

***OS/32 INTERNALS
STUDENT GUIDE***

OS/32 INTERNALS

PERKIN-ELMER

Technical Training
61 Davis Avenue
Neptune, NJ 07753

Copyright © 1985 by The Perkin-Elmer Corporation
All Rights Reserved
Printed in the USA

INTRODUCTION

The Perkin-Elmer Technical Training center has prepared this student guide to supplement what you will learn in the OS/32 Internals course.

You can use the protruding tabs in the student guide to find the section you are looking for. The tabs for the three main sections are printed in red. Each of the main sections has a content page. The Internals flowcharts are presented in sections denoted by black-printed tabs. There is a section in which you can insert lab exercises and another for your notes.

We hope you find this student guide informative and easy to use, and we invite your critique of its content and presentation.

OS/32 INTERNALS
TABLE OF CONTENTS

MNEMONIC LIST

v

SECTION 1 SYSTEM MODULES

Auxiliary Processor Services (APSV)	1-1
Console Driver (CDVR)	1-3
Bulk Device Command Executors (CMDB)	1-4
Command Executors (CMEX)	1-5
Internal Reader (CMIR)	1-7
Console Monitor (CMON)	1-8
Command Processor (CMSP)	1-9
Panic Dump Support (DUMP)	1-11
Error Recording (ERRC)	1-12
Accounting Support (EXAC)	1-14
Executive Interrupts (EXIN)	1-15
Executive I/O Services (EXIO)	1-16
Executive Loader (EXLD)	1-21
Roll Manager (EXMR)	1-22
Memory Manager (EXMY)	1-23
Supervisory Services (EXSP)	1-25
Executive Services (EXSV)	1-26
Time Management (EXTI)	1-27
Task Manager (EXTM)	1-28
Software Floating Point Emulation (FLTP)	1-30
File Manager Services Part A (FMA7)	1-31
File Manager Services Part B (FMB7)	1-32
Contiguous File Manager (FMCO)	1-33
Index File Manager (FMIN)	1-34
File Manager (FMS7)	1-35
File Manager Utilities (FMUT)	1-36
SVC Intercept (INTC)	1-37
Communications Support (ITAM)	1-38
Communications File Manager (ITFM)	1-41
Memory Checks (MCHK)	1-43
Mirror Dummy (MIRR)	1-44
Supervisor Call Zero (SVC0)	1-45
Operating System End (UBOT)	1-46

SECTION 2 SYSTEM STRUCTURES

Access Control Block (ACB)	2-1
Auxiliary Processor Block (APB)	2-3
Channel Control Block (CCB)	2-7
Context Block (CTX)	2-8
Device Dependent Segment (DDCB)	2-9
File Directory (DIR)	2-13
Device Mnemonic Table (DMT)	2-14
Event Coordination Node (EVN)	2-15

TABLE OF CONTENTS (Continued)

File Control Block (FCB)	2-17
File Descriptor (FD)	2-22
Free Block Descriptor (FDE)	2-23
Index File Manager Extended SVC 1	2-24
Intercept Control Block (ICB)	2-25
Input/Output Block (IOB)	2-27
I/O Handler Routine (IOH)	2-29
IOP Processor Block (IPB)	2-31
Intercept PCB (IPCB)	2-39
Intercept Read Control Block (IRCB)	2-41
Initial Value Table (IVT)	2-42
Loader Common/Entry (LCE)	2-43
Loader Information Block (LIB)	2-44
Logical Processor Mapping Table (LPMT)	2-46
Loader Sharable Segment Entry (LSE)	2-47
Loadseg Parameter Block (LSG)	2-48
Loader Task's TCB Redefinitions (LTCB)	2-49
Overlay Control Block (OCB)	2-51
Overlay Descriptor Table (ODT)	2-52
ORT Header (ORH)	2-53
Overlay Reference Table (ORT)	2-54
Device Independent Segment (PDCB)	2-55
Private File Control Block (PFCB)	2-58
Pseudo Device Control Block (PSDCB)	2-61
Pseudo Task Control Block (PSTCB)	2-62
Queue Control Block (QCB)	2-63
Queue Control Block Header (QCH)	2-65
Queue Handler (QH)	2-66
Queue Parameter Block (QPB)	2-67
RS/RSA Context Block (RCTX)	2-68
Roll Selection List (RLST)	2-69
Segment Descriptor (SDE)	2-70
Secondary Directory (SECDIR)	2-71
System Pointer Table (SPT)	2-73
Segment Table Entry (STE)	2-82
System Value Table (SVT)	2-83
System Space (SYP)	2-84
Task Control Block (TCB)	2-85
Accounting Data Extension (TIO)	2-94
Task Queue-Head (TKQ)	2-95
Timer Queue Entry (TMQ)	2-96
Task Event Queue Entry (TQE)	2-97
Task Event Queue Header (TQH)	2-98
Task Trap Block (TTB)	2-99
Volume Descriptor (VD)	2-100
Volume Mnemonic Table (VMT)	2-101

TABLE OF CONTENTS (Continued)

SECTION 3 FLOW DIAGRAMS

Initialization	3A-1
Console Support	3B-1
Internal Reader	3C-1
Resident Loader	3D-1
Crash Handler	3E-1
Time Functions	3F-1
Task Manager	3G-1
I/O Subsystem	3H-1
File Manager	3I-1
File Management Utilities	3J-1
SVC Intercept	3K-1
End of Task	3L-1
Interrupt Handlers	3M-1
System States	3N-1
SVC0	3O-1

LAB EXERCISES

APPENDIX

System Option Bits	A-1
Task Option Bits	A-4
Task Wait Bits	A-7
Task Status Bits	A-9
Device Control Block Flag Bits	A-12
Input/Output Block Flag Bits	A-14
File Control Block Flag Bits	A-16

NOTES

MEANING OF MNEMONICS USED
IN THIS STUDENT GUIDE

ABBREVIATION	MEANING
APU	AUXILIARY PROCESSOR UNIT
CC	CONDITION CODE
CHAR	CHARACTER
COMM	COMMUNICATIONS
CR	CARRIAGE RETURN
DCB	DEVICE CONTROL BLOCK
DIR	DIRECTORY
DSS	DYNAMIC SYSTEM SPACE
D-TASK	DIAGNOSTIC TASK
E-TASK	EXECUTIVE TASK
FC	FUNCTION CODE
FD	FILE DESCRIPTOR
IOB	INPUT/OUTPUT BLOCK
IOP	I/O PROCESSOR
LIBS	LIBRARIES
LOC	LOCATION COUNTER
LU	LOGICAL UNIT
OPTS	OPTIONS
PCB	PARAMETER CONTROL BLOCK
PTR	POINTER
REG	REGISTER
SDE	SEGMENT DESCRIPTOR ENTRY
SEG	SEGMENT
SVC	SUPERIVSOR CALL
PARAM BLK	PARAMETER BLOCK

MNEMONICS (Continued)

ABBREVIATION	MEANING
PSW	PROGRAM STATUS WORD
TCB	TASK CONTROL BLOCK
TGD	TRAP GENERATING DEVICE
TSW	TASK STATUS WORD
UD	USER DEDICATED
UDL	USER DEDICATED LOCATION
U-TASK	USER TASK
VD	VOLUME DESCRIPTOR
VOL	VOLUME
VFC	VERTICAL FORMS CONTROL

***OS/32 INTERNALS
SYSTEM MODULES***

**S
E
C
T
I
O
N

1**

TABLE OF CONTENTS

MODULES

Auxiliary Processor Services (APSV)	1-1
Console Driver (CDVR)	1-3
Bulk Device Command Executors (CMDB)	1-4
Command Executors (CMEX)	1-5
Internal Reader (CMIR)	1-7
Console Monitor (CMON)	1-8
Command Processor (CMSP)	1-9
Panic Dump Support (DUMP)	1-11
Error Recording (ERRC)	1-12
Accounting Support (EXAC)	1-14
Executive Interrupts (EXIN)	1-15
Executive I/O Services (EXIO)	1-16
Executive Loader (EXLD)	1-21
Roll Manager (EXMR)	1-22
Memory Manager (EXMY)	1-23
Supervisory Services (EXSP)	1-25
Executive Services (EXSV)	1-26
Time Management (EXTI)	1-27
Task Manager (EXTM)	1-28
Software Floating Point Emulation (FLTP)	1-30
File Manager Services Part A (FMA7)	1-31
File Manager Services Part B (FMB7)	1-32
Contiguous File Manager (FMCO)	1-33
Index File Manager (FMIN)	1-34
File Manager (FMS7)	1-35
File Manager Utilities (FMUT)	1-36
SVC Intercept (INTC)	1-37
Communications Support (ITAM)	1-38
Communications File Manager (ITFM)	1-41
Memory Checks (MCHK)	1-43
Mirror Dummy (MIRR)	1-44
Supervisor Call Zero (SVC0)	1-45
Operating System End (UBOT)	1-46

SYSTEM MODULES

This section contains a description of all of the OS/32 system modules that are applicable to a standard release of the operating system.

We have explained the function of each module and listed the module's entry labels and referencing modules for your convenience.

(APSV)

AUXILIARY PROCESSOR SERVICES

FUNCTION:

If the operating system is for a Model 3200MPS, the Auxiliary Processor Services module (APSV) provides the operating system with APU commands via the Commands Processor, permits SVC-6 APU functions, and supports SVC-13 requests.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
APJASP	EXAC
APDIS	FMIN
APCMDSP	CMDB
APC	CMSP
AP.TPTR	EXIO
APUWAIT	
APUSERR	CMSP
APURTOUT	
APUPWR	CMSP
APUPWESR	EXIN
APUINIT	EXIN
APUERTBL	EXIO
APUERRR	
APUERR	
APUERMSG	EXIO
APSV.TIM	CMEX CMSP
APSV.TCL	CMEX
BGTPTR	
FIXAPUQ	EXIO
GRABTASK	CMEX EXSP EXSV EXTM
FLIHAPU	
INITRCVR	
MAXSIGNL	
MAXERIOP	EXIO
LPUERR	
LPU	CMSP
LINKCHK	EXIO
MOVEBYTE	EXIO
RCVRON	EXIO
RCVROFF	EXIO
RCVRISR	
RCVRIOPE	EXIO
RCVRESR	
REEXEC1	FMIN
RELAPU	EXSV
OPTNLPU	CMEX
OPTLPU	CMEX
QUEUE	CMSP

(APSV)

AUXILIARY PROCESSOR SERVICES (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
QUEERR	
PASSBACK	
RTSMTOUT	
RTSMDRVR	EXIO
SORT.CHK	EXSV
SIGNALQ	
SVC13	EXIN
TMCKAPU	EXTM
TMENQDIS	
TMAPUFH	EXTM
TMPREMP	EXIO
ZAPWAIT	EXSV
APSV.CHK	CMSP
APSI.LNK	EXIN
APRSOUT	FMIN
APRSIN	FMIN
APQCON	FMIN

(CDVR)

CONSOLE DRIVER

FUNCTION:

The Console Driver module (CDVR) provides the needed routines and tables to perform I/O to the system console. The routines are the initialization and termination Event Service Routines, SVC 1 function entries, and SVC 2.7 support. The tables include the Input/Output Handler (IOH) and the dummy Device Control Block used for the system console.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CDFUTB	
CDEXTM	EXTM
CDEXMY	EXMR
CDEXIN	EXIN
CDESV3	EXSV
CDSV27	
CIOINCPT	EXIN
CONDVR	
DCBCMD	CMON CMSP FMA7
EXECMSG	EXIN EXMR EXSV EXTM
IOH CDVR	
SVC2.7	EXSP

(CMDB)

BULK DEVICE COMMAND EXECUTORS

FUNCTION:

The Bulk Device Command Executors (CMDB) module provides command executor support for the Command Processor and is entered as an extension of the Command Processor. Support executors are used primarily on direct access devices.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ALLOCATE	CMSP
BFILE	CMSP
BACREC	CMSP
CHECK.ID	CMEX EXAC
CMDELETE	CMSP
CHECKID	EXLD EXSV
D.SYS	
DIRFN	
DIREXT	
DSPFHD	
DISPLAY	CMSP
DISPIND2	
DISPIND1	
DISPFD.Q	APSV
DISPFD.L	APSV
DISPFD.A	APSV
DISPFD	CMEX ERRC EXAC
FFILE	CMSP
FORREC	CMSP
MARKSVC1	
LUTST	
MONBLK	CMEX
MOFFBLK	CMEX
MIR.MERR	
REWIND	CMSP
RVOLUME	CMSP
SECTORBF	EXLD
SPOLFILE	CMSP
SVC7PBLK	
TEMPFILE	CMSP
VOLUME	CMSP
WFILE	CMSP
XDELETE	CMSP
XALLOCAT	CMSP

(CMEX)

COMMAND EXECUTORS

FUNCTION:

The Command Executors (CMEX) module provides command execution support for the Command Processor. This module is entered as an extension of the Command Processor on behalf of the console operator.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
\$IFNG	CMSP
\$IFNE	CMSP
\$IFL	CMSP
\$IFG	CMSP
\$IFE	CMSP
\$EXIT	CMSP
\$ENDC	CMSP
\$ELSE	CMSP
\$COPY	CMSP
\$CLEAR	CMSP
\$BUILD	CMSP
.SPL	CMSP
.MTM	CMSP
.ECM	CMSP
\$WRITE	CMSP
\$TRANS	CMSP
\$TERMJOB	CMSP
\$SKIP	CMSP
\$NOCOPY	CMSP
\$JOB	CMSP
\$IFX	CMSP
\$IFNX	CMSP
\$IFNULL	CMSP
\$IFNULL	CMSP
\$IFNL	CMSP
ASGNERR	CMDB
ASSIGN	CMSP
BIAS	CMSP
CANCEL	CMSP
BUILDFLG	CMSP
BUILD DSP	CMSP
BUILD	CMSP
CMDWTASK	CMSP EXSV
CMRENAME	CMSP
CMDCLOSE	CMDB CMSP
CMDBIAS	CMSP
CMDBGTRC	CMSP
CMDBGT	CMDB CMSP EXSV
CMDASGN1	CMDB CMSP
CMCLOSE	CMSP

(CMEX)

COMMAND EXECUTORS (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CUSTOMCK	CMSP
CONTINUE	CMSP
CSSSKIP	CMSP
CSSLIST	CMSP
CSSLEVEL	CMDB CMSP
FDERROR	CMDB
ERRORC	CMSP
GETFRAC	MCHK
IFLVL	CMSP
LOADER	CMSP
MARK	CMSP
JOBSKIP	CMSP
JOBIFLVL	CMSP
JOBFLAG	CMSP
LUERROR	CMDB
MEMERR	CMDB
MEMORY	CMSP
NOPRERR	CMDB ERRC
READCMD	EXIN
READ	CMSP
OPTIONS	CMSP
PAUSE	CMSP
NXTOPT	APSV
PROMODE	CMDB CMIR CMSP
REPROTEC	CMSP
REMOVE	CMSP
SET	CMSP
SEQERR	CMDB CMSP
START	CMSP
SPT.UTCB	CMDB CMIR CMSP
SEND	CMSP
TCOM	CMSP
TASKERR	CMDB
TASK	CMSP
WRITE	CMSP

(CMIR)

INTERNAL READER

FUNCTION:

If the Internal Reader support is included in the operating system at sysgen time, the Internal Reader (CMIR) module allows the user to support the SVC 2.14 functions. These functions permit the foreground task to issue system commands during execution. This module also supports the commands necessary to create the buffers for the Internal Reader.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
IREADER	CMSP
IRDRINIT	CMSP
IRBUFFER	CMSP
SVC2.14	EXSP

(CMON)

CONSOLE MONITOR

FUNCTION:

The Console Monitor (CMON) module controls all messages to and from the system console, including the following:

SVC 2.7 Log message

SVC 1 Read and Write requests

Command Processor requests

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ACTIVELB	CMEX
ATTN.P	CMEX CMSP
CMDP.TCB	APSV CMDB CMEX CMIR CMSP ERRC EXIN
	EXIO EXTM FMUT
CMSP.UDL	
COMMAND	EXTM
LOGFLAG	CMEX CMSP ERRC
LOGCKPTI	CMEX
LOGCKCNT	CMEX

(CMSP)

COMMAND PROCESSOR

FUNCTION:

The Command Processor (CMSP) module services commands, handles errors, expands and processes CSS, and performs logging.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>						
\$CONTIN	CMEX						
BLANKBUF	APSV	CMDB	CMEX	ERRC	EXAC	MCHK	
CMDP.Q							
CMDP.GO	CMON						
CMDNEXT	APSV	CMDB	CMEX	CMIR	ERRC	EXAC	MCHK
CMDERROR	CMDB	CMEX	ERRC	EXAC			
CMDERREX	APSV	CMIR	MIRR				
CMDERMV	CMEX						
CMDEMPTY	APSV	CMDB	CMEX	ERRC	EXCC	EXIO	MIRR
CMDWRITE	APSV	CMDB	CMEX	CMIR	ERRC	EXAC	EXIO
	MCHK	MIRR					
CMDWGBX	CMDB	CMEX	MCHK				
CMDWGB	APSV	CMDB	CMEX	ERRC	EXAC		
CMDTYMV	CMEX						
CMDRDBLK	CMEX						
CMDPSYSG	CMDB	EXAC					
CHECKCSL	CMDB	CMEX					
CMDP.UDL	CMON						
CSSTST	CMEX						
CSSTSK	EXSV						
COMMACK1	CMDB	CMEX	EXAC				
COMMACK	CMDB	CMEX	ERRC	MCHK			
CSSCLOSE	CMEX						
CSSBUF	CMIR						
CSLTSK	CMEX	CMIR	EXSV				
CSKIPA	CMEX						
ERR.COM1	CMEX						
ERR.COM	CMDB	CMEX	MCHK				
ERR.ASGN	CMDB	CMEX	ERRC				
FLTPFLAG	EXIN						
GENBUFFE	CMDB						
GENBUFF	APSV	CMDB	CMEX	ERRC	EXAC	MCHK	
FORMERR	APSV	CMDB	CMEX	MCHK			
IOERR	CMDB	ERRC					
IOEMSG	CMEX						
MNMFINDD	CMDB	CMEX					
MNEMERR							
MSGLOG	CMEX						
PICADDR	EXTI						
PREPRO	CMEX						

(CMSP)

COMMAND PROCESSOR (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
PARMERR	APSV CMDB CMEX ERRC EXAC ITFM MCHK MIRR
PRTYERR	CMEX
SETLU	CMEX
SCANNER	APSV CMDB CMEX CMIR ERRC EXAC MCHK MIRR
SV1ERTB	CMEX EXMR
TESTPRT	CMON
TERMCHK	CMEX
TIMERR	CMEX
SYSPSIZE	CMDB
SYSPADDR	CMDB
XAIDSBRK	

(DUMP)

PANIC DUMP SUPPORT

FUNCTION:

The DUMP module provides the operating system with the necessary code to execute a panic dump in case of a system failure.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CON.PUT	
CON.GET	EXIN
DMP.SOFT	EXIN
DUSI.LNK	
DUMPINIT	
DUMP.CON	EXIN
DUMP.CNW	
DUMIPC	
GETC	
MSG	EXIN EXMY
PANIC	EXIN
TOPCSRCH	EXMY

(ERRC)

ERROR RECORDING

FUNCTION:

If error recording support is included in the operating system at sysgen time, the Error Recording (ERRC) module allows the error logger hardware to periodically update an error log file. This module also supports recording of direct access device errors and system milestones.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CMDPERFG	CMSP
ERC.SQFL	EXIO
ERC.SQDR	EXIO ITAM
ERC.SQDP	EXIO
ERC.RECD	CMEX
ERC.QERR	EXIO
ERC.PFR	CMSP
ERC.PERD	CMEX
ERC.MNNB	CMEX
ERC.MKON	CMDB
ERC.MKOF	CMDB
ERC.MFNB	CMEX
ERC.MENT	EXIO
ERC.LOG	CMEX
ERC.IXPR	FMB7 FMIN
ERC.GINI	CMSP
ERC.FMSQ	EXIO
ERC.ENTR	EXIO
ERC.DVRS	EXIO
ERC.DRPR	FMA7 FMB7
ERC.DRBL	FMUT
ERC.DISP	CMDB
ERC.BTMR	FMUT
ERC.BTMA	FMUT
ERC.AZAP	APSV
ERC.ARLS	APSV
ERC.ARCV	APSV
ERC.AQFL	APSV
ERC.AQDP	APSV
ERC.APER	APSV
ERCF.MXR	
ERCD.STA	
ERCD.LST	
ERC.VOL	CMDB
ERC.TREN	CMSP
ERC.TRE2	CMSP
ERC.TQOV	EXTM
ERC.SQNL	EXIO

(ERRC)

ERROR RECORDING (continued)

ENTRY LABELS AND REFERENCING MODULES.

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ERSI.LNK	EXIN
ERRCINIT	EXIN
MER.ADR	
MER.CNCL	CMEX
MER.LOG	EXTI
MER.REST	
MER.RSCH	CMEX
MER.SCHD	
MER.STRT	
MERD.BUF	
MERD.CNF	
MERD.STA	
MERD.TEF	
REL.MASK	EXIN

(EXAC)

ACCOUNTING SUPPORT

FUNCTION:

If accounting support is included in the operating system at sysgen time, the Accounting (EXAC) module provides for the accounting data accumulation and storage.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ACUMTIME	EXTM
ACT.SMSG	FMB7
ACT.BMSG	FMB7
ACCTZERO	EXTM
DSPACCT	CMDB
HOLDTIME	EXTM
SV4.ENAC	EXSV
SV4.RMAW	EXSV
SV4.RDAC	EXSV
SVC2.25	EXSP
SVC2.24	EXSP
TCPU.TCB	EXIN
TCPU.TIM	EXIN
TCPU.TYP	EXIN
TMCPUON	EXIO EXTM
TMCPUOFF	EXTM

(EXIN)

EXECUTIVE INTERRUPTS

FUNCTION:

The Executive Interrupts (EXIN) module is responsible for the establishment of low core values as required by various processors as well as handling all internal interrupts. This module also contains the mainline code for initialization and the crash handler.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ALINUT	APSV
AFHUT	APSV
CTR162	
DISP2SQS	EXIO EXTM
EREGS	APSV ERRC
INITDLAY	EXIO
IIHUT	APSV
IIH	
ISH	APSV CMIR EXAC EXSP EXSV EXTI SVCO
ISHRS	EXMY EXSV EXTI
INITSQS	EXIO
MEMFLTRS	CDVR EXIO EXMY EXSP EXSV EXTI
MEMFAULT	CMIR EXAC EXIO EXMY EXSP EXSV EXTI
MMPRDCBS	EXIO
MMPARUT	APSV
MMH	ERRC EXMY
MFHUT	APSV
PWESR	EXIO
RTSMDEV	APSV
SI.LINK	
SQS2DISP	EXIO
SYSINIT	
XREGS	ERRC

(EXIO)

EXECUTIVE I/O SERVICES

FUNCTION:

The Executive I/O Services (EXIO) module provides the routines and default tables necessary to perform standard I/O requests for the common drivers. This module also contains the routines needed by System Queue Service and the tables and structures for the null device.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
BARWIOD	
BAREWR	
BARETST	
BARERD	
BAREIOH	
BARCMN	
CMDPIOPS	CMSP
DISKQ	
COMDIS	FMCO
COMQ	CDVR
DIRDONE	CDVR
CONTIG.W	FMCO
DCB25500	ITAM
COMIOH	
COMFIFO	
DSPTCH	FMCO
EVCON	
EVDIS	APSV EXLD FMA7 FMB7 FMIN FMS7
EVMOD	ITAM
EVQCON	APSV EXLD FMA7 FMB7 FMIN FMS7
EVREL	
EVRTE	CDVR EXIN ITAM
EX.DVRS	
GETIOBC	APSV
GETIOB	CDVR
GETQCBI	
GETQCBA	APSV
GETQCB	
H1NCON	
H1EEXIT	
IOSYSINI	EXIN
IOSI.LNK	EXIN
IOSETCC	FMCO
IOREMW	FMCO
IOPTIM	EXTI
IOPMSG	CMSP

(EXIO)

EXECUTIVE I/O SERVICES (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
IOPISR	
IOP.CRSH	EXMY
IODTWT2	
IODTWT	FMCO
IODONE2	
IODONE	CDVR
IODGST2	
IODGST	FMCO
IODCOM2	
IODCOM	
LFIFO	
LEV7DUM	
LEAFQ	CDVR FMCO
LEAFMIOB	
LEAFCON	CDVR FMCO
MDISKINI	
MMPRPWRF	EXIN
MMPRENAB	EXIN
MMPRCHEK	EXIN
MSG0	
MSGJ	
MSGI	
MSGH	
MSGG	
MSGF	
MSGE	
MSGD	
MSGC	
MSGB	
MSGA	
MSG9	
MSG8	
MSG7	
MSG6	
MSG5	
MSG4	
MSG3	
MSG2	
MSG1	
NLIFO	
NFIFO	
NSEVREL	
MTB.RTER	
MTB.RSYP	
MTB.RSHW	
MTB.RSER	

(EXIO)

EXECUTIVE I/O SERVICES (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
MTB.RQCB	
MTB.PRSW	
MTB.PRHW	
MTB.PRER	
MTB.PRBS	
MTB.PISW	
MTB.PIOK	
MTB.PIHW	
MTB.PIER	
MTB.PIBS	
MTB.PFMC	
MTB.INSW	
STB.PRBS	
STB.PISW	
STB.PIOK	
STB.PIHW	
STB.PIER	
STB.PIBS	
STB.PFMC	
STB.NIOB	
STB.MIRD	
STB.INSW	
STB.INHW	
STB.INER	
STB.INBS	
STB.ILVL	
STB.IIOB	
STB.IBRK	
STB.HNON	
STB.FULQ	
STB.ERRL	
STB.DTSQ	
STB.BQAC	
STB.BLSQ	
SV1XFCER	MIRR
SV1FCER	CDVR FMCO FMIN
SV1ADER1	FMCO FMIN
SVCl.IOP	APSV
SVCl.0	FMUT
SVCl	EXIN
SVClWRIT	CDVR
SVClWFM	
SVClWAIT	CDVR ITFM
SVClTEST	CDVR
SVClREW	
SVClREAD	

(EXIO)

EXECUTIVE I/O SERVICES (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
SVC1NOOP	FMCO FMIN
SVC1NINA	FMCO
SVC1NIA	
SVC1INA	FMCO
SVC1IA	FMCO
SVC1HALT	CDVR
SVC1FSR	
SVC1FFM	
SVC1EXIT	FMCO
SVC1DDF	
SVC1BSR	
SVC1BFM	
TIMEOUT	EXIN
TOCHON	ITAM
TOCHOFF	ITAM
WAITIO	
MTB.INHW	
MTB.INER	
MTB.INBS	
MTB.IBRK	
MTB.HMON	
MTB.BQAC	
RCVQ.ISR	APSV
RCVQ.ESR	APSV
NULLIOH	
PRTYQ	
RELIOBC	APSV
RELIOBA	FMIN
RELIOB	CDVR FMCO
S1	
RELQCB I	FMCO
RELQCB A	APSV
RELQCB	
RETQCB	
SENQCB A	APSV
SENQCB	
STB.BAD0	
STB.BESQ	
SANITY	
SAN.TBL	
SAN.REGS	
SAN.FMT	
S9	
S8	
S7	
S6	

(EXIO)

EXECUTIVE I/O SERVICES (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
S5	
S4	
S3.NOP	
S3	
S2	
SQS	EXIN
SQS.EX	APSV ERRC EXTI FMCO ITAM
SQS.MLV	
SQS.SLV	ITAM
SEEKPOS	FMCO
SEEKCHK	
STB.TIMC	
STB.RTER	
STB.RSYP	
STB.RSHW	
STB.RSER	
STB.RQCB	
STB.PRSW	
STB.PRHW	
STB.PRER	

(EXLD)

EXECUTIVE LOADER

FUNCTION:

The Executive Loader (EXLD) module supports all load requests for the operating system. This module is responsible for the loading of tasks, common segments, and partial images.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
LOAD.SRF	EXSV
LOADSEG	CMEX
LOADIMG	EXSV
RELES	EXSV
RRFILE	EXMR
RELPST	EXSV
RELOCIMG	EXSV
SV6.LOAD	EXSV
TCBLOOK2	CMEX EXAC EXSP
TCBLOOK	APSV CMDB CMEX EXSV INTC

(EXMR)

ROLL MANAGER

FUNCTION:

The Roll Manager (EXMR) module is responsible for all roll support within the operating system. This support will include both rolling in and rolling out logic.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ROSIZE	EXTM
ROSCAN	EXTM
ROLLPRIO	EXTI
ROLLOUT	EXMY
ROLLIN	EXTM
ROLIST2	EXTM
ROERTYPE	CDVR
ROERSTYP	CDVR
ROERNAME	CDVR
ROERMSG	CDVR
ROERMSG	CDVR
SEGTYPE	CDVR

(EXMY)

MEMORY MANAGER

FUNCTION:

The Memory Manager (EXMY) module supports the routines necessary to acquire, release, and monitor system and task space, as well as supervisor services for the SVC 2 codes 2, 3, 20, 27, and 21.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>							
ADCHK	CDVR	EXAC	EXIN	EXLD	EXSP	EXSV	EXTI	INTC
ADCHK1	CDVR	EXIN	EXLD	EXSV	EXTM	INTC	ITAM	
ADCHK2	EXSP	EXSV	INTC					
ADCHK2ET	INTC							
ADCHK2F	APSV	INTC						
ADCHK2F1	CMIR	EXSV	EXTI	INTC				
ADCHK2F2	INTC							
ADCHKAPU	APSV							
ADCHKF	EXSP	EXSV	INTC					
ADCHKF1	EXSV	INTC						
ADCHKNS	EXAC	EXIN	EXIO					
ADCHKNS1	INTC							
DE.ADCHK	ITAM							
IMP.MAP	EXLD	EXMR						
GETZERO	CMIR	FMA7						
GETFCB	FMA7	FMB7						
GETTQE	EXTI							
GETTCB	EXLD							
GETSYP	CMDB	CMIR	CMSP	EXIO	EXLD	EXSV	FMA7	INTC
GETSTSYP	EXLD							
GETSSUB	EXLD							
GETSKMEM	EXLD	EXMR						
GETMEM	EXLD							
GETSDE	EXLD	MCHK						
GT.OKROL	EXMR							
GT.NOROL	EXMR							
MYSYSINI	EXIN							
MYSI.LNK	EXIN							
PUR.MAP	EXLD	EXMR						
RELEFCB	FMA7	FMB7						
PSWSAV	EXLD							
RTL.MAP	EXLD							
RELMEM	EXLD	EXMR	EXSV	MCHK				
RELTQE	EXIN	EXTI						
RELTCB1	EXLD							
RELTCB	EXSV							
RELSYP	CMDB	CMIR	CMSP	EXIO	EXLD	EXMR	EXSV	
	FMA7	FMUT	INTC					
RELSSUB								

(EXMY)

MEMORY MANAGER (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
RELSDE	EXLD EXSV MCHK
RETSAV	
SSEG.MAP	EXLD
SSTD	APSV EXIN EXLD
SEG.ADD	EXLD EXMR EXSV MCHK
SEG.DIS	EXSV
SEG.DISZ	EXLD
SEG.RMV	EXLD EXMR EXSV MCHK
SEG.ROLL	EXLD EXMR
SEG.UMAP	EXMR
SEG.USE	EXMR
SEGBAD	
SVC2.2	EXSP
SVC2.20	EXSP
SVC2.21	EXSP
SVC2.27	EXSP
SVC2.3	EXSP
TCM.MAP	EXLD

(EXSP)

SUPERVISORY SERVICES

FUNCTION:

The Supervisory Services (EXSP) module provides second level interrupt support for some SVC 2 services.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CANEOJ	CMEX EXSV
EXECSRSCC	EXMY EXTI
JOURNAL	
SVC2	EXIN
TERM	EXMR
TERM1	EXIN EXTI
UNPACK	CDVR EXIN EXIO EXTI FMUT

(EXSV)

EXECUTIVE SERVICES

FUNCTION:

The Executive Services (EXSV) module provides support for the following SVC calls:

SVC 3	end of task
SVC 4	I/O intercept service
SVC 5	load an overlay (old)
SVC 6	task management services
SVC 9	task status word
SVC 10	auto overlay support
SVC 14	user SVC support

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
COMEOT	
MAI.31	EXLD
MOVESOPX	EXLD
MTMRPTNS	EXTM
PUR.DIS	EXLD
SETUTOP	EXLD EXMR
SETDELAY	EXLD EXMR
SUBCKRPT	
SUBRPORT	EXMR
SUBRPTNS	EXTM
SV3.TERM	EXTM
SV4.ERR3	EXAC
SV4.NEXT	EXAC
SVC10	EXIN
SVC14	EXIN
SV6.ERR2	EXLD
SV6.ENU1	EXLD
SV6ERTB	CMEX
SV6ERCD	CMEX
SVC6	EXIN
SVC5	EXIN
SVC4	EXIN
SVC3	EXIN
SVC9	EXIN

(EXTI)

TIME MANAGEMENT

FUNCTION:

The Time Management (EXTI) module is responsible for all time functions within the operating system including Interrupt Service Routines (ISR) and Event Service Routines (ESR) for the Precision Interval Clock (PIC) and Line Frequency Clock (LFC). This module also supports supervisor services for the SVC 2 codes 8, 9, 10, 11, and 23.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CTR181	
CPU.TIME	EXAC
GETUREG	EXAC
III	APSV EXIN EXIO ITAM
ISRPIC	
ISRLFC	EXIN
ITAMTIMC	ITAM
ITAMFREQ	CMEX ITAM
ITAMCNT	ITAM
ITM.FREQ	CMEX ITAM
PICESR	EXIO
SVC2.23	EXSP
SVC2.10	EXSP
SVC2.11	EXSP
TM.FEB	CMEX
TISYSINI	EXIN
TISI.LNK	EXIN
TIMESR	EXIO
SVC2.9	EXSP
SVC2.8	EXSP
TIFXTCHN	CMEX
TICAN1	EXSV
TICAN	EXSV
TM30DAYS	CMEX
TM.FREQ	CMEX EXIN

(EXTM)

TASK MANAGER

FUNCTION:

The Task Manager (EXTM) module provides the routines and services to manipulate queues, dispatch and monitor tasks, and to perform context switching.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CDISPPSW	EXIN
CTDISP	
DISPPSW	CDVR EXIN EXIO EXLD EXSV FMIN FMUT ITFM
RDISPPSW	CDVR EXIN EXIO EXMR EXSV INTC
TMATQ1	APSV CDVR ERRC EXIN EXIO EXLD EXSV EXTI FMIN FMUT
TMATQ	CMIR EXSV
TMASQ1	CDVR EXSV
TMASQ	
TMENQPRI	APSV
TMENQTL	APSV
TMNSOUT	APSV CMIR EXAC EXMY EXSP EXSV EXTI INTC ITAM
TMDEQ	
TMDEQHD	APSV
TMDISP2	
TMDISPQO	
TMPRIO	EXTI
TMRDYDEQ	CDVR EXLD EXMR EXSV INTC ITFM
TMRDYENQ	APSV CMEX EXIN EXSP EXSV
TMREMW	CDVR EXAC EXIN EXIO EXLD EXMR EXSP EXSV EXTI FMCO
TMRINDEQ	EXSV
TMRINENQ	EXMR EXSV
TMRSAIN	EXIN EXLD
TMRSAOUT	EXLD EXMR EXSV
TMRSARS	EXAC EXIN EXSV FMA7
TMRSIN	APSV CDVR CMIR EXIN EXLD EXMR EXSP EXSV EXTI INTC
TMRSOUT	APSV CDVR CMIR EXIN EXLD EXMR EXSP EXSV EXTI FMA7
TMRSRSA	EXAC EXSV FMA7
TMSETW	EXIO EXLD EXMR EXSP EXSV EXTI FMIN INTC ITFM
TMSI.LNK	EXIN
TMSTART	EXLD EXSV
TMSTOP	EXSV
TMSTOPW	EXSV

(EXTM)

TASK MANAGER (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
TMSTSW	EXIN EXSV
TMSYSINI	EXIN
TMTSWN	

(FLTP)

SOFTWARE FLOATING POINT EMULATION

FUNCTION:

If floating point software support is included at sysgen, the Software Floating Point Emulation (FLTP) module will be in the operating system. If hardware floating point is sysgened or if no floating point is sysgened then the FLTP module will not be included.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
FLTP	

(FMA7)

FILE MANAGER SERVICES PART A

FUNCTION:

The File Manager Services Part A (FMA7) module is responsible for assigning, closing, and check pointing for the file manager.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
F7.CLOSE	FMS7
F7.CHK	FMS7
F7.ASNP	FMB7
F7.ASGN	FMS7
OPEN.DEV	ITFM
OPEN.FIN	

(FMB7)

FILE MANAGER SERVICES PART B

FUNCTION:

The File Manager Services Part B (FMB7) module is responsible for allocating, deleting, renaming, reprotecting, and changing file access privileges for the File Manager.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
F7.CAP	FMS7
F7.REP	FMS7
F7.REN	FMS7
F7.DELTE	FMA7
F7.DELSP	FMA7
F7.DEL	FMS7
F7.ALLO	FMS7

(FMCO)

CONTIGUOUS FILE MANAGER

FUNCTION:

The Contiguous File Manager (FMCO) module provides the executors for contiguous file I/O.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CNTGIOH	FMA7
CNTCIOD	
IODINIT	
IODEOF	
SECTBUMP	

(FMIN)

INDEX FILE MANAGER

FUNCTION:

The Index File Manager (FMIN) module provides the executors for I/O to files that use data and index blocks that may be buffered.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
INDXIOH	FMA7
FMWAIT	FMA7

(FMS7)

FILE MANAGER

FUNCTION:

The File Manager (FMS7) module provides primary SVC 7 support for the operating system.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
EXT7.TAB	EXIN
EXT7.TAE	EXIN
SV7.CMD	FMA7 FMB7 ITFM
SVC7E.OD	FMB7
SVC7E.OC	FMA7
SVC7E.OB	FMA7 FMB7
SVC7E.OA	FMB7
SVC7E.O9	FMA7 FMB7
SVC7E.O8	FMB7
SVC7E.O7	FMA7 FMB7
SVC7E.O6	FMA7 FMB7
SVC7E.O5	FMB7
SVC7E.O4	FMA7 FMB7
SVC7E.O3	FMA7 FMB7
SVC7E.O2	FMA7 FMB7
SVC7E.O1	FMA7 FMB7
SVC7	EXIN

(FMUT)

FILE MANAGER UTILITIES

FUNCTION:

The File Manager Utilities (FMUT) module provides the subroutines used by all the file management modules for SVC 7 support.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>					
ACCHECK	FMB7					
ALLO.TNM	FMA7	FMB7				
ALLOD	FMB7					
APCHECK	FMA7	ITFM				
APCHECK1						
APCHECK2	FMA7					
CDDELETE	FMB7					
CDALLOC	FMB7					
CDRENAME	FMB7					
CLO.BLK	FMA7	FMB7	FMS7			
CLO.BLK	FMA7					
CKWRAUT	FMA7	FMB7				
CKRDAUT	FMA7	FMB7				
DIRLOOK	FMA7	FMB7	FMS7			
DEVCHECK	FMB7					
DATE.DIR	EXAC	EXTM	FMA7	FMB7		
DMTLOOK	CMDB	CMEX	CMON	DUMP	ERRC	EXIO
	EXSV	FMA7	FMB7	FMS7		
FDCHECK	FMA7	FMB7	FMS7			
GETD	FMA7	FMB7	FMS7			
GETSECTR	EXLD	FMA7	FMB7	FMIN		
MAKEXFNM	FMA7	FMB7	FMS7			
LUCHECK	FMA7	FMB7	FMS7			
MOVE	FMIN					
MOVENULL	FMIN					
OPN.BDSK	FMA7	FMB7	FMS7			
PUTB	EXLD	FMA7	FMB7			
PUTD	FMA7	FMB7				
RELEB	EXLD	FMB7				
RESET.RW	FMA7	ITFM				
TESTRS	EXSV					
VMTLOOK	CMDB	EXLD	EXMY	EXSP	FMA7	FMB7
	FMS7	INTC				
WAITRS	EXAC	EXSV	FMA7			

(INTC)

SVC INTERCEPT

FUNCTION:

If intercept support is included in the operating system at sysgen time, then the SVC Intercept (INTC) module supports the creation of SVC intercept paths. This module contains the routines required to create and remove intercept paths and to perform interception on behalf of a task.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
SV4ITERM	EXSV
SV4IROLL	EXSV
SV4IREMV	EXSV
XDMTLOOK	CMDB FMA7 FMB7 FMS7
XTCBLOOK	EXSV
DORSINTC	EXSV FMS7
IPAUSCHK	EXSV EXTM
INTERCPT	
NSINTCPT	EXIN EXIO
SV4ICONT	EXSV
SV4ICRSP	EXSV
SV4ICRTE	EXSV
SV4IGET	EXSV
SV4IPROC	EXSV
SV4IPUT	EXSV
SV4ITRAP	EXSV

(ITAM)

COMMUNICATIONS SUPPORT

FUNCTION:

If Communications Support (ITAM) is included in the operating system at sysgen time, then the communications module will permit the use of communication devices, commands, and SVC 15 support

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CC813	
CC812	
CC811	
CC810	
CC809	
CC808	
CC807	
CC806	
CC805	
CC804	
CC803	
CC802	
CC801	
BAD.ADD	
CANITAM	EXIO
CMEXIT	
CMTERM	
EMT.FREQ	EXTI
HALTITAM	
IEXAMIN	
ICMDINT	
IT.ADCK3	
IT.ADCK1	
IT.ADCK	
IT.DADK3	
IT.DADK1	
IT.DADK	
ISR.JRNL	
ISSEXEC	ITFM
IT..STOP	
IT.HALT1	
IT.HALT	
IT.INIT	
ITAMNULL	
ITAMMODE	
ITAMINIT	EXIN
ITNXTCMD	
ITNODSR	

(ITAM)

COMMUNICATIONS SUPPORT (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ITNOCL2S	
ITNOCAR	
ITMODCOM	
ITMOCW	
ITMOCR	
ITAM.SQS	
ITAM.DU	
IT.ULEAF	
ITDISC	
ITCONCHK	
ITBTCHK	
ITAOC	
ITLSDSR	
ITLSC2S	
ITLSCAR	
ITLDCT	
ITITV	
ITISTOTC	EXTI
ITISSTOP	
ITISSTAT	
ITISPOTC	
ITISKILL	
ITISCRSH	
ITIS.DU	
ITIS.CRS	
ITIMV1	
ITIMUNLK	
ITIMLINK	
ITGETMO2	
ITGETDAT	
ITGETBUF	
ITFC	
ITEMTLF	EXTI
ITDOCW	
ITDOCR	
ITXFRISR	
ITWR.RD	
ITSYCT	
ITSRABS1	
ITSRABS	EXTI
ITSPCW	
ITSPCR	
ITSI.LNK	EXIN
ITSETREA	
ITPARERR	

(ITAM)

COMMUNICATIONS SUPPORT (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ITOVFL	
ITOTV	
MPCRECON	
MPCPOWR	
MPCINIT	
MPCHECK	
MTFIFO	
RAWCHKR	
RAPCHKR	
SVC15	EXIN
SVC15PB	
TOUTITAM	EXTI

(ITFM)

COMMUNICATIONS FILE MANAGER

FUNCTION:

The Communications File Manager (ITFM) module provides the support for tables and buffers required when using some of the communications facilities.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CAP.CHK	
CAP.SVC1	
CLOS.XIT	
CLOS.SV1	
CLOS.RST	
CLOS.RD	
CKPT.NON	
DCTLOOK	
CPCRIFM1	
ERR.01	
ERR.04	
ERR.05	
ERR.06	
ERR.07	
ERR.08	
ERR.09	
ERR.OA	
ERR.COMN	
GETFMH	
FMH.NULL	
FMH.SVCF	
FTCH.DCB	
GETCPH	CMDB
IT.ALLOC	FMB7
IT.CAP	FMB7
IT.CHKPT	FMA7
IT.CLOSE	FMUT
IT.DELET	FMB7
IT.FETCH	FMS7
IT.FMXIT	
IT..WAIT	
IT.OPEN	FMA7
IT.ORJE	FMA7
IT.RFMXT	
IT.RSLU	EXSV
ITDLUNOP	
ITPLOC	
LCBFILL	
LCBLOOK	

(ITFM)

COMMUNICATIONS FILE MANAGER (continued)

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
ITO.DNS	
ITO.AOC	
OPEN.DON	
OPN.SVC1	
OPN.SVCF	
RERR.COM	
RERR.OA	
RERR.09	
RERR.08	
RERR.07	
RERR.06	
RERR.05	
RERR.04	
RERR.01	
WAIT	
WAITRSI	

(MCHK)
MEMORY CHECKS

FUNCTION:

If included in the operating system at sysgen time, the Memory Diagnostics (MCHK) module enables the use of the MEMORY command to test, mark off, or mark on a block of memory. This module is also referenced during system initialization to perform memory checks.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
INITMTST	CMSP
MEMONOFF	CMEX
MEMCHKER	EXIN
MEMCHK2	EXIN
MKOFMEM	EXIN EXMY

(MIRR)

MIRROR DUMMY

FUNCTION:

The Mirror Dummy (MIRR) module is needed to provide the labels as referenced if mirror support were included in the operating system. Mirror disc is a special package and requires libraries not provided with the operating system. Generating an operating system with mirror support, but not having the appropriate libraries, will result in undefined labels at LINK time.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
CMTSR1	FMCO
CMTSW	FMCO
CNTMBIOD	FMCO
CNTMFIOD	FMCO
CNTMRIOD	FMCO
CNTMWIOD	FMCO
MDERRREP	CMSP
MDERRORS	CMSP
MDERRORP	CMSP
MDCBCHK	FMCO
MIR.MOFF	CMEX
MIR.PSCN	CMDB
MIR.BARR	EXIO
MIR.BART	EXIO
MIR.BARW	EXIO
MIR.CLEN	CMDB
MIR.DUPE	CMDB
MIR.LCON	EXIO
MIR.MAIN	CMDB
SWOP	CMSP

(SVC0)

SUPERVISOR CALL ZERO

FUNCTION:

The SVC0 module enables users to define their own SVC function. The operating system will then support an SVC0 by default.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
SOSI.LNK	EXIN
SOSYSINI	EXIN
SVC0	EXIN

(UBOT)

OPERATING SYSTEM END

FUNCTION:

The Operating System End (UBOT) is an object file that is included by link during the sysgen process. This module will define the end of the operating system and the transfer address for the start of the operating system.

ENTRY LABELS AND REFERENCING MODULES

<u>SYMBOL</u>	<u>REFERENCED BY</u>
OS.PATCH	
OS.END	CMSP EXSP
UBOT	DUMP
UBOT.0	

***OS/32 INTERNALS
SYSTEM STRUCTURES***

**S
E
C
T
I
O
N

2**

SYSTEM STRUCTURES

This section of your Internals student guide contains specific operating system structures selected for this OS/32 Internals course.

Structures depict the relationship of operating system modules to each other and to mechanisms such as input/output devices, data files, directories, etc., and are models used as a basis for discussion.

We hope you find these structure depictions helpful and easy to use.

TABLE OF CONTENTS

STRUCTURES

Access Control Block (ACB)	2-1
Auxiliary Processor Block (APB)	2-3
Channel Control Block (CCB)	2-7
Context Block (CTX)	2-8
Device Dependent Segment (DDCB)	2-9
File Directory (DIR)	2-13
Device Mnemonic Table (DMT)	2-14
Event Coordination Node (EVN)	2-15
File Control Block (FCB)	2-17
File Descriptor (FD)	2-22
Free Block Descriptor (FDE)	2-23
Index File Manager Extended SVC 1	2-24
Intercept Control Block (ICB)	2-25
Input/Output Block (IOB)	2-27
I/O Handler Routine (IOH)	2-29
IOP Processor Block (IPB)	2-31
Intercept PCB (IPCB)	2-39
Intercept Read Control Block (IRCB)	2-41
Initial Value Table (IVT)	2-42
Loader Common/Entry (LCE)	2-43
Loader Information Block (LIB)	2-44
Logical Processor Mapping Table (LPMT)	2-46
Loader Sharable Segment Entry (LSE)	2-47
Loadseg Parameter Block (LSG)	2-48
Loader Task's TCB Redefinitions (LTCB)	2-49
Overlay Control Block (OCB)	2-51
Overlay Descriptor Table (ODT)	2-52
ORT Header (ORH)	2-53
Overlay Reference Table (ORT)	2-54
Device Independent Segment (PDCB)	2-55
Private File Control Block (PFCB)	2-58
Pseudo Device Control Block (PSDCB)	2-61
Pseudo Task Control Block (PSTCB)	2-62
Queue Control Block (QCB)	2-63
Queue Control Block Header (QCH)	2-65
Queue Handler (QH)	2-66
Queue Parameter Block (QPB)	2-67
RS/RSA Context Block (RCTX)	2-68
Roll Selection List (RLST)	2-69
Segment Descriptor (SDE)	2-70
Secondary Directory (SECDIR)	2-71
System Pointer Table (SPT)	2-73
Segment Table Entry (STE)	2-82
System Value Table (SVT)	2-83
System Space (SYP)	2-84
Task Control Block (TCB)	2-85
Accounting Data Extension (TIO)	2-94
Task Queue-Head (TKQ)	2-95
Timer Queue Entry (TMQ)	2-96
Task Event Queue Entry (TQE)	2-97
Task Event Queue Header (TQH)	2-98
Task Trap Block (TTB)	2-99
Volume Descriptor (VD)	2-100
Volume Mnemonic Table (VMT)	2-101

ACCESS CONTROL BLOCK (ACB)

0(0)	FCB (4) POINTER TO SYSTEM.DIR FCB	
4(4)	SLU (4) ROOM TO SAVE USER LU	
8(8)	PB (24) PARAMETER BLOCK FOR SYSTEM.DIR	
32(20)	FLGS(4) FLAGS	
36(24)	WORK(4) SCRATCH CELL	
40(28)	HIGH(4) HIGH WATER MARK - LAST LOGICAL RE	
44(2C)	NBK (4) NUMBER OF BLOCKS - SIZE OF FILE	
48(30)	EBLK(4) EMPTY BLOCK	
52(34)	EOFF(4) EMPTY OFFSET	
56(38)	DELK(4) SAVED EMPTY DIRECTORY POINTER	
60(3C)	DOFF(1) SAVED EMPTY OFFSET	61(3D) (3) (RESERVED)
64(40)	BKSZ(2) PAGE SIZE IN SECTORS	66(42) SOFF(2) SECTOR OFFSET IN PAGE

ACCESS CONTROL BLOCK (ACB) continued

68(44)	(12) (ENSURE QUADWORD ALIGNED BUFFER)
80(50)	BUF(?) ? is based upon the CD option

AUXILIARY PROCESSOR BLOCK (APB)

0(0)	(8)	UNUSED
8(8)	CTCB(4)	CURRENT TCB
12(C)	FLGS(2)	APU INTERNAL FLAGS
14(E)	MODE(1)	OPERATING MODE FLAGS
15(F)	ID(1)	APU-ID (SLOT # IN APBDIR)
16(10)	HRQP(4)	PTR HOST RECEIVE TASK-QUEUE-HEAD
20(14)	SSTD(4)	SHARED SEGMENT TBL DESCR.
24(18)	PRCA(4)	APB PASSBACK REASON CODE
28(1C)	PFSP(4)	POWER FAIL SAVE PTR
32(20)	XOP(4)	XIO OLD PSW
36(24)	XOL(4)	XIO OLD LOC
40(28)	XDEV(4)	XIO DEVICE NUMBER
44(2C)	XSTA(4)	XIO DEVICE STATUS
48(30)	XNL(4)	XIO NEW LOC

AUXILIARY PROCESSOR BLOCK (APB) continued

52(34)	ISPS(4) IOP: START OF ISP
56(38)	ISPE(4) IOP: END OF ISP
60(3C)	ARQP(4) PTR APU READY TASK-QUEUE-HEAD
64(40)	PRCB(4) PRC INFORMATION
68(44)	PRCC(4) PRC INFORMATION
72(48)	PRCD(4) PRC INFORMATION
76(4C)	PRCE(4) PRC INFORMATION
80(50)	PRCF(4) PRC INFORMATION
84(54)	STRC(4) SYSTEM DATA STRUC ERROR (PHYS) ADDR
88(58)	MMNP(4) MMF NEW PSW
92(5C)	MMNL(4) MMF NEW LOC
96(60)	ETA(8) 64 -BIT TASK EXECUTION TIME ACCUM.

AUXILIARY PROCESSOR BLOCK (APB) continued

104(68)	(4) RESERVED FOR FUTURE MICROCODE USE
108(6C)	XFLG(2) EXTRA FLAGS HALFWORD
110(6E)	MREV(2) MICROCODE REVISION LEVEL
112(70)	RCVQ(8) PASSBACK QUEUE HEADER
120(78)	ISRS(2) LOCK FIELD FOR APU/CPU COORD.
122(7A)	QUE (1) CURRENT QUEUE NUMBER
123(7B)	QPF (1) QUEUE NUMBER AT POWER FAIL
124(7C)	STAT(2) APU SOFTWARE STATUS
126(7E)	OPT (2) APU OPTIONS
128(80)	CTLR(2) APU CONTROLLER NUMBER (RCVR)
130(82)	CCB (2) APU CCB PTR
132(84)	WTID(2) TCB.ID CAUSING APU WAIT
134(86)	CTID(2) TCB.ID W/CONTROL RIGHTS
136(88)	WCS (16) FD OF WCS IMAGE FILE
152(98)	EVN (60) GENERAL PURPOSE APU EVN LEAF
212(D4)	PSTD(4) PRIV SEG TABLE-APB.PWRF
216(D8)	SSEG(4) SHARED SEG TABLE-APB.PWRF
220(DC)	PSW (4) PSW-APB.PWRF

AUXILIARY PROCESSOR BLOCK (APB) continued

224 (E0)	LOC (4) LOC COUNTER-APB.PWRF
228 (E4)	GRS0 (64) GEN REG SET 0-APB.PWRF
292 (124)	GRS1 (64) GEN REG SET 1-APB.PWRF
356 (164)	GRS2 (64) GEN REG SET 2-APB.PWRF
420 (1A4)	GRS3 (64) GEN REG SET 3-APB.PWRF
484 (1E4)	GRS4 (64) GEN REG SET 4-APB.PWRF
548 (224)	GRS5 (64) GEN REG SET 5-APB.PWRF
612 (264)	GRS6 (64) GEN REG SET 6-APB.PWRF
676 (2A4)	GRSF (64) GEN REG SET F-APB.PWRF
740 (2E4)	SPAD (64) SCRATCHPAD REGS-APB.PWRF
804 (324)	SPFP (32) SINGLE PREC FP REGS-APB.PWRF
836 (344)	DPFP (64) DOUBLE PREC FP REGS-APB.PWRF

CHANNEL CONTROL BLOCK (CCB)

0(0)	CCW (2) CHANNEL COMMAND WORD	2(2)	LB0 (2) LENGTH OF BUFFER 0
4(4)	EB0 (4) END ADDR OF BUFFER 0		
8(8)	CW (2) 16-BIT CHECK 'WORD'	10(A)	LB1 (2) LENGTH OF BUFFER 1
12(C)	EB1 (4) END ADDR OF BUFFER 1		
16(10)	XLT (4) ADDR OF TRANSLATION TABLE		
20(14)	SUBA(2) ADDR OF SUBROUTINE (PURE CODE)	22(16)	MISC(1)
		23(17)	FLGS(1)
24(18)	DCB (4) ASSOCIATED DCB ADDRESS		
28(1C)	XLT2(4) SECONDARY TRANSLATE TABLE		

CONTEXT BLOCK (CTX)

0(0)	FPTR(4) FORWARD CTX POINTER
4(4)	BPTR(4) BACKWARD CTX POINTER
8(8)	TCB (4) TCB POINTER
12(C)	TRCE(4) RETURN LINK FROM STATE SWITCH
16(10)	PSW (8) OLD PSW SAVE AREA
24(18)	PSTD(4) PROCESS SEGMENT TABLE DESCR.
28(1C)	REGS(64) GENERAL REGISTER SAVE AREA
92(5C)	SCRG(64) SCRATCH PAD REGISTER SAVE AREA
156(9C)	SFLT(32) SGL PRECISION FLT REGS SAVE AREA
188(BC)	DFLT(64) DBL PRECISION FLT REGS SAVE AREA

DEVICE CONTROL BLOCK DEPENDENT SEGMENT (DDCB)

0(0)		(184)	
THE DEVICE INDEPENDENT SEGMENT			
184(B8)	CCB (2) CCB POINTER	186(BA)	(2) UNUSED
188(BC)	SDN (2) SELCH DEVICE #	190(BE)	CDN (2) CONTROLLER (DISC) DEVICE #
192(C0)	FCB (4) FILE CONTROL BLOCK CHAIN POINTER		
196(C4)	DIRP(4) DIRECTORY POINTER		
200(C8)	DRHA(4) HEAD OF DIRECTORY ADDR (LOGICAL END)		
204(CC)	BITP(4) BIT MAP POINTER		
208(D0)	STRK(2) SECTORS/TRACK	210(D2)	TCYL(2) TRACKS/CYLINDER
212(D4) BMFN(1)	213(D5) BMLU(1) BIT MAP-LOGICAL UNI	214(D6) BMST(1) BIT MAP-STATUS	215(D7) BMDN(1) BIT MAP-DEVICE NUMB
216(D8)	BMSA(4) BIT MAP STARTING ADDR		
220(DC)	BMEA(4) BIT MAP ENDING ADDR		
224(E0)	BMRA(4) BIT MAP RANDOM ADDR		

DEVICE CONTROL BLOCK DEPENDENT SEGMENT (DDCB) continued

228(E4)	BMLX(4) BIT MAP LENGTH LAST TRANSFER		
232(E8)DRFN(1) DIRECTORY-FC	233(E9)DRLU(1) DIRECTORY LU	234(EA)DRST(1) DIRECTORY STATUS	235(EB)DRDN(1) DIRECTORY DEVICE NUMBER
236(EC)	DRSA(4) DIRECTORY STARTING ADDR		
240(F0)	DREA(4) DIRECTORY ENDING ADDR		
244(F4)	DRRA(4) DIRECTORY RANDOM ADDR		
248(F8)	DRLX(4) DIRECTORY LAST TRANSFER		
252(FC)	BITL(4) BIT MAP LEAF ADDR		
256(100)	DIRL(4) DIRECTORY LEAF ADDR		
260(104)	DSC (4) DRIVER SECONDARY ENTRY		
264(108)	CSEC(4) CURRENT SECTOR		
268(10C)	VINC(4) VOLUME INTERCEPTION POINTER		
272(110)	BUF(256) DIRECTORY BUFFER		

DEVICE CONTROL BLOCK DEPENDENT SEGMENT (DCB) continued

528(210)		BUFB(256) BITMAP BUFFER	
784(310)		RBA (4)	1470G
788(314)		ACB (4) DIRECTORY ACCESS CONTROL BLOCK PTR	
792(318)		WAPB(4) WRITE ACCESS BITMAP POINTER	
796(31C)	LRBA(2) LOWEST RELATIVE BIT MAP SCTR 1470G	798(31E)	LSCT(2) LOWEST SECTOR WITHIN LRBA 1470G
800(320)	FIRP(2) FIRST PAGE OVER PAGE SPAN	802(322)	FIRB(2) FIRST BIT OF PG OVER PG SPAN
804(324)	LASP(2) LAST PAGE OVER PAGE SPAN	806(326)	LASB(2) LAST BIT OF PG OVER PG SPAN
808(328)	LSEC(2) LAST SECTOR USED IN LAST ALLOC	810(32A)	(2) FILLER
812(32C)		LALC(4) # OF SECTORS REQUESTED IN LAST ALLOC	
816(330)		MAST(4) MULTIPLE DCB SUPPORT	
820(334)		NEXT(4)	
824(338)	CNT (2)	826(33A)	NUM (2)

DEVICE CONTROL BLOCK DEPENDENT SEGMENT (DDCB) continued

828(33C)	BITB (4) BIT MAP AREA ADDRESS OF BIT MAP
832(340)	WRTR (4) # WRITES REQUESTED
836(344)	RDR (4) # READS REQUESTED
840(348)	WRTD (4) # SECTORS WRITTEN
844(34C)	RDD (4) #SECTORS READ
848(350)	GOBY (8) COUNT OF SECTORS TO MINIMIZE
856(358)	SEEK (4) NUM OF SEEKS DONE
860(35C)	XFNM(12) MUTATED FILENAME (SVC7)

PRIMARY FILE DIRECTORY (DIR)

0(0)	FNM (8) FILE NAME		
8(8)	EXT (3) EXTENTION	11(B)ACT (1) ACCOUNT NUMBER	
12(C)	FLBA(4) FIRST LOGICAL BLOCK ADDR		
16(10)	LLBA(4) LAST LOGICAL BLOCK ADDR		
20(14)WKEY(1) WRITE KEY	21(15)RKEY(1) READ KEY	22(16)	LRCL(2) LOGICAL RECORD LENGTH
24(18)	DATE(4) DATE FILE ALLOCATED		
28(1C)	LUSE(4) DATE FILE LAST WRITTEN		
32(20)	WCNT(2) WRITE COUNT	34(22)	RCNT(2) READ COUNT
36(24)ATRE(1) ATTRIBUTES	37(25)BKSZ(1) BLOCK SIZE	38(26)INBS(1) INDEX BLOCK SIZE	39(27)SHDS(1) SHARED DISC SUPPORT (BODENS
40(28)	CSEC(4) CURRENT SECTOR /# LOGICAL RECORDS		
44(2C)	LASN(4) DATE FILE LAST ASSIGNED		

DEVICE MNEMONIC TABLE (DMT)

0(0)	(4) NAME IN ASCII
4(4)	(4) DCB ADDRESS
8(8)	(4) NAME IN ASCII
12(C)	(4) DCB ADDRESS
16(10)	(4) NAME IN ASCII
20(14)	(4) DCB ADDRESS
24(18)	(4) NAME IN ASCII
28(1C)	(4) DCB ADDRESS
32(20)	(8) DOUBLE WORD OF ZERO TO END TABLE

EVENT COORDINATION NODE (EVN)

0(0)	CORD(4) UPPER POINTER	
4(4)	FLGS(2) FLAGS - MUST MATCH QCB.FLGS	6(6) LOCK(2) T&S LOCK FOR MULTIPROCESSING
8(8)	(2) SPARE	10(A)CLEV(1) 11(B)TSIZ(1) CONNECTION LEVEL TREE SIZE
12(C)	SQS (4) SQS EXECUTOR	
16(10)	DCB (4) DCB ADDRESS	
20(14)	TCB (4) TCB ADDRESS	
24(18)	QCB (4) Pointer to current QCB	
28(1C)	EVR(4) SAVE AREA FOR EVREL	
32(20)	NIO (2) COUNT OF I/O'S (CONN. + QUEUED)	34(22) HWIO(2) HWM OF EVN.NIO
36(24)	CLC (4) CONNECTED LEAF CHAIN	
40(28)	PREV(4) PREVIOUS (NODE) POINTER	
44(2C)	NEXT(4) NEXT (NODE) POINTER	

EVENT COORDINATION NODE (EVN) continued

48(30) TOP (4) TOP OF WAITING QUEUE		
52(34) BOT (4) BOTTOM OF WAITING QUEUE		
56(38) CYL (2) CURRENT CYLINDER POSITION	58(3A)RDCT(1) REDISPATCH COUNT	59(3B) (1) RESERVED

FILE CONTROL BLOCK (FCB)

0(0)	VMT (4) VOLUME MNEMONIC TABLE ADDRESS	
4(4)	LEAF(4) LEAF ADDRESS	
8(8)	WCNT(2) WRITE COUNT	10(A) RCNT(2) READ COUNT
12(C)	FLGS(4) FLAGS FIELD	
16(10)	1INC(4) SVC 1 DEVICE INTERCEPT	
20(14)	7INC(4) SVC 7 DEVICE INTERCEPT	
24(18)	OFF(1) BYTE OFFSET INTO DIRECTORY SECTO	25(19)DCOD(1) DEVICE CODE (DCB NUMBER)
		26(1A) DN (2) DEVICE NUMBER (PHYSICAL ADDRESS)
28(1C)	ATRB(2) DEVICE ATTRIBUTES	30(1E) RECL(2) RECORD LENGTH
32(20)	INIT(4) DRIVER INITIALIZATION FOR I/O	
36(24)	FUNC(4) DRIVER INITIALIZATION FOR COMMANDS	
40(28)	TERM(4)	
44(2C)	IOBF(4) FMCO IOB QUEUE HEAD	

FILE CONTROL BLOCK (FCB) continued

48(30)	IOBL(4) FMCO IOB QUEUE TAIL	
52(34)	IOCN(2) FMCO IOB QUEUED COUNT	54(36) BKSZ(2) DATA BLOCK SIZE IN SECTORS
56(38)	INBS(2) INDEX BLOCK SIZE IN SECTORS	58(3A) (2) RESERVED
60(3C)	(4) RESERVED	
64(40)	(2) RESERVED	66(42) CLAS(2) FILE CLASS FOR ACCOUNTING
68(44)	IOH (4) PTR TO IOH LIST (DEFAULT IF ZERO)	
72(48)	(1) spare	73(49) LEVL(1) specifies level for ADDSQ
		74(4A) IOP (2) index of IOP in APBDIR (0 for CPU)
76(4C)	IPB (4) Pointer to IPB for this IOP	
80(50)	SQ (4) Pointer to level 6 SIQ or SQ if CPU	
84(54)	Q (4) QUEUE STRATEGY ROUTINE (DEFAULT IF Z	
88(58)	(8) (RESERVED)	
96(60)	IOB (80) INPUT/OUTPUT BLOCK	

FILE CONTROL BLOCK (FCB) continued

176(B0)	(8)	END OF DEVICE INDEPENDENT SECTION
184(B8)	NAME(8)	FILE NAME
192(C0)	EXT (3)	EXTENSION
		195(C3)ACT (1)
		ACCOUNT NUMBER
196(C4)	DATE(4)	DATE LAST WRITTEN ON ASSIGN
200(C8)	FLBA(4)	FIRST LOGICAL BLOCK ADDR
204(CC)	LLBA(4)	LAST LOGICAL BOOCK ADDR
208(D0)	DIR (4)	DIRECTORY BLOCK POINTER
212(D4)	FCB (4)	FCB LINKAGE FIELD
216(D8)	CSEC(4)	CURRENT SECTOR LOGICAL BLOCK ADDR
220(DC)	BUFF(4)	I/O BUFFER FOR FMCO
224(E0)	CTX (92)	RSA CONTEXT BLOCK
316(13C)	PFCB(4)	POINTER TO CURRENT PFCB

FILE CONTROL BLOCK (FCB) continued

320(140)	RING (4) POINTER TO PFCB RING	
324(144)	SEQ (4) I/O SEQUENCE COUNTER	
328(148)	NBLK(4) NUMBER OF DATA BLOCK	[1,N]
332(14C)	NINB(4) NUMBER OF INDEX BLOCKS	[1,N]
336(150)	NLR (4) NUMBER OF LOGICAL RECORDS	[1,N]
340(154)	SPD (4) PSEUDO DEVICE NAME	
344(158)	SPSZ(2) PSEUDO DEVICE RECL	346(15A) HFLG(2) HISTORY UPDATE FLAG
348(15C)	HCNT(2) NUMBER OF HISTORY ENTRIES	350(15E) HPTR(2) CURRENT OFFSET INTO HISTORY
352(160)	DBLS (4) LOOK-AHEAD WORK AREA	
356(164)	HIST(64) LOOK-AHEAD HISTORY DATA	
420(1A4)	DIFF(64) LOOK-AHEAD HISTORY DIFFERENCE	
484(1E4)	SWL (60) FCB LEAF (EVN STRUCTURE)	

FILE CONTROL BLOCK (FCB) continued

544(220)

SAVE(64)
REGISTER SAVE AREA

FILE DESCRIPTOR (FD)

0(0)	VOL (4) VOLUME	
4(4)	FNM (8) FILE NAME	
12(C)	EXT (3) EXTENSION	15(F) (1) ACCOUNT NUMBER LOW BYTE

FREE BLOCK DESCRIPTOR (FDE)

0(0)	PTR (4) LINK POINTER
4(4)	SIZE(4) SIZE FIELD

FMIN EXTENDED SVCL PARAMETER BLOCK

0(0)	(20) STANDARD SVC 1 PARAMETER BLOCK
20(14)	TCB (4) A(TCB) of requestor
24(18)	BLK (4) logical block #
28(1C)	VLD (4) buffer valid flag
32(20)	CNT (4) access count for this buffer
36(24)	SEQ (4) sequence number of last reference

INTERCEPT CONTROL BLOCK (ICB.)

0(0)	EQ (4) IRCB EXECUTING CHAIN
4(4)	PQ (4) IRCB PENDING CHAIN
8(8)	PSW (8) SAVED PSW
16(10)	FLIH(4) LINK TO SVC FIRST LEVEL HANDLER
20(14)	IPCB(4) SYSTEM TASK IPCB CHAIN
24(18)	LINK(4) LINK TO SAVE TCBS WAITING FOR IRCB
28(1C)	HEAD(4) IPCB TCB LINKED TO BY ICB.LINK
32(20)	0CL (4) SVC 0 CALLER INTERCEPT CHAIN
36(24)	1CL (4) SVC 1 CALLER INTERCEPT CHAIN
40(28)	27CL(4) SVC 2.7 CALLER INTERCEPT CHAIN
44(2C)	3CL (4) SVC 3 CALLER INTERCEPT CHAIN
48(30)	6CL (4) SVC 6 CALLER INTERCEPT CHAIN

INTERCEPT CONTROL BLOCK (ICB.) continued

52(34)	7CL (4)	SVC 7 CALLER INTERCEPT CHAIN	
56(38)	6RX (4)	SVC 6 RECIPIENT INTERCEPT CHAIN	
60(3C)	RCNT(2)	62(3E)	FLGS(2)
COUNT OF NON-ROLLABLE INTERCEPTS		FLAG FIELD	

INPUT/OUTPUT BLOCK (IOB)

0(0)	NXT (4) FORWARD POINTER		
4(4)	RFLG(2) REQUEST DEPENDENT FLAG	6(6)PRI (1) I/O PRIORITY	7(7)TYPE(1) TYPE
8(8)	DONE(4) IODONE/SUB EXECUTOR ADDRESS		
12(C)	DCB (4) DCB ADDRESS		
16(10)	TCB (4) CALLER TCB POINTER		
20(14)	QCB (4) Pointer to QCB for this IOB		
24(18)	ESR (4) DRIVER ENTRY (DIR,ESR)		
28(1C)	UPBK(4) UNRELOCATED PARM BLOCK ADDRESS		
32(20)	PBLK(4) RELOCATED PARM BLOCK ADDRESS		
36(24)FC (1) SVC 1 FUNCTION CODE	37(25)LU (1) LU NUMBER(DEFINED	38(26)STAT(1) STATUS - ZERO	39(27)DDPS(1) DEVICE DEPENDENT STATU
40(28)	SADR(4) BUFFER START ADDRESS		
44(2C)	EADR(4) BUFFER ENDING ADDRESS		

INPUT/OUTPUT BLOCK (IOB) continued

48(30)	RAND(4) POSITIONAL ADDRESS	
52(34)	LXF (4) Length of transfer	
56(38)	SV1X(4) EXTENDED SVC 1 WORD	
60(3C)	LUE (4) LU ENTRY	
64(40)	WCHN(4) TASKS WAITING FOR THIS I/O TO FINISH	
68(44)	CYL (2) REQUESTED CYLINDER * 2, FOR DISC	70(46)SECT(1) 71(47)LSEC(1) STARTING RELATIVE POS OF RELATIVE SECTOR LAST SECTOR OF
72(48)	(2) RESERVED	74(4A) (2) RESERVED
76(4C)	(4) RESERVED	

INPUT/OUTPUT HANDLER ROUTINES (IOH)

0(0)	READ(4) READ EXECUTOR (SVC 1)
4(4)	WRIT(4) WRITE EXECUTOR (SVC 1)
8(8)	WAIT(4) WAIT-ONLY EXECUTOR (SVC 1)
12(C)	HALT(4) HALT I/O EXECUTOR (SVC 1)
16(10)	TEST(4) TEST I/O COMPLETION EXECUTOR (SVC 1)
20(14)	SET (4) TEST & SET EXECUTOR (SVC 1)
24(18)	REW (4) REWIND EXECUTOR (SVC 1)
28(1C)	BSR (4) BACKSPACE RECORD EXECUTOR (SVC 1)
32(20)	FSR (4) FORWARD SPACE RECORD EXECUTOR (SVC 1)
36(24)	WFM (4) WRITE FILE MARK EXECUTOR (SVC 1)
40(28)	FFM (4) FORWARD FILE MARK EXECUTOR (SVC 1)
44(2C)	BFM (4) BACKSPACE FILE MARK EXECUTOR (SVC 1)

INPUT/OUTPUT HANDLER ROUTINES (IOH) continued

48(30)	EOT (4) TASK TERMINATION EXECUTOR (SVC 3)
52(34)	INIT(4) DEVICE INITIALIZATION (SYSINIT)
56(38)	DDF (4) DEVICE DEPENDENT FUNCTION EXECUTOR
60(3C)	CON (4) SPECIAL ENTRY FOR OS CONSOLE
64(40)	PWR (4) POWER RESTORE INITIALIZATION

IOP PROCESSOR BLOCK (IPB)

0(0)	(8)	UNUSED
8(8)	CTCB(4)	UNUSED BY IOP (APB.CTCB)
12(C)	FLGS(2)	14(E)MODE(1) 15(F)ID (1)
	UNUSED BY IOP (APB.FLGS)	UNUSED BY IOP IOP SEQUENCE (APB.MODE) IN APBDIR
16(10)	HRQP(4)	UNUSED BY IOP (APB.HRQP)
20(14)	SSTD(4)	UNUSED BY IOP (APB.SSTD)
24(18)	PRCA(4)	APB PASSBACK REASON CODE (APB.PRCA)
28(1C)	PFSP(4)	IOP POWER FAIL SAVE POINTER
32(20)	XOP(4)	UNUSED BY IOP (APB.XOP)
36(24)	XOL(4)	UNUSED BY IOP (APB.XOL)
40(28)	XDEV(4)	UNUSED BY IOP (APB.XDEV)
44(2C)	XSTA(4)	UNUSED BY IOP (APB.XSTA)
48(30)	XNL(4)	UNUSED BY IOP (APB.XNL)

IOP PROCESSOR BLOCK (IPB) continued

52(34)	ISPS(4) IOP START OF ISP
56(38)	ISPE(4) POINTER TO Q/W AFTER ISP END
60(3C)	ARQP(4) UNUSED BY IOP (APB.ARQP)
64(40)	PRCB(4) PRC INFORMATION (APB.PRCB)
68(44)	PRCC(4) PRC INFORMATION (APB.PRCC)
72(48)	PRCD(4) PRC INFORMATION (APB.PRCD)
76(4C)	PRCE(4) PRC INFORMATION (APB.PRCE)
80(50)	PRCF(4) PRC INFORMATION (APB.PRCF)
84(54)	STRC(4) SYSTEM DATA STRUC ERROR (PHYS) ADDR (APB.STRC)
88(58)	MMNP(4) MMF NEW PSW (APB.MMNP)
92(5C)	MMNL(4) MMF NEW LOC (APB.MMNL)
96(60)	ETA (8) UNUSED BY IOP (APB.ETA)

IOP PROCESSOR BLOCK (IPB) continued

104(68)	(4)	RESERVED FOR FUTURE MICROCODE USE
108(6C)	XFLG(2)	EXTRA FLAGS HALFWORD
110(6E)	MREV(2)	MICROCODE REVISION LEVEL
112(70)	RCVQ(8)	PASSBACK QUEUE HEADER (APB.RCVQ)
120(78)	ISRS(2)	LOCK FIELD FOR APU/CPU COORD.
122(7A)	QUE (1)	CURRENT QUEUE NUMBER
123(7B)	QPF (1)	QUEUE NUMBER AT POWER FAIL
124(7C)	STAT(2)	STATUS FIELD TO DISTINGUISH APU/IOP
126(7E)	OPT (2)	APU OPTIONS (APB.OPT)
128(80)	CTLR(2)	ADDRESS OF CPU SIDE OF RTSM
130(82)	CCB (2)	ADDRESS OF RTSM CCB
132(84)	WTID(2)	TCB.ID CAUSING APU WAIT (APB.WTID)
134(86)	CTID(2)	TCB.ID W/CONTROL RIGHTS (APB.CTID)
136(88)	WCS (16)	FD OF WCS IMAGE FILE (APB.WCS)
152(98)	EVH (60)	EMBEDED EVNPFOR RTSM/IOP/APU
212(D4)	PSTD(4)	PRIV SEG TABLE-IPB.PWRF
216(D8)	SSEG(4)	SHARED SEG TABLE-IPB.PWRF
220(DC)	PSW (4)	PSW-IPB.PWRF

IOP PROCESSOR BLOCK (IPB) continued

224 (E0)	LOC (4) LOC COUNTER-IPB.PWRF
228 (E4)	GRS0 (64) GEN REG SET 0-IPB.PWRF
292 (124)	GRS1 (64) GEN REG SET 1-IPB.PWRF
356 (164)	GRS2 (64) GEN REG SET 2-IPB.PWRF
420 (1A4)	GRS3 (64) GEN REG SET 3-IPB.PWRF
484 (1E4)	GRS4 (64) GEN REG SET 4-IPB.PWRF
548 (224)	GRS5 (64) GEN REG SET 5-IPB.PWRF
612 (264)	GRS6 (64) GEN REG SET 6-IPB.PWRF
676 (2A4)	GRSF (64) GEN REG SET F-IPB.PWRF
740 (2E4)	SPAD (64) SCRATCHPAD REGS-IPB.PWRF
804 (324)	SPFP (32) SINGLE PREC FP REGS-IPB.PWRF
836 (344)	DPFP (64) DOUBLE PREC FP REGS-IPB.PWRF

IOP PROCESSOR BLOCK (IPB) continued

900(384)	FLGS(2) Flags	902(386)	RTRY(2) Health Monitor Retry Counter
904(388)	CALL(2) Caller number for sanity check	906(38A)	ERR7(2) Count of references to level 7
908(38C)	ACCL(1) Current Accounting Level	909(38D)	(1) alignment
		910(38E)	IOSC(2) Dev address of CPU side I/O Switch
912(390)			IOBS(4) Pointer to IOB (save area)
916(394)			ADCK(4) Save area for ADCHKNS/TOCHON
920(398)			DCB (4) Parameter from SQSDISP to DIRDONE
924(39C)			DTHD(4) Pointer Head of device timers chain
928(3A0)			HLTQ(4) Pointer to halt list (list of IOBs)
932(3A4)			IQCB(4) SPARE QCB FOR IOP
936(3A8)			CQCB(4) SPARE QCB FOR CPU
940(3AC)			JRNL(4) Pointer to journal for IOP # n
944(3B0)			SIQ4(4) Address Level 4 System Queue

IOP PROCESSOR BLOCK (IPB) continued

948(3B4)	SIQ5(4) Address Level 5 System Queue
952(3B8)	SIQ6(4) Address Level 6 System Queue
956(3BC)	SIQ7(4) Address Level 7 System Queue
960(3C0)	QCB4(4) Pointer to current QCB for level 4
964(3C4)	QCB5(4) Pointer to current QCB for level 5
968(3C8)	QCB6(4) Pointer to current QCB for level 6
972(3CC)	QCB7(4) Pointer to current QCB for level 7
976(3D0)	QMX4(2) Threshold level 4 SQ warning
978(3D2)	QMX5(2) Threshold level 5 SQ warning
980(3D4)	QMX6(2) Threshold level 6 SQ warning
982(3D6)	QMX7(2) Threshold level 7 SQ warning
984(3D8)	SYPL(8) PSW/LOC for GETSYP, RELSYP
992(3E0)	RQLK(2) LOCK FOR ACCESS TO IPB.RCVQ
994(3E2)	SQLK(2) LOCK FOR ACCESS TO IPB.SENQ
996(3E4)	RCVQ(4) Address Receive Dlist - IOP to CPU

IOP PROCESSOR BLOCK (IPB) continued

1000(3E8)	SENQ(4) Address of Send Dlist - CPU to IOP
1004(3EC)	LCKD(4) Pointer to locked data structure
1008(3F0)	QCBS(4) SANITY QCB
1012(3F4)	SAV4(64) Save area (level 4)
1076(434)	SAV5(64) Save area (level 5)
1140(474)	SAV6(64) Save area for File Manager (level 6)
1204(4B4)	SAV7(64) Save area (level 7) {spare}
1268(4F4)	QCBE(64) Emergency QCB
1332(534)	MRSV(64) MIRRORED DISK SAVE AREA
1396(574)	SWSV(40) MIRR DISK SWOP SAVE AREA
1436(59C)	(4) SPARE
1040(5A0)	ICUM(8) IDLE COUNT ACCUMULATOR

IOP PROCESSOR BLOCK (IPB) continued

1048(5A8)

IIOR(4)
I/O REQUESTS

1052(5AC)

NIOR(4)
NON-I/O REQUESTS

INTERCEPT PATH CONTROL BLOCK (IPCB.)

0(0)	IEC (4) IPCB ENTITY CHAIN	
4(4)	IE (4) POINTER TO INTERCEPTED ENTITY	
8(8)	IEH (4) POINTER TO HEAD OF ENTITY CHAIN	
12(C)	STC (4) SYSTEM TASK IPCB CHAIN	
16(10)	TCB (4) TCB ADDR OF SYSTEM TASK	
20(14)	EXE (4) INTERCEPT HANDLER ROUTINE	
24(18)	BFQ (4) ADDR OF BUFFER QUEUE	
28(1C)	QH (4) ADDR OF QUEUE HANDLER	
32(20)	SVR (4) SYSTEM VARIABLE DATA	
36(24)	EIN (2) NUMBER OF EXECUTING IRCBS	38(26) LEN (2) LENGTH OF IPCB AND IRCBS
40(28)	SIZ (2) SIZE OF IRCBS	42(2A) FLG (2) FLAG FIELD
44(2C)	TYP (2) IPCB SVC INTERCEPT TYPE FLAG FIELD	46(2E) PLN (2) SVC PARAMETER BLOCK LENGTH

INTERCEPT PATH CONTROL BLOCK (IPCB.) continued

48(30)	FCH (4) IRCB FREE CHAIN HEAD
52(34)	TC (4) HEAD OF TCB CHAIN WAITING FOR IRCBS
56(38) OFF (2) OFFSET TO PARM BLOCK IN RDB	58(3A) PID (2) INTERCEPT PATH ID
60(3C) TCT (2) # OF OUTSTANDING TERMINATION EVNTS	62(3E) BQL (2) BUFFER (RDB) QUEUE LENGTH

INTERCEPT REQUEST CONTROL BLOCK (IRCB.)

0(0)	CHN (4) IRCB FREE CHAIN	
4(4)	IPC (4) POINTER TO IPCB	
8(8)	IC (4) IRCB EXECUTING OR PENDING CHAIN PTR	
12(C)	PBL (4) ADDR OF PARAMETER BLOCK	
16(10)	SAD (4) ADDR OF SVC INST CAUSING INTERCEPT	
20(14)	RDB (4) ADDR OF RDB IN SYSTEM TASK	
24(18)	PID (2) IRCB PATH ID	26(1A) RID (2) IRCB QUEUE HANDLER ID
28(1C) (1) FILLER	29(1D) PRI (1) TASK PRIORITY	30(1E) FLG (2) FLAG FIELD
32(20)	TID (4) TASK ID OF INTERCEPTED TASK	
36(24)	PB (?) ? IS SIZE OF SVC'S PARAMETER BLOCK	

INITIAL VALUE TABLE (IVT)

0(0)	CSL (4)	
4(4)	MXBX(4) BGRD LIMS:UB MXPRI,3BYTES MAX SYSS	
8(8)	TCMS(4) TASK COMMON SIZE	
12(C)	SYSS(4) SIZE OF SYSTEM SPACE	
16(10)	ERDV(4) ERROR REC. DEFAULT VOLUME	
20(14)	ERDN(8) ERROR REC. DEFAULT FILE NAME	
28(1C)	ERDX(3) ERROR REC. DEFAULT FILE EXTENSION	31(1F)ERDC(1) ERROR REC. DEFAULT FILE CL
32(20)	ERDS(4) ERROR REC. DEFAULT FILE SIZE	
36(24)	ERDP(4) ERROR REC. DEFAULT MER PERIOD	
40(28)	PWF (2) POWER FAIL MODE AUTO=0,MAN=1	42(2A) DLAY(2) DELAY IN SECONDS
44(2C)	OPT (4) IVT OPTIONS WORD	

LOADER COMMON ENTRY (LCE)

0(0)	NAME(8) NAME OF COMMON/ENTRY
8(8)	SIZE(4) SIZE IF COMMON BLOCK
12(C)	ADDR(4) ADDRESS OF COMMON/ENTRY

LOADER INFORMATION BLOCK (LIB)

0(0)SEGT(1) SEGMENT TYPE	1(1)NLIB(1) # OF LIB'S	2(2)NLU (1) # OF LU'S	3(3)NASN(1) # OF ASSIGNS
4(4)MPRI(1) MAXIMUM PRIORITY	5(5)IPRI(1) INITIAL PRIORITY	6(6)PREG(1) PURE SEGMENTATION RE	7(7)NCOM(1) # OF TASK COMMON SEG'S RE
8(8)NLSG(1) # OF LIBRARY SEG'S REQ'D	9(9)TATR(1) TASK COMMON ATTRIBUTES	10(A) OPT (2) TASK OPTIONS	
12(C)		SEGS(4) SIZE OF SEGMENT IN SECTORS	
16(10)		FW2 (4) OVERLAY START, NUM ENTRYS, OR BLK	
20(14)		FW3 (4) OVERLAY END OR MAX SYSTEM SPACE	
24(18)		ITSW(8) INITIAL TSW	
32(20)		SEGN(12) SEGMENT NAME	
44(2C)		DUMN(16) DUMP FILE OR DUMMY NAME	
60(3C) NPRC(2) NUMBER OF PATCH RECORDS		62(3E) SPRC(2) STARTING PATCH RECORD NUMBER	
64(40) NLVL(2) NUMBER OF OVERLAY LEVELS		66(42) NOVL(2) NUMBER OF OVERLAY NODES	
68(44)MXSG(1) HIGHEST SEGMENTATION RE		69(45) (3) RESERVED	

LOADER INFORMATION BLOCK continued

72(48) ORN (2) REC # OF OVLY DESC TAB FOR TASK	74(4A) IRN (2) REC # OF IMPURE SEGMENT
76(4C) NEPC(2) # OF LSE ENTRIES AT VARL	78(4E) (2) (RESERVED)
80(50)	DATE(8) DATE WHEN TETTED
88(58)	TIME(8) TIME WHEN TETTED
96(60) NIOQ(2) MAX QUEUED I/O REQUESTS (UNBLKED)	98(62) ODTL(2) LENGTH OF OVLY DESC TAB IN BYTES
100(64)	CTOP(4) CTOP
104(68)	UTOP(4) UTOP
108(6C)	VARL(148) LIST OF ENTRIES IN RTL SEGMENT

LOGICAL PROCESSOR MAPPING TABLE (LPMT)

0 (0)	HED (4) LPMT HEADER
--------	-------------------------

LOADER SHARABLE-SEGMENT ENTRY (LSE)

0(0)	NAME (8) SHARABLE SEGMENT NAME	
8(8)	(3) SEGMENT NAME CONTINUED	11(B)SREG(1) SEGMENTATION REGISTER NUMBER
12(C)PRIV(1) ACCESS PRIVILIGES	13(D)	MNSZ(3) MINIMUM SIZE
16(10)	MXSZ(4) MAXIMUM SIZE	

LOADSEG PARAMETER BLOCK (LSG)

0(0)	FUN (2) FUNCTION CODE.	2(2)LU (1) LOAD LU.	3(3)STA (1) ERROR STATUS.
4(4)	SIZ (4) SEGMENT SIZE.		
8(8)	NAM (8) SEGMENT NAME.		
16(10)	SV1 (20) SVC 1 PARAMETER BLOCK		

LOADER'S TASK CONTROL BLOCK (LTCB)

0(0)	(744)	STANDARD TASK CONTROL BLOCK
744(2E8)	LU0 (4)	LU 0 ENTRY OF STANDARD TASK CONTROL BLOCK
748(2EC)	LU1 (4) LU 1	
752(2F0)	RTL (4)	POINT TO RTL WORKING AREA
756(2F4)	SAV (64)	REGISTERS SAVE AREA
820(334)	SV1 (20)	SVC 1 PARAMETER BLOCK
840(348)	FUN (4)	SPECIFIED SVC 6 FUNCTION CODES
844(34C)	PBA (4)	UNRELOCATED SVC 6 PARMBLK ADR
848(350) ID (1)	849(351) LU (1)	850(352) OPT (2)
CALLER'S TASK ID	LOAD LU (WAIT)	SPECIFIED X-LOAD OPTIONS
852(354)	SIZ (4)	SPECIFIED SIZE INCREMENT, BYTES
856(358)	IMP (4)	IMPURE SDE POINTER
860(35C)	PUR (4)	PURE SDE POINTER

LOADER'S TASK CONTROL BLOCK (LTCB) continued

864(360)

LIB(256)
LIB BUFFER

OVERLAY CONTROL BLOCK (OCB)

0(0)	ODT (4) ADDRESS OF OVLY DESC TABLE	
4(4)	NOVL(2) NUMBER OF OVLERLAY NODES	6(6) NLVL(2) NUMBER OF OVERLAY LEVELS
8(8)	RORT(4) ADDR OVLY REF TAB IN ROOT	
12(C)	OVLU(4) ADDR OF OVLY FILE FCB	
16(10)	SVC1(20) LOAD OVLY SVC 1 PARM BLOCK	
36(24)	MSG (44) ERR MSG BUFF FOR SVC10	

OVERLAY DESCRIPTOR TABLE (ODT)

0(0)	NAME(8) OVERLAY NAME	
8(8)	AREA(4) OVERLAY AREA	
12(C)	(2) (RESERVED)	14(E) PODE(2) INDEX # OF PARENT NODE
16(10)	LEN (4) LENGTH OF OVERLAY (IN BYTES)	
20(14)	SECT(4) LOGICAL REC # OF OVERLAY	

OVERLAY REFERENCE TABLE HEADER (ORH)

0(0)	OORT(4) ADDR OF OVERLAY ORT ENTRIES
4(4)	END (4) ADDR OF LAST ORT ENTRY

OVERLAY REFERENCE TABLE (ORT)

0(0)	SVCA(6) INVOKING SVC 1 CALL
	6(6) ODTE(2) INDEX # OF OVLY DESC TAB ENTRY
8(8)	LEN (4) LENGTH OF SUB-OVLY
12(C)	AREA(4) ADDR OF SUB-OVLY ARE
16(10)	SECT(4) LOGICAL REC # OF SUB-OVLY

DEVICE CONTROL BLOCK INDEPENDENT PORTION (PDCB)

0(0)	DMT (4) ADDR OF THE DMT ENTRY	
4(4)	LEAF(4) ADDR OF THE EVT LEAF	
8(8)	WCNT(2) WRITE COUNT	10(A) RCNT(2) READ COUNT
12(C)	FLGS(4) FLAGS FIELD	
16(10)	1INC(4) SVC 1 DEVICE INTERCEPT	
20(14)	7INC(4) SVC 7 DEVICE INTERCEPT	
24(18)	(1)	25(19) DCOD(1) 26(1A) DN (2)
(RESERVED)		DEVICE CODE DEVICE NUMBER (PHYSICAL ADDRESS)
		(DCB NUMBER)
28(1C)	ATRB(2) DEVICE ATTRIBUTES	30(1E) RECL(2) RECORD LENGTH
32(20)	INIT(4) ADDR OF DRIVER INITIATION ROUTINE	
36(24)	FUNC(4) ADDR OF DRIVER FUNCTION ROUTINE	
40(28)	TERM(4) ADDR OF DRIVER TERMINATION ROUTINE	
44(2C)	TOUT(2) TIME-OUT CONSTANT	46(2E) RTRY(2) OPERATION RETRY COUNT

DEVICE CONTROL BLOCK INDEPENDENT PORTION (PDCB) continued

48(30)WKEY(1) WRITE KEY	49(31)RKEY(1) READ KEY	50(32) ILVL(2) DEVICE INTERRUPT LEVEL
52(34)	ERRL(4) ERROR LOG DATA AND ON/OFF SWITCH	
56(38)	ISP (4) PTR TO ISPTAB FOR THIS DEVICE	
60(3C)	TOCH(4) TIME-OUT CHAIN ADDR	
64(40) XFLG(2) DEVICE DEPENDENT FLAGS	66(42) CLAS(2) DEVICE CLASS FOR ACCOUNTING	
68(44)	IOH (4) PTR TO IOH LIST (DEFAULT IF ZERO)	
72(48) (1) spare	73(49)LEVL(1) Level for ADDSQ	74(4A) IOP (2) Index of IOP in APBDIR (0 for CPU)
76(4C)	IPB (4) Pointer to IPB for IOP	
80(50)	SQ (4) Pointer to processor's system queue	
84(54)	Q (4) QUEUE STRATEGY ROUTINE (DEFAULT IF 0	
88(58)	EDMA(4) EDMA STRATEGY ROUTINE	
92(5C)	(4) (RESERVED)	

DEVICE CONTROL BLOCK INDEPENDENT PORTION (PDCB) continued

96 (60)	IOB (80) INPUT/OUTPUT BLOCK
176 (B0)	SIZE (4) SIZE (SCTRS OR LINES)
180 (B4)	VFC (4) ADDRESS OF VFDCB

PRIVATE FILE CONTROL BLOCK (PFCB)

0(0)	LRN (4) CURRENT LOGICAL RECORD NUMBER	
4(4)	FCB (4) SHARED FCB POINTER	
8(8)	(4)	
12(C)	(4) (CAN NOT BE USED)	
16(10)	(8)	
24(18)	(1)	25(19) DCD (1) 26(1A) (2) DEVICE CODE
28(1C)	ATB (2) ATTRIBUTE	30(1E) SWO (2) SHARED-WRITE-ONLY FLAG
32(20)	PLF (4) RING POINTER TO NEXT PFCB	
36(24)	HDR (4) HEADER FOR SPOOL OUTPUT RECORD	
40(28)	IXX (4) NEXT FREE ENTRY IN INDEX BLOCK	
44(2C)	OFF (4) BYTE OFFSET INTO DATA BLOCK	
48(30)	SAD (4) ADJUSTED USER BFR START ADDR	

PRIVATE FILE CONTROL BLOCK (PFCE) continued

52(34)	EAD (4) ADJUSTED USER BFR END ADDR	
56(38)	SRT (4) FAKE USER BFR STRT ADR	
60(3C)	END (4) FAKE USER BFR END ADR	
64(40)	LOK (2) T&S LOCK FOR MULTIPROCESSING	66(42) CLS (2) FILE CLASS FOR ACCOUNTING
68(44)	IOH (4) IOH LIST	
72(48) (1) spare	73(49)LVL (1) specifies level for ADDSQ	74(4A) IOP (2) Index of IOP in APBDIR (0 for CPU)
76(4C)	IPB (4) Pointer to IPB for this IOP	
80(50)	SQ (4) Pointer to level 6 SIQ for this IOP	
84(54)	IXP (24) SVCL PARM BLOCK INDEX BUFFER	
108(6C)	IXB (4) BLOCK NUMBER - INDEX BUFFER	
112(70)	IXV (4) VALID FLAG - INDEX BUFFER	
116(74)	IXC (4) USE COUNT - INDEX BUFFER	

PRIVATE FILE CONTROL BLOCK (PFCB) continued

120(78)	IXS (4) I/O SEQUENCE - INDEX BUFFER
124(7C)	D1P (24) SVCL PARM BLOCK DATA BUFFER #1
148(94)	D1B (4) BLOCK NUMBER - DATA BUFFER #1
152(98)	D1V (4) VALID FLAG - DATA BUFFER #1
156(9C)	D1C (4) USE COUNT - DATA BUFFER #1
160(A0)	D1S (4) I/O SEQUENCE - DATA BUFFER #1
164(A4)	D2P (24) SVCL PARM BLOCK DATA BUFFER #2
188(BC)	D2B (4) BLOCK NUMBER - DATA BUFFER #2
192(C0)	D2V (4) VALID FLAG - DATA BUFFER #2
196(C4)	D2C (4) USE COUNT - DATA BUFFER #2
200(C8)	D2S (4) I/O SEQUENCE - DATA BUFFER #2

PSEUDO DEVICE CONTROL BLOCK (PSDCB)

0(0)	(32) DEVICE INDEPENDENT PART
32(20)	NODE(8) NODE NAME
40(28)	NAM (16) REST OF FILE NAME
56(38)	PSPT(4) PSEUDO-DCB CHAIN

PSEUDO TASK CONTROL BLOCK (PSTCB)

0 (0)	(196) STANDARD TASK CONTROL BLOCK
194 (C4)	NODE (8) NODE NAME FOR TASK
202 (CC)	NMI (8) REST OF TASK NAME
210 (D4)	PSPT (4) PSEUDO-TCB CHAIN

QUEUE CONTROL BLOCK (QCB)

0(0)	EXAD(4) Address of IOP request executor
4(4) FLGS(2) FLAGS - MUST MATCH EVN.FLGS	6(6) (2) SPARE
8(8) LEVL(2) SIQ level for QCB on xfer to IOP	10(A) ACTC(2) Action code for CPU passback of QCB
12(C)	ACCT(4) Accounting time this request
16(10)	STIM(4) Accounting start time
20(14)	TCB (4) Pointer to TCB for this request
24(18)	WRD1(4) Word 1..8 of function dependent data
28(1C)	WRD2(4)
32(20)	WRD3(4)
36(24)	WRD4(4)
40(28)	WRD5(4)
44(2C)	WRD6(4)

QUEUE CONTROL BLOCK (QCB) continued

48(30)	WRD7(4)
52(34)	WRD8(4)
56(38)	UPBK(4) ADDRESS OF UNRELOCATED PBLK
60(3C)	(4) FILL TO QUADWORD

QUEUE CONTROL BLOCK HEADER (QCH)

0(0) IOP (1) IOP # of user of this QCB	1(1)	GADR(3) Address GETQCB
4(4)		RADR(4) Address RELQCB

QUEUE HANDLER (QH.)

0 (0)	RTN (4)
4 (4) RDB (2)	6 (6) IRCB (2)

QUEUE PARAMETER BLOCK (QPB)

0(0)		RDYQ(8) READY TASK-QUEUE-HEAD	
8(8)		TPTR(4) TTB POINTER	
12(C)		QWTC(4) Q-WAIT APU COUNT	
16(10) MTID(2) MAPPING RIGHTS TASKID		18(12) ETID(2) EXCLUSIVE TASK ID	
20(14) STAT(2) QUEUE PROCESSING STATUS		22(16) BMAP(2) APU ASSIGNMENT BITMAP	
24(18) ACNT(1) # OF APU'S ASSIGNED	25(19) LCNT(1) # OF LPU'S MAPPED	26(1A) ID (1) QPB ID NUMBER	27(1B) (1) FILLER
28(1C)		(4) FILLER	

RS/RSA CONTEXT BLOCK (RCTX)

0 (0)	FPTR (4) FORWARD CTX POINTER
4 (4)	BPTR (4) BACKWARD CTX POINTER
8 (8)	TCB (4) TCB POINTER
12 (C)	TRCE (4) RETURN LINK FROM STATE SWITCH
16 (10)	PSW (8) OLD PSW SAVE AREA
24 (18)	PSTD (4) PROCESS SEGMENT TABLE DESCR.
28 (1C)	REGS (64) GENERAL REGISTER SAVE AREA

ROLL SELECTION LIST (RLST)

0(0)	FSD (4) FIRST SDE IN ROLL LIST
4(4)	NSD (4) NEXT SEG AFTER ROLL LIST
8(8)	FFD (4) FIRST FREE BLOCK IN ROLL LIST
12(C)	NFD (4) NEXT FREE BLOCK AFTER ROLL LIST
16(10)	SIZ (4) TOTAL SIZE ACQUIRED BY LIST
20(14)	ISZ (4) SIZE OF IMPURE SEGS TO BE ROLLED
24(18)	ADR (4) HIGHEST ADDRESS IN LIST
28(1C)	SAD (4) START OF SELECTED MEMORY
32(20) NUM (2) COUNT OF SEGMENTS IN ROLL LIST	34(22) SEK (2) COUNT OF IMPURE SEGS TO BE ROLLED
36(24) PRI (1) HIGHEST PRIORITY IN ROL	37(25) (3) FILLER

SEGMENT DESCRIPTOR ENTRY (SDE)

0(0)	FPTR(4) FORWARD LINK POINTER		
4(4)	BPTR(4) BACKWARD LINK POINTER		
8(8)	NAME(20) SEGMENT NAME & DATE		
28(1C)	SADR(4) SEGMENT START ADDRESS, 0 IF NO MEMORY		
32(20)	SIZE(4) SEGMENT SIZE IN BYTES		
36(24)	USE (2) USE COUNT	38(26)	ROLL(2) ROLL COUNT
40(28)	FLGS(1) FLAGS	41(29)	TYPE(1) TYPE OF SEGMENT
		42(2A)	PREG(1) PURE SEG REGISTER (IMPUR
		43(2B)	KEY (1) ROLL KEY
44(2C)	ASG (4) ROLL ASSIGNMENT (LU ENTRY)		
48(30)	FSEC(4) FIRST SECTOR NUMBER		
52(34)	SSTP(4) A(1ST STE) IN SST FOR THIS SEGMENT		
56(38)	SSUB(4) HEAD OF SSUB CHAIN FOR THIS SEGMENT		
60(3C)	NSUB(2) NUMBER OF SSUBS/TASKS CONNECTED	62(3E)	PRIV(1) ACCESS PRIVILEGE FOR THE SEGMENT
		63(3F)	RFLG(1) ROLL FLAGS

SECONDARY DIRECTORY (SECDIR)

0(0)	(4) PRIMARY DIRECTORY POINTER 1
4(4)	(12) FILENAME-1
16(10)	(12) FILENAME-2
28(1C)	(12) FILENAME-3
40(28)	(12) FILENAME-4
52(34)	(12) FILENAME-5
64(40)	(4) PRIMARY DIRECTORY POINTER 2
68(44)	(12) FILENAME-1
80(50)	(12) FILENAME-2
92(5C)	(12) FILENAME-3
104(68)	(12) FILENAME-4
116(74)	(12) FILENAME-5

SECONDARY DIRECTORY (SECDIR) continued

128(80)	(4) PRIMARY DIRECTORY POINTER 3
132(84)	(12) FILENAME-1
144(90)	(12) FILENAME-2
156(9C)	(12) FILENAME-3
168(A8)	(12) FILENAME-4
180(B4)	(12) FILENAME-5
192(C0)	(4) PRIMARY DIRECTORY POINTER 4
196(C4)	(12) FILENAME-1
208(D0)	(12) FILENAME-2
220(DC)	(12) FILENAME-3
232(E8)	(12) FILENAME-4
244(F4)	(12) FILENAME-5

SYSTEM POINTER TABLE (SPT)

0(0)	INIT(6) BRANCH TO START OF OS
	6(6) CRSH(2) CRASH CODE
8(8)	FLV (4) FIRST LEAF ADDRESS
12(C)	LLV (4) LAST LEAF ADDRESS
16(10) ISPT(2) ISP END ADDR - MAC START	18(12) MLBL(2) MESSAGE LOG BUFFER LENGTH
20(14)	CSLV(4) NUMBER OF CSS LEVELS
24(18)	CSBF(4) SIZE OF CSS BUFFER+2
28(1C)	CTOP(4) CURRENT ALLOCATION TOP
32(20)	UTOP(4) TASK SPACE END - SYS SPACE BEG
36(24)	UBOT(4) TASK SPACE START
40(28)	PMEM(4) PHYSICAL MEMORY SIZE OF MACHINE
44(2C)	MTOP(4) MEMORY SIZE AVAILABLE TO OS

SYSTEM POINTER TABLE (SPT) continued

48(30)	OSID(8) OS REVISION NUMBER
56(38)	OSUP(4) OS UPDATE LEVEL
60(3C)	SVT (4) ADDR OF SYSTEM VALUE TABLE
64(40)	IVT (4) ADDR OF INITIAL VALUE TABLE
68(44)	TTAB(4) ADDR OF TCB TABLE
72(48)	DMT (4) ADDR OF DEVICE MNEMONIC TABLE
76(4C)	VMT (4) ADDR OF VOLUME MNEMONIC TABLE
80(50)	SVOL(4) VOL NAME OF SYSTEM VOLUME
84(54)	SPVL(4) VOL NAME OF SPOOL VOLUME
88(58)	RVOL(4) VOL NAME OF ROLL VOLUME
92(5C)	TVOL(4) VOL NAME OF TEMP VOLUME
96(60)	JRNL(4) ADDR OF SYSTEM JOURNAL

SYSTEM POINTER TABLE (SPT) continued

100(64)	(1)	101(65)	NTCB(1)	102(66)	60 (2)
			MAX NUMBER OF TASKS		SECONDS COUNTER
104(68)	FREQ(2)			106(6A)	PIC (2)
	2 * LINE FREQUENCY				DEVICE ADDR OF PIC
108(6C)	LFC (2)			110(6E)	CPU (2)
	DEVICE ADDR OF LFC				CPU MODEL NUMBER
112(70)			SOPT(4)		
			SYSTEM OPTIONS FLAGS		
116(74)	CLSS(2)			118(76)	SSTS(2)
	NUMBER OF DEVICE/FILE CLASSES				SST SIZE/NUMBER OF ENTRIES
120(78)			SST(4)		
			SST START ADDR		
124(7C)			SSTE(4)		
			SST ENDING ADDR		
128(80)			ADCK(4)		
			ADCK ROUTINE ADDRESS		
132(84)			ADK1(4)		
			ADK1 ROUTINE ADDRESS		
136(88)			ATQ1(4)		
			TMATQ1 ROUTINE ADDRESS		
140(8C)			ASQ1(4)		
			TMASQ1 ROUTINE ADDRESS		
144(90)			VERN(8)		
			USERS OS VERSION		

SYSTEM POINTER TABLE (SPT) continued

152(98)	MNTH(2) CURRENT MONTH	154(9A)	DAY (2) CURRENT DAY
156(9C)	YEAR(2) CURRENT YEAR	158(9E)	TSL (2) TIME SLICE IN CLOCK TICKS
160(A0)	TIME(4) TIME IN SECONDS SINCE MIDNIGHT		
164(A4)	DTHD(4) PTR TO DEVICE TIME OUT CHAIN		
168(A8)	TQHD(4) PTR TO TIME OF DAY QUEUE		
172(AC)	IQHD(4) PTR TO INTERVAL TIME QUEUE		
176(B0)	CTMD(4) CTH POLL DELAY CONTROL COUNTER		
180(B4)	SCTH(4) SEGMENT CONTROL TABLE - HEAD		
184(B8)	SCTT(4) SEGMENT CONTROL TABLE - TAIL		
188(BC)	PANC(4) ADDR OF PANIC DUMP ROUTINE		
192(C0)	PSV (8) TM PSW SAVE AREA		
200(C8)	RSV (4) TM REGISTER SAVE AREA		

SYSTEM POINTER TABLE (SPT) continued

204(CC)	TSV (4) TM ADDRESS SAVE AREA
208(D0)	SCL (4) PTR TO TOP OF SEGMENT CTRL LIST
212(D4)	FLST(4) FREE MEM LIST IN TASK SPACE PTR
216(D8)	FSYP(4) FREE MEM LIST IN SYS SPACE PTR
220(DC)	EREL(4) ERROR BUFFER LIST ADDRESS
224(E0)	SPL (4) TID OF .SPL
228(E4)	MTM (4) TID OF .MTM
232(E8)	LLE (4) TID OF .MTMLLE
236(EC)	MISS(4) # OF MESSAGES MISSED BY .MTM
240(F0)	CTCB(4) PTR TO CURRENT TASK
244(F4)	RDYQ(4) PTR TO READY-QUEUE-HEAD
248(F8)	RINQ(4) PTR TO ROLL-QUEUE-HEAD

SYSTEM POINTER TABLE (SPT) continued

252(FC)	(4)
256(100)	UTOW(4) OWNER OF UT REGISTERS
260(104)	RSOW(4) OWNER OF RS REGISTERS
264(108)	ESOW(4) OWNER OF ES REGISTERS
268(10C)	MCOW(4) OWNER OF MAC/HAT
272(110)	EFOW(4) OWNER OF SINGLE FLOAT REGISTERS
276(114)	DFOW(4) OWNER OF DOUBLE FLOAT REGS
280(118)	RSON(4) RS STATE PSW WITH SQS ON
284(11C)	RSOF(4) RS STATE PSW WITH SQS OFF
288(120)	RLIO(4) ROLL I/O COUNT
292(124)	(4) ROLL I/O SECTORS
296(128)	OVHD(4) OS OVERHEAD TIME (1ST WORD)

SYSTEM POINTER TABLE (SPT) continued

300(12C)	OVHE(4) (2ND WORD)
304(130)	CPID(4) CPU IDLE TIME (1ST WORD)
308(134)	CPIE(4) (2ND WORD)
312(138)	VALU(4) VALUE IN REG RETURNED BY GETUREG
316(13C)	PAGE(4) PAGE SIZE
320(140)	SCLP(4) PAGE SCALING FACTOR
324(144)	ADDM(4) VALID ADDR BITS=SEG+PAGE+OFF MASK
328(148)	OFFM(4) OFFSET IN PAGE MASK
332(14C)	PAGM(4) PAGE IN SEGMENT MASK
336(150)	SEGM(4) SEGMENT NUMBER MASK
340(154)	NPGM(4) NEGATIVE PAGE SIZE MASK
344(158)	STAM(4) SEGMENT TABLE ALIGNMENT MASK

SYSTEM POINTER TABLE (SPT) continued

348(15C)	NODE(4) LOCAL NODE NAME (1ST WORD)	
352(160)	(4) (2ND WORD)	
356(164)	PSDD(4) PSEUDO-DCB CHAIN POINTER	
360(168)	PSDT(4) PSEUDO TCB CHAIN POINTER	
364(16C)	QH (4) QUEUE HANDLER TABLE POINTER	
368(170)	TID (4) TASK COUNTER ID	
372(174)	PID (2) COUNTER FOR INTC PATH ID	374(176) IPID(2) COUNTER FOR INTC PATH ID
376(178)	QCB (4) POINTER TO QCB POOL	
380(17C)	LPMT(4) PTR LOGICAL PROCESSOR MAPPING TBL	
384(180)	QPBO(4) POINTER TO QPB 0	
388(184)	CQ0 (4) CPU QUEUE 0 LOOKUPS	
392(188)	TPTR(4) TRAP POINTER DIR.	

SYSTEM POINTER TABLE (SPT) continued

396(18C) SPCT(2) SPOOL/TEMP EXTENSION COUNTER	398(18E) NTSK(2) COUNT OF TASKS ON READY QUEUE
400(190) SQDP(2) MOST CURRENT MAX SQ DEPTH	402(192) CALL(2) SANITY CALLER NUMBER
404(194) MON (4) FOR SVC 2.30 MONITOR	
408(198) (8) RESERVED	

SEGMENT TABLE ENTRY (STE)

0 (0)	HSTE (4) HARDWARE STE
4 (4)	SSTE (4) SOFTWARE STE

SYSTEM VALUE TABLE (SVT)

0(0) MBLK(2) MAXIMUM BLOCK SIZE ALLOWED	2(2) DBLK(2) DATA BLOCK SIZE (INDEXED)
4(4) IBLK(2) INDEX BLOCK SIZE (INDEXED)	6(6) DSPL(2) DATA BLOCK SIZE (SPOOLER)
8(8) ISPL(2) INDEX BLOCK SIZE (SPOOLER)	10(A) DNBF(2) DATA BLOCK SIZE (NON-BUFFERED)
12(C) INBF(2) INDEX BLOCK SIZE (NON-BUFFERED)	14(E) (2) RESERVED

SYSTEM SPACE STRUC (SYP)

0(0)	PTR (4) FDE FORWARD POINTER
4(4)	SIZE(4) FDE SIZE
8(8)	TRAC(4) LAST OWNER
12(C)	USIZ(4)

TASK CONTROL BLOCK (TCB)

0(0)	FPTR(4) FORWARD LINK POINTER
4(4)	BPTR(4) BACKWARD LINK POINTER
8(8)	QPTR(4) TASK-QUEUE-HEAD POINTER
12(C)	CTX (4) CONTEXT BLOCK STACK PTR
16(10)	FLGS(4) TCB QUEUE FLAGS
20(14)	RTCV(4) RTSM COUNTER VALUE
24(18)	ETA (8) 64-BIT EXECUTION-TIME ACCUMULATOR
32(20)	TPTR(4) TASK TRAP BLOCK POINTER
36(24)	PRCA(4) PASSBACK REASON CODE
40(28)	PRCB(4) PRC INFO
44(2C)	PRCC(4) PRC INFO
48(30)	PRCD(4) PRC INFO

TASK CONTROL BLOCK (TCB) continued

52(34)	PRCE(4) PRC INFO
56(38)	PRCF(4) PRC INFO
60(3C)	SDE (64)
124(7C)	TID (4) TASK ID + COUNTER
128(80)	STAT(4) TASK STATUS
132(84)	ICB (64) ICB
196(C4)	WAIT(4) WAITS
200(C8)	OPT (4) OPTIONS
204(CC)	CLC (4) CONNECTED RESOURCE CHAIN
208(D0)	TGD (4) CONNECTED TRAP-GEN DEV CHAIN
212(D4)	VOFF(2) OVERLAY SAVE AREA OFFSET
214(D6)	RSV (2) ROLL SAVE AREA OFFSET
216(D8)	RSAC(2) COUNT OF RSA NESTING LEVEL
218(DA)	RCNT(2) ROLL IN COUNT

TASK CONTROL BLOCK (TCB) continued

220(DC)PRI (1) PRIORITY - USER TASK	221(DD)RPRI(1) PRIORITY - DYNAMIC RUN	222(DE)DPRI(1) PRIORITY - DISPATCH	223(DF)MPRI(1) PRIORITY - MAXIMUM
224(E0)	CTSW(4) CURRENT TSW		
228(E4)	SLOC(4) START LOCATION		
232(E8)	CTOP(4) CTOP- (ABSOLUTE ADDRESS)		
236(EC)	UTOP(4) UTOP- (PROGRAM ADDRESS)		
240(F0)	TTOP(4) INITIAL UTOP (PROGRAM ADDRESS)		
244(F4)	OBOT(4) OVLY BOTTOM - (PROGRAM ADDRESS)		
248(F8)	TEQH(4) POINTER TO EVENT QUEUE HEADER		
252(FC)	SEG (4) POINTER TO RELOCATION TABLE		
256(100)	PSTD(4) PSTD SAVE AREA		
260(104)	SSTD(4) SSTD SAVE AREA		
264(108)	ADCK(4) LAST PROG ADDRS CHECKED		502K

TASK CONTROL BLOCK (TCB) continued

268(10C)	MXSP(4) MAXIMUM SYSTEM SPACE	
272(110)	USSP(4) USED SYSTEM SPACE	
276(114)	SYSP(4) MAX SYS SPACE USED BY TASK	
280(118)	SHSZ(4) TOTAL SIZE SHARED SEGMENTS USED	
284(11C)	NSHD(2) NUMBER OF SHARED SEGMENTS	286(11E) MSEG(2) MAX SEGMENT NUMBER OF TASK
288(120)	SOPT(4) START OPTIONS POINTER	
292(124)	DLAY(4) DELAY START TIME	
296(128)	VOL (4) TASK DEFAULT VOLUME FOR PACK FD	
300(12C)	TMPV(4) TASK DEFAULT TEMP VOLUME	
304(130)	SVAD(4) SVC ADDRESS AT THE TIME OF INT	
308(134)	TMP1(4) TEMPORARY SAVE AREA	
312(138)	TMP2(4) TEMPORARY SAVE AREA	

TASK CONTROL BLOCK (TCB) continued

316(13C)	TMP3(4) TEMPORARY SAVE AREA
320(140)	USER(8) RESERVED FOR USER
328(148)	SYS (4) SYSTEM WORDS (SVC3 BUILDS SVC7BLK)
332(14C)	SYS1(4)
336(150)	SYS2(4)
340(154)	SYS3(4)
344(158)	SYS4(4)
348(15C)	SYS5(4)
352(160)	SYS6(4)
356(164)	IWAT(4) IOP WAIT TIME FOR TASK
360(168)	(12) (RESERVED FOR PATCHES)
372(174)	VOLN(4) VOLUME NAME

TASK CONTROL BLOCK (TCB) continued

376(178)	FN (8) FILE NAME	
384(180)	EXT (4) EXTENSION	
388(184)	DATE(4) DATE LAST WRITTEN	
392(188)	ACCT(2) ACCOUNT NUMBER	394(18A) RC (2) RETURN CODE
396(18C)	USRA(8) USER ACCOUNTING DATA	
404(194)	UACT(2) USER ACCOUNT NUMBER	406(196) GACT(2) GROUP ACCOUNT NUMBER
408(198)	RID (1) REPORT ID	409(199) GID (1) GROUP ID
		410(19A) XTIN(2) # SECONDS WAITING TO EXECUTE
412(19C)	CPLM(4) CPU TIME LIMIT	
416(1A0)	CPU (20) CPU TIME COUNTERS	
436(1B4)	WTIM(4) TIME IN WAIT STATE	
440(1B8)	RTIM(4) TIME ON ROLL Q	
444(1BC)	TIWT(4) TIME PUT INTO WAIT	

TASK CONTROL BLOCK (TCB) continued

448(1C0) ROUT(4) TIME PUT ONTO ROLL Q			
452(1C4) STIM(4) TIME TASK WAS STARTED			
456(1C8) TSL (2) TIME SLICE COUNTER		458(1CA) RLSL(2) ROLLED-OUT TIME PENALTY	
460(1CC) TIHR(2) TIMER TRAP COUNTS		462(1CE)LOAD(1) LOAD COUNTS	463(1CF) (1) RESERVED
464(1D0) IOC (2) IO COUNT		466(1D2) IOAC(2) PTR TO IO COUNTERS	
468(1D4) IOBL(4) IOB POOL POINTER			
472(1D8) CIOB(4) MOST RECENT USED IOB			
476(1DC) IOB (80) SPARE I/O BLOCK			
556(22C) UCTX(4) POINTER TO USER CTX			
560(230) RS (92) RS STATE CONTEXT BLOCK			
652(28C)LPU (1) LOGICAL PROCESSOR UNIT	653(28D)QUE (1) APU QUEUE ASSIGNMENT	654(28E) (1) RESERVED	655(28F)NLU (1) NUMBER OF LOGICAL UNITS
656(290) LOCK(2) T/S lock for TCB, IOBL, TCB, IOC		658(292) (2) RESERVED	

TASK CONTROL BLOCK (TCB) continued

660(294)	HLTQ (4) Pointer to halt queue
664(298)	QCB (4) QCB causing connection wait
668(29C)	AF (4) ARITHMETIC FAULT PRC COUNTS
672(2A0)	II (4) ILLEGAL INSTRUCTION PRC COUNTS
676(2A4)	MF (4) MAT FAULT PRC COUNTS
680(2A8)	DF (4) DATA FAULT PRC COUNTS
684(2AC)	SVI (4) SVC INTERRUPT PRC COUNTS
688(2B0)	MM (4) MACHINE MALFUNCTION COUNTS
692(2B4)	RSCH(4) RESCHEDULE INSTRUCTION PRC COUNTS
696(2B8)	XFRS(4) COUNT OF CPU TO APU TRANSFERS
700(2BC)	FMBB(4) FMIN BUFFER-BUFFER MOVES
704(2C0)	NPF (4) NUM OF PAGE FAULTS

TASK CONTROL BLOCK (TCB) continued

708(2C4)	SPMS(4) SYS SPACE PTR FOR SPM
712(2C8)	IIOR(4) I/O REQUESTS TO AN IOP
716(2CC)	OCB (4) ADDRESS OF OVLY CNTL BLK
720(2D0)	REL (4) PTR TO OVLY RELOCATION TABLE
724(2D4)	TUB (4) TERMINAL USER BLOCK ADDRESS
728(2D8)	PRIV(4) MTM USER PRIVILEGES
732(2DC)	UCON(4) MTM USER CONSOLE DEVICE
736(2E0)	LTFL(4) TABLE OF LU FLAGS
740(2E4)	FMLU(4) LU 255 - (FILE MANAGER LU)
744(2E8)	LTAB(4) LOGICAL UNIT TABLE

ACCOUNTING DATA EXTENSION (TIO)

0(0)	CNT (4) NUMBER OF SVCI'S
4(4)	DAT (4) DATA TRANSFERED
8(8)	LIM (4) I/O LIMIT

TASK QUEUE-HEAD (TKQ)

0(0)	LOKS(2) QUEUE ACCESS LOCKS	2(2)	CNT (2) COUNT OF TCBS IN QUEUE
4(4)	FRNT(4) FRONT TCB IN QUEUE		

TIMER QUEUE ENTRY (TMQ)

0(0)	CHN (4) FORWARD CHAIN POINTER
4(4) FLGS(1) FLAGS	5(5) RING(3) BACKWARD RING POINTER
8(8) ID (1) TASK ID	9(9) PARM(3) PARAMETER
12(C)	TIME(4) TIME INCREMENT
16(10)	FACT(4) RECURRENT FACTOR
20(14)	PERD(4) PERIOD

TASK EVENT QUEUE ENTRY (TQE)

0(0)	LEN (2) LENGTH OF TQE	2(2)	NREG(2) NUMBER OF REGISTERS TRANSFERRED
4(4)	NEXT(4) POINTER TO NEXT TQE IN QUEUE		
8(8)	TCB (4) POINTER TO TCB		
12(C)	SYSS(4) POINTER TO SYSTEM SERVICE ROUTINE		
16(10)	TSKS(4) POINTER TO TASK EVENT SERVICE ROUTIN		
20(14)	DATA(4) POINTER TO DATA ASSOC. WITH EVENT		
24(18)	REG0(4) TRANSFER REGISTER 0		
28(1C)	REG1(4) TRANSFER REGISTER 1		
32(20)	REG2(4) TRANSFER REGISTER 2		

TASK EVENT QUEUE HEADER (TQH)

0(0)	SYSQ(4) POINTER TO SYSTEM TQE QUEUE	
4(4)	TSKQ(4) POINTER TO TASK TQE QUEUE	
8(8)	TSW1(4) TSW STATUS SAVE AREA	
12(C)	TSW2(4) TSW LOC. SAVE AREA	
16(10)	SAV0(64) REGISTER SAVE AREA	
80(50)	NREG(2) NUMBER OF REGS SAVED	82(52) CNT(2) NUMBER OF TQE'S QUEUED

TASK TRAP BLOCK (TTB)

0(0)	AFNP(4) ARITH FAULT NEW PSW
4(4)	AFNL(4) ARITH FAULT NEW LOC
8(8)	IINP(4) ILLEG INST NEW PSW
12(C)	IINL(4) ILLEG INST NEW LOC
16(10)	MTNP(4) MAT FAULT NEW PSW
20(14)	MTNL(4) MAT FAULT NEW LOC
24(18)	DFNP(4) FORMAT FAULT NEW PSW
28(1C)	DFNL(4) FORMAT FAULT NEW LOC
32(20)	SVNP(4) SVC NEW PSW
36(24)	SVNL(4) SVC NEW LOC
40(28)	MMNP(4) MMF NEW PSW
44(2C)	MMNL(4) MMF NEW LOC

VOLUME DESCRIPTOR (VD)

0(0)	VOL (4) VOLUME NAME
4(4)	ATRB(4) VOLUME ATTRIBUTES
8(8)	FDP (4) FIRST DIRECTORY BLOCK POINTER
12(C)	OSP (4) POINTER OS IMAGE
16(10)	OSS (4) SIZE OS IMAGE
20(14)	MAP (4) POINTER BIT MAP
24(18)	ILA (4) OS/16 INVERTED LIST ADR. 529G
28(1C)	SDP (4) SECONDARY DIRECTORY POINTER
32(20)	SYNC(4) MIRROR DISCS SYNC STAMP

VOLUME MNEMONIC TABLE (VMT)

0(0)	(4)	VOLUME-1 NAME IN ASCII
4(4)	(4)	VOLUME-1 DMT ENTRY ADDRESS
8(8)	(4)	VOLUME-2 NAME IN ASCII
12(C)	(4)	VOLUME-2 DMT ENTRY ADDRESS
16(10)	(4)	VOLUME-3 NAME IN ASCII
20(14)	(4)	VOLUME-3 DMT ENTRY ADDRESS
24(18)	(4)	VOLUME-n NAME IN ASCII
28(1C)	(4)	VOLUME-n DMT ENTRY ADDRESS
32(20)	(8)	DOUBLE WORD OF ZERO FOR TABLE END

***OS/32 INTERNALS
FLOW DIAGRAMS***



**S
E
C
T
I
O
N

3**

FLOW DIAGRAMS

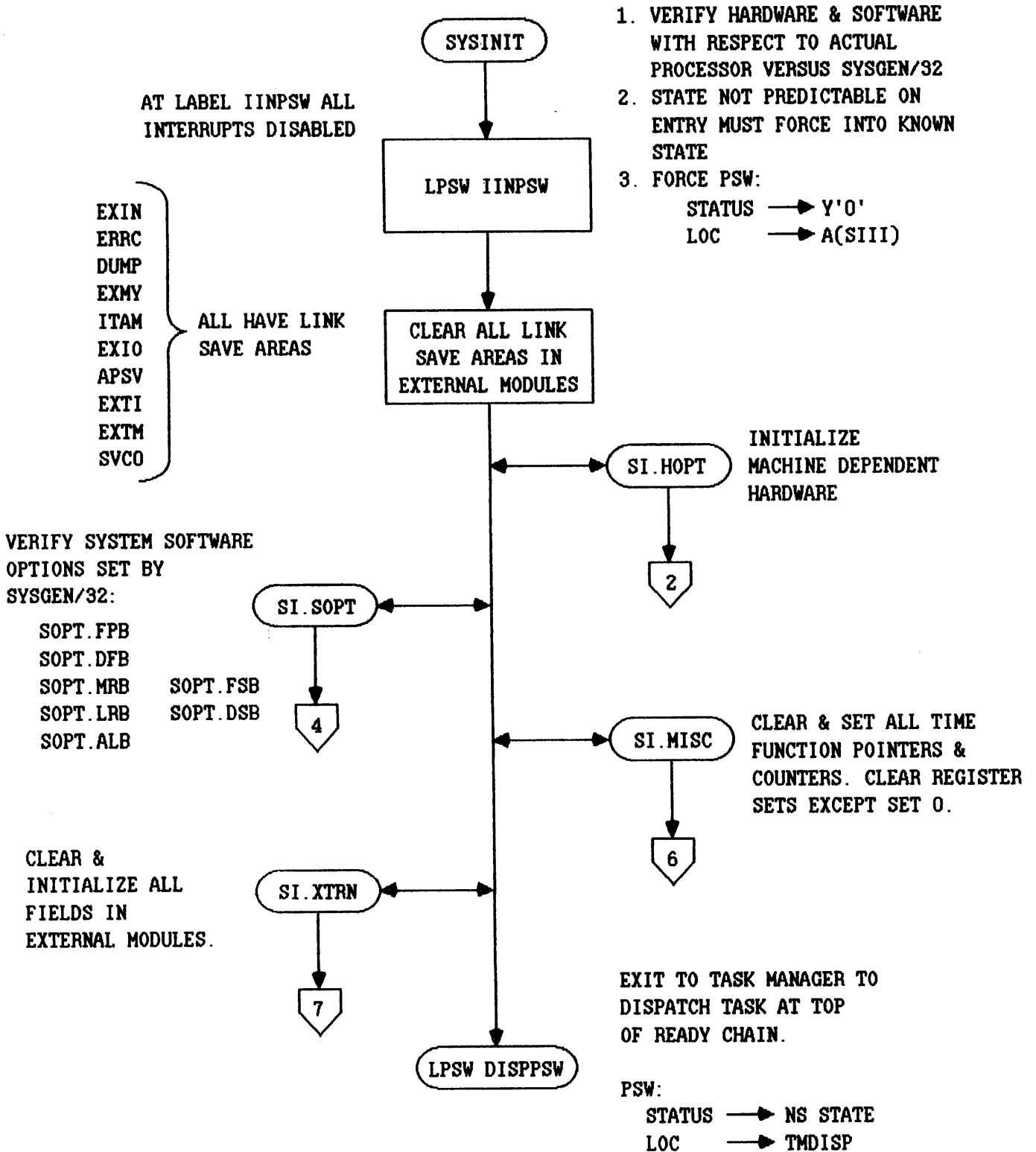
We have designed the flow diagrams in this section as pictorial representations of parts of the operating system code. They represent some of the operating system modules and their functions.

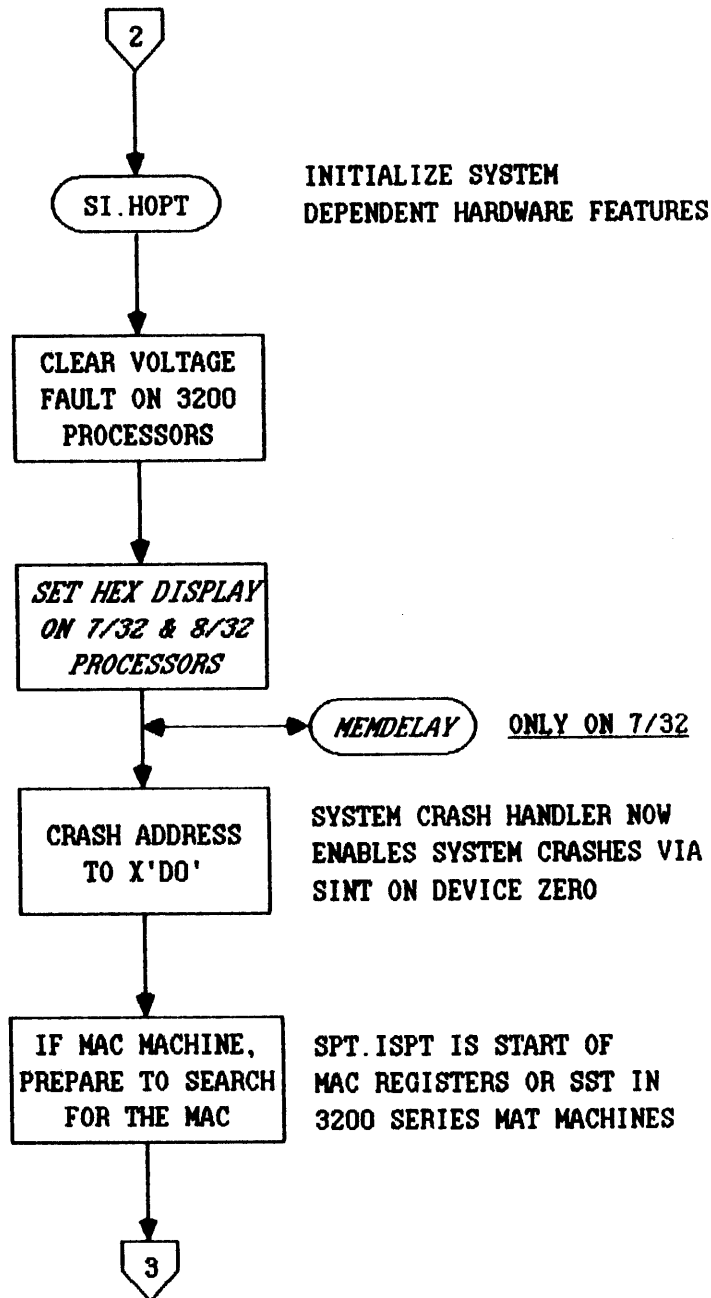
We suggest you use the flow diagrams as an aid to searching the actual assembly language code and as a way to help you understand the flow and function of the operating system modules.

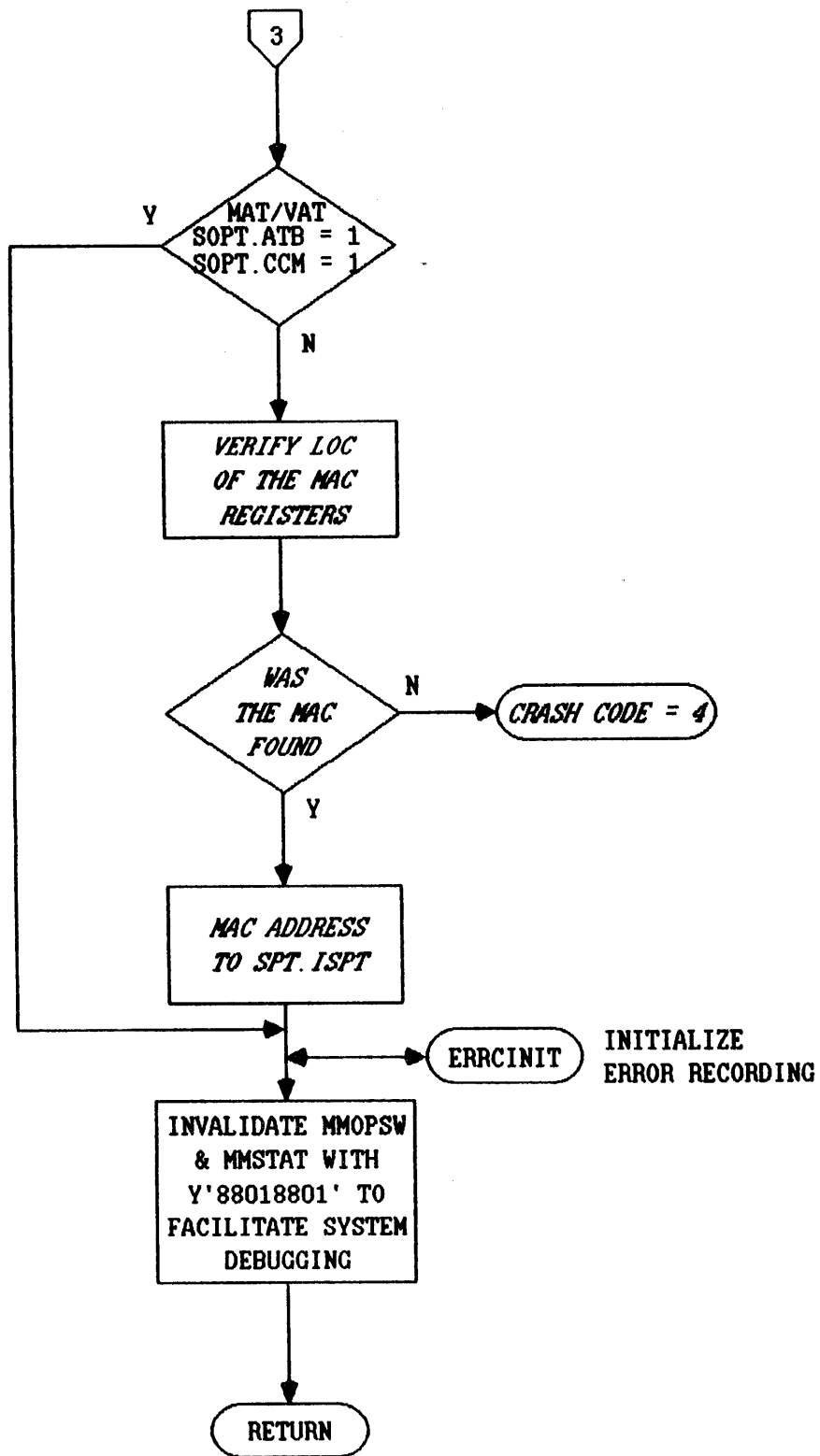
You can use the off-page connectors  to follow the code from page to page. Use the on-page connectors  to follow branches in the code on the same page.

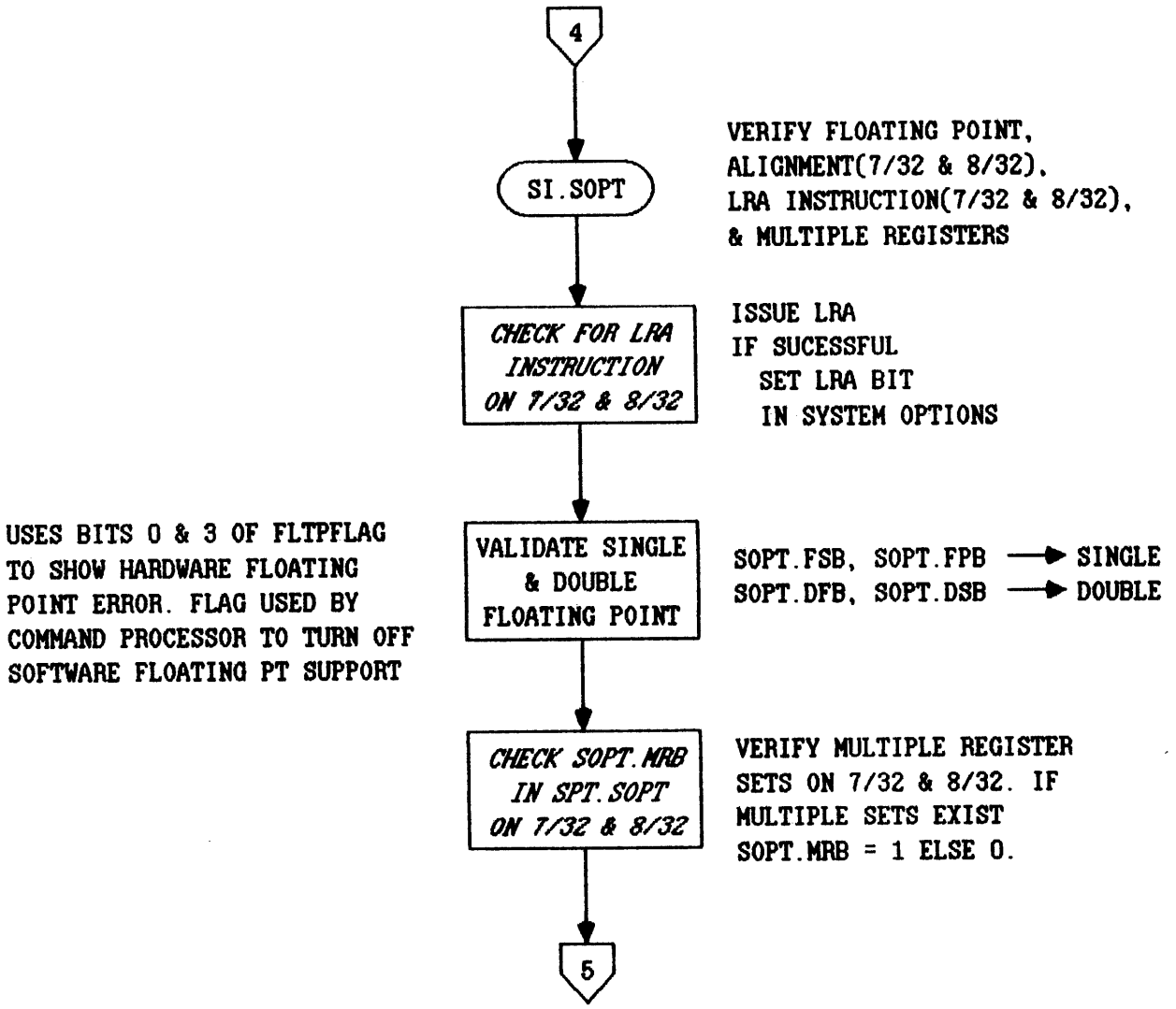
We hope you find the OS/32 Internals flow diagrams helpful and easy to follow.

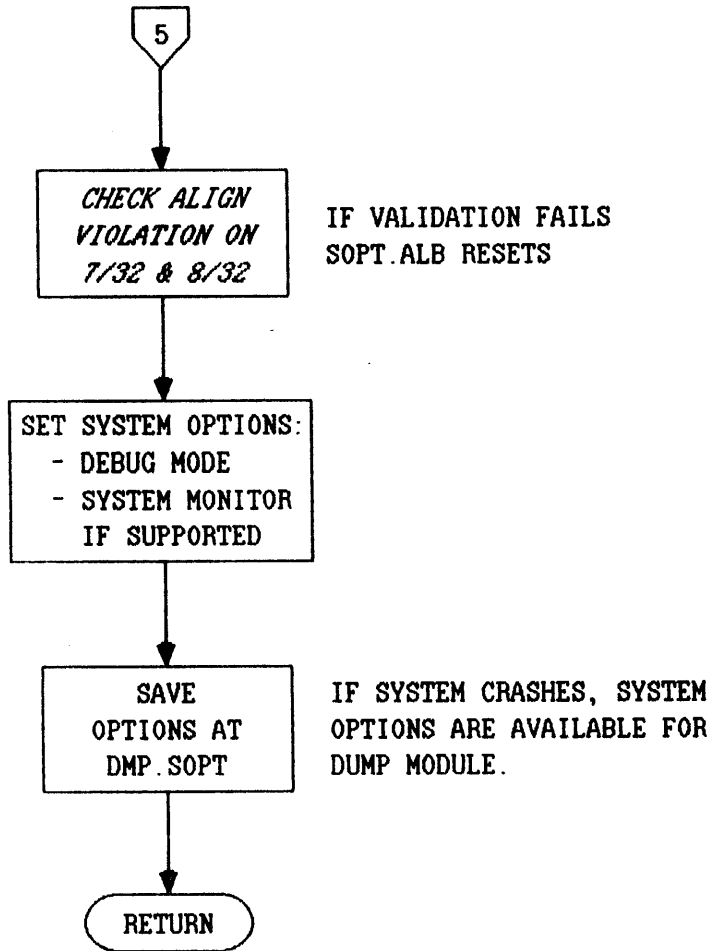
INITIALIZATION

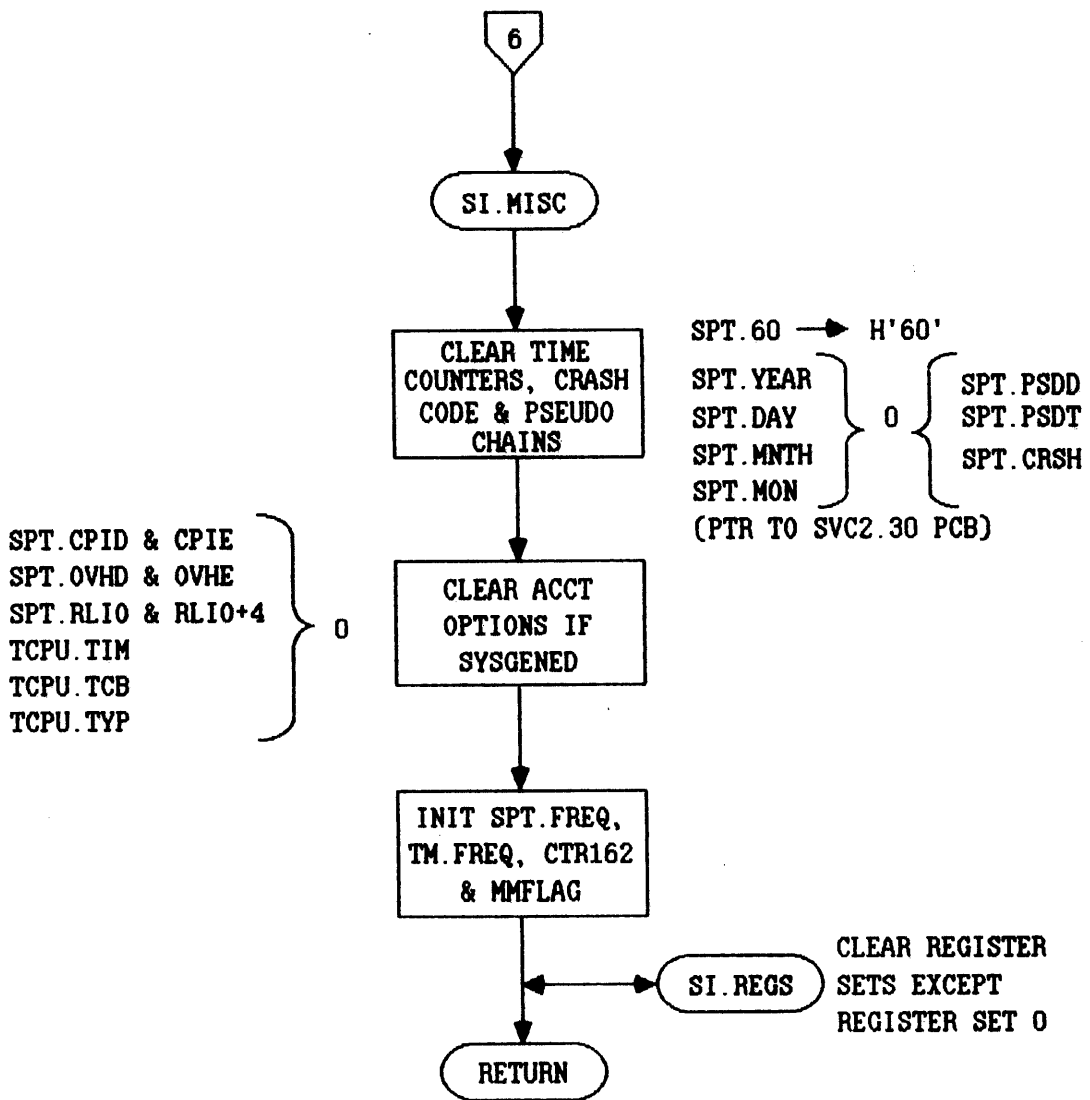


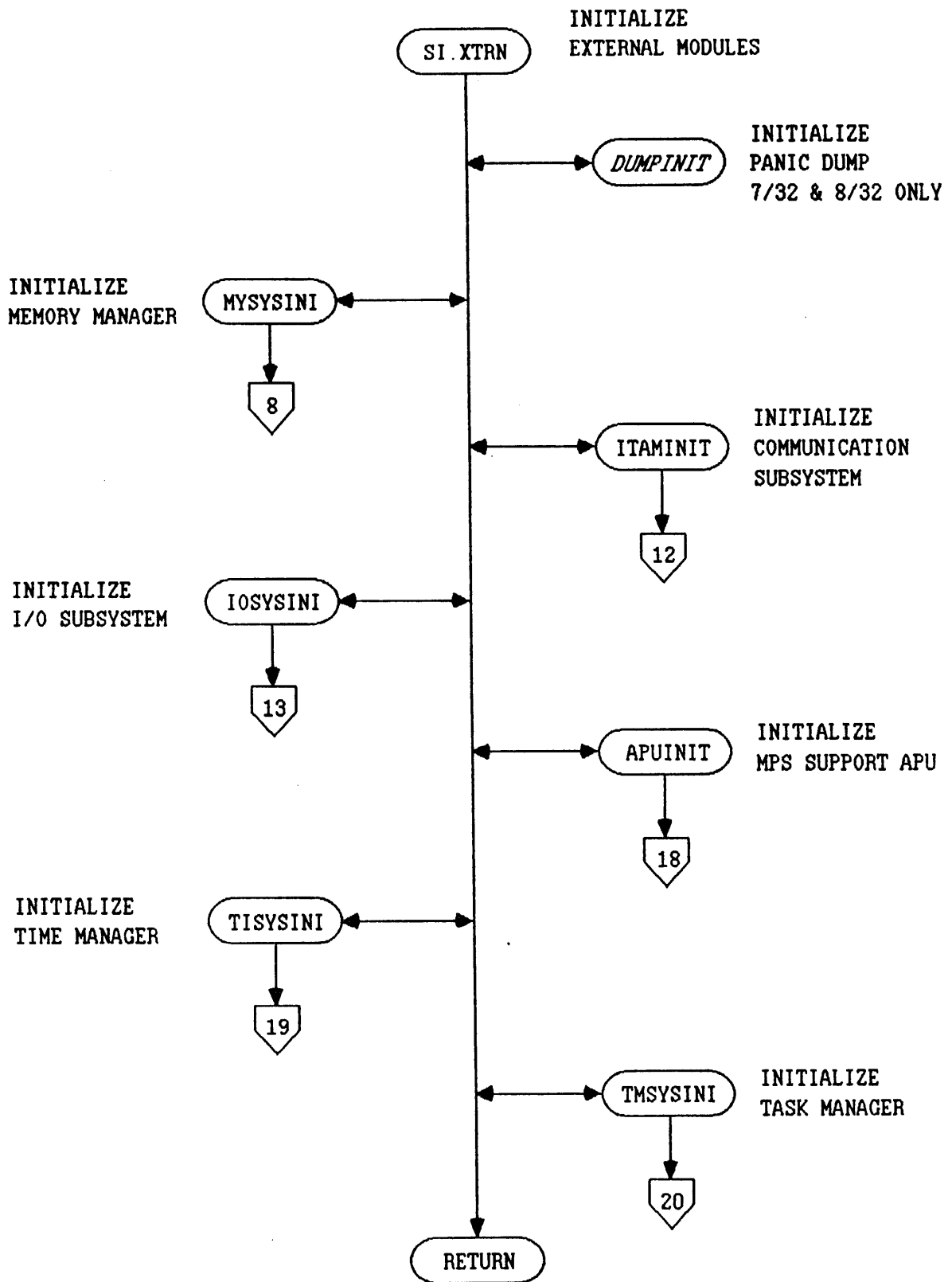


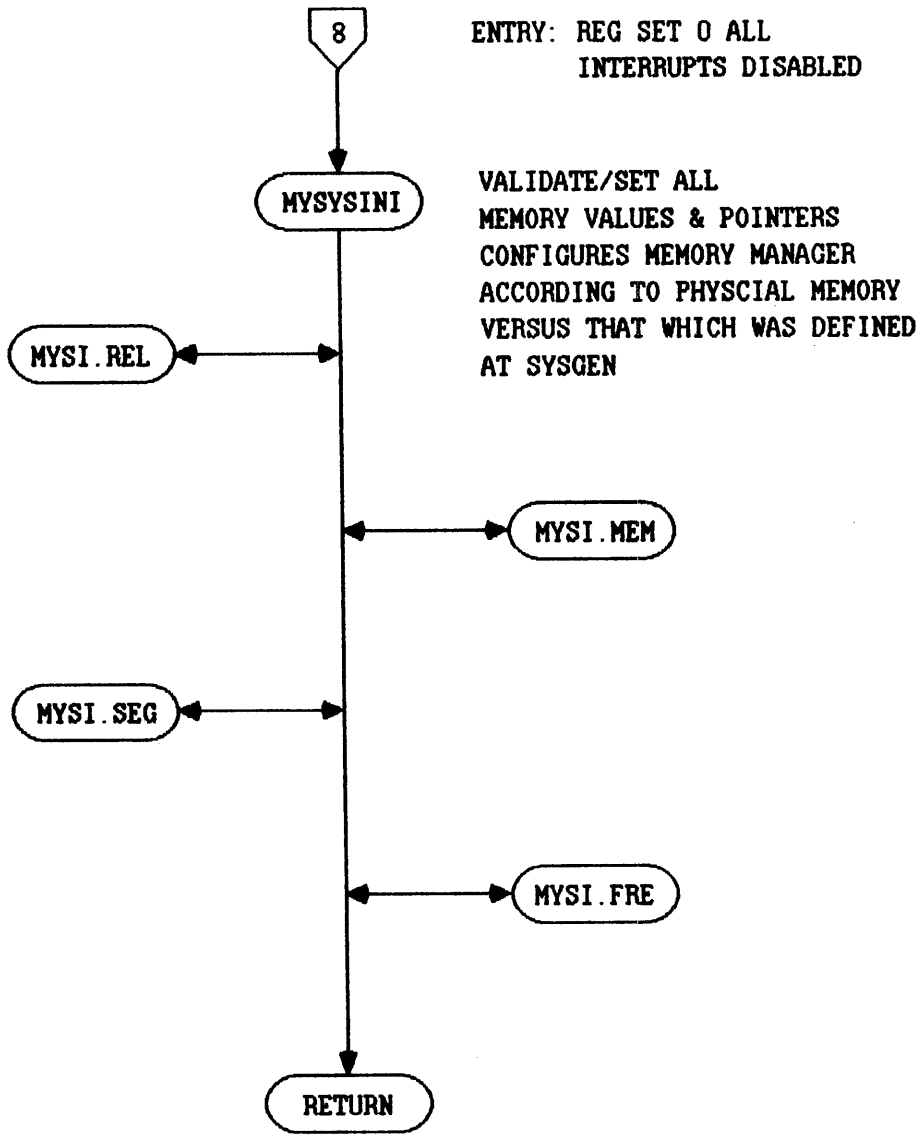


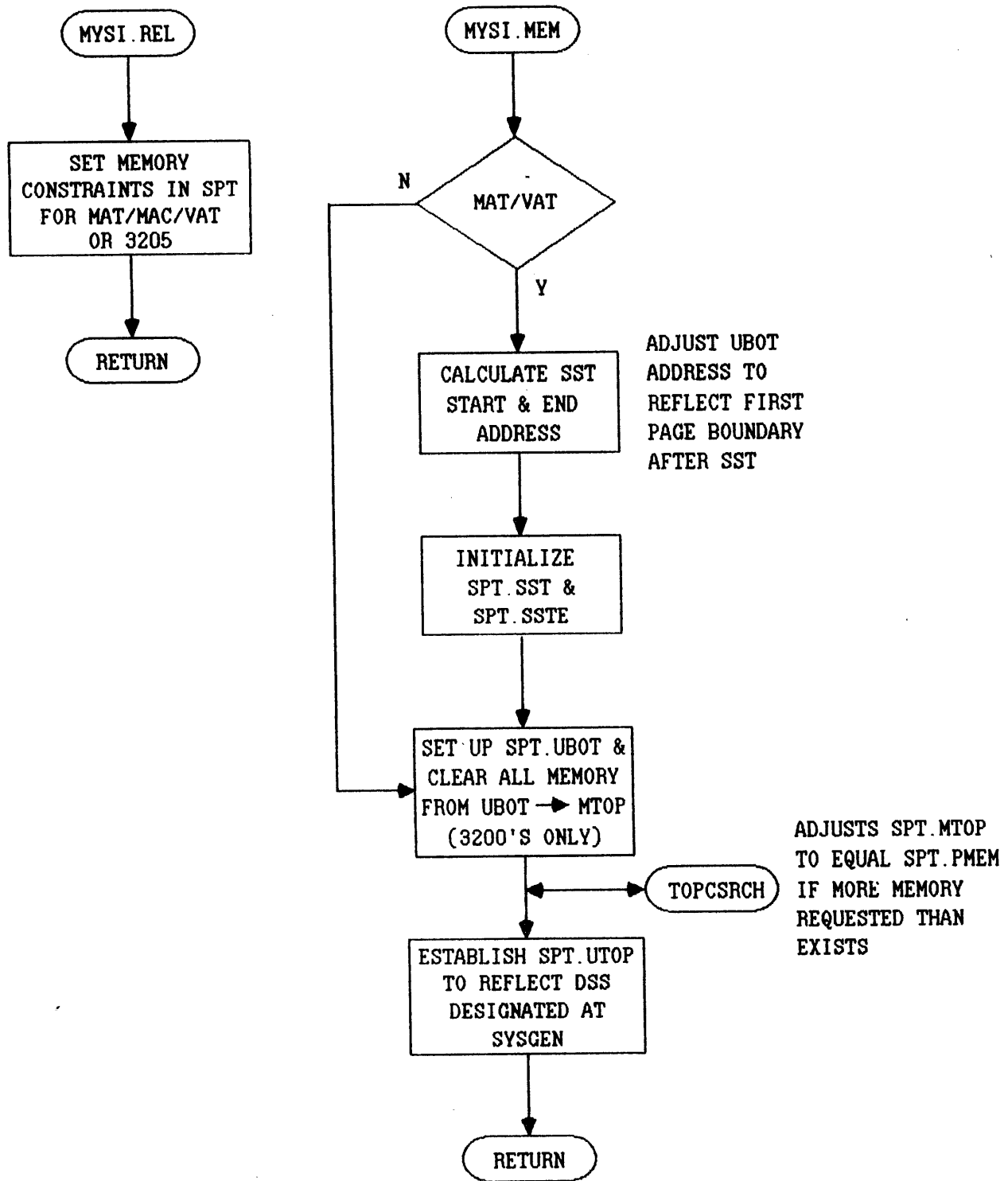


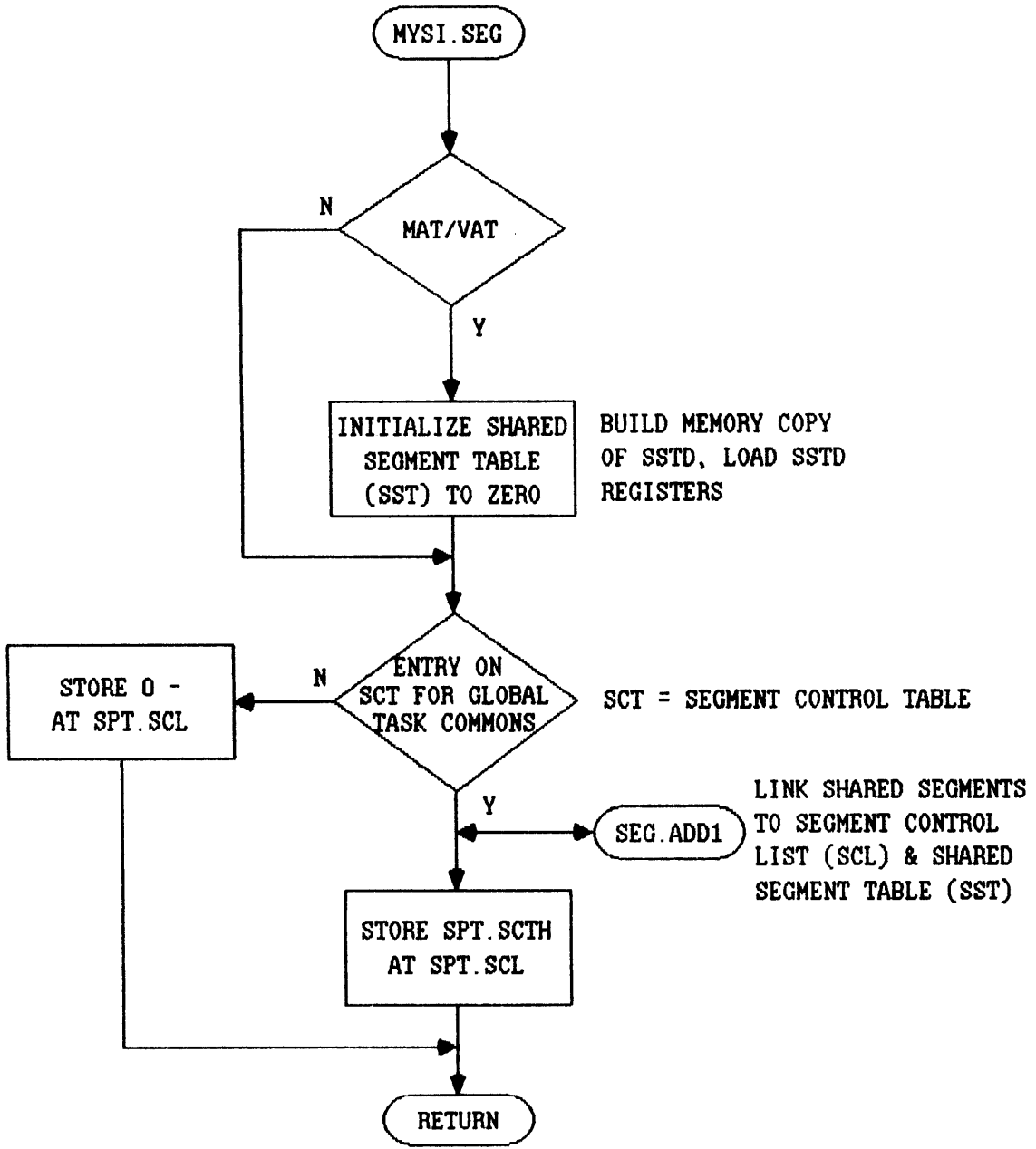


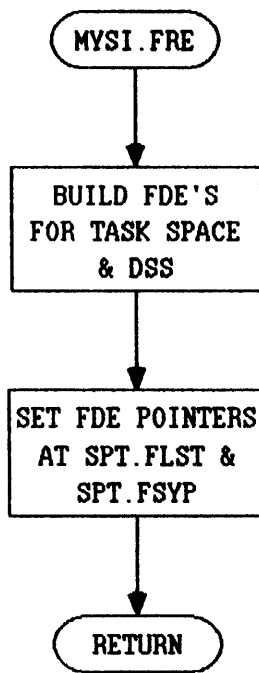


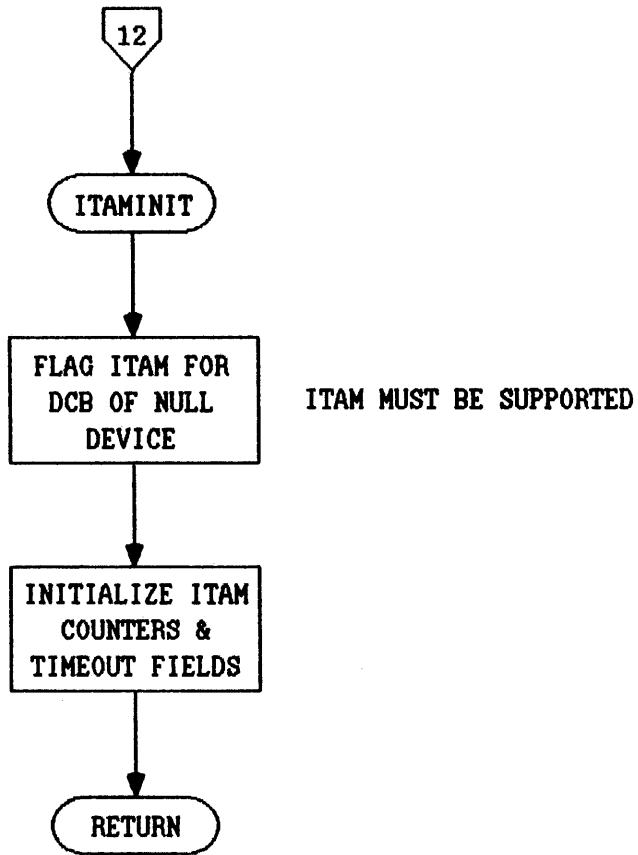


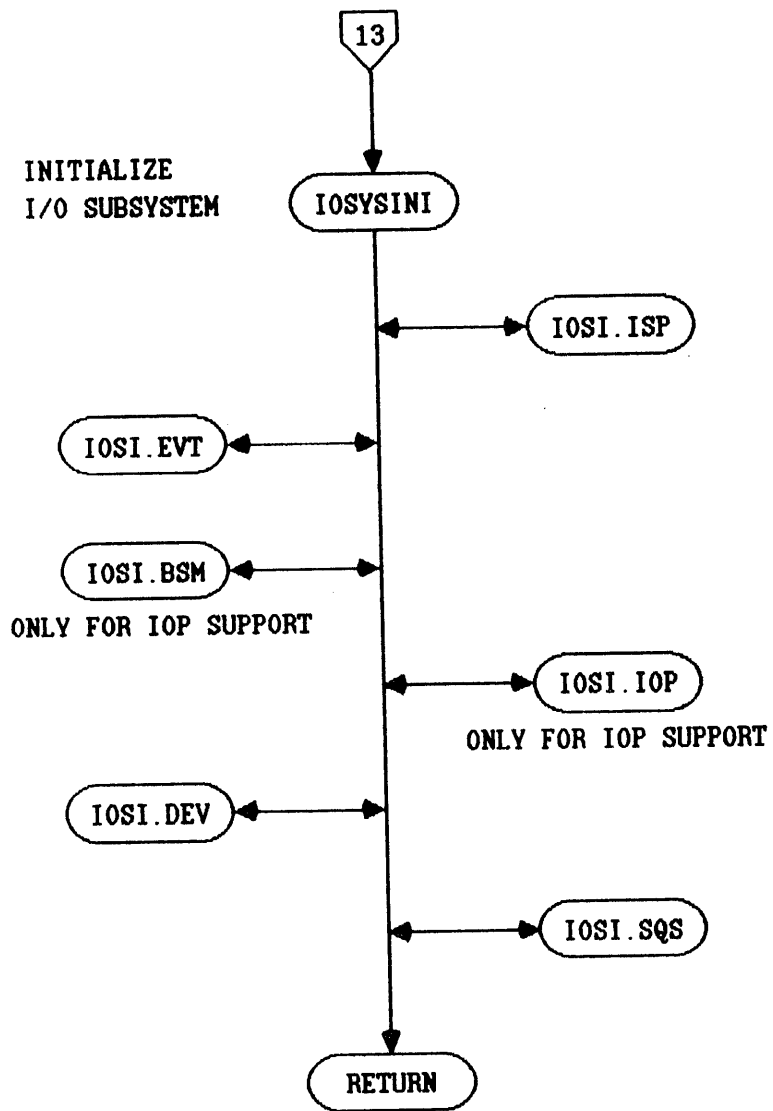




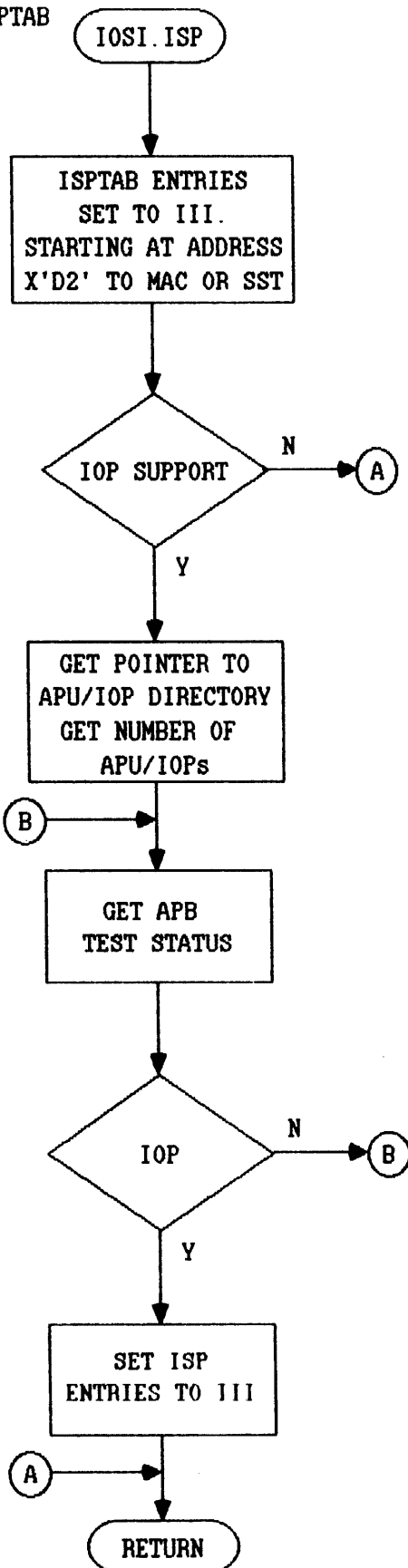






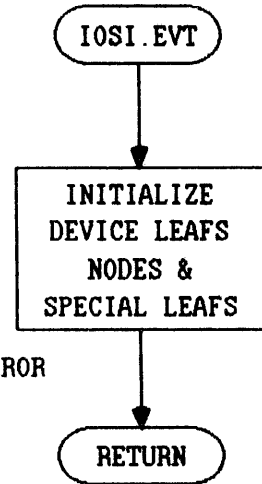


INITIALIZE ISPTAB

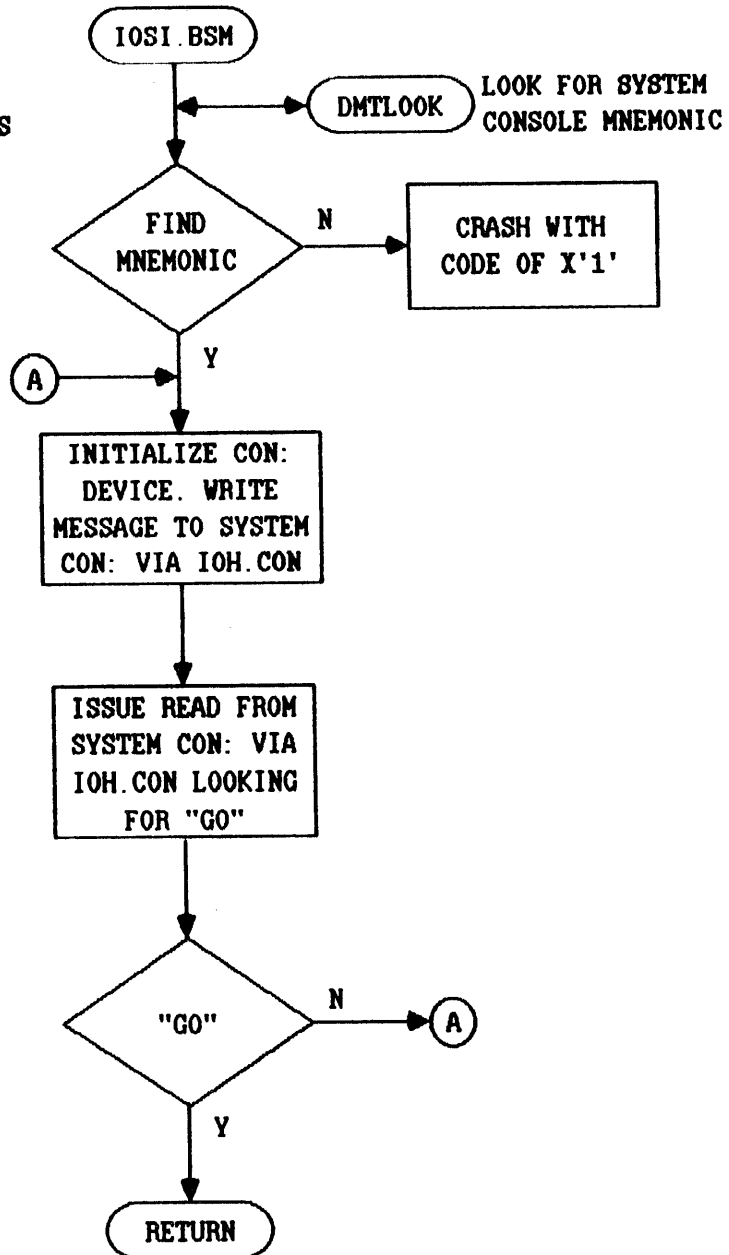


INITIALIZE DEVICE LEAF STRUCTURE

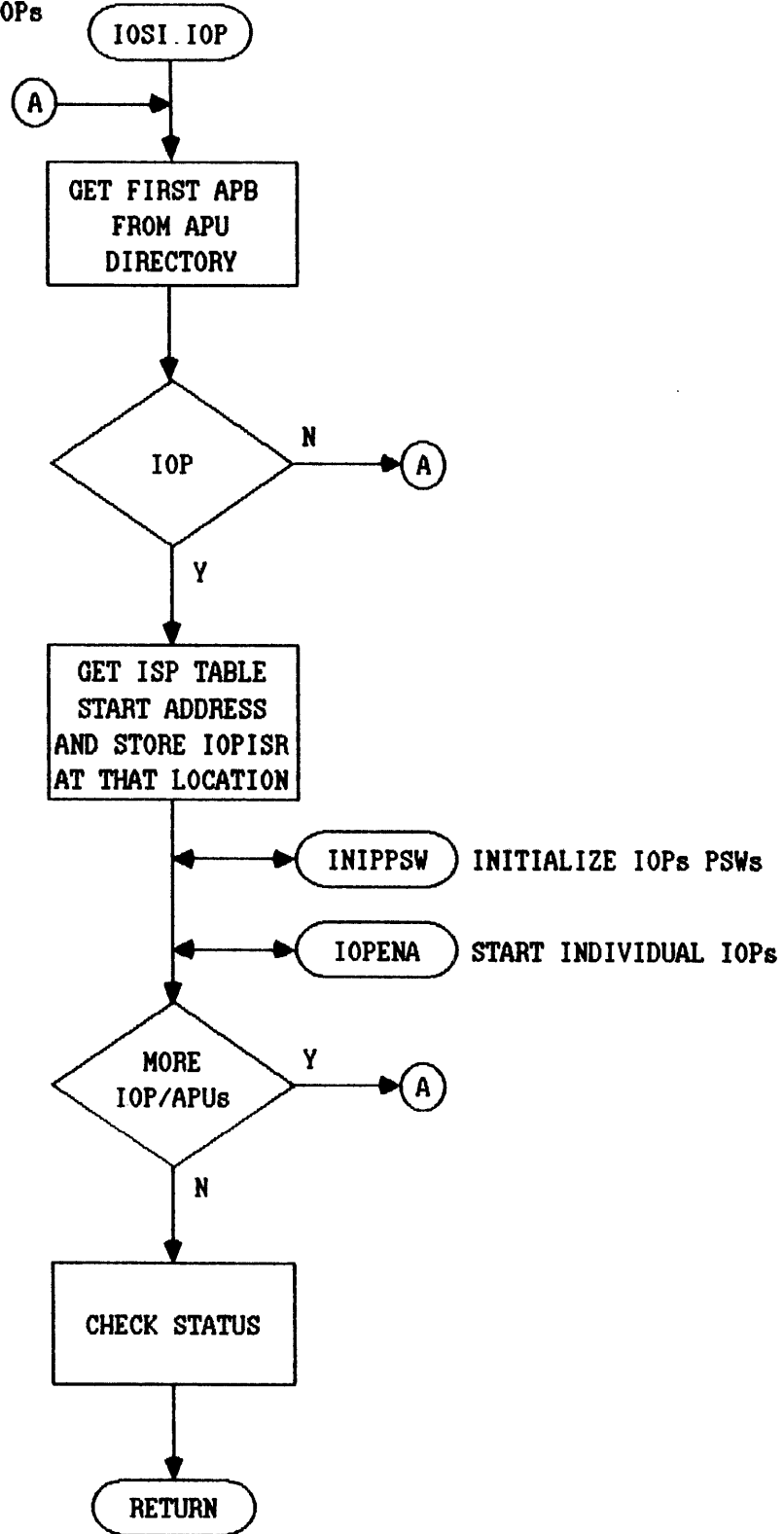
FUNCTIONS OF SPECIAL LEAFS:
TIMER
TIMESLICE
POWER RESTORE
FILE MANAGER ERROR
MEMORY ERROR



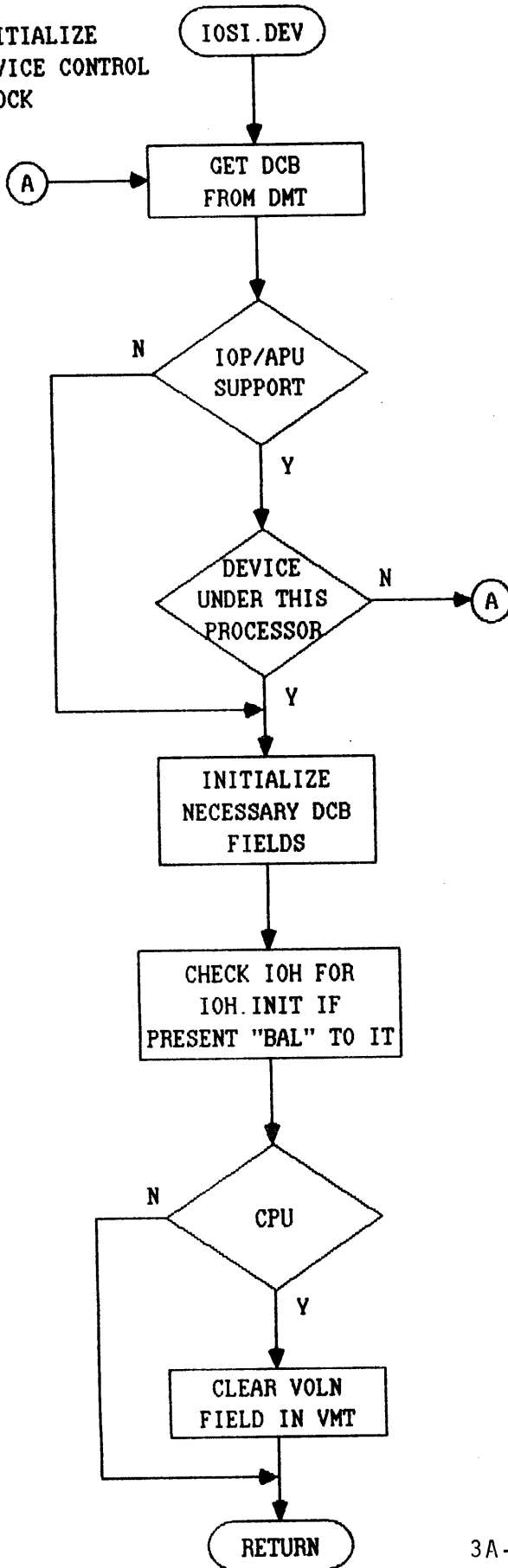
PRINT MESSAGE ON
SYSTEM CONSOLE
REQUESTING OPERATOR
TO PLACE BUS-SWITCHES
ON THE IOPs TO AUTO



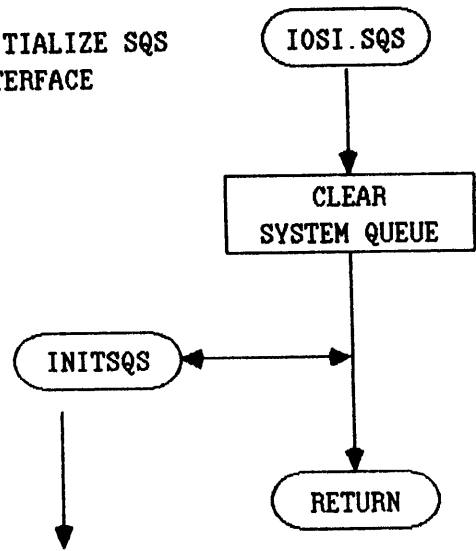
INITIALIZE IOPs



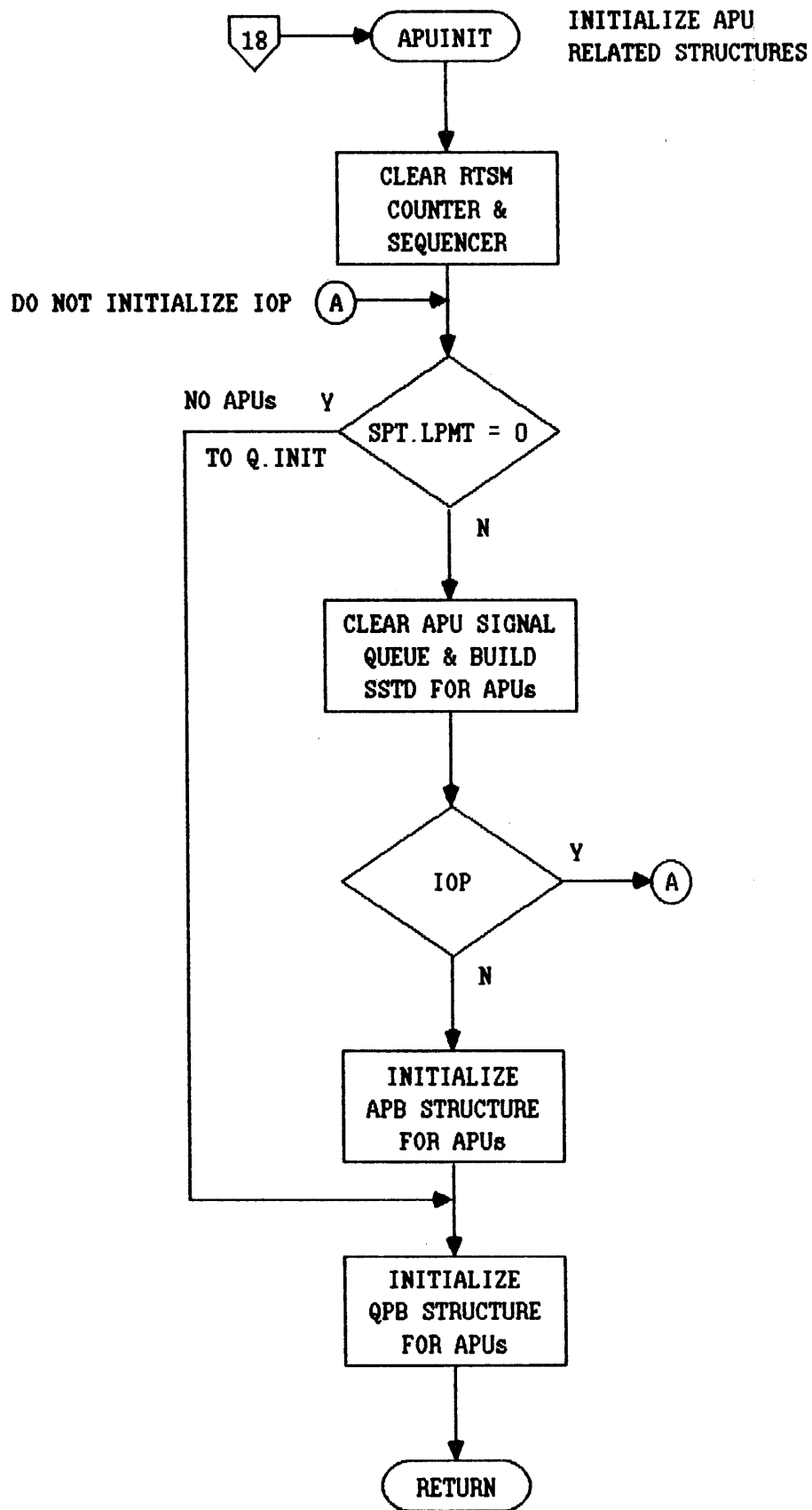
INITIALIZE
DEVICE CONTROL
BLOCK

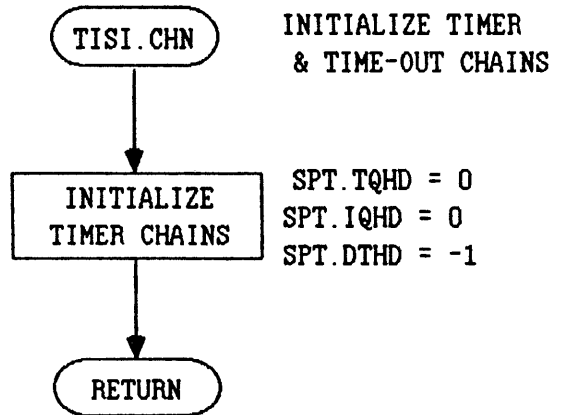
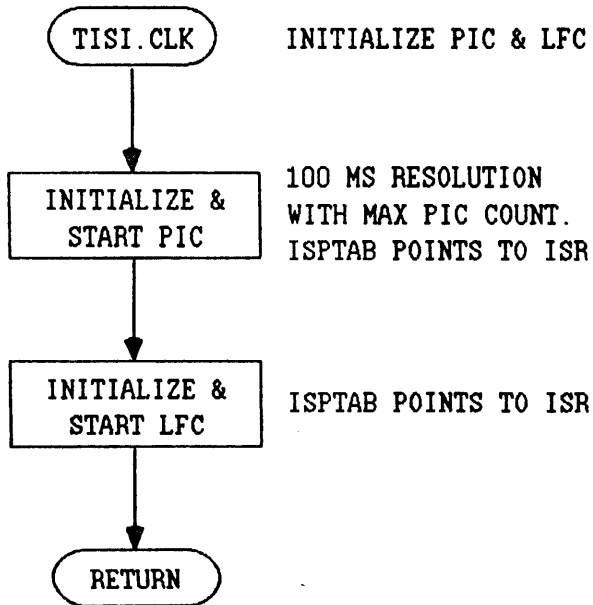
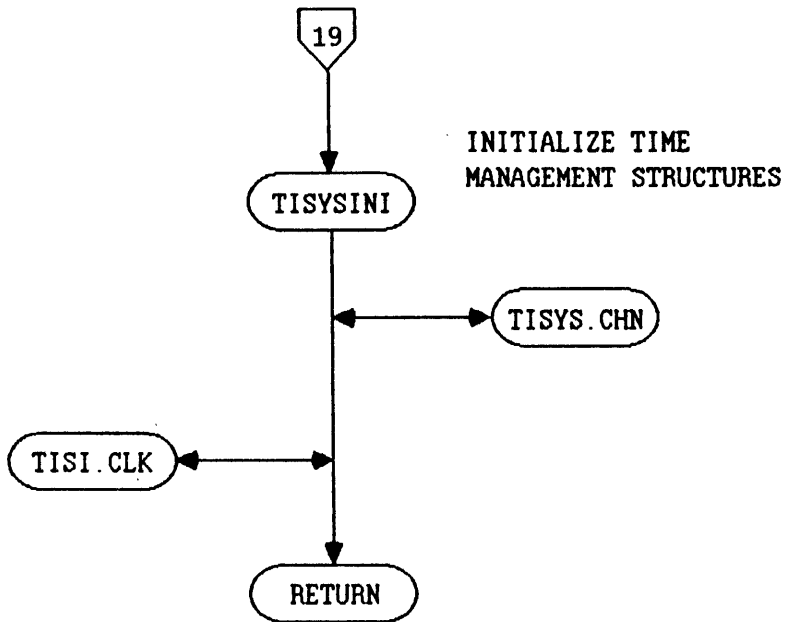


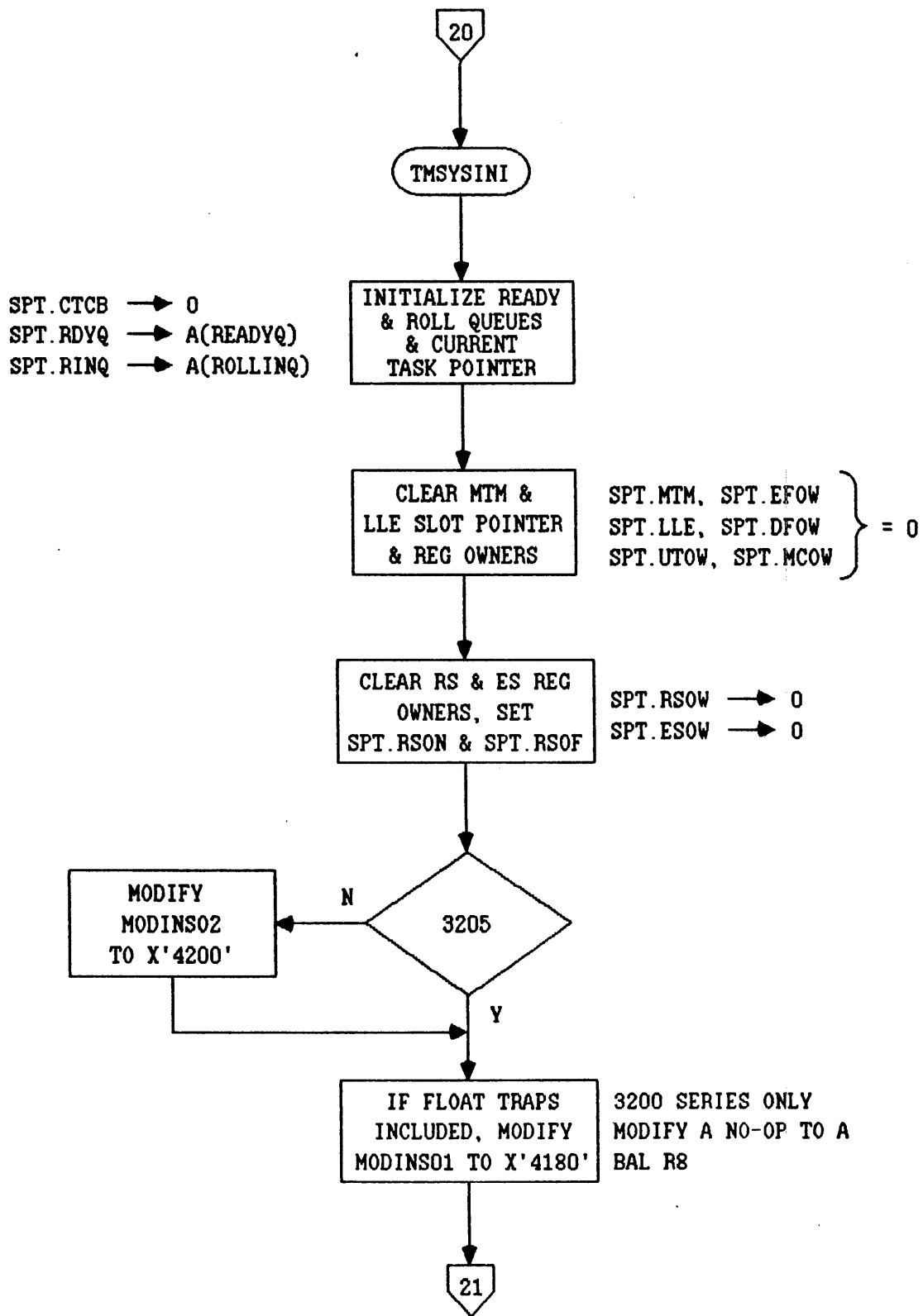
INITIALIZE SQS
INTERFACE



ESTABLISH PSWs FOR ENTERING SQS FROM
QUEUE SERVICE INTERRUPT OR FROM TASK
MANAGER AND SET UP LOCATION TO VECTOR
FROM SQS TO TASK MANAGER



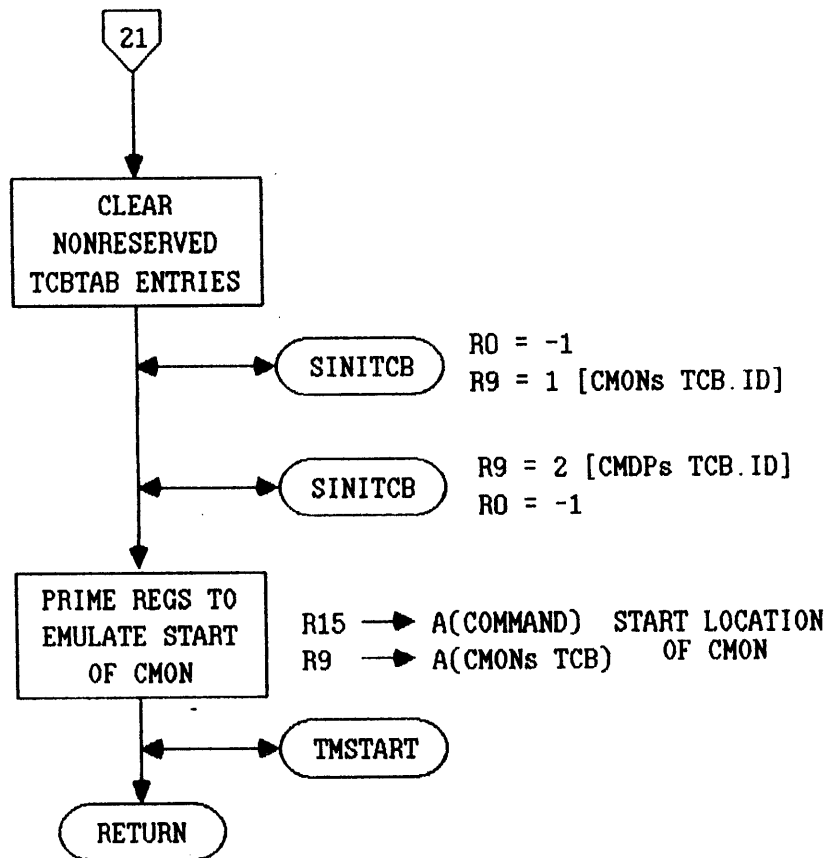




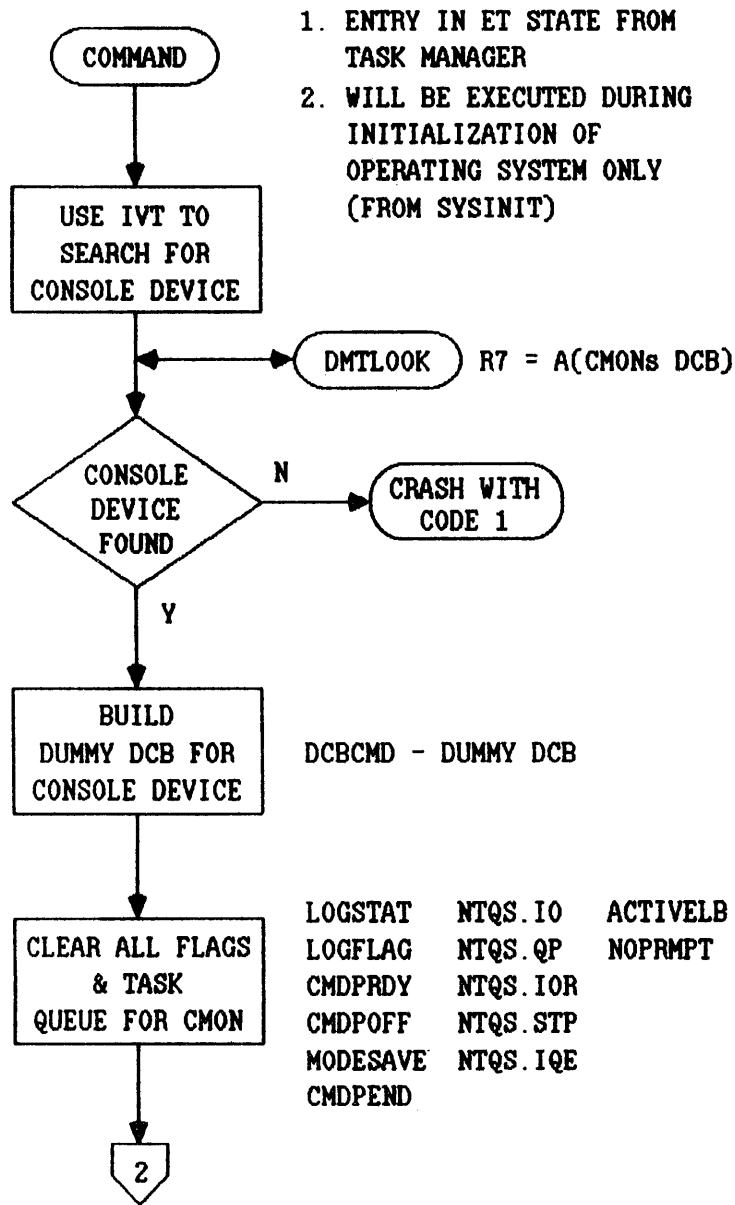
RESULTS OF SINITCB:

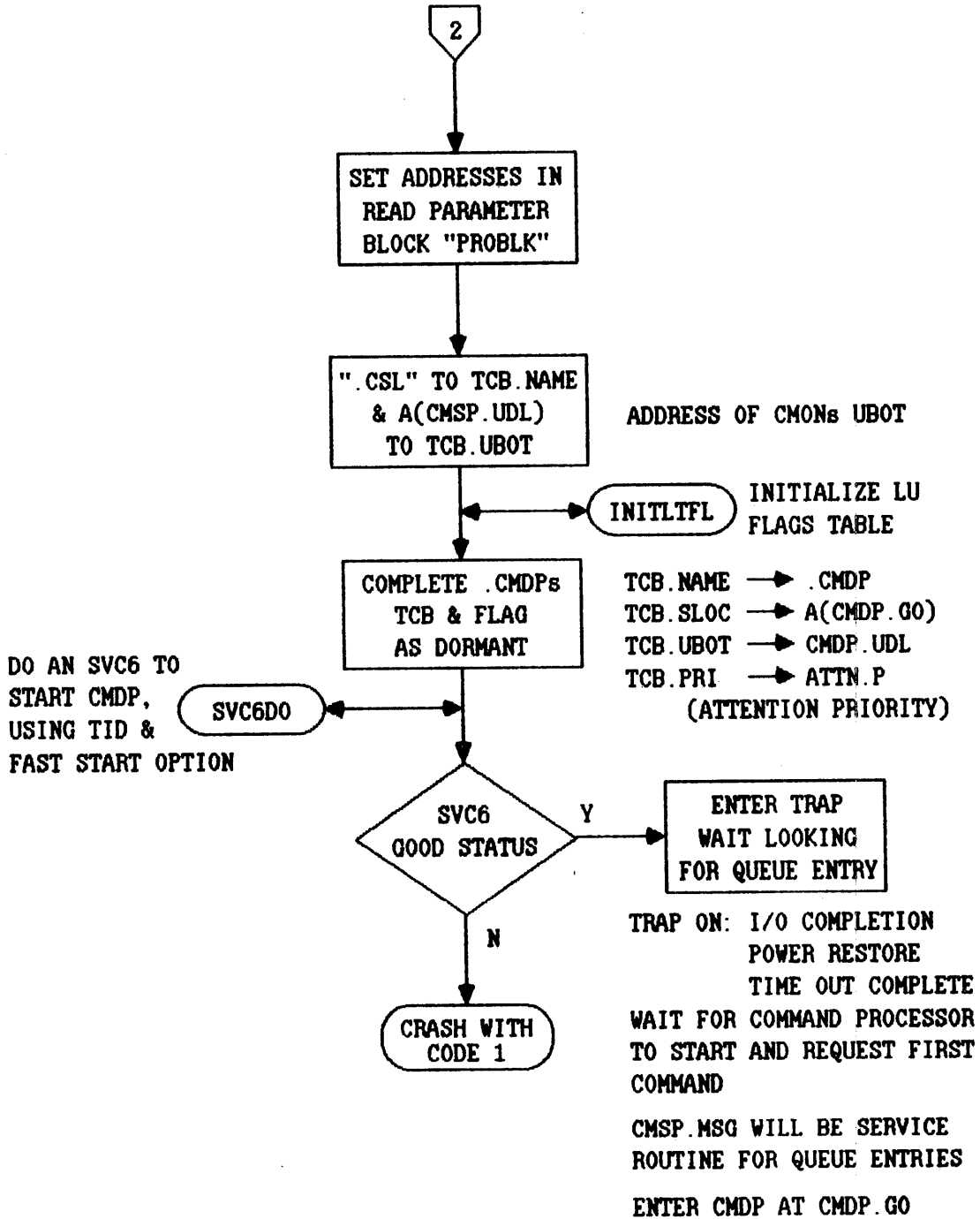
R0 → -1
 R2 → 0
 R9 SLOT # IN TCBTAB

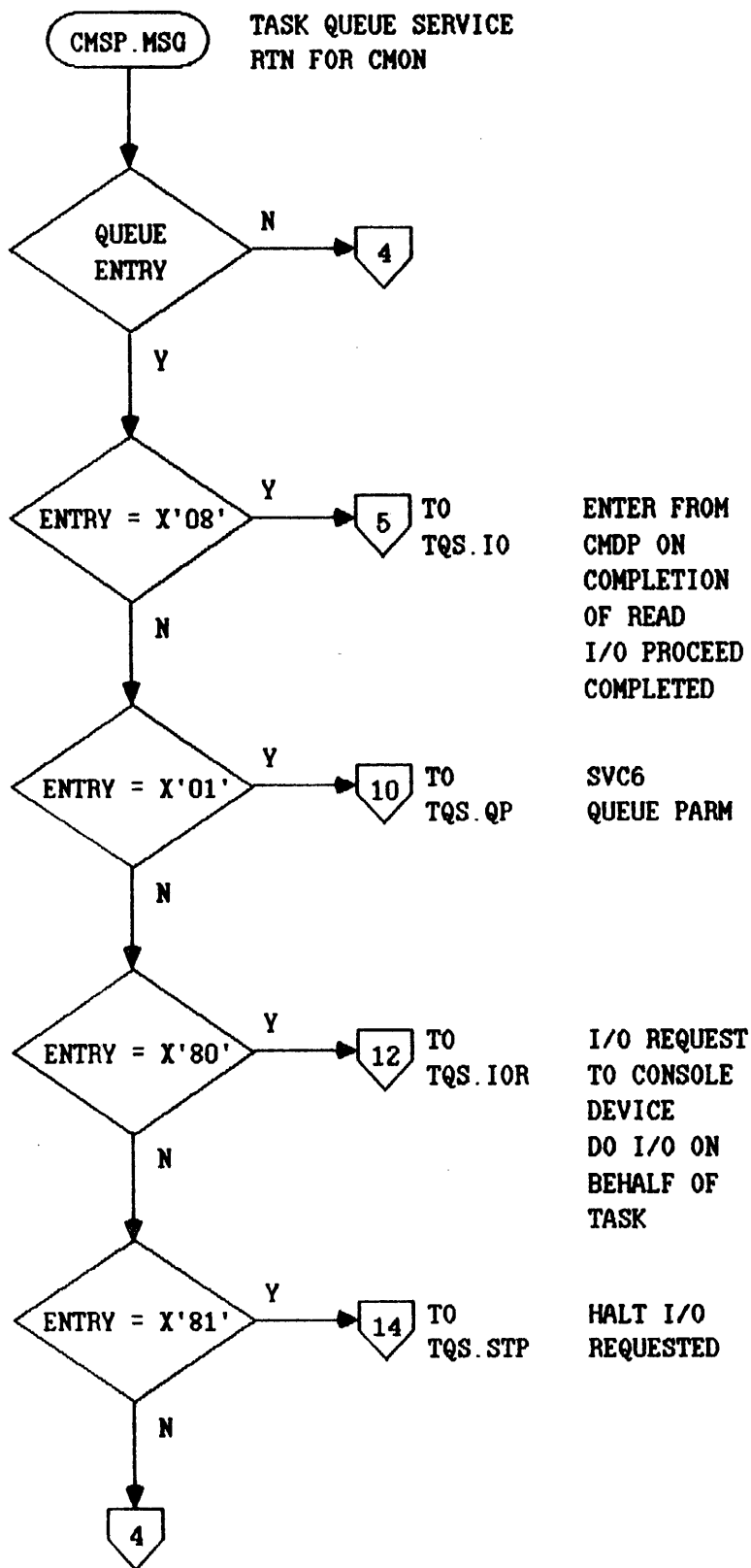
1. CLEAR TCB TO 0
2. SET TCB.NLU
3. SET TCB.UCTX TO POINT TO CONTEXT BLOCK
4. TCB.MPRI, TCB.PRI, TCB.DPRI, TCB.RPRI & TCB.IOB + IOB.PRI SET TO CONTENTS OF R9
5. TCB.ID + TCB.GID SET TO CONTENTS OF R9
6. TCB.IOBL = A(TCB.IOB)
7. TCB.OPT INITIALIZED
8. TCB.MXSP = -1 [UNLIMITED SYSTEM SPACE]

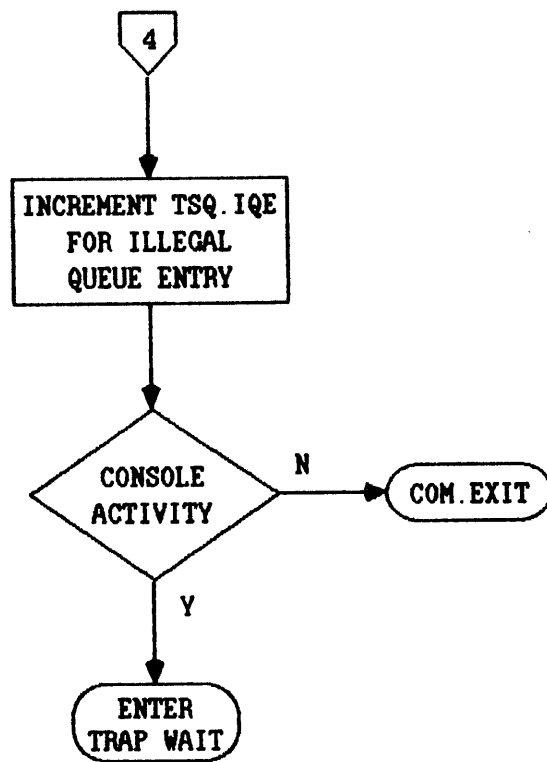


CONSOLE SUPPORT



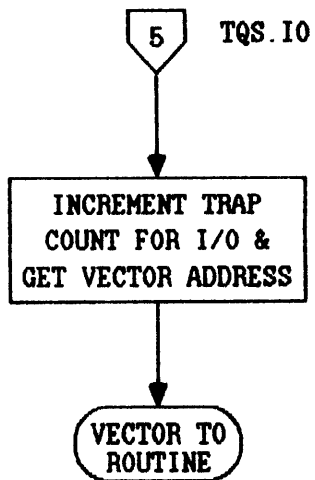


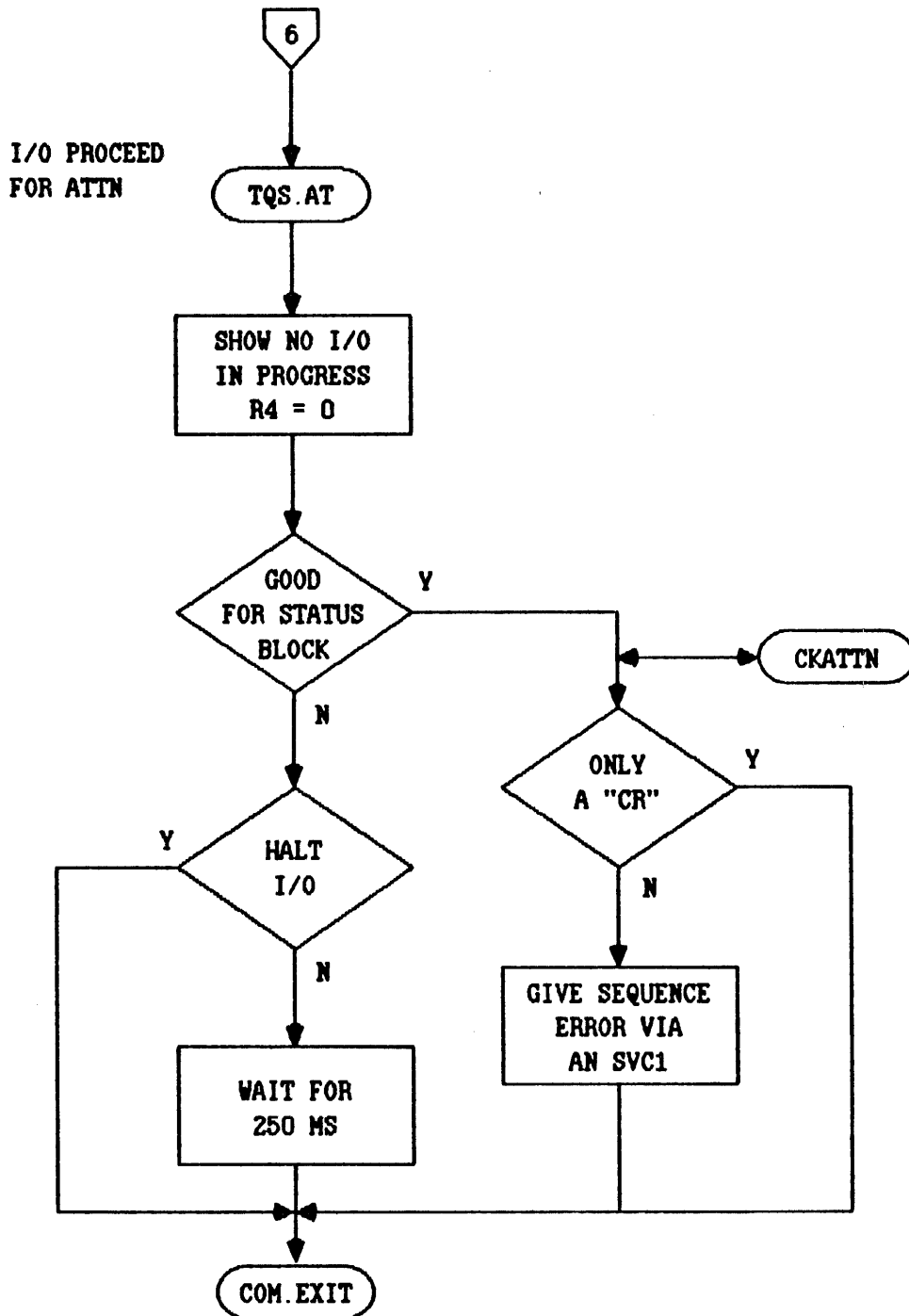


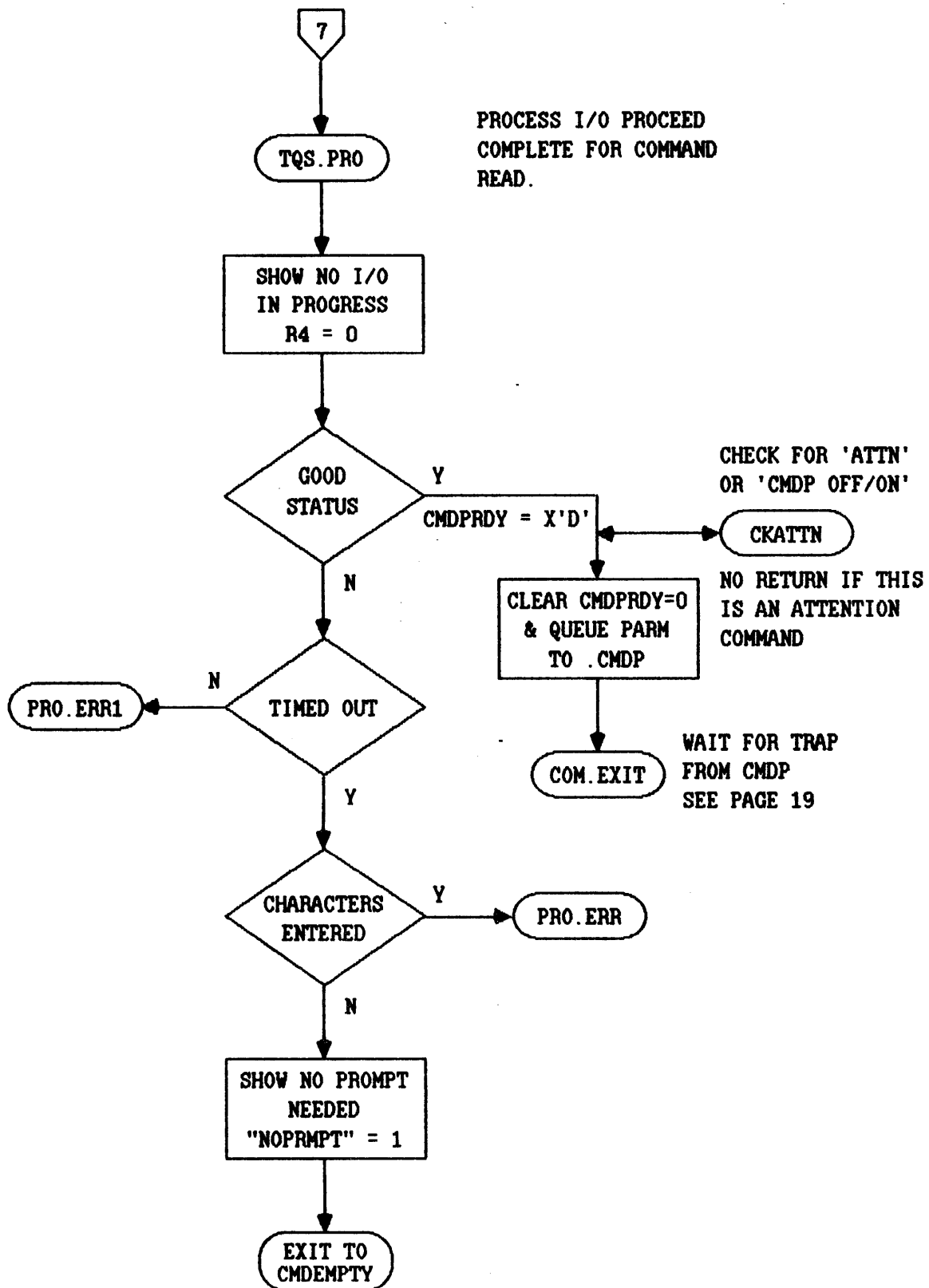


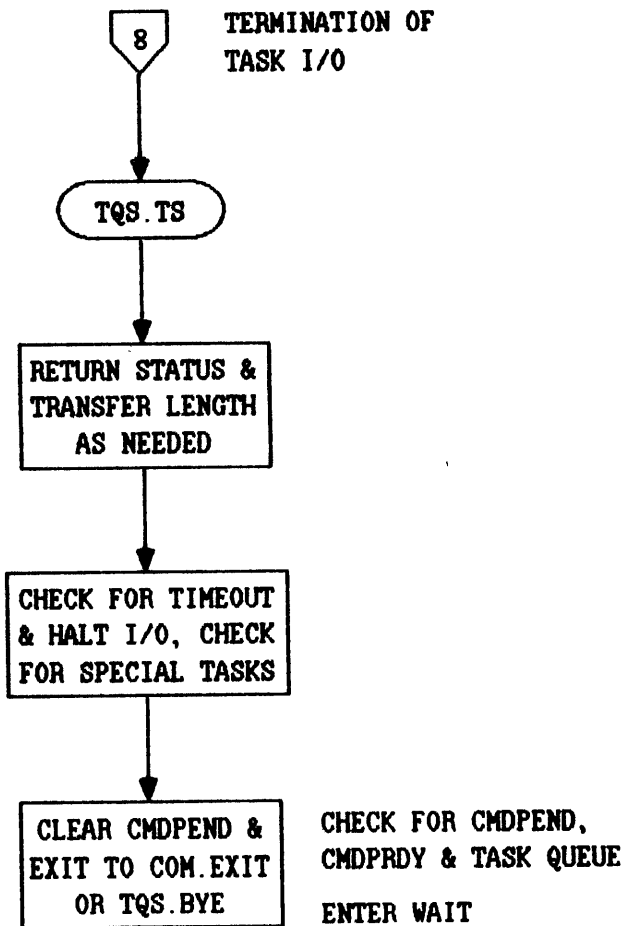
GENERAL I/O PROCEED
COMPLETED. GET TO
PROPER EXECUTOR.

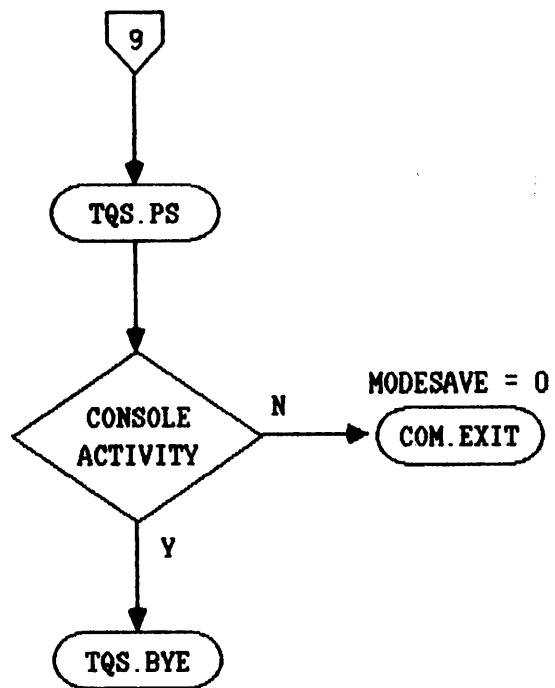
RF	→	A(PCB)
6	TQS. AT	→ ATTENTION READ
7	TQS. PRO	→ COMMAND READ
8	TQS. TS	→ TASK I/O
9	TQS. PS	→ POWER FAIL

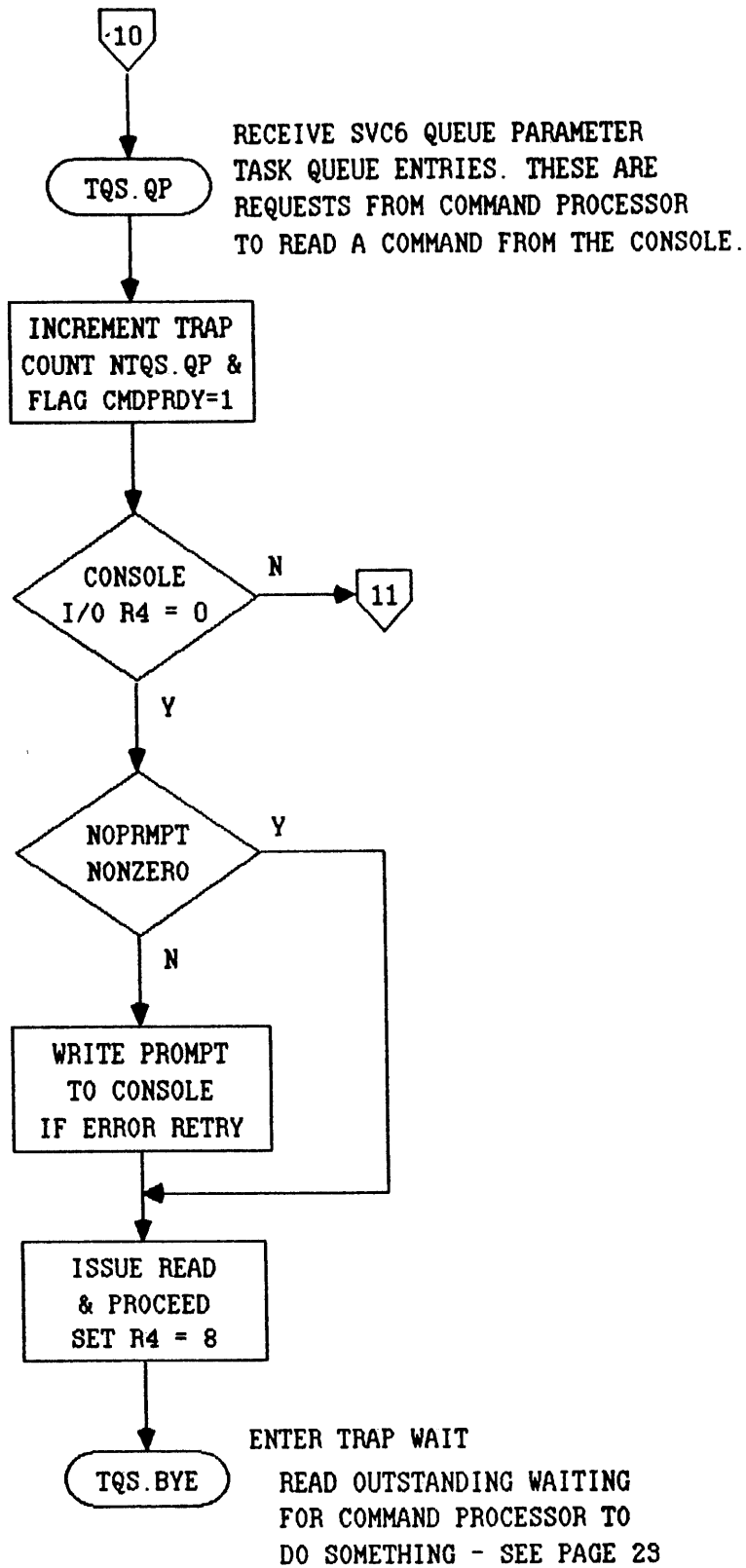


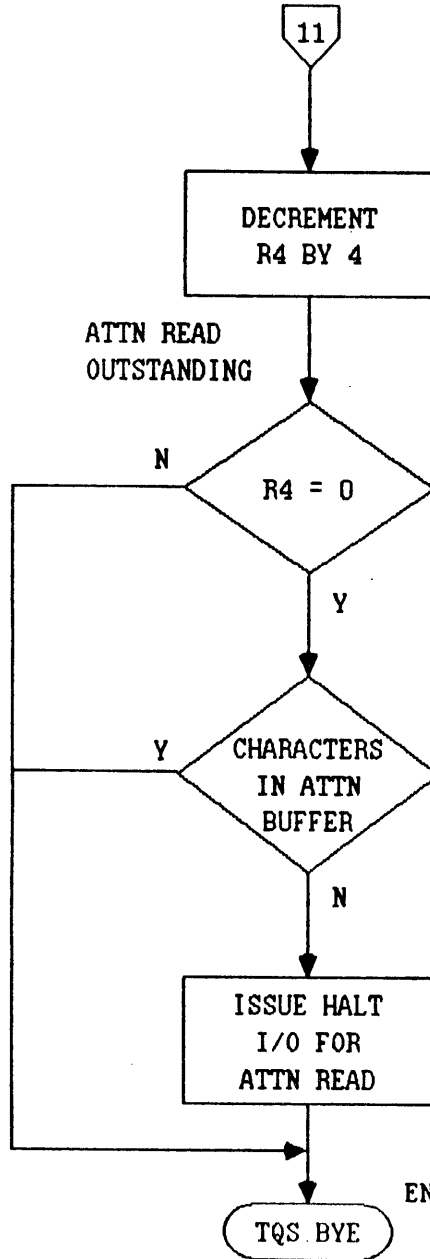






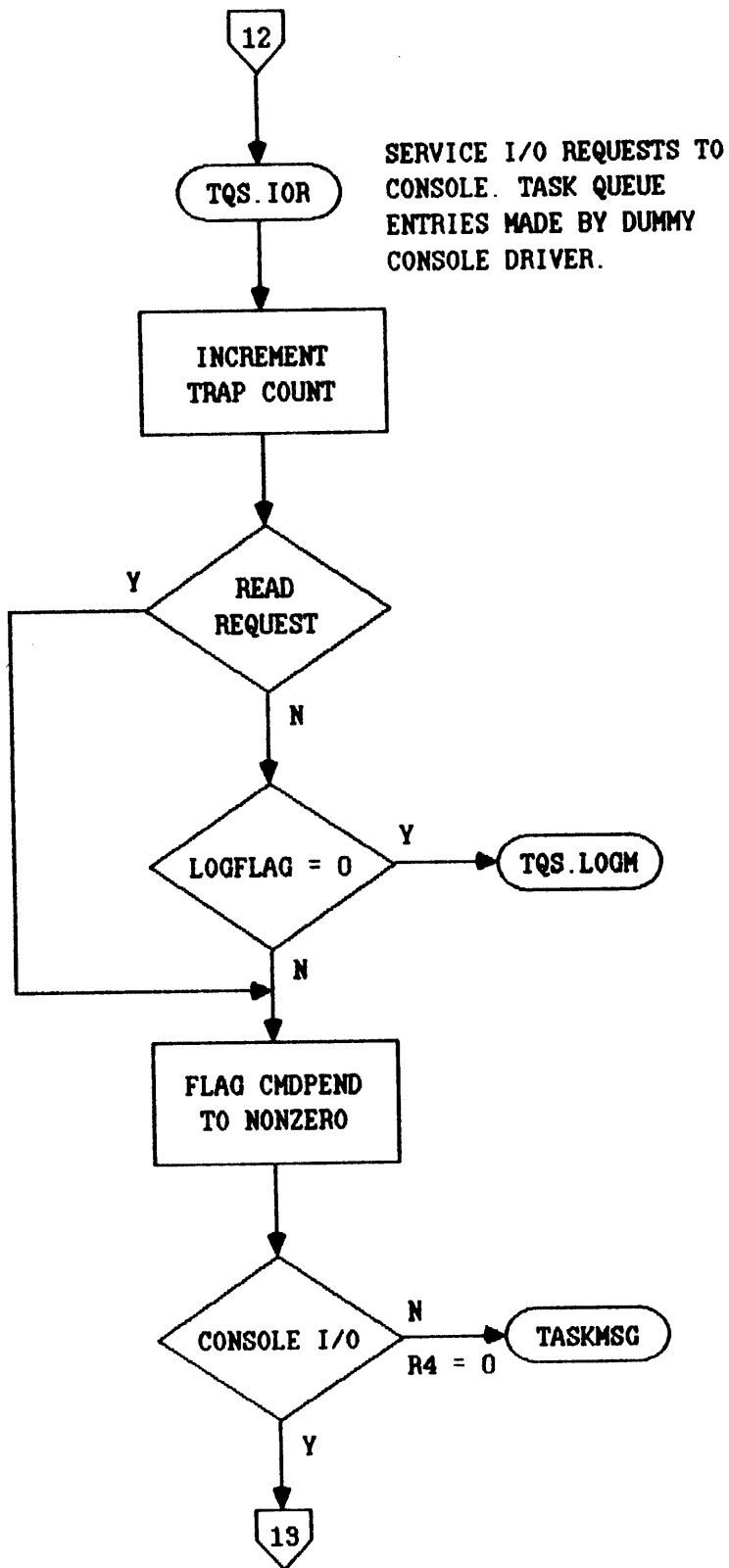


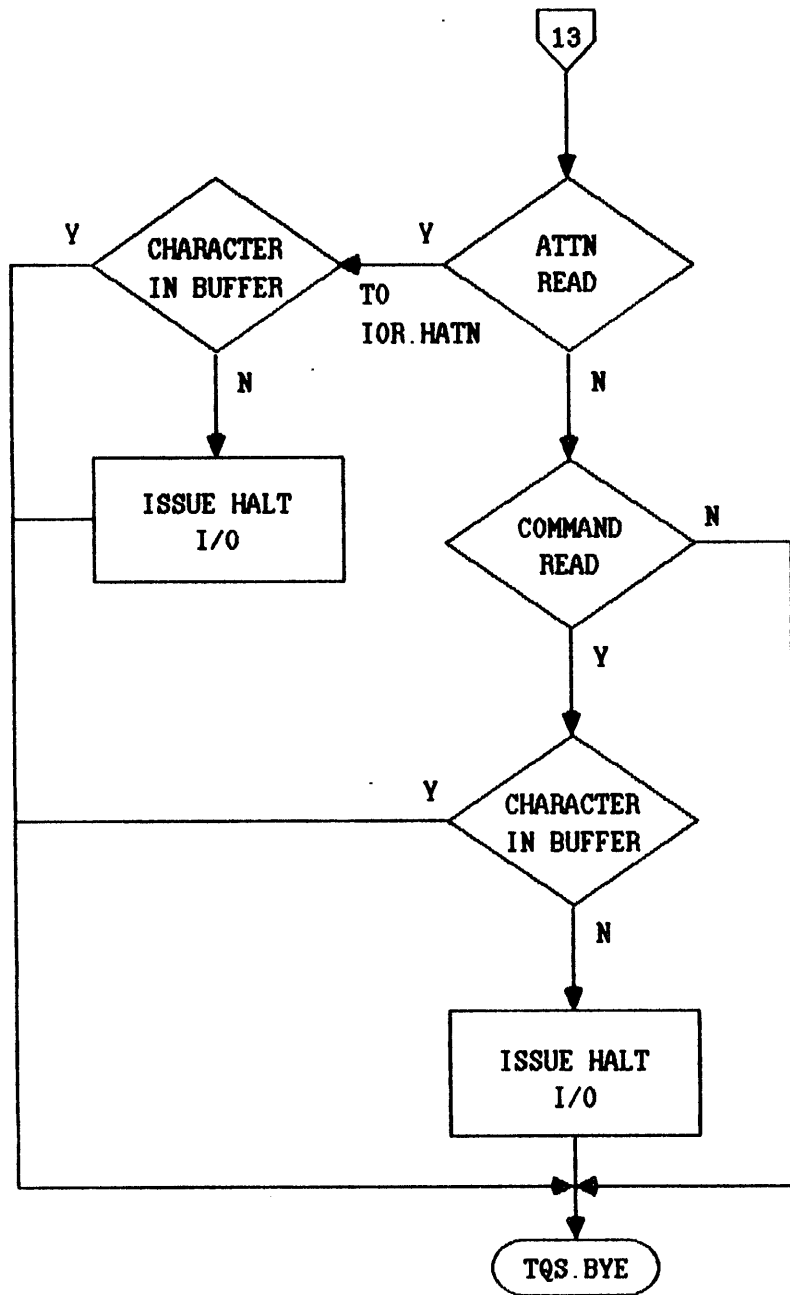


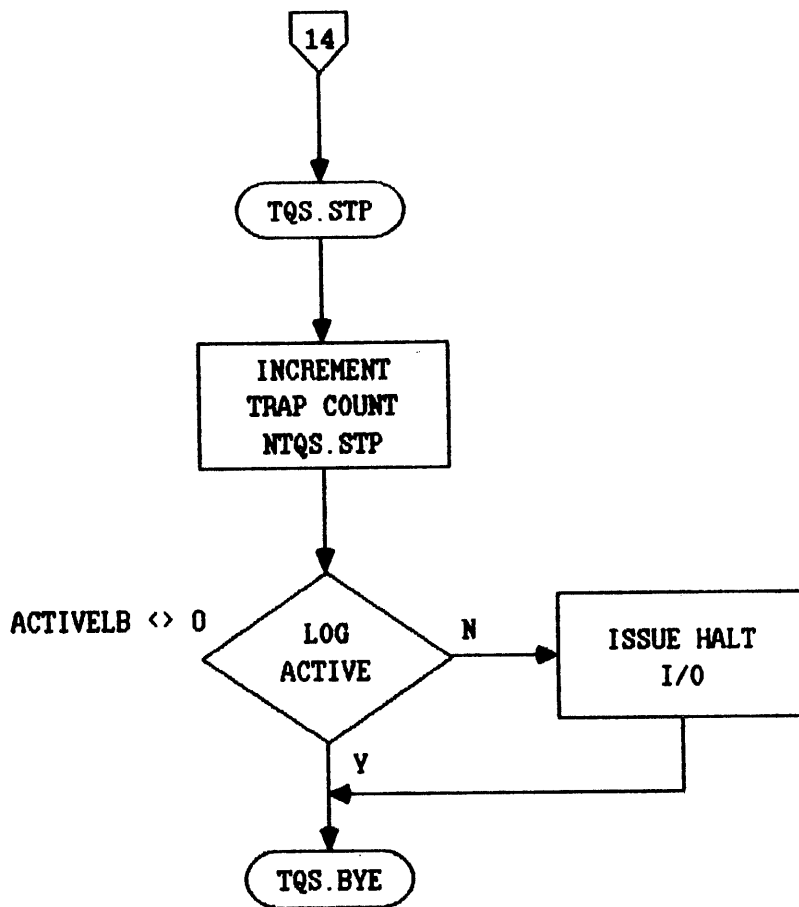


ENTER TRAP WAIT

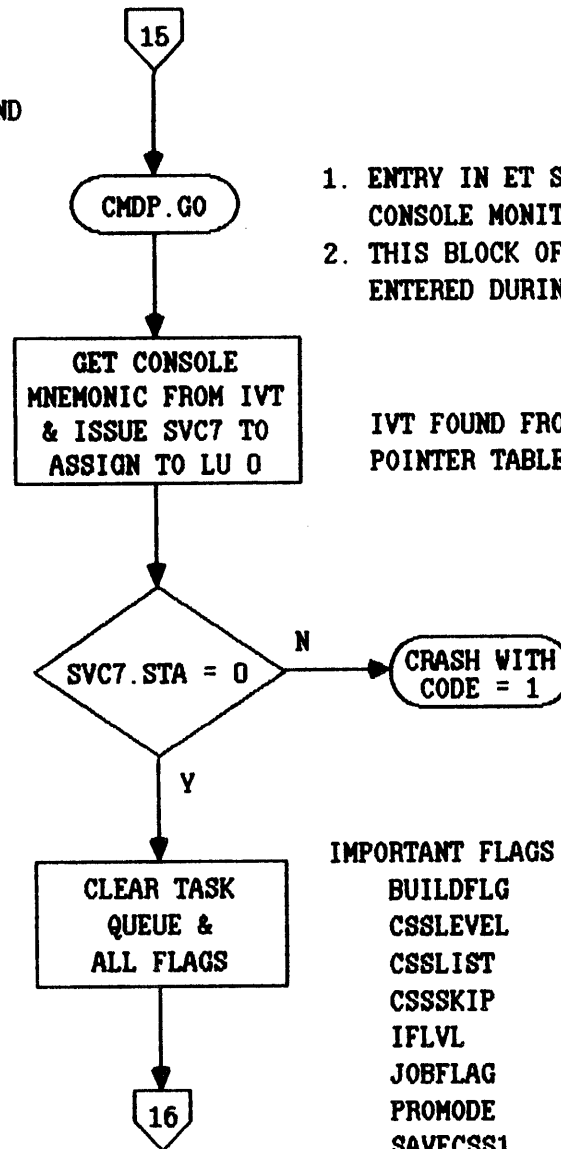
READ OUTSTANDING WAITING
FOR COMMAND PROCESSOR TO
DO SOMETHING - SEE PAGE 23







ENTRY INTO COMMAND
PROCESSOR

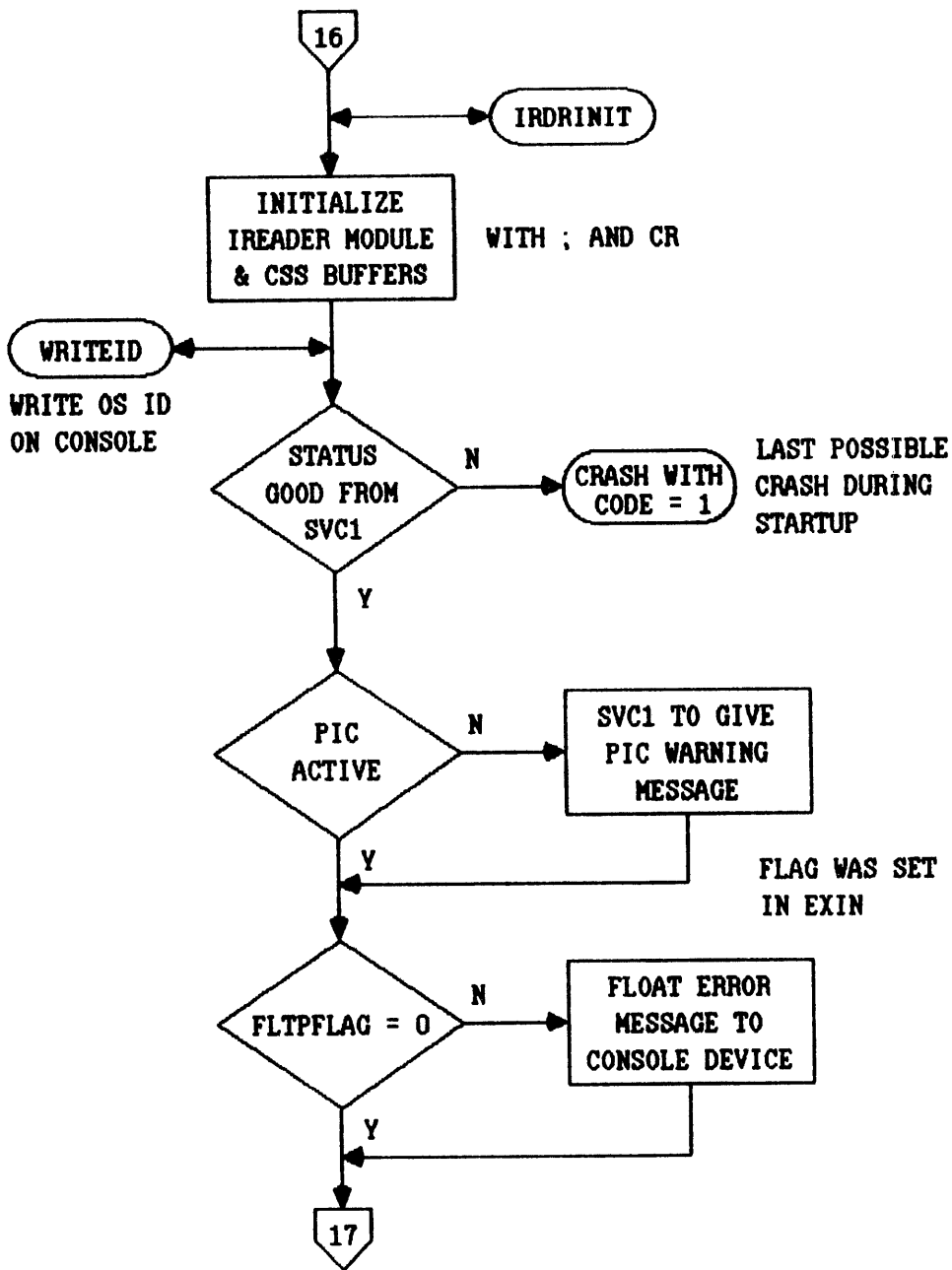


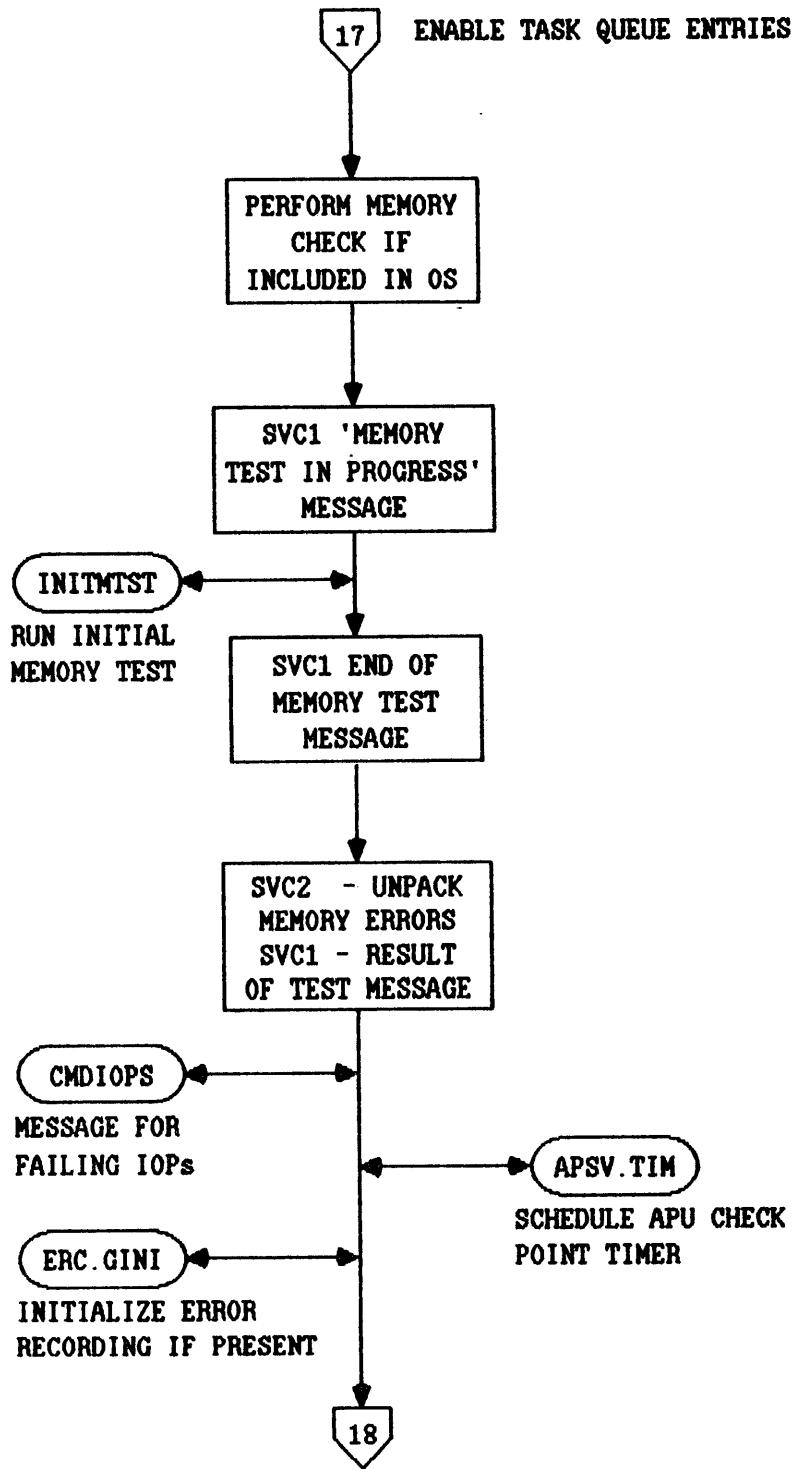
1. ENTRY IN ET STATE FROM CONSOLE MONITOR TASK
2. THIS BLOCK OF CODE IS ONLY ENTERED DURING OS INITIALIZATION

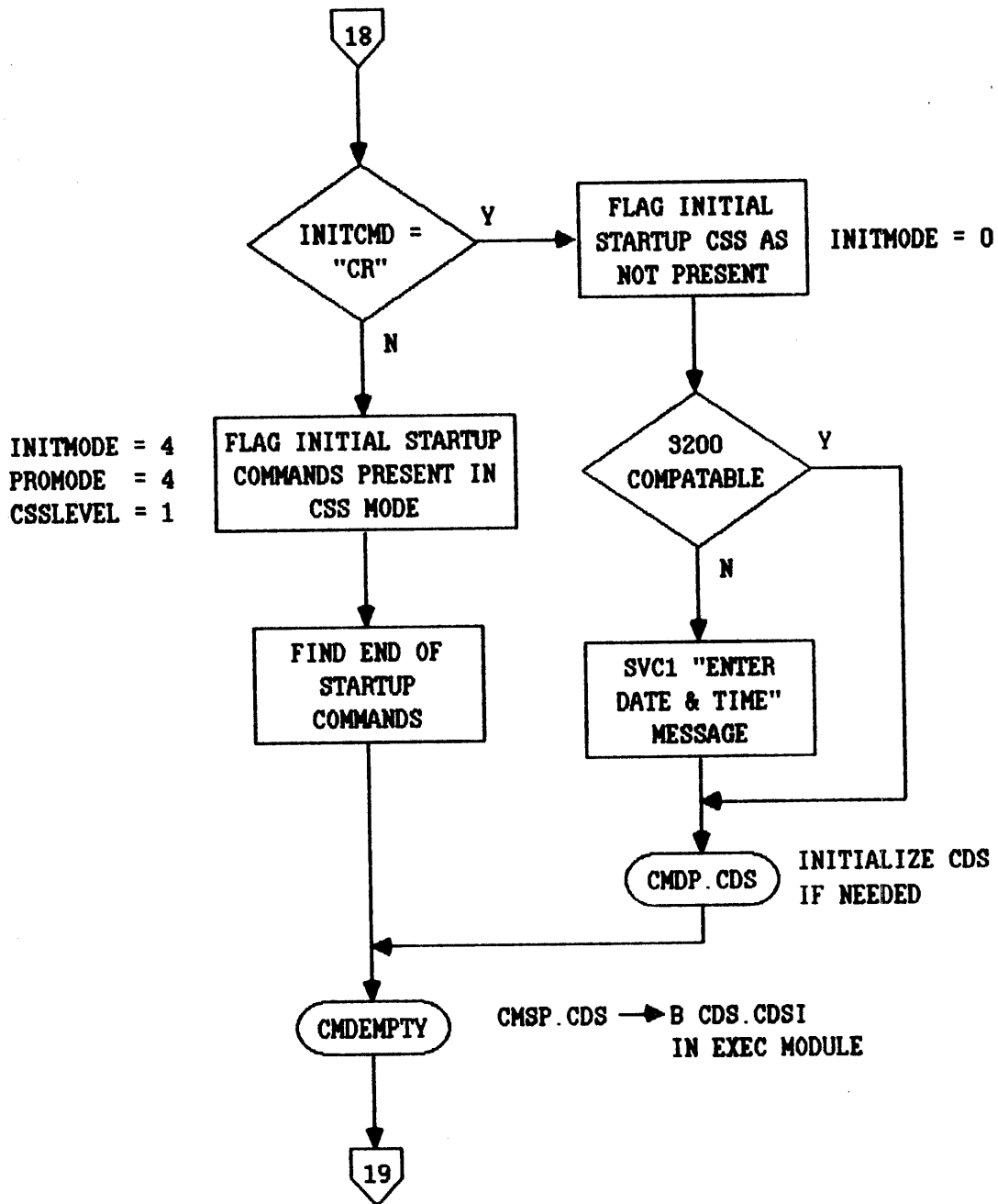
IVT FOUND FROM SYSTEM
POINTER TABLE POINTER

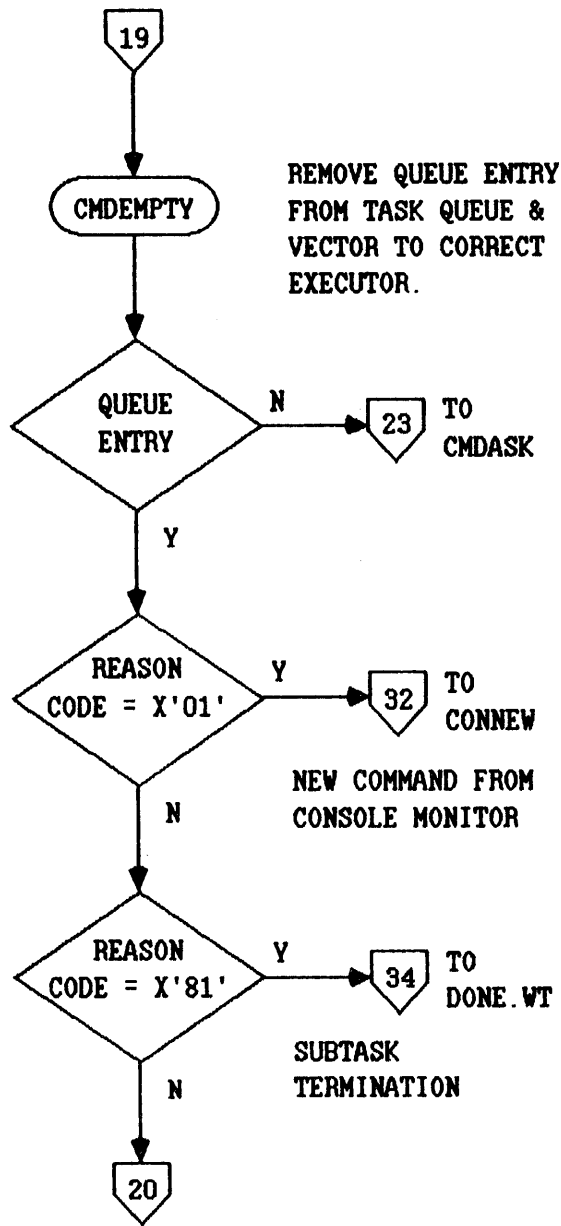
IMPORTANT FLAGS TO NOTE:

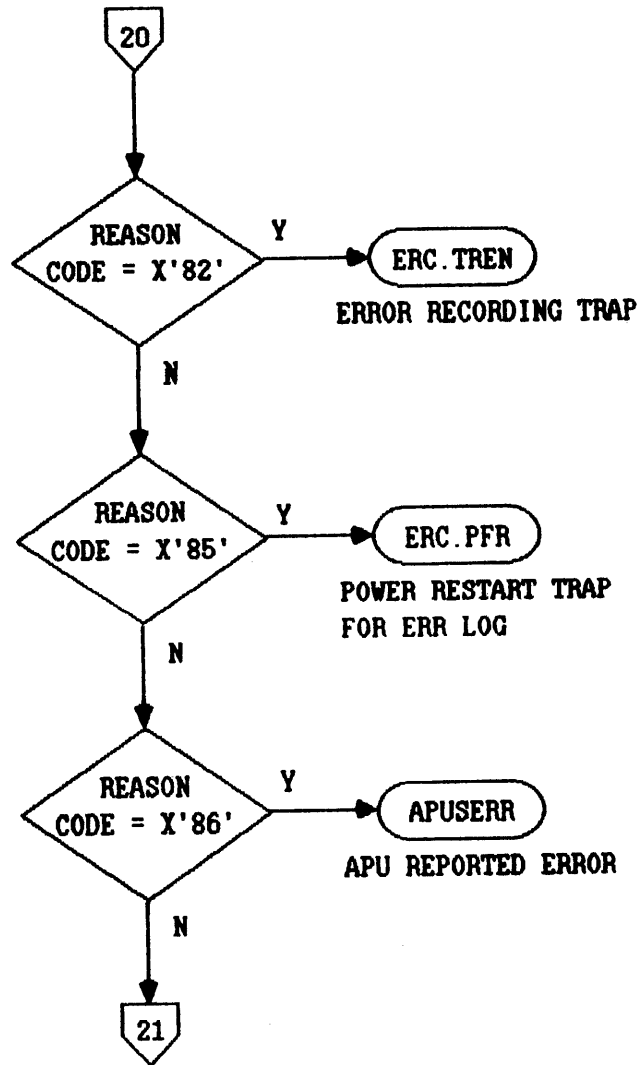
BUILDFLG
CSSLEVEL
CSSLIST
CSSSKIP
IFLVL
JOBFLAG
PROMODE
SAVECSS1

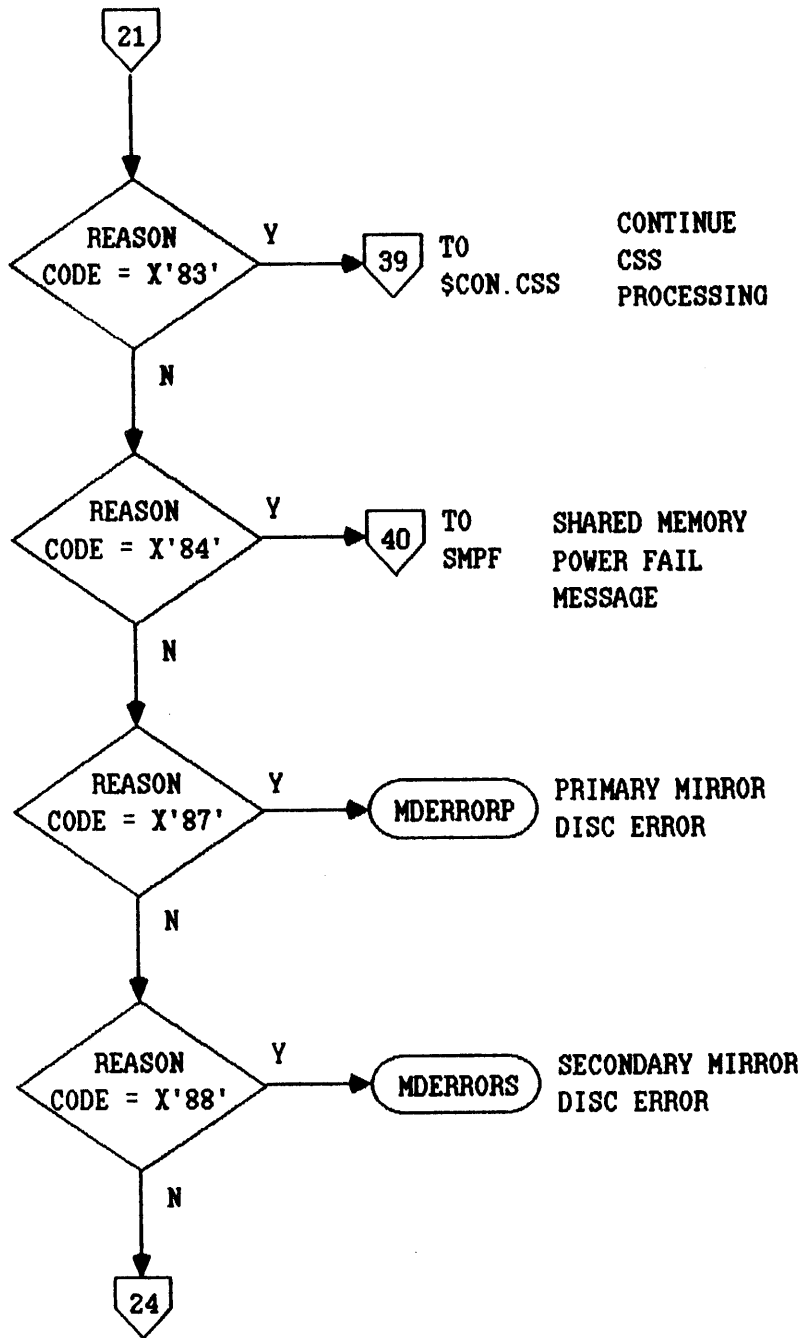


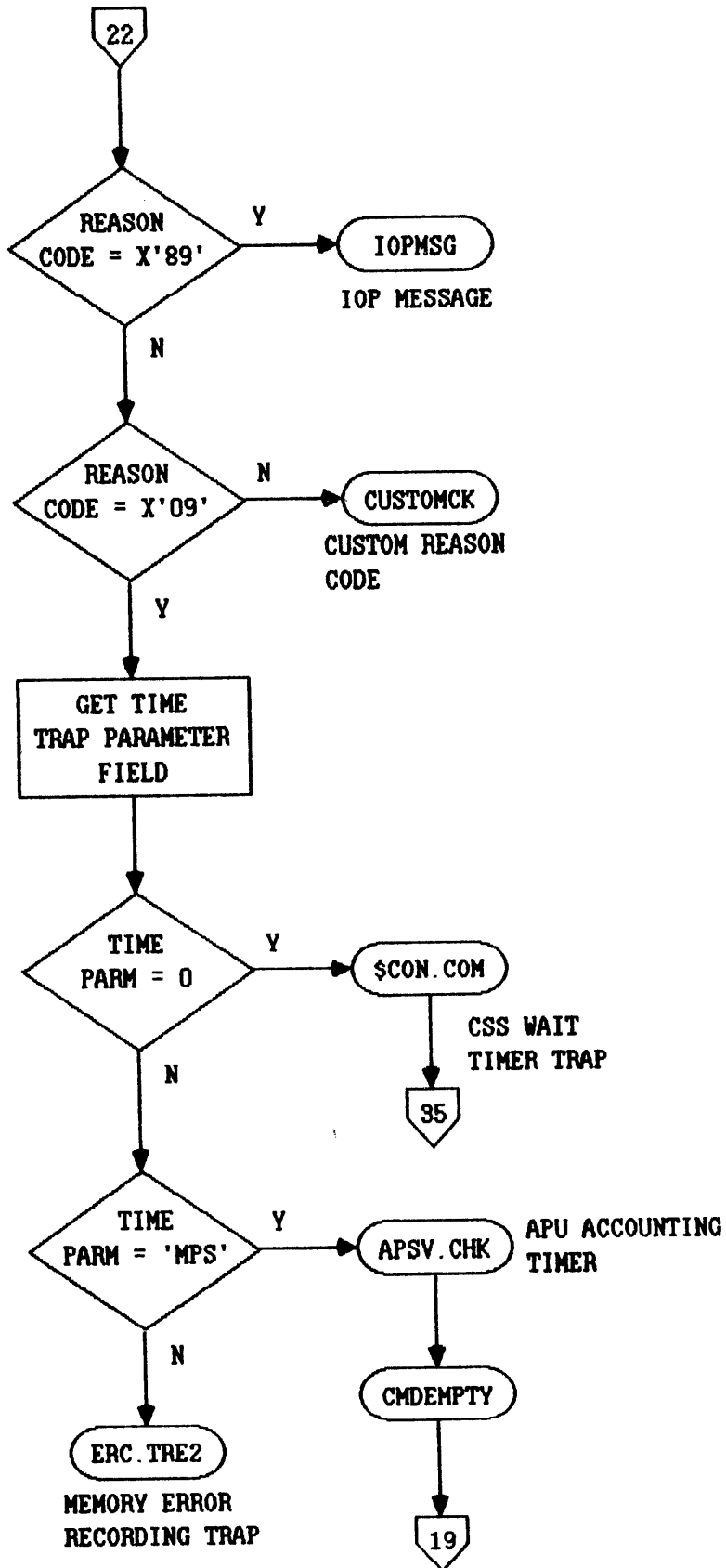






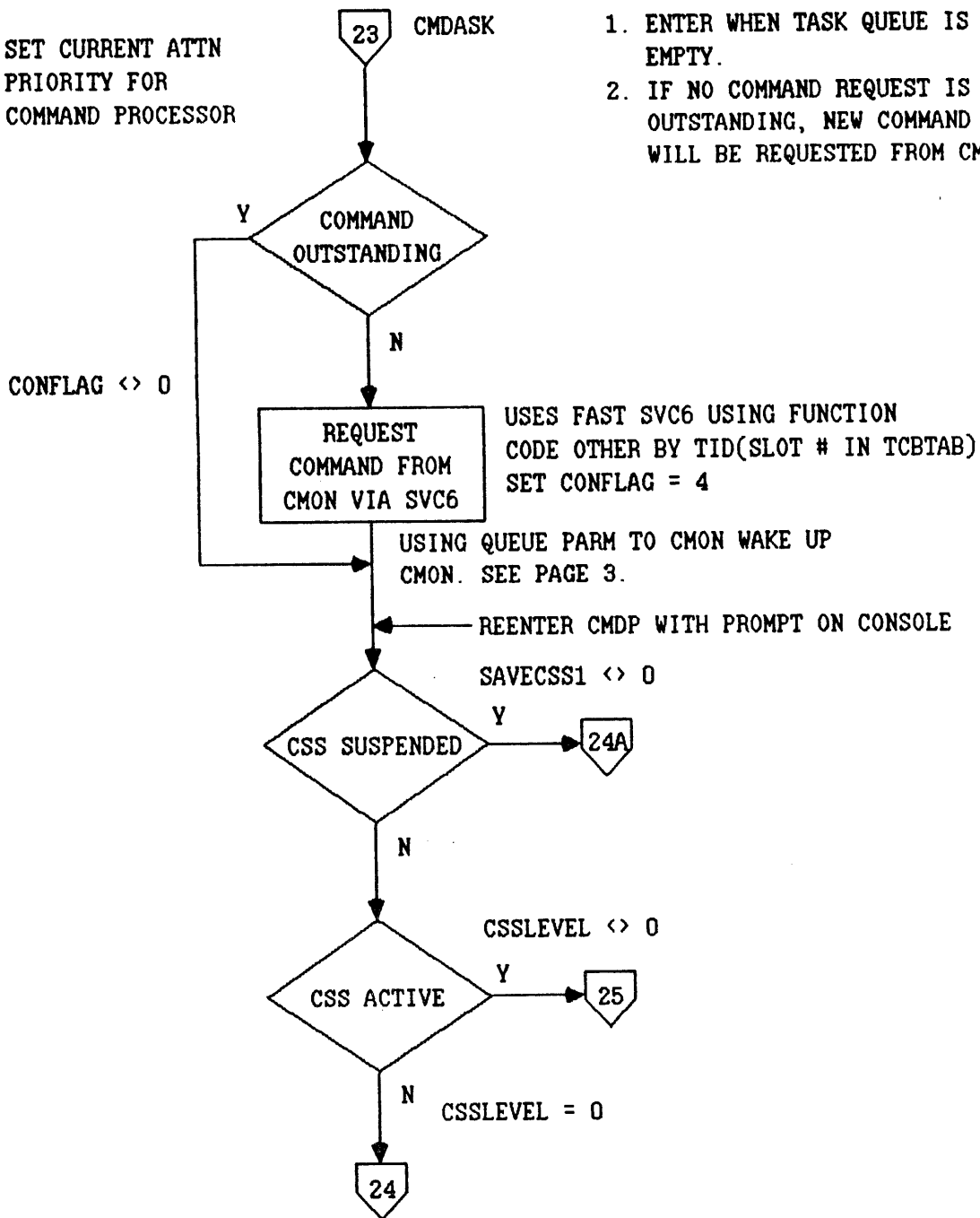


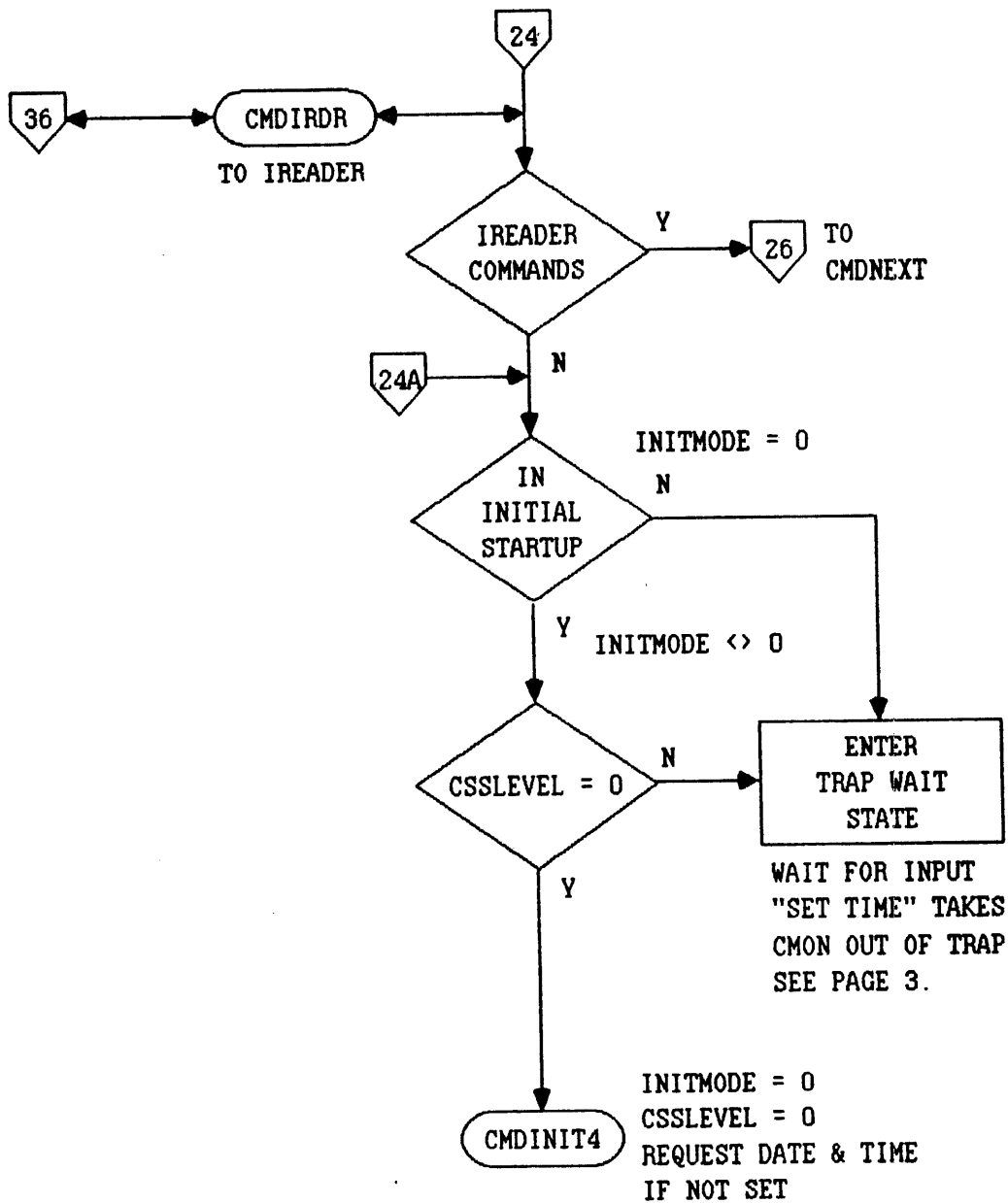


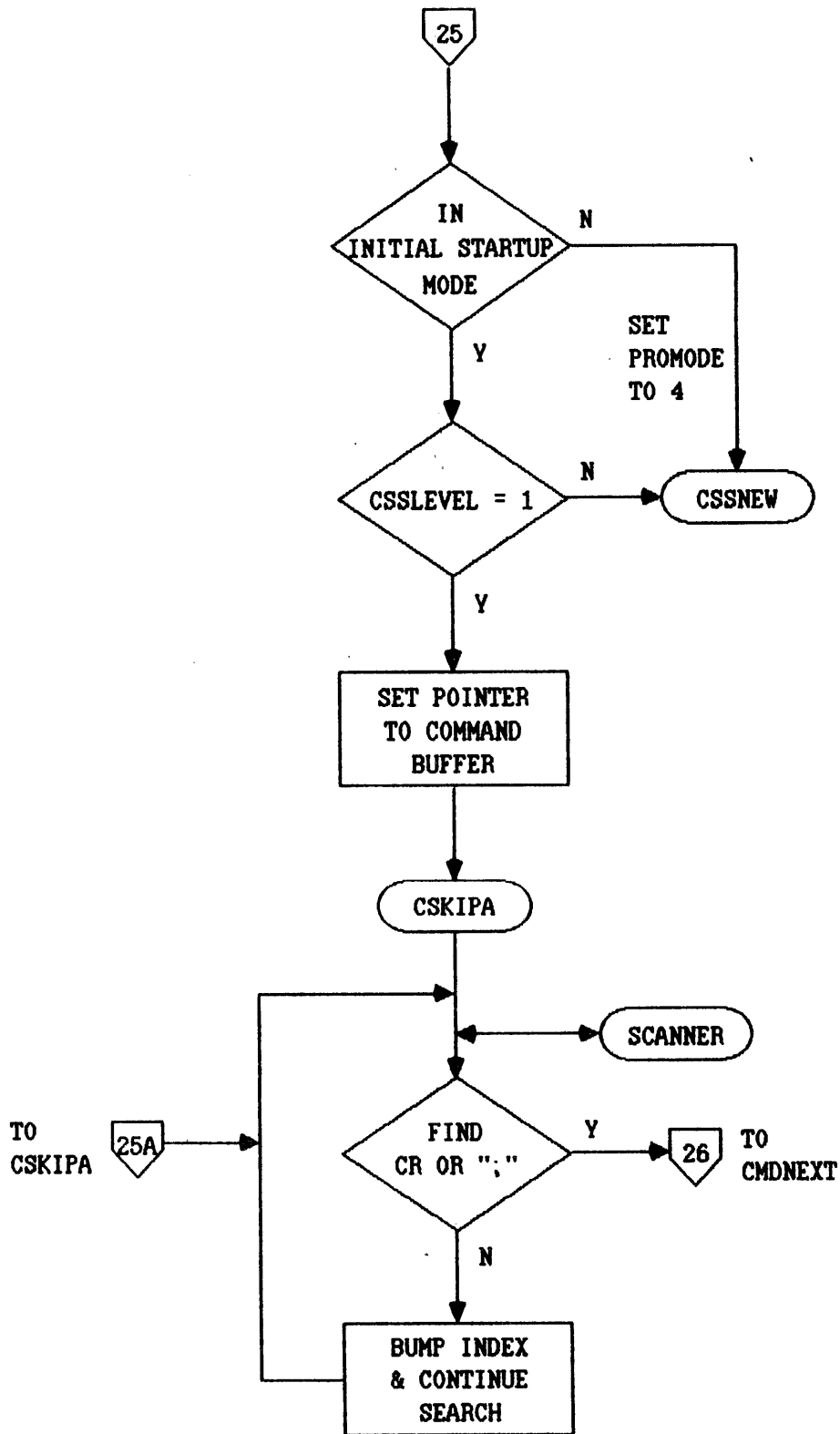


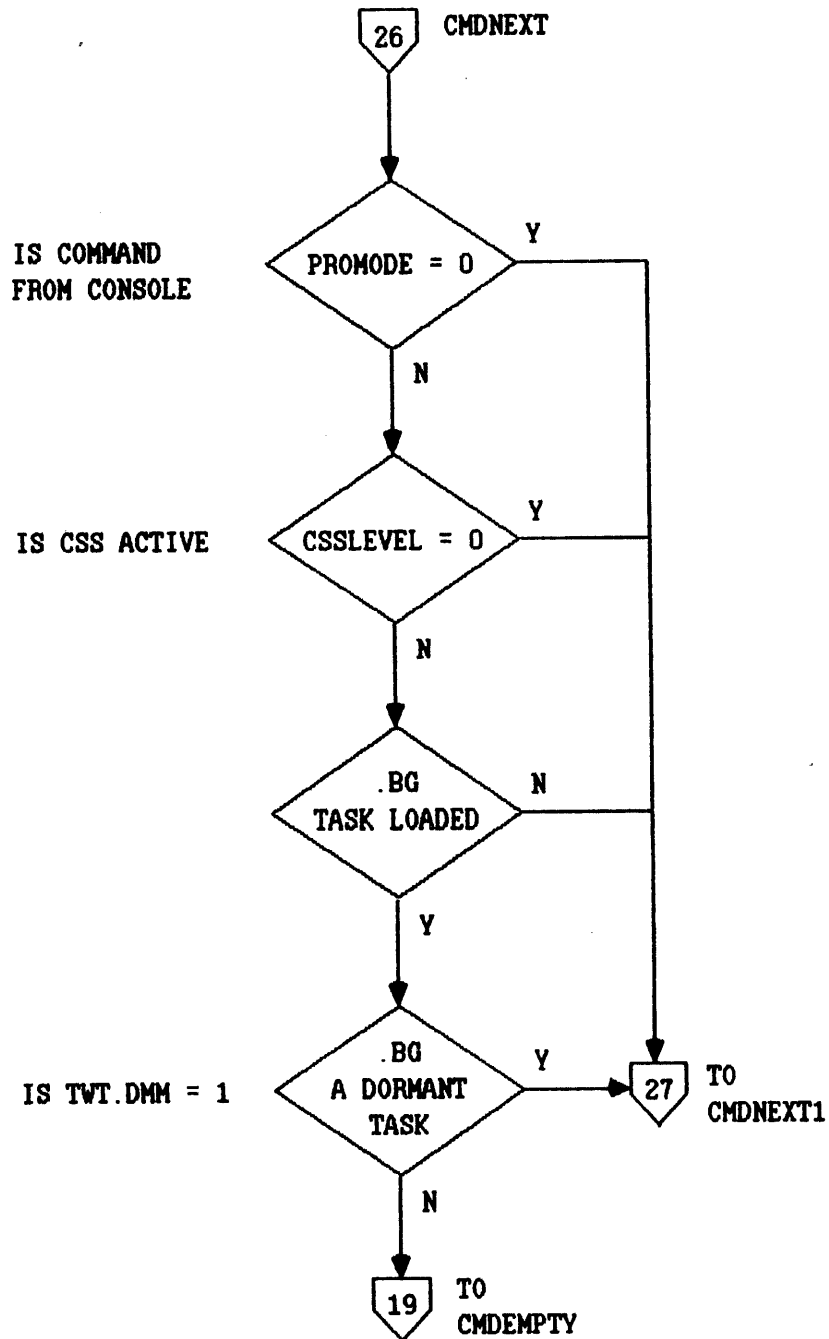
SET CURRENT ATTN
PRIORITY FOR
COMMAND PROCESSOR

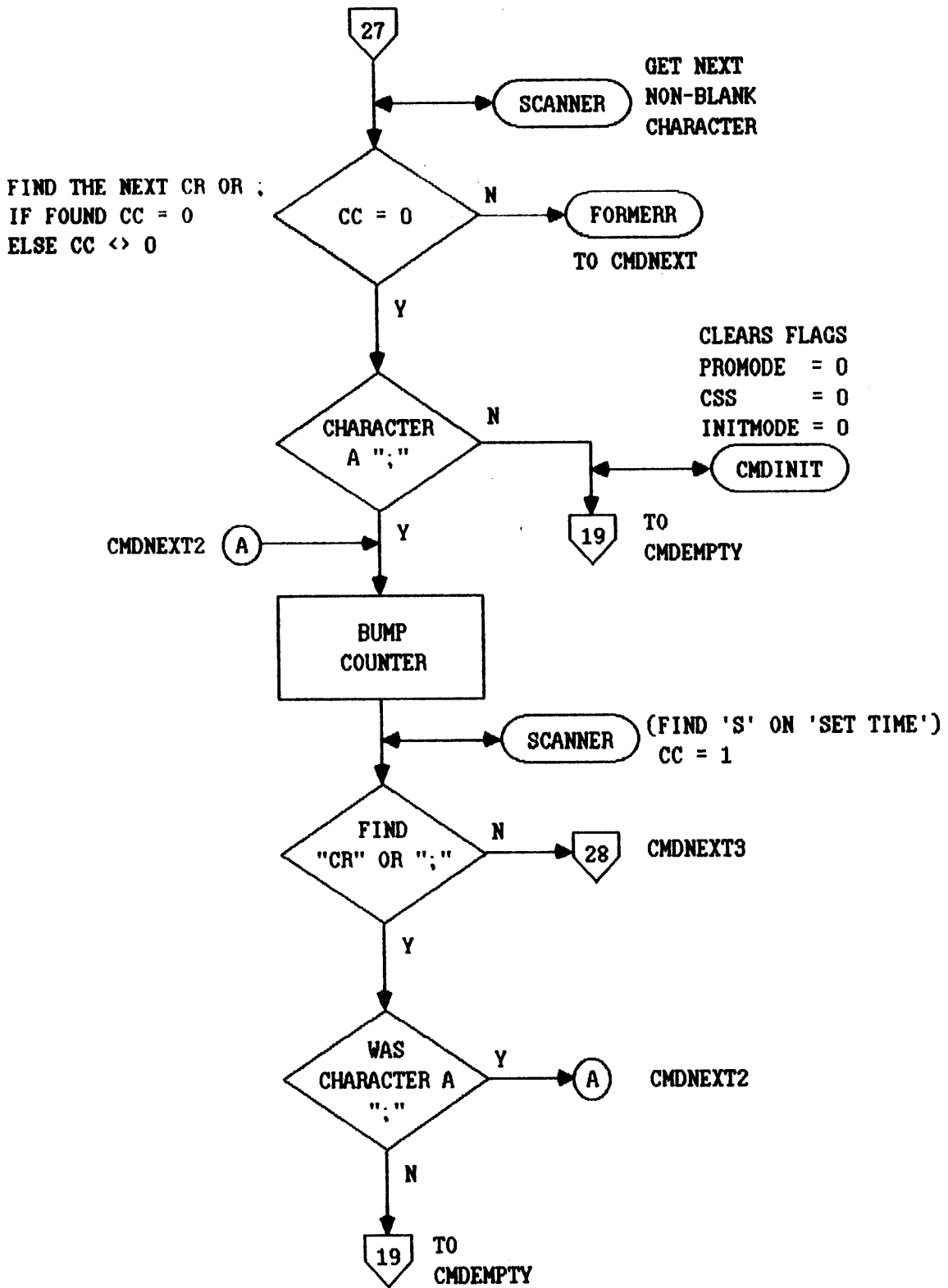
1. ENTER WHEN TASK QUEUE IS EMPTY.
2. IF NO COMMAND REQUEST IS OUTSTANDING, NEW COMMAND WILL BE REQUESTED FROM CMON.



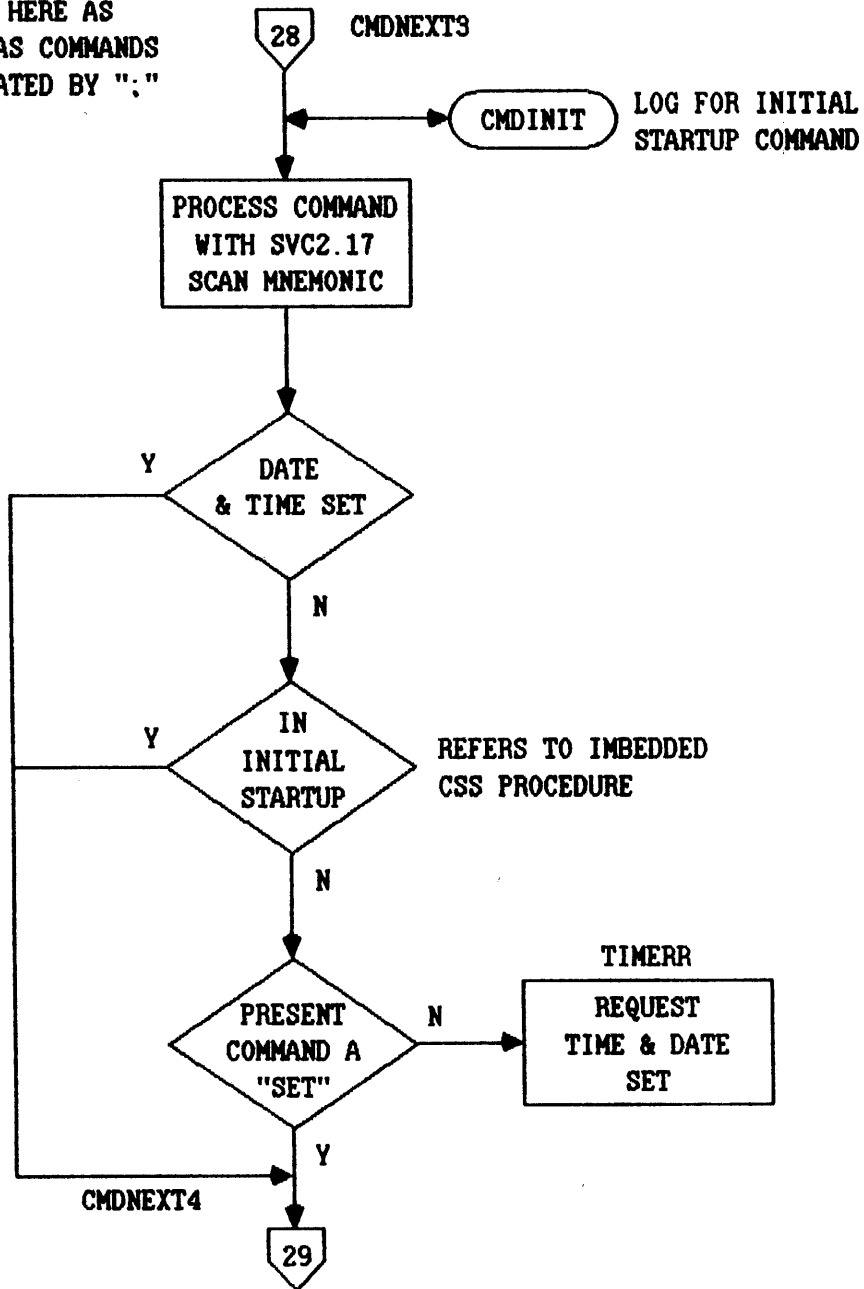


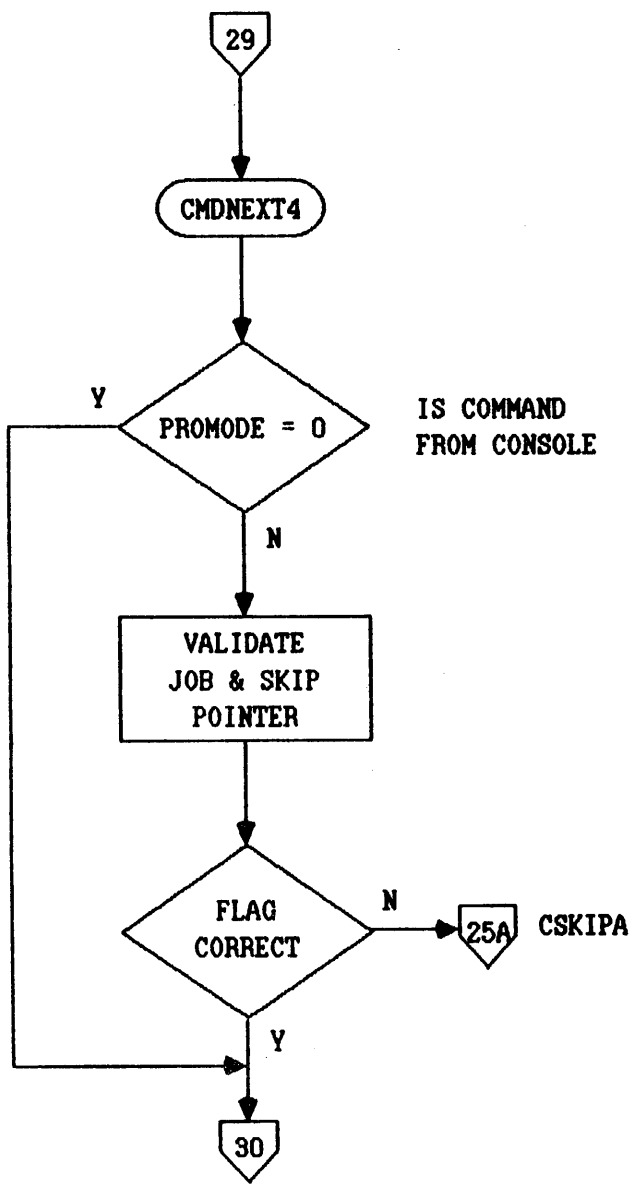


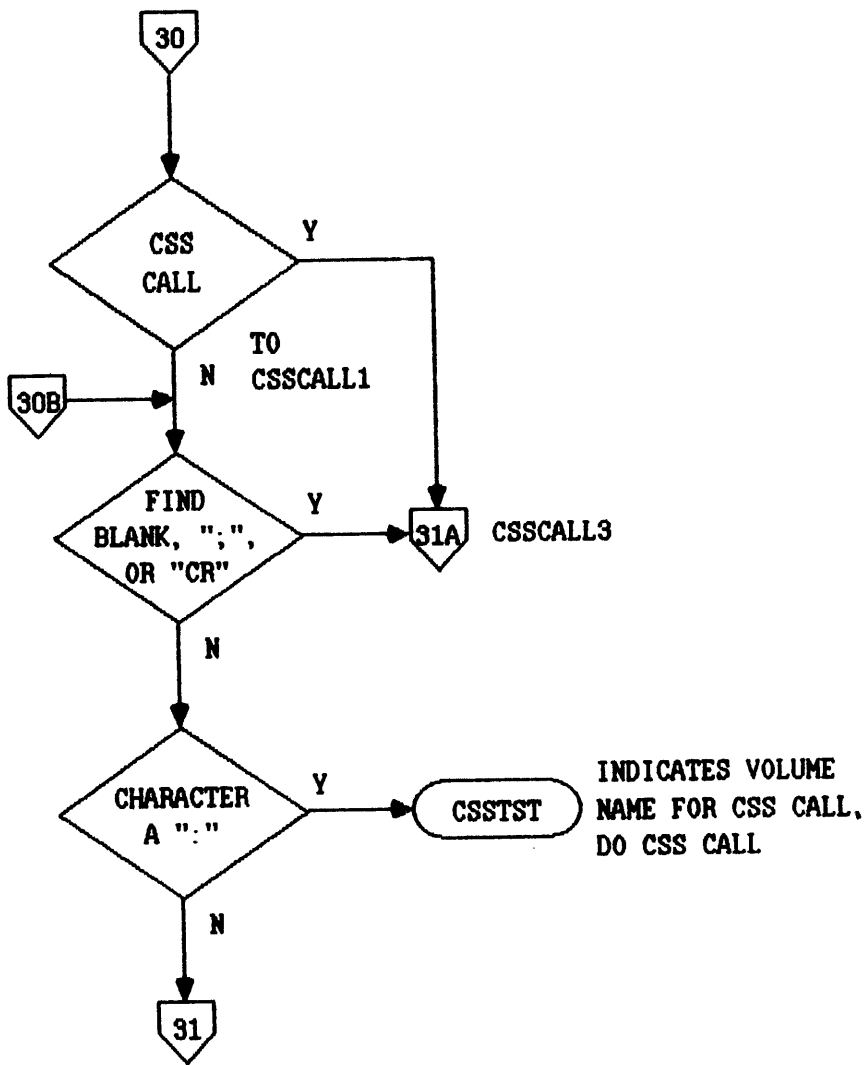


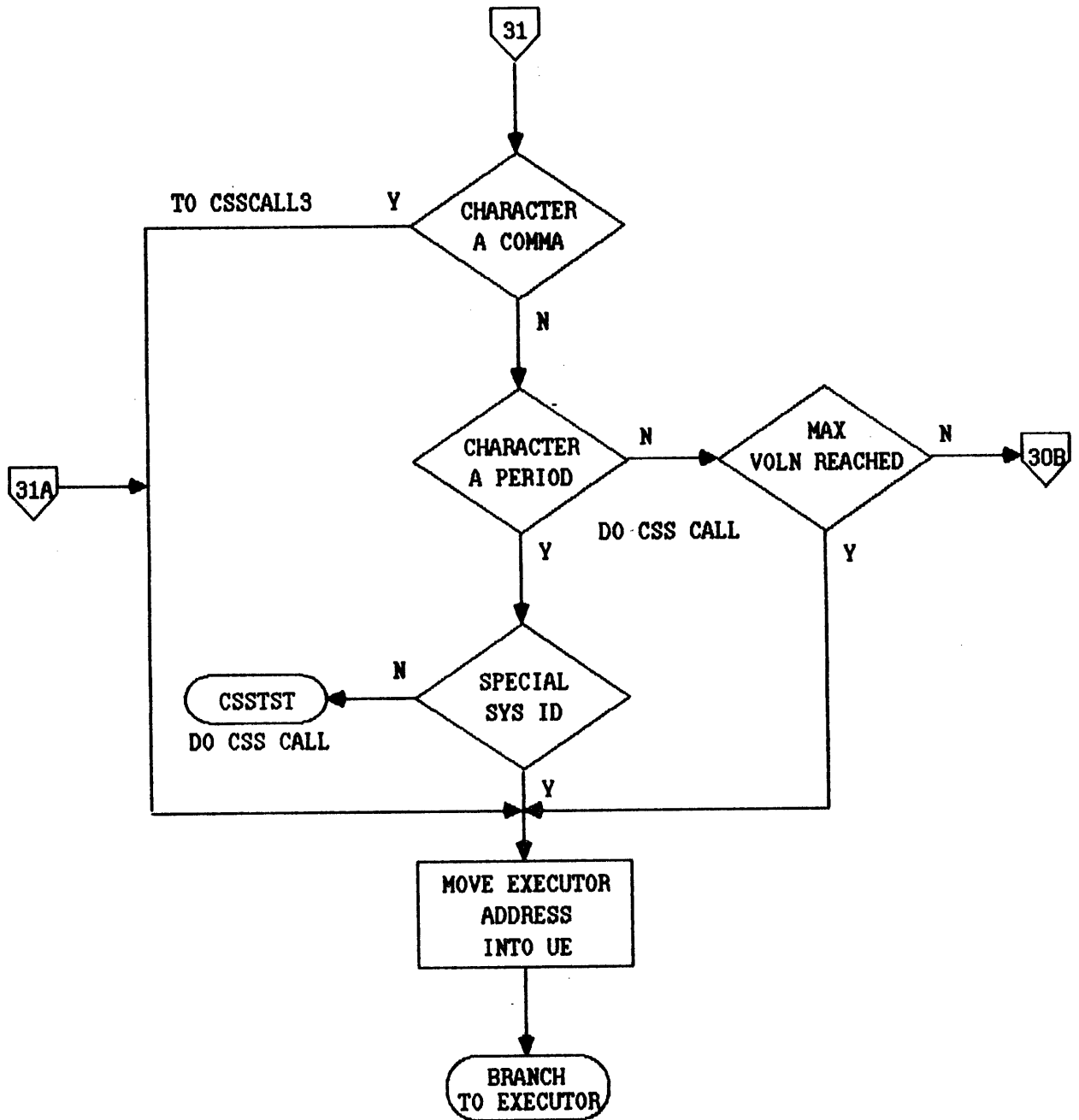


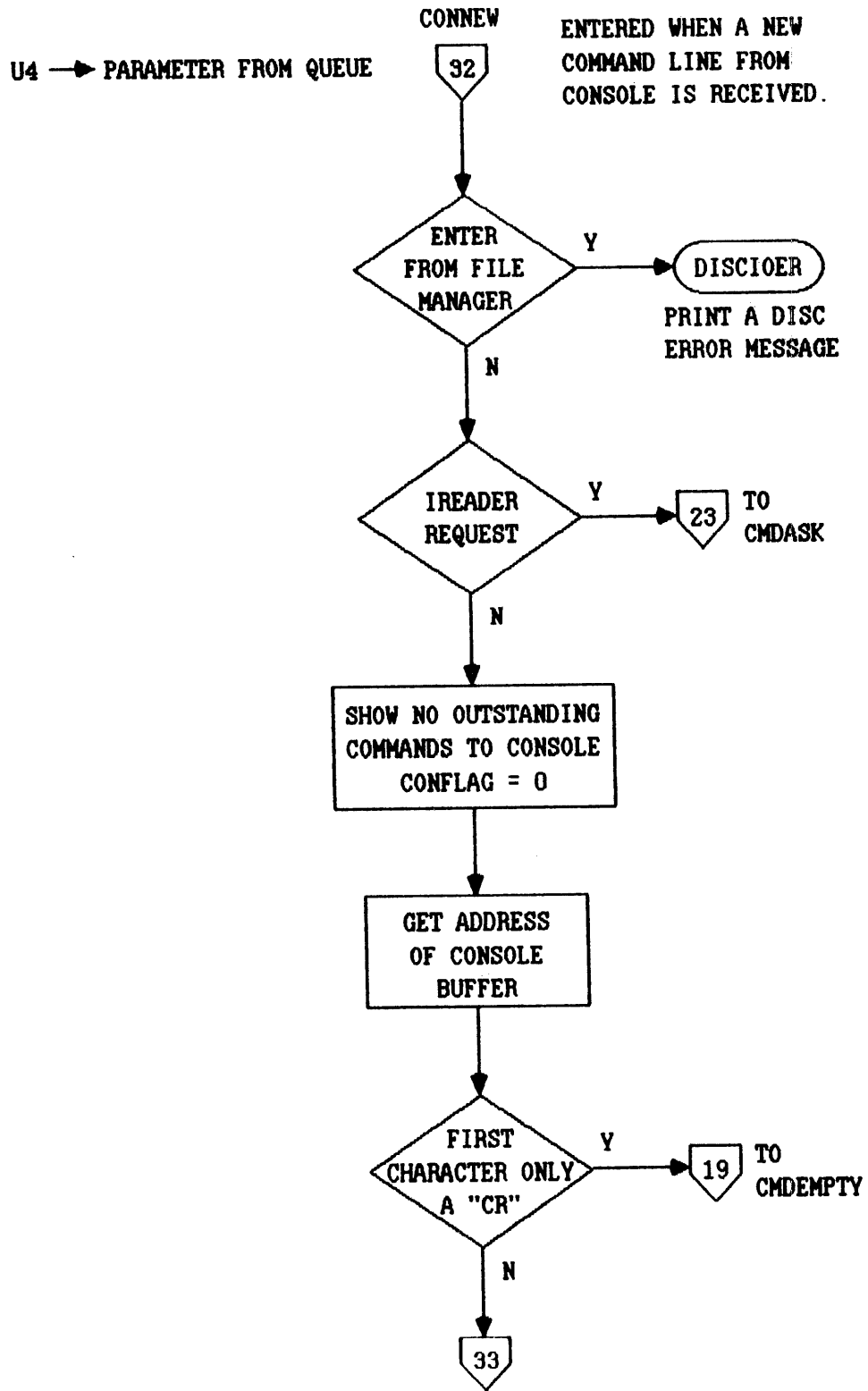
ENTRY HERE AS
LONG AS COMMANDS
SEPARATED BY ";

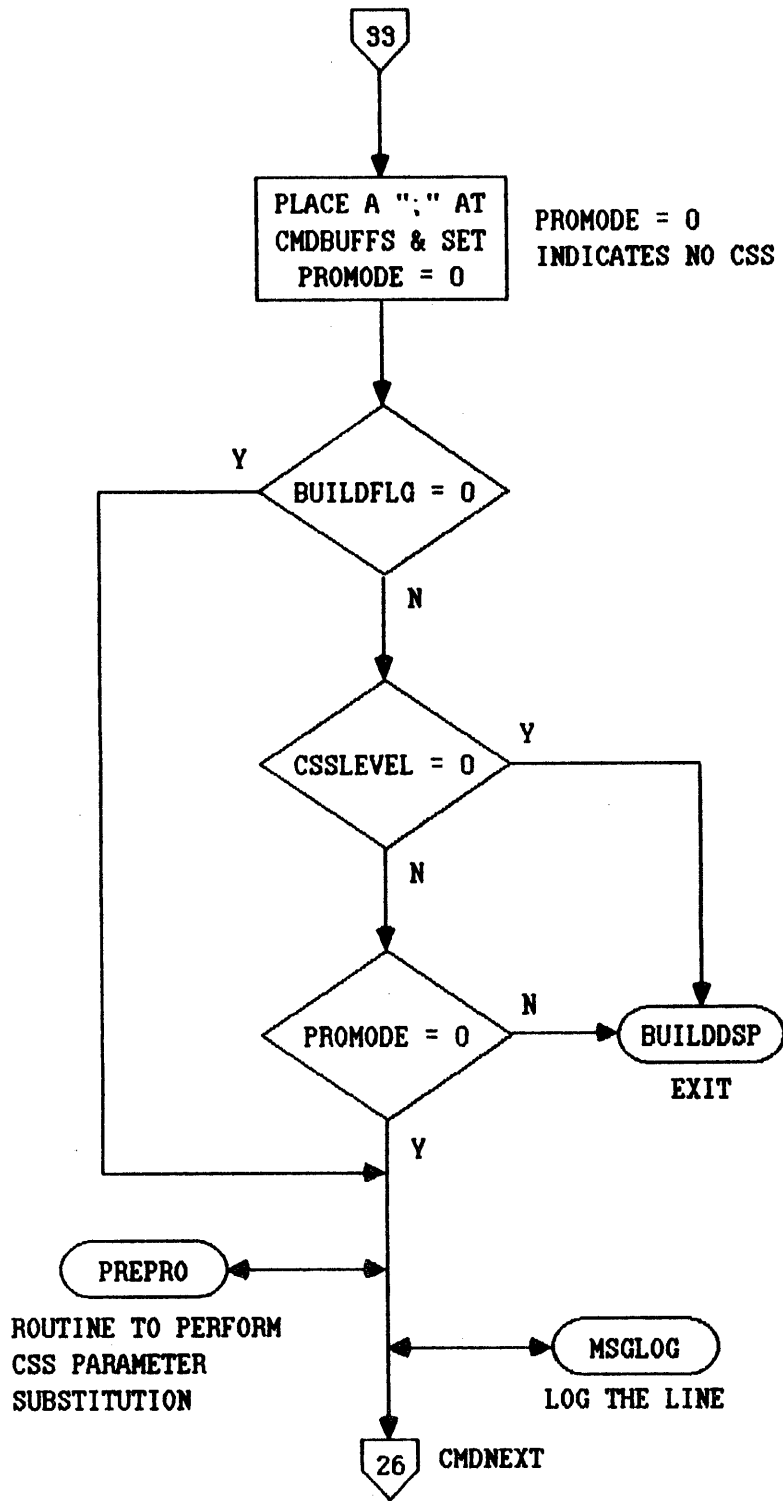


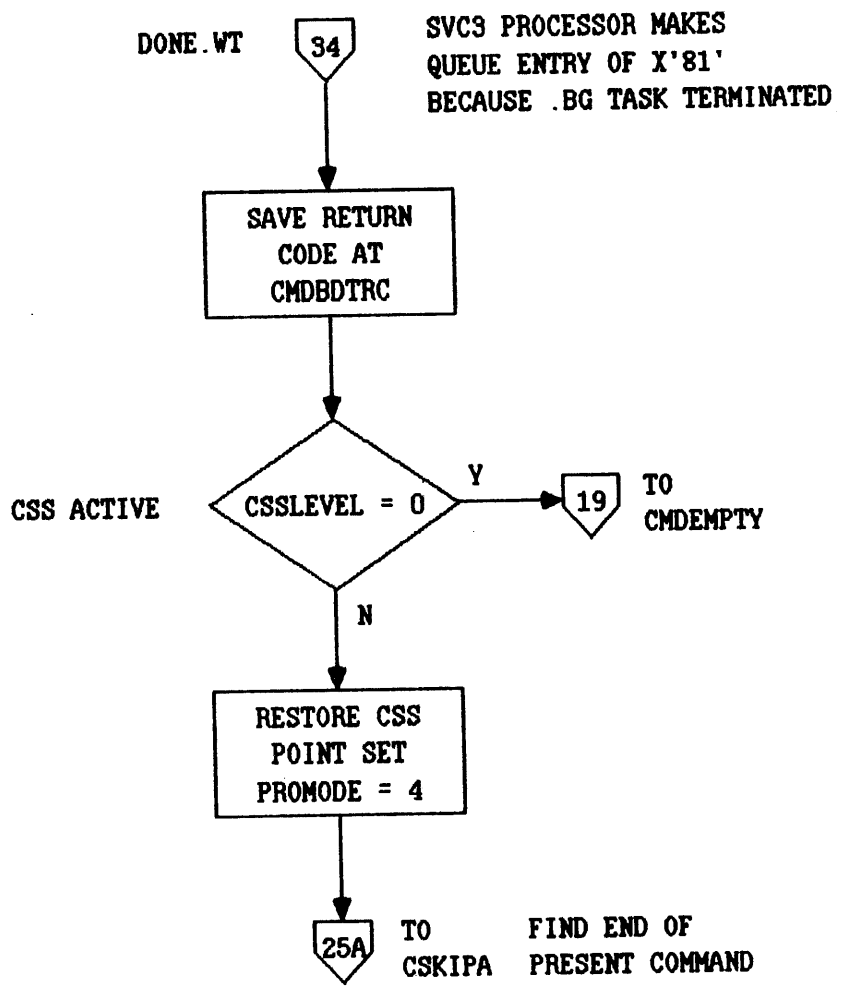






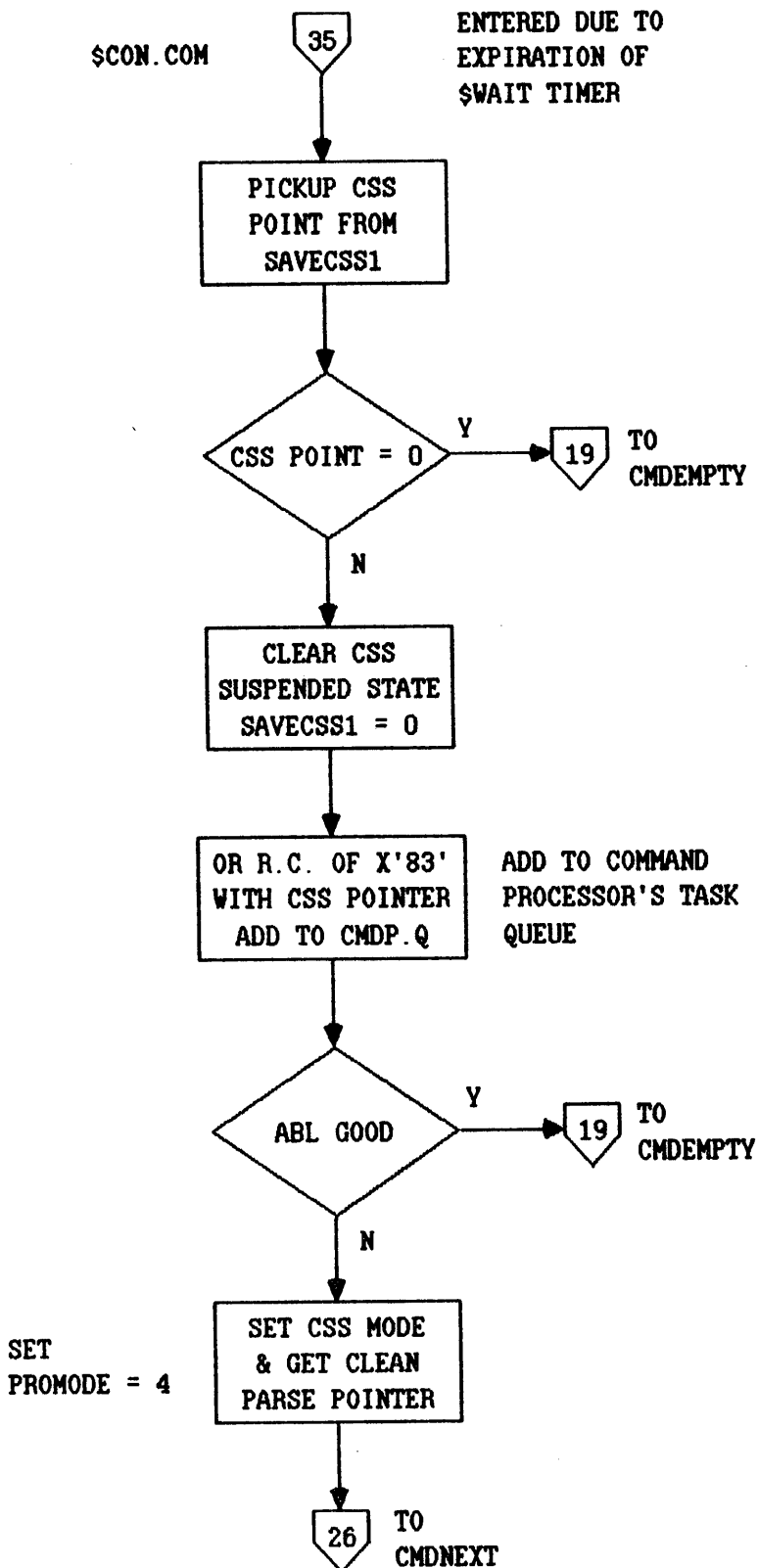






\$CON.COM

ENTERED DUE TO
EXPIRATION OF
\$WAIT TIMER



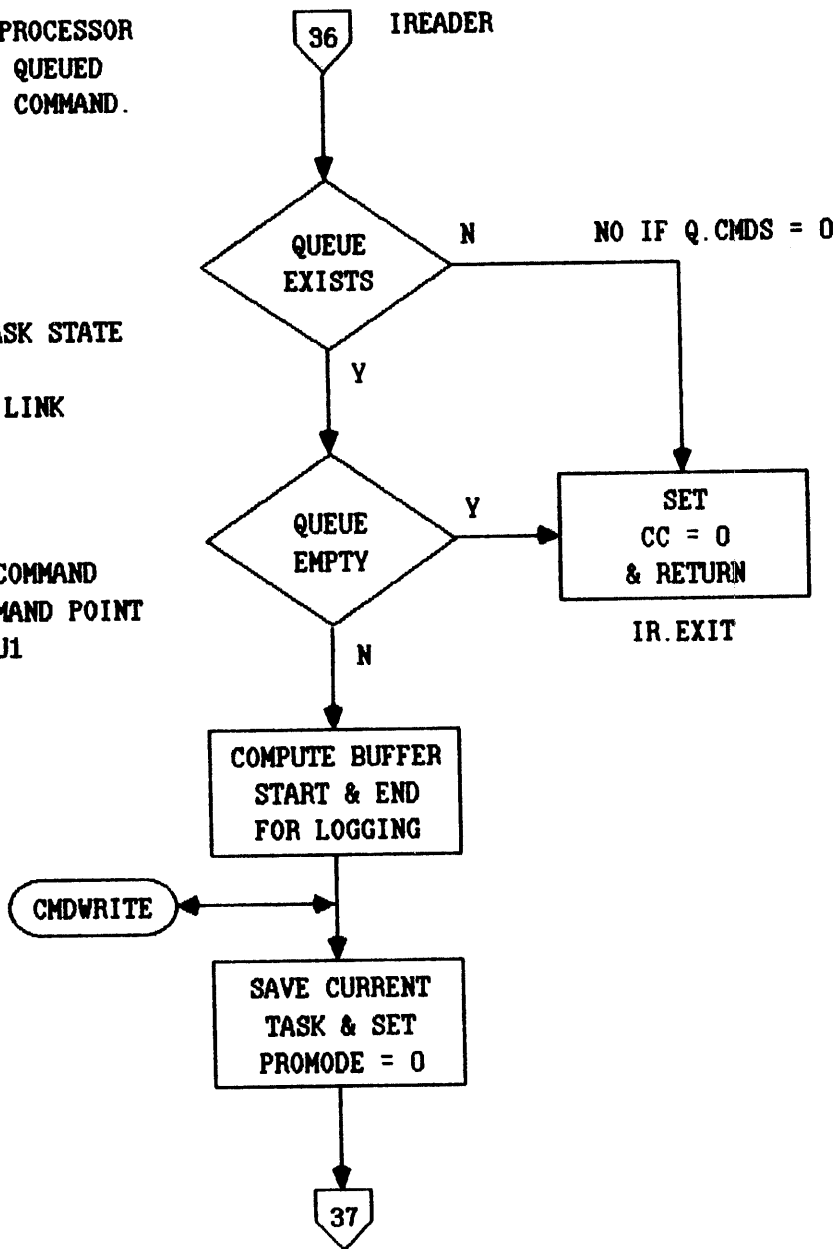
SET UP COMMAND PROCESSOR
TO EXECUTE NEXT QUEUED
INTERNAL READER COMMAND.

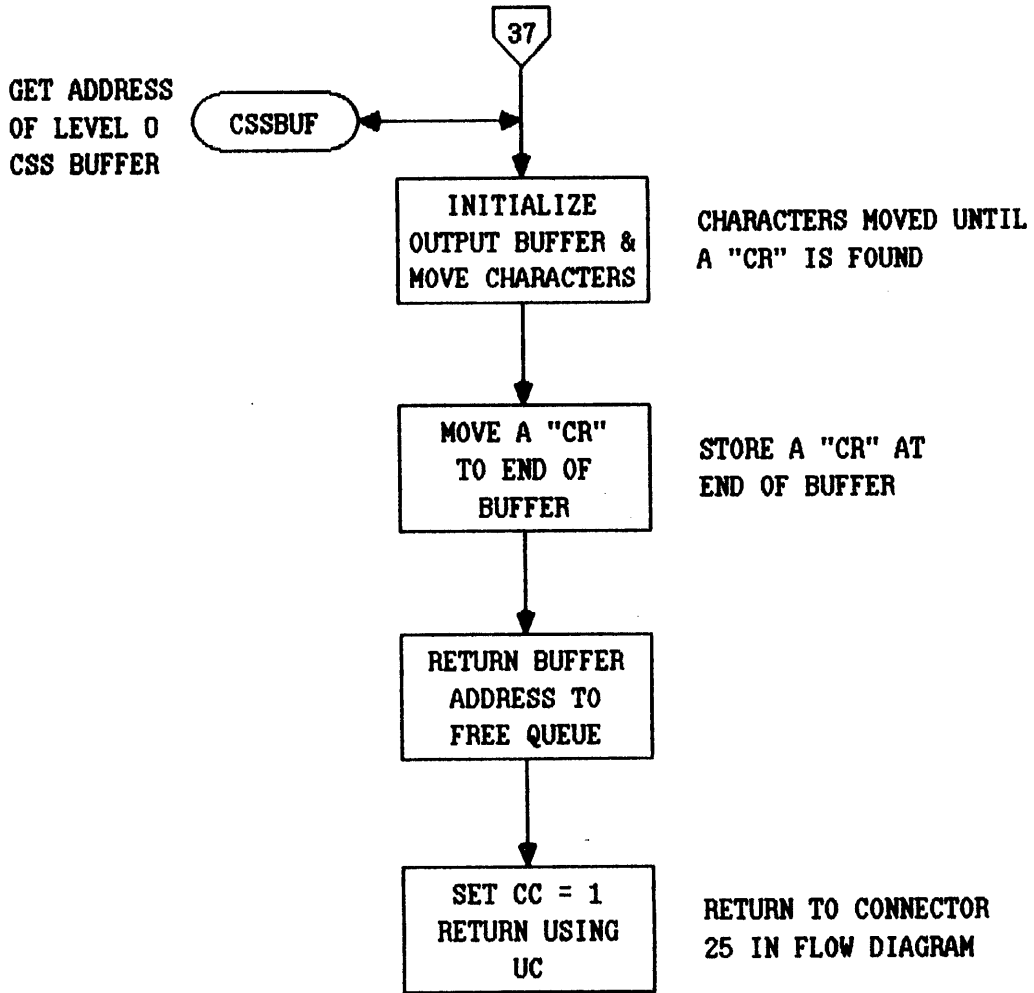
- 1. ENTRY IN ETASK STATE
FROM CMSP
- 2. US - RETURN LINK

EXIT:

- 1. CC = 0 NO COMMAND
- 2. CC = 1 COMMAND POINT
IN U1

LOG IREAD
COMMAND TO
CONSOLE DEVICE

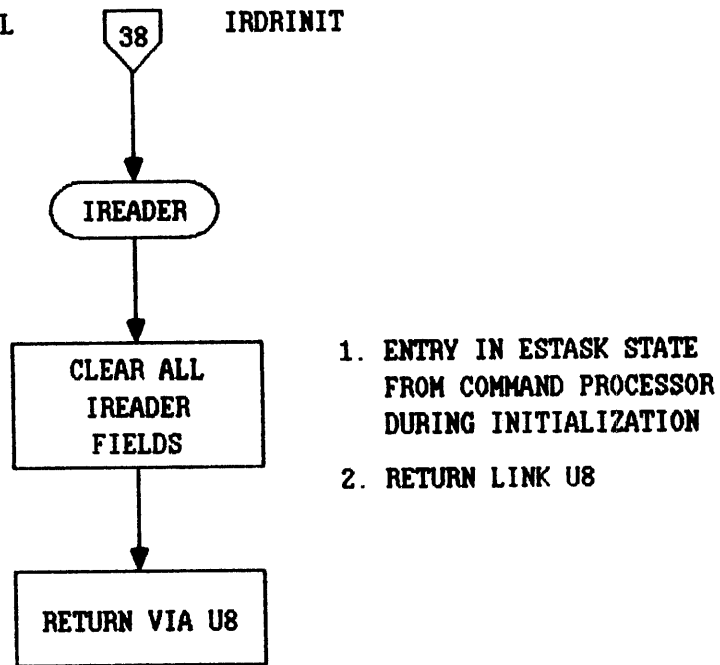




INITIALIZE ALL INTERNAL
READER VARIABLES

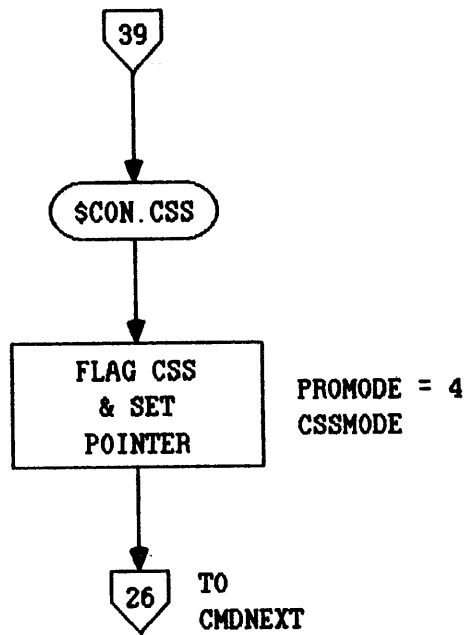
38

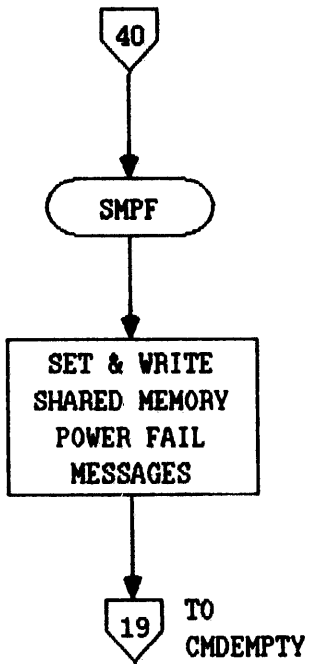
IRDRINIT



THE FOLLOWING ARE SET TO 0:

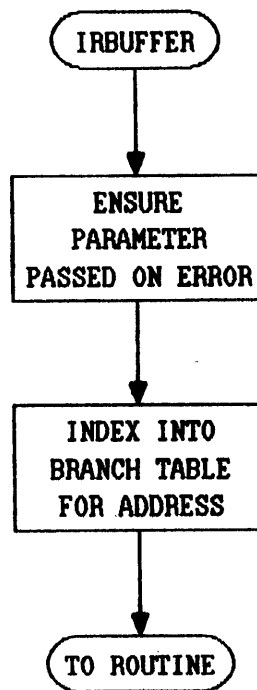
IR.POOL → BUFFER POOL ADDRESS
IR.SIZE → SIZE OF POOL
IR.LEN → BUFFER LENGTH
Q.FREE → FREE POOL ADDRESS
Q.CMDS → QUEUED COMMAND POOL ADDRESS





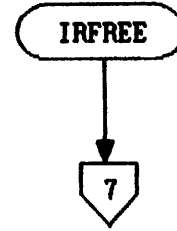
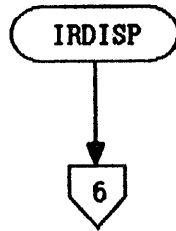
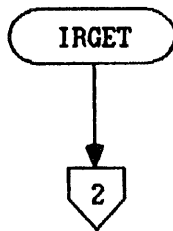
INTERNAL READER

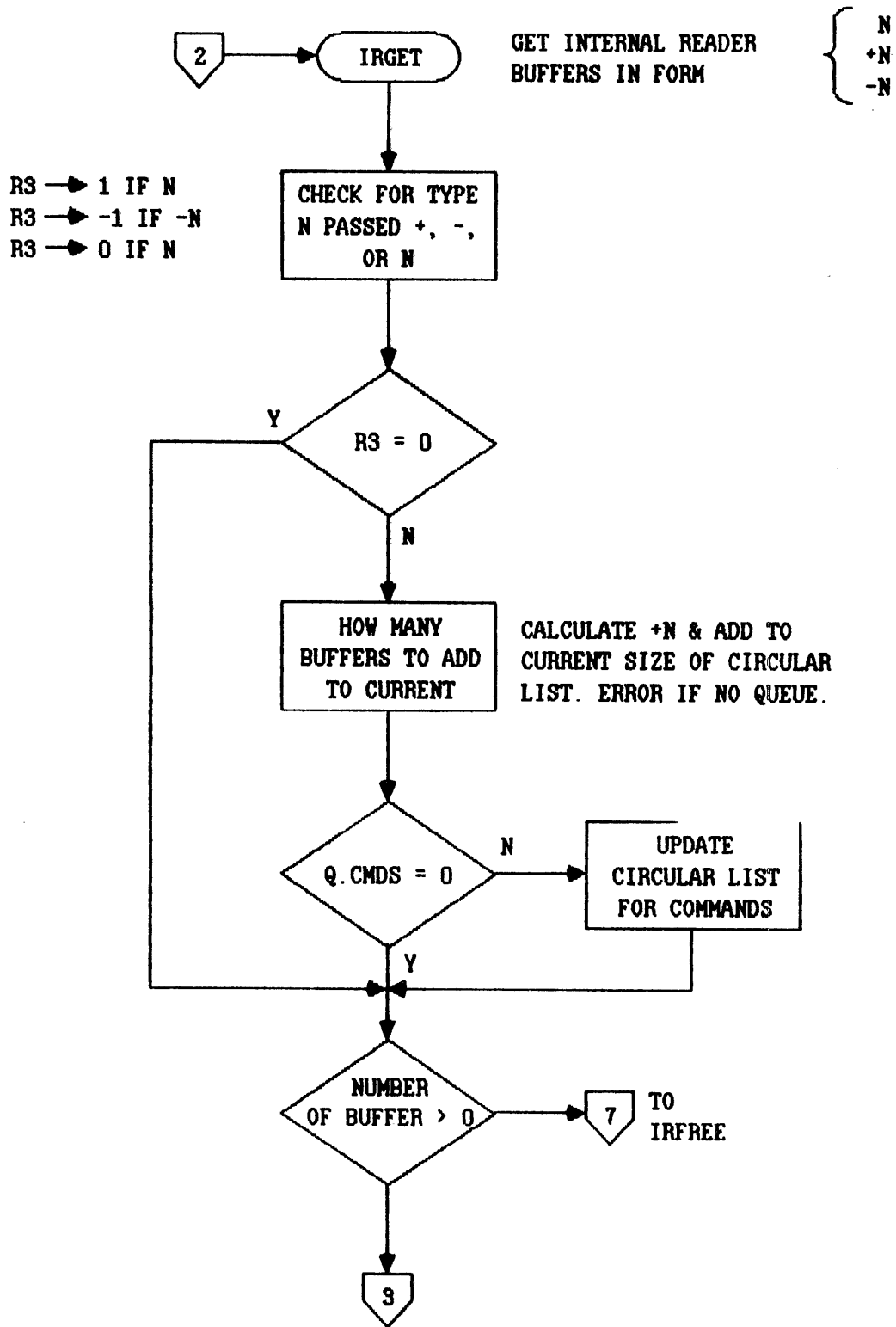
ENTERED FROM BRANCH TABLE
IN COMMAND PROCESSOR

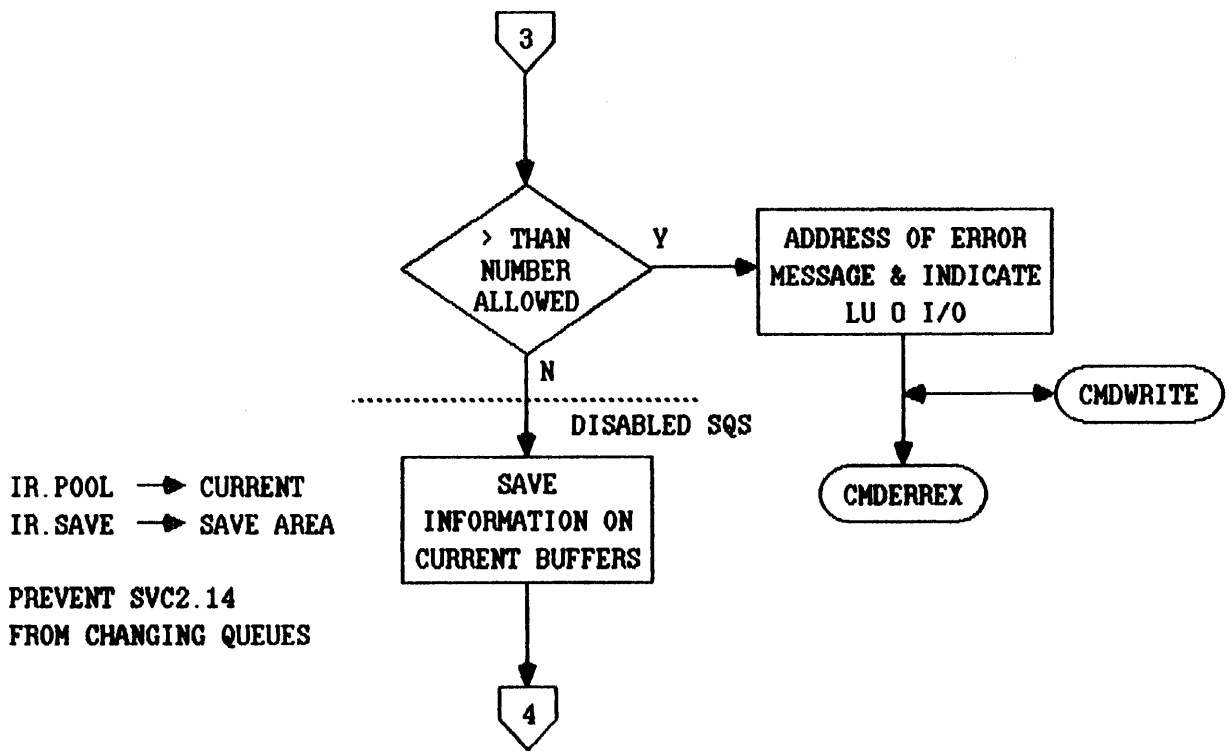


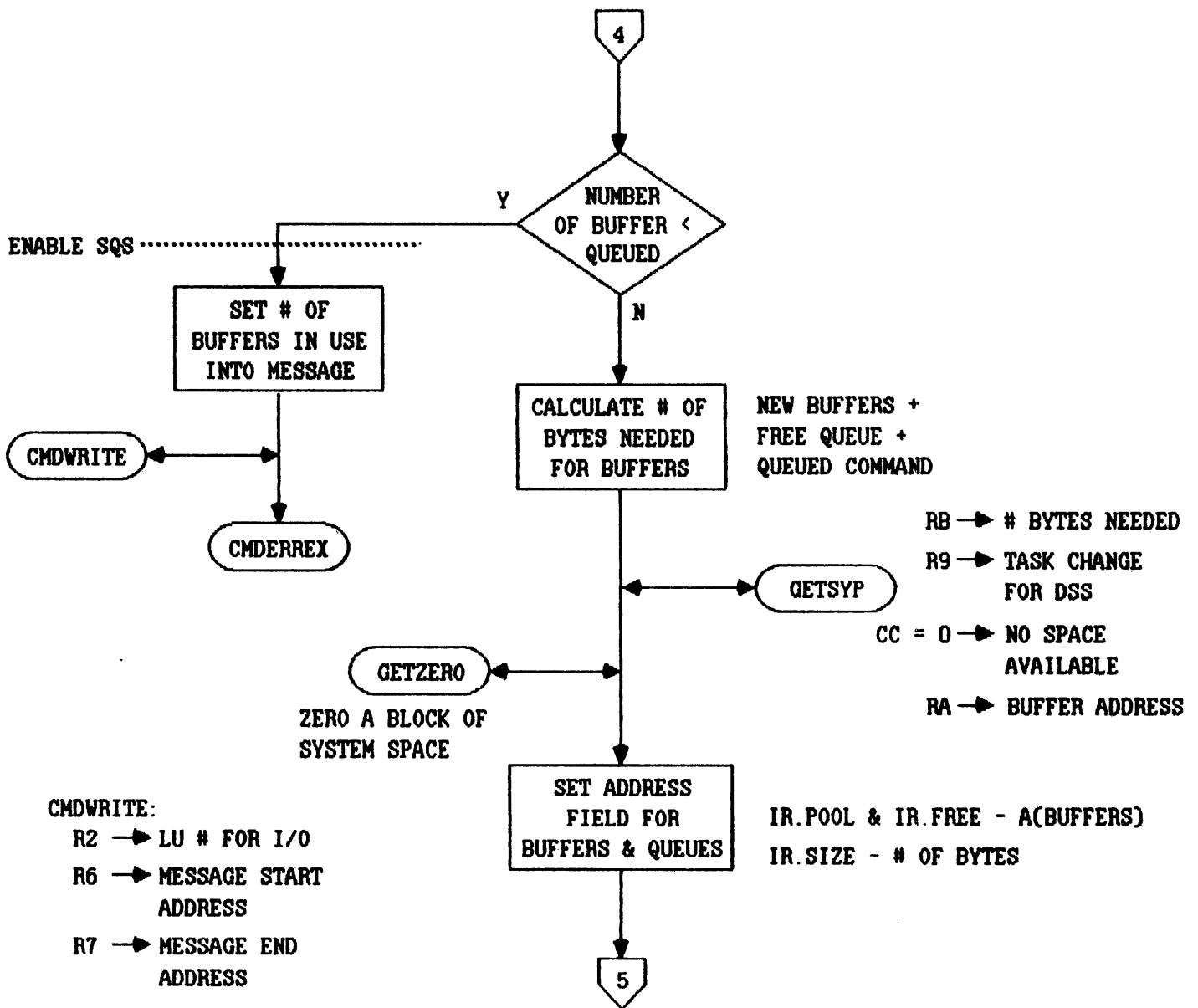
ERROR ROUTINE IS IR.PERR

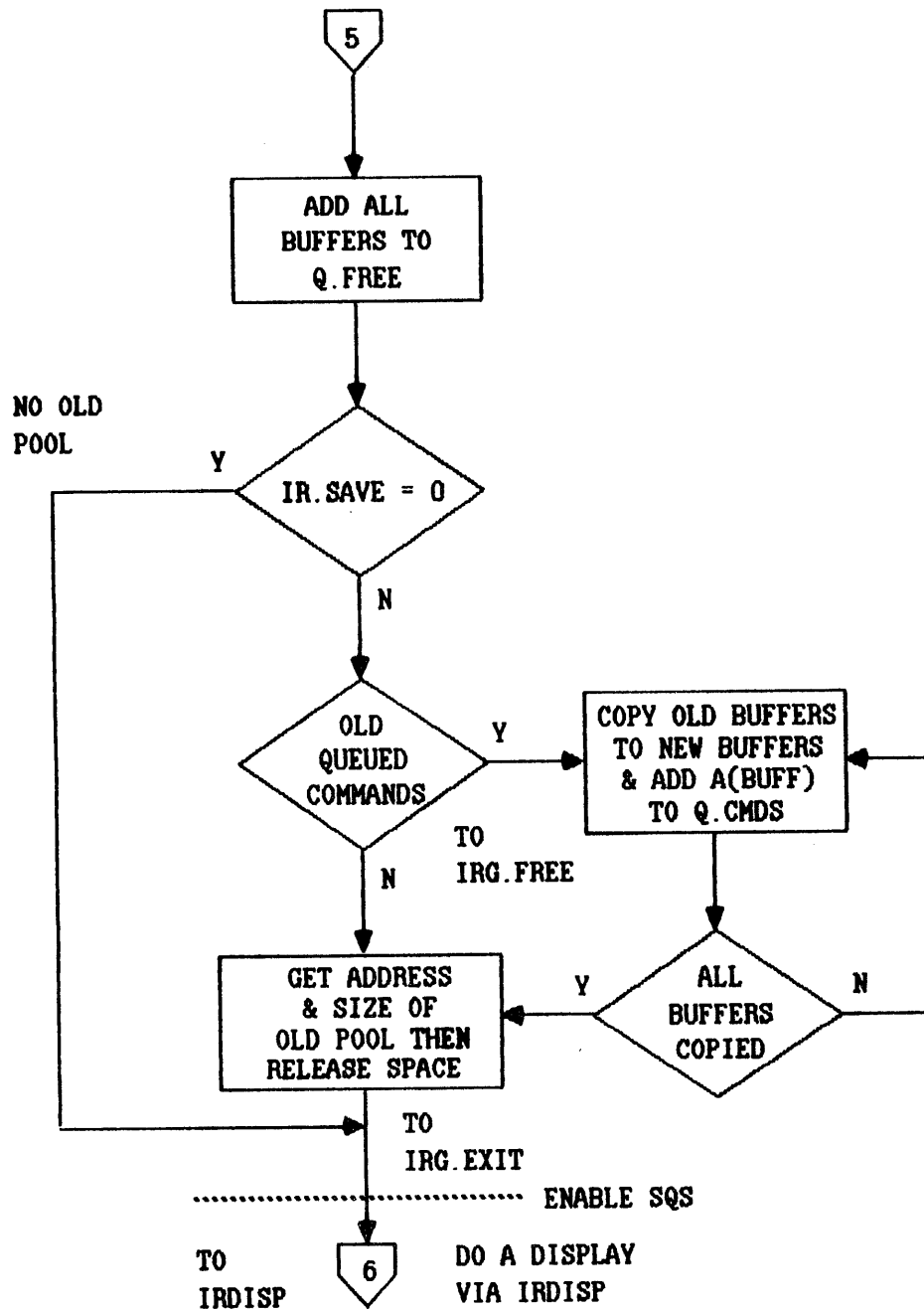
IRBUF.VT {
A(IRGET)
A(IRDISP)
A(IRFREE)



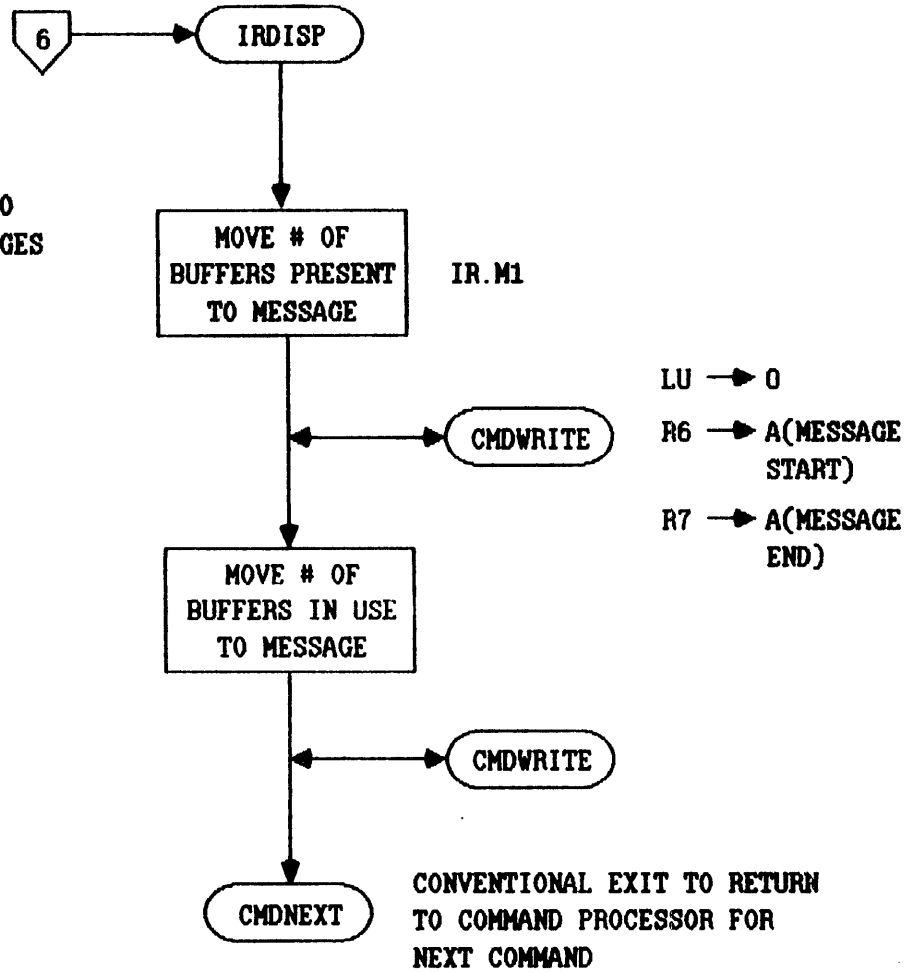


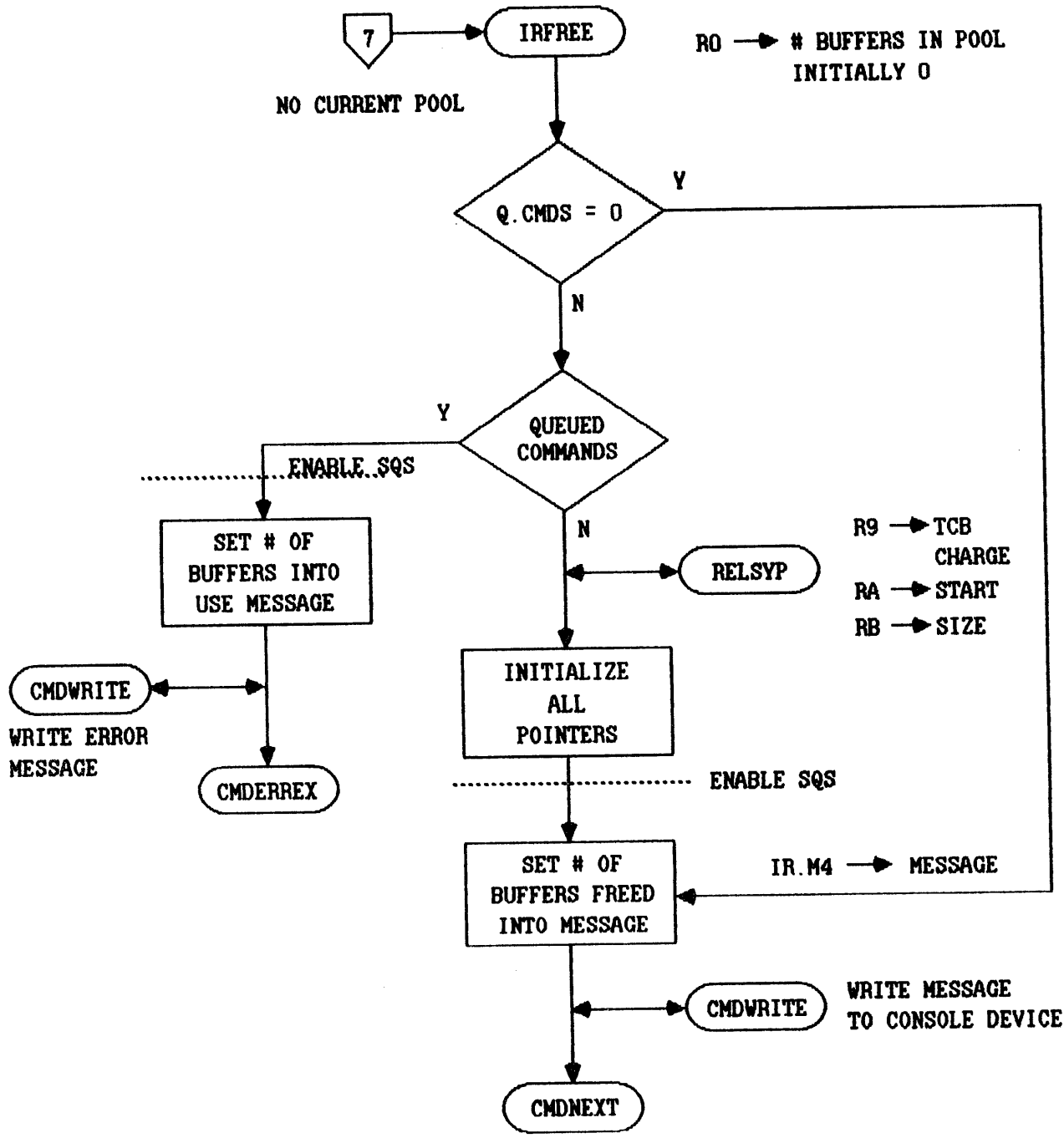






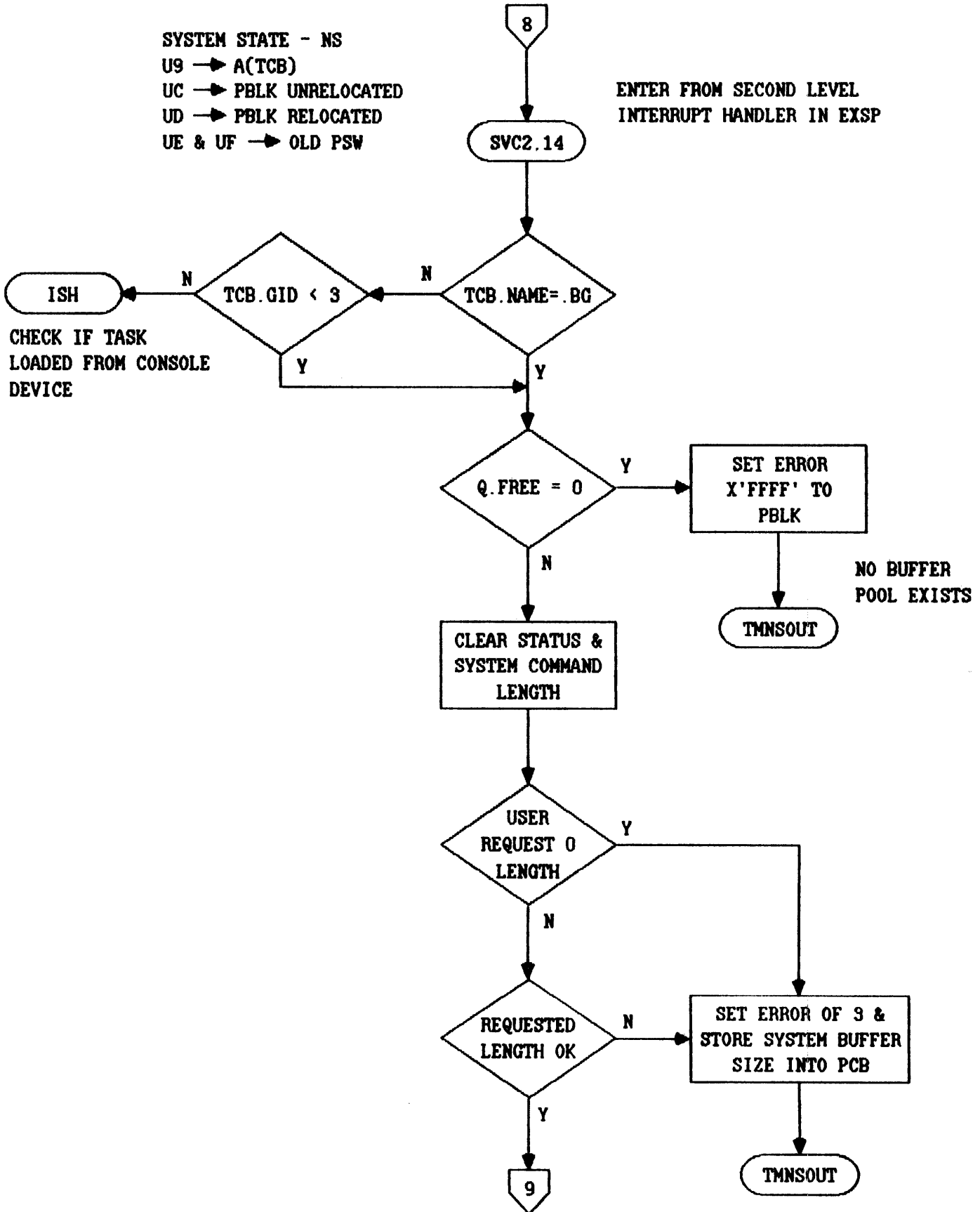
Q. CMDS. IS QUEUED
COMMAND LIST USED TO
GENERATE BOTH MESSAGES

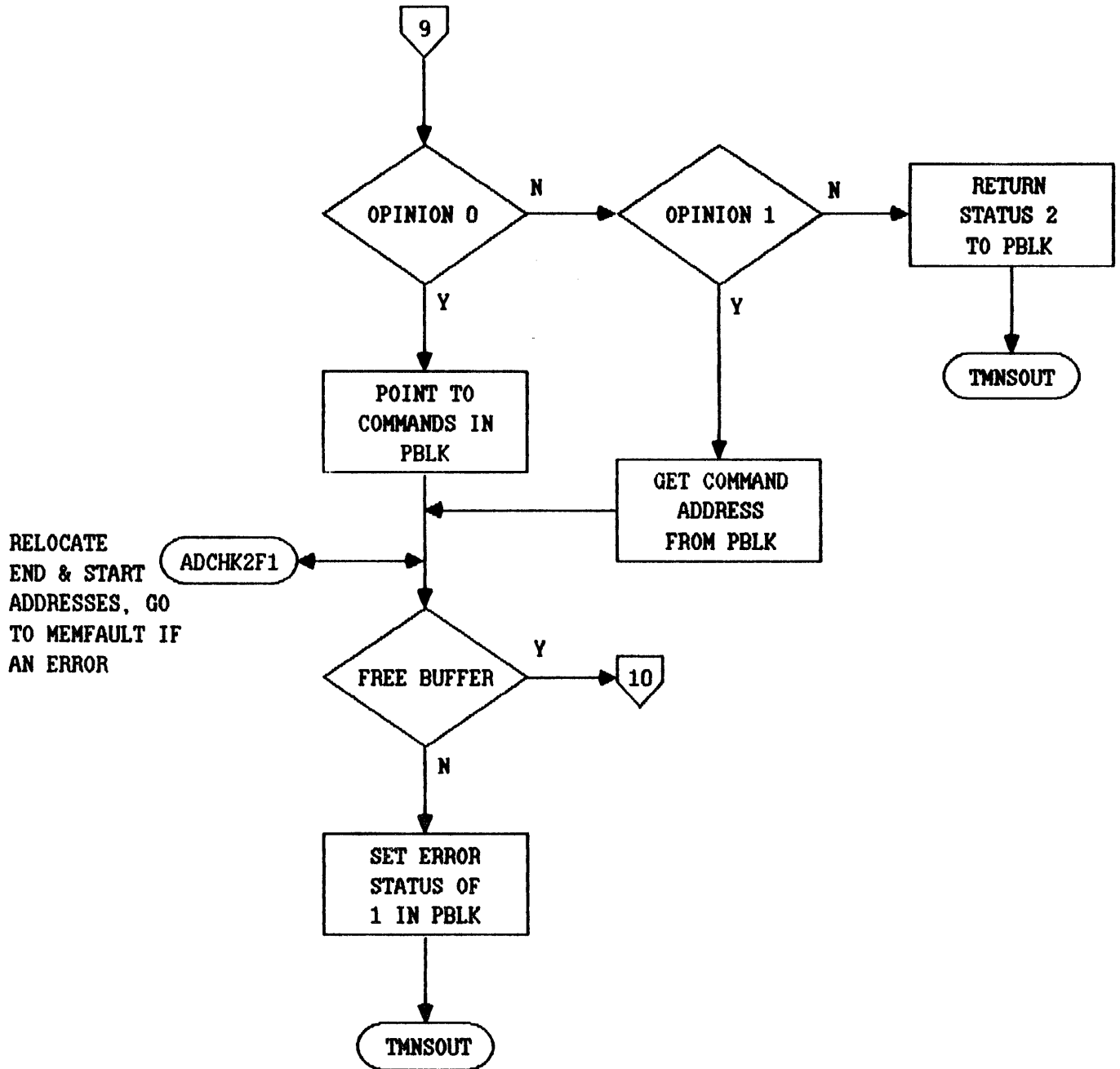


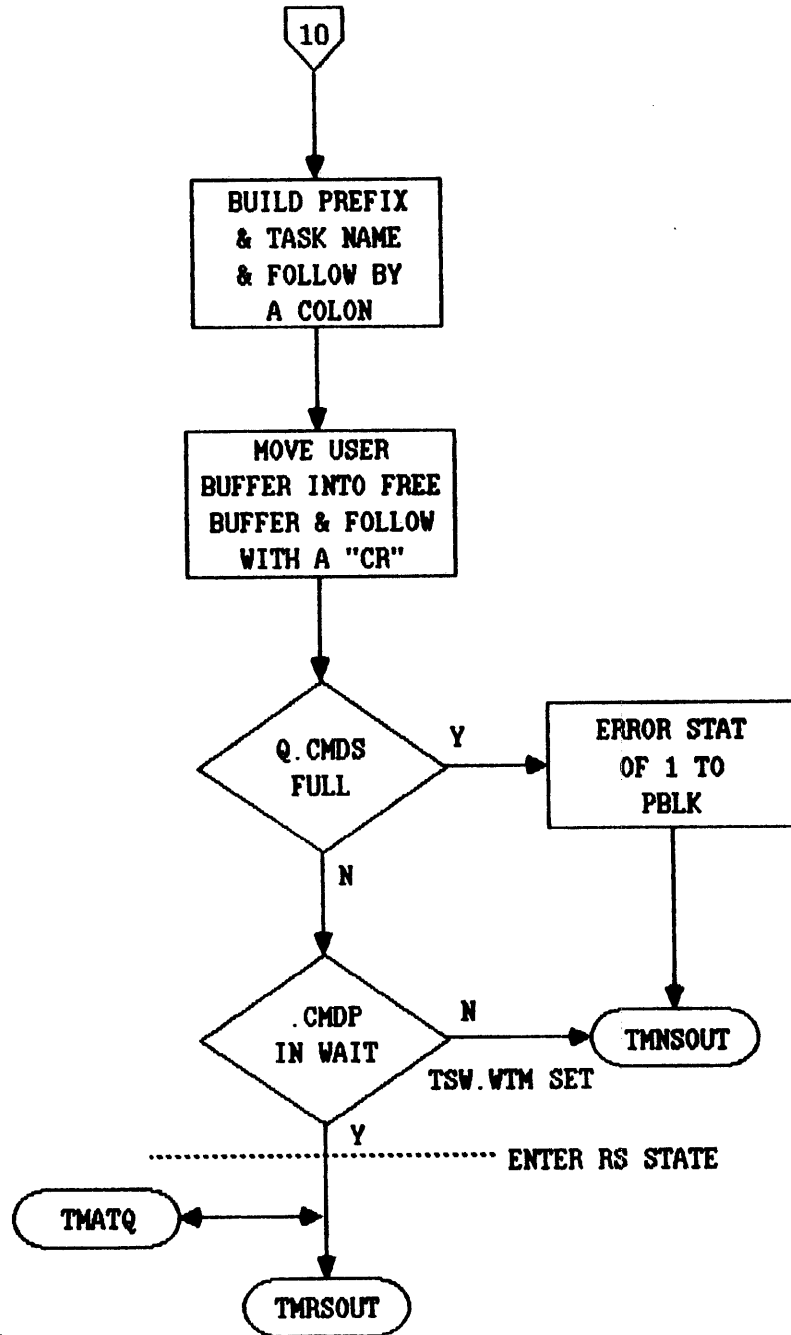


SYSTEM STATE - NS
 U9 → A(TCB)
 UC → PBLK UNRELOCATED
 UD → PBLK RELOCATED
 UE & UF → OLD PSW

ENTER FROM SECOND LEVEL
 INTERRUPT HANDLER IN EXSP



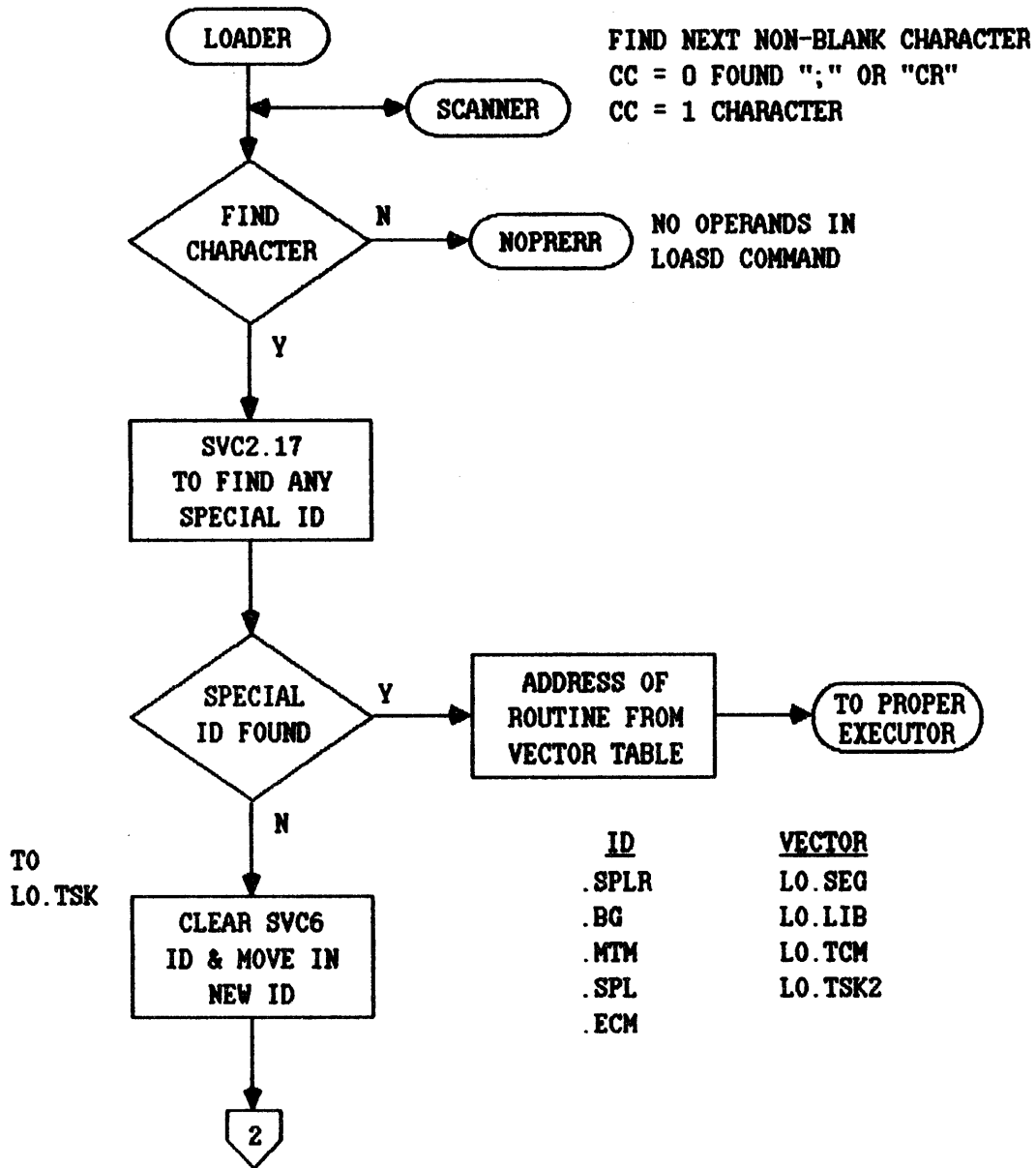


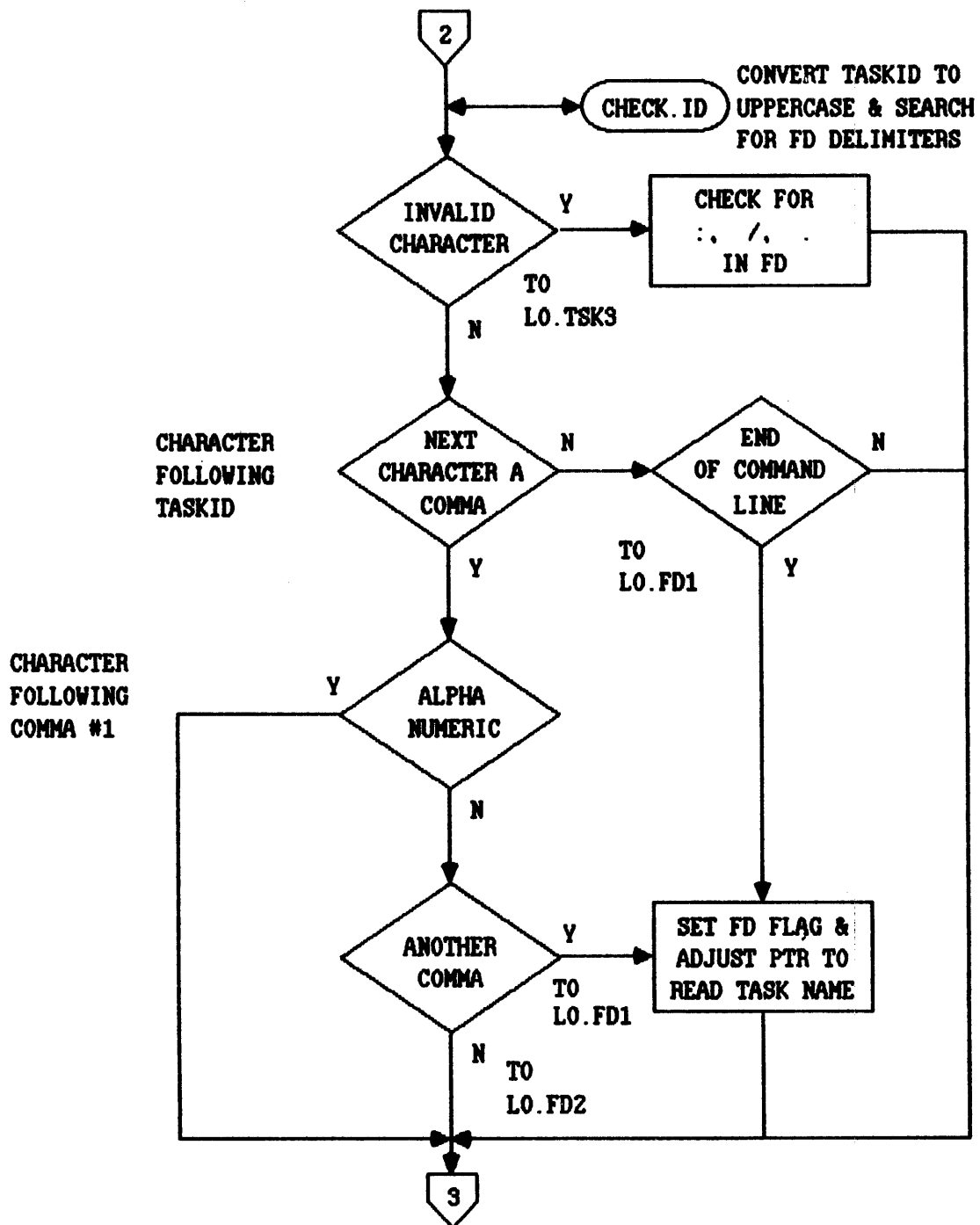


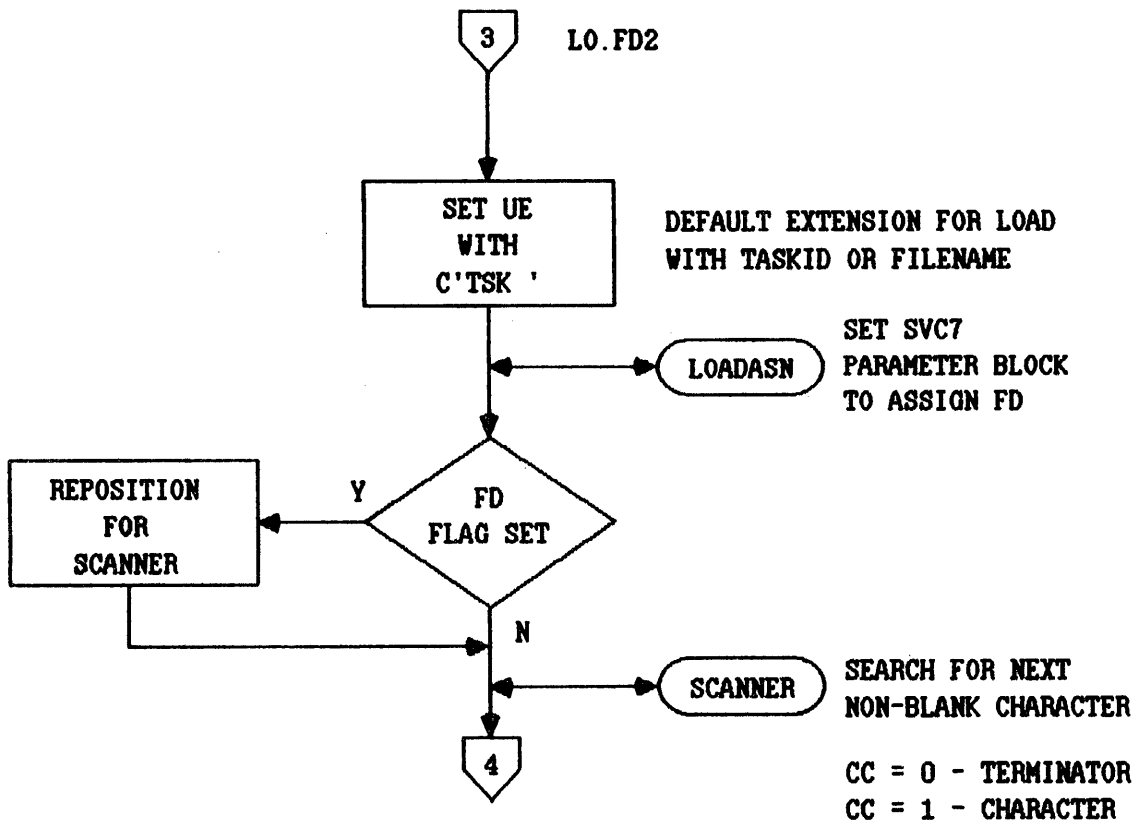
Y'01000000'

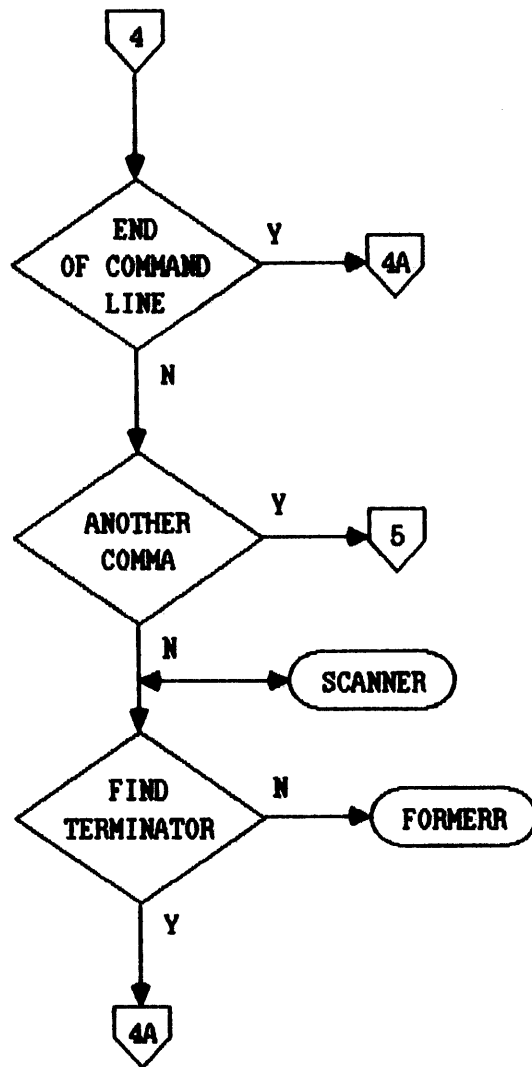
QUEUE A PARAMETER
TO COMMAND PROCESSOR
TASK QUEUE TO INDICATE
A COMMAND EXISTS

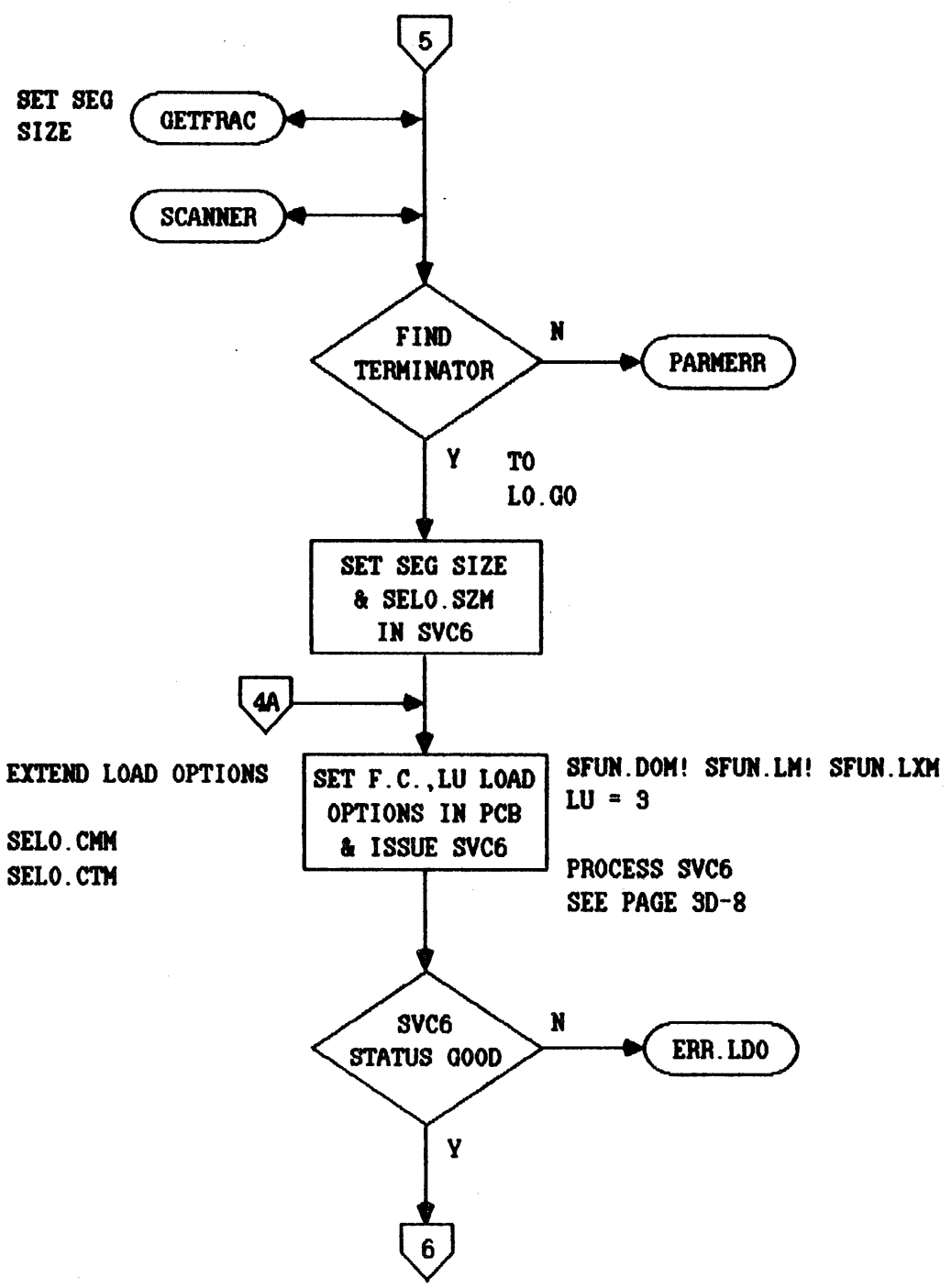
RESIDENT LOADER

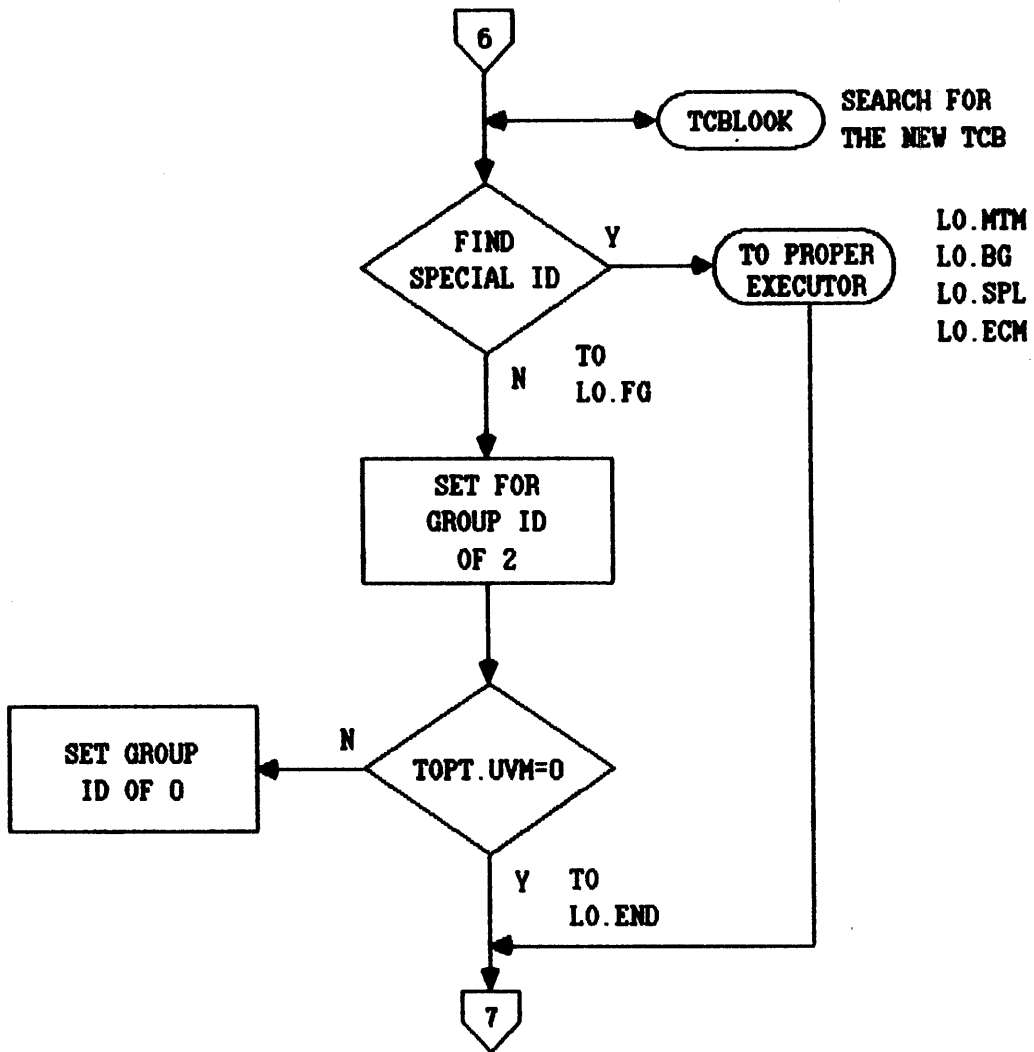


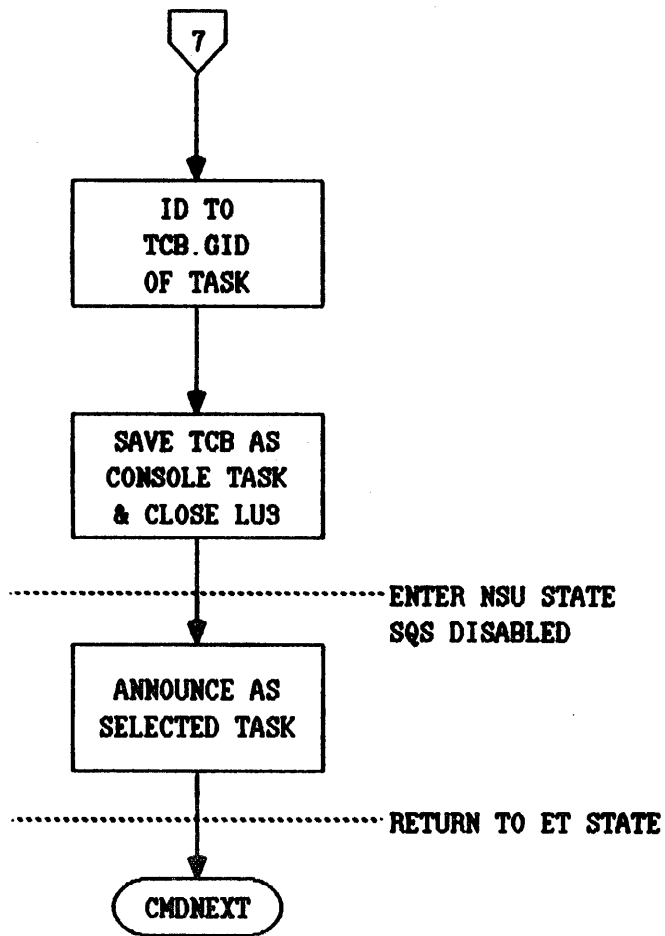










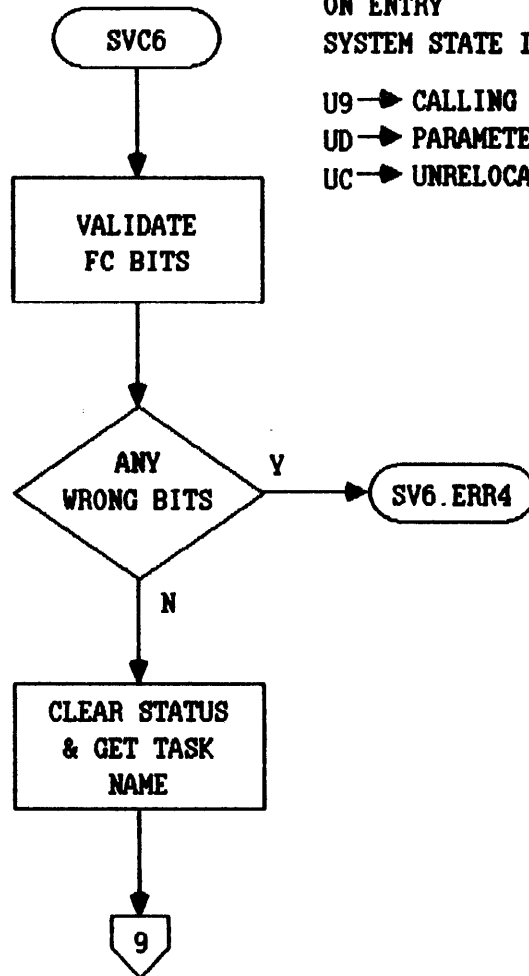


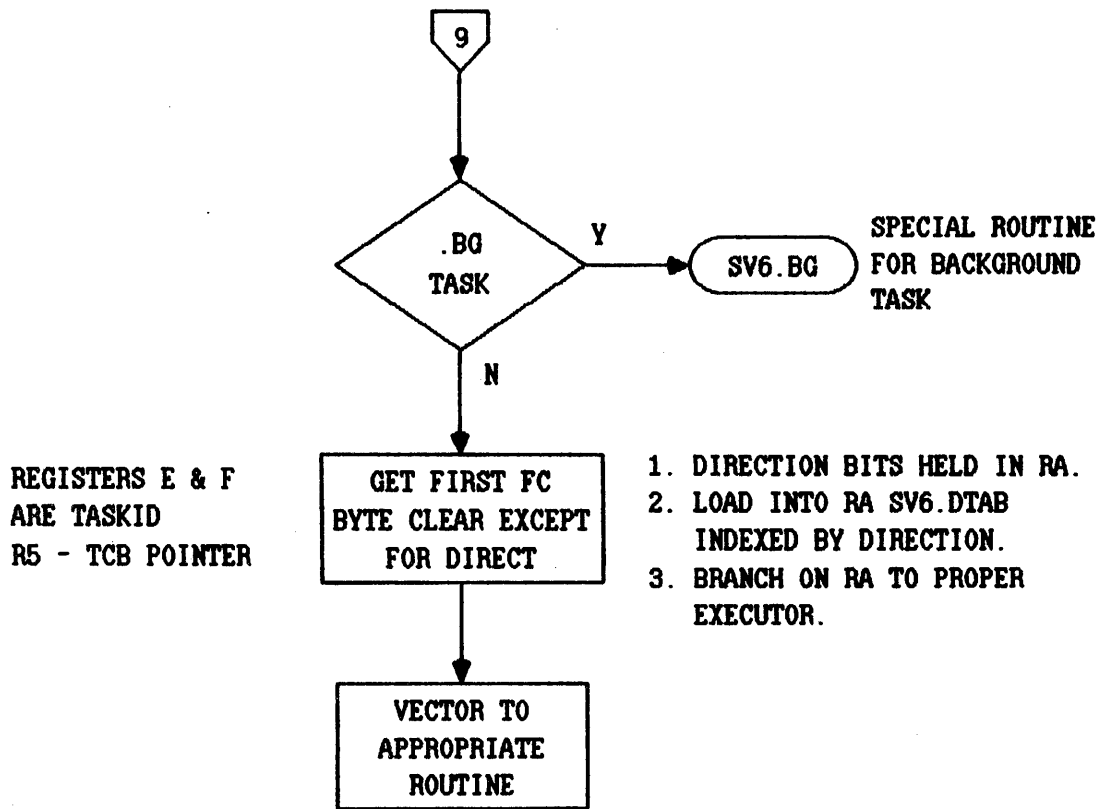
LOAD REQUESTS
SVC6

ON ENTRY
SYSTEM STATE IS RS

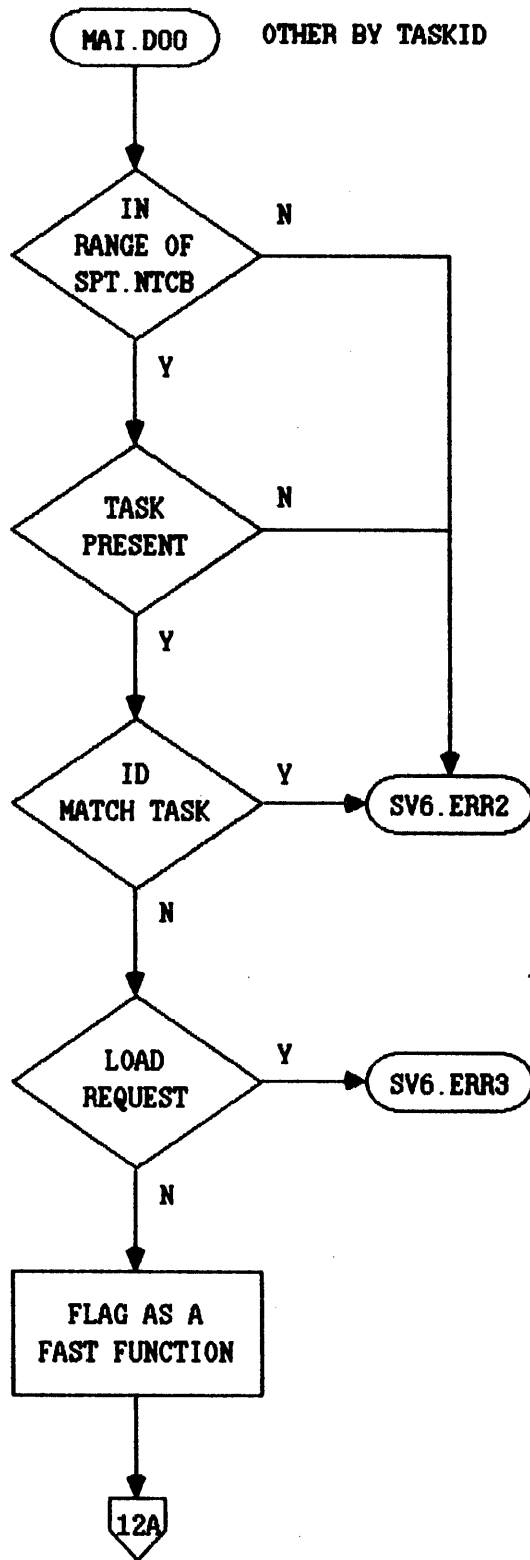
U9 → CALLING A(TCB)
UD → PARAMETER BLOCK ADDRESS
UC → UNRELOCATED BLOCK

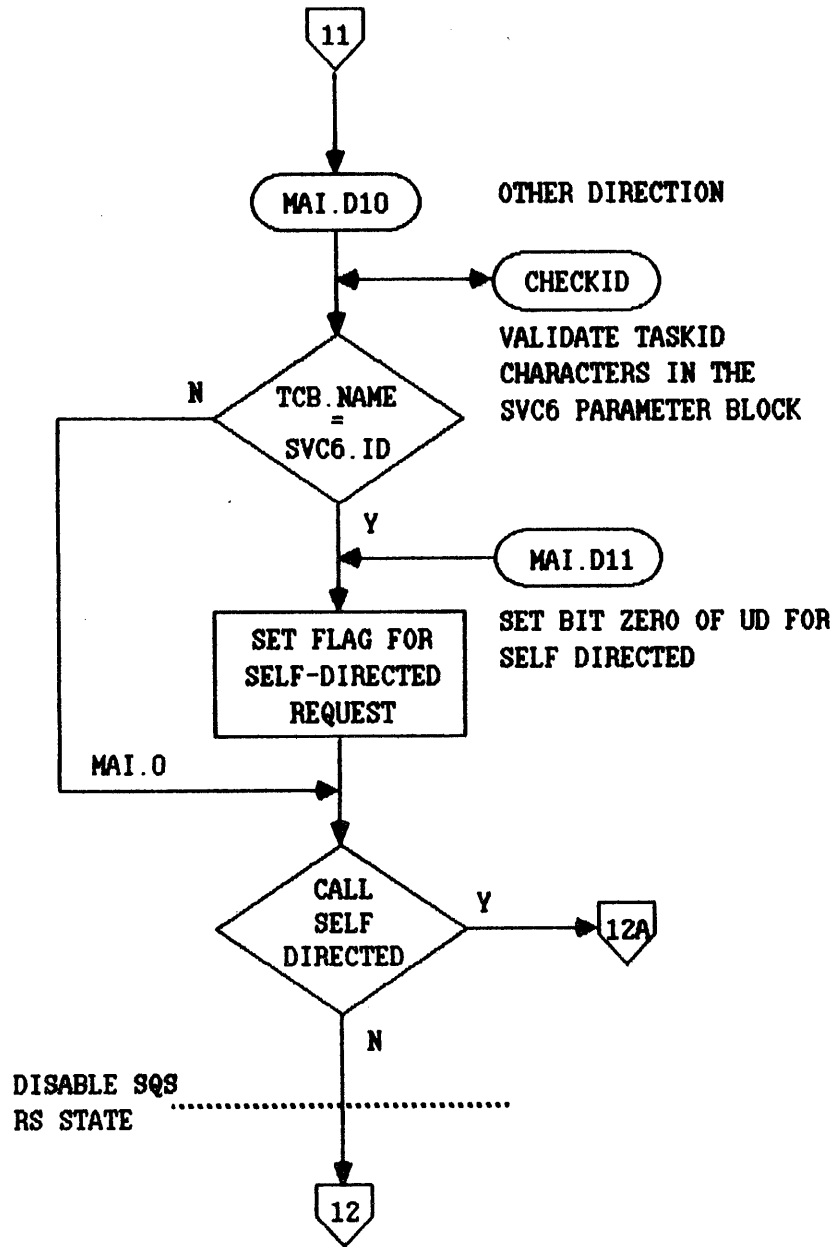
FC HAS
ILLEGAL BITS?

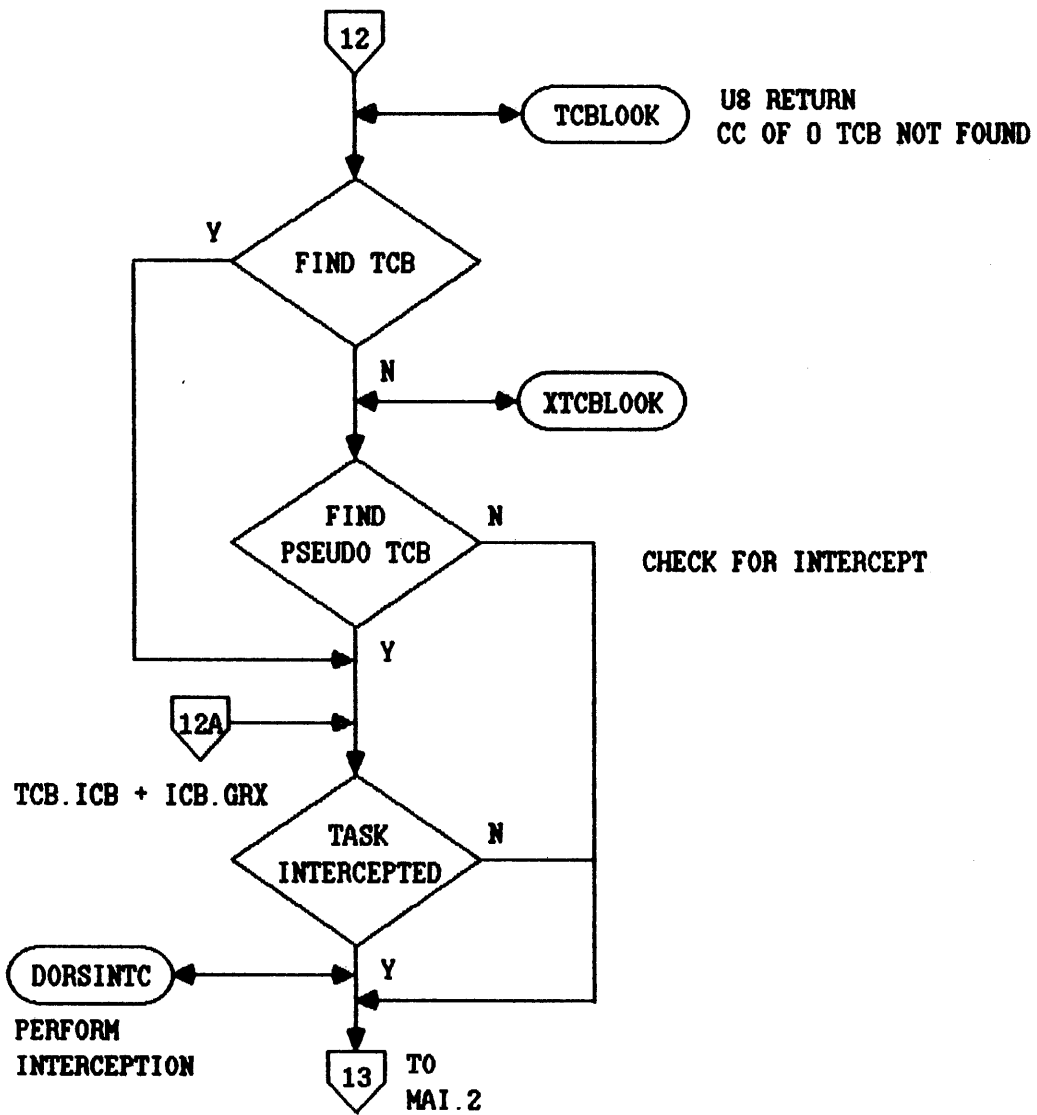


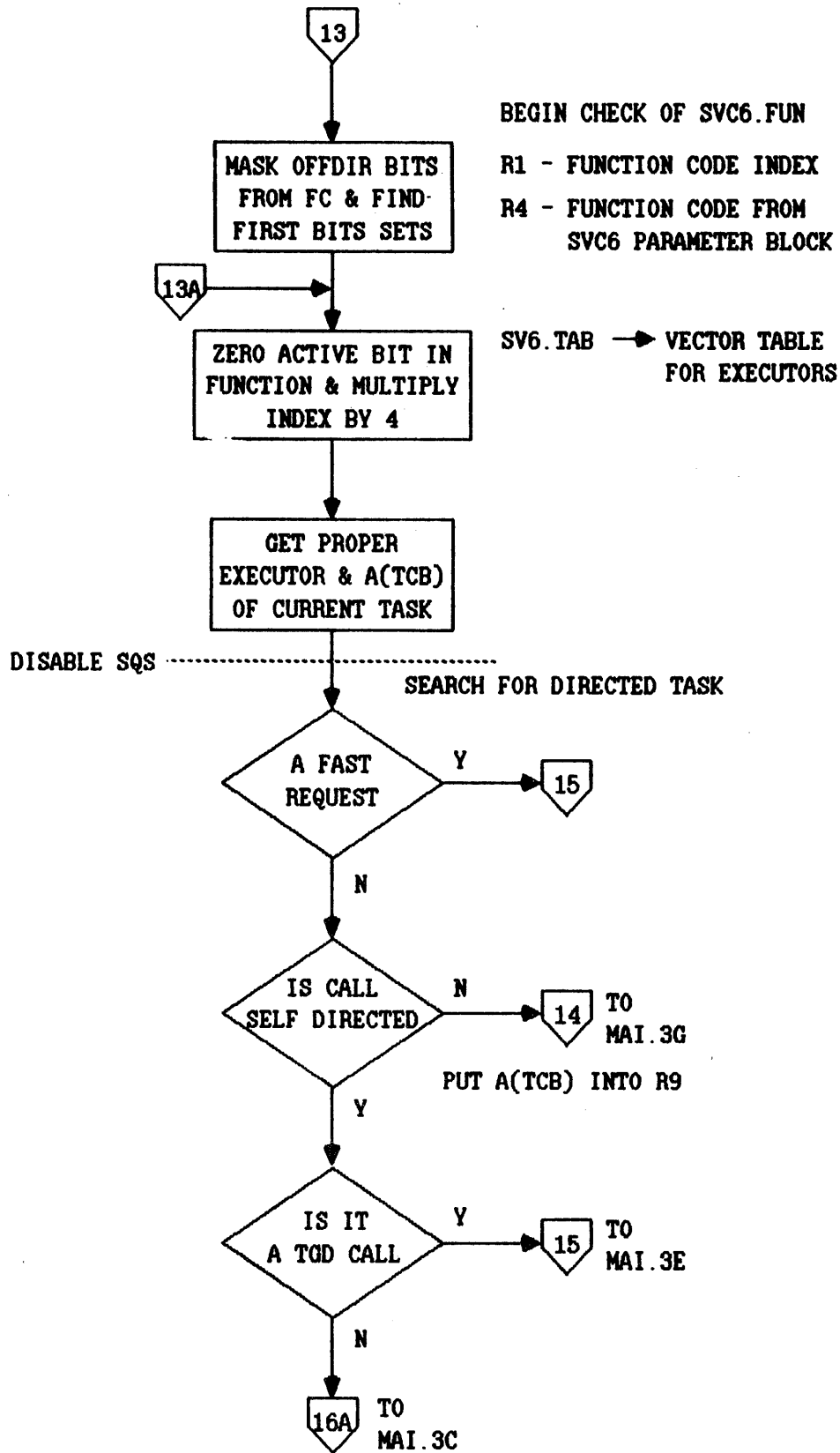


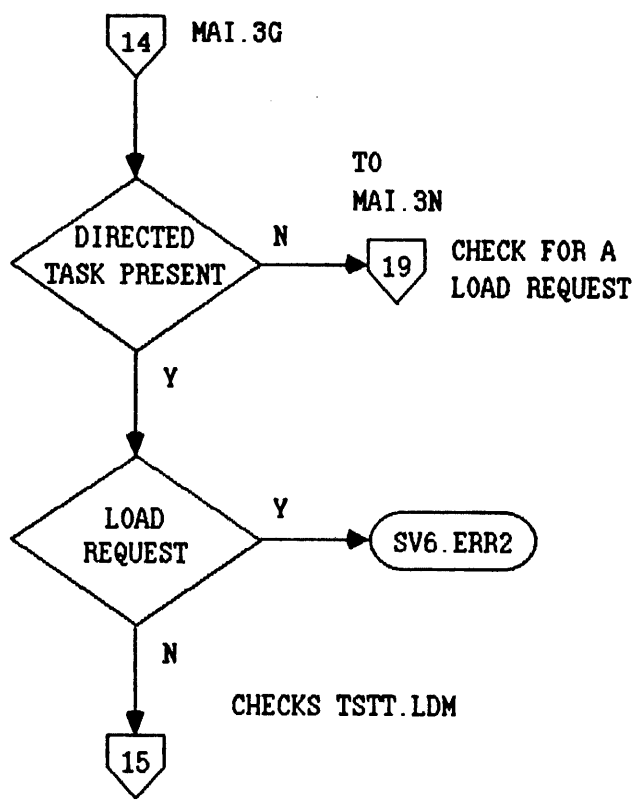
BIT 00 MAI.D00 → OTHER BY TASKID
 BIT 10 MAI.D10 → DIRECTION IS OTHER
 BIT 11 MAI.D11 → DIRECTION IS SELF
 BIT 01 ILLEGAL → SV6.ERR3

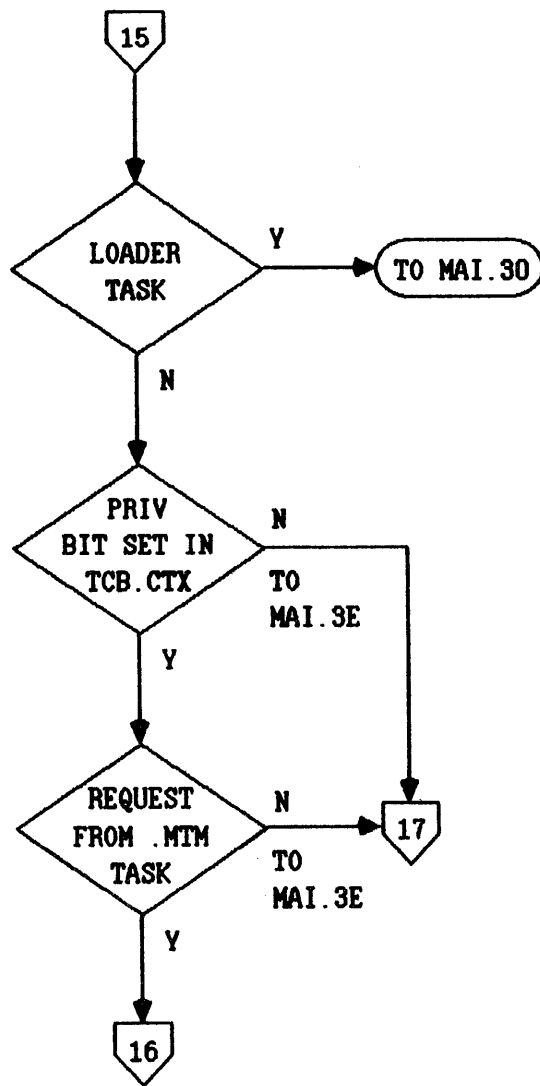


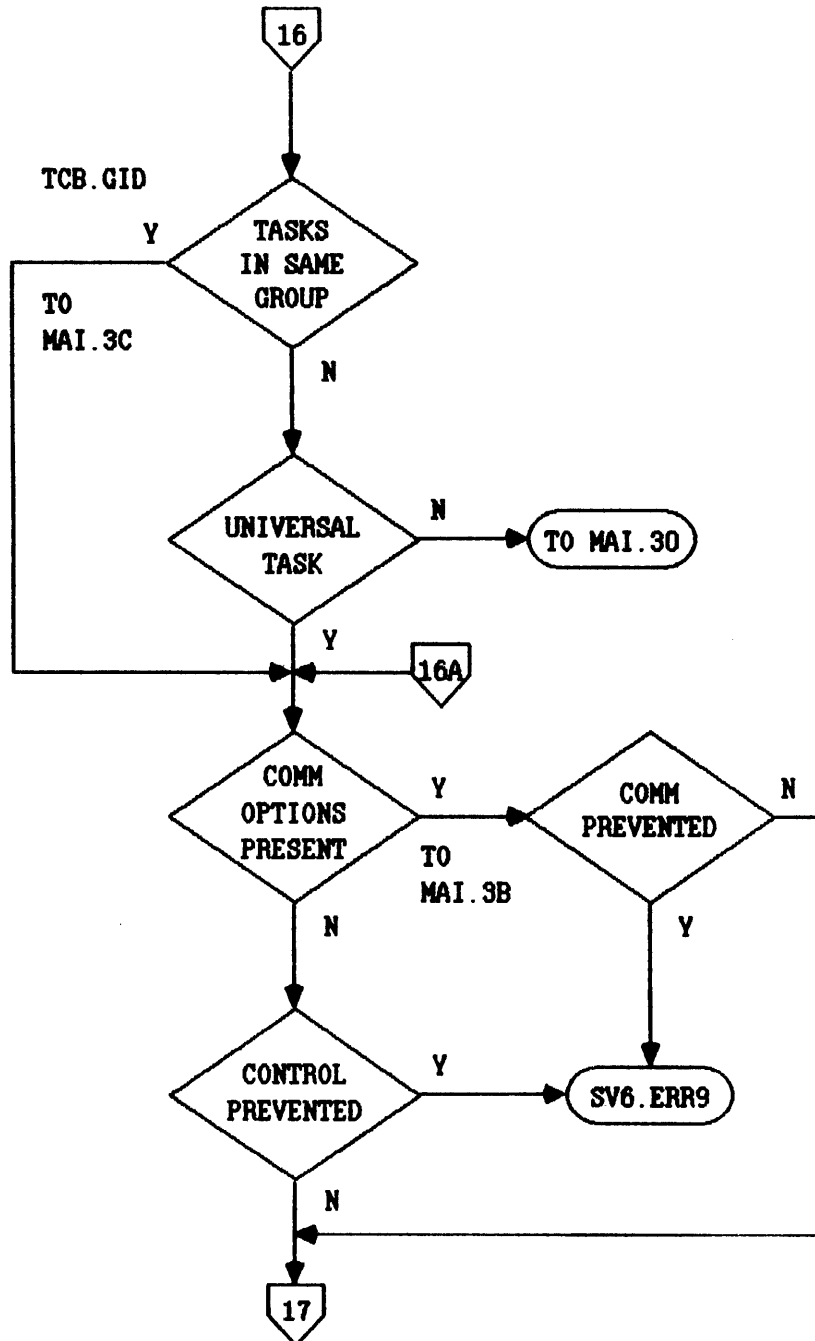




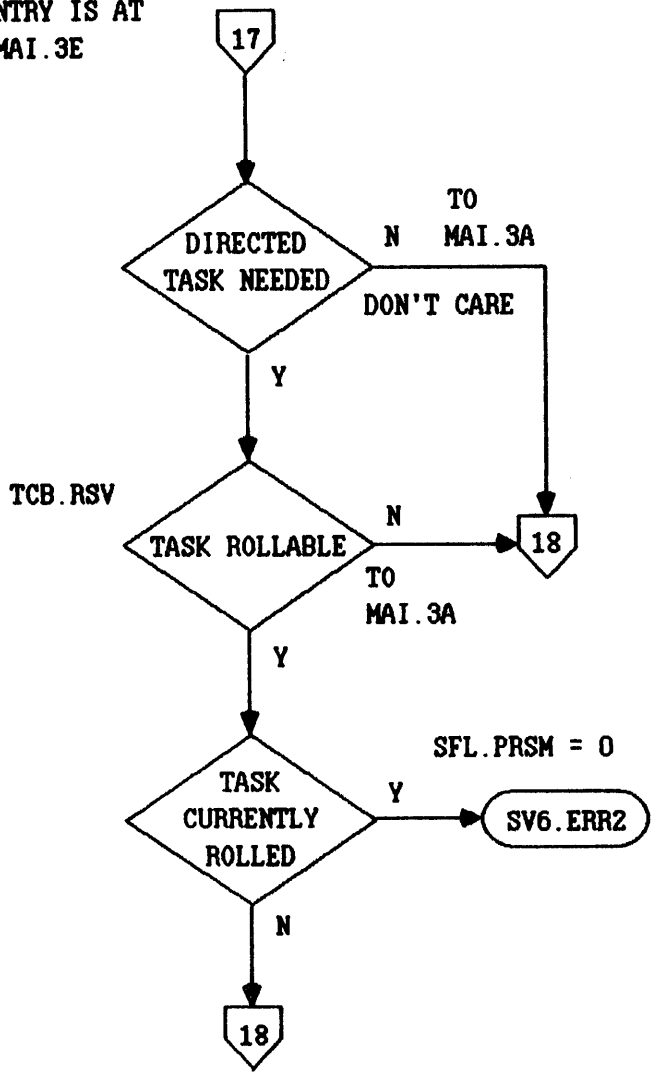


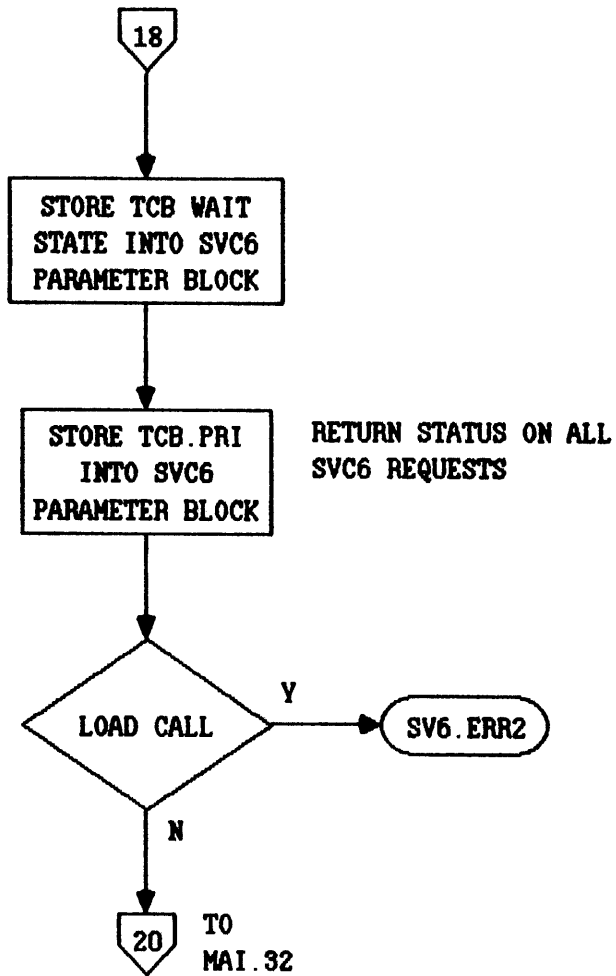






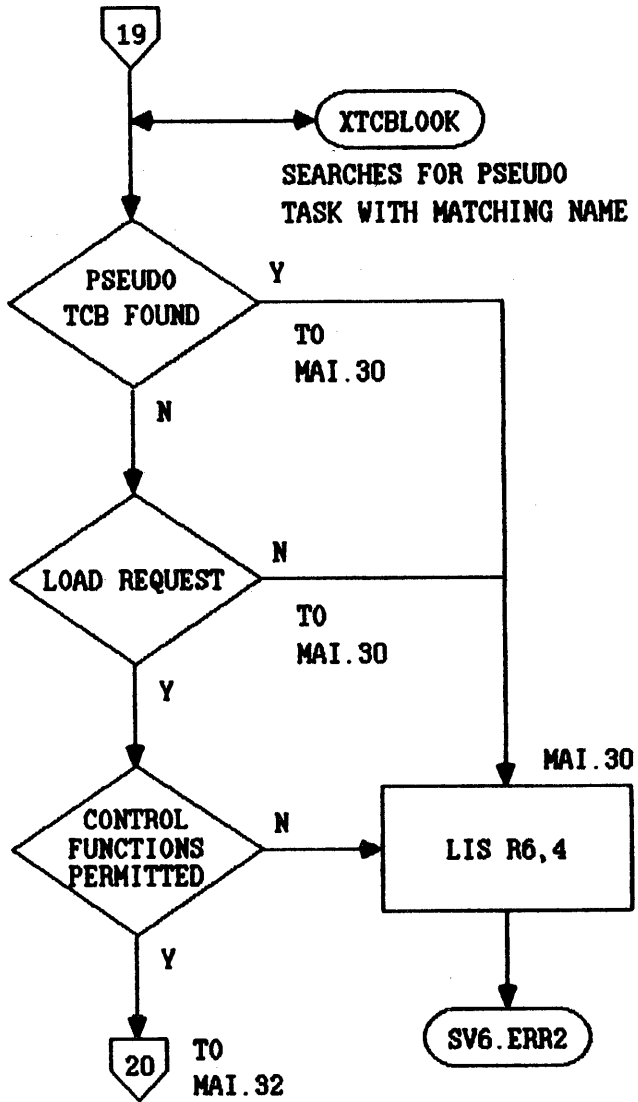
PAGE ENTRY IS AT
LABEL MAI.3E





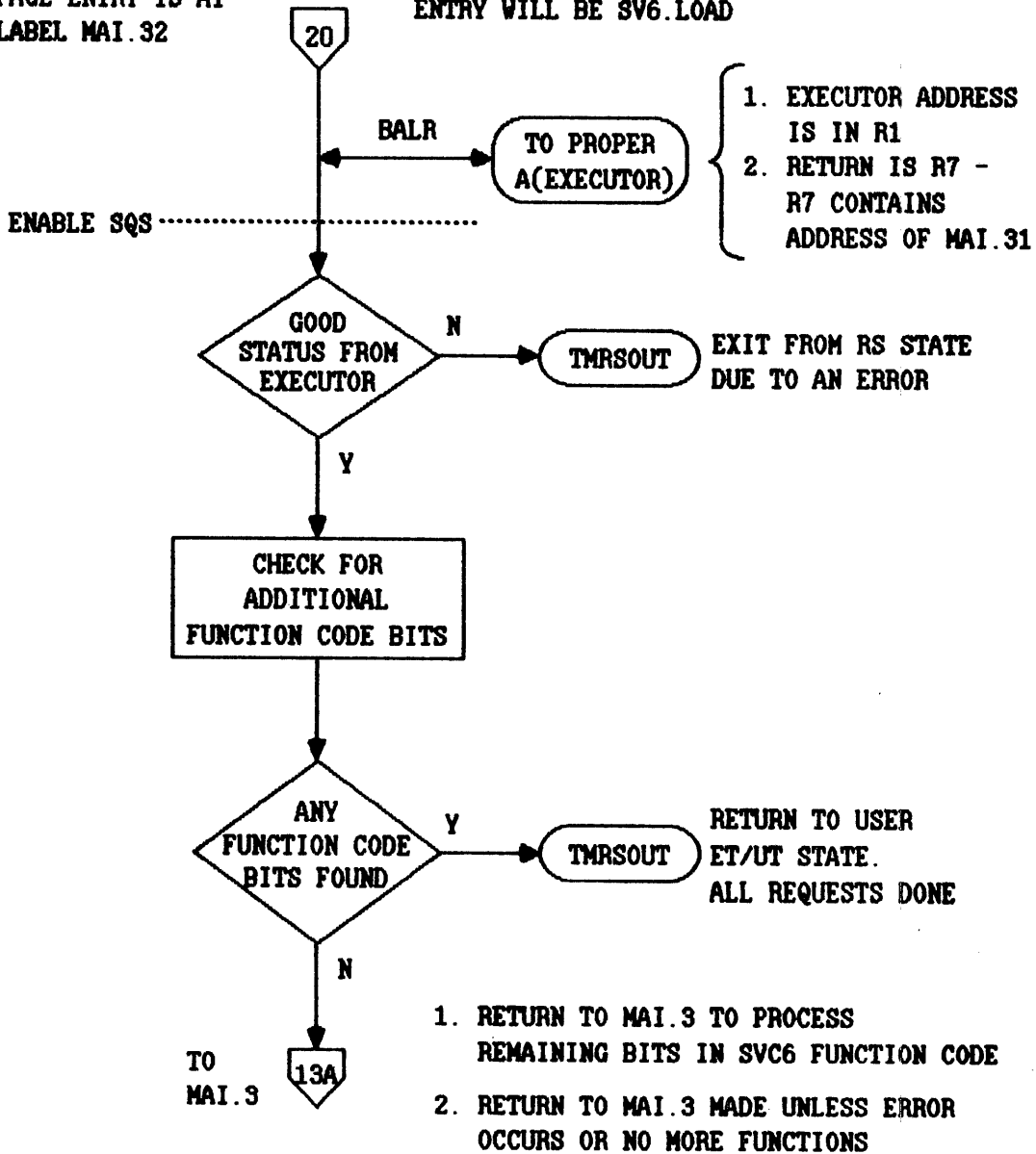
PAGE ENTRY IS AT LABEL MAI.3N

MAI.3N IS USED ONLY IF REQUESTED FUNCTION IS A LOAD



PAGE ENTRY IS AT LABEL MAI.32

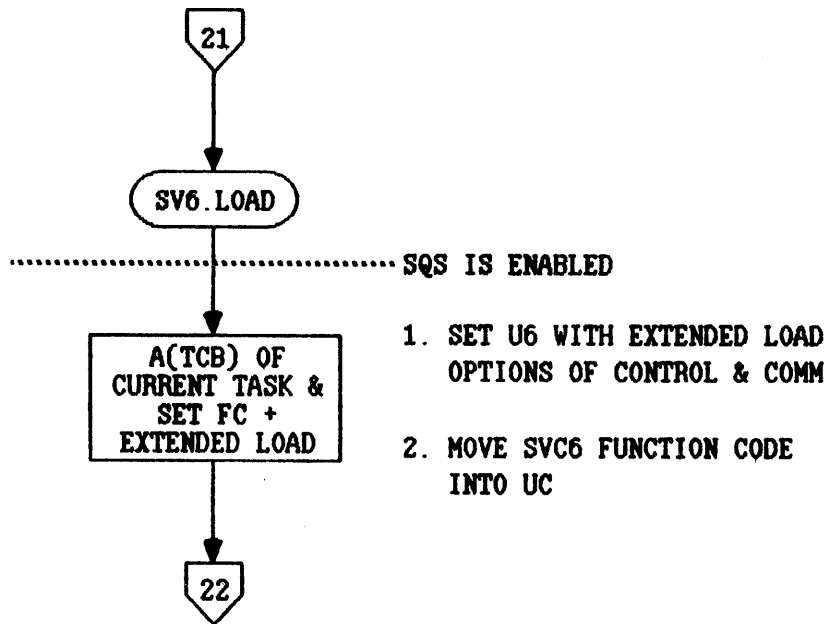
FLOW DIAGRAM TAKES US TO LOAD EXECUTOR: ENTRY WILL BE SV6.LOAD

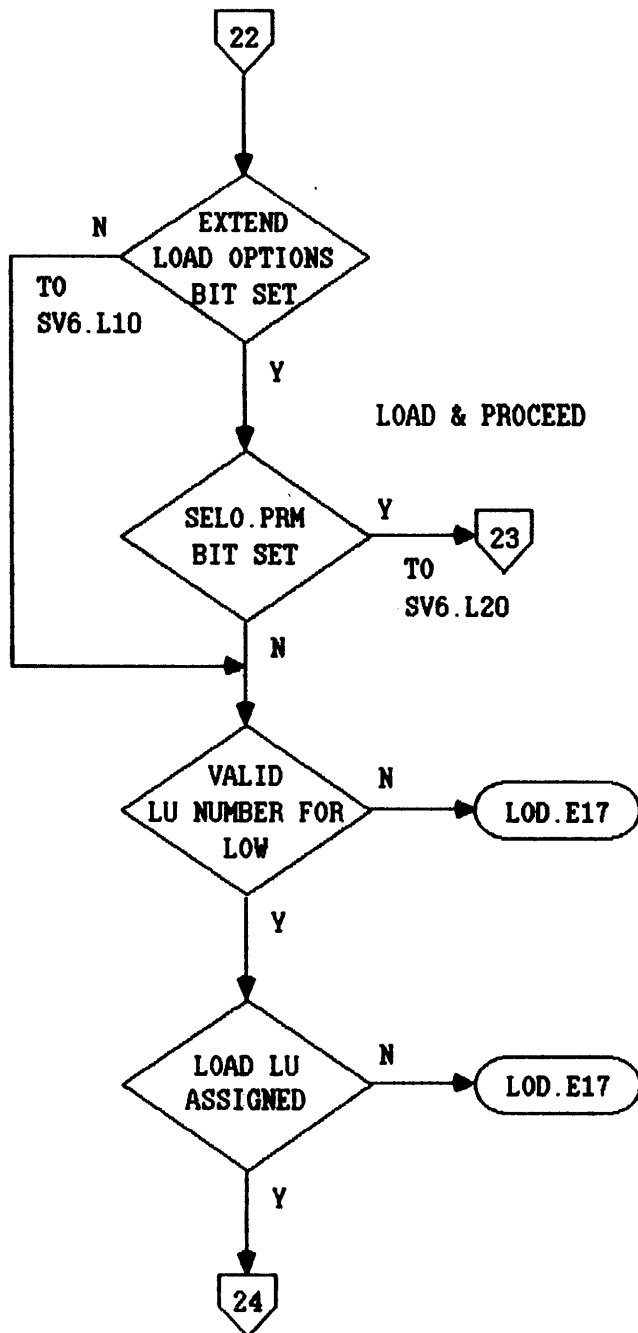


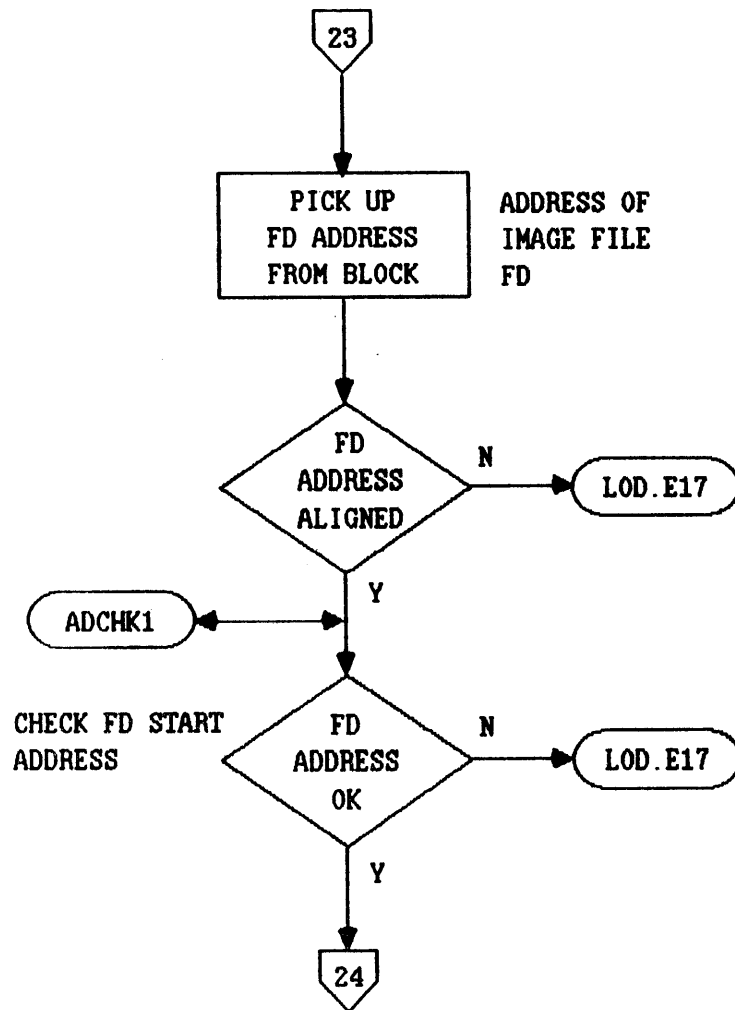
SV6.LOAD

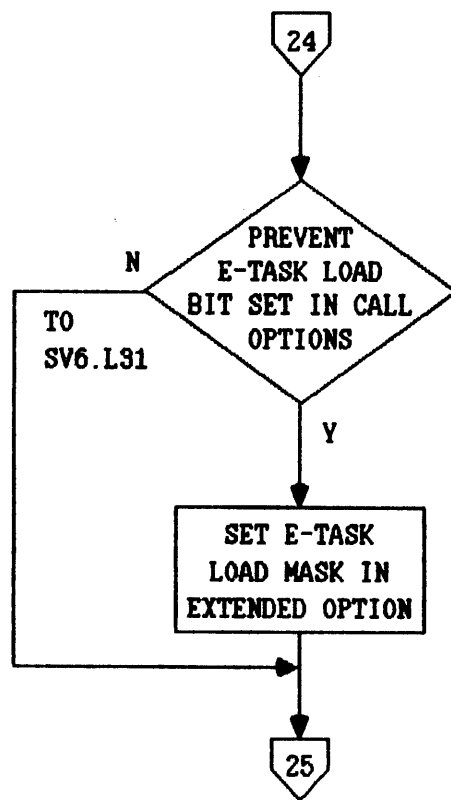
ENTRY IN RS STATE

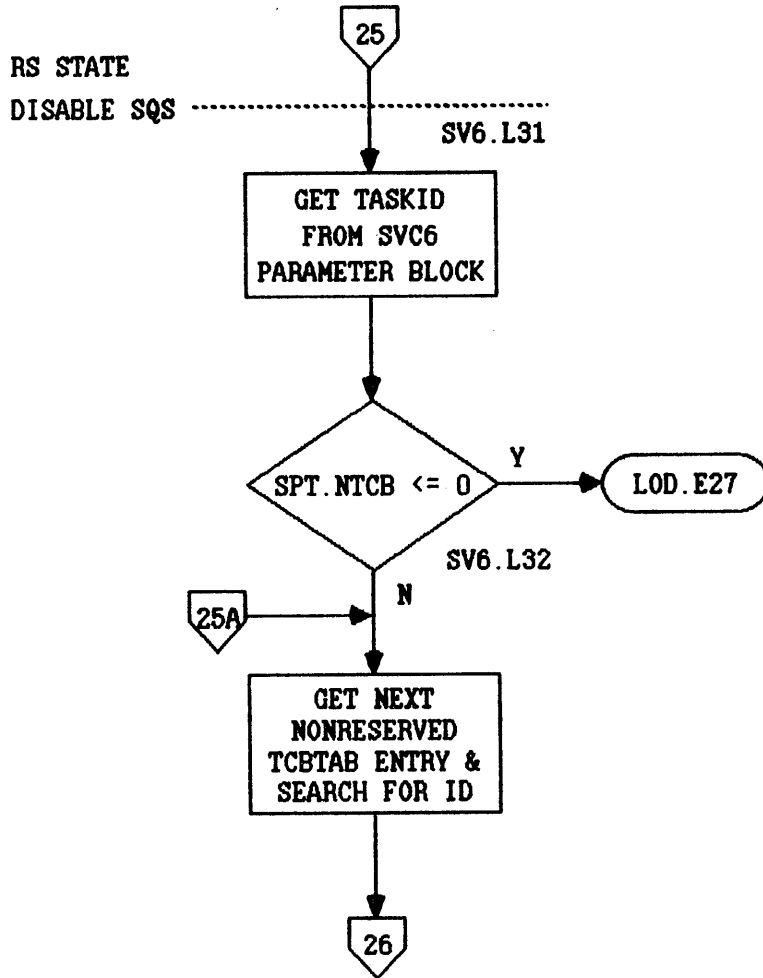
- U4 → FUNCTION CODE POINTER**
- U7 → RETURN**
- UD → SVC6 PARAMETER BLOCK**

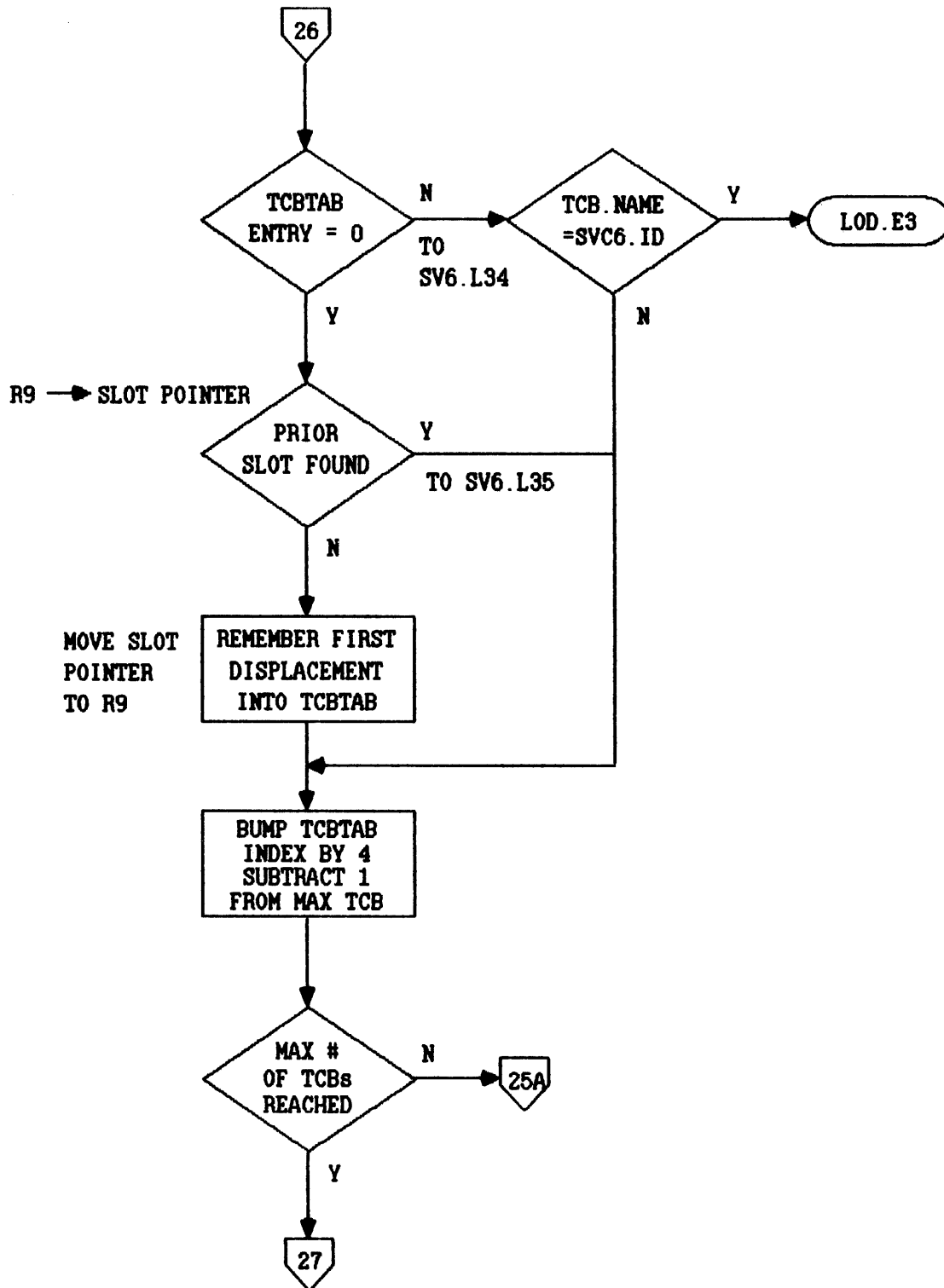


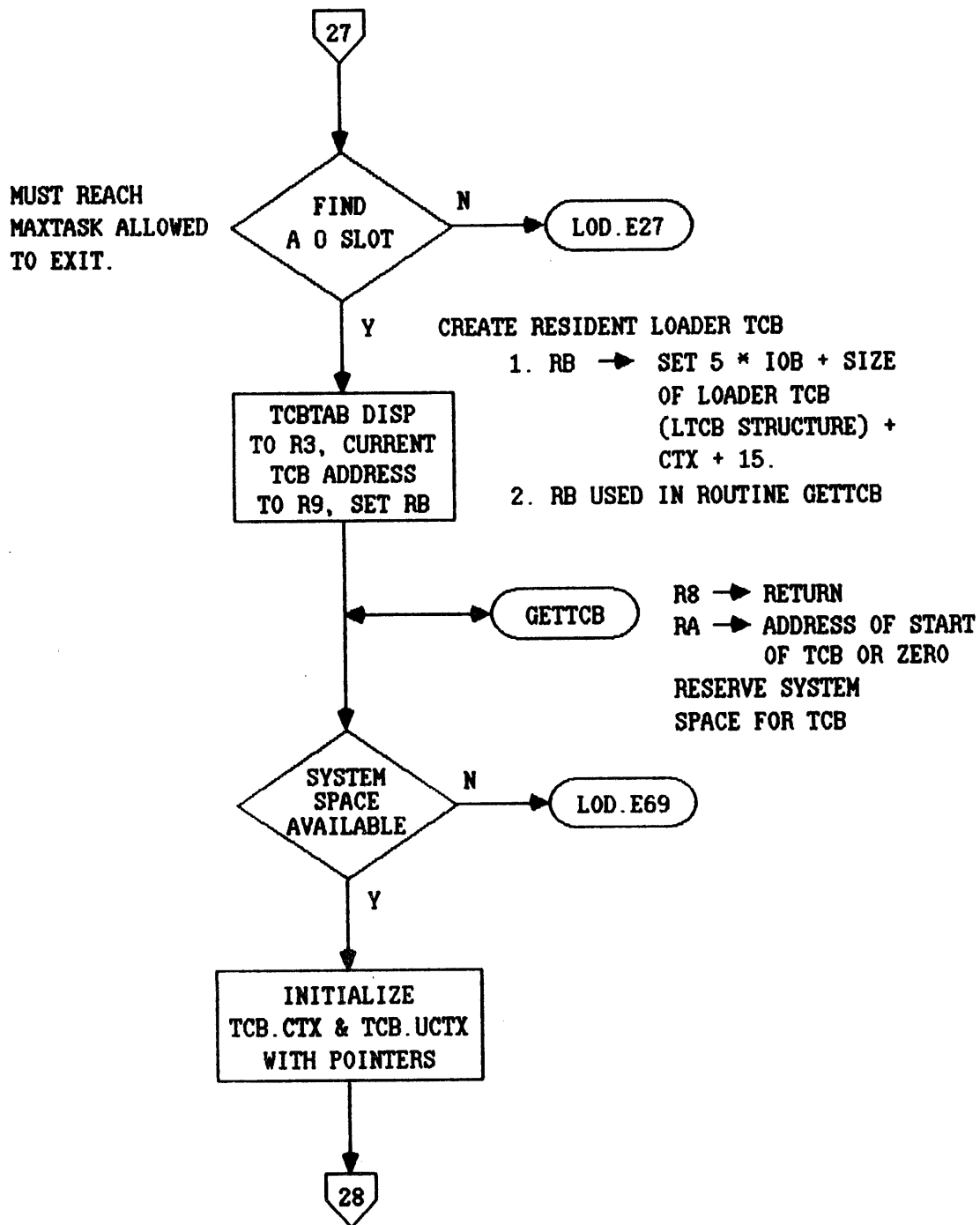


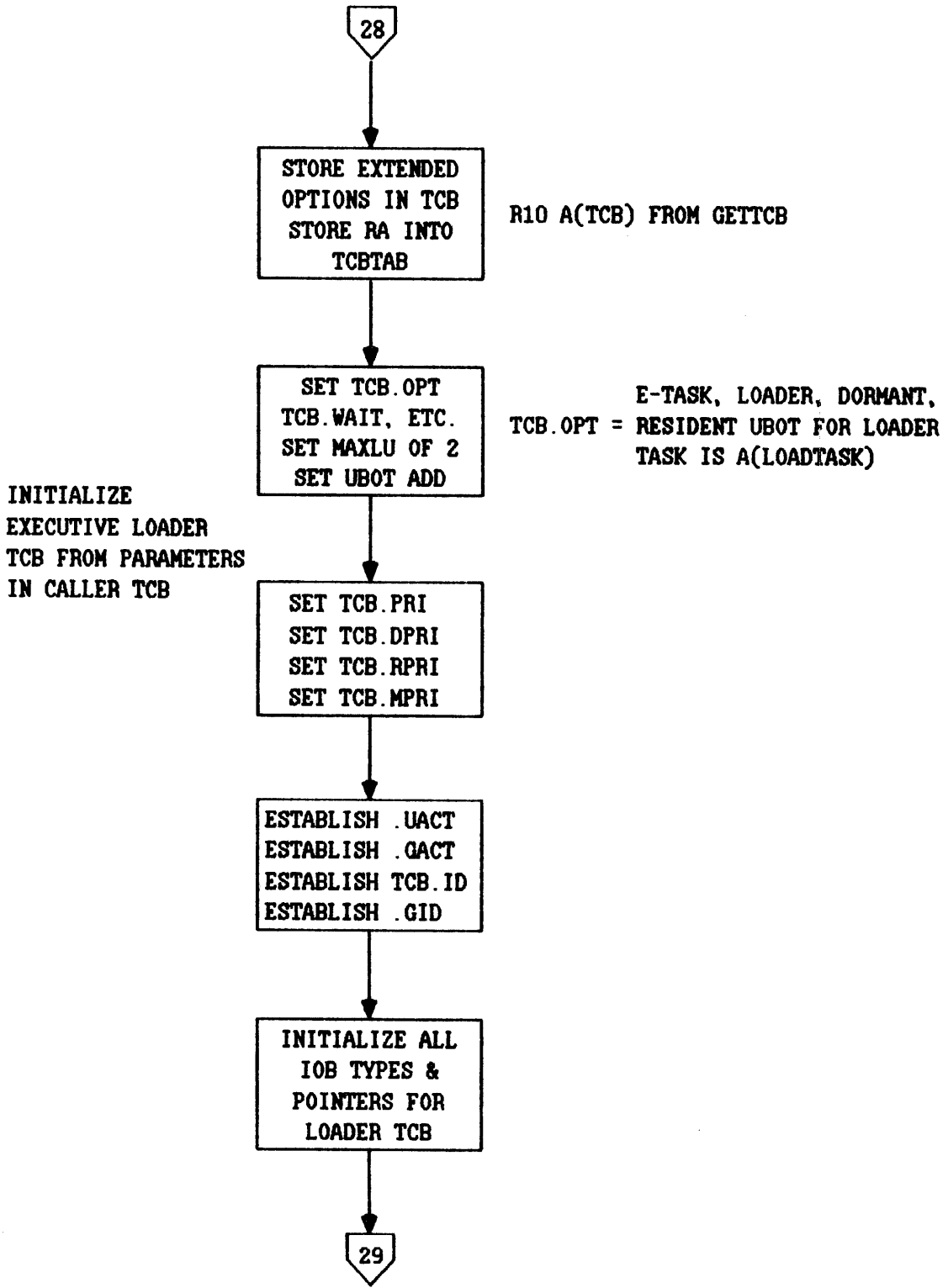


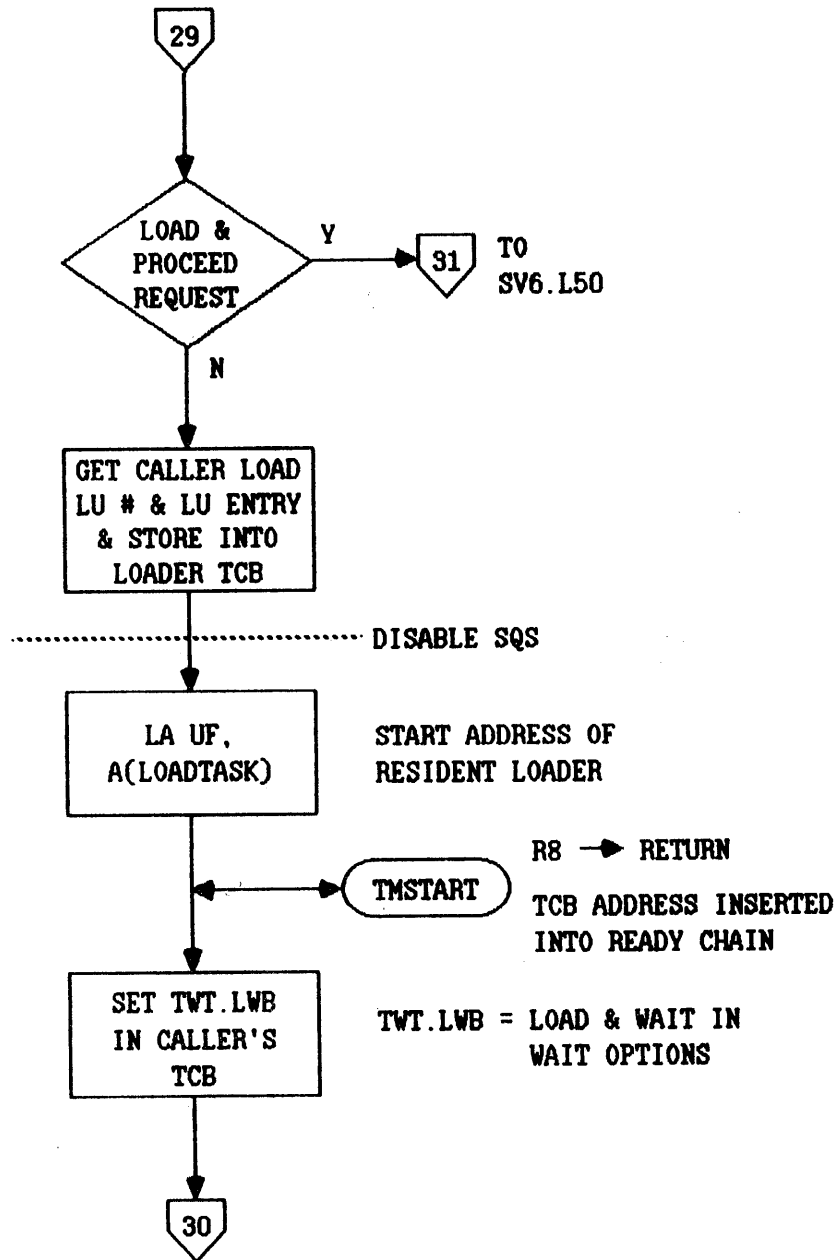


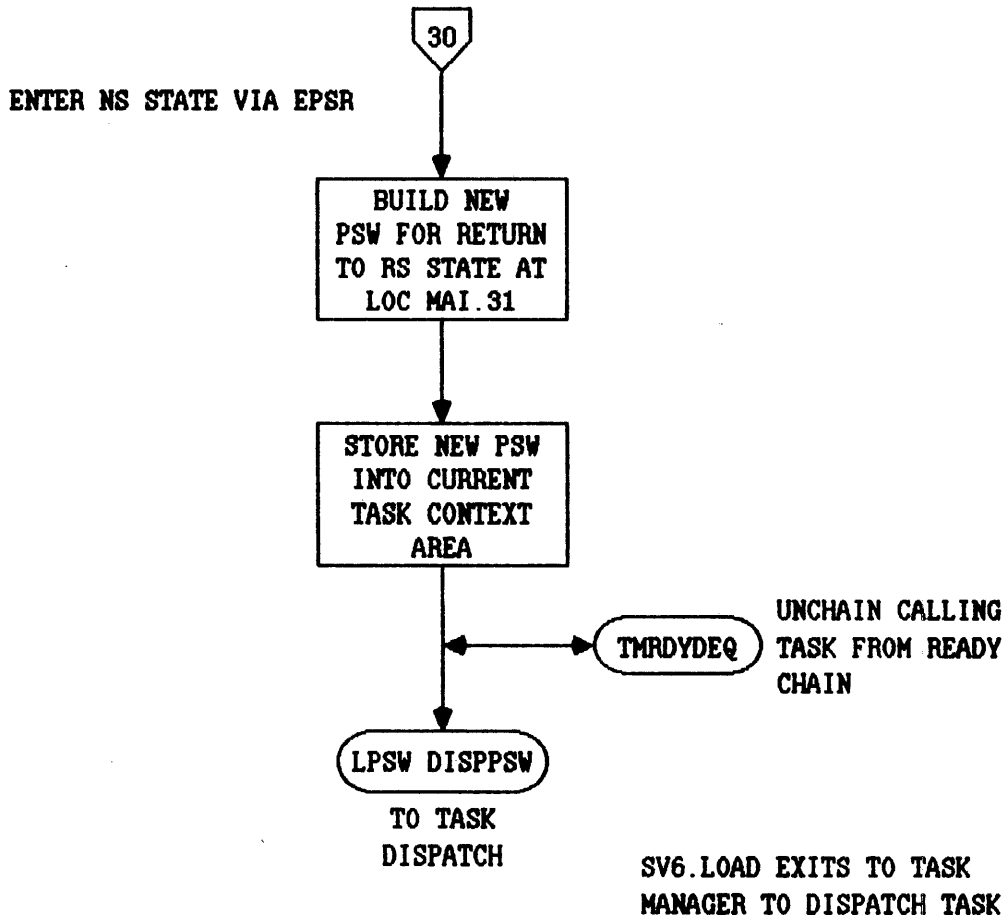












FOR LOAD-WAIT-FLOW CONTINUATION
SEE CONNECTOR 34.(LOADTASK)

LOAD & PROCEED

31

SVC.L50

INCREMENT LOAD
& PROCEED COUNT
IN TCB

GET FD TASK TO
LOAD & STORE IN
SVC7 PARAMETER
BLOCK IN LOADER TCB

START BITS
SET IN FC

N

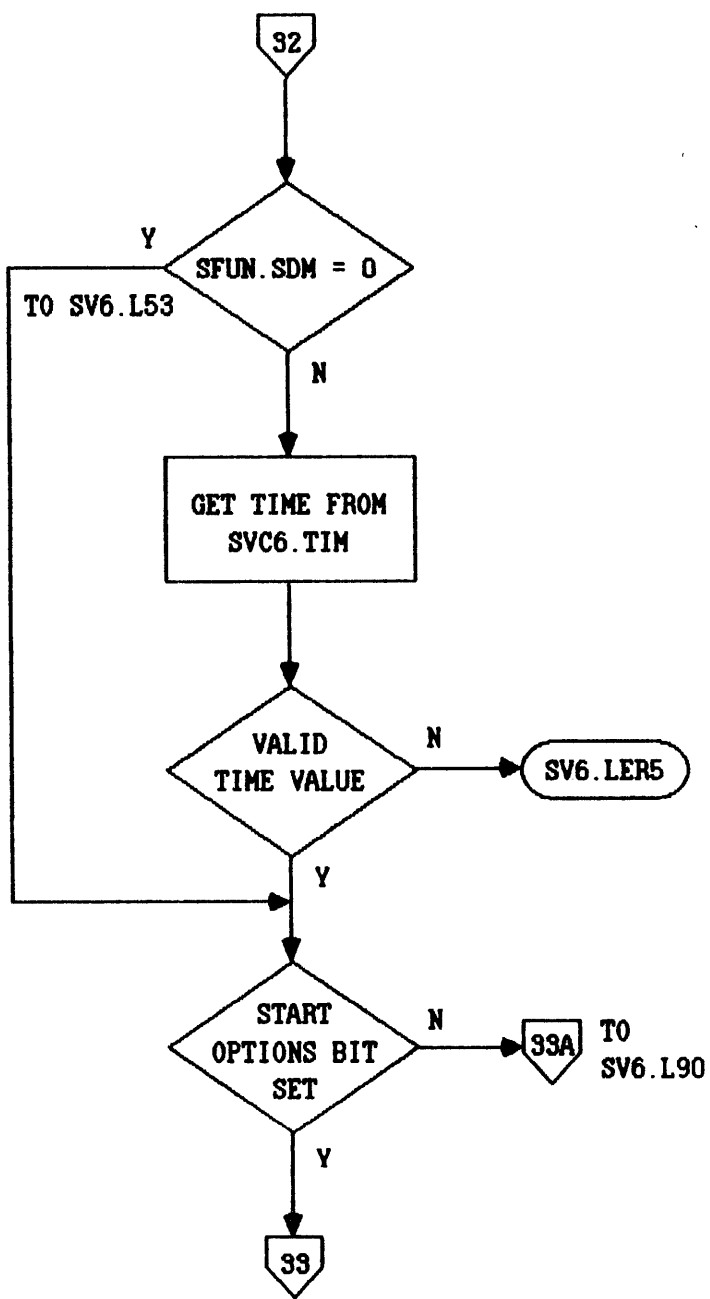
33A

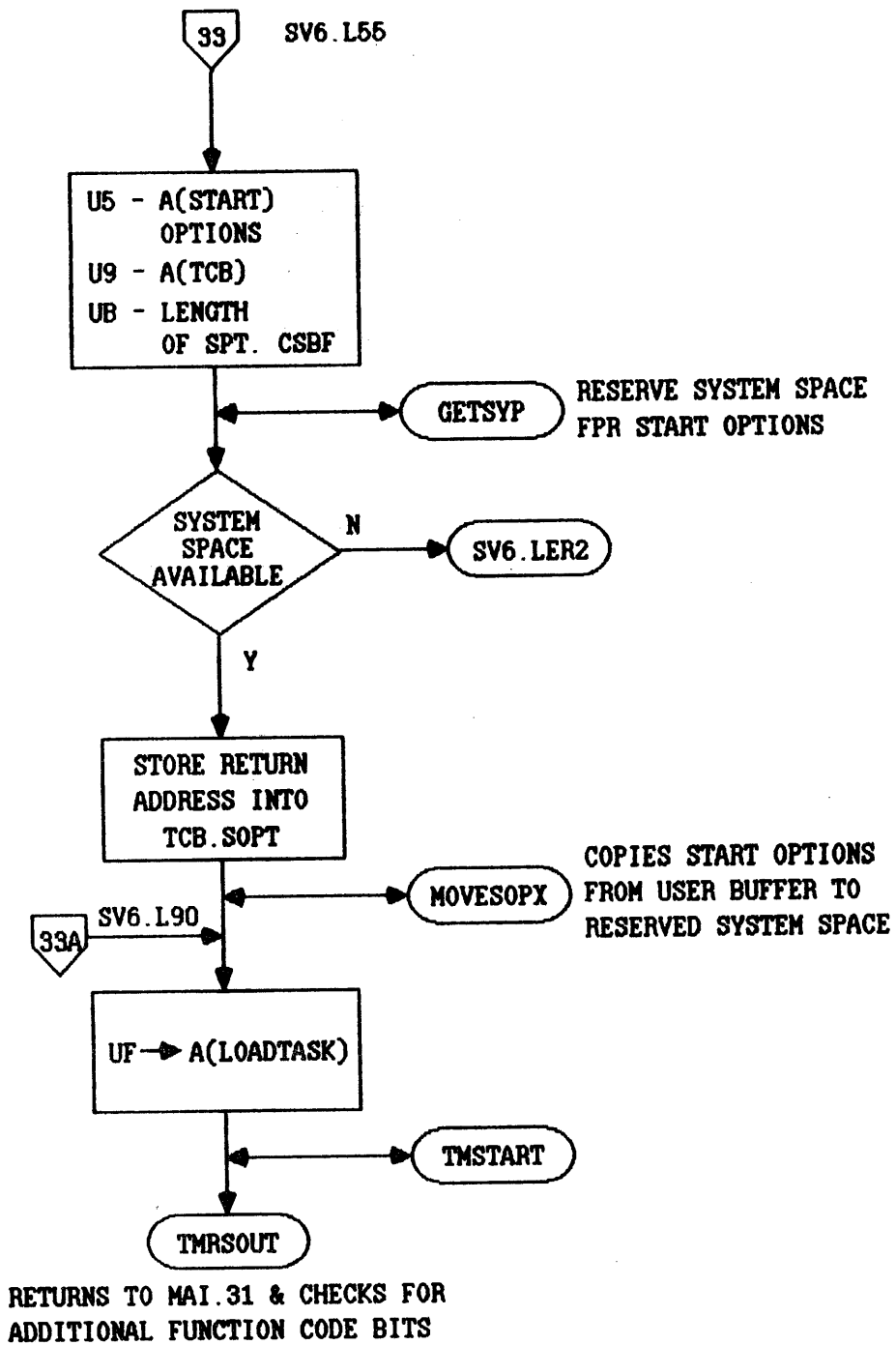
TO
SV6.L90

Y

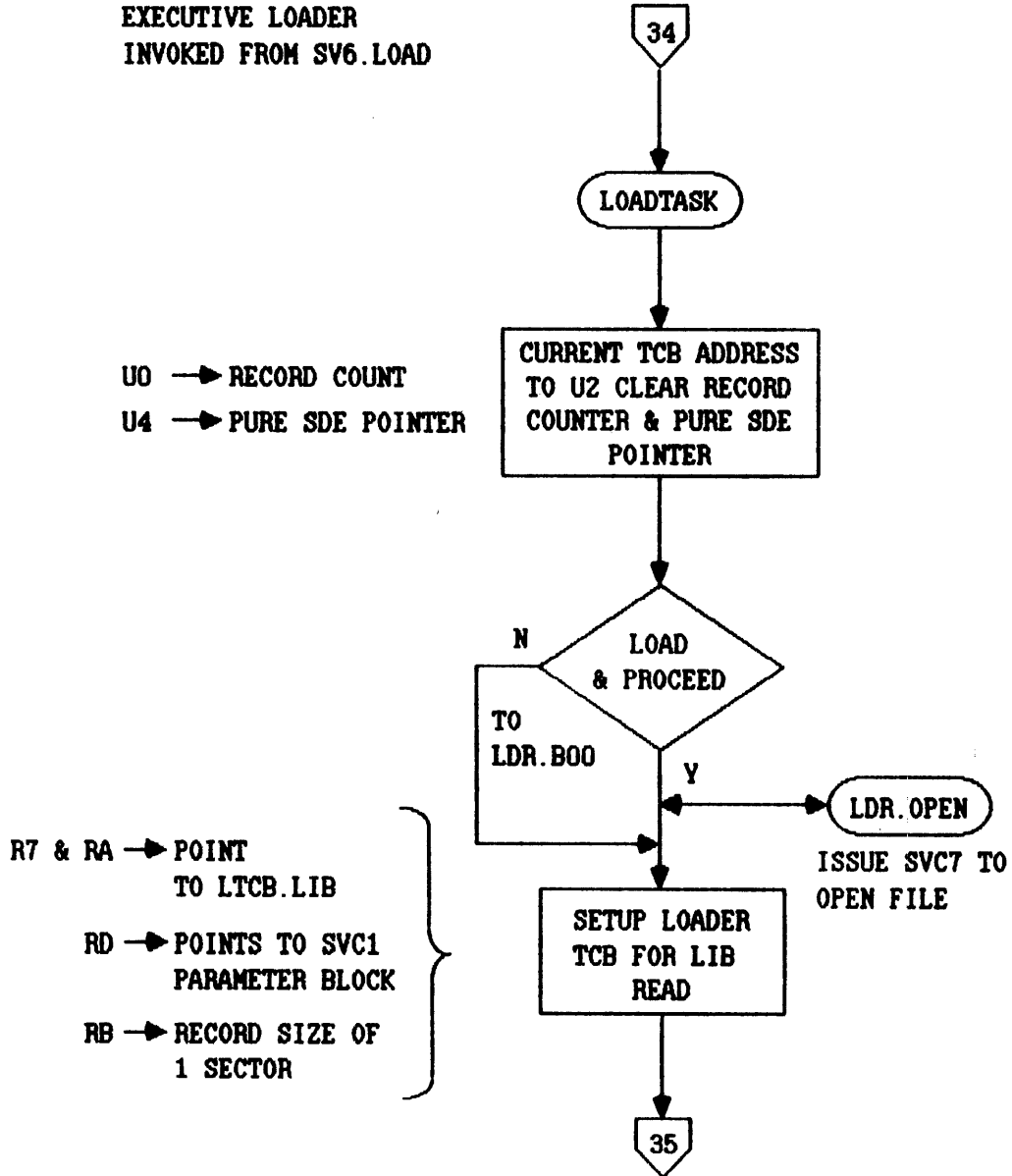
GET SVC6.SAD FROM
PARM BLOCK & STORE
AT TCB.SLOC

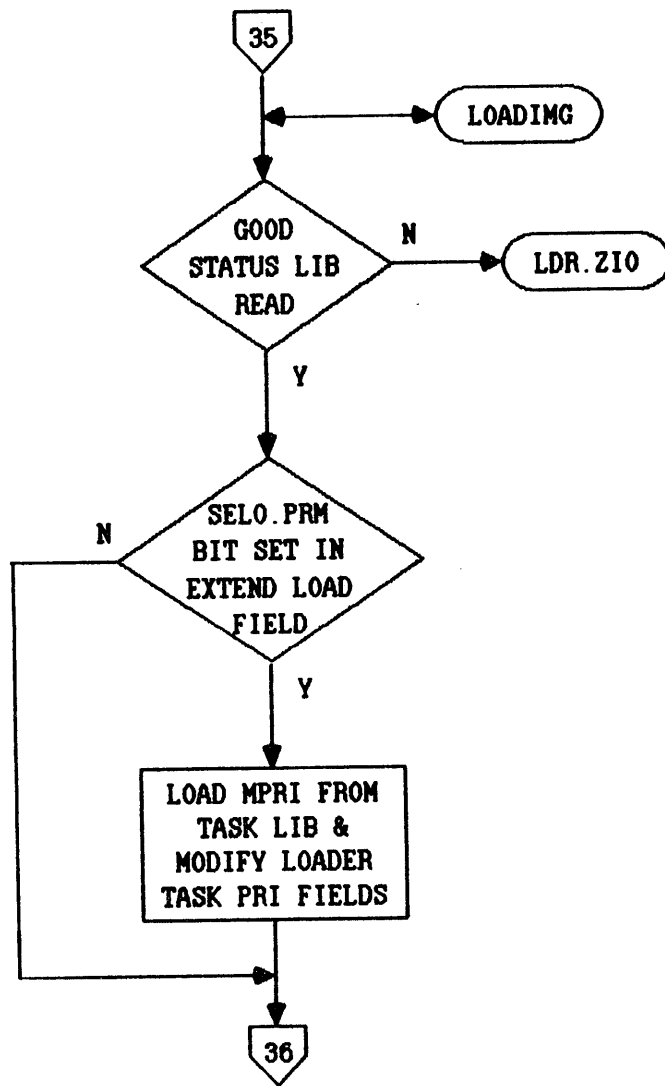
32

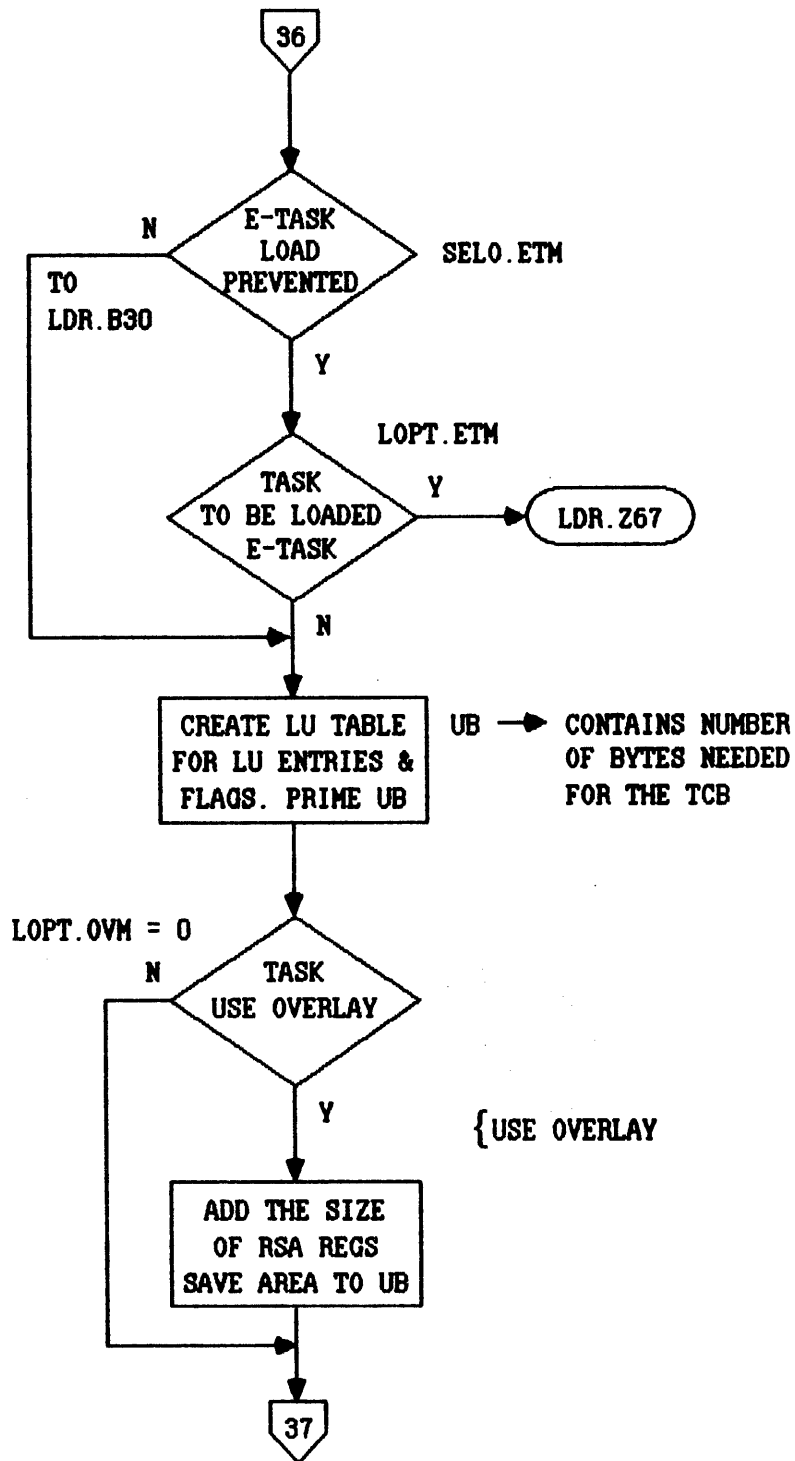


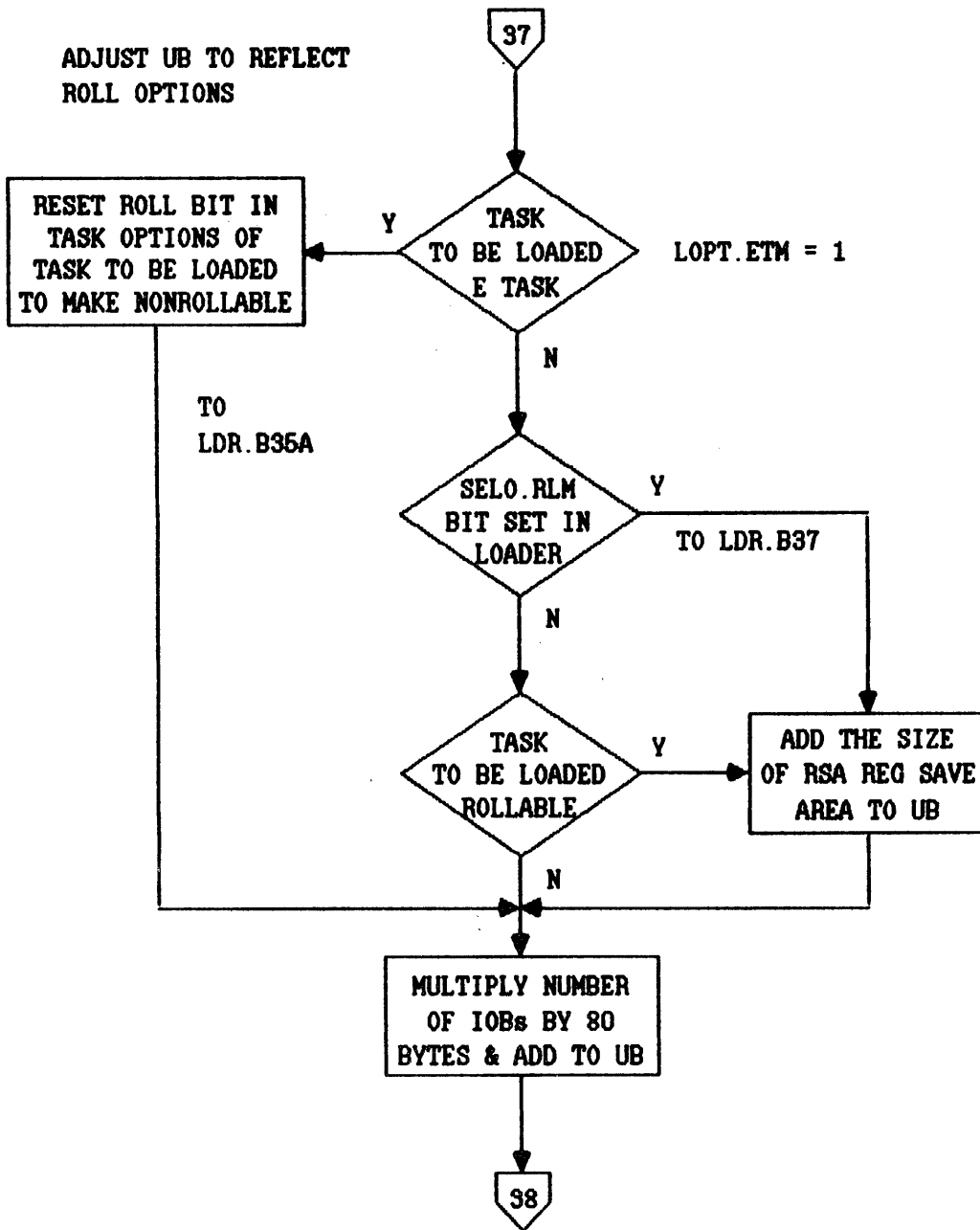


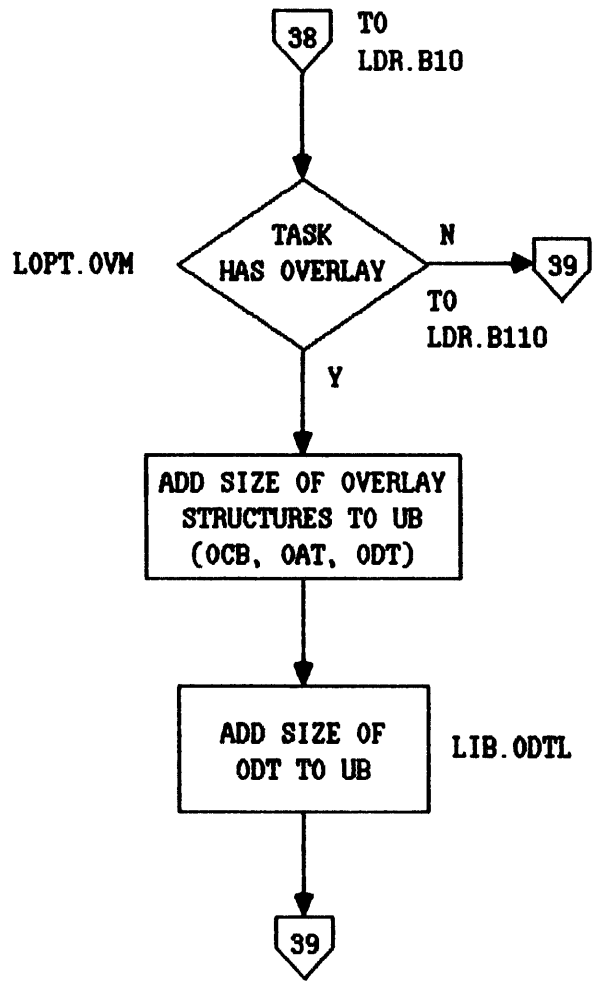
EXECUTIVE LOADER
INVOKED FROM SV6.LOAD

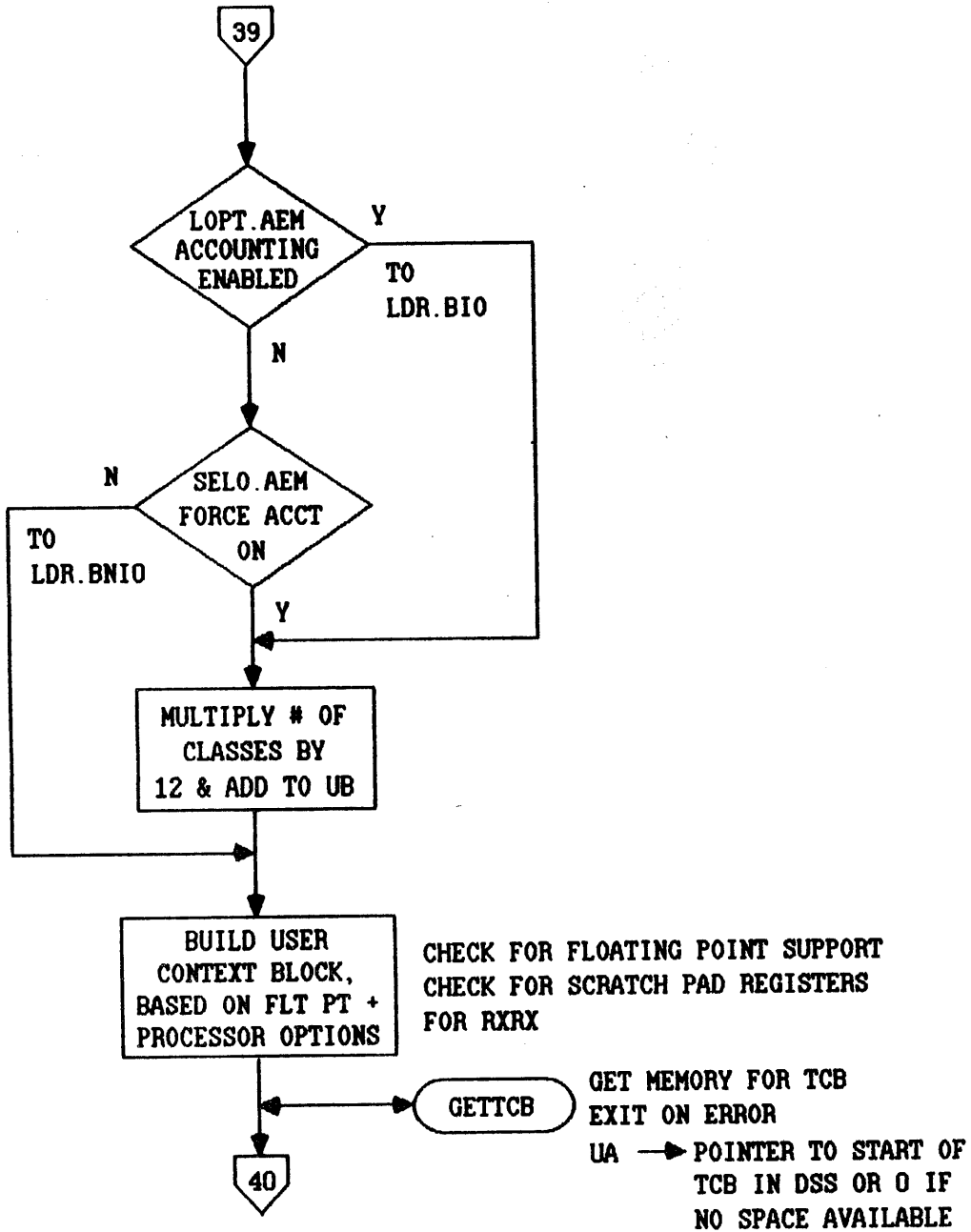


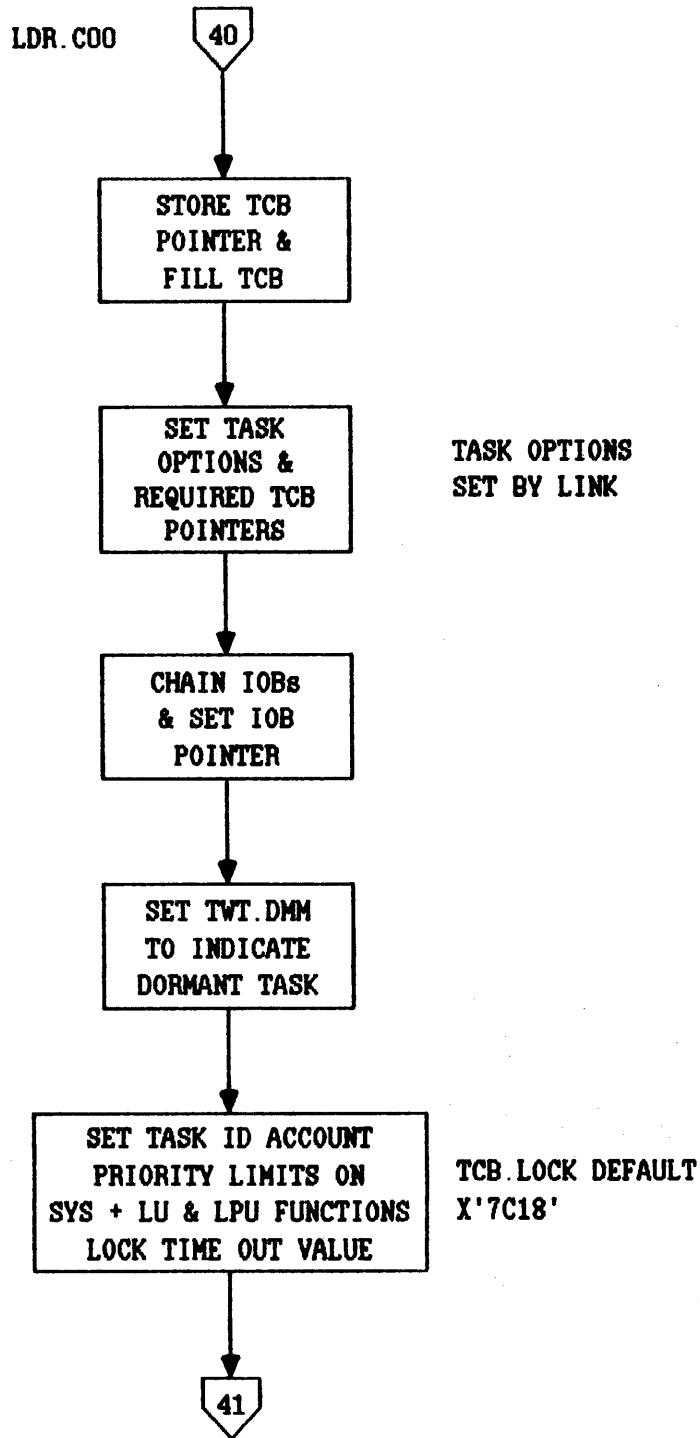


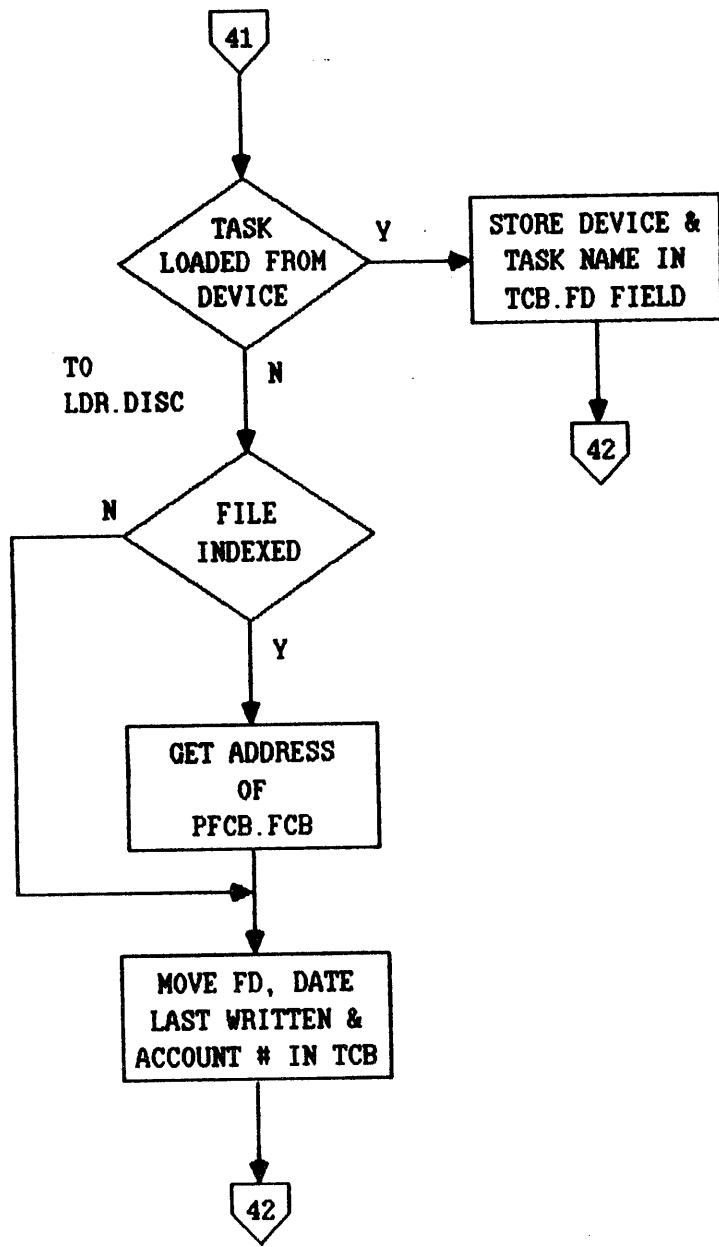


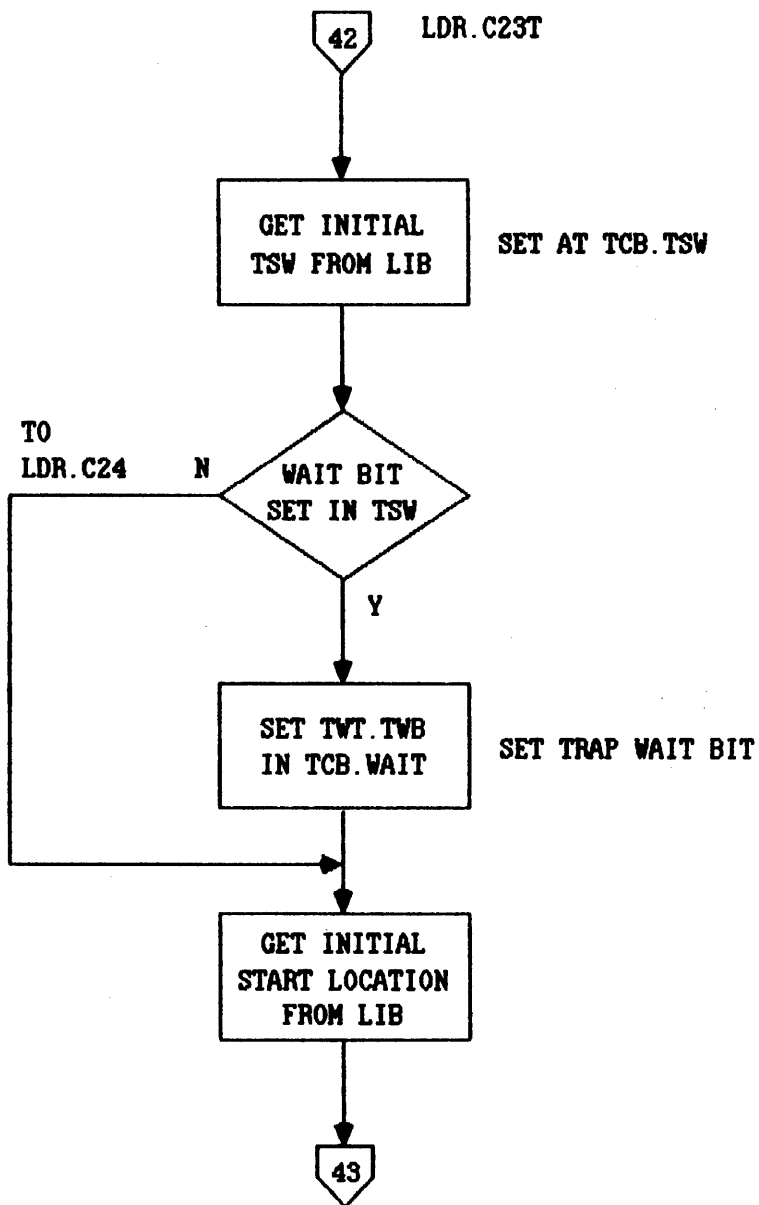


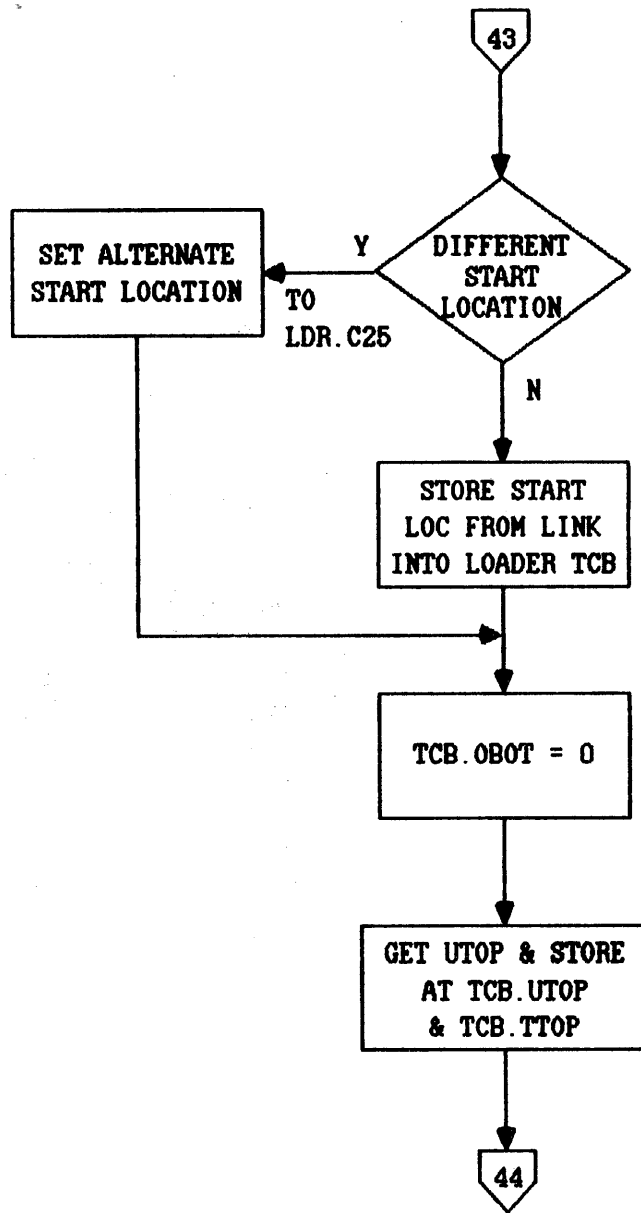


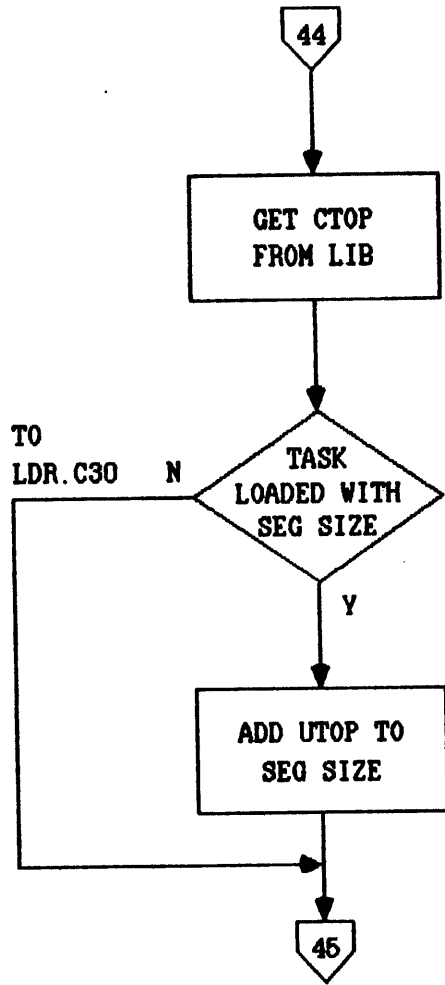


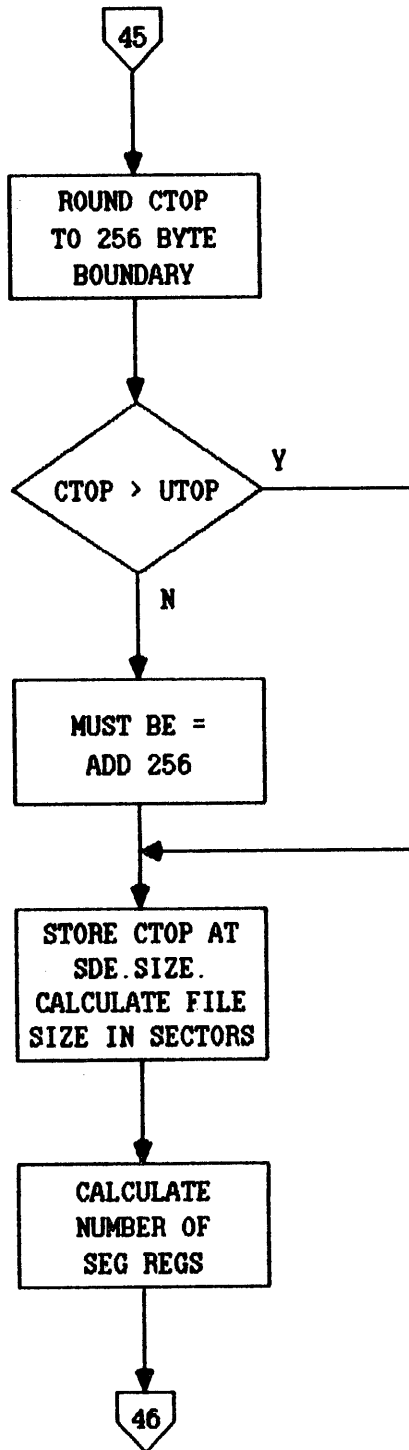




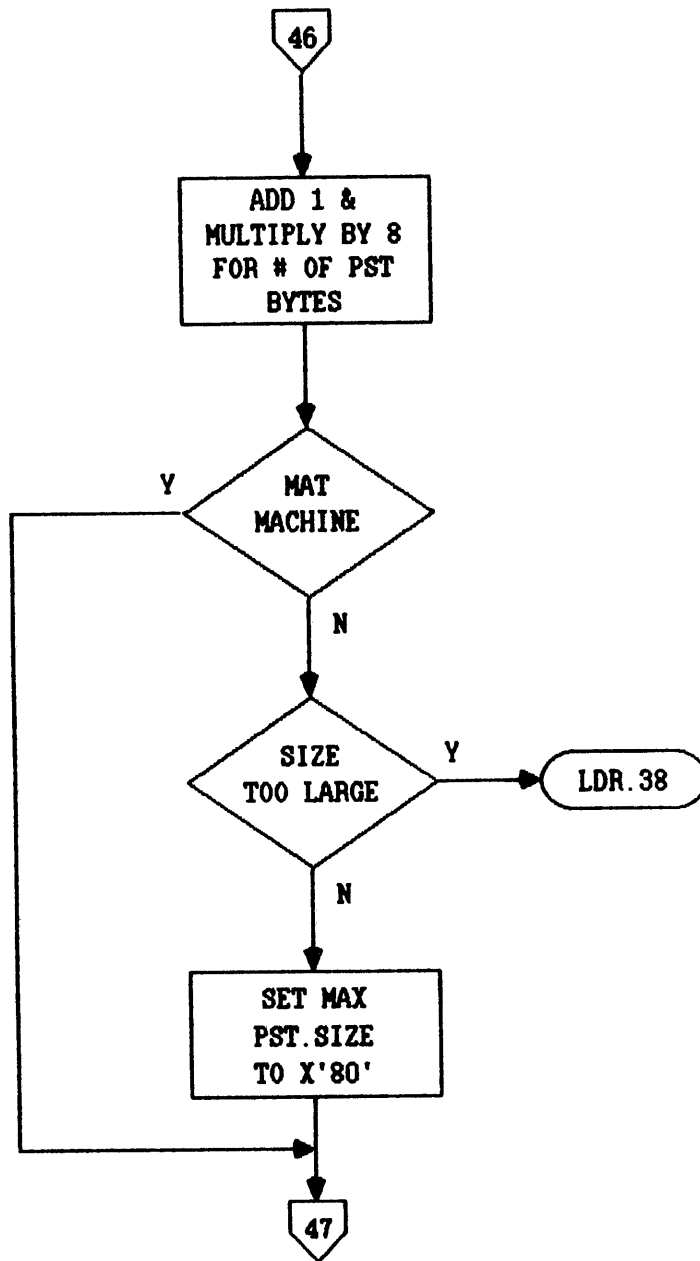


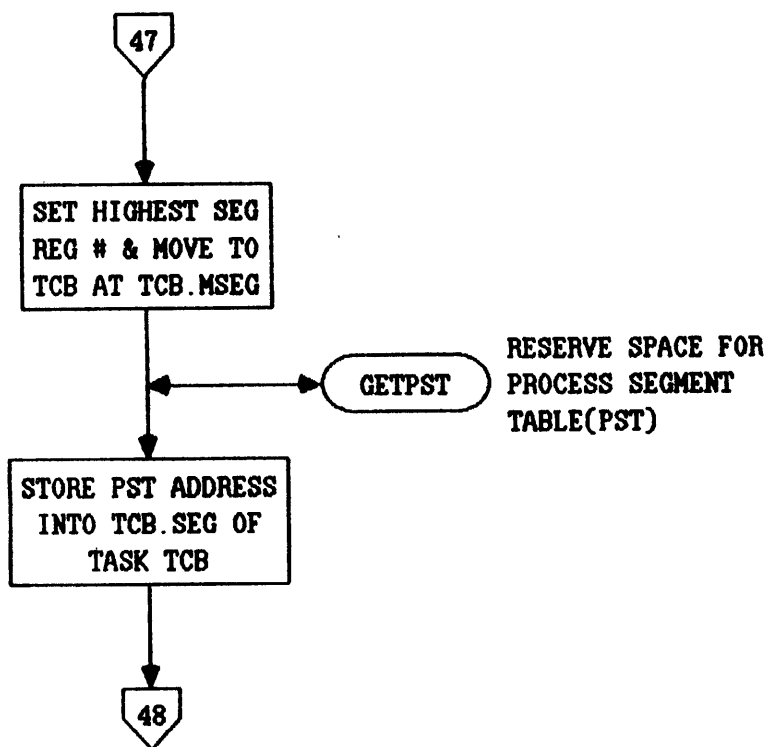


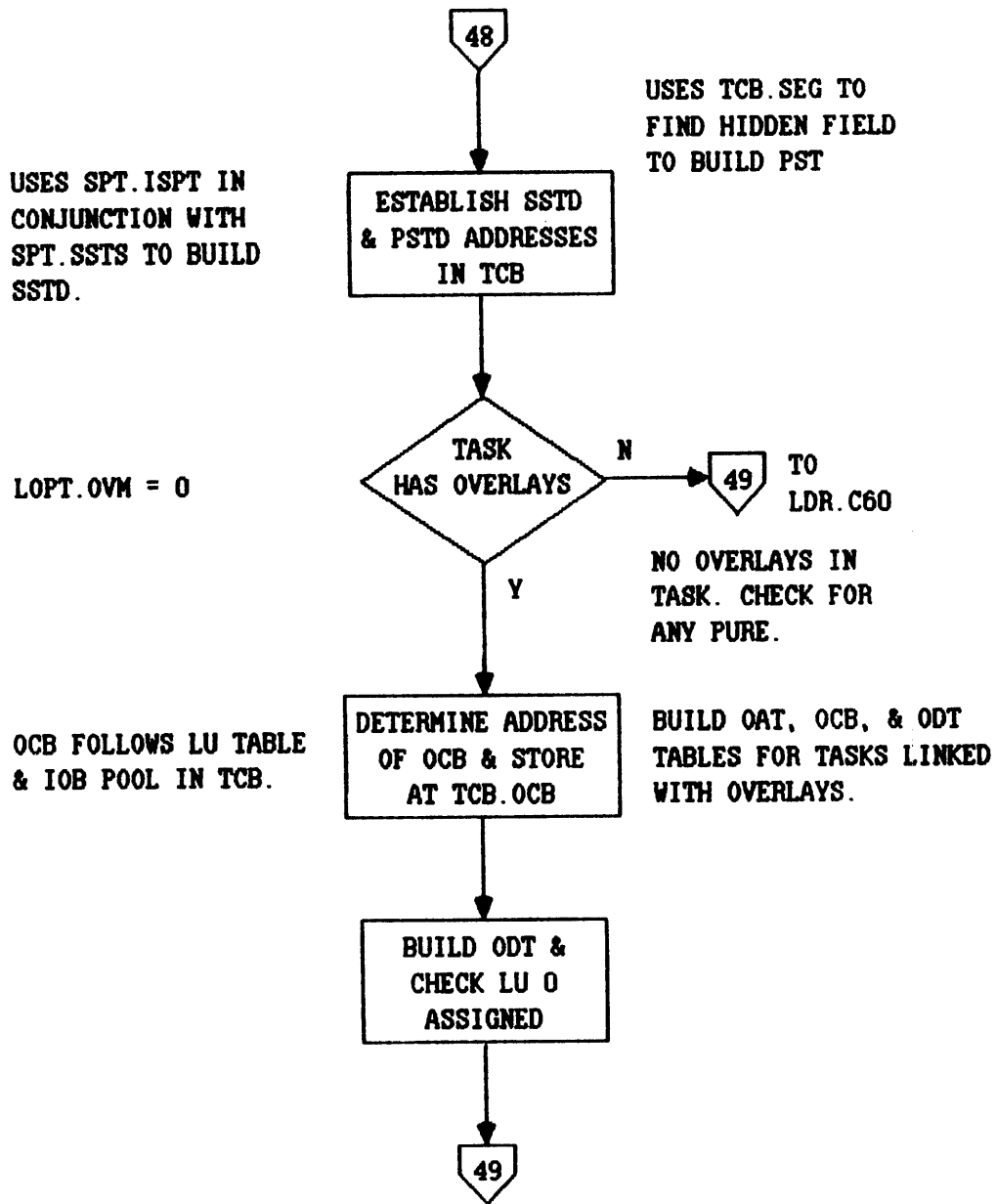


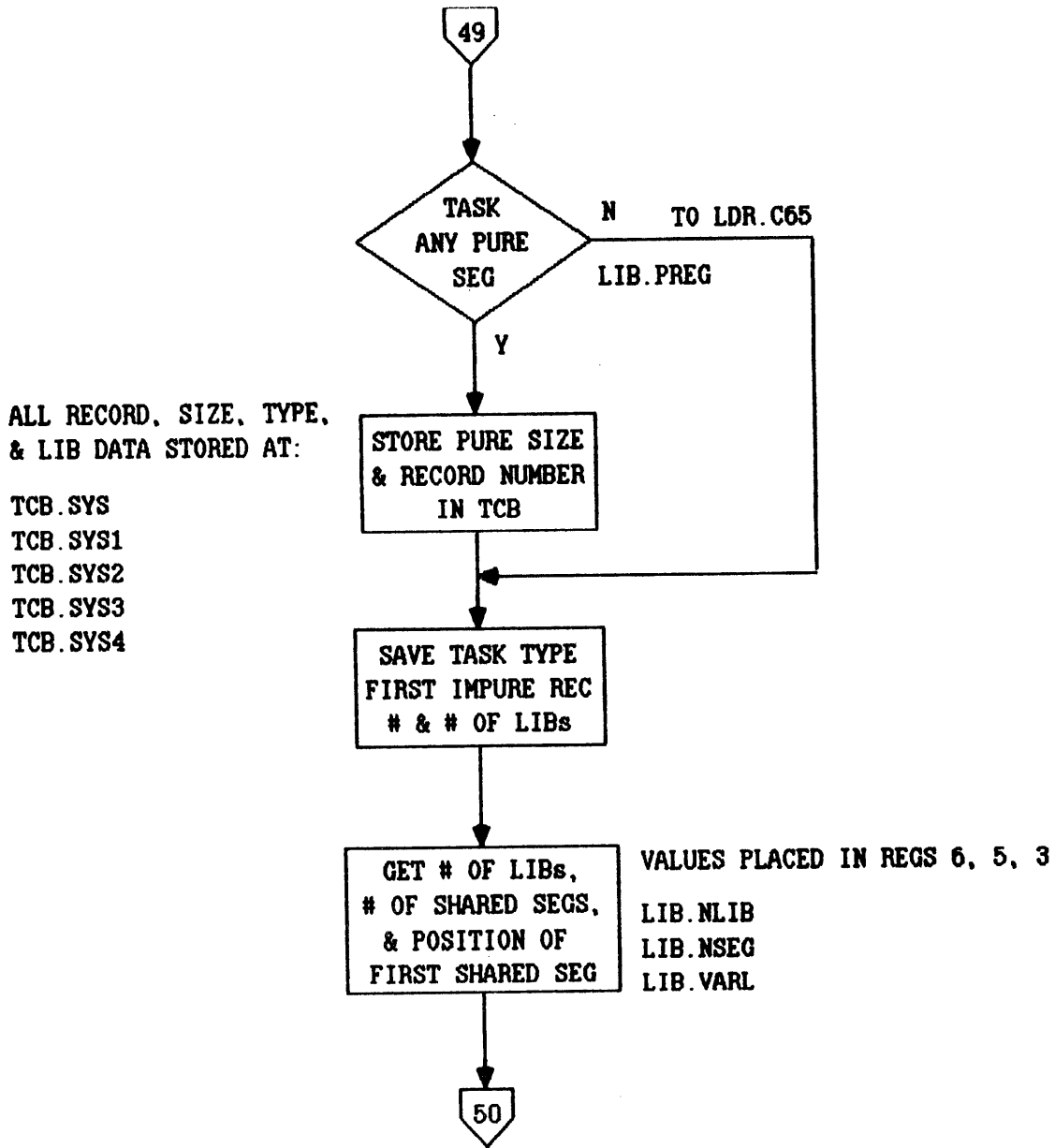


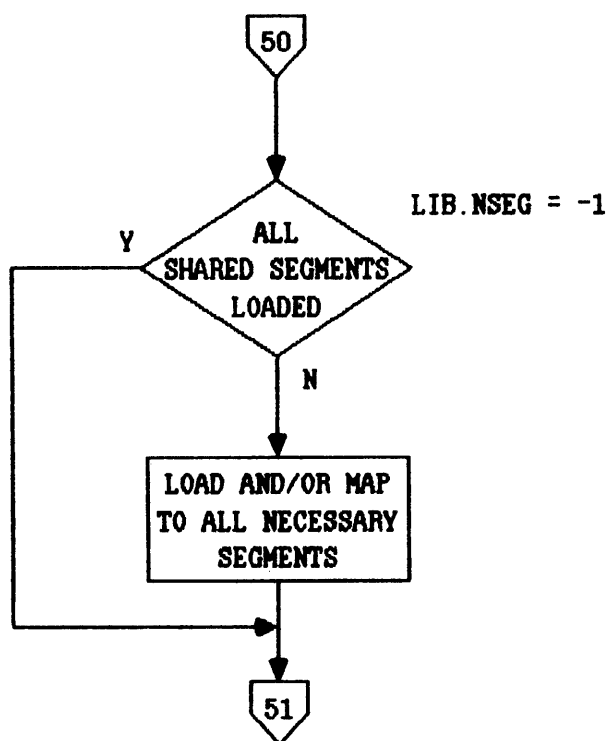
1. FILE SIZE IN SECTORS STORED IN TCB.CTOP
2. TCB.UBOT NOW INITIALIZED TO A -1
3. A(TCB) TO R6

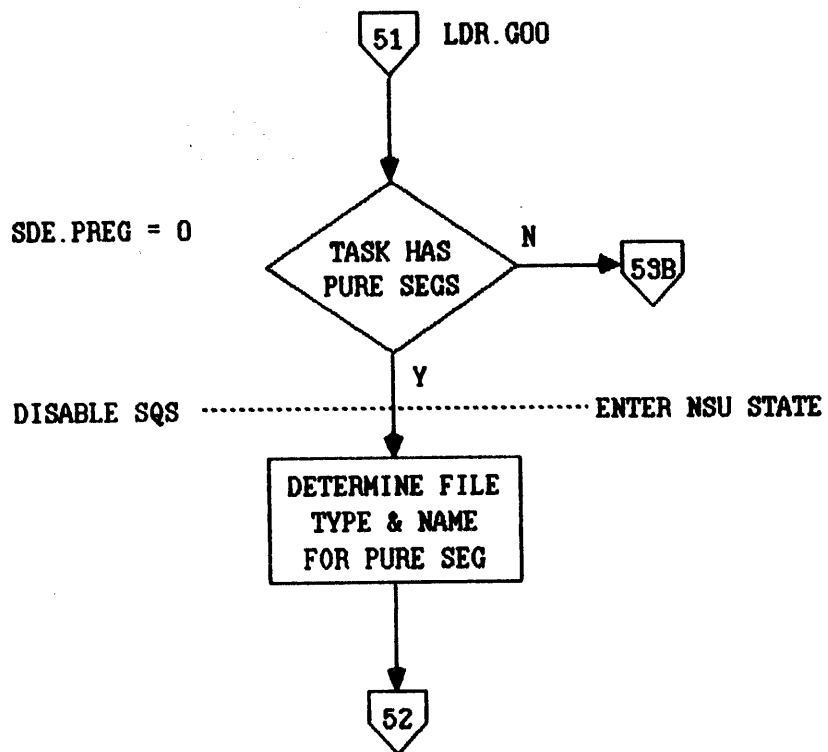


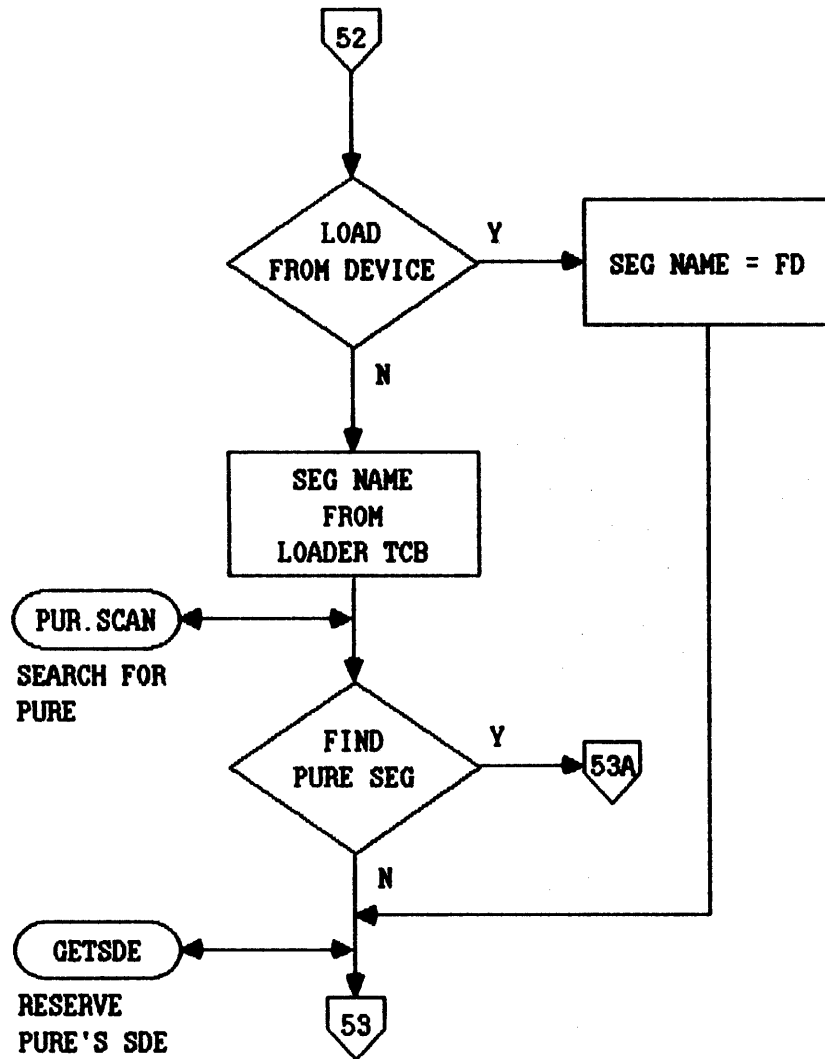


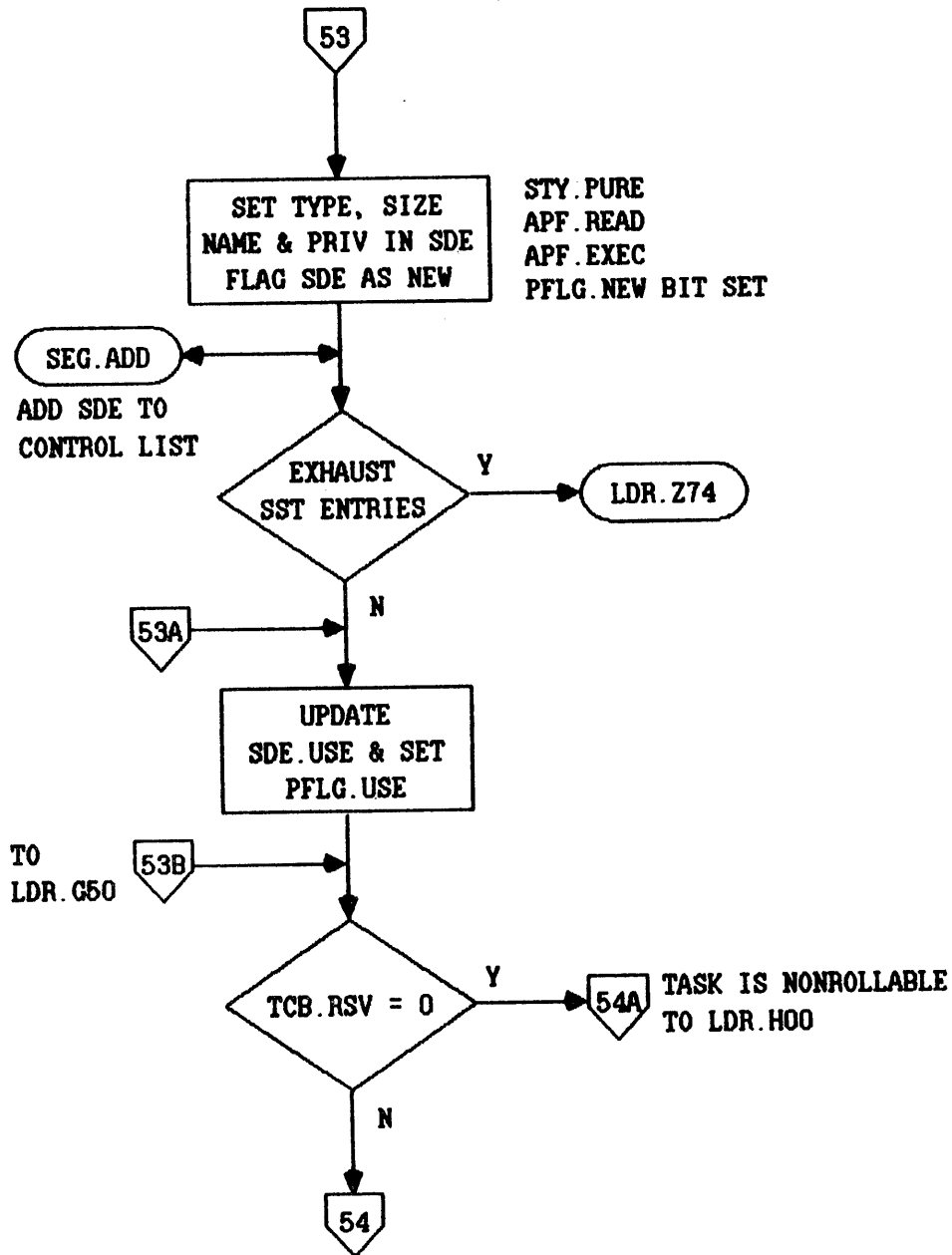


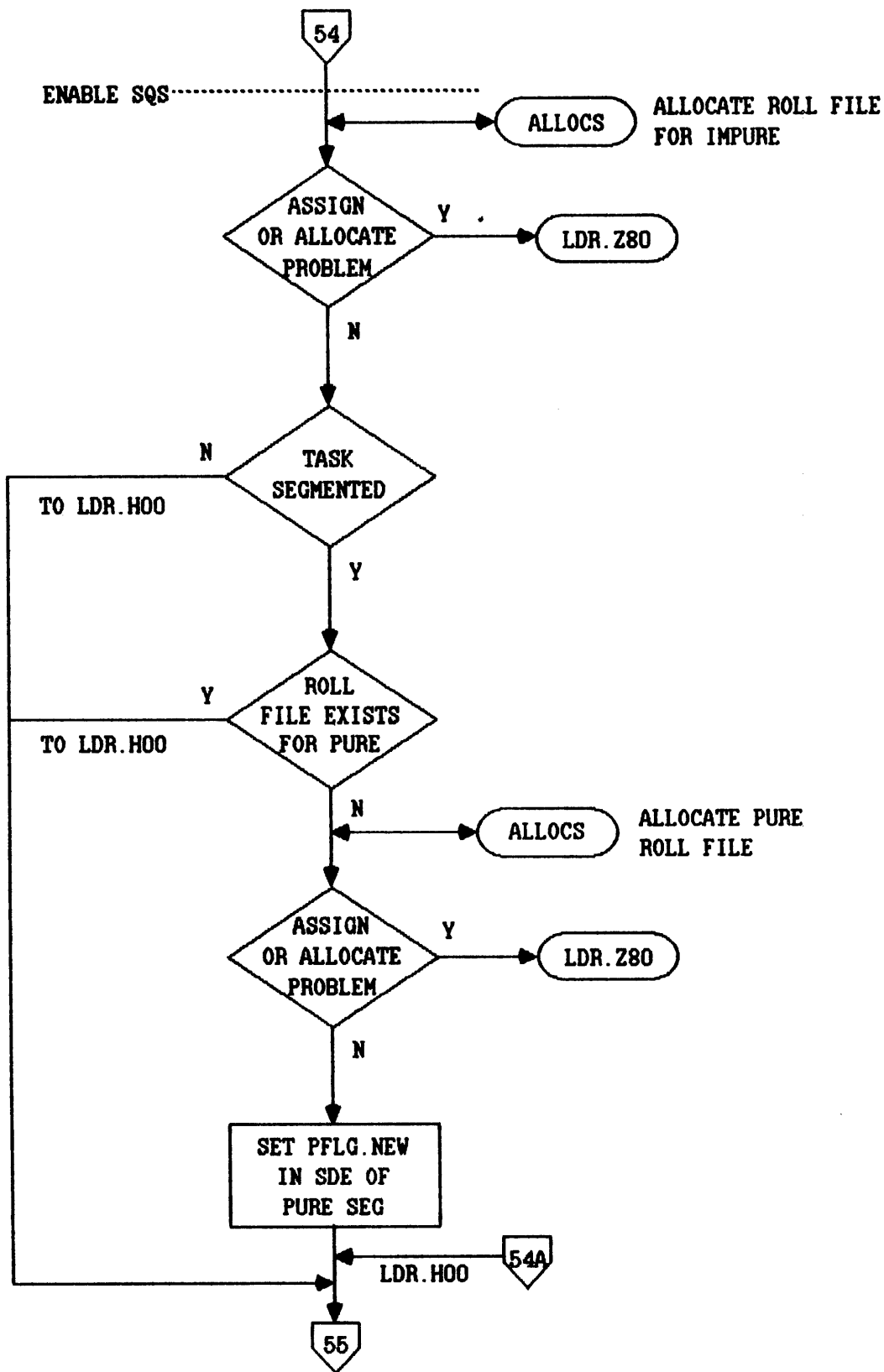


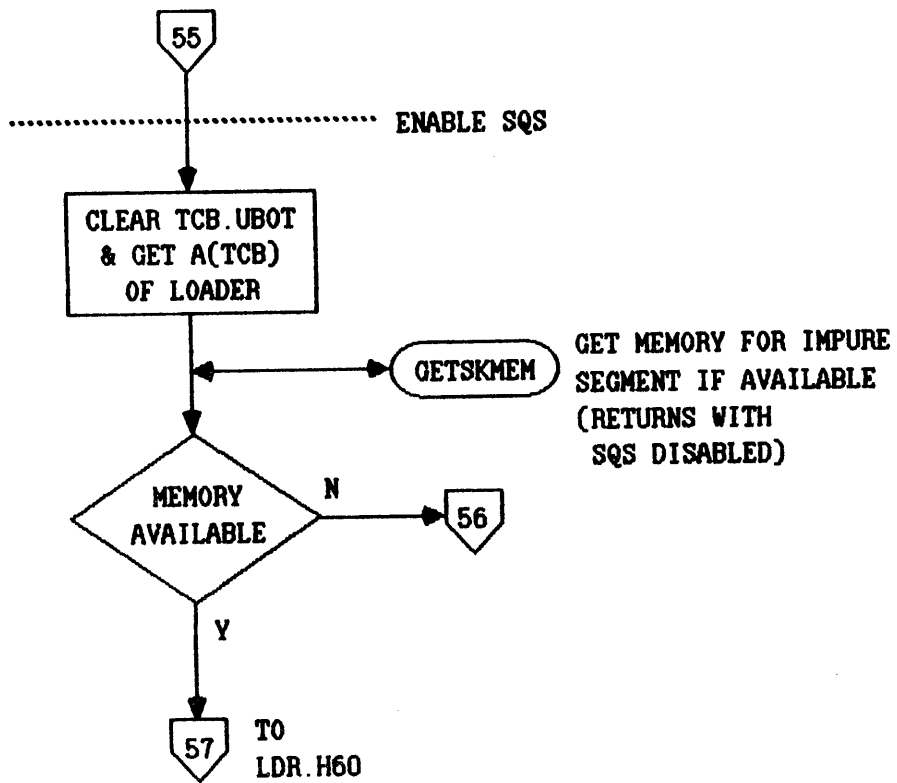


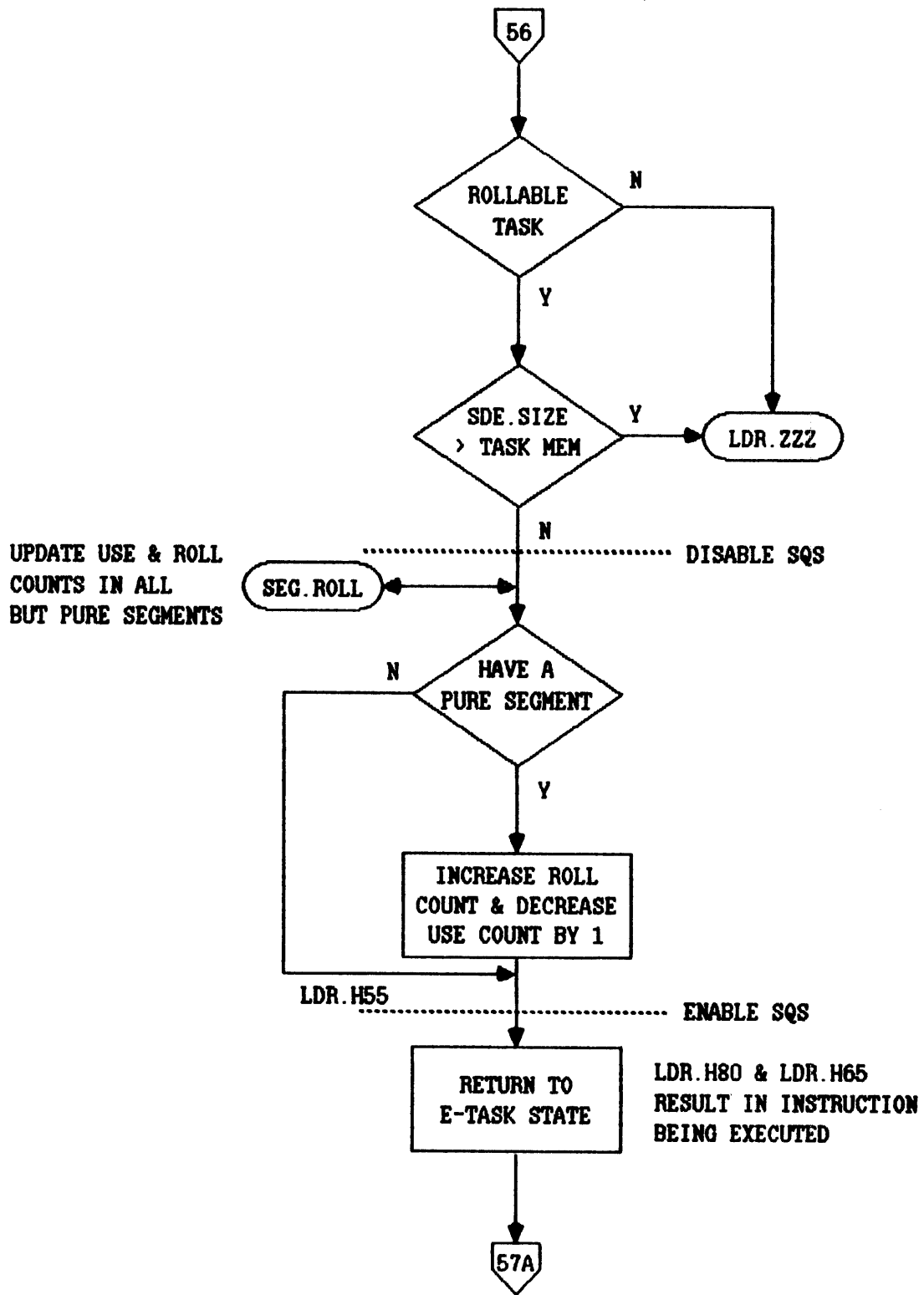


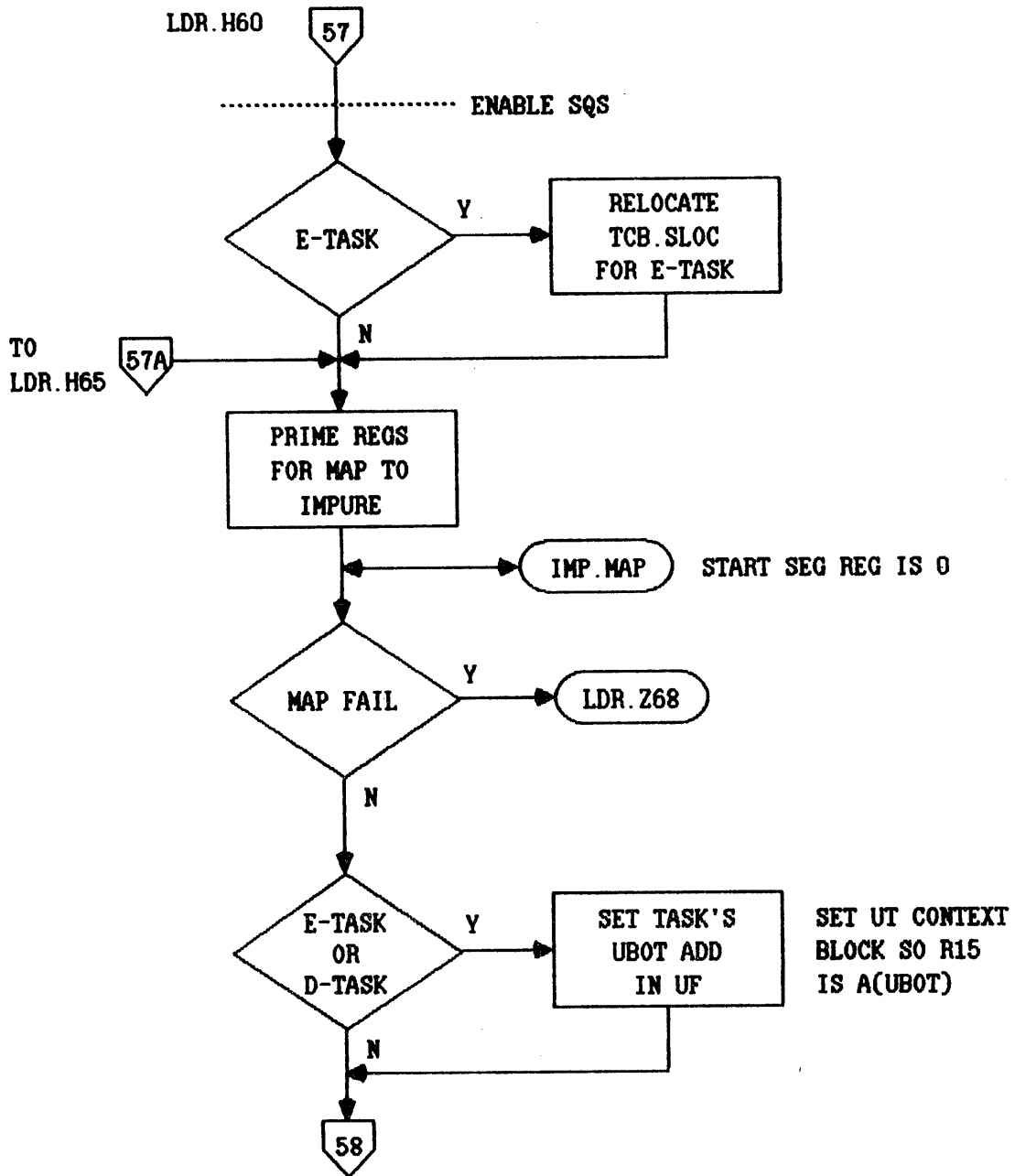


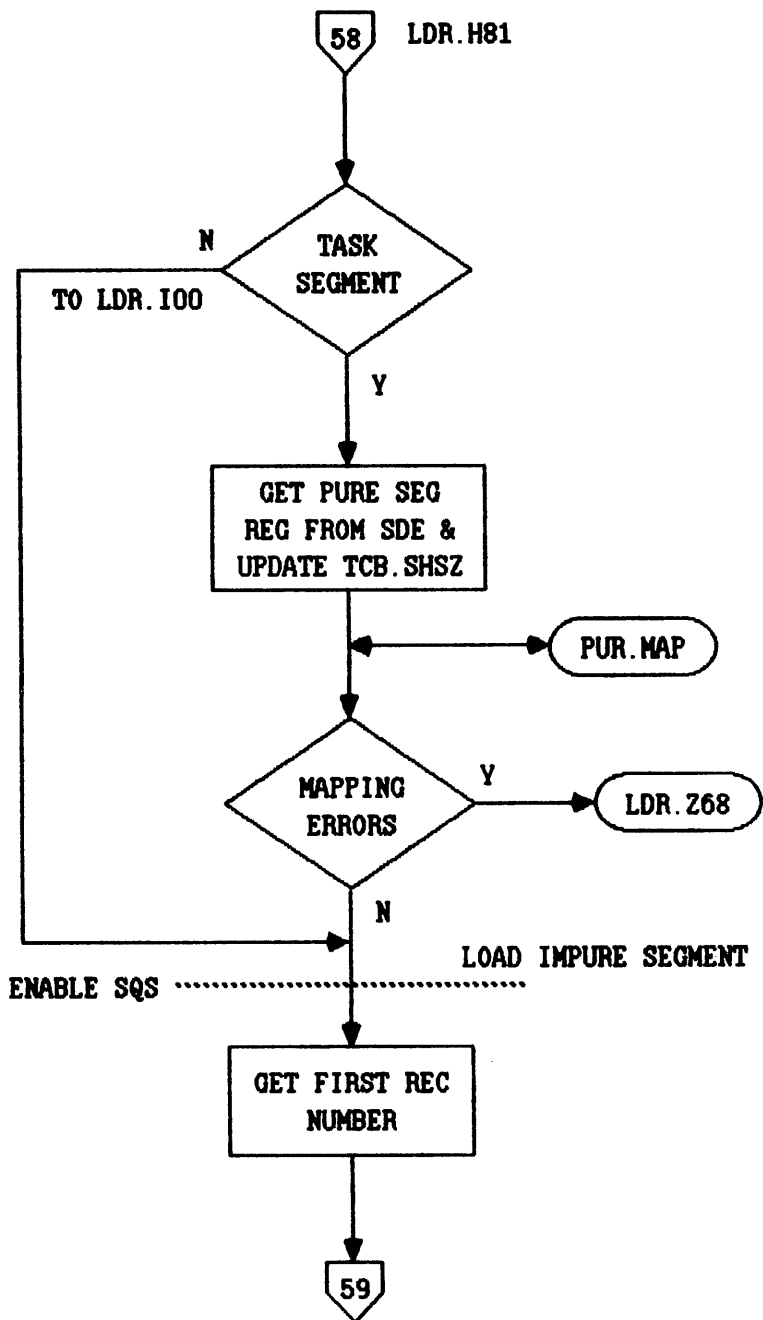


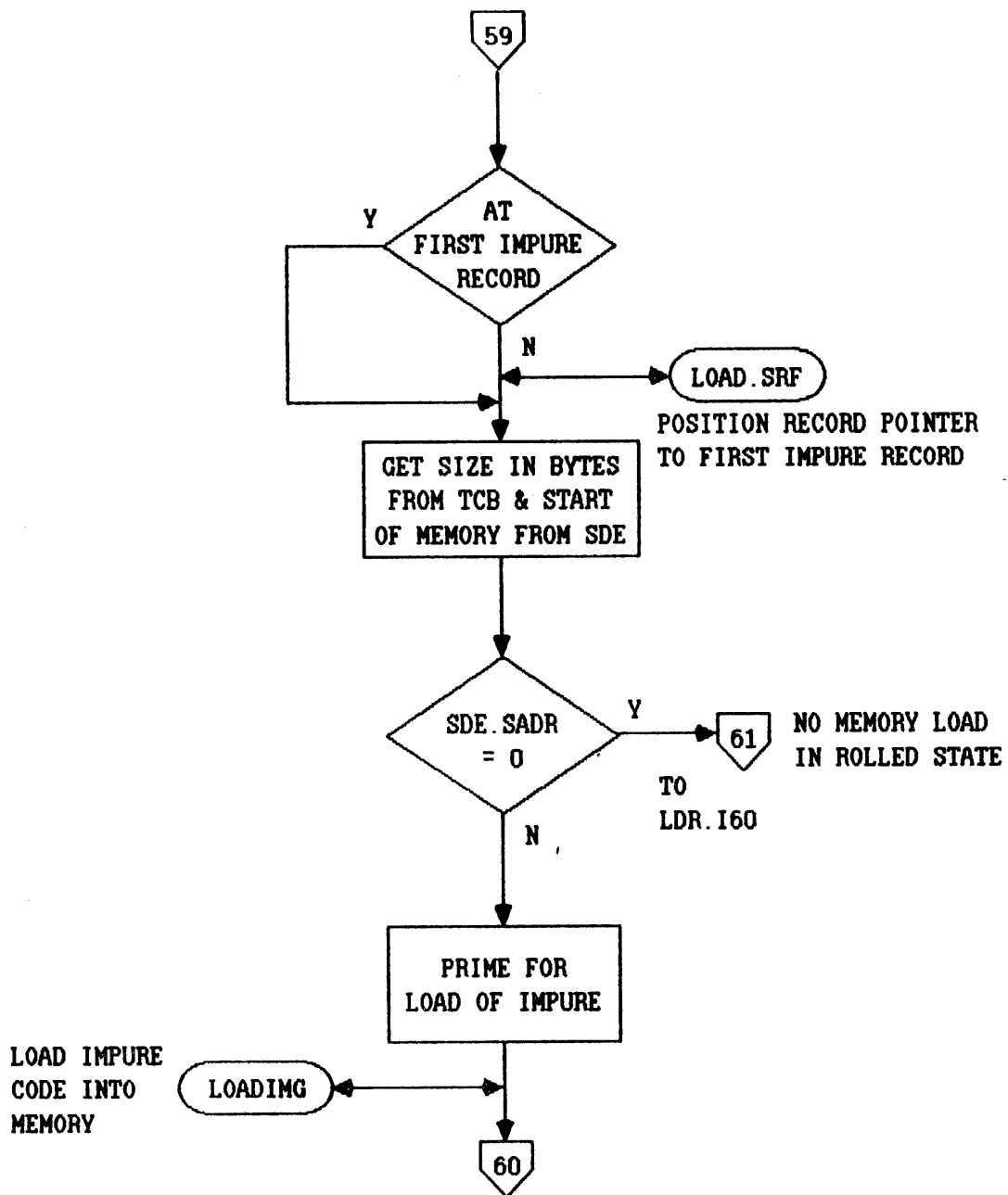


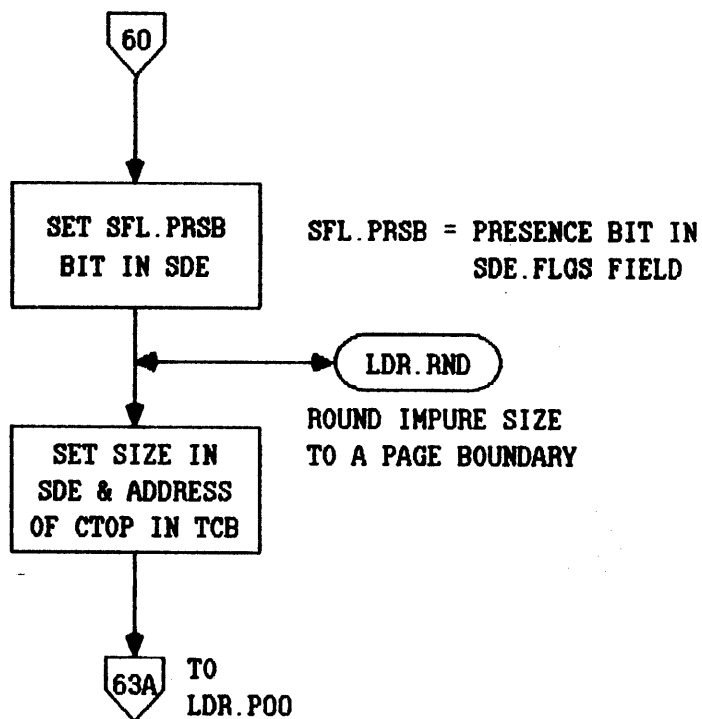






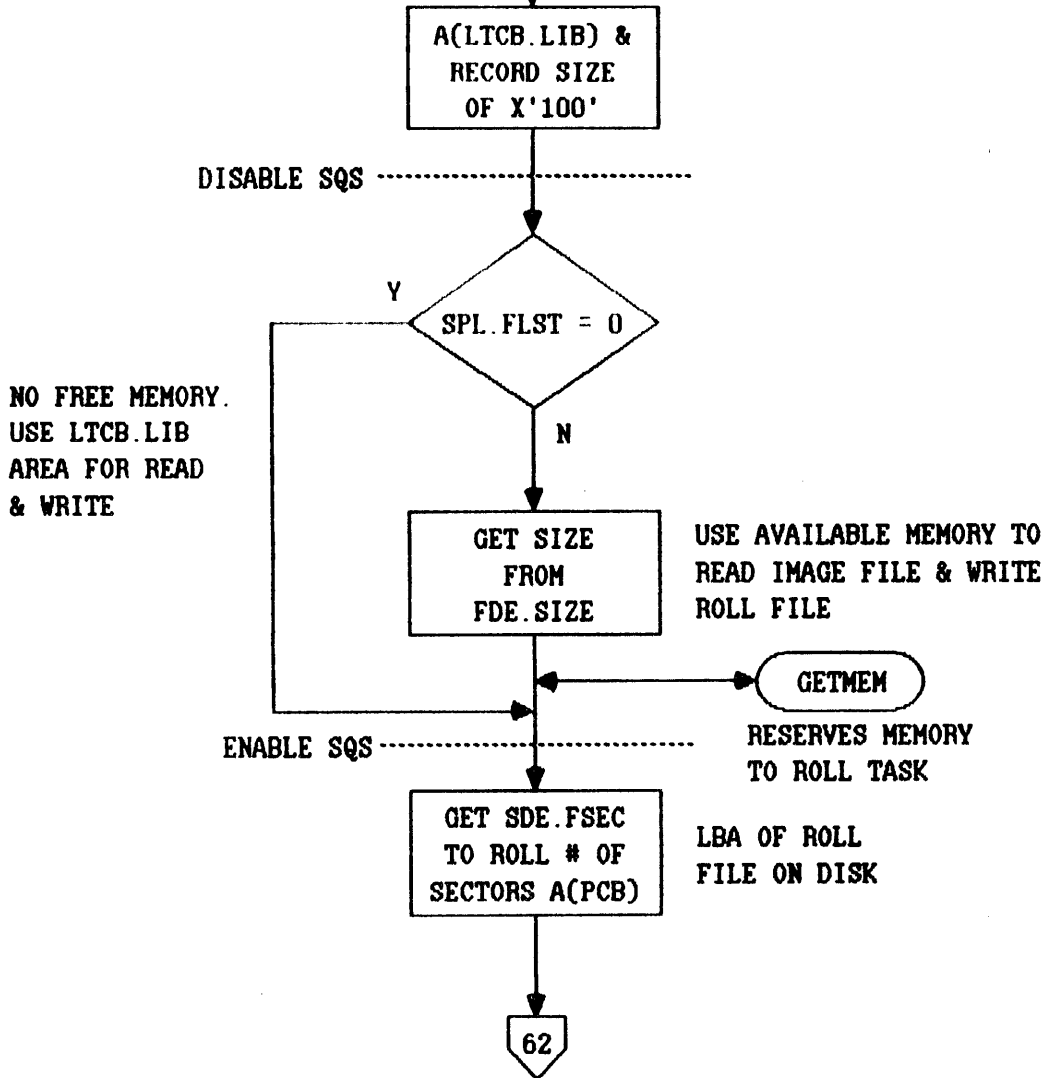


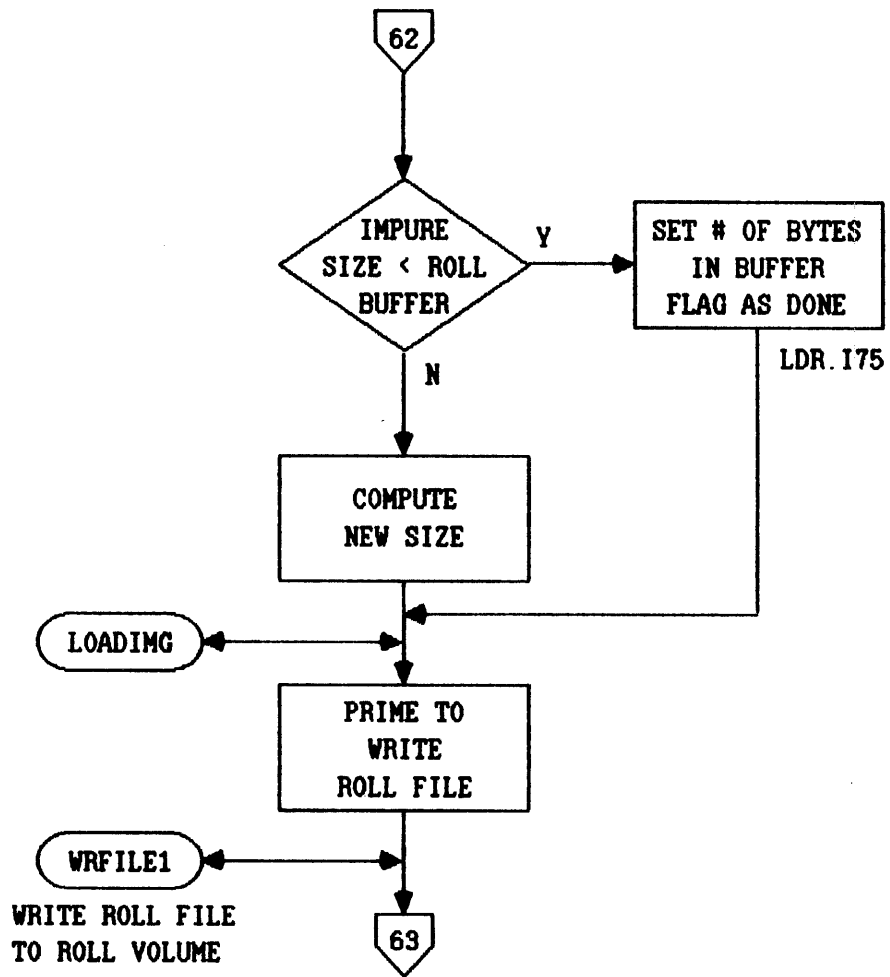


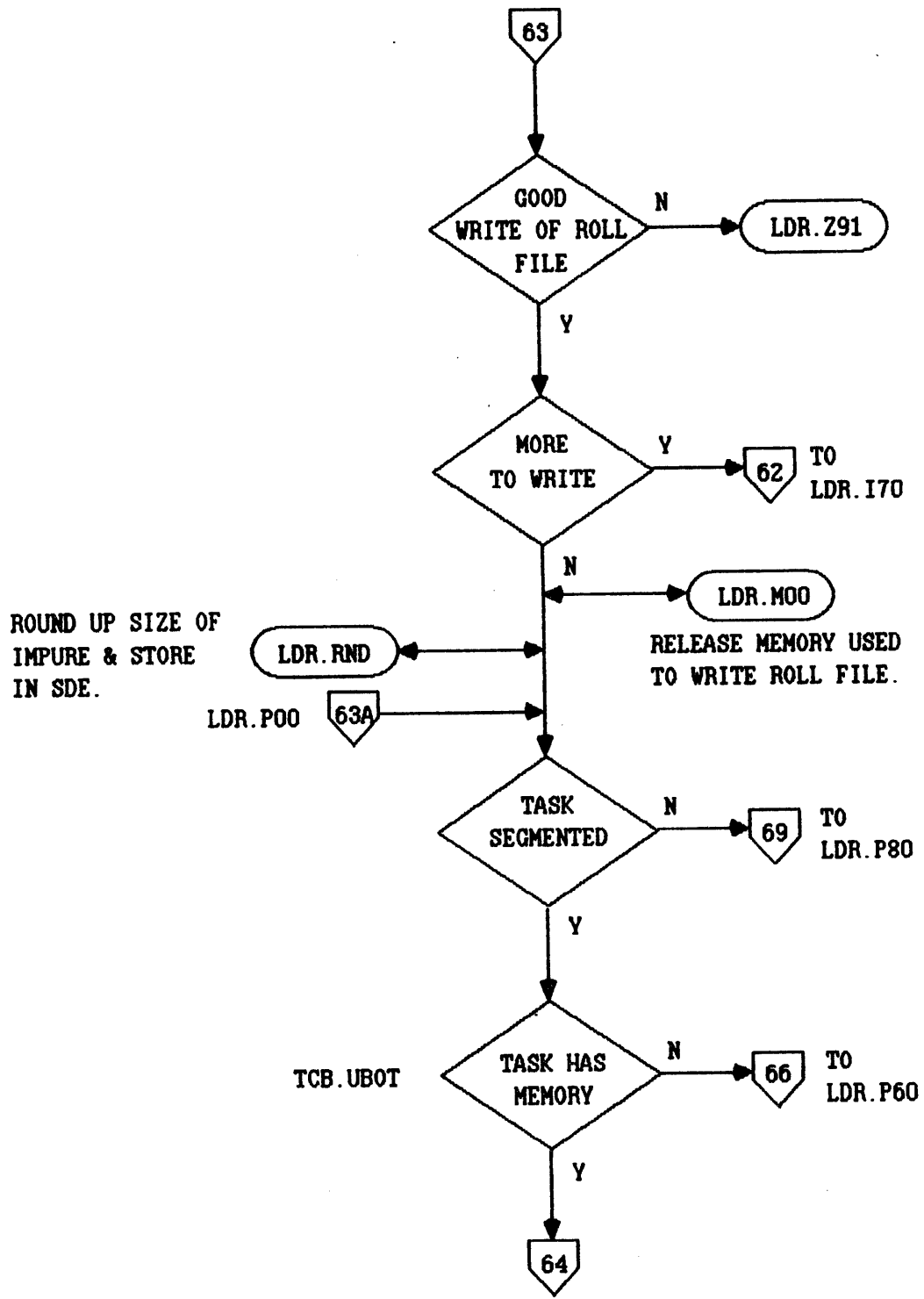


LOAD IMPURE IN ROLLED STATE

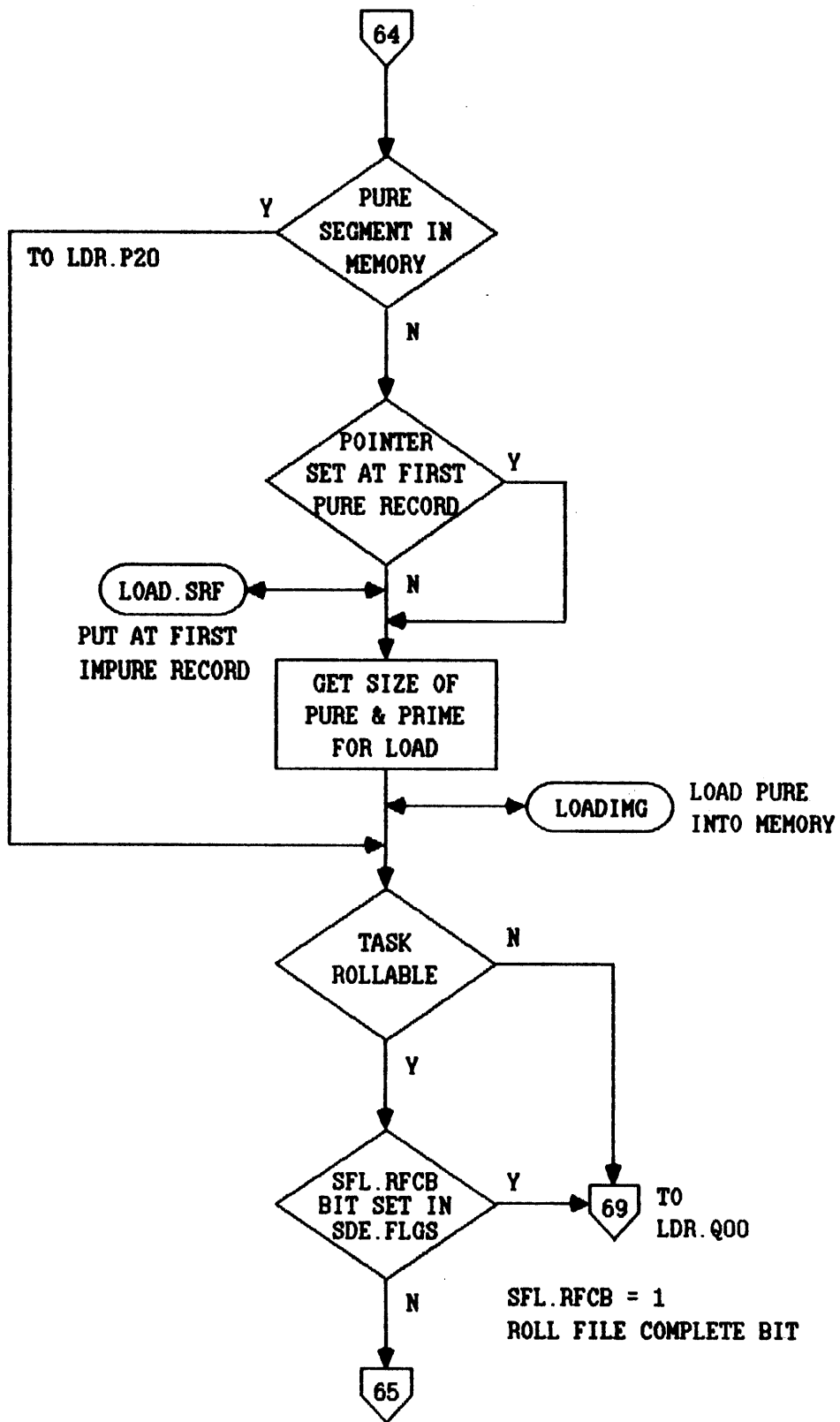
61 LDR. I60

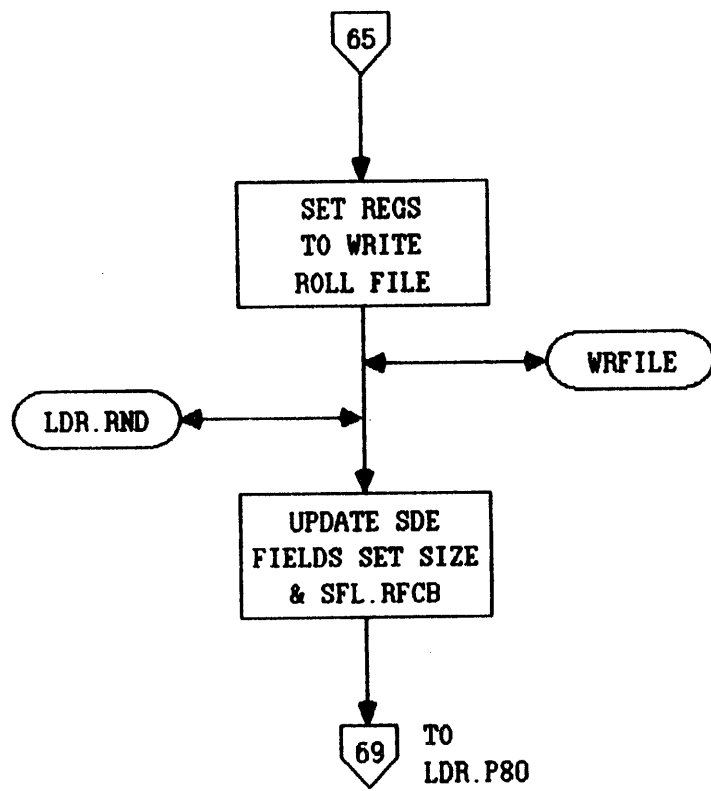


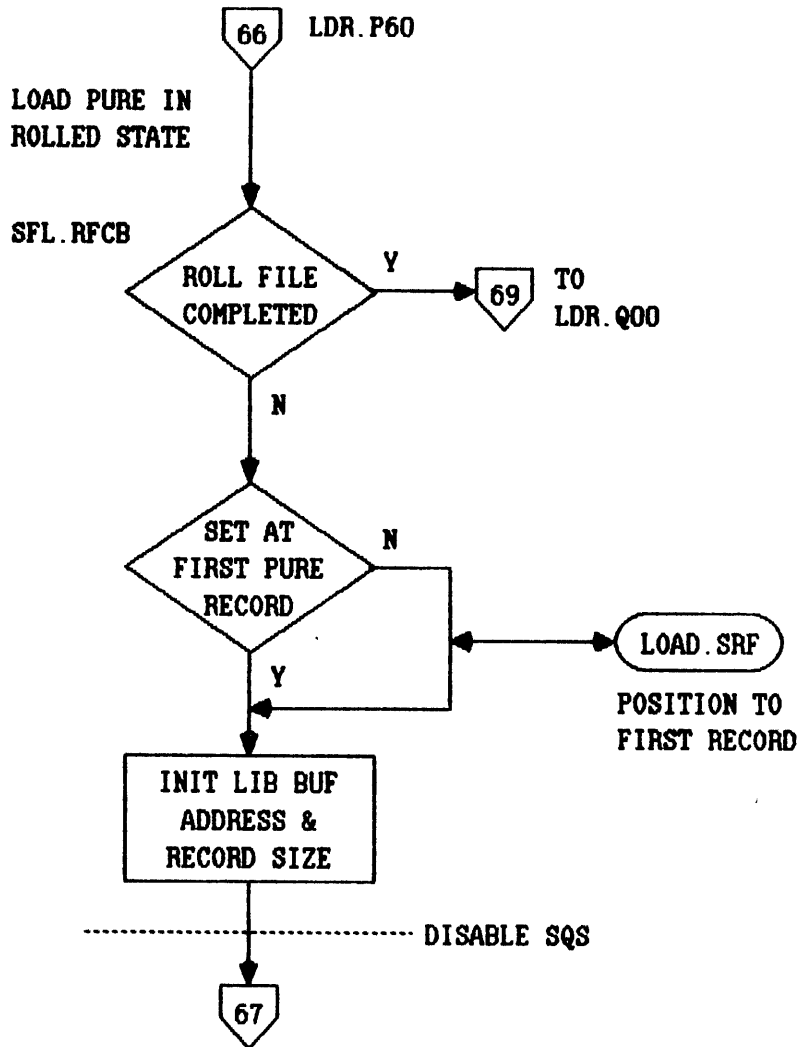


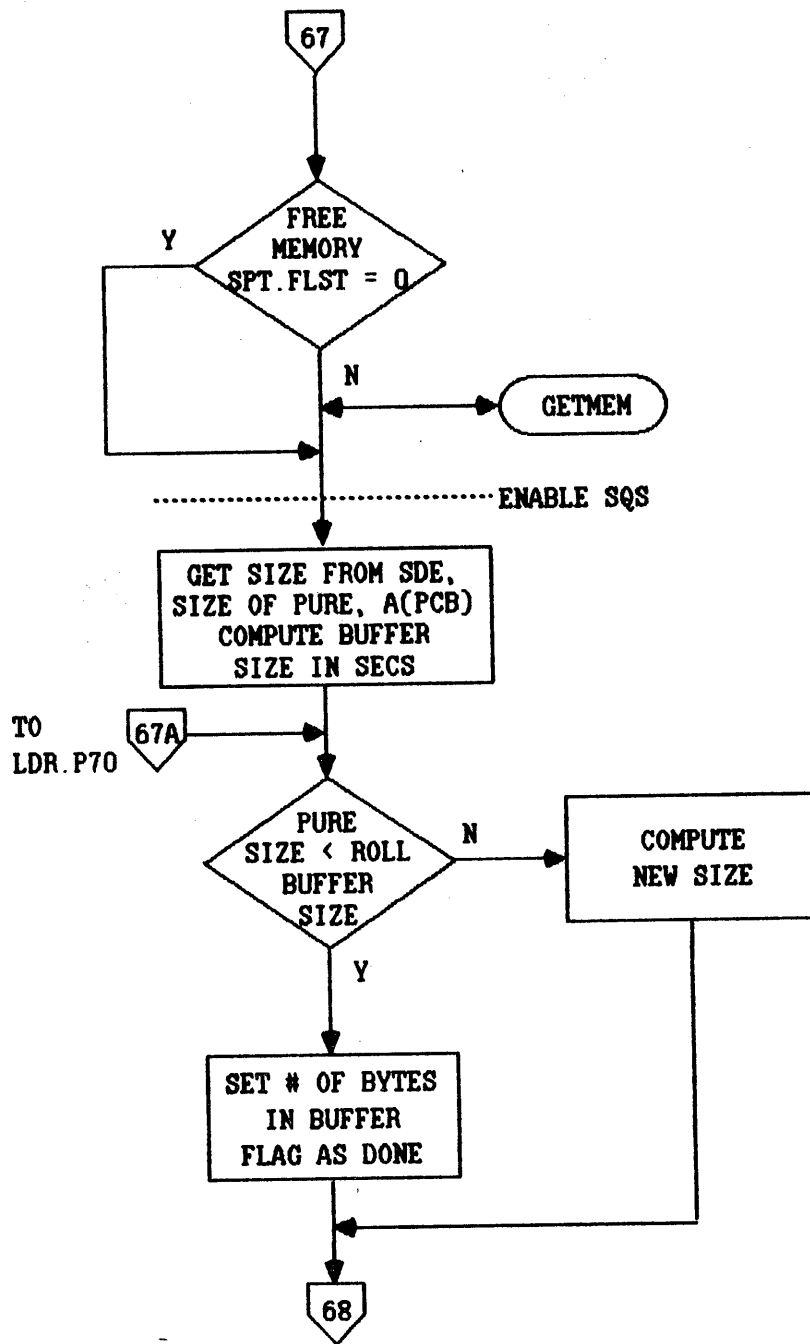


ROUND UP SIZE OF IMPURE & STORE IN SDE.

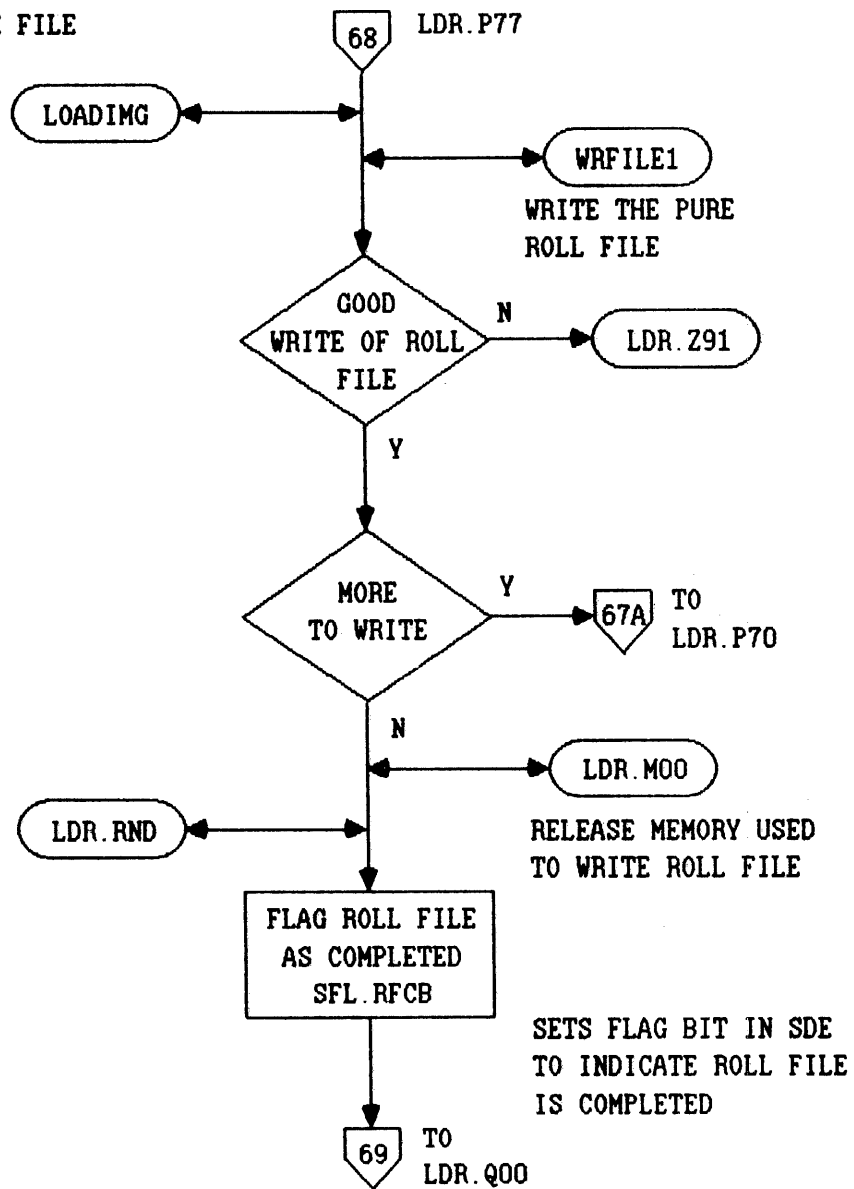


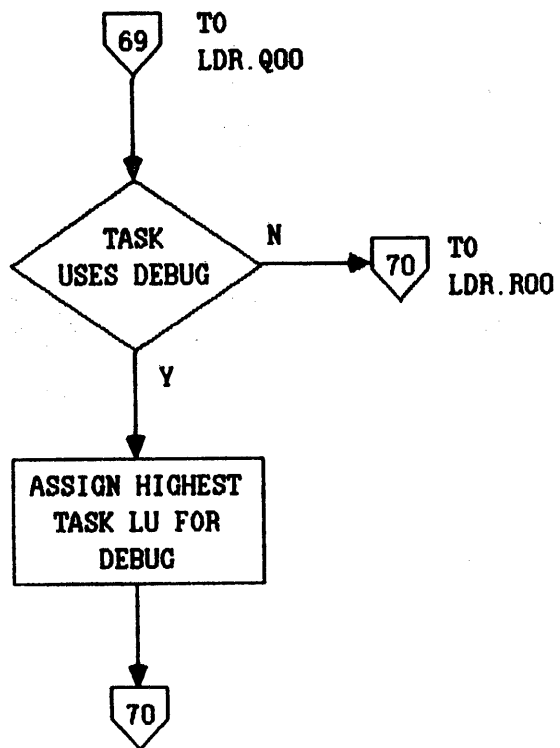


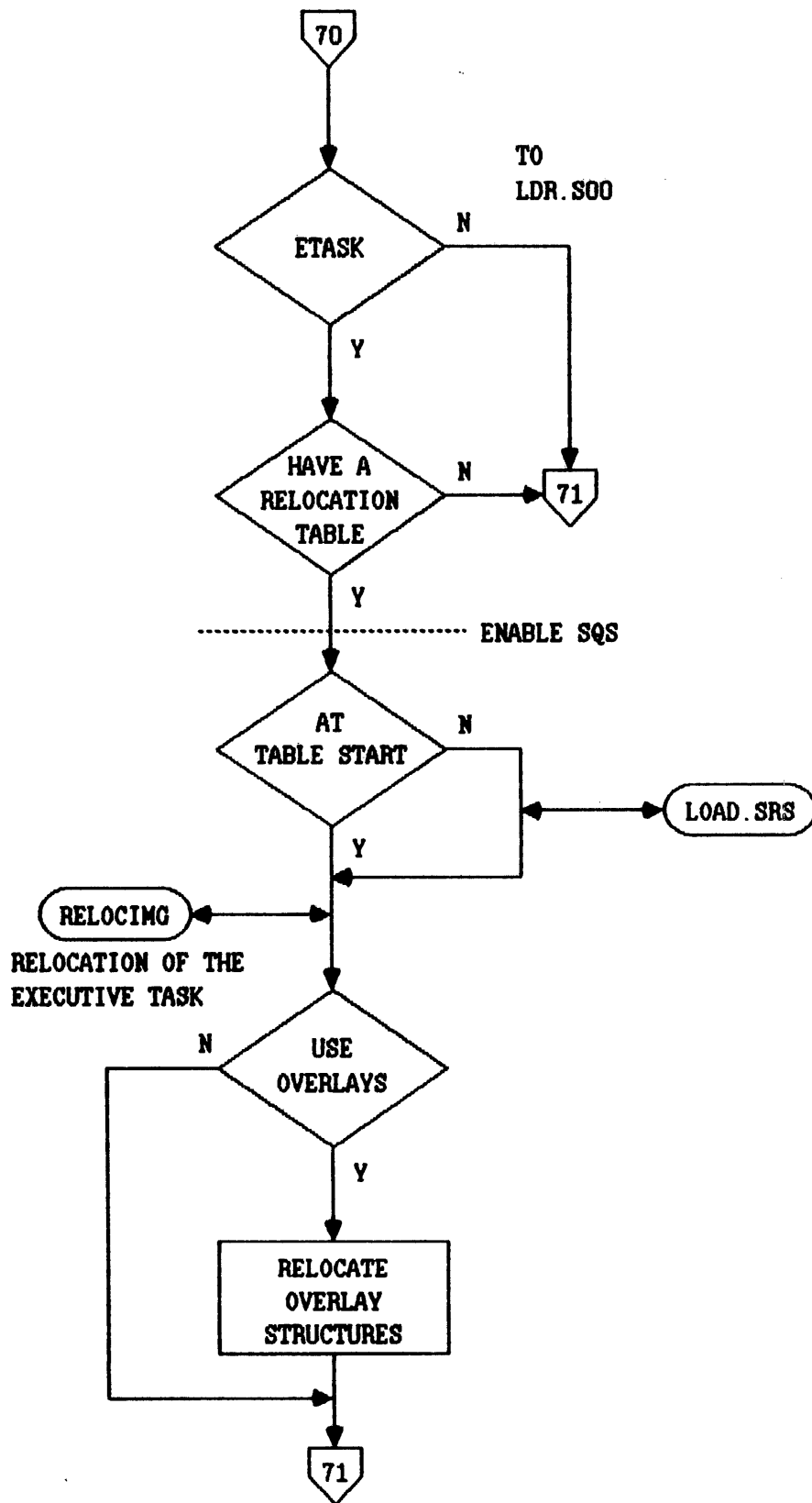


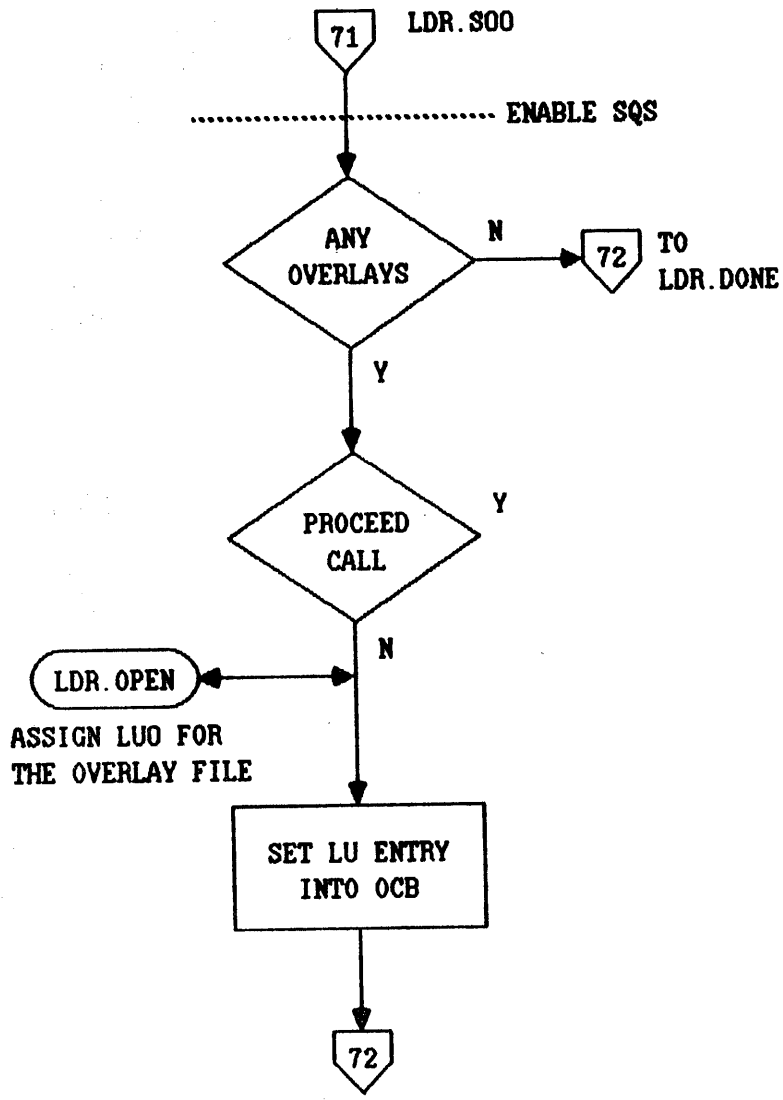


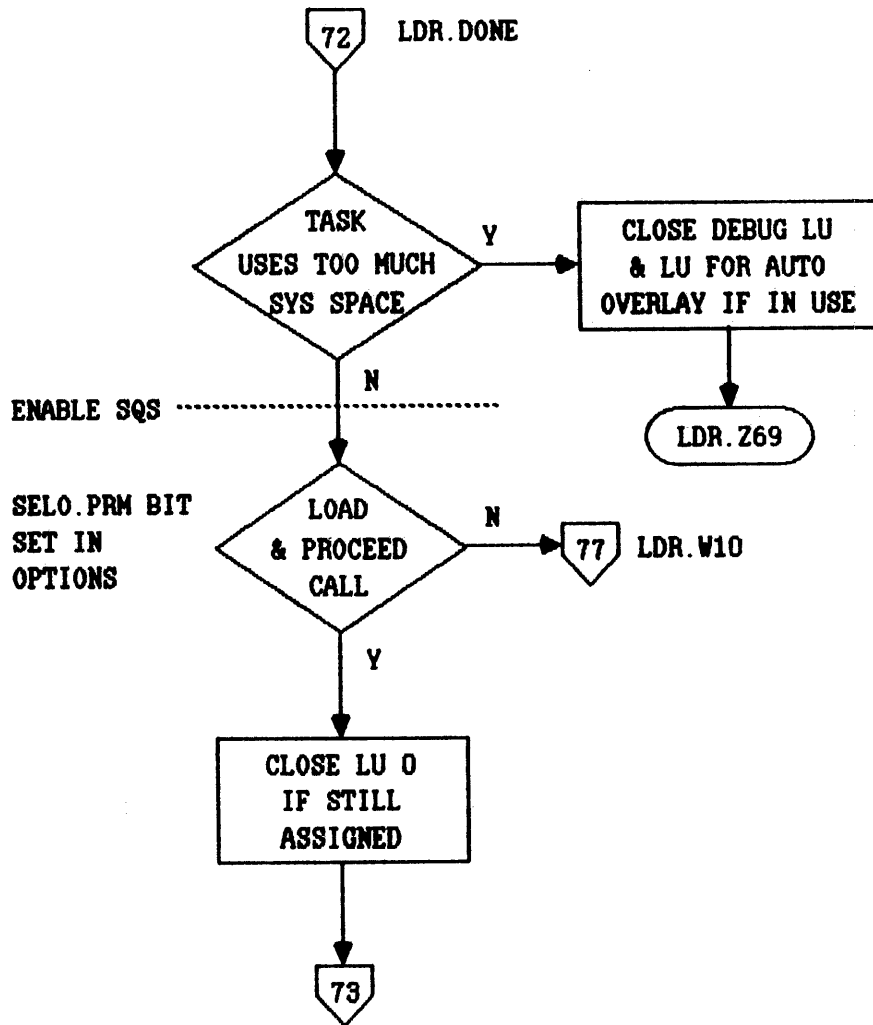
READ RECORDS INTO MEMORY
FROM IMAGE FILE

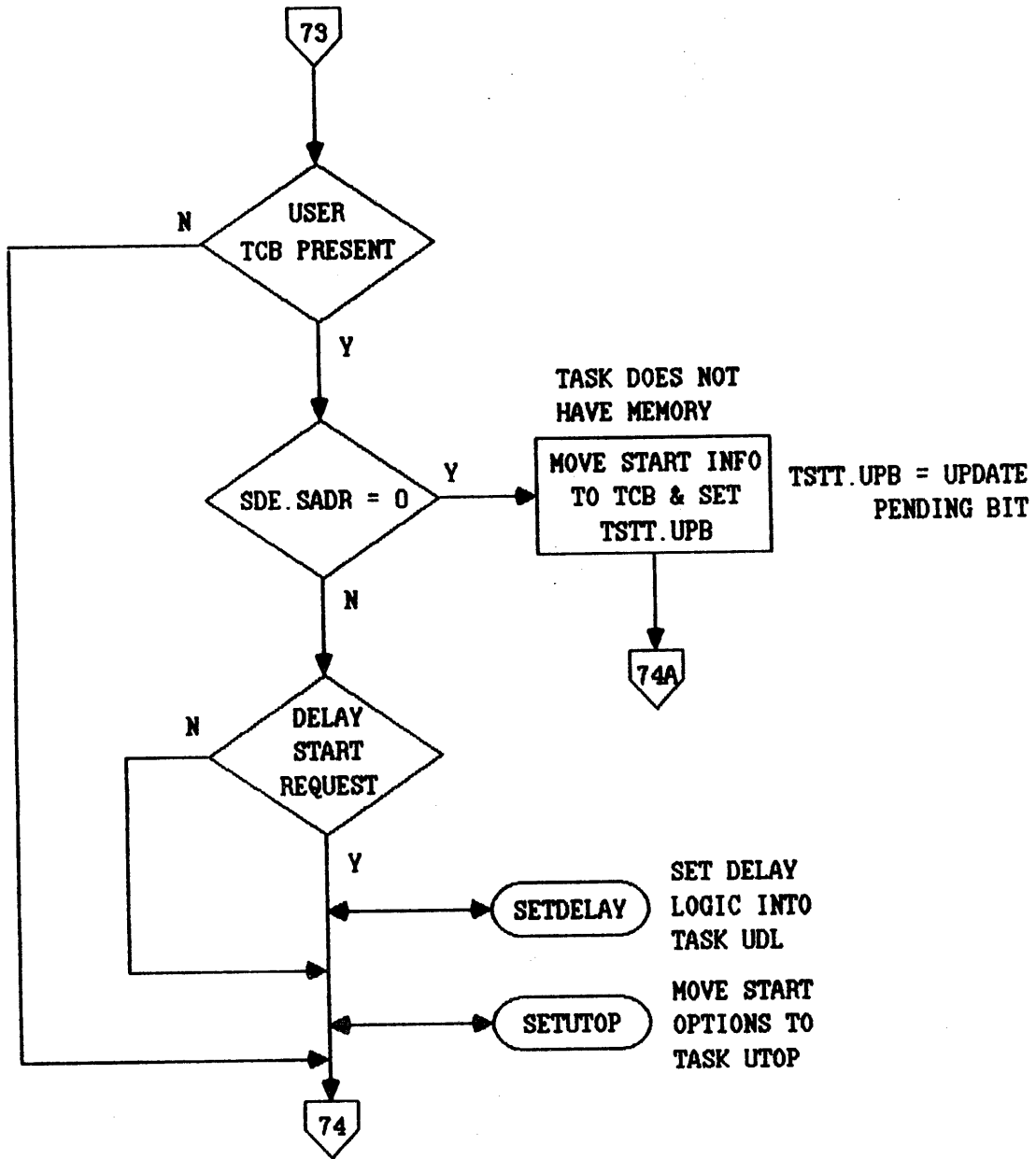


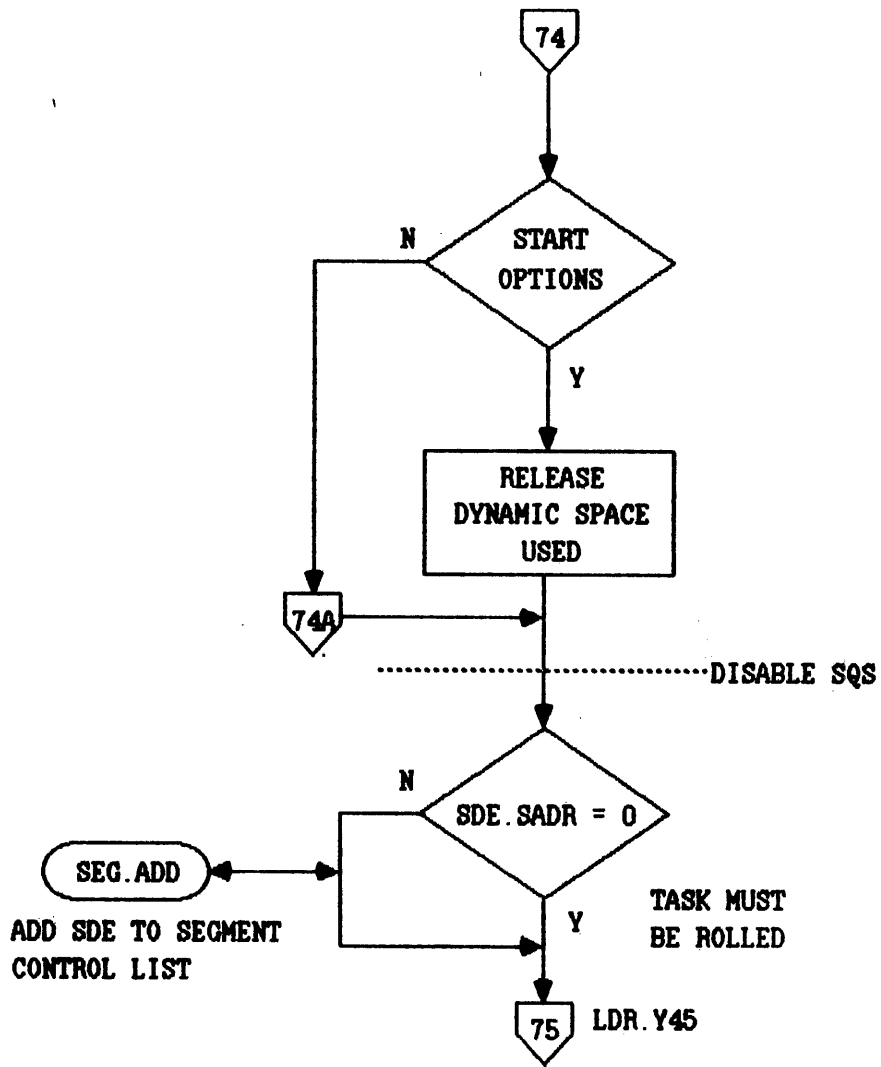


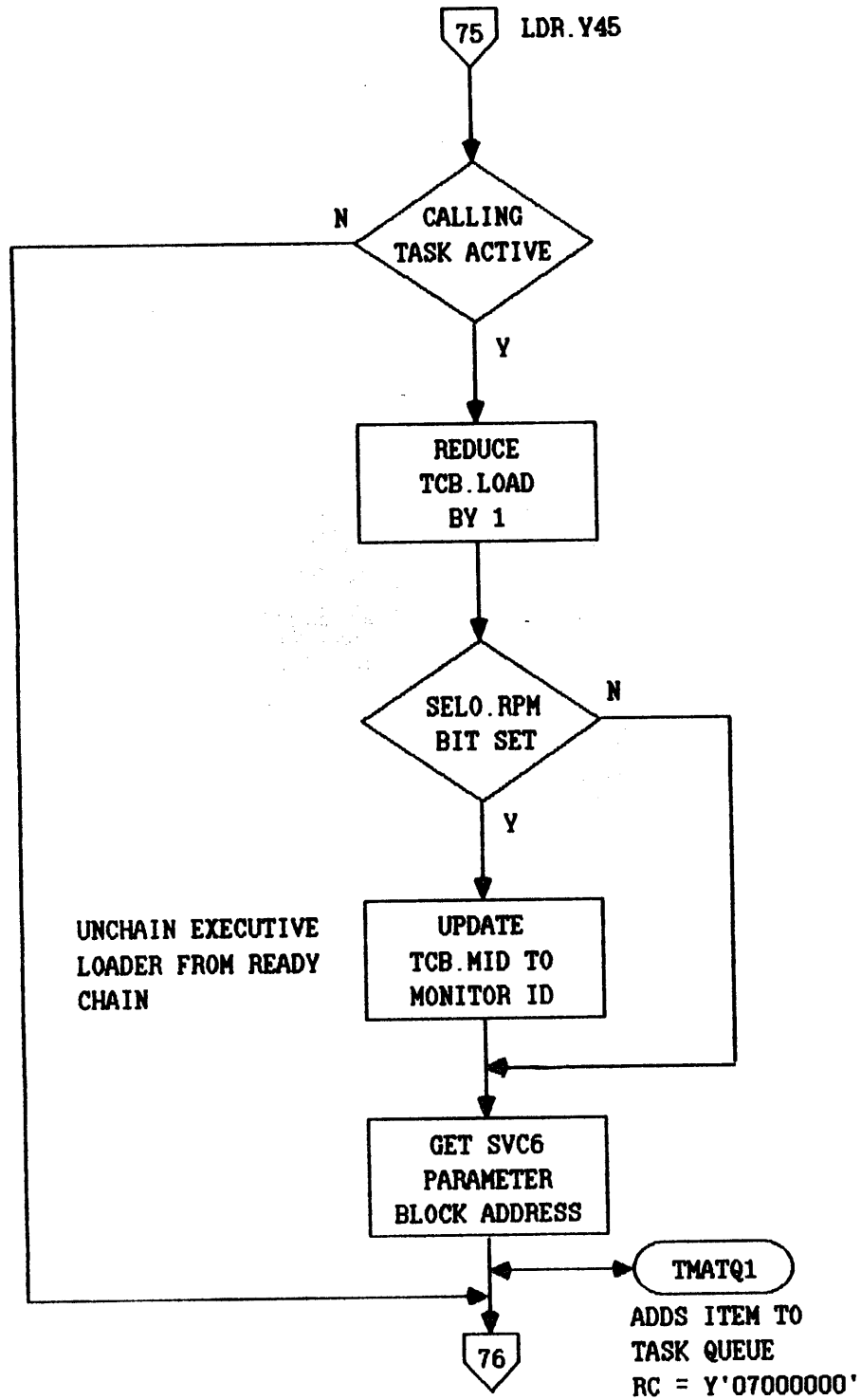


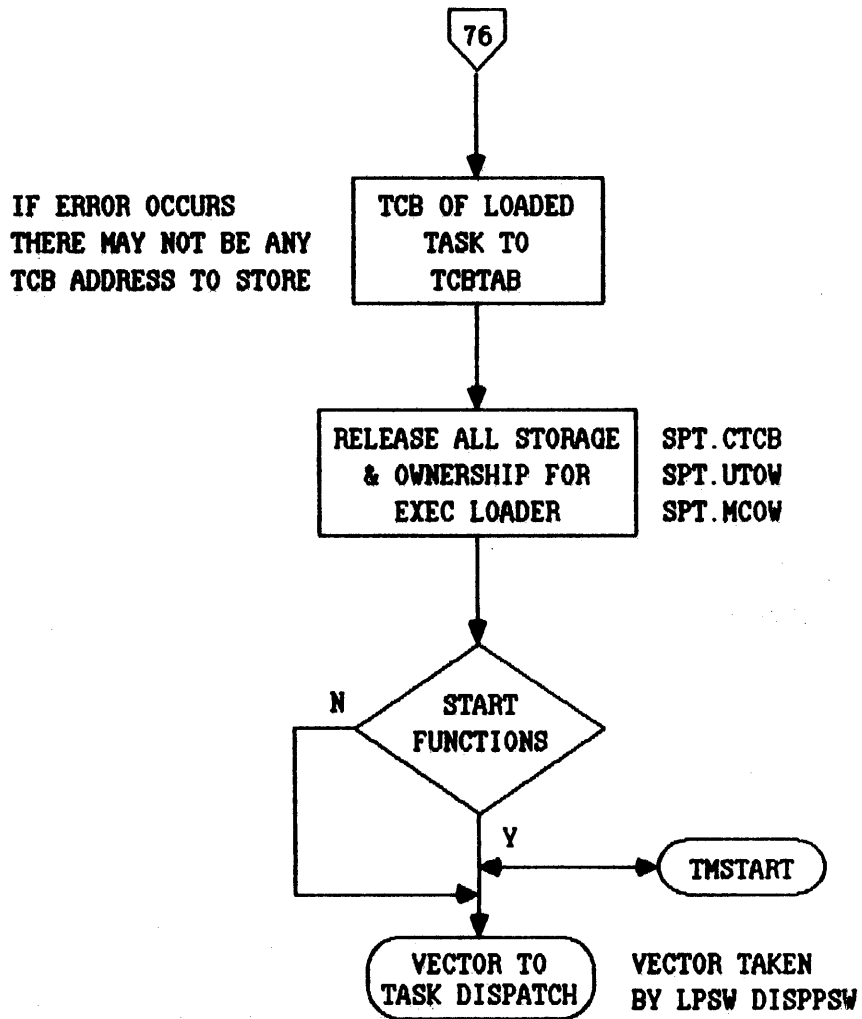










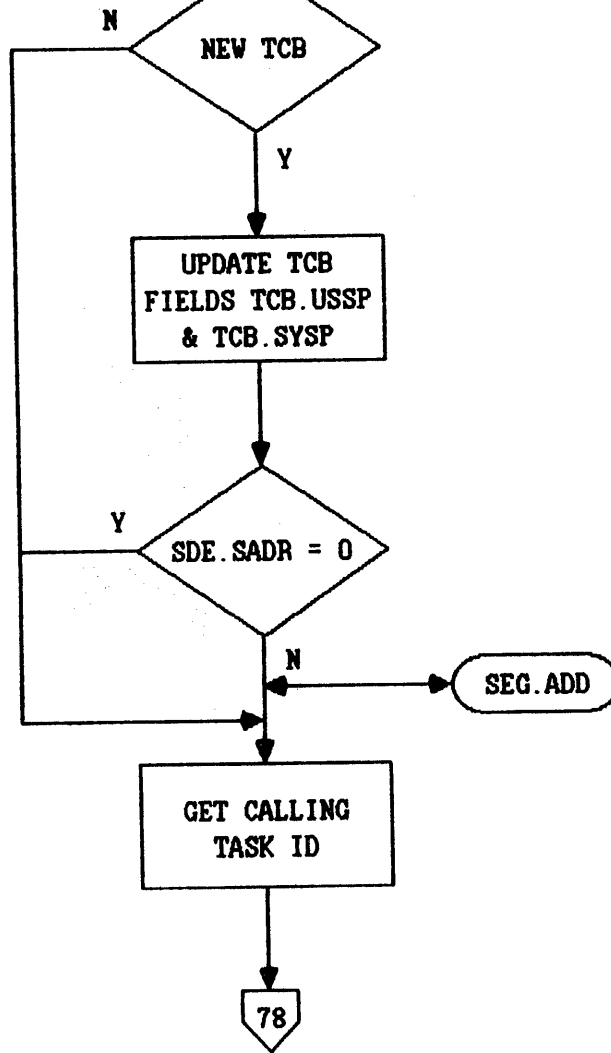


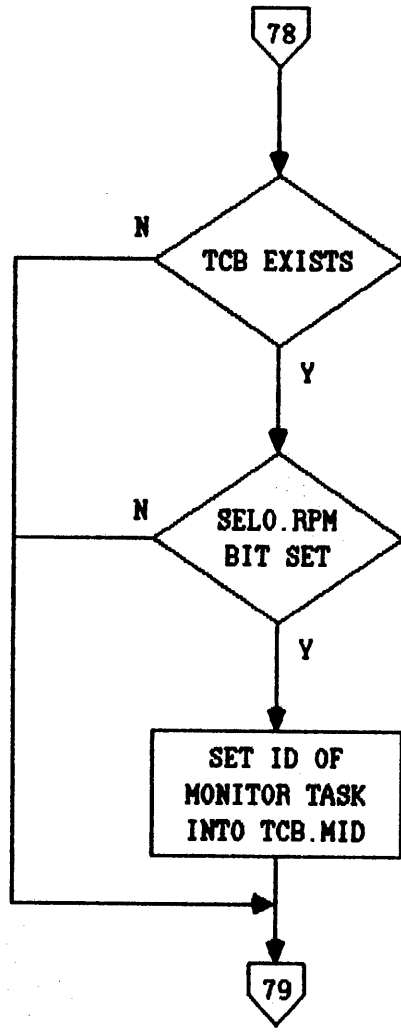
LOAD WAIT SUCCESS
OR FAILURE

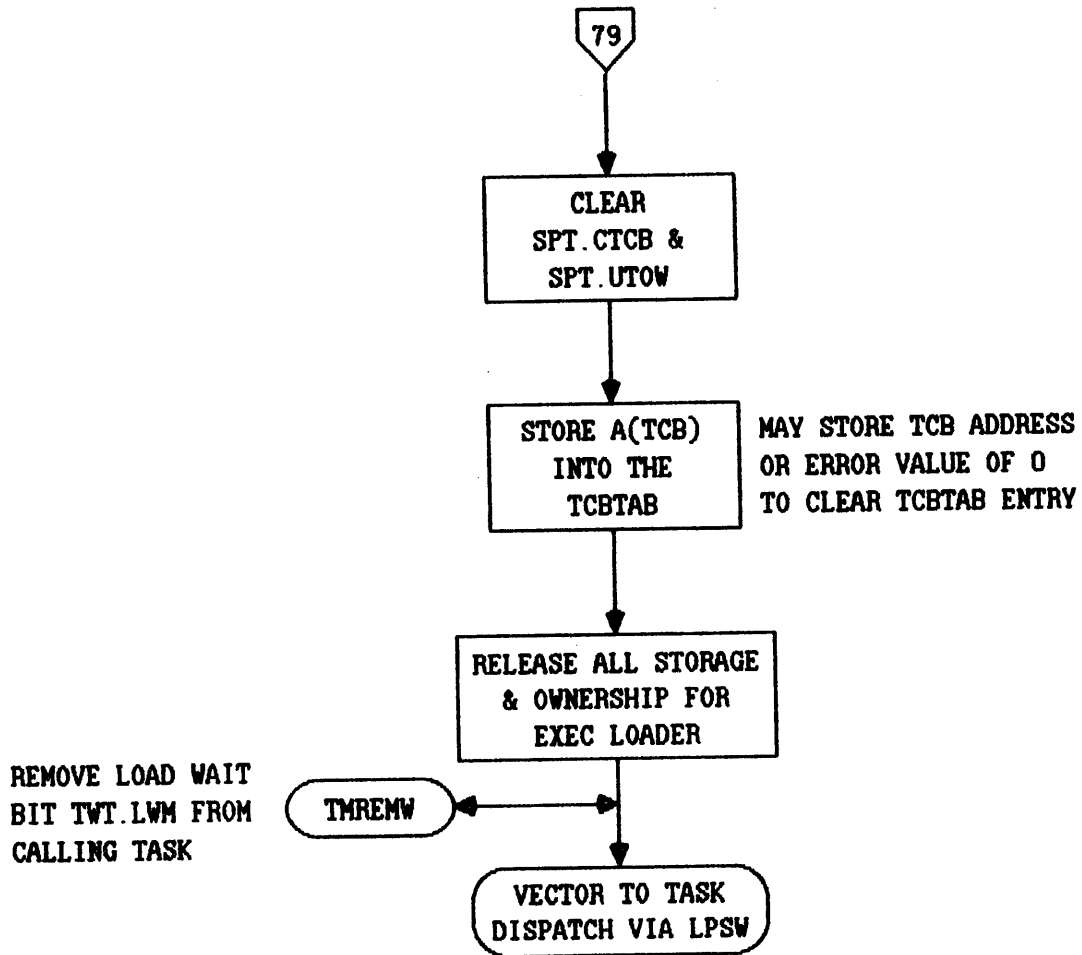
77

LDR.W10

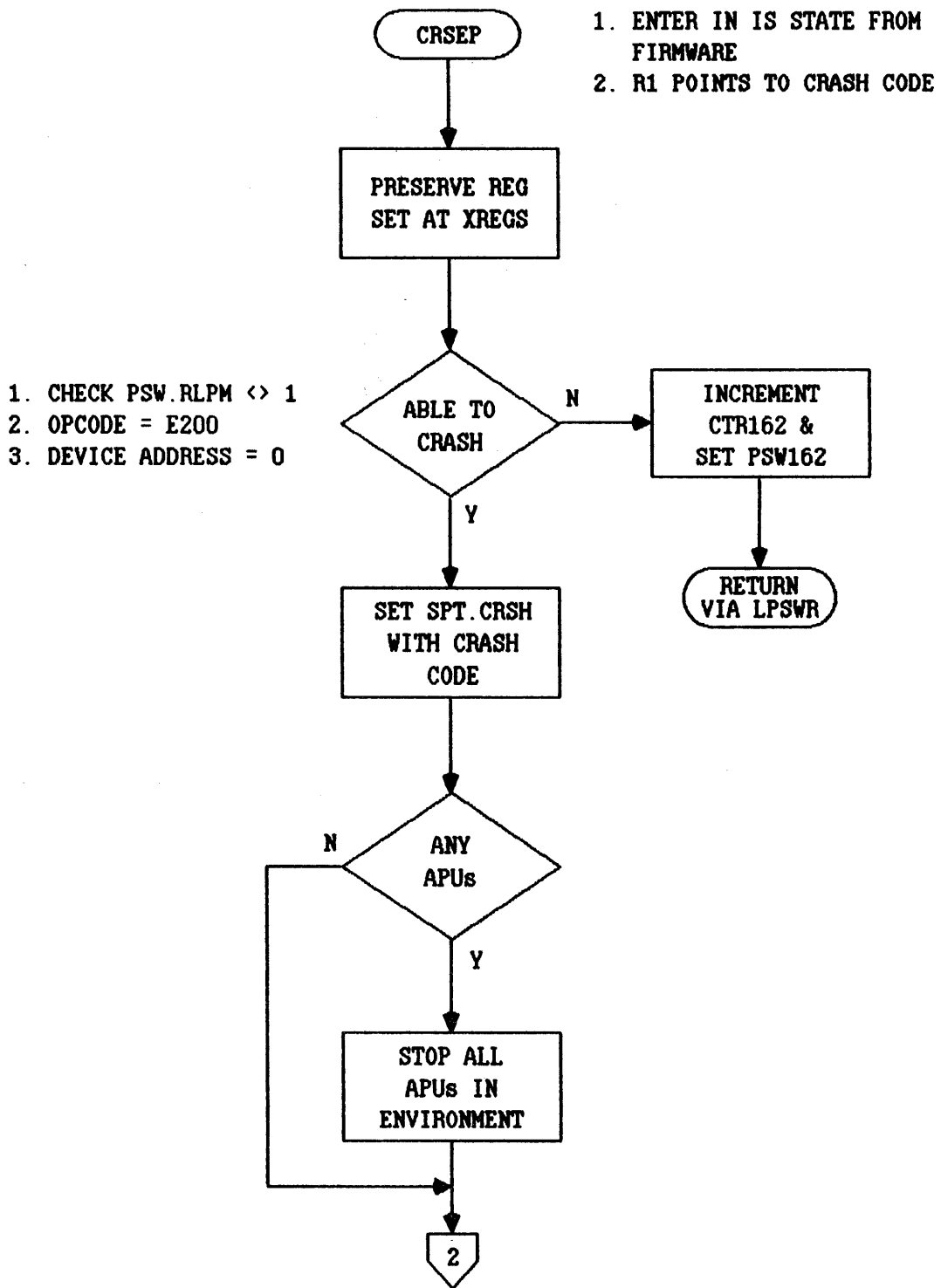
ENTER NSU
DISABLE SQS

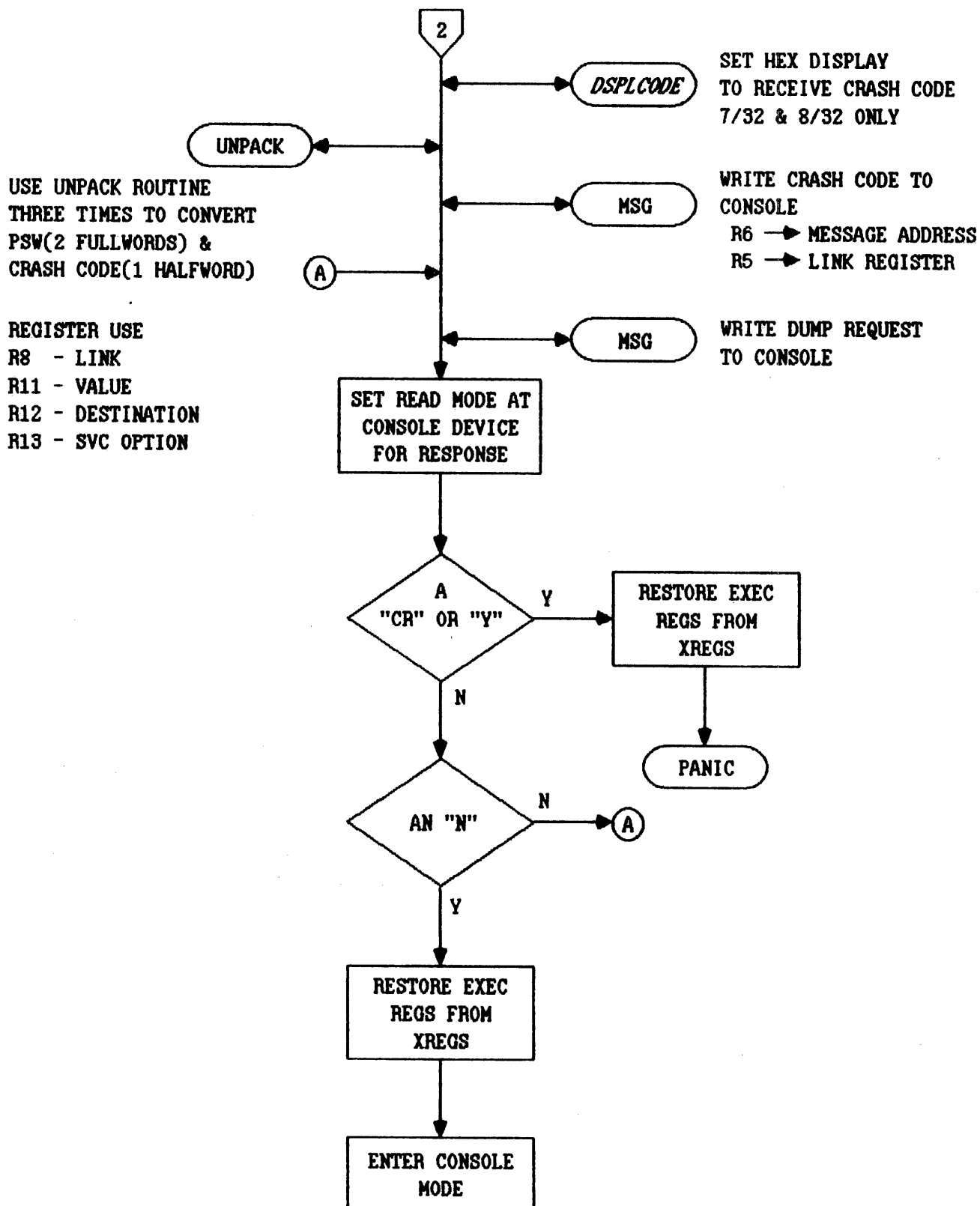






CRASH HANDLER

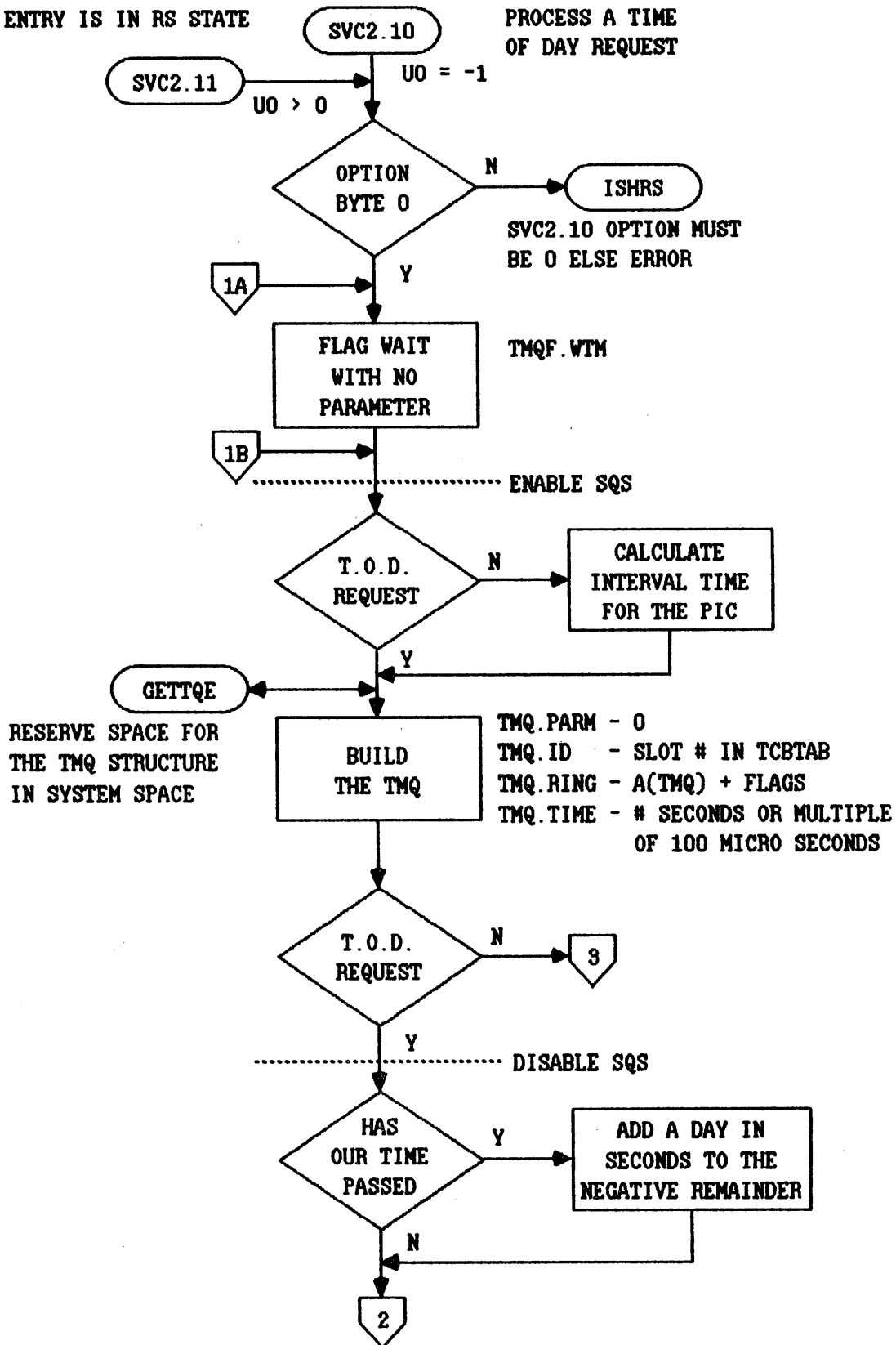


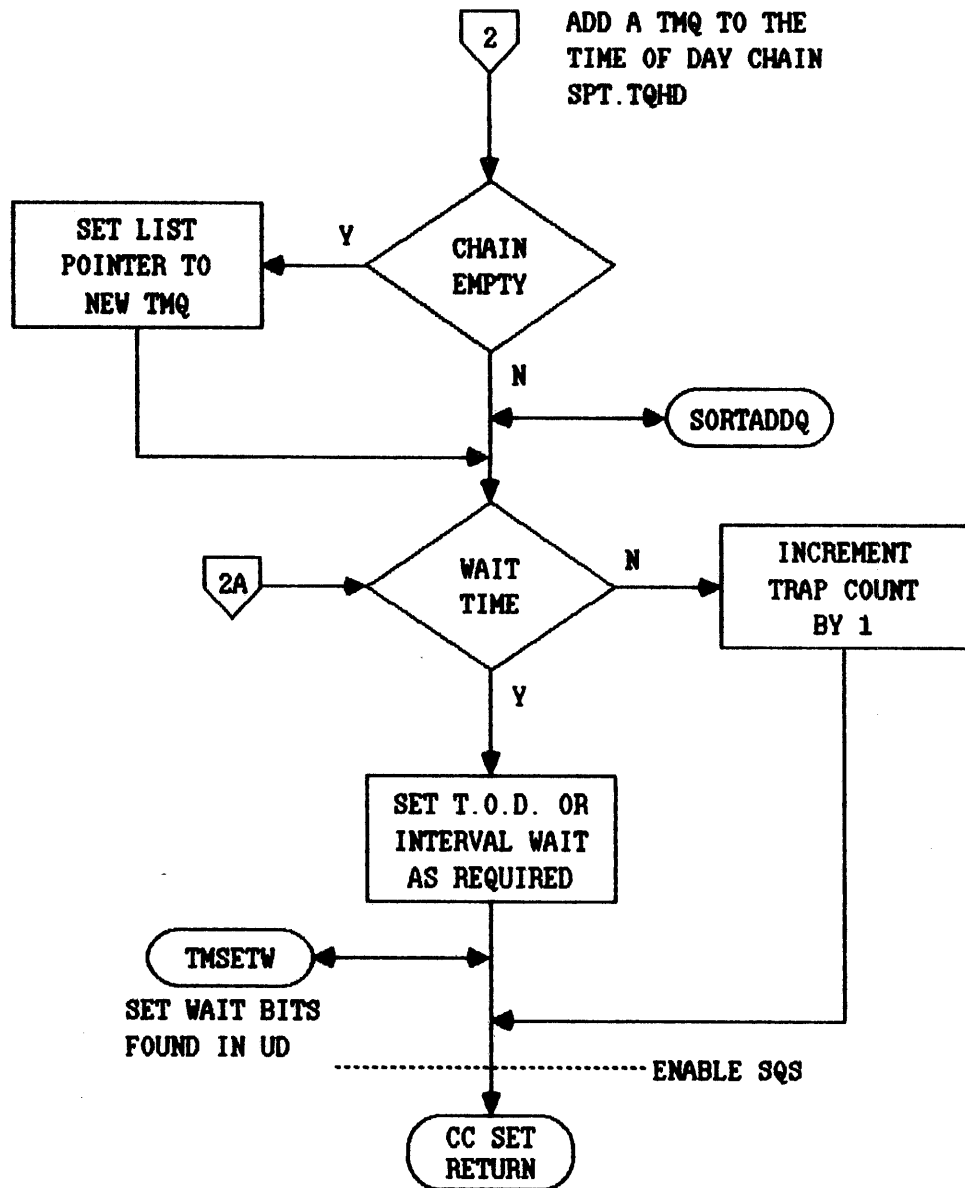


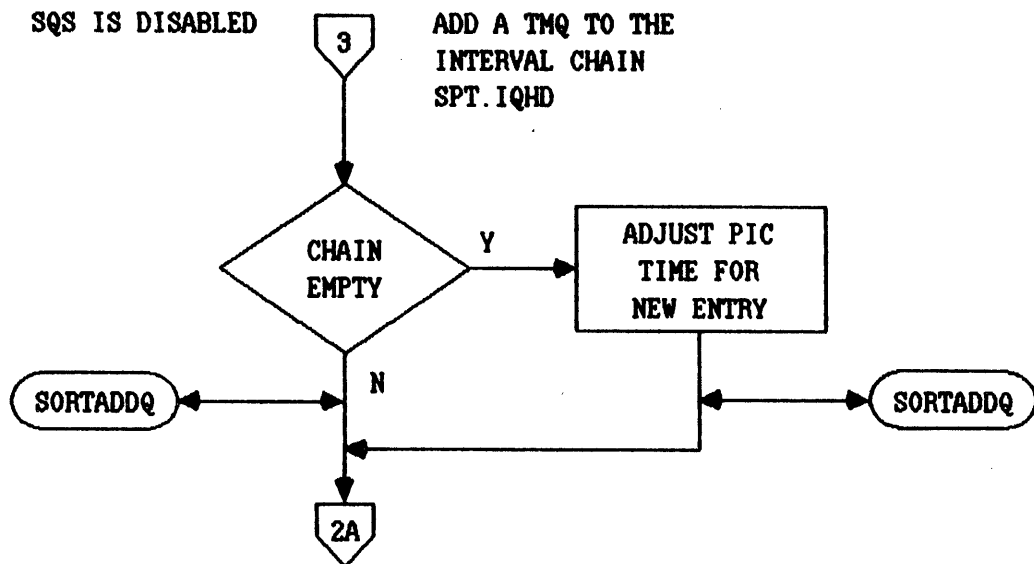
TIME FUNCTIONS

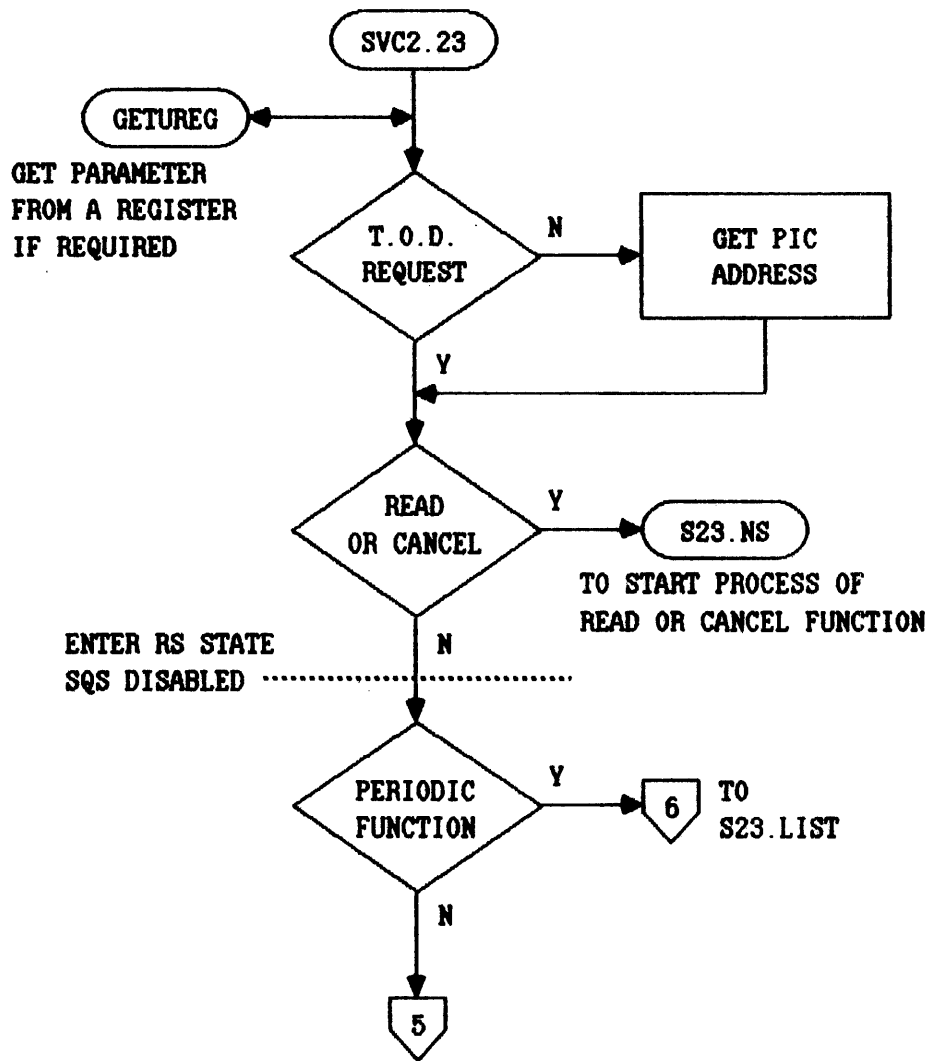
ENTRY IS IN RS STATE

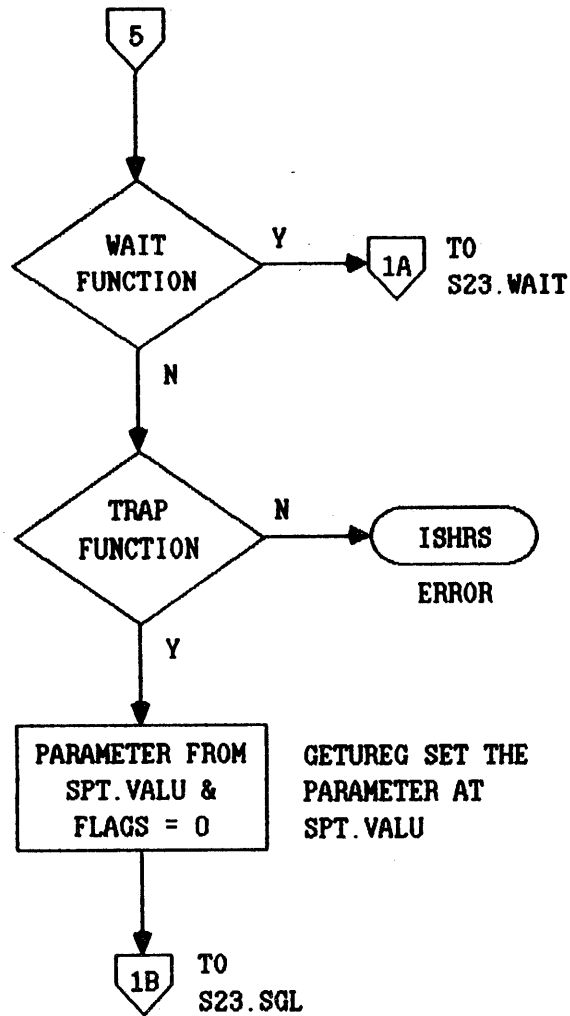
PROCESS A TIME OF DAY REQUEST

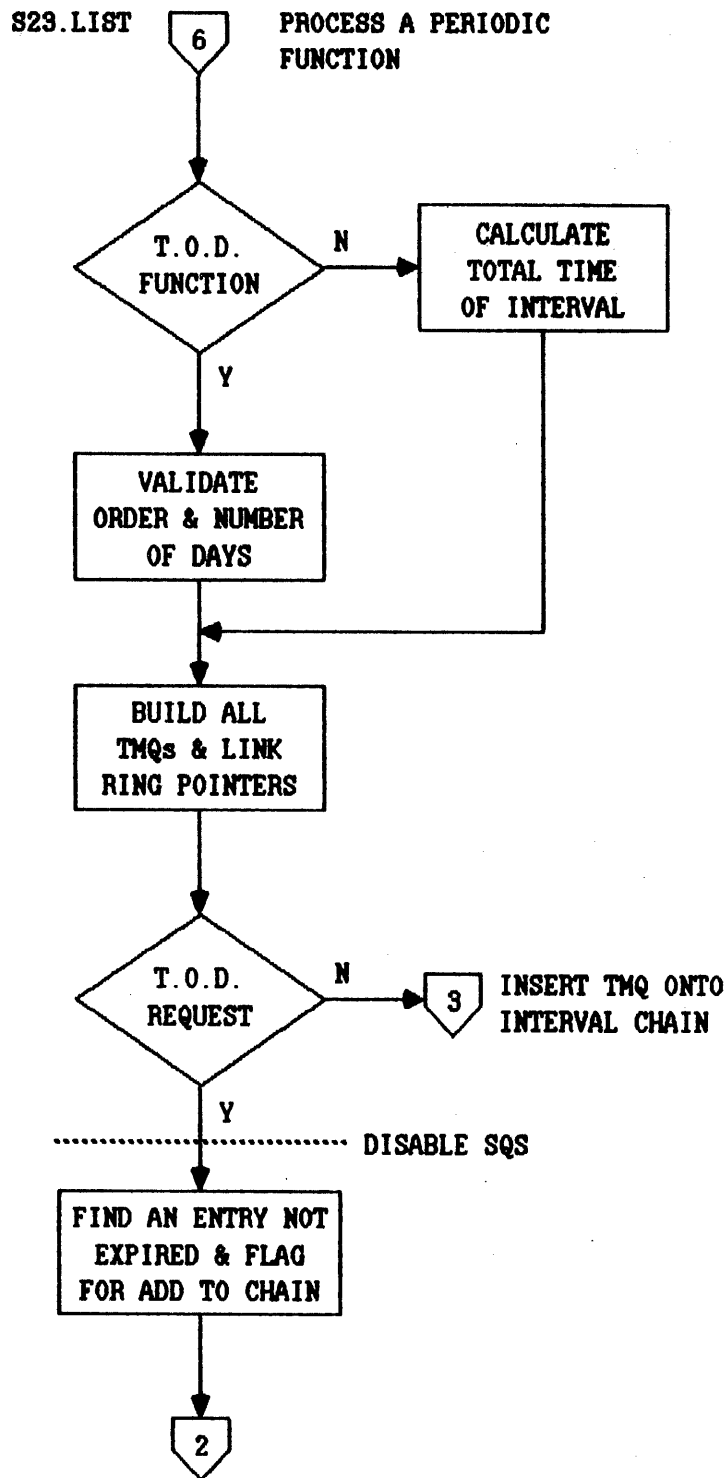






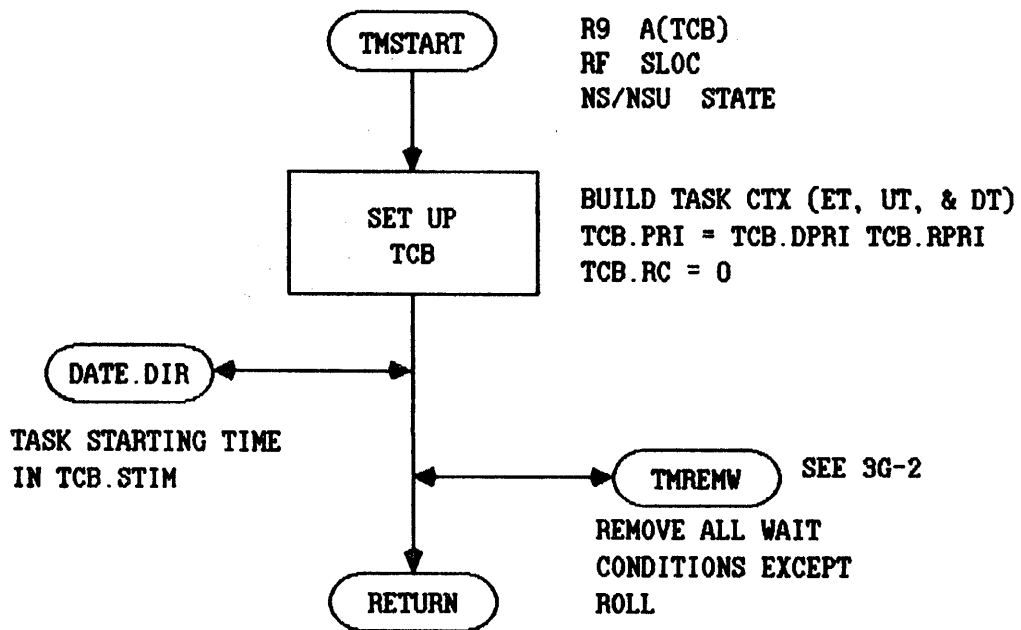






TASK MANAGEMENT

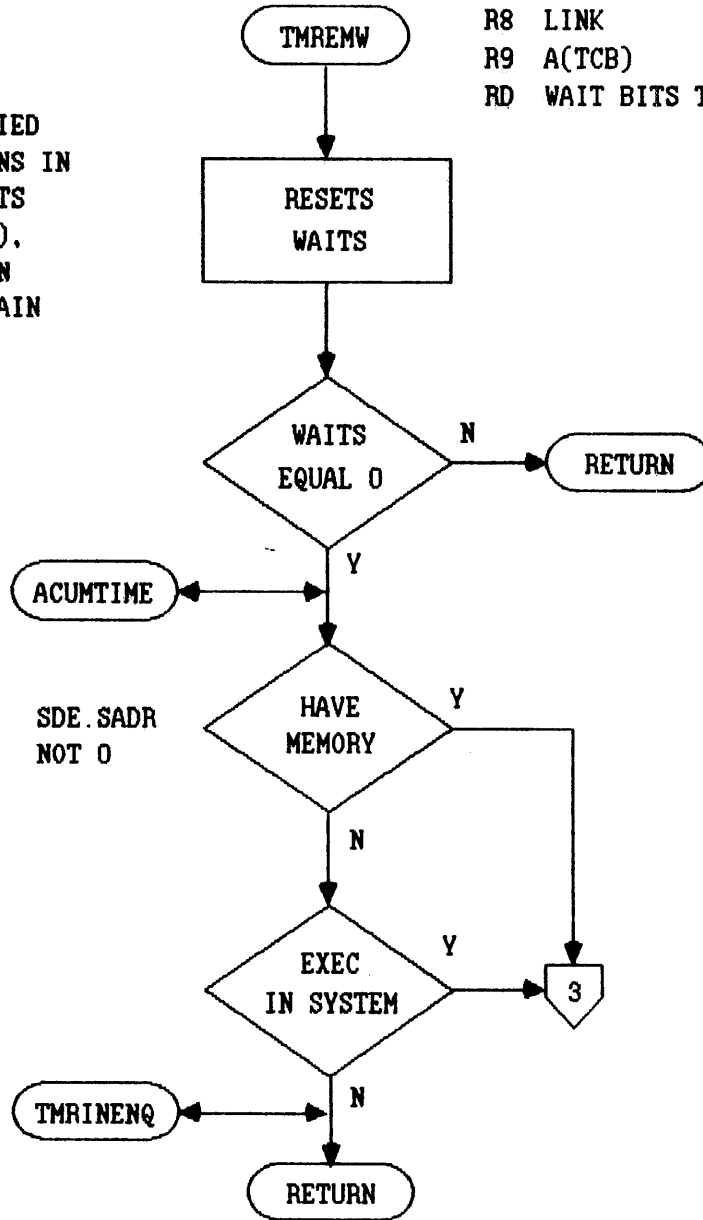
- TASK MANAGER***
- ROLL SUPPORT***

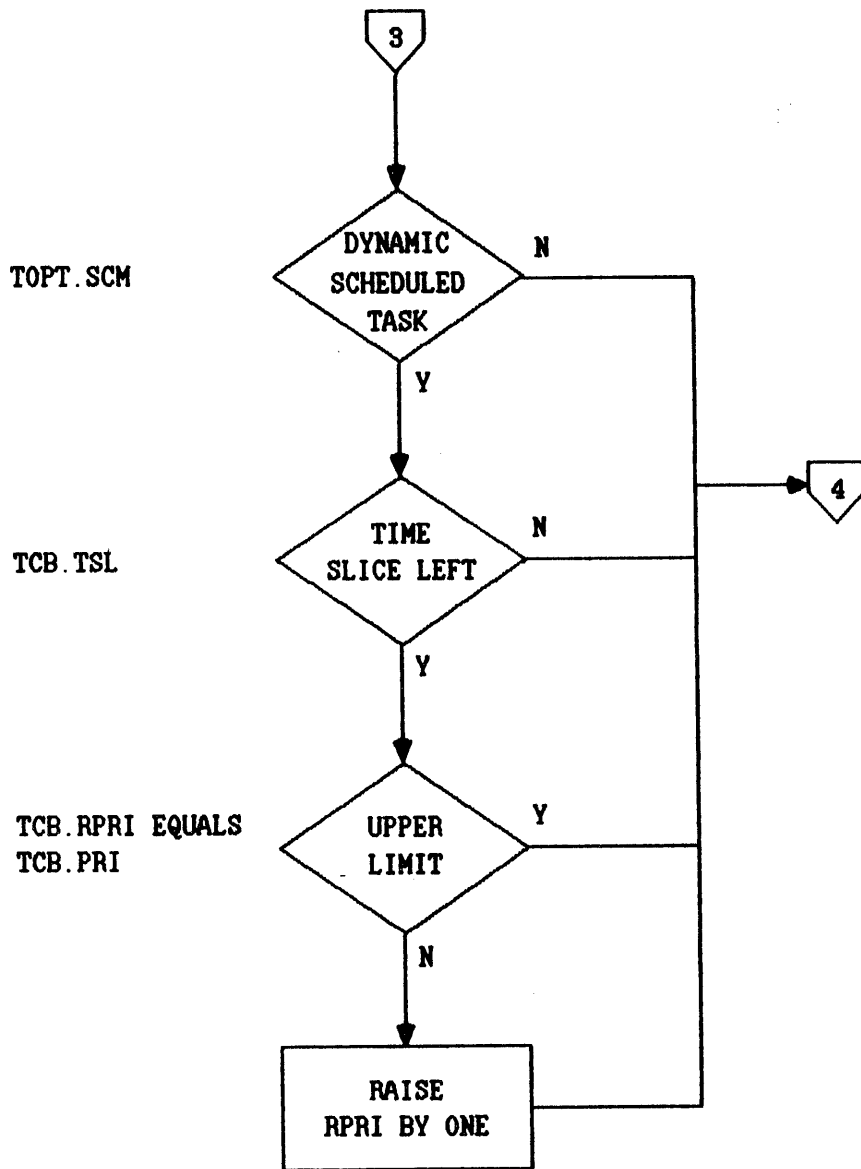


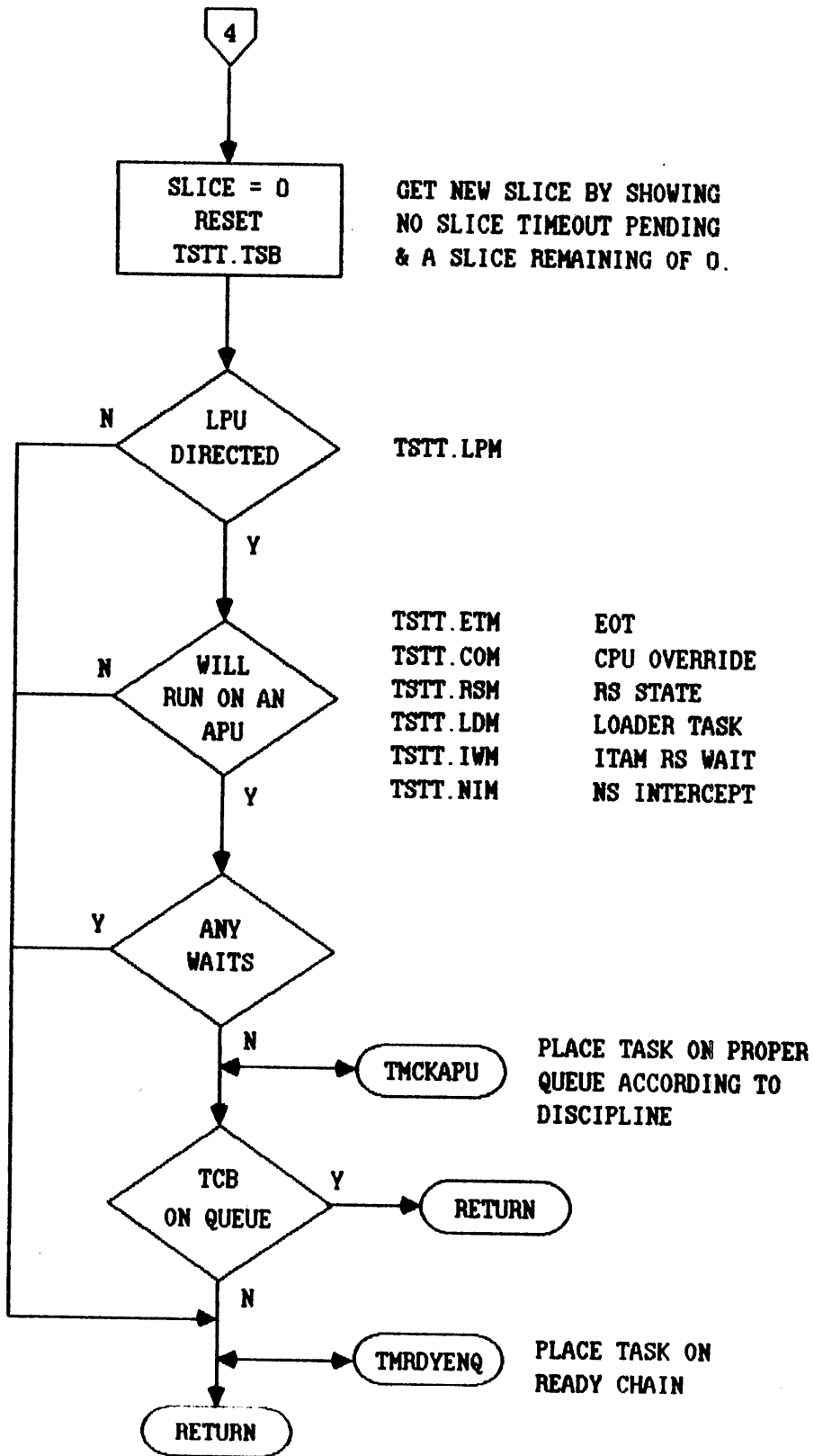
REMOVE SPECIFIED
WAIT CONDITIONS IN
RD. IF NO WAITS
(TCB.WAIT = 0),
INSERTS TCB ON
READY/ROLL CHAIN

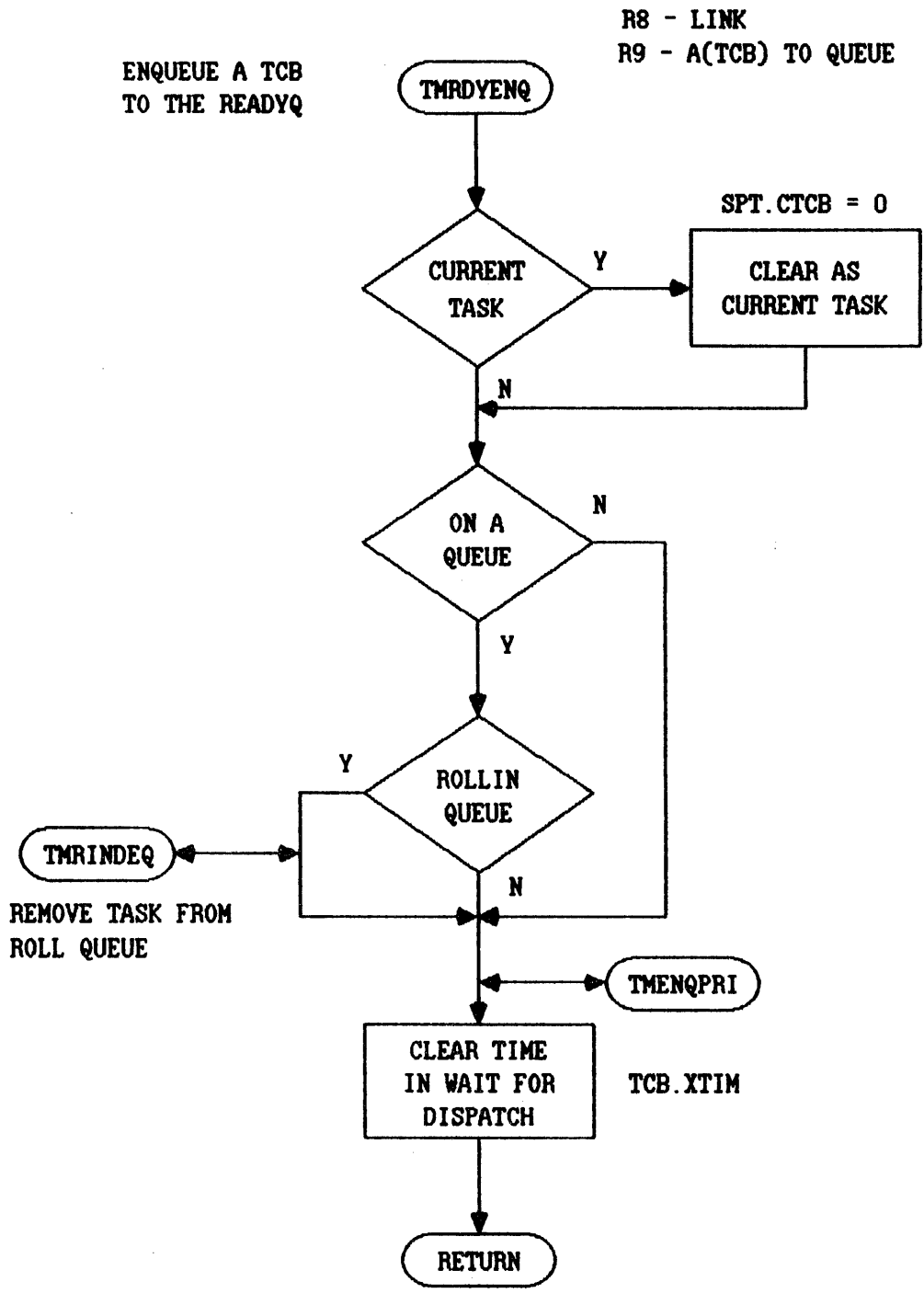
R8 LINK
R9 A(TCB)
RD WAIT BITS TO RESET

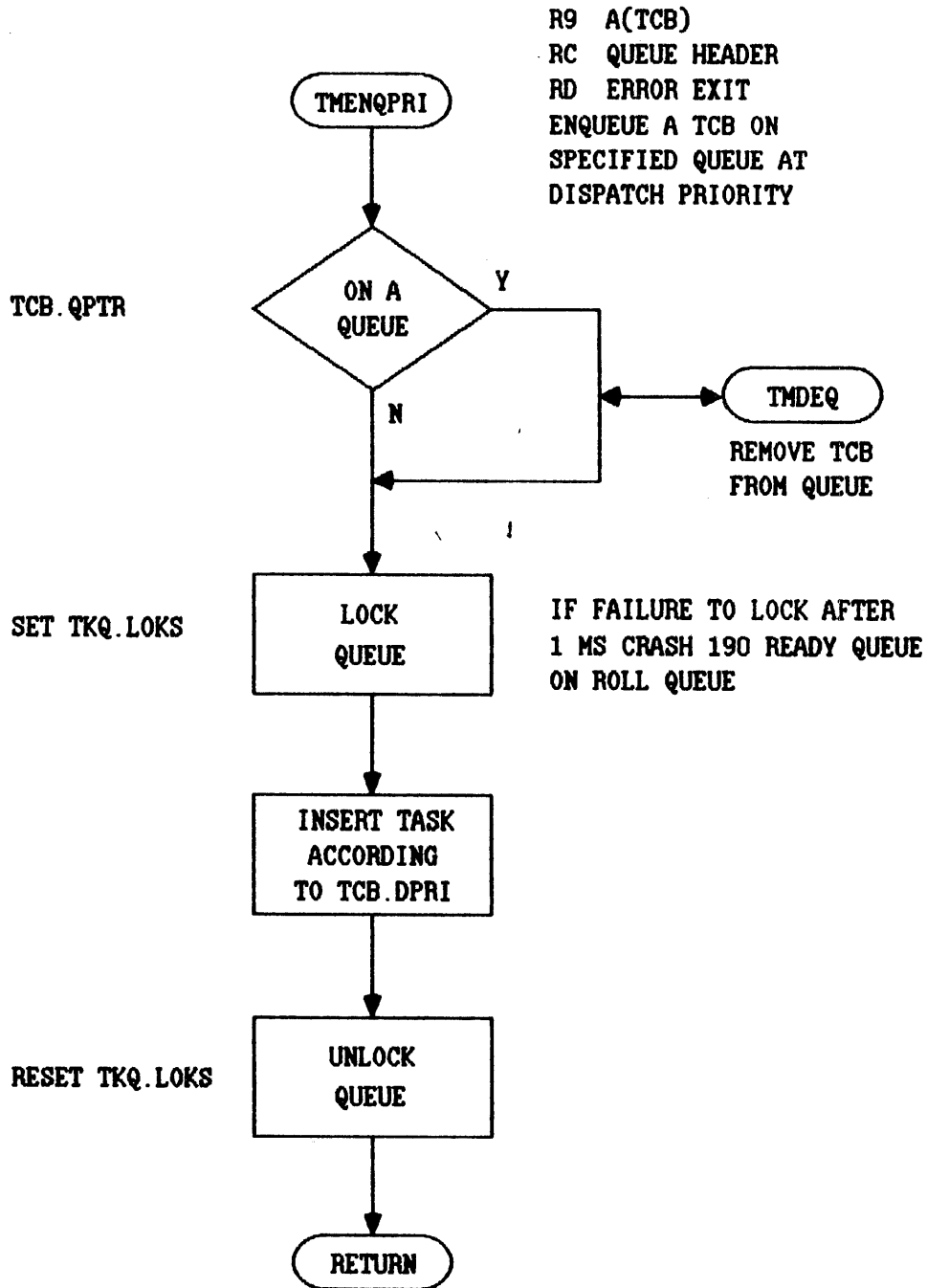
TASK LEAVING WAIT
STATE. ACCUMULATE
WAIT TIME

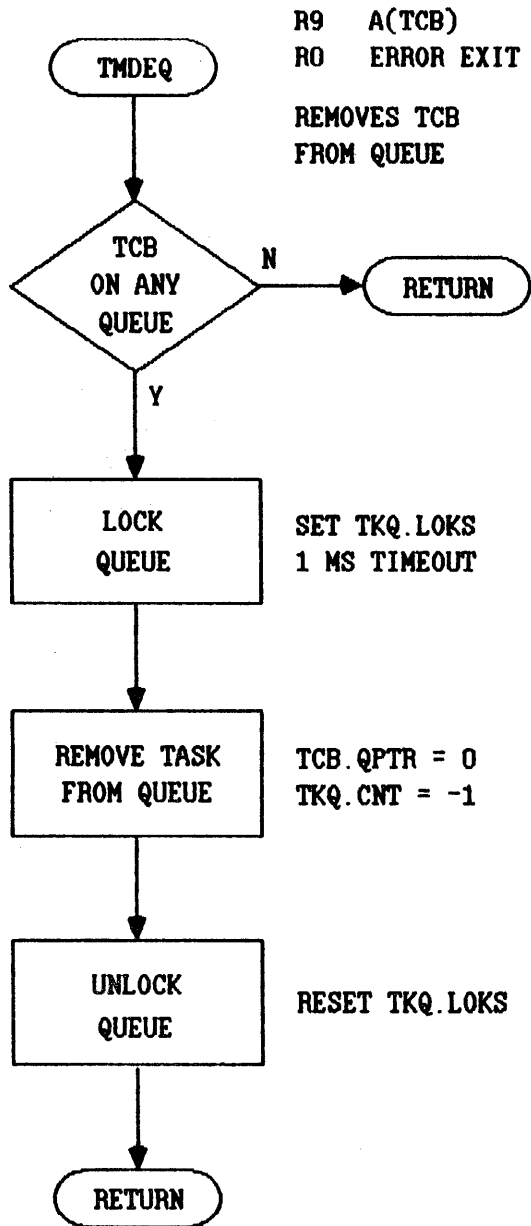


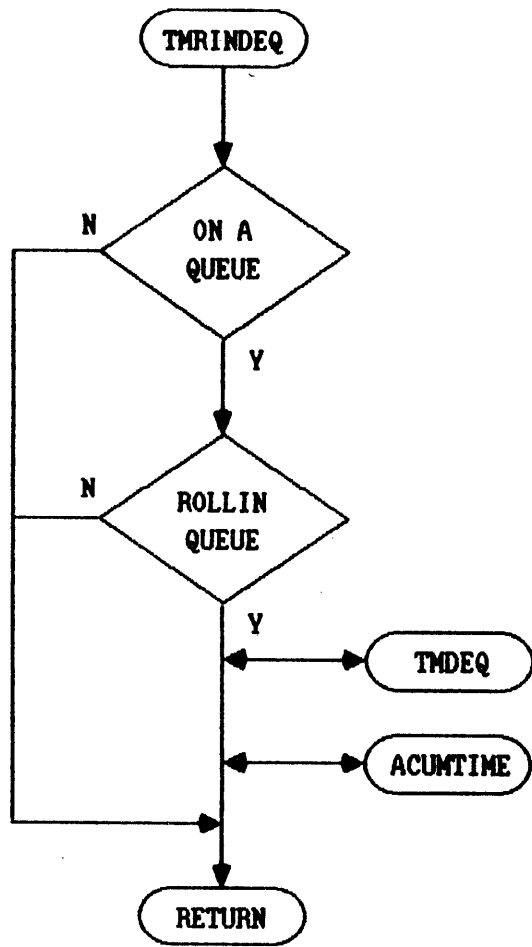


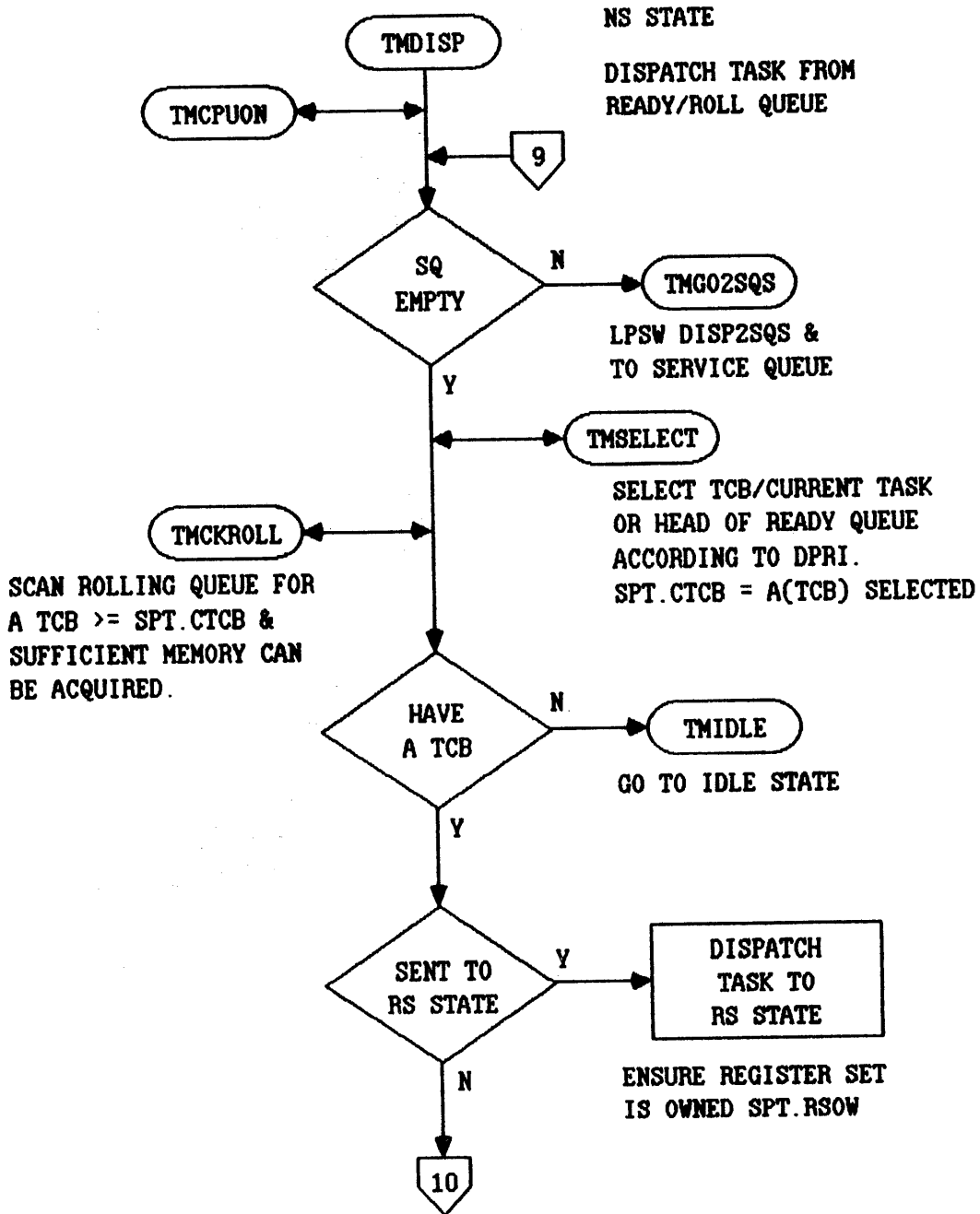


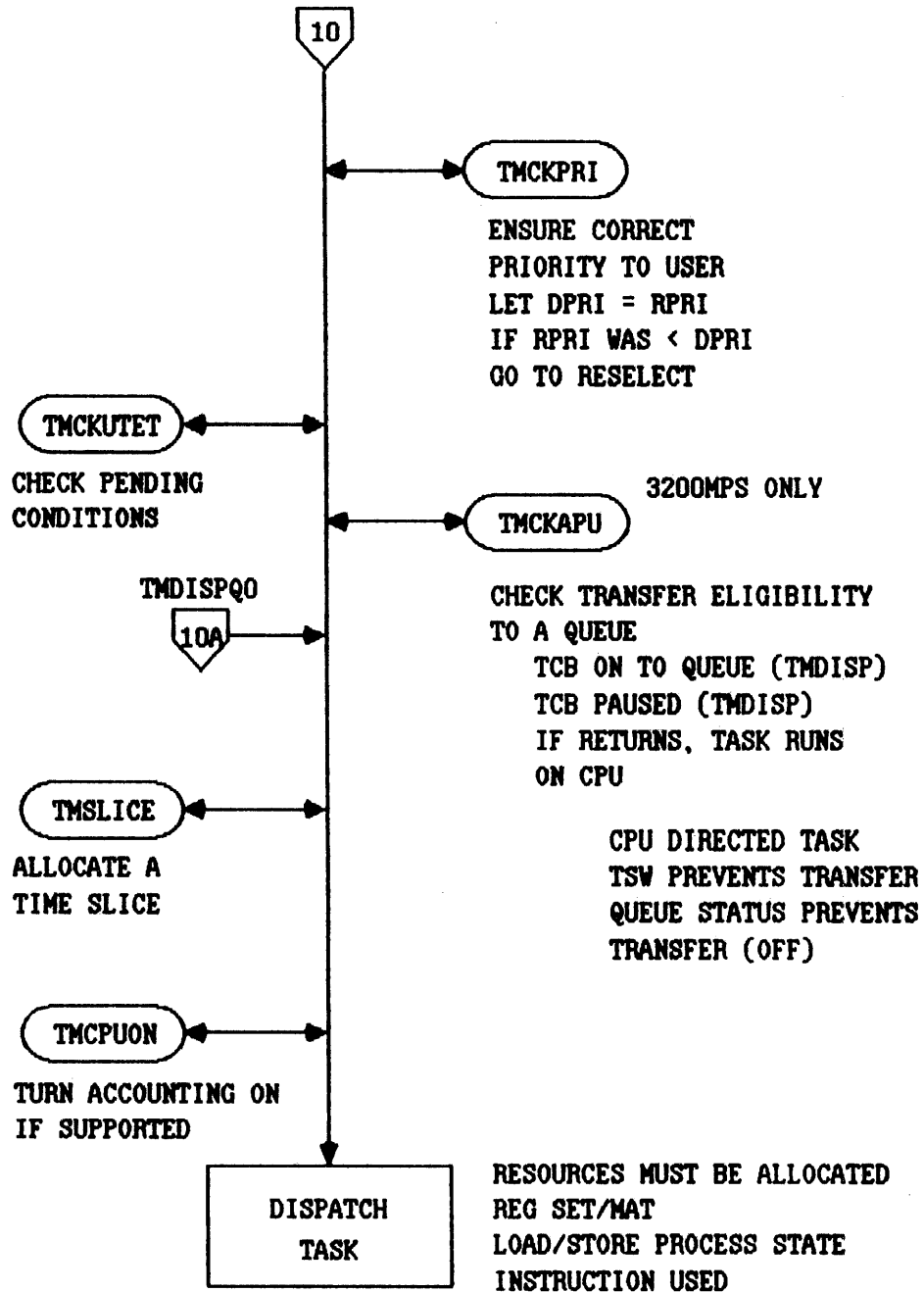


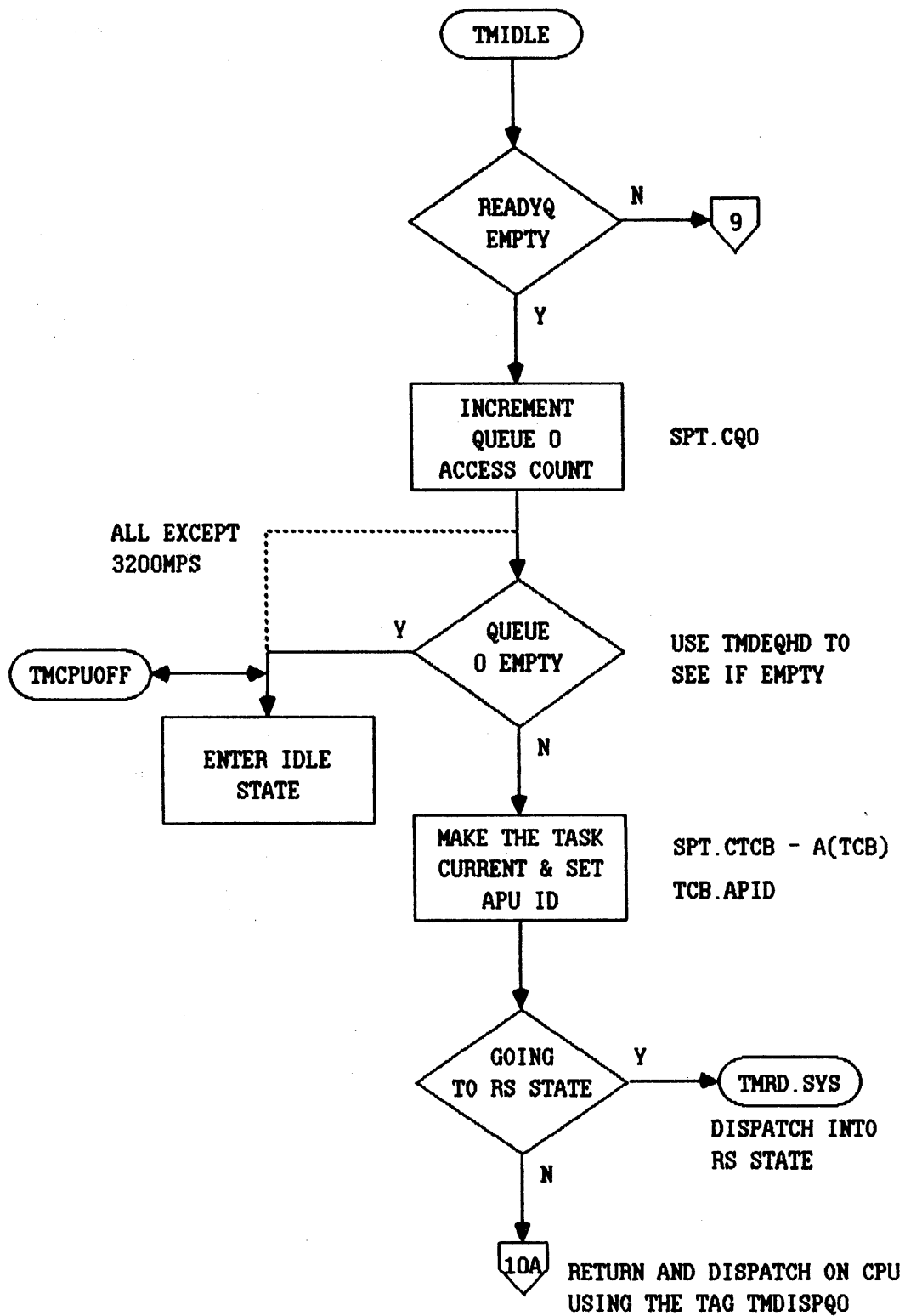


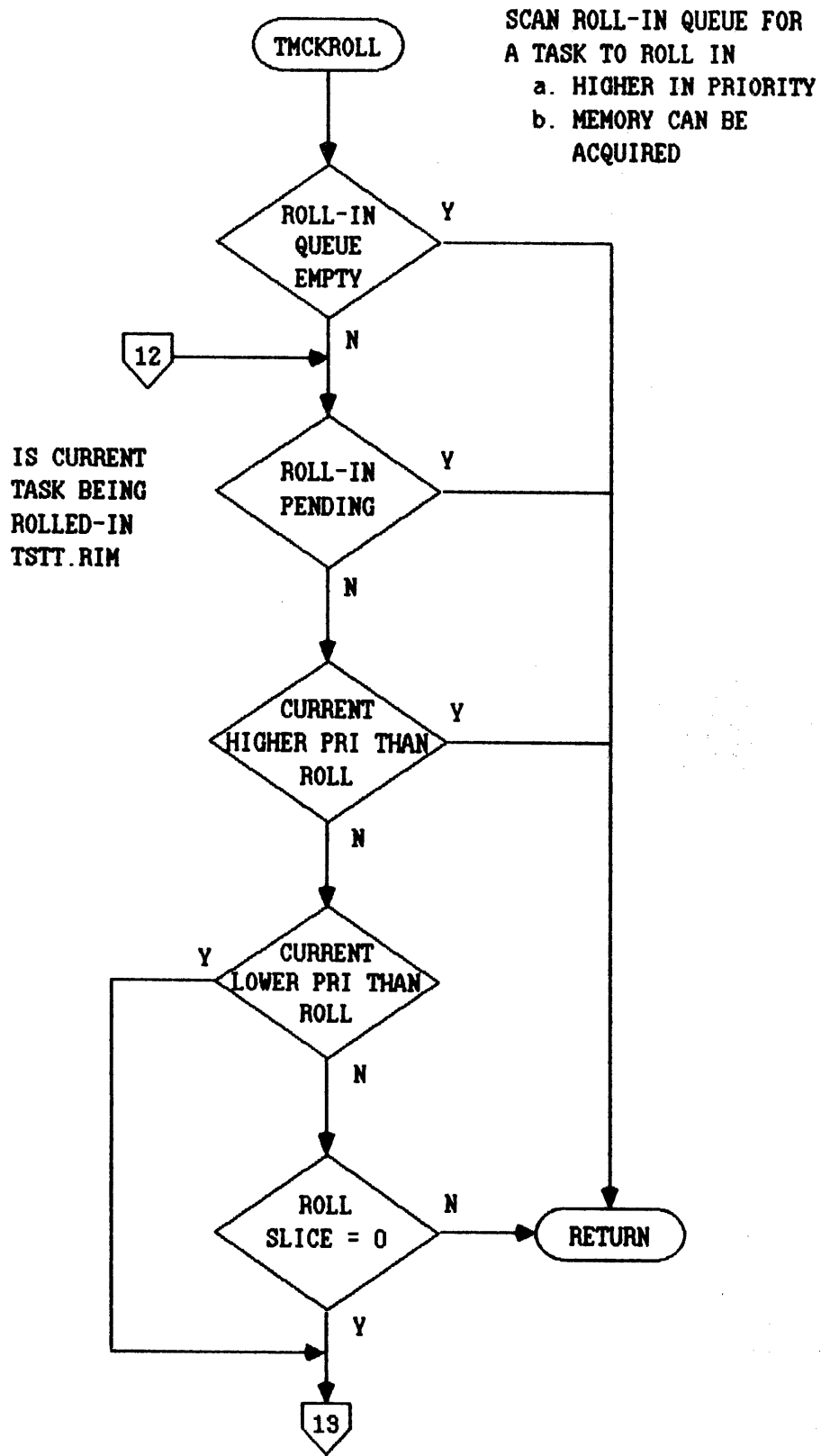


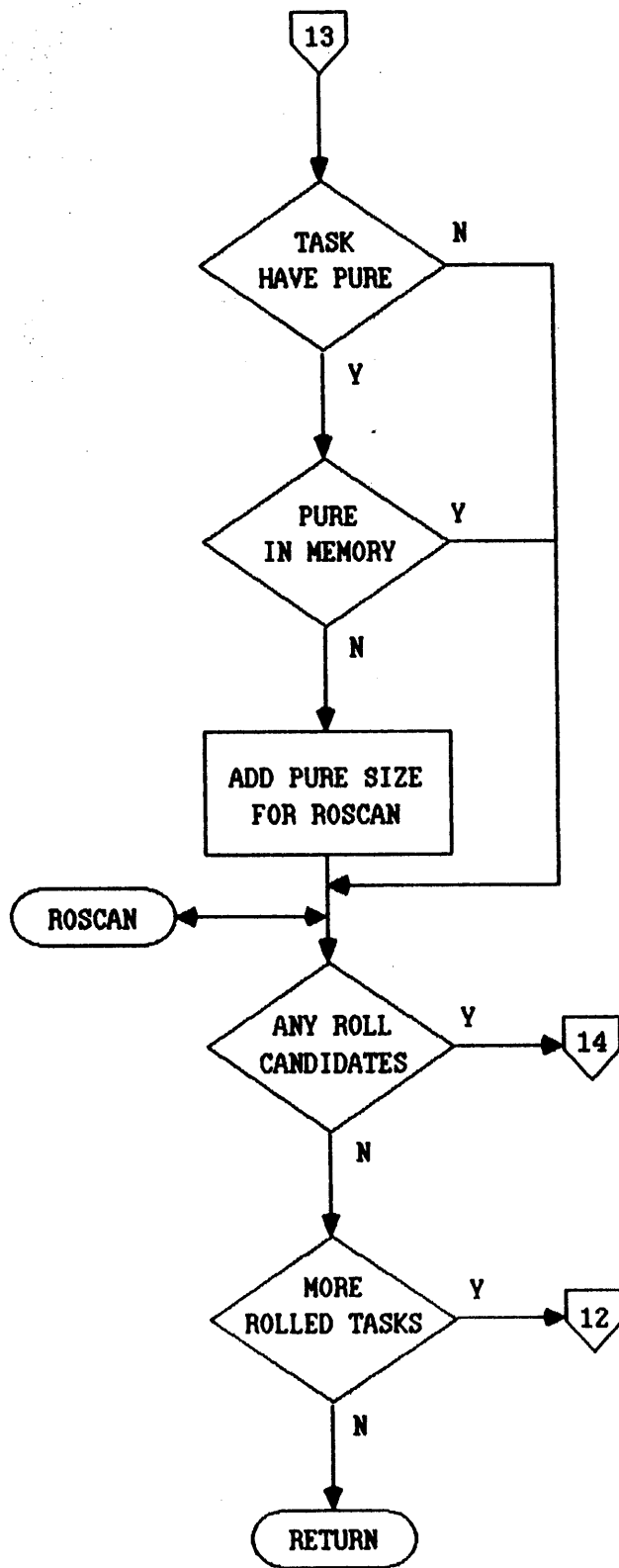


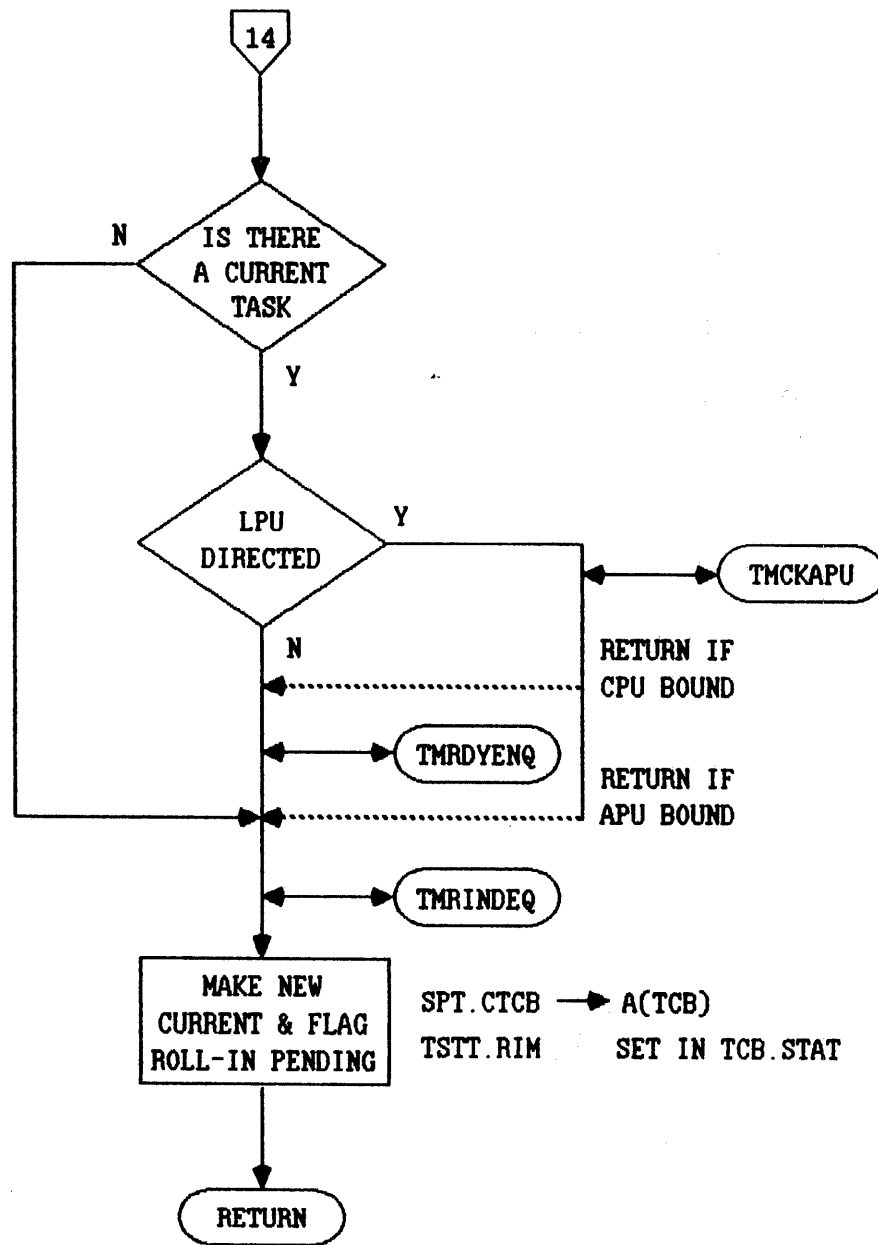


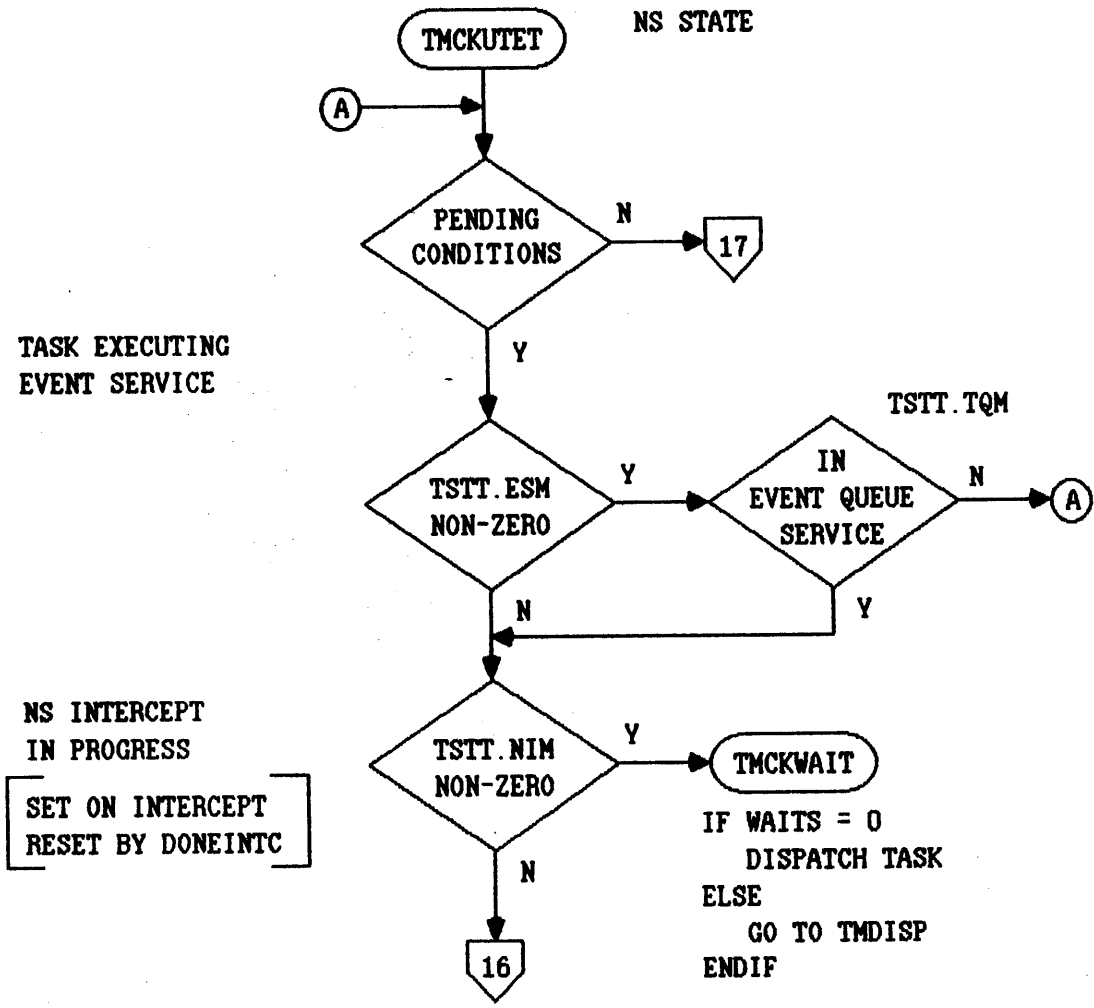


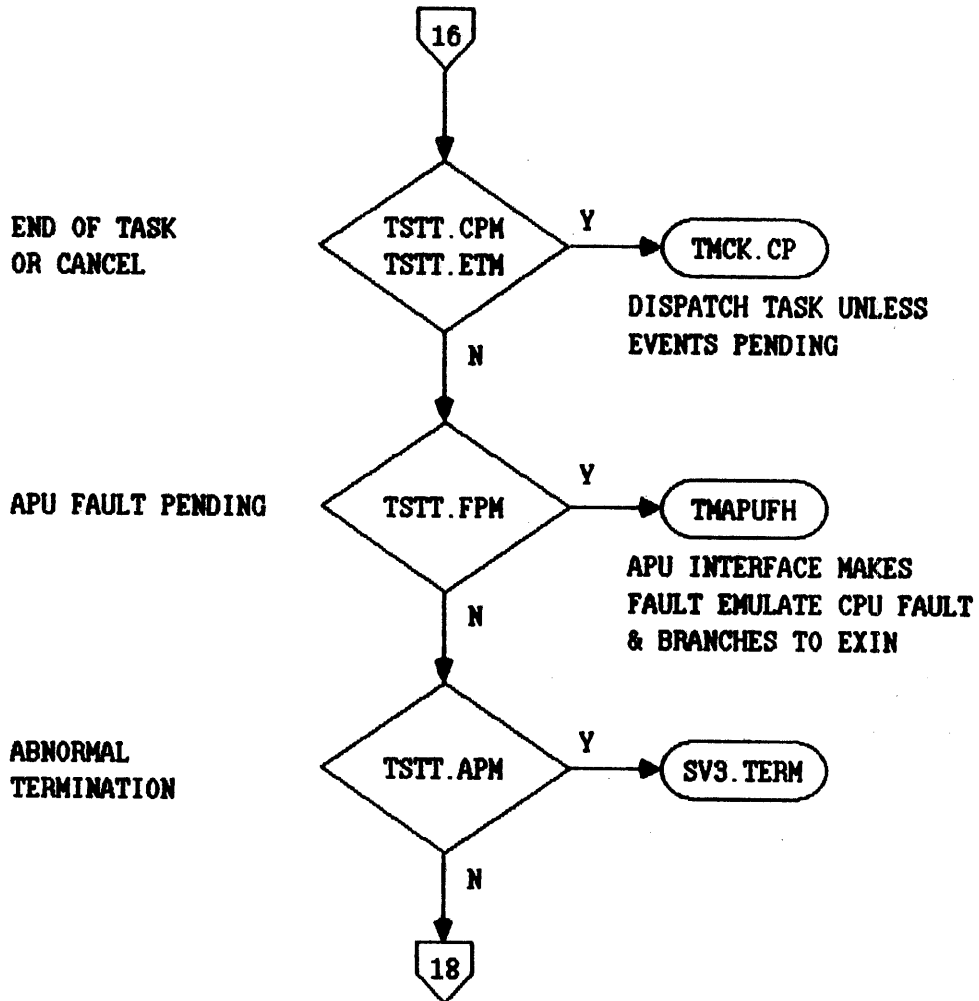


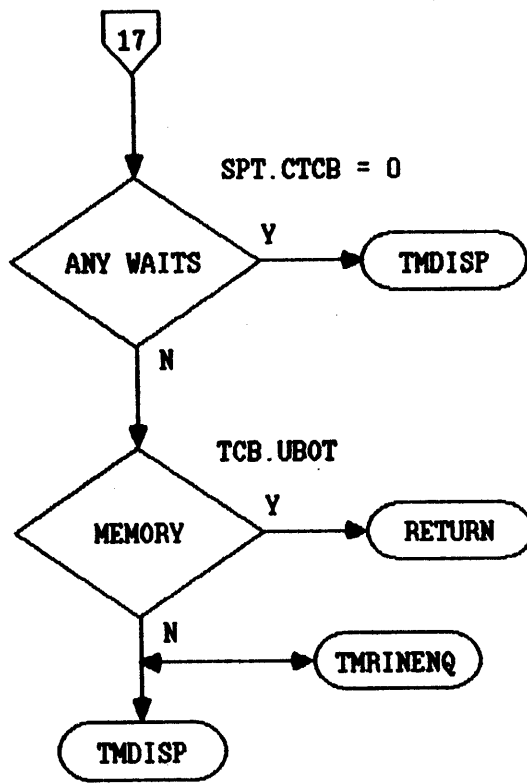


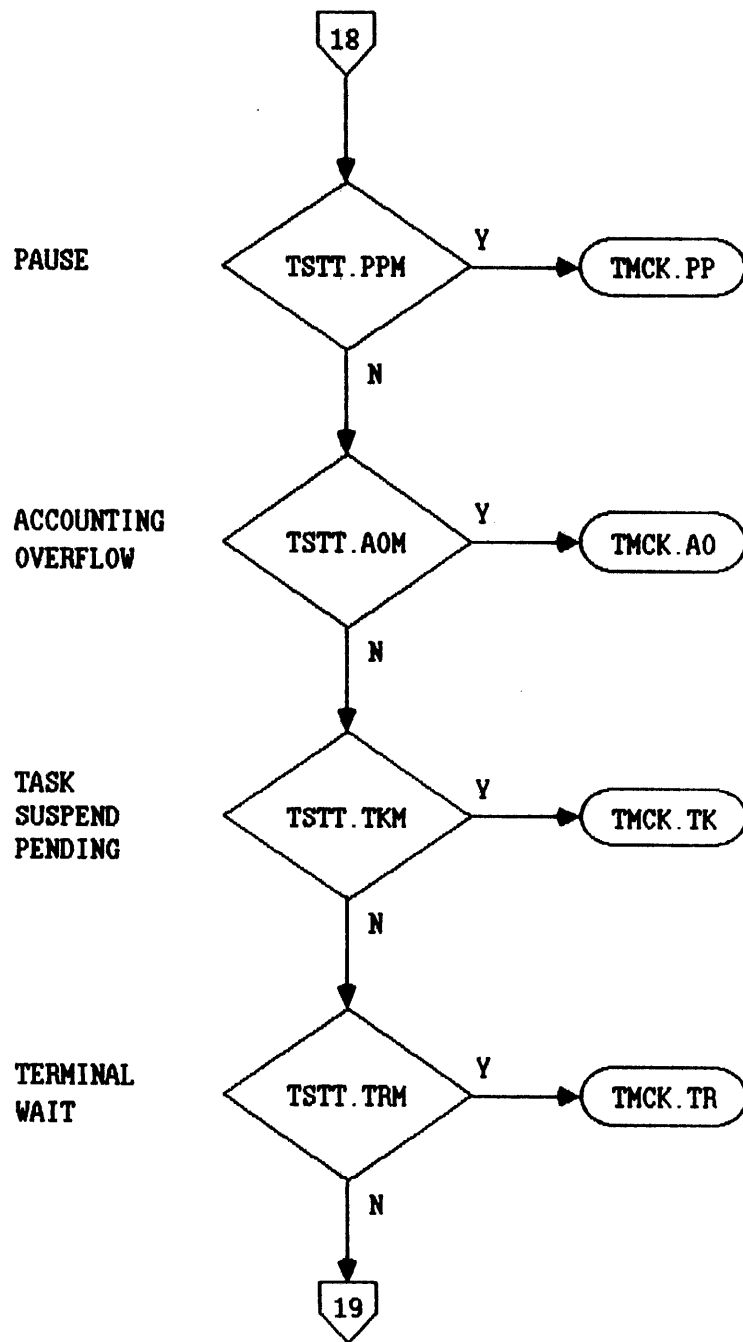


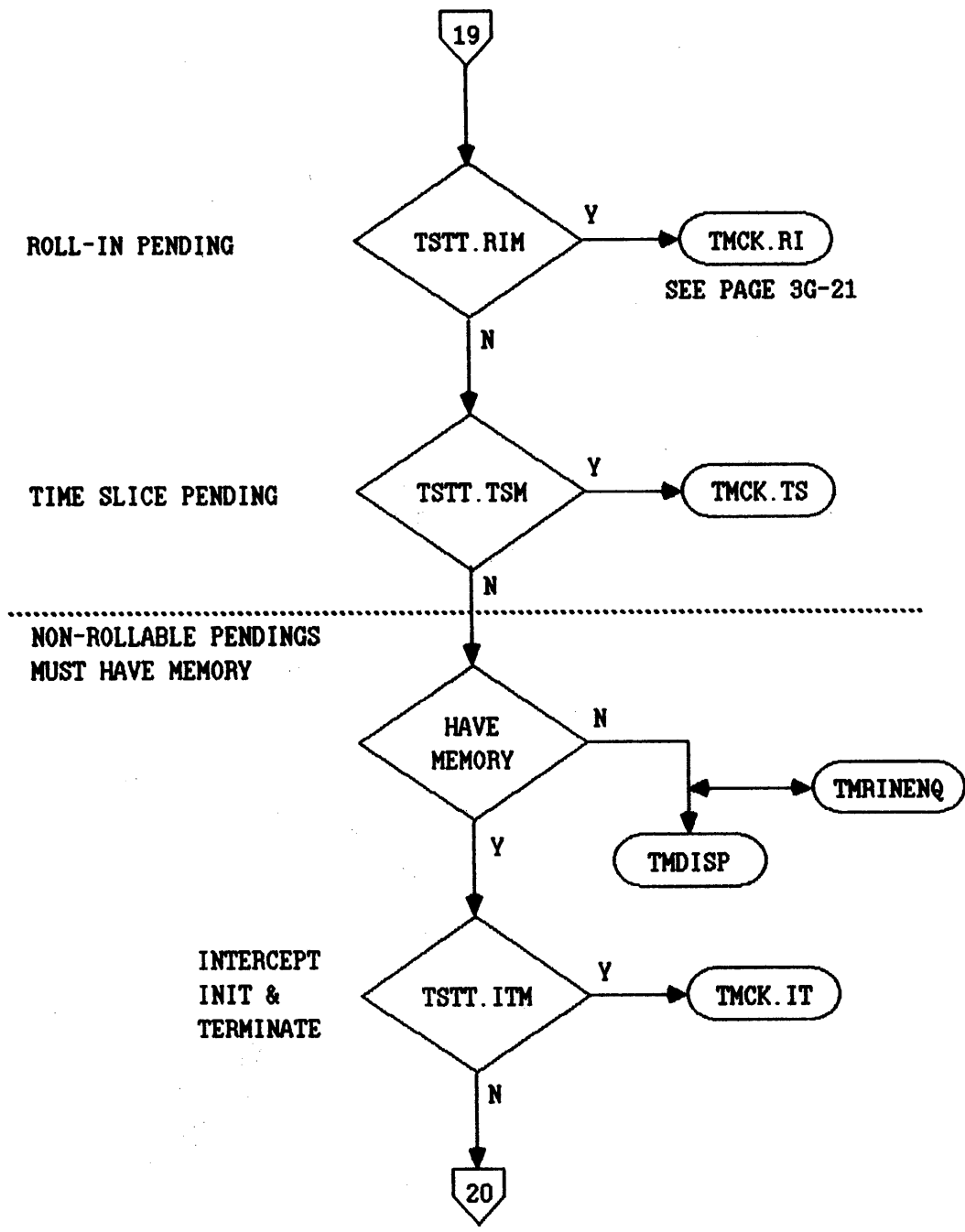


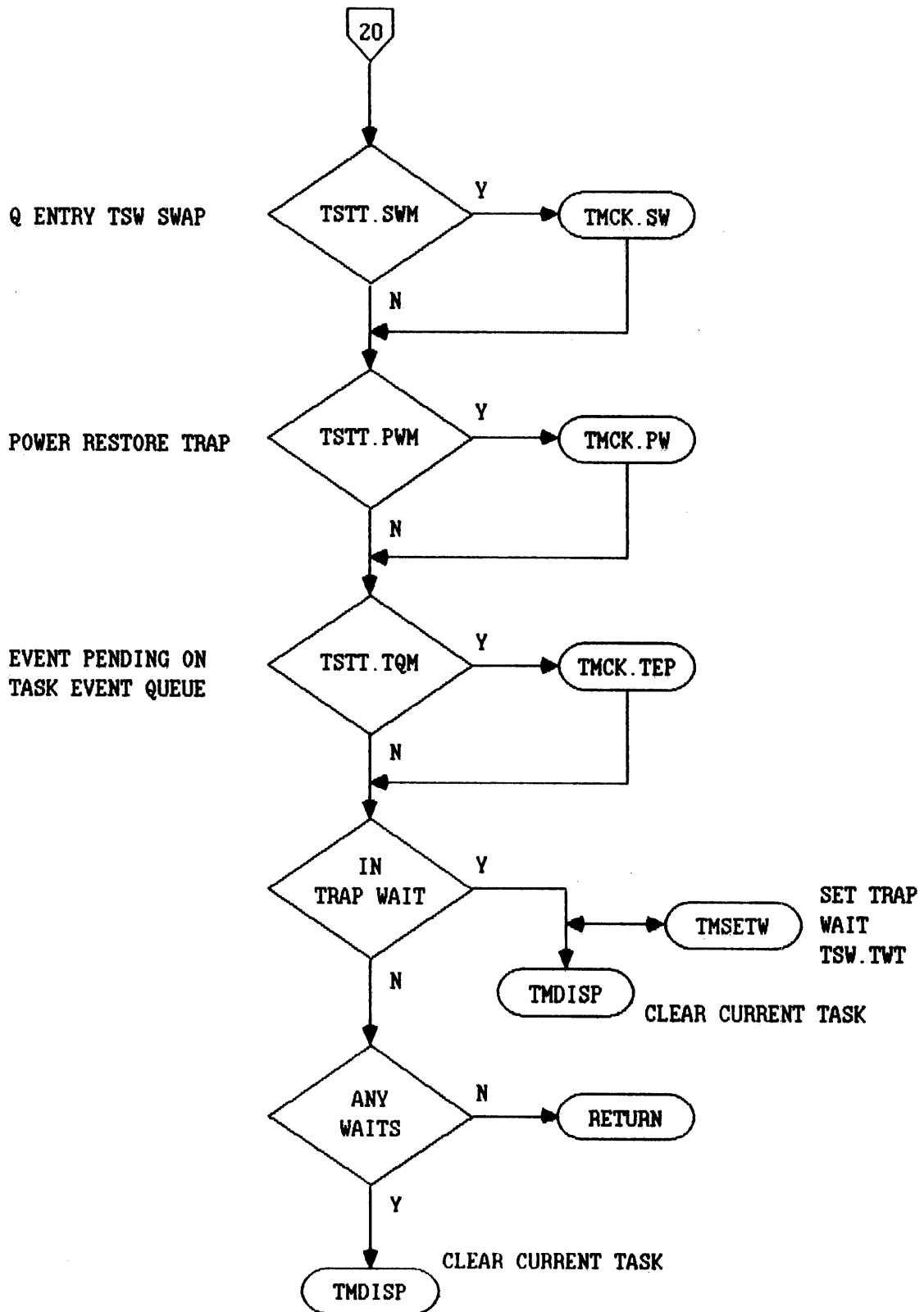


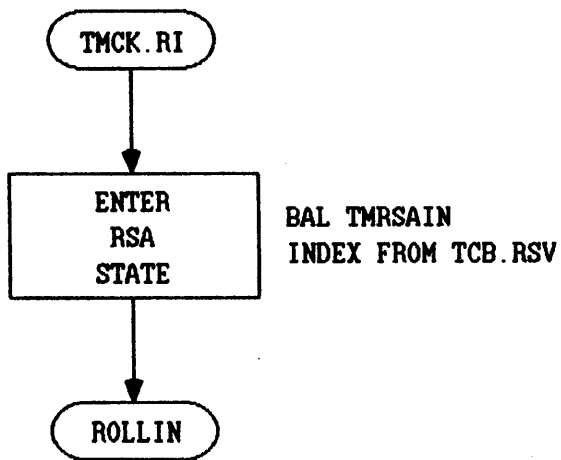


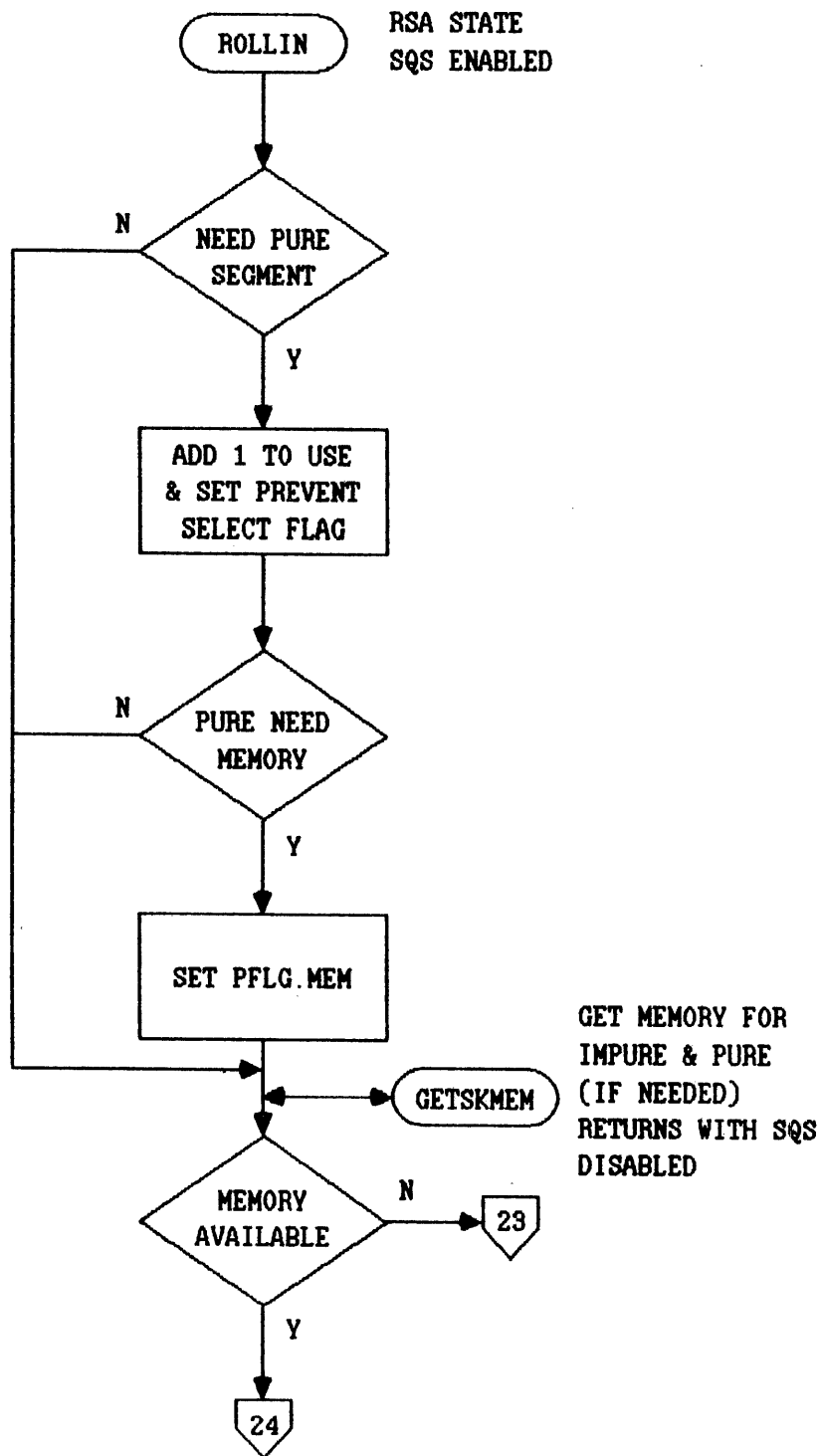


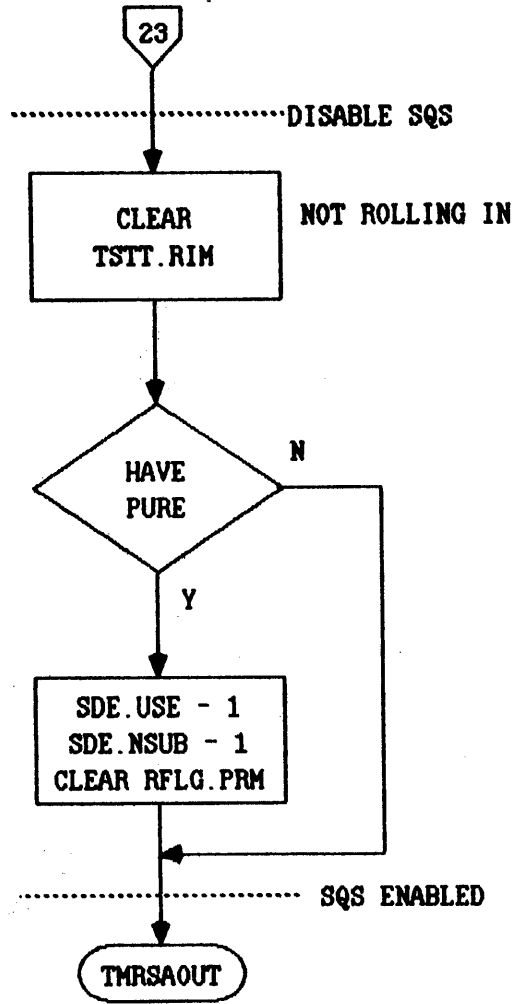


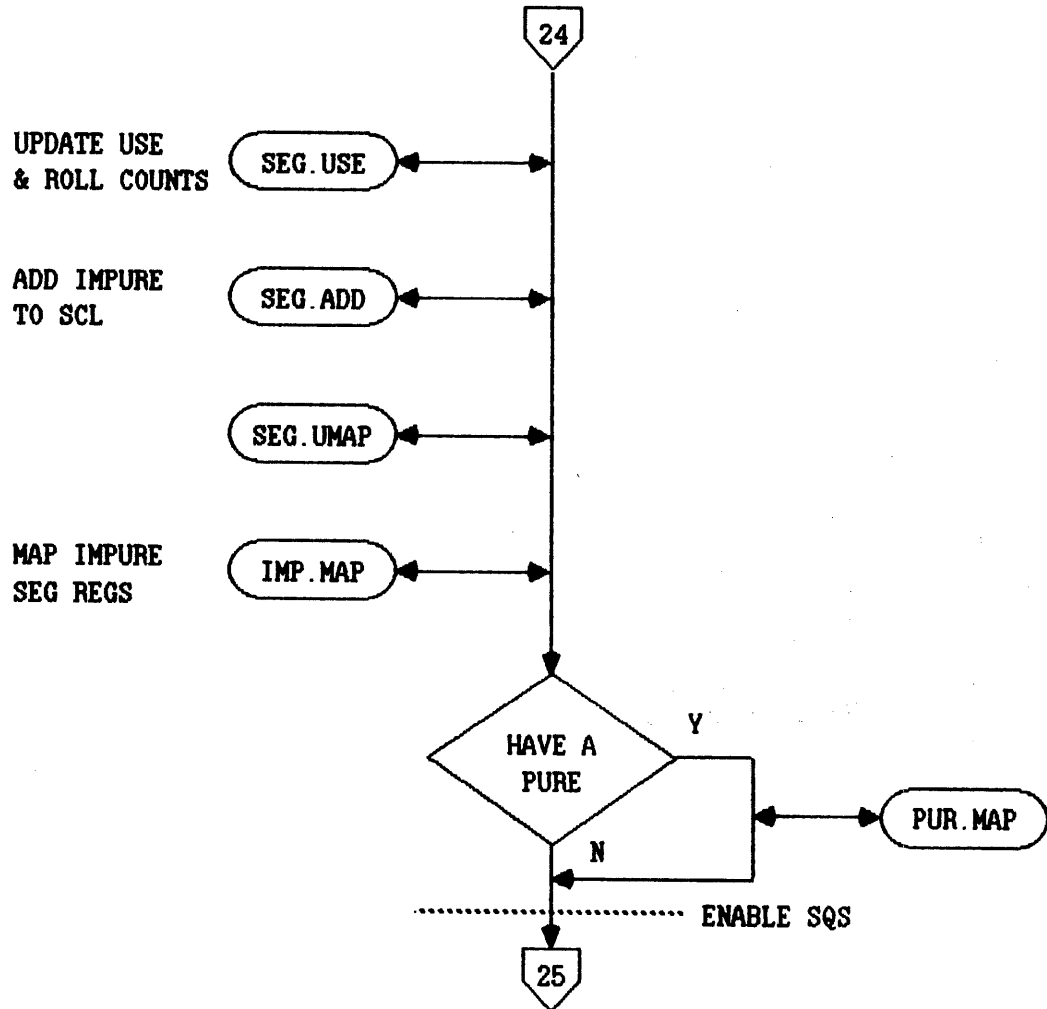


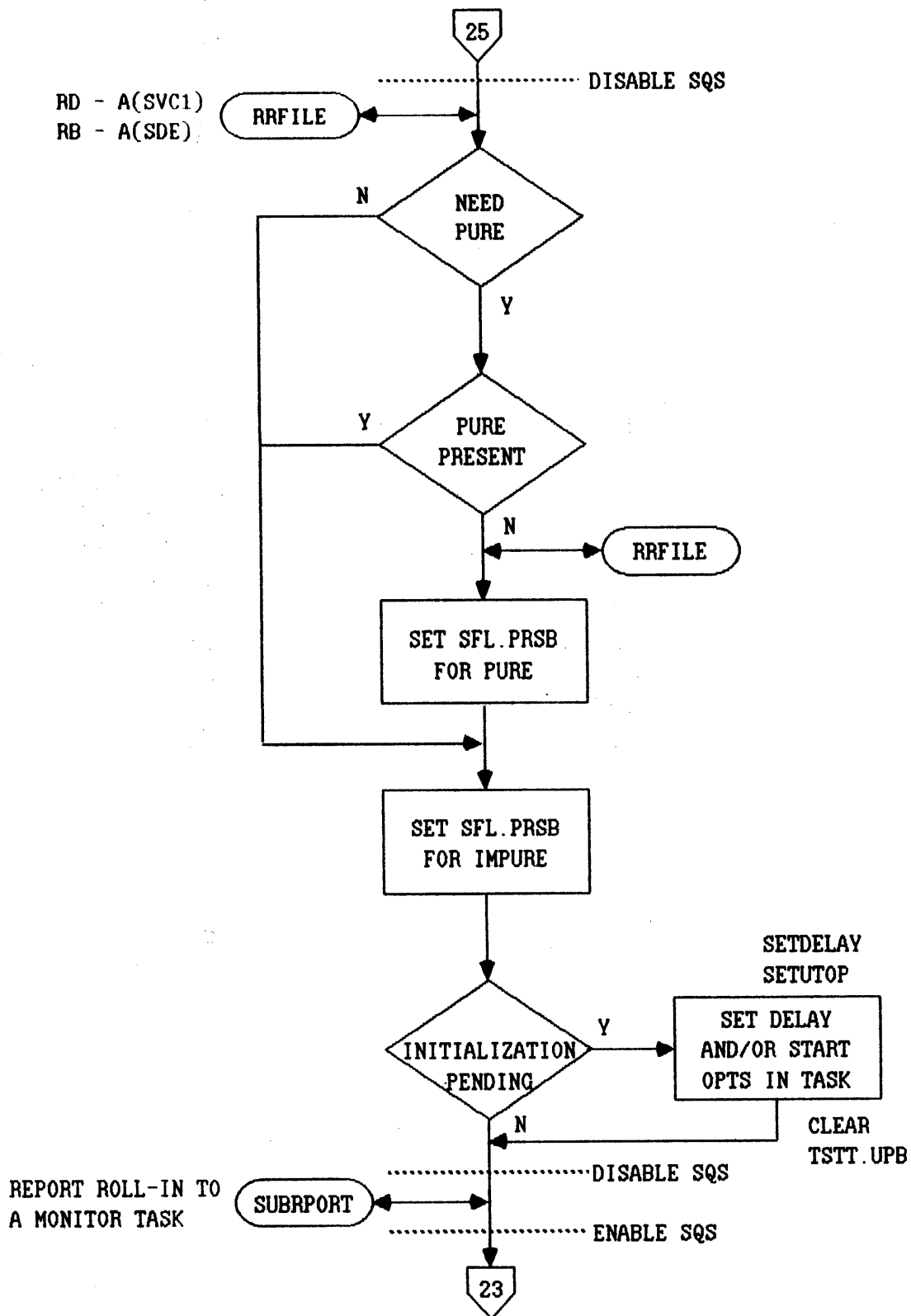








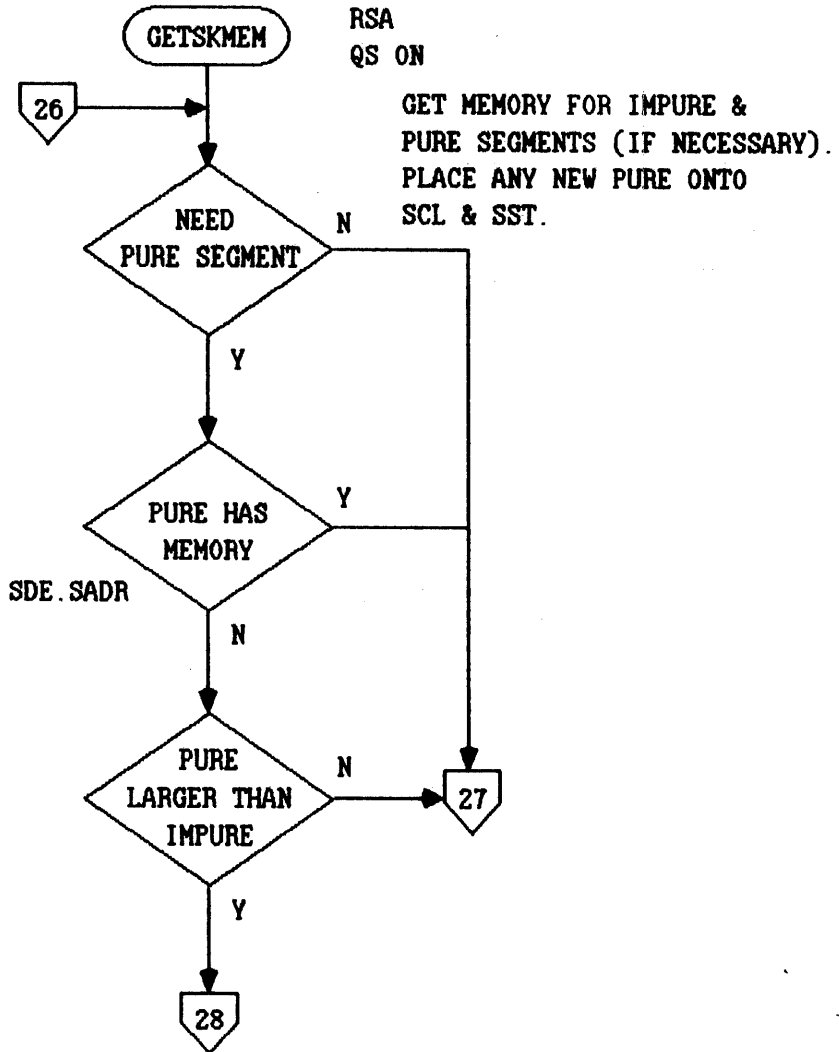




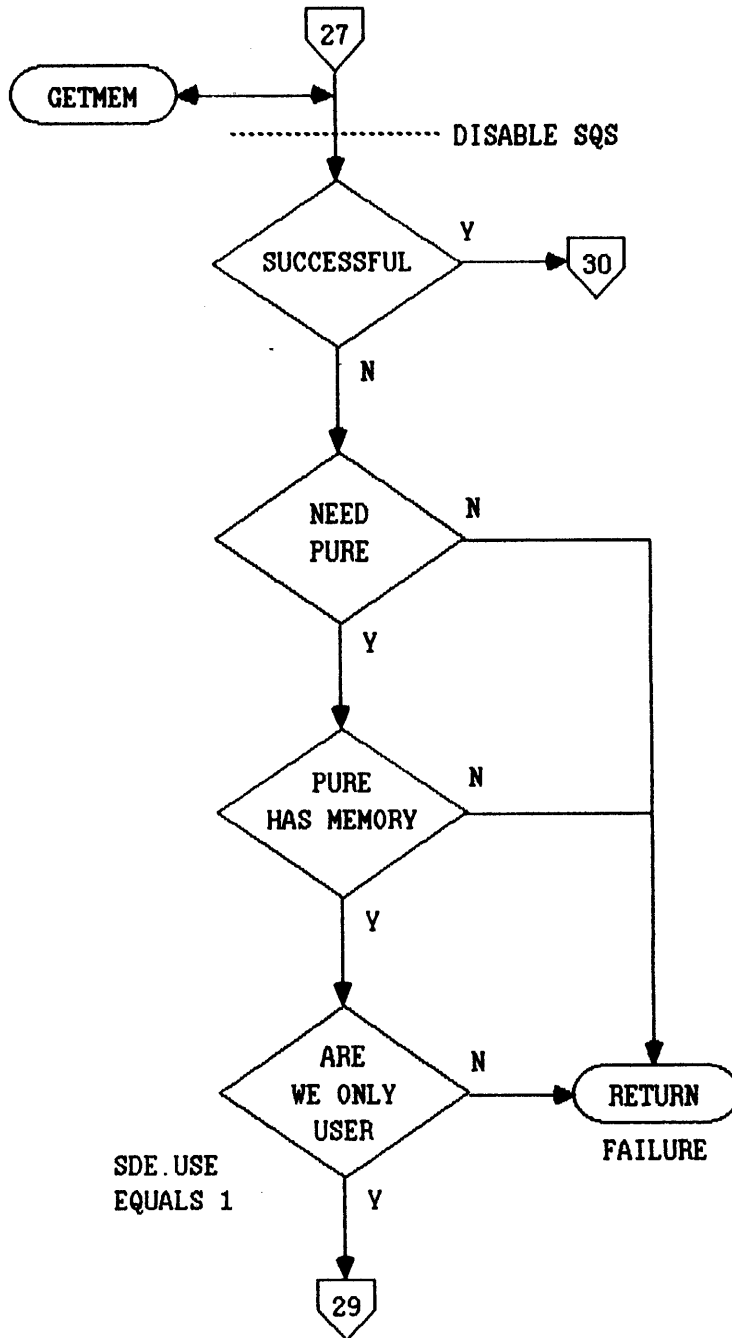
```

IF IMPURE GREATER
  THAN PURE:
  GETMEM (IMPURE)
  GETMEM (PURE)
ELSE
  GETMEM (PURE)
  GETMEM (IMPURE)
ENDIF

```

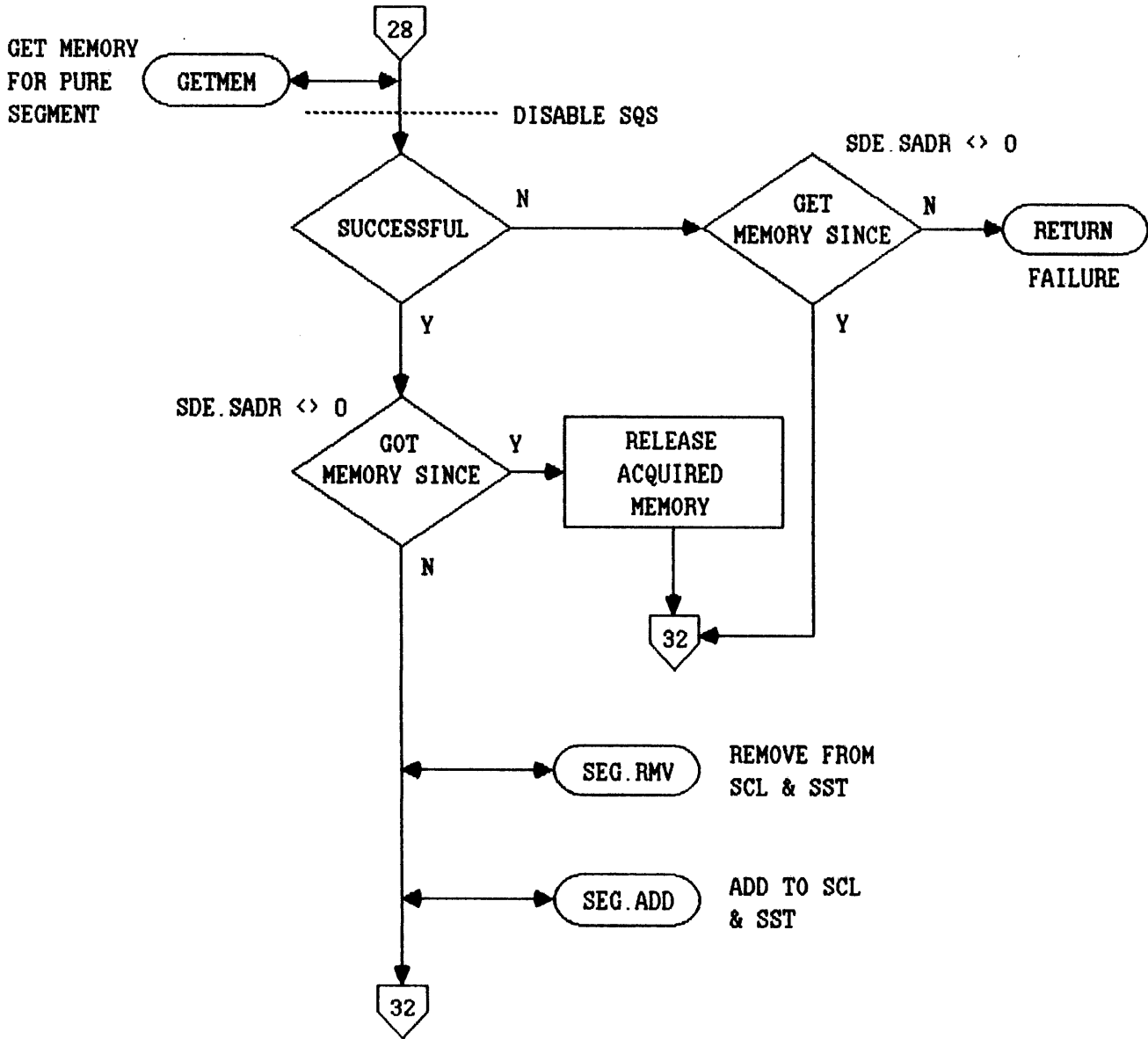


GET MEMORY FOR
IMPURE SEGMENT



SDE . USE
EQUALS 1

REMOVE EXISTING PURE
SEGMENT & RETRY.



IMPURE ROLLIN FAILED
HAVE MEMORY FOR PURE SEGMENT

EXTM/TMROLL

RELEASE MEMORY
RESET PRESENCE BIT
SFL.PRSB

RELMEM

NECESSARY FOR
LOADING IN
ROLLED STATE

SEG.ADD

SEG.RMV

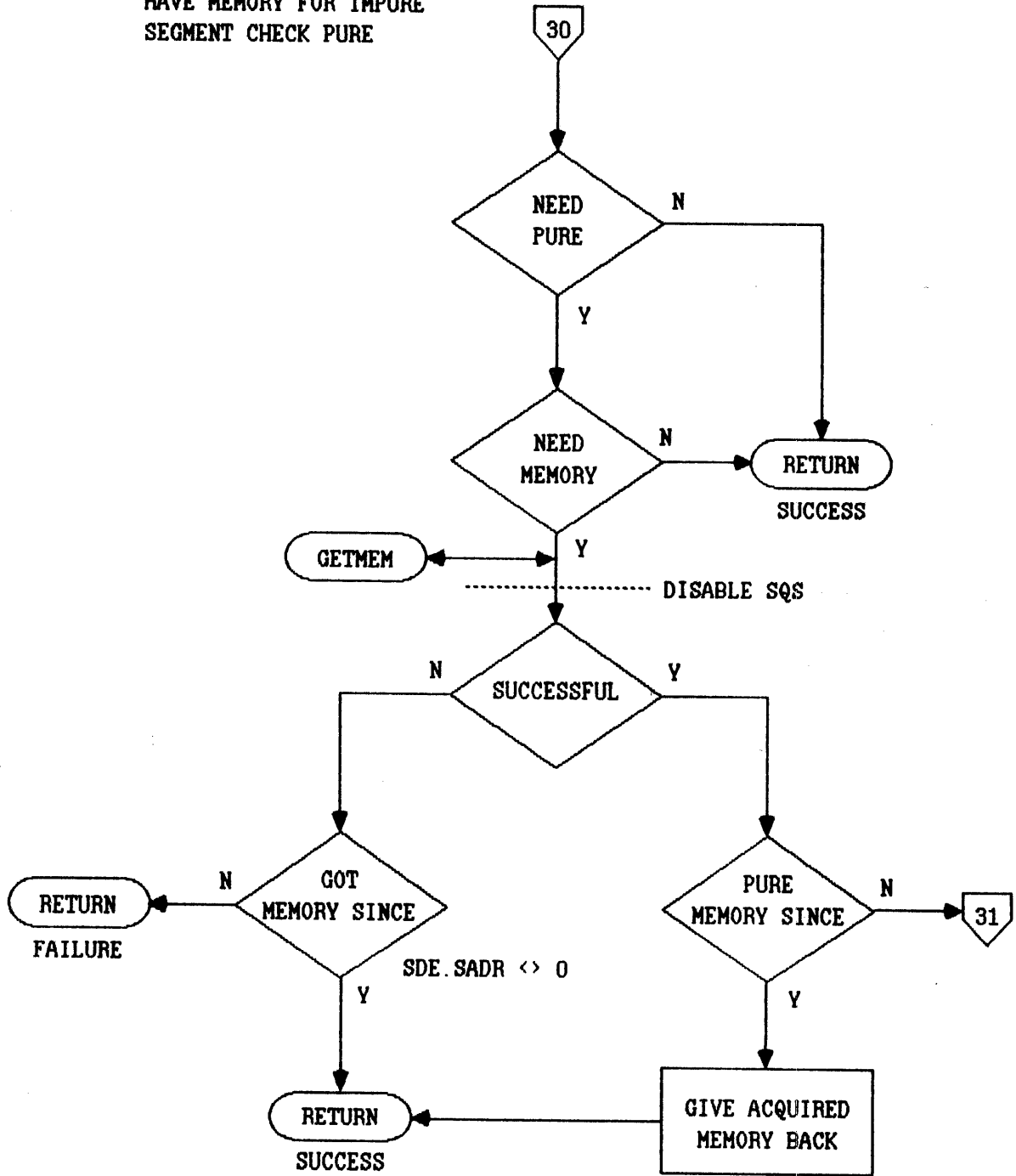
REMOVE FROM
SCL & SST

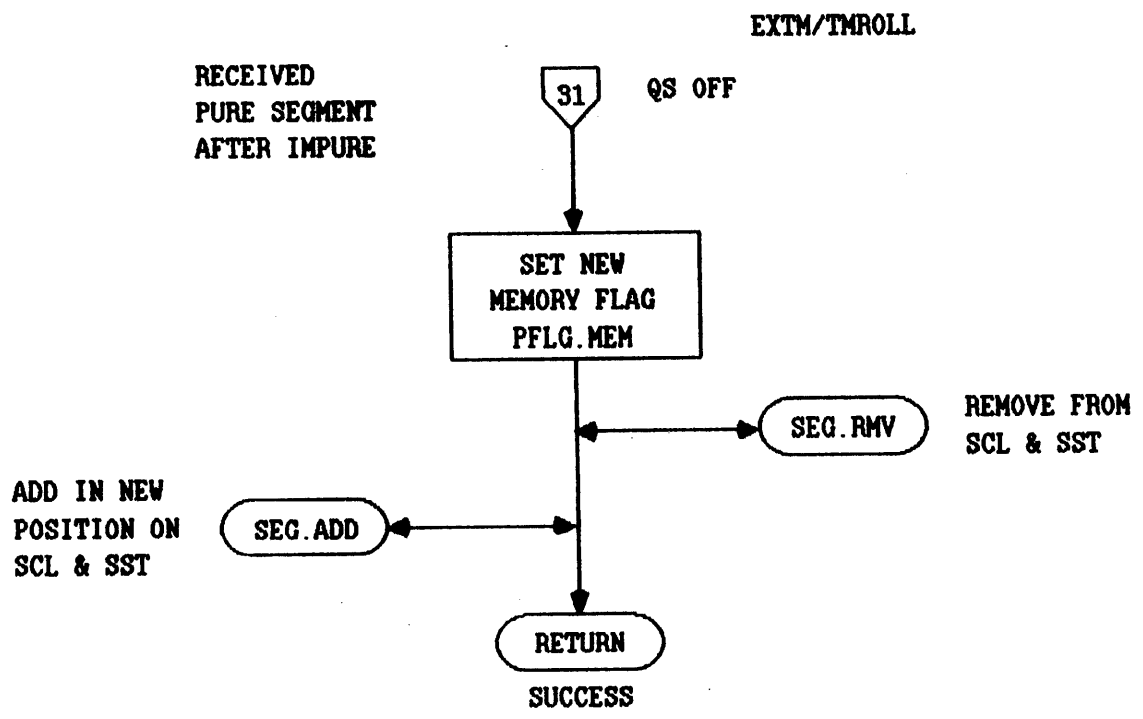
29

26

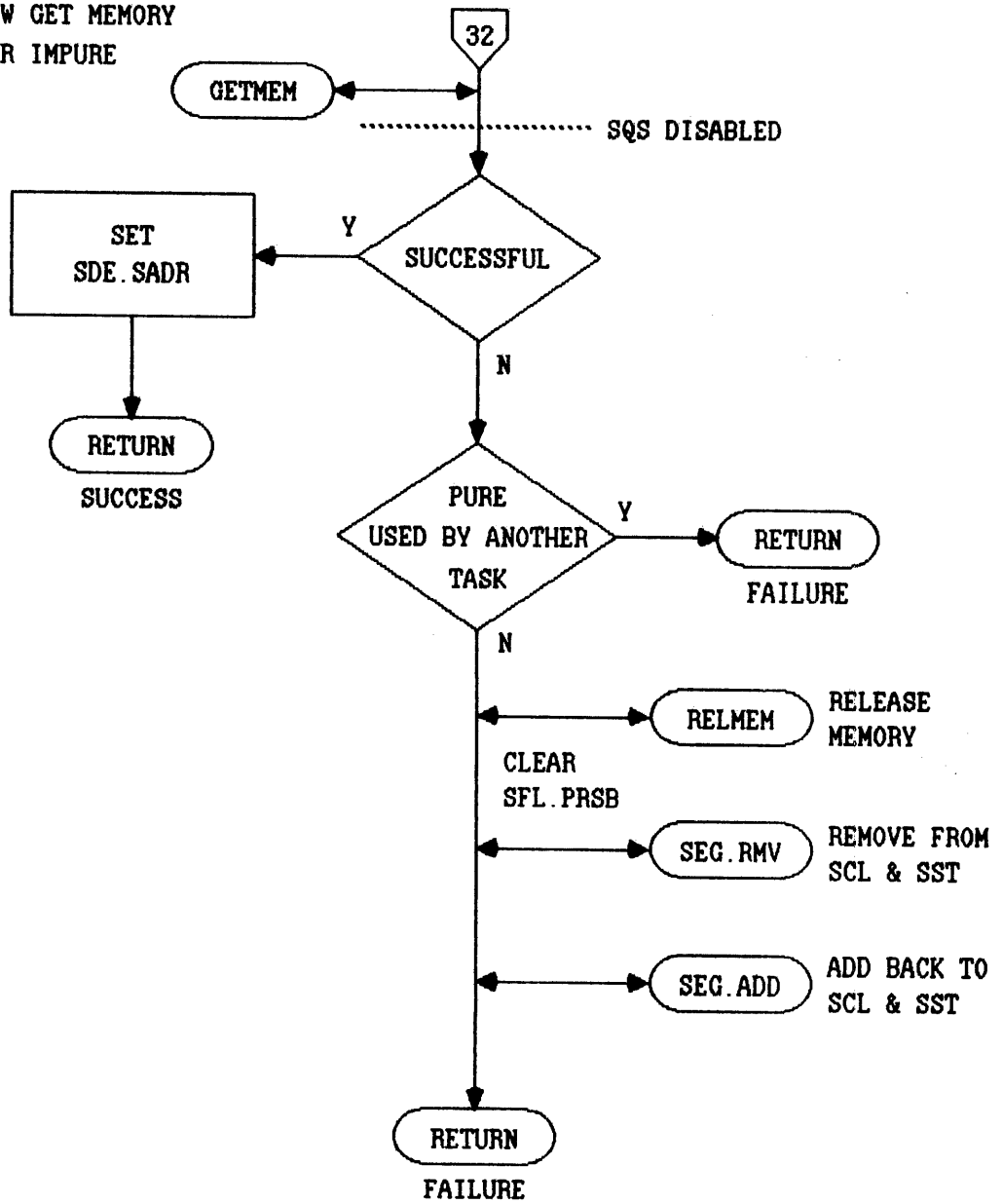
PURE SEGMENT MUST NOW
BE LOADED ALSO

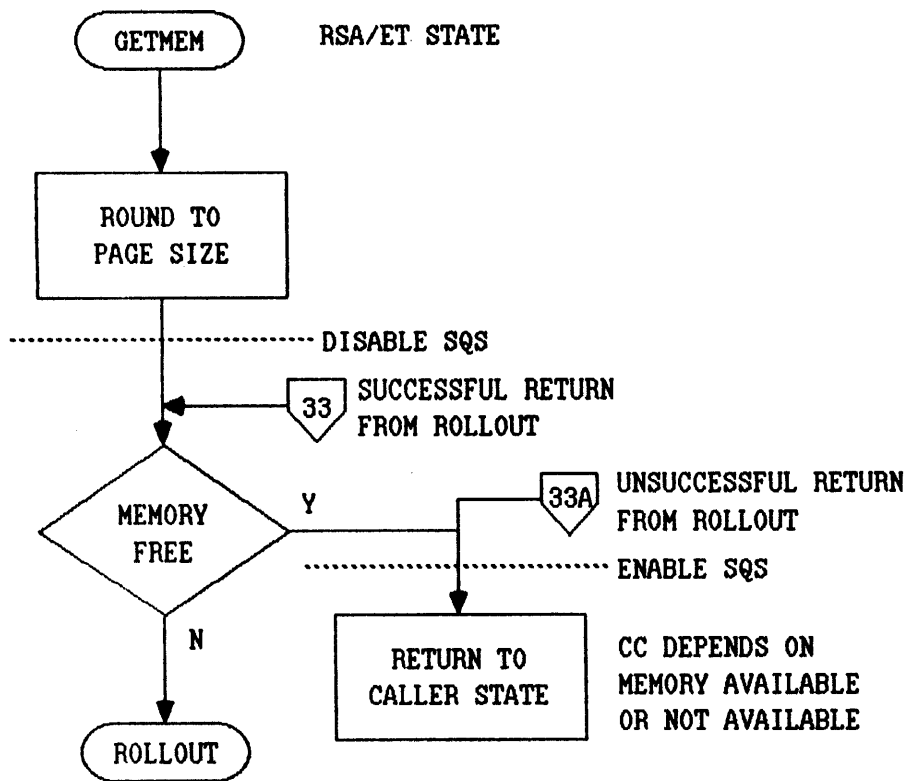
HAVE MEMORY FOR IMPURE
SEGMENT CHECK PURE

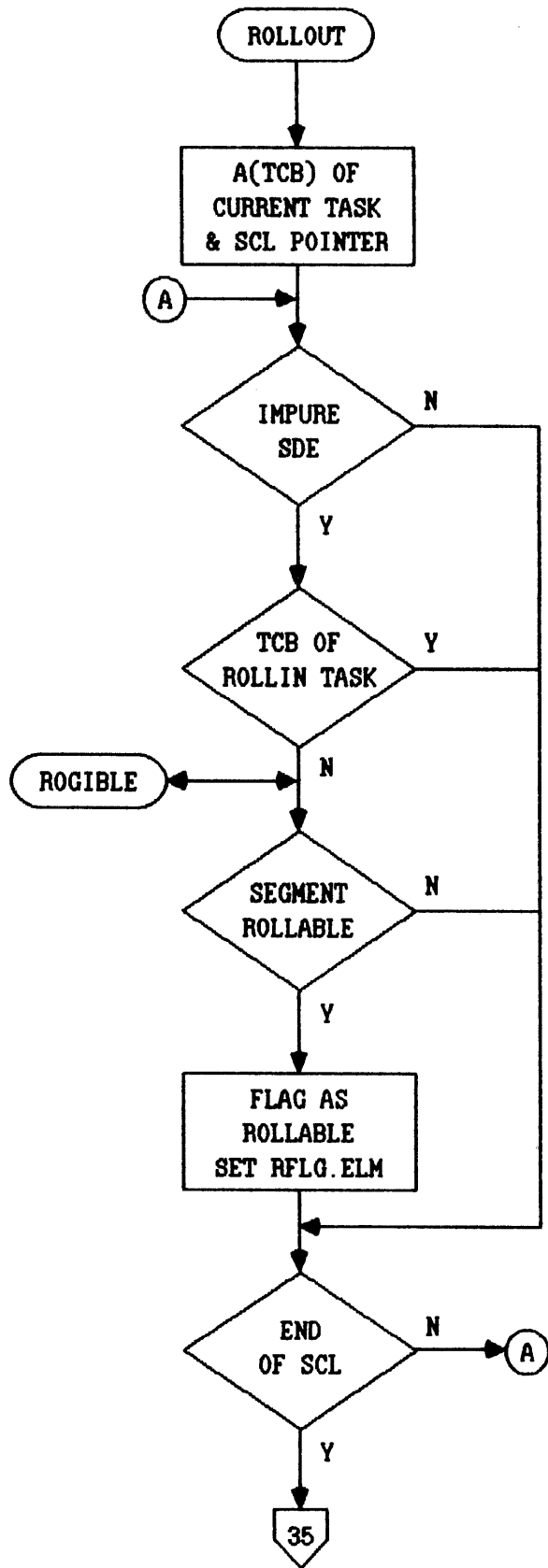


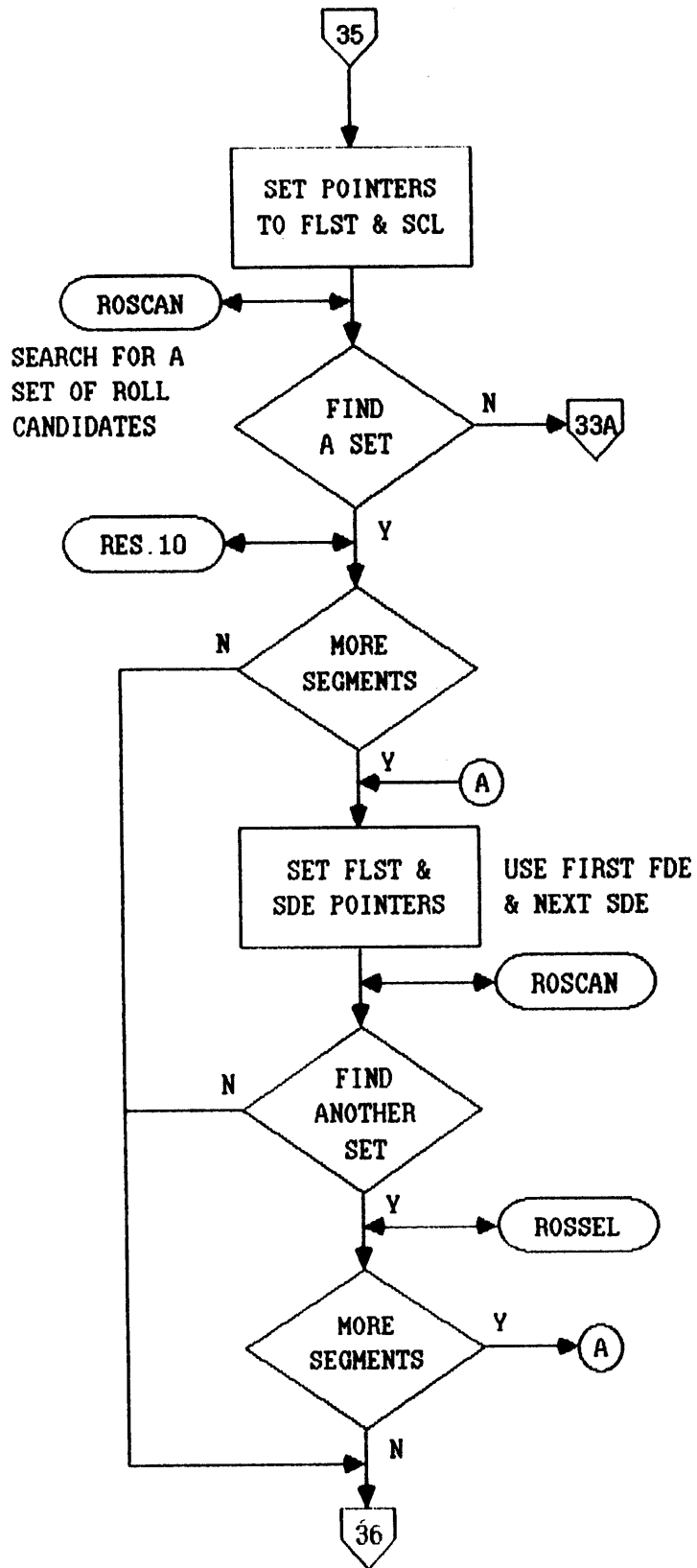


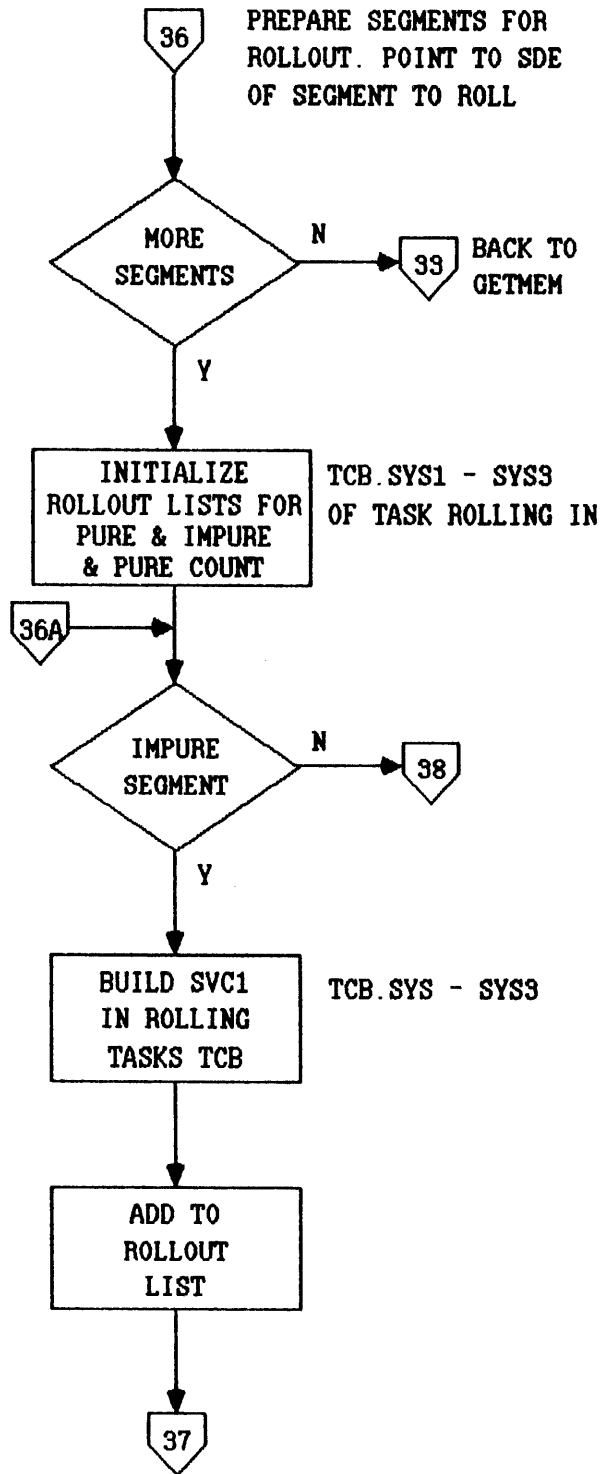
HAVE PURE SEGMENT
NOW GET MEMORY
FOR IMPURE

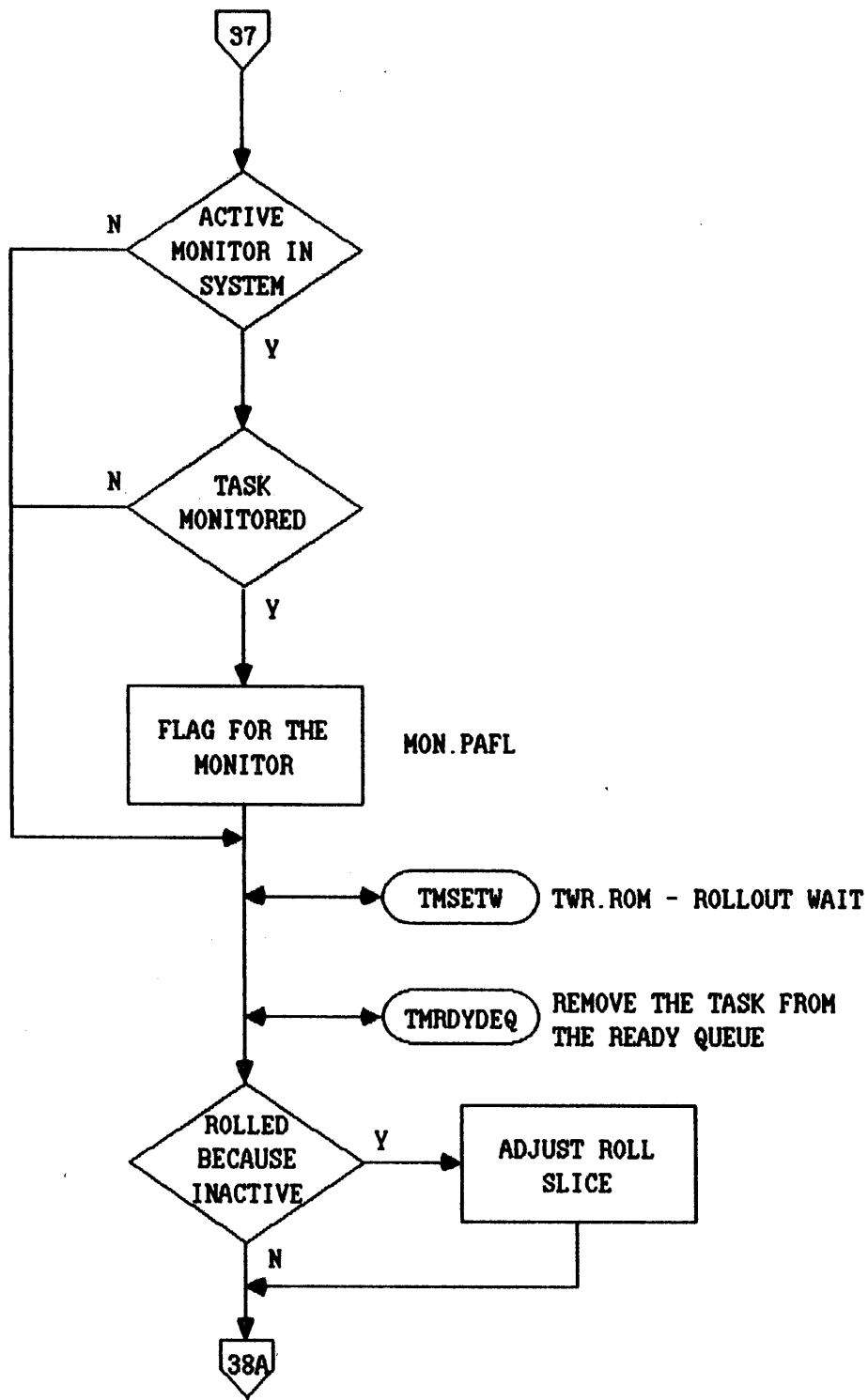


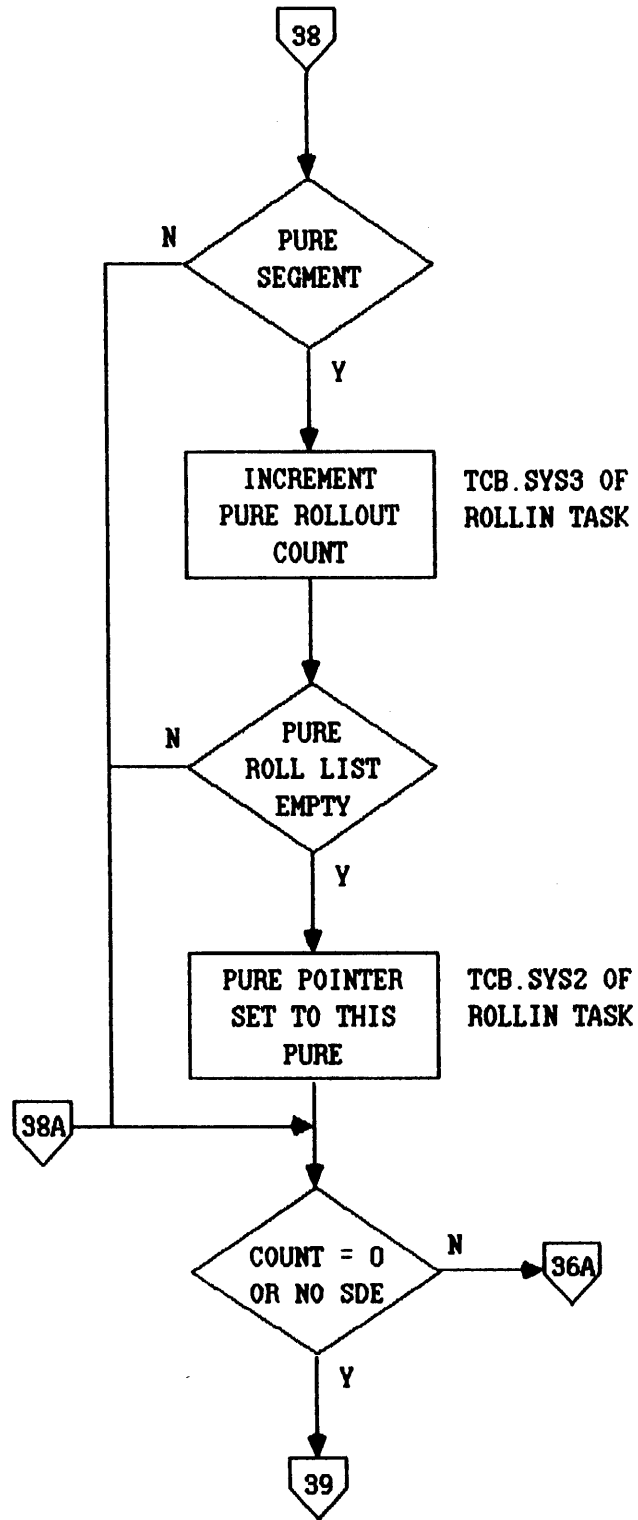


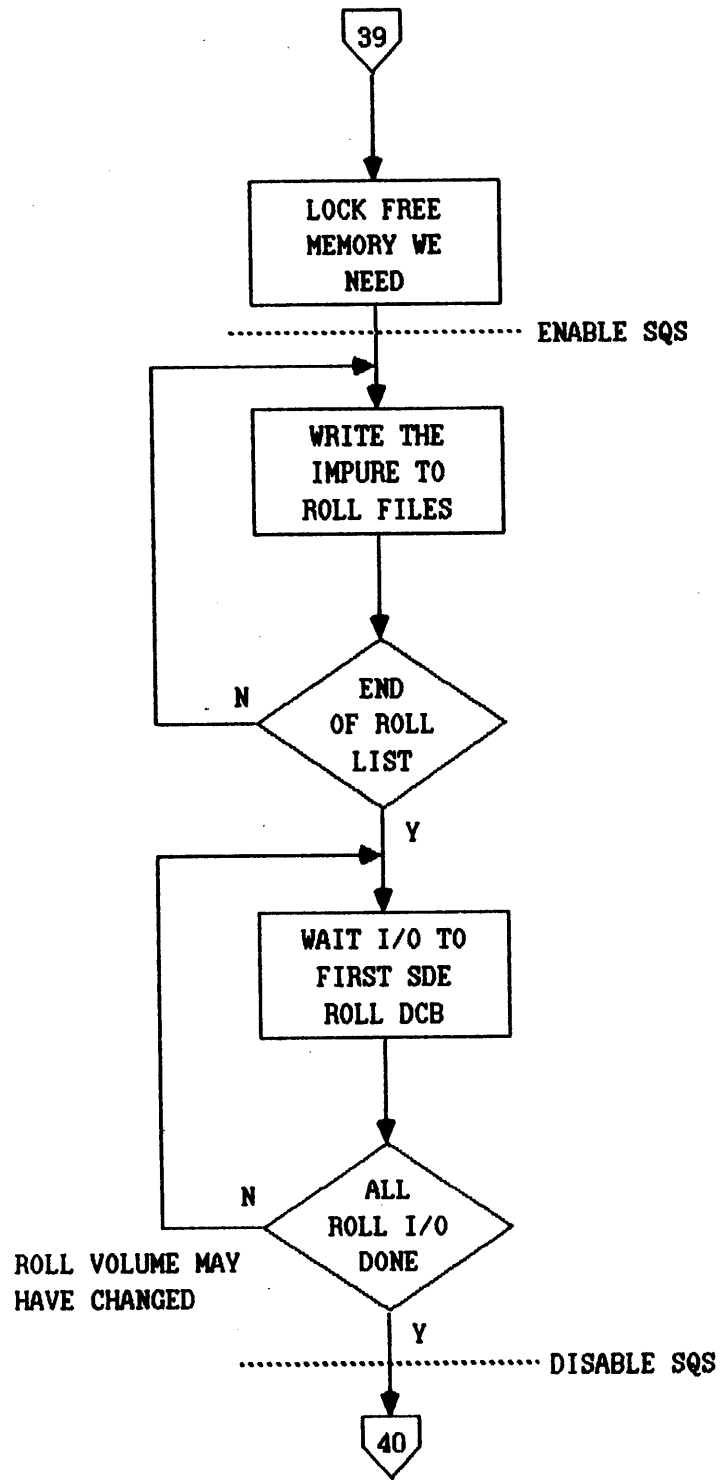


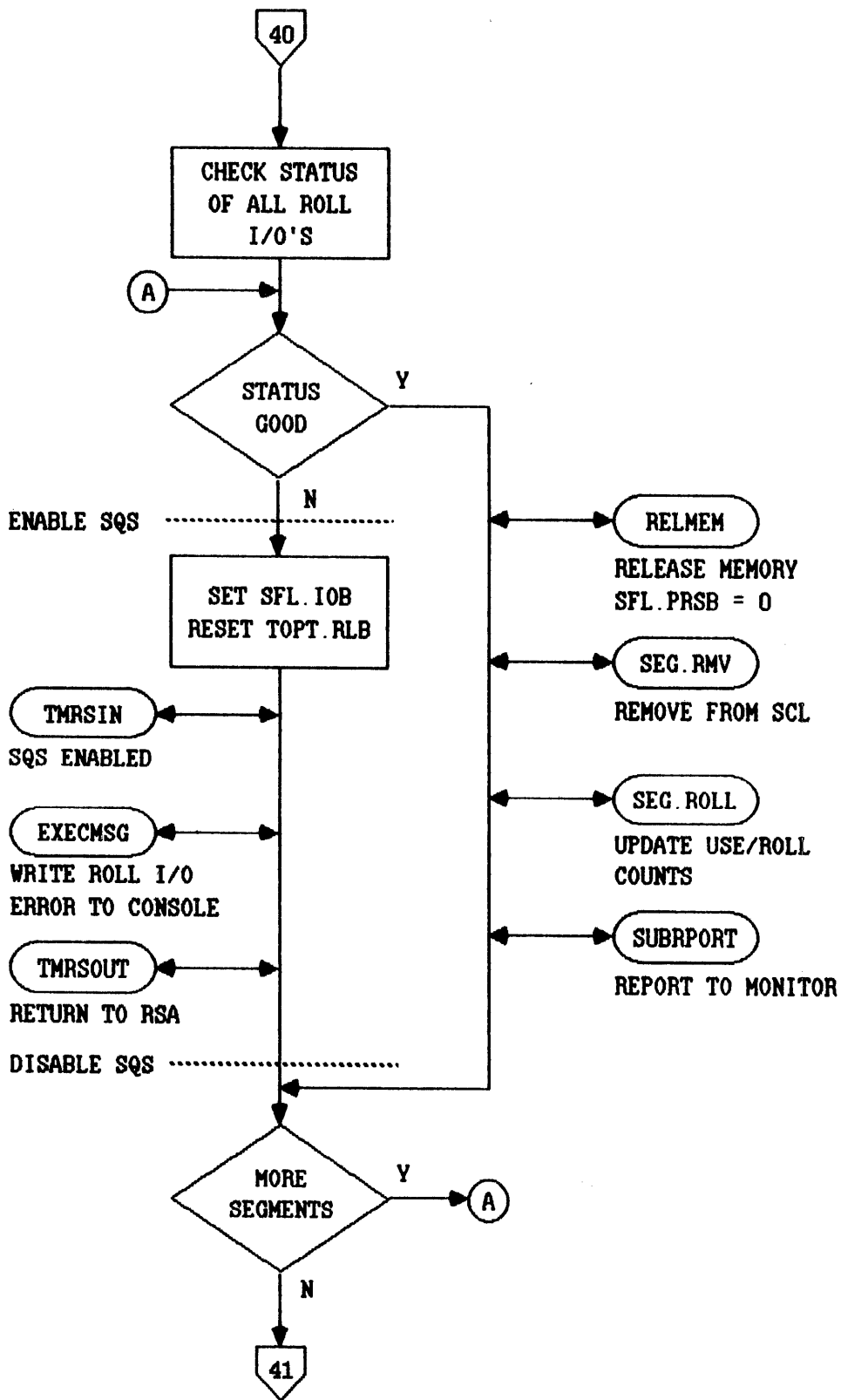


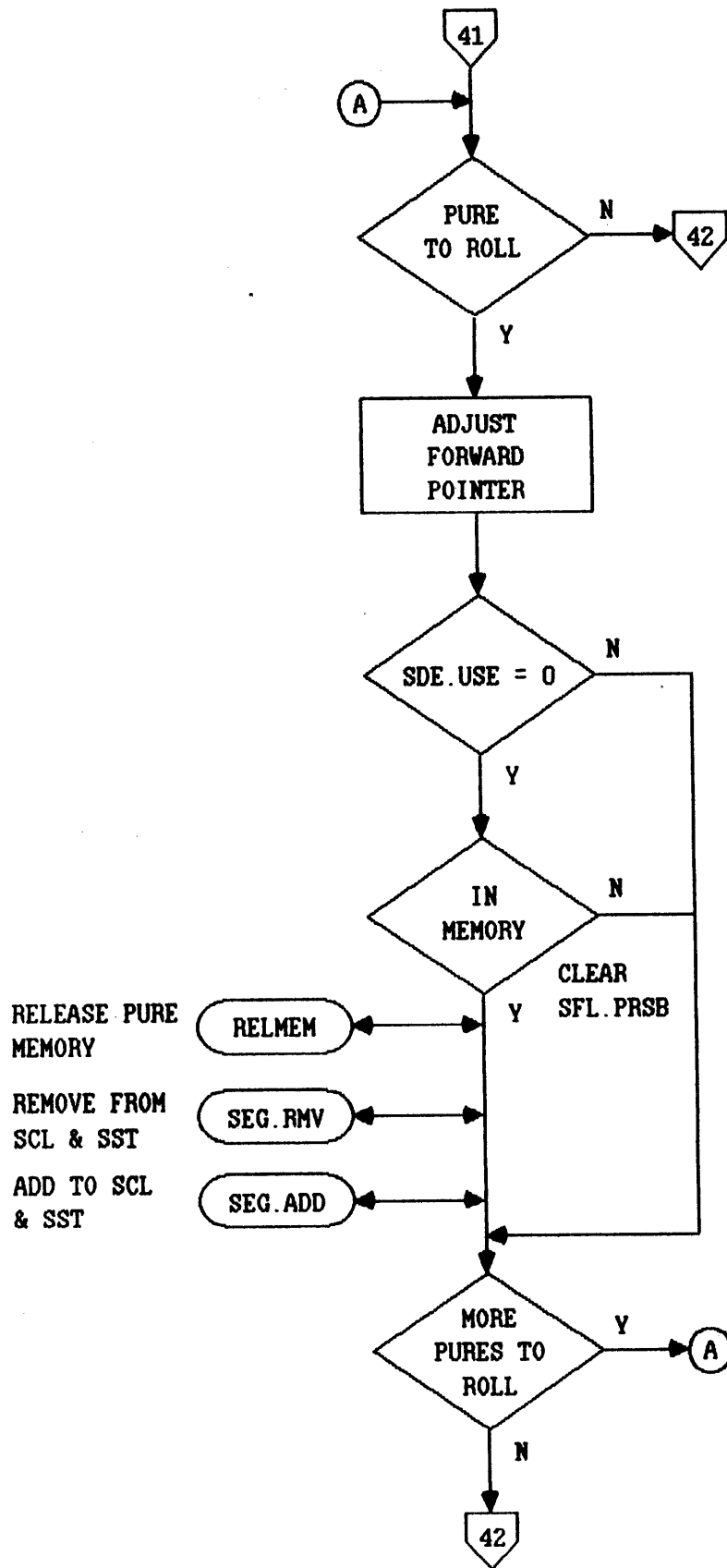


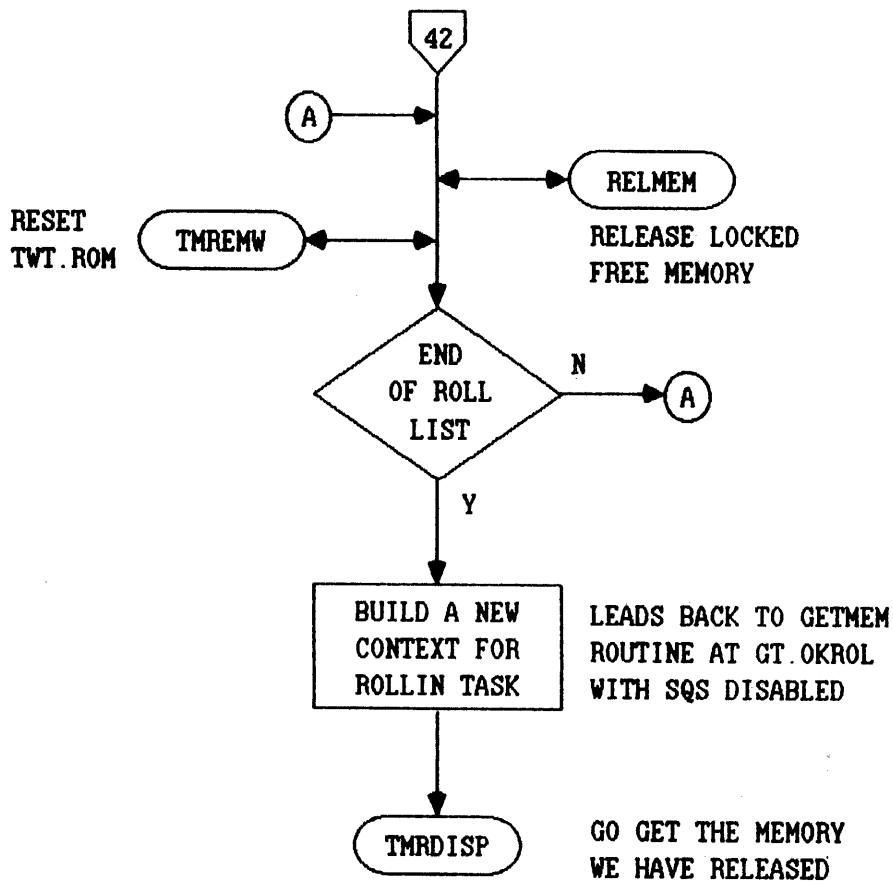




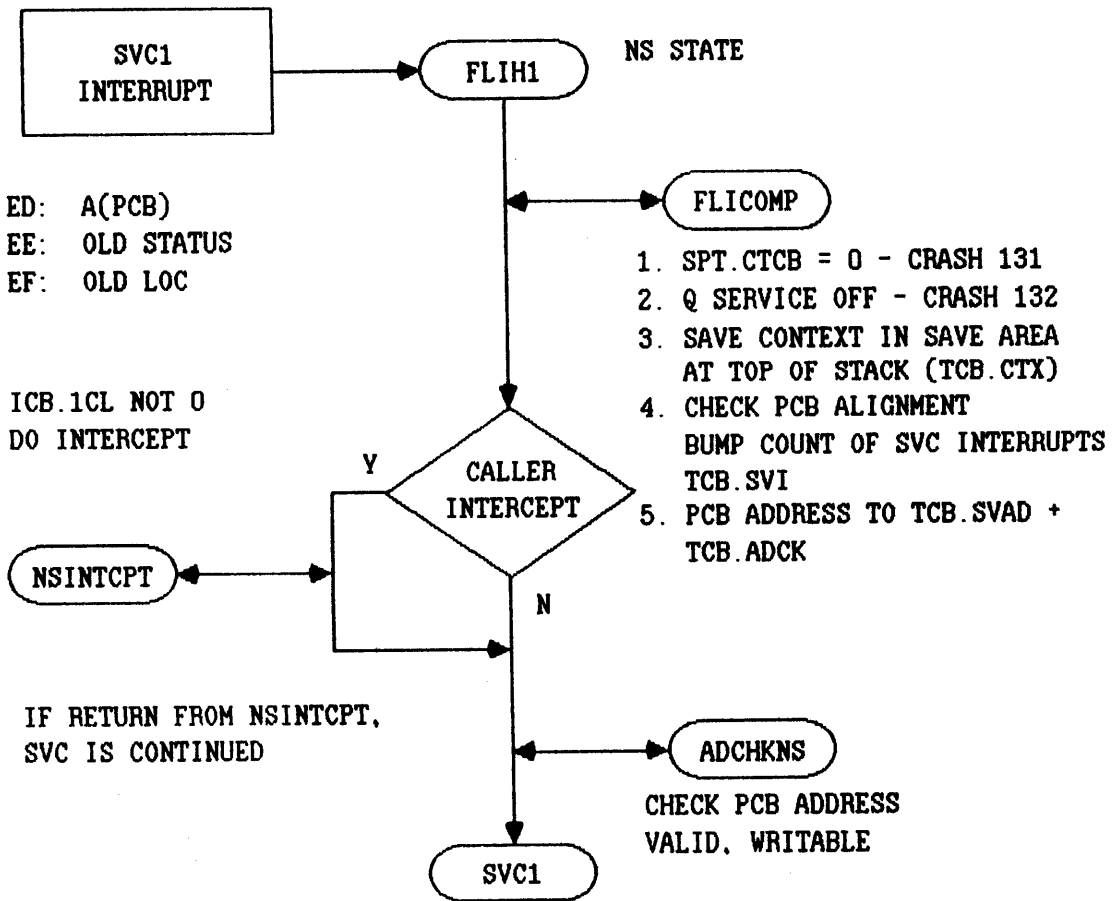


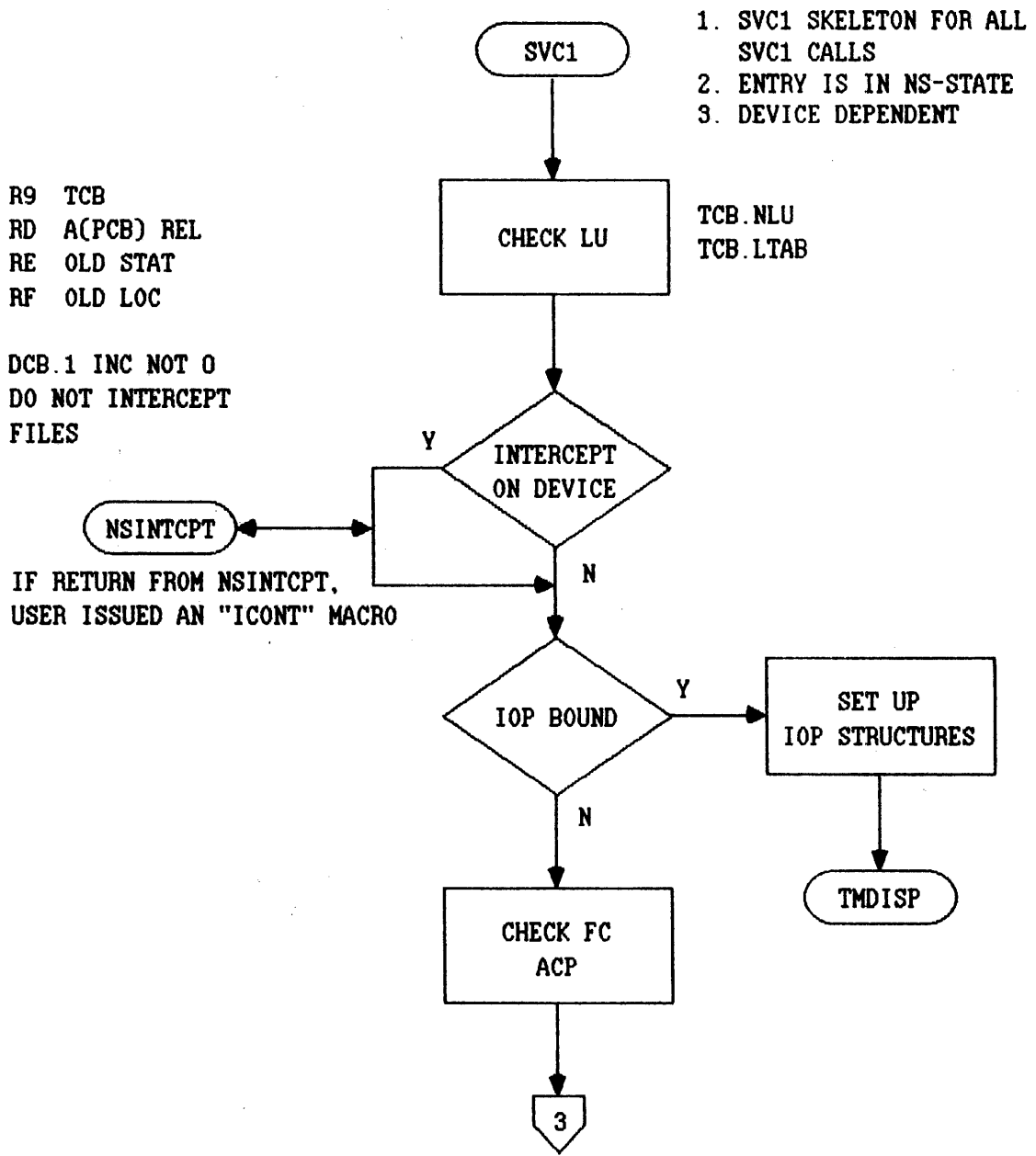


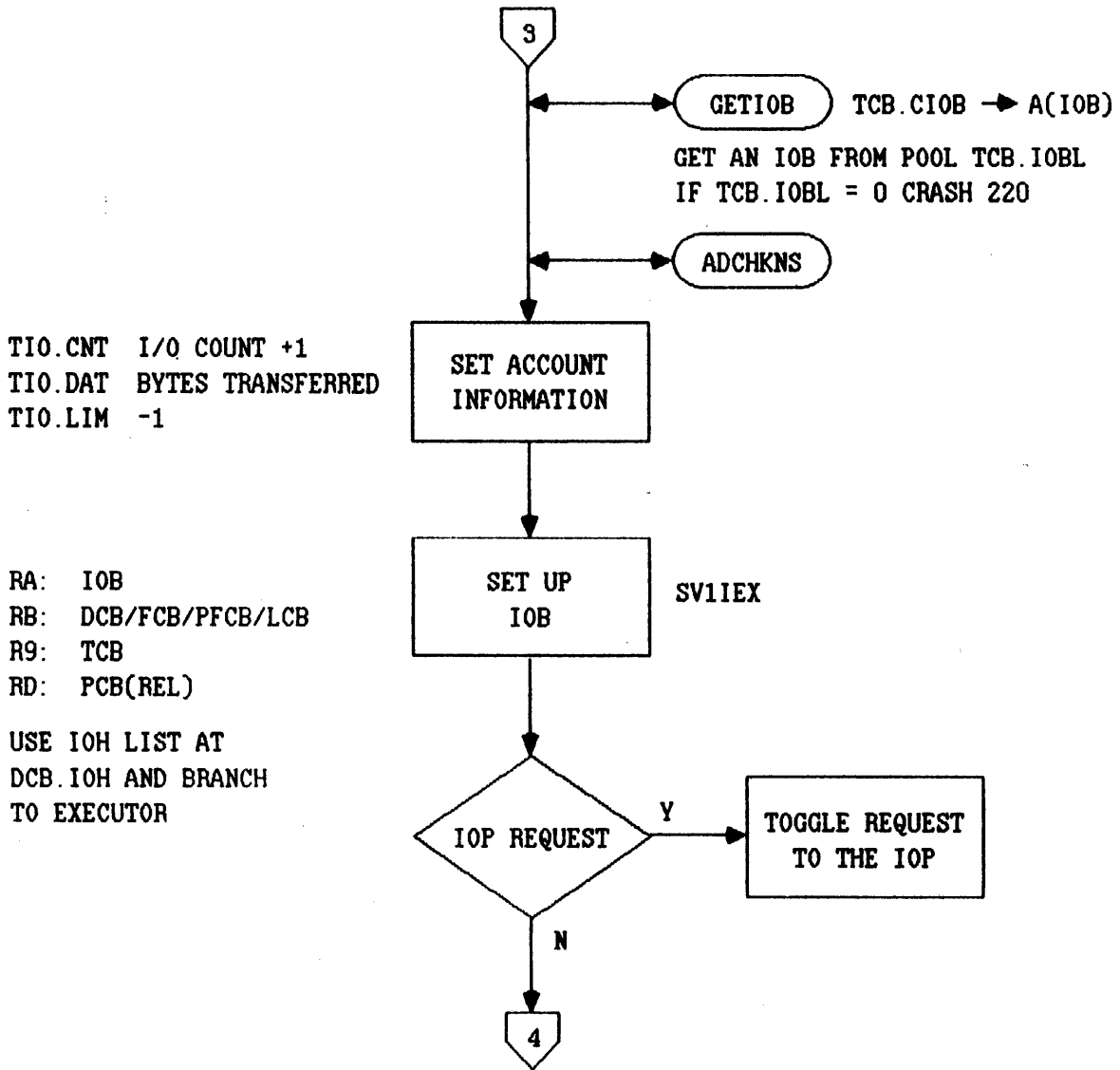


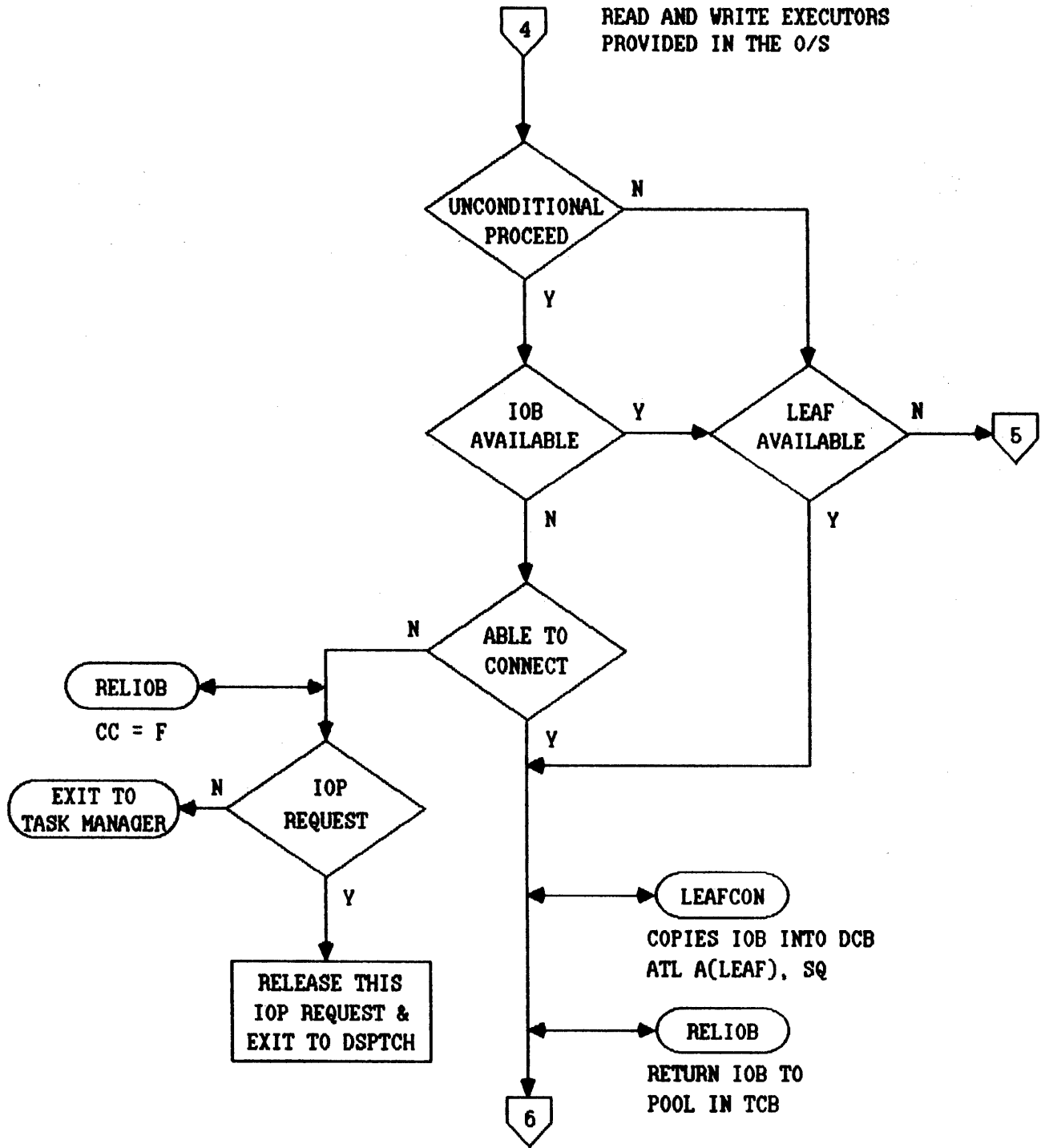


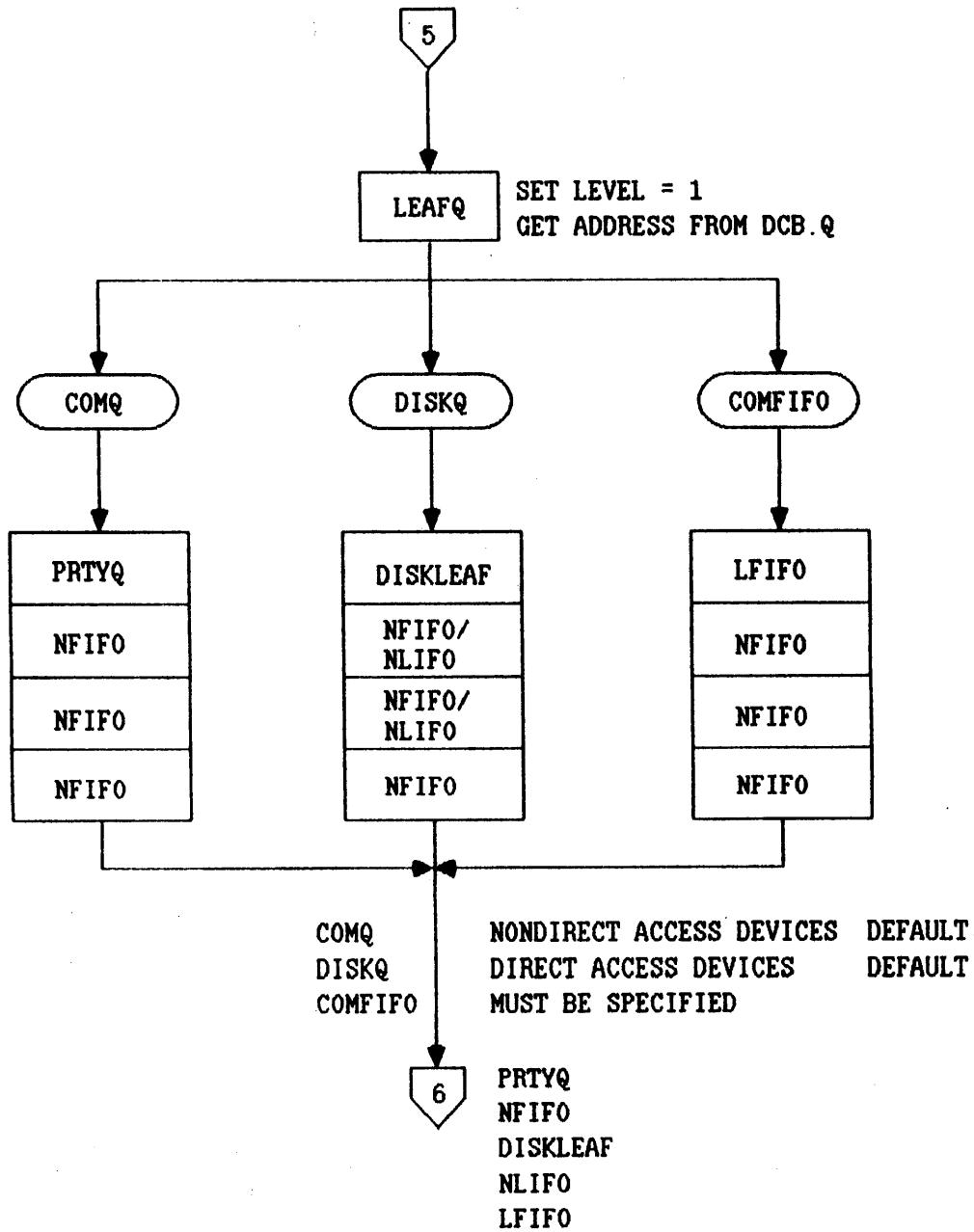
I/O SUBSYSTEM

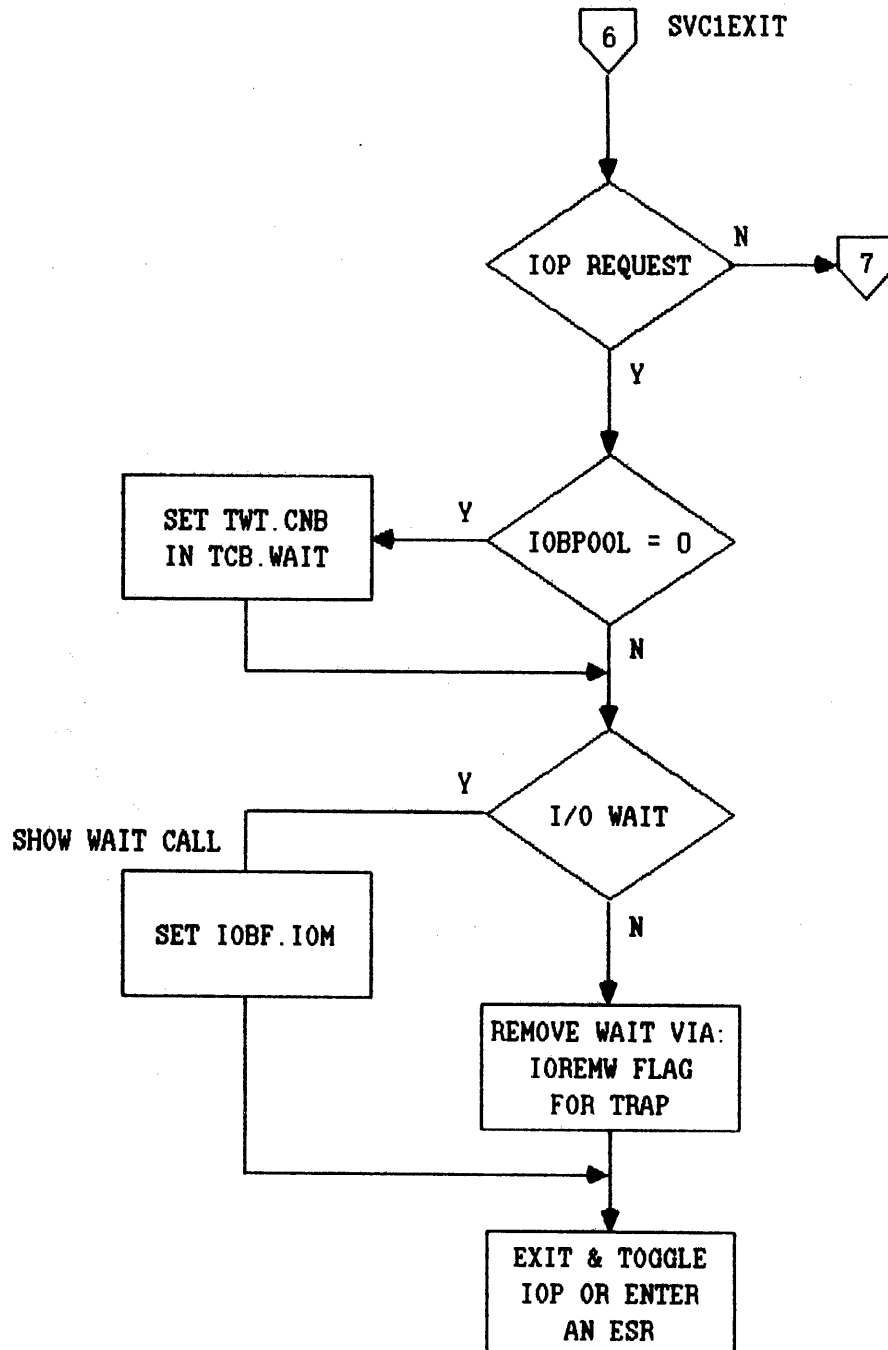


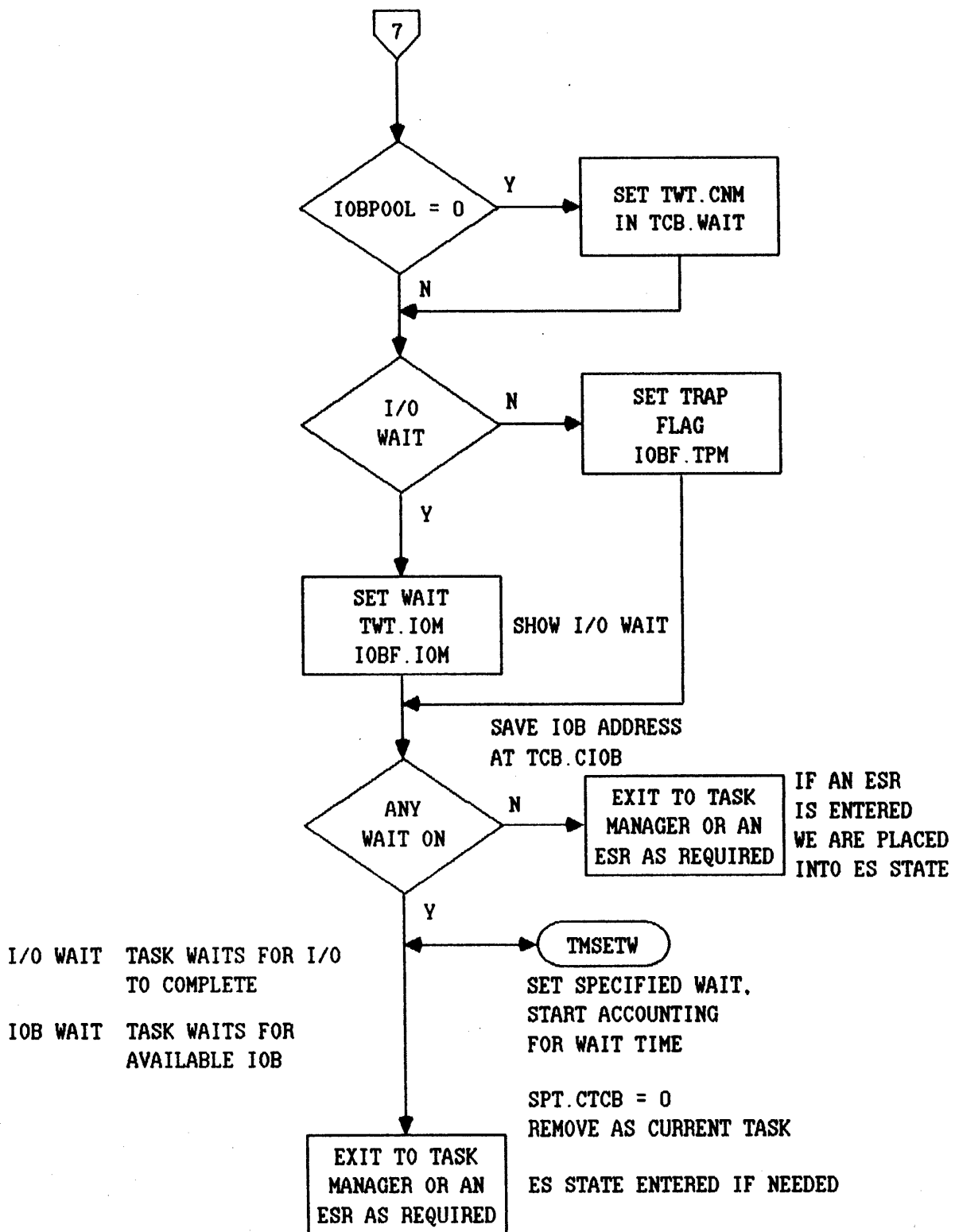


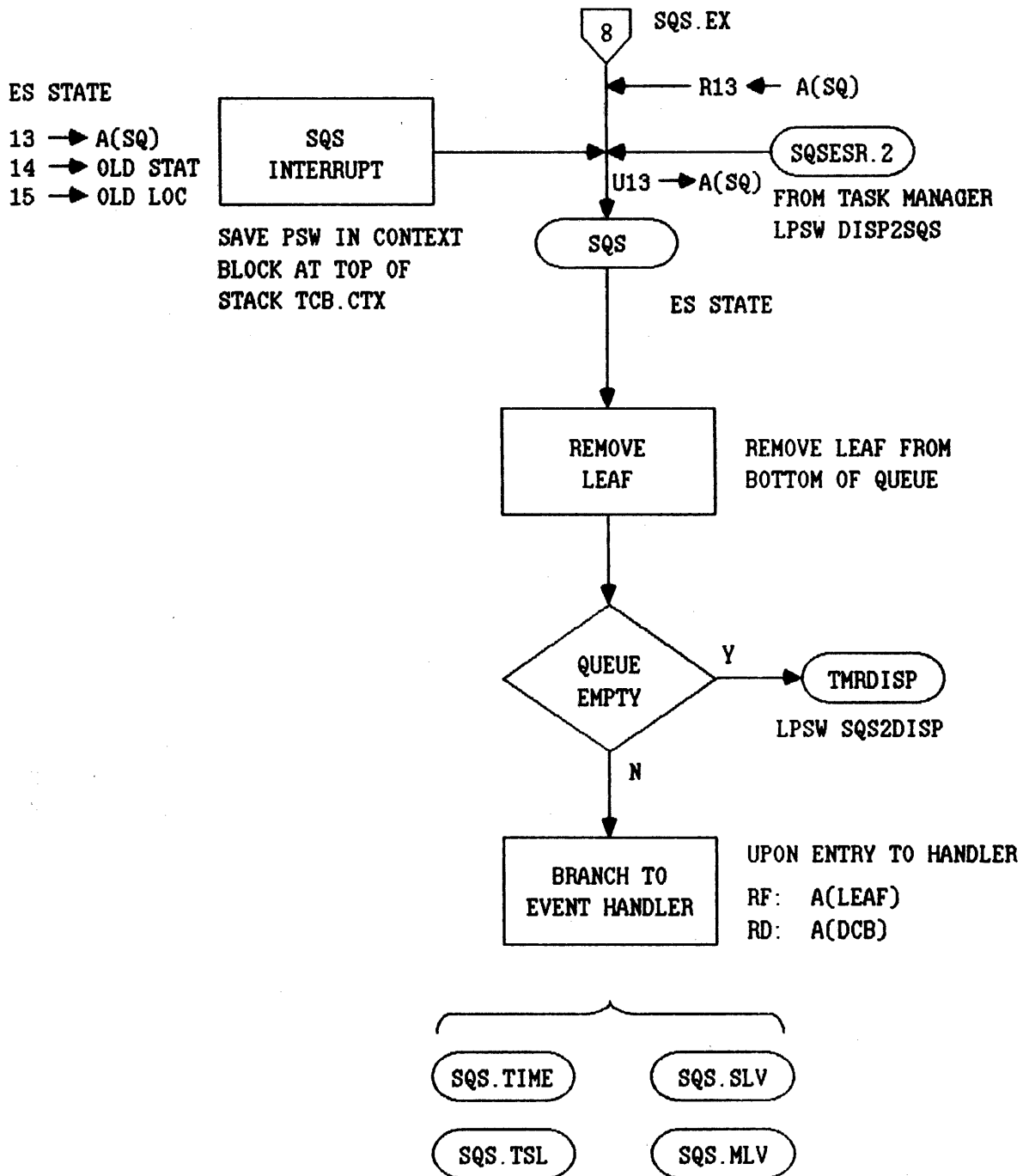


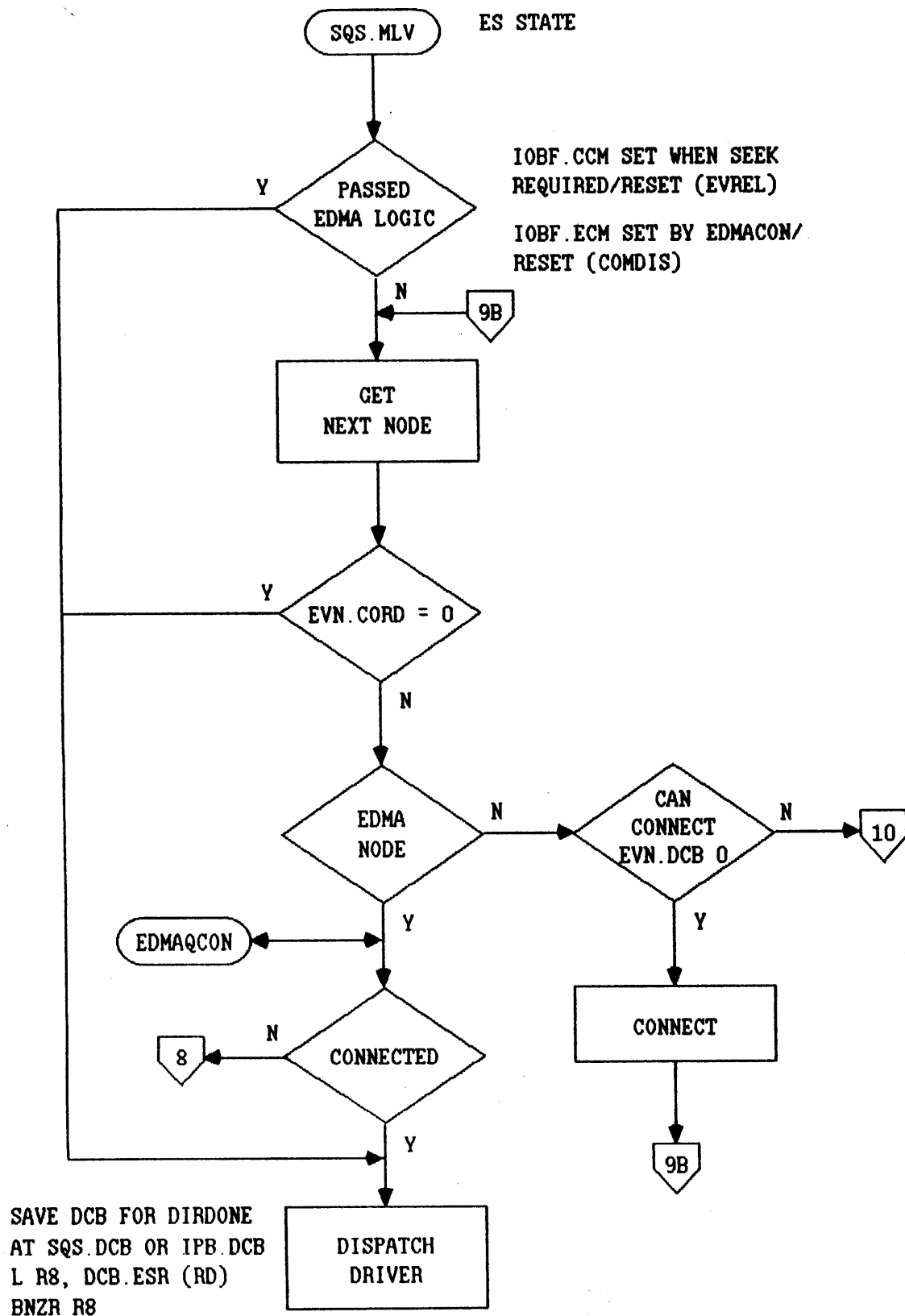


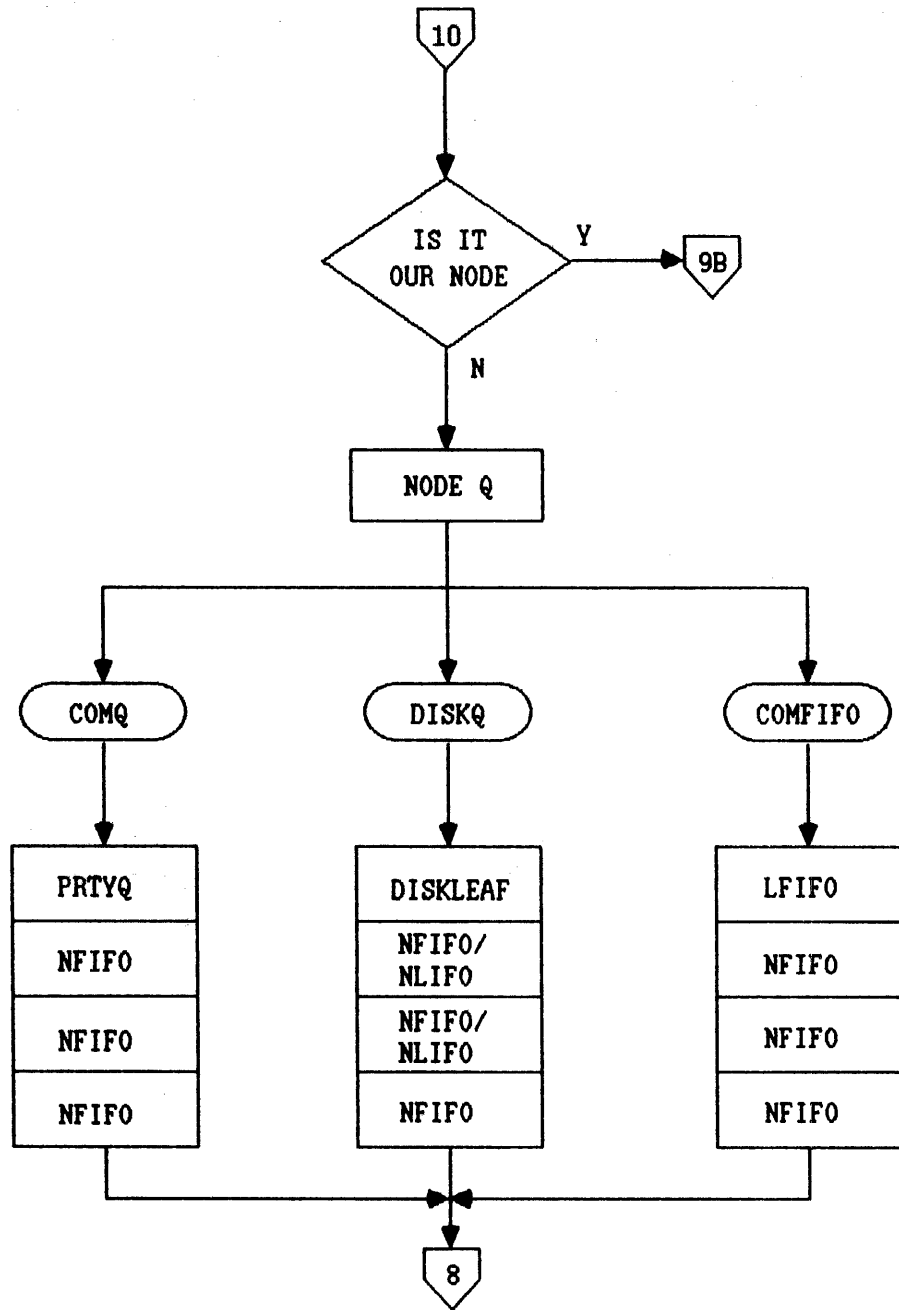


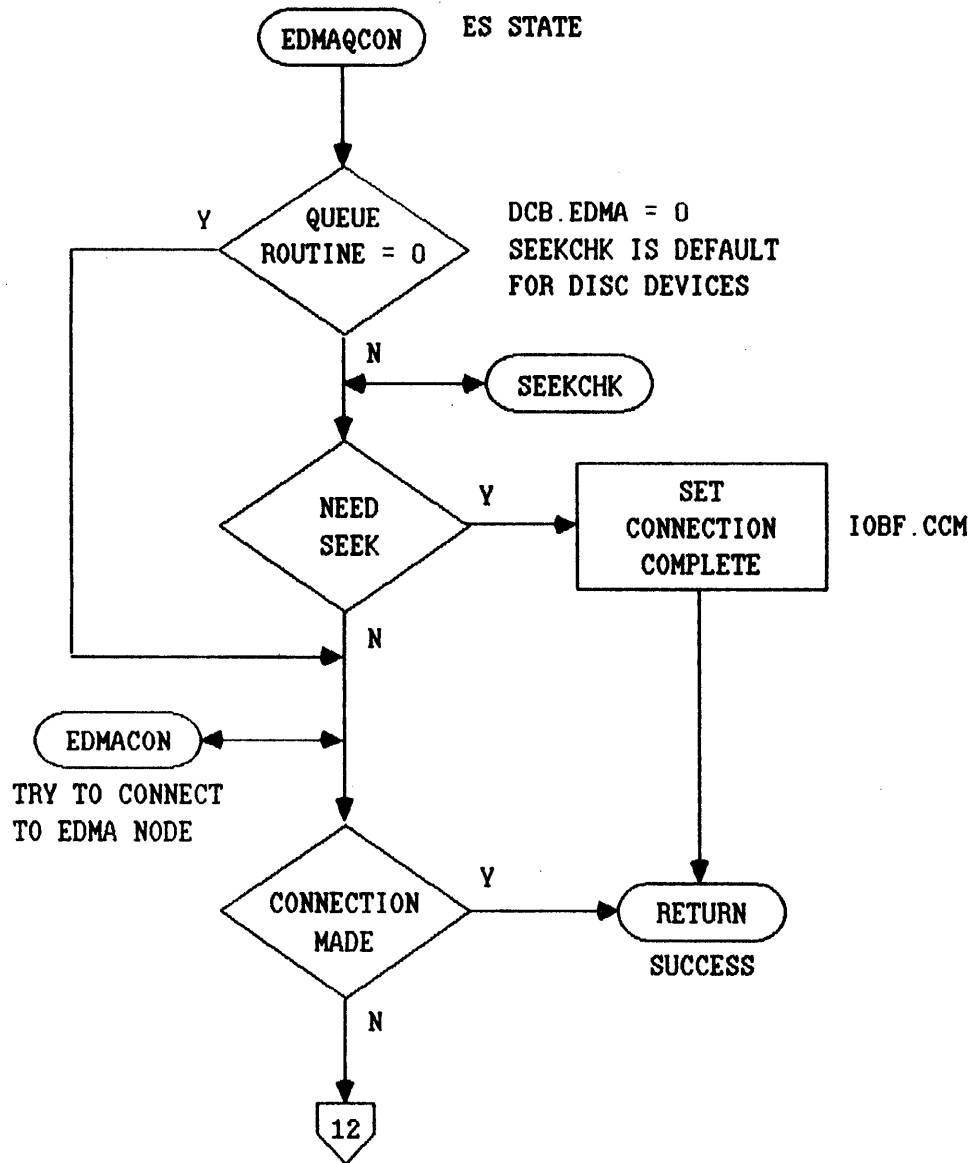


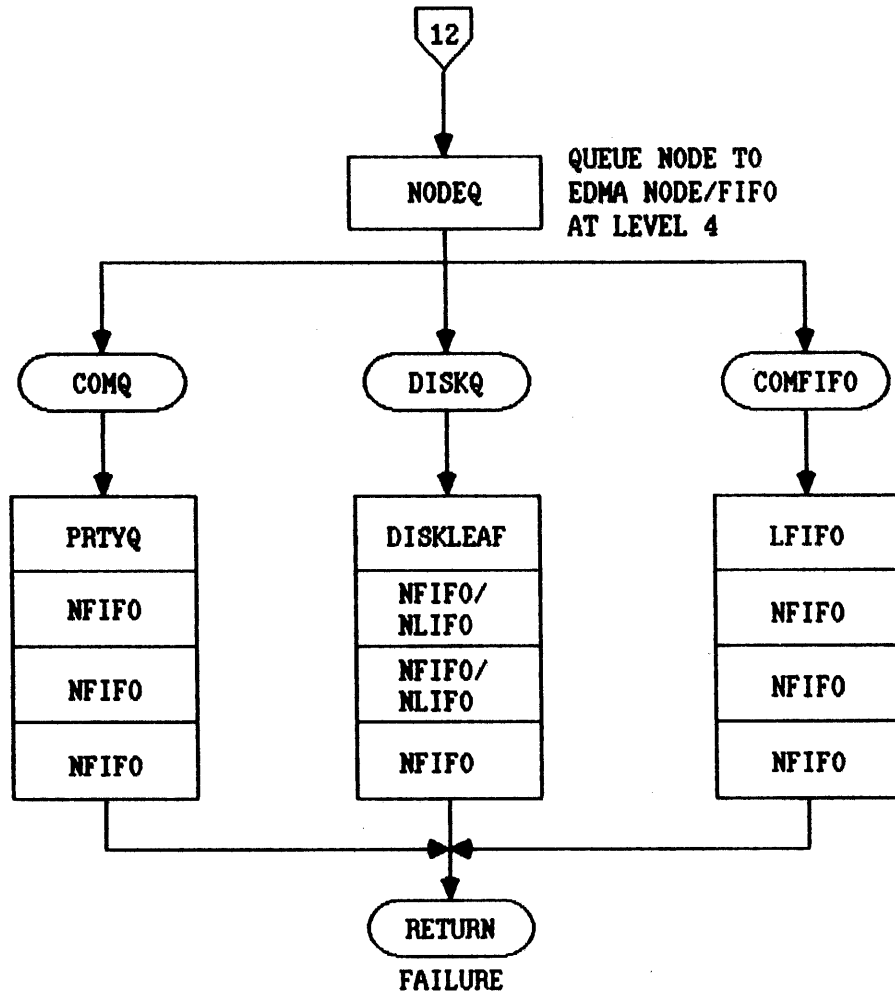


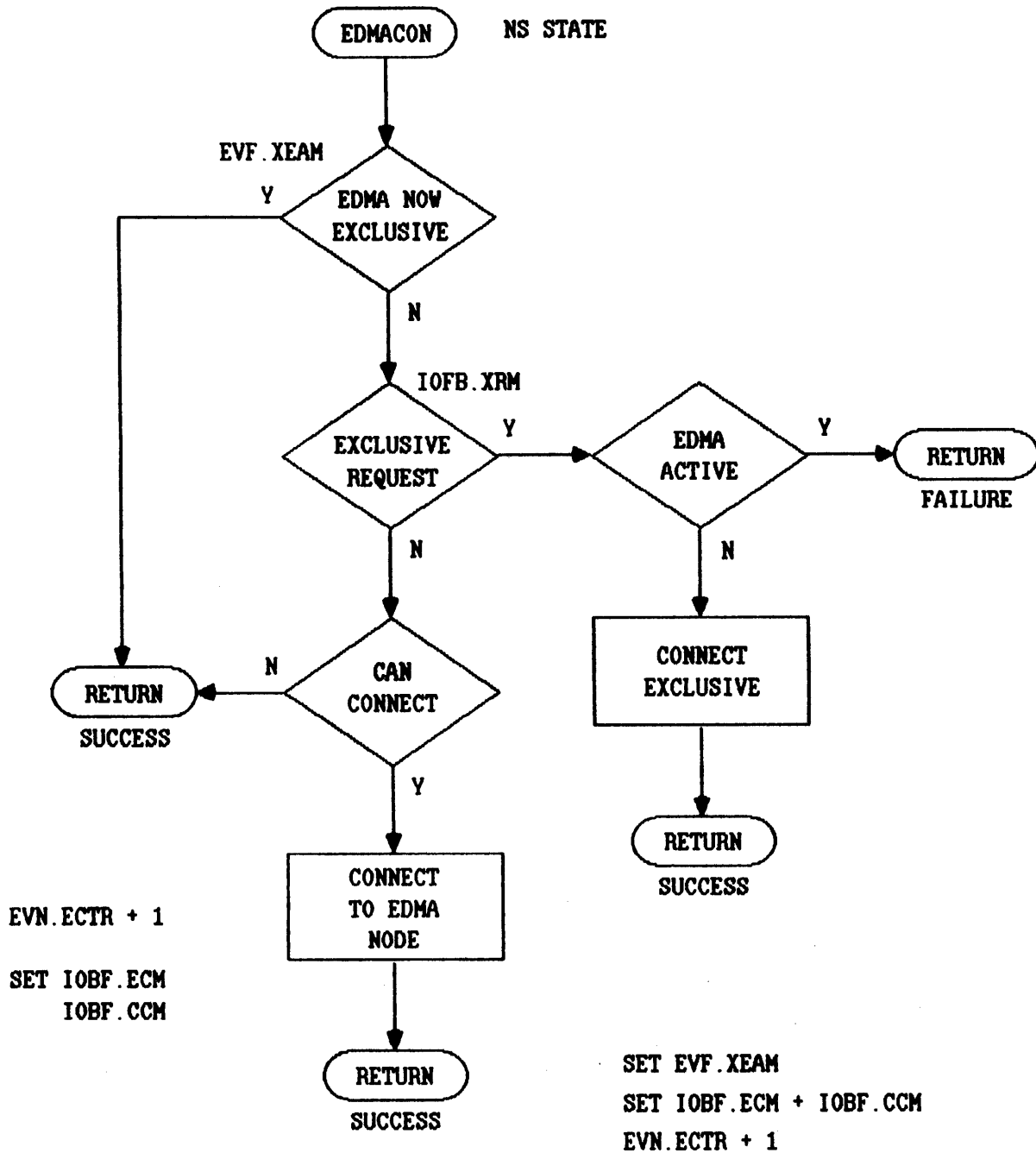




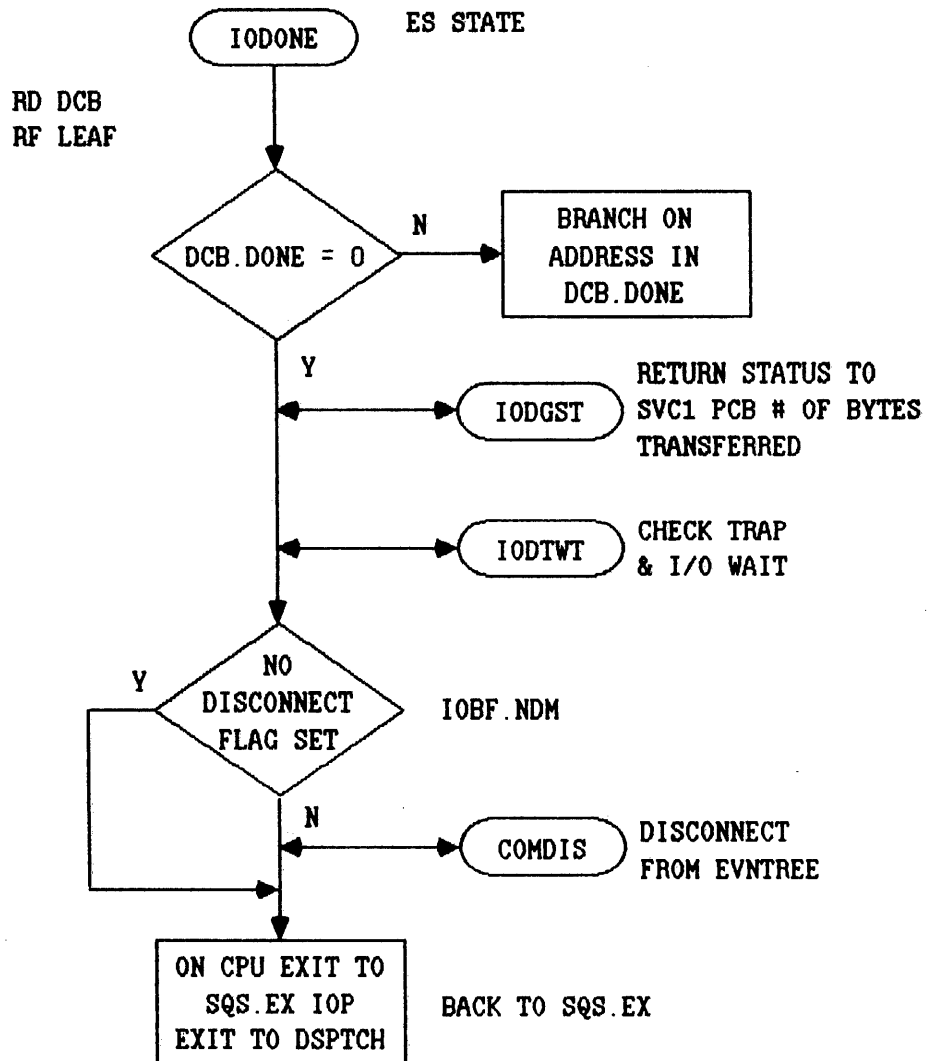


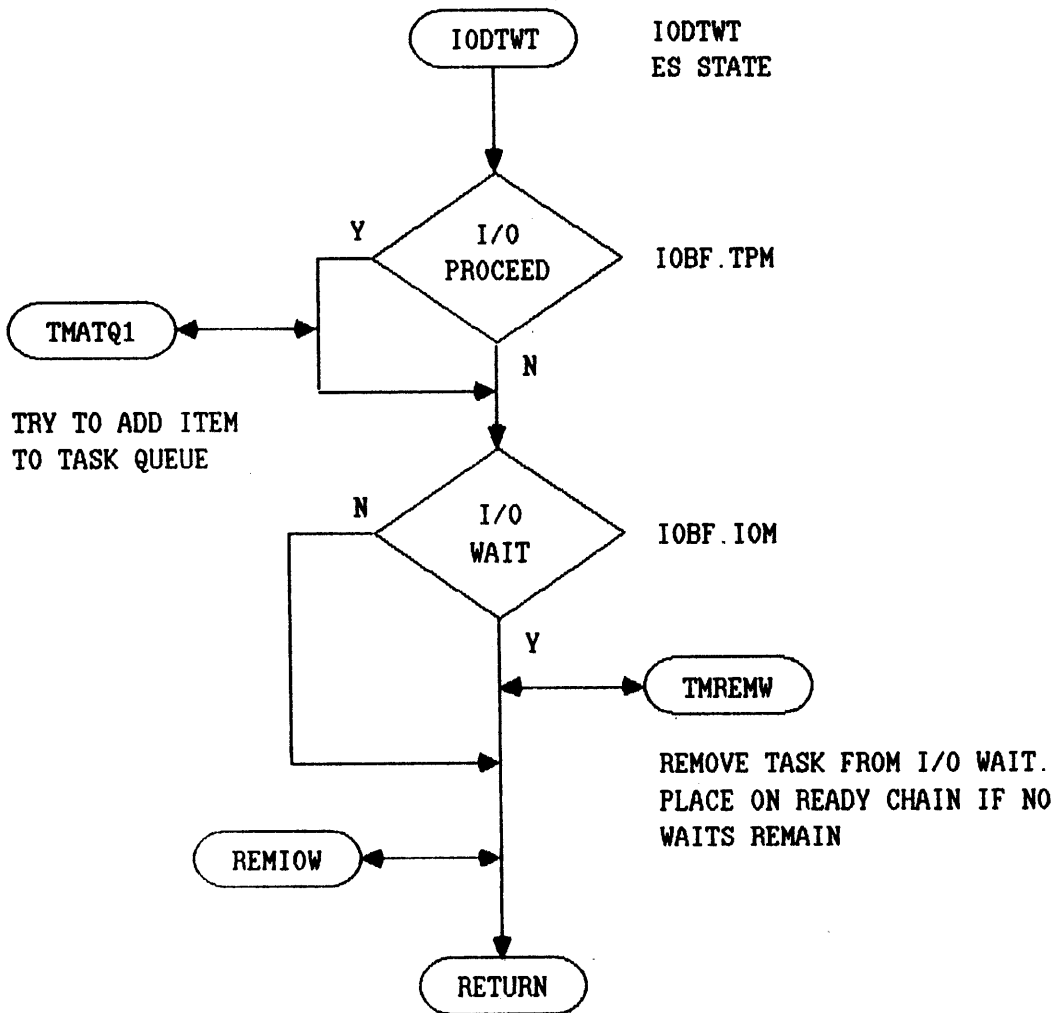




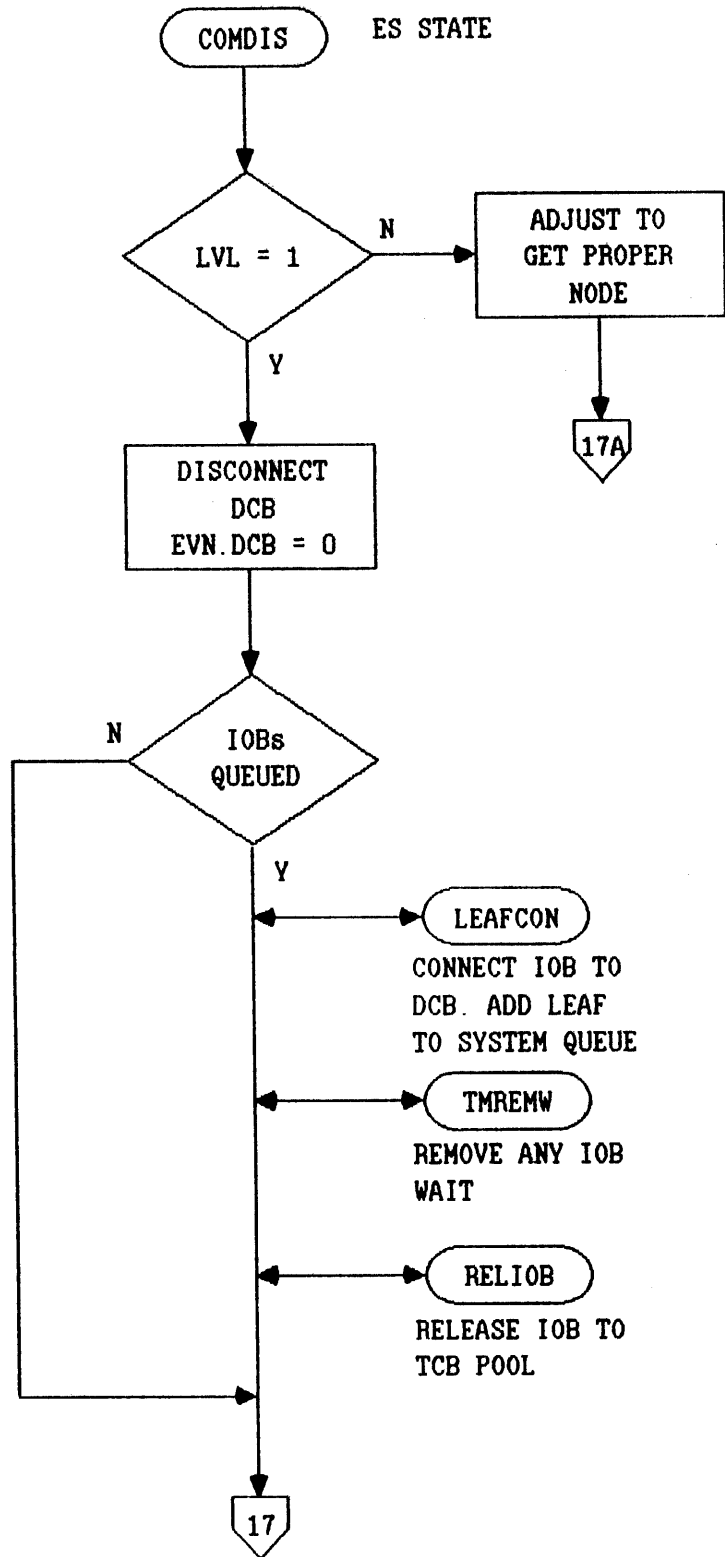


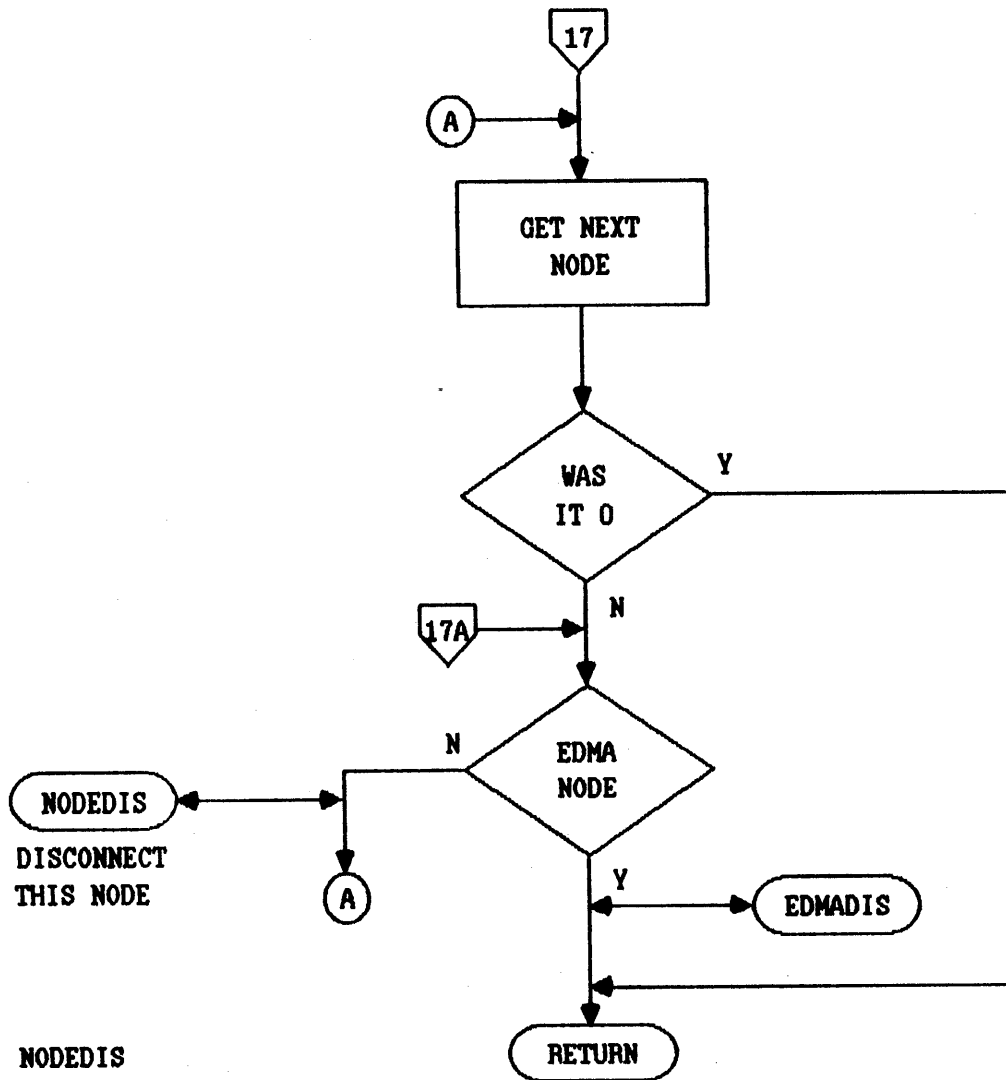
EVN.EMAX MAX NUMBER OF CONNECTIONS
 EVN.ECTR # OF CONNECTIONS





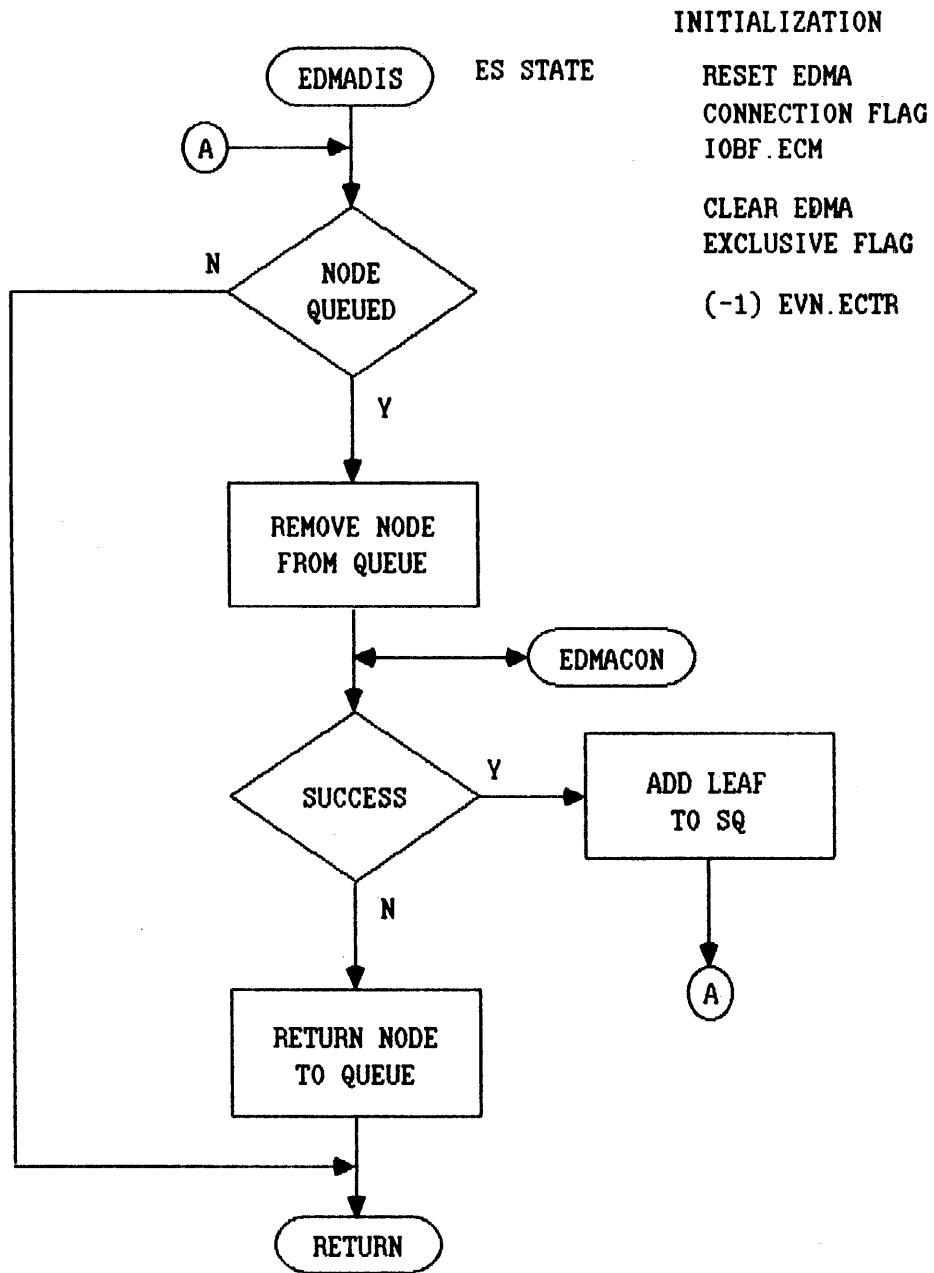
REMOVE TASKS WAITING ON
ANOTHER TASK'S I/O FROM
WAIT STATE



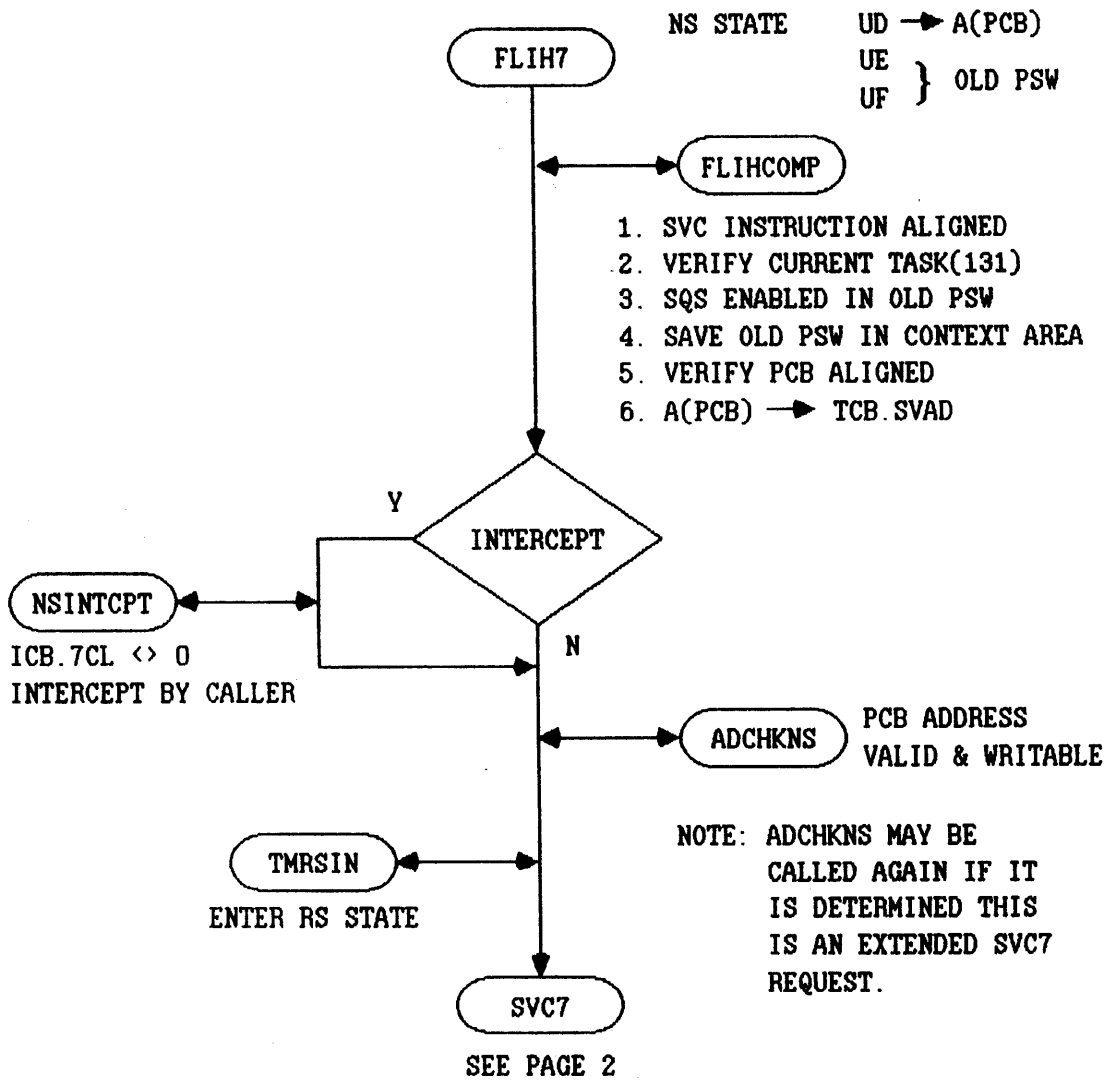


NODEDIS

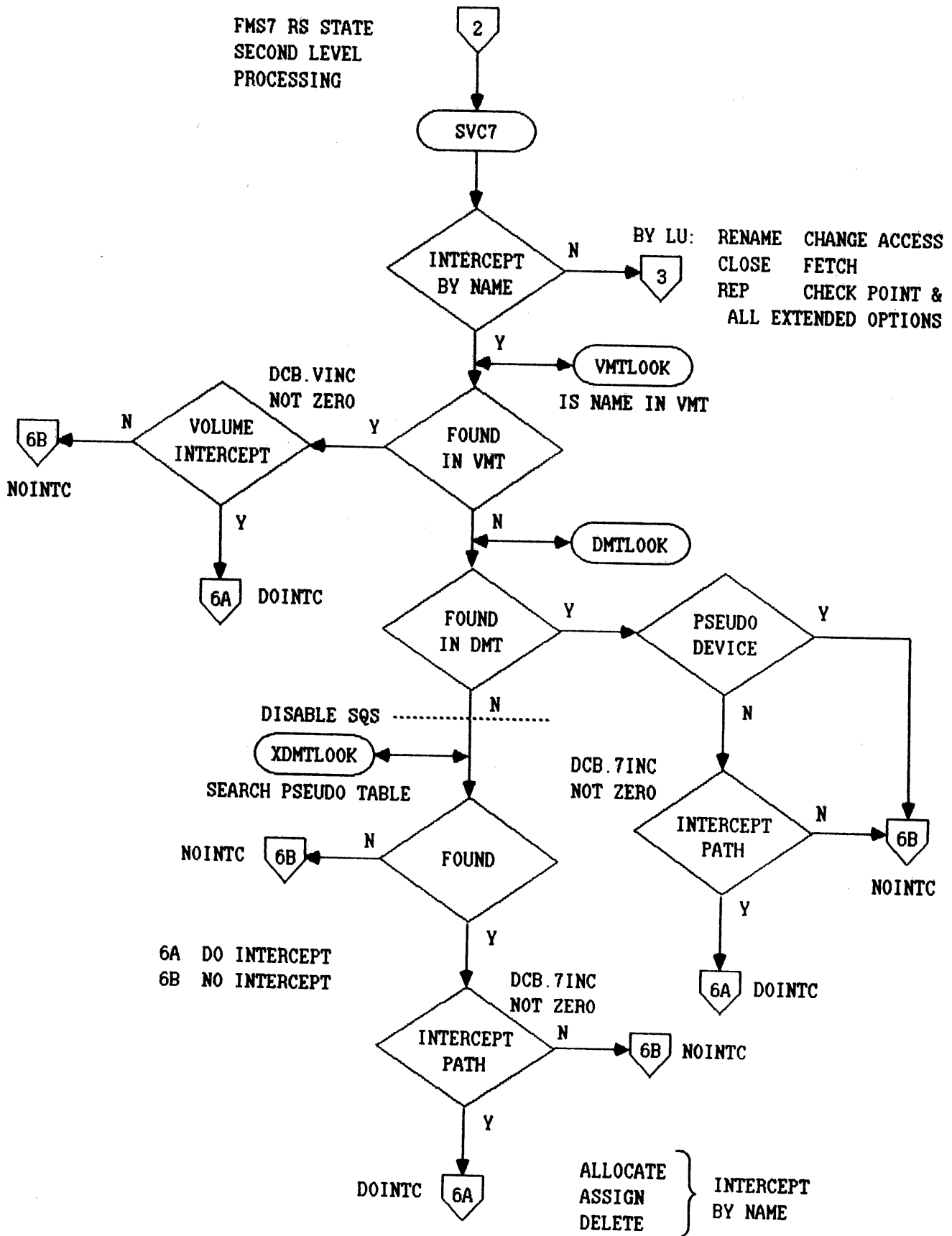
DISCONNECTS A DCB FROM A
 NODE & CONNECTS TOP OF QUEUE.
 ADDING LEAF TO SQ

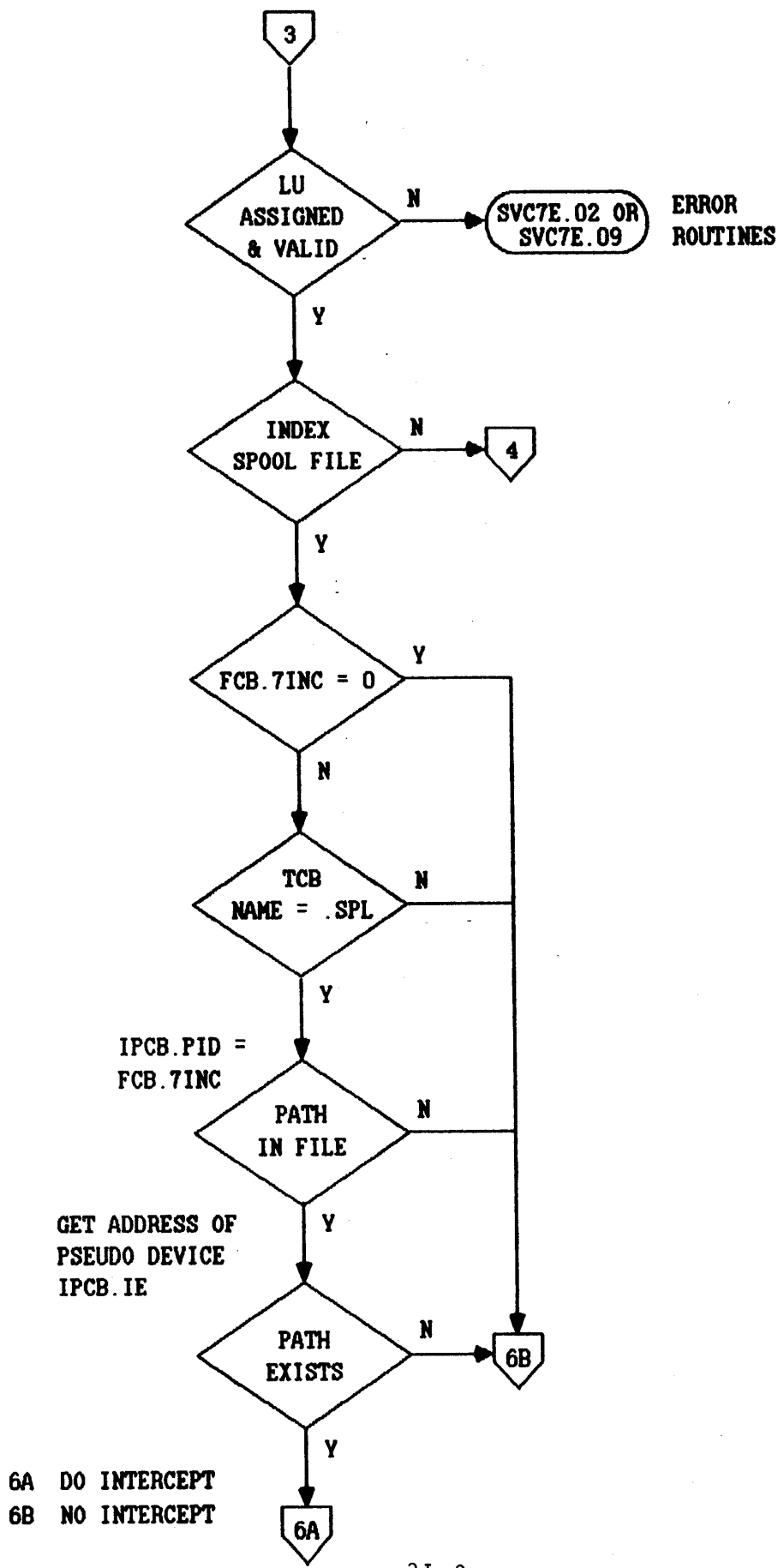


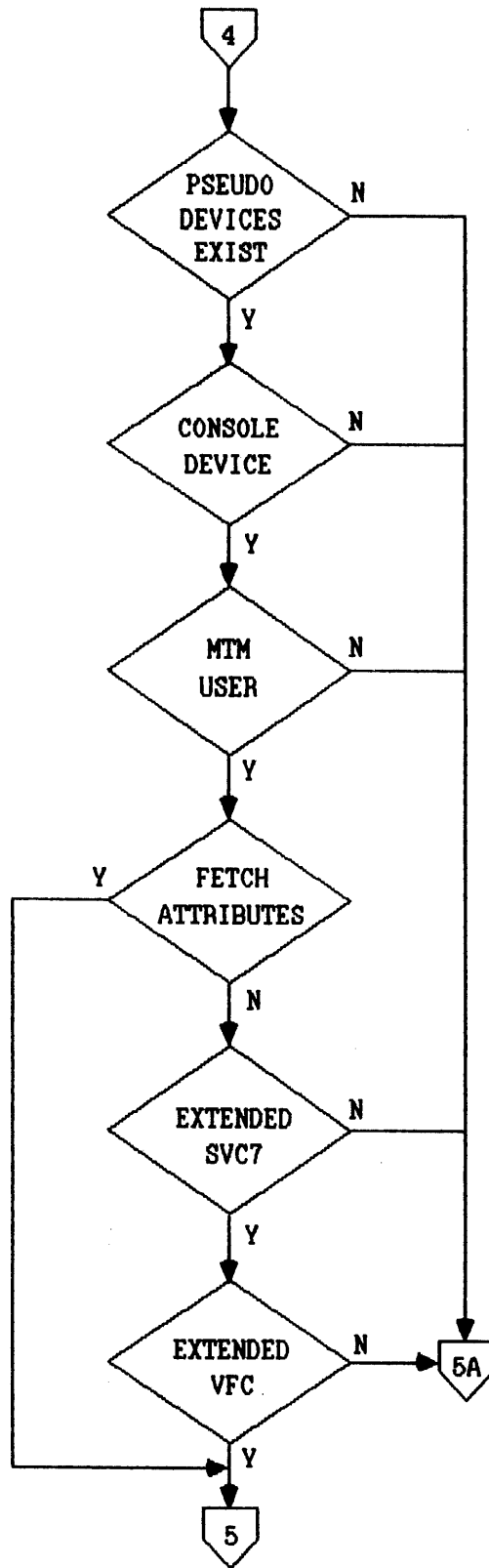
FILE MANAGER

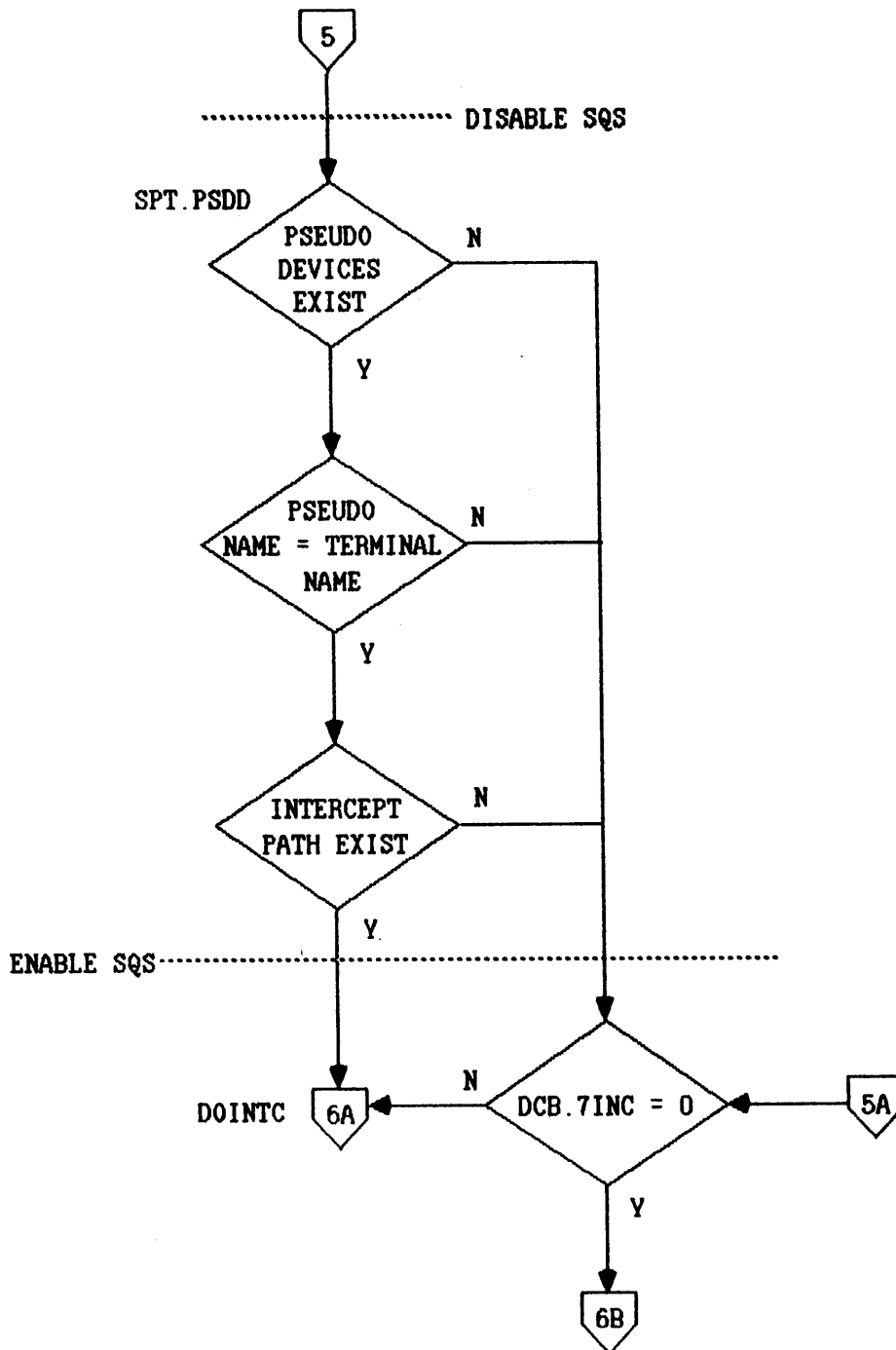


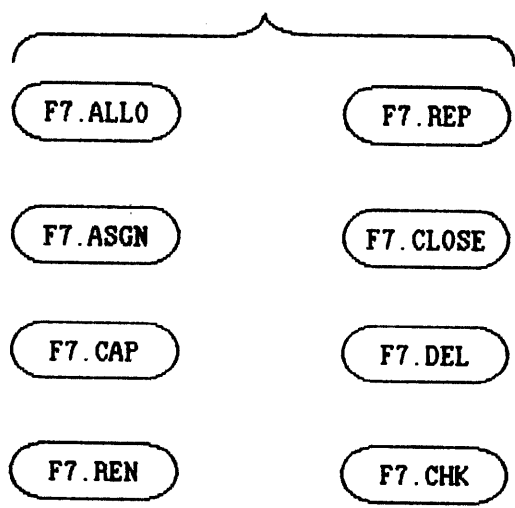
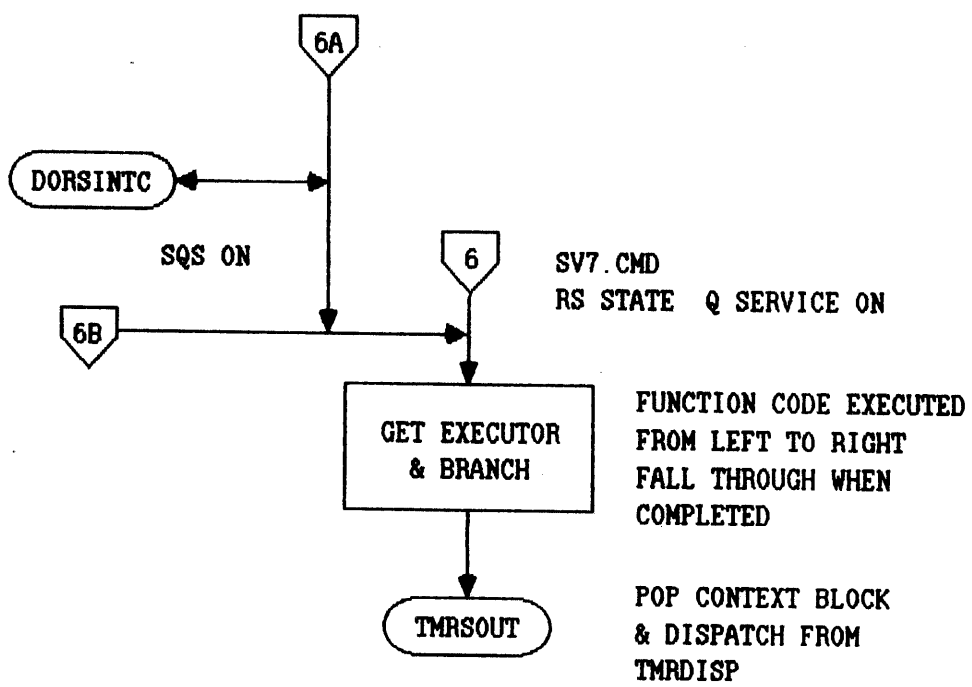
FMS7 RS STATE
SECOND LEVEL
PROCESSING



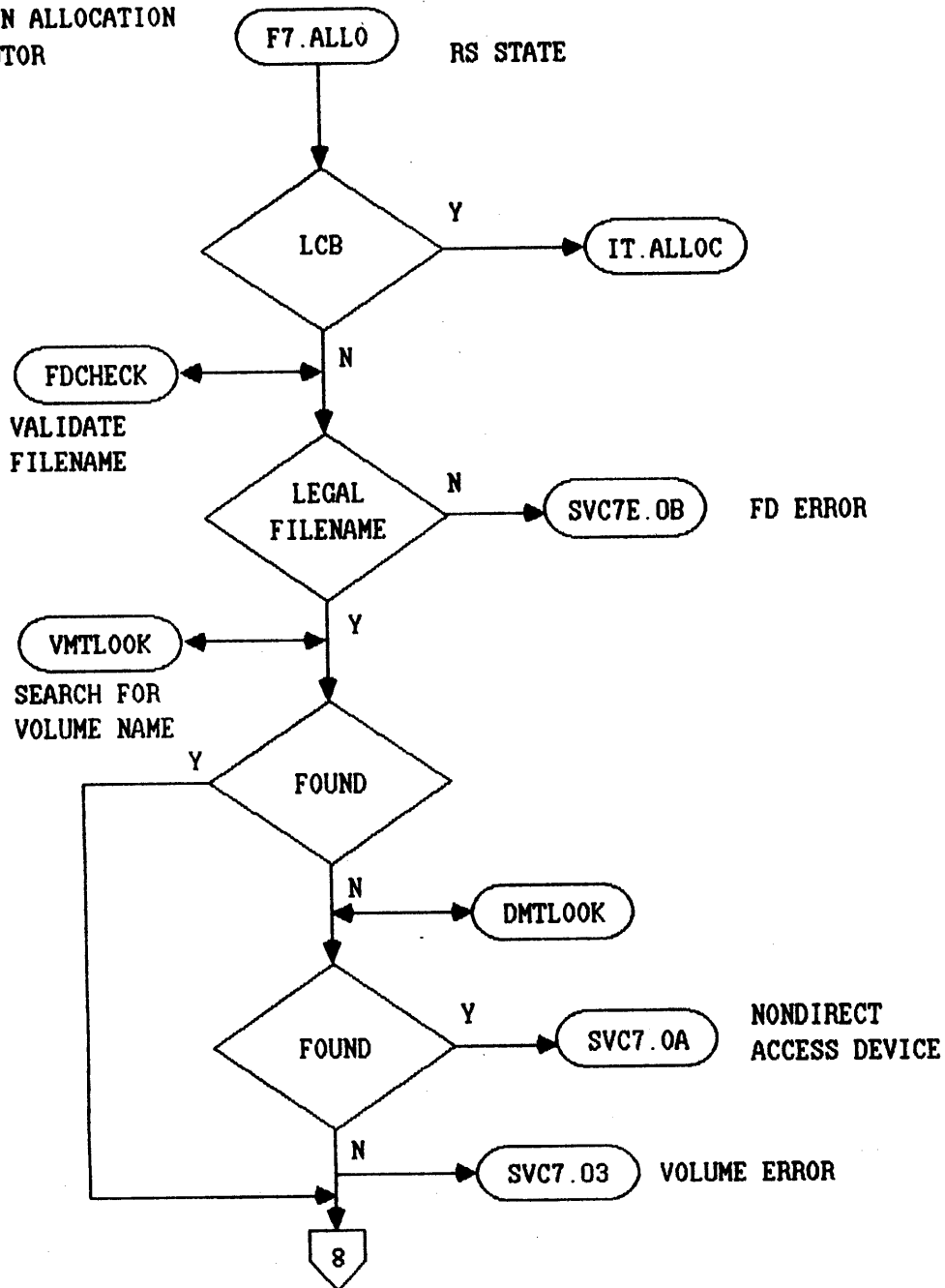


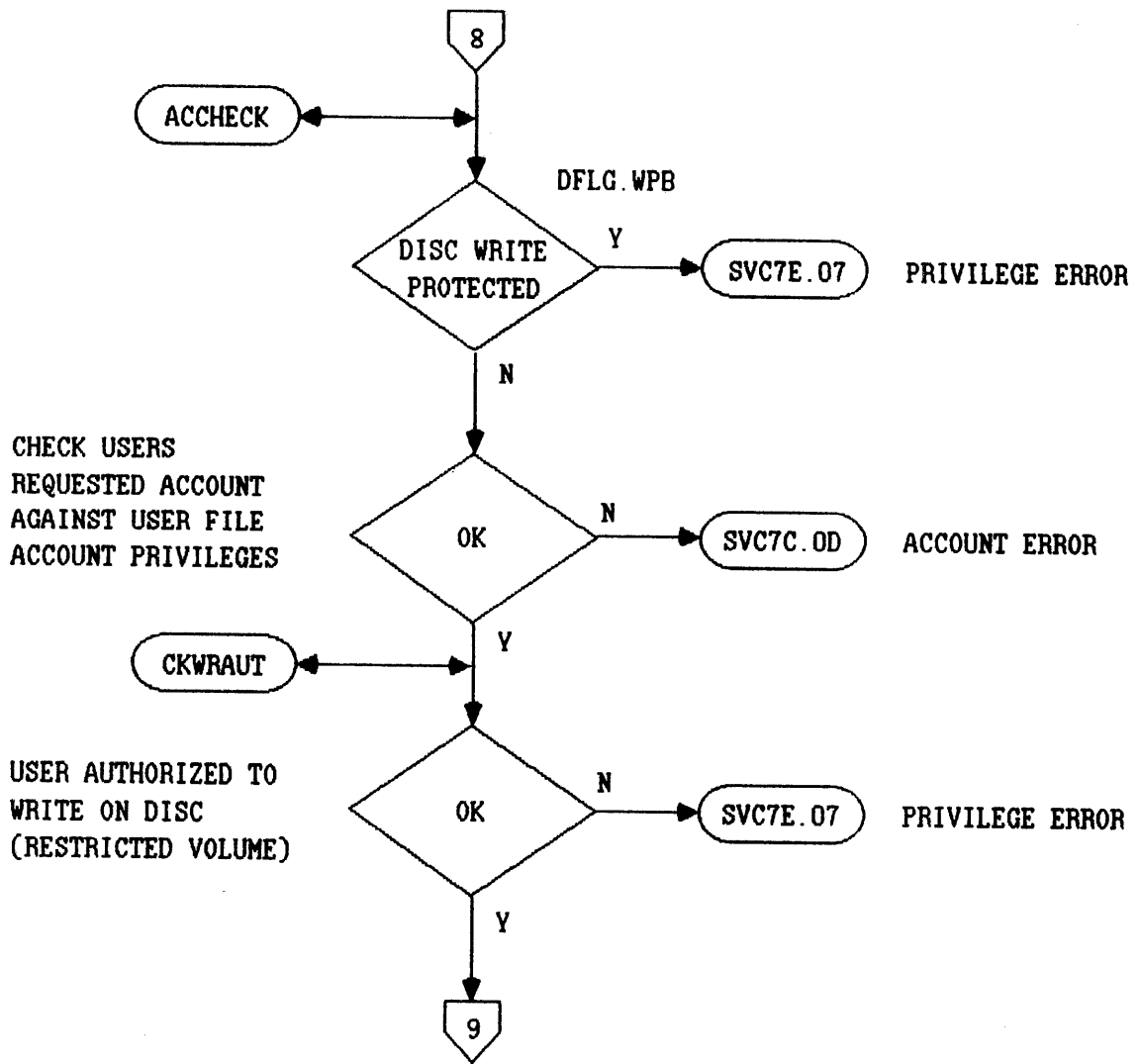


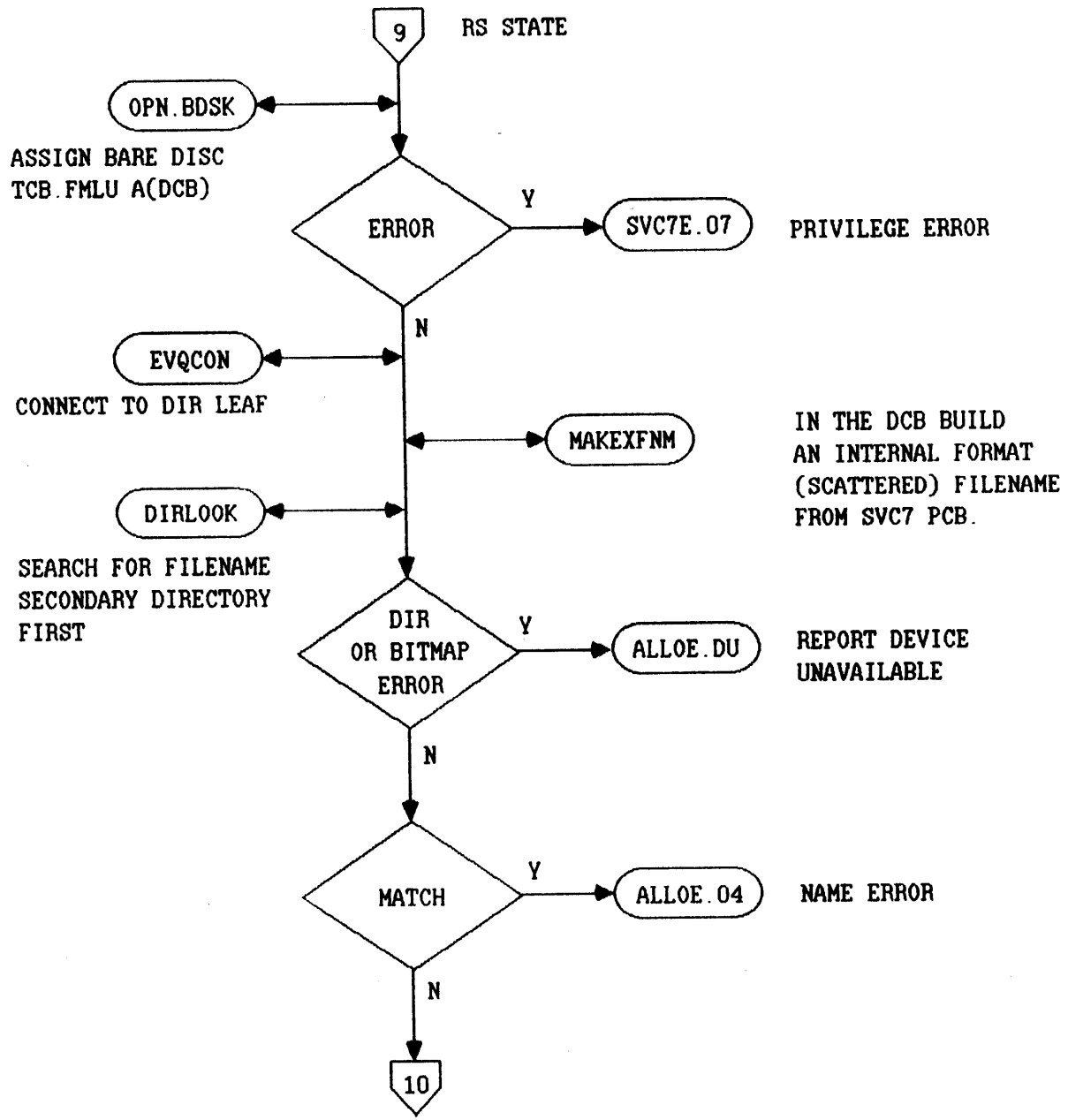


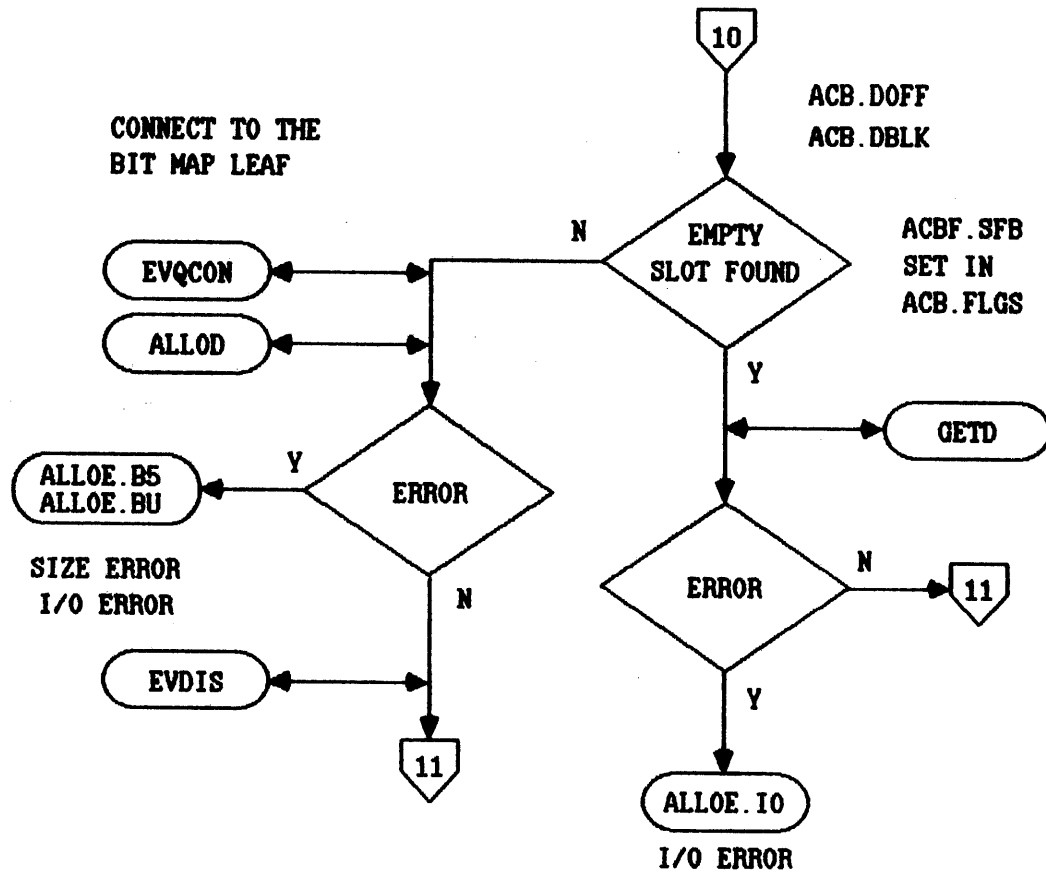


COMMON ALLOCATION
EXECUTOR









11 RS STATE

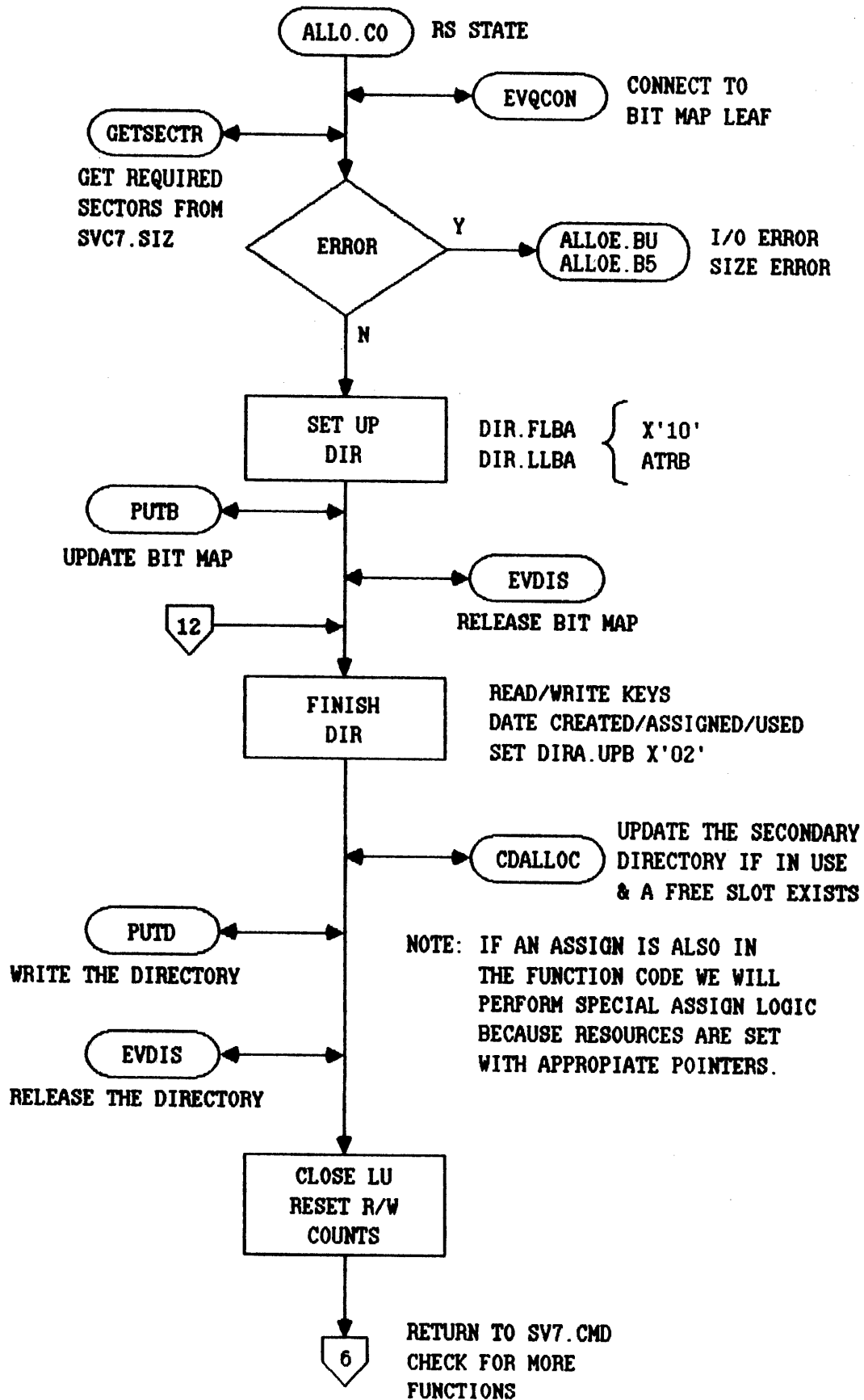
INITIALIZE
DIR ENTRY

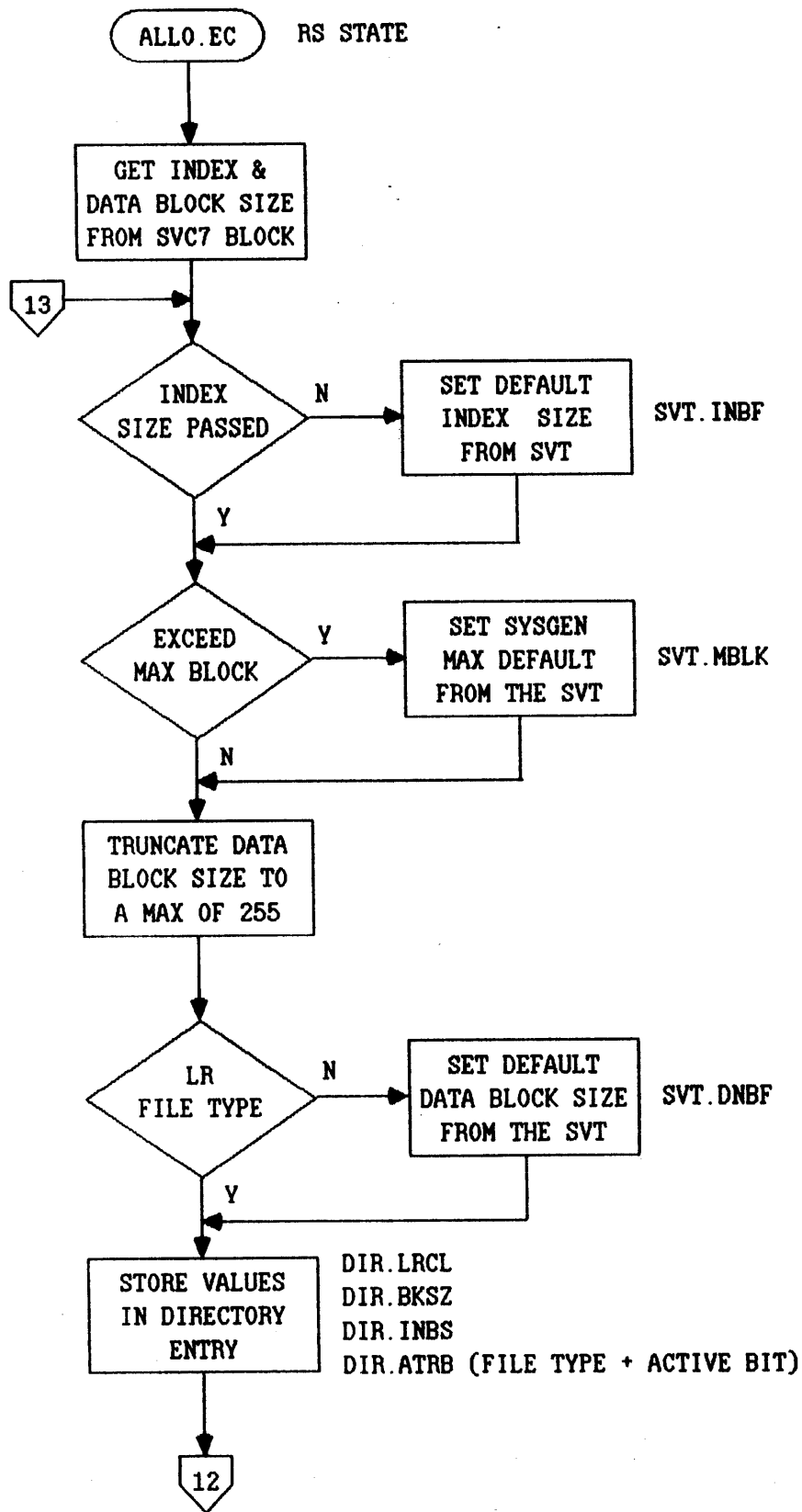
- ZERO THE ENTRY
- STORE INTERNAL FORMAT NAME FROM DCB.XFNM TO ENTRY IN DCB
- SET PREALLOCATED BIT IN ATRB X'01'

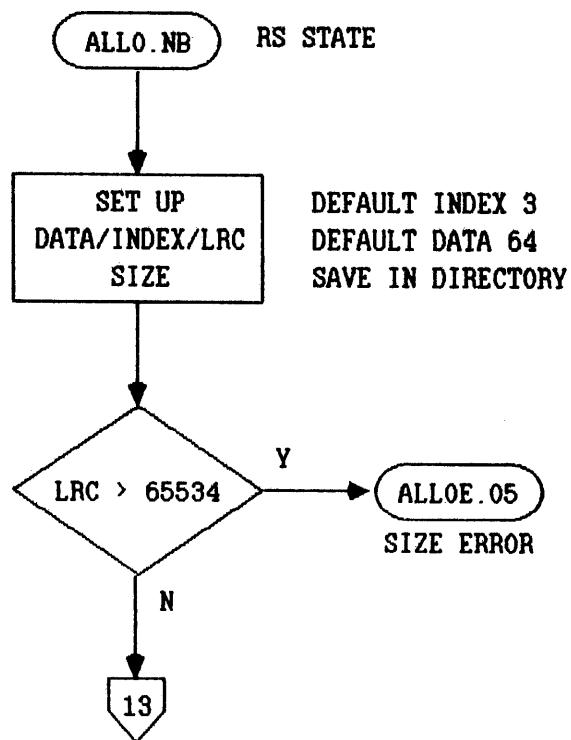
GET
EXECUTOR
BRANCH

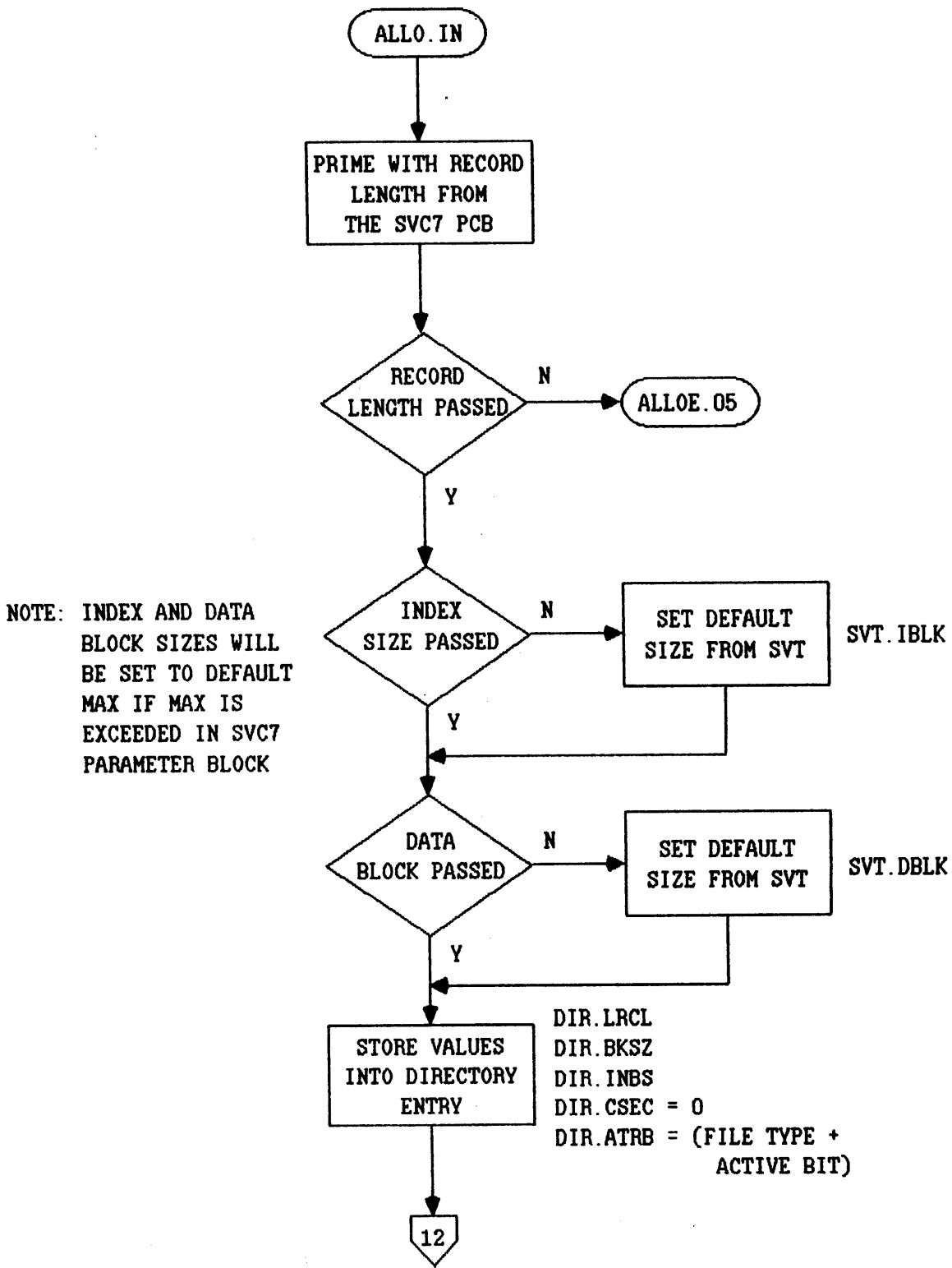


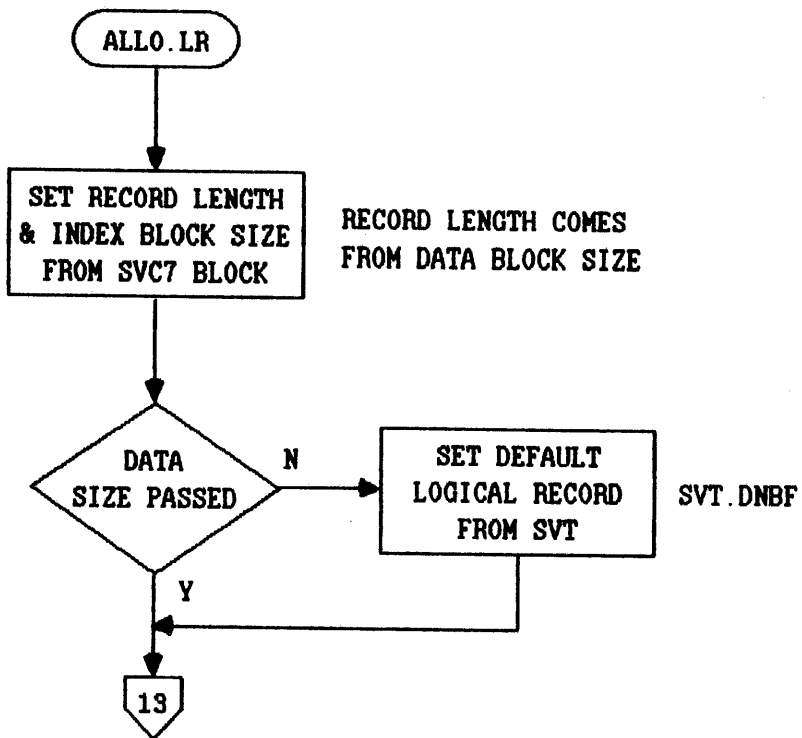
ALLO.CO	CONTIGUOUS
ALLO.EC	EXTENDABLE CONTIGUOUS
ALLO.IN	INDEXED
ALLO.NB	NONBUFFERED INDEX
ALLO.LR	LONG RECORD FILE

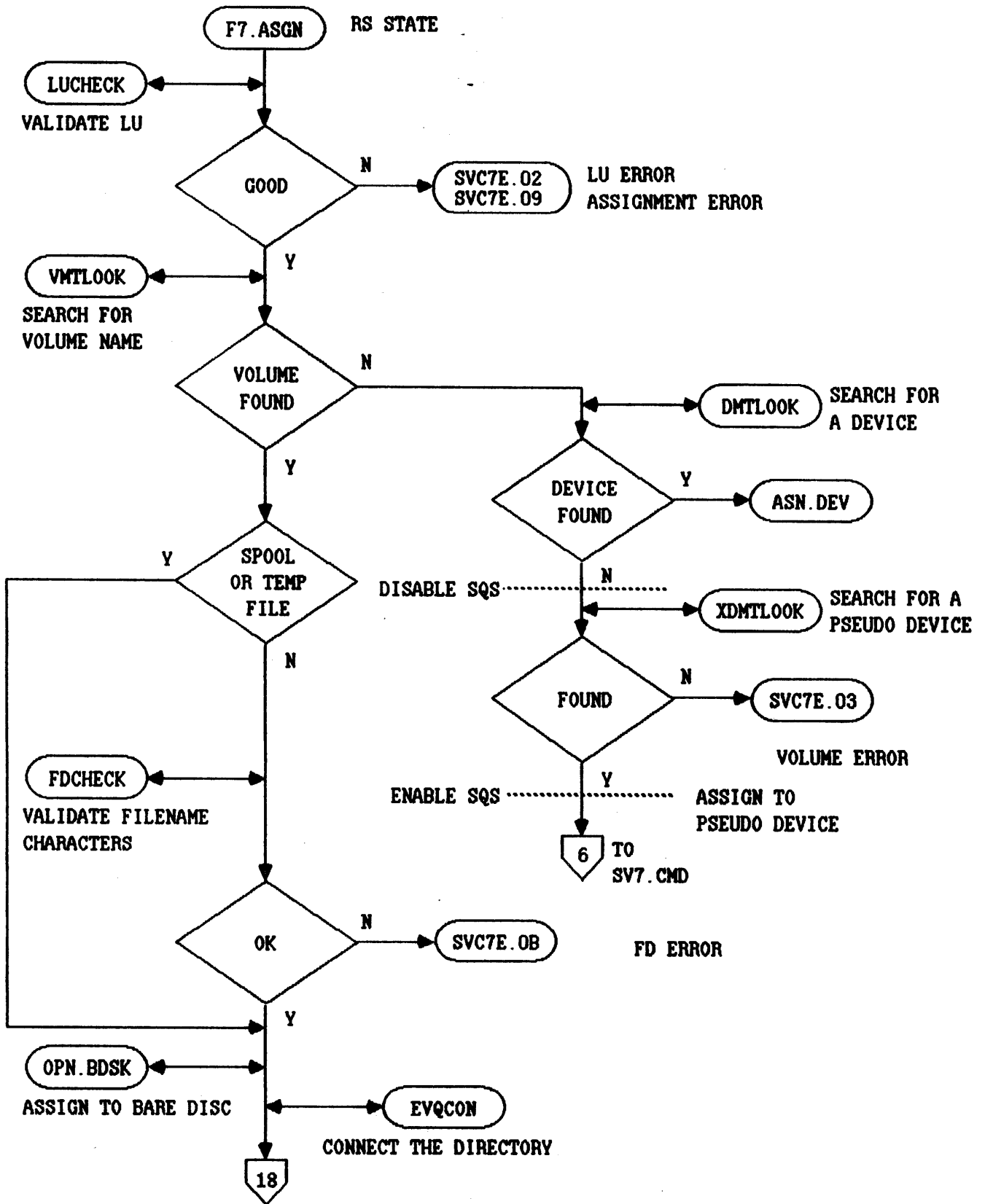


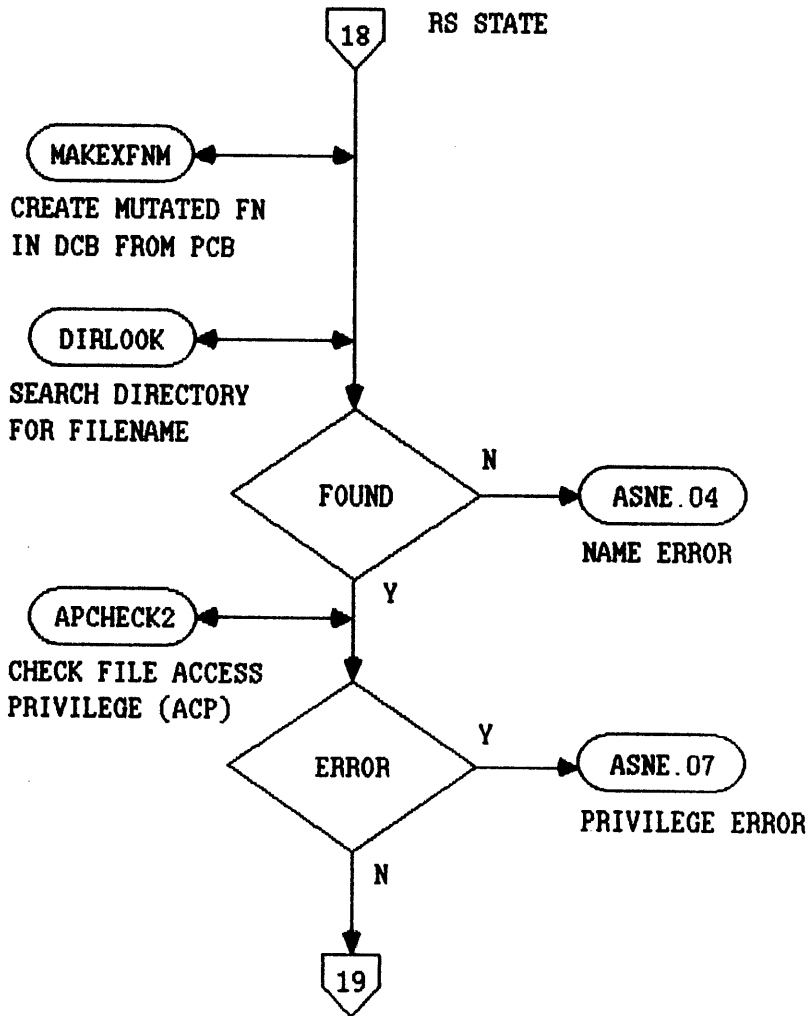


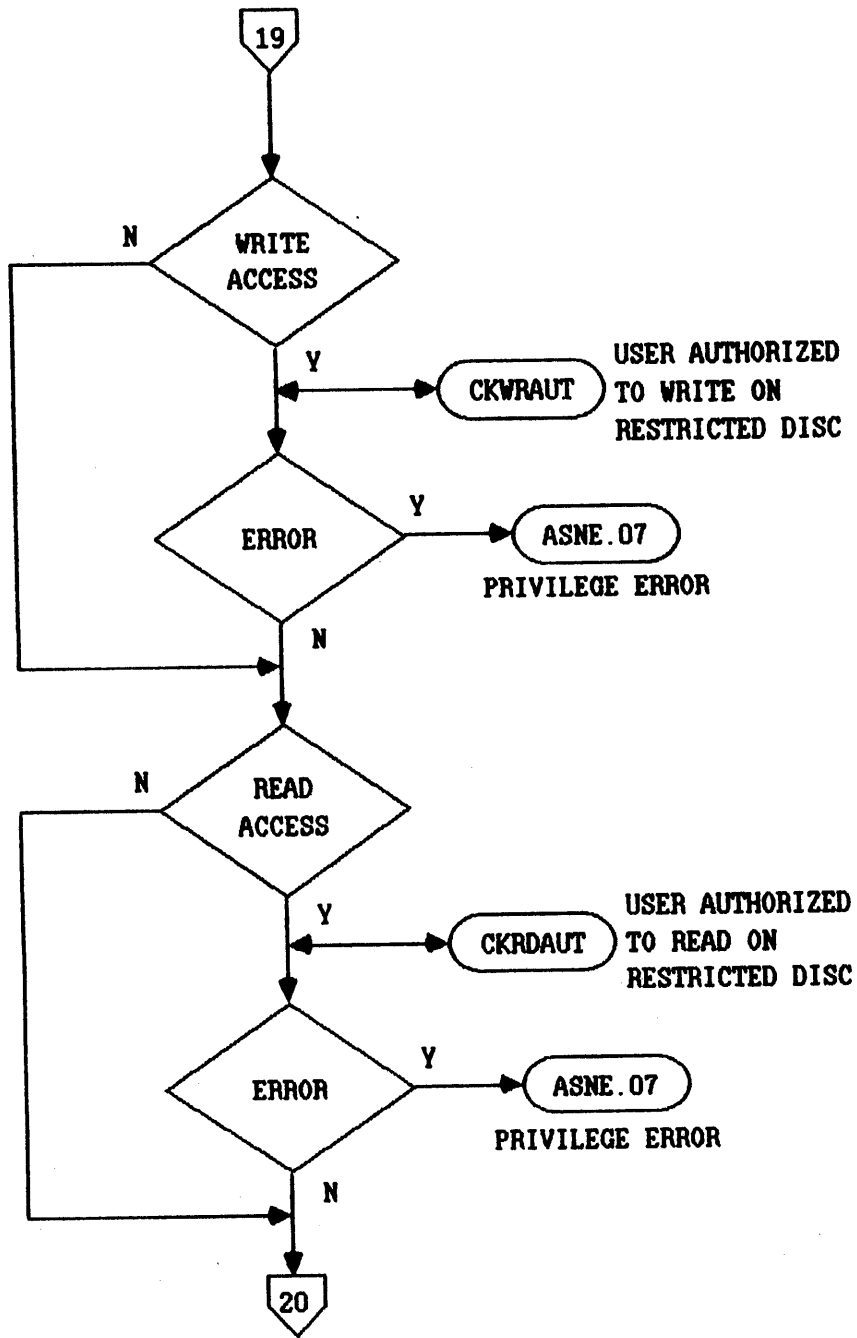


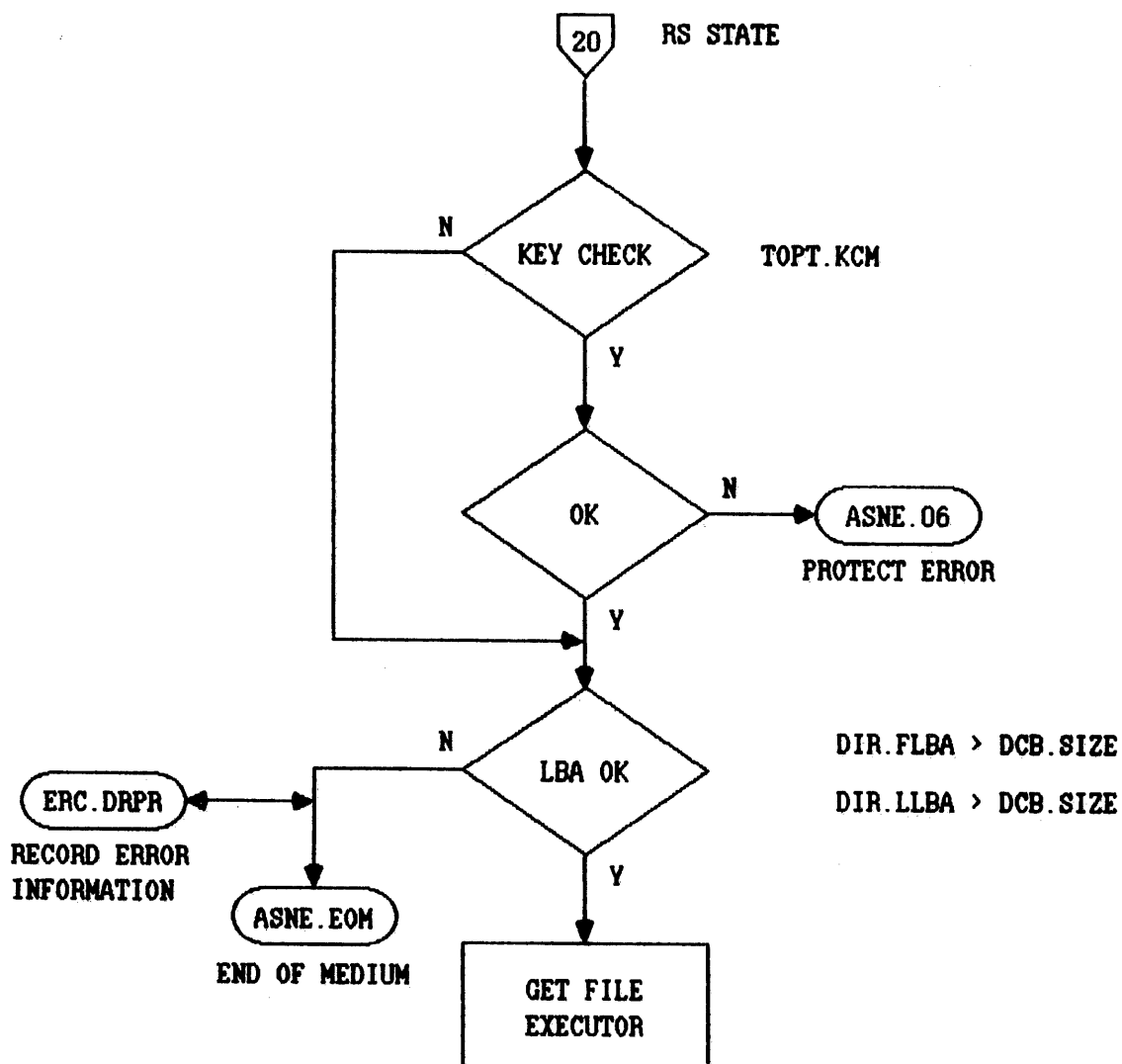




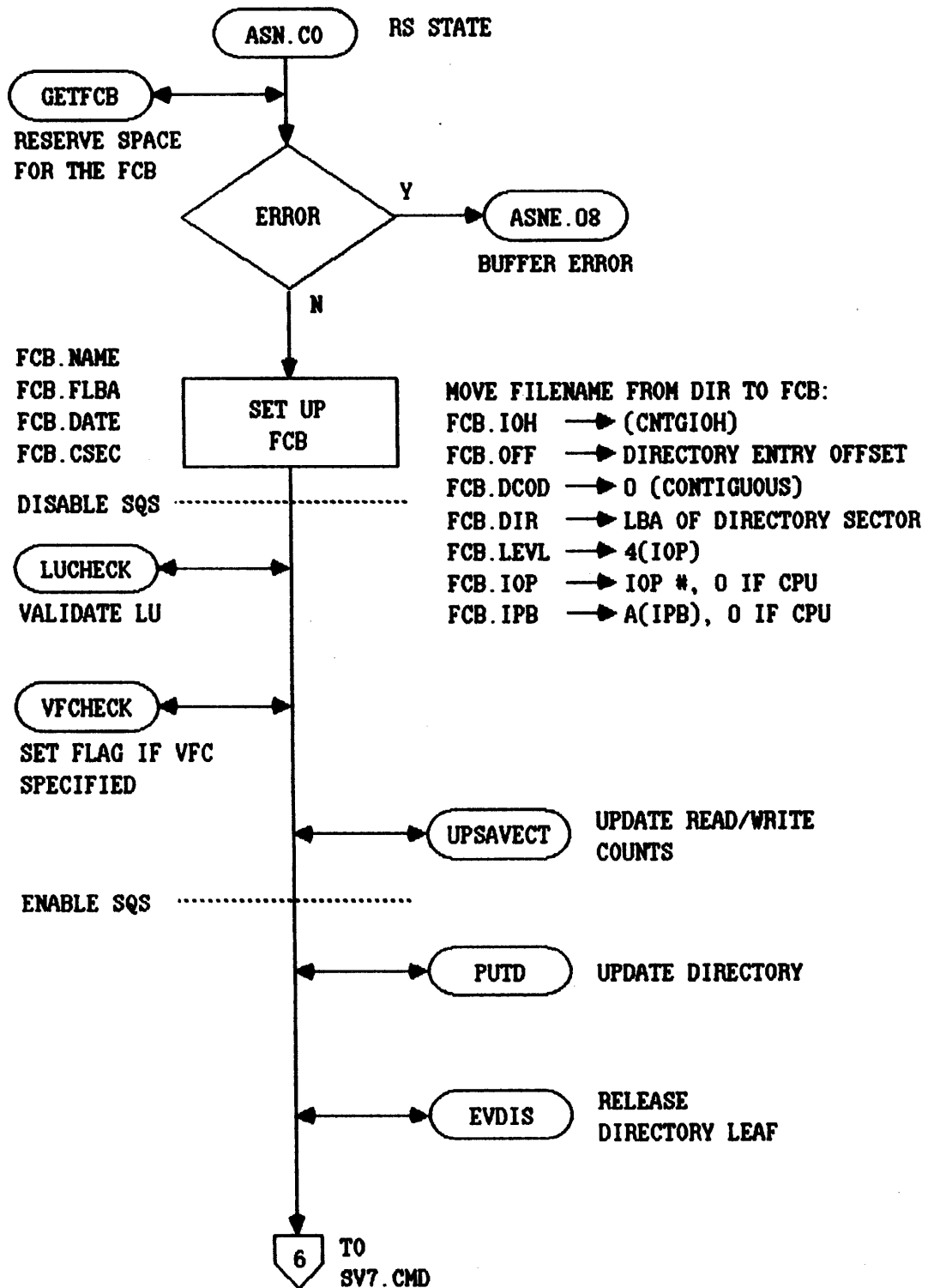


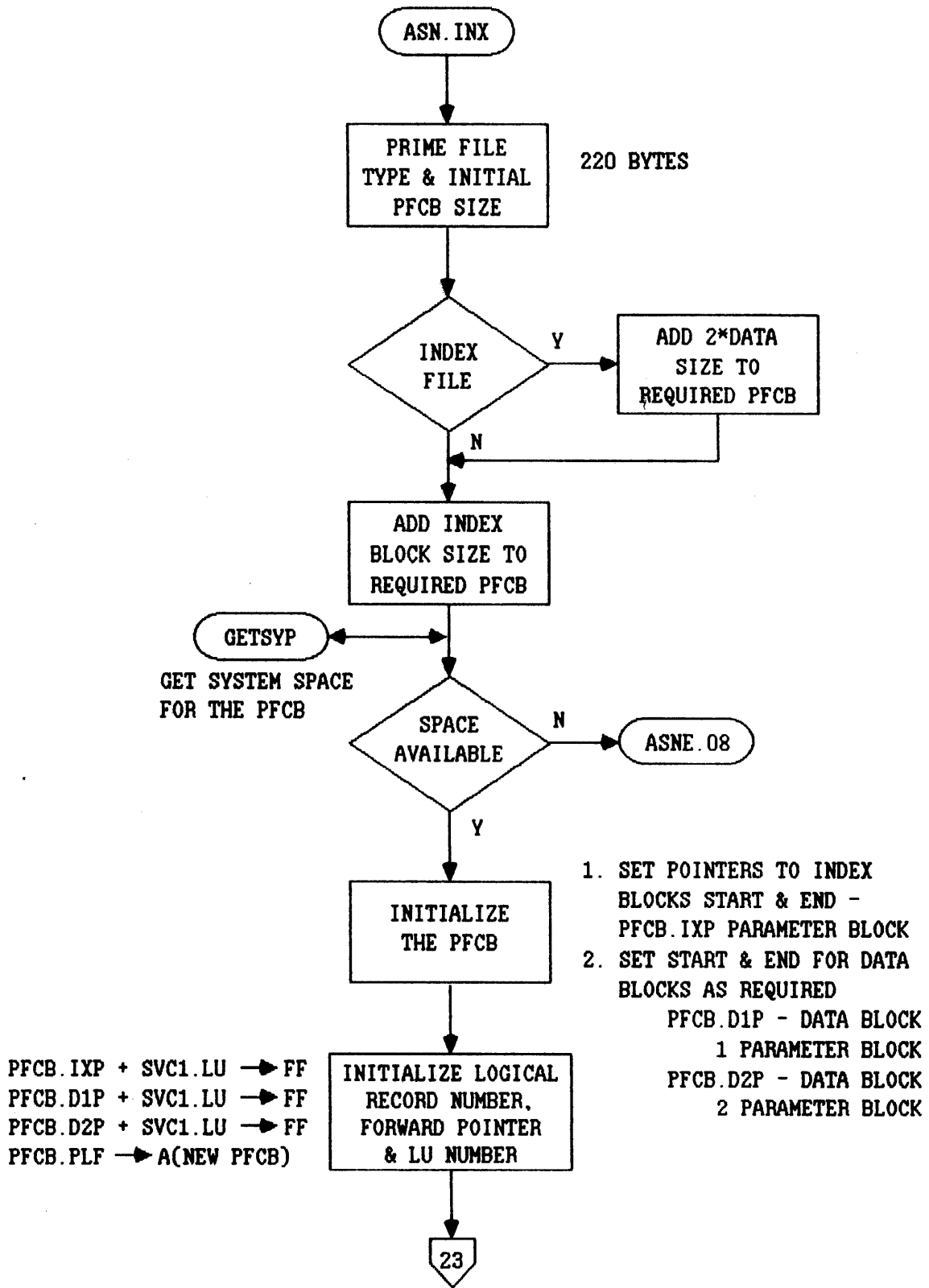


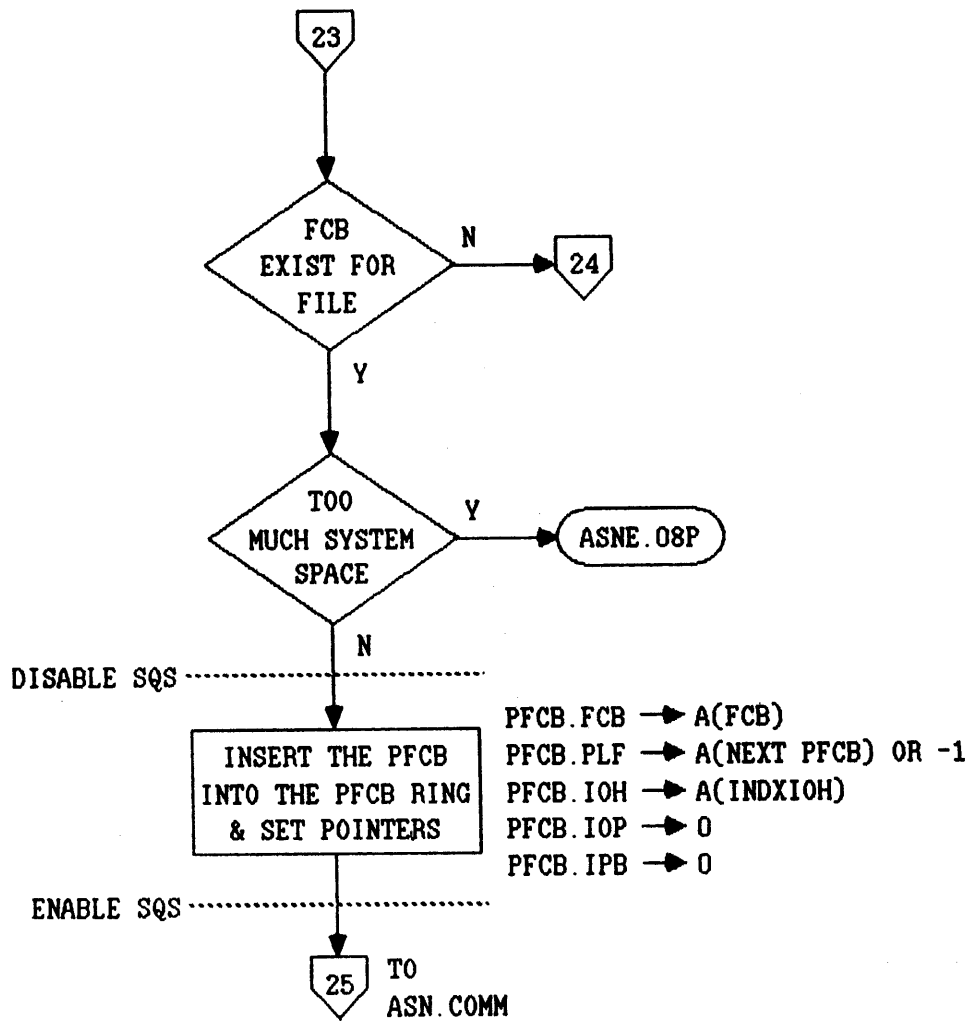


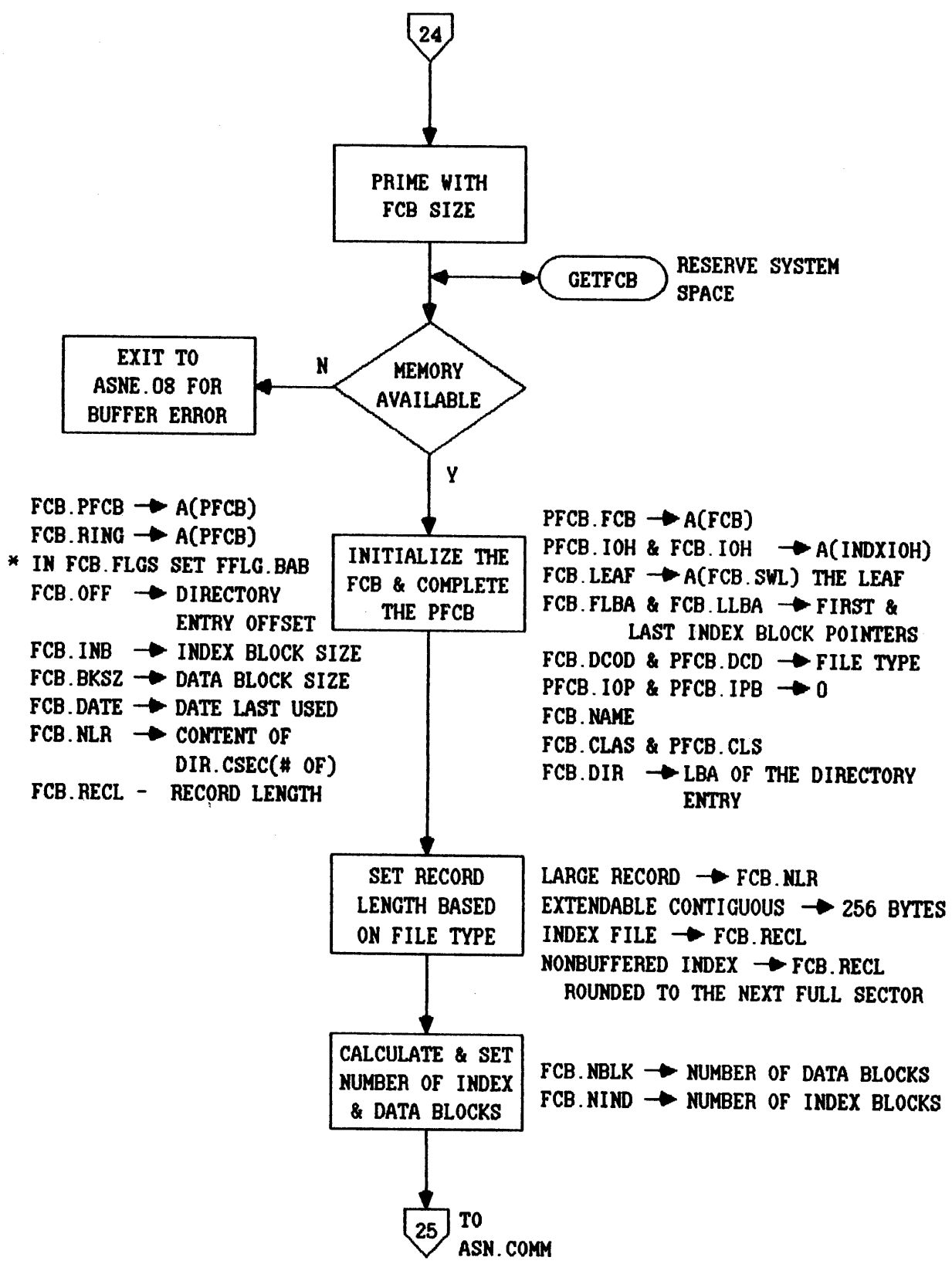


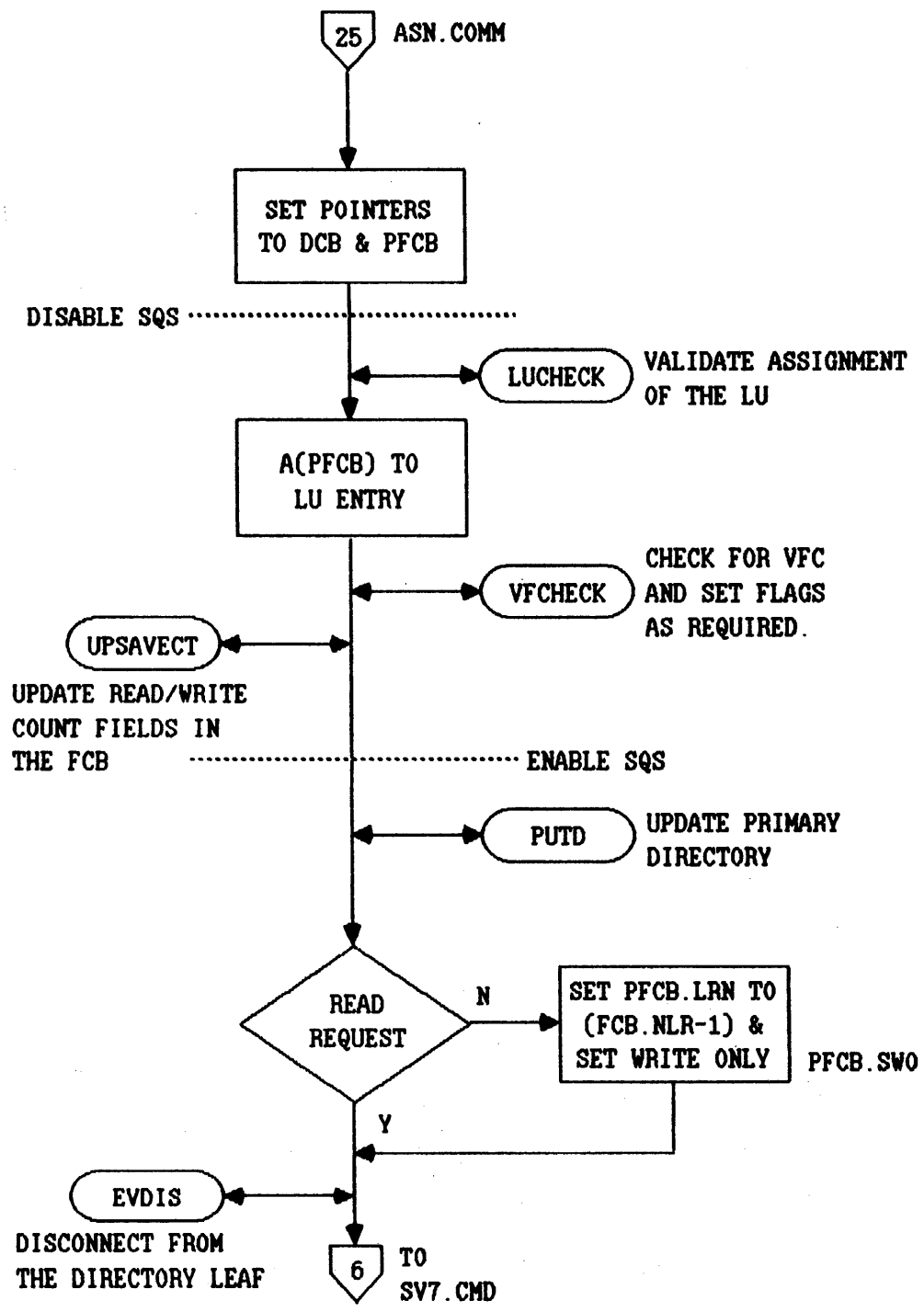
CONTIGUOUS	ASN.CO
NONBUFFERED INDEX	ASN.INX
INDEX	ASN.INX
EXTENDABLE CONTIGUOUS	ASN.INX
LONG RECORD FILE	ASN.INX

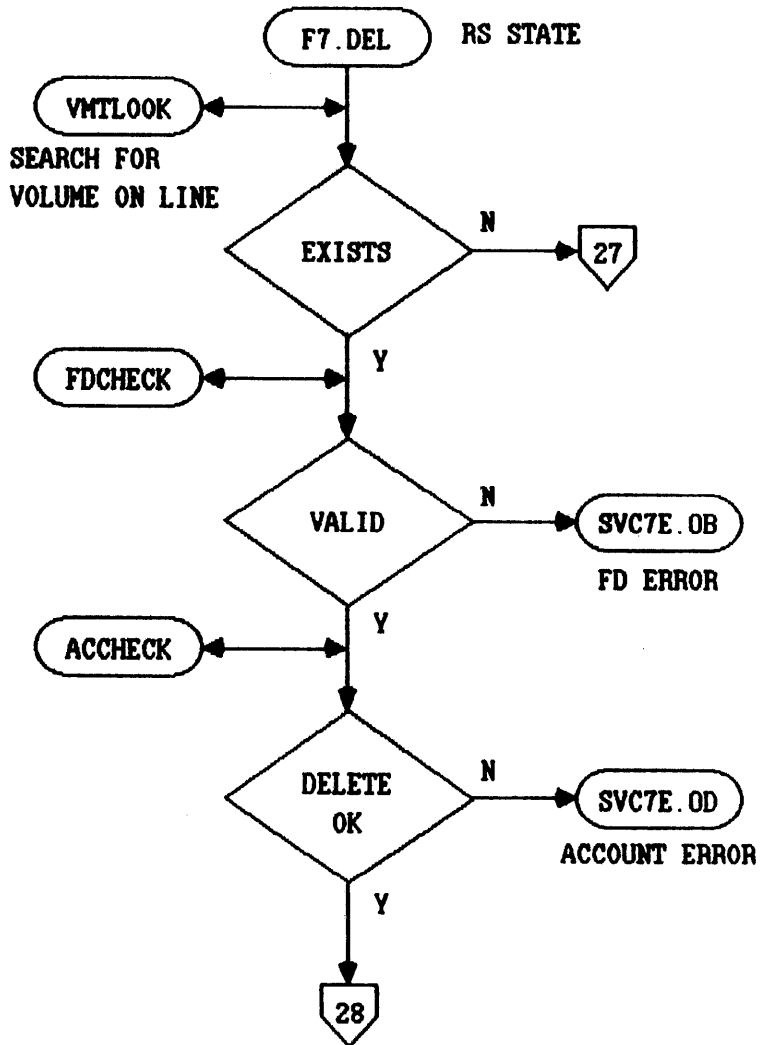


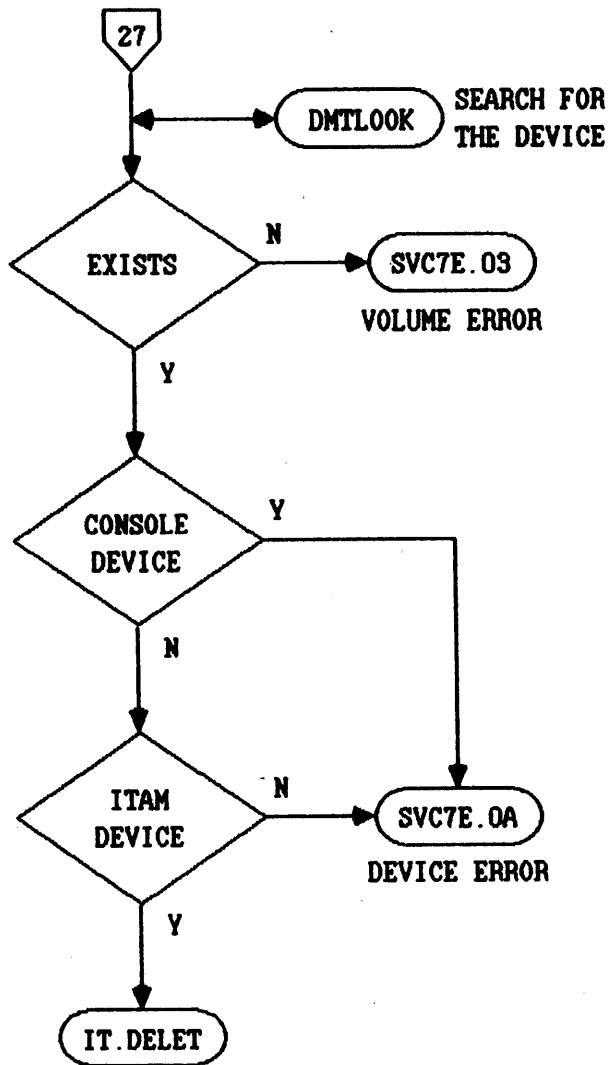


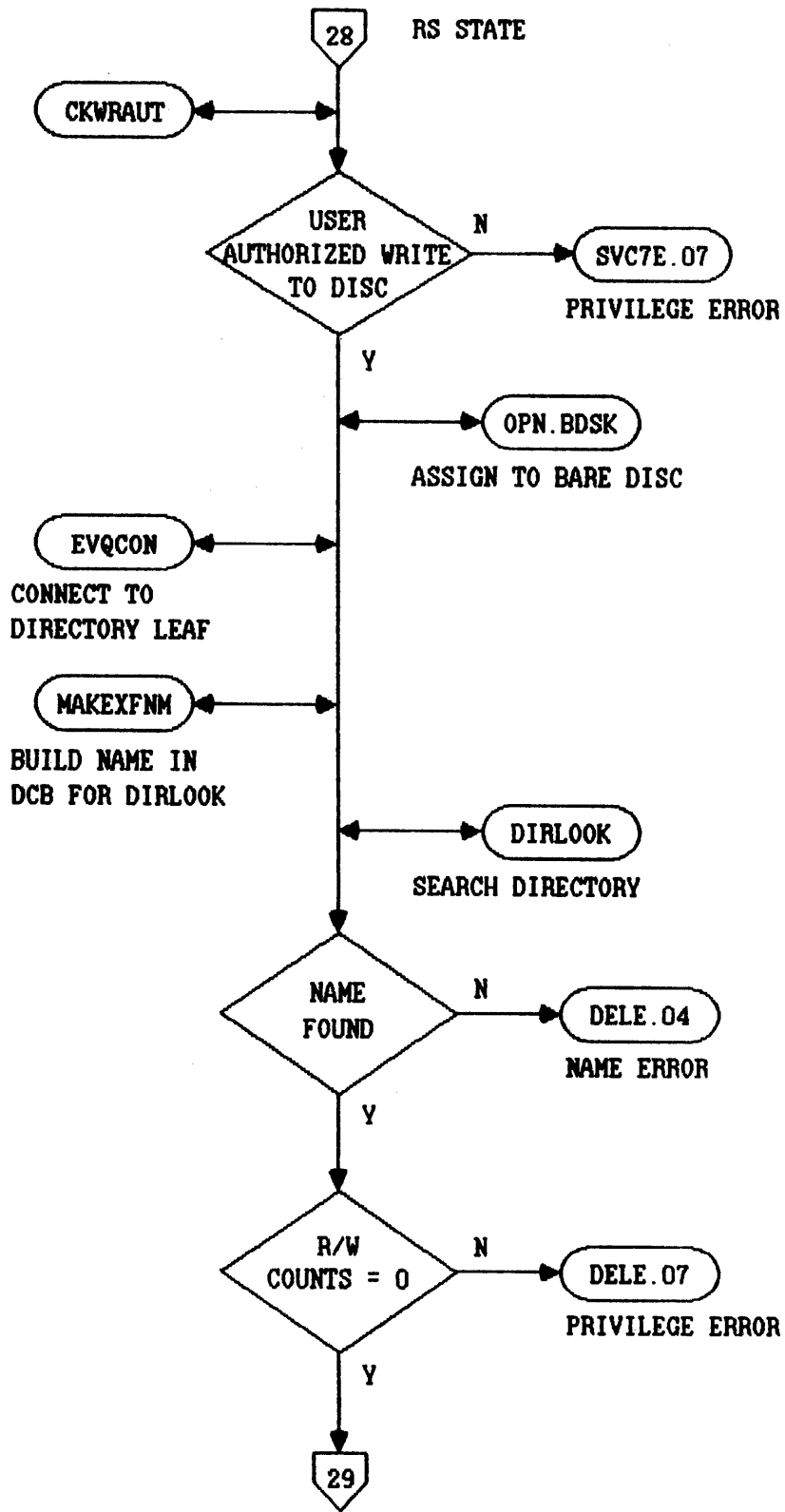


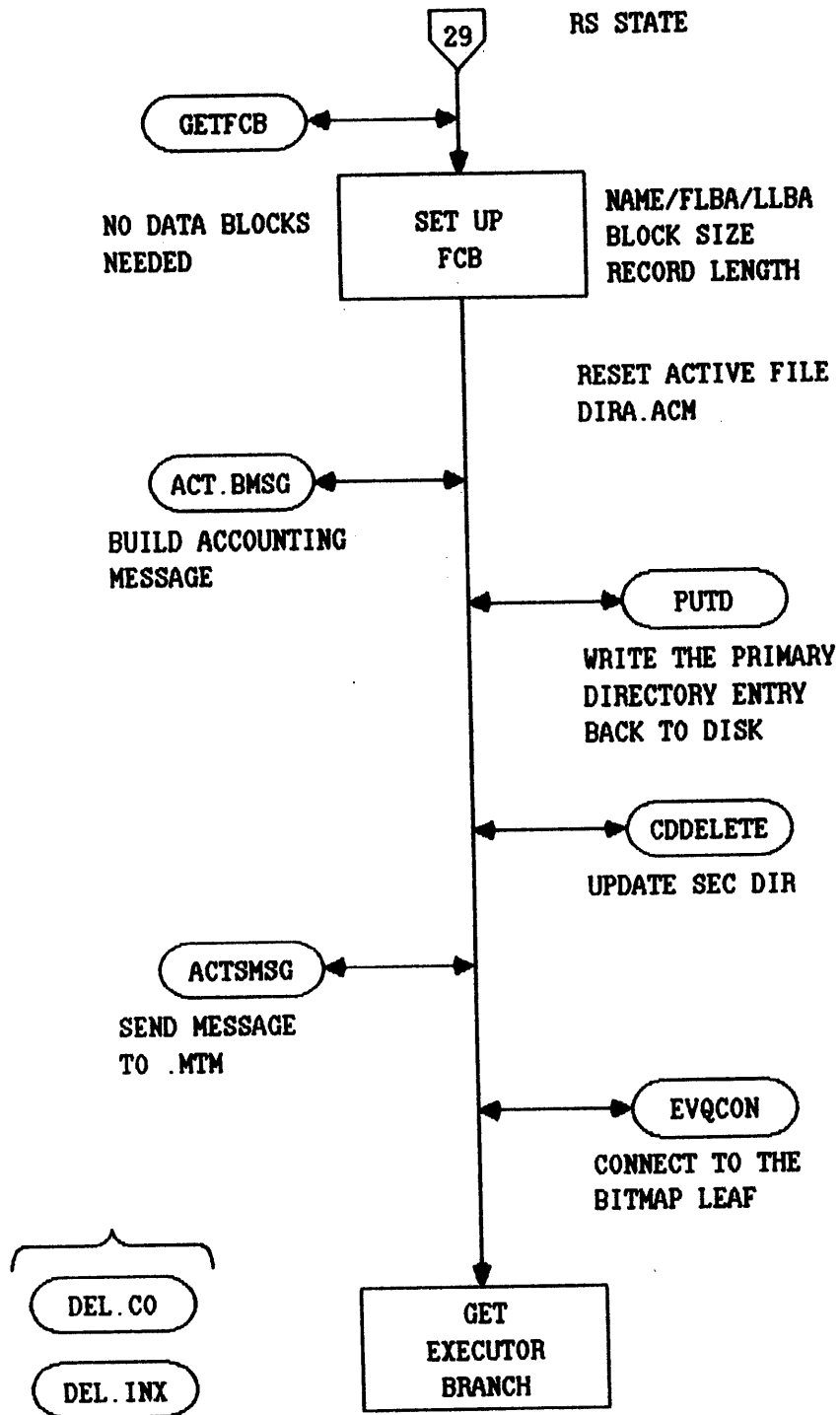


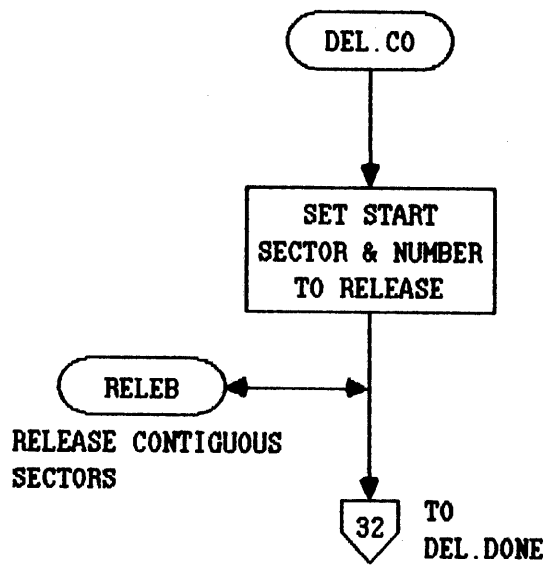


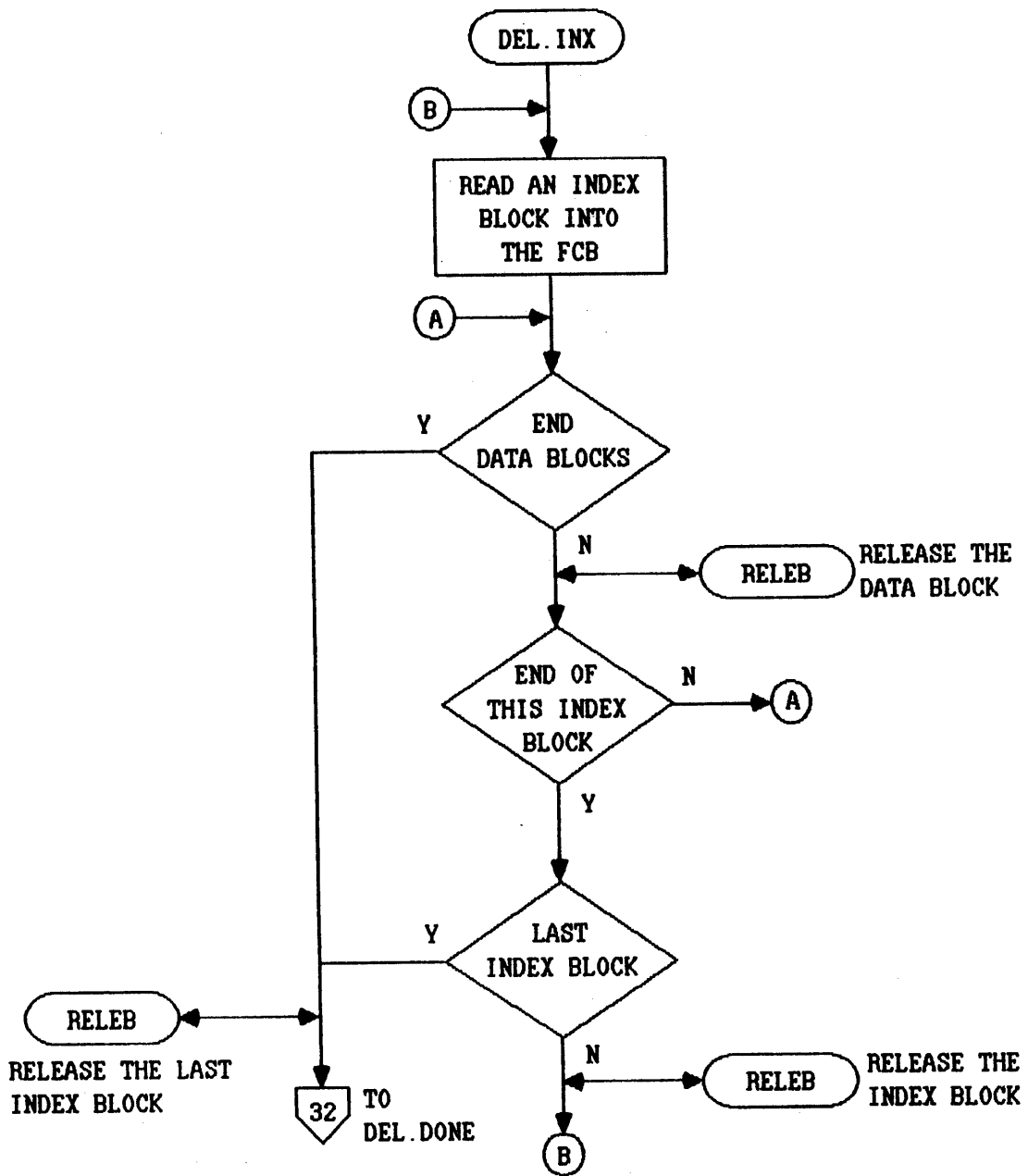


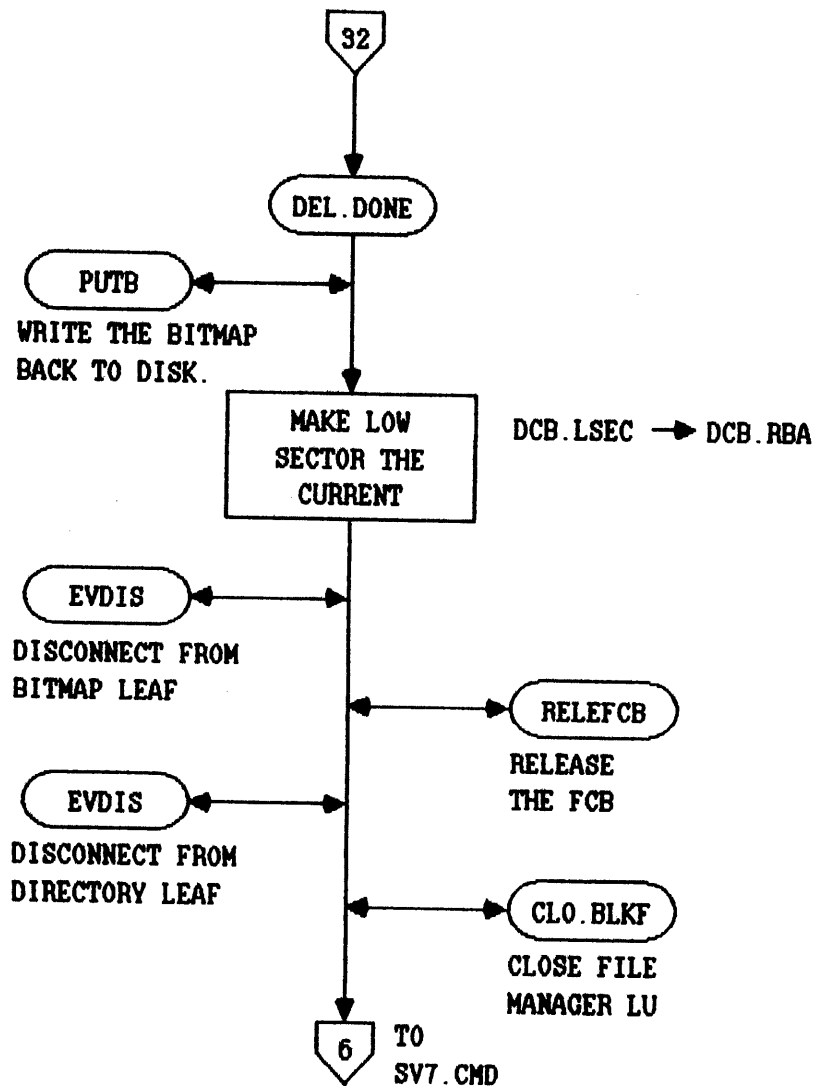




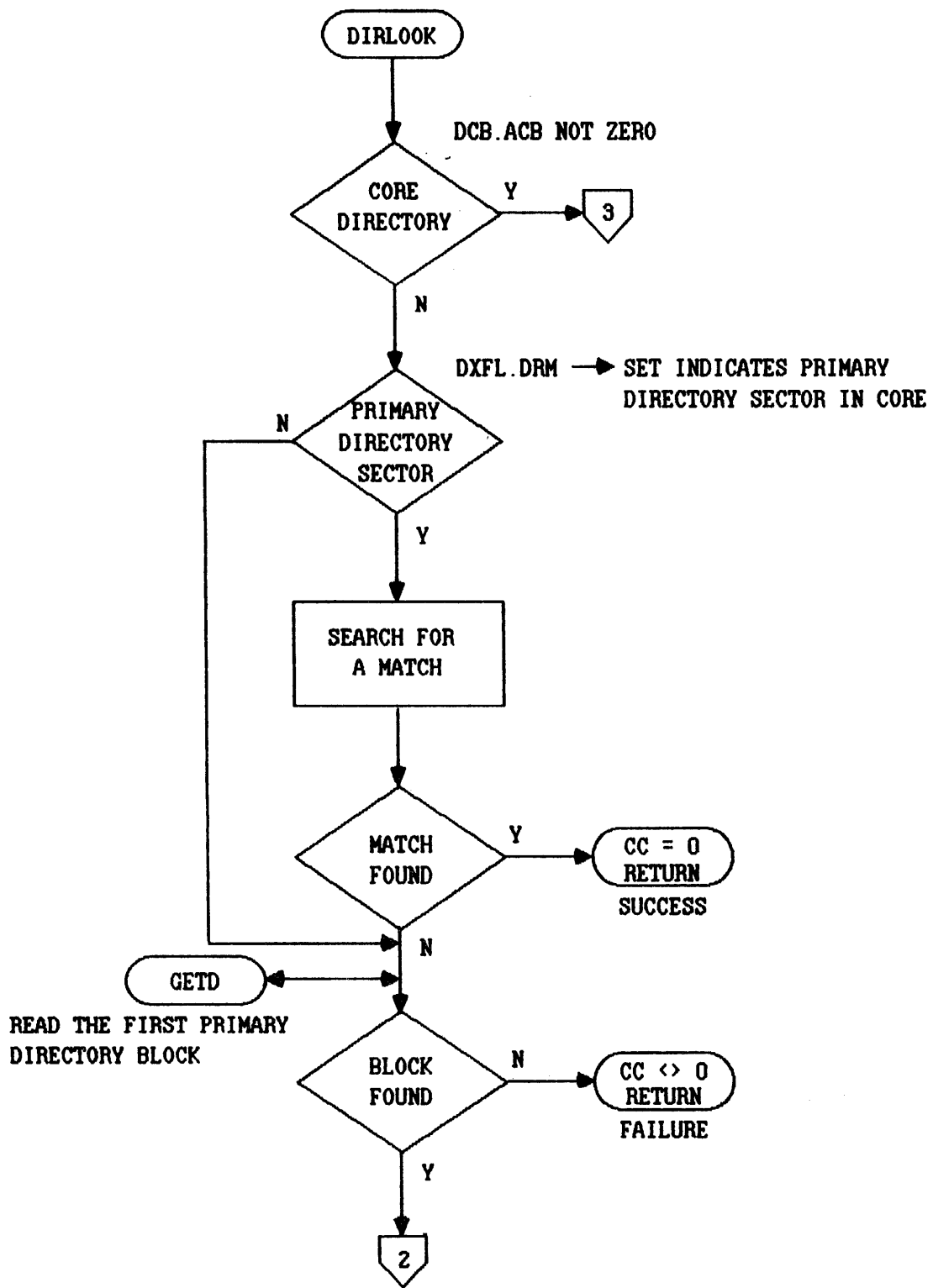


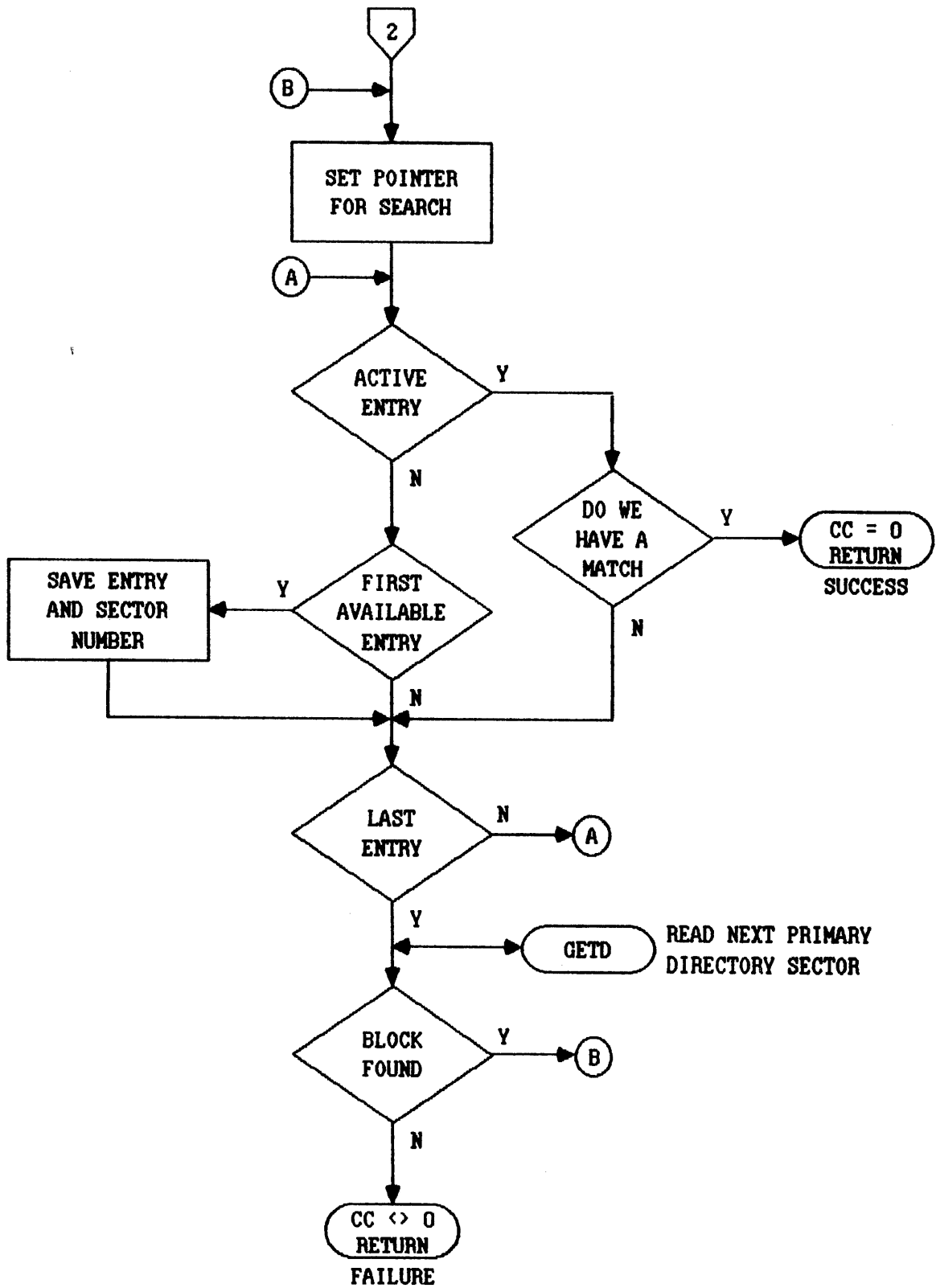






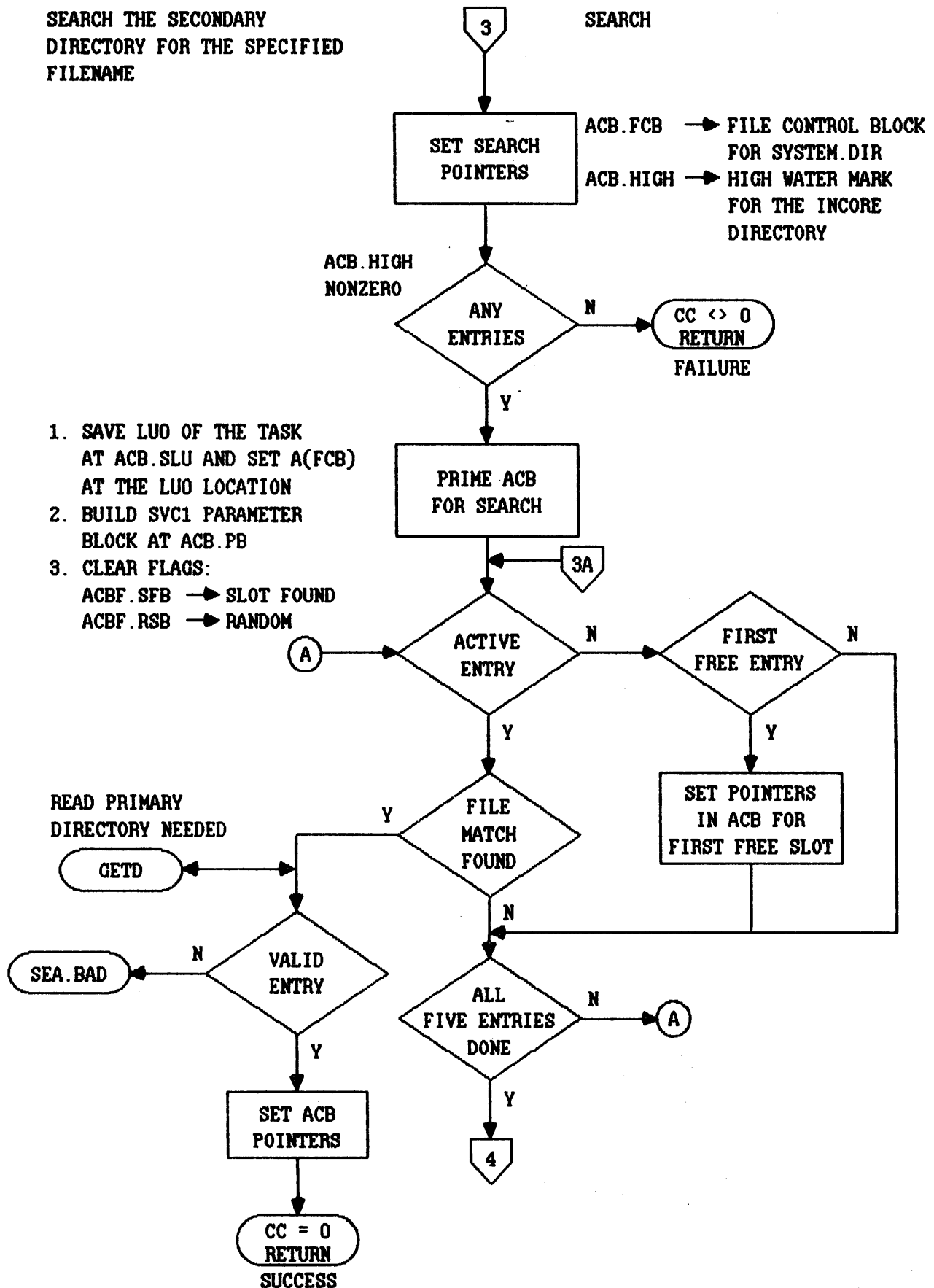
***FILE MANAGEMENT
UTILITIES***

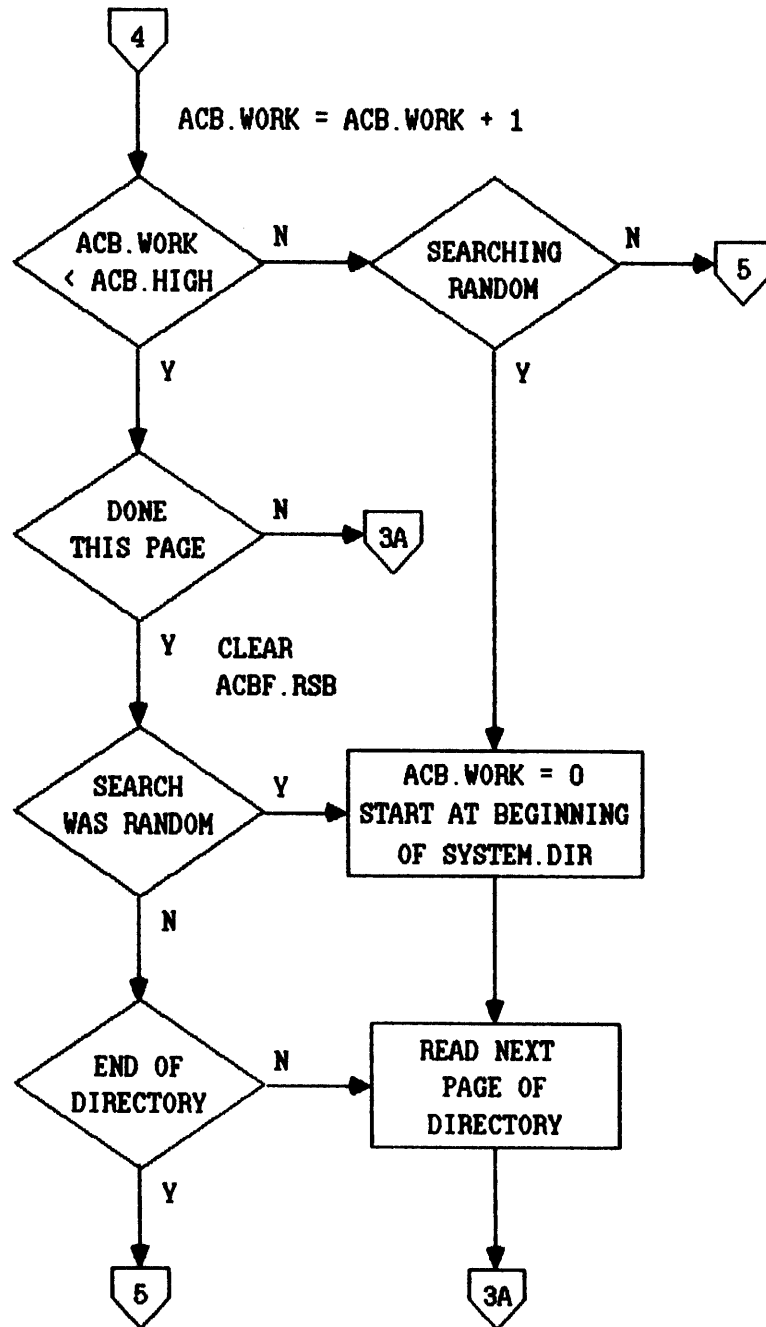


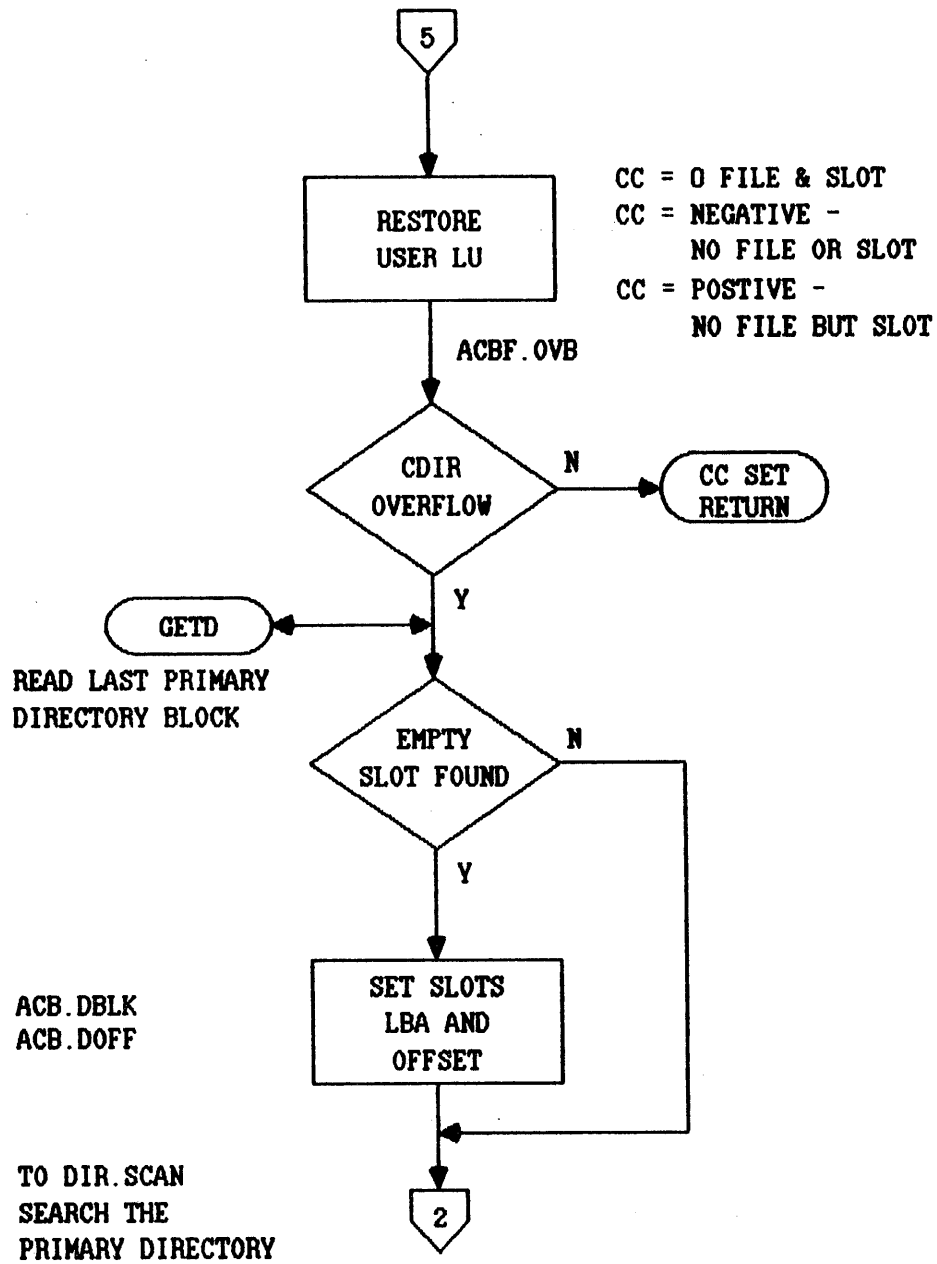


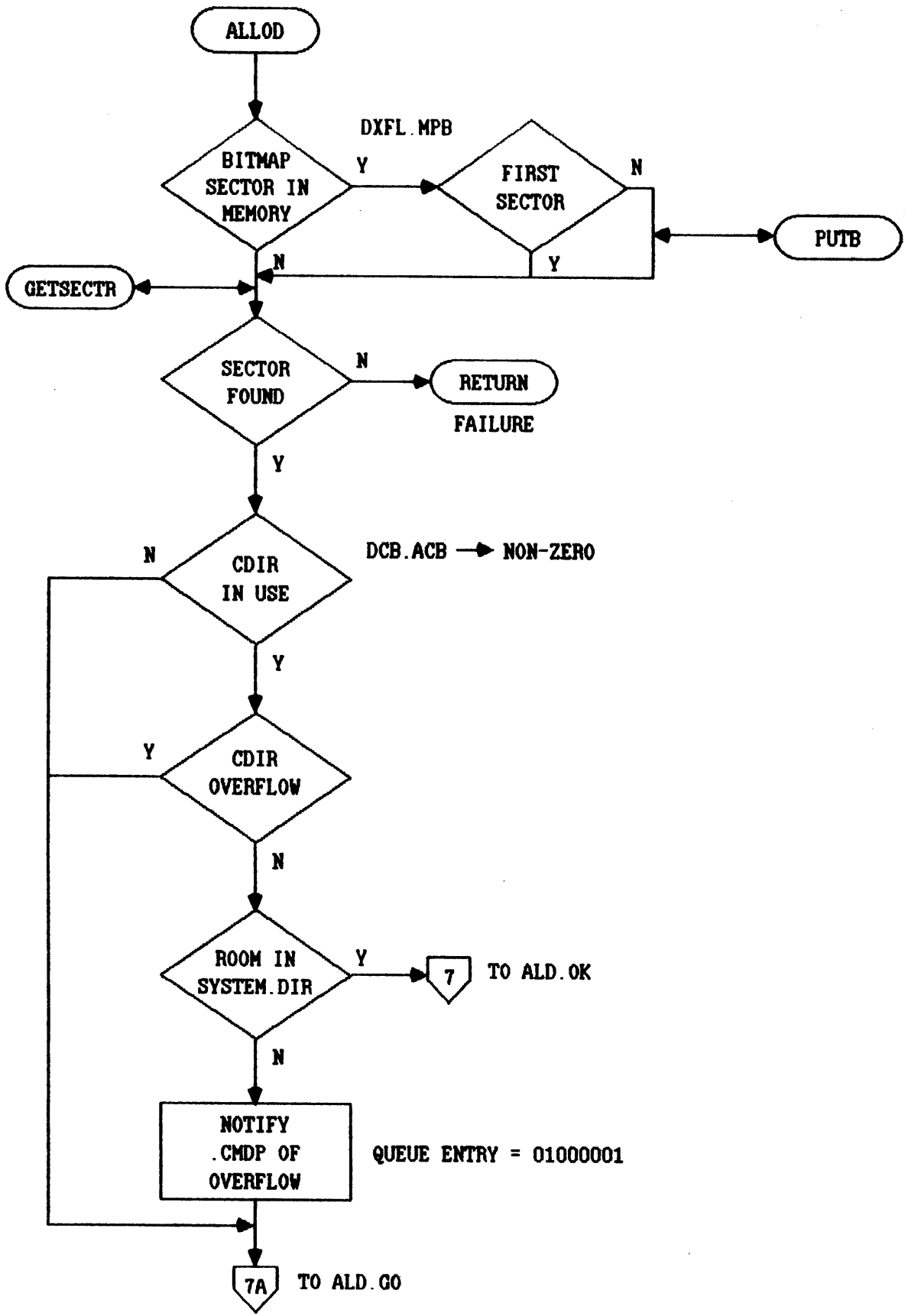
SEARCH THE SECONDARY
 DIRECTORY FOR THE SPECIFIED
 FILENAME

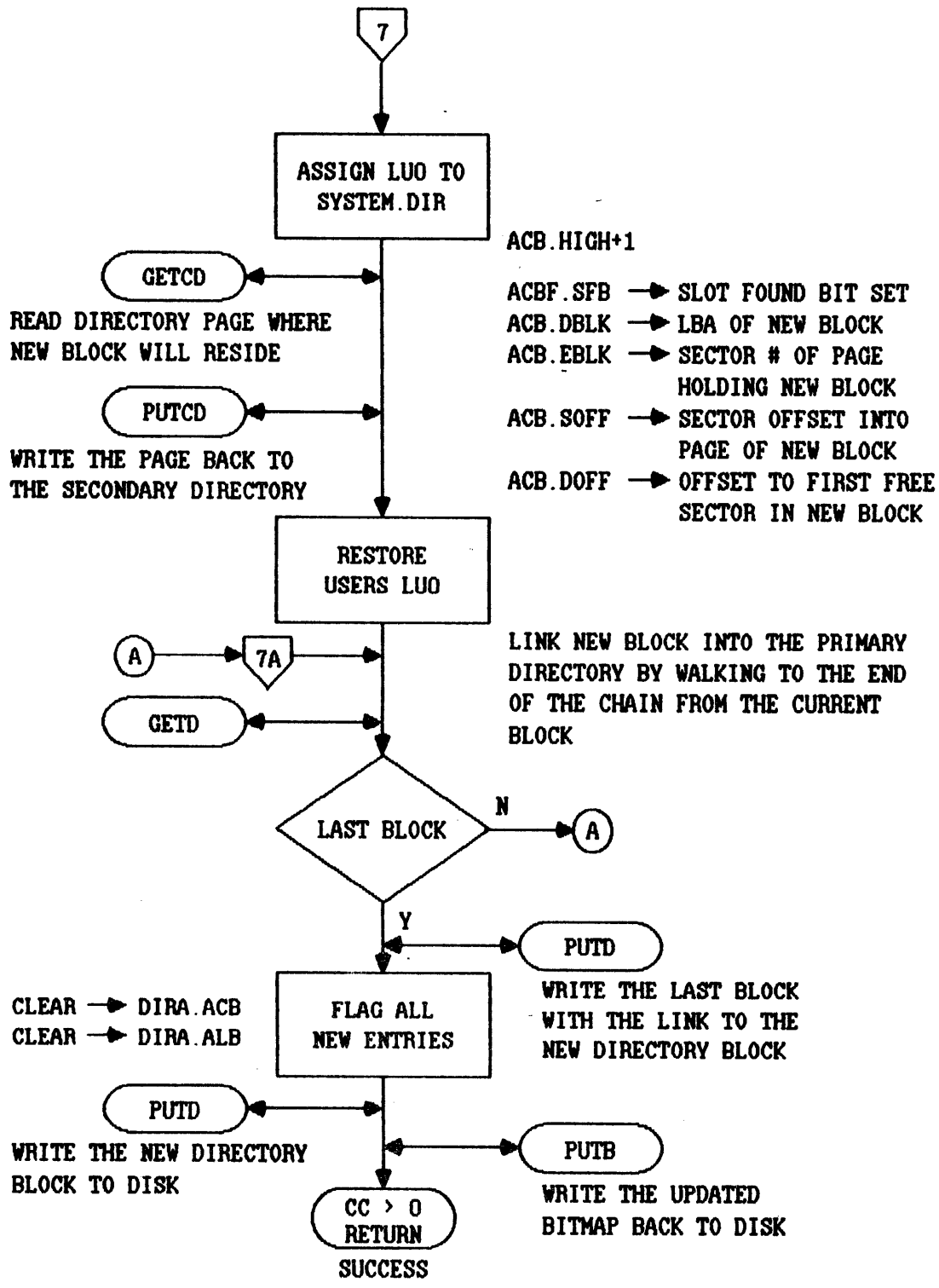
SEARCH

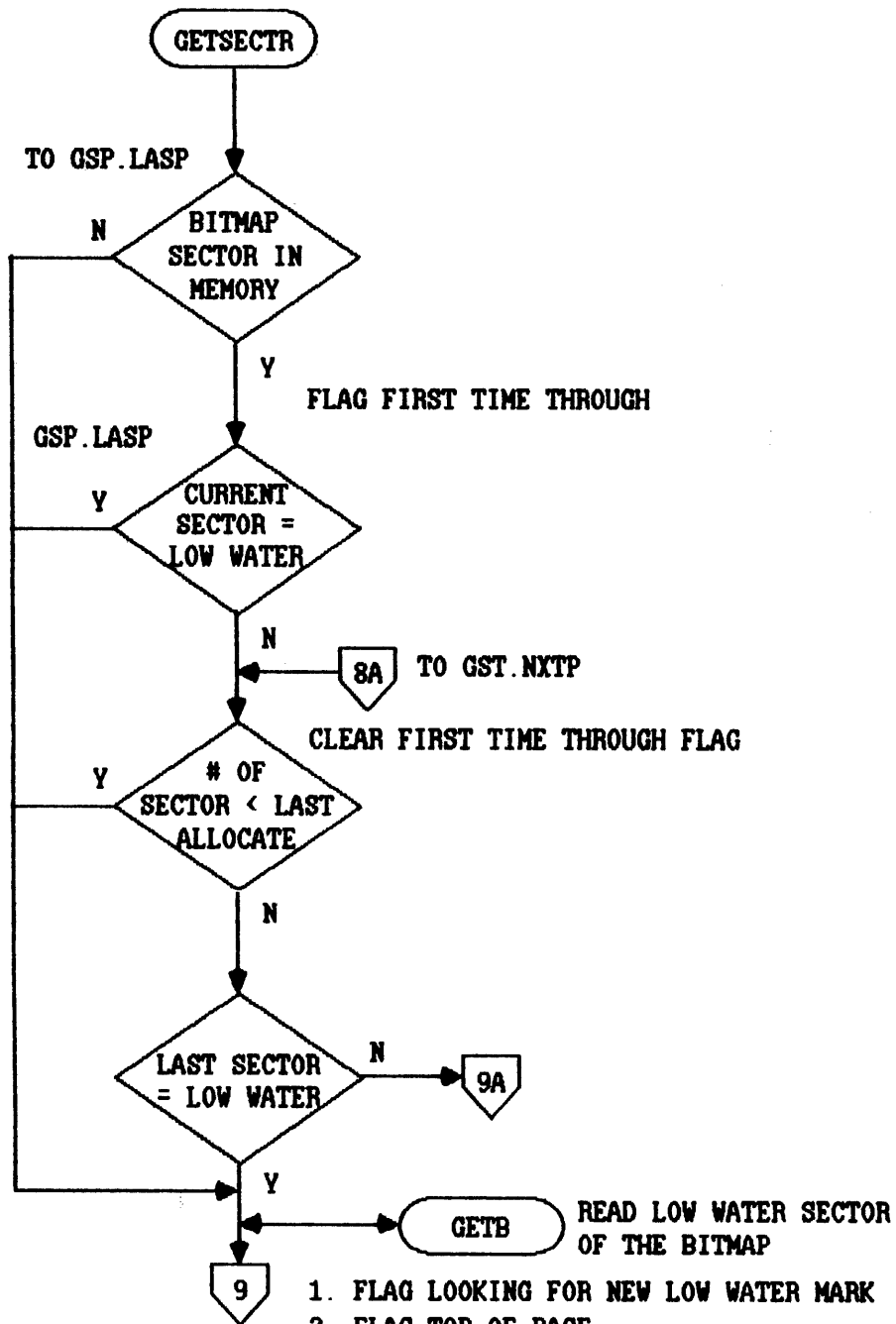


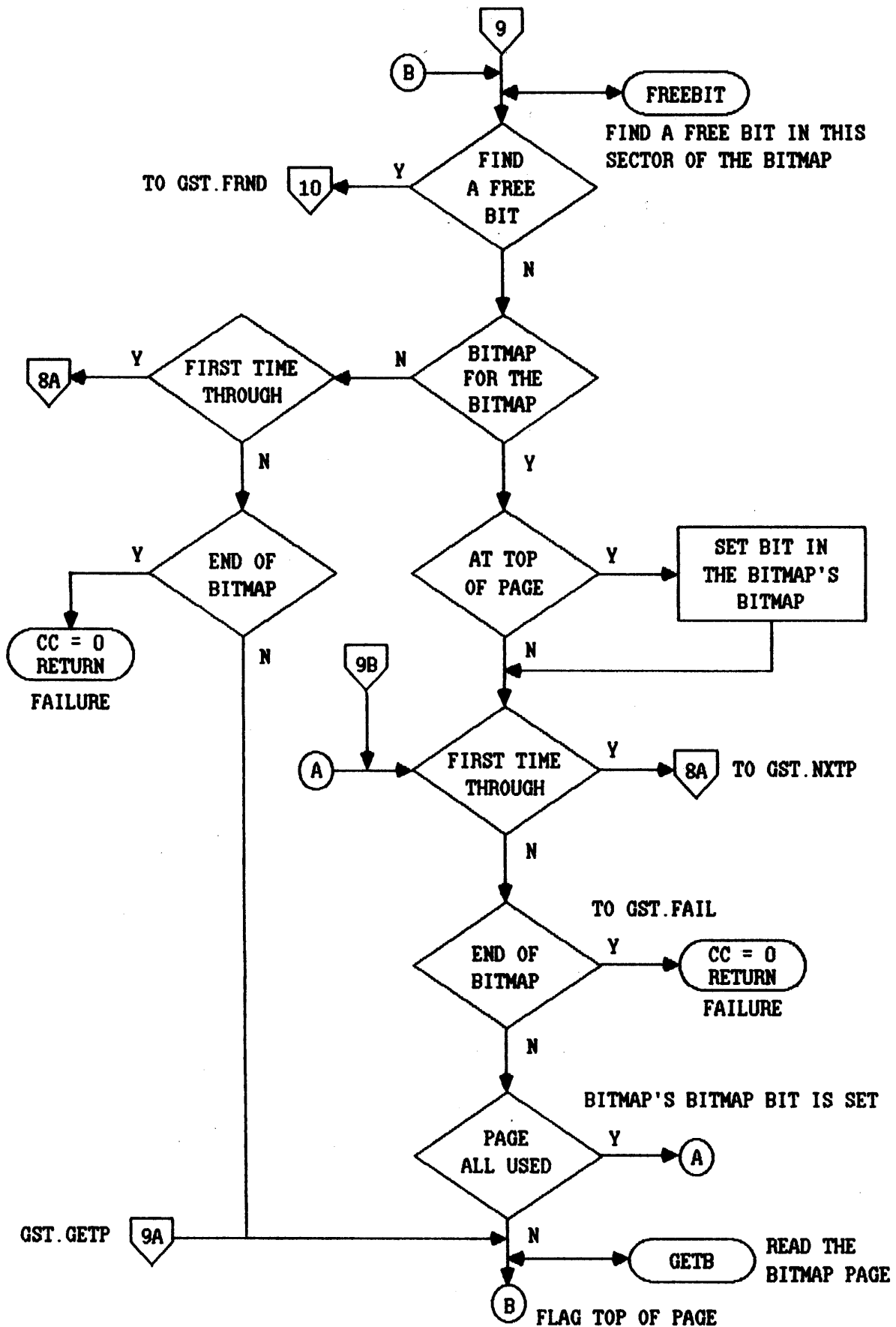


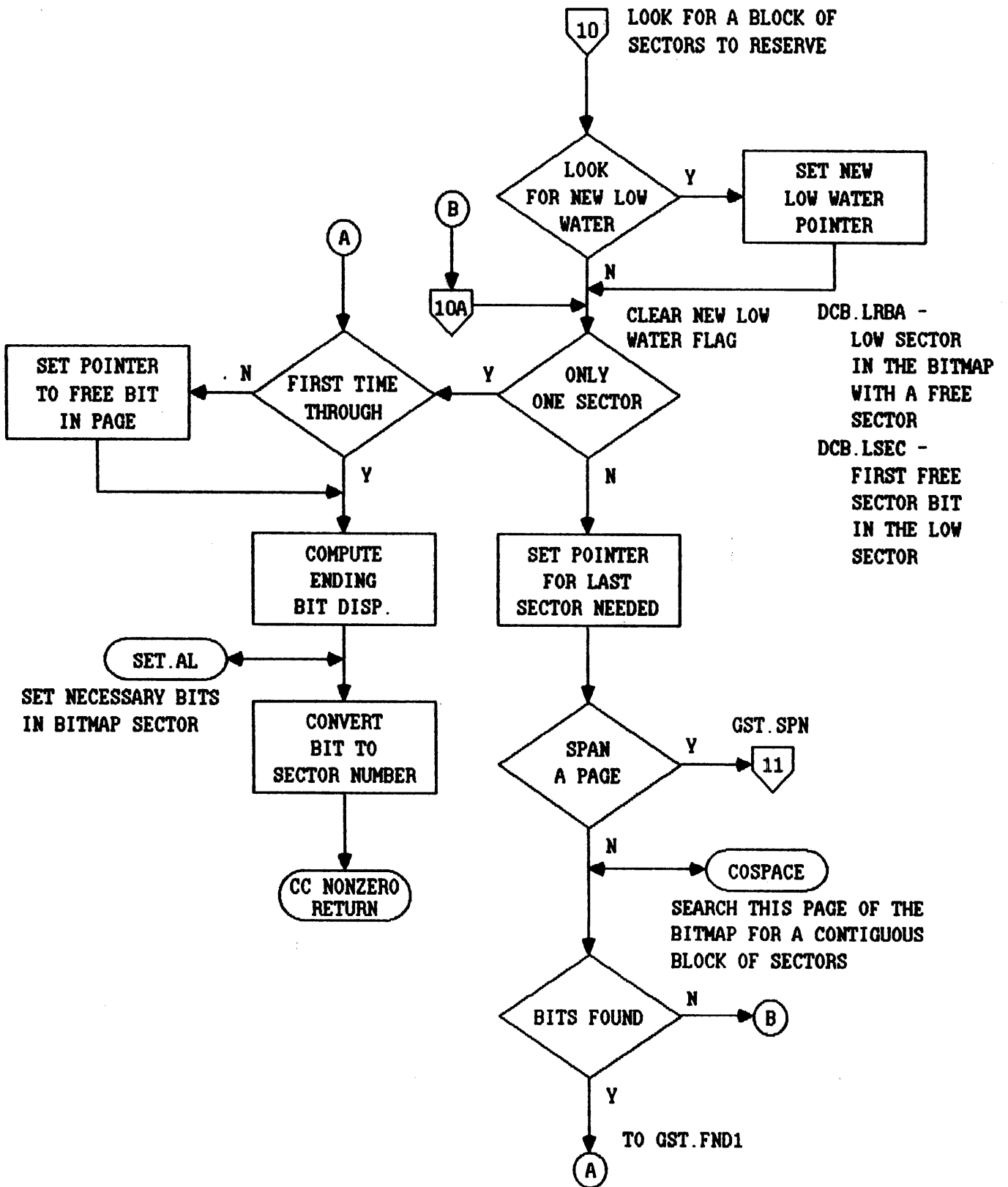


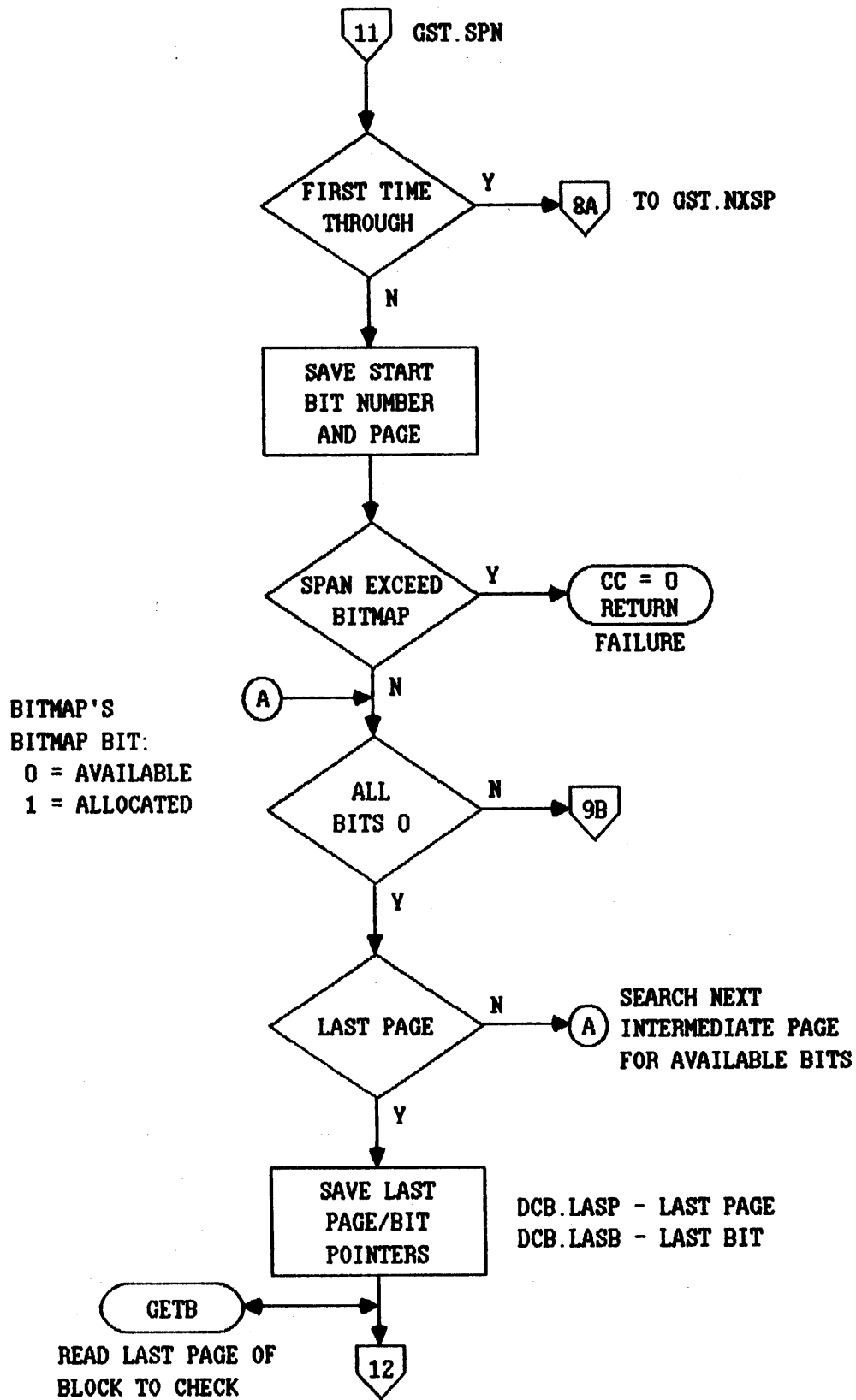


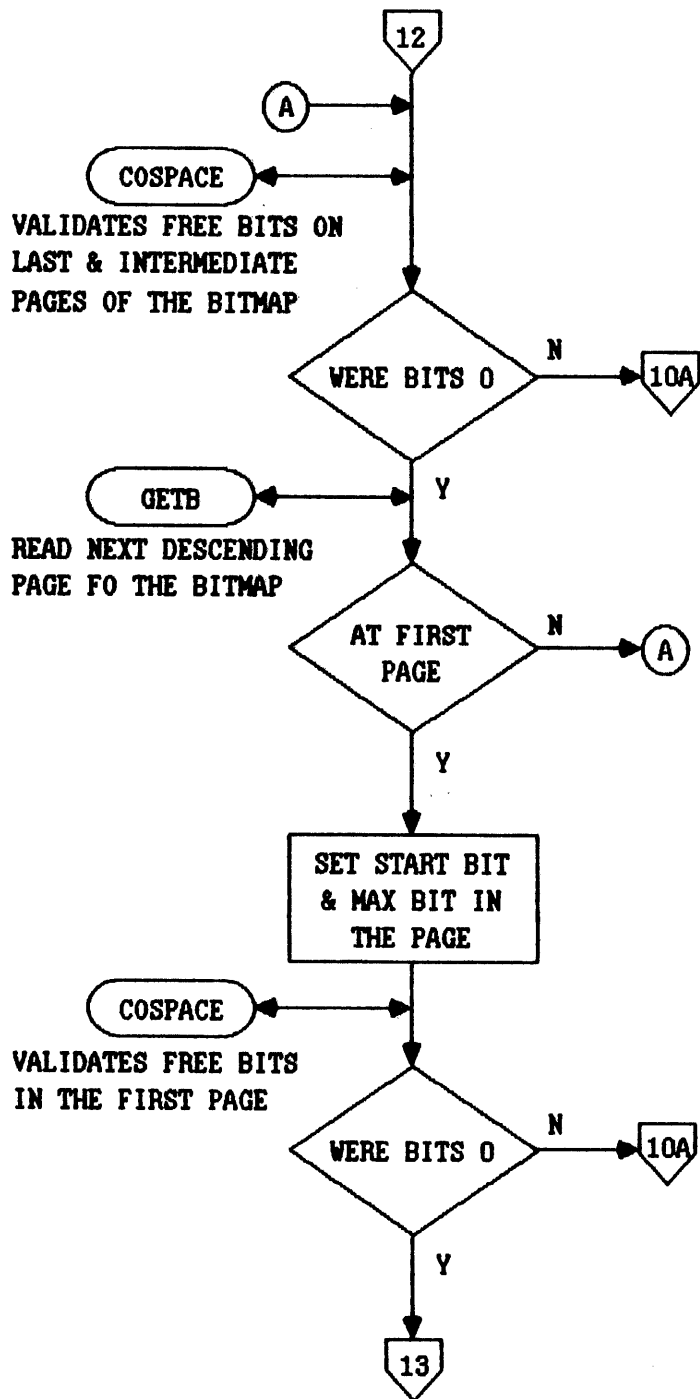


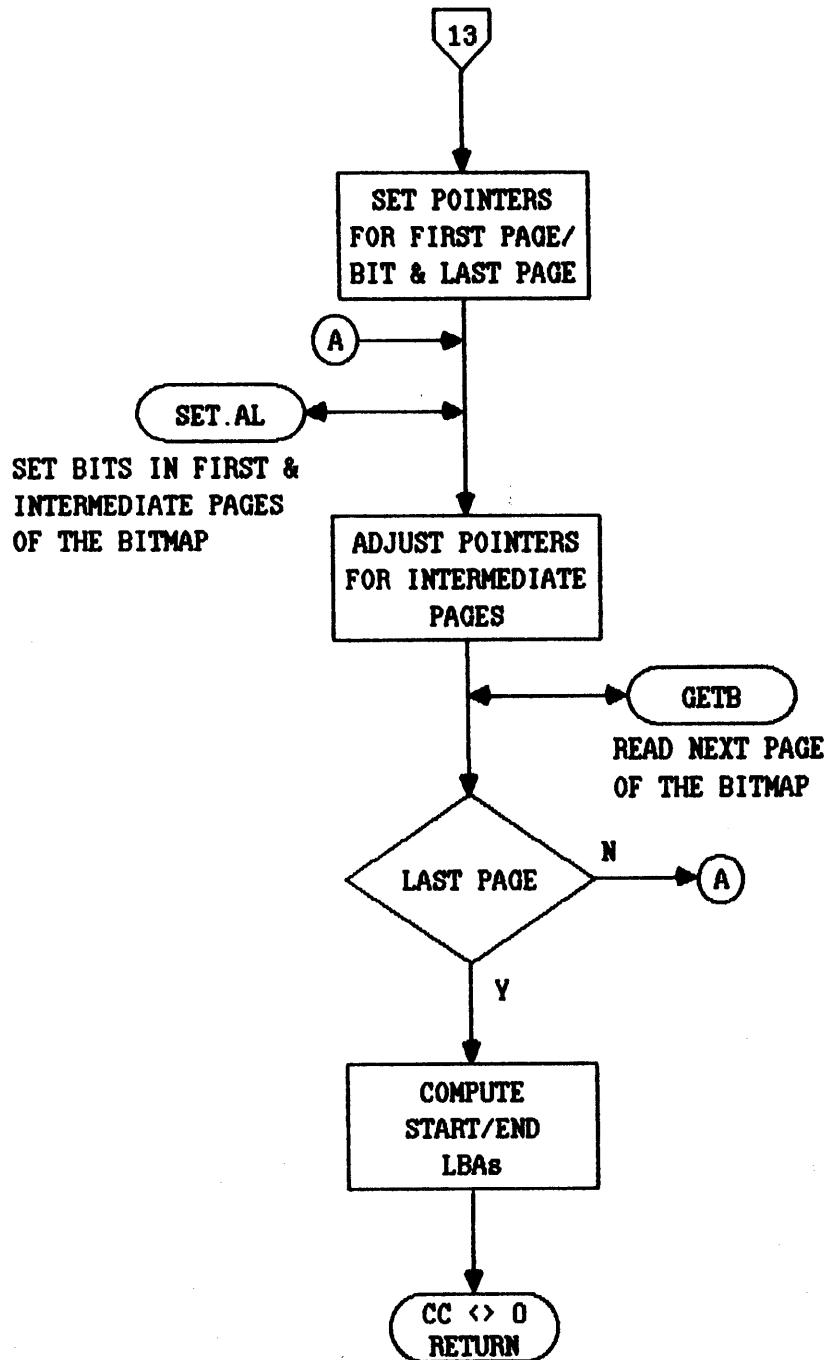


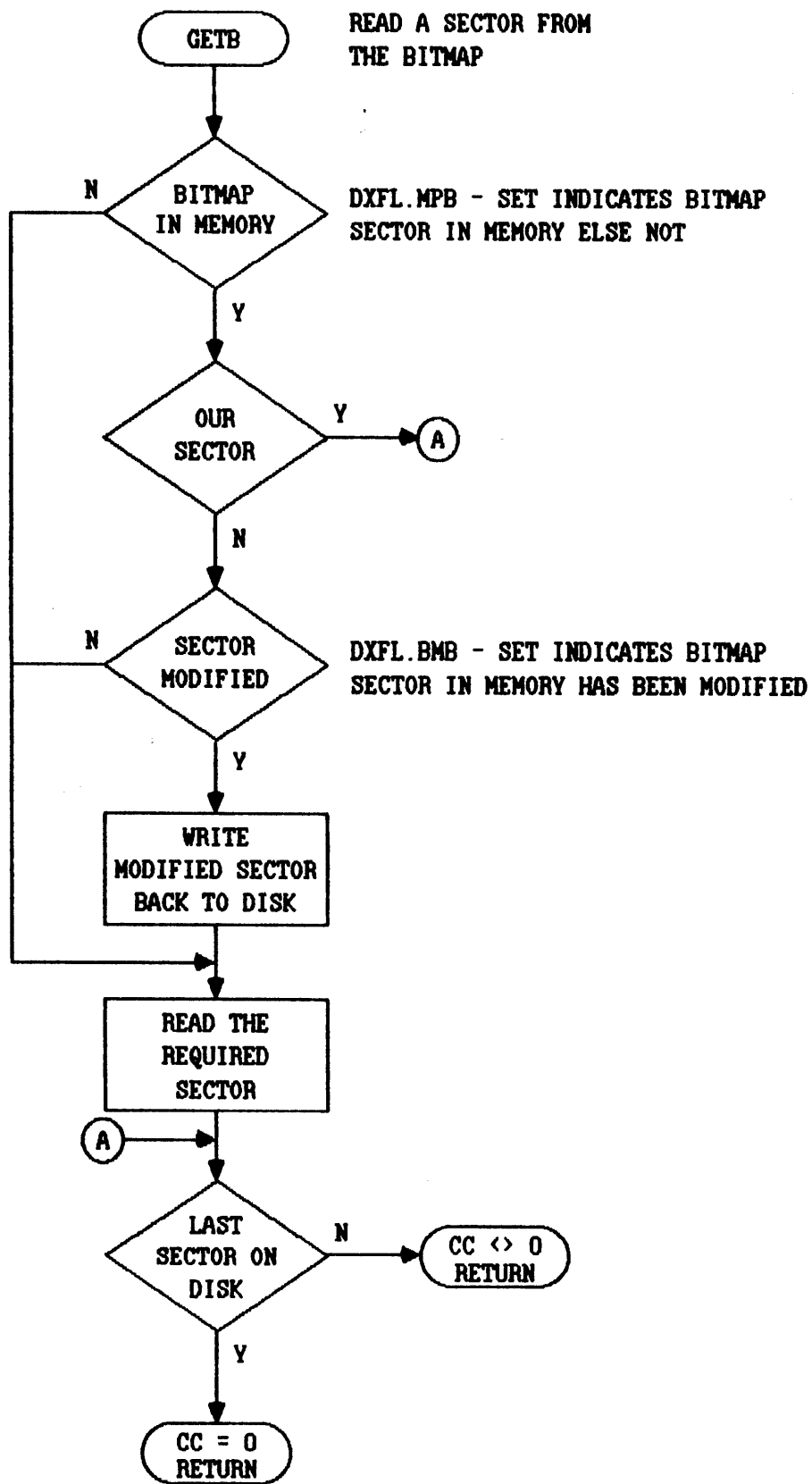


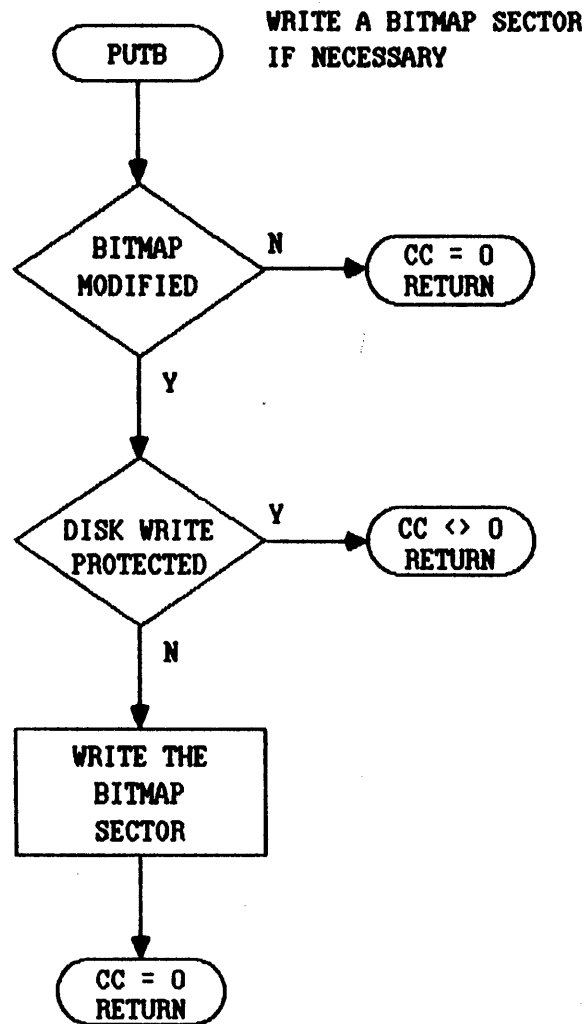


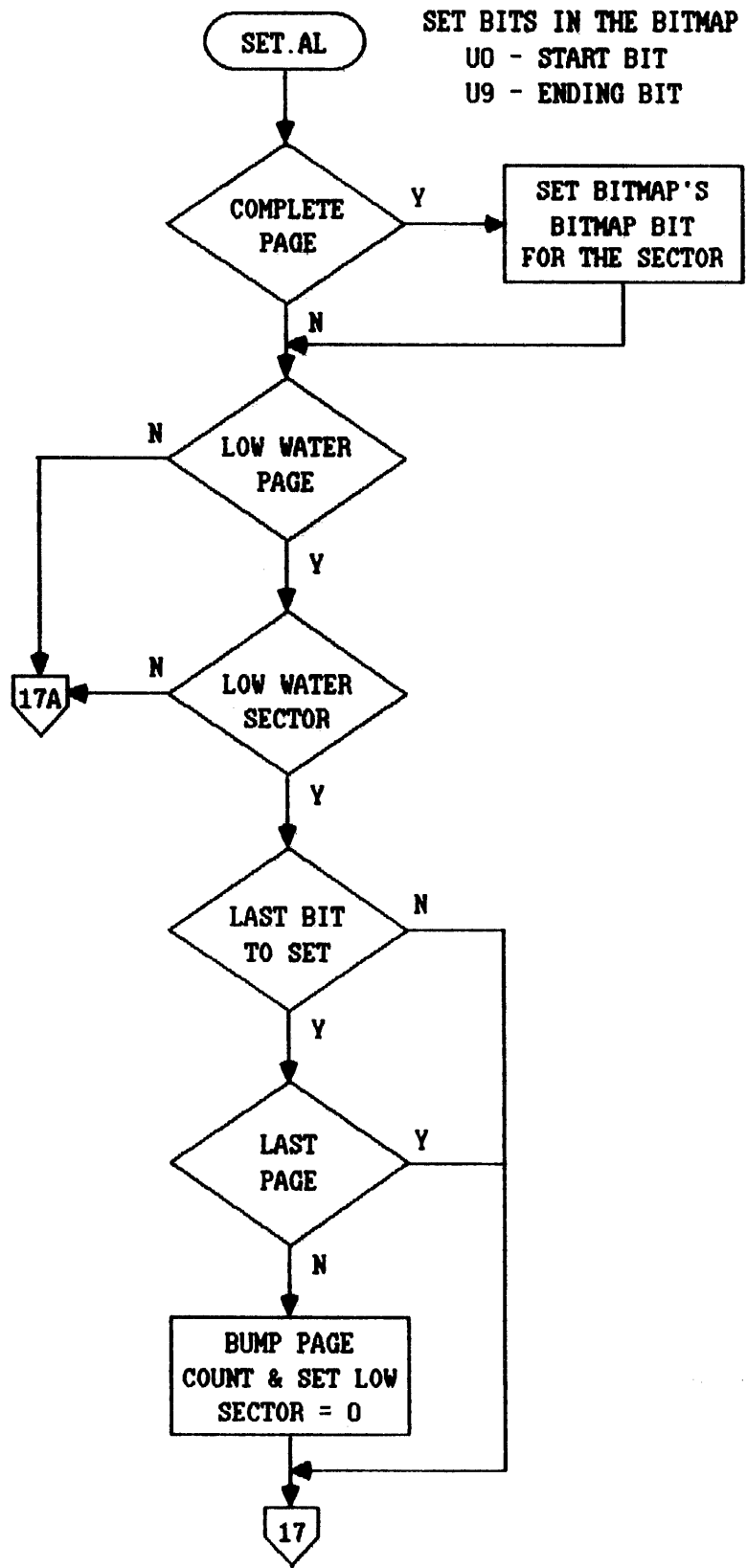


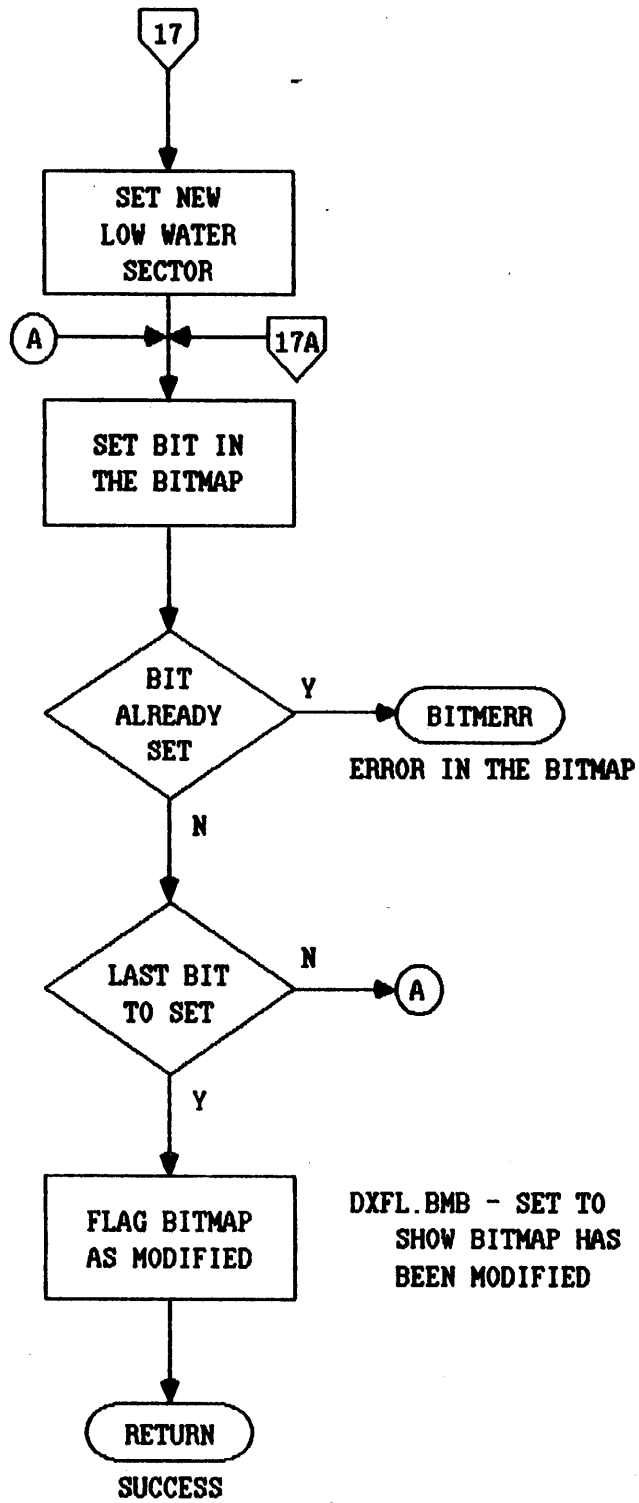


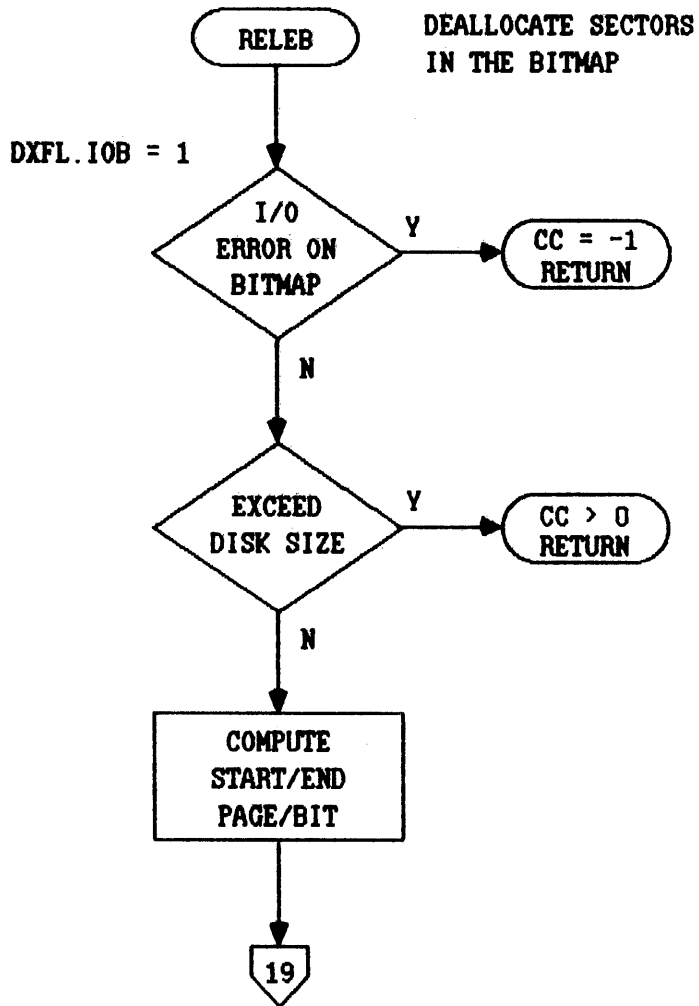


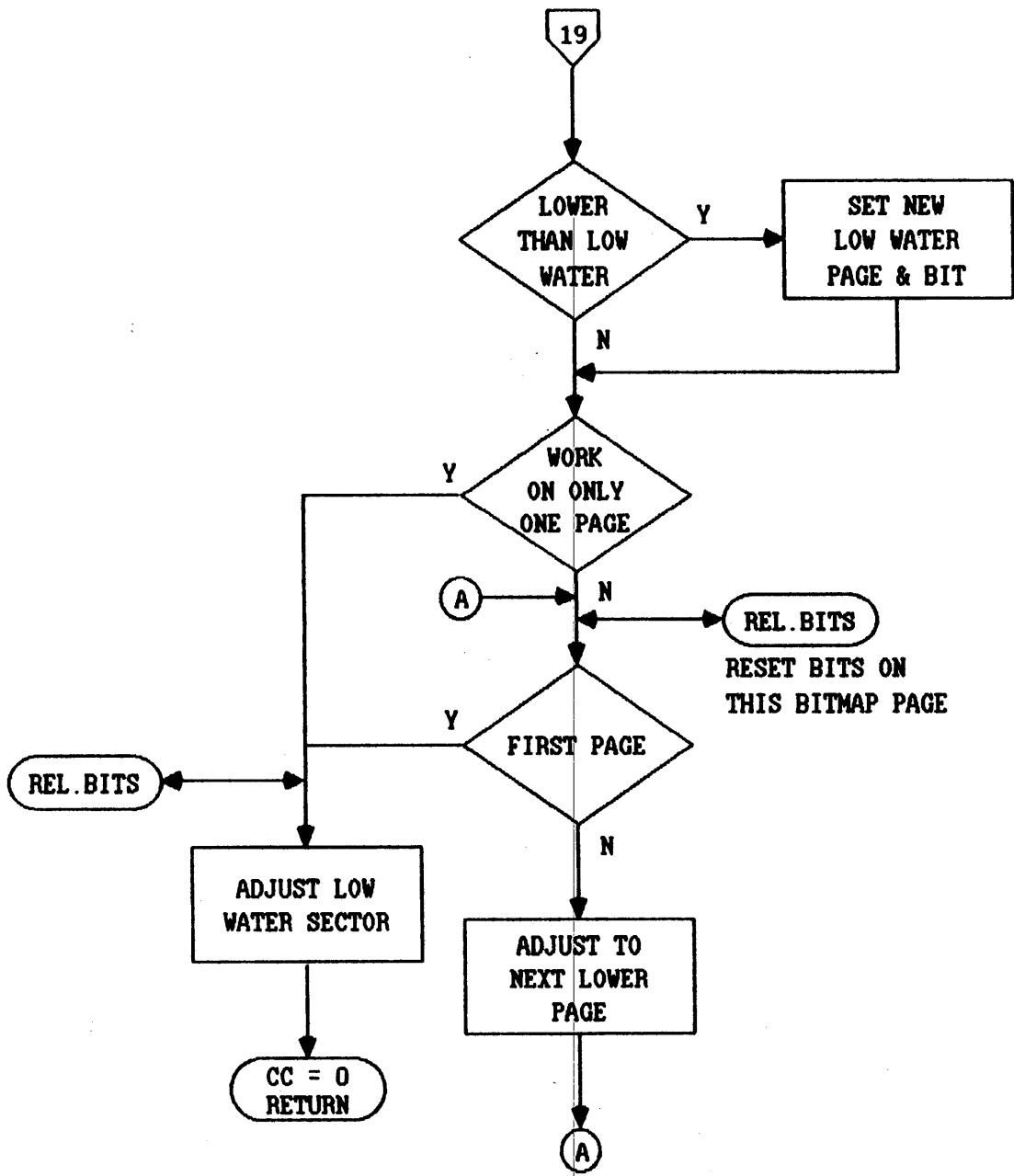


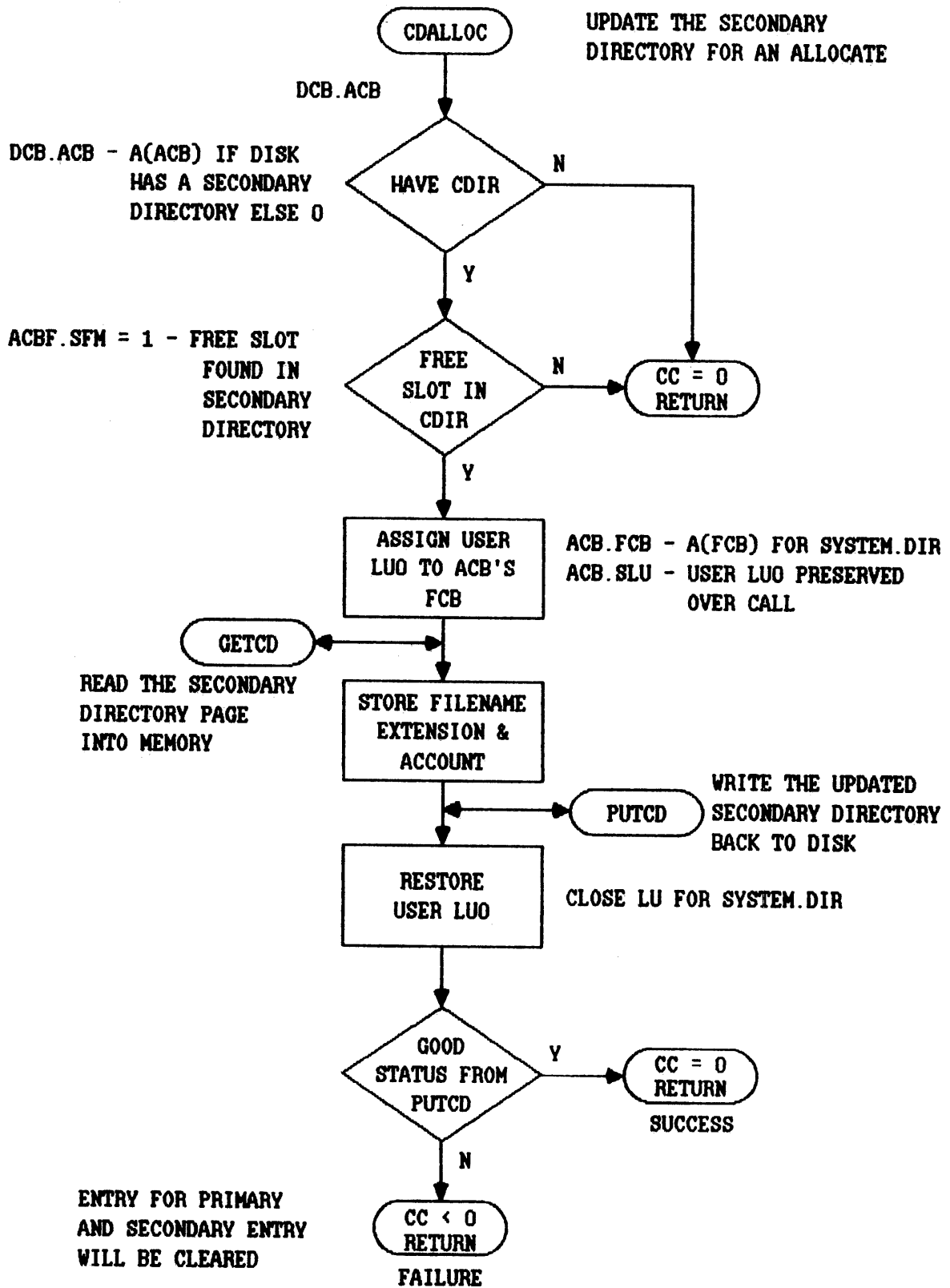


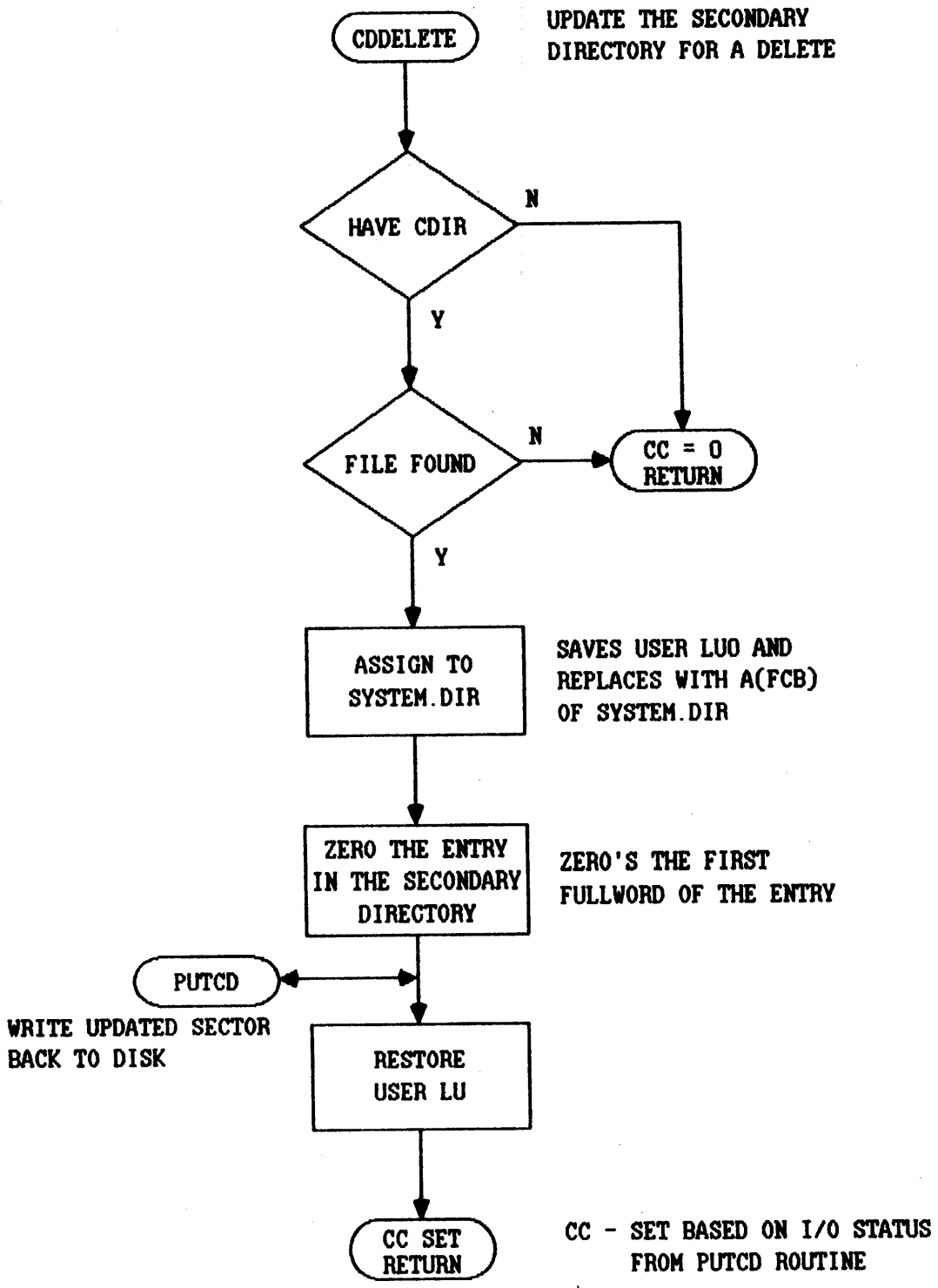




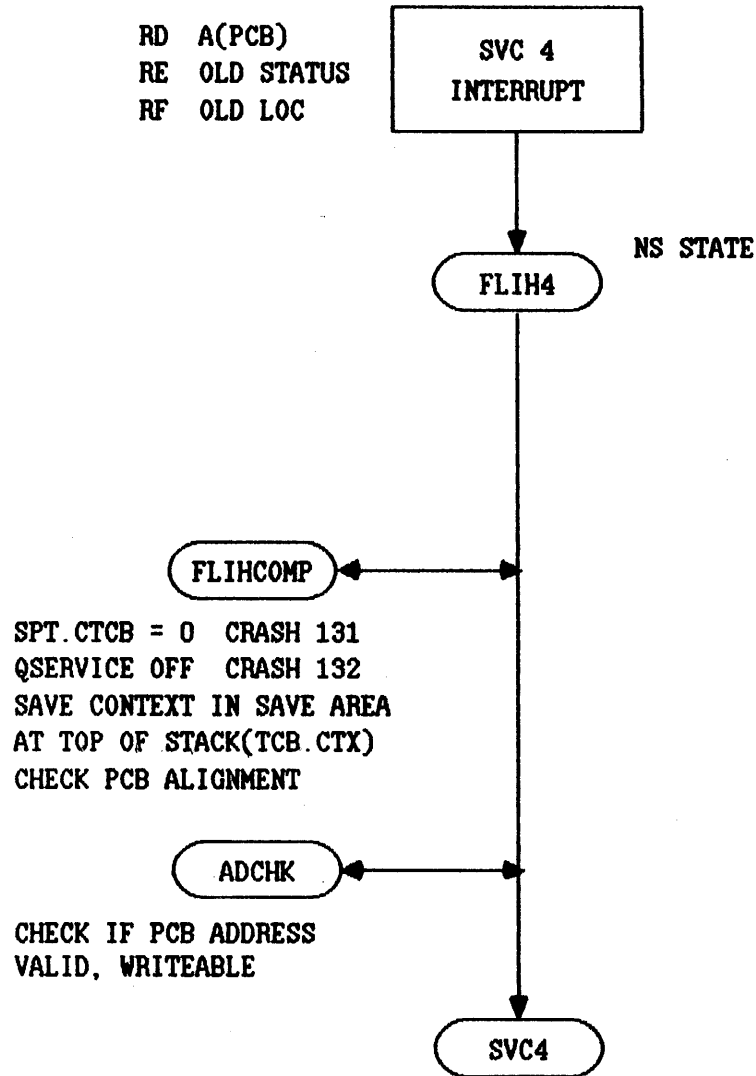


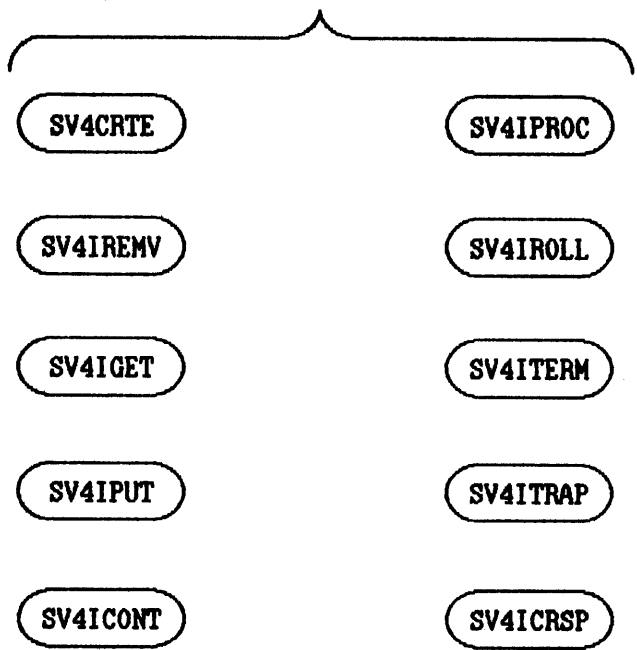
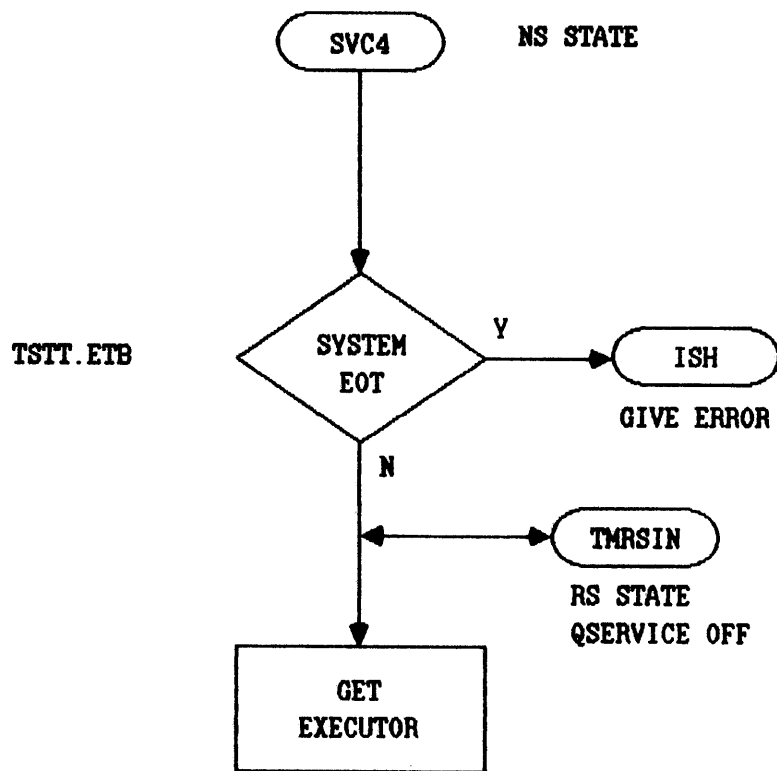






SVC INTERCEPT





SV4ICRTE

RS STATE
QSERVICE OFF

GET
EXECUTOR

INTC1CL

INTC27CL

INTC3CL

INTC6CL

INT7CL

INTC1RX

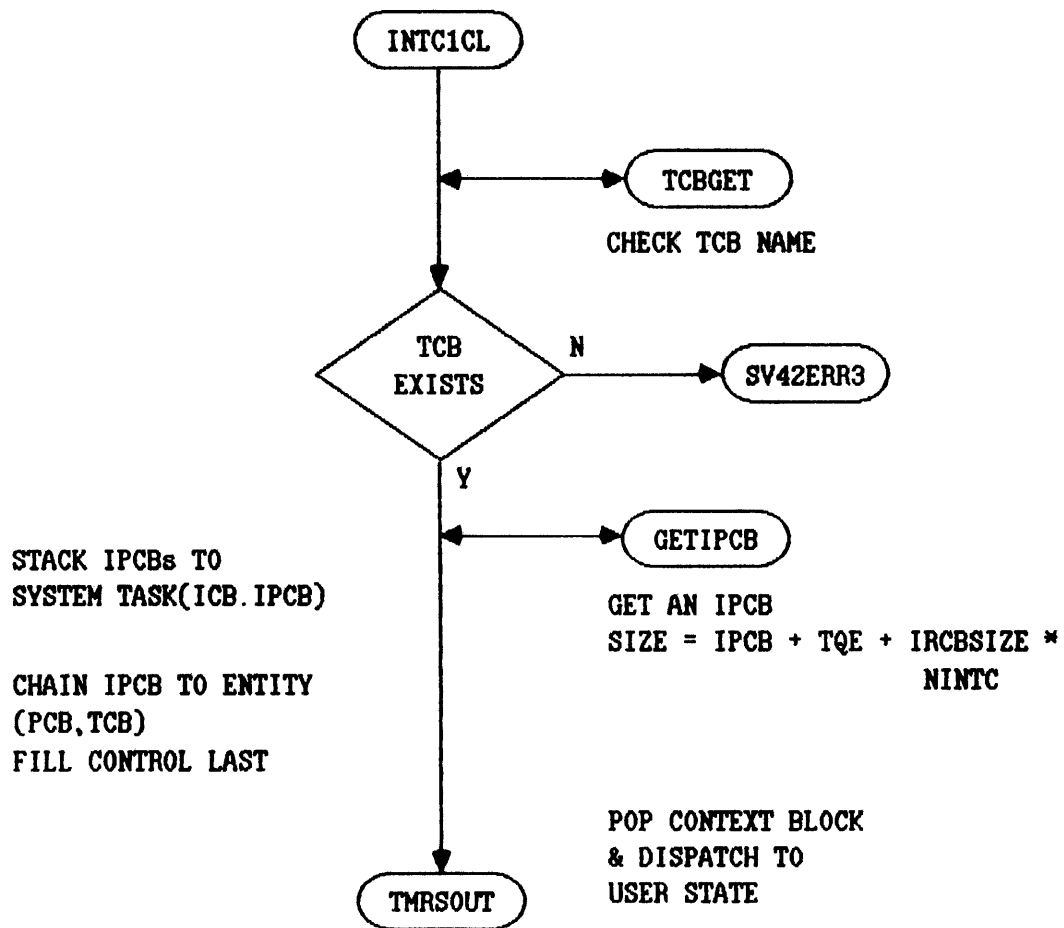
INTC6RX

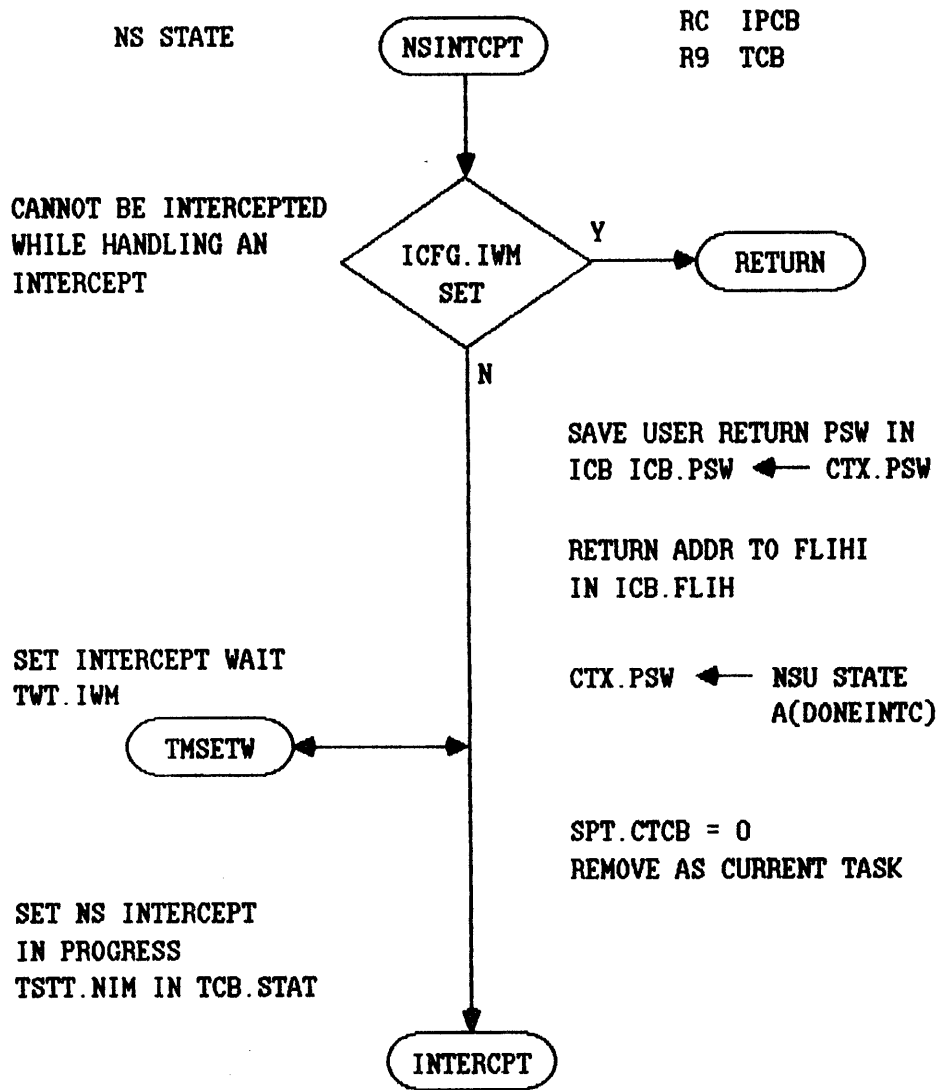
INT7RX

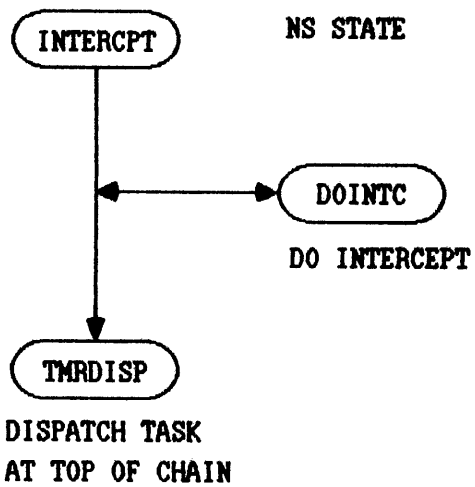
INTC1RN

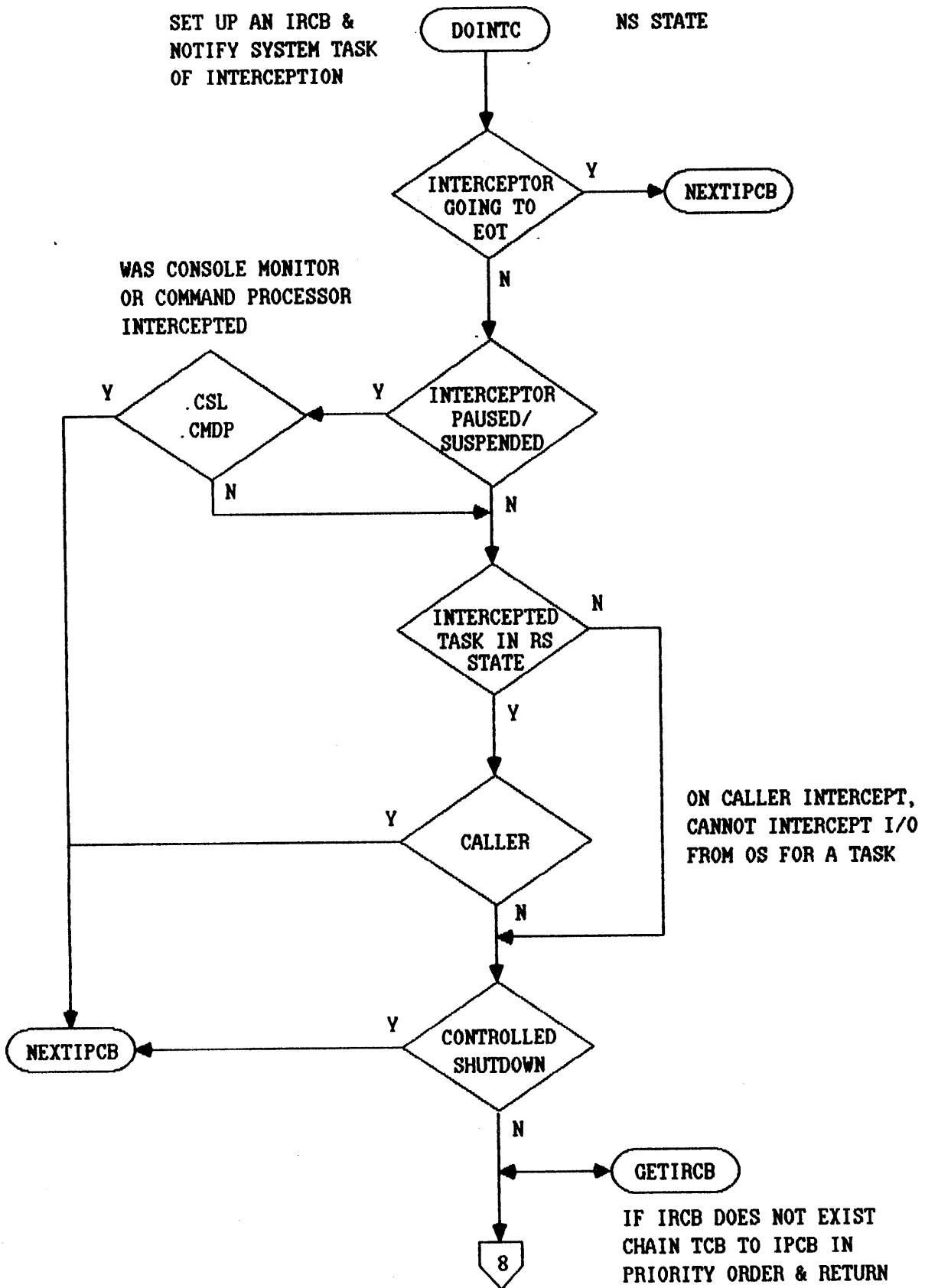
INTC6RN

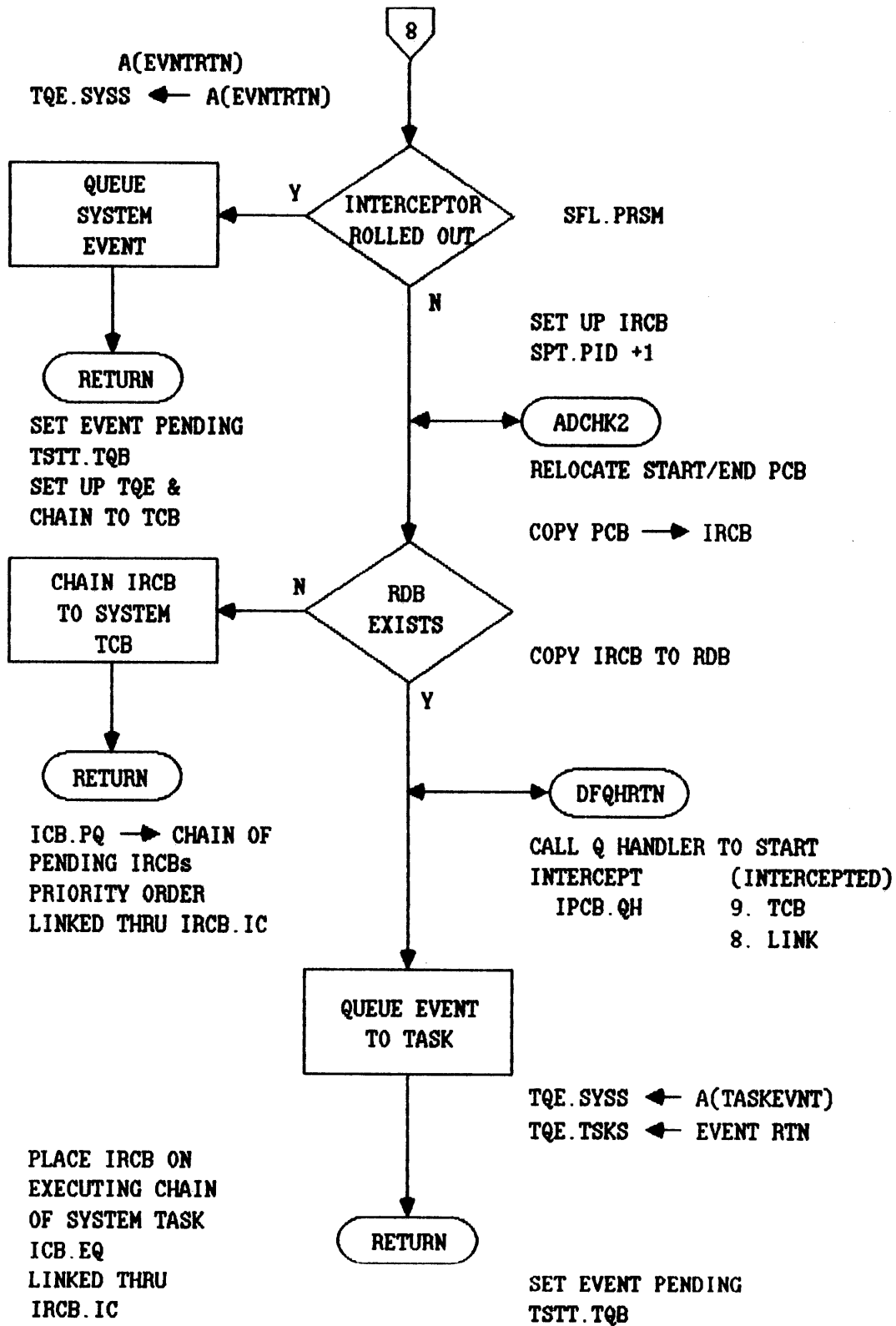
INT7RN



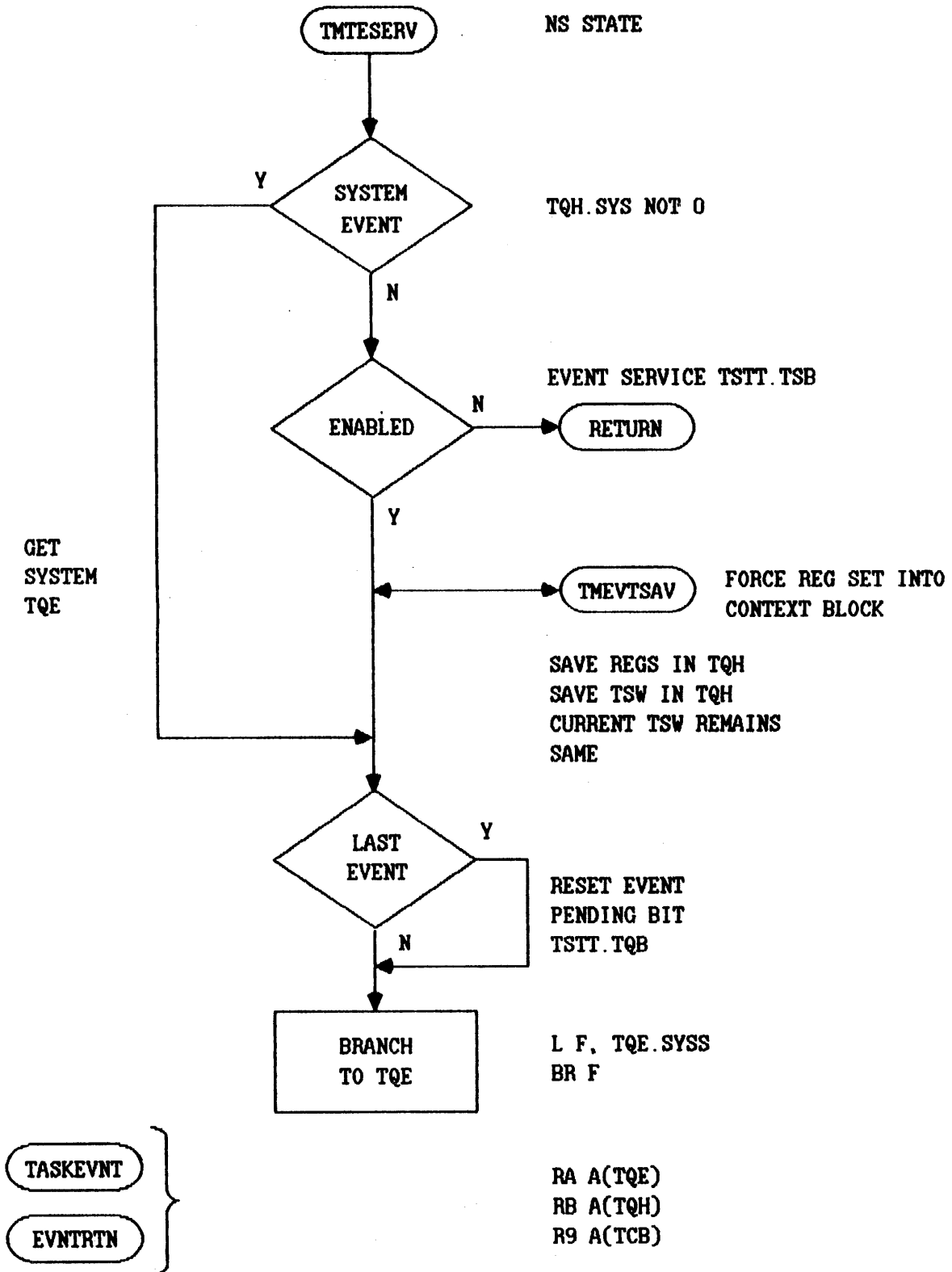








HERE FROM TMCKUTET FINDING TSTT.TQB SET



TASKEVNT

NS STATE

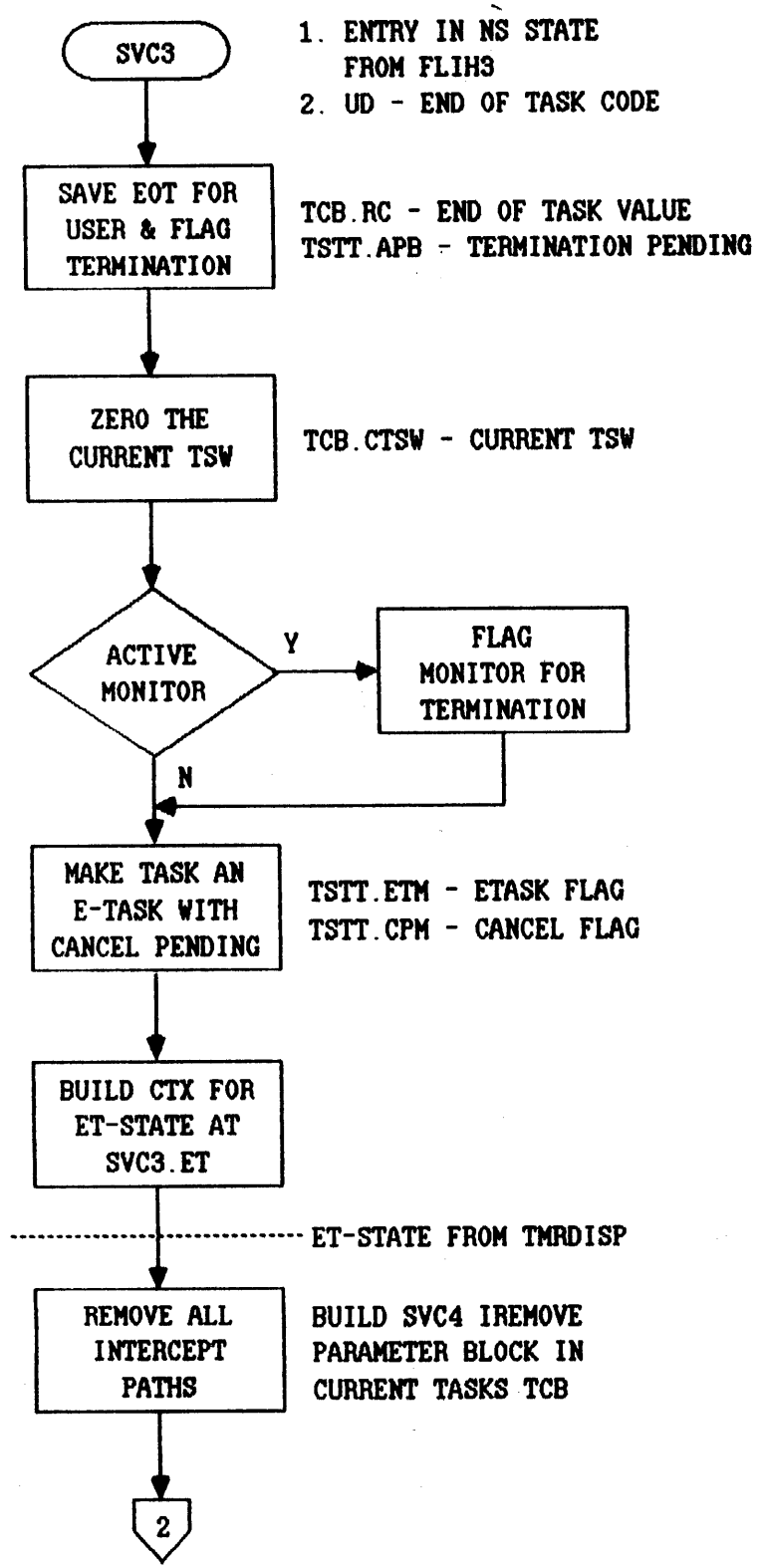
SET UP REGS
R0 PATH ID
R1 A(RDB)

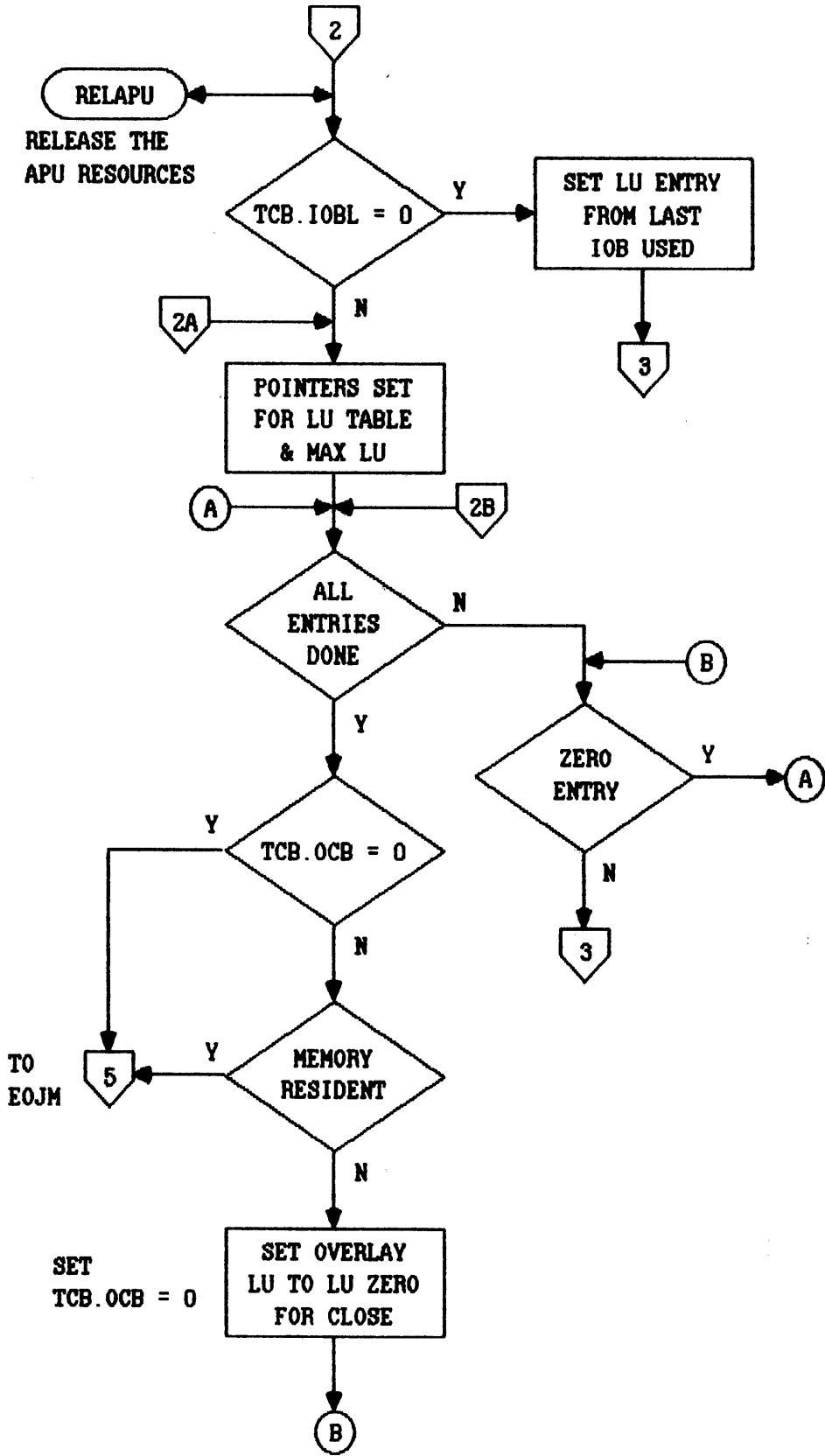
EVENT SERVICE ROUTINE
TO PSW LOC

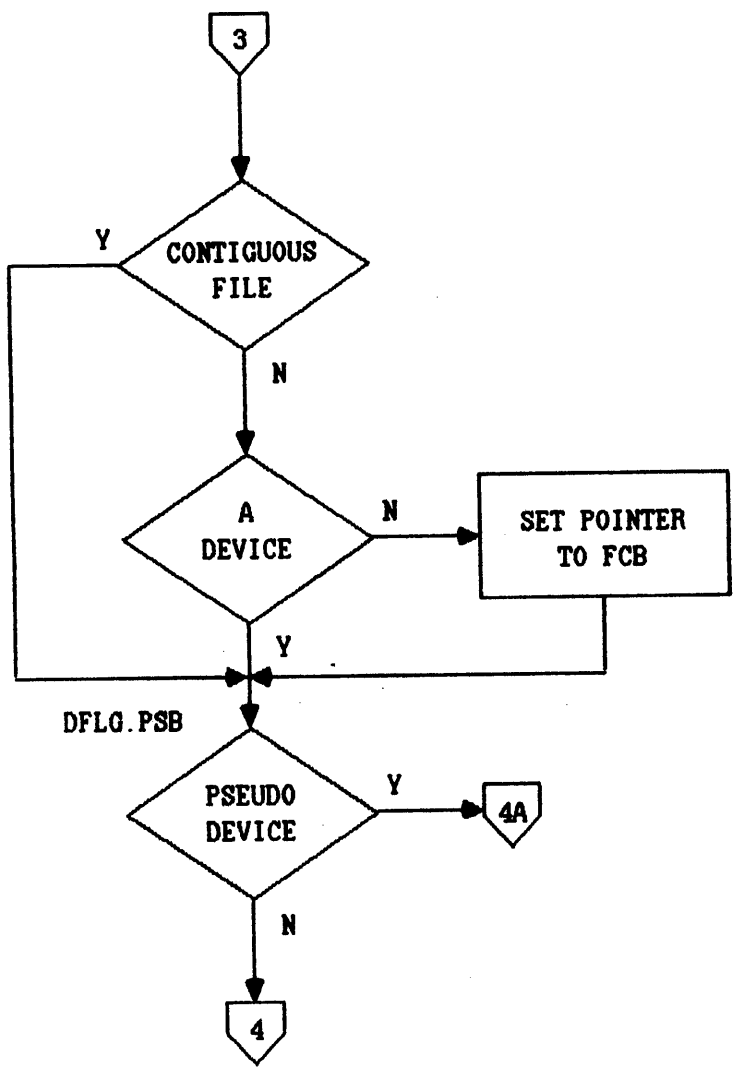
SET TASK EXECUTING
EVENT SERVICE
TSTT.ESB

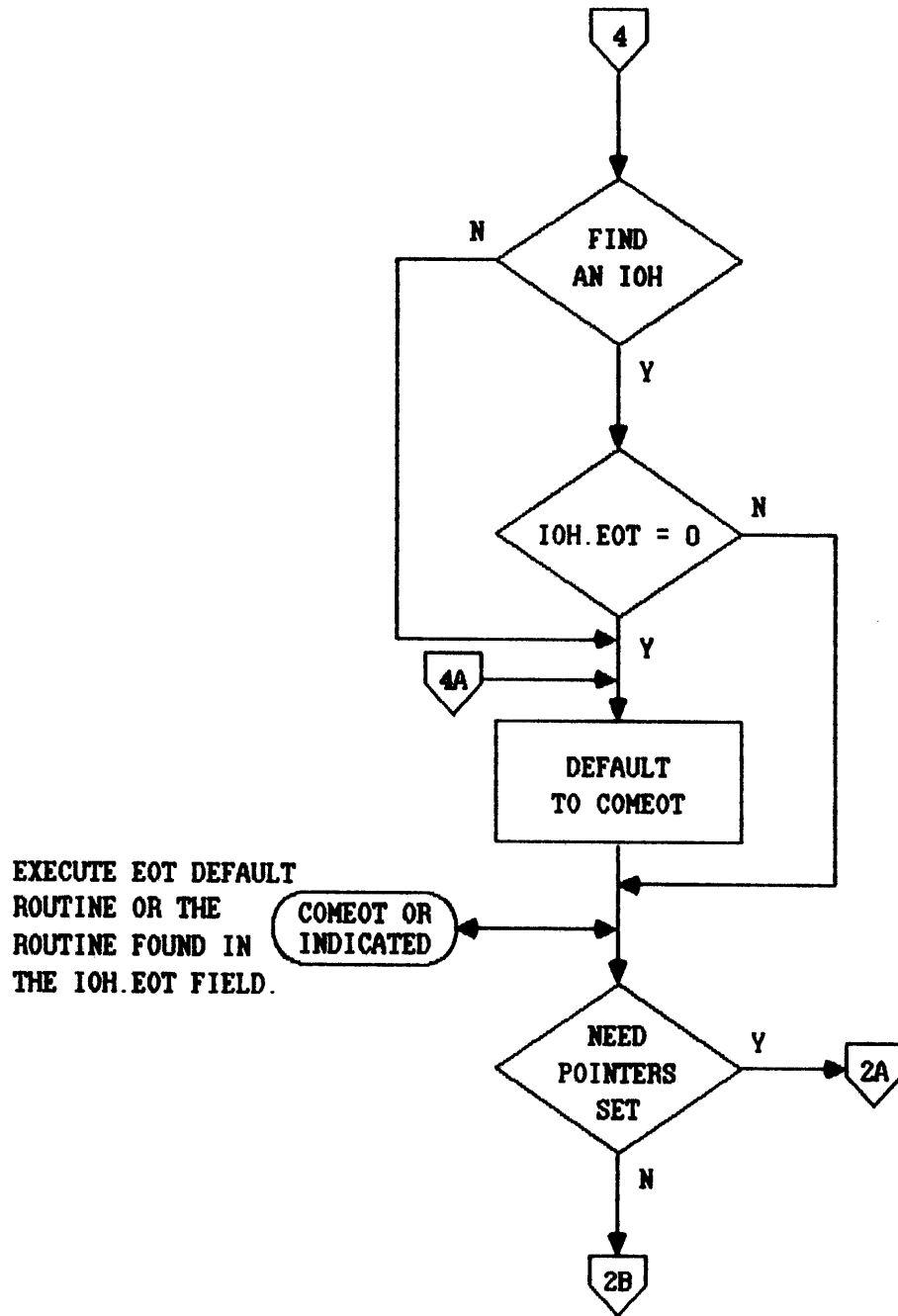
TMRDISP

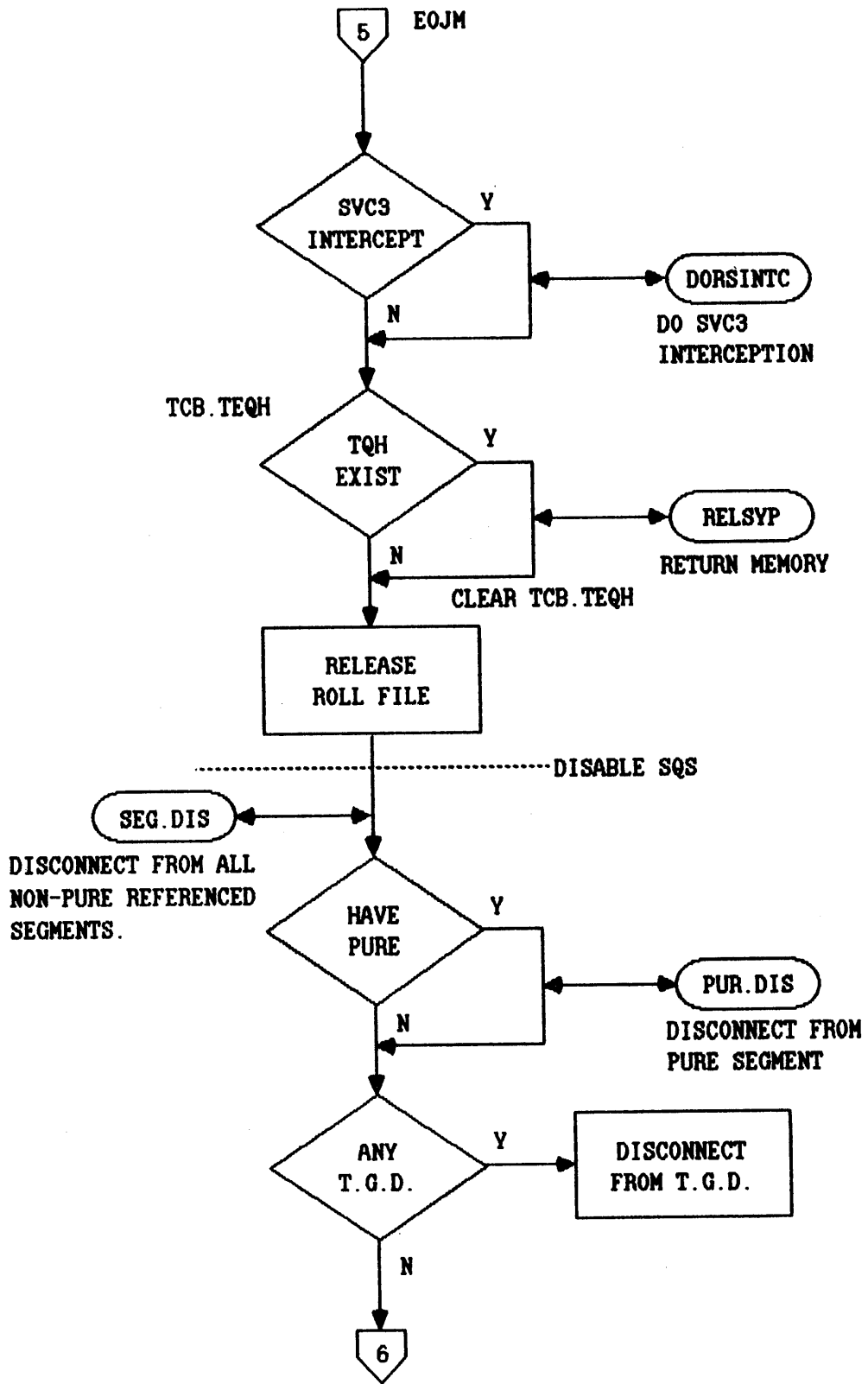
END OF TASK

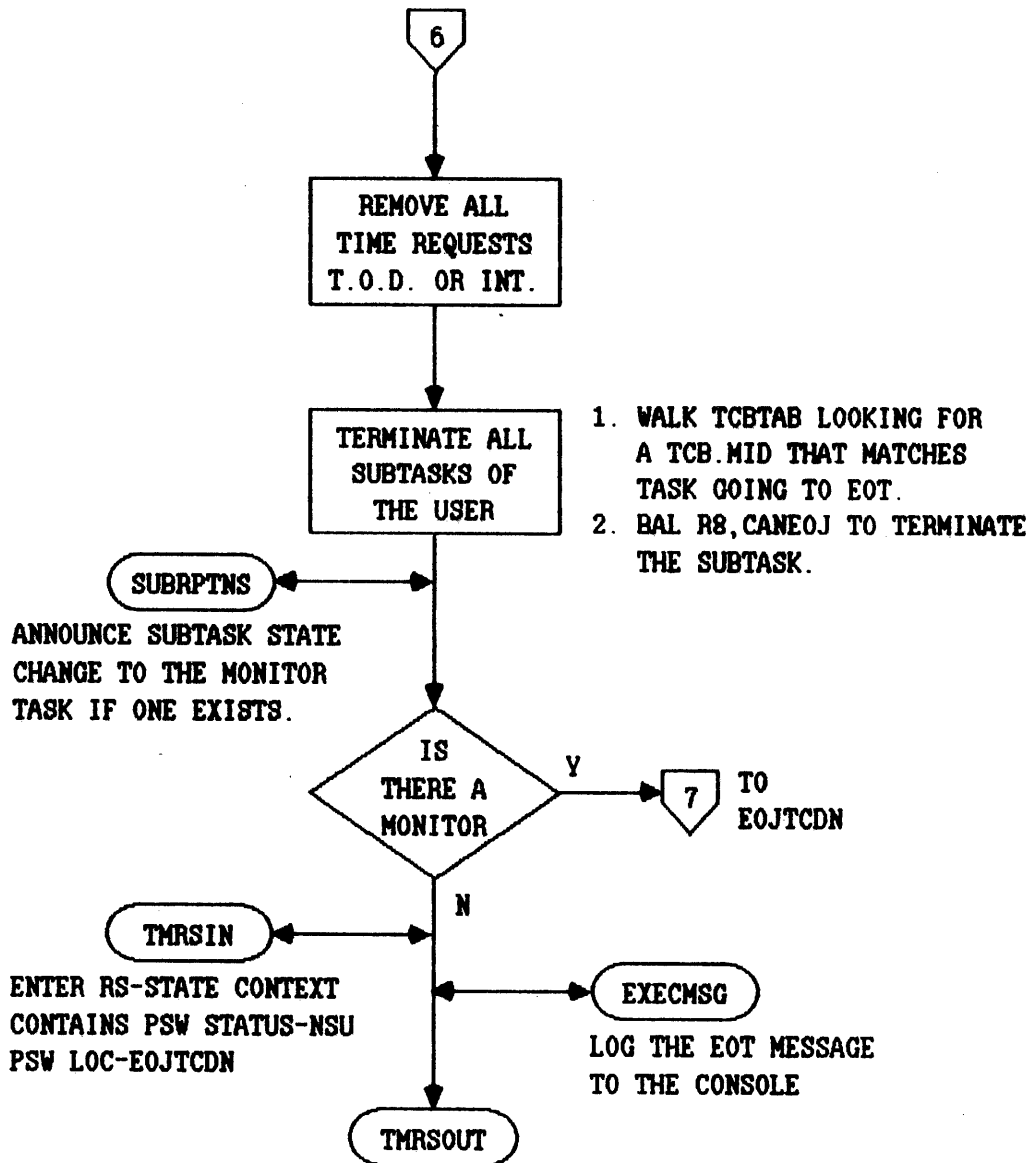




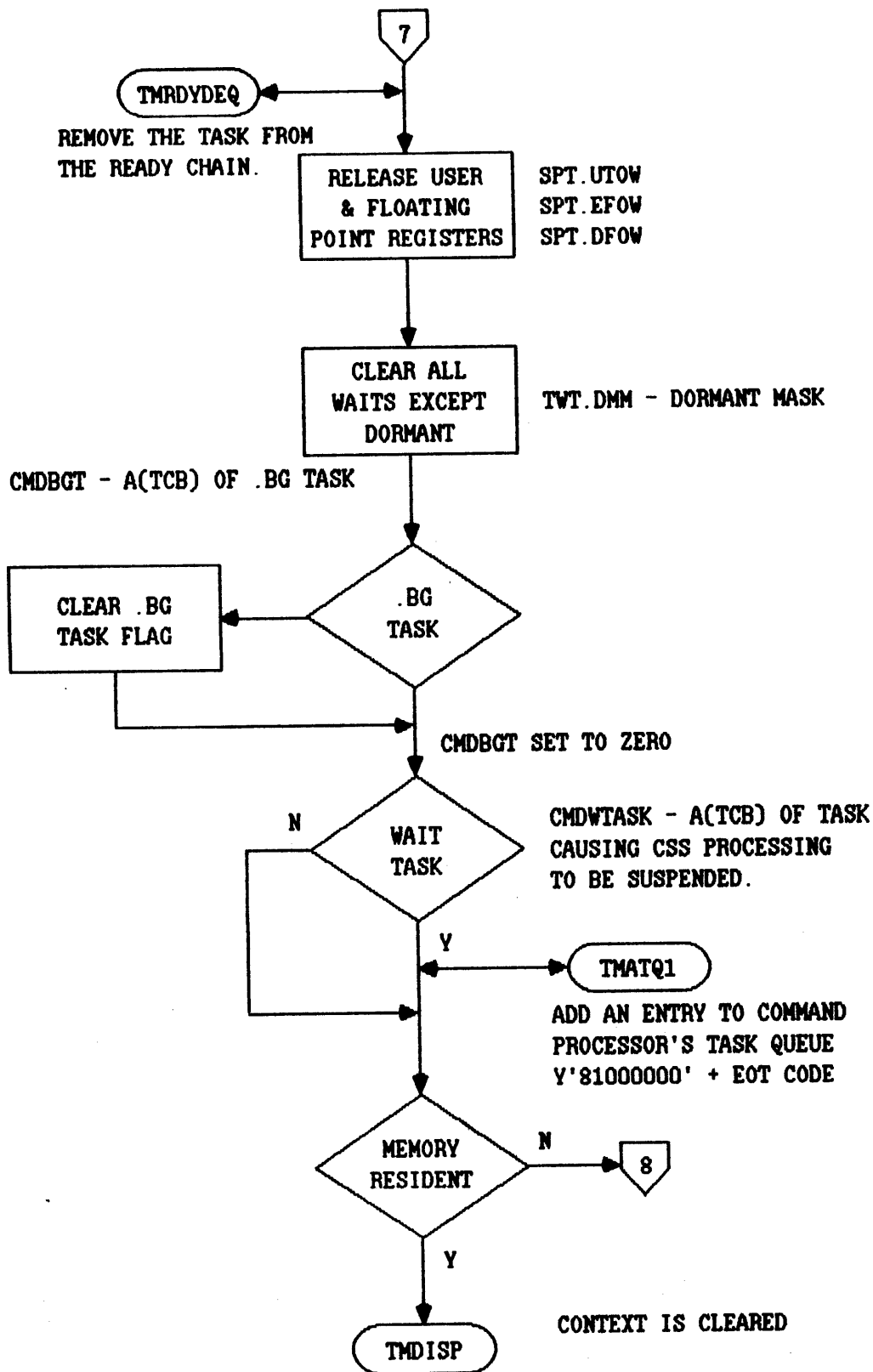


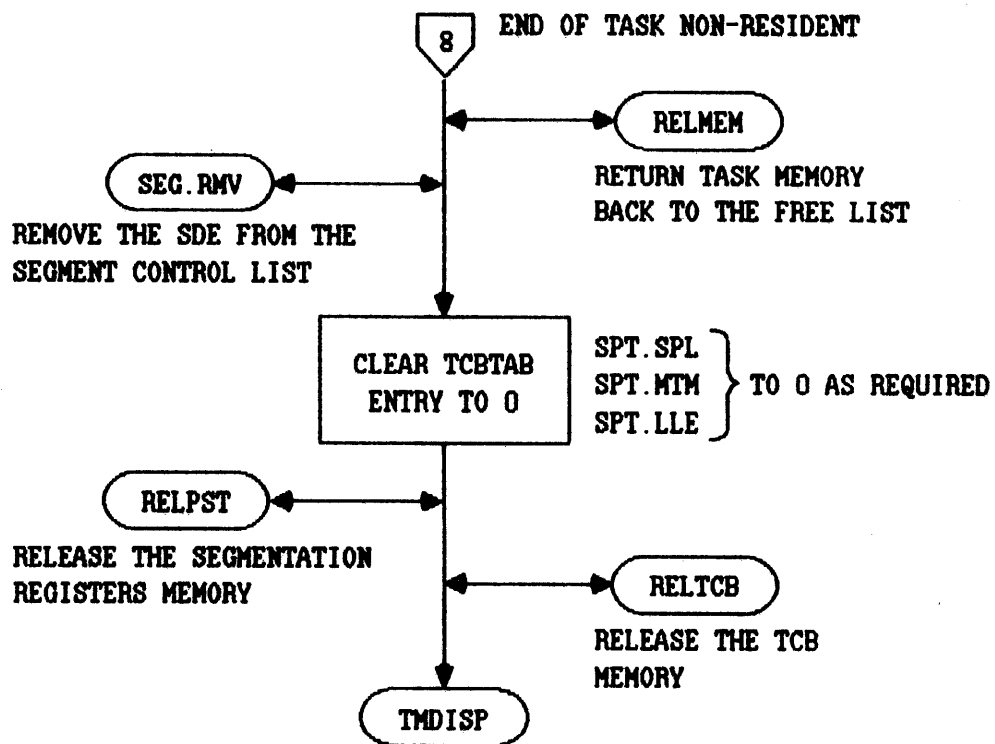




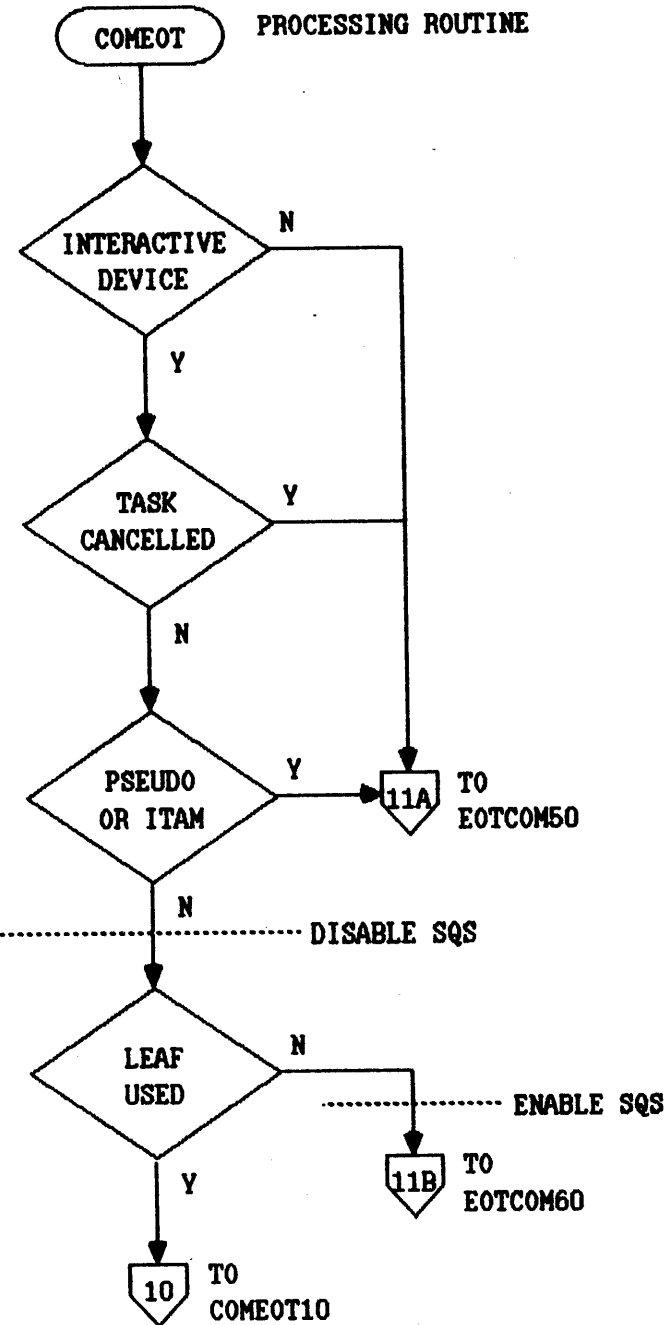


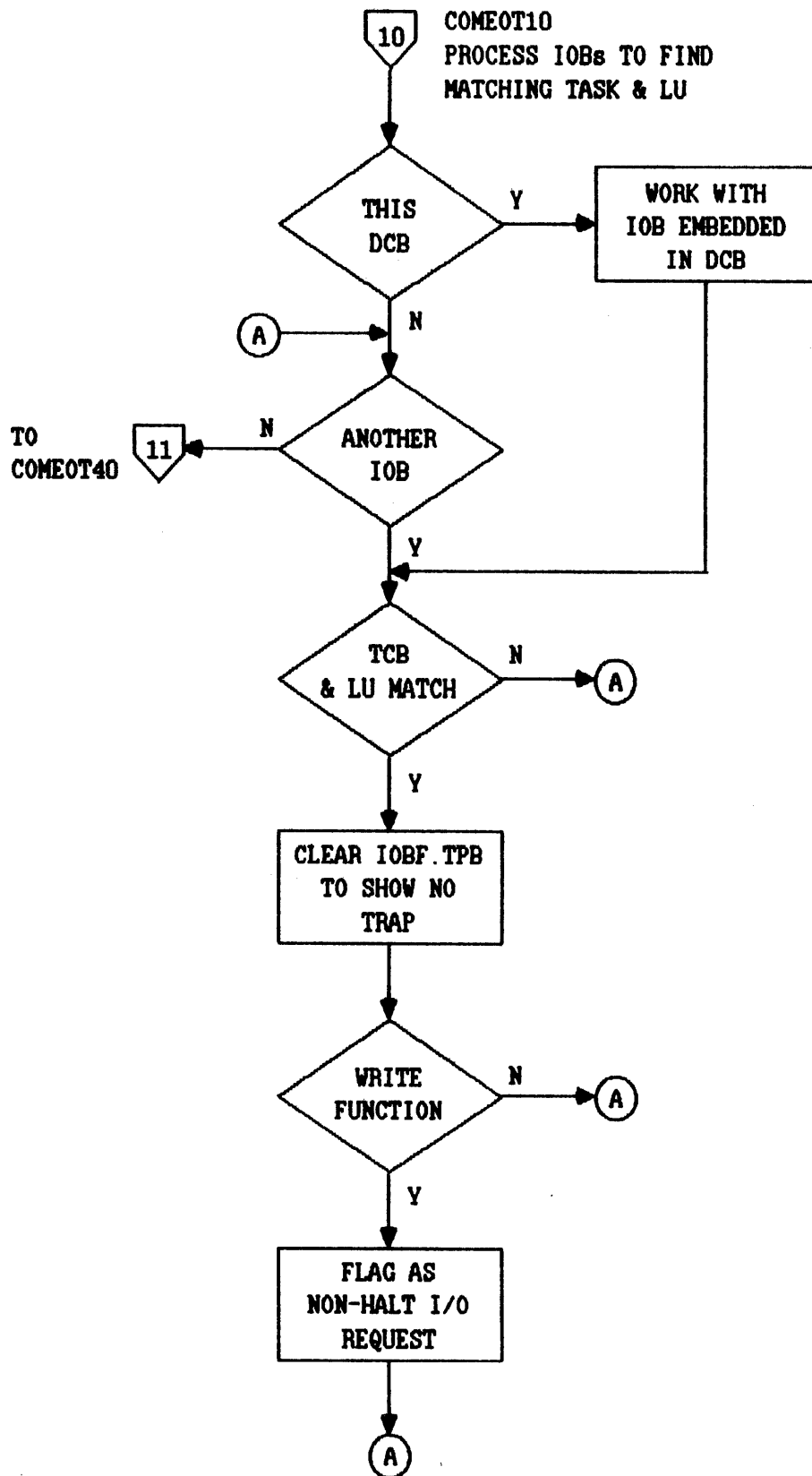
CONTINUE ON PAGE 7



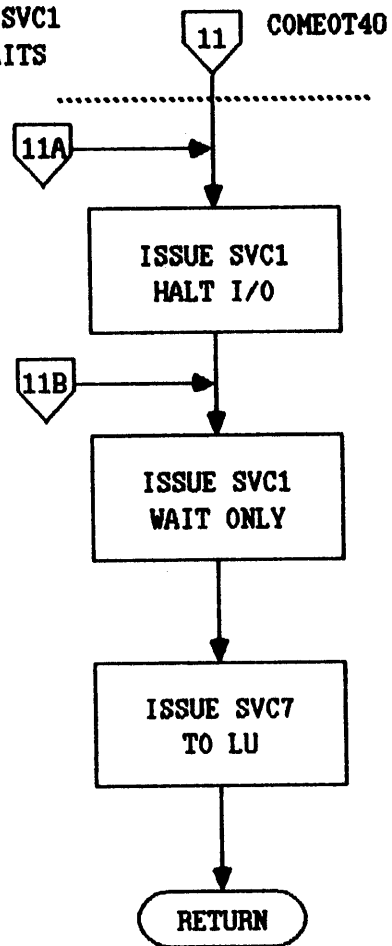


COMMON I/O END OF TASK
PROCESSING ROUTINE





ENTER ET-STATE
TO ISSUE SVC1
HALT & WAITS



INTERRUPT HANDLER

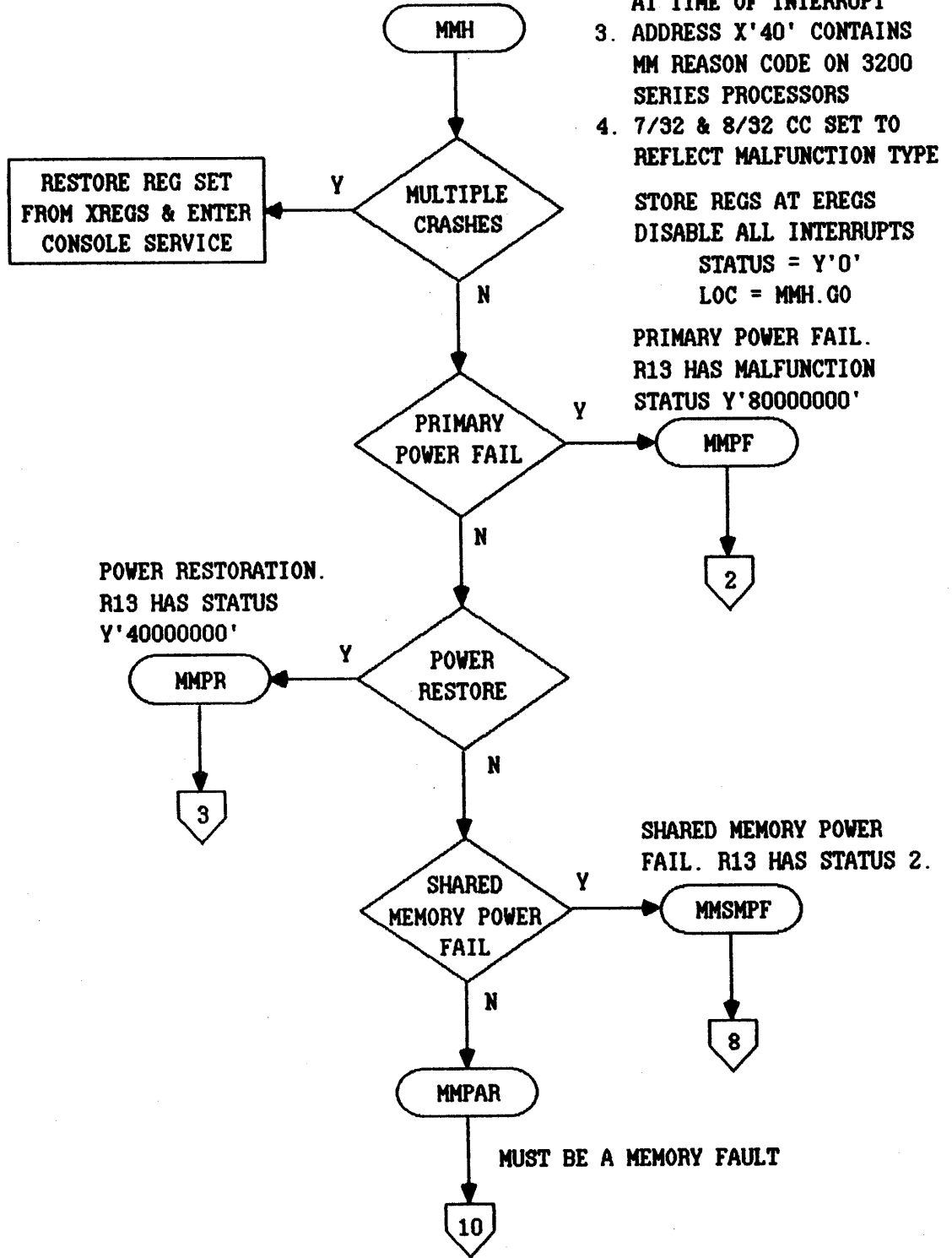
OS/32 INTERNALS
LOW CORE ADDRESSING

<u>ADDRESS IN HEX</u>	<u>LABEL</u>	<u>DESCRIPTION</u>
0-1F	FPREGS	Single precision floating point save area for the 7/32 and 8/32 systems.
20-27	MMOPSW	Machine Malfunction old PSW
28-2B		Reserved
2C-2F	MINFO	Machine Malfunction, LM/STM effective address 3200 series processors.
30-37	IINPSW	Illegal Instruction new PSW.
38-3F	MMNPSW	Machine Malfunction new PSW.
40-43	MMSTAT	Machine Malfunction status word-3200
44-47	MMAFLOC	Memory Malfunction faulting location.
48-4F	AFNPSW	Arithmetic Fault new PSW.
50-57		Initialize the 50 sequence.
58-5F		Reserved.
60-61	SYSSTART	Branch instruction to the label INITBRCH
62-63		Address of the SYSTEM POINTER TABLE.
64-69	INITBRCH	Branch to system initialization the label will be SYSINIT.
6A-6F		Reserved
70-75	PANICBCH	Branch to panic dump routine with the label of PANIC.
76-7F		Reserved.
80-83	SYSQP	Address of the SYSTEM QUEUE.
84-87	PFSAVEP	Auto shut down save area pointers Z(MMASPSW)-- 7/32 and 8/32 Z(MMASREG)-- 7/32 and 8/32 A(MMASREG)-- 3200 series
88-8F	SQSNPSW	SYSTEM QUEUE SERVICE new PSW.
90-97	MFNPSW	EAT/MAC Fault new PSW.

LOW CORE ADDRESSING (continued)

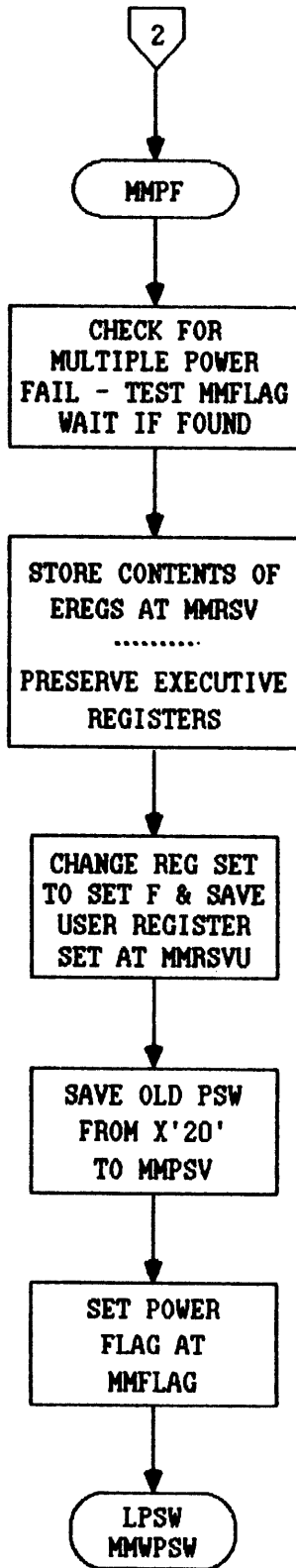
ADDRESS IN HEX	LABEL	DESCRIPTION
98-9B	SVCNSTAT	New status for PSW all SVCs
9C-9D	SVCNLOC	Z (FLIH0) SVC 0
9E-9F		Z (FLIH1) SVC 1
A0-A1		Z (FLIH2) SVC 2
A2-A3		Z (FLIH3) SVC 3
A4-A5		Z (FLIH4) SVC 4 (SVC INTERCEPT)
A6-A7		Z (FLIH5) SVC 5
A8-A9		Z (FLIH6) SVC 6
AA-AB		Z (FLIH7) SVC 7
AC-AD		Z (ISHPUR) SVC 8 (ILLEGAL)
AE-AF		Z (FLIH9) SVC 9
B0-B1		Z (FLIH10) SVC 10 (AUTO OVERLAY)
B2-B3		Z (ISHPUR) SVC 11 (ILLEGAL)
B4-B5		Z (ISHPUR) SVC 12 (ILLEGAL)
B6-B7		Z (FLIH13) SVC 13 (MPS SYSTEMS ONLY)
B8-B9		Z (FLIH14) SVC 14
BA-BB		Z (FLIH15) SVC 15 (COMMUNICATIONS)
BC-BF		Reserved.
C0-C3		Auxiliary Processor Block Directory (3200MPS only)
C4-C7		Max APU number (3200MPS only)
C8-CF	ALDFNPSW	Data Format/Alignment Fault new PSW 3200 series only
D0-2FF	ISPTAB	Interrupt Service Pointer Table supporting 255 device.
-4FF		Expanded Interrupt Service Pointer Table supporting 511 devices.
-8FF		Expanded Interrupt Service Pionter Table supporting 1023 devices.

MACHINE MALFUNCTION
GENERAL HANDLERS



POWER FAIL
 USER REG SET SAVED
 AT MMRSVU

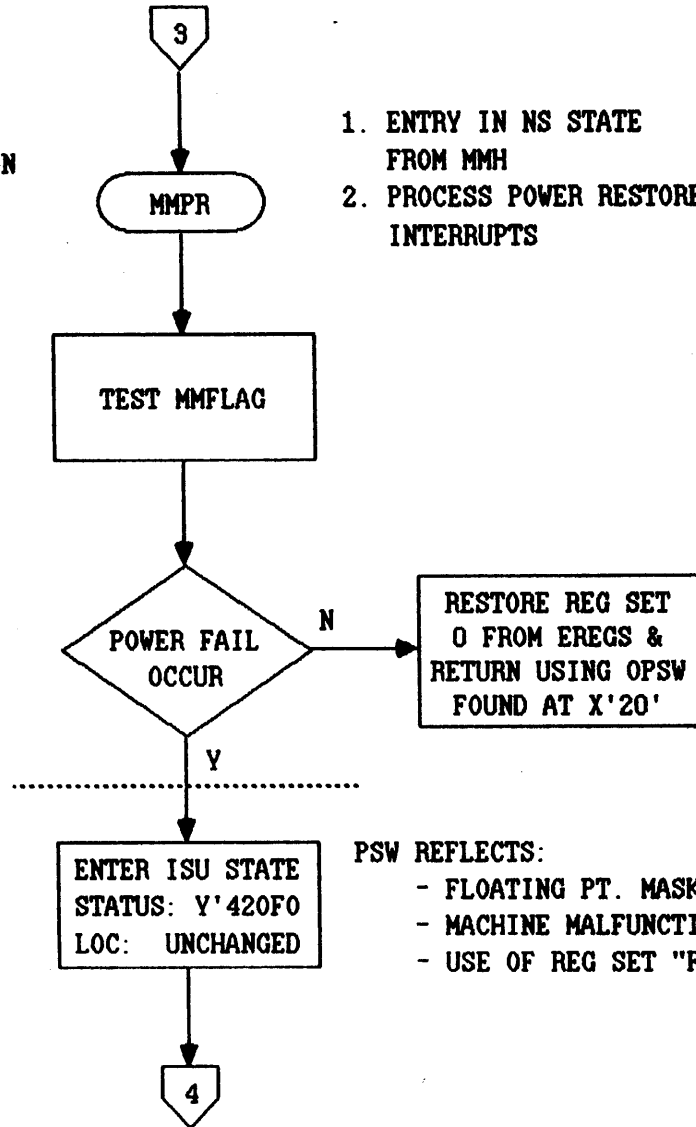
BITS 16 & 18 OF WAIT
 PSW SET WITH LOCATION
 INDICATING A(MMLWAIT)



PLACE PROCESSOR INTO
 WAIT STATE, WAITING
 FOR POWER RESTORATION.

PSW:
 STATUS Y'A000'
 LOC A(MMLWAIT)

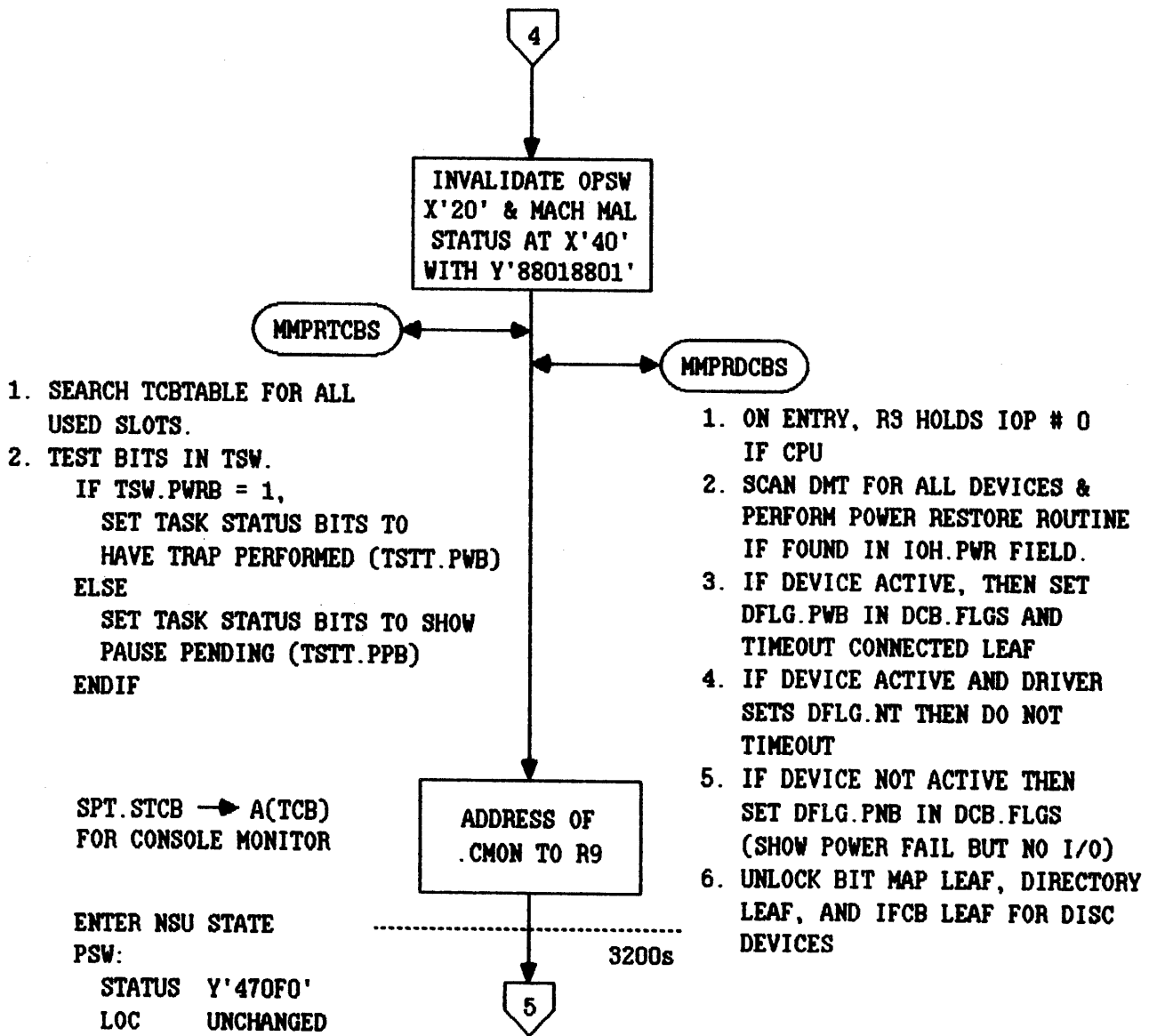
POWER RESTORATION



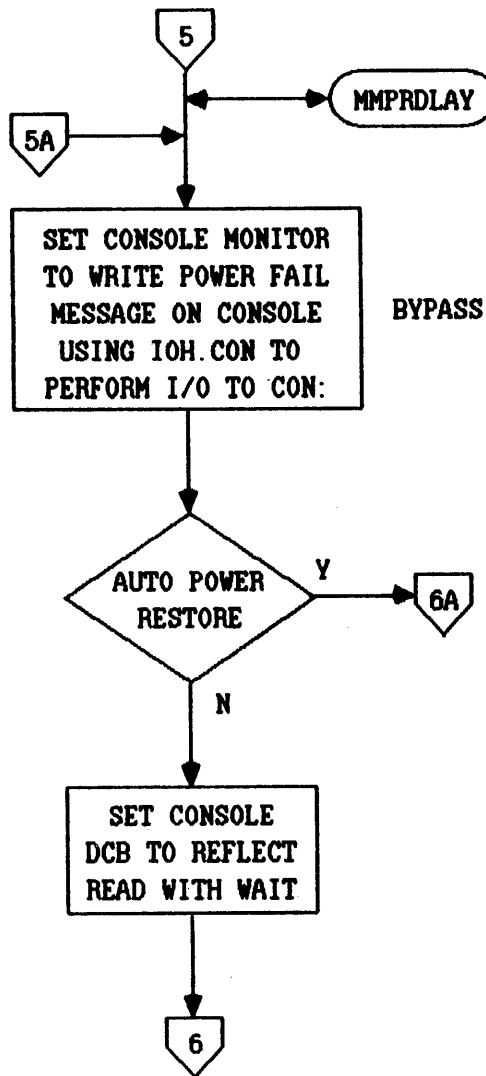
1. ENTRY IN NS STATE FROM MMH
2. PROCESS POWER RESTORE INTERRUPTS

PSW REFLECTS:

- FLOATING PT. MASK ENABLED
- MACHINE MALFUNCTION
- USE OF REG SET "F"

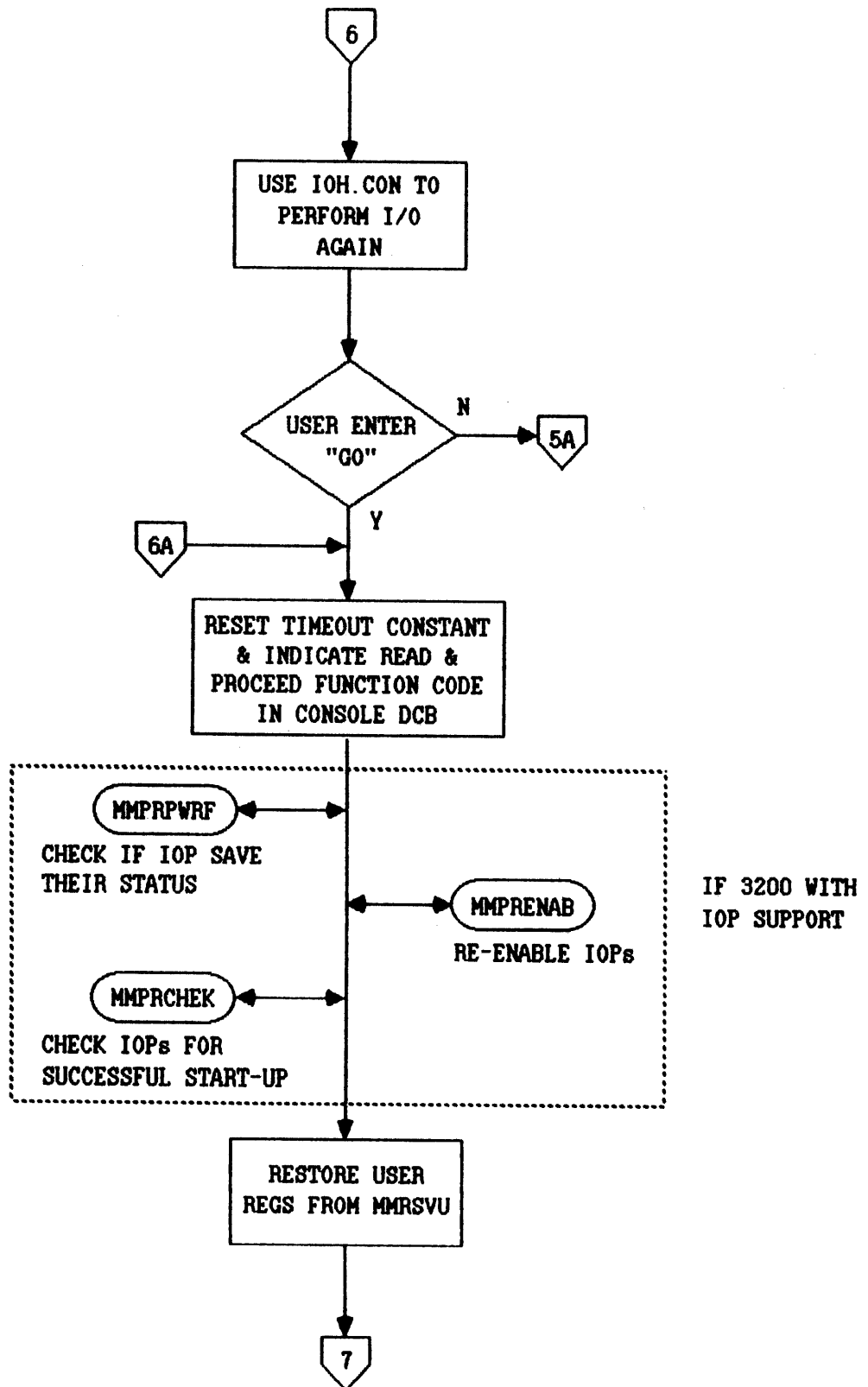


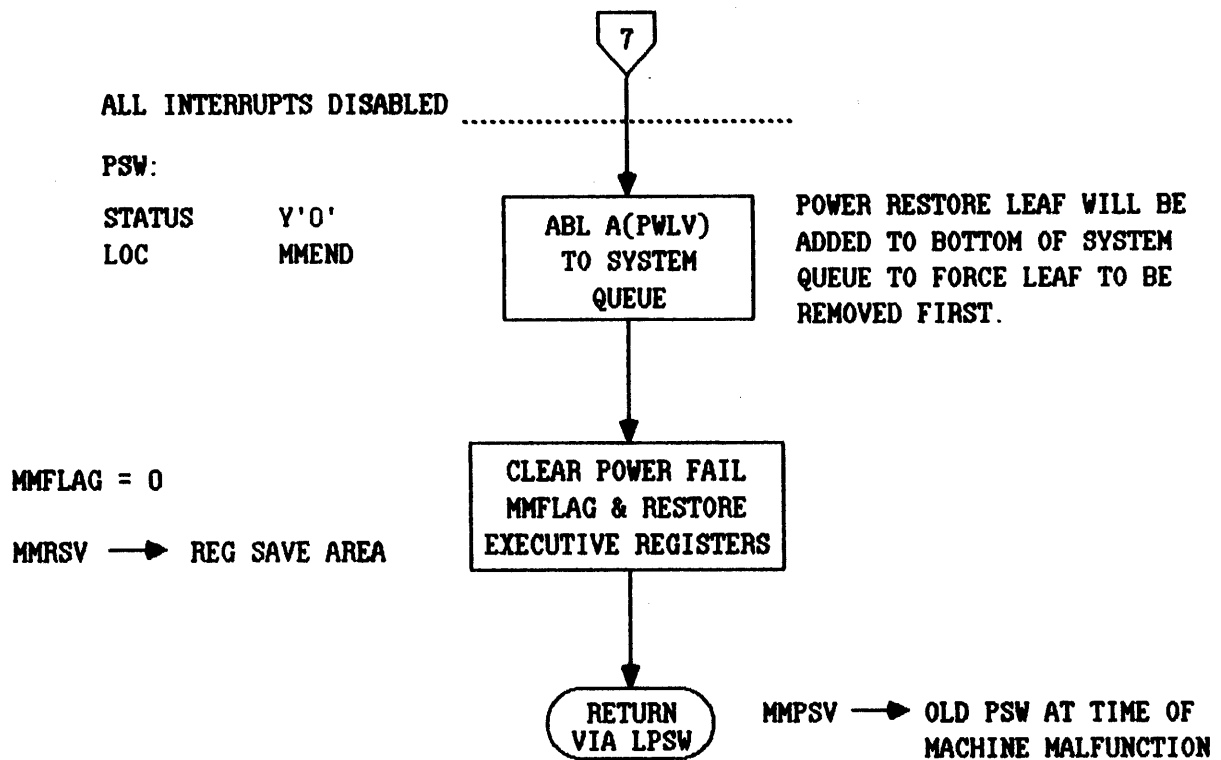
SVC1 PARAMETER
BLOCK WITHIN DCB.
SET UP SO CONSOLE
DEVICE CAN REFLECT
AUTO OR MANUAL
RESTART MESSAGE.



DELAY RESTORATION BY
VALUE AT IVT.DLAY TO
ALLOW BUS SWITCH TO
INITIALIZE ITSELF.

BYPASS SVC1 PROCESSING





SHARED MEMORY POWER
FAIL SERVICE
ENTRY IN NS STATE
FROM MMH

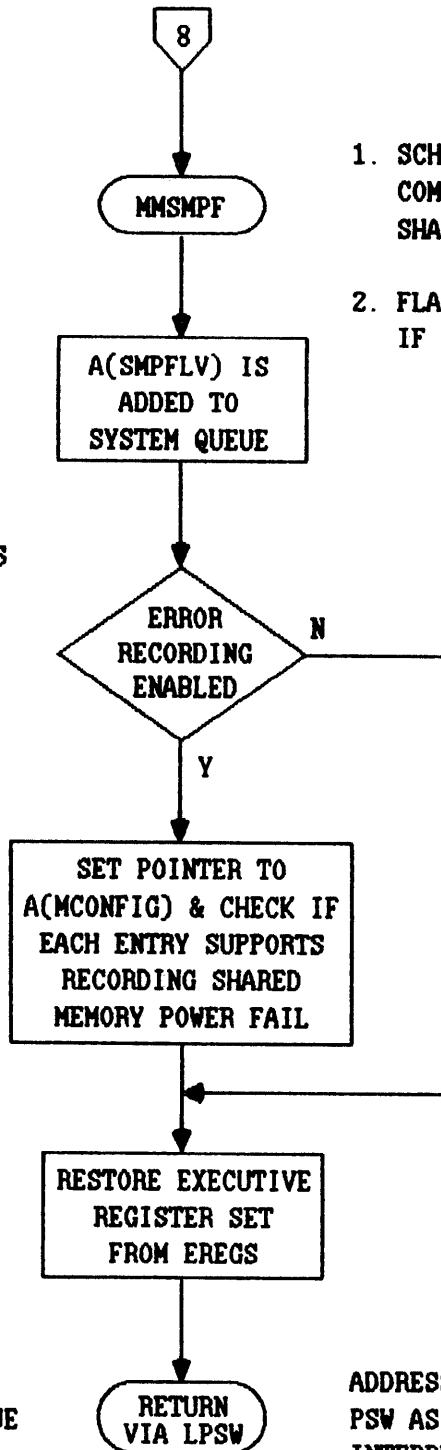
1. SCHEDULE AN ESR TO INFORM
COMMAND PROCESSOR OF
SHARED MEMORY POWER FAIL.
2. FLAG CONFIGURATION FILE
IF ERROR RECORDING ENABLED.

SOPT.MEB → IF SET INDICATES
ERROR RECORDING SUPPORT IN
SYSTEM. SYSTEM POINTER FIELD
SPT.SOPT CONTAINS SYSTEM
OPTION BITS.

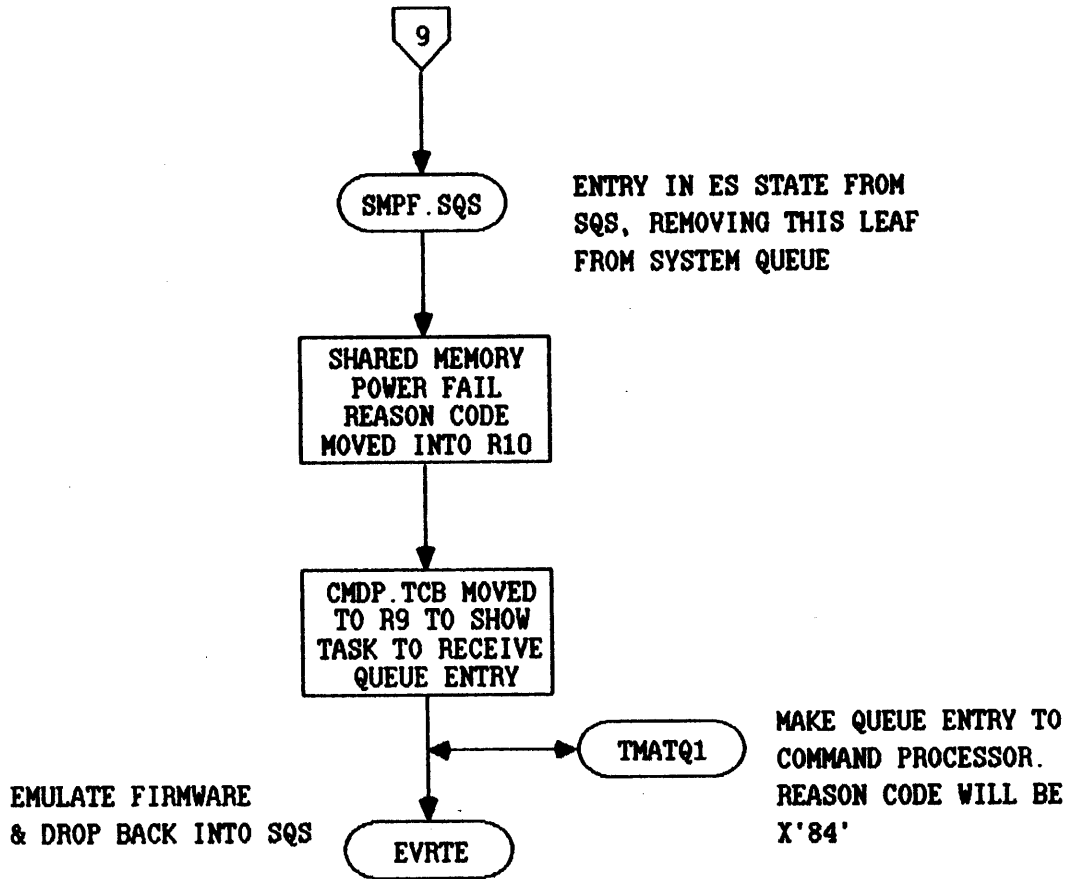
CHECK "MEM CONFIG" TAB
FOR SHARED MEMORY FLAG
(MCT.PFM)

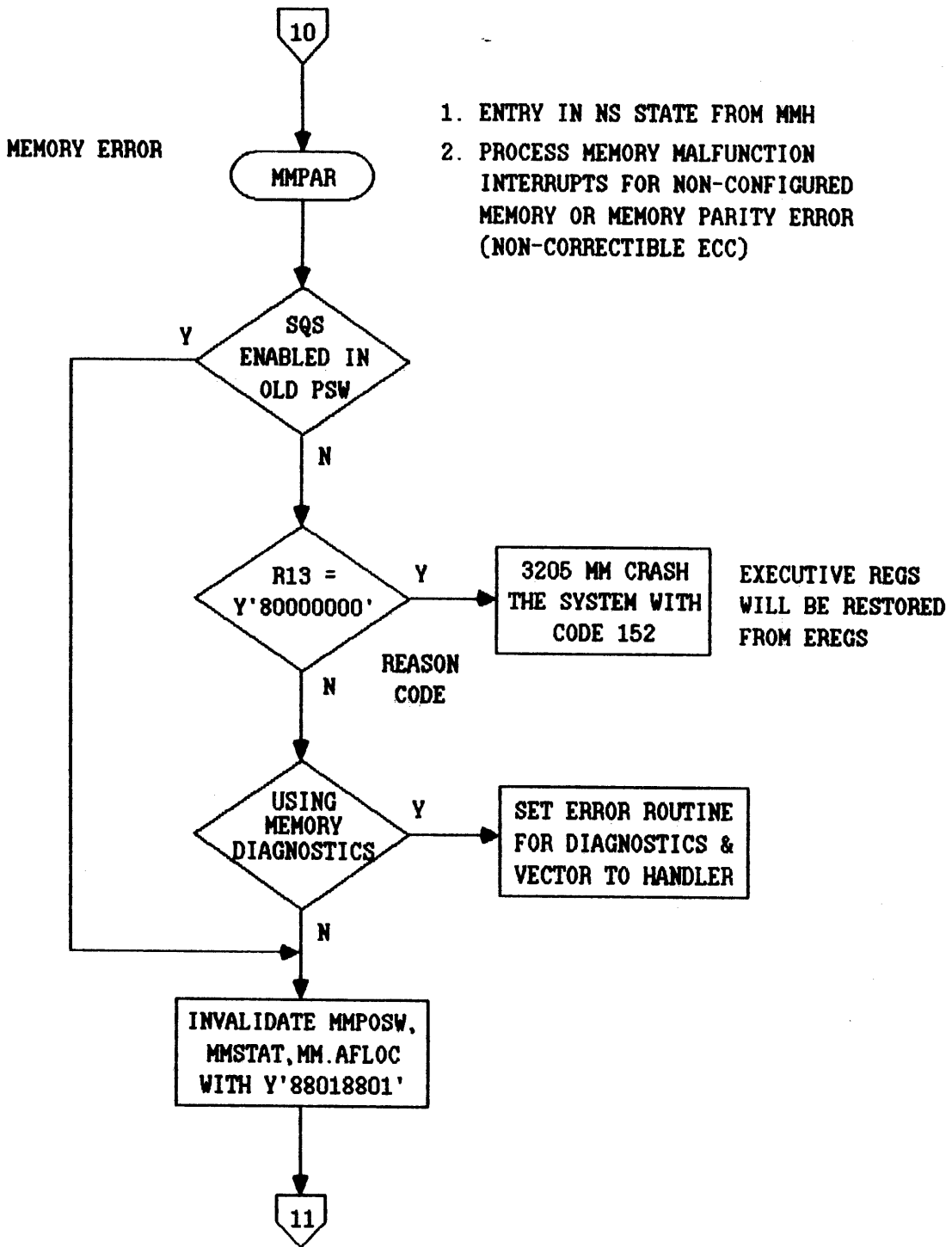
IF MCT.SMM & MCT.RECM = 1,
FLAG SHARED MEMORY
POWER FAIL (MCT.SHAR)
ELSE
NEXT ENTRY
ENDIF

TEST SYSTEM QUEUE



ADDRESS X'20' CONTAINS OLD
PSW AS SET BY FIRMWARE WHERE
INTERRUPT OCCURRED. THIS
LOCATION WILL BE RETURN VECTOR
A(MMOPSW) = X'20'





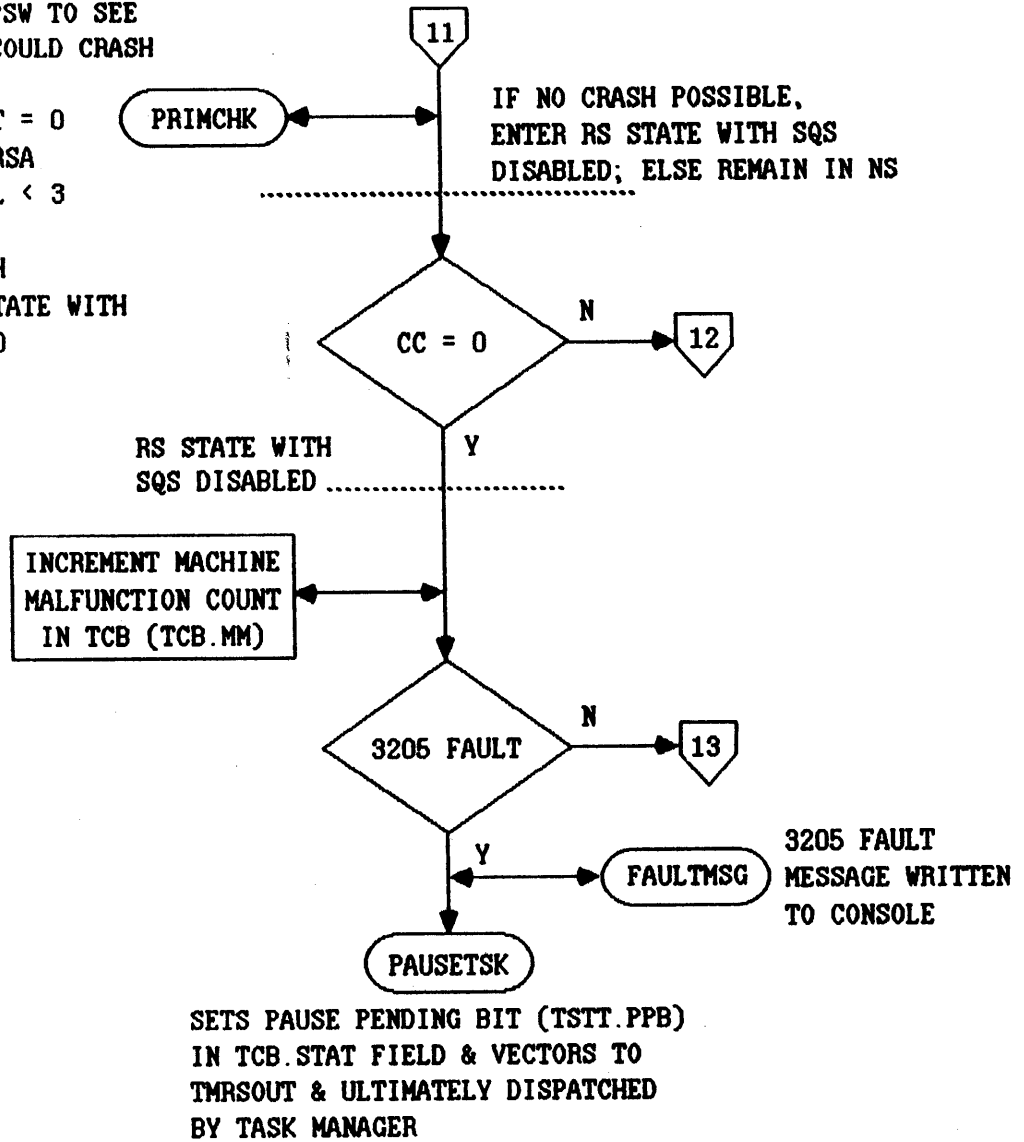
1. CHECK OLD PSW TO SEE
IF SYSTEM COULD CRASH

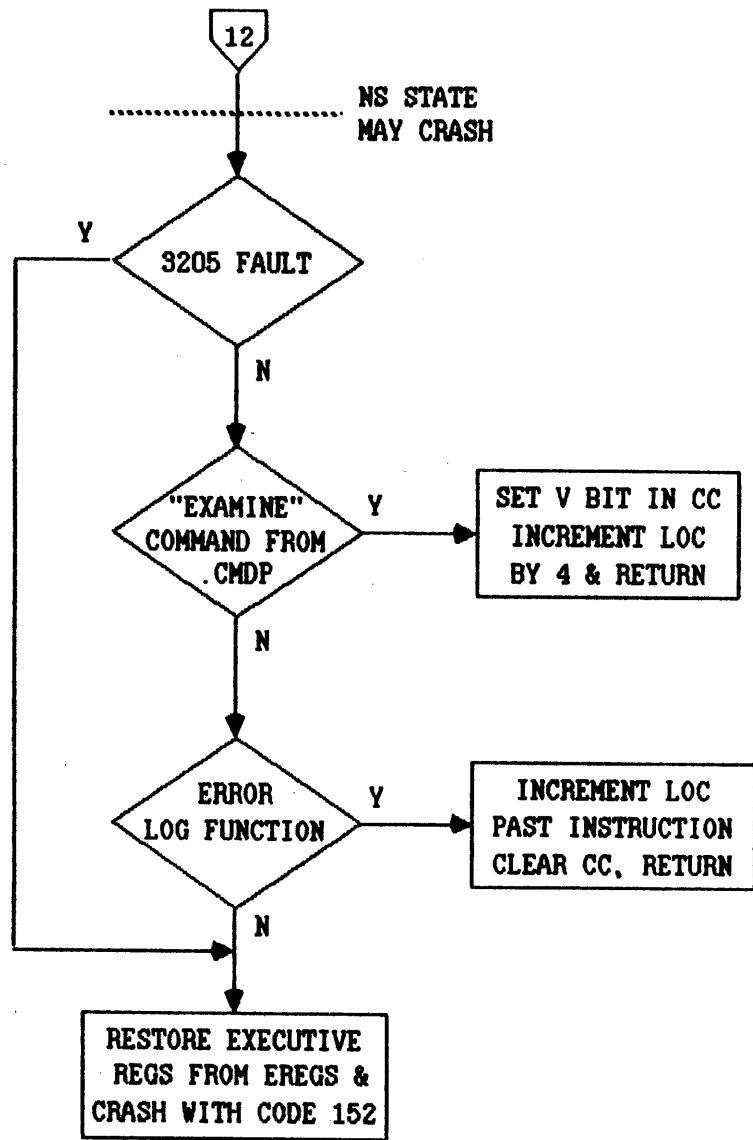
YES IF:

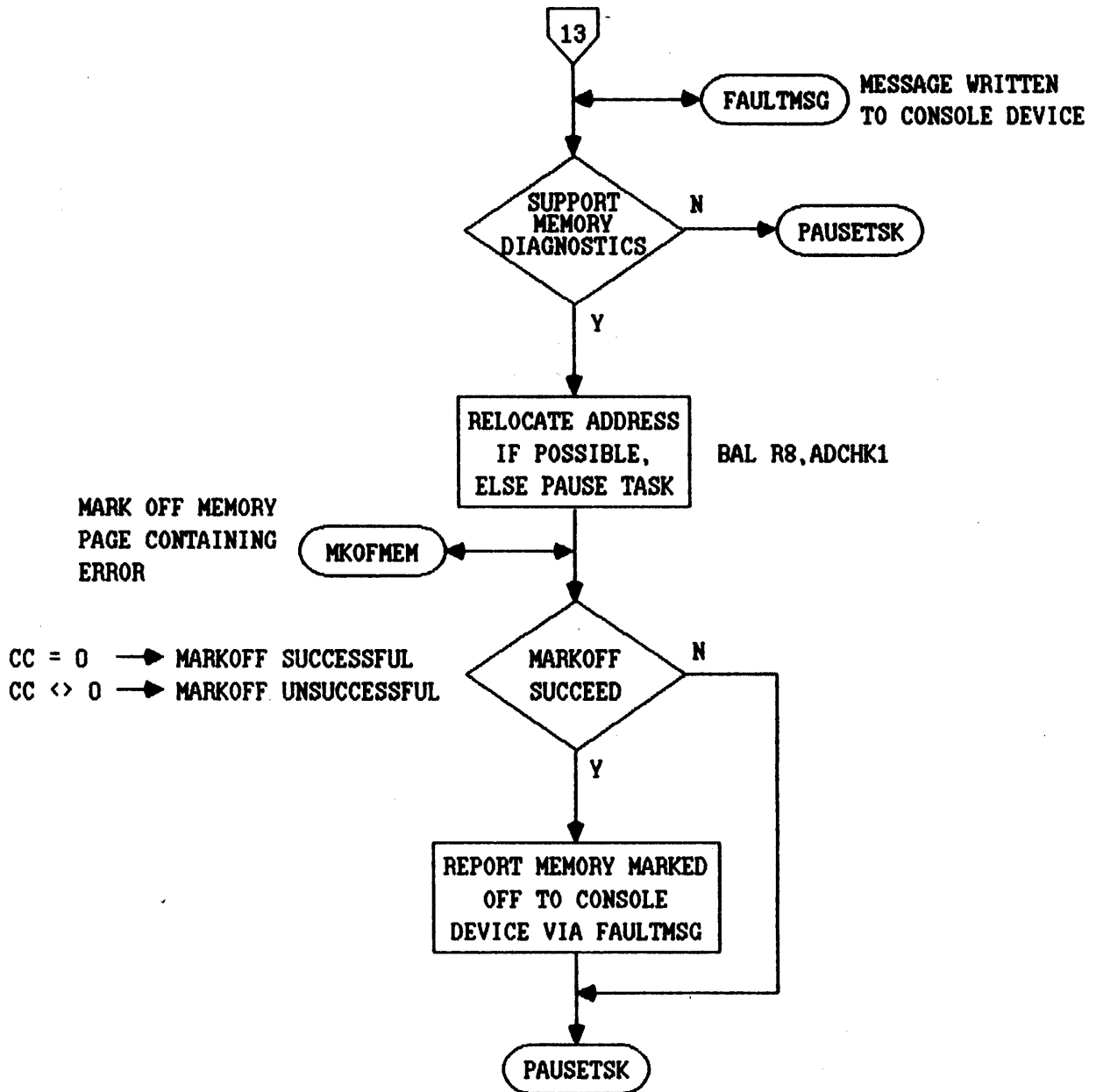
- SQS BIT = 0
- IN RS/RSA
- TCB.IDL < 3

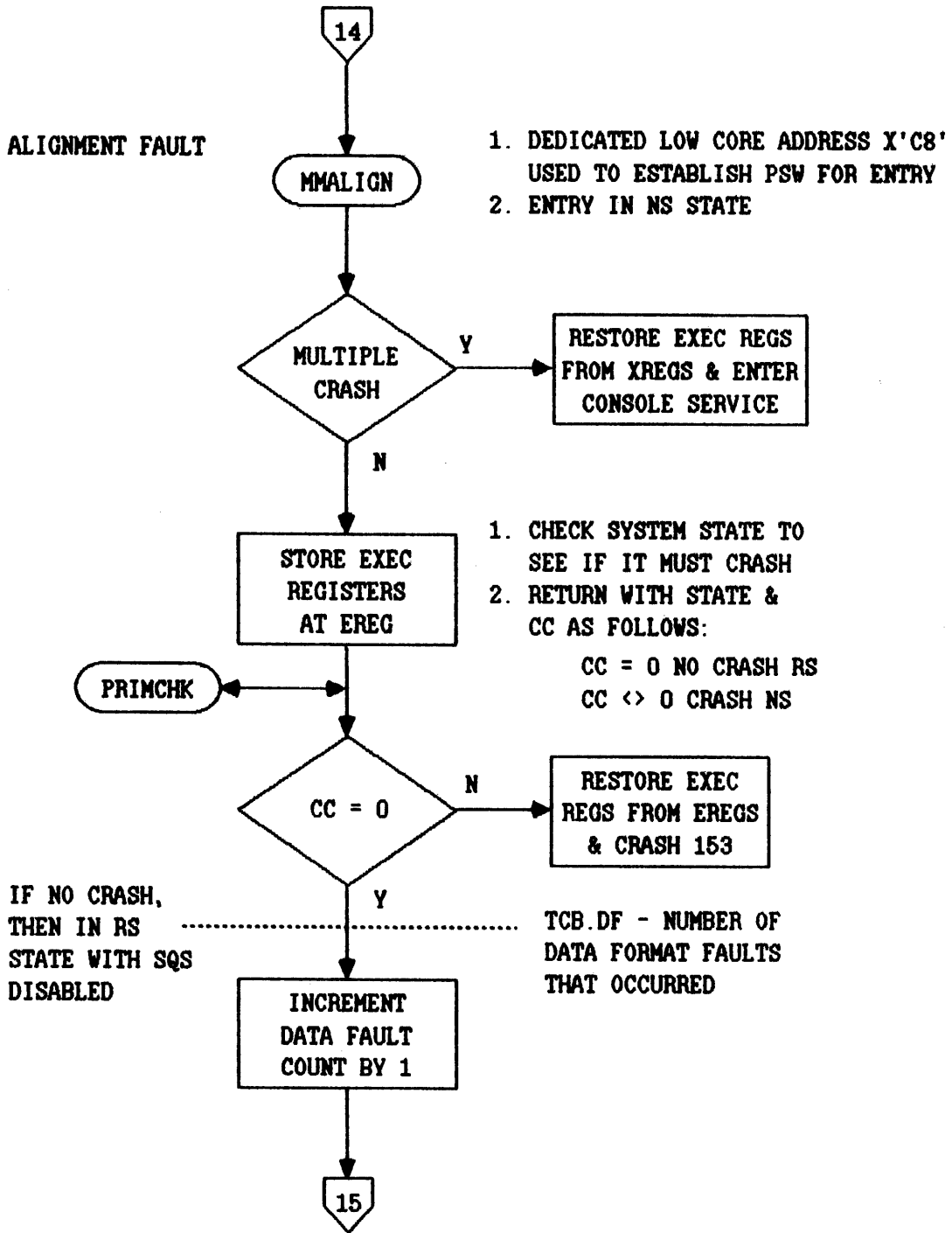
CC <> 0

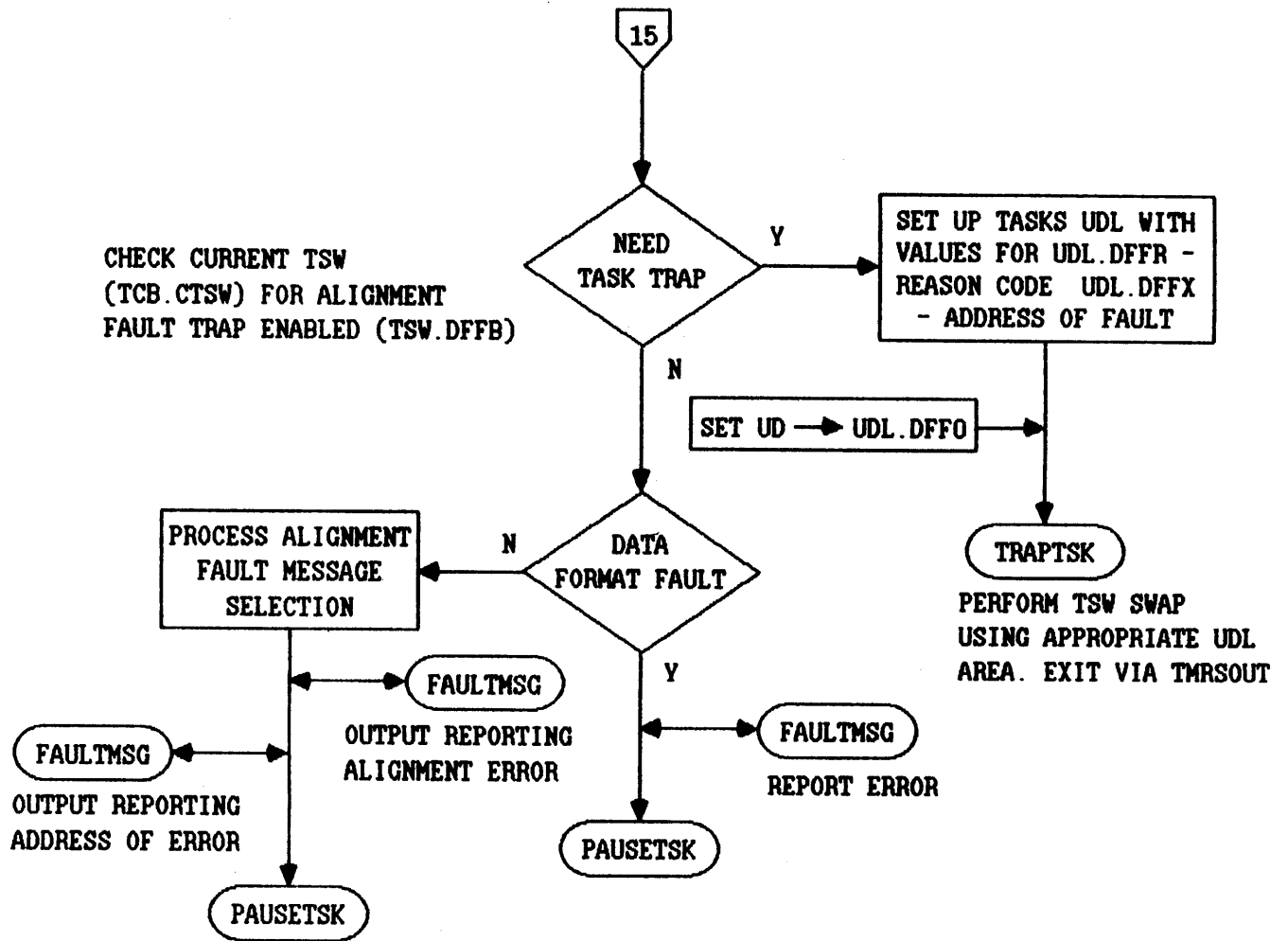
2. IF NO CRASH
ENTER RS STATE WITH
SQS BIT = 0
CC = 0



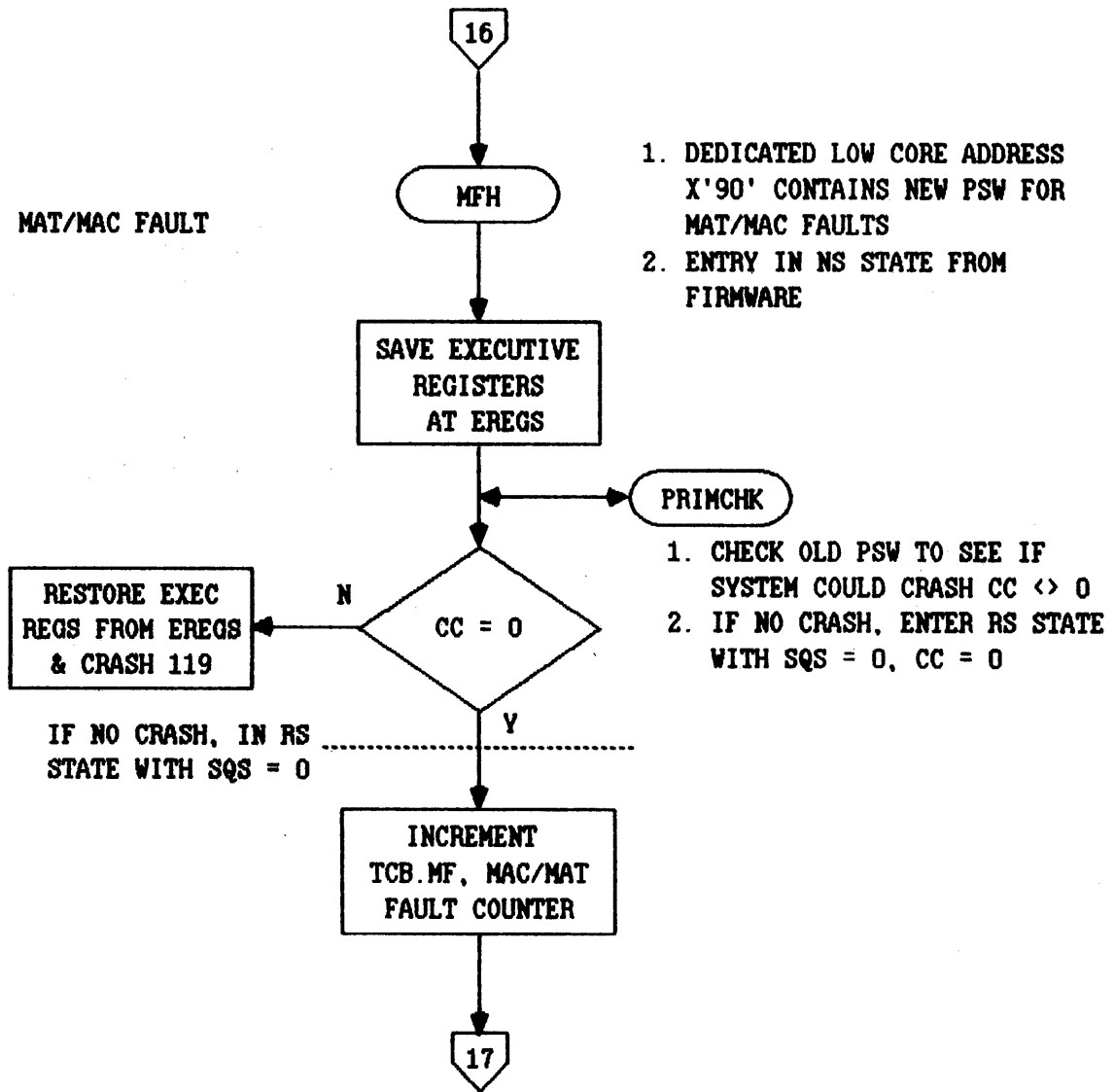


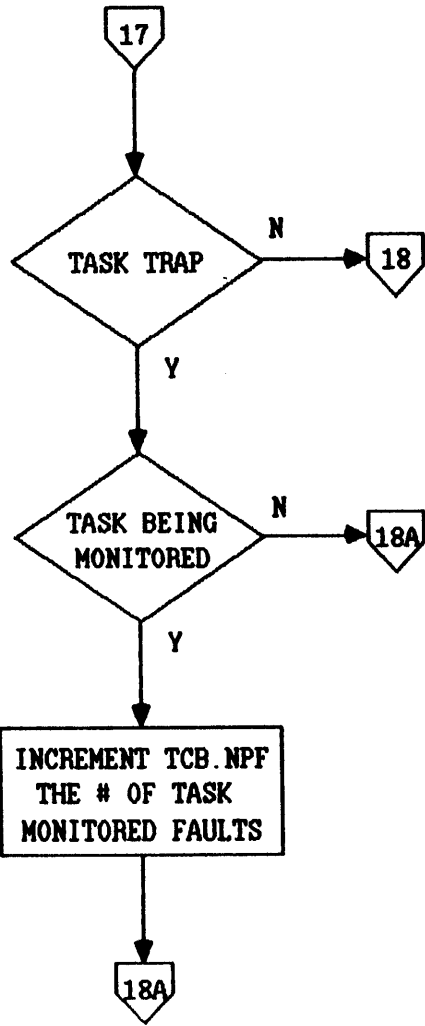


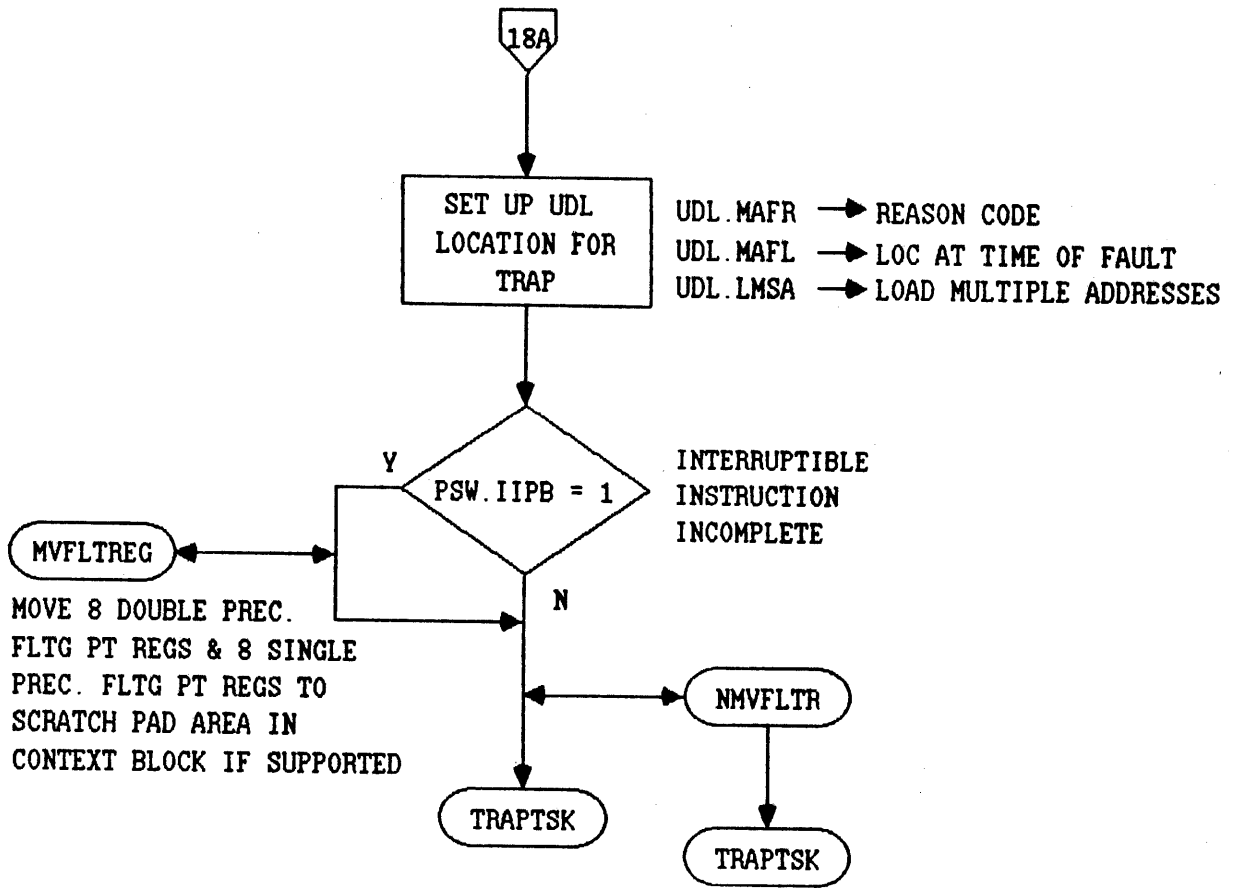


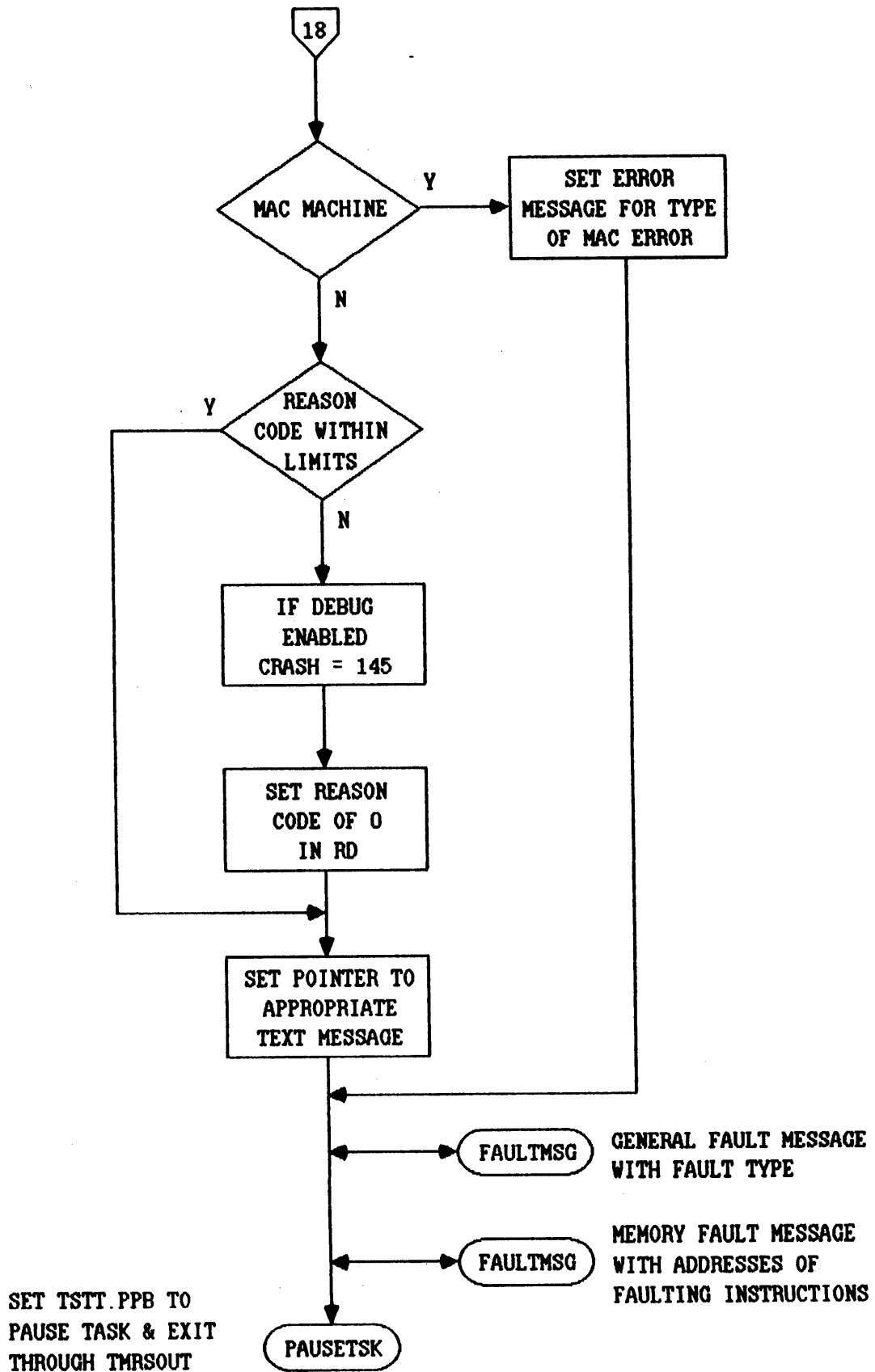


MAT/MAC FAULT







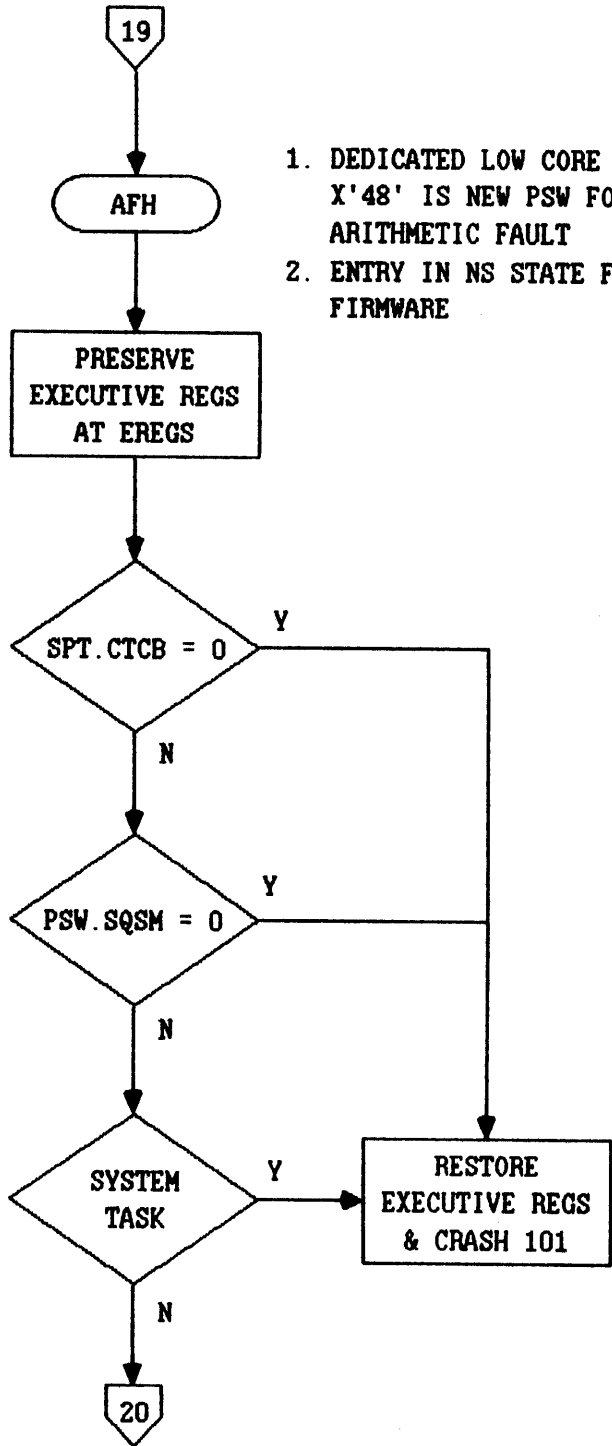


ARITHMETIC FAULT

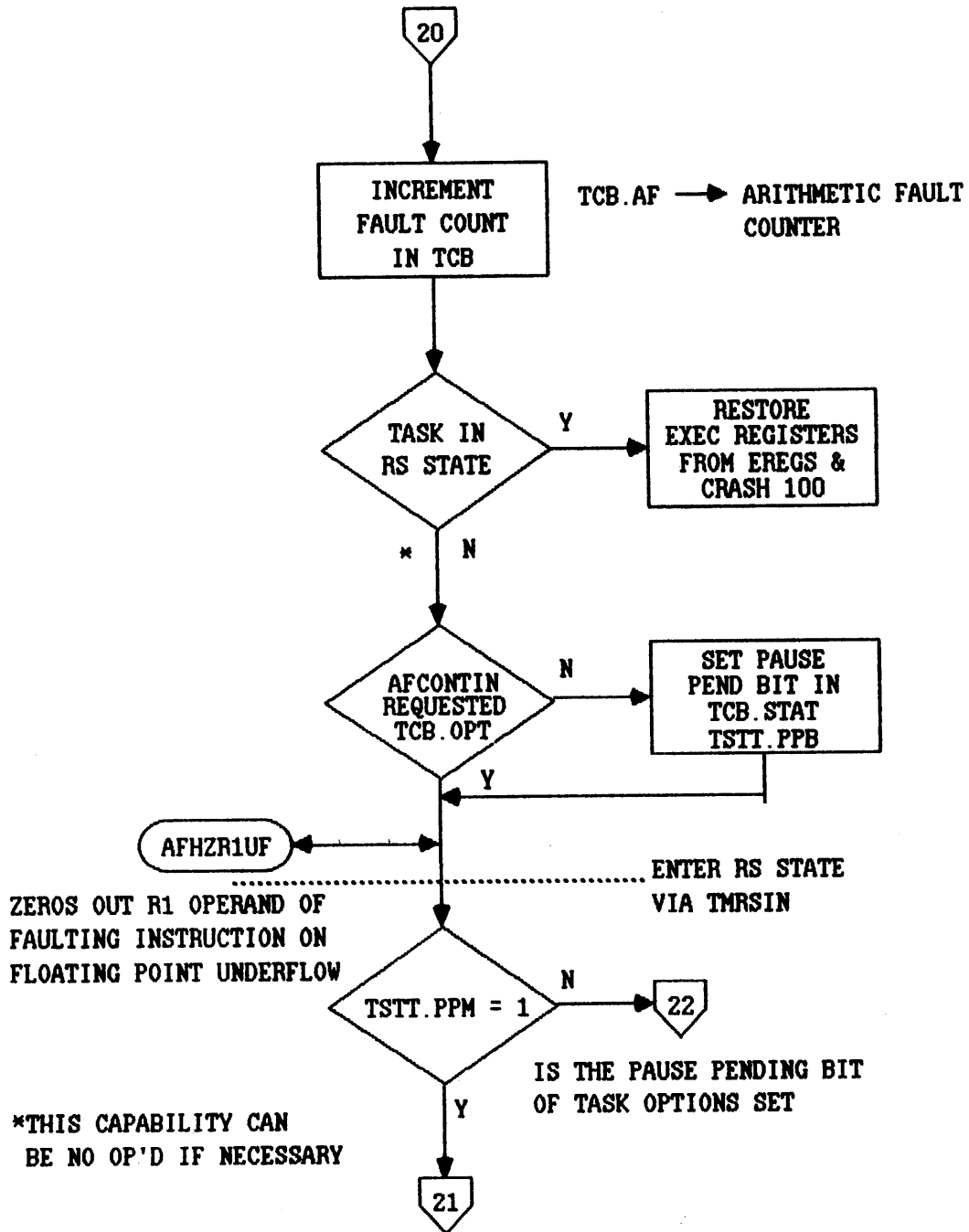
SPT.CTCB → A(TCB) OF
CURRENT TASK

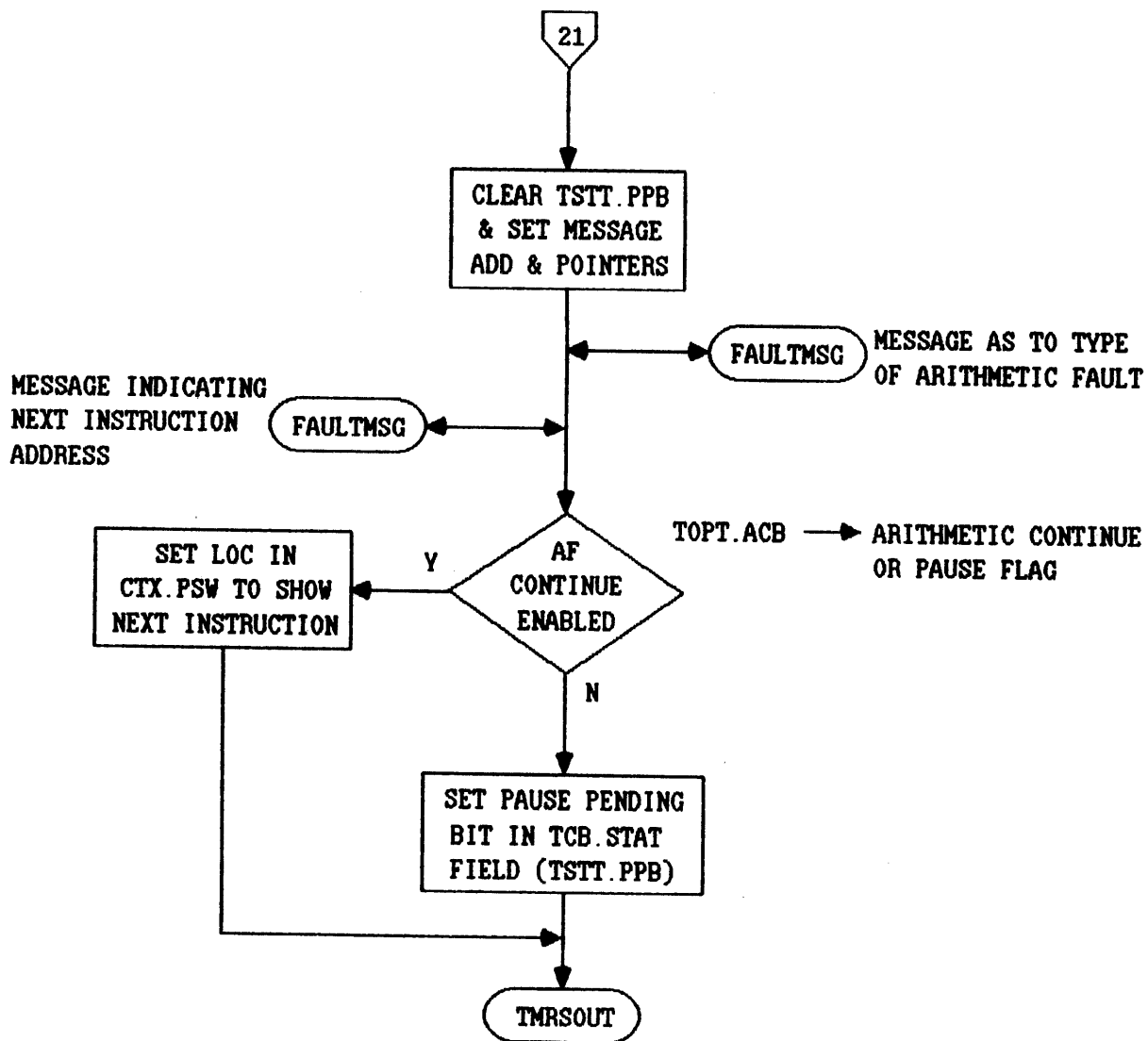
ENSURE SYSTEM WAS
IN TASK STATE

CHECK TCB.ID TO ENSURE
TASK WAS NOT A SYSTEM TASK
TCB.ID < 3 → SYSTEM TASK



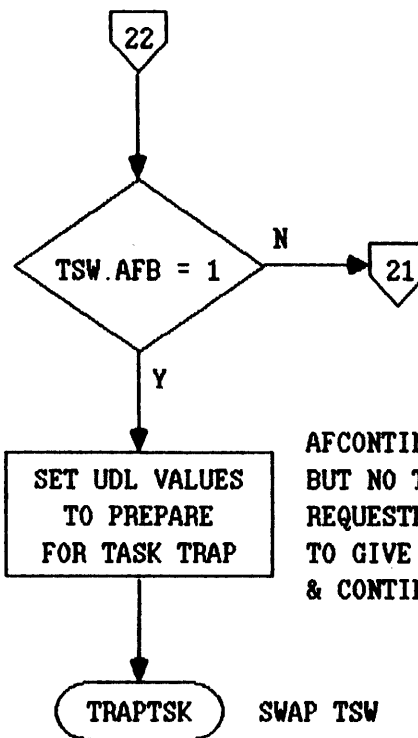
1. DEDICATED LOW CORE ADDRESS X'48' IS NEW PSW FOR ARITHMETIC FAULT
2. ENTRY IN NS STATE FROM FIRMWARE



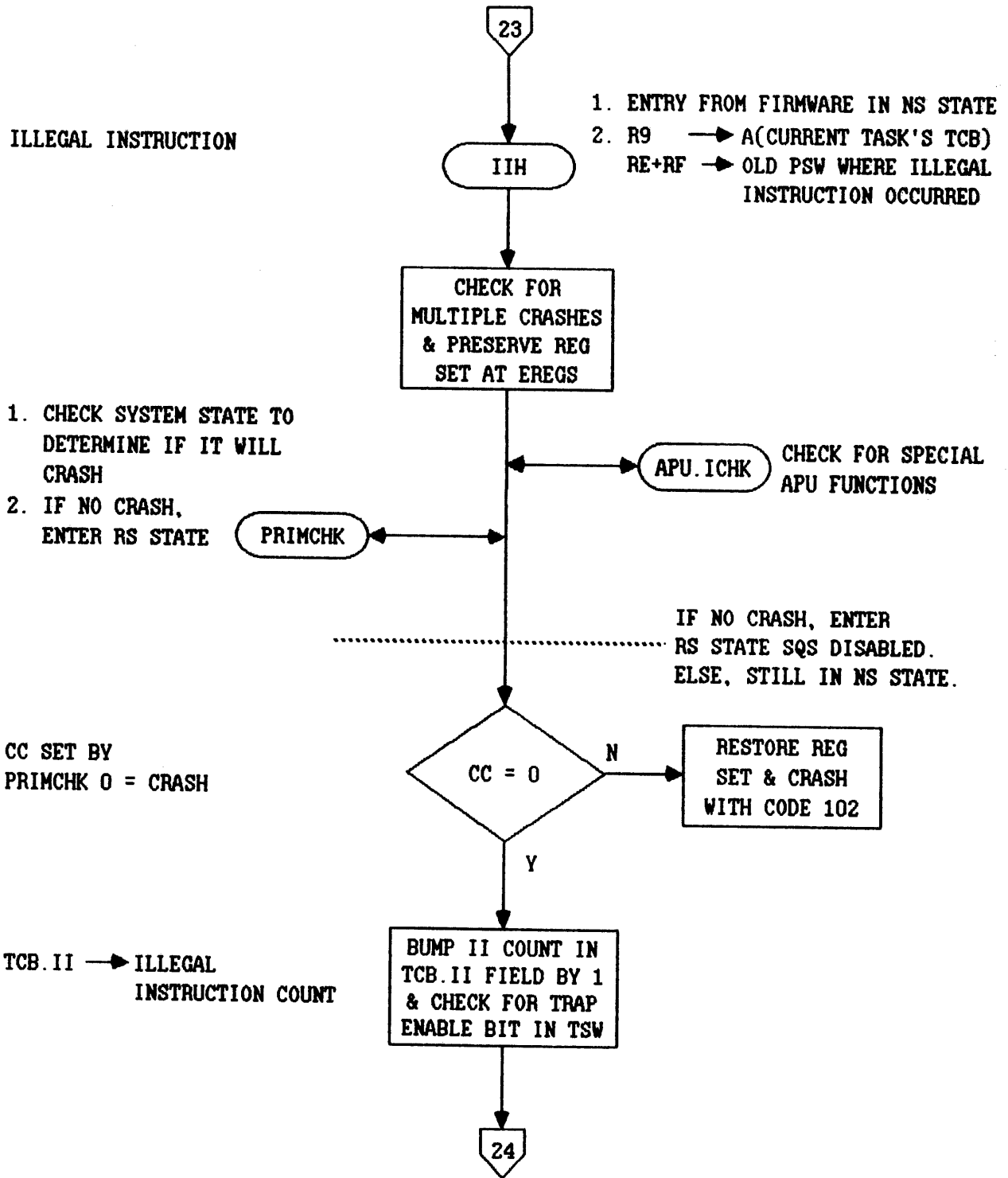


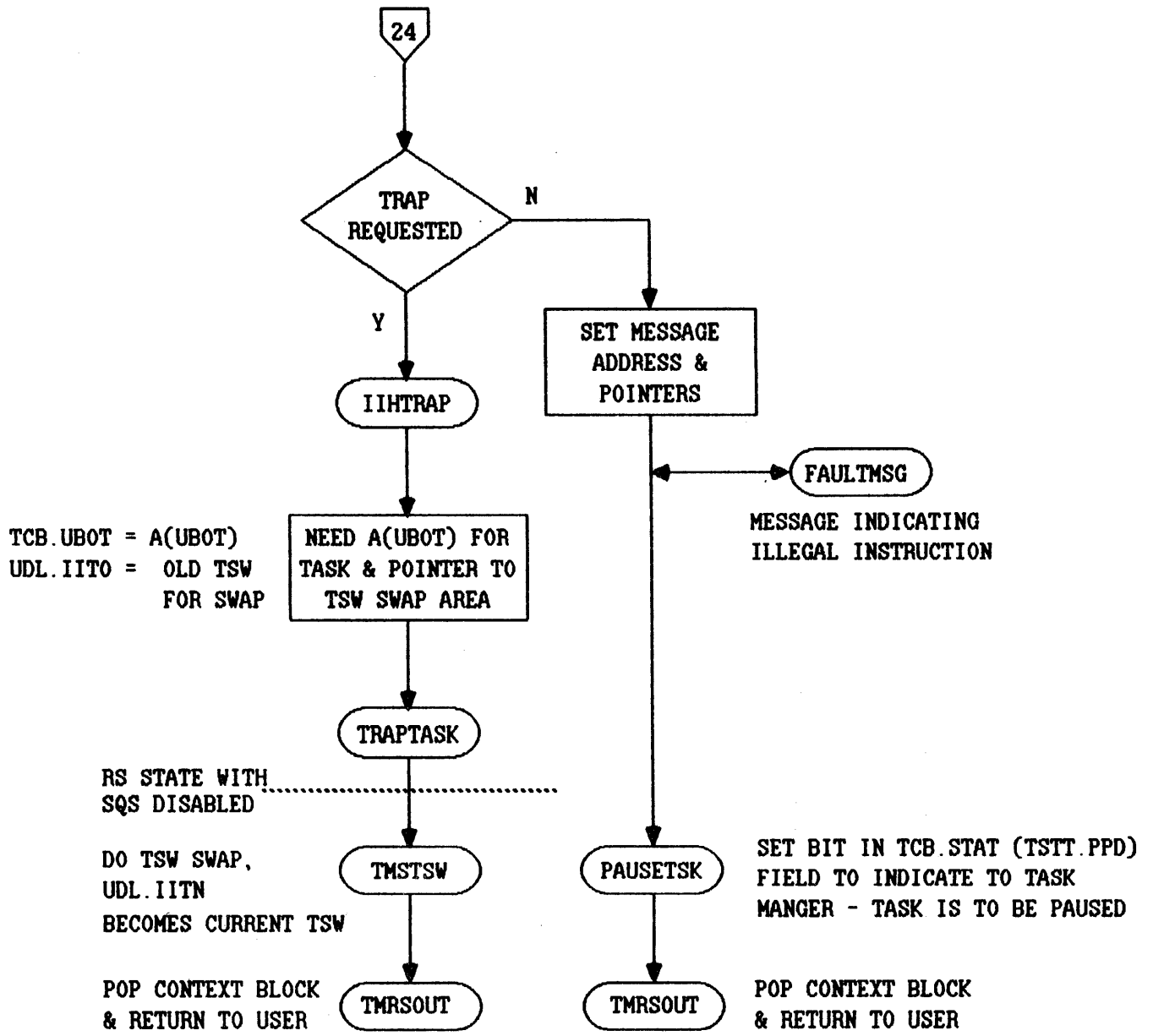
ARITHMETIC FAULT TRAP
OF TSW - IS IT SET

UDL.ARFN → ARITHMETIC FAULT
REASON CODE
UDL.ARFX → NEXT INSTRUCTION
ADDRESS
UDL.ARF0 → OLD TSW IN UDL



AFCONTINUE ENABLED,
BUT NO TRAP REQUEST
REQUESTED IN TSW. EXIT
TO GIVE ERROR MESSAGE
& CONTINUE EXECUTIVE





SYSTEM STATES

SUPERVISOR CALL STATES

DESCRIPTION:

The following table gives the system states used by the various SVC calls available to OS/32. All first level interrupt handlers (FLIH) will be entered in NS STATE from the firmware. This table gives the state on entry to the second level processor.

<u>SVC REQUEST</u>	<u>STATE ENTERED</u>
SVC-0	NS
SVC-1	NS
SVC-2	NS(INITIAL ENTRY FOR CODE NUMBER PROCESSING)

<u>CODE NUMBER</u>	<u>STATE</u>	<u>NUMBER OF REGISTERS</u>
0	NS	0
1	NS	0
2	NS	1
3	NS	0
4	NS	0
5	RS	1
6	RS	0
7	RS	0
8	RS	0
9	RS	0
10	RS	0
11	RS	0
14	NS	0
15	RS	1
16	RS	1
17	RS	2
18	RS	2
19	NS	0
20	NS	0
21	NS	0
23	NS	0
24	NS	0
25	NS	1
26	RS	2
27	RS	2
28	NS	0
29	RS	2
30	RS	1

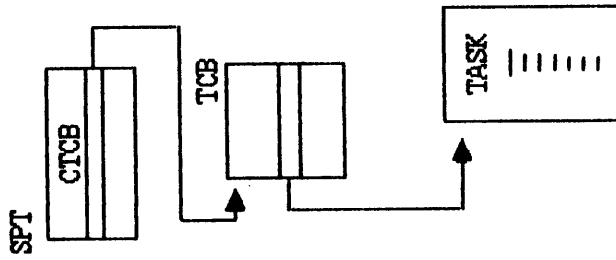
SUPERVISOR CALL STATES(continued)

<u>SVC REQUEST</u>	<u>STATE ENTERED</u>
SVC-3	NS
SVC-4	NS will enter RS
SVC-5	RS
SVC-6	RS
SVC-7	RS
SVC-9	NS
SVC-10	RSA
SVC-13	NS
SVC-14	NS
SVC-15	NS

IDLE

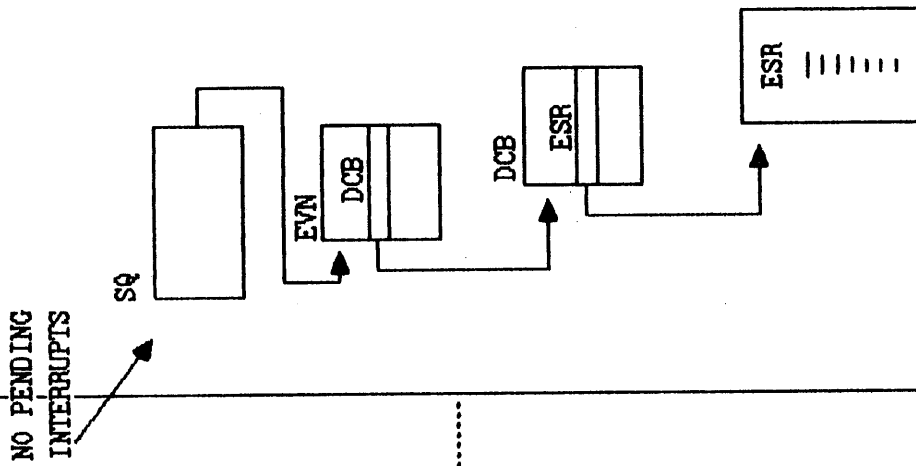
IDLE PSW

TASK MANAGER

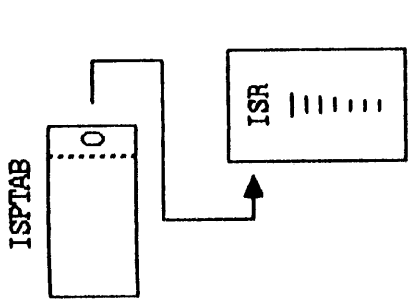


NO READY TASK

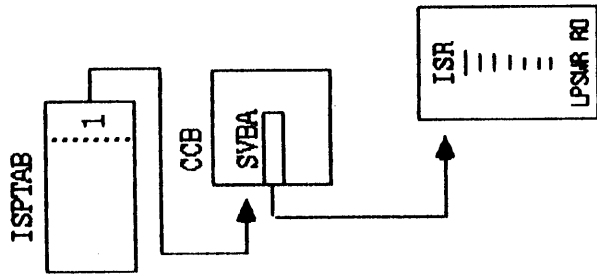
SYSTEM QUEUE SERVICE



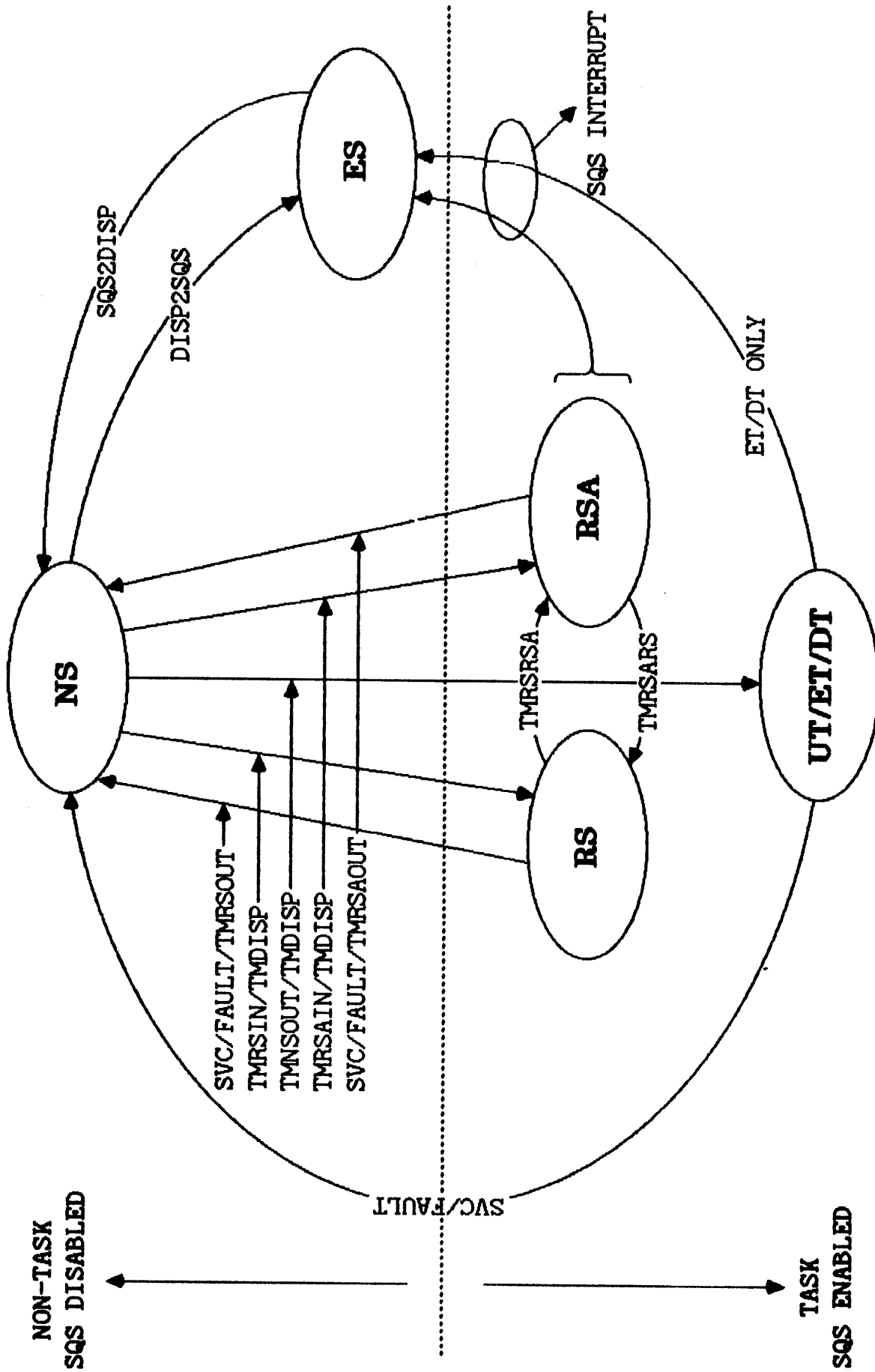
INTERRUPT SERVICE



INTERRUPT DRIVEN



AUTO DRIVER CHANNEL



UT/DT/ET STATE

- TASK-LEVEL STATES: - USE "USER" CONTEXT BLOCK POINTED TO BY TCB.UCTX
 - SQS ENABLED
 - SVCs ARE LEGAL

- UT STATE PSW: - RELOCATION/PROTECTION ENABLED
 - PRIVILEGED INSTRUCTIONS ILLEGAL

- DT STATE PSW: - RELOCATION/PROTECTION ENABLED
 - PRIVILEGED INSTRUCTIONS LEGAL

- ET STATE PSW: - RELOCATION/PROTECTION DISABLED
 - PRIVILEGED INSTRUCTIONS LEGAL

- ENTERED VIA: - TASK DISPATCHER
 - TMNSOUT

- EXIT VIA: - INTERRUPT (SVC OR FAULT)

NS STATE

- USED BY:
 - FIRST LEVEL INTERRUPT HANDLERS
(FAULTS AND SVCs)
 - SOME SHORT SVC SECOND LEVEL INTERRUPT
HANDLERS
 - TASK DISPATCHER

- NON-TASK STATE:
 - SQS INTERRUPTS DISABLED
 - NO CONTEXT BLOCK
 - SVCs ARE ILLEGAL

- REGISTER SET 0
 - RESTRICTED TO REGISTERS 8-F

- ENTERED ONLY VIA:
 - INTERNAL INTERRUPT
(FAULT OR SVC)
 - CALL TO DISPATCHER
(LPSW)

- EXIT VIA:
 - TMNSOUT: RETURN TO CURRENT TASK
 - TMDISP: DISPATCHER
 - TMRS(A)IN: ENTER RS(A) STATE

RS STATE

- USED BY: - SOME SVC SECOND LEVEL INTERRUPT HANDLERS
 - PARTS OF FILE MANAGER

- TASK-STATE: - SQS INTERRUPTS (USUALLY) ENABLED
 - DEDICATED CONTEXT BLOCK AT TCB. RS
 - SVCs (WHICH DO NOT USE RS STATE ARE
 LEGAL)

- REGISTER SET 6 IF AVAILABLE, ELSE SET F

- TSTT. RSM SET IN TCB. STAT

- ENTERED VIA: - TMRSIN (FROM NS STATE)
 - TMRSARS (FROM RSA STATE)
 - TMDISP (TASK DISPATCHER)

- EXIT VIA: - TMRSOUT (TO DISPATCHER)
 - TMRSRSA (TO RSA STATE)
 - INTERRUPT (SVC OR FAULT)

RSA STATE

- **USED BY:**
 - SOME SVC SECOND LEVEL INTERRUPT HANDLERS (E. G. , SVC 10 - OVERLAY LOAD)
 - ROLL-IN

- **TASK-STATE:**
 - IDENTICAL TO RS EXCEPT:
 - * SEPARATE CONTEXT BLOCKS USED
 - * SVCs (INCLUDING THOSE WHICH USE RS STATE) ARE LEGAL

- **REGISTER SET 6 IF AVAILABLE, ELSE SET F**

- **TSTT. ASM AND TSTT. RSM SET IN TCB. STAT**

- **TCB. RSAC COUNTS RSA NESTING LEVEL**

- **ENTERED VIA:**
 - TMRSAIN (FROM NS STATE)
 - TMRSRSA (FROM RS STATE)
 - TASK DISPATCHER

- **EXITS VIA:**
 - TMRSAOUT (TO DISPATCHER)
 - TMRSARS (TO RS STATE)
 - INTERRUPT (SVC OR FAULT)

ES STATE

- USED BY:
 - SYSTEM QUEUE SERVICE
 - DEVICE DRIVER EVENTS
(INITIATION AND TERMINATION)
 - SYSTEM EVENTS
(CLOCKS, POWER RESTORE)

- NON-TASK STATE:
 - SQS INTERRUPTS DISABLED
 - NO CONTEXT BLOCK
 - SVCs ARE ILLEGAL

- REGISTER SET 5 IF AVAILABLE, ELSE SET F

- ENTERED VIA:
 - SYSTEM QUEUE "INTERRUPT"
 - LPSW FROM TASK DISPATCHER

- EXITS VIA:
 - LPSW TO TASK DISPATCHER

SVC0

LAB EXERCISES

APPENDIX

TABLE OF CONTENTS

APPENDIX

System Options Bits	A-1
Task Options Bits	A-4
Task Wait Bits	A-7
Task Status Bits	A-9
Device Control Block Flag Bits	A-12
Input/Output Block Flag Bits	A-14
File Control Block Flag Bits	A-16

SYSTEM OPTIONS BITS

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
SOPT.FPB EQU	0	SINGLE PRECISION FLOATING POINT
SOPT.FPM EQU	Y'80000000'	
SOPT.USB EQU	1	DATE IS MMDDYY IF OFF, DDMYY IF ON
SOPT.USM EQU	Y'40000000'	
SOPT.DIB EQU	2	DISPLAY TIME ON PANEL IF ON
SOPT.DIM EQU	Y'20000000'	
SOPT.DFB EQU	3	DOUBLE PRECISION FLOATING POINT
SOPT.DFM EQU	Y'10000000'	
SOPT.WCB EQU	4	
SOPT.WCM EQU	Y'08000000'	WCS SUPPORT
SOPT.ALB EQU	5	
SOPT.ALM EQU	Y'04000000'	ALIGNMENT ERROR CHECK
SOPT.DAB EQU	6	
SOPT.DAM EQU	Y'02000000'	SYSTEM OPTION-DIRECT ACCESS
SOPT.ITB EQU	7	
SOPT.ITM EQU	Y'01000000'	SYSTEM OPTION-ITAM
SOPT.SPB EQU	8	
SOPT.SPM EQU	Y'00800000'	SYSTEM OPTION-SPOOL SUPPORT
SOPT.RLB EQU	9	
SOPT.RLM EQU	Y'00400000'	SYSTEM OPTION-ROLL SUPPORT
SOPT.TMB EQU	10	
SOPT.TMM EQU	Y'00200000'	SYSTEM OPTION-TEMPORARY FILES SUPPORT
SOPT.MRB EQU	11	
SOPT.MRM EQU	Y'00100000'	MULTIPLE REGISTER SETS

SYSTEM OPTIONS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
SOPT.URB EQU	12	
SOPT.URM EQU	Y'00080000'	UNIVERSAL REPORTING ON
SOPT.ERB EQU	13	GENERAL ERROR RECORDING OPTION
SOPT.ERM EQU	Y'00040000'	
SOPT.MEB EQU	14	MEMORY ERROR RECORDING OPTION
SOPT.MEM EQU	Y'00020000'	
SOPT.LRB EQU	16	
SOPT.LRM EQU	Y'00008000'	SYSTEM OPTION LOAD REAL ADDRESS
SOPT.MCB EQU	17	
SOPT.MCM EQU	Y'00004000'	SYSTEM OPTION- MEMORY DIAGNOSTIC
SOPT.32B EQU	18	
SOPT.32M EQU	Y'00002000'	IF SET, 3200 SERIES, ELSE 8/32-7/32
SOPT.ATB EQU	19	
SOPT.ATM EQU	Y'00001000'	SYSTEM HAS MAT HARDWARE
SOPT.FSB EQU	20	SINGLE PREC. FL. PT. SFTWR PRES
SOPT.FSM EQU	Y'00000800'	
SOPT.DSB EQU	21	DOUBLE PREC. FL. PT. SFTWR PRES
SOPT.DSM EQU	Y'00000400'	
SOPT.CSB EQU	22	COMMERCIAL SPOOLER SUPPORT
SOPT.CSM EQU	Y'00000200'	MUTUALLY EXCLUSIVE WITH SPOOL
SOPT.CCB EQU	23	3200-COMPATIBLE CPU
SOPT.CCM EQU	Y'00000100'	
SOPT.IPB EQU	24	I/O PROCESSOR SUPPORT (IOP)
SOPT.IPM EQU	Y'00000080'	3250XP, 3200MPS ONLY

SYSTEM OPTIONS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
SOPT.MNB EQU	29	
SOPT.MNM EQU	Y'00000004'	SYSTEM MONITOR SUPPORT
SOPT.STB EQU	30	
SOPT.STM EQU	Y'00000002'	STREAMER TAPE ONLY FOR DUMP
SOPT.DBB EQU	31	
SOPT.DBM EQU	Y'00000001'	SYSTEM DEBUG MODE

TASK OPTIONS BITS

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
TOPT.DTB EQU	0	
TOPT.DTM EQU	Y'80000000'	D-TASK
TOPT.APB EQU	1	
TOPT.APM EQU	Y'40000000'	APU ONLY
TOPT.MPB EQU	2	
TOPT.MPM EQU	Y'20000000'	SVC13 MAPPING RIGHTS
TOPT.CLB EQU	3	
TOPT.CLM EQU	Y'10000000'	SVC13 CONTROL RIGHTS
TOPT.SCB EQU	4	
TOPT.SCM EQU	Y'08000000'	ENABLE DYNAMIC SCHEDULING
TOPT.PRB EQU	5	
TOPT.PRM EQU	Y'04000000'	PROMPT SETTING
TOPT.X1B EQU	6	
TOPT.X1M EQU	Y'02000000'	USES EXTENDED SVC 1 PBLK
TOPT.VFB EQU	7	
TOPT.VFM EQU	Y'01000000'	USES VERTICAL FORMS CONTROL
TOPT.DBB EQU	8	
TOPT.DBM EQU	Y'00800000'	DEBUG/32 DTABLES
TOPT.PSB EQU	9	
TOPT.PSM EQU	Y'00400000'	PARTIAL REGISTER SAVE FOR EVENTS
TOPT.ESB EQU	10	
TOPT.ESM EQU	Y'00200000'	FULL OR PART SAVE FOR EVENTS
TOPT.SYB EQU	11	
TOPT.SYM EQU	Y'00100000'	IN SYSTEM GROUP

TASK OPTIONS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER</u> <u>OR MASK</u>	<u>DESCRIPTION</u>
TOPT.CIB EQU	12	
TOPT.CIM EQU	Y'00080000'	CONSOLE I/O INTERCEPT ENABLE (MTM)
TOPT.URB EQU	13	
TOPT.URM EQU	Y'00040000'	ALLOW UNIVERSAL STATUS REPORTS
TOPT.LEB EQU	14	
TOPT.LEM EQU	Y'00020000'	PREVENT E-TASK/D-TASK LOAD
TOPT.PGB EQU	15	
TOPT.PGM EQU	Y'00010000'	PURGE QUEUED I/O'S ON ERROR
TOPT.ETB EQU	16	
TOPT.ETM EQU	Y'00008000'	E-TASK
TOPT.ACB EQU	17	
TOPT.ACM EQU	Y'00004000'	ARITH FAULT CONTINUE
TOPT.FPB EQU	18	
TOPT.FPM EQU	Y'00002000'	USING SINGLE F/P
TOPT.MRB EQU	19	
TOPT.MRM EQU	Y'00001000'	MEMORY RESIDENT
TOPT.CTB EQU	20	
TOPT.CTM EQU	Y'00000800'	PREVENT SVC 6 CONTROL CALL
TOPT.CMB EQU	21	
TOPT.CMM EQU	Y'00000400'	PREVENT SVC 6 COMMUNICATION CALL
TOPT.S6B EQU	22	
TOPT.S6M EQU	Y'00000200'	SVC 6 CONTINUE
TOPT.DFB EQU	23	
TOPT.DFM EQU	Y'00000100'	USING DOUBLE F/P

TASK OPTIONS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
TOPT.RLB EQU	24	
TOPT.RLM EQU	Y'00000080'	ALLOW ROLL-OUT
TOPT.OVB EQU	25	
TOPT.OVM EQU	Y'00000040'	USE OVERLAY
TOPT.AEB EQU	26	
TOPT.AEM EQU	Y'00000020'	ACCOUNTING ENABLED
TOPT.STB EQU	27	
TOPT.STM EQU	Y'00000010'	SYSTEM TASK
TOPT.FAB EQU	28	
TOPT.FAM EQU	Y'00000008'	FILE ACCOUNT PRIVILEGES
TOPT.BDE EQU	29	
TOPT.BDM EQU	Y'00000004'	ALLOW BARE DISC ASSIGN
TOPT.UVB EQU	30	
TOPT.UVM EQU	Y'00000002'	UNIVERSAL
TOPT.KCB EQU	31	
TOPT.KCM EQU	Y'00000001'	DO KEY CHECKS ON ASSIGN, E-TASK

TASK WAIT BITS

<u>LABEL</u>		<u>BIT NUMBER</u> <u>OR MASK</u>	<u>DESCRIPTION</u>
TWT.OIB	EQU	14	I/O WAIT ON ANOTHER TASK'S I/O
TWT.OIM	EQU	Y'00020000'	
TWT.IWB	EQU	15	INTERCEPT WAIT
TWT.IWM	EQU	Y'00010000'	
TWT.IOB	EQU	16	
TWT.IOM	EQU	Y'00008000'	I/O WAIT
TWT.CNB	EQU	17	
TWT.CNM	EQU	Y'00004000'	(ANY) IOB WAIT
TWT.CWB	EQU	18	
TWT.CWM	EQU	Y'00002000'	CONSOLE WAIT
TWT.LWB	EQU	19	
TWT.LWM	EQU	Y'00001000'	LOAD WAIT
TWT.DMB	EQU	20	
TWT.DMM	EQU	Y'00000800'	DORMANT
TWT.TWB	EQU	21	
TWT.TWM	EQU	Y'00000400'	TRAP WAIT
TWT.TDB	EQU	22	
TWT.TDM	EQU	Y'00000200'	TIME-OF-DAY WAIT
TWT.TKB	EQU	23	
TWT.TKM	EQU	Y'00000100'	SUSPENDED
TWT.TMB	EQU	24	
TWT.TMM	EQU	Y'00000080'	INTERVAL WAIT
TWT.TRB	EQU	25	
TWT.TRM	EQU	Y'00000040'	TERMINAL WAIT

TASK WAIT BITS(continued)

LABEL		BIT NUMBER OR MASK	DESCRIPTION	
TWT.ROB	EQU	26		
TWT.ROM	EQU	Y'00000020'	ROLL-PENDING WAIT	
TWT.IIB	EQU	27		
TWT.IIM	EQU	Y'00000010'	INTERCEPT INITIALIZATION	(MTM)
TWT.ITB	EQU	28		
TWT.ITM	EQU	Y'00000008'	INTERCEPT TERMINATION	(MTM)
TWT.COB	EQU	29		
TWT.COM	EQU	Y'00000004'	CONNECTION WAIT	
TWT.ACB	EQU	30		
TWT.ACM	EQU	Y'00000002'	ACCOUNTING WAIT	
TM6VW	EQU	Y'0003FFFE'	VALID WAITS	

TASK STATUS BITS

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
TSTT.FPB EQU	1	
TSTT.FPM EQU	Y'40000000'	FAULT PENDING
TSTT.AWB EQU	2	
TSTT.AWM EQU	Y'20000000'	APU WAITING FOR THIS TASK
TSTT.COB EQU	3	
TSTT.COM EQU	Y'10000000'	CPU OVERRIDE
TSTT.LPB EQU	4	
TSTT.LPM EQU	Y'08000000'	LPU DIRECTED
TSTT.TQB EQU	5	
TSTT.TQM EQU	Y'04000000'	EVENT PENDING ON TASK EVENT QUEUE
TSTT.ESB EQU	6	
TSTT.ESM EQU	Y'02000000'	TASK EXECUTING EVENT SERVICE
TSTT.NIB EQU	7	
TSTT.NIM EQU	Y'01000000'	NS INTERCEPT IN PROGRESS
TSTT.PSB EQU	9	
TSTT.PSM EQU	Y'00400000'	PSEUDO-TCB
TSTT.TSB EQU	10	
TSTT.TSM EQU	Y'00200000'	TIME SLICE PENDING
TSTT.AOB EQU	11	

TASK STATUS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
TSTT.AOH EQU	Y'00100000'	ACCOUNTING OVERFLOW PEND.
TSTT.IWB EQU	12	
TSTT.IWH EQU	Y'00080000'	TASK IN ITAM RS I/O WAIT
TSTT.RIB EQU	13	
TSTT.RIM EQU	Y'00040000'	ROLL IN PENDING
TSTT.TRB EQU	14	
TSTT.TRM EQU	Y'00020000'	TERMINAL WAIT PENDING
TSTT.ITB EQU	15	
TSTT.ITM EQU	Y'00010000'	CONSOLE INTERCEPT TERMINATION PEND
TSTT.IIB EQU	16	
TSTT.IIM EQU	Y'00008000'	INTRPT. INSTRU. INCOMP. ON APU
TSTT.RSB EQU	17	
TSTT.RSM EQU	Y'00004000'	RS STATE
TSTT.PPB EQU	18	
TSTT.PPM EQU	Y'00002000'	PAUSE PENDING
TSTT.GTB EQU	19	
TSTT.GTM EQU	Y'00001000'	GRABTASK SINCE LAST SVCL
TSTT.ASB EQU	20	
TSTT.ASM EQU	Y'00000800'	RSA STATE (RS WITH ALTERNATE SAVE)
TSTT.IPB EQU	21	
TSTT.IPM EQU	Y'00000400'	I/O WAIT PENDING
TSTT.LDB EQU	22	
TSTT.LDM EQU	Y'00000200'	LOADER TASK

TASK STATUS BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
TSTT.CPB EQU	23	
TSTT.CPM EQU	Y'00000100'	CANCEL PENDING
TSTT.APB EQU	24	
TSTT.APM EQU	Y'00000080'	ABNORMAL TERMINATED
TSTT.PWB EQU	25	
TSTT.PWM EQU	Y'00000040'	
TSTT.FVB EQU	26	
TSTT.FVM EQU	Y'00000020'	F/P REGS SAVE FLAG
TSTT.TKB EQU	27	
TSTT.TKM EQU	Y'00000010'	SUSPENDED PENDING
TSTT.UPB EQU	29	
TSTT.UPM EQU	Y'00000004'	INITIALIZATION (UDL,UTOP) PENDING
TSTT.SWB EQU	30	
TSTT.SWM EQU	Y'00000002'	Q ENTRY TSW SWAP PENDING
TSTT.ETB EQU	31	
TSTT.ETM EQU	Y'00000001'	SYSTEM ET STATE (SVC 3)
TSTTPEN1 EQU	TSTT.RIM+TSTT.TRM+TSTT.ITM+TSTT.APM+TSTT.CPM	
TSTTPEN2 EQU	TSTTPEN1+TSTT.PPM+TSTT.TKM+TSTT.SWM+TSTT.PWM+TSTT.TSM	
TSTTPEND EQU	TSTTPEN2+TSTT.AOM+TSTT.TQM+TSTT.FPM	

DCB FLAG BITS

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
DFLG.BLB DS	1	BULK DEVICE FLAG
DFLG.BLM EQU	Y'80000000'	
DFLG.LNB DS	1	ON-LINE DEVICE FLAG
DFLG.LNM EQU	Y'40000000'	
DFLG.SMB DS	1	SYSTEM MESSAGE TO CONSOLE DEVICE
DFLG.SMM EQU	Y'20000000'	
DFLG.ERB DS	1	ESR WAITING FOR DIR TO COMPLETE
DFLG.ERM EQU	Y'10000000'	
DFLG.TOB DS	1	ACTIVE I/O TIMEOUT BEFORE DIR DONE
DFLG.TOM EQU	Y'08000000'	
DFLG.DLB DS	1	DELETE PSEUDO-DCB
DFLG.DLM EQU	Y'04000000'	
DFLG.CNB DS	1	CONSOLE IDENTIFIER
DFLG.CNM EQU	Y'02000000'	
DFLG.UCB DS	1	UNCANCELLABLE DEVICE
DFLG.UCM EQU	Y'01000000'	
DFLG.S6B DS	1	SVC-6 CONNECTION TABLE
DFLG.S6M EQU	Y'00800000'	
DFLG.WPB DS	1	WRITE PROTECTED DEVICE
DFLG.WPM EQU	Y'00400000'	
DFLG.ITB DS	1	ITAM SUPPORTED DEVICE
DFLG.ITM EQU	Y'00200000'	
DFLG.AFB DS	1	ASSIGNED FOR SVC-15 ACCESS
DFLG.AFM EQU	Y'00100000'	

DCB FLAG BITS(continued)

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
DFLG.HIB DS	1	SVC-1 HALT I/O DEVICE
DFLG.HIM EQU	Y'00080000'	
DFLG.PWB DS	1	TIME OUT DUE TO POWER FAIL
DFLG.PWM EQU	Y'00040000'	
DFLG.MDB DS	1	MULTIPLE DCB
DFLG.MDM EQU	Y'00020000'	
DFLG.PSB DS	1	PSEUDO-DCB BIT
DFLG.PSM EQU	Y'00010000'	
DFLG.VFB DS	1	SUPPORTS VFC
DFLG.VFM EQU	Y'00008000'	
DFLG.PNB DS	1	POWER FAIL - NO OUTSTANDING I/O
DFLG.PNM EQU	Y'00004000'	
DFLG.MMB DS	1	MMD-TYPE DISC
DFLG.NMM EQU	Y'00002000'	
DFLG.MGB DS	1	MAGTAPE DEVICE
DFLG.MGM EQU	Y'00001000'	
DFLG.MAB DS	1	DEVICE IS MIRRORABLE
DFLG.MAM EQU	Y'00000800'	
DFLG.NTB DS	1	NO TIMEOUT DURING POWERFAIL
DFLG.NTM EQU	Y'00000400'	

IOB FLAG BITS

LABEL	BIT NUMBER OR MASK	DESCRIPTION
IOBF.CCB EQU	0	CONNECTION COMPLETE
IOBF.CCM EQU	X'8000'	
IOBF.IOB EQU	1	CALLER IN -I/O WAIT
IOBF.IOM EQU	X'4000'	
IOBF.TPB EQU	2	CALLER EXPECTS AN I/O TRAP
IOBF.TPM EQU	X'2000'	
IOBF.COB EQU	3	CALLER IN CONNECTION WAIT
IOBF.COM EQU	X'1000'	
IOBF.ISB EQU	4	ISPTAB RESET FLAG (1=NO RESET)
IOBF.ISM EQU	X'0800'	
IOBF.XRB EQU	5	EDMA EXCLUSIVE REQUEST
IOBF.XRM EQU	X'0400'	
IOBF.ECB EQU	6	EDMA CONNECTED FLAG
IOBF.ECM EQU	X'0200'	
IOBF.NDB EQU	7	NO DISCONNECTION AT I/O COMPLETION
IOBF.NDM EQU	X'0100'	
IOBF.SYB EQU	8	SYSTEM I/O
IOBF.SYM EQU	X'0080'	
IOBF.NPB EQU	9	NO PURGE ON ERROR
IOBF.NPM EQU	X'0040'	
IOBF.NHB EQU	10	NON-HALTABLE REQUEST
IOBF.NHM EQU	X'0020'	
IOBF.WPB EQU	11	WAIT PENDING FLAG
IOBF.WPM EQU	X'0010'	
IOBF.RB EQU	12	RESERVED.

IOB FLAG BITS(continued)

<u>LABEL</u>		<u>BIT NUMBER</u> <u>OR MASK</u>	<u>DESCRIPTION</u>
IOBF.RM	EQU	X'0008'	
IOBF.DRB	EQU	13	DO NOT FREE IOB AT CONNECT TIME
IOBF.DRM	EQU	X'0004'	
IOBF.VFB	EQU	14	
IOBF.VFM	EQU	X'0002'	VERTICAL FORMS CONTROL FLAG
IOBT.TCB	EQU	1	IN TCB POOL RESV'D AT TET TIME
IOBT.SP	EQU	2	SPARE TCB (ONE PER EACH TASK)
IOBT.DCB	EQU	3	CONTAINED IN DCB
IOBT.FCB	EQU	4	CONTAINED IN CONTIGUOUS FILE'S FCB

FCB FLAG BITS

<u>LABEL</u>	<u>BIT NUMBER OR MASK</u>	<u>DESCRIPTION</u>
FFLG.BAB EQU	0	BUFFERED ACCESS METHOD
FFLG.BAM EQU	Y'80000000'	
FFLG.USB EQU	1	USE FLAG
FFLG.USM EQU	Y'40000000'	
FFLG.SPB EQU	8	SPOOL FILE
FFLG.SPM EQU	Y'00800000'	
FFLG.UPB EQU	9	FILE HAS BEEN UPDATED
FFLG.UPM EQU	Y'00400000'	
FFLG.PNB EQU	17	I/O PENDING ON CONTIG FILE
FFLG.PNM EQU	Y'00004000'	
FDCD.CO EQU	0	CONTIGUOUS FILE
FDCD.EC EQU	1	EXTENDABLE CONTIGUOUS FILE
FDCD.IN EQU	2	INDEXED FILE
FDCD.NB EQU	3	NON-BUFFERED INDEXED FILE
FDCD.LR EQU	6	LONG RECORD FILE
FDCD.HI EQU	6	HIGHEST DEFINED FILE TYPE
FDCD.IT EQU	7	DATA COMM LCB
FDCD.DEV EQU	16	FIRST PHYSICAL DEVICE CODE
FINX.PRIV EQU	0	PREVIOUS INDEX POINTER
FINX.NXT EQU	4	NEXT INDEX POINTER

NOTES