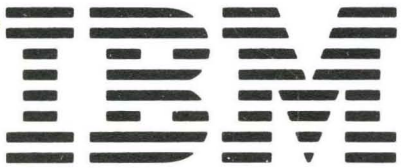


SC21-7703-1

File No. S34-36

IBM System/34
Data Communications
Reference Manual

SC21-7703-1

File No. S34-36

IBM System/34
Data Communications
Reference Manual

Second Edition (July 1978)

This is a major revision of, and obsoletes, SC21-7703-0. Chapters 6 and 7 are new chapters, and many changes have been made to chapters 1 through 5. Please review the portions of the manual that you have been using.

This edition applies to release 2 modification 0 of IBM System/34 System Support Program Product (Program Number 5726-SS1) and to all subsequent releases until otherwise indicated in new editions or technical newsletters. Changes are periodically made to the information herein; before using this publication, refer to the latest *IBM System/34 Bibliography*, GH30-0231, for the editions that are applicable and current.

Portions of this manual are for planning purposes only. (Consult the *Preface*.) The planning information is subject to change before customer shipment.

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

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 Corporation, Publications, Department 245, Rochester, Minnesota 55901. IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

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.

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

The information concerning SRJE is included for planning purposes only. Also, the information in the print utility ($DCSUP) pertaining to SRJE is for planning purposes. This information is subject to change before first customer shipment.

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. Polling and Addressing Characters for System/34 Tributary Stations

Appendix D. System/34 Interface to BSC Line Protocol

Appendix E. MRJE BSC Line Protocol

Appendix F. SRJE Line Protocol

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

- *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 MVS JES2 Commands*, GC23-0007
- *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/VS POWER/VS Installation and Operator's Guide*, GC33-5403
- *DOS/VS System Control Statements*, GC33-5376

Contents

Chapter 1. Introduction

IBM System/34 and its program products provide 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
- Basic Assembler with SNA support

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 enables 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 the line connection, 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 or SRJE utility.

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 System/34 SRJE utility, directed to another work station, or to the host system output devices. The SRJE utility establishes the 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 as follows:

- For a general description of data communications, see *Data Communications Concepts*, GC21-5169.
- For a more detailed discussion of BSC concepts, see *General Information—Binary Synchronous Communications*, GA27-3004.
- For a more detailed discussion of SNA, see *IBM Systems Network Architecture General Information*, GA27-3102.

- For a more detailed discussion of SDLC, see *IBM Synchronous Data Link Control General Information,* GA27-3093.

- 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. The communications adapter operates in half-duplex mode over point-to-point and multipoint networks, including four-wire communications lines. Operation of this 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. System/34 supports automatic answering only when the attached modem (external or integrated) also supports it.

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

- Use the same clocking source – modem, network (for DDSA), or business machine

- Be set to operate at 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 32K bytes

- Communications adapter

- System/34 System Support Program Product

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

System/34 Requirements for SNA

The minimum system configuration required to support the SRJE utility or a basic assembler SNA program is:

- System/34 with minimum main storage size of 48K 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 6K bytes of nonswappable main storage. BSC programs cannot be run from the input job queue.

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

OTHER SYSTEMS

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

- 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 3747 Data Converter
- IBM 5231 Data Collection Controller Model 2 (as a 3741 in transmit mode only)
- IBM 3750 Switching System (World Trade only)

BSC STATION TYPES

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

1. Receive only (receive input data from a remote terminal).
2. Transmit only (transmit data to a remote terminal).
3. Transmit and receive–no conversational reply.
   a. Transmit a file, then receive another file.
   b. Receive a file, then transmit another file.
   c. 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.

RPG II TELECOMMUNICATIONS SPECIFICATIONS

You use the RPG II 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.

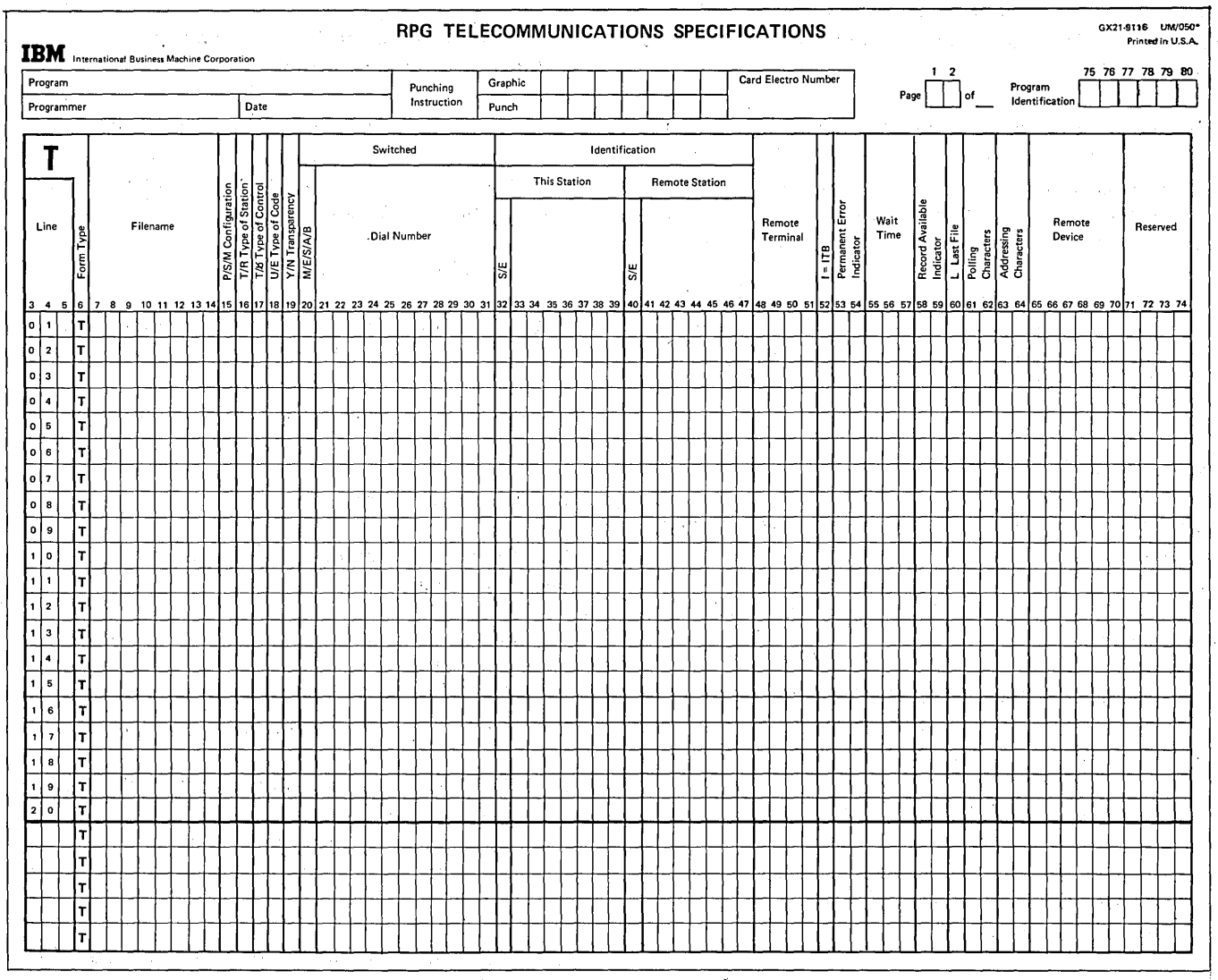


Figure 2-1. Telecommunications Specification Sheet

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
Alpha-meric	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.
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.
S	This is a point-to-point switched network.

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) transmisssion control characters are used. The System/34 does the file translation.
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.

Note: 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's 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's 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
Alpha-meric	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's 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's 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's 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
Alpha-meric	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's 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 file 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.

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 waits with no messages being sent or received before a permanent error occurs.
Blank	The system convention for timeout, 180 seconds, is used.

A permanent-error indication is recognized by the system whenever the wait time on an idle 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 caused by this System/34 program 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
Alpha-meric	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
Alpha-meric	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. 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.

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.

Note: Blank truncation cannot be used with ITB mode. Blank truncation can be specified with transparent mode, however, this negates the truncation feature since no blanks are removed.

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 Restrictions

Do not use /* in the first two positions of a record because that signifies end of file (even for data transmitted and received in a BSC program).

Note: When a table is dumped, a /* is generated to indicate the end of the table. Therefore, if you dump a table to a BSC file, expect an end-of-file condition when the table is received.

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.

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.*

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.

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,* GA21-9170.

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 II 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
60	L	Last-file indicator

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.

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 TRNSMT

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	I/O/U/C/D	P/S/C/R/T/D/F	E	A/D	F/V/S/M/D	Block Length	Record Length	L/R	A/P/I/K	I/X/D/T/R or 2	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	K	Option	Entry	A/U	Number of Tracks for Cylinder Overflow	Number of Extents	Tape Rewind	R/U/N	File Condition U1-U8
02	F	DISKFILE	I	P			F	96	96									DISK														
03	F	WKLYSMRY	O				F	90	30			2						BSCA														
04	F	STOPT	O				F	120	120									PRINTER														
05	F																															
06	F																															
07	F																															
08	F																															
09	F																															
10	F																															
	F																															
	F																															

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

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 code (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.

RPG TELECOMMUNICATIONS SPECIFICATIONS

GX21-9116 U/M050* Printed in U.S.A.

IBM International Business Machine Corporation

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

T

Line	Form Type	Filename	P/S/M Configuration	T/R Type of Station	T/& 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	WKLYSMRYST					E	Y	A		EMYID		EYOURID				25070						
02	T																						
03	T																						
04	T																						

RPG INPUT SPECIFICATIONS

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

IBM International Business Machines Corporation

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

I

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	DISKFILEBB				02		1		C2																				
02	I																				66	75		SLSMANL1						
03	I																				78	83	2	AMOUNT						
04	I																													
05	I																													
06	I																													
07	I																													
08	I																													

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.

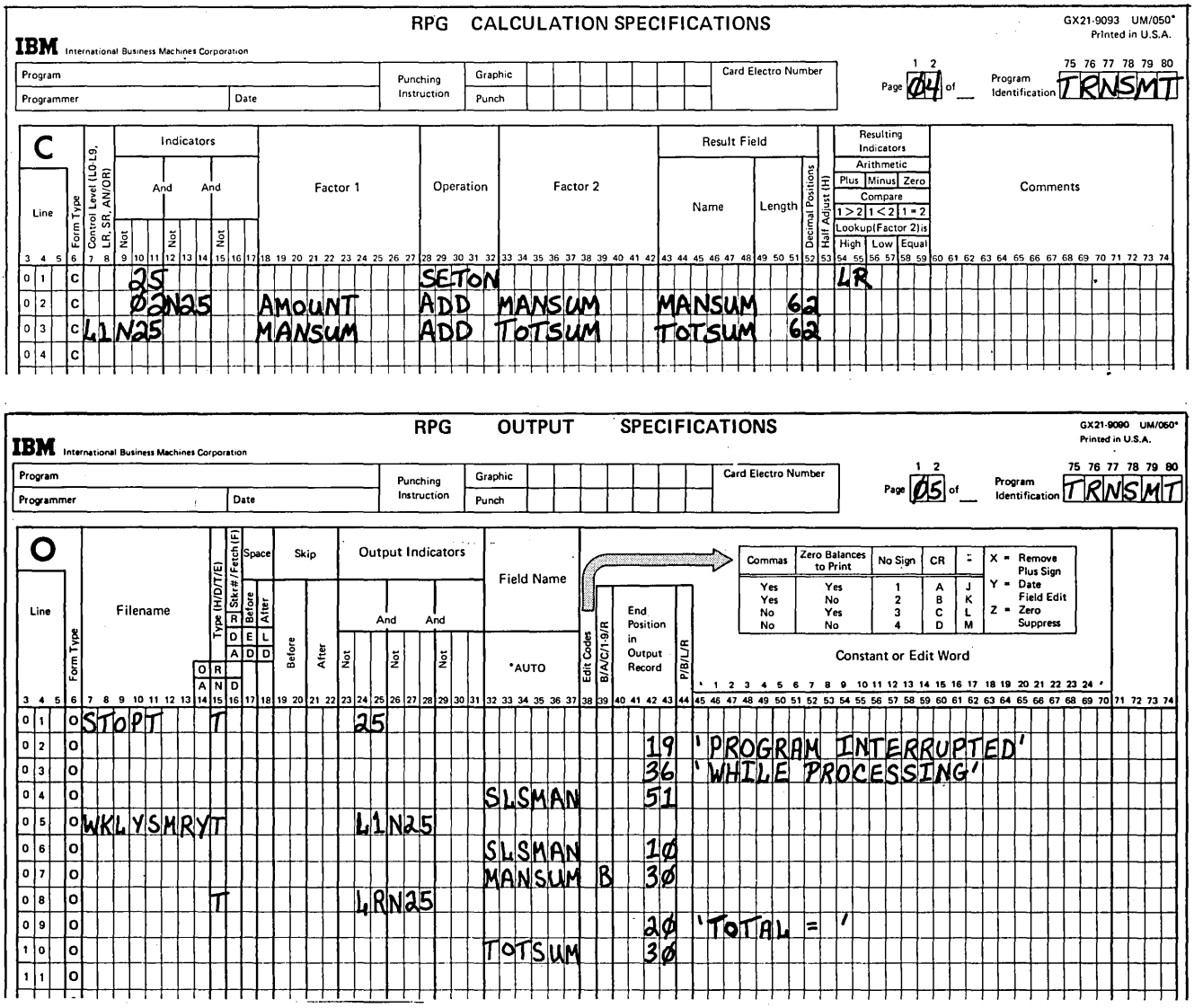


Figure 2-2 (Part 3 of 3). Transmit Program

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 that begin 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.

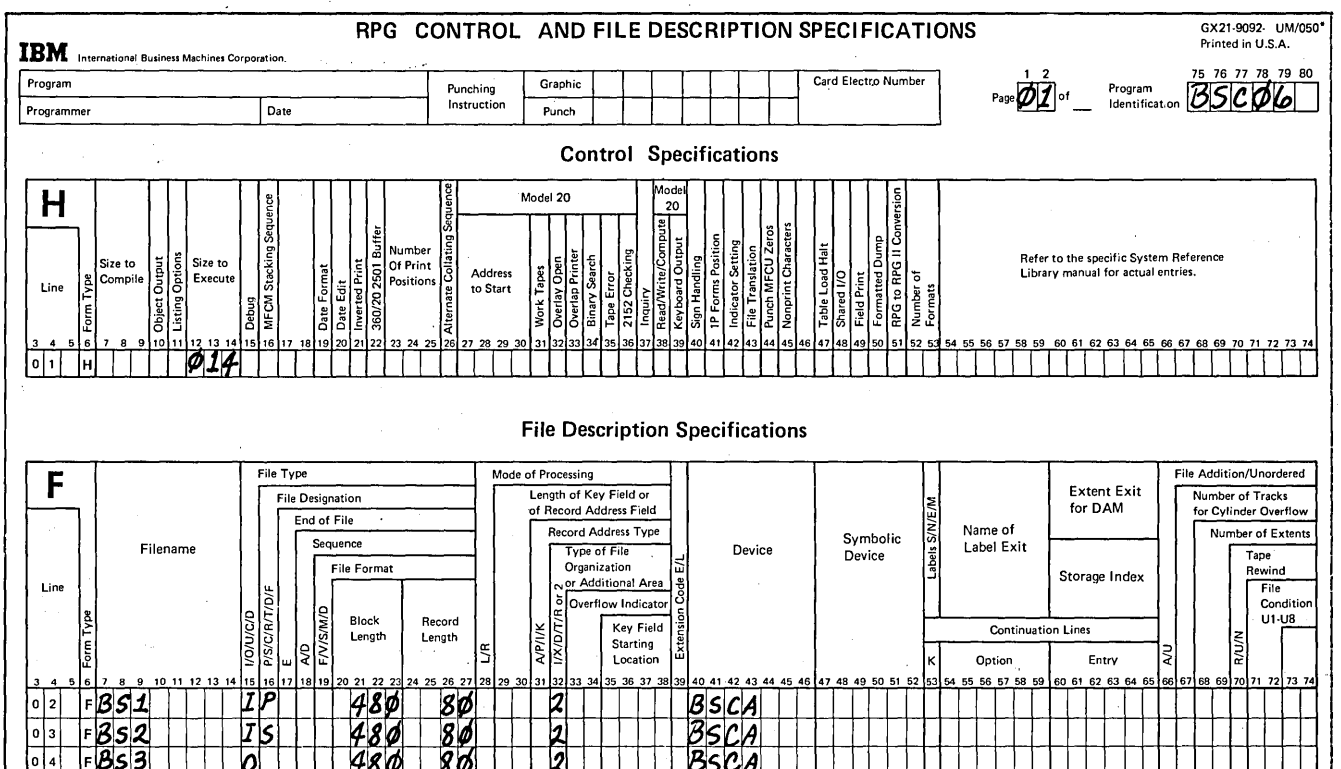


Figure 2-3 (Part 1 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

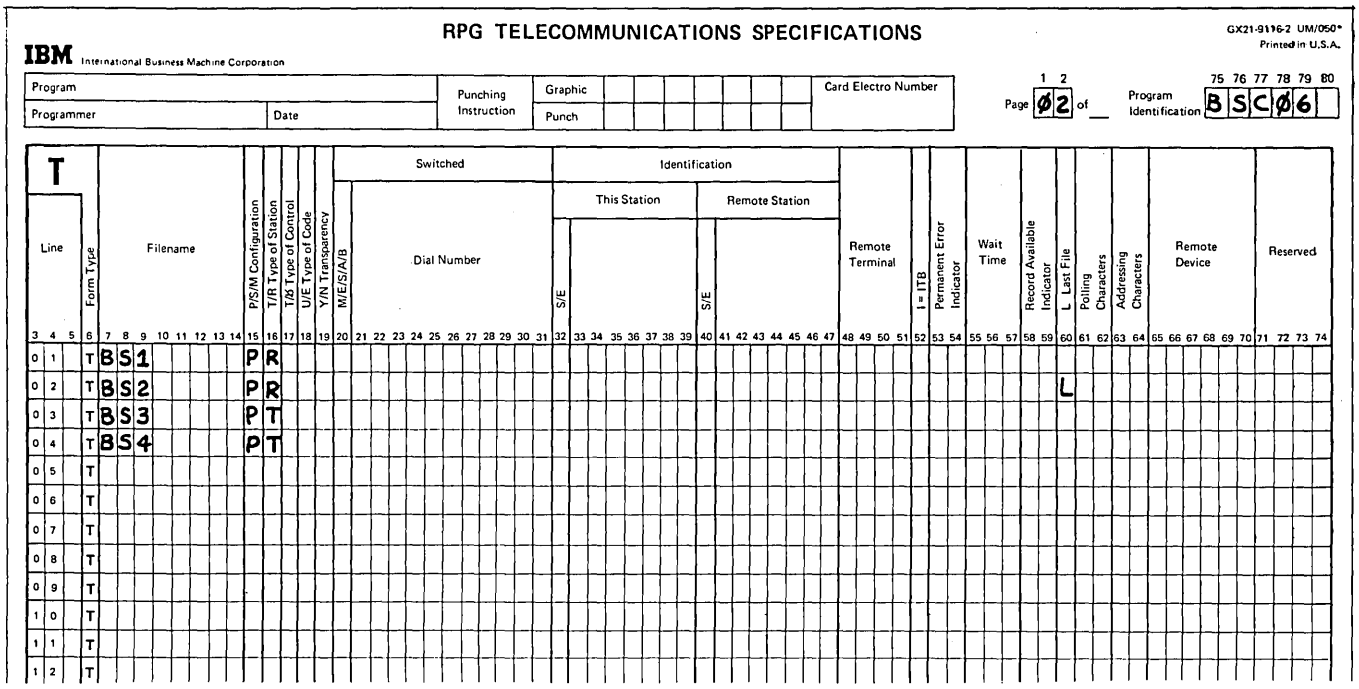


Figure 2-3 (Part 2 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

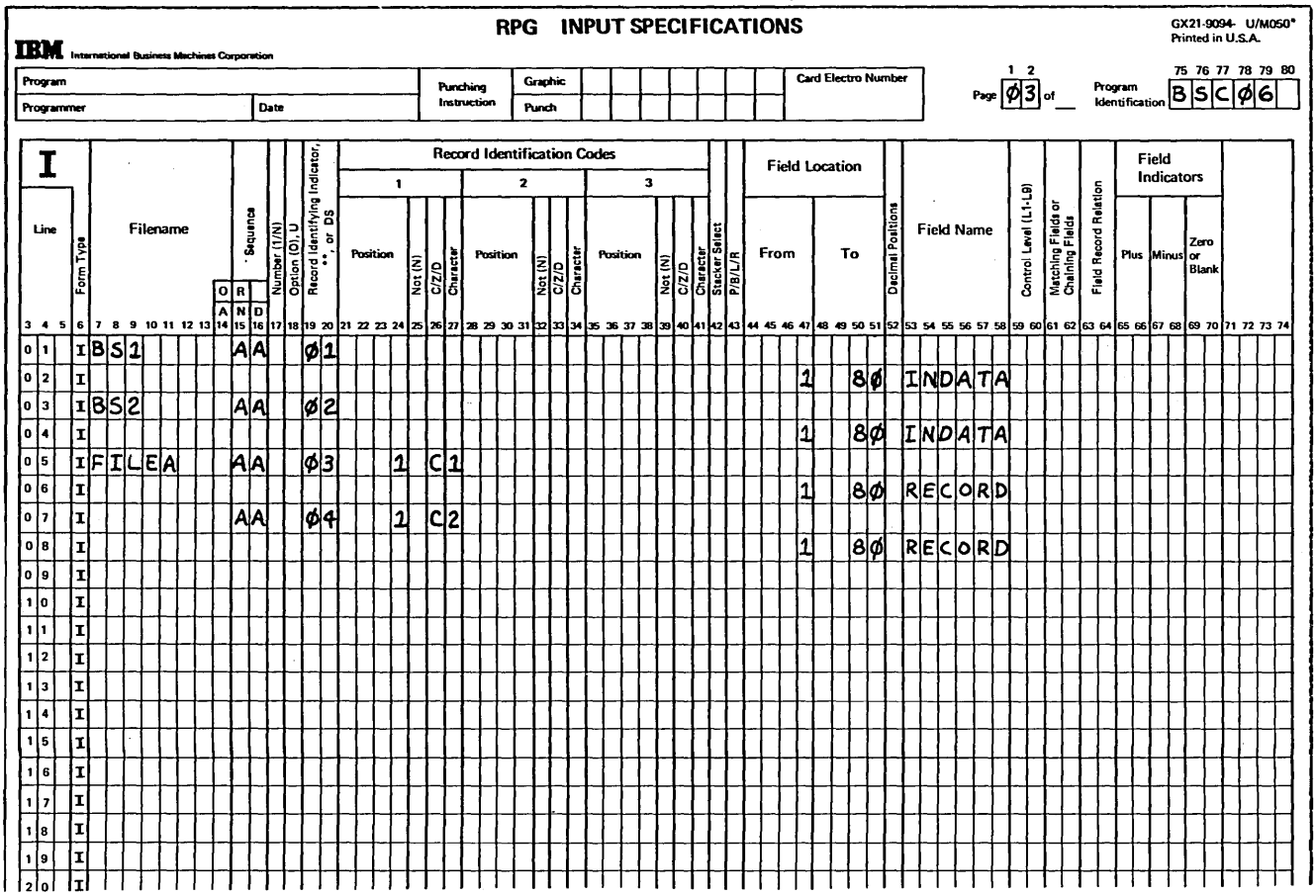


Figure 2-3 (Part 3 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

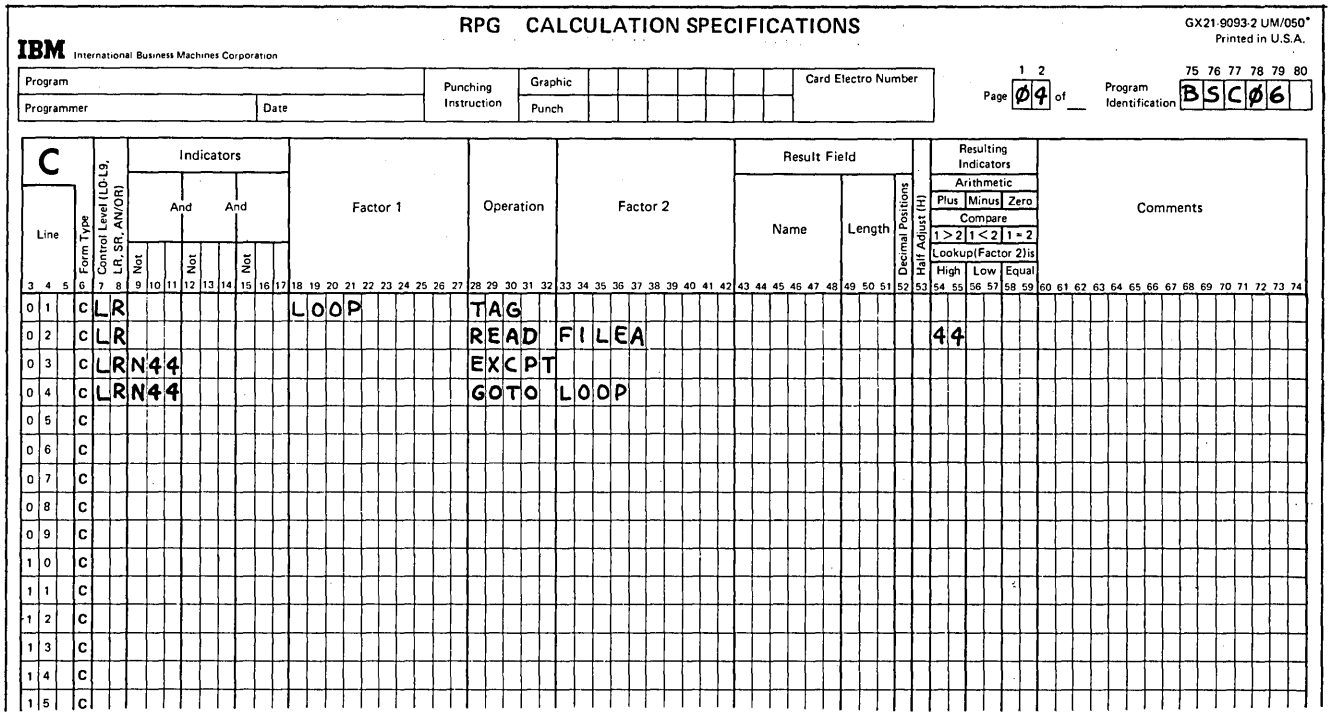


Figure 2-3 (Part 4 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

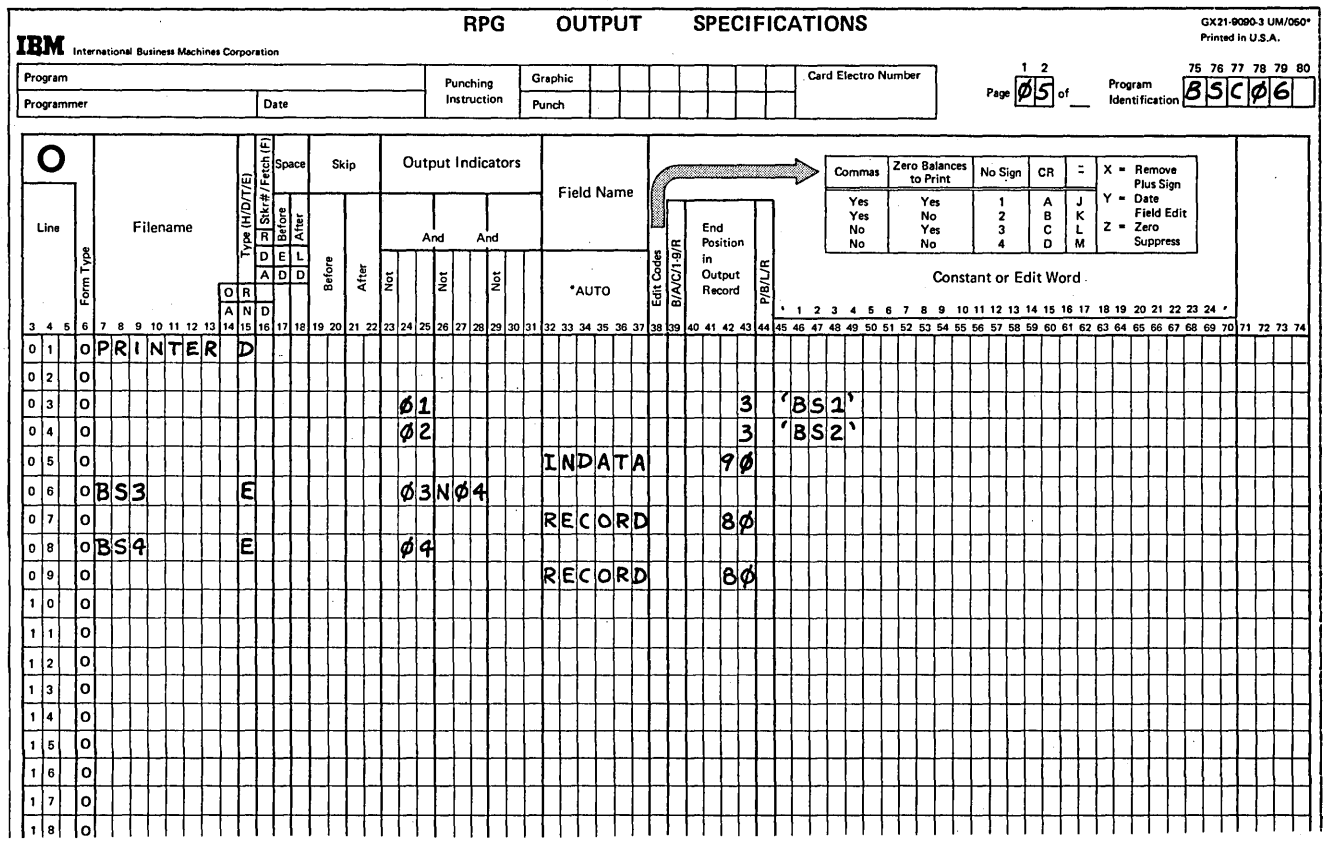


Figure 2-3 (Part 5 of 5). Receiving Two Files, then Transmitting Two Files to a 3740

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.

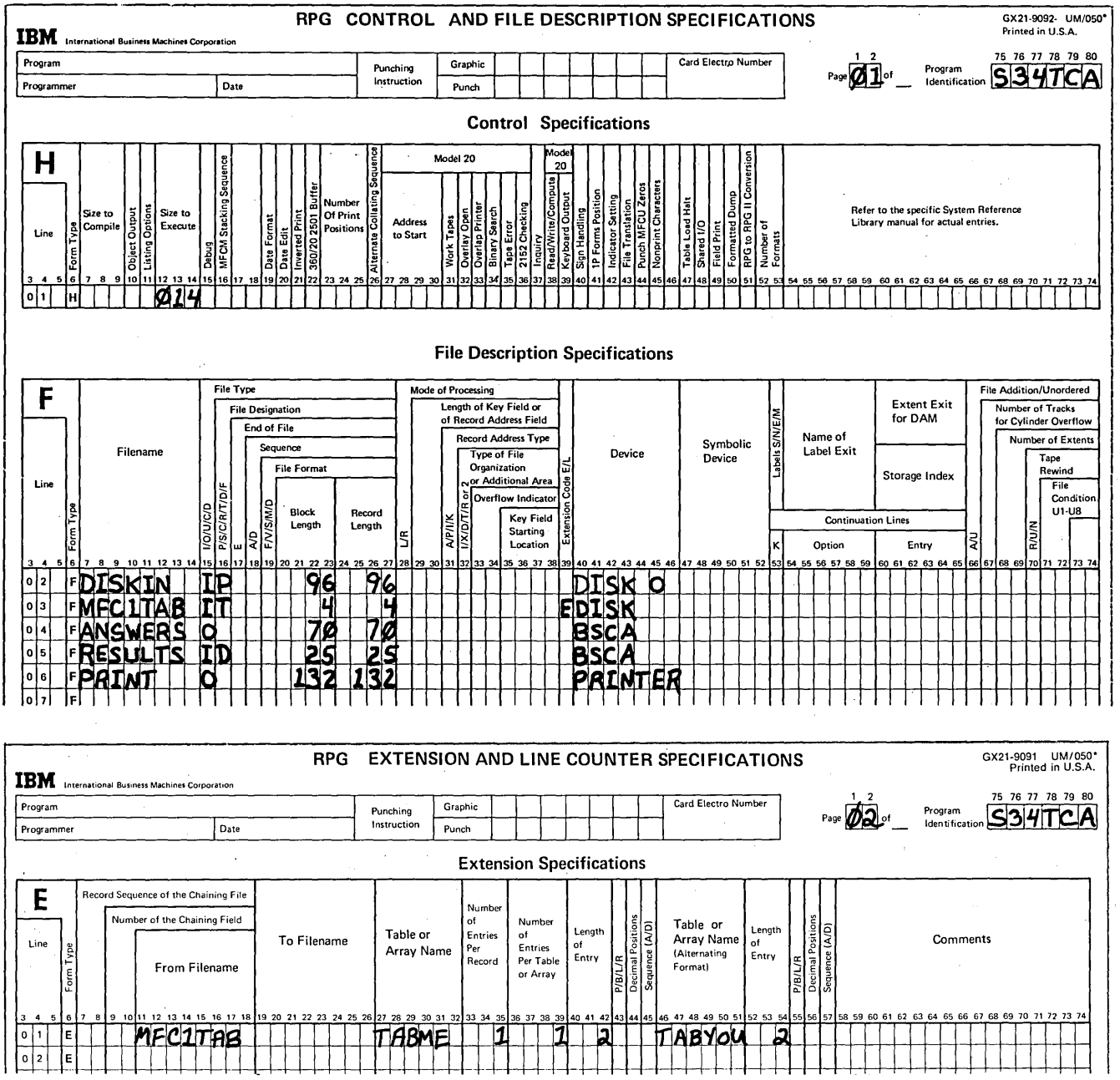


Figure 2-4 (Part 1 of 4). System/34 to TCAM

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

GX21-9116 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/S Type of Control	U/E Type of Code	Y/N Transparency	Switched: M/E/S/A/B	Switched: Dial Number	Identification: This Station S/E	Identification: This Station	Identification: Remote Station S/E	Identification: 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 (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	Field Indicators: Minus	Field Indicators: Zero or Blank
01	I	DISKIN	AA			01	1		C	X																				
02	I																				4	20		NAME						
03	I																				27	28		ØNUMBES						
04	I																				30	79		ANS						
05	I	RESULTS	BB			02	1		C	A																				
06	I																				2	18		NAME						
07	I																				19	20		ØNUM						
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.

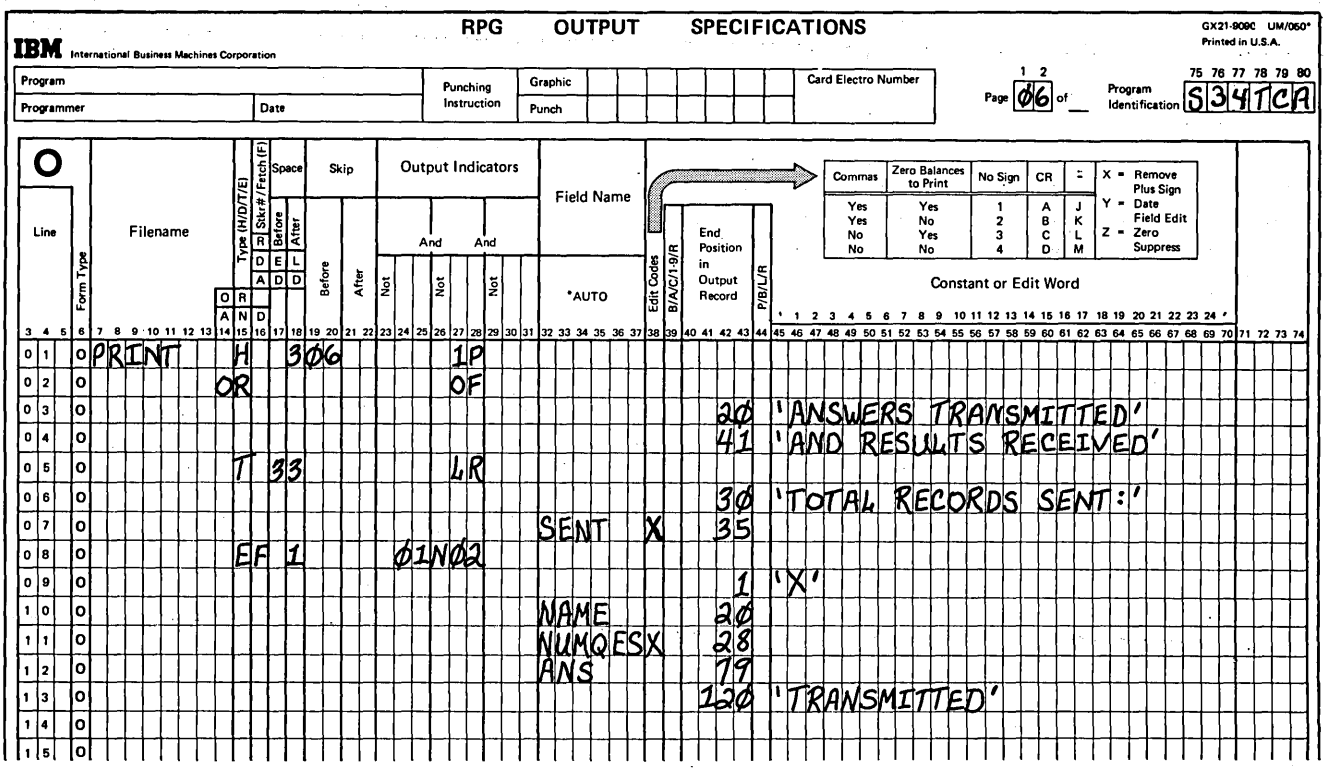
IBM International Business Machines Corporation

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 / Minus / Zero; Compare: 1>2 / 1<2 / 1=2; Lookup(Factor 2)is: High / Low / 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



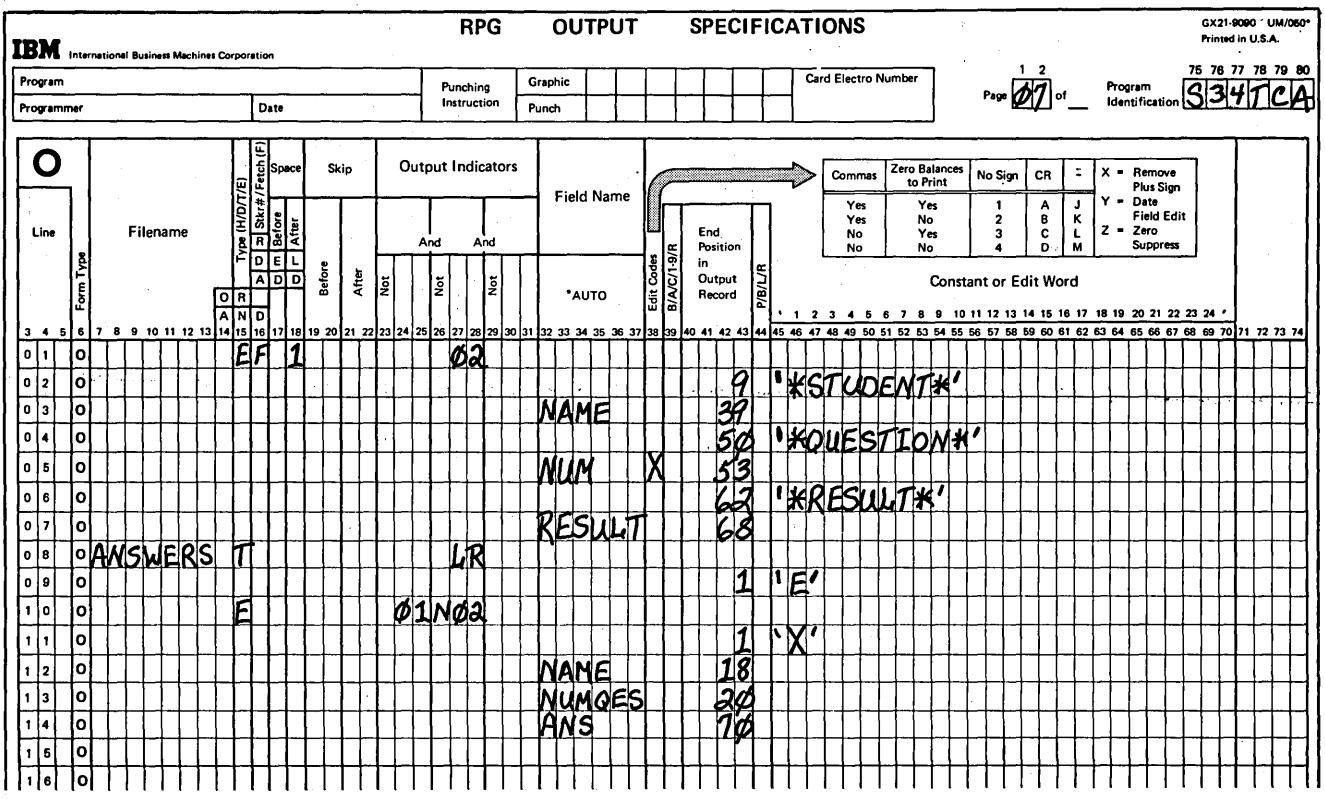


Figure 2-4 (Part 4 of 4). System/34 to TCAM

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

IBM International Business Machines Corporation

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

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

Programmer | Date

Page 01 of __ | Program Identification 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.

Line	Form Type	Size to Execute
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, Option, Entry | A/U | File Addition/Unordered | Number of Tracks for Cylinder Overflow | Number of Extents | Tape Rewind | File Condition U1-U8 | R/U/N

Line	Form Type	Filename	I/O/U/C/D	P/S/C/R/T/D/F	E	A/D	F/V/S/M/D	Block Length	Record Length	L/R	A/P/I/K	1/X/D/T/R or 2	Device
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												
09	F												

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

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.

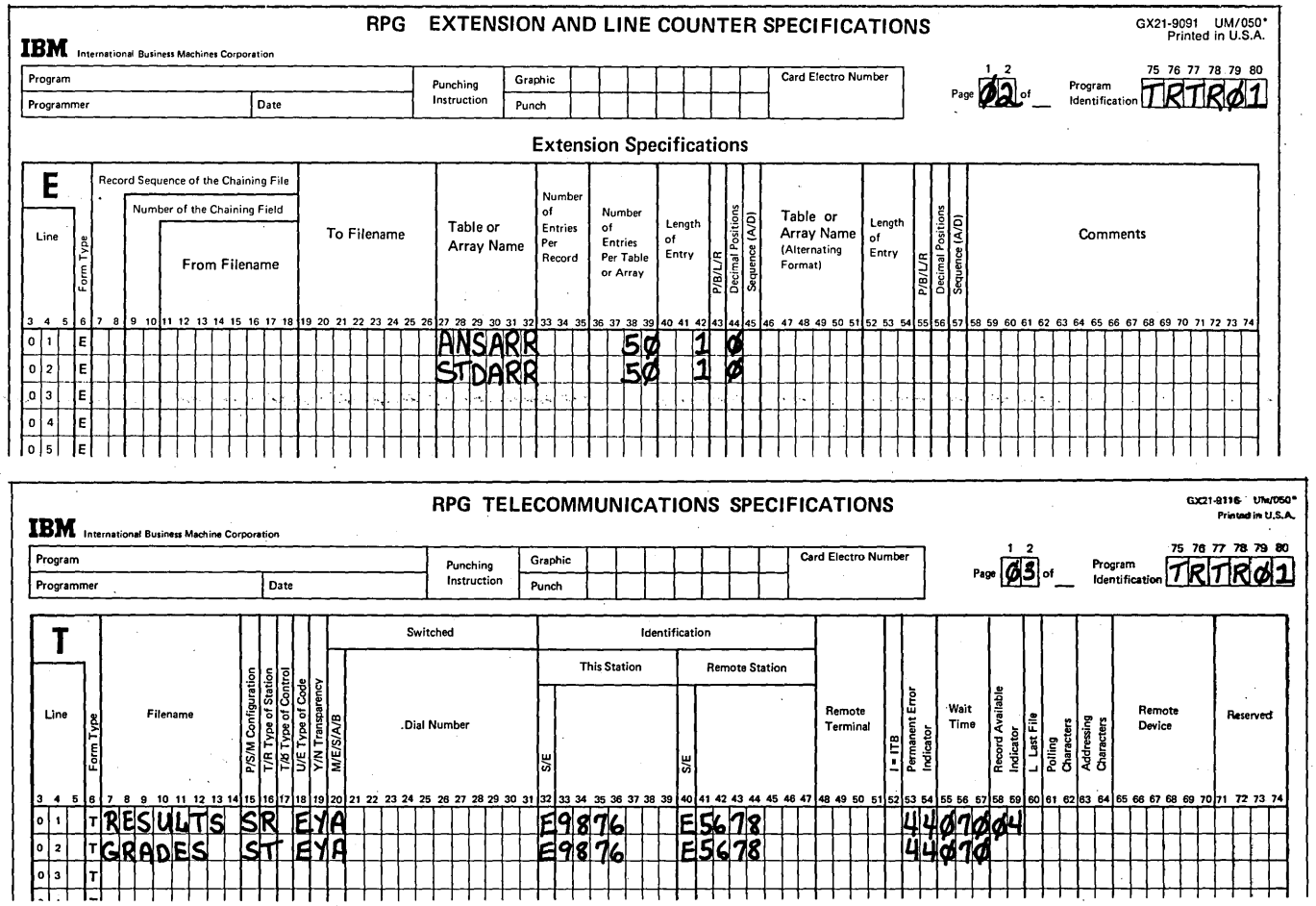
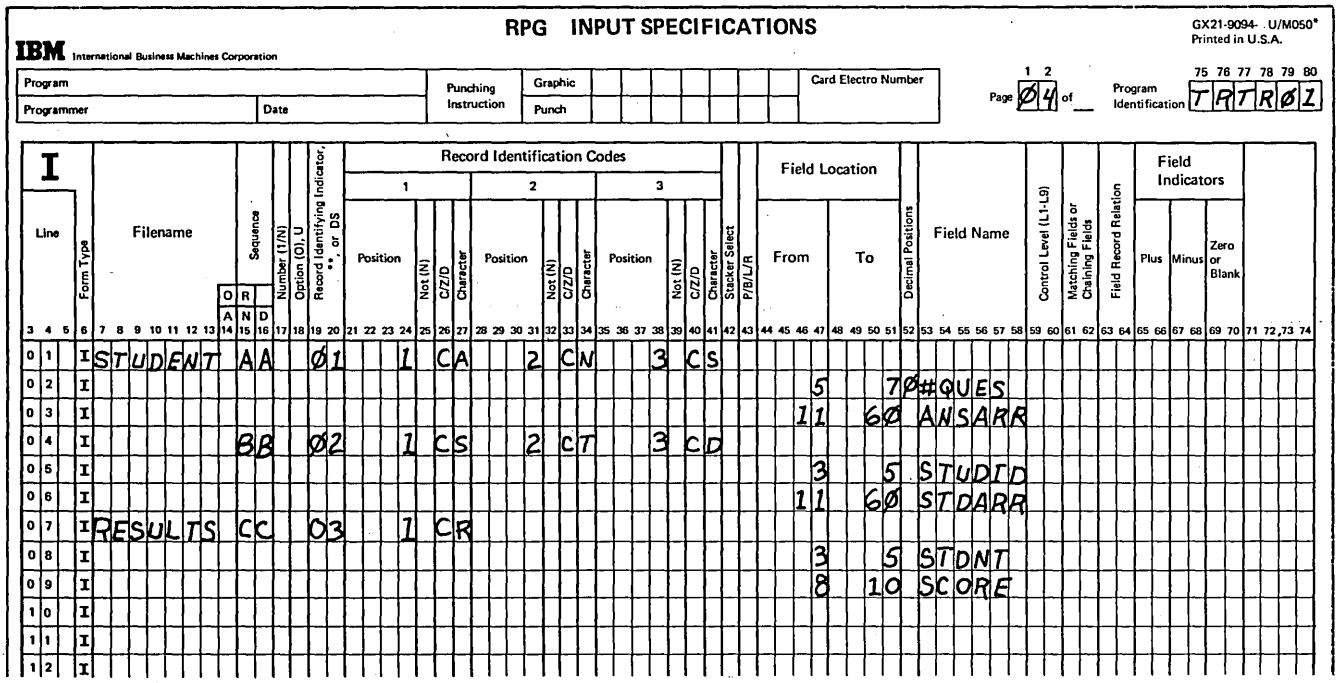


Figure 2-5 (Part 2 of 5). Transmit Interspersed with Receive

RPG Input Specifications (Figure 2-5, Part 3)

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 TRTR01

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	60		ANSARR						
04	I		BB			02	1		CS		2		CT		3		CD													
05	I																				3	5		STUDID						
06	I																				11	60		STDARR						
07	I	RESULTS	CC			03	1		CR																					
08	I																				3	5		STDNT						
09	I																				8	10		SCORE						
10	I																													
11	I																													
12	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

IBM International Business Machines Corporation

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

Program | Punching Instruction | Graphic Punch | Card Electro Number

Programmer | Date

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	LRN44		STDNT	COMP STUDID						090909	
12	C	LR	09		GOTO LOOP2							
13	C											
14	C											
15	C											
16	C											
17	C											
18	C											
19	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.

RPG OUTPUT SPECIFICATIONS

IBM International Business Machines Corporation — GX21-9090 UM/060* Printed in U.S.A.

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

Programmer | Date

Page Ø6 of __ | Program Identification TRTRØ1

Line	Form Type	Filename	Type (H/D/T/E)	Space	Skip	Output Indicators	Field Name	Edit Codes	End Position in Output Record	P/B/L/R	Constant or Edit Word
01	O	GRADES	D			Ø1N44					
02	O						#QUES		7		
03	O						ANSARR		6Ø		
04	O		D			Ø2N44					
05	O						STUDID		5		
06	O						STDARR		6Ø		
07	O	PRINT	T			44					
08	O								14		'ERROR OCCURRED'
09	O		E			Ø3N44					
10	O								13		'STUDENT #'
11	O						STDNT		17		
12	O								25		'SCORE ='
13	O						SCORE		29		
14	O										
15	O										
16	O										
17	O										
18	O										

Commas / Zero Balances to Print / No Sign / CR / -: Yes Yes 1 A J; Yes No 2 B K; No Yes 3 C L; No No 4 D M

X = Remove Plus Sign; Y = Date Field Edit; Z = Zero Suppress

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

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 6K bytes of nonswappable main storage to run. BSC programs cannot be run from the input job queue.

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

OTHER SYSTEMS

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

- Customer Information Control System (CICS/DOS/VS or CICS/VS)
- Information Management System (IMS/VS)
- 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
- 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)
- 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)
- Operating System or Disk Operating System Virtual Telecommunications Access Method (DOS/VS or OS/VS VTAM)

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 EOT (STX ETX if 3740 multiple file mode).
- $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 case of 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 $GET macroinstruction or by checking for X'42' ($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 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, which 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:

```
[name] $DTFB RECL–decdig,RCAD–address,BLKL–decdig,FTYP–{RCV|TSM} [,NAME–filename]
     [,BUFNO–{1|2}] [,ERRCT–decdig] [,RECSEP–number] [,TYPE–{PP|AA|MA|MC|MP}]
     [,CODE–{E|A}] [,UPSI–mask] [,CHAIN–address] [,ITB–{Y|N}]
     [,TRANSP–{Y|N}] [,RVIADR–address] [,RVIMSK–code] [,DLYCT–decdig]
     [,RCVID–address] [,RCVCT–decdig] [,SNDID–address] [,SNDCT–decdig]
     [,TERMAD–address]
```

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: Indicates 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	EBCDIC
X'00'	X'00'
X'01'	X'01'
X'02'	X'02'
X'03'	X'03'
X'04'	X'10'
X'05'	X'1F'
X'11'	X'26'
X'15'	X'2D'
X'16'	X'32'
X'17'	X'37'
X'1F'	X'3D'

TYPE: This operand 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.

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. The delay count is the number of seconds after receiving or transmitting a block of data that BSC will wait to receive or transmit another block of data for the same file. 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. When the delay count is exhausted, EOT is transmitted to the remote station and an error message is displayed for the user.

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.

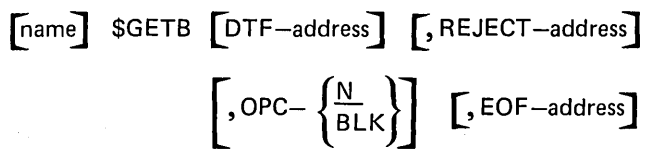
SNDCT: Specifies, in decimal, the length of the identification sequence required by the remote station. Length can be from 1 to 15. If 1 is specified, BSC transmits two characters–duplicates of the character addressed by the SNDID operand (preceding paragraph). SNDCT requires the SNDID operand.

TERMAD: Specifies the hexadecimal representation of the two-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.

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.

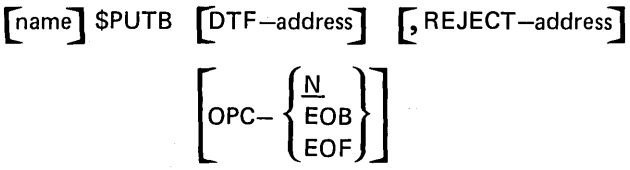
EOF: Specifies your end-of-file routine. If this operand is omitted, control is returned to the user program at the next sequential instruction after the $GETB.

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), then start of text/end of text (STX/ETX).

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:

[name] $TRTB [CODE–{E/A}] [,HEX–hex]

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

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 (3 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).

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.

Note: You cannot use blank truncation with ITB mode. You can specify blank truncation with transparent mode; however, this negates the truncation feature.

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.

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

The following 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

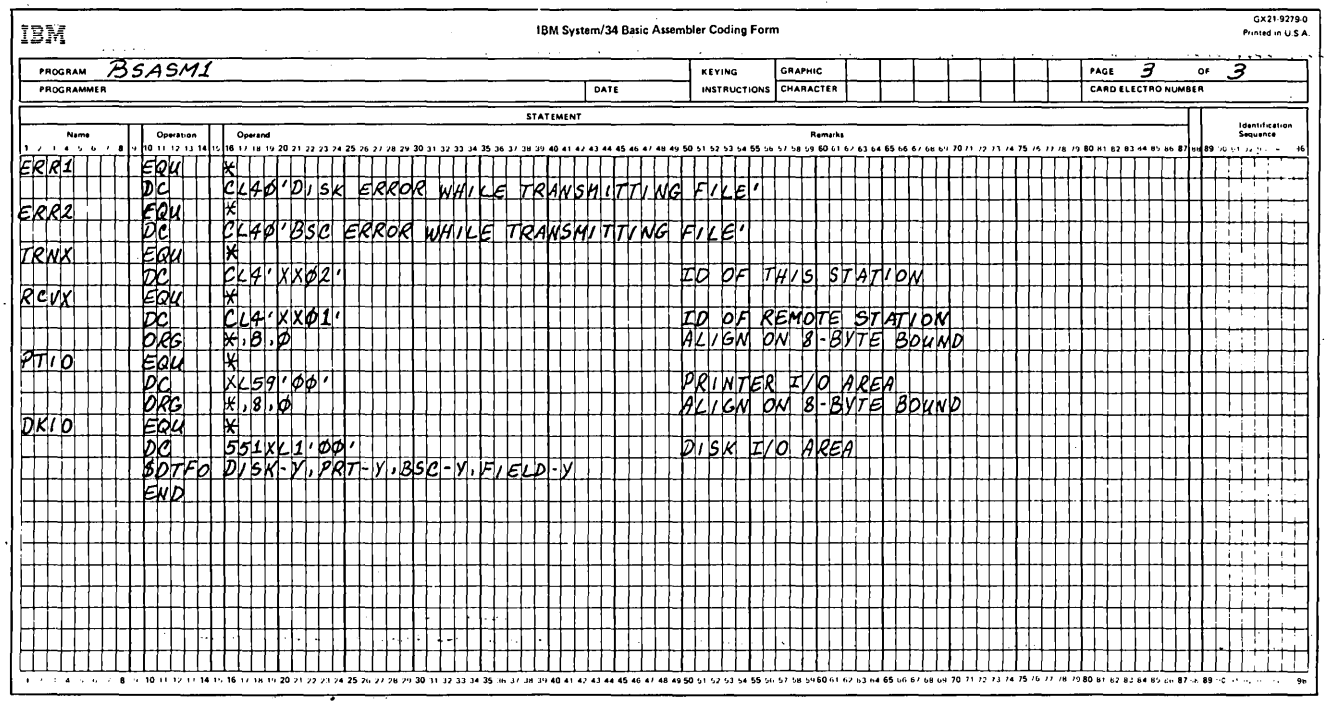
```
XMIT1    START 0
*
*   ALLOCATE AND OPEN CHAIN OF DTF'S
*
         $ALOC DTF-BSCDTF                ALLOCATE ALL DTF'S
         $OPEN DTF-BSCDTF                OPEN ALL DTF'S
*
*        TRANSMIT THE FILE
*
LOOP1    EQU   *
         $GETD DTF-DSKDTF,ERR-DSKERR,EOF-CLOSE   GET A RECORD
         L     DSKDTF+$F1WKB,4           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   $DTFB RECL-80,BLKL-80,RCAD-BUFFR1,FTYP-TSM,TYPE-MC,                C
               CHAIN-PRTDTF,RCVID-RCVX,RCVCT-4,SNOID-TRNX,SNDCT-4
BUFFR1   EQU   *
         DC    1X'80'
PRTDTF   $DTFP RCAD-PTBUF,IOAREA-PTIO,RECL-40,                              C
               PRINT-Y,CHAIN-DSKDTF,SPACEA-1,NAME-PRINT
DSKPTF   $DTFD ACCESS-CG,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'XX02'	ID OF THIS STATION
RCVX	EQU	*	
	DC	CL4'XX01'	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
	$DTFO	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
         $EOJ                            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

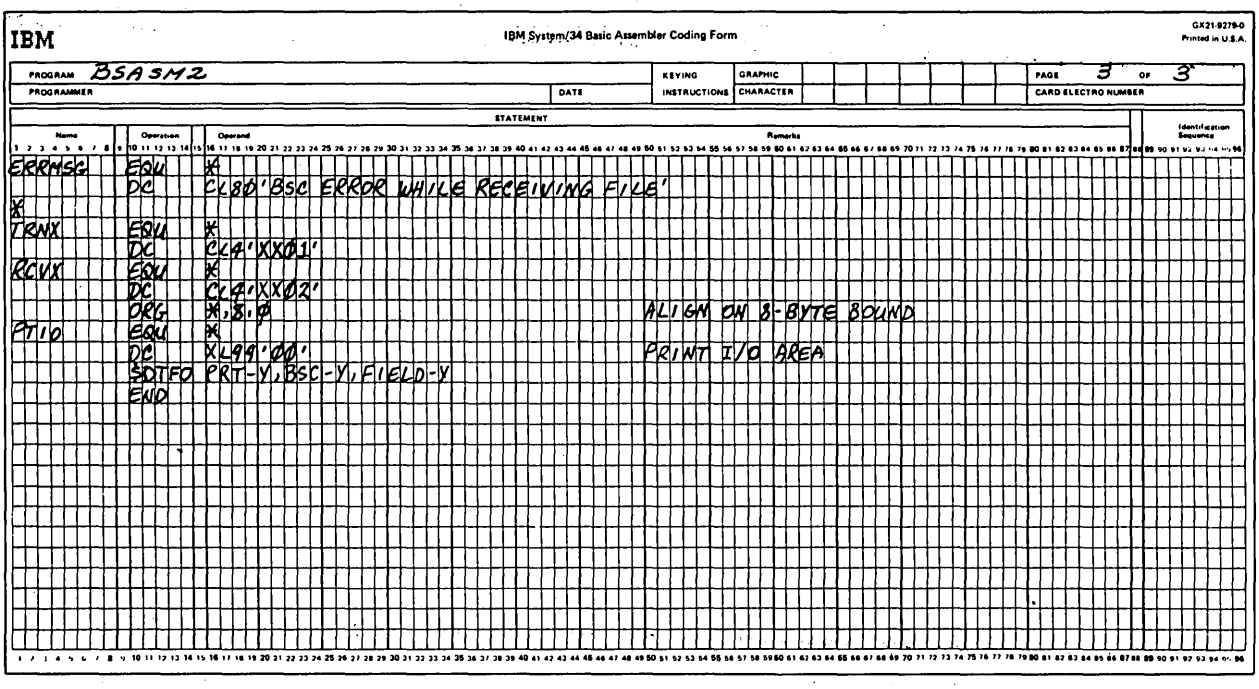


Figure 3-2 (Part 2 of 2). Receive Program

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). 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 — PROGRAMMER — DATE — KEYING INSTRUCTIONS — GRAPHIC CHARACTER — PAGE 1 OF 5 — CARD ELECTRO NUMBER

```
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 — PROGRAMMER — DATE — KEYING INSTRUCTIONS — GRAPHIC CHARACTER — PAGE 2 OF 5 — CARD ELECTRO NUMBER

```
*****************************************
*                                       *
*   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

GX21 9279 Printed in U.S.A.

PROGRAM BSASM 3 | PAGE 3 OF 5

```
*****************************************************
*                                                   *
*   PERFORM EOJ AND ERROR PROCESSING                *
*                                                   *
*****************************************************
DONE     EQU   *
         MVC   PTBUF+39(40),EOJMSG+39      MOVE GOOD MESSAGE
         J     PRINT                       JUMP TO PRINT
ERR1     EQU   *
         MVC   PTBUF+39(40),EBUF1+39       BSC ERROR RECV FILE 1
         J     PRINT                       JUMP TO PRINT
ERR2     EQU   *
         MVC   PTBUF+39(40),EBUF2+39       BSC ERROR RECV FILE 2
         J     PRINT
ERR3     EQU   *
         MVC   PTBUF+39(40),EBUF3+39       BSC ERROR XMIT FILE 1
         J     PRINT                       JUMP TO PRINT
ERR4     EQU   *
         MVC   PTBUF+39(40),EBUF4+39       BSC ERROR XMIT FILE 2
         J     PRINT                       JUMP TO PRINT
DSKERR   EQU   *
         MVC   PTBUF+39(40),DKMSG+39       DISK ERROR MESSAGE
PRINT    EQU   *
         $PUTP DTF-PTRDTF,SPACE-2          PRINT MESSAGE
```

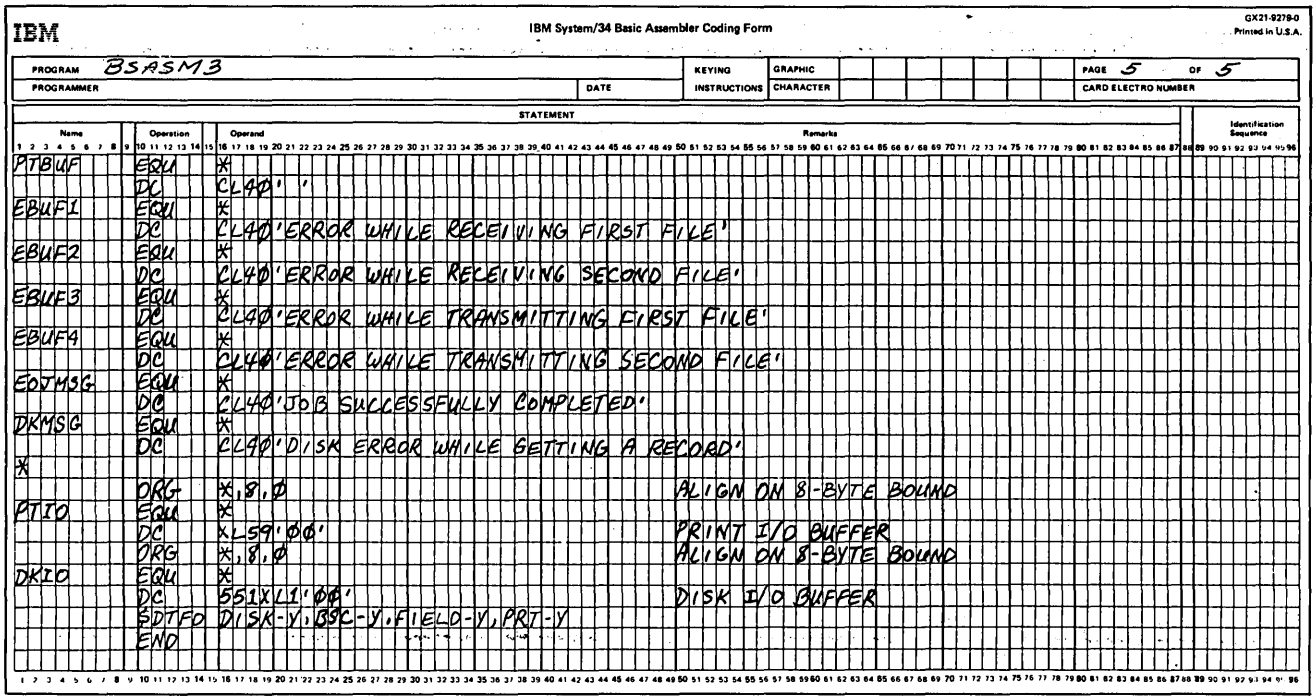
IBM System/34 Basic Assembler Coding Form

GX21 9279 Printed in U.S.A.

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-40,BLKL-40,FTYP-RCV,TYPE-MA,RCAD-PTBUF,CHAIN-BSDTF2
BSDTF2   $DTFB RECL-40,BLKL-40,FTYP-T5M,TYPE-MA,RCAD-DKBUF,CHAIN-BSDTF3
BSDTF3   $DTFB RECL-40,BLKL-40,FTYP-T5M,TYPE-MA,RCAD-DKBUF,CHAIN-DKDTF1
DKDTF1   $DTFD ACCESS-CG,RECL-40,NAME-BSFIL1,IOAREA-DKIO,                C
               BLKL-512,CHAIN-DKDTF2
DKDTF2   $DTFD ACCESS-CG,RECL-40,NAME-BSFIL2,IOAREA-DKIO,                C
               BLKL-512,CHAIN-PTRDTF
PTRDTF   $DTFP RCAD-PTBUF,IOAREA-PTIO,PRINT-Y,RECL-40,NAME-PRINT
*
DKBUF    EQU   *
         DC    CL40' '
```

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



IBM System/34 Basic Assembler Coding Form

GX21-9278-0 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	
	SDTFD	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

The System/34 system support program product includes the MRJE (MULTI-LEAVING Remote Job Entry) utility.

MRJE supports communications 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

The preceding host systems support a line discipline called MULTI-LEAVING. MULTI-LEAVING is fully synchronized, two-directional transmission of a variable number of data streams between two computers using BSC facilities. MULTI-LEAVING permits maximum overlapping of input and output operations in the MRJE utility and the host system by mixing input and output data streams on the communications lines. Communication is in EBCDIC and text transparency can be used. Text transparency allows BSC to send and receive messages containing any of the 256 possible combinations of eight bits.

The MRJE utility can run with either attended operation or automatic operation. For attended operation, an operator must be present while the utility is running. This is the normal method of operation for the System/34 and this utility.

For automatic operation, an operator need not be present because the MRJE utility selects the default option for any MRJE operator attention messages that might occur. When a default option is available for a message, the default is described in the *Displayed Messages Guide*. For some other System/34 SSP error conditions, operator action is required.

When using the automatic mode, the host system operator might be required to control both the line connect and disconnect operations if you have a switched line.

STORAGE REQUIREMENTS

The MRJE utility requires a minimum storage capacity of 16K bytes with a 14K nucleus. This minimum configuration includes one printer, one reader (the system console), and no punch. This configuration also runs with 200-byte buffers and without full compression of reader input. If you use a 16K system nucleus on a 32K system, no other jobs can run because MRJE requires 4K bytes of nonswappable storage. This reduces the available user area to 12K, which is insufficient to initiate another job.

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

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

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

2. Compute the nonswappable storage size:
   a. Multiply the number of buffers[1] by the buffer size (rounded up to a multiple of 8 bytes).
   b. Add 3,296 bytes for the BSC task (required).
   c. Round up to the next multiple of 2,048 and divide by 1,024.

3. Add the swappable and nonswappable storage sizes to give the region size in K-bytes for the REGION OCL statement.

Example 1:

Minimum MRJE configuration–one reader, no punch, no full compression, and four 200-byte buffers.

1. From the chart, swappable storage is 12K bytes.

2. Nonswappable storage is 200*4 + 3,296 = 4,096 bytes. 4,096/1,024 = 4K bytes.

3. 12K + 4K = 16K bytes.

A region size of 16K is needed for the minimum configuration.

Example 2:

Full compression, a punch, three readers, and eight 344-byte buffers.

1. From the chart, swappable storage is 20K bytes.

2. Nonswappable storage is 344*8 + 3,296 = 6,048 rounded up to 6,144 bytes. 6,144/1,024 = 6K bytes.

3. 20K + 6K = 26K bytes.

A region size of 26K is needed for this configuration.

NETWORK CONTROL

The System/34 MRJE utility is always considered the remote station and must initiate the data transmission to the host system. In a point-to-point nonswitched network, this is accomplished by enabling the communications adapter on the System/34 and sending the sign-on command to the host system. In a point-to-point switched network, the system operator calls the host system by manually dialing the host system. When automatic answer is specified, a call initiated by the host system operator is answered automatically if the modem at the System/34 is in automatic answer mode. When manual call is specified, a displayed message directs the operator to manually dial the host system. (When MRJE is in automatic mode, no messages are issued to the operator to answer a call.) When the line connection is made, the sign-on command is sent to the host system. Once communications are established between the MRJE utility and the host system, line control is identical for the switched and nonswitched networks.

The following points must be considered when defining the System/34 MRJE utility to the host system:

- 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 or 1 punch task
- All host systems must define a System/34 as a System/3 work station with console support.
- If text transparency is used, the transparency capability must be specified and active in both the host system and the System/34.
- The System/34 and host system must have compatible modems.
- The host system must support multiple readers if more than one display station is used as a reader.

[1]The minimum number of buffers is equal to the number of print, reader, and punch tasks plus two. MRJE allocates as many buffers as possible within the region (up to nine).

MRJE SESSION

Functions of the MRJE utility are:

- *Initiation*: Loads the MRJE utility.
- *Initialization*: Defines the MRJE utility capabilities and establishes a network link with the host system.
- *Input*: Controls the MRJE utility and sends jobs, data, and host control information to the host system for processing.
- *Output*: Provides operator communication and receives output from the host system.
- *Termination*: Ends the network link with the host system.

Initiation

The MRJE utility can be loaded by entering the MRJE procedure command only from the System/34 console.

MRJE procedure command format:

$$\text{MRJE}\ \left[\text{DISPLAY}-\frac{\#\text{MR01}}{\text{name}}\right]\left[,\text{AUTO}-\frac{\text{N}}{\text{Y}}\right]$$

Parameters:

DISPLAY: Specifies the name of the initialization format for the console display screen. If the DISPLAY parameter is not specified, #MR01 is the default.

#MR01 is the name of the console display screen initialization format that is supplied with the MRJE utility. The #MR01 format has all the values set to communicate with a HASP II host system.

You can specify the name of a display screen initialization format that you created using the $SFGR–screen format generator program. (For information on creating your own initialization display screen format, see *Initialization Parameters*.)

AUTO: Specifies the mode of operation of the MRJE utility.

N indicates attended operation. An operator must be present while MRJE is executing. If the AUTO parameter is not specified, N is assumed.

Y specifies automatic operation. MRJE selects the default options for MRJE operator attention messages that occur. An operator need not be present while MRJE is executing. If any MRJE utility errors occur, operator action is not required; however, system errors require operator action. Dialing the host system might not be required because a call initiated by the host system operator is automatically answered if the modem is in automatic answer mode. When using automatic mode, the host system operator might be required to control both the line connect and disconnect operations.

The MRJE procedure is an MRT procedure (and an NEP) and contains the following OCL:

```
// MEMBER PROGRAM1-#MR#M1
// LOAD $MRJE
// RUN
```

Initialization

After the MRJE utility is loaded, it is initialized through the System/34 console.

The console display screen is formatted with either the MRJE-supplied format (#MR01) or a user-written format. You can change the information, in either format, by positioning the cursor under the value to be changed and keying the change. Press the Enter/Rec Adv key when all changes have been made. To use the initialization information as is, press the Enter/Rec Adv key. This format information is used to initialize and assign the functions of the MRJE utility and to establish the network link with the host system.

If automatic mode was specified during the loading of MRJE, the MRJE session is initialized with the format specified on the MRJE procedure command; the operator cannot modify the screen information.

The following illustration shows the initialization display that is supplied as part of the MRJE utility. This display shows the options that may be used to initialize the MRJE utility.

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


ENTER INITIAL CONFIGURATION 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     RD1C,D - Y
RD1
ENTER INITIAL CARRIAGE 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

ENTER LOGON OR SIGNON INFORMATION BELOW:
/*SIGNON

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

The initial configuration information consists of parameters. Each parameter contains a keyword and a value. In the illustration, HOST is a keyword and H is its value, indicating that the MRJE utility will communicate with a HASP II host system. All parameters with values assigned are default values. Default values appear on the MRJE initialization display from either the MRJE-supplied or user-supplied initialization format. You can change the default values shown, either during initialization or through MRJE utility control statements during the session. In the same way, you can also assign values where none are shown.

HOST: Specifies the host system for this session.

- H–HASP II under OS/VS2
- 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 decimal, of the MRJE utility BSC buffers in bytes. The length must be equal to the BSC buffer length defined by the host system.

The following values are the host defaults:

- HASP II–400
- ASP–400
- RES–344
- JES2–512
- JES3–512
- VM/370 RSCS–400

XPC: Specifies whether text transparency is used.
N indicates text transparency is not used.
Y indicates text transparency is used.

COM: Specifies the compression of duplicate characters for transmitting input to the host system. C indicates full duplicate character compression (compresses consecutive duplicate characters) for this session. N indicates no compression for this session.

Note: The COM options do not apply to data streams received from the host system. Data from the host system always has full duplicate character compression.

FSN: Specifies an identifying number for the first special forms file. File sequence numbers can be up to three digits long using digits 0 through 9. (Leading zeros are not required.) The program uses the three-digit number that you specify to begin generating file labels for the files. The MRJE utility increases the number by one for each new label.

Note: If there already is a file label with the current FSN assigned, the MRJE utility assigns the next number that is available.

PR1: Assigns the print task. PR1-P assigns the print task to the printer. PR1-D assigns the print task to the disk. PR1-N does not assign the print task. If N is selected, the print task can be assigned later to the printer or disk by a MODIFY utility control statement entered from the console input task.

PU1: Assigns the punch task. PU1-D assigns the punch task to the disk. PU1-N does not assign the punch task. If PU1-N is selected, the punch task cannot be used during the session. If the host system attempts to send punch output, the output cannot be received by the MRJE utility. When the session is ended, the MRJE utility terminates abnormally unless punch output is reallocated at the host system. (See the host system operator's guide for further information.)

DID: Specifies the identification character associated with individual 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 in an individual special forms disk file. If the characters are not equal, a message is issued for operator control.

RD1-K: Assigns the reader task to the system console keyboard. The keyboard is always treated as a command file. The RD1 parameter cannot be specified if the RD1NAME parameter is specified. If RD1 is not specified, the system console keyboard is assigned to the console input task.

RDN: Specifies the number of reader tasks (1, 2, or 3) that the MRJE utility is to allocate. The number must conform to the number of readers allocated at the host system.

RD1NAME: Specifies the name of the disk file or library member to be read by the reader task. The RD1NAME parameter cannot be specified if the RD1-K parameter is specified.

RD1DATE: Specifies the creation date of the disk file specified in the RD1NAME parameter. The RD1DATE parameter is required only if more than one file with the specified name exists on the disk. The date must be entered in the system date format.

RD1TYPE: Specifies the type of file specified by the RD1NAME parameter. RD1TYPE-D indicates a disk file. RD1TYPE-S indicates a source member in the system library. RD1TYPE-P indicates a procedure member in the system library.

RD1CMD: Specifies whether the file named in the RD1NAME parameter is a command file or a data file. RD1CMD-Y indicates the RD1NAME file is a command file. RD1CMD-N indicates the RD1NAME file is a data file.

SPCPR1: Specifies the maximum number of blocks on 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 directed to the standard forms file, TDISKPR1. The number of blocks can be up to four decimal digits. Unused space is returned to the system when MRJE terminates.

F: Specifies the initial output forms number for MRJE. The forms number can be any string of four alphameric characters. STD. is the standard MRJE forms number. If you do not wish to receive forms mount messages for each different forms number, use the $DCFUP utility to create a forms control table.

Note: The RES standard forms number is 4 blanks, but MRJE changes this to STD. for operator convenience.

FCTNAME: Specifies the name of the forms control table built by $DCFUP.

The carriage information is used to set the printer forms length and to correlate channel control characters with form line numbers. Complete carriage information is found in the CARRIAGE utility control statement.

The sign-on command must conform to the format and requirements of the host system.

The MRJE utility accepts the following sign-on commands according to the host system specified in the HOST parameter during initialization:

Host System	**Command Entered**
ASP, HASP, JES2, JES3, and RSCS	/*SIGNON
ASP and JES2	//*SIGNON
RES	LOGON

MRJE does not check the sign-on command for errors other than for having the correct sign-on or log-on keyword for the host specified. Refer to the host system publications for information on the appropriate command and the required parameters.

For a point-to-point switched network, a message requiring system operator action is displayed after MRJE is ready to send the sign-on command. An informational message is displayed when communications are established. This message terminates the initialization process, and the MRJE utility is ready to start processing.

For a point-to-point nonswitched network, an informational message is displayed when communications are established. This message terminates the initialization process, and the MRJE utility is ready to start processing.

Also on the display is an area where the System/34 operator can enter a message up to 160 characters long. The message appears on the status display of any System/34 display station attaching to the MRJE utility.

Instead of entering changes on the initialization display each time you initiate MRJE, you can modify the display screen format so that it contains your default values. The SSP includes the MRJE display screen source module, #MR@01 containing #MR01. To modify the values (and thereby change the defaults), change the constant data (columns 57 through 80) to the values you desire, and run $SFGR to add or update it into the MRJE object module #MR@M1. ($SFGR is described in the *System Support Reference Manual.*) (Only add or update the #MR@01 load module because it contains other formats required by the MRJE utility.) Then each time you initiate MRJE, your values are the defaults, and you can run in automatic mode or just press the Enter/Rec Adv key to continue.

Note: The name given to the format when it is placed in the #MR@M1 object module should not be #MR02 through #MR09, as these are reserved for other MRJE formats. However, any other name, including #MR01, can be used. If #MR01 is used as the name, then the IBM-supplied default format is replaced by your format. Up to 23 additional formats can be added to the #MR@M1 object module.

Input

Input is read by the MRJE utility and is either processed by the System/34 or transmitted to the host system. Input can come from either the disk or the keyboard. The keyboard can be at the system console or a display station. Input can consist of the following forms:

Input Task	Input Device	Input
Reader task	Disk • Data file • Source member • Procedure member Keyboard at a display station	Utility control statements Jobs
Console input task	Keyboard at the system console	Host system commands Utility control statements

MRJE utility control statements can be entered through the reader task or the console input task. Jobs for transmission to the host system can be entered only through the reader task.

Reader Input

Display stations (not system console) can be used as readers by specifying the RDN-n parameter during initialization. Both the disk and the keyboard can be used by the reader task.

When the disk is used as input for the reader task, the name of the disk file or library member must be specified during initialization or on the READFILE utility control statement.

Note: Disk files used by the reader cannot be shared with other programs.

The RD1CMD parameter that is specified during initialization or the CMD parameter that is specified on the READFILE utility control statement indicates how reader input is treated. Reader input is treated either as a data file or as a command file.

Data files contain records that are transmitted to the host system. These records can have any record length. A data file can contain JCL or data. (Any utility control statements in a data file are treated as data and are transmitted to the host system.)

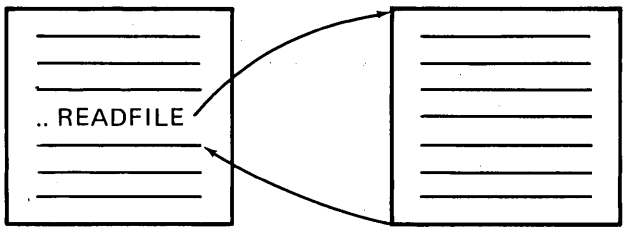
The MRJE utility transmits the information from data files as 80-byte records, regardless of the logical record length. For example, a data file containing four 60-byte logical records is transmitted as three 80-byte physical records. A data file containing four 120-byte logical records is transmitted 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.

Command files contain records that are transmitted to the host system and may contain utility control statements that are executed by the MRJE utility. A command file can contain any of the following:

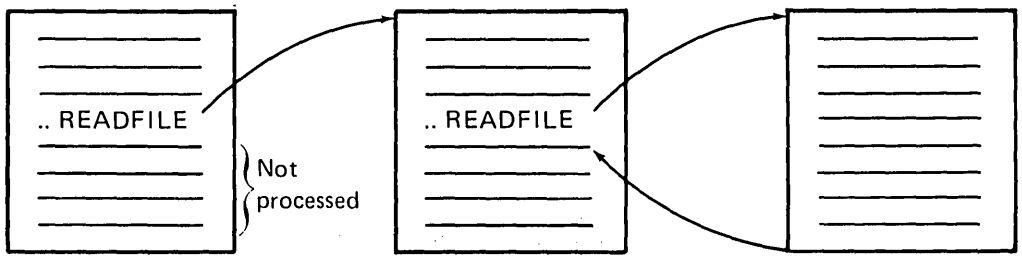
- Data
- JCL
- Utility control statements

If the command file contains utility control statements (READFILE, END, and EOF), the MRJE utility executes them instead of transmitting them to the host system. JCL statements and all data in the command file are transmitted as input to the host system. Command files must have a record length of 80 bytes.

If a READFILE utility control statement specifying a data file is encountered in a command file, reading of the command file suspends and reading resumes at the beginning of the disk file, procedure member, or source member specified by the READFILE utility control statement. When data file reading is complete, the reader returns to the command file and begins reading the next sequential statement.



If the second disk file or library member is specified as a command file, and a READFILE utility control statement for a data file is encountered, reading the second command file suspends, then reading resumes at the beginning of the third disk file or library member. When data file reading is complete, the reader returns to the second command file and begins reading the next sequential statement. When the second command file reading is complete, the reader does *not* return to the first command file.



Assigning the Disk as the Reader: Disk is assigned for reader input if the initialization parameter RD1NAME is used or if a READFILE utility control statement is used. The RD1NAME parameter and the READFILE utility control statement describe the disk files, procedure members, and source members from which data is read.

All disk input is read sequentially by the MRJE utility and accessed consecutively, not by logical records or blocks. Data is transmitted as 80-byte records to the host system.

Disk files can have either sequential, indexed, or direct file organization.

Disk input can consist of the following:

- Data
- JCL
- Utility control statements (command files only)

Assigning the Keyboard as the Reader: The system console keyboard is assigned for reader input by specifying the RD1-K parameter during initialization or on the MODIFY utility control statement.

When a display station operator enters the MRJE procedure command and a reader task is available, that display station is assigned as a reader and the screen appears as follows:

```
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



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

LOGON/SIGNON INFORMATION
XXXXXXXX


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

The x values in this display indicate the initial status of the MRJE configuration. After pressing the Enter/Rec Adv key, the screen appears as follows:

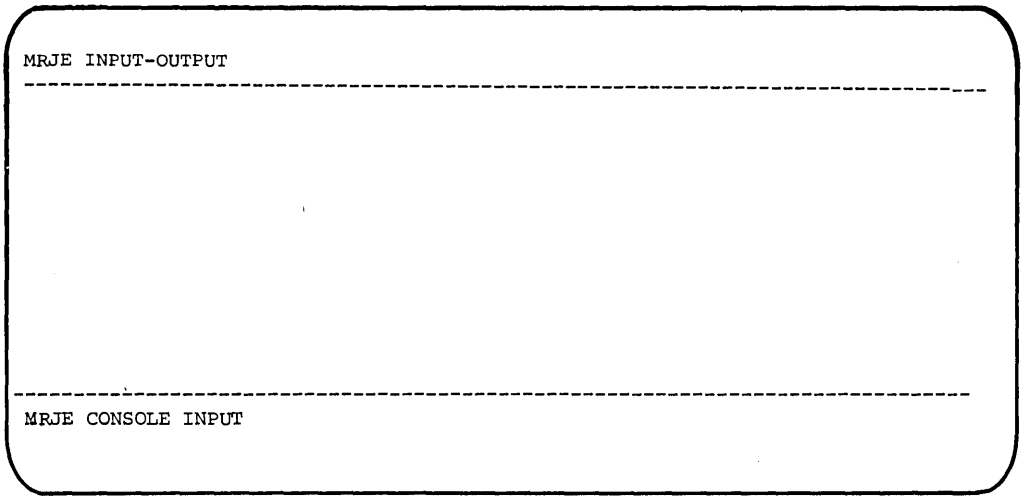
```
MRJE INPUT-OUTPUT
--------------------------------------------------------------------------------









--------------------------------------------------------------------------------
MRJE READER INPUT
```

This screen indicates that utility control statements, JCL statements, 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, and all input is treated as a command file.

After each record is entered, the operator must press the Enter/Rec Adv key to have the record read. The reader task is deactivated when an EOF utility control statement is entered or an end of file occurs in a disk command file. For a display station other than the system console, the Enter/Rec Adv key must be pressed again to return to the command display after the reader task has terminated.

Console Input

Input to the console input task can be entered only through the system console. In console input mode the display screen will appear as follows:



The space between the dashed lines contains host and MRJE messages, which can be scrolled.

Console input can consist of MRJE utility control statements or host system commands. The MRJE utility control statements can be READFILE, MODIFY, END, CARRIAGE, and CANCEL. In console input mode, up to 120 bytes of data can be entered per record. To indicate the end of each record the operator must press the Enter/Rec Adv key.

While the reader task is active on the system console, console input commands can be entered in either of two ways:

- The operator keys the console input command into the reader input field and presses command key 1. The console input task receives the command.
- The operator presses command key 1 without entering any data. The console input field replaces the reader input field, and the operator enters the console input command. After the command is processed, the reader input field reappears (unless the command was a CANCEL command for the console reader).

When a READFILE statement is entered through the system console reader task, the console input task is automatically activated and the console input field replaces the reader input field. The console input task remains active until the file is read. After the end of file is read for a *data* file (a message appears on the console display screen), the reader input field reappears when the Enter/Rec Adv key is pressed. The console input task remains active for a *command* file because the reader task is deactivated when an end of file occurs.

Output

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

- 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 generated by the MRJE utility (console output task)

Print Output

Printer output can be printed or written in a disk file. The print task processes all printer data from the host system and writes records to the printer or to the disk for subsequent processing. If spooling is active, printer data will be spooled to disk, when PR1-P is set.

Printer forms operations are controlled by channel control characters and the CARRIAGE utility control statement. The first character of each record is a channel control character that controls forms movement. The channel control character is not printed; only the 132-character portion of the record is printed. Data that is printed on the printer is not written to disk.

The channel control character has the following bit meanings:

```
        O M C C C C C C
bit-    0 1 2 3 4 5 6 7
```

bit meanings:

```
      O = 1 (must always be 1)
      M = 0 Normal carriage control
        = 1 (reserved)
 CCCCCC = Carriage control information:
        = 1000nn Space immediately nn space
        = 11nnnn Skip immediately to channel
        = 0000nn Space nn lines after printing
        = 01nnnn Skip to channel nnnn after
          printing
        = 000000 Suppress spacing
```

Printer data can also be directed to a standard forms file named TDISKPR1 instead of the printer. Records written in the TDISKPR1 file are in a compressed format. The TDISKPR1 file can be printed by the print utility, $DCSUP, which expands the compressed duplicate characters before printing.

To direct printer data to the disk under the TDISKPR1 file name:

1. Specify a STDPR1 parameter during initialization. The STDPR1 parameter must specify the maximum number of blocks needed to contain all the printer output directed to the TDISKPR1 file for the entire session.
2. Assign the print task to the disk during initialization or with a MODIFY utility control statement after initialization is complete.

If end of extent is reached while printer data is being written in the TDISKPR1 file, a message is issued allowing selection of one of the following actions:

- Save that portion of the printer output already written in the TDISKPR1 file and deactivate the print task. The host system saves the interrupted output for later transmission.

- Delete that portion of the current job printer output already written in the TDISKPR1 file and deactivate the print task. (Deleting that portion of the output already written makes it possible to write additional output streams into the TDISKPR1 file, providing the additional output does not reach the end of extent.) The host system saves the interrupted output for later transmission.

  Note: If the print task is deactivated, the MRJE utility cannot be terminated normally until the printer output is reallocated at the host system. (See the host system operator's guide for further information.) The task can be reactivated by a MODIFY statement that directs printer output to the printer or disk.

For each printer stream written in the TDISKPR1 file, 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.

The TDISKPR1 file can be processed, after the MRJE utility session, by the $DCSUP utility using the file name TDISKPR1.

Printer data to be saved as an individual file or to be printed later on special forms can be directed to an individual special forms file on disk. Special forms files can be processed by RPG II and basic assembler programs or by the $DCSUP print utility. These files always contain 132-character records preceded by a channel control character.

Printer files are directed to individual special forms files as the result of a forms mount message transmitted to the MRJE utility by the host system. The first character of the forms number or name in the forms mount message is compared to the character in the DID parameter specified by initialization or the MODIFY utility control statement. If the characters are the same, an individual disk file is created. If the characters are not the same, the MRJE utility issues a message. In automatic operation, the file can be written on the device assigned to the print task. Under attended operation, the system operator can enter the appropriate host system command(s) to hold the printer output at the host system for later transmission.

Note: The host system transmits a forms mount message to the MRJE utility each time there is a forms change required on the System/34. If the forms do not change from one file to the next, or from one session to the next session, the host system does not normally transmit a forms mount message. However, if the host system is restarted, the host system resets to standard forms.

For processing special forms files later, the files are labeled in the VTOC by the MRJE utility as AXXXFFFF, where:

A = Printer file

XXX = File sequence number (FSN)

FFFF = Character 1 through 4 of the forms name or number

The file sequence number is increased by one for each individual printer file or punch file created. When a print file is created, messages display the file label.

To create individual files:

1. Specify an SPCPR1 parameter during initialization. The SPCPR1 parameter must specify the blocks needed for the largest printer file to be created during the MRJE session. For each file created, excess space is returned to the system.

2. Assign the print task to either the printer or the disk during initialization or with the MODIFY statement after initialization is complete.

3. Specify the disk identifier (DID) parameter either during initialization or in a MODIFY utility control statement after initialization.

4. Specify the FSN parameter during initialization.

   Note: If the FSN value is already used in a file label, the next sequential number will be used.

If end of extent is reached while a file is being written, a message is issued that permits you to select one of the following actions:

- Save that portion of the printer output already written in the special forms file and deactivate the print task. The host system saves the interrupted output for later transmission.

- Delete that portion of the printer output already written in the special forms file area and deactivate the print task. The host system saves the interrupted output stream for later transmission.

Note: If the print task is deactivated, the MRJE utility cannot be terminated normally until the printer output is reallocated at the host system. (See the appropriate host system operator's guide for further information.) The task can be reactivated by a MODIFY statement that directs printer output to the printer or disk.

Punch Output

The punch task processes all punch data from the host system and writes 80-character records in an individual special forms disk file for subsequent processing.

These files are labeled in the VTOC by the MRJE utility as BXXXFFFF, where:

B = Punch file

XXX = File sequence number

FFFF = Characters 1 through 4 of the forms name or number

The MRJE utility increases the file sequence number by one for each punch file created. When a punch file is created, messages display the file label.

Special forms files can be processed by RPG II and basic assembler programs or by the $DCSUP utility.

To create punch special forms files:

1. Specify an SPCPU1 parameter during initialization. The SPCPU1 parameter must specify the number of blocks needed for the largest punch data file. For each file created, MRJE returns excess space to the system when the file is completed.

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

3. Specify the FSN parameter during initialization.

   Note: If the FSN value is already used in a file label, the next sequential number will be used.

If end of extent is reached while a file is being written, a message is issued that permits you to select one of the following actions:

- Save that portion of the punch file already created and terminate the punch task. The host system saves the interrupted output for later transmission.

- Delete that portion of the punch file already created and deactivate the punch task. The host system saves the interrupted output for later transmission.

Note: The MRJE utility cannot be terminated normally until the punch output is reallocated at the host system. See the appropriate host system operator's guide for further information. The punch task can be reactivated by a MODIFY statement that directs punch output to the disk.

Console Output

The console output task displays host system messages on the system console display screen. Console output also displays MRJE utility messages on either the system console or another display station acting as the reader. In addition, all messages are written into the system history file. Host system messages can be up to 120 characters. The first 79 bytes of each message are displayed, and the entire message is written in the system history file and can be printed.

Termination

To terminate a reader (display station) normally, the operator enters an EOF utility control statement. A reader can also terminate: if a physical end of file or an EOF utility control statement is encountered while reading a command file, if the system operator cancels the reader from the console input task, or if a 2 or 3 option is taken to the inquiry display from a display station. If a reader task is canceled, the data already entered is sent to the host system.

The System/34 operator normally terminates an MRJE utility session by entering an END utility control statement from the system console. (In automatic mode, the END statement can be entered from a disk command file.) Under abnormal conditions, the operator can use either the SSP CANCEL statement or take a 2 or 3 option to the inquiry display any time after initialization to terminate the session. (The CANCEL statement is a function of the SSP, not the MRJE utility.) If MRJE is abnormally terminated, a sign-off command is not sent to the host system.

The MRJE utility continues processing until an END utility control statement is read, the time delay specified on the END statement has expired, and all display stations are released. Then the MRJE utility sends the appropriate sign-off command to the host system. When all current data streams are finished, and the time delay has expired, the MRJE utility begins the termination process. The time delay specified on the END statement begins when all data streams (reader, printer, and punch) become inactive. If any data stream becomes active again, the time delay is reset and timing begins again when all data streams are inactive. Entering another END statement resets the time delay or changes the value. Note that once the END statement is entered, no new reader tasks can be started. The END utility control statement might cause some job output not to be received. This occurs when either of the following conditions exist:

- Output is queued, an END statement is read, and the DELAY-0 parameter (of the END statement) is in effect. The sign-off command is sent when transmission of the current output is complete and all readers are terminated.

- Output is inactive for the specified time delay and all readers are terminated.

If in automatic mode the END statement is not entered, the host system operator must disconnect the System/34 when the session is complete.

During termination, the MRJE utility disables the communications adapter, deallocates TDISKPR1 if necessary, and then logs the following messages:

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

The MRJE BSC I/O counters are written to disk, and the MRJE utility then goes to end of job.

MRJE UTILITY CONTROL STATEMENTS

This section contains:

- A description of writing MRJE utility control statements
- The rules for coding MRJE utility control statements
- The format of each utility control statement
- A description of the parameters in the utility control statements

The following summarizes under which task the MRJE utility control statements can be entered, and names the devices from which they can be entered.

Task	Entered From	Utility Control Statements
Reader input	Disk	READFILE EOF END (automatic mode only)
	Display station	EOF READFILE
Console input	System console	CANCEL MODIFY READFILE END CARRIAGE

Writing MRJE Utility Control Statements

The MRJE utility requires utility control statements that control the MRJE 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 are the information supplied to the utility. Each parameter consists of a keyword, which identifies the parameter, followed by the specific information supplied to the utility.

The format of an MRJE utility control statement is:

```
.. statement-identifier parameter, ..., parameter comments
(.. S)
```

When a statement abbreviation is available, it can be used instead of the complete statement identifier. Statement abbreviations are shown in parentheses.

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.

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.

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 system console only under the following conditions:

- The MODIFY statement can be entered only after initialization is complete and before an END statement is entered.
- A MODIFY statement that changes the assignment of the system console reader can be entered only when the system 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 system operator should be aware of any display station using the MRJE utility when the MODIFY statement is entered. A message can be sent from the system operator to display station operators informing them of changes.

MODIFY utility control statement format:

```
.. MODIFY [PR1-{D}] [,RD1-K] [,PU1-{D}]
          {H}               {N}
          {P}
          {R}
          [,DID-x]
(.. 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 system 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-N deactivates the punch task. PU1-D is valid only if PU1-D was also specified on the initialization screen.

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

This statement describes the disk file from which reader input data is retrieved.

READFILE utility control statement format:

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

NAME: Specifies the name of the disk file, or the name of the source or procedure member from the system library 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 indicates the file is a command file.

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.

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.

EOF

The EOF (end-of-file) statement signals the end of the current reader input to the host system. EOF deactivates the reader task at a display station. At the system console, EOF deactivates the reader task and returns the system 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 automatic mode. When entered from a reader not in automatic 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 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 system 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 equates are specified, all channel control character equates remain unchanged.

If one or more channel control character equates is specified, all unspecified channel control characters are equated 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 any time after initialization and only from the system console.

CANCEL utility control statement format:

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

Parameters:

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 deactivated. 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.

Note: The reader task can be reinitialized with the MODIFY utility control statement at the system console, or by entering the MRJE procedure command from a display station. When a reader task is deactivated, an EOF is sent to the host system, which processes any data entered before the EOF.

Notes:
1. The system operator should be aware of any display station using the MRJE utility when the CANCEL statement is entered. A message can be sent from the system operator to the display station operators informing them of changes.
2. Although each parameter is optional, at least one parameter must be specified.

MRJE SESSION EXAMPLE

This example of an MRJE session requires that MRJE utility control statements be entered from the system console. 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 similar 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:

```
HOST-R

LEN-344

DID-M

LOGON USER1/PASS1 TERM(1) PROC(MRJEPROC)
```

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 system 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 second 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 output 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 [filename]
```

The DCPRINT procedure command generates the following OCL statements:

```
// LOAD $DCSUP
// LIBRARY NAME-0
// MEMBER USER-##MSG2
// RUN
// COPYFILE NAME-filename
// GO
// END
```

If the file name is not specified, $DCSUP prompts the operator for the filename. 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.

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–filename [,DATE–date]

[,OUTPUT–{PRINT|PRINTX|NOPRINT}] [,COPYOUT–name]

[,RECL–{80|136}]  [,RECORDS–number]
```

COPYFILE parameters:

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.)

Note: 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.

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]
```

SELECT parameters:

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]
```

CARRIAGE parameters:

L: Specifies the number of print lines per form (1 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.

Default: If no channel control character equates are specified, all channel control character equates remain unchanged.

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, channel control character 1 is 1 and all other channel control characters are 0.
2. Leading zeros are not required on any CARRIAGE parameter.

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-AØØ1PAY,OUTPUT-PRINTX
// SELECT FROM-1ØØ,TO-15Ø
// 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-BØØ112ØØ
// CARRIAGE L-66,2-1Ø,12-6Ø
// 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-AØØ1PAY,OUTPUT-PRINTX
// SELECT FROM-1Ø,TO-5Ø
// GO
// SELECT FROM-75,TO-12Ø
// GO
// COPYFILE NAME-AØØ2SAVE,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-AØØØ1,COPYOUT-FILE136,
// RECL-136,RECORDS-1ØØØ,OUTPUT-NOPRINT
// 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]

[,cc-mmm] ... [,cc–mmm]

HOST: Specifies the host forms number of this entry in the table.

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 forms mount message when this forms number is encountered. The default is NO.

L: Specifies the number of lines per page. The default is the current system lines per page setting.

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.

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.

Bytes 14-27 are the channel control information, beginning with lines per page, bottom margin, and then channel control characters 1 through 12.
Bytes 28-32 are reserved.

END

The END utility control statement indicates the end of the utility control statements. The format of the END statement is:

```
// END
```

PRINTED MESSAGES

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, or a missing or extra comma.

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

The System/34 system support program product includes the SRJE (SNA Remote Job Entry) utility. The SRJE utility on the System/34 communicates in an SNA environment using SDLC line discipline. You use the SRJE utility to submit, execute, and obtain results of jobs from a host System/370. Output from jobs can be returned to the submitting System/34, directed to another RJE work station, or to the host system output devices. The SRJE utility establishes the line connection, sends and receives data, and executes the termination procedures.

The SRJE utility supports communications with the following host systems using VTAM (virtual telecommunications access method) and NCP/VS (network control program/virtual storage):

- OS/VS1 – Remote Job Entry Service (RES)
- OS/VS2 – Job Entry Subsystem 2 (JES2)
- DOS/VS – POWER/VS

SRJE ORGANIZATION

The SRJE utility consists of the following System/34 tasks:

- Console input
- Reader input
- Console output
- Printer output
- Punch output

Each of these tasks is described in greater detail under *SRJE Utility Session.* In addition, tasks exist for the SRJE supervisor, SNA, and SDLC.

STORAGE REQUIREMENTS

The minimum system for the SRJE utility is a main storage capacity of 48K bytes with a nucleus size of 18K bytes. Larger main storage capacities can have larger nucleus sizes. (See the *Planning Guide* for further information on nucleus sizes.) SRJE requires 16K bytes of nonswappable main storage. Of this 16K, 4K is for SDLC buffers (recommended minimum). You specify the number of SDLC buffers during system configuration. See the *Program Product Installation and Modification Reference Manual* for more information on configuration. SRJE also requires 14K bytes of swappable main storage (8K bytes for SNA, and 6K bytes for SRJE tasks).

SYSTEM/34 SNA PROFILE

The SNA definitions for a System/34 using SRJE are:

- System/34 is the same as the 3770 Data Communications System in batch mode. Use the 3770 default values (except the XID value) described in the appropriate host system manuals for generation. The System/34 device identifier for the XID is 00E. System/34 SNA conforms to the following SNA definitions:
  - PU Type 2
  - FM Profile 3
  - TS Profile 3
- System/34 SRJE buffer size is 256 bytes for a request/response unit (RU).
- System/34 SRJE has the following functions available to the host system:
  - One reader
  - One printer
  - One punch
  - One console

- System/34 SRJE requires that the host system be generated with PDIRs (peripheral data set information records) supported.

- The maximum pacing count used when generating NCP (network control program) determines the minimum number of System/34 receive buffers. For example:

Pacing Count	Receive Buffers
3,1	8
7,1	15

Note: Pacing count 3 or 7 is the number of data buffers System/34 can receive before it sends a pacing response. The value 1 indicates that the first buffer (of the 3 or 7) of data from the host system should request the pacing response from System/34.

SRJE SESSION

The parts of an SRJE session are:

- *Initiation* of the SRJE utility via the SRJE procedure command.

- *Initialization* of the SRJE session to establish the link with the host system.

- *Input* to the SRJE utility and the host system.

- *Output* from the SRJE utility and the host system.

- *Termination* of the SRJE session including ending the link with the host system.

Initiation

Load the SRJE utility by entering the SRJE procedure command from any command display station. That display station becomes the SRJE console.

The format of the SRJE procedure command is:

SRJE [#SR@ID / name] [,PRIORITY]

name: Specifies the name of the initialization format member for the display screen. The default is #SR@ID (the format member provided by SRJE).

PRIORITY: Indicates that the tasks activated by SRJE should have user priority. If you omit this parameter, the SRJE tasks will not have priority.

Initialization

After you load the SRJE utility, you initialize it through the SRJE console.

SRJE displays the initialization information in either the SRJE-supplied format or a user-written format. You can modify the information, in either format, by positioning the cursor under the value to be changed and keying the change. Press the Enter/Rec Adv key when all changes have been made. To use the initialization information as is, press the Enter/Rec Adv key. SRJE initializes and assigns the functions of the utility and establishes the network link with the host system.

The following illustration shows the initialization display that is supplied as part of the SRJE utility. This display shows the options that you can use to initialize the SRJE utility.

```
SRJE OPTION MENU - PRESS ENTER TO CONTINUE

ENTER INITIAL CONFIGURATION INFORMATION BELOW:

PLUNAME..            LOGON.... N          FSN..... 001      DEVID.... SYSTEM
FID..... PR1         OUTPUT... P1         SPACE.... 0060
FID..... PU1         OUTPUT... N          SPACE.... 0060
RD1NAME..            RD1TYPE.. D          RD1CMD... Y       RD1DATE.. 000000
RD1LIBR..            RD1XPC... N

ENTER INITIAL CARRIAGE 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

ENTER LOGON INFORMATION BELOW:
LOGON

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

The initial configuration information consists of parameters. Each parameter contains a keyword and a value. In the illustration, LOGON is the keyword and N is its value, indicating that the SRJE utility should not send a logon to the host system. All parameters with values assigned are default values. Default values appear on the SRJE initialization display from either the SRJE-supplied or user-supplied initialization format. You can change the default values shown; you can also assign values where none are shown.

PLUNAME: Specifies the name (eight characters maximum) by which the host system is known in the network. PLUNAME is optional.

LOGON: Indicates whether to use a logon statement during SNA initialization. Y indicates to use the logon command specified below (on the display). N indicates to not use a logon command.

FSN: Specifies a number for identifying temporary disk files. The number is three decimal digits and the default is 001. SRJE uses the number in generating labels for temporary files (see *Print Output, Punch Output*).

RD1NAME: Specifies the name of the disk file or library member to be read by the reader task.

RD1TYPE: Specifies the type of file specified by the RD1NAME parameter. RD1TYPE-D indicates a disk file. RD1TYPE-S indicates a source member. RD1TYPE-P indicates a procedure member.

RD1CMD: Specifies whether the file named in the RD1NAME parameter is a command file or a data file. RD1CMD-Y indicates the RD1NAME file is a command file. RD1CMD-N indicates the RD1NAME file is a data file.

RD1DATE: Specifies the creation date of the disk file specified in the RD1NAME parameter. The RD1DATE parameter is required only if more than one file with the specified name exists on the disk. Enter the date in the system date format.

RD1LIBR: Specifies the name of the library to search for the member specified by the RD1NAME parameter. If you omit this parameter, SRJE searches the system library. This parameter is valid only with RD1TYPE-P or RD1TYPE-S.

RD1XPC: Specifies whether the file to be read contains transparent data (unprintable characters). Y indicates the file contains transparent data. RD1XPC-Y is not valid with RD1CMD-Y. N indicates the file does not contain transparent data; N is the default.

SRJE uses the carriage information to set the printer forms length and to correlate channel control characters with form line numbers. L represents the number of lines per page and B represents the line number of the bottom margin (last printed line). Each of the digits 1 through 12 has a value that represents the channel control character value for that channel. F specifies the initial output forms number for SRJE. The forms number can be any string of alphameric characters. Be sure to generate the host system with PDIRs required, and build a forms control table using $DCFUP, for proper forms handling. FCTNAME specifies the name of the forms control table built by the $DCFUP utility.

The sign-on command must conform to the format and requirements of the host system.

The system operator message is up to 160 characters long. The message appears on the status display of any display station attaching to the SRJE utility.

Input

The SRJE utility reads input and either processes it or transmits it to the host system. Input can be in the following forms:

Input Task	Input Device	Input
Reader task	Disk • Data file • Source member • Procedure member Keyboard at a display station	Utility control statements Jobs
Console input task	Keyboard at the SRJE console	Host system commands Utility control statements

Reader Input

The SRJE reader task begins when an operator at a command display station enters the SRJE procedure command after the SRJE utility is already active, or when the SRJE console operator presses the Attn key followed by command key 1. 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 current values instead of the x's):

```
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

ENTER INITIAL CARRIAGE INFORMATION BELOW:

L.... XXX    B.... XXX    F.... XXXX  FCTNAME..XXXXXX
1.... XXX    2.... XXX    3.... XXX   4.... XXX    5.... XXX    6.... XXX
7.... XXX    8.... XXX    9.... XXX   10... XXX    11... XXX    12... XXX

LOGON INFORMATION BELOW:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

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

The values in this display indicate the current status of SRJE configuration. If the operator chooses not to continue after seeing the status display, the reader can be terminated by pressing the Attn key. After the Enter/Rec Adv key is pressed, the reader display appears the same as the system input/output display with a reverse image RD1 on line 1. The reader display indicates that utility control statements, JCL statements, 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 and all input is treated as a command file.

The reader deactivates when an operator enters an EOF utility control statement, or when an end of file occurs in a command file and the associated display station has been released.

When the disk is used as input for the reader task, the name of the disk file or library member must be specified during initialization or on the READFILE utility control statement.

Note: Disk files used by the reader cannot be shared with other programs, while the reader is accessing them.

Data files contain records that are transmitted to the host system. These records can have any record length. A data file can contain:

- Data (SRJE treats utility control statements in a data file as data and transmits them to the host system.)
- JCL
- Host system commands

The SRJE 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. The customer application program on the host system must return the data file to its original record length.

A command file can contain:

- Data (transmitted to the host system)
- JCL (transmitted to the host system)
- Utility control statements (executed by the SRJE utility)
- Host system commands (transmitted to the host system)

Command files must have a record length of 80 bytes and must not contain transparent data.

Console Input

Input through the console input task can be entered only from the display station that first called the SRJE utility (the SRJE console). The console input display appears when the SRJE operator presses the Enter/Rec Adv key from the initialization display, or the Attn key and the Enter/Rec Adv key any other time.

The console input display is the same as the system input/output display with a reverse image CNI on line 1.

Input can be host system commands or MODIFY, CANCEL, FORMS, or END utility control statements. Up to 120 characters of input can be entered at one time and all console input is logged to the history file. Records consisting of all blanks are ignored.

While the reader task is active on the SRJE console, console input commands can be entered after the operator presses the Attn key and then the Enter/Rec Adv key without entering any data. The console input field replaces the reader input field and the operator enters the console input.

Output

The SRJE utility output can consist of one, all, or a combination of:

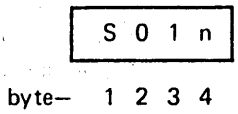
- Print data from the host system (print output)
- Punch data from the host system (punch output)
- Operator messages from either the host system or the SRJE utility (console output)

Print Output

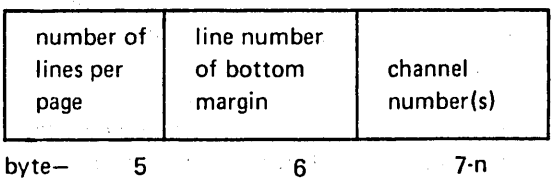
The SRJE operator can specify, via the initialization display and the MODIFY statement, whether print output is written to a printer or to disk. Print output on the disk can be printed by $DCSUP or processed by a customer-written program.

Printer forms operations are controlled by channel control characters and the forms control table. Print output written to disk is a file with either 136-character records or 256-character records. $DCSUP can print the output and can create a file of 132-character records from the 256-character records. The first four bytes of the 136-character record have one of the following formats to control the forms operations for a print file created by the SRJE utility:

1. Vertical tab table identifier



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) is the length of the vertical tab table including the length byte. The vertical tab table format is:



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* channel 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 value 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, then the record following the PDIR in the file has the following format (this record is used by $DCSUP for processing the file):

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 three characters S02 in the first three bytes indicate that this is a 256-character record. The fourth byte (n) is the length of the special record information (253 bytes maximum, including the length byte). Byte 5 (control byte) indicates:

- X'80' = The record is in 256-character record (compressed) format.
- X'20' = A vertical tab table is present in this record.
- X'10' = A horizontal tab table is present in this record.
- X'02' = This is a print file. (Logical records are 132 bytes.)
- X'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 into the new file. The PDIR is the first record in a print file.

Note: You must request PDIRs from the host system during host system generation.

The format of the PDIR is:

Byte	Field Meaning	Value
0-2	Record identifier	C'S03'
3	Length	X'42'
4	Identifier	X'00' = Ordinary data set X'01' = Job separators X'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	FCB 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 number

Punch Output

The SRJE utility writes all punch output from the host system to disk files. The SRJE utility creates punch output on disk as 80-byte records or 256-byte records. Punch output on disk can be printed by $DCSUP or processed by a customer-written program.

Punch output that contains 256-byte records resides on disk in the form that the SRJE utility receives the data. The SRJE utility requires less processing time and disk space, because SNA does not process the data before SRJE writes the data to disk. $DCSUP can print the punch output or can create a disk file of 80-byte records from the 256-byte records. The file created by $DCSUP is the same as the one that the SRJE utility would have created as 80-byte records.

Punch output directed to a disk file is labeled by the SRJE utilitiy, using the FSN parameter on the initialization display, as BXXXFFFF, where:

B = Punch output disk file
XXX = File sequence number (FSN)
FFFF = Forms number (if received via PDIR)

Console Output

The console output task displays host system messages on the SRJE console display screen. Each task displays its own SRJE utility messages. In addition, all messages are written into the system history file. Host system messages can be up to 120 characters. The first 75 bytes of each message are displayed and the entire message is written in 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 message to the SRJE console.

Termination

Either the SRJE operator or the host system operator can terminate a session. The SRJE 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 - nnnnn
OUTPUT RECORDS PRINTED - nnnnn
OUTPUT RECORDS PUNCHED - nnnnn
SNA FORMAT RECORDS WRITTEN - nnnnn

The SRJE operator can terminate the session abnormally by entering the SRJE CANCEL utility control statement. If SRJE terminates abnormally, a sign-off command is not sent to the host system. The System/34 operator can also terminate the session by canceling the SRJE supervisor task or by issuing the System/34 STOP command.

SRJE UTILITY CONTROL STATEMENTS

This section contains:

- An introduction to the writing of SRJE utility control statements.
- The rules for coding SRJE utility control statements.
- The format of each utility control statement.
- A description of the parameters in the utility control statements.

The following summarizes under which task (and from which device) you can enter the SRJE utility control statements.

Task	Entered From	SRJE Utility Control Statements
Reader Input	Keyboard Disk	READFILE END EOF WTO
Console Input	Keyboard	CANCEL MODIFY END FORMS

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 blank must precede the statement identifier. Do not use blanks within the identifier.

Blanks: One or more blanks are required between the 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 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 utility control statement and the comment. Starting a statement with .. * makes the entire statement a comment.

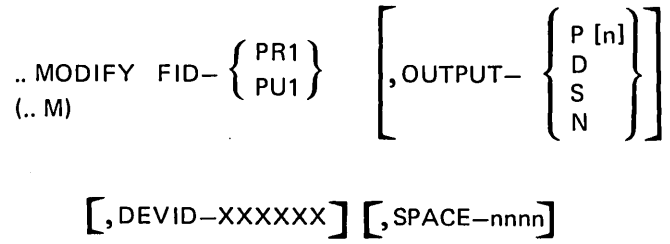
MODIFY

The MODIFY utility control statement reassigns the punch and print tasks. You can enter the MODIFY statement only from the SRJE console, after initialization is complete and before you enter an 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 operator should be aware of any display station using the SRJE utility when entering the MODIFY statement. A message can be sent from the system operator to display station operators informing them of changes.

The format of the MODIFY statement is:



FID: Specifies the task to be modified. FID-PR1 indicates the print task; FID-PU1 indicates the punch task.

OUTPUT: Specifies the device for print or punch data.

OUTPUT-P directs print data to the printer and is valid only with FID-PR1. 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 with 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.

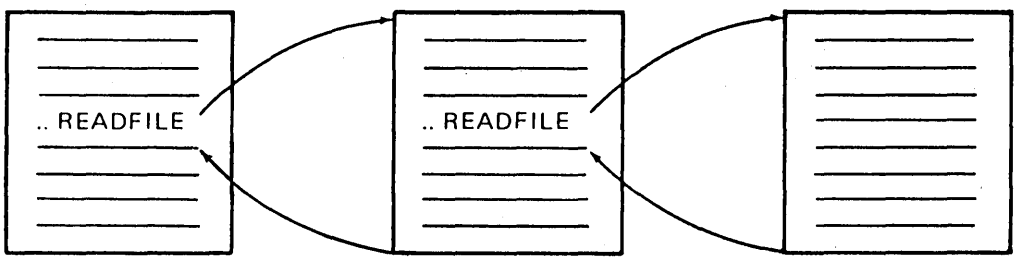
OUTPUT-N indicates no print or punch data is accepted and the task is deactivated when the current file is completed.

DEVID: Specifies the System/34 work station ID (or SYSTEM for the system printer) of the printer to use when print output goes to the printer. DEVID is valid only with FID-PR1 and either OUTPUT-P or print data 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 of D or S, or if output is already going to disk.

READFILE

The READFILE utility control statement describes the disk file, procedure member, or source member from which data is read. READFILE statements can be nested; that is, within a command file, if another READFILE is read, that file is read. After the second file is read, control returns to the first file at the next statement after the READFILE. Nested READFILEs are illustrated as follows:



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 available disk space. See the *Planning Guide* for more information.

You can enter the READFILE statement only through reader input.

The format of the READFILE statement is:

```
.. READFILE NAME–name [,LIBR–library name] [,DATE–date]
(.. R)

[,TYPE–{P}] [,CMD–{Y}] [,XPC–{Y}]
       {S}      {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 to search for the member specified by the NAME parameter. If you omit this parameter, SRJE searches the system library (#LIBRARY). LIBR is valid only with TYPE-P or TYPE-S.

DATE: Specifies the file creation date. DATE is valid only with TYPE-D.

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

CMD: Specifies whether the file to be read is a command file.

CMD-Y indicates the file is a command file. SRJE processes any SRJE utility control statements, and transmits all other records to the host system. Command files are processed consecutively and must have a record length of 80 bytes. CMD-Y is not valid with XPC-Y.

CMD-N 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 to the host system. CMD-N is the default.

XPC: Specifies whether the file to be read contains transparent data (unprintable characters). XPC-Y indicates the file contains transparent data. XPC-Y is not valid with CMD-Y. XPC-N indicates the file does not contain transparent data; N is the default.

Note: If SRJE detects any error in a READFILE statement, the statement is not processed. If the display station has not been released, the operator is given the opportunity to enter a valid READFILE statement. If the display station has been released, the SRJE reader 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 deactivates the reader task. Only the last job entered by the reader needs an EOF at the end.

When the SRJE utility encounters an EOF statement, 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 statement before the end of the procedure or file, SRJE requests input from the device from which the last READFILE was read, at the record immediately following the last READFILE. If the last READFILE was read from a released display station, then SRJE simulates an EOF statement.

You can enter the EOF statement only through reader input. The format of the EOF utility control statement is:

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

WTO

The WTO (write-to-operator) utility control statement sends a message 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 from the reader.

The format of the WTO statement is:

```
.. WTO  ['any text']
(.. W)
```

The total length of the WTO statement cannot exceed 80 characters.

END

The END utility control statement, when entered through console input, causes SRJE to request normal termination from the host system. If you enter the END statement through the reader, SRJE logs the END to the history file and treats it as a comment.

The format of the END statement is:

```
.. END
```

CANCEL

The CANCEL utility control statement immediately terminates the SRJE utility or one of its functions. You can enter CANCEL only through console input.

The format of the CANCEL statement is:

```
.. CANCEL [RD1]
(..C)     [PR1]
          [PU1]
```

RD1: Terminates the reader task.

PR1: Terminates the printer task.

PU1: Terminates the 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).

The format of the FORMS statement is:

```
.. FORMS [NAME-name]
```

NAME: Specifies the host forms name to look for in the forms control table created by $DCFUP. If the name is found and the halt parameter is not on, the SRJE carriage information and System/34 forms name is updated using the values in the forms control table. If the halt parameter is on, the display shown below appears with the values from this record substituted for the x's. If NAME is not specified, the name is not found, or no forms control table exists, the display shown below appears with the current values substituted for the x's.

```
SRJE FORMS CONTROL INFORMATION - PRESS ENTER TO CONTINUE


HOST FORMS NAME.... XXXXXXXX  LOCAL FORMS NAME.... XXXXXXXX

                CARRIAGE INFORMATION

L.... 066       B.... 066
1.... 001       2.... 000   3.... 000  4.... 000     5.... 000    6.... 000
7.... 000       8.... 000   9.... 000  10... 000     11... 000    12... 000
```

The operator can now update any of the fields, except *host forms name*. The values entered are then used for printer output.

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								

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				/		a	j	~		A	J		1
0010	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
0011	3	ETX	DC3	TM / DC3						c	l	t		C	L	T	3
0100	4	PF	RES	BYP	PN					d	m	u		D	M	U	4
0101	5	HT	NL	LF	RS					e	n	v		E	N	V	5
0110	6	LC	BS	EOB / ETB	UC					f	o	w		F	O	W	6
0111	7	DEL	IL	PRE / ESC	EOT					g	p	x		G	P	X	7
1000	8		CAN							h	q	y		H	Q	Y	8
1001	9	RLF	EM						`	i	r	z		I	R	Z	9
1010	A	SMM	CC	SM		¢	!	¦	:								LVM
1011	B	VT	CU1	CU2	CU3	.	$	,	#								
1100	C	FF	IFS		DC4	<	*	%	@					⌐		⌐	
1101	D	CR	IGS	ENQ	NAK	(	)	_	'								
1110	E	SO	IRS	ACK		+	;	>	=					⌐			
1111	F	SI	IUS	BEL	SUB	¦	¬	?	"								EO

Duplicate Assignment

Appendix B. Transmission Control Characters for BSC

The following characters and character sequences are recognized by System/34 BSC. Shown below each character is the hexadecimal configuration. For detailed information on transmission control characters (also called line control characters), see *General Information—Binary Synchronous Communications*, GA27-3004.

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 cancel	XENQ		DLE ENQ 102D
Transparent TTD	XTTD		DLE STX DLE ENQ 10 02 10 20
Data DLE in transparent mode	XDLE		DLE DLE 10 10

Appendix C. Polling and Addressing Characters for System/34 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

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:

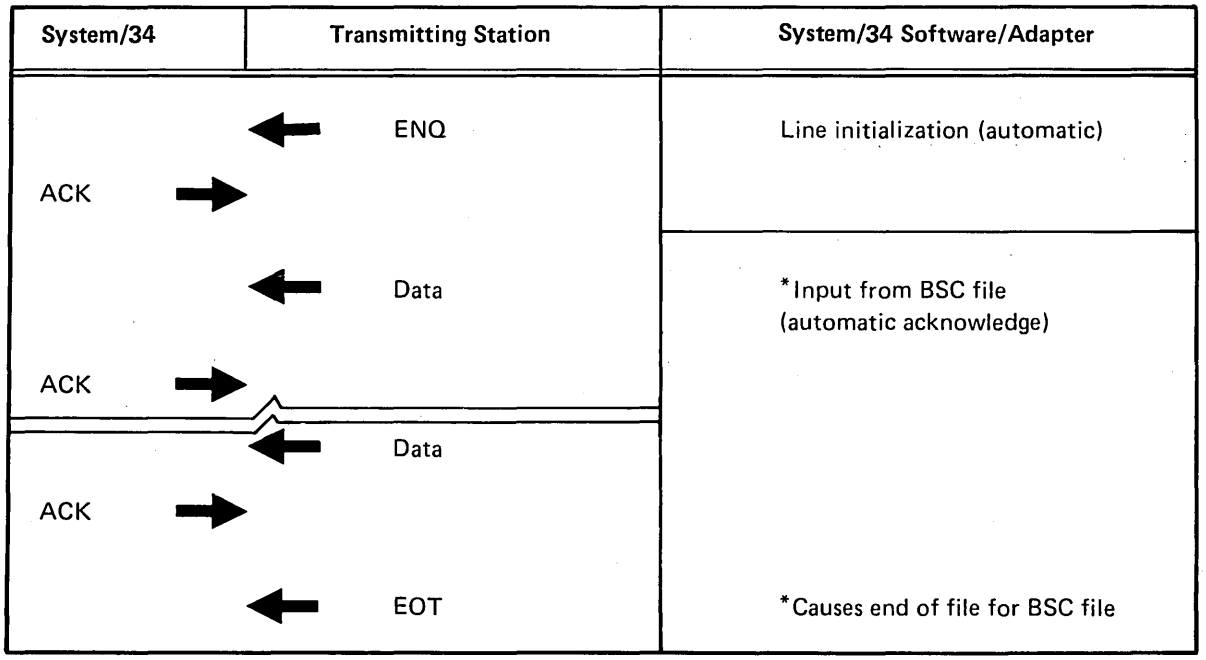
- Write a program for the remote location to interface with System/34 RPG II or assembler.
- Write a program for the System/34 to perform a specific sequence of line protocol.

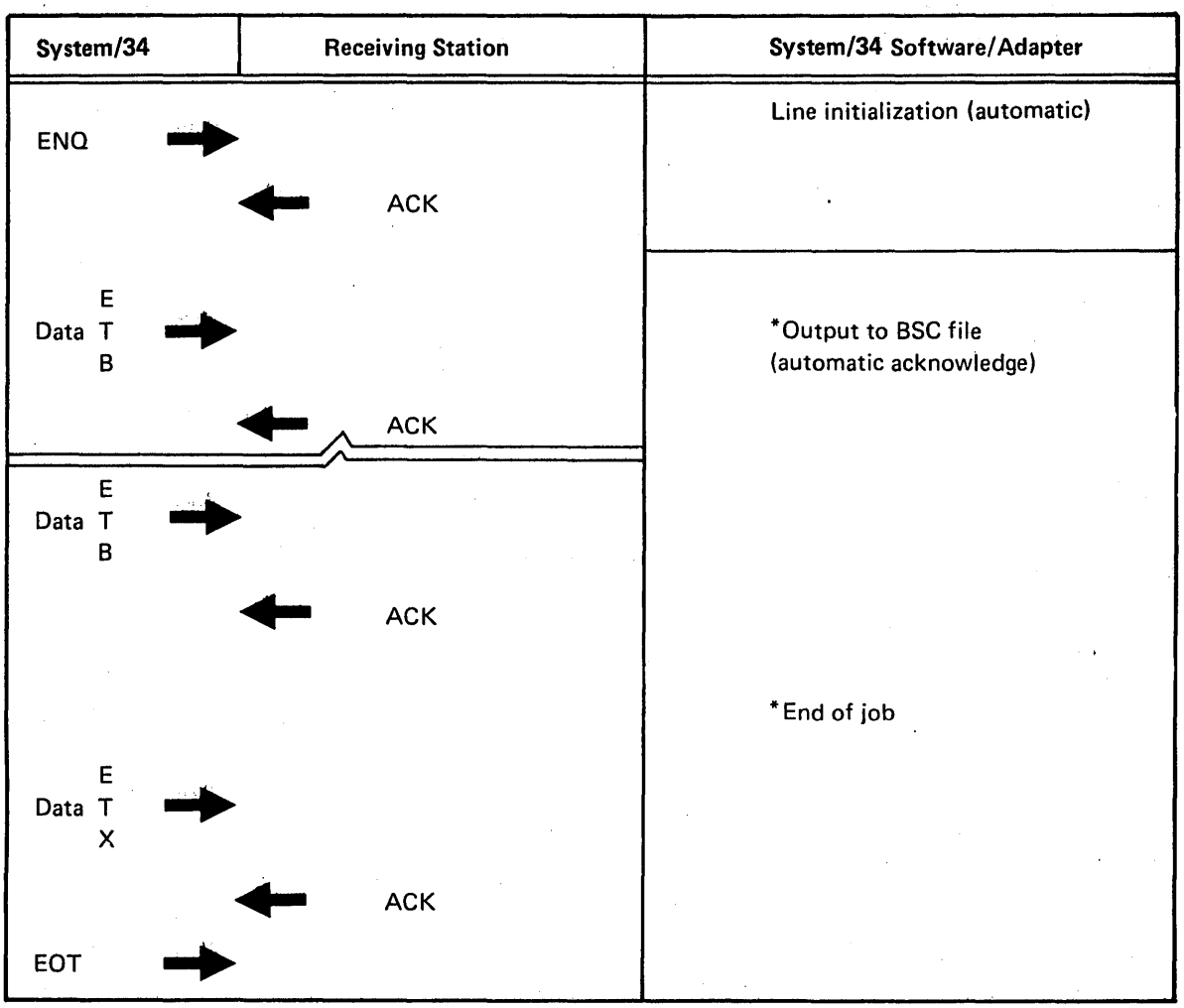
This appendix shows RPG II/assembler binary synchronous communications (BSC) line protocols and indicates which ones are performed automatically and which ones the program controls.

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.)

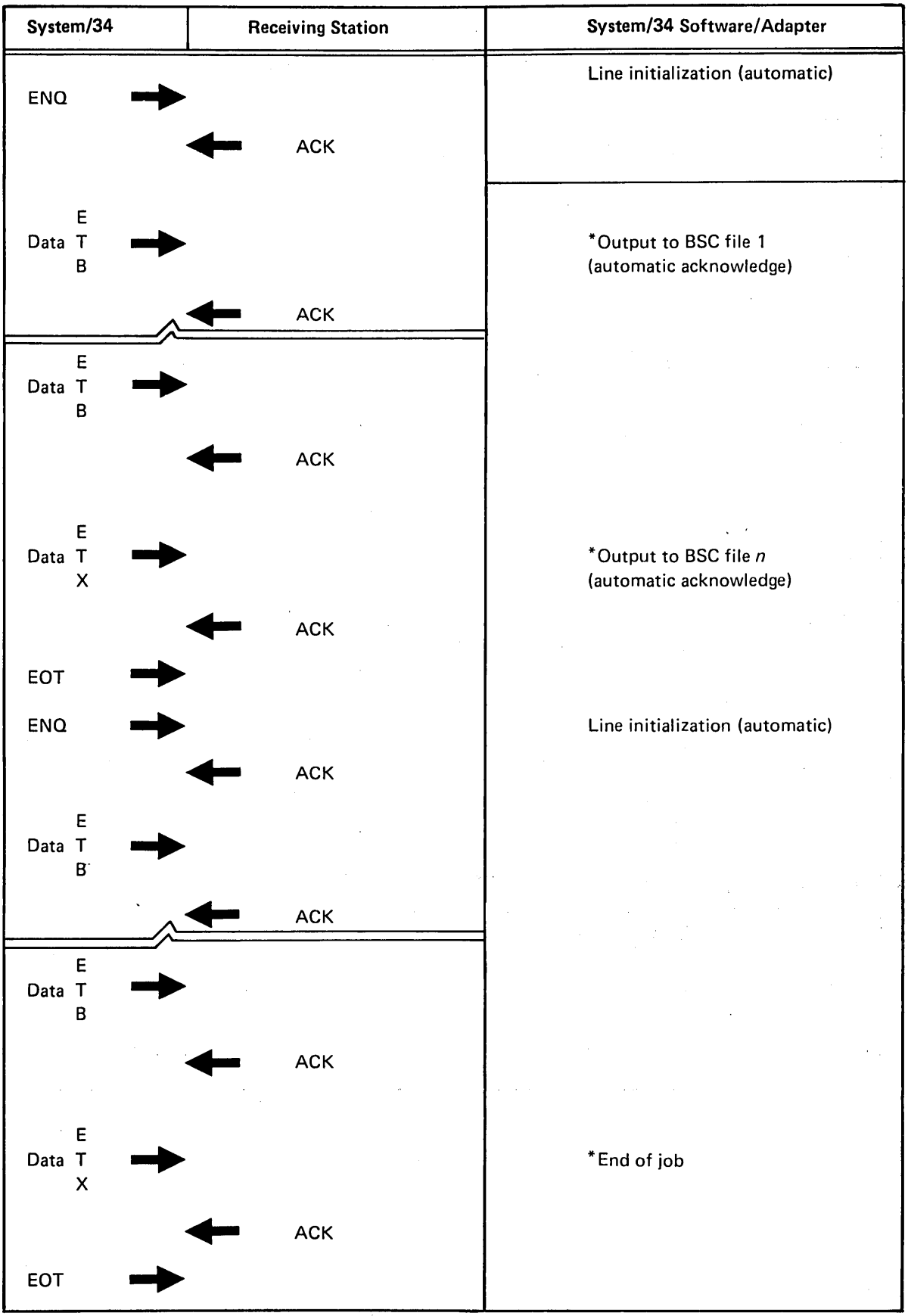
POINT-TO-POINT NONSWITCHED LINE

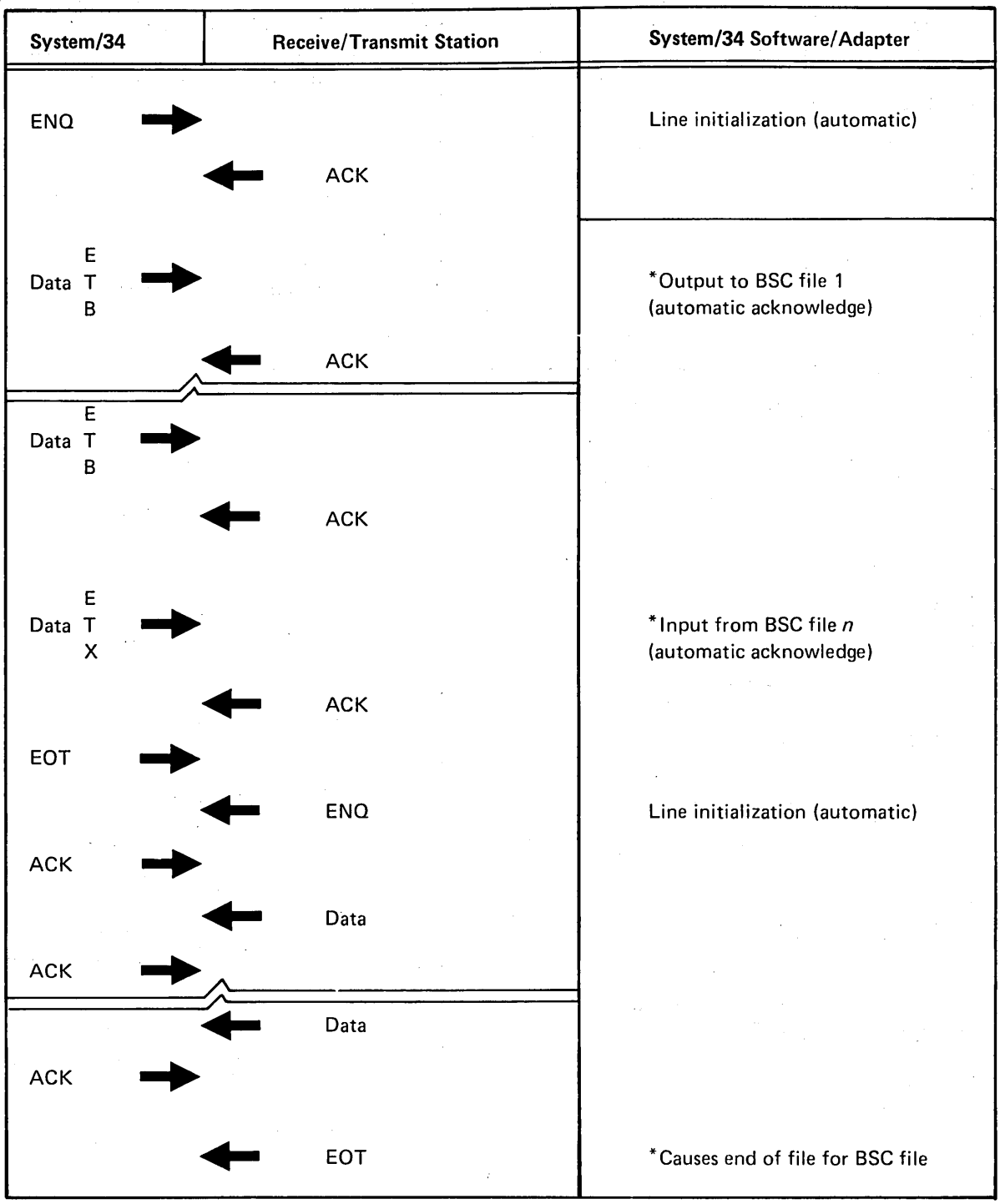
Receive



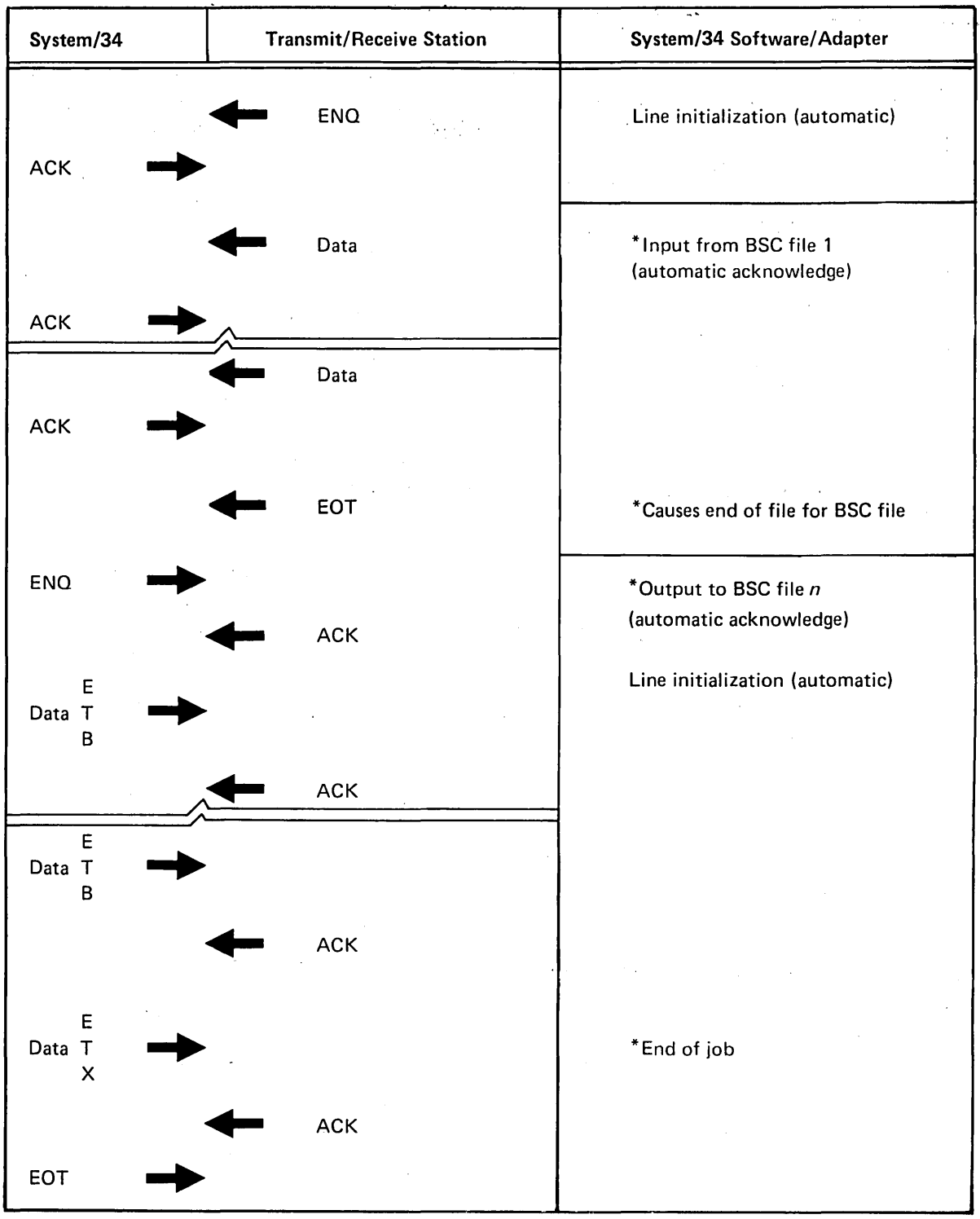

System/34
Receiving Station
System/34 Software/Adapter
Line initialization (automatic)
ENQ
ACK
Data ETB
*Output to BSC file
(automatic acknowledge)
ACK
Data ETB
ACK
*End of job
Data ETX
ACK
EOT

Transmit Multiple Files



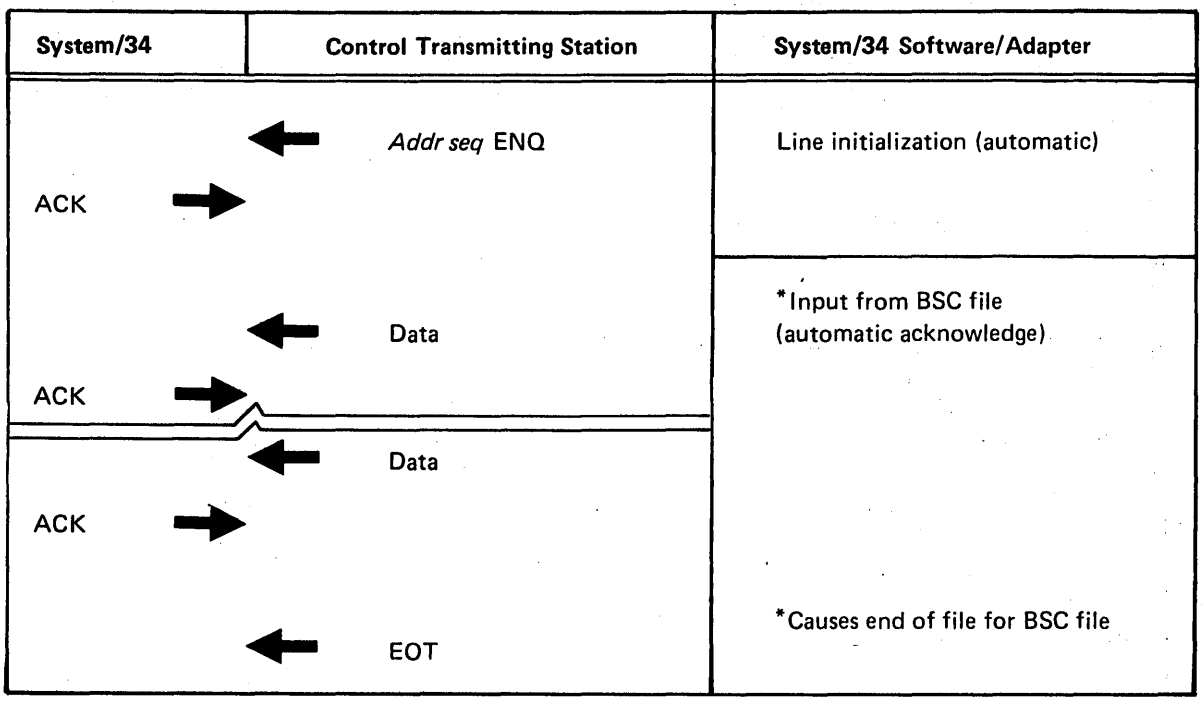

System/34
Receive/Transmit Station
System/34 Software/Adapter
ENQ
ACK
Line initialization (automatic)
Data ETB
*Output to BSC file 1
(automatic acknowledge)
ACK
Data ETB
ACK
Data ETX
*Input from BSC file n
(automatic acknowledge)
ACK
EOT
ENQ
Line initialization (automatic)
ACK
Data
ACK
Data
ACK
EOT
*Causes end of file for BSC file

Receive/Transmit

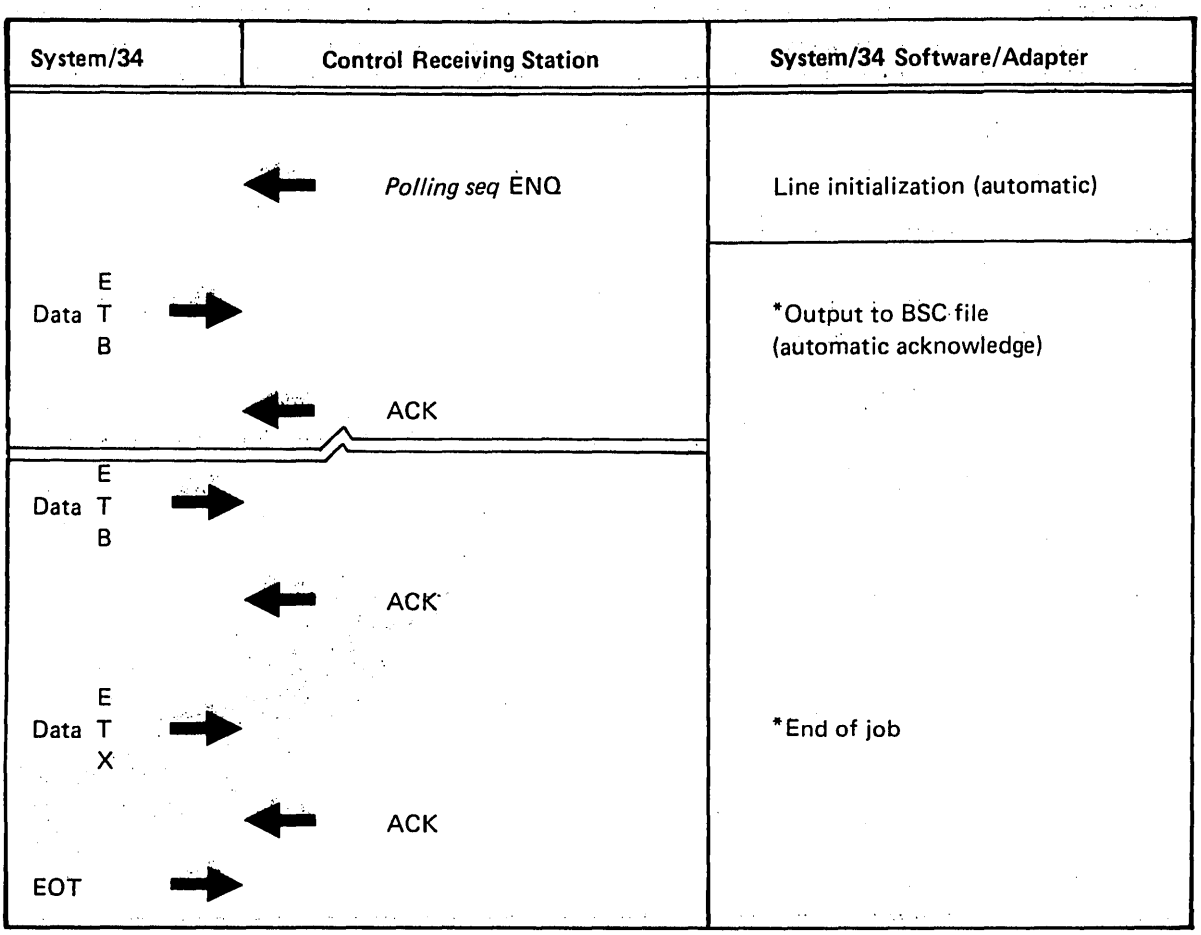


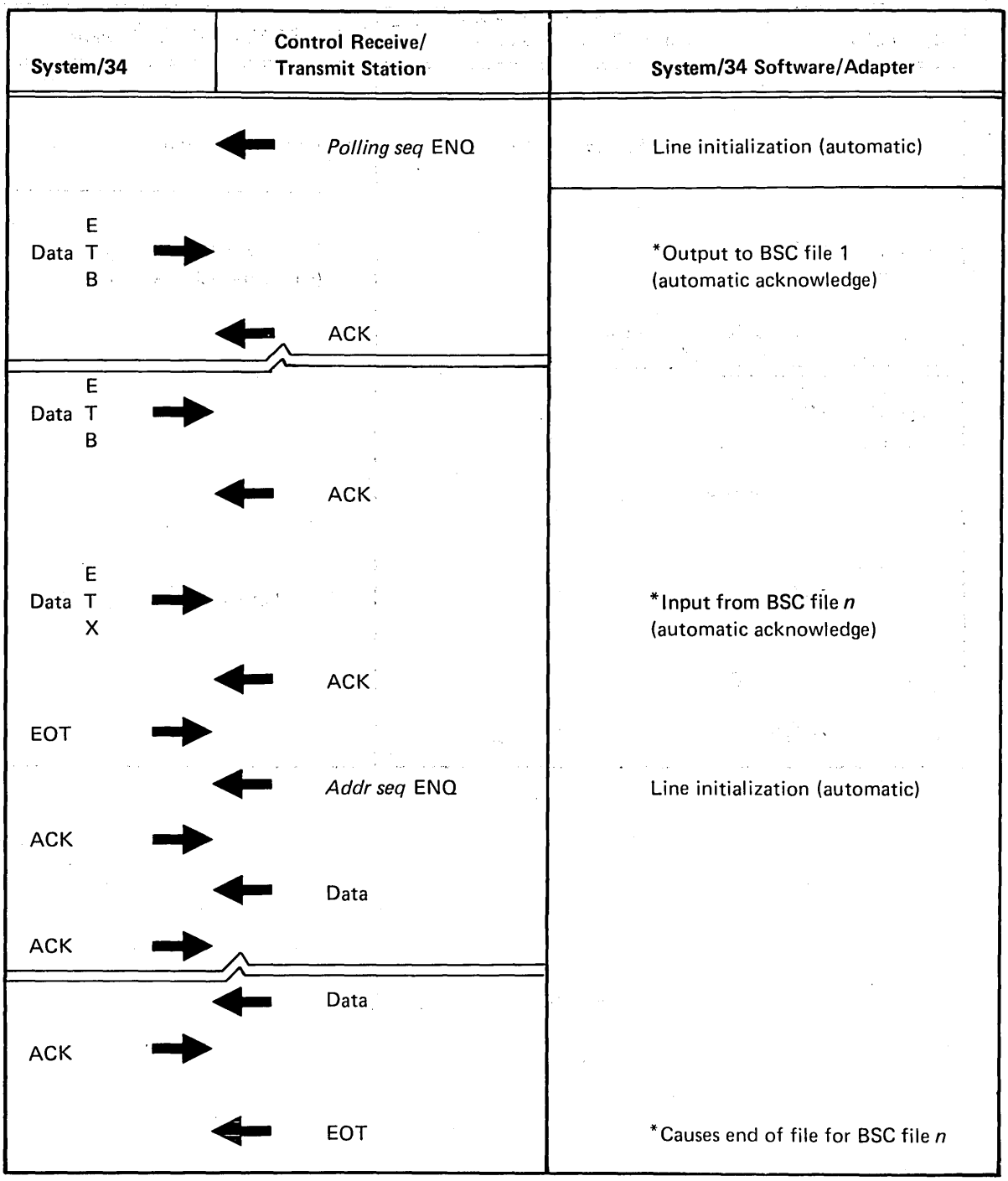
NONSWITCHED MULTIPOINT, SYSTEM/34 TRIBUTARY STATION

Receive

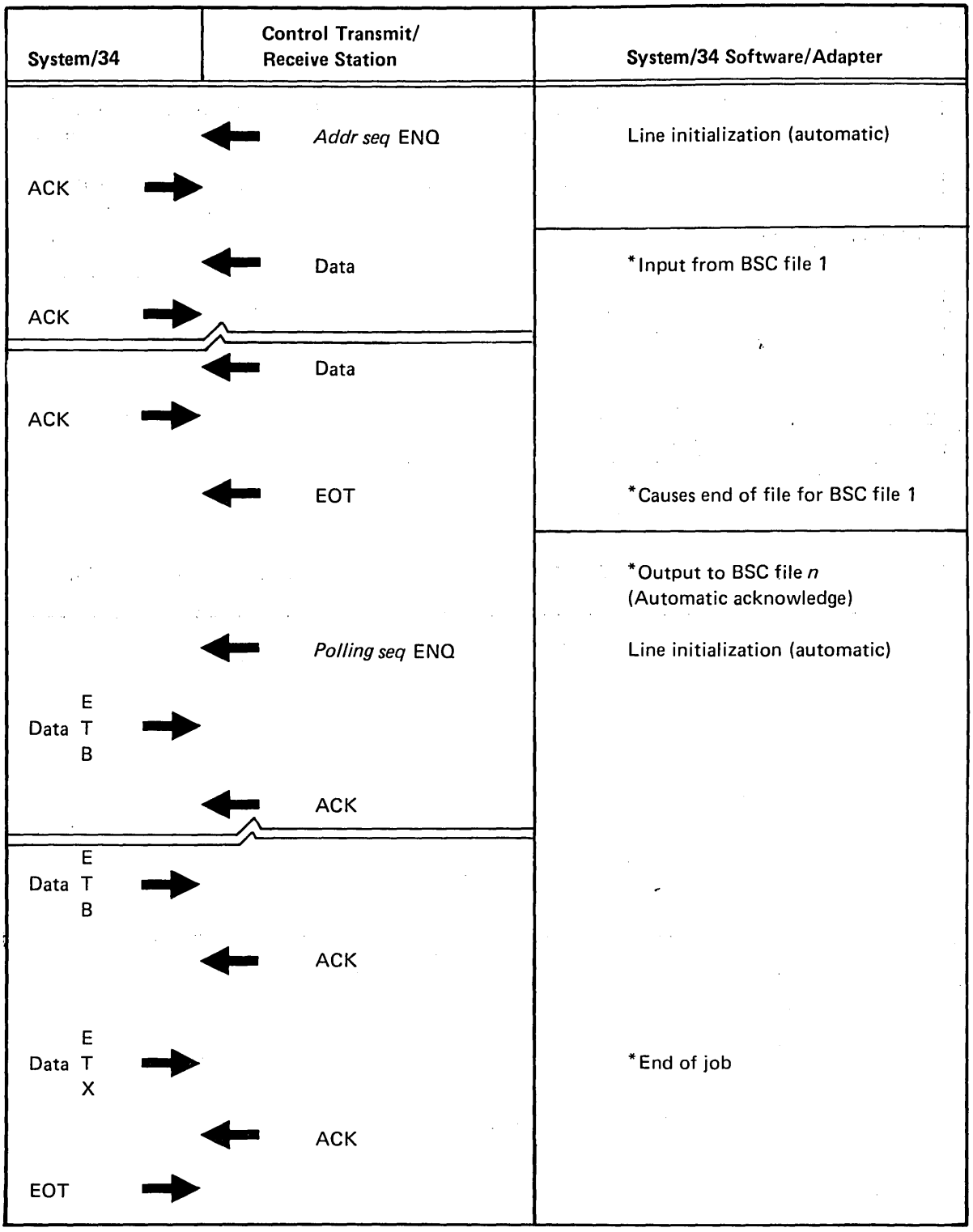


Transmit


System/34
Control Receiving Station
System/34 Software/Adapter
Polling seq ENQ
Line initialization (automatic)
Data ETB
*Output to BSC file
(automatic acknowledge)
ACK
Data ETB
ACK
Data ETX
*End of job
ACK
EOT

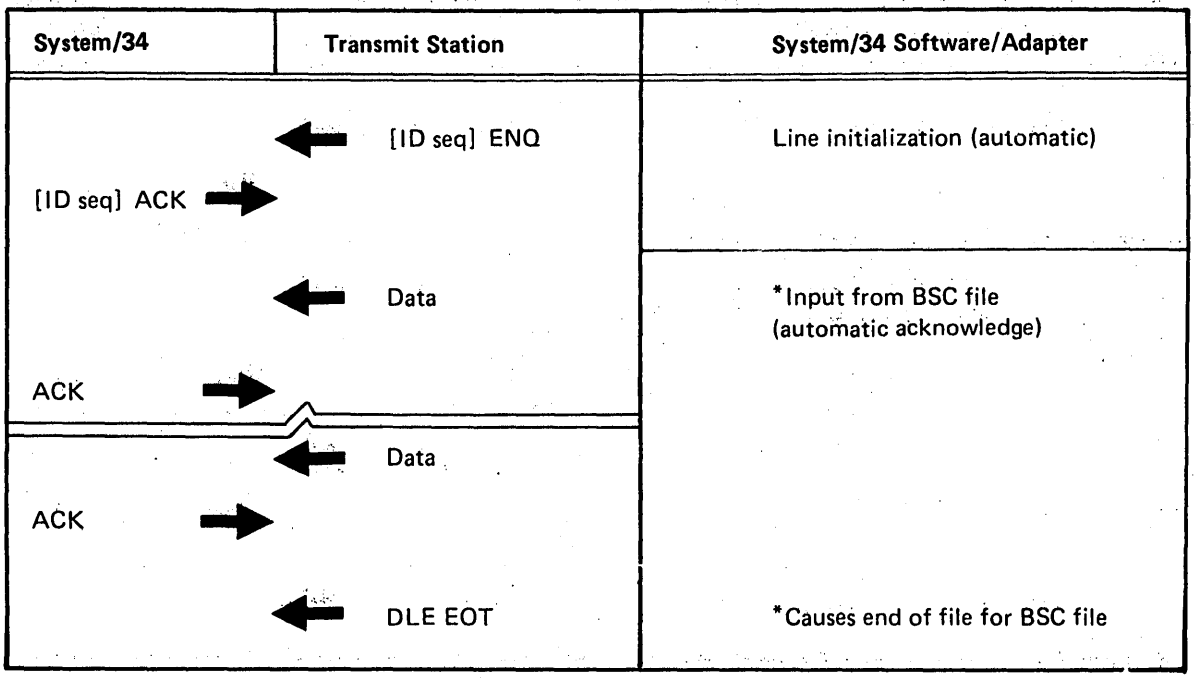

System/34
Control Receive/ Transmit Station
System/34 Software/Adapter
Polling seq ENQ
Line initialization (automatic)
Data ETB
*Output to BSC file 1 (automatic acknowledge)
ACK
Data ETB
ACK
Data ETX
*Input from BSC file n (automatic acknowledge)
ACK
EOT
Addr seq ENQ
Line initialization (automatic)
ACK
Data
ACK
Data
ACK
EOT
*Causes end of file for BSC file n

Receive, Transmit

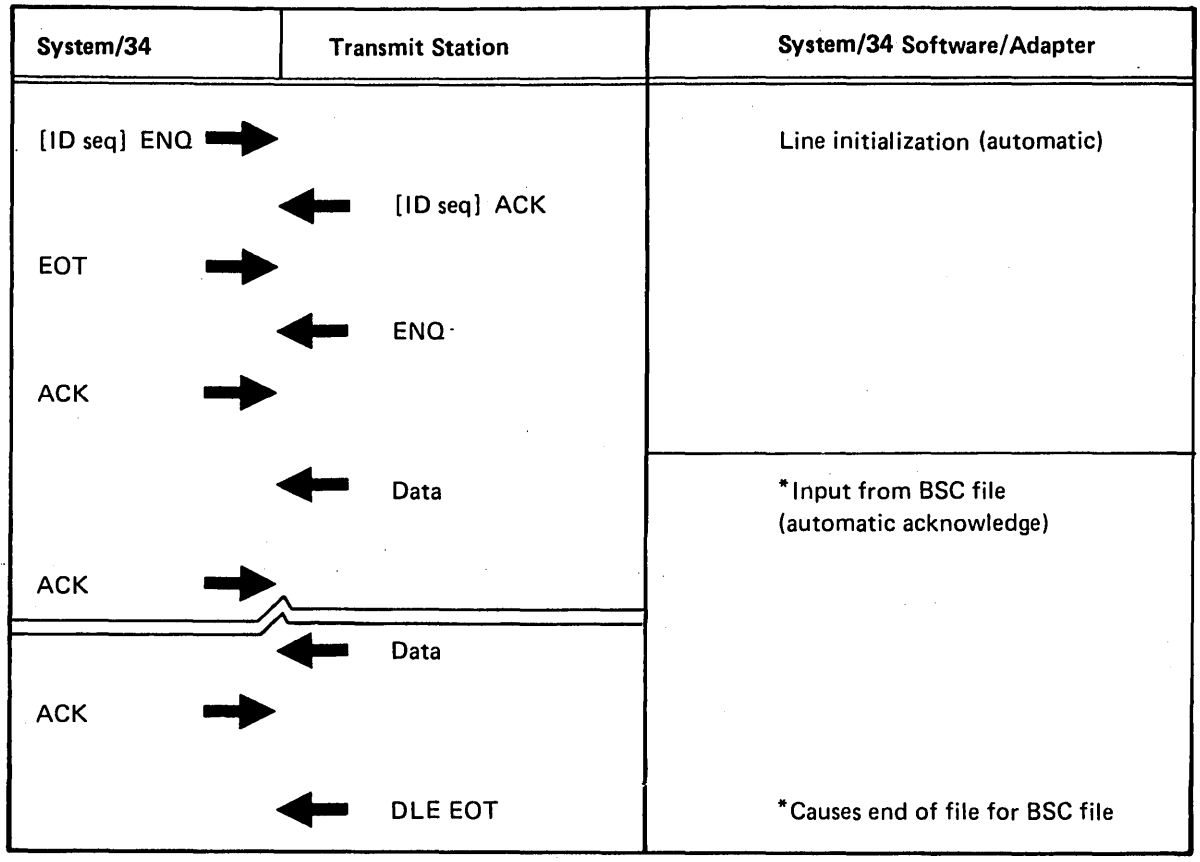


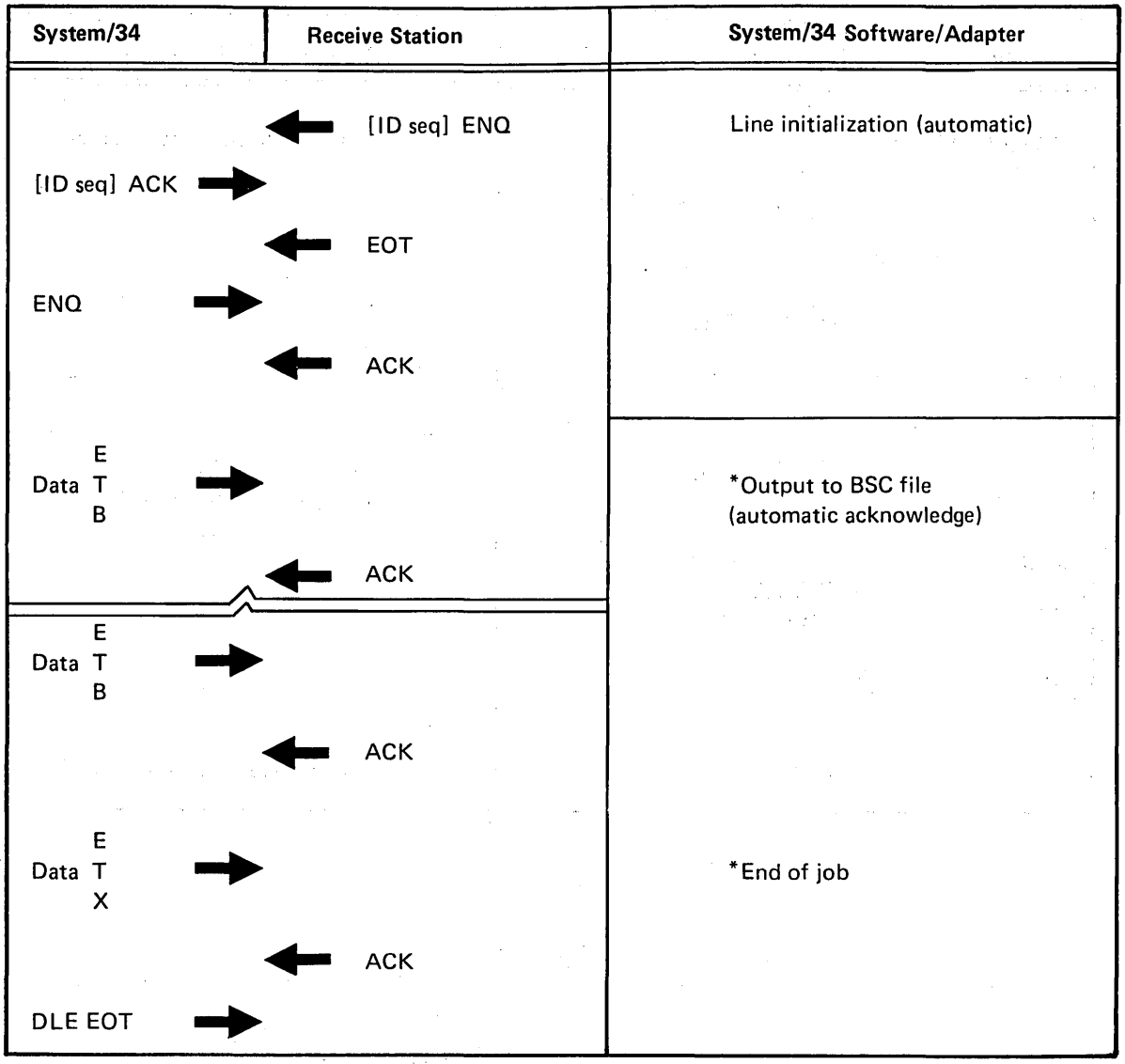
POINT-TO-POINT SWITCHED LINE
ID SEQ: SWITCHED LINE STATION IDENTIFICATION SEQUENCE OPTIONAL

Receive, System/34 Answer Station

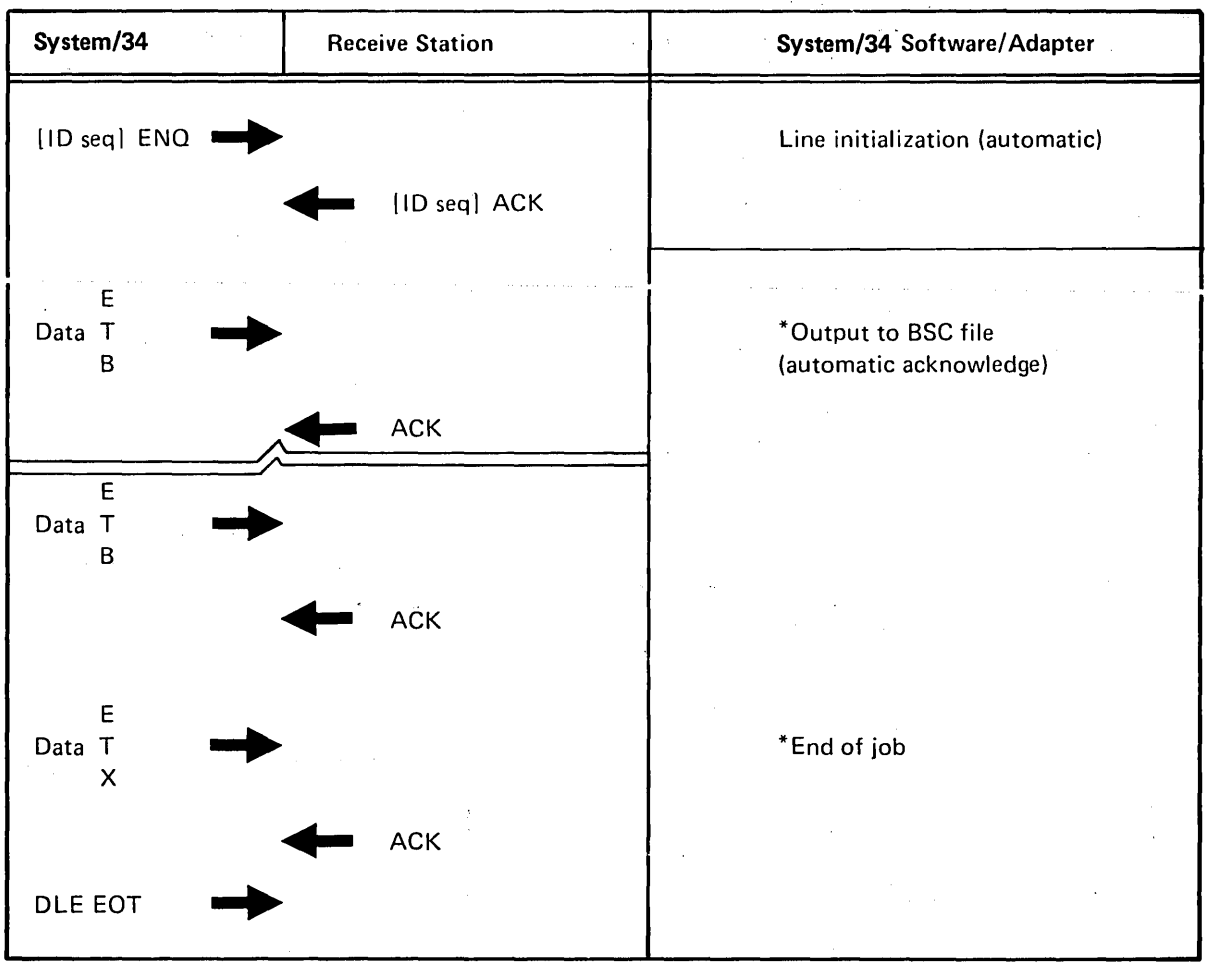


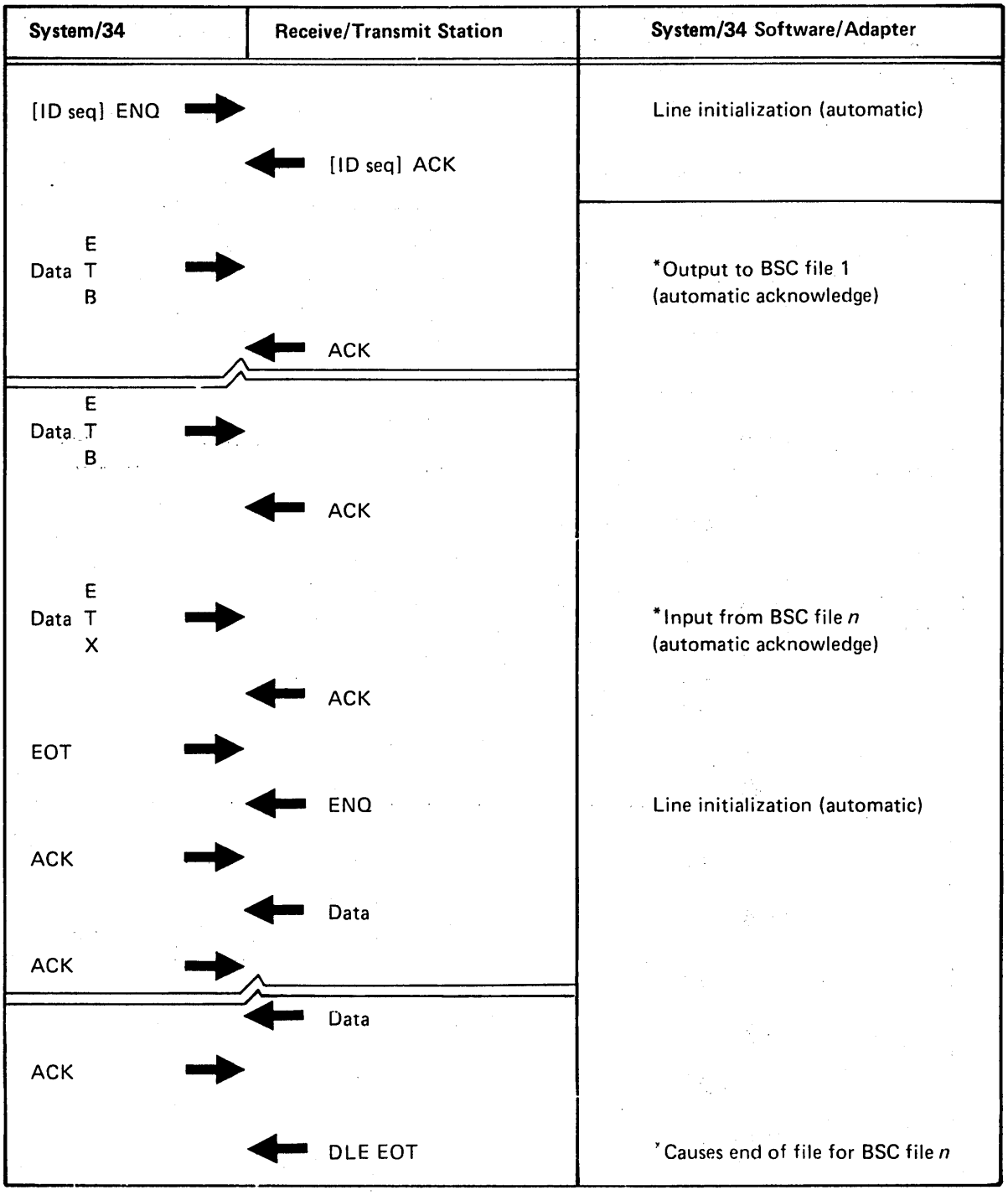
Receive, System/34 Calling Station



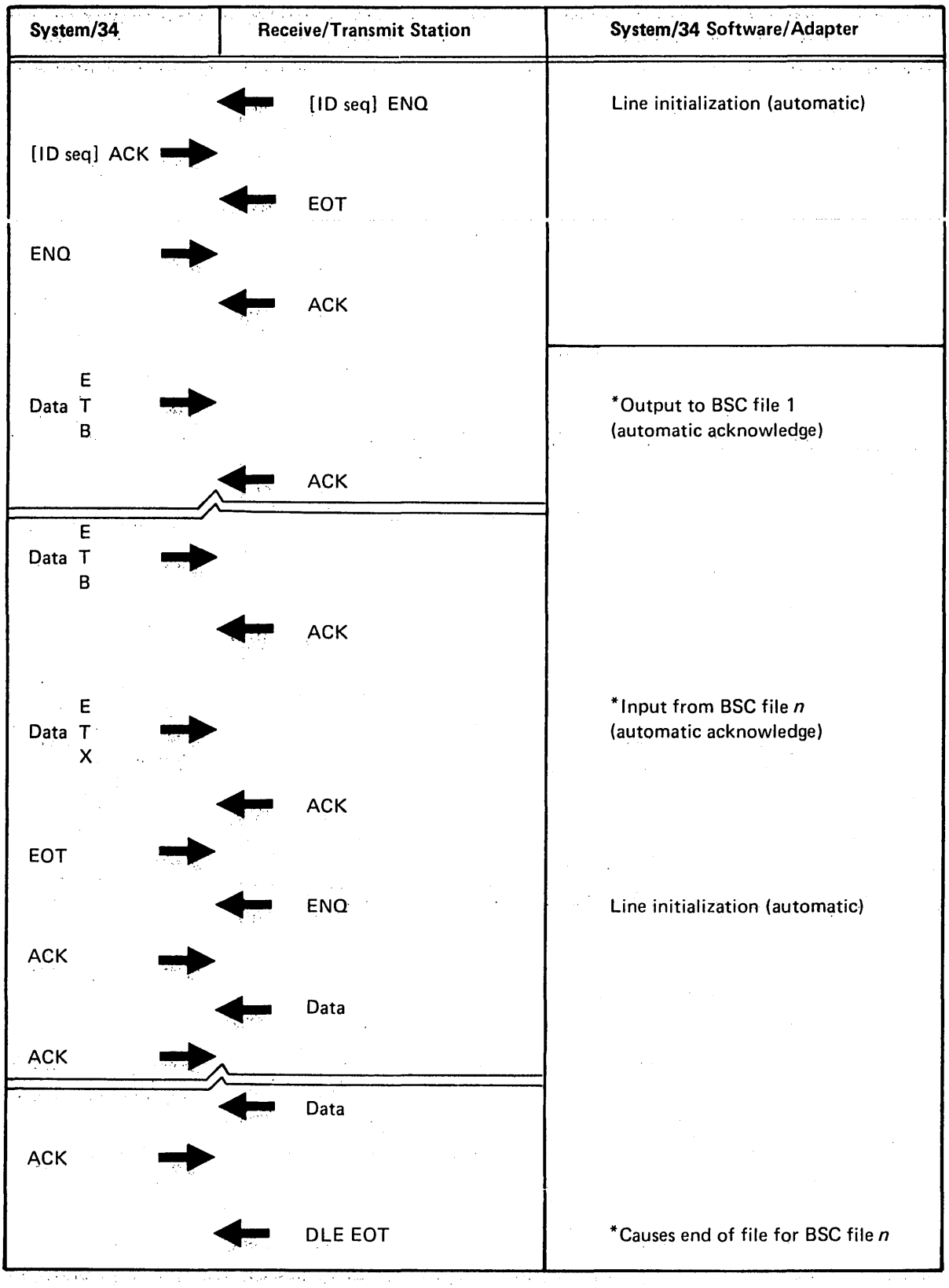

System/34
Receive Station
System/34 Software/Adapter
[ID seq] ENQ
Line initialization (automatic)
[ID seq] ACK
EOT
ENQ
ACK
Data ETB
*Output to BSC file
(automatic acknowledge)
ACK
Data ETB
ACK
Data ETX
*End of job
ACK
DLE EOT

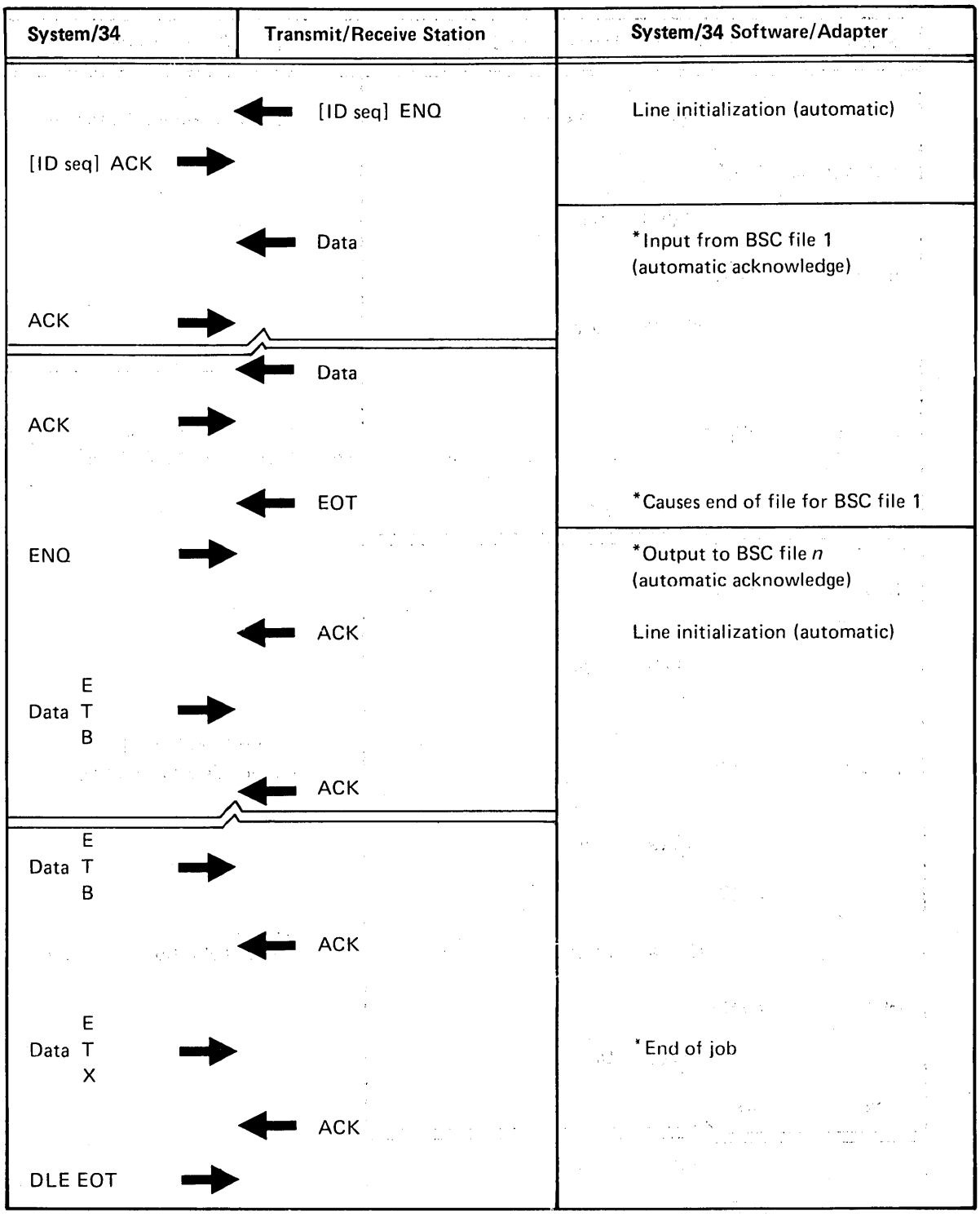
Transmit, System/34 Calling Station



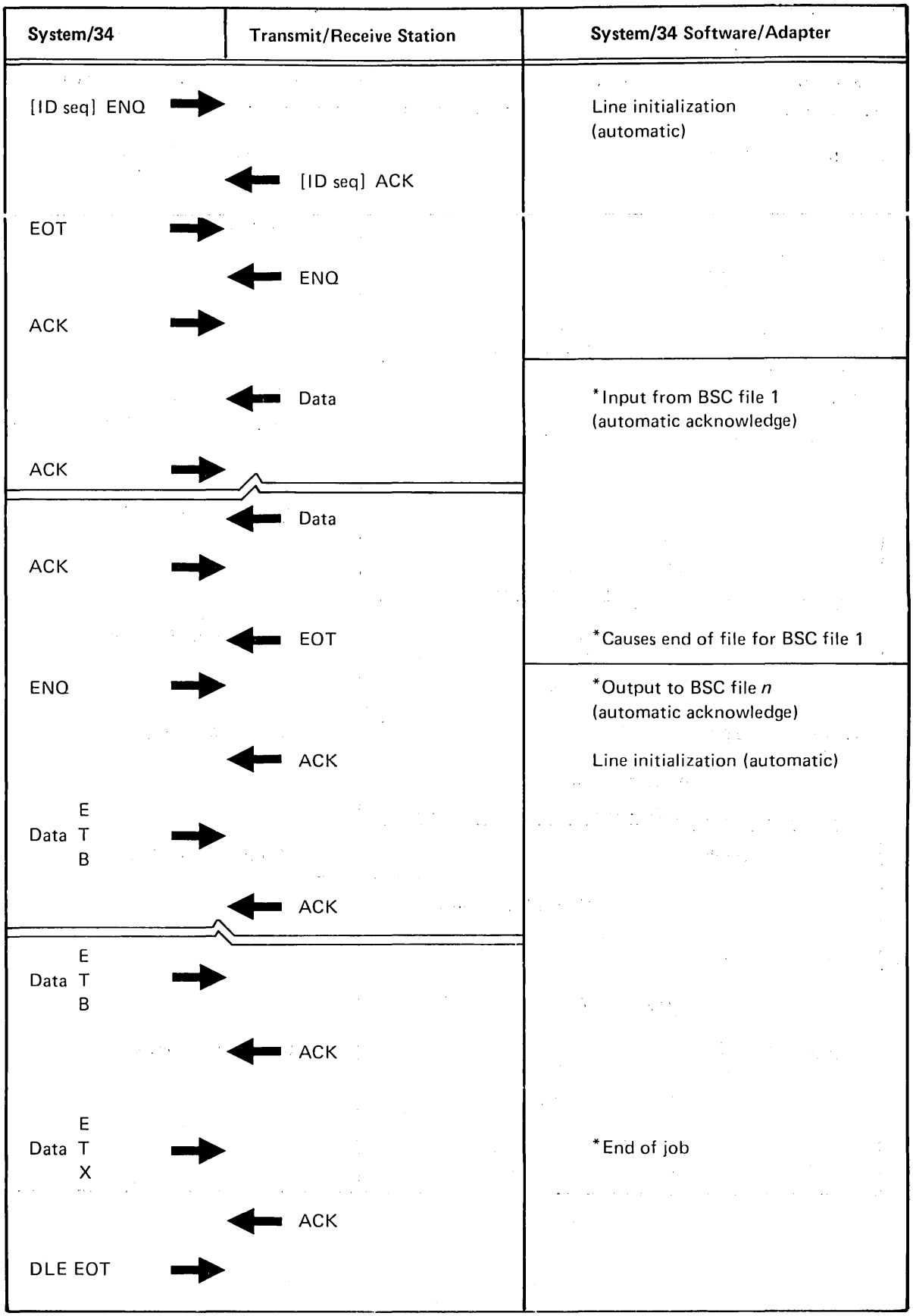

System/34
Receive/Transmit Station
System/34 Software/Adapter
[ID seq] ENQ
[ID seq] ACK
Line initialization (automatic)
Data ETB
ACK
*Output to BSC file 1 (automatic acknowledge)
Data ETB
ACK
Data ETX
ACK
*Input from BSC file n (automatic acknowledge)
EOT
ENQ
Line initialization (automatic)
ACK
Data
ACK
Data
ACK
DLE EOT
*Causes end of file for BSC file n

Transmit/Receive, System/34 Answer Station



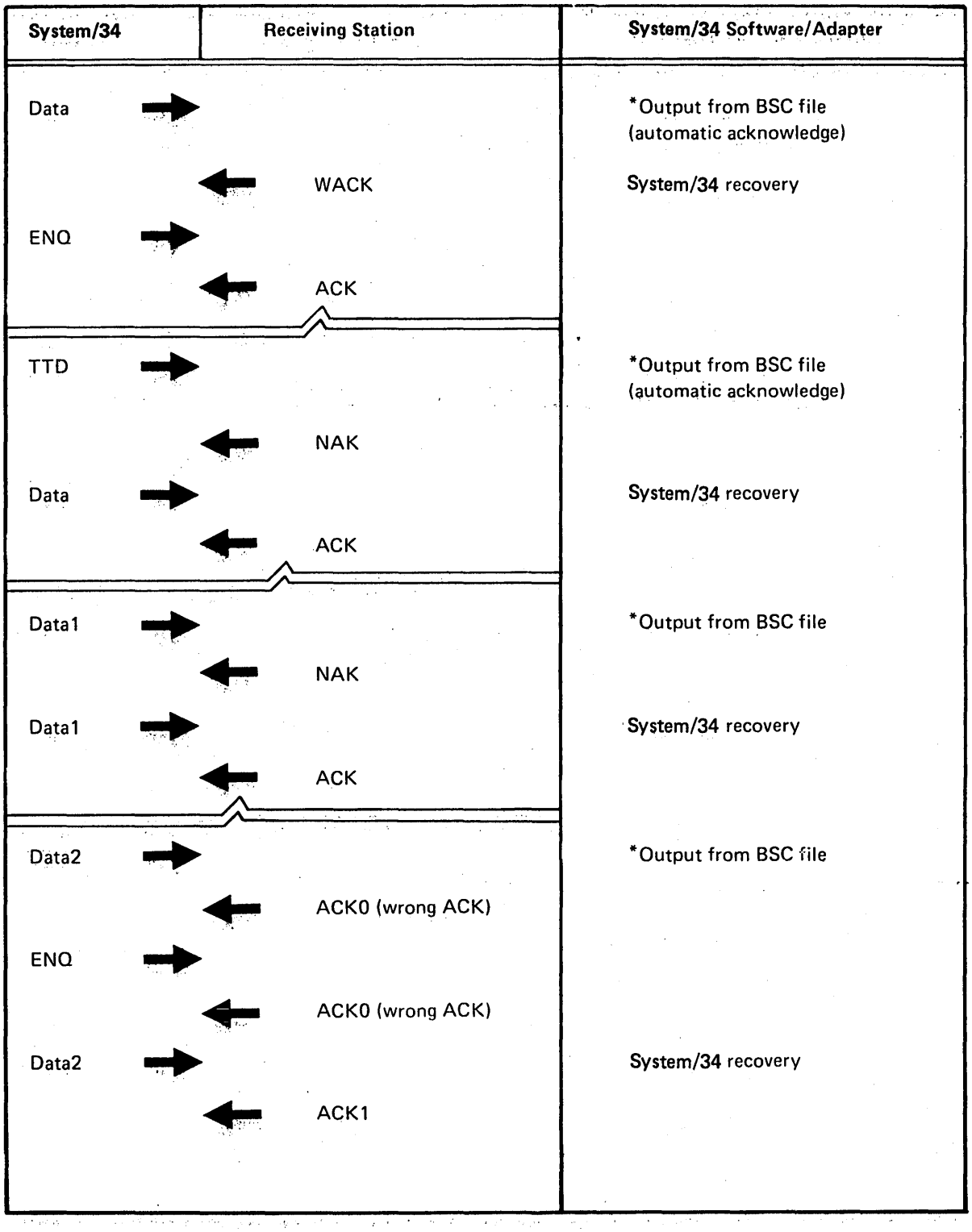

System/34
Transmit/Receive Station
System/34 Software/Adapter
[ID seq] ENQ
Line initialization (automatic)
[ID seq] ACK
Data
*Input from BSC file 1
(automatic acknowledge)
ACK
Data
ACK
EOT
*Causes end of file for BSC file 1
ENQ
*Output to BSC file n
(automatic acknowledge)
ACK
Line initialization (automatic)
Data ETB
ACK
Data ETB
ACK
Data ETX
*End of job
ACK
DLE EOT

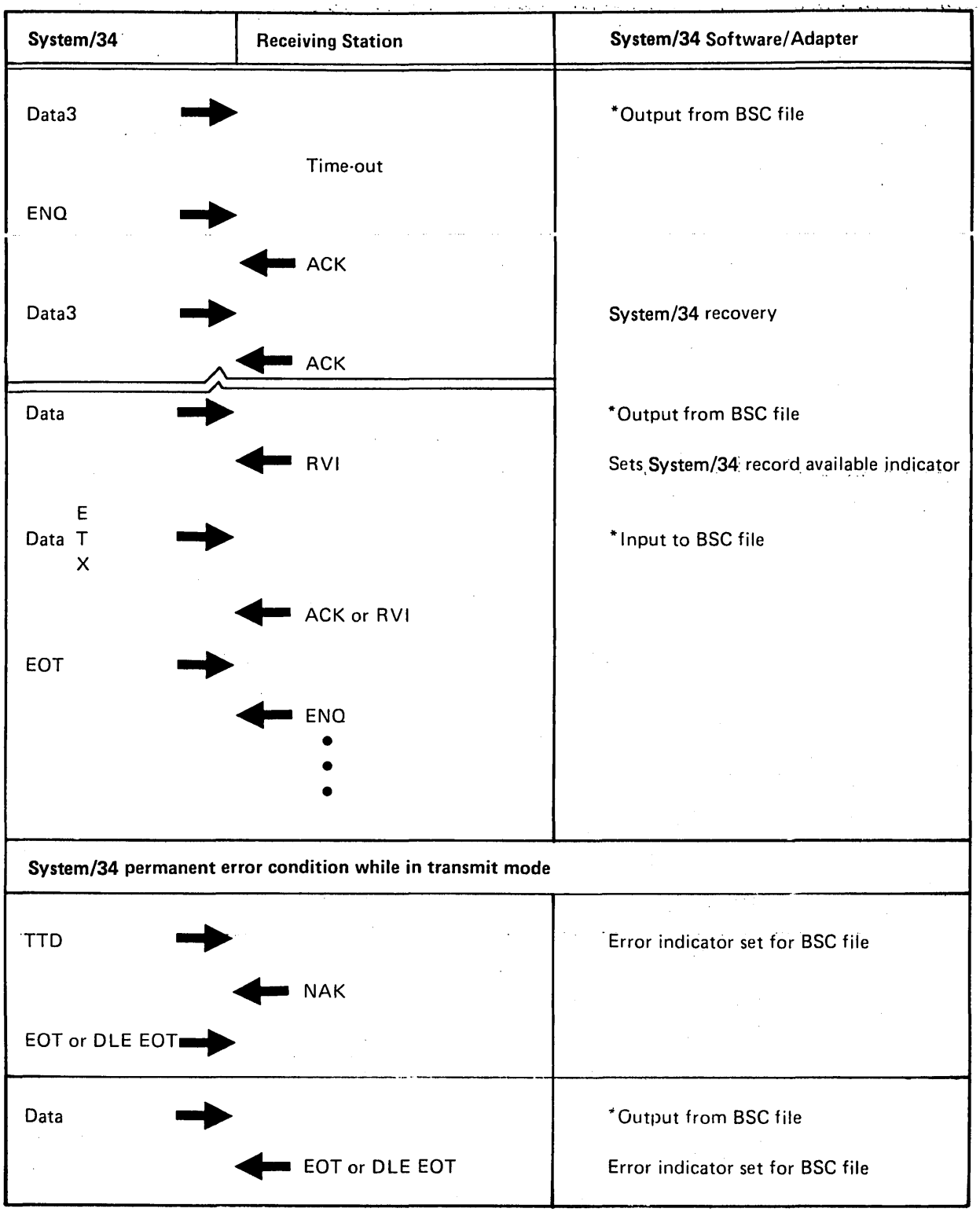
Receive/Transmit, System/34 Calling Station

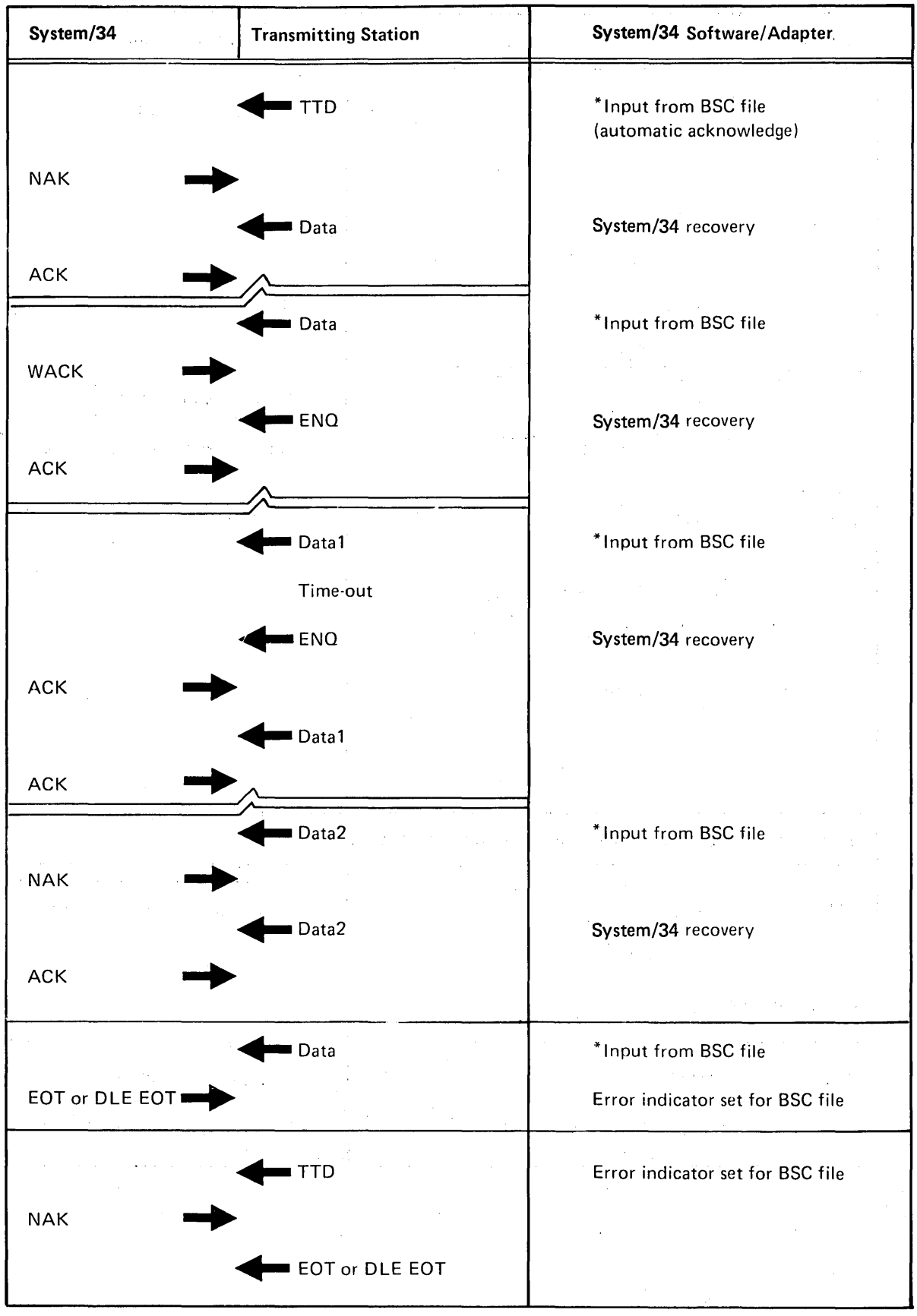


LINE CONDITION–SYSTEM/34 RESPONSES

Data Mode, System/34 Transmitting

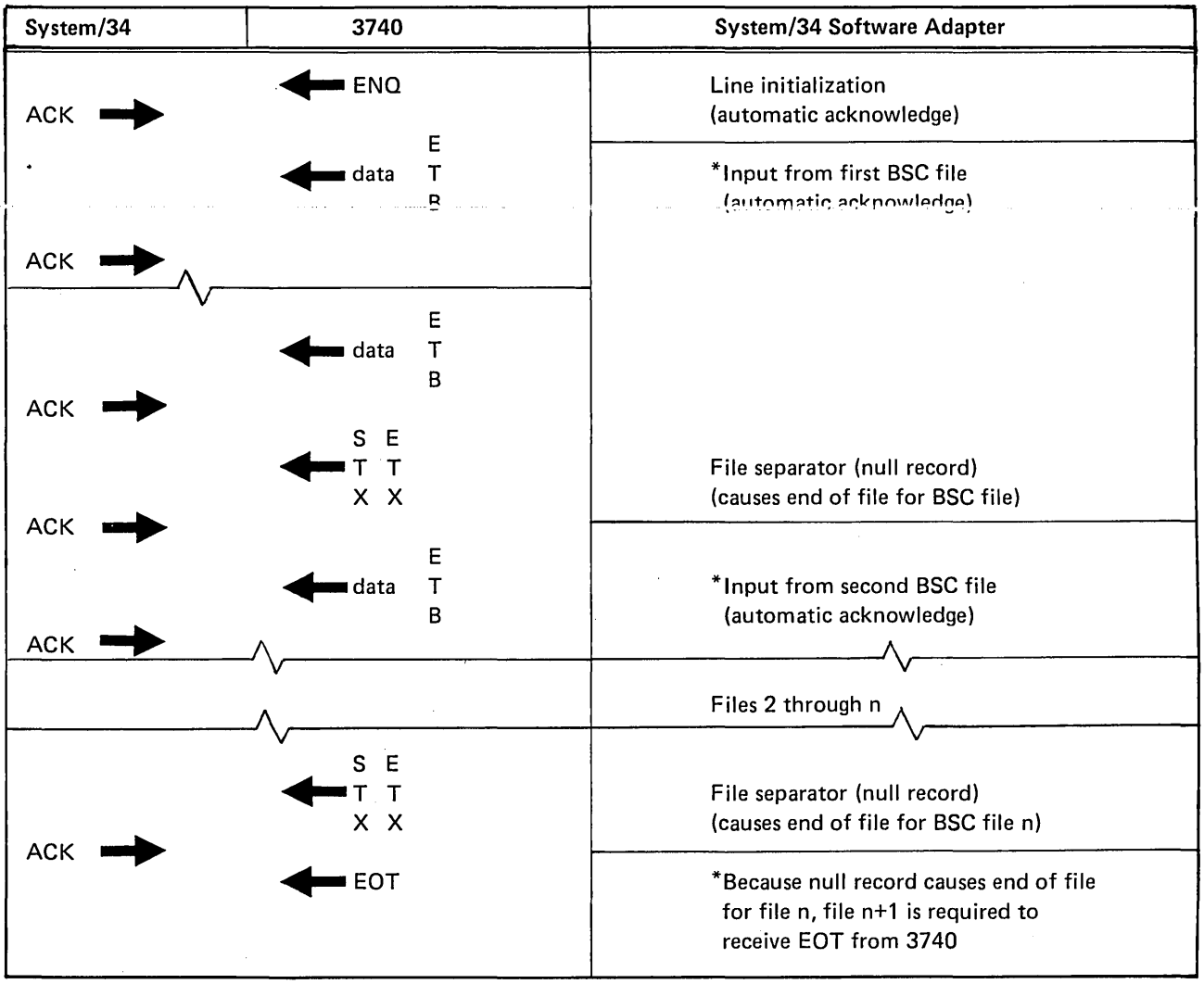



System/34
Receiving Station
System/34 Software/Adapter
Data3
*Output from BSC file
Time-out
ENQ
ACK
Data3
System/34 recovery
ACK
Data
*Output from BSC file
RVI
Sets System/34 record available indicator
Data ETX
*Input to BSC file
ACK or RVI
EOT
ENQ
System/34 permanent error condition while in transmit mode
TTD
Error indicator set for BSC file
NAK
EOT or DLE EOT
Data
*Output from BSC file
EOT or DLE EOT
Error indicator set for BSC file

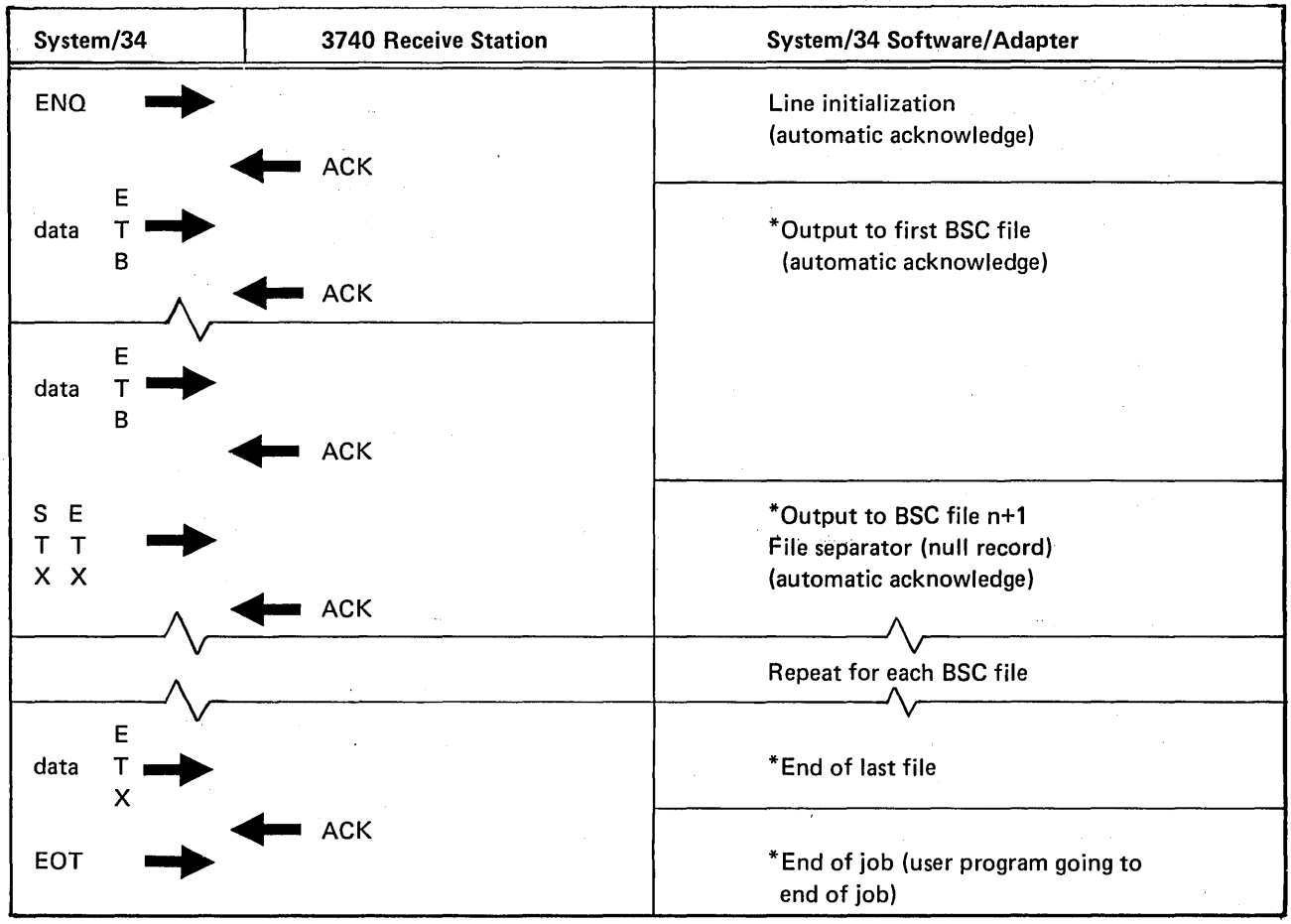

System/34
Transmitting Station
System/34 Software/Adapter
TTD
*Input from BSC file
(automatic acknowledge)
NAK
Data
System/34 recovery
ACK
Data
*Input from BSC file
WACK
ENQ
System/34 recovery
ACK
Data1
*Input from BSC file
Time-out
ENQ
System/34 recovery
ACK
Data1
ACK
Data2
*Input from BSC file
NAK
Data2
System/34 recovery
ACK
Data
*Input from BSC file
EOT or DLE EOT
Error indicator set for BSC file
TTD
Error indicator set for BSC file
NAK
EOT or DLE EOT

3740 MULTIPLE FILE SUPPORT

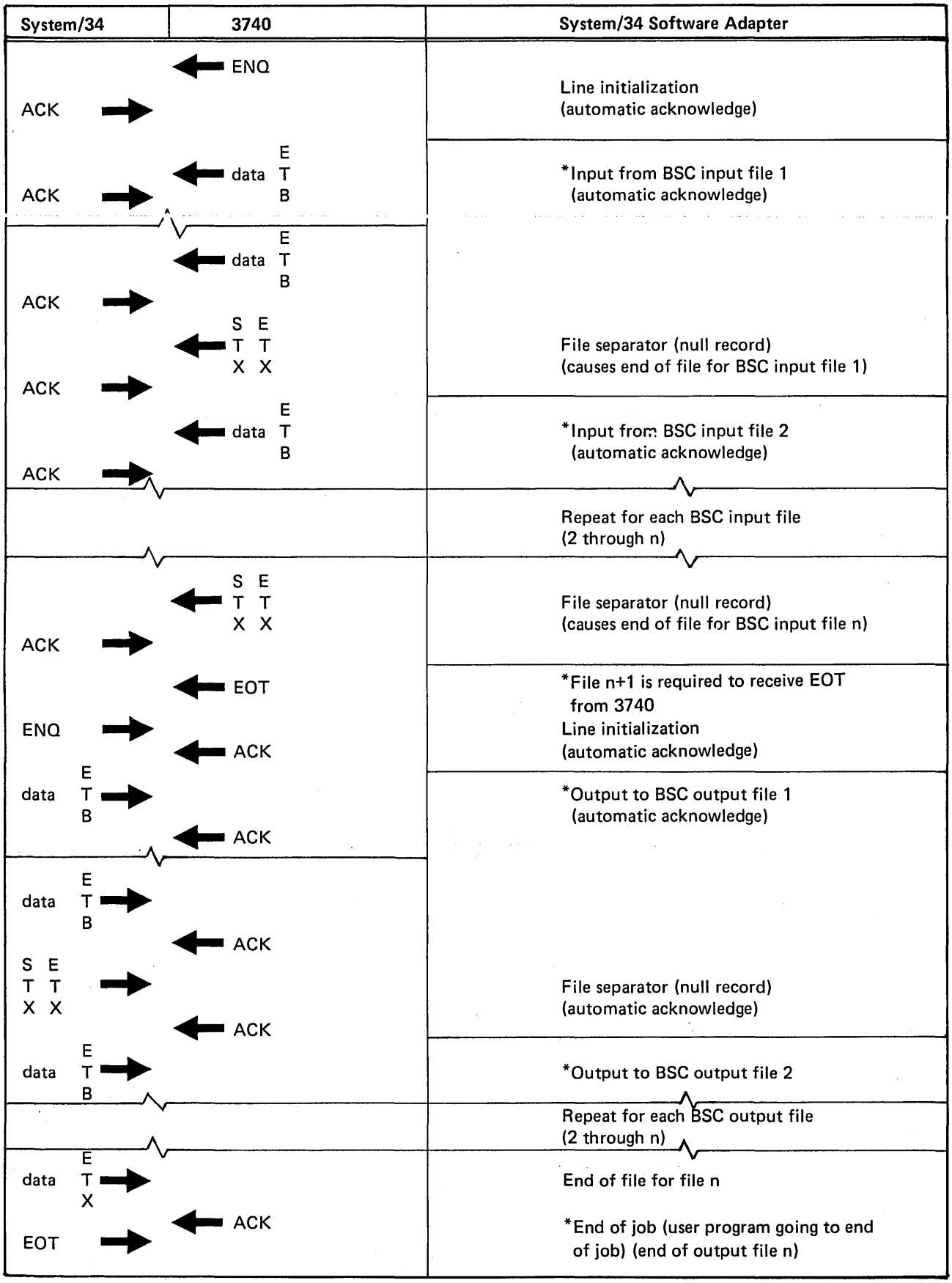
System/34 Receives Multiple Files from 3740



System/34 Transmits Multiple Files to 3740

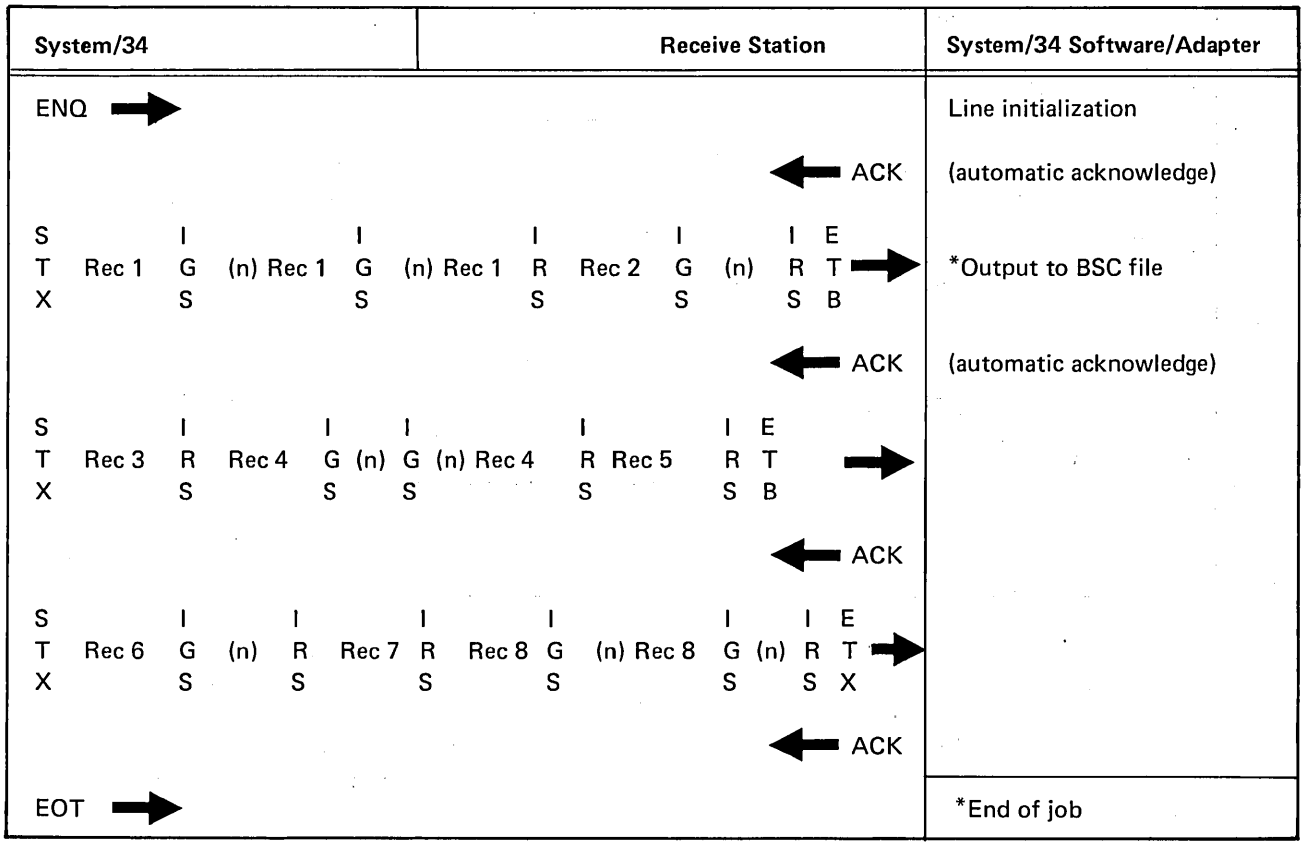


System/34 Receives Multiple Files from and Transmits Multiple Files to 3740



BLANK COMPRESSION/EXPANSION

Transmitting Blocked Records with Compressed Blanks (Using 3780 Protocol Format)

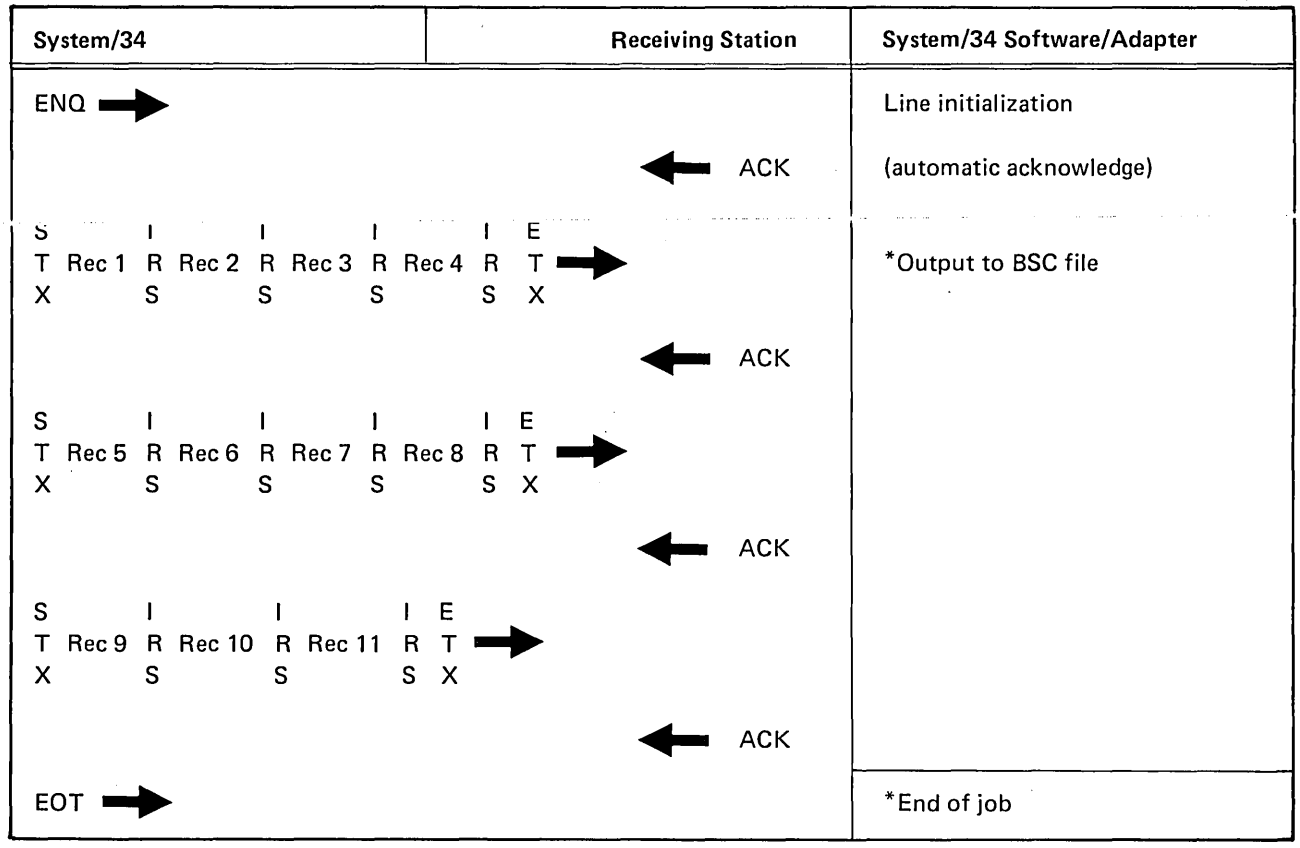


Notes:

- The (n) represents the number (one byte) of compressed blanks. Maximum is 63. An X'40' is always added to the count. For example, if 20 (X'14') blanks are compressed, the count is X'54'.
- Records cannot span blocks.
- Compression cannot be used with transparent output.

BLANK TRUNCATION

Transmitting Blocked Records with Truncated Blanks (Using 3780 Protocol Format)



Appendix E. MRJE BSC Line Protocol

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.

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 acknowledgement) 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.

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 deactivated).
	← 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 deactivates the printer task.
ACK0 →		System/34 sends ACK0 (note 1).
	← ACK0	Host system sends ACK0 (note 1).

SIGN OFF

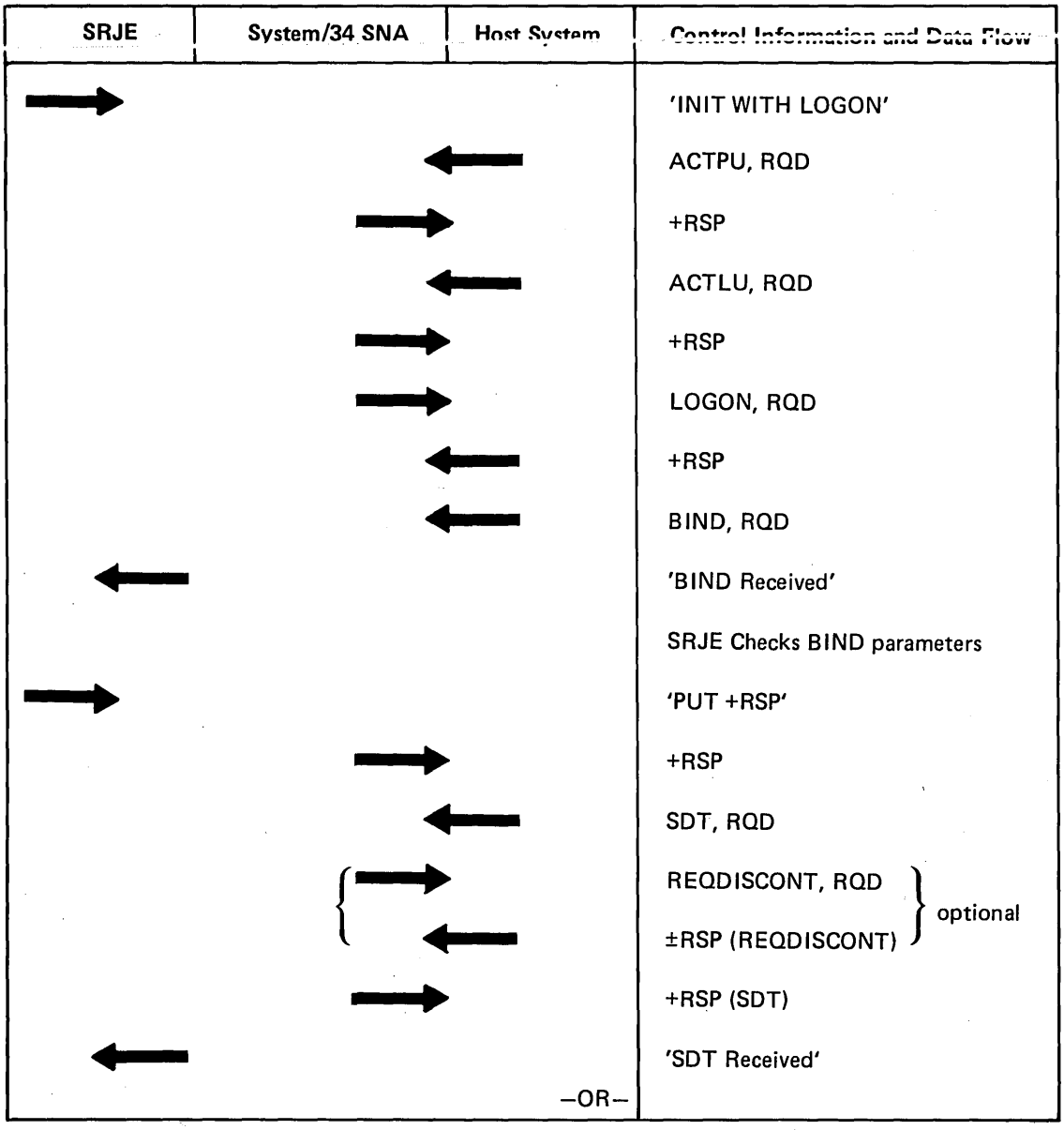
System/34	Host System	Comment
RFT RD1 →		System/34 sends an RFT for the host system RD1 task.
	← GFT RD1	Host system grants the RD1 to the System/34.
SIGN OFF →		System/34 sends a signoff sequence.
	← ACK0	Host system sends an ACK0.

The ACK0 sequence continues until the line is dropped.

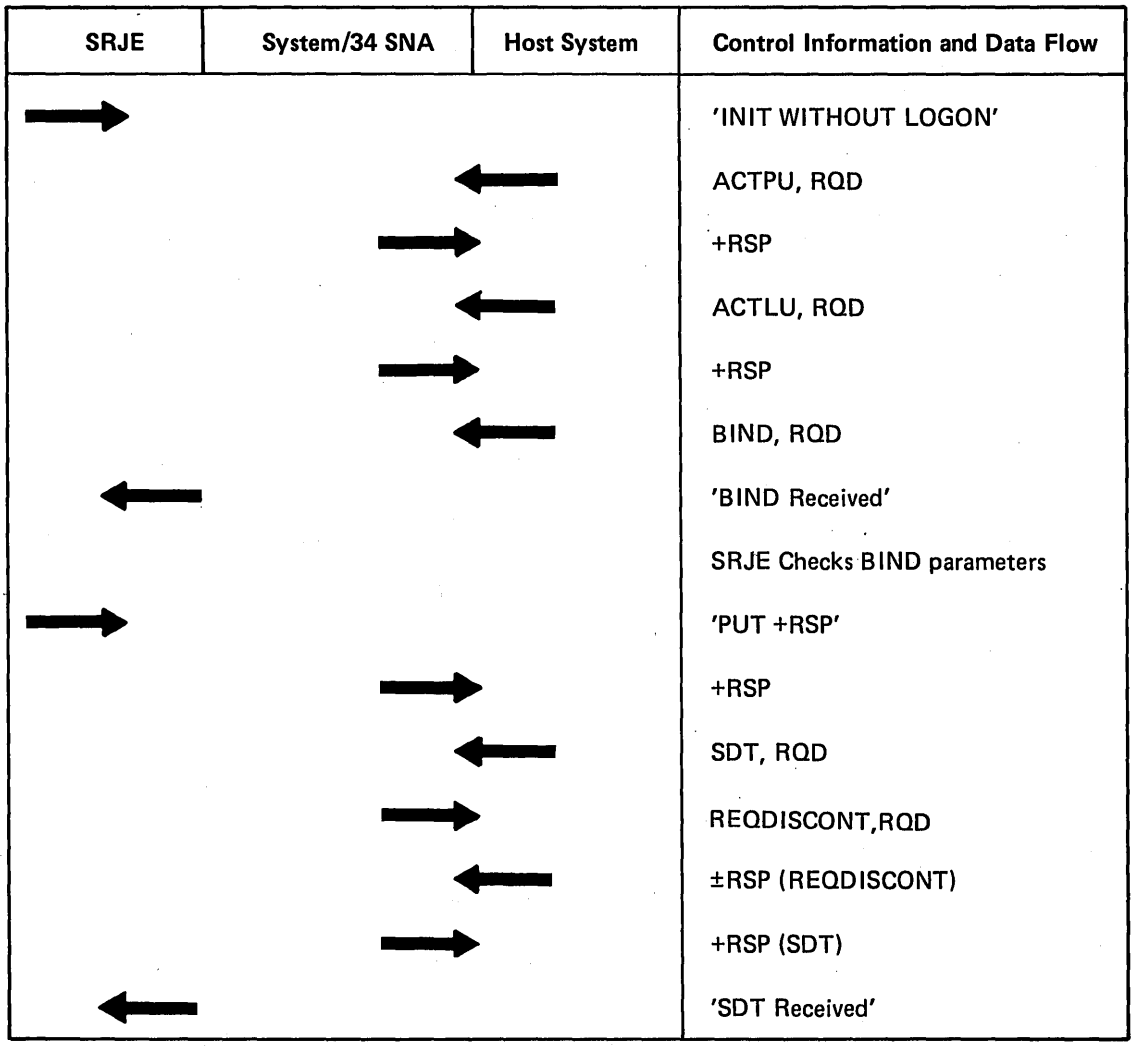
Notes:

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 acknowledgement (ACK0 is not required).

SRJE Session Initialization



SRJE Session Initialization (continued)



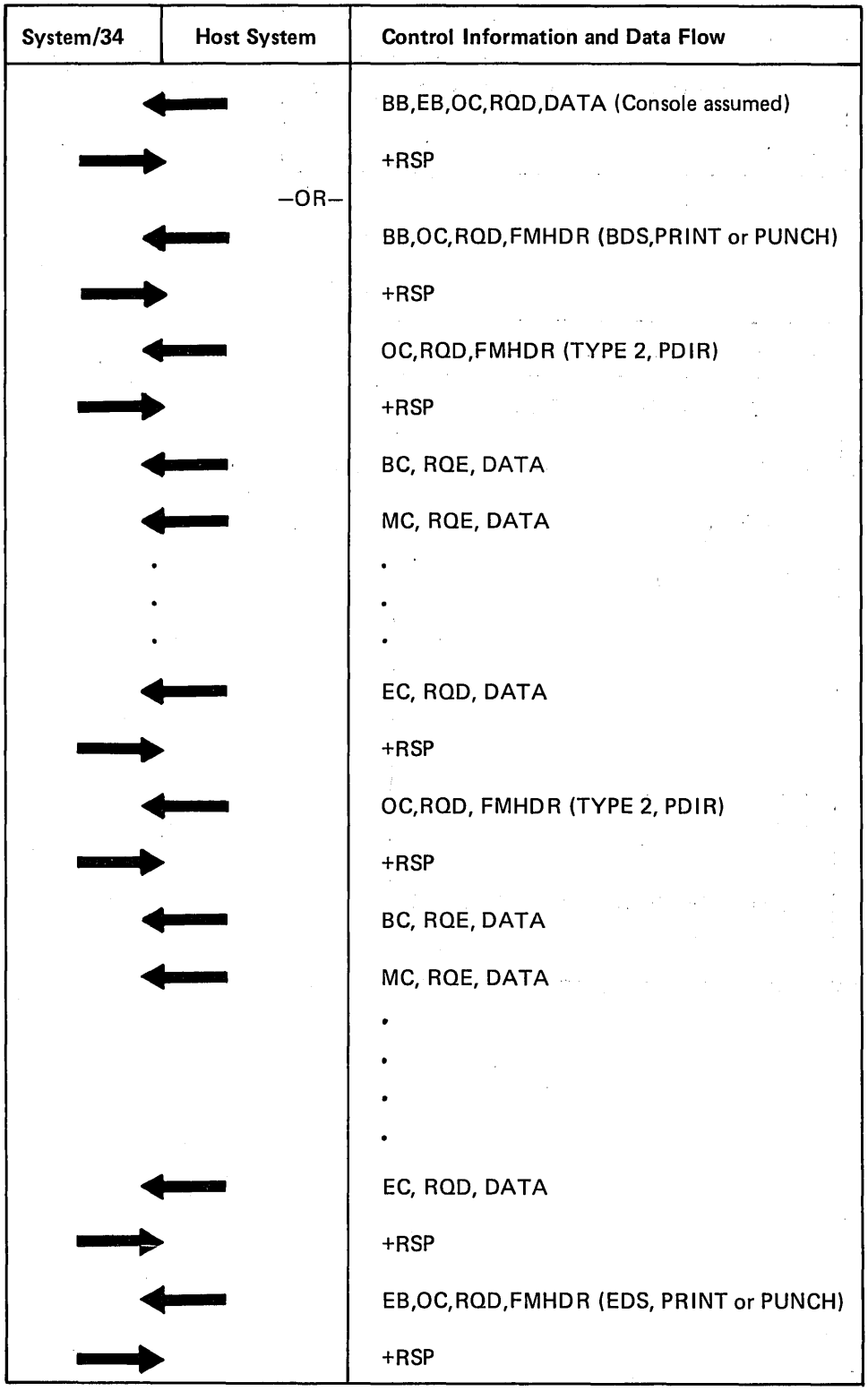
SRJE Initiated Bracket (Console)

System/34	Host System	Control Information and Data Flow
→		BB,EB,OC,RQD,FMHDR (BEDS, CONSOLE), DATA
←		+RSP

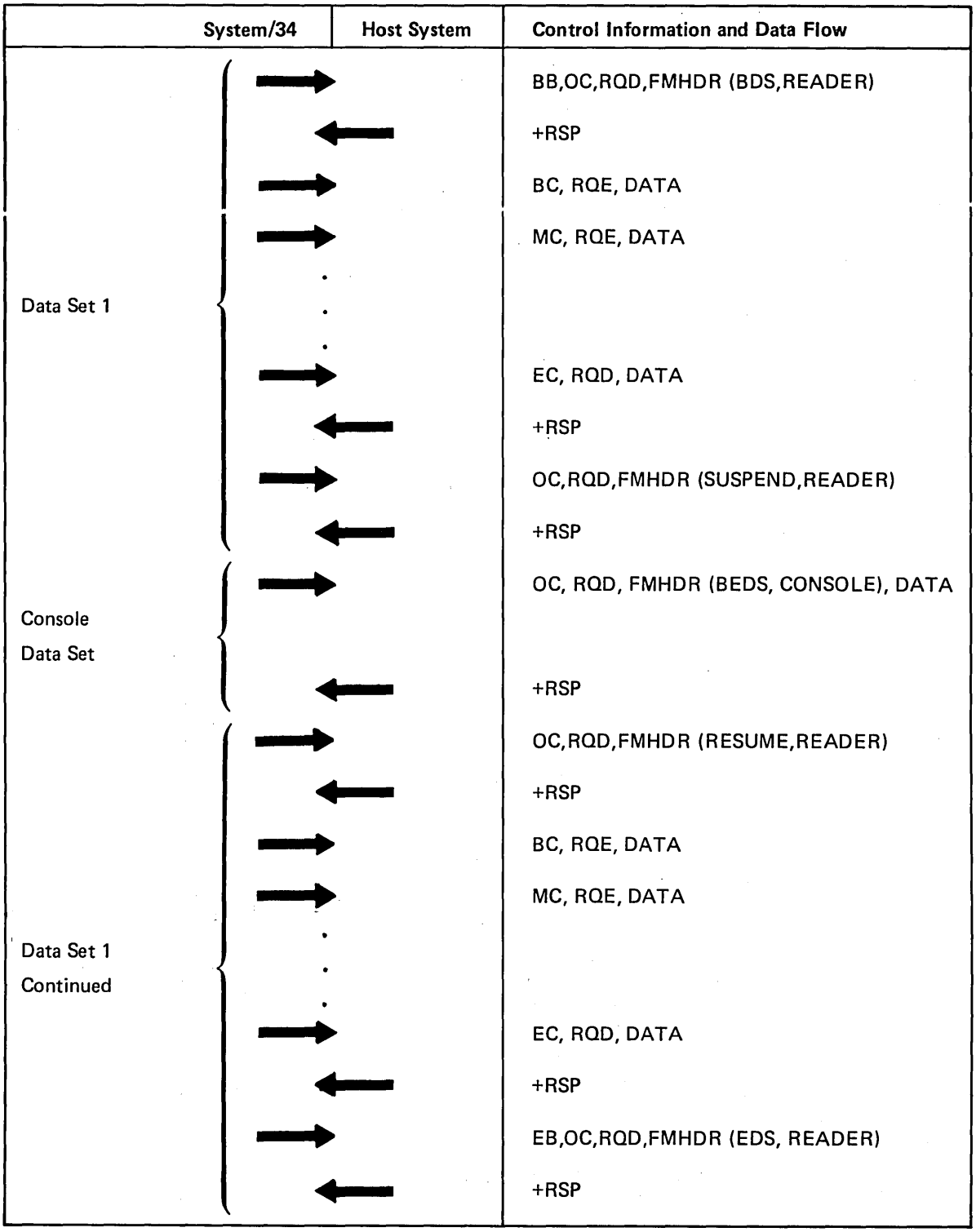
SRJE Initiated Bracket (Reader)

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

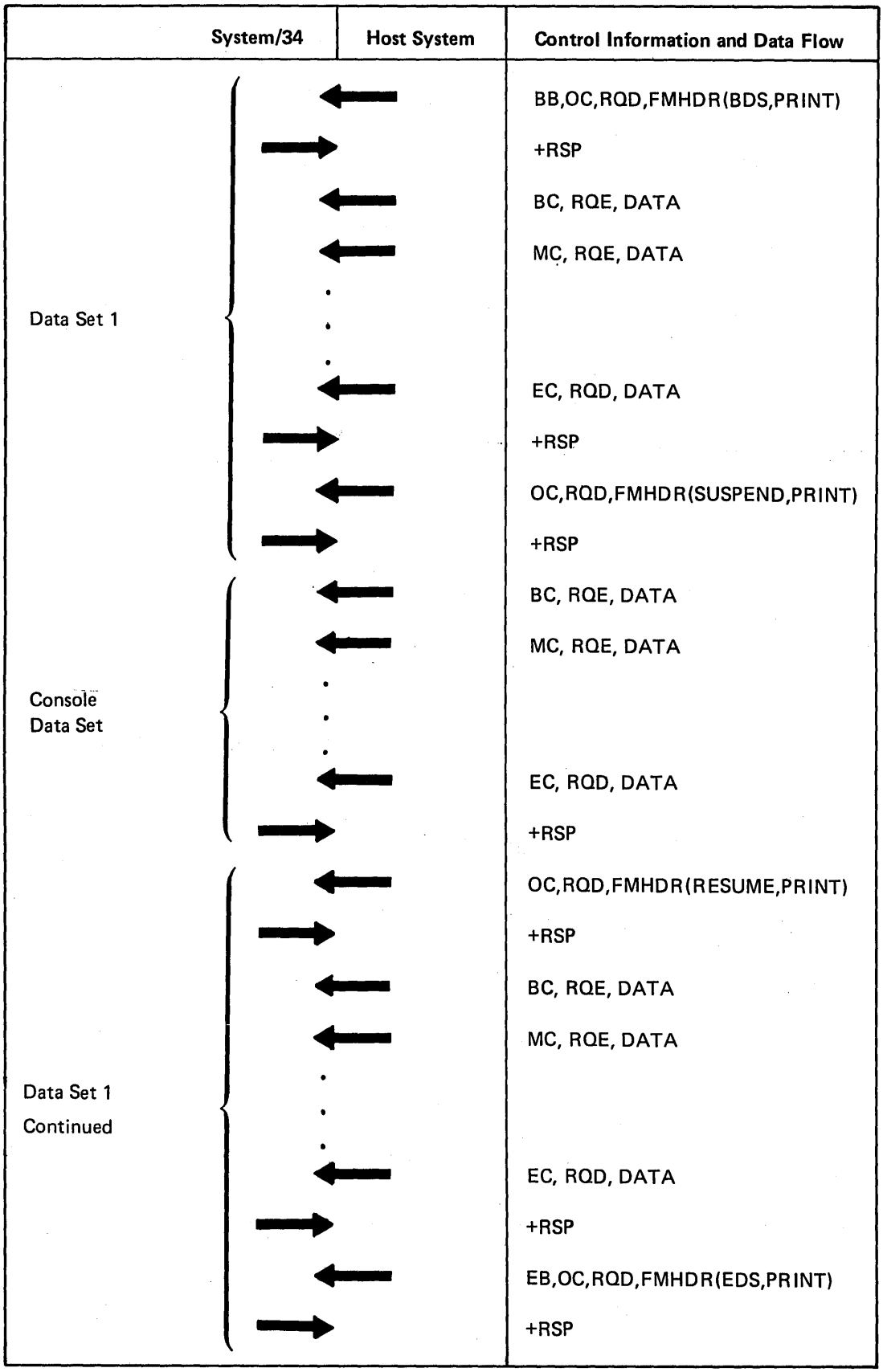
Host System Initiated Bracket



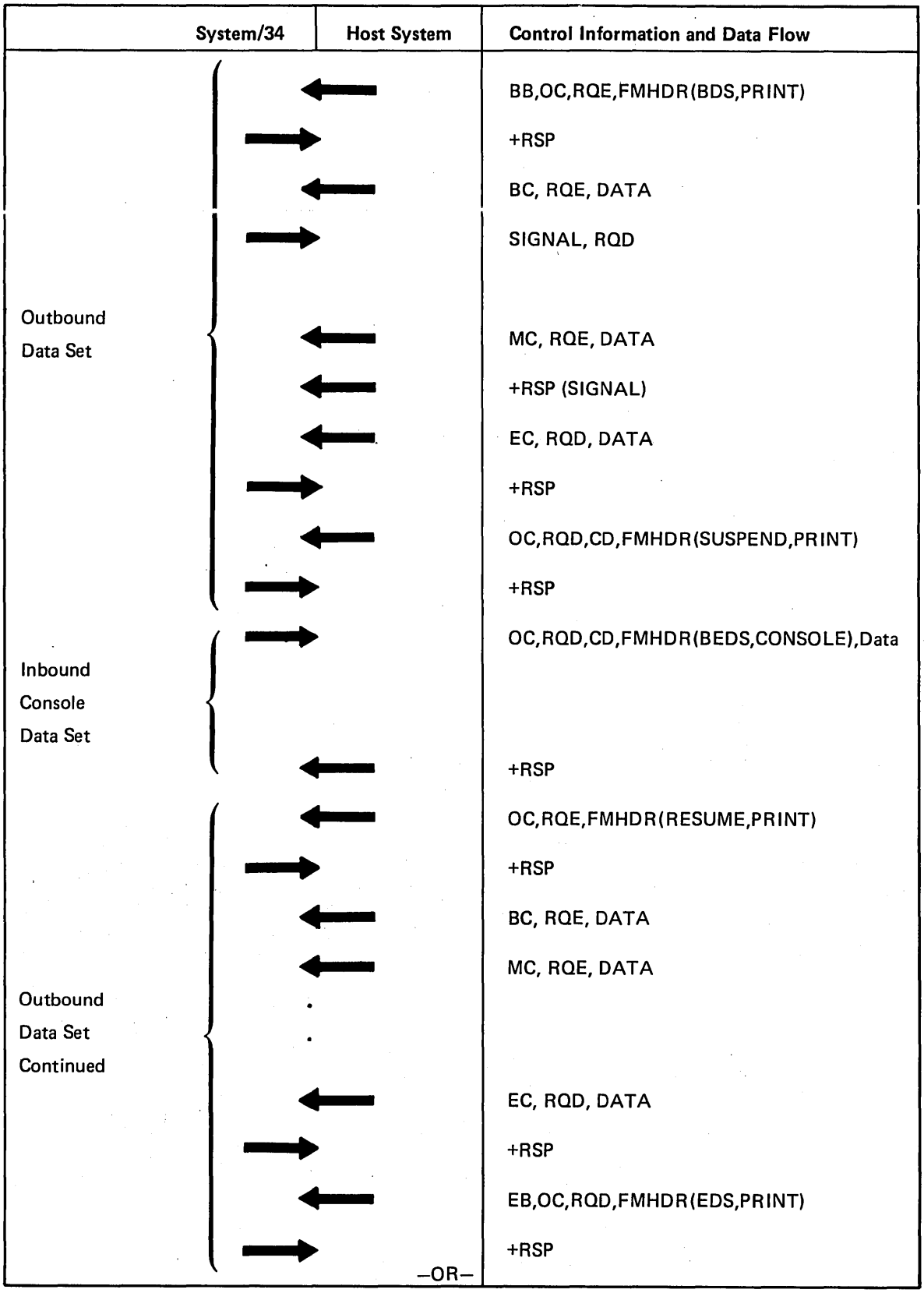
SRJE Interrupted Inbound Data



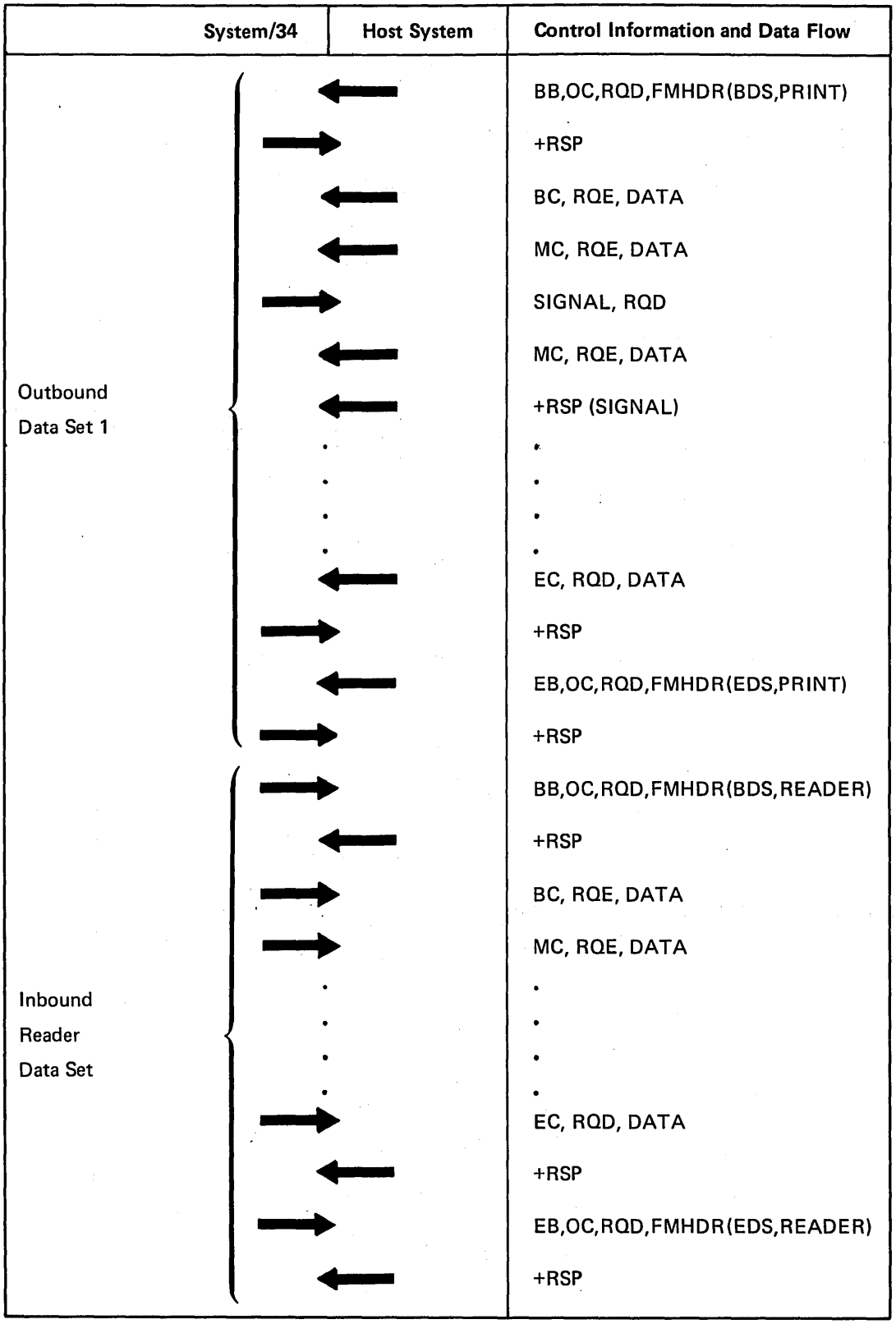
Host System Interrupted Outbound Data



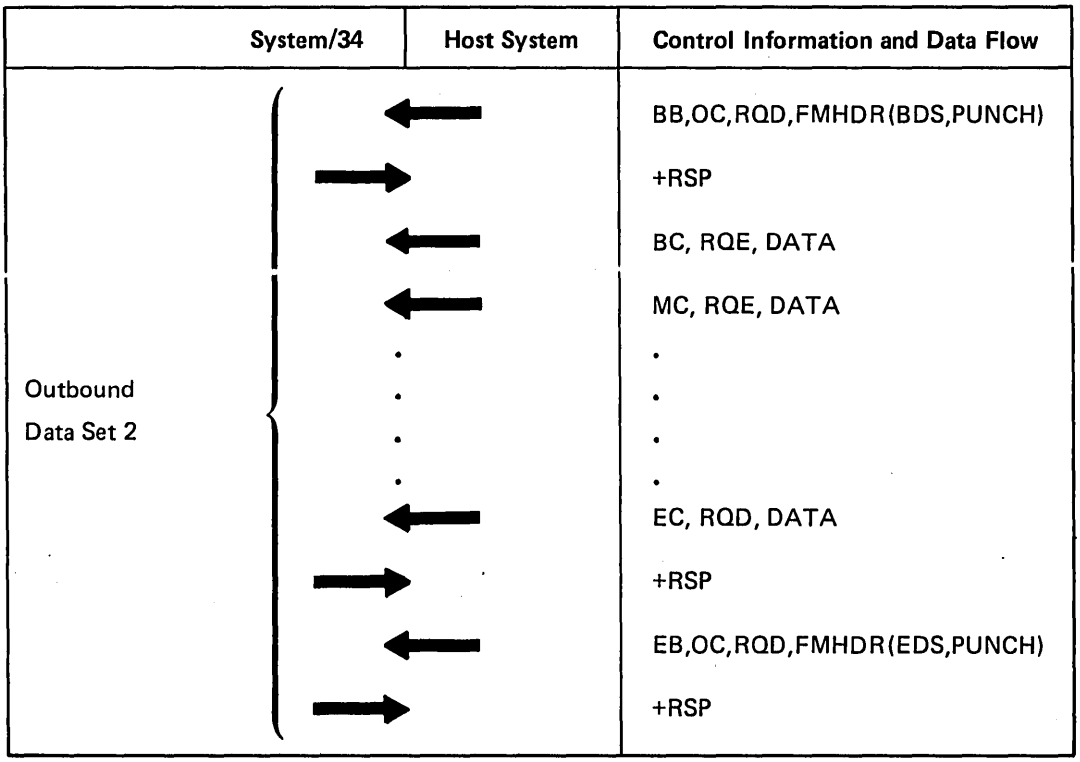
SRJE Requested Direction Change



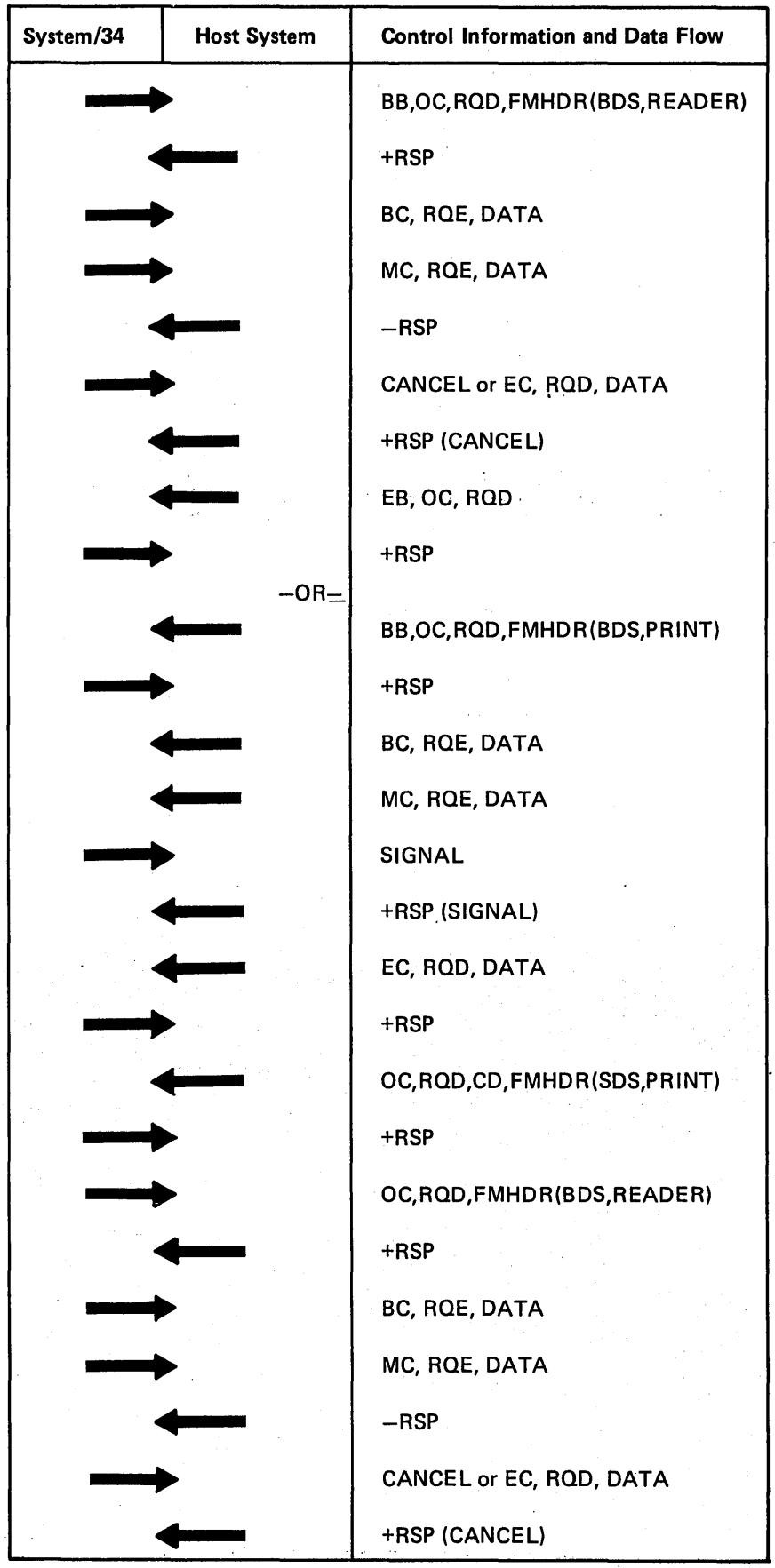
SRJE Requested Direction Change (Continued)



SRJE Requested Direction Change (Continued)



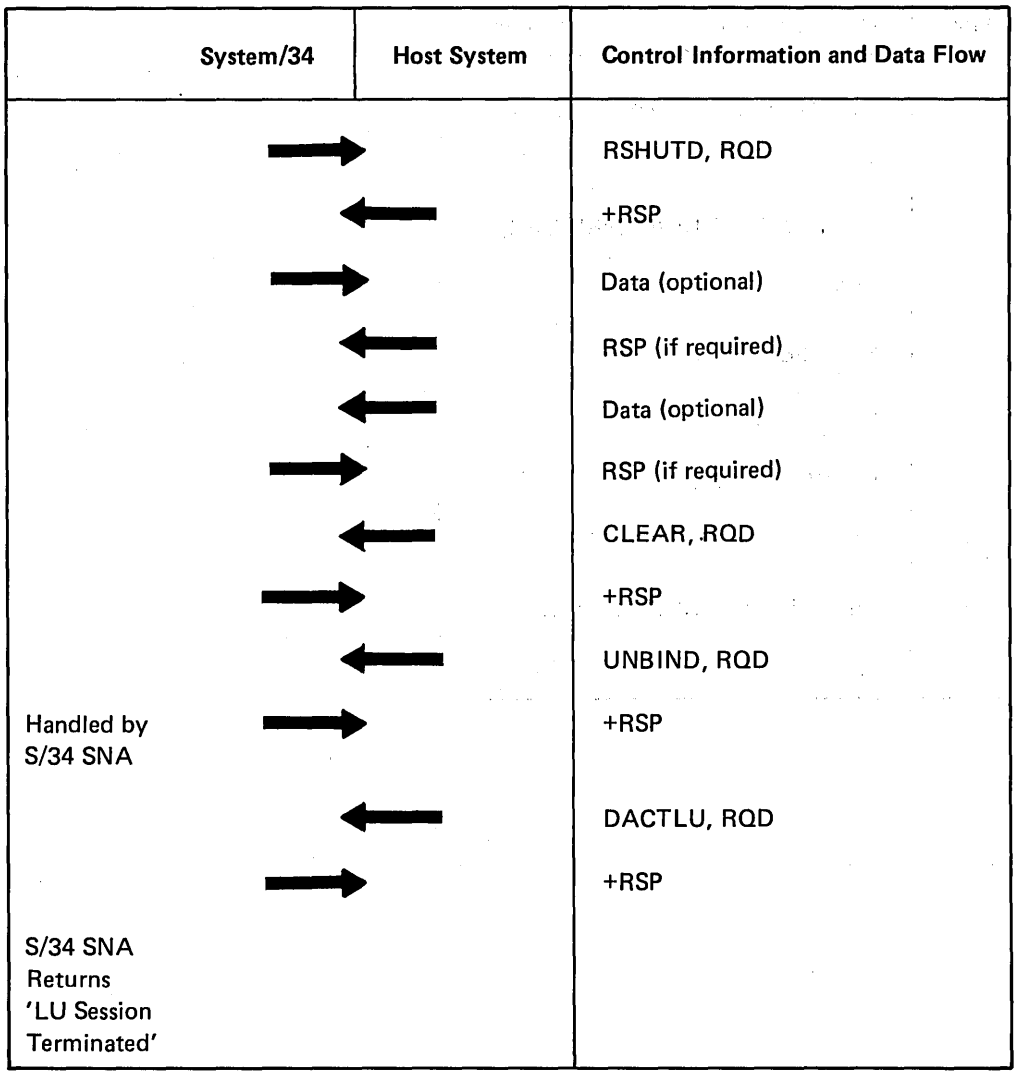
Host System Detected FM Error



Host System Detected FM Error (Continued)

System/34	Host System	Control Information and Data Flow
←		OC,RQD,FMHDR(ADS,READER)
→		+RSP
←		OC,RQD,FMHDR(RDS,PRINTER)
→		+RSP
←		BC, RQE, DATA
←		MC, RQE, DATA
←		EC, RQD, DATA
→		+RSP
←		EB,OC,RQD,FMHDR(EDS,PRINTER)
→		+RSP

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.

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.

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.

CCP: Communications control program.

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.

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: Data-Phone* Digital Service Adapter (*Trademark of American Telephone & Telegraph Co.).

DISC: The transmission control sequence for disconnect on a switched line.

disconnect timeout: 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.

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.

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.

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.

receive mode: A time during which the communications adapter looks for synchronization characters and then stores the data characters in main storage.

receive timeout: An indication that no data has been received by this communications adapter in a given period of time.

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.

RSP: Response.

RVI: Reverse interrupt character.

SDT: Start data traffic.

session: The period of time during which programs or devices can communicate with each other.

SOH: Start-of-heading character.

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.

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.

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.

VM: Virtual machine.

VS: Virtual storage.

VTAM: Virtual telecommunications access method.

WACK (wait before transmit positive acknowledgment): The DLE sequence sent by receiving station to indicate that it cannot receive data at present.

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.

Index

Where more than one page reference is given, the major reference is first.

Please use this form only to identify publication errors or request changes to publications. Technical questions about IBM systems, changes in IBM programming support, requests for additional publications, etc, should be directed to your IBM representative or to the IBM branch office nearest your location.

Error in publication (typographical, illustration, and so on). **No reply.**

Page Number *Error*

Inaccurate or misleading information in this publication. Please tell us about it by using this postage-paid form. We will correct or clarify the publication, or tell you why a change is not being made, providec you include your name and address.

Page Number *Comment*

IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

● No postage necessary if mailed in the U.S.A.

Name _____

Address _____

_____

Cut along line

Fold Fold

FIRST CLASS
PERMIT NO. 40
ARMONK, N. Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
General Systems Division
Development Laboratory
Publications, Dept. 245
Rochester, Minnesota 55901

Fold Fold

IBM

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

System/34 Data Communications Reference Manual (File No. S34-36) Printed in USA SC21-7703-1