A-LOG®

System Administrator's Manual

Version 3.8

Manual Order Number: ALG-380-060-1

This manual is applicable to A-LOG product at Version 3.8 release level. Unless otherwise stated in new editions of this manual or A-LOG technical release notes or document, this manual is also applicable to all subsequent releases of the A-LOG product.

Product information specified in this manual is subject to change without notice. Data used in examples are fictitious information unless otherwise specified. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the written permission of BSP Inc.

A-CONSOLE, A-GATHER, A-GATHER/FS for UNIX, A-AUTO, A-AUTO/ REMOTE, A-AUTO for UNIX, A-AUTO for OS/400, A-VOLMANAGER, A-RECOVERY, A-LOG, A-SPOOL, A-SPOOL/FS for UNIX, A-SPOOL/MAIL, A-SPOOL/ONLINE(CICS), A-SPOOL/RVIEW, A-NAVIGATOR, A-QUALITY and A-SUPERVISION are trademarks of BSP Inc.

All other trademarks are property of their respective companies.

This manual is designed for A-LOG System Administrators. It explains the procedure for installing the A-LOG system and describes the utilities for performing various A-LOG system maintenance functions.

Chapter 1 gives an overview of A-LOG system.

Chapter 2 describes the requirements and procedures for installing A-LOG system.

Chapter 3 provides a list of the datasets used by A-LOG system and their space requirements.

Chapter 4 describes A-LOG system backup and maintenance procedures such as database backup and recovery, database reorganization and expansion, applying PTFs to A-LOG load modules and so on.

Chapter 5 describes the utilities for A-LOG system maintenance.

Chapter 6 describes the A-LOG system libraries and their contents.

Chapter 7 describes the user release files generated by A-LOG system and their record layout, and the various exit routines provided by A-LOG system.

Chapter 8 describes the A-LOG Security Management Facility which is used to prevent unauthorized access to A-LOG functions and reports.

The following A-LOG manuals may be referenced together with this manual when using A-LOG system:

1) A-LOG Introduction Manual
2) A-LOG User's Manual
3) A-LOG Messages & Codes Manual

Readers' comments and suggestions are most welcome. A Suggestion Page is provided at the back of this manual for this purpose. Please duplicate and use this page for sending your comments and suggestions to BSP.

List of Figures

Chapter 1

A-LOG System Overview

1.1 A-LOG System Components

A-LOG system consists of the components as shown in Fig. 1.1.

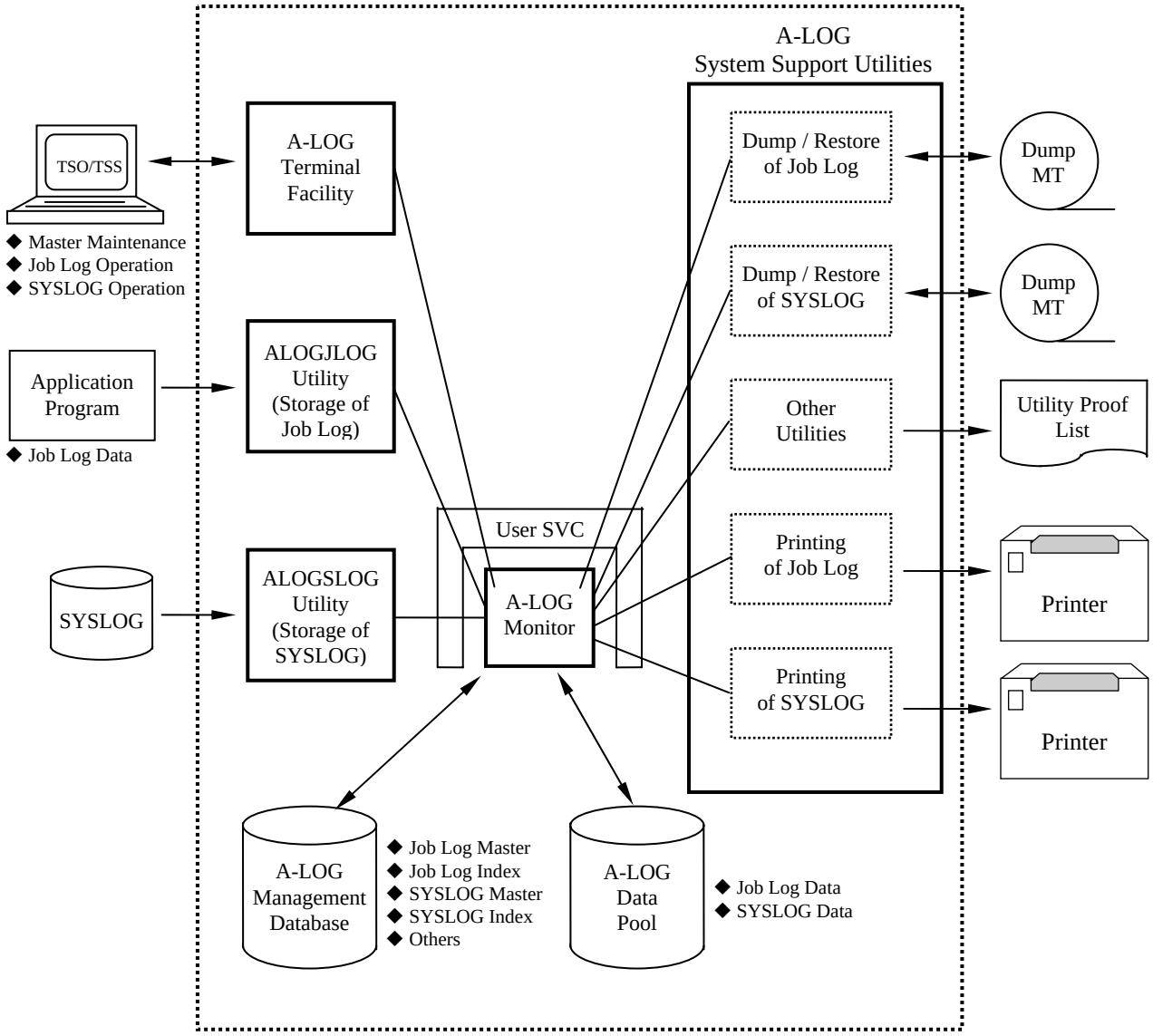


Fig. 1.1: A-LOG System Components

A-LOG Terminal Facility, ALOGJLOG utility, ALOGSLOG utility and A-LOG system support utilities communicate and exchange data with A-LOG Monitor via a user SVC routine of A-LOG system. A-LOG Monitor accesses the A-LOG Management Database and A-LOG Data Pool.

A-LOG Monitor

This is the nucleus of A-LOG system and it is a resident program. A-LOG Monitor performs the following functions:

- Monitors the execution of job log cataloging processes
- Monitors the execution of SYSLOG cataloging processes
- Manages the A-LOG Management Database and A-LOG Data Pool
- Processes A-LOG Monitor commands

ALOGJLOG Utility

This utility selects job logs of specified class from OS Spool and catalogs them into the A-LOG Management Database and A-LOG Data Pool.

ALOGSLOG Utility

This utility issues WRITELOG command at fixed time interval and selects the SYSLOGs of specified class from OS Spool and catalogs them into the A-LOG Management Database and A-LOG Data Pool.

A-LOG Terminal Facility

This terminal facility provides online registration, modification, deletion and retrieval of master records in A-LOG Management Database. It also enables inquiry, printing and deletion of job logs and/or SYSLOGs in A-LOG Data Pool via TSO/TSS terminals.

A-LOG System Support Utilities

These utilities are provided to support the operation of A-LOG system effectively and efficiently. These utilities are shown in Table 1.1.

Utility	General Description
ALOGBLOG	For generating Job Log Index Release File and an output listing.
ALOGCHKA	For checking whether A-LOG Monitor is active.
ALOGCHKL	For checking whether A-LOG Monitor has terminated.
ALOGCPY1	For copying individual files in A-LOG Management Database and output its contents as a sequential file.
ALOGCPY2	For restoring individual files in the sequential file created by ALOGCPY1 utility into the A-LOG Management Database.

Table 1.1.a: A-LOG System Support Utilities

Utility	General Description

ALOGDATE	For setting the Operation Date of A-LOG.
ALOGDLT1	For deleting dumped Job Log or SYSLOG records from A-LOG Data Pool.
ALOGDLT4	For deleting expired Dump MT Information Master records from A-LOG Management Database.
ALOGDLT5	For deleting expired Job Log or SYSLOG records from A-LOG Data Pool.
ALOGDMP1	For conversion of A-LOG Management Database into sequential files. This utility generates 3 files for use by ALOGDMP2 utility and ALOGDMP3 utility.
ALOGDMP2	For dumping job logs based on the file generated by ALOGDMP1 utility or for deleting the dumped job log records from A-LOG Management Database.
ALOGDMP3	For deleting job log records from A-LOG Management Database based on the file generated by ALOGDMP1 utility.
ALOGDMPC	For copying Dump MT created by ALOGJDMP utility or ALOGLDMP utility to another magnetic tape.
ALOGHOLT	For registration, inquiry, modification and deletion of Holiday Master records.
ALOGJDMP	For archiving expired job log records onto magnetic tape.
ALOGJMST	For registration, inquiry, modification and deletion of Job Log or SYSLOG master records.
ALOGJRST	For restoring job log records dumped by ALOGJDMP utility into A-LOG Management Database and A-LOG Data Pool.
ALOGLDMP	For archiving expired SYSLOG records onto magnetic tape.
ALOGLLOG	For generating SYSLOG Index Release File and an output listing.
ALOGLRST	For restoring SYSLOG records dumped by ALOGLDMP utility into A-LOG Management Database and A-LOG Data Pool.
ALOGRSEL	For invoking ALOGJRST utility or ALOGLRST utility to restore job logs or SYSLOGs that have been marked for restoration by users using A-LOG Terminal Facility.
ALOGSLOG	For generating SYSLOG Index Release File and an output listing.
WTRJUTL	For printing job logs to an output device.
WTRSUTL	For printing SYSLOGs to an output device.

Table 1.1.b: A-LOG System Support Utilities

1.2 A-LOG System Operation Requirements

System Software & Utilities Required

1) Sort/Merge Utilities

2) VTAM, TSO or TSS

3) Standard Access Methods:
   - VSAM
   - QSAM
   - BSAM
   - BPAM
   - DAM

4) Job Entry Sub-Systems:
   - JES
   - AFJES
   - JES3
   - JES2
   - JES/E
   - JSS3
   - JES2MAS
   - JSS4

5) Attributes of user SVC routine:
   - Type 3, No Lock, Unauthorized, Resident

Virtual Memory Required

The following virtual memory sizes are required for operation of A-LOG system:

1) Execution Region Sizes
   - A-LOG Monitor (Started Task)

     Execution region size is determined by values of the keywords specified in RUNSPM2 and ABSIPRM1 parameters. If the default values generated by the installation process are used, an execution region size of at least 2,048K bytes is required. If any of the values is changed, the execution region size must be increased accordingly. The following table highlights the necessary increase in the execution region size for each additional unit of the value of a keyword:

Member	Keyword	Value set after installation	Increment for each additional unit
RUNSPM2	NRQE	2000	100 bytes
	NUSQ	20	284 bytes
ABSIPRM1	FINDWK	100	488 bytes
	ACCESS	32	352 bytes
	SETRC	20	488 bytes
	ANALIZE	64	90 bytes

Note: Refer to ”**6.2 Parameter Library**“ for explanation of ABSIPRM1 parameter.

- ALOGJLOG Utility (Started Task)

  Excluding the user exit routine, an execution region size of at least 512K bytes is required.

- ALOGSLOG Utility (Started Task)

  Excluding the user exit routine, an execution region size of at least 512K bytes is required.

- Other Non-Resident Batch Programs

  An execution region size of at least 256K bytes is required.

- A-LOG Terminal Facility

  Execution region size is determined by values of the keywords specified in ASPPRM1 and ASPOPTB parameters. If the default values generated by the installation processes are used, and A-LOG Terminal Facility is started from READY state of a TSO/TSS terminal, an execution region size of at least 1,024K bytes is required. If any of the values is changed, the execution region size of A-LOG Terminal Facility must be increased accordingly. The following table shows the calculation of the work areas required by the various menu screens:

No	Menu Number	Work Area Calculation
✶	1.1	NJAM x 38
✸	1.2	NJBI x 178 + NJBD x 114 + 100 x 40 + MAXLL x (MAXLC +1) x 2
✹	2.2	NDMM x 30
❋	3.1 and 3.2	NSJBI x 178 + NSLOG x 160 + 100 x 40 + MAXLL x (MAXLC +1) x 2

Note: Underlined words are keywords of ASPOPTB parameter
Not underlined words are keywords of ASPPRM1 parameter

Therefore:

Execution Region Size = Maximum of Work Area Size ✶~❋ + NERM x 60 + NDATA + 1,956

2) Common Service Area (CSA)

A-LOG Monitor uses CSA Subpool 228 (Page Fixed Area) and Subpool 241 (Paging Area). The sizes of these areas to be acquired are dependent on the keywords specified in RUNSPM2 parameter. These are shown as follows:

CSA Area	Area Name	Size Calculation
Subpool 228	CQH	180 bytes
	CQE	NCQE x 48 bytes
	MAQ	NMAQ x 44 bytes
	MQE	NMQE x 32 bytes
	SGT	2,048 bytes

CSA Area	Area Name	Size Calculation
Subpool 241	ATB	NATB x 4,096 bytes
	PQH	NPQH x 148 bytes

Note: Underlined words are keywords of RUNSPM2 parameter

Chapter 2

A-LOG System Installation

2.1 A-LOG System Operating Environment

2.1.1 Operating Systems Supported

The operating systems in which A-LOG system can operate are shown below:

IBM	Fujitsu	Hitachi
MVS/SP MVS/XA (SP2.1 and above MVS/ESA (SP3.1 and above)	OSIV/F4 E30 OSIV/F4 E40 OSIV/F4 - MSP/E10~E20 OSIV/F4 - MSP/EX	VOS3 VOS3/ES1 VOS3/AS

Table 2.1: Supported Operating Systems

Depending on the operating system chosen, some A-LOG programs may require minor modifications for A-LOG system to operate properly. Refer to "**4.4 A-LOG Load Module Maintenance**" and "**Appendix 7: PTFs for Operating Systems**" for further details.

2.1.2 Operating System Requirements

Authorizing Attribute

Some A-LOG load modules perform functions that require APF authorization. Therefore, the A-LOG Load Module Library must be defined in SYS1.PARMLIB as an APF-authorized library.

Use of SVC Routine

A-LOG system uses a user SVC routine for its operations. The user SVC routine may be stored in SYS1.LPALIB. Refer to "**2.2.9 Create User SVC Routine**" for further details.

IPL of Operating System

Since the A-LOG Load Module Library must be APF-authorized and a user SVC routine is required for A-LOG system, an IPL of the operating system with 'CLPA' option is required before using the A-LOG system. A schedule for performing IPL should be arranged prior to the installation and testing of A-LOG system.

2.1.3 Access Methods

A-LOG system requires a number of datasets for its operations. These are the A-LOG Management Database, A-LOG Data Pool, A-LOG work files and A-LOG system libraries. These datasets use standard dataset organization and they are accessed by standard access methods provided by the operating system. Therefore, these access methods must be available in the operating system as they are prerequisites for the operation of A-LOG system. The following standard access methods are used:

- For sequential dataset : QSAM and BSAM
- For partitioned dataset : BPAM
- For direct access dataset : BDAM
- For VSAM dataset : VSAM

2.1.4 Terminal

A-LOG system does not require exclusive use of a terminal for its operations. However, the A-LOG Terminal Facility is available for access to A-LOG system via a TSO/TSS terminal.

2.2 A-LOG System Installation Procedure

2.2.1 Installation Procedure

Fig. 2.1. highlights the installation procedure for A-LOG system:

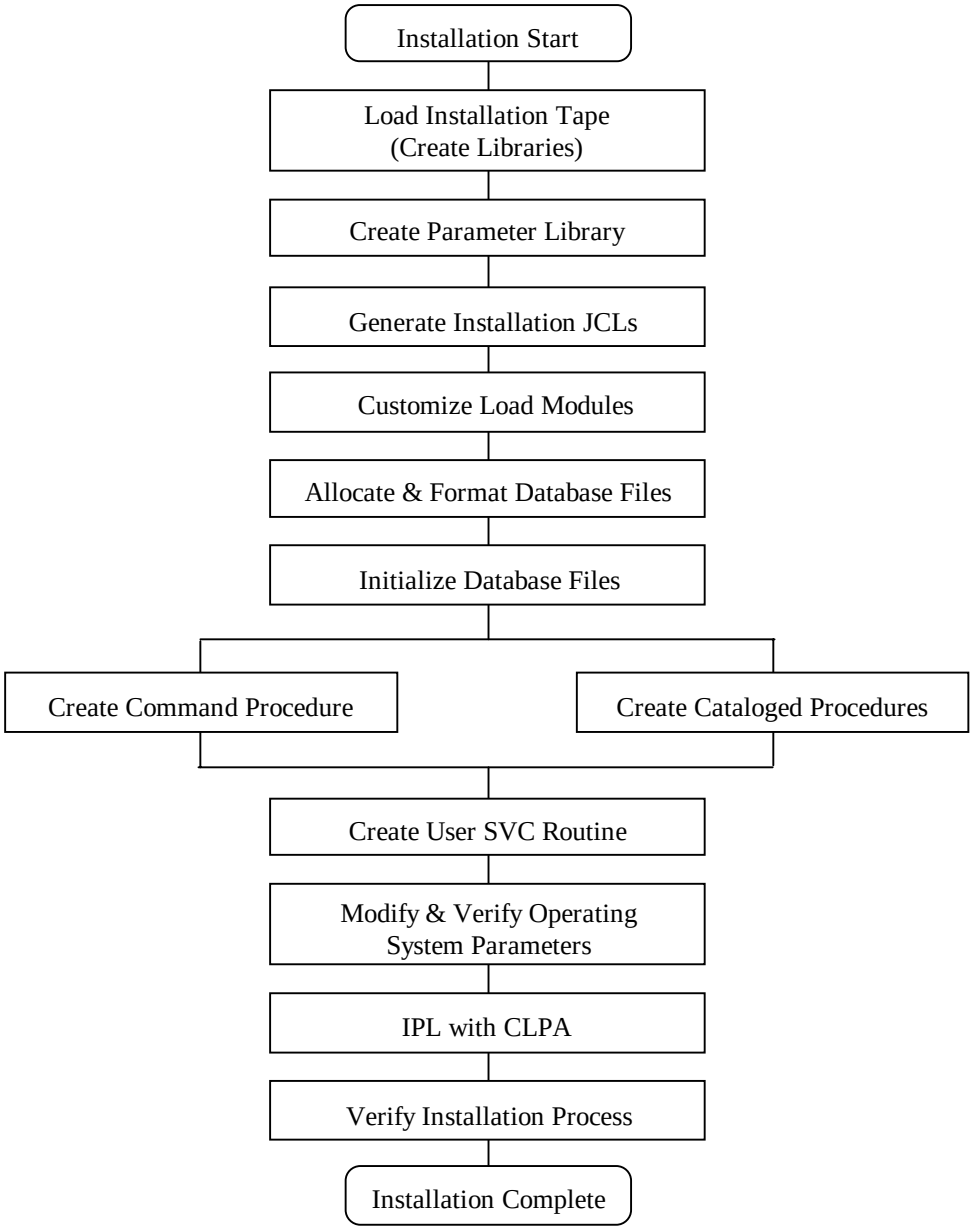


Fig. 2.1: A-LOG Installation Procedure

2.2.2 Load Installation Tape

Table 2.2 highlights the contents of A-LOG installation tape:

Label	Dataset Name	Description
1,SL	V0380.LOAD	A-LOG Load Module Library
2,SL	V0380.MAP	A-LOG Terminal Screen Library
3,SL	V0380.USERS.LOAD	A-LOG User Load Module Library
4,SL	V0380.INSTALL	A-LOG Installation JCL Library
5,SL	V0380.ZAP	A-LOG PTF Library

Table 2.2: Contents of A-LOG Installation Tape

The installation tape is created using the partitioned dataset copy utility supplied by computer system manufacture. Refer to "**5.1.2 Manufacturer-Supplied Utilities"** for further details. To copy the partitioned datasets from the installation tape to disk, modify the following JCLs according to installation environment and submit the job for execution:

```
//ALOGCOPY JOB ,BSP,MSGLASS=C,CLASS=A
//COPY     EXEC PGM=program-name
//STEPCAT  DD DSN=ALOG.UCAT,DISP=SHR
//SYSPRINT DD SYSOUT=*
//IN1      DD DSN=V0380.LOAD,DISP=(SHR,PASS),UNIT=TAPE,
//         VOL=SER=volser-tape,LABEL=(1,SL)
//IN2      DD DSN=V0380.MAP,DISP=(SHR,PASS),UNIT=TAPE,
//         VOL=SER=volser-tape,LABEL=(2,SL)
//IN3      DD DSN=V0380.USERS.LOAD,DISP=(SHR,PASS),UNIT=TAPE,
//         VOL=SER=volser-tape,LABEL=(3,SL)
//IN4      DD DSN=V0380.INSTALL,DISP=(SHR,PASS),UNIT=TAPE,
//         VOL=SER=volser-taper,LABEL=(4,SL)
//IN5      DD DSN=V0380.ZAP,DISP=(SHR,PASS),UNIT=TAPE,
//         VOL=SER=volser-tape,LABEL=(5,SL)
//OUT1     DD DSN=ALOG.V0380.LOAD,DISP=(NEW,CATLG,DELETE),
//         UNIT=SYSDA,VOL=SER= volser-disk,SPACE=(CYL,(15,,100))
//OUT2     DD DSN=ALOG.V0380.MAPLIB,DISP=(NEW,CATLG,DELETE),
//         VOL=SER= volser-disk,SPACE=(CYL,(3,,80))
//OUT3     DD DSN=ALOG.V0380.INSTALL,DISP=(NEW,CATLG,DELETE),
//         VOL=SER= volser-disk,SPACE=(CYL,(10,,100))
//OUT4     DD DSN=ALOG.V0380.ZAPLIB,DISP=(NEW,CATLG,DELETE),
//         VOL=SER= volser-disk,SPACE=(CYL,(5,,200))
```

```
//SYSIN    DD *
 COPY INDD=IN1,OUTDD=OUT1
 COPY INDD=IN2,OUTDD=OUT2
 COPY INDD=IN3,OUTDD=OUT1
 COPY INDD=IN4,OUTDD=OUT3
 COPY INDD=IN5,OUTDD=OUT4
/*
```

2.2.3 Create Parameter Library

JCLs for creating an A-LOG Parameter Library can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'PARMLIB' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for creating A-LOG Parameter Library will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for creating the A-LOG Parameter Library.

2.2.4 Customize Load Modules

A-LOG system is provided in load module form. Depending on the installation environment, customization of some load modules may be necessary. Customization of a load module is performed using the load module modification utility provided by computer system manufacturer.

JCLs for customizing load modules can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'PTF' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for customizing load modules will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for customizing the load modules.

Contents of PTFs are stored in the A-LOG PTF Library (ALOG.V0380.ZAPLIB). PTFs are classified into the following categories:

1) Operating System Zaps

   These are PTFs to customize A-LOG system to support various operating systems. At installation time, these PTFs must be applied to some A-LOG load modules. Refer to "**Appendix 7: PTFs for Operating Systems**" for further details.

2) Installation Environment Zaps

These are PTFs to customize A-LOG system to support specific user installation environment, such as SVC numer for A-LOG system. It is desirable that each user maintains a history of the PTFs applied to its installation environment.

3) Problem Correction Zaps

These are PTFs to correct problems encountered by A-LOG users. The installation tape usually contains the latest load modules with the most recent PTFs applied. The highest PTF number applied becomes the zap level for the installation tape. PTFs exceeding this zap level may be applied selectively if appropriate to the problems encountered by A-LOG users..

2.2.5 Allocate & Format Database Files

JCLs for allocating and formatting the A-LOG database files can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'ALLOCATE' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for allocating and formatting A-LOG database files will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for allocation and formatting of the A-LOG database files.

2.2.6 Initialize Database Files

JCLs for initializing the A-LOG database files can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'INITVSM' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for initializing A-LOG database files will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for initialization of the A-LOG database files.

2.2.7 Create Command Procedure

JCLs for creating the A-LOG command procedure can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'USLP' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for creating A-LOG command procedure will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for creating the A-LOG command procedure.

Setting Up the Initial A-LOG Terminal Screen

The initial A-LOG terminal screen to be displayed when A-LOG Terminal Facility is invoked from TSO or TSS terminal can be specified in options ✶ and ✸ in first line of the command procedure.

```
PROC 0 MODE(C) OPT(ASPL(  ✶  )DCDUSER(  ✸  )) . . .
```

Permissible Initial Screen	Specify in ✶	Specify in ✸
[A-SPOOL/A-LOG Select] Screen	H.0.0	ASPL/ALOG
[Hello] Screen	H.0.0	ALOG
[0.0.0 Initial Menu] Screen	0.0.0	ALOG or ASPL/ALOG (See Note)
[1.0.0 Master Management] Screen	1.0.0	
[1.1.0 Job Log/SYSLOG Master Maintenance] Screen	1.1.0	
[2.0.0 Job Log Database Management] Screen	2.0.0	
[2.1.0 Job Log Index Maintenance] Screen	2.1.0	
[2.2.0 Dump MT Inquiry] Screen	2.2.0	
[3.0.0 SYSLOG Management] Screen	3.0.0	
[3.1.0 SYSLOG Data Inquiry] Screen	3.1.0	
[3.2.0 SYSLOG Index Maintenance] Screen	3.2.0	
[9.0.0 System Management] Screen	9.0.0	
[9.1.0 Terminal/PF-Keys Setting] Screen	9.1.0	

Note: Depending on the option specified in ✸, the following processing is performed when 'X' is entered in [0.0.0 Initial Menu] screen:

- If 'ALOG' is specified : Return to ready state of the TSO or TSS terminal.
- If 'ALOG/ASPL' is specified : Switch to [A-SPOOL/A-LOG Select] screen.

Cautions: If 'ALOG/ASPL' is specified in ✸, the following libraries and files must be common to both A-LOG and A-SPOOL systems:

- Terminal screen libraries : ALOG.V0380.MAP and ASPOOL.V0380.MAPLIB
- Load module libraries : ALOG.V0380.LOAD and ASPOOL.V0380.LOAD
- Parameter libraries : ALOG.V0380.PARMLIB and ASPOOL.V0380.PARMLIB
- Security files : ALOG.SECRTBL and ASPOOL.SECRTBL

The following definitions must also be added to the DCDALCTB parameter.

```
DEFINE(ALOG)
ALLOC F(DDSCT) DA('ALOG.DB.SCT') SHR
ALLOC F(DDBKS) DA('ALOG.DB.BKS') SHR
```

2.2.8 Create Cataloged Procedures

JCLs for creating the A-LOG cataloged procedures can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'PROCLIB' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for creating A-LOG cataloged procedures will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for creating the A-LOG cataloged procedures.

2.2.9 Create User SVC Routine

Two methods are available for creating the user SVC routine. These are: 1) create by link editing into existing system; and 2) create by system generation. These are explained and cautioned as follows:

1) Create by Link Editing into Existing System

Create the user SVC routine into existing system using the linkage editor of the operating system.

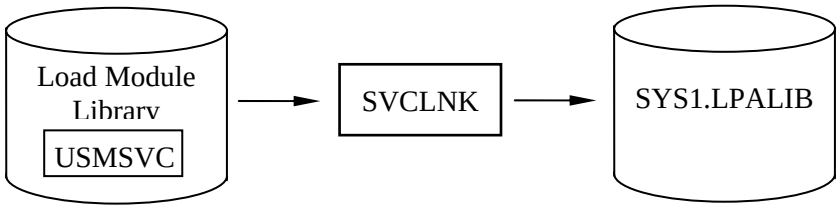
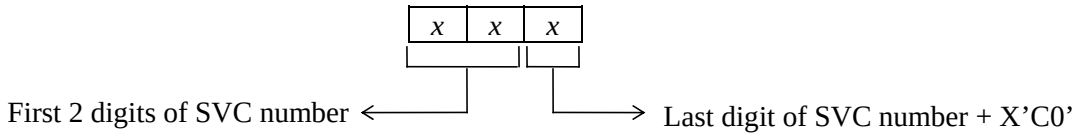


Fig. 2.2: Link Editing of SVC Routine

Member name of the user SVC routine is as follows:

- IBM: IGC00*xxx*
- Fujitsu: JFF00*xxx*
- Hitachi: JFF00*xxx*

Where: *xxx* is derived as follows:



Note: For Fujitsu OS, if last digit of the SVC number is 0, then the last character is '0' and not '{'.

Examples: 250 → IGC0025{ (0 + X'C0' = X'C0', which is character '{')
251 → IGC0025A (1 + X'C0' = X'C1', which is character 'A')
252 → IGC0025B (2 + X'C0' = X'C2', which is character 'B')

JCLs for link editing A-LOG user SVC routine can be generated automatically using A-LOG Installation JCL Generator. Refer to "**Appendix 4: A-LOG Installation JCL Generator**" for further details. If function code 'SVCLNK' is specified in member $INSTALL of A-LOG Installation JCL Library (ALOG.V0380.INSTALL) and $INSTALL is submitted for execution, the JCLs for link editing user SVC routine will be generated automatically upon successful execution of $INSTALL.

Check the generated JCLs and modify them, if necessary, before submitting them for link editing the user SVC routine. The following two steps are contained in generated JCLs:

Step 1: Create user SVC routine using Linkage Editor

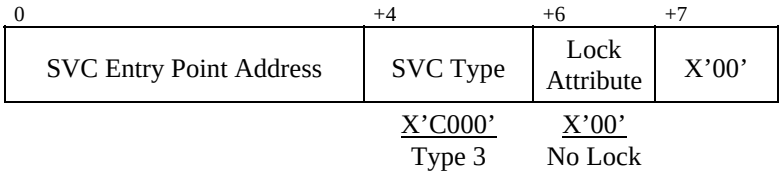
This step creates the user SVC routine into SYS1.LPALIB using the Linkage Editor.

Step 2: Verify SVC table attribute of SYS1.NUCLEUS

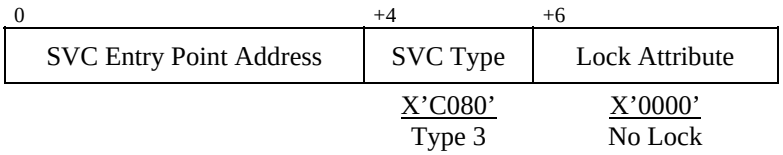
This step gets a module dump of SVCTABLE (SVC table) in SYS1.NUCLEUS and verifies the SVC attributes of the corresponding user SVC routine. SVCTABLE contains an entry for each user SVC routine. Each entry is 4 or 8 bytes depending on operating system and it stores the SVC routine entry address and attributes. If the entry of A-LOG user SVC routine does not have the required attributes (i.e. Type 3 and No Lock), it must be modified with a zap.

Examples of SVCTABLE:

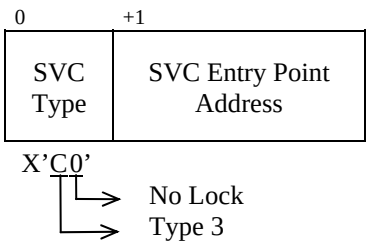
- IBM MVS
- Fujitsu OSIV/F4 MSP/E20



- Hitachi VOS3



- Fujitsu OSIV/F4
- Fujitsu OSIV/E30
- Fujitsu OSIV/E40
- Fujitsu OSIV/MSP/E10



The offset of the user SVC entry in SVCTABLE can be calculated as follows:

- For MVS, VOS3, F4-MSP/E20 : Offset in decimal = SVC number x 8
- For OSIV/F4(E30, E40, MSP/E10) : Offset in decimal = SVC number x 4

Example of offset calculation:

For a user SVC number 250 for IBM MVS/XA operating system:
Offset = 250 x 8 = 2000 (in decimal) = X'7D0' (in hexadecimal)
Required SVC entry is located at start of SVC table + X'7D0'

2) Create by System Generation

To create the user SVC routine during system generation, the following steps must be performed:

Step 1: Create user SVC routine using Linkage Editor

This step creates the user SVC routine into SYS1.LPALIB using the Linkage Editor.

Step 2: Register SVC Number

SVC attributes: Type 3, No Lock, No Authorization

Macro Name	Operand
SVCTABLE	SVC-252-T3-FC00

→ Not Authorized
→ Type 3 No Lock
→ SVC Number

Step 3: Resident assignment and library specification for SVC routine

Macro Name	Operand	
DATASET	LPALIB,	Indicate location in LPA
	PDS=SYS1.LPALIB,	Library name containing SVC routine
	MEMBERS(,…),	Non-resident SVCs
	RESIDENT(JFF0025B,…)	Resident SVCs

→ For VOS3

2.2.10 Modify & Verify Operating System Parameters

Upon completion of the user SVC routine creation, verify the following items prior to an IPL of the operating system:

1) A-LOG Load Module Library is APF-authorized
   - IBM: IEAAPF*xx*
   - Fujitsu: KAAAPF*xx*
   - Hitachi: JAAAPF*xx*

```
SYS1.LINKLIB        SYSRES,
ALOG.V0380.LOAD     ASPL01,
AUTO.LINKLIB        AUTO01
```

2) The user SVC routine is specified with page fixed (resident)
   - IBM: IEAFIX*xx*
   - Fujitsu: KAAFIX*xx*
   - Hitachi: JAAFIX*xx*

```
SYS1.LPALIB     IGC0025B,IGC0025C
```

   Note: For IBM MVS/XA or higher releases, the following item must be specified in the IEASYS*xx* parameter:

```
FIX=(00,NOPROT)
```

3) User SVC routine attribute is registered properly

   Refer to "**2.2.9 Create User SVC Routine**", "1) Create by Link Editing into Existing System", "Step 2: Verify SVC table attribute of SYS1.NUCLEUS" for further details.

2.2.11 IPL with CLPA

If all the above-mentioned installation works are completed successfully, IPL the operating system with CLPA (Create Link Pack Area).

2.2.12 Verify Installation Process

Upon completion of all the installation works, verify the A-LOG system using the following procedure:

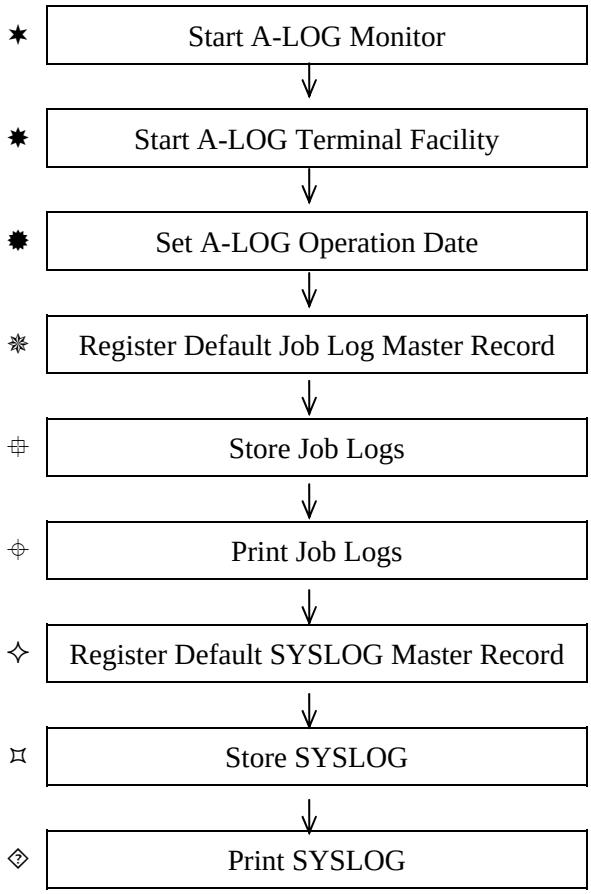


Fig. 2.3: A-LOG Verification Procedure

✶ Start A-LOG Monitor

After installation, always performs a COLD start of A-LOG Monitor by entering the following command from Operator's console:

```
S ALOG,TYPE=COLD
```

Check to ensure that the following messages are displayed on the Operator's console:

```
SPM003I  COLD START OPTION ASSIGNED
SPM004A  ALOG WAITING FOR 'DATE' COMMAND
SPM001I  A-LOG IS ACTIVE
```

✸ Start A-LOG Terminal Facility

After A-LOG Monitor is started successfully, invoke the A-LOG Terminal Facility from a TSO or TSS terminal by entering the A-LOG command procedure name (standard name is ALOG) in the TSO or TSS terminal.

✹ Set A-LOG Operation Date

Enter the DATE command through Operator's console or execute the ALOGDATE utility to set an Operation Date for A-LOG system as follows:

- Date command:

```
F ALOG,DATE
```

After entering above command, check to ensure that the following messages are displayed on Operator's console.

```
SPM011I  OPERATOR TYPED-ID : DATE
SPM202I  CURRENT DATE SET TO yymmdd
SPM200I  START/UP DATE PROCESSOR
SPM201I  FINISH DATE PROCESSOR
SPM305I  WRITER LOGX(Y) WAS CLOSED
SPM304I  WRITER LOGY(X) WAS SELECTED
```

- Executing ALOGDATE utility:

```
S ALOGDATE
```

The A-LOG Operation Date is set to CPU date after the command is entered or after successful execution of ALOGDATE utility.

✵ Register Default Job Log Master Record

Create a default job log master record with Job Log ID 'JOBLOG' using [1.1.0 Job Log/SYSLOG Master Maintenance] screen of A-LOG Terminal Facility. This default master record is used when a job log without the Job Log ID being specified is stored in the A-LOG database. Refer to **A-LOG User's Manual** for further details about the specification and registration methods.

⌖ Store Job Logs

Start ALOGJLOG utility and catalog a job log into the A-LOG database. Ensure that the SYSOUT class of the job log matches the SYSOUT class defined in ALOGJLOG utility. If not, modify the SYSOUT class of ALOGJLOG utility. If the SYSOUT class of the job log matched with the pre-defined SYSOUT class but the job log is not cataloged, check to ensure that the job log is not held.

- Starting ALOGJLOG utility:

```
S ALOGJLOG
```

- Terminating ALOGJLOG utility:

```
P ALOGJLOG
```

- Modifying SYSOUT class:

```
F ALOGJLOG,CLASS=xxxxxxxx
```

Where: *xxxxxxxx* are the SYSOUT classes (up to a maximum of 8)

⊕ Print Job Logs

Check that the job log has been cataloged successfully by inquiring the job log index and data using A-LOG Terminal Facility. Then print a copy of the job log. Refer to **A-LOG User's Manual** for further details about the print method.

✧ Register Default SYSLOG Master Record

Create a SYSLOG master record using [1.1.0 Job Log/ SYSLOG Master Maintenance] screen of A-LOG Terminal Facility. Specify the SYSLOG ID of the registered SYSLOG master record in the SYSLOG control card of ALOGSLOG utility. Refer to **A-LOG User's Manual** for further details about the specification method.

⌑ Store SYSLOG

Start ALOGSLOG utility and catalog a SYSLOG into the A-LOG database. Ensure that the SYSOUT class of the SYSLOG matches the SYSOUT class defined in ALOGSLOG utility.

- Starting ALOGSLOG utility:

```
S ALOGSLOG
```

- Terminating ALOGSLOG utility:

```
P ALOGSLOG
```

⍰ Print SYSLOG

Check that the SYSLOG has been cataloged successfully by inquiring the SYSLOG index and data using A-LOG Terminal Facility. Then print a copy of the SYSLOG. Refer to **A-LOG User's Manual** for further details about the print method.

Chapter 3

A-LOG System Datasets

3.1 List of A-LOG System Datasets

The datasets used by A-LOG system are shown in the following table:

Dataset Name	File Organization	File Type	Record Length	Key Length	Block Size	File Name
ALOG.DBIDEX	VSAM	KSDS	Average 82 Max. 3000	52	4096 (CI Size)	A-LOG Management Database (Index Part)
ALOG.DBDATA	VSAM	KSDS	Average 82 Max. 3000	6	4096 (CI Size)	A-LOG Management Database (Data Part)
ALOG.DB.SCT	DAM	F	See Note	-	See Note	A-LOG Data Pool (Index Part)
ALOG.DB.BKS	DAM	F	See Note	-	See Note	A-LOG Data Pool (Data Part)
ALOG.HOTLOG	DAM	F	4096	-	4096	Hot Log File
ALOG.V0380. SECRTBL	DAM	F	1012	12	1012	Security Management Table
ALOG.V0380. SECRLOGX	DAM	F	1024	-	1024	Security Log File X
ALOG.V0380. SECRLOGY	DAM	F	1024	-	1024	Security Log File Y
ALOG.V0380. LOAD	PO	U	-	-	6774	A-LOG Load Module Library
ALOG.V0380. INSTALL	PO	FB	80	-	800	A-LOG Installation JCL Library
ALOG.V0380. PARMLIB	PO	FB	80	-	800	A-LOG Parameter Library
ALOG.V0380. MAPLIB	PO	FB	80	-	800	A-LOG Terminal Screen Library
ALOG.V0380. ZAPLIB	PO	FB	80	-	800	A-LOG PTF Library

Table 3.1: A-LOG Datasets

Note: The record size and block size of SCT and BKS datasets in A-LOG Data Pool depend on the DASD device type. Refer to Table 3.2 for further details.

Device Type	Tracks per Cylinder	Bytes per Track	SCT Block Size	BKS Block Size	Blocks per Cylinder
IBM 3350 Fujitsu 493 Hitachi 8595	30	19,069	9,442	9,442	60
Fujitsu 6421	20	27,051	8,758	8,758	60
IBM 3380 Fujitsu 6425 Hitachi 8597	15	47,476	9,076	9,076	75
IBM 3390 Hitachi 6587	15	55,664	10,796	10,796	75
IBM 9345	15	46,456	7,215	7,215	75

Table 3.2: Block Size of A-LOG Data Pool

3.2 List of Files in A-LOG Management Database

The files in A-LOG Management Database are shown in the following table:

File Name	File Name Abbreviation	File No.	Content
System File	SYS	10	Operation date
Job Log File	JAM	20	Job log and SYSLOG attributes
Dump MT Information File	DMM	80	Dump MT management information
Job Log Index File	JBI	110	Job log attributes and information, SYSLOG attributes and information

Table 3.3: Files in A-LOG Management Database

3.3 Space Requirements for A-LOG System Datasets

3.3.1 General Space Requirements

The space requirements for A-LOG system datasets are shown in the following table:

Dataset Description	Standard Dataset Name	Space Requirement
A-LOG Management Database	ALOG.DBIDEX	Refer to Section 3.3.2
	ALOG.DBDATA	
A-LOG Data Pool	ALOG.DB.SCT	Refer to Section 3.3.5
	ALOG.DB.BKS	Refer to Section 3.3.3
Hot Log File	ALOG.HOTLOG	Refer to Section 3.3.6
Security Management Table	ALOG.V0380.SECRTBL	2 Cylinders
Security Log Files	ALOG.V0380.SECRLOGX	2 Cylinders
	ALOG.V0380.SECRLOGY	2 Cylinders
A-LOG Libraries	ALOG.V0380.LOAD	10 Cylinders
	ALOG.V0380.INSTALL	3 Cylinders
	ALOG.V0380.PARMLIB	2 Cylinders
	ALOG.V0380.MAPLIB	2 Cylinders
	ALOGV0380.ZAPLIB	4 Cylinders

Table 3.4: Space Requirements of A-LOG System Datasets

Note: The above space requirements are based on IBM 3380 DASD device type. Refer to Table 3.2 for the storage capacity of other device types.

3.3.2 Space Estimation for A-LOG Management Database

A-LOG Management Database consists of two VSAM datasets: ALOG.DBIDEX and ALOG. DBDATA. The disk space required for A-LOG Management Database is the total of the disk space required for each of these datasets. Their requirements can be estimated as follows:

$$\text{Space Requirement for ALOG.DBIDEX} = \frac{\text{Expected Number of Records}}{10 \times \text{TRK} \times \left(\frac{4{,}096}{56 \times \text{Number of Keys}}\right)} \text{ cylinders}$$

$$\text{Space Requirement for ALOG.DBDATA} = \frac{\text{Expected Number of Records}}{10 \times \text{TRK} \times \left(\frac{4{,}096}{\text{Record Length}}\right)} \text{ cylinders}$$

Where: TRK Number of tracks per cylinder

⬚ Number of records per Control Interval

☐ Number of records per cylinder

File Name	No. of Keys	Record Length
SYS	1	86
JAM	2	518
DMM	1	98
JBI	4	794

An example for IBM 3380 DASD device type is shown below:

File Name	No. of Records	ALOG.DBIDEX			ALOG.DBDATA		
		Record Length	No. per Cylinder	Space Required	Record Length	No. per Cylinder	Space Required
SYS	1	56	4,000	0.0	86	7,050	0.0
JAM	100	56	6,600	0.0	518	1,050	0.1
DMM	1,000	56	12,000	0.1	98	6,150	0.2
JBI	30,000	56	3,000	10.0	794	750	40.0
			Total	10.1		Total	40.3

Table 3.5: A-LOG Management Database Space Estimation for IBM 3380 DASD

Note: The number of JBI records is not the number of jobs but the total of job logs and SYSLOGs. Estimate the number of JBI records as follows:

Number of JBI records = (Average Number of Sysout for one Job / 7 + 1) x Number of Target Jobs

3.3.3 Space Estimation for Data Part (BKS) of A-LOG Data Pool

The disk space required for data part of A-LOG Data Pool can be estimated as follows:

$$\text{BKS Space Requirement} = \frac{\text{Total Number of Pages to be Stored}}{\boxed{(\text{BLKSIZE} / \text{AVB})} \times \text{NBLK}} \text{ cylinders}$$

Where: ⬚ Number of pages per block

Notes: 1) Total Number of Pages to be Stored = Number of Jobs per Day x Average Number of Pages per Job Log x Retention Days

2) BLKSIZE = Block size of A-LOG Data Pool (refer to Table 3.2)

3) AVB = Average number of byte per page after compression of blank

4) NBLK = Number of blocks per cylinder (refer to Table 3.2)

Examples: 1) To catalog 100,000 pages of job logs on IBM 3380 DASD device type

Total Number of Pages to be Stored = 100,000

BLKSIZE = 9,076

AVB = 3,000 *

NBLK = 75

$$\text{BKS Space Requirement} = \frac{100{,}000}{(9{,}076 / 3{,}000) \times 75} = \frac{100{,}000}{3 \times 75} = \frac{100{,}000}{225}$$

= 455 cylinders

Note: BKS Space Requirement = 334 cylinders if AVB is 2,000.

2) To catalog 500,000 lines of SYSLOG on IBM 3380 DASD device type. If one page has 180 lines, 500,000 lines of SYSLOG is about 2,778 pages.

Total Number of Pages to be Stored = 2,778

BLKSIZE = 9,076

AVB = 15,000 (15,000 bytes is about 180 lines of SYSLOG)

NBLK = 75

$$\text{BKS Space Requirement} = \frac{2{,}778}{(9{,}076 / 15{,}000) \times 75} = \frac{2{,}778}{0.5 \times 75} = \frac{2{,}778}{37}$$

= 75 cylinders

Caution: DAM file has a restriction that the total TRK number must not be greater than 65,536. Therefore, the estimated space requirement must be reduced when the result of the multiplication of space size (cylinder) estimated by the above-mentioned formula and the TRK number per cylinder (refer to Table 3.2) is larger than 65,536. Similarly, when BKS is a multi-volume dataset, the total TRK number must not be greater than 65,536.

3.3.4 Estimation for MAXREPC Parameter of Member BKSPARM

The value for MAXREPC parameter of member BKSPARM can be estimated using the following formula:

$$\text{MAXREPC Requirement} = \frac{\text{NBLK x BKS Allocation Size}}{\text{Number of BKS Blocks per Job}}$$

Notes: 1) NBLK = Number of blocks per cylinder (refer to Table 3.2)

2) BKS Allocation Size = The unit is in cylinders

3) Number of BKS Blocks per Job = Allocation size of BKS block per extent*

* 1 extent record can catalog up to 5 extents.

Example: Assuming: NBLK = 75
BKS Allocation Size = 520
Number of BKS Blocks per Job = 3

$$\text{MAXREPC Requirement} = \frac{75 \times 520}{3} = 13{,}000$$

3.3.5 Space Estimation for Index Part (SCT) of A-LOG Data Pool

The disk space required for index part of A-LOG Data Pool can be estimated as follows:

$$\text{SCT Space Requirement} = \frac{\text{Total Number of SCT Blocks}}{\text{Number of Blocks per Cylinder}} \text{ cylinders}$$

Notes: 1) Total Number of SCT Blocks = Number of Header Blocks ✶
+ Number of Extent Index Blocks ✸
+ Number of Extent Status Blocks ✹
+ Number of Extent Blocks ❋
+ Number of Status Blocks ⌖

Where: ✶ Number of Header Blocks = 1 (fixed)

✸ Number of Extent Index Blocks $= \left(\frac{M}{\left(\frac{L - 8}{20} \downarrow \right)} \uparrow \right)$

✹ Number of Extent Status Blocks $= \left(\frac{M}{L - 12} \uparrow \right)$

✵ Number of Extent Blocks $= \left(\frac{M}{\left(\frac{L - 8}{84} \downarrow \right)} \uparrow \right)$

⌖ Number of Status Blocks $= \left(\frac{N}{L - 12} \uparrow \right)$

Legend: $(\uparrow)$ Result of calculation is rounded up

$(\downarrow)$ Result of calculation is rounded down

Notes: 1) L : SCT block size (Refer to Table 3.2)

2) M : Maximum number of job logs to be cataloged (MAXREPC value specified in member BKSPARM)

3) N : Number of BKS blocks = Number of blocks per cylinder x BKS allocation size*

* BKSSIZE value specified in member BKSPARM

Example: Estimate the space requirement for cataloging 5,000 job logs on IBM 3380 DASD device type

L = 9,076

M = 5,000

N = Number of blocks per cylinder x BKS allocation size = 75 x 520 = 39,000

Therefore: ✶ = 1

✸ $= \left(\frac{M}{\left(\frac{L - 8}{20} \downarrow \right)} \uparrow \right) = \left(\frac{5,000}{\left(\frac{9,076 - 8}{20} \downarrow \right)} \uparrow \right) = 12$

✹ $= \left(\frac{M}{L - 12} \uparrow \right) = \left(\frac{5,000}{9,076 - 12} \uparrow \right) = 1$

✵ $= \left(\frac{M}{\left(\frac{L - 8}{84} \downarrow \right)} \uparrow \right) = \left(\frac{5,000}{\left(\frac{9,076 - 8}{84} \downarrow \right)} \uparrow \right) = 47$

$$\boxplus = \left(\frac{N}{L - 12} \uparrow \right) = \left(\frac{39{,}000}{9{,}076 - 12} \uparrow \right) = 5$$

Therefore: Total Number of SCT Blocks = 1 + 12 + 1 + 47 + 5 = 66

SCT Space Requirement = 66 / 75 = 1 cylinders

3.3.6 Space Estimation for Hot Log File

A Hot Log file is used to store A-LOG Management Database record inquiry information and recovery records. The space requirement to write 2 recovery records and *n* record inquiry for one user (TSO or TSS Terminal Facility) is calculated in the following formula:

Hot Log File Space Requirement in Cylinders = Space Requirement for 1 User ✶ x Number of Users

✶ = Space Required for *n* Record Inquiry ✸ + Space Required to Write 2 Recovery Records

$$✸ = \frac{n}{\left\lceil \frac{BLKSIZE - 44}{488} \right\rceil \text{ x } 115 \text{ x (NBLK -}}$$

$$✹ = \frac{2}{\left\lceil \frac{BLKSIZE - 44}{1328} \right\rceil \text{ x (NBLK - 3)}}$$

Where: $\lceil \quad \rceil$: Number of Records per Block

n : Number of Records Inquired

BLKSIZE : Hot Log File Block Size (= 4,096)

NBLK : Number of Blocks per Cylinder (150 for IBM 3380 and 3390, Fujitsu 6425 and Hitachi 8597)

Note: The number of recovery records is fixed and it is specified in a parameter when the Hot Log file is initialized. This number is 2 for A-LOG system.

Example: Estimate the space requirement for the Hot Log file when the number of users is 20 and each user performs inquiry of 10,000 records and writes 2 recovery records.

n = 10,000

BLKSIZE = 4096

NBLK = 150 (DASD device type is IBM 3380)

$$✸ = \frac{10{,}000}{\frac{4{,}096 - 44}{488} \times 115 \times (150 - 3)} = \frac{10{,}000}{8 \times 115 \times 147} = \frac{10{,}000}{135{,}240} = 0.074$$

$$✹ = \frac{2}{\frac{4{,}096 - 44}{1{,}328} \times (150 - 3)} = \frac{2}{3 \times 147} = \frac{2}{441} = 0.0045$$

✶ = 0.074 + 0.0045 = 0.0785

Therefore: Hot Log File Space Requirements (Cylinders) = ✶ x Number of Users

= 0.0785 x 20 = 1.57

Chapter 4

A-LOG System Backup & Maintenance

4.1 Backup of A-LOG System Datasets

It is desirable to perform periodical backup of each DASD and dataset used by A-LOG system. These are described as follows:

1) DASD Backup

   To be able to recover from any destruction of the DASDs containing A-LOG datasets, it is recommended that weekly or monthly backup be performed using relevant OS utilities provided by computer system manufacturer.

2) Dataset Backup

   To be able to recover from any destruction of the A-LOG datasets, it is recommended that weekly or monthly backup be performed using relevant A-LOG system or OS utilities.

The A-LOG system or OS utilities that can be used for backup of A-LOG system datasets are shown in Table 4.1. Dataset not shown in the table need not be backup. Refer to "**5.1.1 BSP-Supplied Utilities**" and "**5.1.2 Manufacturer-Supplied Utilities**" for further details.

Dataset	Dataset Name	Type	Utility for Backup	Reference
A-LOG Management Database	ALOG.DBIDEX ALOG.DBDATA	VSAM	ALOGUNLD	Section 5.15
A-LOG Data Pool	ALOG.DB.SCT ALOG.DB.BKS	DAM	ALOGUBKS	Section 5.14
Security Management Table	ALOG.V0380.SECRTBL	DAM	ALOGFORM	Section 5.6
A-LOG Load Module library	ALOG.V0380.LOAD	PO	*	Section 5.1
A-LOG Installation JCL Library	ALOG.V0380.INSTALL	PO	*	Section 5.1
A-LOG Parameter Library	ALOG.V0380.PARMLIB	PO	*	Section 5.1
A-LOG Terminal Screen Library	ALOG.V0380.MAPLIB	PO	*	Section 5.1
A-LOG PTF Library	ALOG.V0380.ZAPLIB	PO	*	Section 5.1

* The appropriate utility provided by computer manufacturer

Table 4.1: Target Backup Files

4.2 Recovery from Destruction of DASD or Dataset

To recover from destruction of a DASD or dataset used by A-LOG system, execute the following recovery procedure accordingly:

1) Recovery Procedure for Destruction of a DASD

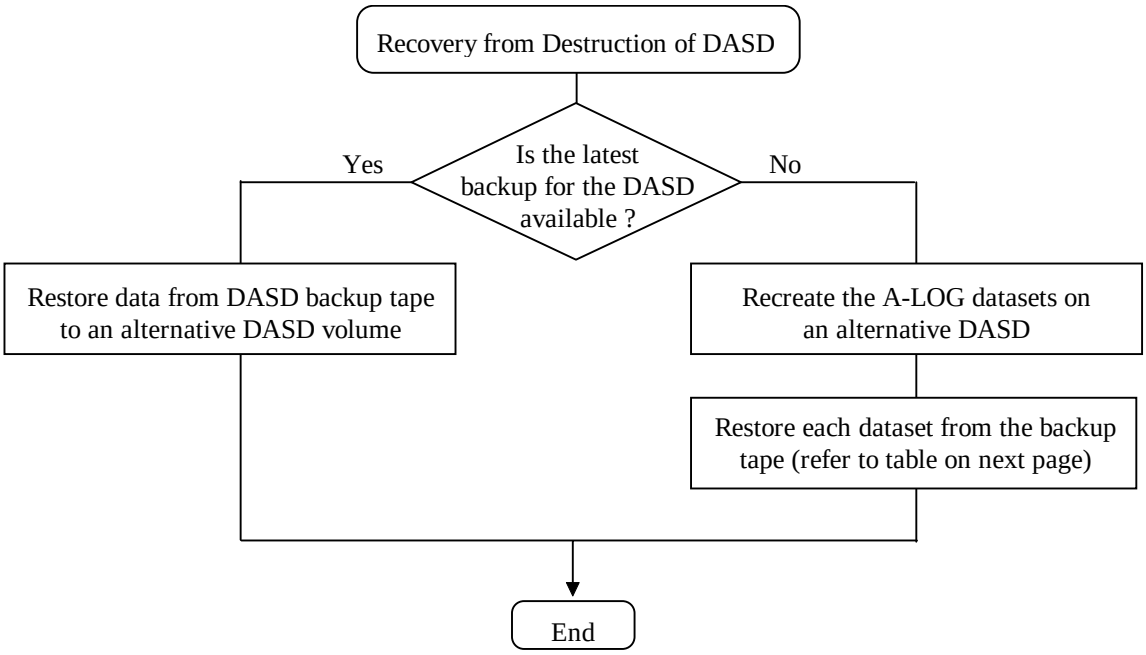


Fig. 4.1: Recovery from Destruction of a DASD

2) Recovery Procedure for Destruction of a Dataset

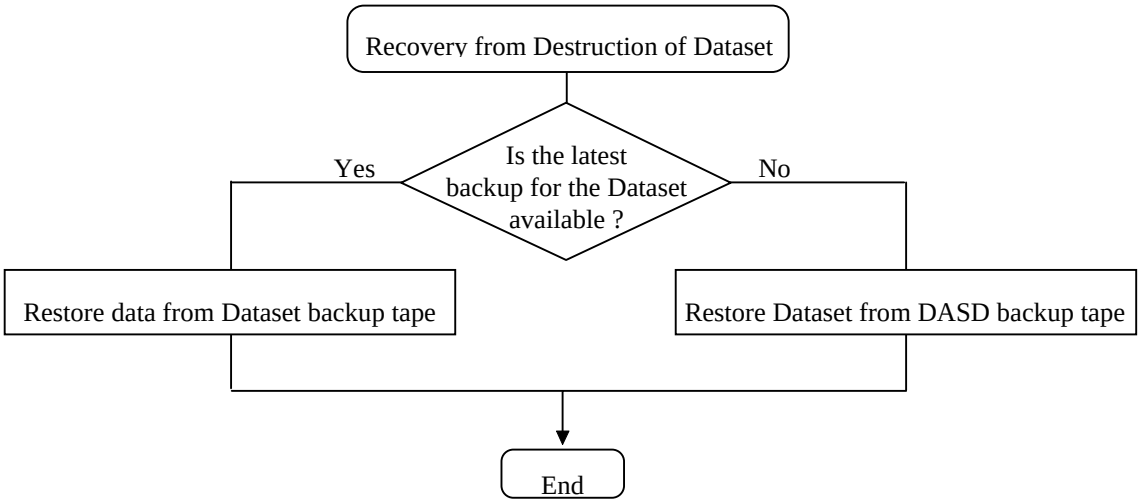


Fig. 4.2: Recovery from Destruction of a Dataset

The A-LOG system or OS utilities that can be used for restore of A-LOG system datasets are shown in Table 4.2. Refer to "**5.1.1 BSP-Supplied Utilities**" and "**5.1.2 Manufacturer-Supplied Utilities**" for further details.

Dataset	**Dataset Name**	**Type**	**Utility for Restore**	**Reference**
A-LOG Management Database	ALOG.DBIDEX ALOG.DBDATA	VSAM	ALOGLOAD	Section 5.11
A-LOG Data Pool	ALOG.DB.SCT ALOG.DB.BKS	DAM	ALOGLBKS	Section 5.10
Security Management Table	ALOG.V0380.SECRTBL	DAM	ALOGDAM0	Section 5.5
A-LOG Load Module library	ALOG.V0380.LOAD	PO	*	Section 5.1
A-LOG Installation JCL Library	ALOG.V0380.INSTALL	PO	*	Section 5.1
A-LOG Parameter Library	ALOG.V0380.PARMLIB	PO	*	Section 5.1
A-LOG Terminal Screen Library	ALOG.V0380.MAPLIB	PO	*	Section 5.1
A-LOG PTF Library	ALOG.V0380.ZAPLIB	PO	*	Section 5.1

* The appropriate utility provided by computer manufacturer

Table 4.2: Target Restore Files

4.3 A-LOG Database Reorganization

4.3.1 A-LOG Database

The A-LOG Database consists of A-LOG Management Database and A-LOG Data Pool. The A-LOG Management Database is made up of two VSAM datasets, which are used to store job log and SYSLOG master records, and jog log and SYSLOG index records. The A-LOG Data Pool is made up of two DAM datasets, which are used to store the actual job log and SYSLOG data.

The A-LOG Management Database and A-LOG Data Pool should be reorganized periodically for the following reasons:

- To remove storage fragmentation
- To improve storage efficiency
- To improve access efficiency

4.3.2 A-LOG Management Database Reorganization

The following table highlights the utilities for reorganizing the A-LOG Management Database:

No.	Utility Name	Function	Input Parameter	Reference
1	ALOGUNLD	Backup files	-	Section 5.15
2	ALOGLOAD	Delete old files Create new files Restore data	DEFIPARM DEFDPARM	Section 5.11

Table 4.3: Reorganization of A-LOG Management Database

4.3.3 A-LOG Data Pool Reorganization

The following table highlights the utilities for reorganizing the A-LOG Data Pool:

No.	Utility Name	Function	Input Parameter	Reference
1	ALOGUBKS	Backup files	-	Section 5.14
2	ALOGIBKS	Initialize files	BKSPARM	Section 5.7
3	ALOGLBKS	Restore data	-	Section 5.10

Table 4.4: Reorganization of A-LOG Data Pool

4.4 A-LOG Load Module Maintenance

An A-LOG load module may need to be modified to correct a fault or to support a specific user environment or requirement. BSP provides modifications to the load modules in the form of PTFs. PTFs are stored in an A-LOG PTF Library (ALOG.V0380.ZAPLIB). Fig. 4.3 shows the naming convention of a PTF.

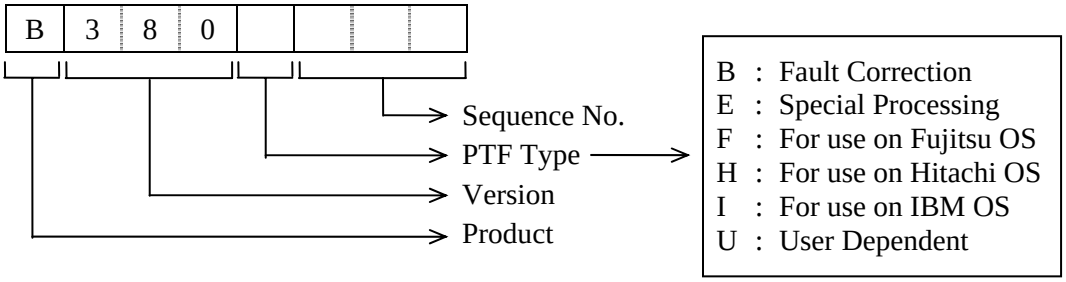


Fig. 4.3: PTF Naming Convention

To apply a PTF supplied by BSP, perform the following procedure:

1) Use the OS utility provided by computer manufacturer to copy the PTFs from the tape supplied by BSP into A-LOG PTF Library as follows:

```
                          { IEBCOPY  }  IBM
//STEP     EXEC PGM=      { JSECOPY  }  Fujitsu
                          { JSDPCPY  }  Hitachi
//SYSPRINT DD   SYSOUT=*
//IN1      DD   DSN=ALOG.V0380.INSTZAP,DISP=SHR,
//             UNIT=TAPE,VOL=SER=BSP001,LABEL=(1,SL)
//OUT1     DD   DSN=ALOG.V0380.ZAPLIB,DISP=SHR
//SYSIN    DD   *
 C I=IN1,O=OUT1
/*
```

2) Backup the A-LOG Load Module Library. Refer to "**4.1 Backup of A-LOG System Dataset**".

3) Use the OS utility provided by computer manufacturer to apply the PTF to the load module as follows:

```
                          { IMASPZAP,PARM=IGNIDRFULL  }  IBM
//STEP     EXEC PGM=      { JQPSPZAP,PARM=IGNIDRFULL  }  Fujitsu
                          { JSPPTCH                   }  Hitachi
//SYSPRINT DD   SYSOUT=*
//SYSLIB   DD   DSN=ALOG.V380.LOAD,DISP=SHR,
//SYSIN    DD   DSN=ALOG.V380.ZAPLIB(member-name),DISP=SHR
```

4) Verify the operation.

4.5 Consistency Check for A-LOG Management Database

A-LOG Management Database consists of two VSAM datasets, one for storing indices of master records and the other for storing the data of master records. The index and data records are inter-related. It is possible that the relationship among some of these records may become invalid due to one of the following circumstances:

- A-LOG Monitor was abnormally terminated
- A-LOG Management Database is full
- A-LOG Management Database was corrupted due to system error

When relationships among the records become invalid, the following situations may occur:

- Inquiries for job log or SYSLOG using A-LOG Terminal Facility cannot be performed properly.
- Job log or SYSLOG records cannot be archived to tape.

In the event of inconsistency in the relationships among the records, the ALOGICHK utility should be executed to check for errors and to list the number of unmatched records. If any unmatched records are found, the ALOGRENO utility must be executed to delete the unmatched records and regenerate the A-LOG Management Database. Refer to **Chapter 5: A-LOG System Maintenance Utilities** for further details.

The procedure for checking inconsistency of A-LOG Management Database is shown as follows:

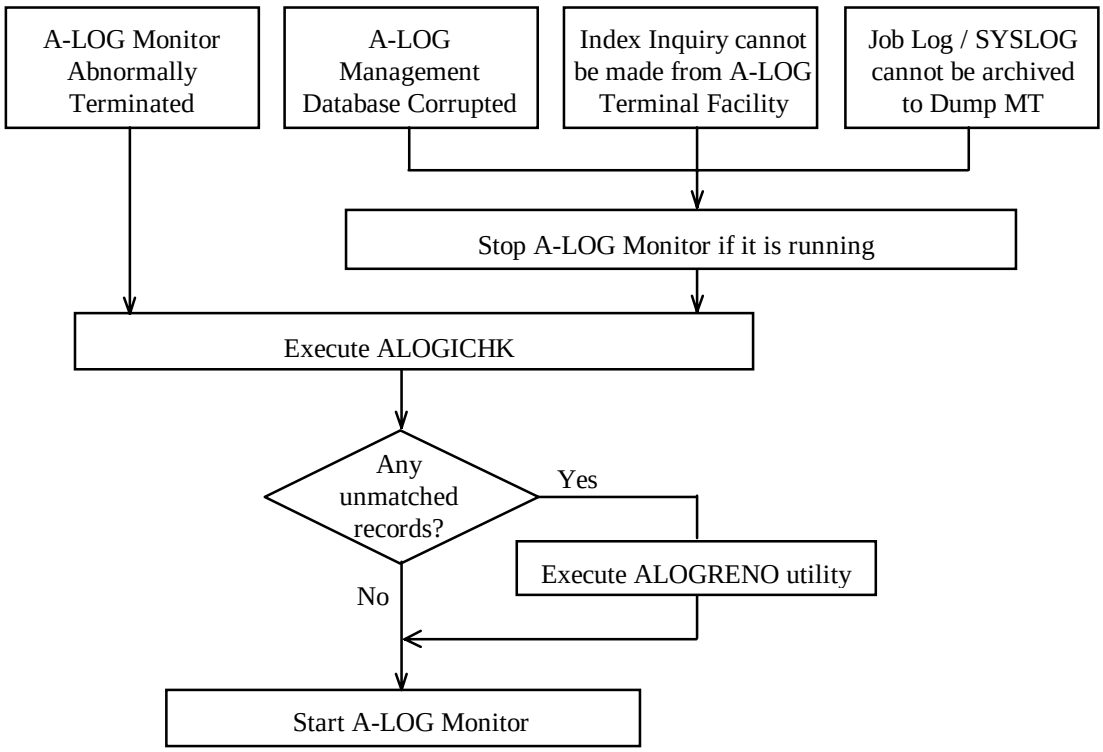


Fig. 4.4: Procedure for Checking Inconsistency

4.6 A-LOG Database & System Library Expansion

4.6.1 Expandable Components of A-LOG Database & System Libraries

The following components of A-LOG database and system libraries can be expanded when necessary:

- A-LOG Management Database
- A-LOG Data Pool
- Hot Log File
- Security Management Table
- Security Log File
- A-LOG Load Module Library
- A-LOG Installation JCL Library
- A-LOG Parameter Library
- A-LOG PTF Library

4.6.2 A-LOG Management Database Expansion

To expand the A-LOG Management Database, perform the procedure for reorganizing A-LOG Management Database using the utilities as shown in the following table:

Dataset Type : VSAM Dataset Name: ALOG.DBIDEX and ALOG.DBDATA				
No.	**Utility Name**	**Function**	**Input Parameter Member**	**Reference**
1	ALOGUNLD	Backup A-LOG Management Database	-	Section 5.15
2	ALOGLOAD	Delete old A-LOG Management Database Create new A-LOG Management Database Restore data	DEFIPARM DEFDPARM	Section 5.11

Table 4.5: Expansion of A-LOG Management Database

Note: Members DEFIPARM and DEFDPARM of A-LOG Parameter Library must be updated with the new space requirements to be allocated.

4.6.3 A-LOG Data Pool Expansion

To expand the A-LOG Data Pool, perform the procedure using the utilities as shown in the following table:

Dataset Type : DAM Dataset Name: ALOG.DB.SCT and ALOG.DB.BKS				
No.	**Utility Name**	**Function**	**Input Parameter Member**	**Reference**
1	ALOGUBKS	Backup A-LOG Data Pool	-	Section 5.14
2	Manufacturer-Supplied	Delete old A-LOG Data Pool	-	Section 5.1
3	DUMFORM0	Create and format new A-LOG Data Pool	See Note	Section 5.16
4	ALOGIBKS	Initialize A-LOG Data Pool	BKSPARM	Section 5.7
5	ALOGLBKS	Restore A-LOG Data Pool	-	Section 5.10

Table 4.6: Expansion of A-LOG Data Pool

Note: Change the content of format parameter when using multi-volume dataset.

4.6.4 Hot Log File Expansion

To expand the Hot Log File, perform the procedure using the utilities as shown in the following table:

Dataset Type : DAM Dataset Name: ALOG.HOTLOG				
No.	**Utility Name**	**Function**	**Input Parameter Member**	**Reference**
1	Manufacturer-Supplied	Delete Hot Log File	-	Section 5.1
2	ABSIHIN	Create and format new Hot Log File	HLGPARM	Section 5.2

Table 4.7: Expansion of Hog Log File

4.6.5 Security Management Table Expansion

To expand the Security Management Table, perform the procedure using the utilities as shown in the following table:

Dataset Type : DAM Dataset Name: ALOG.V0380.SECRTBL				
No.	**Utility Name**	**Function**	**Input Parameter Member**	**Reference**
1	ALOGFORM	Backup Security Management Table	-	Section 5.6
2	Manufacturer-Supplied	Delete old Security Management Table	-	Section 5.1
3	DUMFORM0	Create and format new Security Management Table	-	Section 5.16
4	ALOGDAM0	Restore Security Management Table	-	Section 5.5

Table 4.8: Expansion of Security Management Table

4.6.6 Security Log File Expansion

To expand the Security Log File, perform the procedure using the utilities as shown in the following table:

Dataset Type : DAM Dataset Name: ALOG.V0380.SECRLOGX and ALOG.V0380.SECRLOGY				
No.	**Utility Name**	**Function**	**Input Parameter Member**	**Reference**
1	Manufacturer-Supplied	Delete Security Log File	-	Section 5.1
2	DUMFORM0	Create and format new Security Log File	-	Section 5.16

Table 4.9: Expansion of Security Log File

4.6.7 Expansion of A-LOG System Libraries

To expand any of the A-LOG system libraries, perform the procedure using the utilities as shown in the following table:

Dataset Type : Partitioned Dataset (PO)
Dataset Name: ALOG.V0380.LOAD (A-LOG Load Module Library)
ALOG.V0380.INSTALL (A-LOG Installation JCL Library)
ALOG.V0380.PARMLIB (A-LOG Parameter Library)
ALOG.V0380.ZAPLIB (A-LOG PTF Library)

No.	Utility Name	Function	Input Parameter Member	Reference
1	Manufacturer-Supplied	Back up the partitioned dataset	-	Section 5.1
2	Manufacturer-Supplied	Delete the old partitioned dataset	-	Section 5.1
3	Manufacturer-Supplied	Create and format the new partitioned dataset	-	Section 5.1
4	Manufacturer-Supplied	Restore the partitioned dataset	-	Section 5.1

Table 4.10: Expansion of A-LOG System Libraries

Chapter 5

A-LOG System Maintenance Utilities

5.1 A-LOG System Maintenance Utilities

5.1.1 BSP-Supplied Utilities

The following utilities are provided by BSP for maintenance of A-LOG system:

Utility Name	Utility Function	Reference
ABSIHIN	To create and initialize the Hot Log File.	Section 5.2
ALOGCPY1	To backup files in A-LOG Management Database individually.	Section 5.3
ALOGCPY2	To restore files backed up by ALOGCPY1 utility into the A-LOG Management Database.	Section 5.4
ALOGDAM0	To restore data into the Security Management Table (DAM).	Section 5.5
ALOGFORM	To backup the Security Management Table (DAM).	Section 5.6
ALOGIBKS	To initialize the A-LOG Data Pool (DAM).	Section 5.7
ALOGICHK	To perform consistency check among records in A-LOG Management Database.	Section 5.8
ALOGIUPD	To recreate A-LOG Management Database (VSAM) using the intermediate files generated by ALOGICHK utility. This utility is incorporated into the procedure for ALOGRENO utility. Therefore, it should not be executed alone.	Section 5.9
ALOGLBKS	To restore the A-LOG Data Pool (DAM).	Section 5.10
ALOGLOAD	To recreate the A-LOG Management Database (VSAM) and restore its data.	Section 5.11
ALOGRENO	To remove unmatched records in A-LOG Management Database (DAM).	Section 5.12
ALOGREP	To generate a report showing the storage utilization status of A-LOG Management Database (VSAM) and A-LOG Data Pool (DAM).	Section 5.13
ALOGUBKS	To backup the A-LOG Data Pool (DAM).	Section 5.14
ALOGUNLD	To backup the A-LOG Management Database.	Section 5.15
DUMFORM0	To create and format DAM datasets such as A-LOG Data Pool, Security Management Table and Security Log File.	Section 5.16

Table 5.1: A-LOG System Maintenance Utilities

5.1.2 Manufacturer-Supplied Utilities

The following utilities provided by computer manufacturer are used for maintenance of A-LOG system:

	Manufacturer		
Usage	**IBM**	**Fujitsu**	**Hitachi**
File Creation and Deletion	IEFBR14	KDJBR14	JDJDUMMY
Copy PO (Partitioned) Files	IEBCOPY	JSECOPY	JSDPCPY
Copy PS (Sequential) Files	IEBGENER	JSEGENER	JSDSCPY
VSAM File Maintenance	IDCAMS	KQCAMS	JSCVSUT
Apply Zap to Load Module	IMASPZAP	JQPSPZAP	JSPPTCH

Table 5.2: Manufacturer-Supplied Utilities

Refer to the respective system manuals of the computer manufacturer when using any of the above utilities.

5.2 ABSIHIN Utility

Function

This utility creates a Hot Log File and initializes it according to the parameters specified in member HLGPARM of A-LOG Parameter Library.

Processing Outline

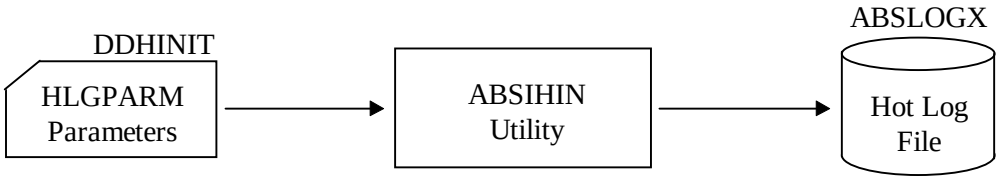


Fig. 5.1: ABSIHIN Processing Outline

Execution Procedure

1) Specify the volume serial number and space requirement for the Hot Log File in ABSLOGX DD statement as follows:

```
//ABSLOGX DD DSN=ALOG.HOTLOG,DISP=(,CATLG,DELETE),
            UNIT=SYSDA,VOL=SER=volser,                    ✶
            SPACE=(CYL,(size)),DCB=(DSORG=DA)             ✸
```

Notes: ✶ Specify the volume serial number of the Hot Log File.

✸ Specify the space requirements in cylinder for the Hot Log File to be created.

2) Specify the parameter values in member HLGPARM.

```
DATASPACE NAME=HOTLOG,SYSID=ABAS,BLKCOUNT=total-blocks,     ✶
          BLKSIZE=block-size,                               ✸
FILE NAME=RCV,BLKCOUNT=recovery-blocks                      ✹
FILE NAME=SET,BLKCOUNT=retrieval-blocks                     ❋
FILE NAME=USER,BLKCOUNT=1
```

Notes: ✶ Specify total number of blocks. Always specify a value less than the number of formatted blocks.

- Total Blocks = Recovery Blocks + Retrieval Blocks + 3
- Formatted Blocks = Number of Blocks per Cylinder* x Number of Cylinders Allocated

* 150 for IBM3380/3390, Fujitsu 6425 and Hitachi 8597

✸ Specify 4,096 for the *block-size*.

✹ Specify 2 for the number of *recovery-blocks*.

❋ Specify the number of *retrieval-blocks*.

- Retrieval Blocks = $\dfrac{\text{Number of Records Searched}}{\dfrac{\text{BLKSIZE - 44}}{488} \times 115}$

3) Execute ABSIHIN utility. The JCLs for executing this utility is provided as a member in A-LOG Installation JCL Library. Sample execution JCLs are shown as follows:

```
//ABSIHIN  JOB ALOG,ALOG,MSGCLASS=A,CLASS=A
//*-------------------------------------------------------*
//*   FORMAT AND INITIALIZE HOTLOG FILE                   *
//*-------------------------------------------------------*
//STEP010  EXEC PGM=ABSIHIN,REGION=512K
//STEPCAT  DD   DSN=ALOG.UCAT,DISP=SHR
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//ABSLOGX  DD   DSN=ALOG.HOTLOG,DISP=(,CATLG,DELETE),     ✶
//              UNIT=SYSDA,VOL=SER=BSP001,
//              SPACE=(CYL,(3)),DCB=(DSORG=DA)
//DDHINIT  DD   DSN=ALOG.V0380.PARMLIB(HLGPARM),DISP=SHR  ✸
//SYSUDUMP DD   SYSOUT=*
```

✶ Hot Log File

✸ Member HLGPARM

5.3 ALOGCPY1 Utility

Function

This utility backups files in A-LOG Management Database individually.

Processing Outline

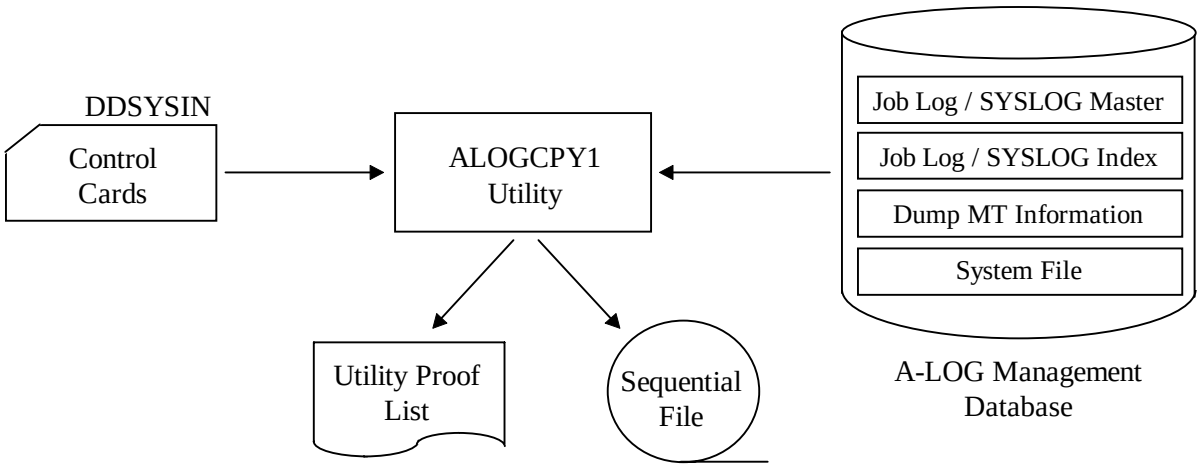


Fig. 5.2: ALOGCPY1 Processing Outline

Execution Procedure

1) Start A-LOG Monitor if it has not been started.

2) Execute ALOGCPY1 utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

A sample proof list is shown in the following page.

ALOGCPY1 Proof List

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.

                                                                    DATE 96/12/15 TIME 10:32:25    PAGE    1

    ✶ SUL443I  JAM FILE WRITE COUNT =       356                      LIST CODE : CPY100
```

Note: ✶ Number of records written to backup file

5.4 ALOGCPY2 Utility

Function

This utility restores the file backup by ALOGCPY1 utility into the A-LOG Management Database. Before restoring the data, no record must exist in the files.

Processing Outline

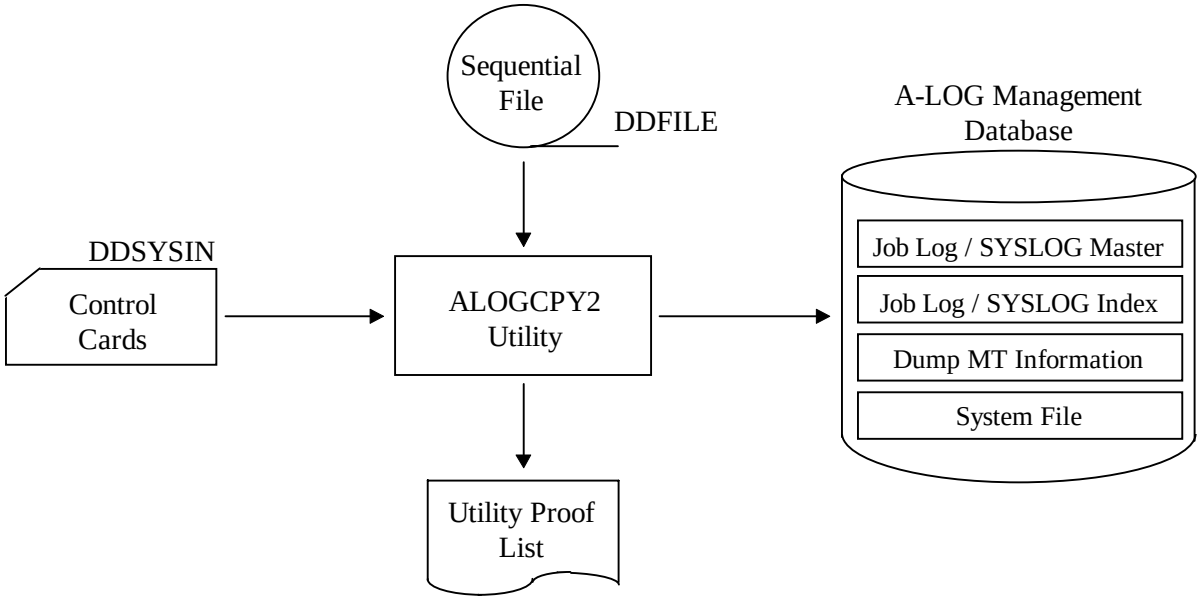


Fig. 5.3: ALOGCPY2 Processing Outline

Execution Procedure

1) Start A-LOG Monitor if it has not been started.

2) Execute ALOGCPY2 utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

A sample proof list is shown in the following page.

ALOGCPY2 Proof List

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.

                                                                    DATE 96/12/15 TIME 10:54:10    PAGE    1

    ✶ SUL449I  JAM FILE WRITE COUNT =      356                       LIST CODE : CPY200
```

Note: ✶ Number of records restored from backup file

5.5 ALOGDAM0 Utility

Function

This utility restores the Security Management Table from a backup file.

Processing Outline

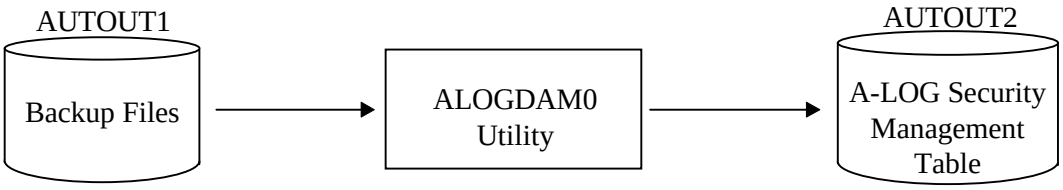


Fig. 5.4: ALOGDAM0 Processing Outline

Execution Procedure

1) Specify the dataset name of backup file in AUTOUT1 DD statement of the execution JCLs as follows:

```
//AUTOUT1  DD DSN=dataset-name,DISP=SHR
```

2) Specify the dataset name of Security Management Table in AUTOUT2 DD as follows:

```
//AUTOUT2  DD DSN=dataset-name,DISP=SHR
```

3) Execute ALOGDAM0 utility. The JCLs for executing this utility is provided as a member in A-LOG Installation JCL Library. Sample execution JCLs are shown as follows:

```
//ALOGDAM0 JOB ,ALOG,MSGCLASS=A,CLASS=A
//*-------------------------------------------------*
//* FUNCTION : LOAD BACKUP SECURITY FILE TO         *
//*            A-LOG SECURITY FILE (DAM FILE)       *
//* AUTOUT1  : PS BACKUP FILE                       *
//* AUTOUT2  : A-LOG SECURITY FILE                  *
//*-------------------------------------------------*
//STEP0010 EXEC PGM=AUTODAMO,PARM=SEC
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//STEPCAT  DD   DSN=ALOG.UCAT,DISP=SHR
//AUTOUT1  DD   DSN=BKUP.SECRTBL,DISP=SHR          ✶
//AUTOUT2  DD   DSN=ALOG.V0380.SECRTBL,DISP=SHR    ✸
```

✶ Backup File

✸ Security Management Table

5.6 ALOGFORM Utility

Function

This utility backups the Security Management Table.

Processing Outline

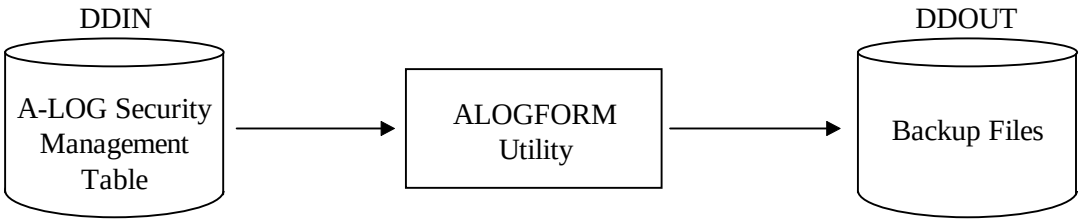


Fig. 5.5: ALOGFORM Processing Outline

Execution Procedure

1) Specify dataset name of the Security Management Table in DDIN DD statement of the execution JCLs as follows:

```
//DDIN  DD DSN=dataset-name,DISP=SHR
```

2) Specify dataset name of the backup file in DDOUT DD statement as follows:

```
//DDOUT DD DSN=dataset-name,DISP=(,KEEP),
          SPACE=(CYL,(primary,secondary)),UNIT=unit-name,
          VOL=SER=volser,
          DCB=(BLKSIZE=10240,RECFM=FB,LRECL=1024)
```

3) Execute ALOGFORM utility. The JCLs for executing this utility is provided as a member in A-LOG Installation JCL Library. Sample execution JCLs are shown as follows:

```
//ALOGFORM JOB ,ALOG,CLASS=A,MSGCLASS=A
//*-----------------------------------------------------*
//*-----------------------------------------------------*
//* FUNCTION : TO BACKUP A-LOG SECURITY FILE TO PS FILE *
//*-----------------------------------------------------*
//AUTOFORM EXEC PGM=AUTOFORM,PARM=S
//STEPCAT  DD   DSN=ALOG.UCAT,DISP=SHR
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDIN     DD   DSN=ALOG.SECRTBL,DISP=SHR                ✶
//DDOUT    DD   DSN=ALOG.BACKUP.SECRTBL,DISP=(,KEEP),    ✸
//              SPACE=(CYL,(2,1)),UNIT=SYSDA,VOL=SER=BSP001,
//              DCB=(BLKSIZE=10240,RECFM=FB,LRECL=1024)
//SYSIN    DD   DUMMY
```

✶ Security Management Table

✸ Backup File

5.7 ALOGIBKS Utility

Function

This utility initializes the A-LOG Data Pool according to the parameter values specified in member BKSPARM of A-SPOOL Parameter Library. The initialization process clears all jog log and SYSLOG records in A-LOG Data Pool. Therefore, it is important to run ALOGUBKS utility to backup the A-LOG Data Pool and verify that it has ended normally before executing this utility.

Processing Outline

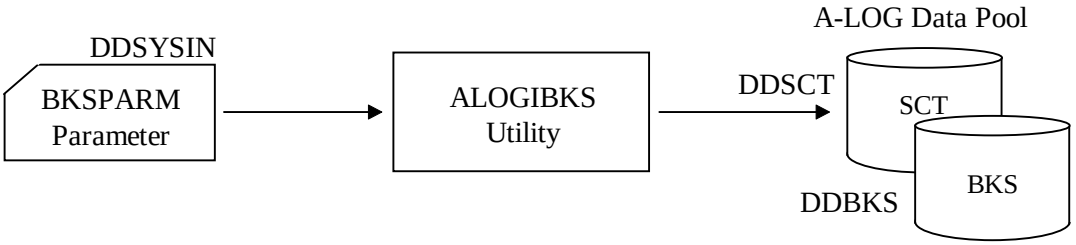


Fig. 5.6: ALOGIBKS Processing Outline

Execution Procedure

1) Modify the parameter values in member BKSPARM as required.

```
BKSIOS SCTDEV=device-type
BKSIOS SCTSIZE=cylinder-size
BKSIOS EXTBLKC=number-of-BKS-blocks-per-extent
BKSIOS MAXREPC=maximum-number-of-reports
BKSIOS BKSDEV=device-type
BKSIOS BKSSIZE=cylinder-size
```

Notes: ✶ Specify the device type of the SCT component of A-LOG Data Pool (ALOG.DB.SCT). The device types supported are shown as follows:

IBM	Fujitsu	Hitachi
3350	493	8595
3380	6421	8597
3390	6425	8598
9345		

✸ Specify the allocation size in cylinder for the SCT component of A-LOG Data Pool (ALOG.DB.SCT).

✹ Specify the number of BKS blocks to be allocated to one extent. Default is 10. Usually, this default value need not be changed.

❋ Specify the maximum number of job logs that can be stored in A-LOG Data Pool. Default is 5,000. Before expanding the A-LOG Data Pool, refer to "**3.3.4 Estimation for MAXREPC Parameter of Member BKSPARM**" to determine the optimum value.

⌖ Specify the device type of the BKS component of A-LOG Data Pool (ALOG.DB.BKS). The device types supported are the same as those mentioned in ✶.

⌖ Specify the allocation size in cylinders for the BKS component of A-LOG Data Pool (ALOG.DB.BKS).

2) Execute ALOGIBKS utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

5.8 ALOGICHK Utility

Function

This utility performs consistency check among the records in A-LOG Management Database. It contains the following steps:

1) STEP0010: Verify the A-LOG Management Database
2) STEP0020: Backup the A-LOG Management Database
3) STEP0030: Sort index portion of the backup file
4) STEP0040: Renumber and create intermediate index and data files without unmatched records

STEP0040 uses a BSP-supplied utility, other steps use manufacturer-supplied utilities.

Processing Outline

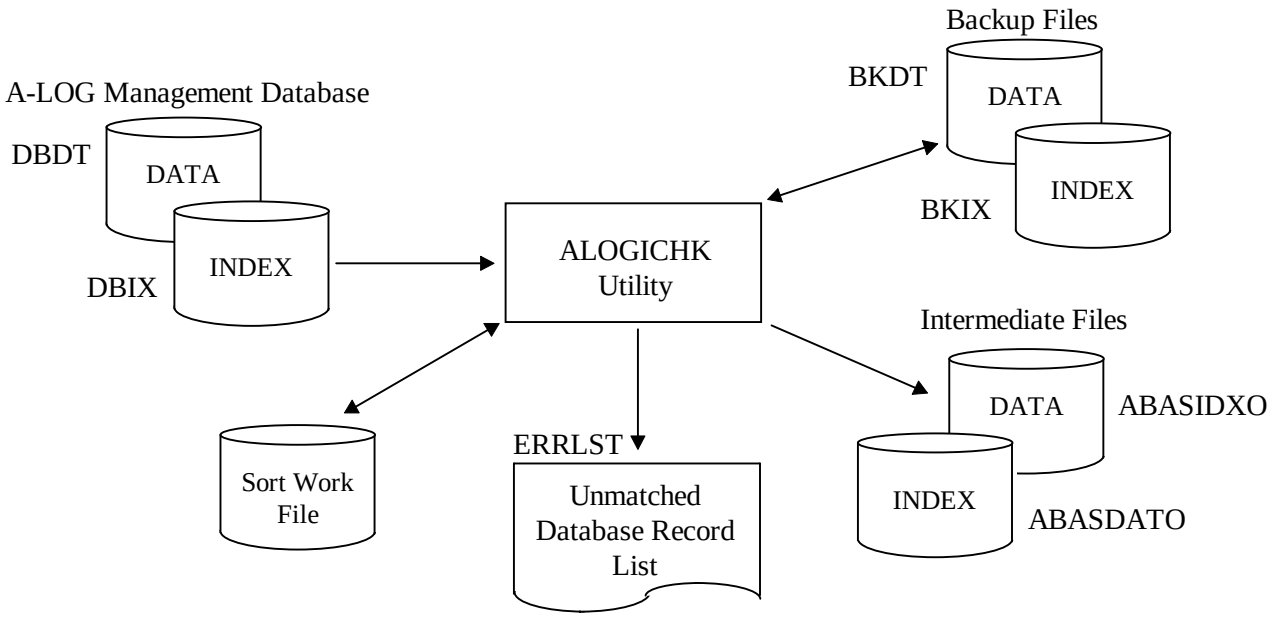


Fig. 5.7: ALOGICHK Processing Outline

Execution Procedure

1) Specify the program name of the manufacturer-supplied utility, the volume serial number and sizes for the backup, intermediate and sort work files in the EXEC parameter as follows:

```
ALOGICHK,UTL=program-name,                          ✶
         WVOL=volser,                               ✹
         ISPC='CYL,(primary,secondary)',            ✹
         DSPC='CYL,(primary,secondary)',            ❋
         WSPC='CYL,(primary,secondary)'             ⌖
```

Notes: ✶ Specify one of the following manufacturer-supplied program names:

- IBM: IDCAMS
- Fujitsu: KQCAMS
- Hitachi: JSCVSUT

✹ Specify the volume serial number for the backup and intermediate file.

✹ Specify the size for the backup and intermediate index files.

❋ Specify the size for the backup and intermediate data files.

⌖ Specify the size for the sort work files.

2) Execute ALOGICHK utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

An Unmatched Database Records List is shown in the following page.

Unmatched Database Records List

```
                ********** A-BAS DATA BASE RECORD UNMATCH LIST **********    96/12/15(10:38:57) PAGE : 0001

UNMATCH KEY

FILE NO. ✶         RECORD NO. ✸                          KIND OF RECORD ✹
     ----   ---------------                              ----------------------
<HEX MODE>         <HEX MODE>
<DEC MODE>         <DEC MODE>

     0014           00000002
       20                  2                            INDEX RECORD NOT FOUND
     0046           00000001
       70                  1                            DATA  RECORD NOT FOUND
     0046           00000002
       70                  2                            INDEX RECORD NOT FOUND
     0046           00000006
       70                  6                            INDEX RECORD NOT FOUND
     0050           00000001
       80                  1                            INDEX RECORD NOT FOUND
     0050           00000002
       80                  2                            INDEX RECORD NOT FOUND
     0050           00000009
       80                  9                            INDEX RECORD NOT FOUND
```

Notes:

✶ File Number (2 lines per unmatched record):
- 1st line: Hexadecimal
- 2nd line: Decimal

✸ Record Number (2 lines per unmatched record):
- 1st line: Hexadecimal
- 2nd line: Decimal

✹ Type of Unmatched Record

5.9 ALOGIUPD Utility

Function

This utility recreates the A-LOG Management Database using the intermediate files generated by ALOGICHK utility. This utility is incorporated into the procedure for ALOGRENO utility. Therefore, it should not be executed by itself. Refer to "**5.12 ALOGRENO Utility**" for further details.

This utility contains the following steps:

1) STEP0050: Sort intermediate index file
2) STEP0060: Sort intermediate data file
3) STEP0070: Delete existing A-LOG Management Database
4) STEP0080: Create new A-LOG Management Database
5) STEP0090: Restore A-LOG Management Database from intermediate index and data files

Processing Outline

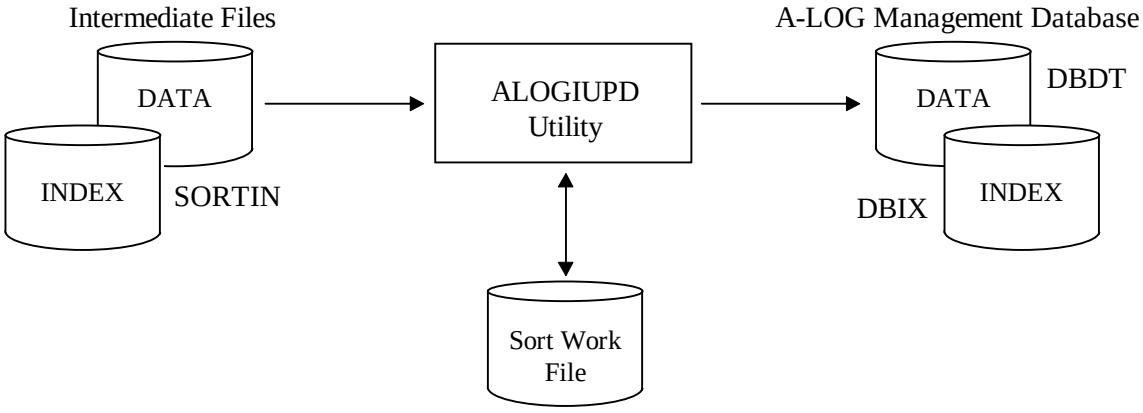


Fig. 5.8: ALOGIUPD Processing Outline

Execution Procedure

Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs. Do not execute this procedure by itself.

5.10 ALOGLBKS Utility

Function

This utility restores the data backuped by ALOGUBKS utility into A-LOG Data Pool.

Processing Outline

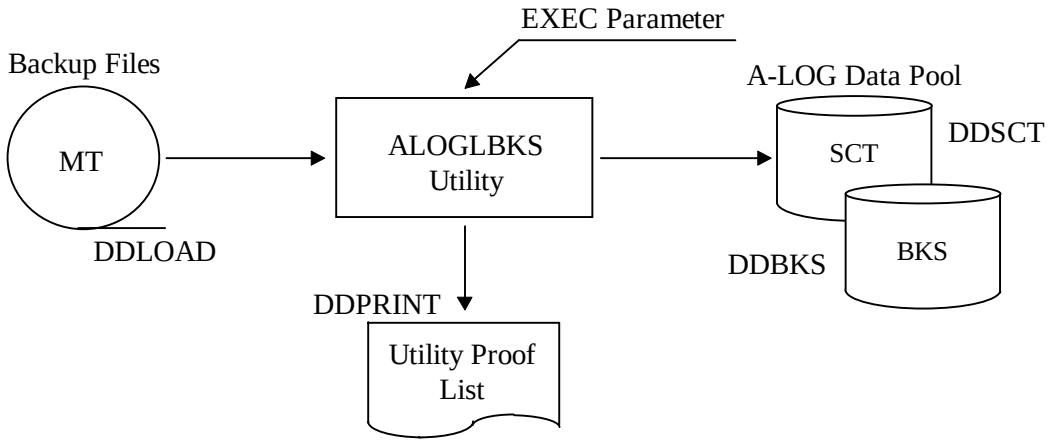


Fig. 5.9: ALOGLBKS Processing Outline

Execution Procedure

1) Execute ALOGIBKS utility to initialize A-LOG Data Pool before executing the ALOGLBKS utility. Refer to "**5.7 ALOGIBKS utility**" for further details.

2) Modify the size of the memory storage area for SCT blocks as required.

```
PARM='SIZE=area-sizeK,TYPE=ALOG'
```

Specifies the size of the memory storage area for the SCT control blocks (extent index, extent status, extent and status). The size is specified in Kbytes. Default value is 2,000K. Minimum value is 20K. The optimum value may be calculated as follows:

```
Optimum Value =  ( Number of Extent Index Blocks + Number of Extent Blocks ) x
                 ( Block Size of SCT File + 4 )
```

Note: Number of Extent Index Blocks and Number of Extent Blocks can be obtained from the reports generated by ALOGREP utility.

Example: Determine the optimum value with the following assumptions:

- Number of Extent Index Blocks : 40
- Number of Extent Blocks : 170
- Block Size of SCT File : 10,796

```
Optimum Value = ( 40 + 170 ) x ( 10,796 + 4 ) = 2,268,000 bytes
```

Therefore:

```
PARM='SIZE=2268K',TYPE=ALOG
```

3) Specify the volume serial number of the backup file in EXEC parameter.

```
ALOGLBKS,BKVOL=volser
```

Specify the volume serial number for the output backup file. If required, modify the DDUNLD DD statement of the execution JCLs, such as the dataset name and UNIT of the output backup dataset.

4) Execute ALOGLBKS utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

A sample proof list is shown in the following page.

ALOGLBKS Proof List

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.

                                                                    DATE 96/12/15 TIME 10:40:53    PAGE    1
A-LOG LOAD UTILITY :

    ✶  WRITE REPORT COUNT =  00000042
    ✸  WRITE PAGE   COUNT =  00002139

    ✹  LOAD VOLUME  = BSPOO1
    ❊  LOAD DATASET = ALOG.DB.BKS

A-LOG LOAD UTILITY NORMAL END :
```

Notes:
- ✶ Number of job logs or SYSLOGs loaded
- ✸ Total number of pages loaded
- ✹ Volume serial number in DDLOAD DD statement
- ❊ Dataset name in DDLOAD DD statement

5.11 ALOGLOAD Utility

Function

This utility recreates the A-LOG Management Database using the backup files created by ALOGUNLD utility. It restores the backup data into A-LOG Management Database. It contains the following steps:

1) DELIX : Delete existing index portion of A-LOG Management Database (ALOG.DBIDEX)
2) DELDT : Delete existing data portion of A-LOG Management Database (ALOG.DBDATA)
3) DEFIX : Create new index portion of new A-LOG Management Database
4) DEFDT : Create new data portion of new A-LOG Management Database
5) LOADIX : Restore backup data to the index portion of new A-LOG Management Database
6) LOADDT : Restore backup data to the data portion of new A-LOG Management Database

Processing Outline

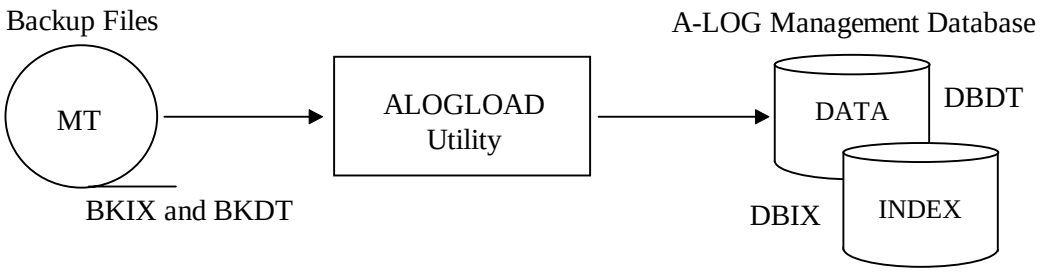


Fig. 5.10: ALOGLOAD Processing Outline

Execution Procedure

1) Modify the SYSIN parameter if required. If the space requirements for A-LOG Management Database are changed, modify the values in member DEFIPARM of step DEFIX and member DEFDPARM of step DEFDT. Refer to OS utility manual regarding the specification method.

2) Specify the program name of the manufacturer-supplied utility, and the volume serial numbers of the backup file and A-LOG Management Database in the EXEC parameter.

```
ALOGLOAD,UTL=program-name,      ✶
         BKVOL=volser,          ✸
         DVOI=volser,           ✹
         DVOD=volser            ❋
```

Notes: ✶ Specify one of the following manufacturer-supplied program names:

- IBM: IDCAMS
- Fujitsu: KQCAMS
- Hitachi: JSCVSUT

✸ Specify the volume serial number for the backup file.

✹ Specify the volume serial number for the index portion of new A-LOG Management Database.

❋ Specify the volume serial number for the data portion of new A-LOG Management Database.

3) Execute ALOGLOAD utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

5.12 ALOGRENO Utility

Function

This utility performs consistency check among the records in A-LOG Management Database. Unmatched records are removed and the A-LOG Management Database is then recreated. This utility is made up of ALOGICHK and ALOGIUPD utilities. Therefore, it contains the following steps from each of the utilities:

1) ALOGICHK Utility
   - STEP0010: Verify the A-LOG Management Database
   - STEP0020: Backup the A-LOG Management Database
   - STEP0030: Sort index portion of backup file.
   - STEP0040: Renumber and create intermediate index and data files without unmatched records.

2) ALOGIUPD Utility
   - STEP0050: Sort intermediate index file.
   - STEP0060: Sort intermediate data file.
   - STEP0070: Delete existing A-LOG Management Database
   - STEP0080: Create new A-LOG Management Database
   - STEP0090: Restore A-LOG Management Database from intermediate index and data files.

Processing Outline

Refers to Fig. 5.11 in the following page.

Execution Procedure

1) Specify the program name of the manufacturer-supplied utility, the volume serial number and sizes for the backup, intermediate and sort work files in the EXEC parameter as follows:

```
ALOGRENO,UTL=program-name,                          ✶
         DVOL=volser,                               ✸
         WVOL=volser,                               ✹
         ISPC='CYL,(primary,secondary)',            ❋
         DSPC='CYL,(primary,secondary)',            ⌖
         WSPC='CYL,(primary,secondary)'             ⌖
```

Notes: ✶ Specify one of the following manufacturer-supplied program names:
- IBM: IDCAMS
- Fujitsu: KQCAMS
- Hitachi: JSCVSUT

- Specify the volume serial number for the A-LOG Management Database.
- Specify the volume serial number for the backup and intermediate files.
- Specify the size for the backup and intermediate index files.
- Specify the size for the backup and intermediate data files.
- Specify the size for the sort work files.

2) Execute ALOGRENO utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

Refer to the output listing generated by ALOGICHK utility in "**5.8 ALOGICHK Utility**"".

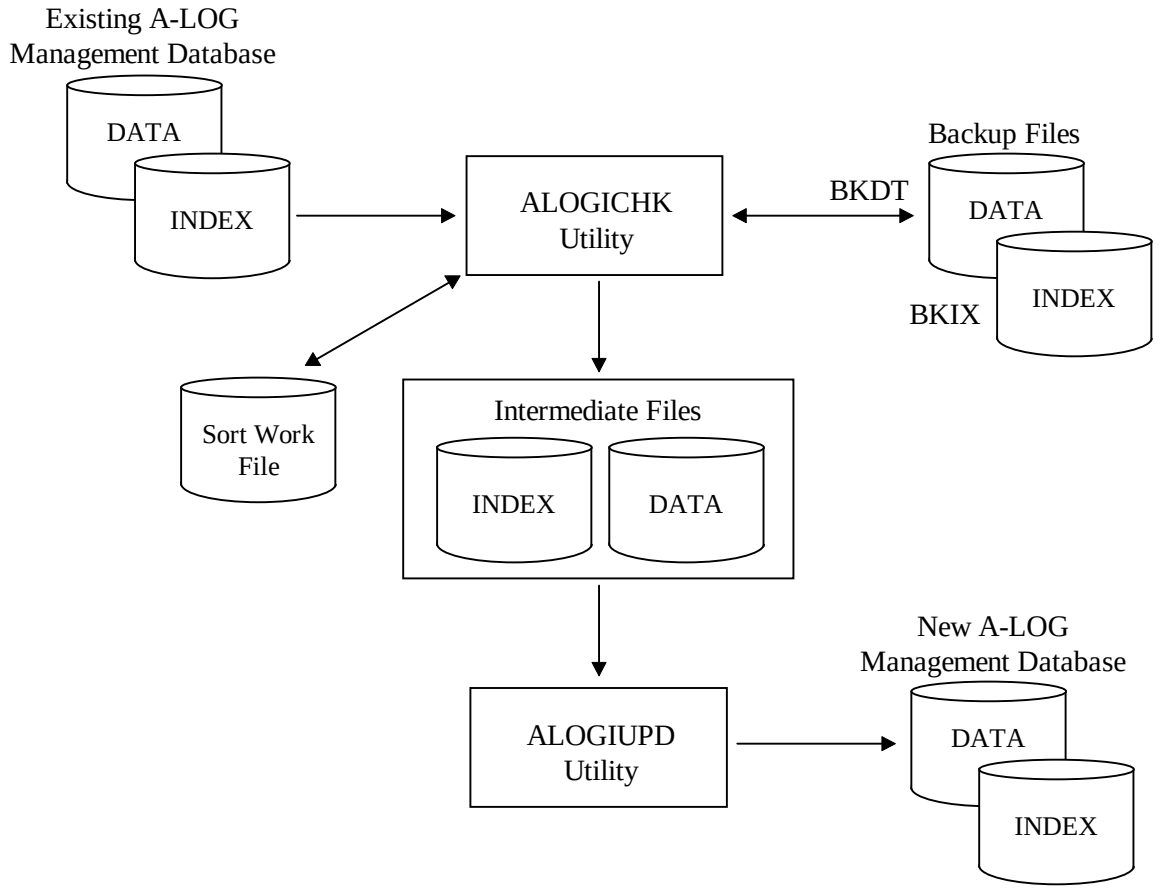


Fig. 5.11: ALOGRENO Processing Outline

5.13 ALOGREP Utility

Function

This utility generates a report showing the storage utilization of A-LOG Management Database and A-LOG Data Pool.

Processing Outline

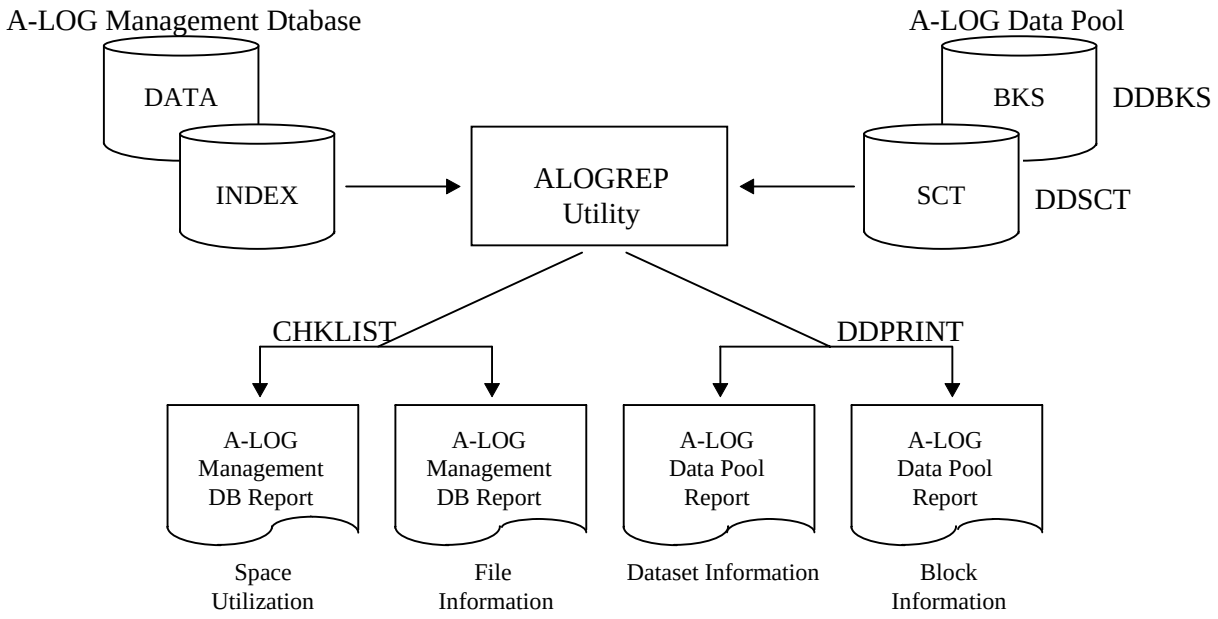


Fig. 5.12: ALOGREP Processing Outline

Execution Procedure

Execute ALOGREP utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listings

Samples of the following output listings with explanations are provided in the next few pages:

- A-LOG Management Database Report (Space Utilization)
- A-LOG Management Database Report (File Information)
- A-LOG Data Pool Report (Dataset Information)
- A-LOG Data Pool Report (Block Information)

A-LOG Management Database Report (Space Utilization)

```
     A-BAS                              * CONTENTS OF A-BAS DATABASE *                  96/12/15(17:08:45)     PAGE :  1
**********************************
*   DATABASE STATUS INFORMATION    *
**********************************

<<< INDEX PART INFORMATION >>>
     ✶              ✸              ✹              ❋              ⌗              ⌖              ✧              ⌑              ⌦
   USED(%)    SPACE-K-BYTES TOTAL-K-BYTES    CI-SIZE    SPACE-CI-CNT  TOTAL-CI-CNT   SPLIT-CI-CNT  EXTENT-COUNT  SPLIT-CA-CNT
------------ ------------- ------------- ------------ ------------- ------------- ------------- ------------- -------------
 00000053      00053850      00114688      00004096      00013147      00028000      00008454      00000001      00000106
     ✪              ✯              ⓡ
   DDNAME      EXCP-COUNT   RECORD-COUNT
------------ ------------- -------------
  ABSIX001      02638685      0393541
     ⓟ              ⓠ              ⓢ              ⓣ              ⓤ              ⓥ              ⓦ
IDX-USED(%) IDX-S-K-BYTES IDX-T-K-BYTES  IDX-CI-SIZE  IDX-S-CI-CNT IDX-T-CI-CNT   INDEX-LEVEL
------------ ------------- ------------- ------------ ------------- ------------- -------------
  00000077      00000184      00000831      00004096      00000045      00000203      00000003

<<< DATA PART INFORMATION >>>

   USED(%)    SPACE-K-BYTES TOTAL-K-BYTES    CI-SIZE    SPACE-CI-CNT  TOTAL-CI-CNT   SPLIT-CI-CNT  EXTENT-COUNT  SPLIT-CA-CNT
------------ ------------- ------------- ------------ ------------- ------------- ------------- ------------- -------------
 00000050      00113614      00229376      00004096      00027738      00056000      00006402      00000002      00000138
   DDNAME      EXCP-COUNT   RECORD-COUNT
------------ ------------- -------------
  ABSID001      05604480      00098512

 IDX-USED(%) IDX-S-K-BYTES IDX-T-K-BYTES  IDX-CI-SIZE  IDX-S-CI-CNT IDX-T-CI-CNT   INDEX-LEVEL
------------ ------------- ------------- ------------ ------------- ------------- -------------
  00000075      00000405      00001642      00004096      00000099      00000401      00000002
```

Refer Comment (1)

Refer Comment (2)

Refer Comment (3)

Refer Comment (4)

Comments: (1) When space utilization exceeds 100%, secondary space is obtained if there is free space on the DASD. If not, ALOGJLOG or ALOGSLOG utility will be terminated abnormally with a User Abend U0077. When the index or data component space utilization approaches 100%, ensure that there is sufficient disk space in the DASD for secondary space allocation. Alternatively, expand the A-LOG Management Database. Refer to **"4.6.2 A-LOG Management Database Expansion**" for further details.

(2) An unmatch occurs if this value is not equal to the total of all the record counts in RECORD-COUNT column on the next page. Refer to Comment (2) on the following page.

(3) An unmatch occurs if this value is not equal to the total of all the record counts in RECORD-COUNT column on the next page. Refer to Comment (1) on the following page.

(4) VSAM file access becomes less efficient when more and more control interval or control area splits occur. A-LOG Management Database should be reorganized if number of splits becomes too high. Refer to **"4.3.2 A-LOG Management Database Reorganization**" for further details.

Notes:

- Used Space
- Unused Space
- Allocated Space
- Control Interval (CI) size
- Number of Unused CIs
- Number of Allocated CIs
- Number of CI Splits
- Number of Extent
- Number of Control Area (CA) Splits
- DDNAME
- Number of EXCPs
- Number of Records
- Used Space (Index Dataset)
- Unused Space (Index Dataset)
- Allocated Space (Index Dataset)
- CI Size (Index Dataset)
- Number of Unused CI (Index Dataset)
- Number of Allocated CI (Index Dataset)
- Index Level (Index Dataset)

A-LOG Management Database Report (File Information)

A-BAS * CONTENTS OF A-BAS DATABASE * 96/12/15(17:08:45) PAGE : 2

```
**********************************
*        FILE INFORMATION        *
**********************************
```

SEQ NO.	FILE NO.	FILE NAME	RECORD-COUNT	TOP RECORD-NO.
1	0	SYSTEM FILE	1	1
2	10	SYS	1	1
3	20	JAM	12	12
4	30	RDM	0	0
5	40	RSM	0	0
6	50	BIS	0	0
7	70	BDS	0	0
8	80	DMM	116	123
9	90	PAS	0	0
10	100	DST	0	0
11	110	JBI	98,339	357,582
12	120	RAM	0	0
13	130	LBP	0	0
14	1	DICTIONARY FILE	43	43

Refer Comment (1) Refer Comment (2) Refer Comment (3)

Comments: (1) An unmatch occurs if the result of following calculation is not equal to the record counts in the RECORD COUNT column of the <<< INDEX PART INFORMATION >>> as shown in A-LOG Management Database Report (Space Utilization) on page 5-26.

Calculation = Record Counts of SYSTEM FILE + SYS + (JAM x 2) + DMM + (JBI x 4) + DICTIONARY FILE

(2) An unmatch occurs when the total of this column is not equal to the record counts in the RECORD COUNT column of the <<< DATA PART INFORMATION >>> as shown in A-LOG Management Database Report (Space Utilization) on page 5-26.

(3) An unmatch occurs when the total of RECORD-COUNT column is greater than the total of this column.

A-LOG Data Pool Report (Dataset Information)

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.

                                                                    DATE 96/12/15 TIME 17:05:35    PAGE    2
 SPACE CONTROL TABLE INFORMATION                                  LIST-CODE=BKSREP
   **********************
   * SCT DATASET INFO.  *
   **********************
```

PHYSICAL SCT-LAYOUT BKS-LAYOUT	DEVICE TYPE	NUMBER OF CYLINDERS	NUMBER OF BLOCKS	BLOCK-LENGTH	DD-NAME	VOL-SER	DATASET-NAME
SCT DATASET	3380	19	1425	9076	DDSCT	BSP001	ALOG.DB.SCT
BKS DATASET	3380	1500	112490	9076	DDBKS	BSP001	ALOG.DB.BKS

✶

LOGICAL SCT-LAYOUT BKS-LAYOUT	DEVICE TYPE	NUMBER OF CYLINDERS	NUMBER OF BLOCKS	BLOCK-LENGTH	DD-NAME	VOL-SER	DATASET-NAME
SCT DATASET	3380	19	1425	9076	DDSCT	BSP001	ALOG.DB.SCT
BKS DATASET	3380	1500	112490	9076	DDBKS	BSP001	ALOG.DB.BKS

UNUSED-STORAGE	DEVICE TYPE	NUMBER OF CYLINDERS	NUMBER OF BLOCKS	BLOCK-LENGTH	DD-NAME	VOL-SER	DATASET-NAME
SCT DATASET	3380	12	946	9076	DDSCT	BSP001	ALOG.DB.SCT
BKS DATASET	3380	526	39479	9076	DDBKS	BSP001	ALOG.DB.BKS

✸

Notes: ✶ Allocated Storage Information ✸ Unused Storage Information

A-LOG Data Pool Report (Block Information)

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.
                                                                DATE 96/12/15 TIME 10:32:25    PAGE    1
  SPACE CONTROL TABLE INFORMATION                                        LIST-CODE=BKSREP
   **********************
   *  SCT BLOCK INFO.    *
   **********************

   SCT DATASET          | BLOCK- | FROM     TO    (TOTAL)  | USED-BLOCK  | UNUSED-BLOCK
   INFO.                | LENGTH | RABN     RABN            |             |
  ----------------------|--------|--------------------------|-------------|---------------
   EXTENT INDEX BLOCK   |   9076 |      1 -    261      261 |       82    |        179
  ----------------------|--------|--------------------------|-------------|---------------
   EXTENT STATUS BLOCK  |   9076 |    262 -    274       13 |        5    |          8
  ----------------------|--------|--------------------------|-------------|---------------
   EXTENT BLOCK         |   9076 |    275 -   1378     1104 |      347    |        757
  ----------------------|--------|--------------------------|-------------|---------------
   STATUS BLOCK         |   9076 |   1379 -   1426       48 |       44    |          4
  ------------------------------------------------------------------------------------------
   BKS BLOCK            | BLOCK- | FROM     TO     TOTAL)  | USED-BLOCK  | UNUSED-BLOCK
   INFO.                | LENGTH | RABN     RABN            |             |
  ----------------------|--------|--------------------------|-------------|---------------
   BKS BLOCK            |  9076  |      1 - 112490  112490  |    73011    |      39479
  ------------------------------------------------------------------------------------------
```

Refer to Comment

Comments: A-LOG Data Pool consists direct access datasets, therefore, secondary space will not be allocated even if all initial allocated space is used. When records are deleted, the deleted space is reused. If the number of unused blocks becomes too low, expired job logs or SYSLOGs should be dumped or deleted to increase unused storage space. If remaining space is still insufficient, the A-LOG Data Pool must be expanded. Refer to **"4.6.3 A-LOG Data Pool Expansion**" for further details.

5.14 ALOGUBKS Utility

Function

This utility backups the A-LOG Data Pool. It offers the following capabilities:

1) Unload A-LOG data pool to a sequential file.
2) Show the number of reports and pages backed up from A-LOG Data Pool in an output listing.

Processing Outline

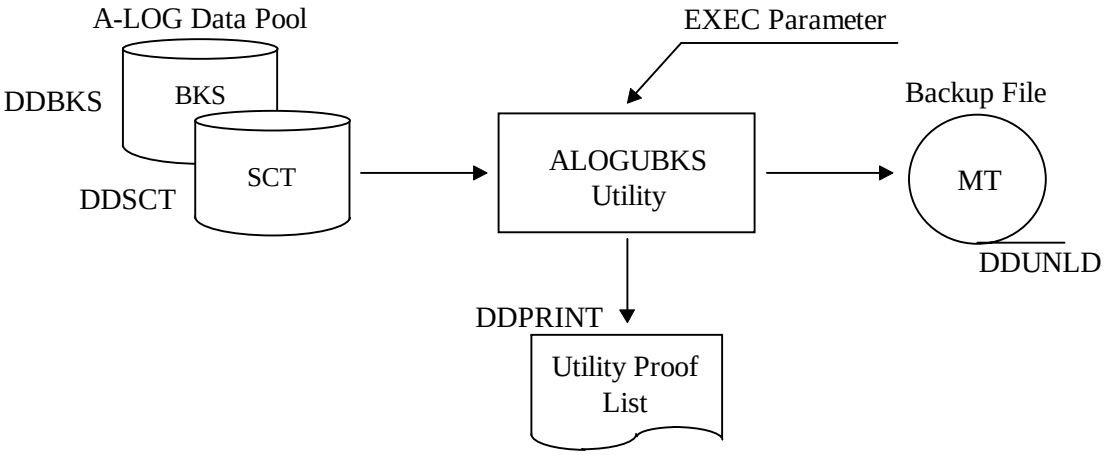


Fig. 5.13: ALOGUBKS Processing Outline

Execution Procedure

1) Shut down A-LOG Monitor.

2) Modify the size of the memory storage area for SCT blocks as required.

```
PARM='SIZE=area-sizeK,TYPE=ALOG'
```

Specifies the size of the memory storage area for the SCT control blocks (extent index, extent status, extent and status). The size is specified in Kbytes. Default value is 2,000K. Minimum value is 20K.

The optimum value may be calculated as follows:

```
Optimum Value = ( 1 + Number of Extent Index Blocks + Number of Extent Status Blocks +
                  Number of Extent Blocks + Number of Status Blocks ) x Block Size of SCT File +
                ( 1 + Number of Extent Index Blocks + Number of Extent Status Blocks +
                  Number of Extent Blocks + Number of Status Blocks ) x 8
```

Note: Number of Extent Index Blocks, Number of Extent Blocks, Number of Extent Status Blocks and Number of Status Blocks can be obtained from the reports generated by ALOGREP utility.

3) Specify the volume serial number of the backup file in EXEC parameter.

```
ALOGUBKS,BKVOL=volser
```

Specify the volume serial number for the output backup file. If required, modify the DDUNLD DD statement of the execution JCLs, such as the dataset name and UNIT of the output backup dataset.

4) Execute ALOGUBKS utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

Output Listing

A sample proof list is shown in the following page.

ALOGUBKS Proof List

```
*** A-LOG (ADVANCED JOBLOG/SYSLOG MANAGEMENT SYSTEM) ***     VERSION 03.80   BY BSP INC.

                                                                    DATE 96/12/15 TIME 10:32:53    PAGE    1
A-LOG UNLOAD UTILITY :

    ✶  READ REPORT COUNT =  00000042
    ✸  READ PAGE   COUNT =  00002139

    ✹  UNLOAD VOLUME  = BSPOO1
    ❋  UNLOAD DATASET = ALOG.DB.BKS

A-LOG UNLOAD UTILITY NORMAL END :
```

Notes:
- ✶ Number of job logs or SYSLOGs unloaded
- ✸ Total number of pages unloaded
- ✹ Volume serial number in DDUNLD DD statement
- ❋ Dataset name in DDUNLD DD statement

5.15 ALOGUNLD Utility

Function

This utility backup the A-LOG Management Database. It contains the following steps:

1) STEP0010 : Verify the A-LOG Management Database
2) UNLDIX : Backup the index portion of A-LOG Management Database
3) UNLDDT : Backup the data portion of A-LOG Management Database

Processing Outline

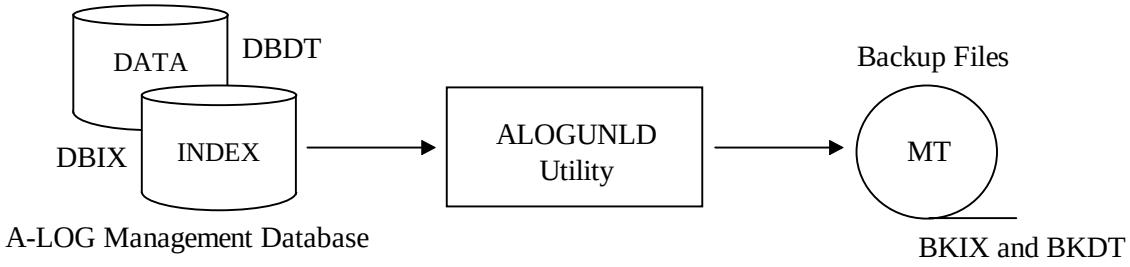


Fig. 5.14: ALOGUNLD Processing Outline

Execution Procedure

1) Specify the program name of manufacturer-supplied utility and the volume serial number for the backup file in EXEC parameter.

```
ALOGUNLD,UTL=program-name,      ✶
                BKVOL=volser    ✷
```

Notes: ✶ Specify one of the following manufacturer-supplied program names:

- IBM: IDCAMS
- Fujitsu: KQCAMS
- Hitachi: JSCVSUT

✷ Specify the volume serial number for the backup file. If you change the dataset name of the backup file, change it as follows:

- Backup file of the Index section : Change BKIX DD statement.
- Backup file of the data section : Change BKDT DD statement.

2) Execute ALOGUNLD utility. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**" for the execution JCLs.

5.16 DUMFORM0 Utility

Function

This utility creates and formats DAM files such as A-LOG Data Pool, Security Management Table and Security Log File. Formatting is performed according to the specifications in the format parameter.

Processing Outline

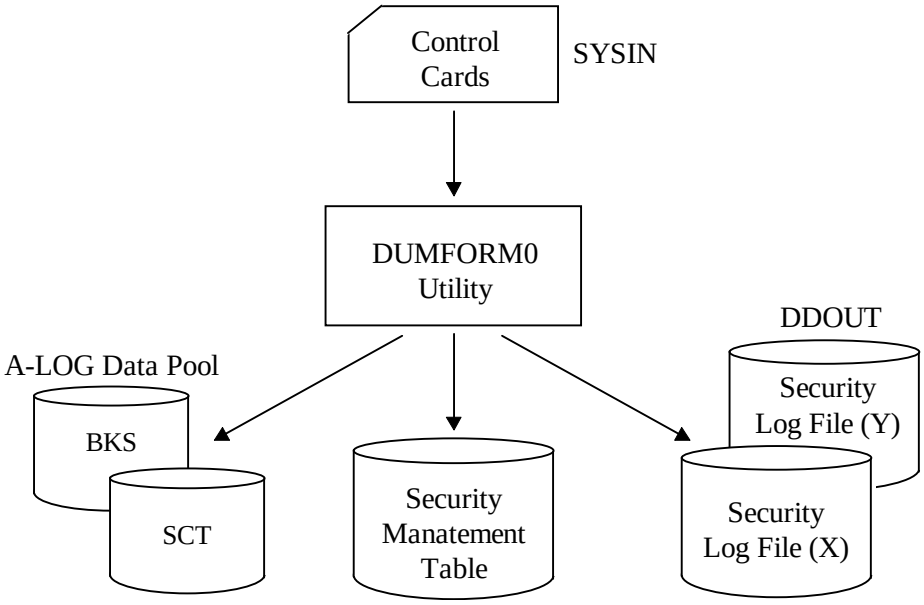


Fig. 5.15: DUMFORM0 Processing Outline

Execution Procedure

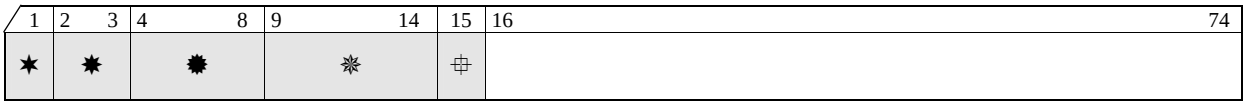
1) Specify the member name of the format parameter in SYSIN DD statement.

   Depending on the DAM file to be formatted and the device type for the DAM file, one of the following format parameter member names can be specified in the SYSIN DD statement of execution JCL:

   - A-LOG Data Pool (SCT) : FRMC3350 (IBM 3350 or equivalent)
     FRMC3380 (IBM 3380 or equivalent)
     FRMC3390 (IBM 3390 or equivalent)
     FRMC9345 (IBM 9345 or equivalent)
   - A-LOG Data Pool (BKS) : FRMK3350 (IBM 3350 or equivalent)
     FRMK3380 (IBM 3380 or equivalent)
     FRMK3390 (IBM 3390 or equivalent)
     FRMK9345 (IBM 9345 or equivalent)

- Security Management Table : FRMCSECR
- Security Log File X : FRMSECRL
- Security Log File Y : FRMSECRL

2) Modify the format parameter as required. Layout of the format parameter is as follows:



Item	Item Name	Col	Format & Length	Description
✶	Dataset Organization	1	Alphabetic 1	Specify the dataset organization as follows: K : Key R : Relative
✷	Key Length	2	Numeric 2	For key organization dataset, specify the key length in bytes. Key length must be specified in two digits (i.e. leading zeroes are required). For relative organization dataset, specify blanks.
✹	Block Size	4	Numeric 5	Specify the block size in bytes. Block size must be specified in 5 digits (i.e. leading zeroes are required).
❋	Total Number of Blocks	9	Numeric 6	For a multi-volume file, specify the total number of blocks to be formatted as follows: Total Number of Blocks = Number of Blocks per Cylinder x Total Number of Cylinder Allocated For a single-volume file, specify blanks.
⌖	Clear Flag	15	Alphabetic 1	Specify a value 'L' as the clear character.

3) Specify the EXEC parameter.

```
//STEP010   EXEC   DUMFORM0,FORMAT=member-name-of-format-parameter,   ✶
                   SPC=size,                                          ✷
                   NAME=dataset-name                                  ✹
```

Notes: ✶ Specify the member name of a format parameter.

✷ Specify the space requirements in cylinder for the file to be created.

✹ Specify the dataset name for the file to be created.

4) Execute DUMFORM0 utility. The JCL for this utility is included as a member in A-LOG Installation JCL Library. Sample execution JCLs are shown as follows:

```
//DUMFORM  JOB ALOG,MSGCLASS=A,CLASS=A
//*-----------------------------------------------------*
//*  FORMAT : BLKSIZE FOR EACH FILE                     *
//*  SPC    : SPACE TO BE ALLOCATED                     *
//*  NAME   : DATASET NAME                              *
//*-----------------------------------------------------*
//DUMFORM  PROC  SPC=,NAME=
//STEP020   EXEC PGM=DUMFORM0,PARM=F
//STEPLIB   DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//STEPCAT   DD   DSN=ALOG.UCAT,DISP=SHR
//DDOUT     DD   DSN=&NAME,DISP=(,CATLG,DELETE),            ✶
//               UNIT=SYSDA,VOL=SER=WORK02,
//               SPACE=(CYL,(&SPC),,CONTIG)
//SYSIN     DD   DSN=ALOG.V0380.PARMLIB(&FORMAT),DISP=SHR   ✸
//          PEND
//*
//STEP010   EXEC DUMFORM
//          FORMAT=FRMLOG,       BLKSIZE
//          SPC=1,               SPACE
//          NAME='ALOG.DB.SCT'   DATASET NAME
```

✶ File to be created

✸ Format Parameter

Chapter 6

A-LOG System Libraries

6.1 Load Module Library

A-LOG programs are supplied in the form of load modules. These load modules are stored in A-LOG Load Module Library. Its standard dataset name is 'ALOG.V0380.LOAD'. Some of the load modules perform operations that require APF authorization. Therefore, A-LOG Load Module Library must be defined as an APF-authorized library. Refer to "**2.1.2 Operating System Requirements**" for further details.

A-LOG system uses a user SVC routine. Therefore, the user SVC routine must be registered into SYS1.LPALIB library of the operating system instead of A-LOG Load Module Library. Refer to "**2.2.9 Create User SVC Routine**" for further details.

A-LOG system is made up of a number of components and procedures. A component may consists of one or more sub-components. These sub-components are the A-LOG load modules. A procedure may consists of one or more steps, A-LOG load modules may be invoked in these steps. The load module names which constitute each component or procedure are shown in Table 6.1.

Procedure or Component Name	Load Module Name	Description
A-LOG Terminal Facility	USLPINIT	It initializes and display the [Hello] screen.
	USLP2	It displays [0.0.0 Initial Menu] screen.
	ALGA000L	It displays [1.0.0 Master Management] screen.
	ASPA300L	It displays [1.1.0 Joblog/SYSLOG Master Management] screen.
	ASPL000L	It displays [2.0.0 Joblog Management] screen.
	ASPL100L	It displays [2.1.0 Joblog Index Maintenance] screen.
	ASPS000L	It displays [3.0.0 SYSLOG Management] screen.
	ASPS100L	It displays [3.1.0 SYSLOG Data Select] screen.
	ASPS200L	It displays [3.2.0 SYSLOG Index Maintenance] screen.
	ALGB300L	It displays [2.2.0 Dump MT Inquiry] screen.
	ALGX000L	It displays [9.0.0 System Management] screen.
	ALGX100L	It displays [9.1.0 Terminal/PF-Keys Setting] screen.
ABSIHIN	ABSIHINL	It creates and initializes the Hot Log File.
ALOG	SPL	It manages A-LOG Management Database and A-LOG Data Pool and receives and replies to commands from other components and the operators. It is a resident module.
ALOGBLOG	LOG900	It creates Job Log Index Release File.
	LOG910	It generates Job Log Audit Trail List.

Table 6.1.a: List of A-LOG Load Modules

Procedure or Component Name	Load Module Name	Description
ALOGCHKA	CHKSPLA	It checks whether A-LOG Monitor is active.
ALOGCHKL	CHKSPLL	It checks whether A-LOG Monitor has stopped.
ALOGCPY1	CPY100L	It creates a sequential file for each file in A-LOG Management Database.
ALOGCPY2	CPY200L	It restores the sequential file created by ASPLCPY1 utility into A-LOG Management Database.
ALOGDAM0	AUTODAM0	It restores data into Security Management Table (DAM).
ALOGDATE	DAT01L	It sets the Operation Date for A-LOG system.
ALOGDLT1	DLT01J	It deletes expired job logs or SYSLOGs from A-LOG Data Pool.
ALOGDLT4	DLT04L	It deletes expired records in Dump MT Information File.
ALOGDLT5	DLT05	It deletes expire job log or SYSLOG records.
ALOGDMP1	CHKSPLL	It checks whether A-LOG Monitor has stopped.
	ABSIDMP	It converts A-LOG Management Database.
	DMPJSELS	It generates files for used by ALOGDMP2 utility and ALOGDMP3 utility.
ALOGDMP2	CHKSPLA	It checks whether A-LOG Monitor is active.
	DUMPJS	It dumps job logs.
	DLT01JS	It deletes dumped job logs and job logs with expired restore retention days.
ALOGDMP3	DLT05JS	It deletes job logs with expired restore retention days.
ALOGDMPC	DMPCPYL	It copies a Dump MT created by ALOGJDMP utility or ALOGLDMP utility to another magnetic tape.
ALOGFORM	AUTOFORM	It back up Security Management Table (DAM).
ALOGHOLT	AUTOHOLT	It maintains the Holiday Master.
ALOGIBKS	SCTRUNL	It initializes A-LOG Data Pool.
ALOGICHK	ABSICNVL	It checks for unmatched records in A-LOG Management Database.
ALOGJDMP	DUMPL	It archives job log records from A-LOG Management Database and A-LOG Data Pool into a Dump MT.
	DLT01L	It deletes dumped job log records from A-LOG Management Database and A-LOG Data Pool.

Table 6.1.b: List of A-LOG Load Modules

Procedure or Component Name	Load Module Name	Description
ALOGJLOG	ASPLJLOG	It is a resident module for accessing the OS Spool. Upon the first call, it loads JSDMLLNK module and passes control to it.
	JSDMLLNK	It is called and loaded by ASPLJLOG module. It is used to store job logs into A-LOG Management Database.
	CATJLOG	It analyses job log information.
ALOGJMST	JAM100	It registers, deletes and generates reports on Job Log or SYSLOG Master.
ALOGJRST	RESTOREL	It restores the job log records dumped by ALOGJDMP utility to A-LOG Management Database and A-LOG Data Pool.
ALOGLBKS	BKSLDEL	It restores the data unloaded by ALOGUBKS utility into A-LOG Data Pool.
ALOGLDMP	DUMPSLOG	It archives SYSLOG records from A-LOG Management Database and A-LOG Data Pool into Dump MT.
	DLT01L	It deletes dumped SYSLOG records from A-LOG Management Database and A-LOG Data Pool.
ALOGLLOG	LOG920	It creates SYSLOG Index Release File.
	LOG930	It generates a SYSLOG Audit Trail List.
ALOGLRST	RESTSLOG	It restores the SYSLOG records dumped by ALOGLDMP utility to A-LOG Management Database and A-LOG Data Pool.
ALOGRENO	ABSICNVL	It removes unmatched records from A-LOG Management Database.
ALOGREP	ABSIREPL	It generates reports on the space utilization of A-LOG Management Database (VSAM) and A-LOG Data Pool (DAM).
ALOGRSEL	RESTSELL	It selects job logs and SYSLOGs that have been reserved for restoration by users of A-LOG Terminal Facility and generates a job to restore these job logs or SYSLOG from archived tape.
ALOGJLOG	ASPLSLOG	It is a resident module for accessing the OS Spool. Upon the first call, it loads LSDMLLNK module and passes control to it.
	LSDMLLNK	It is called and loaded by ASPLSLOG module. It is used to store SYSLOGs into A-LOG Management Database.
	CATSLOG	It analyses SYSLOG information.
ALOGUBKS	BKSUDEL	It unloads the A-LOG Data Pool.
DUMFORM0	DUMFORM0	It generates and formats A-LOG Data Pool (DAM), Security Management Table (DAM), Security Log File (DAM) and Print Log File (DAM).
WTRJUTL	WTRJUTL	It prints job logs specified in the selection parameter cards.
WTRSUTL	WTRSUTL	It prints SYSLOGs specified in the selection parameter cards.

Table 6.1.c: List of A-LOG Load Modules

6.2 Parameter Library

6.2.1 Introduction

The parameters used by A-LOG system are stored in A-LOG Parameter Library. The standard dataset name of A-LOG Parameter Library is 'ALOG.V0380.PARMLIB'. The following table lists the members in the library and the procedures that use these members. The table also indicates whether a member is mandatory and modifiable. The '**Reference**' column gives the name of the manual that has more information about the member and/or detailed description of its parameters.

Member	Procedure	Description	Usage	Modify	Reference
ABSIPRM1	ALOG	Define the type of access to A-LOG Management Database	□	○	This Manual
ALOGJRST	ALOGRSEL & Terminal Facility	Contain default JCL for ALOGJRST utility	□	○	User's Manual Chapter 11
ALOGLRST	ALOGRSEL & Terminal Facility	Contain default JCL for ALOGLRST utility	□	○	User's Manual Chapter 11
ASPOPTB	Terminal Facility	Define initial setting value of A-LOG Terminal Facility		○	This Manual
ASPPRM1	ALOG	Define A-LOG system environment	□	○	This Manual
BKSPARM	ALOGIBKS ALOGREP	Define parameters for initializing A-LOG database	□	○	This Manual Chapter 5
CPYDMM CPYJAM CPYJBI CPYSYS	ALOGCPY1 ALOGCPY2	Each member defines a database file name	□		This Manual Chapter 5
DCDALCTB	Terminal Facility	Define dataset names of A-LOG Data Pool for A-LOG Terminal Facility	□	○	This Manual
DCDTERM	Terminal Facility	Contain definitions for non-Kanji terminals	□	○	—
DCDTERMF	Terminal Facility	Contain definitions for Fujitsu Kanji terminals	□ (Select)	○	—
DCDTERMH	Terminal Facility	Contain definitions for Hitachi Kanji terminals	□ (Select)	○	—

□ Mandatory ○ Modifiable

Table 6.2.a: List of A-LOG Parameters

Member	Procedure	Description	Usage	Modify	Reference
DCDTERMI	Terminal Facility	Contain definitions for IBM Kanji terminals	□ (Select)	○	—
DCDXIOTB	Terminal Facility	Define terminal I/O modules for A-LOG Terminal Facility	□	○	—
DDCATJ	ALOGJLOG	Define job log cataloging module name and output of job log records	□	○	User's Manual Chapter 11
DDCATS	ALOGSLOG	Define SYSLOG cataloging module name and output of SYSLOG records	□	○	
DDJEXT	ALOGJLOG	Define job log exit module name		○	User's Manual Chapter 11
DDJLOG	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGF	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGH	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGH4	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGI	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGIE	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGI3	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGI8	ASPLJCHK ALOGJLOG	Contain job log message definition parameter	□ (Select)	○	User's Manual Chapter 11
DDJLOGP	ASPLJCHK ALOGJLOG	Contain job definition parameter	□	○	User's Manual Chapter 11
DDJPARM	WTRJUTL	Contain separator control statements		○	User's Manual Chapter 11
DDJUTL	ALOGJLOG	Define JCL modify exit module name		○	User's Manual Chapter 11

□ Mandatory ○ Modifiable

Table 6.2.b: List of A-LOG Parameters

Member	Procedure	Description	Usage	Modify	Reference
DDSCNTL	ALOGSLOG	Define the control card for SYSLOG storage	□	○	User's Manual Chapter 11
DDSPARM	WTRSUTL	Contain separator control statements		○	User's Manual Chapter 11
DEFDPARM	ALOGIUPD ALOGLOAD	Contain VSAM command for allocating data part of A-LOG Management Database	□	○	—
DEFIPARM	ALOGIUPD ALOGLOAD	Contain VSAM command for allocating index part of A-LOG Management Database	□	○	—
DLTDPARM	ALOGIUPD ALOGLOAD	Contain VSAM command for deleting data part of A-LOG Management Database	□	○	—
DLTIPARM	ALOGIUPD ALOGLOAD	Contain VSAM command for deleting index part of A-LOG Management Database	□	○	—
DMPJOB	ALOGJDMP	Define job log dump control statement	□	○	User's Manual Chapter 11
DMPSYS	ALOGLDMP	Define SYSLOG dump control statement	□	○	User's Manual Chapter 11
FRMCSECR	During Installation & DUMFORM0	Define the format parameter for Security Management Table	□		This Manual Chapter 5
FRMC3350 FRMC3380 FRMC3390 FRMC9345	During Installation & DUMFORM0	Each member defines the format parameter for a specific disk device to be used by the SCT part of A-LOG Data Pool	□		This Manual Chapter 5
FRMK3350 FRMK3380 FRMK3390 FRMK9345	During Installation & DUMFORM0	Each member defines the format parameter for a specific disk device to be used by the BKS part of A-LOG Data Pool	□		This Manual Chapter 5
FRMPARM	During Installation	Define the format parameter for Recovery File	□		—
FRMPDMM FRMPJAM FRMPJBI FRMPSYS	During Installation	Each member defines the initialization parameters for a database file of A-LOG Management Database	□		—

□ Mandatory ○ Modifiable

Table 6.2.c: List of A-LOG Parameters

Member	Procedure	Description	Usage	Modify	Reference
FRMSECRL	During Installation & DUMFORM0	Define the format parameter for Security Log File	□		This Manual Chapter 5
HLGPARM	During Installation & ABSIHIN	Define the format parameter for Hot Log file	□	○	This Manual Chapter 5
LDDTPARM	ALOGLOAD	Contain VSAM command for re-loading data part of A-LOG Management Database	□		—
LDIXPARM	ALOGLOAD	Contain VSAM command for re-loading index part of A-LOG Management Database	□		—
LOADJAM LOADJBI LOADDMM LOADSYS	During Installation	Each member defines a database file name of the A-LOG Management Database	□		—
LOADPARM	ALOGIUPD	Contain VSAM commands for re-loading A-LOG Management Database	□		—
PRTPARM	ABSPRINT	Contain VSAM commands for printing A-LOG Management Database	□		—
RSELPARM	ALOGRSEL	Define control cards for Restore job generation		○	User's Manual Chapter 11
RUNDMP1	ALOGDMP2	Contain sort parameter	□		—
RUNDMP2	ALOGDMP2	Contain sort parameter	□		—
RUNDMP3	ALOGDMP3	Contain sort parameter	□		—
RUNDMP4	ALOGDMP3	Contain sort parameter	□		—
RUNLOG5	ALOGBLOG ALOGLLOG	Contain sort parameter	□		—
RUNSPM2	ALOG	Define initialization parameters for starting A-LOG Monitor	□	○	This Manual
RUNSPM3	ALOG	Contain VSAM commands for verifying A-LOG Management Database	□		—

□ Mandatory ○ Modifiable

Table 6.2.d: List of A-LOG Parameters

Member	Procedure	Description	Usage	Modify	Reference
SCTINIT	ALOGIBKS	Define the name of the program for initializing A-LOG Data Pool	□		—
SCTREP	ALOGREP	Define the name of the program for generating A-LOG Data Pool	□		—
SORTDATA	ALOGIUPD	Contain sort parameter	□		—
SORTIDX1	ALOGICHK	Contain sort parameter	□		—
SORTIDX2	ALOGIUPD	Contain sort parameter	□		—
ULDDMP	ALOGDMP1	Contain VSAM command for converting A-LOG Management Database	□		—
ULDDPARM	ALOGUNLD	Contain VSAM command for unloading A-LOG Management Database	□		—
ULDIPARM	ALOGUNLD	Contain VSAM command for unloading A-LOG Management Database	□		—
UNLDPARM	ALOGICHK	Contain VSAM commands for unloading A-LOG Management Database	□		—
USMSJRST	ALOGJRST	Define Restore Control Parameter		○	User's Manual Chapter 11
USMSLRST	ALOGLRST	Define Restore Control Parameter		○	User's Manual Chapter 11
VERPARM	ABSIRENO ALOGUNLD	Contain VSAM command for verifying A-LOG Management Database	□		—
WANDMM WANJAM WANJBI WANSYS	ALOG	Each member defines a database file structure of the A-LOG Management Database	□		—
WTRJUTL	—	Contain default JCL for WTRJUTL utility		○	User's Manual Chapter 11
WTRSUTL	—	Contain default JCL for WTRSUTL utility		○	User's Manual Chapter 11

□ Mandatory ○ Modifiable

Table 6.2.f: List of A-LOG Parameters

6.2.2 Contents of Members

Member ABSIPRM1

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
INIT INDEXKEYL=52,           /* INDEX KEY LENGTH    */
     FINDWK=100,             /* FIND WORK           */
     DEBUG=0,                /* DEBUG MODE NO       */
     MODE=S                  /* SINGLE MODE         */
```

This member defines the maximum number of commands for each type of access to A-LOG Management Database. The values specified for these parameters affect the region size required for running A-LOG Monitor. Refer to "**1.2 A-LOG System Operation Requirements**" for further details. The keywords used in the member are described as follows:

Keyword	Description	Maximum Value	Table Size (per unit)
INDEXKEYL	Fixed as '52'	—	—
DEBUG	Fixed as '0'	—	—
MODE	Fixed as 'S'	—	—
FINDWK	See Note (1), Default: 10	32767	488 bytes
TRANS	Fixed as '2'	—	—
USER	Fixed as '2'	—	—
EXCLUSIVE	Fixed as '1'	—	—
ACCESS	See Note (2), Default: 32	999	352 bytes
SETRC	See Note (3), Default: 20	999	488 bytes
ANALIZE	See Note (4), Default: 64	999	90 bytes

Table 6.3: Keywords of Member ABSIPRM1

Notes: (1) FINDWK Keyword

This keyword specifies the maximum number of active Search Tables that can be processed by A-LOG Monitor. Search Tables are used by FIND commands. One Search Table can store 115 records. Error Code 247 will be returned if this value is too small. FINDWK value can be estimated as follows:

FINDWK Value = (Number of Jobs Registered in 1 Day x Retention Days x Average Number of Criteria) / 115

Example: Assuming: a) Number of Jobs Registered in 1 Day = 2,000
b) Retention Days = 14
c) Average Number of Criteria = 2

FINDWK Value = (2,000 x 14 x 2) / 115 = 487 = 500 approximately

Therefore, required memory size = 500 x 488 = 239 Kbytes

(2) ACCESS Keyword

This keyword specifies the maximum number of concurrent commands that can be processed by A-LOG Monitor. Commands may be issued from A-LOG terminal users, A-LOG utilities, etc. Error Code 026 will be returned if this value is too small. Its value can be estimated as follows:

ACCESS Value = Maximum Number of Concurrent A-LOG Terminal Users + Maximum Number of Concurrent Utilities

Example: Assuming: a) Maximum Number of Concurrent A-LOG Terminal Users = 24
b) Maximum Number of Concurrent Utilities = 8

ACCESS Value = 24 + 8 = 32

Therefore, required memory size = 32 x 352 = 11 Kbytes

(3) SETRC Keyword

This keyword specifies the maximum number of active FIND commands that can be processed by A-LOG Monitor simultaneously. A user (A-LOG terminal user, A-LOG utilities, etc.) may have more than one active FIND command at any one time. Error Code 027 will be returned if this value is too small. SETRC value can be estimated as follows:

SETRC Value = Number of Active FIND Commands Per User x ACCESS Value

Example: AssumingNumber of Active FIND Commands Per User = 2

SETRC Value = 2 x 32 = 64

Therefore, required memory size = 64 x 488 = 31 Kbytes

(4) ANALIZE Keyword

This keyword specifies the maximum number of active Field Tables that can be processed by A-LOG Monitor. Field Tables are used by READ commands. One Field Table contains 10 fields. Error Code 028 will be returned if this value is too small. ANALIZE value can be estimated as follows:

ANALIZE Value = (Number of Files x Number of Fields x ACCESS Value) / 10

Example: Assuming a) Maximum Number of Files = 3
b) Maximum Number of Fields = 60

ANALIZE Value = (3 x 60 x 32) / 10 = 576 = 580 approximately

Therefore, required memory size = 580 x 90 = 51 Kbytes

Therefore, the total required memory size = 239 + 11 + 31 + 51 = **332 Kbytes**

Member ALOGJRST

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ALOGJRST JOB ,ALOG,CLASS=A,MSGCLASS=A
//*******************************************************
//*                                                     *
//*     JOBLOG RESTORE  UTILITY                         *
//*                                                     *
//*      MT (EDIT2) ===> BKS(EDIT1)                     *
//*                                                     *
//*      VERSION NO.   V03.80                           *
//*                                                     *
//*******************************************************
//ALOGJRST PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=RESTOREL,PARM='ABEND=N'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//USMSCARD DD   DSN=ALOG.V0380.PARMLIB(USMSJRST),DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR
//*DDIN     DD   DSN=RESTOREL,
//*              DISP=(OLD,KEEP),UNIT=(TAPE,,DEFER),
//*              VOL=SER=XXXXXX
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSOUT   DD   DUMMY
//DDSNAP   DD   DUMMY             /* DYNALLOC MESSAGE */
//SYSPRINT DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//         PEND
//ALOGJRST EXEC ALORJRST
//DDCNTL   DD   *
/*
```

Member ALOGLRST

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ALOGLRST JOB ,ALOG,CLASS=A,MSGCLASS=A
//*******************************************************
//*                                                     *
//*     SYSLOG RESTORE  UTILITY                         *
//*                                                     *
//*      MT (EDIT2) ===> BKS(EDIT1)                     *
//*                                                     *
//*      VERSION NO.   V03.80                           *
//*                                                     *
//*******************************************************
//ALOGLRST PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=RESTSLOG,PARM='ABEND=N'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//USMSCARD DD   DSN=ALOG.V0380.PARMLIB(USMSLRST),DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR
//*DDIN     DD   DSN=RESTOREL,
//*              DISP=(OLD,KEEP),UNIT=(TAPE,,DEFER),
//*              VOL=SER=XXXXXX
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,1)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,1)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,1)
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,10,2),RLSE)
//SYSOUT   DD   DUMMY
//DDSNAP   DD   DUMMY             /* DYNALLOC MESSAGE */
//SYSPRINT DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//         PEND
//ALOGLRST EXEC ALORLRST
//DDCNTL   DD   *
/*
```

Member ASPOPTB

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
USLP NDMM=40,NJBI=100,NJBD=100,NERM=100,NSJBI=100,NDATA=40,
USLP NSLOG=100,NSGAP=20,NSCLS=,NSWTR=WTRSUTL,NJWTR=WTRJUTL,
USLP NJNAM=,NJRJOB=ALOGJRST,NJBKSD=1,NJIDXD=1,
USLP NSRJOB=ALOGLRST,NSBKSD=1,NSIDXD=1,
USLP TIMEC=26,JNOC=38,JNOL=8,JCOML=3,TCOM=:,JCOM=ZERO
```

This member defines the A-LOG Terminal Facility's environment. The size of the work area required for A-LOG Terminal Facility is determined by the values specified for the keywords in this member. Refer to "**1.2 A-LOG System Operation Requirements**" for further details. The keywords used in this member are described as follows:

Keyword	Description	Maximum
NDMM	Maximum number of Dump MT Information records that can be inquired.	32767
NJBI	Maximum number of job log index records that can be inquired.	32767
NJBD	Maximum number of job log index records that can be inquired (number of SYSOUT records in 1 job.	32767
NERM	Maximum number of error messages that can be displayed.	32767
NWORK	Size of internal work area (usually fixed).	32767
NDATA	Size of internal data area (usually fixed).	32767
NSJBI	Maximum number of SYSLOG index records that can be inquired.	32767
NSLOG	Maximum number of SYSLOG data (lines) that can be displayed.	32767
NSCLS	Default class to be displayed in "PRINT CLASS" item in [3.1.0 SYSLOG DATA SELECT] screen.	
NSWTR	Default print job name to be displayed in "PRINT JOB NAME" item in [3.1.0 SYSLOG DATA SELECT] screen.	
NSGAP	Default decrement value (unit: minute) of SYSLOG time to be displayed in "SYSLOG TIME" item in [3.1.0 SYSLOG DATA SELECT] screen.	
NJWTR	Default writer name of job log to be displayed in "WRITER NAME OF WRITER" item in [2.1.0 JOBLOG INDEX MAINTENANCE] screen.	
NJNAM	Default job name to be displayed in "JOB NAME" item in [2.1.0 JOBLOG INDEX MAINTENANCE] screen. If 'U-ID' is specified, the user logon ID is displayed.	

Table 6.4.a: Keywords in Member ASPOPTB

Keyword	Description	Maximum
NJRJOB	Default member name to be displayed in "MEMBER NAME" item in [2.1.0 JOBLOG INDEX MAINTENANCE] screen.	
NJBKSD	Default number of days to be displayed in " RESTORE JOB CONTROL DATA RETENTION DAYS" item in [2.1.0 JOBLOG INDEX MAINTENANCE] screen.	
NJIDXD	Default number of days to be displayed in " RESTORE JOB CONTROL INDEX RETENTION DAYS" item in [2.1.0 JOBLOG INDEX MAINTENANCE] screen.	
NSRJOB	Default member name to be displayed in "MEMBER NAME" item in [3.2.0 SYSLOG INDEX MAINTENANCE] screen.	
NSBKSD	Default number of days to be displayed in " RESTORE JOB CONTROL DATA RETENTION DAYS" item in in [3.2.0 SYSLOG INDEX MAINTENANCE] screen.	
NSIDXD	Default number of days to be displayed in " RESTORE JOB CONTROL INDEX RETENTION DAYS" item in in [3.2.0 SYSLOG INDEX MAINTENANCE] screen.	
TIMEC	Specify the starting position of time field in SYSLOG data.	
TCOM	Specify the character to be used for separating time, minute, second of time field in SYSLOG data.	
JNOC	Specify the starting position of job number field in SYSLOG data.	
JNOL	Specify the length of job number field in SYSLOG data.	
JCOM	One of the following can be selected as suppress character for job number in SYSLOG data: • BLANK : space • ZERO : 0 (default)	
JCOML	Specify the length of identifier for job number in SYSLOG data.	

Table 6.4.b: Keywords in Member ASPOPTB

Member ASPPRM1

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASPL MAJOR=USMSV100,SPMMN=SPMXV-00
ASPL SVC=250,SPMDB=10
ASPL OS=MVS/ESA,JES=JES2
ASPL WTODESC=8,WTOROUT=15
ASPL MAXLC=180,MAXLL=304,BUFFL=304
ASPL DBTYPE=ABSUSER,COMPRESS=YES
ASPL ACTIVATE=GLOBAL
```

The keywords used in this member are described as follows:

Keyword	Description	Maximum
MAJOR	System ENQ MAJOR name.	
SPMMN	If SVC is shared, specify MINOR-ENQ name in SPMXV-*nn* format.	
OS	One of the following operating systems can be selected: • MVS, MVS/XA, MVS/ESA, E40, MSP/E10, MSP/E20, VOS3, VOS3/AS	
JES	One of the following JES system can be selected: • JES, JES2, JES3, JSS3, JSS4	
WTOROUT	WTO route code.	
WTODESC	WTO description code.	
SVC	A-LOG SVC number.	
DBTYPE	Fixed as 'ABSUSER'.	
COMPRESS	Specify whether to use compress function of database.	
SPMDB	If SVC is shared, specify *nn* that was specified in SPMMN. However, '00' cannot be specified.	
MAXLL	Maximum number of columns for one line of a page. This value affects the internal page image area for job log storage module, A-LOG Terminal Facility and print module. See Note.	32767
MAXLC	Maximum number of lines for one page. This value affects the internal page image area for job log storage module, A-LOG Terminal Facility and print module. See Note.	32767
BUFFL	Maximum length of the buffer for storing one record of a job log read by ALOGJLOG utility.	
MODE	Specify 'MULTI' if A-LOG multi-CPU function is in used.	
ACTIVATE	If A-LOG multi-CPU function is in used, one of the following system type where A-LOG is active can be selected: • GLOBAL: Global System • LOCAL: Local System	

Table 6.5: Keywords in Member ASPPRM1

Note: When changing either the MAXLL or MAXLC value, ensure that a value bigger than the current value is specified

Member BKSPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BKSIOS SCTDEV=3380              <- SCT DATASET DEVICE TYPE
BKSIOS SCTSIZE=002              <- SCT DATASET ALLOC SIZE
BKSIOS BKSDEV=3380              <- BKS DATASET DEVICE TYPE
BKSIOS BKSSIZE=010              <- BKS DATASET ALLOC SIZE
BKSIOS EXTBLKC=5                <- BKS BLOCK/EXTENT
BKSIOS MAXREPC=10000 32000/PACK <- MAX REPORT COUNT(MAX 16M)
```

Allocation of A-LOG data pool

Member CPYDMM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
DMM
```

Member CPYJAM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
JAM
```

Member CPYJBI

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
JBI
```

Member CPYSYS

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
SYS
```

Definition of a file name

Member DCDALCTB

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
DEFINE(SHRFILES)
DEFINE(ALOG)
ALLOC F(DDSCT) DA('ALOG.DB.SCT') SHR
ALLOC F(DDBKS) DA('ALOG.DB.BKS') SHR
END
```

A-LOG Data Pool definition for A-LOG Terminal Facility

Notes: ASPL : Report Data Pool for A-SPOOL Terminal Facility
AMAL : Report Data Pool for A-SPOOL/MAIL Terminal Facility
ALOG : A-LOG Data Pool for A-LOG Terminal Facility

Member DCDTERM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
T=I3270,L=4096
```

The terminal type, Kanji code (non-Kanji character), screen TPUT buffer length used by A-LOG Terminal Facility

Member DCDTERMF

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
T=F6650,K=JEF,L=4096
```

The terminal type, Kanji code (Fujitsu), screen TPUT buffer length used by A-LOG Terminal Facility

Member DCDTERMH

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
T=H560/20,K=KEIS,L=4096
```

The terminal type, Kanji code (Hitachi), screen TPUT buffer length used by A-LOG Terminal Facility

Member DCDTERMI

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
T=I3270,K=IBM,L=4096
```

The terminal type, Kanji code (IBM), screen TPUT buffer length used by A-LOG Terminal Facility

Member DCDXIOTB

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
* ------  FOR IBM TERMINAL  ------
I3270   DCDXIOST
* ------  FOR HITACHI TERMINAL ----
H560/20 DCDXIO1T
* ------  FOR FUJITSU TERMINAL (H6650-I,II,III)
F6650   DCDXIOST
```

Terminal I/O modules used by A-LOG Terminal Facility

Member DDCATJ

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
EXT=CATJLOG,SYSUT1=YES
```

Job log storage module name and whether to output job log record

Member DDCATS

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
EXT=CATSLOG,SYSUT1=YES
```

SYSLOG storage module name and whether to output SYSLOG record

Member DDCNTL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
PRINT JOB=ASPLVTAM,MAKED=941204,MAKET=093222
```

Member DDJEXT

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
EXT=ASPJEXT               ASPLJLOG CATALOG USER EXIT ROUTINE
```

Member name of job log exit module

Member DDJLOG

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015006PS0002    SSSODSN
LLJCLLST                015006PS0003    SSSODSN
LSSYSMSG                015006PS0004    SSSODSN
LJJOBST   021007JEM373I                                          002008
LJJOBET   021007JEM395I                                          002008
LJJCLER   021007JDJ452I
LJFAILED  021007JDJ453I
LSJOBSD   002007JDJ375I    008/ START                              005
LSJOBED   002007JDJ376I    008/ STOP                               005
LSCOND    002007JDJ142I    010COND CODE                            004
LSSABEND  002007JDJ472I    007SYSTEM=                              003
LSUABEND  002007JDJ472I    005USER=                                004
LOSYSOUT                017004          SSSODSN
```

Job log message definition parameter

Member DDJLOGF

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015006PS0002    SSSODSN
LLJCLLST                015006PS0003    SSSODSN
LSSYSMSG                015006PS0004    SSSODSN
LJJOBST   021007JEM373I                                          002008
LJJOBET   021007JEM395I                                          002008
LJJCLER   021007JDJ452I
LJFAILED  021007JDJ453I
LJCANCEL  021007JDJ251I
LSJOBSD   002007JDJ375I    008/ START                              005
LSJOBED   002007JDJ376I    008/ STOP                               005
LSCOND    002007JDJ142I    010COND CODE                            004
LSSABEND  002007JDJ472I    007SYSTEM=                              003
LSUABEND  002007JDJ472I    005USER=                                004
LOSYSOUT                017004          SSSODSN
```

Job log message definition parameter

Member DDJLOGH

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015006PS0002    SSSODSN
LLJCLLST                015006PS0003    SSSODSN
LSSYSMSG                015006PS0004    SSSODSN
LJJOBST   021007$JSS463                                          002008
LJJOBET   021007$JSS467                                          002008
LJJCLER   021007JDJ285I
LJFAILED  021007JDJ491I
LJCANCEL  021007JDJ276I
LSJOBSD   002007JDJ797I    008) START                              008
LSJOBED   002007JDJ798I    008) STOP                               008
LSCOND    002007JDJ450I    010,RTN CODE=                           004
LSSABEND  002007JDJ451I    008DUE TO S                             003
LSUABEND  002007JDJ451I    002 U                                   004
LOSYSOUT                017004          SSSODSN
```

Job log message definition parameter

Member DDJLOGH4

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015006PS0002    SSSODSN
LLJCLLST                015006PS0003    SSSODSN
LSSYSMSG                015006PS0004    SSSODSN
LJJOBST   021007JDJ275I                                          002008
LJJOBET   021007JDJ490I                                          002008
LJJCLER   021007JDJ285I
LJFAILED  021007JDJ491I
LJCANCEL  021007JDJ276I
LSJOBSD   002007JDJ797I    008) START                              008
LSJOBED   002007JDJ798I    008) STOP                               008
LSCOND    002007JDJ450I    010,RTN CODE=                           004
LSSABEND  002007JDJ451I    008DUE TO S                             003
LSUABEND  002007JDJ451I    002 U                                   004
LOSYSOUT                017004          SSSODSN
```

Job log message definition parameter

Member DDJLOGI

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015006PS0002    SSSODSN
LLJCLLST                015006PS0003    SSSODSN
LSSYSMSG                015006PS0004    SSSODSN
LJJOBST   021008$HASP373                                          002008
LJJOBET   021008$HASP395                                          002008
LJJCLER   021007IEF452I
LJFAILED  021007IEF453I
LJCANCEL  021007IEF251I
LSJOBSD   002007IEF375I    008/ START                               005
LSJOBED   002007IEF376I    008/ STOP                                005
LSCOND    002007IEF142I    010COND CODE                             004
LSSABEND  002007IEF472I    007SYSTEM=                               003
LSUABEND  002007IEF472I    005USER=                                 004
LOSYSOUT                017004          SSSODSN
```

Job log message definition parameter

Member DDJLOGIE

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                015008P0000002  SSSODSN
LLJCLLST                015008P0000003  SSSODSN
LSSYSMSG                015008P0000004  SSSODSN
LJJOBST   021008$HASP373              MVS/ESA
LJJOBET   021008$HASP395                                          002008
LJJCLER   021007IEF452I
LJFAILED  021007IEF453I
LJCANCEL  021007IEF251I
LSJOBSD   002007IEF375I    008/ START                               005
LSJOBED   002007IEF376I    008/ STOP                                005
LSCOND    002007IEF142I    010COND CODE                             004
LSSABEND  002007IEF472I    007SYSTEM=                               003
LSUABEND  002007IEF472I    005USER=                                 004
LOSYSOUT                019004          SSSODSN
```

Job log message definition parameter

Member DDJLOGI3

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG                  008JESMSGLG  SSSODSN
LLJCLLST                  006JESJCL    SSSODSN
LSSYSMSG                  008JESYSMSG  SSSODSN
LJJOBST   021008$HASP373
LJJOBET   021008$HASP395                                          002008
LJJCLER   021007IEF452I                                           002008
LJFAILED  021007IEF453I
LJCANCEL  021007IEF251I
LSJOBSD   002007IEF375I    008/ START                                 005
LSJOBED   002007IEF376I    008/ STOP                                  005
LSCOND    002007IEF142I    010COND CODE                               004
LSSABEND  002007IEF472I    007SYSTEM=                                 003
LSUABEND  002007IEF472I    005USER=                                   004
LOSYSOUT                  004D000       SSSODSN
```

Job log message definition parameter

Member DDJLOGI8

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
LLJOBLOG               015008PS000002  SSSODSN
LLJCLLST               015008PS000003  SSSODSN
LSSYSMSG               015008PS000004  SSSODSN
LJJOBST   021008$HASP373             IBM3090
LJJOBET   021008$HASP395                                          002008
LJJCLER   021007IEF452I                                           002008
LJFAILED  021007IEF453I
LJCANCEL  021007IEF251I
LSJOBSD   002007IEF375I    008/ START                                 005
LSJOBED   002007IEF376I    008/ STOP                                  005
LSCOND    002007IEF142I    010COND CODE                               004
LSSABEND  002007IEF472I    007SYSTEM=                                 003
LSUABEND  002007IEF472I    005USER=                                   004
LOSYSOUT               019004          SSSODSN
```

Job log message definition parameter

Member DDJLOGP

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
*----  SAMPLE OF JOBLOG-ID SELECTION ----
* JOB-ID  CLASS CODE    JOB   JOBNAMEND
* ........000LEN.... ...COLLENEQ=X;GE=X;LE=X,NE=X;GT=X;LT=X;AND;OR;
* PSJOBLOGA 000003JKU     007002LE=19,AND,GE=00
* PSJOBLOGAS000002JK      001003EQ=ASP* PSJOBLOG99000001J       000000
PSJOBLOG  000001J      000000
```

Job definition parameter

Member DDJPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
JTUL  JOBSEPO=YES,JOBSEPC=YES,OUTSEPS=YES,UNIT=TAPE
```

Separator control statement

Member DDJUTL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
EXT=ALOGJEXT          <= ASPLJLOG,USLP ; JCL ANALIZE EXIT
```

Member name of JCL modification exit module

Member DDSCNTL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
INTERVAL=1440,CLASS=S,SYSLOGID=SYSLOG
```

Control card for cataloging SYSLOG

Member DDSPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
JUTL UNIT=TAPE,OUTSEPS=YES,
JUTL TIMEC=26,JNOC=38,JNOL=8,JCOML=3,TCOM=:,JCOM=ZERO
```

Member DEFDPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  DEFINE CLUSTER                          -
          (NAME(ALOG.DBDATA)              -
          VOLIMES(AL0G01)                 -
          CYLINDERS(002 5)                -
          RECORDSIZE(80 3000)             -
          KEYS(6 0)                       -
          SHAREOPTIONS(3 3)               -
          FILE(VSAMFILE)                  -
          IMBED                           -
          REPLICATE                       -
          FREESPACE(30 20)                -
          UNIQUE )                        -
         DATA                             -
          (NAME(ALOG.DBDATA.DATA)         -
           CONTROLINTERVALSIZE(4096))     -
         INDEX                            -
          (NAME(ALOG.DBDATA.INDEX)        -
           CONTROLINTERVALSIZE(4096))
```

VSAM Command for Allocating Data Part of A-LOG Management Database

Member DEFIPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  DEFINE CLUSTER                          -
          (NAME(ALOG.DBIDEX)              -
          VOLIMES(AL0G01)                 -
          CYLINDERS(001 2)                -
          RECORDSIZE(80 3000)             -
          KEYS(52 0)                      -
          SHAREOPTIONS(3 3)               -
          FILE(VSAMFILE)                  -
          IMBED                           -
          REPLICATE                       -
          FREESPACE(50 20)                -
          UNIQUE )                        -
         DATA                             -
          (NAME(ALOG.DBIDEX.DATA)         -
           CONTROLINTERVALSIZE(4096))     -
         INDEX                            -
          (NAME(ALOG.DBIDEX.INDEX         -
           CONTROLINTERVALSIZE(4096))
```

VSAM Command for Allocating Index Part of A-LOG Management Database

Member DLTDPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  DELETE  ALOG.DBDATA          -
          CLUSTER
```

VSAM command for deleting data part of A-LOG Management Database

Member DLTIPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  DELETE  ALOG.DBIDEX          -
          CLUSTER
```

VSAM command for deleting index part of A-LOG Management Database

Member DMPJOB

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
DUMP RID=JOBLOG
```

Criteria for selecting job logs to be dumped

Member DMPSYS

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
DUMP RID=SYSLOG
```

Criteria for selecting SYSLOGs to be dumped

Member FRMC3350

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09442      L
```

Format parameter for index part of A-LOG Data Pool on IBM 3350 or equivalent

Member FRMC3380

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09076      L
```

Format parameter for index part of A-LOG Data Pool on IBM 3380 or equivalent

Member FRMC3390

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  10796      L
```

Format parameter for index part of A-LOG Data Pool on IBM 3390 or equivalent

Member FRMC9345

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09345      L
```

Format parameter for index part of A-LOG Data Pool on IBM 9345 or equivalent

Member FRMK3350

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09442      L
```

Format parameter for data part of A-LOG Data Pool on IBM 3350 or equivalent

Member FRMK3380

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09076      L
```

Format parameter for data part of A-LOG Data Pool on IBM 3380 or equivalent

Member FRMK3390

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  10796      L
```

Format parameter for data part of A-LOG Data Pool on IBM 3390 or equivalent

Member FRMK9345

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  09345      L
```

Format parameter for data part of A-LOG Data Pool on IBM 9345 or equivalent

Member FRMCSECR

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
K1201012      L
```

Format Parameter for Security Management Table

Member FRMPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  04820000001L
```

Format Parameter for Recovery File

Member FRMSECRL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
R  01024      L
```

Format Parameter for Security Log File X and Y

Member HLGPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
DATASPACE NAME=HOTLOG,SYSID=ABAS,BLKCOUNT=420,BLKSIZE=4096
FILE NAME=RCV,BLKCOUNT=2
FILE NAME=SET,BLKCOUNT=400
FILE NAME=USR,BLKCOUNT=1
```

Format Parameter for Hot Log File

Member LDDTPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(BKDT)                -
        OUTFILE(DBDT)
```

VSAM command for re-loading data part of A-LOG Management Database

Member LDIXPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(BKIX)                -
        OUTFILE(DBIX)
```

VSAM command for re-loading index part of A-LOG Management Database

Member LOADJAM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ABSVIOS FILE=JAM
```

File definition for A-LOG Management Database

Member LOADJBI

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ABSVIOS FILE=JBI
```

File definition for A-LOG Management Database

Member LOADDMM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ABSVIOS FILE=DMM
```

File definition for A-LOG Management Database

Member LOADSYS

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ABSVIOS FILE=SYS
```

File definition for A-LOG Management Database

Member LOADPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(BKIX)                -
        OUTFILE(DBIX) REPLACE
  REPRO INFILE(BKDT)                -
        OUTFILE(DBDT) REPLACE
```

VSAM commands for re-loading A-LOG Management Database

Member PRTPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  VERIFY FILE(DD1)
  PRINT INFILE(DD1)
```

VSAM commands for printing A-LOG Management Database

Member RSELPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RESTSEL  MODE=ALOG,KEEPD=1,KEEPI=5,JOBNM=xxxxxxxx
```

Parameters for restore job generation

Member RUNDMP1

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 SORT FIELDS=(7,8,CH,A,15,8,CH,A,125,8,CH,A,65,8,CH,A,
              95,22,CH,A)
 RECORD TYPE=F,LENGTH=794
 END
```

Sort parameters

Member RUNDMP2

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 SORT FIELDS=(7,8,CH,A,15,8,CH,A,125,8,CH,A,65,8,CH,A,
              95,22,CH,A)
 RECORD TYPE=F,LENGTH=794
 END
```

Sort parameters

Member RUNDMP3

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 SORT FIELDS=(125,8,CH,A)
 RECORD TYPE=F,LENGTH=794
 END
```

Sort parameters

Member RUNDMP4

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 SORT FIELDS=(125,8,CH,A)
 RECORD TYPE=F,LENGTH=794
 END
```

Sort parameters

Member RUNLOG5

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
 SORT FIELDS=(725,2,CH,A,1,22,CH,A,105,8,CH,A,53,8,CH,A)
 END
```

Sort parameters

Member RUNSPM2

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
USMS NRQE=10,NUTL=85,NUTI=01,NBLK=1060,NJQE=50,NCQE=20
USMS DBM=ALGVBAS,NATB=5
```

This member contains A-LOG Monitor startup parameter. The values assigned to these keywords affect the region size required by A-LOG Monitor and the size of CSA area to be reserved. Refer to "**1.2 A-LOG System Operation Requirements**" for further details. These keywords are described in Table 6.5.

Keyword	Description	Default	Maximum
DBM	Fixed as 'ASPVBAS'.	—	—
NATB	Size of the CSA area to be reserved. Reserved CSA size = NATB x 4 Kbytes.	5	32767
NCQE	Maximum number of A-LOG terminals and A-LOG utilities that can be used simultaneously.	20	32767
NUSQ	Maximum number of A-LOG terminals and A-LOG utilities that can be used simultaneously.	20	32767
NMAQ	Maximum number of A-LOG terminals that can be used simultaneously	20	32767
NMQE	Maximum number of A-LOG terminals that can be used simultaneously	20	32767
NUTI	Increment value for triggering another warning message to the system console after NUTL value has been exceeded. When the NUTL value has been exceeded, SPM009A message is displayed for each increment of the NUTI value.	01	99
NUTL	High limit of the A-LOG Management Database space utilization to trigger the warning message to the system console. SPM009A message is displayed when this value is exceeded.	85	99
NJQE	Maximum number of job logs and SYSLOGs in A-LOG database which require exclusive processing.	50	32767
NRQE	Maximum number of job logs and SYSLOGs which require exclusive processing during storage.	10	32767

Table 6.6: Keywords in Member RUNSPM2

Member RUNSPM3

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  VERIFY  FILE(DBIX)
  VERIFY  FILE(DBDT)
```

VSAM commands for verifying A-LOG Management Database

Member SCTINIT

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
SCTRUN PROGRAM=BKSINITL                <- BKS DB INITIALIZE
```

Program name for initializing a Data Pool

Member SCTREP

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
SCTRUN PROGRAM=BKSREP
```

Program name for generating a Data Pool report

Member SORTDATA

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
     SORT FIELDS=(5,6,A),
          FORMAT=CH
```

Sort parameters

Member SORTIDX1

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
     SORT FIELDS=(5,2,A,57,4,A),
          FORMAT=CH
```

Sort parameters

Member SORTIDX2

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
      SORT FIELDS=(5,52,A),
           FORMAT=CH
```

Sort parameters

Member ULDDMP

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  VERIFY FILE(DD1)
  REPRO INFILE(DD1)                -
        OUTFILE(DD2)               -
        FROMKEY(X'006E00000000')
```

VSAM parameters for conversion of A-LOG Management Database

Member ULDDPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(DBDT)
        OUTFILE(BKDT)
```

VSAM parameters for unloading data part of A-LOG Management Database

Member ULDIPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(DBIX)
        OUTFILE(BKIX)
```

VSAM parameters for unloading index part of A-LOG Management Database

Member UNLDPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  REPRO INFILE(DBIX)
        OUTFILE(BKIX)
  REPRO INFILE(DBDT)
        OUTFILE(BKDT)
```

VSAM parameters for unloading A-LOG Management Database

Member USMSJRST

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
USMS UNIT=TAPE
```

Restore control parameters

Member USMSLRST

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
USMS UNIT=TAPE
```

Restore control parameters

Member VERPARM

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
  VERIFY  FILE(DBIX)
  VERIFY  FILE(DBDT)
```

VSAM commands for verifying A-LOG Management Database

Member WTRJUTL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
//WTRJTUL JOB ,ALOG,CLASS=A,MSGCLASS=A
//*********************************************************
//*                                                       *
//*   WTRJUTL    JOBLOG PRINT UTILITY                     *
//*              DDCNTL ...   PRINT JOB SELECTION CARD    *
//*              DDPARM ...   PRINT SEPARATOR OPTION      *
//*                                                       *
//*********************************************************
//WTRJTUL  EXEC PGM=WTRJUTL,PARM=S@0000,REGION=2048K
//STEPLIB  DD  DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD  DSN=ALOG.DB.SCT,DISP=SHR
//DDBKS    DD  DSN=ALOG.DB.BKS,DISP=SHR
//* INPUT MT USED
//* DDIN   DD  DSN=ASPWE,DISP=SHR,
//*         UNIT=(TAPE,,DEFER),VOL=SER=ASPWE
//DDOUT    DD  SYSOUT=*,
//         DCB=(LRECL=133,BLKSIZE=7980,RECFM=FBM)
//SORTIN   DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTOUT  DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO1 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO2 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO3 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//DBSNAP   DD  DUMMY
//DDSNAP   DD  DUMMY               /* DYNALLOC MESSAGE */
//DDKANJI  DD  DUMMY
//USMSOUT  DD  SYSOUT=*
//SYSOUT   DD  DUMMY
//SYSUDUMP DD  SYSOUT=*
//SYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(1,1))
//DDPARM   DD  DSN=ALOG.V0380.PARMLIB(DDJPARM),DISP=SHR
//DDCNTL   DD  DSN=ALOG.V0380.PARMLIB(DDCNTL),DISP=SHR
```

Default JCL for WTRJUTL utility

Member WTRSUTL

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
//WTRSTUL JOB ,ALOG,CLASS=A,MSGCLASS=A
//*************************************************************
//*                                                           *
//*   WTRSUTL    SYSLOG PRINT UTILITY                         *
//*              DDCNTL ...   PRINT JOB SELECTION CARD        *
//*              DDPARM ...   PRINT SEPARATOR OPTION          *
//*                                                           *
//*************************************************************
//WTRSTUL  EXEC PGM=WTRSUTL,PARM=S@0000,REGION=2048K
//STEPLIB  DD  DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD  DSN=ALOG.DB.SCT,DISP=SHR
//DDBKS    DD  DSN=ALOG.DB.BKS,DISP=SHR
//* INPUT MT USED
//* DDIN   DD  DSN=ASPWE,DISP=SHR,
//*         UNIT=(TAPE,,DEFER),VOL=SER=ASPWE
//DDOUT    DD  SYSOUT=*,
//         DCB=(LRECL=133,BLKSIZE=7980,RECFM=FBM)
//SYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(1,1))
//SORTIN   DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTOUT  DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO1 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO2 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//SORTWKO3 DD  UNIT=SYSDA,SPACE=(TRK,(10,5))
//USMSOUT  DD  SYSOUT=*
//SYSOUT   DD  DUMMY
//SYSUDUMP DD  SYSOUT=*
//DBSNAP   DD  DUMMY
//DDSNAP   DD  DUMMY               /* DYNALLOC MESSAGE */
//DDKANJI  DD  DUMMY
//DDPARM   DD  DSN=ALOG.V0380.PARMLIB(DDSPARM),DISP=SHR
//DDCNTL   DD  *
/*
//
```

Default JCL for WTRJUTL utility

6.3 Terminal Screen Library

All A-LOG terminal screens are defined in A-LOG Terminal Screen Library. The standard dataset name of this library is 'ALOG.V0380.MAPLIB'. These screens are used by A-LOG Terminal Facility. Therefore, they should not be updated by the users. The following table highlights the screens defined in A-LOG Terminal Screen Library:

Member Name		Screen	
English Screen	Kanji Screen	Number	Screen Name
ALGAJ0	ALGAJ0K	1.0.0	JOB/SYSLOG MASTER MANAGEMENT
ALGAJ1	ALGAJ1K	1.1.0	JOB/SYSLOG MASTER MAINTENANCE
ALGAJ2	ALGAJ2K	1.1.1	JOB/SYSLOG MASTER DIRECTORY (INQ/UPD)
ALGCJ3	ALGCJ3K	1.1.2	JOB/SYSLOG MASTER (INQ/UPD)
ALGB00	ALGB00K	9.00	SYSTEM MANAGEMENT
ALGB30	ALGB30K	2.2.0	DUMP MT INQUIRY
ALGC10	DCDX00K	9.1.0	TERMINAL/PF-KEYS SETTING
ALGF11	ALGF11K	EMSG	ERROR MESSAGE
ALGH00	ALGH00K		HELLO
ALGI00	ALGI00K	0.0.0	INITIAL MENU
ALGL00	ALGL00K	2.0.0	JOBLOG MANAGEMENT
ALGL20	ALGL20K	2.1.0	JOBLOG INDEX MAINTENANCE
ALGL21	ALGL21K	2.1.1.1	JOBLOG DIRECTORY (INQ/UPD)
ALGL22	ALGL22K	2.1.4	JOBLOG INQUIRY
ALGL23	ALGL23K	2.1.2.1	JOBLOG INDEX -NO.1- (INQ/UPD)
ALGL24	ALGL24K	2.1.3	JOBLOG INDEX -NO.2- (INQ/UPD)
ALGL25	ALGL25K	2.1.2.2	JOBLOG INDEX -NO.1- (INQ/UPD)
ALGL26	ALGL26K	2.1.1.2	JOBLOG DIRECTORY (INQ/UPD)
ALGS00	ALGS00K	3.0.0	SYSLOG MANAGEMENT
ALGS10	ALGS10K	3.1.0	SYSLOG DATA SELECT
ALGS11	ALGS11K	3.1.1	SYSLOG DATA INQUIRY

Table 6.7.a: Screens of A-LOG Terminal Facility

Member Name		Screen	
English Screen	Kanji Screen	Number	Screen Name
ALGS20	ALGS20K	3.2.0	SYSLOG INDEX MAINTENANCE
ALGS21	ALGS21K	3.2.1	SYSLOG DIRECTORY (INQ/UPD)
ALGS23	ALGS23K	3.2.2.1	SYSLOG INDEX -NO.1- (INQ/UPD)
ALGS25	ALGS25K	3.2.2.2	SYSLOG INDEX -NO.2- (INQ/UPD)
ALGX10	ALGX10K	9.1.0	TERMINAL/PF-KEYS SETTING

Table 6.7.b: Screens of A-LOG Terminal Facility

6.4 Cataloged Procedure

All A-LOG cataloged procedures are stored in SYS1.PROCLIB dataset as a standard. These procedures are required for the operation of A-LOG system. Refer to "**Appendix 2: Contents of A-LOG Cataloged Procedures**". The following table highlights a list of the cataloged procedures and their functions:

Procedure	Description of Function
ALOG	To start A-LOG Monitor.
ALOGBLOG	To create Job Log Audit Trail List and Job Log Index Release File.
ALOGDATE	To set A-LOG Operation Date.
ALOGDLT1	To delete job log or SYSLOG data.
ALOGDLT4	To delete Dump MT Information record.
ALOGDLT5	To delete job log or SYSLOG indexes.
ALOGDMPC	To copy Dump MTs.
ALOGDMP1	To create files for use by ALOGDMP2 utility and ALOGDMP3 utility.
ALOGDMP2	To dump job logs and delete job logs.
ALOGDMP3	To delete job logs.
ALOGDMP4	To delete the files created by ALOGDMP1 utility.
ALOGIBKS	To initialize A-LOG Data Pool.
ALOGICHK	To check for unmatched records in A-LOG Management Database.
ALOGIUPD	To re-create A-LOG Management Database.
ALOGJDMP	To dump job logs.
ALOGJLOG	To store job logs.
ALOGJMST	To maintain master files.
ALOGJRST	To restore job logs.
ALOGLBKS	To load A-LOG Data Pool.
ALOGLDMP	To dump SYSLOG.
ALOGLLOG	To create a SYSLOG Audit Trail List and generate a SYSLOG Index Release File.
ALOGLOAD	To load A-LOG Management Database.

Table 6.8.a: List of A-LOG Cataloged Procedures

Procedure	Description of Function
ALOGLRST	To restore SYSLOG.
ALOGREP	To generate A-LOG database utilization reports.
ALOGRSEL	To restore job logs and SYSLOGs that have been scheduled for restoration.
ALOGSECR	To maintain Security Management Table.
ALOGSLOG	To store SYSLOGs.
ALOGUBKS	To unload A-LOG Data Pool.
ALOGUNLD	To unload A-LOG Management Database.

Table 6.8.b: List of A-LOG Cataloged Procedures

6.5 Command Procedure

An A-LOG command procedure is provided for invoking A-LOG Terminal Facility. A-LOG command procedure supports the English-character screen type as well as Kanji-character screen type. Any one of these screen types may be chosen according to the type of terminal. Refer to "**Appendix 3: Contents of Command Procedure**" for further information.

A-LOG command procedure is usually stored in SYS1.CMDPROC. Depending on the screen type desirable, one of the following commands may be used to invoke A-LOG Terminal Facility:

1) For Kanji screens : ALOG

2) For alphanumeric screens : ALOGA

A-LOG command procedure name can be changed during installation of A-LOG system. Also, the first screen to be displayed upon invoking of A-LOG Terminal Facility can be customized. Refer to "**2.2.7 Create Command Procedure**" for further details.

Chapter 7

A-LOG User Release Files & Exit Routines

7.1 A-LOG User Release Files

7.1.1 Introduction

A-LOG system provides two user release files as follows:

- Job Log Index Release File
- SYSLOG Index Release File

The Job Log Index Release File is created by executing ALOGBLOG utility. The SYSLOG Index Release File is created by executing ALOGLLOG utility. The Job Log Index Release File can be referenced from a user exit routine. However, when it is referenced from a user exit routine, "packed" data type is converted to "binary" data type.

7.1.2 Job Log Index Release File

Record Length: 788 Record Type: Fixed Length, Blocked

Position (Hex)	Position (Decimal)	Length	Type	Description
0	0	6	Numeric	Job Log Store Date (*yymmdd*)
6	6	8	Alphanumeric	Job Name
E	14	8	Alphanumeric	Job Number
16	22	12	Numeric	Job Start CPU Date/Time (*yymmddhhmmss*)
22	34	12	Numeric	Job End CPU Date/Time(*yymmddhhmmss*)
2E	46	6	Alphanumeric	Job End Code: • *nnnn* → Normal End • JCLER → JCL Error • S*nnn* → System Abend • U*nnnn* → User Abend
34	52	8	Alphanumeric	Dataset Code
3C	60	6	Numeric	Job Log Store Start Operation Date (*yymmdd*)
42	66	6	Numeric	Job Log Store Start CPU Date (*yymmdd*)

Table 7.1.a: Layout of Job Log Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
48	72	6	Numeric	Job Log Store Start Time (*hhmmss*)
4E	78	6	Numeric	Job Log Store End Operation Date (*yymmdd*)
54	84	6	Numeric	Job Log Store End CPU Date (*yymmdd*)
5A	90	6	Numeric	Job Log Store End Time (*hhmmss*)
60	96	4	Packed	Number of Stored Pages
64	100	4	Packed	Number of Stored Lines
68	104	8	Alphanumeric	Internal Job Log Identification Number
70	112	1	Numeric	Storage Status: • 1 → Currently being stored • 2 → Storage terminated abnormally • 3 → Storage ended normally
71	113	1	Numeric	Data Storage Status: • 0 → Not stored • 1 → Currently being stored • 2 → Storage terminated abnormally • 3 → Storage ended normally
72	114	1	Numeric	Print Status: • 0 → Not printed • 1 → Currently being printed • 2 → Printing terminated abnormally • 3 → Printing ended normally
73	115	1	Numeric	Storage Location: • 0 → None • 1 → A-LOG Database • 2 → Reserved • 3 → Dump MT • 4 → Reserved • 5 → Reserved
74	116	2	Alphanumeric	SYSOUT Class

Table 7.1.b: Layout of Job Log Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
76	118	8	Alphanumeric	SYSOUT Form Code
7E	126	8	Alphanumeric	SYSOUT User Writer Name
86	134	8	Alphanumeric	SYSOUT DEST Code
8E	142	8	Alphanumeric	SYSOUT DSID Code
96	150	24	Alphanumeric	SYSOUT Dataset Name
AE	174	2	Packed	Number of SYSOUT Datasets
B0	176	38	—	Unused
D6	214	6	Numeric	Job Log Master Valid Date (*yymmdd*)
DC	220	8	Alphanumeric	Job Log ID
E4	228	64	Alphanumeric	Job Log Comment
124	292	2	Alphanumeric	Job Attribute Class
126	294	2	Alphanumeric	Immediate Print Output Class
128	296	4	Packed	Dump Retention Days
12C	300	4	Packed	Delete Retention Days
130	304	32	—	Unused
150	336	64	Alphanumeric	User Comment
190	400	64	—	Unused
1D0	464	64	Alphanumeric	LBP Information
210	528	44	Alphanumeric	Dump MT Dataset Name
23C	572	10 x 10	Numeric	(Volume Number + Number of Blocks) x 10
2A0	672	4	Packed	Start Volume Number + Number of Blocks
2A4	676	4	Packed	End Volume Number + Number of Blocks
2A8	680	4	Packed	Number of Blocks Written
2AC	684	4	Packed	Number of Records Written
2B0	688	2	Packed	Tape Label

Table 7.1.c: Layout of Job Log Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
2B2	690	10	—	Unused
2BC	700	2	Packed	SYSOUT Dataset Sequence Number
2BE	702	2	Packed	Unused
2C0	704	4	Packed	Starting Page
2C4	708	4	Packed	Ending Page
2C8	712	4	Packed	Number of Pages for Dataset
2CC	716	4	Packed	Number of Lines for Dataset
2D0	720	2	Packed	Length of Line for Dataset
2D2	722	2	Alphabetic	Record Format of Dataset
2D4	724	2	Numeric	First 2 digits of Calendar Year of Job Log Store Date
2D6	726	2	Numeric	First 2 digits of Calendar Year of Job Start Date
2D8	728	2	Numeric	First 2 digits of Calendar Year of Job End Date
2DA	730	4	Numeric	First 2 digits of Calendar Year of Job Log Store Start Date
2DE	734	4	Numeric	First 2 digits of Calendar Year of Job Log Store End Date
2E2	738	2	Numeric	First 2 digits of Calendar Year of Job Log Master Valid Date
2E4	740	1	Numeric	Restore Flag
2E5	741	1	Numeric	Restore Status
2E6	742	8	Numeric	Restore Date (Operation Date)
2EE	750	4	Packed	Data Retention Days after Restoration
2F2	754	4	Numeric	Index Retention Days after Restoration
2F6	758	30	—	Unused

Table 7.1.d: Layout of Job Log Index Release File

7.1.3 SYSLOG Index Release File

Record Length: 788 Record Type: Fixed Length, Blocked

Position (Hex)	Position (Decimal)	Length	Type	Description
0	0	6	Numeric	Job Log Store Date (*yymmdd*)
6	6	8	Alphanumeric	Job Name
E	14	8	Alphanumeric	Job Number
16	22	12	Numeric	Job Start CPU Date/Time (*yymmddhhmmss*)
22	34	12	Numeric	Job End CPU Date/Time (*yymmddhhmmss*)
2E	46	6	Alphanumeric	Job End Code: • *nnnn* → Normal End • JCLER → JCL Error • S*nnn* → System Abend • U*nnnn* → User Abend
34	52	8	Alphanumeric	Dataset Code
3C	60	6	Numeric	Job Log Store Start Operation Date (*yymmdd*)
42	66	6	Numeric	Job Log Store Start CPU Date (*yymmdd*)
48	72	6	Numeric	Job Log Store Start Time (*hhmmss*)
4E	78	6	Numeric	Job Log Store End Operation Date (*yymmdd*)
54	84	6	Numeric	Job Log Store End CPU Date (*yymmdd*)
5A	90	6	Numeric	Job Log Store End Time (*hhmmss*)
60	96	4	Packed	Number of Stored Pages
64	100	4	Packed	Number of Stored Lines
68	104	8	Alphanumeric	Internal Job Log Identification Number

Table 7.2.a: Layout of SYSLOG Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
70	112	1	Numeric	Storage Status: • 1 → Currently being stored • 2 → Storage terminated abnormally • 3 → Storage ended normally
71	113	1	Numeric	Data Storage Status: • 0 → Not stored • 1 → Currently being stored • 2 → Storage terminated abnormally • 3 → Storage ended normally
72	114	1	Numeric	Print Status: • 0 → Not printed • 1 → Currently being printed • 2 → Printing terminated abnormally • 3 → Printing ended normally
73	115	1	Numeric	Storage Location: • 0 → None • 1 → A-LOG Database • 2 → Reserved • 3 → Dump MT • 4 → Reserved • 5 → Reserved
74	116	2	Alphanumeric	SYSOUT Class
76	118	8	Alphanumeric	SYSOUT Form Code
7E	126	8	Alphanumeric	SYSOUT User Writer Name
86	134	8	Alphanumeric	SYSOUT DEST Code
8E	142	8	Alphanumeric	SYSOUT DSID Code
96	150	24	Alphanumeric	SYSOUT Dataset Name
AE	174	2	Packed	Number of SYSOUT Datasets
B0	176	2	Packed	Number of Printings
B2	178	6	Numeric	Print Start Operation Date (*yymmdd*)

Table 7.2.b: Layout of SYSLOG Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
B8	184	6	Numeric	Print Start CPU Date (*yymmdd*)
BE	190	6	Numeric	Print Start Time (*hhmmss*)
C4	196	6	Numeric	Print End Operation Date (*yymmdd*)
CA	202	6	Numeric	Print End CPU Date (*yymmdd*)
D0	208	6	Numeric	Print End Time (*hhmmss*)
D6	214	6	Numeric	SYSLOG ID Valid Date (*yymmdd*)
DC	220	8	Alphanumeric	SYSLOG ID
E4	228	64	Alphanumeric	SYSLOG Comment
124	292	2	Alphanumeric	Job Attribute Class
126	294	2	Alphanumeric	Immediate Print Output Class
128	296	4	Packed	Dump Retention Days
12C	300	4	Packed	Delete Retention Days
130	304	32	—	Unused
150	336	64	Alphanumeric	User Comment
190	400	64	—	Unused
1D0	464	64	Alphanumeric	LBP Information
210	528	44	Alphanumeric	Dump MT Dataset Name
23C	572	10 x 10	Numeric	(Volume Number + Number of Blocks) x 10
2A0	672	4	Packed	Start Volume Number + Number of Blocks
2A4	676	4	Packed	End Volume Number + Number of Blocks
2A8	680	4	Packed	Number of Blocks Written
2AC	684	4	Packed	Number of Records Written
2B0	688	2	Packed	Tape Label
2B2	690	10	—	Unused
2BC	700	2	Packed	SYSOUT Dataset Sequence Number

Table 7.2.c: Layout of SYSLOG Index Release File

Position (Hex)	Position (Decimal)	Length	Type	Description
2BE	702	2	Packed	Unused
2C0	704	4	Packed	Starting Page
2C4	708	4	Packed	Ending Page
2C8	712	4	Packed	Number of Pages for Dataset
2CC	716	4	Packed	Number of Lines for Dataset
2D0	720	2	Packed	Length of Line for Dataset
2D2	722	2	Alphabetic	Record Format of Dataset
2D4	724	2	Numeric	First 2 digits of Calendar Year of SYSLOG Store Date
2D6	726	2	Numeric	First 2 digits of Calendar Year of Job Start Date
2D8	728	2	Numeric	First 2 digits of Calendar Year of Job End Date
2DA	730	4	Numeric	First 2 digits of Calendar Year of SYSLOG Store Start Date
2DE	734	4	Numeric	First 2 digits of Calendar Year of SYSLOG Store End Date
2E2	738	2	Numeric	First 2 digits of Calendar Year of SYSLOG Master Valid Date
2E4	740	1	Numeric	Restore Flag
2E5	741	1	Numeric	Restore Status
2E6	742	8	Numeric	Restore Date (Operation Date)
2EE	750	4	Packed	Data Retention Days after Restoration
2F2	754	4	Numeric	Index Retention Days after Restoration
2F8	758	30	—	Unused

Table 7.2.d: Layout of SYSLOG Index Release File

7.2 Job Log Storage Exit Routine

Function

This exit routine, if implemented, will receive control from A-LOG system when a job log is cataloged into the A-LOG database. Therefore, a user can use this exit routine to perform special processing if necessary. This exit routine can be designed to perform the following functions:

- Per-job read / search of the Job Log / Sysout data as a PS file are possible.
- Addition or modification to the information in Job Log Information area.
- Immediate print of a job log.

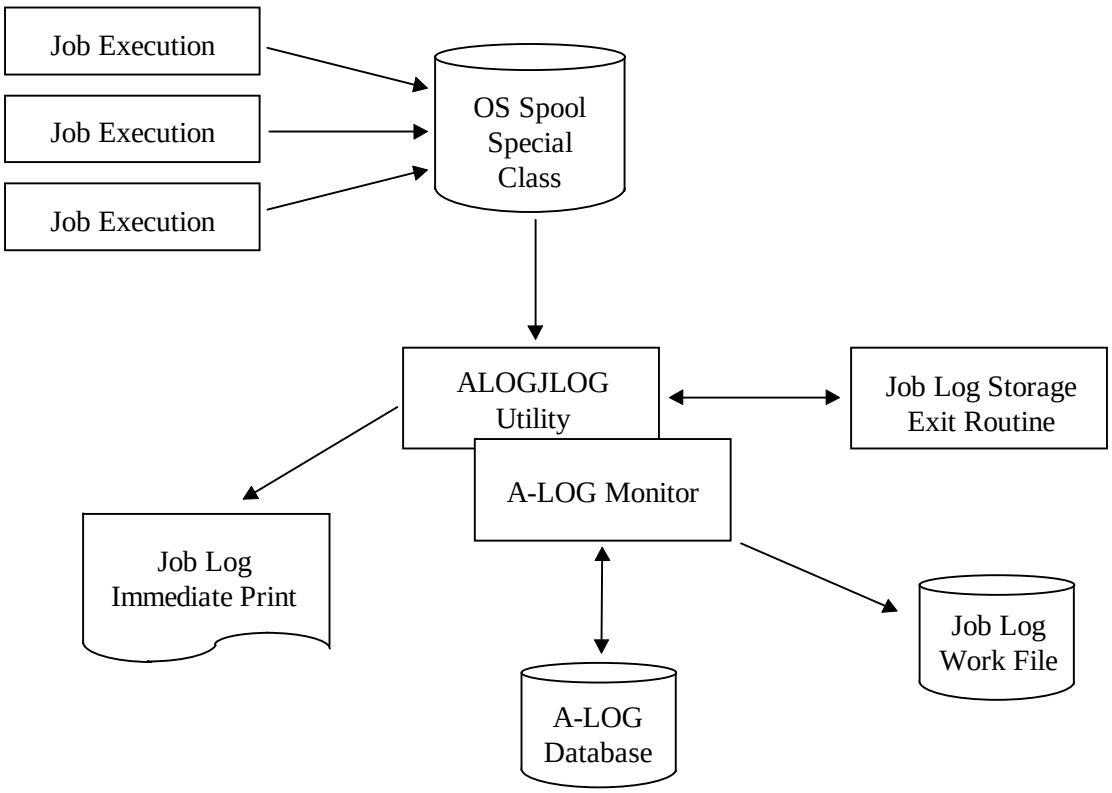


Fig. 7.1: Job Log Storage Exit Routine

Timing for Passing of Control

The exit routine is loaded and called upon by ALOGJLOG utility in the timing as shown below:

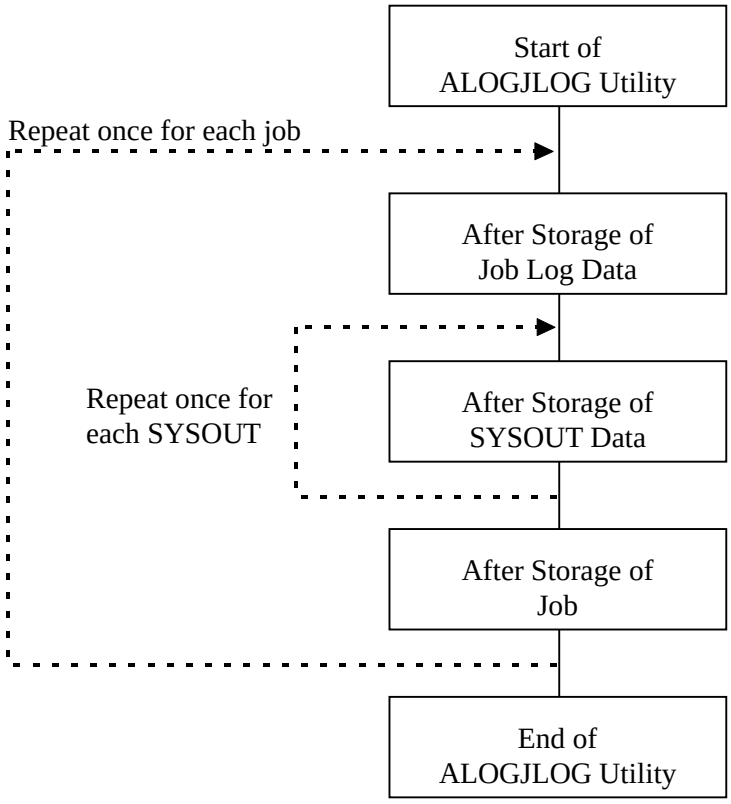


Fig. 7.2: Timing for Calling Job Log Storage Exit Routine

Call Timing	Description
Start of ALOGJLOG Utility	The exit routine is called when ALOGJLOG utility is started.
After Storage of Job Log Data	The exit routine is called when job log index is created, job log data is stored and job log work file is created.
After Storage of SYSOUT Data	The exit routine is called when index of SYSLOG data is created, data is stored and job log work file is created.
After Storage of Job	The exit routine is called when all storage processing for one job has ended.
End of ALOGJLOG Utility	The exit routine is called before ALOGJLOG utility is ended.

Table 7.3: Timing for Calling Job Log Storage Exit Routine

Job Log Storage Exit Routine Specification

Specify the load module name of the exit routine in DDJEXT DD statement of the execution JCLs of ALOGJLOG utility as follows:

```
//DDJEXT    DD *
EXT=load-module-name-of-exit-routine
```

Note: The load module name of the exit routine must not be greater than 8 alphanumeric characters. This load module must exist in the library specified in STEPLIB statement of the execution JCLs of ALOGJLOG utility.

Job Log Storage Exit Routine Interface

The calling convention between the exit routine and ALOGJLOG utility are based on standard OS calling convention. The following table highlights the usage of the registers:

		Input Interface	**Output Interface**
Registers	GR1	Parameter List Address	Parameter List Address
	GR0, GR2~12	Undefined	Undefined
	GR13	Register Save Area Address	Register Save Area Address
	GR14	Return Address	Return Address
	GR15	Job Log Storage Exit Routine Entry Address	Set as '0'
	Parameter List	Refer to Job Log Storage Exit Routine Parameters	—
	Return Code	—	Set in the Job Log Storage Exit Routine Parameter

Table 7.4: Usage of Registers by Job Log Storage Exit Routine

Job Log Storage Exit Routine Parameters

The contents of parameters passed to the exit routine are explained as follows:

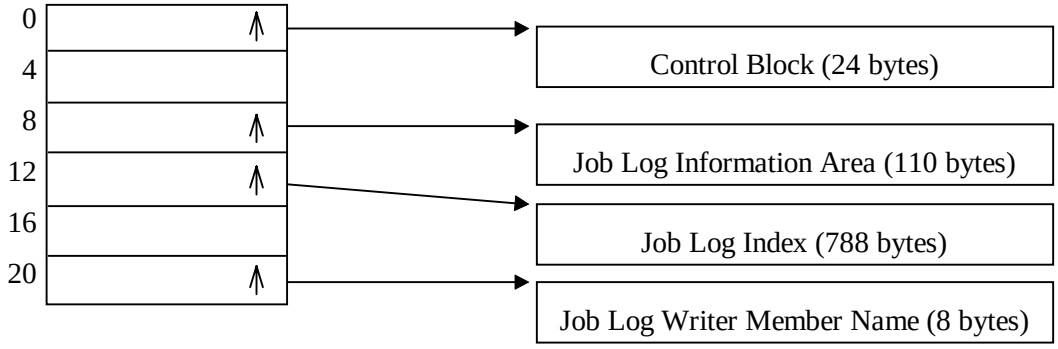


Fig. 7.3: Job Log Storage Exit Routine Parameter List

1) Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
0	0	Display Code	1	Alphabetic	Set the call timing of job log storage exit routine: • F → At the beginning of execution of job log storage utility • J → After storage of job log data • O → After storage of SYSOUT data • E → After storage of job • L → At the end of execution of job log storage utility
1	1	—	1	—	Unused.
2	2	Return Code	1	Alphabetic	Specify the storage of SYSOUT: • When Display Code is 'J': Δ → Store entire SYSOUT N → Do not store entire SYSOUT • When Display Code is 'O': Δ → Store the next SYSOUT N → Do not store the next SYSOUT

Δ → Blank

Table 7.5.a: Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
3	3	Print Code	1	Alphabetic	Specify immediate print option: • When Display Code is 'J': Δ → Immediate print option is set off P → Immediate print option is set on • When Display Code is 'O': Δ → Immediate print option is set off J → Submit the job log print utility and immediately print the data (See Note)
4	4	Job Name	8	Alphanumeric	If Display Code is 'J', 'O' or 'E', the job name currently being stored is set. If Display Code is 'F' or 'L', this item is blank.
12	C	Job Number	8	Alphanumeric	If Display Code is 'J', 'O' or 'E', the job number currently being stored is set. If Display Code is 'F' or 'L', this item is blank.
20	16	Immediate Print Selection Flag	1	Alphabetic	If Print Code is 'P', specify whether to print all data: • Δ → Print all job log data immediately • S → Select job log data to be printed immediately
21	17	Job Log Print Class	1	Alphabetic	If Immediate Print Selection Flag is 'S', specify 'P' to print job log.
22	18	JCL List Print Flag	1	Alphabetic	If Immediate Print Selection Flag is 'S', specify 'P' to print JCL list.
23	19	System Message Print Flag	1	Alphabetic	If Immediate Print Selection Flag is 'S', specify 'P' to print system message.

Δ → Blank

Table 7.5.b: Control Block Format

Note: It submits member name WTRJUTL in the library specified in DDWTR DD statement of the procedure for ALOGJLOG utility.

2) Job Log Information Area

When display code is 'F' or 'L', the job log information area is set to blanks. When display code is 'J', or 'O' or 'E', the job log information area is set as follows:

Position (Decimal)	Position (Hex)	Length	Type	Description
0	0	8	Alphanumeric	Job Name
8	8	8	Alphanumeric	Job Number
16	10	6	Numeric	Job Start Date
22	16	6	Numeric	Job End Date
28	1C	6	Numeric	Job Start Time
34	22	6	Numeric	Job End Time
40	28	6	Alphanumeric	Completion Code (Abend Code or Maximum Condition Code)
46	2E	64	Alphanumeric	User Comment

Table 7.6: Job Log Information Area

3) Job Log Index

Format of Job Log index is the same as the format of the Job Log index release file. Refer to "**7.1.2 Job Log Index Release File**".

4) Job Log Writer Member Name

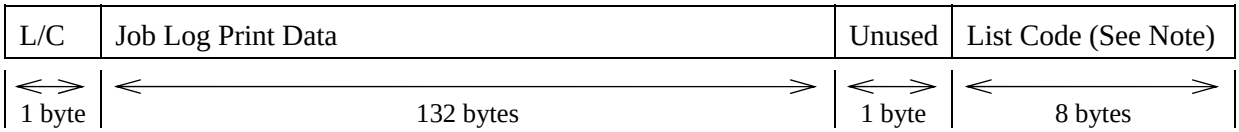
This is the member name of the Job Log Print utility which is submitted under the following conditions:

- Display Code='E'
- Print Code ='J'

Job Log Work File

Job log records are written to Job Log Work file, which is specified in SYSUT1 DD statement of ALOGJLOG procedure. It can be read from a job log storage exit routine. In order to read, perform direct OPEN/GET/CLOSE of DDNAME=SYSUT1 file in a job log storage exit routine.

- Record Format of Job Log Work File (LRECL=142)



Example of Coding a Job Log Storage Exit Routine

- Sample Program Function

  Immediate printing of job log when one of the following execution result of a job is encountered:

  - System abend - JOB FAILED
  - User abend - Cancel
  - JCL error

- Sample Program Processing

  If the job completion code is one of the following when the display code is 'J', 'P' will be returned as a print code. When a display code is not 'J', processing ends.

  - 'JCLER' - First character = 'S'
  - 'FAILED' - irst character = 'U'
  - 'CANCEL'

```
*************************************************************************
*        ASPJEXT        JOB LOG STORAGE EXIT ROUTINE                    *
*                       JOB (ABEND,JCLER,FAILED,CANCEL) WITH PRINT OUT  *
*************************************************************************
R0       EQU   0                    PARAM ADDR
R1       EQU   1
R2       EQU   2
R3       EQU   3
R4       EQU   4
R5       EQU   5
R6       EQU   6
R7       EQU   7
R8       EQU   8
R9       EQU   9
R6       EQU   6
R7       EQU   7
R8       EQU   8
R9       EQU   9
R10      EQU   10
R11      EQU   11
R12      EQU   12                   ASPJEXT BASE
R13      EQU   13
R14      EQU   14
R15      EQU   15
```

```
*--------------- USER EXIT PARAMETER LIST -----------------------------*
*        UEXTPARM DS    0F
*        UEXTACB  DC    A(UEXTCB)
*        UEXTADCB DC    A(0)
*        UEXTAINF DC    A(UEXTINF)
*--------------- USER EXIT CONTROL BLOCK ------------------------------*
UEXTCB   DSECT
UEXTFC   DC    C' '                        * USER EXIT FUNCTION
UEXTRV   DC    C' '                        * USER EXIT RESERVED
UEXTRC   DC    C' '                        * USER EXIT RETURN CODE
UEXTPC   DC    C' '                        * USER EXIT PRINT CODE
UEXTJOB  DC    CL8' '                      * USER EXIT JOB NAME
UEXTJID  DC    CL8' '                      * USER EXIT JOB NUMBER
UEXTCLS  DC    C' '                        * USER EXIT PRINT FLAG
UEXTCLS1 DC    C' '                        * UEXTCLS=S  JOBLOG PRINT
UEXTCLS2 DC    C' '                        * UEXTCLS=S  JCLLIST PRINT
UEXTCLS3 DC    C' '                        * UEXTCLS=S  SYSTEM-MSG PRINT
*
UEXTINF  DSECT
UEXTIJB  DC    CL8' '                      * JOB NAME
UEXTIJD  DC    CL8' '                      * JOB NUMBER
UEXTISD  DC    CL6' '                      * JOB START DATE
UEXTIST  DC    CL6' '                      * JOB START TIME
UEXTIED  DC    CL6' '                      * JOB END DATE
UEXTIET  DC    CL6' '                      * JOB END TIME
UEXTICD  DC    CL6' '                      * JOB END CODE
UEXTIUS  DC    CL20' '                     * JOB LOG USER COMMENT
UEXTIRV  DC    CL14' '                     * JOB LOG RESERVED
UEXTINFL EQU   *-UEXTINF
*
ASPJEXT   CSECT
          SAVE   (14,12)
          LR     R12,R15
          USING  ASPJEXT,R12
          ST     R13,SAVEAREA+4
          LR     R11,R13
          LA     R13,SAVEAREA
          ST     R13,8(,R11)
          B      A100
*
          DC     CL8'ASPJEXT'
          DC     CL8'&SYSDATE'
          DC     CL8'&SYSTIME'
SAVEAREA  DS     18F
          DC     CL8'ZAPLEVEL'
          DC     CL8'00000000'
ZAPAREA   DC     256CL1'Z'
*
```

```
A100      EQU    *
          USING  UEXTCB,R10
          USING  UEXTINF,R9
          L      R10,0(R1)                  * LOAD R10 = UEXTCB ADDR
          L      R9,8(R1)                   * LOAD R9  = UEXTINF ADDR
          CLI    UEXTFC,C'J'                * IF UEXTFC = C'J' ?
          BNE    A900                       *    NOT. RETURN
          MVI    UEXTPC,C' '                * CLEAR PRINT CODE
          CLI    UEXTICD,C'S'               * IF CODE = S999 ?
          BE     A190                       *    YES. SYSTEM ABEND
          CLI    UEXTICD,C'U'               * IF CODE = U9999 ?
          BE     A190                       *    YES. USER ABEND
          CLC    UEXTICD(6)=CL6'JCLER'      * IF CODE = JCLERR ?
          BE     A190                       *    YES. JCL ERROR
          CLC    UEXTICD(6)=CL6'FAILED'     * IF CODE = FAILED ?
          BE     A190                       *    YES. JOB FAILED ERROR
          CLC    UEXTICD(6)=CL6'CANCEL'     * IF CODE = CANCEL ?
          BE     A190                       *    YES. JOB CANCEL ERROR
          B      A900                       * RETURN
A190      DS     0H
          MVI    UEXTPC,C'P'                * SET C'P' TO UEXTPC
A900      DS     0H
          LA     R15,0
RETURN    DS     0H
          L      R13,SAVEAREA+4
          RETURN (14,12),RC=(15)
          LTORG
          END
```

7.3 JCL Modification Exit Routine

Function

This exit routine receives control from the following component which submits the WTRJUTL utility for execution:

- ALOGJLOG utility
- Print command from a terminal

By using this exit routine, a user can perform the following modifications to the JCLs of WTRJUTL utility before it is submitted for execution:

- Change job name
- Set accounting information
- Change job log output attributes, such as output class, DEST code, overlay, etc.

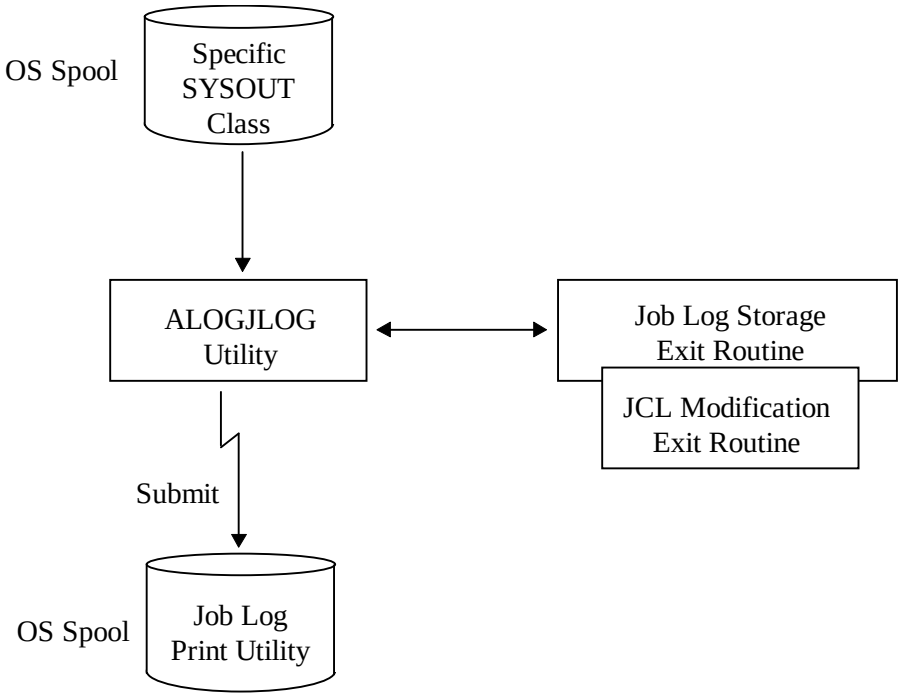


Fig. 7.4: JCL Modification Exit Routine for ALOGJLOG Utility

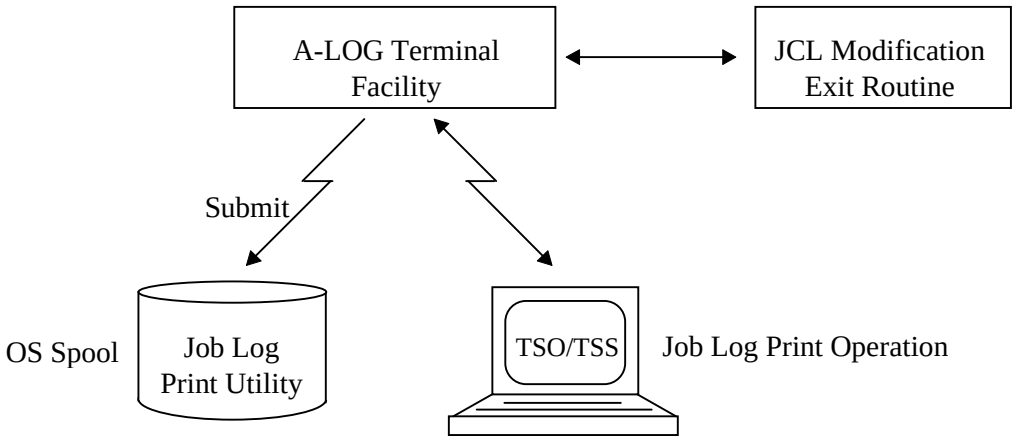


Fig. 7.5: JCL Modification Exit Routine for A-LOG Terminal Facility

Timing for Passing of Control

1) For ALOGJLOG utility

After catalog processing of a job completes, the JCL modification exit routine is called. It is called only when "J" is specified as a Print Code of the exit routine. The exit routine is loaded and called upon by ALOGJLOG utility in the timing as shown below:

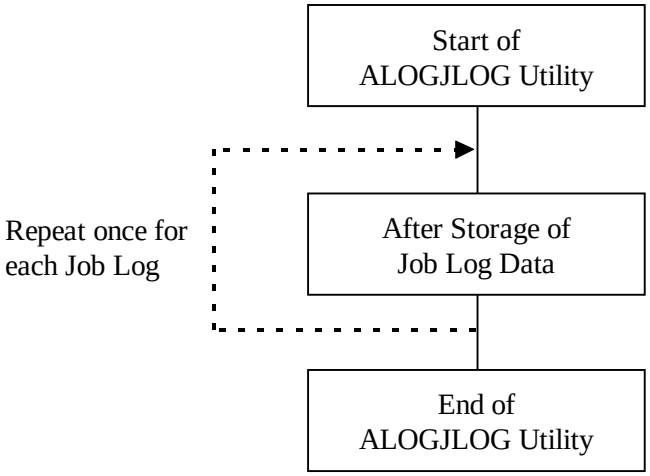


Fig. 7.6: Timing for Calling JCL Modification Exit Routine

2) For A-LOG Terminal Facility

After a print instruction is issued from an A-LOG terminal user, the JCL modification exit routine is called.

JCL Modification Exit Routine Specification

1) For ALOGJLOG Utility

Add the following DDJUTL DD statement into the procedure of ALOGJLOG utility:

```
//DDJUTL   DD   DSN=ALOG.V0380.PARMLIB(DDJUTL),DISP=SHR
```

Content of member DDJUTL in ALOG.V0380.PARMLIB:

```
EXT= module-name-of-JCL-modification-exit-routine
```

Note: The load module name of the exit routine must not be greater than 8 alphanumeric characters. And this load module must exist in library specified in STEPLIB statement of the execution JCLs of ALOGJLOG utility.

Add the following DDWTR DD statement into the procedure of ALOGJLOG utility:

```
//DDWTR   DD   DSN=dataset-name,DISP=SHR
```

Note: Specify the dataset name of the library which catalogs JCL of WTRJUTL utility for execution, in DDWTR DD statement.

2) For A-LOG Terminal Facility

Add the following line to the command procedure of A-LOG Terminal Facility:

```
ALLOC F(DDJUTL) DA('ALOG.V0380.PARMLIB(DDJUTL)') SHR
```

JCL Modification Exit Routine Interface

The calling convention between the exit routine and A-LOG system are based on standard OS calling convention. The following table highlights the usage of the registers:

		Input Interface	**Output Interface**
Registers	GR1	Parameter List Address	Parameter List Address
	GR0, GR2~12	Undefined	Undefined
	GR13	Register Save Area Address	Register Save Area Address
	GR14	Return Address	Return Address
	GR15	JCL Modification Exit Routine Entry Address	Set as '0'
	Parameter List	Refer to JCL Modification Exit Routine Parameters	—
	Return Code	—	Set in the JCL Modification Exit Routine Parameter

Table 7.7: Usage of Registers by JCL Modification Exit Routine

JCL Modification Exit Routine Parameters

The contents of the parameters passed to the exit routine are explained as follows:

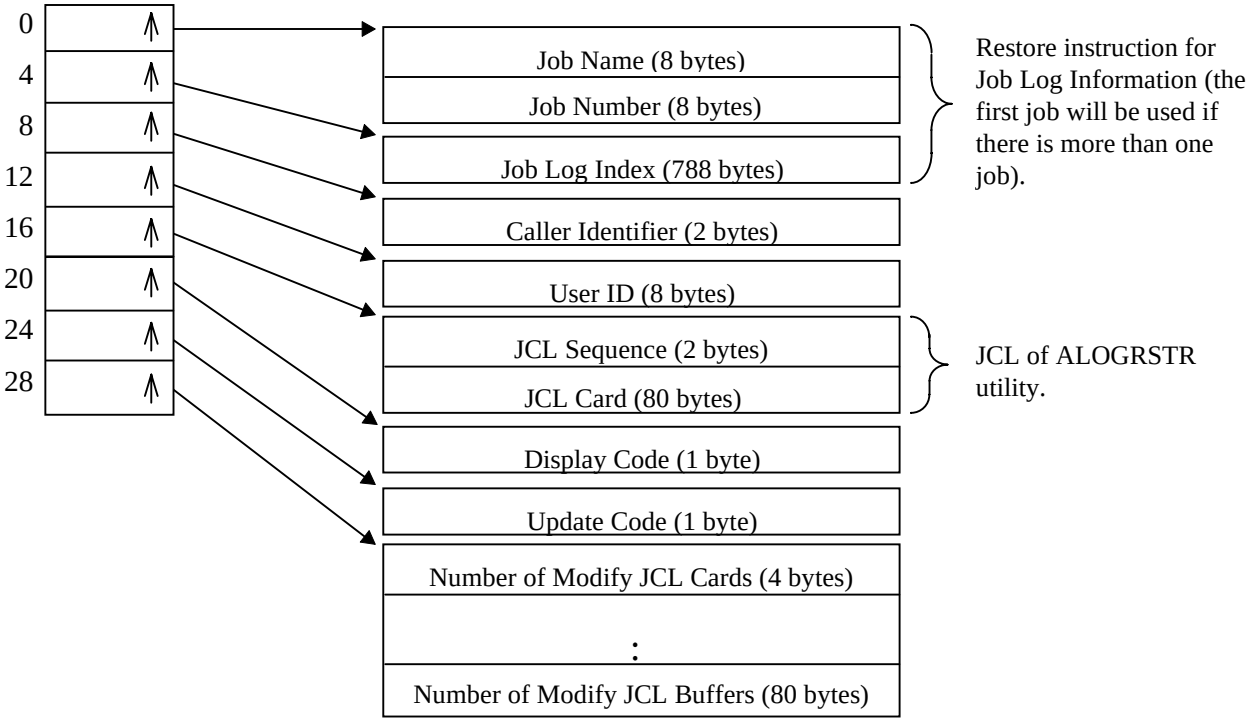


Fig. 7.7: JCL Modification Exit Routine Parameter List

Item Name	Length	Type	Description
Job Name	8	Alphanumeric	Job name of the specified job to be printed is set. See Note (1).
Job Number	8	Alphanumeric	Job number of the specified job to be printed is set. See Note (1).
Job Log Index	788	Alphanumeric Binary	Content of job log index of the specified job to be printed is set. Refer to **7.1.2 Job Log Index Release File** for further details.
Caller ID	2	Alphabetic	The component ID for starting job log print utility is set: • UL → Print command from terminal • LG → Job log storage utility

Table 7.8.a: List of JCL Modification Exit Routine Parameters

Item Name	Length	Type	Description
User ID	8	Alphabetic	When the Caller ID is 'UL', this area contains the logon ID of TSO/TSS user. Otherwise it contains binary zeroes.
JCL Sequence	2	Alphabetic	This is the sequence number of the JCL read from the member of a parameter library that contains sample JCLs.
JCL Card	80	Alphabetic	This is the card image of the JCL read from the member of a parameter library.
Display Code (See Note (2))	1	Alphabetic	Set the call timing of JCL modification exit routine: • O → Immediately before opening parameter library • R → Immediately after reading 1 JCL card from parameter library • C → Immediately after closing parameter library • P → Immediately before purging exit routine after completing the processing of 1 job
Modification Code (See Note (2))	1	Alphabetic	Specify processing for modification of JCL: • Δ → Use the JCL from JCL card instead of JCL from modification JCL buffer • A → Add JCL set in modification JCL buffer following the JCL card. However, if 'A' is specified in Modification Code when Display Code is 'O'. Content of modification JCL buffer is used before referring to JCL in parameter library • D → Do not refer to the JCL card and modification JCL buffer • R → Do not use JCL card, use content specified in modification JCL buffer • E → If JCLs exist in JCL buffer, then add the following JCL card and assume the JCL in parameter library as EOD If Update Code is other than 'D' or 'R', JCL card can be changed directly.
Number of Modification JCL Cards (See Note (3))	4	Binary	Specify the number of modification JCL cards specified in modification JCL buffer. If Display Code is 'O', then '0' is set.
Modification JCL Buffer (See Note (3))	80	Alphabetic	Set the JCL required for addition and modification of JCL. This item can only be used when Update Code is 'A', 'R' or 'E'.

Δ → Blank

Table 7.8.b: List of JCL Modification Exit Routine Parameters

Notes: (1) When printing two or more jobs at a time, it sets the first job name specified by print operation.

(2) The relation between a Display Code and Modification Code is shown as follows:

Display Code	Modification Code				
	Δ	**A**	**D**	**R**	**E**
O	❍	❍	–	–	❍
R	❍	❍	❍	❍	❍
C	❍	❍	–	❍	❍
P	❍	❍	–	–	–

❍ Valid – Invalid

Table 7.9: Relationship between Display Code and Modification Code

(3) When you change JCL, create the modification JCL buffer as follows and set the address in JCL modification exit routine parameter:

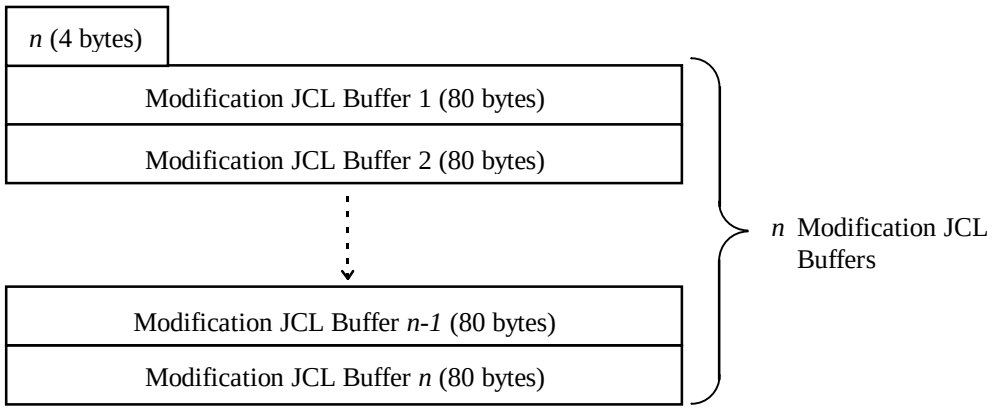


Fig. 7.8: Layout of Modification JCL Buffers

7.4 Job Log Print Exit Routine

Function

This exit routine receives control from the WTRJUTL utility. Using this exit routine, users can perform print control such as inserting user-defined separator in the printing of job logs.

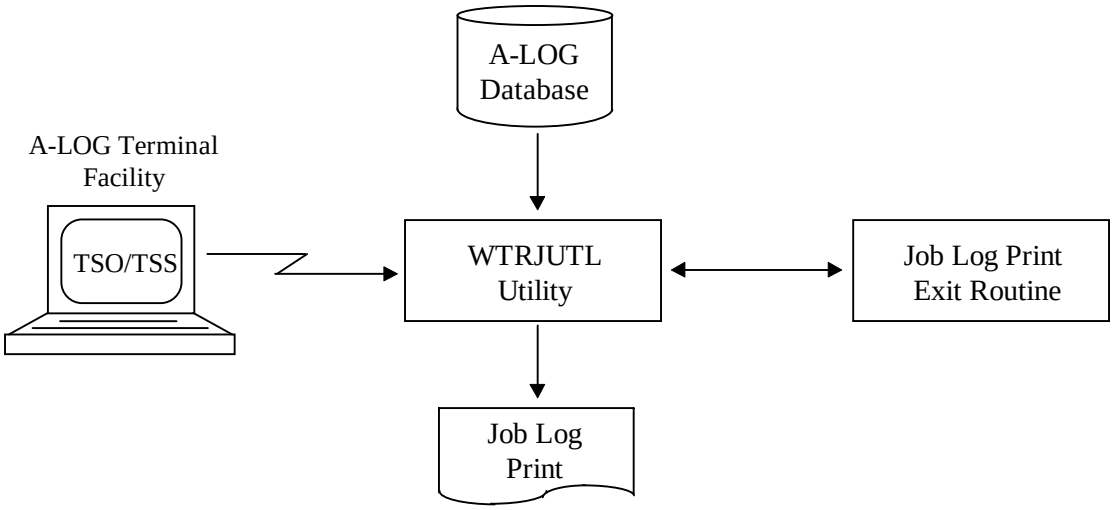


Fig. 7.9: Job Log Print Exit Routine

Timing for Passing of Control

This exit routine is loaded when WTRJUTL utility is invoked and called upon in the timing as shown below:

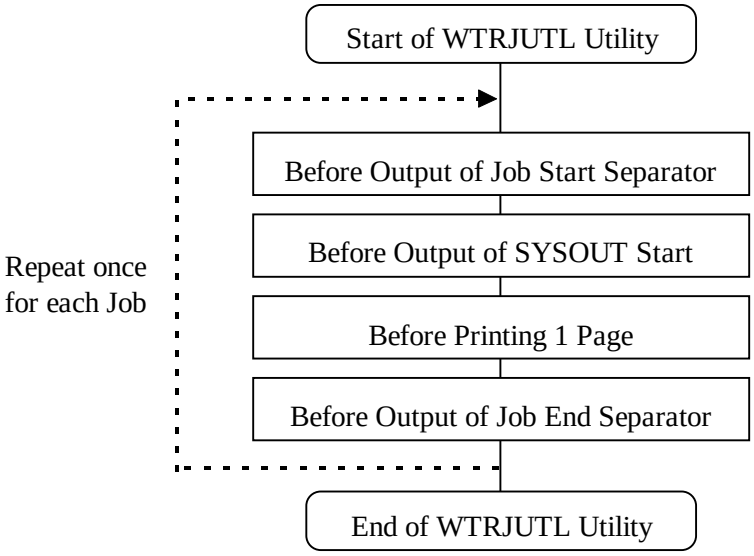


Fig. 7.10: Timing for Calling Job Log Print Exit Routine

Job Log Print Exit Routine Specification

Specify the load module name of the exit routine in the EXEC parameter of the execution JCL of WTRJUTL utility as follows:

```
PARM=S@0000load-module-name-of-exit-routine
```

Note: The load module name of the exit routine must not be greater than 8 alphanumeric characters. This load module must exist in the library specified in STEPLIB statement of the execution JCLs of WTRJUTL utility.

Job Log Print Exit Routine Interface

The calling convention between the exit routine and WTRJUTL utility are based on standard OS calling convention. The following table highlights the usage of the registers:

		Input Interface	**Output Interface**
Registers	GR1	Parameter List Address	Parameter List Address
	GR0, GR2~12	Undefined	Undefined
	GR13	Register Save Area Address	Register Save Area Address
	GR14	Return Address	Return Address
	GR15	Print Exit Routine Entry Address	Set as '0'
	Parameter List	Refer to Job Log Print Exit Routine Parameters	—
	Return Code	—	Set in the Job Log Print Exit Routine Parameter

Table 7.10: Usage of Registers by Job Log Print Exit Routine

Job Log Print Exit Routine Parameters

The contents of parameters passed to the exit routine are explained as follows:

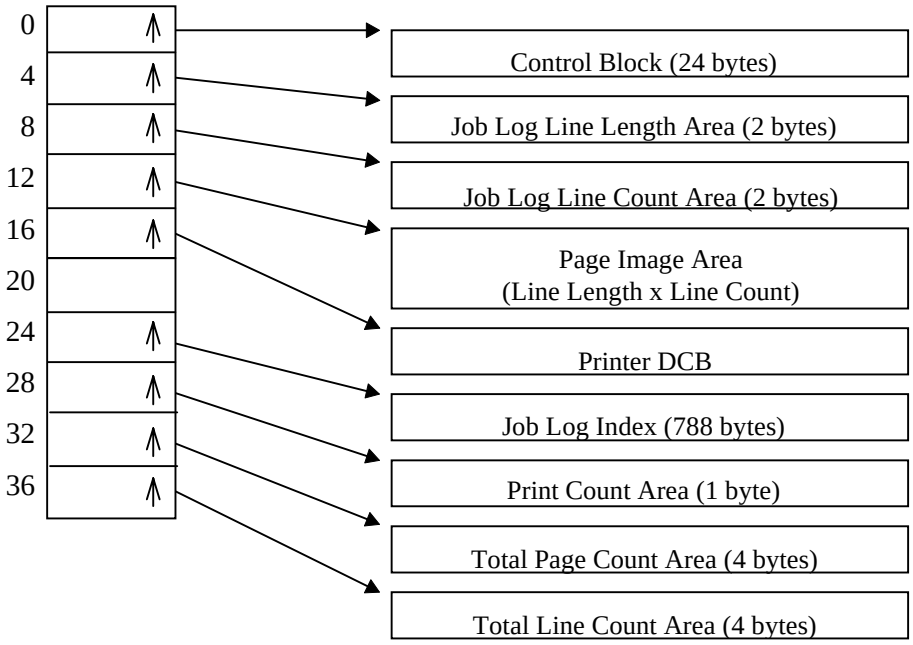


Fig. 7.11: Job Log Print Exit Parameter List

1) Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
0	0	Display Code	1	Alphabetic	Set the call timing of job log print exit routine: • O → Before output of job start separator • S → Before output of SYSOUT start separator • W → Before printing 1 page • C → Before output of job end separator
1	1	—	1	—	Unused.
2	2	Return Code	1	Alphabetic	Specify the output options: • When Display Code is 'O', 'S' or 'C': Δ → Output separator N → Do not output separator • When Display Code is 'W': Δ → Output the particular page N → Do not output the particular page

Δ → Blank

Table 7.11.a: Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
3	3	Print Mode	1	Alphabetic	'R' is set.
4	4	—	8	—	Unused.
6	6	Output Device ID	1	Alphabetic	The first character of EXEC parameter of WTRJUTL utility execution JCL is set: • L → Output media is a printer • S → Output media is OS Spool • 3 → Output media is a PS file
7	7	LBP-ID	1	Alphabetic	The second character of EXEC parameter of WTRJUTL utility execution JCL is set. This item is only valid if the output media is an LBP printer.
8	8	—	16	—	Unused.

Table 7.11.b: Control Block Format

2) Job Log Line Length Area

Item Name	Length	Type	Description
Job Log Line Length	2	Binary	Line length of job log is set from the beginning of printing to the end of printing.

3) Job Log Line Count Area

Item Name	Length	Type	Description
Job Log Line Count	2	Binary	When Display Code is 'W', line count included in a page is set.

4) Page Image Area

Item Name	Length	Type	Description
Page Image	Line Length x Line Count	Alphanumeric	When Display Code is 'W', content of page is set (consecutive area).

5) Printer DCB

This is the output DCB. The exit routine outputs a user defined separator using this DCB.

6) Job Log Index

The format of Job Log index is the same as that of the Job Log index release file. Refer **to "7.1.2 Job Log Index Release File**" for further details.

7) Print Count Area

Item Name	Length	Type	Description
Print Count	1	Binary	Total print count of job log is set from beginning of printing to the end of printing.

8) Total Page Count Area

Item Name	Length	Type	Description
Total Page Count	4	Binary	Total page count by job log is set. If a special separator is to be output by job log print exit, add the extra number of pages.

9) Total Line Count Area

Item Name	Length	Type	Description
Total Line Count	4	Binary	Total line count by job log is set. If a special separator is to be output by job log print exit, add the extra number of lines.

When control is passed to the exit routine, some parameters are valid and others are invalid depending on Display Code. The validity of a parameter is shown below:

Parameter		Display Code			
		O	S	W	C
Control Block	Return Code	❍	❍	❍	❍
	Print Mode	❍	❍	❍	❍
Job Log Line Length		–	–	❍	–
Jog Log Line Count		–	–	❍	–
Page Image		–	–	❍	–
Printer DCB		❍	❍	❍	❍
Job Log Index		❍	❍	❍	❍
Printer Count		❍	❍	❍	❍
Total Page Count		❍	❍	❍	❍
Total Line Count		❍	❍	❍	❍

❍ Valid – Invalid

Table 7.12: Validity of Job Log Print Exit Parameters

7.5 SYSLOG Print Exit Routine

Function

This exit routine receives control from the WTRSUTL utility. Using this exit routine, users can perform print control such as inserting user-defined separator in the printing of job log.

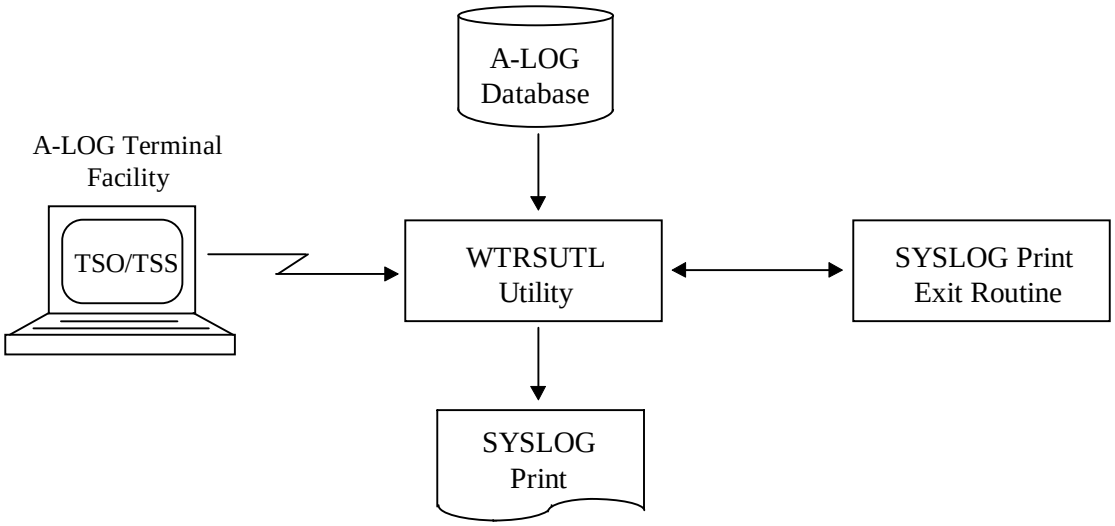


Fig. 7.12: SYSLOG Print Exit Routine

Timing for Passing of Control

The exit routine is loaded when WTRSUTL utility is invoked and called upon in the timing as below:

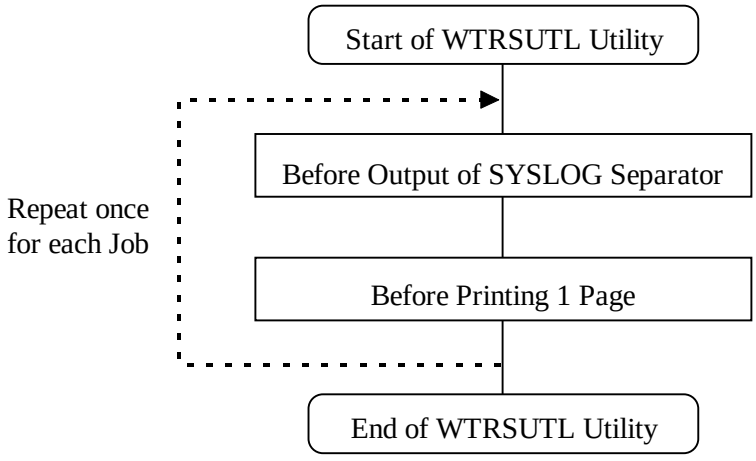


Fig. 7.13: Timing for Calling SYSLOG Print Exit Routine

SYSLOG Print Exit Routine Specification

Specify the load module name of the exit routine in EXEC parameter of the execution JCL of WTRSUTL utility as follows:

```
PARM=S@0000load-module-name-of-exit-routine
```

Note: The load module name of the exit routine must not be greater than 8 alphanumeric characters. This load module must exist in the library specified in STEPLIB statement of the execution JCLs of WTRSUTL utility.

SYSLOG Print Exit Routine Interface

The calling convention between the exit routine and A-LOG system are based on standard OS calling convention. The following table highlights the usage of the registers:

		Input Interface	**Output Interface**
Registers	GR1	Parameter List Address	Parameter List Address
	GR0, GR2~12	Undefined	Undefined
	GR13	Register Save Area Address	Register Save Area Address
	GR14	Return Address	Return Address
	GR15	SYSLOG Exit Routine Entry Address	Set as '0'
	Parameter List	Refer to SYSLOG Print Exit Routine Parameters	—
	Return Code	—	Set in the SYSLOG Print Exit Routine Parameter

Table 7.13: Usage of Registers for SYSLOG Print Exit Routine

SYSLOG Print Exit Routine Parameters

The contents of parameters passed to the exit routine are explained as follows:

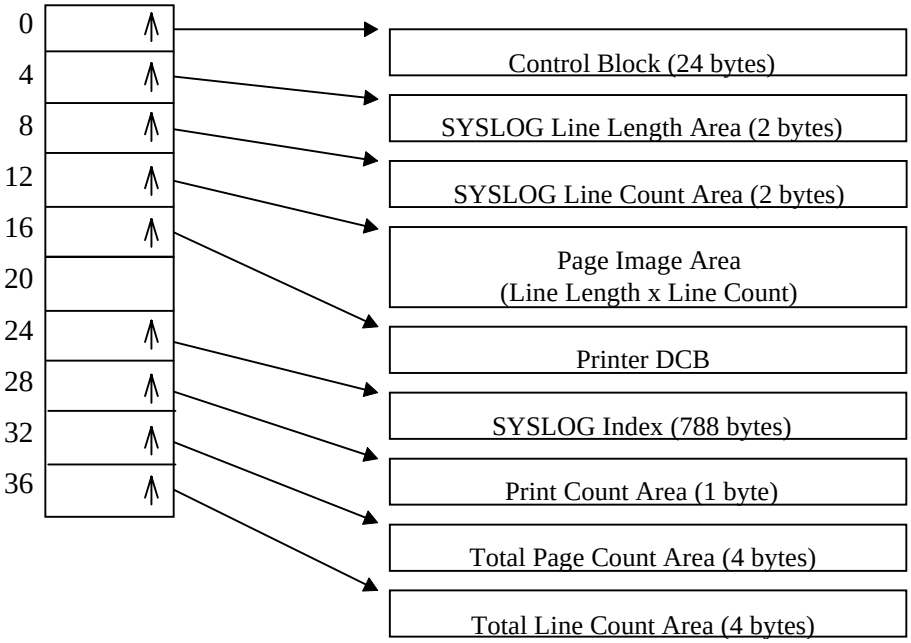


Fig. 7.14: SYSLOG Print Exit Parameter List

1) Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
0	0	Display Code	1	Alphabetic	Set the call timing of SYSLOG print exit routine: • S → Before the output of SYSLOG start separator • W → Before printing each SYSLOG record
1	1	—	1	—	Unused.
2	2	Return Code	1	Alphabetic	Specify the output options: • When Display Code is 'S': Δ → Output separator N → Do not output separator • When Display Code is 'W': Δ → Output the particular record N → Do not output the particular record

Δ → Blank

Table 7.14.a: Control Block Format

Position (Decimal)	Position (Hex)	Item	Length	Type	Description
3	3	Print Mode	1	Alphabetic	'R' is set.
4	4	—	8	—	Unused
6	6	Output Device ID	1	Alphabetic	The first character of EXEC parameter of WTRSUTL utility execution JCL is set: • L → Output media is a printer • S → Output media is OS Spool • 3 → Output media is a PS file
7	7	—	17	—	Unused.

Table 7.14.b: Control Block Format

2) SYSLOG Line Length Area

Item Name	Length	Type	Description
SYSLOG Line Length	2	Binary	Line length of SYSLOG is set from the beginning of printing to the end of printing.

3) SYSLOG Line Count Area

Item Name	Length	Type	Description
SYSLOG Line Count	2	Binary	When Display Code is 'W', line count included in a page is set always as X'0001' (because it is called once for each record).

4) Record Image Area

Item Name	Length	Type	Description
Page Image	Line Length x Line Count	Alphanumeric	When Display Code is 'W', content of record is set.

5) Printer DCB

This is the output DCB. The exit routine outputs a user defined separator using this DCB.

6) Print Count Area

Item Name	Length	Type	Description
Print Count	1	Binary	Total print count of SYSLOG is set from the beginning of printing to the end of printing.

7) Total Line Count Area

Item Name	Length	Type	Description
Total Line Count	4	Binary	Total line count by SYSLOG is set. If a special separator is to be output by SYSLOG print exit, add the extra number of lines.

When control is passed to the exit routine, some parameters are valid and others are invalid depending on Display Code. The validity of a parameter is shown below:

Parameter		Display Code	
		S	W
Control Block	Return Code	❍	❍
	Print Mode	❍	❍
SYSLOG Line Length		–	❍
SYSLOG Line Count		–	❍
Page Image		–	❍
Printer DCB		❍	❍
Printer Count		❍	❍
Total Line Count		❍	❍

❍ Valid – Invalid

Table 7.15: Validity of Job Log Print Exit Parameters

Chapter 8

A-LOG Security Management Facility

8.1 General Overview

8.1.1 Introduction

Generally, the resources in A-LOG system that are accessed via A-LOG Terminal Facility can be protected by using A-LOG Security Management Facility, which provides protection based on the TSO/TSS user-ID of the user. A user can have access to those resources that have been authorized to him/her by the A-LOG System Administrator. No password is required and authorization is based on his/her TSO/TSS user-ID.

8.1.2 A-LOG Security Protection Functions

In A-LOG Security Management Facility, each screen or screen function is classified as a primary resource. The objects that can be displayed in this screen function are the secondary resources. For example, in the [2.1.0 Joblog Index Maintenance] screen, the primary resource is the[2.1.0 Joblog Index Maintenance] screen itself and the secondary resources are the job logs that can be displayed on this screen. A list of primary and secondary resources is given in "**8.1.3 Primary and Secondary Resources of A-LOG Terminal Facility**".

A-LOG Security Management Facility allows the A-LOG System Administrator to define authorized users (represented by User-IDs) or groups (represented by Group-IDs) that are allowed to access the A-LOG Terminal Facility. Group-IDs may be defined to represent different categories of job functions e.g. Administrator group and Computer Operator group. One or more users can then be linked to a Group-ID. The necessary primary and second resources are then assigned to each Group-ID or individual User-ID. Users linked to a group share the same security profile and, therefore, have access to all the resources defined for the group.

For each primary and secondary resources accessible by a group or an individual user, the type of access, i.e. authorization level, can be defined. The authorization level supported are:

- Read : which allows retrieval access only;
- Update : which allows retrieval, modification and addition accesses; and
- Alter : which allows retrieval, modification, addition, delete and copy accesses.

A user switching from one screen to another in A-LOG Terminal Facility may require authorization to different primary and secondary resources. The appropriate primary resources, secondary resources and authorization levels required are explained in detail in **Appendix 1: A-LOG Security Resource Diagrams**.

8.1.3 Primary and Secondary Resources of A-LOG Terminal Facility

The primary and secondary resource names for setting security to the functions of each screen in A-LOG Terminal Facility are listed as follows:

Screen No.	Screen Name	Screen Function	Primary Resource	Secondary Resource
—	Hello Screen	A-LOG Terminal Facility Logon Screen	ABAS.ALOG.DCDUSER	—
1.1.0	Joblog / SYSLOG Master Maintenance	✶ Entry to Job Log or SYSLOG Master Maintenance screen	ABAS.ALOG.SCR.JAM	—
		✸ Registration / deletion / inquiry / update of Job Log or SYSLOG Master	ABAS.ALOG.MST.JAM	Job Log ID or SYSLOG ID
1.1.1	Joblog / SYSLOG Master Directory	Inquiry / deletion of Job Log or SYSLOG Master	ABAS.ALOG.MST.JAM	Job Log ID or SYSLOG ID
1.1.2	Joblog / SYSLOG Master	Inquiry / update of Job Log or SYSLOG Master	ABAS.ALOG.MST.JAM	Job Log ID or SYSLOG ID
2.1.0	Job Log Index Maintenance	✶ Start of Job Log Management Function	ABAS.ALOG.SCR.JLG	—
		✸ Inquiry / deletion/update of Job Log Index	ABAS.ALOG.IDX.JBI	Job Name
2.1.1.1 2.1.1.2	Job Log Directory	✶ Inquiry / deletion of Job Log Index	ABAS.ALOG.IDX.JBI	Job Name
		✸ Display of job log data	ABAS.ALOG.IDX.JBI ABAS.ALOG.IDX.BKS	Job Name
		✹ Printing of job log data	ABAS.ALOG.IDX.JBI	Job Name
			ABAS.ALOG.OPE. COMMAND.LVL0	Writer Name
		❋ Restore instruction for job log	ABAS.ALOG.IDX.JBI	Job Name
			ABAS.ALOG.OPE. COMMAND.LVL0	Restore Job Member Name
		✠ Restore reservation / cancellation for job log	ABAS.ALOG.IDX.JBI	Job Name

Table 8.1.a: Primary and Secondary Resources of A-LOG Terminal Facility

Screen No.	Screen Name	Screen Function	Primary Resource	Secondary Resource
2.1.2.1 2.1.2.2	Job Log Index -No.1-	✶ Inquiry / update of Job Log Index	ABAS.ALOG.IDX.JBI	Job Name
		✸ Restore instruction for job log	ABAS.ALOG.IDX.JBI	Job Name
			ABAS.ALOG.OPE. COMMAND.LVL0	Restore Job Member Name
		✹ Restore reservation / cancellation for job log	ABAS.ALOG.IDX.JBI	Job Name
2.1.3	Job Log Index -No.2-	✶ Inquiry / update of SYSOUT information of Job Log Index and display data by SYSOUT dataset unit	ABAS.ALOG.IDX.JBI	Job Name
		✸ Print by SYSOUT dataset unit	ABAS.ALOG.IDX.JBI	Job Name
			ABAS.ALOG.OPE. COMMAND.LVL0	Writer Name
2.1.4	Job Log Inquiry	Display of job log data	ABAS.ALOG.IDX.JBI	Job Name
2.2.0	Dump MT Inquiry	✶ Start of Dump MT inquiry function	ABAS.ALOG.SCR.DMM	—
		✸ Inquiry/deletion of Dump MT	ABAS.ALOG.IDX.DMM	—
3.1.0	SYSLOG Data Select	✶ Start of SYSLOG data management function	ABAS.ALOG.SCR.SDT	—
		✸ Inquiry of SYSLOG data	ABAS.ALOG.IDX.JBI	SYSLOG ID
		✹ Printing of SYSLOG data	ABAS.ALOG.IDX.JBI	SYSLOG ID
			ABAS.ALOG.OPE. COMMAND.LVL0	Writer Name
3.1.1	SYSLOG Data Inquiry	✶ Display SYSLOG data	ABAS.ALOG.IDX.JBI	SYSLOG ID
		✸ Print SYSLOG data	ABAS.ALOG.IDX.JBI	SYSLOG ID
			ABAS.ALOG.OPE. COMMAND.LVL0	Writer Name
3.2.0	SYSLOG Index Maintenance	✶ Start of SYSLOG index management function	ABAS.ALOG.SCR.SLG	—
		✸ Inquiry / deletion of SYSLOG index	ABAS.ALOG.IDX.JBI	SYSLOG ID

Table 8.1.b: Primary and Secondary Resources of A-LOG Terminal Facility

Screen No.	Screen Name	Screen Function	Primary Resource	Secondary Resource
3.2.1	SYSLOG Directory	✶ Inquiry / deletion of SYSLOG Index	ABAS.ALOG.IDX.JBI	SYSLOG ID
		✷ Restore instruction for SYSLOG	ABAS.ALOG.IDX.JBI	SYSLOG ID
			ABAS.ALOG.OPE. COMMAND.LVL0	Restore Job Member Name
		✹ Restore reservation / cancellation for SYSLOG	ABAS.ALOG.IDX.JBI	SYSLOG ID
3.2.2.1 3.2.2.2	SYSLOG Index -No.1-	✶ Inquiry / update of SYSLOG Index	ABAS.ALOG.IDX.JBI	SYSLOG ID
		✷ Restore instruction for SYSLOG	ABAS.ALOG.IDX.JBI	SYSLOG ID
			ABAS.ALOG.OPE. COMMAND.LVL0	Restore Job Member Name
		✹ Restore reservation/ cancellation for SYSLOG	ABAS.ALOG.IDX.JBI	SYSLOG ID
9.1.0	Terminal / PF Keys Setting	✶ Start of PF-Keys management function	ABAS.ALOG.SCR.PFK	—
		✷ Inquiry/update of terminal PF keys	ABAS.ALOG.SYS. PFKEYSET	—

Table 8.1.c: Primary and Secondary Resources of A-LOG Terminal Facility

Refer to "**Appendix 1: A-LOG Security Resource Diagrams**" for explanation of the primary resources, secondary resources and authorization levels. Refer to "**A-BAS Security Management User's Manual**" for explanation on the registration method for A-LOG Security Management Table.

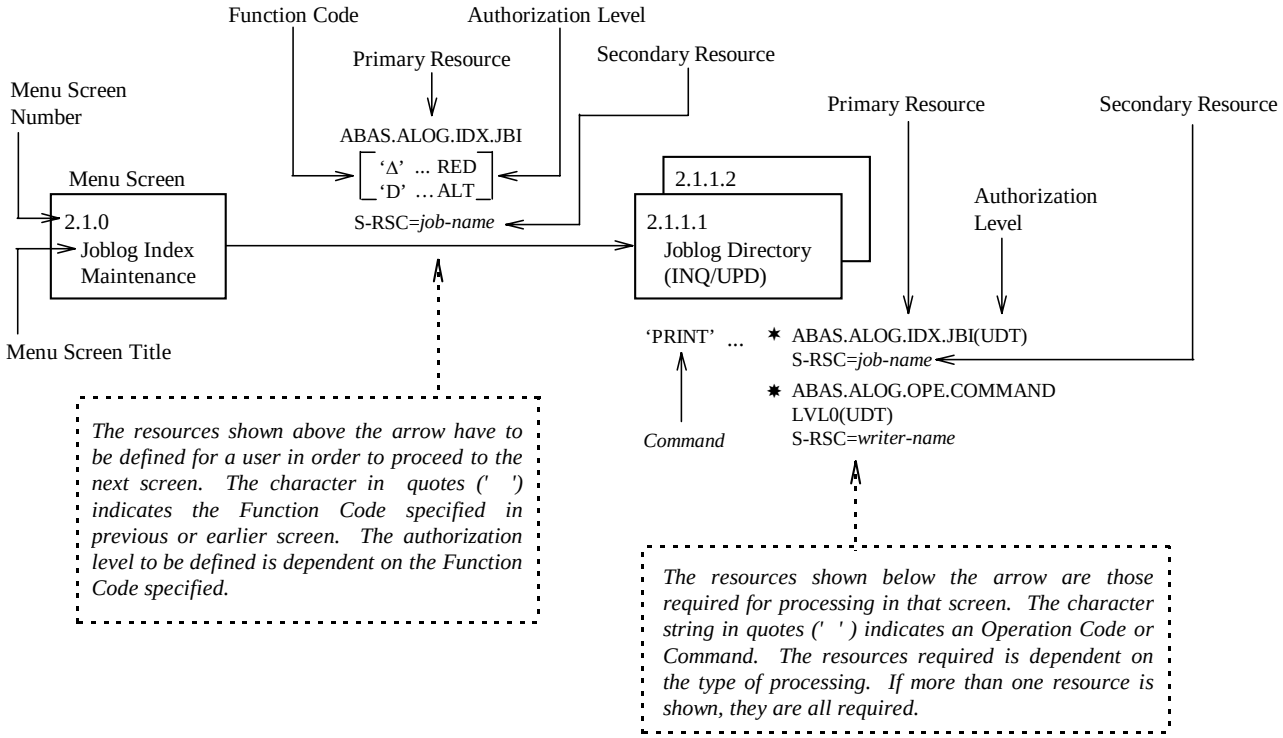
Appendix 1

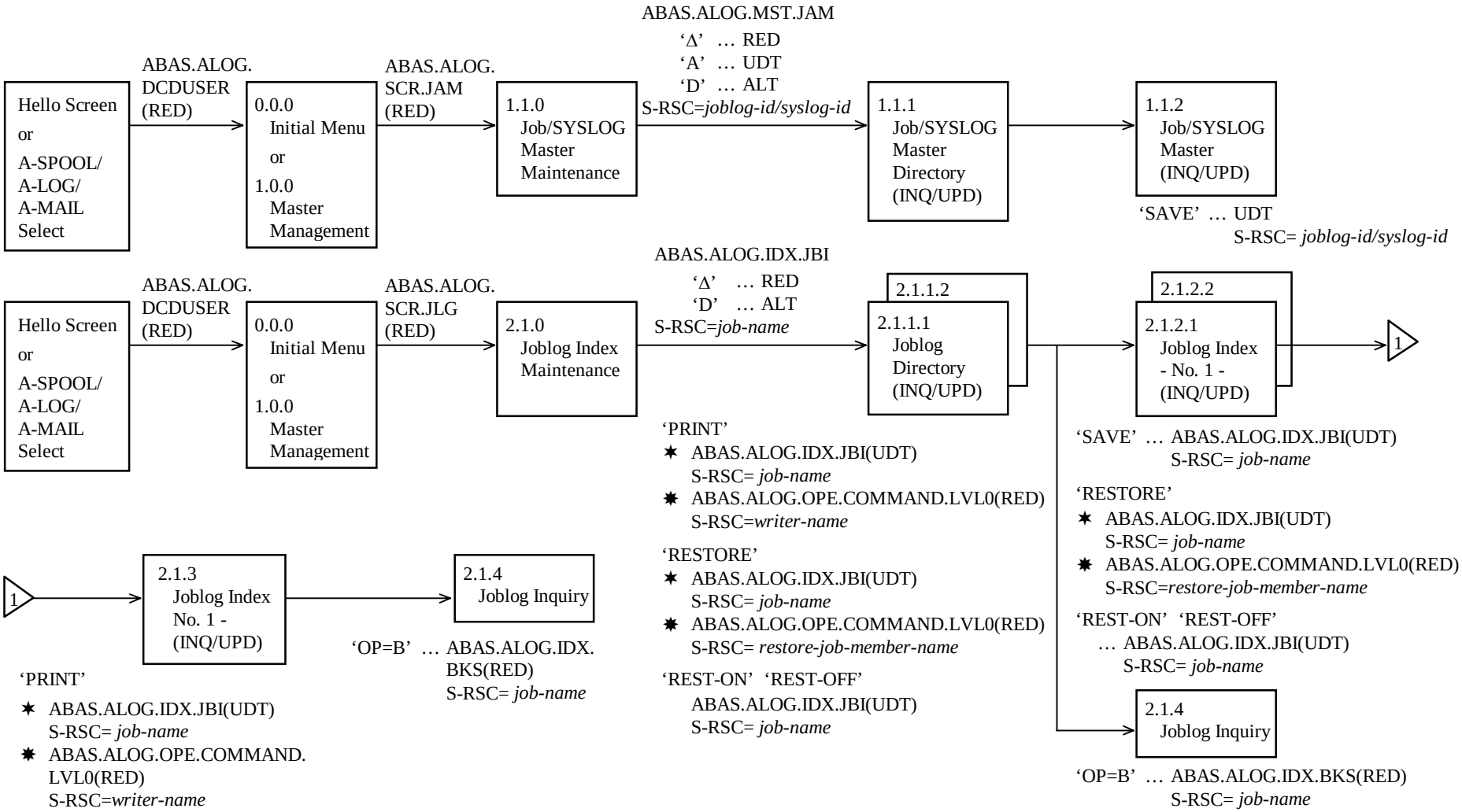
A-LOG Security Resource Diagrams

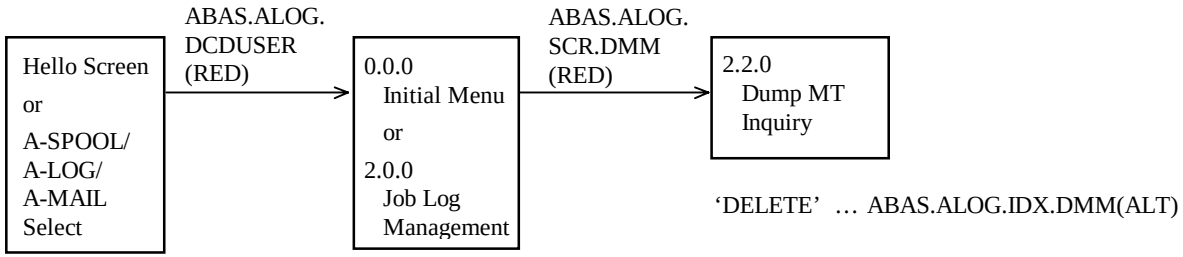
A1.1 Security Resource Diagrams

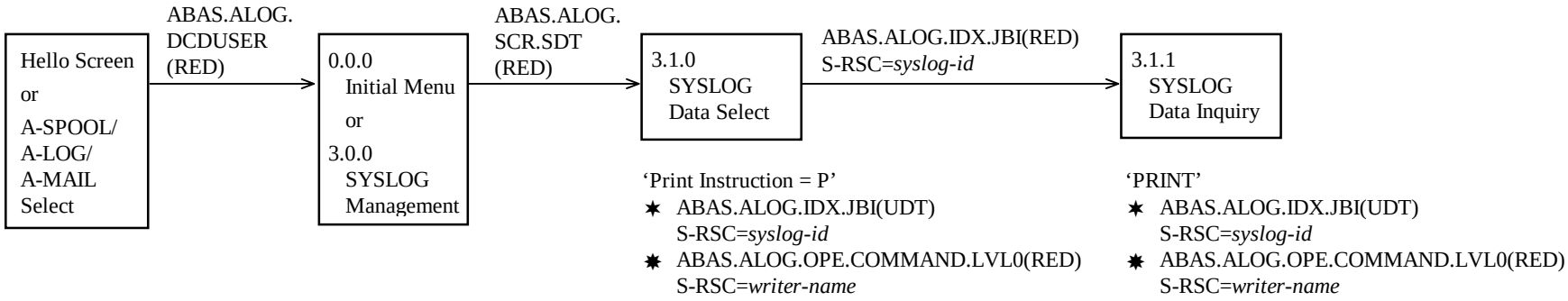
This appendix shows the primary resources, secondary resources and authorization levels required for the various functions of A-LOG Terminal Facility. These functions are represented by various screens, commands, OP codes and function codes in A-LOG Terminal Facility. For a user of A-LOG Terminal Facility to perform the required functions, the appropriate primary resources, secondary resources and authorization levels must be defined for the TSO/TSS User-ID of this user.

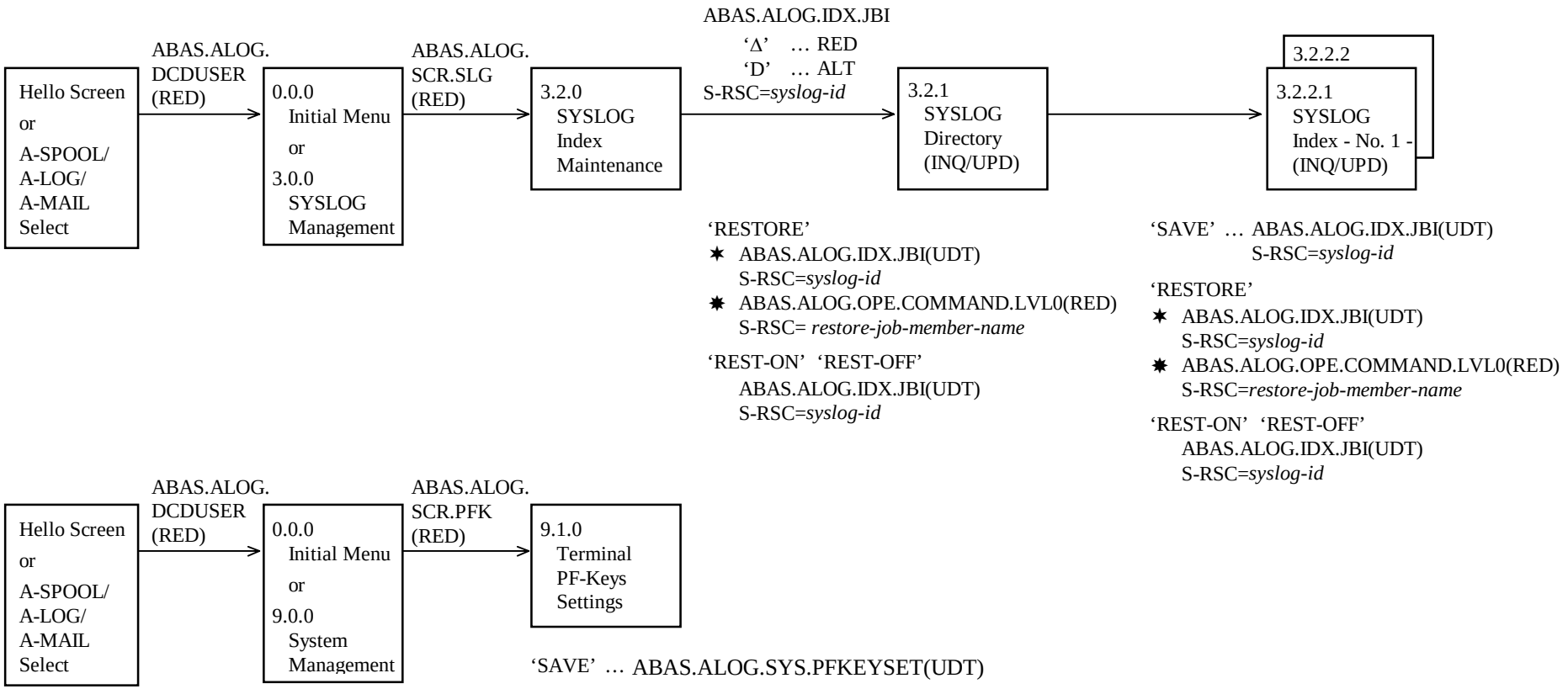
The following diagram is intended to provide an example and explanation about A-LOG security resource diagram:




Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
1.0.0
Master
Management
ABAS.ALOG.
SCR.JAM
(RED)
1.1.0
Job/SYSLOG
Master
Maintenance
ABAS.ALOG.MST.JAM
'Δ' … RED
'A' … UDT
'D' … ALT
S-RSC=joblog-id/syslog-id
1.1.1
Job/SYSLOG
Master
Directory
(INQ/UPD)
1.1.2
Job/SYSLOG
Master
(INQ/UPD)
'SAVE' … UDT
S-RSC= joblog-id/syslog-id
Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
1.0.0
Master
Management
ABAS.ALOG.
SCR.JLG
(RED)
2.1.0
Joblog Index
Maintenance
ABAS.ALOG.IDX.JBI
'Δ' … RED
'D' … ALT
S-RSC=job-name
2.1.1.2
2.1.1.1
Joblog
Directory
(INQ/UPD)
2.1.2.2
2.1.2.1
Joblog Index
- No. 1 -
(INQ/UPD)
1
'PRINT'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
✸ ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC=writer-name
'RESTORE'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
✸ ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC= restore-job-member-name
'REST-ON' 'REST-OFF'
ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
'SAVE' … ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
'RESTORE'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
✸ ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC=restore-job-member-name
'REST-ON' 'REST-OFF'
… ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
2.1.4
Joblog Inquiry
'OP=B' … ABAS.ALOG.IDX.BKS(RED)
S-RSC= job-name
1
2.1.3
Joblog Index
No. 1 -
(INQ/UPD)
2.1.4
Joblog Inquiry
'OP=B' … ABAS.ALOG.IDX.
BKS(RED)
S-RSC= job-name
'PRINT'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC= job-name
✸ ABAS.ALOG.OPE.COMMAND.
LVL0(RED)
S-RSC=writer-name


Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
2.0.0
Job Log
Management
ABAS.ALOG.
SCR.DMM
(RED)
2.2.0
Dump MT
Inquiry
'DELETE' … ABAS.ALOG.IDX.DMM(ALT)


Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
3.0.0
SYSLOG
Management
ABAS.ALOG.
SCR.SDT
(RED)
3.1.0
SYSLOG
Data Select
ABAS.ALOG.IDX.JBI(RED)
S-RSC=syslog-id
3.1.1
SYSLOG
Data Inquiry
'Print Instruction = P'
ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC=writer-name
'PRINT'
ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC=writer-name


Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
3.0.0
SYSLOG
Management
ABAS.ALOG.
SCR.SLG
(RED)
3.2.0
SYSLOG
Index
Maintenance
ABAS.ALOG.IDX.JBI
'Δ' … RED
'D' … ALT
S-RSC=syslog-id
3.2.1
SYSLOG
Directory
(INQ/UPD)
3.2.2.2
3.2.2.1
SYSLOG
Index - No. 1 -
(INQ/UPD)
'RESTORE'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
✸ ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC= restore-job-member-name
'REST-ON' 'REST-OFF'
ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
'SAVE' … ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
'RESTORE'
✶ ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
✸ ABAS.ALOG.OPE.COMMAND.LVL0(RED)
S-RSC=restore-job-member-name
'REST-ON' 'REST-OFF'
ABAS.ALOG.IDX.JBI(UDT)
S-RSC=syslog-id
Hello Screen
or
A-SPOOL/
A-LOG/
A-MAIL
Select
ABAS.ALOG.
DCDUSER
(RED)
0.0.0
Initial Menu
or
9.0.0
System
Management
ABAS.ALOG.
SCR.PFK
(RED)
9.1.0
Terminal
PF-Keys
Settings
'SAVE' … ABAS.ALOG.SYS.PFKEYSET(UDT)

Appendix 2

Contents of A-LOG Cataloged Procedures

Reference to A-LOG Cataloged Procedures

ALOG

```
//*******************************************************
//*                                                     *
//*    PROC NAME = ALOG                                 *
//*                                                     *
//*       FUNCTION = USMS SPOOL MANAGER START           *
//*                                                     *
//*       PARM     = NOTHING                            *
//*                                                     *
//*       SYSIN    = 1. RUNSPM3 ... VSAM DB VERIFY CARD *
//*                  2. RUNSPM2 ... SPL CONTROL CARD    *
//*                                                     *
//*    VERSION NO.     V03.80                           *
//*                                                     *
//*******************************************************
//ALOG     PROC TYPE=WARM,SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=IDCAMS,REGION=512K                                    IBM
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNSPM3),DISP=SHR
//ALOG     EXEC PGM=SPL,REGION=2048K,PARM=&TYPE,TIME=1440
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR          A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR          A-LOG Data Pool (Data Part)
//ABSIX001 DD   DSN=ALOG.DBIDEX,DISP=SHR          A-LOG Management Database (Control Part)
//ABSID001 DD   DSN=ALOG.DBDATA,DISP=SHR          A-LOG Management Database (Data Part)
//ABSLOGX  DD   DSN=ALOG.HOTLOG,DISP=SHR          Hot Log File
//*CPIMCP00 DD   DSN=ALOG.V0380.PARMLIB(CPIMCP00),DISP=SHR
//USMSCARD DD   DSN=ALOG.V0380.PARMLIB(RUNSPM2),DISP=SHR     Monitor Start Parameter
//ASPPRM1  DD   DSN=ALOG.V0380.PARMLIB(ASPPRM1),DISP=SHR     System Definition. Parameter
//ASPCTBL  DD   DSN=ALOG.V0380.LOAD(ASPCTBLL),DISP=SHR
//DDWAN    DD   DSN=ALOG.V0380.PARMLIB,DISP=SHR
//SORTWK10 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK20 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK30 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//DDINF    DD   DUMMY
//USMSOUT  DD   SYSOUT=&SOUT                                      A-LOG Message
//SYSOUT   DD   DUMMY
//DBSNAP   DD   DUMMY
//DDSNAP   DD   DUMMY
//GATSNAP  DD   DUMMY
//DDCARD   DD   DUMMY,DCB=BLKSIZE=20
//SYSUDUMP DD   SYSOUT=&UOUT
//ABSIPRM1 DD   DSN=ALOG.V0380.PARMLIB(ABSIPRM1),DISP=SHR     Database Access Definition.
//ABSILST1 DD   SYSOUT=&SOUT
```

ALOGBLOG

```
//*******************************************************
//*                                                     *
//*   PROC NAME = ALOGBLOG                              *
//*                                                     *
//*   STEP0010 : MAKE JOBLOG FREE FILE                  *
//*              PARM='DATE=YYMMDD(YYMMDD)'             *
//*                    FROM DATE (YYMMDD)               *
//*                    TO   DATE (YYMMDD)               *
//*                                                     *
//*   STEP0020 : SORT JOBLOG FREE FILE                  *
//*              SYSIN=RUNLOG5                          *
//*                                                     *
//*   STEP0030 : LIST OUT JOBLOG FREE FILE              *
//*              PARM='DATE=YYMMDD(YYMMDD),             *
//*                  CODE=ALL,NOR,ERR,                  *
//*                  SYSOUT=YES,NO'                     *
//*                                                     *
//*   VERSION NO.   V03.80                              *
//*                                                     *
//*******************************************************
//ALOGBLOG PROC PDATE=,PCODE=ALL,POUT=YES,
//             SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=LOG900,PARM='PDATE=&PDATE'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDBLOG   DD   DSN=&&BLOG1,DISP=(,PASS),                    Job Log Index Release File
//              UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE)
//DDPRINT  DD   SYSOUT=&SOUT                                 Proof List
//USMSOUT  DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//STEP0020 EXEC PGM=SORT,COND=(0,NE,STEP0010)
//SORTIN   DD   DSN=&&BLOG1,DISP=(SHR,DELETE)
//SORTOUT  DD   DSN=&&BLOG2,DISP=(,PASS),
//              UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSOUT   DD   DUMMY
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNLOG5),DISP=SHR     Sort Parameter
//STEP0030 EXEC PGM=LOG910,
//              PARM='PDATE=&PDATE,CODE=&PCODE,SYSOUT=&POUT',
//              COND=(0,NE,STEP0020)
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDRLOG   DD   DSN=&&BLOG2,DISP=(OLD,DELETE)
//DDPRINT  DD   SYSOUT=&SOUT                                 Job Log Audit Trail List
//USMSOUT  DD   SYSOUT=&SOUT                                 A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGCHKA

```
//*************************************************************
//*                                                           *
//*       PROC NAME = ALOGCHKA                                *
//*                                                           *
//*       FUNCTION  = A-LOG MONITOR ACTIVE CHECK UTILITY      *
//*                                                           *
//*       VERSION NO.     V03.80                              *
//*                                                           *
//*************************************************************
//ALOGCHKA  PROC UOUT='B'
//STEP0010  EXEC PGM=CHKSPLA
//STEPLIB   DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSUDUMP  DD   SYSOUT=&UOUT
```

ALOGCHKL

```
//*************************************************************
//*                                                           *
//*       PROC NAME = ALOGCHKL                                *
//*                                                           *
//*       FUNCTION  = A-LOG MONITOR CLOSE CHECK UTILITY       *
//*                                                           *
//*       VERSION NO.     V03.80                              *
//*                                                           *
//*************************************************************
//ALOGCHKL  PROC SOUT='A',UOUT='B'
//STEP0010  EXEC PGM=CHKSPLL
//STEPLIB   DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSUDUMP  DD   SYSOUT=&UOUT
//SYSPRINT  DD   SYSOUT=&SOUT
```

ALOGCPY1

```
//*************************************************************
//*  ALOG  COPY  100                                          *
//*   FUNCTION : VSAM-FILE BACK-UP UTILITY                    *
//*   FILE NO  :  10 => SYS                                   *
//*            :  20 => JAM                                   *
//*            :  80 => DMM                                   *
//*            : 110 => JBI                                   *
//*   VERSION NO.     V03.80                                  *
//*                                                           *
//*************************************************************
//ALOGCPY1 PROC
//UNLDSYS  EXEC PGM=CPY100L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.SYS.BKUP,DISP=(,KEEP),                    Backup File (SYS)
//              UNIT=SYSDA,VOL=SER=VSAM01,
//              SPACE=(CYL,(10,5),RLSE)
//DDPRINT  DD   SYSOUT=A                                           Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYSYS),DISP=SHR            File Name Definition (SYS)
//*
//UNLDJAM  EXEC PGM=CPY100L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.JAM.BKUP,DISP=(,KEEP),                    Backup File (JAM)
//              UNIT=SYSDA,VOL=SER=VSAM01,
//              SPACE=(CYL,(10,5),RLSE)
//DDPRINT  DD   SYSOUT=A                                           Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYJAM),DISP=SHR            File Name Definition (JAM)
//*
//UNLDDMM  EXEC PGM=CPY100L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.DMM.BKUP,DISP=(,KEEP),                    Backup File (DMM)
//              UNIT=SYSDA,VOL=SER=VSAM01,
//              SPACE=(CYL,(10,5),RLSE)
//DDPRINT  DD   SYSOUT=A                                           Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYDMM),DISP=SHR            File Name Definition (DMM)
//*
//UNLDJBI  EXEC PGM=CPY100L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.JBI.BKUP,DISP=(,KEEP),                    Backup File (JBI)
//              UNIT=SYSDA,VOL=SER=VSAM01,
//              SPACE=(CYL,(10,5),RLSE)
//DDPRINT  DD   SYSOUT=A                                           Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYJBI),DISP=SHR            File Name Definition (JBI)
```

ALOGCPY2

```
//*************************************************************
//*                                                           *
//*    ALOG  COPY  200                                        *
//*                                                           *
//*   FUNCTION : VSAM-FILE BACK-UP UTILITY                    *
//*   FILE NO  :  10 => SYS                                   *
//*            :  20 => JAM                                   *
//*            :  80 => DMM                                   *
//*            : 110 => JBI                                   *
//*   VERSION NO.     V03.80                                  *
//*                                                           *
//*************************************************************
//ALOGCPY2 PROC
//LOADSYS  EXEC PGM=CPY200L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.SYS.BKUP,DISP=SHR,                    Backup File (SYS)
//              UNIT=SYSDA,VOL=SER=VSAM01
//DDPRINT  DD   SYSOUT=A                                       Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYSYS),DISP=SHR        File Name Definition (SYS)
//*
//LOADJAM  EXEC PGM=CPY200L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.JAM.BKUP,DISP=SHR,                    Backup File (JAM)
//              UNIT=SYSDA,VOL=SER=VSAM01
//DDPRINT  DD   SYSOUT=A                                       Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYJAM),DISP=SHR        File Name Definition (JAM)
//*
//LOADDMM  EXEC PGM=CPY200L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.DMM.BKUP,DISP=SHR,                    Backup File (DMM)
//              UNIT=SYSDA,VOL=SER=VSAM01
//DDPRINT  DD   SYSOUT=A                                       Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYDMM),DISP=SHR        File Name Definition (DMM)
//*
//LOADJBI  EXEC PGM=CPY200L,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=ALOG.JBI.BKUP,DISP=SHR,                    Backup File (JBI)
//              UNIT=SYSDA,VOL=SER=VSAM01
//DDPRINT  DD   SYSOUT=A                                       Proof List
//SYSUDUMP DD   SYSOUT=A
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(CPYJBI),DISP=SHR        File Name Definition (JBI)
```

ALOGDATE

```
//*********************************************************
//*                                                       *
//*   PROC NAME = ALOGDATE                                *
//*                                                       *
//*        FUNCTION = SET ALOG DATE TO SPL                *
//*                                                       *
//*        PARM     = DATE ... ALOG DATE ... OPTIONAL     *
//*                                                       *
//*        SYSIN    = NOTHING                             *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGDATE PROC PDATE=,UOUT='A',SOUT='A'
//STEP0010 EXEC PGM=DAT01L,PARM='PDATE=&PDATE'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//USMSOUT  DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
```

A-LOG Message

ALOGDLT1

```
//*********************************************************
//*                                                       *
//* PROC NAME = ALOGDLT1                                  *
//*                                                       *
//*   FUNCTION = DELETE JOBLOG,SYSLOG DUMPED DATA AND,    *
//*              DATA RRETENTION DAYS AFTER RESTORE ENTIRE *
//*                                                       *
//*   PARM     : D=JOB   DUMP  END JOBLOG ONLY DELETE     *
//*              : D=SYS   DUMP  END SYSLOG ONLY DELETE    *
//*                                                       *
//*   SYSIN    = NOTHING                                  *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGDLT1 PROC SOUT='A',UOUT='A',REGN=512K,D=JOB
//STEP0010 EXEC PGM=DLT01L,PARM='D=&D',REGION=&REGN
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                          Delete Proof List
//SYSUDUMP DD   SYSOUT=&UOUT
//USMSOUT  DD   SYSOUT=&SOUT                          A-LOG Message
```

ALOGDLT4

```
//*********************************************************
//*                                                       *
//*   PROC NAME = ALOGDLT4                                *
//*                                                       *
//*        FUNCTION = DELETE ENTIRED DMM RECORD           *
//*                                                       *
//*        PARM     = PRINT... Y/N                        *
//*                   DATE ... YYMMDD                     *
//*                                                       *
//*        SYSIN    = NOTHING                             *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGDLT4 PROC PRINT=N,PDATE=,SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=DLT04L,
//              PARM='PRINT=&PRINT,PDATE=&PDATE'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                          Delete Proof List
//SYSUDUMP DD   SYSOUT=&UOUT
//USMSOUT  DD   SYSOUT=&SOUT                          A-LOG Message
```

ALOGDLT5

```
//*******************************************************
//*                                                     *
//*   PROC NAME = ALOGDLT5                              *
//*                                                     *
//*      FUNCTION = DELETE JBI OF ENTIRED REPORT        *
//*                                                     *
//*      PARM     = PDATE='YYMMDD',D='JOB/SYS',         *
//*                 HOLIDAY='N/Y'                       *
//*                                                     *
//*      SYSIN    = NOTHING                             *
//*                                                     *
//*   VERSION NO.   V03.80                              *
//*                                                     *
//*******************************************************
//ALOGDLT5 PROC PDATE=,SOUT='A',UOUT='A',
//              D1=JOB,D2=SYS,HOLIDAY=N
//STEP0010 EXEC PGM=DLT05,PARM='PDATE=&PDATE,D=&D1,HOLIDAY=&HOLIDAY'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                                    Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                                    A-LOG Message
//SYSOUT   DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//HOLTABLE DD   DUMMY                                           Holiday Master
//*
//STEP0020 EXEC PGM=DLT05,PARM='PDATE=&PDATE,D=&D2,HOLIDAY=&HOLIDAY'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                                    Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                                    A-LOG Message
//SYSOUT   DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//HOLTABLE DD   DUMMY                                           Holiday Master
```

ALOGDMPC

```
//*******************************************************
//*                                                     *
//*    DUMP  TAPE COPY UTILITY                          *
//*     SYSUT1  :  INPUT MT                             *
//*     SYSUT2  :  OUTPUT MT (NEED OUTPUT DCB)          *
//*     DDPRINT :  PROOF-LIST (COPIED JBI)              *
//*                                                     *
//*     VERSION NO.   V03.80                            *
//*                                                     *
//*******************************************************
//ALOGDMPC PROC IVOL=,OVOL=,SOUT='A',UOUT='A'
//DMPCPYL  EXEC PGM=DMPCPYL
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDUT1    DD   DSN=ALOG.DUMPL,DISP=(OLD,KEEP),                    Dump MT of Copy Source
//              UNIT=TAPE,VOL=SER=&IVOL,LABEL=(1,SL)
//DDUT2    DD   DSN=ALOG.DUMPL,DISP=(,KEEP),                       Magnetic Tape of Copy Target
//              UNIT=TAPE,VOL=SER=&OVOL,LABEL=(1,SL),
//              DCB=(LRECL=32756,BLKSIZE=32760,RECFM=VB)
//DDPRINT  DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&SOUT                                      A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGDMP1

```
//*******************************************************
//*                                                     *
//*    ALOGDMP1  :  1. A-LOG MONITOR STOP CHECK         *
//*                 2. A-LOG DB(VSAM) BACKUP            *
//*                 3. SELECT DUMP  JOBLOG              *
//*                                                     *
//*******************************************************
//ALOGDMP1 PROC PDATE=,SOUT='A',UOUT='A',HOLIDAY=N
//*******************************************************
//*    CHKSPLL   :  A-LOG MONITOR STOP CHECK            *
//*******************************************************
//STEP0010 EXEC PGM=CHKSPLL
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//USMSOUT  DD   SYSOUT=&SOUT                                    A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//*******************************************************
//*    REPRO     :  VSAM DB BACKUP (SELECTED JBI FILE)    *
//*******************************************************
//STEP0020 EXEC PGM=IDCAMS,REGION=512K
//DD1      DD   DSN=ALOG.DBDATA,DISP=SHR          A-LOG Management Database (Data Part)
//DD2      DD   DSN=&&TEMP3,DISP=(,PASS),                       Intermediate File
//              UNIT=SYSDA,SPACE=(CYL,(5,2),RLSE),
//              DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(ULDDMP),DISP=SHR          VSAM Parameter
//SYSPRINT DD   SYSOUT=&SOUT
//*******************************************************
//*                  EXPAND JOBLOG PS FILE              *
//*******************************************************
//STEP0030 EXEC PGM=ABSIDMP,REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDIREC   DD   DSN=&&TEMP3,DISP=(OLD,DELETE),                 Intermediate File
//              DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760)
//DDOREC   DD   DSN=&&JBIDATA,DISP=(,PASS),          Intermediate File After Conversion
//              UNIT=SYSDA,SPACE=(CYL,(5,2),RLSE),
//              DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760)
//DDPRINT  DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&SOUT                                    A-LOG Message
//SNAPLST  DD   SYSOUT=&UOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//*******************************************************
//*                                                     *
//* SELECT DUMPL JOBLOG                                 *
//* INPUT : EXEC PARM ABEND=Y/N IF USMI ERROR,ABEND OR NOT *
//*       :          PDATE=YYMMDD DUMPL DATE            *
//*       :         HOLIDAY=Y/N  HOLIDAY TABLE USE OR NOT  *
//*       : DDCNTL                                      *
//*       : DDJBI (FROMKEY X'006E' TOKEY                *
//*       :        X'006E' RECORD)                      *
//*                                                     *
//* OUTPUT: DDDLT5 (DLT5 USE PS FILE  ... JOBLOG REC ONLY) *
//*       : DDDUMP (DUMPL & DLT1 USE ... JOBLOG REC ONLY)  *
//*       : DDSOUT (DUMPL USE         ... SYSOUT REC ONLY) *
//*                                                     *
//*******************************************************
//STEP0040 EXEC PGM=DMPJSELS,REGION=512K,
//              PARM='PDATE=&PDATE,HOLIDAY=&HOLIDAY'
```

```
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDJBI    DD   DSN=&&JBIDATA,DISP=(OLD,DELETE)          Intermediate File After Conversion
//DDDLT5   DD   DSN=ALOG.DLT5DATA,DISP=(,KEEP),         Job Log Index File
//             UNIT=SYSDA,VOL=SER=ALOG01,
//             SPACE=(CYL,(50,5),RLSE)
//DDDUMP   DD   DSN=ALOG.DUMPDATA,DISP=(,KEEP),         Job Log Index File
//             UNIT=SYSDA,VOL=SER=ALOG01,
//             SPACE=(CYL,(10,5),RLSE)
//DDSOUT   DD   DSN=ALOG.SOUTDATA,DISP=(,KEEP),         Index File of SYSOUT Data
//             UNIT=SYSDA,VOL=SER=ALOG01,
//             SPACE=(CYL,(30,5),RLSE)
//DDCNTL   DD   DSN=ALOG.V0380.PARMLIB(DMPJOB),DISP=SHR  Dump Control Statement
//HOLTABLE DD   DUMMY                                    Holiday Master
//USMSOUT  DD   SYSOUT=&SOUT                             A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGDMP2

```
//*******************************************************
//*                                                     *
//*    ALOGDMP2  :     <<< A-LOG MONITOR START >>>      *
//*                 1. A-LOG MONITOR ACTIVE CHECK       *
//*                 2. DUMP  JOBLOG                     *
//*                 3. DELETE BKS FILE (DUMPED JOBLOG)  *
//*                                                     *
//*******************************************************
//ALOGDMP2 PROC VOL=,SOUT='A',UOUT='A'
//*******************************************************
//*    A-LOG MONITOR ACTIVE CHECK                       *
//*******************************************************
//STEP0010 EXEC PGM=CHKSPLA
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//USMSOUT  DD   SYSOUT=&SOUT                                  A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//*******************************************************
//*    SORT DUMP    JOBLOG RECORD                       *
//*******************************************************
//STEP0020 EXEC PGM=SORT,REGION=512K
//SORTIN   DD   DSN=ALOG.DUMPDATA,DISP=(SHR,PASS),            Job Log Index File
//              UNIT=SYSDA,VOL=SER=ALOG01
//SORTOUT  DD   DSN=&&DUMPDATA,DISP=(,PASS),                  Sorted Job Log Index File
//              UNIT=SYSDA,SPACE=(CYL,(10,5),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNDMP1),DISP=SHR      Sort Parameter
//SYSOUT   DD   DUMMY
//*******************************************************
//*    SORT SYSOUT JOBLOG RECORD                        *
//*******************************************************
//STEP0030 EXEC PGM=SORT,REGION=512K
//SORTIN   DD   DSN=ALOG.SOUTDATA,DISP=(SHR,PASS),            Index File of SYSOUT Data
//              UNIT=SYSDA,VOL=SER=ALOG01
//SORTOUT  DD   DSN=&&SOUTDATA,DISP=(,PASS),                  Index File of Sorted SYSOUT Data
//              UNIT=SYSDA,SPACE=(CYL,(30,5),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNDMP2),DISP=SHR      Sort Parameter
//SYSOUT   DD   DUMMY
//*******************************************************
//*                                                     *
//*    REPORT DUMP  UTILITY                             *
//*        BKS(EDIT1) ===> MT (EDIT2)                   *
//*                                                     *
//*       PARM : 'PDATE=YYMMDD' DUMP  DATE              *
//*           'ABEND=Y/N' IF USMI ERROR, ABEND OR NOT   *
//*                                                     *
//*******************************************************
//STEP0040 EXEC PGM=DUMPJS,PARM='ABEND=N',REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SORTOUT  DD   DSN=&&DUMPDATA,DISP=(OLD,PASS)
//DDSOUT   DD   DSN=&&SOUTDATA,DISP=(OLD,DELETE)
```

```
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                   A-LOG Data Pool (Data Part)
//DDOUT    DD   DSN=ALOG.DUMPL,DISP=(,KEEP),               Dump MT
//             UNIT=TAPE,VOL=SER=&VOL,LABEL=(1,SL),
//             DCB=(LRECL=32756,BLKSIZE=32760,RECFM=VB)
//SYSPRINT DD   SYSOUT=&SOUT                               Dump Proof List
//USMSOUT  DD   SYSOUT=&SOUT                               A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//*********************************************************
//*                                                       *
//*    DELETE BKS OF ENTIRED JOBLOG                       *
//*                                                       *
//*        PARM : 'D=JOB' DUMP  JBI FILE DELETE(BKS)      *
//*                                                       *
//*********************************************************
//STEP0050 EXEC PGM=DLT01JS,PARM='D=JOB'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDFILE   DD   DSN=&&DUMPDATA,DISP=(OLD,DELETE)
//SYSPRINT DD   SYSOUT=&SOUT                               Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                               A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGDMP3

```
//*******************************************************
//*                                                     *
//*    ALOGDMP3  :  1. DELETE JBI FILE ON A-LOG DATA BASE  *
//*                                                     *
//*******************************************************
//ALOGDMP3 PROC PDATE=,SOUT='A',UOUT='A',HOLIDAY=N
//*******************************************************
//*    SORT DUMPL   JOBLOG RECORD                       *
//*******************************************************
//STEP0010 EXEC PGM=SORT,REGION=512K
//SORTIN   DD   DSN=ALOG.DLT5DATA,DISP=(SHR,PASS),                  Job Log Index File
//              UNIT=SYSDA,VOL=SER=ALOG01
//SORTOUT  DD   DSN=&&DLT5DATA,DISP=(,PASS),                        Sorted Job Log Index File
//              UNIT=SYSDA,SPACE=(CYL,(50,5),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNDMP3),DISP=SHR            Sort Parameter
//SYSOUT   DD   DUMMY
//*******************************************************
//*    SORT SYSOUT JOBLOG RECORD                        *
//*******************************************************
//STEP0020 EXEC PGM=SORT,REGION=512K
//SORTIN   DD   DSN=ALOG.SOUTDATA,DISP=(SHR,PASS),          Index File of SYSOUT Data
//              UNIT=SYSDA,VOL=SER=ALOG01
//SORTOUT  DD   DSN=&&SOUTDATA,DISP=(,PASS),                Index File of Sorted SYSOUT Data
//              UNIT=SYSDA,SPACE=(CYL,(30,5),RLSE),
//              DCB=(LRECL=794,BLKSIZE=7940,RECFM=FB)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(10,5))
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNDMP4),DISP=SHR            Sort Parameter
//SYSOUT   DD   DUMMY
//*******************************************************
//*                                                     *
//*  DELETE JBI FILE ON A-LOG DATA BASE                 *
//*         DDJBI  ---  JOBLOG INDEX RECORD             *
//*         DDSBI  ---  JOBLOG SYSOUT RECORD            *
//*                                                     *
//*  PARM : 'PDATE=YYMMDD' PURGE DATE                   *
//*          'HOLIDAY=Y/N'  HOLIDAY TABLE USE OR NOT USE  *
//*                                                     *
//*******************************************************
//STEP0030 EXEC PGM=DLT05JS,PARM='PDATE=&PDATE,HOLIDAY=&HOLIDAY'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDJBI    DD   DSN=&&DLT5DATA,DISP=(OLD,DELETE)
//DDSBI    DD   DSN=&&SOUTDATA,DISP=(OLD,DELETE)
//HOLTABLE DD   DUMMY                                               Holiday Master
//SYSPRINT DD   SYSOUT=&SOUT                                        Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                                        A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGDMP4

```
//*******************************************************
//*                                                     *
//*    ALOGDMP4  :  DELETE WORK PS FILE                 *
//*                                                     *
//*******************************************************
//ALOGDMP4 PROC
//DELETE EXEC PGM=IEFBR14
//DEL1   DD   DSN=ALOG.DLT5DATA,DISP=(OLD,DELETE),
//            UNIT=SYSDA,VOL=SER=ALOG01
//DEL2   DD   DSN=ALOG.DUMPDATA,DISP=(OLD,DELETE),
//            UNIT=SYSDA,VOL=SER=ALOG01
//DEL3   DD   DSN=ALOG.SOUTDATA,DISP=(OLD,DELETE),
//            UNIT=SYSDA,VOL=SER=ALOG01
```

ALOGHOLT

```
//*******************************************************
//*                                                     *
//*        A-LOG HOLIDAY TABLE CONTROL                  *
//*                                                     *
//*  FUNCTION = HOLIDAY TABLE CREATE/UPDATE/DELETE/INQ  *
//*                                                     *
//*  PARM     = LIST : YY / NN LIST CONTROL             *
//*           = ID   : MEMBER NAME OF HOLIDAY TABLE     *
//*           = ZZ   : END OF DATA                      *
//*                                                     *
//*     VERSION NO.   V0380                             *
//*                                                     *
//*******************************************************
//ALOGHOLT PROC SOUT='A',UOUT='B'
//STEP010  EXEC PGM=AUTOHOLT,
//              PARM='YY',REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//HOLTABLE DD   DSN=ALOG.HOLTABLE,DISP=SHR                    Holiday Master
//SORTIN   DD   UNIT=SYSDA,SPACE=(TRK,(3,3))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(TRK,(3,3))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(TRK,(3,3))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(TRK,(3,3))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(TRK,(3,3))
//SYSPRINT DD   SYSOUT=&SOUT                          Proof List, Inquiry List
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSOUT   DD   SYSOUT=&SOUT
//*-------------------------------------------------*
//*   HOW TO DEFINE SYSIN CARD                      *
//*       PLEASE REFER TO MEMBER 'ALOGHOLT'         *
//*       OF ALOG/V0380 INSTALL LIBRARY OR MANUAL   *
//*-------------------------------------------------*
//SYSIN    DD   DUMMY                                Holiday Master Definition Card
```

ALOGIBKS

```
//*********************************************************
//*                                                       *
//*    BKS INITIALIZE UTILITY                             *
//*                                                       *
//*********************************************************
//ALOGIBKS PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=SCTRUNL,PARM='TYPE=ALOG',REGION=512K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDCARD   DD   DSN=ALOG.V0380.PARMLIB(SCTINIT),              Initialization Utility Program Name
//              DISP=SHR
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(BKSPARM),DISP=SHR      Initialization Parameter
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                      A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                      A-LOG Data Pool (Data Part)
//DDPRINT  DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&UOUT                                  A-LOG-Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGICHK

```
//*******************************************************
//*                                                     *
//*    ALOGICHK                                         *
//*      STEP0010  VERIFY ALOG VSAM DATA BASE           *
//*      STEP0020  UNLOAD VSAM DB TO SAM FILE           *
//*      STEP0030  SORT INDEX FILE( KEY:FILE-NO,REC-NO) *
//*      STEP0040  RENUMBER REC-NO AND RECORD           *
//*     NOTE : VSAM UTILITY IDCAMS(IBM), KQCAMS(FACOM), *
//*                          JSCVSUT(HITACH)            *
//*                                                     *
//*******************************************************
//ALOGICHK PROC UTL=IDCAMS,
//             WVOL=ALOG01,            VOL OF WORK
//             ISPC='CYL,(10,5)',      SPACE OF INDEX
//             DSPC='CYL,(30,5)',      SPACE OF DATA
//             WSPC='CYL,(10,5)'       SPACE OF SORT WORK
//*
//* -----  STEP0010  VERIFY A-LOG DATABASE-----------
//*
//STEP0010 EXEC PGM=&UTL,REGION=512K
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR                A-LOG Management Database (Index Part)
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR                A-LOG Management Database (Data Part)
//SYSPRINT DD   SYSOUT=A
//SYSUDUMP DD   SYSOUT=A
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(VERPARM),DISP=SHR  VSAM Parameter
//* -----  STEP0020  UNLOAD VSAM DB TO SAM FILE  ---------
//STEP0020 EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=A
//SYSUDUMP DD   SYSOUT=A
//BKIX     DD   DSN=ALOG.BKIDEX,DISP=(,KEEP),           Backup File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&ISPC),
//              DCB=(BLKSIZE=19069,RECFM=VB)
//BKDT     DD   DSN=ALOG.BKDATA,DISP=(,KEEP),           Backup File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&DSPC),
//              DCB=(BLKSIZE=19069,RECFM=VB)
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR                A-LOG Management Database (Index Part)
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR                A-LOG Management Database (Data Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(UNLDPARM),DISP=SHR  VSAM Parameter
//* -----  STEP0030  SORT INDEX FILE( KEY:FILE-NO,REC-NO) -
//STEP0030 EXEC PGM=SORT,COND=(0,NE,STEP0020)
//SORTIN   DD   DSN=ALOG.BKIDEX,DISP=SHR,               Backup File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,
//              DCB=(BLKSIZE=19069,RECFM=VB)
//SORTOUT  DD   DSN=ALOG.S1IDEX,DISP=(,PASS),           Sorted Backup File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&ISPC)
//SYSOUT   DD   SYSOUT=A
//SYSDUMP  DD   SYSOUT=A
//SORTWK01 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(SORTIDX1),DISP=SHR  Sort Parameter
//* -----  STEP0040  RENUMBER REC-NO AND RECORD  ------
//STEP0040 EXEC PGM=ABSICNVL
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//ABASIDXI DD   DSN=ALOG.S1IDEX,DISP=SHR,               Sorted Backup File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL
```

```
//ABASDATI DD   DSN=ALOG.BKDATA,DISP=SHR,                    Backup File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL
//ABASIDXO DD   DSN=ALOG.RNIDEX,DISP=(,PASS),                Intermediate File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&ISPC),
//              DCB=(BLKSIZE=19069,RECFM=VB)
//ABASDATO DD   DSN=ALOG.RNDATA,DISP=(,PASS),                Intermediate File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&DSPC),
//              DCB=(BLKSIZE=19069,RECFM=VB)
//ERRLIST  DD   SYSOUT=A                                     Database Unmatched Record List
//SYSUDUMP DD   SYSOUT=A
//DDSNAP   DD   SYSOUT=A
```

ALOGIUPD

```
//*******************************************************
//*                                                     *
//*    ALOGIUPD                                         *
//*      STEP0050  SORT INDEX FILE (PRIMARY KEY(64))    *
//*      STEP0060  SORT DATA  FILE (PRIMARY KEY(6))     *
//*      STEP0070  DELETE VSAM FILE                     *
//*      STEP0080  DEFINE VSAM FILE                     *
//*      STEP0090  LOAD BACKUP SAM FILE TO VSAM FILE    *
//*     NOTE : VSAM UTILITY IDCAMS(IBM), KQCAMS(FACOM), *
//*                          JSCVSUT(HITACH)            *
//*                                                     *
//*******************************************************
//ALOGIUPD PROC UTL=IDCAMS,
//             DVOL=ALOG01,          VOL OF VSAM FILE
//             WVOL=ALOG01,          VOL OF WORK
//             ISPC='CYL,(10,5)',    SPACE OF INDEX
//             DSPC='CYL,(30,5)',    SPACE OF DATA
//             WSPC='CYL,(10,5)'     SPACE OF SORT WORK
//*
//*     STEP0050  :   SORT INDEX FILE
//*
//STEP0050 EXEC PGM=SORT           ,COND=(4,LT,STEP0030)
//SORTIN   DD   DSN=ALOG.RNIDEX,DISP=SHR,                    Intermediate File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,
//              DCB=(BLKSIZE=19069,RECFM=VB)
//SORTOUT  DD   DSN=ALOG.S2IDEX,DISP=(,PASS),                Sorted Intermediate File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&ISPC)
//SYSOUT   DD   SYSOUT=A
//SYSDUMP  DD   SYSOUT=A
//SORTWK01 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(SORTIDX2),DISP=SHR    Sort Parameter
//*     STEP0060  :   SORT DATA FILE
//STEP0060 EXEC PGM=SORT
//SORTIN   DD   DSN=ALOG.RNDATA,DISP=SHR,                    Intermediate File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,
//              DCB=(BLKSIZE=19069,RECFM=VB)
//SORTOUT  DD   DSN=ALOG.S2DATA,DISP=(,PASS),                Sorted Intermediate File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&DSPC)
//SYSOUT   DD   SYSOUT=A
//SYSDUMP  DD   SYSOUT=A
//SORTWK01 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(&WSPC)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(SORTDATA),DISP=SHR    Sort Parameter
//*     STEP0070  :   DELETE
//STEP0070 EXEC PGM=&UTL,REGION=512K,COND=(0,NE,STEP0050)
//SYSPRINT DD   SYSOUT=A
//VSAMFILE DD   UNIT=SYSDA,                                  Existing A-LOG Management Database
//              VOL=SER=&DVOL,
//              DISP=OLD
//SYSUDUMP DD   SYSOUT=A
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DLTIPARM),DISP=SHR    VSAM Parameter
//         DD   DSN=ALOG.V0380.PARMLIB(DLTDPARM),DISP=SHR    VSAM Parameter
```

```
//*     STEP0080  :   DEFINE
//STEP0080 EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=A
//VSAMFILE DD   UNIT=SYSDA,                                  New A-LOG Management Database
//              VOL=SER=&DVOL,
//              DISP=OLD
//SYSUDUMP DD   SYSOUT=A
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DEFIPARM),DISP=SHR    VSAM Parameter
//         DD   DSN=ALOG.V0380.PARMLIB(DEFDPARM),DISP=SHR    VSAM Parameter
//*     STEP0090  :   ALOG DATA BASE FILE RELOAD
//STEP0090 EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=A
//SYSUDUMP DD   SYSOUT=A
//BKIX     DD   DSN=ALOG.S2IDEX,DISP=OLD,                    Sorted Intermediate File (Index Part)
//              UNIT=SYSDA,VOL=SER=&WVOL
//BKDT     DD   DSN=ALOG.S2DATA,DISP=OLD,                    Sorted Intermediate File (Data Part)
//              UNIT=SYSDA,VOL=SER=&WVOL
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR                     New A-LOG Management Database (Index Part)
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR                     New A-LOG Management Database (Data Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(LOADPARM),DISP=SHR    VSAM Parameter
```

ALOGJDMP

```
//*******************************************************
//*                                                     *
//*  JOBLOG DUMP  UTILITY                               *
//*     BKS(EDIT1) ===> MT (EDIT2)                      *
//*                                                     *
//*     VERSION NO.   V03.80                            *
//*                                                     *
//*******************************************************
//ALOGJDMP PROC SOUT='A',UOUT='A',D=JOB,
//              PDATE=,DMPVOL=VVVVVV,HOLIDAY=N,REGIN=2048K
//STEP0010 EXEC PGM=DUMPL,
//              PARM='ABEND=N,PDATE=&PDATE,HOLIDAY=&HOLIDAY'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                  A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                  A-LOG Data Pool (Index Part)
//HOLTABLE DD   DUMMY                                     Holiday Master
//DDOUT    DD   DSN=ALOG.DUMPJOB,DISP=(,KEEP),            Dump MT
//              UNIT=(TAPE,,DEFER),
//              VOL=SER=&DMPVOL,LABEL=(1,SL),
//              DCB=(LRECL=32756,BLKSIZE=32760,RECFM=VB)
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSOUT   DD   DUMMY
//SYSPRINT DD   SYSOUT=&SOUT                            Dump Proof List
//USMSOUT  DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//DDCNTL   DD   DSN=ALOG.V0380.PARMLIB(DMP&D),DISP=SHR  Dump Control Statement
//STEP0020 EXEC PGM=DLT01L,PARM='D=&D',REGION=&REGIN
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                            Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                            A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGJLOG

```
//*********************************************************
//*                                                       *
//*  ALOGJLOG  ...  JOB LOG CATALOG PROCESS               *
//*                                                       *
//*       DDJLOG  =  JOB LOG CONTROL CARD(INPUT)          *
//*       DDCAT   =  JOB LOG ANALYZE MODULE NAME(INPUT)   *
//*                  'EXT=CATJLOG '                       *
//*        SYSUT1  =  JOB LOG WORK FILE(INPUT/OUTPUT)     *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGJLOG PROC SOUT='A',UOUT='A',
//              CLASS=J,P=DDJLOGI
//JDJPROC  EXEC PGM=ASPLJLOG,PARM='&CLASS',REGION=1024K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                    A-LOG Data Pool (Data Part)
//DDCAT    DD   DSN=ALOG.V0380.PARMLIB(DDCATJ),DISP=SHR     DDCATJ Parameter
//DDJLOG   DD   DSN=ALOG.V0380.PARMLIB(&P),DISP=SHR         Job Definition Card
//         DD   DSN=ALOG.V0380.PARMLIB(DDJLOGP),DISP=SHR    Job Log Message Definition Card
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,(20,10)),             Job Log Work File
//              DCB=(BLKSIZE=14200)
//DDJEXT   DD   DUMMY           /* USER EXIT ROUTINE */     User Exit Module Name
//DDCNTL   DD   DUMMY
//DDEXC    DD   DUMMY
//DDJUTL   DD   DUMMY                                       JCL Analysis Exit Module Name
//*DDWTR   DD   DUMMY                                       JCL Storage Library of WTRJUTL
//*DDSNAP  DD   SYSOUT=&SOUT
//*DBSNAP  DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&SOUT                                A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGJMST

```
//*******************************************************
//*                                                     *
//*   PROC NAME = ALOGJMST                              *
//*                                                     *
//*   FUNCTION  = MAKE JOBLOG MASTER CHKLST             *
//*               MAKE JAM FILE                         *
//*                                                     *
//*   PARM      = &ERCHK&DUPCHK                         *
//*                                                     *
//*   SYSIN     = JOBLOG MASTER FILE CARDS              *
//*                                                     *
//*   VERSION NO.     V03.80                            *
//*                                                     *
//*******************************************************
//ALOGJMST PROC ERCHK='1',DUPCHK='1',
//              SOUT='A',UOUT='A'
//ST010    EXEC PGM=JAM100,PARM='ERCHK=&ERCHK,
//              DUPCHK=&DUPCHK'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SORTIN   DD   UNIT=SYSDA,
//              SPACE=(TRK,(5,5))
//SORTOUT  DD   UNIT=SYSDA,
//              SPACE=(TRK,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSOUT   DD   DUMMY
//USMSOUT  DD   SYSOUT=&SOUT                    A-LOG Message
//SYSPRINT DD   SYSOUT=&SOUT                    Definition Card Check List
//DDJLIST  DD   SYSOUT=&SOUT                    Master List
//SYSUDUMP DD   SYSOUT=&UOUT
//DDKANJI  DD   DUMMY                           Master Definition Card
//SYSIN    DD   DUMMY
```

ALOGJRST

```
//*******************************************************
//*                                                     *
//*     JOBLOG RESTORE  UTILITY                         *
//*                                                     *
//*      MT (EDIT2) ===> BKS(EDIT1)                     *
//*                                                     *
//*      VERSION NO.   V03.80                           *
//*                                                     *
//*******************************************************
//ALOGJRST PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=RESTOREL,PARM='ABEND=N',REGION=2048K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                    A-LOG Data Pool (Data Part)
//*DDIN     DD   DSN=RESTOREL,                              Dump MT
//*              DISP=(OLD,KEEP),UNIT=(TAPE,,DEFER),
//*              VOL=SER=XXXXXX
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,10)
//USMSCARD DD   DSN=ALOG.V0380.PARMLIB(USMSJRST),DISP=SHR  Restore Control Parameter
//SYSOUT   DD   DUMMY
//DDSNAP   DD   DUMMY             /* DYNALLOC MESSAGE */
//SYSPRINT DD   SYSOUT=&SOUT                               Restore Proof List
//USMSOUT  DD   SYSOUT=&SOUT                               A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGLBKS

```
//*******************************************************
//*                                                     *
//*        LOAD BACK UPED DATA TO A-LOG DATA POOL       *
//*                                                     *
//*******************************************************
//ALOGLBKS PROC BKVOL=VVVVVV,    VOLSER FOR BACKUP
//              SOUT=A,
//              UOUT=A
//*
//BKSLDEL  EXEC PGM=BKSLDEL,PARM='TYPE=ALOG,SIZE=2000K',
//              REGION=3000K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDPRINT  DD   SYSOUT=&SOUT                                  Proof List
//USMSOUT  DD   SYSOUT=&UOUT                                  A-LOG Message
//LOGPRINT DD   SYSOUT=&UOUT
//DBSNAP   DD   SYSOUT=&UOUT
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                      A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                      A-LOG Data Pool (Data Part)
//DDLOAD   DD   DSN=ALOG.BKUPBKS,DISP=SHR,
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(1,SL),
//              DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760)      Backup File
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGLDMP

```
//*********************************************************
//*                                                       *
//*   SYSLOG DUMP  UTILITY                                *
//*       BKS(EDIT1) ===> MT (EDIT2)                      *
//*                                                       *
//*       VERSION NO.   V03.80                            *
//*                                                       *
//*********************************************************
//ALOGLDMP PROC SOUT='A',UOUT='A',D=SYS,
//              PDATE=,DMPVOL=VVVVVV,HOLIDAY=N,REGIN=512K
//STEP0010 EXEC PGM=DUMPSLOG,
//              PARM='ABEND=N,PDATE=&PDATE,HOLIDAY=&HOLIDAY',REGION=2048K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                    A-LOG Data Pool (Control Part)
//HOLTABLE DD   DUMMY                                       Holiday Master
//DDOUT    DD   DSN=ALOG.DUMPSYS,DISP=(,KEEP),              Dump MT
//              UNIT=(TAPE,,DEFER),
//              VOL=SER=&DMPVOL,LABEL=(1,SL),
//              DCB=(LRECL=32756,BLKSIZE=32760,RECFM=VB)
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSOUT   DD   DUMMY
//SYSPRINT DD   SYSOUT=&SOUT                              Dump Proof List
//USMSOUT  DD   SYSOUT=&SOUT                              A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//DDCNTL   DD   DSN=ALOG.V0380.PARMLIB(DMP&D),DISP=SHR    Dump Control Statement
//STEP0020 EXEC PGM=DLT01L,PARM='D=&D',REGION=&REGIN
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SYSPRINT DD   SYSOUT=&SOUT                              Delete Proof List
//USMSOUT  DD   SYSOUT=&SOUT                              A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGLLOG

```
//*********************************************************
//*                                                       *
//*   PROC NAME = ALOGLLOG                                *
//*                                                       *
//*   STEP0010 : MAKE SYSLOG FREE FILE                    *
//*              PARM='DATE=YYMMDD(YYMMDD)'               *
//*              FROM DATE (YYMMDD)                       *
//*              TO   DATE (YYMMDD)                       *
//*   STEP0020 : SORT SYSLOG FREE FILE                    *
//*              SYSIN=RUNLOG5                            *
//*   STEP0030 : LIST OUT SYSLOG FREE FILE                *
//*              PARM='DATE=YYMMDD(YYMMDD),               *
//*                    CODE=ALL/NOR/ERR,                  *
//*                    SYSOUT=YES/NO'                     *
//*                                                       *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGLLOG PROC PDATE=,PCODE=ALL,POUT=YES,
//              SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=LOG920,PARM='PDATE=&PDATE'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDBLOG   DD   DSN=&&LLOG1,DISP=(,PASS),                    SYSLOG Index Release File
//              UNIT=SYSDA,SPACE=(CYL,(5,5)RLSE)
//DDPRINT  DD   SYSOUT=&SOUT                                 Proof List
//USMSOUT  DD   SYSOUT=&SOUT                                 A-LOG Message
//SYSUDUMP DD   SYSOUT=&SOUT
//*
//STEP0020 EXEC PGM=SORT,COND=(0,NE,STEP0010)
//SORTIN   DD   DSN=&&LLOG1,DISP=(OLD,DELETE)                SYSLOG Index Release File
//SORTOUT  DD   DSN=&&LLOG2,DISP=(,PASS),                    Sorted SYSLOG Index Release File
//              UNIT=SYSDA,SPACE=(CYL,(5,5)RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSOUT   DD   DUMMY
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(RUNLOG5),DISP=SHR     Sort Parameter
//*
//STEP0030 EXEC PGM=LOG930,COND=(0,NE,STEP0020),
//              PARM='PDATE=&PDATE,CODE=&PCODE,SYSOUT=&POUT'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDRLOG   DD   DSN=&&LLOG2,DISP=(OLD,DELETE)                Sorted SYSLOG Index Release File
//DDPRINT  DD   SYSOUT=&SOUT                                 SYSLOG Audit Trail List
//USMSOUT  DD   SYSOUT=&SOUT                                 A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGLOAD

```
//*******************************************************
//*                                                     *
//*    LOAD BACK A-LOG VSAM DATABASE                    *
//*                                                     *
//*******************************************************
//ALOGLOAD PROC UTL=IDCAMS,
//              BKVOL='VVVVVV',    BACK UP VOLUME OF INDEX
//              DVOI='ALOG01',     DB VOLUME OF INDEX
//              DVOD='ALOG01',     DB VOLUME OF DATA
//              SOUT='A',
//              UOUT='A'
//*******************************************************
//*         DELETE INDEX DB                             *
//*******************************************************
//DELIX    EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DLTIPARM),DISP=SHR          VSAM Parameter
//*******************************************************
//*         DELETE DATA DB                              *
//*******************************************************
//DELDT    EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DLTDPARM),DISP=SHR          VSAM Parameter
//*******************************************************
//*         DEFINE INDEX DB                             *
//*******************************************************
//DEFIX    EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//VSAMFILE DD   UNIT=SYSDA,                    A-LOG Management Database (Index Part)
//              VOL=SER=&DVOI,
//              DISP=OLD
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DEFIPARM),DISP=SHR          VSAM Parameter
//*******************************************************
//*         DEFINE DATA DB                              *
//*******************************************************
//DEFDT    EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//VSAMFILE DD   UNIT=SYSDA,                    A-LOG Management Database (Data  Part)
//              VOL=SER=&DVOD,
//              DISP=OLD
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(DEFDPARM),DISP=SHR          VSAM Parameter
//*******************************************************
//*         LOAD BACK UP INDEX DB                       *
//*******************************************************
//LOADIX   EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//BKIX     DD   DSN=ALOG.BKIDEX,DISP=(SHR,PASS),                    Backup File
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(1,SL)
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR       A-LOG Management Database (Index Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(LDIXPARM),DISP=SHR          VSAM Parameter
```

```
//*******************************************************
//*              LOAD BACK UP DB                        *
//*******************************************************
//LOADDT   EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//BKDT     DD   DSN=ALOG.BKDATA,DISP=(SHR,PASS),                    Backup File
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(2,SL)
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR          A-LOG Management Database (Data Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(LDDTPARM),DISP=SHR          VSAM Parameter
```

ALOGLRST

```
//*********************************************************
//*                                                       *
//*  SYSLOG RESTORE  UTILITY                              *
//*                                                       *
//*     MT (EDIT2) ===> BKS(EDIT1)                        *
//*                                                       *
//*     VERSION NO.   V03.80                              *
//*                                                       *
//*********************************************************
//ALOGLRST PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=RESTSLOG,PARM='ABEND=N',REGION=2048K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                    A-LOG Data Pool (Data Part)
//*DDIN     DD   DSN=RESTOREL,                              Dump MT
//*              DISP=(OLD,KEEP),UNIT=(TAPE,,DEFER),
//*              VOL=SER=XXXXXX
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,10)
//SYSOUT   DD   DUMMY
//DDSNAP   DD   DUMMY             /* DYNALLOC MESSAGE */
//USMSCARD DD   DSN=ALOG.V0380.PARMLIB(USMSLRST),DISP=SHR  Restore Control Parameter
//SYSPRINT DD   SYSOUT=&SOUT                               Restore Proof List
//USMSOUT  DD   SYSOUT=&SOUT                               A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGRENO

```
//*************************************************************
//*    ALOGRENO                                               *
//*      STEPICHK  CHECK THE CONSISTENCY OF VSAM FILE         *
//*      STEPIUPD  UPDATE VSAM FILE IF THERE IS               *
//*                ANY INCONSISTENCY IN VSAM FILE             *
//*     VERSION NO.      V03.80                               *
//*                                                           *
//*************************************************************
//ALOGRENO PROC UTL=IDCAMS,
//             WVOL=ALOG01,             VOL OF WORK
//             DVOL=ALOG01,             VOL OF VSAM FILE
//             ISPC='CYL,(10,5)',       SPACE OF INDEX
//             DSPC='CYL,(30,5)',       SPACE OF DATA
//             WSPC='CYL,(10,5)',       SPACE OF SORT WORK
//*
//****     CHECK VSAM FILE
//*
//STEPICHK EXEC ALOGICHK
//*--------  ALOG.DBIDEX BACK UP FILE  --------
//STEP0020.BKIX     DD DSN=ALOG.WK.BKIDEX,DISP=(,KEEP),
//                     UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&ISPC),
//                     DCB=(BLKSIZE=19069,RECFM=VB)
//*--------  ALOG.DBDATA BACK UP FILE  --------
//STEP0020.BKDT     DD DSN=ALOG.WK.BKDATA,DISP=(,KEEP),
//                     UNIT=SYSDA,VOL=SER=&WVOL,SPACE=(&DSPC),
//                     DCB=(BLKSIZE=19069,RECFM=VB)
//*--------  ALOG.DBIDEX BACK UP FILE  --------
//STEP0030.SORTIN   DD DSN=ALOG.WK.BKIDEX,DISP=SHR,
//                     UNIT=SYSDA,VOL=SER=&WVOL
//*--------  ALOG.DBDATA BACK UP FILE  --------
//STEP0040.ABASDATI DD DSN=ALOG.WK.BKDATA,DISP=SHR,
//                     UNIT=SYSDA,VOL=SER=&WVOL
//*
//****     UPDATE VSAM FILE
//*
//STEPIUPD EXEC ALOGIUPD,COND=(0,NE,STEPICHK.STEP0020)
```

ALOGREP

```
//*********************************************************
//*                                                       *
//*   PROC NAME = ALOGREP                                 *
//*                                                       *
//*        FUNCTION = A-LOG DATABASE REPORT               *
//*                                                       *
//*        PARM     = NOTHING                             *
//*                                                       *
//*       VERSION NO.   V03.80                            *
//*                                                       *
//*********************************************************
//ALOGREP  PROC SOUT='A',UOUT='A'
//VSAMREP  EXEC PGM=ABSIREPL
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//CHKLIST  DD   SYSOUT=&SOUT                    A-LOG Management Database Report
//ABSIX001 DD   DSN=ALOG.DBIDEX,DISP=SHR        A-LOG Management Database (Index Part)
//ABSID001 DD   DSN=ALOG.DBDATA,DISP=SHR        A-LOG Management Database (Data Part)
//SYSUDUMP DD   SYSOUT=&UOUT
//*
//*        BKS REPORT UTILITY
//*        PROGRAM ... BKSREPL
//*        PARM ... BKSPARM
//*        V03.80
//*
//BKSREPL  EXEC PGM=SCTRUNL,REGION=512K,PARM='TYPE=ALOG'
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR        A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR        A-LOG Data Pool (Data Part)
//DDCARD   DD   DSN=ALOG.V0380.PARMLIB(SCTREP),  Report Utility Program Name
//              DISP=SHR
//DDSYSIN  DD   DSN=ALOG.V0380.PARMLIB(BKSPARM),DISP=SHR
//DDPRINT  DD   SYSOUT=&SOUT                    A-LOG Data Pool Report
//SYSUDUMP DD   SYSOUT=&UOUT                    A-LOG Message
```

ALOGRSEL

```
//*************************************************************
//*                                                           *
//*   JOBLOG/SYSLOG RESTORE  INDEX SELECT UTILITY             *
//*                                                           *
//*       VERSION NO.   V03.80                                *
//*                                                           *
//*************************************************************
//ALOGRSEL PROC SOUT='A',UOUT='A'
//RESTSELL EXEC PGM=RESTSELL
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(3,2),RLSE)
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(3,2),RLSE)
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//DDCNTL   DD   DSN=ALOG.V0380.PARMLIB(RSELPARM),DISP=SHR      Restore Job Generation Parameter
//DDJCL    DD   DSN=ALOG.V0380.PARMLIB,DISP=SHR                JCL Library
//DDRDR    DD   SYSOUT=(A,INTRDR)                              Internal Reader
//SYSOUT   DD   DUMMY
//USMSOUT  DD   SYSOUT=&SOUT                                   A-LOG Message
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGSECR

```
//*********************************************************
//*                                                       *
//*   ALOGSECR  ...  A-LOG SECRITY MAINTENANCE UTILITY    *
//*                  DDSYSIN ...  INPUT PARAMETER CARD    *
//*                                                       *
//*   VERSION NO.   V03.80                                *
//*                                                       *
//*********************************************************
//ALOGSECR PROC SOUT='A',UOUT='A'
//STEP0010 EXEC PGM=AUTOSECR
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//SECRTBL  DD   DSN=ALOG.SECRTBL,DISP=SHR                    Security Management Table
//SORTIN   DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTOUT  DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//AUTOUT2  DD   UNIT=SYSDA,SPACE=(CYL,(2,1))
//SECRLOGX DD   DSN=ALOG.SECRLOGX,DISP=SHR                   Security Log File (X)
//SECRLOGY DD   DSN=ALOG.SECRLOGY,DISP=SHR                   Security Log File (Y)
//LOGWK    DD   UNIT=SYSDA,SPACE=(CYL,(1,1))
//LOGPRINT DD   SYSOUT=&SOUT
//AUTOPRT1 DD   SYSOUT=&SOUT                                 Proof List
//AUTOPRT2 DD   SYSOUT=&SOUT                                 Inquiry List
//SYSOUT   DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//DDSYSIN  DD   DUMMY                                        Utility Command
```

ALOGSLOG

```
//*******************************************************
//*                                                     *
//*   ALOGSLOG  ...  SYSLOG  CATALOG PROCESS            *
//*                                                     *
//*        DDSCNTL =  SYSLOG  CONTROL CARD(INPUT)       *
//*                                                     *
//*   VERSION NO.   V03.80                              *
//*                                                     *
//*******************************************************
//ALOGSLOG PROC SOUT='A',UOUT='A',
//              CNTL=DDSCNTL,DDCATS=DDCATS
//JDJPROC  EXEC PGM=ASPLSLOG,REGION=2048K,TIME=1440
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                    A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                    A-LOG Data Pool (Data Part)
//DDCAT    DD   DSN=ALOG.V0380.PARMLIB(&DDCATS),DISP=SHR    DDCATS Parameter
//DDJLOG   DD   DUMMY
//DDSCNTL  DD   DSN=ALOG.V0380.PARMLIB(&CNTL),DISP=SHR      DDSCNTL Parameter
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,(20,10)),             SYSLOG Work File
//              DCB=(BLKSIZE=14200)
//DDJEXT   DD   DUMMY       /* USER EXIT ROUTINEC/*        User Exit Module Name
//DDCNTL   DD   DUMMY
//DDEXC    DD   DUMMY
//USMSOUT  DD   SYSOUT=&SOUT                                A-LOG Message
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGUBKS

```
//*********************************************************
//*                                                       *
//*        BACK UP A-LOG DATA POOL                        *
//*                                                       *
//*********************************************************
//ALOGUBKS PROC BKVOL=VVVVVV,           VOLSER FOR BACK UP
//              SOUT=A,
//              UOUT=A
//*
//BKSUDEL  EXEC PGM=BKSUDEL,PARM='TYPE=ALOG,SIZE=2000K',
//              REGION=3000K
//STEPLIB  DD   DSN=ALOG.V0380.LOAD,DISP=SHR
//DDPRINT  DD   SYSOUT=&SOUT
//USMSOUT  DD   SYSOUT=&UOUT                                    A-LOG Message
//LOGPRINT DD   SYSOUT=&UOUT
//DBSNAP   DD   SYSOUT=&UOUT
//DDSCT    DD   DSN=ALOG.DB.SCT,DISP=SHR                        A-LOG Data Pool (Control Part)
//DDBKS    DD   DSN=ALOG.DB.BKS,DISP=SHR                        A-LOG Data Pool (Data Part)
//DDUNLD   DD   DSN=ALOG.BKUPBKS,DISP=(,KEEP),                  Backup File
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(1,SL),
//              DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760)
//SYSUDUMP DD   SYSOUT=&UOUT
```

ALOGUNLD

```
//*******************************************************
//*                                                     *
//*    UNLOAD A-LOG VSAM DATABASE                       *
//*                                                     *
//*******************************************************
//ALOGUNLD PROC UTL=IDCAMS,                   UTILITY(AMS)
//              BKVOL='VVVVVV',  BACK UP VOLUME FOR A-LOG DB
//              SOUT='A',
//              UOUT='A'
//*******************************************************
//*         VERIFY ABAS DB                              *
//*******************************************************
//STEP0010 EXEC PGM=IDCAMS,REGION=512K
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=OLD                  A-LOG Management Database (Index Part)
//DBDT     DD   DSN=ALOG.DBDATA,DISP=OLD                  A-LOG Management Database (Data Part)
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(VERPARM),DISP=SHR  VSAM Parameter
//*******************************************************
//*    UNLOAD INDEX DB                                  *
//*******************************************************
//UNLDIX   EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//BKIX     DD   DSN=ALOG.BKIDEX,DISP=(,PASS),             Backup File
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(1,SL),
//              DCB=(RECFM=VB,BLKSIZE=19069)
//DBIX     DD   DSN=ALOG.DBIDEX,DISP=SHR                  A-LOG Management Database (Index Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(ULDIPARM),DISP=SHR VSAM Parameter
//*******************************************************
//*    UNLOAD DATA DB                                   *
//*******************************************************
//UNLDDT   EXEC PGM=&UTL,REGION=512K
//SYSPRINT DD   SYSOUT=&SOUT
//SYSUDUMP DD   SYSOUT=&UOUT
//BKDT     DD   DSN=ALOG.BKDATA,DISP=(,PASS),
//              UNIT=TAPE,VOL=SER=&BKVOL,LABEL=(2,SL),
//              DCB=(RECFM=VB,BLKSIZE=19069)             Backup File
//DBDT     DD   DSN=ALOG.DBDATA,DISP=SHR                  A-LOG Management Database (Data Part)
//SYSIN    DD   DSN=ALOG.V0380.PARMLIB(ULDDPARM),DISP=SHR VSAM Parameter
```

Appendix 3

Contents of Command Procedure

A3.1 Command Procedure for A-LOG Terminal Facility

```
PROC 0 MODE(C) OPT(ASPL(H.0.0)DCDUSER(ALOG)) K( )
IF (‘&MODE’ EQ ‘C’) THEN GOTO COLD
IF (‘&MODE’ EQ ‘W’) THEN GOTO WARM
COLD:+
 CONTROL NOMSG
 FREE  F(DDMAP DDSYSIN DDWTR DDLOGX DDLOGY DDKANJI DDSCT DDBKS DDJUTL)
 FREE  F(SECRTBL DBSNAP DD1 DDLOGB LOGPRINT SYSUDUMP DDTERM ASPOPTB)
 FREE  F(SECRLOGX SECRLOGY)
 ALLOC F(DDTERM)   DA(‘ALOG.V0380.PARMLIB(DCDTERMK)’) SHR
 ALLOC F(DDMAP)    DA(‘ALOG.V0380.MAPLIB’) SHR
 ALLOC F(DDSYSUDMP)SYSOUT(C)
 ALLOC F(DDSYSIN)  DA(‘ALOG.V0380.PARMLIB’) SHR
 ALLOC F(DDWTR)    DA(‘ALOG.V0380.PARMLIB’) SHR
 ALLOC F(DBSNAP)   SYSOUT(C)
 ALLOC F(ASPOPTB)  DA(‘ALOG.V0380.PARMLIB(ASPOPTB)’) SHR
 ALLOC F(SECRTBL)  DA(‘ALOG.V0380.SECRTBL’) SHR
 ALLOC F(SECRLOGX) DA(‘ALOG.V0380.SECRLOGX’) SHR
 ALLOC F(SECRLOGY) DA(‘ALOG.V0380.SECRLOGY’) SHR
 ALLOC F(DD1)      DA(‘ALOG.V0380.LOAD’) SHR
 ALLOC F(DDLOGB)   DUMMY
 ALLOC F(LOGPRINT) DUMMY
 ALLOC F(DDJUTL)   DUMMY
 CALL ‘ALOG.V0380.LOAD(USLPINIT)’ ‘&OPT’
    GOTO RETURN
WARM:+
 CONTROL NOMSG
 CALL ‘ALOG.V0380.LOAD(USLPINIT)’ ‘&OPT’
    GOTO RETURN
RETURN:+
 EXIT
```

Appendix 4

A-LOG Installation JCL Generator

A4.1 Functional Overview

JCLs for the installation of A-LOG system can be generated using A-LOG Installation JCL Generator. This installation JCL generator is included in A-LOG system installation tape. It uses the following Assembler macros to generate the installation JCLs:

Macro	Description
$ALOG	Macro for creating the installation JCLs for A-LOG system.
$COMMON	Macro called by $ALOG macro to create the following JCL statements: • JOB Statement • JOBLIB Statement • JOBCAT DD Statement • STEPLIB DD Statement • STEPCAT DD Statement • SORTLIB DD Statement
$ERROR	Macro called by $ALOG macro to display an error message.

Table A4.1: Installation JCL Generation Macros

The macros are stored in A-LOG Installation JCL Library (ALOG.V0380.INSTALL). A sample coding of $ALOG macro is given in member $INSTALL in A-LOG Installation JCL Library. Customize the function codes and parameters in this sample coding according to the specific installation and environment requirements. Function codes and parameters are explained in **A4.3 Function Codes & Parameters** of this appendix.

Submit $INSTALL for batch execution after customization, a set of JCLs suitable for installing A-LOG system will be generated. In some cases, these generated JCLs may still require some modifications. Hence, it is recommended that the generated JCLs be stored as a member in a partitioned dataset (PDS). This member may then be verified before submitting it for installing A-LOG system. A-LOG system installation processes are simplified by the use of A-LOG Installation JCL Generator.

A4.2 Processing Outline

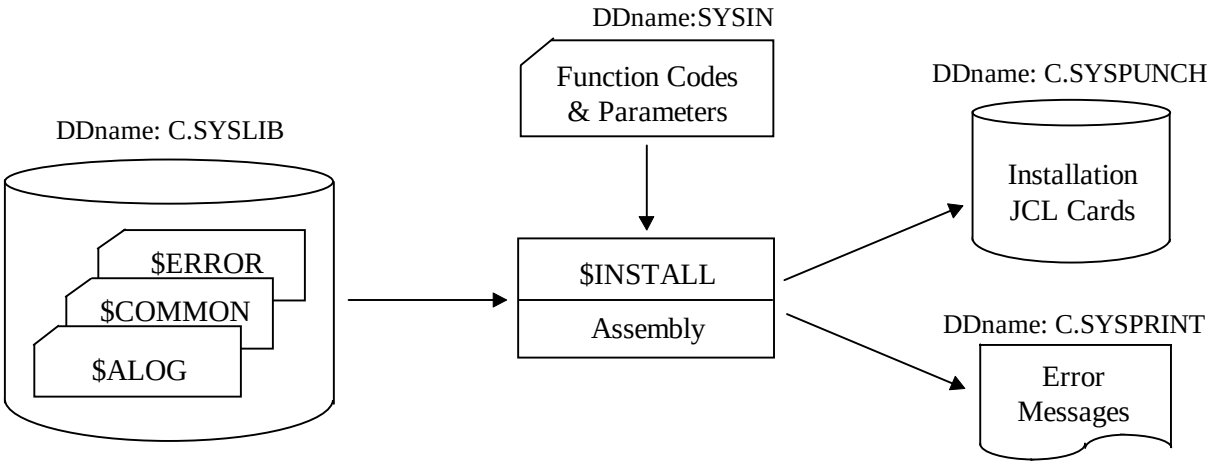


Fig. A4.1: Processing Outline of $INSTALL Assembly

The procedure for generation of A-LOG installation JCLs is as follows:

1) Modify member $INSTALL in A-LOG Installation JCL Library (ALOG.V0380. INSTALL) to suit the installation environment as follows:

```
//INSTALL    JOB
//ASM        EXEC procedure-name,PARM=DECK                    ✶
//C.SYSLIB   DD   DSN=ALOG.V0380.INSTALL,DISP=SHR             ✸
//C.SYSPRINT DD   SYSOUT=*
//C.SYSPUNCH DD   DSN=ALOG.V0380.INSTALL(member-name),        ✹
//                DISP=SHR,UNIT=SYSDA,
//                DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//SYSIN      DD   *                                           ❋
  sysin-cards
/*
//
```

✶ EXEC Statement

$ALOG, $COMMON and $ERROR are Assembler macros that have to be processed by the computer manufacturer's Assembler program. Specify the procedure for assembly these macro as follows:

- For IBM ⌫ ASMHC
- For Fujitsu ⌫ ASMFC
- For Hitachi ⌫ ASMC

- C.SYSLIB DD Statement

  Specify the JCL library containing the A-LOG Installation JCL Generator macros.

- C.SYSPUNCH Statement

  Specify the member name and library for storing the generated JCLs. The member name should be given for same name as the function code specified in $ALOG macro.

- SYSIN DD Statement

  Specify the function code and parameters in the SYSIN cards as shown as follows:

Column 1 10 16 72

```
//SYSIN  DD *
jobname  $ALOG  function-code,                                  C
                parameter,                                      C
                parameter,                                      C
                   :
                   :
                parameter,                                      C
                parameter
                END
```

Where: *jobname* ⌫ Specify the job name for the generated installation JCLs.

function-code ⌫ Specify the required function code. Refer to **A4.3 Function Codes & Parameters** further details.

parameter ⌫ Specify the required parameters. Refer to **A4.3 Function Codes & Parameters** for further details.

2) Submit member $INSTALL for assembly and generation of A-LOG installation JCLs.

A4.3 Function Codes & Parameters

The following table describes the functions codes supported by $ALOG macro:

Function Code	Description
INSTALL	To generate the entire JCLs required for installation of A-LOG system. Specifying this function code has the same effect as specifying all of the following function codes.
PARMLIB	To generate the JCLs for creating A-LOG Parameter Library.
PTF	To generate the JCLs for applying the necessary PTFs (Program Temporary Fixes) to support the target operating system.
ALLOCATE	To generate the JCLs for allocating and formatting A-LOG database.
INITVSM	To generate the JCLs for initializing A-LOG Management Database.
PROCLIB	To generate the JCLs for registering A-LOG cataloged procedures into SYS1.PROCLIB.
USLP	To generate the JCLs for registering a command procedure for A-LOG Terminal Facility into SYS1.CMDPROC.
SVCLNK	To generate the JCLs for registering A-LOG user SVC routine into SYS1.LPALIB.

Table A4.2: List of Function Codes

The following table describes the parameters supported by $ALOG macro:

Parameter	Description	Default	Applicable Function Code
VERSION	Specify the version number of A-LOG system to be installed.	V0380	All
ACCT	Specify the account field for JOB statement of installation JCLs.	A-ASPL-GEN	All
USER	Specify the programmer name field for JOB statement of installation JCLs.	A-ASPL	All
CLS	Specify the class field for JOB statement of installation JCLs.	A	All
MSGCLS	Specify the message class field for JOB statement of installation JCLs.	A	All
ANYJOB	Specify any parameter other than those mentioned above for the JOB statement of installation JCLs.	None	All

Table A4.3.a: List of Parameters

Parameter	Description	Default	Applicable Function Code
OS	Specify one of the following operating systems for A-LOG system installation environment: • VS1 • MVS • VOS2 • VOS3 • E30C • E40 • MSP	MVS	All
OSLEVEL	Specify one of the following operating system levels for A-LOG system installation environment: • VS1 • MVS/R38 • MVS/SE • MVS/XA • MVS/ESA • MVS/SP(MVS/SP13 or MVS/SP22) • E30 • E30C • E40 • E40C • MSP • MSP/E10 • MSP/E20 • VOS2 • VOS3/MA • VOS3/MB • VOS3/SP • VOS3/ES1 Note: For MVS/SP, specify 'MVS/SP13' for MVS/SP21 or earlier levels. Otherwise, specify 'MVS/SP22'.	MVS/SP	PTF
JES	Specify one of the following JES systems for A-LOG system installation environment: • JES2 • JES3 • JESE • JSS3 • JSS4 • JES2MAS • AFJES • JES	JES2	PTF
PREFIX	Specify the prefix for all dataset names of A-LOG database. A maximum of 8 characters can be specified. E.g. PREFIX=ALOG ⌫ ALOG.DB.BKS (A-LOG Data Pool)	ALOG	All
SUFFIX	Specify the suffix for all dataset names of A-LOG database. A maximum of 8 characters may be specified. E.g. SUFFIX=TEST ⌫ ALOG.DB.BKS.TEST (A-LOG Data Pool)	None	All
VOLSER	Specify the volume serial number of the disk volume for A-LOG database. A maximum of 2 disk volumes can be specified.	ASPL00	ALLOCATE PARMLIB
DEVTYPE	Specify one of the following device types for the assigned disk volume: • 3350 • 3380 • 3390	3380	ALLOCATE PARMLIB
DISKUNT	Specify the unit name of the assigned disk volume.	SYSDA	ALLOCATE PARMLIB
TAPEUNT	Specify the standard unit name of the tape drive.	TAPE	PROCLIB

Table A4.3.b: List of Parameters

Parameter	Description	Default	Applicable Function Code
WORKUNT	Specify the unit name of the work disk.	SYSDA	ALLOCATE PROCLIB SVCLNK JCLLNK SPMLNK RESLNK INITVSM PARMLIB
CATALOG	Specify the name of the OS catalog to be used for A-LOG database.	None	All
ASPLLIB	Specify the dataset name of A-LOG Load Module Library. E.g. ASPLLIB = ALOG.V0380.LOAD	ALOG.LOAD	All
SPCSCT	Specify the space, in cylinders, for the control part of A-LOG Data Pool (ALOG. DB.SCT).	2 cylinders	ALLOCATE PARMLIB
SPCBKS	Specify the space, in cylinders, for the data part of A-LOG Data Pool (ALOG.DB. BKS).	20 cylinders	ALLOCATE PARMLIB
SPCIDEX	Specify the space, in cylinders, for the index part of A-LOG Management Database (ALOG.DBIDEX).	10 cylinders	ALLOCATE PARMLIB
SPCDATA	Specify the space, in cylinders, for the data part of A-LOG Management Database (ALOG.DBDATA).	20 cylinders	ALLOCATE PARMLIB
MAXREP	Specify the maximum number of job logs which may be stored into A-LOG Data Pool.	2000	PARMLIB
MASTER	Specify the datasets to be created when the A-LOG database is allocated by Function Code = ‘ALLOCATE‘ or ‘INSTALL’. Value — Dataset Created ALL — All datasets of A-LOG database VSAMDB — ALOG.DBIDX and ALOG.DBDATA HOTLOG — ALOG.HOTLOG SECRTBL — ALOG.SECRTBL SCTBKS — ALOG.DB.SCT and ALOG.DB.BKS	ALL	ALLOCATE
PROCLIB	Specify the dataset name of the library for storing A-LOG cataloged procedures	SYS1.PROCLIB	PROCLIB
SORT	Specify the name of the SORT program.	SORT	PROCLIB

Table A4.3.c: List of Parameters

Parameter	Description	Default	Applicable Function Code
SORTLIB	Specify the dataset name of the library containing the SORT program.	SYS1.SORTLIB	PROCLIB
SORTSLB	Specify the dataset name of the library to be concatenated with STEPLIB when SORT is used.	None	PROCLIB PARMLIB
SOUT	Specify the SYSOUT class for output listings generated by A-LOG utilities.	A	PROCLIB USLP
UOUT	Specify the SYSOUT class for abend dump listings generated by A-LOG utilities.	B	PROCLIB
JLOG	Specify the SYSOUT class of job logs to be selected from OS spool by Job Log Storage Utility (ALOGJLOG). A maximum of 8 SYSOUT classes may be specified.	J	PROCLIB
SVCNO	Specify the user SVC number to be used by SVC routine of A-LOG system.	240	SVCLNK
LPALIB	Specify the dataset name of the library for storing the SVC routine of A-LOG system.	SYS1.LPALIB	SVCLNK
CMDNAME	Specify the member name to be given to the command procedure for A-LOG Terminal Facility. Note: If OS parameter is 'VS1' or 'VOS2', this specification is invalid since TSO/TSS is not available.	USLP	USLP
CMDPROC	Specify the dataset name of the library for storing the command procedure for A-LOG Terminal Facility.	SYS1.CMDPROC	USLP
FUNC19	This item is applicable to FACOM users only. Specify one of the following values: • YES ⌫ Use function FUNC19 of JES • NO ⌫ Do not use function FUNC19 of JES	NO	PTF
PTFNO	Specify the range of PTFs to be applied during A-LOG installation as follows: • PTFNO=(*xxx*,*yyy*) Where: *xxx* to *yyy* are the last 3 digits of the PTF numbers.	None	PTF

Table A4.3.d: List of Parameters

Parameter	Description	Default	Applicable Function Code
EXPAND	Certain PTFs require the affected modules to be expanded before they can be applied. Specify the PTFs to be expanded during A-LOG installation as follows: • EXPAND=*(nnn,mmm,...)* Where: *nnn* to *mmm* are the last 3 digits of the PTF numbers. Note: A maximum of 16 PTF numbers can be specified.	None	PTF
APPID	Specify the VTAM application name to be used for multi-CPU operations. E.g. APPID=(APPL0001,APPL0002) Notes: 1) Prefix must be a VTAM application name of the global CPU. 2) A maximum of 8 CPUs can be specified. Refer to MULTCPU parameter for further details.	None	PROCLIB PARMLIB
LOGID	Specify the SYSLOG ID to be used for multi-CPU operations. E.g. LOGID=(SYSLG1,SYSLG2) Notes: 1) Prefix must be a System ID of the global CPU. 2) A maximum of 8 CPUs can be specified. Refer to MULTCPU parameter for further details.	None	PROCLIB PARMLIB
SYSID	Specify the System ID to be used for multi-CPU operations. E.g. SYSID=(ALOG01,ALOG02) Notes: 1) Prefix must be a System ID of the global CPU. 2) A maximum of 8 CPUs can be specified. Refer to MULTCPU parameter for further details.	None	PROCLIB PARMLIB
MULTCPU	Specify whether or not A-LOG is to be operated with multiple CPUs as follows: • To operate in single CPU mode, specify MULTCPU=NO. In this case, parameters APPID, LOGID and SYSID will be ignored. • To operate in multi-CPU mode, specify SYSID of the A-LOG system to be installed. E.g. When installing global CPU MULTCPU=ALOG01 When installing local CPU MULTCPU=ALOG02 Note: Specify APPLID, LOGID and SYSID in the sequence as stated.	NO	PROCLIB PARMLIB

Table A4.3.e: List of Parameters

A JCL sample for assembly of $ALOG macro is given as follows:

```
//ALOGGEN  JOB CLASS=A,MSGCLASS=A
//*****************************************************************//
//*  A-LOG INSTALLATION JCL GENERATION MACRO                      *//
//*                                                               *//
//*        FUNCTION CODE & DESCRIPTION                            *//
//*                                                               *//
//*        INSTALL   -  GENERATE ALL INSTALLATION JCLS            *//
//*        PARMLIB   -  ESTABLISHING A-LOG PARAMETER LIBRARY      *//
//*        ALLOCATE  -  ALLOCATE AND FORMAT A-LOG FILES           *//
//*        INITVSM   -  INITIALIZE VSAM FILES                     *//
//*        PROCLIB   -  INSTALL A-LOG CATALOGED PROCEDURES        *//
//*        USLP      -  INSTALL A-LOG COMMAND PROCEDURE           *//
//*        PTF.      -  APPLY OS DEPENDENT PTFS TO A-LOG          *//
//*        SVCLNK    -  LINK EDIT A-LOG SVC ROUTINE TO LPALIB     *//
//*                                                               *//
//*  BSP INC.      1995.10.01                                     *//
//*****************************************************************//
//ASMHC      EXEC ASMHC,PARM=DECK    <-----FOR IBM OR FACOM
//*ASMFC     EXEC ASMFC,PARM=DECK    <-----FOR IBM OR FACOM
//*ASMC      EXEC ASMC,PARM=DECK     <-----FOR HITAC
//C.SYSLIB   DD   DSN=ALOG.V0380.INSTALL,DISP=SHR,
//               VOL=SER=ALOG01,UNIT=SYSDA
//C.SYSPUNCH DD   DSN=ALOG.V0380.INSTALL(ALLOCATE),DISP=SHR,
//               UNIT=SYSDA,VOL=SER=ALOG01,DCB=BLKSIZE=3120
//C.SYSPRINT DD   SYSOUT=*
//SYSIN      DD   *
ALOGALC  $ALOG ALLOCATE,               FUNCTION                          C
               JOBKEY=',CLASS=A,MSGCLASS=A', KEYWORD FOR JOB STATEMENT C
               ANYKEY='',              OTHER KEYWORD FOR JOB STATEMENT C
               OS=MVS,                 OS NAME              --- OS ---  C
               OSLEVEL=MVS/ESA,        OS LEVEL NAME                    C
               JES=JES2,               JES NAME                         C
               ASPLLIB=ALOG.LOAD,      A-LOG LOAD MODULE LIBRARY        C
               VERSION=V0380,          VERSION NUMBER FOR LIBRARY       C
               ZAPVER=V0380,           VERSION NUMBER FOR PTF NO        C
               PREFIX=ALOG,            PREFIX OF DATA SET NAME          C
               SUFFIX=,                SUFFIX OF DATA SET NAME          C
               APPID,                  A-LOG APPLICATION ID             C
               LOGID,                  A-LOG SYSLOG ID                  C
               SYSID,                  A-LOG SYSTEM ID                  C
               MULTCPU=NO,             MULTI CPU SUPPORT                C
               VOLSER=ALOGOO,          VOLSER FOR A-LOG(V1,V2,)         C
               DEVTYPE=3380,           DEVICE TYPE FOR INSTALLATION     C
               DISKUNT=SYSDA,          DISK UNIT FOPR INSTALLATION      C
               CATALOG=,               USER CATALOG FOR A-LOG FILES     C
               SPCIDEX=010,            VSAM FILE(INDEX) SPACE(CYL)      C
               SPCDATA=010,            VSAM FILE(DATA ) SPACE(CYL)      C
```

```
               MAXREP=1000,            MAX REPORT COUNT (BKSPARM)      C
               SPCSCT=004,             DATA POOL(INDEX) SPACE(CYL)     C
               SPCBKS=400,             DATA POOL(DATA ) SPACE(CYL)     C
               MASTER=ALL,             ALLOCATE ALL MASTER FILES?      C
               SVCNO=250,              SVC NUMBER TO BE USED           C
               LPALIB=SYS1.LPALIB,     LPALIB LIBRARY                  C
               PROCLIB=SYS1.PROCLIB,   A-LOG PROCEDURE LIBRARY   ----- C
               SORT=SORT,              PROGRAM NAME OF SORT PGM        C
               SORTLIB=SYS1.SORTLIB,   SORTLIB NAME                    C
               SORTSLB=,               SORT NEED STEPLIB,DSNAME OR ''  C
               TAPEUNIT=TAPE,          TAPE UNIT FOR A-LOG USE         C
               WORKUNT=SYSDA,          WORK DISK UNIT                  C
               SOUT=A,                 SYSOUT CLASS FOR SYSPRINT       C
               UOUT=B,                 SYSOUT CLASS FOR ABEND          C
               JLOGCLS=J,              ALOGJLOG CLASS (MAX 8 CHARS)    C
               SLOGCLS=J,              SYSLOG CLASS (MAX 8 CHARS)      C
               CMDPROC=SYS1.CMPROC,    COMMAND PROCEDULE LIBRARY       C
               CMDNAME=ALOG,           COMMAND PROCEDULE NAME          C
               EXPAND=,                A-LOG EXPAND PTF NO (MAX 16)    C
               PTFNO=,                 A-LOG PTF NO (FROM,TO)          C
               USRPTF=,                A-LOG USER PTF NO (MAX 16)      C
               CVTVER=(V034,V034),     CONVERT(FM-VER,TO-VER)    ----- C
               OLDLIB=ALOG.V0340.LOAD, OLD A-LOG LOAD LIBRALY          C
               OLDCAT=,                OLD A-LOG STEPCAT               C
               OLDSCT=ALOG.DB.SCT,     OLD A-LOG SCT FILE              C
               OLDBKS=ALOG.DB.BKS,     OLD A-LOG BKS FILE              C
               CVTVOL=(BUPVSM,BUPBKS)  VOL FOR BACK-UP VSAM,DATA POOL
         END
/*
//
```

A4.4 Error Messages

The following error messages may be displayed during assembly of $ALOG macro:

```
ERROR01  NO FUNCTION CODE SPECIFIED
```

Explanation: Function code is a mandatory field but it was not specified.

```
ERROR02  INVALID FUNCTION CODE
```

Explanation: The specified function code is incorrect.

```
ERROR03  INVALID OPERATING SYSTEM NAME OR OS LEVEL
```

Explanation: The value specified for OS or OSLEVEL parameter is incorrect.

```
ERROR04  INVALID USREXIT PARAMETER
```

Explanation: The SMF exit routine name given in USREXIT parameter is incorrect.

```
ERROR05  INVALID LENGTH OF AUTOPRC
```

Explanation: The value for AUTOPRC parameter must be specified in the following format:

AUTOPRC=(AUTO*nn*,AUTO*mm*,...)

where: 'AUTO' is a fixed value and *nn*, *mm*, etc. can be any two characters representing A-AUTO IDs.

```
ERROR06  PREFIX OF AUTOPRC MUST BE 'AUTO'
```

Explanation: The value for AUTOPRC parameter must be specified in the following format:

AUTOPRC=(AUTO*nn*,AUTO*mm*,...)

where: 'AUTO' is a fixed value and *nn*, *mm*, etc. can be any two characters representing A-AUTO IDs.

```
ERROR07  INVALID JCLMSTR PARAMETER
```

Explanation: The value given in JCLMSTR parameter is incorrectly.

```
ERROR08  ASPLLIB PARAMETER MISSING
```

Explanation: ASPLLIB parameter is mandatory but it was not specified.

```
ERROR09  FAIR PARAMETER MUST BE 'OLD' OR 'NEW'
```

Explanation: The value for FAIR parameter must be either ‘OLD’ or ‘NEW’.

```
ERROR10  INVALID NAME OF JOB ENTRY SUB-SYSTEM
```

Explanation: The value specified for JES parameter is incorrect.

```
ERROR11  AUTOLIB PARAMETER MISSING
```

Explanation: AUTOLIB parameter is mandatory but it was not specified.

```
ERROR12  UNKNOWN DEVTYPE PARAMETER
```

Explanation: The value specified for DEVTYPE parameter is incorrect.

```
ERROR13  INVALID MASTER ID(S)
```

Explanation: The value specified for MASTER parameter is incorrect.

```
ERROR14  THE VALUE OF 'BASEID' SHOULD BE 'YES'
```

Explanation: ‘AUTOBASE’ was specified for MASTER parameter but ‘NO’ was specified in the BASEID parameter.

```
ERROR15  THE VALUE OF 'PARM' SHOULD BE 'YES'
```

Explanation: ‘SPLTBL’ or ‘MPLTBL’ was specified for MASTER parameter but ‘NO’ was specified for PARM parameter.

```
ERROR16  THE VALUE OF 'SECJCL' SHOULD BE 'YES'
```

Explanation: ‘SECNDJCL’ was specified for MASTER parameter but ‘NO’ was specified for SECJCL parameter.

```
ERROR17  A-AUTO PROCESSOR NAME INVALID
```

Explanation: The value for AUTOPRC parameter must be specified in the following format:

AUTOPRC=(AUTO*nn*,AUTO*mm*,...)

where: 'AUTO' is a fixed value and *nn*, *mm*, etc. can be any two characters representing A-AUTO IDs.

```
ERROR18  VOLSER PARAMETER EXCEEDED BY '2'
```

Explanation: More than 2 volume serial numbers were specified for VOLSER parameter.

```
ERROR19  THE VALUE OF 'JERRLOG' SHOULD BE 'YES'
```

Explanation: 'JSS4' was specified for JES parameter but 'NO' was specified for JERRLOG parameter.

```
ERROR20  INVALID SYSPROC PARAMETER
```

Explanation: The value for SYSPROC parameter must be specified in the following format:

SYSPROC=(*value1*,*value2*,...)

```
ERROR21  SYSPROC AND AUTOPRC UNMATCH
```

Explanation: The number of values coded for SYSPROC parameter must be same as the number of values specified in AUTOPRC parameter.

```
ERROR22  UNLDUNT PARAMETER MUST BE 'TAPE' OR 'DISK'
```

Explanation: The value for UNLDUNT parameter must be either 'TAPE' or 'DISK'.

```
ERROR23  JOBNAME PARAMETER MISSING
```

Explanation: JOBNAME parameter is mandatory but it was not specified.

```
ERROR24  INVALID SUBFUNC CODE
```

Explanation: An unknown parameter has been coded.

```
ERROR25  DSN NO. OF ASPLLIB PARAMETER EXCEEDED '16'
```

Explanation: More than 16 datasets were specified for ASPLLIB parameter.

```
ERROR26  DSN NO. OF CATALOG PARAMETER EXCEEDED '16'
```

Explanation: More than 16 datasets were specified for CATALOG parameter.

```
ERROR27  DSN NO. OF SORTLIB PARAMETER EXCEEDED '16'
```

Explanation: More than 16 datasets were specified for SORTLIB parameter.

```
ERROR28  INVALID SVC NO.
```

Explanation: The value specified for SVCNO parameter is incorrect.

```
ERROR29  INVALID PARM PARAMETER
```

Explanation: The value or format specified for a parameter is incorrect.

```
ERROR30  INVALID PANJCL PARAMETER
```

Explanation: The value or format specified for PANJCL parameter is incorrect.

```
ERROR31  INVALID TEMPJCL PARAMETER
```

Explanation: The value or format specified for TEMPJCL parameter is incorrect.

```
ERROR32  INVALID PTFNO PARAMETER
```

Explanation: The value or format specified for PTFNO parameter is incorrect.

```
ERROR33  EXPAND PTF NO IN EXPAND PARAMETER EXCEEDED '16'
```

Explanation: More than 16 PTFs were specified for EXPAND parameter.

How about the following messages ?

```
ERROR34  INVALID JES PARAMETER
```

Explanation: An invalid value has been specified for JES parameter.

```
ERROR35  INVALID CVTVER PARAMETER
```

Explanation: An invalid value has been specified for CVTVER parameter.

```
ERROR36  INVALID CVTVOL PARAMETER
```

Explanation: An invalid value has been specified for CVTVOL parameter.

```
ERROR37  INVALID ZAPVER PARAMETER
```

Explanation: An invalid value has been specified for ZAPVER parameter.

```
ERROR38  INVALID APPID OR SYSID OR LOGID PARAMETER
```

Explanation: The number of values specified for APPID, SYSID and LOGID must be the same.

```
ERROR39  MULTCPU NOT EXIST IN SYSID
```

Explanation: The value specified for MULTCPU parameter does not exist in SYSID parameter.

```
ERROR40  PTF NO OF USRPTF PARAMETER EXCEEDED '16'
```

Explanation: More than 16 values have been specified for USRPTF parameter.

```
ERROR41  INVALID USRPTF PARAMETER
```

Explanation: An invalid value has been specified for USRPTF parameter.

```
ERROR42  ENQNAME MUST BE USMSV + 3 NUMERIC CHARACTERS
```

Explanation: An invalid value was specified for ENQNAME parameter. The value for ENQNAME parameter must be specified as 'USMSV*nnn*' (where *nnn* is any number).

Appendix 5

Non-Compatible Items (V3.6 → V3.8)

Reference

A5.1 Changes Made to A-LOG System Datasets

◆ Deletion of A-LOG Datasets

The following datasets have been deleted from A-LOG system:

Dataset Name	File Organization	File Name
ALOG. DB.LOGX	DAM	Printer Log File (X)
ALOG.DB.LOGY	DAM	Printer Log File (Y)

◆ Deletion of Files in A-LOG Management Database

The following files in A-LOG Management Database, which are used by A-SPOOL system, have been deleted:

File Name	File Number
Report Distribution Master (RDM)	30
Report Schedule Master (RSM)	40
Report Index (BIS)	50
Distribution Index (BDS)	70
Password File (PAS)	90
Distribution Name Master (DST)	100
Report Attribute Master (RAM)	120
Report LBP Information Master (LBP)	130

A5.2 Changes Made to A-LOG Cataloged Procedures

◆ Changes Made to Existing A-LOG Cataloged Procedures

Procedure Name	Changes Made
ALOG	• Addition of DDLOGX and DDLOGY DD statements.
ALOGDATE	• Addition of USMSOUT DD statement. • Modification of EXEC parameter (PDATE=).
ALOGBLOG	• Modification of EXEC parameter (PDATE=).
ALOGLLOG	• Modification of EXEC parameter (PDATE=).
ALOGJDMP	• Modification of EXEC parameter (PDATE=).
ALOGLDMP	• Modification of EXEC parameter (PDATE=).
ALOGDLT4	• Modification of EXEC parameter (PDATE=,PRINT={Y|N}).
ALOGDLT5	• Modification of EXEC parameter (PDATE=).
ALOGJRST	• Addition of USMSCARD DD statement.
ALOGLRST	• Addition of USMSCARD DD statement.
ALOGREP	• Addition of EXEC parameter (TYPE=ALOG).
ALOGUBKS	• Addition/Modification of EXEC parameter (TYPE=ALOG,SIZE=200K).
ALOGIBKS	• Addition of EXEC parameter (TYPE=ALOG).
ALOGLBKS	• Addition/Modification of EXEC parameter (TYPE=ALOG,SIZE=200K).

◆ New A-LOG Cataloged Procedures

Procedure Name	Description
ALOGDMP1	To generate 3 output files from A-LOG Management Database.
ALOGDMP2	To dump job logs and delete the dumped job logs.
ALOGDMP3	To delete job log index and data which have expired.
ALOGDMP4	To delete the 3 files generated by ALOGDMP1 utility.
ALOGRSEL	To restore the job log or SYSLOG which have been marked for restoration.

A5.3 Changes Made to Command Procedure

- Addition of CALL statement parameter
- Deletion of ALLOC statement for DDSCT and DDBKS (changed into dynamic allocation)

A5.4 Changes Made to A-LOG Parameters

◆ Changes Made to Existing Parameters

Procedure	Member	Changes Made
ALOG	ASPPRM1	• Addition of MAXLC, MAXLL and BUFFL keywords.
	RUNSPM2	• Modification of DBM keyword (DBM=ALGVBAS).
	WANDMM	• Modification of date item from 6 to 8 digits.
	WANJAM	• Modification of date item from 6 to 8 digits.
	WANJBI	• Modification of date item from 6 to 8 digits. • Addition of Restore Flag, Restore Status, Restore Date, Restore Data Retention Days, Restore Index Retention Days items.
	WANSYS	• Modification of date item from 6 to 8 digits.
ALOGBLOG	RUNLOG5	• Modification of sort file.
ALOGLLOG		
ALOGIBKS	SCTINIT	• Modification of program name (PGM=BKSINITL).
ALOGREP	SCTREP	• Modification of program name (PGM=BKSREPL).
A-LOG Terminal Facility	ASPOPTB	• Addition of NJWTR, NJNAM, NJRJOB, NJBKSD, NJIDXD, NSRJOB, NSBKSD, NSIDXD, TIMEC, TCOM, JNOC, JNOL, JCOM and JCOML keywords.
INTVSM	FRMPDMM	• Modification of date item from 6 to 8 digits.
	FRMPJAM	• Modification of date item from 6 to 8 digits.
	FRMPJBI	• Modification of date item from 6 to 8 digits. • Addition of Restore Flag, Restore Status, Restore Date, Restore Data Retention Days, Restore Index Retention Days items.
	FRMPSYS	• Modification of date item from 6 to 8 digits.
WTRJUTL	DDJPARM	• Addition of UNIT keyword.
WTRSUTL	DDSPARM	• Addition of UNIT, TIMEC, TCOM, JNOC, JNOL, JCOM and JCOML keywords.

◆ New Parameters

Procedure	Member	Description
ALOGDMP1	ULDDMP	Define VSAM parameters for generating A-LOG Management Database.
ALOGDMP2	RUNDMP1	Contain sort control cards.
	RUNDMP2	Contain sort control cards.
ALOGDMP3	RUNDMP3	Contain sort control cards.
	RUNDMP4	Contain sort control cards.
ALOGJRST	USMSJRST	Define restore control parameters.
ALOGLRST	USMSLRST	Define restore control parameters.
ALOGRSEL	RSELPARM	Define control cards for restore job generation.
Command Procedure	DCDALCTB	Define dataset names of A-LOG Data Pool for A-LOG Terminal Facility

A5.5 Changes Made to A-LOG Load Modules

◆ The load module names shared by A-SPOOL system and A-LOG system have been changed to indicate the original name of A-LOG. The load module names which have been changed are shown as follows:

Procedure Name	Old Module Name	New Module Name	Description
ABSIDLT	ABSIDLT	ABSIDLTL	It deletes files from A-LOG Management Database.
ABSILOAD	ABSILOAD	ABSILODL	It restores files into A-LOG Management Database.
ALLOCATE	ABSIPRE	ABSIPREL	It formats A-LOG Management Database.
ALOG	SPM	SPL	A-LOG Monitor. It manages A-LOG Management Database and A-LOG Data Pool. It receives and replies to commands from other components and operators.
	ASPVBAS	ALGVBAS	
	ASPCTBL	ASPCTBLL	
ALOGCPY1	CPY100	CPY100L	It backup A-LOG Management Database by file.
ALOGCPY2	CPY200	CPY200L	It restores A-LOG Management Database by file.
ALOGDATE	DAT01	DAT01L	It sets the Operation Date for A-LOG system.
ALOGDLT1	DLT01	DLT01J	It deletes records in A-LOG Data Pool.
ALOGDLT4	DLT04	DLT04L	It deletes records in Dump MT Information File.
ALOGDMPC	DMPCPY	DMPCPYL	It copies a Dump MT into another magnetic tape.
ALOGIBKS	SCTRUN	SCTRUNL	It initializes A-LOG Data Pool.
	BKSINIT	BKSINITL	
ALOGICHK	ABSICNV	ABSICNVL	It checks for unmatched records in A-LOG Management Database.
ALOGJDMP	DUMP	DUMPL	It archives job logs into a Dump MT.
	DLT01	DLT01L	
ALOGJRST	RESTORE	RESTOREL	It restores job logs from a Dump MT.
ALOGLBKS	BKSLDE	BKSLDEL	It restores the A-LOG Data Pool.
ALOGLDMP	DLT01	DLT01L	It archives SYSLOG into a Dump MT.
ALOGREP	ABSIREP	ABSIREPL	It generates A-LOG Management Database and A-LOG Data Pool space utilization reports.
	SCTRUN	SCTRUNL	
	BKSREP	BKSREPL	
ALOGUBKS	BKSUDE	BKSUDEL	It unloads the A-LOG Data Pool.

Procedure Name	Old Module Name	New Module Name	Description
Command Procedure for A-LOG Terminal Facility	ASPA000L	ALGA000L	It displays [1.0.0 Master Management] screen.
	ASPB300L	ALGB300L	It displays [2.2.0 Dump MT Inquiry] screen.
	DCDX000L	ALGX000L	It displays [9.0.0 System Management] screen.
	DCDX100L	ALGX100L	It displays [9.1.0 Terminal/PF-Keys Setting] screen.
—	CHKSPMA	SHKSPLA	It checks whether A-LOG Monitor is active.
—	CHKSPML	CHKSPLL	It checks whether A-LOG Monitor is inactive.
—	MSGBKS	MSGBKSL	Message definition.
—	MSGCAT	MSGCATL	Message definition.
—	MSGDUMP	MSGDUMPL	Message definition.
—	MSGRSTR	MSGRSTRL	Message definition.
—	MSGSPM	MSGSPML	Message definition.
—	MSGSUB	MSGSUBL	Message definition.
—	MSGUSLP	MSGUSLPL	Message definition.
—	MSGUTL	MSGUTLL	Message definition.
—	MSGVBAS	MSGVBASL	Message definition.
—	MSGWTR	MSGWTRL	Message definition.
—	USMSLINK	USMSLLNK	User access routine.

A5.6 Changes Made to A-LOG Terminal Facility

◆ Addition of Terminal Screens

- [2.1.1.2 JOBLOG DIRECTORY (INQ/UPD)] screen, the “right” part of 2 inter-related screens.
- [2.1.2.2 JOBLOG INDEX-NO.1.- (INQ/UPD)] screen, the “bottom” part of 2 inter-related screens.
- [3.2.2.2 SYSLOG INDEX-NO.1.- (INQ/UPD)] screen, the “bottom” part of 2 inter-related screens.
- [A-SPOOL/A-LOG SELECT] screen.

◆ Addition of Input/Output Items in Terminal Screens

- [2.1.0 JOBLOG INDEX MAINTENANCE] screen.
- [2.1.1.1 JOBLOG DIRECTORY (INQ/UPD)] screen, the “left” part of 2 inter-related screens.
- [2.1.2.1 JOBLOG INDEX-NO.1.- (INQ/UPD)] screen, the “top” part of 2 inter-related screens.
- [3.2..0 SYSLOG INDEX MAINTENANCE] screen.
- [3.2.1 JOBLOG DIRECTORY (INQ/UPD)] screen.
- [3.2.2.1 SYSLOG INDEX-NO.1.- (INQ/UPD)] screen, the “top” part of 2 inter-related screens.

◆ Job Log or SYSLOG Restore Function from Terminal Screens

A5.7 Changes Made to A-LOG SVC Routine

- The new SVC routine used by A-LOG V3.8 and above supports Database ID and MINOR ENQ name. Therefore, two or more A-LOG and/or A-SPOOL Monitors can share one SVC. For example, A-LOG and A-SPOOL can share the same SVC.

Appendix 6

A-LOG Version Up Procedure (V3.6 → V3.8)

A6.1 Version Up Procedure (V3.6 ⌫ V3.8)

Step	Description	Comment
1	Backup DASD	
2	Start A-LOG Monitor (old version)	S ALOG
3	Backup A-LOG Management Database using CPY100	FILE=SYS/JAM/JBI/DMM
4	Stop A-LOG Monitor	F ALOG,CLOSE
5	Unload A-LOG Data Pool (increase SCT block size)	
6	Backup A-LOG system libraries	Dataset Name = LOAD / MAPLIB / PARMLIB / PROCLIB / CMDPROC
7	Install A-LOG V3.8 system libraries	Dataset Name = LOAD / MAPLIB / PARMLIB / ZAP / USERLOAD
8	Decide PARMLIB (customize user environment)	
9	Create and initialized A-LOG Management Database (VSAM)	Dataset Name = DBIDEX / DBDATA
10	Create control files (DAM)	Dataset Name = HOTLOG / SECRTBL / SECRLOGX / SECRLOGY
11	Initialize A-LOG Data Pool (increase SCT block size)	
12	Load A-LOG Data Pool (increase SCT block size)	
13	Master and index conversion	FILE=SYS,JAM,JBI,DMM
14	Load A-LOG Management Database	FILE=SYS,JAM,JBI,DMM
15	Create Catalog Procedure	
16	Create Command Procedure	
17	Customize load modules	MEM=B38I*nnn* (IBM) / B38F*nnn* (Fujitsu) / B38H*nnn* (Hitachi)
18	Create User SVC Routine	Dataset Name = SYS1/LPALIB MEM=IGC00*nnn* / JFF00*nnn*
19	Re-IPL Operating System with CLPA	
20	Start A-LOG Monitor (New Version)	S ALOG S ALOGJLOG
21	Verify Conversion	

Appendix 7

PTFs for Operating Systems

A7.1 List of PTFs for Various Operating Systems

Manufacturer	Operating Environment	PTF Member	Description
IBM	IBM operating system supported by A-LOG system	B380I001	Set IBM OS flag within A-LOG system
	MVS/XA*	B380I002	CSA GETMAIN support for MVS/XA
	VTAM V3.1	B380I003	ACB length modification for VTAM V3.1
Fujitsu	Fujitsu operating system supported by A-LOG system	B380F001	Set Fujitsu OS flag within A-LOG system
Hitachi	Hitachi operating system supported by A-LOG system	B830H001	Set Hitachi OS flag within A-LOG system
	VOS3	B380H002	Support ESTAE macro of VOS3

* This PTF cannot be applied when using MVS/ESA.

Appendix 8

A-LOG System Recovery Procedure

A8.1 Suggestions

This appendix explains the recovery procedure for A-LOG system when an abnormal situation, such as “CPU down”, “operating system down”, “abnormal termination of A-LOG Monitor”, etc., has occurred during operation of A-LOG system.

◆ Mandatory Recovery Actions

When a “system down”, “CANCEL command”, “A-LOG Monitor user abend” or abnormal termination of A-LOG Monitor occurs, the first task is to perform initiation A-LOG Monitor, then stop A-LOG Monitor. By performing initiation and stopping of A-LOG Monitor, recovery processing of A-LOG database can be executed from the recovery log record. By executing recovery processing, the occurrence of mismatched records as a result of the abnormal termination can be prevented.

◆ Recovery Procedure

The recovery procedure related to the following problems are explained.

✶ A-LOG Monitor did not end abnormally, however the following serious error code was returned:

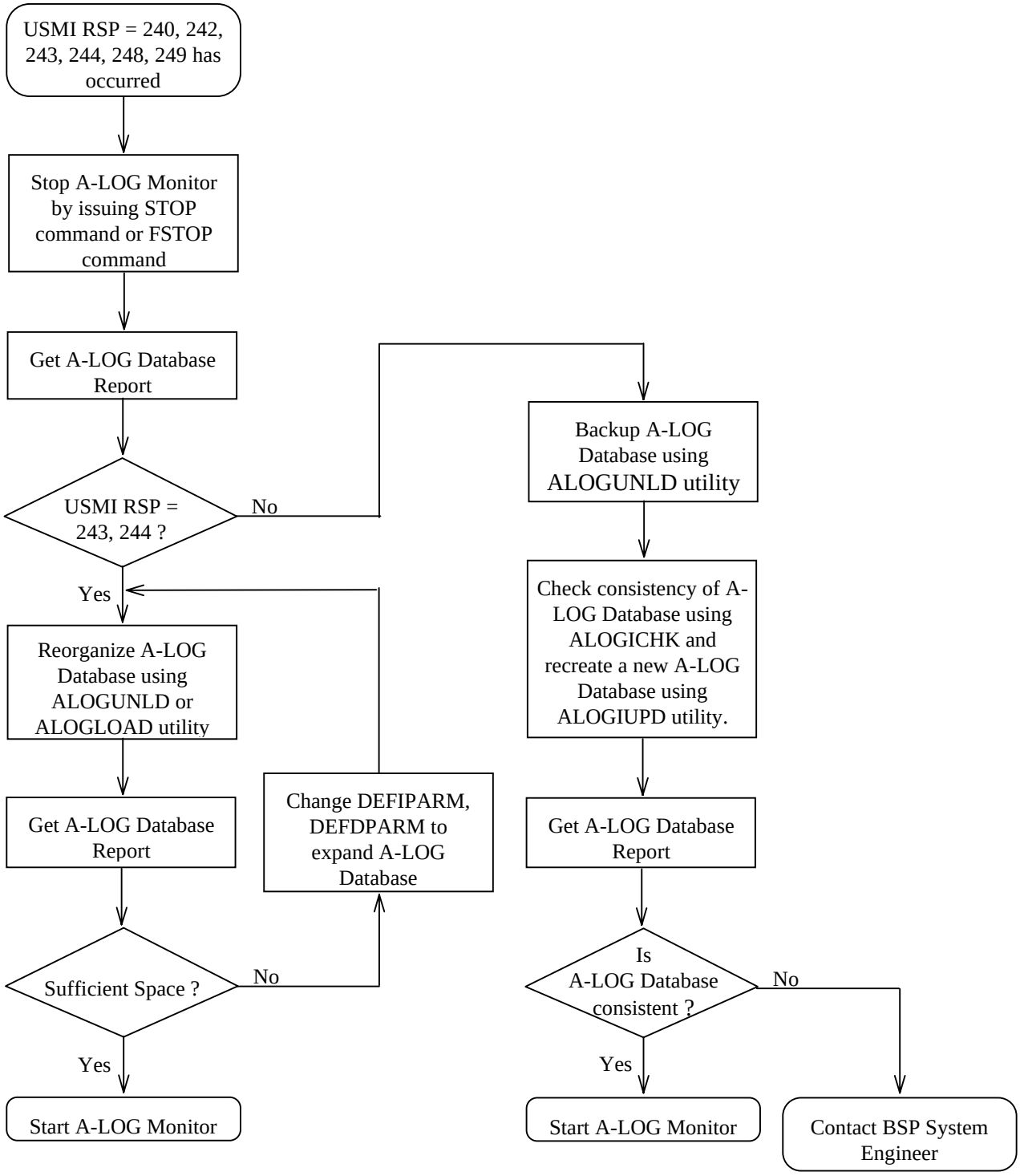
- USMI Response Code = 243, 244 (database out of space)
- USMI Response Code = 240, 242 (database I/O error)
- USMI Response Code = 248, 249 (database I/O error)

✷ A-LOG Monitor ended abnormally:

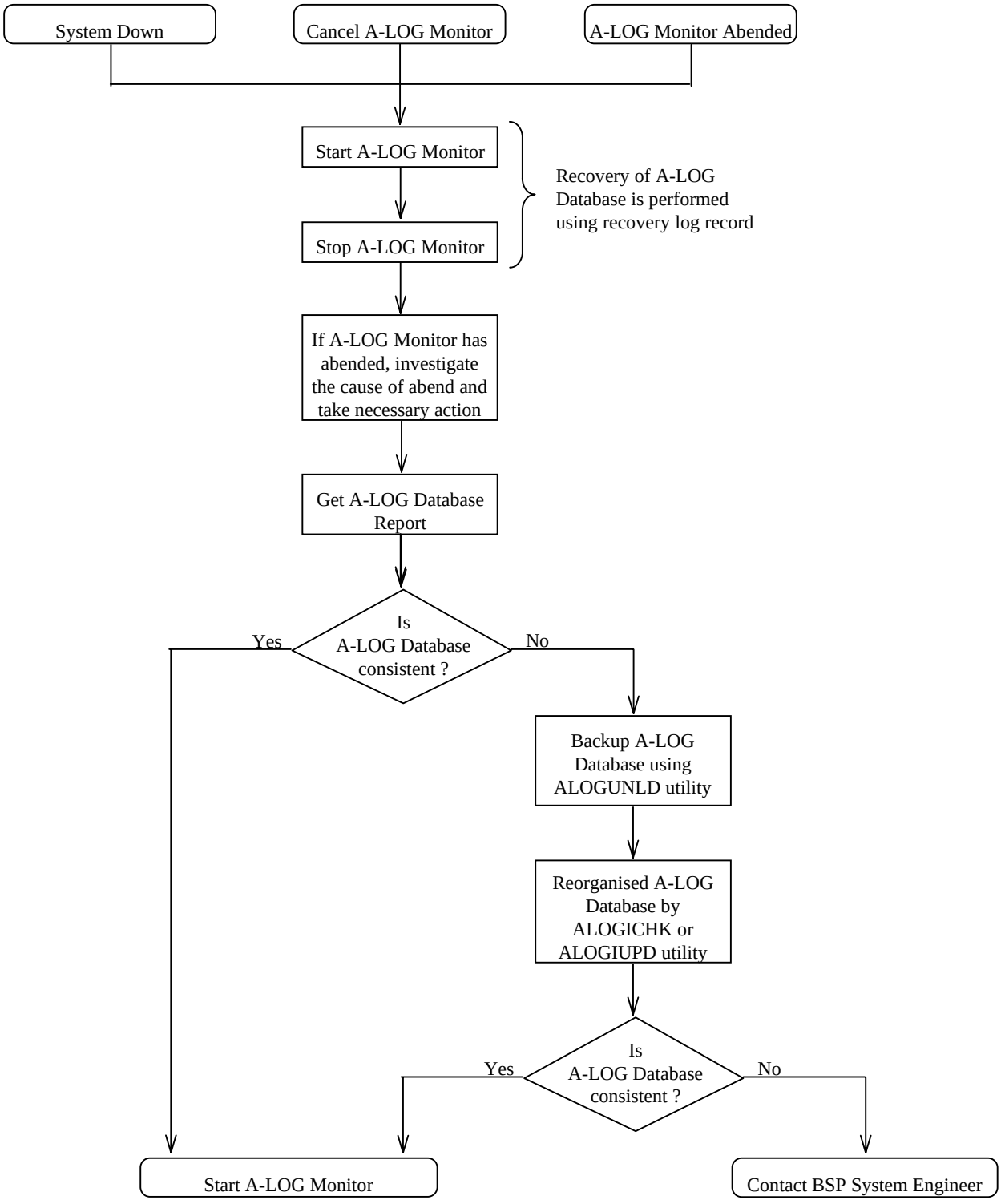
- A system down has occurred
- A-LOG Monitor was canceled
- A-LOG Monitor has abended with a User Abend U0777

✹ A-LOG Monitor has abended with a User Abend U0034 (at the time of displaying message SIO006E)

Case ✶ A-LOG Monitor did not end abnormally, however serious error code was returned.



Case ✸ A-LOG Monitor ended abnormally.



Case ✹ A-LOG Monitor has abended with a User Abend U0034 (at the time of displaying message SIO006E)

For A-LOG Monitor, the maximum degree of concurrent access to A-LOG database is 100 users. If the degree of concurrent access exceeds 100 users (i.e. A-LOG Terminal Facility is used from more than 100 terminals), the degree of concurrent access must be increased by a user customization PTF.

In addition to the above, when a value less or equal to 100 is specified in ‘NCQE’ and 'NUSQ' keywords of RUNSPM2 parameter, it outputs this message. In this case, U0034 abend will occurs although the degree of concurrent access is not over 100 users. A dump list is required for analysis of the problem

The parameter relevant to the degree of concurrent access:

1) RUNSPM2
   - NUSQ ⌫ Number of concurrent user access
   - NCQE ⌫ Number of concurrent commands
   - NMAQ ⌫ Number of concurrent A-LOG Terminal Facility (at the time of multi CPU use under a global machine)

2) ABSIPRM1
   - ACCESS ⌫ Number of concurrent access
   - SETRC ⌫ Number of concurrent access