

GC33-6048-1
File No. S370-36

Systems

**DOS/VS POWER/VS
Installation Guide and Reference**

Release 34

IBM

Second Edition (March 1978)

This is a major revision of, and obsoletes, GC33-6048-0 and Technical Newsletter GN33-9233, if installed. The sections "Job Accounting", "Intermediate Storage Requirements", and Appendixes D and E have been added. All changes to the text and illustrations are indicated by a vertical line to the left of the change.

This edition applies to version 5, release 34, of the Disk Operating System/Virtual Storage, DOS/VS, and to all subsequent versions and releases until otherwise indicated in new editions or Technical Newsletters. Changes are continually made to the information herein; any such changes will be reported in subsequent revisions or Technical Newsletters.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to *IBM Program Product Center, Publications Department (0446), Schwertstrasse 58, D-7032 Sindelfingen, Germany*. Comments become the property of IBM.

Preface

This manual describes the functions of POWER/VS and their usage. POWER/VS is a component program of the DOS/VS System Control Program.

Executing jobs under POWER/VS has the following advantages:

- Faster job throughput because the partition is only used for job execution and needs not to wait for slow unit record (UR) devices.
- Better utilization of UR devices. They can be shared between partitions.
- Support of the major BSC and SNA terminals.

The manual is designed both as a guide to installing POWER/VS under your DOS/VS supervisor and as a reference source in the day-to-day use of POWER/VS as your input and output spooling facility for DOS/VS.

As a system programmer, you will find detailed information concerning:

- The preparation of your DOS/VS supervisor for the installation of POWER/VS.
- The calculation of the real and virtual storage space for the POWER/VS partition.
- The macro instructions provided for the generation of POWER/VS.
- The layout of the POWER/VS account records.

As an application programmer, you will be informed about:

- The use of POWER/VS in application programs, even though these programs might be designed not to execute in a POWER/VS-controlled partition.
- The possibilities of controlling job accounting and output segmentation within your application programs.

As a DOS/VS system operator, you will find detailed information on:

- How to start, control, and terminate the execution of POWER/VS.
- How to start, supervise, and terminate the execution of POWER/VS tasks.
- How to manipulate the characteristics of POWER/VS jobs and to control the processing of POWER/VS queues.

This manual consists of four chapters and five appendixes containing the following information:

Chapter 1 is an introduction to POWER/VS.

Chapter 2 describes how to plan the installation of POWER/VS, how to calculate the POWER/VS storage requirements, and how to code the POWER/VS generation macros. In addition, this chapter lists the terminals that you may use with the POWER/VS RJE support.

Chapter 3 lists the POWER/VS central operator commands, the statements available with the POWER/VS Job Entry Control Language, and the statements provided for the use of the POWER/VS AUTOSTART feature. This chapter also describes the cross-partition communication macros and the macros available for POWER/VS job accounting and output segmentation.

Chapter 4 describes, in many examples, the day-to-day use of POWER/VS and POWER/VS RJE as seen from the central system operator's point of view.

Appendix A is a detailed discussion of a POWER/VS function called data compaction. This function, which is available with the POWER/VS RJE,SNA support, can be selected in the POWER/VS generation procedure.

Appendix B provides specific information concerning the data flows between the central system and remote terminals in the POWER/VS RJE,SNA environment.

Appendix C summarizes the VTAM LOGON BIND requirements with respect to the POWER/VS RJE,SNA support.

Appendix D shows two examples of a status report produced after issuing the PDISPLAY command. These examples apply to the IBM 3800 Printer.

Appendix E provides information on coding and using reader exit routines.

System and application programmers are expected to be familiar with DOS/VS system generation procedures and with multiprogramming functions under DOS/VS. Central system operators should be familiar with the operating procedures in a DOS/VS multipartition environment.

Publications related to the information given in this manual are listed in the Bibliography.

Contents

SUMMARY OF AMENDMENTS	ix
CHAPTER 1. INTRODUCTION TO POWER/VS.	1
Hardware Supported by POWER/VS.	1
Local Unit Record I/O Devices	2
Direct Access Storage Devices	2
Remote Job Entry Devices.	2
Concepts and Functions.	3
User Responsibilities	7
Input/Output Description.	7
Environmental Considerations.	8
Programming Systems	8
System Requirements	9
Timing and Throughput Information	9
CHAPTER 2. INSTALLING POWER/VS	10
Planning the System	10
Machine Requirements.	10
Intermediate Storage Requirements	11
Size of the Data File and Queue File.	11
Block Size of the Data File	12
Number of Track Groups.	13
Size of the Account File.	13
Storage Requirements.	13
Storage Requirements Calculation.	18
DOS/VS Supervisor Generation Requirements	22
Special Considerations for DOS/VS Functions	22
VTAM and NCP Generation Requirements.	23
Generating POWER/VS	24
POWER/VS Generation Macros.	24
Conventions	24
POWER Macro	25
PLINE Macro	32
PRMT Macro.	33
PCPTAB Macro.	38
POWER/VS Phases	41
POWER/VS Generation Procedure	42
Terminals Supported by POWER/VS	44
IBM 2770 Data Communication System.	44
IBM 2780 Data Transmission Terminal (Model 1 or 2).	45
IBM 3741 Data Station (Model 2 or 4).	46
IBM 3771 Communication Terminal	46
IBM 3773 Communication Terminal	46
IBM 3774 Communication Terminal	47
IBM 3775 Communication Terminal	47
IBM 3776 Communication Terminal	47
IBM 3780 Data Communication Terminal.	48
IBM 3790 Communication System	49
POWER/VS Support for other Devices.	49
IBM 2560 and 5425	49
IBM 3525.	50
IBM 3800.	50
CHAPTER 3. USING POWER/VS.	51
POWER/VS Central Operator Commands.	51
Command Descriptions.	53
PACCOUNT Command.	53
PALTER Command.	54
PBRDCST Command	57

PCANCEL Command	58
PDELETE Command	58
PDISPLAY Command.	59
PEND Command.	63
PFLUSH Command.	64
PGO Command	65
PINQUIRE Command.	66
PRELEASE Command.	67
PRESTART Command.	68
PSETUP Command.	69
PSTART Command.	70
PSTOP Command	73
POWER/VS Job Entry Control Language Statements.	75
Functions of JECL Statements.	77
Format of JECL Statements	77
Using JECL Statements	79
* \$\$ CTL Statement.	81
* \$\$ DATA Statement	81
* \$\$ EOJ Statement.	82
* \$\$ JOB Statement.	82
* \$\$ LST (* \$\$ PRT) Statement	85
* \$\$ PUN Statement.	95
* \$\$ RDR Statement.	100
* \$\$ SLI Statement.	101
JECL Delimiter Statements	105
Cross-Partition Communication Macros.	106
Macro Descriptions.	107
SPL Macro	107
PUTSPOOL Macro.	109
GETSPOOL Macro.	111
CTLSPPOOL Macro.	112
Return Codes from PUTSPOOL, GETSPOOL, and CTLSPPOOL.	114
Job Accounting.	116
Reader Account Record	116
List Account Record	117
Punch Account Record.	119
Execution Account Record.	120
RJE Account Records	121
PUTACCT Macro	123
Output Segmentation	124
SEGMENT Macro	124
POWER/VS AUTOSTART Statements	126
Statement Descriptions.	126
FORMAT Statement.	126
READER Statement.	127
PRINTERS Statement.	127
PUNCHES Statement	128
CHAPTER 4. OPERATING A SYSTEM CONTROLLED BY POWER/VS	129
Assumptions	129
Initiating POWER/VS	129
Initiating POWER/VS without AUTOSTART	130
Initiating POWER/VS with AUTOSTART.	134
Assigning Dummy Devices	135
Changing Partition Priorities	136
Issuing POWER/VS Commands	136
Starting and Stopping a User Partition.	138
Bringing a Partition Under the Control of POWER/VS.	138
Changing the Class Specification for a Partition.	138
Stopping and Restarting a Partition	139
Task Control.	139
Starting a POWER/VS Reader Task	139
Starting a POWER/VS Writer (List or Punch) Task	140
Stopping POWER/VS Tasks	140
Remote Job Entry - BSC Terminals.	142

Starting an RJE,BSC Line.	142
Controlling RJE,BSC Operations.	143
Remote Job Entry - SNA Terminals.	143
Establishing the VTAM Interface for POWER/VS with RJE,SNA	143
Controlling RJE,SNA Operations.	144
Stopping an RJE,SNA Session	144
Termination of All RJE,SNA Sessions	145
Terminating POWER/VS.	145
Normal Termination.	145
Emergency Termination	146
Multipurpose Unit-Record Devices.	146
Miscellaneous Devices	147
Using the IBM 3540 Diskette Input/Output Unit	147
The 3540 in SYSIN Mode.	147
The 3540 in Data Mode	148
APPENDIX A. DATA COMPACTION.	151
APPENDIX B. RJE,SNA I/O SPECIFICATIONS	157
Inbound Console Data.	157
Inbound Card Data	157
Outbound Print Data	159
Outbound Punch Data	161
Outbound Console Data	161
Data Flow Control (DFC) Requests.	161
Inbound DFC Requests.	161
Outbound DFC Request.	162
APPENDIX C. LOGON MODE TABLE AND BIND PARAMETER REQUIREMENTS	165
APPENDIX D. EXAMPLES OF STATUS REPORT OUTPUT	167
APPENDIX E. READER EXIT ROUTINE.	169
BIBLIOGRAPHY.	171
INDEX	173

Figures

Figure 1-1.	The Major Functions of POWER/VS.	6
Figure 2-1.	Structure of the Virtual Partition and Storage Requirements	14
Figure 2-2.	Space Calculation for RJE,SNA Support Control Blocks	15
Figure 2-3.	Base Values for the Storage Calculation for the POWER/VS Fixable Area (2 parts).	16
Figure 2-4.	Detailed Storage Calculation Example (2 parts) . . .	19
Figure 2-5.	POWER/VS Storage Calculation Summary	21
Figure 2-6.	POWER/VS Phase Names and Their Functions	41
Figure 2-7.	Work Sheet for Coding POWER/VS Generation Macros (2 parts).	42
Figure 3-1.	POWER/VS Central Operator Commands	52
Figure 3-2.	Examples of the Use of POWER/VS JECL	76
Figure 3-3.	Format of JECL Statements.	78
Figure 3-4.	JECL Statements.	79
Figure 3-5.	PUTSPOOL, GETSPOOL, and CTLSPOOL Return Codes. . . .	115
Figure 3-6.	Reader Account Record.	117
Figure 3-7.	List Account Record.	118
Figure 3-8.	Punch Account Record	119
Figure 3-9.	Execution Account Record	120
Figure 3-10.	RJE,BSC Line Account Record.	121
Figure 3-11.	RJE,SNA Session Account Record	122
Figure 4-1.	Job Stream Example for POWER/VS Initiation	131
Figure 4-2.	Example of a POWER/VS Status Report.	132
Figure 4-3.	POWER/VS Commands.	137
Figure A-1.	Numbers of Master Characters Related to Numbers of Compact Code Characters.	151
Figure A-2.	Example of Compact Code Set.	152
Figure A-3.	Translate-and-Test Table for Compaction (TRTC) . . .	155
Figure A-4.	Translate Table for Compaction (XLTC).	155
Figure B-1.	Inbound Card Data with DOS/VS JCL Job Delimiters . .	158
Figure B-2.	Inbound Card Data with POWER/VS JECL Job Delimiters.	158
Figure B-3.	DPC Request Support for SNA.	161
Figure B-4.	Data Communication Facilities Supported by POWER/VS RJE,SNA.	163
Figure B-5.	SCS Control Characters Supported by POWER/VS RJE,SNA	163
Figure B-6.	Data Interruptions Supported by POWER/VS RJE,SNA . .	164
Figure C-1.	LOGON Mode Table Coding Example.	165
Figure C-2.	BIND Parameters for the Presentation Services. . . .	166

Summary of Amendments

Release 34

Edition GC33-6048-0 describes POWER/VS as a component program of the DOS/VS SCP.

The following enhancements are documented:

- Additional device support
- Increased number of partitions
- Library device independence
- Extended support for RJE,SNA work stations.

TNL GN33-9233 documents changes to support the independent component release (ICR) of the IBM 3800 Printer under POWER/VS.

Edition GC33-6048-1 includes corrections and enhancements to the previous edition.

Changes resulting from the ICR for the IBM 3800 Printer are included in this manual. The pertinent figures and text portions are marked so that users who have not installed the ICR can skip these changes.

The following enhancements are described:

"Job Accounting" in Chapter 3, "Assigning Dummy Devices" and "Changing Partition Priorities" in Chapter 4, "Examples of Status Report Output" in Appendix D, and "Reader Exit Routine" in Appendix E.

Chapter 1. Introduction to POWER/VS

POWER/VS is designed to improve the throughput of a computing system by separating unit-record input and output operations (that is, reading card or diskette input, punching cards, and printing reports) and internal computing operations of application programs in a DOS/VS multiprogramming environment. This process, which is called "spooling" (an acronym derived from the term "Simultaneous Peripheral Operations On-Line"), results in overlapping the unit-record I/O operations for all controlled partitions with the execution of application programs in these partitions. To achieve this, POWER/VS stores all unit-record input and output in input and output queues on disk. When a unit-record device is available to read input or to punch or print output, POWER/VS either stores the input data in the input queue or retrieves list or punch output from an output queue and transfers it to that device. When an application program in a POWER/VS-controlled partition is ready to accept input or is about to produce output, POWER/VS retrieves its input from the queue associated with the application program's partition and stores output in the associated output queue on disk.

Hardware Supported by POWER/VS

POWER/VS operates in virtual mode in one partition of the DOS/VS system. The system dispatching priority of the POWER/VS partition must be higher than the priorities of the partitions it is serving.

POWER/VS maintains input and output queues on direct access storage for the card readers, diskette I/O units (input only), card punches, and printers associated with the partitions under its control. These queues are filled or emptied by the appropriate physical devices. When a program in a POWER/VS-controlled partition makes an input or output request to one of these devices, POWER/VS provides the next record from the appropriate input queue or collects the record and places it in the appropriate output queue. These operations take place at main storage or direct access storage speeds and are overlapped with the execution of programs in other partitions; hence, programs execute faster and take better advantage of the system environment. Application programs executing in POWER/VS-controlled partitions do not require modifications to take advantage of POWER/VS.

The POWER/VS queues are under optional operator control. This allows the DOS/VS system operator to modify the order in which programs are executed by manipulating the queues with POWER/VS commands. Because POWER/VS allows job input and output also from and to remote terminals, the possibility of manipulating queues is also available to terminal operators. Further control is available to the application programmer through the option of using Job Entry Control Language (JECL) statements within the job input stream.

The user generates POWER/VS through the assembly of the POWER/VS macros. These macros allow the user to specify the options appropriate to his system environment. Among the options that the user may select are features such as the inclusion or suppression of job accounting routines, of system network architecture (SNA) support, and others. Details concerning the generation of POWER/VS are described in Chapter 2.

| LOCAL UNIT RECORD I/O DEVICES

POWER/VS supports the following unit-record devices for peripheral input and output:

IBM 1442 Card Read Punch
IBM 2501 Card Reader
| IBM 2520 Card Read Punch
IBM 2540 Card Read Punch
IBM 2560 Multifunction Card Machine
IBM 3501 Card Reader
IBM 3504/3505 Card Reader
IBM 3521 Card Punch
IBM 3525 Card Punch
IBM 5425 Multifunction Card Unit

IBM 1403 Printer
IBM 1443 Printer
IBM 3203 Printer
IBM 3211 Printer
IBM 3784 Line Printer
| IBM 3800 Printer
IBM 5203 Printer

IBM 3540 Diskette Input/Output Unit (input only).

| DIRECT ACCESS STORAGE DEVICES

POWER/VS uses three types of files: the account file, the queue file, and the data file. For these files, POWER/VS requires direct access storage. Different POWER/VS files may reside on different direct access storage devices, which may be any of the following:

IBM 2314 Direct Access Storage Facility
IBM 2319 Disk Storage
| IBM 3330 Disk Storage (including Model 11)
| IBM 3333 Disk Storage and Control
| IBM 3340 Direct Access Storage
IBM 3350 Direct Access Storage.

| REMOTE JOB ENTRY DEVICES

To enter jobs into the system using remote terminals, and to obtain output at such terminals, the POWER/VS remote job entry (RJE) feature supports the following BSC and SDLC terminals:

IBM 2770 Data Communication System
IBM 2780 Data Transmission Terminal
IBM 3741 Data Station (Model 2)
| IBM 3741 Programmable Work Station (Model 4)
| IBM 3770 Data Communication System in BSC or SNA mode
| IBM 3780 Data Communications Terminal
IBM 3790 Communication System.

Notes: In general, all BSC terminals and/or systems are supported that emulate the 2770, 2780, or 3780 line discipline, and all SNA work stations that follow the SNA rules defined for LUTYPE1. For details refer to Appendix B, "RJE,SNA I/O Specifications".

The following applies to 3770 terminals: In BSC mode, 3770s are supported as 2770 or 3780 terminals, in SNA mode, LUTYPE1 protocols must be used.

Up to 25 BSC terminals or up to 200 SDLC and BSC terminals can communicate concurrently with the central system via switched or nonswitched lines. In nonconcurrent operation on switched lines, any number of BSC terminals may be controlled by POWER/VS as long as the maximum number of concurrently operating BSC terminals does not exceed 25.

For further information on POWER/VS RJE, refer to the DOS/VS POWER/VS Work Station User's Guide, GC33-6049.

Concepts and Functions

The basic concept of POWER/VS has not changed since the first appearance of its predecessor, POWER, when non-VS DOS allowed the quasi-simultaneous execution of two or three independent programs in different partitions residing in the main storage of an IBM System/360. The implementation of POWER/VS in DOS/VS maintains this spooling concept mentioned earlier. In detail, POWER/VS operation includes the following major activities:

- Input. POWER/VS reads the job streams (job control statements, programs, and data cards) for the individual partitions and stores these in input queues on disk. The input may be entered from:
 - A local card reader device.
 - A local diskette device.
 - A remote BSC terminal.
 - A remote SNA (SDLC) work station with a console and an input device (card reader or 80-column card-image disk or diskette device).
 - An outside partition not controlled by POWER/VS.
- Execution. From the input queue on disk, the jobs are transferred by POWER/VS to the designated partitions and executed. The jobs contain all required input and may optionally contain JECL statements to indicate individual job execution or I/O requirements.
- Output. Job output (printer and punch) of every job is stored in output queues on disk (or tape) by POWER/VS before it is finally processed as output:
 - For a local printer or punch device.
 - For a remote BSC terminal.
 - For a remote SNA (SDLC) work station where the output can either be produced by an appropriate work station unit-record device or where it can be temporarily stored on disk or diskette. In the latter case, the user can perform remote spooling of his job output.
 - For an outside partition not controlled by POWER/VS.

- Control. Throughout the different steps of input, execution, and output, the jobs running under POWER/VS are within the management of the user through the following command language facilities:
 - Job Entry Control Language (JECL). Along with his input job decks, the user may insert JECL commands to describe individual job execution or I/O requirements.
 - Central operator commands. From the starting of input until the final discharging of list and punch output, the central operator may start and stop the job execution, display or alter execution and I/O performance characteristics, or perform similar functions.
 - Remote terminal operator commands. Similar to the central operator, the remote terminal operator may perform many of the functions as above for all jobs submitted by him or routed to him.
 - Cross-partition communication macros. These macros enable an application program not running under POWER/VS control to supervise jobs running under POWER/VS control in the same way as the central or remote terminal operator. However, this job supervision from outside POWER/VS is restricted to the use of five commands. For details, refer to the description of the CTLSPool macro under "Cross-Partition Communication Macros" in Chapter 3.

Job input as well as job output may be held in the POWER/VS queues for execution or printing/punching at a later time. This allows the user to hold jobs that need, for example, two hours of execution or printing time until the system is less occupied.

- Accounting. POWER/VS optionally creates accounting information for each job running under its control.
- Segmentation. Turnaround time for jobs with extensive printed or punched output can be improved by segmenting the output, which means that parts of the output are printed or punched before the entire job is finished. Segmentation is always done on logical boundaries. For a printer this is a page (skip to channel 1), for a punch this is a card. The following types of output segmentation can be distinguished, based on the event that initiates it:
 - Count-driven output segmentation. You can specify the number of print pages and/or punched cards in the output file, after which physical output should be initiated, by means of the RBS= parameter of the POWER macro during POWER/VS generation.
 - Data-driven output segmentation. You may submit two or more * \$\$ LST (* \$\$ PRT) statements within the boundaries of one POWER/VS job. Each of these statements causes an individual list queue entry to be created.
 - Program-driven output segmentation. In your application program you may use either the SEGMENT macro to separate the output, or the LFCB macro, which causes segmentation before loading the new FCB. You may also issue the SETPRT macro requesting a printer setup that requires operator intervention, changes copy grouping, or specifies a copy number (CINDX) greater than one.

- | • Central Operator Control. The execution of POWER/VS can be controlled by the DOS/VS console operator. Using POWER/VS central operator commands, the operator can perform any of the following functions:
 - Rearrange the order of job streams in input queues or the order in which the output from different jobs is to be printed and/or punched.
 - Delete input job streams or job output waiting in queues.
 - Hold job streams or job output in input or output queues for later consideration or processing.
 - Interrupt printer or punch output, for example, to free a device urgently required by some other task.
 - Restart printer or punch output, either from the beginning or from any desired point within the data stream.
 - Specify the number of copies to be printed or punched from any data file.
 - Retain input job streams in the queues even after processing of the jobs has been completed, or retain printer and/or punch output in the queues even after it has been printed or punched completely.

- | • Teleprocessing Facility. POWER/VS also offers a teleprocessing facility, called Remote Job Entry (RJE). With RJE, jobs may be submitted from remote terminals. Once a job has been entered into the input job queue, the execution proceeds under DOS/VS supervision. All data files required by the job are subject to DOS/VS specifications, just as if the job had been entered locally. RJE job output may be directed to the terminal from which the job was entered, to other terminals, or to the local output unit of the system.

With these functions, POWER/VS ensures a maximum of flexibility for a DOS/VS multiprogramming environment and achieves an optimum usage of the system resources available in any installation.

Figure 1-1 gives an overview of the major functions of POWER/VS.

A Reader/Writer task performs the read/write functions of a POWER/VS job. The operator can call for concurrent execution of as many read/write tasks as he has physical I/O devices available.

An Execution Processor task provides simulated input/output support for the execute function. An execution task is available for each partition under POWER/VS control. An execution output task is automatically initiated when required by a user job.

For more information on POWER/VS tasks, refer to DOS/VS POWER/VS Logic Part 1, SY33-8576.

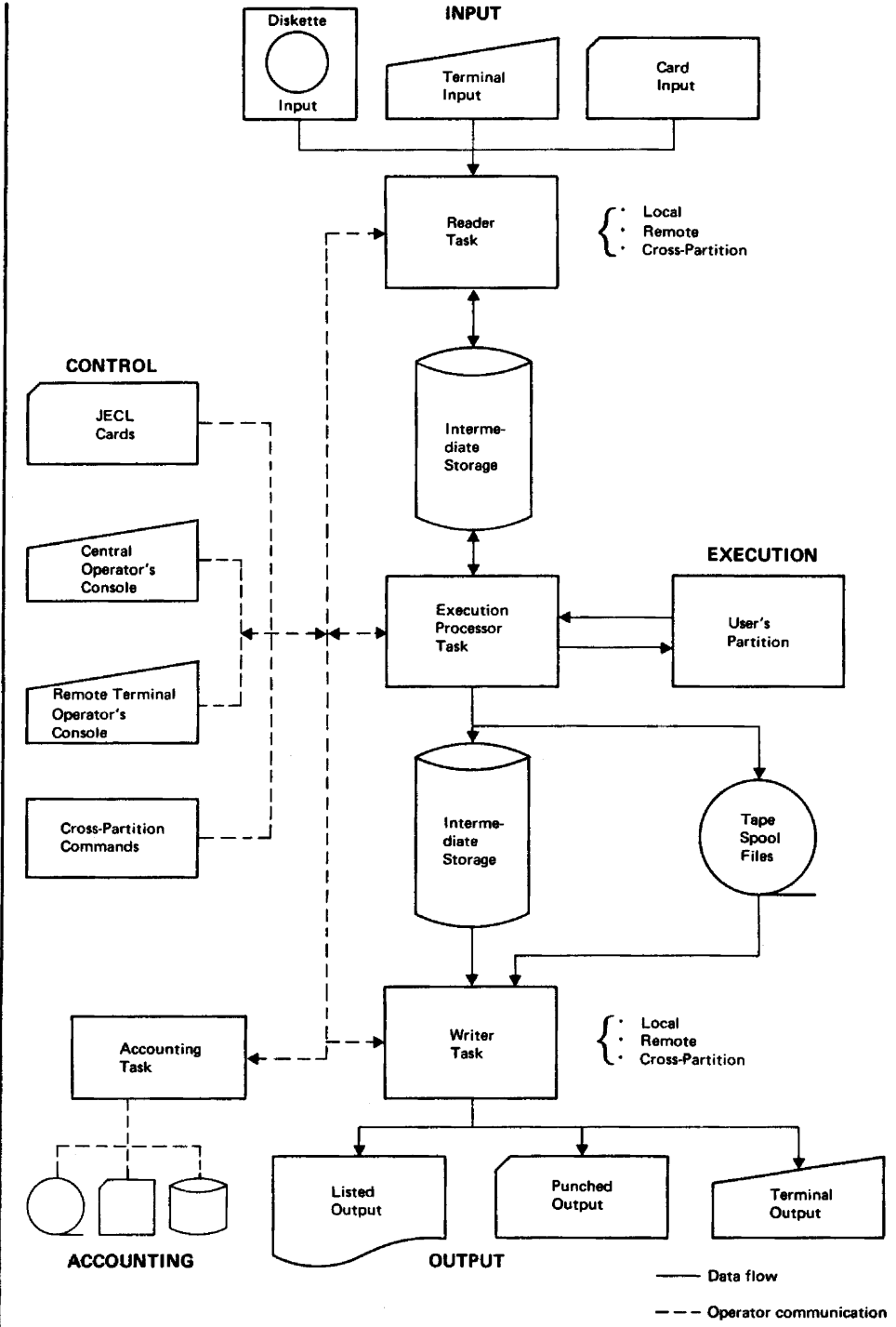


Figure 1-1. The Major Functions of POWER/VS

User Responsibilities

Before you can start to generate POWER/VS, you must tailor your operating system (DOS/VS) to the requirements imposed by POWER/VS. This includes:

- The definition of POWER/VS-specific parameters (one is required) in the macro instructions for the DOS/VS system generation.
- The preparation of POWER/VS generation macros. To tailor POWER/VS to your needs, you can specify optional features of this system, such as the job accounting support, the SNA support, and others in a generation table which you then have to assemble, link-edit, and catalog.
- The allocation of both virtual and real storage space for the POWER/VS partition.
- The assignment of the highest system dispatching priority for the POWER/VS partition over the user partitions assigned to run under POWER/VS.

Details concerning the parameters required for the DOS/VS system generation are contained in the publication DOS/VS System Generation, GC33-5377. For more information about the generation of POWER/VS, refer to Chapter 2 of this manual.

Input/Output Description

POWER/VS intercepts input and output requests issued by application programs that execute in independent DOS/VS partitions. As stated before, these partitions have a lower priority in DOS/VS than the POWER/VS partition that controls them.

Each activity performed by POWER/VS receives input and produces output. That is, to satisfy an input operation requested by an application program, POWER/VS reads the input destined for this application and writes it onto a data file, from where the application program can retrieve it later for processing. Inversely, to satisfy an output request issued by an application program, POWER/VS, after having collected the application program output in a data file, reads this file and prints or punches the data at the appropriate unit-record output device or sends it to the designated remote terminal or work station.

Concurrently with the production of data file entries, POWER/VS creates an entry in a queue file giving summary information on the entries in the data file. There are three types of queue entries corresponding to the three types of data: reader queue, list queue, and punch queue entries. The queue file gives POWER/VS a warm start capability in case of a system breakdown.

In more detail, the input to and the output from POWER/VS consists of one or more of the items listed below.

POWER/VS input can be:

- Control information, which can be either POWER/VS job entry control language (JECL) statements, central operator commands, POWER/VS remote job entry (RJE) commands, or JECL commands contained in cross-partition communication macros.

- DOS/VS job control statements.
- Application-oriented data provided from a card or diskette reader or from a remote terminal or work station.
- DOS/VS source statement library books. These may be included in the input job stream by using a special JECCL command (Source Library Include; see the description of the * \$\$ SLI command in Chapter 3).

In this case, the output consists of data stored in an input data file.

POWER/VS output consists of data stored in an output data file, and of messages. In this case, the output can be:

- Printer (list) output
- Punch output.

This output can be directed to unit-record devices, to magnetic tape devices, to remote terminals or work stations, or to other partitions that do not execute under control of POWER/VS.

Environmental Considerations

This section contains information pertaining to the planning of a POWER/VS installation and discusses installation-related alternatives with respect to the use of POWER/VS.

PROGRAMMING SYSTEMS

The POWER/VS modules are written in IBM System/370 Assembler language. The following programs are used in the support of POWER/VS:

1. Required software support:
 - DOS/VS Release 34 (Program Number 5745-010)
2. Optional software support for RJE,SNA:
 - DOS/VS Virtual Telecommunications Access Method (VTAM) Level 2 (Program Number 5745-010).
 - DOS/VS Network Control Program (NCP) Version 5.0 (Program Number 5747-AS2).
3. Optional software support for the IBM 3800 Printer:
 - DOS/VS POWER/VS IBM 3800 Printing Subsystem Independent Component Release (ICR).

POWER/VS also operates with the above systems in a virtual machine environment under VM/370.

System Requirements

In addition to the system requirements of DOS/VS with the necessary features, of VTAM (optionally), and of user programs, POWER/VS has the following additional hardware system requirements:

1. Minimum storage requirements:

	K bytes (no RJE support)	K bytes (RJE,BSC support)	K bytes (RJE,SNA support)
Main Storage (real)	10	14	14
Main Storage (virtual)*	152 (see note)	174	210 (see note)
*For the 3800 ICR support, add 4K bytes to each value.			

Note: For POWER/VS with RJE,SNA support, or when using an IBM 3800 Printer, you must specify the SIZE= parameter in the // EXEC POWER job control statement.

The minimum requirement may increase due to maintenance activity.

2. Maximum resource requirements:

- Up to 25 BSC terminals concurrently. Through switched operation, more terminals may be in the network nonconcurrently.
- Up to 200 BSC terminals and SNA work stations (SDLC) (concurrently). This figure may include up to 25 BSC terminals. Through switched operation, more BSC terminals may be in the network nonconcurrently as long as 25 or less BSC terminals are operating concurrently with POWER/VS.
- Up to eight logical printers, eight logical punches, and one logical reader may be assigned to all partitions under control of POWER/VS.

To use the RJE,SNA capabilities of POWER/VS for the IBM 3790, the 3790 Communication System with RJE Facility Support is required.

Timing and Throughput Information

Although the execution time for a single DOS/VS job may slightly increase under the control of POWER/VS (which is due to the strict separation of input and output activities from the program execution), the overall system performance increases considerably. By adding further resources in terms of real and virtual storage and peripheral equipment over and above the minimum system requirements for POWER/VS, you can further improve the efficiency of your installed IBM System/370.

Chapter 2. Installing POWER/VS

This chapter guides you in the installation of POWER/VS. Before you start installing the system, you have to consider a number of topics concerning requirements in the available hardware and software. These considerations are discussed in the section "Planning the System". The next section, "Generating POWER/VS", acquaints you with the macros available to generate the system and with the steps to be executed until POWER/VS is available in your DOS/VS environment.

The information in this chapter is supplemented by a summary of the terminals supported with POWER/VS RJE and a discussion of some special unit-record devices.

Planning the System

Before POWER/VS can be installed, you have to ensure the availability of certain features in your computer installation environment. These prerequisites fall into the following categories:

- Machine requirements
- Storage requirements
- DOS/VS supervisor requirements
- VTAM and NCP generation requirements (optionally).

The following sections contain detailed discussions of these topics.

MACHINE REQUIREMENTS

POWER/VS operates on any IBM System/370 CPU that is supported by DOS/VS Release 34. It resides in a virtual partition with a priority higher than that of the partitions it controls. POWER/VS supports programs running in real and virtual mode.

POWER/VS supporting RJE,SNA requires a minimum of 210K virtual and 14K real storage. POWER/VS with RJE,BSC requires a minimum of 174K virtual and 14K real storage. Without RJE support, 152K virtual and 10K real storage are required. Included in these figures is sufficient real and virtual storage for one active POWER/VS task. For performance reasons, however, and for the simultaneous operation of several POWER/VS tasks, you should allocate more real and virtual storage. For details, refer to the example given under "Storage Requirements Calculation" later in this chapter.

With the remote job entry (RJE) feature, POWER/VS accepts input from, and routes output to, remote terminals or work stations. The attachment of BSC terminals to a central system requires one of the following minimum configurations:

- An IBM 2701 Data Adapter Unit (DAU) with Synchronous Data Adapter, Type II, equipped with EBCDIC or ASCII code and transparency, or
- An IBM 2703 Transmission Control Unit (TCU) with Synchronous Attachment, Synchronous Base, Synchronous Terminal Control - EBCDIC or ASCII and Synchronous Line Set, and 3704/3705 Communication Controller in 270x emulation mode, or
- For a Model 115 CPU, an ICA with synchronous line group and synchronous line medium speed (with or without clock).
- For a Model 125 CPU, an ICA either with synchronous line group and synchronous line medium speed (with or without clock), or with synchronous line low load.
- For a Model 135 CPU, and Integrated Communications Adapter (ICA) with a Synchronous Data Adapter, Type II, and transparency.

The attachment of SNA work stations requires:

- An IBM 3704 Communications Controller, or
- An IBM 3705 Communications Controller.

For a detailed list of the terminals supported by POWER/VS, refer to the section "Terminals Supported by POWER/VS" later in this chapter.

INTERMEDIATE STORAGE REQUIREMENTS

Intermediate storage in POWER/VS is on disk (tape for output only) and contains the queue file, the data file, and optionally the account file.

Generally, one IBM 2316 disk pack, 100 cylinders on an IBM 3336 disk pack, or 200 cylinders on an IBM 3348 data module are sufficient for POWER/VS. The POWER/VS files (queue file, data file, and account file) need not be on units of the same type. However, if the data file is on more than one extent or on more than one physical device, these devices must be of the same type. For the 3340, all extents of the data file must reside on the same type of 3348 module. Multiple disk extents and multiple disk volumes are supported to a maximum of five extents for the data file. POWER/VS data files should not reside on a volume containing frequently used files. DASD space assigned must be sufficient to contain both input and one segment of output for any POWER/VS job.

Size of the Data File and Queue File

The data file, which is made up of track groups, and the queue file, which is primarily made up of queue records, are directly related. Each track group has a corresponding queue record. The size of the data file is defined by the total number of track groups, which in turn is limited by the number of records in the queue file.

In estimating the size of the data file and queue file, you should consider the following:

- The maximum number of POWER/VS jobs in the system at any one time
- The largest volume of spooled I/O for any job
- The fact whether or not output segmentation is used.

For the data file extents, estimate separately the total number of input/output card images and the total number of line images spooled to disk in a typical 8-hour shift. Choose a file size large enough to hold half this amount of data. This should prevent POWER/VS from running out of file space. File extents can be respecified if they prove to be too large or too small (check the status report).

The queue file should be large enough to support the entire data file: that is, there should be one queue record for each track group in the data file. It is good practice to allocate six additional queue file records for internal POWER/VS usage.

Block Size of the Data File

The size of the physical records written to the data file is determined by the DBLK parameter. This also influences the size of the data buffers required for each POWER/VS task. If not explicitly specified by the user, the system chooses a default block size, which suits the characteristics of the disk device assigned to the data file. The default values for each device are shown below:

Device Type	Default Data Block Size	Approx. No. of Cards per Block*	Approx. No. of Lines per Block**
2314/2319	920	11	7
3330/3333	952	12	7
3340	808	10	6
3350	960	12	7

* POWER/VS suppresses trailing blanks; hence these figures represent the minimum values.

** Based on 132 print positions per line.

If you specify a value other than the default, it is possible to achieve better performance. In general, the smaller the DBLK value is, the less real storage is required to execute a given number of tasks. Conversely, the larger the DBLK value is, the more real storage is required; however, more efficient use is made of intermediate storage because the larger the block size, the more spool records per track. The more records in a block, the fewer the disk I/O operations to perform. If the data buffer size, which can be increased by 32-byte increments, is larger than 1000 bytes, only one data buffer will fit into a storage page. The largest buffer size is 2008 bytes, which is one data buffer per page with its control information.

Number of Track Groups

After you know your DBLK size, you can determine the track group size. You know the number of blocks per cylinder and approximately how many records are in each block.

If the track group size is small (the smallest is 1), then one queue record is needed for each track on the data file. This results in a larger queue file and an overhead in queue record management, but best utilizes the disk space available in the data file. If the track group size is large (the largest number would be that equal to the number of tracks per cylinder), then fewer queue records (one per cylinder) are needed. However, because there can be only one POWER/VS job for each track group, disk space is wasted on the data file whenever a job does not fill a track group.

If you do not specify a track group size, the system will try to use all of the data file. The system calculates the number of tracks within the extents provided by the data file. It then determines the number of 152-byte records it can write into the queue file. From these two figures it determines the number of track groups to allocate, by calculating the smallest value possible for TRACKGP, which utilizes the largest amount of the data file.

If during POWER/VS initialization, the TRACKGP specified conflicts with the EXTENT information for the data file, the system changes the TRACKGP value. You are informed of the new TRACKGP value by message 1Q09I.

Size of the Account File

To estimate the size of the account file, you should consider that each POWER/VS job can create at least one reader, one list, and one punch account record. In addition, each DOS/VS job step within a POWER/VS job creates one execution account record. For example, for the 3330 approximately 170 jobs can be handled by one cylinder of the account file. This figure is based on an average of five account records per POWER/VS job.

STORAGE REQUIREMENTS

Virtual and real storage space must be allocated for the POWER/VS partition. Figure 2-1 shows the structure of the POWER/VS partition and the types of information stored herein. Figures 2-2 and 2-3 supplement the storage space description given in Figure 2-1. An example how to calculate the size of the POWER/VS partition and of its corresponding storage areas is provided in the section "Storage Requirements Calculation" later in this chapter.

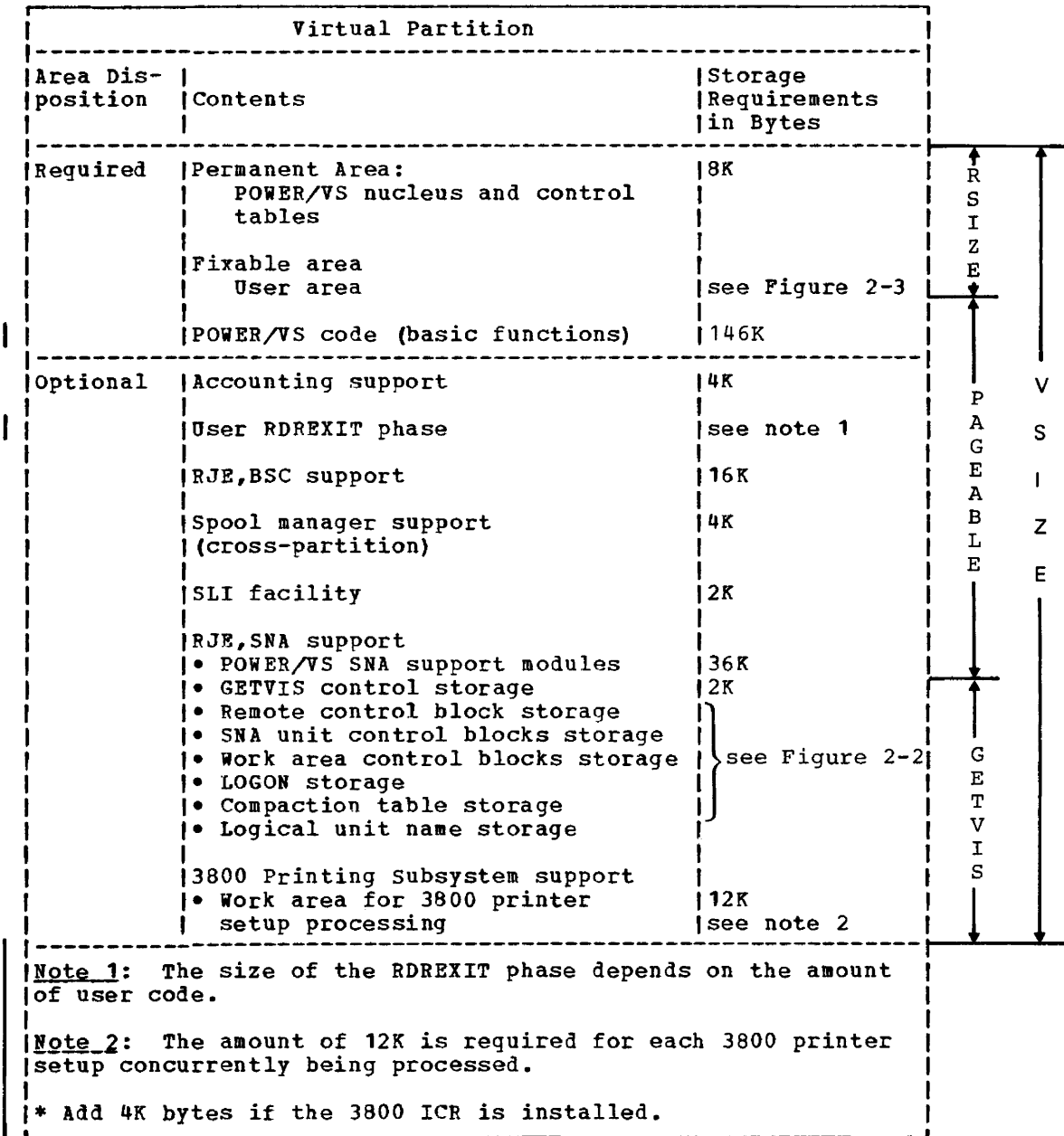


Figure 2-1. Structure of the Virtual Partition and Storage Requirements. The size of the real partition equals the size of the permanent plus the size of the fixable area.

Type of Storage or Control Block	Storage Requirements in Bytes
Remote control block (RMCB) for all work stations	$RMCB + r * 32$ (see note)
SNA unit control blocks (SUCBs and LUCBs) for all work stations	$(SUCB + LUCB * SESSLIM(max)) * WS$ (see note)
Work area control blocks (WACBs)	$(SESSLIM(max) + 1) * WACB * WS$ (see note)
LOGON storage for all work stations	1 page
Compaction table storage for all work stations	$(0.5r + 2) * 2K$ (see note)
Logical unit name storage	1 page
Legend:	
RMCB	= 2K
r	= Number of remote work stations (remote-IDs or number of PRMT macros with TYPE=LUT1).
wscount	= wscount value specified in POWER macro, SNA parameter.
SUCB	= 192
LUCB	= 160
SESSLIM(max)	= Highest number of SESSLIM= specifications in all PRMT macros.
LST	= Number of logical printers. Maximum is 3 per work station.
PUN	= Number of logical punches. Maximum is 1 per work station.
WACB	= 1K
WS	= Number of work stations. Maximum is 200. (wscount or number of PRMT macros).
Note: Rounded to the next higher page boundary.	

Figure 2-2. Space Calculation for RJE,SNA Support Control Blocks

The virtual partition contains:

- The Permanent Area. This area of 8K bytes contains the POWER/VS nucleus and control tables.
- The Fixable Area. This area requires a minimum of 2K space for dynamic control blocks and a variable space for data buffers. The size of the data buffer space depends on user specifications for the execution processors, RJE lines, and reader and writer tasks. Refer to Figure 2-3 for storage sizes and calculations.

- The Pageable Area. This area contains a 142K space for the POWER/VS basic functions, plus space for a number of pageable options.

The pageable area is optional except for the above 142K which, together with the permanent and fixable areas, must be part of the virtual partition.

For POWER/VS with RJE,SNA, the pageable area contains:

- A space of 142K bytes for the basic POWER/VS functions
- The POWER/VS phases for RJE,SNA.

- The GETVIS Area. This area must be provided for RJE,SNA support, or when using the IBM 3800 Printer.

Note: For RJE,SNA support, you must specify SIZE=192K (as a minimum) in the // EXEC job control statement. This value is calculated from the sizes of all required POWER/VS modules (RSIZE + basic function modules + optional function modules).

Note: If the 3800 ICR is installed, add 4K bytes to the base values.

The size of the real partition to be allocated for POWER/VS must correspond to the size of the permanent area plus the size of the fixable area. In other words, the real partition size (RSIZE, see Figure 2-1) equals 8K bytes plus the size of the fixable area.

POWER/VS does not require any storage from a user's application program partition running under POWER/VS control.

POWER/VS Component or Task (Fixable Area)	Storage Requirements in Bytes
Dynamic control blocks (Must always be considered)	2K
SNA control block for RJE,SNA support	192
Each local writer task (one buffer)	$608^1 + 2N + Y$
Each local writer task (two buffers)	$608^1 + 3N + Y$
Each local writer task (four buffers)	$640^1 + 4N + Y$
Each local card reader task (one buffer)	$576^1 + 2N$
Each local card reader task (two buffers)	$576^1 + 3N$
Each BSC writer task	$608^1 + N$
Each SNA writer task	$544^1 + N$
Each BSC reader task	$576^1 + N$
Each SNA reader task	$512^1 + N$

Figure 2-3 (Part 1 of 2). Base Values for the Storage Calculation for the POWER/VS Fixable Area

POWER/VS Component or Task (Fixable Area)	Storage Requirements in Bytes
Each card reader task (3540 connected)	768 ¹ + 2N + E
Each 3540 reader task	704 ¹ + 2N + E
Each execution reader (one per partition)	448 ¹ + N + X
Each execution writer (max. 16 per partition)	448 ¹ + 2N + Z
First RJE,BSC line	576 ¹ + P
Each additional RJE,BSC line	288 ¹ + P
¹ If 3800 ICR is installed add 32 bytes.	
<u>Explanation of Variables</u>	
N = 32 * n, where	
$n = \frac{DBLK+39}{32} \text{ (rounded to the next higher integer)}$	
DBLK is the value specified in the DBLK parameter of the POWER macro. The minimum is 544 bytes, the maximum 2088 bytes.	
E = 32 * e, where	
$e = \frac{(R+8)*26}{32} \text{ (rounded to the next higher integer)}$	
R is the logical record length for the 3540, as specified	
P = Terminal buffer size. To process the first SIGNON card, P = 544 is used.	
X = 320 if SLI statements are to be processed; otherwise X = 0 (applicable only for execution readers).	
Y = 160 if a 3800 printer is being used; otherwise Y = 0.	
Z = 128 if the device to be spooled is a 3800 printer; otherwise Z = 0.	

Figure 2-3 (Part 2 of 2). Base Values for the Storage Calculation for the POWER/VS Fixable Area

STORAGE REQUIREMENTS CALCULATION

The following example illustrates how to calculate the storage space required for an individual POWER/VS installation. It is based on the configuration assumptions given below:

Assumptions

POWER/VS is to support the following:

1. Local POWER/VS operation with:

- One physical reader
- Two physical printers
- One logical printer per partition
- Three partitions controlled by POWER/VS
- SLI facility
- DBLK=544 bytes
- No SPOOL manager support, no user RDREXIT routine, no accounting support.

2. RJE,BSC with:

- One BSC terminal, for example, an IBM 3780.

3. RJE,SNA with:

- One IBM 3770 work station with:
 - One session (SESSLIM=1)
 - One logical printer
 - One logical punch
 - One logical console
 - One logical reader
 - Flow interruption allowed.
- One IBM 3790 work station with:
 - Two sessions (SESSLIM=2)
 - Two logical printers
 - One logical console
 - One logical reader
 - Flow interruption allowed
 - Logical devices running concurrently.
- One IBM 3790 work station with:
 - Five sessions (SESSLIM=5)
 - Three logical printers
 - One logical console
 - One logical reader
 - LU=(name1,...,name5) specified in PRMT macro
 - Default compaction table specified in PRMT macro
 - Logical devices running concurrently.

Figure 2-4 shows the detailed storage calculation for the above configuration:

Ref.	Description	Real Storage (in bytes)	Pageable Storage (in bytes)
<u>1. Local POWER/VS</u>			
Real storage:			
1	Permanent area	8K	
Fixable area			
2	Dynamic control blocks	2K	
	1 local writer task with one print buffer: 608+2*N, where $N = 32 * \frac{544 + 39}{32} = 608$	1 824	
	1 local reader task with one buffer: 576 + 2 * N	1 792	
	3 execution readers: 3 * (448 + 608 + 320)	4 128	
	3 execution writers: 3 * (448 + 2 * 608)	4 992	
3	Total for local tasks	12 736	
Pageable storage:			
4	POWER/VS basic modules		142K
5	SLI support		2K
<u>2. RJE,BSC Support</u>			
Real storage:			
	First RJE,BSC line: 576 + 544	1 120	
	1 BSC writer task 608 + N	1 216	
	Total for BSC tasks	2 336	
Pageable storage:			
7	RJE,BSC module		16K

Figure 2-4 (Part 1 of 2). Detailed Storage Calculation Example

Ref.	Description	Real Storage (in bytes)	Pageable Storage (in bytes)
	<u>3. RJE, SNA Support</u>		
	Real storage:		
	SNA control block	192	
	3770:		
	SNA writer task (printer, punch, or console)	1 152	
	SNA reader task (console)	1 120	
	3790 (2 sessions):		
	2 SNA writer tasks (printers and/or console)	2 304	
	2 SNA reader tasks (reader and/or console)	2 240	
	3790 (5 sessions):		
	4 SNA writer tasks (printers and console)	4 608	
	1 SNA reader task (reader)	1 120	
8	Total for SNA tasks	12 736	
	Pageable storage:		
9	SNA support		36K
	<u>4. GETVIS Storage</u>		
	GETVIS control		2K
	RMCB: 1712 + 3 * 32		2K
	SUCB, LUCBs: (192 + 160 * 5) * 3		4K
	WACBs:		
	For 3770: 6K		
	For 3790 (2 sessions) 6K		
	For 3790 (5 sessions) 6K		
	18K		18K
	LOGON		2K
	lunames (3790):		2K
	Compaction (1 default): ((0.5 * 3) + 2) * 2		8K
10	Total GETVIS storage		38K

Figure 2-4 (Part 2 of 2). Detailed Storage Calculation Example

Figure 2-5 is a summary of the detailed storage calculation above. It is based on the reference numbers in the leftmost column of Figure 2-4.

Ref.	Description	Real	Pageable	GETVIS
	<u>Local POWER</u>			
1	Permanent	8K		
2	Dynamic control blocks	2K		
3	Local tasks (in bytes)	12 736		
4	Basic modules		142K	
5	SLI support		2K	
	<u>RJE,BSC</u>			
6	BSC tasks (in bytes)	2 336		
7	RJE,BSC module		16K	
	<u>RJE,SNA</u>			
8	SNA tasks (in bytes)	12 736		
9	SNA modules		36K	
	GETVIS storage			38K
	Total	38K (*)	196K	38K
	(*) Rounded to the next higher page boundary.			

Figure 2-5. POWER/VS Storage Calculation Summary

For the described example, you must provide the following storage allocations for the POWER/VS partition:

ALLOC: 38K + 196K + 38K = 272K

ALLOCR: 38K

The value of the SIZE parameter in the // EXEC job control statement is calculated from the ALLOC value minus the size of the GETVIS area; hence, for this example, you would have to specify:

// EXEC POWER,SIZE=234K

in the POWER/VS start-up job stream.

DOS/VS SUPERVISOR GENERATION REQUIREMENTS

To include POWER/VS in DOS/VS, you should consider the following before generating the DOS/VS supervisor:

- POWER=YES must be specified in the SUPVR macro; it forces PHO=YES and NPARTS=3 (if NPARTS was not specified or if NPARTS=1 was specified) in the SUPVR macro, and PFIX=YES and ECPREAL=YES in the FOPT macro.
- If POWER/VS accounting support is desired, JA=YES or JA=(n1,n2,...) must be specified in the FOPT macro.
- If POWER/VS cross-partition communication support is desired, XECB=n or XECB=YES must be specified in the FOPT macro.
- If an IBM 3800 printer is being used, AP=YES must be specified in the SUPVR macro.
- If the POWER/VS RJE,SNA support is desired, AP=YES and TP=VTAM must be specified in the SUPVR macro, and one partition must be available for VTAM (see the NPARTS= parameter in the SUPVR macro).
- Both the ALLOC and the ALLOCR macros must be used to define the POWER/VS partition. Refer to the section "Storage Requirements" to determine the appropriate real and virtual storage sizes.
- For the POWER/VS partition, the number of LUB entries for programmer logical units must be large enough to accommodate all reader and writer tasks that may be running concurrently. The number of LUBs is specified in the BGPGR or FnPGR parameter of the IOTAB macro. The default value of 5 is not enough: SYS000 through SYS006 are used for POWER/VS system files. These system files must be assigned by the user.
- The IBM-supplied distribution medium contains a pre-generated POWER/VS system that provides the generation parameter default functions.

SPECIAL CONSIDERATIONS FOR DOS/VS FUNCTIONS

When using POWER/VS, you should be aware of different handling of some of the DOS/VS functions:

- In user-written channel command words (CCWs), the following flags are not supported:
Chain data (CD) - X'80'
Indirect data addressing (IDA) - X'04'.
- No support is provided for parameter EXCP=REAL.
- Since POWER/VS is designed as an interface between DOS/VS operating system and the user's partitions, some DOS/VS commands are affected in their usefulness by POWER/VS:

ALLOC	}	Before using ALLOC or ALLOCR, the PSTOP command must be issued, followed by UNBATCH.
ALLOCR		

CANCEL Not valid for the POWER/VS partition.

STOP If STOP is used, a subsequent PSTOP or PEND command will not be successful.

PRTY Cannot assign the same or a higher partition priority than the one in the POWER/VS partition for any POWER/VS controlled partition.

VTAM AND NCP GENERATION REQUIREMENTS

The DOS/VS Virtual Telecommunications Access Method (VTAM) and the DOS/VS Network Control Program (NCP) are required in your DOS/VS environment if you plan to use POWER/VS with the optional RJE,SNA support.

For POWER/VS with RJE,SNA, your generated version of VTAM must meet the following requirements:

- Using the APPL definition statement, POWER/VS must be defined to VTAM as an application program. The program name must be the same as specified as the APPLID subparameter in the SNA= parameter of the pertinent POWER macro specification.
- The password specified in the PRTCT= parameter of the APPL macro, if included, must match the password specified in the SNA= parameter of the POWER macro specification.
- The names of the SNA logical units (luname) must be supplied to VTAM through the LU macro instruction. The logical unit names must be the same as provided in the PRMT macro as part of the LU= parameter (if specified) and in the POWER/VS RJE terminal commands whenever the value "luname" is requested. Note: Whenever you are requested to specify the parameter "luname" or the LU macro, you must avoid the use of the parameter values EOJ and ALL.
- Your system programmer may use the facilities provided by the USS tailoring services to support a simplified LOGON procedure. However, the use of USS depends on the type of work station: it is possible for a 3770, but it is not allowed for a 3790. Complete information concerning the VTAM generation requirements is contained in the DOS/VS VTAM System Programmer's Guide, GC27-6957.

When using VTAM, you should consider the following: If the buffer limit defined for a logical unit has been exceeded, the session with that logical unit is terminated immediately without an error message. However, this may only occur in an overloaded system. For details, refer to the DOS/VS VTAM System Programmer's Guide.

If you plan to use NCP/VS support in your VTAM network, four NCP modules must be link-edited into the core image library. You can use the IBM-supplied calling book ISTLINK7 for this purpose. The following job stream is necessary to perform this function:

```
// JOB LINKEDIT NCP MODULES
// OPTION CATAL
// INCLUDE ISTLINK7
// EXEC LNKEDT
/ &
```

This job should be executed after the NCP distribution tape has been cataloged to your system.

Generating POWER/VS

To generate POWER/VS, you must specify parameters of the POWER/VS generation macros and subsequently process these macros. This section of the manual describes

- The POWER/VS generation macros and their parameters
- The POWER/VS phases and their functions
- The steps required to generate the system.

POWER/VS GENERATION MACROS

The following is a description of the three macro instructions that allow you to generate POWER/VS for your particular installation. For detailed information on the POWER/VS RJE support, refer also to the POWER/VS Work Station User's Guide, GC33-6049.

The three POWER/VS generation macros are:

- POWER - for the selection of local support options and for the inclusion of SNA support.
- PLINE - for RJE,BSC support (in addition to the POWER and PRMT macros).
- PRMT - for RJE,BSC or RJE,SNA support (in addition to the POWER and PLINE macros, if applicable).

Conventions

- Uppercase letters, numbers, and punctuation marks must be coded exactly as shown.
- Operands can start in any column after the operation code and one blank space. A card can contain more than one operand.
- Lowercase letters and words represent information that must be supplied by the programmer.
- Options contained in braces { } represent alternatives, one of which must be chosen.
- Brackets [] indicate that the element they enclose is optional. Stacked options contained within brackets

[A]
[B]

represent alternatives, one and only one of which may be chosen. The brackets themselves are for descriptive purposes only, and must not be coded.

- Default values are underlined.
- No comma is used before the first parameter specified.
- Keyword parameters consist of a keyword, followed by one or more values.
- Positional parameters must appear in a specified location, relative to other parameters.

POWER Macro

The POWER macro and its parameters are sufficient to generate a non-RJE system. With the parameter SNA= (see below), this macro also provides support for an RJE,SNA system. For an RJE,BSC system, the PLINE and PRMT macros are required additionally. In an RJE,SNA environment, the PLINE macro does not apply; but the PRMT macro must be used. These macros are described later.

Name	Operation	Operands	Col 72
<u>POWER</u> name	POWER	[[DBLK={ <u>0</u> n}]]	[[*]]
		[[,TRACKGP={ <u>0</u> n}]]	[[*]]
		[[,LTAB={ (10,00,05,10,15,20,25,30,35,40, {d0,d1,d2,d3,d4,d5,d6,d7,d8,d9, {45,50,56} d10,d11,d12) }]]	[[*]]
		[[,PRI={ <u>3</u> n}]]	[[*]]
		[[,SUBLIB={ <u>NO</u> s}]]	[[*]]
		[[,ACCOUNT={ <u>NO</u> YES}]]	[[*]]
		[[,STDLINE={ (0,0) (n,m) }]]	[[*]]
		[[,STDCARD={ (0,0) (n,m) }]]	[[*]]
		[[,JLOG={ <u>YES</u> NO}]]	[[*]]
		[[,JSEP={ (0,0) (n,m) }]]	[[*]]
		[[,RBS={ (0,0) (n,m) }]]	[[*]]
		[[,RDREXIT={ <u>NO</u> name}]]	[[*]]
		[[,PAUSE={ <u>NO</u> YES}]]	[[*]]
		[[,SPOOL={ <u>NO</u> YES}]]	[[*]]

(continued on next page)

continued:

Name	Operation	Operands	Col 72
		[[,SNA= ([{ wscount { no. PRMT with LUT1 } }], [password] [, { applid }])] { POWER }]	[*]
		[[,FEED= { YES }] { NO }]	[*]
		[[,MULT12= { YES }] { NO }]	[*]
		[[,COPYSEP= { YES }] { NO }]	[*]
		[[,CLRPRM= { YES }] { NO }]	[*]
		[[,MRKPRM= { YES }] { NO }]	[*]

Programming Notes:

name specifies the phase name to be included in the PHASE card generated as the first card in the object deck resulting from this assembly. The phase name consists of 1 to 7 alphameric characters, the first of which must be alphabetic. If no name is specified, the name POWER is assumed as default.

DBLK= specifies the block size of data file records (both disk and tape) and, consequently, the size of each data buffer. Data buffers are used as I/O areas by the execution processors, readers/writers and RJE routines. The value n must be an integer from 544 to 2008; however, if you have IBM 3211 printers installed and wish to use the * \$\$ LST (* \$\$ PRT) statement with the UCS parameter option, the value n must be an integer from 608 to 2008. If DBLK is omitted or set to 0 and a cold start is performed, the system assumes a block size appropriate to the device type used for the data file, which is shown below:

Device Type	Data Block Size
2314	920
3330	952
3330 Model 11	952
3340	808
3350	960

Note: When you are generating two or more POWER/VS systems and plan to transport disk or tape intermediate storage between the systems, you must specify identical block sizes in all these systems to avoid block size incompatibility problems.

TRACKGP= specifies the number of tracks in a track group. A track group is the amount of disk space that POWER/VS allocates upon a request for space by any POWER/VS routine. The value n may be any integer from 1 to the number of tracks on a cylinder of the device assigned to the data file. If the TRACKGP parameter is omitted or set to 0 (or if conflicts are detected during initialization) and a cold start of POWER/VS is performed, the system calculates a suitable track group size. When a warm start is performed, the track group size established at the previous cold start is used.

Note: The maximum number of track groups is the number of track groups per cylinder times the number of cylinders. The number of track groups per cylinder is the number of tracks per cylinder divided by the track group number. The remainder is ignored.

LTAB= describes the forms control tape or the forms control buffer of the printers. If LTAB is specified, all entries in the sublist (d0 through d12) must be present. Each entry is specified as two digits. The sum of d0 and d12 is considered to be the page depth (= number of print lines on each page). If channel 12 is not specified (d12=00), the value specified by d0 is taken to be the page depth.

Note: For FCB printers POWER/VS will load the default FCB if none is specified. The LTAB used for the current POWER/VS job will be updated by the FCB load routines, but the value specified here will remain in effect as long as POWER/VS is active.

PRI= specifies the default priority of a job entry when a priority is not specified in the * \$\$ JOB statement. The value n may be any digit from 1 to 9, where 9 is the highest priority. If PRI is not specified, the default value is 3.

SUBLIB= specifies the sublibrary of the source statement library to be searched when the sublibrary is not specified in the * \$\$ SLI statement. The value s may be A through Z, 0 through 9, \$, #, or @. You should note, however, that sublibraries A, C, D, E, F, and Z are already special-purpose libraries in DOS/VS. If omitted or if SUBLIB=NO is specified, SLI statements will be treated as data.

ACCOUNT= specifies whether POWER/VS is to maintain accounting information. If NO is specified, no POWER/VS accounting information is maintained. If YES is to be effective, the DOS/VS supervisor must support the job accounting interface (see the JA parameter in the FOPT macro). All accounting information is stored in the POWER/VS account file IJAFILE. If you want job accounting while POWER/VS is active, you should specify ACCOUNT=YES in order to get a correct SIO count.

STDLINE= n specifies the number of list records that may be printed for a job entry before a warning message is issued to the operator. m specifies the additional number of list records that may be printed before the same message is reissued. Both n and m may be up to 6 digits.

A specification of 0 means that no warning will be given. If one of the entries is omitted, the value provided is used for both entries, as shown in the example below:

STDLINE=(,800) generates STDLINE=(800,800)
STDLINE=(10000) generates STDLINE=(10000,10000)
STDLINE=12000 generates STDLINE=(12000,12000).
If STDLINE is omitted, (0,0) is assumed.

STDCARD= n specifies the number of records that each job entry may punch before a warning message is issued to the operator. m specifies the additional number of records that may be punched before the same message is reissued. Both n and m may be up to 6 digits.

A specification of 0 means that no warning will be given. If one of the entries is omitted, the value provided is used for both entries in the same way as shown for STDLINE. If STDCARD is omitted, (0,0) is assumed.

JLOG= specifies whether POWER/VS is to display each jobname together with the job number whenever a * \$\$ JOB statement is read in a partition under POWER/VS control.

If YES is specified, all such statements will be logged on the system console (SYSLOG). If NO is specified, no such logging will take place. If this parameter is omitted, JLOG=YES is assumed as default.

JSEP= specifies that special pages or cards be included in the output to separate job entries. n specifies the number of separator pages. m specifies the number of separator cards. n and m may be any digit from 0 to 9. The minimum number of separator cards is 3. For print output, one additional page as specified by n will be printed. If JSEP is omitted, (0,0) is assumed.

The JSEP specification can be overwritten by a JECL statement for a particular job.

Note: Stacker selection is ignored if job separation is requested. The default stacker for the given device is used instead.

Separator pages are printed with the line depth in effect during job execution. The line depth is obtained from either the printer forms control buffer (FCB) or from the LTAB values in effect at that time. If the page size differs from the line depth, the printout will not be in the correct format.

RBS= specifies the number of list pages (n = number of skips to channel 1) and the number of punch cards (m = number of cards) that can be processed before the output is segmented. This parameter is used only when spooling to disk intermediate storage. n and m may be from one to six digits. A specification of 0 means that no segmentation will take place. If RBS is omitted, (0,0) is assumed. This parameter is ignored when DISP=T is specified in the * \$\$ LST (* \$\$ PRT) or * \$\$ PUN command.

Note: For segmentation on a 3525, and when using print commands, all segments must be punched in a row, or separator cards must be specified. This avoids loss of the last two print lines.

RDREXIT= specifies whether a user-written POWER/VS reader task is supported. Upon initialization of POWER/VS, the phase with the name specified in the RDREXIT parameter is loaded into the POWER/VS partition. If this parameter is omitted, it is assumed that no user-written reader task exists.

For a detailed description of the reader exit routine refer to Appendix E.

PAUSE= specifies whether a punch writer task is to wait before every punch job entry. If NO is specified, the punch writer task will not wait. This is also the default assumption if this parameter is omitted. If PAUSE=YES is specified, the punch writer task will cause message 1Q40A to be displayed before each job entry. Punched cards can then be taken from the stacker of the punch device. To reactivate the writer task, issue a PGO command.

SPOOL= If YES is specified, the POWER/VS cross-partition communication macros PUTSPOOL, GETSPOOL, and CTLSPPOOL are supported. If NO is specified, no interface is supported. This is the default assumption. If SPOOL=YES, then XECB support must be specified in the supervisor.

SNA= specifies parameter values for the POWER/VS RJE,SNA support. The meaning of the positional subparameters is as follows:

wscount allows you to optionally control the reservation of the GETVIS pool size for SNA work station control blocks (SUCBs, LUCBs). The GETVIS pool size is calculated and the space reserved during POWER/VS initialization.

Any value from 0 to 200 may be specified. When coding this operand, consider the following:

1. A default value of 0 is assumed if this operand is omitted or invalidly specified. In this case, the GETVIS pool size is calculated from the total number of SNA work stations and the largest SESSLIM= value specified in the PRMT macro. The largest SESSLIM= value is used to avoid fragmentation.

2. If a value is specified that is smaller than the total number of work stations defined in the PRMT macro, this wscount value is taken during POWER/VS initialization to calculate the GETVIS pool size for SUCBs and LUCBs. This smaller wscount value can limit the number of work stations and sessions that can log on concurrently because the GETVIS pool size is reduced. The limitation depends on momentary system conditions.
3. If a value greater than or equal to the number of work stations defined in the PRMT macro is specified, the information in the PRMT macro is used to calculate the GETVIS pool size.

password - This specifies the password (1 to 8 characters) to be used by POWER/VS when it issues an OPEN call for a VTAM ACB. This password must be identical to the password specified in the PRTCT operand of the VTAM APPL statement. If no password was specified in the PRTCT operand, the password specified in the SNA parameter is ignored during POWER/VS execution.

applid - This identifies the POWER/VS application name used to establish the VTAM API. This name must be identical to the name specified to VTAM during the VTAM generation. If this subparameter is omitted, the name POWER is assumed as default.

Note: If you want to omit subparameters of the SNA= parameter, you must substitute each omitted subparameter by a comma unless the omission occurs at the end of the parameter list.

Examples:

SNA=(wscount,,applid)
 SNA=(,,applid)
 SNA=(wscount)
 SNA=(wscount,password)

FEED= determines the POWER/VS default to eject the previous and feed a new 3540 diskette at end-of-file of the previous one. This default may be overridden by a JECL statement.

MULT12= provides the option for the repeated posting of the printer channel 12 condition on each page to offer compatibility for DOS/VS POWER users.

The repeated posting of channel 12 (also referred to as multiple channel 12 posting) means that once channel 12 has been reached, each subsequent line being printed on the same page will signal the channel 12 condition to the user program until channel 1 has been reached. If MULT12=NO is specified explicitly or by default, the channel 12 condition is indicated to the user program only once when channel 12 has been reached, but not when subsequent lines are printed on the same page. Multiple channel 12 punches may be specified in a FCB phase, however only the last channel 12 location will be recognized.

Note: If MULT12=YES is specified for an FCB-type printer program output may be different from that output, received when running under DOS/VS, not using POWER/VS.

COPYSEP= specifies whether POWER/VS should produce separator pages/cards between each copy of an output file. If NO is specified, the separator pages/cards are suppressed. If YES is specified, separator pages/cards are included between each copy of an output file. However, copy separation is done only for a nonzero JSEP specification, defined either during POWER/VS generation or in a JECL statement.

If this parameter is omitted, COPYSEP=YES is the default. The COPYSEP specification can be overridden by a JECL statement.

CLRPRT= 1 specifies whether POWER/VS is to clear the page buffer of the IBM 3800 Printer at end of job and is to wait until printing of the job is complete. If NO is specified, POWER/VS begins to fill the page buffer with the next job's output without waiting for completion of the printing of the previous job. If YES is specified, the page buffer of the 3800 is cleared at end-of-job. If this parameter is omitted, CLRPRT=YES is the default.

Note: If you specify CLRPRT=NO, be aware that if an error occurs on the 3800 while printing the previous job, the result may be incomplete. The previous job would already have been dequeued according to its disposition and POWER/VS is not able to reprint any lost data. However, if you define the disposition K in the * \$\$ LST statement, the output remains in the LST queue after processing. After verification that the output is complete, the operator can then delete the output job from the queue.

Specifying CLRPRT=YES may degrade performance of the 3800, because the printer must wait for the page buffer to be emptied.

MRKFRM= 1 specifies whether the Mark Form function of the IBM 3800 Printer is used when producing job separator pages. If NO is specified, POWER/VS includes job separator pages according to the number of separator pages specified during POWER/VS generation or in the JECL statement. If YES is specified, POWER/VS creates only one separator page and then issues the Mark Form channel command, which causes a row of dark marks to be printed on the perforation between pages. The 3800 reproduces either three or five copies of the separator page (depending on the page size) with the black marks printed on the perforations. The printer then produces a sufficient number of blank pages to advance the last row of black marks beyond the fuser station.

The MRKFRM option is activated only when job separator pages are to be printed. If the parameter is omitted, MRKFRM=YES is the default.

POWER/VS uses the Mark Form function of the 3800 only when producing start separator pages. (The separator pages between copies are also considered as start separator pages.) When producing end separator pages, POWER/VS reprints the separator page according to the JSEP specification.

(1) Operand is valid only for the IBM 3800 Printer.

PLINE Macro

The PLINE macro instruction and its parameters define the hardware characteristics of an RJE,BSC line, that is, the integrated communication adapter or the control unit. The PLINE macro(s) must follow the POWER macro and precede the PRMT macro(s).

One PLINE macro must be specified for each line. Up to 25 RJE,BSC lines are supported by POWER/VS.

Name	Operation	Operands	Col
	PLINE	ADDR=X'cuu'	[*]
		[,SWITCH={ <u>NO</u> YES}]	[*]
		[,TIMEOUT={ <u>NO</u> n}]	[*]
		[,PSWRD=password]	[*]
		[,TRNSP={ <u>YES</u> NO}]	[*]
		[,CODE={ <u>EBCDIC</u> ASCII}]	[*]
		[,MODSET={ <u>AA</u> AB BA BB}]	

Programming Notes:

ADDR= specifies the line address (channel and unit number) of the control unit. This address is the same as used to identify the control unit to DOS/VS. This parameter is required. Although hexadecimal notation is not required, note that a decimal address will be converted to hexadecimal (that is, 030 is assembled as X'01E').

SWITCH= specifies whether a switched or leased line connection is being used. A specification of SWITCH=NO indicates leased line operation, which is the default assumption. SWITCH=YES indicates switched line operation.

TIMEOUT= specifies the number of minutes the terminal is allowed to remain idle before SIGNOFF is forced. The value n may be any number from 0 to 255. If NO is specified or this parameter is omitted, the terminal is allowed to remain idle for three minutes. This parameter is useful to save line charges on switched lines.

Note: Forcing SIGNOFF to an inactive terminal is not only desirable because the terminal may be keeping a switched line busy, but also because the CPU is then processing timeouts every three seconds. In such a case, the user may be charged for unused line or CPU time.

PSWRD= specifies the security password to be submitted by the remote operator whenever he wants his terminal to be connected to the central system. The password may be up to 8 characters in length. It may be overridden (until the next PSTOP command) by the central operator via the PSTART command.

TRNSP= specifies the transparency feature on the integrated communication adapter or the transmission control unit. This feature is required for transmission of object decks.

NO may be specified only if no object decks are to be transmitted over the line. All data characters between X'00' and X'40' are then converted to zeros, when transmitting to the terminal.

CODE= specifies the transmission code, which may be either EBCDIC or ASCII.

MODSET= Specify this parameter only when using a 2701 control unit. The first character of the operand specifies Interface A or B, when a dual communication interface is used. The second character specifies code A or B, when the Dual Code feature is installed. Refer to the component description of the IBM 2701 Data Adapter Unit, GA22-6864, for more details.

PRMT Macro

The PRMT macro and its parameters define the hardware characteristics of an RJE terminal, identify the user, and describe where his output is to be routed. At least one PRMT macro must be supplied for each user, and all PRMT macros must be supplied after the PLINE macro(s), if present. The PRMT macros must be specified in ascending sequence according to their remote-IDs (see the REMOTE= parameter below), and the definitions for non-SNA remote-IDs must precede the definitions for SNA remote-IDs. If any feature is specified for a terminal on which the feature is not supported, the parameter is ignored.

Name	Operation	Operands	Used in	Col 72
[name]	PRMT	REMOTE=nnn,	SNA,BSC	*
		TYPE= $\left\{ \begin{array}{l} 2770 \\ 2780 \\ 3741 \\ 3780 \\ \text{LUT1} \end{array} \right\}$	SNA,BSC	[*]
		[,LSTROUT={remid}] { nnn }	SNA,BSC	[*]
		[,PUNROUT={remid}] { nnn }	SNA,BSC	[*]
		[,REF=mmm]	SNA,BSC	[*]
		[,CONSOLE={YES}] { NO }	SNA	[*]
		[,PSWRD=password]	SNA	[*]
		[,SESSLIM={n}] { 1 }	SNA	[*]
		[,CMPACT=name]	SNA	[*]
		[,LU= (name ,name ,name ,...)]	SNA	[*]
		[,LIST= $\left\{ \begin{array}{l} 80 \\ 132 \\ 144 \\ 120 \end{array} \right\}$	BSC	[*]
		[,TRNSP={NO}] { YES }	BSC	[*]
		[,BE={YES}] { NO }	BSC	[*]
		[,HFC={YES}] { NO }	BSC	[*]
		[,SCE={YES}] { NO }	BSC	[*]
		[,ABE={YES}] { NO }	BSC	[*]
		[,MRF={YES}] { NO }	BSC	[*]
		[,CS={YES}] { NO }	BSC	[*]
		[,TRACE={YES}] { NO }	BSC	[*]

If the terminal characteristics of a given terminal or work station to be described by a PRMT macro are identical to those specified in a preceding PRMT macro instruction, a short form of this macro may be used (see also the explanation for the REF parameter). The short form of the PRMT macro includes the following parameters:

Name	Operation	Operands	Used in	Col
[name]	PRMT	REMOTE=nnn[name]	SNA, BSC	[*]
		[,LSTROUT={remid}] { nnn }	SNA, BSC	[*]
		[,PUNROUT={remid}] { nnn }	SNA, BSC	[*]
		[,REF=mmm]	SNA, BSC	

Programming Notes:

REMOTE= specifies the remote-ID ("remid") identifying an RJE terminal (BSC terminal or SNA work station) to which the specifications apply. The value nnn is a decimal integer in the range from 1 to 200. The first remote-ID should be specified as 1; any succeeding remote-IDs must be incremented by 1. The lowest SNA remote-ID must be greater than the highest BSC remote-ID. All following SNA remote-IDs must be in consecutive ascending order.

Note: If REMOTE=1 is specified, the parameter REF= cannot be used.

TYPE= specifies the type of terminal, which may be a 2770, 2780, 3741, 3780, or an SNA terminal. For 3770, specify TYPE=2770 or 3780 depending on the buffer size. For SNA support, the terminal must always be specified as LUT1.

LSTROUT= specifies where the list output from jobs submitted by this remote-ID is to be routed by default. If this parameter is omitted, the same remote-ID as specified in the value nnn of the REMOTE parameter is assumed.

PUNROUT= specifies where the punch output from jobs submitted by this remote-ID is to be routed by default. If this parameter is omitted, the same remote-ID as specified in the value nnn of the REMOTE parameter is assumed.

REF= specifies the remote-ID of another PRMT macro, which describes the terminal or work station characteristics of this PRMT macro. The value mmm can be any integer lower than the value nnn used in the REMOTE parameter. 'mmm' can therefore be in the range from 1 to 199.

Parameters other than REMOTE=, LSTROUT=, and PUNROUT are ignored when specified in a PRMT macro with REF=mmm. MNOTE statements in the Assembly listing inform you about this fact.

Note: Check the referenced PRMT macro instruction for correct assembly. No warning is issued in the referring macro if the referenced macro is in error.

CONSOLE= may only be specified for an SNA work station and only if this work station has a line printer in addition to the normal terminal console.

YES causes POWER/VS to interrupt the outbound printer flow whenever messages are generated for the remote operator unless a free session is available with the work station (MLU environment only). For multiple logical units, messages are always sent when a free session is found, regardless whether or not jobs are being transmitted, even if CONSOLE=NO is specified.

For single logical units, NO prevents POWER/VS from interrupting an outbound printer flow for the transmission of messages to the remote operator. In this case, the messages are collected and submitted to the operator after POWER END OF JOB has been reached.

If the SNA configuration does not contain an additional line printer and CONSOLE=YES has been specified, the printout will contain interspersed messages to the remote operator.

If this parameter is omitted, CONSOLE=NO is assumed as default.

Note: For the 3790, CONSOLE=YES should always be specified.

PSWRD= specifies the security password that may have been defined for the remote work station being described. The password must match the password contained in the user data of the LOGON command. If the password is invalid, it is set to blanks, and an MNOTE statement will appear in the Assembly listing.

SESSLIM= specifies the maximum number of sessions that may be logged on by an SNA work station. Any value from 1 to 6 may be specified.

If this parameter is omitted or invalidly specified, the value 1 is assumed as default.

CMPACT= specifies the default name of the compaction table to be used for a work station if no compaction table is called for in the * \$\$ LST or * \$\$ PRT statement. Data compaction can be requested by means of a JECL statement even though no compaction was selected originally, either by default or explicitly. The name of the compaction table may be up to 4 characters long; the first character must be a letter (A through Z) or a \$, a @, or a #.

LU= specifies a list of secondary logical unit names that may log on using the remote-ID specified in the REMOTE= parameter. These names must be specified in accordance with the definitions for VTAM. If this parameter is specified, LOGON requests for a given remote-ID will only be accepted if the name of the logical unit attempting to log on appears in the list of logical unit names for this remote-ID. If the name is not contained in the name list, the LOGON request is rejected.

If this parameter is not specified, no logical unit name test is performed during the LOGON procedure.

Note: The number of names to be specified for one remote-ID may be larger than the number specified in the SESSLIM parameter. The number of logical unit names is only limited by the DOS/VS Assembler restrictions.

- LIST= specifies the number of print positions in a line. Any number can be used from 80 to 150. The maximum value is that of the maximum number of print characters per line of the printer. The following maximum values of LIST apply: 144 for the 2770, 2780, and 3780; 126 for the 3741.
- Note: For the 2770 and 132 print positions, the buffer expansion feature is required and must be specified with the parameter BE=YES.
- TRNSP= specifies the transparency feature on the terminal. This feature is required for the transmission of object decks.
- NO may be specified only if no object decks are to be transmitted to this terminal. All data characters between X'00' and X'40' are then converted to zeros, when transmitting to the terminal.
- BE= specifies buffer expansion, which applies to the 2770 only. The size of the buffer is then 256 bytes.
- BE=YES or ABE=YES are necessary if LIST=132 is specified.
- HFC= specifies horizontal format control on the 2770 and 2780. Horizontal format control is a standard feature of the 3780; a specification of HFC, irrespective of the operand (NO or YES), is ignored.
- Note: If HFC and SCE are both specified, HFC is ignored.
- SCE= specifies space compression/expansion, which applies to the 2770 and 3780 only.
- ABE= specifies additional buffer expansion for the 2770. The size of the buffer is then 512 bytes.
- MRP= specifies the multiple record feature for the 2780 only.
- CS= specifies the component selection feature on the 3780 terminal. Required if a 3781 punch is connected to the 3780.
- TRACE= specifies that all I/O interrupts and timeout records for RJE,BSC lines are to be traced for this terminal as soon as this remote-ID signs on. For more information on this I/O trace, refer to the DOS/VS POWER/VS Logic manuals, SY33-8576 and SY33-8577.

Following are two examples of macro definition sequences for the generation of POWER/VS systems.

Example 1. POWER with RJE,BSC and RJE,SNA support:

```

NAME1  POWER SNA=(3)                UP TO 4 WORK STATIONS MAY BE LOGGED
*                                             ON CONCURRENTLY
      PLINE ADDR=x'cuu'             DEFINES BSC LINE
      PLINE ADDR=x'cuu'             DEFINES BSC LINE
      PLINE ADDR=x'cuu'             DEFINES BSC LINE
      PLINE ADDR=x'cuu'             DEFINES BSC LINE
      PRMT  REMOTE=1,                DEFINES BSC REMOTE USER
            TYPE=2780
      PRMT  REMOTE=2,                DEFINES BSC REMOTE USER
            TYPE=2770
      PRMT  REMOTE=3,                DEFINES BSC REMOTE USER
            TYPE=3780
      PRMT  REMOTE=4,                DEFINES BSC REMOTE USER
            TYPE=2770
      PRMT  REMOTE=5,                DEFINES SNA REMOTE USER
            TYPE=LUT1,
            CONSOLE=YES
      PRMT  REMOTE=6,                DEFINES SNA REMOTE USER
            TYPE=LUT1,
            CONSOLE=YES
      PRMT  REMOTE=7,                DEFINES SNA REMOTE USER
            TYPE=LUT1,
            CONSOLE=YES,
            SESSLIM=4,
            CMPACT=TEXT

```

Example 2. POWER with RJE,SNA support only:

```

NAME2  POWER SNA=(3,,POW)
      PRMT  REMOTE=1,
            TYPE=LUT1,
            CONSOLE=YES,
            SESSLIM=5
      PRMT  REMOTE=2,
            TYPE=LUT1,
            CONSOLE=YES,
            SESSLIM=2,
            CMPACT=TEXT,
            LU=(LU01,LU02)
      PRMT  REMOTE=3,
            TYPE=LUT1

```

PCPTAB Macro

By means of the PCPTAB macro, you define a compaction table set for the POWER/VS SNA support. Any number of compaction tables may be generated for a specific POWER/VS installation; they are identified and referred to by their names.

Compaction tables need not be generated during the POWER/VS system generation. They can be provided at any time, as required. For more information on compaction tables refer to "Data Compaction" in Appendix A.

To code the parameters of the PCPTAB macro correctly, consider the following:

The maximum length of a character string defined as an operand of a System/370 Assembler instruction is 255 bytes. This means that, in the 80-column card image format, the operands of the NOMASTn parameters may have to be continued in subsequent continuation cards pertaining to the same parameter. Any card that is to be continued on a subsequent card must have a nonblank continuation character in column 72. The asterisks shown below (in brackets) in column 72 of the macro format description indicate that new parameters may follow.

Name	Operation	Operands	Col 72
name	PCPTAB	MASTER=(mc1,...,mc16)	[[*]]
		[[,NOMAST1=(nmc(1),...,nmc(n))]	[[*]]
		[[,NOMAST2=(nmc(n+1),...,nmc(p))]	[[*]]
		[[,NOMAST3=(nmc(p+1),...,nmc(q))]	[[*]]

Programming Notes:

name specifies the symbolic name of the compaction table. The name may be from one to four characters long; the first character must be alphabetic (A-Z, \$, @, or #). During POWER/VS generation, this name will become the phase name of the default compaction table. If no name is specified, PCPT is used as the name.

MASTER= specifies the set of master characters. From 3 to 16 master characters may be specified. For details concerning the selection of master characters, refer to Appendix A. The master characters can be specified as character or hexadecimal values in any sequence, separated by commas; for example, MASTER=(A,B,40,D,7F).

NOMASTn= specifies the set of non-master characters. The number of non-master characters to be specified depends on the number of master characters defined in the MASTER= parameter. If m is the number of master characters and nm the required number of non-master characters, then

- If $m = 16$, then $nm = 0$
- If $m < 16$, then $nm = 256 - m(m+1)$.

For details concerning the selection of non-master characters, refer to Appendix A. The non-master characters can be specified as character or hexadecimal values in any sequence, separated by commas; for example, NOMAST1=(X,N,3A,1C,B0,H).

The maximum length of the character strings that may be specified in the NOMASTn= parameters is 255 bytes, including the separating commas and the enclosing parentheses. However, you may continue the non-master character specifications of, for example, NOMAST1= by using the NOMAST2= parameter even though the maximum string length in NOMAST1= has not yet been reached.

A number of rules apply for the character specifications of the MASTER= and NOMASTn= operands. These are:

- A master or non-master character must not be specified more than once.
- The following characters, if specified as master or non-master characters, must be enclosed in single quotes when specified in character notation in the respective operands:
 - Left parenthesis: '(' or hexadecimal 4D
 - Right parenthesis: ')' or hexadecimal 5D
 - Comma: ',' or hexadecimal 6B
 - Blank: ' ' or hexadecimal 40.
- The following characters, if specified in character notation as master or non-master characters, must be specified twice while they count as one:
 - Ampersand: '&&' or hexadecimal 50
 - Quote: '' or hexadecimal 7D.
- For performance reasons, forms-feed and new-line (FF (X'0C') and NL (X'15')) characters should be specified as non-master characters. Select-channel characters (SELn or X'04' and SEL2 through SEL12 or X'82' through X'7C') and the carriage-return (CR or X'0D') character should be specified as required. (Refer to "Outbound Print Data" in Appendix A).

After assembly, the resulting object module must be link-edited and cataloged in a core image library (system or private) with the phase name specified in the name field of the PCPTAB macro.

POWER/VS PHASES

The system core image library of the DOS/VS system contains one version of POWER/VS, which is generated with default values for all macro options and is cataloged under the phase name POWER. The names of the phases that constitute POWER/VS are listed in Figure 2-6.

Phase Name	Function
\$\$BPOWIN	Initialization and termination
\$\$BSGMNT	Output setup for unit record (macro-called)
IPW\$\$AQ	Add to queue
IPW\$\$AS	Asynchronous service function (for 3800 printer only)
IPW\$\$CP	Command processor
IPW\$\$DD	Display POWER/VS files
IPW\$\$DQ	Delete from queue
IPW\$\$ER	3540 physical reader
IPW\$\$FQ	Free queue
IPW\$\$GA	Get account record
IPW\$\$GD	Get data record
IPW\$\$IB	Inbound processor
IPW\$\$IC	Initiate command processor
IPW\$\$I1	Initialization
IPW\$\$I2	Initialization
IPW\$\$LF	Logoff processor
IPW\$\$LH	SNA logon processor 1
IPW\$\$LN	SNA logon processor 2
IPW\$\$LR	Logical reader
IPW\$\$LU	LUB and PUB routines
IPW\$\$LW	Logical writer
IPW\$\$MD	Message definition
IPW\$\$MP	Message processor
IPW\$\$MS	Remote message handler
IPW\$\$NQ	Next from queue
IPW\$\$NU	Nucleus
IPW\$\$OB	Outbound processor
IPW\$\$OC	Outbound compaction
IPW\$\$OE	Open 3540
IPW\$\$OT	Open tape
IPW\$\$PA	Put account record
IPW\$\$PD	Put data record
IPW\$\$PL	Physical list
IPW\$\$PP	Physical punch
IPW\$\$PR	Physical reader
IPW\$\$PS	Print status report
IPW\$\$RQ	Reserve queue
IPW\$\$SA	Save account record
IPW\$\$SC	Scan and check parameter
IPW\$\$SL	Get source library record
IPW\$\$SM	Spool management
IPW\$\$SN	SNA manager
IPW\$\$TM	Remote job entry
IPW\$\$TR	Task termination
IPW\$\$VE	VTAM exits
IPW\$\$XJ	Execution JECL analysis
IPW\$\$XR	Execution reader
IPW\$\$XW	Execution writer
POWER	Minimum POWER/VS system

Figure 2-6. POWER/VS Phase Names and Their Functions

POWER/VS GENERATION PROCEDURE

This section describes four steps for generating POWER/VS:

1. Select the various macro options as required. For this purpose, use a copy of the coding work sheet illustrated in Figure 2-7 to record your choices.
2. Assemble the POWER/VS generation macros, which create the POWER/VS generation table. The resulting object module contains the code necessary to invoke the POWER/VS initialization module.
3. Link-edit the object module to the system core image library.
4. For ease of serviceability, check if the translating system dump transients are still in the core image library.

After generation, POWER/VS can be initialized by the start-up procedures described in Chapter 4 of this manual.

Columns	10	16	72
user-name	POWER	DBLK=	
		TRACKGP=	
		LTAB=	
		PRI=	
		SUBLIB=	
		ACCOUNT=	
		STDLINE=	
		STDCARD=	
		JLOG=	
		JSEP=	
		RBS=	
		RDREXIT=	
		PAUSE=	
		SPOOL=	
		SNA=	
		FEED=	
		MULT12=	
		COPYSEP=	

Figure 2-7 (Part 1 of 2). Work Sheet for Coding POWER/VS Generation Macros

Columns	10	16	72
	cont'd	CLRPRT=	
		MRKPRM=	
	PLINE	ADDR=	
		SWITCH=	
		TIMEOUT=	
		PSWRD=	
		TRNSP=	
		CODE=	
		MODSET=	
user-name	PRMT	REMOTE=	
		TYPE=	
		LSTROUT=	
		PUNROUT=	
		REF=	
		CONSOLE=	
		PSWRD=	
		SESSLIM=	
		CMPACT=	
		LU=	
		LIST=	
		TRNSP=	
		BE=	
		HFC=	
		SCE=	
		ABE=	
		MRF=	
		CS=	
		TRACE=	
	END		

Figure 2-7 (Part 2 of 2). Work Sheet for Coding POWER/VS Generation Macros

Terminals Supported by POWER/VS

For RJE,BSC operation, the POWER/VS RJE program can support up to 25 remote terminals of the types IBM 2770, 2780, 3741, 3780 or 3771, 3773, 3774, and 3775 (as a 2770) or 3776 (as a 3780); however, only four terminals can be attached to a System/370 Model 115 and six terminals to a Model 125.

For RJE,SNA operation, POWER/VS supports up to 200 work stations (single or multiple logical units) of the types IBM 3770 and 3790.

IBM 2770 DATA COMMUNICATION SYSTEM

This system may be used as an RJE terminal. It can be connected to the central system by means of common carrier leased or switched facilities. The devices that are attachable to the IBM 2770 are described in System Components: IBM 2770 Data Communication System, GA27-3013.

Supported I/O Units

- 545 Output Punch, Model 3 or 4 (output switch 2)
- 2203, Model A1 or A2, or 2213 Printer, Model 1 or 2 (output switch 1)
- 2502 Card Reader, Model A1 or A2 (input switch 2).

Notes:

1. A card reader and a printer are required, a card punch is optional.
2. The standard keyboard provided with the 2770 may be used as a 2502 reader for text that is compatible with card input. Such input is limited to entry of commands and short job streams (the complete stream must fit entirely into the 2770 buffer). Lowercase characters are not translated to uppercase; therefore, uppercase characters must be entered specifically if so required.

Required Features

- EBCDIC or ASCII transmission code
- Extended retry transmission
- WACK response (wait before transmit positive acknowledgment)
- Line termination.

Supported Features

- Buffer expansion (256 characters)
- Additional buffer expansion (512 bytes)
- Space compression/expansion
- Horizontal format control
- Automatic answering
- 144-character print line
- EBCDIC transparency (required for object decks only).

Only punched output to a remote terminal may be in transparent mode. If punched output is not a requirement, or if the user transmits in non-transparent mode only, the EBCDIC transparency feature on the 2770 and transparency on the 2701 Data Adapter Unit or Integrated Communications Adapter may be omitted.

IBM 2780 DATA TRANSMISSION TERMINAL (MODEL 1 OR 2)

This terminal can be connected to the central system by common carrier leased or switched facilities or privately owned communication facilities. The devices that are attachable to the IBM 2780 are described in Component Description: IBM 2780 Data Transmission Terminal, GA27-3005.

Required Features

- EBCDIC or ASCII transmission code
- Extended retry transmission
- Line termination
- WACK response (wait before transmit positive acknowledgment).

Supported Features

- Multiple record transmission (up to 7 records)
- Horizontal format control
- Automatic answering
- Automatic turnaround (Model 2)
- 120-character print line
- 144-character print line
- Selective character set
- EBCDIC transparency (required for object decks only).

Note: Only punched output to a remote terminal may be in transparent mode. If punched output is not a requirement, or if the user transmits his punched output in non-transparent mode only, the EBCDIC transparency feature on the 2780 and transparency on the 2701 Data Adapter Unit, 2703 Transmission Control Unit, or Integrated Communications Adapter may be omitted.

IBM 3741 DATA STATION (MODEL 2 OR 4)

This data station is supported as an RJE terminal compatible with the IBM 2780 without the multiple record transmission feature. It can be connected to the central system by means of common carrier leased or switched facilities. Devices attachable to the 3741 are described in IBM 3741 Data Station Reference Manual, GA21-9183.

There are no required features on the 3741.

IBM 3771 COMMUNICATION TERMINAL

The IBM 3771 Communication Terminal is a multipurpose keyboard/printer terminal to which the following I/O devices may be attached:

- 3501 Card Reader.
- 3521 Card Punch, with or without Read Feature, attached via an IBM 3782 Card Attachment Unit Model 1.

For details concerning the components of the 3771, refer to the publication IBM 3770 Data Communication System, System Components, GA27-3097.

POWER/VS supports the 3771 in RJE,BSC mode (as a 2770) and in RJE,SNA mode.

IBM 3773 COMMUNICATION TERMINAL

The IBM 3773 Communication Terminal is a multipurpose keyboard/printer terminal with a Magnetic Diskette Storage device with a capacity of 242 688 bytes of storage. For details concerning the components of the 3773, refer to the publication IBM 3770 Data Communication System, System Components, GA27-3097.

POWER/VS supports the 3773 in RJE,BSC mode (as a 2770) and in RJE,SNA mode.

IBM 3774 COMMUNICATION TERMINAL

The IBM 3774 Communication Terminal is a multipurpose keyboard/printer terminal to which the following I/O devices may be attached:

- One or two IBM Magnetic Diskette Storage devices (248 688 bytes of storage per device).
- One card reader: either a 2502 Card Reader, attached via a 3782 Card Attachment Unit, or a 3501 Card Reader.
- One 3521 Card Punch with or without Read Feature, attached via a 3782 Card Attachment Unit.
- One 3784 Line Printer.

For details concerning the components of the 3774, refer to the publication IBM 3770 Data Communication System, System Components, GA27-3097.

POWER/VS supports the 3774 in RJE,BSC mode (as a 2770) and in RJE,SNA mode.

IBM 3775 COMMUNICATION TERMINAL

The IBM 3775 Communication Terminal is a multipurpose keyboard/printer terminal to which the following I/O devices may be attached:

- One or two IBM Magnetic Diskette Storage devices (242 688 bytes of storage per device).
- One card reader: either a 2502 Card Reader, attached via a 3782 Card Attachment Unit, or a 3501 Card Reader.
- One 3521 Card Punch with or without Read Feature, attached via a 3782 Card Attachment Unit.

For details concerning the components of the 3775, refer to the publication IBM 3770 Data Communication System, System Components, GA27-3097.

POWER/VS supports the 3775 in RJE,BSC mode (as a 2770) and in RJE,SNA mode.

IBM 3776 COMMUNICATION TERMINAL

The IBM 3776 Communications Terminal is a multipurpose, medium-speed remote job entry terminal. To the basic unit, which is a keyboard/printer console, the following I/O devices may be attached:

- One or two IBM Magnetic Diskette Storage devices (242 688 bytes of storage per device).

- One card reader: either a 2502 Card Reader, attached via a 3782 Card Attachment Unit, or a 3501 Card Reader.
- One 3521 Card Punch, with or without Read Feature, attached via a 3782 Card Attachment Unit.

For details concerning the components of the 3776, refer to the publication IBM 3770 Data Communication System, System Components, GA27-3097.

POWER/VS supports the 3776 in RJE,BSC mode (as a 3780) and in RJE,SNA mode.

IBM 3780 DATA COMMUNICATION TERMINAL

This terminal may also be used as an RJE terminal. It can be connected to the central system by means of common carrier leased or switched facilities. The devices that are attachable to the IBM 3780 are described in Component Information for the IBM 3780 Data Communications Terminal, GA27-3063.

The 3780 terminal consists of:

- A card reader
- A printer
- A terminal console
- A 3781 card punch (optional; must be hardware-attached as device 2 or 3).

Required Features

- EBCDIC or ASCII transmission code
- Extended retry transmission
- WACK response (wait before transmit positive acknowledgment).

Supported Features

- Additional print positions
- Component selection for printer/punch device
- EBCDIC transparency (required for object decks only).

Notes:

1. Only punched output to a remote terminal may be in transparent mode. If punched output is not a requirement, or if the user transmits in nontransparent mode only, the EBCDIC transparency feature on the 3780 and transparency on the 2701 or 2703 or on the Integrated Communications Adapter may be omitted.

2. Online mode settings for RJE terminals attached to a CPU Model 115 or 125: The bit settings of the ICA line mode specification for an RJE terminal that is attached to a Model 115 or 125 CPU are described in IBM System/370 Model 115 Functional Characteristics, GA33-1510, and in IBM System/370 Model 125 Functional Characteristics, GA33-1506, respectively.

IBM 3790 COMMUNICATION SYSTEM

The IBM 3790 Communication System is a programmable, operator-oriented terminal system. POWER/VS supports the 3790 Communication System Version 6 as an SNA work station with the 3790 RJE Work Station Package (WSP) operating on the 3791 Controller. The 3790 may be attached to the host system either by the local channel attachment or by an SDLC telecommunications line controlled by a 3704 or 3705 Communications Controller. Up to three logical printers, one logical reader and one logical console are supported. For details, refer to the POS/VS POWER/VS Work Station User's Guide, GC33-6049.

| POWER/VS Support for Other Devices

IBM 2560 AND 5425

POWER/VS supports the IBM 2560 and the IBM 5425 as input or output devices. POWER/VS will handle 96-column cards on input and output for the 5425. If SYSPCH is assigned to the 2560 or the 5425, POWER/VS supports:

- Program-controlled stacker selection
- Punch and interpret
- Card print
- Punch/print.

The 2560 and 5425 are subject to the following restrictions:

- POWER/VS cannot update or interpret a card input file.
- POWER/VS ignores stacker selection for input cards.
- Read column binary is not supported.
- In a given partition, POWER/VS will only spool both punch and read functions if a unique physical device address is specified for each function (user dummy for at least one function), for instance, X'00C' for the read function and X'01C' for the punch function.
- POWER/VS uses hopper 1 for input.
- POWER/VS uses the hopper specified by the user for output, provided separator cards are not specified. If separator cards are specified, hopper 2 is used for output.

IBM 3525

POWER/VS supports the basic read and punch functions of the IBM 3525 card reader. Punch/interpret and punch/print are also supported, as well as the card print feature. If SYSPCH is assigned to the 3525, POWER/VS supports the following functions:

- Program-controlled stacker selection
- Print
- Punch/print (multiline)
- Automatic line positioning
- User-controlled line positioning
- Print overflow.

The assignment of a logical unit to a 3525 card reader or card punch is subject to the same rules that apply to the assignment of a logical unit to a 2560 or a 5425.

IBM 3800

POWER/VS supports the IBM 3800 Printer as an output device. Before installing a 3800, you should be familiar with the contents of the DOS/VS IBM 3800 Printing Subsystem Programmer's Guide, GC26-3900. You can find information in the section "POWER/VS Considerations" in the chapter "Planning for the 3800".

Chapter 3. Using POWER/VS

POWER/VS is a reader/writer spooling processor that provides automatic staging of unit-record input and output, class scheduling, and priority scheduling for all DOS/VS programs executed under its control. The Remote Job Entry (RJE) facility of POWER/VS permits jobs to be submitted from remote terminals to the central system and job output to be routed from the central system to the remote terminal.

This chapter describes how to use POWER/VS. At the central system, you have two sets of commands, or statements, to control the operation of POWER/VS and to submit jobs to the system. These are:

- The POWER/VS central operator (POCL) commands
- The POWER/VS job entry control language (JECL) statements.

The formats and functions of these two sets of commands are explained in detail in this chapter.

| In addition, this chapter also covers four other topics. These are:

- The cross-partition communication macros. These macros enable you to request certain POWER/VS functions from within an application program not running under POWER/VS control.
- Specific macros to provide job accounting under POWER/VS and to perform output segmentation, for example, for long-running jobs.
- The control statements available to start POWER/VS using the AUTOSTART feature.
- | • General rules for the POWER/VS user.

At a remote terminal or work station, you use another set of commands. These are called the POWER/VS Remote Job Entry (RJE) terminal commands. A detailed description of their formats and functions is contained in the POWER/VS Work Station User's Guide, GC33-6049. A summary of the POWER/VS statements and commands is provided with the POWER/VS reference card, DOS/VS POWER/VS Reference Summary, GX33-9004.

POWER/VS Central Operator Commands

The POWER/VS central operator commands allow you to control POWER/VS and POWER/VS RJE operation from the central operator's console. They are entered in the same way as attention routine commands. At the display operator console of a System/370 Model 115 or 125, these commands can be entered directly without pressing the request key.

For almost all POWER/VS command codes, a 1-character short form is available. All the options of a command are valid for both the short form and the extended command code.

The POWER/VS central operator commands may be divided into the following three groups:

- Queue Management Commands. These commands control the various POWER/VS input and output queues.
- Task Management Commands. These commands are used to control the reader/writer tasks and the execution processors.
- Control Commands. These commands allow special actions concerning lines, print forms, accounting, and broadcasting activities.

Figure 3-1 gives an overview of the three groups of POWER/VS central operator commands.

<u>Type of Command</u>	<u>Command Code (Operation)</u>	<u>Short Form</u>	<u>Function</u>
Queue Management Commands	PDISPLAY	D	Display the status of POWER/VS job(s).
	PALTER	A	Alter the processing attributes of a POWER/VS job.
	PDELETE	L	Delete queue entries or messages.
	PRELEASE	R	Release POWER/VS jobs for further processing.
Task Management Commands	PSTART	S	Start a task or place a partition under the control of POWER/VS.
	PSTOP	P	Stop a task or release a partition from POWER/VS control.
	PGO	G	Reactivate a task or a partition.
	PEND		Terminate POWER/VS.
	PCANCEL	C	Cancel a POWER/VS status report.
	PFLUSH	F	Terminate processing of the current POWER/VS job.
	PRESTART	T	Restart a writer task.
Control Commands	PBRDCST	B	Transmit a message.
	PACCOUNT	J	Save accumulated account file records.
	PINQUIRE	I	Display the status of RJE lines.
	PSETUP	U	Print page layout.

Figure 3-1. POWER/VS Central Operator Commands

The general format of the POWER/VS central operator commands is as follows:

Each central operator command has two fields, the operation field and the operand field. The operation field can be specified in either the extended or the abbreviated format. The operand field contains one or more parameters separated by commas, or contains no parameters at all.

If operands are specified, they must be separated from the operation field by at least one blank. An optional parameter of a command is enclosed in brackets. Stacked options represent alternatives, one and only one of which may be chosen. Braces enclose alternatives, one of which must be chosen. The operator commands can be entered in either uppercase or lowercase characters.

Commas have to be set as indicated, trailing commas must not be coded.

COMMAND DESCRIPTIONS

PACCOUNT Command

The PACCOUNT (or J) command empties the POWER/VS account file and saves the processed account file records. POWER/VS informs you with message 1Q31I when the file is nearly full. Message 1Q79I will indicate to you when the account file has been saved. If any POWER/VS task requires an account record and no space is available, you will receive message 1Q32I.

This command can be used at any time.

Operation	Operands
{ PACCOUNT } { J }	[tapeaddr, [density] [, tfilename]] DISK, dfilename DEL

tapeaddr specifies the address of the tape drive on which the file is to be written. The format is the following:

cuu
X'cuu'

where cuu is the channel and unit number.

density specifies the density of the tape on which the file has to be written. The format is the following:

ss
X'ss'

tfilename specifies the file name of the labeled tape file to be created. If no file name is specified, the system treats the file as a standard unlabeled DOS/VS tape file.

Your programmer must ensure that // ASSGN and // TLBL information (for the labeled tape) is provided in the POWER/VS partition before initialization of POWER/VS. The // TLBL card must have the filename "tfilename".

DISK,dfilename specifies that the POWER/VS account file is to be written to disk. Your programmer must ensure that // ASSGN, // DLBL, and // EXTENT information (for a disk sequential output file) is provided in the POWER/VS partition before initialization of POWER/VS.

The device type on which the saved account file is to reside must be the same as that for the IJAFILE.

The // DLBL card must have the filename "dfilename".

DEL specifies that the account file records are to be deleted.

Examples:

PACCOUNT 180 Write the account file to an unlabeled tape mounted on drive 180.

PACCOUNT 180,ACCTFLE Write the account file onto tape as a standard label file.

PACCOUNT DEL Delete all account file records.

PACCOUNT Spool account file records to punch queue.

If the command is issued without an operand, the account file is spooled to disk and stored in the punch queue with jobname=PACCOUNT,priority=1, class=P, and disposition=HOLD assigned. To retrieve the account file records, start a punch writer task for output class P.

Each card contains the following information:

<u>Column</u>	<u>Contents</u>
1	Account record identifier (byte 43 of the account record).
2-72	Data (bytes 0-70 of the account record) punched in the same positions as it appears in the account record, including the record identifier.
73-78	Record number of the account file.
79-80	Sequence number of continuation cards. One account record may require two or three punched cards.

PALTER_Command

The PALTER (or A) command alters the attributes of a POWER/VS job after it has been logged by POWER/VS. Any number of attributes may be modified with one command.

Changes of the class or priority parameters place the job at the end of the specified class or priority group.

If the command specifies a job that is being executed, POWER/VS issues message 1R88I, indicating that the job attributes cannot be altered. If the command requests a change of job attributes for a job whose output is being processed (the job is already in the list or punch queue), only the number-of-copies attribute can be changed, except if the job is being transmitted to a 3790 workstation (with PDIR=FMH2). All other

parameters are ignored and message 1R88I is issued to indicate that the job attributes cannot be altered. If the output is segmented, however, and the first segment has already been deleted, not even the number-of-copies parameter can be changed.

Note that the job's class, disposition, and priority attributes may be different in the input and output queues, because input class, disposition, and priority are defined in the * \$\$ JOB statement, whereas output class, disposition, and priority are defined in the * \$\$ LST (* \$\$ PRT) and * \$\$ PUN statements.

If the output job is destined for a 3800 printer, you are able to change only the transmission count. The transmission count is displayed in the first line of the status report. The total number of copies still to be produced (including the transmission in progress) is displayed in the second display line of the status report.

If there is copy grouping (for more information refer to the COPYG parameter in the * \$\$ LST statement) and the output is assigned to a 3800, and you reduce the number of transmissions and increase it with a later PALTER command, the previously truncated group values are restored. If the new number of transmissions is greater than the original copy grouping, then values of 1 are added.

Operation	Operands
{PALTER}	queue, {jobname[, jobnumber]}
{A}	{ALL *abc class1}
	[, PRI=priority]
	[, DISP=disposition]
	[, CLASS=class2]
	[, COPY=number-of-copies]
	[, REMOTE=remid]
	[, CMPACT={* name NO}]

queue specifies the queue for which the command is to be executed in the form:

LST for the list queue
PUN for the punch queue
RDR for the reader queue.

jobname specifies the job name by which the POWER/VS job is known to POWER/VS. The job name can be 2 to 8 alphanumeric characters (including "/", ".", and "-") long.

jobnumber specifies the 1- to 5-digit job number assigned to the POWER/VS job by POWER/VS. The PDISPLAY command may be used to determine the correct job number.

ALL requests that specific attributes for all POWER/VS jobs in the specified queue are to be altered.

*abc requests that attributes are to be altered for POWER/VS jobs whose names begin with the specified characters. For abc, any combination of up to seven alphanumeric characters (including "/", ".", and "-") may be specified.

class1 requests that attributes for POWER/VS jobs with the specified class are to be altered. Valid specifications are A - Z and 0 - 6.

PRI= specifies the new priority of the indicated POWER/VS job(s). Priority is specified as a single digit from 0 to 9. Nine is the highest priority.

DISP= specifies the new disposition of the indicated POWER/VS job(s). You can specify:

- H for hold
- K for keep
- L for leave
- D for dispatchable.

CLASS= specifies the new class (class 2) of the indicated POWER/VS job(s). You can specify any alphabetic character from A to Z. Specification of digits 0 to 6 is permitted for partition input classes.

COPY= specifies the number of copies to be produced after the command is issued, including the one in progress. The number-of-copies specification, which may range from 0 to 99, applies only to output, and is ignored if specified for an input entry.

REMOTE= may be specified as any number from 0 to 200. '0' indicates the central location. The REMOTE= specification applies only to output, and is ignored if specified for an input entry.

COMPACT= specifies whether data compaction is to be performed for the transmission of job output to an SNA work station.

- * indicates that the default compaction table (defined in the PRMT macro) is to be used.
- name identifies the 4-character name of the compaction table to be used.
- NO indicates that no compaction is to be performed.

Examples:

PALTER LST,PAYROLL,DISP=H,CLASS=B Change the disposition of the output of job PAYROLL to H and change the class to B.

A PUN,*PAYRL,DISP=D Alter the disposition to D of all POWER/VS jobs in the punch queue whose names begin with PAYRL.

PALTER LST,P,CLASS=Q Change all class P jobs to class Q.

PBRDCST Command

The PBRDCST (or B) command enables you to send (broadcast) messages to remote users. Up to 16 messages are queued and made available to one or all remote users on request.

Messages to all remote users (ALLUSERS) are queued and given a sequence number. This number can be determined using the PDISPLAY command.

Broadcast messages routed to one specific operator are displayed as soon as the terminal is ready to receive the message.

Operation	Operands
{ PBRDCST B }	{ remid }, 'text' { ALLUSERS }

remid specifies the remote-id of the user for whom the broadcast message is intended. Remote users are identified by numbers from 1 to 200. Zero is reserved for the central location.

ALLUSERS specifies that the broadcast message is intended for all remote users.

'text' specifies the text of the broadcast message, entered between single quotation marks. Single quotation marks within the message must be entered as two quotation marks. The maximum length of the message text is as follows:

- For ALLUSERS-type messages: 46 characters.
- For messages from the terminal to the host system: 60 characters.
- For messages from the host system to the terminal: 49 characters.

If the length of a message text exceeds these maximum values, issue two or more PBRDCST commands, because all characters beyond the maximum text length will be truncated.

Examples:

PBRDCST 150, 'RJE WILL SHUT DOWN IN 30 MINUTES'
Broadcast this message to the user with the remote-ID 150.

PBRDCST ALLUSERS, 'RJE STARTS AT 800 HRS A.M.'
Broadcast this message to all users in the system.

PCANCEL Command

The PCANCEL (or C) command terminates printing of output initiated by the PDISPLAY command.

Operation	Operands
{ PCANCEL } { C }	{ [STATUS] }

STATUS specifies that the printing initiated by a PDISPLAY command is to be terminated.

Example:

PCANCEL Terminate printing initiated by a PDISPLAY command.

PDELETE Command

The PDELETE (or L) command removes one or more POWER/VS jobs from the specified queue. Jobs being executed in a partition or being processed by a writer task are not affected by this command.

Operation	Operands
{ PDELETE } { L }	{ queue, jobname[,jobnumber] queue, ALL queue, class queue, *abc MSG[,n] }

queue specifies the queue for which the command is to be executed in the form:

LST for list queue
PUN for punch queue
RDR for reader queue.

jobname specifies the 2- to 8-character job name by which the entry is known to POWER/VS.

jobnumber specifies the 1- to 5-digit job number assigned to the queue entry by POWER/VS. The PDISPLAY command can be used to determine the correct job number.

ALL specifies that all POWER/VS jobs in the specified queue are to be deleted.

class specifies that all POWER/VS jobs of a certain class are to be deleted from the specified queue. Valid specifications are A - Z and 0 - 6.

*abc specifies that all POWER/VS jobs whose names begin with the specified characters are to be deleted from the specified queue. You can specify up to 7 alphameric characters (including "/", ".", and "-") following the asterisk.

MSG,n specifies that the ALLUSERS-type message with number n is to be deleted. The message number can be determined using the PDISPLAY MSG command. If n is omitted, all ALLUSERS-type messages originated by the central operator are deleted.

Examples:

PDELETE RDR,ALL Delete the entire reader queue.

PDELETE LST,B Delete all class B entries in the list queue.

PDELETE MSG Delete all ALLUSERS-type messages.

| PDELETE LST,*ASSM Delete all entries in the list queue starting with the letters ASSM.

PDISPLAY Command

The PDISPLAY (or D) command can be used to display POWER/VS jobs, ALLUSERS-type messages in the POWER/VS queues, or the status of POWER/VS resources. An example of a POWER/VS status report is given in "Initiating POWER/VS Without AUTOSTART" in Chapter 4.

The status reports are produced at the logical device SYSLOG or at a printer (see the operand "listaddr"). Each queue entry is represented by one line containing the following items:

Items	Characters
jobname	8
jobnumber	5
priority	1
disposition	1
class	1
FROM remote-ID	3
TO remote-ID	3
Number-of-records/pages	6
Number-of-copies	2
Forms-number	4

The last four items are displayed for output only. For punch output, the number of records is displayed, whereas for list output the number of pages is displayed.

Each line of the status report starts with the message number 1R46I. If the status report is made up of entries from different queues, the name of the queue is printed before the first entry is displayed.

The disposition of a queue entry may appear as H for hold, K for keep after processing, L for leave, D for delete after processing or * if an execution processor or a writer task is currently processing the entry.

The class can appear as an alphanumeric character from A to Z, or from 0 to 6.

Remote-ID is displayed as a number from 0 to 200. The remote-ID of the central location is 0.

The number-of-copies column for an impact printer (such as the IBM 1403) displays the number of copies still left (including the copy for the job being printed), or the total number of copies to be produced when the output file has not yet started to print. For an IBM 3800 Printer assigned at job execution time, the column displays the transmission count. In order to distinguish between the transmission count value and the number of copies, the transmission count is preceded by an asterisk (*).

If the output file is destined for an IBM 3800 Printer, a second display line is issued in the status report indicating:

- The forms overlay to be flashed
- The "BURST" request
- The total or still-to-do number of copies.

The forms overlay and burst indications are displayed only when they are requested for the output job. For more information on the BURST and FLASH parameters refer to the * \$\$ LST statement later in this chapter.

If CMPACT=name or CMPACT=NO is found in a queue entry, an additional line is issued in the status report indicating:

- The job name (8 characters)
- The job number (5 characters)
- The compaction option (NO or 'name', up to 4 characters).

If a writer task is processing a queue entry when the PDISPLAY command is entered, the number-of-records/pages is the number of records or pages left to be produced for the current copy. The operator can use this information to determine whether to stop a writer task or let it continue until the current queue entry is completed.

When a job is not active because it was flushed (by means of the PFLUSH command) or stopped (using the PSTOP ... RESTART command) before the PDISPLAY command is issued, the number of copies displayed is the original copy count as specified in the * \$\$ LST(* \$\$ PRT) OR * \$\$ PUN statement. If the ICR is installed the number of copies displayed is the current number of copies left to be produced, including the one that was in progress when the job was flushed or stopped. However, the number of pages/cards displayed is the original total number.

When processing of a flushed (or stopped) job is resumed again and a PDISPLAY is issued subsequently, the number-of-copies value displayed is again the number of copies left to be produced, including the one in progress.

Operation	Operands
{ PDISPLAY } { D }	{ queue, jobname[,jobnumber] queue[,ALL] queue, HOLD queue, FREE queue, RJE[,remid] queue, LOCAL queue, *abc queue, class ALL[,listaddr] HOLD FREE RJE[,remid] LOCAL *abc MSG A M Q T }

queue specifies the queue for which the PDISPLAY command is to be executed in the form:

LST for list queue
PUN for punch queue
RDR for reader queue.

jobname specifies the 2- to 8-character job name by which the POWER/VS job is known to POWER/VS.

jobnumber specifies the 1- to 5-digit job number assigned to the POWER/VS job by POWER/VS.

ALL specifies a request for the status of all POWER/VS jobs in the specified queue. If the queue operand is not specified, status information for all jobs in all queues is displayed.

HOLD specifies a request for the status of all POWER/VS jobs in the specified queue that are not available for processing (that is, in hold or leave state). If the queue operand is not specified, status information is displayed for all such jobs in all queues.

FREE specifies a request for the status of all POWER/VS jobs in the specified queue that are available for processing (that is, in the keep or dispatchable state). If the queue operand is not specified, status information is displayed for all such jobs in all queues.

RJE specifies a request for the status of all RJE-type POWER/VS jobs in the specified queue. If the queue operand is not specified, status information is displayed for all such jobs in all queues.

remid Specified with the RJE operand, remid indicates a request for the status of all POWER/VS jobs in the specified queue that were submitted by, or routed to a specific remote user. Valid specifications are 0 for the central operator and 1 to 200 for remote operators.

LOCAL specifies a request for the status of all POWER/VS jobs in the specified queue that were submitted from, or routed to the central location. If the queue operand is not specified, status information is displayed for all such jobs in all queues.

***abc** specifies a request for the status of all POWER/VS jobs whose names begin with the specified characters. For abc, you can specify any combination of up to seven alphameric characters, including "/", ".", and "-". If the queue operand is not specified, status information for all such jobs in all queues is displayed.

class specifies a request for the status of all POWER/VS jobs with the specified class in the specified queue. Class can be specified as a character from A to Z, or from 0 to 6 (input class only).

listaddr specifies the physical printer on which the status report is to be listed. Enter the channel and unit address in the format

cuu or X'cuu'.

MSG specifies a request to display all ALLUSERS-type messages, and to identify the users who submitted them. The number displayed with each message can be used to delete them.

A specifies a request for a list of all active reader and writer tasks, together with the jobs they are currently processing.

M specifies a request to redisplay a list of all system messages for which an operator response is still required.

Q specifies a request for the number of free queue records, and POWER/VS account records (if applicable).

T specifies a request for the current time and date. The current number of virtual storage pages reserved by PFI macros and the current number of POWER/VS tasks is also displayed.

Examples:

PDISPLAY RDR,*ASSM Display the status of all POWER/VS jobs in the reader queue whose names begin with ASSM.

PDISPLAY RDR,ASSEM Display the status of POWER/VS job ASSEM in the reader queue.

PDISPLAY ALL Display the status of all POWER/VS jobs in all queues.

PEND Command

The PEND command terminates POWER/VS operation, releases all POWER/VS-supported partitions and restores the POWER/VS partition to normal DOS/VS operation. All unit-record devices in the POWER/VS partition and in the partitions controlled by POWER/VS are unassigned. Unless KILL is specified, all active POWER/VS tasks are allowed to continue until they have finished processing the current POWER/VS job (that is, for the job which is being executed, or whose output is being printed or punched). If KILL is specified, POWER/VS is terminated immediately.

Operation	Operands
PEND	[uraddr [KILL[,uraddr]]

no operand terminates the POWER/VS system normally without a dump or status report.

KILL specifies immediate termination of the POWER/VS system and causes a dump to be printed if a printer is assigned. POWER/VS-supported partitions are canceled, but reader and writer tasks are terminated as for the PSTOP command.

uraddr specifies the unit-record device address of the printer on which the status report (dump, if KILL is specified) is to be printed. The format can be either cuu or X'cuu'. If uraddr is not specified, no status report (dump) will be provided. The printer remains assigned after POWER/VS has been terminated.

Note: SYSLST must be assigned to uraddr in the POWER/VS partition.

Examples:

PEND KILL,00E Immediate termination with system dump on printer 00E.

PEND Normal termination without status report or dump.

PEND 00E Normal termination with status report. No system dump will be provided.

PFLUSH Command

The PFLUSH (or F) command terminates processing of a POWER/VS job by a POWER/VS task. When a reader task is flushed, further input processing for the job is terminated and a reader queue entry for the job is not built.

When a writer task is flushed, further output processing for the job is terminated and the job's output queue entry is deleted unless HOLD was specified or the number-of-copies value in the * \$\$ LST (* \$\$ PRT) or * \$\$ PUN statement is greater than 1. The specified reader/writer task continues processing for the next available POWER/VS job.

When a writer task for a job with two or more output copies is flushed, its queue entry is also placed in the HOLD state. This output can be started again (from the beginning of the copy whose output was interrupted) by means of a PALTER or a PRELEASE command.

Operation	Operands
{PFLUSH} {F}	{uraddr[,HOLD]} {partition[,HOLD]}

uraddr specifies the unit record device associated with the reader/writer task.

HOLD specifies that the corresponding queue entry is not to be deleted. Once the entry has been placed in the HOLD state, it cannot be deleted when a partition or writer task is flushed. This operand is ignored if it is specified for a reader task.

You may require the HOLD option, for example, to delay processing of a job that needs more resources than are available at the time. Jobs requiring the same resources may thus be grouped to optimize the use of system resources.

partition specifies that the POWER/VS job is to be terminated and the corresponding reader queue entry is to be deleted unless HOLD or KEEP has been specified. In the case of DISP=K, the disposition is changed to L. If the reader queue entry of a POWER/VS job is deleted, POWER/VS builds a list queue entry containing a message to the programmer. This message informs him that the job has been flushed by the PFLUSH command. Valid partition specifications are BG or Fn, where n is the partition number. Note, however, that only partitions under POWER/VS control can be flushed by means of this command.

Note: If the command PSTOP cuu,EOJ was previously issued for a writer task, the command PFLUSH causes this task to be stopped immediately, but for safety reasons, the job's output queue entry is not deleted.

The current DOS/VS job will be canceled without DUMP, regardless of any dump options specified.

Examples:

PFLUSH 00E Delete list queue entry currently being printed and continue with next entry.

PFLUSH BG Delete reader queue entry currently being processed and continue with next entry.

PFLUSH BG,HOLD Terminate processing of reader queue entry currently being processed. Place the entry in the hold state and continue with the next entry.

PGO Command

The PGO (or G) command reactivates a POWER/VS routine that is waiting for an operator response. You would normally use this command after you have responded to an action-type message. You cannot use PGO to restart a partition or task that was stopped by a PSTOP command.

Note: Reader tasks operating on card readers that give a "device end" interrupt when the READY key has been pressed are automatically reactivated when the device end occurs. You do not need the PGO command for such tasks.

Operation	Operands
{PGO} {G}	{uraddr {partition,cuu}}

uraddr specifies the unit record device associated with the writer task (Note: Reader tasks are reactivated automatically. They do not need a PGO command). It has the format:

cuu or X'cuu'

partition,cuu specifies the partition and device whose output is being spooled to tape. Valid partition specifications are BG or Fn, where n is the partition number. Note, however, that only partitions under POWER/VS control can be reactivated by means of this command.

Examples:

PGO 00E Reactivate a list writer task, for example after the required form has been set up.

PGO BG Reactivate BG execution, for example after a new tape has been mounted.

PGO BG,00E Reactivate BG execution after mounting a tape for the output of printer 00E.

PINQUIRE Command

The PINQUIRE (or I) command provides you with status information for a specific BSC line or logical unit or for all BSC lines and active logical units.

Note: When POWER/VS executes on a virtual machine under VM/370 and, for whatever reason, this machine entered CP mode, you must not specify the short form (I) of the PINQUIRE command, because this would result in the attempt of the Control Program (CP) to perform an IPL.

Operation	Operands
{PINQUIRE} {I}	{luname lineaddr ALL}

luname specifies the name of the logical unit for which the status information is desired.

lineaddr specifies the address of the BSC line for which status information is requested, specified in the format cuu or X'cuu'.

ALL specifies that status information for all supported lines is to be displayed.

The status of a line or a logical unit may be one of the following:

- PROCESSING - The remote user has entered a valid SIGNON or LOGON command. For BSC users, the remote-ID is displayed together with the line status. For SNA users, the session is logged on and a processor (for example, Inbound or Outbound) is active on the session.
- NOT LOGGED ON - No SNA user is currently logged on to the specified logical unit ("luname" specification). This status information is not applicable to BSC lines.
- LOGGED ON - The session is logged on, but no processor is active (not applicable to BSC lines).
- LOGGING ON - The session is in process of being logged on (not applicable to BSC lines).
- INACTIVE - The line has been started by the central operator with a PSTART command, but no user is currently signed on (applicable only to BSC lines).
- NOT INITIATED - The line has not been started by the central operator (applicable only to BSC lines).
- NOT SUPPORTED - Support for the line was not provided during POWER/VS generation (applicable only to BSC lines).

Example:

PINQUIRE ALL Display the status of all supported lines.

Line status report produced:

```
030    PROCESSING 22
031    PROCESSING 150
032    PROCESSING 4
033    NOT INITIATED
034    INACTIVE
RT74   LOGGED ON 041
```

PRELEASE Command

The PRELEASE (or R) command takes one or more POWER/VS jobs out of the hold or leave state and makes them available for processing. If ALL or "class" was specified in the PRELEASE command, jobs with disposition L are not released. After processing, a job originally in the leave state has its queue entry returned to disposition L. Output of a POWER/VS job that is directed to a terminal can be released only from the terminal.

Operation	Operands
{PRELEASE}	{queue, jobname[,jobnumber]}
{R}	{queue, ALL}
	{queue, class}
	{queue, *abc}

queue specifies the queue for which the command is to be executed in the form:

```
LST    for list queue
PUN    for punch queue
RDR    for reader queue.
```

jobname specifies the job name, consisting of 2 to 8 alphanumeric characters including "/", ".", and "-", by which the entry is known to POWER/VS.

jobnumber specifies the 1- to 5-digit job number assigned to the entry by POWER/VS. Use the PDISPLAY command to determine the job number.

ALL specifies that all POWER/VS jobs in the queue, except those with disposition L, are to be released.

class specifies that all POWER/VS jobs, except those with disposition L, of a certain class in the specified queue are to be released. Valid specifications are A - Z and 0 - 6.

*abc specifies that POWER/VS jobs in the specified queue whose names begin with the specified characters are to be released. You can specify up to seven alphanumeric characters (including "/", ".", and "-") in place of abc.

Note: The central operator may not release jobs routed to a remote terminal or work station.

Examples:

PRELEASE LST,PAYROLL Release POWER/VS job PAYROLL in the list queue.
PRELEASE RDR,ALL Release all POWER/VS jobs in the reader queue.
PRELEASE LST,*PAY Release all POWER/VS jobs in the list queue whose names begin with PAY.

PRESTART Command

The PRESTART (or T) command restarts printed or punched output for a POWER/VS job from the beginning or from a specified page or card, within the segment. It can only be used while the task is active.

Operation	Operands
{PRESTART}	uraddr,[n][,i]
{T}	

| uraddr specifies the unit record device associated with a list or punch writer task.

n specifies a signed or unsigned value from 0 to 9999. A plus sign indicates page or card forward, a minus sign indicates page or card backward from the point where processing was interrupted. No sign indicates a specified count from the beginning. If you specify too great a backward count, or if the operand is omitted, printing or punching is resumed from the first page or card. If you specify too great a forward count, a message is issued and the command is ignored.

| i specifies the copy group index to be used for the restart. Any value between 0 and 8 is valid. When 0 is specified, a value of 1 is used. When i is omitted, output processing continues with the copy group index currently used. This operand is valid only for an IBM 3800 Printer.

Example:

PRESTART 00E,-10 Reprint the previous ten pages and continue.

| Note: After a "PSTOP uraddr,RESTART" command has been issued, PRESTART cannot be used. A PSTART must be given before to reactivate the task.

PSETUP Command

The PSETUP (or U) command prints one or more pages of list output with all printable characters of a page replaced by the character X. When the PGO command is issued to resume printing, the pages are printed again with their original contents. The command allows you to check and adjust forms alignment. Adjustment is made manually while printing of the setup pages is in progress. The PSETUP command may be used only after message 1Q40A or 1QA5A has been received.

Operation	Operands
{PSETUP}	uraddr[,n]
{U}	

uraddr specifies the device address of the printer on which the pages are to be printed.

n specifies the number of pages to be printed as one or two digits. If n is omitted, only one page is printed.

Note: A page is defined as a skip to channel 1. That means: if a job output does not contain skips to channel 1, POWER/VS will count the entire output as one page. If a PSETUP is issued in this case, the complete job output will be printed with X characters.

Example:

PSETUP 00E,2 Print two setup pages

Page 1

```
XXXX      XXXXXX      XXXXXX
XX XXXXXX  XXXXX  X  XXXXXXXX
XX XXXXXXXX XXXXX  X  XXXXXX
```

Page 2

```
XXXX      XXXXXX      XXXXXX
XX XXXXX  XXXXX  X  XXXXXX
XX XXXX   XXXXX  X  XXXXXX
```

When the PGO command is issued to resume printing, the pages are printed again with their original contents.

Page 1

```
NAME          NUMBER      SALARY
L. KEEFFE     69060    $  1,080.05
N. REYNOLDS   41240    $    798.83
```

Page 2

```
NAME          NUMBER      SALARY
D. HALEY      33110    $    497.11
A. WATT       76779    $    699.42
```

PSTART Command

A PSTART (or S) command starts either a partition or a reader, writer, or RJE task.

Operation	Operands
{PSTART} {S}	{ task,uraddr,[class][,n] partition,[class],[outclass][,MT] task,uraddr,tapeaddr RDR ,craddr,[class],duaddr[,2] RDR,duaddr,[class], 'file-id',[vols],[S][,V] RJE,lineaddr[,password] RJE,SNA }

task specifies the task to be started in the form:

RDR for reader task
LST for list task
PUN for punch task.

uraddr specifies the unit-record device associated with the reader or writer task. You may specify X'cuu' or cuu.

class For a writer task, this operand specifies the output class or classes that the task may process. Up to four classes may be specified by any four alphabetic characters from A to Z. The order in which classes are specified will be the order of processing. The default class is A.

For a reader task, this operand defines the input class that is to be assigned to all jobs without a class specification in their * \$\$ JOB cards (provided no * \$\$ CPL statement is in effect). "class" is specified as an alphameric character from A to Z, or from 0 to 6. The default class is A.

For a partition, this operand is used to define the input class (or classes) of the jobs that may be executed in the partition. Up to four classes may be specified using from one to four alphameric characters: A to Z or 0 to 6. The order in which classes are specified will be the order of execution. If class is not specified, only partition-dependent jobs (that is, POWER/VS jobs with a matching partition-type input class) are selected.

n specifies the number of buffers that the task will use. Valid specifications are 1, 2, or D, where:

1 = 1 buffer
2 = 2 buffers
D = 4 buffers (2 input and 2 output buffers).

The specification of D is valid only for a list writer task and if the ICR is installed. The operand n is ignored for a punch writer task. If the operand is omitted, the task is started with one buffer.

Note: Storage for the buffers is used only as long as the task is active. When the task becomes inactive, the buffers are released and the storage is returned to the POWER/VS storage pool.

partition	specifies the partition to be brought under the control of POWER/VS. If the partition is already controlled by POWER/VS, the PSTART command acts the same as a normal attention routine START command. A message is issued if the partition is still active or not supported, or has a higher priority than the POWER/VS partition. Valid specifications are BG or Fn, where n is the partition number.
outclass	specifies the default class of this partition's output. May be any alphabetic character. The default is A.
MT	specifies that this partition is to support a never ending (or long running) multitask job. When no more cards are available for the partition, only the task which reads cards is placed in the wait state. Other tasks continue and provide output. Segmentation of the output is controlled via the LFCB macro (forms control buffer printer only) or the SEGMENT macro (printer and punch output). The initial job stream for this partition should have a LST and PUN statement for each spooled output device that may be used. A PSTOP command for this partition also cancels the multitasking parameter.
tapeaddr	specifies the address of the tape drive on which a spool tape is mounted. Only list and punch tasks can be started from a tape. "tapeaddr" must be specified as X'cuu'.
craddr	specifies the address of the card reader for a reader task that is to read input from both a card reader and a diskette. You may specify X'cuu' or cuu.
duaddr	specifies the address of the 3540 diskette unit whose input is to be spooled. You may specify X'cuu' or cuu.
2	specifies that a reader task is to be started with two buffers. If this operand is not specified, the task is started with one buffer.
'file-id'	specifies the name of the diskette file to be read; it must be identical with the file name in the HDR1 label on the diskette. One to eight alphameric characters, including blanks, may be entered between the quotes.
vols	specifies the maximum number of diskettes to be read. The reader task terminates either after the specified number of diskettes has been read or at the end of a diskette whose HDR1 label indicates that this is the last diskette of the file. Any decimal number from 1 to 255 may be specified; the default is 1.

S specifies that volume sequence checking is desired. The sequence number of the first volume must be 1 and the numbers of succeeding volumes must be specified in increments of 1.

V specifies that only diskette files that have been verified are to be accepted. If 'V' is specified and the diskette file to be read has not been verified, the file will be rejected. If the parameter is omitted, no checking for whether diskette files have been verified will be carried out.

RJE specifies that the command was given to start an RJE,BSC task.

lineaddr specifies the line address associated with the RJE task. You may specify the line address as X'cuu' or as cuu.

password specifies a password consisting of up to eight characters. If you specify a password, the remote operator can only sign on by specifying the same password in his * .. SIGNON command.

RJE,SNA causes POWER/VS to activate the VTAM interface. Upon completion, a message is sent to the central operator indicating that the SNA work stations may now log on.

Examples:

PSTART RDR,00C,2 Start reader task on X'00C'. Assign class 2 to all jobs without class assignments.

PSTART RJE,030,NEWYORK Start an RJE task on line 030. NEWYORK is the security password.

PSTART LST,00B,AB Start a list task on X'00B' to print output output from POWER/VS jobs with classes A and B.

PSTART PUN,00D,X'285' Start a punch task on X'00D' to punch output that was spooled onto tape mounted on drive X'285'.

PSTART F1,AB Bring F1 under control of POWER/VS. Select (for execution) input classes A and B only.

PSTART RDR,00C,,2 Start a reader task on X'00C' with two buffers. Assign class A to all jobs without class assignments.

PSTART F2,Z,B,MT Bring F2 under control of POWER/VS. Select (for execution) input class Z only. The default output class for this partition is B. This is a multitask partition.

Note: If you want to process a diskette containing more than one SYSIN file, one specific PSTART command is required for each SYSIN file to be processed.

PSTOP Command

The PSTOP (or P) command stops a reader, writer, or RJE task, or releases a partition from POWER/VS control.

Unless modified by the EOJ operand, PSTOP terminates tasks in the following way: when a reader task is stopped, further input processing for the POWER/VS job being read is terminated. No entry for that job is placed in the input queue. When a writer task is stopped, further output processing is terminated, but the terminated output is not deleted from the output queue. When the task is started again, processing of this output starts with the first record.

For reader/writer or RJE,BSC tasks, POWER/VS acknowledges the PSTOP command with a message at both the central location and the terminal.

Operation	Operands
{ PSTOP P }	{ uraddr [,EOJ ,RESTART] partition lineaddr [,EOJ] RJE,SNA [,EOJ] RJE,SNA,luname [,EOJ] }

uraddr specifies the unit-record device address associated with the reader or writer task. You may specify X'cuu' or cuu.

EOJ specifies that the task will not stop until it has completed processing for the current POWER/VS job. This means (1) for a reader task that it will not stop until all input for the POWER/VS job being read has been spooled and (2) for a writer task that it will not stop until all currently processed printed (or punched) output for a particular POWER/VS job is complete. EOJ should always be specified for RJE tasks, unless an emergency occurs.

RESTART specifies that when the task is restarted, processing will begin at the record following the last one processed before the PSTOP command was issued. This applies also when the number-of-copies value was specified greater than 1. RESTART applies only to output processing.

Note: If neither RESTART nor EOJ is specified when an output task is stopped, processing will begin with the first record of the same output when the task is started again by a PSTART command.

The PRESTART command cannot be used after a PSTOP command.

partition specifies the partition to be stopped. When a partition is stopped by means of a PSTOP command, the partition not only is stopped, but is completely released by POWER/VS. This form of the PSTOP command is not effective until all processing for the POWER/VS job currently being executed is completed. Valid specifications are BG or Fn, where n is the partition number. Note, however, that only partitions under POWER/VS control can be stopped by means of this command.

After the partition has been released by POWER/VS, the partition may be started as a normal batch partition using the START command. To restart it as a POWER/VS-controlled partition, use the PSTART command.

lineaddr specifies the line address (X'cuu' or cuu) associated with the RJE task.

RJE,SNA causes POWER/VS to deactivate the VTAM interface immediately. The optional specification EOJ allows an orderly termination; that is, active inbound or outbound processes may continue until the current job entry is completed.

luname specifies, in combination with the operands RJE,SNA (and in addition, optionally, EOJ) that POWER/VS is to deactivate only the session identified by the "luname" operand. For multiple logical unit work stations, individual sessions may be logged off by the central operator without logging off the entire work station.

Examples:

PSTOP 00E,EOJ Stop the list writer task operating on 00E after it has completed printing the list output of the current POWER/VS job.

PSTOP 030 Stop the RJE task operating on line 030 immediately. If this task is reading, the input records already spooled for the current RJE job are lost. If the task is writing, output will start from the first record of that output when the task is restarted.

PSTOP 00D,RESTART Stop the punch task operating on 00D. When the task is started again, processing of the stopped entry will continue at the point of interruption.

PSTOP F3 Release partition F3 from control of POWER/VS after execution of the POWER/VS job currently being processed is complete.

PSTOP RJE,SNA,EOJ Stop all RJE,SNA sessions as soon as processing of all jobs in all sessions has been completed.

PSTOP RJE,SNA,SES1 Terminate the SNA session with the name SES1 immediately.

POWER/VS Job Entry Control Language Statements

Under POWER/VS, a job is not necessarily the same as a job under DOS/VS. To define a job to POWER/VS and have that job logged in the system, the POWER/VS job entry control language (JECL) or the DOS/VS job control language (JCL) may be used. Examples are shown in Figure 3-2.

If JCL is used, a job is logged under the name specified in the // JOB statement, and this statement, together with the /& statement, delimits the job. If JECL is used, the * \$\$ JOB and * \$\$ EOJ statements delimit a job, and the name specified in the * \$\$ JOB statement is used to log the particular job. A job defined to POWER/VS by using JECL may comprise several DOS/VS jobs or only part of such a job.

You should be aware that POWER/VS will generate a /& in the job stream in two instances:

- If POWER/VS reads a // JOB statement without an immediately preceding /& statement.
- If several POWER/VS jobs are defined for one DOS/VS job, where an * \$\$ JOB statement is not immediately preceded by an * \$\$ EOJ statement.

Because JECL statements begin with an asterisk-blank, job control treats them as comments when POWER/VS is not in operation. This also means that for a writer-only system, dual entries would be created for LST, PUN, and PRT statements. Because this is not allowed, // OPTION NOLOG should precede the JECL statements.

POWER/VS assigns a number to each job name that it logs in order to distinguish between jobs for which identical names were specified. You must use that name and, possibly, the number assigned to it whenever you issue an operator command to control the processing of that job entry. Once a job has been defined to, and logged in, the system by POWER/VS, that job is referred to as a POWER/VS job. The first LST (and the first PUN) output from a POWER/VS job will have the same number as the input job. However, any subsequent LST or PUN entries will have unique job numbers in order to facilitate queue manipulation. All segments of count-driven segmentation (RBS=xxx) will have the same number as the first segment. For accounting purposes, all output contains the job number of the reader queue entry when a record is placed in the account file.

DOS/VS Job Stream	Comments
// JOB ONE * // EXEC JOBSTPA * /&	DOS/VS Job, with no JCL changes.
// JOB TWO * // EXEC JOBSTEPB * *\$\$ PUN CLASS=X // EXEC JOBSTEP C * /&	No *\$\$ JOB/EOJ required for LST or PUN statements.
*\$\$ JOB THIRD // JOB THREE * // EXEC JOBSTEPD * /&	Optional POWER/VS JECL, Required if SYSIPT statements include a // JOB card.
*\$\$ EOJ *\$\$ JOB FORTH // JOB FOUR * // EXEC JOBSTEPE * // EXEC JOBSTEPF * /&	No *\$\$ EOJ required, if POWER/VS job is followed by *\$\$ JOB statement.
*\$\$ CTL CLASS=B *\$\$ JOB FIFTH // JOB FIVE * // EXEC JOBSTEPG * /&	Default CLASS changed to B.
*\$\$ JOB SIXTH // JOB SIX * // EXEC JOBSTEPH * /&	No *\$\$ EOJ required.
// JOB SEVEN * // EXEC JOBSTEP I * /&	Multiple DOS/VS jobs in one POWER/VS job. (*\$\$ JOB and *\$\$ EOJ are both required for this.)
*\$\$ EOJ *\$\$ JOB SEVENTH // JOB EIGHT * // EXEC JOBSTEPJ * *\$\$ EOJ * *\$\$ JOB EIGHTH * // EXEC JOBSTEPK * /&	Multiple POWER/VS jobs for one DOS/VS job.
*\$\$ EOJ *\$\$ CTL CLASS=A // JOB NINE * // EXEC JOBSTEPL * // JOB TEN *\$\$ LST FNO=8X11 * // EXEC JOBSTEPM * *\$\$ LST JSEP=2,RBS=100 // EXEC JOBSTEPN * /&	Default CLASS reset to A. POWER/VS will generate the missing /&. Multiple LST outputs per job (2nd report is segmented.)

Figure 3-2. Examples of the Use of POWER/VS JECL

FUNCTIONS OF JECL STATEMENTS

JECL provides a convenient means for the programmer to specify how POWER/VS is to handle a particular job. He may use JECL statements for the following:

1. Under POWER/VS and POWER/VS RJE to:

- Assign job execution priorities.
- Assign input and output dispositions.
- Hold a job entry in one or all queues.
- Direct unit-record print or punch output to tape rather than to disk.
- Suppress spooling of unit record print or punch output.
- Specify the partition in which the job entry is to be executed.
- Specify the input or output class.
- Request segmentation of lengthy output.
- Insert source statement library data into the input stream.
- Specify 3800 printer character style selection, bursting, forms overlay flashing, and copy modification.

2. Under POWER/VS RJE only to:

- Direct output to another user or messages to all terminal users.
- Direct output to be returned to a terminal or to be processed locally at the central system.
- Specify compaction table names (SNA only).

Normally you will not be concerned with the completion of JECL statements; you may, however, be called upon to change or correct JECL statements that your programmer prepared and inserted in the job stream.

For the JECL statements * \$\$ LST, * \$\$ PRT, and * \$\$ PUN, POWER/VS provides the possibility of error correction at job execution time. If one of these statements was incorrectly specified, the central operator receives a message indicating the error. He may then do one of the following:

- Correct the erroneous statement
- Ignore the error
- Flush the job.

To correct the error, the central operator types in a correct JECL statement, upon which the system processes this statement and continues executing the job.

To ignore the error, the central operator enters End-of-Block or a corresponding command (depending on the type of console) at the system console. The system then selects default values for the JECL statement parameters in error and continues processing.

To flush the job, the central operator enters FLUSH. The system starts processing the next job in the associated queue.

FORMAT OF JECL STATEMENTS

A JECL statement must be completely contained in columns 1-71; column 72 of a single statement must be blank. If the statement is continued, column 72 must contain a non-blank character. However, LST and PUN

statements cataloged in SLI books cannot be continued. Columns 73-80 may be used as a sequence field. This is illustrated in Figure 3-3.

Columns 1-4	5-71	72	73-80		
Identifi- cation Field	Operation Field	Operand Field	Comments Field	Conti- uation Field	Sequence Field
* \$\$					
①	②	③	④	⑤	⑥

Figure 3-3. Format of JECL Statements

Each JECL statement consists of the following fields:

1. Identification field: Contains the characters * \$\$ in columns 1 through 4. Column 2 must be blank. This field can be used for POWER/VS purposes only.
2. Operation field: Specifies the JECL operation. It may either directly follow the second dollar sign or be separated from the second dollar sign by one or more blanks. At least one blank must separate the operation field from the operand field.
3. Operand field: Contains one or more positional or keyword parameters, separated by commas. A blank terminates the operand field; therefore, no parameter may contain embedded blanks. Keyword or positional parameters cannot be mixed within one statement.

When a keyword parameter contains subparameters, the subparameters must be separated by commas and enclosed in parentheses. When only one subparameter is specified, the parentheses can be omitted.

Note: Some POWER/VS features are supported only through keyword notations.

Positional parameters must be specified in a fixed order. If a positional parameter is omitted, its delimiting comma must be coded. If trailing parameters are omitted, trailing commas can also be omitted. If no parameters are specified, the entire operand field can be omitted. When a comment field is present and the entire operand field is blank, the omitted operand field is indicated by a comma, which is preceded and followed by at least one blank. If the operand field exceeds column 71, the field may be continued according to the following rules:

- The operand field can be broken after a comma that separates two parameters, and can be continued on the next card starting from columns 6 through 16. Column 72 of the first card must contain a nonblank character. This process can be repeated for as many cards as are necessary. Continuation cards are not supported for LST or PUN statements within SLI books.
- Columns 1 to 4 of each continuation statement must contain * \$\$ (asterisk-blank-dollar-dollar). Column 72 of a one-card statement or a last continuation card must be blank.

4. Comments field: May contain any information. This field can be broken off anywhere and continued on the next card in any column, subject to the same rules given for the operand field.
5. Continuation field: As stated above, this field (column 72) is used to indicate that the information in the present card is continued in the next card.
6. Sequence field: May contain up to eight characters of optional information used for control statement identification. If present, the sequence field is positionally dependent and must start in column 73.

Note: POWER/VS converts all POWER/VS JECL statements (including continuation cards) and DOS/VS JCL statements (starting with // in columns 1 and 2) to uppercase characters before they are written onto the POWER/VS data files. The sequence field (columns 73 to 80) will not be converted.

The DOS/VS commands /*, /&, permanent ASSGN and comment statements are not converted to uppercase.

USING JECL STATEMENTS

Figure 3-4 lists all JECL statements, together with a brief explanation of their functions, in alphabetic order. This overview is followed by detailed discussions of the function and the parameters of each of these statements.

JECL Statement	Function
* \$\$ CTL	Assigns a new default input class to POWER/VS jobs.
* \$\$ DATA	Inserts data from the reader queue into a book being read from a source statement library.
* \$\$ EOJ	Indicates the end of a POWER/VS job.
* \$\$ JOB	Indicates the beginning of a POWER/VS job and provides handling information.
* \$\$ LST	Provides handling information for printed output.
* \$\$ PRT	
* \$\$ PUN	Provides handling information for punched output.
* \$\$ RDR	Inserts a diskette file into the input stream from a card reader.
* \$\$ SLI	Inserts data from a source statement sublibrary into the job stream.
* \$\$ /*	Indicates the end of a POWER/VS job step (used with the SLI statement only).
* \$\$ /&	Indicates the end of a POWER/VS job (used with the SLI statement only).

Figure 3-4. JECL Statements

The CTL, JOB, and EOJ statements should be presented at DOS/VS job control time. All other JECL statements can be included anywhere in the input stream. More than one LST or PUN statement is allowed per POWER/VS job.

JECL (* \$\$ JOB and * \$\$ EOJ) must always be used in a writer-only partition (a partition that uses POWER/VS only for its output) and is used with the following restrictions:

- RDR, CTL, SLI, and DATA statements are treated as comments.
- LST and PUN statements are recognized at job control time, or when user-generated and written to SYSLOG.
- A number of JOB statement parameters are not meaningful and are ignored.
- Output produced outside the bounds of a POWER/VS-defined job will be ignored.

Note: Data-driven output segmentation is not supported in a writer-only partition.

JECL statements contained in a user data stream are considered to be data if read by a writer-only partition. This allows source statement books containing JECL statements to be cataloged. Specifically, LST, PUN, and DATA statements should be included in source statement books. But a JOB statement will only appear on SYSLOG, and an EOJ statement will act as "no operation". SLI is invalid in a source statement book which is to be cataloged.

Note: For most of the JECL statements, you may specify parameters either in the keyword or in the positional form; but do not use both forms in the same statement. If you use the positional form, you must code a comma whenever you omit an operand, except at the end of a statement. For example, in the statement

```
* $$ PUN H,,03
```

omission of the "forms-number" operand is indicated by the second comma following the character H, while omission of the "norbm1" operand at the end of the statement requires no extra comma.

An optional parameter of a command is shown enclosed in brackets. Braces enclose alternatives, one of which must be chosen. Stacked options represent alternatives, one and only one of which may be chosen.

* \$\$ CTL Statement

The * \$\$ CTL statement allows you to assign a new default input class specification for POWER/VS jobs that are not controlled by JECL, or for which the class parameter was omitted from the * \$\$ JOB statement. The * \$\$ CTL statement may be placed at any POWER/VS job boundary. It overrides all input default class specifications presently in force and remains in effect until the next * \$\$ CTL statement is encountered or the reader task is stopped with a PSTOP command. The statement is ignored in a writer-only system.

Operation	Operands
* \$\$ CTL	CLASS= { A class }

Note: This statement is only available in the keyword format.

CLASS= specifies the input class to which all subsequent jobs are assigned if they do not contain a class specification in their * \$\$ JOB statement. The CLASS= parameter may be used in order to group jobs that require the same I/O configuration, partition, or other resources for execution. If CLASS= is not specified, class A is assumed. The class specification may be specified as any alphabetic character from A to Z or as a single number from 0 through 6. When specified as a number, the input is partition-dependent.

The CLASS= parameter in the * \$\$ CTL statement overrides the class specified in the PSTART command, which overrides default class A.

* \$\$ DATA Statement

The * \$\$ DATA statement allows you to insert data into a book that is retrieved from the source statement library by an * \$\$ SLI statement. Input following the DATA statement must be terminated by a /* or /& statement. DATA statements must be preceded by an * \$\$ SLI statement; otherwise, they are treated as comments. * \$\$ SLI update statements (\$SLI in columns 73-76) and * \$\$ DATA statements may be used together. However, they must appear in the job stream in the same order as the corresponding images appear in the source statement library book. The statement is ignored in a writer-only partition.

Operation	Operands
* \$\$ DATA	name

Note: This statement is only available in the positional format.

name specifies the name of the corresponding * \$\$ DATA statement in the source statement library book where data is to be inserted. The name can be from one to eight alphabetic characters, the first of which must be alphabetic.

* \$\$ EOJ Statement

The * \$\$ EOJ statement marks the end of a POWER/VS job and, if used, is normally placed at the end of a DOS/VS job or job step. This statement is required in a writer-only partition. If this statement is submitted at any other time than DOS/VS job control time, it is still acted upon by POWER/VS.

Operation	Operands
* \$\$ EOJ	

This statement has no operands.

* \$\$ JOB Statement

The * \$\$ JOB statement may be placed anywhere in a DOS/VS job stream to specify POWER/VS job attributes or to begin a POWER/VS job. This statement is required in a writer-only partition.

An * \$\$ JOB statement encountered in the input stream without a preceding * \$\$ EOJ statement is used as a delimiter for the previous POWER/VS job. If the * \$\$ JOB statement is omitted, POWER/VS jobs are delimited by the // JOB and /& job control statements. The POWER/VS job name is then copied from the DOS/VS job name.

Note: In a writer-only partition, an * \$\$ EOJ statement must precede any subsequent * \$\$ JOB statement.

Positional format:

Operation	Operands
* \$\$ JOB	[AUTONAME], [D jobname], [disposition], [priority],[class]

Keyword format:

Operation	Operands
* \$\$ JOB	[JNM={AUTONAME jobname }] [,DISP={D disposition }] [,PRI=priority] [,CLASS=class] [,USER=user information]

Note: The USER operand can only be specified in the keyword format.

JNM= specifies the name by which the job and its queue entries are known to POWER/VS. If the job contains several DOS/VS jobs, they are all logged in POWER/VS by the job name in the * \$\$ JOB statement. The default name AUTONAME is assigned if "jobname" is not specified. The job name may be specified as two to eight alphanumeric characters (0 - 9, A - Z, #, \$, @, /, -, or period (.)). The use of the job names ALL, HOLD, FREE, RJE, LOCAL, or one-character job names should be avoided as they may conflict with some POWER/VS operator commands.

DISP= specifies how POWER/VS will route and schedule the associated entry in the read queue. It may be one of the following: H for hold, K for keep, L for leave, or D for delete after processing. The default is D.

D - Delete after processing. The job is automatically scheduled by POWER/VS according to its class and priority. After job execution, the read queue entry is deleted from the read queue and the data file space is released.

H - Hold job. The job remains in the read queue; it is not dispatched by POWER/VS until the operator changes the disposition to D or K by means of the PALTER command or issues the PRELEASE command.

K - Keep after processing. The job will be automatically scheduled by POWER/VS according to its class and priority. After job execution, the read queue entry is not deleted from the read queue, but the disposition becomes L.

L - Leave in queue. The job remains in the read queue; it is not dispatched by POWER/VS until the operator changes the disposition to K or D by means of the PALTER command or issues the PRELEASE command.

Note: The specification of a disposition is ignored in a writer-only partition.

PRI= specifies the priority assigned to this job in both input and output queues. It is specified as a single digit from 0 to 9. Nine is the highest priority. If the priority parameter is not specified, POWER/VS assigns the default priority defined in the PRI= parameter of the POWER macro. Jobs are dispatched on a first-in-first-out basis within priority and class. The specification of a priority is ignored in a writer-only partition.

CLASS= specifies the class that is assigned to the job entry in the input queue. It may be specified as an alphabetic character from A through Z, or as a numeric character from 0 through 6. This parameter may be used to group jobs that require the same I/O configuration, partition, or other resources for execution. The specification of a class is ignored in a writer-only partition.

Classes 0 through 6 have a specific meaning and are called the partition-dependent classes. Jobs with class 0 can only be selected for execution in the background partition, jobs with class 1 in partition F1, and so on.

If the class parameter is omitted, the class specification is taken from the CTL statement. If no CTL statement is specified, the class is established by the PSTART command. If no class is specified at all, default class A is assigned.

Note: When the * \$\$ JOB statement is coded in its positional form, class 0 may be coded as BG, and classes 1 through 6 may be coded as F1 through F6.

USER= Up to 16 bytes of user information may be specified. If JLOG=YES was specified during POWER/VS generation, this information is printed on SYSLOG when the execution read task encounters an * \$\$ JOB statement. The POWER/VS user information also appears in the POWER/VS account records and in separator pages. The user information must be specified as alphameric unless it is enclosed in single quotes.

* \$\$ LST (* \$\$ PRT) Statement

The * \$\$ LST (or * \$\$ PRT) statement defines the attributes of the intercepted list output of a POWER/VS job. LST and PRT are equivalent operation codes. Any number of * \$\$ LST statements is permitted for each POWER/VS job, except in a reader/writer partition, where only one statement for each device and for each job is allowed. If the * \$\$ LST statement is not included in the job, the default values established during POWER/VS generation are assumed. Additionally, when an IBM 3800 Printer is being used, the default printer setup values established for the 3800 by the SETDF operator command are taken.

Certain parameters (BURST, CHARS, COPYG, DFLT, FLASH, and MODIFY) on the * \$\$ LST statement are only valid for the IBM 3800 Printer. For more information on the 3800 and the use of these parameters see the DOS/VS IBM 3800 Printing Subsystem Programmer's Guide, GC26-3900.

* \$\$ LST statements do not require that * \$\$ JOB and * \$\$ EOJ statements also be used.

The specifications of a particular * \$\$ LST statement remain in effect until a subsequent * \$\$ LST statement for the same printer (LST=) is issued, or until the end of the POWER/VS job.

For each * \$\$ LST statement within a POWER/VS job, a separate list queue entry and job number is created (data-driven output segmentation). The first list entry will contain the same job number as the associated RDR queue entry. However, any subsequent list entries will have a unique job number. Repeated use of the SEGMENT macro in the same job will have the same effect.

Note: Data-driven output segmentation is not supported in a writer-only partition.

// ASSGN statements must have been issued for the printer referred to by the LST or PRT statement.

If an error is detected in the * \$\$ LST (* \$\$ PRT) statement, the central operator may correct the statement during job execution.

Positional format:

Operation	Operands
* \$\$ LST or * \$\$ PRT	[<u>D</u> disposition [<u>A</u> class]], [forms-number], [<u>1</u> number-of-copies], [norbm1], [linetab] [tapeaddr]

Keyword format:

Operation	Operands
* \$\$ LST or * \$\$ PRT	<pre>[DISP={D disposition}] [,CLASS={A class}] [,PRI=priority] [,REMOTE=remid] [,FNO={4 blanks forms-number}] [,JSEP=(sep[,option])] [,COPY={1 number-of-copies}] [,COPYG=(groupvalue[,groupvalue...])] [,TADDR=tapeaddr] [,RBM=(norbm1,norbm2)] [,LTAB=linetab] [,RBS=norbs] [,UCS=(phasename[,option])] [,FCB=phasename] [,LST=lstaddr] [,CMPACT={name} NO }] [,BURST={Y} N }] [,CHARS=(tablename[,tablename...])] [,DFLT={Y} N }] [,FLASH=([overlayname] [,count])] [,MODIFY=(copymodname[,tablename])]</pre>

Note: The operands PRI, REMOTE, JSEP, COPYG, RBS, UCS, FCB, LST, CMPACT, BURST, CHARS, DFLT, FLASH, and MODIFY can only be specified in the keyword form of the statement.

DISP=

specifies the disposition of the printed output after execution of the job. The disposition specification options are listed below:

- D - Delete after processing. The list queue entry is written to a printer according to its class and priority.
- H - Hold. The list queue entry remains in the list queue; it is not written to a printer by POWER/VS until the operator changes the disposition to D or K by means of the PALTER command or issues the PRELEASE command.
- K - Keep after processing. The list queue entry is automatically written to a printer according to its class and priority. When printing is completed, the list queue entry is not deleted from the list queue, and the disposition of the queue entry becomes L.
- L - Leave in queue. The list queue entry remains in the list queue; it is not written to a printer until the operator changes the disposition to D or K by means of the PALTER command or issues the PRELEASE command.
- N - Output without spooling. If N is specified, print requests are not intercepted. Output is written directly, without spooling. In this case, all other parameters (except "LST = ...") are ignored. If the N option is specified, and the device to which SYSLST is assigned is not available at execution time, message OP31D (device not operational) may be issued repeatedly.
- T - Spool to tape intermediate storage. PSTART is used to initiate printing.

If the disposition specification is omitted, D is assumed.

D, H, K, and L are only valid when the output is spooled to disk.

When the output is to be printed at a terminal, N and T are invalid options.

CLASS=

specifies the class of the printed output after execution of the job. It may be specified as any alphabetic character (A through Z). A class may be specified in conjunction with "forms-number" to group together the same types of printed output. "class" is only meaningful when the output is spooled to disk. If a class is specified and the positional form is used, a disposition must also be specified. If "class" is omitted, A is assigned. If the * \$\$ LST statement is omitted, an output class can be specified in the PSTART command.

Note: There is no comma between disposition and class in the positional form. A blank must separate these two parameters.

PRI=

specifies the priority (0-9) of the LST output. Default is the priority of the job.

REMOTE= specifies the TO remote-ID to which list output is to be routed. Each list queue entry also has assigned to it a FROM remote-ID, which identifies the user who initially submitted the job. The "remid" specification can be a number from 0 to 200. Numbers 1 to 200 represent specific remote users. If "remid" is 0, list output is directed to the central installation. If this parameter is omitted or if the positional form of the * \$\$ LST statement (which does not support this parameter) is used, list output is routed to the central location if the job was submitted locally, or is routed according to the LSTROUT specification in the PRMT macro if the job was submitted remotely.

FNO= specifies the type of forms to be used according to installation requirements. The forms number is specified by one to four alphanumeric characters, /, -, or period (.). If "forms-number" is not specified, it is assumed to be four blanks, representing the installation's standard type of forms. When a list writer task is to process output from a list queue entry, it determines whether this forms number matches that of the previous list queue entry. If they match, the list writer task processes the output. If they do not, message 1Q40A is issued, or 1QA5A for the 3800 Printer. If the first queue entry processed after a PSTART has a forms ID of four blanks, no mount message is issued. If the output is directed to a 3741, this parameter must not be specified.

A mount message is issued for FCB printers every time an FCB is loaded. Forms alignment is then possible.

JSEP= specifies the number of job separator pages (0 to 9). If not specified, the value defined during POWER/VS generation in the first entry of the JSEP parameter of the POWER macro is assumed.

Following "sep", an option (either Y or N) can be specified, where:

Y - specifies that the separator pages are also produced between copies of the output file.

N - suppresses the generation of separator pages between each copy of an output file.

When the option is not specified, the COPYSEP specification defined during POWER/VS generation is assumed.

Note: This option is available only with the 3800 ICR and is in effect only when a nonzero specification for JSEP is defined, either during POWER/VS generation or in the * \$\$ LST (or * \$\$ PRT) statement.

Each separator page contains the POWER/VS job name and number, user information from the * \$\$ JOB statement, and the date and time. Separator pages that delimit the last or only segment contain also the word "LAST".

For a nonzero specification, one more page than specified will be printed. This is because printing may have occurred on the paper perforation. The page length is determined by the LTAB specification.

COPY= specifies the number of copies of printed output from the list queue entry. This parameter is ignored when the output is not spooled to disk. The parameter is specified by one or two digits. If omitted or 0 or 00 is specified, one copy is printed. If additional copies are required when disposition is T, the tape must be reprinted.

Note: If the COPY parameter is used together with the RBS parameter, the page count on the printed output will not be in proper sequence.

COPYG= describes (for the IBM 3800 Printer) how the printed copies of the output file are to be grouped. Each "groupvalue" specifies the number of copies of each individual page that is to be printed in a group before starting to print the next page. POWER/VS retransmits the entire output file for each group value specified. Up to eight group values can be specified. The copy group index is used for counting of the group values. No single group value can exceed 255, nor can the sum of those specified exceed 255.

This parameter has effect only when the output file is directed to an IBM 3800 Printer. Otherwise it is ignored. This parameter is also ignored when the output is not spooled to disk. If the COPYG parameter is omitted, no copy grouping occurs.

TADDR= specifies the channel and unit address (and, optionally, the recording density) of the tape drive to be used for spooling to tape intermediate storage when the job is being executed. This parameter may be specified in one of the following ways:

cuu
X'cuu'
(cuu,ss)
(X'cuu',ss)
(cuu,X'ss')
(X'cuu',X'ss').

"ss" can only be specified in the keyword format. It shows the density of the output tape. The specification of "tapeaddr" forces a disposition of T. If DISP=T is specified and "tapeaddr" is omitted, the operator is requested to specify the tape address during job execution via message 1Q55D.

Two or more files can be written to a single tape. Multivolume files are also supported.

Note: The output of a multivolume file is split at logical boundaries. In case of printer output, this is a skip to channel 1. That means that a print output that does not contain skips to channel 1 cannot be split into two or more volumes.

The RBS parameter is not active when spooling onto tape.

RBM=
norbm1

specifies the number of lines (list records) intercepted before message 1Q52I is sent to the operator. This parameter is specified by one to six digits. If "norbm1" is omitted, the value defined during POWER/VS generation in the first entry of the STDLINE parameter of the POWER macro is assumed.

norbm2

specifies the additional number of list records intercepted each time before message 1Q52I is reissued. It is specified by one to six digits. If "norbm2" is omitted, the preceding comma and the parentheses may be omitted and the value defined during POWER/VS generation in the second entry of the STDLINE parameter of the POWER macro is assumed. This value is also assumed when the positional form of the LST statement is used.

LTAB=

specifies the carriage control tape format to be used by POWER/VS when print requests are intercepted for a non-FCB-type printer. As a default, the value of the LTAB= parameter specified in the POWER macro is used. This parameter, coded as 13 two-digit decimal values, represents the positions of the first line on the page and channels 1 through 12. Channel 1 must be the first channel of a page. The format is:

d0d1d2d3d4d5d6d7d8d9d10d11d12

where d0 is the number of lines between channel 12 of the preceding page and channel 1 of the new page; d1 is the number of lines between channel 1 and the first line of the printed page, it should always be 00. d2 is the number of lines from channel 1 to channel 2; and finally, d12 is the number of lines from channel 1 to channel 12. If channel 1 is the same as the first line of the printed page, d1 therefore is coded as 00. Similarly, if a channel is not represented, it must be coded 00, but channel 1 must always be present on the control tape or in the FCB. The sum of d0 and d12 is assumed to specify the maximum number of lines that can physically fit on a page. If d12 is specified as 00, a default value is taken from the communication region. If a skip is encountered to a channel that was not specified, message 0P73I is issued.

An LFCB macro issued in the problem program updates the LTAB format.

Notes:

1. It is advisable to use the "forms-number" parameter in conjunction with the "linetab" parameter. This causes the list writer task to pause, thus allowing the operator to mount the correct carriage control tape on the printer.
2. Channel 1 on the carriage tape is used as an indicator for a page boundary and should therefore be specified.

RBS= specifies the number of list pages (that is, the number of skips to channel 1) intercepted before the output is segmented (count-driven output segmentation). It is used only when spooling to disk intermediate storage. This operand is ignored if DISP=T was specified. It is specified by one to six digits and overrides the POWER/VS generation specification of the RBS= parameter of the POWER macro. A zero means that no segmentation will take place.

When output segmentation occurs, the operator is notified via message 1Q53I.

UCS=
or
FCB=

specifies the name of the phase under which an FCB (Forms Control Buffer) or UCB (Universal Character Set Buffer) image is cataloged in the core image library. Together with the UCB phase name, an option may be specified which can be either F, C, FC, or CF, where

F - signifies that the UCB is to be loaded with the folding operation code to permit printing of uppercase for lowercase bit configurations.

C - Prevents the data checks from being generated because of print line mismatches with the UCB.

The specified core image library phase is loaded into the FCB or UCB when a list writer task starts processing the output related to the * \$\$ LST statement.

If UCS is coded and CHARS is not coded in the * \$\$ LST (or * \$\$ PRT) statement and if the output file is assigned to a 3800 printer, the UCS name is used as the first character arrangement table name. For more information on the CHARS parameter, see description of the * \$\$ LST statement later in this chapter.

Return to a default UCB can only be made via a subsequent * \$\$ LST statement.

When an * \$\$ LST statement with an FCB phase name specification is encountered by the execution processor, the specified buffer image is read to update the internal LTAB representation to reflect the new buffer. Any LTAB specification in the * \$\$ LST statement is therefore ignored if FCB is also specified. The FCB specification for a non-PCB printer is accepted, but is only used to update the LTAB. For output directed to forms control buffer printers, POWER/VS will ensure that the standard FCB (depending on the printer type) is loaded if no FCB parameter or LST statement is supplied for the job.

Return to a default FCB is made at the end of the processing of the current LST entry.

When the IBM 3800 Printer is installed and the first four characters of the FCB phase name are \$\$\$\$, the execution processor changes the \$\$\$\$ depending on the device for which the FCB is processed. The prefixes that are inserted for \$\$\$\$ are:

FCB1 for a 3800 Printing Subsystem
FCB2 for a 3211 printer
FCB3 for a 3203 printer
FCB4 for a 5203 printer.

This gives you device-independent processing for FCBs. However, when using this feature, it is still your responsibility to supply FCB images with phase names according to the above convention.

This parameter is not supported by the positional form of the * \$\$ LST statement.

When using SNA terminals or workstations for the production of list output, you should consider the following:

As POWER/VS does not support Select Vertical Format (SVF), it is the responsibility of the remote operator to set up the vertical tab tables at the remote workstation. However, the FCB name is sent if PDIR is supported and if the name has been specified in the JECL statements * \$\$ LST or * \$\$ PRT.

Note: When the output file is directed to an 3800 printer, the first four characters of the FCB phase name must be FCB1.

It is advisable to use the "forms-number" parameter in conjunction with the "FCB" parameter. This causes the list writer task to pause, thus allowing the operator to mount the correct carriage control tape on the printer.

LST=

specifies either the logical or physical address of the output printer for the list queue entry being created. The address can be either in the form SYSxxx, where xxx is LST or any valid programmer logical unit, or in the form cuu (or X'cuu'), which is the channel and unit number of an actual or a dummy printer. SYSxxx must be assigned to the correct printer before the * \$\$ LST statement is processed. If a program produces two or more different reports (which, in addition, may require different forms), this parameter must be used to define specific printers for each of the reports. If this parameter is omitted, the default is the first printer address specified at partition start-up. For example, if at start-up, the following printers were specified on SYSLOG: PRINTERS=00E,01E,02E, then printer 00E would be the default. In order to have the * \$\$ LST statement parameters take effect for printer 01E or 02E, LST=01E or LST=02E must be specified.

This parameter is not supported by the positional form of the * \$\$ LST statement.

CMPACT= specifies the name of the compaction table to be used. This name may consist of 1 to 4 alphameric characters. If NO is specified, no compaction will be performed for this job.

If the CMPACT parameter is omitted, the default compaction table specified in the PRMT macro will be used.

BURST= This parameter applies to 3800 printers only. BURST specifies whether the output printed by the 3800 is to go to the optional Burster-Trimmer-Stacker. Valid specifications are Y or N, where:

Y - indicates that the printed output is to be burst into separate sheets.

N - indicates that the output is to be continuously fanfold.

When the BURST parameter is omitted, the default paper threading request is taken (this default may have been established using the SETDF operator command). Also, when DFLT=N has been specified, BURST=N is assumed as the default.

When you direct your output to the Burster-Trimmer-Stacker, stacked sheets of each output file are automatically separated from those of the preceding file. Copies and copy groups are also separated.

The BURST parameter is syntax-checked but ignored for printers other than the 3800.

CHARS= This parameter applies to 3800 printers only. CHARS specifies the character arrangement tables to be used by the 3800 for printing of the output file. "table name" specifies the 1- to 4-character name of the character arrangement table (not including the system-assigned prefix, XTB1, of the phase name). Up to four table names can be specified. For more information on character arrangement tables and their use, see the DOS/VS IBM 3800 Printing Subsystem Programmer's Guide, GC26-3900.

If CHARS is coded and UCS is not coded in the * \$\$ LST (or * \$\$ PRT) statement and if the output file is assigned to an impact printer with the UCS feature, then the first character arrangement table name is used as the UCS name.

The CHARS parameter is syntax-checked but ignored for printers other than the 3800.

DFLT=

This parameter applies to 3800 printers only. It specifies whether the IBM 3800 Printer is to be set with the defaults that are specified by the operator in the SETDF command.

Y - specifies that the defaults previously established for the particular printer are to be used. The default values are used when the corresponding parameter is not specified. Defaults can be established by means of the SETDF command for the following parameters: BURST, CHARS, FCB, FLASH, FNO (forms), and MODIFY.

N - specifies not to use the defaults established with the SETDF command, but to use the 3800's hardware defaults instead, when needed.

When this parameter is omitted, DFLT=Y is the default.

The DFLT parameter is syntax-checked but ignored for printers other than the 3800.

FLASH=

This parameter applies to 3800 printers only. It specifies the 1- to 4-character name of the forms overlay frame to be used by the 3800 printer. Following overlayname a flash count can be specified. It specifies the number of copies to be flashed with the overlay, beginning with the first copy. Any specification from 0 to 255 is valid. If the overlay name is specified without a count or a comma after the name, or if the count value is greater than the number of copies specified (with the COPY or COPYG parameter), all copies are flashed. If the name is followed by a comma but no count is specified, a value of 0 is assumed as default (no copies are flashed).

If a count is specified without an overlay name, the forms overlay frame loaded at the time of printing is used. If a count of 0 is specified, the operator is asked to load the requested forms overlay frame but the overlay is not flashed.

The FLASH parameter has effect only for a 3800 printer. When specified for another printer, the parameter is syntax-checked but ignored.

If the FLASH parameter is omitted and DFLT=Y has been specified, the default forms overlay name established for the printer by means of the SETDF operator command is taken. If no default forms overlay name has been established, or if DFLT=N was specified, no flashing occurs for the data set.

When a list writer task for an IBM 3800 Printer is to process output from a list queue entry, it determines whether the overlay name matches that of the previous list queue entry. If they match, the list writer task processes the output. If they do not, message 1QA5A is issued, requesting the operator to insert that particular forms overlay frame into the 3800. There is, however, no verification by POWER/VS that the requested frame has been inserted.

MODIFY= This parameter applies to 3800 printers only. Predefined data can be printed on all pages of an output file for the 3800 printer. The data can be specified by using the MODIFY parameter.

"copymodname" specifies the 1- to 4-character name of the copy modification phase that was assigned when it was built (not including the system-assigned prefix MOD1).

"tablename" is the 1- to 4-character name of the character arrangement table to be used when printing the copy modification text. This character arrangement table need not be one of those specified with the CHARS parameter, if the 3800 has enough character generation storage. If "tablename" is not specified, the first character arrangement table specified or defaulted with the CHARS parameter is used for the copy modification.

The MODIFY parameter is syntax-checked but ignored for printers other than the 3800.

* \$\$ PUN Statement

The * \$\$ PUN statement defines the attributes of the intercepted punch output of a POWER/VS job until a subsequent * \$\$ PUN or an end-of-job (* \$\$ EOJ or /&) statement is encountered. Any number of * \$\$ PUN statements is permitted per POWER/VS job, except in a reader/ writer partition, where only one statement per device and per job is permitted. If the * \$\$ PUN statement is not included in the job, the default values established during POWER/VS generation are assumed. * \$\$ PUN statements do not require * \$\$ JOB and * \$\$ EOJ statements also to be used.

For each * \$\$ PUN statement within a POWER/VS job, a separate punch queue entry and job number is created (data-driven output segmentation; see the section "Concepts and Functions" in Chapter 1). The first punch entry will contain the same job number as the associated RDR queue entry. However, any subsequent punch entries will have a unique job number. Repeated use of the SEGMENT macro in the same job will have the same effect.

Note: Data-driven output segmentation is not supported in a writer-only partition.

If an error is detected in the * \$\$ PUN statement, the central operator may correct the statement during job execution.

Positional format:

Operation	Operands
* \$\$ PUN	[<u>D</u> disposition [<u>A</u> class]], [forms-number], [<u>1</u> number-of-copies], [norbm1] tapeaddr

Keyword format:

Operation	Operands
* \$\$ PUN	<pre> [DISP={D {disposition}} [,CLASS={A {Class}} [,PRI=priority] [,REMOTE=remid] [,FNO=forms-number] [,JSEP=(sep[,option])] [,COPY={1 {number-of-copies}} [,TADDR=tapeaddr[,X'ss']] [,RBM=(norbm1,norbm2)] [,RBS=norbs] [,PUN=punaddr] </pre>

Note: The operands PRI, REMOTE, JSEP, RBS and PUN can only be specified in the keyword form.

- DISP= specifies the disposition of the punched output after execution of the job. The disposition specification options are listed below. D, H, I, K, and L, are only valid when the output is spooled to disk. When the output is punched at a terminal, N and T are invalid.
- D - Delete after processing. The punch queue entry is automatically written to a punch according to its class and priority.
 - H - Hold. The punch queue entry remains in the punch queue; it is not written to a punch by POWER/VS until the operator changes the disposition to D or K by means of the PALTER command or issues the PRELEASE command.
 - I - Return output to input queue. In some situations it may be useful to return the punch output of a POWER/VS job directly to the input queue. For example, the assembler normally stores its output (including JCL) on tape, for subsequent input to a catalog run; under POWER/VS however, it can store output directly in the input queue.

The newly-built input queue entry has the same jobname, remote-ID, and priority as the entry that produces the output. Class becomes that specified in the * \$\$ PUN statement; disposition becomes D. This facility should only be used for jobs producing punch output in executable format.

The I option is ignored in a writer-only partition. The option suppresses count-driven output segmentation and also ignores the multiple-copies specification.

- K - Keep after processing. The output is automatically written to a punch according to its class and priority. When punching is complete, the punch queue entry is not deleted from the punch queue, and the disposition of the queue entry becomes L.
- L - Leave in queue. The job remains in the punch queue; it is not written to a punch until the operator changes the disposition to K or D by means of the PALTER command or issues the PRELEASE command.
- N - No spooling. If N is specified, punch requests are not intercepted. It allows direct writer capabilities without spooling. If the N option is specified and the device is not available at execution time, message 1Q46I is issued and punch output is spooled.
- T - Tape spooling. The output is spooled to tape that is used as intermediate storage.

CLASS= specifies the class of the punch output of the job. It may be specified as any alphabetic character (A through Z). "class" may be used in conjunction with "forms-number" to group together the same types of punch output. Punch tasks can be specified by class to process the output. A class specification is only meaningful when the output is spooled to disk.

If "class" is specified and the positional form is used, "disposition" must also be specified. If "class" is omitted, class A is assigned. If the * \$\$ PUN statement is omitted, an output class can be specified in the PSTART command.

Note: There is no comma between "disposition" and "class" in the positional form.

PRI= specifies the priority (0-9) of the punch output. If omitted, the priority of the job is taken as default.

REMOTE= specifies the TO remote-ID to which punch output is to be routed. Each punch queue entry also has a FROM remote-ID assigned to it, which identifies the user who initially submitted the job. "remid" can be a number from 0 to 200. Numbers 1 to 200 are specific remote users. If "remid" is specified as 0, punch output is directed to the central installation. If this parameter is omitted or if the positional form of the * \$\$ PUN statement is used, punch output is routed to the central location for locally submitted jobs or is routed according to the PUNROUT specification in the PRMT macro for jobs submitted remotely.

FNO= specifies, by one to four alphanumeric characters, /, -, or period (.), the type of card forms to be used. If a forms number is not specified, it is assumed to be four blanks, representing the installation's standard type of card forms. When a punch writer task is to process output from a punch queue entry, it determines whether this forms number matches that of the previous punch queue entry. If they match, the punch writer task processes the output. If they do not match, message 1Q40A is issued. If the first queue entry processed after a PSTART has a forms ID of four blanks, no mount message is issued. If the output is directed to a 3741, this parameter must not be specified.

JSEP= specifies job separation. 0-9 can be specified. If 0 is specified, no job separation occurs. Job separation means that before the job's punched output one card containing the POWER/VS jobname (punched so that it can be read on the back of the card) and from two to eight cards containing 12-11-0-8-9 punches (in all 80 columns) are added and that behind the job's punched output two blank cards are added. This occurs if 1, 2, or 3 is specified. If 4 is specified, one additional 12-11-0-8-9 card is punched; if 5 is specified, two additional 12-11-0-8-9 cards are punched, and so on.

Note: Stacker selection is ignored if job separation is requested. The default stacker for the given device is used instead.

If the 3800 ICR is installed, an option (either Y or N) can be specified following the "sep" parameter, where:

Y - specifies that separator cards are also included between copies of each punch output.

N - suppresses the inclusion of separator cards between copies of the punch output.

When the option is not specified, the COPYSEP specification defined during POWER/VS generation is assumed.

Note: The option is only in effect when a nonzero JSEP specification is defined, either during POWER/VS generation or in the * \$\$ PUN statement.

COPY= specifies the number of copies to be punched from the punch queue entry. The parameter is specified by one or two digits. If it is not specified, 1 is assumed. If 0 or 00 is specified, one copy is punched. This parameter is ignored when the output is not spooled to disk. If additional copies are required when the disposition is T, the tape must be repunched.

TADDR= specifies the channel and unit address (and, optionally, the recording density) of the tape drive to be used for spooling when the job is executed. Its format may be specified in one of the following ways:

```
cuu
X'cuu'
(cuu,ss)
(X'cuu',ss)
(cuu,X'ss')
(X'cuu',X'ss').
```

"ss" can only be specified in keyword format. It shows the density of the output tape. The specification of "tapeaddr" forces disposition T. If DISP=T is specified and "tapeaddr" is omitted, the operator is requested via message 1Q55D to specify the tape address at job execution time.

Two or more files can be written to a single tape. Multivolume files are also supported.

RBM=
norbm1

specifies the number of punch records intercepted before message 1Q52I is sent to the operator. This parameter is specified by one to six digits. If "norbm1" is omitted, the value defined during POWER/VS generation in the first entry of the STDCARD parameter of the POWER macro is assumed.

norbm2

specifies the additional number of punch records intercepted before message 1Q52I is issued again. It is specified by one to six digits. If "norbm2" is omitted, the preceding comma as well as the parentheses may be omitted and the value defined during POWER/VS generation in the second entry of the STDCARD parameter of the POWER macro is assumed. This value is also assumed if the positional form of the * \$\$ PUN statement is used.

RBS=

specifies the number of punch records intercepted before the output is segmented (count-driven output segmentation). It is used only when spooling to disk intermediate storage. This parameter, which is ignored if DISP=T was selected, is specified by one to six digits and overrides the POWER/VS generation specification of the RBS parameter of the POWER macro. A zero means that no segmentation will take place. When output segmentation occurs, the operator is notified via message 1Q53I. For segmentation on a 3525, and if print commands are used, all segments must be punched in a row, or separator cards must be specified. This is necessary to ensure that the last two records will not be lost.

PUN=

specifies either the logical or physical address of the output punch for the punch queue entry being created. The address can be either in the form SYSxxx, where xxx is PCH or any valid programmer logical unit, or in the form cuu (or X'cuu'), which is the channel and unit number of an actual or a dummy punch. If this parameter is omitted, the default is the first punch address specified at partition start-up. This parameter is needed when two or more punches are used to punch two or more card files produced by a single program.

* \$\$ RDR Statement

The * \$\$ RDR statement is used to insert a 3540 diskette file into input being read from the card reader. There are two modes of input processing, which are:

1. SYSIN-mode processing; the physical unit number of the 3540 diskette device must not be specified. The file to be read from the diskette must contain 80- or 81-byte records. Only the last 80 bytes of an 81-byte record are passed to the reader. The file to be read from the diskette must contain the necessary JECL and/or JCL job stream if the POWER/VS command PSTART RDR does not specify both card reader and 3540 physical unit addresses.
2. Data-mode processing; the physical unit number of the diskette from which the input data file is read must be specified. The JECL and/or JCL statements are read from the card reader. If two or more RDR statements are issued in one POWER/VS job, they must specify the same physical unit number.

Note: In a writer-only partition, this statement is ignored.

Positional format:

Operation	Operands
* \$\$ RDR	[diskette-addr],['file-id'], [1 number-of-diskettes] , [S]

Keyword format:

Operation	Operands
* \$\$ RDR	[DEV=diskette-addr] [,FID='file-id'] [,NOD={ 1 number-of-diskettes }] [,VSC={ NO } YES }] [,VER={ NO } YES }] [,FEED={ NO } YES }]

Note: The operands VER and FEED can only be specified in the keyword form of the statement.

DEV= specifies the physical unit number (X'cuu' or cuu) used by the user program to read the input data file. The physical unit must not be specified for SYSIN files. In one POWER/VS job, more than one RDR statement can be specified to insert more than one 3540 data file, but the different RDR statements must indicate the same physical unit number in the DEV parameter.

FID= specifies the file name (in quotes) as it appears in the HDR1 label on the 3540 diskette. One to eight alphameric characters can be specified. If this parameter is omitted, POWER/VS reads the file specified by the first non-secured HDR1 label found on the diskette that is currently mounted. The size of the record to be read is taken from the HDR1 label.

NOD= specifies the maximum number of diskettes to be read. One to three digits can be specified. Allowed values are 1 to 255. The default value is 1. Reading stops after the specified number of diskettes are processed, or after reading a diskette whose HDR1 label indicates that it was the last volume of the file.

VSC= specifies whether volume sequence checking is desired. If omitted, no sequence checking will take place. Sequence numbers must start with 1 and must be incremented by 1 up to a maximum of 99.

VER= specifies optional file verification. If VER=YES is specified, POWER/VS checks that file verification had been performed on the diskette file (verify field in the HDR1 label of the file set to 'V'). If VER=YES and the verify field was not set to 'V', the file is rejected during file OPEN. If VER=NO or if the parameter is omitted, the verify field in the HDR1 label is ignored during processing.

FEED= specifies the option to eject and feed a new diskette at end-of-file of the preceding diskette. This parameter can be used in combination with the FEED parameter of the POWER macro instruction. During the POWER/VS generation, you have established the system default value for the FEED option (YES or NO); this system default can be overridden by the FEED parameter of the * \$\$ RDR JECL statement. Any time a * \$\$ RDR statement is going to be processed, the FEED parameter is reset to the system default as specified in the POWER macro instruction.

* \$\$ SLI Statement

During program execution, the * \$\$ SLI statement causes the private and the system source statement library to be searched for a book, which, when found, is inserted into the job stream. The private source statement library, if assigned to the POWER/VS partition, is searched first.

An * \$\$ SLI statement does not require that * \$\$ JOB and * \$\$ EOJ statements also be used.

Statements may be added to or deleted from the source statement library books. Such update statements may be inserted before or after SSL statements or may replace or delete them. They have special codes in columns 73-80:

columns 73-76 contain '\$SLI'
 columns 78-80 contain a sequence number
 column 77 contains a special code:

- A - Insert after book statement with the same number or before the one with the next higher sequence number.
- B - Insert before book statement with equal or higher sequence number.
- D - Delete the book statement with the same sequence number.
- Other - Replace the book statement with same sequence number or insert this statement.

Source statement library update statements and * \$\$ DATA statements, when used, must follow the * \$\$ SLI statement in the same sequence as the statements in the book they correspond to.

Books are cataloged into the source statement library with the DOS/VS program MAINT. To include /* and /& statements in a book, submit them as * \$\$/* and * \$\$/& statements since /* and /& statements are not accepted as data by MAINT. To include LST and PUN statements, submit them as * \$x LST or * \$x PUN, where x can be any character other than blank or \$. If the book is not found in the specified sublibrary, message 1Q44I is issued and the job is flushed.

Operation	Operands
* \$\$ SLI	[[sublib.]bookname

Note: This statement is only available in the positional format.

sublib. specifies the name of the sublibrary to be searched. Valid sublibrary names are A through Z, 0 through 9, \$, #, or @, followed by a period (.). Sublibraries A, C, D, E, F, and Z are already defined as special-purpose libraries in DOS/VS. This parameter is optional. If omitted, the sublibrary specification at POWER/VS generation is used.

bookname specifies the book name to be searched for and inserted into the job stream. If "bookname" begins with \$\$, it will be changed depending upon the partition in which the SLI book is being processed. \$\$ becomes \$B in BG or \$n in the foreground partition Fn.

The following examples illustrate the use of the * \$\$ SLI and * \$\$ DATA statements:

1. Job to insert data from the card reader into the book "ASSM".

<u>Job from card reader</u>	<u>Source book "ASSM"</u>
* \$\$ JOB INSERT	BKEND ASSM
* \$\$ SLI ASSM	// JOB INSERT
* \$\$ DATA INPUTA	// EXEC ASSEMBLY
POWER ACCOUNT=YES	* \$\$ DATA INPUTA
END	* \$\$/&
/*	BKEND
* \$\$ EOJ	
 <u>Resulting job stream</u>	
// JOB INSERT	(From book "ASSM")
// EXEC ASSEMBLY	(From book "ASSM")
POWER ACCOUNT=YES	(From card reader)
END	(From card reader)
/*	(From card reader)
/*	(From book "ASSM")

2. Assume that a private source statement library is assigned to the POWER/VS partition. A job (read in from the card reader) references the book "XMPL" in that library (sublibrary B) and contains SL update statements as well as DATA statements. The book statements replace the SLI statement. However, statements XMPL0020, XMPL0100, and XMPL0120 are overridden by the corresponding SL update statements in the input stream. In addition, statement \$SLIA100 is inserted. Also, the cards between the DATA statement and the first /* statement in the input job stream replace the DATA statement in XMPL. The resulting job stream is placed in the read queue.

Job from card reader

Col. 73-80

```
* $$ JOB ASSEMBLI
* $$ SLI B.XMPL
// ASSGN SYSPCH,X'182' $SLI0020
* $$ DATA INPUTA
DEFAULT POWER ACCOUNT=YES
END
/*
// ASSGN SYSIPT,X'182' $SLI0100
// ASSGN SYS010,X'183' $SLIA100
// MTC RUN,X'182' $SLI0120
* $$ EOJ
```

Book "XMPL" in source statement library

Col. 73-80

```
BKEND B.XMPL
// JOB ASSEMBLE XMPL0010
// ASSGN SYSPCH,X'180' XMPL0020
// EXEC ASSEMBLY XMPL0030
* $$ DATA INPUTA XMPL0040
// MTC WTM,SYSPCH XMPL0050
// MTC WTM,SYSPCH XMPL0060
// MTC REW,SYSPCH XMPL0070
* $$/& XMPL0080
// JOB CATALOG XMPL0090
// ASSGN SYSIPT,X'180' XMPL0100
// EXEC MAINT XMPL0110
// MTC RUN,X'180' XMPL0120
* $$/* XMPL0130
* $$/& XMPL0140
// JOB LINK XMPL0150
// OPTION CATAL XMPL0160
INCLUDE DEFAULT XMPL0180
* $$/* XMPL0200
// EXEC LNKEDT XMPL0210
* $$/& XMPL0220
BKEND
```

Resulting job stream placed in read queue

Col. 73-80

```
* $$ JOB      ASSEMSLI
// JOB        ASSEMBLE           XMPL0010
// ASSGN      SYSPCH,X'182'      $SLI0020      updated
// EXEC       ASSEMBLY           XMPL0030
DEFAULT POWER ACCOUNT=YES      inserted
END                               inserted
/*                               inserted
// MTC        WTM,SYSPCH         XMPL0050
// MTC        WTM,SYSPCH         XMPL0060
// MTC        REW,SYSPCH         XMPL0070
/&                               XMPL0080      modified
// JOB        CATALOG           XMPL0090
// ASSGN      SYSIPT,X'182'      $SLI0100      updated
// ASSGN      SYS010,X'183'      $SLIA100      inserted
// EXEC       MAINT              XMPL0110
// MTC        RUN,X'182'         $SLI0120      updated
/*                               XMPL0130      modified
/&                               XMPL0140      modified
// JOB        LINK               XMPL0150
// OPTION     CATAL              XMPL0160
INCLUDE DEFAULT                  XMPL0180
/*                               XMPL0200      modified
// EXEC       LNKEDT             XMPL0210
/&                               XMPL0220      modified
* $$ EOJ
```

JECL Delimiter Statements

The end-of-jobstep and end-of-job statements are used to delimit DOS/VS job definitions within the source statement library books (see the * \$\$ SLI statement above).

The end-of-jobstep statement delimits DOS/VS job definitions within the source statement library books.

Operation	Operands
* \$\$ /*	

This statement has no operands.

The end-of-job statement delimits DOS/VS jobs within the source statement library books.

Operation	Operands
* \$\$ /&	

This statement has no operands.

Notes:

1. LST and PUN statements within the SLI books may not be continued from one record to the next.

2. The following rules and restrictions must be considered:
 - Nested SLI statements are not permitted.
 - An SLI book may invoke a DOS/VS procedure.
 - A DOS/VS procedure may not invoke an SLI book.
 - Each SLI book represents a POWER/VS procedure and may not include embedded * \$\$ JOB statements. However, one POWER/VS job may include two or more SLI statements.
 - * \$\$ LST and * \$\$ PUN statements may not be continued in an SLI book. However, there is no restriction on the use of // JOB, * \$\$ /*, and * \$\$ /& statements.
 - SYSRDR, SYSIPT, and SYSIN cannot be reassigned within an SLI book.
3. The private source statement library containing the SLI books must be assigned to the POWER/VS partition before POWER/VS is started.

Cross-Partition Communication Macros

This section describes the macros that allow access to POWER/VS functions from a partition not controlled by POWER/VS. These macros can be used only if SPOOL=YES was specified in the POWER macro during POWER/VS generation, and if XECB support was specified during DOS/VS generation.

POWER/VS answers spooling requests originating from the SVA; however, the DTFs (or CCBs), CCWs, and data areas must reside in a partition and not in the SVA.

To connect to POWER/VS, use the XECBTAB macro with the following operands:

```

XECBTAB TYPE=DEFINE,
        XECB={SPMXECB},
           {ICRXECB}
        ACCESS=XWAIT
```

SPMXECB is required for the GETSPOOL and CTLSPPOOL macros; ICRXECB is required for the PUTSPOOL macro. The 3-byte address of the SPL must be placed in SPMXECB+5 and/or ICRXECB+5 by the user program.

Notes: For each program using the GETSPOOL, PUTSPOOL, and CTLSPPOOL macros, the specification of the SPL TYPE=MAP macro is required.

If the partition using the cross-partition support has a higher priority than POWER/VS, a system error may be issued if an abnormal end or shut down occurs before all cross-partition active tasks have completed.

If the cross-partition user issues JECL statements to POWER/VS through the support macros, the format is not variable. Only one blank must appear between the "*" and "\$\$" and the first parameter.

Only one partition may actively use the cross-partition support at a time.

MACRO DESCRIPTIONS

SPL Macro

The SPL (Spool Parameter List) macro creates a parameter list for the PUTSPOOL, GETSPOOL, and CTLSPOOL macros.

Name	Operation	Operands	Col
[name]	SPL	TYPE=DEFINE	[*]
		[,JOBN=jobname]	[*]
		[,DISP=disposition]	[*]
		[,CLASS=class]	[*]
		[,REQ= { PRI DISP CLASS REMOTE CANCEL SCRATCH STATUS COMMAND }]	[*]
		[,CBUF=firstbufaddr]	[*]
		[,PBUF=bufaddr]	[*]
		[,PBUFL=buflength]	[*]
		[,NEWVAL=value]	

or

[name]	SPL	TYPE=MAP	[*]
		[,SPMXECB= { YES NO }]	[*]
		[,ICRXECB= { YES NO }]	

Programming Notes

TYPE= causes, with the specification DEFINE, a parameter list to be initially set up with the keywords and values to be used in the PUTSPOOL, GETSPOOL, and CTLSPOOL macros. These values can later be changed by specifying the corresponding keywords in the macros themselves.

With the specification MAP, this operand causes a DSECT of the SPL to be generated. TYPE=MAP must be specified once in every program using GETSPOOL, PUTSPOOL, CTLSPOOL. If TYPE=MAP is specified, only the parameters SPMXECB= and ICRXECB= (see below) may be coded.

JOBNAME= specifies a unique job name to be assigned to the POWER/VS input queue. If omitted, the default name is DUMMY.

DISP= specifies the output disposition code for the submitted job as it will appear on the POWER/VS output queue. If omitted, the default is K.

CLASS= specifies the POWER/VS output class (A-Z) for the submitted job as it will appear on the POWER/VS output queue. If omitted, the default is A.

REQ= specifies the requested operation for a CTLSPPOOL macro.

PRI alters the priority of the named job on the output queue (see note below).

DISP alters the disposition of the named job on the output queue (see note below).

CLASS alters the job class of the job on the output queue (see note below).

REMOTE alters the remote terminal identification to which list output of the job is to be routed (see note below).

CANCEL causes the job to be cancelled from the input queue if it has not yet executed.

SCRATCH causes the named job to be deleted from the POWER/VS output queue.

STATUS causes the disposition of the named job on the input or output queue to be returned in the SPQD field of the SPL, and the queue indicator in the SPSQ field.

COMMAND indicates that you have supplied a valid POWER/VS command in the PBUF area.

Note: If PRI, DISP, CLASS or REMOTE are used, the NEWVAL= operand must also be specified.

CBUF= specifies the address of the first buffer containing the job stream. The format of each 88-byte buffer area is as follows: a four-byte pointer to the next buffer in the chain (or 0 for the last buffer), four bytes reserved, and an 80-byte data buffer area.

PBUF= specifies the address of a buffer area which is for use by POWER/VS and for POWER/VS feedback information on certain error conditions. The length of this buffer area must be at least 88 bytes.

PBUFL= indicates the length of PBUF. It must be set to the maximum length of the output data records, with trailing blanks truncated. The minimum allowed is 88 bytes. If omitted, the default is 88 bytes.

NEWVAL= indicates the new value that is to be assigned to the PRI, DISP, CLASS, or REMOTE attribute of the job named in a CTLSPPOOL request. The meaning of the one-byte value depends on the REQ operand.

SPMXECB=
and
ICRXECB=

specify that a DSECT for each of these XECBs is to be generated. The format of the DSECTS is as follows:

	DSECT	XECB	FOR GETSPOOL AND CTLSPool
SPMXECBU	DS	F	XECB
SPMERROR	DS	X	ERROR RETURN CODE
SPMSPLP	DS	AL3	SPL pointer
	DSECT	XECB	FOR PUTSPOOL
ICRXECBU	DS	F	XECB
ICRERROR	DS	X	ERROR RETURN CODE
ICRSPLP	DS	AL3	SPL POINTER

PUTSPOOL Macro

The PUTSPOOL macro is used to submit a job stream from the user's buffer area to the POWER/VS input queue for later execution in a partition under control of POWER/VS.

Only the statements of the initial PUTSPOOL job stream are analyzed for the existence of any POWER/VS JECL statements, in particular * \$\$ JOB, LST, or PRT statements. If no JECL statements are detected at the beginning of the initial PUTSPOOL job stream, * \$\$ JOB and LST/PRT statements are built in accordance with the given SPL parameters. If any JECL statements are encountered at the beginning of the initial PUTSPOOL job stream, they are first placed in the input queue, and then any missing * \$\$ JOB and/or LST/PRT statements are constructed.

* \$\$ RDR statements cannot be submitted using the PUTSPOOL macro.

Note: When register notation is allowed for parameters of this macro (indicated by "(reg)" as a possible parameter value), any general register except registers 0, 1, 14, and 15 may be specified.

The register contains the parameter value, except for JOBN, whose register must point to an eight-byte field with the jobname right-adjusted.

Name	Operation	Operands	Col
[name]	PUTSPOOL	SPL=(reg)	[*]
		[,JOBN={jobname} {(reg)}]	[*]
		[,PBUF={bufaddr} {(reg)}]	[*]
		[,CONT=(reg)]	[*]
		[,DISP={disposition} {(reg)}]	[*]
		[,CLASS={class} {(reg)}]	[*]
		[,CBUF={firstbufaddr} {(reg)}]	[*]

Programming Notes:

name	specifies the macro label. Required if the CONT parameter is specified.
SPL=	specifies the address of the spool parameter list (SPL) to be used by the PUTSPOOL macro. The SPL defines the POWER/VS request for the job stream indicated by the JOBN operand.
JOBN=	specifies a unique job name to be assigned to the POWER/VS input queue. The same name must be used for the output queue when it is referenced by the GETSPOOL or CTLSPOOL macros.
PBUF=	specifies the address of a buffer area which is for use by POWER/VS and for POWER/VS feedback information on certain error conditions. The length of this buffer area must be at least 88 bytes.
CONT=	If the buffers processed by this execution of PUTSPOOL do not contain the complete job stream, this operand should be used to give the address of a continuation routine. In this routine, you can submit further data buffers associated with the same job stream. However, no other parameters may be changed in the continuation routine. Return to POWER/VS is via register 14. When this parameter is used, CBUF= must also be specified and the macro must have a label.
DISP=	specifies the output disposition code for the submitted job as it will appear on the POWER/VS output queue (LST or PUN). The disposition must be a valid POWER/VS disposition. If you supply a * \$\$ LST or * \$\$ PRT statement, it must contain the same disposition code.
CLASS=	specifies the POWER/VS output class (A-Z) for the submitted job as it will appear on the POWER/VS output queue (LST or PUN). Specify a unique class if you want to have controlled access to the file(s) by means of GETSPOOL. If you supply a * \$\$ LST or * \$\$ PRT statement, it must contain the same output class.
CBUF=	specifies the address of the first buffer containing the job stream. The format of each 88-byte buffer area is as follows: a four-byte pointer to the next buffer in the chain (or 0 for the last buffer), four bytes reserved, and an 80-byte data buffer area. If the CONT operand is specified together with CBUF, the same buffers must be reused for the continuation data. If register notation is used, the continuation routine must reset the contents of the previously used register if different buffers are used.

GETSPOOL Macro

The GETSPOOL macro is used to request retrieval of list output currently held in POWER/VS output queues on disk. The output is to be returned to the buffer area of the partition issuing the GETSPOOL macro. Retrieval may be sequential or direct.

GETSPOOL can request list output only if the disposition of the queue entry is D or K. However, as for a list task, the disposition of the queue entry is changed to L after processing. Therefore, before a subsequent GETSPOOL request can be issued, the disposition of the entry has to be changed back to D or K, using the CTLSPool macro.

Note: When register notation is allowed for parameters of this macro (indicated by "(reg)" as a possible parameter value), any general register except registers 0, 1, 14, and 15 may be specified.

The register contains the parameter value, except for JOBN, whose register must point to an eight-byte field with the jobname right-adjusted.

Name	Operation	Operands	Col
[name]	GETSPOOL	SPL= (reg)	[*]
		[,JOBN={jobname} (reg)]	[*]
		[,CLASS={class} (reg)]	[*]
		[,PBUF={bufaddr} (reg)]	[*]
		[,PBUFL={buflength} (reg)]	[*]
		[,CC={YES } {NO }]	[*]
		[,LINENO={number} (reg)]	

Programming Notes:

- SPL=** specifies the address of the spool parameter list (SPL) to be used by the GETSPOOL macro. The SPL defines the POWER/VS request for the job stream indicated by the JOBN= operand. If the LINENO= operand was used in a previous direct GETSPOOL request, the same SPL must be specified in the present request, or line positioning will be lost.
- JOBN=** specifies the POWER/VS job name for the job stream as assigned by PUTSPOOL. If the LINENO= operand was used in a previous direct GETSPOOL request, the same job name must be specified in the present request, or line positioning will be lost.
- CLASS=** specifies the POWER/VS output class (A-Z) as assigned by PUTSPOOL.

PBUF= specifies the address of a buffer area which is to receive the retrieved output record and/or POWER/VS feedback information on certain error conditions. If this operand is used, PBUFL= must also be specified.

PBUFL= indicates the length of PBUF. It must be set to the maximum length of the output data records, with trailing blanks truncated. The length of the output data record, with trailing blanks truncated, is returned in the 4-byte SPRL field of the SPL. The minimum allowed is 88 bytes.

CC= If CC=YES is specified, the command code of the CCW associated with this data record is returned in the SPCC field of the SPL. If the record is a control record, the buffer contains blanks; if it is a data record, the SPCC field is always updated.

LINENO= If this is the first GETSPOOL request, sequential retrieval of the output records begins at the line specified (relative to the beginning of the output file), or, if LINENO= has been omitted, from the beginning of the file. Since any subsequent GETSPOOL request would reposition to the line number specified, the LINENO= operand should be omitted for subsequent sequential retrieval.

CTLSPOOL Macro

The CTLSPOOL macro is used to

- Alter the attributes of a POWER/VS job
- Cancel a submitted job prior to its execution
- Delete the list output of a job after its execution
- Display the status of a job
- Issue a valid user-supplied POWER/VS command.

Note: When register notation is allowed for parameters of this macro (indicated by "(reg)" as a possible parameter value), any general register except registers 0, 1, 14, and 15 may be specified.

The register contains the parameter value, except for JOBN, whose register must point to an eight-byte field with the jobname right-adjusted.

Name	Operation	Operands	Col
[name]	CTLSPOOL	SPL= (reg)	[*]
		[,JOBN={jobname}] (reg)	[*]
		[,PBUF={bufaddr}] (reg)	[*]
		[,REQ={ PRI DISP CLASS REMOTE CANCEL SCRATCH STATUS COMMAND (reg) }]	[*]
		[,NEWVAL={value}] (reg)	

Programming Notes:

- SPL= specifies the address of the spool parameter list (SPL) to be used by the CTLSPOOL macro. The SPL defines the POWER/VS request for the job stream indicated by the JOBN= operand.
- JOBN= specifies the POWER/VS job name for the job stream as assigned by PUTSPOOL.
- PBUF= specifies the address of a buffer area which is for use by POWER/VS and for POWER/VS feedback information. The length of this buffer area must be 88 bytes.
- REQ= specifies the requested operation as follows:
- PRI alters the priority of the named job on the output queue (see note below).
 DISP alters the disposition of the named job on the output queue (see note below).
 CLASS alters the job class of the job on the output queue (see note below).
 REMOTE alters the remote terminal identification to which list output of the job is to be routed (see note below).
 CANCEL causes the job to be cancelled from the input queue if it has not yet executed.
 SCRATCH causes the named job to be deleted from the POWER/VS output queue.
 STATUS causes the disposition of the named job on the input or output queue to be returned in the SPQD field of the SPL, and the queue indicator in the SPSQ field.
 COMMAND indicates that you have supplied a valid POWER/VS command in the PBUF area.
 (reg) indicates that the request code is provided in a general register. You may specify one of the following codes in this register:

X'01' (PRI) - Alter routing priority (PALTER)
 X'02' (DISP) - Alter routing disposition (PALTER)
 X'04' (CLASS) - Alter routing class (PALTER)
 X'08' (REMOTE) - Alter routing remote-ID (PALTER)
 X'10' (CANCEL) - Cancel input (PDELETE RDR)
 X'20' (SCRATCH) - Scratch output (PDELETE LST)
 X'40' (STATUS) - Display status of job (PDISPLAY)
 X'80' (COMMAND) - Issue POWER/VS command.

Note: If PRI, DISP, CLASS, or REMOTE are used, the NEWVAL= operand must also be specified.

NEWVAL= indicates the new value that is to be assigned to the PRI, DISP, CLASS, or REMOTE attribute of the named job. The meaning of the one-byte value depends on the REQ operand and can be as follows:

- For PRI, the new priority of the job
- For DISP, the new disposition of the job
- For CLASS, the new class of the job
- For REMOTE, the new terminal identifier.

The value specified can be either a character (for example, NEWVAL='A') or a number (for example, NEWVAL=16). If register notation is used, all values must be hexadecimal.

Note: When using the CTLSPool macro, you may specify one of the following POWER/VS operator commands:

- PALTER (A) command
- PCANCEL (C) command
- PDELETE (L) command
- PDISPLAY (D) command
- PRELEASE (R) command.

Any of these commands must be submitted in the 88-byte buffer addressed by the PBUF= parameter and invoked by the request type REQ=COMMAND.

The remaining operator commands are not available in the cross-partition communication environment.

RETURN CODES FROM PUTSPOOL, GETSPOOL, AND CTLSPool

Figure 3-5 shows the return codes that are supplied following the execution of a PUTSPOOL, GETSPOOL, or CTLSPool macro. These return codes are displayed in the SPER field of the SPL and in the error field of the DSECT generated by the TYPE=MAP operand of the SPL (at SPMXECB+4 or ICRXECB+4), except for return codes X'41' and X'88', which are indicated in XECB+4 only.

XWAIT and XPOST return codes in register 15 are to be diagnosed by the user (see macro expansions: XPOST errors are doubled to maintain uniqueness). The XWAIT and XPOST macros and the corresponding return codes are described in the section "Cross-Partition Event Control Macros" in the publication DOS/VS Supervisor and I/O Macros, GC33-5373.

Return Code	Meaning	Displayed at	
		XECB+4	SPER
X'08'	End of data encountered during a GETSPOOL request, or invalid LINENO specified in GETSPOOL.	X	X
X'11'	Command not allowed.	X	X
X'12'	Invalid POWER/VS output disposition in SPL.	X	X
X'14'	Invalid output class (not A-Z) in SPL.	X	X
X'18'	Invalid job name in SPL.	X	X
X'21'	The PBUF buffer area is smaller than 88 bytes or not large enough to hold the largest output data record (PBUFL in GETSPOOL too small).	X	X
X'22'	GETSPOOL was unable to locate output file by specified job name, job class, and dispatchable POWER/VS disposition, or requested output file is in use.	X	X
X'24'	A loop occurred in the PUTSPOOL buffer chain, or more than 4096 buffers were used per request.	X	X
X'28'	Invalid CTLSPOOL REQ operand.	X	X
X'41'	POWER/VS terminated normally, or POWER/VS terminated abnormally, or POWER/VS spool management task terminated abnormally.	X X X	X
X'42'	A POWER/VS error occurred during CTLSPOOL (see notes below).	X	X
X'44'	A POWER/VS error occurred during GETSPOOL (see notes below).	X	X
X'48'	A POWER/VS error occurred during PUTSPOOL (see notes below).	X	X
X'82'	Invalid data buffer chain (PUTSPOOL).	X	X
X'84'	Invalid POWER/VS buffer address (PBUF).	X	X
X'88'	Invalid SPL address.	X	

Notes:

- The first 44 characters of the POWER/VS message text are displayed in PBUF, starting at offset 28 of PBUF.
- No spool management error detection is done for a CTLSPOOL request with REQ=COMMAND specified. You must analyze the message returned by POWER/VS in the PBUF area. If the command in the CTLSPOOL request results in more than one message, only the first message is returned.
- All values specified in the NEWVAL operand of the CTLSPOOL macro must conform with the current POWER/VS values for priority, disposition, and related parameters.

Figure 3-5. PUTSPOOL, GETSPOOL, and CTLSPOOL Return Codes

Job Accounting

A single source program macro is provided for use in conjunction with POWER/VS: the PUTACCT macro. This macro relates to job accounting.

When using POWER/VS, the job accounting interface information and POWER/VS job accounting information is collected in the POWER/VS account file. The account file consists of five types of records:

- Reader account records
- List account records
- Punch account records
- Execution account records
- BSC line and SNA session account records.

The layouts of the account records are shown in Figures 3-6 to 3-11. On the account file, and after saving from the account file onto tape or disk, the records are included as physical sequential records. When saved on punched cards, account data for the reader, list, punch, and execution account records will appear in columns 2-72, and for the RJE account records in columns 2-57.

Column 1 of every card contains a copy of the record identifier (byte 42) of the record, columns 73-78 contain the sequential account record number and columns 79-80 contain the sequence number within that record.

READER ACCOUNT RECORD

A reader account record is created for each read queue entry that is entered into the POWER/VS system. Whether or not the queue entry has actually been placed in the queue file is indicated by the POWER/VS cancel code. The record is copied from the first 58 bytes of the corresponding queue record.

Bytes	Description	Format ¹
00-07	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	Start time of read, in packed decimal (OHHMMSSP; F = sign).	p
12-15	Stop time of read (OHHMMSSP; F = sign).	p
16-31	16 bytes of user information.	a
32-39	POWER/VS job name.	a
40-41	Job number assigned by POWER/VS.	b
42	Record identifier.	a
43	POWER/VS cancel code.	b
44	Reserved.	b
45-47	Reader device address, or SNA.	a
48	FROM remote ID.	b
49	TO remote ID.	b
50	Input class.	a
51	Input priority number.	a
52-55	Number of records read (including record added or deleted by an RDR exit routine).	b
56-57	Number of tracks for input storage.	b
¹ a: for alphameric b: for binary p: for packed decimal.		
<u>Note:</u> The TO remote ID is a dummy entry in the Reader Account Record and in the Execution Account Record. It is copied from a FROM remote ID.		

Figure 3-6. Reader Account Record

LIST ACCOUNT RECORD

A list account record is created for each list queue entry that is processed by a list task. It is a copy of the first 72 bytes of the corresponding queue record.

Bytes	Description	Format ³
00-07	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	Start time of list, in packed decimal (OHHMMSSP; F = sign).	p
12-15	Stop time of list (OHHMMSSP; F = sign).	p
16-31	16 bytes of user information from * \$\$ JOB card.	a
32-39	POWER/VS job name.	a
40-41	Job number assigned by POWER/VS.	b
42	Record identifier.	a
43	POWER/VS cancel code.	b
44	Reserved.	b
45-47	Printer device address, or SNA.	a
48	FROM remote ID.	b
49	TO remote ID.	b
50	Printed output class.	a
51	Printed output priority number.	a
52-55	Number of lines printed.	b
56-57	Number of tracks for output storage. ¹	b
58	Job suffix number assigned by POWER/VS. ⁴	b
59	Number of printed copies. ²	b
60-63	Print forms identification.	a
64-67	Number of extra records printed due to PRESTART, PSETUP, separator lines, or extra copies.	b
68-69	Number of pages printed (skips to channel 1).	b
70-71	Number of extra pages printed due to PRESTART, PSETUP, separator cards, or extra copies.	b

¹ Only for spooling to disk. When spooling to tape, field is zero.

² If more than one copy is provided, the statistics are totals for all copies.

³ a: alphameric
b: binary
p: packed decimal.

⁴ Will be X'FF' for last or only segment of job.

Figure 3-7. List Account Record

PUNCH ACCOUNT RECORD

A punch account record is created for each punch queue entry that is processed by a punch task. It is a copy of the first 68 bytes of the corresponding queue record.

Bytes	Description	Format ³
00-07	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	Start time of punch, in packed decimal (OHHMMSSP; F = sign).	p
12-15	Stop time of punch, (OHHMMSSP; F = sign).	p
16-31	16 bytes of user information from * \$\$ JOB card.	a
32-39	POWER/VS job name from * \$\$ JOB card.	a
40-41	Job number assigned by POWER/VS.	b
42	Record identifier.	a
43	POWER/VS cancel code.	b
44	Reserved.	b
45-47	Punch device address, or SNA.	a
48	FROM remote ID.	b
49	TO remote ID.	b
50	Punched output class.	a
51	Punched output priority number.	a
52-55	Number of records punched.	b
56-57	Number of tracks for output storage. ¹	b
58	Job suffix number assigned by POWER/VS. ⁴	b
59	Number of punched copies. ²	b
60-63	Punch forms identification.	a
64-67	Number of additional cards punched due to restart, separator cards, or extra copies.	a

¹ Only for spooling to disk. When spooling to tape, field is zero.
² If more than one copy is provided, the statistics are totals for all copies.
³ a: alphanumeric
b: binary
p: packed decimal.
⁴ Will be X'FF' for last or only segment of job.

Figure 3-8. Punch Account Record

EXECUTION ACCOUNT RECORD

An execution account record is created for each queue set (user job step) that has been processed by POWER/VS.

Bytes	Label	Description	Format ²
00-07	AEDY	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	AEST	Start time of job (OHHMMSSP; F = sign).	p
12-15	AEST	Stop time of job (OHHMMSSP; F = sign).	p
16-31	AEUI	16 bytes of user information.	a
32-39	AENM	Current POWER/VS job name.	a
40-41	AENO	Job number assigned by POWER/VS.	b
42	AERI	Record identifier.	a
43	AECN	POWER/VS cancel code.	b
44	AERJ	Reserved.	b
45-47	AECU	Reserved.	b
48	AEPJ	FROM remote ID.	b
49	AETJ	TO remote ID.	b
50	AECL	Class.	a
51	AEPY	Priority.	b
52-55	-	Number of lines spooled.	b
56-59	-	Number of cards spooled.	b
60-61	-	Number of pages spooled.	b
62-63	-	Length of SIO table.	b
64-65	-	Length of total account record.	b
66-71	-	Reserved.	
72-79	-	DOS/VS job name from // JOB card.	a
80-95	-	16 bytes user information from // JOB card.	a
96-97	-	Partition ID in EBCDIC format.	a
98	-	DOS/VS cancel code.	b
99	-	Type of record; S =job step, L =last step.	a
100-103	-	Reserved.	
104-111	-	Phase name, taken from // EXEC card.	a
112-115	-	End addr. of active program phase, COMREG.	b
116-119	-	CPU time elapsed in a job step; counted in 300ths of a second.	b
120-123	-	Overhead time; elapsed time not charged to any partition, in 300ths of a second.	b
124-127	-	All-bound time; system wait state time divided between running partitions, in 300ths of a second.	b
128-	-	SIO tables: 6 bytes for each device speci- fied by SYSGEN options, as follows: two bytes for device address (0cuu), four bytes for count of SIOs in current job step. ¹	b
128+m ⁽³⁾		Overflow byte: normally X'20', but X'30' if more devices are used within a partition than specified by SYSGEN options.	b
128+m+1 ⁽⁴⁾		User account information (provided via user PUTACCT macro).	

¹ POWER/VS will update the SIO tables in the execution account record with the number of I/Os it has intercepted for spooling purposes.
² a: alphameric, b: binary, p: packed decimal.
⁽³⁾ m = Total length of SIO tables.
⁽⁴⁾ Maximum length of execution account record = 2008 bytes.

Figure 3-9. Execution Account Record

RJE ACCOUNT RECORDS

A line account record is created for each RJE,BSC user session when signoff or line stop is processed. It is a copy of the first 56 bytes of the Line Control Block.

Bytes	Description	Format ¹
00-07	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	SIGNON time in packed decimal (OMMMSSP; F = sign).	p
12-15	SIGNOFF time (OMMMSSP; F = sign).	p
16-31	16 bytes user information.	a
32-39	Line password.	a
40-41	Number of invalid responses during transmission (4).	b
42	Record identifier.	a
43	SIGNOFF code X'01' - Normal SIGNOFF X'02' - SIGNOFF forced due to central stop X'04' - SIGNOFF forced due to excessive idle time X'08' - SIGNOFF forced due to unrecoverable I/O error.	b
44	Terminal error count.	b
45-47	Line address.	a
48	FROM remote ID.	b
49	TO remote ID.	b
50-51	Transmission count per session (1).	b
52-53	Timeout count per session (2).	b
54-55	Error count per session (3).	b
¹ a: alphameric b: binary p: packed decimal.		

Figure 3-10. RJE,BSC Line Account Record

Comparing (1) to (2) gives an indication of idle time per session. Comparing (1), (3), and (4) gives an indication of line quality. (1), (2), and (3) are also printed locally at SIGNOFF.

A session account record is created when an RJE,SNA user session is terminated. It is a copy of bytes 32-79 of the SNA unit control block.

Bytes	Description	Format ¹
00-07	Date in format specified at SYSGEN (mm/dd/yy or dd/mm/yy).	a
08-11	SIGNON time (OHHMMSSP; P = sign).	p
12-15	SIGNOFF time (OHHMMSSP; P = sign).	p
16-31	16 bytes of user information.	a
32-39	Logical unit name.	a
40-41	Number of invalid responses during transmission.	b
42	SNA record identifier.	a
43	Session termination code X'01' - normal termination (LOGOFF or SIGNOFF) X'02' - abnormal termination.	b
44-47	Remote identifier.	b
¹ a: alphameric b: binary p: packed decimal.		

Figure 3-11. RJE,SNA Session Account Record

PUTACCT MACRO

The PUTACCT macro adds user account information to the end of the execution account record. The macro is included in the user-written \$JOBACCT phase that is called at the end of each job or job step.

The exact location of the field for the user account information can vary, and is dependent on the length of the SIO tables field that directly precedes it.

Name	Operation	Operands	Col
			72
[name]	PUTACCT	(reg1), (reg2)	

Programming Notes:

(reg1) must specify the address of the area containing the user account information. Register notation must be used. Any general register except registers 0, 1, 14, and 15 may be specified.

(reg2) must contain the length of the area. Register notation must be used. Any general register except registers 0, 1, 14, and 15 may be specified. The maximum length may not exceed 2008 bytes minus 8 bytes for the control field minus the length of the execution account record.

Note: If POWER/VS does not support job accounting or if the partition is not running under control of POWER/VS, the PUTACCT macro is ignored.

In the following, an example routine is shown to insert user information in the POWER/VS execution account records:

```

      .
      .
      .
      COMRG          GET PARTITION COMMUNICATION REGION
      USING CMRG,R1  DECLARE ADDRESSABILITY
      TM POWFLG1,X'80' ACCOUNT SUPPORT FOR THIS PARTITION
      BNO EXIT       BRANCH IF NOT
      LA R1,ADAC     ADDRESS ADDITIONAL INFO
      LA R0,L'ADAC   LENGTH ADDITIONAL INFO
      PUTACCT (R1),(R0) PASS INFO TO POWER/VS
EXIT   DS OH
      BR RE          RETURN TO $JOBCTLN
      .
      .
      .
ADAC   DC C'ADDITIONAL ACCOUNT INFORMATION'
R1     EQU 1        REGISTER 1
R0     EQU 0        REGISTER 0
RE     EQU 14       REGISTER 14
      .
      .
      .
CMRG   DSECT
      DS CL164
POWFLG1 EQU *
      END

```


Output Segmentation

SEGMENT MACRO

The SEGMENT macro allows control of output segmentation and the specification of new output parameters to POWER/VS for a job running in a POWER/VS-controlled partition.

Name	Operation	Operands	Col
			72
[name]	SEGMENT	DEVADDR=SYSxxx	[*]
		[,FORMS=yyyy]	[*]
		[,JECL={addr} {(reg)}]	[*]
		[,NAME=name]	

Programming Notes:

- DEVADDR=** specifies the system or programmer logical unit assigned to the device on which the segmentation is to occur. This may be SYSLST, SYSPCH, or SYS000 - SYSmax. This operand is required.
- FORMS=** specifies the new 1- to 4-character forms identification to be supplied to POWER/VS. The forms identification must be alphameric, a slash (/), a hyphen (-), or a period (.). If this operand (as well as the JECL operand) is omitted, the forms identification will be blank.
- JECL=** specifies, in place of the forms identification, the address of a 71-byte save area in which a POWER/VS * \$\$ LST, PUN or JOB statement is stored. The LST or PUN parameters of the * \$\$ LST or PUN statements need only be coded if the default values are not sufficient. For details, refer to "POWER/VS Job Entry Control Language Statements" in this chapter. No continuation statements are allowed. Any valid POWER/VS parameters may be placed in this JECL area. This operand may be specified as a symbolic address or in register notation (registers 2-12).
- NAME=** specifies a 1- to 8-character name which will be given to the new segment. If this operand is omitted, the name of the segment active on the task will be used.

Note: If JECL JOB statements are to be passed via the JECL operand, the NAME operand should be omitted. The SEGMENT macro causes segmentation to occur and establishes new values for a new segment. If a JECL JOB statement is passed, it will rename the current segment and should be passed after a JECL LST or PUN statement is passed. JECL JOB statements should only be used to provide new user information.

If the output device is not being spooled by POWER/VS, a message will be issued on SYSLOG so that the operator can take the required action.

If output segmentation is requested for output directed to a 3800 printer, be aware that POWER/VS uses the default printer setup for the new segment unless you have supplied a JECL LST (or PRT) statement defining the same printer setup. After issuing the SEGMENT macro you can also issue in your application program a SETPRT macro requesting the proper printer setup.

Example of SEGMENT macro use:

```

      .
      .
      .
      LA    R2,LSTCARD
      SEGMENT DEVADDR=SYSLST,JECL=(R2),          *
            NAME=TESTOUT
      .
      .
LSTCARD DC    CL71'* $$ LST FNO=ACB,DISP=H,PRI=1'

```

RETURN CODES

Successful completion of the SEGMENT macro is indicated to the problem program by a return code of 0 in register 15. If the operation fails, register 15 contains one of the return codes listed below.

Return Code		Meaning
Dec	Hex	
4	X'04'	One of the following: <ul style="list-style-type: none"> • The device addressed by the DEVADDR operand is not a device currently being spooled by POWER/VS. • POWER/VS is not active. • The partition in which your program is running is not supported by POWER/VS. • The spooled device has a disposition of N. • The JECL statement provided was not a * \$\$ JOB, LST, or PUN statement.
8	X'08'	POWER/VS cannot accept the JECL statement because the partition was not started as a multitasking partition and the partition is waiting for work, or the partition was started as a multitasking partition and is waiting for work, but no JECL statement was submitted for this device.

POWER/VS AUTOSTART Statements

POWER/VS offers an automatic start-up facility, called AUTOSTART, that facilitates the POWER/VS initiation, especially when POWER/VS must be terminated and restarted frequently and with identical spooling device assignments.

The following four control statements are available to automatically initiate POWER/VS:

- The FORMAT statement
- The READER statement
- The PRINTERS statement
- The PUNCHES statement.

These statements, which may be entered from a diskette file, via a cataloged procedure, or in cards, and which must be submitted to the system via the SYSIN device assigned to the POWER/VS partition, are described in the following sections.

If an error is detected in an AUTOSTART statement (card deck, diskette file, or cataloged procedure), POWER/VS expects the corrected statement and all subsequent start-up statements to be entered from the console. The remaining statements in the card deck, the diskette file, or in the cataloged procedure are ignored.

Note: The POWER/VS AUTOSTART statements start always in card column 1.

STATEMENT DESCRIPTIONS

FORMAT Statement

By means of the FORMAT statement, you specify whether you want to perform a warm start, or, if not, which of the POWER/VS files are to be formatted. You use this statement, which must immediately follow the // EXEC POWER job control statement, in a way similar to the response you would provide for message 1Q11D if POWER/VS AUTOSTART were not available.

FORMAT=	{	NO	}
		Q	}
		D	}
		A	}
		combination	}

- | | |
|----|--|
| NO | indicates that no POWER/VS file is to be formatted. This is a POWER/VS warm start. |
| Q | indicates that the POWER/VS queue file is to be formatted. |
| D | indicates that the POWER/VS queue and data files are to be formatted. |
| A | indicates that the POWER/VS account file is to be formatted. |

combination indicates that two of the parameters Q, D, and A may be specified. The parameters can be specified in any sequence; they must be separated by commas. For example, to indicate that all three files are to be formatted, specify FORMAT=A,D.

One blank must follow the parameter (or the parameter sequence) specified in this statement. Only one FORMAT statement may be issued for the POWER/VS AUTOSTART procedure.

READER Statement

Using the READER statement, you define the input spooling device of the POWER/VS-controlled partition identified in the immediately preceding PSTART command. The value to be specified for this statement is similar to the response you would provide for message 1R50D if POWER/VS AUTOSTART were not available.

```
READER={NO  
        CUU}
```

NO indicates that no input spooling device is to be assigned for this partition. For POWER/VS, this means that the partition being started is a writer-only partition.

CUU specifies the channel and unit address of the partition's input spooling device. Only one input spooling device may be specified per partition; however, you may specify the same device for several partitions.

PRINTERS Statement

By means of the PRINTERS statement, you define the list (print) output spooling devices of the POWER/VS-controlled partition identified in the preceding PSTART command. The value to be specified for this statement is similar to the response you would provide for message 1R50D if POWER/VS AUTOSTART were not available.

```
PRINTERS={NO  
          CUU1,...,CUU8}
```

NO indicates that no list (print) output spooling device is to be assigned for this partition. If PRINTERS=NO and PUNCHES=NO (see below) is specified, this means that the partition being started is a reader-only partition.

```
{CUU1  
.  
.  
.  
CUU8}
```

specifies the channel and unit addresses of up to eight list (print) output spooling devices in this partition. If more than one device is to be specified, any two "CUU" definitions must be separated by a comma. The same device or devices may be specified in several partitions.

PUNCHES Statement

Using the PUNCHES statement, you define the punch output spooling devices of the POWER/VS-controlled partition identified in the preceding PSTART command. The value to be specified for this statement is similar to the response you would provide for message 1R50D if POWER/VS AUTOSTART were not available.

```
PUNCHES={NO  
          {cuu1,...,cuu8}}
```

NO indicates that no punch output spooling device is to be assigned for this partition. If PUNCHES=NO and PRINTERS=NO (see above) is specified, this means that the partition being started is a reader-only partition.

```
{cuu1  
.  
.  
.  
cuu8}
```

specifies the channel and unit addresses of up to eight punch output spooling devices in this partition. If more than one device is to be specified, any two "cuu" definitions must be separated by a comma. The same device or devices may be specified in several partitions.

An example of how to use the POWER/VS AUTOSTART statements is contained in "Initiating POWER/VS with AUTOSTART" in Chapter 4 of this manual.

Chapter 4. Operating a System Controlled by POWER/VS

There are two ways of submitting jobs to POWER/VS and of retrieving them:

- By starting the POWER/VS partition explicitly as in the procedure described below.
- By using the PUTSPOOL macro as described in Chapter 3.

The POWER/VS cross-partition facility requires no commands (for example, PSTART or PSTOP) to be given. Spool management tasks are created and deleted dynamically by POWER/VS.

Assumptions

The following sample procedure for initiating POWER/VS and running jobs under the control of POWER/VS is based on the assumption that POWER/VS resides in partition P1 and that the following I/O devices are available for spooling input and output:

IBM 2540 Card Reader with device address X'00C'

IBM 2540 Card Punch with device address X'00D'

IBM 1403 Printer with device address X'00E'

IBM 3203 Printer with device address X'01E'

| IBM 3800 Printer with device address X'118'

IBM 3340 Direct Access Storage with device address X'260'

IBM 3330 Disk Storage with device address X'133'

IBM 2780 Data Transmission Terminal with device address X'060'.

The direct access storage is used for the POWER/VS files; the data transmission terminal for remote job entry. The example also assumes that POWER/VS will support BG, P2, and P3.

| Initiating POWER/VS

POWER/VS can be initiated at any time. POWER/VS is initiated in a DOS/VS system like any program. In addition, the AUTOSTART facility is available, which allows you to start POWER/VS with a minimum of intervention. The facility is particularly useful if POWER/VS must be initiated frequently with unchanged device addresses, tasks, spooled partitions, and RJE lines. For details concerning the POWER/VS AUTOSTART facility, consult the section "POWER/VS AUTOSTART Statements" in Chapter 3.

The start-up procedure (with and without AUTOSTART) includes the following steps:

- Formatting of POWER/VS queues. (Not necessary when the information already accumulated is to be used. This procedure is called a warm start.) If the extents used to define the POWER/VS files are changed between one execution of POWER/VS and another, the POWER/VS files should be reformatted. These files should also be reformatted if a new generation of POWER/VS was link-edited since the files were used last.
- Verifying that the disk packs used for the POWER/VS files are mounted on the proper disk drives and that these drives have been made ready.
- Starting POWER/VS tasks.
- Starting POWER/VS controlled partitions.
- Specifying the devices to be spooled in each controlled partition.

INITIATING POWER/VS WITHOUT AUTOSTART

The following is a description of the steps required to start POWER/VS when the AUTOSTART facility cannot be used:

Step_1

Make the reader, the printer, and the punch available for POWER/VS use by unassigning these devices in the partition that owns them; for example:

```
ASSGN SYSIN,UA
ASSGN SYSLST,UA
ASSGN SYSPCH,UA
ASSGN SYSnnn,UA      (if the device was used as a programmer logical
                      unit).
```

You may also enter the sequence

```
DVCDN X'cuu'
DVCUP X'cuu'
```

for each of the affected devices.

If the partition into which POWER/VS is to be loaded is still active, stop it by issuing the STOP command.

Step_2

Allocate sufficient real and virtual storage to the POWER/VS partition. If the supervisor of your system was generated with the required real and virtual storage allocations, this step may be omitted. Your system programmer will tell you how much storage you must allocate. A typical example:

```
ALLOCR F1R=30K
ALLOCR F1=196K
```

Note: Before you can allocate the required real and virtual storage space for the POWER/VS partition, you must stop all partitions in your DOS/VS system by means of the DOS/VS command UNBATCH partition. After successful storage allocation, you may start the DOS/VS partitions again.

Step 3

Start the POWER/VS partition using the START command. For example, enter:

```
START P1
```

Step 4

You can omit this step if you intend to enter the POWER/VS initiation statements via SYSLOG (see step 6). Otherwise, assign SYSRDR to the device from which the POWER/VS initiation statements are to be read.

Step 5

Assign SYSLST if your installation requires

- A status report after a warm start.
- A dump of the POWER/VS partition, should POWER/VS terminate abnormally during initiation.

See Figure 4-2 for an example of a status report after a warm start.

Step 6

Place the POWER/VS initiation statements into SYSRDR assigned in step 4. An example of these statements, which vary from one installation to another, is given in Figure 4-1.

```
// JOB POWER/VS INITIATION
// ASSGN SYS000,X'260'
// DLBL IJAFIL,'POWER/VS ACCTFILE',99/365,DA
// EXTENT SYS000,POWER1,1,0,12,60
// ASSGN SYS001,X'260'
// DLBL IJQFILE,'QFILE POWER/VS',99/365,DA
// EXTENT SYS001,POWER1,1,0,72,48
// ASSGN SYS002,X'133'
// ASSGN SYS003,X'134'
// DLBL IJDFIL,'DATAFIL POWER/VS',99/365,DA
// EXTENT SYS002,POWER1,1,0,95,855
// EXTENT SYS003,POWER2,1,0,3800,1045
// EXEC POWER (see note 3 below)
```

Figure 4-1. Job Stream Example for POWER/VS Initiation

Notes:

1. DLBL and EXTENT statements may have been stored on the label cylinder(s) of your system pack. In this case, they need not be provided as part of the initiation job stream.
2. SYS000, SYS001, and SYS002 may be assigned to different device types.
3. If your POWER/VS system includes RJE,SNA support, or if you are using an IBM 3800 printer, you must specify the SIZE= parameter in the // EXEC statement; for example:
// EXEC POWER,SIZE=234K.

POWER/VS STATUS REPORT		04/30/77
TOTAL NUMBER OF QUEUE RECORDS		1178 RECORDS
NUMBER OF FREE QUEUE RECORDS		1079 RECORDS
MAX. NO. OF QUEUE RECORDS USED IN PRESENT SESSION		118 RECORDS
DATA FILE IJDFILE		
TOTAL NUMBER OF TRACKS		1178 TRACKS
TRACK GROUP SIZE		1 TRACK
DATA BLOCK SIZE		952 BYTES
ACCOUNT FILE IJAFILE		
TOTAL NUMBER OF TRACKS		19 TRACKS
PERCENTAGE OF FILE THAT IS FILLED		14 %
REAL STORAGE ALLOCATED TO POWER/VS PARTITION		11 PAGES
NUMBER OF TIMES WAITING FOR REAL STORAGE		0 TIMES
MAXIMUM NUMBER OF REAL PAGES FIXED		9 PAGES
MAXIMUM NUMBER OF TASKS ACTIVE AT ONE POINT IN TIME		13 TASKS

Figure 4-2. Example of a POWER/VS Status Report

Notes:

1. If POWER/VS has been generated without the job accounting support, the account file statistics are replaced by the message
NO ACCOUNT SUPPORT.
2. Statistics provided by POWER/VS following a warm start supply information about activities from the time POWER/VS was started until the time the statistics are displayed. The maximum number of queue records used in a report after warm start gives the number of active entries.
3. Statistics provided by POWER/VS following a PEND command give information about the ended POWER/VS session.

Up to five extents can be specified if additional extents are needed for the POWER/VS data file. They can be either on the same volume, or if they are on different volumes, logical unit assignments must be in ascending sequence of volumes to be used. SYS002-SYS006 can be used for the POWER/VS data file; they must be assigned to the same type of device.

Step 7

When the message

```
1Q11D FORMAT QUEUES=
```

is displayed, your response can be one of the following:

Q[,A] = Format the queue file [and, optionally, the account file].

D[,A] = Format the queue file and the data files [and, optionally, the account file].

A = Format the account file.

NO = No formatting is required (warm start).

Note: Instead of entering NO, you may simply press END/ENTER.

If you perform a warm start, and if you assigned SYSLST in step 5, POWER/VS prints a status report (see Figure 4-2 for an example).

In case of an abnormal warm start (warm start after an abnormal termination of POWER/VS), the following job number will be one higher than expected to prevent duplicate job numbers for accounting information.

When initiation is complete, the message

```
1Q12I POWER/VS INITIATION COMPLETED
```

is displayed. SYSRDR, SYSIPT, SYSLST, and SYSPCH are then unassigned in the POWER/VS partition.

Step 8

Use the PSTART command to start POWER/VS tasks. For example, the statement:

```
PSTART RDR,00C,A
```

causes a POWER/VS reader task to spool input from 00C and assign class A to all reader queue entries from that reader, unless another class is specified in the JECL for an entry.

The statement:

```
PSTART F2,2A
```

places a partition under control of POWER/VS. Note that, for this purpose, F2 must have lower priority than the POWER/VS partition. How to start POWER/VS tasks, partitions and RJE lines that are to operate under control of POWER/VS, is described in more detail later in this procedure.

When you enter the PSTART command to bring a partition under control of POWER/VS, you are prompted to specify the devices that are to be spooled for that partition. The prompting messages are (for the above example):

```
1Q86I PLEASE SPECIFY DEVICES TO BE SPOOLED
1R50D F2 READER=
1R50D F2 PRINTERS=
1R50D F2 PUNCHES=
```

SYSLOG stops after each message to allow you to specify a device address (or NO if a specific device type is not used) as shown in the example below:

```
1R50D F2 READER=00C
1R50D F2 PRINTERS=00E,01E
1R50D F2 PUNCHES=NO
```

Note: If your response is READER=NO, but you specify one or more addresses for PRINTERS or for PUNCHES or for both, POWER/VS operates as a writer-only spooling program for the partition concerned. POWER/VS spools the input only if your response is READER=CUU, PRINTERS=NO and PUNCHES=NO. A device can be specified only once, even if it can both read and write.

Step 9

After a partition has been brought under control of POWER/VS, I/O requests from that partition to the specified devices are intercepted by POWER/VS. If the reader queue does not contain an entry for processing in this particular partition, POWER/VS displays the message:

```
1Q34I F2 WAITING FOR WORK
```

INITIATING POWER/VS WITH AUTOSTART

The following are examples for the steps required to start POWER/VS using the AUTOSTART facility:

Step 1

Start the POWER/VS partition, using the START command. You may have to stop the partition before you can issue the START command.

Note: The initiation statements, and also the AUTOSTART cards, may be supplied via a SYSIN file on disk, tape, or diskette, or via a card reader, but the // EXEC POWER statement should not be supplied via a procedure. If this is done, the DOS/VS procedure library is in use during the entire POWER/VS run and may not be updated until POWER/VS is ended.

Step 2

Assign SYSIN for the POWER/VS partition to the device from which the POWER/VS initiation statements and the POWER/VS AUTOSTART statements are to be read.

Step 3

Place the required statements in the SYSIN device assigned in step 2 and make the device ready. You may also supply the initiation and AUTOSTART statements via a cataloged procedure. An example of the AUTOSTART statements that may be presented to POWER/VS immediately following the // EXEC POWER statement is given below:

```

FORMAT=NO
PSTART RDR,00C
PSTART BG,ABC
READER=00C
PRINTERS=00E
PUNCHES=00D
PSTART F2,DEF
READER=00C
PRINTERS=01E
PUNCHES=00D
PSTART LST,00E
PSTART LST,01E
PSTART PUN,00D
PSTART RJE,060
/*

```

The FORMAT statement must be the first one after the // EXEC statement. If the FORMAT statement is omitted, POWER/VS prompts you for acceptable format specifications in the same way as it does when you start POWER/VS without AUTOSTART.

The statements that start a system task (partition) can be entered in any sequence. However, once a partition has been started, the device specifications for that partition must follow immediately and in the sequence:

1. READER= statement
2. PRINTERS= statement
3. PUNCHES= statement.

After POWER/VS has processed the FORMAT statement and read the remaining statements from SYSIPT, it unassigns SYSRDR, SYSIPT, SYSPCH, and SYSLST for the POWER/VS-controlled partitions and processes the PSTART commands that were submitted via SYSIPT. If any of the PSTART commands are invalid, POWER/VS prompts you to specify the correct device address.

Job streams to be processed under control of POWER/VS can immediately follow the POWER/VS AUTOSTART statements.

Spooling device specifications can be made as follows:

```

READER={NO
        {cuu}
}

PRINTERS={NO
          {cuu1,cuu2,...,cuu8}
}

PUNCHES={NO
          {cuu1,cuu2,...,cuu8}
}

```

If the spooling device specifications in the AUTOSTART statements are incorrect or incomplete, POWER/VS prompts you for acceptable specifications in the same way as it does when you start POWER/VS without AUTOSTART.

ASSIGNING DUMMY DEVICES

POWER/VS intercepts I/O requests addressed to specific physical devices, regardless of the symbolic units that are assigned to these physical devices. If I/O requests are intercepted by POWER/VS, the assignment for the physical unit is in fact a dummy assignment, because the physical device is not used by the problem program. With POWER/VS you

can assign logical units in different partitions to the same physical unit-record device. Such assignments are regarded as dummy assignments, since the assigned physical device is not used by all the partitions in which it is assigned. Dummy devices, however, are not required, except for multifunction card devices. The use of writer-only partitions as well as cardless systems normally require dummy devices.

Each ASSGN statement or command in a POWER/VS-controlled partition is checked by job control to determine if I/O requests for the specified logical unit are to be intercepted by POWER/VS. If requests for a certain physical unit are to be intercepted by POWER/VS, job control will not check for conflicting I/O assignments. As a result two or more assignments are permitted from different partitions to the same unit-record device, as long as no more than one of these assignments implies physical ownership of the device.

CHANGING PARTITION PRIORITIES

The priority of the POWER/VS partition must always be higher than that of the partitions under its control.

If you want to change the priorities of the partitions while POWER/VS is active, the DOS/VS PRTY command is rejected if you make an attempt to give one of the partitions supported by POWER/VS a higher priority than the POWER/VS partition.

POWER/VS initialization is canceled if the priorities of the partitions conflict with the POWER/VS requirements.

Issuing POWER/VS Commands

POWER/VS commands can be entered in one of three ways:

- At the system console by the central operator
- At a remote terminal or work station by the terminal operator
- Within an application program by means of the CTLSPool macro.

At the system console, POWER/VS operator commands are entered in the same way as attention routine commands (omit steps 1 and 2 if your system operates on a Model 115 or 125):

1. Press REQUEST
2. Wait for AR to appear
3. Enter the command
4. Press END or ENTER.

Most of the POWER/VS commands available to the central operator are either queue management or task management commands. Queue management commands enable you to manipulate the queue entries for the POWER/VS jobs; task management commands allow you to control POWER/VS operation. Figure 4-3 shows the two groups of commands by the function they control.

<u>Type of Command</u>	<u>Command Code</u>	<u>Short Form</u>	<u>Function</u>
Queue Management Commands	PDISPLAY	D	Display the status of POWER/VS job(s).
	PALTER	A	Alter the processing attributes of a POWER/VS job.
	PDELETE	L	Delete queue entries or messages.
	PRELEASE	R	Release POWER/VS jobs for further processing.
Task Management Commands	PSTART	S	Start a task, place a partition under the control of POWER/VS, or establish the POWER/VS-VTAM interface (RJE,SNA).
	PSTOP	P	Stop a task, release a partition from POWER/VS control, terminate an SNA user session, or terminate RJE,SNA processing.
	PGO	G	Reactivate a task or a partition.
	PEND		Terminate POWER/VS.
	PCANCEL	C	Cancel a POWER/VS status report.
	PFLUSH	F	Flush (cancel processing of) the current queue entry.
	PRESTART	T	Restart a writer task.

Figure 4-3. POWER/VS Commands

Whenever you enter a queue management command related to a set of queue entries for a particular POWER/VS job, make sure that your command correctly identifies the job by the name under which it was logged by POWER/VS and, possibly, also by the number that POWER/VS assigned to it. A number is required when two or more POWER/VS jobs have been entered under identical names.

If you are in doubt about the correct identification of a POWER/VS job, enter a PDISPLAY command specifying either the queue and the job name or the queue and ALL before you enter the queue management command in question. The status report you receive provides you with a list of POWER/VS jobs currently logged in a specific queue, and under a specific name if you specified a job name.

For the terminal operator, a set of commands similar to the JECL commands is available for entry at a remote terminal or work station. These commands, which are referred to as the Remote Operator Command Language (ROCL), are described in the POWER/VS Work Station User's Guide, GC33-6049.

To invoke a POWER/VS service from an application program executing in a partition other than the POWER/VS partition, the macro instruction CTLSPPOOL may be used with the parameter specification REQ=COMMAND. However, this macro allows only five commands to be issued. These are:

- The PALTER command
- The PCANCEL command
- The PDELETE command
- The PDISPLAY command
- The PRELEASE command.

The use of this macro and the related additional macros is explained in detail in "Cross-Partition Communication Macros" in Chapter 3.

STARTING AND STOPPING A USER PARTITION

Bringing a Partition Under the Control of POWER/VS

When you bring a partition under the control of POWER/VS, you activate the POWER/VS task that makes spooled input available for processing in the named partition.

To bring a partition under the control of POWER/VS, use the PSTART command.

For example, the statement:

```
PSTART F2,AB2
```

causes those POWER/VS jobs in the reader queue with a class specification of A, B, or 2 to be executed in partition F2.

Note: A class specification of AB3 in the above command would have been flagged as invalid because the partition-dependent class 3 does not match the partition identifier specified in the command (F2).

The statement:

```
S BG,0A,X
```

causes POWER/VS to place the BG partition under control of POWER/VS. Jobs of class '0' and 'A' may be processed by this partition. All print and punch output will have a default class of 'X'.

Changing the Class Specification for a Partition

When a partition is placed under POWER/VS control, one or more program classes can be specified to ensure that only programs with one of these classes can be executed in that partition. If it is desirable to change this class specification, proceed as follows:

1. Stop the partition (see "Stopping POWER/VS Tasks" in the section "Task Control" below).
2. Start the partition again using a PSTART command that specifies the new program classes to be handled by that partition.

Stopping and Restarting a Partition

A user may have special spooling requirements not supported by POWER/VS, for example when using the special punch features of the IBM 2560 Multifunction Card Unit. This requires special handling for a job.

In the following, a procedure is described to stop a partition under POWER/VS, start it again under control of the DOS/VS supervisor, execute the job, and bring the partition back under POWER/VS control after the job is finished.

- Stop the partition on a job boundary using PSTOP.
- Stop any POWER/VS reader or writer task that uses devices which may be required by the job to be run.
- Issue the DOS/VS START command.
- Assign the necessary devices for the partition.
- Make the devices ready and start the partition using the ENTER key.
- When the job has finished, stop the partition using the DOS/VS STOP command.
- Issue the PSTART command to start the POWER/VS partition.

TASK CONTROL

Starting a POWER/VS Reader Task

A reader task reads records from a card reader, a tape, or a diskette device and writes them onto intermediate storage on disk. You may start a reader task any time you have a physical reader available, regardless of the number of reader tasks currently active.

To start a reader task, use the PSTART command. For example, the statement:

```
PSTART RDR,00C,2
```

starts a reader task with a partition-dependent class specification. The task causes the job streams in the specified reader (00C) to be read and spooled onto disk. Unless a different class is specified in the job streams, the POWER/VS jobs spooled by this task will be executed in partition F2.

The statement:

```
PSTART RDR,00C,B
```

starts a reader task with a partition-independent class. In this case, a POWER/VS job spooled by the task is assigned class B, unless a different class is specified in the job stream of that job.

The statement:

```
PSTART RDR,00B,, 'INPUT',2
```

starts a reader task to read a file named INPUT from two diskette volumes on device 00B.

The statement:

```
PSTART RDR,480
```

starts a reader task from a tape on device 480.

Starting a POWER/VS Writer (List or Punch) Task

A list (print) or punch writer task transfers data from intermediate storage to the printer or punch, respectively. You may start a list or punch task whenever you have a physical printer or punch available, regardless of the number of writer tasks currently active.

To start a writer task, use the PSTART command. For example, the statement:

```
PSTART LST,00E,AB
```

starts a list (print) task to print output of POWER/VS jobs that had classes A and B assigned. The task uses the printer whose device address is 00E.

The statement:

```
PSTART LST,118,T,2
```

starts a list (print) task with two print buffers to print output of POWER/VS jobs that have class T assigned. The task uses the printer whose device address is 118.

The statement:

```
PSTART PUN,00D,X'285'
```

starts a punch task to punch on the device 00D the output that was spooled onto a tape. That tape is mounted on the drive with the address X'285'.

Stopping POWER/VS Tasks

The PSTOP command is available to stop a POWER/VS task, to take control of a partition away from POWER/VS, and to stop transmission of data via an RJE line.

To stop a reader task, you should specify EOJ as the second operand. If you fail to do so, the reader task terminates input processing immediately, the queue entry which was being built by the task is not included in the reader queue, and the rest of the statements will not be read. Examples of PSTOP commands to stop a reader task are given below.

The statement:

```
PSTOP 00C
```

causes the associated reader task to stop input processing immediately.

The statement:

```
PSTOP 00C,EOJ
```

causes the associated reader task to stop after it has (1) finished processing the current POWER/VS job and (2) placed the queue entry which was being built into the reader queue.

To stop a list task, you can use the PSTOP command without a second operand, or with EOJ or RESTART specified as the second operand. For example, the statement:

```
PSTOP 00E
```

causes the associated list task to stop output processing immediately, but the current queue entry is not deleted from the output queue. When the task is started again (by issuing a PSTART command), processing of the output for that queue entry starts again with the first output record.

The statement:

```
PSTOP 00E,RESTART
```

has the same immediate effect as the above example. However, when the task is started again (by PSTART), processing of the output for the queue entry starts with the record following the last one processed before the PSTOP command was issued, and when a 3800 printer was used, with the copy group index being processed when the PSTOP command was issued. For more information on copy grouping refer to the description of the * \$\$ LST statement in Chapter 3.

The statement:

```
PSTOP 00E,EOJ
```

causes the associated list task to stop after it has finished printing the output for the current list queue entry. When the task is started again (by PSTART), it processes the output of the next entry in the list queue.

POWER/VS acknowledges the PSTOP command by message 1Q33I.

The information provided above for stopping a list task also applies to stopping a punch task. Examples of PSTOP commands to stop a punch task follow:

```
PSTOP 00D
PSTOP 00D,RESTART
PSTOP 00D,EOJ
```

POWER/VS acknowledges the command by message 1Q33I.

The following procedure is recommended for using the PSTOP command to stop RJE lines, except in cases where a line must be stopped without regard for whether it is currently in the processing or inactive state:

1. Issue a PINQUIRE ALL command to determine the current status of your RJE lines.
2. Issue a PSTOP command for a line which is inactive. Specify EOJ as the second operand in your PSTOP command, in case a remote entry was started between the time processing of your PINQUIRE command was completed and processing of your PSTOP command begins.

If, for example, you issue the command

```
PSTOP 060
```

then all read and write activities are stopped immediately. If the line was being used to transmit data from the terminal, input data is lost.

If the line was being used to transmit data to the terminal, that output data is still in the output queue. If you issue the command

```
PSTOP 060,EOJ
```

then transmission of input or output is allowed to continue until the end of the current queue entry.

POWER/VS acknowledges the command by message 1R02I.

To take control of a partition away from POWER/VS, you also use the PSTOP command. For example, assuming that partition F2 was originally placed under POWER/VS control by an appropriate PSTART command, then

```
PSTOP F2
```

causes program execution in partition F2 for a particular reader queue entry to be completed and the partition to be returned to DOS/VS control.

After the partition is released by POWER/VS, SYSRDR and all assignments to spool devices for this partition are also released.

POWER/VS acknowledges the command by message 1Q33I. After display of this message, the partition is available for processing under DOS/VS.

To return the partition to POWER/VS control, you have to issue the DOS/VS STOP or UNBATCH command, followed by a PSTART command.

REMOTE JOB ENTRY - BSC TERMINALS

Starting an RJE,BSC Line

By starting an RJE,BSC line, you actually start a POWER/VS RJE task. This task processes job streams from a terminal and creates reader queue entries (one per POWER/VS job transmitted from the terminal). On request from the terminal operator, an RJE print or punch task makes output available for transmission to the terminal. POWER/VS allows you to start up to 25 RJE,BSC lines for concurrent operation.

To start an RJE,BSC line, use the PSTART command. For example, the statement:

```
PSTART RJE,060
```

causes the line with the physical device address 060 to be started. When POWER/VS has finished executing this command, a terminal operator whose terminal is attached to this line can sign on.

The statement:

```
PSTART RJE,060,RJESYS12
```

causes the same POWER/VS action as the above example. However, the SIGNON command from a terminal operator whose terminal is attached to line 060 is accepted only when he specifies the password RJESYS12 in his SIGNON command.

Note: If a password is specified in the PSTART command for an RJE,BSC line, this password overrides the password specified in the PLINE macro for this line.

Controlling RJE,BSC Operations

The following POWER/VS commands can be used to monitor and control RJE,BSC operations from the central installation:

```
PSTART      Start an RJE line (this is discussed above).
PINQUIRE   Display the status of one or all supported RJE lines.
PDISPLAY    Display the status of RJE-type queue entries.
PALTER      Change the destination of output for remotely entered
            POWER/VS jobs.
PDELETE     Delete ALLUSERS messages originated by the central operator.
PBRDCST     Send a message to an individual terminal user or to all
            terminal users.
PSTOP       Stop an RJE line.
```

REMOTE JOB ENTRY - SNA TERMINALS

Establishing the VTAM Interface for POWER/VS with RJE,SNA

Before SNA terminal users can log on to POWER/VS, you must have established the interface between POWER/VS and VTAM, as VTAM handles all physical line management for SNA terminals, using Synchronous Data Link Control (SDLC).

| VTAM must run in a partition with a higher priority than POWER/VS.

To establish the interface, use the PSTART command in the format:

```
PSTART RJE,SNA
```

This command causes POWER/VS to activate the VTAM interface by opening the Application Control Block (ACB), after which logical units that have been identified to VTAM can log on. If the command is valid, the message `↑V04I RJE,SNA STARTED` is displayed. If the command is invalid, an information-type message is displayed. Refer to DOS/VS POWER/VS Workstation User's Guide, GC33-6049 for possible causes.

Controlling RJE,SNA Operations

The following POWER/VS commands can be used to monitor and control RJE,SNA operations from the central installation:

- PSTART Establish the VTAM interface.
- PINQUIRE Display the status of one logical unit, or display the logical unit names of all active logical units (status: PROCESSING and LOGGED ON), together with the remote-IDs of the users operating them.
- PDISPLAY Display the status of RJE (BSC and SNA)-type queue entries or list all ALLUSERS-type messages.
- PALTER Change the destination, class, disposition, or copy count of output for remotely entered POWER/VS jobs.
- PDELETE Delete one or more entries from the specified job queue; alternatively delete one specific ALLUSERS-type message or delete all ALLUSERS-type messages.
- PBRDCST Send a message to an individual terminal user or to all terminal users.
- PSTOP Terminate a specific user session or deactivate the VTAM interface and thereby all current user sessions.

Stopping an RJE,SNA Session

To stop a session identified by 'luname', the following format applies:

PSTOP RJE,SNA,luname[,EOJ]

- RJE,SNA Informs POWER/VS that the command applies to an SNA session or to all SNA processing. If no more parameters are specified, POWER/VS immediately deactivates the VTAM interface.
- luname Causes POWER/VS to terminate only the session with the logical unit whose name is specified by this parameter.
- EOJ All active input or output processing will be completed for current POWER/VS job entries. If this parameter is omitted, all processing is stopped immediately for the affected session.

In case of emergency, the central operator may also terminate a session. To do this, he issues the VTAM command

VARY NET,INACT,I,ID=luname

upon which VTAM will notify POWER/VS in the LOSTERM exit. POWER/VS handles this termination in a similar way as an unconditional LOGOFF request.

Termination of All RJE,SNA Sessions

The following lists the possible central operator commands to terminate RJE,SNA operations and the actions taken by the system:

HALT QUICK	This VTAM command causes VTAM to notify POWER/VS that an emergency stop is required for all active SNA sessions.
HALT	This VTAM command causes VTAM to notify POWER/VS to begin an orderly termination of all sessions. The same action is taken as if a conditional LOGOFF terminal command had been requested for all sessions.
PSTOP RJE,SNA,EOJ or PEND	POWER/VS makes a conditional termination for all sessions.
PSTOP RJE,SNA or PEND KILL	POWER/VS makes an unconditional LOGOFF for all sessions.

TERMINATING POWER/VS

POWER/VS may be terminated normally, for example when programs that do not need a POWER/VS environment must be run, or in an emergency.

Normal Termination

Before you terminate POWER/VS, inform any operators of remote terminals that the shutdown procedure is about to begin. Then enter PEND. For example, the statement:

PEND

causes (1) all active tasks to continue processing until they encounter the end of the current POWER/VS job, (2) the POWER/VS-controlled partition to be released when program execution for the currently processed reader queue entries is finished, and (3) the POWER/VS partition to be released. Your operating system is restored to normal DOS/VS operation when message 1Q21I has been displayed.

The statement:

PEND 00E

causes the same action as in the above example. In addition, a status report is printed on the printer with the address 00E by POWER/VS before it restores your operating system to normal DOS/VS operation. This status report is similar to the one provided by POWER/VS after a warm start (see Figure 4-2).

Emergency Termination

When it is necessary to terminate POWER/VS immediately because of an emergency situation, use the PEND command with the operand KILL. This command causes POWER/VS to terminate immediately and, if a printer address is specified, to produce a system dump (printout of virtual storage) on the specified printer.

As a result of a PEND command with the KILL operand, input and output of jobs is broken off. A warm start enables you to restart output from the beginning of the interrupted job. Input of a job entry that was interrupted by the command must be resubmitted completely.

Two examples of the PEND command with the KILL operand follow. The statement:

```
PEND KILL
```

causes an emergency termination of POWER/VS without a system dump being produced.

The statement:

```
PEND KILL,00E
```

causes an emergency termination of POWER/VS with a system dump being produced on the printer with the address 00E.

Your operating system is restored to normal DOS/VS operation when message 1Q21I has been displayed.

MULTIPURPOSE UNIT-RECORD DEVICES

If your installation uses any of these devices for POWER/VS, remember that they cannot be used simultaneously for a reader and a punch task.

If it is necessary to start the reader task for one of these devices while the punch routine is still processing, you must first stop the punch task using the PSTOP command. For example, the statement:

```
PSTOP 00D,E0J
```

causes the device to stop punching on completion of the output for the punch queue entry currently being processed by the associated punch task.

The statement:

```
PSTOP 00D,RESTART
```

causes the device to stop punching output immediately.

In both cases, POWER/VS displays the message 1Q33I STOPPED PUN,00D when the punching operation has stopped. You may then start the reader task using the command

```
PSTART RDR,00D
```

After the reader task finishes reading the new input, stop it and restart the interrupted punch task. For the above example, the commands would be:

```
PSTOP 00D
```

(assuming that the last card in the read hopper has been read) and, following message 1Q33I:

```
PSTART PUN,00D
```

Miscellaneous Devices

USING THE IBM 3540 DISKETTE INPUT/OUTPUT UNIT

You can use the 3540 unit as a POWER/VS input device, either as a card reader (SYSIN mode) or as a supplement to the card reader (data mode). POWER/VS can, for example, read the job control statements from the card reader and then switch to the diskette unit to read a data file, or it can read control statements and card image data from the diskette unit, using no card reader at all.

The 3540 in SYSIN Mode

For 3540 SYSIN files, a reader task will read either 80- or 81-character records from diskette and put 80-character records onto the spool disk. The size of the records to be read is obtained from the HDR1 label on the file and must be 80 or 81 bytes. If the input records are 81 characters in length, only the last 80 bytes will be copied to the POWER/VS data file.

If an * \$\$ RDR statement is read from the diskette, POWER/VS issues a message (1Q90I INVALID * \$\$ RDR STATEMENT) and flushes to the next POWER/VS job on the diskette file currently being processed.

The SYSIN records can be read only by a user program that is reading from a card reader specified at partition start-up as a unit-record device to be spooled. Because DTFDU cannot be assigned to a card reader, DTFDU cannot be used to access these files.

Example: Job control statements and data are both on one 3540 SYSIN file. The operator enters a PSTART command to start a reader task on a 3540 diskette (X'00B'):

```
PSTART RDR,X'00B',B,'FILE-ID',31
```

Up to 31 diskettes of the file called FILE-ID will be read. Reading stops after 31 diskettes or after reading a diskette that does not have a continuation indicator in its label. One 3540 file may contain many DOS/VS jobs and/or POWER/VS jobs. Jobs with no class specification in their * \$\$ JOB card, or jobs for which no * \$\$ CTL statement is in effect, are put into class B.

During program execution, the records will only be passed to programs reading from a card reader that has been specified as the reader spool device when starting the partition.

The 3540 in Data Mode

For data files, a reader task can read records of from 1 to 128 characters. These records are not examined for control statements and are written onto the spool file exactly as read. The data records cannot be read by programs accessing a card reader. They can only be read by a user program that is reading from the physical unit specified on the * \$\$ RDR statement. This unit must be assigned to a 3540. Either DTFDU or DTFDI can be used to access these files.

Two or more files per 3540 volume may be specified through the use of two or more * \$\$ RDR statements in the card reader.

Note that although a diskette file may be inserted into a card input stream, it is not possible to insert card data into the middle of a diskette file or to insert one diskette file into another.

The * \$\$ RDR statement causes a POWER/VS task to insert information from a 3540 file into the input being read from the card reader. You do not need to submit other JECL statements for a job containing a RDR statement. This statement is ignored in a writer-only partition.

Example 1: The job control statements are in the card reader, data is on the 3540. The operator enters a PSTART command for the card reader (X'00C') and input class A:

```
PSTART RDR,00C,A,00B
```

This command informs POWER/VS to start a reader task at address X'00C' with the ability to read from a 3540 at address X'00B' also. Both input devices belong to the reader task and cannot be used physically by any other partition or POWER/VS task until the reader task terminates. The following cards are in the card reader (X'00C'):

```
// JOB EX1
// ASSGN SYS008,X'00B'
// DLBL FILE,'FILE-ID',,DU
// EXTENT SYS008
// EXTENT SYS008
// EXEC PROG
* $$ RDR DEV=X'00B',FID='FILE-ID',NOD=2
/*
/&
```

The SYS008 specification in the // EXTENT statements is not required if the symbolic unit was assembled into the DTFDU.

The RDR statement causes the reader to suspend card reading to read up to two diskettes of the data file named FILE-ID. Records on the 3540 may be from 1 to 128 bytes long and will not be examined for control statements by either the reader task or the execution processor. When the end-of-file identifier is reached, card reading is resumed.

During the execution of the user program, not all of the FILE-ID records spooled by the reader task may be read. To prevent the remainder of the records from being passed to job control as SYSIN data (once the * \$\$ RDR statement is reached), any request to the card input spool device will cause POWER/VS to skip records until the end of the FILE-ID file.

Example 2: Some job control statements are in the card reader, additional job control statements and data are on a 3540 diskette. The operator enters a PSTART command for the card reader (X'00C') and input class A:

```
PSTART RDR,00C,A,00B
```

This command causes the reader task to insert 3540 data from X'00B' into the input stream on the spool disk when an * \$\$ RDR statement is encountered in the card input stream. The following cards are in the card reader:

```
// JOB EX1
.
.
.
/&
* $$ RDR FID='TESTJOB'
// JOB EX2
.
.
.
```

The * \$\$ RDR statement causes the reader task to suspend card reading and to read one diskette of a SYSIN file named TESTJOB from the 3540 specified in the PSTART command (X'00B'). The TESTJOB file could contain the following statements, for example:

```
// JOB ASSEM
// EXEC ASSEMBLY
.
.
.
source code
.
.
.
/&
```

For more information on the use of the * \$\$ RDR statement, refer to the section "POWER/VS Job Entry Control Language Statements" in Chapter 3.

Note: POWER/VS will only spool 80- or 81-byte records from the 3540 for programs that ask for card input. A user converting from a Multifunction Card Unit to a 3540 must change his programs if 96-byte records were used.

Appendix A. Data Compaction

Compaction provides a method by which nonduplicate characters may be compressed. It takes advantage of the fact that most data streams use a relatively small subset of the 256 possible EBCDIC characters. Although there are a number of ways to do compaction, only the algorithm to be implemented for the POWER/VS SNA support is discussed.

For compaction, you must supply a compaction table to POWER/VS. The process is relatively simple, but you must know the data to be transmitted. First, determine the set of all possible characters that may appear in the data stream. Next, count the number of possible characters, then using Figure A-1, match this number with the closest (but not smaller) number appearing in the first column. The second column shows how many master characters must be selected for the corresponding number of compact code characters. If the number of compact code characters is greater than 247, you should specify no compaction.

Number of Compact Code Characters	Number of Master Characters
247	3
240	4
231	5
220	6
207	7
192	8
175	9
156	10
135	11
112	12
87	13
60	14
31	15
16	16

Figure A-1. Numbers of Master Characters Related to Numbers of Compact Code Characters

For example, to choose a master character subset for this appendix, you would define the possible characters that may occur as shown in Figure A-2.

Thus you have determined that this appendix may include up to 85 different characters. In the first column of Figure A-1, the number closest to 85 (but not smaller) is 87. Now increase the number of characters in the character set to equal the number from Figure A-1. For this appendix, you might choose:

< Less than (X'4C')
> Greater than (X'6E').

Now, from this character subset, choose the m most frequently used characters, where m is the number of master characters in Figure A-1. For a large subset of 87, m equals 13. For the m most frequent characters in this document, you might select:

a, d, e, g, i, l, n, o, r, s, t, u, and blank

Characters	Name	Hex Code
A - I		C1 - C9
J - R	Uppercase letters	D1 - D9
S - Z		E2 - E9
a - i		81 - 89
j - r	Lowercase letters	91 - 99
s - z		A2 - A9
0 - 9	Numerals	F0 - F9
␣	Blank	40
[Left bracket	AD
.	Period	4B
(Left parenthesis	4D
+	Plus sign	4E
&	Ampersand	50
]	Right bracket	BD
\$	Dollar sign	5B
*	Asterisk	5C
)	Right parenthesis	5D
;	Semicolon	5E
-	Hyphen	60
/	Slash	61
,	Comma	6B
%	Percent	6C
_	Underscore	6D
?	Question mark	6F
:	Colon	7A
#	Pound sign	7B
@	At sign	7C
'	Single quote	7D
=	Equal sign	7E
"	Double quote	7F

Figure A-2. Example of Compact Code Set

To define the compaction table for POWER/VS, you can specify the parameters in the PCPTAB macro as follows:

```

name      PCPTAB MASTER=(a,d,e,g,i,l,n,o,r,s,t,u,40)
          NOMAST1=(A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,
          T,U,V,W,X,Y,Z,b,c,f,h,j,k,m,p,q,
          v,w,x,y,z,0,1,2,3,4,5,6,7,8,9,AD,
          4B,4D,4E,50,BD,5B,5C,5D,5E,60,61,6B,
          6C,6D,6F,7A,7B,7C,7D,7E,7F,4C,6E)

```

The macro PCPTAB builds tables using these parameters. For details concerning the PCPTAB macro, refer to the section "POWER/VS Generation Macros" in Chapter 2.

To compress and compact the data stream, POWER/VS performs the following:

1. Scans the data, looking for consecutive blank character strings of length 2 or greater.
2. Scans the data, looking for consecutive duplicate character strings of length 3 or greater.

3. For each string of nonblank and nonduplicate characters, POWER/VS then checks if any character is not a member of the set of compact code characters (master or non-master characters). If so, POWER/VS creates a string control byte (SCB) for noncompressed and noncompact characters and continues with the next string.
4. If all characters are in the compact code subset, POWER/VS translates the whole string to compact code. Then POWER/VS scans the string, looking for two consecutive master characters. Each time this occurs, POWER/VS compacts the two bytes into a single byte and continues. A fourth kind of SCB describes the compacted string.

It is important that the compact code subset include all (or almost all) characters in the data stream. Otherwise, an entire string may not be compacted.

To illustrate how much reduction can be realized, operate the algorithm on the sentence for step 2 above using the example for PCPTAB specifications (as coded before). This character string contains 103 characters, assuming one blank between each word. For step 2, the following pairs of master characters are found in order:

```
| an,s ,e ,da,ta, l,oo,in,g ,or,on,se,ut,e ,du,li,at,
  e ,ar,te,r ,st,ri,ng,s ,l,en,gt, o, r,gr,ea,te.
```

```
| There are 33 pairs of master characters. Potentially, this sentence can
  be reduced from 96 characters (two SCBs added for nonduplicate character
  string with compression only) to 62. A reduction of about 35 percent.
```

A discussion follows of how POWER/VS does compression and compaction and how it builds its tables for compaction and decompaction.

The string control byte (SCB) as defined for SNA is a byte consisting of a two-bit string identifier and a six-bit count field. It is defined as follows:

SCB Bits	Explanation
-----	-----
xx000000	Reserved. Invalid if present.
00xxxxxx	Noncompressed characters. The SCB is followed by 1-63 data bytes. The number of data bytes is given by xxxxxx. The next SCB follows the last data byte.
01xxxxxx	Compact code characters. The SCB is followed by 1-63 bytes. Each byte may represent two consecutive master characters, or a single character in the compact code subset; and both kinds of bytes may exist in the same string. The number of bytes is given by xxxxxx; and it represents the number of bytes following the SCB, not the number of data bytes represented by the string (before compaction). The next SCB follows the last byte.
10xxxxxx	Duplicate blanks. The SCB represents 1-63 consecutive blanks (X'40'). The number of blanks represented is given by xxxxxx. The next SCB follows this SCB immediately.
11xxxxxx	Duplicate next character. The SCB and the character immediately following the SCB represent 1-63 consecutive duplicate characters. The duplicated character is the character following the SCB. The number of duplicate characters is given by xxxxxx. The next SCB follows the duplicated character.

If compression is being done but not compaction, SCBs of the form B'01xxxxxx' are not valid.

There are two important rules that must be followed for SCBs:

- An SCB and its string may not span request unit (RU) boundaries
- Compression and compaction operate on byte strings.

There is not necessarily any relationship between the boundaries of an SCB string and the boundaries of a logical record. A logical record may begin or end in the middle of an SCB string. In the same manner, single-character string (SCS) control characters may be compressed as well as data.

The PCPTAB macro builds a set of compaction tables consisting of 256-byte tables - a translate-and-test table for compaction (TRTC), and a translate table for compaction (XLTC), - and an FMH3 (function management header) ready for transmission to the work station.

The PCPTAB macro builds the TRTC so that it contains zeros at the displacements corresponding to each member of the compact code subset, and nonzero values at all other displacements. The XLTC table contains meaningful entries only at displacements corresponding to the compact code subset members. The values at those displacements are such that no compact code character will translate into a byte whose high-order digit and low-order digit are less than m , where m is the number of master characters.

The value for the first master character is always X'F0'. The second master character gets X'F1', and so forth, until the master character subset is exhausted. If the number of master characters is 16, there are no non-master characters, and the XLTC is then complete. If m is less than 16, the first non-master character gets X'Fm+1', and so forth, up to X'FF'.

If m is less than 15, the next 16 non-master characters are assigned X'EO' - X'EP'. If m is less than 14, the next 16 non-masters are assigned X'DO' - X'DF', and so forth, until the point is reached such that if X'x0' were assigned to the next non-master, both digits would be less than m . When that point is reached, the next non-master gets X'yz', where $y=m-1$ and $z=m$. The following non-master gets X'yz+1', and so forth, up to X'yF'. The next non-masters get X'y-1z', X'y-1z+1', up to X'y-1F'. Each time X'yF' is reached, the high-order digit is reduced by 1, until the last non-master gets X'0F'.

Using the example from this appendix, the tables generated with the PCPTAB macro are as shown in Figures A-3 and A-4. For the TRTC table, a dash represents any nonzero value. For the XLTC table, a dash represents any value, because such a character will not occur in the string.

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4x	00	-	-	-	-	-	-	-	-	00	00	00	00	00	-	-
5x	00	-	-	-	-	-	-	-	-	00	00	00	00	00	-	-
6x	00	00	-	-	-	-	-	-	-	-	00	00	00	00	00	-
7x	-	-	-	-	-	-	-	-	-	00	00	00	00	00	00	-
8x	-	00	00	00	00	00	00	00	00	00	-	-	-	-	-	-
9x	-	00	00	00	00	00	00	00	00	00	-	-	-	-	-	-
Ax	-	-	00	00	00	00	00	00	00	00	-	-	-	-	-	-
Bx	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cx	-	00	00	00	00	00	00	00	00	00	-	-	-	-	-	-
Dx	-	00	00	00	00	00	00	00	00	00	-	-	-	-	-	-
Ex	-	-	00	00	00	00	00	00	00	00	-	-	-	-	-	-
Fx	00	00	00	00	00	00	00	00	00	00	-	-	-	-	-	-

Figure A-3. Translate-and-Test Table for Compaction (TRTC)

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4x	FC	-	-	-	-	-	-	-	-	-	6D	6E	FD	6F	5D	-
5x	5E	-	-	-	-	-	-	-	-	-	5F	4D	4E	4F	3D	-
6x	3E	3F	-	-	-	-	-	-	-	-	-	2D	2E	2F	FE	1D
7x	-	-	-	-	-	-	-	-	-	-	1E	1F	FF	0D	0E	0F
8x	-	F0	E0	E1	F1	F2	E2	F3	E3	F4	-	-	-	-	-	-
9x	-	E4	E5	F5	E6	F6	F7	E7	E8	F8	-	-	-	-	-	-
Ax	-	-	F9	FA	FB	E9	EA	EB	EC	ED	-	-	-	-	-	-
Bx	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cx	-	EE	EF	D0	D1	D2	D3	D4	D5	D6	-	-	-	-	-	-
Dx	-	D7	D8	D9	DA	DB	DC	DD	DE	DF	-	-	-	-	-	-
Ex	-	-	CD	CE	CF	BD	BE	BF	AD	AE	-	-	-	-	-	-
Fx	AF	9D	9E	9F	8D	8E	8F	7D	7E	7F	-	-	-	-	-	-

Figure A-4. Translate Table for Compaction (XLTC)

POWER/VS uses the TRTC and XLTC tables in the following way to perform data compaction:

POWER/VS first scans the data to separate strings of duplicate data from strings of nonduplicate characters. Then POWER/VS scans the nonduplicate strings for characters outside the compact code subset using the TRTC table.

If a character outside the compact code subset is found, POWER/VS does not attempt to perform compaction. The string goes into the request unit as a noncompressed string. If all characters are members of the compact code subset, POWER/VS translates them to compact code, using the XLTC table.

POWER/VS then scans the data, looking for two adjacent master characters and combines these characters into one byte.

The example below shows how this algorithm works for the following sentence:

"1. Scans the data, looking for consecutive duplicate character strings of length 3 or greater."

This would appear in storage (in hexadecimal) as:

```
F14B40E2 838195A2 40A38885 408481A3 816B4093 96969289
95874086 96994083 9695A285 83A4A389 A5854084 A4979389
8381A385 40838881 998183A3 859940A2 A3998995 87A24096
86409385 9587A388 40F34096 99408799 8581A385 994B
```

POWER/VS would first scan the string, looking for consecutive duplicate character strings of length 3 or greater. No such string exists in the first 63 characters, so POWER/VS would establish the 1 (X'F1') as the beginning of a non-duplicate string, and the blank (X'40') character between the words 'character' and 'strings' as the end of the nonduplicate string. POWER/VS then tests this string, using the example TRTC table, and finds that all characters are members of the compact code subset. POWER/VS then translates the whole string to compact code, using the XLTC table. The resulting string (first 63 bytes only) becomes:

```
9D6EFCCD E1F0F6F9 FCFAE3F2 FCF1F0FA F02DFCF5 F7F7E5F4
F6F3FCE2 F7F8FCE1 F7F6F9F2 E1FBFAF4 E9F2FCF1 FBE7F5F4
E1F0FAF2 FCE1E3F0 F8F0E1FA F2F8FC
```

Now POWER/VS scans the string, looking for two adjacent master characters. The target buffer for this example then becomes (first byte is the SCB):

```
6A9D6EFC CDE1069C FAE32C10 A02DC577 E5463CE2 78FCE176
92E1BAF4 E92C1BE7 54E10A2C E1E308F0 E1A28C
```

The underlined bytes represent two master characters. Thus, 63 characters have been reduced to 43, including the SCB.

Appendix B. RJE, SNA I/O Specifications

This appendix describes the major functions of POWER/VS RJE,SNA with respect to the handling of input and output information. Data directed from a peripheral device to the host system is referred to as inbound data; information directed from the host system to a peripheral device is called outbound data.

POWER/VS processes the following types of inbound data:

- Console data
- Card data.

In the outbound direction, the following information can be processed:

- Print data
- Punch data
- Console data.

The following sections discuss the handling characteristics for these types of data. Additionally, a summary of the support for Data Flow Control (DFC) requests is also provided. Refer to Figure B-4 for a support summary of the POWER/VS RJE,SNA data communication facilities, to Figure B-5 for a summary of the single character string (SCS) control characters supported by POWER/VS RJE,SNA, and to Figure B-6 for an overview of the data stream interruptions supported by POWER/VS for SNA terminals.

Inbound Console Data

POWER/VS assumes that inbound console data exclusively consists of commands, and treats this information as such. It deblocks each RU into 80-byte images using the same scanning algorithm as described later in the section "Inbound Card Data". Inbound console data may be transparent or nontransparent; however, POWER/VS does not support compression for inbound console data.

Inbound Card Data

Inbound card data consists of one or more jobs, delimited either by DOS/VS JCL job delimiter statements (// JOB and /&) or by POWER/VS JECL job delimiter statements (* \$\$ JOB and * \$\$ EOJ). Figures B-1 and B-2 give examples of these two types of inbound card data streams.

```

// JOB A                Start job
.
.
.
/&                      End job
// JOB B                Start job
.
.
.
/&                      End job
* .. START LST,class   See note
* .. START PUN,class
* .. DISPLAY RDR,class
// JOB C                Start job
// EXEC PGM
data
.
.
.
/*
/&                      End job

```

Figure B-1. Inbound Card Data with DOS/VS JCL Job Delimiters

Note: You may enter remote operator commands punched in cards; however, these will only be recognized and acted upon when they are submitted between DOS/VS or POWER/VS job boundaries.

```

* $$ JOB                Start job
.
.
.
* $$ EOJ                End job
* .. START LST,class   See note
* .. START PUN,class
* .. DISPLAY RDR,class
* $$ JOB                Start job
// JOB A
.
.
.
/&
// JOB B
// EXEC PGM
data
.
.
.
/*
/&
* $$ EOJ                End job

```

Figure B-2. Inbound Card Data with POWER/VS JECL Job Delimiters

Note: You may enter remote operator commands punched in cards; however, these will only be recognized and acted upon when they are submitted between DOS/VS or POWER/VS boundaries.

Inbound card data may be transmitted from a card reader or a disk (optional device on certain work stations). POWER/VS treats both device types identically.

If POWER/VS did not encounter a valid job delimiter at the end of inbound data transmission (that is, either a /& or a * \$\$ EOJ statement), it will add the appropriate job delimiter. Then the job is placed in the Hold queue, and a message is sent to the originating remote operator stating the disposition of the job. The operator may then issue a command to release or to delete this job.

Data from a work station is transmitted in the form of request/response units (RUs) which may be up to 256 bytes long. POWER/VS deblocks each RU and creates 80-byte card images using the following algorithm:

1. The first 80 bytes of the RU are scanned for a record delimiter (SCS character).
2. If no record delimiter is detected, an interchange record separator (IRS) is assumed to follow the 80 bytes.
3. If the next (81st) byte is a record delimiter, it is ignored and the next logical record is assumed to start with the 82nd byte.
4. If the 81st byte is not a record delimiter, this byte is assumed to be the first byte of the next logical record.
5. If a record delimiter is found before the 80th byte, this delimiter is discarded and the record is padded with blanks to 80 bytes. The byte following the record delimiter is considered to be the first byte of the next logical record.

Outbound Print Data

For print jobs, POWER/VS supports the following SCS functions:

- SELECT CHANNEL nn
- FORMS FEED
- CARRIAGE RETURN
- NEW LINE

These functions are briefly discussed below.

SELECT CHANNEL nn (SELnn), X'04nn'

This command is inserted after a print line if the request was to skip to channel nn after printing. The following codes apply:

<u>Channel</u>	<u>Hexadecimal Representation</u>
1	X'81' (see note below)
2	X'82'
3	X'83'
4	X'84'
5	X'85'
6	X'86'
7	X'87'
8	X'88'
9	X'89'
10	X'7A'
11	X'7B'
12	X'7C'

Note: If SEL01 is specified, this value is converted to, and transmitted as, the FORMS FEED character (X'0C').

FORMS FEED (FF), X'0C'

This command is transmitted instead of the SELECT CHANNEL 1 command.

CARRIAGE RETURN (CR), X'0D'

This command is inserted after the print line if the request was to print with no space.

NEW LINE (NL), X'15'

This command is inserted before or after the print line depending on whether the request was to space one line before or after printing. If two or more blank lines are requested, the appropriate number of NL commands are inserted.

Notes:

1. POWER/VS does not support the Select Vertical Format (SVF) feature. Therefore, it is the responsibility of the remote operator to set up the vertical tab tables at the remote work station. However, the FCB name is sent if PDIR is supported and if the name has been specified in the * \$\$ LST or * \$\$ PRT statement.
2. The maximum logical record length is 132 bytes.

Outbound Punch Data

Outbound punch data consists of one or more punch jobs from the output class specified in the START PUN[,class] command by the remote operator.

The punch data stream is transmitted in transparent mode unless ASCII was specified in the BIND parameters for that work station. Compression and compaction are not supported for punch output.

Each logical record is preceded by a TRN byte and a 1-byte binary count indicating the number of bytes of transparent data. Trailing blanks are truncated so that the data count for each record is less than, or equal to 80 bytes. An IRS character follows each logical record. The TRN and the IRS characters are the only two SCS characters supported for punch output.

Outbound Console Data

Outbound console data consists of messages. The message processor uses the NL character to properly format each message. The NL character is the only SCS character used by the message processor.

Data Flow Control (DFC) Requests

Figure B-3 shows the DFC requests supported for both inbound and outbound SNA support.

DFC Request	Inbound	Outbound
CANCEL	YES	YES
SIGNAL	YES	NO
LUSTAT	YES	N/A
CHASE	NO	NO
SHUTD	N/A	NO
SHUTC	NO	N/A
RSHUTD	YES	N/A

Figure B-3. DFC Request Support for SNA

INBOUND DFC REQUESTS

CANCEL: CANCEL may be received at any time. If received during in-chain-state, POWER/VS will discard the current job being received, that is, POWER/VS will not place it into the input queue and will free the resources. A positive response is sent to CANCEL. If received during between-chain state, POWER/VS will positively respond to CANCEL but take no further action.

SIGNAL: SIGNAL indicates a request for change in direction. If outbound processing is performed, POWER/VS will force the end-of-chain and send CD with FMH1 indicating suspend.

If POWER/VS is in receive or standby state, SIGNAL will be ignored. The positive response to SIGNAL is sent by VTAM automatically.

LUSTAT: The following LU status codes are supported by POWER/VS:

X'0001XX00' Component now available.
X'081CXX00' Component failure: permanent error for an outbound device.

where XX = device selection byte of FMH1.

All other codes and conditions will be negatively responded to, the outbound processor sends ADS (if in DS state) and terminates.

CHASE: CHASE is not supported. A negative response indicating function not supported is returned.

SHUTC: Shutdown Complete is not supported. This should never be received as it is only returned in response to SHUTD which is not sent by POWER/VS.

RSHUTD: Request Shutdown is interpreted by POWER/VS as a request for CLEAR,UNBIND when the current active processor completes.

The inbound processor waits for an FM Header indicating Abort or End Data Stream with EB or CD.

The outbound processor completes the sending of the current job(s) and any pending messages, then terminates the session.

OUTBOUND DFC REQUEST

CANCEL: CANCEL will be sent by the outbound processor whenever a negative response is received, and during in-chain-state. There are no other conditions under which POWER/VS will send CANCEL.

POWER/VS RJE,SNA Facilities	Inbound Data		Outbound Data		
	Console	Card	Print	Punch	Console
Compression	No	No	Yes (1)	No	No
Compaction	No	No	Yes (1)	No	No
RU Spanning	No	No	Yes	No	No
ASCII Characters	Yes	Yes	Yes	Yes	Yes
Truncate Blanks	Yes (2)	Yes (2)	No	Yes	-
Transparency	Yes (1)	Yes (1)	No	Yes (1)	No
FMH Type	1 and default	1	1,2,3	1	Default

Notes:
1. If ASCII is not specified.
2. Depending on the work station.

Figure B-4. Data Communication Facilities Supported by POWER/VS RJE,SNA

POWER/VS SNA SCS (Single character string) control characters supported	Value	Data Inbound to Host System		Data Outbound from Host System		
		Card	Console	Punch	Print	Console
New Line (NL)	X'15'	Yes	Yes	No	Yes	Yes
Carriage Return (CR)	X'0D'	Yes	Yes	No	Yes	No
Forms Feed (FF)	X'0C'	Yes	Yes	No	Yes	No
Interchange Record Separator (IRS)	X'1E'	Yes	Yes	Yes	No	No
Select Channel (SEL)	X'04xx'	No	No	No	Yes	No
Transparency (TRN)	X'35'	Yes	Yes	Yes	No	No
Required New Line (RNL)	X'06'	Yes	Yes	No	No	No

Figure B-5. SCS Control Characters Supported by POWER/VS RJE,SNA

Interrupt of	Interrupt by	Inbound data		Outbound data		
		Console	Card	Print	Punch	Console
Inbound data						
• Console	-	No	No	No	No	No
• Card	Yes (2)	-	No	No	No	No
Outbound data						
• Print	Yes	Yes	-	No	Yes (1)	
• Punch	Yes	Yes	No	-	Yes	
• Console	No	No	No	No	-	
Outbound data already interrupted by inbound card data						
• Print	Yes (2)	No	-	No	No	
• Punch	Yes (3)	No	No	-	No	
Notes:						
1. If CONSOLE=YES in PRMT macro.						
2. Function not available for the 3770.						
3. Function not available for the 3770 and 3790.						

Figure B-6. Data Interruptions Supported by POWER/VS RJE,SNA

Appendix C. LOGON Mode Table and BIND Parameter Requirements

To use POWER/VS with RJE,SNA, you must have generated VTAM under your DOS/VS operating system. This appendix concentrates on the LOGON mode (LOGMODE) table to be defined during the VTAM generation. One entry is required in the LOGMODE table for each individual group of sessions with identical VTAM BIND characteristics.

The LOGMODE table, which must be cataloged in the VTAM core image library (private or system), contains the BIND parameter specifications as required for each individual group of sessions. As a remote operator, you must specify the name of the appropriate entry in the LOGMODE table when you issue the LOGON command. For details concerning the BIND parameters in general, refer to the DOS/VS VTAM System Programmer's Guide, GC27-6957.

The following example (Figure C-1) shows the definition of a LOGMODE table entry by means of the MODEENT macro. Note that the continuation characters that must be specified in column 72 have been omitted in the example.

	Byte no. in BIND
username MODETAB	
username MODEENT LOGMODE=name,	
FMPROP=X'03',	2
TSPROP=X'03',	3
PRIPROT=X'A3', =X'A1', (without compression (LST,PUN))	4
SECPROT=X'A1',	5
COMPROT=X'7080', =X'7880', (with ASCII (3770 only))	6,7
RUSIZES=X'8585'	8,9

Figure C-1. LOGON Mode Table Coding Example

Figure C-2 shows the BIND parameters for the presentation services to be used with a 3790 or 3777 work station. The string of 12 hexadecimal values applies to bytes 14 through 25 of the BIND parameters.

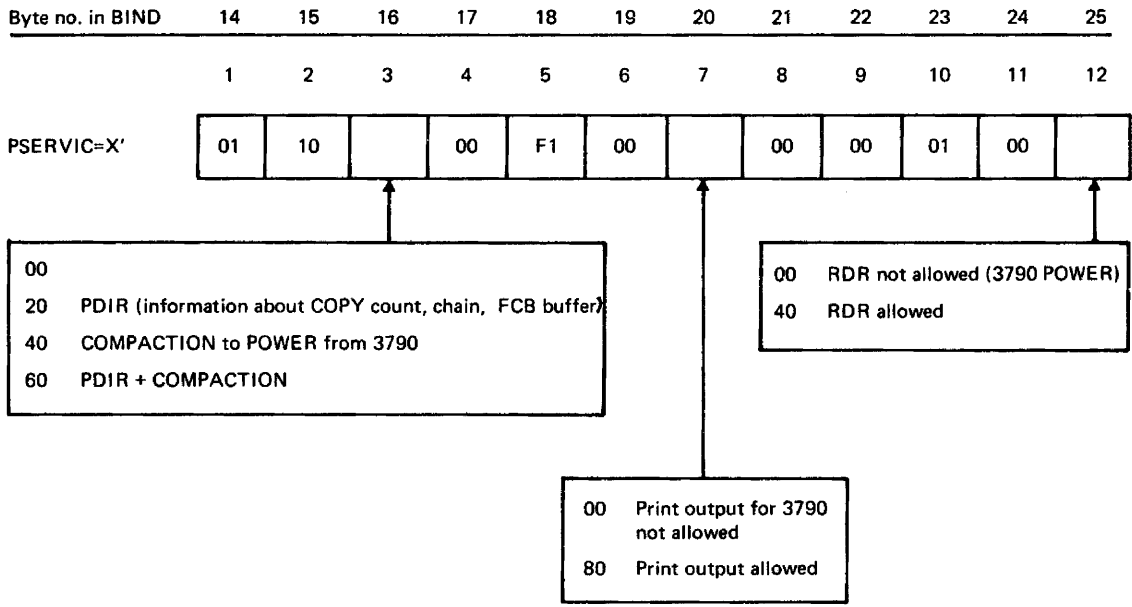


Figure C-2. BIND Parameters for the Presentation Services

Appendix D. Examples of Status Report Output

This appendix describes two status report examples for the IBM 3800 Printer. They show the status report output that results when you issue the PDISPLAY command shown. In each example, the numbered notes describe the item indicated with the same circled number on the listing.

Example 1. If you issue the PDISPLAY command:

d lst

the status report has the following form:

	F1	IR46I	LIST QUEUE	P	D	C	FROM TO	PAGES	CC	FORM
First job	{	F1	IR46I	PWR001	00162	3	D B 000 000	3	*8	FFFF
	{	F1	IR46I	FLASH=FLSH,BURST,COPIES= 36						
Second job	{	F1	IR46I	PWR002	00170	4	H C 000 000	518	1	ABCD
Third job	{	F1	IR46I	PAYROLL	00172	8	K P 000 000	1815	*2	PAYF
	{	F1	IR46I	FLASH=PAYR,COPIES= 4						

(1) →
(4) →
(5) →
(2) →
(3) →
(6) →
(7) →

Notes:

- 1 The name of the output job is PAYROLL, and it has an associated job number of 172, a priority of 8, a disposition of K (keep), and an assigned class of P. The job input has been read from a local reader device, and the output is to be printed on a local printer. The job output contains 1815 pages.
- 2 The asterisk indicates that the output file is destined for printing on a 3800 Printing Subsystem.
- 3 The output is printed twice (that is, the transmission count is 2).
- 4 The forms overlay name is PAYR.
- 5 The total number of copies produced for this job is 4 (although the output is only transmitted twice across the channel).
- 6 The forms number requested for the job is PAYF.
- 7 "BURST" indicates that the paper must be threaded through the Burster-Trimmer-Stacker before printing of job PWR001 can be started.

Example 2. If you issue the PDISPLAY command:

d a

the status report has the following form:

```

F1 IR48I LST,00E,T ,2 POWERM12 00008 A
F1 IR48I F2,00C,A
F1 IR48I RDR,00C,A INACTIVE
INACTIVE

```

Notes:

1 "RDR" indicates the type of task. Possible types are:

- LST -- writer task
- PUN -- punch task
- RDR -- reader task
- BG-F6 -- execution task.

2 "00C" is the unit record device address associated with this task.

3 This position shows the class or classes that can be processed by the task. The order in which the classes are specified shows the order of processing.

4 This position indicates the number of buffers used, where:

- 1 = 1 print buffer
- 2 = 2 print buffers, 1 input buffer for data file
- D = 2 print buffers, 2 input buffers for data file.

5 This identifies the job name and job number of the job currently being processed when the task is active.

6 This is the job class associated with the job.

7 This indicates that the corresponding task is inactive.

Appendix E. Reader Exit Routine

For a user-written reader exit routine, you must consider the following:

POWER/VS performs a conversion to uppercase characters for all POWER/VS JECL and DOS/VS JCL statements (column 1 to 72) that start with // in columns 1 and 2. DOS/VS comment statements and ASSGN statements without // (permanent assignments) will not be translated.

The routine receives control from POWER/VS only for DOS/VS JCL or POWER/VS JECL statements. It must be reentrant if it is called from more than one partition; and it must be self-relocating unless the relocating loader option is in effect.

Any JCL or JECL statement can be changed or deleted and other statements can be inserted. The original statement is presented again after each inserted statement has been processed. When all the insertions have been made, a return code of X'00' or X'04' is placed in register 15 to accept or delete the original statement.

When the reader exit routine receives control from POWER/VS, register 0 contains the address of the statement read and register 1 the length of the statement. To return to POWER/VS, issue a BR 14 instruction. Between entry and exit of this routine, no operation may be performed that might cause a WAIT condition for the POWER/VS partition.

The user-written exit routine must not alter the contents of registers 10, 11, 12, and 13. These registers are reserved for POWER/VS internal use. Register 11 points to the task control block of the read task and may be used to identify the task.

Before returning control to POWER/VS, you must specify a return code in register 15. POWER/VS expects one of the following return codes:

- X'00' Normal return; process this statement.
- X'04' Delete; ignore this statement.
- X'08' Insert; process new statement, return original statement to the user. Any number of statements may be inserted. Preceding every job control statement that has to be inserted, one byte must be reserved. This byte is used as an interface with the POWER/VS message service when the job control statement is incorrect.
- X'0C' Flush the DOS/VS job.
- X'10' Flush the POWER/VS job.

When new statements are added (return code X'08'), the address of the statement must be provided in register 0 and its length in register 1. The length must be X'50'.

Flush conditions (return codes X'0C' and X'10') at job boundary (first card of a POWER/VS job) will be ignored and message 1R57I will be logged.

If ACCOUNT=YES was specified, the number of records read in the reader account record will include records added or deleted through the RDREXIT routine.

If a reader exit was generated and AUTOSTART is to be performed, the user must return to POWER/VS with X'00' in register 15.

Bibliography

DOS/VS System Management Guide, GC33-5371

DOS/VS Supervisor and I/O Macros, GC33-5373

DOS/VS System Generation, GC33-5377

DOS/VS Messages, GC33-5379

DOS/VS IBM 3800 Printing Subsystem Programmer's Guide, GC26-3900

DOS/VS VTAM System Programmer's Guide, GC27-6957

DOS/VS POWER/VS Work Station User's Guide, GC33-6049

DOS/VS POWER/VS Reference Summary, GX33-9004

DOS/VS POWER/VS Logic Part 1, SY33-8576

DOS/VS POWER/VS Logic Part 2, SY33-8577

IBM System/370 Model 115 Functional Characteristics, GA33-1510

IBM System/370 Model 125 Functional Characteristics, GA33-1506

IBM 2701 Data Adapter Unit, GA22-6864

System Components: IBM 2770 Data Communication System, GA27-3013

Component Description: IBM 2780 Data Transmission Terminal, GA27-3005

IBM 3770 Data Communication System, System Components, GA27-3097

Component Description for the IBM 3780 Data Communications Terminal, GA27-3063.

Index

\$JOBACCT phase 123
*abc operand
 in PALTER 55
 in PDELETE 58
 in PDISPLAY 61
 in PRELEASE 67
// EXEC statement 21

ACB
 (see application control block)
account file 116,2
 formatting of 133
 size of 13
account records 116
adding resources 9
additional extents for data
 file 132
adjusting forms alignment 69
algorithm for RU deblocking 159
alignment of forms 69
ALLOC 21
allocation of storage 130,7
ALLOCR 21
ALLUSERS
 message processing 57
 parameter 57
application control block 143
ASCII 33
attention routine commands 51
automatic start-up 126
AUTONAME 82
AUTOSTART facility 126,130
AUTOSTART feature 51
AUTOSTART statements
 errors in 126
 example of use 134,135
 FORMAT 126
 PRINTERS 127
 PUNCHES 128
 READER 127
 sequence of 135

BIND parameters for presentation
 services 166
bit settings for ICA line mode 49
breakdown 7
broadcast message text length 57
BSC line 142
BSC terminals 2
buffer expansion 37
 additional 37
Burster-Trimmed-Stacker (IBM 3800) 93

calculation of GETVIS pool
 size 30
carriage control tape 90
carriage return command 161
central operator commands 51
 short from of 51
central operator functions 5
changing attributes of
 POWER/VS jobs 54

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

channel selection codes 161
channel 1 skip 90
channel 12
 repeated posting of 30
character arrangement table
 specification 93
class specification
 changing of 138
 partition-dependent 138
 partition-independent 138
clear the 3800 page buffer 31
CMPACT parameter 36
coding work sheet 42,43
compact code 151
 example of 152
compaction 151
compaction rules 40
compaction table 151,36
 definition of 152
 generation 38
 identification 38
 name 39
 phase name 40
 storage 15
component selection (on IBM
 3780) 37
compression 154
 of nonduplicate characters 151
COMPROT parameter 165
conditional termination 145
CONSOLE parameter
 restrictions 36
 for single logical units 36
 for multiple logical units 36
 for the 3790 36
copy count display 60
copy group index 68
copy groups (IBM 3800) 89
copy modification (IBM 3800) 95
copy separator pages/cards 31,83,98
correction of JECL statements
 during job execution 77
 * \$\$ LST 85
 * \$\$ PRT 85
 * \$\$ PUN 95

CR
 (see carriage return command)
cross-partition communication 22,51
 macros 106,4
 return codes 115
CTLSPOOL macro 112,29
 NEWVAL parameter in 114
 request code specification in 113
 restrictions 4,112
 use of 136
 use of operator commands in 114

data block size
 assumptions 26
 for IBM 3211 printers 26
 for tape or disk intermediate
 storage 26

- data compaction 151,56
 - example 152,154
- data-driven output segmentation 85
- data file 2
 - additional extents 132
 - block size 12
 - entries in 7
 - formatting of 133
 - track group size 13
- data flow control request 157
 - inbound 162
 - outbound 163
 - support of 162
- data mode 100,148
- data stream interruptions 164,157
- deblocking of RUs, algorithm 159
- default priority 27
- defaults for IBM 3800 94
- defining RJE,BSC lines 32
- DFC request
 - (see data flow control request)
- direct access storage devices
 - support of 2
- direction of RJE job output 7
- diskette files 100
 - in SYSIN mode 147
 - in Data mode 148
- diskette unit 147
- display of copy count 60
- DOS/VS
 - JCL statements, conversion of 79
 - procedure library 134
 - special considerations 22
 - supervisor generation requirements 22
- dual code feature (IBM 2701) 33
- dual communication interface (IBM 2701) 33
- dummy devices 135
- dump 146
- EBCDIC 33
- eject-and-feed option 101
- emergency termination 146
- entering POWER/VS commands 136
- EOJ operand of PSTOP 73
- errors in AUTOSTART statements 126
- example
 - of AUTOSTART procedure 134,135
 - of POWER/VS initiation 130
 - of POWER/VS status report 132
- FCB
 - (see forms control buffer)
- FF
 - (see forms feed command)
- file verification 101
- fixable area 14,15
- flashing forms overlays 94
- flushing reader task 64
- flushing writer task 64
 - with several output copies 64
- FMPROF parameter 165
- FOPT macro
- YECB specification in 22

- format
 - of central operator commands 52
 - of JFCL statements 77
- FORMAT statement 126
- formatting POWER/VS files 133
- forms alignment 69
- forms control buffer 91,27
- forms feed command 160
- forms overlay (IBM 3800) 94
- forms specification 88
- generating DOS/VS supervisor 22
- generating POWER/VS 24
- GETSPOOL macro 111,29
 - LINENO parameter 112
 - required queue entry disposition 111
- GETVIS area 14,16
- GETVIS pool size
 - calculation of 29
- horizontal format control 37
- I/O trace 37
- IBM 2560 49
- IBM 2770
 - required features 44
 - supported features 45
 - supported I/O units 44
 - transparency feature 45
- IBM 2780
 - required features 45
 - supported features 45
 - transparency feature 46
- IBM 3525 50
- IBM 3540 diskette I/O unit 147
- IBM 3741
 - compatibility with IBM 2780 46
 - required features 46
- IBM 3771 46
 - attachable devices 46
- IBM 3773 46
 - integrated diskette 46
- IBM 3774 47
 - attachable devices 47
- IBM 3775 47
 - attachable devices 47
- IBM 3776 47
 - attachable devices 47
- IBM 3780
 - components 48
 - required features 48
 - supported features 48
 - transparency feature 48
- IBM 3790 49,9
 - attachment to host system 49
 - logical devices 49
 - RJE work station package 49
- IBM 3800
 - JECL statement parameters 89,93
 - POWER/VS generation parameters 31
 - support 50
- IBM 5425 49

- ICA line mode
 - bit settings 49
- ICRXECB 107,109
- improving system efficiency 9
- INACTIVE status 66
- inbound card data
 - characteristics 157
 - interruptions 164
- inbound console data
 - characteristics 157
 - interruptions 164
- inbound data 157
- initiating POWER/VS 129
 - with AUTOSTART 134
 - without AUTOSTART 130
- inserting diskette files 100
- installation planning 10
- installation prerequisites 10
- installing POWER/VS 10
- interchange record separator 150,162
- intermediate storage requirements 11
- IOTAB macro
 - specification of LUB entries in 22
- IRS
 - (see interchange record separator)
- ISTLINK7 24
- JECL
 - (see job entry control language)
- JECL commands 136
- JECL delimiter statements
 - end of job 105
 - end of jobstep 105
- JECL functions 77
- JECL statements
 - * \$\$ CTL 81
 - * \$\$ DATA 81
 - examples 103,104
 - * \$\$ EOJ 82
 - * \$\$ JOB
 - positional 82
 - keyword 83
 - * \$\$ LST
 - positional 85
 - keyword 86
 - * \$\$ PRT
 - positional 85
 - keyword 86
 - * \$\$ PUN
 - positional 95
 - keyword 96
 - * \$\$ RDR
 - positional 100
 - keyword 100
 - * \$\$ SLI 101
 - examples 103,104
- execution-time corrections 77
- format of 77
- functions of 77
- keyword parameters 78,80
- positional parameters 78,80
- restrictions 80
- summary 79
- use of 79

- job accounting 116,51,4
- job control 3
- job delimiter statements
 - DOS/VS job control 157,75
 - POWER/VS JECL 157,75
- job entry control language 51,75
- job execution 3
- job input 3
- job names
 - one-character names 83
 - restrictions 83
- job output 3
- job segmentation 29
- job separation 28,88,98
- length of broadcast message text 57
- LINEWO parameter in GETSPOOL 111
- link-editing NCP modules
 - job stream 24
- list queue entry 85
- list task 140
- local unit record I/O devices
 - support of 2
- LOGGED ON status 66
- LOGGING ON status 66
- logical boundaries for segmentation 124
- logical reader 9
- logical printer 9
- logical punch 9
- logical unit 143
- logical unit name storage 15
- logical unit buffer limit 23
- logical unit names
 - avoiding names EOJ and ALL 23
- LOGMODE 165
- LOGON mode table 165
 - coding example 165
- LOGON storage 15
- LOSTERM exit 144
- LST operand
 - use for different reports 92
- LU
 - (see logical unit)
- machine requirements 10
- macros
 - accounting support 123
 - cross-partition communication 106
 - POWER/VS generation 24
 - segmentation 124
- mark form function (IBM 3800) 31
- master characters (compaction)
 - number of 39,151
 - range 151
 - rules 40
 - selection of 151
 - special characters 40
- maximum resource requirements 9
- minimum storage requirements 9,10
- MLU
 - (see multiple logical unit)

MODEENT macro 165
multiple channel 12 posting 30
multipurpose UR devices 134
 interrupting operations on 146
 restarting interrupted operations 147
multivolume files 89,148

NCP

(see network control program)
NCP requirements 23
nested SLI statements 106
network control program 8
new line command 161
NL
(see new line command)
NOLOG option 75
non-master characters (compaction) 39
 rules 40
 number of 40
 special characters 40
normal termination 145
NOT INITIATED status 66
NOT LOGGED ON status 66
NOT SUPPORTED status 66

outbound console data
 characteristics 161
 interruptions 164
outbound data 158
outbound print data
 characteristics 159
 interruptions 164
 SCS functions supported 160
outbound punch data
 characteristics 161
 interruptions 164
output segmentation 124,51
 count-driven 4,75
 data-driven 4,85
 program-driven 4
output spooling to tape 89,98

PACCOUNT command 53
 examples 54
pageable area 14,16
PALTER command 54
 examples 56
 use in CTLSPOOL macro 114
partition priority 1
password
 overriding of 33
password definition
 in POWER/VS generation 33
 during POWER/VS execution 33
password specification
 in PLINE macro 143
 in PSTART command 143
PBRDCST command 57
 examples 57
 length of message text 57
PCANCEL command 58
 use in CTLSPOOL macro 114

PCPTAB macro 38
PDELETE command 58
 examples 59
 use in CTLSPOOL macro 114
PDISPLAY command 59,138
 copy count indication
 for active task 60
 for inactive task 60
 examples 63
 items displayed 59,60
 status report 59
 example of 167
 use in CTLSPOOL macro 114
PEND command 63
 examples 63
 requesting status report 63
permanent area 14,15
PFLUSH command 64
 examples 65
PGO command 65,69
 examples 65
phase name of compaction table 39
PINQUIRE command 66
 example 67
 restriction for short form 66
 status information 66
planning the installation 10
PLINE macro 32
 applicability 25
POCL
(see control operator commands)
POWER macro 25
POWER/VS
 account file 123
 accounting support 22
 AUTOSTART 126
 central operator commands 51
 format of 52
 concept 3
 files 2
 initiation
 with AUTOSTART 134
 without AUTOSTART 130
 initiation statements 131
 example 131
 installation 10
 job
 end 82
 start 83
 major functions 6
 partition
 changing priorities of 136
 structure of 14
 phase names 41
 queues
 manipulation of 1
 operator control of 1
 requirements in DOS/VS 7
 RJE task 142
 status report example 132
 storage allocation 130
 storage requirements 13
 task

POWER/VS (continued)

- activation of 138
- deactivation of 140
- terminal support 44
- termination
 - emergency 146
 - normal 145
- POWER/VS application name
 - specification for VTAM 30
- POWER/VS generation macros 24
 - coding conventions 24
 - macro sequence 32
 - examples 38
- PRELEASE command 67
 - examples 68
 - use in CTLSPPOOL macro 114
- presentation services 165
 - BIND parameters for 166
- PRESTART command 68
 - backward/forward count in 68
 - example 68
 - page definition 68
- PRINTERS statement 127
- priority 1,136
- PRIPROT parameter 165
- PRMT macro 33,25
 - short form of 35
- PROCESSING status 66
- programming systems 8
 - required 8
 - optional 8
- PSETUP command 69
 - adjusting forms alignment 69
 - example 69
 - page definition 69
- PSTART command 70
 - buffers 70
 - examples 72
 - long-running jobs 71
 - segmentation 71
 - use of 133
 - verifying diskette files 72
 - volume sequence checking 72
- PSTOP command 73,140
 - EOJ operand 73,141
 - examples 74
 - RESTART operand 73,141
- punch task 140
- PUNCHES statement 128
- PUTACCT macro 123
 - example of 123
- PUTSPOOL macro 29,109
- queue entries
 - creation of 7
- queue file 2,13
 - formatting of 133
 - size of 11
- queue management commands 136

- RDREXIT phase 14
- reader exit
 - conventions 169
 - parameter 29
 - return codes 169
 - user-written task 29
- READER statement 127
- reader task 139
- reader-only partition 127,128
- real partition size 14,16
- REF parameter
 - to use short PRMT from 35
 - restrictions 35
 - errors in referenced PRMT 35
- remote control block 15
- remote job entry 7,51,142
 - devices 2
- remote operator command language 137
- repeated posting of channel 12 30
- request/response unit 159
- requested operation for CTLSPPOOL 113
- requesting display
 - of active tasks 62
 - of ALLUSERS messages 62
 - of free queue records 62
 - of system messages 62
 - of time and date 62
- resource requirements 9
- RESTART operand of PSTOP 73
- restarting
 - flushed writer task 64
 - partition 139
- return codes
 - from PUTSPOOL, GETSPOOL, CTLSPPOOL 115
 - from reader exit task 169
 - from SEGMENT macro 125
- returning partitions
 - to DOS/VS control 142
 - to POWER/VS control 142
- RJE
 - (see remote job entry)
- RJE job output
 - direction of 7
- RJE task 142
- RJE,BSC operations
 - control of 143
- RJE,SNA facilities 163
- RJE,SNA operations
 - control of 144
- RJE,SNA session
 - start-up 143
 - termination 145
- RJE,SNA support
 - NCP requirements 23
 - SIZE parameter requirement 132
 - specification in POWER macro 29
 - VTAM requirements 23
- RMCB
 - (see remote control block)

ROCL
 (see remote operator command language)
 RSIZE 11,13
 RU
 (see request/response unit)
 RU deblocking algorithm 159
 RUSIZES parameter 165

 SCB
 (see string control byte)
 SCS
 (see single character string)
 SCS control characters 163,154
 SDLC
 (see synchronous data link control)
 SDLC terminals 2
 secondary logical unit names 36
 SECPROT parameter 165
 SEGMENT macro 124
 return codes 125
 segmentation boundaries 124
 segmentation of output 4
 select vertical format 160
 SEL01 160
 separator pages/cards
 copy separation 31,88,98
 job separation 28,88,98
 sequence of AUTOSTART statements 135
 SESSLIM parameter 36
 SETDP operator command 85
 SETPRT macro 4,125
 sign on 143
 simplified LOGON procedure 23
 single character string 154
 SIZE parameter (in // EXEC) 9,132
 short form
 of central operator commands 52
 of PRMT macro 35
 SLI update statements 81,101
 SNA
 (see system network architecture)
 SNA logical unit 22
 SNA support 7
 SNA terminals 143
 SNA unit control block 15
 source statement library 81
 reserved sublibraries 102
 search sequence 101
 space calculation
 for SNA control blocks 15
 space compression/expansion 37
 SPL
 (see spool parameter list)
 SPL DSECT 107
 SPL macro 107
 SPL TYPE=MAP requirement 106
 SPMXECB 106,109
 SPOOL interface 29
 spool parameter list 107
 DSECT 107
 spool parameter list definition
 in CTLSPPOOL macro 113
 in GETSPPOOL macro 111
 in PUTSPPOOL macro 109
 spooling 1
 spooling to tape 89
 START command
 use of 131
 starting lines 142
 status report 145
 example 132
 stopping lines 141
 storage allocation 7,21,130
 storage calculation
 base values 16,17
 example 18
 storage requirements, minimum 9
 string control byte 153,156
 sublibrary 27
 SUCB
 (see SNA unit control block)
 SUPVR macro 22
 SVP
 (see select vertical format)
 synchronous data link control 143
 SYSIN files
 on diskette 72
 SYSIN mode 100,147
 SYSRDR assignment 131
 system breakdown 7
 system dispatching priority 7
 system dump 146
 system network architecture 7
 system requirements 9

 tailoring DOS/VS 7
 task management commands 136
 teleprocessing facility 7
 terminal attachment requirements
 for BSC 10
 for SNA 11
 terminal commands 51
 terminals
 support of 2
 termination of POWER/VS
 emergency 146
 normal 145
 termination of SNA sessions
 all sessions 145
 by central operator 145
 conditional 145
 individual sessions 144
 unconditional 145
 timeout 32
 track group 27
 translate table for compaction 154
 example 155
 translate-and-test table for
 compaction 154
 example 155

transmission code 33
 transparency 163
 transparency feature 33,37
 TRN
 (see transparency)
 TRTC
 (see translate-and-test table for
 compaction)
 TSPROF parameter 165

 UCB
 (see universal character set buffer)
 unconditional LOGOFF 145
 unit-record devices
 support of 2
 universal character set buffer 91
 user exit
 (see reader exit)
 user information
 in *\$\$ JOB statement 84
 user partition 138
 user responsibilities 7
 using POWER/VS 51
 USS tailoring services 23

 VARY command 144
 Virtual Telecommunications
 Access Method 8,143
 VM/370 8
 volume sequence checking 101
 VSIZE 14
 VTAM
 (see Virtual Telecommunications
 Access Method)
 VTAM requirements 23
 VTAM interface
 activating 72
 deactivating 74
 partition priority 143

 WACB
 (see work area control block)
 warm start 7,133
 work area control block 15
 work sheet for POWER/VS generation 42,43
 work station package 49
 writer task 140
 writer-only partition 127
 WSP
 (see work station package)

 XECB specification
 for cross-partition communication 29
 XECBTAB macro 106
 XLTC
 (see translate table for compaction)

READER'S COMMENT FORM

**DOS/VS POWER/VS
Installation Guide and Reference**

GC33-6048-1

Please comment on the usefulness and readability of this publication, suggest additions and deletions, and list specific errors and omissions (give page numbers). All comments and suggestions become the property of IBM. If you wish a reply, be sure to include your name and address.

COMMENTS

**THANK YOU FOR YOUR COOPERATION
PLEASE FOLD ON TWO LINES, STAPLE AND MAIL**

YOUR COMMENTS, PLEASE

Your comments on the other side of this form will help us improve future editions of this publication. Each reply will be carefully reviewed by the persons and department responsible for writing and publishing this material.

Please note that requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or the IBM branch office serving your locality.

IBM Program Product Center
Publications Department (0446)
Schwertstrasse 58

D-7032 Sindelfingen
Germany

