



SPIN-X/ CENTRAL  
Reference Guide  
for the Xerox LPS Feature  
Version 4R5

---

Corsair Technology Inc.  
Atlanta, Georgia 30303

Document revision date - May 12, 1997

Central Document Version 4R5

Printed in the United States of America

SPIN-X® is a trademark of Corsair Technology Incorporated.

Xerox® is a trademark of Xerox Corporation.

Unisys® is a trademark of Unisys Corporation.

IBM® is a trademark of International Business  
Machines Corporation.

© 1997 by Corsair Technology Inc.

The contents of this document are proprietary to  
Corsair Technology Inc., and are not to be disclosed to others or  
used for purposes other than intended without  
the written approval of Corsair Technology Inc.

January 9, 1996

Version 1.1

**Platform:** Unisys Series 1100/XX and 2200/XXX**Originator:** Corsair Technology Inc., Atlanta**Overview:** In Section I this document describes enhancements between SPIN-X/Central 4R4B and 4R5e. (Central 4R4C and 4R4D were not generally released versions). Section II describes problem resolutions for Central 4R4B and FDP 4R4.15. Section III details incompatibilities between 4R5 and 4R4B.

## **I. ENHANCEMENTS:**

### **All Devices:**

- **Year 2000 processing was fixed.**
- **Central 2200/500/900 Alternate Console Support**

Central has been updated to support alternate consoles on the 2200/500/900 (M-Series) systems. Changes to the EXEC Master Configuration Table (MCT) for the M-Series necessitated SPIN-X/Central modifications. This section describes this implementation.

#### Alternate Console Keyin Syntax for M-Series

L\* device CONS #n

"n" is a console message group number in the possible range 0..63. The following is a list of console message group numbers and any corresponding message group name:

- 0 is the main system message group name SYMSG.
- 1 is the I/O activity message group name IOMSG.
- 2 is the communication message group name RSICOM.
- 3 is the hardware confidence message group name HDWCON.
- 4-7 are for user message group names USER4 - USER7.
- 8-63 are available for site defined message group names.

#### **Examples:**

```
L* device CONS #0
L* device CONS #1
L* device CONS #4
L* device CONS #8
```

The first example routes messages to the main EXEC console SYMSG. The second example routes messages to the I/O activity console IOMSG. The third example routes messages to the USER4 console. The fourth routes messages to the site defined console message group number 8.

For the M-Series, the message group names would be directed to a console with SCMS or AC keyins. See the Unisys SCMS Administration Guide,

Volume 1 (7832 8861) sections 4.6 and 8.12 for information on managing console message groups with SCMS.

In response to the L\* CONS keyin, SPIN-X replies with the following message for the M-Series:

***device* 000240: MESSAGE ROUTING TO MESSAGE GROUP NUMBER *n***

"*device* " is the name of the device (usually a printer) for which the messages are intended. The device name is the same as the one defined in FDP.

For C-Series systems, the message group names would be assigned to consoles in the EXEC sysgen or with AC keyins.

#### **Alternate Console Routing using FDP DEVICE Keywords**

Two new FDP Device keywords are provided for alternate console message routing: MESSAGE\_GROUP\_NUMBER and CONSOLE\_TERMINAL.

#### **Examples:**

**DEVICE NAME=LX,....,MESSAGE\_GROUP\_NUMBER= *n*  
DEVICE NAME=LY,....,CONSOLE\_TERMINAL= *siteid***

The first example routes the messages for LX to console message group number *n* . The second example routes messages for printer LY to terminal *siteid* . The FDP MESSAGE\_GROUP\_NUMBER and CONSOLE\_TERMINAL parameters are mutually exclusive. The alternate console routing keyin overrides these keywords' values.

#### **Central CONS keyin**

The Central "L\* device CONS" console status keyin responds with the following for the M-Series:

***device* 000240: MESSAGE ROUTING TO MESSAGE GROUP NUMBER *n***

"*device* " is the name of the device from the L\* CONS keyin.

- **User Image Truncation to Byte Boundary Enhancement**

Central has been enhanced to truncate a user image to a partial word if specified rather than padding the image with spaces to the word boundary. The FDP Format WIDTH= parameter in combination with Central 4R4E now truncates images to a byte boundary if specified. The WIDTH= syntax has not changed.

## **Xerox:**

One more patch for the strange way that Xerox printers behave in off-line mode as opposed to on-line mode. While printing to ANSI tape this feature makes it possible to choose to output a page eject character separately when it occurs on a printline containing data. ( When the Xerox is in offline mode and the data happens to be a DJDE, it ignores the page eject). We put this feature, called TAPE\_CCBYTE\_MOD, in the Device statement. The FORMAT statement also has features which deal with such offline / online issues.

## **Siemens:**

- **Siemens Nixdorf Printing Systems (SNPS) Printers**

Central now has support for SNPS 2090, 2140 and 2240 printers. These continuous forms printers are rated at 90 and 140 pages per minute. Central's support is a redesign from the original implementation for the STK 6100 laser printers. This new implementation supports the Unisys 061 print control functions in addition to Format keyword specifications. FDP is employed to substitute Siemens fonts for fonts defined in legacy applications' 061 print controls without modifying the application or the printfile.

Two types of electronic forms are supported: FOB and EFO. The FOB (Forms Overlay Buffer) is created on a PC and downloaded from the host. The EFO (Electronic Form Overlay) is also created on a PC, but it is loaded from a diskette in the printer for additional security and compatibility with applications' 061 print controls flash operations.

- Central now supports the Special Forms 061 print control (Subtype 8). For each Special Forms record read, Central posts the following message to the console:

```
device 000472: SPECIAL FORMS REQUEST: form_id A
```

- For additional information, call the SPIN-X support line. Three Central guides specifically for the Siemens feature accompany the software: *Installation*, *Operator*, and *Reference*. Sites interested in ordering Central with the Siemens feature may request these guides from the SPIN-X support line.

## **Channel Connected Devices:**

Central now recognizes a transparent mode for block mux connected printers. If the code type 077 is specified for a user data image, then that data image is transferred to the printer without modification. The character code type for the user data image is specified when written to the printfile via ER SYMB\$. Any spacing for the image is serviced. The data is assumed to be 8-bit bytes aligned in the standard 9-bit byte fields. If any data image contains a byte with the 9th bit set, then the block mux transfer for that image stops at that point.

## II. PROBLEM RESOLUTIONS

### All Devices:

- A bug (blank page printed) in the No-banner option was fixed.
- For 2200/500/900 systems, if a device contained a DEVICE SGS in the sysgen, but it was not configured as a device with SCMS, then Central would request that the device be RV'ed. The following error message is now displayed if this error occurs:

#### **device 000242: DEVICE NOT CONFIGURED IN ODB; DROPPED**

*"device "* is the non-configured device name.

- Central mishandled the "M" 060 print control function when lines per page = zero. (Normally, the Exec would not permit lines per page = zero.) Central page ejected for each line rather than issuing a syntax error. This condition now results in the following error message to the output:

#### **000471: AN ERROR OCCURRED IN A TYPE 060 IMAGE ON PAGE n: x**

*"n "* is the page number and the *"x"* is the complete 060 image containing the error.

- Previously, Central might guard mode if it read a corrupt printfile containing bogus SDF control records such that the image length had the most significant bit set. This caused Central's image length variable to go negative. This problem is now fixed.

### Impact Printers:

FDP output a Format of Type=3211 with Dynamic=Yes to the SPIN-X database with the incorrect key. This caused a no-find condition when Central attempted to access a Type=3211 Format from the database. This is corrected in 4R4.15. The Format parameters now provide a setup for the queue regardless if Dynamic=Yes or No (default) is specified. Sites with STK 5000 printers should check their FDP-INPUT files to see if any Formats with Type=3211 AND Dynamic=Yes exist. If yes, these Formats will become active with FDP 4R4.15.

### Xpress:

- The Release Announcement for 4R4 incorrectly stated that the bulk file transfer option's FDP Format parameter is BULK=Yes, actually the syntax is Transfer=Bulk. Sites should avoid using Central to the Xpress Server because it requires the printfile to be completely copied before being @SYM'ed to the Xpress Server.
- Central 4R4 caused a no-find for Xpress Formats with Dynamic=Yes in 4R4. Central now obtains these Formats.

## Siemens

- **Documentation:**

Siemens Nixdorf Printing Systems (SNPS) sites performing an update install of a new Central version should consult section 4.1.1.1 of the *SPIN-X Central Installation Guide* for the Siemens printers, 4R5, for the ECL to perform the update install.

## III. INCOMPATIBILITIES

### All Devices:

SPIN-X/Central 4R5 contains new @MAP commands in the \*SRO.MAP element. Any user exits will need to be collected with the Central 4R5 relocatable using these new collector commands. Location counter \$21 (025) is now used by Central for the Siemens feature. See the *SPIN-X Reference Guide* 4R5 for information on collecting user exits.

<b>Preface</b>	<b>i-i</b>
<b>1. Introduction</b>	<b>1-1</b>
<b>2. Exec Queues and SPIN-X Formats</b>	<b>2-1</b>
2.1. Characteristics of the Print Formats	2-1
2.2. Using the @SYM Command to Specify Print Formats	2-3
<b>3. How to Use the Format Definition Program (FDP)</b>	<b>3-1</b>
3.1. Introduction to FDP Processor	3-1
3.2. The FORMAT command	3-3
3.3. The DEVICE Command	3-8
3.4. Executing the Format Definition Program	3-10
<b>4. Application Programming Considerations</b>	<b>4-1</b>
4.1. Controlling Margins and Line Spacing - FCB's and VFU's	4-1
4.1.1. Controlling Margins with the SPIN-X Format Definition File	4-1
4.1.2. Controlling Margins and Line Spacing through Programming	4-2
4.1.3. Controlling Margins and Line Spacing Using VFU's	4-3
4.2. Using Unisys 1100/2200 Print Control Images	4-3
4.3. Using Xerox Dynamic Job Descriptor Entries (DJDE's)	4-3
4.3.1. Processing Carriage Control Associated with DJDE Records	4-4
4.4. Print File Errors	4-5
4.5. Transparent Data Mode	4-5
<b>5. Downloading Files from the Unisys to the Xerox</b>	<b>5-1</b>
5.1. Using the Hostcopy Utility	5-1
5.1.1. Transferring User-created Files	5-2
5.1.2. Transferring Xerox Format Tape Files	5-4
5.2. Using the FILE DJDE	5-5
5.2.1. Downloading LPS Labeled Files	5-5
5.2.2. Downloading Card Image Files	5-6
<b>6. Setting-up a Test SPIN-X Run</b>	<b>6-1</b>
6.1. Steps for Setting-up a Test SPIN-X	6-1
6.2. Example ECL for Test SPIN-X Batch Job	6-2

---

<b>7. The SPIN-X Environment</b>	<b>7-1</b>
7.1. SPIN-X Internals Overview	7-1
7.2. Why the Bus and Tag Cable Adapters are Needed	7-2
7.3. The Unisys to IBM 3211 Environment Connection	7-3
7.3.1. Direct Connection	7-3
7.3.2. Connection Via a Unisys Byte Channel Transfer Switch	7-3
7.3.3. Connection Via a Channel Interface Switch	7-4
7.3.4. Connection Via a Channel Extender	7-4
7.3.5. Considerations of Multiple Host / Device Connections	7-4
<b>8. User Exits</b>	<b>8-1</b>
8.1. SPIN-X Defined User Exits	8-1
8.1.1. Beginning Printer Controls User Exit	8-3
8.1.2. Banner Page User Exit	8-4
8.1.3. Format Processing User Exit	8-5
8.1.4. Trailer Page User Exit	8-6
8.1.5. Ending Print Controls	8-7
8.1.6. Symbiont End of Processing Log Entry	8-8
8.2. SPIN-X Data Structures Passed to User Exits	8-9
8.2.1. Format Definition Table (FDT)	8-10
8.2.2. Printer Control Block (PCB)	8-10
8.2.3. File Control Table (FCT)	8-11
8.2.4. Printer Control Record (PCR)	8-11
8.2.5. SMOQUE Entry (SQE)	8-12
8.2.6. ASCII Edit Packet (AED)	8-13
8.3. Character Translation Table User Exit	8-14
8.3.1. ASCII to ASCII	8-14
8.3.2. ASCII to EBCDIC	8-14
8.4. Collecting SPIN-X with User Exits	8-15

<b>A. SPIN-X Log Entries</b>	<b>A-1</b>
<b>A.1. Fieldata Log Entries</b>	A-1
<b>A.1.1. Symbiont End of Processing Log Entry (Type 14)</b>	A-1
<b>A.1.2. Symbiont Start of Processing Log Entry (Type 15)</b>	A-4
<b>A.1.3. Symbiont Handler Log Entry (Type 35)</b>	A-6
<b>A.2. ASCII Log Entries</b>	A-9
<b>A.2.1. Symbiont Start of Processing Log Entry (Type 10112 Subtype 1)</b>	A-10
<b>A.2.2. Symbiont End of Processing Log Entry (Type 10112 Subtype 2)</b>	A-12
<b>A.2.3. Symbiont Device Error Log Entry (Type 11502)</b>	A-15
<b>A.2.4. Symbiont I/O Fault Error Log Entry (Type 11503)</b>	A-17
<b>B. ECL to Create a SPIN-X Absolute (MAP)</b>	<b>B-1</b>
<b>C. Listings of Xerox JSL and FSL Files</b>	<b>C-1</b>
<b>C.1. ON1100.JSL</b>	C-1
<b>C.2. P1100.JSL</b>	C-3
<b>C.3. SP2BNR.JSL</b>	C-4
<b>C.4. GRABAR.FSL</b>	C-4
<b>C.5. BANNER.FSL</b>	C-4
<b>C.6. BANNR2.FSL</b>	C-5
<b>D. Listing of the Distributed FDP Input File</b>	<b>D-1</b>
<b>E. Appendix E: FDP File Linkage</b>	<b>E-1</b>
<b>F. ECL to Copy FDP Files to Production (FDP/TOPRODUCTION)</b>	<b>F-1</b>
<b>G. SPIN-X Utilities</b>	<b>G-1</b>
<b>G.1. CLEANQ</b>	G-1
<b>G.2. FILE COPY</b>	G-1
<b>G.3. TRANSLATE/ASCII-EBCDIC</b>	G-1
<b>G.4. READ/XEROX-FILE</b>	G-1
<b>H. Sample "L" Processor Call Option Listing</b>	<b>H-1</b>

---

<b>I. SPIN-X ANSI Output Tape Feature</b>	<b>I-1</b>
<b>I.1. General Description</b>	<b>I-1</b>
<b>I.2. Installation</b>	<b>I-1</b>
<b>I.2.1. EXEC Configuration</b>	<b>I-1</b>
<b>I.2.2. Spin-X Configuration</b>	<b>I-1</b>
<b>I.2.3. Xerox Considerations</b>	<b>I-3</b>
<b>I.3. Operations</b>	<b>I-3</b>
<b>I.3.1. Operator Keyins</b>	<b>I-3</b>
<b>I.3.2. Processing of Keyins</b>	<b>I-3</b>
<b>I.3.3. Errors</b>	<b>I-4</b>
<b>I.3.3.1. Input File Errors</b>	<b>I-4</b>
<b>I.3.3.2. Tape Facility Errors</b>	<b>I-4</b>
<b>I.3.3.3. Tape Output Errors</b>	<b>I-4</b>
<b>I.4. Translate Table User Exit (UXTAPETABLE)</b>	<b>I-4</b>
<b>I.5. IBM Host Processing</b>	<b>I-5</b>
<b>I.5.1. Overview</b>	<b>I-5</b>
<b>I.5.2. Installation Instructions for JESPRINT</b>	<b>I-5</b>
<b>I.5.3. Example JCL</b>	<b>I-5</b>
<b>I.6. Xerox Offline Processing</b>	<b>I-6</b>
<b>I.6.1. Listing of Xerox Offline JSL</b>	<b>I-6</b>
<b>I.6.2. Problems With Offline Processing</b>	<b>I-7</b>

---

## Tables and Figures

<b>Table 2-1</b> Characteristics of the Formats Supplied by SPIN-X.	2-2
<b>Figure 7-1</b> Diagram of Internal SPIN-X Functions	7-2
<b>Figure 7-2</b> The Basic 1100/2200 to 3211 Connection	7-3
<b>Figure A-1</b> Depiction of the SPIN-X End of Processing Log Table Entries	A-1
<b>Figure A-2</b> Depiction of the SPIN-X Start of Processing Log Table Entries	A-4
<b>Figure A-3</b> Depiction of the SPIN-X Symbiont Handler Log Entries	A-6
<b>Figure A-4</b> Depiction of the SPIN-X Start of Processing Log Table Entries	A-10
<b>Figure A-5</b> Depiction of the SPIN-X End of Processing Log Table Entries	A-12
<b>Figure A-6</b> Depiction of the SPIN-X Symbiont Device Error Log Table Entries	A-15
<b>Figure A-7</b> Depiction of the SPIN-X Symbiont I/O Fault Error Log Table Entries	A-17

The SPIN-X software package provides on-line access to IBM 3211 printer environments from Unisys Series 1100/2200 mainframe computers via a byte or block multiplexor or FIPS channel. SPIN-X allows a site to configure a Xerox LPS as a standard Unisys line printer and use the standard Unisys ECL @SYM command to direct printfiles to the Xerox printers. SPIN-X not only provides control of the IBM 3211 channel environment needed for on-line access, but also performs automatic insertion of Xerox DJDE's (Dynamic Job Descriptor Entries) needed for accessing the extra functionality of the Xerox printers.

This manual explains how the SPIN-X administrator can define the different print formats to be used and how the user can access them. A discussion of SPIN-X's internal and external environment is given which includes details on various hardware connections. Information is also included for sites wishing to modify certain SPIN-X functions through user exits.



A queue to SPIN-X is an Exec queue configured in a DEVICE, OUTPUT, or STATION LOCAL SGS. Basically, queues are where users @SYM printfiles. Each Exec queue accessed by SPIN-X has defined a corresponding SPIN-X "format" of the same name. A SPIN-X format provides to SPIN-X various printing characteristics for printfiles obtained from a specific Exec queue. SPIN-X uses the queue name to look-up the corresponding format. For example, if SPIN-X obtains a printfile from queue MYQ, then it uses SPIN-X format MYQ for printing the file.

Section 2.1 of this chapter provides a description of printing attributes that are specified by a format. The information is independent of the method SPIN-X provides for declaring formats. The next chapter provides detailed information on declaring formats to SPIN-X using the Format Definition Program (FDP). Section 2.2 illustrates use of the @SYM command for selecting a queue and thus specifying a SPIN-X format.

---

## 2.1 Characteristics of the Print Formats

---

Table 2-1 on the following page gives a summary of the characteristics of the fourteen print formats distributed with SPIN-X. The terms used in describing these characteristics are defined as follows.

1. **Font:** A set of characters having the same size and style. The font used at the printer is determined by the Xerox PDE chosen in the FDP Format command. The features which distinguish one font from another are...
  - a. the set of characters available (e.g some sets have no lower case or special characters.)
  - b. the height and width of the characters,
  - c. the weight of the characters (e.g. medium or bold),
  - d. the stress of the characters (e.g. Roman or italic),
  - e. and the typestyle (e.g. Helvetica, Times, Optima, etc.).
2. **Orientation:** Orientation is either portrait or landscape. The Xerox is capable of printing a page length-wise (portrait), or width-wise (landscape).
3. **Simplex/Duplex:** The Xerox can print on one side of the paper only (simplex), or on both sides of the paper (duplex).
4. **Number and Position of Logical Pages:** The Xerox is capable of having up to 66 logical pages defined to start anywhere on the page.
5. **Lines per Inch / Lines per Page:** Using Xerox laser printer fonts can result in a wide variety of line spacing values and number of lines per page. These can vary greatly from the standard 66 lines per page which is usually sysgened as the default for a printer. The SPIN-X administrator can assign to each print format the maximum number of lines to be printed before a page eject is sent by SPIN-X.
6. **Top Margin, Left Margin:** These are the values in inches for where the first line will begin printing.

**Legend:**

<b>PPS</b>	= Pages Per Side
<b>LPI</b>	= Lines Per Inch (low density)
<b>ALPI</b>	= Alternate Lines Per Inch (high density)
<b>LPP</b>	= Lines Per Page
<b>ALPP</b>	= Alternate Lines Per Page
<b>CPL</b>	= Columns Per Line
<b>TM</b>	= Top Margin (as defined in the Xerox PDE)
<b>LM</b>	= Left Margin (as defined in the Xerox PDE)

Format Name	Orientation	Simplex/ Duplex	PPS	LPI/ ALPI	LPP/ ALPP	CPL	TM (inches)	LM (inches)
PR	Portrait	Duplex	1	8.1/ 10.7	66/88	132	0.18	0.66
LS132	Landscape	Simplex	1	8.1/ 10.7	66/ 88	132	0.18	0.66
† LS132G	Landscape	Simplex	1	8.1/10.7	66/ 88	132	0.18	0.66
LD132	Landscape	Duplex	1	8.1/ 10.7	66/ 88	132	0.18	0.66
† LD132G	Landscape	Duplex	1	8.1/ 10.7	66/ 88	132	0.18	0.66
LD150	Landscape	Duplex	1	8.1/ 10.7	66/ 88	150	0.18	0.5
PS80	Portrait	Simplex	1	6.0/ 8.0	63/ 84	0.5	80	0.5
PD80	Portrait	Duplex	1	6.0/ 8.0	63/ 84	80	0.5	0.5
PS95	Portrait	Simplex	1	6.0/ 8.0	63/ 84	95	0.5	0.5
PD95	Portrait	Duplex	1	6.0/ 8.0	63/ 84	95	0.5	0.5
PD106	Portrait	Duplex	1	8.1/ 8.1	80/ 80	106	0.57	0.58
PD132	Portrait	Duplex	1	12.5/12.5	132/ 132	132	0.22	0.51
PD132T	Portrait	Duplex	2	12.5/12.5	66/ 66	132	0.22	0.51
LABELS	Portrait	Simplex	33	6.0/8.0	5/7	33	0.05	0.06

**Table 2-1** Characteristics of the Formats Supplied by SPIN-X.

†Formats LS132G and LD132G have gray shaded bars highlighting alternating three line blocks.

‡Format LABELS prints 33, 5-line labels per page.

---

## 2.2 Using the @SYM Command to Specify Print Formats

---

Users specify a format to SPIN-X by issuing an @SYM of their printfile to an Exec queue of the same name as the format. Once SPIN-X receives a file from a queue it obtains the format that matches the queue, and it sends printer controls to the Xerox based upon information from the format. The following example would result in SPIN-X printing the file, MYPRINTFILE, using format LS132:

```
@SYM,U MYPRINTFILE . , 1 , LS132
```

Consult the *Unisys Series 1100 Executive System Programmer Reference* for details of using the @SYM command.

---

## 3 How to Use the Format Definition Program (FDP)

The Format Definition Program (FDP) is a processor provided from the SPIN-X Release Tape that accepts keyword commands for declaring formats and identifying printers for SPIN-X control. FDP is supplied with 14 predefined formats and an example device command. The local SPIN-X Administrator can add, retain, or remove formats and device commands as deemed appropriate. The remainder of this chapter describes the FDP commands, supporting files and how to run the FDP processor.

The purpose of the SPIN-X Format Definition Program (FDP) is to allow the specification of print characteristics for printfiles contained in a particular Exec queue. The specifications for a particular queue are called a SPIN-X "format". Each Exec queue should have defined for it a corresponding SPIN-X format bearing the same name as the queue.

---

### 3.1 Introduction to FDP Processor

---

The input to the Format Definition Program for SPIN-X Central consists of two types of commands:

1. The **FORMAT** command, and
2. The **DEVICE** command.

The **FORMAT** command is used to describe the printing characteristics of a single print format or queue. The **DEVICE** command is used to specify a printing device to be controlled by SPIN-X. Therefore, there must be a **FORMAT** command for each available print format, and a **DEVICE** command for each device to be controlled by SPIN-X. The parameters for these commands are keyword driven and may be entered in any order. The FDP commands are read from the SDF file \*FDP-INPUT.

The Format Definition Program creates as output a database consisting of three files:

1. **FORMS-DEF** - the Format Definition File. This SDF file contains device definitions and print formats to be kept in core by SPIN-X during execution. This is the traditional location for print formats to be stored.
2. **SRI\$DATABASE** - this MSAM file contains an index of device definitions and print formats. This index points to the location of an object in a flatfile. SPIN-X "looks-up" information on the object it requires in the index when needed; it then obtains the object, such as a format, from the flatfile.
3. **SRI\$FLATFILE** - this file contains the actual objects that are indexed in **SRI\$DATABASE**. This file is structured for efficient access by SPIN-X during runtime, and it does not conform to any standard OS1100 file definition.

The Format Definition Program will also create a report file containing a list of the devices and print formats that were defined within the input file \*FDP-INPUT.

Sections 3.2 and 3.3 define and explain the syntax used in the FORMAT and DEVICE commands, respectively. In the syntax explanations, the parentheses indicate a value for a parameter: brackets, which are separated by a vertical bar (|), indicate an either/or choice (the parentheses, brackets and vertical bar would not actually appear in the input file). Each separate FORMAT and DEVICE command must appear on a new line, but may be continued from one line to the next, by ending with a semicolon (;). Anything appearing after a ' . ' (space-period-space) will be treated as a comment, however, any characters appearing after column 80 will be ignored. Parameters may be delimited by commas or spaces and may appear in any given order. FDP treats hyphens and underscores as if they were the same. A command is terminated when there is no continuation indicated.

Finally, Section 3.4 explains how to execute the Format Definition Program and generate the Format Definition File and its related databases.

**NOTE:** Beginning with SPIN-X 4R1 references to the "Form Definition Program" and the "form" command were changed to the "Format Definition Program" and "format" respectively. The change was made to avoid conflict with industry standard terminology referring to electronic forms. The FDP FORM command is still accepted, however sites are encouraged to convert their FDP FORM commands to FORMATs now because future implementation plans call for introducing a new FDP FORM command based upon its common meaning. The only syntax change needed is the conversion of the "FORM" command to "FORMAT".

**NOTE:** The FDP absolute on the SPIN-X Release Tape for the 1100/2200 uses no common banks and must be used if you have ACOB 5R1 installed; however, if you have ACOB 6R1 and SORT 14R1 installed you may remap FDP to use your site's common banks.

## 3.2 The FORMAT command

The FORMAT command allows the installation administrator to describe the print characteristics of each print format or queue using simple, keyword parameters. The syntax of the FORMAT command is shown below: when applicable, default values are shown in bold type, otherwise they are specified on the right hand side in braces.

### FORMAT;

**NAME** = (a 1-6 alphanumeric queue-name);

**MODE** = [P | L]; . P = portrait, L = landscape

**DUPLEX** = [Y | N]; . Y = yes, N = no

**PAGES** = (an integer between 1 and 63);

**LINES** = (an integer between 1 and 262,000); **{66}**

**JDL** = (a 1-6 alphanumeric Xerox JDL filename); **{ON1100}**

**JDE** = (a 1-6 alphanumeric Xerox JDE filename); **{NONE}** (or leave blank)

**PDE6** = (a 1-6 alphanumeric Xerox PDE filename for normal line density); **{NONE}**

**PDE8** = (a 1-6 alphanumeric Xerox PDE filename for higher line density); **{NONE}**

**LINES8** = (an integer between 1 and 262,000); **{88}**

**DENSITY** = [6 | 8];

**OVERLAY** = (a 1-6 alphanumeric Xerox FRM filename); **{NONE}**

**WIDTH** = (an integer between 1 and 512); **{150}**

**TOP\_MARGIN** = (an integer indicating number of blank lines at top); **{0}**

**BOTTOM\_MARGIN** = (an integer indicating number of blank lines at bottom); **{0}**

**SHIFT** = [Y | N]; . Y = yes, N = no

**MCONTROL** = [Y | N]; . Y = yes, N = no

**OVERRIDE** = [Y | N]; . Y = yes, N = no

**DYNAMIC** = [Y | N]; . Y = yes, N = no

**OFFLINE** = [Y | N]; . Y = yes, N = no

**DJDE** = (flag value for Xerox DJDE command - see FLAG VALUES); **{%%DJDE}**

**RAUX** = (flag value for Xerox RAUX command - see FLAG VALUES);  
**{x'01020102'}**

**RSTACK** = (flag value for Xerox RSTACK command - see FLAG VALUES);  
**{x'02010102'}**

**RPAGE** = (flag value for Xerox RPAGE command - see FLAG VALUES);  
**{x'02010201'}**

**BANNER\_FORM** = (form name) **{NONE}**

**BANNER\_PAGE** = [Y|N]; . Y = yes, N = no

**NOTE:** The SPIN-X administrator must confirm that the Xerox resident JSL code agrees with the FORMAT command parameters. The administrator must also understand how the FORMAT command parameters interact with one another.

The following is a detailed explanation of each keyword in the FORMAT command:

- NAME** This keyword indicates the name of the individual format and queue, which will be used in the third field of the @SYM command. It must be a six character or less alphanumeric name (e.g., NAME = PD80;).
- MODE** This keyword specifies the orientation of the print format, which may be either portrait or landscape. SPIN-X does not send a "PMODE=" DJDE with the print file, instead the determination of the print mode is based on the "PMODE=" parameter in the PDE for the format on the Xerox. This parameter is for documentation purposes when producing the report.
- DUPLEX** If DUPLEX=YES is coded, then SPIN-X will put a "DUPLEX=YES" parameter in the DJDE which is sent in front of the report. This value will override the "DUPLEX=" option on the OUTPUT command in the JDE on the Xerox. If "DUPLEX=NO" is coded, or the parameter is omitted then no "DUPLEX=" parameter will be sent.
- PAGES** This keyword indicates the number of logical pages per sheet of paper (for example, PAGES=4 for a duplex format with 2 logical pages per side, whereas PAGES=1 for a simplex format with 1 logical page per side). This is specified to SPIN-X mainly for error recovery purposes. The actual determination of pages per sheet is made by the Xerox based on the number of "BEGIN=" parameters in the PDE defining the logical pages on each side, and the "DUPLEX=" parameter in the JDE.
- LINES** This parameter indicates the total number of lines allowed to be printed before SPIN-X forces a top-of-page. This number includes both the BOTTOM-MARGIN and TOP-MARGIN values, and should be the same as the Xerox "BOF" parameter (if it is used) in the JDL. See Section 4.1 for information on how programmers may override the default page size specifications for a format.
- JDL** This parameter is the name of the Job Descriptor Library, which is the main program on the Xerox which performs all processing of print files. The JDL contains all the JDE's and all the commands defining the format of the input processing and the output processing requirements for all jobs. This field should be the same for every FORMAT command, however it is conceivable that users may in the future need separate JDL's for processing XICS or other nonstandard output. The default is ON1100, which should be the same JDL used by the operator on the Xerox START command.
- JDE** This keyword indicates the Job Descriptor Entry in the JDL stored on the Xerox. The administrator can utilize the JDE to establish a wide variety of output characteristics for the print format. This is where the greatest amount of flexibility is accessible through the use of various Xerox printer parameters. Some of the Xerox options which may be included in the JDE are COPIES, COVER, DUPLEX, FEED, FORMAT, FORMS, OFFSET, SHIFT, OTEXT, and RTEXT. Each different JDE can also have associated with it different VFU, PCC, LINE and accounting processing functions (these are also Xerox printer parameters). Commonly, many of these options will be set to the same values for most formats. In this case, there could be just one JDE for all duplex prints, one for all simplex prints, and one for each format requiring special processing (such as, special paper or multiple copies). Note that the Xerox "FORMAT=" parameter can be included in the JDE stored on the Xerox printer or it can be selected with the PDE6 and PDE8 parameters to the Format Definition Program. A PDE defines the fonts, print mode and page-begins for each report. There may be several formats using the same JDE but different PDE's. Any PDE specified in the JDE on the Xerox will be overridden by the PDE specified in the PDE6 or PDE8 parameter in the Format Definition File unless "NONE" is specified. See PDE6 and PDE8 for more information about PDE's.

**PDE6 and PDE8** These parameters specify the Print Descriptor Entries (PDE's) to be used for normal and higher line density, respectively. In this sense they can be likened to the "lines per inch" and "alternate lines per inch" entries in Table 2-1. A PDE defines 3 print characteristics:

1. the print mode or orientation (landscape or portrait),
2. the font(s) used and its associated line spacing,
3. the number and beginning coordinates of each logical page.

SPIN-X sends a "FORMAT=*pdename*" DJDE parameter to the Xerox calling for the PDE specified in the Format. PDE6 will be chosen for standard 6 lines per inch print files unless the Density=8 parameter is specified. If a change of print density is encountered or if 8 lines per inch is the Unisys system default then PDE8 will be chosen. Although the actual line spacing of the font used in the PDE may vary greatly from 6 or 8 lines per inch, it is suggested that a PDE8 be established for each format which allows for the printing of up to 88 lines per page rather than the usual 66. In some cases, such as 2-up duplex, it is not possible to print two 88 line pages on a side. Sites having no applications calling Unisys print density changes may ignore this problem. Note that any PDE specified to the Format Definition Program will override any PDE specified in the JDE 'FORMAT=' parameter predefined on the Xerox. If "NONE" is specified for PDE6 and/or PDE8 then no 'FORMAT=' DJDE will be sent for the specified density and the PDE which is called for by the JDE stored on the Xerox will be used.

**OVERLAY** This parameter is utilized to specify the Xerox FRM filename for overlaying the report data. For example, the provided formats, LS132G and LD132G, have shaded gray-bar overlays (see Appendix J). The FRM file specified for the formats LS132G and LD132G contains the compiled FSL statements to create the shaded gray-bar overlay.

**SHIFT** This parameter is used to offset the print by one quarter inch to allow room for binding or using three-hole punched paper. If 'YES' is specified, SPIN-X will send a "SHIFT=YES" DJDE along with each file sent to the print format. **NOTE:** If the job is duplex and the logical page begins within less than one quarter inch of the edge, then the back page would be shifted off the edge of the paper and will not be printed.

**TOP\_MARGIN**  
**BOTTOM\_MARGIN** These parameters refer to the number of blank lines to be appended to the top and bottom of each page, respectively. Note that this is performed by SPIN-X on the 1100/2200 before the data reaches the printer. Both of these values should be included as part of the lines per page calculation specified by the LINES parameter. See Section 4.1 for information about how programmers may override the default page size specifications for a format.

**LINES8** This parameter is the same as the LINES parameter except that it indicates the number of lines per page for 8 LPI mode.

**DENSITY** This parameter specifies the default LPI of the format. This value must be either 6 or 8 LPI.

**WIDTH** This parameter informs SPIN-X of the maximum number of characters allowed per line. If a print line reaching the Xerox is still too long to fit on the page, it "disappears" off the right edge of the paper and the message:

**DATA ON PAGE EXCEEDS PAGE SIZE**

will appear on the Xerox console. In this case, a Xerox accounting summary sheet will be given at the end of the print file, whether it was requested or not. To avoid this the administrator may wish to ensure that the WIDTH parameter is set to the proper value by evaluating the following equation:

**WIDTH** = (*pagewidth-leftmargin*) \* *fontsize*      where...

*pagewidth* is 8.5 or 11.0 depending upon the orientation of the format,  
*leftmargin* is the left margin value specified on the "BEGIN=" in the PDE,  
*fontsize* is the number of characters per inch of the font.

**MCONTROL** This parameter allows a site to designate whether printfiles sent to this print format will use the default margin settings as defined on the TOP-MARGIN, BOTTOM-MARGIN, LINES, and DENSITY parameters for the format, or if these default settings should come from those defined for the printer in the EXEC configuration during sysgen. This will only come into consideration when a print control image of 'M,\*,\*,\*' is encountered, which normally means 'set margins to device default values'. This can occur when a COBOL program writes to a file assigned to 'PRINTER' but no LINAGE clause is specified.

A value of 'Y' or 'YES' instructs SPIN-X to use the defaults as defined by the FORMAT command for that print format. A value of 'N' or 'NO' indicates that the default values defined for the device will be used.

Note that any file containing any margin control image with values other than '\*' will be interpreted and acted upon by SPIN-X. The values defined will override the defaults provided to the Format Definition Program and those defaults defined for the device regardless of the MCONTROL parameter. See Section 4.1 for more details on margin processing.

**OVERRIDE** If this parameter is set to 'YES' or 'Y', then any OS1100 'M' or 'B' 060 print functions in printfiles sent to this FORMAT are ignored; this guarantees the margins, lines per page, and density specifications come from the FORMAT statement rather than the printfile. Other print functions are processed, even if they are contained within the same 060 image as the 'M' and/or 'B' record(s). The default is 'NO'. See the *Unisys OS1100 Executive System Software, Executive Requests, Programming Reference Manual* (UP-4144), Chapter 16, for information on print functions.

**DYNAMIC** SPIN-X Release 4R1 introduced the database files that allow the compiled formats to be stored on disk rather than being maintained in memory while SPIN-X is running. If DYNAMIC=N is coded for a FORMAT, then that format will be maintained in SPIN-X's memory rather than on disk. The default is DYNAMIC=N. See section 3.1 for a discussion of the SPIN-X database files where the dynamic formats reside.

Sites with only a few formats defined may wish to set DYNAMIC=NO for all formats to avoid the extra amount of I/O processing required for retrieving the format record from the database (although this is slight). Or, a few of the most often used formats could be kept in core while the bulk of the less frequently used formats could be kept on disk.

**OFFLINE** This option instructs SPIN-X to handle the spacing request associated with a DJDE record in a manner equivalent to the way in which it would be handled in an offline tape processing mode on the Xerox. Specifically, any "write and space" carriage control associated with a DJDE record will be ignored, any "write and skip" request (such as a top of page carriage control) will be performed. See the discussion of DJDE handling in Section 5.2. This option is meaningless if an actual tape device is being used. For offline printing to tape, see the DEVICE statement.

**FLAG VALUES** (overview) **DJDE, RAUX, RPAGE, RSTACK** - The flag values established to indicate the DJDE, RAUX, RPAGE and RSTACK commands are defined for SPIN-X with the Format Definition Program. Although these flags are defined within the FORMAT command and can vary from format to format, they must be constant for all formats using the same JDL. For this reason, these flags should be set in the first FORMAT command in order to establish a default (described later in this section) for all other FORMAT commands. If any site uses different values for any of these flags the administrator should insure that the values in their JDL match the values specified in the FORMAT command.

The flags may be defined using their hexadecimal or octal equivalents or as regular ASCII text. They must be enclosed within quotes and preceded by a one letter code specifying whether the quoted string is to be interpreted as ASCII (A), octal (O) or hexadecimal (X). The flag value may be up to 255 characters in length and may be continued from one line to the next by closing the quoted string at a whole byte boundary and suffixing the string with a hyphen ('-'). The next line should begin with a quoted string containing the rest of the flag value. See the example at the end of this section for an illustration of the use of the DJDE, RAUX, RPAGE and RSTACK flag values.

**DJDE** This flag specifies the value used to signal the Xerox that the current line is a Dynamic Job Descriptor Entry and is to be treated as a command to the Xerox rather than data to be printed. This value must match the value specified in the IDEN statement in the JDL on the Xerox. Since this is only set once in the JDL it should be the same for all formats. It must be followed in the FDP input file by the OFFSET number specifying the number of characters preceding the flag. The OFFSET number is then followed by the SKIP value which specifies the number of characters preceding the first actual DJDE parameter. See the Xerox Reference documentation for a further explanation of setting up the DJDE flag. Although the Xerox Reference documentation states that the SKIP and OFFSET numbers may be negative SPIN-X only supports positive values for these numbers. Also, the SKIP value must always be greater than or equal to the OFFSET plus the length of the flag. In other words, the DJDE flag must appear before the DJDE commands within the record. See the FLAG VALUES overview for more information.

**RAUX** This flag specifies the value used to signal the Xerox to switch the paper feed to the auxiliary input tray. SPIN-X relies on this value to pull a separator sheet from the aux tray to print the header page for each report.

**RPAGE** This flag indicates the value used to signal the Xerox to reposition printing to the front of a new sheet of paper regardless of which logical page it is currently printing or whether the format is simplex or duplex. SPIN-X sends an RPAGE flag at the end of each banner page to insure proper pagination when the report begins printing. See the FLAG VALUES overview for more information.

**RSTACK** This flag specifies the value used to indicate an "end-of-report" condition separating reports for the purpose of accounting information and formatting criteria. SPIN-X sends an RSTACK flag at the end of each report. See the FLAG VALUES overview for more information. Note that setting "JOB-DISPLAY=YES" will result in the RSTACK flag being sent in front of the report. See the explanation of the JOB-DISPLAY parameter in Section 3.3 for details.

**BANNER\_FORM** An alternate electronic form than the SPIN-X Banner Form may be specified to FDP for printing the banner page for a given format.

**BANNER\_PAGE** If this option is set to NO, no banner page will be printed. The default behavior is to print the banner.

**Defaults** To avoid having to list every parameter value for each format, the Format Definition Program allows the administrator to choose the default values for any missing parameters. These default values must be specified in the first FORMAT command, thus, every parameter must be specified in the first FORMAT command in order to establish your defaults. The following parameter, however, cannot be set in this way:

DYNAMIC = [Y | N]

The following FORMAT commands are examples of those included with the SPIN-X software (located within the filename: *qual\**FDP-INPUT where *qual* is the qualifier used to catalog this file during the installation of SPIN-X; *qual* is usually "SPIN-X"). The default parameter values are established by the first FORMAT \*FDP-INPUT command for the format PR.

**Examples**

```
FORMAT NAME=PR, MODE=P, DUPLEX=Y, PAGES=4, LINES=66, ;
      JDL=ON1100, JDE=PR, PDE6=FMT1X, PDE8=FMT3X, ;
      OVERLAY=NONE, WIDTH=132, TOP=6, BOT=3, SHIFT=N;
      DJDE=A'%%DJDE', 0, 7, RAUX=X'01020102', 0, ;
      RPAGE=X'02010201', 0, RSTACK=X'02010102', 0

FORMAT NAME=PD80, MODE=P, DUPLEX=Y, PAGES=1, ;
      JDE=PD80, PDE6=FMT8X, PDE8=FMT6X, WIDTH=80

FORMAT NAME=LS132, MODE=L, DUPLEX=N, PAGES=1, ;
      JDE=LS132, PDE6=FMT1X, PDE8=FMT3X
```

---

### 3.3 The DEVICE Command

---

The DEVICE command allows the SPIN-X administrator to specify which printing devices SPIN-X will be allowed to control. The syntax of the DEVICE command is shown below: defaults, where applicable, are in bold type. The default is used when a DEVICE statement parameter is not included in the DEVICE command.

**DEVICE;**

**NAME** =(a legal Unisys device name),;

**TYPE** =[9700 | 9790 | 8700 | 8790 | 4090 | 4050 | 4650 | 4850 | 4135] , ;

**JOB-DISPLAY** =[Y | N]; . Y = yes, N = no

**DJDE-REORDER** =[Y | N]; . Y = yes, N = no

**DCODE** =(an octal number between 0 and 77);

**MESSAGE\_GROUP** =(Exec Console Message Group or Terminal Siteid);

**INIT\_RSTACK** = [Y | N]; . Y = yes, N = no

**EOF\_SKIP\_MESSAGE** = [Y | N];

**MESSAGE\_GROUP\_NUMBER**= [0,1,..63];

**CONSOLE\_TERMINAL**=CMS\_SITEID ;

**TAPE\_CCBYTE\_MOD** = [Y|N] . Y = yes, N = no (For Tape devices only)

The following page gives a detailed explanation of each keyword in the DEVICE command:

<b>NAME</b>	The mnemonic for the printer from the Exec DEVICE SGS.
<b>TYPE</b>	The type of printer assigned to this device name. The printer models listed are all driven in the same way regardless of the specific model of LPS printer given in the TYPE statement.
<b>JOB-DISPLAY</b>	When this parameter is set to YES, SPIN-X causes the RUNID for the print file to be displayed on the Xerox console when a JOBS command is issued or the "STATUS" key is pressed. This is made possible by sending the RSTACK flag symbol followed by the RUNID in the HRPTNA field as defined in the Xerox JSL. Because the RSTACK must be sent before the printfile and not at the end, this causes the last page of the report to remain in the printer until flushed out by the RSTACK on the next report or until the END command is issued at the Xerox console. In the Xerox JSL the RSTACK command must be coded with the parameter "HRPTNA=(4,16)".
<b>DJDE-REORDER</b>	If this command is set to YES then SPIN-X will send the JDE= and JDL= commands in the DJDE that is sent before the banner page. These commands are normally sent in the DJDE that SPIN-X sends after the banner page and before the report. By reordering these commands any parameters associated with the JDE will go into effect at the very beginning of the report. Certain newer JSL parameters, such as STAPLE, RESOLUTION, FACEUP, and STOCKSET cannot be invoked at any point in the job other than the very beginning.
<b>DCODE</b>	This parameter allows the SPIN-X Administrator to specify what equipment code value should be used in the types 14, 15, 35, 10112, 11502, and 11503 log entries. By specifying a unique DCODE, accounting programs can easily discern SPIN-X generated symbiont log records from others in the system logfile. Allowed values are 0 through 77 and are considered octal values.
<b>MESSAGE_GROUP</b>	This parameter provides a convenient alternative to the L* CONS keyin for specifying an alternate console or console terminal for the display of messages for the given device. The value specified should be an Exec Console Message Group if messages are to be routed to an alternate console. If messages are to be routed to a terminal console, then the terminal's siteid should be specified. For 2200/500/900 systems, alternate system consoles must be specified using the newer MESSAGE_GROUP_NUMBER keyword.
<b>INIT_RSTACK</b>	If this parameter is set to YES then SPIN-X will send an RSTACK at the beginning of a print job if the previous printjob did not have an RSTACK following it because of an intervention required condition. The RSTACK is necessary to delimit to the printer the previous print job from the current print job. This obviates the need for ENDing the job on the printer after an ABORT O command. The INIT_RSTACK parameter is unnecessary if the JOB_DISPLAY=YES parameter is specified.
<b>EOF_SKIP_MESSAGE</b>	If this parameter is set to YES then SPIN-X will post message 000409: <p style="text-align: center;"><b>SKIPPED TO EOF - REPRINT? PAGES OR N</b></p> if a skip keyin was entered anywhere within the printfile and an end of file is encountered. This ensures the Operator of an opportunity to specify the number of pages to reprint once the end of file is encountered regardless of the page count specified on the L* device R+x skip keyin.
	<b>MESSAGE_GROUP_NUMBER</b> This message specifies the console message group number for directing this device's console messages after Central initialization.
<b>CONSOLE_TERMINAL</b>	This parameter specifies the CMS SITEID for directing this device's console messages after Central initialization.

**TAPE\_CCBYTE\_MOD**

The DEVICE TYPE **must** be TAPE for this option (see Appendix I). If set to **Y**, a page eject control character (1) on a printline containing data is always output separately with no data, followed by the data image with a no-spacing (+) control character. When the Xerox printer in on-line mode encounters a page eject control character on the same line as DJDE data it sends the page eject, but in off-line mode an eject associated with DJDE data is ignored. By putting the eject on a separate line we get the Xerox to send the eject whether it is in on-line or off-line mode. This should provide print output compatibility between on-line and off-line printing.

Please note the following with regard to page overflow processing.

Normally, when a printline overflows a page length boundary a page eject is generated. The spacing for the line (space one line and print) is ignored so that it will print on the **first** line of a new page. When a page eject is caused by an embedded DJDE however, the spacing is honored and printing begins on the second line. SPIN-X is written so that this behavior by Xerox printers has been made the same for both on-line and off-line printing. There is however an "OFFLINE" parameter on the FORMAT statement which causes spacing associated with DJDEs for online printers to be ignored in just the same way as is the case with offline printing: thus data following a DJDE will be printed on the **first** line.

**Example:** The following DEVICE command is included with the SPIN-X software and is located in the same \*FDP-Input file as the FORMAT commands in the previous section.

```
DEVICE NAME=LX,TYPE=9700
```

Note that the DEVICE name must match the name chosen for the printer during EXECsystem generation.

## 3.4 Executing the Format Definition Program

This section explains the procedure to follow in order to execute the Format Definition Program.

First, the input file, \*FDP-INPUT., containing the FORMAT and DEVICE commands must be updated to contain the new print formats and/or devices (if modifications are desired). Next, issue the following ECL commands:

```
@QUAL qual
@*FDP.FDP
```

...where *qual* is the qualifier used to catalog the FDP file during the initial installation of SPIN-X (The qualifier is normally "SPIN-X". An alternate qualifier could be used if, for instance, an alternate \*FDP-INPUT file were to be tested ).

**NOTE:** Beginning with SPIN-X 4R3, the FDP processor handled all file assignments and linkages internally instead of using an ECL addstream. Appendix E lists the files used by FDP and their internal link names.

The FDP processor will assign the input file (\*FDP-INPUT), create a new cycle of the report file Format Definition File (\*FORMS-DEF), new cycles of the database files (\*SRI\$DATABASE and \*SRI\$FLATFILE) and a new cycle of the report file (\*REPORT-FILE).

If any errors in the input file were detected then appropriate error messages will be displayed on the terminal as well as in the report file. Note that only an error-free run will replace or alter the information in the Format Definition File or its related database files.

Finally, the new Format Definition File and its related database components must be copied to the production versions from which SPIN-X reads its information. To do this, issue the following command:

```
@ADD,L *FDP.FDP/TOPRODUCTION
```

A listing of this addstream is contained in Appendix F for reference by the SPIN-X Administrator.

**NOTE:** SPIN-X must be terminated and started again in order for the new configuration to take effect.

---

## 4 Application Programming Considerations

---

### 4.1 Controlling Margins and Line Spacing - FCB's and VFU's

---

The Xerox Laser Printing Systems, while providing compatibility with line printer applications, also have capabilities related to *page* printing which introduce new concepts to the forms designer and the application programmer. The capability of defining multiple logical pages to begin anywhere on the paper can allow the programmer to forego a great deal of line and column counting when positioning text in a print file. Also, fonts can use a much wider variety of line spacing than is possible on a standard line printer. This advantage in flexibility should be taken into consideration when developing new applications. However, the first priority for most sites will be to convert existing line printer jobs to run on the Xerox without having to change application programs.

Most applications designed for line printers assume top of form to be the top of the physical sheet of paper and then skip a certain number of blank lines to position the first print line. This may be accomplished through SPIN-X in several ways as described below.

#### 4.1.1 Controlling Margins with the SPIN-X Format Definition File

---

In the simplest scenario, if a datafile which is not a printfile is @SYM'ed to a print queue accessed by SPIN-X, the top margin, bottom margin, lines per page and density will be determined by the parameters for that print format as defined in the Format Definition Program Input File (refer to Chapter 3). SPIN-X examines the values specified for these parameters and builds an FCB (Forms Control Buffer) containing this information. This FCB record is then downloaded in front of the print file and is used by the Xerox to control line spacing for that job.

For example, the FDP FORMAT command for print format LS132 assigns the value of 6 to the TOP-MARGIN parameter, meaning 6 lines will be skipped from the top of the page before the first line of data will be printed. Although the printer default is 66 lines per page with 6 blank lines at the top and 3 blank lines at the bottom, based on the DEVICE SGS specified during installation, any other combination may be defined by FDP. Any datafile then @SYM'ed to a queue utilizing this format will automatically use these new parameters.

The line spacing parameters defined by FDP establish defaults only for files which have no OS1100 margin controls embedded in them, such as text files created with an editor. Printfiles produced by application programs contain internally defined line spacing which will override the parameters defined to FDP as discussed in the next section. Optionally, FDP can specify that its format parameters override the line spacing specifications in the printfile. This is discussed in Chapter 3.

## 4.1.2 Controlling Margins and Line Spacing through Programming

---

The line spacing and margins of a print file may be controlled by an application program. A COBOL program can assign a file to "PRINTER" and control its line spacing with a LINAGE clause. A FORTRAN program can OPEN a file as type 'APRNTA'. In fact, any language which contains the ability to send an **ER SYMB\$** or **ER PRTCNS\$** can be used to control the margins, density and lines per page.

The **@HDG** command may also be used to select the total number of lines per logical page along with the number of blank lines at the top and bottom of the page. For example, the following ECL commands will print a file with 132 lines per page and no blank lines at the top and bottom of the page.

```
@CAT TEMP. . catalog a print file
@BRKPT PRINT$/TEMP
@HDG,N .M,132,0,0 . change top & bottom margin to 0
@DATA,L filename
@END
@BRKPT PRINT$
@SYM TEMP,1,PD132
```

The print density (lines per inch) is controlled in the same way as the lines per page, that is, the **SYMB\$** and **PRTCNS\$** executive requests may be used to indicate the print density as well as the lines per page. The **@HDG** command may also be used to specify the print density, as shown below:

```
@CAT TEMP. . catalog a print file
@BRKPT PRINT$/TEMP
@HDG,N .M,132,0,0,8 . indicate 8 lines per inch
@DATA,L filename
@END
@BRKPT PRINT$
@SYM TEMP,1,PD132
```

In the above examples the print file has embedded within it the margin control information. SPIN-X will recognize these commands and create an FCB (Forms Control Buffer) based on these values and download the FCB to the printer in front of the print file. If the Xerox JSL has FCB=PROCESS coded for the JDE used, then the Xerox will interpret skip to channel commands according to the way they are defined in this FCB. SPIN-X will, in this case, ignore the default line spacing values defined for that print format using the Format Definition Program (FDP). Refer, also, to the discussion of the MCONTROL parameter of the FORMAT command described in Section 3.1.

Setting FCB=PROCESS is the recommended method since it insures that SPIN-X knows exactly how many lines and pages are printed for accounting purposes and error recovery. It also requires the least amount of maintenance on the Xerox since FCB=PROCESS is all that is needed in the JSL. The only other alternative is to use VFU's on the Xerox as described below.

---

### 4.1.3 Controlling Margins and Line Spacing Using VFU's

---

Another method of controlling margins and line spacing is by using VFU's (Vertical Format Units) coded in the Xerox JSL. This method is most common in sites which have been running a Xerox LPS in an offline mode where FCB's are not allowed. Defining VFU's requires greater attention on the part of the Xerox administrator since an assignment must be made for each channel defined on a job by job basis. Also, since these commands are defined on the printer, SPIN-X has no way of knowing at the host what these channel assignments are. If the parameters defined using the FDP program are not compatible with the VFU definition, or if the print file has embedded margin controls which differ from the VFU definition then incorrect positioning of print lines and possibly extra page ejects may result. To insure that the FCB definitions do not interfere with VFU definitions, any site wishing to use the VFU feature should code **FCB=IGNORE** in their JSL on the Xerox. However, sites which have relied on VFU's in the past, are strongly encouraged to attempt to create a new JDL which has no VFU's defined and makes use of the FCB processing facility available with SPIN-X.

---

## 4.2 Using Unisys 1100/2200 Print Control Images

---

SPIN-X supports all standard Unisys 060 print control images. Refer to the *Unisys Series 1100 Executive Programmer Reference* for detailed information about using Unisys print control images.

---

## 4.3 Using Xerox Dynamic Job Descriptor Entries (DJDE's)

---

One of the features of the Xerox printer is the ability to modify the print format dynamically by inserting a DJDE within the file to be printed (see the *Xerox LPS Reference Manual* for details). SPIN-X fully supports the Xerox DJDE facility by allowing the site administrator to define an alphanumeric tag to identify DJDE's. For example, if the identification field for a DJDE has been defined as the character string:

```
%%DJDE
```

then each DJDE must begin with this character string in order for the laser printer to recognize that the record is a DJDE. The following example uses this identification field to specify a DJDE to modify the left margin:

```
%%DJDE MARGIN=(1.5,IN), END;
```

The DJDE identifier is defined using the DJDE parameter on the **FORMAT** command using the Format Definition Program (see Chapter 3). The distributed default value for all print formats is "%%DJDE". This value may be changed by the site administrator.

In the standard SPIN-X installation a DJDE packet will be automatically created and sent in front of the banner page data. It is followed by the banner page data which is followed by another DJDE packet used to define the JDL, JDE, PDE and the form specified for the specific format. The last thing to be sent is the report data. The report may also include the user's own DJDE packet(s) if that is desired.

### 4.3.1 Processing Carriage Control Associated with DJDE Records

---

Unisys carriage control commands are normally processed as "space and print" as opposed to the IBM 3211 standard of "print and space". SPIN-X must compensate for this by taking the spacing request for each line and associating it with the previous line when it is sent to the printer.

However, when the Xerox printers encounter DJDE records they ignore any "print and space" carriage control commands encountered on the line containing the DJDE record. To avoid losing the spacing request for the record following the DJDE, SPIN-X provides two alternatives.

#### Executing SPIN-X with the "H" Option

With the "H" option, the OS1100 spacing associated with a DJDE is honored, and the spacing for the line after the DJDE is also performed. DJDE records written with OS1100 line spacing of zero will not affect the line spacing of the printfile. See Section 4.1.4 of the *SPIN-X Installation Guide* for instructions on how to start SPIN-X with the "H" processor call option.

#### The OFFLINE=YES Parameter of the FORMAT Command

A parameter is provided for the FORMAT command which will instruct SPIN-X to process spacing requests associated with DJDE records as they would be processed in an offline tape environment. This is also equivalent to the way DJDE's would be processed in the IBM environment. The effect of setting OFFLINE=YES is that OS1100 spacing requests associated with a DJDE record are ignored; top of form requests on a DJDE record are processed. The spacing request for any line after the DJDE record is honored. Refer to Chapter 3 for information on how to specify parameters for the FORMAT command using the Format Definition Program.

## 4.4 Print File Errors

This section explains the possible error messages that may appear in the printed output from user print files. When these errors occur, only the portion of the print file before the occurrence of the error is printed. The possible print file errors and their explanations are listed below:

**000404: PRINTING TERMINATED; IO ERROR xx**

An I/O error occurred while processing the print file, and printing could not continue. The specific I/O error is indicated by the 2 digit number, xx which may be found in the *Unisys Series 1100 Executive Programmer Reference*.

**000405: ---- FILE FORMAT ERROR. FILE TERMINATED.**

The file or a portion of the file is not in a format that SPIN-X recognizes. The printing of the file abnormally terminated.

**000434: This file is empty.**

The file has no granules assigned to it.

**000471:AN ERROR OCCURRED IN A TYPE 060 IMAGE ON PAGE n :x**

The "n" is the page number and the "x" is the complete 060 image containing the syntax error. The printing of the file is terminated if auto-recovery has not been initiated using a type 060 "A" control image or if the SPIN-X "W" processor call option is not specified.

## 4.5 Transparent Data Mode

If a user data record is of character set type = 077 (63 decimal), then Central bypasses any translation. (The maximum record length of 512 bytes is still applicable, but the Format WIDTH= specification is inapplicable.) The character set type for the record is specified when the image is written via ER SYMB\$. The ER SYMB\$ packet with character set type = 077 in word 5 (S2) follows:

00	filename			
01				
02	function W\$		mode SPEC\$	
03	status	iostatus		
04	character count		image address	
05	ttn	077		
06	spacing			
07	(reserved)			
010				
011				

The ER SYMB\$ packet is documented in the *OS 2200 Exec System Software, Executive Requests Programming Reference Manual*, section 13.2.

---

## 5 Downloading Files from the Unisys to the Xerox

SPIN-X allows Unisys sites to download files over the online channel to be permanently stored on the laser printer's disk. This allows Xerox files to be kept on the Unisys for backup and provides an alternative to floppy diskettes for loading files. There are two different methods of downloading files:

- **HOSTCOPY:** The older, more cumbersome, method of downloading requires the printer to be taken out of production and rebooted under a special file transfer mode. Using Hostcopy with SPIN-X is discussed in detail in Section 5.1.
- **"FILE=" DJDE:** A new method of file transfer is available to users of Xerox printers running Version 2 or higher levels of the Xerox Operating System. This makes use of the DJDE "FILE=" command and may be used with all files except those of type SYS, TSK, SAF and OSD. This method is discussed below in Section 5.2.

---

### 5.1 Using the Hostcopy Utility

---

SPIN-X allows the Unisys 1100/2200 to take advantage of Xerox's Hostcopy utility, which is fully documented in the Xerox reference manual. Hostcopy provides a means of transferring files from the Unisys host to the Xerox EPS disk. It is designed to accept files in two different formats:

- **User-created files-** This includes source code in files of type JSL, FSL, CMD, MSC and TMP. These files must be in 80 character EBCDIC records and must be preceded by the \$\$\$START command and ended with the \$\$\$END command as discussed in the Xerox reference manual. For files which are created on the Unisys host in ASCII, SPIN-X provides a short utility program to translate your files to EBCDIC before transferring them to the Xerox. Instructions for using this program are given in Section 5.1.1.
- **Xerox Format Tape Files-** Xerox distribution tapes usually contain, as the last file on the tape, a file which is a concatenation of all previous files on the tape. This concatenated file is in a format which can be transmitted across the host's I/O channel to the Xerox using Hostcopy. SPIN-X provides a utility on the 1100/2200 which will read this file from the tape and deblock it from 128 and 512 byte blocks to 128 byte records stored in an SDF file. Details of how to use this utility are discussed in Section 5.1.2. A Xerox user with a tape drive configured to the printer may create tape files in this format. The method for creating a tape file compatible with Hostcopy is discussed in the Xerox reference manual in the chapter dealing with editor commands (specifically the TAPE NOEOF and TAPE ENDFILE commands). Note that files created in this manner do not need the \$\$\$START or \$\$\$END commands appended to them, nor do they need to be stored in EBCDIC records. The label record preceding each file is created by the command...

```
COPY TAPE WRITE LABEL filename.filetype
```

...and provides all the information required by Hostcopy to process and store these files. Details on creating and transferring these files are given in Section 5.2.

If your site uses Hostcopy frequently, it is suggested that a Unisys 1100/2200 symbiont queue be configured solely for the purpose of file transfer to the Xerox (in order to avoid conflict with other user print files). For example, this queue could be configured as a STATION LOCAL rather than an OUTPUT queue (refer to the *Unisys Series 1100 System Generation User Guide* for details of configuring symbiont queues). This allows files to be transferred to the Xerox via the **L\* device P queue keyin**. The symbiont queue used in the examples for this section is HOSTQ, however, any valid queue would suffice.

### 5.1.1 Transferring User-created Files

---

The user may find it convenient to use the Xerox editor to create the source code FSL and JSL files needed for the Xerox, but this resident editor is not very friendly and the usage of it requires taking the printer out of production. To create a file which can be transferred to the Xerox, begin by entering the file using CTS, IPF, ED or any other editor on the Unisys or a PC. The first line of this file should be:

```
$$$START filename.filetype
```

...where *filename* is the 1 to 6 character name under which the file is to be stored on the Xerox, and *filetype* is the 3 letter file type such as JSL or FSL. The last line of the file should be:

```
$$$END
```

For example, if the Forms Source Language file TSTFRM were created it would look like this:

```
$$$START TSTFRM.FSL
FORM TSTFRM;
PORTRAIT;
.
.
(FSL text)
.
.
END;
$$$END
```

This text should be saved in an SDF file. Before it can be sent to the laser printer, it must be converted from ASCII to EBCDIC. This can be accomplished by running the program UTILITY.TRANSLATE /ASCII-EBCDIC, which is provided with the SPIN-X package. The ASCII input file should be specified in field 1 of the processor call line. The output file should be specified in field 2 of the processor call line. After the program is executed, the output file is ready to be sent to the laser printer via the special Hostcopy transfer mechanism mentioned earlier in this section. The following Unisys ECL commands will translate the SDF file, IN, to the EBCDIC file, OUT, and place the output file on the transfer queue:

```
@ASG,A IN. . assign the input file
@ASG,A OUT. . assign the output file
@qual*UTILITY.TRANSLATE/ASCII-EBCDIC IN.,OUT. . xlate ASCII to EBCDIC
@SYM,U OUT.,1,HOSTQ . transfer the file
```

Next, the Xerox must be rebooted and placed in Hostcopy mode. This can be done by pressing the BOOT switch, typing **BD** and choosing boot option **H** (after this, the Xerox should inform you that it is ready to receive files from the host). At this time, make sure that the file to be transferred has first priority in the transfer queue and that no regular print files are on this queue, otherwise these print requests may be lost.

Now, issue the following SPIN-X keyin (assuming that SPIN-X has been started):

**L\* device P HOSTQ**

The Xerox console should respond with the following messages...

```
*TAPE NOW BEING PROCESSED*
CREATING FILE TSTFRM.FSL
END OF TAPE FOUND. IS ANOTHER FILE TO BE SENT?
Y OR N
```

Once you are finished sending all the files you wish to send, enter N. The Xerox will then respond with:

```
VARY OFFLINE AT HOST. THEN ENTER 'C'
```

At this point, SPIN-X may respond with a device error at the Unisys console. Enter T to terminate the print task (since the file has already been sent), which will lock out the device.

After the file has been transferred and the printer is rebooted under the normal Xerox OSS, the file can be compiled in the usual manner with the FDL or PDL command. If the file is to be read into the editor it must first be properly sequenced with line numbers in columns 73-80. These can be entered on the host before transferring the file, or this can be avoided by using the MERGE command on the Xerox editor which will read the file and automatically add the sequence numbers. In our example, to add the sequence numbers to the file, enter the following Xerox commands:

```
EDIT
MERGE TSTFRM.FSL
SAVE TSTFRM.FSL
File already exists. Do you want to overwrite? Y
CLEAR
END
```

The procedure described in this section (Transferring User-created Files), is effective for any of the textual files created on the Unisys of the types JSL, FSL, CMD, MSC or TMP. However, fonts, images, logos, JSL's and compiled forms must be created and sent in a different manner as discussed in the next section.

## 5.1.2 Transferring Xerox Format Tape Files

The procedures discussed in this section are useful for transferring files which are stored on tape in Xerox format. This is useful for an on-line only system which needs to load a Xerox-distributed tape such as a font, logo, patch or SST tape. It is also useful for sites with multiple Xerox's where files are to be transferred from a printer with a tape drive to other on-line only Xerox's. In the case of a Xerox-distributed tape, there will be a file at the end of the tape which is a concatenated file of all the files on the tape. In the case of a user-created tape, the concatenated file must have been created with the editor commands:

```

TAPES VOLINIT
EDIT
TAPES NOEOF
COPY TAPES WRITE LABEL file1.type
COPY TAPES WRITE LABEL file2.type
.
.
COPY TAPES WRITE LABEL fileN.type
TAPES ENDFILE
    
```

The concatenated Hostcopy-compatible file created by these commands can then be read by the Unisys host (using a utility provided by SPIN-X) and transferred to the laser printer with the following procedure:

1. Assign the tape file (the tape will be in 1600 BPI and unlabeled). Assuming that the tape volume has been called "XEROX1" the file can be assigned with the command:

```
@ASG,TJ TAPES.,U9V/////Q,XEROX1,,NORING
```

If the tape is a multiple file tape and the file to be copied is not the first file, then the FURPUR @MOVE command can be used to position the tape accordingly. For example, if the tape is a Xerox font tape with four rotations of a single font it will have each font in a separate file followed by a file called "ENDFIL.END". The file after this is the one which contains the concatenated version in Hostcopy form. The tape can be positioned to read this file with the command:

```
@MOVE TAPES.,5
```

...since there are 5 files preceding this file (including ENDFIL.END).

2. Assign an output file (e.g., FILEOUT) on disk to receive the copied tape file, link this file and the Xerox tape to the SPIN-X utility and execute the program as shown below:

```

@ASG,A FILEOUT. . assign the output file
@USE 13,FILEOUT. . link it to the program
@USE 12,TAPES. . link the Xerox tape too
@XQT qual*UTILITY.READ/XEROX-FILE
    
```

3. Finally, after the program has executed, the output file is ready to be sent to the laser printer via the special Hostcopy transfer queue mentioned earlier in this section. Set up the Xerox in Hostcopy mode and transfer the output file to the printer as follows:

```
@SYM,U FILEOUT.,1,HOSTQ . transfer the file
```

Once Hostcopy has finished reboot the Xerox and resume normal operations. All of the transferred files should now be ready for editing, compiling or executing.

## 5.2 Using the FILE DJDE

A new DJDE command is available in version 2.0 and higher levels of the Xerox Operating System which allows files to be downloaded to the printer's disk over the online channel while the printer is running normal print jobs. This can be used in most cases as an alternative to the Hostcopy procedure which requires taking the printer out of production. The FILE DJDE is fully documented in the Xerox reference manual for your specific printing system. Just as with Hostcopy, there are two different methods for using the FILE DJDE dependent on whether the file is a "card image file" or in "LPS Labeled Format". Both of these are discussed below in detail.

### 5.2.1 Downloading LPS Labeled Files

A COBOL utility program has been provided with SPIN-X which allows downloading files stored in "Xerox LPS Labeled Format". This is the format in which files are stored on the Xerox disk. A key characteristic of this format is the header record which precedes the file. This format includes files on font and logo tapes distributed by Xerox or third party vendors, and all files on Xerox System Software Tapes (note, however, files of type SYS, TSK, SAF and OSD are not allowed to be downloaded with the FILE DJDE). Also, files taken as output from SPIN-X Host Form Design Language can be downloaded using this method. Any file copied to tape from the Xerox using the command:

```
COPY TAPE WRITE LABEL filename
```

will also be in this format. A problem which can be encountered when downloading these files is that any trailing blanks on the end of the records will get stripped off during printing. To avoid this, and to make inserting the required DJDE command easier, SPIN-X provides a utility program which will read in a Xerox LPS Labeled file, append termination symbols to the end of each record, reblock each record to an even word boundary, and prefix the file with the proper "FILE=" DJDE command. More than one Xerox file can be concatenated at a time and processed in a single execution of the program. The output can be sent directly to the printer with the "@SYM" command and may be mixed with regular print jobs. When this file reaches the printer it will be stored on disk instead of printed and a message to that effect will be displayed. One banner page will be printed for each group of files processed in this manner. The source code for this program is provided in COBOL in the file SPIN-X\*UTILITY.FILECOPY. A sample runstream is shown below which reads a file from the Unisys disk and reformats it for downloading with the FILE DJDE.

```
@ASG,A INPUT-FILE.           . file to be reformatted
@USE HFDLHC-IN.,INPUT-FILE.
@ASG,U FILE-DOWNLOD.         . name of formatted file to be created
@USE HFDLHC-OUT.,FILE-DOWNLOD.
@XQT SPIN-X*UTILITY.FILECOPY
@FREE FILE-DOWNLOD.
@SYM,U FILE-DOWNLOD.,1,LS132
```

The above addstream assigns the file named INPUT-FILE, which is assumed to be in Xerox LPS labeled format, and links it to the internal name of HFDLHC-IN. The output will go to the file named FILE-DOWNLOD linked to the internal name HFDL-OUT. (If FILE-DOWNLOD already exists, use ASG,A). As the program executes, each record is checked to determine if it is a valid label record and if the file is of the proper type. If so a message is displayed showing the name of each file as it is processed. After the utility is executed the output file is freed and printed to the desired format, in this case LS132. The format used is immaterial since no data will be printed except for the banner page. Sites wishing to use this program must insure that the DJDE command flag symbol specified on the IDEN statement on the printer and in the FDP input file matches the DJDE flag symbol defined in the program. It is distributed with the SPIN-X default value of ASCII '%DJDE'.

## 5.2.2 Downloading Card Image Files

---

Card image files are text files which a user may have created using an editor on the host or a PC. This would include Forms Source Language and Jobs Source Language files which a user may prefer to create and maintain without having to use the Xerox printer's editor. Since these files will not have a standard Xerox disk file structure and label record, they must be handled separately. A FILE DJDE must precede the file which specifies its name and type. There must also be a "FILE=(,C)" DJDE following the last record. No utility program is included for performing this since it can be accomplished with a system editor. An example of downloading an FSL file follows.

```

%%DJDE FILE=( SAMPLE , FSL , C , P , 999 ) , END ;
FORM SAMPLE ;
PORTRAIT ;
GRID IS FMT6 ;
.
.
.
END ;
%%DJDE FILE=( , , C ) , END ;

```

The above file can be of variable length records and does not require a termination symbol in the last column since card image type files are automatically space filled to column 80. Once stored on the LPS disk they must first be brought into the Xerox editor using the MERGE command to have line numbers appended before editing can be performed.



Sites may choose to run a separate SPIN-X job from the production SPIN-X for the purposes of testing new releases. A test environment may be particularly useful to sites having multiple printers and that develop user exits. With multiple SPIN-X runs, the user exit programmer could test the new SPIN-X on one printer while the production SPIN-X continues driving another.

---

## 6.1 Steps for Setting-up a Test SPIN-X

---

Four steps are involved in setting-up a test SPIN-X:

1. Install the SPIN-X software to be tested under a different qualifier than the production environment.

Rather than copy SPIN-X into SYSS\$PRINTER\$\*SPIN-X as specified in Section 4.1.1 of the Installation Guide perform the @COPY,G into a different file, e.g. SPIN-X4R4\*SPIN-X. Also, utilize a different @QUAL qualifier, e.g. SPIN-X4R4, for the install of the SPIN-X test software than was used for the production install as shown in Section 4.1.1 of the Installation Guide.

2. Create a new \*FDP-INPUT file under the test qualifier.

The test \*FDP-INPUT file may contain formats already defined in the production \*FDP-INPUT, however, device statements must be unique between the two because separate SPIN-Xs cannot have the same printer assigned simultaneously. A printer normally under the production SPIN-X that needs to be temporarily used for testing could be commented-out of the production \*FDP-INPUT and restored once testing is complete. The following ECL would execute FDP for a test environment:

```
@QUAL SPIN-X4R4
@*FDP.FDP
```

**NOTE:** The FDP/TOPRODUCTION addstream should not be performed from the test environment.

3. The SPIN-X batch job ECL should be modified for the test run.

The projectid or an @QUAL for the run should correspond to the qualifier under which the test SPIN-X was installed under in Step 1 above, e.g. SPIN-X4R4. An example of ECL for a test SPIN-X batch job is shown in Section 6.2.

4. A keyin prefix other than L\* should be selected for the test SPIN-X job.

The prefix is specified to the SPIN-X processor (without the "\*") in the first field, element sub-field as follows:

```
@TPF$.SPIN-X,AK P
```

SPIN-X suffixes an "\*" to the value specified on the processor call therefore the SPIN-X keyin for the example above is "P\*". The prefix provided on the processor call may be up to 7 characters in length. The prefix should consist of characters A through Z, 0 through 9, but the first character should not be 0 through 9. The initiation keyin for the test SPIN-X based upon the above example follows:

```
P* LY I
```

---

## 6.2 Example ECL for Test SPIN-X Batch Job

---

```
@RUN,/KL SP4R4,account#/userid,SPIN-X
@QUAL SPIN-X4R4
@ASG,AX *SRI$DATABASE
@USE SRI$DATABASE,*SRI$DATABASE
@ASG,AX *SRI$FLATFILE
@USE SRI$FLATFILE,*SRI$FLATFILE
@ASG,AX *FORMS-DEF.
@USE FORMSDEF,*FORMS-DEF.
@COPY,A *SRO.SPIN-X,TPF$.
@TPF$.SPIN-X,AK P
@PMD,PALBE
@FIN
```

This section discusses the internal environment of SPIN-X, as well as its external environment.

Section 7.1 gives a basic explanation of how SPIN-X works internally, that is, it describes the functions that SPIN-X performs in order to send print files to the printer. Section 7.2 explains the purpose of the SPIN-X bus and tag cable adapters. Section 7.3 discusses several considerations for the connection of the Unisys host to the IBM 3211 printer environment, which is necessary for the proper placement of the SPIN-X bus and tag cable adapters.

---

## 7.1 SPIN-X Internals Overview

---

This section is intended to provide insight into the key concepts incorporated in the internal design of SPIN-X. The information presented is in no way intended as a description of the internal logic, but this should not cause any concern because SPIN-X is not released in source form.

The SPIN-X software consists of two sections: the code for the 1100/2200 and the code for the Xerox. The code for the 1100/2200 performs the print queue manipulation, file interpretation, channel program generation and error recovery. The code for the Xerox is standard Xerox Job Source Language (JSL) defining the on-line environment for the printer and the print format descriptions.

The SPIN-X software for the 1100/2200 is a real-time, multi-activity program which interfaces to the Unisys 1100/2200 EXEC in order to perform many of its specialized functions. Because SPIN-X is a user-program and not local code in the Unisys 1100/2200 Executive, it will appear as a batch run. The internal SPIN-X functions include:

1. Accessing the Unisys 1100/2200 symbiont queues to select print files; print files reside in the symbiont queues ordered in a first-in-first-out basis within priority classes,
2. Interpreting the print files,
3. Generating the channel program and acting upon status information from the printers,
4. Interfacing with the Unisys operator via the L\* keyin and console messages.

An illustration of these internal SPIN-X functions is shown in Figure 7-1. The numbers shown in the figure correspond to the numbered functions listed above.

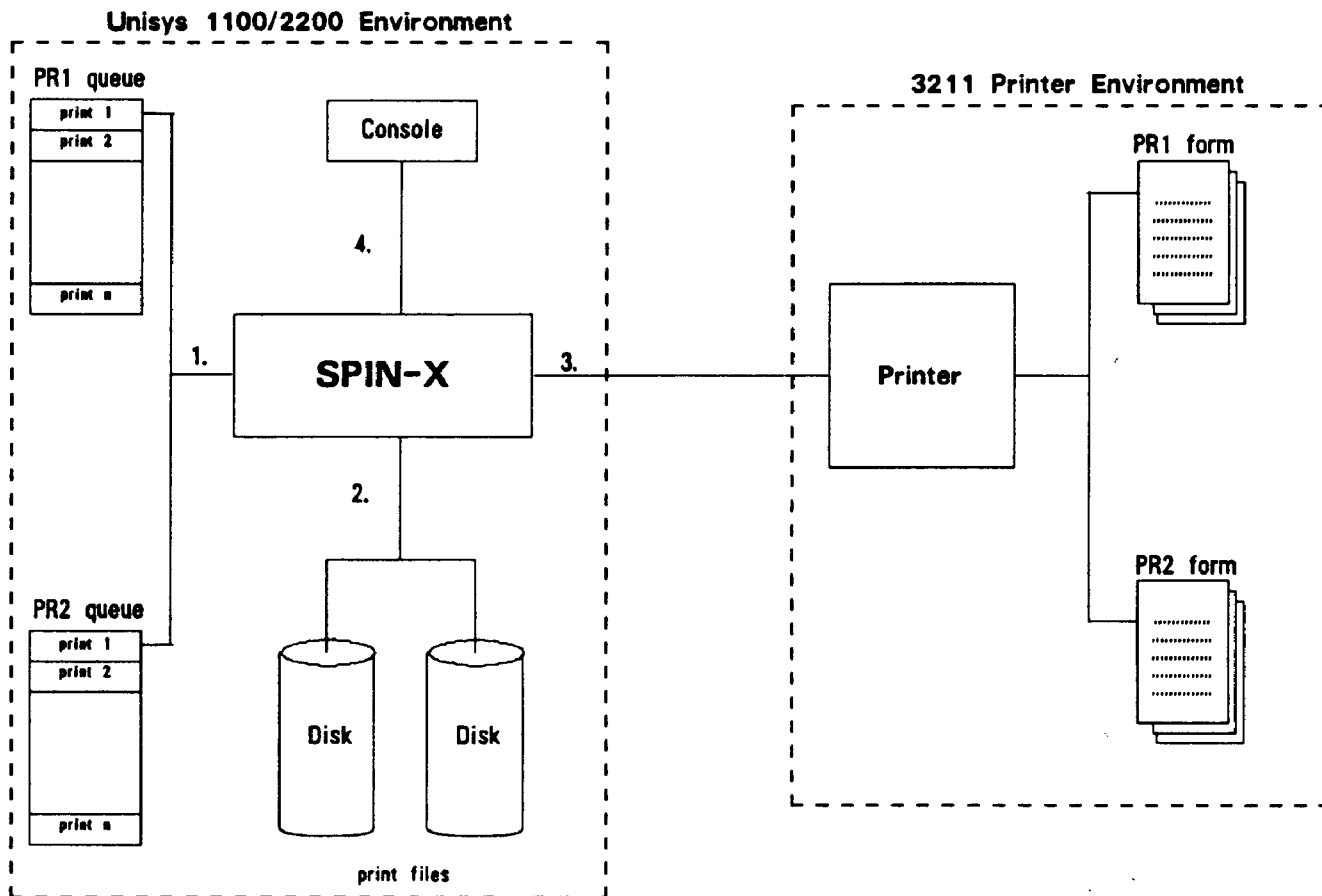


Figure 7-1 Diagram of Internal SPIN-X Functions

## 7.2 Why the Bus and Tag Cable Adapters are Needed

The SPIN-X bus and tag cable adapters consist of a pair of 48-pin cables approximately three feet in length. The adapter cables open the pins carrying the 1100/2200 6 volt Propagate Select Out (PSO) signal. Some Unisys devices require this signal, but IBM 3211 devices cannot tolerate it. **Therefore, if your site daisy chains Unisys devices with IBM emulators, then all IBM devices must follow the Unisys devices in the chain to assure that the Unisys devices receive the PSO signal.** In this case, the bus and tag cable adapters must be placed between the Unisys devices and the IBM 3211 devices in the chain.

In general, the bus tag cable adapters must be placed immediately after the Unisys environment, but immediately before entrance to the IBM environment. This organization is illustrated by Figure 7-2. Note, in the special case where an intermediate box (such as a channel extender or channel interface switch) is used, the bus and tag cable adapters must be placed before any box of this type.

**NOTE:** The bus and tag cable adapter is not required for sites connecting printers to a FIPS channel. Therefore, if the daisy-chain contains all FIPS compatible devices, then the printers may be placed anywhere in the chain.

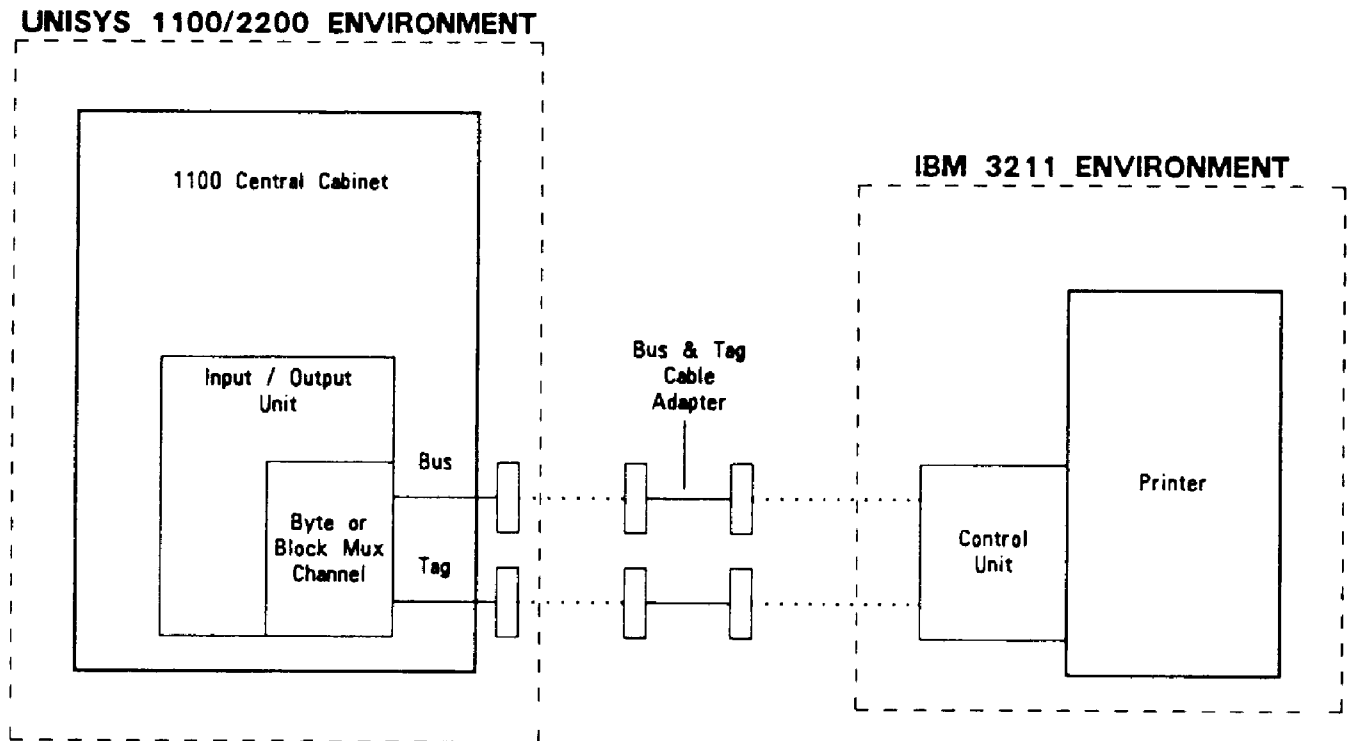


Figure 7-2 The Basic 1100/2200 to 3211 Connection

## 7.3 The Unisys to IBM 3211 Environment Connection

This section discusses various considerations for the connection of the Unisys host to the IBM 3211 printer environment. The placement of the SPIN-X bus and tag cable adapters depends on the type of connection(s) between the host and printer.

### 7.3.1 Direct Connection

In the case where the Unisys 1100/2200 host is directly connected to the printer, one end of the bus and tag cable adapters is connected to a standard Unisys bus and tag cable, while the other end is connected to the printer's controller.

### 7.3.2 Connection Via a Unisys Byte Channel Transfer Switch

In the case where a Unisys Byte Channel Transfer (BCTS) switch, model 2521, is being used for device connection, one end of the SPIN-X bus and tag cable adapters will connect to the standard bus and tag cable leading from the BCTS, while the other end connects directly to the printer's control unit.

### 7.3.3 Connection Via a Channel Interface Switch

---

When the printer is being shared between many hosts, some kind of channel interface switch (often referred to as a matrix switch) is necessary. The SPIN-X bus and tag cable adapters are placed after the Unisys bus and tag cable but immediately before any switch of this type.

### 7.3.4 Connection Via a Channel Extender

---

When the printer is located farther than a standard Unisys bus and tag cable is allowed, some form of channel extension is required. The SPIN-X bus and tag cable adapter must be placed after the Unisys cable but before the channel extender is installed.

### 7.3.5 Considerations of Multiple Host / Device Connections

---

In some cases, multiple hosts may wish to share one or more printers. If a Unisys Byte Channel Transfer Switch (BCTS) is used to share the printer(s), then a bus and tag cable adapter is needed for each 3211 printer which is connected to the BCTS. In this case, the bus and tag cable adapter(s) would be placed directly after the cable leading from the BCTS, but before the printer's control unit.

If some type of channel interface switch is used to share the printer(s), then a bus and tag cable adapter is needed for each Unisys host that uses SPIN-X to control the 3211 printer(s). The bus and tag cable adapters would be placed in the same location as discussed in Section 7.3.2.

**NOTE:** If your site daisy chains Unisys devices with IBM devices, then all 3211 devices must follow the Unisys devices in the chain to assure that the Unisys devices receive the PSO signal. In this case, the bus and tag cable adapter must be placed between the Unisys devices and the IBM 3211 devices in the chain. This restriction is inapplicable to printers connected to a FIPS channel.

A user exit is a site developed subroutine that is collected into SPIN-X to replace a predefined function of SPIN-X. This chapter defines the entry points that SPIN-X recognizes to substitute a user exit subroutine for its default subroutine.

User exits are considered local code. Support for design, development, debugging, and maintenance of user exits is the customer's responsibility. GSURF recommends that user exits be attempted by only knowledgeable MASM programmers. The user exits supplied on the SPIN-X release tape are supported to the extent that GSURF issues bug fixes solely to the code it provides; any modifications are the customer's responsibility.

This section provides the information you will need to locally modify the SPIN-X processor if desired. Section 8.1 describes each SPIN-X defined user exit in detail. Section 8.2 gives information about certain data structures which are necessary to write the user exits. Section 8.3 explains how the character translation table(s) may be modified. Finally, Section 8.4 provides the steps for collecting SPIN-X with the user exits.

---

## 8.1 SPIN-X Defined User Exits

---

This section supplies the SPIN-X information needed to write user exit routines. It also defines conventions that should be followed in order for the exits (and SPIN-X) to function properly. SPIN-X recognizes user exits to control format selection, accounting entries, and execution before the banner page, before the trailer page, and after the trailer page. The SPIN-X data structures which are relevant to user exits are described in Section 8.2.

- SPIN-X runs in basic mode.
- SPIN-X checks for the inclusion of a user exit by testing its predefined entry point to check its address. If the entry point is zero, SPIN-X jumps to its default subroutine for processing. If the entry point is non-zero, SPIN-X jumps to the user exit instead of its default subroutine. Sections 8.1.1 - 8.1.5 explain the arguments passed to each user exit and the entry point SPIN-X checks.
- User exits should not cause dynamic memory expansion, e.g. ER MCORES\$, because SPIN-X runs at real-time and cannot be swapped to obtain memory.
- User exit programmers need to consider writing re-entrant code because SPIN-X may be used to drive multiple printers simultaneously. SPIN-X is already registered for test and set queuing.
- The major register set is available to the user exit except R15 should never be changed.

- SPIN-X saves a user exit's registers including Designator Register between calls for processing the same SMOQUE entry except A0, R0, and X0 are not preserved by SPIN-X between user exit calls. The register contents are not preserved by SPIN-X between processing separate SMOQUE entries. The user exits are entered in third word mode.

- Starting with SPIN-X 4R1, the user exits are collected into their own IBANK. SPIN-X jumps to the user exit routine using X11. Therefore, return from the user exit to SPIN-X could be made by the following instruction:

```
LBJ X11,0,X11
```

- Various SPIN-X data structures are passed by address to the user exit routine. Most of the fields should not be altered because that may abnormally impact SPIN-X's functioning. Section 8.2 discusses the data structures passed to the user exits and identifies which fields are writable and which should be treated as read-only.
- The EQUFs that define the SPIN-X data structures are provided in the \*SRO program file. When moving to a new release of SPIN-X each user exit should be recompiled using the latest MASM omnibus elements supplied from the Release Tape. The omnibus elements are provided in the \*SRO file. SPIN-X 4R4E was assembled with MASM level 6R1F.
- Two 64 word data areas are allocated in SPIN-X data structures for writable use of the user exits. The fields PCBSITE and FCTSITE are discussed in Section 8.2. These fields are allocated for the individual device so no test and set protection is needed for these fields. If more work memory is required by the user exit, then another dbank could be created in the addressing hole between the end of the user exit ibank UEXIT\$IBANK and the beginning of the SPIN-X control dbank MAIN\$DBANK. The user exit dbank can be based upon BDR 1; it should be rebased upon re-entry by the user exit. Currently, SPIN-X bases its control dbank on BDR 2, and its stack bank on BDR 3. Both these banks need to be visible to the user exit routine because data items in both banks may be referenced by the user exit.
- The SPIN-X 4R4E relocatable provided in the \*SRO file was produced by level 33R1E of @MAP.
- Each call to a user exit results in one image being returned to SPIN-X at a time. In most cases, SPIN-X continues jumping to the user exit until the latter sets a flag to terminate the calls for that SMOQUE entry. A 128 word buffer EQUF'ed by FCTIBBUF is passed to the user exit subroutine for returning a single image or printer control to SPIN-X for transfer to the printer.

Sections 8.1.1 - 8.1.6 provide specific information on each SPIN-X user exit argument passing conventions. Section 8.2 provides definitions of the SPIN-X data structures such as the Printer Control Block (PCB) and SPIN-X File Control Table (FCT) that are passed to the user exits. Section 8.4 contains instructions for collecting SPIN-X with user exits.

SPIN-X recognizes user exits to control before the banner page, the banner page, format selection, trailer page, after the trailer page, and accounting entries. SPIN-X supplies the MASM source code to its subroutines that perform these functions to provide a foundation for the development of user exits.

## 8.1.1 Beginning Printer Controls User Exit

### User Exit Entry Point: UXRBEGPC

- **Description**

This user exit is designed to permit printer controls, e.g. DJDEs, to be sent before generation of the banner page. In the SPIN-X supplied subroutine, a Xerox RSTACK is sent if the FDP Device command includes the JOB\_DISPLAY=Yes parameter xor the INIT\_RSTACK=Yes parameter and the previous print was terminated by an Intervention Required condition.

- **Environment/Point of Processing**

The beginning printer controls user exit is called before any printer controls or data for the banner page have been sent. At this point of processing, the SMOQUE entry for the print request has been obtained, the file has been assigned, and the file's header label is in FCTLBL50.

- **Register contents when control is passed to the user exit routine:**

**A4**= a pointer to the Printer Control Buffer (PCB).

**A5**= a pointer to the File Control Table (FCT).

**A6**= a pointer to the Format Description Table (FDT).

**A7**= a pointer to the SMOQUE\$ entry for the print request (SQE).

**A8**= a pointer to the FCTIBBUF (Image Builder Buffer within the FCT).

**A9**= 0 (the first time UXRBEGPC is called for each SMOQUE entry. ).

- **(Expected) register contents when control is passed back to SPIN-X:**

**A10**= the number of words to be printed.

**A11**= the number of lines to space before printing the image.

**A12**= the termination flag. SPIN-X tests this flag to determine whether to call the exit again. If the flag is equal to zero, then the exit is not finished processing the beginning printer controls user exit and is called again. If the flag is non-zero, then the exit is finished and is not called again for the current print file.

**A14**= (Not used with Xerox printers) Should always contain 0.

- **Programming Considerations**

If the beginning printer controls user exit is supplied, SPIN-X calls UXRBEGPC for each SMOQUE entry processed. Continuous jumps are made to UXRBEGPC for each image or printer control, e.g. DJDE, to be returned until the termination flag is set (A12 is non-zero). The programmer will probably need to establish a counter (A9) to assist in logic flow control for the user exit. Because A9=0 the first time UXRBEGPC is called for each SMOQUE entry, the programmer will know when this user exit is initially called.

## 8.1.2 Banner Page User Exit

---

### User Exit Routine Name: UXRBANPG

- **Description**

The banner page user exit allows a site to create its own individual header page to precede each print file printed by SPIN-X. By default, SPIN-X generates a banner page designed by Georgia State University.

- **Environment / Point of Processing**

The banner page user exit routine is invoked before the transfer of a print file. At this point of processing, the SMOQUE entry for the print request has been obtained, the file has been assigned, and the file's header label is in FCTLBL50.

- **Register contents when control is passed to the user exit routine: UXRBANPG**

**A4**= a pointer to the Printer Control Buffer (PCB).

**A5**= a pointer to the File Control Table (FCT).

**A6**= a pointer to the Format Description Table (FDT).

**A7**= a pointer to the SMOQUE\$ entry for the print request (SQE).

**A8**= a pointer to the Image Builder Buffer within the FCT.

**A9**= 0 (the first time UXRBANPG is called).

- **(Expected) register contents when control is passed back to SPIN-X:**

**A10**= the number of words in the line to be printed.

**A11**= the number of lines to space before printing the image.

**A12**= the termination flag. SPIN-X tests this flag to determine whether to call the exit again. If the flag is equal to zero, then the exit is not finished processing the banner page and is called again. If the flag is non-zero, then the exit is finished and is not called again for the current SMOQUE entry.

**A14**= (Not used with Xerox printers) Should always contain 0.

- **Programming Considerations**

If the banner page user exit is supplied, then SPIN-X will call UXRBANPG for each SMOQUE entry processed. A jump is made to UXRBANPG for each image or printer control, e.g. DJDE, to be returned for each SMOQUE entry until the termination flag is set (A12 is non-zero). A 0200 octal words buffer, FCTIBBUF, is provided by A8 for returning the image/printer control to SPIN-X. Register A9 is equal to zero the first time UXRBANPG is called for each SMOQUE entry, thus the user exit is notified when the next printfile is being processed.

### 8.1.3 Format Processing User Exit

#### User Exit Entry Point: UXRFORSL

- **Description**

The print format selection user exit allows extra control in how each available print format will physically appear. Specifically, this exit is used to build the Xerox DJDE that is sent to the printer before each print file in order to create the print characteristics for the requested print format. If this user exit is not supplied, SPIN-X uses its own print format selection routine, which builds and sends the DJDE mentioned above and sends a Xerox RPAGE to advance to the front of a new sheet of paper for the print file. This SPIN-X routine gets information about each format from the Format Definition Table (FDT) which is built by SPIN-X from information that is supplied to FDP.

Basically, the SPIN-X supplied print format selection routine generates a DJDE from the information provided to FDP for the given format. If more control of print formats is desired, then the format selection user exit can be written to include the additions.

- **Environment / Point of Processing**

The format processing user exit routine is invoked after the banner page has been transferred but before the print file has been sent. At this point of SPIN-X processing, the SMOQUE entry for the print file has been obtained, the FDP format information has been fetched, and the header label for the printfile has been read into FCTLBL50.

- **Register contents when control is passed to the user exit routine:**

**A4**= a pointer to the Printer Control Buffer (PCB).

**A5**= a pointer to the File Control Table (FCT).

**A6**= a pointer to the Format Description Table (FDT).

**A7**= a pointer to the SMOQUE entry for the print request (SQE).

**A8**= a pointer to the FCTIBBUF.

**A9**= 0 (the first time UXRFORSL is called for each SMOQUE entry).

- **(Expected) register contents when control is passed back to SPIN-X:**

**A10**= the number of words in the image to be printed.

**A11**= the number of lines to space before printing the image.

**A12**= the termination flag. SPIN-X tests this flag to determine whether to call the exit again. If the flag is equal to zero, then the exit is not finished processing the banner page and it is called again. If the flag is non-zero, then the exit is finished and is not called again for the current SMOQUE entry.

**A14**= Not used with Xerox printers, should always contain 0.

- **Programming Considerations**

If the format selection user exit is supplied, then SPIN-X will call UXRFORSL for each SMOQUE entry processed. A jump is made to UXRFORSL for each image or printer control, e.g. DJDE, to be returned for each SMOQUE entry until the termination flag is set (A12 is non-zero). A 0200 octal words buffer, FCTIBBUF, is provided by A8 for returning the image/printer control to SPIN-X. For example, the data returned in FCTIBBUF might be a data image, DJDE, or RPAGE. Register A9 is equal to zero the first time UXRFORSL is called for each SMOQUE entry, thus the user exit is notified when the next printfile is being processed.

## 8.1.4 Trailer Page User Exit

---

### User Exit Entry Point : UXRTRAPG

- **Description**

The trailer page user exit routine allows a site to generate a trailer page to be printed after each file printed by SPIN-X. By default, SPIN-X does not generate a trailer page for the Xerox LPS.

The source to the default trailer page is supplied in the \*SRO file that was copied during installation from the SPIN-X 1100/2200 Release Tape. This source may be used in your user exit; however, the entry point name, FHRTRAPG, must be changed to UXRTRAPG before it is collected into SPIN-X.

- **Environment / Point of Processing**

The trailer page user exit routine is invoked after the transfer of a print file. At this point of processing, the SMOQUE entry for the print request is still available.

- **Register contents when control is passed to the user exit routine:**

**A4**= a pointer to the Printer Control Buffer (PCB).

**A5**= a pointer to the File Control Table (FCT).

**A6**= a pointer to the Format Description Table (FDT).

**A7**= a pointer to the SMOQUE entry for the print request (SQE).

**A8**= a pointer to the FCTIBBUF.

**A9**= 0 (the first time UXRTRAPG is called for each SMOQUE entry).

- **(Expected) register contents when control is passed back to SPIN-X:**

**A10**= the number of words in FCTIBBUF to be printed.

**A11**= the number of lines to space before printing the image.

**A12**= the termination flag. SPIN-X tests this flag to determine whether to call the exit again. If the flag is equal to zero, then the exit is not finished processing the trailer page, and it is called again. If the flag is non-zero, then the exit is finished and it is not called again for the current print file.

**A14**= Not used with Xerox printers, should always contain 0.

- **Programming Considerations**

If the trailer page user exit is supplied, then SPIN-X will call UXRTRAPG for each SMOQUE entry processed. A jump is made to UXRTRAPG for each image or printer control, e.g. DJDE, to be returned for each SMOQUE entry until the termination flag is set (A12 is non-zero). A 0200 octal words buffer, FCTIBBUF, is provided by A8 for returning the image/printer control to SPIN-X. Register A9 is equal to zero the first time UXRTRAPG is called for each SMOQUE entry, thus the user exit is notified when the next printfile is being processed.

## 8.1.5 Ending Print Controls

### User Exit Entry Point: UXRENDPC

- **Description**

The ending print control user exit may be used to send any Xerox-defined print controls after the printing of a trailer page (if any). If this user exit is not supplied, SPIN-X uses its own ending print controls routine. The SPIN-X ending print control subroutine sends an RSTACK to the printer if the FDP JOB\_DISPLAY=NO parameter in the DEVICE command is in effect.

- **Environment / Point of Processing**

The ending print control user exit routine is invoked after the transfer of a trailer page (if any). At this point of processing, the SMOQUE entry for the print request is still available.

- **Register contents when control is passed to the user exit routine:**

A4= a pointer to the Printer Control Buffer (PCB).  
 A5= a pointer to the File Control Table (FCT).  
 A6= a pointer to the Format Description Table (FDT).  
 A7= a pointer to the SMOQUE entry for the print request (SQE).  
 A8= a pointer to the FCTIBBUF.  
 A9= 0 (the first time UXRENDPC is called for each SMOQUE entry).

- **(Expected) register contents when control is passed back to SPIN-X:**

A10= the number of words from FCTIBBUF to be printed.  
 A11= the number of lines to space before printing the image.  
 A12= the termination flag. SPIN-X tests this flag to determine whether to call the exit again. If the flag is equal to zero, then the exit is not finished processing the ending print controls and it is called again. If the flag is non-zero, then the exit is finished and is not called again for the current print file.  
 A14= Not used with Xerox printers, should always contain 0.

- **Programming Considerations**

If the ending print control user exit is supplied, then SPIN-X will call UXRENDPC for each SMOQUE entry processed. A jump is made to UXRENDPC for each image or printer control, e.g. an RSTACK, to be returned for each SMOQUE entry until the termination flag is set (A12 is non-zero). A 0200 octal words buffer, FCTIBBUF, is provided by A8 for returning the image/printer control to SPIN-X. Register A9 is equal to zero the first time UXRENDPC is called for each SMOQUE entry, thus the user exit is notified when the next printfile is being processed.

## 8.1.6 Symbiont End of Processing Log Entry

---

### User Exit Entry Point: UXRLOG14

- **Description**

This user exit may be used to generate a symbiont end of processing log entry. This log entry is commonly type 10112, Subtype 2 for Exec 40R2 or higher and Type 14 for previous Execs. Refer to Appendix A.1.1 and A.1.2 for a description of the format that SPIN-X provides for both log entries.

- **Environment / Point of Processing**

The end of processing log entry user exit is called after any trailer page has been transferred and after any ending printer controls have been sent. The SMOQUE entry for the print file is passed to the user exit.

- **Register contents when control is passed to the user exit routine:**

A4= a pointer to the Printer Control Buffer (PCB).

A5= a pointer to the File Control Table (FCT).

A6= a pointer to the Format Description Table (FDT).

A7= a pointer to the SMOQUE entry for the print request (SQE).

A8= a pointer to FCTIBBUF.

- **(Expected) register contents when control is passed back to SPIN-X:**

No return termination flag is necessary for this exit because it is only called once for each SMOQUE entry.

- **Programming Considerations**

The programmer should consider whether the user exit needs to generate ASCII, Fieldata, or both log entries depending on the Exec levels to be supported. The Exec level in ER MCT\$ format may be obtained from FCTEXEC for the system version and FCTELOCL for the site version. The system model, e.g. 2200/600, may be obtained from FCTSYSID. Each of these data items (FCTEXEC, FCTELOCL, FCTSYSID) is in ASCII. FCTLLP gives the number of lines transferred for this SMOQUE entry. (This number includes the banner page). FCTLLP gives the number of lines transferred, FCTPR provides the number of pages re-transferred, and FCTPS gives the number of pages skipped.

---

## 8.2 SPIN-X Data Structures Passed to User Exits

---

This section provides the information about each SPIN-X data structure that is provided to write user exits. Each SPIN-X device is allocated one of each type of data structure.

Each data structure is composed of fields, each of which identifies a MASM \$EQUF. The fields that are mentioned are of two types:

1. **Read-only fields** - These fields are intended as information to the user and should not be modified (e.g., a SMOQUE entry field).
2. **Alterable fields** - These fields are intended as working areas for the programmer and may be modified (e.g., FCTIBBUF).

Each data structure that SPIN-X supplies is located within the program file:

*qual*\*SRO.

...where, *qual* is the qualifier used to catalogue the SRO program file during the installation of SPIN-X (this file is usually SPIN-X\*SRO). The omnibus elements within this program file have been compiled using MASM definition mode assembly (\$DEF).

The SPIN-X supplied subroutines require their data structures referenced in \$INCLUDE directives to be linked to the definition mode elements using an @USE LIB,\*SRO command. The following example demonstrates how the SPIN-X definition mode element, SQE, that defines a SMOQUE entry is utilized:

```
$INCLUDE 'LIB.SQE'
```

...where, LIB, was specified in the @USE command to link to \*SRO. To reference a field in the SQE data structure the \$EQUF for that field is specified. The following instruction uses the \$EQUF, SQERUNID, from the SQE definition mode element to obtain the runid from the SMOQUE entry of the run that @SYM'ed the printfile:

```
DL    A1,SQERUNID,X7
```

In the example above, index register X7 contains the beginning address of the SMOQUE entry.

Two alterable fields are provided to the user exit programmer for editing data: PCBSITE and FCTSITE. PCBSITE remains unchanged by SPIN-X for the life of the program execution. FCTSITE is null filled each time a device is initiated by the SPIN-X "I" or "P" keyins.

The following charts identify each data structure and its available fields. (Type A = Alterable fields and R = Read-only fields):

## 8.2.1 Format Definition Table (FDT)

The FDT contains the information from the current FDP Format command.

<b>\$EQUF Name</b>	<b>Description</b>	<b>Length (words)</b>	<b>Type</b>
FDTFNM	The format name (6 ASCII Characters).	2	R
FDTBITS	A word of flag bits indicating format orientation (0: Duplex, 1: Landscape, 2: Shift, 3-35: Reserved).	1	R
FDTLPS	The number of logical pages per side (half word).	1/2	R
FDTLPP	The number of lines per logical page (half word).	1/2	R
FDTJDL	The name of the Job Descriptor Library (6 ASCII Characters).	1 1/2	R
FDTJDE	The name of the Job Descriptor Entry (6 ASCII Characters).	1 1/2	R
FDPDE6	The PDE name used for low density printing (6 ASCII Characters).	1 1/2	R
FDPDE8	The PDE name used for high density printing (6 ASCII Characters).	1 1/2	R
FDTDLW	The data line width (in bytes).	1/2	R
FDTFOI	The format overlay item; Xerox form name.	1	R
FDTTOF	The number of lines used for the top margin.	1/2	R
FDTBOF	The number of lines used for the bottom margin.	1/2	R
FDTDJDE	The DJDE trigger (see Section 3.1 for details).	64	R
FDTRAUX	The RAUX trigger (see Section 3.1 for details).	64	R
FDTRSTCK	The RSTACK trigger (see Section 3.1 for details).	64	R
FDTRPAGE	The RPAGE trigger (see Section 3.1 for details).	64	R
FDTDJDEL	The length of the DJDE trigger.	1/4	R
FDTRAUXL	The length of the RAUX trigger.	1/4	R
FDTRSTKL	The length of the RSTACK trigger.	1/4	R
FDTRPAGL	The length of the RPAGE trigger.	1/4	R

## 8.2.2 Printer Control Block (PCB)

The PCB contains information for a given printer that exists for the life of the SPIN-X execution.

<b>\$EQUF Name</b>	<b>Description</b>	<b>Length (words)</b>	<b>Type</b>
PCBDEVID	The device ID (6 ASCII Characters)	1 1/2	R
PCBDVTPE	The device type (01: Xerox LPS)	1/2	R
PCBPROPT	The SPIN-X processor call options (in master bit notation)	1	R
PCBSITE	An area for the site's programmer use	64	A
PCBHARD	Hardware feature bits	2	R
PCBMSGGP	The alternate console from the SPIN-X CONS keyin or FDP	2	R
PCBCLASS	The Exec message group number of the alternate console	1	R
PCBROUTE	The terminal from the SPIN-X CONS keyin for ER COM\$	2	R

### 8.2.3 File Control Table (FCT)

The FCT contains information for a given device that exists for the span of an initiation through termination of that device.

Tag Name	Description	Length (words)	Type
FCTDATE	The date of print initiation (in DATE\$ ASCII format)	1 1/2	R
FCTPITIM	The time of print initiation (Also in DATE\$ format)	1 1/2	R
FCTSITE	An area for the site's programmer use	64	A
FCTIBBUF	A buffer which is used to pass return data to SPIN-X (Image Builder Buffer in FCT)	128	A
FCTLBL50	The 050 printfile's header label control record	28	R
FCTPCRPT	Pointer to the Printer Control Record (PCR)	1/2	R
FCTLS	The number of lines to skip before printing	1	A
FCTPTNAM	The printfile's part name from the header label	3	R
FCTPTNUM	The printfile's part number from the header label	1/6	R
FCTFILEF	The file sequence number	1	A
FCTINDEV	The input device from printfile header label	2	R
FCTFTDAT	The file creation time/date in TDATE\$ format	1	R
FCTELOCL	The Exec local version	3	R
FCTCUSID	The Customer local site id	2	R
FCTEXEC	The Exec level in ASCII	3	R
FCTSYSID	The 1100/2200 system model, e.g. 2200/500, in ASCII	2	R
FCTPC	The number of pages transferred (includes FCTPR)	1/2	R
FCTLLP	The number of lines transferred	1	R
FCTPR	The number of pages reprinted	1/2	R
FCTPS	The number of pages skipped	1/2	R

**NOTE:** Refer to the *Unisys Series 1100 Data Structures Programming Reference Manual* for a description of the 050 file label control record.

**NOTE:** FCTSITE will be cleared between initializations of the printer; use PCBSITE for values that must be maintained between printer initializations.

### 8.2.4 Printer Control Record (PCR)

The PCR contains information supplied to SPIN-X from the Report Management System (RMS).

\$EQUF Name	Description	Length (words)	Type
PCRCTIMG	If non-zero, indicates an RMS printfile	1	R
PCRRDF1	RMS Report Designator Field 1	38	R
PCRRDF2	RMS Report Designator Field 2	38	R
PCRRDF3	RMS Report Designator Field 3	38	R
PCRRDF4	RMS Report Designator Field 4	38	R
PCRURF1	RMS User Routing Field 1	20	R
PCRURF2	RMS User Routing Field 2	20	R
PCRURF3	RMS User Routing Field 3	20	R
PCRURF4	RMS User Routing Field 4	20	R
PCRURF5	RMS User Routing Field 5	20	R
PCRURF6	RMS User Routing Field 6	20	R
PCRFTF1	Device Id	2	R

## 8.2.5 SMOQUE Entry (SQE)

SQE provides \$EQUFs to reference the fields of the ER SMOQUE\$ entry. These EQUFs are utilized by the SPIN-X subroutines.

\$EQUF Name	Description	Length (words)	Type
SQERUNID	The Runid of the run that symmed the file	2	R
SQEACCNT	The Account of the run that symmed the file	3	R
SQEPRJID	The Project ID of the run that symmed the file	3	R
SQEUSRID	The Userid of the run that symmed the file	3	R
SQEQEID	The Queue-id from the @SYM command	2	R
SQEOUTID	The Output-id of SPIN-X	2	R
SQEUSENM	The @USE name	3	R
SQEQUAL	The qualifier of the symmed file	3	R
SQEFILE	The filename of the symmed file	3	R
SQEFCYCL	The f-cycle of the symmed file	1	R
SQEBANNR	The banner specified on the @SYM command	3	R
SQEFACBT	Facility status bits	1	R
SQEINTID	The ID of the initial SMOQUE entry	1	R
SQEENTID	The current entry ID	1	R
SQEIDLAB	The ID of label block for the SMOQUE entry	1	R
SQEESTPG	The estimated number of pages	1	R
SQEFLGBT	Flag bits	1	R
SQEBRKPT	The breakpoint number	1/2	R
SQEPRINX	The priority index	1/2	R
SQEQTIMS	The number of times the file is queued	1/2	R
SQEGENCY	The GENF\$ recovery cycle	1/2	R

**NOTE:** Refer to the *Unisys Series 1100 Executive System Programmer Reference* for more information about the SMOQUE entry fields .

## 8.2.6 ASCII Edit Packet (AED)

AED provides \$EQUFs to reference the fields in the SYSLIB AEDIT\$ packets utilized by SPIN-X. These EQUFs are utilized by the SPIN-X subroutines.

Tag Name	Description	Length (words)	Type
AEDTS	Test and Set cell	1/6	R
AEDQWM	Quarter word mode	1/6	R
AEDIMGLN	Image length	1/6	A
AEDIMGAD	Image address	1/2	A
AEDCHRIN	Character index	1/6	R
AEDRELWI	Relative word index	1/6	R
AEDMSGCH	AEMSG\$ character index	1/6	R
AEDMSGWD	AEMSG\$ word index	1/2	R
AEDFPS	Number of digits to the left of the decimal point	1/6	A
AEDFPR	The floating point rounding flag	1/6	A
AEDFLAG	A\$EDIT flag	1/8	R
AEDRETAD	Character store return address	1/8	R
AEDUSRAD	User's return address	1/2	R
AEDSAVX1	X1 save area	1/2	R
AEDSAVX2	X2 save area	1	R
AEDSAVX3	X3 save area	1	R
AEDMSG	Stop character	1/4	A
AEDDPC	Mantissa separator double	1/4	A
AEDSPC	Mantissa separator single	1/4	A
AEDILX	Alternate image length	1/4	A

NOTE: Refer to the *Unisys Series 1100 System Service Routines Library (SYSLIB) Reference* for details of AEDIT\$ use. If the SYSLIB ASCII Edit routines are used, then use either the FCTSITE or PCBSITE area for the actual packet and use the above \$EQUFs as displacements within.

## 8.3 Character Translation Table User Exit

---

Each character SPIN-X sends to the device is run through a SPIN-X translation table. For the Xerox LPS, the SPIN-X translation table is based upon ASCII ordering because ASCII is the character set SPIN-X normally utilizes with the LPS. The user exit programmer may modify the translation by specifying a replacement translate table with the externally defined tag "UXASTAS". The translate table SPIN-X references is supplied in \*SRO.

This section provides the information needed to modify the character translation table which SPIN-X uses before sending data to the printer. There are two available translate tables:

1. ASCII to ASCII - used for printers which operate in ASCII mode.
2. ASCII to EBCDIC - used for printers which operate in EBCDIC mode.

For example, both the Xerox 4060/4075 and the IBM 3211 printers operate in EBCDIC mode only; while the Xerox 87xx, 97xx, 4090 and 4050 printers normally operate in ASCII mode.

### 8.3.1 ASCII to ASCII

---

**Translation Table Name: UXASTAS**

**Description**

The ASCII to ASCII character translation table allows the SPIN-X administrator to map Unisys's standard ASCII character set to the printer's ASCII character set. In order to change this table, modify the program element:

*qual*\*SRO.UXASTAS

...to reflect the desired translation table, compile and store the relocatable in *qual*\*SRO, then delete the @MAP EQU statement associated with UXASTAS within the @MAP statements, and re-link SPIN-X as discussed in Section 8.4.

### 8.3.2 ASCII to EBCDIC

---

**Translation Table Name: UXASTEB**

**Description**

The ASCII to EBCDIC character translation table allows the SPIN-X administrator to map Unisys's standard ASCII characters to the printer's EBCDIC character set. In order to change this table, modify the program element:

*qual*\*SRO.UXASTEB

...to reflect the desired translation table, compile and store the relocatable in *qual*\*SRO, then delete the @MAP EQU statement associated with UXASTEB within the @MAP statements, and collect SPIN-X as discussed in Section 8.4.

## 8.4 Collecting SPIN-X with User Exits

This section explains the procedure for linking SPIN-X to include user exits or character translation tables. Follow the steps below:

1. Make sure that the relocatables of all user exits and translate tables you wish to link to the main SPIN-X relocatable are in the \*SRO program file.
2. Examine the element \*SRO.MAP and find the @MAP EQU(s) that correspond to the user exit(s) or translate table you wish to include and then comment out those EQUs. For example, if the banner page user exit is supplied, then "UXRBANPG EQU/0" should be commented out.

SPIN-X uses the value of the user exit tags to determine whether or not a particular user exit is being used (SPIN-X will jump to the tag if it is not equal to 0). Specifically, if an individual user-exit entry point is equal to 0 then SPIN-X assumes that no user exit has been provided; however, if the tag is not equal to 0 then SPIN-X will assume that the exit has been provided and will jump to the user exit. If the EQU statement for a user exit has not been removed, then the value of the tag is still 0, and SPIN-X assumes that the user exit routine has not been supplied. If the EQU has been removed and the user exit provided then SPIN-X jumps to the user exit subroutine.

After the appropriate @MAP EQU(s) have been removed, no further modifications of the \*SRO.MAP element are required if the user exit relocatables reside in the \*SRO file. The user exits' code will be collected into the bank UEXIT\$IBANK; the data will be included in MAIN\$DBANK.

3. Next, issue the following commands (you should @BRKPT PRINT\$ to an alternate file prior to performing the @ADD to capture the results of the collection):

```
@QUAL qual
@ADD,L *SRO.MAP
```

where *qual* is the qualifier used to catalog the SRO file. Inspect the results to verify that the link was successful.

4. If the results of step 3 were successful then the new SPIN-X absolute is ready for testing. Copy the new SPIN-X absolute to the file SYSS\$PRINTER\$\*SPIN-X using the following command:

```
@COPY,A *SRO.SPIN-X,SYSS$PRINTER$*SPIN-X.
```

**NOTE:** Skip this step if a test SPIN-X environment is being utilized. See Chapter 6 for more information on setting up a test SPIN-X.

5. After the new SPIN-X absolute has been satisfactorily tested, you may choose to remove the " ." (comment) from the BANKINFO statement associated with UEXIT\$IBANK and re-collect SPIN-X; this will make UEXIT\$IBANK read-only. The READONLY attribute will provide for more efficient bank management of UEXIT\$IBANK. However, any write operations to a READONLY bank result in a guard mode (IGDM). Therefore, any write operations into UEXIT\$IBANK, such as L\$SNAPS, should be removed prior to re-collecting SPIN-X with UEXIT\$IBANK marked READONLY.



## A.1 Fieldata Log Entries

This section describes the contents of the symbiont log entries that SPIN-X logs, which may be helpful while performing accounting or log analysis. The log entries in section A.1 are in the Fieldata format used in Exec Level 40R1 and earlier releases. Section A.2 contains the ASCII log entries associated with Exec Levels 40R2 and higher.

### A.1.1 Symbiont End of Processing Log Entry (Type 14)

When SPIN-X finishes processing a file, a log entry is made with information about the file. The format of this entry is:

0	entry type	num.words in-entry	system indicator	reserved	reserved	num.-log entries
1	equipment code		file type	total num. of pages printed		
2	symbiont name					
3	number of lines printed					
4	run id					
5 6	account number					
7 8	userid					
9 10	filename					

**Figure A-1** Depiction of the SPIN-X End of Processing Log Table Entries  
(continued on next page)

11	qualifier			
12				
13	number of pages reprinted	number of pages skipped		
14	reserved	reserved	reserved	type of termination
15	project id			
16	symbiont queue			
17	reserved			
.				
.				
24				
25	date and time of log entry			
26	reserved			
27	Exec 8			

**Figure A-1** Depiction of the SPIN-X End of Processing Log Table Entries

**Definitions of Terms for Figure A-1**

<b>entry type</b>	The type of log entry
<b>num words in entry</b>	A count of words used in the entry
<b>system indicator</b>	An indicator used by log editing programs to distinguish changes in log file formats produced by different levels of the Exec
<b>num log entries</b>	Used only in the first entry of a 224-word block
<b>equipment code</b>	The equipment code as defined by system generation or from the DCODE= parameter of the @FDP device statement. For example, the equipment code for a 0770 printer is octal 056.
<b>file type</b>	Indicates the type of file (always output pages)
<b>total num of pages printed</b>	The total number of pages printed for the file
<b>symbiont name</b>	The Fieldata symbiont name used to assign the printer
<b>number of lines printed</b>	The total number of lines printed for the file
<b>run id</b>	The Fieldata run-id of the run which symmed the file
<b>account number</b>	The Fieldata account number of the run which symmed the file
<b>userid</b>	The Fieldata userid of the run which symmed the file

<b>filename</b>	The Fielddata filename of the file which was symmed.
<b>qualifier</b>	The Fielddata qualifier of the file which was symmed
<b>num of pages reprinted</b>	The total number of pages reprinted (if any) for the file
<b>num of pages skipped</b>	The total number of pages skipped (if any) for the file
<b>type of termination</b>	The type of termination. 0 indicates normal termination, 1 indicates termination by an E keyin or error message response, 2 indicates termination by a Q keyin or error message response, 3 indicates termination by a T keyin or error message response, 4 indicates abnormal termination.
<b>project id</b>	Contains the Fielddata project id of the run that @SYM'ed the file.
<b>symbiont queue</b>	The Fielddata name of the symbiont queue that the file was in when it was printed.
<b>date and time of log-entry</b>	The TDATE\$ format date and time of the log entry.
<b>Exec 8</b>	The run-id of the Exec

### A.1.2 Symbiont Start of Processing Log Entry (Type 15)

When SPIN-X begins processing a file, a log entry is made with information about the file. The format of this log entry is:

0	entry type	num. words in-entry	system indicator	reserved	reserved	num.-log entries
1	equipment code		file type	not used		
2	symbiont name					
3	reserved					
4	run-id					
5	account number					
6						
7	userid					
8						
9	filename					
10						
11	qualifier					
12						

**Figure A-2** Depiction of the SPIN-X Start of Processing Log Table Entries  
(continued on following page)

13	not used
14	
15	project id
16	not used
17	sybiont queue
18	
.	
.	reserved
24	
25	date and time of log entry
26	reserved
27	Exec 8

**Figure A-2** Depiction of the Start of Processing Log Table Entries  
(continued from previous page)

**Definitions of Terms for Figure A-2**

<b>entry type</b>	The type of log entry
<b>num words in entry</b>	A count of words used in the entry
<b>system indicator</b>	An indicator used by log editing programs to distinguish changes in log file formats produced by different levels of the Exec
<b>num log entries</b>	Used only in the first entry of a 224-word block
<b>equipment code</b>	The sybiont equipment code from the Exec Unit Status Table or from the DCODE= parameter of the @FDP DEVICE statement. For example, the equipment code for a 0770 printer in octal 056.
<b>file type</b>	Indicates the type of file
<b>sybiont name</b>	The Fieldata sybiont name used to assign the printer
<b>run id</b>	The Fieldata run-id of the run which symmed the file
<b>account number</b>	The Fieldata account number of the run which symmed the file
<b>userid</b>	The Fieldata user-id of the run which symmed the file
<b>filename</b>	The Fieldata filename of the file which was symmed
<b>qualifier</b>	The Fieldata qualifier of the file which was symmed
<b>project id</b>	Contains the Fieldata project-id of the run that symmed the file
<b>sybiont queue</b>	The Fieldata name of the sybiont queue that the file was in when it was printed
<b>date and time of log entry</b>	The TDATE\$ format date and time of the log entry
<b>Exec 8</b>	The run-id of the Exec

### A.1.3 Symbiont Handler Log Entry (Type 35)

The Type 35 log entry is used by output symbiont handlers that run as user programs, such as SPIN-X. There are two formats of this log entry. Log entries of format 2 indicate an internal error in SPIN-X. All format 2 logs should be forwarded to the address given in Section 1.4 of the *SPIN-X Installation Guide*.

The log entry for format 1 is indicated below:

0	entry type	num. words in-entry	system indicator	reserved	reserved	num. log entries
1	entry format	level indicator	relative entry num	total num log entries	identification number	
2	error type	num of sense bytes	reserved		reserved	
3	device name					
4	date and time of error					
5	software fct value	software cnd code	reserved		number of retry attempts	
6	ADH Status	Exec level	hardware sys type	SPIN-X error number		
7	number of completed I/O's					
8 9	failing CCW's					
10 11 12	CSW's					

**Figure A-3** Depiction of the SPIN-X Symbiont Handler Log Entries (continued on next page)

3 . . 17	sense bytes			
18	reserved			
19	run id			
20	console response	reserved	reserved	symbiont equip code
21 22	CCW command bytes			
23	reserved			
24	handler id			
25	date and time of log entry			
26	run id			
27	Exec 8			

**Figure A-3** Depiction of the Symbiont Handler Log Entries  
(Continued from previous page)

#### Definitions of Terms for Table A-3

<b>entry type</b>	The type of log entry.
<b>num words in entry</b>	A count of words used in the entry
<b>system indicator</b>	An indicator used by log editing programs to distinguish changes in log file formats produced by different levels of the Exec.
<b>num log entries</b>	Used only in the first entry of a 224-word block.
<b>entry format</b>	The format of the remainder of the entry: <ol style="list-style-type: none"> <li><b>1</b> indicates that this log entry specifies information relating to an I/O error.</li> <li><b>2</b> indicates that this log entry contains device error log information.</li> </ol>
<b>level indicator</b>	The level of SPIN-X that produced the log entry.
<b>relative entry num</b>	Entry number relative to a chain of related entries.
<b>total num log entries</b>	Total number of log entries that are related to this one.
<b>identification number</b>	A unique identification number supplied by SPIN-X in order to distinguish one set of related log entries from another set.

<b>error type</b>	The type of error represented by this log entry: <b>0</b> implies a normal error condition, such as a unit check or subchannel status <b>1</b> implies a nonattention interrupt while waiting for an attention interrupt.
<b>num-of-sense-bytes</b>	The number of valid sense bytes that are in words 13 and 17 of the log entry.
<b>name</b>	The Fielddata name used to assign the device.
<b>date and time of error</b>	is the TDATE\$ format date and time that SPIN-X initially detected the error.
<b>software fnct value</b>	The software function value that was used in the ADH packet.
<b>software cnd code</b>	The value of the software condition code which was returned within the ADH packet.
<b>number of retry attempts</b>	The number of retry attempts as a result of the error before this log entry was made. The value is a positive value if the last retry was successful and negative if the retry failed.
<b>ADH status</b>	The status returned in the ADH packet.
<b>Exec level</b>	A number indicating the Exec level in use at the time of the log entry. A value of 1 indicates Exec level 39R3 or higher.
<b>hardware sys type</b>	A number indicating the Unisys hardware system type: 1 indicates an 1100/80 or 1100/80A, 2 indicates an 1100/60 or 1100/70, 3 indicates an 1100/90, System 11, or 2200.
<b>SPIN X error number</b>	The SPIN-X error number minus 500 of the operator message that was sent to the operator console in response to a printer status. If no message was sent, this value is zero. See Appendix A of the <i>SPIN-X Operator Guide</i> for a listing of the error messages .
<b>number completed I/O's</b>	Indicates the number of completed I/O's before the last error was logged.
<b>failing CCW's</b>	Contains the failing Channel Command Words (CCW's) words 0 and 1, respectively.
<b>CSW's</b>	Contains the Channel Status Words (CSW's) words 0, 1 and 2, respectively.
<b>sense bytes</b>	Contains sense bytes 0 through 5. The remaining positions are nulls.
<b>run id</b>	The Fielddata run id of the run that symmed the file which is in progress.

<b>console response</b>	A bit map indicating the response to a type and read console message, thus only one bit will be set:  Bit 0 indicates that the message was type only, 1 at least one A was given, 2 an E was given, 3 a G was given, 4 an M was given, 5 a Q was given, 6 an R was given, 7 a T was given.
<b>symbiont equip code</b>	The symbiont equipment code defined by SPIN-X: 0 indicates IBM 3211 mode, 1 indicates Xerox 87xx/97xx/4050/4090 mode, 2 indicates Xerox 4060/4075 CPS mode, 040 indicates a Siemens IBM 3800 emulator.
<b>CCW command bytes</b>	Contains the last eight command bytes (left justified) that were sent in the CCW list.
<b>handler id</b>	The SPIN-X handler ID.
<b>date and time of log entry</b>	The TDATE\$ format date and time that the log entry was made.
<b>run id</b>	The run id of SPIN-X.
<b>Exec 8</b>	The Exec's run id.

---

## A.2 ASCII Log Entries

---

The following is a description of the type-specific section of the log entries. The format of the log entry description section can be found in the *Unisys 1100 Series New System Log Migration Guide* (UP-10966-A). These log entries are pertinent to Exec Levels 40R2 and higher. For earlier versions of the Exec refer to Section A.1.

**NOTE:** Effective with the introduction of the ASCII log file format, the Exec no longer generates the same log entry types for symbiont processing as are produced by stand-alone handlers such as SPIN-X. For example, the Exec generates log entry type 109 when it starts processing a printfile; correspondingly, SPIN-X generates log entry type 10112, subtype 1.

### A.2.1 Symbiont Start of Processing Log Entry (Type 10112 Subtype 1)

When SPIN-X begins processing a file, a log entry is made with information about the file. The format of this entry is:

0	equipment code	file type	reserved
1	device name		
2			
3	run-id		
4			
5	account number		
6			
7			
8	user-id		
9			
10			
11	project-id		
12			
13			
14	filename		
15			
16			
17	qualifier		
18			
19			
20	queue name		
21			
22	concatenated filename		
23			
24			
25	concatenated qualifier		
26			
27			
28	physical page number		

**Figure A-4** Depiction of the SPIN-X Start of Processing Log Table Entries

**Definitions of Terms for Figure A-4**

<b>equipment code</b>	The symbiont equipment code from the Exec Unit Status Table or from the DCODE= parameter of the @ FDP DEVICE statement . For example, the equipment code for a 0770 printer is octal 056.
<b>file type</b>	Indicates the type of file
<b>device name</b>	The name of the print device
<b>run id</b>	The run id that queued the print file
<b>account number</b>	The account number of the run that queued the print file.
<b>user id</b>	The user id of the run that queued the print
<b>project id</b>	The project id of the run that queued the print
<b>filename</b>	The name of the file that was queued
<b>qualifier</b>	The qualifier of the file that was queued
<b>queue name</b>	The name of the queue the file was printed from
<b>concatenated filename</b>	Unused by SPIN-X, always spaces
<b>concatenated qualifier</b>	Unused by SPIN-X, always spaces
<b>physical page number</b>	Unused by SPIN-X, always zeroes

## A.2.2 Symbiont End of Processing Log Entry (Type 10112 Subtype 2)

When SPIN-X ends processing of a file, a log entry is made with information about the file. The format of this entry is:

0	equipment code	file type	reserved
1	total page count		
2	device name		
3			
4	line count		
5	run id		
6			
7	account number		
8			
9			
10	user id		
11			
12			
13	project id		
14			
15			
16	filename		
17			
18			

**Figure A-5** Depiction of the SPIN-X End of Processing Log Table Entries  
(continued on next page)

19 20 21	qualifier	
22 23	queue name	
24 25 26	concatenated filename	
27 28 29	concatenated qualifier	
30	physical page number	
31	reprint count	
32	skip count	
33	reserved	termination type

**Figure A-5** Depiction of the SPIN-X End of Processing Log Table Entries  
(continued from previous page)

**Definitions of Terms for Figure A-5**

<b>equipment code</b>	The symbiont equipment code from the Exec Unit Status Table or from the DCODE= parameter of the @ FDP DEVICE statement . For example, the equipment code for a 0770 printer is octal 056.
<b>file type</b>	Indicates the type of file
<b>total page count</b>	The total number of pages printed for the file
<b>device name</b>	The name of the print device
<b>line count</b>	The number of lines printed for the file
<b>run id</b>	The run id that queued the print file
<b>account number</b>	The account number of the run that queued the print file.
<b>user id</b>	The userid of the run that queued the print
<b>project id</b>	The project id of the run that queued the print
<b>filename</b>	The name of the file that was queued
<b>qualifier</b>	The qualifier of the file that was queued
<b>queue name</b>	The name of the queue the file was printed from
<b>concatenated filename</b>	Unused by SPIN-X, always spaces
<b>concatenated qualifier</b>	Unused by SPIN-X, always spaces
<b>physical page number</b>	Unused by SPIN-X, always zero
<b>reprint count</b>	The total number of pages reprinted (if any) for the file
<b>skip count</b>	The total number of pages skipped (if any) for the file
<b>termination type</b>	The type of termination  0 - normal termination 1 - caused by E keyin or error message response 2 - caused by Q keyin or error message response 3 - caused by T keyin or error message response 4 - abnormal termination

### A.2.3 Symbiont Device Error Log Entry (Type 11502)

When a symbiont device handler errs a log entry is made with information about the error. The format of this log entry is:

0	handler level	number of entries	reserved
1	device name		
2			
3	equipment code	error log byte count	
4	error log bytes - 1		
5			
.			
.			
.			
23			
24	error log bytes - 2		
25			
.			
.			
47	error log bytes - 3		
48			
49			
67			

**Figure A-6** Depiction of the SPIN-X Symbiont Device Error Log Table Entries

**Definitions of Terms for Table A-6**

<b>handler level</b>	The level of the printer handler that created this log entry. This will always be binary 1 for SPIN-X.
<b>number of entries</b>	The number of log entries
<b>device name</b>	The name of the print device
<b>equipment code</b>	The symbiont equipment code from the Exec Unit Status Table or from the DCODE= parameter of the @ FDP DEVICE statement For example, the equipment code for a 0770 printer is octal 056
<b>error log byte count</b>	The number of error bytes in the log entry
<b>error log bytes 1</b>	The number of bytes in section 1
<b>error log bytes 2</b>	The number of bytes in section 2
<b>error log bytes 3</b>	The number of bytes in section 3

## A.2.4 Symbiont I/O Fault Error Log Entry (Type 11503)

When a symbiont I/O fault occurs, a log entry is made with information about the error. The format of this log entry is:

0	handler level	error type	sense byte count	reserved
1	device name			
2	time of error			
3				
4	ADH function	software condition code	reserved	retry count
5	ADH status	reserved	machine type	error number
6	reference since last error			
7	failing CCW (word 0)			
8	failing CCW (word 1)			
9	channel status (word 0)			
10	channel status (word 1)			
11	channel status (word 2)			
12	sense bytes (0 to 3)			
13	sense bytes (4 to 7)			
14	sense bytes (8 to 11)			
15	sense bytes (12 to 15)			
16	sense bytes (16 to 19)			
17	file run-id			
18				
19	reserved	reserved	reserved	equipment code
20	command byte			
21				

**Figure A-7** Depiction of the SPIN-X Symbiont I/O Fault Error Log Table Entries

### Definitions of Terms for Table A-7

<b>handler level</b>	The level of the printer handler that created this log entry. This will always be binary 1 for SPIN-X.
<b>error type</b>	The type of error represented by this log entry: <b>0</b> -implies a normal error condition, such as a unit check or subchannel status. <b>1</b> -implies a nonattention interrupt while waiting for an attention interrupt.
<b>sense byte count</b>	The number of valid sense bytes stored in the log entry
<b>device name</b>	The name of the print device on which the error occurred
<b>time of error</b>	The TDATE\$ format date and time that SPIN-X initially detected the error
<b>ADH function</b>	The software function value that was used in the ADH packet
<b>software condition code</b>	The value of the software condition code which was returned within the ADH packet
<b>retry count</b>	The number of retry attempts as a result of the error before this log entry was made. The value is positive if the retries were successful and negative if the retries failed.
<b>ADH status</b>	The status returned in the ADH packet
<b>machine type</b>	A number indicating the Unisys hardware system type:  1-indicates an 1100/80 2-indicates an 1100/60, /70, /90, 2200
<b>reference since error</b>	The number of successful I/O's that have occurred since the last error or since the device was last initialized.
<b>failing CCW's</b>	Contains the failing Channel Command Words (CCW's) 0 and 1, respectively
<b>CSW's</b>	Contains the Channel Status Words (CSW's) 0, 1 and 2 respectively
<b>sense bytes</b>	Contains sense bytes 0 through 19, the remaining positions are null
<b>file run id</b>	The run id of the run that symmed the file which is in progress
<b>command byte</b>	Byte 1 is the command byte issued just prior to the command that failed and bytes 2 to 8 are the preceding seven command bytes with byte 8 being the one issued first. If less than 8 bytes preceded the failure on this I/O, then the remaining bytes are set to 0777



# B ECL to Create a SPIN-X Central Absolute (MAP)

```
@PREP,F SRO.
@MAP,CEFINX ,SRO.SPIN-X
TYPE      BLOCKSIZE64,EXTDIAG,QUARTER,REALTIME
EQU       UXASTAS/0
EQU       UXASTEB/0
EQU       UXTAPETABLE/0
EQU       UXRBANPG/0
EQU       UXRBEGPC/0
EQU       UXRENDPC/0
EQU       UXRFORSL/0
EQU       UXRGENCL/0
EQU       UXRLOG14/0
EQU       UXRTRAPG/0
EQU       UXRIMAGE/0
EQU       BFDASC$/FDASC$
EQU       BSDFIC$/SDFIC$
EQU       BSDFIO$/SDFIO$
EQU       BSDFI$/SDFI$
IBANK     INIT$IBANK,01000
          BANKINFO  INITIAL(MAINI),DYNAMIC,READONLY
          IN        SRO.SPIN-X($15)
          IN()      FDASC$
          IN()      TABLE$
          IN()      INFOR$
          IN()      AEDIT$
          IN()      AEDIT$T
          IN()      SFDT$
          IN()      SFDTBL$
          IN()      AEDIT$SFDT
IBANK     MAIN$IBANK,01000
          BANKINFO  BDR(0),READONLY
          $1,3,5,7,23,25,27,31,33,35,37,39,43,45,49
          IN        SRO.SPIN-X
          IN()      FDASC$
          IN()      TABLE$
          IN()      AEDIT$
          IN()      AEDIT$T
          IN()      SFDT$
          IN()      SFDTBL$
          IN()      AEDIT$SFDT
          IN()      ID$
          IN()      BSP$
          IN()      SDFI
IBANK     CONT$IBANK,01000          . CONTINGENCY ROUTINES IBANK
          BANKINFO  BDR(0),DYNAMIC
          IN        SRO.SPIN-X($9)
          IN()      FDASC$
          IN()      TABLE$
          IN()      AEDIT$
          IN()      AEDIT$T
          IN()      SFDT$
          IN()      SFDTBL$
          IN()      AEDIT$SFDT
```

```

IBANK      SEPS$IBANK,01000
           BANKINFO  BDR(0),READONLY
           IN        SRO.SPIN-X($21)
           IN()      FDASC$
           IN()      TABLE$
           IN()      AEDIT$
           IN()      AEDIT$T
           IN()      SFDT$
           IN()      SFDTBL$
           IN()      AEDIT$SFDT
IBANK      PIM$IBANK,01000
           BANKINFO  BDR(0),READONLY
           IN        SRO.SPIN-X($41)
           IN()      FDASC$
           IN()      TABLE$
           IN()      AEDIT$
           IN()      AEDIT$T
           IN()      SFDT$
           IN()      SFDTBL$
           IN()      AEDIT$SFDT
IBANK      UEXIT$IBANK,01000 . user exits are collected into this bank
           BANKINFO  BDR(0) . ,READONLY
           IN()      FDASC$
           IN()      TABLE$
           IN()      AEDIT$
           IN()      AEDIT$T
           IN()      SFDT$
           IN()      SFDTBL$
           IN()      AEDIT$SFDT
DBANK      MAIN$DBANK,(INIT$IBANK,MAIN$IBANK,PIM$IBANK,UEXIT$IBANK,SEPS$IBANK)
           BANKINFO  INITIAL(MAIND),CONTROL
           $EVEN
           IN        SRO.SPIN-X
LIB        SRO.(UEXIT$IBANK/$ODD,MAIN$DBANK/$EVEN)
END

```

---

# C

# Listings of Xerox JSL and FSL Files

The following pages are listings of the Xerox source code for the Job Source Language and Form Source Language files needed on the Xerox by SPIN-X.

---

## C.1 ON1100.JSL

---

```
ON1100:JDL;

        ACODE:CODE  DEFAULT=ASCII,
          ASSIGN=(X'7B',X'C0'),
          ASSIGN=(X'7C',X'FA'),
          ASSIGN=(X'7D',X'D0');

/* The above CODE reassignments are for redefining the left      */
/* curly brace, vertical bar and right curly brace to the       */
/* positions where Xerox fonts have them defined.  If these    */
/* changes are not required change the ACODE below to ASCII.   */
/*                                                                */

VOLUME HOST=IBMONL,
        CODE=ACODE,
        OPTIMIZE=(NCC,NDC,NPR);

RECORD LENGTH=150;
ACCT USER=NONE;

IDEN PREFIX=A'%%DJDE',SKIP=7,OFFSET=0,OPRINFO=NO;

T1: TABLE CONSTANT=(X'01020102');
C1: CRITERIA CONSTANT=(0,4,EQ,T1);
T2: TABLE CONSTANT=(X'02010201');
C2: CRITERIA CONSTANT=(0,4,EQ,T2);
T3: TABLE CONSTANT=(X'02010102');
C3: CRITERIA CONSTANT=(0,4,EQ,T3);

RAUX TEST=C1;
RPAGE TEST=C2,WHEN=NOW;
RSTACK TEST=C3,DELIMITER=YES,HRPTNA=(4,16);

LINE DATA=(0,132),PCCTYPE=IBM3211,FCB=PROCESS,UCSB=IGNORE;

DFLT:JDE;
      OUTPUT DUPLEX=YES,FORMAT=FMT1X;

LS132:JDE;
      OUTPUT DUPLEX=NO,FORMAT=FMT1X;

LD132:JDE;
      OUTPUT DUPLEX=YES,FORMAT=FMT1X;

LD150:JDE;
      OUTPUT DUPLEX=YES,FORMAT=FMT2X;
      LINE DATA=(0,150);

PS80:JDE;
      OUTPUT DUPLEX=NO,FORMAT=FMT8X;
      LINE DATA=(0,80);

PD80:JDE;
      OUTPUT DUPLEX=YES,FORMAT=FMT8X;
      LINE DATA=(0,80);
```

```
PS95:JDE;
  OUTPUT DUPLEX=NO,FORMAT=FMT7X;
  LINE DATA=(0,95);

PD95:JDE;
  OUTPUT DUPLEX=YES,FORMAT=FMT7X;
  LINE DATA=(0,95);

PD106:JDE;
  OUTPUT DUPLEX=YES,FORMAT=FMT6X;
  LINE DATA=(0,106);

PD132:JDE;
  OUTPUT DUPLEX=YES,FORMAT=FMT10X;

PD132T:JDE;
  OUTPUT DUPLEX=YES,FORMAT=TWOUP;

LABELS:JDE;
  OUTPUT DUPLEX=NO,FORMAT=LABEL6;
  LINE DATA=(0,35);
END;
```

## C.2 P1100.JSL

```

/*                                     */
/* Print Descriptor Entries for use with SPIN-X */
/*                                     */
SPXBNR:PDE FONTS=(P08ITA,5),BEGIN=(3.11 IN,4.4 IN),PMODE=PORTRAIT;
/* The above PDE is used for the banner page for sites not using */
/* the print tape support feature */
/*                                     */
SP2BNR:PDE FONTS=(P08ITA,6),BEGIN=(2.65 IN,4.2 IN),PMODE=PORTRAIT;
/* The above PDE is used for the banner page for sites using the */
/* print tape support feature. */
/*                                     */
TWOUP:PDE BEGIN=(.2,.5),BEGIN=(5.5,.5),PMODE=PORTRAIT,
        FONTS=(P1012B,12.5);
/* Used with JDE PD132T to print 2 logical pages per side */
/*                                     */
LABEL6:PDE BEGIN=(.05,.06),BEGIN=(.05,2.9),BEGIN=(.05,5.75),
        BEGIN=(1.05,.06),BEGIN=(1.05,2.9),BEGIN=(1.05,5.75),
        BEGIN=(2.05,.06),BEGIN=(2.05,2.9),BEGIN=(2.05,5.75),
        BEGIN=(3.05,.06),BEGIN=(3.05,2.9),BEGIN=(3.05,5.75),
        BEGIN=(4.05,.06),BEGIN=(4.05,2.9),BEGIN=(4.05,5.75),
        BEGIN=(5.05,.06),BEGIN=(5.05,2.9),BEGIN=(5.05,5.75),
        BEGIN=(6.05,.06),BEGIN=(6.05,2.9),BEGIN=(6.05,5.75),
        BEGIN=(7.05,.06),BEGIN=(7.05,2.9),BEGIN=(7.05,5.75),
        BEGIN=(8.05,.06),BEGIN=(8.05,2.9),BEGIN=(8.05,5.75),
        BEGIN=(9.05,.06),BEGIN=(9.05,2.9),BEGIN=(9.05,5.75),
        BEGIN=(10.05,.06),BEGIN=(10.05,2.9),BEGIN=(10.05,5.75);
        PMODE=PORTRAIT,FONTS=P08TBC;
LABEL8:PDE BEGIN=(.05,.06),BEGIN=(.05,2.9),BEGIN=(.05,5.75),
        BEGIN=(1.05,.06),BEGIN=(1.05,2.9),BEGIN=(1.05,5.75),
        BEGIN=(2.05,.06),BEGIN=(2.05,2.9),BEGIN=(2.05,5.75),
        BEGIN=(3.05,.06),BEGIN=(3.05,2.9),BEGIN=(3.05,5.75),
        BEGIN=(4.05,.06),BEGIN=(4.05,2.9),BEGIN=(4.05,5.75),
        BEGIN=(5.05,.06),BEGIN=(5.05,2.9),BEGIN=(5.05,5.75),
        BEGIN=(6.05,.06),BEGIN=(6.05,2.9),BEGIN=(6.05,5.75),
        BEGIN=(7.05,.06),BEGIN=(7.05,2.9),BEGIN=(7.05,5.75),
        BEGIN=(8.05,.06),BEGIN=(8.05,2.9),BEGIN=(8.05,5.75),
        BEGIN=(9.05,.06),BEGIN=(9.05,2.9),BEGIN=(9.05,5.75),
        BEGIN=(10.05,.06),BEGIN=(10.05,2.9),BEGIN=(10.05,5.75);
        PMODE=PORTRAIT,FONTS=P06BOB;
/* Used with LABELS JDE to print 33 to a sheet gummed labels */
/*                                     */
/* The following PDES are the standard Xerox distributed PDEs */
FMT1X:PDE PMODE=LANDSCAPE,BEGIN=(.18 IN,.66 IN),FONTS=(L0112B);
FMT2X:PDE PMODE=LANDSCAPE,BEGIN=(.18 IN,.50 IN),FONTS=(L0212A);
FMT3X:PDE PMODE=LANDSCAPE,BEGIN=(.14 IN,.66 IN),FONTS=(L0312A);
FMT4X:PDE PMODE=LANDSCAPE,BEGIN=(.18 IN,.66 IN),FONTS=(L0412A);
FMT5X:PDE PMODE=LANDSCAPE,BEGIN=(.18 IN,.50 IN),FONTS=(L0512A);
FMT6X:PDE PMODE=PORTRAIT,BEGIN=(.57 IN,.58 IN),FONTS=(P0612C);
FMT7X:PDE PMODE=PORTRAIT,BEGIN=(.50 IN,.50 IN),FONTS=(P07TYA);
FMT7X8:PDE PMODE=PORTRAIT,BEGIN=(.50 IN,.50 IN),FONTS=(P07TYA,8.0);
FMT8X:PDE PMODE=PORTRAIT,BEGIN=(.50 IN,.50 IN),FONTS=(P08TAC);
FMT9X:PDE PMODE=LANDSCAPE,BEGIN=(.25 IN,.25 IN),FONTS=(L0912A);
FMT10X:PDE PMODE=PORTRAIT,BEGIN=(.22 IN,.51 IN),FONTS=(P1012B);
FMT11X:PDE PMODE=PORTRAIT,BEGIN=(.22 IN,.50 IN),FONTS=(P1112A);
FMT12X:PDE PMODE=LANDSCAPE,BEGIN=(.18 IN,.66 IN),FONTS=(L0112B);
FMT13X:PDE PMODE=PORTRAIT,BEGIN=(.22 IN,.51 IN),FONTS=(P0612C);

```

---

### C.3 SP2BNR.JSL

---

```
SP2BNR:PDE FONTS=(P08ITA,6),BEGIN=(2.65 IN,4.2 IN),PMODE=PORTRAIT;
/* The above PDE is used for the banner page for sites using the */
/* print tape support feature. */
```

---

### C.4 GRABAR.FSL

---

```
FORM GRABAR;                                000010
COMMENT CREATED 7-20-83 AT GSU COMPUTER CENTER; 000020
LANDSCAPE;                                  000030
GRID UNIT IS FMT1;                          000040
AT 0,-2 DRAW 11 BOXES 136 WIDE BY 3 HIGH USING SHA MED 000050
    AND REPEAT VERTICALLY EVERY 6;          000060
END;                                          000070
```

---

### C.5 BANNER.FSL

---

```
FORM BANNER;                                000010
COMMENT BANNER FORM FOR SITES NOT USING THE PRINT TAPE OPTION; 000020
PORTRAIT;                                    000030
GRID IS FMT6;                                000040
FONTS PR224B,PR118B;                        000050
                                                000060
AT 1 DRAW LINES FROM 1 TO 100 USING SOLID 0 000070
    AND REP AT 6,10;                          000080
AT 1 DRAW 2 VER LINES FROM 1 TO 10 USING SOLID 0 000090
    AND REP AT 100;                            000100
TEXT USI FON 1 IN BOX 1,1 'SPIN-X';          000102
TEXT USI FONT 1 IN BOX 6,1 'XEROX EPS';     000104
AT 19,5 DRAW BOX 47 WIDE BY 62 HIGH USI SOL 0; 000110
TEXT SPACED 1.5 LPI ALI RIGHT USI FON 2 IN BOX 19,5 000115
    'USERID -'                                  000120
    'ACCOUNT NUMBER -'                          000125
    'RUNID -'                                    000130
    'FILENAME -'                                 000135
    'DATE -'                                    000150
    'TIME -'                                    000155
    'PRINTER -'                                 000160
    'FORM -'                                    000165
    'ROUTE TO -';                               000170
AT 0,2 DRAW BOX 96 WIDE BY 79 HIGH USI SOL 1; 000180
AT -1 DRAW 2 LINES FROM 0 TO 100 USI SOL 2 AND REP AT 80; 000190
AT 0 DRAW 2 VER LINES FROM -1 TO 80 USI SOL 2 AND REP AT 100; 000200
AT -1 DRAW 2 LINES FROM 0 TO 100 USING SOL 2 AND REP AT 80; 000240
AT 0 DRAW 2 VER LINES FROM -1 TO 80 USING SOL 2 AND REP AT 100; 000250
END;                                          000260
```

---

## C.6 BANNR2.FSL

---

```

FORM      BANNR2;                                000010
COMMENT   BANNER FORM FOR SITES WITH THE PRINT TAPE OPTION; 000020
PORTRAIT;                                000030
GRID IS FMT6;                              000040
FONTS     PR224B,PR118B;                    000050
                                                000066
AT 1 DRAW LINES FROM 1 TO 100 USING SOLID 0 000070
    AND REP AT 6,10;                        000080
AT 1 DRAW 2 VER LINES FROM 1 TO 10 USING SOLID 0 000090
    AND REP AT 100;                          000100
TEXT USI FON 1 IN BOX 1,1 'SPIN-X';         000102
TEXT USI FONT 1 IN BOX 6,1 'XEROX EPS';     000104
AT 19,5 DRAW BOX 47 WIDE BY 62 HIGH USI SOL 0; 000110
TEXT SPACED 1.5 LPI ALI RIGHT USI FON 2 IN BOX 19,5 000115
    'USERID -'                                000120
    'ACCOUNT NUMBER -'                       000125
    'RUNID -'                                 000130
    'FILENAME -'                              000135
    'PARTNAME -'                              000140
    'PARTNUMBER -'                            000145
    'DATE -'                                  000150
    'TIME -'                                  000155
    'PRINTER -'                               000160
    'FORM -'                                  000165
    'ROUTE TO -';                             000170
AT 0,2 DRAW BOX 96 WIDE BY 79 HIGH USI SOL 1; 000180
AT -1 DRAW 2 LINES FROM 0 TO 100 USI SOL 2 AND REP AT 80; 000190
AT 0 DRAW 2 VER LINES FROM -1 TO 80 USI SOL 2 AND REP AT 100; 000200
AT -1 DRAW 2 LINES FROM 0 TO 100 USING SOL 2 AND REP AT 80; 000240
AT 0 DRAW 2 VER LINES FROM -1 TO 80 USING SOL 2 AND REP AT 100; 000250
END;                                          000260

```

## D Listing of the Distributed FDP Input File

```
Format Name=pr,MODE=L,DUPLEX=NO,PAGES=1,LINES=66,JDL=ON1100,JDE=LS132 ;
    LINES8=88,PDE6=FMT1X,PDE8=FMT3X,OVERLAY=NONE,WIDTH=132,DYNAMIC=YES, ;
    ,TOP=6,BOT=3,SHIFT=N,DJDE=A'%%DJDE',0,7,TYPE=LASER;
    RAUX=X'01020102',0,RPAGE=O'002001002001',0,RSTACK = X '02010102',0
    . Default print queue - undefined parameters in the following formats
    . will take their values from the above entry
    .
FORMAT NAME=LS132,MODE=L,DUPLEX=NO,PAGES=1,LINES=66,LINES8=88,JDE=LS132 ;
    PDE6=FMT1X,PDE8=FMT3X,OVERLAY=NONE,WIDTH=132,TOP=6,BOT=3
    .
FORMAT NAME=LS132G,MODE=L,DUPLEX=NO,PAGES=1,JDE=LS132,OVERLAY=GRABAR, ;
    PDE6=FMT1X,PDE8=FMT3X,WIDTH=132,TOP=6,BOT=3
    . This is the same as LS132 but uses a 3-line shaded gray bar overlay
    .
FORMAT NAME=LD132,MODE=L,DUPLEX=YES,PAGES=2,JDE=LD132, ;
    PDE6=FMT1X,PDE8=FMT3X,WIDTH=132,TOP=6,BOT=3
    .
FORMAT NAME=LD132G,MODE=L,DUPLEX=YES,PAGES=2,JDE=LD132,OVERLAY=GRABAR, ;
    PDE6=FMT1X,PDE8=FMT3X,WIDTH=132,TOP=6,BOT=3
    . This is the same as LD132 but uses a 3-line shaded gray bar overlay
    .
FORMAT NAME=LD150,MODE=L,DUPLEX=YES,PAGES=2,WIDTH=150,PDE6=FMT2X, ;
    PDE8=FMT4X,JDE=LD150
    .
FORMAT NAME=PS80,MODE=P,DUPLEX=NO,PAGES=1,WIDTH=80,PDE6=FMT8X,PDE8=FMT6X, ;
    JDE=PS80,LINES=63,LINES8=80,TOP=3,BOT=3
    . This is the standard 'typewriter style' format in simplex
    .
FORMAT NAME=PD80,MODE=PORTRAIT,DUPLEX=YES,WIDTH=80,LINES=63, ;
    JDE=PD80,TOP=3,BOT=3,JDE=PD80,PDE6=FMT8X,PDE8=FMT6X
    . This is the standard 'typewriter style' format in duplex
    .
FORMAT NAME=PS95,MODE=P,DUPLEX=NO,PAGES=1,WIDTH=95,LINES=63, ;
    JDE=PS95,TOP=3,BOT=3,PDE6=FMT7,PDE8=FMT7X8
    . Serifed Courier Font at 12 pitch - simplex
    .
FORMAT NAME=PD95,MODE=P,DUPLEX=YES,PAGES=2,WIDTH=95,LINES=63, ;
    JDE=PD95,TOP=3,BOT=3,PDE6=FMT7,PDE8=FMT7X8
    . Serifed Courier Font at 12 pitch - duplex
    .
FORMAT NAME=PD106,MODE=P,DUPLEX=YES,PAGES=2,WIDTH=106,LINES=80, ;
    JDE=PD106,TOP=6,BOT=3,PDE6=FMT6X,PDE8=FMT6X
    . Xerox font at 13.6 Characters per inch, 8.1 lines per inch
    .
FORMAT NAME=PD132,MODE=P,DUPLEX=YES,PAGES=2,WIDTH=132,LINES=132, ;
    LINES8=132,PDE6=FMT10X,PDE8=FMT10X,JDE=PD132,TOP=6,BOT=3
    . 132 lines by 132 columns, 1 logical page per side
    .
FORMAT NAME=PD132T,MODE=P,DUPLEX=YES,PAGES=4,WIDTH=132,LINES=66, ;
    LINES8=66,PDE6=TWOUP,PDE8=TWOUP,JDE=PD132T,TOP=6,BOT=3
    . 66 lines by 132 columns, 2 logical pages per side (quadruplex)
    .
FORMAT NAME=LABELS,MODE=P,DUPLEX=NO,PAGES=33,LINES=5,JDE=LABELS ;
    LINES8=7,PDE6=LABEL6,PDE8=LABEL8,OVERLAY=NONE,WIDTH=35, ;
    ,TOP=0,BOT=0 . LABELS PRINTS FOR LX
    . designed for 33 to a page - 3 by 11 - gummed label sheets
    .
    .
DEVICE NAME=LX,TYPE=9700
```

---

# E

# Appendix E: FDP File Linkage

Beginning with SPIN-X version 4R3 all file assignments for the Format Definition Program have been internalized within the program using calls to ERACSF\$. Below is a listing of the filenames used and their internal linknames.

Linkname	External Filename	Description
*FDP-INPUT	FDP-FORMS-IN	Input file of FORMAT and DEVICE commands
*FORMS-DEF	FDP-FORMSDEF	Output file (Non-database version)
*SRI\$DATABASE	SRI\$DATABASE	Index part of database
*SRI\$FLATFILE	SRI\$FLATFILE	Data part of database
*FDP-REPORT	REPORT-FILE	Listing of forms, devices, or errors
*FDP-TMP-FRMS	FDP-TMP-FRMS	Temporary Scratch file
*FDP-TMP-DEVS	FDP-TMP-DEVS	Temporary Scratch file
*FDP-TMP-JOBS	FDP-TMP-JOBS	Temporary Scratch file
*FDP-TMP-JCLS	FDP-TMP-JCLS	Temporary Scratch file
*FDP-TMP-SVRS	FDP-TMP-SVRS	Temporary Scratch file
*FDP-TMP-USER	FDP-TMP-USER	Temporary Scratch file
*FDP-TMP-QUES	FDP-TMP-QUES	Temporary Scratch file
*FDP-SRT-FRMS	FDP-SRT-FRMS	Temporary Scratch file
*FDP-SRT-DEVS	FDP-SRT-DEVS	Temporary Scratch file
*FDP-SRT-QUES	FDP-SRT-QUES	Temporary Scratch file
R\$SCORE	R\$SCORE	Temporary SORT file

---

## **F ECL to Copy FDP Files to Production (FDP/TOPRODUCTION)**

The following ECL should be used each time the FDP file is updated. This addstream should be run after a successful FDP execution to copy the new database files to production. After this addstream is performed, the changes will take place the next time SPIN-X is started.

```
#####@FREE SYS$PRINTER$*SPIN-X.  
@FREE SYS$PRINTER$*SRI$DATABASE.  
@FREE SYS$PRINTER$*SRI$FLATFILE.  
. @COPY,I *FORMS-DEF.,SYS$PRINTER$*SPIN-X.FORMS-DEF  
@CAT SYS$PRINTER$*SRI$DATABASE(+1).,///1024  
@COPY *SRI$DATABASE.,SYS$PRINTER$*SRI$DATABASE.  
@CHG,V SYS$PRINTER$*SRI$DATABASE.  
@CAT SYS$PRINTER$*SRI$FLATFILE(+1).,///1024  
@COPY *SRI$FLATFILE.,SYS$PRINTER$*SRI$FLATFILE.  
@CHG,V SYS$PRINTER$*SRI$FLATFILE.  
@FREE SYS$PRINTER$*SPIN-X.  
@FREE SYS$PRINTER$*SRI$DATABASE.  
@FREE SYS$PRINTER$*SRI$FLATFILE.
```

There are several utility programs provided with the SPIN-X package which are not required for day to day operation but can be useful in performing certain SPIN-X related functions. These utilities are unsupported but their source code is supplied. These programs can be found in the SPIN-X utility file (\*UTILITY) on the release tape for the Unisys 1100/2200.

---

**G.1 CLEANQ**

---

Under certain circumstances, such as a SPIN-X internal error during printing, a print queue entry may become "hung". It may be listed at the host console as being in the process of printing although nothing will actually be going down the channel to the printer. Since the system thinks the file is currently being printed it will not allow the file to be deleted from the queue. @CLEANQ allows the operator to free this file by clearing the in-progress bit that is set on its queue entry. This should be run under an account that has SMOQUE\$ privileges. @CLEANQ has one parameter, which is the queue that should be cleared. This only clears the in-progress bit and does not remove the queue entry from the queue.

**NOTE:** When @CLEANQ is executed, SPIN-X should not be accessing the target queue.

---

**G.2 FILE COPY**

---

This COBOL utility program is used for downloading files from the Unisys host disk to the Xerox printer's disk using the "FILE" DJDE. Complete instructions on using this utility are provided in Section 5.2 of this guide.

---

**G.3 TRANSLATE/ASCII-EBCDIC**

---

When downloading files to the Xerox disk using the Xerox Hostcopy utility it is necessary to have any text files, such as FSL and JSL files, converted to EBCDIC since this is the only coding scheme that Hostcopy recognizes. An assembly language utility has been provided which will perform this function. The usage of this utility is explained in Section 5.1.1 of this guide.

---

**G.4 READ/XEROX-FILE**

---

This short FORTRAN utility is used for reading a Xerox formatted tape onto the Unisys disk and reblocking it into a format in which it can then be downloaded to the Xerox disk using the Hostcopy utility. Usage of this utility is discussed more fully in Section 5.1.2 of this guide.

---

## H Sample "L" Processor Call Option Listing

The following listing is a sample of the data written to the PRINT\$ file if the "L" option is specified on the SPIN-X processor call. This information provides an alternative to searching the system logfile to obtain a listing of the files printed. Also, this data can be input into a post-processor to accumulate statistics on the files printed.

DATE	TIME	USERID	ACCOUNT	RUNID	FILENAME	FORMAT	DEVICE	PAGES	BYTES /PAGE
010892	162501	INSTALLATION	INSTALLATION	ROLBAK	SYS\$*PR@000ROLBAK(1)	PR	PRT2	4	490
010892	162506	4EID	S999945-4231	ASLATE	SYS\$*PR@000ASLATE(1)	PR	PRT2	2	234
010892	162510	INSTALLATION	INSTALLATION	ROLBAL	SYS\$*PR@000ROLBAL(1)	PR	PRT2	4	462
010892	162514	CEID	S999945-4231	ASLATE	CED*ASLATE-PRINT(50)	PR	PRT2	9	1507
010892	162519	CEID	S999945-4231	ASLATE	CED*SLAREM(1)	PR	PRT2	5	1401
010892	162523	CEID	S999945-4231	ASLATE	SYS\$*PR@001ASLATE(1)	PR	PRT2	3	348

---

# I SPIN-X ANSI Output Tape Feature

This appendix describes the Spin-X ANSI Output Tape Feature. This feature will allow sites to generate ANSI formatted tapes containing print files to be processed by a Xerox LPS in offline mode. The print files may also be loaded on computers that can handle ANSI compatible tapes such as an IBM host.

---

## I.1 General Description

---

The Spin-X Tape Feature provides Unisys hosts the ability to create ANSI formatted tapes which contain print files. These print files may be processed by a Xerox EPS in offline mode or may be loaded on an IBM or compatible host. The print files will be generated with ANSI carriage control. The ANSI formatted tapes may be written in 800 bpi, 1600 bpi 6250 bpi or square tape.

---

## I.2 Installation

---

The Spin-X ANSI Tape Feature requires that the tape device be defined to EXEC and to Spin-X. The following sections will describe required configuration parameters. Note: This installation description presumes that the customer is familiar with the installation procedures listed in the Spin-X Installation Guide. Information in this document describes additional installation procedures required for the Spin-X ANSI Tape Feature.

### I.2.1 EXEC Configuration

---

The tape device should be configured as an 0770 printer within the EXEC sysgen. As with printer devices, OUTPUT SGSs may be defined and associated with the tape device. The EXEC configuration requirements will be the same as defined for dummy devices in Section 5.1.2 of the Spin-X Installation Guide.

### I.2.2 Spin-X Configuration

---

The Spin-X ANSI Tape device must be defined to Spin-X in the configuration file (\*FDP-INPUT). The FDP DEVICE statement has been modified to support the Spin-X ANSI Tape Feature. No changes have been made to the FORMAT statement.

The Exec device mnemonic used for the tape device must be defined to Spin-X with the FDP DEVICE command. The syntax for the SPIN-X tape DEVICE command follows:

```
DEVICE;  
  NAME = (Exec 0770 device mnemonic);  
  TYPE = TAPE;  
  MODE = [STANDARD|SINGLE|MULTI]  
  LABEL= [YES | NO];  
  EQUIPMENT = (Exec tape assign mnemonic, e.g. U40);  
  RECORD-FORMAT = (a PCIOS record/block specification);  
  BLOCK-SIZE = [512 | 1024];  
  REELS = (a list of up to 8 reel names, separated by "/")  
  CATALOGUE = [YES | NO];
```

A description of each keyword in the DEVICE command follows:

<b>NAME</b>	This keyword indicates the name of the device under Spin-X control. This is a valid Unisys device name from the sysgen.
<b>TYPE</b>	This keyword indicates the device type. This should be set to TAPE for devices to be used for the Spin-X ANSI Output Tape Feature.
<b>LABEL</b>	This keyword indicates the use of labeled or unlabeled tapes. If set to YES then Spin-X will expect a label on the tape. If set to NO then Spin-X will not expect a label on the tape. The default value is YES.
<b>RECORD-FORMAT</b>	<p>This keyword indicates the PCIOS record format of the data file. The default record format is DB (Variable-Blocked). Valid record formats are as follows:</p> <p><b>U</b>-(Undefined) Each data block contains only one record, and the records may be variable in length.</p> <p><b>F</b>-(Fixed-Unblocked) Each data block contains only one record, and all records are the same length.</p> <p><b>FB</b>-(Fixed-Blocked) Each data block holds one or more fixed-length records, with each block having the same number of records. The SPIN-X record size is 512 bytes.</p> <p><b>D</b>-(Variable-Unblocked) Each data block contains only one variable-length record. The record length is specified by four ASCII characters of control information.</p> <p><b>DB</b>-(Variable-Blocked) Each data block contains as many complete records as possible. The record length is specified by four ASCII characters of control information.</p>
<b>BLOCK-SIZE</b>	This keyword indicates the maximum number of characters per block of the data file. Valid block sizes are 512 and 1024. The default block size is 1024.
<b>EQUIPMENT</b>	This indicates the tape assign mnemonic. This indicates the equipment type required for the output file. (e.g. U9V, U95, U40, U47)
<b>REELS</b>	This keyword parameter should be followed by a list of the reels to be used with the tape device. The list may contain one to eight reel names. Each reel name should be separated with the "/" character and the last reel field should be followed by /NOMORE. The REELS = parameter may not be specified for modes single or multi.
<b>MODE</b>	<p>This parameter specifies one of three possible modes of tape processing: standard, single or multi.</p> <p>__ <b>Standard</b> mode indicates the traditional method of tape handling that caused SPIN-X to request a tape when the L* device I keyin was issued, and SPIN-X released the tape when the L* device L keyin was issued.</p> <p>__ <b>Single</b> mode indicates that a single file is placed on tape following the L* device I keyin, and then the tape is released.</p> <p>__ <b>Multi</b> mode indicates that multiple files are placed on a tape following the L* device I keyin until no files remain in the queues, then the tape is released.</p>
<b>CATALOG</b>	Yes or No determines whether or not the tape file will be catalogued by SPIN-X.

An example of the FDP DEVICE statement configured for the Spin-X ANSI Tape Feature is as follows:

```
DEVICE ;
  NAME=ANSTAP ,
  TYPE=TAPE ,
  LABEL=YES ,
  EQUIPMENT=U9V , ;
  RECORD-FORMAT=DB ,
  BLOCK-SIZE=1024 , ;
  MODE = MULTI ,
  CATALOG = YES ,
  REELS=SPX001/SPX002/SPX003/SPX004/SPX005/NOMORE
```

### I.2.3 Xerox Considerations

A recent patch has been developed for the Xerox Operating System Software which corrects a problem with processing ANSI labeled tapes with more than one record per block. Previously, if the blocked record format was used (either fixed-blocked or variable-blocked) the tape was required to be unlabeled. If a labeled tape was used, the record format had to be unblocked. The patch to fix this problem is SAC66A.

## I.3 Operations

### I.3.1 Operator Keyins

The operator may start and stop the Spin-X tape device with the following keyins:

```
L* tape I
L* tape P [queue | device]
L* tape L [Q | T | E]
```

...where *tape* is the FDP tape device mnemonic.

The **I** keyin causes the tape device to initialize and spool prints from its associated print queues.

The **P** keyin causes the specified tape device to spool the print files from a specific queue or spool the print files that are associated with another device.

The **L**, **Q**, **T**, and **E** keyins are the same for the tape device as for other Spin-X controlled devices.

### I.3.2 Processing of Keyins

When initiated, the tape device will begin by assigning reel number 1, this is the first reel number as specified in the REELS parameter on the DEVICE statement. (If no REELS parameter was specified, then a load (un) labelled request is made.) When reel number 1 is full, the Exec will perform a tape swap to the next reel. Spin-X will then assign reel number 2 and continue processing the print file(s). This is repeated until all prints are spooled or the last reel specified has been processed. If the last reel number has been filled and more print files are to be processed, then EXEC will request a labeled scratch tape to be mounted for further processing. This will be continued until all print files have been processed for the tape device and the tape device has been locked with the L keyin. If all print files have been processed for the tape device, the tape device will go waiting as a printer device would. The operator may issue an L keyin at this point to lock the tape device. In response to a L keyin Spin-X will close the spool tape, free the last reel, post a message to the operator, and lock the tape device. The operator may restart the process with an I or P keyin.

### **I.3.3 Errors**

---

There are three types of errors from which Spin-X must recover. These errors are input file errors, tape facility errors, and tape output errors.

#### **I.3.3.1 Input File Errors**

---

When an input file error occurs, Spin-X will generate an appropriate message, with ANSI spacing, and place it on the output tape. These messages will consist of messages currently defined in the *Spin-X Operator Guide*.

#### **I.3.3.2 Tape Facility Errors**

---

When the Spin-X **I** or **P** keyin is entered, Spin-X will attempt a tape assignment on reel number 1. If a tape drive is not available, then the SPIN-X tape activity will go into a facility hold until a tape drive is free. If a facility error occurs, then an error message will be posted to the operator console and the SPIN-X tape activity will terminate. Additional error messages will be output to the PRINT\$ file. These additional messages are defined in Table 9-1 of the *Processor Common Input/Output System (PCIOS) Interface Description* (UP-8478.2).

#### **I.3.3.3 Tape Output Errors**

---

If an error occurs while writing data to the spool tape, EXEC will post the standard I/O error messages and perform the recovery action prescribed by the operator. Spin-X will handle tape I/O errors only when the EXEC I/O error has been responded to with a "B". Spin-X will assume this to be the same as a Q keyin and will requeue the print file. Additional error messages will be output to the PRINT\$ file. These additional messages are defined in Chapter 9 of the *Processor Common Input/Output System (PCIOS) Interface Description* (UP-8478.2).

---

## **I.4 Translate Table User Exit (UXTAPETABLE)**

---

SPIN-X writes ASCII data to the ANSI output tape. The user exit programmer may modify the characters written to the ANSI tape by supplying a translate table with the entry point UXTAPETABLE. The SPIN-X user exit example UXASTAS that is supplied in the \*SRO program file may be useful as a starting point for development of the translate table.

See Section 7.4 for information on collecting user exits with SPIN-X.

---

## I.5 IBM Host Processing

---

### I.5.1 Overview

The ANSI formatted tape produced with the Spin-X ANSI Tape Feature may be processed by an IBM or compatible host. The print files that reside on the ANSI formatted tape contain ANSI carriage control and the tapes are in ANSI Version 1 Labeled. **NOTE:** MVS systems must ensure that the ASCII translate routine, IGC0010C, is included in the system.

### I.5.2 Installation Instructions for JESPRINT

The JESPRINT install tape, called JESPRT, has three files on it:

- File 1: The install jobstream
- File 2: The JESPRINT PDS containing source code, JCL to assemble and run the tape build JCL.
- File 3: The JESPRINT load module PDS.

Below is an IEBGENER jobstream to pull the first file - the install JCL - off the tape:

```
//FILE1 EXEC PGM=IEBGENER,COND=(8,LE)
//SYSUT1 DD DSN=JESPRINT.INST,DISP=(OLD,KEEP),UNIT=TAPE,
// VOL=SER=JESPRT,
// LABEL=(1,SL,EXPDT=98000)
//SYSUT2 DD DSN= <install.file>,DISP=(NEW,CATLG),
// UNIT=3380,VOL=SER=WRK001,SPACE=(TRK,(1,1)),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=6320)
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
```

### I.5.3 Example JCL

What follows is a sample JCL listing that will allow an IBM host to process a 1600 bpi ANSI formatted tape that was generated with the Spin-X ANSI Tape Feature.

```
//JESPRINT EXEC PGM=JESPRINT
/*JOBPARM LINECT=0,BYTES=99999
//STEPLIB DD DSN=SYS.JESPRINT.LOAD,DISP=SHR
//INDATA DD DSN='SPIN-X*SPX-TAPE',DISP=(OLD,KEEP),UNIT=(TAPE,,DEFER),
// VOL=SER=(SPX001,SPX002),
// DCB=(DEN=3,RECFM=DB,OPTCD=Q,BUFOFF=0),
// LABEL=(1,AL,EXPDT=98000)
//OUTDATA DD SYSOUT=A,DCB=(RECFM=FA,LRECL=260)
```

**NOTE:** The DSN parameter above consists of 'SPIN-X\*SPX-*device*'. Where *device* is the device name on the Unisys host.

## I.6 Xerox Offline Processing

The ANSI formatted tape may also be processed offline on a Xerox EPS printer. An offline JSL has been provided in order to read the tape offline.

### I.6.1 Listing of Xerox Offline JSL

The following is a listing of the Xerox source code for the offline Job Source Language (JSL) needed on the Xerox by SPIN-X.

```
SPTAPE:JDL;

ACODE:CODE DEFAULT=ASCII,
      ASSIGN=(X'7B',X'C0'),
      ASSIGN=(X'7C',X'FA')
,      ASSIGN=(X'7D',X'D0');

/* The above CODE reassignments are for redefining the left      */
/* curly brace, vertical bar and right curly brace to the      */
/* positions where Xerox fonts have them defined. If these     */
/* changes are not required change the ACODE below to ASCII.   */
/*                                                              */

VOLUME HOST=ANSI,
      CODE=ACODE
,      LABEL=ANSI;

BLOCK LENGTH=1024,PREAMBLE=0,LTHFLD=0;
RECORD LENGTH=150,PREAMBLE=4,STRUCTURE=VB,LTHFLD=4,FORMAT=DEC;
V1:VFU TOF=1,BOF=66,ASSIGN=(1,1),ASSIGN=(12,66),ASSIGN=(2,1);
V2:VFU TOF=1,BOF=63,ASSIGN=(1,1),ASSIGN=(12,63),ASSIGN=(2,1);
V3:VFU TOF=1,BOF=80,ASSIGN=(1,1),ASSIGN=(12,80),ASSIGN=(2,1);
V4:VFU TOF=1,BOF=132,ASSIGN=(1,1),ASSIGN=(12,132),ASSIGN=(2,1);

PCC1: PCC DEFAULT=ANSI, INITIAL=TOF, ADVTAPE=NO,
      ASSIGN=(X'20',SP1P), ASSIGN=(X'2B',SP0P),
      ASSIGN=(X'2D',SP3P), ASSIGN=(X'31',SK1P);

ACCT USER=NONE;

IDEN PREFIX=A'%%DJDE',SKIP=8,OFFSET=1,OPRINFO=NO;

T1: TABLE CONSTANT=(X'01020102');
C1: CRITERIA CONSTANT=(1,4,EQ,T1);
T2: TABLE CONSTANT=(X'02010201');
C2: CRITERIA CONSTANT=(1,4,EQ,T2);
T3: TABLE CONSTANT=(X'02010102');
C3: CRITERIA CONSTANT=(0,4,EQ,T3);
```

```

RAUX TEST=C1;
RPAGE TEST=C2,WHEN=BOTTOM;
RSTACK TEST=C3,DELIMITER=YES,HRPTNA=(4,16);

LINE DATA=(1,150),PCCTYPE=PCC1,VFU=V1,UCSB=IGNORE;

DFLT:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT1X;

LS132:JDE;
    OUTPUT DUPLEX=NO,FORMAT=FMT1X;

PR:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT1X;

PRT:JDE;
    OUTPUT DUPLEX=NO,FORMAT=FMT1X;

LD132:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT1X;

LD150:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT2X;
    LINE DATA=(1,150);

PS80:JDE;

    OUTPUT DUPLEX=NO,FORMAT=FMT8X;
    LINE DATA=(1,80); VFU=V2;

PD80:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT8X;
    LINE DATA=(1,80); VFU=V2;

PS95:JDE;
    OUTPUT DUPLEX=NO,FORMAT=FMT7X;
    LINE DATA=(1,95);

PD95:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT7X;
    LINE DATA=(1,95);

PD106:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT6X;
    LINE DATA=(1,106); VFU=V3;

PD132:JDE;
    OUTPUT DUPLEX=YES,FORMAT=FMT10X;
    LINE DATA=(1,132); VFU=V4;

PD132T:JDE;
    OUTPUT DUPLEX=YES,FORMAT=TWOUP;

LABELS:JDE;
    OUTPUT DUPLEX=NO,FORMAT=LABEL6;
    LINE DATA=(1,35);
END;

```

## **I.6.2 Problems With Offline Processing**

Due to certain Xerox requirements concerning carriage control processing during file downloading, the ANSI tape feature cannot be used to download files to the printer's disk. This prohibits using the "BATCH=" DJDE to download image files such as those output by XPPI (formerly EPIC). Image files (type .IMG) may still be called up and printed using the IMAGE DJDE.

---

# J

# Samples of Available Formats

The following pages are examples of printed output using the available print formats provided with the SPIN-X package.