SC21-7703-3

File No. S34-36

IBM System/34
Data Communications
Reference Manual

Program Number 5726-SS1

SC21-7703-3

File No. S34-36

IBM System/34
Data Communications
Reference Manual

Program Number 5726-SS1

Fourth Edition (April 1983)

This is a major revision of, and obsoletes, SC21-7703-2 and Technical Newsletters SN21-8153, SN21-8254. Support for Multiple Logical Units (MLU) for SRJE has been added. Changes or additions to the text are marked with a vertical bar to the left of the change or addition.

This edition applies to release 9 modification 0 of IBM System/34 System Support Program Product (Program Number 5726-SSP) and to all subsequent releases until otherwise indicated in new editions or technical newsletters. Changes are periodically made to the information herein; changes will be reported in technical newsletters or in new editions of this publication.

Use this publication only for the purposes stated in the *Preface*.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address below. Requests for copies of IBM publications and for technical information about the system should be made to your IBM representative or to the branch office serving your locality.

This publication could contain technical inaccuracies or typographical errors. Use the Reader's Comment Form at the back of this publication to make comments about this publication. If the form has been removed, address your comments to IBM Canada Ltd., Information Development, Department 849, 1150 Eglinton Avenue East, Don Mills, Ontario, Canada M3C 1H7. IBM may use and distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Preface

This publication provides programmers with reference information necessary to write System/34 BSC programs, and to use the MRJE (MULTI-LEAVING Remote Job Entry) and SRJE (SNA Remote Job Entry) utilities.

Information concerning the interactive communications feature of the SSP is in the *Interactive Communications Feature Reference Manual*.

Information in this manual is intended primarily for the System/34 user who has experience with data communications programming.

This manual contains the following chapters:

Chapter 1. Introduction describes the data communications capabilities provided with System/34.

Chapter 2. Data Communications Programming with BSC—RPG II describes the capabilities of BSC support for RPG II. The specifications required to write an RPG II BSC program for System/34 and sample programs are included to illustrate programming techniques and possible applications.

Chapter 3. Data Communications Programming with BSC—Basic Assembler describes the capabilities of BSC support for basic assembler. The macroinstructions required to write a basic assembler BSC program for System/34 and sample programs are included to illustrate programming techniques and possible applications.

Chapter 4. MULTI-LEAVING Remote Job Entry Utility—MRJE describes the MRJE utility for remote job entry host systems. The procedure command that runs the utility, the functions of the utility, and the utility control statements required to operate MRJE are described.

Chapter 5. Data Communications Print Utility describes the $DCSUP print utility, which processes disk output from MRJE and SRJE sessions. The OCL statements and procedure command that run the print utility are described. The utility control statements to control the operation of the print utility are also included.

Chapter 6. Forms Control Table Utility describes the $DCFUP utility which builds a forms control table for use by the MRJE or SRJE utility. The OCL statements and procedure command that run the utility are described. The utility control statements to control the operation of the utility are also included.

Chapter 7. SNA Remote Job Entry Utility—SRJE describes the SRJE utility for remote job entry host systems. The procedure command that runs the utility, the functions of the utility, and the utility control statements required to operate SRJE are described.

This publication contains the following appendixes:

Appendix A. ASCII and EBCDIC

Appendix B. Transmission Control Characters for BSC

Appendix C. BSC Polling and Addressing Characters for Tributary Stations

Appendix D. System/34 Interface to BSC Line Protocol

Appendix E. MRJE BSC Line Protocol

Appendix F. Flow of SNA Commands with SRJE Implementation

CONVENTIONS FOR ILLUSTRATING STATEMENT AND MACROINSTRUCTION FORMATS

In descriptions of utility control statements and macroinstructions, capitalized terms, brackets, braces, and underlining have special meanings.

Capitalized terms must be entered as they are shown. Numbers and special characters within a capitalized term must also be entered as they are shown. You must replace terms that are not capitalized with appropriate values. For example, the statement:

.. READFILE NAME-filename

could be entered

.. READFILE NAME-PAYROLL

Brackets ([ ]) are not entered as part of the statement. Brackets indicate that the expression they enclose is optional. For example:

[DATE-date]

means that you need not enter the expression.

Braces ({}) are not entered as part of the statement. Braces indicate that you must select one of the values enclosed within the braces. For example, in the parameter:

CMD-{N / Y}

either Y or N must be selected.

Underlining (__): If a parameter value is underlined, it is a default value. A default value is one that is automatically assigned when an optional parameter is omitted. In the example shown for braces, N is the default value.

RELATED PUBLICATIONS

- *Data Communication Concepts*, GC21-5169
- *General Information—Binary Synchronous Communications*, GA27-3004
- *IBM Systems Network Architecture General Information*, GA27-3102
- *IBM Synchronous Data Link Control General Information*, GA27-3093
- *IBM System/34 System Support Reference Manual*, SC21-5155
- *IBM System/34 Planning Guide*, GC21-5154
- *IBM System/34 Installation and Modification Reference Manual: Program Products and Physical Setup*, SC21-7689
- *IBM System/34 Functions Reference Manual*, SA21-9243
- *IBM System/34 Operator's Guide*, SC21-5158
- *IBM System/34 Displayed Messages Guide*, SC21-5159
- *IBM System/34 RPG II Reference Manual*, SC21-7667
- *IBM System/34 Basic Assembler and Macro Processor Reference Manual*, SC21-7705
- *IBM 3741 Data Station Reference Manual*, GA21-9183
- *IBM 3747 Data Converter Reference Manual and Operator's Guide*, GA21-9170
- *IBM System/32 Data Communications Reference Manual*, GC21-7691
- *OS/VS2 HASP II Version 4 Operator's Guide*, GC21-6993

- *IBM System/34 Interactive Communications Feature Reference Manual*, SC21-7751
- *IBM System/34 3270 Device Emulation User's Guide*, SC21-7868
- *IBM System/34 Data Areas and Diagnostic Aids Manual*, LY21-0049
- *Data Communications Concepts*, GC21-5169
- *Processor System (ASP) Version 2 Console Operator's Manual*, GH20-0321
- *OS/VS1 RES System Programmer's Guide*, GC28-6878
- *OS/VS1 VTAM System Programmer's Guide*, GC27-6996
- *OS/VS1 RES Workstation User's Guide*, GC28-6879
- *Operator's Library: OS/VS2 Remote Terminals*, GC38-0225
- *Operator's Library: OS/VS1 Reference*, GC38-0110
- *OS/VS2 System Programming Library: VTAM*, GC28-0688
- *Operator's Library: OS/VS2 MVS System Commands*, GC38-0229
- *Operator's Library: OS/VS2 JES2 Command Language Reference Summary*, GX38-0227
- *MVS/System Product Release 2 Operator's Library: JES2 Component Commands*, SC23-0048
- *Operator's Library: OS/VS2 MVS JES3 Commands*, GC23-0008
- *IBM Virtual Machine Facility/370: Remote Spooling Communications System (RSCS) User's Guide*, GC20-1816
- *Introduction to the IBM 3704 and 3705 Communications Controllers*, GA27-3051
- *IBM 3704/3705 Communications Network Control Program/VS Generation and Utilities Guide and Reference Manual (OS/VS and DOS/VS VTAM Users)*, GC30-3008
- *DOS/VSE POWER/VSE Installation and Operator's Guide*, GC33-5403
- *DOS/VS System Control Statements*, GC33-5376

Contents

Figures

Chapter 1. Introduction

IBM System/34 System Support Program Product provides the following data communications capabilities:

- RPG II with BSC support
- Basic Assembler with BSC support
- MRJE utility
- Print utility
- Forms Control Table utility
- SRJE utility

In addition, the System/34 SSP also supports the Interactive Communications Feature, a separately priced feature, and the 3270 Device Emulation Program Product, Program Number 5726-EM1. Information concerning these products can be found in the *Interactive Communications Feature Reference Manual* and the *3270 Device Emulation User's Guide*.

IBM System/34 SSP provides the BSC (binary synchronous communications) support for System/34 RPG II. The BSC support with RPG II enables you to develop RPG II application programs that use the communications adapter to transmit and receive data over communications lines connected with other systems or terminals.

IBM System/34 SSP provides BSC support for the System/34 Basic Assembler and Macro Processor Program Product. The BSC support with basic assembler allows you to develop basic assembler programs that use the communications adapter to transmit and receive data over communications lines connected with other systems or terminals.

IBM System/34 SSP includes the MULTI-LEAVING Remote Job Entry (MRJE) utility, which enables you to submit, execute, and obtain results of jobs from a host system via the communications adapter using BSC line protocol. Output from jobs can be returned to the submitting System/34 MRJE utility, directed to another work station, or directed to the host system output devices. The MRJE utility establishes single line connections, sends and receives data, and executes the termination procedures.

IBM System/34 SSP includes the $DCSUP print utility, which enables you to print disk files created by the MRJE and SRJE utilities.

IBM System/34 SSP includes the $DCFUP forms control table utility, which enables you to build a forms control table for use by the MRJE or SRJE utility.

IBM System/34 includes the SNA Remote Job Entry (SRJE) utility, which enables you to submit, execute, and obtain results of jobs from a host system via the communications adapter using SDLC line protocol. Output from jobs can be returned to the submitting SRJE utility, directed to another work station, or to the host system output devices. The SRJE utility establishes a single line connection, sends and receives data, and executes the termination procedures.

SYSTEM/34 DATA COMMUNICATIONS CONCEPTS

This section describes specific System/34 characteristics and operational concepts of a data communications system. More detailed information is available in the following publications:

- For a general description of data communications, see *Data Communications Concepts*.
- For a more detailed description of BSC concepts, see *General Information—Binary Synchronous Communications*.
- For a more detailed discussion of SNA, see *IBM Systems Network Architecture General Information*.
- For a more detailed discussion of SDLC, see *IBM Synchronous Data Link Control General Information*.
- For more detail concerning the System/34 communications adapter and data communications functions, see the *Functions Reference Manual*.

Operating Characteristics of the Communications Adapter Feature

The communications adapter feature allows the System/34 to communicate at rates up to 9600 bps. Two communications adapters can be installed on the System/34 with a maximum combined rate of up to 9600 bps when both adapters operate concurrently. The MLCA (multiline communications adapter) provides up to four communications lines. The maximum speed for any one communications line is 56,000 bps with the MLCA, and the aggregate speed of all lines cannot exceed 65,600 bps. If any one of the communications lines has a speed greater than 9600 bps, the aggregate of the other three lines cannot exceed 9600 bps. The features available with the MLCA include the Autocall feature, which allows calls to be placed automatically by the adapter, and the X.21 feature, which allows access to public data networks. These features are mutually exclusive. The Autocall feature or the X.21 feature is available only with the MLCA. The communications adapters operate in half-duplex mode over point-to-point and multipoint networks, including four-wire communications lines. Operation of the communications adapter feature on a System/34 is overlapped with processing operations and all input/output device operations.

A System/34 on a switched network supports manual dialing and manual answering except on lines equipped with the X.21 feature. X.21 permits only autoanswer and autocall on a switched network. System/34 supports automatic answering only when the attached modem (external or integrated) also supports it. Automatic calling of numbers is supported when the MLCA with the Autocall feature or the X.21 feature is installed on the system. For additional information, refer to the *Installation and Modification Reference Manual*.

All devices in the network, including the System/34, must:

- Use the same clocking source – modem, network (for DDSA or X.21), or business machine
- Use the same transmission rate (bps)
- Use the same transmission code (ASCII, EBCDIC, or EBCDIC transparency)
- Use compatible modems at each station

System/34 Requirements for BSC

The minimum system configuration required to support the MRJE utility, an RPG II BSC program, or a basic assembler BSC program is:

- System/34 with minimum main storage size of 32 K bytes
- Communications adapter
- System/34 System Support Program Product

Note: A main storage size of 48 K bytes is recommended if you use BSC.

System/34 Requirements for SNA

The minimum system configuration required to support the SRJE utility is:

- System/34 with minimum main storage size of 48 K bytes
- Communications adapter
- System/34 System Support Program Product

Chapter 2. Data Communications Programming with BSC–RPG II

IBM System/34 binary synchronous communications support gives you the ability to send and receive binary synchronous data over a communications line. The BSC support performs all functions necessary to establish the line connection, exchange identification sequences, send and receive data, and execute the correct termination or disconnect procedures. Appendix D contains a description of the System/34 line interface to BSC. Sample programs are included in this chapter to illustrate programming techniques and possible applications for System/34 RPG II data communications programming.

System/34 BSC support runs as a separate task from the RPG II program, thus allowing the RPG II program to be swapped into and out of main storage. The BSC task requires 6 K bytes of nonswappable main storage.

Note: The user area of main storage must be at least 20 K bytes or no other programs can run. If you wish to run BSC and spooling concurrently in a 32 K system, you must start the BSC program first and then start spooling.

OTHER SYSTEMS

Binary synchronous data transfers are possible between System/34 and the following:

- Another System/34 with either RPG II or basic assembler
- System/32 with either RPG II or basic assembler
- System/3 with RPG II, CCP, or MLMP
- System/7 with MSP/7
- Operating System or Disk Operating System Basic Telecommunications Access Method (OS, OS/VS, DOS/VS, or DOS BTAM)
- System/360 Model 20 Input/Output Control System for the Binary Synchronous Communications Adapter
- Operating System Telecommunications Access Method (OS or OS/VS TCAM)
- Operating System or Disk Operating System Virtual Telecommunications Access Method (DOS/VS or OS/VS VTAM)
- Customer Information Control System (CICS/DOS/VS or CICS/VS)
- Information Management System (IMS/VS)
- IBM 3741 Model 2 Data Station or Model 4 Programmable Work Station

- IBM System/38 with BSC support using RPG II or COBOL
- IBM 3747 Data Converter
- IBM 5231 Data Collection Controller Model 2 (as a 3741 in transmit mode only)
- IBM 3750 Switching System (World Trade only)
- IBM 5110 (in 3741 mode)
- IBM Series 1 (in System/3 mode)
- IBM 5260 Point of Sale Terminal (in 3740 mode)
- IBM 5280 Distributed Data System (in 3740 mode)
- IBM 6640 Document Printer[1]
- IBM Office System 6 Information Processor[1]
- IBM Mag Card II Typewriter–Communicating[1]
- IBM 6670 Information Distributor[1]
- IBM 6240 Magnetic Card Typewriter–Communicating[1]
- IBM Displaywriter System[1]

For a description of how to write RPG II programs, see the *RPG II Reference Manual*.

BSC STATION TYPES

RPG II permits System/34 to function as any of the following station types:

- Receive only (receive input data from a remote terminal).
- Transmit only (transmit data to a remote terminal).
- Transmit and receive (no conversational reply).

  Three modes of operation are possible:
  - Transmit a file, then receive another file
  - Receive a file, then transmit another file
  - Transmit records of one file interspersed with receiving records of another file

These functions, including sample RPG II programs, are described later in this chapter.

---

[1]Binary synchronous communications between these devices and System/34 is available via an RPQ.

RPG TELECOMMUNICATIONS SPECIFICATIONS

You use the RPG telecommunications specifications sheet (Figure 2-1) to enter the information necessary to establish and maintain the BSC link. Each BSC file (defined on the RPG file description specifications sheet as BSCA) must have a corresponding entry on the RPG II telecommunications sheet.

Note: Each data communications program implies a counterpart at the other end of the line. The two programs must be compatible. For example, a program which transmits ASCII data requires a program which receives ASCII data.

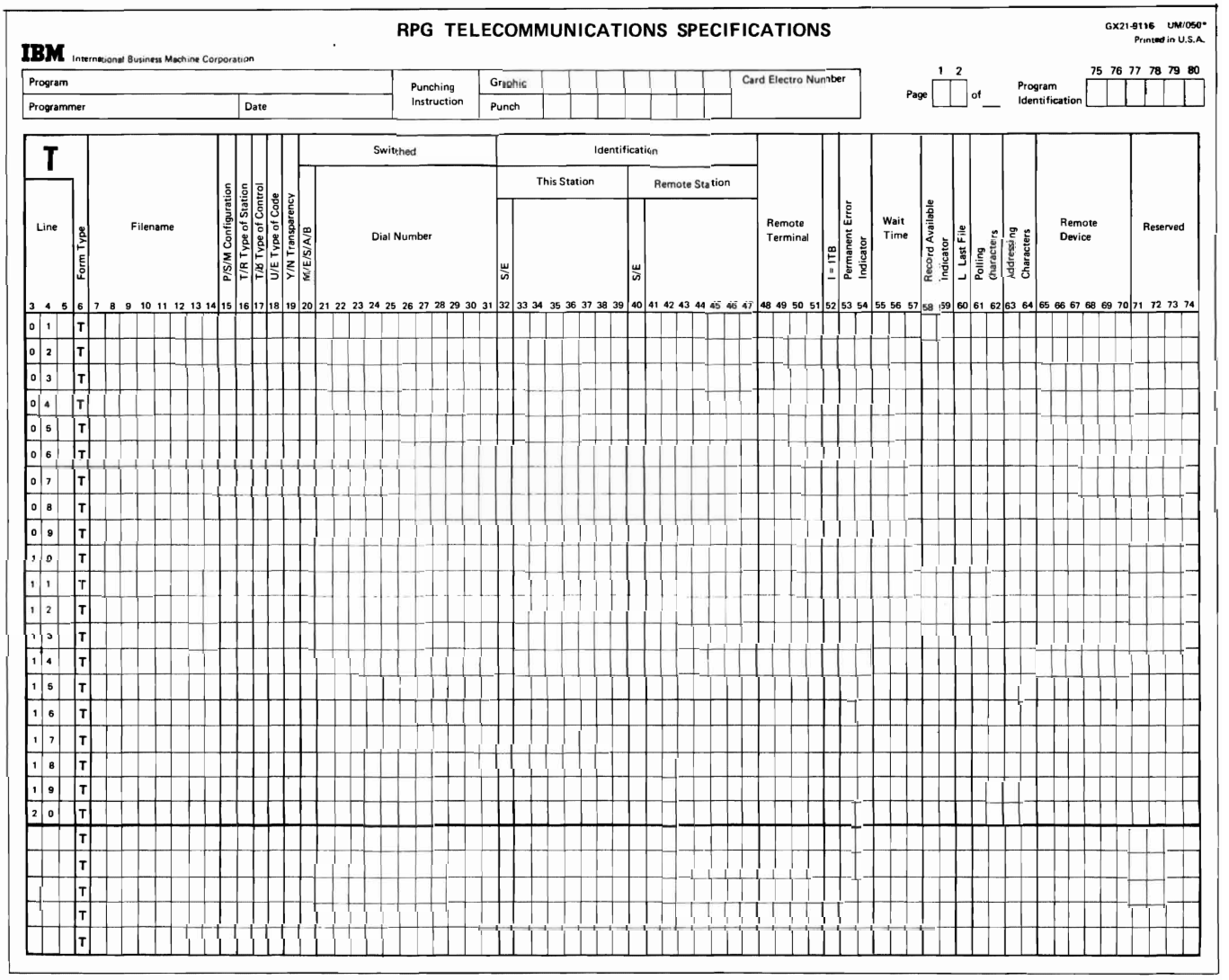


Figure 2-1. Telecommunications Specification Sheet

Columns 1-2 (Page)

Entry	Explanation
01-99	Page number

Columns 1 and 2 in the upper right corner of the specifications sheet are used to indicate the page number of the program. The telecommunications specifications must follow the file description and any extension and line counter specifications.

Columns 3-5 (Line)

Entry	Explanation
Any number	Line numbers

Columns 3 through 5 are used to number the lines on the specification page. Columns 3 and 4 are preprinted, so line numbering is done for you.

Column 6 (Form Type)

Entry	Explanation
T	Column 6 must contain a T. T identifies this line as a telecommunications specification.

Column 7 (Comments)

Entry	Explanation
*	Comment line

You often want to write comments that help you understand or remember what you are doing in a certain section of coding. RPG II allows you to use an entire line for these comments. The comment line is identified by placing an asterisk (*) in column 7. Comments are *not* instructions to the RPG II program. They serve only as a means of documenting your program.

Columns 7-14 (Filename)

Entry	Explanation
Alphameric	This entry must be the same as the file name associated with the BSC device on the file description specifications sheet.

Column 15 (Configuration)

Entry	Explanation
P or blank	This is a point-to-point nonswitched network.
S	This is a point-to-point switched network.
M	This is a multipoint network, where the control station selects the tributary station through polling or addressing. *Note*: System/34 cannot be the control station.

Note: Column 17 must contain a T if this column contains an M.

Column 16 (Type of Station)

Entry	Explanation
T	This station transmits messages from the file named in columns 7 through 14. The file must be designated as an output file on the file description specifications and must appear on the output specifications.
R	This station receives messages into the file named in columns 7 through 14. The file must be designated as an input file on the file description specifications and must appear on the input specifications.

Note: This entry is independent of the entry in column 20.

Column 17 (Type of Control)

Entry	Explanation
T	This is a tributary station on a multipoint network. *Note*: System/34 cannot be the control station.
Blank	This is not a tributary station.

Note: Column 17 must contain a T if column 15 contains an M (multipoint network).

Column 18 (Type of Code)

Entry	Explanation
A, U	ASCII (formerly referred to as USASCII) transmission control characters are used. When ASCII is used, each station must provide file translation when it is required.
E or blank	EBCDIC transmission control characters are used.

ASCII and EBCDIC charts are in Appendix A.

Column 19 (Transparency)

Entry	Explanation
Y	EBCDIC transparency is used. The data being transferred may contain transmission control characters and/or packed numeric or alphameric characters. EBCDIC transmission control characters are listed in Appendix B. (Column 18 must be E or blank.)
N or blank	EBCDIC transparency is not used. Zoned decimal or alphameric data is transmitted and received. The data being transferred cannot contain transmission control characters.

Column 20 (Switched)

Entry	Explanation
Blank	This is not a switched network.
M	The operator using the program makes the connection by dialing the number (manual dial).
A	This program uses automatic answer.
B	This program uses manual answer.

Notes:

1. If you are using an autocall line, the switch type specified has no effect. However, if no phone list is specified in the COMM OCL statement, the switch type specified here is established.
2. This entry is independent of the entry in column 16.

Columns 21-31 (Dial Number)

Columns 21 through 31 are not used. Leave them blank.

Column 32 (Location of Identification–This Station)

Entry	Explanation
S	This is a switched network. This station identification is at the position specified by the symbolic name in columns 33 through 39.
E	This is a switched network. The entry in columns 33 through 39 is this station identification.
Blank	This a nonswitched network or a switched network where no ID is used for this station.

Columns 33-39 (Identification–This Station)

Entry	Explanation
Alphameric	When column 32 contains an E, this entry is the actual identification sequence of this station (minimum of two characters). When column 32 contains an S, this entry is the symbolic name of the location of this station identification. The symbolic name must not be an array name. If the BSC file is a primary or secondary file, this symbolic name must refer to the first element of a table. (The table might have only one element.) The station identification can be from two to 15 characters. A station identification must not contain a control character sequence (Appendix B). The station identification is translated if the BSC files are being translated.

Column 40 (Location of Identification–Remote Station)

Entry	Explanation
S	This is a switched network. The remote station identification is at the position specified by the symbolic name in columns 41 through 47.
E	This is a switched network. The entry in columns 41 through 47 is the remote station identification.
Blank	This is a nonswitched network or a switched network where no ID is used for the remote station.

Columns 41-47 (Identification–Remote Station)

Entry	Explanation
Alphameric	When column 40 contains an E, this entry is the actual identification sequence of the remote station (minimum of two characters). When column 40 contains an S, this entry is the symbolic name of the location of the remote station identification. The symbolic name must not be an array name. If the BSC file is a primary or secondary file, this symbolic name must refer to the first element of a table. (The table might have only one element.) This ensures that the station identification is in storage before the communications line is opened. The station identification can be from two to 15 characters. A station identification must not contain a control character sequence (Appendix B). The station identification is translated if the BSC files are being translated.

Columns 48-51 (Remote Terminal)

Columns 48 through 51 are not used and must be blank.

Column 52 (ITB)

Entry	Explanation
I	Intermediate block check (ITB) is used.
Blank	ITB is not used.

Intermediate block checking can be used only if records are blocked.

Note: ITB and EBCDIC transparency cannot both be specified for the same BSC output file.

Columns 53-54 (Permanent-Error Indicator)

Entry	Explanation
Blank	No permanent-error indicator is specified. If a permanent error occurs when no permanent-error indicator is specified, a system halt occurs. The program cannot be restarted.
01-99, L1-L9, LR, H1-H9	A permanent-error indicator can be specified for every BSC file. If you are using more than one BSC file, each file can have a permanent-error indicator. Specifying a permanent-error indicator is recommended when running in unattended mode. The indicator does not have to be unique for each file. When a permanent error occurs, the specified error indicator is set on. You can then use the permanent-error indicator to condition appropriate programming response, such as printing a message or performing a controlled cancel. *Do not attempt further transmission while the permanent-error indicator is on.* This includes attempts to transmit more than one record during detail, total, or exception output. To avoid sending records while the error exists, condition each record to be transmitted with the not permanent-error indicator; either on the calculations specifications (columns 9 through 11), or on the output specifications (columns 23 through 31). To retry an operation after a permanent error occurs, turn off the permanent-error indicator. The RPG II program can then access the BSC file on which the error occurred. If an error occurs on the retry operation, the permanent-error indicator is turned on again; otherwise, processing continues.

Columns 53-54 (Permanent-Error Indicator) (continued)

Entry	Explanation
01-99, L1-L9, LR, H1-H9 (continued)	Keep the following points in mind when you retry an operation: • The permanent-error indicator is the only indication to the RPG II program that an error has occurred. A BSC information message will be displayed describing type of error. If a halt (H1-H9) is not issued as part of the permanent error routine, the BSC information message might not be preserved on the display screen. • Any data in the BSC buffers at the time of an error is lost. • Switched lines are not disabled when an error occurs, unless a disconnect sequence is received or the hardware detects disconnect. • Any data transmitted while the permanent-error indicator is on is invalid. Unless your program is designed to recognize all data, the error condition may cause an unidentified record halt. • Your program should limit the number of times an error can occur before the program is stopped.

Note: Avoid using H1-H9 as permanent-error indicators if you wish to condition operations on the permanent-error indicator being off. Since H1-H9 are reset at the end of the detail logic cycle, they can be set off before the program cycle in which the error occurred is complete.

Also, when H1-H9 is used as a permanent-error indicator, the H1-H9 display may preempt the system halt display. If the H1-H9 display appears before the system display, take the 0 option to prompt the system halt display.

Columns 55-57 (Wait Time)

Entry	Explanation
Numeric	This is the length of time in seconds, 1 to 999, that BSC will wait with no data being sent or received before an error message is displayed and before an abort (DLE-EOT) is sent to the remote station.
Blank	The system convention for timeout, 180 seconds, is used.

A permanent-error indication is recognized by the system whenever the wait time on a line elapses. Therefore, when you determine your wait time, consider time the operator may require to respond to halts and other processing interruptions, and also time the program might require for special operation such as table searches and computing square roots.

Note: The wait time limit specified applies *only* to delays at the System/34 and does not apply to the remote device. In addition, the time limit applies only during the transmission or reception of a file, not between file transmissions.

Columns 58-59 (Record Available Indicator)

Entry	Explanation
01-99, L1-L9, LR, H1-H9	A record available indicator should be specified if RVI is to be received. See Figure 2-5 for examples of using a record available indicator. This indicator is set on whenever a reverse interrupt (RVI) is received.
Blank	Record available indicator is not used.

Column 60 (Last File)

Entry	Explanation
L	This BSC input file is processed only after all other input files are processed.
Blank	This BSC file may not be the last input file processed.

Note: This entry does not affect demand files.

Columns 61-62 (Polling Characters)

Entry	Explanation
Alphameric	The polling identification of this station is needed, if this station is part of a multipoint network and the BSC file is a transmit (output) file. Polling and addressing characters must be used in pairs, as listed in Appendix C.
Blank	This station is not transmitting on a multipoint network.

Columns 63-64 (Addressing Characters)

Entry	Explanation
Alphameric	The addressing identification of this station is needed, if this station is part of a multipoint network and the BSC file is a receive (input) file. Polling and addressing characters must be used in pairs, as listed in Appendix C.
Blank	This station is not receiving on a multipoint network.

Note: Enter polling and addressing characters in EBCDIC; the compiler converts the characters to the form required by the code specified in column 18. (If ASCII was specified, enter uppercase addressing characters; and they are converted to lowercase ASCII characters.)

Columns 65-70 (Remote Device)

Columns 65 through 70 are not used and must be blank.

Columns 71-74 (Reserved)

Columns 71 through 74 are not used and must be blank.

Columns 75-80 (Program Identification)

Columns 75 through 80 may contain any characters. These columns may contain the program name used in the control specifications, or they may contain characters that identify a certain portion of the program. The entry is ignored by the compiler, but appears in the source program listing.

FILE DESCRIPTION SPECIFICATION ENTRIES FOR BSC FILES

The entries in the following columns are used to define a BSC file on the file description specifications. The entries in columns not listed are the same as for the basic RPG II language (*RPG II Reference Manual*).

Columns 7-14 (Filename)

This is the name of a BSC file. The same file name must be used on the RPG telecommunications specifications.

Note: Look-ahead fields must not be specified for a BSC file.

Column 15 (File Type)

Entry	Explanation
I	This is an input (receive) file.
O	This is an output (transmit) file.

Column 16 (File Designation)

This is the same as for the basic RPG II language except that:

- D (demand file) is the required entry for *transmit interspersed with receive.* BSC files also should be designated as demand files for any receiving program which does not address the BSC files immediately. For example, if your BSC file is defined as a secondary file, the communications line opens as soon as the program begins. This means that your wait time might elapse before you are ready to process the BSC file (see columns 55-57 of T specifications (Wait Time)). If the BSC file is defined as a demand file, however, the line does not open until the program is ready to receive the first record for the BSC file.
- R (record address file) is an invalid entry. A BSC file cannot be a record address file.

Column 17 (End of File)

Enter an E if end of file on the input (receive) file is to determine end of job. The BSC input file might be the only file with an E in column 17. However, if any other input file has an E in column 17, all BSC input files should also have an E in column 17. This E is not necessary for the BSC files; but when it is not specified and end of file is reached on another input file, the BSC files close and the system on the other end of the communications line has no indication of what has happened. When an E is specified for the BSC files, all systems can come to a successful end of job.

Column 19 (File Format)

Enter an F (fixed length) for BSC files.

Columns 20-23 (Block Length)

Enter the size of the blocks of data processed by BSC, right-justified. Block length for your BSC files must be a multiple of record length. The maximum block length is 4075. If block length is not specified, it defaults to record length.

Columns 24-27 (Record Length)

Enter the length of your BSC records, right-justified. If you do not specify a record length, the record length defaults to the maximum record length (4075).

Note: A record that has data of 0 length is ignored unless 3740 mode is used, in which case it is considered a file separator. If your program receives a record that has a length greater than 0 but shorter than the record size you specify, the remainder of that record contains blanks.

Column 32 (Additional Area)

Assign dual I/O areas in this column. Any number, 1 through 9, assigns two I/O areas. If this column is blank, only one I/O area is assigned. Dual I/O areas may improve throughput.

Columns 40-46 (Device)

Entry	Explanation
BSCA	This is the device entry for BSC files.

PROGRAMMING CONSIDERATIONS

RPG II First-Time Logic

RPG II first-time logic opens all primary and secondary input files during the first-time cycle. That is, one record is read from each primary and secondary input file before any input file is processed. However, depending upon the particular application, you might want to delay first-time logic for processing of your BSC input files. You can delay first-time logic by designating each BSC input file as a demand file (D in column 16 of the file description specifications). One or more BSC input file can also be designated as the last file (L in column 60 of the telecommunications specifications). If 3740 multiple file support is being used, all secondary input files should have the L in column 60. Remember that an entire BSC input file must be received before another BSC input file can be received.

Autocall Support

When the System/34 is configured with the MLCA (multiline communications adapter) and the Autocall feature, remote locations can be called automatically without operator intervention. Because there is no reference to the autocall capabilities in user programs, existing programs can use autocall without modification. You specify autocall by using the PHONE parameter on the COMM OCL statement. The COMM statement is described in the *System Support Reference Manual*.

The phone list specified in the COMM statement can contain up to 120 phone numbers and is generated by the DEFINEPN procedure, which is described in the *System Support Reference Manual*.

If a batch BSC job is run on an autocall line and no phone list is specified in the COMM statement (or there is no COMM statement), the call mode defaults to the mode specified in the user's DTF or the display station communications configuration record. The mode can be manual answer, manual call, or auto answer. If the phone list is specified in the COMM statement but the line is not an autocall line, or the autocall task was not loaded at IPL time, then the line is considered to be a manual or auto answer line, depending on the switch type defined for the line.

X.21 Support

When the System/34 is configured with the MLCA and the X.21 feature, remote locations are called automatically on switched lines. Existing programs can use the X.21 capabilities without modification. You specify the list of numbers for the public data network with the PHONE parameter of the COMM OCL statement. The COMM statement is described in the *System Support Reference Manual*.

The list specified in the COMM statement can contain up to 84 phone numbers and is generated by the DEFINX21 procedure, which is described in the *System Support Reference Manual*.

When the first request during any BSC job step is made to BSC data management, the phone list is searched for a number to call. The first time the list is referenced, the search begins with the first number. For each succeeding reference, the search begins with the next available number. If a number cannot be reached, its retry count is decremented and the next number is called. If no number in the list can be reached, either the permanent error indicator is turned on or the permanent error return code is passed to the user program. A message is displayed indicating each number that could not be reached. When a number is reached, a message is displayed indicating the number reached and communications proceed. When the job step ends, you can use the IF statement to reexecute the step and call the next number. You can also use the same phone list in a succeeding step of the job.

The ability to call multiple locations within a single BSC job step is primarily useful when the System/34 is receiving data from those locations. Because any number may be called during a request, transmission of data to a particular location should be performed with a single number phone list.

If, during the receiving of data, a permanent error occurs, the phone number associated with the communications link is not reset. Because the number is not reset, it cannot be called again on subsequent passes through the list. The recovery associated with that particular job step is the responsibility of the user.

Blank Compression/Expansion

In order to use the line more effectively and decrease communications line costs the System/34 BSC offers the RPG II and assembler users the capability of transmitting and/or receiving data with all contiguous blanks (two or more) removed. This is done by using the same format used by the IBM 3780.

For put files, data is moved from the logical buffer to the BSC I/O buffer with blanks removed and compression control characters inserted. After each record, an IRS (intermediate record separator) character is inserted.

For get files, the procedure is reversed; the System/34 recognizes the IRS control characters, inserts the blanks removed by the remote station, and moves the record from the BSC I/O buffer to the logical buffer.

To use blank compression/expansion, execute an OVERRIDE procedure or a SETR utility control statement with BLANK-C specified before executing the BSC program. See Appendix D for an example of blank compression/expansion.

Note: Blank compression/expansion cannot be used with transparent or ITB modes.

Blank Truncation

System/34 BSC is also capable of transmitting and/or receiving data with only the trailing blanks removed.

For put files, data is moved from the logical buffer to the BSC I/O buffer with all trailing blanks removed. After each record, an IRS character is inserted.

For get files, the data in the BSC I/O buffer is scanned until an IRS is encounterd. All data up to the IRS is moved to the logical buffer. The remainder of the logical buffer is blanked.

To use blank truncation, execute an OVERRIDE procedure or a SETR utility control statement with BLANK-T specified before executing the BSC program. See Appendix D for an example of blank truncation.

Notes:

1. Blank truncation cannot be used with ITB mode. Blank truncation can be specified with transparent mode; however, this negates the truncation capability since no blanks are removed. When truncation is used with transparent mode, the record length must be equal to the block length.
2. When you use blank compression/expansion or blank truncation with blocked records, the number of records per block will vary depending on the number of blanks in each record.

Control Breaks and Overflow

Take care when transmitting data during total time in any RPG II program that both transmits and receives. Because of the sequence of total and detail operations in the RPG II program cycle, data might not be available for output even though it is read.

Take similar care when assigning the overflow indicator to a BSC file in a program that both transmits and receives. A potential problem exists when RPG II, because of the program cycle, tries to transmit an overflow record while receiving.

Data Formats

System/34 RPG II support uses the following data formats for transmitting data; these formats must be used when sending data to System/34 from a processing unit.

- Nontransparent, non-ITB: STX-data-ETX(ETB)
- Nontransparent, ITB: STX-data-ITB-data-ITB-data-ETX(ETB)
- Transparent, non-ITB: DLE-STX-data-DLE-ETX(ETB)
- Transparent, ITB (receive files only):
  DLE-STX-data-DLE-ITB-DLE-STX-data
  DLE-ITB-DLE-STX-data-DLE-ETX(ETB)

Data can be fixed-length and either unblocked or blocked.

Errors

If an error occurs at either station, System/34 retries the operation up to seven times or up to the number of retries specified by the ALTERBSC procedure command or the SETB utility control statement. (See the *System Support Reference Manual* for more information on the SETB utility control statement and the ALTERBSC procedure command.)

RPG II Diagnostics

Refer to *Displayed Messages Guide,* for a discussion of RPG II diagnostics.

BSC Environment

BSC configuration information is altered by System/34 ALTERBSC and OVERRIDE SSP procedures. When you are running BSC programs from the input job queue, the configuration information from the system console is used for the job. The SSP retrieves the information at the time the job is run.

The ALTERBSC and OVERRIDE procedures run the $SETCF utility. Instead of using these procedures to alter the BSC environment, you can use the SETB and SETR utility control statements of the $SETCF utility. These procedures are described in *Modifying the MRJE Environment* in Chapter 4.

For information on coding System/34 procedure commands and utility control statements, see the *System Support Reference Manual.*

DESCRIPTIONS OF BSC FUNCTIONS

This section describes the functions that System/34 can perform as part of a data communications network. The sample RPG II programs later in this chapter illustrate these functions.

Receive Only Function

The receive only function allows you to receive input data from another station. The file can be either a primary, demand, or secondary file. Blocked records are permitted. Dual I/O areas can be used to achieve greater throughput for primary and secondary files. Dual I/O areas cannot be specified for a demand file.

The *receive only* file is defined as an input file on the RPG file description specifications sheet and as a *receive* file on the RPG II telecommunications specifications sheet.

Transmit Only Function

The *transmit only* function allows you to transmit BSC data to a remote location. Dual I/O areas and blocking of data can be used to increase throughput.

The *transmit only* file is defined as an output file on the RPG file description specifications and as a *transmit* file on the RPG telecommunications specifications.

Transmit and Receive

Two files are defined, one as an input file on the RPG file description specifications and as a receive file on the RPG telecommunications specifications. The other file is defined as an output file on the RPG file description specifications and as a transmit file on the RPG telecommunications specifications.

In any BSC program which transmits and receives, columns 15 and 17 through 47 must be identical in the two RPG telecommunications specifications lines.

Transmit and receive RPG II BSC programs can be written three ways:

1. Transmit a file, then receive a file.
2. Receive a file, then transmit a file.
3. Transmit records interspersed with receive records.

Transmit a File, Then Receive a File

The receive file must *not* be defined as the primary input file on the RPG file description specifications. If the receive file is a secondary file, column 60 of the telecommunications specifications must contain an L. Matching fields and the record available indicators must *not* be defined for the BSC files.

Note: An L entry is not needed if the receive file is a demand file.

Receive a File, Then Transmit a File

The receive file is defined as a primary, secondary, or demand file on the RPG file description specifications. The record available indicator must be blank on the RPG telecommunications specifications.

Transmit Interspersed with Receive

A *transmit interspersed with receive* program transmits data from one file and receives data in another; the data in the two files might not be related. Unlike conversational programs, a transmit interspersed with receive program might intersperse several records or several blocks of data at a time.

The receive file must be defined as a demand file on the RPG file description specifications sheet. The record available indicator must be defined on the RPG telecommunications specifications. System/34 must initiate the transmission and then suspend its transmit file to receive data incoming from the other station. (See Figure 2-5 for an example of this type of program.)

Programming Considerations: Once System/34 BSC begins to process the last record in the transmit file, System/34 ignores the record available indicator, whether or not the last record was actually transmitted. When BSC accepts for transmission the last record in the file, RPG II completes LR processing and begins to close the file.

Hence, if the next-to-last record or block of records intended for transmission prompts a request to transmit to System/34, the request might be ignored. The request is always ignored if it is prompted by the final record or block of records.

The System/34 programmer can avoid problems by adding a special record, which is agreed upon with the programmer of the other station, to the end of the System/34 transmit file. This record signals that System/34 went to end of job and cannot honor a request to receive, even though that request was just transmitted.

DEVICE-DEPENDENT CONSIDERATIONS

OS and OS/VS TCAM and OS/VS VTAM

System/34 can be part of a data communications network that includes the Operating System (OS and OS/VS) Telecommunications Access Method (TCAM) or the OS/VS Virtual Telecommunications Access Method (VTAM). (See *IBM System/360 and IBM System/370 Bibliography*, GA22-6822, for the order numbers and titles of publications on TCAM and VTAM.)

System/34 communicates with TCAM and VTAM in the same ways it communicates with another System/34. The System/34 programmer may approximate conversational mode by using RPG II.

The RPG II technique for approximating conversational mode consists of using the READ operation code to receive data and the EXCPT operation code to transmit data.

Instead of using one combined file for transmitting and receiving data, the RPG II technique requires two files: a demand file for input from BSC, and another file for output to BSC. No record available indicator is used. The System/34 program must know what TCAM or VTAM is going to do next (that is, send or receive) and perform the appropriate BSC receive or transmit operation.

A restriction when you communicate with TCAM is that to maintain a switched line connection, you must begin transmitting to TCAM within nine seconds after receiving end of file from TCAM; otherwise, you must dial to reestablish the line connection.

IBM 3740 Data Entry System

RPG II data communications programming supports the IBM 3741 Model 2 Data Station, the IBM 3741 Model 4 Programmable Work Station, and the IBM 3747 Data Converter in communicate mode as a remote device via the communications adapter on the System/34.

This section contains a description of the RPG II specifications required to communicate with the 3740 Data Entry System.

Restrictions

The following items should be noted when communicating between a 3740 and a System/34:

- A 3741 with an Expanded Communications Buffer Feature (ECB) (Feature Number 1680) has a maximum buffer size of 512 bytes.
- The Operator Identification Card Reader Feature (Feature Number 5450) and the Expanded Communications/Multipoint Data Link Control Feature (Feature Number 1685) on the 3741 are not supported by System/34 RPG II.
- A 3747 with the Blocking/Reformatting Feature (Feature Number 1480) has a maximum buffer size of 8050 bytes. However, System/34 RPG II will only handle a maximum of 4075 bytes.
- Through RPG II you can communicate with the 3741 or 3747 either by using single file support (single input and/or single output) or by using multiple file support (multiple input and/or multiple output). In the latter case, multiple files can be received from and/or transmitted to the 3740 system.
- Through RPG II, you may also transmit to and receive blocked records from a 3741 with ECB Feature or a 3747 with the Blocking/Reformatting Feature.

Single File Support

When communicating with the 3741, a maximum of two BSC files are allowed (one input and/or one output) per RPG II program for single file support. If two BSC files are used, the input file must be processed completely before processing the output file.

When communicating with the 3747 Data Converter, only one BSC file is allowed (either input or output).

Multiple File Support

Through RPG II you can communicate with the 3740 Data Entry System using the multiple file support of System/34. Multiple file support must be indicated via either the OVERRIDE procedure or the SETR utility control statement by specifying the MLTFL-Y parameter before executing the RPG II program. (See the *System Support Reference Manual* for OVERRIDE and SETR.)

When communicating with the 3741, multiple files may be either received, transmitted, or received and then transmitted. All 3740 input files must be received before System/34 can begin transmitting files to the 3740. When communicating with a 3747, multiple files may be either received or transmitted. When communicating the 5110, multiple files can be either received, transmitted, received and then transmitted, or transmitted and then received.

Blocked Record Support

Blocked records may be transmitted to and received from a 3741 with the ECB feature for either single or multiple 3740 files. Blocked record support is indicated via the OVERRIDE procedure or the SETR utility control statement by specifying the RCSP-1E parameter before executing the RPG II program. (See the *System Support Reference Manual,* for OVERRIDE and SETR.)

Blocked records may be transmitted to a 3747 with the Blocking/Reformatting Feature via the above method or in a manner similiar to normal RPG II blocking.

Note: Blocked record support can be used with either single or multiple file support.

RPG II input files can be primary, secondary, or demand files. However, the 3740 files must be processed one file at a time to the end of the file and in the order that the 3740 transmits them.

Secondary files are processed in the order listed on the file description specifications in the source program.

Demand files are processed in the order determined by the user's logic on the calculation specifications.

Output files must be processed one file at a time. That is, all records for a file must be transmitted before the first record for the next file is sent. When communicating with a 3741, if multiple files are received and then multiple files are transmitted in the same program, all input files must be processed before any output files are processed.

RPG II Specifications

Use of the 3740 affects RPG II file description, telecommunications, and output specifications.

Only the entries unique to the 3740 are described here.

RPG II File Description Specifications

Columns 20-23 (Block Length): Maximum block length is 128 bytes without the ECB feature on the 3741 or blocking/reformatting on the 3747.

If blocked records are to be transmitted to a 3741 with the ECB feature, the block length may be any multiple of the record length not exceeding 512 bytes.

If blocked records are received from a 3741 with the ECB feature, the block length *must be* N times the record length, where N is the result (disregarding the remainder) of dividing 512 by the record length plus one. For example, if the record length is 128:

- Record length plus one = 129
- 512 divided by 129 = 3 remainder 125
- N = 3
- Block length = 3 times 128 = 384

When communicating with a 3747 with the Blocking/Reformatting Feature, the block length is dependent on the use of the data at the 3747 and the amount of storage available (Feature Numbers 7690, 7691, or 7692). Blocking on the 3747 can be identical to that of the 3741 with ECB through the use of C3 control records. Blocking can also be handled in a manner that is similar to the RPG II blocking through use of the C3 control records. For an explanation of the C3 control records format, see *IBM 3747 Data Converter Reference Manual and Operator's Guide*.

Columns 24-27 (Record Length): Maximum record length is 128 bytes when communicating with a 3741. The 3747 maximum record length depends on the use of the data at the 3747 and whether the Blocking/Reformatting Feature is installed.

RPG Telecommunications Specifications

The 3740 files require some restrictions to the telecommunication specifications. Only the columns affected are listed here:

Column	Entry Not Allowed	Description
15	M	Multipoint network
17	T	Tributary station on a multipoint network
52	I	ITB

Columns 61-74: Must be blank.

RPG II Output Specifications

Columns 17-22: Must be blank.

USING RPG II FOR COMMUNICATION BETWEEN A SYSTEM/34 AND THE IBM 3750 (WORLD TRADE ONLY)

When a System/34 is connected to an IBM 3750 Switching System, the RPG II data communications program must allow message exchanges between the two systems. The application program can be written for message exchange related to the following 3750 functions:

1. Call recording
2. Contact monitoring under data processing system control
3. Inquiry to data processing system with recorded answer
4. Real time data collection to data processing system
5. Recording announcement to extension under data processing system
6. Transfer of recorded data to data processing system

Communications between the System/34 and the 3750 are binary synchronous, point-to-point operations in transparent mode. Only EBCDIC can be used. The System/34 operates as a transmit and receive station.

Programming Procedure

When end of transmission (EOT) is received, and the next step in your program is to receive more data, issue another read to the same BSC input file.

SAMPLE PROGRAMS

The following four sample programs are provided as examples of the various types of RPG II BSC programs. The first example is a transmit program; the second is a receive program; the third is a System/34-to-TCAM program; and the fourth is a transmit-interspersed-with-receive program.

Transmit Program

In the following transmit only program, a file is read and then transmitted. The week's data is sorted by name of salesman. The amount of each sale is written on the disk; and the total sales for each salesman is transmitted to the branch office. After all disk records containing sales information are read, the total of all sales is transmitted to the branch office.

RPG II File Description Specifications (Figure 2-2, Part 1)

Column	Description
7-14	WKLYSMRY is a BSC file.
15	Since WKLYSMRY is to be transmitted, it is an output file.
19	BSC files always have a fixed-length format.
20-27	Records are blocked.
32	Dual I/O areas are used.
40-46	BSCA is the device name.

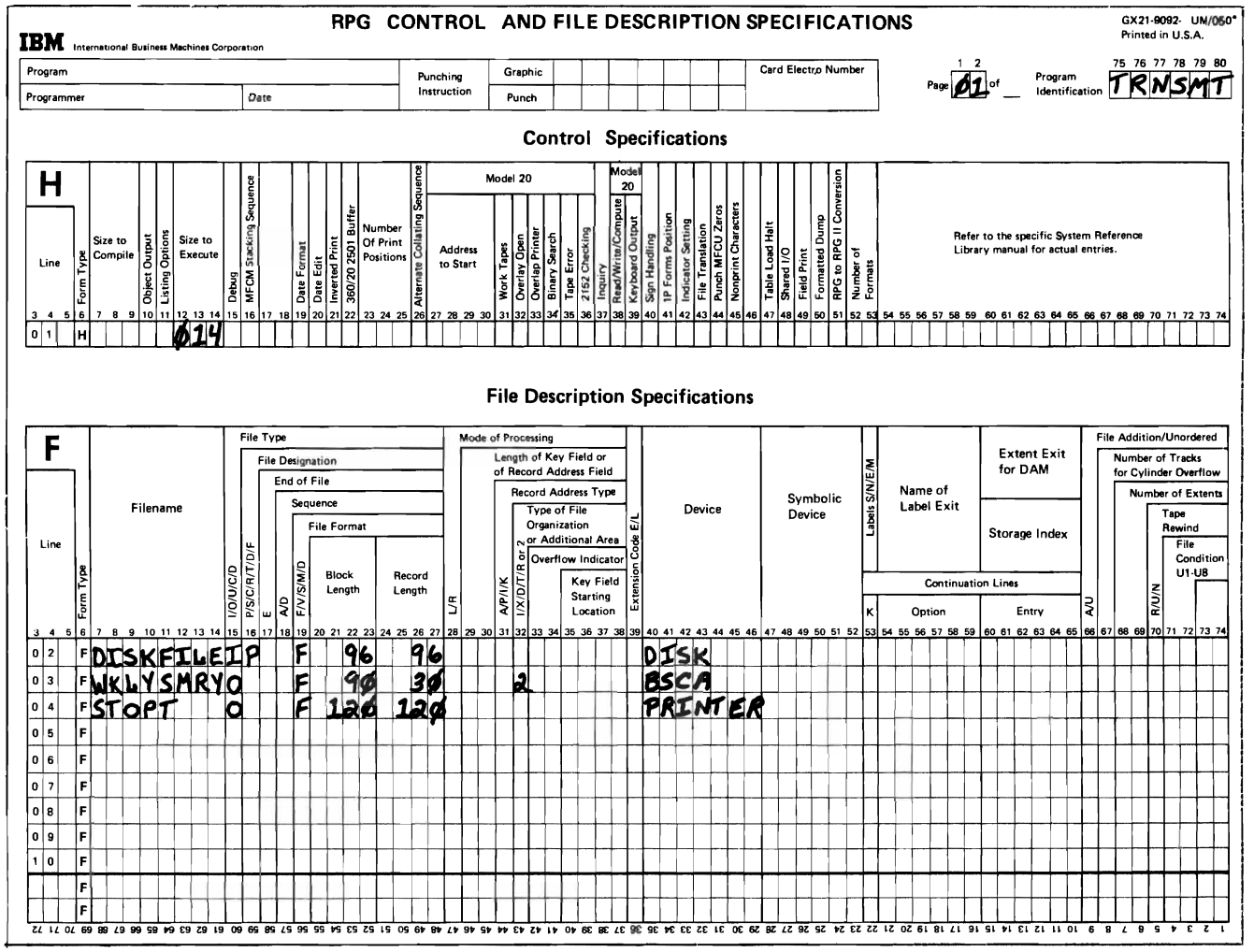
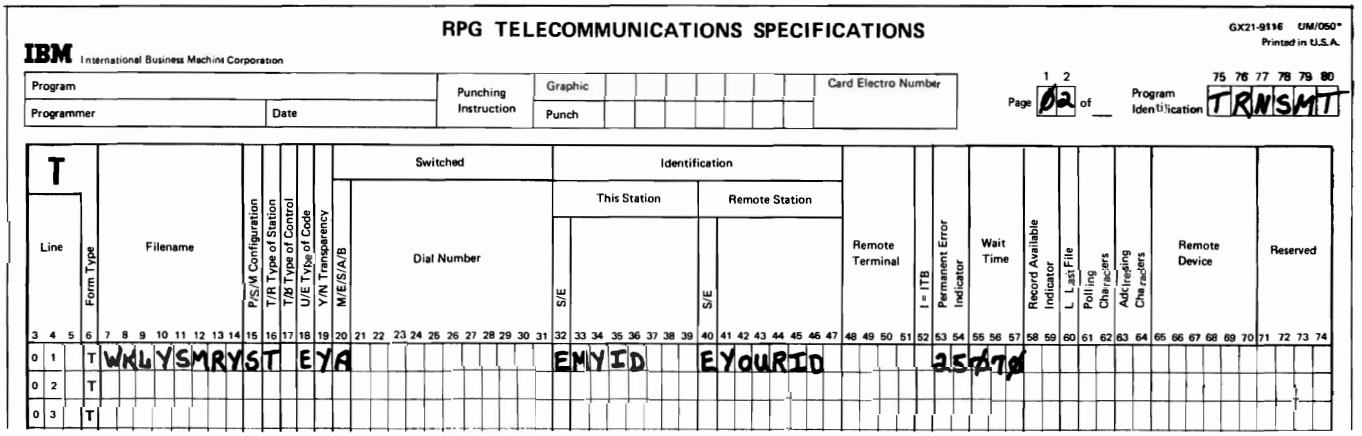


Figure 2-2 (Part 1 of 3). Transmit Program

RPG Telecommunications Specifications (Figure 2-2, Part 2)

Column	Description
7-14	WKLYSMRY is the BSC file for this program.
15-17	This station is part of a switched network (S), and is transmitting (T). Polling is not used, so column 17 is blank.
18-19	EBCDIC (E) and the transparency feature (Y) are both used.
20	Automatic answer (A) is used by this called station.
32-47	Explicit station identification sequences are given for each station. Station IDs help ensure data security on the switched network.
53-54	The permanent-error indicator used is 25.
55-57	The data communications line is kept open for 70 seconds when no messages are being sent or received. After 70 seconds elapse, a permanent error condition results if the line is still not being used.



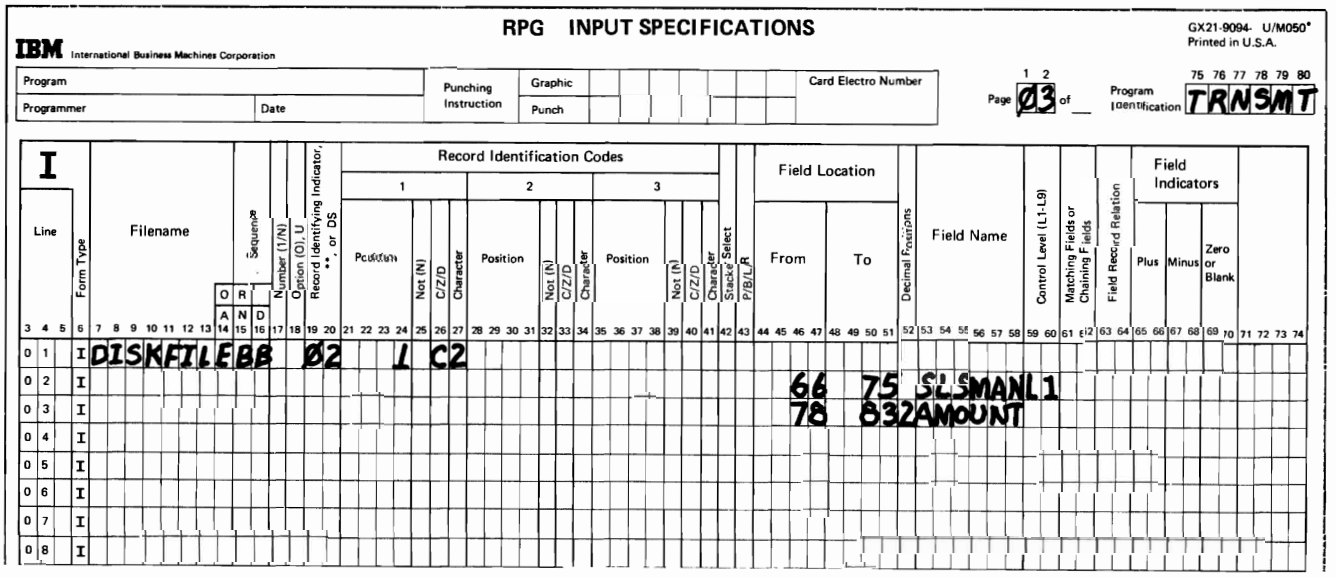


Figure 2-2 (Part 2 of 3). Transmit Program

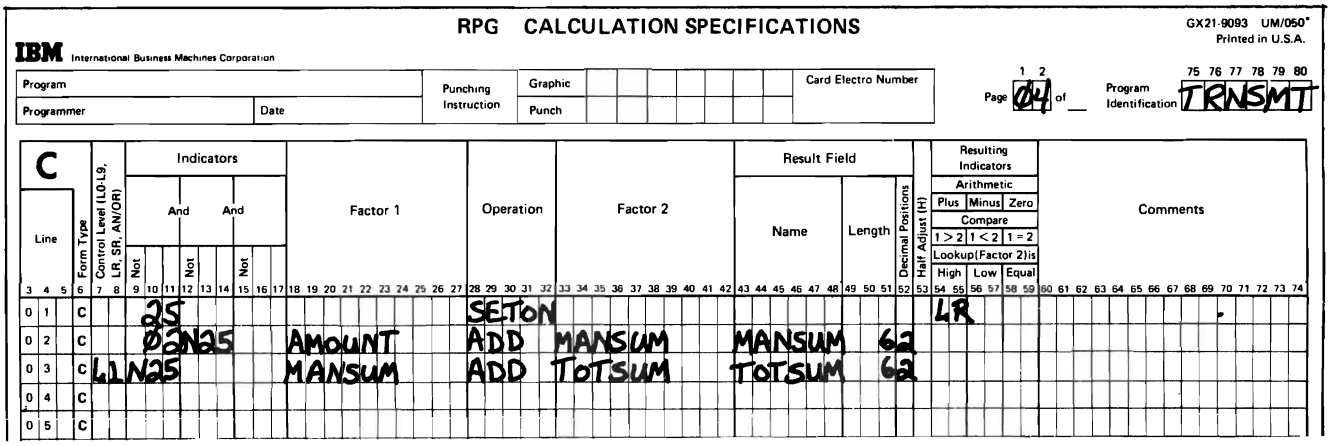
RPG Calculation Specifications (Figure 2-2, Part 3)

If a permanent error occurs, LR is set on to enter end-of-job processing.

RPG Output Specifications (Figure 2-2, Part 3)

Lines 01-04: If a permanent error occurs, a message printed at total time identifies the record being processed when the permanent error occurred. However, because of record blocking and the use of dual I/O areas, not all the records preceding the record identified were transmitted.

Lines 05-10: When on, the permanent-error indicator prevents the program from transmitting totals.



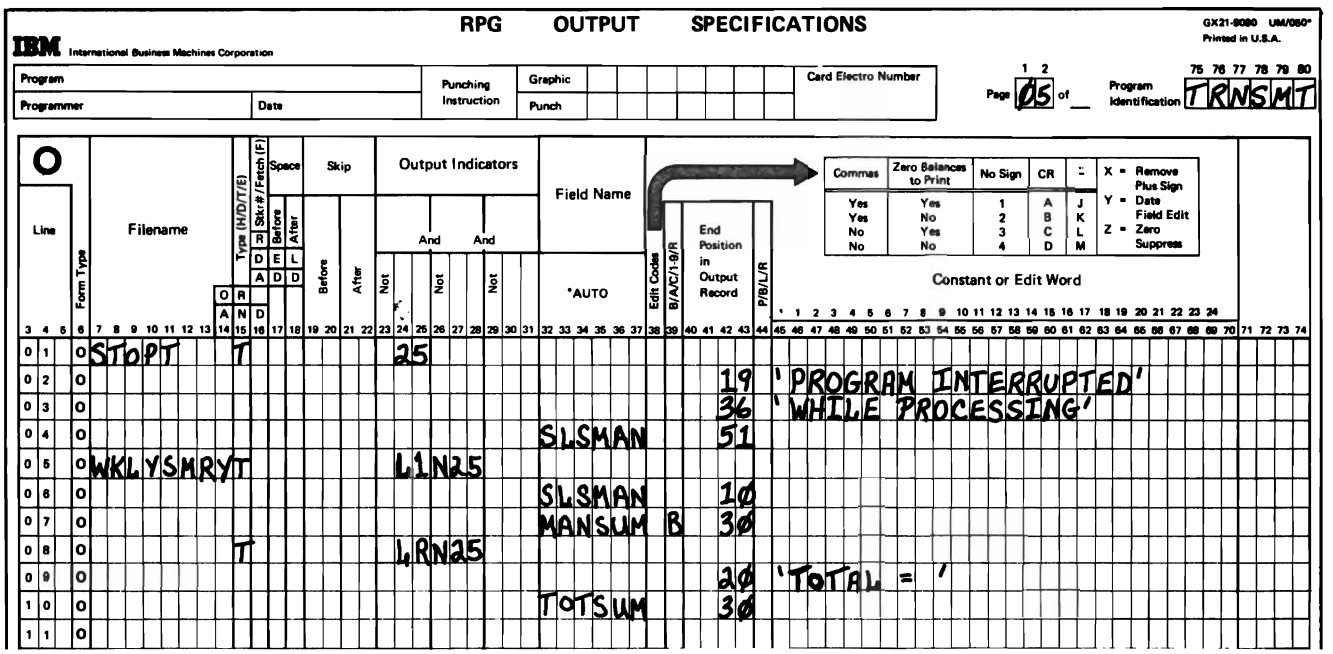


Figure 2-2 (Part 3 of 3). Transmit Program

Receiving Two Files, then Transmitting Two Files to a 3740

The IBM 3740 transmits two files to the System/34 and then receives two files from the System/34. The first file forms the input file BS1, the second file forms BS2. The data received as input to BS1 and BS2 is written by System/34 to the file, PRINTER. The disk file, FILEA, is then read. Records that start with a 1 are transmitted to the 3741 via file BS3. Records that start with a 2 are sent to the 3741 via file BS4. (All records with a 1 must precede any records that begin with a 2.) The job ends when the last record from the disk has been read.

Note: OVERRIDE must be run with MLTFL-Y, RCSP-1E.

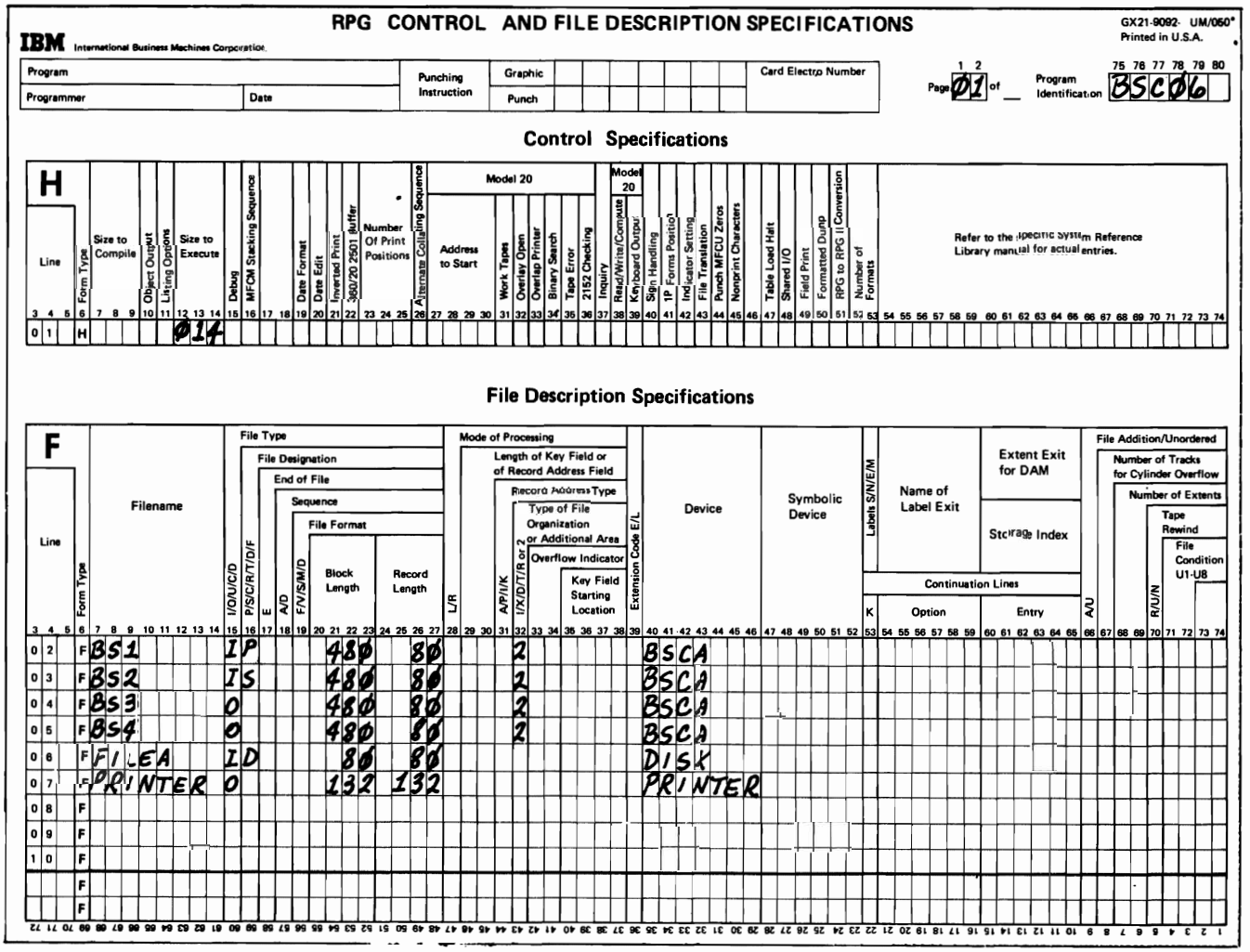


Figure 2-3 (Part 1 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

RPG TELECOMMUNICATIONS SPECIFICATIONS

GX21-9196-2 U/M050*
Printed in U.S.A.

IBM International Business Machine Corporation

Program		Punching Instruction	Graphic		Card Electro Number
Programmer	Date		Punch		

Page 02 of __ Program Identification BSC06

T

Line	Form Type	Filename	P/S/M Configuration	T/R Type of Station	T/R Type of Control	U/E Type of Code	Y/N Transparency	M/E/S/A/B	Switched .Dial Number	Identification: This Station S/E	Identification: Remote Station S/E	Remote Terminal	I = ITB	Permanent Error Indicator	Wait Time	Record Available Indicator	L Last File	Polling Characters	Addressing Characters	Remote Device	Reserved
01	T	BS1	P	R																	
02	T	BS2	P	R													L				
03	T	BS3	P	T																	
04	T	BS4	P	T																	
05	T																				
06	T																				
07	T																				
08	T																				
09	T																				
10	T																				
11	T																				

Figure 2-3 (Part 2 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

RPG INPUT SPECIFICATIONS

GX21-9094- U/M050*
Printed in U.S.A.

IBM International Business Machines Corporation

Program		Punching Instruction	Graphic		Card Electro Number
Programmer	Date		Punch		

Page 03 of __ Program Identification BSC06

I

Line	Form Type	Filename	Sequence (ORANDA)	Number (1/N)	Option (O), U	Record Identifying Indicator, **, or DS	Record Identification Codes 1: Position	1: Not (N)	1: C/Z/D	1: Character	2: Position	2: Not (N)	2: C/Z/D	2: Character	3: Position	3: Not (N)	3: C/Z/D	3: Character	Blank after	P/B/L/R	Field Location: From	Field Location: To	Decimal Positions	Field Name	Control Level (L1-L9)	Matching Fields or Chaining Fields	Field Record Relation	Field Indicators: Plus	Field Indicators: Minus	Field Indicators: Zero or Blank
01	I	BS1	A A			01																								
02	I																				1	80	INDATA							
03	I	BS2	A A			02																								
04	I																				1	80	INDATA							
05	I	FILEA	A A			03		1	C1																					
06	I																				1	80	RECORD							
07	I		A A			04		1	C2																					
08	I																				1	80	RECORD							
09	I																													
10	I																													
11	I																													
12	I																													
13	I																													
14	I																													
15	I																													
16	I																													
17	I																													
18	I																													
19	I																													

Figure 2-3 (Part 3 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

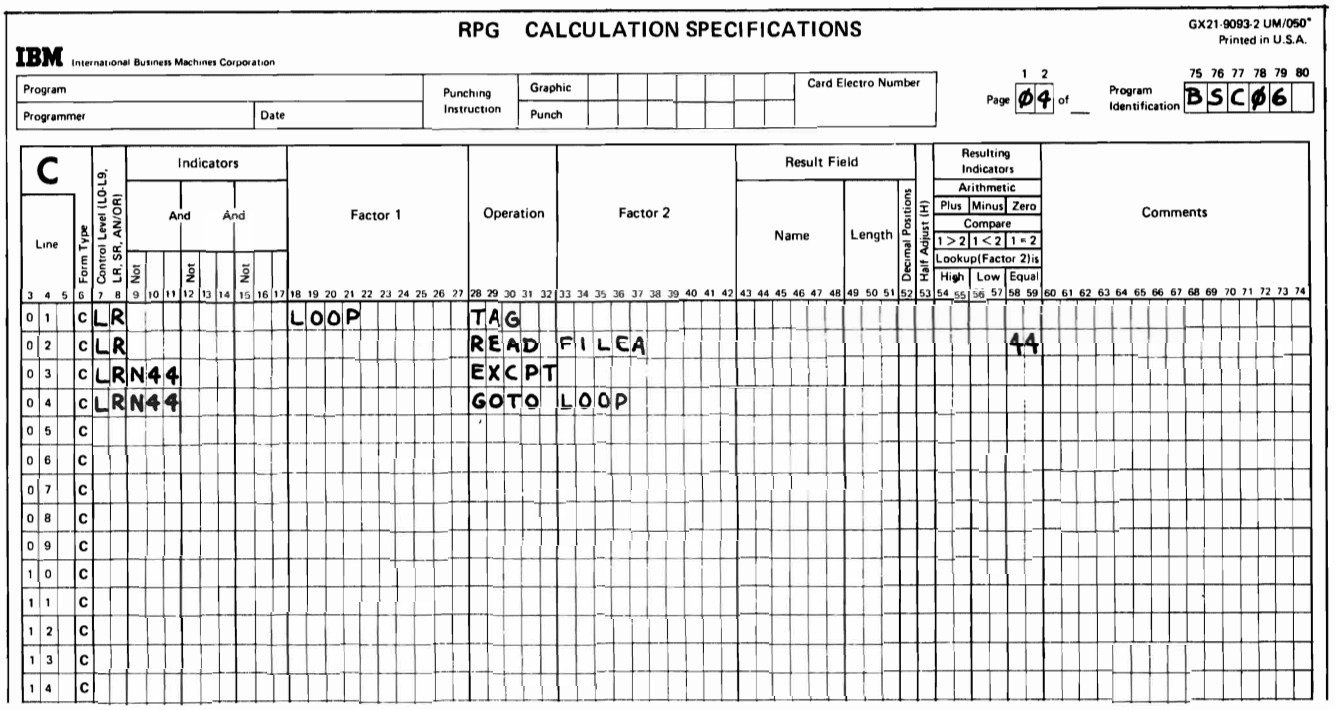


Figure 2-3 (Part 4 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

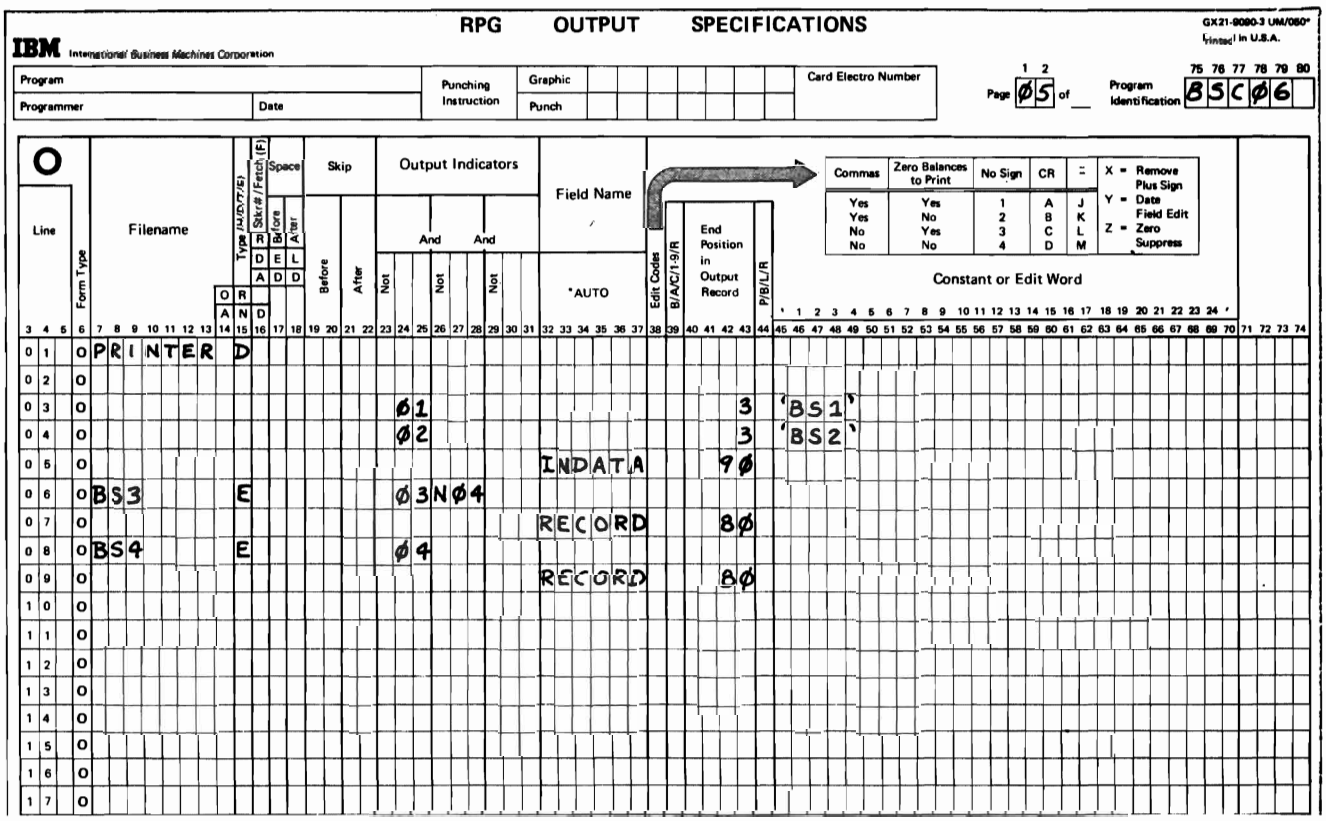


Figure 2-3 (Part 5 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

RPG File Description Specifications (Figure 2-4, Part 1)

System/34 reads student answers from DISK. The answers are transmitted to TCAM. MFC1TAB is a table file containing station IDs. ANSWER and RESULTS are used to transmit and receive, respectively. RESULTS must be a demand file (D in column 16) because the READ operation code is used to receive records to the file.

RPG Extension Specifications (Figure 2-4, Part 1)

MFC1TAB contains the tables TABME and TABYOU. TABME is the System/34 station ID; TABYOU is the TCAM station ID.

RPG CONTROL AND FILE DESCRIPTION SPECIFICATIONS

IBM International Business Machines Corporation — GX21-9092- UM/050* Printed in U.S.A.

Program | Punching Instruction | Graphic | Card Electro Number
Programmer | Date | Punch

Page 01 of __ | Program Identification S34TCA

Control Specifications

H

Line	Form Type	Size to Compile	Object Output	Listing Options	Size to Execute	Debug	MFCM Stacking Sequence	Date Format	Date Edit	Inverted Print	360/20 2501 Buffer	Number Of Print Positions	Alternate Collating Sequence	Model 20: Address to Start	Model 20: Work Tapes	Model 20: Overlay Open	Model 20: Overlap Printer	Model 20: Binary Search	Model 20: Tape Error	Model 20: 2152 Checking Inquiry	Model 20: Read/Write/Compute Keyboard Output	Sign Handling	1P Forms Position	Indicator Setting	File Translation	Punch MFCU Zeros	Nonprint Characters	Table Load Halt	Shared I/O	Field Print	Formatted Dump	RPG to RPG II Conversion	Number of Formats	Refer to the specific System Reference Library manual for actual entries.
3 4 5	6	7 8 9	10	11	12 13 14	15	16	17 18	19	20	21	22 23 24 25	26	27 28 29 30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52 53	54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
0 1	H				Ø14																														

File Description Specifications

F

Line	Form Type	Filename	File Type: I/O/U/C/D	File Designation: P/S/C/R/T/D/F	End of File: E	Sequence: A/D	File Format: F/V/S/M/D	Block Length	Record Length	Mode of Processing: L/R	Length of Key Field or of Record Address Field	Record Address Type: A/P/I/K	Type of File Organization or Additional Area: I/X/D/T/R or 2	Overflow Indicator	Key Field Starting Location	Extension Code E/L	Device	Symbolic Device	Labels S/N/E/M	Name of Label Exit	Extent Exit for DAM	Storage Index	Continuation Lines: K	Continuation Lines: Option	Continuation Lines: Entry	File Addition/Unordered: A/U	Number of Tracks for Cylinder Overflow	Number of Extents	Tape Rewind: R/U/N	File Condition U1-U8
3 4 5	6	7 8 9 10 11 12 13 14	15	16	17	18	19	20 21 22 23	24 25 26 27	28	29 30	31	32	33 34	35 36 37 38	39	40 41 42 43 44 45 46	47 48 49 50 51 52	53	54 55 56 57 58 59	60 61 62 63 64 65	66	67	68 69	70	71 72	73 74
0 2	F	DISKIN	IP					96	96								DISK Ø													
0 3	F	MFC1TAB	IT					4	4							E	DISK													
0 4	F	ANSWERS	O					7Ø	7Ø								BSCA													
0 5	F	RESULTS	ID					25	25								BSCA													
0 6	F	PRINT	O					132	132								PRINTER													

RPG EXTENSION AND LINE COUNTER SPECIFICATIONS

IBM International Business Machines Corporation — GX21-9091 UM/050* Printed in U.S.A.

Program | Punching Instruction | Graphic | Card Electro Number
Programmer | Date | Punch

Page Ø2 of __ | Program Identification S34TCA

Extension Specifications

E

Line	Form Type	Record Sequence of the Chaining File	Number of the Chaining Field	From Filename	To Filename	Table or Array Name	Number of Entries Per Record	Number of Entries Per Table or Array	Length of Entry	P/B/L/R	Decimal Positions	Sequence (A/D)	Table or Array Name (Alternating Format)	Length of Entry	P/B/L/R	Decimal Positions	Sequence (A/D)	Comments
3 4 5	6	7 8	9 10	11 12 13 14 15 16 17 18	19 20 21 22 23 24 25 26	27 28 29 30 31 32	33 34 35	36 37 38 39	40 41 42	43	44	45	46 47 48 49 50 51	52 53 54	55	56	57	58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
0 1	E			MFC1TAB		TABME	1	1	2				TABYOU	2				
0 2	E																	

Figure 2-4 (Part 1 of 4). System/34 to TCAM

RPG Telecommunications Specifications (Figure 2-4, Part 2)

ANSWER and RESULTS are transmit and receive files, respectively. Both are part of a switched network.

RPG Input Specifications (Figure 2-4, Part 2)

DISKIN provides data from a disk file. RESULTS provides data from TCAM.

RPG TELECOMMUNICATIONS SPECIFICATIONS

IBM International Business Machine Corporation

GX21-9196- U/M050* Printed in U.S.A.

Program		Punching Instruction	Graphic								Card Electro Number
Programmer	Date		Punch								

Page 03 of __ Program Identification S34TCA

Line	Form Type	Filename	P/S/M Configuration	T/R Type of Station	T/B Type of Control	U/E Type of Code	Y/N Transparency	M/E/S/A/B	Switched: Dial Number	Identification: This Station S/E	This Station	Remote Station S/E	Remote Station	Remote Terminal	I = ITB	Permanent Error Indicator	Wait Time	Record Available Indicator	L Last File	Polling Characters	Addressing Characters	Remote Device	Reserved
01	T	ANSWERS	ST					M			STABME		STABYOU				60						
02	T	RESULTS	SR					M			STABME		STABYOU				60						
03	T																						
04	T																						

RPG INPUT SPECIFICATIONS

IBM International Business Machines Corporation

GX21-9094- U/M050* Printed in U.S.A.

Program		Punching Instruction	Graphic								Card Electro Number
Programmer	Date		Punch								

Page 04 of __ Program Identification S34TCA

Line	Form Type	Filename	Sequence	Number (1/N)	Option (O), U	Record Identifying Indicator, *, or DS	Record Identification Codes 1: Position	1: Not (N)	1: C/Z/D	1: Character	2: Position	2: Not (N)	2: C/Z/D	2: Character	3: Position	3: Not (N)	3: C/Z/D	3: Character	Stacker Select	P/B/L/R	Field Location: From	To	Decimal Positions	Field Name	Control Level (L1-L9)	Matching Fields or Chaining Fields	Field Record Relation	Field Indicators: Plus	Minus	Zero or Blank
01	I	DISKIN	AA			01		1	CX																					
02	I																				4	20		NAME						
03	I																				27	28	0	NUMQES						
04	I																				30	79		ANS						
05	I	RESULTS	BB			02		1	CA																					
06	I																				2	18		NAME						
07	I																				19	20	0	NUM						
08	I																				21	25		RESULTS						
09	I																													
10	I																													
11	I																													

Figure 2-4 (Part 2 of 4). System/34 to TCAM

RPG Calculation Specifications (Figure 2-4, Part 3)

System/34 records the number of tests transmitted to TCAM for correction. The tests are transmitted by exception output to BSC. The system receives (using the READ operation code) results from TCAM until the system reaches end of file on RESULTS. System/34 sets indicators off at end-of-file time to prepare for the next RPG cycle.

RPG Output Specifications (Figure 2-4, Part 4)

System/34 prints data by exception output as the data is read from the disk and received by BSC. The system transmits data to TCAM by exception output as the data is read from the disk. The system signals TCAM (with an E) when no more records are sent.

RPG CALCULATION SPECIFICATIONS

GX21-9093 UM/050* Printed in U.S.A.

IBM International Business Machines Corporation

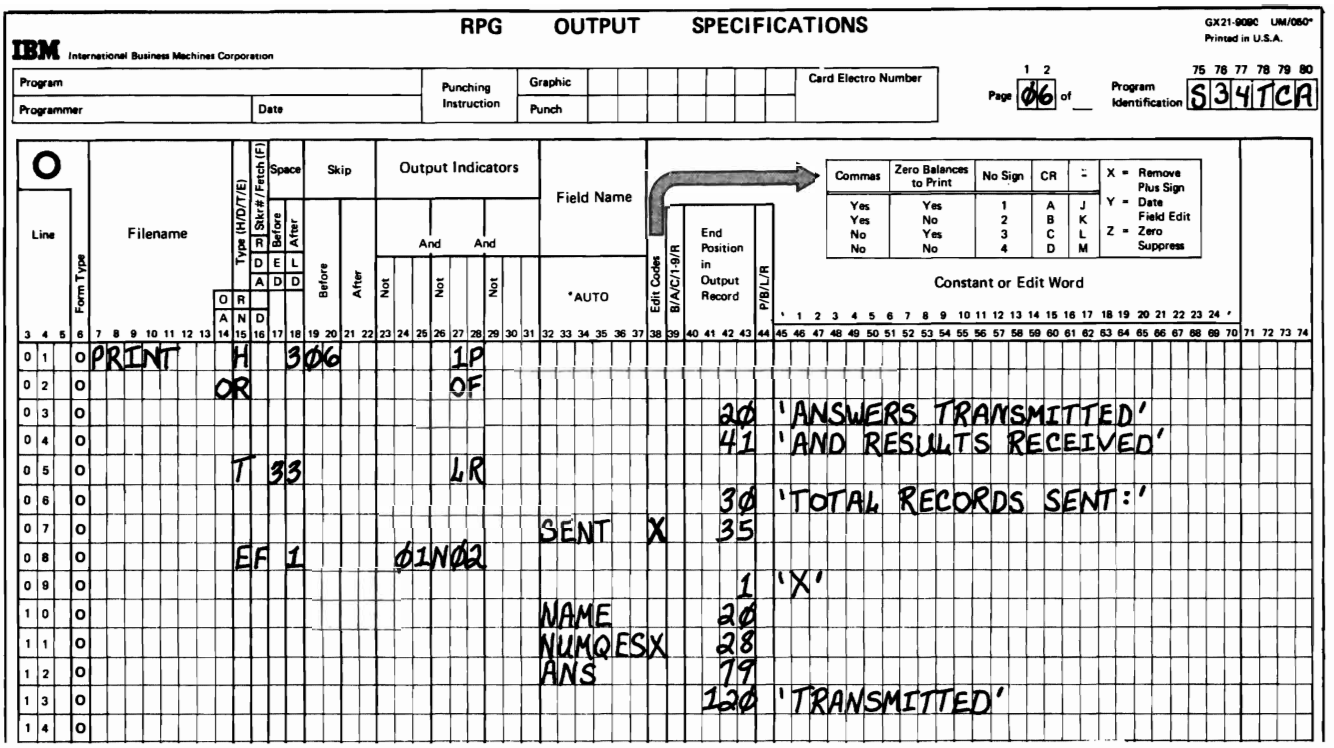
Program | Punching Instruction | Graphic | Card Electro Number

Programmer | Date | Punch

Page 05 of __ | Program Identification S34TCA

Line	Form Type	Control Level (L0-L9, LR, SR, AN/OR)	Indicators: And / Not	Factor 1	Operation	Factor 2	Result Field: Name	Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Arithmetic Plus	Arithmetic Minus	Arithmetic Zero	Compare 1>2	Compare 1<2	Compare 1=2	Lookup(Factor 2)is High	Lookup(Factor 2)is Low	Lookup(Factor 2)is Equal	Comments
01	C		01	SENT	ADD 1		SENT	30												
02	C		01		EXCPT															
03	C			LOOP	TAG															
04	C		01		READ RESULTS														72	
05	C		02N72		EXCPT															
06	C		02N72		GOTO LOOP															
07	C		72		SETOF												0272			
08	C																			
09	C																			
10	C																			
11	C																			
12	C																			
13	C																			
14	C																			

Figure 2-4 (Part 3 of 4). System/34 to TCAM



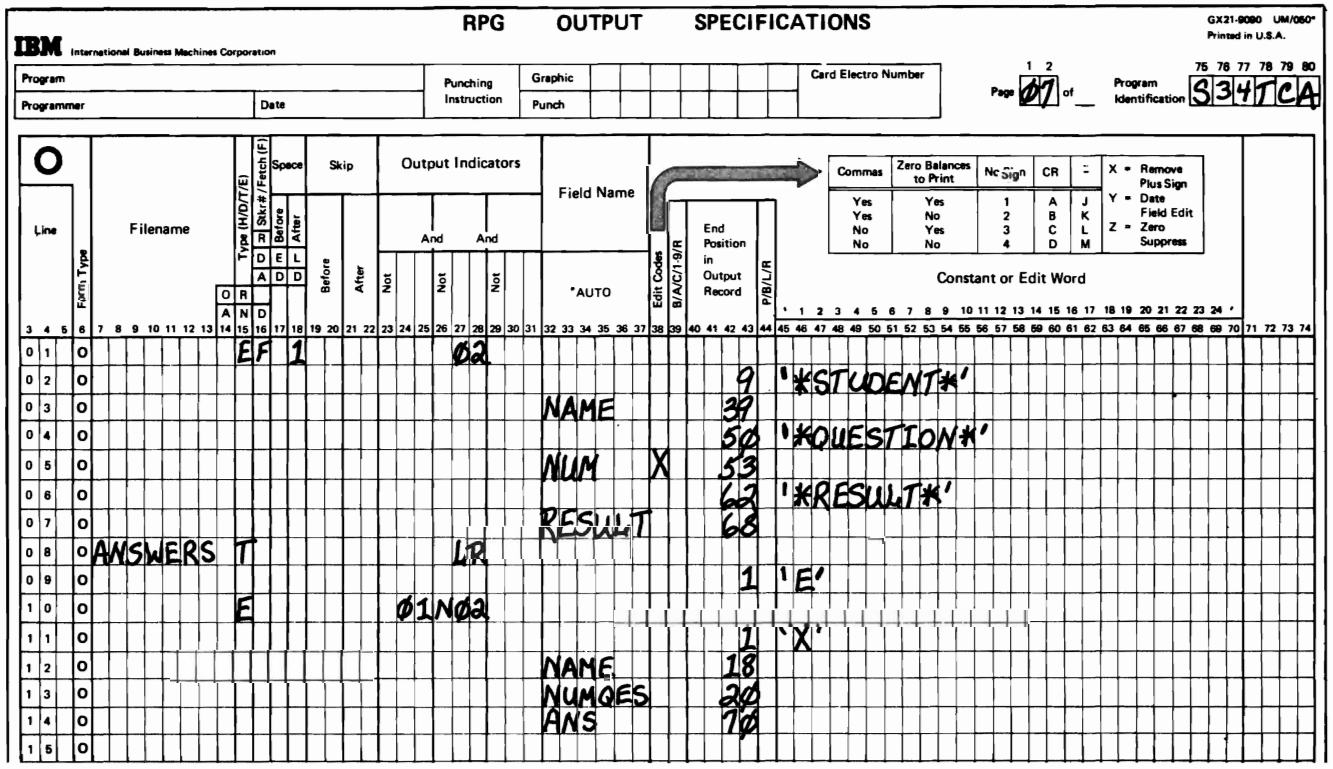


Figure 2-4 (Part 4 of 4). System/34 to TCAM

Transmit Interspersed with Receive–No Conversational Reply

In the following program, a file containing information about a student test is read and transmitted to a program for processing. During transmission, the other station might interrupt to send back results. When this occurs, RPG turns on record available indicator 04. The input file is then read and the incoming data is printed on the printer. System/34 then resumes transmission.

RPG File Description Specifications (Figure 2-5, Part 1)

Column	Description
7-14	RESULTS and GRADES are BSC files.
15	RESULTS receives data; therefore, it is an input file. GRADES is transmitted and is an output file.
16	To receive data intermittently, a BSC file must be a demand file.
17	End of file on RESULTS does not determine end-of-job time.
19	BSC files always have a fixed-length format.
20-27	Neither BSC file is blocked.
40-46	BSCA is the device for BSC files.

RPG CONTROL AND FILE DESCRIPTION SPECIFICATIONS

GX21-9092- UM/050* Printed in U.S.A.

IBM International Business Machines Corporation

Program | Punching Instruction | Graphic / Punch | Card Electro Number

Programmer | Date

Page 01 of __ | Program Identification 75 76 77 78 79 80: TRTR01

Control Specifications

H

Line	Form Type	Size to Compile	Object Output	Listing Options	Size to Execute	Debug	MFCM Stacking Sequence	Date Format	Date Edit	Inverted Print	360/20 2501 Buffer	Number Of Print Positions	Alternate Collating Sequence	Model 20: Address to Start	Work Tapes	Overlay Open	Overlap Printer	Binary Search	Tape Error	2152 Checking	Inquiry	Model 20: Read/Write/Compute Keyboard Output	Sign Handling	1P Forms Position	Indicator Setting	File Translation	Punch MFCU Zeros	Nonprint Characters	Table Load Halt	Shared I/O	Field Print	Formatted Dump	RPG to RPG II Conversion	Number of Formats	Refer to the specific System Reference Library manual for actual entries.
01	H				014																													

File Description Specifications

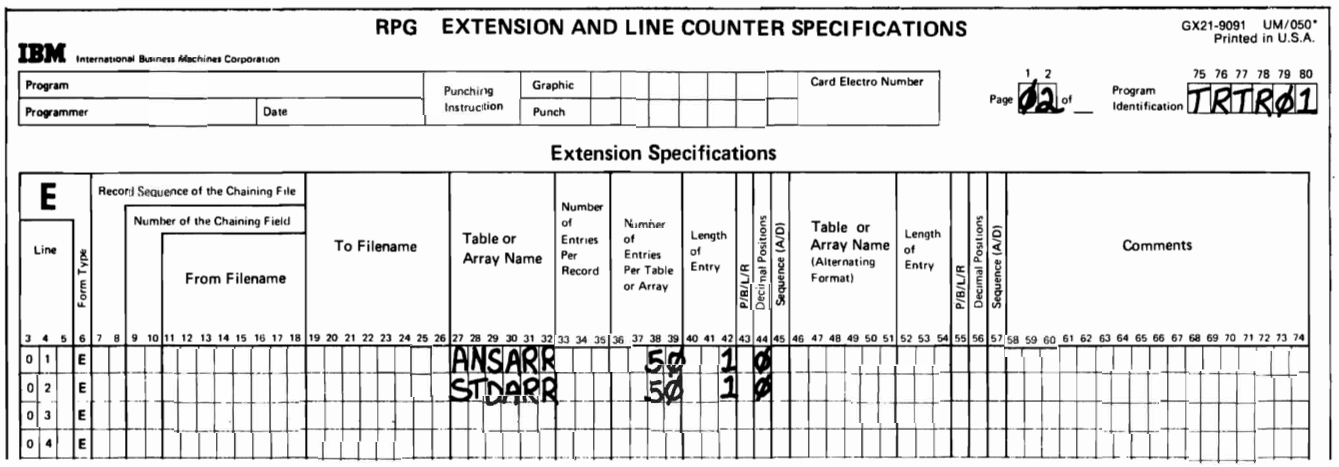
F

Line	Form Type	Filename	File Type: I/O/U/C/D	File Designation: P/S/C/R/T/D/F	End of File Sequence: E	Record Address Type: A/D	File Format: F/V/S/M/D	Block Length	Record Length	Mode of Processing: L/R	Length of Key Field or of Record Address Field	Type of File Organization or Additional Area: A/P/I/K	Record Address Type: I/X/D/T/R or 2	Overflow Indicator	Key Field Starting Location	Extension Code E/L	Device	Symbolic Device	Labels S/N/E/M	Name of Label Exit	Extent Exit for DAM	Storage Index	Continuation Lines: K	Option	Entry	File Addition/Unordered: A/U	Number of Tracks for Cylinder Overflow	Number of Extents	Tape Rewind: R/U/N	File Condition U1-U8
02	F	STUDENT	I	P			F	96	96								DISK													
03	F	RESULTS	I	D			F	80	80								BSCA													
04	F	GRADES	O				F	80	80				2				BSCA													
05	F	PRINT	O				F	132	132								PRINTER													
06	F																													
07	F																													
08	F																													

Figure 2-5 (Part 1 of 5). Transmit Interspersed with Receive

RPG Telecommunications Specifications (Figure 2-5, Part 2)

Column	Description
7-14	RESULTS and GRADES are the BSC files for this program.
15	This station is part of a switched network (S).
16	RESULTS is an input file and receives data (R). GRADES is an output file and is transmitted (T).
17	Polling is not used, so column 17 is blank.
18-19	EBCDIC (E) and the transparency feature (Y) are both used.
20	Automatic answer is used by this station.
32-47	Explicit station identification sequences are given. Station IDs help ensure data security on the switched network.
53-54	The permanent-error indicator used is 44.
55-57	The data communications line is kept open for 70 seconds when no messages are being sent or received. After 70 seconds elapse, a permanent error condition results if the line still is not being used.
58-59	The record available indicator is 04; it is set on when the other system sends an RVI (reverse interrupt). This indicates it is ready to send a record to be received by RESULTS.



RPG EXTENSION AND LINE COUNTER SPECIFICATIONS

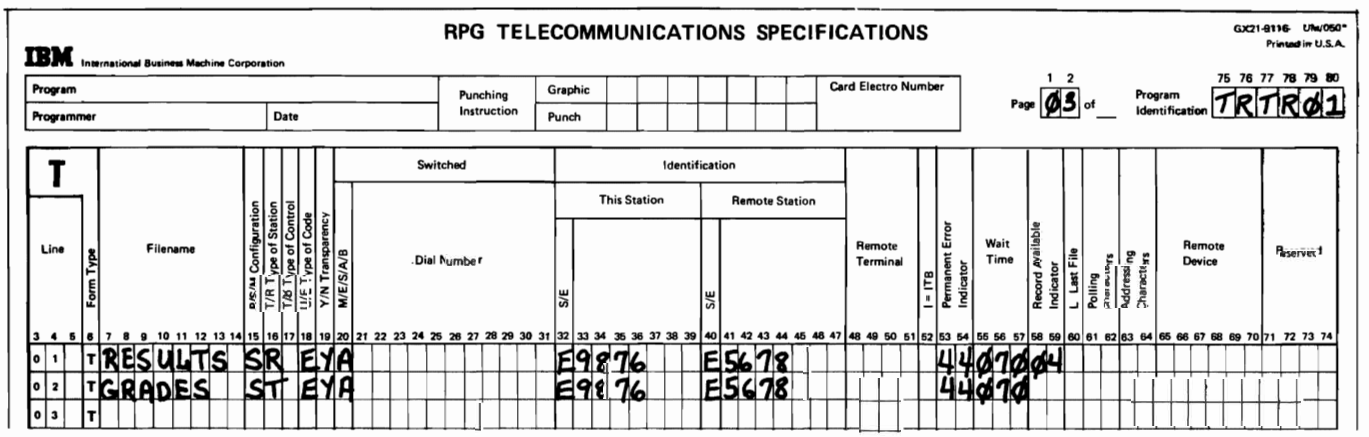
GX21-9091 UM/050* Printed in U.S.A.

IBM International Business Machines Corporation

Program | Programmer | Date | Punching Instruction | Graphic | Punch | Card Electro Number | Page 02 of __ | Program Identification TRTR01

Extension Specifications

Line	Form Type	Record Sequence of the Chaining File	Number of the Chaining Field	From Filename	To Filename	Table or Array Name	Number of Entries Per Record	Number of Entries Per Table or Array	Length of Entry	P/B/L/R	Decimal Positions	Sequence (A/D)	Table or Array Name (Alternating Format)	Length of Entry	P/B/L/R	Decimal Positions	Sequence (A/D)	Comments
01	E					ANSARR		50	1		0							
02	E					STDARR		50	1		0							
03	E																	
04	E																	



RPG TELECOMMUNICATIONS SPECIFICATIONS

GX21-9116 UM/050* Printed in U.S.A.

IBM International Business Machine Corporation

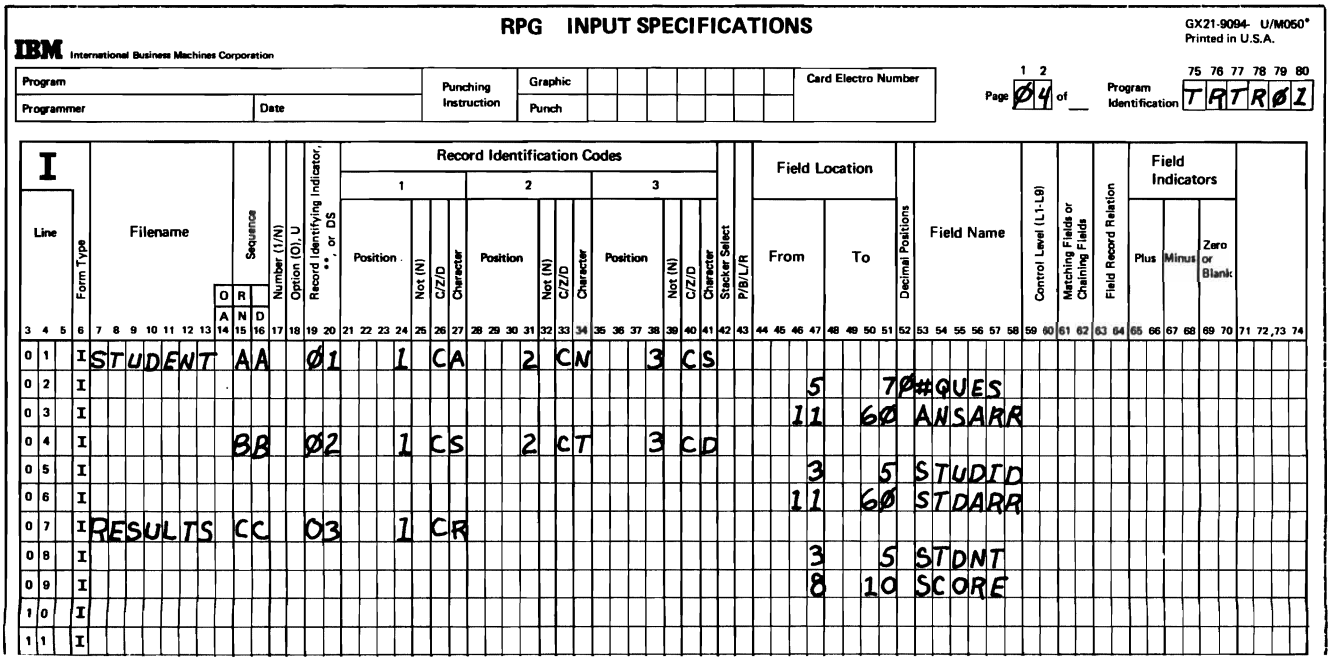
Program | Programmer | Date | Punching Instruction | Graphic | Punch | Card Electro Number | Page 03 of __ | Program Identification TRTR01

T

Line	Form Type	Filename	BSCM Configuration	T/R Type of Station	T/R Type of Control	T/R Type of Code	Y/N Transparency	M/E/S/A/B	Switched: Dial Number	Identification: This Station S/E	Identification: Remote Station S/E	Remote Terminal	I = ITB	Permanent Error Indicator	Wait Time	Record Available Indicator	L Last File	Polling Characters	Addressing Characters	Remote Device	Reserved
01	T	RESULTS	S	R			E	Y	A	E9876	E5678			44	07	00	4				
02	T	GRADES	S	T			E	Y	A	E9876	E5678			44	07	0					
03	T																				

Figure 2-5 (Part 2 of 5). Transmit Interspersed with Receive

Lines 07-09: RESULTS receives records from the remote station. These records contain a student number and the student's score. Student scores are calculated by the remote station from the data transmitted from GRADES.



RPG INPUT SPECIFICATIONS

IBM International Business Machines Corporation

GX21-9094- U/M050* Printed in U.S.A.

Program | Programmer | Date | Punching Instruction | Graphic | Punch | Card Electro Number

Page 04 of __ | Program Identification TATR01

I

Line	Form Type	Filename	Sequence (OR AND)	Number (1/N)	Option (O), U	Record Identifying Indicator, **, or DS	Record Identification Codes 1: Position	1: Not (N)	1: C/Z/D	1: Character	2: Position	2: Not (N)	2: C/Z/D	2: Character	3: Position	3: Not (N)	3: C/Z/D	3: Character	Stacker Select	P/B/L/R	Field Location From	Field Location To	Decimal Positions	Field Name	Control Level (L1-L9)	Matching Fields or Chaining Fields	Field Record Relation	Field Indicators Plus	Minus	Zero or Blank
01	I	STUDENT	AA			01	1		CA	2		CN	3		CS															
02	I																				5	7	0	#QUES						
03	I																				11	6	0	ANSARR						
04	I		BB			02	1		CS	2		CT	3		CD															
05	I																				3	5		STUDID						
06	I																				11	6	0	STDARR						
07	I	RESULTS	CC			03	1		CR																					
08	I																				3	5		STDNT						
09	I																				8	1	0	SCORE						
10	I																													
11	I																													

Figure 2-5 (Part 3 of 5). Transmit Interspersed with Receive

RPG Calculation Specifications (Figure 2-5, Part 4)

The calculation sequence obtains the scores computed by the remote station from GRADES and makes them available for exception output to PRINT.

Lines 01-06: Loop 1 is executed during detail time; that is, before end of file is reached on STUDENTS. When the remote station causes the record available indicator (04) to be set on, the READ operation accepts a record from the remote station and places it in RESULTS. Indicator 10 is set on when READ encounters an end-of-file condition. Until indicator 04 is set off and indicator 10 is set on, records are placed in RESULTS and are available for exception output to PRINT. Loop 1 can be entered or reentered anytime 04 is set on, except after end of file is reached on STUDENTS (indicator 10 is set off). Therefore, after one group of records is read by the receiving station, the receiving station must set off the end-of-file indicator (10) so that the next group of records can be read.

Lines 07-12: Loop 2 is executed during total time; that is, after end of file is reached on STUDENTS and the LR indicator is set on. Loop 2 processes records for RESULTS in the same way as loop 1 with one difference: Loop 2 always compares the student number received by RESULTS (STDNT) to the last student number transmitted (STUDID). Loop 2 continues execution until these numbers are equal. This ensures that the results for all student records transmitted are received.

If a permanent error occurs during a transmit or receive operation, LR and 10 are set on to enter end-of-job processing. Indicator 09 is then set on to exit from the total time loop.

RPG CALCULATION SPECIFICATIONS

GX21-9093 UM/050* Printed in U.S.A.

IBM International Business Machines Corporation

Program | Punching Instruction | Graphic | Card Electro Number

Programmer | Date | Punch

Page 05 of __ | Program Identification TRTR01

Line	Form Type	Control Level (L0-L9, LR, SR, AN/OR)	Indicators: And / Not	Factor 1	Operation	Factor 2	Result Field: Name	Length	Decimal Positions	Half Adjust (H)	Resulting Indicators: Arithmetic (Plus / Minus / Zero); Compare (1>2 / 1<2 / 1=2); Lookup(Factor 2)is (High / Low / Equal)	Comments
01	C			LOOP1	TAG							
02	C		04		READ RESULTS						10	
03	C		03N10		EXCPT							
04	C		44		SETON						LR10	
05	C		03N10		GOTO LOOP1							
06	C				SETOF						10	
07	C	LR		LOOP2	TAG							
08	C	LR	04		READ RESULTS						10	
09	C	LR	03N10		EXCPT							
10	C	LR	44		SETOF						09	
11	C	LRW	44	STDNT	COMP STUDID						0909	
12	C	LR	09		GOTO LOOP2							
13	C											
14	C											
15	C											
16	C											
17	C											

Figure 2-5 (Part 4 of 5). Transmit Interspersed with Receive

RPG Output Specifications (Figure 2-5, Part 5)

Lines 01-06: GRADES is transmitted to the remote stations.

Lines 07-08: A permanent error causes an error message to be printed.

Lines 09-13: Records received by RESULTS are printed.

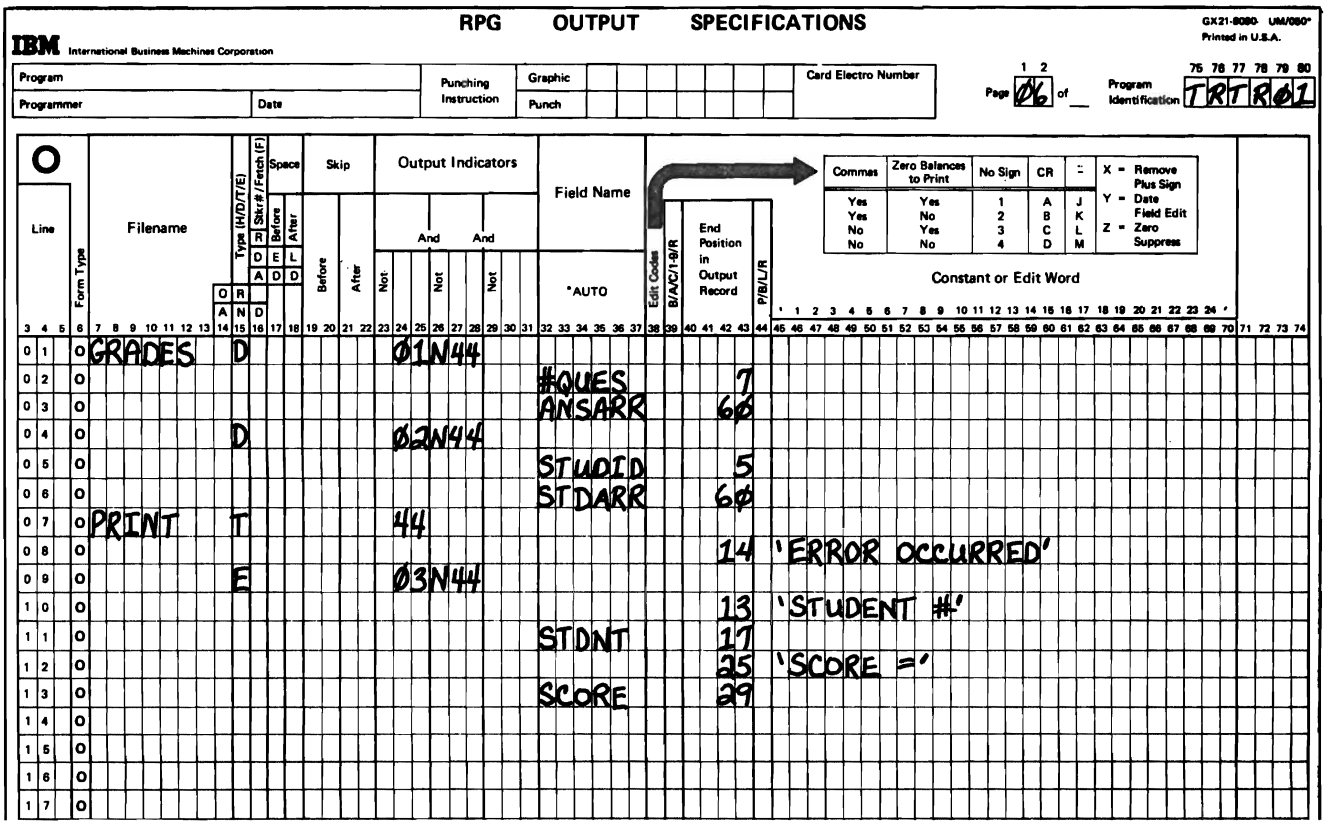
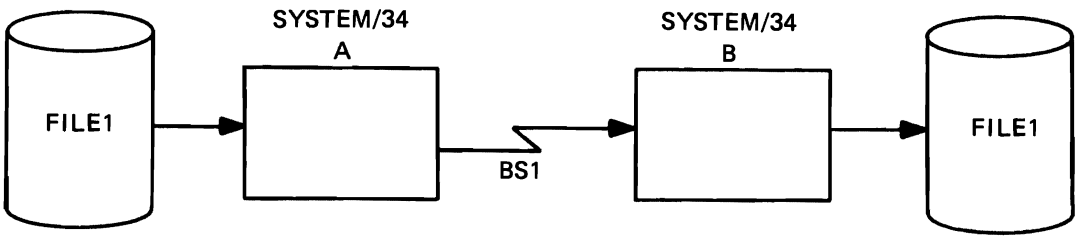


Figure 2-5 (Part 5 of 5). Transmit Interspersed with Receive

System/34 to System/34

In the following program, System/34 A performs a point-to-point transmission of one file to receiving System/34 B. The data from input file FILE1 on System/34 A is sent to System/34 B via the output file BS1 (refer to FIGURE 2-6). FILE1 is received by System/34 B by the input file BS1 (refer to FIGURE 2-7). BS1 is saved on disk in the output file FILE1.



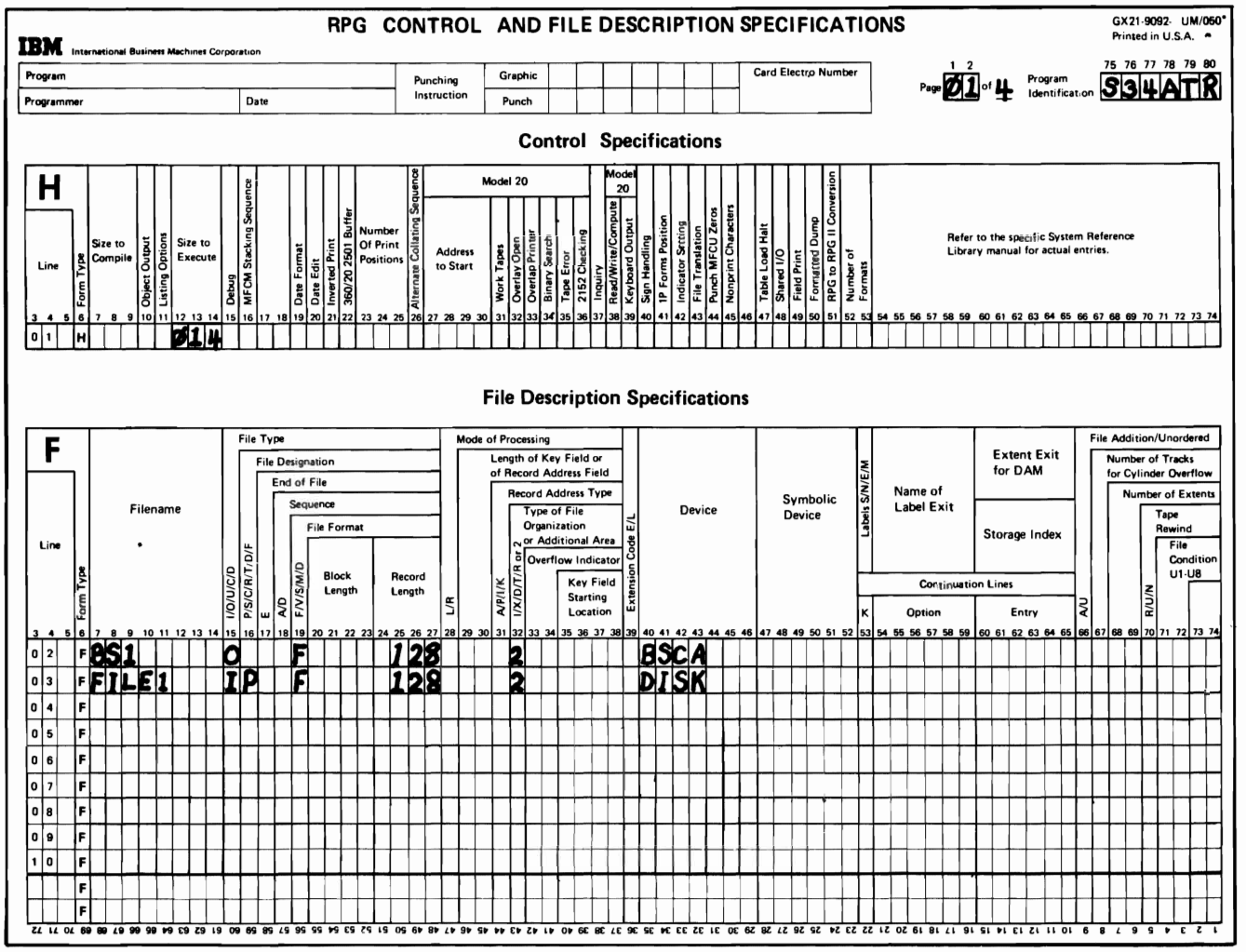


Figure 2-6 (Part 1 of 4). System/34 A Transmit Program

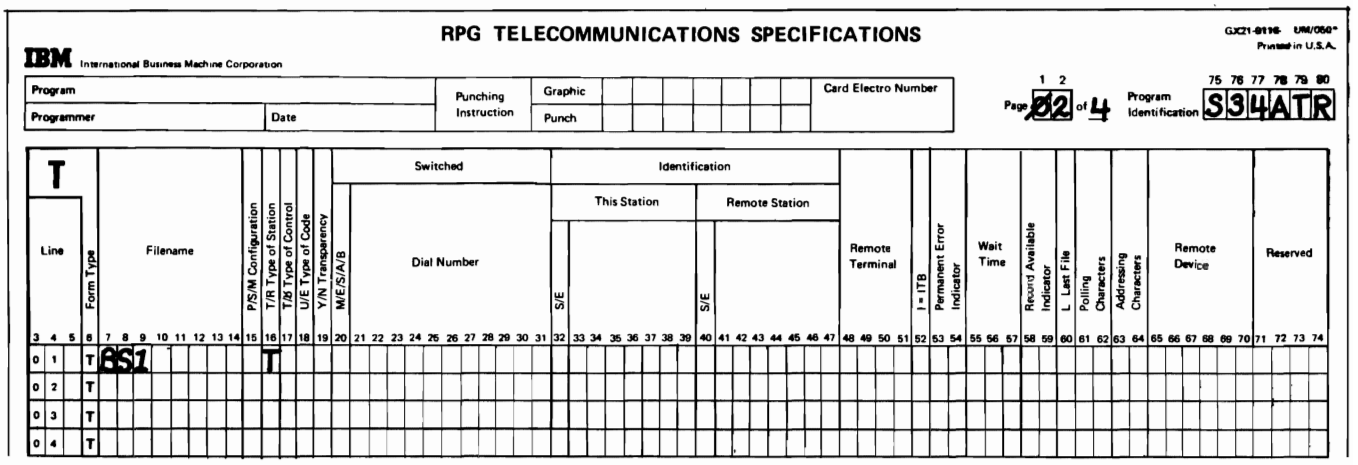


Figure 2-6 (Part 2 of 4). System/34 A Transmit Program

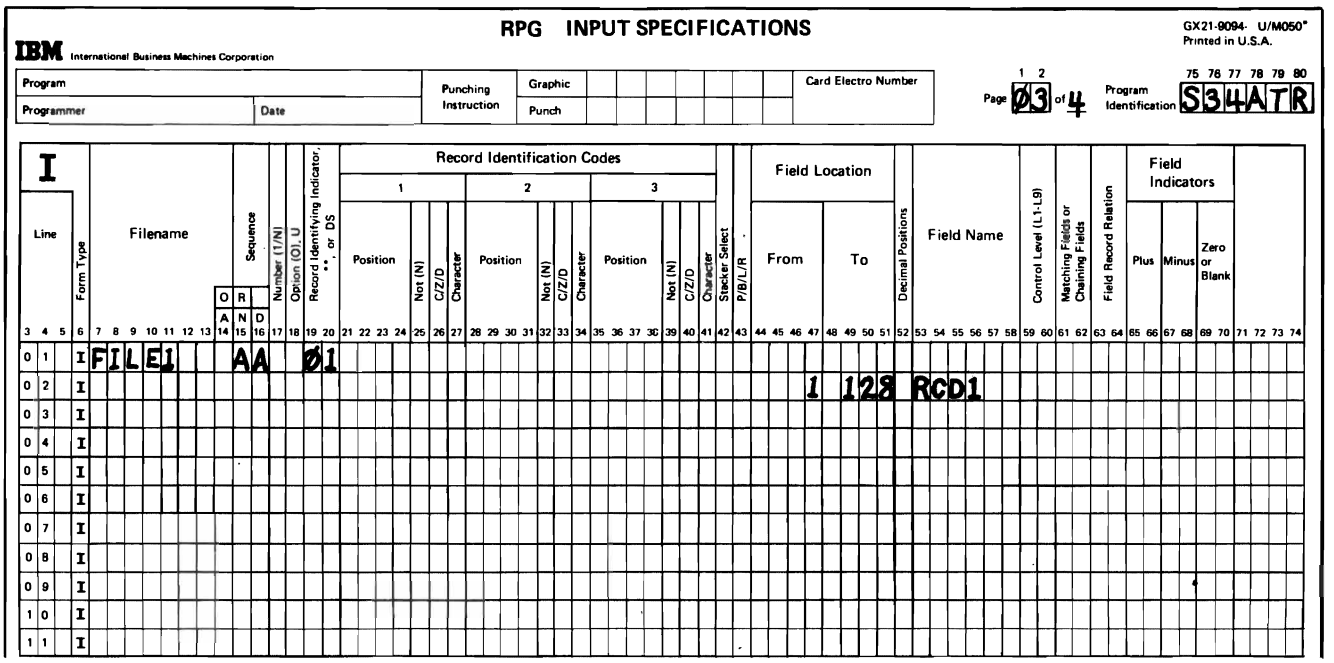


Figure 2-6 (Part 3 of 4). System/34 A Transmit Program

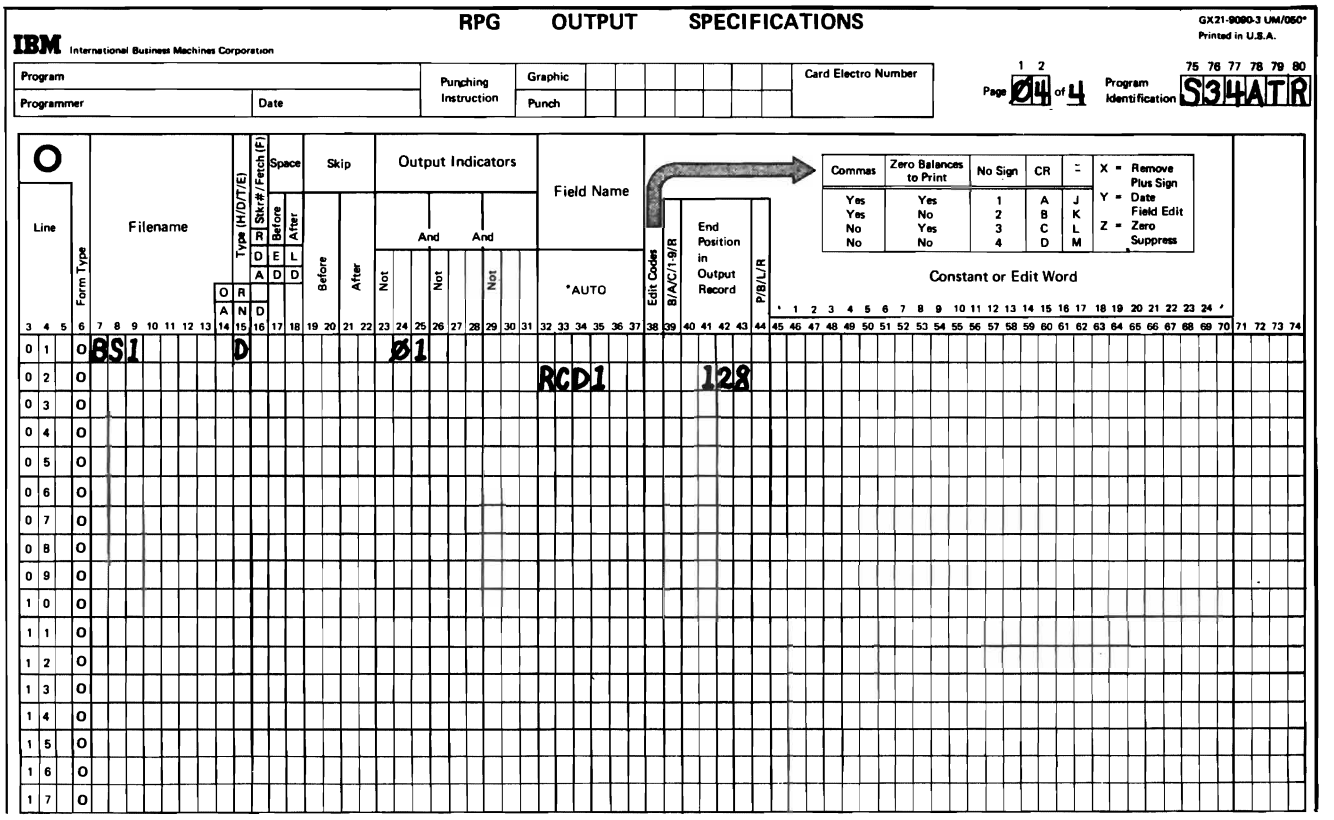


Figure 2-6 (Part 4 of 4). System/34 A Transmit Program

RPG CONTROL AND FILE DESCRIPTION SPECIFICATIONS

GX21-9092- UM/050* Printed in U.S.A.

IBM International Business Machines Corporation

Program | Punching Instruction | Graphic | Card Electro Number

Programmer | Date | Punch

Page 01 of 4 | Program Identification S34BRC

Control Specifications

H

Line	Form Type	Size to Compile	Object Output	Listing Options	Size to Execute	Debug	MFCM Stacking Sequence	Date Format	Date Edit	Inverted Print	360/20 2501 Buffer	Number Of Print Positions	Alternate Collating Sequence	Model 20: Address to Start	Work Tapes	Overlay Open	Overlap Printer	Binary Search	Tape Error	2152 Checking	Model 20: Inquiry	Read/Write/Compute	Keyboard Output	Sign Handling	1P Forms Position	Indicator Setting	File Translation	Punch MFCU Zeros	Nonprint Characters	Table Load Halt	Shared I/O	Field Print	Formatted Dump	RPG to RPG II Conversion	Number of Formats	Refer to the specific System Reference Library manual for actual entries.
01	H				014																														

File Description Specifications

F

Line	Form Type	Filename	File Type	File Designation	End of File	Sequence	File Format	Block Length	Record Length	Mode of Processing	Length of Key Field or of Record Address Field	Record Address Type	Type of File Organization or Additional Area	Overflow Indicator	Key Field Starting Location	Extension Code E/L	Device	Symbolic Device	Labels S/N/E/M	Name of Label Exit	Extent Exit for DAM	Storage Index	Continuation Lines: K	Continuation Lines: Option	Continuation Lines: Entry	A/U	File Addition/Unordered	Number of Tracks for Cylinder Overflow	Number of Extents	Tape Rewind	R/U/N	File Condition U1-U8
			I/O/U/C/D	P/S/C/R/T/D/F	E	A/D	F/V/S/M/D			L/R		A/P/I/K	I/X/D/T/R or 2																			
02	F	BS1	I	P			F		128				2				BSCA																
03	F	FILE1	O				F		128				2				DISK																
04	F																																
05	F																																
06	F																																
07	F																																
08	F																																
09	F																																
10	F																																
	F																																
	F																																

Figure 2-7 (Part 1 of 4). System/34 B Receive Program

RPG TELECOMMUNICATIONS SPECIFICATIONS

GX21-9196- UM/050* Printed in U.S.A.

IBM International Business Machine Corporation

Program | Punching Instruction | Graphic | Card Electro Number

Programmer | Date | Punch

Page 02 of 4 | Program Identification S34BRC

T

Line	Form Type	Filename	P/S/M Configuration	T/R Type of Station	T/B Type of Control	U/E Type of Code	Y/N Transparency	M/E/S/A/B	Switched: Dial Number	Identification: This Station S/E	Identification: Remote Station S/E	Remote Terminal	I = ITB	Permanent Error Indicator	Wait Time	Record Available Indicator	L Last File	Polling Characters	Addressing Characters	Remote Device	Reserved
01	T	BS1		R																	
02	T																				
03	T																				
04	T																				

Figure 2-7 (Part 2 of 4). System/34 B Receive Program

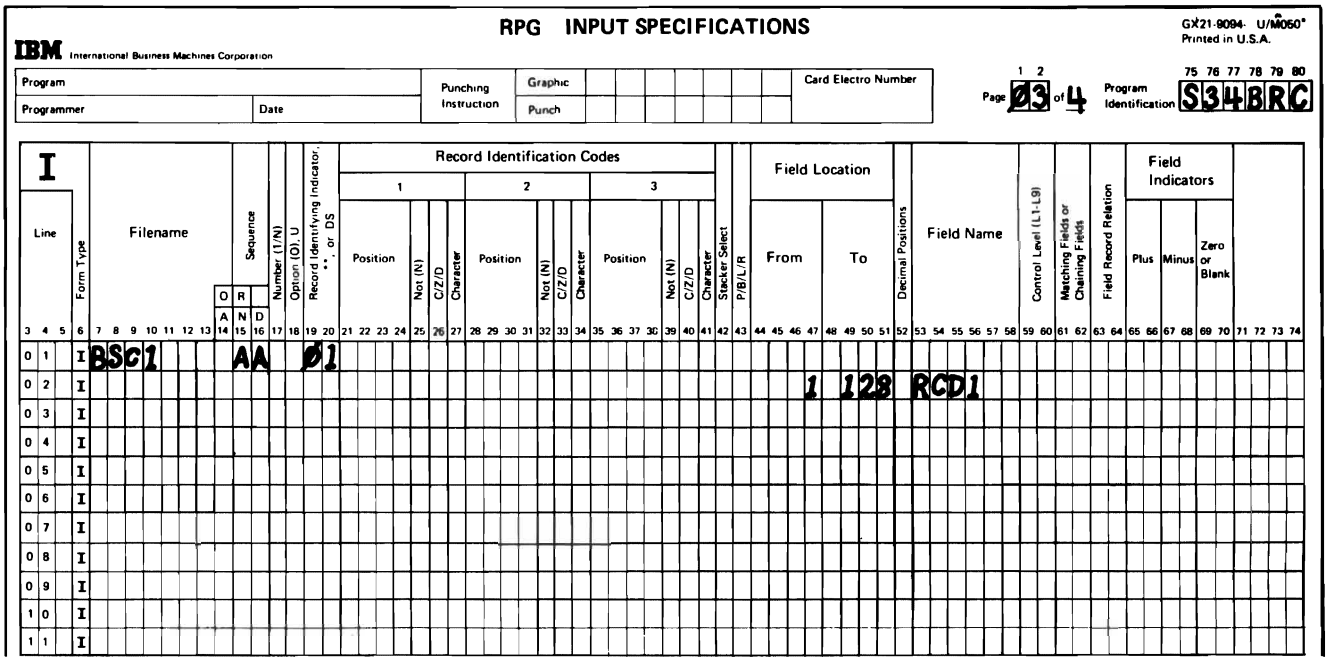


Figure 2-7 (Part 3 of 4). System/34 B Receive Program

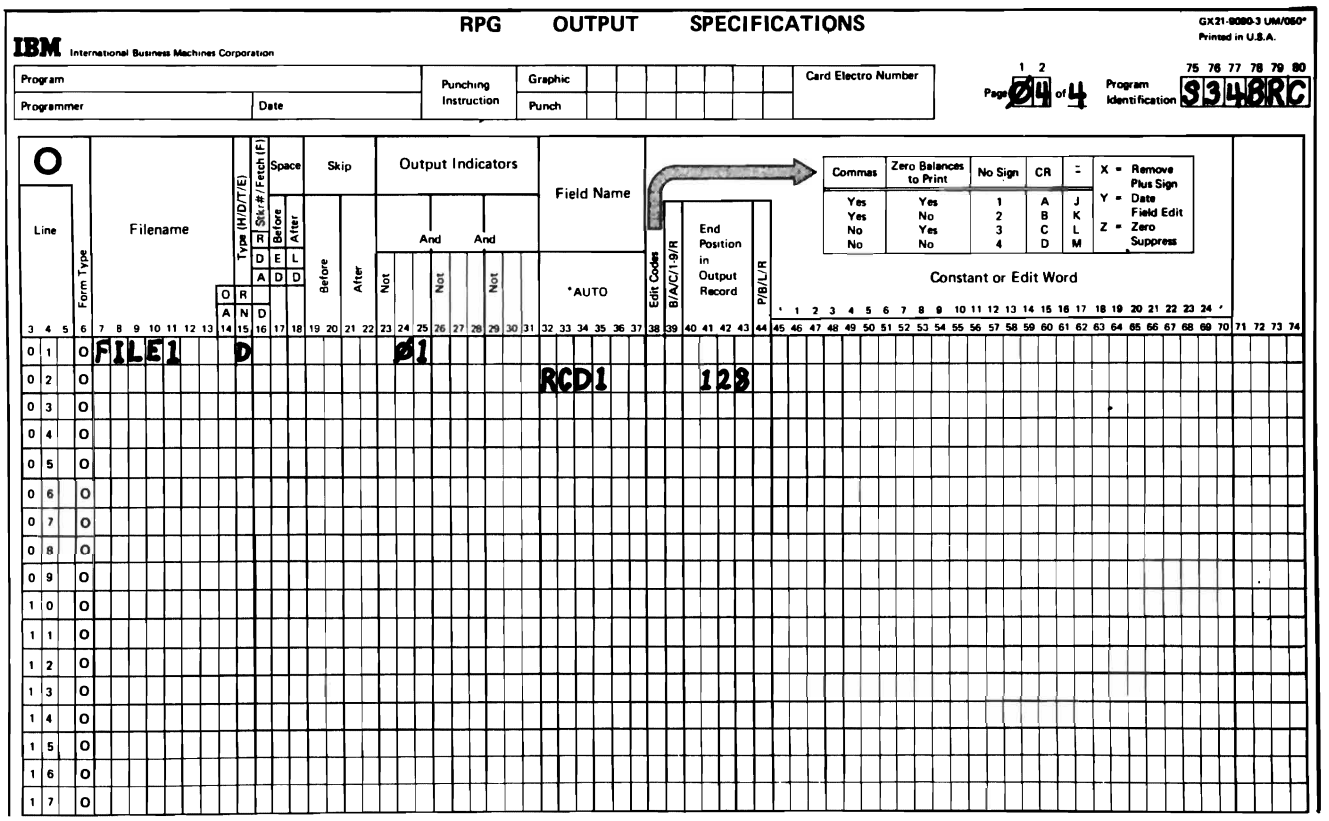


Figure 2-7 (Part 4 of 4). System/34 B Receive Program

Chapter 3. Data Communications Programming with BSC–Basic Assembler

The IBM System/34 BSC support provides BSC macroinstructions via the basic assembler language. BSC macroinstructions enable the basic assembler program to send and receive data over communications lines. The BSC support performs all functions necessary to establish the link connections, exchange identification sequences, send and receive data, and execute the correct termination or disconnect procedures.

Appendix D contains a description of the System/34 BSC line interface and is referred to throughout this chapter.

System/34 BSC support runs as a separate task from the basic assembler program, thus allowing the basic assembler program to be swapped into and out of main storage. The BSC task requires 6 K bytes of nonswappable main storage to run.

Note: The user area of main storage must be at least 20 K bytes or no other programs can run. If you wish to run BSC and spooling concurrently in a 32 K system, you must start the BSC program first and then start spooling.

OTHER SYSTEMS

Binary synchronous data transfers are possible between System/34 and the following:

- Another System/34 with either basic assembler or RPG II
- System/32 with either basic assembler or RPG II
- System/3 with RPG II, MLMP, or CCP
- System/7 with MSP/7
- System/360 Model 20 Input/Output Control System for the Binary Synchronous Communications Adapter
- Operating System Telecommunications Access Method (OS or OS/VS TCAM)
- Operating System or Disk Operating System Basic Telecommunications Access Method (OS, OS/VS, DOS/VS, or DOS BTAM)
- System/360 Model 20 Input/Output Control System for the Binary Synchronous Communications Adapter
- Operating System Telecommunications Access Method (OS or OS/VS TCAM)

- Operating System or Disk Operating System Virtual Telecommunications Access Method (DOS/VS or OS/VS VTAM)
- Customer Information Control System (CICS/DOS/VS or CICS/VS)
- Information Management System (IMS/VS)
- IBM 3741 Model 2 Data Station or Model 4 Programmable Work Station
- IBM 3747 Data Converter
- IBM 5231 Data Collection Controller Model 2 (as a 3741 in transmit mode only)
- IBM 3750 Switching System (World Trade only)
- IBM 5110 (in 3741 mode)
- IBM Series 1 (in System/3 mode)
- IBM 5260 Point of Sale Terminal (in 3740 mode)
- IBM 5280 Distributed Data System (in 3740 mode)

For System/34 data communications operation procedures, see *System/34 Operator's Guide*.

For a description of how to write basic assembler programs, see the *Basic Assembler and Macro Processor Reference Manual*.

BSC STATION TYPES

BSC macroinstructions permit System/34 to function as any of the following station types:

- Receive only (receive input data from a remote terminal)
- Transmit only (transmit data to a remote terminal)
- Transmit and receive (no conversational reply). Three modes of operation are possible:
  - Transmit a file, then receive another file
  - Receive a file, then transmit another file
  - Transmit records of one file interspersed with receiving records of another file

BASIC ASSEMBLER BSC PROGRAMS

Every BSC program you write must do these two functions:

- Prepare BSC DTFs for data reception, data transmission, or both.
- Initiate and terminate the transfer of data (receive data, transmit data, or both).

Preparing For Data Transfer

When preparing for data transfer, always include the following three steps:

1. Generate field displacements and labels for the BSC DTFs by using the $DTFO macroinstruction coded with BSC-Y and FIELD-Y.
2. Prepare BSC data files. Define each BSC file ($DTFB), allocate it ($ALOC), and open it ($OPEN).
3. If data in your BSC files requires translation, either before it is transmitted or after it is received, you must provide for data translation by constructing translate tables ($TRTB macroinstruction for EBCDIC/ASCII tables) and generating a translate parameter list ($TRL). When you translate data, generate the interface to the translate routine ($TRAN).

   Note: If you are transmitting or receiving ASCII data, be sure you have given the polling and addressing characters and station identification sequences in ASCII.

Initiating and Terminating the Transfer of Data

To initiate data transfer you must issue the following requests:

- Get requests to receive data ($GETB)
- Put requests to transmit data ($PUTB)

The first get or put request causes BSC to establish line connection with the remote station.

Termination of data transfer depends on whether the System/34 is receiving data ($GETB) or transmitting data ($PUTB).

If System/34 is transmitting, then terminate the data to the current file by one of the following means:

- $PUTB with OPC-EOF. This transmits the last block of data. System/34 then transmits ETX and EOT (in 3740 mode, System/34 waits for the next user operation).
- $PUTB to another transmit file. This transmits the last block of data from the current file. System/34 sends EOT and line initialization for the new file takes place. (In 3740 multiple file mode, STX ETX replaces the EOT and line initialization.)
- $GETB to a receive file. This transmits the same sequences as issuing a $PUTB to another transmit file.
- $CLOS to the current file. This transmits the last block of data and EOT (or DISC if switched lines). (In the case of 3740 multiple file mode, use $CLOS to transmit EOT.)

If System/34 is receiving, the remote station initiates data termination. You can detect this by coding EOF on the $GET macroinstruction or by checking for hex 42 ($BSEOF) in the $BSCMP field of the BSC DTF after each $GETB request. Issue successive $GETB requests until detecting EOF or an error. You can detect a BSC error by coding REJECT on the $GETB macroinstruction.

Move Mode

System/34 executes all BSC get and put requests in move mode. BSC moves data from the BSC I/O buffers to the logical buffer on get requests, and from the logical buffer to the BSC I/O buffers on put requests.

A single get or put request does not necessarily result in the actual data transmission over the communications line. For a get request, the remote station transmits data only if the get request moves to the logical buffer the last record contained in the BSC I/O buffer.

A put request transmits data to the remote station only if the record to be moved to a BSC I/O buffer cannot be contained in the current I/O buffer.

BSC MACROINSTRUCTIONS

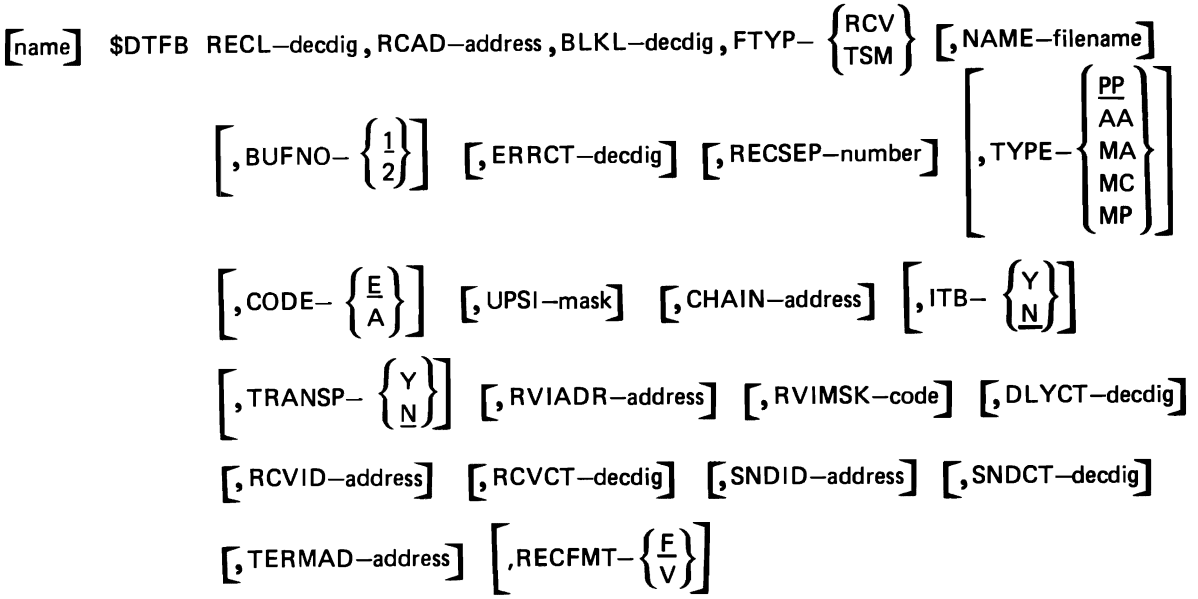
This section describes the following macroinstructions that support BSC:

- $DTFB builds a DTF for BSC get and put operations.
- $GETB builds the interface to get a BSC record.
- $PUTB builds the interface to put a BSC record.
- $TRL, $TRTB, and $TRAN build the interfaces required to translate data from ASCII to EBCDIC or from EBCDIC to ASCII.

Define the File for BSC ($DTFB)

The DTF provides information needed to allocate, open, close, and access a BSC file. This macroinstruction generates the code that builds the BSC DTF.

The format of the $DTFB macroinstruction is:



RECL: Specifies, in decimal, the maximum record length for this file, excluding transmission control character. The maximum allowable record length is 4075 bytes. However, if data is being blocked (with ITBs or record separators), the record length cannot be so large as to force the physical I/O buffer to be longer than 4096 bytes. Buffer size = (record length * number of records per block) + number of bytes needed for ITBs or record separators + 21. (Rounded up to a multiple of eight.) Number of bytes needed for ITBs = number of records per block minus 1 (nontransparent), or (number or records per block minus 1) times 3 (transparent).

Number of bytes needed for record separator = number of records per block.

Note: For get-a-block operations (OPC-BLK), the record length in the DTF ($BSRCL) is modified by BSC to reflect the length of the block (including transmission control characters) received.

RCAD: Specifies the symbolic address identifying the leftmost byte of your logical buffer. The logical buffer must be large enough to contain one record for this file.

Records are moved from the logical buffer to the BSC I/O buffers on put requests ($PUTB macroinstruction), and are moved from the BSC I/O buffers to the logical buffer on get requests ($GETB macroinstruction).

BLKL: Specifies, in decimal, the maximum block length for this file, excluding line control characters. Block length must be equal to or greater than the record length (RECL operand). For maximum block length, see RECL.

FTYP: Specifies whether put requests (TSM) or get requests (RCV) are to be performed on this file.

NAME: Specifies the name of the BSC file to be accessed. If this operand is omitted, no file name is used. The file name is used in certain SSP error messages.

BUFNO: Specifies the number of I/O buffers and IOBs to be contained in the I/O area for this file (either 1 or 2). If this operand is omitted, 1 is assumed.

ERRCT: Specifies the number of times an unsuccessful BSC operation is retried before an error condition is posted. Valid entries for this parameter are 1 through 255. If this operand is omitted, 7 is assumed.

RECSEP: Specifies a 1-byte, 2-character hexadecimal value. For put files, BSC inserts the specified byte between blocked records. For get files, this parameter indicates that the data being received has an intermediate record separator to be removed. Any valid ASCII or EBCDIC character can be used. The following is a list of *invalid* characters:

ASCII (hex)	**EBCDIC (hex)**
00	00
01	01
02	02
03	03
04	10
05	1F
11	26
15	2D
16	32
17	37
1F	3D

TYPE: Specifies the type of line connection to be established for this file.

- *PP* specifies that this file will use a point-to-point nonswitched line. PP is assumed if no line type is specified.
- *AA* specifies that this file will use a switched line with automatic answer.
- *MA* specifies that this file will use a switched line with manual answer.
- *MC* specifies that this file will use a switched line with manual call.
- *MP* specifies that this file will use a multipoint line, and that this station is a tributary station. TYPE-MP requires the TERMAD operand.

Note: If you are using an autocall line, the switch type specified has no effect. However, if no phone list is specified in the COMM OCL statement, the switch type specified here is established.

CODE: Specifies whether the character code used on your communications link is EBCDIC (E) or ASCII (A). If this operand is omitted, E is assumed.

UPSI: Specifies the settings of the external (SWITCH statement) indicators used for conditionally opening files. The code must be specified as 8 binary bits. For example, to test bits 0, 3, 5, and 7, you would enter UPSI-10010101. If this operand is omitted, zeros are assumed.

CHAIN: Specifies the symbolic address of the next DTF in the chain. Chained DTFs are allocated, opened, or closed with the first DTF in the chain. To decrease the execution time of your program, all BSC DTFs should be chained together.

ITB: Specifies whether intermediate block checking is requested: Y if yes, N if no. ITB is not valid with transparent transmit files. If this operand is omitted, N is assumed.

TRANSP: Specifies whether data for this file will be transmitted or received in transparent mode: Y if yes, N if no. If this operand is omitted, N is assumed.

RVIADR: Specifies the symbolic address of a 1-byte field you provide. The field is used with the mask specified in the RVIMSK operand (following paragraph) to indicate when a reverse interrupt request (RVI) is received. RVIADR-address requires the RVIMSK operand.

RVIMSK: Specifies two hexadecimal digits to represent the reverse interrupt (RVI) mask. The bits represented by the mask are set on by BSC in the RVIADR field (preceding paragraph) if reverse interrupt request is received.

DLYCT: Specifies a decimal delay count of the number of seconds you want BSC to wait before sending an EOT and displaying an error message. This wait begins after a block of data for a file has been sent or received, and ends when another block of data for that file is sent or received, or when the delay count is completed. The number must be within the range of 1 through 999. If you do not specify a number, a 180-second delay count is allowed for such things as device errors, halts, and readying I/O devices.

RCVID: Specifies the symbolic address of the leftmost byte of the identification sequence required from the remote station. RCVID requires the RCVCT operand. Using RCVID and RCVCT may improve security on switched lines; these operands are valid for switched lines only. If the IDs do not match, initialization terminates.

RCVCT: Specifies, in decimal, the length of the identification sequence required from the remote station. The length can be from 1 to 15. If 1 is specified, BSC expects to receive two characters–two duplicates of the character addressed by the RCVID operand (preceding paragraph). If no length is specified, 0 is assumed. RCVCT requires the RCVID operand.

SNDID: Specifies the symbolic address of the leftmost byte of the identification sequence required by the remote station. SNDID requires the SNDCT operand. Using the SNDID and SNDCT operands may improve security on switched lines; these operands are valid for switched lines only.

SNDCT: Specifies, in decimal, the length of the identification sequence required by the remote station. Length can be from 1 to 15. If 1 is specified, BSC transmits two characters–duplicates of the character addressed by the SNDID operand (preceding paragraph). SNDCT requires the SNDID operand.

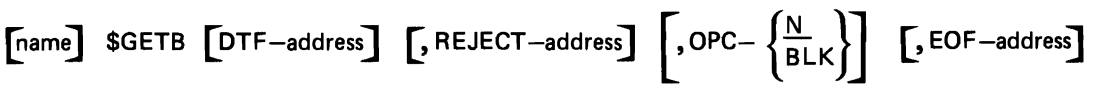
TERMAD: Specifies the hexadecimal representation of the 2-character polling or addressing sequence used by this file. If this is a transmit file (FTYP-TSM), TERMAD specifies polling characters; if this is a receive file (FTYP-RCV), TERMAD specifies addressing characters. Each tributary station on a multipoint line must have unique polling and addressing characters. The TERMAD operand is used only when TYPE-MP is specified.

RECFMT: Specifies whether records of fixed (F) or variable (V or VARY) length are to be received. If you specify V (or VARY), TRANSP-N and ITB-N must also be specified, and you may not use 3780 blank compression or truncation. If the record separator is not supplied, it will default to X'1E'. The received record length will be returned in the $BSRCL field of the get file of the BSC DTF. The record length specified when the DTF is opened or allocated must be the maximum record length expected.

Issue a Get Request ($GETB)

The $GETB macroinstruction generates code to move data from a BSC I/O buffer to your logical buffer. To use this macroinstruction, construct a BSC DTF for the file and use the $DTFO macroinstruction to generate the labels and establish the offsets for the DTF.

The format of the $GETB macroinstruction is:



DTF: Specifies the address of the DTF (file) for which the get was issued. If this operand is omitted, the address of the DTF is assumed to be in register 2.

REJECT: Specifies the routine to receive control if this get request is rejected by BSC. If this operand is omitted, control is returned to the user program at the next sequential instruction after the $GETB.

OPC: Specifies how BSC handles the record received for this program. N indicates normal deblocking by BSC before the record is passed to the receiving program. That is, BSC removes transmission control characters and moves the data to the logical buffer (RCAD in $DTFB) one record at a time. BLK indicates the entire block (including control characters) is passed to the receiving program. BSC places the length of the block in $BSRCL in the DTF. If this operand is omitted, N is assumed.

Note: If you specify OPC-BLK, be sure your logical buffer (RCAD in $DTFB) is large enough to hold an entire block of data plus transmission control characters.

EOF: Specifies your end-of-file routine. If this operand is omitted, control is returned to the user program at the next sequential instruction after the $GETB.

If EOF or REJECT addresses are not specified, your program should check the return code in the DTF to determine the outcome of the operation.

Issue a Put Request ($PUTB)

The $PUTB macroinstruction generates code to move data from your logical buffer to a BSC I/O buffer. To use this macroinstruction, construct a BSC DTF for the file and use the $DTFO macroinstruction to generate the labels and establish the offsets for the DTF.

The format of the $PUTB macroinstruction is:

DTF: Specifies the address of the DTF (file) for which the put was issued. If this operand is omitted, the address is assumed to be in register 2.

REJECT: Specifies the routine to receive control if the put request is rejected by BSC. If this operand is omitted, control is returned to the user program at the next sequential instruction after the $PUTB. You should check the return code to determine the outcome of the operation.

Note: To prevent issuing BSC requests after a BSC error has occurred, this parameter should always be coded.

OPC: Specifies how BSC should send this record.

- *N:* Specifies normal record blocking before the record is sent. If this operand is omitted, *N* is assumed.
- *EOB:* Specifies the block is terminated with this record.
- *EOF:* Specifies end of file. The put file is closed by transmitting the last block of data with end of text (ETX), then transmitting end of transmission (EOT). If operation is in 3740 multiple file mode, the last block of data is transmitted with end-of-text block (ETB), and System/34 waits for the next user operation.

Note: No new data is sent when EOF is issued. The ETX or ETB is placed at the end of the previous block of data.

Generate an Interface to the Translate Routine ($TRAN)

The $TRAN macroinstruction generates an interface to the translate routine.

The format of the $TRAN macroinstruction is:

[name] $TRAN [TRL-address]

TRL: Specifies the symbolic address of the translate parameter list. If this operand is omitted, the address is assumed to be in register 1. If the $TRL macroinstruction is used to generate the parameter list, this address should be the label assigned to the $TRL macroinstruction. The parameter list is described as follows:

Field Length	Field Description
2	Address of the translate table (Your program must define the translate table.)
2	FROM field address, for translation
2	TO field address, for translation
2	Number of bytes to translate
1	Completion code: Hex 00: Translation completion, no errors Hex FF: Invalid character encountered

Generate a Translate Parameter List ($TRL)

The $TRL macroinstruction generates a parameter list used by the translate routine. $TRL does not generate executable code.

The format of the $TRL macroinstruction is:

[name] $TRL TO-address,FROM-address,LEN-decdig,TRT-address

TO: Specifies the symbolic address of the leftmost byte of the field to which the translated data will be moved.

FROM: Specifies the symbolic address of the leftmost byte of the data field to be translated. This address may be the same as the address specified in the TO operand.

LEN: Specifies, in decimal, the number of characters to be translated.

TRT: Specifies the symbolic address of the leftmost byte of the translate table. If the $TRTB macroinstruction is used to generate the translate table, this address should be the label assigned to the $TRTB.

Generate a Translate Table ($TRTB)

This macroinstruction generates an EBCDIC to ASCII or an ASCII to EBCDIC translation table. The table is generated in the format required by the $TRL macroinstruction, and can be addressed by $TRL when you translate data.

The format of the $TRTB macroinstruction is:

CODE: Specifies whether the data is to be translated from EBCDIC to ASCII (E) or from ASCII to EBCDIC (A). If this operand is omitted, E is assumed. If CODE-E is specified, $TRTB generates a 258-byte translate table; if CODE-A is specified, $TRTB generates a 130-byte translate table.

HEX: Specifies the hexadecimal digits with which to replace any invalid characters found during translation. If the HEX operand is not specified, the replacement character is hex 3F for ASCII to EBCDIC or hex 1A for EBCDIC to ASCII.

Translate tables generated by the $TRTB macroinstruction are generated in the following format:

Byte	Field Description
0	Byte used to identify an invalid character (a character that is not to be translated).
1	Byte substituted for characters that are not to be translated.
2-257	256-byte translate table for EBCDIC to ASCII.
2-129	128-byte translate table for ASCII to EBCDIC.

Construct the translate table so that the displacement from the beginning of the table equals the hexadecimal representation of the untranslated character. The contents of that location is the character to be translated to. (For example, if you want to translate hex C1 to hex 41, you should construct a translate table in which the value at displacement hex C1 in the table is hex 41.)

The translate routine processes a field, specified by the $TRL macroinstruction, one byte at a time.

The byte at a given displacement is compared with the first byte in the translate area (byte 0). If they are equal, the character is considered to be invalid, and the following actions are performed:

- The completion code in the parameter list is set to indicate that an invalid character was detected.
- The second byte of the translate area (byte 1) is substituted for the original character.
- Translation continues with the next character. After the translate routine is finished, control is returned to your program with a completion code in the translate routine parameter list.

PROGRAMMING CONSIDERATIONS

Autocall Support

When the System/34 is configured with the MLCA (multiline communications adapter) and the Autocall feature, remote locations can be called automatically without operator intervention. Because there is no reference to the autocall capabilities in user programs, existing programs can use autocall without modification. You specify autocall by using the PHONE parameter on the COMM OCL statement. The COMM statement is described in the *System Support Reference Manual.*

The phone list specified in the COMM statement can contain up to 120 phone numbers and is generated by the DEFINEPN procedure, which is described in the *System Support Reference Manual.*

If a batch BSC job is run on an autocall line and no phone list is specified in the COMM statement (or there is no COMM statement), the call mode defaults to the mode specified in the user's DTF or the display station communications configuration record. The mode can be manual answer, manual call, or auto answer. If the phone list is specified in the COMM statement but the line is not an autocall line, or the autocall task was not loaded at IPL time, then the line is considered to be a manual call, manual answer, or auto answer line, depending on the switch type defined for the line.

X.21 Support

When the System/34 is configured with the MLCA and has the X.21 switched feature, remote locations are called automatically on switched lines. Existing programs can use the X.21 capabilities without modification. You specify the list of numbers for the public data network with the PHONE parameter of the COMM OCL statement. The COMM statement is described in the *System Support Reference Manual.*

The list specified in the COMM statement can contain up to 84 connection numbers and is generated by the DEFINX21 procedure, which is described in the *System Support Reference Manual.*

When the first request during any BSC job step is made to BSC data management, the phone list is searched for a number to call. The first time the list is referenced, the search begins with the first number. For each succeeding reference, the search begins with the next available number. If a number cannot be reached, its retry count is decremented and the next number is called. If no numbers in the list can be reached, either the permanent error indicator is turned on or the permanent error return code is passed to the user program. A message is displayed indicating each number that could not be reached. When a number is reached, a message is displayed indicating the number reached and communications proceed. When the job step ends, you can use the IF statement to reexecute the step and call the next number. You can also use the same phone list in a succeeding step of the job.

The ability to call multiple locations within a single BSC job step is primarily useful when the System/34 is receiving data from those locations. Because any number may be called during a request, transmission of data to a particular location should be performed with a single number phone list.

If, during the receiving of data, a permanent error occurs, the phone number associated with the communications link is not reset. Because the number is not reset, it cannot be called again on subsequent passes through the list. The recovery associated with that particular job step is the responsibility of the user.

Blank Compression/Expansion

In order to use the line more effectively and decrease communications line costs, the System/34 BSC offers the RPG II and assembler users the capability of transmitting and/or receiving data with all contiguous blanks (2 or more) removed. This is done by using the same format used by the IBM 3780.

For put files, BSC moves data from the logical buffer to the BSC I/O buffer with blanks removed and compression control characters inserted. After each record, BSC inserts an IRS (intermediate record separator).

If printing of the record is to take place from the logical buffer, it should be done before a put because BSC alters the record with IGS characters and count characters while compressing the record.

For get files, the procedure is reversed; the System/34 BSC recognizes the intermediate record separator control characters, inserts the blanks removed by the remote station, and moves the record from the BSC I/O buffer to the logical buffer.

To use blank compression/expansion, execute an OVERRIDE procedure or a SETR utility control statement with BLANK-C specified before executing the BSC program. See Appendix D for an example of blank compression/expansion.

Note: You cannot use blank compression/expansion with transparent or ITB modes.

Blank Truncation

System/34 BSC can also transmit and/or receive data with only the trailing blanks removed.

For put files, BSC moves data from the logical buffer to the BSC I/O buffer with all trailing blanks removed. After each record, BSC inserts an IRS character.

For get files, BSC scans the data in the BSC I/O buffer for an IRS. BSC then moves all data up to the IRS to the logical buffer and blanks the remainder of the logical buffer.

To use blank truncation, execute an OVERRIDE procedure or a SETR utility control statement with BLANK-T specified before executing the BSC program. See Appendix D for an example of blank truncation.

Notes:

1. You cannot use blank truncation with ITB mode. You can specify blank truncation with transparent mode; however, this negates the truncation feature.
2. When you use blank compression/expansion or blank truncation with blocked records, the number of records per block will vary depending on the number of blanks in each record.

Data Formats

System/34 BSC support uses the following data formats for transmission of data; use these formats when sending data to System/34 from a processing unit.

- Nontransparent, non-ITB: STX-data-ETX(ETB)
- Nontransparent, non-ITB, blocked:

  STX-rec 1/rec 2/.../rec n-1/rec n-ETX(ETB)
- Nontransparent, ITB: STX-data-ITB-data-ITB-data-ETX(ETB)
- Transparent, non-ITB: DLE-STX-data-DLE-ETX(ETB)
- Transparent, non-ITB, blocked:
  DLE-STX-rec 1/rec 2/.../rec n-1/rec n-DLE-ETX(ETB)
- Transparent, ITB (receive files only): DLE-STX-data-DLE-ITB-DLE-STX-data-DLE-ITB-DLE-STX-data-DLE-ETX(ETB)

Data can be either fixed length and unblocked, or fixed length and blocked.

Errors

If an error occurs at either station, System/34 retries the operation up to the number of times specified by the $DTFB macroinstructions or up to the number of retries specified by the ALTERBSC procedure command or the SETB utility control statement. (See the *System Support Reference Manual* for information on the SETB utility control statement and the ALTERBSC procedure command.)

BSC Environment

BSC configuration information is altered by System/34 ALTERBSC and OVERRIDE procedures. When you are running BSC programs from the input job queue, the configuration information from the system console will be used for the job. The SSP retrieves this information at the same time the job is run.

The ALTERBSC and OVERRIDE procedures run the $SETCF utility. Instead of using these procedures to alter the BSC environment, you can use the SETB and SETR utility control statements of the $SETCF utility.

For information on coding System/34 procedure commands and utility control statements, see the *System Support Reference Manual*.

SAMPLE PROGRAMS

On the following pages, the sample programs illustrate the use of the BSC macroinstructions in basic assembler programs. One sample program transmits data, one receives data, and one transmits and receives data.

Transmit

This program reads a file from disk (BSCFIL) and transmits it to another System/34.

IBM System/34 Basic Assembler Coding Form

GX21-9279 Printed in U.S.A.

PROGRAM BSASM1 | PROGRAMMER | DATE | KEYING INSTRUCTIONS | GRAPHIC CHARACTER | PAGE 1 OF 3 | CARD ELECTRO NUMBER

```
XMIT1    START 0
*
*     ALLOCATE AND OPEN CHAIN OF DTF'S
*
         $ALOC DTF-BSCDTF                ALLOCATE ALL DTF'S
         $OPEN DTF-BSCDTF                OPEN ALL DTF'S
*
*          TRANSMIT THE FILE
*
LOOP1    EQU   *
         $GETD DTF-DSKDTF,ERR-DSKERR,EOF-CLOSE   GET A RECORD
         L     DSKDTF+$F1,KB,3           LOAD ADDRESS OF RECORDS
         MVC   BUFFR1+79(80),79(2)       MOVE RECORD
         $PUTB DTF-BSCDTF,REJECT-BSCERR
         B     LOOP1                     LOOP UNTIL END OF FILE
*
*     PRINT RESULTS OF TRANSMISSION
*
```

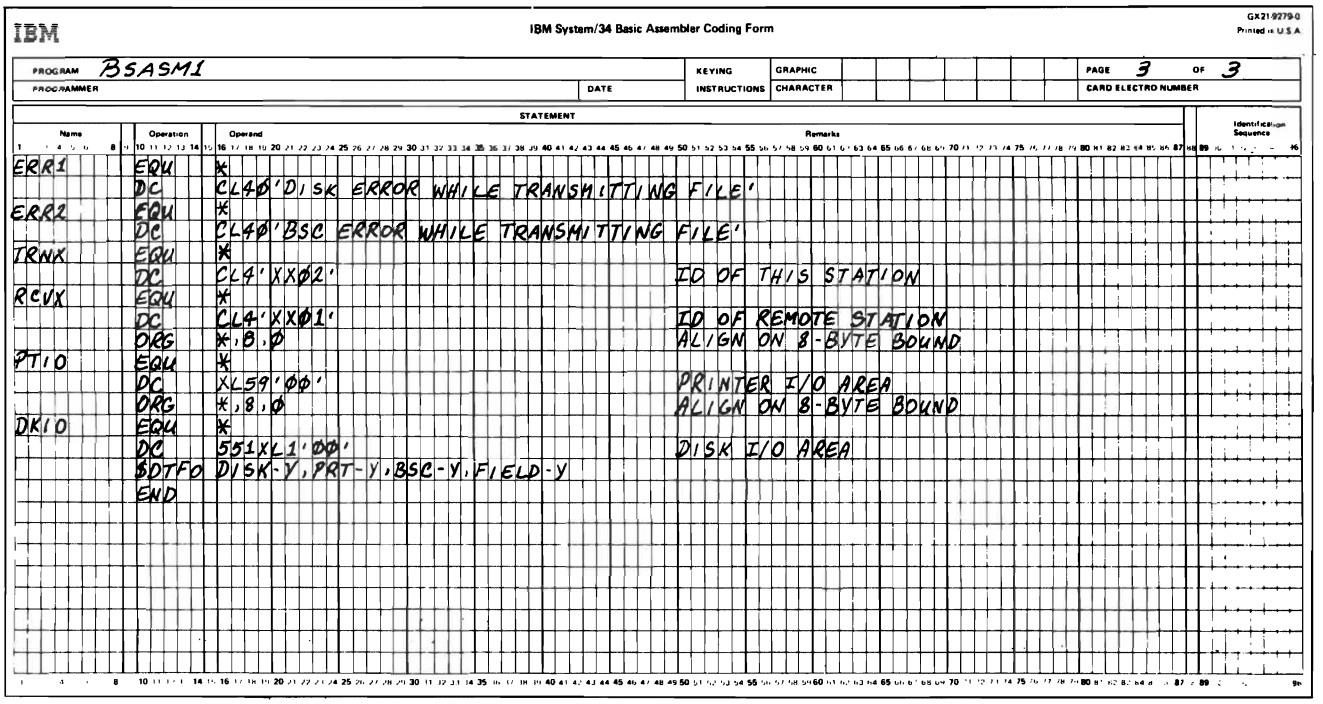
IBM System/34 Basic Assembler Coding Form

GX21-9279 Printed in U.S.A.

PROGRAM BSASM1 | PROGRAMMER | DATE | KEYING INSTRUCTIONS | GRAPHIC CHARACTER | PAGE 2 OF 3 | CARD ELECTRO NUMBER

```
DSKERR   EQU   *
         MVC   PTBUF+39(40),ERR1+39      DISK ERROR MESSAGE
         J     CLOSE
BSCERR   EQU   *
         MVC   PTBUF+39(40),ERR2+39      BSC ERROR MESSAGE
CLOSE    EQU   *
         $PUTP DTF-PRTDTF
         $CLOS DTF-BSCDTF
         $EOJ
*
*     DTF'S, BUFFERS, AND EQUATES
*
BSCDTF   $DTFIB RECL-80,BLKL-80,RCAD-BUFFR1,FTYP-TSM,TYPE-MC,                  C
               CHAIN-PRTDTF,RCVID-RCVX,RCVCT-4,SNDID-TRNX,SNDCT-4
BUFFR1   EQU   *
         DC    1XL80'00'
PRTDTF   $DTFIP RCAD-PTBUF,IOAREA-PTIO,RECL-40,                                C
               PRINT-Y,CHAIN-DSKDTF,SPACEA-1,NAME-PRINT
DSKDTF   $DTFID ACCESS-SG,RECL-80,NAME-BSCFIL,BLKL-512,IOAREA-DKIO
PTBUF    EQU   *
         DC    CL40'FILE SUCCESSFULLY TRANSMITTED' PRINT RECORD AREA
```

Figure 3-1 (Part 1 of 2). Transmit Program



IBM System/34 Basic Assembler Coding Form

GX21-9279-0
Printed in U.S.A.

PROGRAM BSASM1 | KEYING | GRAPHIC | PAGE 3 OF 3
PROGRAMMER | DATE | INSTRUCTIONS | CHARACTER | CARD ELECTRO NUMBER

STATEMENT

Name	Operation	Operand	Remarks
ERR1	EQU	*	
	DC	CL40'DISK ERROR WHILE TRANSMITTING FILE'	
ERR2	EQU	*	
	DC	CL40'BSC ERROR WHILE TRANSMITTING FILE'	
TRNX	EQU	*	
	DC	CL4'X'02'	ID OF THIS STATION
RCVX	EQU	*	
	DC	CL4'X'01'	ID OF REMOTE STATION
	ORG	*,8,0	ALIGN ON 8-BYTE BOUND
PTIO	EQU	*	
	DC	XL59'00'	PRINTER I/O AREA
	ORG	*,8,0	ALIGN ON 8-BYTE BOUND
DKIO	EQU	*	
	DC	551XL1'00'	DISK I/O AREA
	$DTFD	DISK-Y,PRT-Y,BSC-Y,FIELD-Y	
	END		

Figure 3-1 (Part 2 of 2). Transmit Program

Receive

The following program receives data from a System/32 and prints that data.

IBM System/34 Basic Assembler Coding Form — GX21 9279 Printed in U.S.A.

PROGRAM BSASM 2 — PAGE 1 OF 3

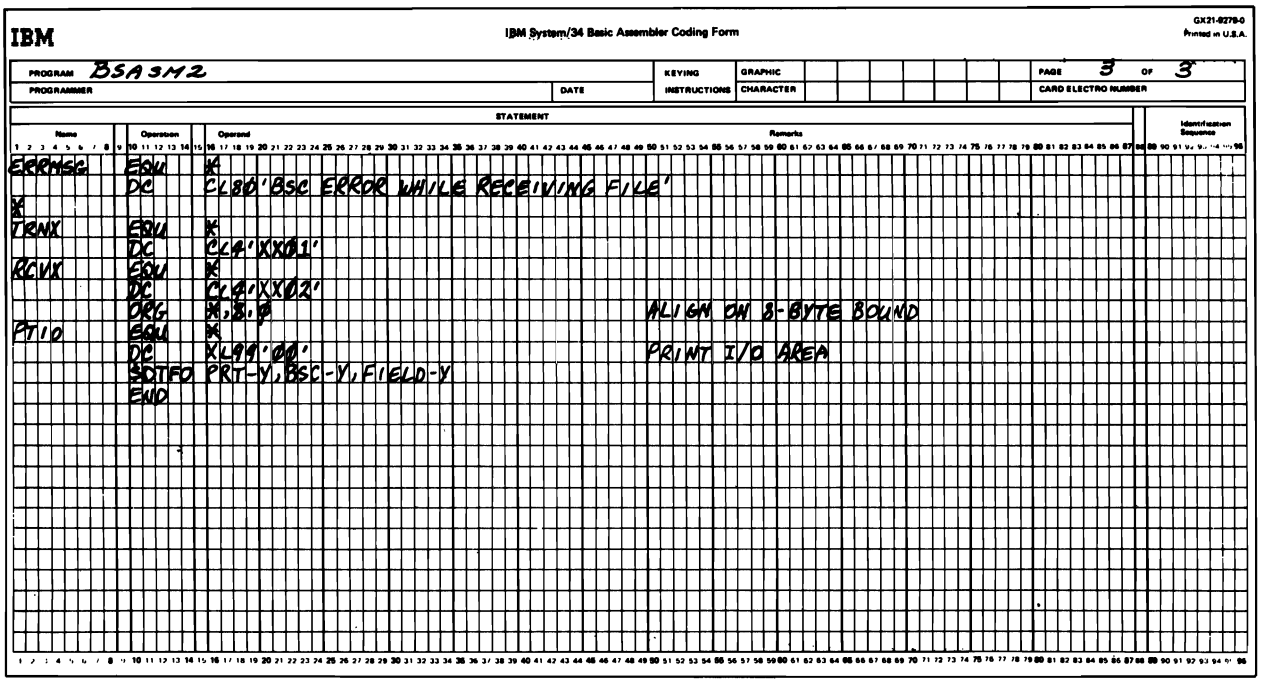
```
RECV     START 0
*
*        ALLOCATE AND OPEN DTF'S
*
         $ALOC DTF-BSCDTF                ALLOCATE BOTH DTF'S
         $OPEN DTF-BSCDTF                OPEN BOTH DTF'S
*
*        RECEIVE AND PRINT FILE
*
LOOP1    EQU   *
         $GETB DTF-BSCDTF,EOF-CLOSE,REJECT-BSCERR GET A RECORD
         $PUTP DTF-PRTDTF,ERR-CLOSE      PRINT THE RECORD
         B     LOOP1                     LOOP UNTIL END OF FILE
*
*        IF ERROR PRINT MESSAGE
*
```

IBM System/34 Basic Assembler Coding Form — GX21 9279 Printed in U.S.A.

PROGRAM BSASM2 — PAGE 2 OF 3

```
BSCERR   EQU   *
         MVC   PTBUF+79(80),ERRMSG+79    BSC ERROR MESSAGE
         $PUTP DTF-PRTDTF,SKIPB-1        PRINT MESSAGE
*
*        CLOSE DTF'S AND END JOB
*
CLOSE    EQU   *
         $CLOS DTF-BSCDTF                CLOSE BOTH DTF'S
         $ECJ                            END OF JOB
*
*        DTF'S, BUFFERS, AND EQUATES
*
BSCDTF   $DTFB RECL-80,BLKL-80,RCAD-PTBUF,FTYP-RCV,TYPE-MA,RCVID-RCVX, C
               RCVCT-4,SNDID-TRNX,SNDCT-4,CHAIN-PRTDTF
PRTDTF   $DTFP RCAD-PTBUF,RECL-80,PRINT-Y,IOAREA-PTIO,SPACEA-1,NAME-PR
*
PTBUF    EQU   *
         DC    80CL1' '                  BSC/PRINT RECORD AREA
```

Figure 3-2 (Part 1 of 2). Receive Program



IBM System/34 Basic Assembler Coding Form

GX21-9278-0
Printed in U.S.A.

PROGRAM BSA3M2 | KEYING INSTRUCTIONS | GRAPHIC CHARACTER | PAGE 3 OF 3

PROGRAMMER | DATE | CARD ELECTRO NUMBER

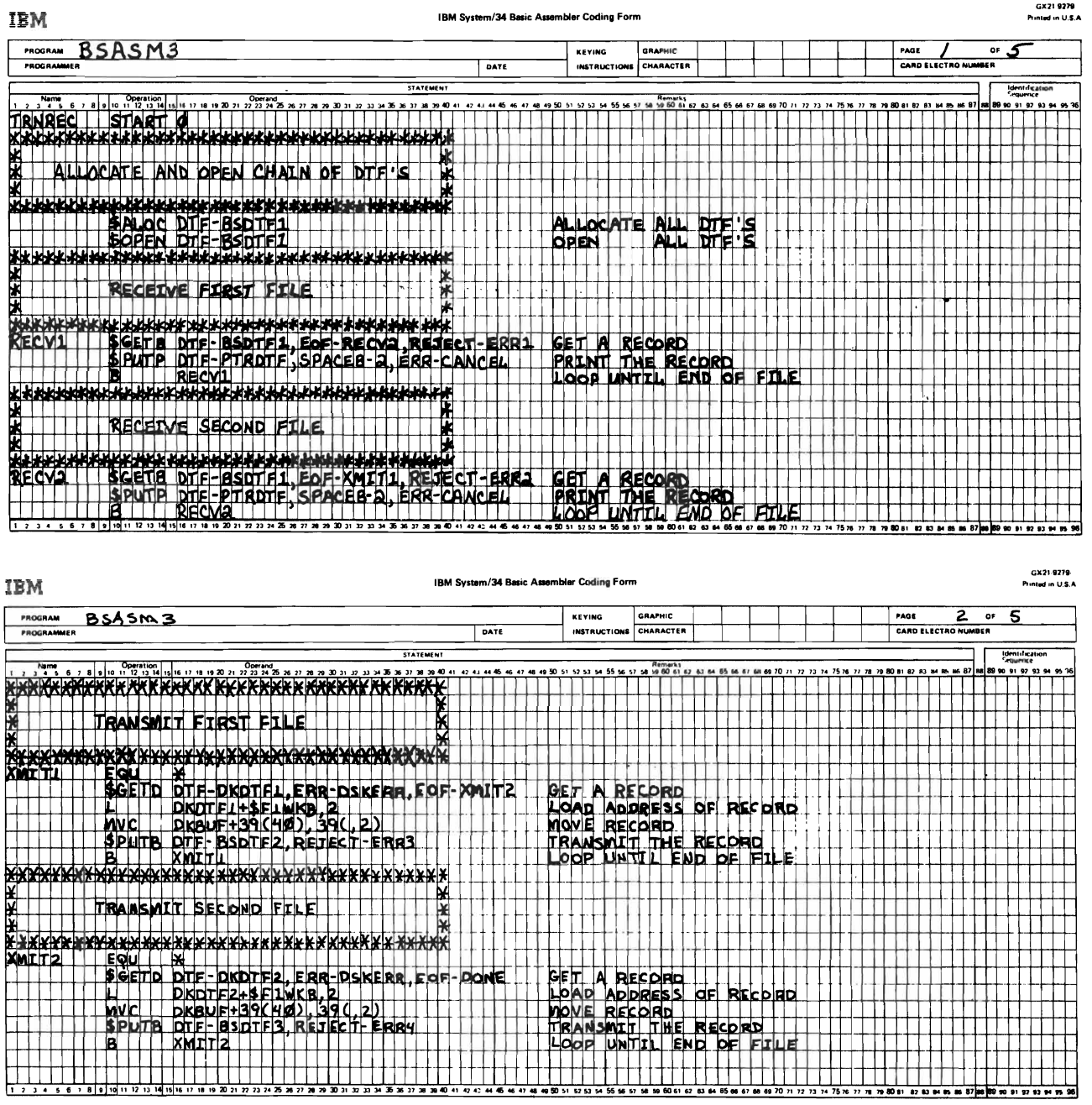
STATEMENT

Name	Operation	Operand	Remarks
ERRMSG	EQU	*	
	DC	CL80'BSC ERROR WHILE RECEIVING FILE'	
*			
TRNX	EQU	*	
	DC	CL4'XX01'	
RCVX	EQU	*	
	DC	CL4'XX02'	
	ORG	*,8,0	ALIGN ON 8-BYTE BOUND
PTIO	EQU	*	
	DC	XL99'00'	PRINT I/O AREA
	SOTFO	PRT-Y,BSC-Y,FIELD-Y	
	END		

Figure 3-2 (Part 2 of 2). Receive Program

Transmit and Receive

This program receives two files from a 3741 and then transmits two files to a 3741. The data is transmitted from two disk files (BSFIL1 and BSFIL2). The data received is printed. Before running this program, run an OVERRIDE with MLTFL-Y.



```
IBM System/34 Basic Assembler Coding Form                    GX21 9279 Printed in U.S.A.

PROGRAM BSASM3                                  PAGE 1 OF 5
PROGRAMMER            DATE                      CARD ELECTRO NUMBER

Name     Operation  Operand                    Remarks
TRNREC   START  0
*
*    ALLOCATE AND OPEN CHAIN OF DTF'S
*
         $ALOC  DTF-BSDTF1                    ALLOCATE ALL DTF'S
         $OPEN  DTF-BSDTF1                    OPEN     ALL DTF'S
*
*        RECEIVE FIRST FILE
*
RECV1    $GETB  DTF-BSDTF1,EOF-RECV2,REJECT-ERR1  GET A RECORD
         $PUTP  DTF-PTRDTF,SPACE0-2,ERR-CANCEL    PRINT THE RECORD
         B      RECV1                             LOOP UNTIL END OF FILE
*
*        RECEIVE SECOND FILE
*
RECV2    $GETB  DTF-BSDTF1,EOF-XMIT1,REJECT-ERR2  GET A RECORD
         $PUTP  DTF-PTRDTF,SPACE0-2,ERR-CANCEL    PRINT THE RECORD
         B      RECV2                             LOOP UNTIL END OF FILE
```



```
IBM System/34 Basic Assembler Coding Form                    GX21 9279 Printed in U.S.A.

PROGRAM BSASM3                                  PAGE 2 OF 5
PROGRAMMER            DATE                      CARD ELECTRO NUMBER

Name     Operation  Operand                    Remarks
*
*        TRANSMIT FIRST FILE
*
XMIT1    EQU    *
         $GETD  DTF-DKDTF1,ERR-DSKERR,EOF-XMIT2   GET A RECORD
         L      DKDTF1+$F1WKB,2                   LOAD ADDRESS OF RECORD
         MVC    DKBUF+39(40),39(,2)               MOVE RECORD
         $PUTB  DTF-BSDTF2,REJECT-ERR3            TRANSMIT THE RECORD
         B      XMIT1                             LOOP UNTIL END OF FILE
*
*        TRANSMIT SECOND FILE
*
XMIT2    EQU    *
         $GETD  DTF-DKDTF2,ERR-DSKERR,EOF-DONE    GET A RECORD
         L      DKDTF2+$F1WKB,2                   LOAD ADDRESS OF RECORD
         MVC    DKBUF+39(40),39(,2)               MOVE RECORD
         $PUTB  DTF-BSDTF3,REJECT-ERR4            TRANSMIT THE RECORD
         B      XMIT2                             LOOP UNTIL END OF FILE
```

Figure 3-3 (Part 1 of 3). Transmit and Receive Program

IBM System/34 Basic Assembler Coding Form

PROGRAM: BSASM 3 — PAGE 3 OF 5

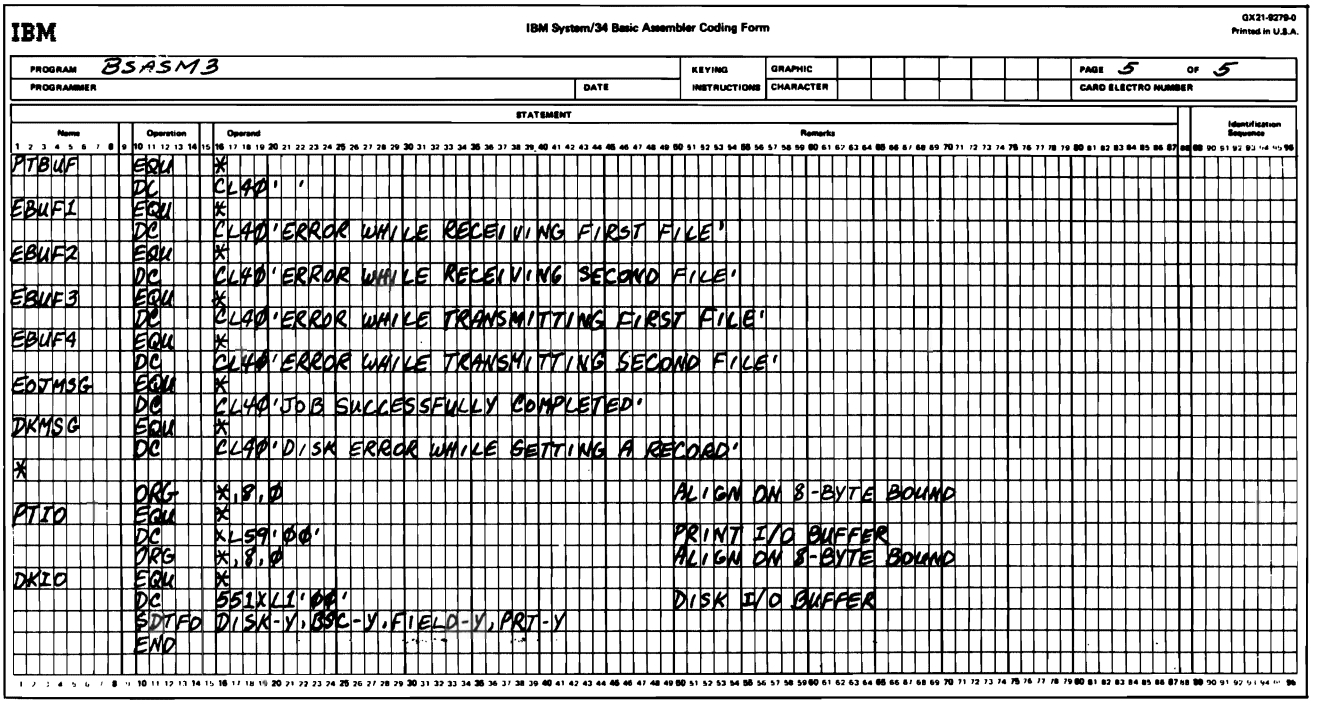
```
****************************************
*                                      *
*   PERFORM EOJ AND ERROR PROCESSING   *
*                                      *
****************************************
DONE     EQU   *
         MVC   PTBUF+39(4Ø),EØJMSG+39          MOVE GOOD MESSAGE
         J     PRINT                           JUMP TO PRINT
ERR1     EQU   *
         MVC   PTBUF+39(4Ø),EBUF1+39           BSC ERROR RECV FILE 1
         J     PRINT                           JUMP TO PRINT
ERR2     EQU   *
         MVC   PTBUF+39(4Ø),EBUF2+39           BSC ERROR RECV FILE 2
         J     PRINT
ERR3     EQU   *
         MVC   PTBUF+39(4Ø),EBUF3+39           BSC ERROR XMIT FILE 1
         J     PRINT                           JUMP TO PRINT
ERR4     EQU   *
         MVC   PTBUF+39(4Ø),EBUF4+39           BSC ERROR XMIT FILE 2
         J     PRINT                           JUMP TO PRINT
DSKERR   EQU   *
         MVC   PTBUF+39(4Ø),DKMSG+39           DISK ERROR MESSAGE
PRINT    EQU   *
         $PUTP DTF-PTRDTF,SPACEA-2             PRINT MESSAGE
```

IBM System/34 Basic Assembler Coding Form

PROGRAM: BSASM 3 — PAGE 4 OF 5

```
****************************************
*                                      *
*        CLOSE DTF'S AND END JOB       *
*                                      *
****************************************
CANCEL   $CLOS DTF-BSDTF1                      CLOSE ALL DTF'S
         $EOJ                                  END OF JOB
****************************************
*                                      *
*      DTF'S, BUFFERS, AND EQUATES     *
*                                      *
****************************************
BSDTF1   $DTFB RECL-4Ø,BLKL-4Ø,FTYP-RCV,TYPE-MA,RCAD-PTBUF,CHAIN-BSDTF2
BSDTF2   $DTFB RECL-4Ø,BLKL-4Ø,FTYP-TSM,TYPE-MA,RCAD-DKBUF,CHAIN-BSDTF3
BSDTF3   $DTFB RECL-4Ø,BLKL-4Ø,FTYP-TSM,TYPE-MA,RCAD-DKBUF,CHAIN-DKDTF1
DKDTF1   $DTFD ACCESS-CG,RECL-4Ø,NAME-BSFIL1,IOAREA-DKIO,               C
               BLKL-512,CHAIN-DKDTF2
DKDTF2   $DTFD ACCESS-CG,RECL-4Ø,NAME-BSFIL2,IOAREA-DKIO,               C
               BLKL-512,CHAIN-PTRDTF
PTRDTF   $DTFP RCAD-PTBUF,IOAREA-PTIO,PRINT-Y,RECL-4Ø,NAME-PRINT
*
DKBUF    EQU   *
         DC    CL4Ø' '
```

Figure 3-3 (Part 2 of 3). Transmit and Receive Program



IBM

IBM System/34 Basic Assembler Coding Form

GX21-8279-0
Printed in U.S.A.

PROGRAM BSASM3 | KEYING | GRAPHIC | PAGE 5 OF 5

PROGRAMMER | DATE | INSTRUCTIONS | CHARACTER | CARD ELECTRO NUMBER

STATEMENT

Name	Operation	Operand	Remarks	Identification Sequence
PTBUF	EQU	*		
	DC	CL40' '		
EBUF1	EQU	*		
	DC	CL40'ERROR WHILE RECEIVING FIRST FILE'		
EBUF2	EQU	*		
	DC	CL40'ERROR WHILE RECEIVING SECOND FILE'		
EBUF3	EQU	*		
	DC	CL40'ERROR WHILE TRANSMITTING FIRST FILE'		
EBUF4	EQU	*		
	DC	CL40'ERROR WHILE TRANSMITTING SECOND FILE'		
EOJMSG	EQU	*		
	DC	CL40'JOB SUCCESSFULLY COMPLETED'		
DKMSG	EQU	*		
	DC	CL40'DISK ERROR WHILE GETTING A RECORD'		
*				
	ORG	*,8,0	ALIGN ON 8-BYTE BOUND	
PTIO	EQU	*		
	DC	XL59'00'	PRINT I/O BUFFER	
	ORG	*,8,0	ALIGN ON 8-BYTE BOUND	
DKIO	EQU	*		
	DC	551XL1'00'	DISK I/O BUFFER	
	SDTFO	DISK-Y,BSC-Y,FIELD-Y,PRT-Y		
	END			

Figure 3-3 (Part 3 of 3). Transmit and Receive Program

Chapter 4. MULTI-LEAVING Remote Job Entry Utility–MRJE

INTRODUCTION

The System/34 System Support Program Product includes the MRJE (MULTI-LEAVING Remote Job Entry) utility, which allows for the submission of jobs to a host computer for processing and for the receipt of output from the host computer. MULTI-LEAVING is a communications technique that consists of the fully synchronized, bidirectional transmission of a variable number of data streams between two or more computers using the BSC (binary synchronous communications) facilities. MULTI-LEAVING permits maximum overlap of input and output operations in the MRJE utility and the host system by mixing input and output data streams on the communications line.

Use of MRJE

A System/34 with communications capabilities can use the MRJE utility to provide an effective and efficient communications link with a remote host system. MRJE is designed for applications that require the collection of data at the System/34, the processing of that data at the host system, and the return of the job output to the System/34.

An example of this type of application is a situation where the System/34 is located in a regional sales office in Chicago, and the host system is located at corporate headquarters in New York. The regional sales office sends weekly sales totals to the host system, which processes the data and returns printed reports to the regional office. In addition, the host system updates the corporate sales files and generates weekly sales reports for corporate management, giving sales figures for the entire corporation.

The System/34 in Chicago has communications capabilities and is configured with the MRJE utility. An application program has been written that allows the operators to enter sales invoices directly into a System/34 disk file. A file has also been generated that contains the needed MRJE commands and the Job Control Language (JCL) statements for the host. The host system in New York is an IBM System/370 with HASP II running under the OS/VS2 operating system. Figure 4-1 illustrates this application environment.

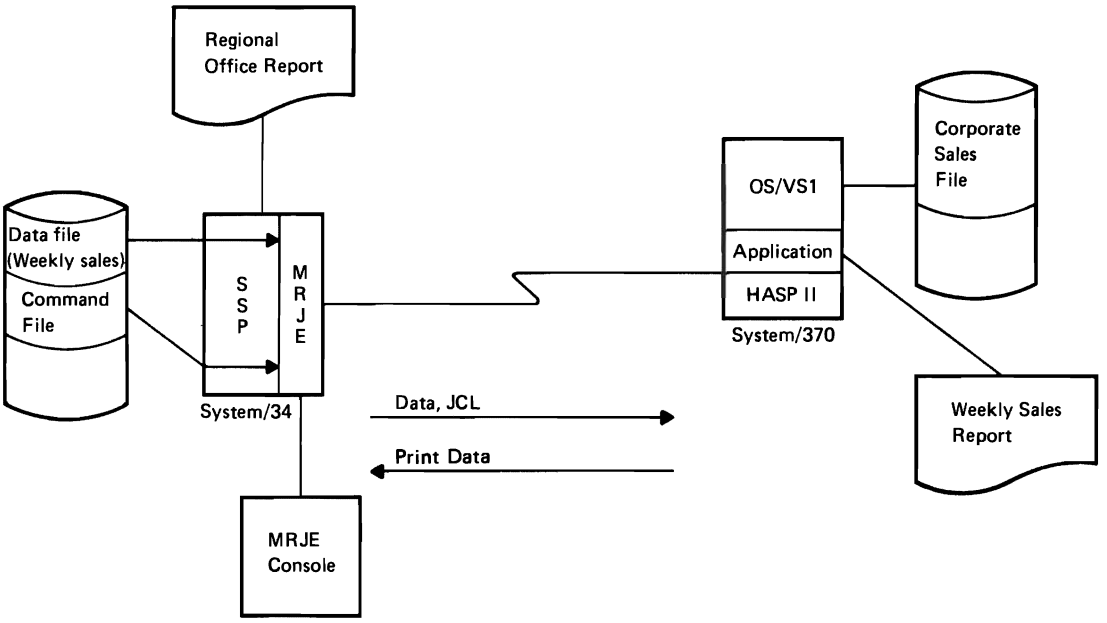


Figure 4-1. MRJE Application Environment

From Monday through Friday, the operators in the Chicago office enter the sales invoice information into the disk file on the System/34. On Friday afternoon, the operator activates the MRJE utility to transmit the data to the host system. The operator first specifies the amount of memory needed to run the MRJE session and then enters the MRJE procedure command to initiate the MRJE utility. When the initialization displays are presented, the operator specifies the appropriate parameters, including specifying the names of the files that contain the host JCL and MRJE statements and the data itself. All other parameters are used as they appear on the display. The displays and the parameters are discussed later in this chapter. After you have entered the necessary parameters, press the Enter/Rec Adv key and MRJE establishes the communications link with the host system.

Once the communications link is established, MRJE reads the file that contains the host JCL and MRJE statements. The appropriate job at the host system is selected and the data file is read by the System/34 and transmitted to the host. The data is processed at the host system and the output data is returned to the System/34 and printed. Once all data has been received, the System/34 operator terminates the MRJE communications session.

MRJE Environment

The MRJE utility operates in a binary synchronous communications (BSC) environment using EBCDIC data. MRJE operates in either a switched or nonswitched network, therefore the utility can communicate with the following host systems:

- ASP under OS/VS2
- HASP II under OS/VS2
- RES under OS/VS1
- JES2 under OS/VS2
- JES3 under OS/VS2
- VM/370 RSCS

Host considerations for communicating using RES, ASP, and HASP II are given later in this section.

You should have the appropriate host documentation on hand to properly operate with your host system. Some of this documentation is referenced under *Related Publications*.

MRJE CONFIGURATION

The MRJE utility is a part of the System/34 SSP and can be included with the SSP during System/34 configuration. To include MRJE, you must select it on the communications support display (display 9.0) during configuration. By selecting MRJE, you are also indicating that the BSC support for the MRJE utility is to be included as a part of the SSP.

When the communications support display is presented during CNFIGSSP, you select MRJE by entering a 1 for the MRJE support prompt. If you are going to be using autocall, you must select 1 for MLCA Support and 1 for Autocall Feature Support. If you are going to be using X.21, you must select 1 for MLCA support.

9.0 Communications Support

```
1. BSC Support . . . . . (0-No,1-Yes). .0
2. MRJE Support . . . . . (0-No,1-Yes). .1
3. SRJE Support . . . . . (0-No,1-Yes). .0
4. Secondary SNA/SDLC Support? . . . . . (0-No,1-Yes). .0
5. Remote Work Station Support? . . . . . (0-No,1-Yes). .0
6. SSP-ICF Support? . . . . . (0-No,1-Yes). .0
7. MLCA Support? . . . . . (0-No,1-Yes). .0
8. Autocall Feature Support? . . . . . (0-No,1-Yes). .0
```

For additional information, refer to the *Installation and Modification Reference Manual*.

DATA SECURITY AND MRJE

If any of the files you intend to use with MRJE are protected under the resource security capablities of the System/34, you must ensure that the operator who signs on to MRJE is authorized to use those files. Security of files and libraries is based on the user ID and password.

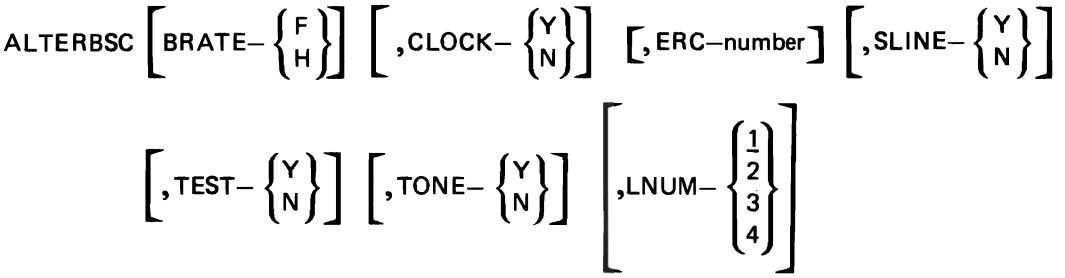
Whenever MRJE attempts to access a file or library (specified on the initialization display or in the READFILE or LIBRARY statement) and the system determines that the file or library is protected, the user ID is compared to the list of authorized users. If the user ID is valid, MRJE can access the file. If the user is not authorized, an error message is displayed.

MODIFYING THE MRJE ENVIRONMENT

The ALTERBSC and OVERRIDE procedures can be used to modify the environment in which MRJE is to run.

ALTERBSC Procedure

The ALTERBSC procedure is used to modify parameters for the communications adapter. The STATUS COMM control command can be used to determine the current values before the ALTERBSC procedure is used. Use of the ALTERBSC procedure alters the values only in the communications configuration record for the display station that requested the procedure. The new values apply only if MRJE is initiated from the same display station that ran the ALTERBSC procedure. The ALTERBSC procedure is supported by the System/34 HELP facility.



BRATE: BRATE-F specifies that the full rated speed of the modem is to be used. BRATE-H specifies that the modem is to be used at one half its rated speed.

CLOCK: CLOCK-Y specifies that the System/34 must provide the programmed clocking facility. CLOCK-N specifies that an external source, such as the modem, is to provide the clocking facility.

ERC: ERC-nnn specifies the number of error retries that are to be attempted. Any decimal number from 1 to 255 can be specified.

SLINE: SLINE-Y specifies that a switched network backup line is to be used as the backup (standby) for the nonswitched primary line. SLINE-N specifies that a switched network backup line is not used.

TEST: TEST-Y specifies that an IBM modem is being used and that modem testing will be performed when a permanent error occurs. TEST-N specifies that a non-IBM modem is being used and that modem testing will not be automatically performed when an error occurs.

TONE: TONE-Y specifies that a non-US special tone is required for manual answer and automatic answer. TONE-N specifies that a non-US special tone is not required.

LNUM: LNUM specifies the number of the communications line for which the parameters being defined by ALTERBSC are to apply. Valid entries are 1, 2, 3, or 4, with the default being 1.

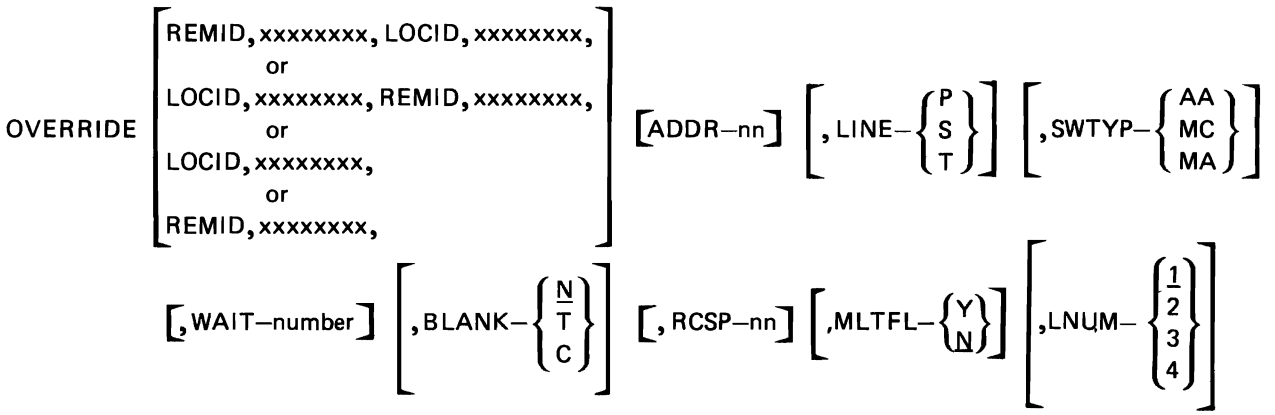
Any changes made with the ALTERBSC procedure remain in effect until:

- The items are changed by the ALTERBSC procedure or the $SETCF utility.
- The system is configured again.
- The system library is reloaded. The parameters are then set as follows:
  - CLOCK, TEST, and TONE are set to the values specified during microcode configuration.
  - The switched network backup line is not used (SLINE-N).
  - The full rated speed of the modem is used (BRATE-F).
  - The error retry count (ERC) from the executing program is used.

Additional BSC operational items that can be altered are included in the OVERRIDE procedure. To identify the current values of these parameters, use the STATUS control command.

OVERRIDE Procedure

The OVERRIDE procedure is used to modify the BSC parameters set by a user program, or to specify those parameters that cannot be set within a user program. You can use the STATUS COMM control command to determine the current values of the parameters before the OVERRIDE procedure is used.



REMID: Specifies the remote station switched line ID, with *xxxxxxxx* representing the ID. The ID is specified in hexadecimal, and can be 2, 4, 6, or 8 hexadecimal digits.

LOCID: Specifies the local station switched line ID, with *xxxxxxxx* representing the ID. The ID is specified in hexadecimal and can be 2, 4, 6, or 8 hexadecimal digits.

ADDR: Specifies the hexadecimal representation of one of the tributary station addressing characters. Tributary station addressing characters are always used in pairs, so the value specified here is duplicated by the system. The addressing characters correspond to polling characters. For example, the addressing characters SS (hexadecimal E2E2) correspond to the polling characters BB (hexadecimal C2C2). Refer to Appendix C for more information on addressing and polling characters.

LINE: Specifies the line facility. LINE-P specifies a point-to-point nonswitched line; LINE-S specifies a point-to-point switched line; and LINE-T specifies a tributary station on a multipoint line. If the LINE parameter is not specified, the value in the communications configuration record remains unchanged.

SWTYP: Specifies how a switched line is to be handled. SWTYP-AA specifies that if the modem is in automatic answer mode, the System/34 will automatically answer an incoming call. SWTYP-MC specifies that the operator must manually initiate the call. SWTYP-MA specifies that the operator must manually answer an incoming call. If autocall is used on the line being altered by the OVERRIDE procedure, the switch type specified here is ignored.

WAIT: Specifies the number of seconds that the BSC program is to wait before it indicates a permanent error condition when no messages have been sent or received. The value specified can be any decimal value from 1 to 999.

BLANK: Specifies whether 3780-format blank compression or truncation is to be used. BLANK-C specifies that blank compression is to be used; BLANK-T specifies that blank truncation is to be used; and BLANK-N, the default value, specifies that neither compression nor truncation is to be used.

RCSP: Specifies the record separator that is to be used. The hexadecimal representation of the desired character is specified by *nn*. If you do not define a record separator and the BLANK parameter is specified as either BLANK-C or BLANK-T, then the record separator used is hex 1E. If you do not specify the RCSP parameter and BLANK-N is specified, no record separator is used.

MLTFL: MLTFL-Y specifies that 3740 multiple files can be transmitted or received with null records separating the files. MLTFL-N specifies that 3740 multiple files are not supported. If MLTFL is not specified, MLTFL-N is assumed.

LNUM: Specifies the communications line for which the override is to apply. Valid entries are 1, 2, 3, or 4. The default is 1.

Running the OVERRIDE procedure alters the values only in the communications configuration record for the display station that requested OVERRIDE. The new values apply only if MRJE is initiated from that display station.

Any changes made by the OVERRIDE procedure remain in effect until OVERRIDE is run again or until the system library is reloaded.

STORAGE REQUIREMENTS

Before starting the MRJE utility, you must allocate sufficient storage for the utility. Storage is allocated by using the REGION OCL statement.

```
// REGION SIZE-nn
```

SIZE: Specifies the amount of storage, in K-bytes, needed to run this MRJE session. The maximum size that can be specified is 60 K bytes.

The size specified includes both the swappable and nonswappable storage requirements. The portion of the MRJE utility that must always be in memory is stored in the nonswappable storage area. Nonswappable storage contains the communications buffers and the BSC task. Swappable storage contains the portion of the MRJE utility that can be swapped to disk if another program requires the storage area. Additional swappable storage is required for certain configurations if compression of duplicate characters is desired. Compression is selected during the initialization of MRJE and is discussed later in this chapter.

To determine the region size for a particular configuration, perform the following steps:

1. Determine the swappable storage size from the following chart:

Configuration	No Full Compression	Full Compression
1 reader no punch	12 K	14 K
2 readers no punch	16 K	16 K
3 readers no punch	18 K	18 K
1 reader 1 punch	16 K	16 K
2 readers 1 punch	18 K	18 K
3 readers 1 punch	20 K	20 K

2. Determine the nonswappable storage requirements:
   a. Count the total number of reader, print, and punch tasks.
   b. Add two to the number of tasks (obtained in step a) to determine the minimum number of buffers.
   c. Round the buffer size up to the next higher multiple of eight. The buffer size is defined on the initialization display during the initialization phase. The buffer size depends on the value defined at the host system.
   d. Multiply the rounded buffer size (obtained in step c) by the number of buffers (obtained in step b).
   e. Add 3296 (for the BSC task) to the value obtained in step d.
   f. Divide the total obtained in step e by 1024 to determine the number of K-bytes.
   g. Round the quotient obtained in step f up to the next higher even integer. This value represents the amount of nonswappable storage required for the MRJE utility.

3. Add the totals obtained in steps 1 and 2 together to determine the total region size required. Use this value in your REGION statement.

The following example shows how the storage requirements are determined for a configuration with one reader, one printer, no punch, no full compression, and 200-byte buffers.

1. Using the chart, we find that swappable storage is 12 K bytes.

2. Nonswappable storage is calculated:
   a. Number of tasks is 2 (1 reader + 1 printer).
   b. Number of buffers is 4 (2 obtained in step a plus 2 additional).
   c. Buffer size = 200, which is already a multiple of 8.
   d. (200 bytes per buffer * 4 buffers) = 800 bytes.
   e. 800 bytes, obtained in step d, + 3296 bytes (for the BSC task) = 4096 bytes.
   f. 4096 bytes / 1024 bytes = 4 K bytes.
   g. 4 is already even.

3. Total storage: 12 K + 4 K = 16 K.

The following example shows the calculations for a configuration with one printer, one punch, three readers, full compression, and 344-byte buffers.

1. Using the chart, we find that swappable storage is 20 K bytes.

2. Nonswappable storage is calculated:
   a. Number of tasks = 5 (1 printer, 1 punch, 3 readers).
   b. Number of buffers is 7 (5 obtained in step a plus 2 additional).
   c. Buffer size is 344, already a multiple of 8.
   d. 344 bytes per buffer * 7 buffers = 2408 bytes.
   e. 2408 bytes + 3296 bytes (for the BSC task) = 5704 bytes.
   f. 5704 bytes / 1024 = 5.57 K bytes.
   g. 5.57 rounded up to the next higher even integer is 6. Since there is now enough storage available for an additional buffer, MRJE allocates 8 buffers instead of 7.

3. Total storage required is 26 K (20 K + 6 K).

RUNNING AN MRJE SESSION

An MRJE session consists of four phases, as shown in Figure 4-2.

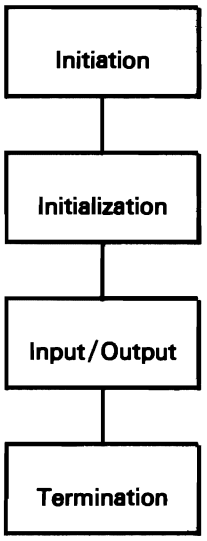


Figure 4-2. Phases of an MRJE Session

Each of these phases is performed in the order shown. The initiation phase is the loading of the MRJE utility; the initialization phase is the defining of the environment for the session; the input/output phase is the actual communications with the host; and the termination phase provides for the orderly termination of the MRJE utility. A description of each phase follows.

Initiation

The MRJE utility can be initiated from any display station. The display station that first initiates the MRJE utility becomes the MRJE console for the session. The MRJE console can operate as a reader input device or as an MRJE console. The modes of operation for the console are described later in this chapter. Before you initiate MRJE, you must have allocated sufficient storage area for the utility by using the REGION OCL statement, as previously described in this chapter. The MRJE utility is initiated by the MRJE procedure command.

```
MRJE [DISPLAY- {name | #MR01}] [,AUTO-{N | Y}] [,LINE-{1 | 2 | 3 | 4}] [,PHONE-list name]
```

(Defaults, underlined in the diagram: #MR01, N, 1)

DISPLAY: Specifies the name of the initialization format for the display. If the display parameter is not specified, the IBM-supplied initialization display format, named #MR01, is used. The #MR01 format has all values set for communicatons with a HASP II host system.

You can specify the name of a display screen format that you created using $SFGR, the screen format generator utility. Information on altering the display screen format can be found later in this chapter.

AUTO: Specifies the mode of operation for this session of the MRJE utility. AUTO-N specifies attended mode of operation. During attended mode, an operator must be present to respond to MRJE messages and to specify initialization parameters. If the AUTO parameter is omitted, AUTO-N is assumed.

AUTO-Y specifies unattended mode of operation. During unattended mode, MRJE uses default values whenever a response to a message is required. The initialization parameters are taken from the initialization display format specified, and the operator cannot change the parameter values. It is essential that you specify the correct name for the DISPLAY parameter to ensure that MRJE operates correctly in unattended mode.

Figure 4-3 illustrates the differences between MRJE attended and unattended modes during initiation.

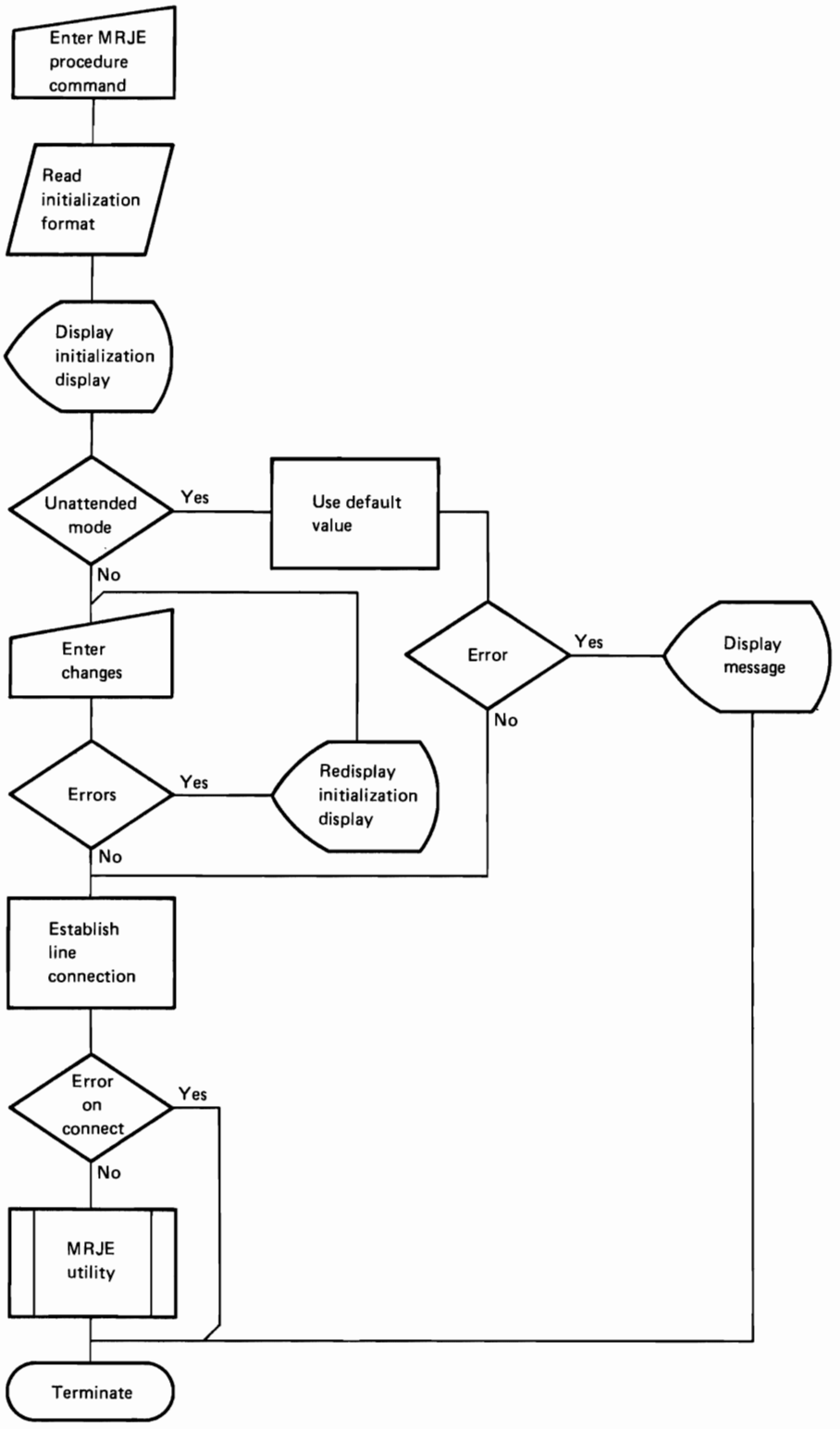


Figure 4-3. Initiation Differences Between Unattended and Attended Modes

LINE: Specifies the communications line on which the MRJE session is to take place. The value entered can be 1, 2, 3, or 4. Lines 3 and 4 are only available if your system has the Multiline Communications Adapter. The default is 1.

PHONE: Specifies the name of the list to be used for autocall or switched X.21. The phone list is created by the DEFINEPN procedure or the DEFINX21 procedure, which are described in the *System Support Reference Manual*. The phone list specified must be in either the system library or the current user library. This parameter applies only if your system has the Multiline Communications Adapter with the Autocall feature or the switched X.21 feature for the specified communications line.

A phone list containing multiple numbers is practical for MRJE only if all the numbers are for the same location. If the first number cannot be reached, the next is tried. Because MRJE is a MRT program, the MRJE procedure goes to end of job when it goes to end of step. When the procedure goes to end of job, the phone list is freed from main storage. If the procedure is executed again, a new list is brought into main storage and the first number in that list is called.

Note: The OCL generated by the MRJE procedure command can be altered. However, this is not recommended. Use of the PRINTER OCL statement with the MRJE OCL is not supported. Refer to the *System Support Reference Manual* for more information.

Initialization

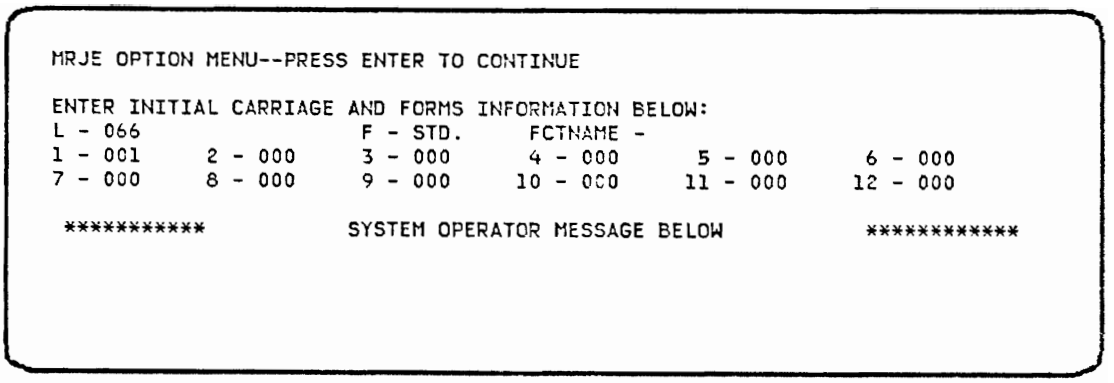
After you have entered the MRJE procedure command to initiate MRJE, the initialization display appears. If you specified unattended mode in the MRJE procedure command, the display appears and then disappears almost immediately. You cannot change any of the values on the screen in unattended mode. In attended mode, the screen appears and you are able to change the entries. In either mode, the MRJE console enters console mode as soon as initialization is completed.

The following represents the display for a 1920-character screen:

```
MRJE OPTION MENU--PRESS ENTER TO CONTINUE


ENTER INITIAL CONFIGURATION INFORMATION BELOW:
HOST - H          LEN - 0400        XPC -  N          COM - C
PR1 - P           PU1 - N           FSN -  001        DID - 9
SPCPR1 - 0060     SPCPU1 - 0060     STDPR1 - 0060     RDN - 1
RD1NAME -         RD1DATE - 000000  RD1TYPE - D       RD1CMD - Y
RD1LIBR - 0       RD1DEL - N        RD1 -

ENTER LOGON OR SIGNON INFORMATION BELOW:
/*SIGNON

ENTER INITIAL CARRIAGE AND FORMS INFORMATION BELOW:
L - 066               F - STD.     FCTNAME -
1 - 001     2 - 000     3 - 000      4 - 000      5 - 000     6 - 000
7 - 000     8 - 000     9 - 000     10 - 000     11 - 000    12 - 000


************          SYSTEM OPERATOR MESSAGE BELOW             ************
```

For a 960-character screen:

```
MRJE OPTION MENU--PRESS ENTER TO CONTINUE

ENTER INITIAL CONFIGURATION INFORMATION BELOW:
HOST - H          LEN - 0400        XPC - N           COM - C
PR1 - P           PU1 - N           FSN - 001         DID - 9
SPCPR1 - 0060     SPCPU1 - 0060     STDPR1 - 0060     RDN - 1
RD1NAME -         RD1DATE - 000000  RD1TYPE - D       RD1CMD - Y
RD1LIBR - 0       RD1DEL - N        RD1 -

ENTER LOGON OR SIGNON INFORMATION BELOW:
```



The initialization screen consists of a set of parameters with a default value given for each parameter. If you want to run MRJE with the default values, simply press the Enter/Rec Adv key. If you want to change any value, position the cursor under the value you want to change and enter the new value. After you have made all the changes, press the Enter/Rec Adv key.

The parameters and the options for each are discussed in the following paragraphs.

HOST: Specifies the type of host system to be communicated with during the MRJE session. Valid entries are:

Entry	Description
H	HASP II under OS/VS2. This is the default on the IBM-supplied initialization display.
A	ASP under OS/VS2.
R	RES under OS/VS1.
J2	JES2 under OS/VS2.
J3	JES3 under OS/VS2.
VM	VM/370 RSCS.

LEN: Specifies the length, in bytes, of the MRJE utility buffers. The value is specified in decimal. The length you specify must be equal to the BSC buffer length defined by the host system. This value is also used in calculating the region size for an MRJE session. The default on the IBM-supplied initialization display is 400.

The standard values for each host system are:

Host	Entry
H	400 for HASP II.
A	400 for ASP.
R	512 for RES.
J2	512 for JES2.
J3	512 for JES3.
VM	400 for VM/370 RSCS.

XPC: Specifies whether text transparency is to be used. The use of text transparency allows the transmission of any of the 256 hexadecimal byte values. In nontransparent mode, certain hexadecimal codes (such as 01, 02, 03, and 10) are reserved for use as control characters and cannot be sent as data. Use of transparency permits all hexadecimal codes to be transmitted as data.

XPC-N, the default on the IBM-supplied initialization display, specifies that text transparency is not to be used. XPC-Y specifies that text transparency is to be used.

If you intend to use text transparency, you must ensure that the host system is also using text transparency. In general, nontransparent mode is used unless you have data to be sent or received that might contain hexadecimal values that correspond to control characters (generally any value less than hex 40).

COM: Specifies whether duplicate characters are to be compressed for transmission to the host system. For example, if you have a string of 30 A's, the use of compression causes that string to be replaced by a control character containing the count of the number of characters (30), and one A. Only two characters are transmitted. Without compression, all 30 characters are transmitted. Each control character can contain a count up to 31. If more than 31 duplicate characters are compressed, the control character and character sequence is repeated.

The primary advantage of compression is that communications line time is decreased since fewer actual characters are transmitted, but System/34 processing time increases to perform compression. The line time savings can be significant for large data sets and for data sets with many duplicate characters. The use of compression requires an extra 2 K bytes of storage for some MRJE configurations. Refer to *Storage Requirements* earlier in this chapter for information on the storage needed for MRJE.

COM-C specifies the compression of duplicate characters and is the default on the IBM-supplied initialization display. COM-N specifies that compression of duplicate characters is not to be used.

Note that this parameter applies only to System/34 to host transmissions; data sent from the host to the System/34 is always compressed.

PR1: Assigns the print task. PR1-P specifies that print output received from the host is to go to the printer associated with the MRJE console. This is the default on the IBM-supplied initialization display. PR1-D specifies that print output received from the host is to be stored in the standard forms disk file, TDISKPR1 (using TDISKPR1 is described in *Print Output* later in this chapter). PR1-N specifies that print output from the host will be ignored.

After MRJE has been initialized, you can reassign the print task by using the MODIFY utility control statement.

PU1: Assigns the punch task. PU1-D specifies that punch output received from the host is to be stored on disk in a special forms file (special forms files and their use are described in *Punch Output* later in this chapter). PU1-N specifies that punch output will not be accepted by the MRJE utility. PU1-N is the default on the IBM-supplied initialization display.

During the MRJE session, you can reassign the punch task from the D option to the N option by using the MODIFY utility control statement. The MODIFY statement can be used to change the option from N to D only if the D option was selected during initialization and the MODIFY statement was subsequently used to change the option to N.

If the N option is selected during initialization and the host system has punch output to send, you will not be able to terminate MRJE normally using the END utility control statement. You will have to use the System/34 inquiry display and select option 2 or option 3 to cancel the MRJE utility. Therefore, be certain that the host system will not be sending punch data before you specify the N option.

Punch data can be printed using the $DCSUP utility, which is described in Chapter 5. If your requirements call for punched cards, the punch data must be copied to diskette and processed at a system that can convert the data from diskette to punched cards.

FSN: Specifies the identifying number for the first special forms file on disk. The value specified must be in the range 001 to 999. Any print or punch data that requires a special forms file can be stored on disk with the output for each job being stored in a unique special forms file. The number specified for the *FSN* parameter is used to number the first special forms file and the number is incremented by one for each subsequent file created. If a file number is already in use, MRJE assigns the next higher available number.

Note that all punch output is sent to a special forms file.

DID: Specifies the single identification character that indicates whether print output is to go to a special forms file. Whenever a forms mount message is received from the host system, the first character of the form name or number is compared to the DID character. If the characters match, the output is written to a special forms file. If the characters do not match, one of the following occurs:

- If the print task is assigned to the printer (PR1-P), a forms mount message is sent to the system console operator if the output is going directly to the printer. If the output does not go directly to the printer, it is sent to the spool file.

- If the print task is assigned to disk (PR1-D), no forms mount message is issued and the output is written to TDISKPR1.

- If the print task is not active (PR1-N) or has been halted (PR1-H), the forms mount message and the print output are not transmitted by the host. If the print task is then activated (PR1-P) or reactivated (PR1-R), the forms mount message is sent to the system console operator.

Figure 4-4 illustrates the processing of a forms mount message when a DID character has been specified.

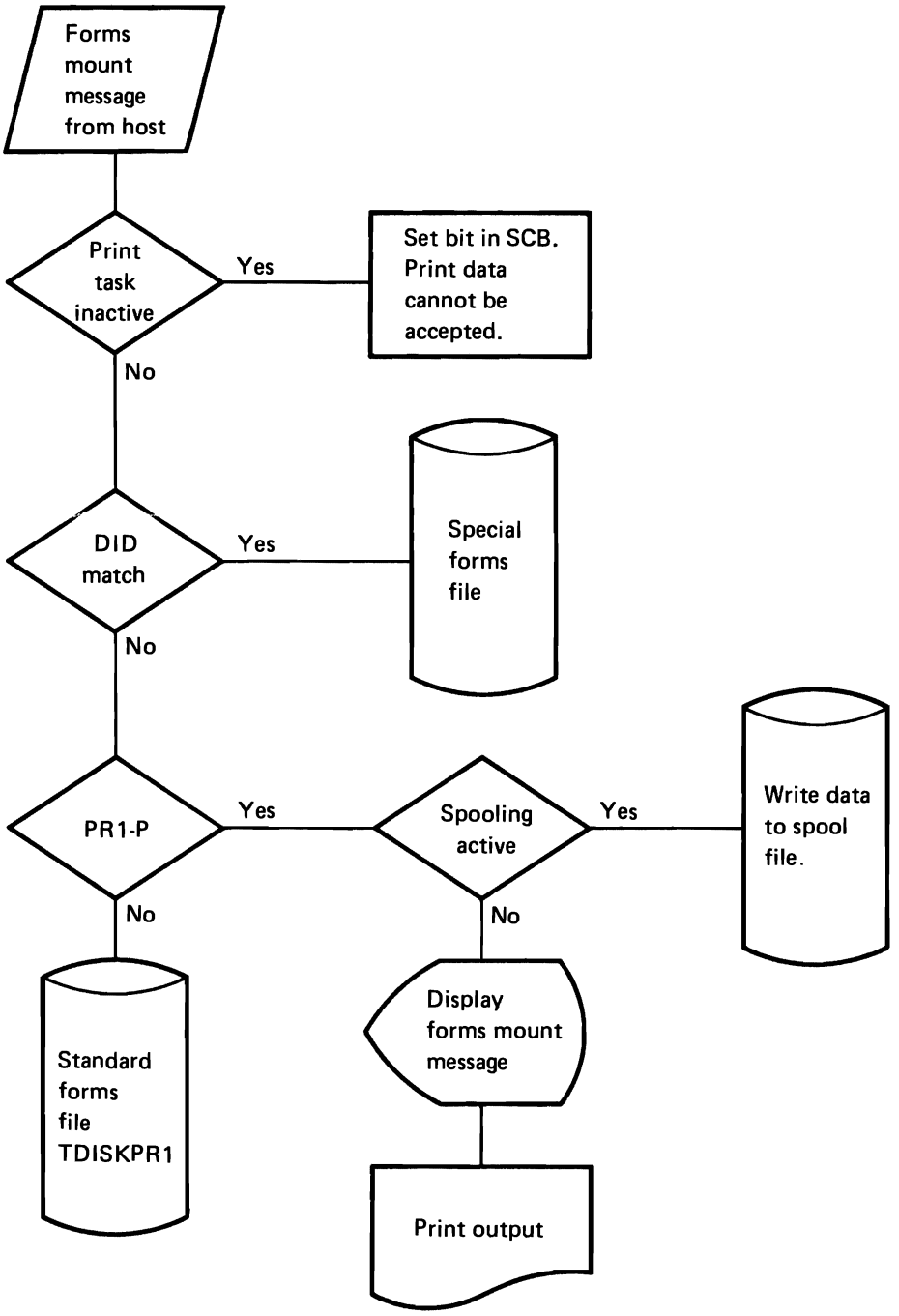


Figure 4-4. Forms Processing with DID Character

SPCPR1: Specifies the maximum number of blocks on a disk required to contain an individual special forms file that is created by the print task. The number of blocks can be up to four decimal digits. Unused space for each file is returned to the system when the file is completed.

SPCPU1: Specifies the maximum number of blocks on disk required to contain an individual special forms file that is created by the punch task. The number of blocks can be up to four decimal digits. Unused space for each file is returned to the system when the file is completed.

STDPR1: Specifies the maximum number of blocks on disk required for the printer output from an entire MRJE session that is directed to the standard forms file, TDISKPR1. The number of blocks specified can be up to four digits. Unused space is returned to the system when MRJE terminates.

RDN: Specifies the number of reader tasks that are to be allocated by MRJE. The value specified must be 1, 2, or 3. The number of reader tasks includes the console reader task and any reader tasks assigned to other display stations. The value specified must not be greater than the number of reader tasks defined at the host system. The default on the IBM-supplied initialization display is 1.

RD1NAME: Specifies the disk file or library member that is to be automatically read by the reader task. If a name is specified, that disk file or library member is read by MRJE upon completion of the initialization phase. If no entry is made, no file or library member is read automatically. The default on the IBM-supplied initialization display is a blank entry.

Note that if a name is specified by RD1NAME, the RD1 parameter (described later) cannot be specified. Conversely, if the RD1 parameter is specified, RD1NAME must be blank.

The RD1NAME parameter is only used to specify a file or library member that is to be automatically read after initialization. Other files or library members can be read by using the READFILE utility control statement, which is described later in this chapter.

The RD1NAME parameter must be specified for unattended mode to operate properly.

RD1DATE: Specifies the creation date of the disk file specified for the RD1NAME parameter. The RD1DATE parameter is used only when a file is specified for RD1NAME and more than one file with the specified name exists on disk. The date is entered in the system date format. If the RD1DATE parameter is not to be used and the RD1NAME parameter is specified, you must specify all blanks for RD1DATE. The default on the IBM-supplied initialization display is 000000.

RD1TYPE: Specifies the type of file or library member specified for RD1NAME. RD1TYPE-D specifies a disk file. This is the default of the IBM-supplied initialization display. RD1TYPE-S specifies a source member in the library defined by the RD1LIBR parameter (described later). RD1TYPE-P specifies a procedure member in the library defined by RD1LIBR (defined later).

RD1CMD: Specifies whether the file or library member specified for RD1NAME is a command file. RD1CMD-Y specifies that the file is a command file. When MRJE reads a command file, it processes any utility control statements it encounters. RD1CMD-Y is the default on the IBM-supplied initialization display.

RD1CMD-N specifies that the file is a data file. Any utility control statements in a data file are treated as data and transmitted to the host system. Utility control statements in a data file are not processed by MRJE.

RD1LIBR: Specifies the library that is to be searched for the member specified by the RD1NAME parameter. The default is zero (0) and indicates the system library.

RD1DEL: Specifies whether deleted records are to be transmitted to the host system. A data file can be either delete-capable or non-delete-capable. If a file is delete-capable, it contains records of hexadecimal FF to indicate deleted records.

RD1DEL-Y specifies that deleted records are to be transmitted to the host system.

RD1DEL-N specifies that deleted records are not to be transmitted to the host system. This is the default on the IBM-supplied initialization display.

The RD1DEL parameter is valid only if extended disk management has been configured on the system. RD1TYPE-D should be specified and the designated file must be delete capable. If deleted records are to be transmitted, the host system and the System/34 must be using transparent mode for data transmission.

RD1: Specifies whether the reader task is to be assigned to the MRJE console. RD1-K specifies that the reader task is to be assigned to the console. RD1- specifies that the console is reserved for console input. This is the default on the IBM-supplied initialization display.

If the RD1NAME parameter is specified, then RD1 must be blank. IF RD1 is specified, RD1NAME must be blank. Both parameters may be left blank. You can alter the assignment made by the RD1 parameter by using the MODIFY utility control statement. The EOF statement terminates the reader task and returns the console to console mode.

ENTER LOGON OR SIGN-ON INFORMATION BELOW: Specifies the sign-on or logon command that is to be sent to the host system when the communicaitons link is established. The command entered and the parameters must be in accordance with the expectations of the host system.

- The ASP and HASP II systems use the following command:

  /*SIGNON RMnnn password

  The command begins in column 1. The *RMnnn* parameter begins in column 16 and specifies the remote terminal number as defined at the host system in the BSCATERM card. The *password* parameter begins in column 25 and can be up to eight characters in length. If used, the password specified must match the password defined at the host for the terminal identified by *RMnnn*.

- The RSCS system uses the following command format:

  /*SIGNON REMOTErn password

  The command begins in column 1. The *REMOTErn* parameter begins in column 16 and defines the remote terminal number as specified in the START command at the host system. The *password* parameter, if used, begins in column 25 and must match the password specified at the host system in the START command for this terminal. (See *IBM Virtual Machine Facility/370: Remote Spooling Communications System (RSCS) User's Guide*, GC20-1816.)

- The JES2 system uses the following command format:

```
/*SIGNON  { RMTnn    }  password1  [password2]
         { REMOTEnn }
```

Column 1	Column 16	Column 25	Column 73
	a remote terminal number matching $.RMTSYS3 in the host's remote terminal control statement.	password matching LINEnn specified in host's initialization parameter.	password matching &PASSWD specified in host's remote terminal parameter.

- The JES3 system uses the following command format:

```
/*SIGNON  namex  [A]  password1  password2
```

Column 1	Column 16	Column 22	Column 25	Column 35
	5-character work station name matching RJPTERM in the host's initialization statement.	specifies programmable work station automatic reader.	1- to 8-character password matching RJPLINE in the host's initialization statement.	1- to 8-character password matching RJPTERM in the host's initialization statement.

- RES uses the following command format:

```
LOGON userid [/password]TERM(termid) [PROC(procname)] [NOTICES  ] [MAIL  ]
                                                      [NONOTICES]  [NOMAIL]
```

The command begins in column 1. The *userid* parameter identifies the terminal to the host system. The value entered must match the entry made at the host in the ADD subcommand to the ACCOUNT command. The *password* entry is optional and, if used, must be separated from *userid* by a slash (/). In addition, the value entered must match the entry made in the ADD subcommand at the host system for the specified user. *TERM(termid)* specifies the terminal identification number. The value entered can be from 1 to 200 and must match the entry made at the host for the TERMID parameter in the TERMINAL macro. The *PROC(procname)* parameter allows you to specify a catalogued procedure (procname) at the host that is to be executed as a part of the logon process. *NOTICES*, the default, allows you to specify that all general notices from the host are to be listed automatically after logon. *NONOTICES* inhibits the listing of notices. *MAIL*, the default, allows you to specify that all messages directed to your terminal be automatically listed after logon. *NOMAIL* inhibits the listing of the messages.

The next group of parameters is used to define carriage control and forms information.

L: Specifies the forms length, in lines per page, and may be any integer value from 1 to 112. The default on the IBM-supplied initialization display is 66.

F: Specifies the forms name or number (four characters). This value should be the same as the forms name or number that was on the printer at the end of your last MRJE session (the last session in which your user ID was used). The value must be left-justified to avoid embedded blanks in the names generated for the special forms disk file. As long as the host system has not been shut down since the last session, the host retains the name or number of the last form you used. When you sign off and then sign on again later, the host assumes that you have not changed forms since the last session.

STD. is the default on the IBM-supplied initialization display.

FCTNAME: Specifies the name of the file that contains the forms control table. The forms control table serves as a translation table by converting the forms name or number, which is in the host system's vocabulary, into a corresponding name or number that has more meaning to the System/34 user. When the host sends a forms mount message, the forms control table is searched for the host defined name or number. If the name or number is found, it is replaced by the corresponding user term and the message is sent. For example, assume that the host name for the forms used to print checks is X2QH, and that you want to refer to that form as CHEQ. Both these names were placed in the forms control table when it was created, so when a forms mount message contains the name X2QH, that name is replaced by CHEQ before the forms mount message is sent to the MRJE console operator. If the host system is JES3, all host forms names must be 8 characters in length.

The forms control table is generated by the $DCFUP utility, which is described in Chapter 6. The forms control table also defines other forms control parameters, including whether print output is to be spooled. Figure 4-5 illustrates the flow of processing when a forms control table is used.

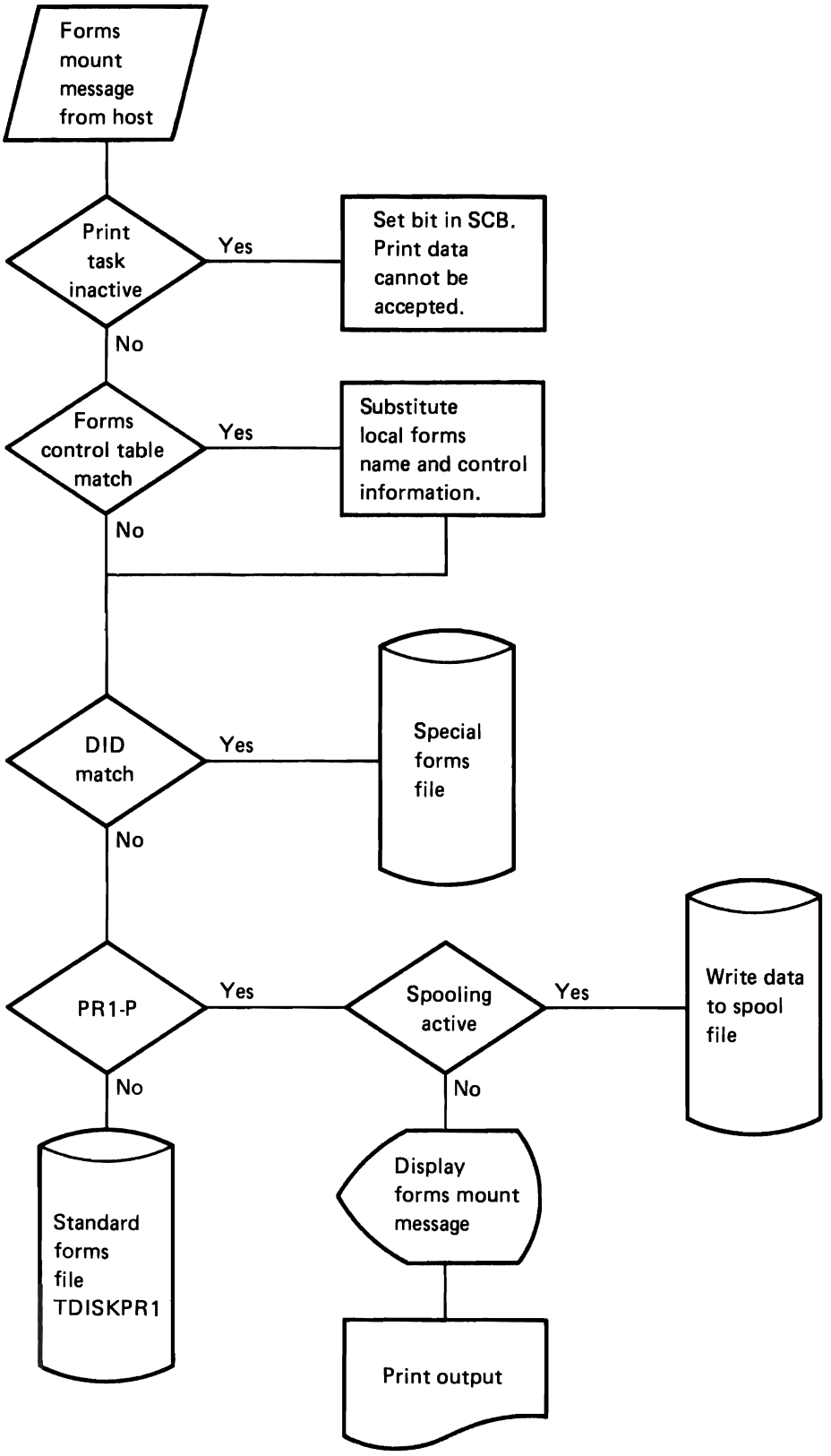


Figure 4-5. Processing with a Forms Control Table

The remaining parameters, the integers 1 through 12, are used to define forms movement. The value for each parameter can be any integer from 0 to the length specified for the L parameter. The value specified for a parameter identifies a line number within a page of a form. When the host sends print output to the MRJE utility, it will send 133 characters with the first character being a channel control character and the remaining 132 being the actual data. The first character is used to define how the form is to be moved before or after the data is printed. For example, if the channel control character 1 is received, the form will be moved to the line specified for the 1 parameter before printing the data. Assume that the channel control characters are defined as follows:

Character	Definition
1	20
2	0
3	55
4	0
5	0
6	80
7	70
8	0
9	0
10	0
11	0
12	0

The following data is then received from the host:

- *OUTPUT-A* with channel control character 1 specified
- *OUTPUT-B* with channel control character 3 specified.
- *OUTPUT-C* with channel control character 6 specified.
- *OUTPUT-D* with channel control character 7 specified.
- *OUTPUT-E* with channel control character 2 specified.

The data is then printed as follows:

- Skip to line 20 and print *OUTPUT-A*.
- Skip to line 55 and print *OUTPUT-B*.
- Skip to line 80 and print *OUTPUT-C*.
- Skip to line 70 on the next page and print *OUTPUT-D*.
- Space one line and print *OUTPUT-E*.

The defaults on the IBM-supplied initialization display are 1 for parameter 1 and 0 for all other parameters.

The last area on the initialization screen is the system operator message area. In this area, you can enter any message, up to 160 characters, that you want to appear on the status display of any System/34 display station that attaches to the MRJE utility.

After all changes have been made to the display, press the Enter/Rec Adv key and MRJE is initialized. If any errors are detected in the options entered, the errors appear in reverse image on the initialization display and the following message appears:

ERRORS – CORRECT FIELDS AND PRESS ENTER TO RETRY

For each field that appears in reverse image, position the cursor to that field and enter the appropriate correction. Once all corrections have been made, press the Enter/Rec Adv key again. If no invalid entries are detected, the MRJE initialization function proceeds.

Note: If you are running on a switched line and the System/34 is to place the call, the MRJE console operator is prompted to place the call. If autocall is used, this message does not appear.

Input

Input to the MRJE utility can come from the disk, a display station, or the MRJE console. The types of input allowed are data, host job control language (JCL) statements, host commands, and MRJE utility control statements. Note that not all types of input can be entered from all the devices. Under normal operating conditions, data, JCL, and host commands are sent to the host system for processing, but MRJE utility control statements are processed by the MRJE utility. Disk and display station input occurs under the control of a reader task; console input can be under the control of a reader task (only if the RD1-K parameter was selected on the initialization screen or specified by the MODIFY utility control statement) or the console input task.

The following table defines the input classifications and the device relationships.

Task	Input Device	Input Type
Reader	Disk Data file Source member Procedure member	Data, JCL, utility control statements[1]
	Display station keyboard	
	MRJE console keyboard in reader mode	
MRJE console	MRJE console keyboard in console mode	Host system commands, utility control statements

[1]Not all utility control statements are allowed from the reader task. Refer to the description of the utility control statements later in this chapter for more information.

Figure 4-6. Input Classifications and Device Relationships

Disk Input

Disk input to the MRJE utility is requested when you enter the READFILE utility control statement or specify the RD1NAME parameter on the initialization display. When the disk is to be used as an input device for MRJE, you specify the following:

- The name of the file or library member to be read.
- Whether the file or library member is a command file.
- The type of file or library member (disk file, source, procedure).
- The file creation date (optional and for disk files only).
- The name of the library to be searched for a member (optional and for library members only).
- Whether deleted records are to be transmitted to the host (optional and for disk files only).

For a more complete description of file specification, refer to the discussion of the initialization phase earlier in this chapter (specifically, the RD1NAME parameter) or the READFILE statement later in this chapter.

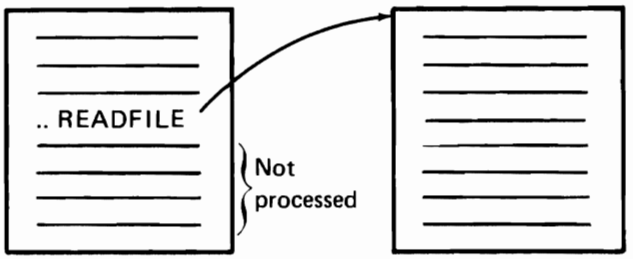
Disk files used as input to MRJE cannot be shared by other executing programs.

All files or library members used as input to MRJE are designated as being either command or data files. Both types can contain data and JCL, which are transmitted to the host system. A command file can also contain some of the MRJE utility control statements (READFILE, EOF, LIBRARY, and, if in unattended mode, END), which are processed by the MRJE utility rather than transmitted to the host system. If utility control statements are encountered by the reader task in a data file, they are ignored by the reader task and transmitted to the host system as data.

The MRJE utility transmits the information from data files as 80-byte records, regardless of the logical record length. For example, a data file containing four 60-byte logical records is transmitted as three 80-byte physical records. A data file containing four 120-byte logical records is transmitted as six 80-byte physical records. It is the responsibility of the host system program to return the data file to its original record length.

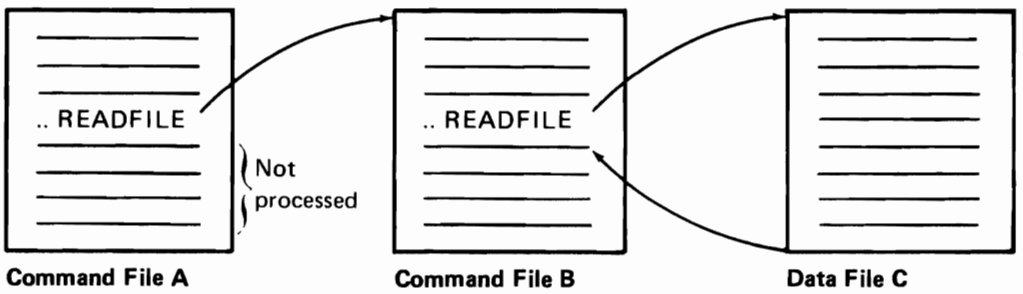
When a READFILE statement that specifies a data file is detected while the reader task is processing a command file, processing of the command file is suspended and the reader task begins processing the data file. Once the reader task completes the processing of the data file, it returns to the command file at the statement immediately following the READFILE statement that caused the data file to be processed.

When a READFILE statement that specifies another command file (B) is encountered by the reader task while processing a command file (A), processing of command file A terminates and the reader task begins processing command file B. When processing of command file B has completed, the reader task does not return to command file A for further processing. Any statements after the READFILE statement in command file A are not processed, as shown by the following.



The reader task always terminates when the end of a command file is reached.

To further illustrate the use of disk files as input to MRJE, assume that you have three files – command file A, command file B, and data file C. Command file A was defined for the RD1NAME parameter and specified as being a command file (RD1CMD-Y specified) during the initialization phase of MRJE, and MRJE began reading it upon completion of the initialization phase. While processing command file A, the reader task detects a READFILE statement before the end of command file A. The READFILE statement specifies that command file B is to be processed. Therefore, the reader task terminates its processing of command file A and begins processing command file B. While processing command file B, the reader task detects a READFILE statement that specifies data file C. The reader task suspends its processing of command file B and begins reading data file C. When the reader task completes its processing of data file C, it returns to command file B and begins reading immediately after the READFILE statement that specifies data file C. Any statements after the READFILE statement in command file A that specified command file B are not processed. The following graphically illustrates this example.



Display Station Input to the Reader Task

To use a display station as an input device for the MRJE reader task, you must have declared more than one reader during initialization by specifying RDN-2 or RDN-3. To activate the reader task at a display station, enter the MRJE procedure command with no parameters. If you are using a 1920-character display, the following MRJE status menu display appears:

```
MRJE STATUS MENU--PRESS ENTER TO CONTINUE



HOST - XX          LEN - XXX          XPC - X          COM - X
PR1 - X            PU1 - X            FSN - XXX        DID - X
SPCPR1 - XXXX      SPCPU1 - XXXX      STDPR1 - XXXX    RDN - X



LOGON/SIGNON INFORMATION
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

CARRIAGE AND FORMS INFORMATION
L - XXX                 F - XXXX      FCTNAME - XXXXXXXX
1 - XXX     2 - XXX     3 - XXX        4 - XXX    5 - XXX     6 - XXX
7 - XXX     8 - XXX     9 - XXX       10 - XXX   11 - XXX    12 - XXX



***********             SYSTEM OPERATOR MESSAGE         *****************
```

If the MRJE command is entered from a 960-character display station, the following two MRJE status displays are presented:

```
MRJE STATUS MENU--PRESS ENTER TO CONTINUE

ENTER INITIAL CONFIGURATION INFORMATION BELOW:
HOST - XX          LEN - XXX          XPC - X          COM - X
PR1 - X            PU1 - X            FSN - XXX        DID - X
SPCPR1 - XXXX      SPCPU1 - XXXX      STDPR1 - XXXX    RDN - X



LOGON/SIGNON INFORMATION
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
MRJE STATUS MENU--PRESS ENTER TO CONTINUE

CARRIAGE AND FORMS INFORMATION BELOW:

L - XXX                 F - XXXX    FCTNAME - XXXXXXXX
1 - XXX     2 - XXX     3 - XXX     4 - XXX     5 - XXX     6 - XXX
7 - XXX     8 - XXX     9 - XXX    10 - XXX    11 - XXX    12 - XXX

**********              SYSTEM OPERATOR MESSAGE BELOW        **********
```

The x values on the displays indicate the status of the MRJE configuration specified during the initialization phase. You cannot change these values on the status display. Press the Enter/Rec Adv key to activate the reader task. The MRJE input display then appears on the display screen.

```
MRJE INPUT-OUTPUT
_____________________________________________________________________


_____________________________________________________________________
MRJE READER INPUT
```

The reader task at a display station operates as if it were reading a disk command file. Input from the station can be data, JCL, and certain utility control statements (READFILE, EOF, and LIBRARY). Records input from the display station can be up to 80 characters in length. When a READFILE utility control statement is entered from a display station while the reader task is active, the reader task begins processing the specified file. The reader task is suspended at the display station until the file has been processed. A message is then displayed at the display station, indicating that the file has been processed and that the reader task is active at the display station. The display station is not available for input operations while the file is being processed.

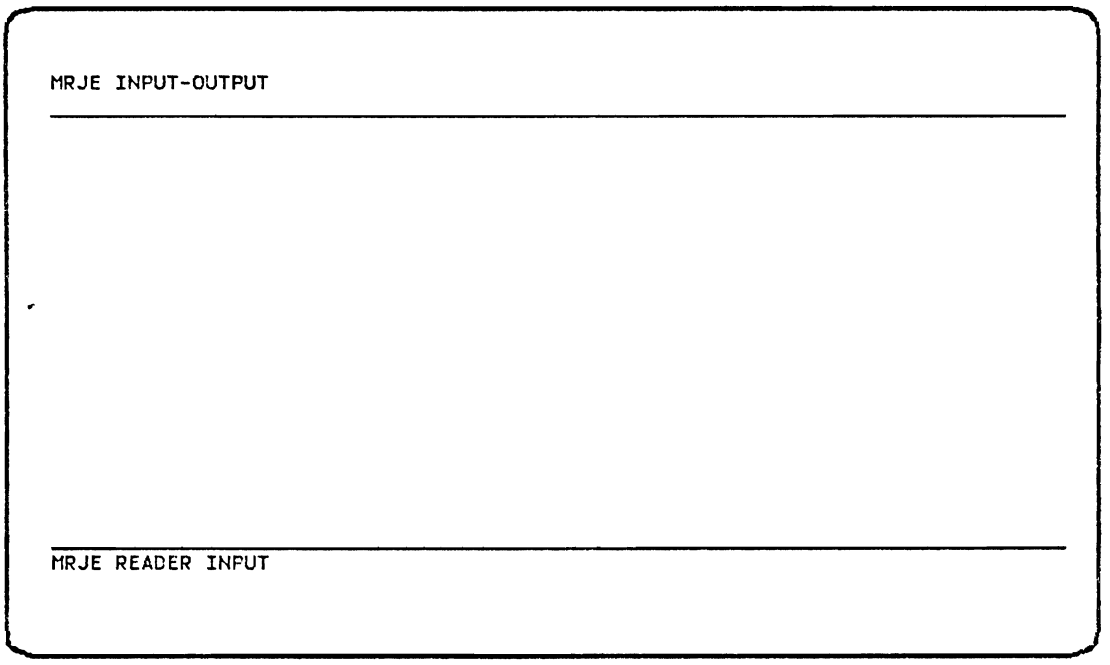
The reader task terminates at a display station when any of the following conditions occur:

- An EOF utility control statement is entered from the display station.
- A CANCEL command for the display station reader task is entered at the MRJE console.
- An EOF utility control statement is encountered in a command file, the reading of which was specified by a READFILE utility control statement entered from the station.
- The physical end of a command file is reached.

When the reader task terminates at a display station, press the Enter/Rec Adv key to return to the System/34 command display.

Console Input to the Reader Task

If you intend to use the MRJE console as an input device for the reader task, you must either have specified RD1-K on the initialization display or have used the MODIFY utility control statement to designate the console as a reader device. When the console is ready to accept reader input, the following display appears:



When the reader task is activated at the console, it processes the input from the console in the same manner as it processes a disk command file. Input from the console can be data, JCL, or certain utility control statements (READFILE, EOF, and LIBRARY). Input is entered one record at a time, with each record containing up to 80 characters.

Termination of the reader task at the console occurs under any of the following conditions:

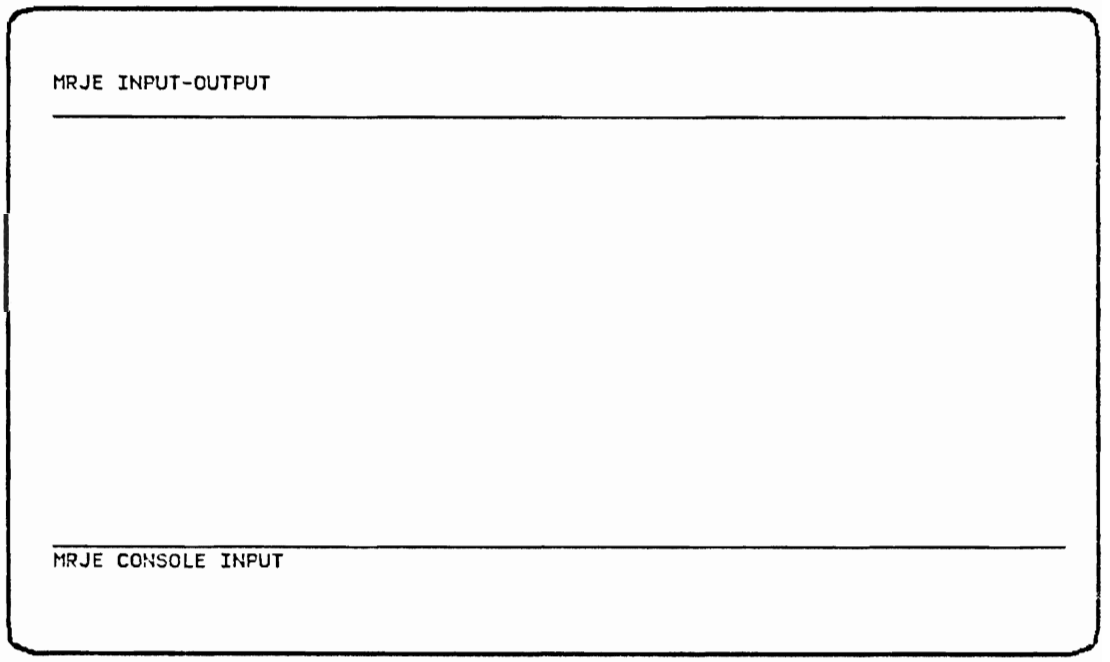
- An EOF utility control statement is entered from the console.
- A CANCEL command for the reader task at the console is entered from the console while it is in console mode.
- An EOF utility control statement is encountered within a command file, the reading of which was specified by a READFILE statement entered from the console.
- The physical end of file is reached while a command file is being processed.

Once the reader task terminates at the console, the console is automatically prepared to accept console input (the console task is activated).

When a READFILE statement is entered from the MRJE console while the reader task is active, the reader task begins processing the specified file and the console task is activated at the MRJE console. The console task remains active until MRJE has completed processing the specified file. If the file is a command file, the console task remains active after processing of the file completes. If the file is a data file, a message appears on the console screen indicating that the file has been processed. To return to using the console as an input device for the reader task, press the Enter/Rec Adv key and the reader input field display appears.

Console Input to the Console Task

Input to the console task can occur only at the MRJE console. In console input mode, the display appears as follows:



The space between the lines contains the host and MRJE messages, which are rolled on the screen. The messages are also logged to the system history file.

Input to the console task can consist of MRJE utility control statments and host system commands. The MRJE utility control statements that can be entered are READFILE, MODIFY, END, LIBRARY, CARRIAGE, and CANCEL. Records entered for the console task can be up to 120 characters in length.

When the reader task is active at the console, you can enter commands for the console input task in either of two ways:

- Key the console command into the reader input field and press command key 1. The command is then presented to the console input task.
- Press command key 1 before entering the console command. The console input field display replaces the reader input field display, and you can enter the console input command. Once the command has been processed, the reader input field display reappears, unless the command entered was the CANCEL command for the reader task at the console.

Output

Output from an MRJE session can consist of any one, all, or combination of the following:

- Print data from the host system (print task)
- Punch data from the host system (punch task)
- Operator messages from the host system (console output task)
- Operator messages from the System/34 (console output task)
- Operator messages generated by the MRJE utility (console output task)

Print Task

Print data can be directed to either the printer associated with the MRJE console or to a disk file. The printer associated with the MRJE console is the printer specified for the work station during System/34 configuration. You can specify a different printer by using the SET OCL statement. Output that is printed is not written to disk; and output that is written to disk cannot be printed by the MRJE utility (use the $DCSUP utility, as described in chapter 5, to print the data written to disk). Print data can be directed to the printer associated with the MRJE console by specifying either PR1-P during initialization or by starting/restarting the printer using the MODIFY statement. If spooling is active, print data is spooled to the disk spool file if PR1-P or PR1-R has been specified, unless SPOOL-NO is specified in the forms control table.

Printer forms operations are controlled by channel control characters and the CARRIAGE utility control statement. The first character of each print record received from the host is a channel control character that controls forms movement. The channel control character is not printed. The channel control character bits have the following meaning:

```
        o m c c c c c c

bit     0 1 2 3 4 5 6 7
```

o: Always set to 1.

m: Set to 0 for normal carriage movement.

cccccc: Specifies carriage control information. 1000nn causes the printer to immediately space nn lines; 11nnnn causes the form to skip to the line value specified for channel nnnn; 0000nn causes the printer to space nn lines after printing; 01nnnn causes the printer to skip to the line value specified for channel nnnn after printing; and 000000 suppresses the listing.

The channel specifications are defined during MRJE initialization. Channels 1 through 12 are used, and the line values for each are specified on the initialization screen.

Print data can be directed to a standard forms file named TDISKPR1 instead of to the printer. Records are written in TDISKPR1 in a compressed format. Use the $DCSUP print utility, described in Chapter 5, to print the TDISKPR1 file. $DCSUP expands the records before printing.

To direct data to TDISKPR1, do the following:

1. Specify the STDPR1 parameter during initialization. This parameter defines the number of blocks needed to contain all print output directed to the TDISKPR1 file during an MRJE session.

2. Assign the print task to the disk either during initialization (PR1-D) or by using the MODIFY utility control statement after initialization.

If the end of extent (which means TDISKPR1 is full) is reached while print data is being directed to TDISKPR1, a message is displayed that allows you to select one of the following options:

- Option 1: Save that portion of the print data already written in TDISKPR1 and de-activate the print task. The portion of the output that could not be written is retained at the host for later transmission.

- Option 2: Delete the portion of the current job print data already written to TDISKPR1 and de-activate the print task. Deleting the portion of the output already written makes it possible for the host to write additional data to TDISKPR1. The host saves the interrupted data for later transmission.

By specifying the proper value for the STDPR1 parameter during initialization, you can usually prevent overflow conditions. However, if the end of extent is reached and you do not want to lose any data, the following options are available:

- Select option 1 for the file is full message. Use the MODIFY utility control statement to reactivate the print task and to specify the printer associated with the MRJE console as the output device (PR1-P). The remainder of the print data is printed on the printer.

- Select option 1 for the file is full message. Enter the System/34 inquiry mode by pressing the Attn key and select either option 2 or option 3 to cancel MRJE. Use the RENAME procedure command to change the name of the TDISKPR1 file. The format of the RENAME procedure command is:

  RENAME TDISKPR1,newname

  where TDISKPR1 is the existing name, and *newname* is the name you want to assign to the file. TDISKPR1 is now available for input. You must initiate and initialize MRJE. As soon as initialization is complete and the communications link has been established, an abnormal termination occurs. You must initiate and initialize MRJE once more, ensuring that the PR1 parameter is specified for the disk (PR1-D). Communications now proceeds, with print data being written to the new TDISKPR1 file. The data written in the original TDISKPR1 file now exists in the file you specified using the RENAME procedure. This entire process may be repeated if another end of extent condition occurs.

For each printer stream written to TDISKPR1, messages are displayed indicating the relative record numbers of the first and last records in the file and the number of blocks still available in TDISKPR1. After the MRJE session has completed, you can use the $DCSUP utility to print the data written in TDISKPR1.

If the BSC communications line is disconnected during a printer stream update to TDISKPR1, a message (SYS-4565) allows you to select one of the following options:

- Option 1: Save that portion of the print data already written in TDISKPR1. The portion of the output that could not be written is retained at the host for later transmission.

- Option 2: Delete the portion of the current job print data already written to TDISKPR1. The host saves the complete stream for later transmission.

Print data that is to be written to an individual file or that is to be printed later on a special form can be directed to individual special forms files on disk. Special forms files can be processed by user written RPG II or basic assembler programs, or printed by the $DCSUP utility. These files always contain records with 132 data characters preceded by one channel control character, for a total of 133 characters per record.

Print data is directed to an individual special forms file as a result of a forms mount message transmitted from the host system to the MRJE utility. The first character of the form name or number is compared to the character specified for the DID parameter on the initialization display or in the MODIFY utility control statement. If the characters are the same, an individual special forms file is created on disk for the print data. If the characters are not the same, the MRJE utility issues the forms mount message to the system console operator if spooling is not active.

The host system transmits a forms mount message to the MRJE utility each time a forms change is required on the System/34. If the forms do not change from one file or session to the next, the host does not normally transmit a forms mount message. However, if the host system is restarted, it normally resets to the standard forms. There are host commands that can modify the format of the forms mount message. If these commands are used, ensure that the job ID and time stamp are not deleted from the message.

The special forms files created by MRJE are labeled in the disk volume table of contents (VTOC) as Axxxffff where A designates a printer file; xxx is the file sequence number (FSN); and ffff is the first four characters of the forms name or number.

The file sequence number is incremented by one each time an individual print or punch file is created. When a print file is created, the label is displayed in a message to the MRJE console operator.

You can create an individual file by:

1. Specifying the SPCPR1 parameter during initialization. You specify the number of blocks that are needed for the largest print file that will be created during the MRJE session. Any excess space after a file is created is made available for other files.

2. Assigning the print task to either the printer (PR1-P) or the disk (PR1-D) during initialization or with a MODIFY utility control statement.

3. Specifying the disk identifier (DID) parameter either during initialization or with a MODIFY statement.

4. Specifying the FSN parameter during initialization.

If the end of extent (which indicates that the file is full) is reached before all data has been written to an individual file, you can save the data already written and continue processing by performing the following:

1. Select option 1 for the displayed message.

2. Use the MODIFY utility control statement to specify PR1-D.

The utility creates a new special forms file, numbered accordingly, and continues writing data to the new file. Note that the print data for one job is now in two separate files.

If the BSC communications line is disconnected during the creation of a special forms print disk file, a message (SYS-4565) allows you to select one of the following options:

- Option 1: Save that portion of the special forms file already written to the disk. The portion of the output that could not be written is retained at the host for later transmission.

- Option 2: Delete the portion of the special forms file already written on the disk. The host saves the complete file for later transmission.

Punch Task

The punch task processes all punch data from the host system and writes 80-character records in an individual special forms file. Because the System/34 does not support a card punch, actual punched card output cannot be produced by the MRJE utility.

The files created for punch output are labeled in the VTOC by MRJE as Bxxxffff, where B designates a punch file; xxx is the file sequence number (FSN); and ffff is the first four characters of the form name or number.

The file sequence number is incremented by one each time an individual print or punch file is created. When a punch file is created, the label is displayed in a message to the MRJE console operator.

You can create an individual file by:

1. Specifying the SPCPU1 parameter during initialization. You specify the number of blocks that are needed for the largest punch file that will be created during the MRJE session. Any excess space after a file is created is made available for other files.

2. Assigning the punch task to the disk (PU1-D) during initialization.

3. Specifying the FSN parameter during initialization.

If the end of extent is reached while punch data is being directed to the disk, a message is displayed that allows you to select one of the following options:

- Option 1: Save that portion of the punch data already written in the disk and de-activate the punch task. The portion of the output that could not be written is retained at the host for later transmission.

- Option 2: Delete the portion of the current job punch data already written and de-activate the punch task. The host saves the interrupted data for later transmission.

If the end of extent (which indicates that the file is full) is reached before all data has been written to an individual file, you can save the data already written and continue processing by performing the following:

1. Select option 1 for the displayed message.

2. Use the MODIFY utility control statement to specify PU1-D.

The utility creates a new special forms file, numbered accordingly, and continues writing data to the new file. Note that the punch data for one job is now in two separate files.

If the BSC communications line is disconnected during the creation of a special forms punch disk file, a message (SYS-4565) allows you to select one of the following options:

- Option 1: Save that portion of the special forms file already written to the disk. The portion of the output that could not be written is retained at the host for later transmission.

- Option 2: Delete the portion of the special forms file already written on the disk. The host saves the complete file for later transmission.

Console Output Task

Console output consists of operator messages from the host and the System/34 MRJE utility. Messages from the host are presented only at the MRJE console; messages from the MRJE utility can be presented at either the MRJE console or at a display station attached to the MRJE reader task. All messages from the host or the MRJE utility are written to the system history file. Because messages from the host can contain up to 120 characters, a message may be displayed on two lines. The message is also written to the history file. To display the MRJE entries in the history file, use the HISTORY procedure command (as described in the *System Support Reference Manual*) with the SYSTEM parameter specified.

Termination

Normal termination of the MRJE session occurs when the MRJE console operator enters the END utility control statement. When the END statement is entered, the MRJE utility sends a sign-off message to the host system to terminate the communications link. Normal termination can occur only after all print and punch data for jobs processed has been received by the MRJE utility. Therefore, you should ensure that all output data from the host has been received before entering the END statement. The following suggestions should help you terminate the MRJE session:

- Enter the END statement only after all output has been received.
- Use the DELAY parameter in the END statement to postpone the sending of the sign-off message to the host.
- Enter an additional END statement, with the DELAY parameter specified to extend the current delay period.

During the termination phase of MRJE, you should note the following conditions:

- Once the END statement has been entered, no additional reader tasks will be started. Any reader input entered before the END statement is processed.
- The sign-off command is not sent to the host until all reader tasks have terminated.
- The sign-off command is not sent until the time period specified by the DELAY parameter in the END statement has elapsed.
- The sign-off command is not sent if any task (reader, print, or punch) is active. Note that inactivity of the print and punch tasks does not indicate that the host has no more data to send.
- The timer is set after all reader tasks have terminated and no print or punch tasks are active.
- The timer is reset to zero if either the print or punch task becomes active. Actual timing does not begin again until all tasks are inactive.
- The timer is reset to zero if a second END command is entered. The DELAY value in the new END statement replaces the DELAY value specified in the previous END statement.

You can terminate MRJE by entering the SSP CANCEL command (this command is not the same as the MRJE CANCEL command), or by selecting either option 2 or option 3 from the System/34 inquiry display. These techniques of terminating MRJE cause an abnormal termination and no sign-off command is sent to the host system.

As the final phase of termination, MRJE ends communications with the host, deallocates all open disk files, and issues a series of messages regarding the MRJE session. These messages are shown in the following table. The MRJE utility then goes to end of job. Press the Enter/Rec Adv key to return the display to the System/34 command display.

MRJE Session Messages

```
MRJE INPUT RECORDS READ       nnnnn
MRJE OUTPUT RECORDS PRINTED   nnnnn
MRJE OUTPUT RECORDS PUNCHED   nnnnn
MRJE TOTAL TRANSMISSIONS      nnnnn
MRJE TIMEOUTS                 nnnnn
MRJE DATA CHECKS              nnnnn
MRJE NEGATIVE RESPONSES       nnnnn
```

The MRJE BSC I/O counters are written to disk. A time-out indicates a lack of response to a transmission. A data check indicates some form of bad data received. A negative response indicates some form of bad data transmitted.

MRJE UNATTENDED MODE OF OPERATION

Unattended mode allows the MRJE utility to operate with a minimum of operator intervention. Operator action is required only if a System/34 message is displayed. For MRJE messages (those containing MRJE in the text) and host messages (for example, forms mount messages), the MRJE utility selects the default option and continues processing. Refer to the *Displayed Messages Guide* for information on the MRJE messages and defaults.

Before MRJE can be run in unattended mode, you must have an initialization format that contains the necessary information for the MRJE session. In unattended mode, no changes can be made to the default values given on the screen.

The following points must be considered before initiating MRJE in the unattended mode:

- Initialization in unattended mode occurs without operator intervention. Therefore, the default values on the initialization screen specified must be the correct values for the MRJE session.
- Only one reader task (RDN-1) is allowed in unattended mode, and it must be assigned to the disk, using the RD1NAME parameter.
- The initialization screen must specify the RD1NAME, RD1DATE, RD1CMD, RD1LIBR, and RD1DEL appropriately so that a file or library member will be read immediately after initialization.

Initiation for Unattended Mode

To initiate MRJE for unattended mode of operation, specify AUTO-Y in the MRJE procedure command and specify the proper initialization format name for the DISPLAY parameter.

Initialization in Unattended Mode

Once the MRJE procedure command has been entered with AUTO-Y specified, the initialization phase proceeds without operator intervention. Communications are established in the same manner as for attended mode. Note that the operator may be required to dial the host system to establish the connection or to answer the host call to establish the connection.

If autocall is used in conjunction with unattended mode, the System/34 places the call. If the connection cannot be made, the return code is set and MRJE terminates. You can interrogate the return code by using the IF statement, as described in the *System Support Reference Manual*.

Input in Unattended Mode

In unattended mode, reader input is always from the disk and is generally initiated by the RD1NAME parameter from the initialization screen. The processing of files is the same as for the attended mode. Proper use of the command and data files allows for multiple files to be processed in one MRJE session. For example, the file specified for the RD1NAME parameter would normally be a command file, and it would contain one or more READFILE statements that caused one or more data files to be read. It could also reference another command file in a READFILE statement, but control would never be returned to the RD1NAME file.

The END utility control statement is also permitted in a command file during unattended mode and its processing causes MRJE to enter the termination phase.

Console input, as described for attended mode, is permitted during unattended mode of MRJE operation. However, this overrides the purpose of unattended mode since it requires an operator's presence.

Output in Unattended Mode

Output is handled in much the same way during unattended mode as during attended mode. The only difference is that during unattended mode any messages from the host or generated by the MRJE utility are handled automatically. When in unattended mode, MRJE selects the default option for all messages except System/34 system messages. The System/34 operator may have to respond to messages generated by the System/34. For a description of the MRJE messages and the default options, refer to the *Displayed Messages Guide.*

Termination of MRJE in Unattended Mode

Termination of MRJE while in unattended mode occurs whenever the END utility control statement is encountered in a command file or whenever the END statement is entered from the console. Once the END statement has been entered, termination proceeds in the same manner as described for attended mode of operation. It is recommended that you specify a DELAY parameter in the END statement to ensure that all output from the host is received.

ALTERING THE INITIALIZATION SCREEN FORMAT

If you are going to be running MRJE with the same values each time, and those values differ from the values on the IBM-supplied initialization screen, you can modify the display screen format so that it has your needed default options. You can also modify the format more than once and generate a display screen for each configuration you will be running.

You can have up to 24 different formats at any time. Each screen format is identified by a unique name of eight characters or less. The name of the screen you want to use when you initiate MRJE is specified as the DISPLAY parameter in the MRJE procedure command. In choosing a name for a screen format, you can select any string of eight or fewer characters, except that the names #MR02 through #MR09 cannot be used because they are reserved for other MRJE functions. The IBM-supplied initialization format is named #MR01. You can assign the name #MR01 to one of your formats. The IBM-supplied initialization format and its defaults are then lost until they are restored at a configuration, such as a release update. Whenever you initiate MRJE and do not specify a format name for the DISPLAY parameter, #MR01 will be used as the initialization format.

The following steps can be followed to modify the values on the IBM-supplied initialization format and generate a new initialization format with your desired default values:

1. Enter: SEU #MR@01,S,,80

2. The format name begins in column 7 of line 4. You can retain the existing name or key in a new name beginning in column 7. Enter the number of the line you want to change and press the Enter/Rec Adv key.

3. The values for all other parameters start in column 57 of a given line. If a change is necessary, key the new value and press the Enter/Rec Adv key to record the changed record. If no changes are necessary, press the Enter/Rec Adv key to advance to the next line. Certain parameters (such as SIGNON and MESSAGE) may require more than one line. For these parameters, begin keying in column 57 of the first line and continue keying through column 79, enter an X in column 80 to indicate continuation, and press the Enter/Rec Adv key. You may continue keying on the next line beginning in column 7.

   Note: Be certain that you change only the parameter values; do not alter any other characters in the format.

4. When you have made all necessary changes, press command Key 7 to exit from SEU. Select one of the options from the menu that is displayed to end the job.

5. To convert your source member to an object member:
   a. If no format currently exists with the same name as the one you are installing, *enter* FORMAT ADD,#MR@M1,,#MR@01.
   b. If a format exists with the same name as the one you are installing, *enter* FORMAT UPDATE,#MR@M1,,#MR@01. The format you are installing replaces the existing format of the same name on the file. A warning message appears, and can be ignored if you select the 0 option.

To run MRJE using your default values, specify your format name for the DISPLAY parameter in the MRJE procedure. The initialization screen that appears is the one you generated and it contains your default values. You can modify any value on the screen in the same manner as you modified those values on the IBM-supplied initialization screen format. Press the Enter/Rec Adv key after all changes have been made to the display or to accept all default values.

MRJE UTILITY CONTROL STATEMENTS

The MRJE utility control statements provide you with the means to control an MRJE session after the initialization phase has completed. The utility control statements can be used to affect the input, output, and termination phases of an MRJE session. The utility control statements and the phases they affect are shown in the following table:

Phase	Statement
Input	READFILE LIBRARY EOF MODIFY
Output	CANCEL MODIFY CARRIAGE
Termination	END

Figure 4-7 (Part 1 of 2). MRJE Utility Control Statements

Each utility control statement can also be classified as to whether it serves as reader input or console input and as to the devices from which it may be entered.

Task	Entered From	Statement
Reader input	Disk	READFILE EOF LIBRARY END (unattended mode only)
	Display station	EOF READFILE LIBRARY
Console input	MRJE console	CANCEL MODIFY READFILE END CARRIAGE LIBRARY

Figure 4-7 (Part 2 of 2). MRJE Utility Control Statements

Writing MRJE Utility Control Statements

Each utility control statement consists of a statement identifier and parameters. The statement identifier is always the first word of the statement. The parameters supply the necessary information to the utility. Each parameter consists of a parameter keyword, which identifies the parameter, and the actual value that is to be supplied to the utility.

The syntax of the MRJE utility control statement is:

.. statement-identifier parameter-value,...,parameter-value comment
(.. s)

Rules for Coding MRJE Utility Control Statements

The rules for coding utility control statements are:

Statement Identifier: .. (two periods) followed by one blank must precede the statement identifier. Do not use blanks within the identifier. Certain statements can use a single letter in place of the full statement identifier. This abbreviation is indicated by the letter in parentheses.

Blanks: One or more blanks are required between the identifier and the first parameter.

Statement parameters: Parameters can be in any order. A comma is required to separate one parameter from another (do not use blanks between parameters), and a hyphen (-) is required within each parameter to separate the keyword from the value, except for the CANCEL statement.

Comments: Comments can be included in utility control statements when parameters are entered. Leave one or more blanks between the last parameter in the utility control statement and the comment.

MODIFY

The MODIFY statement changes the assignment of the reader, punch, and print tasks. The MODIFY statement can be entered from the MRJE console only under the following conditions:

- The MODIFY statement can be entered from the console only after initialization is complete and before an END statement is entered.
- A MODIFY statement that changes the assignment of the MRJE console reader can be entered only when the MRJE console reader task is not active.

If a MODIFY statement that changes the print or punch assignment is entered while output is being processed by the specified task, the assignment does not change until the current file output is terminated.

Although each parameter of the MODIFY statement is optional, at least one parameter must be specified.

The MRJE console operator should be aware of any display station using the MRJE utility when the MODIFY statement is entered. You can use the system MSG command to send a message from the MRJE console operator to display station operators informing them of changes.

MODIFY utility control statement format:

(.. M)

PR1: Assigns the print task. PR1-P assigns the print task to the printer. PR1-D assigns the print task to the disk. PR1-H halts the print task without stopping the entire system. This parameter is valid only when the printer is assigned for output. PR1-R restarts the print task if it was halted by the PR1-H parameter.

RD1-K: Assigns the console reader task to the keyboard. The keyboard is always treated as a command file.

PU1: Assigns the punch task. PU1-D assigns the punch task to the disk. PU1-D is valid only if PU1-D was also specified on the initialization screen. PU1-N de-activates the punch task.

DID: Specifies the identification character associated with special forms files. The value is compared with the first character of the forms number or name in each forms mount message from the host system to determine the disposition of the output. If the two characters are equal, the file is written to an individual special forms disk file. If the characters are not equal, the file is written to the device assigned to the print task.

READFILE

The READFILE statement directs the reader task to retrieve data from a disk file, source member, or procedure member. A READFILE statement may be either console or reader input, and it may be entered from a display station, the MRJE console, or a command file.

READFILE utility control statement format:

```
.. READFILE NAME–name [,CMD–{Y}] [,TYPE–{P}] [,DATE–date]
(.. R)               {N}       {S}       {D}

[,LIBR–{name}] [,DEL–{Y}]
       {0   }        {N}
```

NAME: Specifies the name of the disk file, or the name of the source or procedure member to be read by the reader task.

CMD: Specifies whether the file is a command file or a data file. CMD-N indicates the file is a data file. CMD-Y specifies the file is a command file. The default is CMD-N.

TYPE: Specifies the type of the file specified in the NAME parameter. TYPE-D indicates disk file. TYPE-S indicates source member. TYPE-P indicates procedure member. TYPE-D is the default.

DATE: Specifies the creation date of the disk file specified by the NAME parameter. This parameter is optional if there are no duplicate file names. The date must be entered in the system date format.

LIBR: Specifies the library to be searched for the source or procedure members specified in the READFILE statement. The LIBR parameter overrides any LIBRARY statement previously entered. After reading a library member, the system will use the library specified on the LIBRARY statement as the active user library. If you specify 0 for the LIBR parameter, the system will use #LIBRARY. If you do not specify the LIBR parameter, the default is the library specified on the previous LIBRARY statement.

DEL: Specifies whether deleted records should be transmitted to the host system. DEL-Y indicates that deleted records should be transmitted to the host system. DEL-N indicates that deleted records should not be transmitted to the host system. If deleted records are to be transmitted, both the host system and the System/34 must be in transparent mode. DEL-N is the default.

EOF

The EOF (end-of-file) statement signals the end of the current reader input to the host system. EOF de-activates the reader task at a display station. At the MRJE console, EOF de-activates the reader task and returns the MRJE console to the console input task.

EOF utility control statement format:

```
.. EOF
(.. E)
```

No parameters are required for this statement.

Note: When the host system receives an end-of-file indication, it queues the last job entered for execution.

END

This statement signals the end of the MRJE utility session when entered from the console input task or from a reader command file in unattended mode. When entered from a reader not in unattended mode, END is logged to the history file and treated as a comment. The termination process can be delayed a specified amount of time by using the delay parameter.

END utility control statement format:

```
.. END DELAY-nn
```

DELAY: Specifies the amount of time, in minutes, that the MRJE utility waits before sending the sign-off command to the host (after all reader, printer, and punch tasks have received an end-of-file). The delay parameter can be any value from 0 to 99. If this parameter is not specified, 0 is assumed. DELAY-0 means that the MRJE utility sends the appropriate sign-off command to the host system as soon as all current reader, printer, or punch tasks have received an end-of-file. The timer is reset whenever print or punch data is received from the host.

CARRIAGE

The CARRIAGE statement specifies the printer forms length and correlates channel control characters with forms line numbers. This statement is optional, and can be entered only from the console. If the CARRIAGE statement is entered, it takes effect at the beginning of the next file received.

CARRIAGE utility control statement format:

.. CARRIAGE [L–nnn] [,cc–mmm] ... [,cc–mmm],

L: Specifies forms length (112 lines maximum). The current MRJE utility lines per page setting is assumed if this parameter or the entire statement is omitted.

cc: Equates a channel control character with a forms line number. The cc is a channel control character between 1 and 12.

The mmm is a forms line number between 1 and the maximum number of lines on the form. A channel control character cc (1 through 12) causes a skip to line mmm (1 through 112). A cc–mmm combination can be specified for all 12 channel control characters.

If no channel control character values are specified, all channel control character values remain unchanged.

If one or more channel control character values are specified, all unspecified channel control characters are set to zero. (A specification of zero causes a single space if the specified channel control character is encountered.)

Notes:

1. Initially, the channel control character values are obtained from the initialization screen.
2. Leading zeros are not required on any CARRIAGE parameter.
3. Check the host system (using host system commands) regarding the status of the output queues before entering a CARRIAGE statement. See the appropriate host manuals for the commands to use.

CANCEL

The CANCEL statement immediately cancels the current input or output (at the host) being processed by a task. This statement can be entered only from the console.

CANCEL utility control statement format:

```
.. CANCEL {PR1
           PU1
           work station id}
```

PR1: The current print output is removed from the queue at the host system. (This parameter is invalid if the host system is VM/RSCS.)

PU1: The current punch output is removed from the queue at the host system. (This parameter is invalid if the host system is VM/RSCS.)

work station ID: The reader task corresponding to the work station ID is de-activated. It might be necessary to check the host system (using host system commands) regarding the status of the input queue before reactivating the reader task.

Notes:

1. The reader task can be reinitialized with the MODIFY utility control statement at the MRJE console, or by entering the MRJE procedure command. When a reader task is deactivated, an EOF is sent to the host system, which processes any data entered before the EOF.
2. The MRJE console operator should be aware of any display station using the MRJE utility when the CANCEL statement is entered. The system MSG command can be used to send a message from the MRJE console operator to the display station operators informing them of changes.

LIBRARY

The LIBRARY statement specifies the library to be searched for source and procedure members. The specified library will remain in effect until the next LIBRARY statement is entered or until the task terminates. If you enter 0 or omit the library statement, the SSP uses the system library.

LIBRARY utility control statement format:

```
.. LIBRARY NAME- {name
                 0}
```

NAME: Specifies the name of the library to be used for READFILE statements. If you enter 0, the system library is used.

DIAGNOSTICS/DEBUGGING AIDS FOR MRJE

The System/34 SSP provides a trace facility that can be used to diagnose problems that might occur during MRJE operation. The trace facility is an IBM-supplied service procedure that logs a history of occurrences as they happen within the system. Enter the TRACE command from the system console to evoke the function. You are then prompted to select the type of activity you want traced. To trace MRJE operation, select the Y response when you are asked whether you want to trace communications-related events for the communications line that MRJE is using. The prompt occurs once for each communications line. You are also prompted for the disk logging option, which allows you to log the trace information to disk. Normally, the trace information is stored in a system table that can store 512 16-byte entries. After the system table has filled, it wraps around and begins logging the new information on top of the original information.

The MRJE trace facility remains active until reset by specifying N to the TRACE communications prompt. The system table stores MRJE BSC interrupts during the tracing session. Intermediate text block (ITB) interrupts, enables, disables, and 2-second time-outs are not recorded. Receive data is recorded in the table when an I/O interrupt occurs. Transmit data is recorded when the supervisor call (SVC) is called. In general, an entry is made each time a record is sent or received by the MRJE utility. Refer to the *Data Areas and Diagnostic Aids Manual* for additional information on the TRACE procedure and for the format of the 16-byte trace table entries.

HOST CONSIDERATIONS

When the System/34 with the MRJE utility is to be used as a terminal to a remote system, the host system must be aware of certain considerations.

- The tasks used in the MRJE utility must be defined in the host system remote definition to include:
  - 1 console
  - 1 to 3 reader tasks
  - 1 print task
  - 0 to 1 punch task

- All host systems must define a System/34 as a System/3 work station with console support.

Further considerations are given in the following paragraphs.

For the RES host system, the TERMINAL macro defines the terminal characteristics. (See *OS/VS1 RES System Programmer's Guide*, GC28-6878.) The format is as follows:

```
TERMINAL TERMID=n,TDESCR=(w,t,d,f) [,LNUM=n] [,RDRS=n] [,PTRS=n]
         [,PCHS=n] [,COMPRES=YES/NO] [,BUFXSIZ=512]
         [,CNMSGNO=wtonum]
```

TERMID = n: Specifies the number of the remote terminal. Corresponds to the termid parameter in the MRJE LOGON command.

TDESCR = (w,t,d,f): Describes the terminal. The parameters are as follows:

w – printer width. Options are 0, 1, 2, 3, 4, 5, 6, and 7. Select 3, 132-character width, for the System/34.

t – terminal type. Options are 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. Select 6 (System/3) for the System/34.

d – data transmission format. Options are 0, 3, 4, and 5. Select option 4, programmable interface, for the System/34.

f – work station feature. Option 2 is for console without transparency; option 3 is for console with transparency.

LNUM = n: Specifies the communications line that is to be used.

RDRS = n: Specifies the number of reader tasks that can be active. Options are 0, 1, 2, and 3. Corresponds to the RDN parameter on the initialization display.

PTRS = n: Specifies the number of print tasks. Select 1.

PCHS = n: Specifies the number of punch tasks. Specify 1.

COMPRES: Specifies whether space compression is to be used. The default selection is NO.

BUFXSIZ: Specifies maximum size of expanded buffers. Select 512.

CNMSGNO: Specifies the number of buffers to be used for the console messages to the terminal. The number specified can be from 1 to 255, with the default being 5.

For JES2, the command: $TRnnn.PRI,F=AUTOM is required if a forms mount message should be issued.

The host command $TRnnn. PRI, F = AUTOM is required to automatically send print data. This option issues a forms mount message. At the end of an MRJE session, the command $TRnnn. PRI, F= STD will set the print to standard.

On the ASP system, the BSCTERM card defines the terminal characteristics. The format is as follows:

BSCTERM,NAME=RMnnn,TYPE=S3,BLKSIZE=nnnn,LINESIZE=132
,XPARENT=YES/NO,CONSOLE=NONE/INQUIRY/TERMINAL/ALL

NAME: Specifies the identifier of the remote terminal. RMnnn represents the terminal number (1 to 127). Corresponds to the RMnnn parameter in the MRJE SIGNON statement.

TYPE: Specifies the type of terminal being defined. Specify TYPE=S3.

BLKSIZE: Specifies the maximum size of communications blocks. The default is 400.

LINESIZE: Specifies the printer line width for the remote terminal printer.

XPARENT: Specifies whether transparent mode is to be used for communications. The default is NO. Corresponds to the XPC parameter on the initialization display.

CONSOLE: Specifies how the host system will handle messages from the terminal console. NONE, the default, specifies that the system will not accept console messages from the terminal. INQUIRY specifies that only MESSAGE(Z) and INQUIRY(I) messages having no effect on system status will be accepted. TERMINAL specifies that only messages affecting the terminal jobs will be accepted. ALL specifies that all messages except SWITCH, DUMP, and RETURN will be accepted.

On the HASP II system, &BSCCPU=YES and &MLBFSIZE=400 should be specified. In the RMTnn macro, the following are specified:

RMTnn=mmooppiillwtdf

- nn – Specifies the remote terminal number.
- mm – Specifies the communications line number for HASP II.
- oo – Specifies the print data routing.
- pp – Specifies the punch data routing.
- ii – Specifies the priority increment.
- ll – Specifies the priority limit.
- w – Specifies the printer width. Select 3 (132 characters).
- t – Specifies the terminal type. Select 6 (System/3).
- d – Specifies the data format. Select 4 (MRJE).
- f – Specifies the terminal features. Select 2 for a console without transparency, or select 3 for a console with transparency.

MRJE SESSION EXAMPLE

The host system JCL is read from a file on the System/34 disk. Then the host transmits printer output to the System/34. Printer output directed to standard forms is printed, and printer output directed to special forms is written in the disk file. Repeat this process with the print task assigned to the disk. Each step of the session is identified and explained. This example is for a host system using RES; however, this example is similiar to other host systems.

Load the MRJE utility and initialize it by entering MRJE from the keyboard. Change the display screen initialization parameters as follows:

```
MRJE OPTION MENU--PRESS ENTER TO CONTINUE


ENTER INITIAL CONFIGURATION INFORMATION BELOW:
HOST - H          LEN - 0400          XPC - N        COM - C
PR1 - P           PU1 - N             FSN - 001      DID - 9
SPCPR1 - 0060     SPCPU1 - 0060       STDPR1 - 0060  RDN - 1
RD1NAME -         RD1DATE - 000000    RD1TYPE - D    RD1CMD - Y
RD1LIBR - 0       RD1DEL - N          RD1 -

ENTER LOGON OR SIGNON INFORMATION BELOW:
/*SIGNON

ENTER INITIAL CARRIAGE AND FORMS INFORMATION BELOW:
L - 066                F - STD.     FCTNAME -
1 - 001     2 - 000    3 - 000      4 - 000     5 - 000     6 - 000
7 - 000     8 - 000    9 - 000     10 - 000    11 - 000    12 - 000


**********          SYSTEM OPERATOR MESSAGE BELOW          **********
```

HOST-R
LEN-0512
DID-M
LOGON USER1/PASS1,TERM(1) PROC(MRJEPROC)

```
MRJE OPTION MENU--PRESS ENTER TO CONTINUE


ENTER INITIAL CONFIGURATION INFORMATION BELOW:
HOST - R          LEN - 0512          XPC - N         COM - C
PR1 - P           PU1 - N             FSN - 001       DID - M
SPCPR1 - 0060     SPCPU1 - 0060       STDPR1 - 0060   RDN - 1
RD1NAME -         RD1DATE - 000000    RD1TYPE - D     RD1CMD - Y
RD1LIBR - 0       RD1DEL - N          RD1 -

ENTER LOGON OR SIGNON INFORMATION BELOW:
LOGON USER1/PASS1,TERM(1),PROC(MRJEPROC)

ENTER INITIAL CARRIAGE AND FORMS INFORMATION BELOW:
L - 066                F - STD.     FCTNAME -
1 - 001     2 - 000    3 - 000      4 - 000     5 - 000     6 - 000
7 - 000     8 - 000    9 - 000     10 - 000    11 - 000    12 - 000


**********          SYSTEM OPERATOR MESSAGE BELOW          **********
```

In a point-to-point switched network, dial the host system. Then specify which disk file to read by entering the following statement from the console input task:

.. READFILE NAME-PMRS003,TYPE-P,CMD-Y

The display screen initialization parameters initialize the MRJE utility. The READFILE utility control statement specifies which file to read.

The procedure PMRS003 contains the following data (JCL) that is transmitted to the host system:

```
//JOBNAME JOB (accounting parameters),programmer name

//*JOB TO PRODUCE PRINTED OUTPUT TO STANDARD FORMS

//STEP1 EXEC PGM=IEBPTPCH

//SYSUT1 DD DSN=SYS1.MACLIB,DISP=SHR

//SYSUT2 DD SYSOUT=A

//SYSPRINT DD SYSOUT=A

//SYSIN DD *
  PRINT TYPORG=PO,MAXNAME=10
  MEMBER NAME=SAVE

//

.. EOF
```

The host system reads the JCL and transmits the printer output to the System/34.

Now enter:

```
.. READFILE NAME-PMRS004,TYPE-P
```

The procedure member PMRS004 contains:

```
//JOBNAME JOB (accounting parameters),programmer name

//STEP1 EXEC PGM=IEBPTPCH

//SYSUT1 DD DSN=SYS1.MACLIB,DISP=SHR

//SYSUT2 DD SYSOUT=(A,,MRJE)

//SYSPRINT DD SYSOUT=A

//SYSIN DD *
  PRINT TYPORG=PO,MAXNAME=10
  MEMBER NAME=SAVE

//
```

For this job, an individual special forms file is created on disk because the first number in the forms number is M, which is the same as the DID parameter entered during initialization. The file label assigned by the MRJE utility is logged by the System/34.

Enter the following statements from the MRJE console to modify the print task to the disk, read the same file of JCL, and transmit it to the host:

```
.. MODIFY PR1-D

.. READFILE NAME-PMRS003,TYPE-P,CMD-Y

.. READFILE NAME-PMRS004,TYPE-P
```

The System/34 reads and transmits the same JCL from the disk to the host system. In this example, the printer output is directed to the TDISKPR1 file and a special forms file is created.

Terminate the session by entering an END statement:

```
.. END
```

If this entire example is run, the standard forms printer output is printed once and written to disk once. The printer output is directed to individual special forms files twice. The labels in the VTOC are TDISKPR1 for the standard forms and A001MRJE and A002MRJE for the individual files (if FSN was initially 001).

Chapter 5. Data Communications Print Utility Program

The System/34 System Support Program Product includes a print utility ($DCSUP). The print utility prints punch output and printer output that was directed to the disk during an MRJE or SRJE session. The print utility can process any number of disk files and any number of contiguous records within each file.

The print utility can also create a disk file of 80- or 136-character records. This option is available only when you process 256-character records that were written to disk during an SRJE session.

Input to the print utility consists of:

1. Disk files from an MRJE or SRJE session.
2. OCL to load the print utility.
3. Print utility control statements.

Output from the print utility can consist of:

- Printer output in either EBCDIC character format or in both EBCDIC and hexadecimal character format. When printer output is printed in both EBCDIC and hexadecimal character format, carriage control information is ignored.
- A disk file of either 80- or 136-character records, created after the processing of a 256-character record file that was written to disk during an SRJE session.

DISK FILES

Punch files contain 80-character records. You can print these files either in EBCDIC characters or in both EBCDIC and hexadecimal characters.

MRJE print files contain either compressed or 133-character records. The first character of each record is a channel control character that controls forms movement.

SRJE print files can contain either 136-character records or 256-character records. The first four characters of the 136-character records contain channel control information. The 256-character records reside on disk in the form received from the host system.

The channel control character(s) of a print file are not printed. Only the 132-character portion of the record is printed, even if a hexadecimal format is requested.

The TDISKPR1 file written by the MRJE utility contains compressed records that can be printed either in EBCDIC characters or in both EBCDIC and hexadecimal characters.

DCPRINT PROCEDURE COMMAND

The print utility can be loaded and executed using the following System/34 procedure command:

DCPRINT [filename1] [,filename2]

filename1: Specifies the name of the file to be printed.

filename2: Specifies the name of the forms control table file created by either the DCFORMS procedure or the $DCFUP utility.

The DCPRINT procedure command generates the following OCL statements:

```
// LIBRARY NAME-0
// MEMBER USER1-##MSG2
// LOAD $DCSUP
// RUN
// COPYFILE NAME-filename1, FCTNAME-filename2
// GO
// END
```

If filename1 is not specified, $DCSUP prompts the operator for an entry. If the DCPRINT command is used from the input job queue, the file name must be included, or the system operator receives an error message and must cancel the job. If filename2 is not specified, the last carriage control information supplied is used. If no carriage control information has been supplied, L-66, B-66, and 1-1 are used.

The PRINTER OCL statement, as described in the *System Support Reference Manual,* can be inserted in the OCL used by the DCPRINT procedure command. The name specified in the OCL statement for the $DCSUP utility must be HARDCOPY. The PRINTER statement allows you to specify additional parameters for the printing of data, including the number of copies desired.

OCL TO LOAD AND RUN THE PRINT UTILITY

The following OCL statements are required to load and run the print utility:

```
// LOAD $DCSUP

// RUN
```

Utility control statements are entered after the RUN statement to control the execution of the print utility.

$DCSUP UTILITY CONTROL STATEMENTS

The COPYFILE, SELECT, CARRIAGE, GO, and END utility control statements control the execution of the print utility. These statements provide the output format, page format, and multiple copies of disk files. Following is a description of each of these statements.

COPYFILE

The COPYFILE statement specifies the file to be printed and the format of the printed output.

COPYFILE utility control statement format:

```
// COPYFILE NAME-filename1 [,DATE–date] [,OUTPUT–{PRINT}]
                                                {PRINTX}
                                                {NOPRINT}

[,COPYOUT–name] [,RECL–{80}] [,RECORDS–number]
                       {136}

[,FCTNAME-filename2 ]
```

NAME: Specifies the name of the disk file to be printed.

DATE: Specifies the creation date of the file to be printed. The date must be entered in the system date format. If the DATE parameter is omitted, the file with the specified name and the latest creation date is processed.

OUTPUT: Specifies the format of the printed output. PRINT specifies EBCDIC character output. PRINTX specifies both EBCDIC and hexadecimal character output. (The channel control character(s) are not printed.) NOPRINT specifies no printing of a file created by SRJE. (Use NOPRINT only with the COPYOUT parameters.)

The following parameters (COPYOUT, RECL, and RECORDS) are valid only when processing 256-character records from an SRJE session.

COPYOUT: Specifies the name of the disk file to be created.

RECL: Specifies the record length of the disk file to be created. Only 80 and 136 are valid record lengths.

RECORDS: Specifies the number of records to be allocated for the disk file being created. The number of records can be from 1 to 999999. Leading zeros are not required.

FCTNAME: Specifies the name of the forms control table created either by the DCFORMS procedure or the $DCFUP utility program. If you do not specify FCTNAME, the default becomes the last carriage control information supplied. If no former carriage control information exists, L-66, B-66, and 1-1 are the defaults.

Note: The keyword parameter FCTNAME can be specified only for files created by SRJE.

SELECT

The SELECT statement specifies a portion of a file to be processed. SELECT is required only if you do not want to print the entire file.

SELECT utility control statement format:

```
// SELECT [FROM-nnnnnn] [,TO-nnnnnn]
```

FROM: Specifies the relative record number of the first record to be processed. FROM-1 is assumed if the parameter is omitted.

TO: Specifies the relative record number of the last record to be processed. TO-last (where last is the relative record number of the last record in the file) is assumed if the parameter is omitted.

Note: To process only one record, make the FROM and TO parameter numbers the same. If the specified record number does not exist, an error message is displayed.

CARRIAGE

The CARRIAGE statement specifies the number of print lines per form and also the line numbers to which the channel control characters equate.

CARRIAGE utility control statement format:

```
// CARRIAGE [L-nnn] [,B-nnn] [,cc-mmm] ...[,cc-mmm]
```

L: Specifies the number of print lines per form (12 through 112 maximum). If this parameter (or the entire statement) is omitted, 66 is assumed.

B: Specifies the line number of the last print line (bottom margin). The bottom margin must be between channel control character 1 and the number of lines per page. (Bottom margin is not used by MRJE).

cc: Equates a channel control character to a forms line number. The cc is a channel control character and can be any number from 1 through 12.

The *mmm* is a forms line number between 1 and the maximum number of lines on the form (for MRJE) or the bottom margin (for SRJE). One cc-mmm combination can be specified for each of the 12 channel control characters.

If one or more channel control character equates are specified, all unspecified channel control characters are equated to zero except channel 1, which defaults to 1. (A specification of zero causes a single space if the specified channel control character is encountered.)

Notes:

1. Initially, and for each carriage utility control statement, channel control character 1 is set to 1 and all other channel control characters not specified are set to 0.
2. Do not use leading zeros on any parameter of the CARRIAGE control statement.
3. If you use the CARRIAGE control statement, do not equate channel control characters to 0.

When a CARRIAGE statement is entered specifying any new channel control character equates, all previous channel control character equates are nullified.

GO

The GO statement causes the information specified in the last COPYFILE and SELECT statements entered to be printed using the current carriage control information. Each GO statement produces a copy of the file.

GO utility control statement format:

```
// GO
```

END

The END statement indicates the end of the utility control statements for a job.

END utility control statement format:

```
// END
```

$DCSUP EXAMPLES

Example 1

Print the entire file TDISKPR1 in EBCDIC characters:

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-TDISKPR1
// GO
// END
```

Note: This example can also be run by entering the DCPRINT TDISKPR1 procedure command.

Example 2

Print the file A001PAY in hexadecimal and EBCDIC characters beginning with record 100 through record 150:

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-A001PAY,OUTPUT-PRINTX
// SELECT FROM-100,TO-150
// GO
// END
```

Example 3

Print the file B0011200 in EBCDIC characters using 66 lines per page. Channel control character 2 is equated to line 10 and channel control character 12 is equated to line 60. A skip to channel 2 causes the printer to skip to line 10 for printing. A skip to channel 12 causes the printer to skip to line 60 for printing.

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-B00112000
// CARRIAGE L-66,2-10,12-60
// GO
// END
```

Example 4

Print records 10 through 50 and 75 through 120 from the file A001PAY in EBCDIC and hexadecimal characters, then print two copies of the entire file A002SAVE in EBCDIC characters:

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-A001PAY,OUTPUT-PRINTX
// SELECT FROM-10,TO-50
// GO
// SELECT FROM-75,TO-120
// GO
// COPYFILE NAME-A002SAVE,OUTPUT-PRINT
// GO
// GO
// END
```

Example 5

Create a disk file of 136-character records called FILE136, with space allocated for 1000 records, from a disk file (created by SRJE) of 256-character records called A0001. Do not print FILE136.

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-A0001,COPYOUT-FILE136,RECL-136,RECORDS-1000
,OUTPUT-NOPRINT
// GO
// END
```

Example 6

Print the file A001 by using the file control table FCT1 to supply carriage information.

```
// LOAD $DCSUP
// RUN
// COPYFILE NAME-A001,FCTNAME-FCT1
// GO
// END
```

Chapter 6. Forms Control Table Utility

The System/34 System Support Program Product includes a forms control table utility ($DCFUP). The forms control table utility uses an input source member to build a disk file containing forms control information for use by either the MRJE utility or the SRJE utility.

The forms control table allows you to associate carriage information and a System/34 forms name with a host forms name. The input source member consists of CARRIAGE utility control statements and an END statement.

RUNNING THE FORMS CONTROL TABLE UTILITY

You can run the forms control table utility by entering the DCFORMS procedure command. The format of the DCFORMS command is:

DCFORMS filename, [source] , [library / #LIBRARY]

filename: Specifies the name of the forms control file to be created.

source: Specifies the name of the source member containing the CARRIAGE utility control statements. If you do not specify *source*, a halt is issued that requests the source member name.

library: Specifies the name of the library where the source member resides. The default is #LIBRARY.

FORMS CONTROL TABLE UTILITY CONTROL STATEMENTS

The forms control table utility allows two utility control statements: CARRIAGE and END.

CARRIAGE

Each CARRIAGE statement results in one forms control table entry.

The format of the CARRIAGE statement is:

```
// CARRIAGE HOST-xxxxxxxx [,LOCAL-xxxx] [,HALT- YES/NO] [,L-nnn]
[,B-nnn] [,SPOOL-YES/NO] [,DEFER YES/NO] [,LPI- 4/6/8] [,CPI-10/15]
[,cc-mmm] ... [,cc-mmm]
```

HOST: Specifies the host forms name or number of this entry in the table. If the forms name or number is to be blanks, a comma (,) must be inserted immediately after the HOST- keyword, and one of the optional parameters must follow the comma.

LOCAL: Specifies the local forms number that the host forms number should be translated to. If omitted, no forms number translation is done.

HALT: Specifies whether to issue the normal System/34 forms mount message when this forms number is encountered. The default is NO.

L: Specifies the number of lines per page. The default is 66 lines per page.

B: Specifies the line number of the bottom margin (the last printed line). This number must be at least the value of channel control character 1 and at most the number of lines per page (L). The default is the number of lines per page.

SPOOL: Specifies whether print data is to be spooled. If spooling is not active when the forms control table is being used, this parameter is ignored.

DEFER: Specifies whether printing from the spool file is to be deferred. This parameter applies only when SPOOL-YES is specified. The default is YES for MRJE and NO for SRJE, unless overridden by the forms control table.

LPI: Specifies the number of lines per inch. The default is 6 lines per inch. This parameter is ignored if not supported by the hardware.

CPI: Specifies the number of characters per inch. The default is 10 characters per inch. This parameter is ignored if not supported by the hardware.

cc: Equates a channel control character with a forms line number. The cc is a channel control character between 1 and 12. The mmm is a forms line number between 1 and the maximum number of lines on the form. A channel control character, cc, causes a skip to the line mmm. You can specify a cc-mmm combination for all 12 channel control characters.

If no channel control character values are specified, all channel control characters remain unchanged. If you specify one or more channel control character values, all unspecified values are set to zero, except channel control character 1, which is set to 1.

The format of a forms control table entry is as follows:

- Bytes 1-8 are the host forms number.
- Bytes 9-12 are the local forms number.
- Byte 13 is a flag byte:
  - Hex 80 indicates to issue a halt when a forms mount is processed.
  - Hex 40 indicates carriage control information is included.
  - Hex 20 indicates a local forms number is specified.
  - Hex 10 indicates printed output is not to be spooled.
  - Hex 08 indicates printed output is to be deferred.
  - Hex 04 indicates printed output is not to be deferred.
- Bytes 14-27 are the channel control information, beginning with lines per page, bottom margin, and then channel control characters 1 through 12.
- Byte 28 is the second flag byte.
  - Hex 80 indicates 8 lines per inch is specified.
  - Hex 40 indicates 4 lines per inch is specified.
  - Hex 20 indicates 15 characters per inch is specified.
- Bytes 29-32 are reserved.

The statement can be continued onto additional lines. Refer to the *System Support Reference Manual* for the rules of continuation.

END

The END utility control statement indicates the end of the utility control statements. The format of the END statement is:

```
// END
```

The following messages can be printed by $DCFUP:

SYS-4655 INVALID UTILITY CONTROL STATEMENT

Additional Explanation: The statement preceding this message was not a valid CARRIAGE or END utility control statement.

SYS-4656 SAME KEYWORD SPECIFIED MORE THAN ONCE

Additional Explanation: The CARRIAGE statement preceding this message contains the same keyword specified in more than one parameter.

SYS-4657 INVALID PARAMETER SPECIFIED

Additional Explanation: The CARRIAGE statement preceding this message contains an invalid parameter. Check for an invalid keyword, an invalid parameter value, a missing or extra comma, or a continuation record expected.

SYS-4658 LINE NUMBER INVALID OR GREATER THAN 112

Additional Explanation: The CARRIAGE statement preceding this message contains an invalid line number. Valid line numbers for the L parameter are 1 through 112. Valid line numbers for the B parameters are from the value of the channel control character 1 to a maximum of the number of lines per page (the L parameter value). Valid channel equivalency parameter line numbers are from 0 to a maximum of the number of lines per page (the L parameter value).

SYS-4659 HOST PARAMETER MISSING

Additional Explanation: The CARRIAGE statement preceding this message is missing the HOST parameter. The HOST parameter is required.

SYS-4660 BOTTOM MARGIN GREATER THAN LINES/PAGE

Additional Explanation: The CARRIAGE statement preceding this message contains a B (bottom margin) parameter whose value is greater than the number of lines per page. The number of lines per page is either specified in the L parameter or, if the L parameter is not specified, the default is the current system lines per page.

SYS-4661 CHANNEL VALUE GREATER THAN LINES/PAGE

Additional Explanation: The CARRIAGE statement preceding this message contains a channel equivalency parameter whose forms line number value is greater than the number of lines per page. The number of lines per page is either specified in the L parameter or, if the L parameter is not specified, the default is the current system lines per page.

Chapter 7. SNA Remote Job Entry Utility–SRJE

INTRODUCTION

The System/34 System Support Program Product includes the SRJE (SNA Remote Job Entry) utility. The SRJE utility on the System/34 communicates with an IBM System/370 in an SNA (systems network architecture) environment using the SDLC line protocol. SRJE allows for the submission of jobs to a remote host system for processing and for the receipt of output from the host system. The output of jobs submitted to the host system can be returned to the SRJE utility, directed to another RJE work station, or directed to the host system output devices. The SRJE utility establishes the line connection, sends and receives data, and executes the termination procedures.

SRJE Environment

The SRJE utility operates in a synchronous data link control (SDLC) environment using systems network architecture (SNA). The SRJE utility supports communications with the following host systems using VTAM (virtual telecommunication access method) and NCP/VS (network control program/virtual storage):

- Remote Job Entry Service (RES) under OS/VS1.
- Job Entry Subsystem 2 (JES2) under OS/VS2.
- Job Entry Subsystem 3 (JES3) under OS/VS2.
- POWER/VSE under DOS/VSE.

You should have the appropriate host documentation on hand to properly operate with your host system. Some of this documentation is listed under *Related Publications*.

SRJE STORAGE REQUIREMENTS

The minimum requirement for the SRJE utility is a main storage capacity of 48 K bytes with a nucleus size of 18 K bytes. Systems with a larger main storage capacity can have a larger nucleus size (see the *Planning Guide* for more information on nucleus size). SRJE requires 14 K bytes of nonswappable[1] main storage, with 8 K bytes of that storage being used by SRJE, 4 K bytes being used by the SDLC task, and 2 K bytes being used for the minimum number of SDLC buffers (7). You specify the number of buffers during System/34 configuration. Refer to *SRJE Configuration* in this chapter for more information on SDLC buffers. SRJE also requires 16 K bytes of swappable main storage, with 10 K bytes of that storage being used by the SNA task and 6 K bytes being used by SRJE tasks.

[1] Multiple Logical Units (MLU) for SRJE will require 16 K bytes of nonswappable mainstorage, 10 K for use by SRJE.

SRJE CONFIGURATION

The SRJE utility is a part of the System/34 SSP and can be included with the SSP during System/34 configuration. To include SRJE, you must select it on the communications support display (display 9.0) during CNFIGSSP. You must also select the Secondary SNA/SDLC support on display 9.0. If autocall is to be used, you must also specify 1 for the Autocall Feature support prompt.

9.0 Communications Support

1. BSC Support	(0-No,1-Yes). .0
2. MRJE Support	(0-No,1-Yes). .0
3. SRJE Support	(0-No,1-Yes). .1
4. Secondary SNA/SDLC Support?	(0-No,1-Yes). .1
5. Remote Work Station Support?	(0-No,1-Yes). .0
6. SSP-ICF Support?	(0-No,1-Yes). .0
7. MLCA Support?	(0-No,1-Yes). .0
8. Autocall Feature Support?	(0-No,1-Yes). .0

Once you have completed display 9.0 and have selected 1 for both SRJE support and Secondary SNA/SDLC support, the SNA/SDLC parameters display (display 10.0) is presented.

10.0 SNA/SDLC Parameters

	Line 1	2	3	4
1. Station address? (Two hexadecimal digits)	____	____	____	____
2. Exchange ID? (Five hexadecimal digits)	____	____	____	____
3. Logical unit mode? (A-Single B-Multiple)	____	____	____	____
4. Receive data buffers?	____	____	____	____
5. Transmit data buffers?	____	____	____	____
6. Switch type?	____	____	____	____

(A-Auto answer B-Manual answer
C-Manual call D-Auto call)

Station address: The secondary SDLC station address is the means by which a physical unit (communication line) on the System/34 is known to the network control program (NCP) or primary station. This address is used by the System/34 SDLC hardware to determine whether data is destined for this station. You must specify a 2-character hexadecimal value for each line configured. The default is C1.

Exchange ID: A character sequence that identifies a particular secondary station at a link level. The exchange ID is required primarily for use on switched networks to provide station identification. You must specify a 5-character hexadecimal value for each line configured. The default is AAAAA. The value specified must be the same as that specified for the IDNUM parameter in the PU (physical unit) statement of the switched node definition at the host.

Logical unit mode: Single logical unit mode (A) indicates to the SNA task that: System/32 mode of operation is desired or that only one program will be using the SNA/SDLC capability, and that SIMLOGON messages may be received from the primary station. Multiple logical unit mode (B) indicates that there will be multiple users of the SNA task and that SIMLOGON messages from the primary station are not allowed. You must specify A (single) or B (multiple) for each line configured, with the default being A (single). B must be specified if SRJE is to operate with multiple SNA sessions.

Receive data buffers: The number of SDLC receive buffers to be allocated from nonswappable main storage for use by all active programs. This number should be the sum of the buffer requirements for all planned concurrent active sessions and should be calculated based on the host's receive pacing count (n) expected for each session. The number of receive data buffers should equal 2n + 1 for each session. If the value entered is less than 3, three buffers are allocated. The default is 7.

Transmit data buffers: The number of transmit data buffers to be allocated from nonswappable main storage for use by all active programs. If you enter 0, one buffer is allocated. The default is 7.

Note: Each receive data or transmit data buffer is 272 bytes in length.

Switch type: For switched line connections between the secondary physical unit and the primary station, it is necessary to specify the switch type for the System/34. The type can be A (auto answer), B (manual answer), C (manual call), or D (autocall). The switch type must be specified for each switched line.

DATA SECURITY AND SRJE

If any of the files you intend to use with SRJE are protected under the resource security capabilities of the System/34, you must ensure that the operator who signs on to SRJE is authorized to use those files. Security of files and libraries is based on the user ID and password. Whenever SRJE attempts to access a file or library (specified on the initialization display or in a READFILE statement) and the file or library is protected, the user ID is compared to the list of authorized users. If the user is authorized, the file or library can be accessed by SRJE. If the user is not authorized, an error message is displayed.

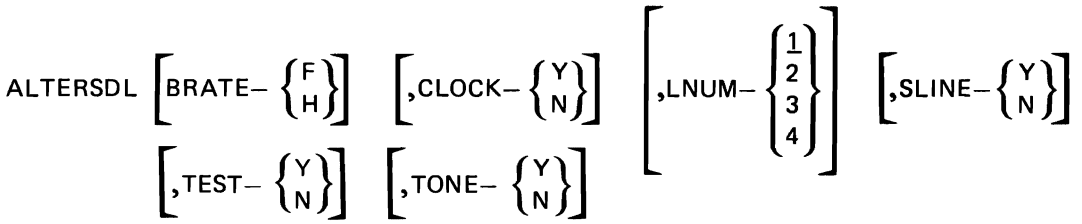
If file security is active, you should not release a reader terminal (when it is not the SRJE console) because file security cannot be performed without access to a terminal. If you do release the terminal, an error will occur even though the person who signed on to the reader terminal is authorized to use the file.

MODIFYING THE SRJE ENVIRONMENT

The ALTERSDL and SPECIFY procedures can be used to modify the environment in which SRJE is to run.

ALTERSDL Procedure

The ALTERSDL procedure is used to modify parameters for the communications adapter. You can use the STATUS COMM control command to determine the current parameter values before you use the ALTERSDL procedure. Use of ALTERSDL alters the values in the communications configuration record for the display station that requested the procedure. The new values apply only if SRJE is loaded from the same display station that ran the ALTERSDL procedure.



BRATE: BRATE-F specifies that the full rated speed of the modem is to be used. BRATE-H specifies that the modem is to be used at one half its rated speed.

CLOCK: CLOCK-Y specifies that the System/34 must provide the programmed clocking facility. CLOCK-N specifies that the modem or some other external device must provide the clocking facility.

LNUM: Specifies the number of the communications line to which the parameters being specified by ALTERSDL are to apply. The value can be 1, 2, 3, or 4. The default value is 1.

SLINE: SLINE-Y specifies that a switched network backup line is to be used as the backup (standby) for the nonswitched primary line. SLINE-N specifies that the switched network backup line is not used.

TEST: TEST-Y specifies that an IBM modem is being used and that automatic modem testing will be performed when a permanent error occurs. TEST-N specifies that a non-IBM modem is being used and that automatic modem testing will not be performed.

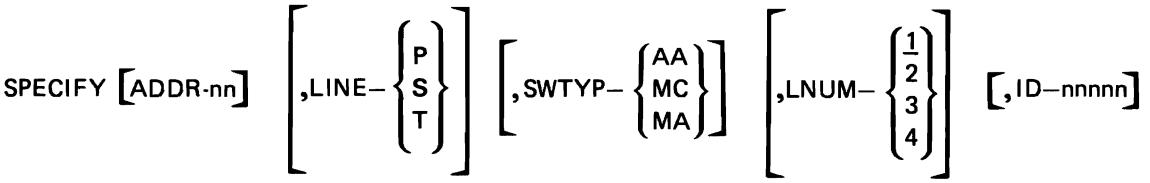
TONE: TONE-Y specifies that a non-US special tone is required for manual answer and automatic answer. TONE-N specifies that a special tone is not required.

Any changes made by the ALTERSDL procedure remain in effect until:

- The items are changed again by the ALTERSDL procedure or the $SETCF utility program.
- The system is configured again.
- The system library is reloaded. The parameters are then set as follows:
  - CLOCK, TEST, and TONE are set to the values specified during microcode configuration.
  - The switched network backup line is not used (SLINE-N).
  - The full rated speed of the modem is used (BRATE-F).

SPECIFY Procedure

The SPECIFY procedure is used to specify parameters relative to the SDLC communications line. You can use the STATUS COMM control command to determine the current values of the parameters before you use SPECIFY. Use of SPECIFY alters the values in the communications configuration record for the display station that requested the procedure. The new values apply only if SRJE is initiated from the same display station that ran SPECIFY.



ADDR: Specifies the 2-character hexadecimal SDLC address of the secondary station.

LINE: Specifies the line facility. LINE-P specifies a point-to-point nonswitched line; LINE-S specifies a point-to-point switched line; and LINE-T specifies a secondary station on a multipoint line.

SWTYP: Specifies the switch type for a switched line (LINE-S). SWTYP-AA specifies that if the modem is in automatic answer mode, the System/34 will automatically answer an incoming call. SWTYP-MC specifies that the System/34 operator must manually initiate the call. SWTYP-MA specifies that the System/34 operator must manually answer an incoming call.

LNUM: Specifies the number of the communications line for which the parameters being defined by SPECIFY are to apply. Valid values are 1, 2, 3, or 4. The default value is 1.

ID: Specifies the 5-character hexadecimal number to be used in an exchange of identification between the host system and the System/34 secondary station.

Any changes you make with the SPECIFY procedure remain in effect until:

- The items are changed again by the SPECIFY procedure or the $SETCF utility program.
- The system is configured again.
- The system library is reloaded. All parameters are reset to the values specified during microcode configuration.

RUNNING AN SRJE SESSION

An SRJE session consists of four separate phases, as shown by the following illustration:

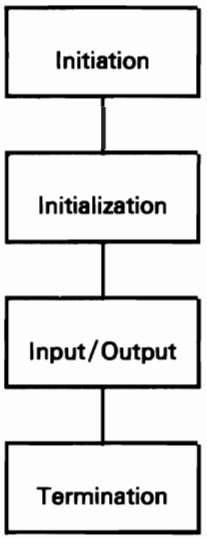


Figure 7-1. The Four Phases of SRJE Operation

The phases must be performed in the order shown in the illustration. A description of each phase follows.

Initiation

The SRJE utility can be initiated from any display station. The display station that first initiates SRJE becomes the SRJE console for that session. The SRJE console can operate as a reader input device or as the SRJE console. The modes of operation for the SRJE console are discussed later in this chapter. The SRJE utility is initiated when you enter the SRJE procedure command from a display staton.

```
SRJE [name     ] ,[PRIORITY] ,[NOAUTO] ,[YES] ,[1 ] ,[phone list name] ,[1 ]
     [#SR@ID   ]               [AUTO  ]  [NO ]  [2 ]                    [2 ]
                                         [   ]  [3 ]                    [3 ]
                                         [   ]  [4 ]                    [4 ]
                                                                        [5 ]
                                                                        [6 ]
                                                                        [7 ]
                                                                        [8 ]
                                                                        [9 ]
                                                                        [10]
```

name: Specifies the name of the initialization display format. The name specified can be a user generated format or the default format, #SR@ID. Information on developing your own initialization display format is given later in this chapter.

PRIORITY: Specifies that the tasks activated by SRJE will have user priority. If this parameter is omitted, SRJE-initiated tasks will not have priority.

NOAUTO/AUTO: Specifies whether this SRJE session is to be run in attended or unattended mode. AUTO specifies unattended operation for this SRJE session. During unattended operation, SRJE uses the default values given on the initialization display. The System/34 operator cannot change any of the initialization display format values; therefore, it is essential that the correct initialization display format be specified. During unattended mode, SRJE will take the default response to any message presented by the utility. An operator may have to be available to respond to System/34 messages.

NOAUTO, the default value, specifies attended mode of operation. An operator must be present during attended mode of operation.

Figure 7-2 illustrates the differences between unattended and attended modes during initiation.

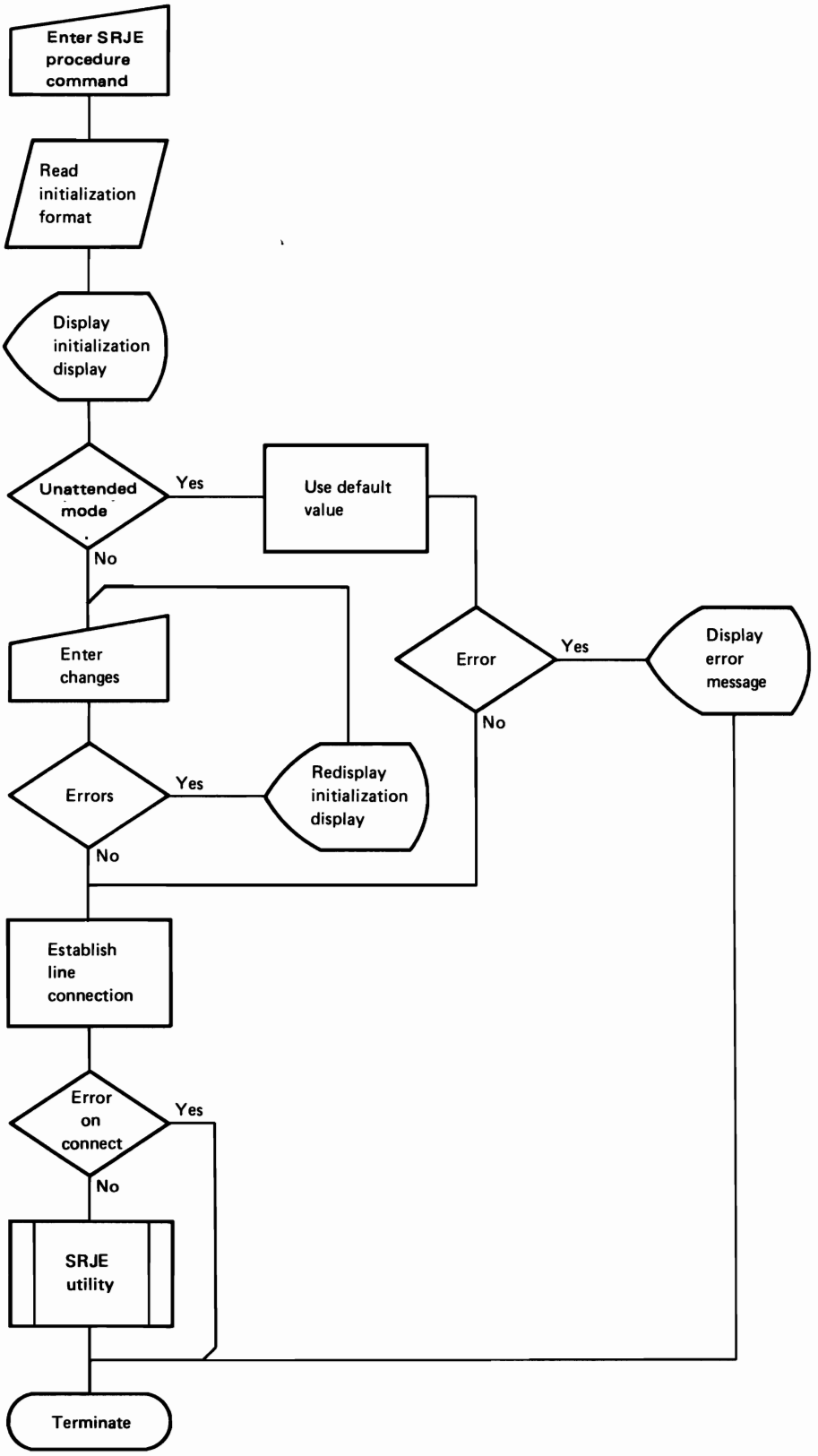


Figure 7-2. Initiation Differences Between Unattended and Attended Modes

YES/NO: Specifies whether the request discontact network services request is to be sent. If request discontact is sent (YES), the line connection to the host system ends when no SNA sessions remain active. If the request discontact is not sent (NO), the line connection remains established until ended by some other means. The default is NO.

1/2/3/4: Specifies the communications line on which the SRJE session is to run. The default is 1.

phone list name: Specifies the name of the list to be used for autocall or switched X.21. The phone list is created by the DEFINEPN procedure or the DEFINX21 procedure, which are described in the *System Support Reference Manual*. This parameter applies only if your system has the multiline communications adapter with the Autocall feature or the switched X.21 feature for the communications line specified.

1/2/3/4/5/6/7/8/9/10: Specifies the number of SNA sessions to establish. The default is 1.

INITIALIZATION

After you have entered the SRJE procedure command to initiate SRJE, the initialization display appears. If you specified unattended mode in the SRJE procedure command, the initialization display appears, but you cannot make any changes to the information. If attended mode was selected in the SRJE procedure command, the initialization display appears and you are able to make changes to the displayed information.

The following illustrations show the 1920-character and 960-character initialization displays that are provided as a part of the SRJE utility.

```
SRJE OPTION MENU--PRESS ENTER TO CONTINUE

ENTER CONFIGURATION INFORMATION BELOW:

PLUNAME..            LOGON.... N         FSN...... 001
FID...... PR1        OUTPUT... P1        SPACE.... 0060      DEVID.... SYSTEM
FID...... PU1        OUTPUT... N         SPACE.... 0060
RD1NAME..            RD1TYPE.. D         RD1CMD... Y         RD1DATE.. 000000
RD1LIBR..            RD1XPC... N         RD1DEL - N

ENTER LOGON INFORMATION BELOW:

LOGON

ENTER CARRIAGE AND FORMS INFORMATION BELOW:

L.... 066    B.... 066    F.... STD.   FCTNAME..
1.... 001    2.... 000    3.... 000    4.... 000   5.... 000    6.... 000
7.... 000    8.... 000    9.... 000    10... 000   11... 000    12... 000


**********          SYSTEM OPERATOR MESSAGE BELOW          **********
```

```
SRJE OPTION MENU--PRESS ENTER TO CONTINUE

ENTER CONFIGURATION INFORMATION BELOW
PLUNAME..            LOGON.... N         FSN...... 001
FID...... PR1        OUTPUT... P1        SPACE.... 0060      DEVID.... SYSTEM
FID...... PU1        OUTPUT... N         SPACE.... 0060
RD1NAME..            RD1TYPE.. D         RD1CMD... Y         RD1DATE.. 000000
RD1LIBR..            RD1XPC... N         RD1DEL - N

ENTER LOGON INFORMATION BELOW:
LOGON
```

```
SRJE OPTION MENU--PRESS ENTER TO CONTINUE

ENTER CARRIAGE AND FORMS INFORMATION BELOW:

L.... 066    B.... 066    F.... STD.   FCTNAME
1.... 001    2.... 000    3.... 000    4.... 000   5.... 000    6.... 000
7.... 000    8.... 000    9.... 000    10... 000   11... 000    12... 000


**********          SYSTEM OPERATOR MESSAGE BELOW          **********
```

The initialization display consists of a set of parameters with a default value given for each parameter. If you want to run SRJE with the values shown, key the logon information and press the Enter/Rec Adv key. If you want to change any of the values, position the cursor under the value you wish to change and enter the new value. After you have made all the changes, press the Enter/Rec Adv key.

PLUNAME: Specifies the name (eight characters maximum) by which the host system is known in the network. PLUNAME is optional and should be specified when simulated logon is used. The logon ensures that SRJE is communicating with the correct network application program.

LOGON: Specifies whether a logon statement is to be used during SNA initialization. Y indicates that the logon specified on the initialization display is to be used. N specifies that a logon command is not to be used. A logon command must be sent unless the host system requests a simulated logon for the SRJE terminal. The LOGON parameter must be 'Y' if the logical unit mode was specified as 'B-Multiple' in the SNA/SDLC parameter display (see *SRJE Configuration*).

FSN: Specifies the identifying number for the first file on disk. The value specified must be in the range 001 to 999, with the default being 001. Any print or punch data can be stored on disk with the output for each job being stored in a unique file. The number specified for the FSN parameter is used to number the first file and is then incremented by one for each subsequent file created. If a file number is already in use, SRJE uses the next higher available number. The format of the file name is nxxxffff with the n representing A for print files or B for punch files, the xxx representing the numeric identifier (the first file created will have the value specified for the FSN parameter), and ffff represents the first four characters of the forms name, if one is given, or four blanks if no name is given.

FID: Specifies the task to be modified. FID-PR1 indicates the print task; FID-PU1 indicates the punch task.

Note: All print tasks (PR2 and PR3) will start with the same configuration as PR1. PU2 and PU3 will start with the same configurations PU1.

OUTPUT: Specifies the device for print or punch data.

OUTPUT-P directs print data to the printer and is meaningful only for FID-PR1. In the initialization display, OUTPUT-P1 is specified. The 1 indicates the spool priority of the data. This entry is optional and, if specified, must be a value from 0 to 5, with the default being 1.

OUTPUT-D directs print or punch data to temporary disk files. Print data is written in 136-byte records and punch data in 80-byte records.

OUTPUT-S directs print or punch data to temporary disk files with 256-byte (SNA) records.

OUTPUT-N specifies that SRJE will not accept data for the appropriate task, as specified for FID.

SPACE: Specifies the number of blocks to be allocated for disk files. The value entered is a decimal value in the range 1 to 9999. SPACE is meaningful only when either OUTPUT-D or OUTPUT-S has been specified.

DEVID: Specifies the System/34 work station ID (or SYSTEM for the system printer) of the printer to use when print data is directed to the printer. DEVID is only applicable when OUTPUT-P is selected for FID-PR1.

RD1NAME: Specifies the disk file or library member that is to be read automatically by the reader task. If a name is specified, that disk file or library member will be read by SRJE upon completion of the initialization phase. If no entry is made, no file or library member is read automatically. The default on the IBM-supplied initialization display is a blank entry.

The RD1NAME parameter is used only to name a file or library member that is to be read automatically after initialization. Other files or library members can be read by using the READFILE utility control statement, which is described later in this chapter.

If you are defining an initialization format for unattended mode, you must specify the name of a file or library member for the RD1NAME parameter.

RD1TYPE: Specifies the type of file or library member named for the RD1NAME parameter. RD1TYPE-D, the default on the IBM-supplied initialization display, specifies a disk file. RD1TYPE-S specifies a source member in the library defined by the RD1LIBR parameter. RD1TYPE-P specifies a procedure member in the library defined by the RD1LIBR parameter.

RD1CMD: Specifies whether the file named for the RD1NAME parameter is a command file. RD1CMD-Y, the default on the IBM-supplied initialization display, specifies that the file is a command file. SRJE processes any utility control statements it encounters when reading a command file. RD1CMD-N specifies that the RD1NAME file is a data file. Utility control statements encountered by SRJE in a data file are not processed, but are transmitted to the host system as data.

RD1DATE: Specifies the creation date of the disk file named for the RD1NAME parameter. The RD1DATE parameter is used only when a file is specified for RD1NAME and more than one file with that name exists on disk. The date is entered in the system date format. The default on the IBM-supplied initialization display is 000000.

RD1LIBR: Specifies the name of the library to be searched for the source or procedure member specified for the RD1NAME parameter. The default is the system library.

RD1XPC: Specifies whether the file specified for the RD1NAME parameter contains any transparent data (byte values less than hex 40). RD1XPC-Y indicates that the file does contain transparent data. If you slected RD1CMD-Y, RD1XPC-Y cannot be selected. RD1XPC-N, the default on the IBM-supplied initialization display, specifies that the file does not contain transparent data.

RD1DEL: Specifies whether deleted records should be transmitted to the host system. RD1DEL-Y indicates that deleted records should be transmitted to the host system. This selection is valid only on systems configured with extended disk management. In addition, the file specified for RD1NAME must be delete-capable and RD1TYPE-D must be selected. RD1DEL-N, the default on the IBM-supplied initialization display, specifies that deleted records are not to be transmitted to the host system.

ENTER LOGON INFORMATION BELOW: Specify the logon command that is to be sent to the host system when the communications link is established. Enter the command only if the LOGON parameter is specified as Y. The format of the logon command is:

LOGON APPLID(name1) LOGMODE(name2) DATA(userdata)

APPLID: Specify the name of the host system application program with which the session is to be established. Standard entries are JES2 for the OS/VS2 JES2 host, JES3 for the OS/VS2 JES3 host, RTAM for the OS/VS1 RES host system, and POWER for the DOS/VS POWER/VS host system.

LOGMODE: Specify the name of a set of parameters that are to be used for the session. The name specified here must be the same as the name specified for the LOGMODE parameter in the MODEENT macro for the BIND construction at the host.

DATA: Specifies a set of user parameters for the host. For example, DATA('A0 T(5) PROC(FSPROC)') defines the user A0, the terminal number 5, and the name of the host procedure that is to be run, FSPROC.

ENTER CARRIAGE AND FORMS INFORMATION BELOW: This group of parameters is used to specify information related to forms and carriage control.

L: Specify the number of lines per page for print data. The default on the IBM-supplied initialization display is 66.

B: Specify the number of the last line to be printed on a page (the bottom margin). The default on the IBM-supplied initialization display is 66.

F: Specify the initial forms output number or name for SRJE. Any string of characters can be entered. Note that the host system must be generated with PDIRs required for proper forms control. STD. is the default on the IBM-supplied initialization display.

FCTNAME: Specify the name of the forms control table built with the $DCFUP utility. Refer to *Print Output* later in this chapter for more information on forms processing.

The remaining parameters, the numbers 1 through 12, are used to define forms movement. The value specified for each parameter can be any decimal number from 0 to the value specified for the forms length (the L parameter). The value specified for a parameter corresponds to line number withn a page of the form. When the host system sends print output to the SRJE utility, the first four characters are channel control characters that defines how the form is to be moved before or after printing. For example, if channel control character 1 is received, the form is moved to the line specified for the 1 parameter. Refer to the description of the print task later in this chapter for more information on the channel control characters and forms movement. The default values on the IBM-supplied initialization display are 1 for parameter 1 and 0 for all other parameters.

The last area on the initialization screen is the system operator message area. In this area, you can enter any message, up to 160 characters, that you want to appear on the status display of any System/34 work station that attaches to the SRJE utility (by entering the SRJE procedure command).

After all the changes have been made, press the Enter/Rec Adv key and the SRJE utility is initialized. If any errors are detected in the options entered, the errors appear in reverse image and the following message appears (attended mode only):

ERRORS – CORRECT FIELDS AND PRESS ENTER TO RETRY

In unattended mode, a message is displayed and SRJE terminates.

For each field that appears in reverse image, position the cursor to that field and enter the appropriate correction. Once all corrections have been made, press the Enter/Rec Adv key. If no invalid entries are detected, the SRJE initialization phase proceeds.

Input

Input to the SRJE utility can come from disk, a display station, or the SRJE console. The types of input allowed are data, host job control language (JCL) statements, host commands, and SRJE utility control statements. Not all types of input can be entered from all the input devices. Under normal operating conditions, data, JCL, and host commands are sent to the host system for processing, but SRJE utility control statements are processed by the SRJE utility. Disk and display station input occurs under the control of a reader task; SRJE console input can be under the control of the SRJE console task or the reader task.

Figure 7-3 defines the input classifications and device relationships.

Task	Input Device	Input Type
Reader	Disk Data file Source member Procedure member	Data, JCL, utility control statements,[1] host system commands
	Display station keyboard	
	SRJE console keyboard in reader mode	
SRJE console	SRJE console keyboard in console mode	Host system commands, utility control statements

[1]Not all utility control statements are allowed from the reader task. Refer to the description of the utility control statements later in this chapter for more information.

Figure 7-3. Input Classifications and Device Relationships

A description of the four types of input listed under *Input Device* follows.

Disk Input

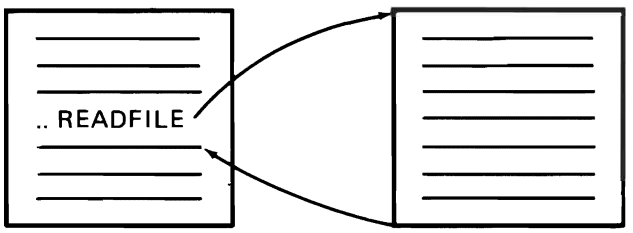
Disk input to the SRJE utility is specified by using the READFILE utility control statement or by specifying the RD1NAME parameter on the initialization display. When the disk is to be used as an input device for SRJE, you specify the following:

- The name of the file or library member to be read
- Whether the file or member is a command file
- The type of file or member (data, source, procedure)
- The file creation date
- The name of the library to be searched for a member
- Whether deleted records are to be transmitted

For a more complete description of these parameters, refer to the discussion of the initialization phase earlier in this chapter or the READFILE statement later in this chapter. Note that any disk file or library member used as input to the SRJE utility cannot be shared by other executing programs.

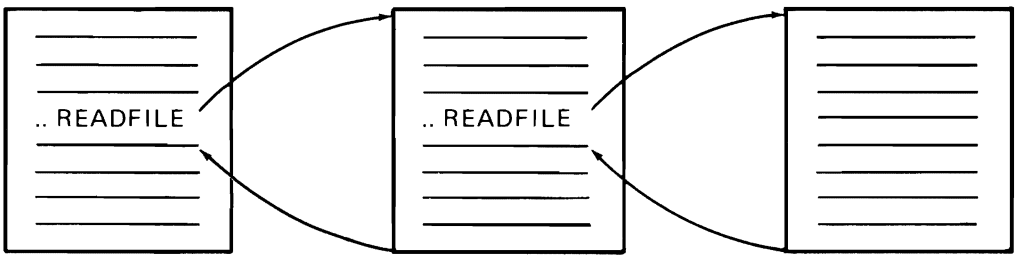
All files or library members used as input to SRJE can be designated as either command or data files. Both types can contain data, JCL, and host system commands. A command file can also contain some of the SRJE utility control statements (READFILE, EOF, LIBRARY, and, if in unattended mode, END), which are processed by the SRJE utility rather than being transmitted to the host system. If utility control statements are encountered in a data file, they are ignored by the SRJE utility and transmitted to the host system as data.

When a READFILE statement that specifies a data file is detected while SRJE is reading a command file, reading of the command file is suspended and reading of the data file begins. Once processing of the data file has completed, the reader task returns to the command file at the statement immediately following the READFILE statement.



When a READFILE statement that specifies another command file (B) is encountered while processing a command file (A), processing of command file A suspends and SRJE begins processing command file B. Once processing of command file B has completed, the reader task returns to command file A at the statement immediately following the READFILE statement.

Command file B could also contain a READFILE statement that specifies another command file or a data file. For example, command file A contains a READFILE statement that specifies command file B, which contains a READFILE statement that specifies data file C. When the READFILE statement in command file A is read, processing of command file A is suspended and the reader task begins processing command file B. When the READFILE statement in command file B is read, processing of command file B suspends and the reader task begins processing data file C. When processing of data file C completes, the reader task returns to command file B at the statement immediately following the READFILE statement that specified data file C. When processing of command file B has completed, the reader task returns to command file A at the statement immediately following the READFILE statement that specified command file B.



The maximum number of levels of nesting is 255, but the practical limit depends on the amount of space available in your assign/free area and the amount of available disk space. See the *Planning Guide* for more information.

The SRJE utility transmits the information from data files as 80-byte records, regardless of the logical record length within the file. For example, a file containing four 60-byte logical records is transmitted as three 80-byte records. A data file containing four 120-byte records is transmitted as six 80-byte records. The customer application program at the host system must return the data file to its original record length.

A command file must always contain 80-byte logical records.

Data files may be delete-capable or non-delete-capable. A delete-capable file contains records of hex FF if the records have been deleted. If the data file is allocated as a direct file, all records in that file are initialized to hex FF.

SRJE transmits delete-capable files with the deleted records removed (RD1DEL-N on the initialization screen or DEL-N in the READFILE statement), or with deleted records in place (RD1DEL-Y or DEL-Y).

Display Station Input to the Reader Task

The SRJE reader task begins when an operator at a display station enters the SRJE procedure command, with no parameters, after SRJE has been initialized. If the reader task is not available, SRJE issues a message to the display station.

When the reader task begins, the following display appears (with the initialized values displayed instead of the x's) on a 1920-character screen:

```
 SRJE STATUS MENU--PRESS ENTER TO CONTINUE



 PLUNAME.. XXXXXXXX  LOGON.... X         FSN...... XXX
 FID...... PR1       OUTPUT... XX        SPACE.... XXXX      DEVID.... XXXXXX
 FID...... PU1       OUTPUT... X         SPACE.... XXXX



 LOGON INFORMATION BELOW:

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

 CARRIAGE AND FORMS INFORMATION BELOW:

 L.... XXX     B.... XXX     F.... XXX   FCTNAME.. XXXXXXXX
 1.... XXX     2.... XXX     3.... XXX   4.... XXX     5.... XXX      6.... XXX
 7.... XXX     8.... XXX     9.... XXX   10... XXX     11... XXX      12... XXX


 ************            SYSTEM OPERATOR MESSAGE BELOW            ************
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

On a 960-character screen, two displays are presented:

```
SRJE STATUS MENU--PRESS ENTER TO CONTINUE

PLUNAME.. XXXXXXXX LOGON.... X          FSN...... XXX
FID...... PR1      OUTPUT... XX         SPACE.... XXXX      DEVID.... XXXXXX
FID...... PU1      OUTPUT... X          SPACE.... XXXX


LOGON INFORMATION BELOW:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
SRJE STATUS MENU--PRESS ENTER TO CONTINUE

CARRIAGE AND FORMS INFORMATION BELOW:

L.... XXX    B.... XXX    F.... XXXX   FCTNAME.. XXXXXXXX
1.... XXX    2.... XXX    3.... XXX    4.... XXX   5.... XXX     6.... XXX
7.... XXX    8.... XXX    9.... XXX    10... XXX   11... XXX     12... XXX

************           SYSTEM OPERATOR MESSAGE BELOW           ************
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

The values in the display indicate the current status of the SRJE configuration. If you choose not to continue after seeing the status display, press the Attn key to terminate the reader task and return to the System/34 command display. If you wish to continue, press the Enter/Rec Adv key and the display appears the same as the system input/output display with a reverse image RD1, RD2, or RD3 on line 1. The reader display indicates that utility control statements, JCL, and data can be entered from the display station keyboard, one record at a time. Up to 80 bytes of data per record can be entered from the keyboard.

If you enter the SRJE procedure command while SRJE is operating in unattended mode, the status display does not appear. If the reader is available, the reader input display is presented and you can enter reader data.

The reader task at the display station operates as if it were reading a disk command file. Input from the display station can be data, JCL, and certain utility control statements (READFILE, EOF, and LIBRARY). The reader task terminates at a display station when any of the following conditions occur:

- An EOF utility control statement is entered from the station.
- A CANCEL command for the display station reader task is entered from the SRJE console.
- An EOF utility control statement is encountered by the reader task in a command file; the reading of which was specified by a READFILE utility control statement entered from the station.
- Option 2 or option 3 is selected from the System/34 inquiry display at the display station.

Whenever a reader task is canceled, any data already entered is transmitted to the host system. When the reader task terminates at a display station, press the Enter/Rec Adv key to return the station to the System/34 command display screen.

When a READFILE utility control statement is entered from the display station while the reader task is active, the reader task begins reading the specified file. The reader task is suspended at the display station until the file has been processed. A message is then displayed at the display station indicating that the file has been processed and that the reader task is active at the display station. The display station is not available for input operations while the file is being processed.

Console Input to the Reader Task

To activate the reader task at the SRJE console, press the Attn key, wait for the console input screen to appear, and then press command key 1 any time after initialization has completed. When the Enter/Rec Adv key is pressed, the SRJE reader task input display appears. The SRJE reader task input display is the same as the System/34 input/output display with a reverse image RD1, RD2, or RD3 appearing on line 1.

The reader display indicates that the SRJE utility is ready to accept data, JCL, or certain utility control statements (READFILE, EOF, LIBRARY, WTO, and END if in unattended mode) that are entered from the SRJE console. Records entered can be up to 80 bytes in length.

The reader task at the console processes data in the same manner as it processes a disk command file. When the READFILE statement is entered from the SRJE console while the reader task is active, the reader task processes the specified file.

Termination of the reader task at the SRJE console occurs under any of the following conditions:

- An EOF utility control statment is entered.
- A CANCEL command for the reader task is entered from the SRJE console while it is in console mode.
- An EOF statement is encountered within a command file, the reading of which was specified by a READFILE statement entered from the console.
- Cancel by system console operator.

Once the reader task terminates at the SRJE console, the console is automatically prepared for console output.

Console Input to the Console Task

Input to the console input task can be entered only from the SRJE console. The console input display appears when the SRJE console operator presses the Attn key any time after initialization.

The console input display is the same as the system input/output display with a reverse image CNI on line 1.

Input to the console task can be host commands or MODIFY, CANCEL, FORMS, or END utility control statements. Up to 120 characters of input can be entered at one time and all SRJE console input is logged to the history file. Any record consisting of blanks only is ignored.

While the reader task is active at the SRJE console, you can enter console input mode by pressing the Attn key and then the Enter/Rec Adv key. The console input field replaces the reader input field and you can enter console input. You can then enter console input one record at a time. When you press the Enter/Rec Adv key, the data is entered and the display station returns to reader mode.

Output

The SRJE utility can consist of one, all, or a combination of the following types of output:

- Print data from the host system (print output)
- Punch data from the host system (punch output)
- Operator messages from either the host system or the SRJE utility (console output)

The SRJE operator can specify, by using the initialization display or the MODIFY utility control statement, whether print output is to be written to a printer or to disk. Print output written to disk can be printed using the $DCSUP utility, which is discussed in Chapter 5 of this manual, or processed by a user-written program.

Printer forms control operations are controlled by channel control characters and the forms table. Initial carriage information is taken from the SRJE initialization display. The values can be changed directly from a forms control table entry or, they may be changed by the SRJE operator when the following display appears:

```
PR1 RECEIVING JOB OUTPUT FOR

          JOB NAME... XXXXXXXX   DATE... XX/XX/XX TIME... XX.XX.XX
            RECORDS... XXXXXXXX   PRINT IMAGE NAME... XXXXXXXX

ENTER CARRIAGE AND FORMS INFORMATION BELOW
        HOST FORMS NAME... XXXXXXXX   LOCAL FORMS NAME... XXXX
  L... XXX   B... XXX   1... XXX   2... XXX   3... XXX   4... XXX   5... XXX
  6... XXX   7... XXX   8... XXX   9... XXX   10.. XXX   11.. XXX   12.. XXX
```

The forms control table is created by the $DCFUP utility, which is described in Chapter 6 of this manual. Direct substitution from the forms control table occurs when a PDIR (peripheral device information record) is received from the host or a FORMS statement is entered through the SRJE console, and the following conditions are met:

- A valid forms control table name was specified for the FCTNAME parameter on the initialization display.
- The host forms name received in the PDIR or specified in the FORMS statement is found in the forms control table.
- The halt flag is off in the table entry for that forms name.

If the halt flag is on, carriage information from the forms control table is displayed and may be changed by the SRJE console operator. If either of the other conditions is not met, the current values are displayed and may be changed by the SRJE console operator. The flow of processing for print output is shown in Figure 7-3.

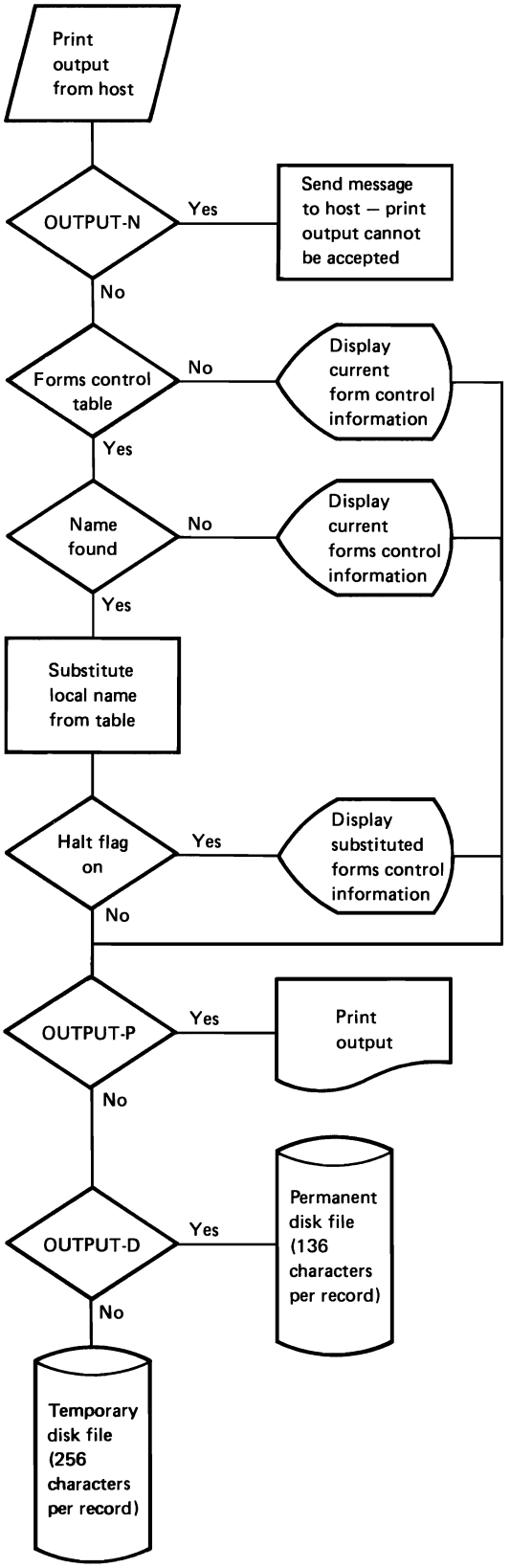


Figure 7-4. Print Output Processing by SRJE

Print output is written to disk in a file of either 136-character records or 256-character records. SRJE does not create delete-capable files. The $DCSUP utility, described in Chapter 5, can print these files and can create 136-character record files from the 256-character record files. The first four bytes of the 136-character records have one of the following formats to control the forms movement for a print file created by SRJE:

1. Vertical tab table identifier

	S	0	1	n
byte	1	2	3	4

The characters S01 in the first three bytes of a record in the print file indicate that a vertical tab table follows. The fourth byte (n) specifies the length of the vertical tab table, including the length byte. The vertical tab table format is:

	number of lines per page	line number of bottom margin	channel number(s)
byte	5	6	7-n

2. Channel control character bytes

	first channel operation before printing	second channel operation before printing	first channel operation after printing	second channel operation after printing
byte	1	2	3	4

The *before* channel operations are performed before the record is printed and the *after* operations are performed after the record is printed.

Each of the four channel control character bytes has the following bit meanings:

0 = 0 Space operation
= 1 Channel operation

For space operations:
1-7 = The number of lines to be spaced (up to 127)

For channel operations:
1-7 = 0 Vertical tab

The vertical tab table is searched for the next entry whose value is greater than the current line number. This number is used as the line number for a skip operation.

1-7 = n Channel number

The value of bits 1-7 is a channel number. The channel number is used as an index into the vertical tab table where a line number is found that corresponds to the channel number.

If the records are 256 bytes long, the record following the PDIR in the file has the following format:

	S	0	2	n	control byte	not used	vertical tab table length (v)	horizontal tab table length (h)	vertical tab table area	horizontal tab table area
byte	1	2	3	4	5	6	7	8	next v bytes	next h bytes

The characters S02 in the first three bytes indicate that this is a 256-character record. The fourth byte (n) specifies the length of the special record information (253 bytes maximum, including the length byte). Byte 5 (the control byte) indicates:

Hex 80 = The record is in compressed format.
Hex 20 = A vertical tab table is present in this record.
Hex 10 = A horizontal tab table is present in this record.
Hex 02 = This is a print file (logical records are 132 bytes).
Hex 01 = This is a punch file (logical records are 80 bytes).

The SRJE utility requires peripheral data set information records (PDIR). When SRJE receives a PDIR, the print task writes a special disk record if SRJE is directing print data to the disk. SRJE closes the current file, allocates and opens a new file, and writes the special record to disk. The PDIR is the first record in a print file.

Note: You must request PDIRs from the host system during host system configuration.

The format of the PDIR is:

Byte	Field Meaning	Value
0-2	Record identifier	S03
3	Length	Hex 42
4	Identifier	Hex 00 = Ordinary data set Hex 01 = Job separators Hex 02 = System messages data set
5-12	Date	MM/DD/YY
13-20	Time	HH.MM.SS.
21-28	Forms name	8 characters (standard forms if blank)
29-36	Forms control block name	8 characters (standard FCB if blank)
37-44	Train name	8 characters (standard train if blank)
45-52	Number of additional copies	EBCDIC characters for digits (leading zeros suppressed except last digit), right-justified
53-60	Volume of I/O	EBCDIC characters for digits (leading zeros suppressed, except last digit), right-justified; for printer, approximate number of print lines; for card, approximate number of card images.
61-68	Job name	8 characters

Printer output (either 136- or 256-character records) directed to a disk file is labeled by the SRJE utility, using the FSN parameter from the initialization display, as Axxxffff, where:

A = Printer output disk file
xxx = File sequence number (FSN)
ffff = Forms name or number, which is the local forms name if FCT was specified on the initialization display or the host forms name if FCT was not specified.

Punch Output

The SRJE utility writes all punch output from the host to disk files. The 256-byte records from the host contain compressed punch records. The SRJE utility creates punch output on disk as 80-byte records or 256-byte records. Punch output written as 256-character records resides on the disk in the form that the SRJE utility received the data. Punch output written as 80-character records has been decompressed to 256-byte records. Punch output on disk can be printed using the $DCSUP utility, which is described in Chapter 5, or processed by a user-written program.

The SRJE utility requires less disk space and processing time because SNA does not process punch data before it is written to disk. $DCSUP can print the data or create a disk file with 80-byte records from the 256-byte record file. The 80-byte records created by $DCSUP are the same as the 80-byte records created by SRJE.

Punch output directed to a disk file is labeled by the SRJE utility, using the FSN parameter from the initialization display, as Bxxxfff, where:

B = Punch output disk file
xxx = File sequence number (FSN)
ffff = Forms number, which is the host forms name

Note: The SRJE utility does not create delete-capable files.

Console Output

The console output task displays host system messages on the SRJE console display screen. Each task displays its own SRJE messages. In addition, all messages are written to the system history file. Host system messages can be up to 120 characters in length. If a message is longer than 75 bytes, it is displayed on two lines. The entire message is written to the system history file and can be printed.

The SRJE utility handles all reader messages as 75-character messages (for both logging and displaying). If the display station that was the reader has been released, SRJE logs the messages to the SRJE console.

Termination

Either the SRJE console operator or the host system operator can terminate a session. The SRJE console operator terminates a session normally by entering a host system sign-off command or an END utility control statement.

When SRJE terminates normally, it logs the following:

INPUT RECORDS READ - nnnnnn
OUTPUT RECORDS PRINTED - nnnnnn
OUTPUT.RECORDS PUNCHED - nnnnnn
SNA FORMAT RECORDS WRITTEN - nnnnnn

The SRJE console operator can terminate the session abnormally by entering the CANCEL SRJE utility control statement. If SRJE terminates abnormally, no sign-off command is sent to the host system.

SRJE UNATTENDED MODE OF OPERATION

Unattended mode allows the SRJE utility to operate with a minimum of operator intervention. Operator action is required only if a System/34 message is displayed. For SRJE messages (those containing SRJE in the text) and host messages (such as forms mount messages), the SRJE utility selects the default option and continues processing. Refer to the *Displayed Messages Guide* for information on the SRJE messages and defaults.

Before SRJE can be run in unattended mode, you must have an initialization display format that contains the necessary information for the SRJE session. In unattended mode, no changes can be made to the default values given on the display.

The following points must be considered before you initiate SRJE in unattended mode:

- Initialization in unattended mode occurs without operator intervention. Therefore, the default values on the initialization format specified must be the correct values for the session.

- The initialization format must specify RD1NAME, RD1DATE, RD1CMD, RD1LIBR, and RD1DEL parameters appropriately so that a file or library member will be read immediately after initialization.

Initiation for Unattended Mode

To initiate SRJE for unattended mode of operation, specify AUTO in the SRJE procedure command and specify the proper initialization display format name for the *name* parameter.

Initialization in Unattended Mode

Once the SRJE procedure command has been entered with AUTO specified, the initialization phase proceeds without operator intervention. Communications are established in the same manner as for attended mode. Note that the operator may be required to dial the host system to establish the connection or to answer the host call to establish the connection.

Input in Unattended Mode

In unattended mode, reader input is usually from disk and is usually initiated by the RD1NAME parameter on the initialization display. The processing of files is the same as for attended mode. Proper use of command files and data files allows multiple files to be processed in one SRJE session. For example, the file specified for the RD1NAME parameter would normally be a command file, and it would contain one or more READFILE statements that cause one or more data files to be read. That command file could also reference another command file in a READFILE statement.

The END utility control statement is permitted in a command file during unattended mode, and its processing causes SRJE to enter the termination phase. You can use the DELAY parameter in the END utility control statement to delay the sending of the sign-off command to the host for a specified number of minutes.

Console input, as described for attended mode, is permitted during unattended mode of SRJE processing. However, use of console input overrides the purpose of unattended mode since it requires the presence of an operator.

Output in Unattended Mode

During unattended mode, output is handled in much the same way as during attended mode. The only difference is that during unattended mode any messages generated by the SRJE utility are handled automatically. When in unattended mode, SRJE selects the default option for all messages except System/34 messages. The System/34 operator may have to respond to messages generated by the System/34. For a description of the SRJE messages and the defaults, refer to the *Displayed Messages Guide.*

Termination of SRJE in Unattended Mode

Termination of SRJE while in unattended mode occurs whenever the END utility control statement is encountered in a command file or whenever the END statement is entered from the console. Once the END statement has been entered, termination proceeds in the same manner as described for attended mode of operation. It is recommended that you specify a DELAY parameter in the END statement to ensure that all output from the host is received.

ALTERING THE INITIALIZATION SCREEN FORMAT

If you are going to be running SRJE with the same values each time, and those values differ from the values on the IBM-supplied initialization screen, you can modify the display screen format so that it has your needed default options. You can also modify the format more than once to generate a display screen for each configuration you will be running.

Each screen format is identified by a unique name of eight characters or less. The name of the SRJE initialization display format must always be #SR@ID. To use one of the formats you generate for the initialization display, specify the name of the load member for that format as the *name* parameter in the SRJE procedure command. You can modify the values on the IBM-supplied initialization format, #SR@ID, but by doing so you lose the values originally supplied for the format.

The following steps can be followed to modify the values on the IBM-supplied initialization format and generate a new initialization format with your desired default values:

1. Enter: SEU #SR@ID,S,,80

2. The format name begins in column 7 of line 4. You must not change the #SR@ID name. Enter the number of the line you want to change and press the Enter/Rec Adv key.

3. The values for all other parameters start in column 57 of a given line. If a change is necessary, key the new value and press the Enter/Rec Adv key to record the changed record. If no changes are necessary, press the Enter/Rec Adv key to advance to the next line. Certain parameters (such as SIGNON and MESSAGE) may require more than one line. For these parameters, begin keying in column 57 of the first line and continue keying through column 79, enter an X in column 80 to indicate continuation, and press the Enter/Rec Adv key. You may continue keying on the next line beginning in column 7.

   Note: Be certain that you change only the parameter values; do not alter any other characters in the format.

4. When you have made all necessary changes, press Command Key 7 to exit from SEU. Select one of the options from the menu that is displayed to end the job.

5. Enter: FORMAT CREATE,name,,#SR@ID

   This step converts your source member to an object member.

To run SRJE using your default values, specify your format load member (the *name* parameter in the FORMAT procedure) for the *name* parameter in the SRJE procedure. The initialization screen that appears is the one you generated and it contains your default values. You can modify any value on the screen in the same manner as you modified those values on the IBM-supplied initialization screen format. Press the Enter/Rec Adv key after all changes have been made to the display or to accept all default values.

SRJE UTILITY CONTROL STATEMENTS

This section contains:

- An introduction to the writing of SRJE utility control statements.
- The rules for coding SRJE control statements.
- The format of each utility control statement.
- A description of the parameters in the utility control statements.

Figure 7-5 summarizes under which task (and from which device) you can enter the SRJE utility control statements.

Task	Entered From	SRJE Utility Control Statement
Reader input	Keyboard Disk	READFILE END* EOF WTO LIBRARY
SRJE console input	SRJE console keyboard	CANCEL MODIFY END FORMS

Figure 7-5. Entering SRJE Utility Control Statements

*Only in unattended mode

Writing SRJE Utility Control Statements

The SRJE utility requires statements that control the SRJE session.

Each utility control statement is made up of a statement identifier and parameters. The statement identifier is always the first word of the statement; the parameters supply the information to the SRJE utility. Each parameter consists of a keyword, which identifies the parameter, followed by the specific information supplied to the utility.

The general format of an SRJE utility control statement is:

```
.. statement-identifier parameter,...,parameter comments
(.. s)
```

where .. s is the abbreviated form of the statement identifier. When a statement abbreviation is available, it can be used instead of the complete statement identifier. Statement abbreviations are shown in parentheses, but the parentheses are not entered.

Rules for Coding SRJE Utility Control Statements

The rules for coding utility control statements are:

Statement Identifier: .. (two periods) followed by one or more blanks must precede the statement identifier. Do not use blanks within the statement identifier.

Blanks: One or more blanks are required between the statement identifier and the first parameter.

Statement Parameters: Parameters can be in any order. A comma separates one parameter from another (do not use blanks between parameters), and a hyphen (-) within each parameter separates the keyword from the parameter value.

Comments: You can include comments in utility control statements if you enter parameters. Leave one or more blanks between the last parameter in the statement and the comment. Starting a statement with ..* makes the entire statement a comment.

Continuation: A utility control statement may be continued on several logical records. Each record must contain at least one parameter and must begin with two periods (..) followed by one or more blanks. Each record (except the last) must end with a comma immediately following the last parameter coded on the record. Comments may follow the final comma after one or more blanks. If there is an error on any record of a continuation, the entire utility control statement is ignored and must be reentered from the beginning of the first record.

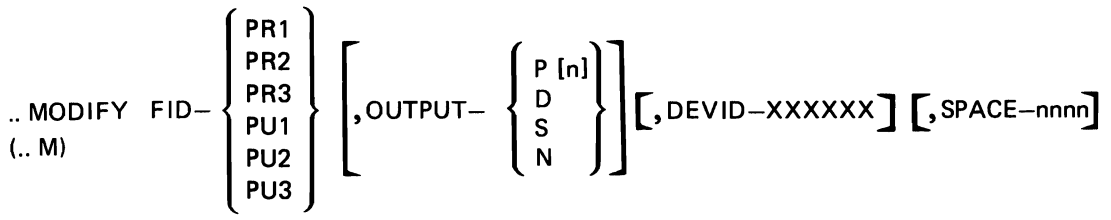
MODIFY

The MODIFY utility control statement reassigns the print and punch tasks. You can enter the MODIFY statement only from the SRJE console, after initialization is complete and before you enter the END statement.

If you enter a MODIFY statement while the specified task is processing output, the assignment does not change until the current file output ends.

The SRJE console operator should be aware of any display station using the SRJE utility when entering the MODIFY statement. A message can be sent from the SRJE console to the display station operators informing them of changes.

The format of the MODIFY statement is:



FID: Specifies the task to be modified. FID-PRn indicates a print task; and FID-PUn indicates a punch task.

OUTPUT: Specifies the device for the print or punch data.

OUTPUT-P directs print data to the printer and is valid only with FID-PRn. OUTPUT-P also activates the print task if it is not active. The optional n is the priority assigned if the data is spooled; n ranges from 0 through 5. If n is not specified, it defaults to 1.

OUTPUT-D directs print or punch data to temporary disk files. Print data is written in 136-byte records, and punch data is written in 80-byte records. OUTPUT-D also activates the appropriate task if it is not active.

OUTPUT-S directs print or punch data to temporary disk files with 256-byte (SNA) records. OUTPUT-S also activates the appropriate task if it is not active.

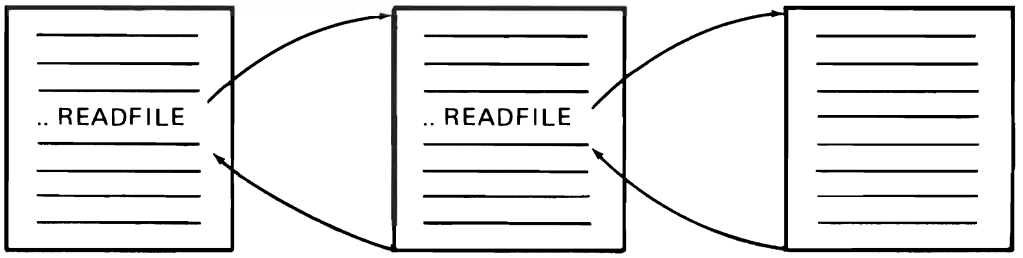
OUTPUT-N indicates that no print or punch data will be accepted and that the task specified by FID will be de-activated when the current file is completed.

DEVID: Specifies the System/34 work station ID (or SYSTEM for the system printer) of the printer to use when print output is directed to a printer. DEVID is valid only with FID-PRn and either OUTPUT-P or when print data is already going to the printer.

SPACE: Specifies the number of blocks to be allocated for disk files. The value is a decimal number from 1 to 9999. SPACE can be specified only with OUTPUT-D or OUTPUT-S, or if output for the specified task is already going to disk.

READFILE

The READFILE utility control statement describes the disk file, procedure member, or source member that is to be read by SRJE. READFILE statements can be nested; that is, a READFILE statement can specify a command file that contains another READFILE statement. After the second file is read, SRJE returns to the first file at the statment immediately following the READFILE statement that specified the second file. The following illustrates a situation where the READFILE statement in the first command file references a second command file, and the second command file contains a READFILE statement that references a data file.



When the READFILE statement in the first command file is processed, SRJE begins reading the second command file. When the READFILE statement in the second command file is processed, SRJE begins reading the data file. When the data file has been completely processed, SRJE begins reading the second command file at the statement immediately following the READFILE statement that specified the data file. When processing of the second command file has completed, SRJE returns to the first command file at the statement immediately following the READFILE that specified the second command file.

The maximum number of level of nesting is 255, but the practical limit depends on the amount of space available in your assign/free area and available disk space. See the *Planning Guide* for more information.

You can enter the READFILE statement only as reader input.

The format of the READFILE statement is:

```
.. READFILE NAME–name [,LIBR–{ name }] [,DATE–date]
(.. R)                       { 0    }

[,TYPE–{P}] [,CMD–{Y}] [,XPC–{Y}] [,DEL–{Y}]
       {S}        {N}        {N}        {N}
       {D}
```

NAME: Specifies the name of the disk file or library member that is to be read.

LIBR: Specifies the name of the library that is to be searched for the member specified by the NAME parameter. If you omit this parameter, SRJE searches the library specified on the last effective LIBRARY statement. If you enter 0, SRJE searches the system library. LIBR is only valid with TYPE-S or TYPE-P.

DATE: Specifies the file creation date. DATE is valid only with TYPE-D.

TYPE: Specifies the type of file or member specified by the NAME parameter. TYPE-S indicates a source member. TYPE-P indicates a procedure member. TYPE-D, the default, indicates a disk file.

CMD: Specifies whether the file to be read is a command file.

CMD-Y indicates that the file is a command file. SRJE processes any utility control statements it encounters while reading a command file, and transmits all other records to the host system as data. Command files are processed consecutively and must have a record length of 80 bytes. CMD-Y is not valid if XPC-Y is specified.

CMD-N, the default, indicates that the file is a data file. Data files are processed consecutively, but can be sequential, direct, or indexed with any record length. SRJE transmits all records within a data file to the host as data. If utility control statements are present in a data file, they are not processed by SRJE but are treated as data and transmitted to the host system as such.

XPC: Specifies whether the file to be read contains transparent data (byte values less than hex 40). XPC-Y indicates that the file does contain transparent data. XPC-Y is not valid if CMD-Y is specified. XPC-N, the default, indicates that the file does not contain transparent data.

DEL: Specifies whether deleted records should be transmitted to the host system. DEL-Y indicates that deleted records should be transmitted. DEL-N indicates that deleted records are not to be transmitted.

Note: If SRJE detects an error in a READFILE statement, the statement is not processed. If the display station has not been released, the operator can enter a valid READFILE statement. If the display station has been released, the SRJE reader task terminates abnormally.

EOF

The EOF (end-of-file) utility control statement signals the end of the current reader input to the host system. The host system then queues the job for execution. EOF de-activates the SRJE reader task. Only the last job entered by the reader task needs an EOF statement at the end.

When the SRJE utility encounters an EOF, no more data is requested from either the procedure or file being read from disk or from the operator via the keyboard. If SRJE does not find an EOF before the end of the file or procedure, SRJE requests input from the device from which the last READFILE was read, starting at the statement immediately following the last READFILE. If the last READFILE was entered from a display station that has since been released, SRJE simulates an EOF statement.

You can enter the EOF statement only as reader input. The format of the EOF statement is:

```
.. EOF
(.. E)
```

WTO

The WTO (write-to-operator) utility control statement causes a message to be sent from the reader to the SRJE console. The WTO statement is treated as a comment, except it is logged at the SRJE console. You can enter the WTO statement only as reader input.

The format of the WTO statement is:

```
.. WTO 'any text'
```

The total length of the WTO statement cannot exceed 80 characters.

If the text contains blanks or commas, the entire text must be enclosed in apostrophes ('). If the text is enclosed in apostrophes, any apostrophe within the text itself must be entered as two consecutive apostrophes. For example, the message *it's* would be entered as *'it''s'*.

END

The END utility control statement, when entered through SRJE console or reader input while in unattended mode, causes SRJE to request normal termination from the host system. If you enter the END statement as reader input while in attended mode, SRJE logs the END to the history file and treats it as a comment.

The format of the END statement is:

```
.. END [DELAY–xx]
```

DELAY: Specifies the amount of time, in minutes, that SRJE is to remain active after the END statement is processed. The value specified can be any decimal number from 0 to 99, with the default being 0. The delay time is reset to the entered value whenever print or punch data is received from the host system during the delay interval. The initial interval timing begins after the END statement is detected in the data. The timer resets to the specified value whenever punch or print data is received from the host during the interval; the timer begins interval timing when the data has been completely received.

CANCEL

The CANCEL utility control statement immediately terminates the SRJE utility or one of its functions. You can enter the CANCEL statement as SRJE console input only.

The format of the CANCEL statement is:

```
.. CANCEL [ { RD1 }  ]
(.. C)    [ { RD2 }  ]
          [ { RD3 }  ]
          [ { PR1 }  ]
          [ { PR2 }  ]
          [ { PR3 }  ]
          [ { PU1 }  ]
          [ { PU2 }  ]
          [ { PU3 }  ]
```

RD1, RD2, RD3: Terminates a reader task.

PR1, PR2, PR3: Terminates a print task.

PU1, PU2, PU3: Terminates a punch task.

blank: Terminates the SRJE utility.

FORMS

The FORMS utility control statement allows the operator to change forms and carriage information for SRJE using the forms control table. The operator can enter this utility control statement only when the printer task is active and output is being directed to the printer (not spooled). Refer to *Print Output* under *SRJE Output* in this chapter for more information on forms processing.

The format of the FORMS statement is:

```
. . FORMS NAME-name

(. . F)
```

NAME: Specifies the host forms name to look for in the forms control table created by $DCFUP.

LIBRARY

The LIBRARY statement specifies the library to be searched for source and procedure members. The library specified remains in effect until SRJE finishes reading the file or library member that contains the LIBRARY statement, until another LIBRARY statement is encountered, or until the reader task is terminated.

Note: For nested READFILE statements, the scope of a LIBRARY statement includes the file or member in which it is contained, plus any further nested files or members. When a return is made from a nested file or member, the library that was active when the nesting began becomes the active library.

The format of the LIBRARY statement is:

```
. . LIBRARY NAME– {name}
                  {0   }

(. . L)
```

NAME: Specifies the name of the library to be searched for members specified in a READFILE statement. If you enter 0, the system library is used.

DIAGNOSTIC/DEBUGGING AIDS FOR SRJE

The System/34 provides an SDLC trace facility that can be used to diagnose problems that might occur during SRJE operation. The trace facility is an IBM-supplied service procedure that allows you to log a history of occurrences as they happen within the system. The function is evoked by entering the TRACE command from the system console. When the TRACE display appears, answer Y to the prompt that asks whether you want to trace communications related events. The default SVC, plus '02' and '1D' are helpful when debugging SRJE problems. The next display prompts you for the line you want to trace. Answer Y for the line on which SRJE is running. You are also prompted for the disk logging option, which allows you to log the trace information to disk. Normally, trace information is stored in a system table that can hold 512 16-byte entries. Once the system table has filled, it wraps around and begins logging the new information over the old.

The SDLC trace facility remains active until it is reset by specifying N to the TRACE communications related events prompt for the line or until an IPL is performed. The system table or the disk stores all SDLC commands and responses during the tracing session. Refer to the *Data Areas and Diagnostic Aids Manual* for additional information on the SDLC trace.

HOST CONSIDERATIONS

When the System/34 is used in an SDLC environment with the SRJE utility, the host system must be aware of certain considerations. The following paragraphs describe some of these considerations.

The host system must not send unformatted system service messages in response to the SRJE LOGON command.

For a RES system, the TERMINAL macro defines the terminal characteristics. (See *OS/VS1 RES System Programmer's Guide*, GC28-6878.) The following parameters apply to the System/34:

TERMID = n: Specifies the terminal identification number. This value is used in the DATA parameter within the SRJE LOGON command to identify the terminal to the RES system.

TDESCR = (w,t,d,f): Describes the terminal characteristics, as follows:

w – Specifies the printer width. Select option 3, 132-characters, for the System/34.

t – Terminal type. Selection option 8, SNA, for the System/34.

d – Data transmission format. Select option 5.

f – Work station feature. Normal selection for the System/34 is option 3, console support and transparency.

RDRS = n: The number of reader tasks that can be active at the terminal. Select 1, 2, or 3.

PTRS = n: The number of print tasks that can be active at the terminal. Select 1, 2, or 3.

PCHS = n: The number of punch tasks that can be active at the terminal. Select 1, 2, or 3.

BUFXSIZ = n: Specifies the maximum size of expanded buffers. Specify 256.

SESSLIM = n: Specifies the maximum number of sessions that can be initiated for the work station. Specify 1 to 10.

SRJE supports the VTAM IDBLK parameter as hex '00E'.

On a JES2 system, see *System Programmer's Library; Network Job Entry Facility for JES2,* SC23-0003. The following parameters in the RMTnnn macro apply to the System/34:

LUTYPE1: Specifies an SNA terminal.

BUFSIZE = 256: Specifies the buffer size as being 256 bytes.

NOCMPCT: Specifies that compaction is not to be used with this remote terminal.

COMP: Specifies that blank compression and multiple duplicate character compression may be used with this remote terminal.

CONSOLE: Specifies that the terminal has a device separate from the printer that can be used to receive console messages.

NUMPR = 1-3: Specifies the number of logical printers at the terminal.

NUMPU = 1-3: Specifies the number of logical punches at the terminal.

NUMRD = 1-3: Specifies the number of logical readers at the terminal.

SETUPHDR: Specifies that forms information is to be sent in the data stream. This requires that the host be generated with PDIRs.

The Rnnn.PRm macro is used to define the remote printer. The nnn corresponds to the terminal number in the RMTnnn macro and the m is the printer number, which is 1, 2, or 3 for the System/34. The following parameters apply to the System/34:

AUTO: Specifies that forms control will be handled automatically.

CCTL: Specifies that carriage control characters are to be sent in the data stream directed to the printer.

PRWIDTH = 132: Specifies the printer width as being 132 characters.

On a POWER system, the PRMT macro is used to define the terminal. (See *DOS/VSE POWER/VSE Installation and Operator's Guide, GC33-5403.)* The following parameters are applicable to the System/34:

TYPE = LUT1: Defines the terminal type.

CONSOLE = YES: Specifies that the terminal supports a separate device for the logging of console messages.

SNA Profile Used for SRJE

- The System/34 SNA conforms to the following definitions:
  - PU Type 2
  - FM Profile 3
  - TS Profile 3

- System/34 is similar to the 3770 Data Communications System in batch mode. The values for the 3770 (as described in the appropriate host system manuals) may be used for system generation, with the following exceptions:
  - The System/34 device identifier for the XID command is hex 00E
  - System/34 requires that the host system be generated with PDIRs supported

- System/34 SRJE buffer size is 256 bytes for a request/response unit (RU).

- System/34 SRJE has the following tasks available for communications:
  - 1, 2, or 3 reader tasks
  - 1, 2, or 3 punch tasks
  - 1, 2, or 3 printer tasks
  - One console input task
  - One console output task

- The maximum pacing used when generating NCP determines the minimum number of System/34 receive data buffers. For example:

Pacing Count	**Receive Buffers**
2,1	5
7,1	15
x,1	2x + 1

Note: Pacing count 2, 7, or x is the number of data buffers the System/34 can receive before it sends a pacing response. The 1 indicates that the first buffer (of the 2, 7, or x) of data from the host should request the pacing response from the System/34.

SNA Bind Parameters

The following operands are applicable to SRJE in the BIND construction:

LOGMODE: Specifies the name to be used as a key for the session parameters. This is the same name used in the LOGON command for the LOGMODE parameter.

FMPROF: Specifies the function management profile to be used. Specify hex 03 for SRJE.

TSPROF: Specifies the transmission service profile. Specify hex 03 for SRJE.

PRIPROT: Specifies the primary logical unit profiles. Hex A3 specifies multiple element chains, definite response to chains, blank compression, and send bracket indicators.

SECPROT: Specifies the secondary logical unit protocols. Hex A3 specifies the same values as for PRIPROT.

COMPROT: Specifies the common logical unit protocols. Hex 7080 specifies that function management headers are used, brackets are used, bracket termination rule 1 is used, and alternate code is used.

RUSIZES: Specifies the maximum request unit length. Specify hex 8585 to define 256 bytes.

PSERVIC: Specifies the logical unit presentation services profile. Consists of 12 bytes defined as follows:

Byte	Definition
Byte 1	Specify hex 01 for logical unit presentation services profile 1.
Byte 2	Specifies the function management subset. Specify hex 10 for function management header subset 1.
Bytes 3 and 4	Specify the primary function management header subset flags. Specify hex 2000 to send PDIRs.
Byte 5	Specifies the primary data stream subset flags. Hex F1 specifies interactive data stream, horizontal format data stream, vertical tab data stream, vertical select data stream, and transparency data stream.
Byte 6	Not used. Specify hex 00.
Byte 7	Specifies the primary media flags. Hex C0 specifies document output and card format.
Bytes 8 and 9	Specifies the secondary function management header subset flags. Specify hex 0000.
Byte 10	Specifies secondary data stream subset flags. Specify hex 01 for a transparency data stream.
Byte 11	Not used. Specify hex 00.
Byte 12	Specifies the secondary media flags. Specify hex 40 for card format.

SRJE SESSION EXAMPLE

The following is an example of an SRJE session. The session is started from the System/34 and the host system JCL is read from a procedure in #LIBRARY. When the JCL is read, a job is executed at the host system, and printer output is returned to the System/34. The process is then repeated with the print output being directed to the System/34 disk. The example uses OS/VS1 RES as the host system; however, the operating characteristics for a different host would be similar.

To start the SRJE session, initiate the SRJE utility by entering:

```
SRJE ,,,YES,1
```

at a command display station. The command specifies that the default initialization display, #SR@ID is to be used, attended mode is desired, tasks will not have user priority, the request discontact network services request is to be sent, communications is to be on line 1, and autocall is not requested. Once the utility has been loaded, the initialization display appears.

Change the parameters as follows:

```
   RD1NAME..CMDFILE2
   LOGON....Y
   RD1TYPE..P
   RD1CMD...Y
   RD1DATE..
   ENTER LOGON INFORMATION BELOW:
LOGON APPLID(RTAM) LOGMODE(S34SRJE1) DATA('A0 T(5) PROC(FSPROC)')
   FCTNAME..SRJEFCT
```

After you enter the preceding values, the initialization screen appears as follows:

```
SRJE OPTION MENU--PRESS ENTER TO CONTINUE

ENTER CONFIGURATION INFORMATION BELOW:

PLUNAME..            LOGON.... Y         FSN..... 001
FID..... PR1         OUTPUT... P1        SPACE.... 0060      DEVID.... SYSTEM
FID..... PU1         OUTPUT... N         SPACE.... 0060
RD1NAME.. CMDFILE2   RD1TYPE.. P         RD1CMD... Y         RD1DATE..
RD1LIBR..            RD1XPC... N         RD1DEL... Y

ENTER LOGON INFORMATION BELOW:

LOGON APPLID(RTAM) LOGMODE(S34SRJE1) DATA('A0 T(5) PROC(FSPROC)')

ENTER CARRIAGE AND FORMS INFORMATION BELOW:

L.... 066    B.... 066    F.... STD.  FCTNAME.. SRJEFCT
1.... 001    2.... 000    3.... 000   4.... 000    5.... 000     6.... 000
7.... 000    8.... 000    9.... 000   10... 000    11... 000     12... 000


************          SYSTEM OPERATOR MESSAGE BELOW          ************
```

CMDFILE2 contains the JCL read by the host system. The contents of CMDFILE2 are:

```
//CMDFILE2  JOB  parameters
//          EXEC PGM=IEBPTPCH
//SYSPRINT  DD   SYSOUT=1
//SYSUT2    DD   SYSOUT=A
//SYSUT1    DD   *
.. READFILE   NAME-DATAFIL2,TYPE-P,CMD-N,XPC-N
/*
//SYSIN     DD   *
//PRINT     MAXFLDS=1
//RECORD    FIELD=(80)
/*
```

After you modify the initialization screen, the system displays the following messages:

```
SRJE INITIALIZATION COMPLETED
SYS-4423 RD1 ACTIVATED
//
.SF ,RD1
.. R NAME-CMDFILE2,TYPE-P,CMD-Y,XPC-N
.SF ,PR1
.. READFILE NAME-DATAFIL2,TYPE-P,CMD-N,XPC-N
```

After the host has read CMDFILE2 and DATAFIL2, the following messages are display:

```
SYS-4461 RD1 FINISHED READING FILE/MEMBER DATAFIL2
SYS-4461 RD1 FINISHED READING FILE/MEMBER CMDFILE2
```

The system console keyboard is now ready for reader input.

Enter:

```
.. EOF
```

The system responds with the following message:

```
SYS-4424 RD1 TERMINATED NORMALLY
```

After the program executes at the host, the System/34 receives and prints the output.

To direct printer output to the disk, press the Attn key. This displays the CNI screen. When the CNI screen appears, enter the following command:

```
.. M FID-PR1,OUTPUT-D,SPACE-40
```

Then, press the Attn key again and, when the CNI screen appears, press command key 1. This displays an SRJE status screen. Press the Enter key again. An RD1 screen is displayed with the following message:

```
SYS-4423 RD1 ACTIVATED
```

```
SRJE STATUS MENU--PRESS ENTER TO CONTINUE


PLUNAME..           LOGON.... Y         FSN..... 001
FID..... PR1        OUTPUT... D         SPACE.... 0040      DEVID.... SYSTEM
FID..... PU1        OUTPUT... N         SPACE.... 0060


LOGON INFORMATION BELOW:

LOGON APPLID(RTAM) LOGMODE(S34SRJE1) DATA('A0 T(5) PROC(FSPROC)')

CARRIAGE AND FORMS INFORMATION

L.... 066    B.... 066    F.... STD.  FCTNAME.. SRJEFCT
1.... 001    2.... 000    3.... 000   4.... 000    5.... 000     6.... 000
7.... 000    8.... 000    9.... 000   10... 000    11... 000     12... 000


**********          SYSTEM OPERATOR MESSAGE BELOW          **********
```

Enter:

```
.. READFILE NAME-CMDFILE2,TYPE-P,CMD-Y
```

The following command from CMDFILE2 and system messages appear:

```
.. READFILE NAME-DATAFIL2,TYPE-P,CMD-N,XPC-N
SYS-4461 RD1 FINISHED READING FILE/MEMBER DATAFIL2
SYS-4461 RD1 FINISHED READING FILE/MEMBER CMDFILE2
```

Enter:

```
.. EOF
```

The system responds with the following message:

```
SYS-4424 RD1 TERMINATED NORMALLY
```

Once again, the program is executed at the host. However, this time the output is written to the System/34 disk. In this case the system issues the following messages, which indicate that a disk file is being created and processed:

```
SYS-4459 PR1--FILE A001L068 BEING CREATED
SYS-4465 PR1--FILE A001L068 CREATION COMPLETED
 IEF868I PR1 WTR WAITING FOR WORK
```

Press the Attn key and then enter:

```
.. END
```

Appendix A. ASCII and EBCDIC

The coded character sets for ASCII and EBCDIC are shown in the following charts. The transmission control characters recognized by System/34 are listed in Appendix B.

ASCII

		Main Storage Bit Positions 0, 1, 2, 3															
Main Storage Bit Positions		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
4, 5, 6, 7	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	SP	0	@	P	`	p								
0001	1	SOH	DC1	!	1	A	Q	a	q								
0010	2	STX	DC2	"	2	B	R	b	r								
0011	3	ETX	DC3	#	3	C	S	c	s								
0100	4	EOT	DC4	$	4	D	T	d	t								
0101	5	ENQ	NAK	%	5	E	U	e	u								
0110	6	ACK	SYN	&	6	F	V	f	v								
0111	7	BEL	ETB	'	7	G	W	g	w								
1000	8	BS	CAN	(	8	H	X	h	x								
1001	9	HT	EM	)	9	I	Y	i	y								
1010	A	LF	SUB	*	:	J	Z	j	z								
1011	B	VT	ESC	+	;	K	[	k	{								
1100	C	FF	FS	,	<	L	\	l	|							
1101	D	CR	GS	-	=	M	]	m	}								
1110	E	SO	RS	.	>	N	¬	n	~								
1111	F	SI	US	/	?	O	_	o	DEL								

Figure A-1. ASC11 Coded Character Sets

EBCDIC

Main Storage Bit Positions 4, 5, 6, 7	Hex	Main Storage Bit Positions 0, 1, 2, 3															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	DS		SP	&	-						{	}	\	0
0001	1	SOH	DC1	SOS		RSP		/		a	j	~		A	J	NSP	1
0010	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
0011	3	ETX	DC3	WUS	IR					c	l	t		C	L	T	3
0100	4	SEL	ENP/RES	INP/BYP	PP					d	m	u		D	M	U	4
0101	5	HT	NL	LF	RS					e	n	v		E	N	V	5
0110	6		BS	ETB	NBS					f	o	w		F	O	W	6
0111	7	DEL	POC	ESC	EOT					g	p	x		G	P	X	7
1000	8	GE	CAN		SBS					h	q	y		H	Q	Y	8
1001	9	SPS	EM		IT				`	i	r	z		I	R	Z	9
1010	A	RPT	UBS	SM/SW	RFF	¢	!	¦	:					SHY			
1011	B	VT	CU1	FMT	CU3	.	$	,	#								
1100	C	FF	IFS		DC4	<	*	%	@					⌐		⌐	
1101	D	CR	IGS	ENQ	NAK	(	)	_	'								
1110	E	SO	IRS	ACK		+	;	>	=					⌐			
1111	F	SI	ITB/IUS	BEL	SUB	|	¬	?	"								EO

Figure A-2. EBCDIC Coded Character Sets

Duplicate Assignment

Appendix B. Transmission Control Characters for BSC

The following characters and character sequences are recognized by System/34 BSC. The hexadecimal configuration is shown below each character. For detailed information on transmission control characters (also called line control characters), see *General Information—Binary Synchronous Communications*.

Name	Mnemonic	ASCII	EBCDIC
Start of heading	SOH	SOH 01	SOH 01
Start of text	STX	STX 02	STX 02
End of transmission block	ETB	ETB 17	ETB 26
End of text	ETX	ETX 03	ETX 03
End of transmission	EOT	EOT 04	EOT 37
Enquiry	ENQ	ENQ 05	ENQ 2D
Negative acknowledge	NAK	NAK 15	NAK 3D
Synchronous idle	SYN	SYN 16	SYN 32
Data link escape	DLE	DLE 10	DLE 10
Intermediate block character	ITB	US 1F	IUS 1F
Even acknowledge	ACK0	DLE 0 1030	DLE (70) 1070
Odd acknowledge	ACK1	DLE 1 1031	DLE/ 1061
Wait before transmit-positive acknowledge	WACK	DLE; 103B	DLE, 106B
Mandatory disconnect	DISC	DLE EOT 1004	DLE EOT 1037
Inter group separator	IGS	GS 1D	IGS 1D
Inter record separator	IRS		IRS 1E
Reverse interrupt	RVI	DLE< 103C	DLE@ 107C
Temporary text delay	TTD	STX ENQ 0205	STX ENQ 022D
Transparent start of text	XSTX		DLE STX 1002
Transparent intermediate block	XITB		DLE IUS 101F
Transparent end of text	XETX		DLE ETX 1003
Transparent end of transmission block	XETB		DLE ETB 1026
Transparent synchronous idle	XSYN		DLE SYN 1032
Transparent block control	XENQ		DLE ENQ 102D
Transparent TTD	XTTD		DLE STX DLE ENQ 10 02 10 2D
Data DLE in transparent mode	XDLE		DLE DLE 10 10

Appendix C. BSC Polling and Addressing Characters for Tributary Stations

Polling and addressing characters must be used together in certain pairs: that is, once a polling character is selected, the complementary addressing character is determined; once an addressing character is selected, the complementary polling character is determined.

The pairs of valid polling and addressing characters for both EBCDIC and ASCII are as follows:

EBCDIC

BB	C2C2	SS	E2E2
CC	C3C3	TT	E3E3
DD	C4C4	UU	E4E4
EE	C5C5	VV	E5E5
FF	C6C6	WW	E6E6
GG	C7C7	XX	E7E7
HH	C8C8	YY	E8E8
II	C9C9	ZZ	E9E9
JJ	D1D1	11	F1F1
KK	D2D2	22	F2F2
LL	D3D3	33	F3F3
MM	D4D4	44	F4F4
NN	D5D5	55	F5F5
OO	D6D6	66	F6F6
PP	D7D7	77	F7F7
QQ	D8D8	88	F8F8
RR	D9D9	99	F9F9
ƀƀ[1]	4040	–	6060

[1] This is not a valid address when using RPG.

ASCII

AA	4141	aa	6161
BB	4242	bb	6262
CC	4343	cc	6363
DD	4444	dd	6464
EE	4545	ee	6565
FF	4646	ff	6666
GG	4747	gg	6767
HH	4848	hh	6868
II	4949	ii	6969
JJ	4A4A	jj	6A6A
KK	4B4B	kk	6B6B
LL	4C4C	ll	6C6C
MM	4D4D	mm	6D6D
NN	4E4E	nn	6E6E
OO	4F4F	oo	6F6F
PP	5050	pp	7070
QQ	5151	qq	7171
RR	5252	rr	7272
SS	5353	ss	7373
TT	5454	tt	7474
UU	5555	uu	7575
VV	5656	vv	7676
WW	5757	ww	7777
XX	5858	xx	7878
YY	5959	yy	7979
ZZ	5A5A	zz	7A7A

Addressing or polling characters can be specified in the DTF. In addition, they may be overridden before executing the user program.

To specify polling or addressing characters in the ADDR-nn parameter of the SETR utility control statement or the OVERRIDE command statement format, give the hex representation of one of the addressing characters. It will be duplicated by the system to provide two characters. The polling characters corresponding to the specified address characters are set automatically.

For example, ADDR-E7 is given to specify addressing characters XX. The corresponding polling characters GG are also set. ADDR-70 is given to specify the ASCII address character pp and the corresponding polling character PP.

Appendix D. System/34 Interface to BSC Line Protocol

This appendix is intended for a person who is knowledgeable about BSC line protocol. This person should have the following information to:

- Write a program for the remote location to interface with System/34 RPG II or assembler.
- Write a program for the System/34 to perform a specific sequence of line protocol.

This appendix shows RPG II/assembler binary synchronous communications (BSC) line protocols and indicates which ones are performed automatically and which ones the program controls.

Each System/34 transmission of data over the communication line is equal to BLOCK SIZE on the BSCA's file description specification for RPG II, or the BLKL operand on the $DTFB macroinstruction for assembler. The size of the data blocks received by System/34 is controlled by the transmitting station. (The asterisk in the System/34 *Software Adapter* column indicates customer coding logic.)

POINT-TO-POINT NONSWITCHED LINE

Receive

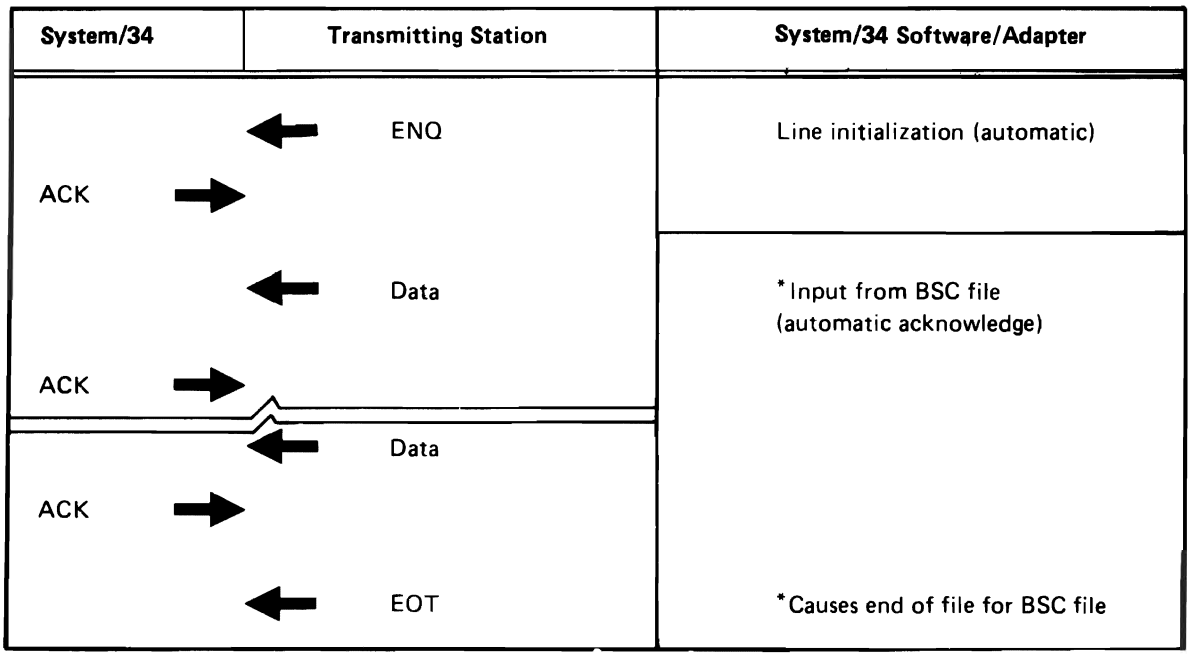


Figure D-1. Point-to-Point Nonswitched Line: Receive

Transmit

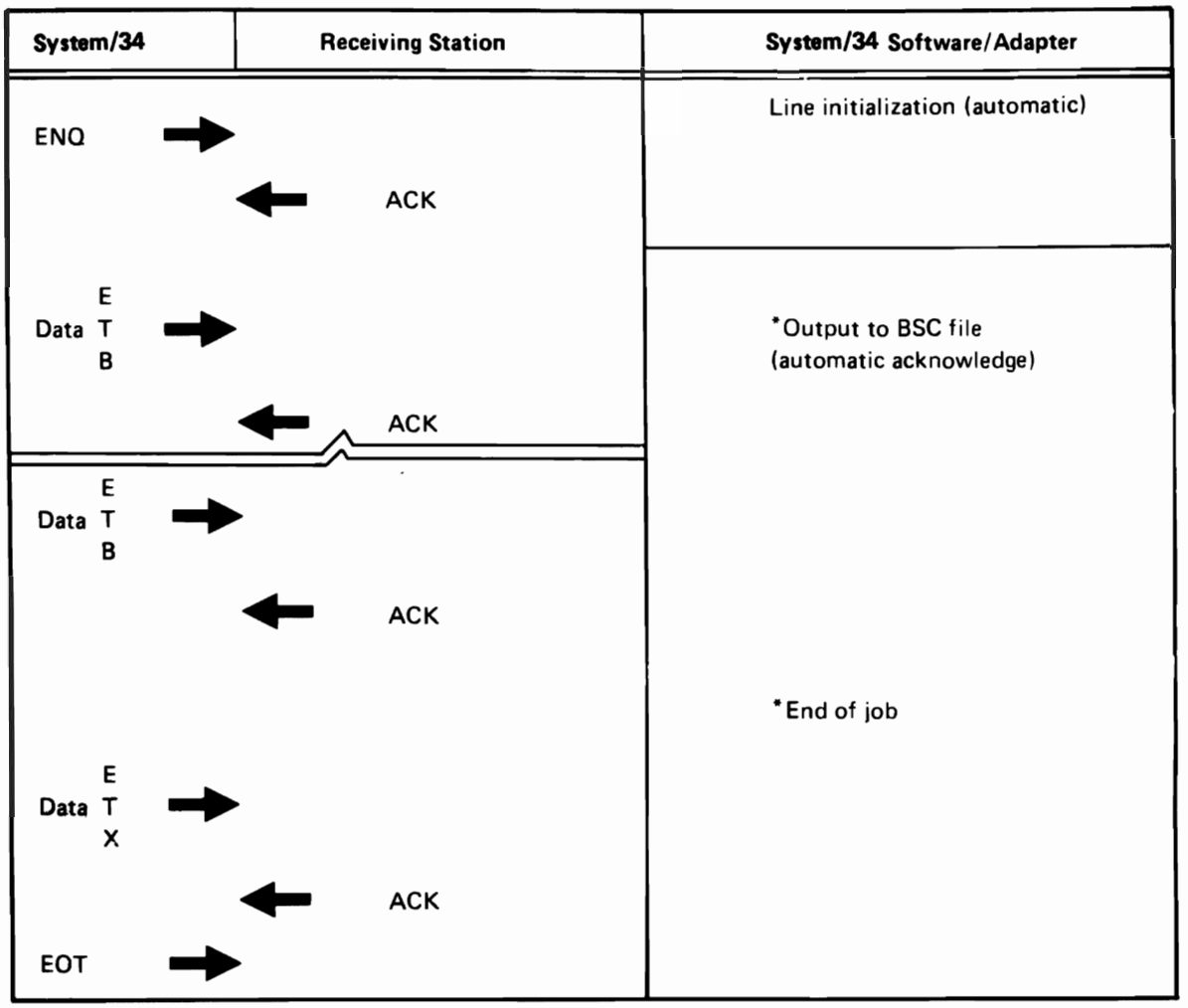


Figure D-2. Point-to-Point Nonswitched Line: Transmit

Transmit Multiple Files

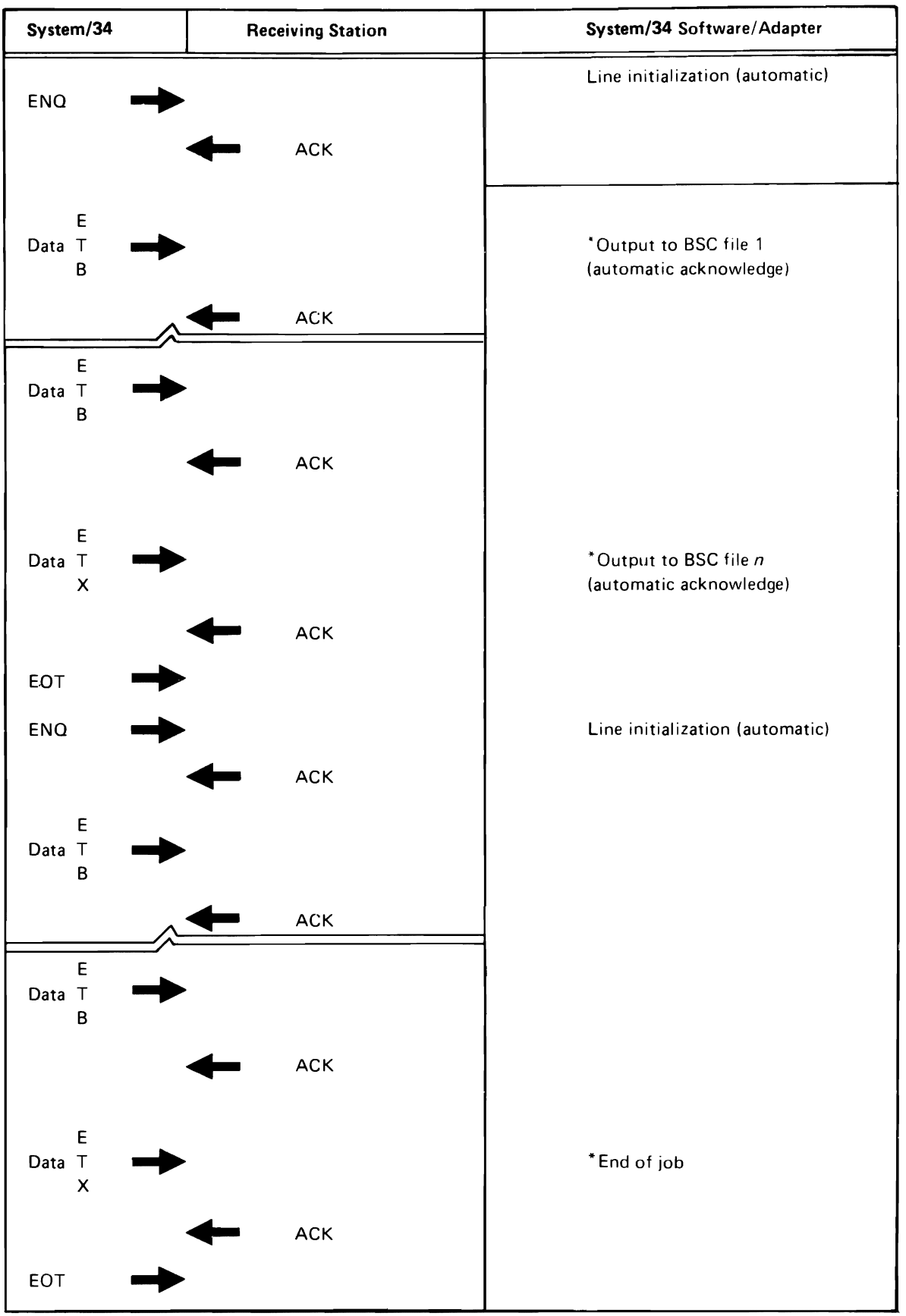


Figure D-3. Point-to-Point Nonswitched Line: Transmit Multiple Files

Transmit/Receive

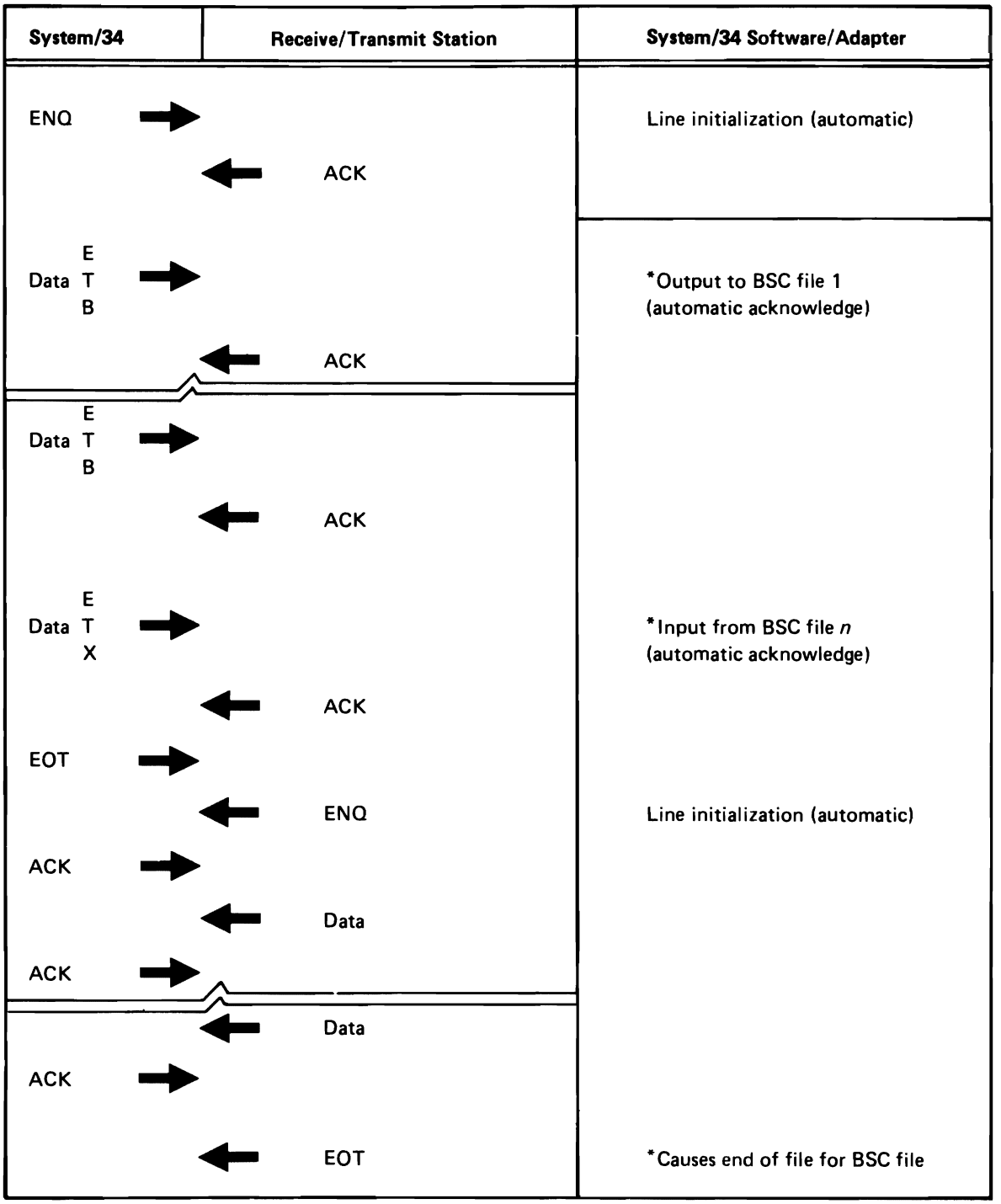


Figure D-4. Point-to-Point Nonswitched Line: Transmit/Receive

Receive/Transmit

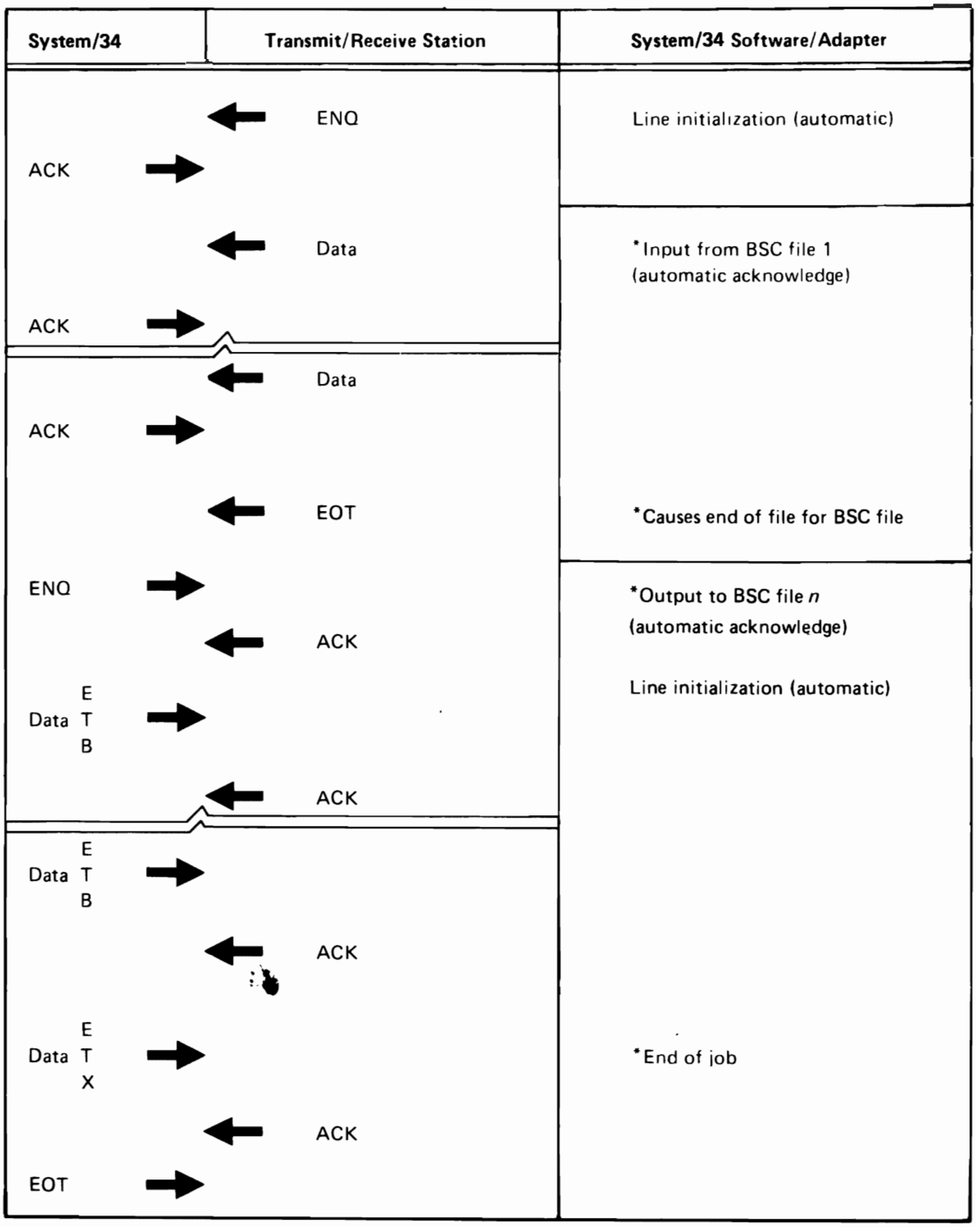


Figure D-5. Point-to-Point Nonswitched Line: Receive/Transmit

NONSWITCHED MULTIPOINT, SYSTEM/34 TRIBUTARY STATION

Receive

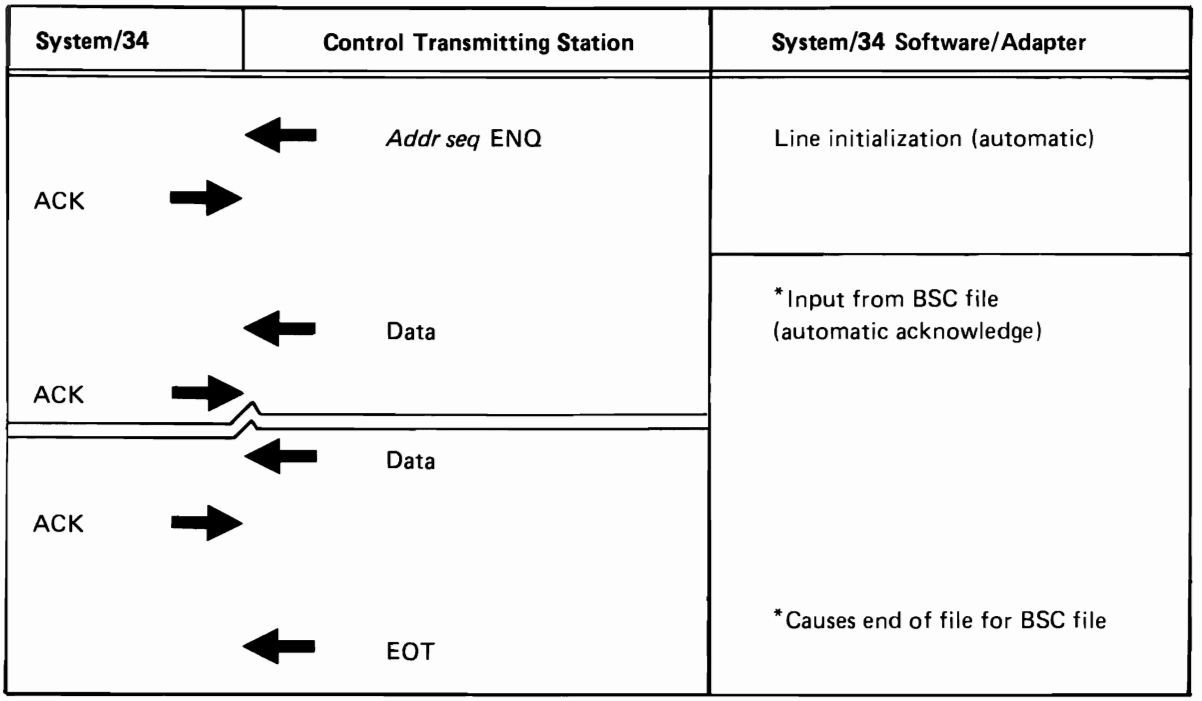


Figure D-6. Nonswitched Multipoint, System/34 Tributary Station: Receive

Transmit

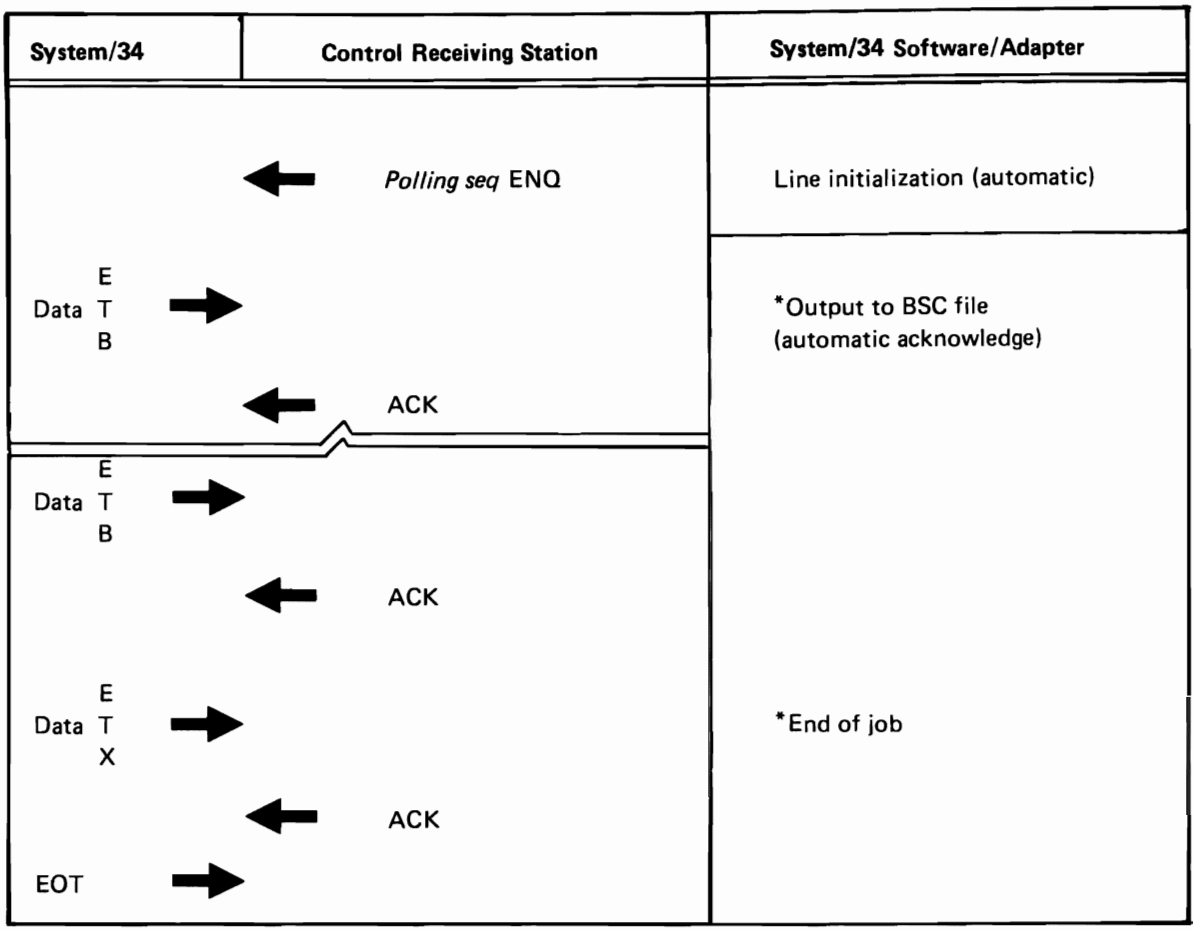


Figure D-7. Nonswitched Multipoint, System/34 Tributary Station: Transmit

Transmit, Receive

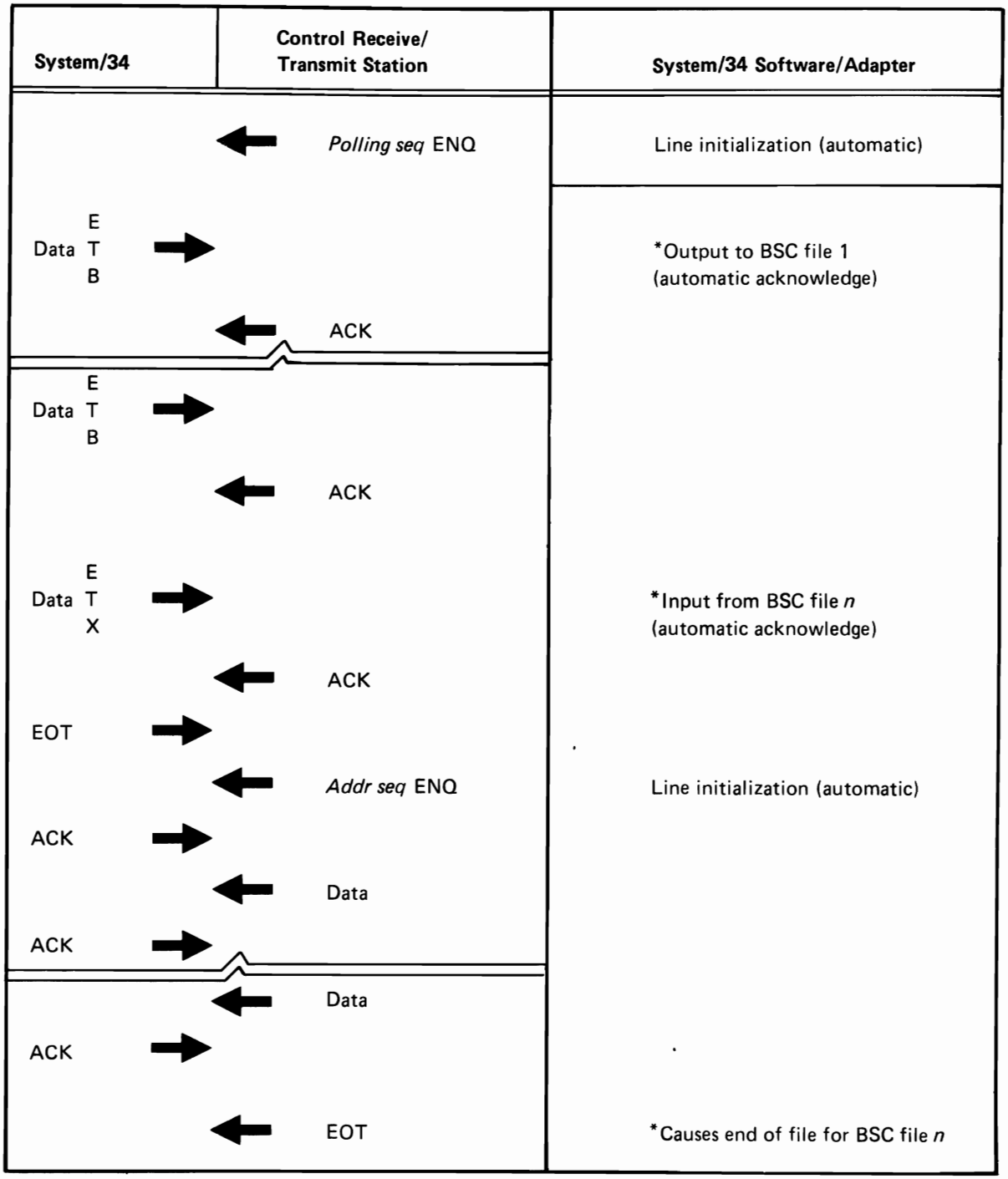


Figure D-8. Nonswitched Multipoint, System/34 Tributary Station: Transmit, Receive

Receive, Transmit

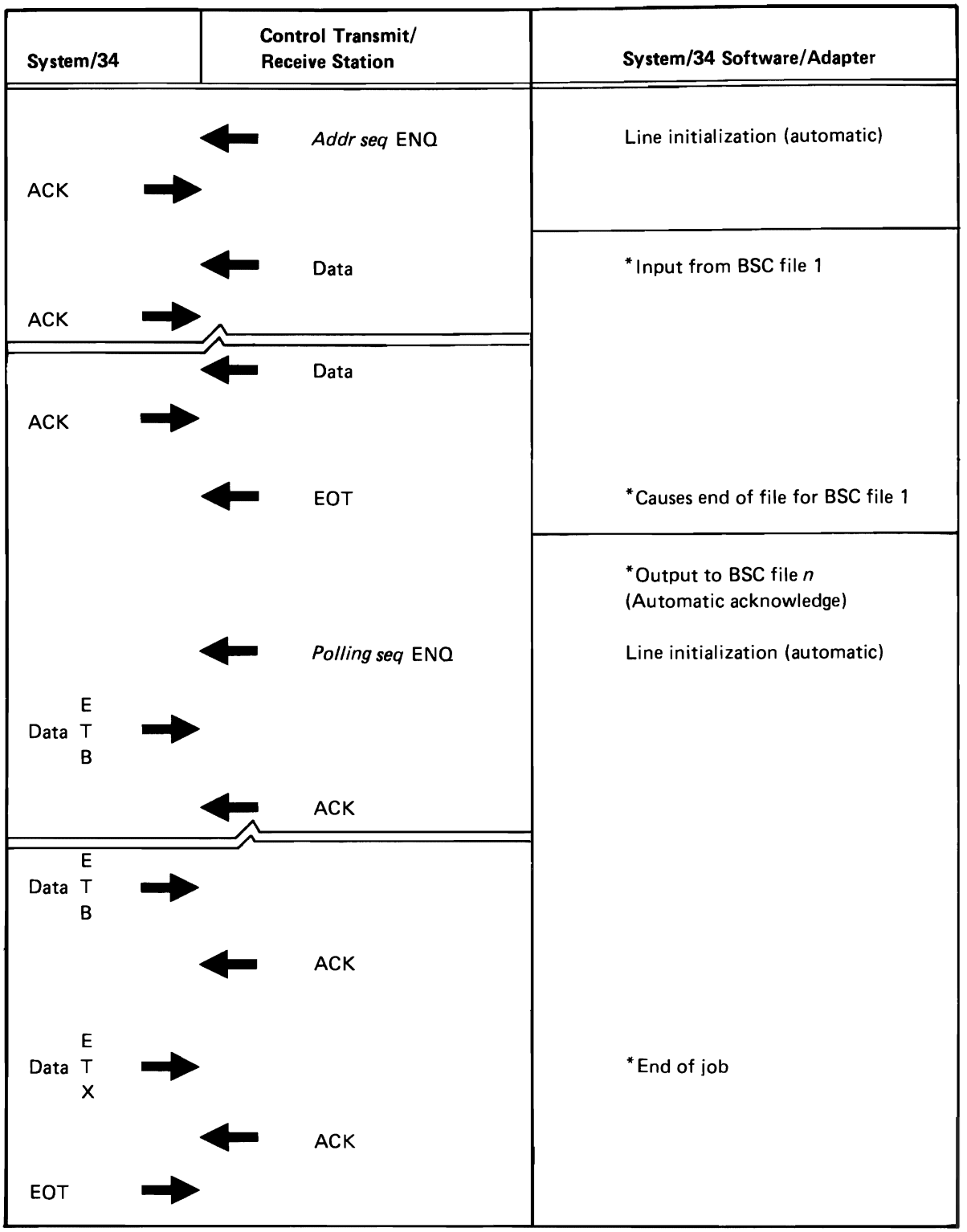


Figure D-9. Nonswitched Multipoint, System/34 Tributary Station: Receive, Transmit

POINT-TO-POINT SWITCHED LINE ID SEQ: SWITCHED LINE STATION IDENTIFICATION SEQUENCE OPTIONAL

Receive, System/34 Answer Station

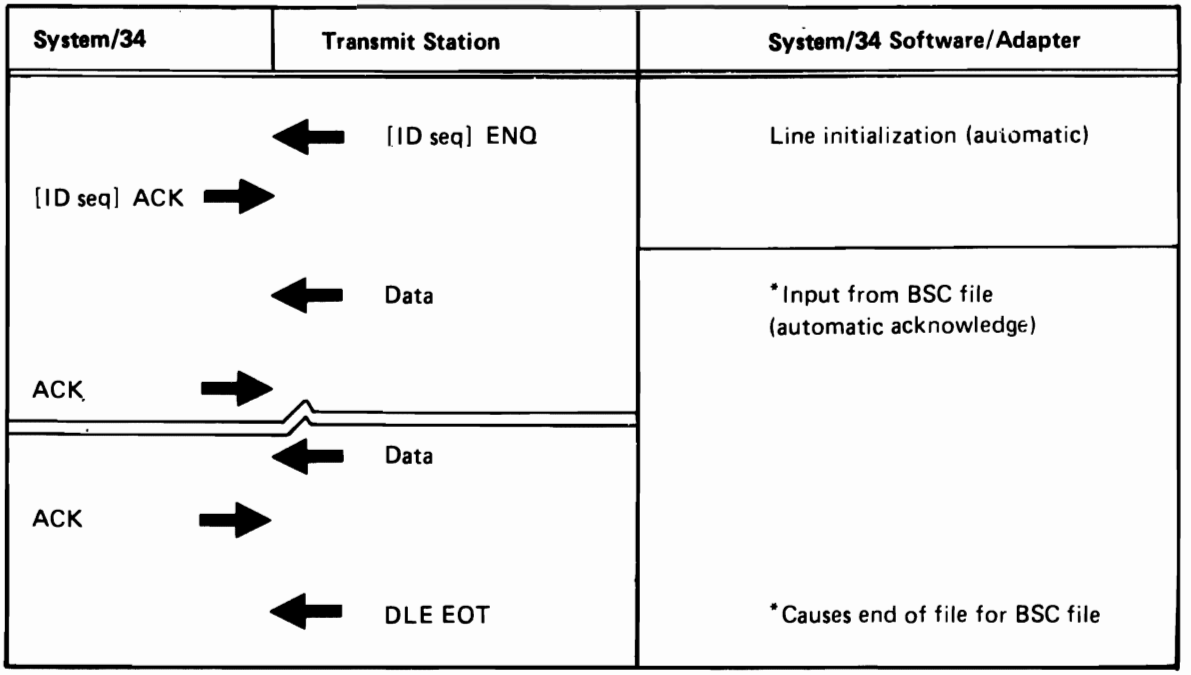


Figure D-10. Point-to-Point Switched Line ID Seq: Receive, System/34 Answer Station

Receive, System/34 Calling Station

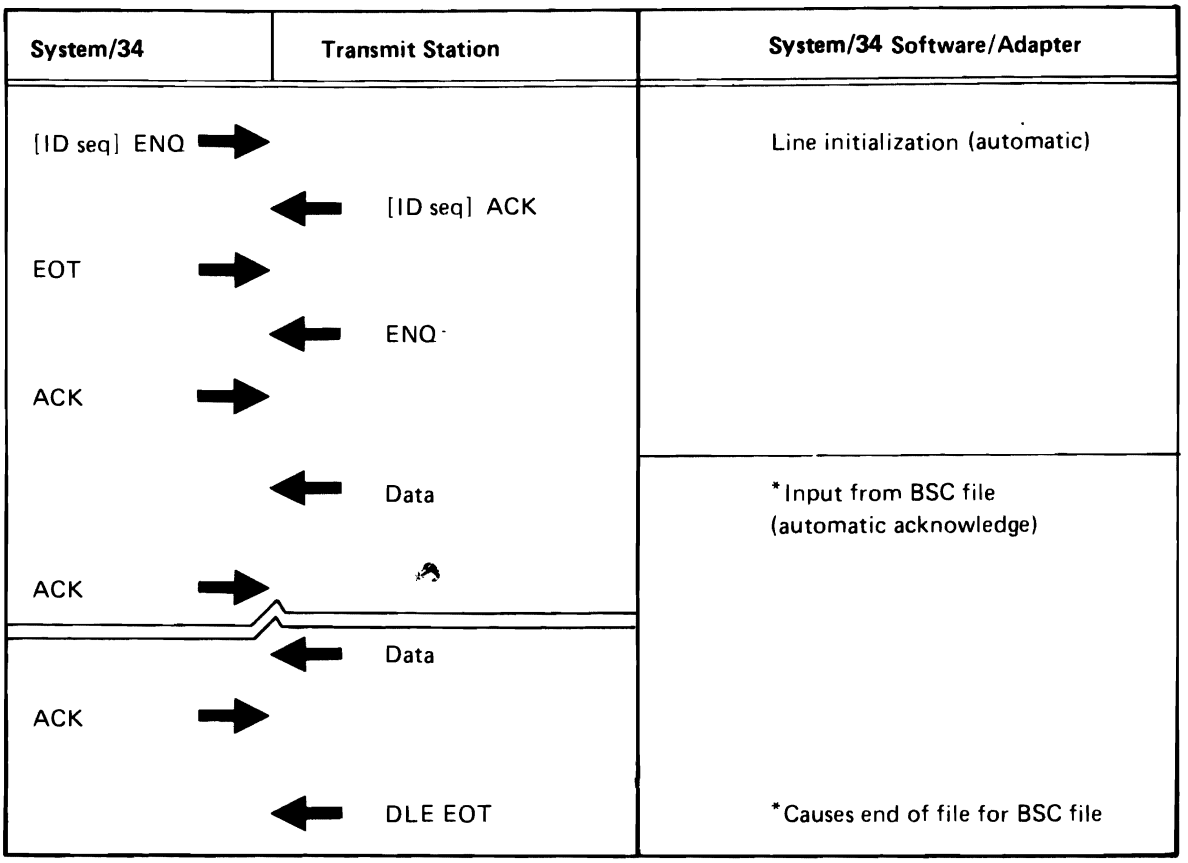


Figure D-11. Point-to-Point Switched Line ID Seq: Receive, System/34 Calling Station

Transmit, System/34 Answer Station

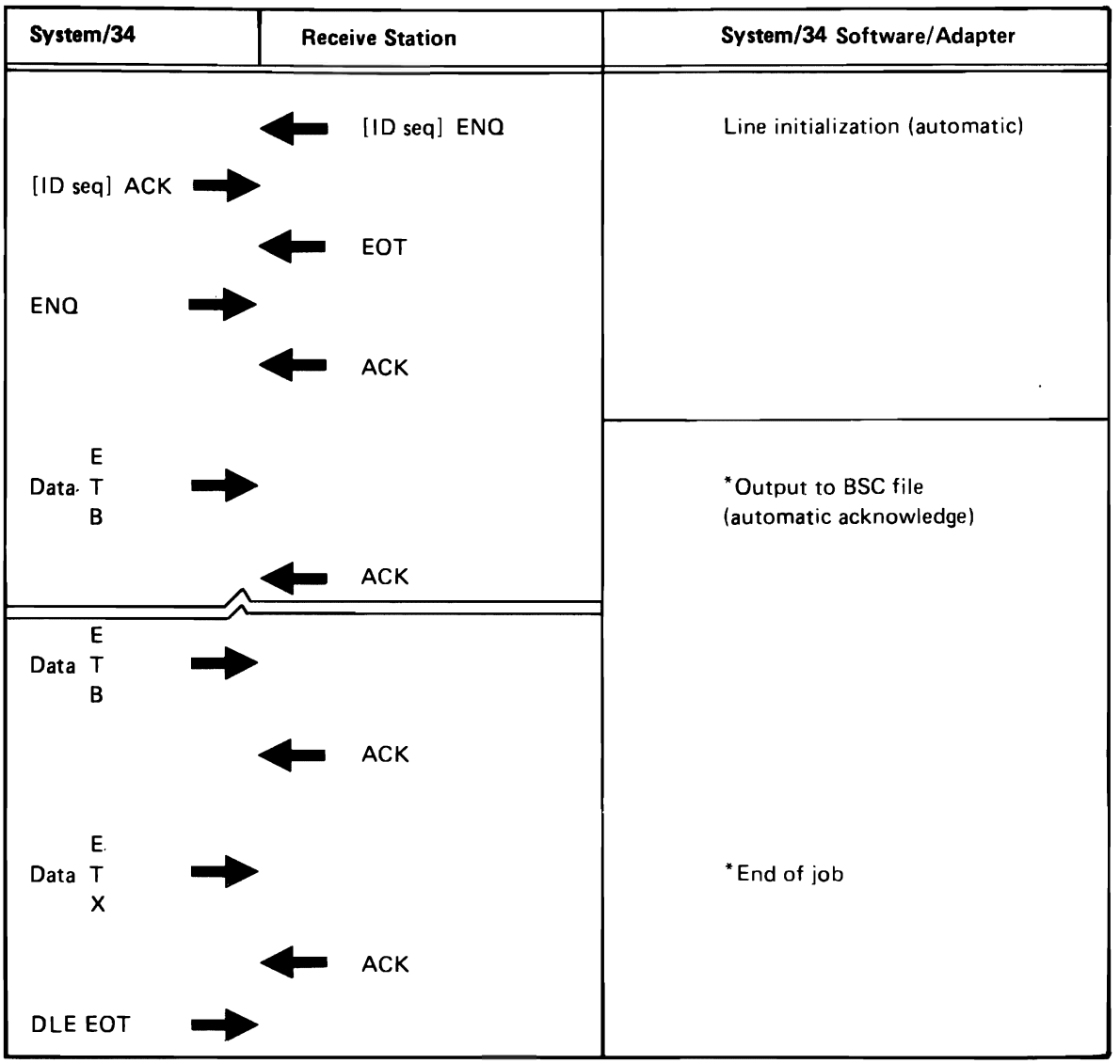


Figure D-12. Point-to-Point Switched Line ID Seq: Transmit, System/34 Answer Station

Transmit, System/34 Calling Station

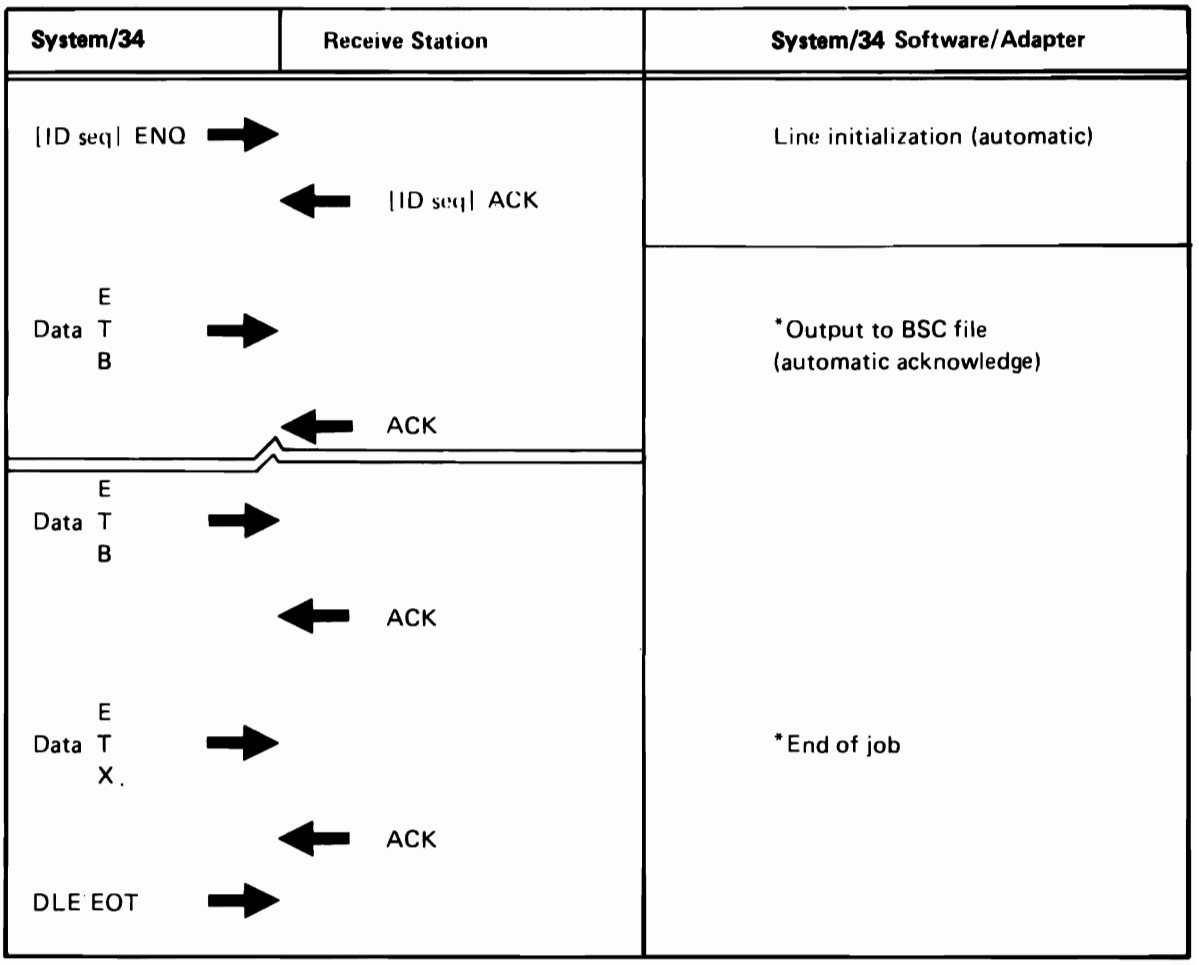


Figure D-13. Point-to-Point Switched Line ID Seq: Transmit, System/34 Calling Station

Transmit/Receive, System/34 Calling Station

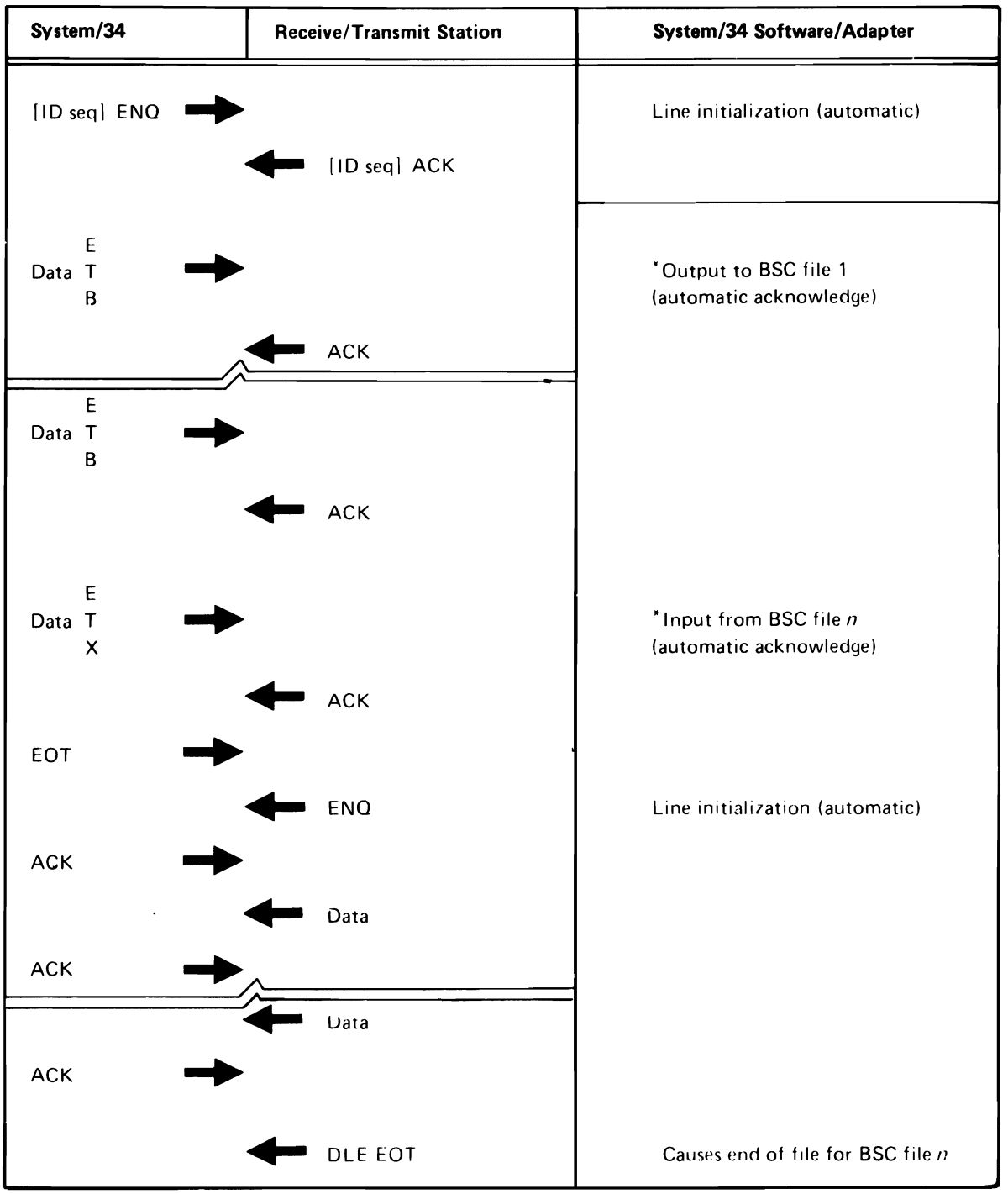


Figure D-14. Point-to-Point Switched Line ID Seq: Transmit/Receive, System/34 Calling Station

Transmit/Receive, System/34 Answer Station

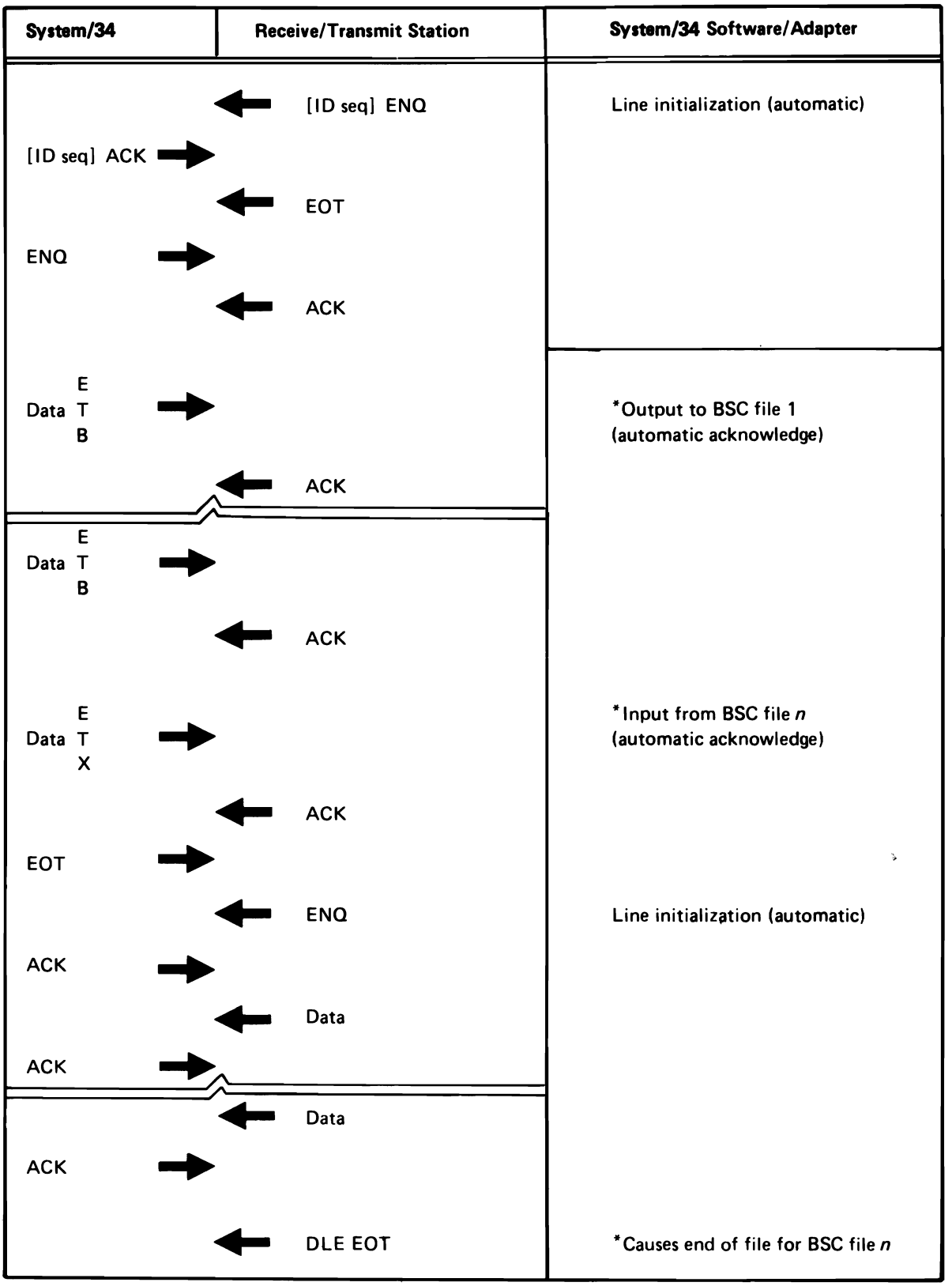


Figure D-15. Point-to-Point Switched Line ID Seq: Transmit/Receive, System/34 Answer Station

Receive/Transmit, System/34 Answer Station

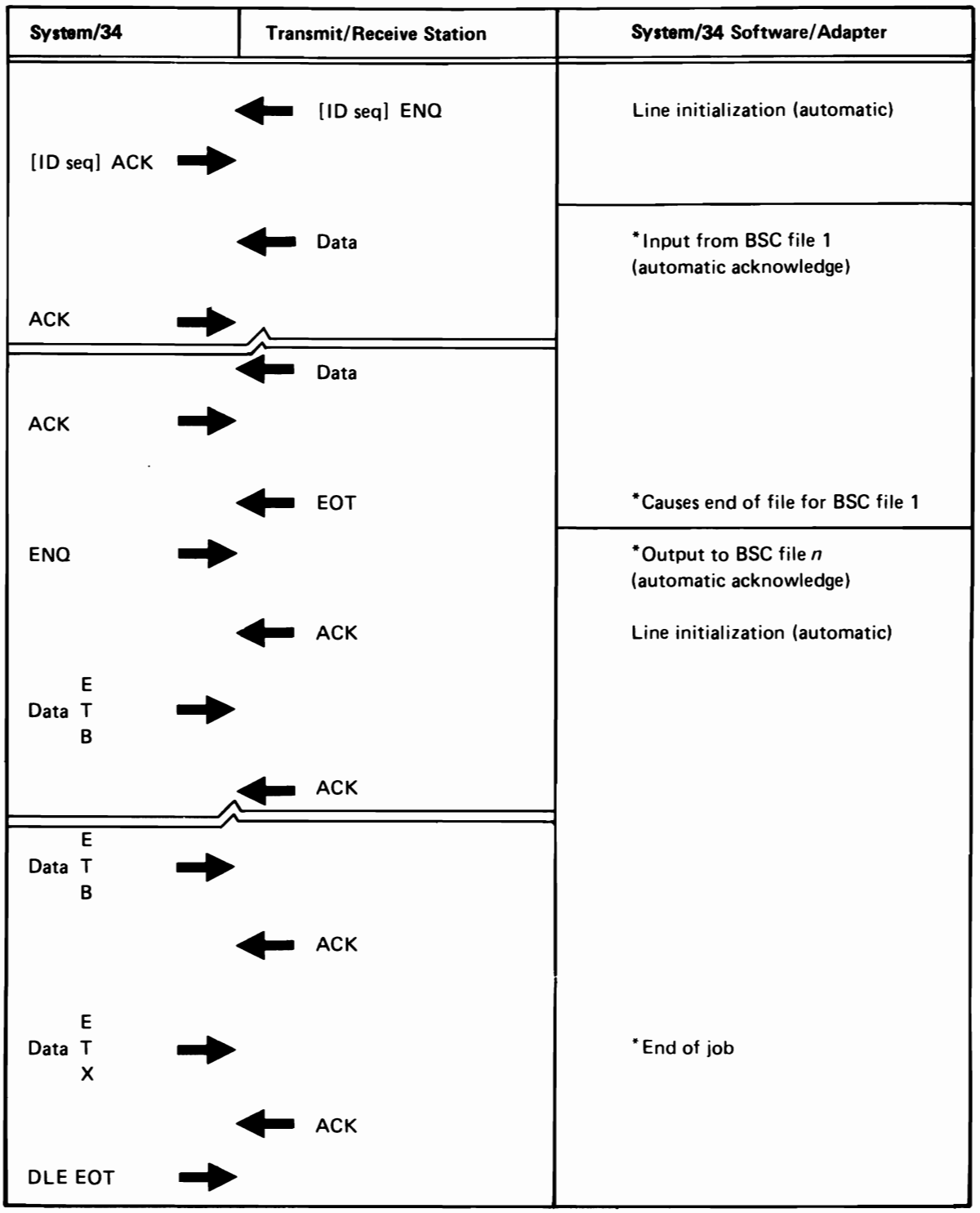


Figure D-16. Point-to-Point Switched Line ID Seq: Receive/Transmit, System/34 Answer Station

Receive/Transmit, System/34 Calling Station

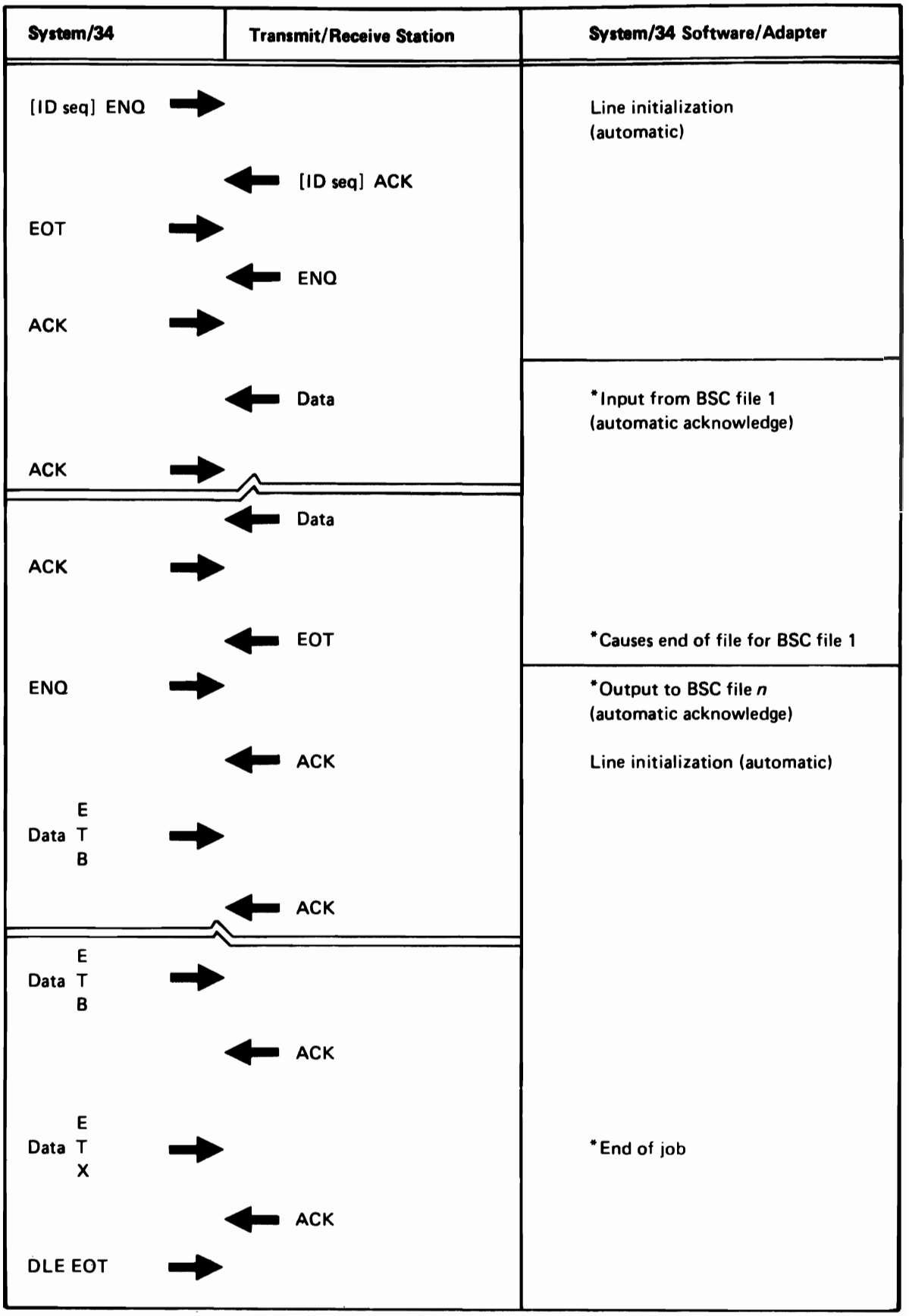


Figure D-17. Point-to-Point Switched Line ID Seq. Receive/Transmit, System/34 Calling Station

LINE CONDITION–SYSTEM/34 RESPONSES

Data Mode, System/34 Transmitting

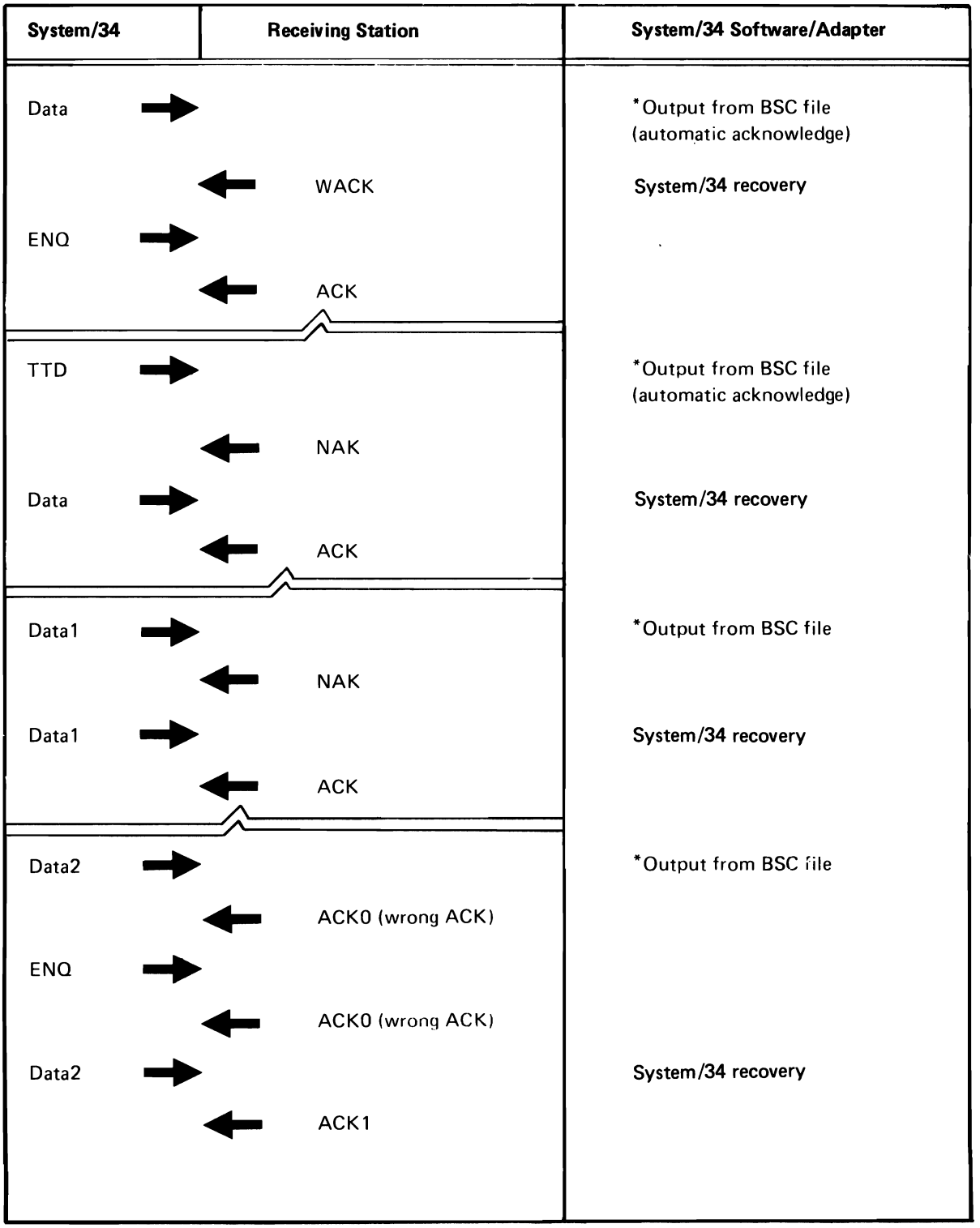


Figure D-18 (Part 1 of 2). Line Condition — System/34 Responses: Data Mode, System/34 Transmitting

Data Mode, System/34 Transmitting (continued)

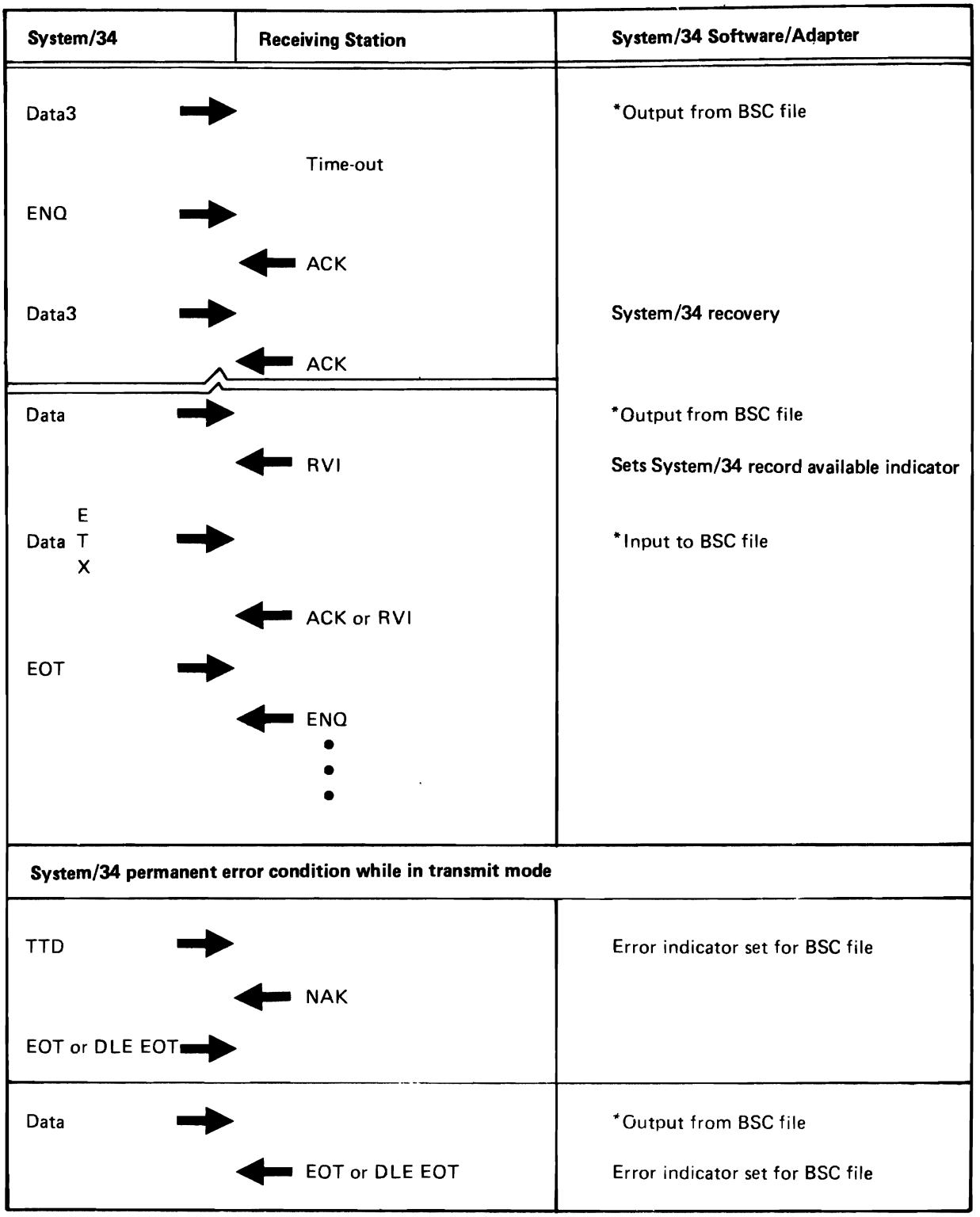


Figure D-18 (Part 2 of 2). Line Condition — System/34 Responses: Data Mode, System/34 Transmitting

Data Mode, System/34 Receiving

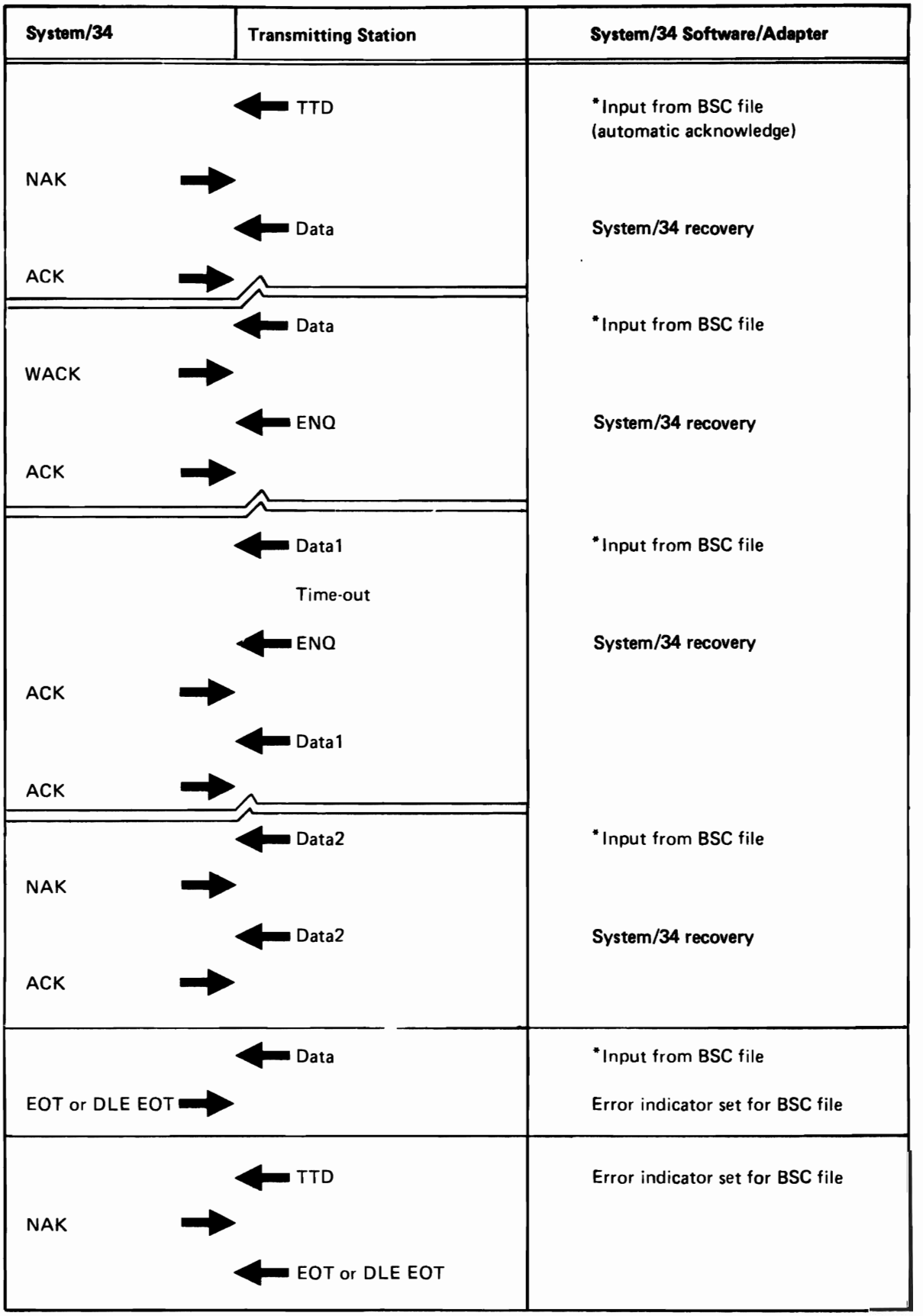


Figure D-19. Line Condition — System/34 Responses: Data Mode, System/34 Receiving

3740 MULTIPLE FILE SUPPORT

System/34 Receives Multiple Files from 3740

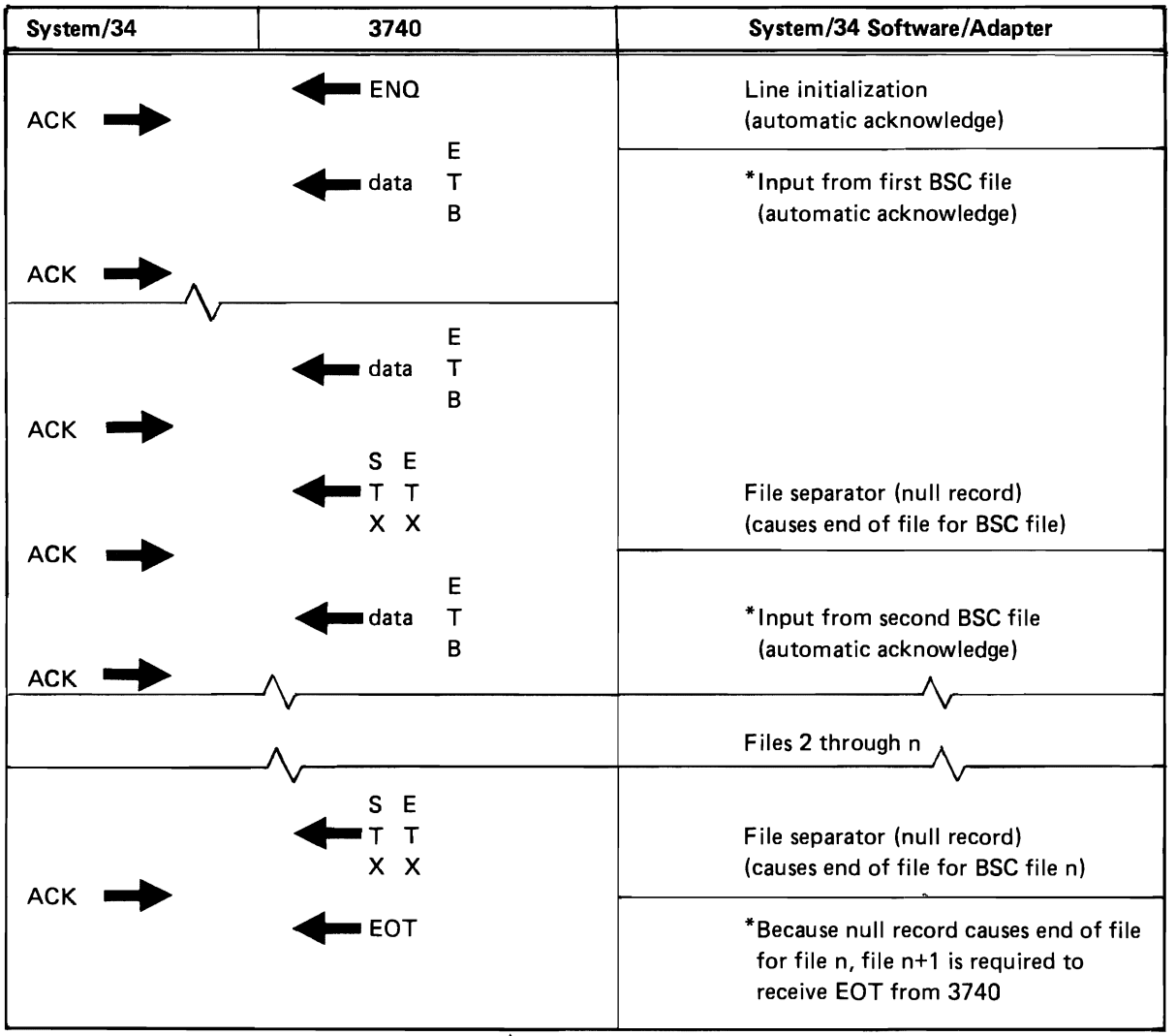


Figure D-20. System/34 Receives Multiple Files from 3740

System/34 Transmits Multiple Files to 3740

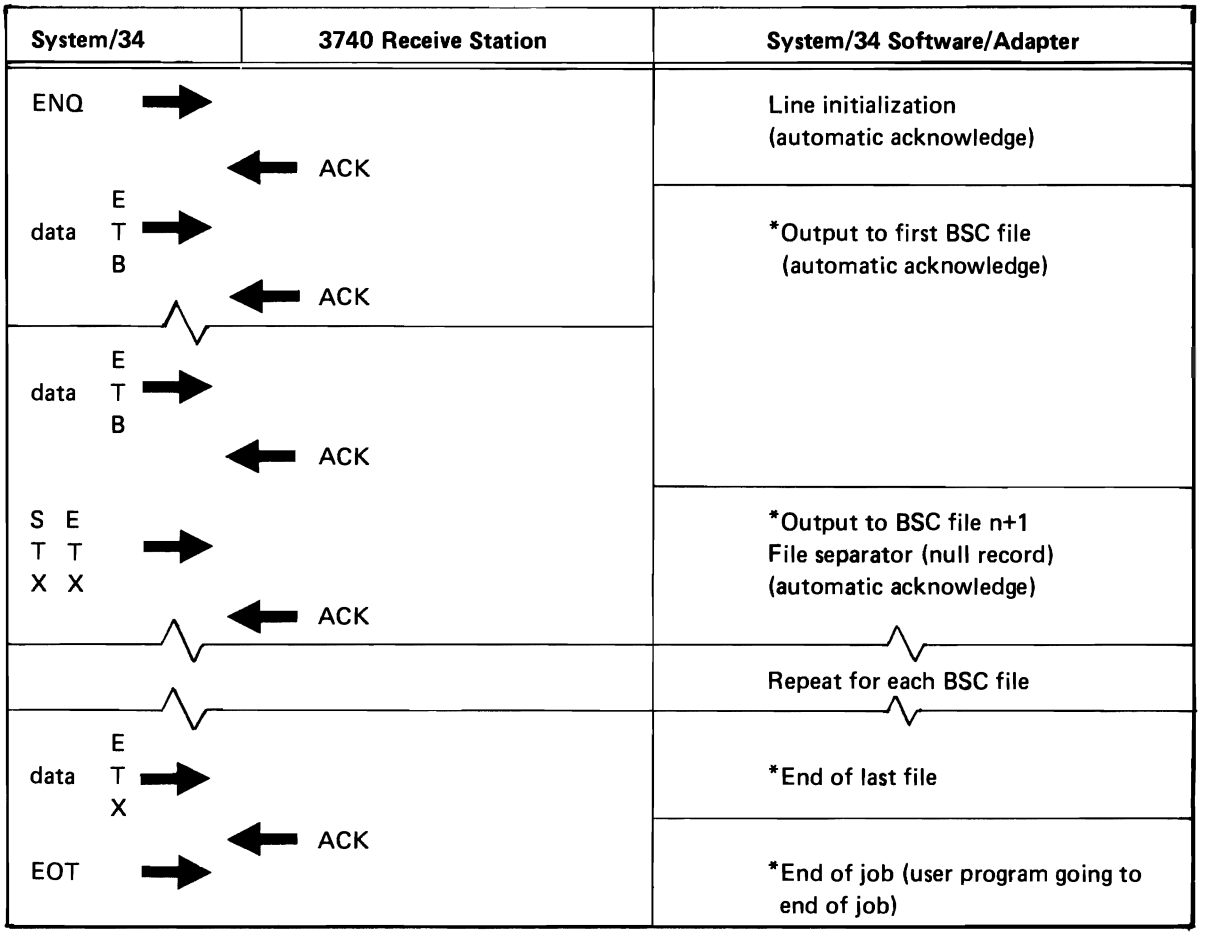


Figure D-21. System/34 Transmits Multiple Files to 3740

System/34 Receives Multiple Files from and Transmits Multiple Files to 3740

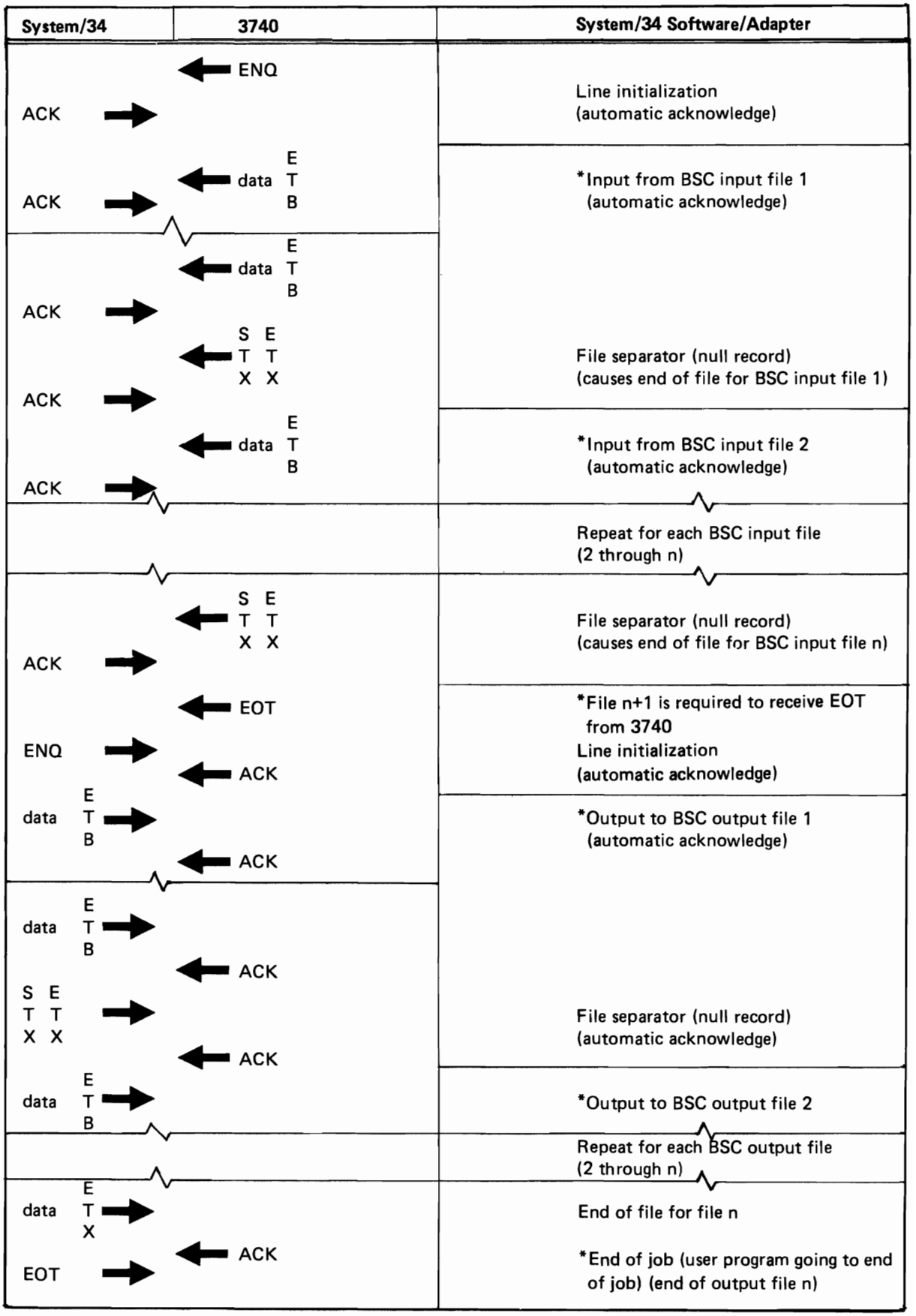


Figure D-22. System/34 Receives Multiple Files from and Transmits Multiple Files to 3740

BLANK COMPRESSION/EXPANSION

Transmitting Blocked Records with Compressed Blanks (Using 3780 Protocol Format)

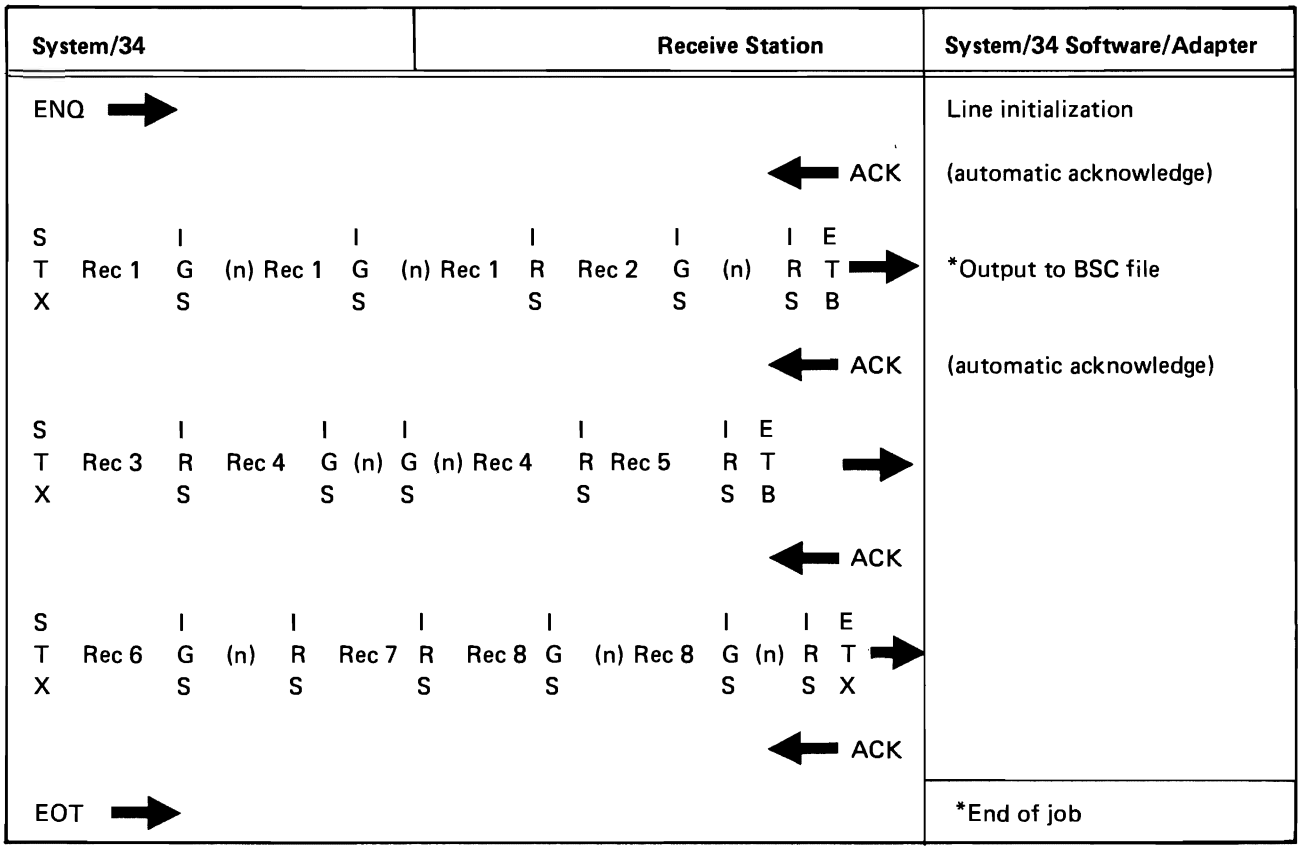


Figure D-23. Blank Compression/Expansion

Notes:

1. The (n) represents the number (one byte) of compressed blanks. Maximum is 63. A hex 40 is always added to the count. For example, if 20 (hex 14) blanks are compressed, the count is hex 54.
2. Records cannot span blocks.
3. Compression cannot be used with transparent output.

BLANK TRUNCATION

Transmitting Blocked Records with Truncated Blanks (Using 3780 Protocol Format)

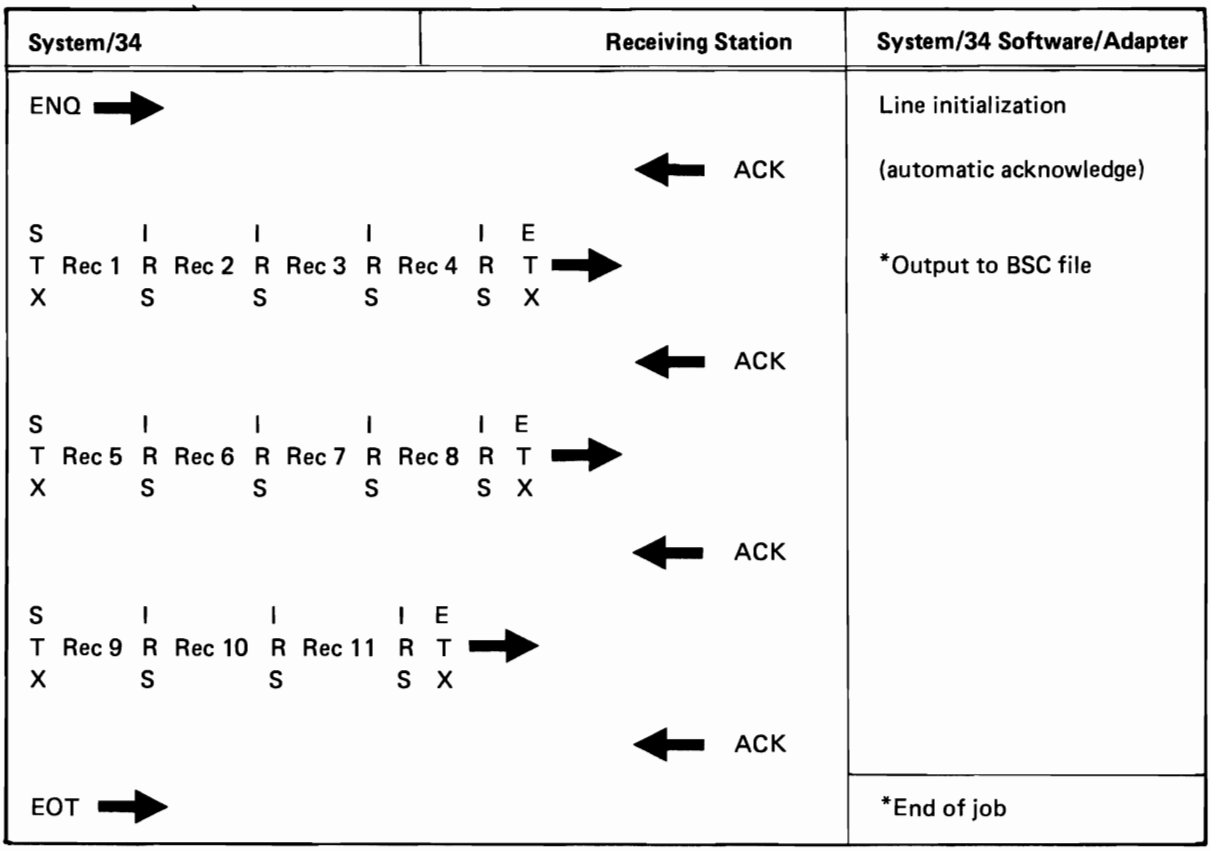


Figure D-24. Blank Truncation

This appendix shows MRJE BSC line protocol in a basic functional sequence.

SIGN ON

System/34	Host System	Comment
ENQ →		Calling station sends ENQ to establish communication.
←	ACK0	Host system responds with ACK0 when ready.
SIGN ON →		Sign-on sequence is sent to the host.
←	ACK0	Host system receives sign on and responds with ACK0.

Figure E-1 (Part 1 of 4). MRJE BSC Line Protocol.

DATA TRANSMISSION

System/34	Host System	Comment
ACK0 →		System/34 sends ACK0 because there is nothing to send (note 1).
←	ACK0	Host system responds with ACK0 because there is nothing to send (note 1).
RFT RD1 →		System/34 sends a request function transmission for the host system's reader one (RD1).
←	GFT RD1	Host system sends a grant function transmission allowing use of RD1.
ACK0 →		System/34 sends ACK0 because there is no data to send to the host system RD1 (note 1).
←	ACK0	Host system responds with ACK0 because there is nothing to send to the System/34 (note 1).
RD1 DATA →		System/34 sends RD1 data stream to the host system.
←	ACK0	Host system responds with ACK0 (positive acknowledgment) if there is no information to send (note 2).
RD1 DATA →		System/34 sends another text block to RD1.
←	Console DATA	Host system responds with data transmitted to the System/34 console (note 2).
RD1 DATA →		System/34 responds with another text block for RD1.
←	RFT PR1	Host system responds with a request function transmission for the System/34 printer task.

Figure E-1 (Part 2 of 4). MRJE BSC Line Protocol

DATA TRANSMISSION (continued)

System/34	Host System	Comment
GFT PR1 →		System/34 sends a grant function transmission for the printer task.
←	PR1 DATA	Host system sends a printer data stream to the System/34.
RD1 DATA (EOF) →		System/34 sends an RD1 text block with end of file (when EOF is sent, the RD1 task is de-activated.
←	PR1 DATA	Host system sends another PR1 text block.
ACK0 →		System/34 responds with ACK0 because PR1 data has been successfully received and there is no more data to send (note 1).
←	PR1 DATA	Host system sends another PR1 text block.
ACK0 →		System/34 sends ACK0 (note 1).
←	PR1 DATA (EOF)	Host system sends a PR1 text block with end of file. This de-activates the printer task.
ACK0 →		System/34 sends ACK0 (note 1).
←	ACK0	Host system sends ACK0 (note 1).

Figure E-1 (Part 3 of 4). MRJE BSC Line Protocol.

SIGN OFF

System/34	Host System	Comment
RFT RD1 →		System/34 sends an RFT for the host system RD1 task.
←	GFT RD1	Host system grants the RD1 to the System/34.
SIGN OFF →		System/34 sends a sign-off sequence.
←	ACK0	Host system sends an ACK0.

Figure E-1 (Part 4 of 4). MRJE BSC Line Protocol

The ACK0 sequence continues until the line is dropped.

Notes:

1. When there is no information to be transferred, the line connection is maintained by sending and receiving ACK0s. This sequence continues until either the host system or the System/34 has information to transmit.
2. If the host system receives a transmission successfully and has data ready to transmit, the data transmission is a positive acknowledgment (ACK0 is not required).

Appendix F. Flow of SNA Commands with SRJE Implementation

SRJE Session Initialization

Note: The SNA terms in this appendix are defined in the glossary.

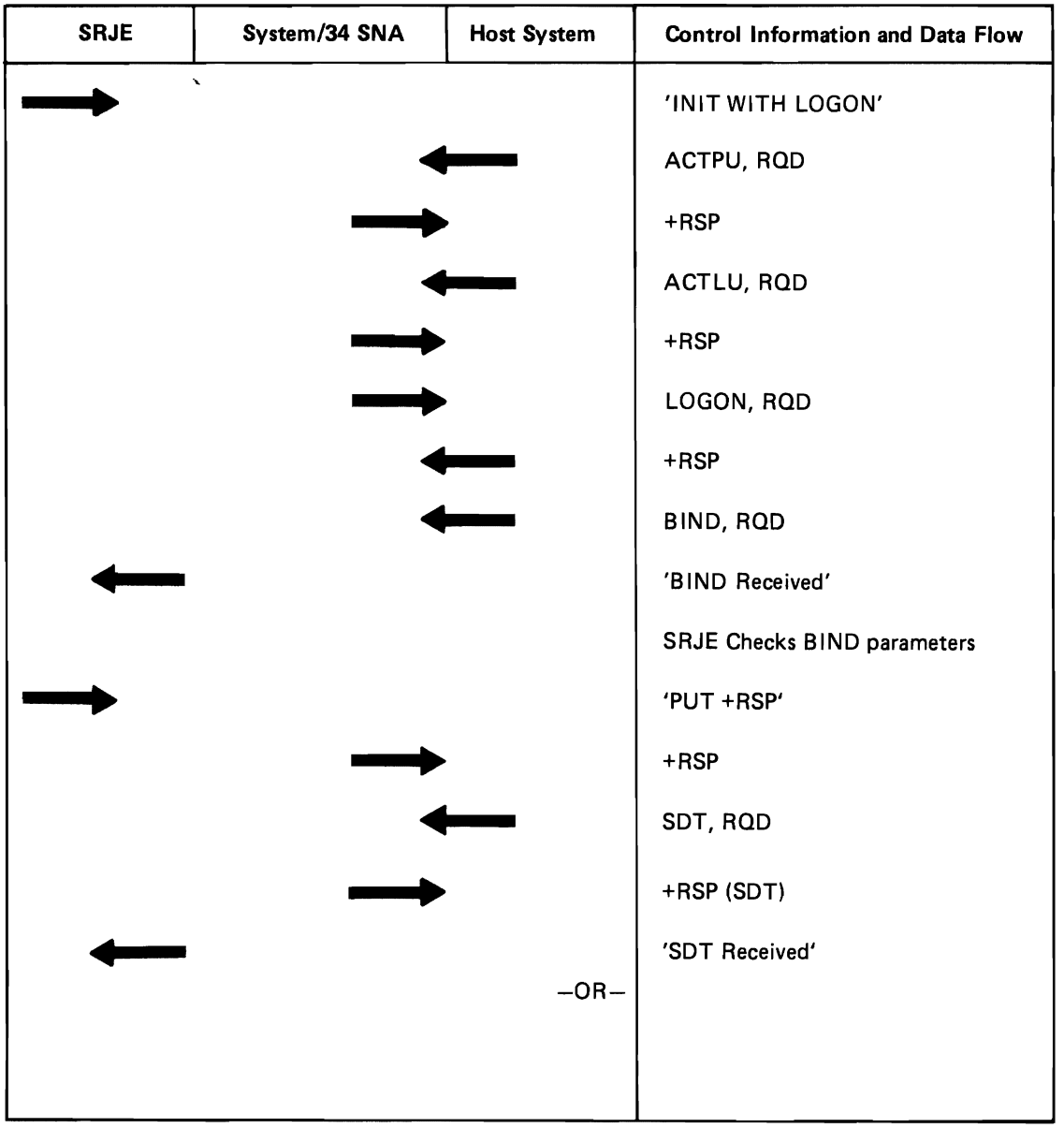


Figure F-1 (Part 1 of 2). SRJE Session Initialization

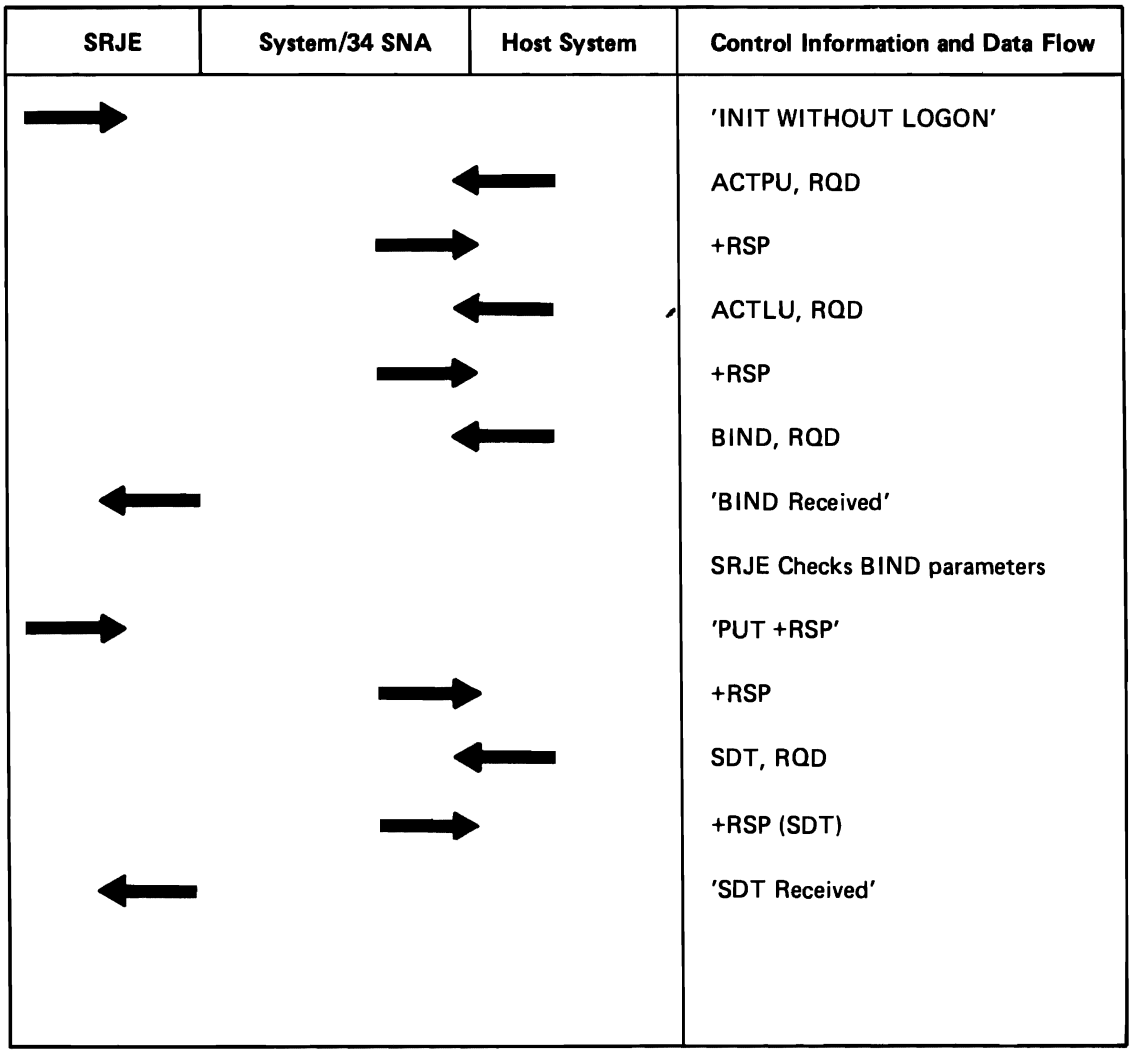


Figure F-1 (Part 2 of 2). SRJE Session Initialization

SRJE Initiated Bracket (Console)

System/34	Host System	Control Information and Data Flow
→		BB,EB,OC,RQD,FMHDR (BEDS, CONSOLE), DATA
←		+RSP

Figure F-2. SRJE Initiated Bracket (Console).

SRJE Initiated Bracket (Reader)

System/34	Host System	Control Information and Data Flow
→		BB,OC,RQD,FMHDR (BDS, READER)
←		+RSP
→		BC, RQE, DATA
→		MC, RQE, DATA
.		.
.		.
.		.
.		.
→		EC, RQD, DATA
←		+RSP
→		EB, OC, RQD, FMHDR (EDS, READER)
←		+RSP

Figure F-3. SRJE Initiated Bracket (Reader)

Host System Initiated Bracket

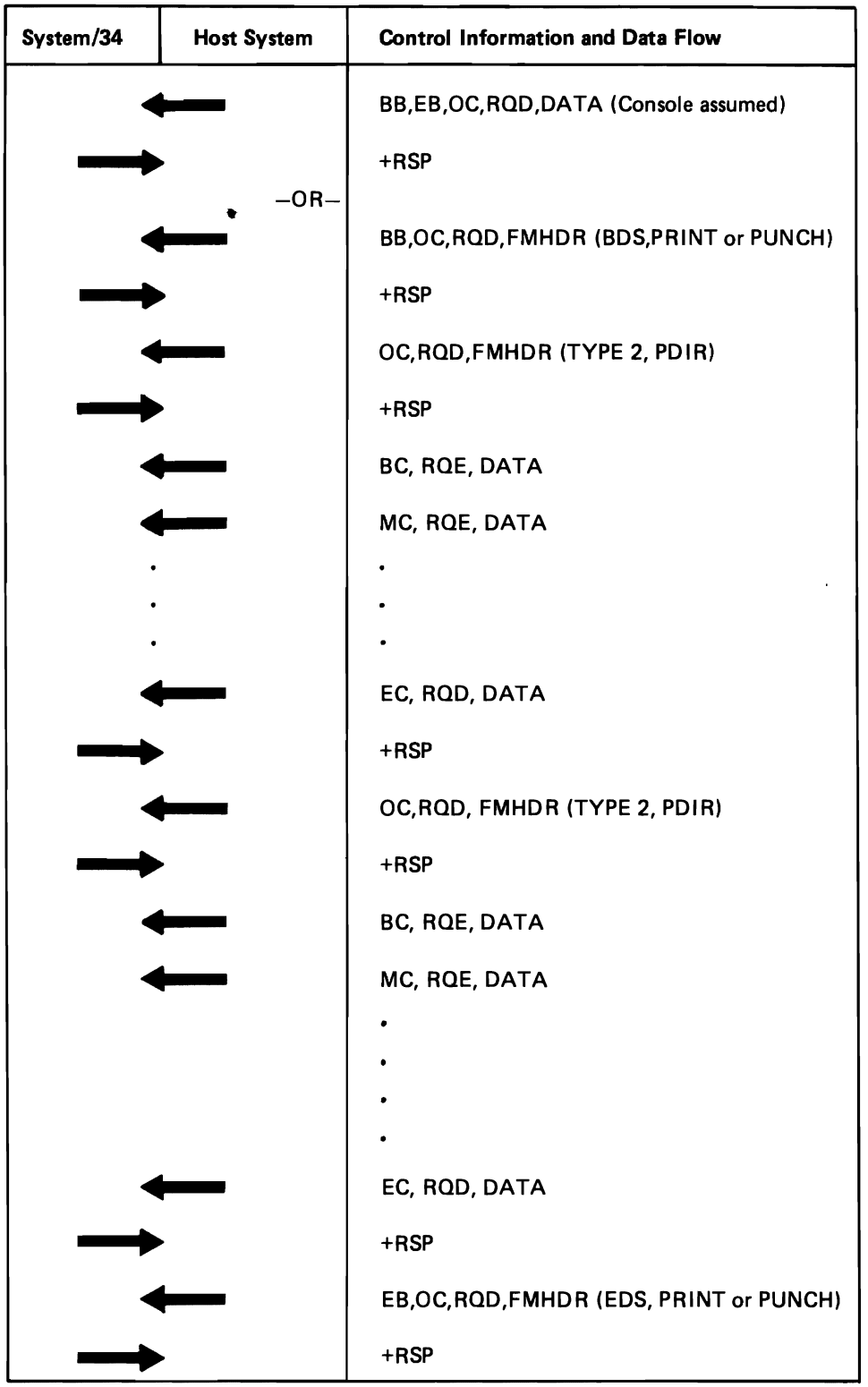


Figure F-4. Host System Initiated Bracket

SRJE Interrupted Inbound Data

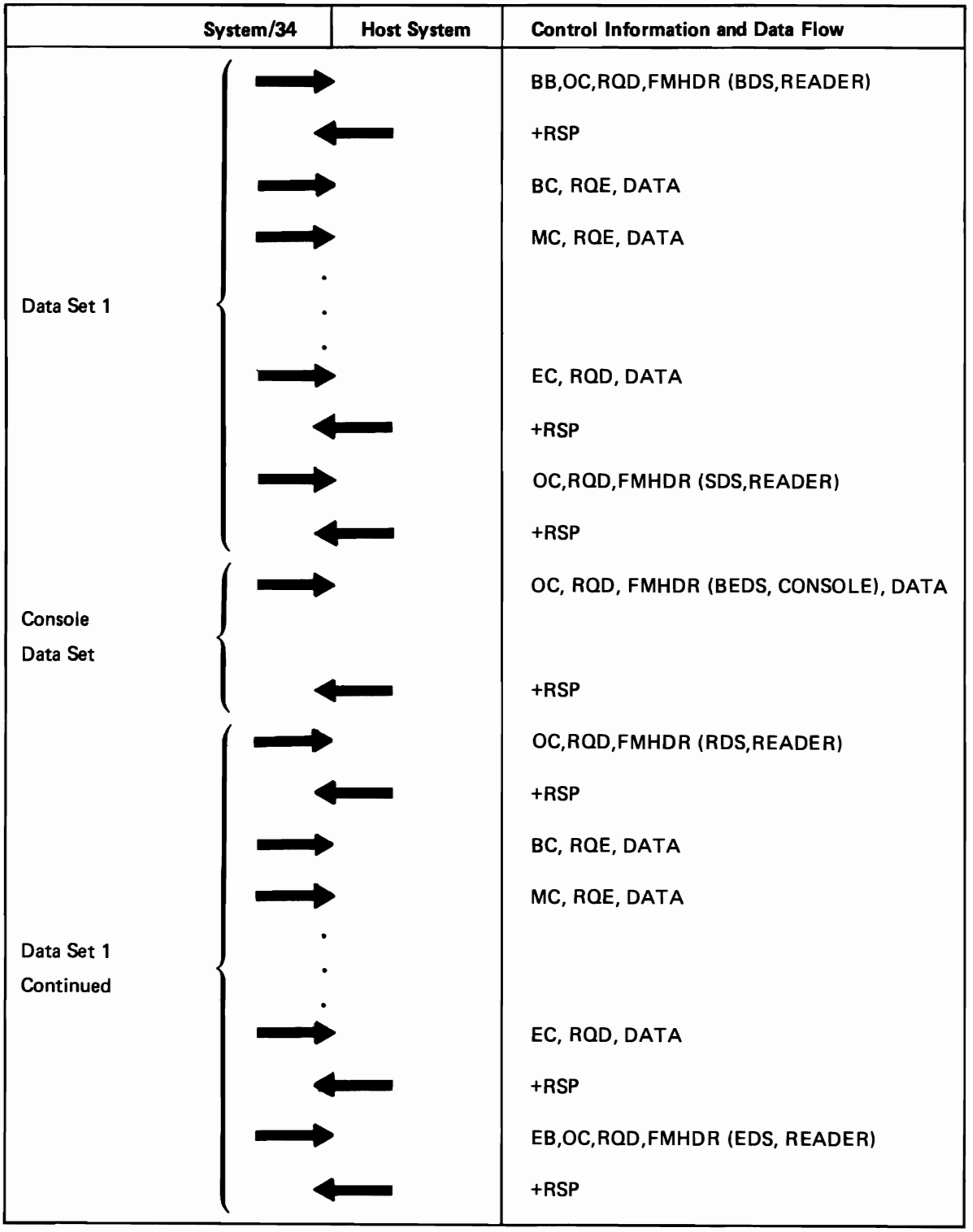


Figure F-5. SRJE Interrupted Inbound Data

Host System Interrupted Outbound Data

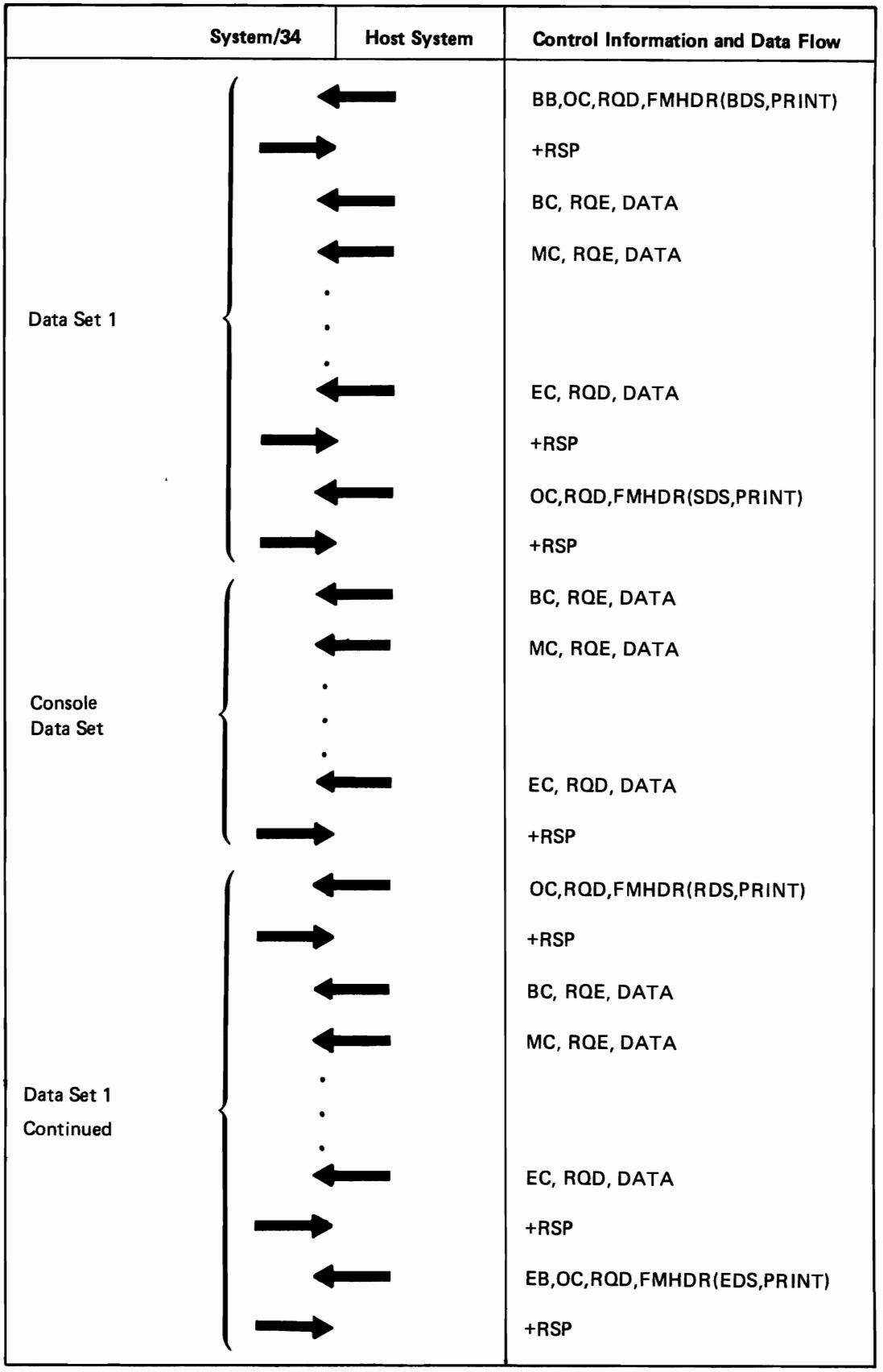


Figure F-6. Host System Interrupted Outbound Data

SRJE Requested Direction Change

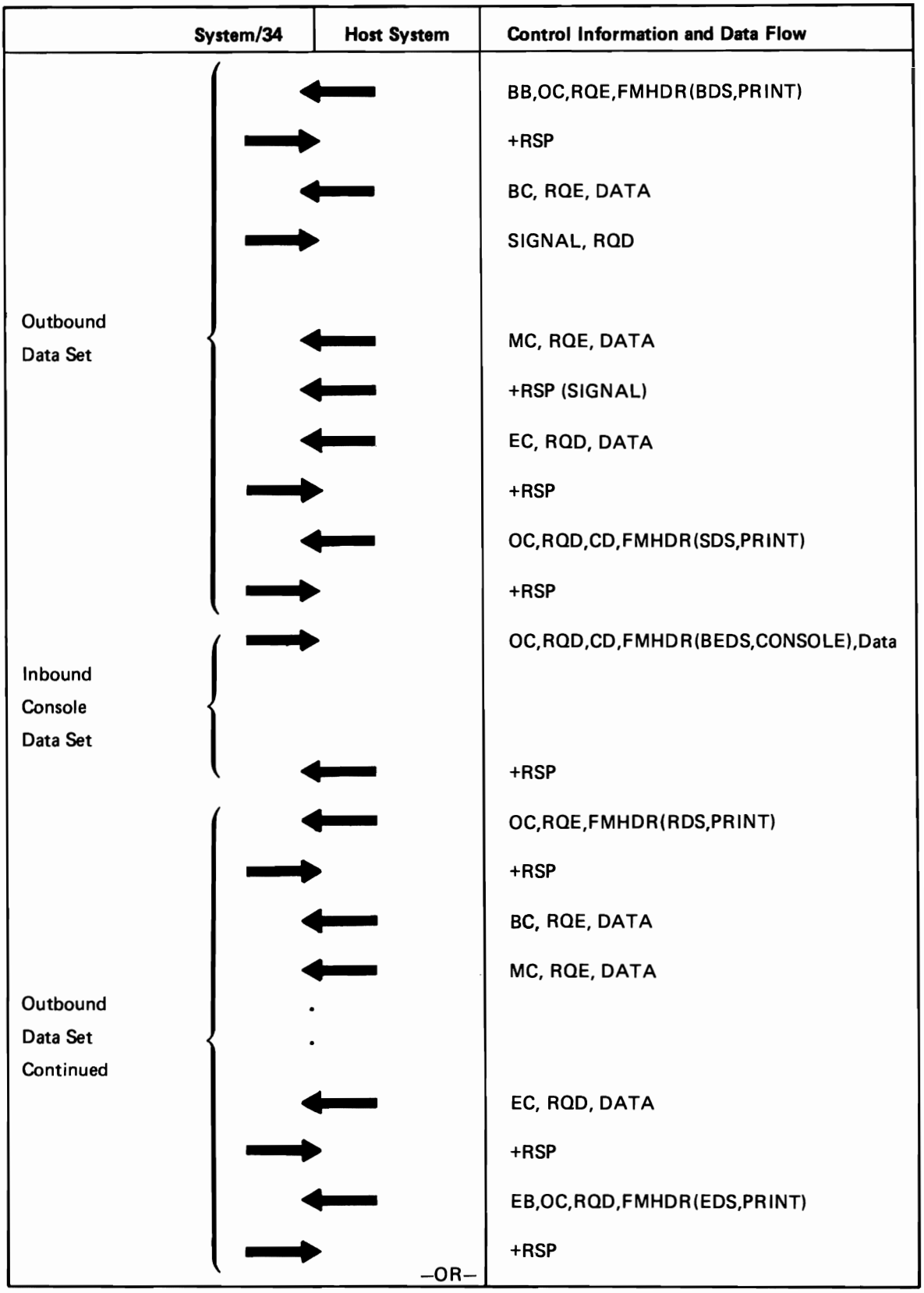


Figure F-7 (Part 1 of 3). SRJE Requested Direction Change

SRJE Requested Direction Change (continued)

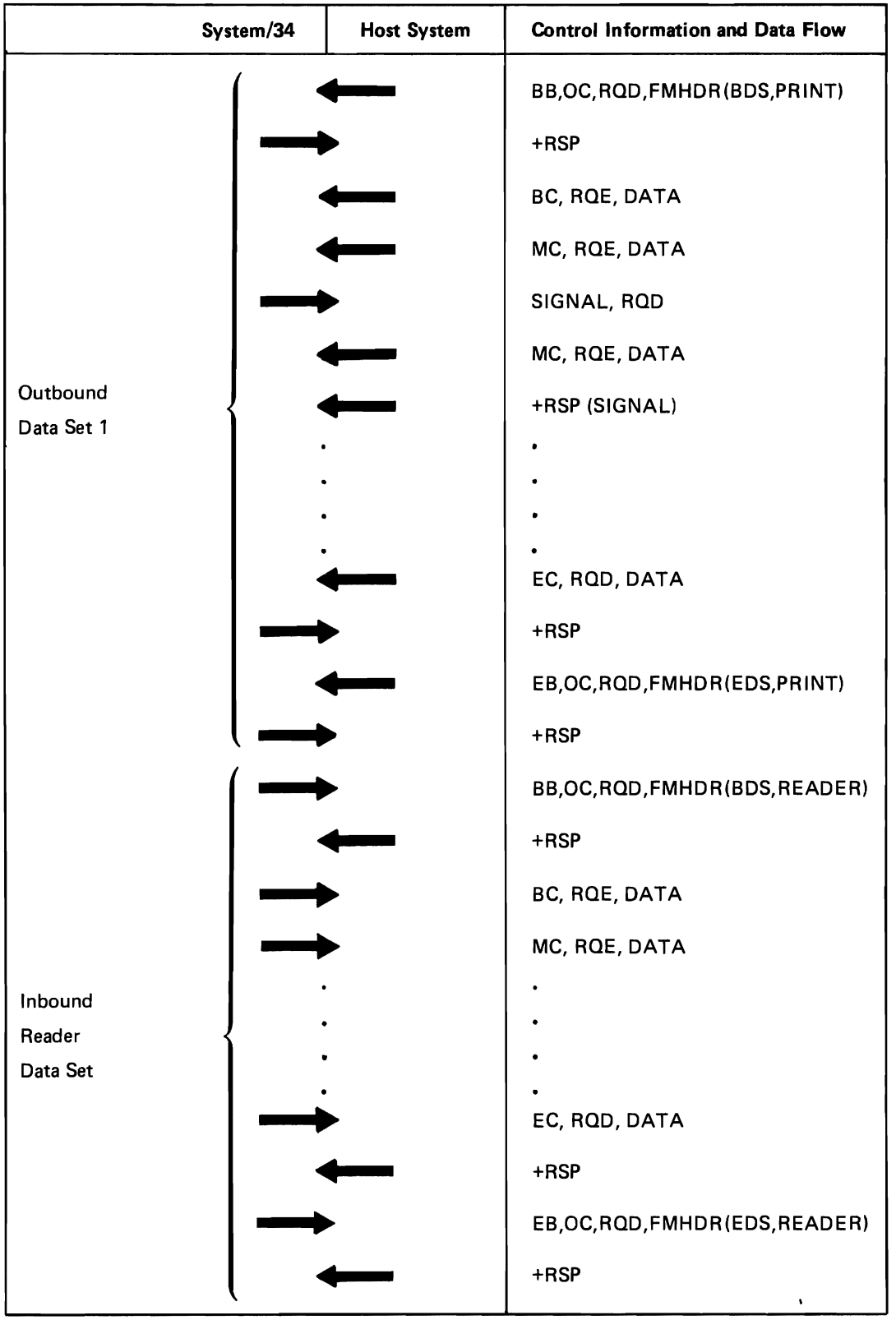


Figure F-7 (Part 2 of 3). SRJE Requested Direction Change

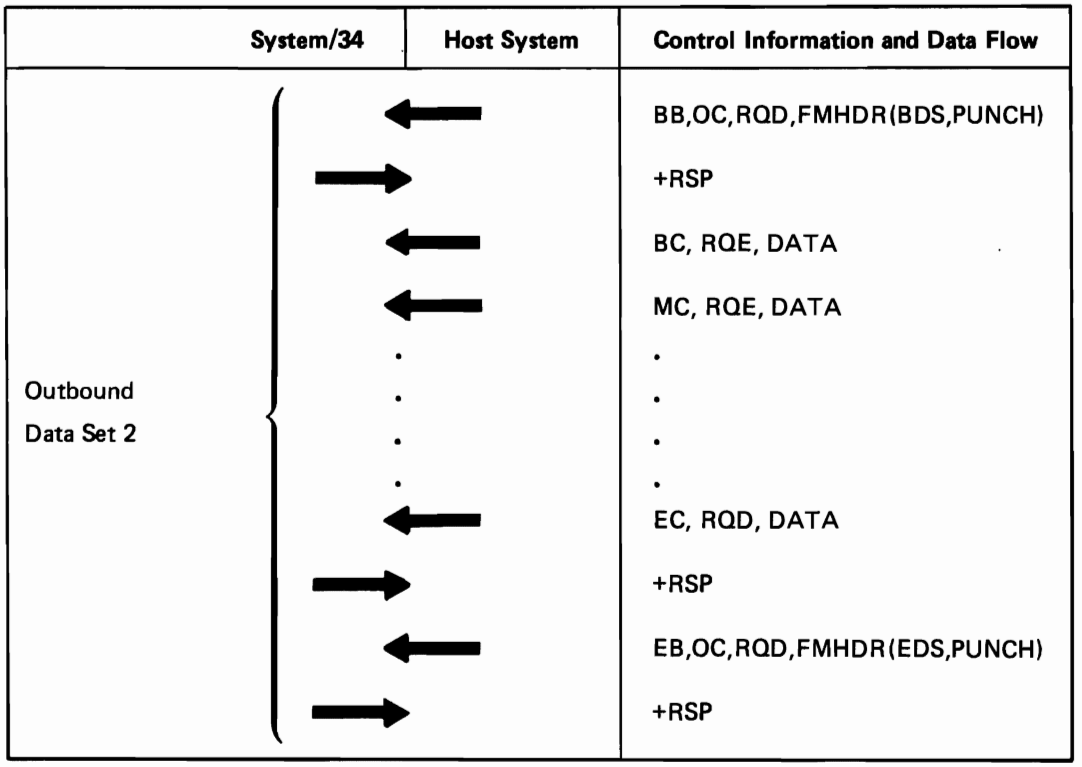


Figure F-7 (Part 3 of 3). SRJE Requested Direction Change

Host System Detected FM Error

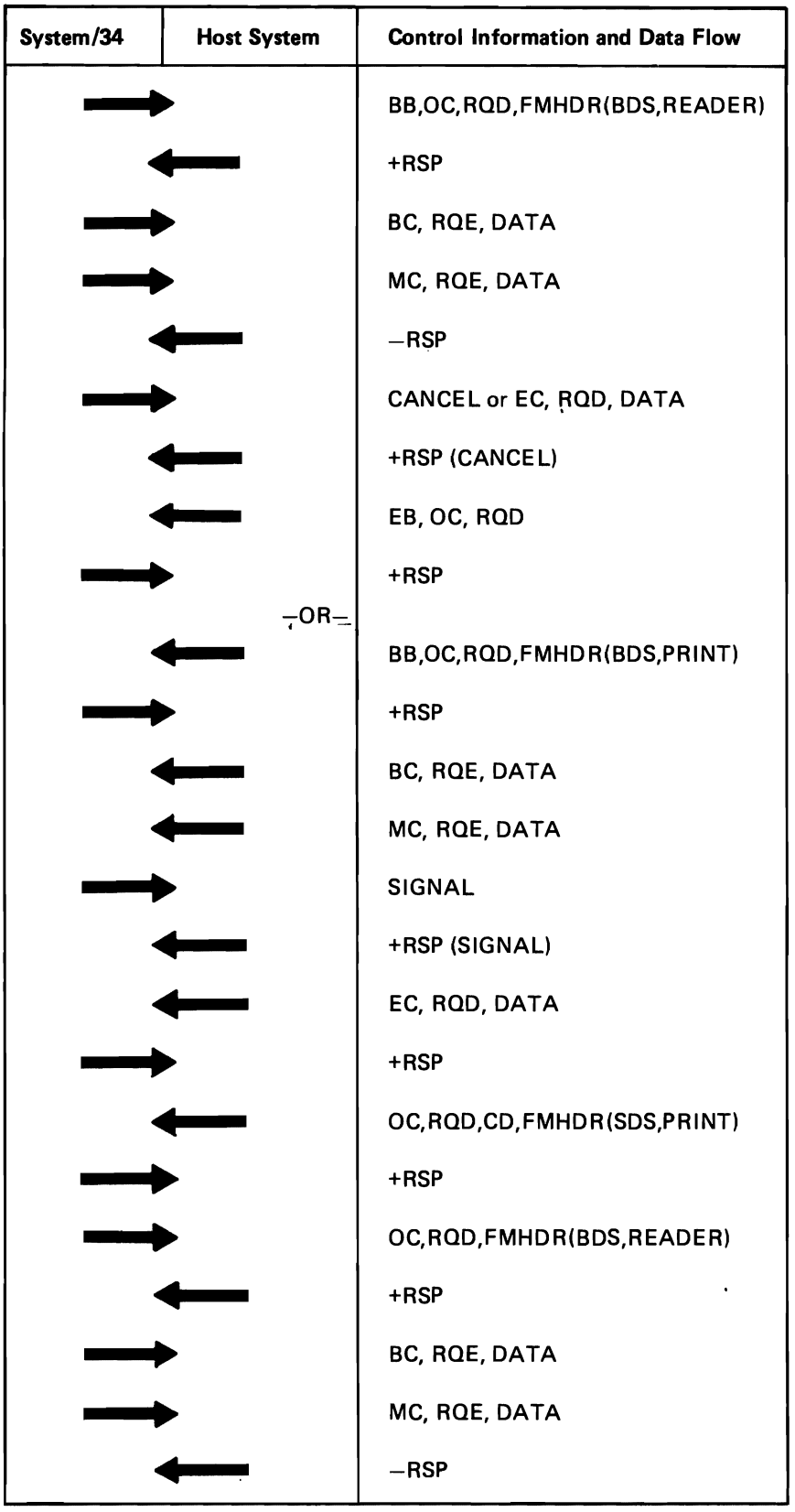


Figure F-8 (Part 1 of 2). Host System Detected FM Error

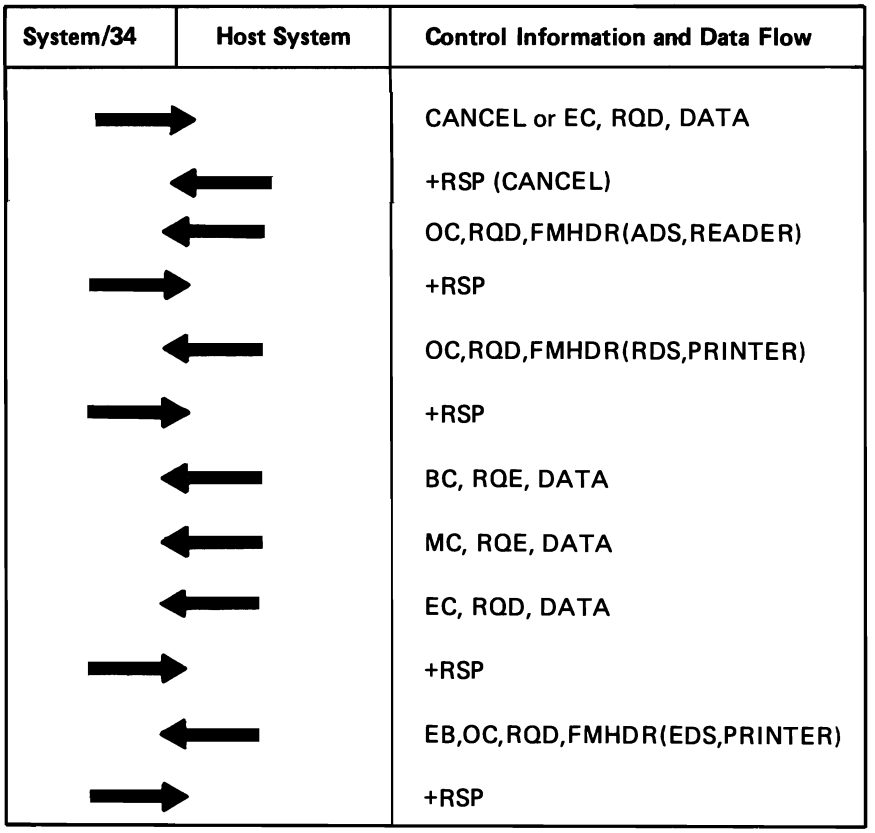


Figure F-8 (Part 2 of 2). Host System Detected FM Error

Host System Initiated Termination Sequence

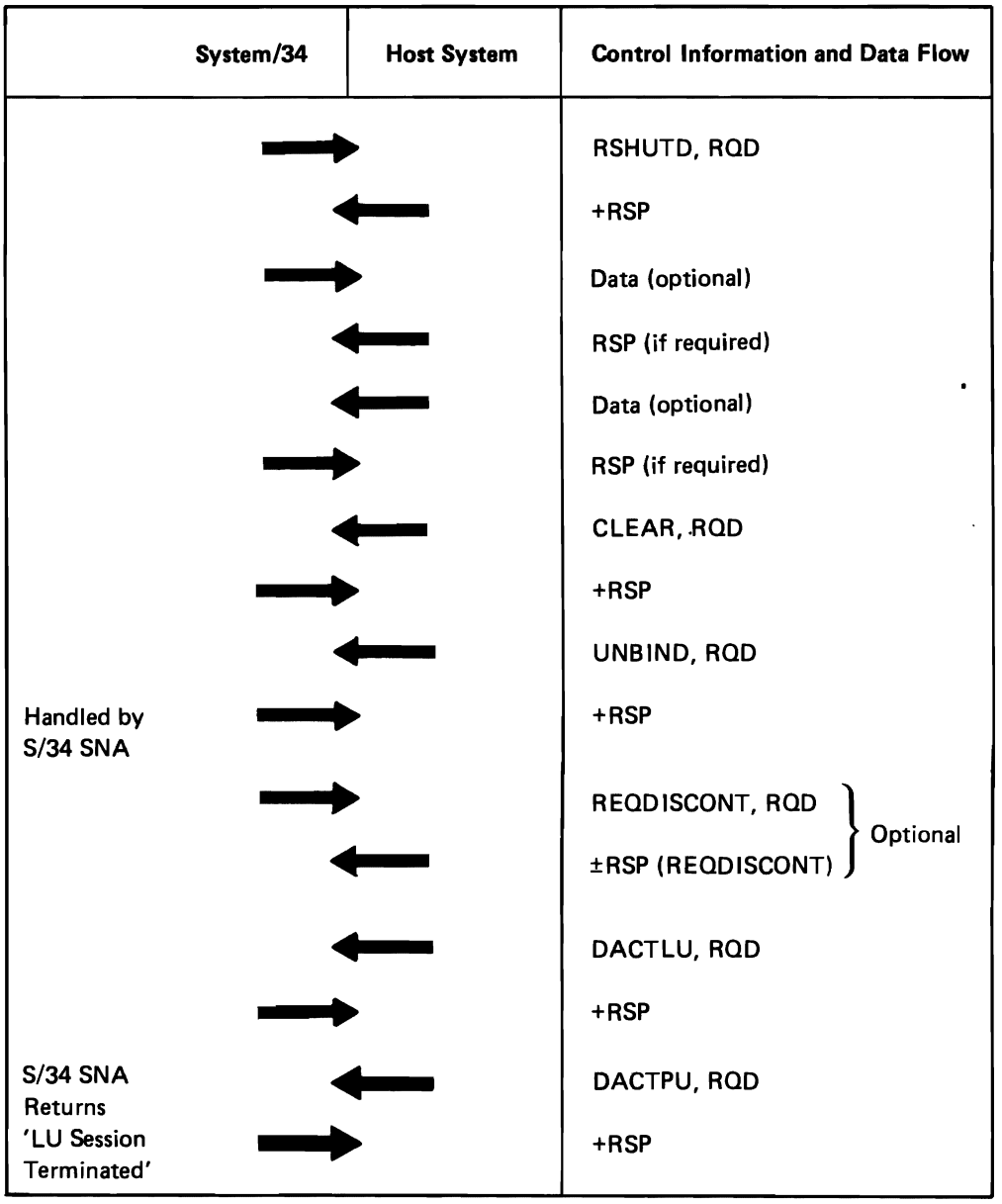


Figure F-9. Host System Initiated Termination Sequence

Glossary

ACK0: The even-numbered positive acknowledge sequence.

ACK1: The odd-numbered positive acknowledge sequence.

ACTLU: Activate logical unit.

ACTPU: Activate physical unit.

addressing: The means by which a sending or control station selects the unit to which it will send a message.

ADS: All data stream.

ASCII: American National Standard Code for Information Interchange.

ASP: Asymmetric multiprocessing system.

automatic answer: A machine feature that permits a station to respond to a call it receives over a switched line without operator action.

automatic call: A machine feature that permits a station to initiate a connection with another station over a switched line without operator action.

BB: Begin bracket.

BC: Begin chain.

BCC: Block check character.

BDS: Begin data stream.

BEDS: Begin/end data stream.

bind: An SNA command used to define the protocols for a session.

bps: Bits per second.

BSC (binary synchronous communications): A form of line control that provides a set of rules for transferring data over a communications line connecting two or more devices that use a communications adapter.

BTAM: Basic telecommunications access method.

call: The action performed by the requesting party, or the operations necessary to make a request or the effective use made of a connection between two stations.

CANCEL: Cancel stream.

CCP: Communications control program.

CD: Change direction.

central station: See *control station.*

clocking: A method of controlling the number of data bits sent on a data communications line in a given time.

command file: A disk file, procedure member, or source member that is designated as a command file by a data communications utility program. A command file can contain utility control statements and/or records to be transmitted to the host system.

communications adapter: A hardware feature that enables System/34 to become a part of a data communications network.

compression: A technique for removing strings of duplicate characters and truncating trailing blanks before transmitting data.

configuration record: See *system configuration record; display station configuration record.*

control station: The primary or controlling computer in a multipoint data communications configuration. The control station controls the sending and receiving of data.

conversational file: A BSC file that allows receiving or sending data characters as an acknowledgment instead of the ACK0 or ACK1 sequence.

DACTLU: De-activate logical unit.

data communications: The transmission of data between systems and/or remote devices over a communications line.

data file: A disk file, procedure member, or source member that is designated as a data file by a data communications utility program. A data file can contain only records to be transmitted to the host system.

data link: The equipment and rules (protocols) used for sending data over a communications line.

data mode: A time at which BSC is transmitting or receiving characters on the line.

data stream: All data transmitted over a data link in a single read or write operation.

DDSA: Digital Data Service Adapter (*Trademark of American Telephone & Telegraph Co.).

DISC: The transmission control sequence for disconnect on a switched line.

disconnect time-out: An indication that the BSC station you were communicating with has gone on hook or hung up.

display station: An input/output device containing a display screen on which data is displayed and an attached keyboard from which data is entered.

display station configuration record: An area on disk that describes a command display station's environment. The display station's configuration record contains information such as the session date, the work station ID of the printer to be used for the display station's printed output, and the region size for jobs submitted from the display station.

DLE (data link escape): A control character used exclusively to provide supplementary line-control signals (control character sequences or DLE sequences).

DOS: Disk operating system.

duplex: A data communications network that permits concurrent transmission and reception of data.

EB: End bracket.

EBCDIC: Extended binary coded decimal interchange code.

EBCDIC transparency: See *transparent text mode.*

EC: End chain.

EDS: End data stream.

ENQ: Enquiry character.

EOT: End of transmission.

error history file: A push-down stack of the last BSC errors that have occurred.

ETB: End-of-transmission-block character.

ETX: End-of-text character.

expanded communications buffer: A special feature of the 3741 which allows multiple records to be transmitted or received in one block of data.

extent: A continuous space on disk or diskette that is occupied by, or reserved for, a particular file.

FMHDR: File management header.

half duplex: Permitting data communications in opposite directions, but not at the same time.

host system: The primary or controlling computer in the communications network. See also *control station.*

intermediate block check: A function that permits checking of each record, instead checking the contents of the total buffer when large buffers of data are received.

interrecord separator: The last character of a record which signals the end of that record and the beginning of another on a data communications network.

ITB: Intermediate text block character.

JES2: Job entry subsystem 2.

JES3: Job entry subsystem 3.

line control characters: See *transmission control characters.*

manual answer: Operator actions to make a station ready when a station receives a call over a switched line.

manual call: Operator actions to make a connection with a station over a switched line.

MC: Middle in chain.

MLMP: Multiline multipoint.

modem (modulator/demodulator): A device that connects a communications adapter to a communications line.

monitor mode: A time during which the communications adapter is looking for BSC synchronization characters.

MRJE: MULTI-LEAVING remote job entry.

MRT program: Multiple requestor terminal program.

MSP/7: Modular system programs.

MULTI-LEAVING remote job entry: An SSP function that allows the user to communicate with a system over a communications line using BSC.

multidropped terminal: See *tributary station.*

multipoint data link: A network configuration in which connected stations communicate with each other over a common communications line on a time-shared basis. The primary station controls and maintains the data link.

NAK: Negative acknowledgment character.

NCP: Network control program.

NEP: Never-ending program.

network: A configuration by which two or more stations can communicate.

nonswitched line: A connection between systems or devices that does not have to be made by dialing.

OC: Only in chain.

OS: Operating system.

PDIR: Peripheral data set information record.

peripheral data set information record (PDIR): A special control record sent from the host to SRJE that describes the printer data set that is to follow.

point-to-point line: A data communications facility that connects a single remote station to a data processing system. A point-to-point line can be either switched or nonswitched.

polling: In a multipoint environment, a request to send, transmitted from the primary station to a specific secondary station.

POWER: Priority output writers, execution processors, and input readers.

RDS: Resume data stream.

receive mode: A time during which the communications adapter looks for synchronization characters and then stores the data characters in main storage.

receive time-out: An indication that no data has been received by this communications adapter in a given period of time.

REQDISCONT: Request discontact.

RES: Remote entry services.

reverse interrupt (RVI): A request by the receiving station to the sending station to stop transmitting and receive a message.

RH: Request/response header.

RJE: Remote job entry.

RQD: Request definite response.

RQE: Request exception response.

RSCS: Remote spooling communications system.

RSHUTD: Request shut down.

RSP: Response.

RVI: Reverse interrupt character.

SDLC: Synchronous data link control.

SDS: Suspend data stream.

SDT: Start data traffic.

session: The period of time during which programs or devices can communicate with each other.

SIGNAL: Ready to transmit.

SNA: Systems network architecture.

SNA remote job entry (SRJE): An SSP utility that allows the user to communicate with a host system in an SNA environment using SDLC line discipline.

SOH: Start-of-heading character.

SRJE: SNA remote job entry.

standby line: A modem feature that allows a point-to-point nonswitched line modem to also function on a point-to-point switched line.

station: A system or device that can send or receive data over a communications line.

STX: Start-of-text character.

switched line: A connection between two stations that is established by dialing.

SYN: Line synchronization character.

synchronous data link control (SDLC): A discipline for the management of information transfer over a data communications channel.

system console: A display station designated to activate specific system functions, and to control and monitor system operation, in addition to performing as a command display station.

system configuration record: Information stored on disk that describes system characteristics and programming support; for example, system data format, disk capacity, and main storage capacity.

systems network architecture: An IBM communications protocol for controlling data transfers in a data communications network.

TCAM: Telecommunications access method.

text transparency: A provision that allows BSC to send and receive messages containing any or all of the 256 character combinations in EBCDIC, including transmission control characters. EBCDIC and control characters are all sent as text.

transmission control characters: Special characters that are included in a message to control communication over a data link.

transparency: See *transparent text mode.*

transparent text mode: A method of binary synchronous transmission in which only transmission control characters preceded by the DLE control character are processed as transmission control characters.

tributary station: A secondary or noncontrolling device in a multipoint data communications configuration.

TTD: Temporary text delay sequence.

UNBIND: Terminate the BIND for this session.

VM: Virtual machine.

VS: Virtual storage.

VTAM: Virtual telecommunications access method.

WACK (wait before transmit positive acknowledgment): A BSC data link control sequence for wait before transmitting. The WACK sequence indicates to the transmitting station a positive acknowledgement and a temporary not ready condition.

work station: A device that lets a person transmit information to or remove information from a computer, or both, as required to perform the job.

wrap test: A test that checks attachment or controller circuitry.

READER'S COMMENT FORM

IBM System/34
Data Communications
Reference Manual

SC21-7703-3

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you. Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Please use this form only to identify publication errors or to request changes in publications.

Possible topics for comment are:

Clarity Accuracy Completeness Organization Retrieval Legibility

☐ If your comment does not need a reply (for example, pointing out a typing error) check this box and do not include your name and address below. If your comment is applicable, we will include it in the next revision of the manual.

☐ If you would like a reply, check this box. Be sure to print your name and address below.

Page number (s): Comment(s):

Please print

Name _____

Company or
Organization _____

Address _____

_____
City State Zip Code

No postage necessary if mailed in the U.S.A.

Cut Along Line

Fold and tape | Please do not staple | Fold and tape

NO POSTAGE
NECESSARY IF
MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N. Y.

POSTAGE WILL BE PAID BY . . .

International Business Machines Corporation
Development Laboratory
Information Development, Department 532
Rochester, Minnesota 55901

Fold and tape | Please do not staple | Fold and tape

IBM

International Business Machines Corporation

System/34 Data Communications Reference Manual (File No. S34-36) Printed in USA SC21-7703-3

International Business Machines Corporation

System/34 Data Communications Reference Manual (File No. S34-36) Printed in USA SC21-7703-3

SC21-7703-3