

PERKIN-ELMER

**SPOOLER (SPL/32)
SYSTEM ADMINISTRATION**

Reference Manual

48-056 F01 R00

The information in this document is subject to change without notice and should not be construed as a commitment by The Perkin-Elmer Corporation. The Perkin-Elmer Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license, and it can be used or copied only in a manner permitted by that license. Any copy of the described software must include the Perkin-Elmer copyright notice. Title to and ownership of the described software and any copies thereof shall remain in The Perkin-Elmer Corporation.

The Perkin-Elmer Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Perkin-Elmer.

The Perkin-Elmer Corporation, Data Systems Group, 2 Crescent Place, Oceanport, New Jersey 07757

© 1982, 1984 by The Perkin-Elmer Corporation

Printed in the United States of America

TABLE OF CONTENTS

PREFACE		xi
CHAPTERS		
1	INTRODUCTION TO SPL/32	
1.1	INTRODUCTION	1-1
1.2	COMPONENTS OF THE SPOOLING PROCESS	1-3
1.2.1	Input Spooling	1-3
1.2.2	Output Spooling	1-3
1.2.3	The Spool Queue	1-3
1.2.4	The Pseudo Device	1-3
1.2.5	The Spool File	1-3
1.2.6	The Uniform Spooler Control Interface (USCI)	1-3
1.3	REVIEW	1-4
1.4	COMPONENTS OF SPOOLER CONTROL	1-5
1.4.1	Control and Subcontrol Tasks	1-5
1.4.2	Levels of Authorization	1-5
1.5	REQUIREMENTS FOR SPL/32	1-7
1.5.1	Hardware	1-7
1.5.2	Memory Requirements	1-7
1.5.3	Software Requirements	1-8
1.6	INITIALIZING SPL/32	1-8
1.6.1	Loading SPL/32	1-8
1.6.2	Linking SPL/32	1-9
1.6.3	Starting the Spooler	1-9
1.6.3.1	Creating a START Command File	1-10
1.6.4	SPL/32 System Console Interface	1-11
1.6.5	Ending the Spooler Task	1-12
2	INPUT SPOOLING	
2.1	INTRODUCTION	2-1
2.2	ESTABLISHING INPUT SPOOLING	2-1
2.3	INPUT SPOOLING CONTROL STATEMENTS	2-1
2.3.1	The /@INPUT Control Statement	2-1

CHAPTERS (Continued)

2.3.2	The /@SUBMIT Control Statement	2-3
2.3.3	Ending Input Spooling of a Card File	2-4
2.4	REMOVING INPUT SPOOLING DEVICES	2-5
3 THE SPOOL FILE AND THE SPOOL QUEUE		
3.1	INTRODUCTION	3-1
3.2	THE SPOOL FILE	3-1
3.2.1	Naming Spool Files	3-2
3.3	THE SPOOL QUEUE	3-4
3.3.1	Creating the Spool Queue	3-4
3.3.2	Displaying the Spool Queue File Name	3-4
3.3.3	Respecifying the Spool Queue	3-5
3.3.4	Respecifying the Spool Queue While the Spooler is Active	3-5
3.3.5	Interrogating the Spool Queue	3-5
3.3.5.1	Displaying Entries in the Spool Queue	3-5
3.3.5.2	Displaying Queue Entries According to Preprinted Form Requests	3-7
3.3.5.3	Displaying Files by Assigned Pseudo Device	3-7
3.3.5.4	Displaying Information About a Particular File	3-8
3.3.5.5	Displaying the Spooler Environment Configuration	3-8
3.3.5.6	Displaying All Spool Queue and Spooler Configuration Information	3-9
3.3.5.7	Functional Discussion of the INTERROGATE Command	3-10
4 OUTPUT SPOOLING		
4.1	INTRODUCTION	4-1
4.2	THE PSEUDO DEVICE	4-2
4.2.1	The Global Pseudo Device	4-2
4.2.2	Creating Pseudo Devices in the Spooling Environment	4-3
4.2.3	Associating Pseudo Devices with Physical Devices	4-4
4.2.4	Directing Output to a Pseudo Device	4-5
4.2.5	Limiting Default Access to a Pseudo Device	4-6
4.2.6	Establishing Unique Default Print Location	4-6
4.2.7	Disassociating Pseudo Devices From Physical Devices	4-7

CHAPTERS (Continued)

4.2.8	Deleting Pseudo Devices From the Spooling Environment	4-7
4.3	PHYSICAL OUTPUT DEVICE CONTROL	4-8
4.3.1	Altering the Status of Physical Output Devices	4-8
4.3.2	Removing Files From the Spool Queue	4-9
4.3.3	Removing Physical Devices From the Spooling Environment	4-10
4.4	USING PREPRINTED FORMS	4-10
4.4.1	Naming and Characterizing Preprinted Forms	4-11
4.4.2	Associating a Form With a Physical Device	4-11
4.4.3	Removing Forms From the Spooling Environment	4-12
4.5	USING IDENTIFICATION PAGES	4-13
4.6	USING THE CHECKPOINT FEATURE	4-14
4.6.1	Turning Checkpointing OFF	4-15
4.7	QUEUE HANDLING	4-16
4.7.1	Changing the Global Default Priority	4-16
4.7.2	Altering the Output Priority of a Spool File	4-17
5	TASK AND LEVELS OF AUTHORIZATION	
5.1	INTRODUCTION	5-1
5.2	CONTROL TASKS AND THEIR COMMUNICATION WITH THE SPOOLER	5-2
5.2.1	Command Messages	5-2
5.2.2	Acceptance Messages	5-2
5.2.3	Request For Mnemonic Table Messages	5-2
5.3	SPOOLER COMMUNICATION WITH CONTROL TASKS	5-3
5.3.1	Service Response Messages	5-3
5.3.2	Exception Response Messages	5-3
5.3.3	Authority Messages	5-3
5.3.4	Attention Messages	5-3
5.3.5	Mnemonic Messages	5-3
5.4	INCLUDING CONTROL TASKS IN THE SPOOLING ENVIRONMENT	5-4
5.5	DISPLAYING CONTROL TASKS IN THE SPOOLING ENVIRONMENT	5-4
5.6	REMOVING CONTROL TASKS FROM THE SPOOLING ENVIRONMENT	5-4

CHAPTERS (Continued)

5.7	SUBCONTROL TASKS AND THEIR COMMUNICATION THE SPOOLER	5-5
5.7.1	Command Messages	5-5
5.8	SPOOLER COMMUNICATION WITH SUBCONTROL TASKS	5-5
5.8.1	Service Response Messages	5-6
5.8.2	Exception Response Messages	5-6
5.9	INCLUDING SUBCONTROL TASKS IN THE SPOOLING ENVIRONMENT	5-6
5.10	DISPLAYING SUBCONTROL TASKS IN THE SPOOLING ENVIRONMENT	5-6
5.11	REMOVING SUBCONTROL TASKS FROM THE SPOOLING ENVIRONMENT	5-7
5.12	USER-LEVEL TASKS AND THEIR COMMUNICATION WITH THE SPOOLER	5-7
5.13	LEVELS OF AUTHORIZATION	5-8
5.13.1	Authorization Identifiers (AID)	5-10
5.13.2	Origin Identifiers (OID)	5-10
6	SPL/32 COMMANDS	
6.1	INTRODUCTION	6-1
6.2	ATTACH COMMAND	6-2
6.3	BANNER COMMAND	6-4
6.4	BLOCK COMMAND	6-7
6.5	CHANGE COMMAND	6-8
6.6	CHECKPOINT COMMAND	6-10
6.7	CREATE COMMAND	6-12
6.8	CTASK COMMAND	6-13
6.9	DELETE COMMAND	6-15
6.10	DETACH COMMAND	6-16
6.11	DEVICE COMMAND	6-17
6.12	END COMMAND	6-20
6.13	FLUSH COMMAND	6-21

CHAPTERS (Continued)

6.14	FORM COMMAND	6-23
6.15	GLOBAL COMMAND	6-26
6.16	GROUP COMMAND	6-27
6.17	INPUT COMMAND	6-29
6.18	INTERROGATE COMMAND	6-31
6.19	PRIMARY COMMAND	6-33
6.20	PRINT COMMAND	6-35
6.21	PRIORITY COMMAND	6-37
6.22	PUNCH COMMAND	6-38
6.23	QUEUE COMMAND	6-40
6.24	SCTASK COMMAND	6-42
6.25	SECONDARY COMMAND	6-44
6.26	SPOOLFILE COMMAND	6-45
6.27	SPVOLUME COMMAND	6-48
6.28	TRAILER COMMAND	6-50

APPENDIXES

A	SPL/32 COMMAND SUMMARY	A-1
B	SPL/32 COMMAND MESSAGE SUMMARY	B-1
C	SPL/32 VS. SPOOLER COMMAND CORRELATION	C-1
	C.1 INTRODUCTION	C-1
D	MESSAGE HANDLING	D-1
	D.1 INTRODUCTION	D-1
	D.1.1 Message Overview	D-1
	D.2 THE UNIFORM SPOOLER CONTROL INTERFACE (USCI)	D-1
	D.2.1 Establishing Communication With the Spooler	D-2
	D.2.2 Losing or Terminating Communications With Task	D-2
	D.2.2.1 Normal Termination of a Control Task	D-2

APPENDIXES (Continued)

D. 2. 3	Response Routing (Spooler to Task)	D-3
D. 2. 4	Dialogue Termination	D-3
D. 3	MESSAGE HEADER	D-4
D. 4	MESSAGE FORMATS	D-8
D. 4. 1	Task to SPL/32 Message Formats	D-8
D. 4. 1. 1	The Command Message	D-8
D. 4. 1. 2	Acceptance Response Message Format	D-10
D. 4. 1. 3	Request For Mnemonic Table Message	D-12
D. 4. 2	SPL/32 to Task Message Formats	D-14
D. 4. 2. 1	Service Response Message Format	D-14
D. 4. 2. 2	Buffer Set Format	D-15
D. 4. 3	Individual Command Service Response Messages	D-20
D. 4. 3. 1	ATTACH Command Service Record	D-20
D. 4. 3. 2	BLOCK Command Service Record	D-22
D. 4. 3. 3	CHECKPOINT Command Service Record	D-24
D. 4. 3. 4	CTASK Command Service Record	D-26
D. 4. 3. 5	FORM Command Service Record	D-28
D. 4. 3. 6	GLOBAL Command Service Record	D-30
D. 4. 3. 7	GROUP Command Service Record	D-32
D. 4. 3. 8	INPUT Command Service Record	D-34
D. 4. 3. 9	INTERROGATE Command Service Record	D-36
D. 4. 3. 10	PRIMARY Command Service Record	D-42
D. 4. 3. 11	PRIORITY Command Service Record	D-44
D. 4. 3. 12	QUEUE Command Service Record	D-46
D. 4. 3. 13	SCTASK Command Service Record	D-48
D. 4. 3. 14	SECONDARY Command Service Record	D-50
D. 4. 3. 15	SPVOLUME Command Service Record	D-52
D. 4. 4	Exception Response Messages	D-54
D. 4. 5	Attention Messages	D-59
D. 4. 6	Mnemonic Message Format	D-64
D. 4. 7	Authority Message Format	D-76
D. 5	MESSAGE RECOVERY PROCEDURES	D-68
E	SPL/32 COMMANDS PREPARED FORMAT	
E. 1	INTRODUCTION	E-1
E. 2	ATTACH COMMAND	E-2
E. 3	BANNER COMMAND	E-3
E. 4	BLOCK COMMAND	E-5
E. 5	CHANGE COMMAND	E-6
E. 6	CHECKPOINT COMMAND	E-8
E. 7	CREATE COMMAND	E-9
E. 8	CTASK COMMAND	E-10

APPENDIXES (Continued)

E. 9	DELETE COMMAND	E-11
E. 10	DETACH COMMAND	E-12
E. 11	DEVICE COMMAND	E-13
E. 12	END COMMAND	E-15
E. 13	FLUSH COMMAND	E-16
E. 14	FORM COMMAND (INCLUDE)	E-18
E. 15	FORM COMMAND (REMOVE)	E-20
E. 16	FORM COMMAND (MOUNT, MOUNT VERIFY)	E-21
E. 17	GLOBAL COMMAND	E-23
E. 18	GROUP COMMAND	E-24
E. 19	INPUT COMMAND (INCLUDE)	E-25
E. 20	INPUT COMMAND (REMOVE)	E-26
E. 21	INTERROGATE COMMAND	E-27
E. 22	PRIMARY COMMAND	E-29
E. 23	PRINT COMMAND	E-30
E. 24	PRIORITY COMMAND	E-32
E. 25	PUNCH COMMAND	E-33
E. 26	QUEUE COMMAND	E-35
E. 27	SCTASK COMMAND	E-37
E. 28	SECONDARY COMMAND	E-39
E. 29	SPVOLUME COMMAND	E-40
E. 30	TRAILER COMMAND	E-41
E. 31	SUMMARY	E-43

FIGURES

1-1	The Components of SPL/32	1-4
1-2	Control and Subcontrol Tasks Communications Flow	1-6

FIGURES (Continued)

2-1	Control Statement Sequences For Single and Stacked Card Files	2-5
3-1	Evolution of a Spool File	3-3
D-1	SPL/32 Message Header Record Format	D-5
D-2	Command Message Format	D-9
D-3	Acceptance Response Message Record Format	D-11
D-4	Request for Mnemonic Table Record Format	D-13
D-5	Service Response Message - Initial Buffer Format	D-16
D-6	Service Response Message - Continuation Buffer Format	D-18
D-7	ATTACH Command Service Message Record Format	D-21
D-8	BLOCK Command Service Message Record Format	D-23
D-9	CHECKPOINT Command Service Message Record Format	D-25
D-10	CTASK Command Service Message Record Format	D-27
D-11	FORM Command Service Message Record Format	D-29
D-12	GLOBAL Command Service Message Record Format	D-31
D-13	GROUP Command Service Message - Header Record and Data Record Formats	D-33
D-14	INPUT Command Service Message Record Format	D-35
D-15	INTERROGATE Command Service Message Header and Display Record Format	D-37
D-16	INTERROGATE Configuration Command Service Message Record Format	D-40
D-17	PRIMARY Command Service Message Record Format	D-43
D-18	PRIORITY Command Service Message Record Format	D-45
D-19	QUEUE Command Service Message Record Format	D-47
D-20	SCTASK Command Service Message Record Format	D-49
D-21	SECONDARY Command Service Message Record Format	D-51
D-22	SPVOLUME Command Service Message Record Format	D-53
D-23	EXCEPTION Response Message Format	D-55
D-24	SVC1 and 7 Attention Messages Record Format	D-60
D-25	General Attention Message Format	D-63
D-26	Mnemonic Message Format	D-65
D-27	Authority Message Format	D-67

TABLES

3-1	OPTION SYMBOLS DISPLAYED BY THE INTERROGATE COMMAND	3-6
5-1	AUTHORIZATION LEVELS AND ASSOCIATED SPOOLER COMMANDS	5-9
D-1	EXCEPTION MESSAGE ERROR CODES, MESSAGE TEXT, AND MESSAGE RELATED DATA SUMMARY	D-56
D-2	ERROR TEXT FIELD ENTRIES FOR SVC1 AND 7 ATTENTION MESSAGES	D-62
D-3	GENERAL ATTENTION MESSAGE FIELD ENTRIES	D-64
D-4	SPL/32 ERROR STATUS AND RECOVERY INFORMATION	D-69

TABLES (Continued)

E-1 SUMMARY OF COMMAND IDENTIFIERS (BYTE 54) AND
COMMAND FUNCTION IDENTIFIERS (BYTE 55) FOR
PREPARED COMMANDS

E-43

INDEX

IND-1

PREFACE

This manual describes the SPL/32 spooling system for Perkin-Elmer 32-bit computer systems. This information is useful to the system administrator and applications programmer for the initialization, administration and internal function of the SPL/32 spooling process. The commands section may prove useful to the system operator. Chapter 1 presents the spooling concept, summarizes the basic components of SPL/32, outlines the hardware and software required to run SPL/32 and details the initialization procedures for starting the spooling process. Chapter 2 explains input spooling and related commands. Chapter 3 explains the function of the spool queue, spool files and related commands. Chapter 4 presents output spooling and its associated commands. Chapter 5 presents the concept of task interfacing, the function of the uniform spooler control interface (USCI), and the concept of capabilities based on levels of authority and associated commands. Chapter 6 details each SPL/32 command. Complete format and optional parameter lists are presented for each command. Appendix A summarizes SPL/32 commands. Appendix B presents messages in a tabular format. Appendix C presents a comparison of SPL/32 commands and existing spooler commands in a tabular format. Appendix D presents message handling internals. Appendix E presents the SPL/32 commands in prepared format.

Revision F01 R00 of this manual includes changes made to the PRIORITY command in Chapters 4 and 6, and also adds new information about global pseudo devices in Chapter 4.

For information on the contents of all Perkin-Elmer 32-bit manuals, see the 32-Bit Systems User Documentation Summary.

1.5 REQUIREMENTS FOR SPL/32

The following sections detail the hardware, memory and software required to run SPL/32.

1.5.1 Hardware

The following minimum hardware requirements must be met in order to run SPL/32.

- One disk volume to store spool files and the spool queue.
- One terminal device to receive messages and enter SPL/32 commands.
- One card reader if input spooling is required.
- One printer for output of spooled print files.
- One card punch for output of spooled punch files.

SPL/32 will support peripheral devices listed in the Perkin-Elmer OS/32 System Generation (Sysgen) Reference Manual.

NOTE

At this time, the Perkin-Elmer Network (PENnet) and SPL/32 are incompatible. Users who want the capabilities of PENnet must configure their system with OS/32 Spooler support. See the OS/32 System Generation/32 (Sysgen/32) Reference Manual.

1.5.2 Memory Requirements

SPL/32 memory requirements are approximately 164.00kb plus:

- 32.50kb for each output device specified, and
- 25.00kb for each input device specified.

If no monitor control task is used (such as a multi-terminal monitor (MTM), an additional 35kb are required for the SPL/32 monitor (SPLC).

1.5.3 Software Requirements

SPL/32 will run on systems operating in an OS/32 Release 6.2 or higher environment with vertical forms control (VFC) and supervisor call (SVC) intercept options as required features.

SPL/32 and the original OS/32 spooler are mutually exclusive software products. Only one spooler can be active on a machine at a given time. The spooler to be used is selected at system generation (sysgen) time. Refer to the System Generation/32 (Sysgen/32) Reference Manual for details.

1.6 INITIALIZING SPL/32

The following sections outline the procedures involved in preparing and starting SPL/32.

1.6.1 Loading SPL/32

SPL/32 is provided in object format with the OS/32 operating system package. SPL/32 must be built as a universal task and must be loaded with the task identifier (taskid).SPLR.

Format:

```
LOAD .SPLR,SPL32,segsz increment
```

Parameters:

SPLR	is the name of the taskid for the spooler.
SPL32	is the name of the load file on disk.
segsz increment	is a decimal number, in kilobytes, specifying the amount of get storage area in the task's impure memory segment. If specified, this value overrides the GET and EXPAND values used when the task is linked.

Example:

```
LOAD .SPLR,SPOOLER.TSK,20
```

where 20 indicates that the spooler should be loaded into a larger segment. Complete information regarding the LOAD command is presented in the Perkin-Elmer OS/32 Operator Reference Manual.

SPL/32 can also be loaded using an SVC6 load-function option. For information regarding this procedure, see the Perkin-Elmer OS/32 Supervisor Call (SVC) Reference Manual.

1.6.2 Linking SPL/32

SPL/32 consists of five separate tasks: the main spooler, an intercept handler, an input task, an output task and a control task. Each task can be built as an overlaid task or as a nonoverlaid task. There is, however, one exception; the intercept task is always built as a nonoverlaid task. The SPL/32 package includes nine command files for building each of these tasks.

See the packaging information document for a list of file names and descriptions for SPL/32 task files and Link command files.

1.6.3 Starting the Spooler

Creating the desired initial spooler environment is accomplished through the use of the START command.

Format:

```
START [,COMMAND = fd]
```

Parameters:

fd specifies a file descriptor for the location of the commands to configure the spooling environment.

Functional Details:

These commands can be in a disk file, card deck, magnetic tape or can be entered from an interactive device. Any combination of valid SPL/32 commands (as listed in Chapter 6) can be used in the command file.

Examples:

START, COMMAND = M300:PARAM.DLK/118

File PARAM.DLK in account 118 on volume M300: contains the SPL/32 startup parameters.

START, COMMAND = CRD:

The card deck in card reader CRD: contains the SPL/32 startup parameters.

START, COMMAND = MAG1:

The tape mounted on tape drive MAG1: contains the SPL/32 startup parameters.

START, COMMAND = CON:

If the optional parameter [COMMAND = fd] is not included in the START command, the system console becomes, by default, the device where the initial SPL/32 commands for configuring the spooler environment will be entered.

1.6.3.1 Creating a START Command File

A START command file is simply a sequential text file that contains spooler commands. The commands in Chapter 6 are those available for use in the START command file. Commands must be presented one per line with the correct syntax. The end of a START command file is signified by the following for the media listed:

DEVICE	END ID
Cards	/* on the final card
Disk file	end of file (EOF) condition, or /*
Mag tape	EOF mark, or /*
Interactive device	Two consecutive carriage returns, or /*

CHAPTER 3 THE SPOOL FILE AND THE SPOOL QUEUE

3.1 INTRODUCTION

A spool file is a disk file created by SPL/32 to receive output data that will subsequently be sent to a physical output device. Spool files are allocated by the spooler in response to assignments to pseudo devices via an ASSIGN or SPOOLFILE command.

The spool queue is a disk file that contains status information regarding each file to be output via the spooler. These can be spool files (files allocated by the spooler) or user files (already existent files) temporarily assigned to the spooler for output via a PRINT, PUNCH or CHANGE command. The following sections explain the structure, function and use of these two components of SPL/32.

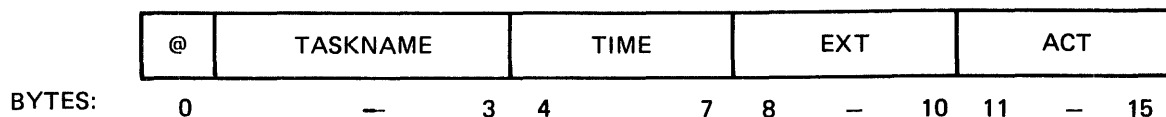
3.2 THE SPOOL FILE

In a spooled environment, output data assigned to a pseudo device via an ASSIGN or SPOOLFILE command is written to a disk file called a spool file. Each time an assignment is made to a pseudo device, OS/32 intercepts the supervisor call 7 (SVC7) generated and routes it to SPL/32. SPL/32 translates the assign and allocates a spool file. Spool files are allocated on multiple spool volumes (if present) in a rotating fashion so that no volume receives a disproportionate number of spool files. The index and data blocksize of spool files can be specified using the BLOCK command. The default value for both data and index blocksize is one 256-byte sector each. When changing the default blocksize, the value specified cannot exceed the values specified at system generation (sysgen) time or by the OS/32 SET BLOCKS command. The record length of spool files is determined by the spooler. When an assignment is made to a pseudo device, the spooler reads the device attributes of the physical device associated with the pseudo device. Spool files allocated as a result of the assignment to that pseudo device have the record length of the physical device. If multiple physical devices are associated with a single pseudo device, the record length of the device with the largest record length is used. Therefore, similar output devices should be grouped under one pseudo device. If a pseudo device has not been associated with any physical devices, the spooler assigns a default record length of 137 bytes.

3.2.1 Naming Spool Files

SPL/32 uses the following format when naming spool files:

3622-1



Fields:

@ is the 1-character at sign that precedes the filename and identifies the file as a spool file.

TASKNAME is the first three characters of the taskname on whose behalf the assignment is being made.

TIME is a 4-digit hexadecimal number representing the current time of day in seconds since midnight.

EXT is a 3-character extension used to distinguish between spool files if naming conflicts arise in the first eight characters of the spool file name. This field can contain any combination of the letters A through Z and the hexadecimal numbers 0 through F.

ACT is a 1- to 5-digit decimal number specifying the user account from which the assignment originated.

Example:

@TESAD94.000/119

The preceding example designates a spool file with the following information:

- assigned from a task called TESTRUN,
- file created at AD94 (00044436 seconds or approximately 12:20:36 PM),
- has the extension 000, and
- originated from account number 119.

When the lu assigned to a spool file is closed, the SVC7 is intercepted by OS/32 and routed to the spooler, which places the spool file name on the spool queue. Spool files can be placed on the queue in either the order of task priority, or in a first-in/first-out (FIFO) basis. The PRIORITY command is used to select the output method. Figure 3-1 illustrates the steps followed in the evolution of a spool file.

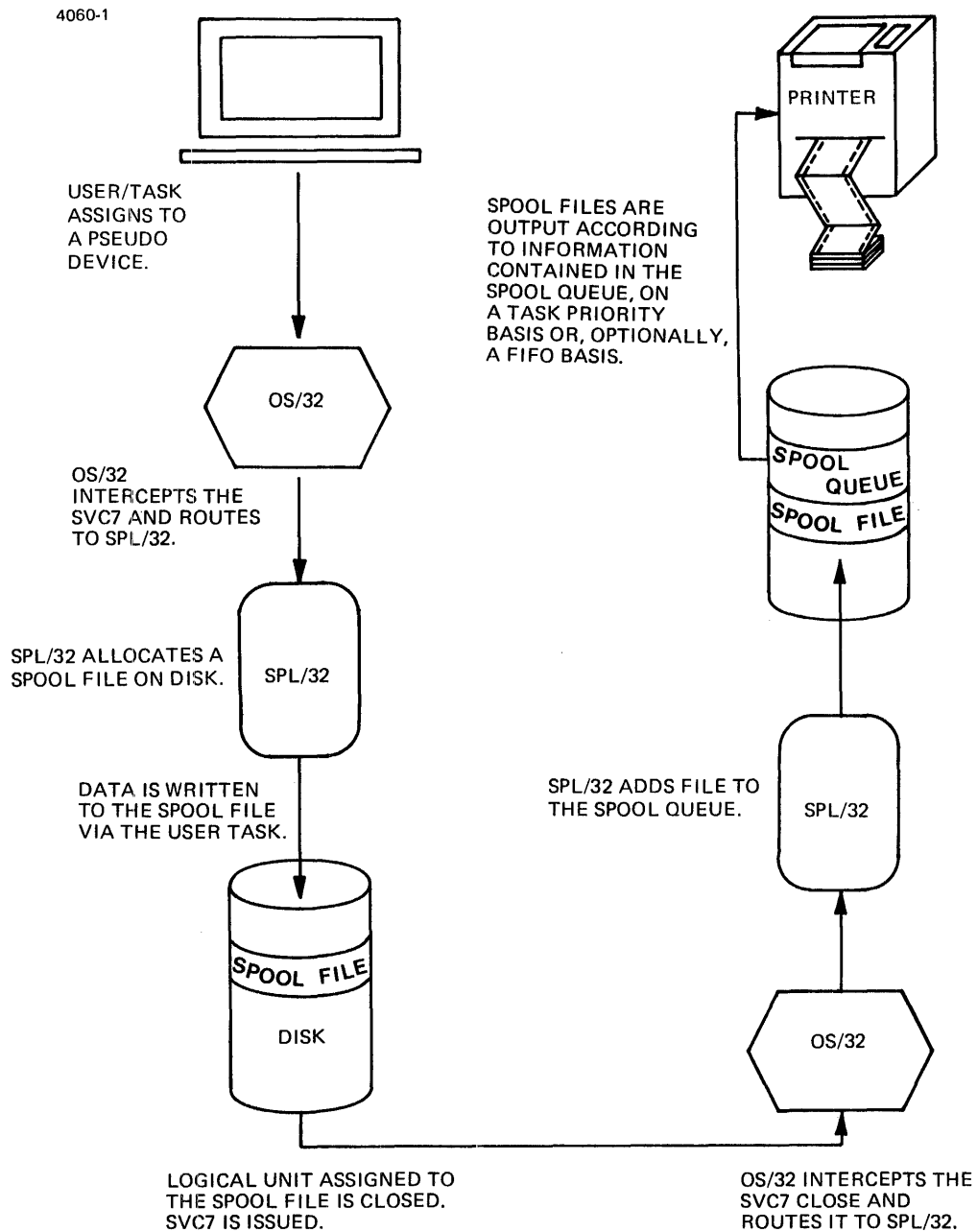


Figure 3-1 Evolution of a Spool File

3.3 THE SPOOL QUEUE

The spool queue is a disk file that contains an entry for each spool file on the system. Each entry contains origin and output requirements for each spool file.

3.3.1 Creating the Spool Queue

The spool queue is created at SPL/32 startup time in one of two ways:

1. The spool queue can be explicitly created, named, and allocated on a specified disk volume using the QUEUE command.

Example:

```
QUEUE M67B:SPLQUEUE.NEW
```

SPLQUEUE.NEW is allocated as the spool queue on disk volume M67B. If the disk volume is omitted, the spool queue is allocated on the first volume specified in a previous SPVOLUME command, or on the default system volume if no SPVOLUME command was previously entered. If no spool volumes were previously specified, and the system volume cannot be written to, the TEMP volume must be available or else the spool queue cannot be allocated.

2. The spool queue is implicitly created by the spooler if the QUEUE command is not used. The default spool queue, SPL32.QUE, is allocated on the first volume specified in a previous SPVOLUME command or, by default, on the system volume if no previous SPVOLUME command was entered. If no spool volumes were previously specified, and the system volume cannot be written to, the TEMP volume must be available or else the spool queue cannot be allocated.

The spool queue is a dynamic file with a default block size of 1 and a record length of 136.

3.3.2 Displaying the Spool Queue File Name

The spool queue currently in use can be determined by entering the QUEUE command without parameters. The following example represents this with a resultant display.

Example:

```
* QUEUE  
*  
* QUEUEFILE = M301:SPL32.QUE/0
```

3.3.3 Respecifying the Spool Queue

If a system failure occurs while the spooler is active or the spooler task is cancelled, spool file entries listed on the spool queue at the time of the failure will be held. After the system is restarted and the spool queue name is specified, the spool queue is read sequentially and rebuilt by copying valid records to a temporary file and then renaming the temporary file. The new queue is relinked in sequential order. If the default spool queue, SPL32.QUE, was in use prior to the failure, it is not necessary to respecify the spool queue because the spooler will automatically reassign the default spool queue.

3.3.4 Respecifying the Spool Queue While the Spooler is Active

SPL/32 allows the user to respecify the spool queue while the spooler is active. This is accomplished via the QUEUE command. When a new spool queue is specified, those entries on the old queue that were being output at the time of the interruption are copied by SPL/32 into the new spool queue. Spooling then resumes with these entries. Issuing the QUEUE command produces a reconstruction of the new queue.

3.3.5 Interrogating the Spool Queue

Entries on the spool queue and the status and configuration of the spooling environment can be displayed using the INTERROGATE command. This information can be selectively displayed according to the type of information desired. The following five sections detail the optional displays available.

3.3.5.1 Displaying Entries in the Spool Queue

The QUEUE parameter of the INTERROGATE command displays all entries in the spool queue relative to the user's authorization level.

Example:

```
INTERROGATE QUEUE
```

Resultant Display:

PSEUDO DEVICE	FILE NAME	FORMNAME	COP	PRI	OPTIONS
PD1:	M67B:@COPY003.000/118	STD	10	128	H I D
PD2:	M300:@TEST005.000/119	CHECKS	95	129	H I
PD3:	M300:@TEST004.001/120	STD	1	130	R I D
PD4:	M67B:TEST.PRT/100	FORMA	12	128	R I

The letters (or blanks) displayed in the options columns represent options specified in the PRINT/PUNCH request when each file was placed on the queue. These option symbols (or blanks) are explained in Table 3-1.

Note that the NOIMAGE, NODELETE, and NOVFC options are represented in the display by blanks in columns 2, 3, and 4 under the options header.

TABLE 3-1 OPTION SYMBOLS DISPLAYED BY THE INTERROGATE COMMAND

OPTION SYMBOL	DEFINITION	CHARACTER POSITION IN OPTION COLUMN	RESULT
H	Hold	1	Output request is held on the queue until released.
R	Release		Allows a held output request to become eligible for output.
I	Image	2	File is to be output as is.
	Noimage		File is to be output in formatted mode.
D	Delete	3	File is deleted after output.
	Nodelete		File is not deleted after output.
V	VFC	4	File is to be output with vertical forms control (VFC).
	NOVFC		File is to be output without VFC.

CHAPTER 4 OUTPUT SPOOLING

4.1 INTRODUCTION

The SPL/32 output spooling feature provides a variety of functions designed to handle the processing of output files. The printing or punching of a spool file is handled by the output spooler. The management of files on the spool queue waiting to be output is performed by the uniform spooler control interface (USCI). This interface is discussed in Appendix D.

The configuration of the output spooling environment can be tailored to optimize the benefits of spooling for each particular system. This can be done when the spooler is started or dynamically while the spooler is active.

The following features are provided as a function of output spooling:

- The number of output devices is dependent only on the amount of available memory.
- Capability of retaining a spooled output file after it is sent to a device.
- Capability of holding spooled files from output processing.
- The option to backspace, forward space, or rewind a file that is currently being output by the spooler, and then resume output.
- The option to produce up to 255 copies of an output file.
- The option to print informative header and trailer pages to identify output files.
- The capability of using preprinted forms and testing for form alignment before output.
- The capability to alter the output requirements of a file waiting to be output.
- The capability to alter the order in which files are output.
- The capability to control devices within the output spooling environment.
- The capability to quiesce the entire output spooling function or individual devices in an orderly fashion.

4.2 THE PSEUDO DEVICE

A pseudo device is a symbolic device reference whose function is to establish a correlation between a user/task and a physical output device. Within the output spooling environment, output files are directed to pseudo devices rather than directly to physical output devices. This provides two general benefits:

- The routing of output data is not limited to a single physical output device. A pseudo device can be associated with many physical output devices, and output can be directed by the spooler to the first physical device available.
- The configuration and relationships between pseudo and physical devices can be altered while the spooler is active.

When a task or user assigns a file for output to a pseudo device, the spooler translates the assign as a request to use a physical output device associated with the pseudo device.

Pseudo devices can be created and deleted while the spooler is active and can be logically associated with one or more physical output devices. When a physical device is associated with a pseudo device, it becomes assigned exclusively to the spooler and cannot be used by any other task except through the spooler. Pseudo devices are represented by 1- to 4-character alphanumeric identifiers; the first character must be alphabetic, the remaining may be alphanumeric.

The following sections detail the process of creating, configuring and using pseudo devices during output spooling.

4.2.1 The Global Pseudo Device

The global pseudo device is the default system pseudo device. If a pseudo device is not specified in a PRINT or PUNCH request, the file will automatically be assigned to the global pseudo device. This enables a user to request the output of a file even though the user is not aware of the specific pseudo device names present in the spooling environment or is not concerned with having a file output on a specific device.

When the spooler is started, a global pseudo device called PR: is the default pseudo device for the output spooling environment. The global pseudo device PR: must be created by the user. The system will not automatically create one. All requests to output files that do not explicitly specify a pseudo device are assigned to the global pseudo device, which is associated with all physical output devices in the spooling environment. The global pseudo device name can be changed via the GLOBAL command. Primary authorization is required to change the global pseudo device.

Example:

```
GLOBAL PR9:
```

The global pseudo device name is changed from PR: to PR9:.

4.2.2 Creating Pseudo Devices in the Spooling Environment

Pseudo devices can be created in the spooling environment in two ways:

- explicitly, via the CREATE command, or
- implicitly, via the ATTACH command.

The CREATE command is used to specify new pseudo devices in the spooling environment.

Example:

```
CREATE PD1:,PD2:,PD3:
```

PD1:, PD2:, and PD3: are to be recognized as valid pseudo device names. Up to 20 pseudo devices can be specified in a single CREATE command.

If the pseudo device name specified in an ATTACH command does not exist in the spooling environment, SPL/32 will create it and the following message is displayed:

```
IMPLICITLY CREATED PSEUDO DEVICE: pd:
```

Pseudo devices can be created when the spooler is started by including the CREATE command in the start command file, or they can be created while the spooler is active. Once a pseudo device has been created, it must then be associated with physical output devices. If a pseudo device is not associated with a physical output device, spool files assigned to that pseudo device will remain on the spool queue and will not be output.

4.2.3 Associating Pseudo Devices with Physical Devices

The process of associating a pseudo device with physical output devices is accomplished via the ATTACH command.

Example:

```
ATTACH PDEVICE = PD1: RDEVICE = PRT1:
```

The pseudo device PD1: is logically associated with the physical print device PRT1:. If pseudo device PD1: does not exist, SPL/32 will create it and then notify the initiator, as detailed in the previous section.

Multiple physical devices can also be associated with a single pseudo device.

Example:

```
ATTACH PDEVICE = PD1: RDEVICE = PRT2:,PRT3:,PRT4:
```

The pseudo device PD1: is logically associated with the physical print devices PRT2:, PRT3:, and PRT4:. When a pseudo device is associated with multiple physical devices, the output files assigned to that pseudo device can be sent to any one of the multiple physical devices based on immediate availability.

When a pseudo device is associated with multiple physical devices, the device attributes for the pseudo device are defined by the last physical device attached to the pseudo device. The logical record length for the pseudo device is determined by the physical device with the largest logical record length associated with the pseudo device via an ATTACH command. Therefore, insure that the physical devices associated with a single pseudo device have like device attributes.

NOTE

If a pseudo device is not associated with any physical output device, output assigned to that pseudo device will not be output and will remain on the spool queue until the pseudo device is associated with a physical device or the files are flushed.

Example:

```
CHECKPOINT ON,80
```

This example turns the CHECKPOINT feature on and specifies that a checkpoint is to be performed after each 80 records output. The number specified represents the largest number of records that would have to be reprinted if a system failure occurs. Once the global CHECKPOINT feature has been turned on, as the above example shows, a user can request that a file be checkpointed by specifying the CHECKPOINT option in the PRINT, PUNCH, or CHANGE command.

Examples:

```
PRINT REPORT.NO1/119,DEVICE=PR2:,CHECKPOINT
```

```
PUNCH TEST32.CAL/120,DEVICE=PR3:,CHECKPOINT
```

```
CHANGE SCAN32.CAL/119,DEVICE=PD1:,CHECKPOINT
```

The files in these examples will be checkpointed when output, as long as the global checkpoint feature is on. If the global checkpoint feature is off, checkpointing will not be performed. Therefore, it is advisable to make sure the global checkpoint feature is on before attempting to use the CHECKPOINT feature. Entering the CHECKPOINT command without parameters will result in a display of the checkpoint value if global checkpointing is on. If no value is displayed, global checkpointing is off.

4.6.1 Turning Checkpointing Off

Global checkpointing is turned off via the OFF parameter of the CHECKPOINT command.

Example:

```
CHECKPOINT OFF
```

The global checkpoint feature is off and checkpointing cannot be performed. Checkpointing can also be turned off at the user-level if the global checkpoint feature is on. This is accomplished by specifying the NOCHECKPOINT parameter in a PRINT, PUNCH, ASSIGN, SPOOLFILE, or CHANGE command.

Example:

```
PRINT TESTPROG.CAL/119,DEVICE=PR:,NOCHECKPOINT
PUNCH STATTEST.CAL/120,DEVICE=PD2:,NOCHECKPOINT
CHANGE REPORT.TEX/118,DEVICE=PD7:,NOCHECKPOINT
```

4.7 QUEUE HANDLING

| Spool files are output either on a priority basis or a
| first-in/first-out (FIFO) basis. Each file assigned for output
| is given a priority value, either implicitly or explicitly.

| For priority handling, the assigned priority number is used by
| the spooler to determine the order in which files will be output;
| files with the lower priority number are output first. The
| priority of a spool file is explicitly assigned in the PRIORITY=
| parameter of a PRINT, PUNCH or CHANGE request. When this
| priority is not specified, the file is given the default priority
| of the system. When a file is placed on the spool queue, it has
| two priorities associated with it:

- a global priority, and
- a user priority.

The global priority is the priority of the file with respect to all other spool files. The user priority is the priority of the file with respect to all spool files initiating from the same user and having the same global priority.

When a PRINT, PUNCH or CHANGE command is issued from a unit with primary or secondary authorization, the PRIORITY= keyword refers to the global priority of the file. When a PRINT, PUNCH or CHANGE command is issued from a unit with user authorization, the PRIORITY= keyword refers to the user priority of the file. See Chapter 5 for a discussion of authorization levels.

| FIFO queue handling results in increased spooler throughput when
| many files are on the queue. Files placed on the queue are still
| given a priority value so priority queue handling can be resumed
| at anytime.

4.7.1 Changing the Global Default Priority

The global default priority value is changed via the PRIORITY command.

Example:

PRIORITY 132

This example sets the global default priority to 132. If the PRIORITY command is omitted at spooler startup, the default priority is set at 128. The priority can be changed while the spooler is active, but only primary authorization can change the global default priority.

4.7.2 Altering the Output Priority of a Spool File

SPL/32 provides the capability to alter the output priority of spool files waiting on the spool queue. This is accomplished via the PRIORITY= keyword of the CHANGE command.

For example, if the file PAYROLL.CHK in account 119 was residing on the spool queue waiting to be printed, the file has a priority of 118. To alter that priority the user enters the following CHANGE command:

Example:

CHANGE PAYROLL.CHK/119,PRIORITY=120

The output priority of the file PAYROLL.CHK is altered to 120.

The degree to which the output priority of a file can be altered depends upon the authorization level of the CHANGE command. The following general conventions are used:

- Primary authorization allows the output priority of a file to be altered relative to all other files in the spooling environment.
- Secondary authorization allows the output priority of a file to be altered relative to all other files in the spooling environment that were placed on the spool queue with the same secondary authorization.
- User authorization allows a user to alter the output priority of a file relative to any other files placed on the queue by that user.

6.4 BLOCK COMMAND

The BLOCK command is used to display the current physical data block size and index block size or to set the default values for these two parameters.

Format:

```
BLOCK blksize/indexsize
```

Parameters:

blksize is a decimal number indicating the data block size in 256-byte sectors.

indexsize is a decimal number indicating the index block size in 256-byte sectors.

Functional Details:

The BLOCK command can be used to display or set the data block size and index block size, in 256-byte sectors, for buffering and debuffering operations on spool files. To display the current values, BLOCK is entered without any additional parameters. To set the default block size (data and/or index) or change the current values, the blksize/indexsize parameters are used with the BLOCK command. When setting new values, do not exceed the maximum block size established at system generation (sysgen). The default value is one sector. Primary control unit authorization is required to set the data and index block sizes. Secondary and user-level control unit authority may display the current data and index block sizes.

Examples:

```
BLOCK 5/3
```

The default data block size for the allocation of spool files is 5, the default index blocking factor is 3.

```
BLOCK
```

The current data block size and index block size will be displayed.

CHANGE

6.5 CHANGE COMMAND

The CHANGE command is used to modify the characteristics of a spool file that is currently residing on the spool queue.

Format:

```
CHANGE fd, DEVICE=pseud dev: [,COPIES=n] [,FORM=formname] [,PRIORITY=n]
  [ {IMAGE} ] [ {NOIMAGE} ] [ {DELETE } ] [ {CHECKPOINT } ] [ {HOLD } ]
  [ {VFC } ] [ {NOVFC } ] [ {NODELETE} ] [ {NOCHECKPOINT} ] [ {RELEASE} ]
```

Parameters:

- fd is the file descriptor (name) of the file to be modified.
- DEVICE allows the pseudo device specification to be changed.
- COPIES allows a change in the number of copies to be output.
- FORM allows preprinted forms specification to be changed.
- PRIORITY allows the output priority of a file to be altered. The capability of a user to alter the priority of a file is dependent upon the authorization level of that user. Primary authorization can change the priority of any file on the spool queue. Secondary authorization can change the priority of any file on the spool queue with the same secondary authorization identifier. User authorization can change the priority of files that the user placed on the spool queue.
- IMAGE or VFC allows a change in the output format of a file; either output in IMAGE (as is) or with VFC (vertical forms control). These parameters are mutually exclusive.

6.14 FORM COMMAND

The FORM command has three functions:

- to declare valid form names,
- to show which forms to mount on each of the printers, and
- to display valid form names.

Format:

FORM	{	INCLUDE frm1	[<table style="border: none; border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">PLENGTH=p</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">BMARGIN=b</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">TMARGIN=t</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">PWIDTH=pw</td> <td style="padding: 0 5px;">)</td> </tr> </table>	(PLENGTH=p)	(BMARGIN=b)	(TMARGIN=t)	(PWIDTH=pw)]	[..., frm4]	[<table style="border: none; border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">PLENGTH=p</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">BMARGIN=b</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">TMARGIN=t</td> <td style="padding: 0 5px;">)</td> </tr> <tr> <td style="padding: 0 5px;">(</td> <td style="padding: 0 5px;">PWIDTH=pw</td> <td style="padding: 0 5px;">)</td> </tr> </table>	(PLENGTH=p)	(BMARGIN=b)	(TMARGIN=t)	(PWIDTH=pw)]	}
(PLENGTH=p)																																
(BMARGIN=b)																																
(TMARGIN=t)																																
(PWIDTH=pw)																																
(PLENGTH=p)																																
(BMARGIN=b)																																
(TMARGIN=t)																																
(PWIDTH=pw)																																
		REMOVE frm1	[frm2]	[..., frm5]																										
		MOUNT frm1	[VERIFY=n	,	dev1:]	[..., dev15:]																								

Parameters:

INCLUDE	is used to specify a valid form name (1- to 8-characters long) for each unique paper type. The default form name is STD. Up to four forms can be included in a single command line.
PLENGTH	is the number of lines per page for the form specified in the INCLUDE command. Default is 66 lines per page.
BMARGIN	is the number of blank lines to be left at the bottom of the page. Default is 0 lines or no margin.
TMARGIN	is the number of blank lines to be left at the top of the page. Default is 0 lines or no margin.
PWIDTH	is the number of print positions horizontally across the page.

REMOVE causes a currently valid form name to be removed from the list of valid form names.

MOUNT this option of the FORM command is used to associate a particular form type with a particular printer. Subsequent output that requires that particular form can then be directed to the device that has the form mounted.

VERIFY this option of the MOUNT parameter allows the user to specify that forms alignment is to be verified. The user can specify (in number of pages) at what point alignment should be verified. Output of the first file on the device will be suspended after n pages have been output. The operator can then verify forms alignment and continue the device, or realign the form, rewind to beginning of file, and resume output.

devl:,..., is the device referred to by the FORM MOUNT
devl5: command. Up to 15 devices can be specified in a single command line.

Functional Details:

The FORM command has three distinct functions dealing with printer paper control. The INCLUDE, REMOVE and MOUNT keywords denote the functions available. INCLUDE enables new forms to be added to the system and format of these forms can be defined. REMOVE enables currently valid forms to be removed from system availability. MOUNT enables form mounting requests to be issued and verification of alignment of these forms to be requested. The FORM command is only valid when issued from a primary control unit or from a secondary control unit for the output device in question. Issuing a FORM command with no parameters results in forms information for all devices in the spooler environment to be displayed. Whenever a file is placed on the spool queue that requires a form not currently mounted on any device, the message:

REQUEST PENDING FOR FORM: formname

is generated.

6.21 PRIORITY COMMAND

The PRIORITY command is used to set the default output priority, change to FIFO handling, or display the current default output priority of files on the queue.

Format:

PRIORITY [{ (number) }
(FIFO)]

Parameters:

number is a number from 1 to 255 that becomes the default priority for the spooling environment.

FIFO specifies that queue searching is to be performed on a first-in/first-out basis.

Functional Details:

Primary control unit authority is required to change the default priority with the PRIORITY command. Secondary control units and user control units can display the current default priority by entering the PRIORITY command with no option. If the PRIORITY command is entered without a new value, the current default output priority is displayed.

Example:

PRIORITY 240

The default priority for print or punch requests becomes 240.

PUNCH

6.22 PUNCH COMMAND

The PUNCH command is used to direct a user file to be output on a card punch device.

Format:

```
PUNCH fd[,DEVICE=pseud dev:] [,COPIES=n][,FORM=form name]  
[ {DELETE } ] [ {CHECKPOINT } ] [, PRIORITY=p]  
[ {NODELETE } ] [ {NOCHECKPOINT } ]
```

Parameters:

fd is the name of the file to be punched. This is a required parameter.

DEVICE= identifies the pseudo device the file is assigned to for output. If omitted, the default punch location is determined by spooler.

COPIES= n is the number of copies to be output. n may range from 1 through 255. If this parameter is omitted, one copy is output.

FORM= specifies that the file be output onto a special preprinted or prepunched card. If the specified form name was not previously declared using a FORM command, an error message is generated and the punch request is output using the default form name STD.

DELETE specifies that the file be deleted after output. This is the default for files created by the spooler through an ASSIGN or SPOOLFILE command.

NODELETE specifies that the file is not to be deleted after output. This is the default option for standard user files output by the spooler.

CHECKPOINT CHECKPOINT turns on checkpointing for the specified punch file. This is the default condition.

APPENDIX A
SPL/32 COMMAND SUMMARY

ATTACH PDEVICE=dev RDEVICE=dev1 [,dev2 ... ,dev19]

BANNER [pseudo dev:] [{DOUBLE }] [{SINGLE }] [{NONE }] [,CHARACTER=character, [fd]]

BLOCK blksize/indexsize

CHANGE fd, DEVICE=pseud dev: [,COPIES=n] [,FORM=formname] [,PRIORITY=n]
[{IMAGE }] [{NOIMAGE }] [{DELETE }] [{CHECKPOINT }] [{HOLD }]
[{YFC }] [{NOYFC }] [{NODELETE }] [{NOCHECKPOINT }] [{RELEASE }]

CHECKPOINT [{QN,N }]
[{QEF }]

CREATE pseud dev1: [, ...pseud dev20:]

CTASK [{INCLUDE task1 ,...,task10 }]
[{REMOVE task2 ,...,task10 }]

DELETE [{pseud dev1: ,...,pseud dev20: }]
[{real dev1: ,...,real dev20: }]

DETACH PDEVICE=pseud dev RDEVICE=real dev1: [, ...real dev19:]

DEVICE {

 pseudo device name [{ (RESTRICT) }]

 real device name [{ (FLINE=n }

 { (FPAGE=n }

 { (BLINE=n }

 { (BPAGE=n }

 { (REWIND }

 { (CONTINUE) }]

 [, { (SUSPEND { (IMMEDIATE) } }]

 [, { (DELAYED) }]
 }

END [{ (DELAYED) }]

 [(IMMEDIATE)]

FLUSH [{ fd [DELETE] }]

 [{ real device: }]

FORM {

 INCLUDE frm1 [{ (PLENGTH=p) }] [..., frm4] [{ (PLENGTH=p) }]

 [{ (BMARGIN=b) }] [{ (BMARGIN=b) }]

 [{ (TMARGIN=t) }] [{ (TMARGIN=t) }]

 [{ (PWIDTH=pw) }] [{ (PWIDTH=pw) }]

 REMOVE frm1 [, frm2] [..., frm5]

 MOUNT frm1 [VERIFY=n , dev1:] [..., dev15:]
 }

GLOBAL pseud dev:

GROUP PDEVICE=pseud dev1 {

 [INCLUDE origin id1 , ..., origin id19]

 [REMOVE origin id1 , ..., origin id19]
 }

INPUT [{ (CTASK=taskname [INCLUDE cr1:] [..., cr8:]) }]

 [{ REMOVE cr1: [..., cr8:] }]

INTERROGATE $\left[\begin{array}{l} \text{QUEUE} \\ \text{RD pd:} \\ \text{FORM formname} \\ \text{ED filename} \\ \text{CONFIGURATION} \end{array} \right]$

PRIMARY ctask/aid

PRINT fd [,DEVICE=pseud dev:] [,COPIES=n] [,FORM=formname] $\left[\begin{array}{l} \text{IMAGE} \\ \text{VFC} \end{array} \right]$
 $\left[\begin{array}{l} \text{NOIMAGE} \\ \text{NOVFC} \end{array} \right]$ $\left[\begin{array}{l} \text{NODELETE} \\ \text{DELETE} \end{array} \right]$ $\left[\begin{array}{l} \text{NOCHECKPOINT} \\ \text{CHECKPOINT} \end{array} \right]$ [, PRIORITY=p]

PRIORITY $\left[\begin{array}{l} \text{number} \\ \text{FIFO} \end{array} \right]$

PUNCH fd [,DEVICE=pseud dev:] [,COPIES=n] [,FORM=form name]
 $\left[\begin{array}{l} \text{DELETE} \\ \text{NODELETE} \end{array} \right]$ $\left[\begin{array}{l} \text{CHECKPOINT} \\ \text{NOCHECKPOINT} \end{array} \right]$ [, PRIORITY=p]

QUEUE fd

SCTASK $\left\{ \begin{array}{l} \text{CTASK=taskname [INCLUDE task1 ,task7]} \\ \text{REMOVE task1 [,...,task8]} \end{array} \right\}$

SECONDARY ctask/aid,pd:

SPOOLFILE lu & lul,pseud dev,FORM=formname $\left[\begin{array}{l} \text{VFC} \\ \text{IMAGE} \end{array} \right]$
 $\left[\begin{array}{l} \text{NOIMAGE} \\ \text{NOVFC} \end{array} \right]$ $\left[\begin{array}{l} \text{CHECKPOINT} \\ \text{NOCHECKPOINT} \end{array} \right]$ [,COPIES=n] $\left[\begin{array}{l} \text{HOLD} \\ \text{RELEASE} \end{array} \right]$
[,BLOCK= blocksize/indexsize] $\left[\begin{array}{l} \text{DELETE} \\ \text{NODELETE} \end{array} \right]$ [, PRIORITY=p]

SPVOLUME [{ INCLUDE volumel [, ..., volume20] }
 { REMOVE volumel [, ..., volume20] }]

TRAILER [pseudo dev:] [{ DOUBLE }
 [real dev:] [SINGLE } [, CHARACTER=character] [,fd]
 [NONE]

APPENDIX B
SPL/32 COMMAND MESSAGE SUMMARY

CANNOT ASSIGN DISPLAY MESSAGE FILE |

The display message file (SPLDISPL.MES) cannot be assigned. |

CANNOT COMMUNICATE WITH CTASK: ctask

The specified control task is not present in the spooling environment or has gone to end of task.

CORRUPTED QUEUE |

The spooler cannot read the spool queue because the queue entry links have been corrupted. |

DEVICE ACTIVE: device name given

A command requiring the specified device to be in a suspended state was issued to an active device.

DEVICE INACTIVE: device name given

A command requiring the specified device to be in an active state was issued to an inactive device.

DEVICE NOT INCLUDED: device name

The specified device is not in the spooler environment.

DUPLICATE DEVICE: dev:

The specified device name already exists in the spooler environment.

DUPLICATE TASK NAME: task name

The specified task already exists in the spooling environment.

DUPLICATE UNIT NAME: unit id

The specified unit name already exists and cannot be used again.

DUPLICATE VOLUME: volume name

The specified volume is already assigned for the allocation of spool files.

FILE DESCRIPTOR ERROR: fd

The file descriptor was syntactically incorrect.

FILE NOT FOUND: fd

The specified file was not found on the spool queue or in the disk directory. Check the filename.

FORM NOT INCLUDED: formname

The specified form has not been declared a valid form prior to use in a PRINT/PUNCH request.

INVALID AUTHORIZATION-ID: authorization identifier

The authorization ID specified in a PRIMARY or SECONDARY command is invalid.

INVALID AUTHORIZATION LEVEL: authorization level

The command attempted is not available to the specified authorization level.

INVALID COMMAND: command

Command is syntactically incorrect or is not a valid SPL/32 command.

INVALID CONTROL CARD

A card deck with an incorrect control card was read.

INVALID DATA BLOCK SIZE: blocksize

The data block size specified exceeds the maximum established at system generation (sysgen) time.

INVALID DEVICE: dev:

The specified device is not assigned to the spooler.

INVALID FORM: form

The specified form is not a valid form name.

INVALID INDEX BLOCK SIZE: i

The index block size specified exceeds the maximum established at sysgen time.

INVALID KEYWORD: keyword

Keyword is syntactically incorrect or is not valid with the associated command.

INVALID PARAMETER: parameter

The specified parameter is not valid with the associated command.

INVALID PRIORITY: priority

The requested priority exceeds the maximum set at Link time, or the requested priority is outside the range of 10 to 249.

INVALID SEPARATOR: separator

The specified separator is invalid or is syntactically incorrect.

INVALID TASK: taskname

The specified taskname does not exist.

INVALID VOLUME: vol:

The specified volume is syntactically incorrect.

I/O TASK WAS CANCELLED: I/O taskname

The specified input/output (I/O) task has been cancelled.

MISSING CONTROL CARD

A card deck without an initial control card was read.

MISSING PARAMETER

A required parameter is missing from the command.

NO AVAILABLE LUS

There are no logical units available for use.

NO SPOOL VOLUMES

There are no spool volumes specified in the spooling environment.

NO SUBCONTROL TASKS

There are no subcontrol tasks in the spooling environment.

NO INPUT DEVICES

There are no spooled card readers in the spooling environment.

NO QUEUE ENTRIES

There are presently no files in the spool queue.

NO GROUPS

There are currently no groups specified in the spooling environment.

| OS DOES NOT SUPPORT SPL32

| The operating system was configured without support for
| SPL/32. Resysgen the operating system to include SPL/32
| support.

PRIMARY CTASK WAS CANCELLED: taskname

The primary control task has been cancelled.

PARAMETER CONFLICT: parameter1, parameter2

Two mutually exclusive parameters are present.

PRIVILEGE ERROR

The access privilege mnemonic was syntactically incorrect.

TASK NOT INCLUDED: taskname

The specified task was not included in the spooling environment.

UNIT NOT INCLUDED: unit id

The specified unit was not included in the grouping request of a GROUP command.

VERIFY COMPLETED ON: device name

A request to verify forms alignment has been completed on the specified device.

VOLUME NOT INCLUDED: volume name

The specified volume was not included as a spool volume.

VOLUME NOT IN SYSTEM: vol:

The specified volume is not present in the system and therefore cannot be assigned as a spool volume.

TABLE D-1 EXCEPTION MESSAGE ERROR CODES, MESSAGE TEXT, AND MESSAGE RELATED DATA SUMMARY (Continued)

ERROR CODE (DECIMAL)	MESSAGE TEXT	MESSAGE RELATED DATA
68	NO DEVICES INCLUDED	-
69	OS DOES NOT SUPPORT SPL/32	-
70	PLEASE CONTINUE	-
71	NO SECONDARY AUTHORI- ZATION	-
72	NO FILES TO BE PRINTED	-
73	INVALID BANNER/ TRAILER PARAMETER:	invalid parameter
74	INVALID COPIES:	invalid # of copies
75	NO QUEUE ENTRIES, NO DEVICES	-
77	CORRUPTED QUEUE	-
78	CANNOT ASSIGN DISPLAY MESSAGE FILE	--
79	INTERCEPT TASK WAS CANCELLED	--

D.4.5 Attention Messages

There are two types of attention messages:

- SVC1 and 7 error messages
- General attention messages

The format of the SVC error messages is presented in Figure D-24.

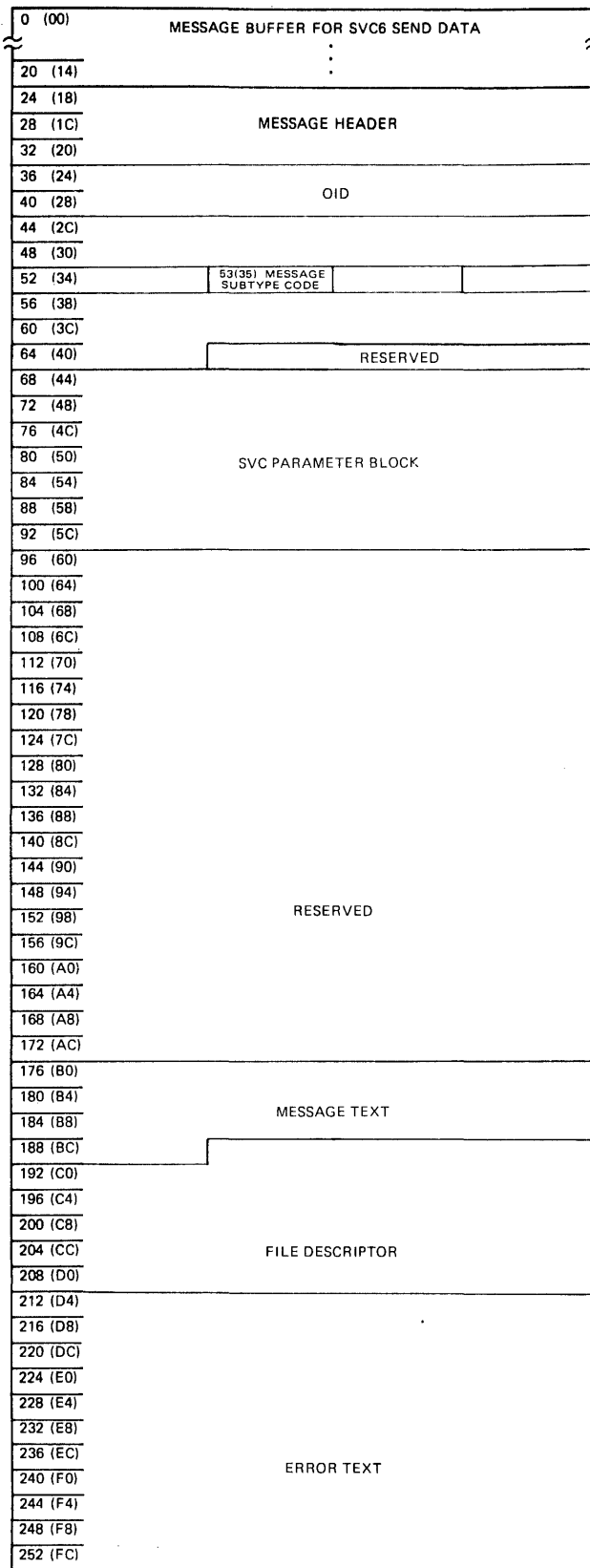


Figure D-24 SVC1 and 7 Attention Messages Record Format

Account	contains the account number of the file to be printed.
Formname	contains the name of the form required for output.
Pseudo device	contains the name of the pseudo device the output is to be directed to.
Options	contains one or a combination of the following settings: Y'0001' = VFC (if NOVFC set) Y'0002' = NOVFC (if VFC set) Y'0004' = IMAGE (if NOIMAGE set) Y'0008' = NOIMAGE (if IMAGE set) Y'0010' = DELETE (if NODELETE set) Y'0020' = NODELETE (if DELETE set) Y'0040' = CHECKPOINT (if NOCHECKPOINT set) Y'0080' = NOCHECKPOINT (if CHECKPOINT set) Y'0100' = HOLD (if RELEASE set) Y'0200' = RELEASE (if HOLD set)
Number of copies	contains the number of copies to be printed.
Priority	contains the output priority of the file to be printed.

E.24 PRIORITY COMMAND

Command mnemonic: PRIORITY

Command identifier (byte 54 of message header): 18(12)

Command function identifier (byte 55 of the message header):

X'00' = to display

X'03' = to specify

Record format:

4023-1

0 (00)	PRIORITY
4 (04)	FIFO
8 (08)	
12 (C)	
16 (10)	
20 (14)	
24 (18)	
28 (1C)	
32 (20)	
36 (24)	
40 (28)	BLANKS
44 (2C)	
48 (30)	
52 (34)	
56 (38)	
60 (3C)	
64 (40)	
68 (44)	
72 (48)	
76 (4C)	

Fields:

Priority contains the default priority value.

FIFO X'00' = No FIFO
X'01' = FIFO

MANUAL UPDATE PACKAGE COVER SHEET

MANUAL TITLE: SPOOLER (SPL/32) SYSTEM ADMINISTRATION
Reference Manual

PUBLICATION
NUMBER:

48-056

OLD REVISION LEVEL: F00 R00

NEW REVISION LEVEL: F01 R00

This package of affected pages updates the current version of the subject manual. New features, as well as changes, deletions and additions to information in this manual are indicated by change bars in the page margins. Please discard the indicated old pages and replace them with the supplied new pages.

OLD PAGES	NEW PAGES
Title Sheet/Disclaimer, F00 R00	Title Sheet/Disclaimer, F01 R00
Sheets i through ix, F00 R00	Sheets i through ix, F01 R00
Sheet xi, F00 R00	Sheet xi, F01 R00
Sheet 1-7, F00 R00	Sheet 1-7, F01 R00
Sheet 1-8, F00 R00	Sheet 1-8, F01 R00
Sheet 1-9, F00 R00	Sheet 1-9, F01 R00
Sheet 1-10, F00 R00	Sheet 1-10, F00 R00
Sheet 3-1, F00 R00	Sheet 3-1, F01 R00
Sheet 3-2, F00 R00	Sheet 3-2, F01 R00
Sheet 3-3, F00 R00	Sheet 3-3, F01 R00
Sheet 3-4, F00 R00	Sheet 3-4, F00 R00
Sheet 3-5, F00 R00	Sheet 3-5, F01 R00
Sheet 3-6, F00 R00	Sheet 3-6, F00 R00
Sheet 4-1, F00 R00	Sheet 4-1, F00 R00
Sheet 4-2, F00 R00	Sheet 4-2, F01 R00
Sheet 4-3, F00 R00	Sheet 4-3, F01 R00
Sheet 4-4, F00 R00	Sheet 4-4, F00 R00
Sheet 4-15, F00 R00	Sheet 4-15, F00 R00
Sheet 4-16, F00 R00	Sheet 4-16, F01 R00
Sheet 4-17, F00 R00	Sheet 4-17, F01 R00
Sheet 6-7, F00 R00	Sheet 6-7, F00 R00
Sheet 6-8, F00 R00	Sheet 6-8, F01 R00
Sheet 6-23, F00 R00	Sheet 6-23, F01 R00
Sheet 6-24, F00 R00	Sheet 6-24, F01 R00
Sheet 6-37, F00 R00	Sheet 6-37, F01 R00
Sheet 6-38, F00 R00	Sheet 6-38, F00 R00

OLD PAGES	NEW PAGES
Sheet A-1, F00 R00	Sheet A-1, F01 R00
Sheet A-2, F00 R00	Sheet A-2, F00 R00
Sheet A-3, F00 R00	Sheet A-3, F01 R00
Sheet A-4, F00 R00	Sheet A-4, F00 R00
Sheets B-1 through B-5, F00 R00	Sheets B-1 through B-5, F01 R00
Sheet D-59, F00 R00	Sheet D-59, F01 R00
Sheet D-60, F00 R00	Sheet D-60, F00 R00
Sheet E-31, F00 R00	Sheet E-31, F00 R00
Sheet E-32, F00 R00	Sheet E-32, F01 R00

PERKIN-ELMER
Technical Systems Division

DOCUMENTATION CHANGE NOTICE

The purpose of this documentation change notice (DCN) is to provide a quick and efficient way of making technical changes to manuals before they are formally updated or revised. The manual affected by these changes is:

48-056 F00 R00 SPL/32 Spooler System Administration Reference Manual

- Page 1-7

After the last paragraph in Section 1.5.1, please insert:

NOTE

At this time, the Perkin-Elmer Network (PENnet) and SPL/32 are incompatible. Users who want the capabilities of PENnet must configure their system with OS/32 Spooler support. See the OS/32 System Generation/32 (SYSGEN/32) Reference Manual.

- Page B-1

Before the first message, please insert the following:

CANNOT ASSIGN DISPLAY MESSAGE FILE

The display message file (SPLDISPL.MES) cannot be assigned.

- Page B-1

Immediately after the above inserted message, please insert:

CORRUPTED QUEUE

The Spooler cannot read the spool queue because the queue entry links have been corrupted.

- Page B-4

After the sixth message, please insert:

OS DOES NOT SUPPORT SPL32

The operating system was configured without support for SPL/32. Re-sysgen the operating system to include SPL/32 support.

- Page D-59

Message code 69. Please change this message to read:

OS DOES NOT SUPPORT SPL32

- Page D-59

After message code 75, please insert the following messages:

77 CORRUPTED QUEUE

78 CANNOT ASSIGN DISPLAY MESSAGE FILE