patterns it receives from the remote processor or from the specified input file.

## Table C-1. ASCII, EBCDIC, and Hollerith Codes

### HOW TO USE THIS TABLE

- The table is sorted by character code, each code being represented by its decimal, octal, and hexadecimal equivalent.
- Each row of the table gives the ASCII and EBCDIC meaning of the character code, the ASCII↔EBCDIC conversion code, and the Hollerith representation (punched card code) for the ASCII character.

The following examples describe several ways of using the table:

Example 1: Suppose you want to determine the ASCII code for the $ character. Scan down the ASCII graphic column until you locate $, then look left on that row to find the character code – 36 (dec), 044 (oct), and 24 (hex). This is the code used by an ASCII device (terminal, printer, computer, etc.) to represent the $ character. Its Hollerith punched card code is 11-3-8.

Example 2: The character code 5B (hex) is the EBCDIC code for what character? Also, when 5B is converted to ASCII (for example, by FCOPY with the EBCDICIN option), what is the octal character code? First, locate 5B in the hex character code column and move right on that row to the EBCDIC graphic which is $. The next column to the right gives the conversion to ASCII, 044. As a check, find 044 (oct) in the character code column, look right to the ASCII graphic column and note that $ converted to EBCDIC is 133 (oct) which equals 5B (hex).

| CHAR CODE | | | ASCII | | | EBCDIC | |
|---|---|---|---|---|---|---|---|
| Dec | Oct | Hex | Cntl/ Gph | to EBCDIC (Oct) | Hollerith | Cntl/ Gph | to ASCII (Oct) |
| 0 | 000 | 00 | NUL | 000 | 12-0-1-8-9 | NUL | 000 |
| 1 | 001 | 01 | SOH | 001 | 12-1-9 | SOH | 001 |
| 2 | 002 | 02 | STX | 002 | 12-2-9 | STX | 002 |
| 3 | 003 | 03 | ETX | 003 | 12-3-9 | ETX | 003 |
| 4 | 004 | 04 | EOT | 067 | 7-9 | PF | 234 |
| 5 | 005 | 05 | ENQ | 055 | 0-5-8-9 | HT | 011 |
| 6 | 006 | 06 | ACK | 056 | 0-6-8-9 | LC | 206 |
| 7 | 007 | 07 | BEL | 057 | 0-7-8-9 | DEL | 177 |
| 8 | 010 | 08 | BS | 026 | 11-6-9 | | 227 |
| 9 | 011 | 09 | HT | 005 | 12-5-9 | | 215 |
| 10 | 012 | 0A | LF | 045 | 0-5-9 | SMM | 216 |
| 11 | 013 | 0B | VT | 013 | 12-3-8-9 | VT | 013 |
| 12 | 014 | 0C | FF | 014 | 12-4-8-9 | FF | 014 |
| 13 | 015 | 0D | CR | 015 | 12-5-8-9 | CR | 015 |
| 14 | 016 | 0E | SO | 016 | 12-6-8-9 | SO | 016 |
| 15 | 017 | 0F | SI | 017 | 12-7-8-9 | SI | 017 |
| 16 | 020 | 10 | DLE | 020 | 12-11-1-8-9 | DLE | 020 |
| 17 | 021 | 11 | DC1 | 021 | 11-1-9 | DC1 | 021 |
| 18 | 022 | 12 | DC2 | 022 | 11-2-9 | DC2 | 022 |
| 19 | 023 | 13 | DC3 | 023 | 11-3-9 | TM | 023 |
| 20 | 024 | 14 | DC4 | 074 | 4-8-9 | RES | 235 |
| 21 | 025 | 15 | NAK | 075 | 5-8-9 | NL | 205 |
| 22 | 026 | 16 | SYN | 062 | 2-9 | BS | 010 |
| 23 | 027 | 17 | ETB | 046 | 0-6-9 | IL | 207 |
| 24 | 030 | 18 | CAN | 030 | 11-8-9 | CAN | 030 |
| 25 | 031 | 19 | EM | 031 | 11-1-8-9 | EM | 031 |
| 26 | 032 | 1A | SUB | 077 | 7-8-9 | CC | 222 |
| 27 | 033 | 1B | ESC | 047 | 0-7-9 | CU1 | 217 |
| 28 | 034 | 1C | FS | 034 | 11-4-8-9 | IFS | 034 |
| 29 | 035 | 1D | GS | 035 | 11-5-8-9 | IGS | 035 |
| 30 | 036 | 1E | RS | 036 | 11-6-8-9 | IRS | 036 |
| 31 | 037 | 1F | US | 037 | 11-7-8-9 | IUS | 037 |
| 32 | 040 | 20 | SP | 100 | Blank | DS | 200 |
| 33 | 041 | 21 | ! | 117 | 12-7-8 | SOS | 201 |
| 34 | 042 | 22 | " | 177 | 7-8 | FS | 202 |
| 35 | 043 | 23 | # | 173 | 3-8 | | 203 |
| 36 | 044 | 24 | $ | 133 | 11-3-8 | BYP | 204 |
| 37 | 045 | 25 | % | 154 | 0-4-8 | LF | 012 |
| 38 | 046 | 26 | & | 120 | 12 | ETB | 027 |
| 39 | 047 | 27 | ' | 175 | 5-8 | ESC | 033 |
| 40 | 050 | 28 | ( | 115 | 12-5-8 | | 210 |
| 41 | 051 | 29 | ) | 135 | 11-5-8 | | 211 |
| 42 | 052 | 2A | * | 134 | 11-4-8 | SM | 212 |
| 43 | 053 | 2B | + | 116 | 12-6-8 | CU2 | 213 |
| 44 | 054 | 2C | , | 153 | 0-3-8 | | 214 |
| 45 | 055 | 2D | – | 140 | 11 | ENQ | 005 |
| 46 | 056 | 2E | . | 113 | 12-3-8 | ACK | 006 |
| 47 | 057 | 2F | / | 141 | 0-1 | BEL | 007 |

| CHAR CODE | | | ASCII | | | EBCDIC | |
|---|---|---|---|---|---|---|---|
| Dec | Oct | Hex | Cntl/ Gph | to EBCDIC (Oct) | Hollerith | Cntl/ Gph | to ASCII (Oct) |
| 48 | 060 | 30 | 0 | 360 | 0 | | 220 |
| 49 | 061 | 31 | 1 | 361 | 1 | | 221 |
| 50 | 062 | 32 | 2 | 362 | 2 | SYN | 026 |
| 51 | 063 | 33 | 3 | 363 | 3 | | 223 |
| 52 | 064 | 34 | 4 | 364 | 4 | PN | 224 |
| 53 | 065 | 35 | 5 | 365 | 5 | RS | 225 |
| 54 | 066 | 36 | 6 | 366 | 6 | UC | 226 |
| 55 | 067 | 37 | 7 | 367 | 7 | EOT | 004 |
| 56 | 070 | 38 | 8 | 370 | 8 | | 230 |
| 57 | 071 | 39 | 9 | 371 | 9 | | 231 |
| 58 | 072 | 3A | : | 172 | 2-8 | | 232 |
| 59 | 073 | 3B | ; | 136 | 11-6-8 | CU3 | 233 |
| 60 | 074 | 3C | < | 114 | 12-4-8 | DC4 | 024 |
| 61 | 075 | 3D | = | 176 | 6-8 | NAK | 025 |
| 62 | 076 | 3E | > | 156 | 0-6-8 | | 236 |
| 63 | 077 | 3F | ? | 157 | 0-7-8 | SUB | 032 |
| 64 | 100 | 40 | @ | 174 | 4-8 | SP | 040 |
| 65 | 101 | 41 | A | 301 | 12-1 | | 240 |
| 66 | 102 | 42 | B | 302 | 12-2 | | 241 |
| 67 | 103 | 43 | C | 303 | 12-3 | | 242 |
| 68 | 104 | 44 | D | 304 | 12-4 | | 243 |
| 69 | 105 | 45 | E | 305 | 12-5 | | 244 |
| 70 | 106 | 46 | F | 306 | 12-6 | | 245 |
| 71 | 107 | 47 | G | 307 | 12-7 | | 246 |
| 72 | 110 | 48 | H | 310 | 12-8 | | 247 |
| 73 | 111 | 49 | I | 311 | 12-9 | | 250 |
| 74 | 112 | 4A | J | 321 | 11-1 | ¢ | 133 |
| 75 | 113 | 4B | K | 322 | 11-2 | . | 056 |
| 76 | 114 | 4C | L | 323 | 11-3 | < | 074 |
| 77 | 115 | 4D | M | 324 | 11-4 | ( | 050 |
| 78 | 116 | 4E | N | 325 | 11-5 | + | 053 |
| 79 | 117 | 4F | O | 326 | 11-6 | \| | 041 |
| 80 | 120 | 50 | P | 327 | 11-7 | & | 046 |
| 81 | 121 | 51 | Q | 330 | 11-8 | | 251 |
| 82 | 122 | 52 | R | 331 | 11-9 | | 252 |
| 83 | 123 | 53 | S | 342 | 0-2 | | 253 |
| 84 | 124 | 54 | T | 343 | 0-3 | | 254 |
| 85 | 125 | 55 | U | 344 | 0-4 | | 255 |
| 86 | 126 | 56 | V | 345 | 0-5 | | 256 |
| 87 | 127 | 57 | W | 346 | 0-6 | | 257 |
| 88 | 130 | 58 | X | 347 | 0-7 | | 260 |
| 89 | 131 | 59 | Y | 350 | 0-8 | | 261 |
| 90 | 132 | 5A | Z | 351 | 0-9 | ! | 135 |
| 91 | 133 | 5B | [ | 112 | 12-2-8 | $ | 044 |
| 92 | 134 | 5C | \ | 340 | 0-2-8 | * | 052 |
| 93 | 135 | 5D | ] | 132 | 11-2-8 | ) | 051 |
| 94 | 136 | 5E | ^ | 137 | 11-7-8 | ; | 073 |
| 95 | 137 | 5F | _ | 155 | 0-5-8 | ¬ | 136 |

## Table C-1. ASCII, EBCDIC, and Hollerith Codes (Continued)

| CHAR CODE | | | ASCII | | | EBCDIC | |
|---|---|---|---|---|---|---|---|
| Dec | Oct | Hex | Cntl/ Gph | to EBCDIC (Oct) | Hollerith | Cntl/ Gph | to ASCII (Oct) |
| 96 | 140 | 60 | ` | 171 | 1-8 | - | 055 |
| 97 | 141 | 61 | a | 201 | 12-0-1 | / | 057 |
| 98 | 142 | 62 | b | 202 | 12-0-2 | | 262 |
| 99 | 143 | 63 | c | 203 | 12-0-3 | | 263 |
| 100 | 144 | 64 | d | 204 | 12-0-4 | | 264 |
| 101 | 145 | 65 | e | 205 | 12-0-5 | | 265 |
| 102 | 146 | 66 | f | 206 | 12-0-6 | | 266 |
| 103 | 147 | 67 | g | 207 | 12-0-7 | | 267 |
| 104 | 150 | 68 | h | 210 | 12-0-8 | | 270 |
| 105 | 151 | 69 | i | 211 | 12-0-9 | | 271 |
| 106 | 152 | 6A | j | 221 | 12-11-1 | ¦ | 174 |
| 107 | 153 | 6B | k | 222 | 12-11-2 | , | 054 |
| 108 | 154 | 6C | l | 223 | 12-11-3 | % | 045 |
| 109 | 155 | 6D | m | 224 | 12-11-4 | _ | 137 |
| 110 | 156 | 6E | n | 225 | 12-11-5 | > | 076 |
| 111 | 157 | 6F | o | 226 | 12-11-6 | ? | 077 |
| 112 | 160 | 70 | p | 227 | 12-11-7 | | 272 |
| 113 | 161 | 71 | q | 230 | 12-11-8 | | 273 |
| 114 | 162 | 72 | r | 231 | 12-11-9 | | 274 |
| 115 | 163 | 73 | s | 242 | 11-0-2 | | 275 |
| 116 | 164 | 74 | t | 243 | 11-0-3 | | 276 |
| 117 | 165 | 75 | u | 244 | 11-0-4 | | 277 |
| 118 | 166 | 76 | v | 245 | 11-0-5 | | 300 |
| 119 | 167 | 77 | w | 246 | 11-0-6 | | 301 |
| 120 | 170 | 78 | x | 247 | 11-0-7 | | 302 |
| 121 | 171 | 79 | y | 250 | 11-0-8 | ` | 140 |
| 122 | 172 | 7A | z | 251 | 11-0-9 | : | 072 |
| 123 | 173 | 7B | { | 300 | 12-0 | # | 043 |
| 124 | 174 | 7C | ¦ | 152 | 12-11 | @ | 100 |
| 125 | 175 | 7D | } | 320 | 11-0 | ' | 047 |
| 126 | 176 | 7E | ~ | 241 | 11-0-1 | = | 075 |
| 127 | 177 | 7F | DEL | 007 | 12-7-9 | " | 042 |
| 128 | 200 | 80 | | 040 | 11-0-1-8-9 | | 303 |
| 129 | 201 | 81 | | 041 | 0-1-9 | a | 141 |
| 130 | 202 | 82 | | 042 | 0-2-9 | b | 142 |
| 131 | 203 | 83 | | 043 | 0-3-9 | c | 143 |
| 132 | 204 | 84 | | 044 | 0-4-9 | d | 144 |
| 133 | 205 | 85 | | 025 | 11-5-9 | e | 145 |
| 134 | 206 | 86 | | 006 | 12-6-9 | f | 146 |
| 135 | 207 | 87 | | 027 | 11-7-9 | g | 147 |
| 136 | 210 | 88 | | 050 | 0-8-9 | h | 150 |
| 137 | 211 | 89 | | 051 | 0-1-8-9 | i | 151 |
| 138 | 212 | 8A | | 052 | 0-2-8-9 | | 304 |
| 139 | 213 | 8B | | 053 | 0-3-8-9 | | 305 |
| 140 | 214 | 8C | | 054 | 0-4-8-9 | | 306 |
| 141 | 215 | 8D | | 011 | 12-1-8-9 | | 307 |
| 142 | 216 | 8E | | 012 | 12-2-8-9 | | 310 |
| 143 | 217 | 8F | | 033 | 11-3-8-9 | | 311 |
| 144 | 220 | 90 | | 060 | 12-11-0-1-8-9 | | 312 |
| 145 | 221 | 91 | | 061 | 1-9 | j | 152 |
| 146 | 222 | 92 | | 032 | 11-2-8-9 | k | 153 |
| 147 | 223 | 93 | | 063 | 3-9 | l | 154 |
| 148 | 224 | 94 | | 064 | 4-9 | m | 155 |
| 149 | 225 | 95 | | 065 | 5-9 | n | 156 |
| 150 | 226 | 96 | | 066 | 6-9 | o | 157 |
| 151 | 227 | 97 | | 010 | 12-8-9 | p | 160 |
| 152 | 230 | 98 | | 070 | 8-9 | q | 161 |
| 153 | 231 | 99 | | 071 | 1-8-9 | r | 162 |
| 154 | 232 | 9A | | 072 | 2-8-9 | | 313 |
| 155 | 233 | 9B | | 073 | 3-8-9 | | 314 |
| 156 | 234 | 9C | | 004 | 12-4-9 | | 315 |
| 157 | 235 | 9D | | 024 | 11-4-9 | | 316 |
| 158 | 236 | 9E | | 076 | 6-8-9 | | 317 |
| 159 | 237 | 9F | | 341 | 11-0-1-9 | | 320 |
| 160 | 240 | A0 | | 101 | 12-0-1-9 | | 321 |
| 161 | 241 | A1 | | 102 | 12-0-2-9 | ~ | 176 |
| 162 | 242 | A2 | | 103 | 12-0-3-9 | s | 163 |
| 163 | 243 | A3 | | 104 | 12-0-4-9 | t | 164 |
| 164 | 244 | A4 | | 105 | 12-0-5-9 | u | 165 |
| 165 | 245 | A5 | | 106 | 12-0-6-9 | v | 166 |
| 166 | 246 | A6 | | 107 | 12-0-7-9 | w | 167 |
| 167 | 247 | A7 | | 110 | 12-0-8-9 | x | 170 |
| 168 | 250 | A8 | | 111 | 12-1-8 | y | 171 |
| 169 | 251 | A9 | | 121 | 12-11-1-9 | z | 172 |
| 170 | 252 | AA | | 122 | 12-11-2-9 | | 322 |
| 171 | 253 | AB | | 123 | 12-11-3-9 | | 323 |
| 172 | 254 | AC | | 124 | 12-11-4-9 | | 324 |
| 173 | 255 | AD | | 125 | 12-11-5-9 | | 325 |
| 174 | 256 | AE | | 126 | 12-11-6-9 | | 326 |
| 175 | 257 | AF | | 127 | 12-11-7-9 | | 327 |

| CHAR CODE | | | ASCII | | | EBCDIC | |
|---|---|---|---|---|---|---|---|
| Dec | Oct | Hex | Cntl/ Gph | to EBCDIC (Oct) | Hollerith | Cntl/ Gph | to ASCII (Oct) |
| 176 | 260 | B0 | | 130 | 12-11-8-9 | | 330 |
| 177 | 261 | B1 | | 131 | 11-1-8 | | 331 |
| 178 | 262 | B2 | | 142 | 11-0-2-9 | | 332 |
| 179 | 263 | B3 | | 143 | 11-0-3-9 | | 333 |
| 180 | 264 | B4 | | 144 | 11-0-4-9 | | 334 |
| 181 | 265 | B5 | | 145 | 11-0-5-9 | | 335 |
| 182 | 266 | B6 | | 146 | 11-0-6-9 | | 336 |
| 183 | 267 | B7 | | 147 | 11-0-7-9 | | 337 |
| 184 | 270 | B8 | | 150 | 11-0-8-9 | | 340 |
| 185 | 271 | B9 | | 151 | 0-1-8 | | 341 |
| 186 | 272 | BA | | 160 | 12-11-0 | | 342 |
| 187 | 273 | BB | | 161 | 12-11-0-1-9 | | 343 |
| 188 | 274 | BC | | 162 | 12-11-0-2-9 | | 344 |
| 189 | 275 | BD | | 163 | 12-11-0-3-9 | | 345 |
| 190 | 276 | BE | | 164 | 12-11-0-4-9 | | 346 |
| 191 | 277 | BF | | 165 | 12-11-0-5-9 | | 347 |
| 192 | 300 | C0 | | 166 | 12-11-0-6-9 | { | 173 |
| 193 | 301 | C1 | | 167 | 12-11-0-7-9 | A | 101 |
| 194 | 302 | C2 | | 170 | 12-11-0-8-9 | B | 102 |
| 195 | 303 | C3 | | 200 | 12-0-1-8 | C | 103 |
| 196 | 304 | C4 | | 212 | 12-0-2-8 | D | 104 |
| 197 | 305 | C5 | | 213 | 12-0-3-8 | E | 105 |
| 198 | 306 | C6 | | 214 | 12-0-4-8 | F | 106 |
| 199 | 307 | C7 | | 215 | 12-0-5-8 | G | 107 |
| 200 | 310 | C8 | | 216 | 12-0-6-8 | H | 110 |
| 201 | 311 | C9 | | 217 | 12-0-7-8 | I | 111 |
| 202 | 312 | CA | | 220 | 12-11-1-8 | | 350 |
| 203 | 313 | CB | | 232 | 12-11-2-8 | | 351 |
| 204 | 314 | CC | | 233 | 12-11-3-8 | ∫ | 352 |
| 205 | 315 | CD | | 234 | 12-11-4-8 | | 353 |
| 206 | 316 | CE | | 235 | 12-11-5-8 | ⊢ | 354 |
| 207 | 317 | CF | | 236 | 12-11-6-8 | | 355 |
| 208 | 320 | D0 | | 237 | 12-11-7-8 | } | 175 |
| 209 | 321 | D1 | | 240 | 11-0-1-8 | J | 112 |
| 210 | 322 | D2 | | 252 | 11-0-2-8 | K | 113 |
| 211 | 323 | D3 | | 253 | 11-0-3-8 | L | 114 |
| 212 | 324 | D4 | | 254 | 11-0-4-8 | M | 115 |
| 213 | 325 | D5 | | 255 | 11-0-5-8 | N | 116 |
| 214 | 326 | D6 | | 256 | 11-0-6-8 | O | 117 |
| 215 | 327 | D7 | | 257 | 11-0-7-8 | P | 120 |
| 216 | 330 | D8 | | 260 | 12-11-0-1-8 | Q | 121 |
| 217 | 331 | D9 | | 261 | 12-11-0-1 | R | 122 |
| 218 | 332 | DA | | 262 | 12-11-0-2 | | 356 |
| 219 | 333 | DB | | 263 | 12-11-0-3 | | 357 |
| 220 | 334 | DC | | 264 | 12-11-0-4 | | 360 |
| 221 | 335 | DD | | 265 | 12-11-0-5 | | 361 |
| 222 | 336 | DE | | 266 | 12-11-0-6 | | 362 |
| 223 | 337 | DF | | 267 | 12-11-0-7 | | 363 |
| 224 | 340 | E0 | | 270 | 12-11-0-8 | \ | 134 |
| 225 | 341 | E1 | | 271 | 12-11-0-9 | | 237 |
| 226 | 342 | E2 | | 272 | 12-11-0-2-8 | S | 123 |
| 227 | 343 | E3 | | 273 | 12-11-0-3-8 | T | 124 |
| 228 | 344 | E4 | | 274 | 12-11-0-4-8 | U | 125 |
| 229 | 345 | E5 | | 275 | 12-11-0-5-8 | V | 126 |
| 230 | 346 | E6 | | 276 | 12-11-0-6-8 | W | 127 |
| 231 | 347 | E7 | | 277 | 12-11-0-7-8 | X | 130 |
| 232 | 350 | E8 | | 312 | 12-0-2-8-9 | Y | 131 |
| 233 | 351 | E9 | | 313 | 12-0-3-8-9 | Z | 132 |
| 234 | 352 | EA | | 314 | 12-0-4-8-9 | | 364 |
| 235 | 353 | EB | | 315 | 12-0-5-8-9 | | 365 |
| 236 | 354 | EC | | 316 | 12-0-6-8-9 | ⊣ | 366 |
| 237 | 355 | ED | | 317 | 12-0-7-8-9 | | 367 |
| 238 | 356 | EE | | 332 | 12-11-2-8-9 | | 370 |
| 239 | 357 | EF | | 333 | 12-11-3-8-9 | | 371 |
| 240 | 360 | F0 | | 334 | 12-11-4-8-9 | 0 | 060 |
| 241 | 361 | F1 | | 335 | 12-11-5-8-9 | 1 | 061 |
| 242 | 362 | F2 | | 336 | 12-11-6-8-9 | 2 | 062 |
| 243 | 363 | F3 | | 337 | 12-11-7-8-9 | 3 | 063 |
| 244 | 364 | F4 | | 352 | 11-0-2-8-9 | 4 | 064 |
| 245 | 365 | F5 | | 353 | 11-0-3-8-9 | 5 | 065 |
| 246 | 366 | F6 | | 354 | 11-0-4-8-9 | 6 | 066 |
| 247 | 367 | F7 | | 355 | 11-0-5-8-9 | 7 | 067 |
| 248 | 370 | F8 | | 356 | 11-0-6-8-9 | 8 | 070 |
| 249 | 371 | F9 | | 357 | 11-0-7-8-9 | 9 | 071 |
| 250 | 372 | FA | | 372 | 12-11-0-2-8-9 | \| | 372 |
| 251 | 373 | FB | | 373 | 12-11-0-3-8-9 | | 373 |
| 252 | 374 | FC | | 374 | 12-11-0-4-8-9 | | 374 |
| 253 | 375 | FD | | 375 | 12-11-0-5-8-9 | | 375 |
| 254 | 376 | FE | | 376 | 12-11-0-6-8-9 | | 376 |
| 255 | 377 | FF | | 377 | 12-11-0-7-8-9 | EO | 377 |

# SUMMARY OF EMULATOR COMMANDS

APPENDIX D

This appendix summarizes the ten Emulator commands. For each command, the following information is presented:

- The command mnemonic
- The allowable parameters
- A brief description of the function of the command
- A paragraph number indicating where the command is described in this manual

In the presentation of the parameters, brackets [ ] indicate that the particular item is optional and braces { } indicate that one of the enclosed items is to be chosen. Uppercase items, punctuation, and parentheses must be used as shown in the parameter format. Lowercase items are variables which you must replace with the desired file reference, password, value, etc.

| Command Mnemonic | Parameters | Command Function | Text Reference |
|---|---|---|---|
| #RJLINE | {2780 / 3780}<br>[;LINECODE={ASCII / EBCDIC}]<br>[;CONNECT={DIAL[,"*phone number*"] / ANSWER}]<br>[;MAXRPB=*blocking factor*]<br>[;ID=*terminal identification*]<br>[;RIN=*rin*[,*password*]]<br>[;CHNL3=x]<br>[;XEND]<br>[;TRACE=[[ALL][,[*mask*][,[*numentries*][,WRAP]]]]]<br>[;DEV=n] [;LOCK={YES / NO}] [;PRI={HIGH / NORMAL}] | Defines the communications link. | ¶ 3-1 |
| #RJIN | [*input file reference*]<br>[;REC=[*start rec*][,*end rec*]]<br>[;COMPRESS={YES / NO}]<br>[;XPARENT={YES / NO}]<br>[;TRUNCATE={YES / NO}] | Transmits one input data set to the remote processor. | ¶ 3-7 |

| Command Mnemonic | Parameters | Command Function | Text Reference |
|---|---|---|---|
| #RJIN | [;INCODE= {ASCII / EBCDIC / BINARY}]<br>[;MAXSIZE=*nnn*]<br>[;XEOF] | | |
| #RJOUT | [*output file reference*]<br>[,*count*]<br>[;WAIT=[*m*][,*s*]]<br>[;OUTCODE= {ASCII / EBCDIC / BINARY}]<br>[;OUTSIZE=*nnn*] | Initiates the receipt of routed output data sets and/or unrouted list output data sets from the remote processor. | ¶ 3-17 |
| #RJLIST | [*list file reference*]<br>[, *count*]<br>[;WAIT= [*m*] [ ,s]]<br>[;SOURCE=*source file reference*]<br>[;OUTCODE= {ASCII / EBCDIC / BINARY}]<br>[ ;FORMSMSG=*message*]<br>[;OUTSIZE=*nnn*]<br>[;AUTOPAGE= {YES / NO}] | Initiates the receipt of unrouted list output data sets from the remote processor<br>or<br>Initiates the printing of list output data sets from disc or magnetic tape. | ¶ 3-25 |
| #RJPUNCH | [*punch file reference*]<br>[,count]<br>[;WAIT=[m][,s]]<br>[;SOURCE=*source file reference*]<br>[;OUTCODE= {ASCII / EBCDIC / Binary}]<br>[;OUTSIZE=*nnn*] | Initiates the receipt of unrouted punched output data sets from the remote processor<br>or<br>Initiates the punching of punched output data sets from disc or magnetic tape. | ¶ 3-34 |

| Command Mnemonic | Parameters | Command Function | Text Reference |
|---|---|---|---|
| #RJEOD | (none) | Transmits an EOT control character (end-of-file) to the remote processor. | ¶ 3-43 |
| #RJEND | (none) | Terminates the HP 2780/3780 Emulator subsystem. | ¶ 3-47 |
| #RJIO | #RJIO *message* (batch)<br>#[RJIO] *message* (session) | Initiates transmission of a one-line message to the remote processor and initiates receive mode for the receipt of routed data from the remote processor. RJIO optional only during session and if first character of message is a special character. | ¶ 3-50 |
| #RJINFO | (none) | Initiates a file display printing of the communications line. | ¶ 3-53 |
| #RJDEBUG | (none) | Sets the emulator into the debug mode, allowing the user access to the debugging facilities. | ¶ 3-56 |

# CONFIGURING THE EMULATOR

APPENDIX E

This appendix describes how to configure the MPE Operating System to include the Emulator software.

## INTRODUCTION

First, the Emulator hardware (the HP 30055A Synchronous Single-Line Controller) must be installed as described in the *HP 30055A Synchronous Single-Line Controller Installation and Service Manual.* During hardware configuration, determine the SSLC Device Address (the hardware DRT), the highest Device Address (highest DRT), and the logical device number (ldn) that you want to assign to the SSLC. This information is needed during the configuration dialogue.

The programs that constitute the HP 2780/3780 Emulator are supplied in one of two ways:

- On the MPE Master Installation Tape (MIT) under groups HP 30131 and HP 30130 of the Support Account.
- On two reels of magnetic tape (Maintenance Tapes). One contains CS/3000 (HP 30131); the other, RJE (HP 30130). The programs are in STORE format and must be RESTORED to the Support Account in groups HP 30131 and HP 30130 respectively.

Each group (HP 30131 and HP 30130) contains an unnumbered instruction file named INSTRUCT which can be listed using Editor/3000. INSTRUCT in HP 30131 summarizes Emulator configuration, and lists the necessary MPE program and library changes. INSTRUCT in HP 30130 tells how to prep and save RJE, and how to run a test program which exercises the RJE facility.

The MPE Operating System is modified during Emulator configuration. The changes are made during a SYSDUMP operation and the tape generated is subsequently used to COLDSTART the system. If configuring one or more SSLCs is the only modification to be made during the SYSDUMP, the instructions in this appendix are complete. If additional changes are to be made, you may need to refer to the System Manager/System Supervisor Reference Manual. The step numbers in the following dialogue correspond to the System Modification step numbers given in Section VI of that manual.

Special consideration must be given to the method of communication employed at your facility so that you will know whether to establish full or half duplex mode during the Configurator/User Dialogue.

FULL DUPLEX Mode. Configure the Emulator to operate in full duplex mode if your facility uses one of these:

- A leased line with a four wire point-to-point installation.
- A dial network with two lines (four wire).
- A dial network with Wide Band Service.

In the Configurator/User dialogue, the following prompts are relevant to full duplex configuration:

| | |
|---|---|
| SUB TYPE? 0 or 1 | (Step Number 3.10) |
| TRANSMISSION MODE? 0 | (Step Number 3.27) |

HALF DUPLEX Mode. Configure the Emulator to operate in half duplex mode if your facility uses a dial network with a single-line (two wire) installation.

In the Configuration/User dialogue, the following prompts are relevant to half duplex configuration:

| | |
|---|---|
| SUB TYPE? 0 | (Step Number 3.10) |
| DIAL FACILITY? YES | (Step Number 3.20) |
| ANSWER FACILITY? YES | (Step Number 3.21) |
| AUTOMATIC ANSWER? YES or NO | (Step Number 3.22) |
| TRANSMISSION MODE? 1 | (Step Number 3.27) |

## CONFIGURATION DIALOGUE

In the dialogue, return underlined means you should press the RETURN key in response to the prompt. Also, all responses are underlined to distinguish them from computer prompts. Some responses are shown as YES or NO; you can type Y or N instead.

Log on the Support account, define the output device files, and initiate the System Modification dialogue:

:HELLO FIELD.SUPPORT,HP30131 ← The Support account must have System Manager capability in order to perform a system dump. If it does not, log on under the MANAGER.SYS account.
:FILE T;DEV=TAPE
:FILE L;DEV=LP
:SYSDUMP *T,*L

| Step No. | Dialogue |
|---|---|
| 1 | ANY CHANGES? YES |
| 2 | SYSTEM ID = HP 32002.v.uu.ff.? return |
| 3 | MEMORY SIZE = xxx.? return |
| 3.1 | I/O CONFIGURATION CHANGES? YES |
| 3.2 | LIST I/O DEVICES? YES |

All I/O devices currently configured on the system are listed with the following column headings:

| | |
|---|---|
| LOG DEV | Logical device number |
| DRT # | Hardware device address of SSLC |
| UNIT # | Always 0 for SSLC |
| CHAN | Always 0 for SSLC |
| TYPE | Device type, always 18 for SSLC |
| SUBTYPE | Device subtype |
| TERM TYPE | Terminal type |
| REC WIDTH | Record width, decimal words |
| OUTPUT DEV | Device class name or device ldn |
| MODE | J = Accept jobs<br>A = Accept data<br>I = Interactive device<br>D = Duplicative device<br>S = Spooled device |
| DRIVER NAME | CSSBSC0 for SSLC |
| DEVICE CLASSES | Class names assigned to SSLC |

3.3 LIST CS DEVICES? YES

This prompt appears only if a CS device is currently configured in the system. The list of CS devices is printed with the following column headings:

| | |
|---|---|
| LDN | Logical device number |
| PM | Always 0 for SSLC |
| PRT | Protocol |
| LCL MOD | Local mode |
| TC | Transmission code |
| RCV TMOUT | Receive timeout |
| LCL TMOUT | Local timeout |
| CON TMOUT | Connect timeout |
| MODE | O = Dial out<br>I = Manual answer<br>A = Automatic answer<br>D = Dual speed<br>H = Half speed<br>C = Speed changeable |

| Step No. | Dialogue |
|---|---|

| | |
|---|---|
| TRANSMIT SPEED | Transmission speed (char per sec) |
| TM | Transmission mode |
| BUFFER SIZE | Default buffer capacity, in words |
| DC | Driver changeable or not changeable |
| DRIVER OPTION | Driver options |

3.4 HIGHEST DRT = xx.?

xx denotes the current highest hardware device number that can be assigned. Enter return if the number displayed is large enough to include the SSLC (or SSLCs) you are adding. Otherwise, enter a higher DRT number.

3.5 LOGICAL DEVICE #? 13

Enter the logical device number of the particular SSLC board being configured. This prompt is repeated later in the modification dialogue so that you can return to this point to configure more than one SSLC.

3.6 DRT #? 18

To add an SSLC, enter its hardware device address.

3.7 UNIT #? 0 (Always 0 for SSLC)

3.8 CHANNEL #? 0 (SSLC does not use selector channel)

3.9 TYPE? 18 (Always 18 for SSLC)

3.10 SUB TYPE?

0 = Point-to-point switched line with modem (telephone dial network). A zero subtype enables "data set ready" causing you to receive the dial and answer prompts in Steps 3.20 through 3.22.

1 = Point-to-point nonswitched line with modem (leased line, 4-wire). When subtype equals 1, Steps 3.20 through 3.22 are skipped.

3.14 PROTOCOL? 1 (Binary Synchronous Communications)

3.15 LOCAL MODE?

1 = Local is a primary contention station
2 = Local is a secondary contention station

To resolve the contention problem in point-to-point operations, each station is assigned a priority (primary or secondary). Because the secondary station can gain control of the line for a transmission only when the line is left free by the primary station, the Emulator is usually configured as a primary station.

Local mode determines the amount of time a local station will wait in response to a line bid; primary station timeout is two seconds and secondary is three seconds. If a response from the remote system is not received within the allowed time (two seconds for primary or three seconds for secondary), the line bid is re-transmitted until the number of retries permitted by the Emulator is exhausted.

3.16 TRANSMISSION CODE?

1 = Automatic code sensing of ASCII and EBCDIC if initially receiving; ASCII if initially sending.
2 = ASCII
3 = EBCDIC

Select the most frequently used method of transmission because your response establishes the configuration default. Users not transmitting by the default method can use the LINECODE= parameter of the #RJLINE command to override this specification.

| Step No. | Dialogue |
|---|---|
| 3.17 | RECEIVE TIMEOUT?<br>A number, 0 to 32000. Default is 20 seconds.<br><br>Your response establishes the number of seconds the Emulator will wait to receive communication before terminating the read mode. A zero response disables the timeout feature.<br><br>This timeout is set when an ACK is sent to the remote system. Return communication is expected from the remote system in the form of text, EOT, or TTD.<br><br>If a timeout occurs, a batch job terminates and the Emulator displays an error message on the system console.<br><br>In session mode, the Receive Timeout is disabled. |
| 3.18 | LOCAL TIMEOUT?<br>A number, 0 to 32000. Default is 60 seconds.<br><br>Your response establishes the number of seconds a connected local station will wait to transmit or receive before disconnecting. A zero response disables the timeout feature.<br><br>This timeout promotes efficient line use by timing between requests to the SSLC driver.<br><br>In session mode, the Local Timeout is disabled. |
| 3.19 | CONNECT TIMEOUT?<br>A number, 0 to 32000. Default is 900 seconds.<br><br>Your response establishes the number of seconds the local station will wait after one attempt to make a connection to a remote station. A zero response disables the timeout feature.<br><br>The "data set ready" signal must indicate the communications link is established within the allotted time, otherwise, a timeout occurs.<br><br>The Connect Timeout is disabled when a user, working in session mode, includes the optional CONNECT=ANSWER parameter in an #RJLINE command.<br><br>The Connect Timeout is operational when a user specifies CONNECT=ANSWER from a job (in batch mode).<br><br>NOTE<br><br>If the SSLC was configured as subtype 0 in Step 3.10 (it is connected with a switched telephone line through a modem), continue with Step 3.20.<br><br>If the SSLC was configured as subtype 1 in Step 3.10 (it is connected with a non-switched leased line through a modem), skip to Step 3.23. |
| 3.20 | DIAL FACILITY?<br>YES Calls can be dialed from the local station.<br>NO Calls cannot be dialed. |
| 3.21 | ANSWER FACILITY?<br>YES Local modem can answer calls, either manually or automatically.<br>NO Calls cannot be answered. After a NO response, the dialogue skips the next step. |
| 3.22 | AUTOMATIC ANSWER?<br>YES Local modem can answer calls automatically.<br>NO Modem cannot answer calls automatically. |

| Step No. | Dialogue |
|---|---|
| 3.23 | DUAL SPEED?<br>YES Local modem is dual speed (European models).<br>NO Local modem is single speed. Dialogue skips to Step 3.25 after a NO response. NO |
| 3.24 | HALF SPEED? (This prompt is relevant only to European models.)<br>YES Modem operates at half speed.<br>NO Modem operates at full speed.<br><br>After you respond to this prompt, the dialogue skips to Step 3.26. |
| 3.25 | SPEED CHANGEABLE? NO (Always NO for SSLC) |
| 3.26 | TRANSMISSION SPEED?<br>Enter the line transmission speed, expressed as characters per second. |

| Baud Rate | Characters per Second |
|---|---|
| 2000 | 250 |
| 2400 | 300 |
| 4800 | 600 |
| 9600 | 1200 |

The transmission speed you specify is ignored for modems that provide internal clocking signals. This allows modems of different speeds to be used without reconfiguring the Emulator.

| | |
|---|---|
| 3.27 | TRANSMISSION MODE?<br>0 = Full duplex<br>1 = Half duplex<br><br>Your response must agree with the host system's configuration and with the characteristics of the communication line. (Refer to the beginning of this appendix for more information about transmission modes.) |
| 3.28 | PREFERRED BUFFER SIZE? 1024<br>Buffer size, in words, not greater than 4096. The RJE Subsystem will override your specification with a suitable buffer size, depending on 2780/3780 emulation. |
| 3.29 | DRIVER CHANGEABLE? NO |
| 3.30 | DRIVER OPTIONS? 0 |
| 3.50 | DRIVER NAME? CSSBSCO |
| 3.51 | CONTROL LENGTH? 0 |
| 3.52 | PHONELIST? NO or YES<br>Phone lists are not used by the Emulator Subsystem. However, you can supply one number (usually a frequently dialed number) which will be the system default. A NO response causes the next step to be skipped. |
| 3.53 | PHONE NUMBER?<br>Enter a string of numbers and hyphens not more than 20 characters in length. |
| 3.54 | LOCAL ID SEQUENCE? return<br>The Emulator subsystem does not use configured ID sequences. An ID sequence is transmitted only when a user includes the optional ID= parameter in the #RJLINE command. |
| 3.55 | REMOTE ID SEQUENCE? return<br>The Emulator ignores all ID sequences sent by a remote processor. |

| Step No. | Dialogue |
|---|---|
| 3.70 | DEVICE CLASSES? RJLINE (see note)<br>Note: You must specify RJLINE, but you may follow RJLINE with additional names. Names should be meaningful, up to eight alphanumeric characters, and begin with a letter. Separate the names with commas and terminate your list with return. |
| 3.5 | LOGICAL DEVICE #?<br>This prompt repeats the original Step 3.5 prompt. If you plan to add another SSLC, enter the logical device number for the second board to return to the original Step 3.5. If configuration is complete, press return to continue with Step 3.80. |
| 3.80 | MAX # OF OPEN SPOOLFILES = xx.? return |
| 3.81 | LIST I/O DEVICES? YES |
| 3.82 | LIST CS DEVICES? YES |
| 3.83 | CLASS CHANGES? return |
| 3.93 | LIST I/O DEVICES; return |
| 3.94 | ADDITIONAL DRIVER CHANGES? return |
| 4 | SYSTEM TABLE CHANGES? return |
| 5 | MISC CONFIGURATION CHANGES? return |
| 6 | LOGGING CHANGES? return |
| 7 | DISC ALLOCATION CHANGES? return |
| 8 | SCHEDULING CHANGES? return |
| 9 | SEGMENT LIMIT CHANGES? return |
| 10 | SYSTEM PROGRAM CHANGES? YES<br>A YES response assumes this is the first time an SSLC has been configured into the system. If an SSLC already exists, respond NO and skip to Step 11. |
| 10.1 | ENTER PROGRAM NAME, REPLACEMENT FILE NAME? CSSBSC0,P06P131A.HP30131.SUPPORT |
| 11 | SYSTEM SL CHANGES? YES<br>A YES response assumes this is the first time an SSLC has been configured into the system. If an SSLC already exists, respond NO and skip to Step 12. |
| 11.1 | LIST LIBRARY? NO |
| 11.2 | DELETE SEGMENT? NO |
| 11.3 | REPLACE SEGMENT? NO |
| 11.4 | ADD SEGMENT? YES |

| Step No. | Dialogue |
|---|---|
| 11.4.1 | ENTER SEGMENT NAME, USLFILE NAME [,S/C/P]?<br>?COMSYS1,U00U131A.HP30131.SUPPORT,S<br>?COMSYS2,U00U131A.HP30131.SUPPORT,S<br>?COMSYS3,U00U131A.HP30131.SUPPORT,S<br>?COMSYS4,U00U131A.HP30131.SUPPORT,S<br>?COMSYS5,U00U131A.HP30131.SUPPORT,S<br>?COMSYS6,U00U131A.HP30131.SUPPORT,S<br>?COMSYS7,U00U131A.HP30131.SUPPORT,S<br>?CSUTILITY,U01U131A.HP30131.SUPPORT,S<br>?BSCLCM,U03U131A.HP30131.SUPPORT,S<br>?BSCSLCP0,U04U131A.HP30131.SUPPORT,S<br>?DVRSSLC,U05U131A.HP30131.SUPPORT,S<br>?return |
| 12 | ENTER DUMP DATE? return<br>return — Dumps current I/O configuration, MPE, and PUB.SYS only. When this tape is used for COLDSTART, the account structure and all files remain intact.<br>future date — Dumps current I/O configuration, MPE, and the account structure (but no files).<br>current date — Dumps current I/O configuration, MPE, the account structure, and all files modified since the last SYSDUMP.<br>0 — Dumps everything. |
| 12.1 | LIST FILES DUMPED? NO |
| 13 | The System Operator must now use the =REPLY command to assign the magnetic tape device on which you have arranged for a fresh tape reel to be mounted. |

After the SYSDUMP is complete, the tape produced should be used to COLDSTART the system. During COLDSTART, the old I/O device configuration is replaced with the new one from your SYSDUMP tape.

# SUMMARY OF ERROR MESSAGES

APPENDIX F

The more frequent of the HP 2780/3780 Emulator error messages are summarized in Table F-1. Section V of this manual contains a more complete and detailed description of error messages.

Table F-1. Emulator Error Messages

| MESSAGE | MEANING |
|---|---|
| ****COMMAND FILE ERROR: x,yy<br>****INPUT FILE ERROR: x,yy<br>****LIST FILE ERROR: x,yy<br>****PUNCH FILE ERROR: x,yy<br>****OUT FILE ERROR: x,yy<br>****SOURCE FILE ERROR: x,yy | The specified file could not be opened, closed, read from, written into, or (in the case of the LIST, PUNCH, or OUT files) was not large enough.<br><br>x: 0 = FOPEN error.<br>1 = FCLOSE error.<br>2 = FREAD or FWRITE error.<br>3 = Ran out of file space.<br><br>yy: FCHECK error code |
| ****COMMAND ERROR: 0 | Invalid command (the command was totally unrecognizable). |
| ****COMMAND ERROR: 1 | Command too long. |
| ****COMMAND ERROR: 2 | Too many parameters. |
| ****CS ERR 0,11 or 0,13 | Line is in use or unavailable for some reason. |
| ****CS ERR 0, 12 | Erroneous x in :FILE RJLINE;DEV=x file equation<br>or<br>no :FILE RJLINE;DEV=x file equation was issued and there is no Synchronous Single Line Controller (SSLC) with the device class name RJLINE. |
| ****CS ERR 1,57 or 2,57 | You (or the computer operator) entered NO in response to the DIAL REQD message. |
| ****CS ERR 1,101 or 2,101 | The Synchronous Single Line Controller (SSLC) failed to respond to an I/O instruction. |
| ****CS ERR 1,102 or 2,102 or 1,105 or 2,205 | A parity error occurred on the SIO-bus. |
| ****CS ERR 1,103 or 2,103 | The Data Set Ready (CC) signal from the MODEM has dropped (changed from "set" to "clear"). |
| ****CS ERR 1,153 or 2,153 or 2,203 or 2,204 | Line bid refusal. The remote processor responded with a NAK or EOT when the Emulator tried to initiate a transmission. |
| ****CS ERR 1,158 or 2,158 | A DLE-EOT sequence was received. |
| ****CS ERR 1,201 or 2,201 | Operator abort. The computer operator of your HP 3000 Computer System entered an =ABORTIO command for the Synchronous Single Line CONTROLLER. |
| ****CS ERR 1,209 | Receive Text time-out. No data was received within the allowed time (20 seconds). |

| MESSAGE | MEANING |
|---|---|
| ****CS ERR 1,217 | Line Bid time-out. The remote processor did not bid for the line within the allowed time (3 minutes or the specified WAIT= time). |
| ****CS ERR 2,205 | An ENQ was received in response to a line bid. |
| ****CS ERR 2,206 | Transmit mode only.<br>An RVI was received when a positive or negative acknowledgement (ACK0, ACK1, WACK, or NAK) was expected. |
| ****CS ERR 2,207,2 | An irrecoverable parity error occurred on the transmission line. |
| ****CS ERR 2,207,4 | Line bid time-out. The remote processor did not respond within 15 retries (48 seconds) when the Emulator tried to initiate a transmission.<br><br>Write Continue time-out. The remote processor did not respond to a transmission block within 15 retries (48 seconds). |
| ****CS ERR 2, 210 | Transmit mode only.<br>An EOT was received when a positive or negative acknowledgement (ACK0, ACK1, WACK, or NAK) was expected. |
| ****LINE ERROR: 0 | You tried to perform remote I/O without having established the line (you forgot to issue an #RJLINE command). |
| ****PROCEDURE ERROR: xx | Either the Emulator could not find the specified procedure or something was wrong with the procedure.<br><br>xx: Loader error code. This code is described in the Loader Errors table under MPE Diagnostic Messages in the MPE Intrinsics Reference Manual. |
| ****RIN ERROR: 0 | You used an invalid resource identification number (RIN) or an invalid RIN password. |
| ****ROUTING ERROR: 0 | Routed list output data set was received when it was not expected. |
| ****ROUTING ERROR: 1 | Routed punched output data set was received when it was not expected. |
| ****SYNTAX ERROR: 0,x | Invalid keyword. x specifies erroneous parameter. |
| ****SYNTAX ERROR; 1,x | Duplicate keyword. x specifies erroneous parameter. |
| ****SYNTAX ERROR: 2,x | Invalid command construction. x specifies erroneous parameter. |
| ****SYNTAX ERROR: 3,x | Invalid parameter. x specifies erroneous parameter. |
| ****SYNTAX ERROR: 4,x | Invalid numeric parameter. x specifies erroneous parameter. |
| ****SYNTAX ERROR: 5,x | Required parameter missing. x specifies missing parameter. |

# OUTPUT PROCEDURE

APPENDIX G

Figure G-1 shows an example of a procedure used on output-type commands such as #RJPUNCH. This is a generalized version intended to show the structure used for an output procedure. Based upon this structure, you ean write your own procedure as required by your particular application. As shown by a COMMENT in the example, you would enter your own code to process data at line 26.

The procedure is privileged and as such requires the PM capability for Account, Group, and User.

> Note: When this or any procedure attempts to expand the locked stack, a System Failure 150 results. To prevent this, you can use the LOCK= NO parameter in the #RJLINE command to allow for stack expansion or the RJE Emulator can be prepared (:PREP or :PREPRUN) with a larger STACK= parameter (refer to J00J130C).

The procedure takes each received block and writes it into RTNDATA. The Emulator provides the user with ten words of global storage from DB+0 to DB+9: this storage area is initialized to zeros. You should:

- Compile the procedure.
- Put USL into the appropriate SL using SEGMENTER.

```
00001000  00000 0    $CONTROL SUBPROGRAM,PRIVILEGED,SEGMENT=PUNCHPROC
00002000  00000 0    BEGIN
00003000  00000 1
00004000  00000 1    INTRINSIC FOPEN,FWRITE,FCLOSE,FCHECK,QUIT;
00005000  00000 1
00006000  00000 1    PROCEDURE PUNCHPROC(TARGET,COUNT);
00007000  00000 1      VALUE COUNT;
00008000  00000 1      INTEGER COUNT;
00009000  00000 1      ARRAY TARGET;
00010000  00000 1      OPTION PRIVILEGED,UNCALLABLE;
00011000  00000 1    BEGIN
00012000  00000 2      LOGICAL FILEOPEN = DB+0;
00013000  00000 2      INTEGER FILENUM  = DB+1;
00014000  00000 2      INTEGER N;
00015000  00000 2      BYTE ARRAY FILENAME(0:7);
00016000  00000 2
00017000  00000 2      IF NOT FILEOPEN THEN
00018000  00007 2      BEGIN
00019000  00007 3        MOVE FILENAME := "PRTDATA ";
00020000  00021 3        FILENUM := FOPEN(FILENAME,0,1);
00021000  00031 3        IF <> THEN GO ERROR;
00022000  00032 3        FILEOPEN := TRUE;
00023000  00034 3      END;
00024000  00034 2      IF COUNT <> 0 THEN
00025000  00037 2      BEGIN
00026000  00037 3        COMMENT USER CODE TO PROCESS DATA;
00027000  00037 3        FWRITE(FILENUM,TARGET,COUNT,0);
00028000  00044 3        IF <> THEN GO ERROR;
00029000  00045 3      END ELSE
00030000  00047 2      BEGIN                  << END-OF-DATA >>
00031000  00047 3        FCLOSE(FILENUM,1,0);
00032000  00053 3        IF <> THEN
00033000  00054 3        BEGIN
00034000  00054 4    ERROR:
00035000  00054 4          FCHECK(FILENUM,N);
00036000  00061 4          QUIT(N);
00037000  00063 4        END;
00038000  00063 3      END;
00039000  00063 2    END;     <<  PUNCHPROC  >>
00040000  00000 1    END.
 PRIMARY DB STORAGE=%000;    SECONDARY DB STORAGE=%00000
 NO. ERRORS=000;             NO. WARNINGS=000
 PROCESSOR TIME=0:00:01;     ELAPSED TIME=0:00:13
```

Figure G-1. Sample Output Procedure

# CS/3000 TRACE FACILITY

APPENDIX H

The CS/3000 Trace Facility can provide a record of the line actions, states and events that occur during Emulator operation. When problems occur during operation, the trace facility provides the means to pinpoint the problem area.

The Emulator's internal procedures for controlling the line are called CS intrinsics. There are four of these procedures, called CREAD, CWRITE, CCONTROL and CCLOSE. CREAD is used when the Emulator issues a read to the line. CWRITE is used when it issues a write. CCONTROL is used to change the characteristics of the line, to reset and to disconnect. CCLOSE releases the line back to the system.

Each CS intrinsic call generates a series of actions, states and events. An action is something that the CS driver performs, and an event is an external happening that requires an action from the driver according to the driver's state.

The trace facility is invoked only at the user's request. Tracing can be invoked for any communication line that the Emulator uses. The trace request is made at the time when an RJLINE line is opened.

Once invoked for a particular communications line, the trace facility continues to record line activity until either the Emulator terminates or the user issues a new RJLINE command. The trace facility keeps track of actions, states and events in the form of trace entries.

The trace entries are grouped into trace records: one trace record for each CREAD, CWRITE, CCONTROL or CCLOSE intrinsic call by the Emulator. The trace records are permanently stored in a user-specified trace file. The contents of a CS/3000 trace file can be formatted and printed through the use of a trace dump utility program (described later in this appendix).

## DEFINING THE TRACE FILE

To define the trace file, enter the following MPE command prior to running the Emulator:

:FILE CSTRACE= *file name*

If no file named CSTRACE exists in your group and no file equation is specified, the Emulator builds one. If CSTRACE already exists, it will be used by overwriting the existing entries. The file must be purged and rebuilt in order to change any file characteristics (for example, *numentries*). Renaming the file(s) and using a file equation is useful for retaining the trace file and for allowing various users to trace in the same group without interference.

### INVOKING THE TRACE FACILITY

To invoke the CS/3000 trace facility, include the following trace parameter in the RJLINE command:

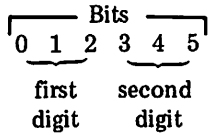
; TRACE $= \left[ [\text{ALL}] \left[ , [mask] \left[ , [numentries] [, \text{WRAP}] \right] \right] \right]$

where

ALL
generates trace records for all CREAD, CWRITE CCONTROL and CCLOSE requests. If ALL is not specified, then trace records are written only when a CREAD, CWRITE, CCONTROL or CCLOSE procedure is completed with a transmission error. If ALL is omitted, only I/O errors are traced. ERROR appears on the trace listing.

*mask*
is an octal integer preceded by a % sign (%nn). It consists of six bit positions (bits 0 to 5) representing the two-digit (nn) mask field. From bits 0 to 5 below, select the type of trace entries required and enter two octal numbers derived from the bit values:

Bit 0 = Generate STN entries (octal 40)
Bit 1 = Generate OPR and EDT entries (octal 20) } First octal digit
Bit 2 = Generate RCT entries (octal 10)




Bit 3 = Generate RTX entries (octal 4)
Bit 4 = Generate SCT entries (octal 2) } Second octal digit
Bit 5 = Generate STX entries (octal 1)

Note: CMP entries are generated automatically. If bits 0 through 5 are set to all zeros, all entry types except STN are generated.

*numentries*
is a decimal integer for the maximum number of trace entries in a trace record, but not greater than 255. If set to zero, there are a maximum of 25 trace entries per trace record.

WRAP
causes trace entries that overflow the trace area (greater than numentries) to overlay the prior trace entries. If WRAP is omitted, overflow trace entries are discarded, and NOWRAP appears on the trace listing.

If only line I/O errors are to be traced, and ALL is not specified, the trace facility deposits in the trace file only those records in which a line I/O error occurred. The types of entries described for *mask* are explained later in this appendix.

If WRAP is specified then entries are deposited in a trace record in a circular pattern. For example, with a maximum of 35 trace entries per trace record, trace entries beyond the 35th will overlay the first, second, third (and so on) trace entries in the record. When this happens, the overlaid trace entries will be missing from the listing; a warning message will appear at the start of the record in the listing stating that the records are missing.

## THE TRACE FILE

Refer to Section II of the MPE Intrinsics Reference Manual for a description of the FOPEN and FCLOSE intrinsic calls. If tracing has been requested, the CS/3000 trace facility issues an FOPEN intrinsic call with the following parameters:

| Parameter | | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | CSTRACE | |
| FOPTIONS | Bits 14,15 | 11 | Old file |
| | Bit 13 | 0 | Binary file |
| | Bits 10,11,12 | 111 | Use actual file designator |
| | Bits 8,9 | 11 | Variable length records |
| | Bit 7 | 0 | No carriage control |
| AOPTIONS | Bits 0 to 7 | 00000000 | None |
| | Bits 12 to 15 | 1111 | Write only; purge old contents |
| | Bit 11 | 0 | No multi-record option |
| | Bit 10 | 0 | Disallow dynamic locking/unlocking |
| | Bits 8,9 | 00 | Exclusive access |
| | Bits 0 to 7 | 00000000 | None |
| BLOCKFACTOR | | 1 | |

If the trace file cannot be opened because it does not exist, then a new file is opened in the system domain. If an error occurs when trying to open the trace file, the particular RJLINE command fails and the trace file printout will be displayed on the user's session/job output device.

When the line is closed, the CS/3000 trace facility issues an FCLOSE intrinsic call with the following parameters:

| Parameter | Value | Meaning |
|---|---|---|
| DISPOSITION | 1 | Save |
| SECCODE | 0 | Unrestricted access |

## TRACE DUMP PROGRAM

There is a CS/3000 trace dump utility program (CSDUMP) in the Systems Account that formats and prints the contents of the trace files generated by the CS/3000 trace facility. The trace dump program requires a trace file and a list file.

## LIST FILE

The formal file designator of the list file is LIST. The list file may be defined as a CRT terminal, a line printer, or a disc file. To define the list file, enter an MPE :FILE command prior to invoking the trace dump program. Some typical examples are:

:FILE LIST;DEV=LP (LP is assumed to be the device class name for one or more line printers)
:FILE LIST;=FILENAME (FILENAME is assumed to be the name of an old temporary or permanent disc file).

If a list file does not exist or is not designated by a :FILE command, and PARM of the RUN command is not a one, the trace dump program employs the user's session/job output device as the list file. If PARM is set to one, then the dump program attempts to open the file LIST as an old job or system file. If this fails because LIST does not exist, then LIST is opened as a new file in the system domain. After the dump program has run, the contents of this file may be used by the HP Text Editor with the /TEXT LIST, UNNUMBERED command.

## INVOKING THE TRACE DUMP PROGRAM

After the trace and list files have been defined, enter the following command:

:RUN CSDUMP.PUB.SYS [;PARM=1]

The trace dump program uses the trace file as input and produces a formatted trace listing on the list file. The format of the trace listing is described in the following text.

## FORMATTED TRACE LISTING

As shown in figure H-1, the trace listing opens with a header message followed by a display box identifying the line being traced. The line information display box contains information on the line state at the time of opening. The third box contains information on the CS request, in this example a CREAD. Any information on missing (lost) entries would appear after the third box. No missing entries are shown in the example.

Each trace entry is sequentially numbered, starting at zero, and the lapsed time in seconds from line opening (12:01 AM in this case) is shown. The body of each entry shows line activity that has happened and is about to happen. Trace entry five contains the actual text received (P08) along with the control characters. Trace entry six shows the portion of the message (the processed text) and the message format word (MFW) set into the buffer. Trace entry seven summarizes the line activity in terms of message activity and errors. The next box shows that the CCLOSE intrinsic call was issued.

Trace entries 8 to 11 are delay sequences sent by the driver to the remote station while waiting for the next Emulator request. Trace entry 12 contains the Emulator request which stops the delay sequence and honors the CCLOSE intrinsic call. Trace entries 13 to 15 show the disconnect sequence and message activity. The line information display contains the same type of information as in the previous display, updated to reflect the preceding received message. When ALL is not specified only errors appear in the line entries.

```
*** CS TRACE DUMP FACILITY ***    TUE, JAN  6, 1976, 12:45 AM

TRACE FILE IS CSTRACE.CS30.DC

LAST OPENED ON TUE, JAN  6, 1976, 12:01 AM

SYSTEM ID=3C.17



*****************************
* BEGIN TRACING FOR DEVICE 32 *
*****************************

*********************************************
*-L-I-N-E---I-N-F-O-R-M-A-T-I-O-N---L-I-S-T-I*
*********************************************
*  LINE NUMBER: 4        LOGICAL DEV. NUMBER: 32    *
*  DEV. TYPE: 18         SUBTYPE: 3    VER=A.00.01  *
*             01234567890123456                     *
*  OPTIONS: 0010000010000001                        *
*  AOPTIONS: 0000000100001100                       *
*  DOPTIONS: 0000000000000600                       *
*  NUMBUFFERS: 1            BUFFSIZE: 512  (WORDS) *
*  POLLREPEAT: 0            POLLDELAY: 0     SECS. *
*  INSPEED: 300             OUTSPEED: 300          *
*  MISCARRAY:         RECEIVE TIMEOUT: 7200  SECS. *
*                       LOCAL TIMEOUT: 7200  SECS. *
*                     CONNECT TIMEOUT: 7200  SECS. *
*                    RESPONSE TIMEOUT: 3     SECS. *
*                  NO. ERROR RETRIES: 7            *
*  DRIVERNAME: CSSBSC0     LINESTATE: UNCONNECTED *
*  CTRACEINFO:    ENTRIES=25      MASK=011111       *
*                 TYPE OF TRACE = ALL, NOWRAP       *
*  POLLIST:       ENTRIES=0         INDEX=0         *
*  PROBELIST:     ENTRIES=0         INDEX=0         *
*  IOLIST:        ENTRIES=0         INDEX=0         *
*  SUBLIST:       STATIONS=0        COMPONENTS=0    *
*  ERRORCODE:   RECOVERABLE=0    IRRECOVERABLE=0    *
*  MSGSENT: 0                 MSGRECV: 0           *
*  RECOVERRORS: 0             IRRECOVERRORS: 0     *
*********************************************

*********************************************************************
* CREAD                          LOGICAL DEVICE=32                  *
* CALLER: SEGMENT=PRG  000       ADDRESS= 000122                    *
* STATE:  LINE STATE=DISCONNECT  OPTIONS= 020201  DOPTIONS= 000000  *
* INPUT:  IN BUF= 000076 LENGTH=-28     STATION #=0   COMPONENT #=0  *
* OUTPUT: TRANSMISSION LOG=-4           STATION #=0   COMPONENT #=0  *
*********************************************************************


0         1.722 CPR  WAIT FOR CONNECTION THEN RECEIVE CONTROL SEQ
                     TIMEOUT= 7200.000
                     IN BUFR= 0.021145 LENGTH=-16

1         6.611 RCT  205.377 377.377


2         6.612 EDT  RECV ENQUIRY      XLOG=0

3         6.616 CPR  SEND CONTROL SEQ THEN RECEIVE TEXT
                     SEND SEQ=ID ACK    TIMEOUT= 7200.000
                     OUT BFR= 0.000000 LENGTH=0
                     IN BUFR= 1.000044 LENGTH=-28

4         6.617 SCT  020.260
                        0

5         6.725 RTX  020.002 120.060 070.040 020.203 026.240 377.000
                           F   0   8
                     377.377
```

Figure H-1. Sample Trace Listing

```
8          6.727 EOT  RECV TEXT          XLOG=-6
                      200.000 120.040 070.040
                              F   0   E

7          6.730 CMP  ERROR CODE=0       LAST RECOVERABLE ERROR CODE= 0
                      #MSG SENT=0        #MSG RECV=1        STATE=TEXT
                      # RECOVERABLE ERR=0        # IRRECOVERABLE ERR=0


**************************************************************************
* CCLOSE                          LOGICAL DEVICE=32                      *
* STATE:  LINE STATE=DISCONNECT   COPTIONS= 020201  DOPTIONS= 000000     *
**************************************************************************

6          8.452 CPR  SEND CONTROL SEQ THEN RECEIVE CONTROL SEQ
                      SEND SEQ=WACK       TIMEOUT= 7200.000
                      IN BUFR= 0.021145 LENGTH=1

9          8.453 SCT  020.073
                        :

10         8.625 SCT  205.377


11         8.665 EOT  RECV ENQUIRY       XLOG=0


12         8.912 CPR  UNCONDITIONAL CLEAR OF ANY CURRENT OPERATION

13         8.914 CPR  SEND CONTROL SEQ THEN DISCONNECT
                      SEND SEQ=EOT        TIMEOUT=     3.000

14         8.914 SCT  020.004


15         8.975 CMP  ERROR CODE=0       LAST RECOVERABLE ERROR CODE= 0
                      #MSG SENT=0        #MSG RECV=1        STATE=DISCONNECT
                      # RECOVERABLE ERR=0        # IRRECOVERABLE ERR=0


******************************
* END OF TRACE FOR DEVICE  32  *
******************************

**********************************************
*-L-I-N-E---1-A-F-O-R-M-A-T-I-O-N---D-I-S-P-L-A-Y*
**********************************************
*  LINE NUMBER: 4        LOGICAL DEV. NUMBER: 32    *
*  DEV. TYPE: 18         SUBTYPE: 3    VER=A.00.01  *
*             012345678901234567                    *
*  COPTIONS: 0010000010000010                       *
*  AOPTIONS: 0000000100001100                       *
*  DOPTIONS: 0000000000000000                       *
*  NUMBUFFERS: 1            BUFFSIZE: 512  (WORDS) *
*  POLLREPEAT: 0            POLLDELAY: 0      SECS. *
*  INSPEED: 300             OUTSPEED: 300          *
*  MISCARRAY:          RECEIVE TIMEOUT: 7200  SECS. *
*                        LOCAL TIMEOUT: 7200  SECS. *
*                      CONNECT TIMEOUT: 7200  SECS. *
*                    RESPONSE TIMEOUT: 3      SECS. *
*                   NO. ERROR RETRIES: 7            *
*  DRIVERNAME: CSSPSC0     LINESTATE: UNCONNECTED   *
*  CTRACEINFO:    ENTRIES=25      MASK=011111       *
*                 TYPE OF TRACE = ALL,NOWRAP        *
*  POLLIST:      ENTRIES=0          INDEX=0          *
*  PRONELIST:    ENTRIES=0          INDEX=0          *
*  IOLIST:       ENTRIES=0          INDEX=0          *
*  SUPLIST:      STATIONS=0         COMPONENTS=0     *
*  ERRORCODE:  RECOVERABLE=0    IRRECOVERABLE=0     *
*  MSGSENT: 0                MSGRECV: 1             *
*  RECOVERRORS: 0            IRRECOVERRORS: 0       *
**********************************************

END OF JOB.
```

Figure H-1. Sample Trace Listing (Continued)

## TRACE LISTING HEADER MESSAGE

At the start of the trace listing is a header message (figure H-2) that tells the date and time-of-day when the listing was printed and the fully-qualified name of the trace file. The meanings of the two remaining items in the header message are:

| Item | Meaning |
|---|---|
| LAST OPENED ON . . . . | This tells you the date and time-of-day when the trace was performed. |
| SYSTEM ID=nn.mm | This tells you the version number (nn) and the fix level (mm) of the MPE/3000 operating system that was being used when the trace was performed. |

```
*** CS TRACE DUMP FACILITY ***     TUE, JAN  6, 1976, 12:45 AM

TRACE FILE IS CSTRACE.CS30.CC

LAST OPENED ON TUE, JAN  6, 1976, 12:01 AM

SYSTEM ID=32.17
```

Figure H-2. Trace Listing Header

## BEGIN TRACING AND LINE INFORMATION MESSAGES

The "BEGIN TRACING . . . ." message appears in the listing when the line to be traced is opened. The message tells you the decimal logical device number of the line (32 in the example in figure H-3). It indicates the line's activities are now being monitored by the trace facility. It is followed by the Line Information Display describing the state of the line when tracing started.

```
*****************************
* BEGIN TRACING FOR DEVICE 32 *
*****************************

*************************************************
*-L-I-N-E---I-N-F-O-R-M-A-T-I-O-N---D-I-S-P-L-A-Y*
*************************************************
*  LINE NUMBER: 4        LOGICAL DEV. NUMBER: 32   *
*  DEV. TYPE: 18         SUBTYPE: 3   VER=A.00.01  *
*             0123456789012345                     *
*  COPTIONS: 0010000010000001                      *
*  AOPTIONS: 0000000100001100                      *
*  BOPTIONS: 0000000000000000                      *
*  NUMBUFFERS: 1           BUFFSIZE: 512  (WORDS)  *
*  POLLREPEAT: 0           POLLDELAY: 0     SECS.  *
*  INSPEED: 300            OUTSPEED: 300           *
*  MISCARRAY:        RECEIVE TIMEOUT: 7200  SECS.  *
*                      LOCAL TIMEOUT: 7200  SECS.  *
*                    CONNECT TIMEOUT: 7200  SECS.  *
*                   RESPONSE TIMEOUT: 3     SECS.  *
*                 NO. ERROR RETRIES: 7             *
*  DRIVERNAME: CSSBSC0    LINESTATE: UNCONNECTED   *
*  CTRACEINFO:   ENTRIES=25      MASK=011111        *
*                TYPE OF TRACE = ALL, NOWRAP        *
*  POLLIST:      ENTRIES=0         INDEX=0          *
*  PRONELIST:    ENTRIES=0         INDEX=0          *
*  IOLIST:       ENTRIES=0         INDEX=0          *
*  SUPLIST:      STATIONS=0        COMPONENTS=0     *
*  ERRORCODE:  RECOVERABLE=0   IRRECOVERABLE=0      *
*  MSGSENT: 0              MSGRECV: 0               *
*  RECOVERRORS: 0          IRRECOVERRORS: 0         *
*************************************************
```

Figure H-3. Begin Tracing and Line Information

## TRACE RECORD HEADER MESSAGE

The trace listing is organized into a series of trace records, each consisting of a series of trace entries. Every trace record pertains to a particular Emulator request (intrinsic call).

A trace record is signified by a trace record header message immediately preceding the trace entries. It specifies the decimal logical device number of the line being traced (32 in the example in figure H-4), the type of intrinsic, the location in the Emulator where the intrinsic was invoked, the line's state and the Emulator's calling parameters.

```
**************************************************************************
* CREAD                              LOGICAL DEVICE=32                   *
* CALLER: SEGMENT=PRG  000           ADDRESS= 000122                     *
* STATE:  LINE STATE=DISCONNECT      OPTIONS= 020201  DOPTIONS= 000000   *
* INPUT:  IN BUFF= 000076 LENGTH=-28      STATION #=0    COMPONENT #=0    *
* OUTPUT: TRANSMISSION LOG=-4             STATION #=0    COMPONENT #=0    *
**************************************************************************
```

Figure H-4. Trace Record Header

## TRACE ENTRY FORMAT

All entries in a trace listing contain a prefix consisting of three fields:

1. An entry number (8 in the example in figure H-5).
2. A "time stamp" in seconds and thousands of seconds (8.852 in the example).
3. An entry-type mnemonic (OPR in the example).

The first entry is numbered zero and successive entries throughout the rest of the listing of this trace are numbered consecutively in ascending order (1, 2, 3 and so on). The "time stamp" makes it possible for you to determine the elapsed time in seconds and milliseconds between one trace entry and another. The mnemonic tells you what type of trace entry you are examining. There are eight types of trace entries, summarized in table H-1 and described in greater detail in following pages of this appendix. The body of each trace entry tells you the pertinent information for the particular activity that has happened or is about to happen.

```
8.852 OPR   SEND CONTROL SEQ THEN RECEIVE CONTROL SEQ
            SEND SEQ=WACK       TIMEOUT= 7200.000
            IN BUFF= 0.021145 LENGTH=1
```

Figure H-5. Trace Entry

## END OF TRACE AND LINE INFORMATION MESSAGES

The "END OF TRACE . . . ." message appears in the listing when the line being traced is closed. The message tells you the decimal logical device number of the line (32 in the example in figure H-6) and indicates that the line's activities are no longer being monitored by the trace facility. It is followed by the Line Information Display, showing the state of the line just before tracing was stopped.

```
********************************
* END OF TRACE FOR DEVICE  32  *
********************************

************************************************
*-L-I-N-E---I-N-F-O-R-M-A-T-I-O-N---D-I-S-P-L-A-Y*
************************************************
*  LINE NUMBER: 4       LOGICAL DEV. NUMBER: 32   *
*  DEV. TYPE: 18        SUBTYPE: 3    VER=A.00.01  *
*            0123456789012345                     *
*  COPTIONS: 0010000010000010                     *
*  AOPTIONS: 0000000100001100                     *
*  DOPTIONS: 0000000000000000                     *
*  NUMBUFFERS: 1          BUFFSIZE: 512  (WORDS)  *
*  POLLREPEAT: 0          POLLDELAY: 0     SECS.  *
*  INSPEED: 300           OUTSPEED: 300           *
*  MISCARRAY:       RECEIVE TIMEOUT: 7200  SECS.  *
*                     LOCAL TIMEOUT: 7200  SECS.  *
*                   CONNECT TIMEOUT: 7200  SECS.  *
*                  RESPONSE TIMEOUT: 3     SECS.  *
*                NO. ERROR RETRIES: 7             *
*  DRIVERNAME: CSSBSC0    LINESTATE: UNCONNECTED  *
*  CTRACEINFO:   ENTRIES=25     MASK=011111       *
*                TYPE OF TRACE = ALL,NOWRAP       *
*  POLLIST:     ENTRIES=0         INDEX=0         *
*  PHONELIST:   ENTRIES=0         INDEX=0         *
*  IDLIST:      ENTRIES=0         INDEX=0         *
*  SUPLIST:     STATIONS=0        COMPONENTS=0    *
*  ERRORCODE:  RECOVERABLE=0   IRRECOVERABLE=0    *
*  MSGSENT: 0              MSGRECV: 1             *
*  RECOVERRORS: 0          IRRECOVERRORS: 0       *
************************************************

END OF JOB.
```

Figure H-6. End of Trace and Line Information

## MISSING ENTRIES MESSAGE

If "MISSING ENTRIES" appears at the start of a trace record in the listing, it means that the record was not large enough to accommodate all of the trace entries and some entries were lost. If WRAP was not specified (NOWRAP), then the missing entries were at the end just before the CMP entry; otherwise they are missing from the beginning where they were overlaid by the trace entries that extended past the end of the record. If the missing entries are crucial:

1. Purge the trace file.
2. Change the "number of entries per record" specification by way of the ;TRACE=,,*numentries* parameter in an #RJLINE command for the line.
3. Change the MASK field to include only the trace entry types that are required for your particular situation.
4. Rerun the Emulator.

## OPR (OPERATION) TRACE ENTRIES

An OPR trace entry is generated each time the physical driver is called upon to perform an operation. An example is shown in figure H-7.

```
3          6.616 OPR  SEND CONTROL SEQ THEN RECEIVE TEXT
                      SEND SEQ=10 ACK   TIMEOUT= 7200.000
                      OUT BFR= 0.000000 LENGTH=0
                      IN BUFR= 1.000044 LENGTH=-28
```

Figure H-7. OPR Trace Entry

| Item | Meaning |
|---|---|
| SEND CONTROL SEQ . . . . SEND SEQ | This item tells you what operation is being performed. this item tells you what BSC control character sequence, if any, is about to be sent to the remote station (ID ACK in the example). |
| TIMEOUT | This item tells you the starting value of the applicable timeout timer. In the example the driver sends a ID ACK to the remote station and then waits for a character response. "TIMEOUT= 30.000" specifies that the response timeout timer will be activated and set to 30 seconds. (The value of this parameter is input in seconds.) |
| OUT BFR=n.mmmmmm | This item specifies the memory bank number (n) and the octal address (m) of the Emulator/CS output buffer. |
| IN BFR=n.mmmmmm | This item specifies the memory bank number (n) and octal address (m) of the Emulator/CS input buffer. |
| LENGTH | For output, this specifies the amount of text to be sent (+ = words; − = bytes). For input, this specifies the maximum amount of text that can be received for this message (+ = words; − = bytes). |

Table H-1. Trace Entry Type Mnemonics

| Mnemonic | Entry Type | Definition |
|---|---|---|
| OPR | Operation | Generated each time the physical driver (a segment of the CS driver) is called upon to perform an operation. The OPR trace entry tells what operation is to be performed. |
| STN | State Transition | Generated each time the driver transfers from one internal state to another. The STN entry is for internal HP use and should be ignored by the user. |
| EDT | Editor | Generated each time a text message or control character sequence is received from the remote station. In the case of a text message, the EDT trace entry shows the first 14 words of the Emulator's buffer; BSC control characters, pad characters, and CRC parity sequences are omitted. In the case of a BSC control character sequence, the EDT trace entry supplies a mnemonic phrase telling what was received. |
| RCT | Receive Control Sequence | Generated each time a BSC control character sequence is received from the remote station. The RCT trace entry shows (in octal) sequentially byte-for-byte exactly what was received. |
| SCT | Send Control Sequence | Generated each time the driver sends a BSC control character sequence to the remote station. The SCT trace entry shows (in octal) sequentially byte-for-byte exactly what was sent. |
| RTX | Receive Text | Generated each time a text message is received from the remote station. The RTX trace entry shows (in octal) sequentially byte-for-byte exactly what was received. |
| STX | Send Text | Generated each time the driver sends a text message to the remote station. The STX entry shows (in octal) sequentially byte-for-byte exactly what was sent. |
| CMP | User Request Completed | Generated each time an Emulator request (a CREAD, CWRITE, CCONTROL or CCLOSE intrinsic call) is completed. The CMP trace entry summarizes the line activity, such as the number of text messages sent and received and the number of errors that have occurred. |

## EDT (EDITOR) TRACE ENTRIES

EDT trace entries are generated to identify received text messages and received BSC control character sequences. An example is shown in figure H-8.

```
0          6.767 EDT  RECV TEXT          XLOG=-6
                      200.000 120.060 070.040
                              P   0   8
```

Figure H-8. EDT Trace Entry

In the case of a received text message, the body of an EDT trace entry shows, in octal, the message format word (MFW) and the first 13 words of the text message. Regardless of how long the text message is, only one EDT trace entry is generated for each received text message. The text message is shown as it appears in the Emulator's buffer. BSC control characters, pad characters, and CRC parity sequences are omitted. XLOG specifies the total number of words or bytes (+ = words; – = bytes) deposited into the Emulator's buffer, consisting of the MFW and received text. The MFW is described at the end of this appendix.

In the case of a received BSC control character sequence, the EDT trace entry includes a mnemonic phrase telling what control character sequence was received. (Appendix I contains a description of the BSC "handshaking" sequence and control characters.) If the control character sequence was accompanied by ar. ID sequence, the ID sequence is shown in octal below the mnemonic phrase. XLOG is normally zero, except in the case when an ID sequence was also received. In the latter case, XLOG specifies the length of the ID sequence ( + = words; – = bytes).

Whenever possible, the trace dump program converts the received octal codes to a character and displays the character beneath its code. The translation from code to character is performed for EBCDIC transmissions as well as ASCII transmissions.

In the example in figure H-8, the body of the EDT trace entry is interpreted as follows:

A text message was received from the remote station. The total number of characters received was six (XLOG= –6), of which two are MFW characters and four are data characters. These characters are:

200.00 = MFW (message format word)
120 = P  060 = 0
070 = 8  040 = Space

## RCT (RECEIVE CONTROL SEQUENCE) TRACE ENTRIES

An RCT trace entry is generated each time a BSC control character sequence is received from the remote station. The body of an RCT trace entry shows you sequentially byte-for-byte exactly what was received. An example is shown in figure H-9.

```
6          1.026 RCT  020.260 377.377
                            0
```

Figure H-9. RCT Trace Entry

Control character sequences are terminated by a trailing pad character (377 octal). When interpreting the body of an RCT trace entry, ignore anything following the 377 code. In the example, an ACKO control character sequence was received. The octal codes are interpreted as follows:

020 = DLE
260 = 0 (060 with parity bit set) } ACKO sequence
377 = Trailing pad character

Whenever possible, the trace dump program converts the octal codes to a character and displays the character beneath its code. The translation from code to character is performed for EBCDIC transmissions as well as ASCII transmissions.

## SCT (SEND CONTROL SEQUENCE) TRACE ENTRIES

An SCT trace entry is generated each time the driver sends a BSC control character to the remote station. The body of an SCT trace entry shows you sequentially byte-for-byte exactly what was sent to the remote station. An example is shown in figure H-10.

```
5          0.956 SCT  205.377
```

Figure H-10. SCT Trace Entry

In the above example, an ENQ control character was sent. The octal codes are interpreted as follows:

205 = ENQ (005 with parity bit set)
377 = Trailing pad character

Whenever possible, the trace dump program converts the octal codes to a character and displays the character beneath its code. The translation from code to character is performed for EBCDIC transmissions as well as ASCII transmissions.

## RTX (RECEIVE TEXT) TRACE ENTRIES

RTX trace entries are generated each time a text message is received from the remote station. The body of an RTX trace entry shows you sequentially byte-for-byte exactly what was received. An example is shown in figure H-11.

```
5          6.725 RTX  020.002 120.060 070.040 020.203 026.240 377.000
                                 P   0   8
                      377.377
```

Figure H-11. RTX Trace Entry

Most text messages are terminated by a two-byte CRC (cylic redundancy check) parity sequence, while a few may use a one-byte LRC (longitudinal redundancy check). This is followed by a trailing PAD character (377 octal). When interpreting the body of an RTX trace entry, ignore anything following the 377 code.

In the example, the octal codes are interpreted as follows:

| | |
|---|---|
| 020 = DLE | 002 = STX |
| 120 = P | 060 = 0 |
| 070 = 8 | 040 = Space |
| 020 = DLE | 203 = ETX (003 with parity bit set) |
| 026 = First CRC byte | 240 = Second CRC byte |
| | 377 = Trailing pad character |

Each RTX trace entry can show a maximum of 32 bytes. If a text message exceeds this length, as many successive RTX trace entries are generated as are necessary.

Whenever possible, the trace dump program converts the octal codes to a character and displays the character beneath its code. The translation from code to character is performed for EBCDIC transmissions as well as ASCII transmissions.

## STX (SEND TEXT) TRACE ENTRIES

STX trace entries are generated each time the driver sends a message to the remote station. The body of an STX trace entry shows you sequentially byte-for-byte exactly what was sent to the remote station. An example is shown in figure H-12.

```
13         1.037 STX  020.002 120.060 070.040 020.203 026.240 377
                                 P   0   8
```

Figure H-12. STX Trace Entry

In the example, the octal codes are interpreted as follows:

| | |
|---|---|
| 020 = DLE | 002 = R |
| 002 = STX | 040 = Space |
| 120 = P | 060 = 0 |
| 070 = 8 | 040 = Space |
| 020 = DLE | 203 = ETX (003 with parity bit set) |
| 026 = First CRC byte | 240 = Second CRC byte |

Each STX trace entry can show a maximum of 32 bytes. If a text message exceeds this length, as many successive STX trace entries are generated as are necessary.

Whenever possible, the trace dump program converts the octal codes to a character and displays the character beneath its code. The translation from code to character is performed for EBCDIC transmissions as well as ASCII transmissions.

## CMP (I/O COMPLETION) TRACE ENTRIES

A CMP trace entry is generated each time an Emulator request (a CREAD, CWRITE, CCONTROL or CCLOSE intrinsic call) is completed. An example is shown in figure H-13.

```
22      3.332 CMP  ERROR CODE=210    LAST RECOVERABLE ERROR CODE= 0
                   #MSG SENT=0       #MSG RECV=0       STATE=CONTROL
                   # RECOVERABLE ERR=0       # IRRECOVERABLE ERR=0
```

Figure H-13. CMP Trace Entry

The meanings of the various items are as follows:

| Item | Meaning |
|---|---|
| ERROR CODE | The error code of the request's most recent recoverable error (see section V of this manual for error codes). |
| LAST RECOVERABLE ERROR CODE | If a recoverable error occurred, this identifies its error code. |
| #MSG SENT | The total number of text messages that have so far been sent for this connection. |
| #MSG RECV | The total number of text messages that have been received so far for this connection. |
| #RECOVERABLE ERR | The total number of recoverable errors that have occurred so far for this connection. |
| #IRRECOVERABLE ERR | The total number of irrecoverable errors that have occurred so far for this connection. |
| STATE | The line state after the completion of the user request. In the example it is in the control state. |

## SSLC DRIVER MESSAGE FORMAT WORD (MFW)

All BSC control characters are deleted from the incoming text stream by the driver. The driver describes the received text's format via the message format word (MFW) that is contained in word zero of the user's buffer. The MFW fields, which have the same meaning as in sent text, are as follows:

| Field | Meaning |
|---|---|
| MFW bit 0 | 0 = The received text was transparent. If the text contained ITB's, then this bit means that at least one intermediate text block was transparent.<br>1 = The received text was non-transparent. |
| MFW bit 1 | 0 = The received text contained no ITB's.<br>1 = There were ITB's. The length of each ITB immediately precedes its block. The length is a 16-bit quantity and always begins on a word boundary. The last ITB is followed by a delimiter word set to all ones. |
| MFW bit 2 | Note: This field is significant only when there are ITB's in non-transparent text.<br><br>0 = Not all ITB's began with an SOH or STX character.<br>1 = Each ITB was begun with an SOH or STX character. |
| MFW bit 3 | 0 = The text block ended with an ETX character.<br>1 = The text block ended with an ETB character.<br><br>Note: Regardless of whether a text block ends in ETB or ETX, its successful transmission or reception will increment its respective "message" counter (MSGSENT/MSGRECV in the CCHECK intrinsic). |

# BINARY SYNCHRONOUS COMMUNICATIONS

APPENDIX I

## HANDSHAKING

With BSC and other line protocols of this type, "handshaking" is the term commonly used to describe the interaction between stations. Typically the following information is exchanged:

- Message available for transmission
- Start of text transmission
- Acknowledgment or rejection of the text
- Detection of errors
- Retransmission after error detection
- End of transmission

A simplified handshaking sequence is summarized in figure I-1. This shows handshaking between a terminal and computer, but it could also be between two terminals or two computers. In this sequence, a terminal tells a computer it has a message to transmit. The computer recognizes the terminal and tells it to proceed with the message. When the computer receives the message, it detects an error in the text and requests a retransmission. The retransmission is error-free and the computer asks the terminal for another message. The terminal does not have anything more to transmit and so informs the computer. If the computer had a message to transmit to the terminal it could now do so. Since it does not, it disconnects from the terminal.

## MESSAGE SEQUENCE

A message exchange is initiated when a location sends a synchronization (SYN) sequence and an enquiry (ENQ) to another location. If the other location can accept a message it acknowledges (ACK) the enquiry. Throughout the handshaking sequence, each acknowledgment is alternately numbered one and zero. When an acknowledgment is not received by the sender then the next one will be out of sequence and an error detected. As shown in figure I-2, ENQ is transmitted to initiate the exchange and the response is ACK 0, even positive acknowledgement. The next step is to synchronize (SYN) the two locations. This is obtained by transmitting two or more SYN characters, followed by the message block.

When the computer looks at the message block it detects an error and transmits a negative acknowledgment (NAK). The terminal then retransmits the message block. This time it is error-free and the computer transmits ACK 1, odd positive acknowledgment. Synchronization is performed at the start of each message block, and the terminal sends SYN SYN followed by the message. The transmission is error-free and the computer responds with ACK 0, even positive acknowledgment. Since the terminal has no more messages to transmit it sends the end of transmission (EOT) character. Unless the computer has something to transmit to the terminal, this completes the exchange and the computer disconnects from the terminal.

The line protocol characters used in the above example represent only a few of the BSC control characters. Table I-1 lists character mnemonics along with their meaning and function.

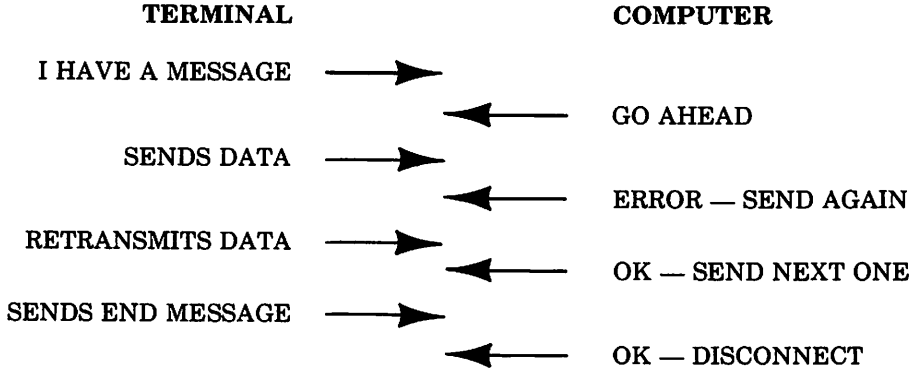



Figure I-1. Simplified Handshaking Sequence

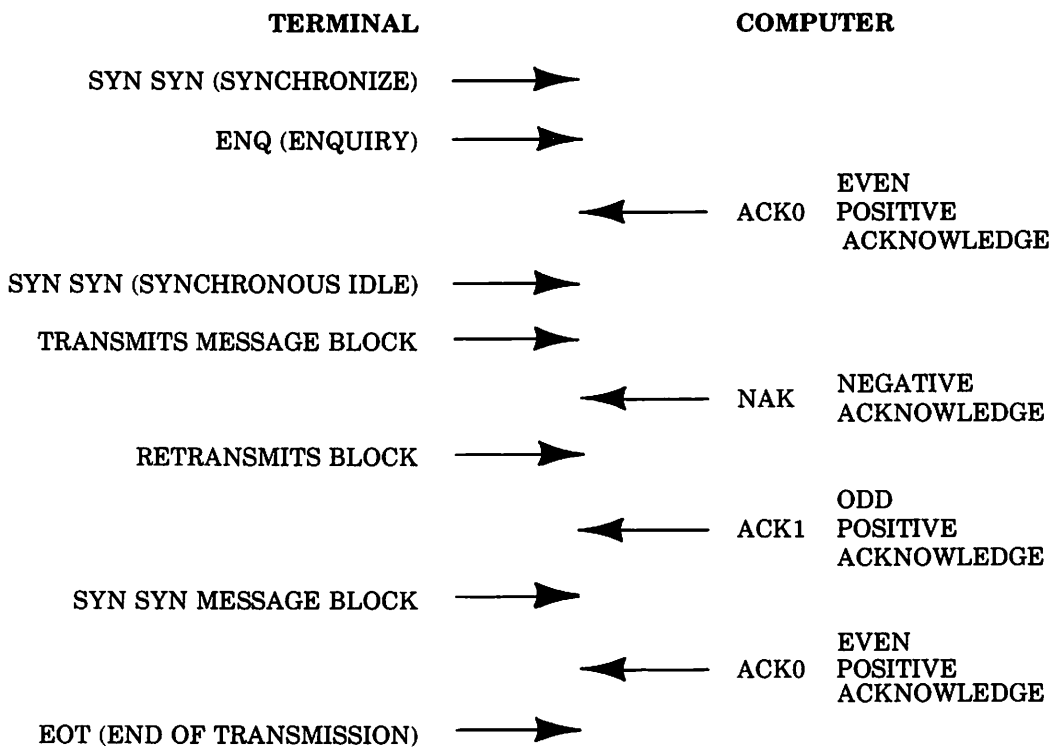



Figure I-2. BSC Handshaking Sequence

Table I-1. BSC Control Characters

| Character | Meaning | Function |
|---|---|---|
| SYN | Synchronous Idle | Establishes and maintains character synchronization prior to the message block. Also used as time fill in the absence of control characters and data. |
| STX | Start of Text | Transmitted before the first control characters. These characters contain information such as the routing and priority of the message. |
| ETB | End of Transmission Block | Indicates the end of the text block starting with STX or SOH. BCC is sent after ETB, requiring the receiver to respond with ACK, NAK or optionally WACK or RVI. |
| US/ITB | End of Intermediate Transmission Block | Divides a message for error checking purposes without the turnaround required by ETB. BCC follows ITB and resets the block-check count to zero. STX or SOH is not required for following text blocks, but STX is required if a header is followed by text. |
| ETX | End of Text | Terminates a block of characters, began with SOH or STX, after the last block in a sequence of blocks. BCC immediately follows ETX, requiring a receiver status reply. |
| EOT | End of Transmission | Concludes transmission, resets all stations to control mode (neither transmitter or receiver). Also a non-transmit response to a poll and an abort signal for a malfunction. |
| ENQ | Enquiry | Bids for the line in a point-to-point and multipoint connection: requests retransmission of last acknowledgment; or preceding block to be ignored. Also indicates completion of poll or selection sequence. |

Table I-1. BSC Control Characters (Continued)

| Character | Meaning | Function |
|---|---|---|
| *ACK | Affirmative Acknowledgment | Previous block accepted and error-free, receiver ready for next block. Also a positive response to selection (multi-point) or line bid (point-to-point). |
| SOH | Start of Heading | Transmitted before the header characters. These contain information such as the routing and priority of the message. |
| NAK | Negative Acknowledgment | Previous block unacceptable and retransmission required. Also a negative response to a selection or line bid. |
| *TTD | Temporary Text Delay | Transmitter not ready to commence transmission but wants to maintain connection. Sent two seconds after message received to avoid three second timeout. Also initiates an abort. |
| *RVI | Reverse Interrupt | Sent to a transmitter by a receiver in place of ACK, indicating the receiver has a high priority message waiting transmission. |
| *WACK | Wait Before Transmit Positive Acknowledgment | Previous block accepted and error-free, but receiver not ready for next block. Will continue to respond with WACK until ready to receive. Also a positive response to a text or heading block selection sequence (multi-point), line bid (point-to-point) or identification line bid sequence (switched network). |
| DLE | Data Link Escape | Prefix for control characters during transparent mode, when control characters have no control meaning unless prefixed by DLE. The two-character sequences ACK, WACK, and RVI have DLE as the first character. |
| DLE EOT | Disconnect Sequence For A Switched Line | Transmitted on a switched line when all message exchanges are complete. Can optionally be transmitted at any time instead of EOT to cause a disconnect. |
| Pad | None | Added before (leading pad) a transmission and after (trailing pad) a transmission to ensure the first and last characters are properly transmitted. |

*Two-character sequence.

# EMULATOR FILE OPTIONS

APPENDIX J

There are seven Emulator files whose characteristics should be known to the Emulator user. These are:

- Command File
- Input File
- List File
- Out File
- Punch File
- Source File
- CS/3000 Trace File

This appendix describes the above files except for the CS/3000 trace file, which is described in appendix H. For those files that can be specified on the :RJE command, the MPE command interpreter issues a :FILE command for each file specified by the user. Consequently, the user's formal file designator effectively overrides the nominal file designator used by the Emulator. All files are closed with a security code of zero (unrestricted access).

FOPEN and FCLOSE intrinsic calls, and the FOPEN parameters AOPTIONS and FOPTIONS, are described in the MPE Intrinsics Reference Manual, section II.

## COMMAND FILE

The command file is opened after the RJE command has been issued. It can be a disc file name; a backward *file reference; or unspecified in which case the default is #STDINX. The command file is opened with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | RJECOM | |
| FOPTIONS | | %2054 if unspecified<br>7 if user-specified | |
| Domain Specification | 14,15 | 00 | Newfile if unspecified |
| | | 11 | Oldfile if user-specified |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 101 | $STDINX if unspecified |
| | | 000 | Formal designator if user-specified |
| Record Format | 8,9 | 00 | Fixed |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 1 | Disallow if unspecified |
| | | 0 | Allow if user-specified |
| AOPTIONS | | 0 | |
| Access Type | 12 to 15 | 0000 | Input only |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 00 | Default |
| Inhibit Buffering | 7 | 0 | No |
| Record Size | | -80 | |
| Device | | Not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

If end-of-file or an error is encountered on the command file, then if the command file was not $STDINX it is closed and $STDINX is opened as the command file, with the same options as describe above for unspecified command file. Additionally, if the current input file is the same device as $STDINX, the input file is closed and the command and input files are accessed as the same file. If the command file was $STDINX, the Emulator terminates.

## INPUT FILE

The default input file is opened after the :RJE command has been issued, and after the command file has been opened. If neither the command file nor the input file is specified by the user, then the command and input files are accessed as the same file ($STDINX). Additionally, any time the input file is opened (either the default input file from the :RJE command, or an input file specified on a #RJIN command) and is the same as the command file, the input file is closed, and both files are accessed as the same file. The default input file is initially opened (following the :RJE command) with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | RJEIN | |
| FOPTIONS | | %2054 if unspecified<br>7 if user-specified | |
| Domain Specification | 14,15 | 00 | Newfile if unspecified |
| | | 11 | Oldfile if user-specified |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 101 | $STDIN if unspecified |
| | | 000 | Formal designator if user-specified |
| Record Format | 8,9 | 00 | Fixed |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 1 | Disallow if unspecified |
| | | 0 | Allow if user-specified |
| AOPTIONS | | 0 | |
| Access Type | 12 to 15 | 0000 | Input only |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 00 | Default |
| Inhibit Buffering | 7 | 0 | No |
| Record Size | | Not passed | |
| Device | | Not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

When the default input file is used (by executing a #RJIN command with no input device or file specification), then if REC= is not specified, the default input file is closed when the #RJIN operation completes, and the default input file is set equal to the command file. If REC= is specified, then the default input file is left open.

When #RJIN specifies a device or file, the input file is opened with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | As user-specified, or not passed | |
| FOPTIONS | | 7 | |
| Domain Specification | 14,15 | 11 | Old file |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 000 | Formal designator |
| Record Format | 8,9 | 00 | Fixed |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 0 | Allow |
| AOPTIONS | | Not passed | |
| Record Size | | Not passed | |
| Device | | As user-specified, or not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

As noted previously, if the input file opened is the same as the current command file, then the input file is closed and both files are accessed as the same file.

## LIST FILE/OUT FILE

The default list file is opened after the :RJE command has been issued, with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | RJELIST | |
| FOPTIONS | | %2514 if unspecified<br>%507 if user-specified | |
| Domain Specification | 14,15 | 00 | Newfile if unspecified |
| | | 11 | Oldfile if user-specified |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 001 | $STDLIST if unspecified |
| | | 000 | Formal designator if user-specified |
| Record Format | 8,9 | 01 | Variable |
| Carriage Control Option | 7 | 1 | Yes |
| File Equation | 5 | 1 | Disallow if unspecified |
| | | 0 | Allow if user-specified |
| AOPTIONS | | 1 | |
| Access Type | 12 to 15 | 0001 | Write only, delete old data |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 00 | Default |
| Inhibit Buffering | 7 | 0 | No |

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Record Size | | Not passed | |
| Device | | Not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

If the FOPEN of a user-specified old file fails, then an attempt is made to open the file new. The following parameters are changed from those above:

| | | | |
|---|---|---|---|
| FOPTIONS | | %504 | |
| Domain Specification | 14,5 | 00 | Newfile |
| File Size | | 5000 records | |
| FCLOSE Disposition Code | | 1 | Save permanently |

Output will be written to the default list file (the file opened from the :RJE command) under the following conditions:

- #RJLIST is processed, with no file, device, or list procedure specified, and unrouted output or routed list output is received.
- #RJOUT is processed, with no file, device, or output procedure specified, and unrouted output or routed list output is received.

The default list file is closed when the Emulator terminates.

When the #RJLIST or #RJOUT specifies a device or file, the list or output file is opened with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | As user-specified, or not passed | |
| FOPTIONS | | %507 | If filename |
| | | %504 | If logical device or device class |
| Domain Specification | 14,15 | 11 | Oldfile if device |
| | | 00 | Newfile if file |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 000 | Formal designator |
| Record Format | 8,9 | 01 | Variable |
| Carriage Control Option | 7 | 1 | Yes |
| File Equation | 5 | 0 | Allow |
| AOPTIONS | | 1 | |
| Access Type | 12 to 15 | 1 | Write only, delete old data |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 00 | No |
| Inhibit Buffering | 7 | 0 | No |
| Record Size | | Not passed | |
| Device | | As user-specified, or not passed | |
| Forms Message | | As user-specified, or not passed | |
| User Labels | | Not passed | |

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

If the FOPEN of a user-specified old file fails, then an attempt is made to open the file new. The following parameters are changed from those above:

| | | | |
|---|---|---|---|
| FOPTIONS | | %504 | |
| Domain Specification | 14,15 | 00 | Newfile |
| FCLOSE Disposition Code | | 1 | Save permanently |

Output will be written to the list file specified on #RJLIST if unrouted output or routed list output is received. If an output file is specified on #RJOUT, then any received data, routed or unrouted, will be written to it. Note that there is no default output file opened on the :RJE command; an output file is opened only when specified on the #RJOUT command. List and output files specified on the #RJLIST and #RJOUT command are closed when the command completes processing.

The value of the control parameter passed to FWRITE for list and output files depends on the received data and, in the case of list files specified on #RJLIST, on the specifications of the AUTOPAGE parameter. If received data is non-transparent and the first byte of a received logical record is an ESC character (%33 if LINECODE=ASCII, %47 if LINECODE=EBCDIC), then the following byte is used to determine vertical forms control; otherwise vertical forms control is not determined by received data. The following table summarized the conversions used for vertical forms control, according to Emulator type and LINECODE:

| Function | 2780 ASCII | 2780 EBCDIC | 3780 ASCII | 3780 EBCDIC | HP 3000 Control | (AUTOPAGE=YES) |
|---|---|---|---|---|---|---|
| Single Space | Q | / | Q | / | %201 | (%40) |
| Double Space | R | S | R | S | %202 | (%60) |
| Triple Space | S | T | S | T | %203 | (%304) |
| Skip to 1 | A | A | A | A | %300 | |
| Skip to 2 | B | B | B | B | %301 | |
| Skip to 3 | C | C | C | C | %302* | |
| Skip to 4 | D | D | D | D | %303 | |
| Skip to 5 | E | E | E | E | %304 | |
| Skip to 6 | F | F | F | F | %305 | |
| Skip to 7 | G | G | G | G | %306 | |
| Skip to 8 | H | H | H | H | %307 | |
| Skip to 9 | | | I | I | %310 | |
| Skip to 10 | | | J | J | %311 | |
| Skip to 11 | | | K | K | %312 | |
| Skip to 12 | | | L | L | %313 | |
| Suppress Space | | | M | M | %53 | |

%If CHNL3=n is specified, the value n+%277 is used.

If vertical forms control is not determined by received data, than a value of %40 is used unless AUTOPAGE=NO is specified, in that case %201 is used. This is the only case where where omitting the AUTOPAGE parameter is equivalent to AUTOPAGE= YES; when vertical forms control is determined by received data, omitting the AUTOPAGE parameter is equivalent to AUTOPAGE= NO.

If offline listing is being performed (the SOURCE parameter of #RJLIST was specified), then the control parameter used for FWRITE is 0; unless the SOURCE file FOPTIONS specifies carriage control option, in which case the control parameter is 1.

## PUNCH FILE

The default punch file is opened after the :RJE command has been issued, with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | RJEPUNCH | |
| FOPTIONS | | %2132 if unspecified<br>3 if user-specified | |
| Domain Specification | 14,15 | 10 | Old temporary file if unspecified |
| | | 11 | Old file if user-specified |
| ASCII/Binary | 13 | 0 | Binary |
| Default File Designator | 10,11,12 | 011 | $OLDPASS if unspecified |
| | | 000 | Formal designator if user-specified |
| Record Format | 8,9 | 01 | Variable if unspecified |
| | | 00 | Fixed if user-specified |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 1 | Disallow if unspecified |
| | | 0 | Allow if user-specified |
| AOPTIONS | | %102 | |
| Access Type | 12 to 15 | 0010 | Write only save old data |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 01 | Exclusive |
| Inhibit Buffering | 7 | 0 | No |
| Record Size | | 40 | |
| Device | | Not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Disposition Code | | 0 | No change |

If the punch file is not specified and $OLDPASS was opened but did not have a file code of 1060, then $OLDPASS is closed. If this condition occurred or if FOPEN of an old file failed, then an attempt is made to open the file new. The following parameters are changed from the above values:

| | | | |
|---|---|---|---|
| FOPTIONS | | %2120 | If unspecified |
| | | 0 | If user-specified |
| Domain Specification | 14,15 | 00 | New file |
| Default File Designator | 10,11,12 | 010 | $NEWPASS if unspecified |
| | | 000 | Formal designator if user-specified |

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| File code | | 1060 | |
| Fileclose Disposition Code | | 2 | Save temporary file if unspecified |
| | | 1 | Save permanent file if user-specified |

Output will be written to the default punch file (the file opened on the :RJE command) only if routed punch data is received while processing #RJOUT with no file, device, or punch procedure specified. The default punch file is closed when the Emulator terminates.

**When #RJPUNCH specifies a device or file, the punch file is opened with the following parameters:**

| | | | |
|---|---|---|---|
| Formal File Designator | | As user-specified, or not passed | |
| FOPTIONS | | 3 | If filename |
| | | 0 | If logical device or device class |
| Domain Specification | 14,15 | 11 | Oldfile if device |
| | | 00 | Newfile if file |
| ASCII/Binary | 13 | 0 | Binary |
| Default File Designator | 10,11,12 | 000 | Formal Designator |
| Record Format | 8,9 | 00 | Fixed |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 0 | Allow |
| AOPTIONS | | %102 | |
| Access Type | 12 to 15 | 0010 | Write only, save old data |
| Multi-Record Access | 11 | 0 | No |
| Dynamic Locking | 10 | 0 | No |
| Exclusive Option | 8,9 | 01 | Exclusive |
| Inhibit Buffering | 7 | 0 | No |
| Record Size | | 40, or as specified on OUTSIZE | |
| Device | | As user-specified, or not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | 1060 | |
| FCLOSE Disposition Code | | 0 | No change |

If the FOPEN of a user-specified old file fails, then an attempt is made to open the file new. The following parameters are changed from those above:

| | | | |
|---|---|---|---|
| FOPTIONS | | 0 | |
| Domain Specification | 14,15 | 00 | New file |
| FCLOSE Disposition Code | | 1 | Save permanently |

Output will be written to the punch file specified on #RJPUNCH if unrouted output or routed punch output is received. Punch files specified on the #RJPUNCH command are closed when the command completes processing.

# SOURCE FILE

A source file is opened when the SOURCE parameter is specified on the #RJLIST or #RJPUNCH, indicating that an offline list or punch operation is to be performed. The source file is opened with the following parameters:

| Parameter | Bits | Value | Meaning |
|---|---|---|---|
| Formal File Designator | | As user-specified, or not passed | |
| FOPTIONS | | 7 | |
| Domain Specification | 14,15 | 11 | Old file |
| ASCII/Binary | 13 | 1 | ASCII |
| Default File Designator | 10,11,12 | 000 | Formal designator |
| Record Format | 8,9 | 00 | Fixed |
| Carriage Control Option | 7 | 0 | No |
| File Equation | 5 | 0 | Allow |
| AOPTIONS | | Not passed | |
| Record Size | | Not passed | |
| Device | | As user-specified, or not passed | |
| Forms Message | | Not passed | |
| User Labels | | Not passed | |
| Blocking Factor | | Not passed | |
| Number of Buffers | | Not passed | |
| File Size | | Not passed | |
| Number of Extents | | Not passed | |
| Initial Allocation | | Not passed | |
| File Code | | Not passed | |
| FCLOSE Designation Code | | 0 | No change |

**The source file is closed when the #RJLIST or #RJPUNCH command completes processing.**

# INDEX

# READER COMMENT SHEET

**HP 3000 Series II Computer System**
**2780/3780 Emulator**
**Reference Manual**

**30000-90047** **Jun 1977**

We welcome your evaluation of this manual. Your comments and suggestions help us improve our publications. Please use additional pages if necessary.

| | | |
|---|---|---|
| **Is this manual technically accurate?** | Yes ☐ No ☐ | (If no, explain under Comments, below.) |
| **Are the concepts and wording easy to understand?** | Yes ☐ No ☐ | (If no, explain under Comments, below.) |
| **Is the format of this manual convenient in size, arrangement, and readability?** | Yes ☐ No ☐ | (If no, explain or suggest improvements under Comments, below.) |

**Comments:**

---

**FROM:**

**Name** _____

**Company** _____

**Address** _____

_____

_____

FOLD FOLD

FIRST CLASS
PERMIT NO. 1020
SANTA CLARA
CALIFORNIA

# BUSINESS REPLY MAIL

No Postage Necessary if Mailed in the United States. Postage will be paid by

Publications Manager
Hewlett-Packard Company
General Systems Division
5303 Stevens Creek Boulevard
Santa Clara, California 95050

FOLD FOLD

HEWLETT hp PACKARD

# SALES & SERVICE OFFICES

## AFRICA, ASIA, AUSTRALIA

**ANGOLA**
Telectra
Empresa Técnica de Equipamentos Eléctricos, S.A.R.L.
R. Barbosa Rodrigues, 42-1°DT.°
Caixa Postal, 6487
**Luanda**
Tel: 35515/6
Cable: TELECTRA Luanda

**AUSTRALIA**
Hewlett-Packard Australia Pty. Ltd.
31-41 Joseph Street
**Blackburn**, Victoria 3130
P.O. Box 36
**Doncaster East**, Victoria 3109
Tel: 896351
Telex: 31-024
Cable: HEWPARD Melbourne

Hewlett-Packard Australia Pty. Ltd.
31 Bridge Street
**Pymble**
New South Wales, 2073
Tel: 4496566
Telex: 21561
Cable: HEWPARD Sydney

Hewlett-Packard Australia Pty. Ltd.
153 Greenhill Road
**Parkside**, S.A., 5063
Tel: 2725911
Telex: 82536
Cable: HEWPARD Adelaide

Hewlett-Packard Australia Pty.Ltd.
141 Stirling Highway
**Nedlands**, W.A. 6009
Tel: 3865455
Telex: 93859
Cable: HEWPARD Perth

Hewlett-Packard Australia Pty. Ltd.
121 Wollongong Street
**Fyshwick**, A.C.T. 2609
Tel: 804244
Telex: 62650
Cable: HEWPARD Canberra

Hewlett Packard Australia Pty. Ltd.
5th Floor
Teachers Union Building
495-499 Boundary Street
**Spring Hill**, Queensland 4000
Tel: 2291544
Cable: HEWPARD Brisbane

**BANGLADESH**
The General Electric Co. of Bangladesh Ltd.
Magnet House 72
Dilkusha Commercial Area
**Motijheil**, Dacca 2
Tel: 252415, 252419
Telex: 734
Cable: GECDAC Dacca

**ETHIOPIA**
Abdella Abdulmalik
P.O. Box 2535
**Addis Ababa**
Tel: 11 93 40

**GUAM**
Medical Only
Guam Medical Supply, Inc.
Suite C, Airport Plaza
P.O. Box 8947
**Tamuning** 96911
Tel: 646-4513
Cable: EARMED Guam

**HONG KONG**
Schmidt & Co. (Hong Kong) Ltd.
Wing On Centre, 28th Floor
Connaught Road, C.
**Hong Kong**
Tel: 5-455644
Telex: 74766 SCHMC HX

**INDIA**
Blue Star Ltd.
Kasturi Buildings
Jamshedji Tata Rd
**Bombay** 400 020
Tel: 29 50 21
Telex: 011-2156
Cable: BLUEFROST

Blue Star Ltd.
Sahas
414/2 Vir Savarkar Marg
Prabhadevi
**Bombay** 400 025
Tel: 45 78 87
Telex: 011-4093
Cable: FROSTBLUE

Blue Star Ltd.
Band Box House
Prabhadevi
**Bombay** 400 025
Tel: 45 73 01
Telex: 011-3751
Cable: BLUESTAR

Blue Star Ltd.
Bhavdeep
Stadium Road
**Ahmedabad** 380 014
Tel: 42880
Telex: 234
Cable: BLUEFROST

Blue Star Ltd.
7 Hare Street
P.O. Box 506
**Calcutta** 700 001
Tel: 23-0131
Telex: 021-7655
Cable: BLUESTAR

Blue Star Ltd.
Bhandari House
7th & 8th Floor
91 Nehru Place
**New Delhi** 110 024
Tel: 634770 & 635166
Telex: 031-2463
Cable: BLUESTAR

Blue Star Ltd.
Blue Star House
11/11A Magarath Road
**Bangalore** 560 025
Tel: 55668
Telex: 043-430
Cable: BLUESTAR

Blue Star Ltd.
Meeakshi Mandiram
xxx/1678 Mahatma Gandhi Rd.
**Cochin** 682 016
Tel: 32069,32161,32282
Telex: 0885-514
Cable: BLUESTAR

Blue Star Ltd.
1-1-117/1
Sarojini Devi Road
**Secunderabad** 500 003
Tel: 70126, 70127
Telex: 015-459
Cable: BLUEFROST

Blue Star Ltd.
2/34 Kodambakkam High Road
**Madras** 600 034
Tel: 82056
Telex: 041-379
Cable: BLUESTAR

**INDONESIA**
BERCA Indonesia P.T.
P.O. Box 496/Jkt.
Jln.Abdul Muis 62
**Jakarta**
Tel: 349255, 349886
Telex: 46748 BERSIL IA
Cable: BERSAL

BERCA Indonesia P.T.
P.O. Box 174/Sby.
23 Jln. Jimerto
**Surabaya**
Tel: 42027
Cable: BErcacon

**ISRAEL**
Electronics Engineering Div. of Motorola Israel Ltd.
16, Kremenetski Street
P.O.Box 25016
**Tel-Aviv**
Tel: 38973
Telex: 33569, 34164
Cable: BASTEL Tel-Aviv

**JAPAN**
Yokogawa-Hewlett-Packard Ltd.
Chuo Bldg., 4th Floor
4-20, Nishinakajima 5-chome
Yodogawa-ku, Osaka-shi
**Osaka**, 532
Tel: 06-304-6021
Telex: 523-3624

Yokogawa-Hewlett-Packard Ltd.
29-21, Takaido-Higashi 3-chome
Suginami-ku, **Tokyo** 168
Tel: 03-331-6111
Telex: 232-2024 YHP-Tokyo
Cable: YHPMARKET TOK 23 724

Yokogawa-Hewlett-Packard Ltd.
Nakamo Building
24 Kami Sasajima-cho
Nakamura-ku, **Nagoya**, 450
Tel: 052 571-5171

Yokogawa-Hewlett-Packard Ltd.
Tanigawa Building
2-24-1 Tsuruya-cho
Kanagawa-ku
**Yokohama**, 221
Tel: 045-312-1252
Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.
Mito Mitsui Building
105, 1-chome, San-no-maru
**Mito**, Ibaragi 310
Tel: 0292-25-7470

Yokogawa-Hewlett-Packard Ltd.
Inoue Building
1348-3, Asahi-cho, 1-chome
Atsugi, Kanagawa 243
Tel: 0462-24-0452

Yokogawa-Hewlett-Packard Ltd.
Kumagaya Asahi
Hachijuni Building
4th Floor
3-4, Tsukuba
**Kumagaya**, Saitama 360
Tel: 0485-24-6563

**KENYA**
Advanced Communications Ltd.
P.O. Box 30070
**Nairobi**
Tel: 331955
Telex: 22639

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 19012
Nairobi Airport
**Nairobi**
Tel: 336055/56
Telex: 22201/22301
Cable: INTAERIO Nairobi

Medical Only
International Aeradio (E.A.) Ltd.
P.O. Box 95221
**Mombasa**

**KOREA**
Samsung Electronics Co., Ltd.
15th Floor, Daeyongak Bldg.,
25-5, 1-KA
Choong Moo-Ro, Chung-Ku,
**Seoul**
Tel: (23) 6811, 778-3401/2/3/4
Telex: 2257S

**MALAYSIA**
Hewlett-Packard Sales SDN BHD
Suite 2.21/2.22
Bangunan Angkasa Raya
Jalan Ampang
**Kuala Lumpar**
Tel: 23320/27491

Protel Engineering
P.O. Box 1817
Lot 259, Satok Road
Kuching, **Sarawak**
Tel: 53544
Cable: PROTELENG

**MOZAMBIQUE**
A.N. Goncalves, Ltd.
162, 1° Apt. 14 Av. D. Luis
Caixa Postal 107
**Maputo**
Tel: 27091, 27114
Telex: 6-203 NEGON Mo
Cable: NEGON

**NEW GUINEA**
Hewlett-Packard Australia Pty. Ltd.
Development Bank Building
Ground Floor
Ward Strip
**Port Moresby**, Paupua
Tel: 258933

**NEW ZEALAND**
Hewlett-Packard (N.Z.) Ltd.
4-12 Cruickshank Street
Kilbirnie, Wellington 3
P.O. Box 9443
Courtney Place
**Wellington**
Tel: 877-199
Cable: HEWPACK Wellington

Hewlett-Packard (N.Z.) Ltd.
Pakuranga Professional Centre
267 Pakuranga Highway
Box 51092
**Pakuranga**
Tel: 569-651
Cable: HEWPACK Auckland

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
Scientific Division
79 Carlton Gore Road, Newmarket
P.O. Box 1234
**Auckland**
Tel: 75-289
Cable: DENTAL Auckland

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
Norrie and Parumoana Streets
**Porirua**
Tel: 75-098
Telex: 3858

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
P.O. Box 309
239 Stanmore Road
**Christchurch**
Tel: 892-019
Cable: DENTAL Christchurch

Analytical/Medical Only
Medical Supplies N.Z. Ltd.
303 Great King Street
P.O. Box 233
**Dunedin**
Tel: 88-817
Cable: DENTAL Dunedin

**NIGERIA**
The Electronics Instrumentations Ltd.
N6B/770 Oyo Road
Oluseun House
P.M.B. 5402
**Ibadan**
Tel: 461577
Telex: 31231 TEIL NG
Cable: THETIEL Ibadan

The Electronics Instrumentations Ltd.
144 Agege Motor Road, Mushin
P.O. Box 6645
**Lagos**
Cable: THETEIL Lagos

**PAKISTAN**
Mushko & Company Ltd.
Oosman Chambers
Abdullah Haroon Road
**Karachi-3**
Tel: 511027, 512927
Telex: 2894
Cable: COOPERATOR Karachi

Mushko & Company, Ltd.
388, Satellite Town
**Rawalpindi**
Tel: 41924
Cable: FEMUS Rawalpindi

**PHILIPPINES**
The Online Advanced Systems Corporation
Rico House
Amorsolo cor. Herrera Str.
Legaspi Village, Makati
P.O. Box 1510
Metro **Manila**
Tel: 85-35-81, 85-34-91,85-32-21
Telex: 3274 ONLINE

**RHODESIA**
Field Technical Sales
45 Kelvin Road North
P.O. Box 3458
**Salisbury**
Tel: 705231 (5 lines)
Telex: RH 4122

**SINGAPORE**
Hewlett-Packard Singapore (Pte.) Ltd.
1150 Depot Road
Alexandra P.O. Box 58
**Singapore 4**
Tel: 270-2355
Telex: HPSG RS 21486
Cable: HEWPACK, Singapore

**SOUTH AFRICA**
Hewlett-Packard South Africa (Pty.) Ltd.
Private Bag Wendywood,
Sandton, Transvaal, 2144
Hewlett-Packard Centre
Daphne Street, Wendywood,
**Sandton**, 2144
Tel: 802-1040/8
Telex: 8-4782
Cable: HEWPACK Johannesburg

Hewlett-Packard South Africa (Pty.), Ltd.
P.O. Box 120
Howard Place, Cape Province, 7450
Pine Park Centre, Forest Drive,
**Pinelands**, Cape Province, 7405
Tel: 53-7955 thu 9
Telex: 57-0006

**SRI LANKA**
Metropolitan Agencies Ltd.
209/9 Union Place
**Colombo** 2
Tel: 35947
Telex: 1377METROLTD CE
Cable: METROLTD

**SUDAN**
Radison Trade
P.O. Box 921
**Khartoum**
Tel: 44048
Telex: 375

**TAIWAN**
Hewlett-Packard Far East Ltd.
Taiwan Branch
39 Chung Hsiao West Road
Section 1, 7th Floor
**Taipei**
Tel: 3819160-9,3141010
Cable: HEWPACK TAIPEI

Hewlett-Packard Far East Ltd.
Taiwan Branch
68-2, Chung Cheng 3rd. Road
**Kaohsiung**
tel: (07) 242318-Kaohsiung

Analytical Only
San Kwang Instruments Co., Ltd.
20 Yung Sui Road
**Taipei**
Tel: 3615446-9 (4 lines)
Telex: 22894 SANKWANG
Cable: SANKWANG Taipei

**TANZANIA**
Medical Only
International Aeradio (E.A.), Ltd.
P.O. Box 861
**Dar es Salaam**
Tel: 21251 Ext. 265
Telex: 41030

**THAILAND**
UNIMESA Co. Ltd.
Elcom Research Building
2538 Sukumvit Ave.
**Bangchak,Bangkok**
Tel: 3932387, 3930338
Cable: UNIMESA Bangkok

**UGANDA**
Medical Only
International Aeradio (E.A.), Ltd.
P.O. Box 2577
**Kampala**
Tel: 54388
Cable: INTAERIO Kampala

**ZAMBIA**
R.J. Tilbury (Zambia) Ltd.
P.O. Box 2792
**Lusaka**
Tel: 73793
Cable: ARJAYTEE, Lusaka

**OTHER AREAS NOT LISTED, CONTACT:**
Hewlett-Packard Intercontinental
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 856-1501
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

## CANADA

**ALBERTA**
Hewlett-Packard (Canada) Ltd.
11620A - 168th Street
**Edmonton** T5M 3T9
Tel: (403) 452-3670
TWX: 610-831-2431

Hewlett-Packard (Canada) Ltd.
210,7220 Fisher St. S.E.
**Calgary** T2H 2H8
Tel: (403) 253-2713
Twx: 610-821-6141

**BRITISH COLUMBIA**
Hewlett-Packard (Canada) Ltd.
10691 Shellbridge Way
**Richmond** V6X 2W7
Tel: (604) 270-2277
TWX: 610-925-5059

**MANITOBA**
Hewlett-Packard (Canada) Ltd.
380-550 Century St.
**Winnipeg** R3H 0Y1
Tel: (204) 786-6701
TWX: 610-671-3531

**NOVA SCOTIA**
Hewlett-Packard (Canada) Ltd.
800 Windmill Road
**Dartmouth** B3B 1L1
Tel: (902) 469-7820
TWX: 610-271-4482

**ONTARIO**
Hewlett-Packard (Canada) Ltd.
1020 Morrison Dr.
**Ottawa** K2H 8K7
Tel: (613) 820-6483
TWX: 610-563-1636

Hewlett-Packard (Canada) Ltd.
6877 Goreway Drive
**Mississauga** L4V 1M8
Tel: (416) 678-9430
TWX: 610-492-4246

Hewlett-Packard (Canada) Ltd.
552 Newbold Street
**London** N6E 2S5
Tel: (519) 686-9181

**QUEBEC**
Hewlett-Packard (Canada) Ltd.
275 Hymus Blvd.
**Pointe Claire** H9R 1G7
Tel: (514) 697-4232
TWX: 610-422-3022
TLX: 05-821-521 HPCL

**FOR CANADIAN AREAS NOT LISTED:**
Contact Hewlett-Packard (Canada) Ltd. in Mississauga.

2/79

## CENTRAL AND SOUTH AMERICA

**ARGENTINA**
Hewlett-Packard Argentina S.A.
Av. Leandro N. Alem 822 - 12°
**1001 Buenos Aires**
Tel: 31-6063,4,5,6
Telex: 122443 AR CIGY
Cable: HEWPACKARG

Biotron S.A.C.I.y M.
Bolivar 177
**1066 Buenos Aires**
Tel: 30-4846, 34-9356, 34-0460, 33-2863
Telex: 011-7595
Cable: Biotron Baries

**BOLIVIA**
Casa Kavlin S.A.
Calle Potosi' 1130
P.O. Box 500
**La Paz**
Tel: 41530,53221
Telex: CWC BX 5298,ITT 3560082
Cable: KAVLIN

**BRAZIL**
Hewlett-Packard do Brasil I.e.C. Ltda.
Alameda Rio Negro, 750
Alphaville
**06400 Barueri SP**
Tel: 429-3222
Cable: HEWPACK Sao Paulo

Hewlett-Packard do Brasil I.e.C. Ltda.
Rua Padre Chagas, 32
**90000-Pôrto Alegre-RS**
Tel: (0512) 22-2998, 22-5621
Cable: HEWPACK Pôrto Alegre

Hewlett-Packard do Brasil I.e.C. Ltda.
Av. Epitacio Pessoa, 4664
Lagoa
**20000-Rio de Janeiro-RJ**
Tel:
Telex: 021-21905 HPBR-BR
Cable: HEWPACK Rio de Janeiro

**CHILE**
Jorge Calcagni y Cia. Ltda
Vicuna MacKenna 3, Ofic. 1204
Casilla 16475
Correo 9, **Santiago**
Tel: 34152
Telex: JCALCAGNI

**COLOMBIA**
Instrumentación
Henrik A. Langebaek & Kier S.A.
Carrera 7 No. 48-75
Apartado Aéreo 6287
**Bogotá**, I D.E.
Tel: 269-8877
Telex: 44400
Cable: AARIS Bogotá

Instrumentacion
H.A. Langebaek & Kier S.A.
Carrera 63 No. 49-A-31
Apartado 54098
**Medellin**
Tel: 304475

**COSTA RICA**
Cientifica Costarricense S.A.
Avenida 2, Calle 5
San Pedro de Montes de Oca
Apartado 10159
**San Jose**
Tel: 24-38-20, 24-08-19
Telex: 2367 GALGUR CR
Cable: GALGUR

**ECUADOR**
Computadoras y Equipos Electrónicos del Ecuador
P.O. Box 6423 CCI
Eloy Alfaro No. 1824,3°Piso
**Quito**
Tel: 453 482
Telex: 2548 CYEDE ED
Cable: CYEDE-Quito

Medical Only
Hospitalar S.A.
Casilla 3590
Robles 625
**Quito**
Tel: 545-250
Cable: HOSPITALAR-Quito

**EL SALVADOR**
IPESA
Bulevar de los Heroes 11-48
**San Salvador**
Tel: 252787

**GUATEMALA**
IPESA
Avenida Reforma 3-48,
Zona 9
**Guatemala City**
Tel: 316627,314786,66471-5,ext.9
Telex: 4192 Teletro Gu

**MEXICO**
Hewlett-Packard Mexicana, S.A. de C.V.
Av. Periférico Sur No. 6501
Tepepan, Xochimilco
**Mexico** 23, D.F.
Tel: 905-676-4600
Telex: 017-74-507

Hewlett-Packard Mexicana, S.A. de C.V.
Ave. Constitución No. 2184
**Monterrey**, N.L.
Tel: 48-71-32, 48-71-84
Telex: 038-410

**NICARAGUA**
Roberto Terán G.
Apartado Postal 689
Edificio Terán
**Managua**
Tel: 25114, 23412,23454,22400
Cable: ROTERAN Managua

**PANAMA**
Electrónico Balboa, S.A.
Aparatado 4929
Panama 5
Calle Samuel Lewis
Edificio "Alfa", No.2
**Cuidad de Panama**
Tel: 64-2700
Telex: 3483103 Curundu, Canal Zone
Cable: ELECTRON Panama

**PERU**
Compañía Electro Médica S.A.
Los Flamencos 145
San Isidro Casilla 1030
**Lima 1**
Tel: 41-4325
Telex: Pub. Booth 25424 SISIDRO
Cable: ELMED Lima

**SURINAME**
Surtel Radio Holland N.V.
Grote Hofstr. 3-5
P.O. Box 155
**Paramaribo**
Tel: 72118, 77880
Cable: Surtel

**TRINIDAD & TOBAGO**
CARTEL
Caribbean Telecoms Ltd.
P.O. Box 732
69 Frederick Street
**Port-of-Spain**
Tel: 62-53068

**URUGUAY**
Pablo Ferrando S.A.C.ei.
Avenida Italia 2877
Casilla de Correo 370
**Montevideo**
Tel: 40-3102
Telex: 702 Public Booth Para Pablo Ferrando
Cable: RADIUM Montevideo

**VENEZUELA**
Hewlett-Packard de Venezuela C.A.
P.O. Box 50933
Caracas 105
Los Ruices Norte
3a Transversal
Edificio Segra
**Caracas** 107
Tel: 239-4133 (20 lines)
Telex: 25146 HEWPACK
Cable: HEWPACK Caracas

**FOR AREAS NOT LISTED, CONTACT:**
Hewlett-Packard
Inter-Americas
3200 Hillview Ave.
**Palo Alto**, California 94304
Tel: (415) 856-1501
TWX: 910-373-1260
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

## EUROPE, NORTH AFRICA AND MIDDLE EAST

**AUSTRIA**
Hewlett-Packard Ges.m.b.H.
Handelskai 52
P.O. Box 7
A-1205 **Vienna**
Tel: 351621-27
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

**BAHRAIN**
Medical Only
Wael Pharmacy
P.O. Box 648
**Bahrain**
Tel: 54886, 56123
Telex: 8550 WAEL GJ
Cable: WAELPHARM

Analytical Only
Al Hamidiya Trading
and Contracting
P.O. Box 20074
**Manama**
Tel: 259978, 259958
Telex: 8895 KALDIA GJ

**BELGIUM**
Hewlett-Packard Benelux
S.A./N.V.
Avenue du Col-Vert, 1,
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 660 50 50
Cable: PALOBEN Brussels
Telex: 23-494 paloben bru

**CYPRUS**
Kypronics
19 Gregorios Xenopoulos Street
P.O. Box 1152
**Nicosia**
Tel: 45628/29
Cable: Kypronics Pandehis
Telex: 3018

**CZECHOSLOVAKIA**
Vyvojova a Provozni Zakladna
Vyzkumnych Ustavu v Bechovicich
CSSR-25097 **Bechovice u Prahy**
Tel: 89 93 41
Telex: 12133

Institute of Medical Bionics
Vyskumny Ustav Lekarskej Bioniky
Jedlova 6
CS-88346
**Bratislava-Kramare**
Tel: 4251
Telex: 93229

**DDR**
Entwicklungslabor der TU Dresden
Forschungsinstitut Meinsberg
DDR-7305
**Waldheim/Meinsberg**
Tel: 37 667
Telex: 518741

Export Contact AG Zuerich
Guenther Forgber
Schlegelstrasse 15
1040 **Berlin**
Tel: 42-74-12
Telex: 111889

**DENMARK**
Hewlett-Packard A/S
Datavej 52
DK-3460 **Birkerod**
Tel: (02) 81 66 40
Cable: HEWPACK AS
Telex: 37409 hpas dk

Hewlett-Packard A/S
Navervej 1
DK-8600 **Silkeborg**
Tel: (06) 82 71 66
Telex: 37409 hpas dk
Cable: HEWPACK AS

**EGYPT**
I.E.A.
International Engineering Associates
24 Hussein Hegazi Street
Kasr-el-Aini
**Cairo**
Tel: 23 829
Telex: 93830
Cable: INTENGASSO

SAMITRO
Sami Amin Trading Office
18 Abdel Aziz Gawish
**Abdine-Cairo**
Tel: 24932
Cable: SAMITRO CAIRO

**FINLAND**
Hewlett-Packard OY
Nahkahousunti 5
P.O. Box 6
SF-00211 **Helsinki 21**
Tel: (90) 6923031

**FRANCE**
Hewlett-Packard France
Avenue des Tropiques
Les Ulis
Boite Postale No. 6
91401 **Orsay-Cedex**
Tel: (1) 907 78 25
TWX: 600048F

Hewlett-Packard France
Chemin des Mouilles
B.P. 162
69130 **Ecully**
Tel: (78) 33 81 25,
TWX: 310617F

Hewlett-Packard France
Péricentre de la Cépière
31081 **Toulouse-Le Mirail**
Tel:(61) 40 11 12
TWX: 510957F

Hewlett-Packard France
Le Ligoures
Bureau de vente de Marseilles
Place Rouée de Villeneuve
13100 **Aix-en-Provence**
Tel: (42) 59 41 02

Hewlett-Packard France
2, Allee de la Bourgnette
35100 **Rennes**
Tel: (99) 51 42 44
TWX: 740912F

Hewlett-Packard France
16, rue du Canal de la Marne
67300 **Schiltigheim**
Tel: (88) 83 08 10
TWX: 890141F

Hewlett-Packard France
Immeuble péricentre
Rue van Gogh
59650 **Villeneuve D Ascq**
Tel: (20) 91 41 25
TWX: 160124F

Hewlett-Packard France
Bureau de Vente
Centre d' affaires Paris-Nord
Bâtiment Ampère
Rue de la Commune de Paris
B.P. 300
93153 **Le Blanc Mesnil Cédex**
Tel: (01) 931 88 50

Hewlett-Packard France
Av. du Pdt. Kennedy
33700 **Merignac**
Tel: (56) 97 22 69

Hewlett-Packard France
"France-Evry" immeuble Lorraine
Boulevard de France
91035 **Evry-Cedex**
Tel: 077 96 60

Hewlett-Packard France
60, Rue de Metz
57130 **Jouy aux Arches**
Tel: (87) 69 45 32

**GERMAN FEDERAL REPUBLIC**
Hewlett-Packard GmbH
Vertriebszentrale Frankfurt
Berner Strasse 117
Postfach 560 140
D-6000 **Frankfurt** 56
Tel: (0611) 50-04-1
Cable: HEWPACKSA Frankfurt
Telex: 04 13249 hpffm d

Hewlett-Packard GmbH
Technisches Büro Böblingen
Herrenberger Strasse 110
D-7030 **Böblingen**, Württemberg
Tel: (0703) 667-1
Cable: HEWPACK Böblingen
Telex: 07265739 bbn

Hewlett-Packard GmbH
Technisches Büro Düsseldorf
Emanuel-Leutze-Str.1 (Seestern)
D-4000 **Düsseldorf**
Tel: (0211) 59711
Telex: 085/86 533 hpdd d

Hewlett-Packard GmbH
Technisches Büro Hamburg
Wendenstrasse 23
D-2000 **Hamburg** 1
Tel: (040) 24 13 93
Cable: HEWPACKSA Hamburg
Telex: 21 63 032 hphh d

Hewlett-Packard GmbH
Technisches Büro Hannover
Am Grossmarkt 6
D-3000 **Hannover 91**
Tel: (0511) 46 60 01
Telex: 092 3259

Hewlett-Packard GmbH
Technisches Büro Nürnberg
Neumeyerstrasse 90
D-8500**Nürnberg**
Tel: (0911) 56 30 83
Telex: 0623 860

Hewlett-Packard GmbH
Technisches Büro München
Eschenstrasse 5
D-8021 **Taufkirchen**
Tel: (089) 6117-1

Hewlett-Packard GmbH
Technisches Büro Berlin
Kaithstrasse 2-4
D-1000 **Berlin** 30
Tel: (030) 24 90 86
Telex:018 3405 hpbln d

**GREECE**
Kostas Karayannis
8 Omirou Street
**Athens** 133
Tel: 32 30 303/32/37 731

Analytical Only
INTECO
G. Papathanassiou & Co.
17 Marni Street
**Athens** 103
Tel: 5522 915/5221 989
Telex: 21 5329 INTE GR
Cable: INTEKNIKA

Medical Only
Technomed Hellas Ltd.
52 Skoufa Street
**Athens** 135
Tel: 3626 972

**HUNGARY**
MTA
Müszerügy és Méréstechnikai
Szolgalata
Hewlett-Packard Service
Lenin Krt. 67, P.O.Box 241
1391**Budapest** VI
Tel: 42 03 38
Telex: 22 51 14

**ICELAND**
Medical Only
Elding Trading Company Inc.
Hafnarvoli - Tryggvagötu
P.O.Box 895
IS-**Reykjavik**
Tel: 1 58 20/1 63 03
Cable: ELDING Reykjavik

**IRAN**
Hewlett-Packard Iran Ltd.
No. 13, Fourteenth St.
Mir Emad Avenue
P.O. Box 41/2419
**Tehran**
Tel: 851082-5
Telex: 213405 hewp ir

**IRELAND**
Hewlett-Packard Ltd.
King Street Lane
**Winnersh**, Wokingham
Berks, RG11 5AR
Tel: 08-England
Tel: (0734) 78 47 74
Telex: 847178
Cable: Hewple London

Hewlett-Packard Ltd.
2C Avonbeg Industrial Estate
Long Mile Road
**Dublin** 12, Eire
Tel: (01) 514322
Telex: 30439

Medical Only
Cardiac Services (Ireland) Ltd.
Kilmore Road
Artane
**Dublin** 5, Eire
Tel: (01) 315820

Medical Only
Cardiac Services Co.
95A Finaghy Rd. South
**Belfast** BT10 0BY
GB-Northern Ireland

**ITALY**
Hewlett-Packard Italiana S.p.A.
Via G.Di Vittorio , 9
20063 Cernusco
. Sul Naviglio (MI)
Tel: (2) 903691
Telex: 311046 HEWPACKIT

Hewlett-Packard Italiana S.p.A.
Via Turazza , 14
35100 **Padova**
Tel: (49) 664888
Telex: 41612 HEWPACKI

Hewlett-Packard Italiana S.p.A.
Via G. Armellini 10
1-00143 **Roma**
Tel: (06) 54 69 61
Telex: 61514
Cable: HEWPACKIT Roma

Hewlett-Packard Italiana S.p.A.
Corso Giovanni Lanza 94
I-10133 **Torino**
Tel:(011) 682245/659308

Medical/Calculators Only
Hewlett-Packard Italiana S.p.A.
Via Principe Nicola 43 G/C
I-95126 **Catania**
Tel:(095) 37 05 04

Hewlett-Packard Italiana S.p.A.
Via Nuova San Rocco A.
Capodimonte, 62A
I-80131 **Napoli**
Tel: (081) 7913544

Hewlett-Packard Italiana S.p.A.
Via E. Masi, 9/8
I-40137 **Bologna**
Tel: (051) 307887/300040

**JORDAN**
Mouasher Cousins Co.
P.O. Box 1387
**Amman**
Tel: 24907/39907
Telex: SABCO JO 1456
Cable: MOUASHERCO

**KUWAIT**
Al-Khaldiya Trading &
Contracting
P.O. Box 830-Safat
**Kuwait**
Tel:42 4910/41 1726

**LUXEMBURG**
Hewlett-Packard Benelux
S.A./N.V.
Avenue du Col-Vert, 1
(Groenkraaglaan)
B-1170 **Brussels**
Tel: (02) 672 22 40
Cable: PALOBEN Brussels
Telex: 23 494

**MOROCCO**
Dolbeau
81 rue Karatchi
**Casablanca**
Tel: 3041 82
Telex: 23051/22822
Cable: MATERIO

Gerep
3, rue d'Agadir
**Casablanca**
Tel: 272093/5
Telex: 23739
Cable: GEREP-CASA

Cogedir
31 rue Omar Slaoui
**Casablanca**
Tel: 27 65 40
Telex: 21737/23003
Cable: COGEDIR

**NETHERLANDS**
Hewlett-Packard Benelux N.V.
Van Heuven Goedhartlaan 121
P.O. Box 667
NL-**Amstelveen** 1134
Tel: (020) 47 20 21

**NORWAY**
Hewlett-Packard Norge A/S
Osterdalen 18
P.O. Box 34
1345 **Osteraas**
Tel: (02) 1711 80
Telex: 16621 hpnas n

Hewlett-Packard Norge A/S
Nygaardsgaten 114
5000 **Bergen**

**POLAND**
Biuro Informacji Technicznej
Hewlett-Packard
Ul Stawki 2, 6P
00-950 **Warszawa**
Tel: 33.25.88/39.67.43
Telex: 81 24 53 hepa pl

**UNIPAN**
Biuro Obslugi Technicznej
01-447 Warszawa
ul **Newelska** 6
Poland

Zaklady Naprawcze Sprzetu
Medycznego
Plac Komuny Paryskiej 6
90-007 **Lódz**
Tel: 334-41, 337-83
Telex: 886981

**PORTUGAL**
Telectra-Empresa Técnica de
Equipamentos Eléctricos S.a.r.l.
Rua Rodrigo da Fonseca 103
P.O. Box 2531
P-**Lisbon** 1
Tel: (19) 68 60 72
Cable: TELECTRA Lisbon
Telex: 12598

Medical only
Mundinter
Intercambio Mundial de Comércio
S.a.r.l.
P.O. Box 2761
Avenida Antonio Augusto
de Aguiar 138
P - **Lisbon**
Tel: (19) 53 21 31/7
Telex: 16691 munter p
Cable: INTERCAMBIO Lisbon

**QATAR**
Nasser Trading & Contracting
P.O. Box 1563
**Doha**
Tel: 22170
Telex: 4439 NASSER
Cable: NASSER

**RUMANIA**
Hewlett-Packard Reprezentanta
Bd.n. Balcescu 16
**Bucuresti**
Tel: 15 80 23/13 88 85
Telex: 10440

I.I.R.U.C.
Intreprinderea Pentru
Intretinerea
Si Repararea Utilajelor de Calcul
B-dul Prof. Dimitrie Pompei 6
**Bucuresti**-Sectorul 2
Tel: 88-20-70, 88-24-40, 88-67-95
Telex: 118

**SAUDI ARABIA**
Modern Electronic
Establishment (Head Office)
P.O. Box 1228, Baghdadiah Street
**Jeddah**
Tel: 27 798
Telex: 40035
Cable: ELECTA JEDDAH

Modern Electronic
Establishment (Branch)
P.O. Box 2728
**Riyadh**
Tel: 62596/66232
Cable: RAOUFCO

Modern Electronic
Establishment (Branch)
P.O. Box 193
**Al-Khobar**
Tel: 44678-44813

**SPAIN**
Hewlett-Packard Española, S.A.
Calle Jerez 3
**E-Madrid** 16
Tel: (1) 458 26 00 (10 lines)
Telex: 23515 hpe

Hewlett-Packard Espadñola S.A.
Colonia Miraslerra
Edificio Juban
% Costa Brava, 13
**Madrid** 34

Hewlett-Packard Española, S.A.
Milanesado 21-23
**E-Barcelona** 17
Tel: (3) 203 6200 (5 lines)

Hewlett-Packard Española, S.A.
Av Ramón y Cajal, 1
Edificio Sevilla, planta 9°
-**Seville** 5
Tel: 64 44 54/58

Hewlett-Packard Española S.A.
Edificio Albia II 7° B
E-**Bilbao** 1
Tel: 23 83 06/23 82 06

Hewlett-Packard Española S.A.
C/Ramon Gordillo 1
(Entlo.)
E-**Valencia**-10
Tel: 96-361.13.54/361.13.58

**SWEDEN**
Hewlett-Packard Sverige AB
Enighetsvägen 3, Fack
S-161 **Bromma** 20
Tel: (08) 730 05 50
Telex: 10721
Cable: MEASUREMENTS
Stockholm

Hewlett-Packard Sverige AB
Frötallsgatan 30
S-421 32 **Västra Frölunda**
Tel: (031) 49 09 50
Telex: 10721 via Bromma office

**SWITZERLAND**
Hewlett-Packard (Schweiz) AG
Zürcherstrasse 20
P.O. Box 307
CH-8952 **Schlieren-Zurich**
Tel: (01) 7305240
Telex: 53933 hpag ch
Cable: HPAG CH

Hewlett-Packard (Schweiz) AG
Château Bloc 19
CH-1219 **Le Lignon-Geneva**
Tel: (022) 96 03 22
Telex: 27333 hpag ch
Cable: HEWPACKAG Geneva

**SYRIA**
General Electronic Inc.
Nuri Basha-Ahnaf Ebn Kays Street
P.O. Box 5781
**Damascus**
Tel: 33 24 87
Telex: 11215 ITIKAL
Cable: ELECTROBOR DAMASCUS

Medical/Personal Calculator only
Sawah & Co.
Place Azmé
B.P. 2308
**Damascus**
Tel: 16 367-19 697-14 268

Suleiman Hilal El Mlawi
P.O. Box 2528
Mamoun Bitar Street, 56-58
**Damascus**Tel: 11 46 63

**TUNISIA**
Tunisie Electronique
31 Avenue de la Liberte
**Tunis**
Tel: 280 144

Corema
1 ter. Av. de Carthage
**Tunis**
Tel: 253 821
Telex: 12319 CABAM TN

**TURKEY**
TEKNIM Company Ltd.
Riza Sah Pehlevi
Caddesi No. 7
Kavaklidere, **Ankara**
Tel: 275800
Telex: 42155

Teknim Com., Ltd.
Barbaros Bulvari 55/12
Besiktyas, **Istanbul**
Tel: 613 546
Telex: 23540

Medical only
E.M.A.
Muhendislik Kollektif Sirketi
Mediha Eldem Sokak 41/6
Yüksel Caddesi
**Ankara**
Tel: 17 56 22
Cable: EMATRADE/Ankara

Analytical only
Yilmaz Ozyurek
Milli Mudafaa Cad 16/6
Kizilay

**Ankara**
Tel: 25 03 09 - 17 80 26
Telex: 42576 OZEK TR
Cable: OZYUREK ANKARA

**UNITED ARAB EMIRATES**
Emitac Ltd. (Head Office)
P.O. Box 1641
**Sharjah**
Tel: 354121/3
Telex: 8135

Emitac Ltd. (Branch Office)
P.O. Box 2711
**Abu Dhabi**
Tel: 331370/1

**UNITED KINGDOM**
Hewlett-Packard Ltd.
King Street Lane
**Winnersh**, Wokingham
Berks. RG11 5AR
Tel: 07347 78/4774
Telex:847178/9

Hewlett-Packard Ltd.
Trafalgar House
Navigation Road
**Altrincham**
Cheshire WA14 1NU
Tel: (061) 928 6422
Telex: 668068

Hewlett-Packard Ltd.
Lygon Court
Hereward Rise
Dudley Road
**Halesowen**,
West Midlands B62 8SD
Tel: (021) 550 9911
Telex: 339105

Hewlett-Packard Ltd.
Wedge House
799, London Road
**Thornton Heath**
Surrey CR4 6XL
Tel: (01) 6840103
Telex: 946825

Hewlett-Packard Ltd
10, Wesley St.
**Castleford**
Yorks WF10 1AE
Tel: (0977) 550016
Telex: 557355

Hewlett-Packard Ltd
1, Wallace Way
**Hitchin**
Hertfordshire, SG4 0SE
Tel: (0462) 31111
Telex: 82.59.81

**USSR**
Hewlett-Packard
Representative Office USSR
Pokrovsky Boulevard 4/17-kw 12
**Moscow** 101000
Tel: 294.20.24
Telex: 7825 hewpak su

**YUGOSLAVIA**
Iskra-Standard/Hewlett-Packard
Mildosiceva 38/VII
61000 **Ljubljana**
Tel: 31 58 79/32 16 74

**SOCIALIST COUNTRIES NOT SHOWN PLEASE CONTACT:**
Hewlett-Packard Ges.m.b.H
Handelskai 52
P.O. Box 7
A-1205 **Vienna**, Austria
Tel: (0222) 35 16 21 to 27

**MEDITERRANEAN AND MIDDLE EAST COUNTRIES NOT SHOWN PLEASE CONTACT:**
Hewlett-Packard S.A.
Mediterranean and Middle
East Operations
35, Kolokotroni Street
Platia Kefallariou
GR-Kifissia-**Athens**, Greece
Tel: 8080337/259/429

**FOR OTHER AREAS NOT LISTED CONTACT**
Hewlett-Packard S.A.
7, rue du Bois-du-Lan
P.O. Box
CH-1217 Meyrin 2 - **Geneva**
Switzerland
Tel: (022) 82 70 00

## UNITED STATES

**ALABAMA**
P.O. Box 4207
8290 Whitesburg Dr.
**Huntsville** 35802
Tel: (205) 881-4591

8933 E. Roebuck Blvd.
**Birmingham** 35206
Tel: (205) 836-2203/2

**ARIZONA**
2336 E. Magnolia St.
**Phoenix** 85034
Tel: (602) 244-1361

2424 East Aragon Rd.
**Tucson** 85706
Tel: (602) 889-4661

***ARKANSAS**
Medical Service Only
P.O. Box 5646
Brady Station
**Little Rock** 72215
Tel: (501) 376-1844

**CALIFORNIA**
1579 W. Shaw Ave.
**Fresno** 93771
Tel: (209) 224-0582

1430 East Orangethorpe Ave.
**Fullerton** 92631
Tel: (714) 870-1000

3939 Lankershim Boulevard
**North Hollywood** 91604
Tel: (213) 877-1282
TWX: 910-499-2671

5400 West Rosecrans Blvd.
P.O. Box 92105
World Way Postal Center
**Los Angeles** 90009
Tel: (213) 776-7500
TWX: 910-325-6608

***Los Angeles**
Tel: (213) 776-7500

3003 Scott Boulevard
**Santa Clara** 95050
Tel: (408) 988-7000

***Ridgecrest**
Tel: (714) 446-6165

646 W. North Market Blvd
**Sacramento** 95834
Tel: (916) 929-7222

9606 Aero Drive
P.O. Box 23333
**San Diego** 92123
Tel: (714) 279-3200

***Tarzana**
Tel: (213) 705-3344

**COLORADO**
5600 DTC Parkway
**Englewood** 80110
Tel: (303) 771-3455

**CONNECTICUT**
12 Lunar Drive
**New Haven** 06525
Tel: (203) 389-6551
TWX: 710-465-2029

**FLORIDA**
P.O. Box 24210
2727 N.W. 62nd Street
**Ft. Lauderdale** 33309
Tel: (305) 973-2600

4428 Emerson Street
Unit 103
**Jacksonville** 32207
Tel: (904) 725-6333

P.O. Box 13910
6177 Lake Ellenor Dr.
**Orlando** 32809
Tel: (305) 859-2900

P.O. Box 12826
Suite 5, Bldg. 1
Office Park North
**Pensacola** 32575
Tel: (904) 476-8422

**GEORGIA**
P.O. Box 105005
450 Interstate North Parkway
**Atlanta** 30348
Tel: (404) 955-1500

Medical Service Only
***Augusta** 30903
Tel: (404) 736-0592

P.O. Box 2103
1172 N. Davis Drive
**Warner Robins** 31098
Tel: (912) 922-0449

**HAWAII**
2875 So. King Street
**Honolulu** 96826
Tel: (808) 955-4455

**ILLINOIS**
5201 Tollview Dr.
**Rolling Meadows** 60008
Tel: (312) 255-9800
TWX: 910-687-2260

**INDIANA**
7301 North Shadeland Ave.
**Indianapolis**46250
Tel: (317) 842-1000
TWX: 810-260-1797

**IOWA**
2415 Heinz Road
**Iowa City** 52240
Tel: (319) 338-9466

**KENTUCKY**
Medical Only
3901 Atkinson Dr.
Suite 407 Atkinson Square
**Louisville** 40218
Tel: (502) 456-1573

**LOUISIANA**
P.O. Box 1449
3229-39 Williams Boulevard
**Kenner** 70063
Tel: (504) 443-6201

**MARYLAND**
7121 Standard Drive
Parkway Industrial Center
**Hanover** 21076
Tel: (301) 796-7700
TWX: 710-862-1943

2 Choke Cherry Road
**Rockville** 20850
Tel: (301) 948-6370
TWX: 710-828-9684

**MASSACHUSETTS**
32 Hartwell Ave.
**Lexington** 02173
Tel: (617) 861-8960
TWX: 710-326-6904

**MICHIGAN**
23855 Research Drive
**Farmington Hills** 48024
Tel: (313) 476-6400

724 West Centre Ave.
**Kalamazoo** 49002
Tel: (606) 323-8362

**MINNESOTA**
2400 N. Prior Ave.
**St. Paul** 55113
Tel: (612) 636-0700

**MISSISSIPPI**
322 N. Mart Plaza
**Jackson** 39206
Tel: (601) 982-9363

**MISSOURI**
11131 Colorado Ave.
**Kansas City** 64137
Tel: (816) 763-8000
TWX: 910-771-2087

1024 Executive Parkway
**St. Louis** 63141
Tel: (314) 878-0200

**NEBRASKA**
Medical Only
7101 Mercy Road
Suite 101
**Omaha** 68105
Tel: (402) 392-0948

**NEVADA**
***Las Vegas**
Tel: (702) 736-6610

**NEW JERSEY**
W. 120 Century Rd.
**Paramus** 07652
Tel: (201) 265-5000
TWX: 710-990-4951

Crystal Brook Professional
Building, Route 35
**Eatontown** 07724
Tel:(201) 542-1384

**NEW MEXICO**
P.O. Box 11634
Station E
11300 Lomas Blvd., N.E.
**Albuquerque** 87123
Tel: (505) 292-1330
TWX: 910-989-1185

156 Wyatt Drive
**Las Cruces** 88001
Tel: (505) 526-2484
TWX: 910-9983-0550

**NEW YORK**
6 Automation Lane
Computer Park
**Albany** 12205
Tel: (518) 458-1550
TWX: 710-444-4961

650 Perinton Hill Office Park
**Fairport** 14450
Tel: (716) 223-9950
TWX: 510-253-0092

No.1 Pennsylvania Plaza
55th floor
34th street & 8th Avenue
**New York** 10001
Tel: (212) 971-0800

5858 East Molloy Road
**Syracuse** 13211
Tel: (315) 455-2486

1 Crossways Park West
**Woodbury** 11797
Tel: (516) 921-0300

**NORTH CAROLINA**
5605 Roanne Way
**Greensboro** 27405
Tel: (919) 852-1800

**OHIO**
Medical/Computer Only
Bldg. 300
1313 E. Kemper Rd.
**Cincinnati** 45246
Tel: (513) 671-7400

16500 Sprague Road
**Cleveland** 44130
Tel: (216) 243-7300
TWX: 810-423-9430

330 Progress Rd.
**Dayton** 45449
Tel: (513) 859-8202

1041 Kingsmill Parkway
**Columbus** 43229
Tel: (614) 436-1041

**OKLAHOMA**
P.O. Box 32008
6301 N. Meridan Avenue
**Oklahoma City** 73112
Tel: (405) 721-0200

9920 E. 42nd Street
Suite 121
**Tulsa** 74145

**OREGON**
17890 SW Lower Boones
Ferry road
**Tualatin** 97062
Tel: (503) 620-3350

**PENNSYLVANIA**
111 Zeta Drive
**Pittsburgh** 15238
Tel: (412) 782-0400

1021 8th Avenue
King of Prussia Industrial Park
**King of Prussia** 19406
Tel: (215) 265-7000
TWX: 510-660-2670

**PUERTO RICO**
Hewlett-Packard Inter-Americas
Puerto Rico Branch Office
Calle 272,
Edif. 203 Urg. Country Club
**Carolina** 00924
Tel: (809) 762-7255
Telex: 345 0514

**SOUTH CAROLINA**
P. O. Box 6442
6941-0 N. Trenholm Road
**Columbia** 29260
Tel: (803) 782-6493

**TENNESSEE**
8914 Kingston Pike
**Knoxville** 37922
Tel: (615) 523-0522

3027 Vanguard Dr.
Director's Plaza
**Memphis** 38131
Tel: (901) 346-8370

***Nashville**
Medical Service only
Tel: (615) 244-5448

**TEXAS**
4171 North Mesa
Suite C110
**El Paso** 79902
Tel: (915) 533-3555

P.O. Box 1270
201 E. Arapaho Rd.
**Richardson** 75080
Tel: (214) 231-6101

P.O. Box 42816
10535 Harwin Dr.
**Houston** 77036
Tel: (713) 776-6400

***Lubbock**
Medical Service only
Tel: (806) 799-4472

205 Billy Mitchell Road
**San Antonio** 78226
Tel: (512) 434-8241

**UTAH**
2160 South 3270 West Street
**Salt Lake City** 84119
Tel: (801) 972-4711

**VIRGINIA**
P.O. Box 12778
**Norfolk** 23502
Tel: (804) 460-2671

P.O. Box 9669
2914 Hungary Springs Road
**Richmond** 23228
Tel: (804) 285-3431

**WASHINGTON**
Bellefield Office Pk.
1203-114th Ave. S.E.
**Bellevue** 98004
Tel: (206) 454-3971
TWX: 910-443-2446

P.O. Box 4010
**Spokane** 99202
Tel: (509) 535-0864

***WEST VIRGINIA**
Medical/Analytical Only
**Charleston**
Tel: (304) 345-1640

**WISCONSIN**
9004 West Lincoln Ave.
**West Allis** 53227
Tel: (414) 541-0550

**FOR U.S. AREAS NOT LISTED:**
Contact the regional office
nearest you: Atlanta, Georgia...
North Hollywood, California...
Rockville, Maryland...Rolling Meadows,
Illinois.Their complete
addresses are listed above.

***Service Only**

Part No. 30000-90047
Printed in U.S.A.
3RJE.320.30000-90047

HEWLETT hp PACKARD

Sales and service from 172 offices in 65 countries.
5303 Stevens Creek Blvd., Santa Clara, California 95050