
Corsair Technology, Inc.



CENTRAL

**Reference Guide
for the Siemens-Nixdorf Electronic Printers
Version 4R5**

Document Revision February 1, 1996

Corsair Technology, Inc.
10 Park Place South, Suite LLE
Atlanta, Georgia 30303

Document revision date - February 1, 1996

Central Document Version 4R5

Printed in the United States of America

SPIN-X® is a trademark of Corsair Technology, Inc. Georgia State
University
Research Foundation Incorporated.

Siemens® is a trademark of the Siemens Corporation

SNPS is a trademark of the Siemens Nixdorf Printing Systems
Corporation

Unisys® is a trademark of Unisys Corporation.

IBM® is a trademark of International Business
Machines Corporation.

© 1996 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.

Contents

Preface	I-i
1. Introduction	1-1
2. Overview	2-1
2.1. Page Layout	2-1
2.2. Font	2-1
2.3. Electronic Forms	2-2
2.3.1. EFOs	2-2
2.3.2. FOBs	2-2
3. How to Use the Format Definition Program (FDP)	3-1
3.1. The DEVICE Command	3-3
3.2. The FONTFILE Command	3-4
3.3. The SUBST Command	3-4
3.4. The FORMAT command	3-5
3.5. Executing the Format Definition Program	3-8
4. Forms Overlay Buffer	4-1
4.1. Overview of the FOB process	4-1
4.2. Creating FOBs	4-2
4.2.1. Form Generation Utilities	4-2
4.2.2. Converting Scanned Images to FOBs	4-2
4.3. Using the FOB Utilities	4-3
4.3.1. The SPFOBCAT Processor	4-3
4.3.2. The FOB2EFO Processor	4-3
4.3.3. The SPFOB2DB Processor	4-4
4.3.4. The FOBCON Processor	4-4
5. The PLIST Processor	5-1
5.1. Running PLIST	5-1
5.2. Output of the PLIST Processor	5-1
6. The Central Environment	6-1
6.1. SPIN-X Central Internals Overview	6-1
6.2. Why the Bus and Tag Cable Adapters are Needed	6-2
6.3. The Unisys to IBM 3800 Environment Connection	6-3
6.3.1. Direct Connection	6-3
6.3.2. Connection Via a Unisys Channel Transfer Switch	6-3
6.3.3. Connection Via a Channel Interface Switch	6-4
6.3.4. Connection Via a Channel Extender	6-4
6.3.5. Considerations of Multiple Host / Device Connections	6-4

7. User Programming with SPIN-X Central	7-1
7.1. Using the @SYM Command to Specify Print Formats	7-1
7.2. Using Unisys 1100/2200 Print Control Images	7-1
7.3. Print Output Error Messages	7-2
7.4. Alternative Font Switching Using Siemens Escape Codes	7-3
7.5. Siemens Bar Codes With SPIN-X Central	7-4
7.5.1. Transparent Data Mode	7-4
7.5.2. Siemens Nixdorf Bar Codes	7-5
7.5.3. Siemens Nixdorf POSTNET ZIP4 Font	7-5
8. Local Modifications	8-1
8.1. Central Defined User Exits	8-1
8.1.1. Banner Page	8-3
8.1.2. Trailer Page	8-4
8.1.3. Type 14 Log Entry	8-5
8.1.4. Central Data Structures Passed to User Exits	8-6
8.1.4.1. ASCII Edit Packet (AED)	8-7
8.1.4.2. Printer Control Buffer	8-8
8.1.4.3. SMOQUE Entry (SQE)	8-8
8.2. Modifying the Character Translation Table	8-9
8.3. Collecting Central	8-10
9. Setting-up a Test Central Run	9-1
9.1. Steps for Setting-up a Test Central	9-1
9.2. Example ECL for Test Central Batch Job	9-2

A. Central Log Entries	A-1
A.1. Fieldata Log Entries	A-1
A.1.1. Symbiont End of Processing Log Entry (Type 14)	A-1
A.1.2. Symbiont Start of Processing Log Entry (Type 15)	A-4
A.1.3. Symbiont Handler Log Entry (Type 35)	A-6
A.2. ASCII Log Entries	A-9
A.2.1. Symbiont Start of Processing Log Entry (Type 10112 Subtype 1)	A-9
A.2.2. Symbiont End of Processing Log Entry (Type 10112 Subtype 2)	A-11
A.2.3. Symbiont Device Error Log Entry (Type 11502)	A-14
A.2.4. Symbiont I/O Fault Error Log Entry (Type 11503)	A-16
B. ECL to Create a SPIN-X Central Absolute (MAP)	B-1
C. Merging Font Input Files	C-1
D. Listing of the Distributed FDP Input File	D-1
E. 061 Print Controls	E-1
E.1. The PLIST Utility	E-1
E.2. Using 061 Images in Printfiles	E-1
E.2.1. Accessing Fonts with the Load Character Arrangement (LCA)	E-2
E.2.1.1. Substituting for a Font on Diskette	E-2
E.2.1.2. Substituting for Element Fonts	E-4
E.2.1.3. LCA Technical Notes	E-5
E.2.2. The Select Electronic Form (SEF) Print Control	E-5
E.2.2.1. SEF Technical Notes	E-6
E.2.3. The Load Copy Number (LCN) 061 Image	E-7
E.2.3.1. LCN Technical Notes	E-7
E.2.4. The Vertical Format Buffer (VFB)	E-7
E.2.5. The Load FLASH Number (LFN) 061 Image	E-9
E.2.5.1. LFN Technical Notes	E-9
E.2.6. The Load Special Forms (Subtype 010) 061 Image	E-10
E.2.7. PRINT\$ Data Records	E-10
F. ECL to Copy FDP Files to Production (FDP/TOPRODUCTION)	F-1
G. The SPIN-X Central Utilities	G-1
G.1. CLEANQ	G-1
G.2. SMOQUETEST	G-1
H. Sample "L" Processor Call Option Listing	H-1
I. ASCII to EBCDIC Translation Table	I-1

The Central software provides on-line access to Siemens electronic printer environments from Unisys Series 1100/2200 or System 11 computers via a byte or block multiplexor channel.

This reference guide contains information which is necessary to tailor Central to your site's specific needs.

The following three documents are included with the Central software:

- *SPIN-X Central Installation Guide.*
- *SPIN-X Central Reference Guide.*
- *SPIN-X Central Operator Guide.*

This reference guide assumes the reader has access to the following documents (for EXEC Level 39R3A or higher):

- *OS2200 Exec System Software Executive Control Language (7830 7949-004)*
- *OS2200 Exec System Software Administrative Reference Manual (7831 0323-007).*
- *OS 2200 System Service Routines Library (SYSLIB), Programming Reference Manual (7833 1733-00x)*
- *Series 1100 Data Structures, Programming Reference Manual (7833 3481-000)*
- *Siemens Enhanced Mode Software (EMS) Programmer's Guide (U3606-J-Z47-1-7600)*

This SPIN-X software package provides on-line access to Siemens electronic printers from Unisys 1100/2200 computers via a byte or block multiplexor channel. The information provided in this manual is applicable to SPIN-X Central's support of the following printer models that use the APA2 controller:

Siemens Electronic Printers:

2090
2140
2240

Central utilizes Siemens High Performance Mode software (HPM) for support of multiple font printing and special print formatting. These capabilities support font switching between lines or amongst characters in a line. The Central support of these features is discussed throughout this guide.

The Siemens printer is configured to EXEC as a Unisys printer. Commonly, sites using Central configure the Siemens printer as a Unisys 0770 or 0776 printer. Central recognizes this printer as a Siemens electronic printer because it is also defined separately in a SPIN-X Central database. The creation and maintenance of this database is controlled by the Format Definition Program (FDP) as documented in Chapter 3.

This reference guide also contains accounting information for the traditional fielddata log entries for EXEC levels before 40R2 and the more recent ASCII log entries for EXEC 40R2 and higher; see Appendix A. Also, SPIN-X Central "user exits" are documented that permit a site to supply its own code to replace certain default functions of Central such as the layout of the banner page; see Chapter 7.

SPIN-X Central utilizes information on page layout, fonts, and electronic forms to format an output print with a Siemens printer. Central receives this format information from two sources:

- 1) 060 and 061 print control records contained within a printfile
- 2) Central's Format Definition Program (FDP)

The remainder of this chapter discusses how Central receives page layout, fonts, and electronic forms for the Siemens printer using FDP and 060/061 print controls. The details of running the FDP processor are provided in the next chapter. The details of the 060/061 print controls are provided in Chapter 4 of the Series 1100 Data Structures manual. The interpretation by Central of the 061 print controls is explained in Appendix E.

2.1 Page Layout

The page layout consists of the number of lines per page, the top and bottom margins expressed in lines, and the lines per inch such as 6 or 8. A default page layout is defined to Central by the FDP Format command. The Format's page layout can be overridden by the printfile with a 060 "M" print control or 061 Load Vertical Format Buffer control. The 060 and 061 print controls are created by ER SYMB\$.

2.2 Font

The font consists of a set of characters to use for printing. The 061 print control Load Character Arrangement defines fonts to be utilized for the printfile. The FDP SUBST command is used to substitute Siemens fonts for the fonts in the Load Character Arrangement control. This permits fonts to be substituted without changing the original application. For printfiles not containing the Load Character Arrangement control, the FDP Format command can provide a default set of fonts for the print.

Included with the release package is a utility called PLIST which automates the process of searching already existing printfiles and locating the print controls. See Chapter 5 for a description of this processor.

2.3 Electronic Forms

Two varieties of electronic forms are supported by Central to the Siemens printer: Electronic Form Overlay (EFO) and Forms Overlay Buffer (FOB). (On earlier generation printers, the EFO was called the "flash".)

2.3.1 EFOs

The EFO consists of an electronic form created on a diskette by PC software. The diskette is inserted in the printer by the operator when called for by a 061 print control Load Flash Number. This 061 print control can turn on or off the EFO for a page of the print.

2.3.2 FOBs

A FOB is a data structure that consists of one or more electronic forms. A FOB is created on the PC and resides as an element in the Central file SYSS\$PRINTER\$*SPIN-X. Central downloads a FOB to the Siemens printer in response to the 061 print control Select Electronic Forms. Each electronic form is assigned a copy reference number with the PC software. The 061 print control Load Copy Number is used to select electronic form(s) for a page of a print.

For more information on creating FOBs and EFOs, see Chapter 4.

3 How to Use the Format Definition Program (FDP)

The FDP processor creates a database that Central references during execution to facilitate driving the Siemens printer. The information maintained in the database consists of three major categories:

1. Printer name(s) - these are the printer name(s) to be driven by Central; they match device names from the EXEC sysgen. In the SPIN-X database, the true printer type is identified such as a Siemens 3800 electronic printer emulation. The FDP DEVICE command defines a Central printer.
2. Print formats - these objects define default print attributes, e.g. the print density, to be used for printing files obtained from a particular EXEC queue. The name of the print format matches the queue name from the EXEC sysgen. The FDP FORMAT command declares a print format.
3. Fonts - these are the fonts that are downloaded to the printer by Central. A font consists of a translate table and the bit patterns necessary to define the font to the printer. The FDP FONTFILE command defines an SDF file containing fonts.

The SPIN-X Central administrator provides four keyword-oriented commands to FDP to determine information placed in the database:

1. **DEVICE** - this command specifies a device to be controlled by Central.
2. **FORMAT** - this command defines a print format.
3. **FONTFILE** - this command informs FDP of the input file containing fonts for inclusion in the Central database.
4. **SUBST** - this command gives FDP the ability to substitute fonts for 061 images already existing in the printfile. This makes it possible to use FDP to specify substitute fonts for the print output without actually changing the 061 images in the application.

Each of these keyword commands are discussed in this chapter.

The "database" created by FDP consists of three files:

- 1. FORMS-DEF** - this SDF file contains device definitions.
- 2. SRI\$DATABASE** - this MSAM file contains an index of device definitions, print formats, and downloadable fonts. This index points to the location of an object in a flatfile. Central "looks-up" the object it requires in the index when needed; it then obtains the object, such as a font, 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 Central during runtime, and it does not conform to any standard OS1100 file definition.

The purpose of SRI\$DATABASE and SRI\$FLATFILE is to allow Central to maintain most of the information it needs on disk while it is executing, thus reducing the amount of core memory required. Keeping the objects on disk is particularly necessary for downloadable fonts which are each commonly 8k or 16k bytes in size.

Sections 3.1, 3.2, 3.3 and 3.4 define and explain the syntax used in the DEVICE, FONTPFILE, SUBST and FORMAT commands respectively.

In the syntax explanations, the parentheses indicate a value for a parameter; while bracketed text whose elements 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, also, any characters appearing after column 80 will be ignored.

Parameters may be delimited by commas or spaces and may appear in any given order. A command is terminated when there is no continuation indicated.

Only one FONTPFILE command is contained in an FDP-INPUT.

Finally, Section 3.5 explains how to execute the Format Definition Program and generate the FORMS-DEF and database files.

3.1 The DEVICE Command

The DEVICE command allows the SPIN-X Central administrator to specify which printing devices SPIN-X will be allowed to control. The syntax of the DEVICE command is shown below:

```

DEVICE;
  NAME =(a legal Unisys device name), ;
  TYPE =(a valid Siemens electronic printer model)
  DCODE =(an octal number between 0 and 77);
  MESSAGE_GROUP =(Exec Console Message Group or Terminal Siteid);
  MESSAGE_GROUP_NUMBER=[0,1,..63];
  CONSOLE_TERMINAL=cms_siteid;
  EOF_SKIP_MESSAGE = [Y | N];

```

The following is a detailed explanation of each keyword in the DEVICE command.

NAME	A 1.6 character name where the first character is in the alphabet and each of the others is in the alphabet or is a digit.
TYPE	Valid DEVICE TYPE values are: <pre> SIEMENS_2090 SIEMENS_2140 SIEMENS_EPS </pre>
DCODE	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.
MESSAGE_GROUP	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.
MESSAGE_GROUP_NUMBER	This parameter specifies the console message group number for directing this device's console messages after Central initialization.
CONSOLE_TERMINAL	This parameter specifies the <i>cms siteid</i> for directing this device's console messages after Central initialization.
EOF_SKIP_MESSAGE	If this parameter is set to YES then SPIN-X will post message 000409:

SKIPPED TO EOF - REPRINT? PAGES OR N

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.

The following example DEVICE command is included with the SPIN-X software in the file *FDP-INPUT that was copied during installation of the 1100/2200 Release Tape.

DEVICE NAME=SEPS, TYPE=SIEMENS_EPS

NOTE: the device name must match the name chosen for the printer during system generation.

3.2 The FONTFILE Command

The FONTFILE command informs FDP of the file for input that contains the Siemens fonts for output to the SPIN-X database. The fontfile from the SPIN-X release tape is "**FONTFILE*".

example:

FONTFILE *SPIN-X *FONTFILE*

3.3 The SUBST Command

The SUBST command permits a font defined in FDP to be substituted for a font defined in a 061 LCA print control. The syntax for the SUBST command follows:

```
SUBST NAME=(substitute_id);  
      TYPE=(device_type);  
      FONT=(font_name);  
      ELEMENT=(061_element_name);  
      DISKETTE=(061_diskette_id, font_number);  
      DEFAULT=[No | Yes]
```

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

- **NAME** - a 1..8 character name for identifying this substitution statement within the FDP configuration. A substitute_id may not have the value "default\$".
- **TYPE** - a valid FDP mnemonic for Siemens printers, e.g. SIEMENS_EPS.
- **FONT** - This field specifies the font that will be substituted for the one in the print file. This font would be a Siemens font in SPIN-X*FONTFILE.
- **ELEMENT** - 061_element_name should be the same as the element name specified in a user's 061 print control Load Character Arrangement. Central uses 061_element_name as a key to look up the appropriate replacement font specified in the FONT= parameter. This parameter is mutually exclusive to the DISKETTE= parameter.
- **DISKETTE** - 061_diskette_id, font_number should be the same as the corresponding values specified in a user's 061 print control Load Character Arrangement. Central uses 061_diskette_id, font_number as a key to look up the appropriate replacement font specified in the FONT= parameter. This parameter is mutually exclusive to the ELEMENT= parameter.
- **DEFAULT** - This parameter set to Yes indicates that this SUBST should be put in the default pool of SUBST commands. The default for DEFAULT= is "No".

A set of SUBST commands is included in the *FDP-INPUT from the Central release tape as samples. See Appendix D for examples of the SUBST command.

Currently, Central imposes the limit of 200 SUBST commands with DEFAULT=Y.

3.4 The FORMAT command

The FORMAT command allows the SPIN-X 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:

FORMAT;

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

TYPE = (a valid alphanumeric device type name),;

SUBSTITUTES = (substitute_id1,substitute_id2,...),;

BANNER_FONT= (the downloadable font for printing the banner page),;

REPORT_FONT= (the downloadable font(s) for printing the report body),;

TRAILER_FONT= (the downloadable font for printing the trailer page),;

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

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

DENSITY =[6 | 8 | 12],;

BANNER_DENSITY =[6 | 8],;

TRAILER_DENSITY =[6 | 8],;

TOP_MARGIN = (an integer indicating number of blank lines at top),; {**0**}

BOTTOM_MARGIN = (an integer indicating number of blank lines at bottom),; {**0**}

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

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

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

BYPASS_FONT_61=[Y|N]; .Y=yes, N=no

BYPASS_FONT_TTN=[Y|N] .Y=yes, N=no

TWOUP=[Y|N] .Y=yes, N=no

ROTATE=[0 | 90 | 180 | 270]

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

- **NAME** - This parameter provides a queue name to the print format being defined by this FORMAT command, e.g. NAME=BILRPT. The format name must match the name of the configured EXEC queue that is referenced by the third field of the @SYM command.
- **TYPE** - This parameter identifies the device type for this format. For this value put "SIEMENS_EPS".
- **SUBSTITUTES** - This parameter indicates which fonts defined by the SUBST commands will be substituted for Unisys 061 print functions. This parameter takes the form:

SUBSTITUTES=(substitute_id1, substitute_id2,...)

Currently, SPIN-X supports up to 50 substitute_ids in the SUBSTITUTES= parameter. The default pool of SUBST commands are searched to satisfy a font reference in a printfile if the SUBSTITUTES= parameter does not specify the substitute_id for that font.

- **BANNER_FONT** - This parameter specifies the font to be used for printing the banner page. It is 1-4 alphanumeric characters.
- **REPORT_FONT** - This parameter specifies the fonts to be used for printing the report body. A font name consists of from 1-4 alphanumeric characters. If more than one font is specified, the font names must be enclosed in parentheses. Central permits up to sixteen fonts to be specified. The 2090 and 2140 printers standardly permit 46 simultaneous fonts. The following example specifies one font for printing the report; the succeeding example specifies four fonts for printing the report:

REPORT_FONT = (004M,004M) . defaults for Fielddata, ASCII
REPORT_FONT = (004M,004M,008M,031M) .Fonts with TTN's 0 - 3

In the second example above, 004M is font zero, 004M is font one, 008M is font two, and 031M is font three. The application programmer will need to be made aware of the font ordering in the REPORT-FONT parameter for proper font selection within the user's printfile; see section E.2.7 for information on selecting fonts within applications.

The **REPORT_FONT =** provides an alternative to using SUBSTs and 061 controls for specifying fonts to be downloaded.

- **TRAILER_FONT** - This parameter specifies the font to be used for printing the trailer page. It is 1-4 alphanumeric characters. See the accompanying font guide for a listing of available fonts.

NOTE: A substantial performance improvement is realized if the **BANNER_FONT** and **TRAILER_FONT** each specify the same font.

- **DENSITY** - This parameter specifies the default LPI of the format. This value must be either 6, 8 or 12 LPI.
- **BANNER_DENSITY** - This parameter indicates the lines per inch density for the banner page; the choices are 6 or 8.
- **TRAILER_DENSITY** - This parameter indicates the lines per inch density for the trailer page; the choices are 6 or 8.
- **LINES** - This parameter indicates the total number of lines allowed to be printed before Central forces a top-of-page. This number includes both the **BOTTOM-MARGIN** and **TOP-MARGIN** values.
- **LINES8** - This parameter is the same as the **LINES** parameter only it indicates the number of lines per page for 8 LPI mode.
- **BOTTOM_MARGIN, TOP_MARGIN** - These parameters refer to the number of blank lines to be defined for 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. The standard is six blank lines at the top and three blank lines at the bottom. Both of these values should be included within the calculation of the **LINES** parameter. These parameters may be abbreviated to "TOP" and "BOT".
- **WIDTH** - This parameter informs SPIN-X of the maximum number of characters allowed per line. Any characters in excess of this number will be truncated by Central before being sent to the printer.
- **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 Central 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 Central. 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.

- **DYNAMIC** - For the Siemens printers, DYNAMIC=Y is assumed irregardless of the value to which it has been set.
- **BYPASS_FONT_61** - If the FDP Format parameter Bypass_Font_61=Yes is provided, then Central will ignore font download specifications in LCA 061 requests. The fonts for the report then should come from the Format Report_Font= parameter. The Bypass_Font_61= parameter is independent of the Bypass_Font_TTN= parameter.
- **BYPASS_FONT_TTN** - If the FDP Format parameter Bypass_Font_TTN=Yes is provided, then Central will ignore the translate table number in PRINT\$ data records; thus, Central will not switch fonts.
- **TWOUP** - If this option is set to Y, then on 17 in. wide paper two pages will be printed side by side. This option can be used in conjunction with ROTATE.
- **ROTATE** - The entire body of text on the page will be rotated according to the option selected

Examples:

```
FORMAT NAME=SPINX1,TYPE=SIEMENS_EPS,;
  BANNER_FONT=001M,REPORT_FONT=(003M,003M),TRAILER_FONT=001M,;
  LINES=66,LINES8=88, TOP=0,BOT=0,DENSITY=6
```

```
FORMAT NAME=SPINX2,TYPE=SIEMENS_EPS,;
  BANNER_FONT=001M,REPORT_FONT=(020M,020M),TRAILER_FONT=001M,;
  LINES=66,LINES8=88, TOP=0,BOT=0,DENSITY=8
```

```
FORMAT NAME=REPRT1,TYPE=SIEMENS_EPS,;
  BANNER_FONT=001M,REPORT_FONT=NONE,TRAILER_FONT=001M,;
  LINES=66,LINES8=88, TOP=0,BOT=0,DENSITY=6
. font substitutes are from default=y substs
```

Examples of Font Substitution using SUBSTITUTES parameter

```
FORMAT NAME=FORMT1, TYPE=SIEMENS_EPS,;
  SUBSTITUTES=(S2_012M)
```

```
FORMAT NAME=FORMT2, TYPE=SIEMENS_EPS,;
  SUBSTITUTES=(S2_012M, S2_005M)
```

3.5 Executing the Format Definition Program

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

First, the input file, *FDP-INPUT., containing the FDP commands must be updated to contain the desired changes. Next, issue the following ECL command:

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

...where *qual* is the qualifier used to catalogue the SPIN-X Central files during the initial installation of Central.

This addstream will assign the input file (*FDP-INPUT) and fonts file (*FONTFILE), create a new Format Definition File (*FORMS-DEF), a new SPIN-X Repository Database consisting of two files (*SRI\$DATABASE and *SRI\$FLATFILE) and create a new report file which lists all formats and devices. All output files are created as new cycles of existing files.

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 a syntax 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 Central reads its information. To do this, issue the following command:

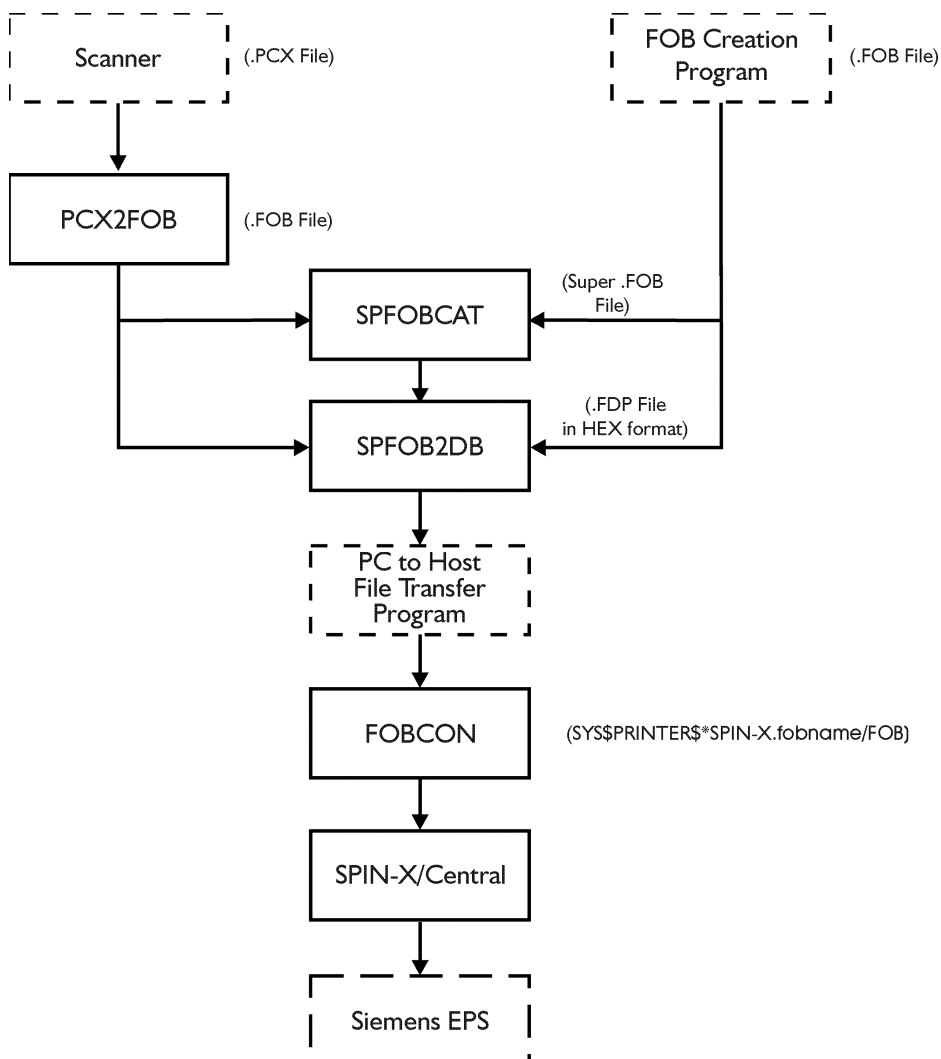
```
@ADD,L *FDP.FDP/TOPRODUCTION
```

Central should be terminated and re-started for the new configuration to be recognized.

A Forms Overlay Buffer (FOB) is an image which is in a format compatible with Siemens EPS printers. FOBs are downloaded by Central to the Siemens EPS. This section describes the issues involved in creating, manipulating, and transmitting FOBs and explains how to use the various FOB utilities which are included in the release package. Section 4.1 graphically depicts the various FOB processors and their relation to each other. Section 4.2 is divided into subsections containing the syntax for using each of the processors.

4.1 Overview of the FOB process

Figure 4.1



In the diagram to the left, each processor supplied in the SPIN-X Central package is outlined in solid lines.

The output of each processor is shown in parentheses to the right of each box.

When the output of one processor is the input of another, this is indicated by an arrow.

At the top, two alternatives are shown for creating FOBs: these are shown feeding into SPFOBCAT. This step is optional, however if several FOBs need to be downloaded, this step would splice all the FOBs together so that they could be downloaded all at once.

SPFOB2DB converts the binary FOB files into .fdb files in hex format suitable for transfer to the 1100/2200.

FOBCON converts .fdb files into sdf files which are output as elements to SYS\$PRINTER\$*SPIN-X.

4.2 Creating FOBs

4.2.1 Form Generation Utilities

There exist several utilities for generating FOBs . Among these are SIEFORM and ELIXER. For a more complete set of options, contact your Siemens representative.

4.2.2 Converting Scanned Images to FOBs

FOBs can be generated in a variety of ways. One simple method is to take a scanned or generated image in PCX format and save it as a one bit black and white image. This could be done with PC Paintbrush from Windows. Then use the PCX2FOB utility on your PC to convert it into a FOB. This processor should be invoked as follows:

```
pcx2fob /b
```

The /b switch instructs the processor to output the FOB in binary form. This is the format which is used for input to all the other FOB utilities. Failure to use the /b switch will cause the output to be in an ASCII format suitable for transfer to an IBM mainframe.

Next the user will be prompted to enter the PCX file name for conversion and the FOB file name to create.

Finally, the user will be prompted to enter the "positive horizontal and vertical shift" in Pels (there are 240 pels per in.). When these values are entered the processor computes the total image width and height as it will appear on the page with the given offsets . These values are displayed to the screen. If the user intends to convert the FOB into an EFO, then these values should be noted because the FOB2EFO processor will ask for them.

4.3 Using the FOB Utilities

Included in the release package are several utilities which are useful when working with FOBs. Two or more FOBs can be spliced into one using SPFOBCAT and FOBs in binary format can be converted to doublebyte hex using SPFOB2DB. The FOBCON processor converts the output of SPFOB2DB and places it in the specified program file on the Unisys host. If an image is to be loaded onto the Siemens printer directly from a diskette as an EFO, FOB2EFO is used to put the FOB in the EFO format which is acceptable to the printer.

4.3.1 The SPFOBCAT Processor

The instructions below are taken from the first page of the SPFOBCAT.C program file. This utility is used for concatenating two or more FOBs into one on your PC. The spliced FOBs will then require the printer to accept only a single downloaded FOB file rather than several.

Routine for splicing two or more Form Overlay Buffers (FOBs) together.

Type "spfobcat fob1 fob2 ...fobn" and press Enter. The spliced FOBs will be in the file "fobn". If "fobn" already exists, then it will need to be empty, else an error message will be generated.

There are two binary files fob_pre and fob_suf, used for error checking. These files must reside in the same directory as the SPFOBCAT processor. If it should become necessary to recreate these files, type "spfobcat -s fobname" where **fobname** is the name of a valid FOB .

4.3.2 The FOB2EFO Processor

The FOB2EFO processor is used on your PC for putting FOBs into an EFO format which can then be read directly from a diskette to the Siemens printer. Before the processor is run, the FOB image width and height in Pels should be known. The width value will indicate the horizontal displacement of the image and the height value will give the vertical displacement. The Pel is the smallest addressable unit on the printed page: for a 2140/2190 printer, there would be 240 Pels per inch. Values for image height and width are output from the PCX2FOB processor and should be noted when they are displayed. However, if the file size of the EFO is of no consequence to the user, then the maximum values (4080 Pels) for width and height can be used without any deleterious effect.

Usage:

Type "fob2efo". The processor will prompt for a FOB name. Enter the path and FOB name to be converted.

Now the processor will prompt for the name of the EFO. The format of an EFO name is as follows:

DIAxxxx.EFO

where *xxxx* is a 5 digit diskette i.d. number. This is the same number that will have to be keyed into the front panel of the printer in order to identify the EFO that is to be utilized. Enter the EFO name in the format shown above.

Next the processor will prompt for the width and height values discussed above. If these are not known, enter the maximum value of 4080 for each. Upon receiving this information, the processor will create an EFO.

4.3.3 The SPFOB2DB Processor

This is a routine used on the PC for converting binary files into their equivalents in doublebyte hex for the purpose of transmitting files to the Unisys 1100/2200. Output files will have same name but a .fdb extension.

Usage: Type "spfob2db [switches] [arguments]"

Type "spfob2db infile_1 infile_2 ...infile_n" where infile is a Form Overlay Buffer (FOB) in the current directory having the .fob extension. Each output file will have the same name but a .fdb extension.

Type "spfob2db -d dir_1 dir_2 " where dir_1 is a full path to a directory where FOBs are stored and dir_2 is a full path to a directory where the doublebyte output is stored. All FOBs in dir_1 will be converted and stored in dir_2. If -b and -d are used in conjunction, all files in the directory will be converted, not just FOBs, and error checking will be bypassed.

The binary files fob_pre and fob_suf, which are used for error checking, must reside in the same directory as SPFOB2DB.

Use -b switch if the file(s) to be converted are arbitrary binary files. The -b option will bypass FOB error checking so that the processor will accept a binary file that is not a FOB.

Use -o switch if files which already exist are to be overwritten without an overwrite prompt being written to the screen.

Use -s switch only if the files fob_pre and fob_suf are lost or need to be changed.

4.3.4 The FOBCON Processor

The FOBCON processor resides on the 1100/2200. It accepts a FOB (Forms Overlay Buffer) in SDF format and outputs it as an element in SYSSPRINTER\$*SPIN-X for use by SPIN-X Central.

FOBCON is invoked with the following processor call statement. (Brackets denote an optional item to be included).

```
@SPIN-X*UTILITY.FOBCON sdf_filename.[,element[/FOB]][,prog_file.]
```

where **sdf_filename** is an SDF file which contains the data.

where **[element[/FOB]]** is the name of the omnibus element that will contain the output FOB data. The "/FOB" can be optionally supplied by the user - otherwise it will be assumed upon output to the program file. If the entire element name is omitted, then the element name will be assumed to be the same as the one that comes from the original PC filename.

and where **[prog_file]** is an optional parameter specifying the program file for the output element specified in field 2 above. If this parameter is omitted, then SYSSPRINTER\$*SPIN-X is assumed as the output program file.

The PLIST host processor accepts an SDF print file as input and produces as output a listing of useful information concerning the records making up the print file. PLIST should be run on a printjob before making font substitutions in FDP to obtain a listing of the appropriate 061 control images. This chapter will explain how to run PLIST and also how to use the output of PLIST.

5.1 Running PLIST

PLIST is invoked with the following processor call statement.

```
@SPIN-X*UTILITY.PLIST[,options] sdf-input-file [,sdf-output file]
```

where options are:

A - Lists 060 type control records and the Data records in ASCII.

O - Lists 060 type control records and the Data records in Octal words.

P - Lists only the 060 and 061 type control records.

where sdf-input-file is the source SDF file

and where sdf-output-file is an optional temporary file for capturing the output of the processor. If this file is not specified, the output will be directed to the terminal screen.

5.2 Output of the PLIST Processor

1. Control word of the record - All records in the print file are preceded by a control word. The control word will indicate whether the record is in ASCII or FIELDATA and also whether the record is a data record or a print control record.

2. Record offset in the source print file - This gives the location of the record.

3. Words in the Record image, excluding the control word - this is the size of the record in Words.

4. Record image:

For 050 type control image: FIELDATA or OCTAL or both.

For 060 type control record: ASCII or FIELDATA or OCTAL or both FIELDATA and OCTAL.

For 061 type control record: OCTAL dump and the values of all the subtype/fields in the control record.

For all other types of control records: OCTAL dump.

For Data records: ASCII or OCTAL or both.

This section discusses the internal environment of Central, as well as its external environment.

Section 6.1 gives a basic explanation of how Central works internally, that is, it describes the functions that Central performs in order to send print files to the printer. Section 6.2 explains the purpose of the SPIN-X bus and tag cable adapters. Section 6.3 discusses several considerations for the connection of the Unisys host to the printer environment, which is necessary for the proper placement of the SPIN-X bus and tag cable adapters.

6.1 SPIN-X Central Internals Overview

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

The Central 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 Central is a user-program and not local code in the Unisys 1100/2200 Executive, it will appear as a normal batch run. The internal Central 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 Central functions is shown in Figure 6-1. The numbers shown in the figure correspond to the numbered functions listed above.

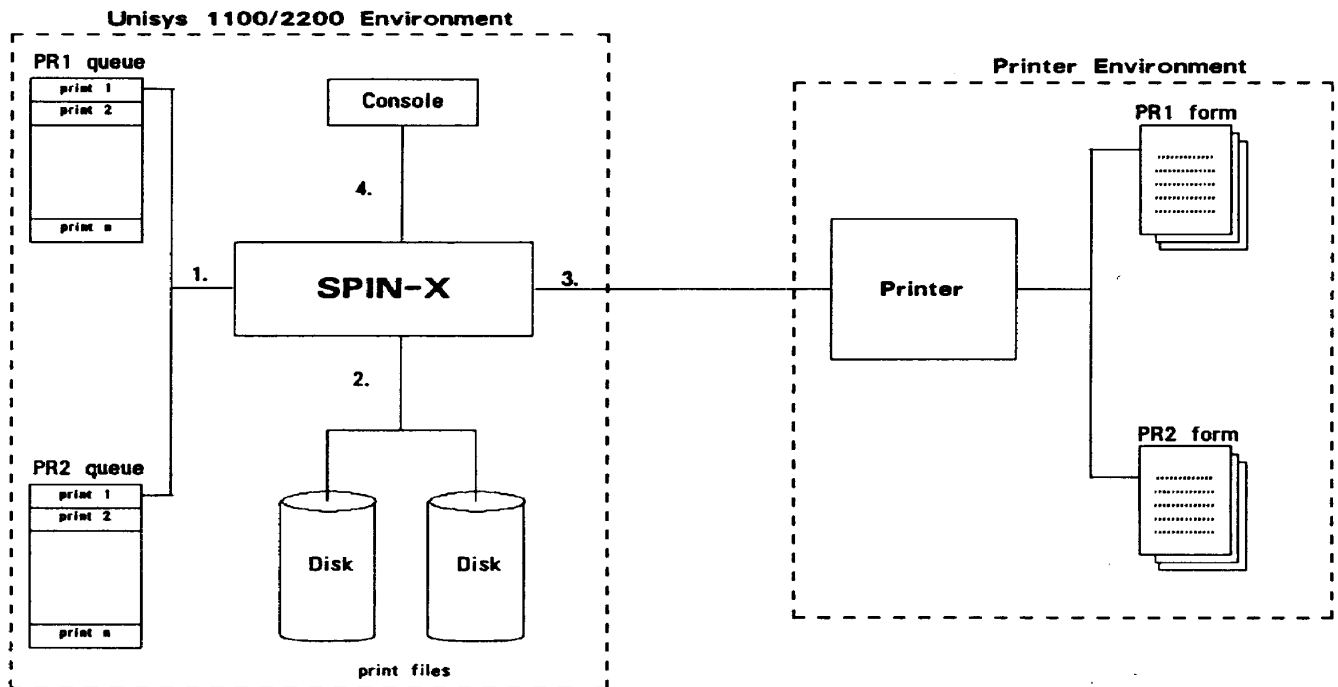


Figure 6.1 Diagram of Internal SPIN-X Central Functions

6.2 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 emulators cannot accept it. **Therefore, if your site daisy chains Unisys devices with IBM emulators, then all IBM emulators 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 emulators in the chain.

In general, the bus tag cable adapters must be placed immediately after the Unisys environment, but immediately before entrance to the printer environment. This organization is illustrated by Figure 6-2. 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.

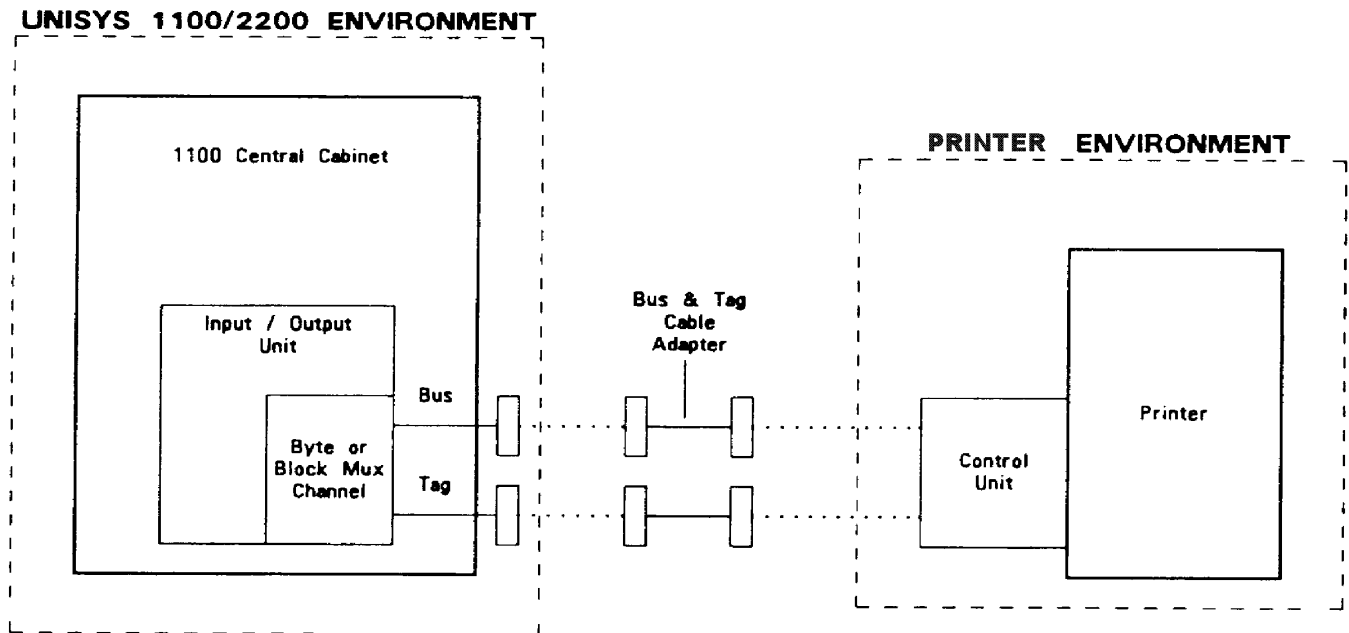


Figure 6-2 The Basic 1100/2200 to Siemens Connection

6.3 The Unisys to IBM 3800 Environment Connection

This section discusses various considerations for the connection of the Unisys host to the IBM 3800 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.

6.3.1 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.

6.3.2 Connection Via a Unisys Channel Transfer Switch

In the case where a Unisys Channel Transfer (CTS) switch 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 CTS, while the other end connects directly to the printer's control unit.

6.3.3 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.

6.3.4 Connection Via a Channel Extender

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

6.3.5 Considerations of Multiple Host / Device Connections

In some cases, multiple hosts may wish to share one or more printers. If a Unisys Channel Transfer Switch (CTS) is used to share the printer(s), then a bus and tag cable adapter is needed for each 3800 printer which is connected to the CTS. In this case, the bus and tag cable adapter(s) would be placed directly after the cable leading from the CTS, 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 Central to control the 3800 printer(s). The bus and tag cable adapter(s) would be placed in the same location as discussed in Section 6.3.2.

NOTE: If your site daisy chains Unisys devices with IBM devices, 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 adapter must be placed between the Unisys devices and the IBM 3800 devices in the chain.

7 User Programming with SPIN-X Central

This chapter contains information the programmer may need to utilize SPIN-X Central with the Siemens printers. Information is provided on using the @SYM command, generating Unisys 061 print controls, and Central error messages that may appear in the printed output.

7.1 Using the @SYM Command to Specify Print Formats

Print formats are controlled on the Unisys 1100/2200 by configuring each format name as an EXEC queue. This allows an 1100/2200 user to select a desired print format by specifying the corresponding queue in the third field of the @SYM command. The following example prints one copy of the file, DATAFILE, using the format SPINX1.

```
@SYM,U DATAFILE.,1,SPINX1
```

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

7.2 Specifying Fonts for Printfiles

Central supports printfiles that contain operations for fonts. These operations are contained within the OS2200 Load Character Arrangement (LCA) 061 print function. The LCA records are written to the printfile via the Executive Request SYMB\$. Central reads an LCA from the printfile and downloads the font requested. The LCA specifies a Translate Table Number (TTN) for identifying the font within this printfile. As mentioned, the LCA specifies a font for downloading, but not the font for a print image. The font for a print image is specified by TTN when the image is written via ER SYMB\$. A maximum of sixty-four TTN's may be active for a printfile at any given moment.

Central supports printfiles that do not contain LCA print records. The SPIN-X processor FDP specifies fonts to Central for printfiles that do not contain LCA's. The FDP Format command's Report_Font= keyword allows up to fifteen fonts for downloading. The TTN for each font is specified by its position in the REPORT_FONT keyword. The TTN for a print image in a printfile is specified via ER SYMB\$. For printfiles containing LCA's, REPORT_FONT=NONE should be specified for the Format.

Printfiles that do not contain LCA print functions that only require one font may utilize a Format that specifies the font in the REPORT_FONT keyword. The Bypass_Font_TTN=Yes keyword should be specified to ensure no undesired font selection occurs by Central based upon the print image's TTN. This option may be particularly useful for printfile's that contain ECL or compiler listings.

Printfiles that use the Format Bypass_Font_TTN=Yes keyword included may switch fonts by using Siemens Escape Codes. This is an alternative to specifying the TTN with ER SYMB\$. See section 7.6 for more information regarding Siemens Escape Codes.

Printer throughput can be improved by reducing the number of font downloads required. SPIN-X Central does not re-download a font that has previously been downloaded with the same TTN until the printer is re-initialized by the SPIN-X "I" or "P" keyin. The font is not re-downloaded regardless if it's specified with LCA records, REPORT_FONT keyword, or both. A site may choose to by convention pre-assign TTN's to fonts to decrease the number of font downloads required between printfiles. Also, any fonts in the REPORT_FONT= keyword should be positioned to correspond to the assigned TTN.

Printfiles should avoid LCA's that specify fonts that are not needed by the printfile; this results in unnecessary font downloads. Also, REPORT_FONT keywords should avoid specifying fonts that are not needed by the Format's printfiles for the same reason.

7.3 Using Unisys 1100/2200 Print Control Images

Central supports the standard Unisys 060 print control records, and a subset of the OS1100 061 print control records. See Appendix E of this guide for more information on the 061 print control functions Central supports. Refer to the *Unisys Executive Requests Programming Reference Manual* for details on employing ER SYMB\$ with the SM\$ and CN\$ functions to output 060 and 061 print control records to a printfile.

7.4 Print Output Error Messages

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 Central recognizes. The printing of the file abnormally terminated.

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

The "*n*" is the page number and the "*x*" is the entire 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 the Central "W" processor call option has not been specified; see Section 4.1.4 of the *SPIN-X Central Installation Guide* for more information on Central processor call options.

000434: THIS FILE IS EMPTY.

The file has no granules assigned to it.

**000444: SUBSTITUTE FONT NOT DEFINED FOR 061 RECORD,
SUBTYPE 02, font *x1/x2/x3***

x1: The diskette number from the 061 operation record
x2: The diskette font number from the 061 operation
x3: The format used to process this printfile

The above Central error message is printed if no substitute font is defined to FDP for the ER SYMB\$, CN\$ function Load Character Arrangement (Subtype 2, Operation 0).

**000444: SUBSTITUTE FONT NOT DEFINED FOR 061 RECORD,
SUBTYPE 02, font *x1/x2***

x1: The element name from the 061 operation record
x2: The format used to process this printfile

The above Central error message is printed if no substitute font is defined to FDP for the ER SYMB\$, CN\$ function Load Character Arrangement (Subtype 2, Operation 1).

000448: FONT SPECIFICATION X OUT OF BOUNDS.

The translate table field for an image is out of bounds 0 - 63.

000464: INVALID COPY REFERENCE NUMBER X IN 061 CNEF\$ RECORD

The copy reference number field of a Select Electronic Form print record, subtype 05, contained an invalid copy reference number.

000466: FOB *x* CANNOT BE ACCESSED

The named FOB could not be obtained from SYS\$PRINTER\$*SPIN-X.

000467: FOB *x* RECORD GREATER THAN MAXIMUM IMAGE SUPPORTED

A record of FOB *x* is greater than the maximum record length supported by Central. The FOB should be rebuilt.

7.5 Siemens Bar Codes With SPIN-X/Central

This section provides information needed by the programmer to print Siemens Nixdorf bar codes on the 2140/2090 using Central. Section 7.5.1 specifies how data may be transparently transferred to the printer from a user's application. Section 7.5.2 identifies the Siemens bar codes currently available with Central. Section 7.5.3 specifies how data should be coded for Central using the Siemens Nixdorf ZIP4 POSTNET font.

7.5.1 Transparent Data Mode

Central usually performs an ASCII to EBCDIC translation for user data sent to the printer. If a user data image is of character set type = 077 (63 decimal), then Central bypasses its ASCII to EBCDIC translation, and the image is sent untranslated to the printer. The character set type for the image is specified when the image is written via ER SYMB\$. The syntax for 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.

7.5.2 Siemens Nixdorf Bar Codes

Coding information for bar code fonts may be obtained from the *Siemens Laser Printing Systems, Bar Code Printing, Description manual* (U2495-J-Z49-4-7600). The following optional bar code fonts are supplied by GSURF for Central:

129E 137E 150E 153E 154E
163E 164E 165E 167E 168E
169E 170E 171E 172E 173E

These fonts require binary code sequences to be sent to the printer. Central provides a transparent mode for transferring user's binary data to the printer that bypasses ASCII to EBCDIC translation. See section 7.5.1 for information on transparent mode.

7.5.3 Siemens Nixdorf POSTNET ZIP4 Font

This section describes the format of the application's data needed to print the POSTNET barcodes on the Siemens electronic printer. Consult the *United States Postal Service, Automation Plan for Business Mailers, Publication 67*, for specific requirements for POSTNET. This USPS publication documents the logic requirements for POSTNET. The reader needs to understand information in the *Automation Plan for Business Mailers* to implement a POSTNET application correctly. The remainder of this section provides information for using the Siemens Nixdorf ZIP4 POSTNET font with Central.

A single byte is used to precede the zipcode in the programmer's application, and a different single byte is used to follow the application's zipcode. Each of these bytes causes a single bar to be printed on the page. The application must provide a **two byte** sequence to print the five bars required to represent a single zip code digit or correction digit. The following table gives the required byte sequence(s) to print the desired barcode:

POSTNET Value	Byte(s)	
	Octal	Decimal
-----	-----	-----
Start Byte	074	60
Zero	060260	24752
One	061261	25265
Two	062262	25778
Three	063263	26291
Four	064264	26804
Five	065265	27317
Six	066266	27830
Seven	067267	28343
Eight	070270	28856
Nine	071271	29369
End Byte	076	62

Table 7-1 POSTNET ZIP4 Byte Selection Sequences

NOTE: If a site provides their own translate table for ASCII to EBCDIC translation using the Central user exit capability, changing the translation for values 0260 - 0271 will invalidate the above table.

NOTE: The SPIN-X System Administrator must include an FDP SUBST command for each font referenced in a user's printfile. This SUBST command should match the font name specified in a printfile's 061 LCA print control. For example, the following SUBST command could be used for a ZIP4 font:

```
SUBST NAME=S1_ZIP4, TYPE=SIEMENS_EPS, FONT=ZIP4, ELEMENT=ZIP4, Default=Y
```

NOTE: See Appendix E for further information on 061 print controls with Central.

7.6 Example ER SYMB\$ Packets for Writing Images

The Exec Request SYMB\$ writes images to printfiles that control printing. This section depicts four SYMB\$ packets that write various images with specific requirements. Section 7.6.1 depicts a SYMB\$ packet for writing 061 print control records to the printfile. Section 7.6.2 depicts three SYMB\$ packets for writing print images with differing output requirements. Section 7.6.3 provides information on using ER SYMB\$ from languages other than MASM.

Each call to ER SYMB\$ writes one and only one image to the printfile.

The example packets in this section, show user supplied fields in *italics*.

7.6.1 Example ER SYMB\$ Packet for 061 Print Record

The following example SYMB\$ packet is for writing a 061 print function to a printfile:

0	<i>@use name (Fielddata)</i>		
1	<i>function CN\$</i>		<i>mode ASCII\$</i>
2	<i>status</i>	<i>io status</i>	
3	<i>word count</i>		<i>image address</i>
4			
5			
6			
7			
8			
9			

The 061 print function would start at *image address*. For example, *image address* could contain an LCA operation 0 image as illustrated in section E.2.1.1.

7.6.2 Examples of ER SYMB\$ Packets for User Print Images

This section contains various example SYMB\$ packet layouts for writing a user print image to the printfile.

The following basic SYMB\$ packet could be used for writing an ASCII image using the font with translate table number (TTN) 1:

0	<i>@use name (Fielddata)</i>		
1	<i>function W\$</i>		<i>mode ASCII\$</i>
2	status	io status	
3	<i>character count</i>		<i>image address</i>
4			
5			
6	<i>spacing</i>		
7			
8			
9			

The following SYMB\$ packet layout would be used for writing an ASCII image using the font specified by TTN:

0	<i>@use name (Fielddata)</i>		
1	<i>function W\$</i>		<i>mode SPEC\$</i>
2	status	io status	
3	<i>character count</i>		<i>image address</i>
4			
5	<i>spacing</i>		
6			
7			
8			
9			

The following SYMB\$ packet layout would be used for writing a binary image using the font specified by TTN:

0	<i>@use name (Fielddata)</i>		
1	<i>function W\$</i>		<i>mode SPEC\$</i>
2	status	io status	
3	<i>character count</i>		<i>image address</i>
4	<i>ttn</i>	<i>077</i>	
5	spacing		
6			
7			
8			
9			

This packet specifies transparent data mode to SPIN-X Central. See section 7.5.1 for more information.

7.6.3 ER SYMB\$ for Languages other than MASM

The following chart provides the numeric values for the constants used in the packet layouts:

Constant Name	Octal Value	Decimal Value
CN\$	055	45
W\$	010	8
ASCIIS\$	1	1
SPEC\$	020	16

The Unisys compilers provide an interface for performing the Executive Request SYMB\$ so that programmers can create printfiles at a discrete level. For example, ACOB provides the subprogram 'ERSYMB\$' for interfacing with SYMB\$. See the Unisys *ASCII COBOL, Programming Reference Manual*, section 18.8.2, for information on 'ERSYMB\$'. UCOB provides the UCS\$GENERAL interface. See the Unisys *OS2200 Exec System Software Executive Requests, Programming Reference Manual*, section 14.2 for complete documentation on ER SYMB\$.

7.7 Alternative Font Switching Using Siemens Escape Codes

The application programmer, as an alternative, may specify Siemens escape codes to switch fonts rather than vary the translate table field with ER SYMB\$. Font switching may occur between print lines or within print lines by the programmer issuing the proper escape code. The programmer must be aware of the order in which the fonts were specified by the SPIN-X Central administrator on the Report_Fonts= parameter of the Format command, or the translate table numbers specified in the Load Character Arrangement 061 print controls, so that the proper escape code for the desired font can be specified.

The following two byte escape codes invoke the desired font:

Font	Escape Code	Escape Code
	Octal	Decimal
----	-----	-----
Zero	377100	130624
One	377101	130625
Two	377102	130626
Three	377103	130627
Four	377104	130628
Five	377105	130629
Six	377106	130630
Seven	377107	130631
Eight	377110	130632
Nine	377111	130633
Ten	377112	130634
Eleven	377113	130635
Twelve	377114	130636
Thirteen	377115	130637
Fourteen	377116	130638
Fifteen	377117	130639
Sixteen	377120	130640
.	.	.
.	.	.
.	.	.
Sixty-two	377176	130686
Sixty-three	377177	130687

Table 7-2 Font Selection Escape Codes

Escape codes may appear on lines by themselves, or preceding the data for a line, or embedded within a line to permit multiple font switching within the same line.

If the programmer decides to perform font switching employing Siemens escape codes, then the Bypass_Font_TTN=Yes parameter should be specified for the Format. This parameter will cause Central to ignore the translate table specification of ER SYMB\$ for the user data images. Otherwise, Central will switch the font based upon the ER SYMB\$ translate table specification.

NOTE: The programmer must be sure the Siemens electronic printer has sufficient matrix memory configured to support the number of fonts used. The 2090 and 2140 printers standardly permit 46 simultaneous fonts.

A user exit is a site developed subroutine that is collected into Central to replace a predefined function of Central. This chapter defines the entry points that Central 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 Central 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 Central processor if desired. Section 8.1 describes each Central 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 Central with the user exits.

8.1 Central Defined User Exits

This section supplies the Central information needed to write user exit routines. It also defines conventions that should be followed in order for the exits (and Central) to function properly. Central 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 Central data structures which are relevant to user exits are described in Section 8.2.

- Central runs in basic mode.
- Central checks for the inclusion of a user exit by testing its predefined entry point to check its address. If the entry point is zero, Central jumps to its default subroutine for processing. If the entry point is non-zero, Central 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 Central checks.
- User exits should not cause dynamic memory expansion, e.g. ER MCORE\$, because Central runs at real-time and cannot be swapped to obtain memory.
- User exit programmers need to consider writing re-entrant code because Central may be used to drive multiple printers simultaneously. Central is already registered for test and set queuing.
- Central 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 Central between user exit calls. The register contents are not preserved by Central between processing separate SMOQUE entries. The user exits are entered in third word mode.

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

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

- Various Central data structures are passed by address to the user exit routine. Most of the fields should not be altered because that may abnormally impact Central'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 Central data structures are provided in the *SRO program file. When moving to a new release of Central 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. Central 4R4E was assembled with MASM level 6R1F.
- Two 64 word data areas are allocated in Central 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 Central 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, Central 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 Central 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 Central at a time. In most cases, Central 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 Central for transfer to the printer.

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

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

The major register set is available to the user exit except R15 should never be changed.

8.1.1 Banner Page

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 Central. By default, Central generates a banner page designed by Georgia State University. If this banner page is not desired, then you have the following options:

1. Modify GSU's banner page to your satisfaction.
2. Write the UXRBANPG user exit to design your own banner page.
3. Provide the O option on the Central processor call (within SYSSLIB\$*RUN\$.SPIN-X) to suppress the printing of the banner page altogether.

NOTE: The source to the banner page user exit routine that GSU uses is included on the Central Release Tape for the 1100/2200, and may be used as an example user exit. However, the entry point name, FHRBANPG, must be changed to UXRBANPG.

- **Environment / Point of Processing**

The banner page user exit routine is invoked before the printing of a print file. At this point of processing, the SMOQUE entry for the print request has been obtained, the file has been assigned, however no file reading has been performed.

- **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).
 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 Central:**

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. Central 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 print file.
 A14= (Not used with Siemens printers) Should always contain 0.

- **Programming Considerations**

If the banner page user exit is supplied, then Central will jump to UXRBANPG for each buffer of images to be sent to the printer for the banner page. The Image Builder Buffer (in the FCT) is 200 octal words long, and must be used by this exit to send any image to the printer. The programmer must establish a counter (A9) to sequence and control the processing of the banner page user exit. Because A9=0 the first time UXRBANPG is called, the programmer will know when this user exit is initially called.

8.1.2 Trailer Page

User Exit Routine Name: UXRTRAPG

- **Description**

The trailer page user exit allows a site to create its own page to follow each file printed by Central. By default, Central generates a trailer page designed by Georgia State University. The source to the default trailer page is supplied in the *SRO file that was copied during installation from the Central 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 Central.

- **Environment / Point of Processing**

The trailer page user exit routine is invoked after the printing 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: UXRTRAPG**

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

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

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 UXRTRAPG is called).

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

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. Central 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 is not called again for the current print file.

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

- **Programming Considerations**

If the trailer page exit is supplied, Central jumps to UXRTRAPG for each buffer of images to be sent to the printer for the trailer page. The Image Builder Buffer is 200 octal words long, and must be used by this exit to send any image to be printed. The programmer must establish a counter to sequence and control the processing of the trailer page exit. Because A9=0 the first time UXRTRAPG is called, the programmer will know when this user exit is initially called.

8.1.3 Type 14 Log Entry

User Exit Routine Name: UXRLOG14

- **Description**

The type 14 log entry user exit may be used to generate a symbiont termination log entry (type 14) which is different than the one which Central generates by default. Refer to Appendix A for a description of the format that Central uses for the symbiont termination log entry. This user exit may also be used to control the generation of the ASCII end of processing log entry 10112, subtype 2.

- **Environment / Point of Processing**

The type 14 log entry user exit routine is invoked after the printing of a print file (or trailer page if supplied) and after any ending print controls. At this point of processing, the SMOQUE entry for the print request is still available and the file has been @FREE'd.

- **Register contents when control is passed to the user exit routine: UXRLOG14**

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

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

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

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

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

There are no return codes necessary for this exit (because it is only called once for each print file).

- **Programming Considerations**

The type 14 log entry user exit is only called once, not repetitively like other user exits, therefore there is no need for a counter for sequencing and control.

8.1.4 Central Data Structures Passed to User Exits

This section provides the information about each Central data structure that is provided to write user exits. Each Central 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 Central 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 Central (this file is usually SPIN-X*SRO). The omnibus elements within this program file have been compiled using MASM definition mode assembly (\$DEF).

The Central 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 Central 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 Central for the life of the program execution. FCTSITE is null filled each time a device is initiated by the Central "I" or "P" keyins.

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

8.1.4.1 ASCII Edit Packet (AED)

AED provides \$EQUFs to reference the fields in the SYSLIB AEDIT\$ packets utilized by Central. These EQUFs are utilized by the Central 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 the AEDIT\$ packet layout. If the SYSLIB ASCII Edit routines are used, then use either the FCTSITE or PCBSITE area for the actual packet and use the above tags as displacements within.

8.1.4.2 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	R
FCTPITIM	The time of print initiation (Also in DATE\$ format)	1	R
FCTSITE	An area for the site's programmer use	64	A
FCTIBBUF	A buffer which is used to pass return data to Central (Image Builder Buffer in FCT)	128	A
FCTLBL50	The 050 printfile's header label control record		

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.

8.1.4.3 Printer Control Buffer (PCB)

Tag Name	Description	Length (words)	Type
PCBDEVID	The device ID (6 ASCII Characters)	1/4	R
PCBDVTPE	The device type (040: Siemens IBM 3800 printer emulator)	1/2	R
PCBPROPT	The Central 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

8.1.4.4 SMOQUE Entry (SQE)

SQE provides \$EQUFs to reference the fields of the ER SMOQUE\$ entry. These EQUFs are utilized by the Central 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
SQEQUEID	The Queue-id from the @SYM command	2	R
SQEOUTID	The Output-id of Central	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
SQEFYCL	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 format.

8.2 Modifying the Character Translation Table User Exit

This section provides the information needed to modify the character translation table which Central uses before sending data to the printer. The ASCII to EBCDIC translation is bypassed for the character code type = 077 (63 decimal).

Translation Table Name: UXASTEB

Description

The ASCII to EBCDIC character translation table allows the SPIN-X Central 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 @MAPEQU statement associated with UXASTEB within the @MAP statements, and collect Central as discussed in Section 7.3.

See Appendix I for a chart of Central's ASCII to EBCDIC character translation.

8.3 Collecting Central with User Exits

This section explains the procedure for linking Central 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 Central 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.

Central uses the value of the user exit tags to determine whether or not a particular user exit is being used (Central 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 Central assumes that no user exit has been provided; however, if the tag is not equal to 0 then Central 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 Central assumes that the user exit routine has not been supplied. If the EQU has been removed and the user exit provided then Central 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 Central absolute is ready for testing. Copy the new Central 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 Central environment is being utilized. See Chapter 9 for more information on setting up a test Central.

5. After the new Central absolute has been satisfactorily tested, you may choose to remove the " ." (comment) from the BANKINFO statement associated with UEXIT\$IBANK and re-collect Central; 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 Central with UEXIT\$IBANK marked READONLY.

See Appendix B for a listing of the *SRO.MAP element.

Sites may choose to run a separate Central job from the production Central 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 Central runs, the user exit programmer could test the new Central on one printer while the production Central continues driving another.

9.1 Steps for Setting-up a Test Central

Four steps are involved in setting-up a test Central:

1. Install the Central software to be tested under a different qualifier than the production environment.

Rather than copy Central 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 Central 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 Centrals cannot have the same printer assigned simultaneously. A printer normally under the production Central 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 Central 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 Central was installed under in Step 1 above, e.g. SPIN-X4R4. An example of ECL for a test Central batch job is shown in Section 9.2.

4. A keyin prefix other than L* should be selected for the test Central job.

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

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

Central suffixes an "*" to the value specified on the processor call therefore the Central 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 Central based upon the above example follows:

```
P* LY I
```

9.2 Example ECL for Test Central 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 P
@PMD,PALBE
@FIN
```

A.1 Fieldata Log Entries

This section describes the contents of the symbiont log entries that Central 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 the printer 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	account number					
6						
7	userid					
8						
9	filename					
10						

Table A-1 Description of the 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			

Table A-1 Description of the End of Processing Log Table Entries

Definitions of Terms for Table A-1

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

filename	The (fielddata) filename of the file which was symmed.
qualifier	The (fielddata) qualifier of the file which was symmed.
num-of-pages-reprinted	The total number of pages reprinted (if any) for the file.
num-of-pages-skipped	The total number of pages skipped (if any) for the file.
type-of-termination	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.
project-id	Contains the (fielddata) project-id of the run that @SYM'ed the file.
symbiont-queue	The (fielddata) name of the symbiont queue that the file was in when it was printed.
date-and-time-of-log-entry	The (TDATE\$ format) date and time of the log entry.
EXEC 8	The run-id of the EXEC

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

When the printer 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	unit status table		file type	not used		
2	symbiont name					
3	reserved					
4	run-id					
5 6	account number					
7 8	userid					
9 10	filename					
11 12	qualifier					

Table A-2 Description of the Start of Processing Log Table Entries
(continued on following page)

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

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

Definitions of Terms for Table A-2

entry-type	The type of log entry
num-words-in-entry	A count of words used in the entry
system-indicator	A indicator used by log editing programs to distinguish changes in log file formats produced by different levels of the EXEC
num-log-entries	Used only in the first entry of a 224-word block
unit-status-table	The Unit Status Table (UST) symbiont equipment code
file-type	Indicates the type of file
symbiont-name	The (fielddata) symbiont name used to assign the printer
run-id	The (fielddata) run-id of the run which symmed the file
account-number	The (fielddata) account number of the run which symmed the file
userid	The (fielddata) user-id of the run which symmed the file
filename	The (fielddata) filename of the file which was symmed
qualifier	The (fielddata) qualifier of the file which was symmed
project-id	Contains the (fielddata) project-id of the run that symmed the file
symbiont-queue	The (fielddata) name of the symbiont queue that the file was in when was printed
date-and-time-of-log-entry	The (TDATE\$ format) date and time of the log entry
EXEC 8	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 Central. There are two formats of this log entry. Log entries of format 2 indicate an internal error in Central. All format 2 logs should be forwarded to SPIN-X Support at the address given in Section 1.4 of the *SPIN-X Central Installation Guide*.

The log entry of 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 numlog entries	identification number	
2	error type	num of sense bytes	reserved		reserved	
3	name					
4	date and time of error					
5	software fctt-value	software cnd-code	reserved		number of retry-attempts	
6	ADH Status	EXEC level	hardware sys-type	Central error number		
7	number of completed I/O's					
8	failing CCW's					
9	failing CSW's					
10	CSW's					
11						
12						

Table A-3 Description of the Symbiont Handler Log Entries
(continued on next page)

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

Table A-3 Description of the Symbiont Handler Log Entries
(Continued from previous page)

Definitions of Terms for Table A-3

entry-type	The type of log entry.
num-words-in-entry	A count of words used in the entry
system-indicator	An indicator used by log editing programs to distinguish changes in log file formats produced by different levels of the EXEC.
num-log-entries	Used only in the first entry of a 224-word block.
entry-format	The format of the remainder of the entry: 1 indicates that this log entry specifies information relating to an I/O error. 2 indicates that this log entry contains device error log information.
level indicator	The level of Central that produced the log entry.
relative entry-num	Entry number relative to a chain of related entries.
total-num-log-entries	Total number of log entries that are related to this one.
identification number	A unique identification number supplied by Central in order to distinguish one set of related log entries from another set.
error-type	The type of error represented by this log entry: 0 implies a normal error condition, such as a unit check or subchannel status. 1 implies a nonattention interrupt while waiting for an attention interrupt.

num-of-sense-bytes	The number of valid sense bytes that are in words 13 and 17 of the log entry.
name	The (fielddata) name used to assign the device.
date-and-time-of-error	is the (TDATE\$ format) date and time that Central initially detected the error.
software-funct-value	The software function value that was used in the ADH packet.
software-cnd-code	The value of the software condition code which was returned within the ADH packet.
number-of-retry-attempts	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.
ADH-status	The status returned in the ADH packet.
EXEC-level	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.
hardware-sys-type	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.
Central error-number	The Central error number minus 500 of the operator message that was sent to the system console. If no message was sent, this value is zero. See the <i>SPIN-X Central Operator Guide</i> for further information.
number-completed I/O's	Indicates the number of completed I/O's before the last error was logged.
failing CCW's	Contains the failing Channel Command Words (CCW's) 0 and 1, respectively.
CSW's	Contains the Channel Status Words (CSW's) 0, 1 and 2, respectively.
sense-bytes	Contains sense bytes 0 through 5. The remaining positions are nulls.
run-id	The (fielddata) run-id of the run that symmed the file which is in progress.
console-response	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.
symbiont-equip-code	The symbiont equipment code defined by Central: 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.
CCW command-bytes	Contains the last eight command bytes (left justified) that were sent in the CCW list.

handler-id	The Central handler ID.
date-and-time-of-log-entry	The (TDATE\$ format) date and time that the log entry was made.
run-id	The run-id of Central.
EXEC 8	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 Central. For example, the EXEC generates log entry type 109 when it starts processing a printfile; correspondingly, Central generates log entry type 10112.

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

When the printer 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			

Table A-4 Description of the Start of Processing Log Table Entries
(continued on next page)

17 18 19	qualifier
20 21	queue name
22 23 24	concatenated filename
25 26 27	concatenated qualifier
28	physical page number

Table A-4 Description of the Start of Processing Log Table Entries
(continued from previous page)

Definitions of Terms for Table A-4

equipment-code	The symbiont equipment code from the EXEC Unit Status Table. For example, the equipment code for a 0770 printer is octal 056.
file-type	Indicates the type of file
device-name	The name of the print device
run-id	The run-id that queued the print file
account number	The account number of the run that queued the print file.
user-id	The userid of the run that queued the print
project-id	The project-id of the run that queued the print
filename	The name of the file that was queued
qualifier	The qualifier of the file that was queued
queue-name	The name of the queue the file was printed from
concatenated-filename	Unused by Central, always spaces
concatenated-qualifier	Unused by Central, always spaces
physical-page-number	Unused by Central, always zeroes

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

When the printer 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			

Table A-5 Description of the 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

Table A-5 Description of the End of Processing Log Table Entries
(continued from previous page)

Definitions of Terms for Table A-5

equipment-code	The symbiont equipment code from the EXEC Unit Status Table. For example, the equipment code for a 0770 printer is octal 056.
file-type	Indicates the type of file
total-page-count	The total number of pages printed for the file
device-name	The name of the print device
line-count	The number of lines printed for the file
run-id	The run-id that queued the print file
account number	The account number of the run that queued the print file.
user-id	The userid of the run that queued the print
project-id	The project-id of the run that queued the print
filename	The name of the file that was queued
qualifier	The qualifier of the file that was queued
queue-name	The name of the queue the file was printed from
concatenated-filename	Unused by Central, always spaces
concatenated-qualifier	Unused by Central, always spaces
physical-page-number	Unused by Central, always zero
reprint-count	The total number of pages reprinted (if any) for the file
skip-count	The total number of pages skipped (if any) for the file
termination-type	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 errors 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						
48	error log bytes - 3					
49						
.						
67						

Table A-6 Description of the Symbiont Device Error Log Table Entries

Definitions of Terms for Table A-6

handler-level	The level of the printer handler that created this log entry. This will always be binary 1 for Central.
number of entries	The number of log entries
device name	The name of the print device
equipment code	The symbiont equipment code from the EXEC Unit Status Table, for example, the equipment code for a 0770 printer is octal 56
error log byte count	The number of error bytes in the log entry
error log bytes 1	The number of bytes in section 1
error log bytes 2	The number of bytes in section 2
error log bytes 3	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 errors 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				
3	time of error			
4	ADH function	software condition code	reserved	retry count
5	ADHstatus	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)			

Table A-7 Description of the Symbiont I/O Fault Error Log Table Entries
(continued on next page)

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	message response	reserved	equipment code
20	command			
21	byte			

Table A-7 Description of the Symbiont I/O Fault Error Log Table Entries
(continued from previous page)

Definitions of Terms for Table A-7

handler-level	The level of the printer handler that created this log entry. This will always be binary 1 for Central.
error-type	The type of error represented by this log entry: 0 -implies a normal error condition, such as a unit check or subchannel status. 1 -implies a nonattention interrupt while waiting for an attention interrupt.
sense-byte-count	The number of valid sense bytes stored in the log entry
device-name	The name of the print device that the error occurred on
time-of-error	The (TDATE\$) format date and time that Central initially detected the error
ADH function	The software function value that was used in the ADH packet
software-condition-code	The value of the software condition code which was returned within the ADH packet
retry-count	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.
ADH-status	The status returned in the ADH packet
machine-type	A number indicating the Unisys hardware system type: 1-indicates an 1100/80 2-indicates an 1100/60, /70, /90, 2200
reference-since-error	The number of successful I/O's that have occurred since the last error or since the device was last initialized.
failing CCW's	Contains the failing Channel Command Words (CCW's) 0 and 1, respectively
CSW's	Contains the Channel Status Words (CSW's) 0, 1 and 2 respectively
sense-bytes	Contains sense bytes 0 through 19, the remaining positions are null
file-run-id	The run-id of the run that symmed the file which is in progress
command-byte	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

Merging Font Input Files

If you order a non-standard font(s) for your Siemens printer, it will be contained on a separate tape from the 1100/2200 Release Tape. The SPIN-X Central administrator should copy the new font(s) to a temporary file on disk via @COPY,G. Then the new font(s) need to be merged with the already existing fonts contained in *FONTFILE; the following SORT/MERGE runstream will combine the new fonts with the old fonts:

```
@QUAL  qualifier  . QUALIFIER USED FOR FDP FILES
@ASG,TJ NEWFONTSTAPE,U9S,reel number,,NORING      . ASG TAPE CONTAINING NEW FONTS
@ASG,T  TEMPFONTS      .
@.ASG TEMPORARY FILE FOR FONTS
@COPY,G NEWFONTSTAPE.,TEMPFONTS.      . COPY NEW FONTS TO DISK
@ASG,A  *FONTFILE.      . @ASG CURRENT FONTFILE
@USE    FONTFILE,*FONTFILE      . LINK TO CURRENT *FONTFILE
@ASG,U  *FONTFILE(+1)      . CREATE NEW CYCLE OF *FONTFILE
@USE    NEWFONTFILE,*FONTFILE(+1)      . LINK TO NEW CYCLE OF *FONTFILE
@SORT,E      . SORT/MERGE OLD AND NEW FONTS
FILESIN=FONTFILE,TEMPFONTS  MODE=SDF
RSZ=13120,CHARACTERS
KEY=1,4,S,A LINKSZ=288,CHARACTERS
FILEOUT=NEWFONTFILE  MODE=SDF
@EOF
@FREE FONTFILE      . RELEASE ORIGINAL FONTFILE
@FREE NEWFONTFILE  . CATALOG MERGED FONTFILE
```

Now the new fonts may be referenced in FDP SUBST and FORMAT commands. See Chapter 3.

D Listing of the Distributed FDP Input File

FONTFILE *FONTFILE. . specify file containing the fonts from Central install

. The SUBST command specifies a font from FONTFILE as a substitute for
. a font in a 061 print control record.

```
SUBST NAME=S1_017M,TYPE=SIEMENS_EPS,FONT=017M,DISKETTE=(381700,0),Default=Y
SUBST NAME=S2_012M,TYPE=SIEMENS_EPS,FONT=012M,DISKETTE=(381700,0)
SUBST NAME=S1_018M,TYPE=SIEMENS_EPS,FONT=018M,DISKETTE=(381700,1),Default=Y
SUBST NAME=S1_019M,TYPE=SIEMENS_EPS,FONT=019M,DISKETTE=(381700,2),Default=Y
SUBST NAME=S1_MJ15,TYPE=SIEMENS_EPS,FONT=MJ15,DISKETTE=(381700,5),Default=Y
SUBST NAME=S2_005M,TYPE=SIEMENS_EPS,FONT=005M,DISKETTE=(381700,5)
SUBST NAME=S1_034M,TYPE=SIEMENS_EPS,FONT=034M,DISKETTE=(381700,12),Default=Y
SUBST NAME=S1_035M,TYPE=SIEMENS_EPS,FONT=035M,DISKETTE=(381700,13),Default=Y
SUBST NAME=S1_036M,TYPE=SIEMENS_EPS,FONT=036M,DISKETTE=(381700,14),Default=Y
SUBST NAME=S1_004M,TYPE=SIEMENS_EPS,FONT=004M,DISKETTE=(381701,0),Default=Y
SUBST NAME=S2_004M,TYPE=SIEMENS_EPS,FONT=004M,DISKETTE=(381701,1),Default=Y
SUBST NAME=S1_OCRA,TYPE=SIEMENS_EPS,FONT=AOAW,DISKETTE=(381701,10),Default=Y
SUBST NAME=S2_034M,TYPE=SIEMENS_EPS,FONT=034M,DISKETTE=(381701,12),Default=Y
SUBST NAME=S2_035M,TYPE=SIEMENS_EPS,FONT=035M,DISKETTE=(381701,13),Default=Y
SUBST NAME=S2_036M,TYPE=SIEMENS_EPS,FONT=036M,DISKETTE=(381701,14),Default=Y
SUBST NAME=S3_036M,TYPE=SIEMENS_EPS,FONT=036M,ELEMENT= FONTELT123,Default=Y
```

. SAMPLE FORMAT COMMANDS FOR THE SIEMENS EPS FEATURE

```
FORMAT NAME=SPINX1,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=(003M,003M),TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=6
```

```
FORMAT NAME=SPINX2,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=(020M,020M),TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=8
```

```
FORMAT NAME=REPRT1,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=NONE,TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=6
  . font substitutes are from default=y substs
```

```
FORMAT NAME=REPRT2,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=NONE,TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=6,;
  SUBSTITUTE=(S2_012M) . use 012M instead of 017M from default=y subst
```

```
FORMAT NAME=REPRT3,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=NONE,TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=6,;
  SUBSTITUTE=(S2_012M,S2_005M) . use 012M and 005M not defaults
```

```
FORMAT NAME=SPINX3,TYPE=SIEMENS_EPS,;
  BANNER_FONT=004M,REPORT_FONT=(005M,020M,025M),TRAILER_FONT=004M,;
  LINES=66,LINES8=88,TOP=0,BOT=0,DENSITY=8
```

```
DEVICE NAME=SEPS,TYPE=SIEMENS_EPS,BANNER=UNISYS,TRAILER=UNISYS
```

E

061 Print Controls

This chapter provides graphical depictions, definitions and examples as well as technical information on Central's support of the 061 print control images for the Siemens EPS printers. The reader should be familiar with Central's FDP SUBST command and Format's Substitute= parameter; see Chapter 3 of this reference guide.

Central drives the Siemens EPS printers using compatibility mode with the Siemens High Performance extension for font and Forms Overlay Buffer (FOB) support.

E.1 The PLIST Utility

The PLIST processor can be used to display 061 print records from a printfile. This PLIST information can be used to create the appropriate SUBST and FORMAT commands to FDP. The following sections depict the 061 print controls Central supports and that PLIST displays.

E.2 Using 061 Images in Printfiles

This section discusses the following 061 CN\$ functions of ER SYMB\$: load character arrangement (Subtype 2), load vertical format buffer (Subtype 3), select electronic forms (Subtype 5), load copy number (Subtype 011), load flash number (Subtype 012), and load special forms (Subtype 010).

See Section 13.2 of the Unisys ER Programming Reference Manual (7830 7899) for information on using the CN\$ function of ER SYMB\$ to output 061 print control records to printfiles.

E.2.1 Accessing Fonts with the Load Character Arrangement (LCA)

Three types of 061 images are used for accessing fonts. These are the Subtype 2 Load Character Arrangement Operation 0 and Operation 1 and 2. These are discussed in sections E.2.1.1 and E.2.1.2. (Operation 2 is supported in the same manner as Operation 1 so everything in E.2.1.2 applies to Operation 2 as well). Each section consists of a graphical representation of the print control followed by definitions, examples and finally by technical considerations that are applicable to the LCA. For Operations 0, 1, and 2 the "translate table number" field attaches a font identifier to the downloaded font. The font number permits the printer to distinguish one font downloaded from other fonts downloaded. (This font number to the printer must not be confused with the font number in Operation 0 that identifies a font on a diskette.) LCA Operations 3, 4 and 5 of this image are bypassed.

The LCA image results in Central downloading a font from its database to the printer. The font specified by the LCA function is substituted by a font determined from information supplied to FDP in the SUBST and FORMAT commands. Currently, Central supports 64 simultaneously active fonts (0 - 63) on the printer.

The substitute font consists of a translate table and from 1 to 4 graphic character modifications (GCM). The translate table name identifies the translate table and the 1 to 4 GCMs.

If a font is specified in a LCA print control for which no substitute font has been defined to FDP, then the following error message is printed in the output and the printfile processing terminates at that point:

```
000444: Substitute font not defined for 061 record, Subtype 02, font x1/x2[/x3]
```

The values of the above variable items vary depending upon the operation. See the following subsections for information on a specific operation.

A FDP SUBST command should be specified for each LCA print control Operations 0, 1, or 2.

E.2.1.1 Substituting for a Font on Diskette

To make a font substitution, the SUBST command requires two pieces of information from the Load Character Arrangement Operation 0 061 image below: these are the diskette_id and the font number. When Central encounters an LCA with this diskette_id and font_number the Siemens font from the SUBST command is downloaded. The Siemens font is downloaded from the Central database.

Below is a graphical representation of a 061 image which specifies a font on a diskette. Each row is equivalent to one 36 bit word. This image would commonly be found at the beginning of a printfile.

Load Character Arrangement 0 Operation

02	operation 0	translate table number	font number
diskette i.d.			
diskette i.d. (continued)			

Figure E.1 Definitions of the Table Entries

Operation 0	This number identifies the type of LCA control image
Translate Table Number	This number, which must be in 0 - 63, identifies this font when it is downloaded to the printer. Within the printfile, whenever this font index occurs , this particular font will be selected .
Font Number	This number, which identifies the font on the diskette, must be a number in 0 - 15. NOTE: This "font number" field in this image should not be confused with the translate table number specified to the printer when a font is downloaded.
Diskette-id	This identifier, composed of 6 ASCII characters, identifies the diskette on which the original font resided.

Example:

```
SUBST NAME=S1_004M TYPE=SI EMENS_EPS, FONT=004M DISKETTE=(381701, 10), DEFAULT=N
```

NOTE: "DEFAULT=N" does not need to be specified - it is the default: for purposes of demonstration the field has been made explicit.

```
FORMAT NAME=FORM2, TYPE=SI EMENS_EPS, ;
SUBSTITUTES=(S1_004M)
```

If the LCA's diskette-id and font number are found in a SUBST command, the replacement font named in the "FONT=" parameter is downloaded. The SUBSTITUTES= parameter need only be specified if a needed SUBST is not specified with Default=Y.

LCA Operation 0 error message

The following error message is printed if no substitute font is defined for the diskette font specified:

```
000444: Substitute font not defined for 061 record, Subtype 02, font x1/x2/x3
x1: The diskette id from the 061 operation record
x2: The diskette font number from the 061 operation
x3: The FDP format used to process this printfile
```

E.2.1.2 Substituting for Element Fonts

To make a font substitution, the SUBST command requires one piece of information from the Load Character Arrangement Operation 1 061 image below: this is the element name field. When Central encounters an LCA with this element name, the Siemens font from the SUBST command is downloaded.

Below is a graphical representation of the 061 image which specifies a font to be downloaded from a program file on the Unisys host. Each row is equivalent to one 36 bit word. This image would commonly be found at the beginning of a printfile.

Load Character Arrangement Operation 1

02	operation 1	translate table number	unused
element name			
element name (continued)			
element name (continued)			

Figure E.2 Definitions of the Table Entries

Operation 1	This number identifies the type of control image
Translate Table Number	This number, which must be in 0 - 63, identifies this font when it is downloaded to the printer. Within the printfile, whenever this translate table is specified, this particular font will be selected. This translate number corresponds with the translate number specified with a user data image using ER SYMB\$ function SPEC\$.
Element Name	This specifies the element name that contained the original font.

Example:

```
SUBST NAME=S3_036M TYPE=SIEMENS_EPS, FONT=036M ELEMENT=FONTELT123, Default=Y
FORMAT NAME=FORM2, TYPE=SIEMENS_EPS
```

In the FORMAT command above, the "SUBSTITUTES=" field is empty, nevertheless the substitution of font 036M named in the SUBST command can still be made because the "DEFAULT=Y" field of the SUBST command has put this font in the default pool. Because the FORMAT command has not specified any substitutes, Central will look in the default pool for a replacement.

LCA Operation 1 error message

The following error message is printed if no substitute font is defined for the font element specified:

```
000444: Substitute font not defined for 061 record, Subtype 02, font x1/x2
```

x1: The element name from the 061 operation record

x2: The format used to process this printfile

E.2.1.3 LCA Technical Notes

Font Selection and Switching

Central selects the font for printing from those downloaded based upon the "translate table number" field (S5) of the PRINT\$ data record for each print data image (see section E.2.6.). This translate table number is the same number as the translate table number of a font downloaded based upon an LCA print control record. Different fonts are selected, or switched, between images by varying the value of the translate table number. By default, Fielddata PRINT\$ records frequently specify zero for the translate table number, and ASCII records specify one.

Central switches to the desired font based upon the translate table number if it is not currently active for the printfile. Central selects a font downloaded to the printer by sending a Siemens Type 1 Control Byte to the printer specifying the font number. For more information on Siemens Control bytes, see Chapter 3 of the Siemens Enhanced Mode Software, Programmer's Guide.

NOTE: If the FDP Format parameter Bypass_Font_61=Yes is provided, then Central will ignore font download specifications in LCA requests. The fonts for the report then should come from the Format Report_Font= parameter.

The Bypass_Font_61= parameter is independent of the Bypass_Font_TTN= parameter.

E.2.2 The Select Electronic Form (SEF) Print Control

Creating and manipulating FOBs is done independently of Central. For details of the processes involved see Chapter 4.

Below is a graphical representation of the 061 image which specifies a FOB to be downloaded from a program file on the host. Each row is equivalent to one 36 bit word.

Select Electronic Forms Command Format (Subtype 05)

05	operation	copy reference number	1
element name			
element name (continued)			
element name (continued)			

Figure E.3 Definitions of the Table Entries

05	This number identifies the subtype of the control image
Operation	This is the type of electronic forms operation to use 0 - only use electronic forms on the following base page. 1 - use this electronic form on all the following base pages until turned off or until the form is changed. 2 - turn off any prior electronic forms.
Copy Reference Number	This is the copy reference number for the FOB. The copy reference number must be in the range 1-255. This 061 print control downloads the FOB; the FOB is activated by a Load Copy Number 061 print control that specifies the FOB's copy reference number.
1	This is the number of copies for a page that is restricted to 1 by Central.
Element Name	This is a 12 - character ASCII name of an element in SYS\$PRINTER\$*SPIN-X that contains the FOB data.

E.2.2.1 SEF Technical Notes

The SPIN-X processor FOBCON accepts a FOB output from the SPFOB2DB processor, and writes the FOB as an element in SYS\$PRINTER\$*SPIN-X with the /FOB version name. Central can then obtain the FOB from SYS\$PRINTER\$*SPIN-X in support of an SEF print control. The element in SYS\$PRINTER\$*SPIN-X is expected to contain the version name /FOB; the /FOB is not specified in the SEF print control.

The FOB element may contain one or more electronic images. A specific image is activated by specifying a Load Copy Number command with the copy reference number that corresponds to that image.

When a subsequent FOB is downloaded, any previously downloaded FOB is erased. Any downloaded FOB is also erased at the end of the job.

The SEF print record results in a page eject if not already at the home paper position.

E.2.3 The Load Copy Number (LCN) 061 Image

This image is required to make a Forms Overlay Buffer active. Also this print image should immediately precede any Load Flash Number (LFN) print image.

Load Copy Number Command Format (Subtype 011)

011	operation	reserved	copy reference number
-----	-----------	----------	-----------------------

Figure E.4 Definitions of the Table Entries

011	This number identifies the subtype of this 061 print control.
Operation	This is the type of operation to perform.
	1 Use the copy number for all subsequent base pages until a new "load copy number" command is received.
Copy Reference Number	The copy reference number of a FOB or 1.

E.2.3.1 LCN Technical Notes

This image results in a Load Copy Number command that specifies the copy reference number for subsequent pages.

The LCN image results in a page eject if not already at home paper.

The LCN command must precede a Load Flash Number (LFN) command. The LCN command activates any downloaded FOB with the same copy reference number.

E.2.4 The Vertical Format Buffer (VFB)

Load Vertical Format Buffer (Subtype 03)

03	operation	space type	reserved (must be zero)
		reserved	
		reserved	
		reserved	
	page length	top margin	bottom margin
	line number	lines per inch	reserved
repeat of above word until end of record			

NOTE: If the last word shown is zero, then it must not be included in the ER SYMB\$ packet's word count.

Figure E.5 Definitions of the Table Entries

03	This number identifies the subtype of the control image
Operation	This field is used to determine how the VFB information is to be generated. 0 - This indicates that the VFB data is contained within words 4 - n. This field must be "0".
Space type	This indicates the spacing mode to be used. 0 - This indicates that normal spacing is to be used. This field must be "0".
Page Length	This field contains the page length of the VFB in lines or an alternate value. 01 - 172 The number of lines per page. Includes the top and bottom margins. 07776 Do not change the current page length. 07777 The device default is used for page size.
Top Margin	This is the size of the top margin. 0 - 171 The number of lines for the top margin 07776 Do not change the current top margin. 07777 The device default is used for the top margin
Bottom Margin	This is the size of the bottom margin. 0 - 171 The number of lines for the bottom margin 07776 Do not change the current bottom margin. 07777 The device default is used for the bottom margin.
Line Number	This is the line-number to which the lines-per-inch specification in T2 applies. This lines-per-inch density remains in effect for successive lines until the next line number (if any) in the VFB is specified.
Lines-per-inch	This is print the density. The following values can be specified. 6 - six lines per inch 8 - eight lines per inch 12 - twelve lines per inch

NOTE: The paper length needs to be set on the printer with the 1.0 Menu option. If the form length specified in the VFB print control does not match the printer's paper length, then a specification check error results.

E.2.5 The Load Flash Number (LFN) 061 Image

The LFN activates the diskette flash.

Load Flash Number Command Format (Subtype 012)

012	operation	reserved	0 or 1
		flash-id	
		flash-id	
		flash-id	

Figure E.6 Definitions of the Table Entries

012	This number identifies the subtype of the 061 control image
Operation	This determines the type of control desired. 0 - Only the base page is to be flashed. 1 - All subsequent pages are to be flashed until the next "load copy number" or "load flash number" command.
0 or 1	This is the number of copies. Currently this is limited to one. A zero turns off the flashing.
Flash - id	This is the name of the EFO to use for flashing. The name is in ASCII. If this field is present, an operator message 000454 is sent to the console for the operator to mount the EFO in the printer. The absence of this field implies that the EFO is already mounted.

E.2.5.1 LFN Technical Notes

If a flash name is identified in the LFN image, then the following message is posted on the console:

```
000454: MOUNT EFO: x A
```

The diskette containing flash "x" should be mounted and selected on 1.3 of the printer Menu. Then, the 000454 message should be answered "A".

The LFN results in a page eject if not already at home paper. This image should be preceded by a Load Copy Number print control.

If an LFN image is encountered at the home paper position it takes effect for that page.

E.2.6 The Load Special Forms (Subtype 010) 061 Image

A message is written to the operator console when a special forms 061 print control is encountered in a printfile. The format of the print control is shown below.

010	reserved
	form-id
	form-id
	form-id

The **form-id** fields specifies the name that was given by this site to the special form. It can be made up of one, two or three words of ASCII data. If the last word or the last two words are not included, then they are assumed to be spaces.

If the paper is not already at top of form, then it will go there when this print control is encountered in the printfile.

E.2.7 PRINT\$ Data Records

Below is the layout for a user data image. The translate table number (S5) specifies the font for printing the data image; Central switches to this font if it is not already active.

Print\$ Data Record Format

data count	space count	translate table number	code type
	data word 0		
	data word 1		
	.		
	.		
	.		
	data word n		

Figure E.7 Definitions of the Table Entries

The user image, which begins with data word 0, is preceded by one word of information that is identified below.

Data Count	This field gives the number of words in the following user image. The Central limit is 128 words or 512 bytes.
Space Count	This field gives the number of lines to space down the page before printing the image.
Translate Table Number	This identifies the font to use for printing this image. It is specified by the translate table number field for this font in the 061 LCA print control.
Code Type	This should be either ASCII or Fielddata or binary mode <ul style="list-style-type: none"> 0 - Fielddata 1 - ASCII 077 - transparent mode; the user data is transferred to the printer without any character translation. The data should consist of 8-bit bytes with each byte contained within the normal 9-bit field. If any byte has its 9th bit set, then the data transfer to the printer for this image stops at that point.

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

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 Central 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 two utility programs provided with the SPIN-X Central package which are not required for day to day operation but can be useful in performing certain Central 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 Central 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 it 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.

G.2 SMOQUETEST

This utility is used to test SMOQUE\$ compatibility relative to Central and a site's current OS1100 EXEC level. Sites using later releases will have no use for this utility.

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 Central processor call. This information provides an alternative to searching the system logfile to obtain a listing of 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 ASCII to EBCDIC Translation Table

SPIN-X/CENTRAL ASCII TO EBCDIC TRANSLATION

CHARACTER	ASCII (HEX)	EBCDIC (HEX)	CHARACTER	ASCII (HEX)	EBCDIC (HEX)
	00	00		20	40
	01	01	!	21	5A
	02	02	"	22	7F
	03	03	#	23	7B
	04	04	\$	24	5B
	05	05	%	25	6C
	06	06	&	26	50
	07	07	'	27	7D
	08	08	(28	4D
	09	09)	29	5D
	0A	0A	*	2A	5C
	0B	0B	+	2B	4E
	0C	0C	,	2C	6B
	0D	0D	/	2D	60
	0E	0E	-	2E	4B
	0F	0F	.	2F	61
	10	10	0	30	F0
	11	11	1	31	F1
	12	12	2	32	F2
	13	13	3	33	F3
	14	14	4	34	F4
	15	15	5	35	F5
	16	16	6	36	F6
	17	17	7	37	F7
	18	18	8	38	F8
	19	19	9	39	F9
	1A	1A	:	3A	7A
	1B	1B	;	3B	5E
	1C	1C	<	3C	4C
	1D	1D	=	3D	7E
	1E	1E	>	3E	6E
	1F	1F	?	3F	6F

CHARACTER	ASCII (HEX)	EBCDIC (HEX)	CHARACTER	ASCII (HEX)	EBCDIC (HEX)
@	40	7C	a	60	79
A	41	C1	b	61	81
B	42	C2	c	62	82
C	43	C3	d	63	83
D	44	C4	e	64	84
E	45	C5	f	65	85
F	46	C6	g	66	86
G	47	C7	h	67	87
H	48	C8	i	68	88
I	49	C9	j	69	89
J	4A	D1	k	6A	91
K	4B	D2	l	6B	92
L	4C	D3	m	6C	93
M	4D	D4	n	6D	94
N	4E	D5	o	6E	95
O	4F	D6	p	6F	96
P	50	D7	q	70	97
Q	51	D8	r	71	98
R	52	D9	s	72	99
S	53	E2	t	73	A2
T	54	E3	u	74	A3
U	55	E4	v	75	A4
V	56	E5	w	76	A5
W	57	E6	x	77	A6
X	58	E7	y	78	A7
Y	59	E8	z	79	A8
Z	5A	E9	{	7A	A9
[5B	AD		7B	8B
\	5C	E0	}	7C	4F
]	5D	BD	~	7D	9B
^	5E	5F	"	7E	A1
_	5F	6D		7F	7F

CHARACTER	ASCII (HEX)	EBCDIC (HEX)	CHARACTER	ASCII (HEX)	EBCDIC (HEX)
	80	80		A0	A0
a	81	81	~	A1	A1
b	82	82	s	A2	A2
c	83	83	t	A3	A3
d	84	84	u	A4	A4
e	85	85	v	A5	A5
f	86	86	w	A6	A6
g	87	87	x	A7	A7
h	88	88	y	A8	A8
i	89	89	z	A9	A9
	8A	8A		AA	AA
{	8B	8B		AB	AB
	8C	8C		AC	AC
	8D	8D		AD	AD
	8E	8E		AE	AE
	8F	8F		AF	AF
	90	90		B0	B0
j	91	91		B1	B1
k	92	92		B2	B2
l	93	93		B3	B3
m	94	94		B4	B4
n	95	95		B5	B5
o	96	96		B6	B6
p	97	97		B7	B7
q	98	98		B8	B8
r	99	99		B9	B9
	9A	9A		BA	BA
}	9B	9B		BB	BB
	9C	9C		BC	BC
	9D	9D		BD	BD
	9E	9E		BE	BE
	9F	9F		BF	BF

CHARACTER	ASCII (HEX)	EBCDIC (HEX)	CHARACTER	ASCII (HEX)	EBCDIC (HEX)
	C0	C0	\	E0	E0
A	C1	C1		E1	E1
B	C2	C2	S	E2	E2
C	C3	C3	T	E3	E3
D	C4	C4	U	E4	E4
E	C5	C5	V	E5	E5
F	C6	C6	W	E6	E6
G	C7	C7	X	E7	E7
H	C8	C8	Y	E8	E8
I	C9	C9	Z	E9	E9
	CA	CA		EA	EA
	CB	CB		EB	EB
	CC	CC		EC	EC
	CD	CD		ED	ED
	CE	CE		EE	EE
	CF	CF		EF	EF
	D0	D0	0	F0	F0
J	D1	D1	1	F1	F1
K	D2	D2	2	F2	F2
L	D3	D3	3	F3	F3
M	D4	D4	4	F4	F4
N	D5	D5	5	F5	F5
O	D6	D6	6	F6	F6
P	D7	D7	7	F7	F7
Q	D8	D8	8	F8	F8
R	D9	D9	9	F9	F9
	DA	DA		FA	FA
	DB	DB		FB	FB
	DC	DC		FC	FC
	DD	DD		FD	FD
	DE	DE		FE	FE
	DF	DF		FF	FF

