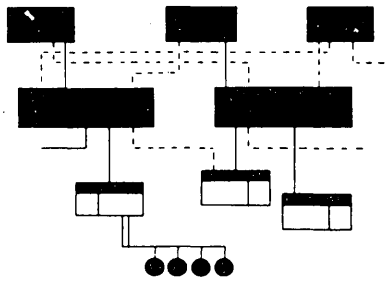
GE-625/635 GECOS-III Flowcharts

Information Systems

Information Systems Equipment

SOFTWARE MAINTENANCE DOCUMENT



GENERAL ELECTRIC

CPB-1500

CPB-1500

GE-625/635 GECOS-III Flowcharts

SOFTWARE MAINTENANCE DOCUMENT

March 1968

INFORMATION SYSTEMS

GENERAL ELECTRIC

PREFACE

GE-625/635 GECOS-III FLOWCHARTS contains flowcharts for selected sections of the GECOS-III system. The flowcharts are arranged in CPB-1500 in the same sequential order as the corresponding text in the series of documents CPB-1488 to CPB-1499.

The initial page of each flowchart carries the full title of the corresponding program or routine. In addition, each flowchart page has a quick reference box at the top of the page to identify the flowchart on that page. The quick reference box contains the symbolic name of the program, the symbolic name of the higher level module, and when available, the entry point into the program.

Flowchart symbols follow the United States American Standard, USAS X3.5-1966.

Suggestions and criticisms relative to form, content, purpose, or use of this manual are invited. Comments may be sent on the Document Review Sheet in the back of this manual or may be addressed directly to Documentation Standards and Publications, B-90, Computer Equipment Department, General Electric Company, 13430 North Black Canyon Highway, Phoenix, Arizona 85029.

© 1968 by General Electric Company

(2M 3-68)

CONTENTS

1. DISPATCHER AND PERIPHERAL ALLOCATION..... 1

REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP..... 3
DSP (EP1) .MDISP..... 3
GERDAD ROADBLOCK, GRD..... 17
GRD (EP2) .MDISP..... 17
FORCED RELINQUISH, FRLC..... 19
FRLC (EP3) .MDISP..... 19
GERELC RELINQUISH CONTROL, RLC..... 20
RLC (EP4) .MDISP..... 20
GEENDC END COURTESY CALL, ENCC..... 21
ENCC (EP5) .MDISP..... 21
ENABLE PROGRAM, ENB..... 22
ENB (EP6) .MDISP..... 22
PROGRAM NUMBER AT FRONT OF QUEUE, DSPQH..... 24
DSPQH (EP7) .MDISP..... 24
PROGRAM NUMBER AT END OF QUEUE, DSPQT..... 25
DSPQT (EP8) .MDISP..... 25
ACCUMULATE PROCESSOR TIME, DACNB..... 26
DACNB (EP9) .MDISP..... 26
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR..... 27
DRSTR (EP10) .MDISP..... 27
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT..... 28
DSCNT (EP11) .MDISP..... 28
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA..... 29
DMIOA (EP12) .MDISP..... 29
SET ALARM, SCK..... 32
SCK (EP13) .MDISP..... 32
PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT, DSPQM.... 33
DSPQM (EP14) .MDISP..... 33
SAVE TRACE BUFFER AND WRITE OUT, TRACF..... 34
TRACF (EP15) .MDISP..... 34
INITIALIZATION, .IDISP..... 35
.IDISP .MDISP..... 35
.CALL MACRO, HCL..... 36
HCL .MDISP..... 36
.EXIT MACRO, HEX..... 38
HEX .MDISP..... 38
.GOTO MACRO, HGT..... 40
HGT .MDISP..... 40
SYSTEM TRACE, TRACE..... 49
TRACE .MDISP..... 49
OPEN SYSTEM GATES, OPGAT..... 51
OPGAT .MDISP..... 51
CLOSE SYSTEM GATES, SHUG..... 52
SHUG .MDISP..... 52

2. FAULT PROCESSING AND SERVICE MME'S..... 53

SECONDARY FAULT VECTOR, FPRC..... 55
FPRC .MFALT..... 55
RECOGNIZE FAULT TYPE AND PROCESS, FLT..... 56
FLT .MFALT..... 56
TAKE DUMP FOR SYSTEM ABORT, FSB..... 77
FSB (EP1) .MFALT..... 77
GET ADDRESS FOR FILE CODE, FGAD (.MFALT)..... 78
FGAD (EP2) .MFALT..... 78
PUT INFORMATION IN FCB, FGCON..... 79
FGCON (EP3) .MFALT..... 79
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL..... 80
FGFIL (EP4) .MFALT..... 80
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP..... 81
FGLAP (EP5) .MFALT..... 81
RESET PROGRAM SWITCH WORD BITS, FGRET..... 82
FGRET (EP6) .MFALT..... 82
SET PROGRAM SWITCH WORD BITS, FGSET..... 83
FGSET (EP7) .MFALT..... 83
PROVIDE DATA AND TIME OF DAY, FGTIM..... 84
FGTIM (EP8) .MFALT..... 84
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT)..... 85
FGLP (EP9) .MFALT..... 85
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM..... 86
FEMM (EP10) .MFALT..... 86
FIND PAT FROM FILE CODE, FNDE..... 87
FNDE (EP11) .MFALT..... 87
DEALLOCATE MEMORY, GMRL (.MFALT)..... 88
GMRL (EP12) .MFALT..... 88
CALL .MDUMP AT EP2, BOOT..... 89
BOOT (EP13) .MFALT..... 89
FAULT PROCESSOR INITIALIZATION, .IFALT..... 91
.IFALT .MFALT..... 91
GET ADDRESS FOR FILE CODE, FGAD (.MFLT1)..... 95
FGAD (EP1) .MFLT1..... 95
PUT INFORMATION IN FCB, FCON..... 96
FCON (EP2) .MFLT1..... 96
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW..... 100
FLSW (EP3) .MFLT1..... 100
DEALLOCATE MEMORY, GMRL (.MFLT1)..... 101
GMRL (EP4) .MFLT1..... 101
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1)..... 105
FGLP (EP5) .MFLT1..... 105
GIVE I/O ACTION PRIORITY, FPRIO..... 106
FPRIO (EP6) .MFLT1..... 106
RESET BAR TO SMALLER AREA, FLBAR..... 107
FLBAR (EP7) .MFLT1..... 107
ALLOW A PROGRAM TO DELAY, FWAKE..... 109
FWAKE (EP8) .MFLT1..... 109
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM..... 110
GMRLM (EP9) .MFLT1..... 110

3. I/O SUPERVISION..... 111

LINK I/O TO END OF QUEUE, LINK..... 113
LINK (EP1) .MIOS..... 113
LINK I/O TO FRONT OF QUEUE, LINKF..... 116
LINKF (EP2) .MIOS..... 116
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR..... 117

LINKR (EP3) .MIOS..... 117
ASSIGN AN I/O ENTRY, QUEUE..... 118
QUEUE (EP4) .MIOS..... 118
GEINOS MME GEINOS PROCESSOR, INOS..... 120
INOS (EP5) .MIOS..... 120
GESPEC MME GESPEC PROCESSOR, SPEC..... 133
SPEC (EP6) .MIOS..... 133
MASTER MESSAGE PROCESSOR, ITYM..... 135
ITYM (EP7) .MIOS..... 135
POINTER VALIDATION, PTRVL..... 138
PTRVL .MIOS..... 138
DCW POINTER VALIDATION, DCWCK..... 139
DCWCK .MIOS..... 139
FIND FILE CODE, FNDFC..... 141
FNDFC .MIOS..... 141
FORMAT ERROR ACCOUNTING RECORD, FMTAR..... 142
FMTAR .MIOS..... 142
INTERRUPT HANDLER, IOTRM..... 144
IOTRM .MIOS..... 144
UNLINK I/O ENTRY, UNLNK..... 169
UNLNK .MIOS..... 169
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX..... 170
ILPCX .MIOS..... 170
STATUS RETURN, STRET..... 171
STRET .MIOS..... 171
START I/O, STIO..... 175
STIO .MIOS..... 175
CONNECT MULTIRECORD SIMULATION DCW, STIOM..... 187
STIOM .MIOS..... 187
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER... 188
TYPER .MIOS..... 188
CONNECT SELECTED GESPECED ENTRY, STGPC..... 189
STGPC .MIOS..... 189
RESUME I/O FOR PROGRAM, RSMIO..... 190
RSMIO (EP8) .MIOS..... 190
ABORT I/O FOR PROGRAM, ABTIO..... 192
ABTIO (EP9) .MIOS..... 192
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT..... 196
GSTRT (EP10) .MIOS..... 196
RESUME I/O ON CHANNEL, RSMCH..... 197
RSMCH (EP12) .MIOS..... 197
ACCOUNTING FILE REQUEST, ACTFL..... 198
ACTFL (EP13) .MIOS..... 198
MAIN IOS MODULE INITIALIZATION, .IIOS..... 201
.IIOS .MIOS..... 201
CARD PUNCH INTERRUPT HANDLER, CPIT..... 204
CPIT (EP1) .MCPIO..... 204
CARD PUNCH REQUEST, CPIO..... 207
CPIO (EP2) .MCPIO..... 207
CARD PUNCH INITIALIZATION, .ICPIO..... 208
.ICPIO .MCPIO..... 208
MDS200 INTERRUPT HANDLER, DRIT..... 209
DRIT (EP1) .MDR20..... 209
MDS200 REQUEST, DRIO..... 212
DRIO (EP2) .MDR20..... 212
MDS200 SELECT, DRSL..... 214
DRSL (EP3) .MDR20..... 214
MDS200 ERROR AND EOF RECOVERY, DRGP..... 217
DRGP (EP4) .MDR20..... 217
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP..... 222

NEGATIVE EP .MDR20..... 222
MDS200 INITIALIZATION, .IDR20..... 230
.IDR20 .MDR20..... 230
DSU200 INTERRUPT HANDLER, DSIT..... 231
DSIT (EP1) .MDS20..... 231
DSU200 REQUEST, DSIO..... 233
DSIO (EP2) .MDS20..... 233
DSU200 SELECT, DSSL..... 234
DSSL (EP3) .MDS20..... 234
DSU200 ERROR AND EOF RECOVERY, DSGP..... 236
DSGP (EP4) .MDS20..... 236
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP..... 247
NEGATIVE EP .MDS20..... 247
DSU200 INITIALIZATION, .IDS20..... 254
.IDS20 .MDS20..... 254
CARD READER INTERRUPT HANDLER, CRIT..... 255
CRIT (EP1) .MGPIO..... 255
CARD READER REQUEST, CRIO..... 257
CRIO (EP2) .MGPIO..... 257
CARD READER INITIALIZATION, .IGPIO..... 258
.IGPIO .MGPIO..... 258
MAGNETIC TAPE INTERRUPT HANDLER, MTIT..... 259
MTIT (EP1) .MMTAP..... 259
MAGNETIC TAPE REQUEST, MTIO..... 264
MTIO (EP2) .MMTAP..... 264
MAGNETIC TAPE INITIALIZATION, .IMTAP..... 265
.IMTAP .MMTAP..... 265
PRINTER INTERRUPT HANDLER, PRIT..... 266
PRIT (EP1) .MPRIO..... 266
PRINTER REQUEST, PRIO..... 269
PRIO (EP2) .MPRIO..... 269
PRINTER INITIALIZATION, .IPRIO..... 270
.IPRIO .MPRIO..... 270
PAPER TAPE INTERRUPT HANDLER, PTIT..... 271
PTIT (EP1) .MPTAP..... 271
PAPER TAPE REQUEST, PTIO..... 272
PTIO (EP2) .MPTAP..... 272
PAPER TAPE INITIALIZATION, .IPTAP..... 273
.IPTAP .MPTAP..... 273
TYPEWRITER INTERRUPT HANDLER, TYIT..... 274
TYIT (EP1) .MTYPE..... 274
TYPEWRITER REQUEST, TYIO..... 277
TYIO (EP2) .MTYPE..... 277
TYPEWRITER SELECT, TYSL..... 279
TYSL (EP3) .MTYPE..... 279
TYPEWRITER INITIALIZATION, .ITYPE..... 280
.ITYPE .MTYPE..... 280
NORMAL CLOSE OF ACCOUNTING FILE, ACTS1..... 281
ACTS1 (EP1) .MACTS..... 281
ERROR CLOSE OF ACCOUNTING FILE, ACTS2..... 282
ACTS2 (EP2) .MACTS..... 282
EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3..... 289
ACTS3 (EP3) .MACTS..... 289

4. TERMINATION AND SYSTEM OUTPUT..... 291

GESYOT
PROCESS MME GESYOT, MME..... 293
MME (EP1) .MSYOT..... 293

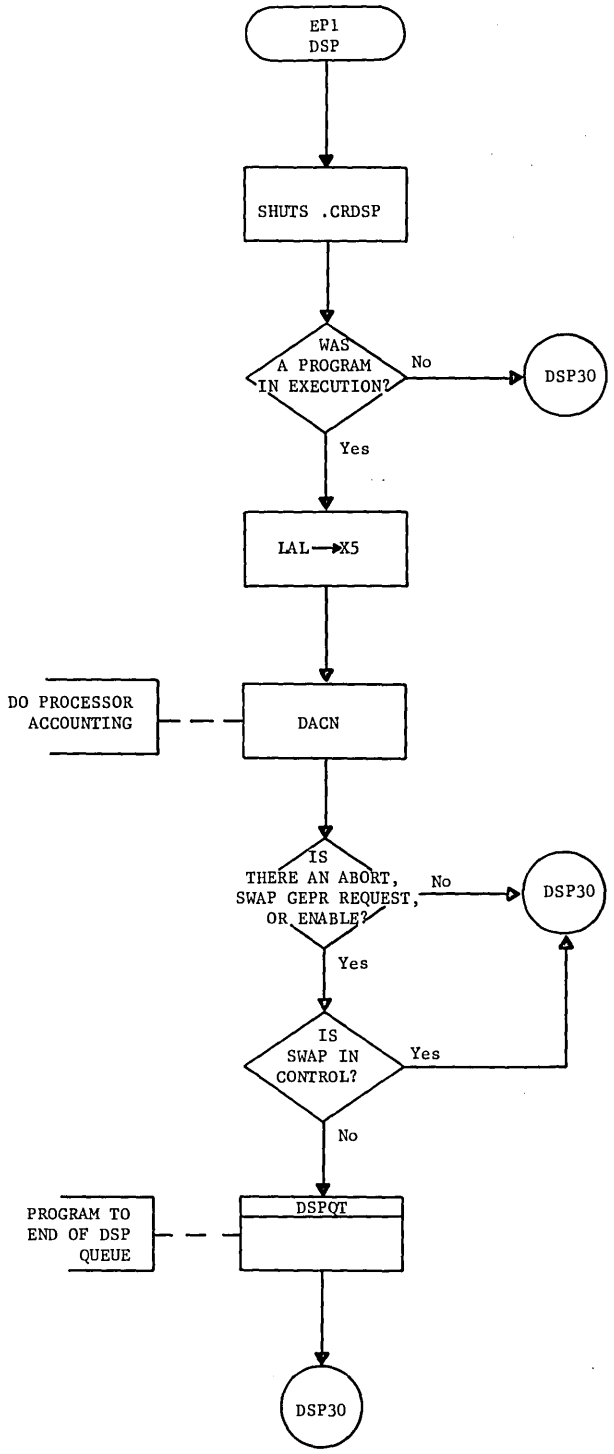
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC..... 310
EXEC (EP2) .MSYOT..... 310
PROCESS END OF ACTIVITY, EOA..... 311
EOA (EP3) .MSYOT..... 311
PROCESS END OF JOB, EOJ..... 313
EOJ (EP4) .MSYOT..... 313
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA..... 324
EXCA (EP5) .MSYOT..... 324
PROCESS CALL FROM GEIN, GEIN..... 325
GEIN (EP6) .MSYOT..... 325
DISPATCHER, DISP..... 326
DISP .MGEOT..... 326
START REPORT/ACTIVITY, STACT..... 336
STACT .MGEOT..... 336
READ INTO BUFFER, READ..... 338
READ .MGEOT..... 338
WRITE FROM BUFFER, WRITE..... 340
WRITE .MGEOT..... 340
COURTESY CALL FOR READ, CCCRD..... 351
CCCRD .MGEOT..... 351
COURTESY CALL FOR WRITE, CCCWR..... 354
CCCWR .MGEOT..... 354
COURTESY CALL FOR GESPEC, CCGSP..... 356
CCGSP .MGEOT..... 356
ASSIGN DEVICE AND JOB, ASIGN..... 366
ASIGN .MGEOT..... 366
START JOB, STRJB..... 368
STRJB .MGEOT..... 368
ANALYZE INPUT REQUEST, NEWJB..... 369
NEWJB .MGEOT..... 369
RELEASE BLINKS, FLUSH..... 373
FLUSH .MGEOT..... 373
CREATE FRONT BANNER, FOR..... 375
FOR (EP1) .MGOU1..... 375
CREATE TRAILER ENSIGN, AFT..... 377
AFT (EP2) .MGOU1..... 377
CREATE IDENT BANNER, IDENT..... 378
IDENT (EP3) .MGOU1..... 378
CREATE REPORT HEADER, START..... 379
START (EP1) .MGOU3..... 379

INDEX..... 381

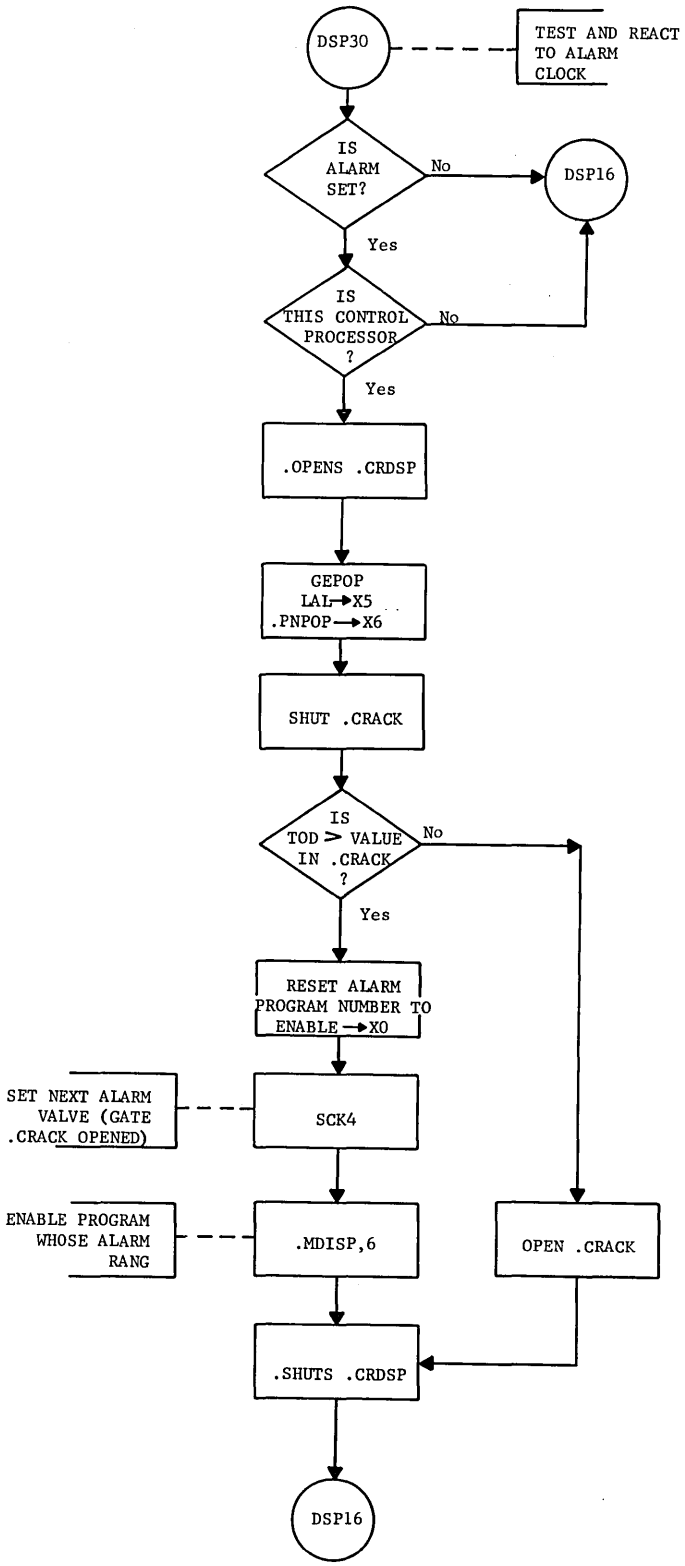
1. DISPATCHER AND PERIPHERAL ALLOCATION

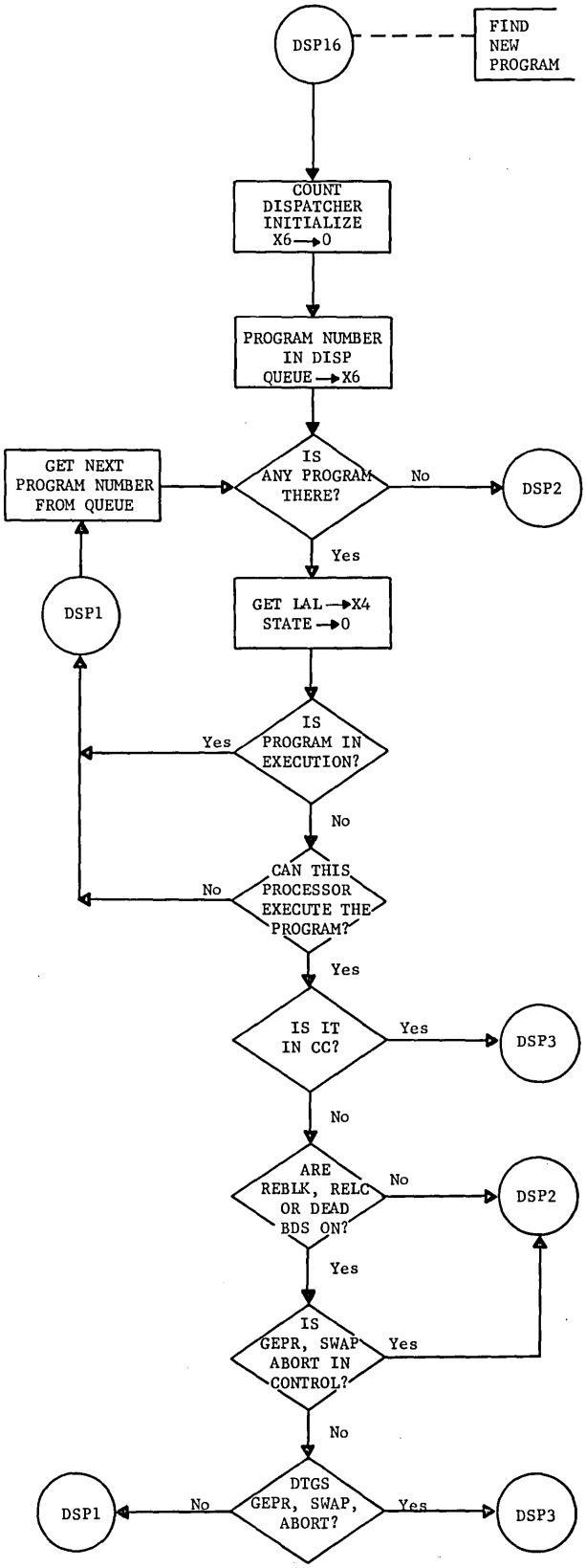
(Reference CPB-1491)

REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT

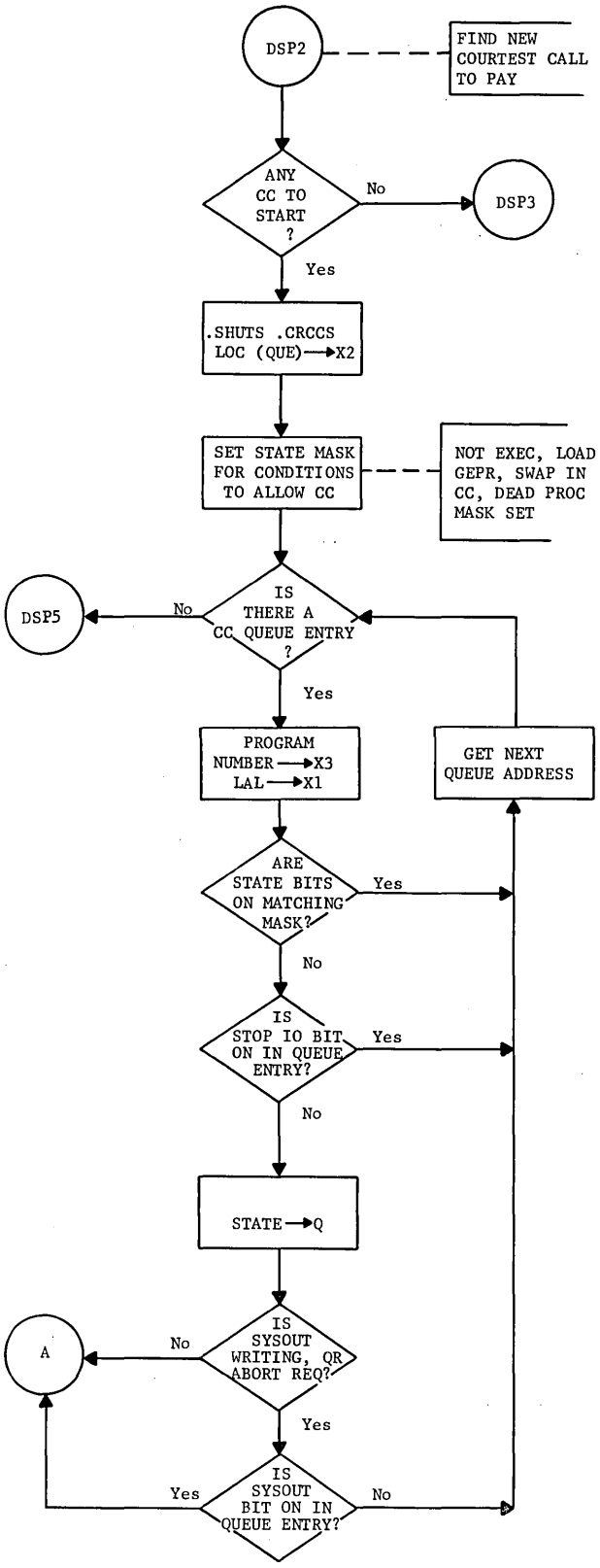


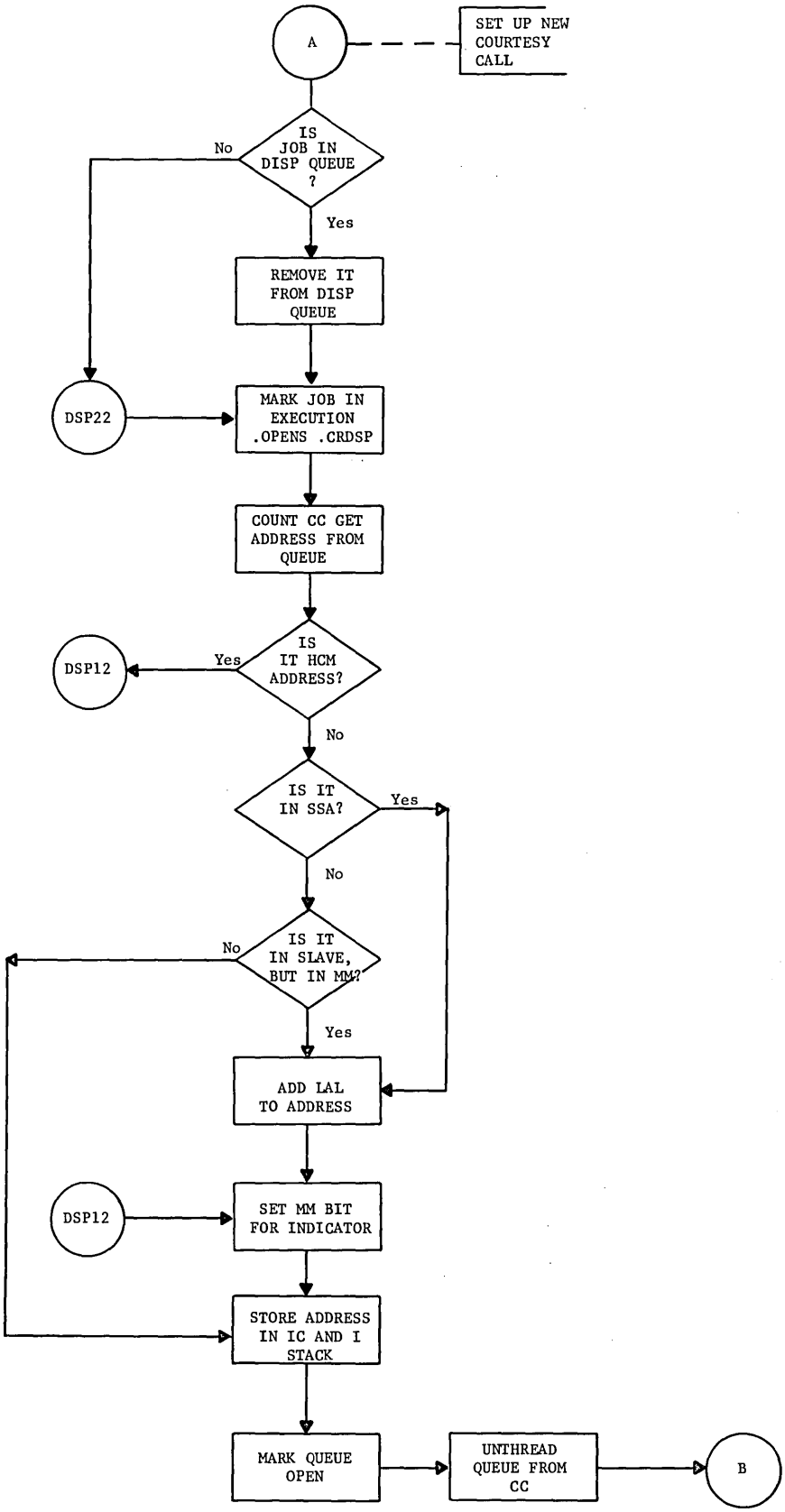
DSP (EP1)
.MDISP



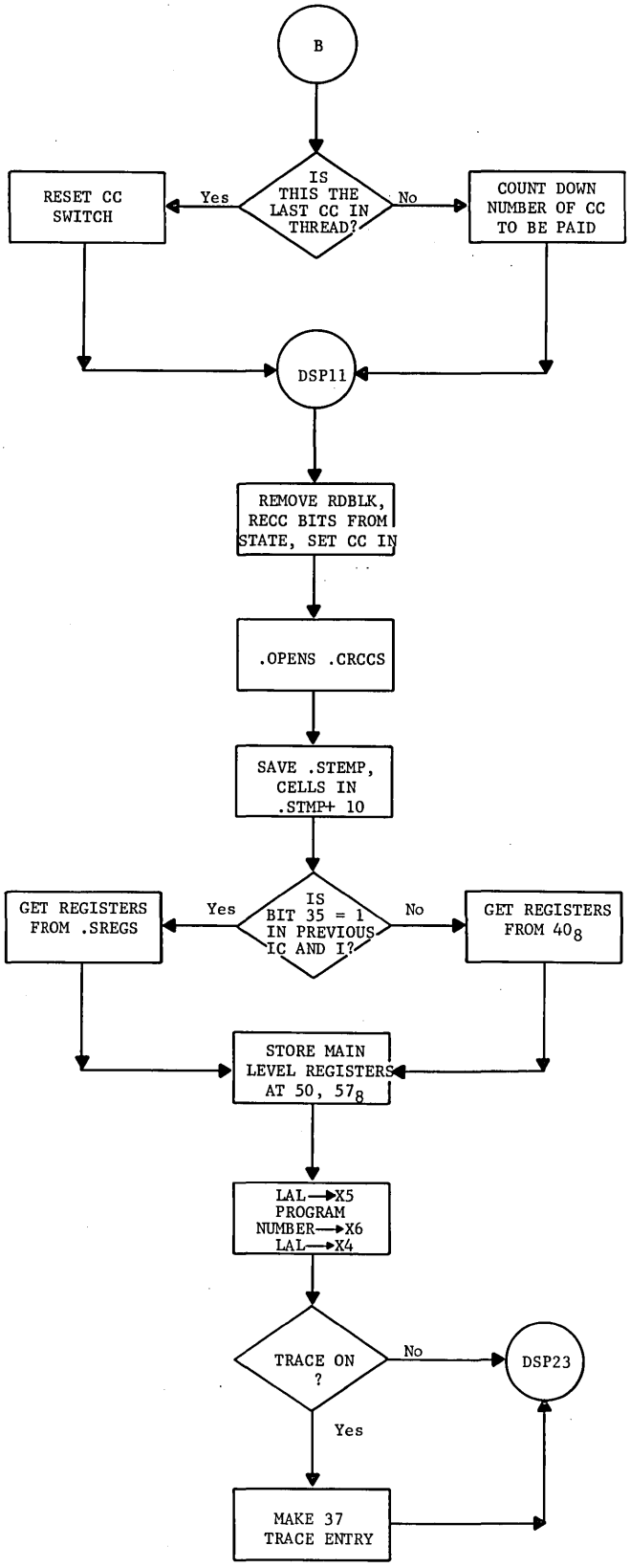

DSP16
FIND NEW PROGRAM
COUNT DISPATCHER INITIALIZE X6→0
PROGRAM NUMBER IN DISP QUEUE→X6
GET NEXT PROGRAM NUMBER FROM QUEUE
IS ANY PROGRAM THERE?
No
DSP2
Yes
DSP1
GET LAL→X4 STATE→0
IS PROGRAM IN EXECUTION?
Yes
No
CAN THIS PROCESSOR EXECUTE THE PROGRAM?
No
Yes
IS IT IN CC?
Yes
DSP3
No
ARE REBLK, RELC OR DEAD BDS ON?
No
DSP2
Yes
IS GEPR, SWAP ABORT IN CONTROL?
Yes
No
DTGS GEPR, SWAP, ABORT?
No
DSP1
Yes
DSP3

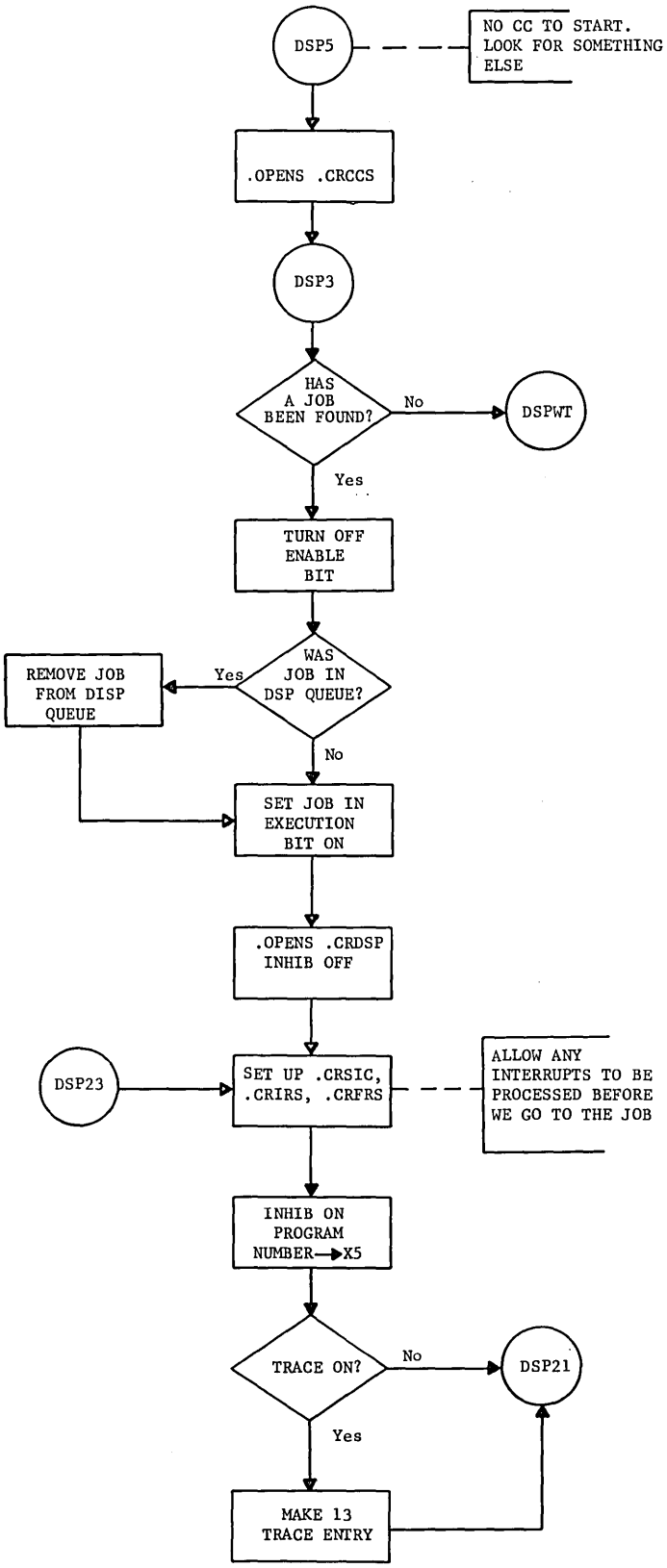
DSP (EP1)
.MDISP



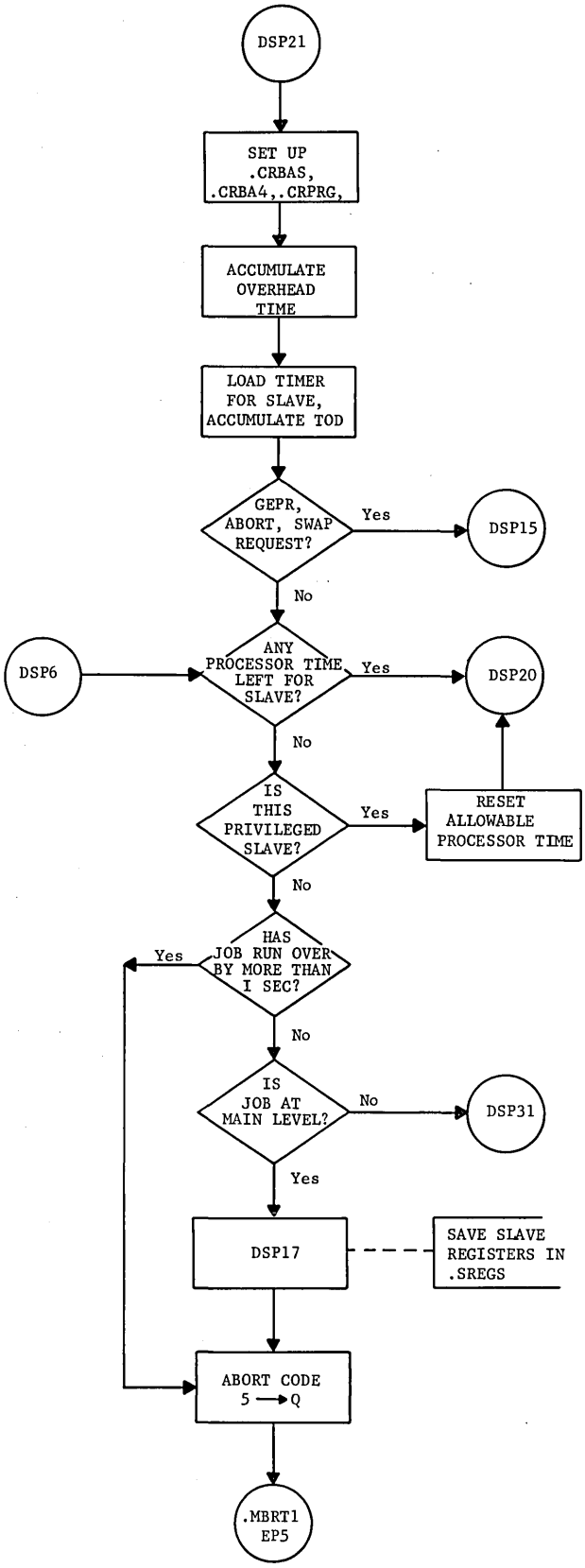

A
SET UP NEW COURTESY CALL
IS JOB IN DISP QUEUE ?
No
Yes
REMOVE IT FROM DISP QUEUE
DSP22
MARK JOB IN EXECUTION .OPENS .CRDSP
COUNT CC GET ADDRESS FROM QUEUE
IS IT HCM ADDRESS?
DSP12
Yes
No
IS IT IN SSA?
Yes
No
IS IT IN SLAVE, BUT IN MM?
No
Yes
ADD LAL TO ADDRESS
DSP12
SET MM BIT FOR INDICATOR
STORE ADDRESS IN IC AND I STACK
MARK QUEUE OPEN
UNTHREAD QUEUE FROM CC
B

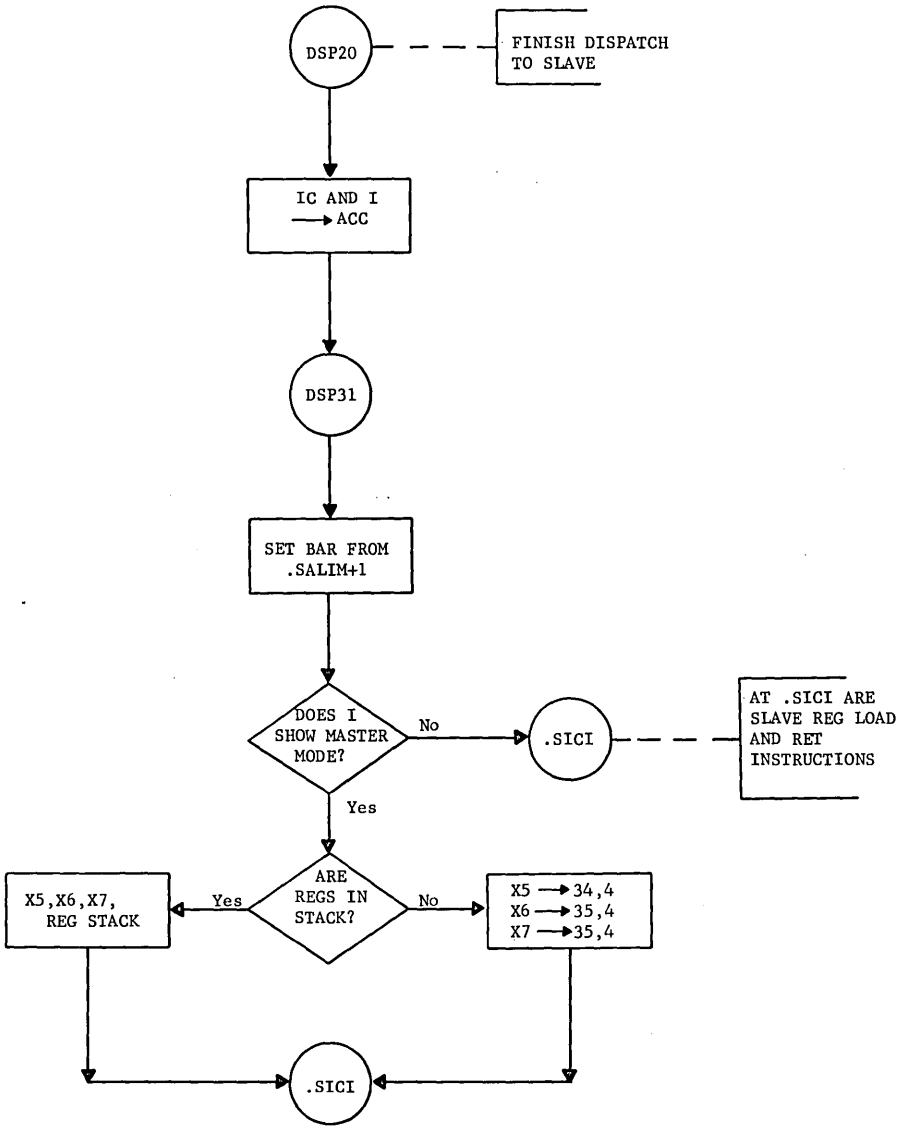
DSP (EP1)
.MDISP



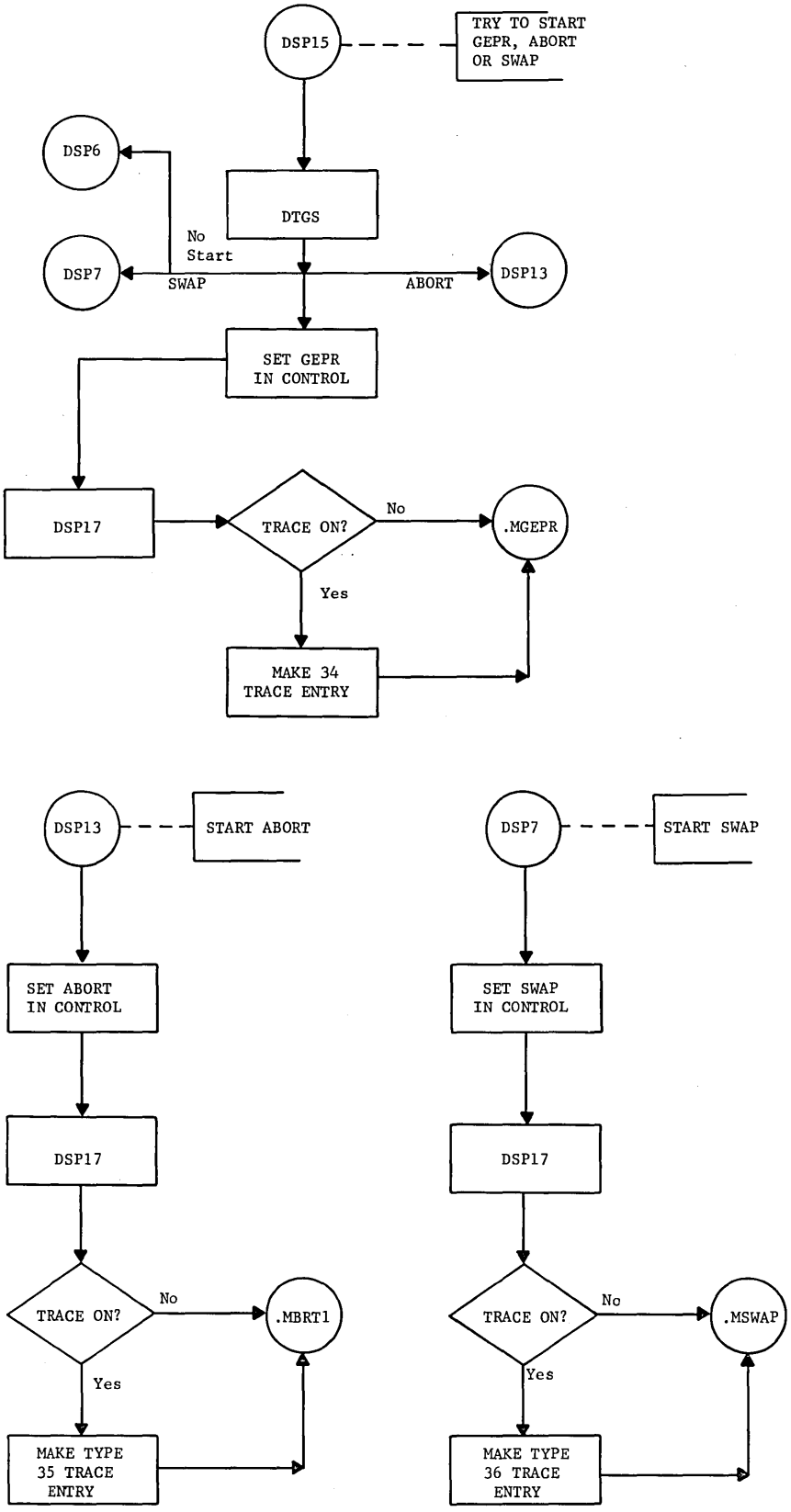

DSP5
NO CC TO START. LOOK FOR SOMETHING ELSE
.OPENS .CRCCS
DSP3
HAS A JOB BEEN FOUND?
No
DSPWT
Yes
TURN OFF ENABLE BIT
WAS JOB IN DSP QUEUE?
Yes
REMOVE JOB FROM DISP QUEUE
No
SET JOB IN EXECUTION BIT ON
.OPENS .CRDSP INHIB OFF
DSP23
SET UP .CRSIC, .CRIRS, .CRFRS
ALLOW ANY INTERRUPTS TO BE PROCESSED BEFORE WE GO TO THE JOB
INHIB ON PROGRAM NUMBER→X5
TRACE ON?
No
DSP21
Yes
MAKE 13 TRACE ENTRY

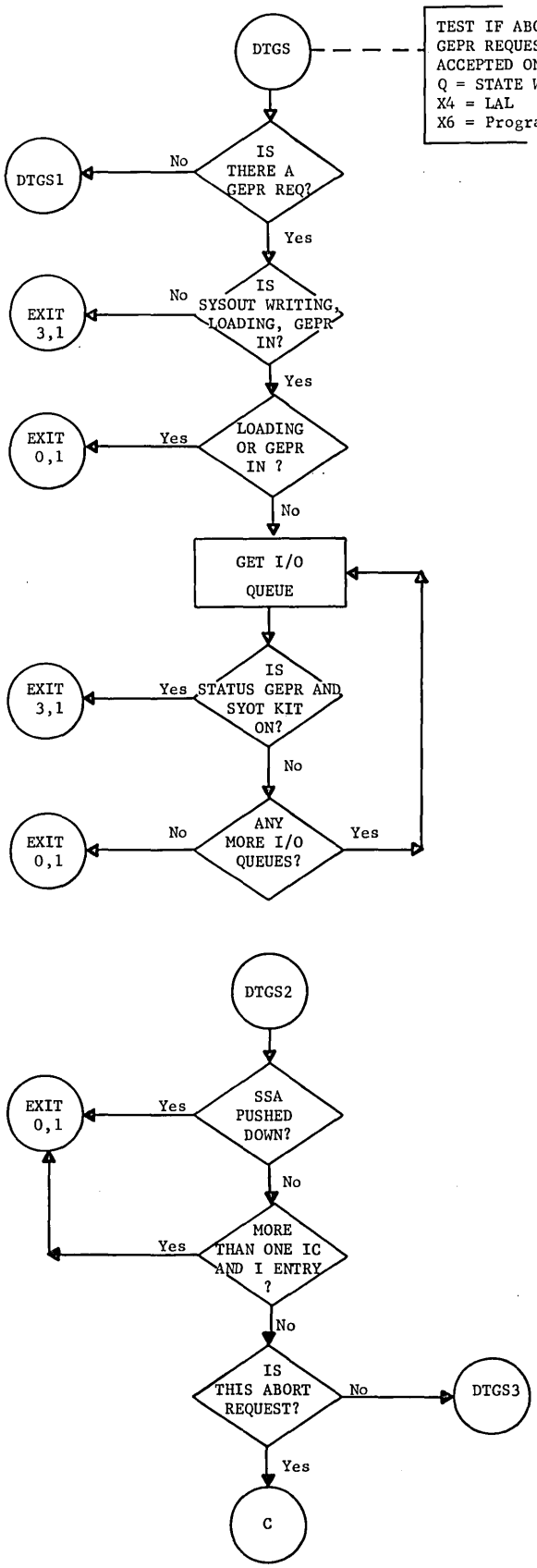
DSP (EP1)
.MDISP



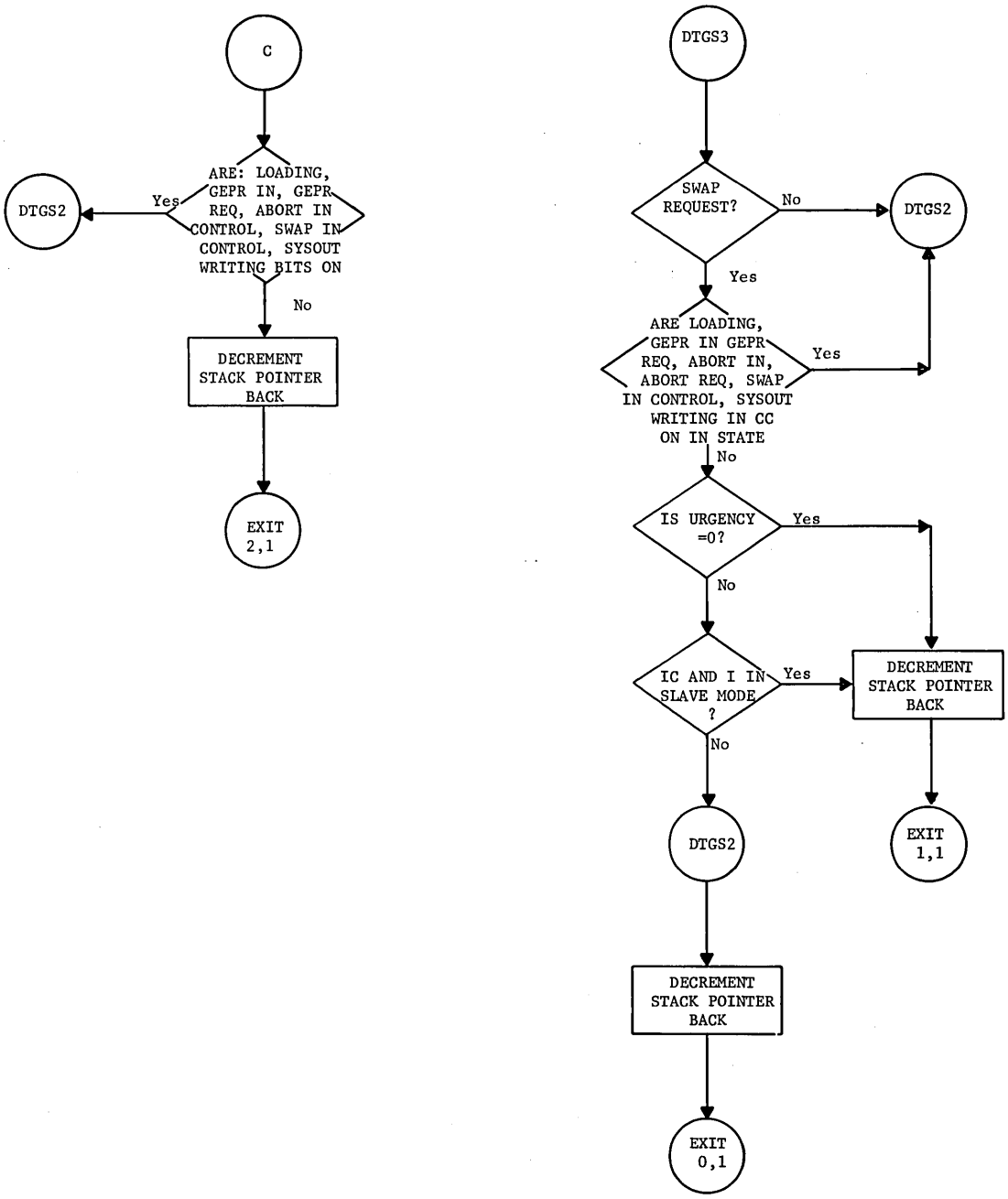

DSP20
FINISH DISPATCH TO SLAVE
IC AND I → ACC
DSP31
SET BAR FROM .SALIM+1
DOES I SHOW MASTER MODE?
No
.SICI
AT .SICI ARE SLAVE REG LOAD AND RET INSTRUCTIONS
Yes
ARE REGS IN STACK?
Yes
X5,X6,X7, REG STACK
No
X5 → 34,4
X6 → 35,4
X7 → 35,4
.SICI

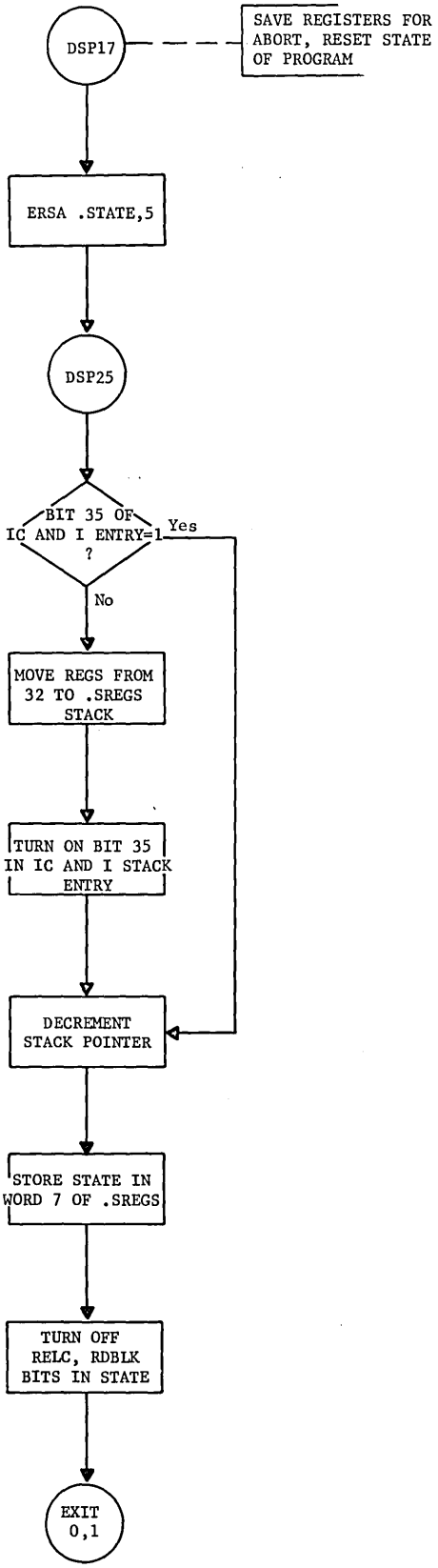
DSP (EP1)
.MDISP



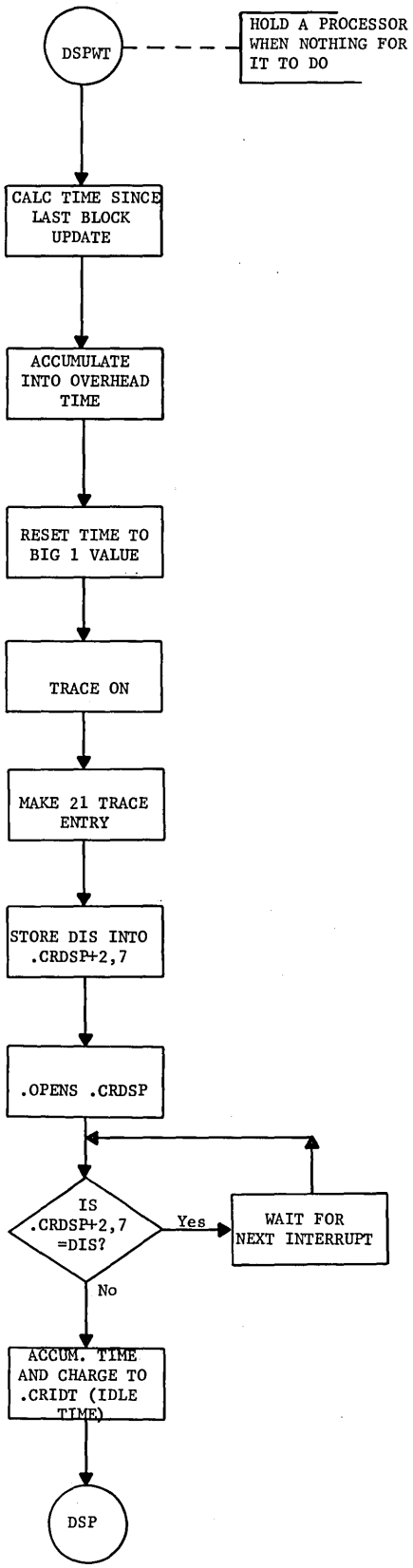

DTGS
TEST IF ABORT, SWAP, GEPR REQUEST CAN BE ACCEPTED ON ENTRY
Q = STATE WORD
X4 = LAL
X6 = Program Number
IS THERE A GEPR REQ?
No
DTGS1
Yes
IS SYSOUT WRITING, LOADING, GEPR IN?
No
EXIT 3,1
Yes
LOADING OR GEPR IN ?
Yes
EXIT 0,1
No
GET I/O QUEUE
IS STATUS GEPR AND SYOT KIT ON?
Yes
EXIT 3,1
No
ANY MORE I/O QUEUES?
No
EXIT 0,1
Yes
DTGS2
SSA PUSHED DOWN?
Yes
EXIT 0,1
No
MORE THAN ONE IC AND I ENTRY ?
Yes
No
IS THIS ABORT REQUEST?
No
DTGS3
Yes
C

DSP (EP1)
.MDISP

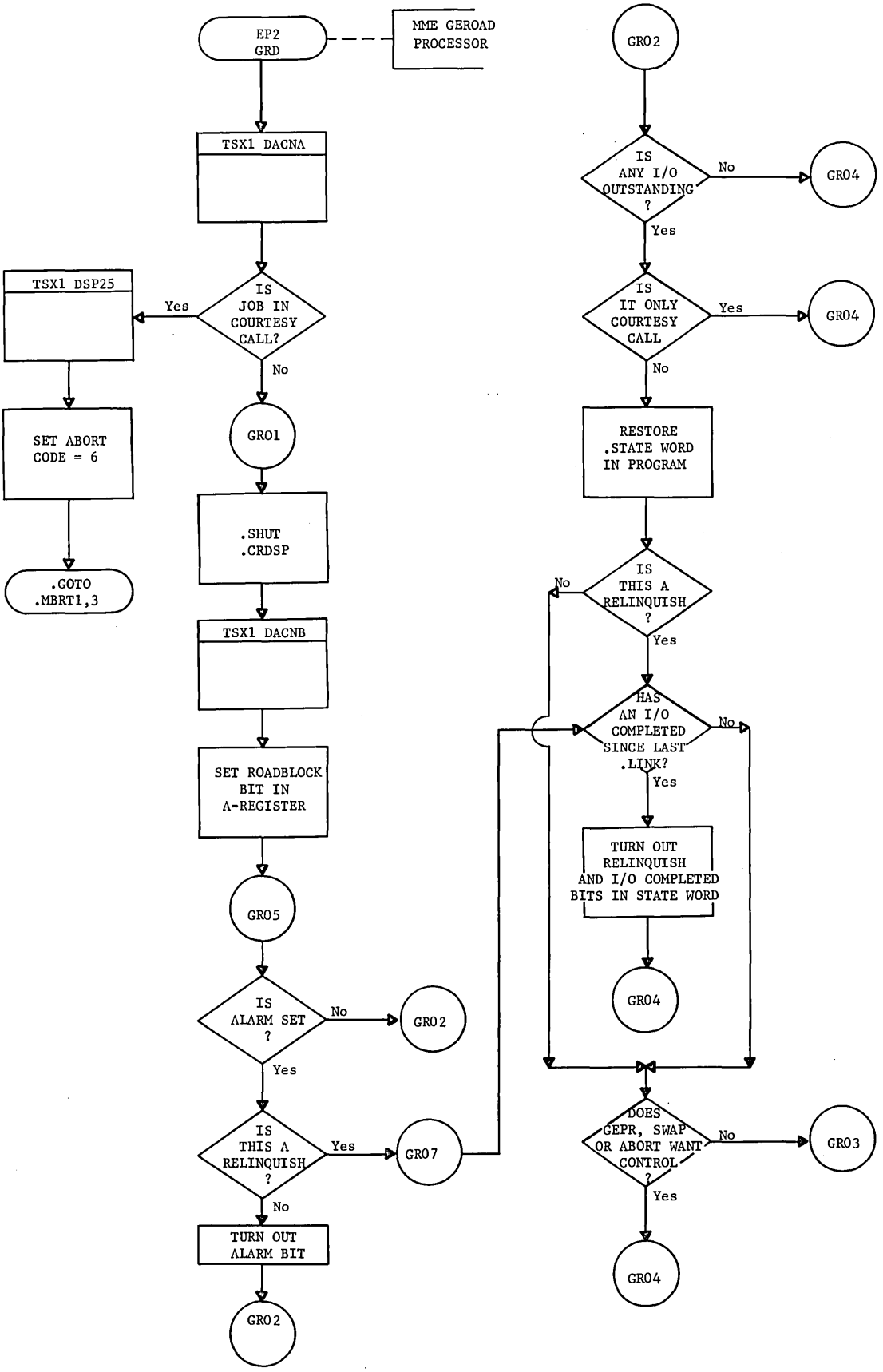



DSP17
SAVE REGISTERS FOR ABORT, RESET STATE OF PROGRAM
ERSA .STATE,5
DSP25
BIT 35 OF IC AND I ENTRY=1 ?
Yes
No
MOVE REGS FROM 32 TO .SREGS STACK
TURN ON BIT 35 IN IC AND I STACK ENTRY
DECREMENT STACK POINTER
STORE STATE IN WORD 7 OF .SREGS
TURN OFF RELC, RDBLK BITS IN STATE
EXIT 0,1

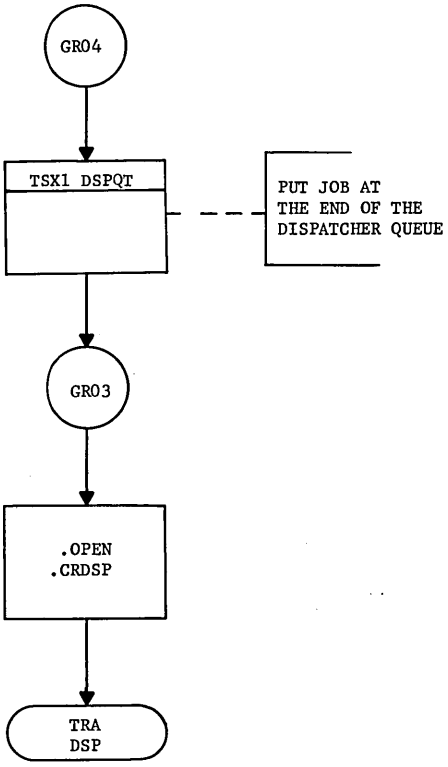
DSP (EP1)
.MDISP



ROADBLOCK

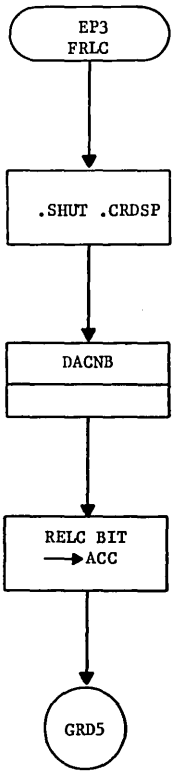

EP2
GRD
MME GEROAD
PROCESSOR
TSX1 DACNA
IS
JOB IN
COURTESY
CALL?
Yes
TSX1 DSP25
SET ABORT
CODE = 6
.GOTO
.MBRT1,3
No
GR01
.SHUT
.CRDSP
TSX1 DACNB
SET ROADBLOCK
BIT IN
A-REGISTER
GR05
IS
ALARM SET
?
No
GR02
Yes
IS
THIS A
RELINQUISH
?
Yes
GR07
No
TURN OUT
ALARM BIT
GR02
GR02
IS
ANY I/O
OUTSTANDING
?
No
GR04
Yes
IS
IT ONLY
COURTESY
CALL
Yes
GR04
No
RESTORE
.STATE WORD
IN PROGRAM
IS
THIS A
RELINQUISH
?
No
Yes
HAS
AN I/O
COMPLETED
SINCE LAST
.LINK?
No
Yes
TURN OUT
RELINQUISH
AND I/O COMPLETED
BITS IN STATE WORD
GR04
DOES
GEPR, SWAP
OR ABORT WANT
CONTROL
?
No
GR03
Yes
GR04

GRD (EP2)
.MDISP



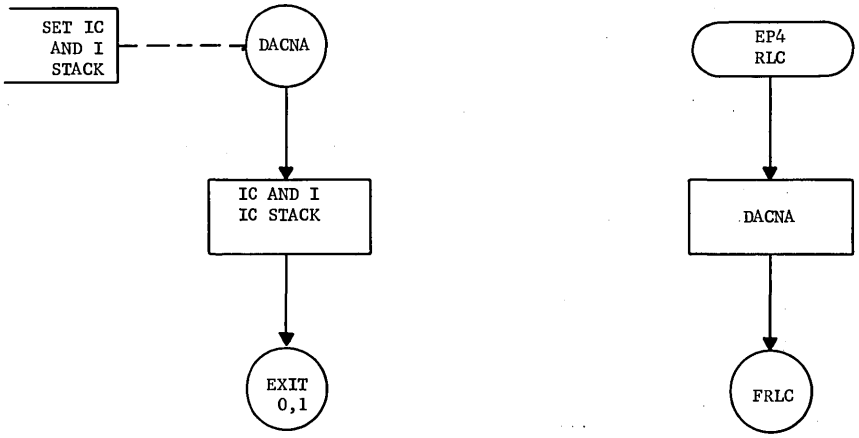
FRLC (EP3)
.MDISP

FORCED RELINQUISH



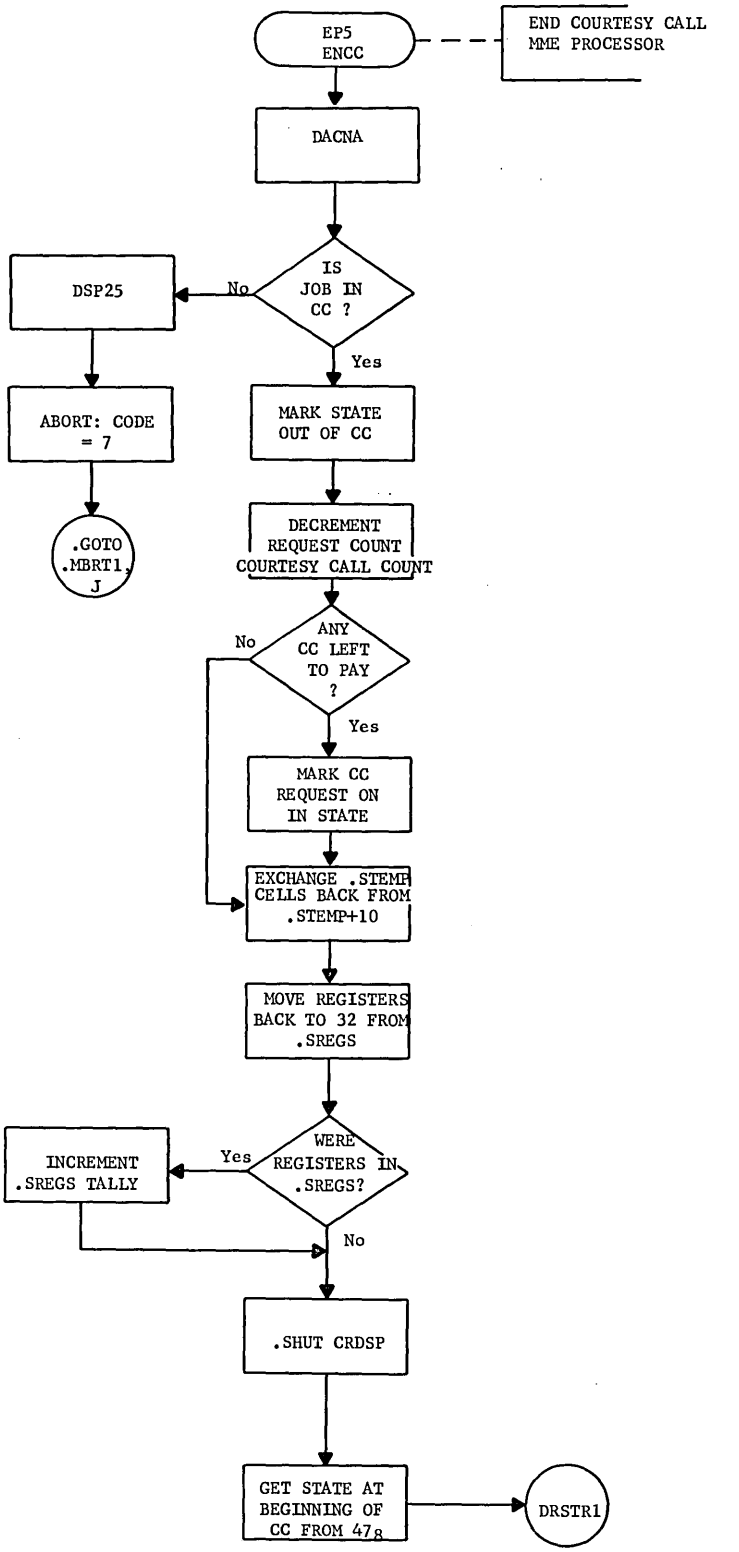
RLC (EP4)
.MDISP

RELINQUISH CONTROL



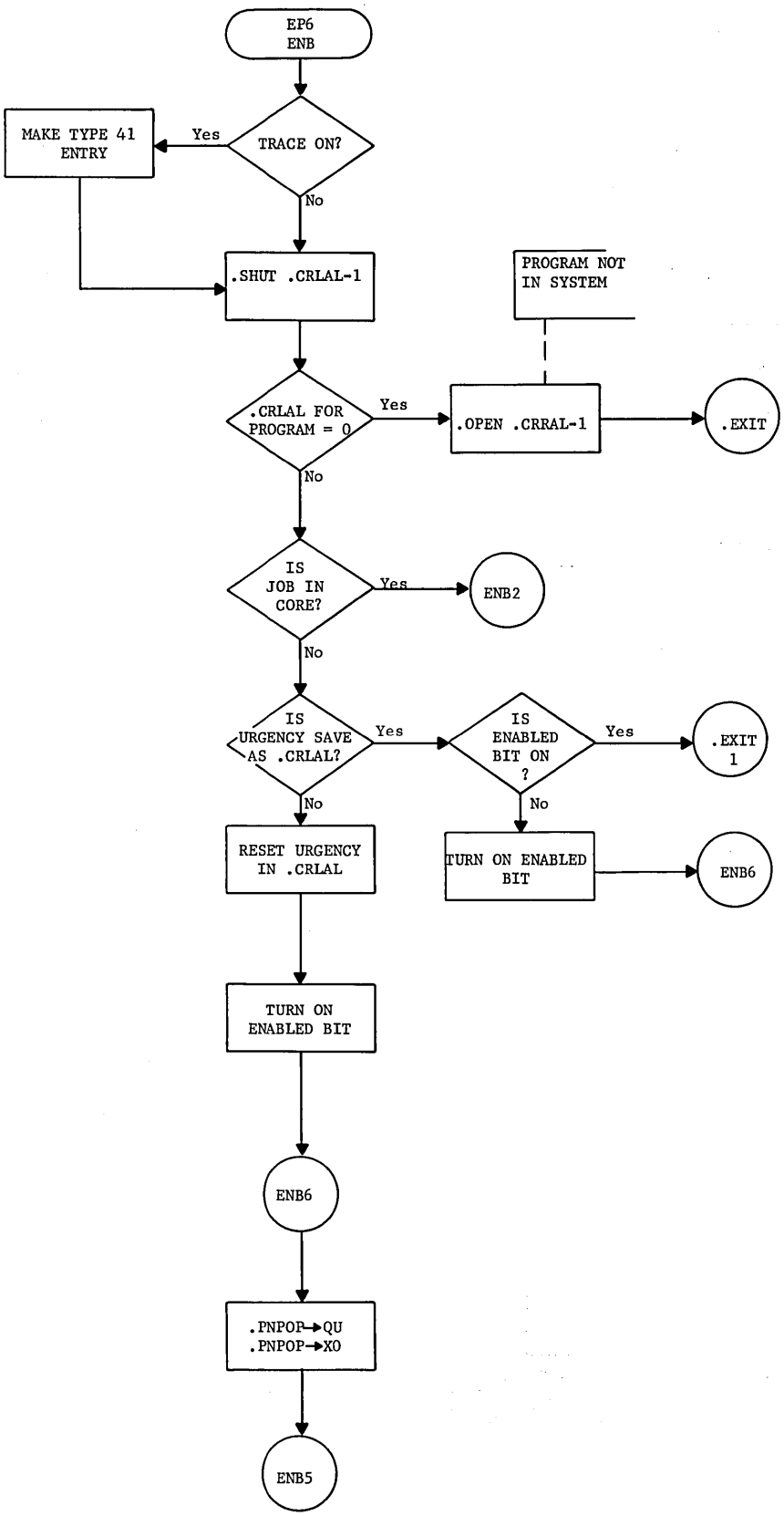
ENCC (EP5)
.MDISP

END COURTESY CALL

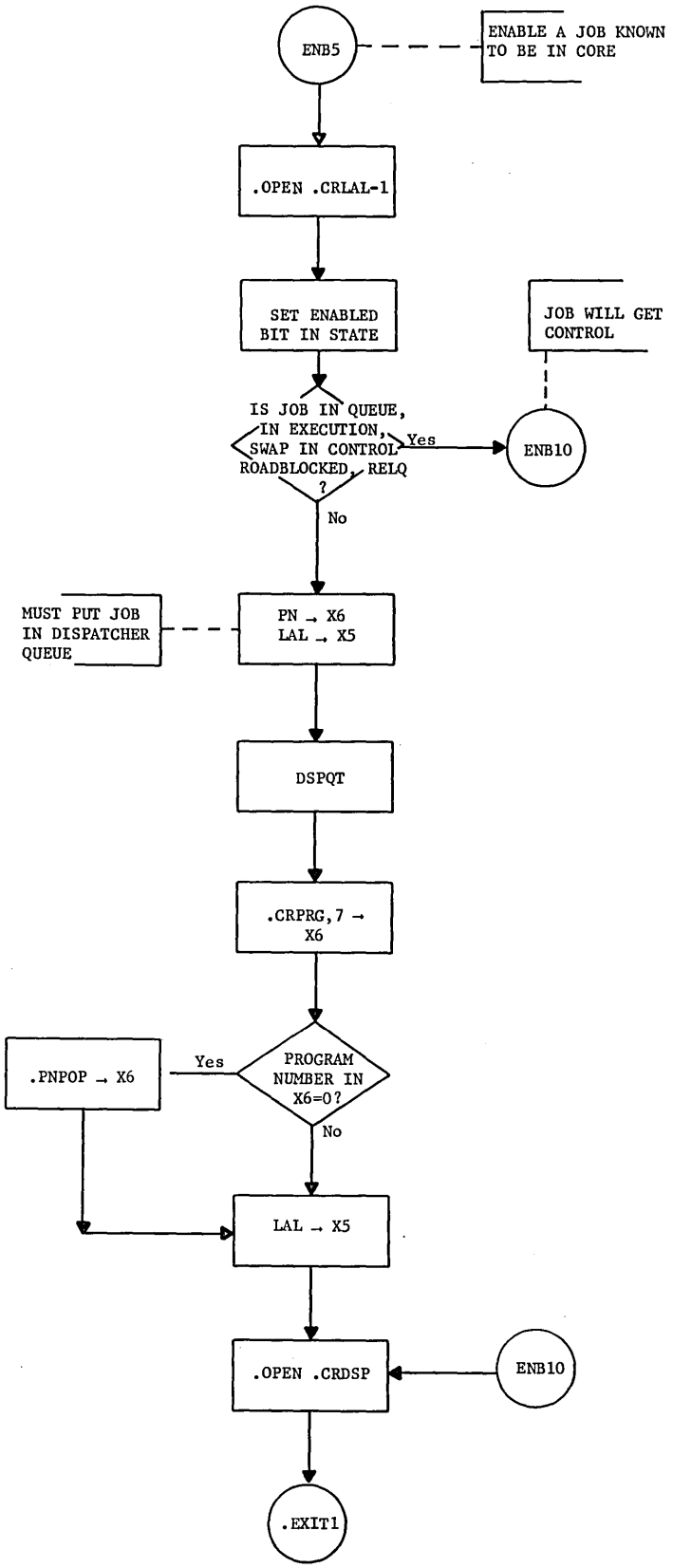


ENB (EP6)
.MDISP

ENABLE PROGRAM

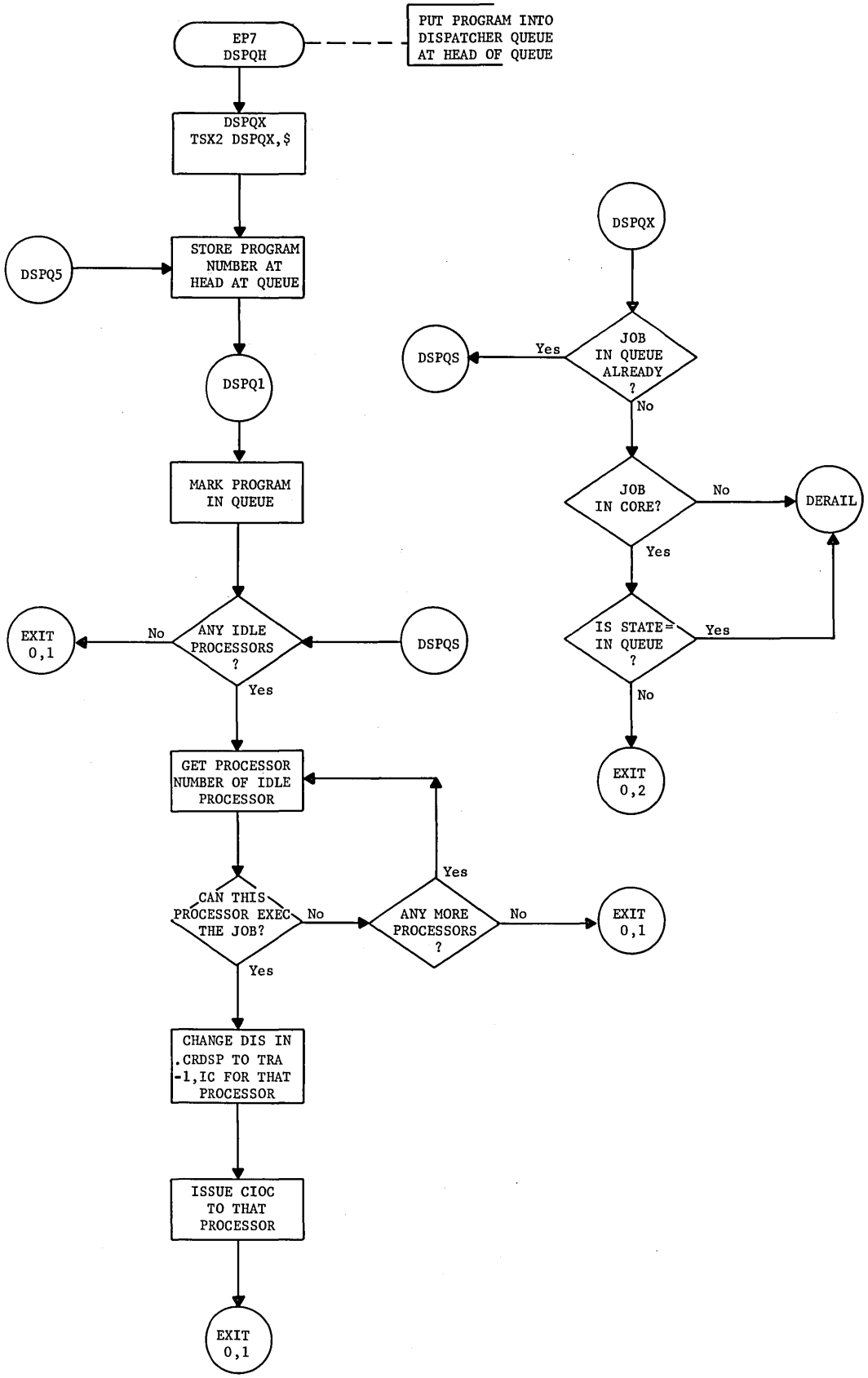


END (EP6)
.MDISP

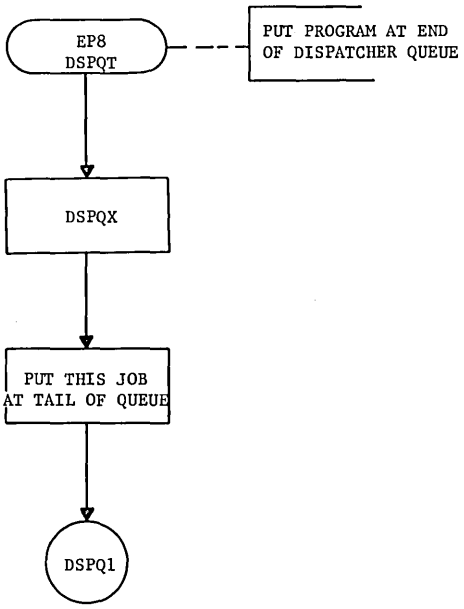


DSPQH (EP7)
.MDISP

PROGRAM NUMBER AT FRONT OF QUEUE

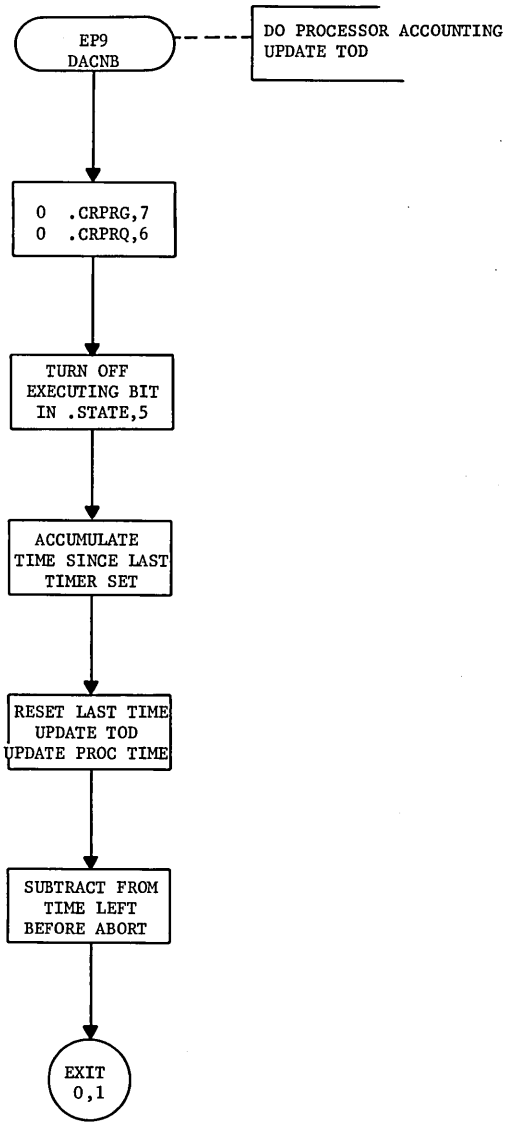


PROGRAM NUMBER AT END OF QUEUE

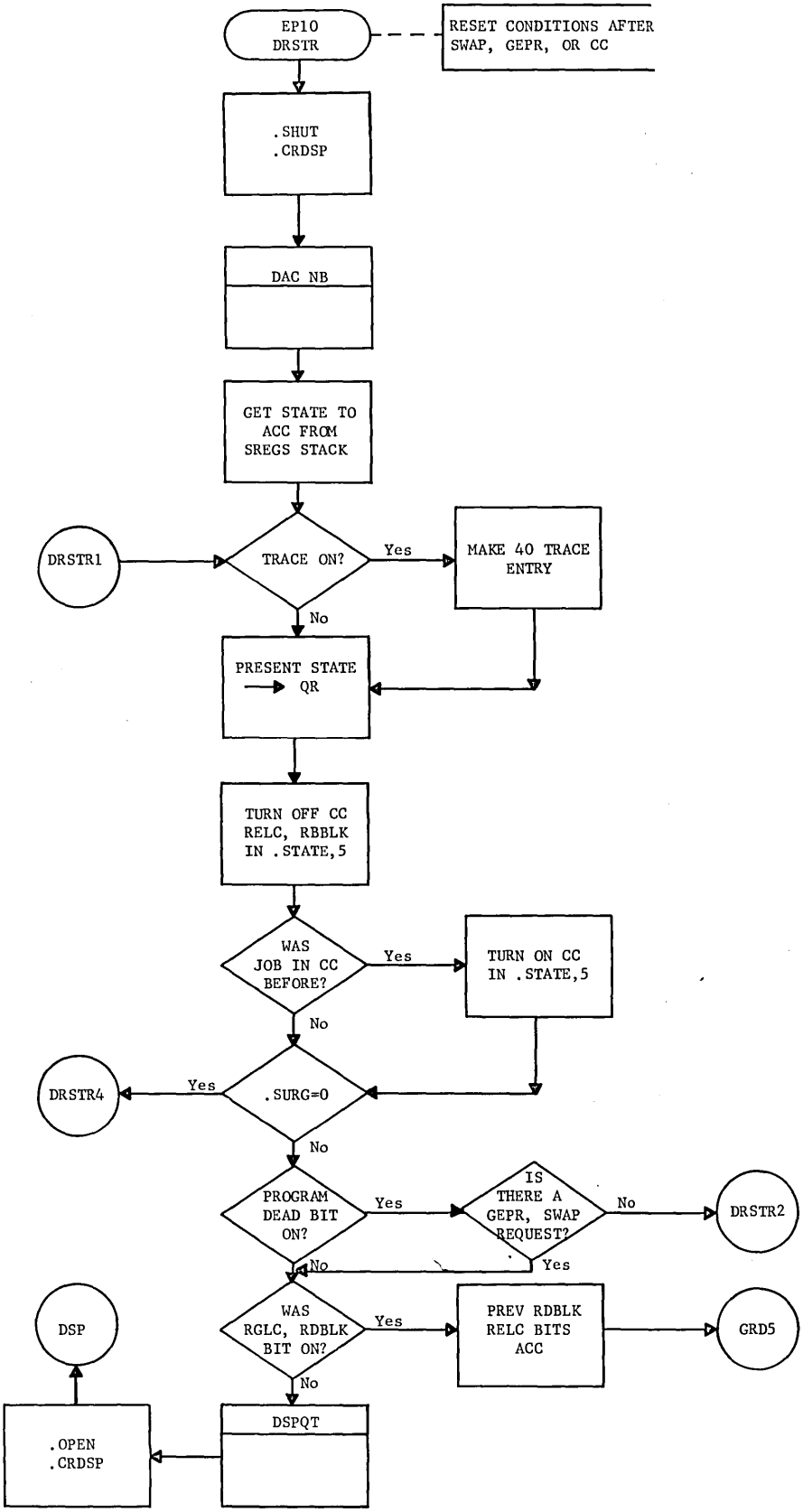


DACNB (EP9)
.MDISP

ACCUMULATE PROCESSOR TIME

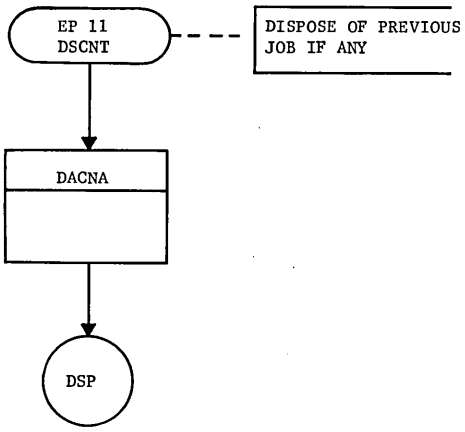


RESTORE STATE AFTER SWAP, MOVE, GEPR

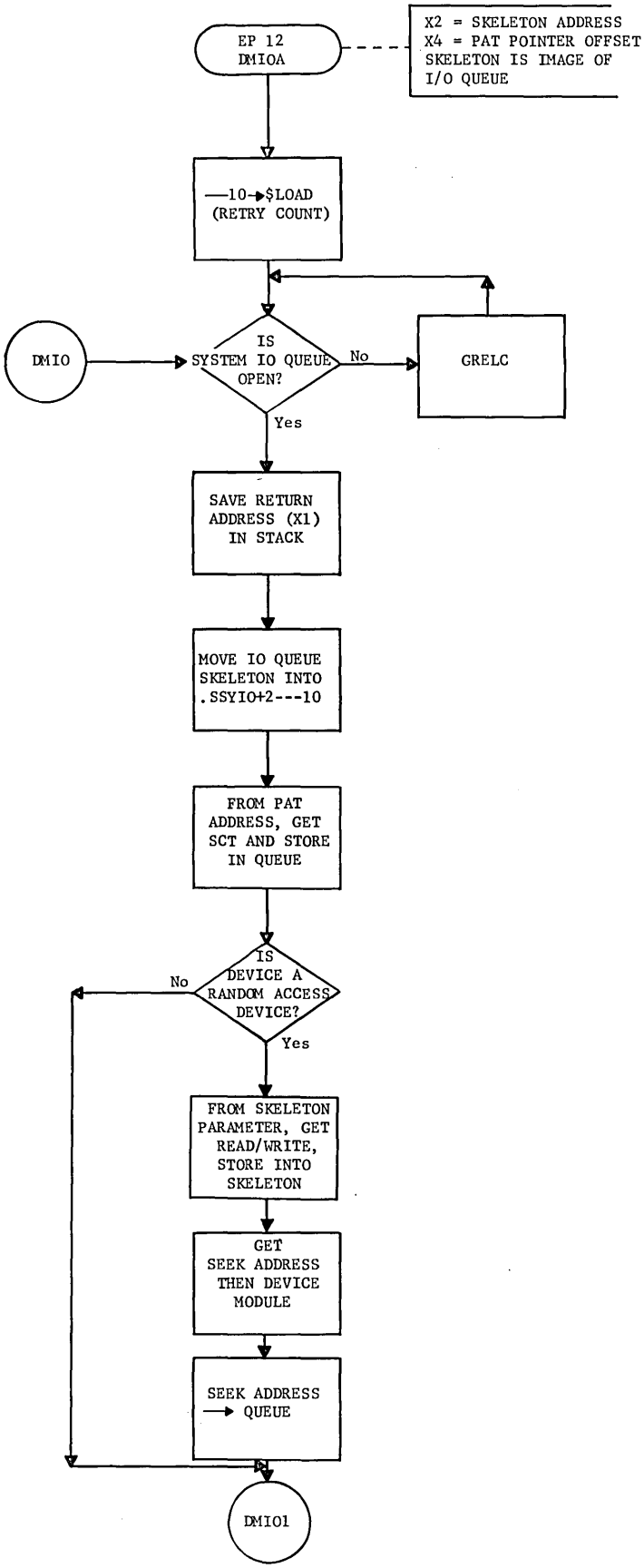


DSCNT (EP11)
.MDISP

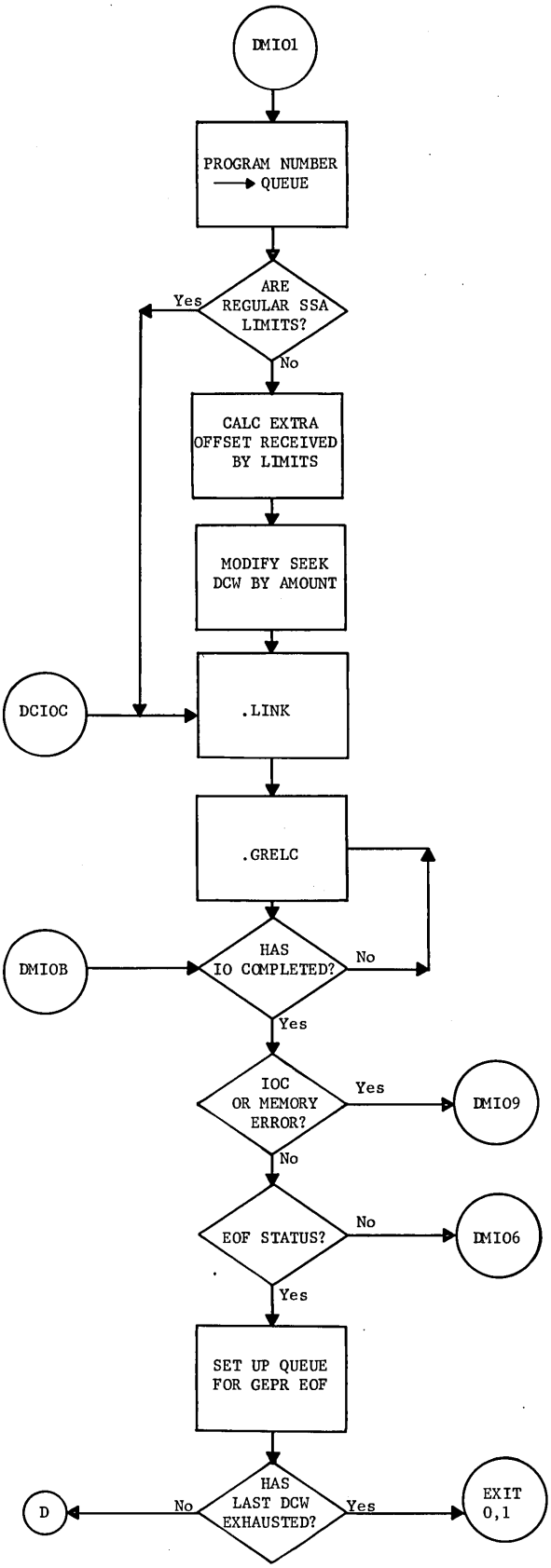
RELINQUISH CONTROL UNTIL PROGRAM ENABLED

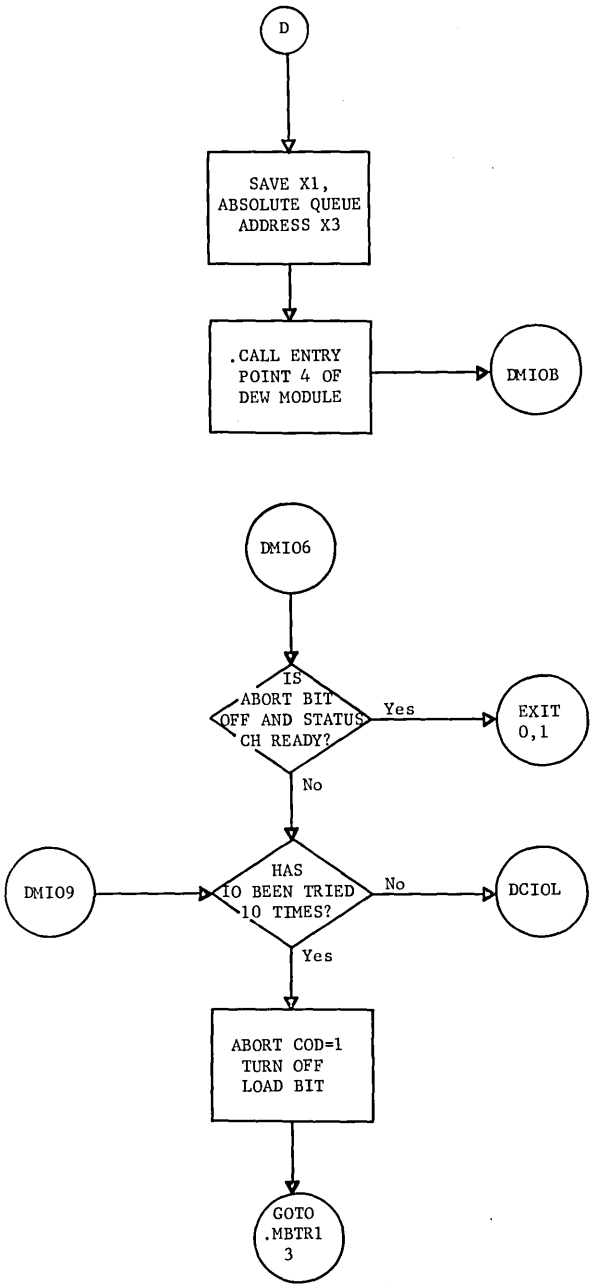


DO DISC I/O USING SYSTEM I/O QUEUE

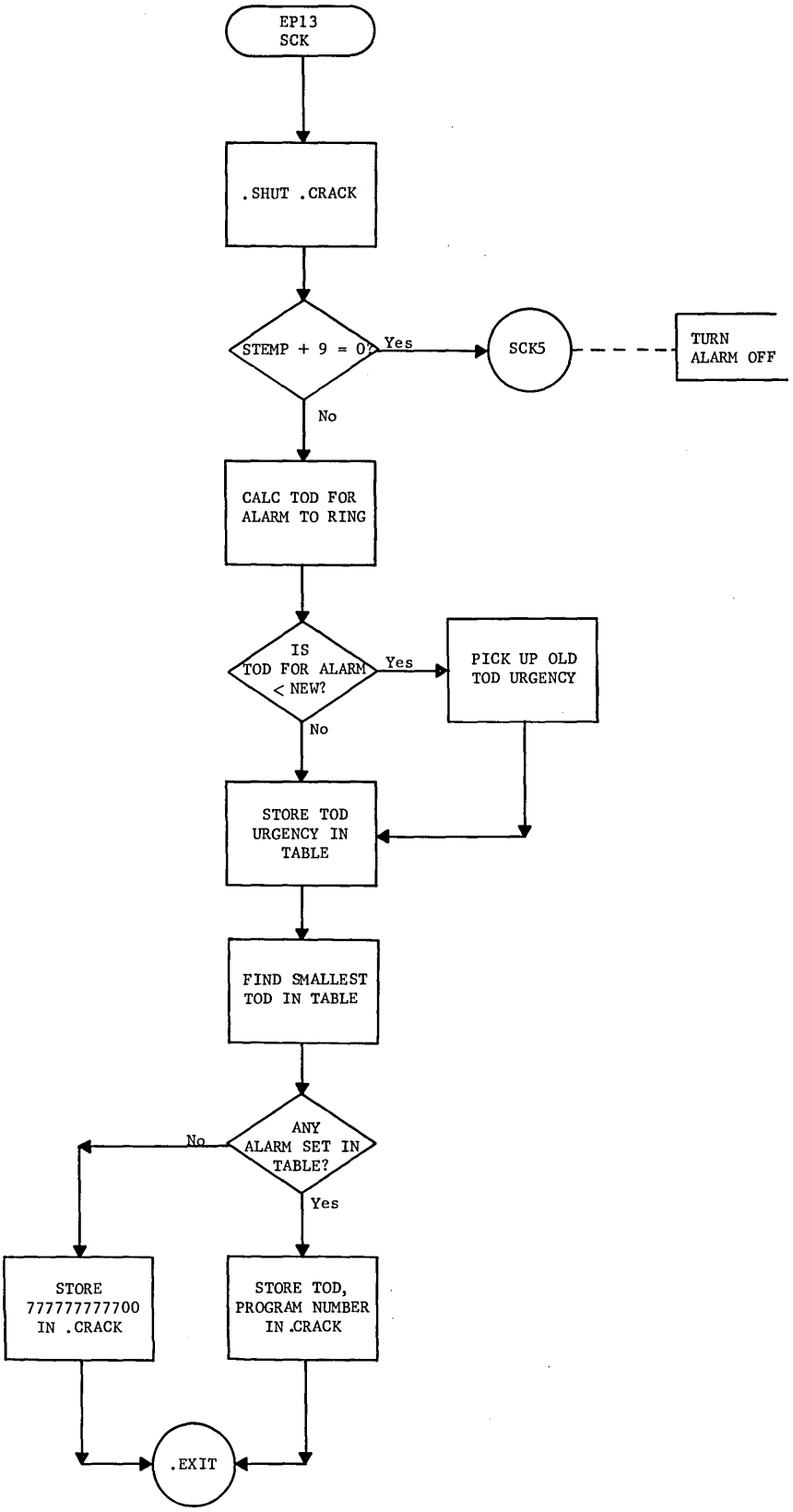


DMIOA (EP12)
.MDISP

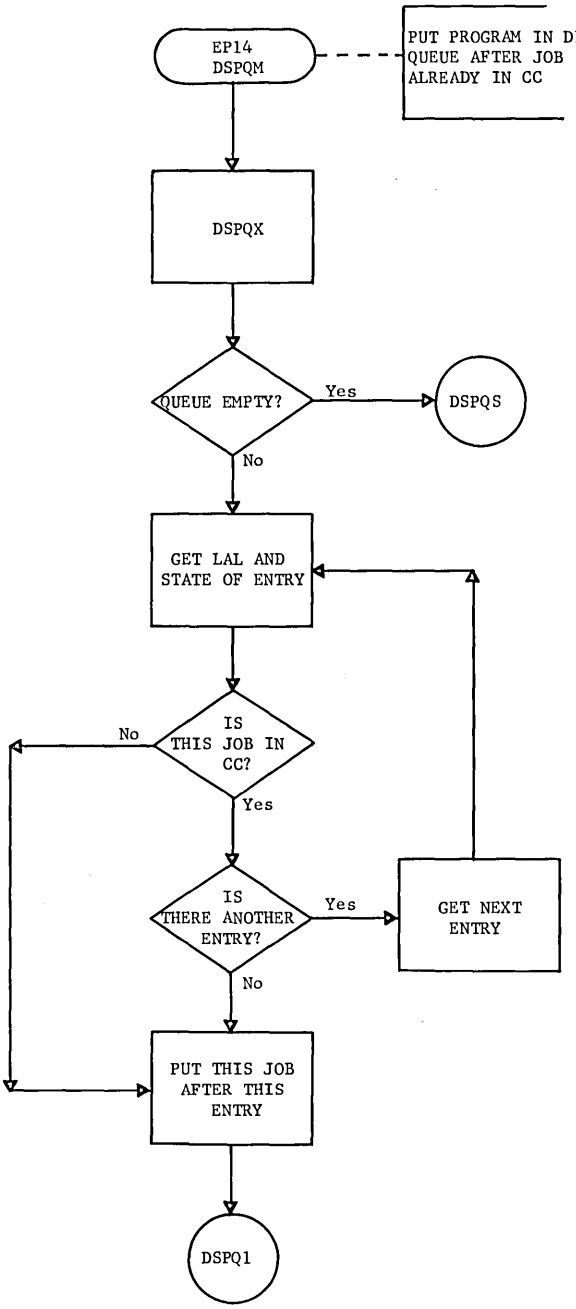



D
SAVE X1, ABSOLUTE QUEUE ADDRESS X3
.CALL ENTRY POINT 4 OF DEW MODULE
DMIOB
DMIO6
IS ABORT BIT OFF AND STATUS CH READY?
Yes
EXIT 0,1
No
DMIO9
HAS IO BEEN TRIED 10 TIMES?
No
DCIOL
Yes
ABORT COD=1 TURN OFF LOAD BIT
GOTO .MBTR1 3

SET ALARM

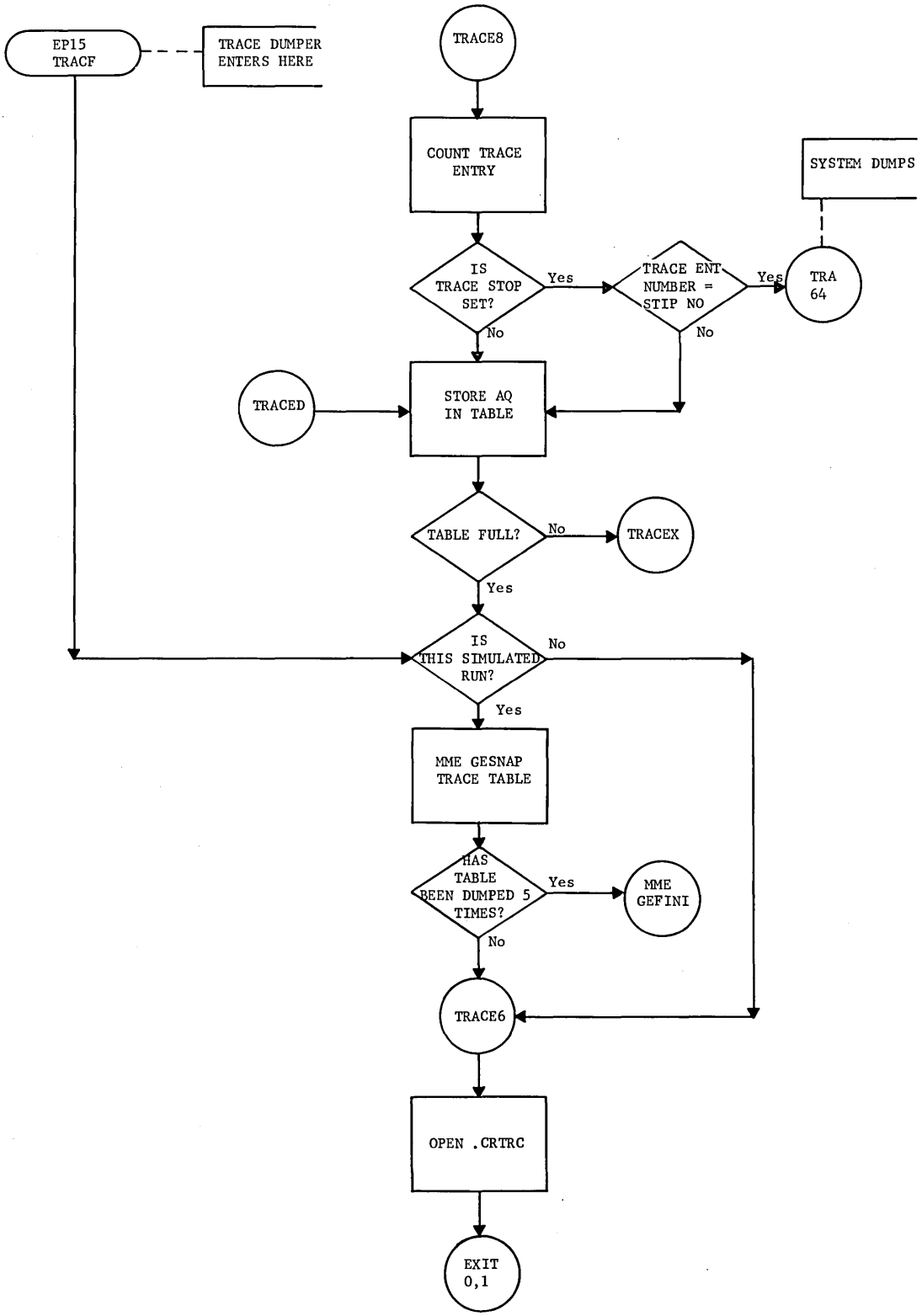

EP13
SCK
.SHUT .CRACK
STEMP + 9 = 0?
Yes
SCK5
TURN
ALARM OFF
No
CALC TOD FOR
ALARM TO RING
IS
TOD FOR ALARM
< NEW?
Yes
PICK UP OLD
TOD URGENCY
No
STORE TOD
URGENCY IN
TABLE
FIND SMALLEST
TOD IN TABLE
ANY
ALARM SET IN
TABLE?
No
Yes
STORE
777777777700
IN .CRACK
STORE TOD,
PROGRAM NUMBER
IN .CRACK
.EXIT

PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT



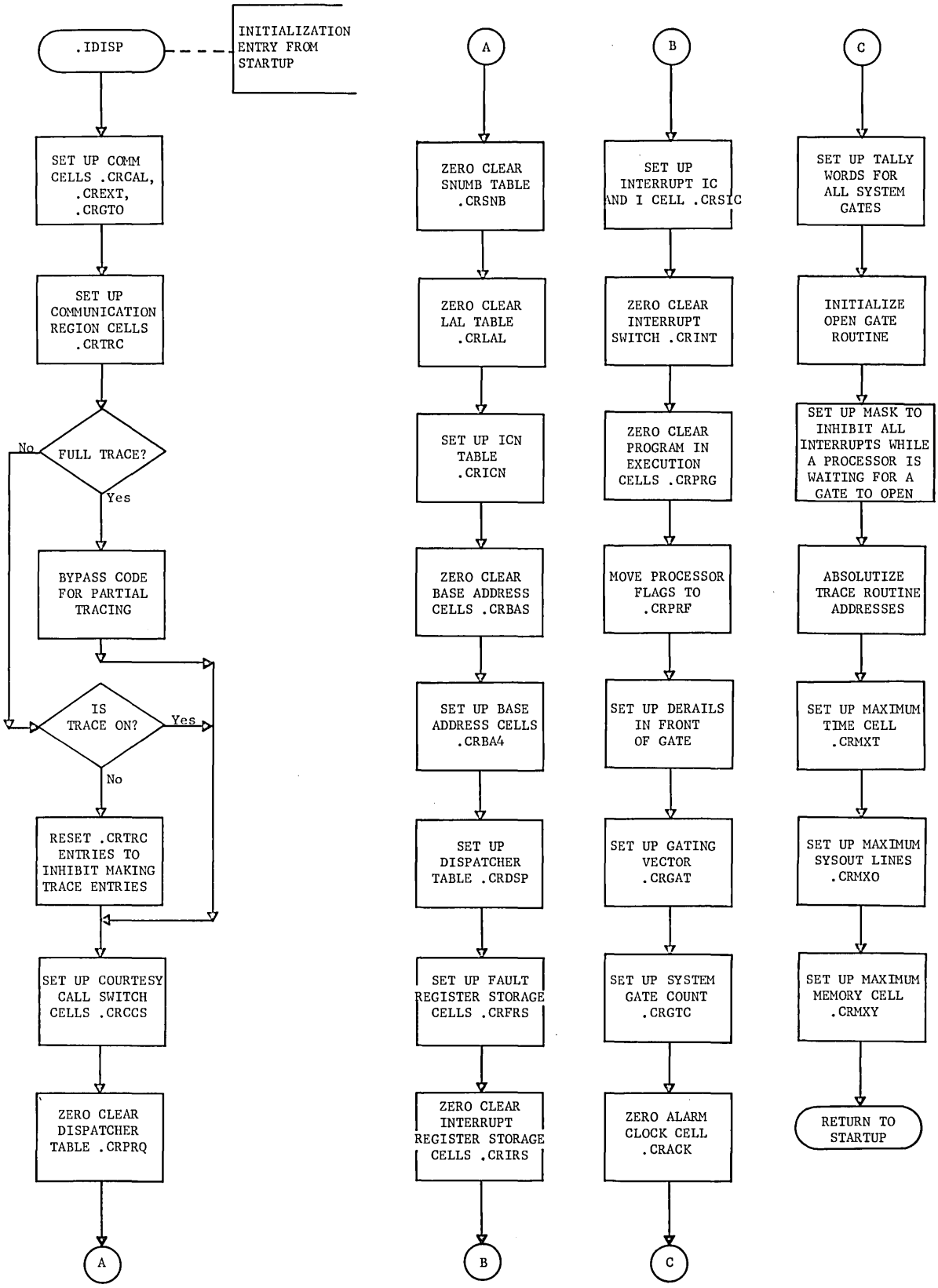
TRACF (EP15)
.MDISP

SAVE TRACE BUFFER AND WRITE OUT



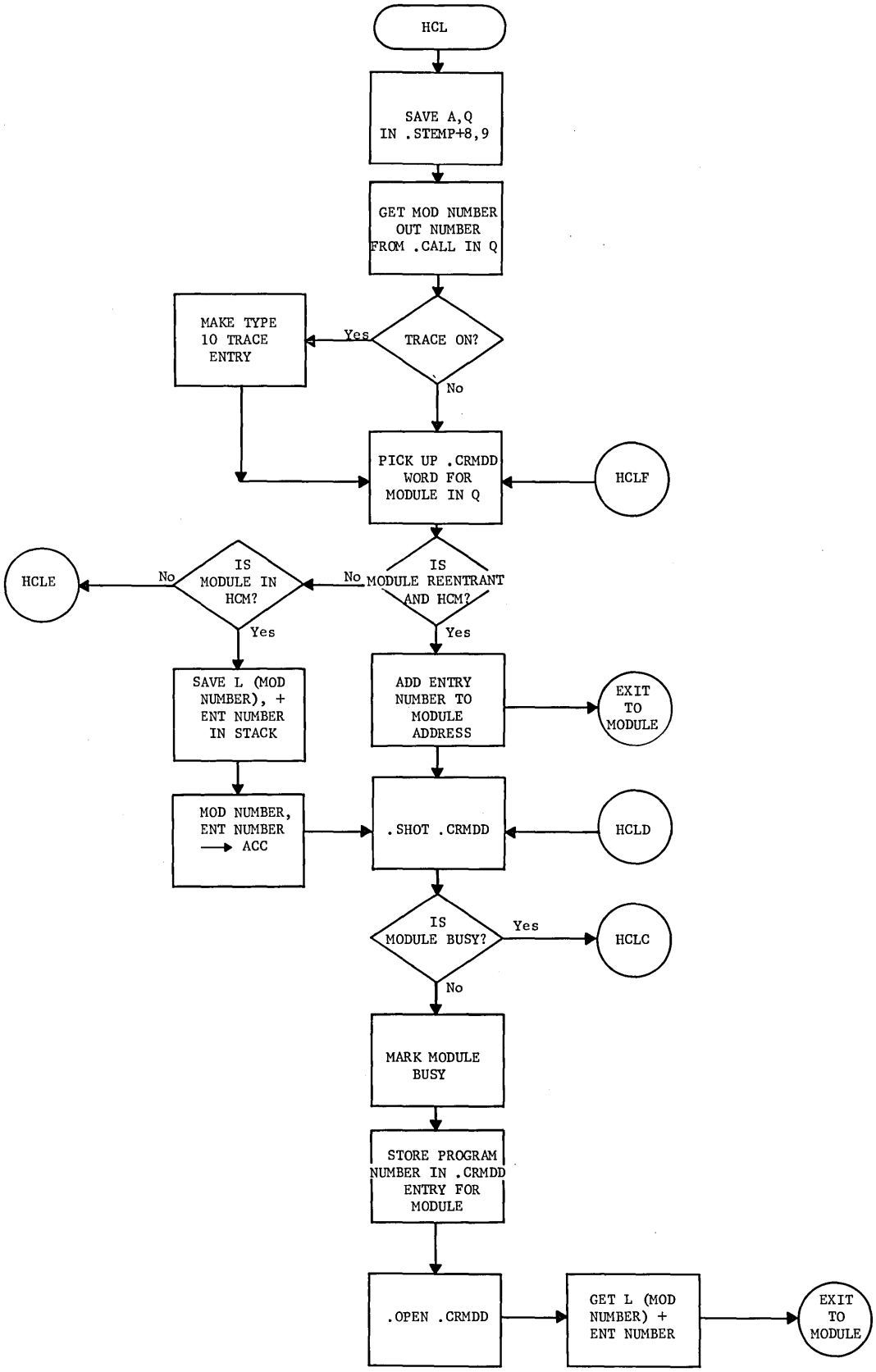
.IDISP
.MDISP

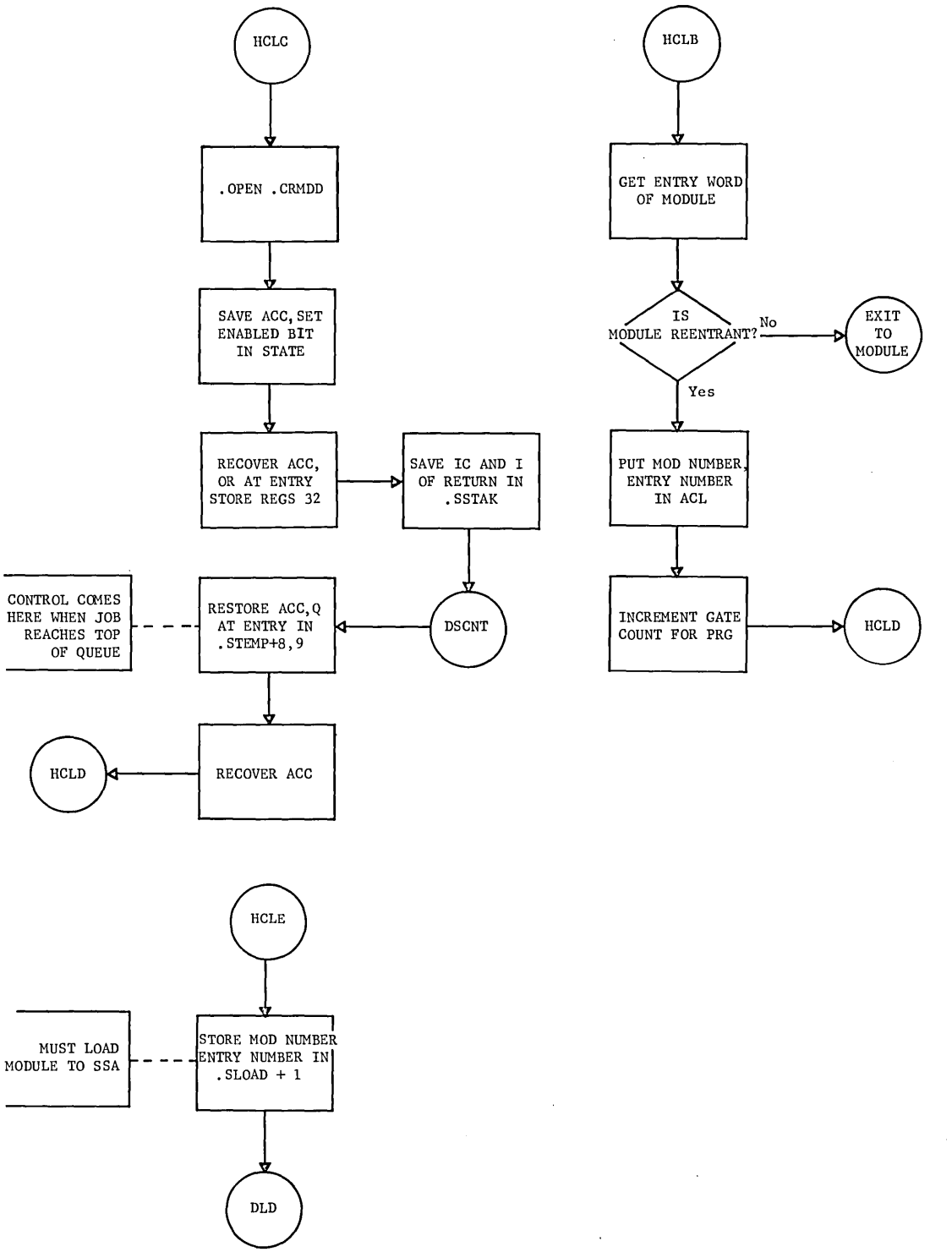
INITIALIZATION



HCL
.MDISP

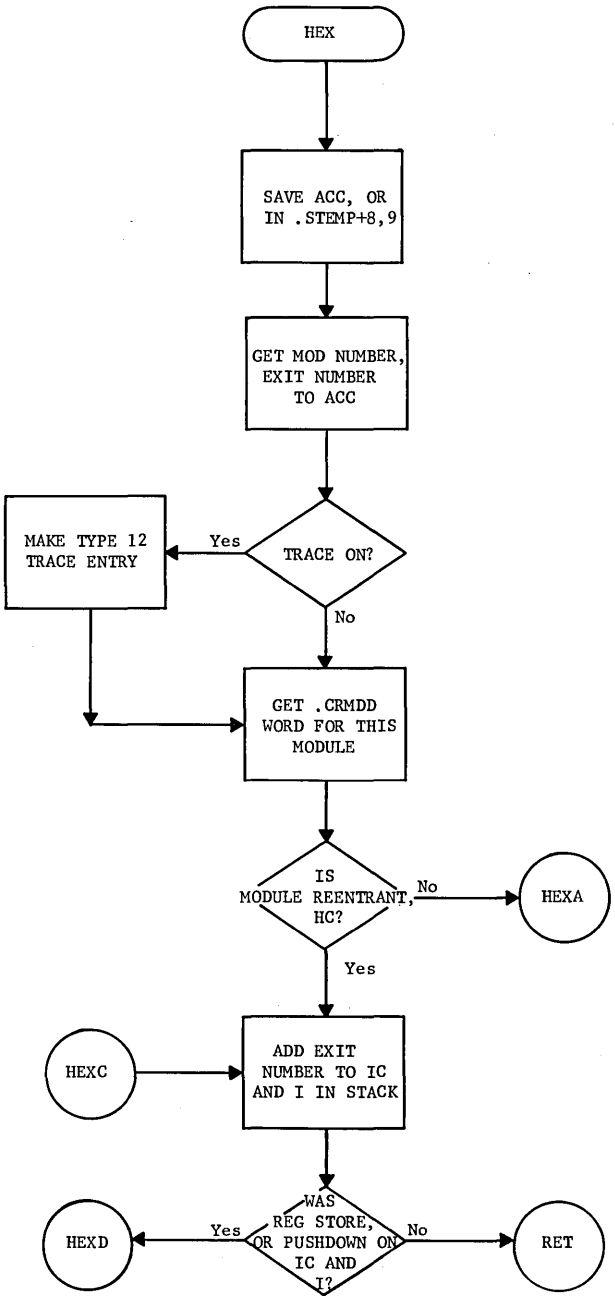
.CALL MACRO

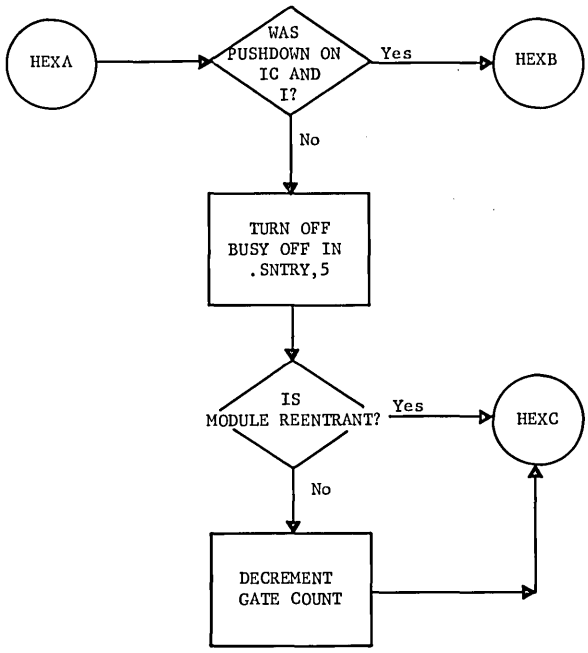


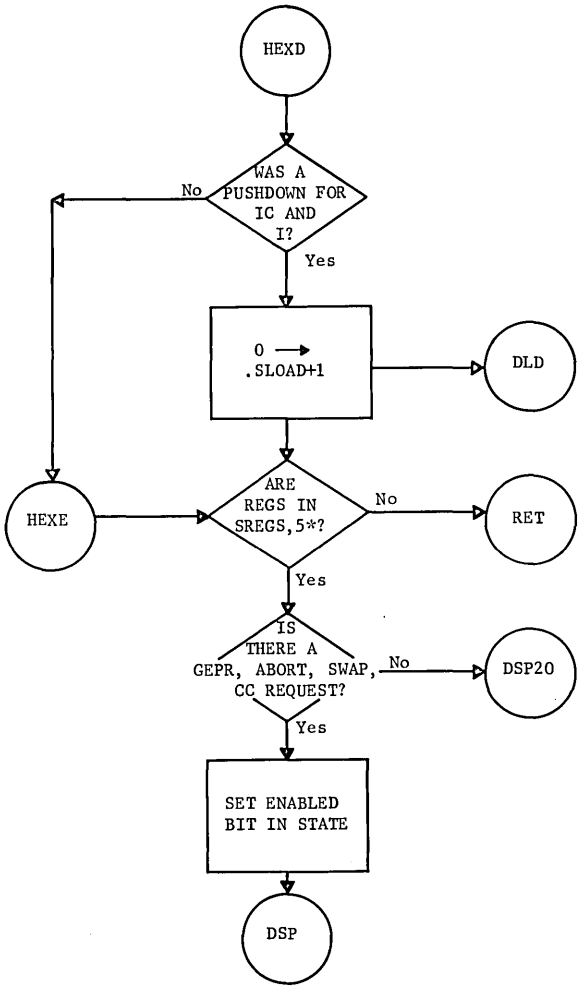

HCLC
.OPEN .CRMDD
SAVE ACC, SET ENABLED BIT IN STATE
RECOVER ACC, OR AT ENTRY STORE REGS 32
SAVE IC AND I OF RETURN IN .SSTAK
DSCNT
CONTROL COMES HERE WHEN JOB REACHES TOP OF QUEUE
RESTORE ACC, Q AT ENTRY IN .STEMP+8, 9
RECOVER ACC
HCLD
HCLB
GET ENTRY WORD OF MODULE
IS MODULE REENTRANT?
No
EXIT TO MODULE
Yes
PUT MOD NUMBER, ENTRY NUMBER IN ACL
INCREMENT GATE COUNT FOR PRG
HCLD
HCLE
MUST LOAD MODULE TO SSA
STORE MOD NUMBER ENTRY NUMBER IN .SLOAD + 1
DLD

HEX
.MDISP

.EXIT MACRO

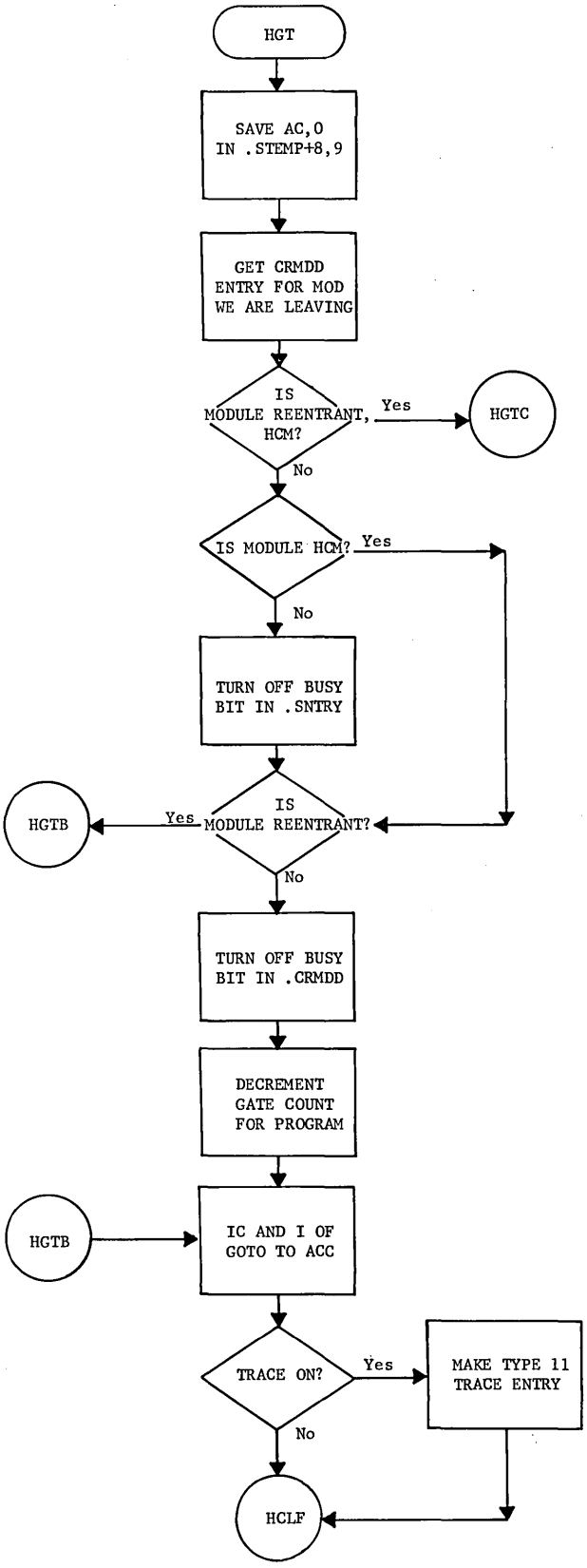


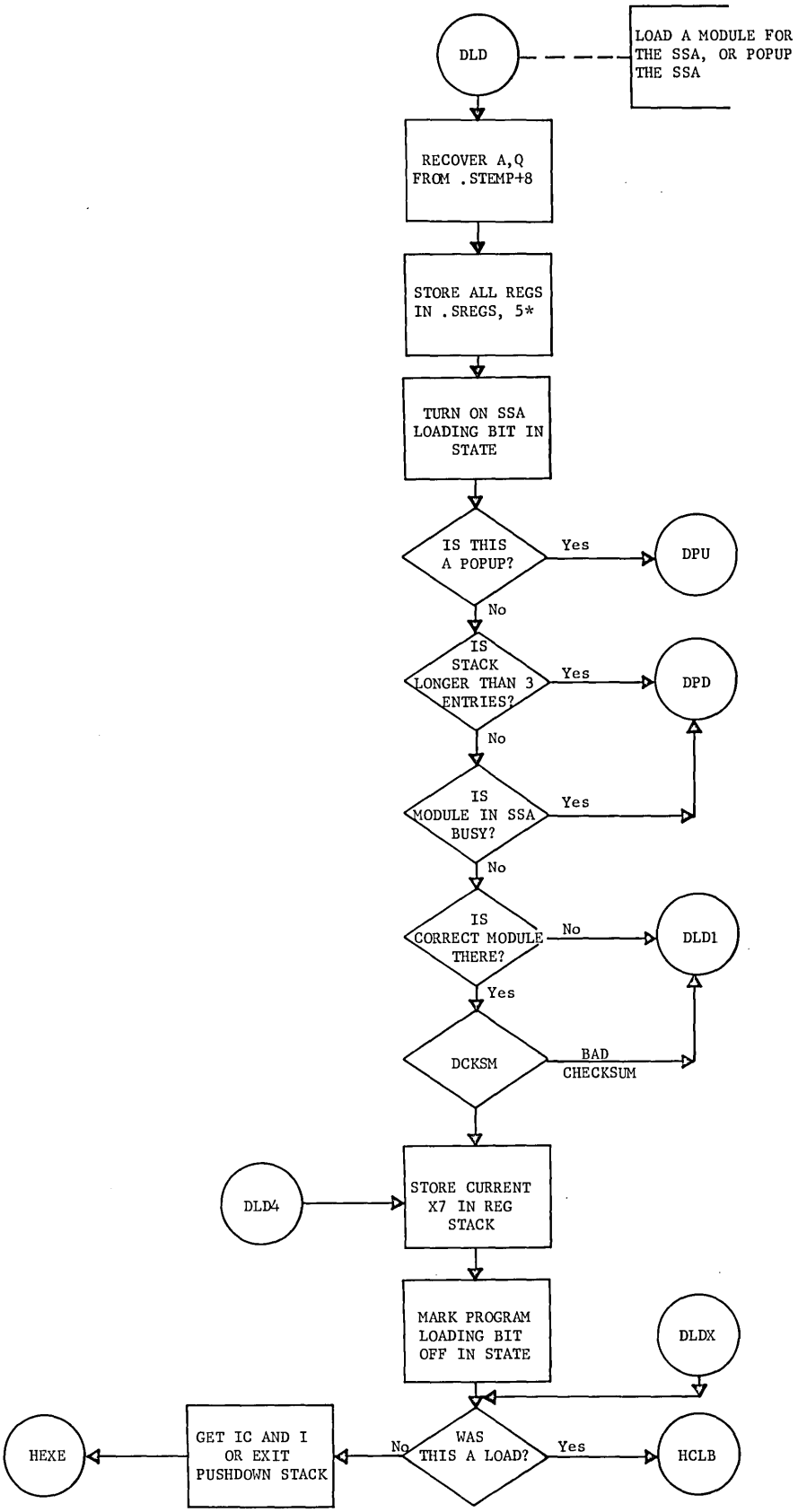

HEXA
WAS PUSHDOWN ON IC AND I?
Yes
HEXB
No
TURN OFF BUSY OFF IN .SNTRY,5
IS MODULE REENTRANT?
Yes
HEXC
No
DECREMENT GATE COUNT


HEXD
WAS A PUSHDOWN FOR IC AND I?
No
Yes
0 → .SLOAD+1
DLD
HEXE
ARE REGS IN SREGS,5*?
No
RET
Yes
IS THERE A GEPR, ABORT, SWAP, CC REQUEST?
No
DSP20
Yes
SET ENABLED BIT IN STATE
DSP

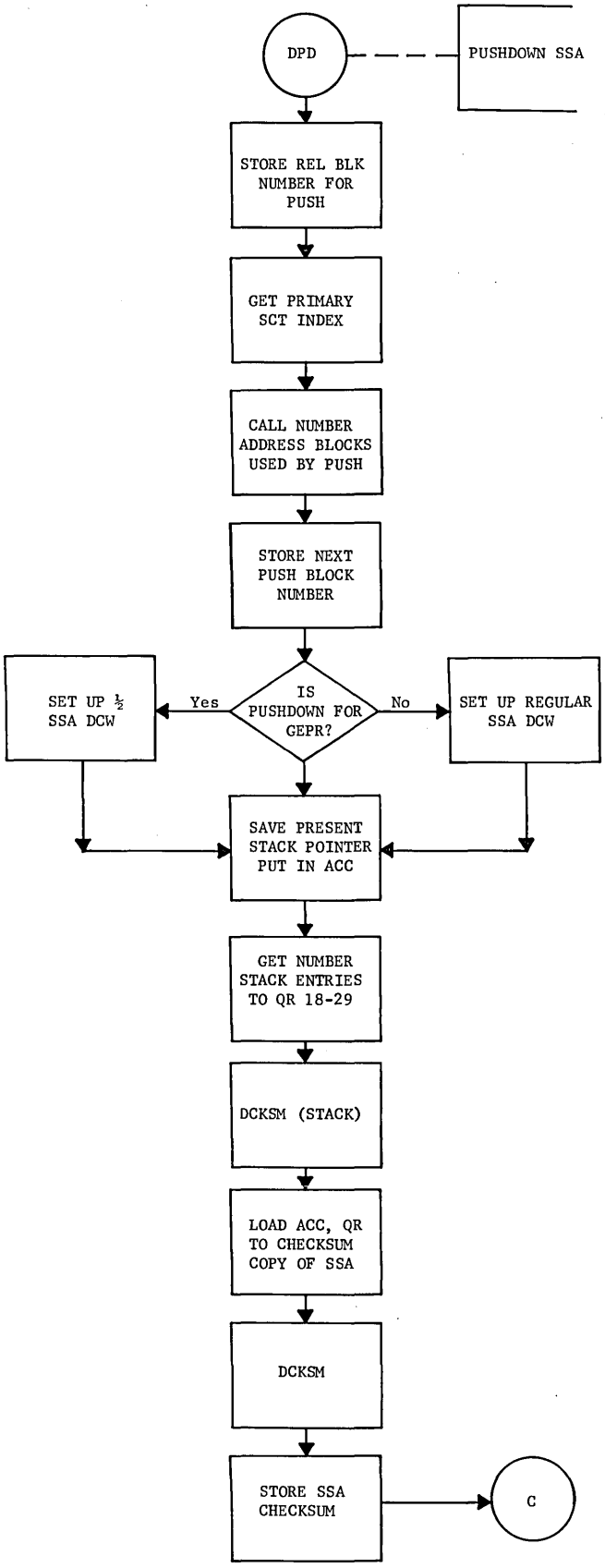
HGT
.MDISP

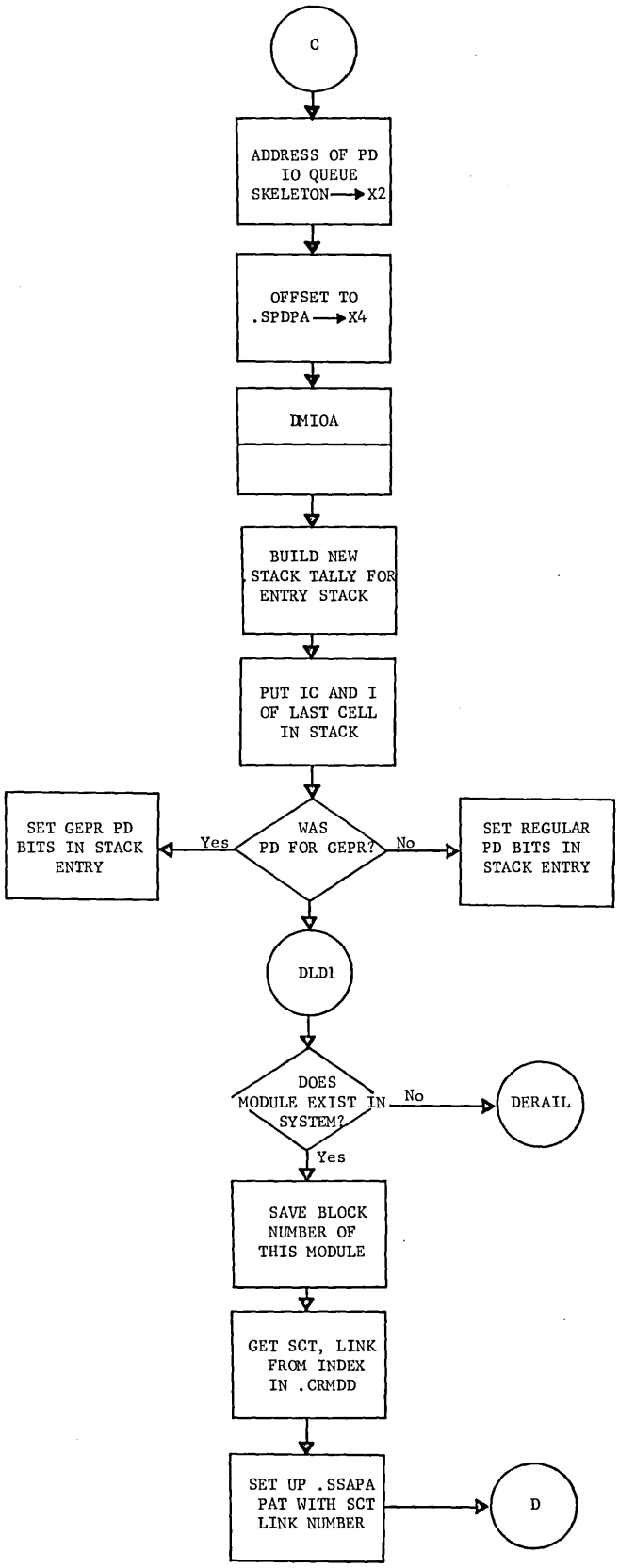
.GOTO MACRO



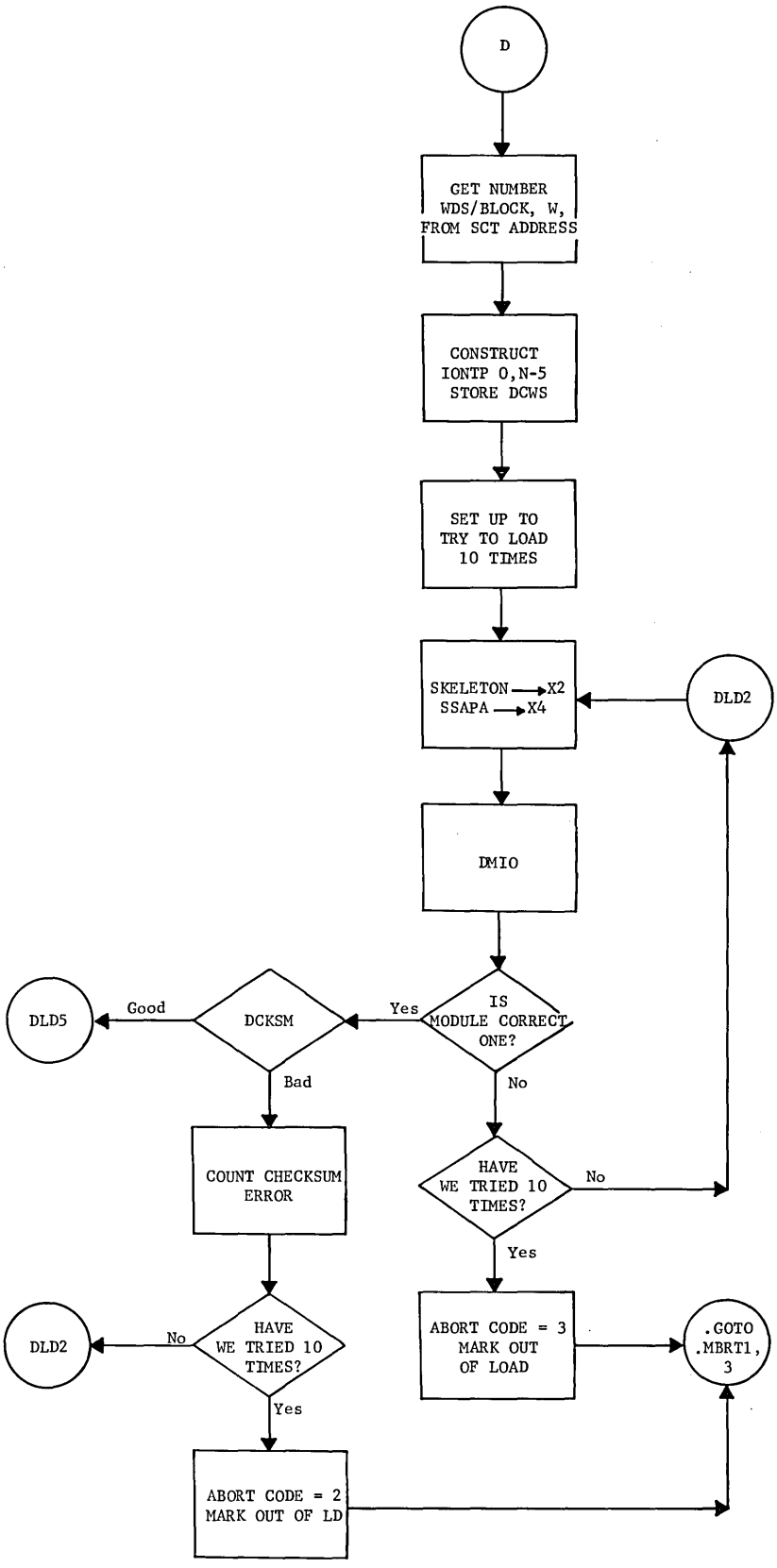

DLD
LOAD A MODULE FOR THE SSA, OR POPUP THE SSA
RECOVER A,Q FROM .STEMP+8
STORE ALL REGS IN .SREGS, 5*
TURN ON SSA LOADING BIT IN STATE
IS THIS A POPUP?
Yes
DPU
No
IS STACK LONGER THAN 3 ENTRIES?
Yes
DPD
No
IS MODULE IN SSA BUSY?
Yes
No
IS CORRECT MODULE THERE?
No
DLD1
Yes
DCKSM
BAD CHECKSUM
DLD4
STORE CURRENT X7 IN REG STACK
MARK PROGRAM LOADING BIT OFF IN STATE
DLDX
HEXE
GET IC AND I OR EXIT PUSHDOWN STACK
No
WAS THIS A LOAD?
Yes
HCLB

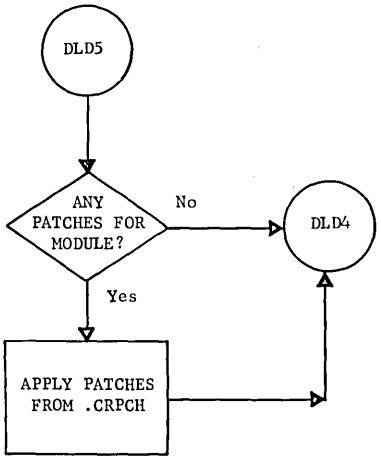
HGT
.MDISP


DPD
PUSHDOWN SSA
STORE REL BLK NUMBER FOR PUSH
GET PRIMARY SCT INDEX
CALL NUMBER ADDRESS BLOCKS USED BY PUSH
STORE NEXT PUSH BLOCK NUMBER
IS PUSHDOWN FOR GEPR?
Yes
No
SET UP ½ SSA DCW
SET UP REGULAR SSA DCW
SAVE PRESENT STACK POINTER PUT IN ACC
GET NUMBER STACK ENTRIES TO QR 18-29
DCKSM (STACK)
LOAD ACC, QR TO CHECKSUM COPY OF SSA
DCKSM
STORE SSA CHECKSUM
C

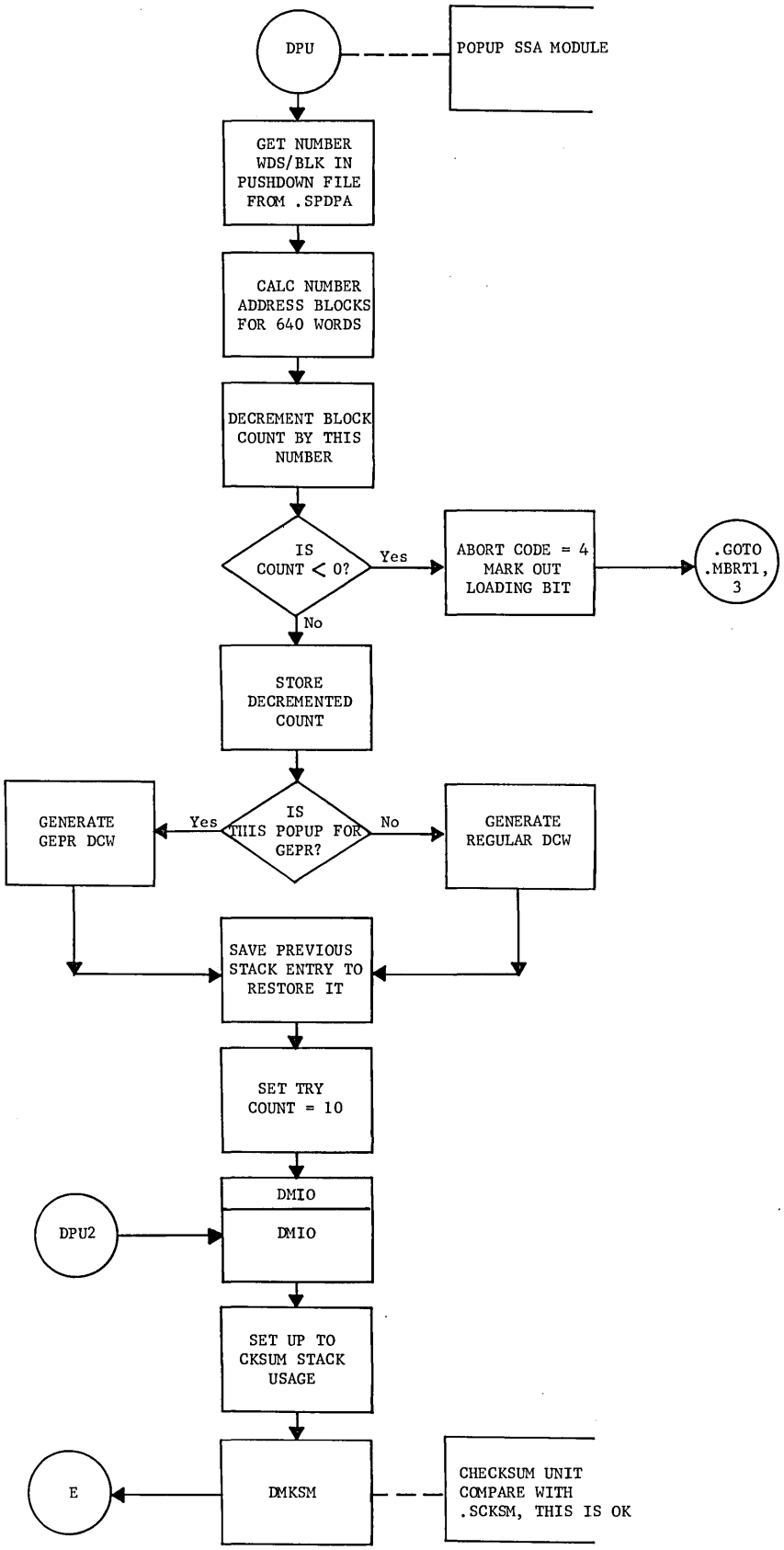

C
ADDRESS OF PD IO QUEUE SKELETON → X2
OFFSET TO .SPDPA → X4
DMIOA
BUILD NEW STACK TALLY FOR ENTRY STACK
PUT IC AND I OF LAST CELL IN STACK
WAS PD FOR GEPR?
Yes
SET GEPR PD BITS IN STACK ENTRY
No
SET REGULAR PD BITS IN STACK ENTRY
DLD1
DOES MODULE EXIST IN SYSTEM?
No
DERAIL
Yes
SAVE BLOCK NUMBER OF THIS MODULE
GET SCT, LINK FROM INDEX IN .CRMDD
SET UP .SSAPA PAT WITH SCT LINK NUMBER
D

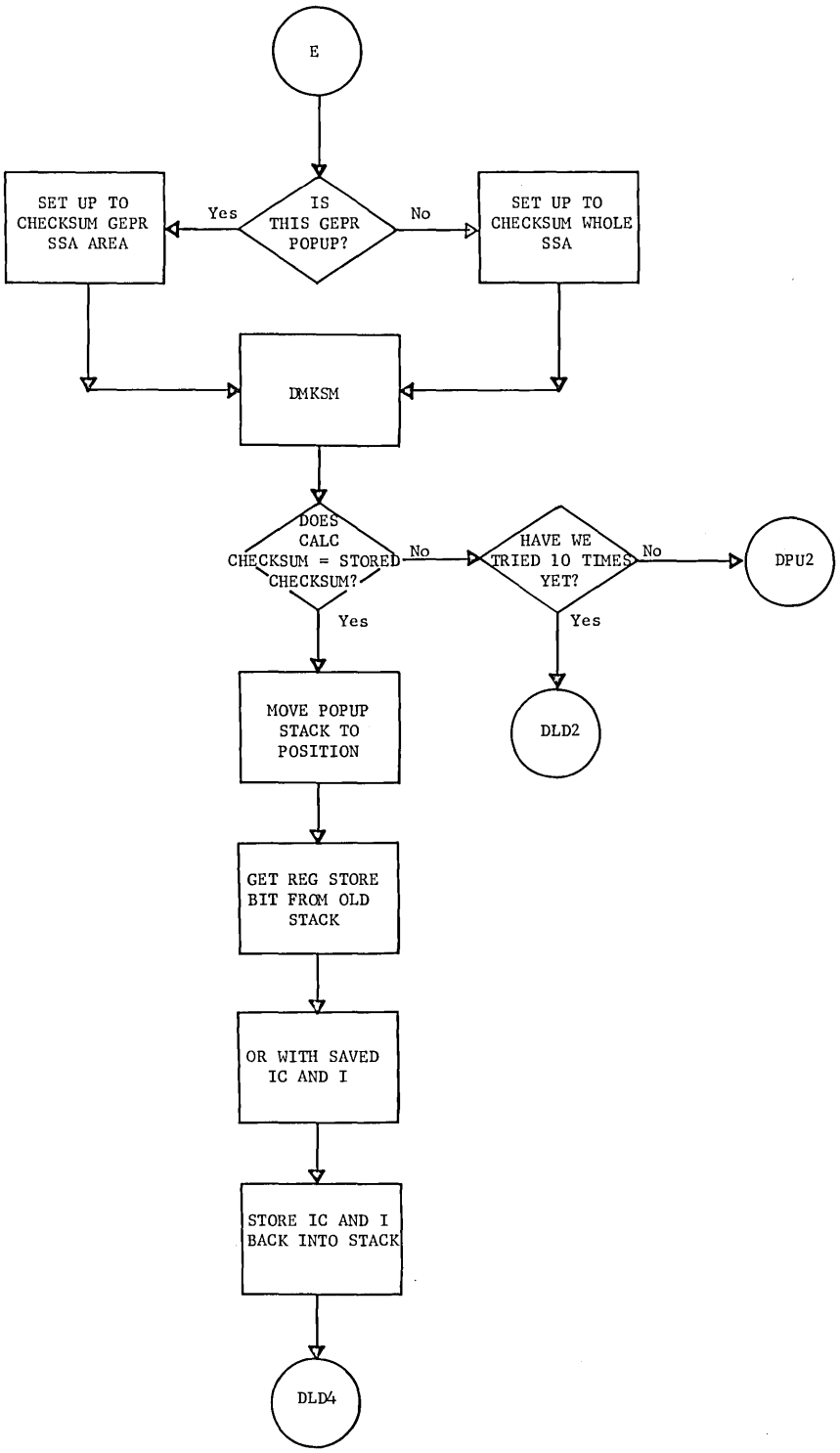
HGT
.MDISP


D
GET NUMBER WDS/BLOCK, W, FROM SCT ADDRESS
CONSTRUCT IONTP 0,N-5 STORE DCWS
SET UP TO TRY TO LOAD 10 TIMES
SKELETON → X2
SSAPA → X4
DLD2
DMIO
IS MODULE CORRECT ONE?
Yes
DCKSM
Good
DLD5
Bad
No
COUNT CHECKSUM ERROR
HAVE WE TRIED 10 TIMES?
No
Yes
ABORT CODE = 3 MARK OUT OF LOAD
.GOTO .MBRT1, 3
HAVE WE TRIED 10 TIMES?
No
DLD2
Yes
ABORT CODE = 2 MARK OUT OF LD

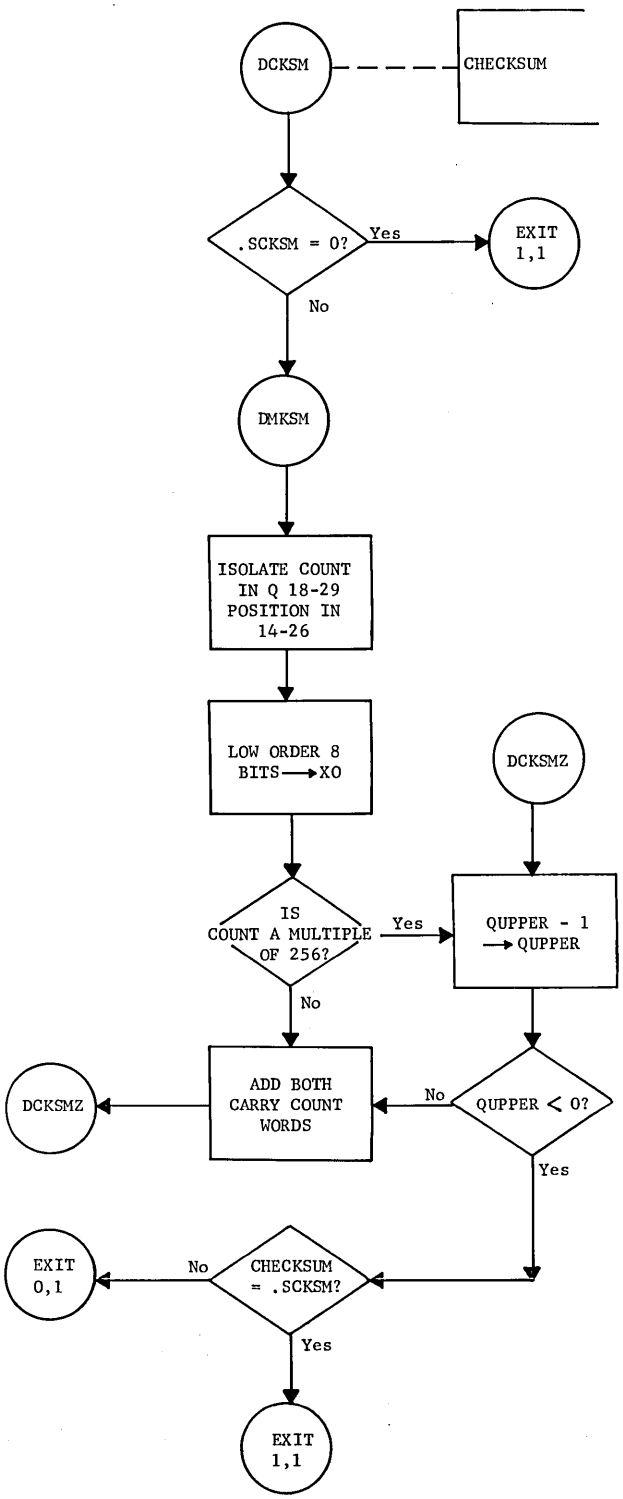

DLD5
ANY PATCHES FOR MODULE?
No
DLD4
Yes
APPLY PATCHES FROM .CRPCH

HGT
.MDISP

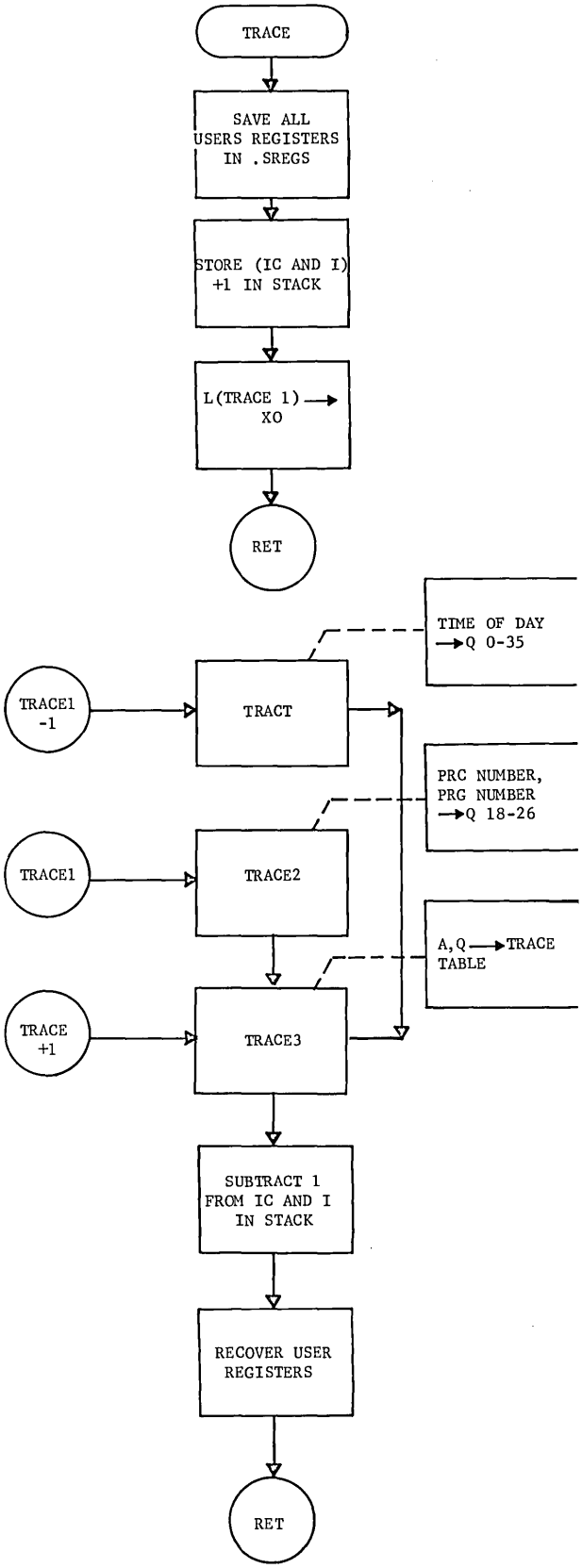

DPU
POPUP SSA MODULE
GET NUMBER WDS/BLK IN PUSHDOWN FILE FROM .SPDPA
CALC NUMBER ADDRESS BLOCKS FOR 640 WORDS
DECREMENT BLOCK COUNT BY THIS NUMBER
IS COUNT < 0?
Yes
ABORT CODE = 4 MARK OUT LOADING BIT
.GOTO .MBRT1, 3
No
STORE DECREMENTED COUNT
GENERATE GEPR DCW
Yes
IS THIS POPUP FOR GEPR?
No
GENERATE REGULAR DCW
SAVE PREVIOUS STACK ENTRY TO RESTORE IT
SET TRY COUNT = 10
DMIO
DPU2
DMIO
SET UP TO CKSUM STACK USAGE
E
DMKSM
CHECKSUM UNIT COMPARE WITH .SCKSM, THIS IS OK

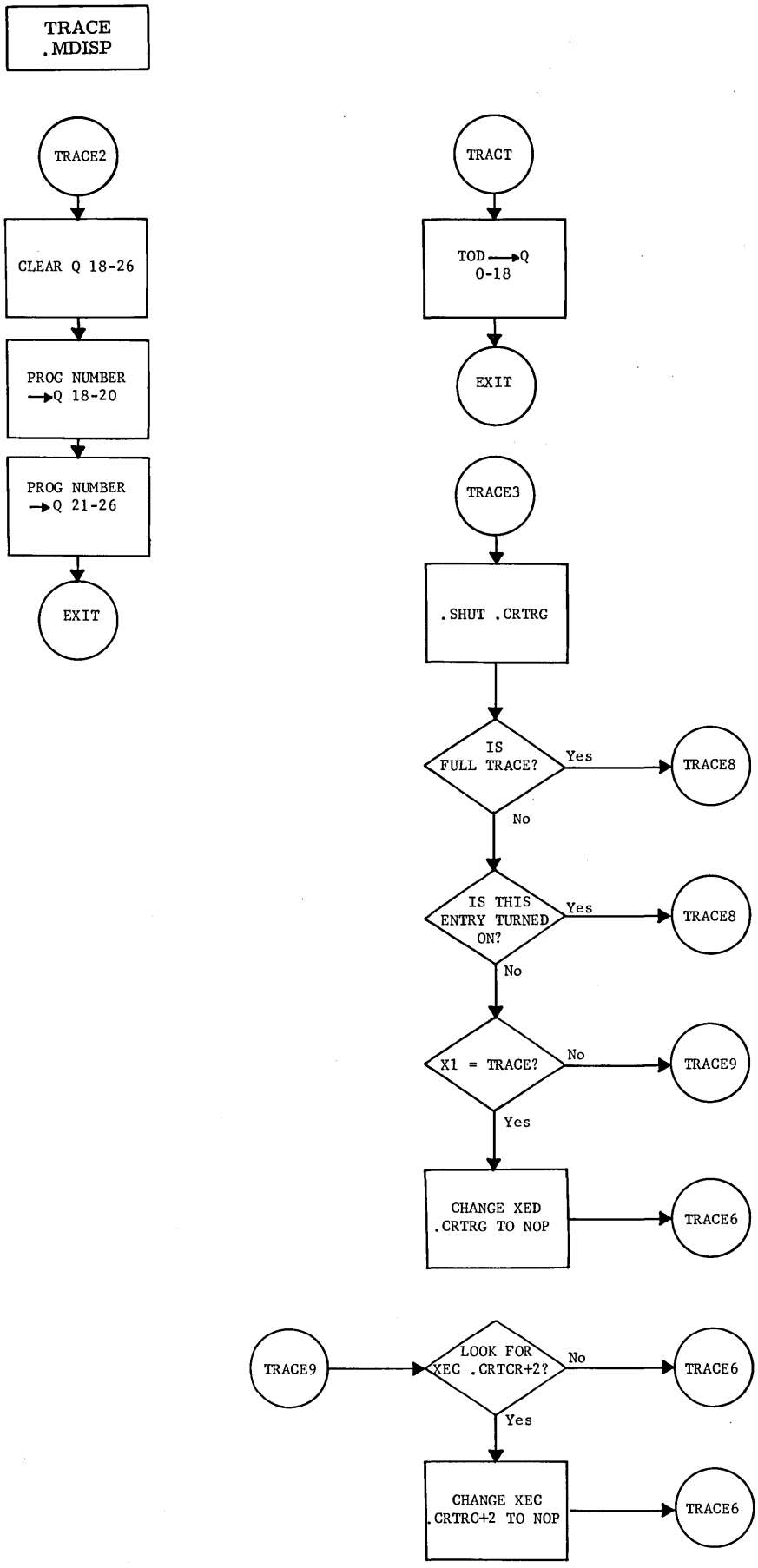

E
IS THIS GEPR POPUP?
Yes
No
SET UP TO CHECKSUM GEPR SSA AREA
SET UP TO CHECKSUM WHOLE SSA
DMKSM
DOES CALC CHECKSUM = STORED CHECKSUM?
No
Yes
HAVE WE TRIED 10 TIMES YET?
No
Yes
DPU2
DLD2
MOVE POPUP STACK TO POSITION
GET REG STORE BIT FROM OLD STACK
OR WITH SAVED IC AND I
STORE IC AND I BACK INTO STACK
DLD4

HGT
.MDISP

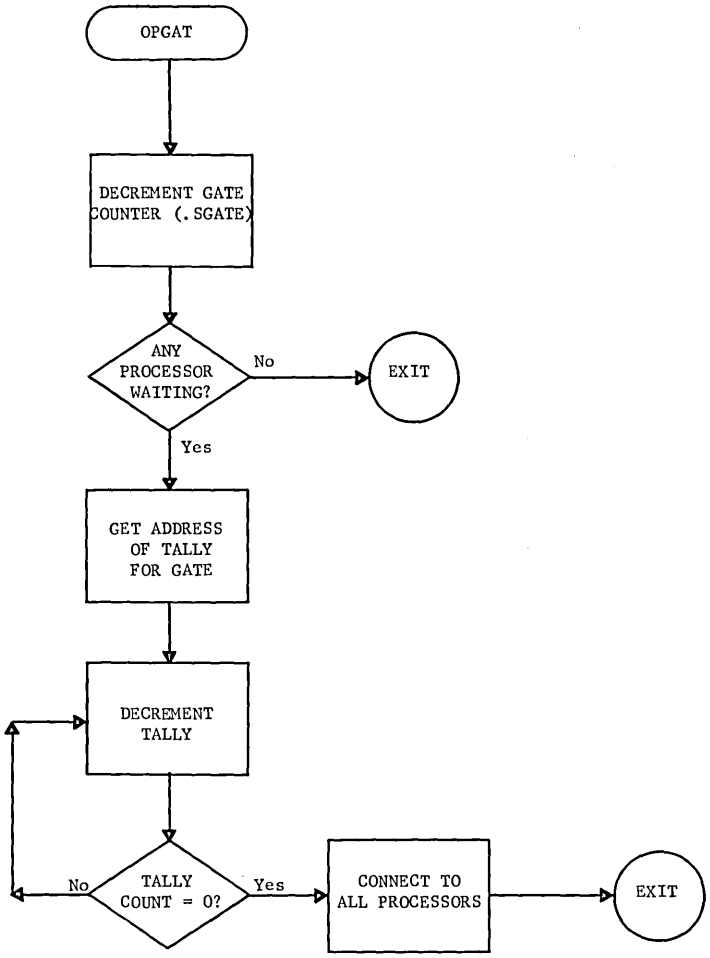

DCKSM
CHECKSUM
.SCKSM = 0?
Yes
EXIT
1,1
No
DMKSM
ISOLATE COUNT
IN Q 18-29
POSITION IN
14-26
LOW ORDER 8
BITS → XO
DCKSMZ
IS
COUNT A MULTIPLE
OF 256?
Yes
QUPPER - 1
→ QUPPER
No
DCKSMZ
ADD BOTH
CARRY COUNT
WORDS
No
QUPPER < 0?
Yes
EXIT
0,1
No
CHECKSUM
= .SCKSM?
Yes
EXIT
1,1

SYSTEM TRACE



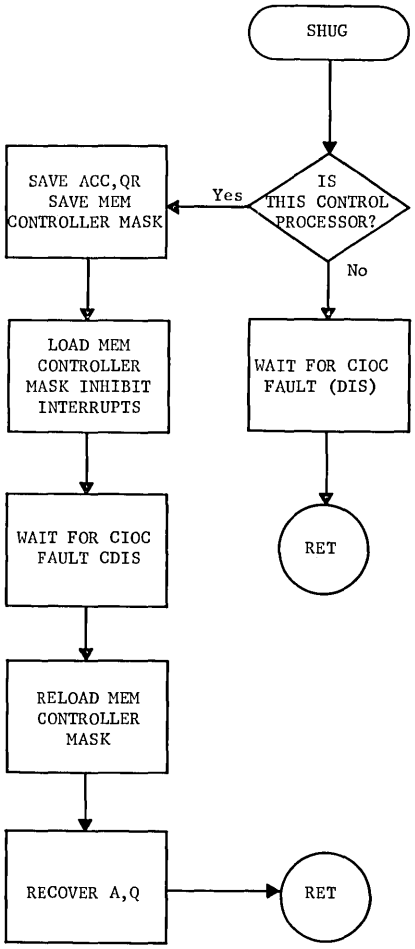

TRACE
.MDISP
TRACE2
CLEAR Q 18-26
PROG NUMBER
→Q 18-20
PROG NUMBER
→Q 21-26
EXIT
TRACT
TOD → Q
0-18
EXIT
TRACE3
.SHUT .CRTRG
IS
FULL TRACE?
Yes
TRACE8
No
IS THIS
ENTRY TURNED
ON?
Yes
TRACE8
No
X1 = TRACE?
No
TRACE9
Yes
CHANGE XED
.CRTRG TO NOP
TRACE6
TRACE9
LOOK FOR
XEC .CRTCR+2?
No
TRACE6
Yes
CHANGE XEC
.CRTRC+2 TO NOP
TRACE6

OPEN SYSTEM GATES



SHUG
.MDISP

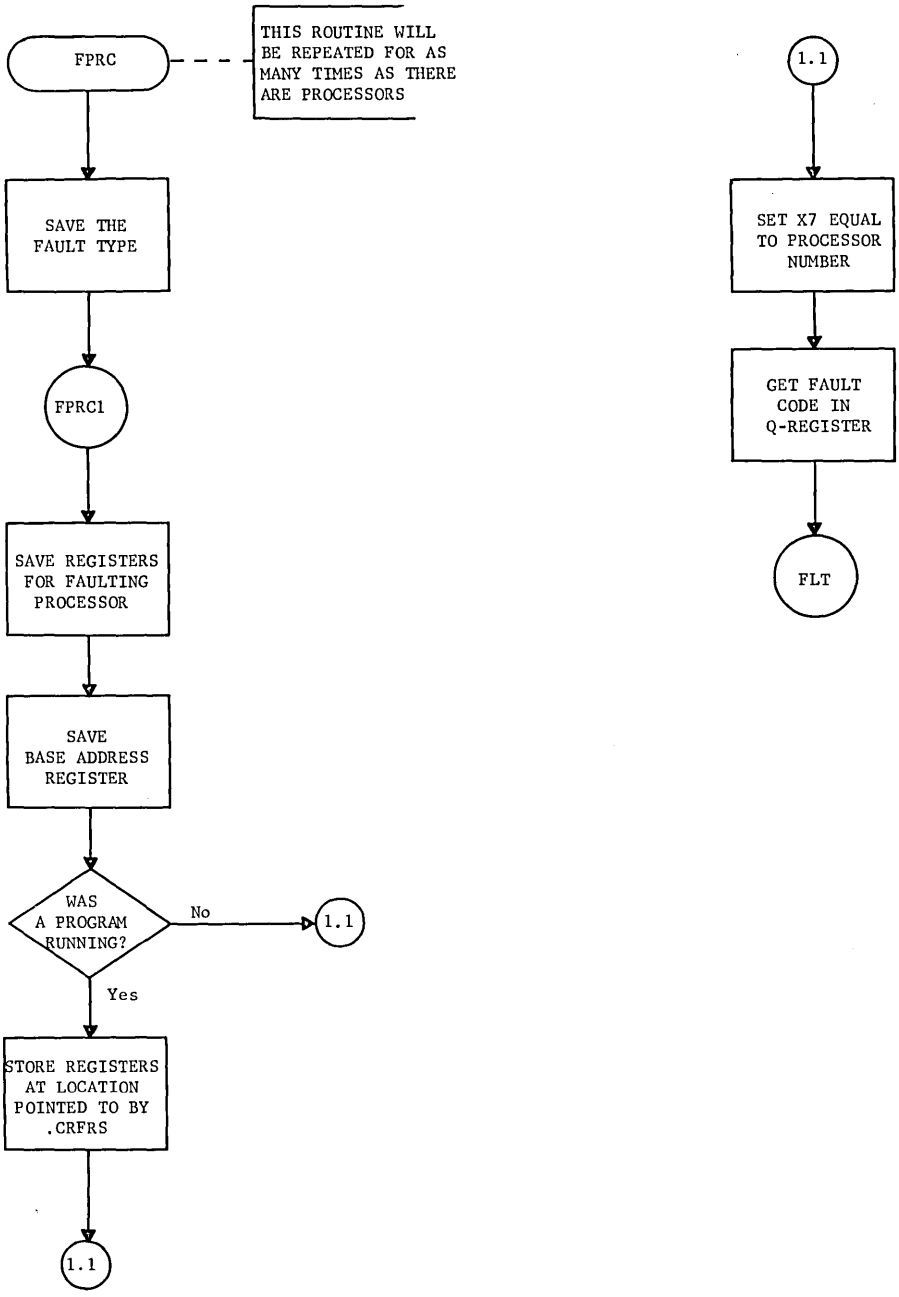
CLOSE SYSTEM GATES



2. FAULT PROCESSING AND SERVICE MME'S

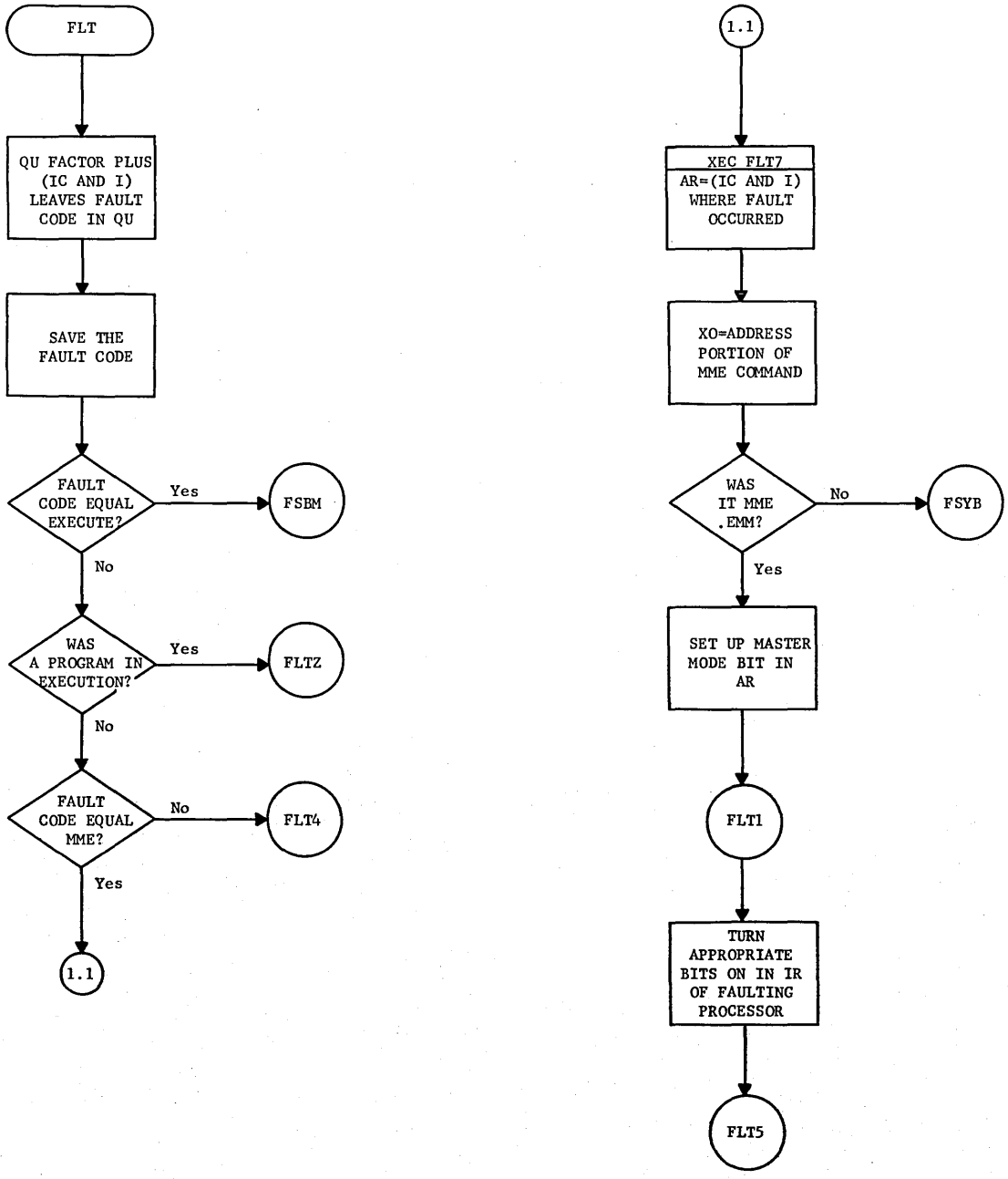
(Reference CPB-1493)

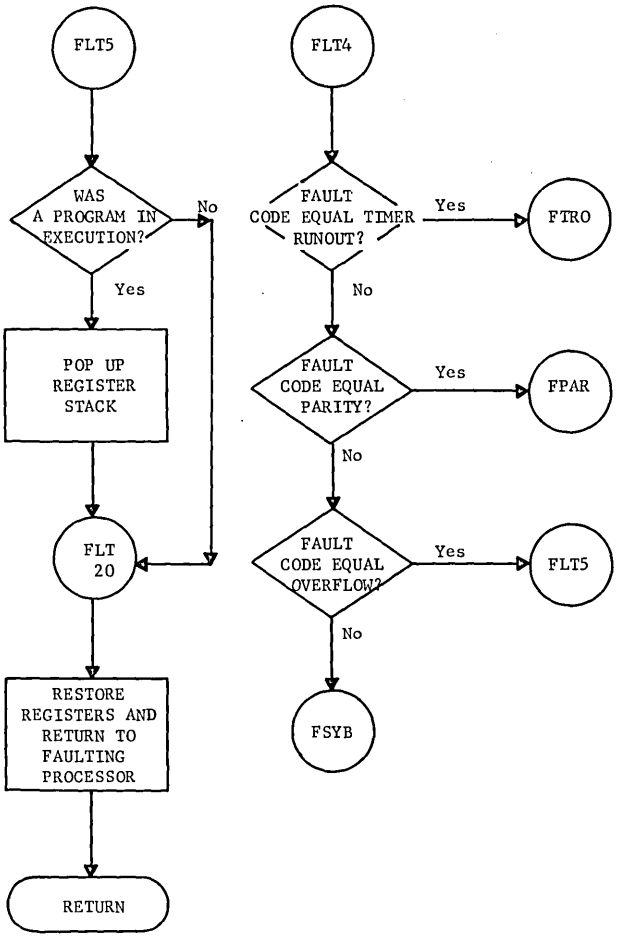
SECONDARY FAULT VECTOR

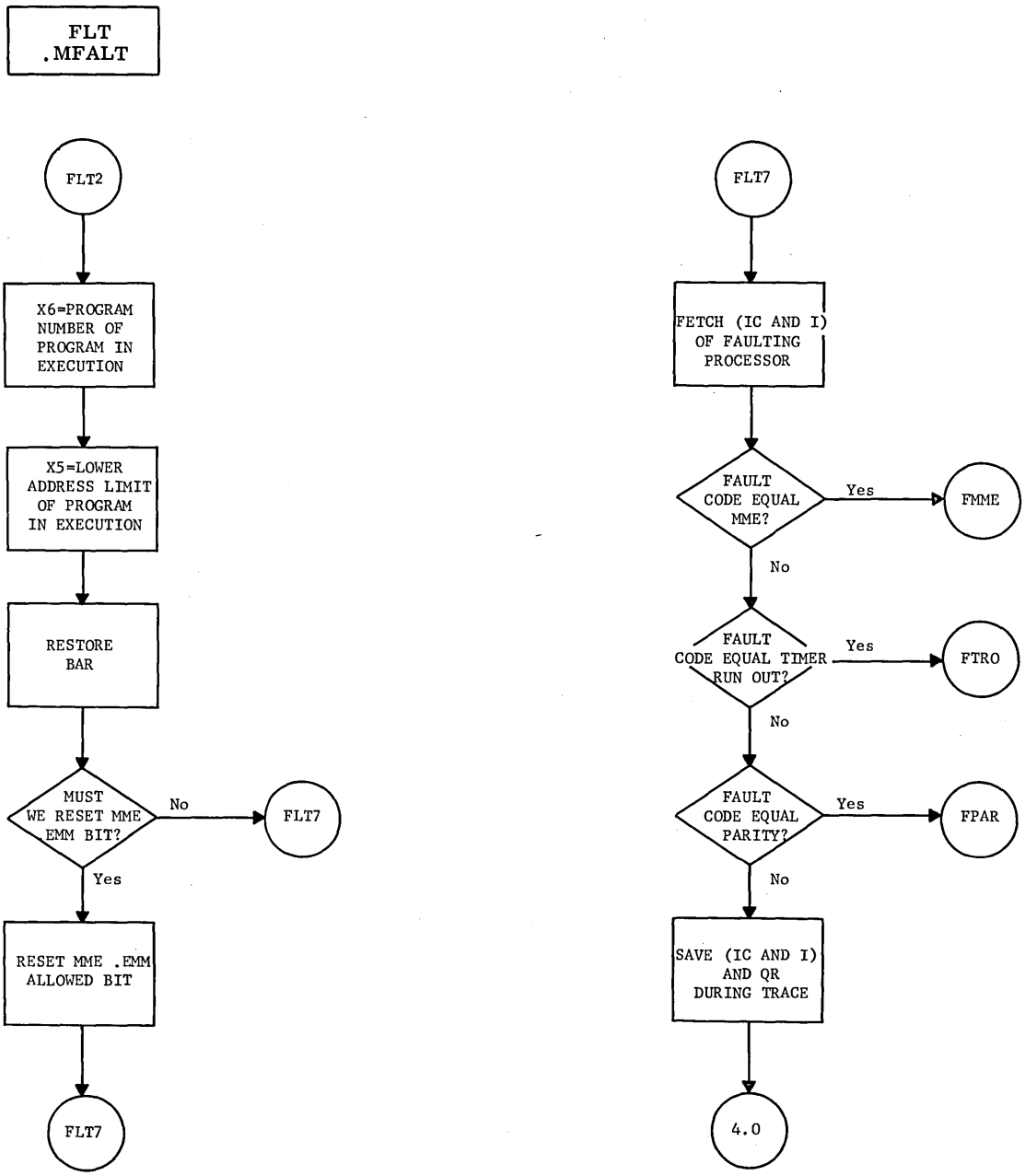


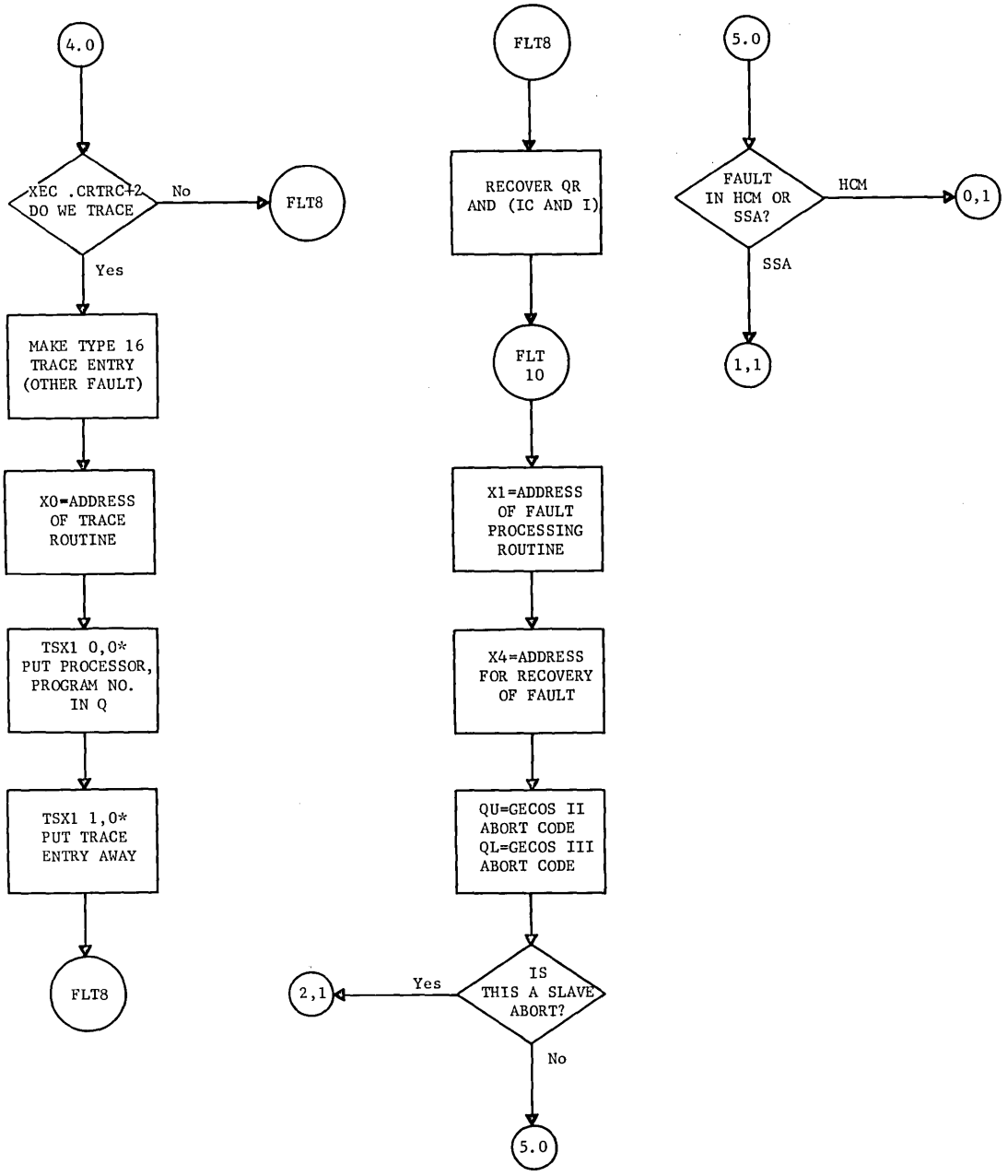
FLT
.MFALT

RECOGNIZE FAULT TYPE AND PROCESS FAULT

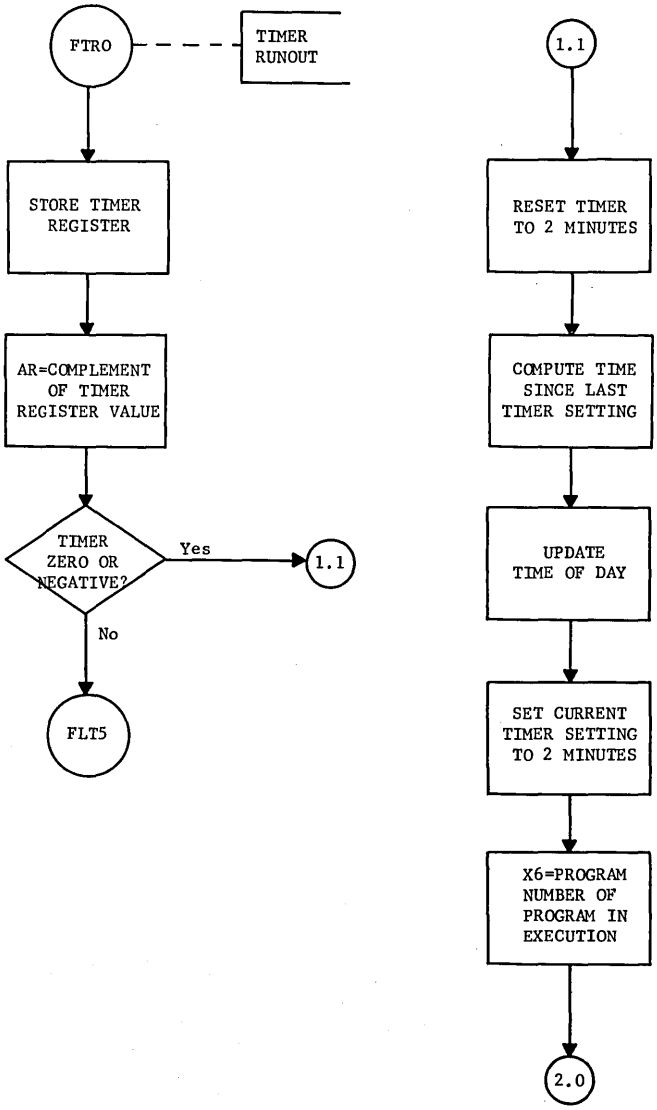


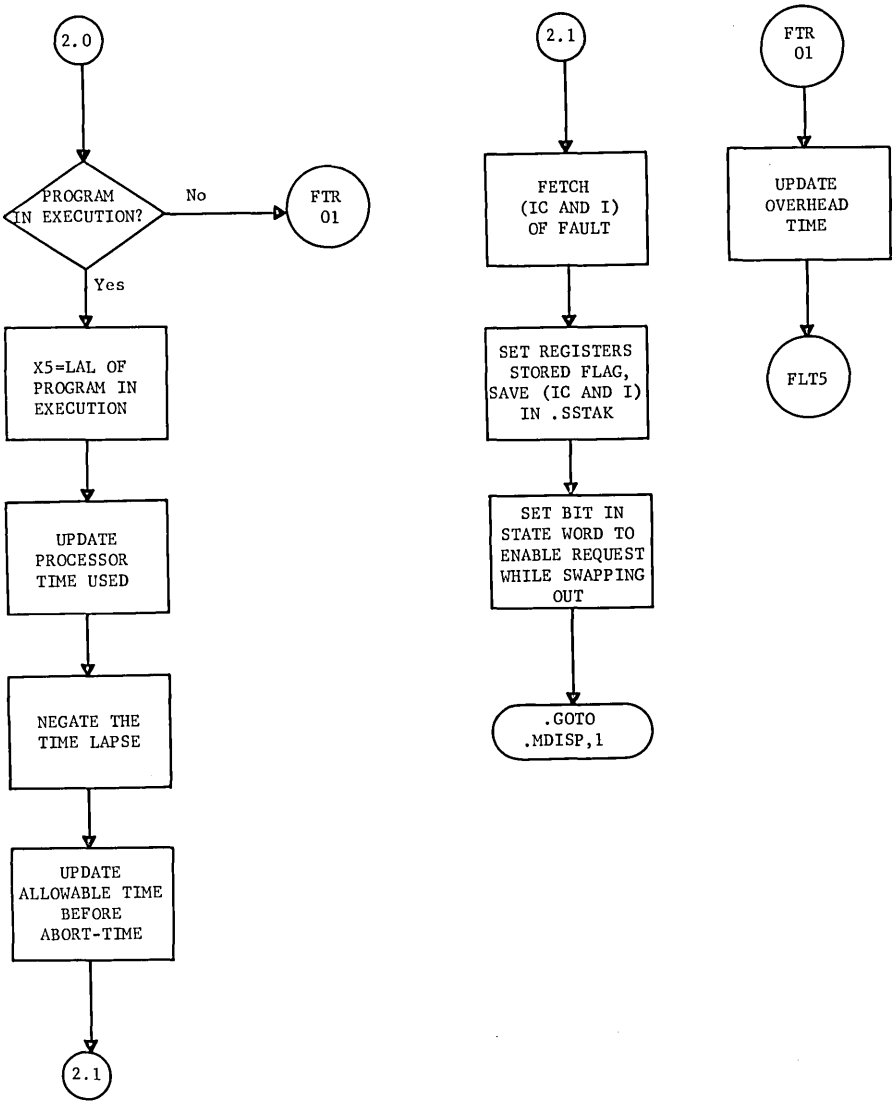

FLT5
WAS A PROGRAM IN EXECUTION?
No
Yes
POP UP REGISTER STACK
FLT 20
RESTORE REGISTERS AND RETURN TO FAULTING PROCESSOR
RETURN
FLT4
FAULT CODE EQUAL TIMER RUNOUT?
Yes
FTRO
No
FAULT CODE EQUAL PARITY?
Yes
FPAR
No
FAULT CODE EQUAL OVERFLOW?
Yes
FLT5
No
FSYB


FLT
.MFALT
FLT2
X6=PROGRAM NUMBER OF PROGRAM IN EXECUTION
X5=LOWER ADDRESS LIMIT OF PROGRAM IN EXECUTION
RESTORE BAR
MUST WE RESET MME .EMM BIT?
No
FLT7
Yes
RESET MME .EMM ALLOWED BIT
FLT7
FLT7
FETCH (IC AND I) OF FAULTING PROCESSOR
FAULT CODE EQUAL MME?
Yes
FMME
No
FAULT CODE EQUAL TIMER RUN OUT?
Yes
FTRO
No
FAULT CODE EQUAL PARITY?
Yes
FPAR
No
SAVE (IC AND I) AND QR DURING TRACE
4.0

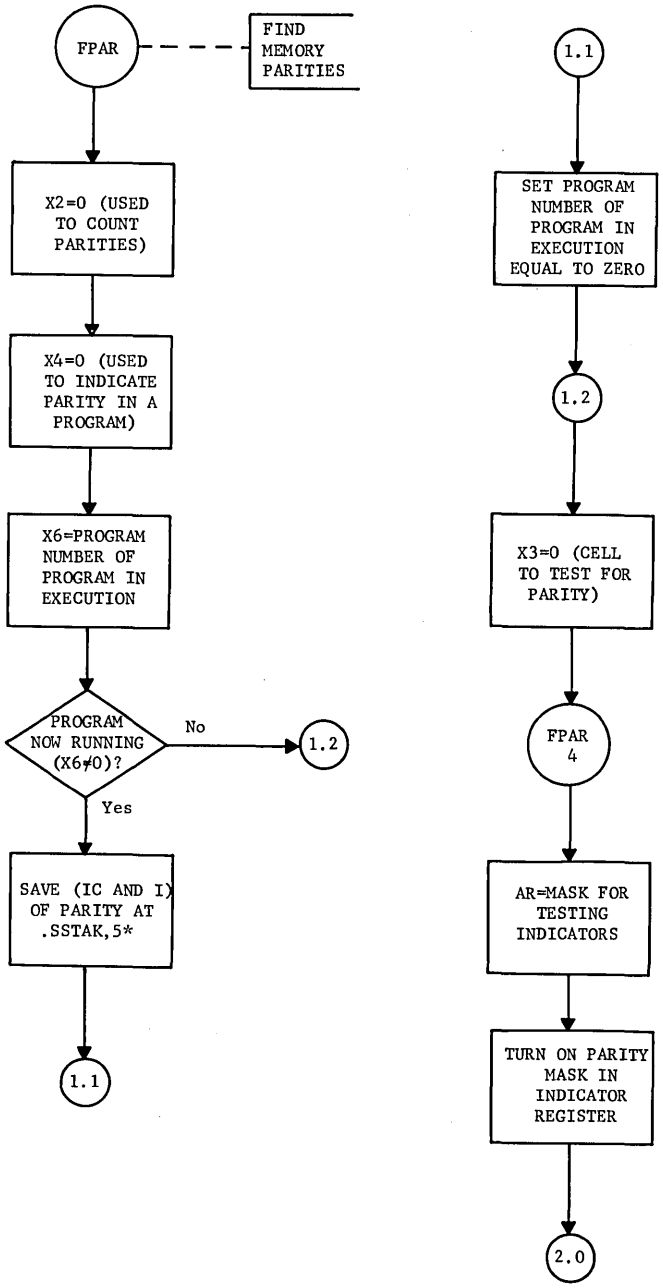

4.0
XEC .CRTRC+2
DO WE TRACE
No
FLT8
Yes
MAKE TYPE 16
TRACE ENTRY
(OTHER FAULT)
X0=ADDRESS
OF TRACE
ROUTINE
TSX1 0,0*
PUT PROCESSOR,
PROGRAM NO.
IN Q
TSX1 1,0*
PUT TRACE
ENTRY AWAY
FLT8
FLT8
RECOVER QR
AND (IC AND I)
FLT
10
X1=ADDRESS
OF FAULT
PROCESSING
ROUTINE
X4=ADDRESS
FOR RECOVERY
OF FAULT
QU=GECOS II
ABORT CODE
QL=GECOS III
ABORT CODE
IS
THIS A SLAVE
ABORT?
Yes
2,1
No
5.0
5.0
FAULT
IN HCM OR
SSA?
HCM
0,1
SSA
1,1

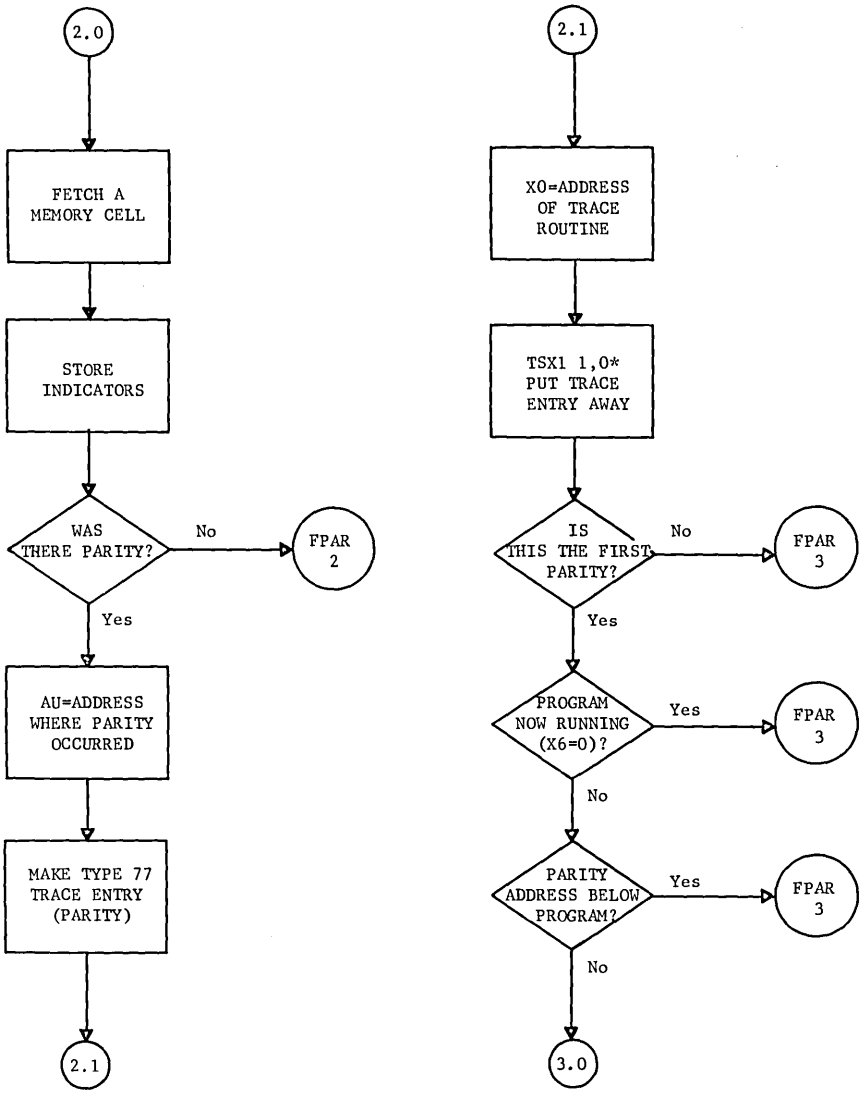
FLT
.MFALT


FTRO
TIMER RUNOUT
STORE TIMER REGISTER
AR=COMPLEMENT OF TIMER REGISTER VALUE
TIMER ZERO OR NEGATIVE?
Yes
1.1
No
FLT5
1.1
RESET TIMER TO 2 MINUTES
COMPUTE TIME SINCE LAST TIMER SETTING
UPDATE TIME OF DAY
SET CURRENT TIMER SETTING TO 2 MINUTES
X6=PROGRAM NUMBER OF PROGRAM IN EXECUTION
2.0

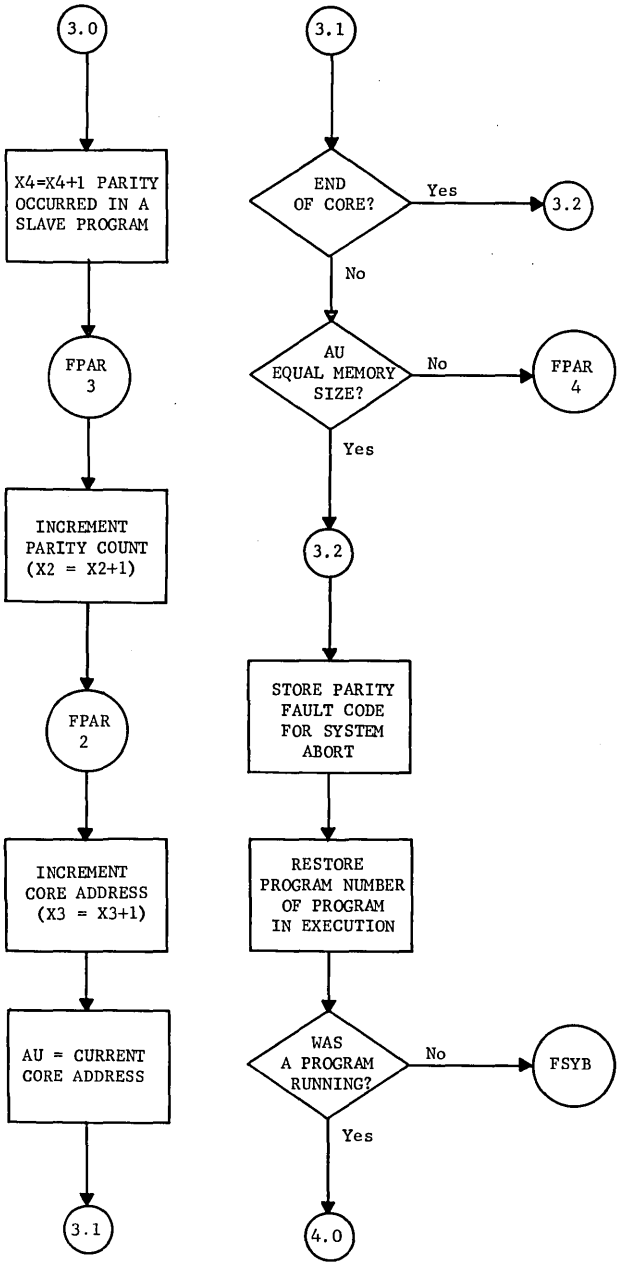

2.0
PROGRAM IN EXECUTION?
No
FTR 01
Yes
X5=LAL OF PROGRAM IN EXECUTION
UPDATE PROCESSOR TIME USED
NEGATE THE TIME LAPSE
UPDATE ALLOWABLE TIME BEFORE ABORT-TIME
2.1
2.1
FETCH (IC AND I) OF FAULT
SET REGISTERS STORED FLAG, SAVE (IC AND I) IN .SSTAK
SET BIT IN STATE WORD TO ENABLE REQUEST WHILE SWAPPING OUT
.GOTO .MDISP,1
FTR 01
UPDATE OVERHEAD TIME
FLT5

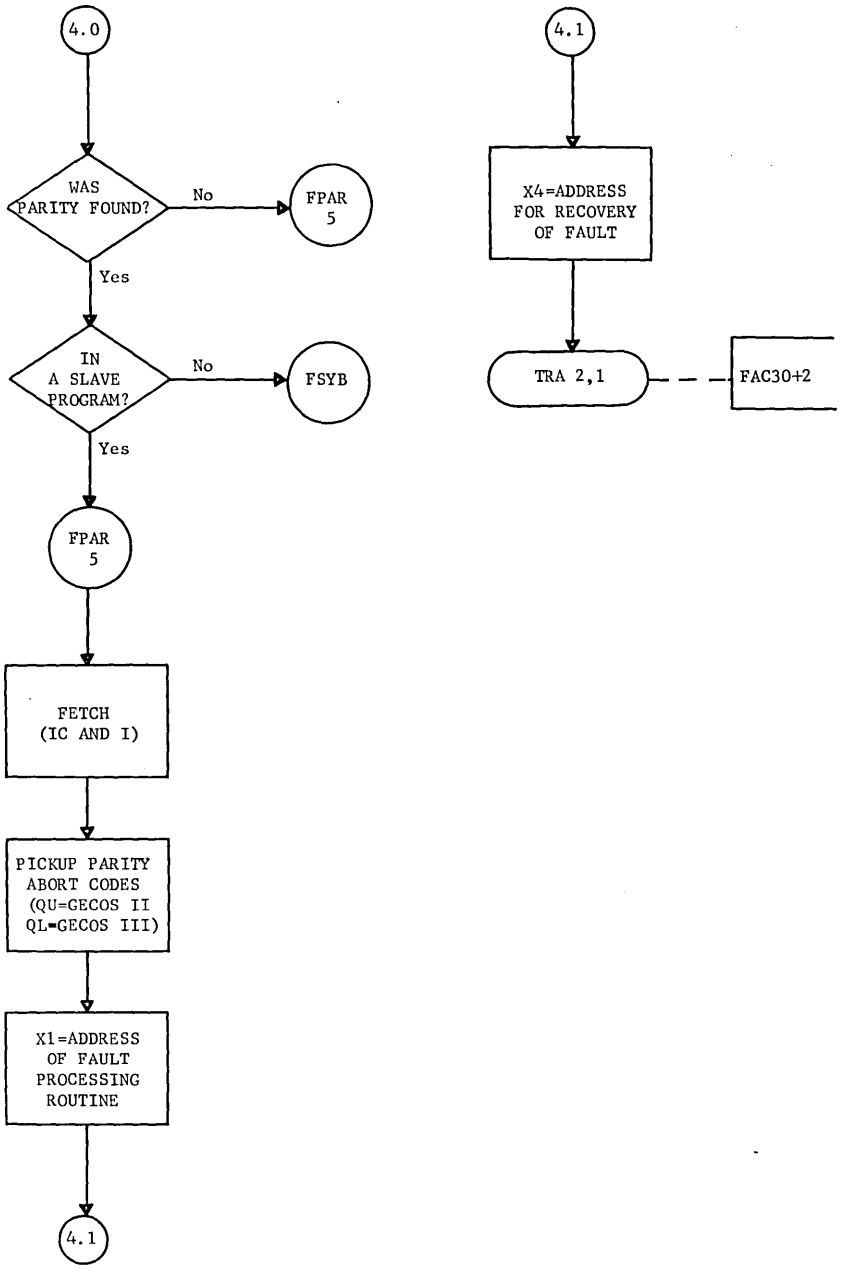
FLT
.MFALT

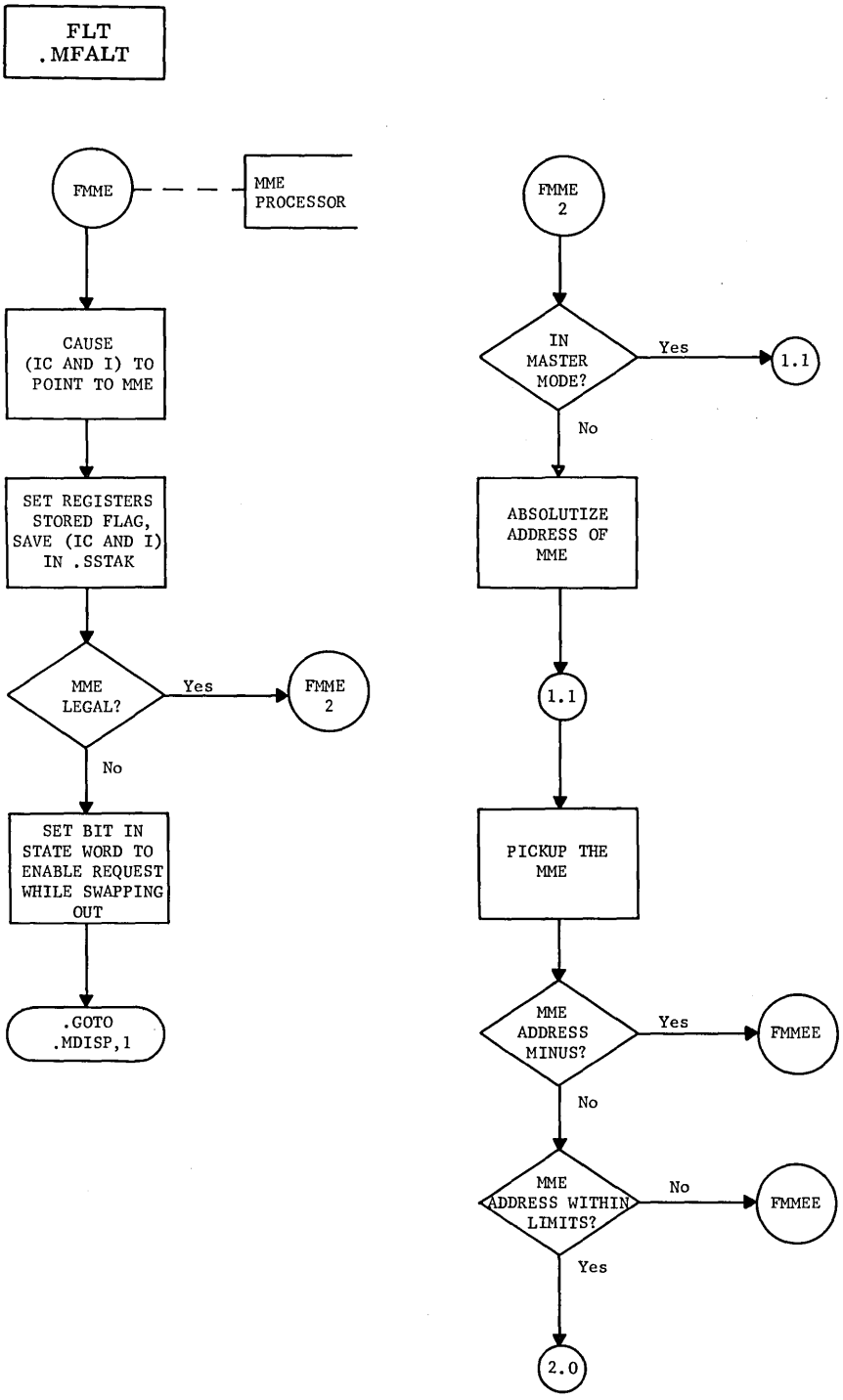

FPAR
FIND MEMORY PARITIES
X2=0 (USED TO COUNT PARITIES)
X4=0 (USED TO INDICATE PARITY IN A PROGRAM)
X6=PROGRAM NUMBER OF PROGRAM IN EXECUTION
PROGRAM NOW RUNNING (X6≠0)?
No
1.2
Yes
SAVE (IC AND I) OF PARITY AT .SSTAK,5*
1.1
1.1
SET PROGRAM NUMBER OF PROGRAM IN EXECUTION EQUAL TO ZERO
1.2
X3=0 (CELL TO TEST FOR PARITY)
FPAR 4
AR=MASK FOR TESTING INDICATORS
TURN ON PARITY MASK IN INDICATOR REGISTER
2.0

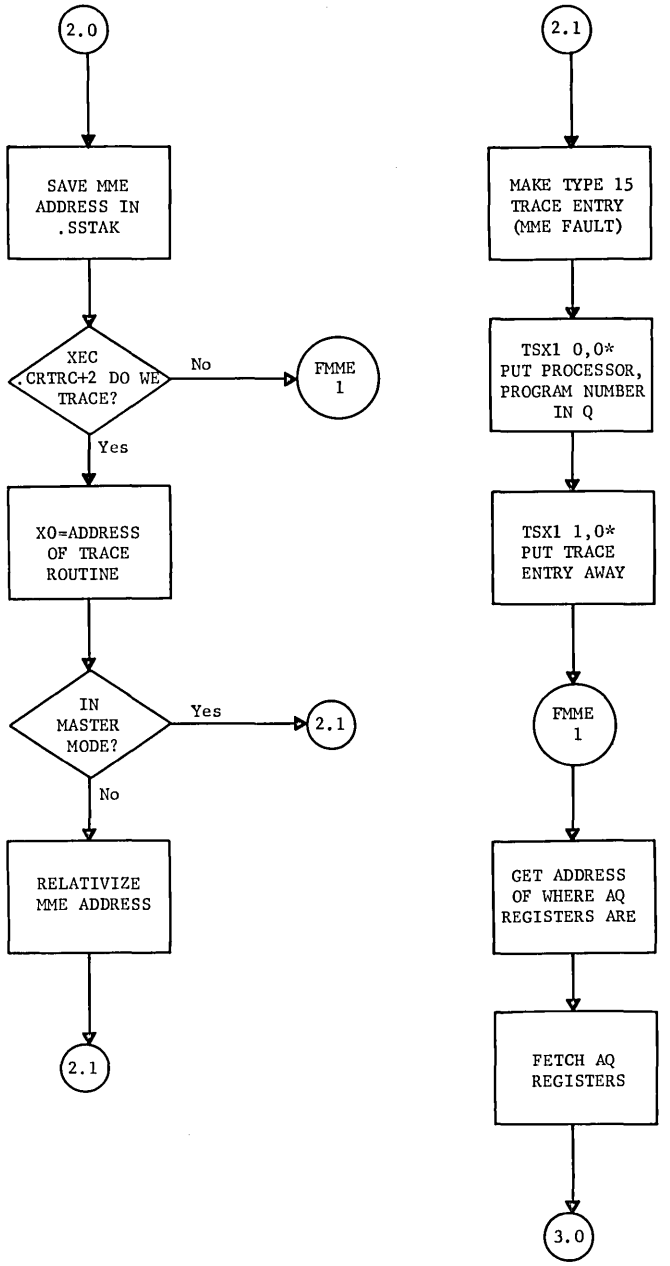

2.0
FETCH A MEMORY CELL
STORE INDICATORS
WAS THERE PARITY?
No
FPAR 2
Yes
AU=ADDRESS WHERE PARITY OCCURRED
MAKE TYPE 77 TRACE ENTRY (PARITY)
2.1
2.1
X0=ADDRESS OF TRACE ROUTINE
TSX1 1,0* PUT TRACE ENTRY AWAY
IS THIS THE FIRST PARITY?
No
FPAR 3
Yes
PROGRAM NOW RUNNING (X6=0)?
Yes
FPAR 3
No
PARITY ADDRESS BELOW PROGRAM?
Yes
FPAR 3
No
3.0

FLT
.MFALT

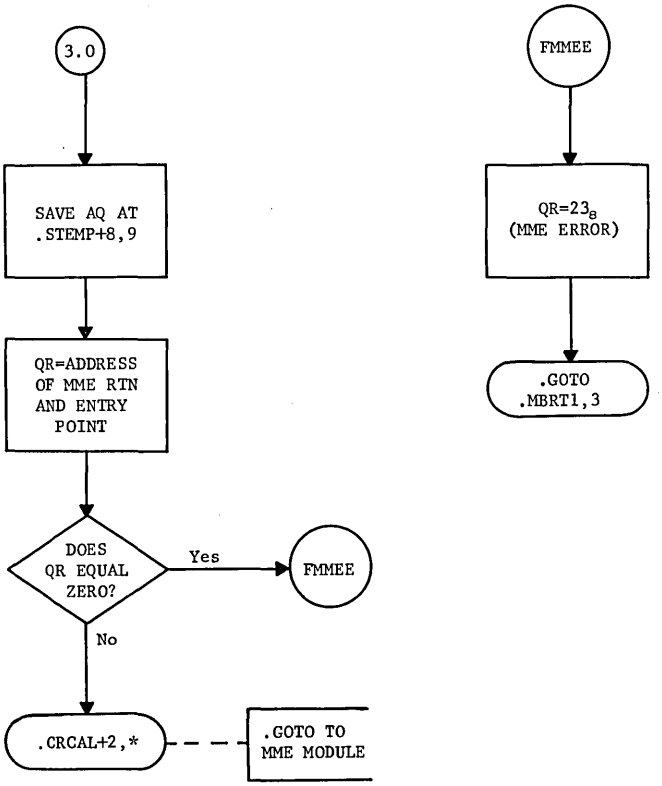

3.0
X4=X4+1 PARITY OCCURRED IN A SLAVE PROGRAM
FPAR 3
INCREMENT PARITY COUNT (X2 = X2+1)
FPAR 2
INCREMENT CORE ADDRESS (X3 = X3+1)
AU = CURRENT CORE ADDRESS
3.1
3.1
END OF CORE?
Yes
3.2
No
AU EQUAL MEMORY SIZE?
No
FPAR 4
Yes
3.2
STORE PARITY FAULT CODE FOR SYSTEM ABORT
RESTORE PROGRAM NUMBER OF PROGRAM IN EXECUTION
WAS A PROGRAM RUNNING?
No
FSYB
Yes
4.0

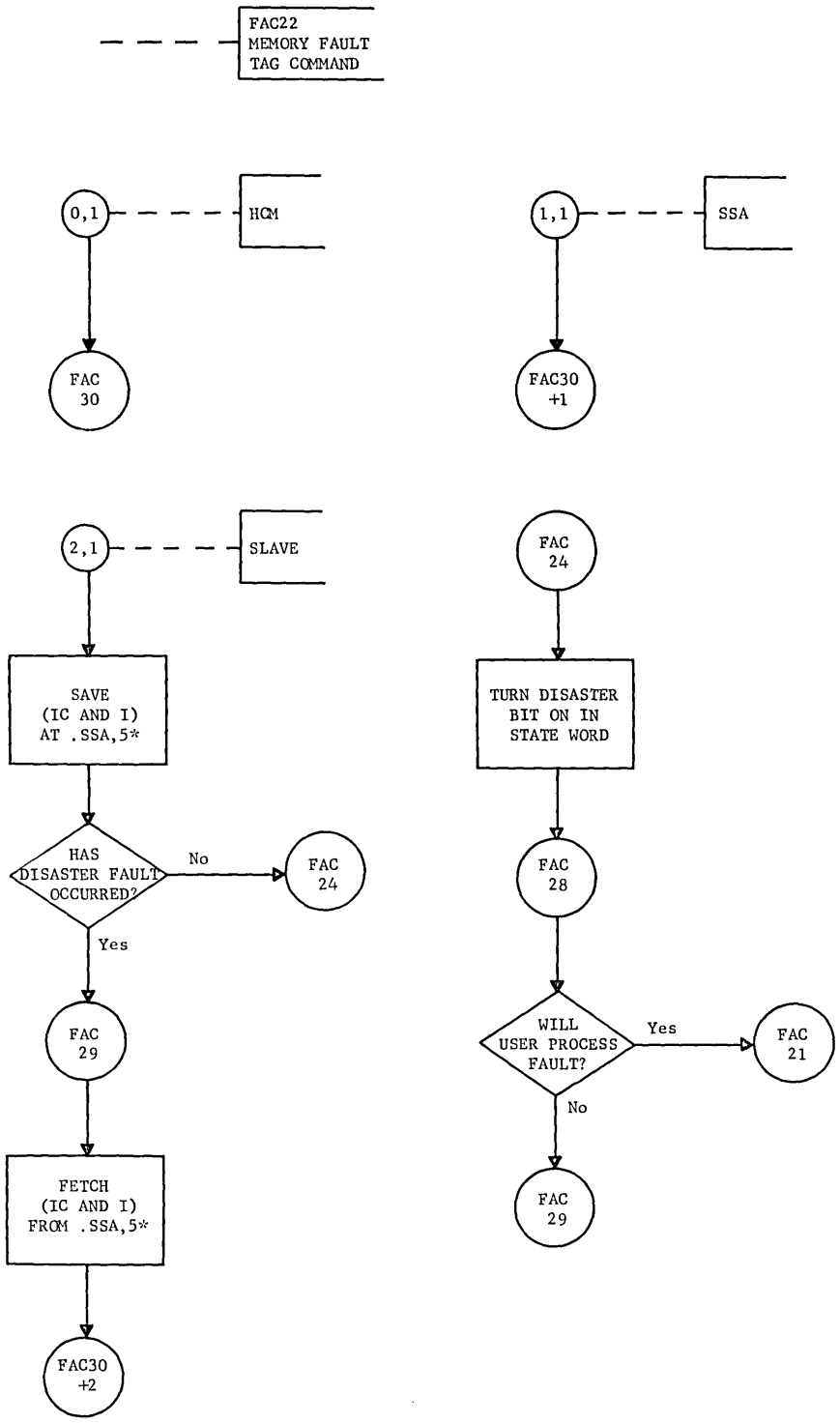

4.0
WAS PARITY FOUND?
No
FPAR 5
Yes
IN A SLAVE PROGRAM?
No
FSYB
Yes
FPAR 5
FETCH (IC AND I)
PICKUP PARITY ABORT CODES (QU=GECOS II QL=GECOS III)
X1=ADDRESS OF FAULT PROCESSING ROUTINE
4.1
4.1
X4=ADDRESS FOR RECOVERY OF FAULT
TRA 2,1
FAC30+2


FLT
.MFALT
FMME
MME
PROCESSOR
CAUSE
(IC AND I) TO
POINT TO MME
SET REGISTERS
STORED FLAG,
SAVE (IC AND I)
IN .SSTAK
MME
LEGAL?
Yes
FMME
2
No
SET BIT IN
STATE WORD TO
ENABLE REQUEST
WHILE SWAPPING
OUT
.GOTO
.MDISP,1
FMME
2
IN
MASTER
MODE?
Yes
1.1
No
ABSOLUTIZE
ADDRESS OF
MME
1.1
PICKUP THE
MME
MME
ADDRESS
MINUS?
Yes
FMMEE
No
MME
ADDRESS WITHIN
LIMITS?
No
FMMEE
Yes
2.0

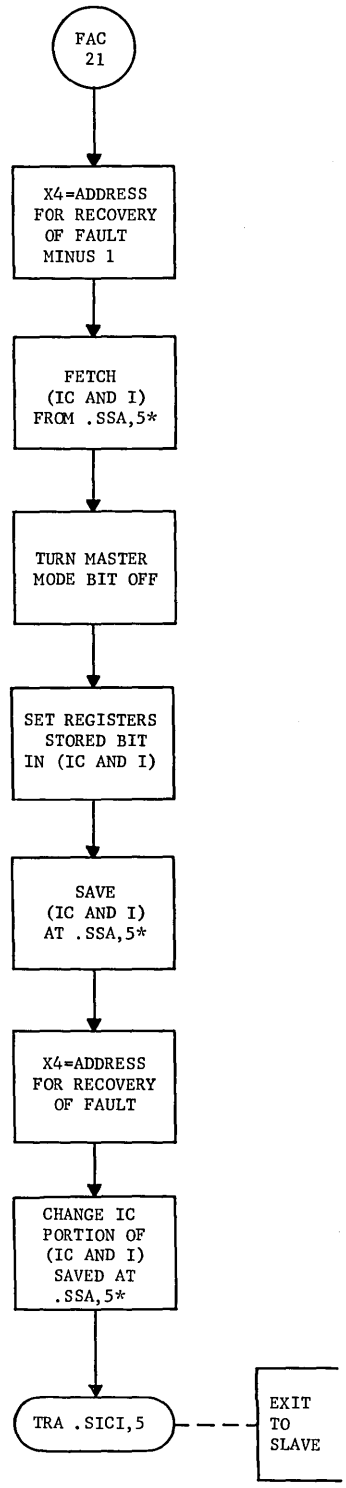

2.0
SAVE MME ADDRESS IN .SSTAK
XEC .CRTRC+2 DO WE TRACE?
No
FMME 1
Yes
XO=ADDRESS OF TRACE ROUTINE
IN MASTER MODE?
Yes
2.1
No
RELATIVIZE MME ADDRESS
2.1
2.1
MAKE TYPE 15 TRACE ENTRY (MME FAULT)
TSX1 0,0* PUT PROCESSOR, PROGRAM NUMBER IN Q
TSX1 1,0* PUT TRACE ENTRY AWAY
FMME 1
GET ADDRESS OF WHERE AQ REGISTERS ARE
FETCH AQ REGISTERS
3.0

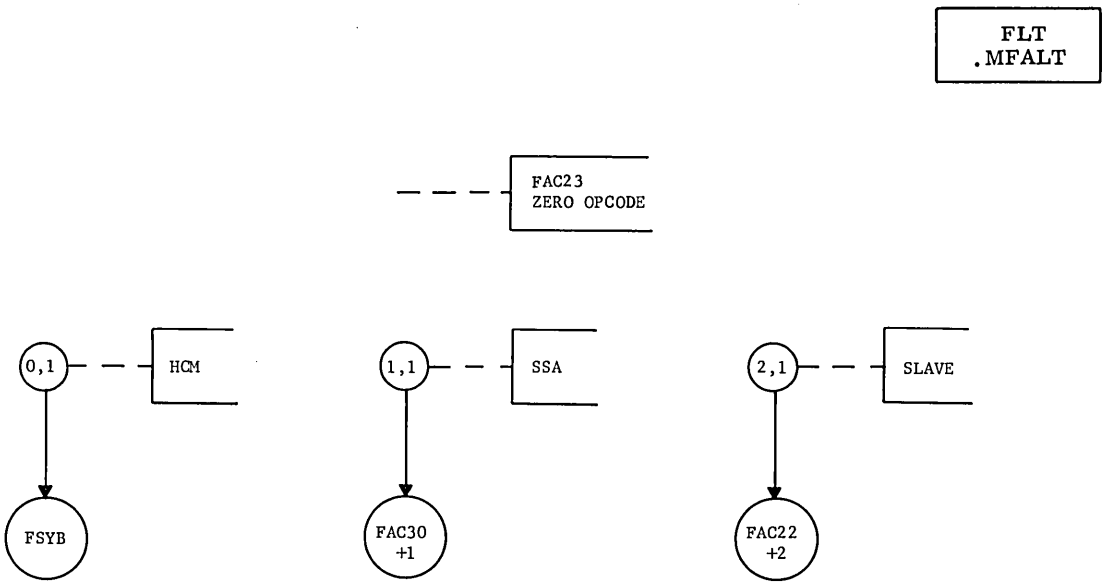
FLT
.MFALT

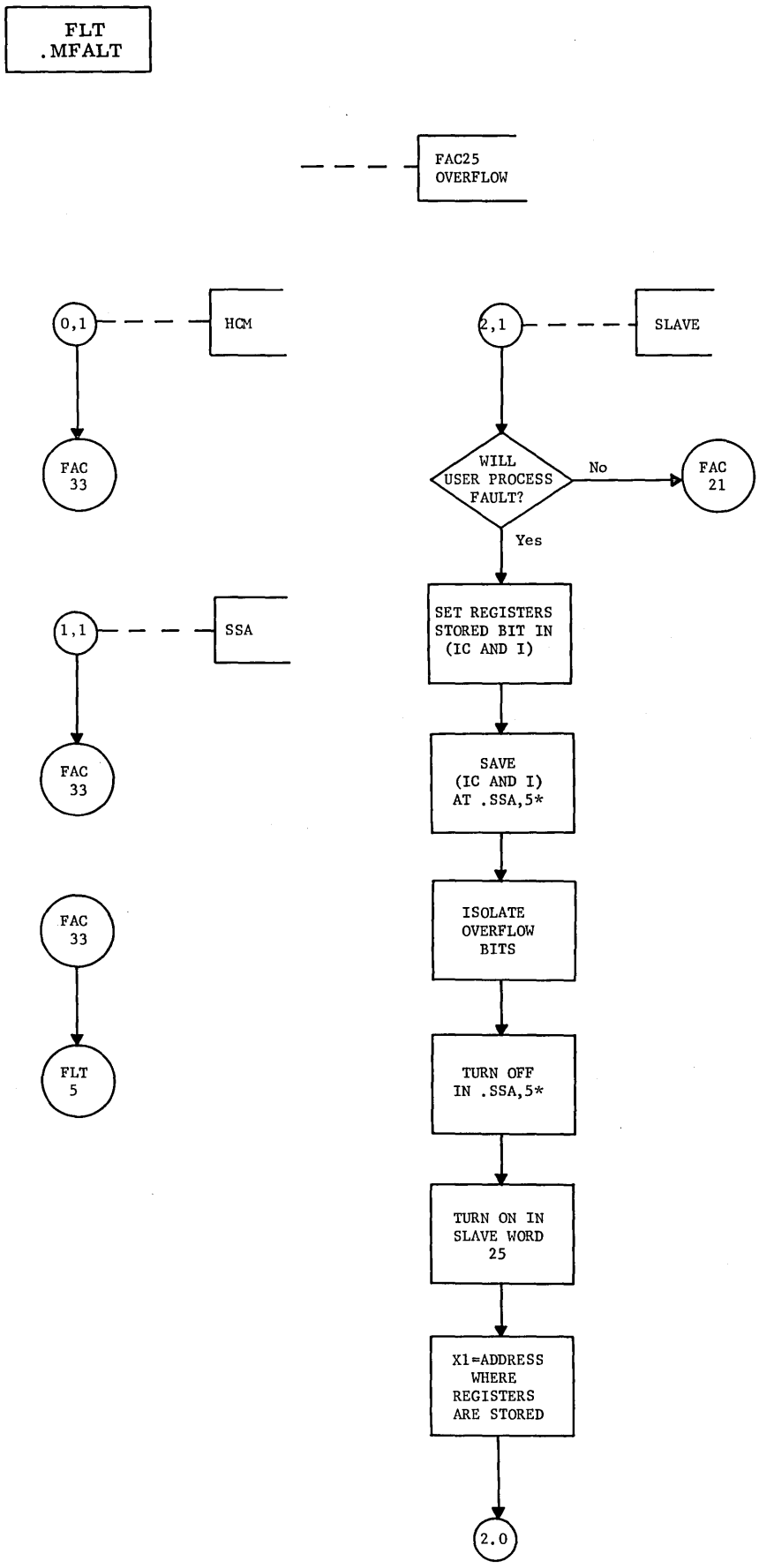

3.0
SAVE AQ AT
.STEMP+8,9
QR=ADDRESS
OF MME RTN
AND ENTRY
POINT
DOES
QR EQUAL
ZERO?
Yes
FMMEE
No
.CRCAL+2,*
.GOTO TO
MME MODULE
FMMEE
QR=23$_8$
(MME ERROR)
.GOTO
.MBRT1,3

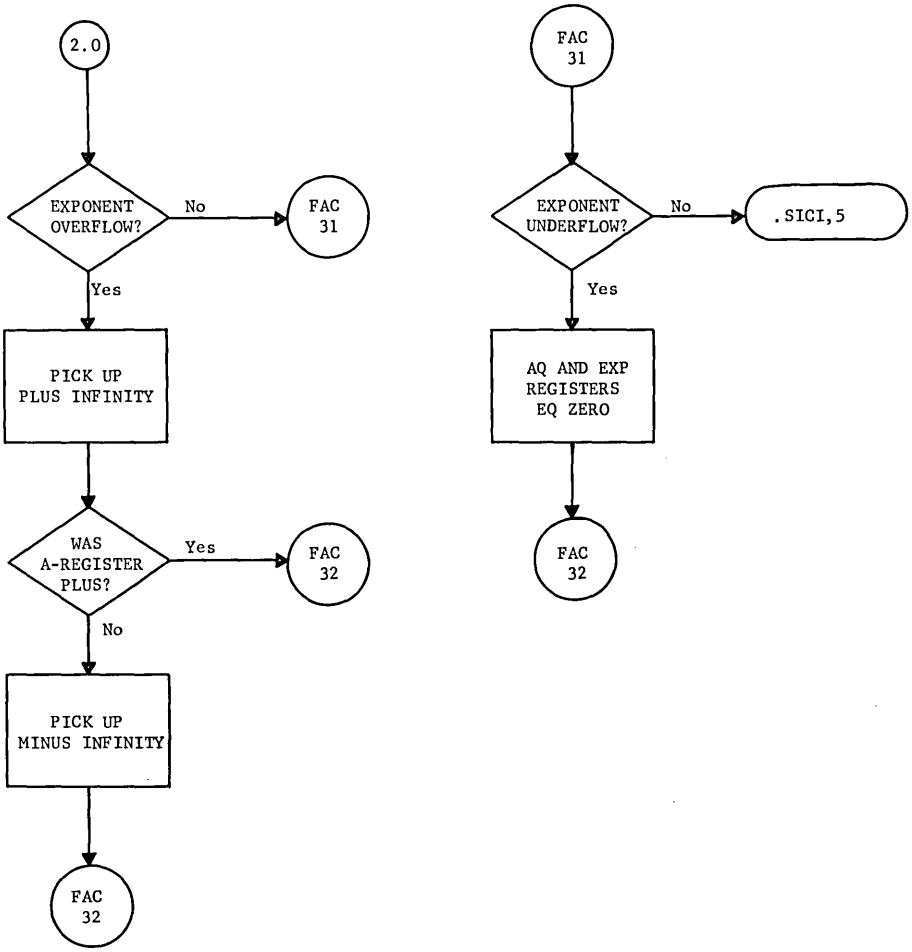

FAC22
MEMORY FAULT
TAG COMMAND
0,1
HCM
FAC
30
1,1
SSA
FAC30
+1
2,1
SLAVE
SAVE
(IC AND I)
AT .SSA,5*
HAS
DISASTER FAULT
OCCURRED?
No
FAC
24
Yes
FAC
29
FETCH
(IC AND I)
FROM .SSA,5*
FAC30
+2
FAC
24
TURN DISASTER
BIT ON IN
STATE WORD
FAC
28
WILL
USER PROCESS
FAULT?
Yes
FAC
21
No
FAC
29

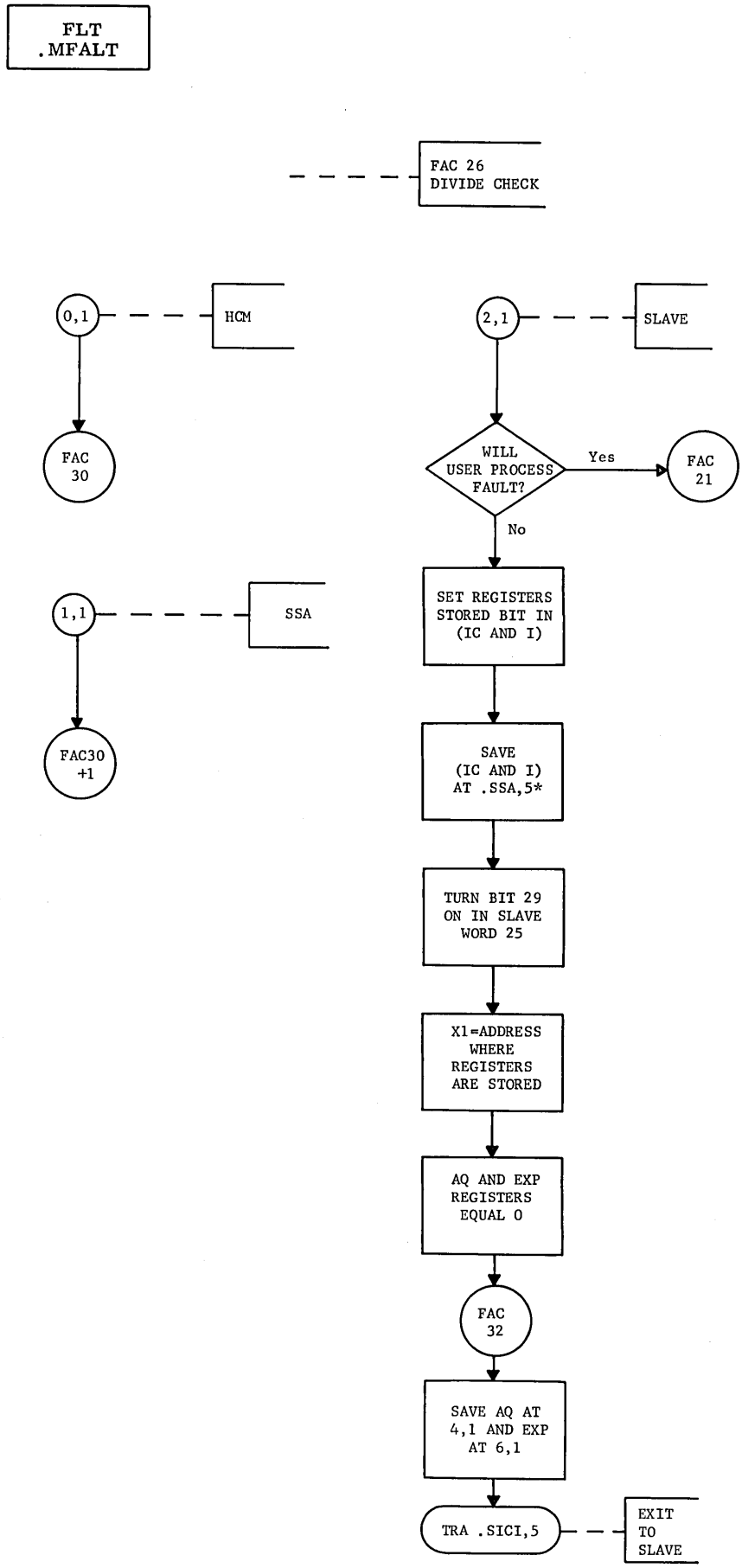
FLT
.MFLAT

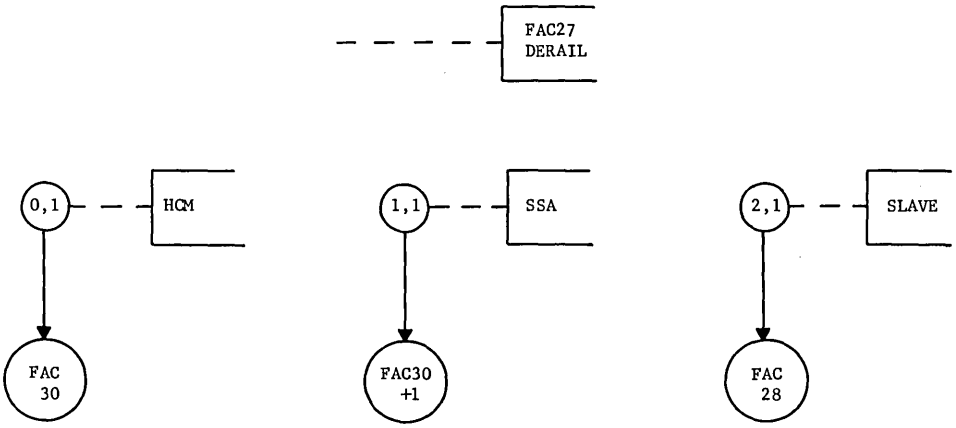

FAC
21
X4=ADDRESS
FOR RECOVERY
OF FAULT
MINUS 1
FETCH
(IC AND I)
FROM .SSA,5*
TURN MASTER
MODE BIT OFF
SET REGISTERS
STORED BIT
IN (IC AND I)
SAVE
(IC AND I)
AT .SSA,5*
X4=ADDRESS
FOR RECOVERY
OF FAULT
CHANGE IC
PORTION OF
(IC AND I)
SAVED AT
.SSA,5*
TRA .SICI,5
EXIT
TO
SLAVE

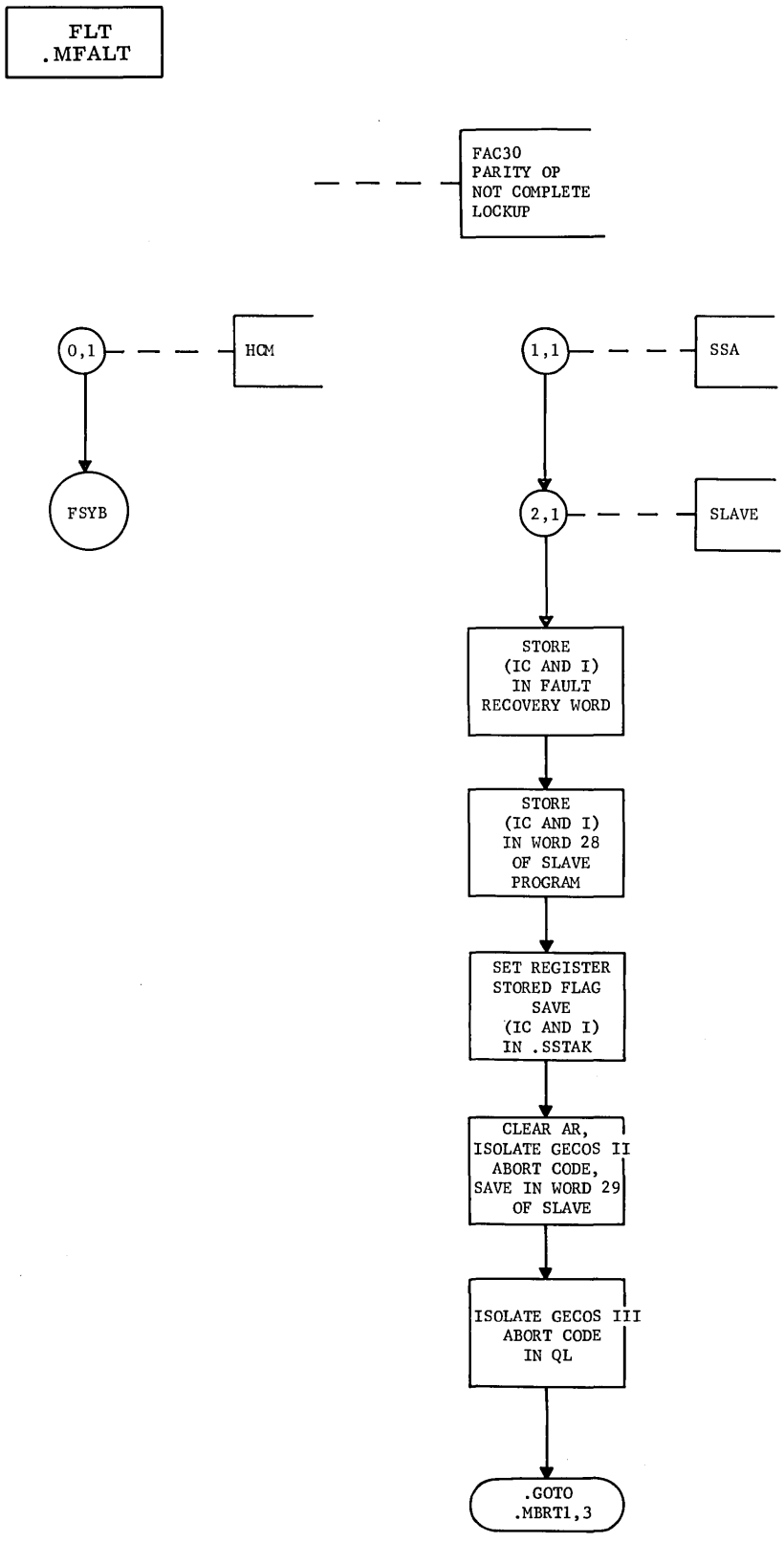

FLT
.MFALT
FAC23
ZERO OPCODE
0,1
HCM
FSYB
1,1
SSA
FAC30
+1
2,1
SLAVE
FAC22
+2


FLT
.MFALT
FAC25
OVERFLOW
0,1
HCM
FAC
33
1,1
SSA
FAC
33
FAC
33
FLT
5
2,1
SLAVE
WILL
USER PROCESS
FAULT?
No
FAC
21
Yes
SET REGISTERS
STORED BIT IN
(IC AND I)
SAVE
(IC AND I)
AT .SSA,5*
ISOLATE
OVERFLOW
BITS
TURN OFF
IN .SSA,5*
TURN ON IN
SLAVE WORD
25
X1=ADDRESS
WHERE
REGISTERS
ARE STORED
2.0


2.0
EXPONENT OVERFLOW?
No
FAC 31
Yes
PICK UP PLUS INFINITY
WAS A-REGISTER PLUS?
Yes
FAC 32
No
PICK UP MINUS INFINITY
FAC 32
FAC 31
EXPONENT UNDERFLOW?
No
.SICI,5
Yes
AQ AND EXP REGISTERS EQ ZERO
FAC 32

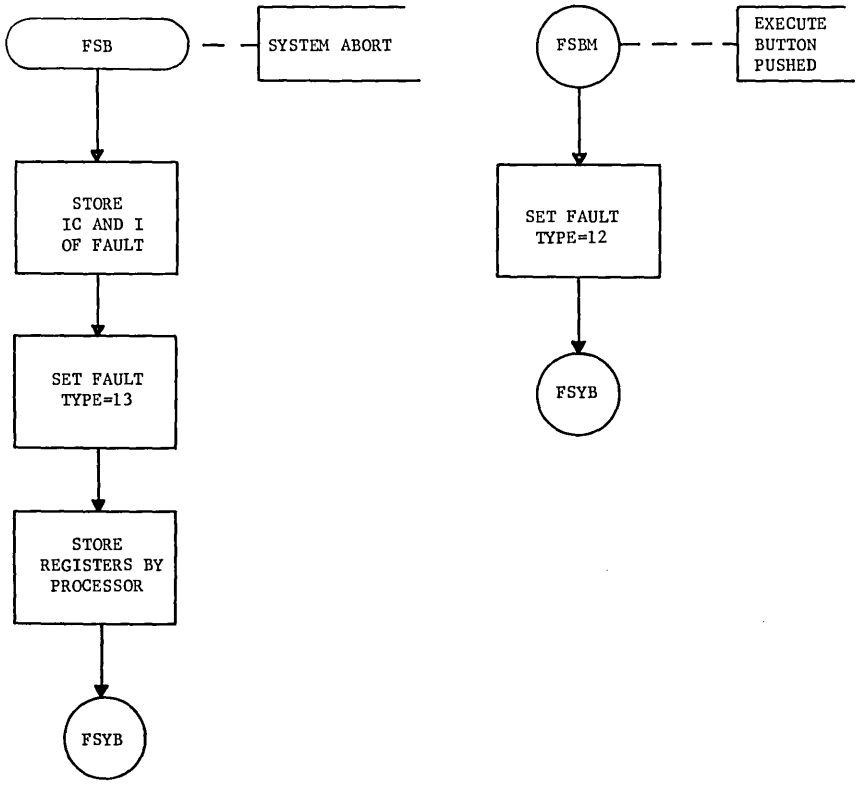

FLT
.MFALT
FAC 26
DIVIDE CHECK
0,1
HCM
FAC
30
2,1
SLAVE
WILL
USER PROCESS
FAULT?
Yes
FAC
21
No
1,1
SSA
FAC30
+1
SET REGISTERS
STORED BIT IN
(IC AND I)
SAVE
(IC AND I)
AT .SSA,5*
TURN BIT 29
ON IN SLAVE
WORD 25
X1=ADDRESS
WHERE
REGISTERS
ARE STORED
AQ AND EXP
REGISTERS
EQUAL 0
FAC
32
SAVE AQ AT
4,1 AND EXP
AT 6,1
TRA .SICI,5
EXIT
TO
SLAVE


FAC27
DERAIL
0,1
HCM
FAC
30
1,1
SSA
FAC30
+1
2,1
SLAVE
FAC
28


FLT
.MFALT
FAC30
PARITY OP
NOT COMPLETE
LOCKUP
0,1
HCM
FSYB
1,1
SSA
2,1
SLAVE
STORE
(IC AND I)
IN FAULT
RECOVERY WORD
STORE
(IC AND I)
IN WORD 28
OF SLAVE
PROGRAM
SET REGISTER
STORED FLAG
SAVE
(IC AND I)
IN .SSTAK
CLEAR AR,
ISOLATE GECOS II
ABORT CODE,
SAVE IN WORD 29
OF SLAVE
ISOLATE GECOS III
ABORT CODE
IN QL
.GOTO
.MBRT1,3

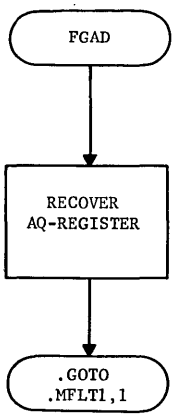
FSB (EP1)
.MFALT

TAKE DUMP FOR SYSTEM ABORT

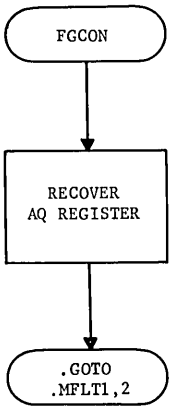


FGAD (EP2)
.MFALT

GET ADDRESS FOR FILE CODE

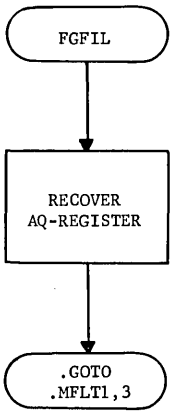


PUT INFORMATION IN FCB

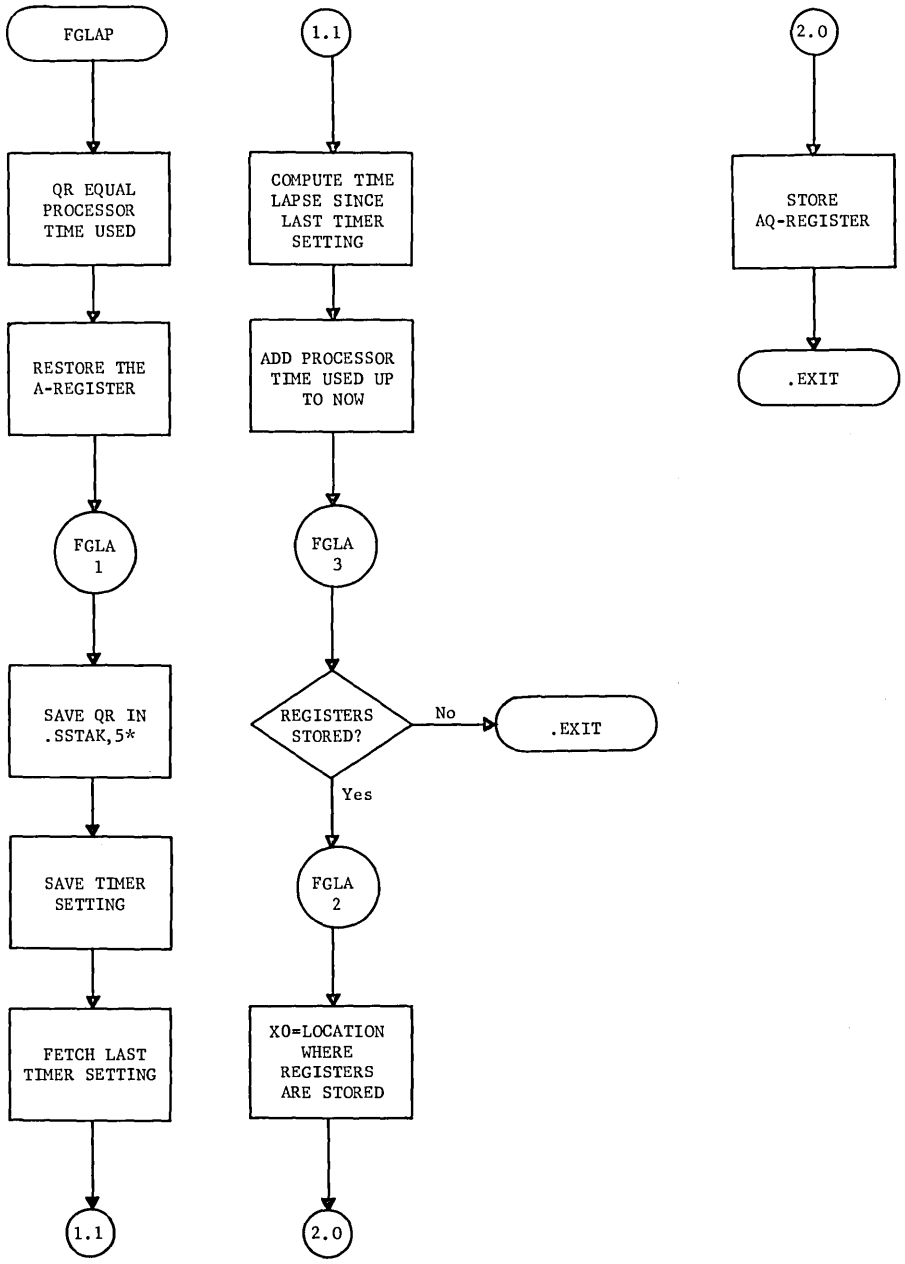


FGFIL (EP4)
.MFALT

SWITCH PRIMARY AND SECONDARY LOGICAL UNITS

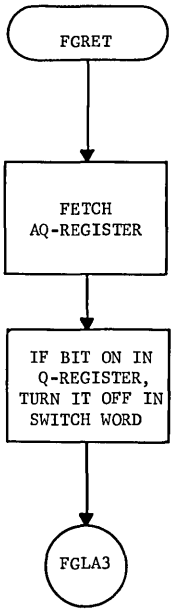


PROVIDE PROCESSOR TIME PRIOR TO REQUEST

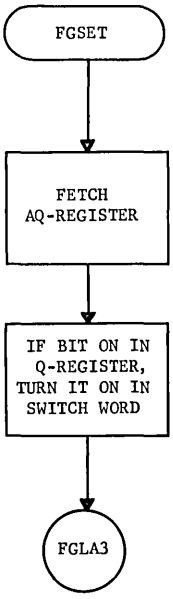


FGRET (EP6)
.MFALT

RESET PROGRAM SWITCH WORD BITS

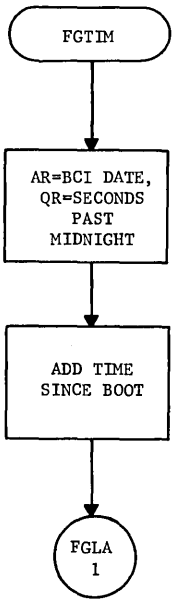


SET PROGRAM SWITCH WORD BITS



FGTIM (EP8)
.MFALT

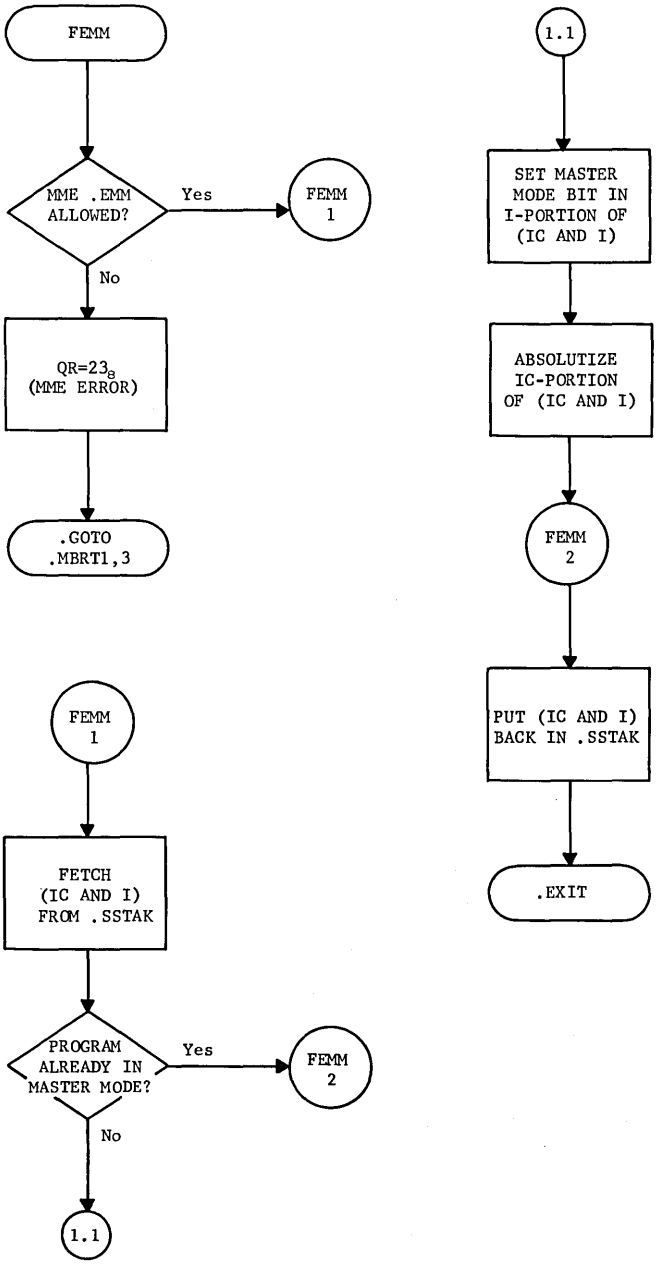
PROVIDE DATE AND TIME OF DAY



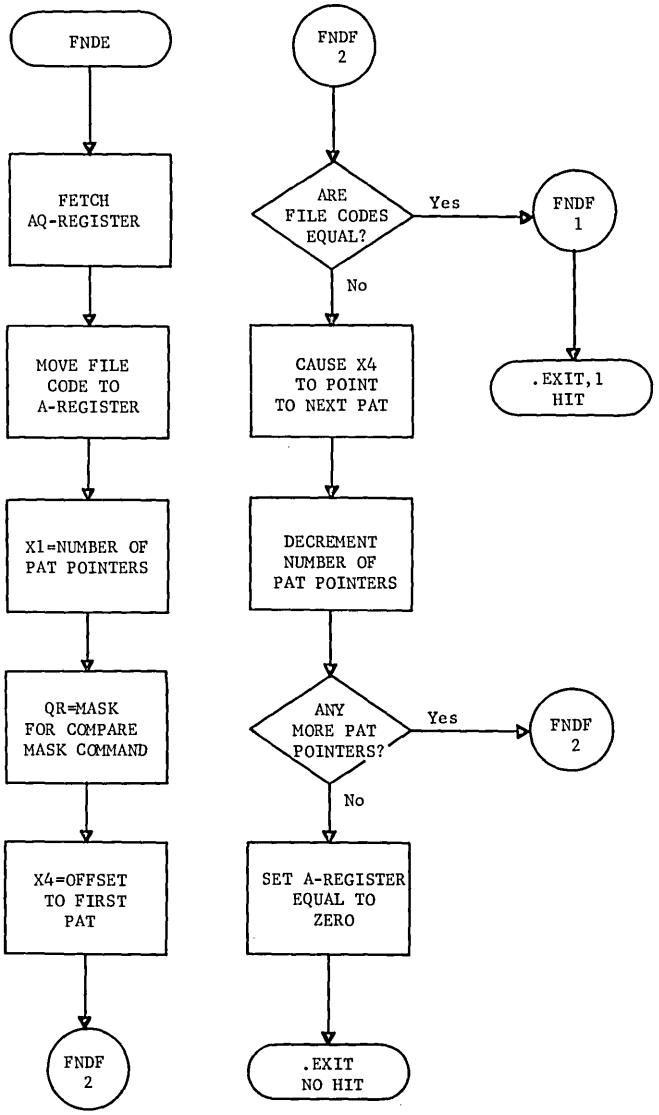
DIVIDE EXECUTION TIME INTO SECONDS

FEMM (EP10)
.MFALT

LET SLAVE PROGRAM ENTER MASTER MODE

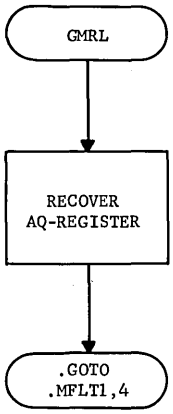


FIND PAT FROM FILE CODE

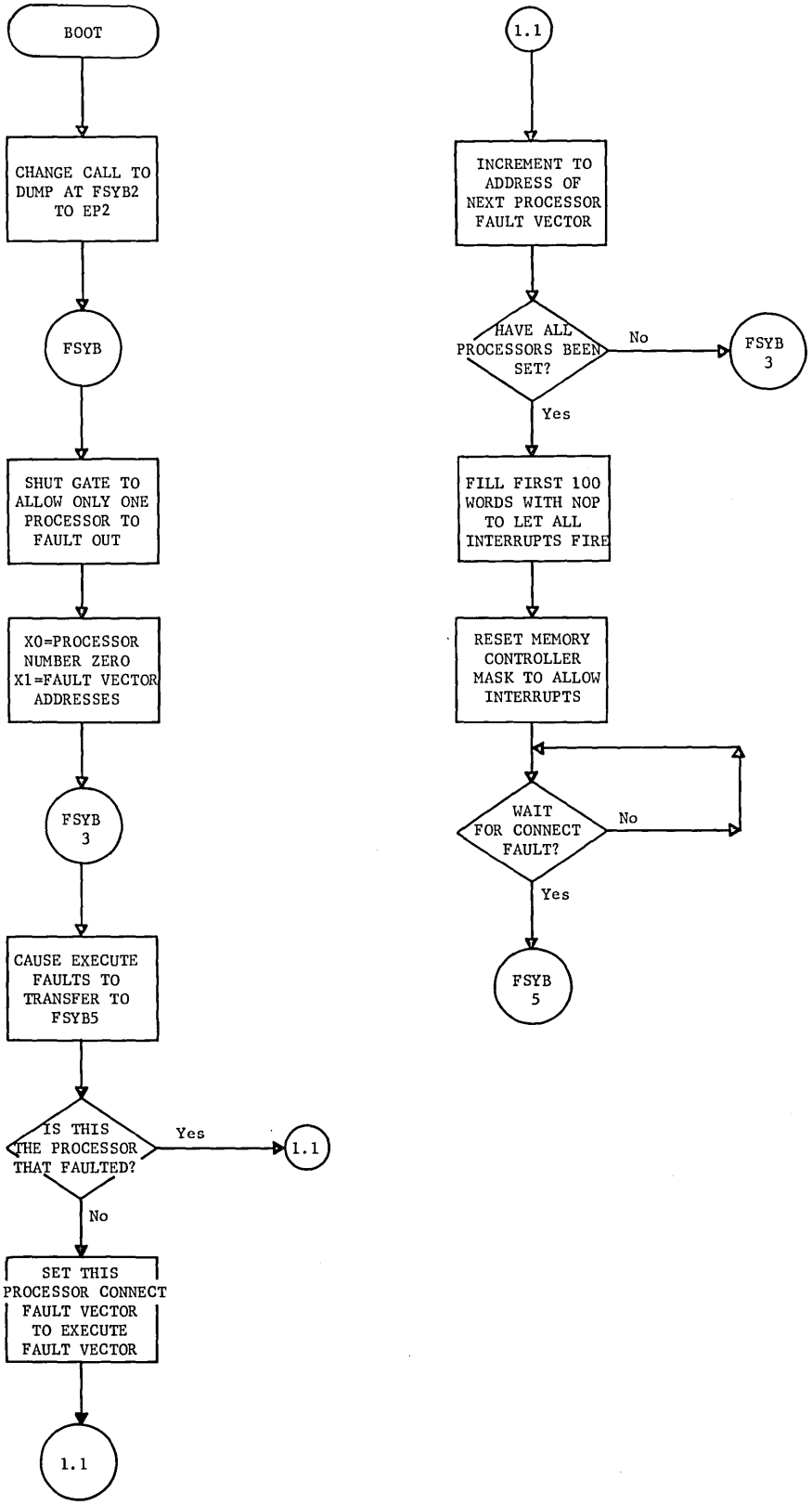


GMRL (EP12)
.MFALT

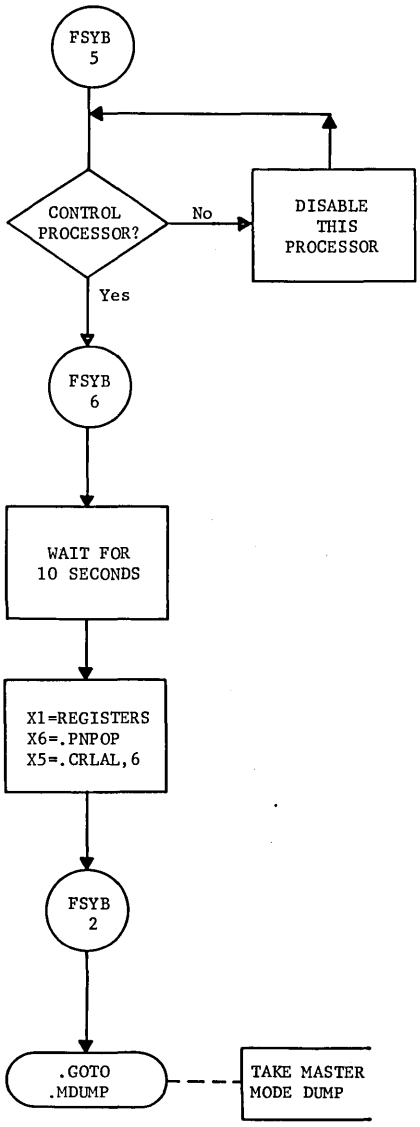
DEALLOCATE MEMORY



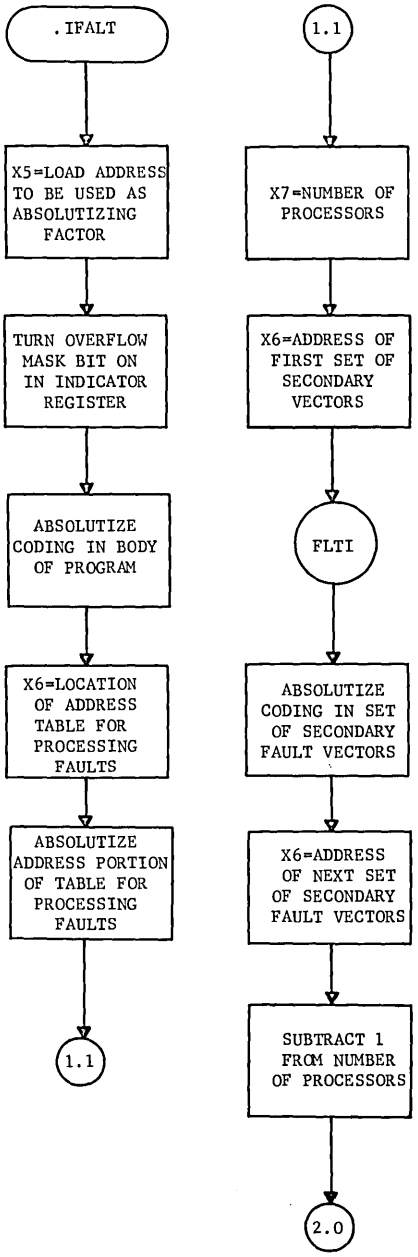
CALL .MDUMP AT EP2



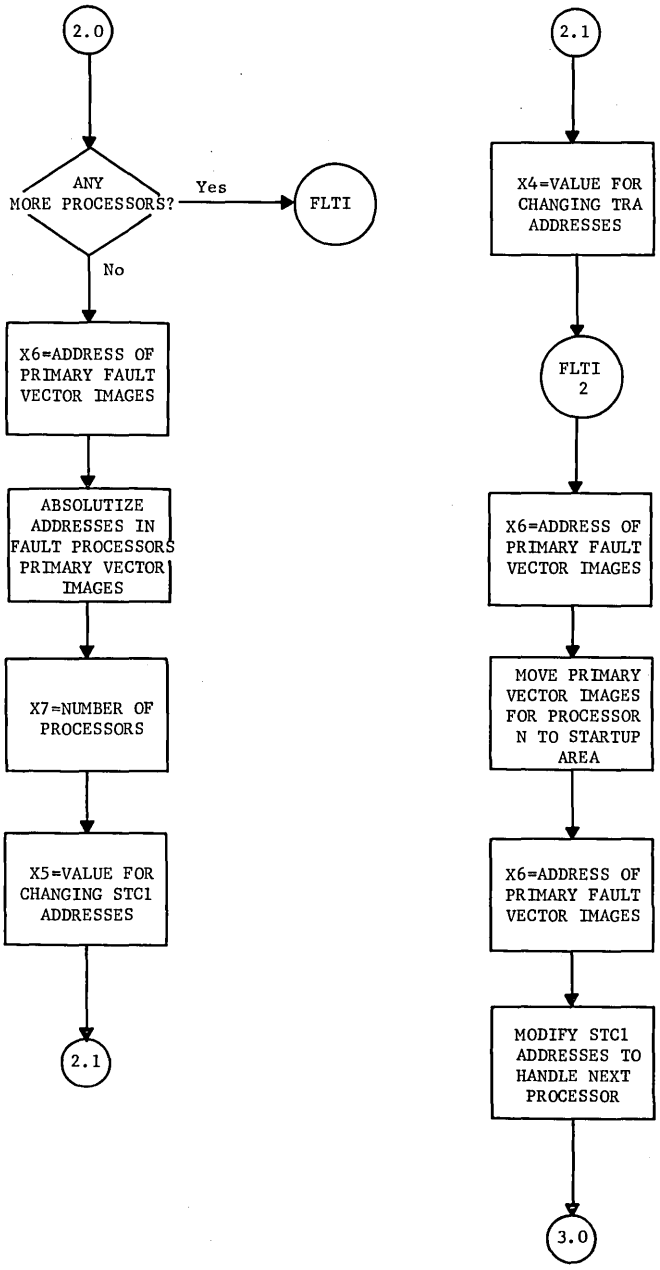
BOOT (EP13)
.MFAULT

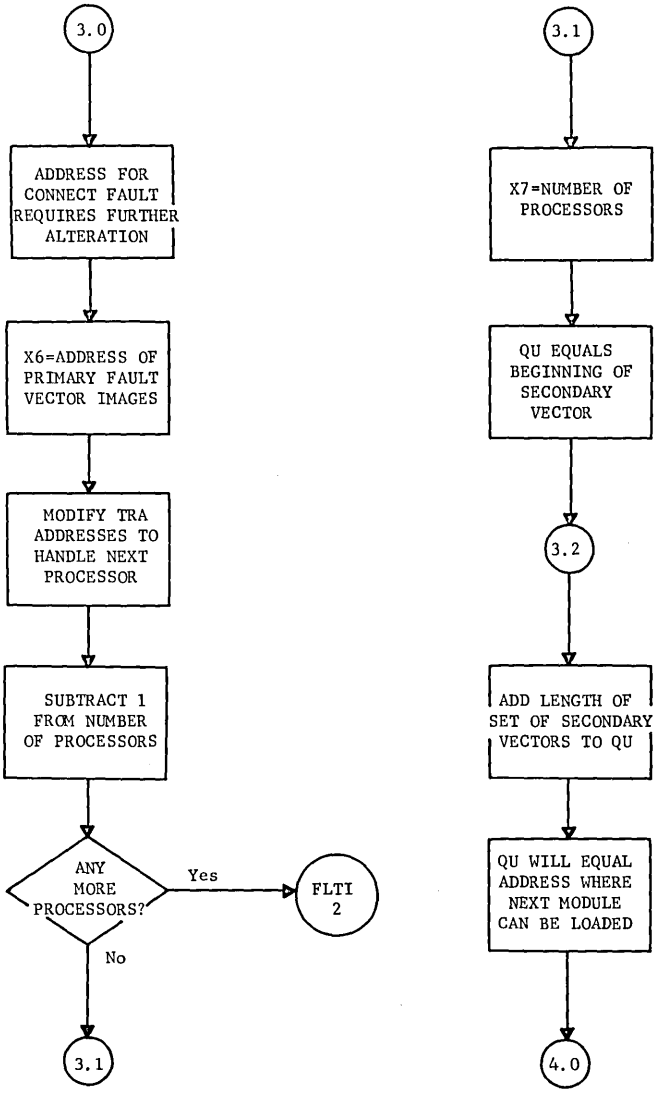


FAULT PROCESSOR INITIALIZATION

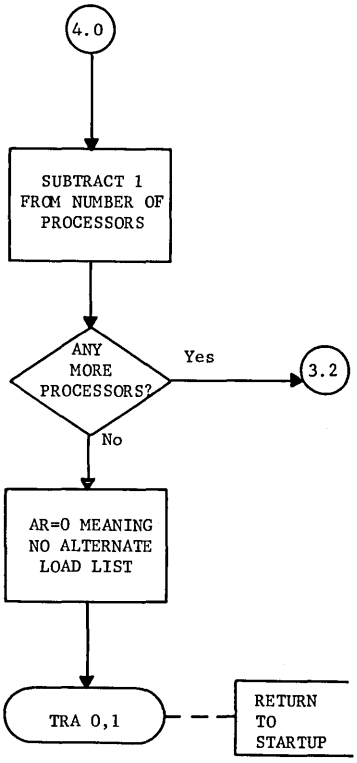


.IFALT
.MFALT

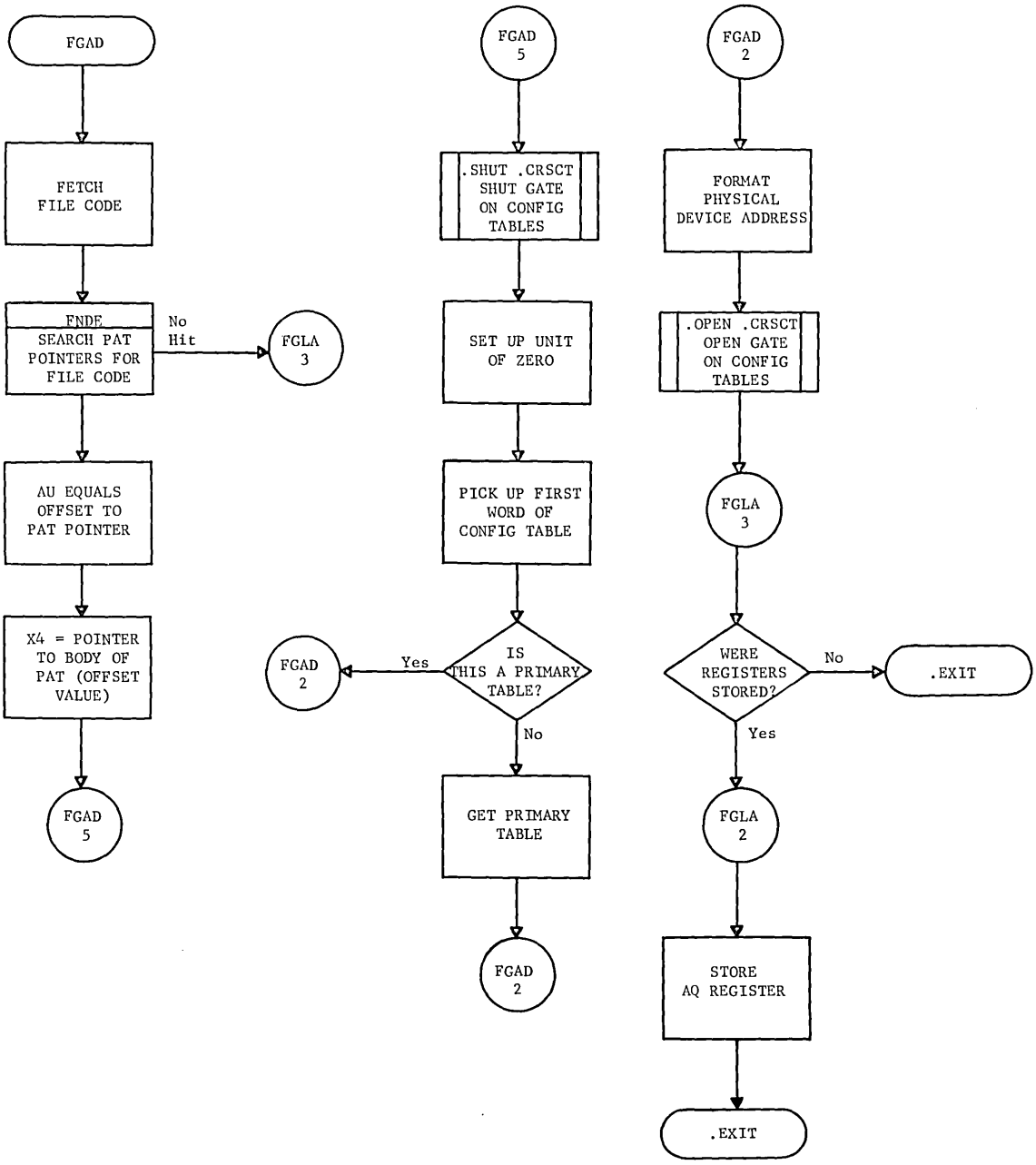

2.0
ANY MORE PROCESSORS?
Yes
FLTI
No
X6=ADDRESS OF PRIMARY FAULT VECTOR IMAGES
ABSOLUTIZE ADDRESSES IN FAULT PROCESSORS PRIMARY VECTOR IMAGES
X7=NUMBER OF PROCESSORS
X5=VALUE FOR CHANGING STC1 ADDRESSES
2.1
2.1
X4=VALUE FOR CHANGING TRA ADDRESSES
FLTI 2
X6=ADDRESS OF PRIMARY FAULT VECTOR IMAGES
MOVE PRIMARY VECTOR IMAGES FOR PROCESSOR N TO STARTUP AREA
X6=ADDRESS OF PRIMARY FAULT VECTOR IMAGES
MODIFY STC1 ADDRESSES TO HANDLE NEXT PROCESSOR
3.0


3.0
ADDRESS FOR CONNECT FAULT REQUIRES FURTHER ALTERATION
X6=ADDRESS OF PRIMARY FAULT VECTOR IMAGES
MODIFY TRA ADDRESSES TO HANDLE NEXT PROCESSOR
SUBTRACT 1 FROM NUMBER OF PROCESSORS
ANY MORE PROCESSORS?
Yes
FLTI 2
No
3.1
3.1
X7=NUMBER OF PROCESSORS
QU EQUALS BEGINNING OF SECONDARY VECTOR
3.2
ADD LENGTH OF SET OF SECONDARY VECTORS TO QU
QU WILL EQUAL ADDRESS WHERE NEXT MODULE CAN BE LOADED
4.0

.IFALT
.MFALT

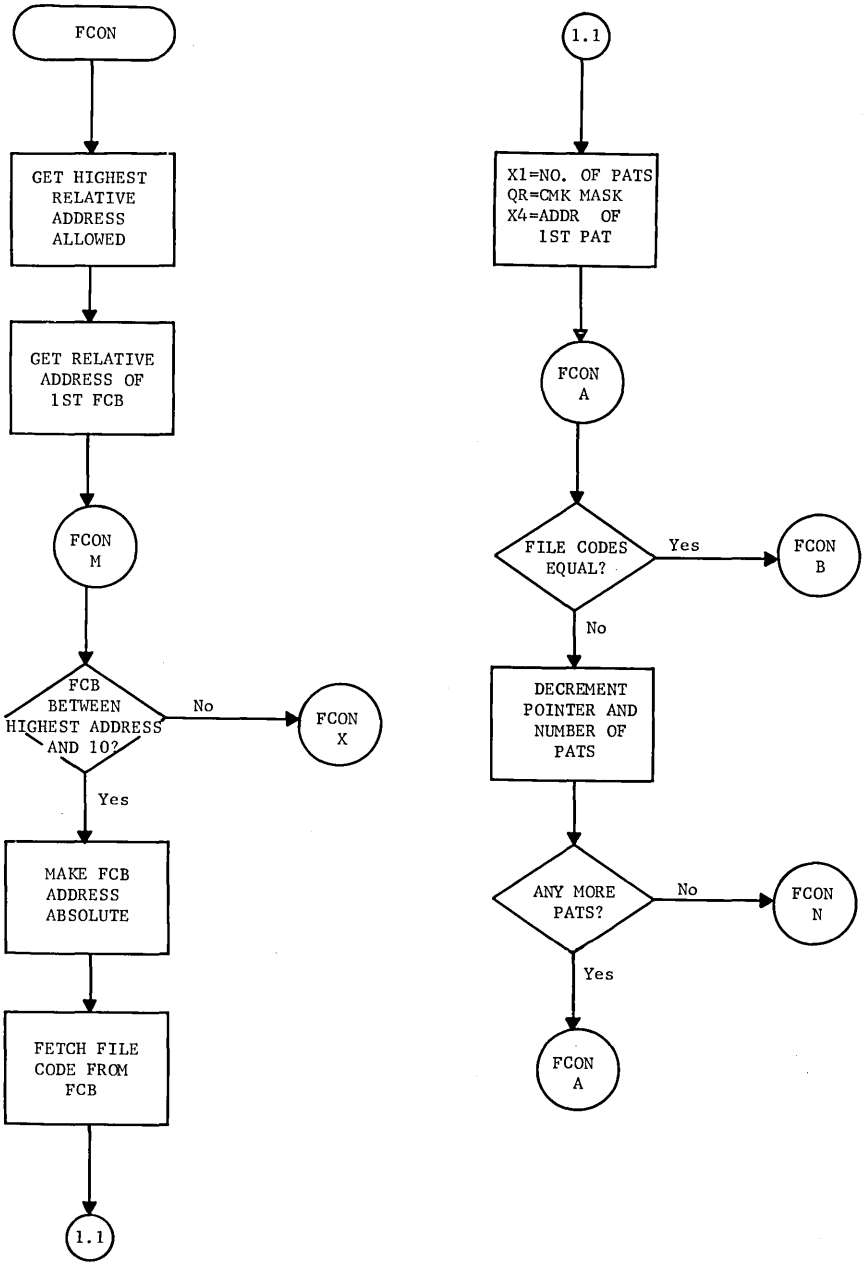

4.0
SUBTRACT 1 FROM NUMBER OF PROCESSORS
ANY MORE PROCESSORS?
Yes
3.2
No
AR=0 MEANING NO ALTERNATE LOAD LIST
TRA 0,1
RETURN TO STARTUP

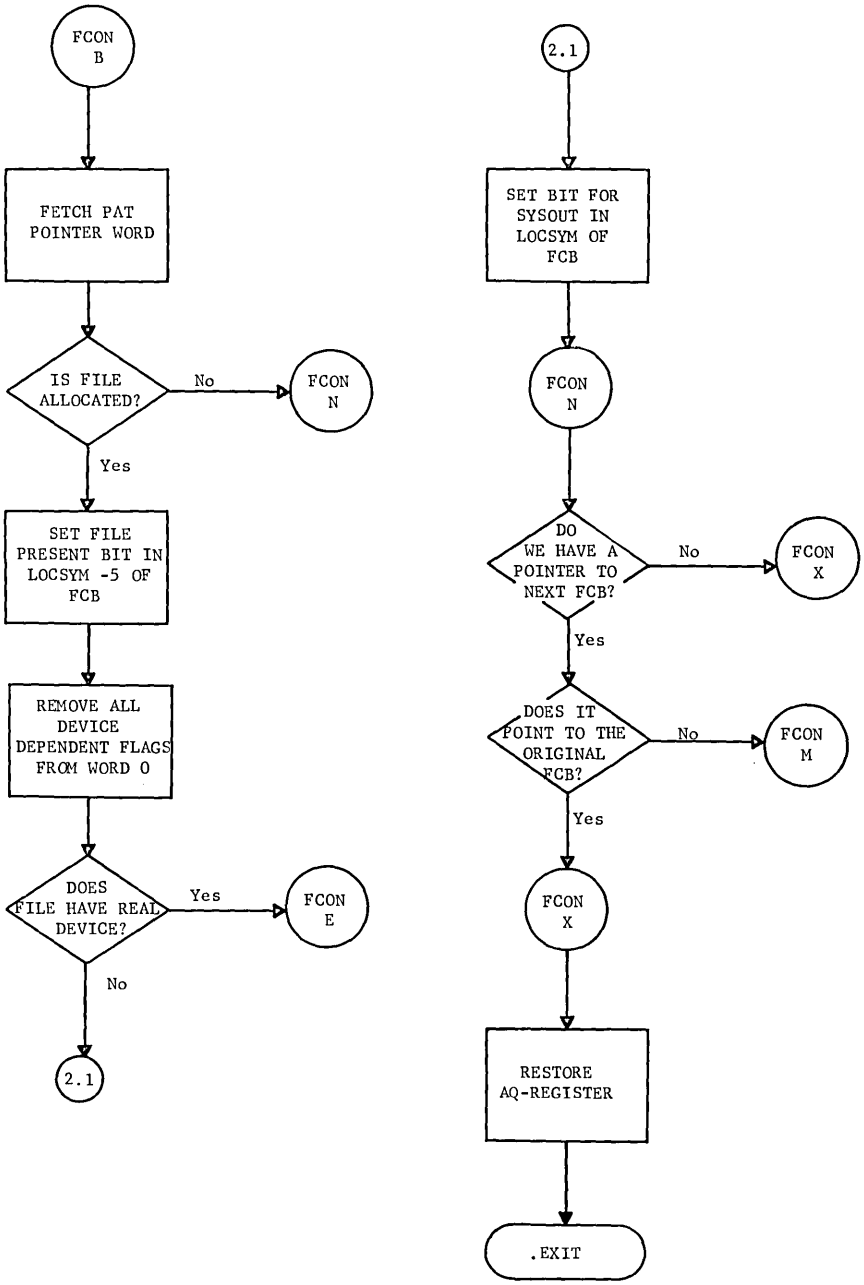
GET ADDRESS FOR FILE CODE

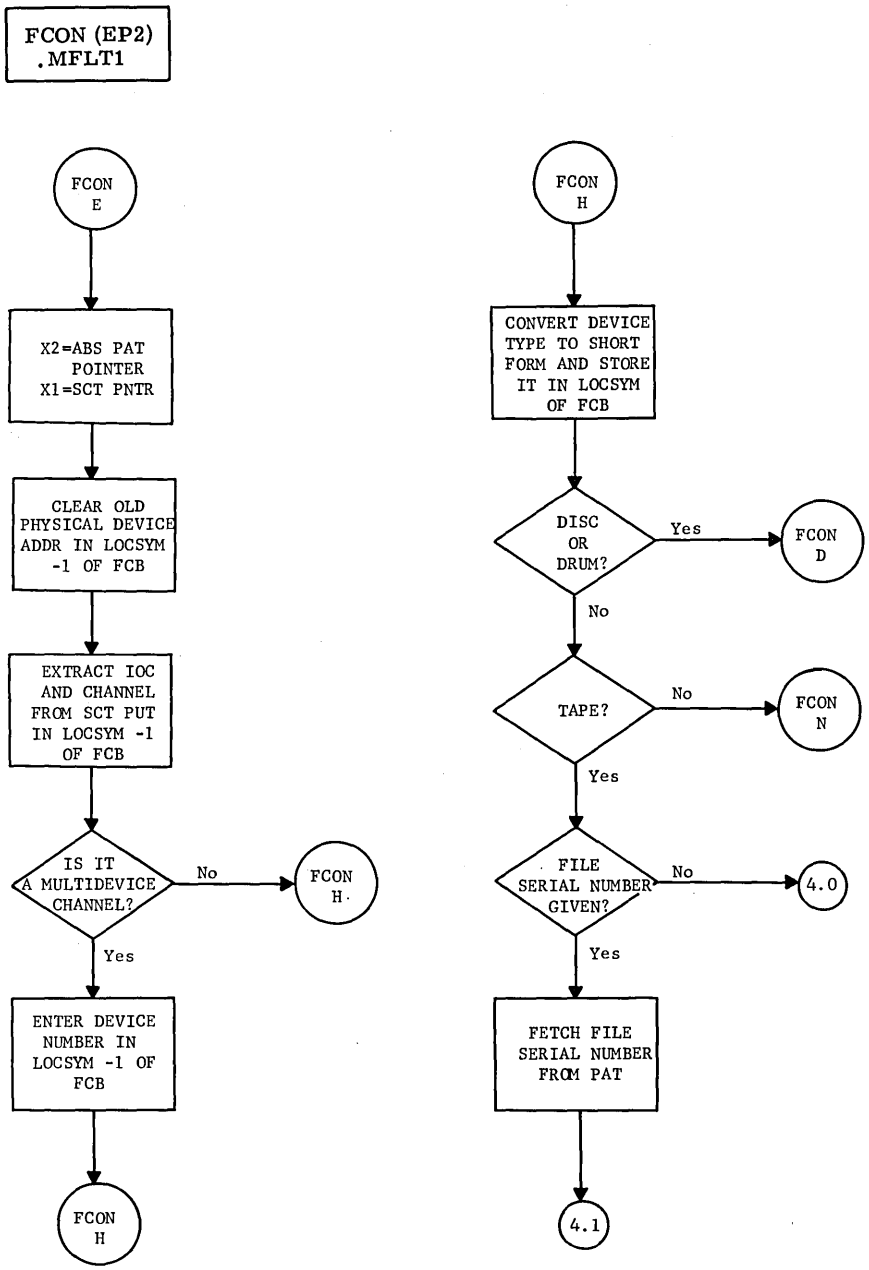


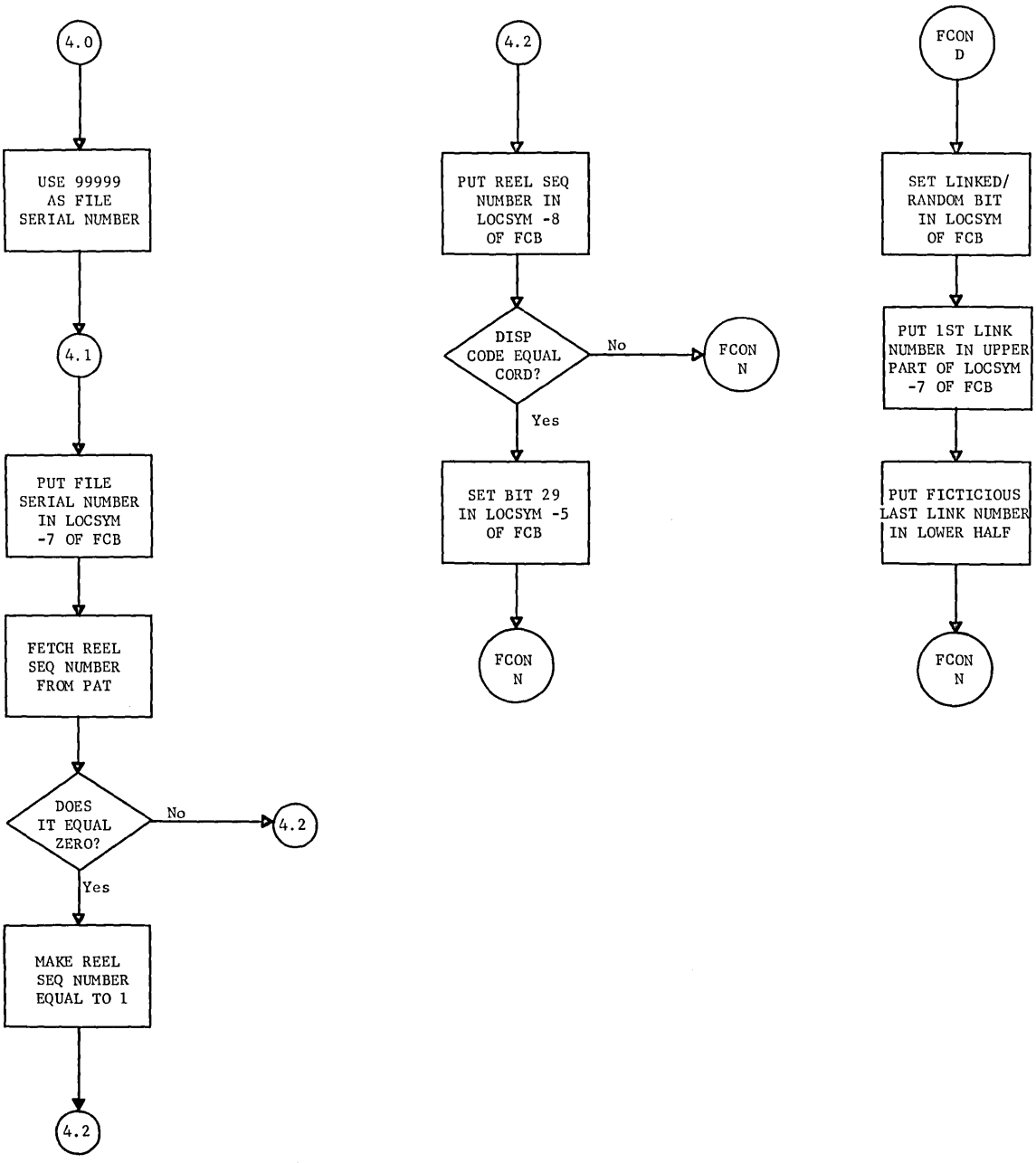
FCON (EP2)
.MFLT1

PUT INFORMATION IN FCB



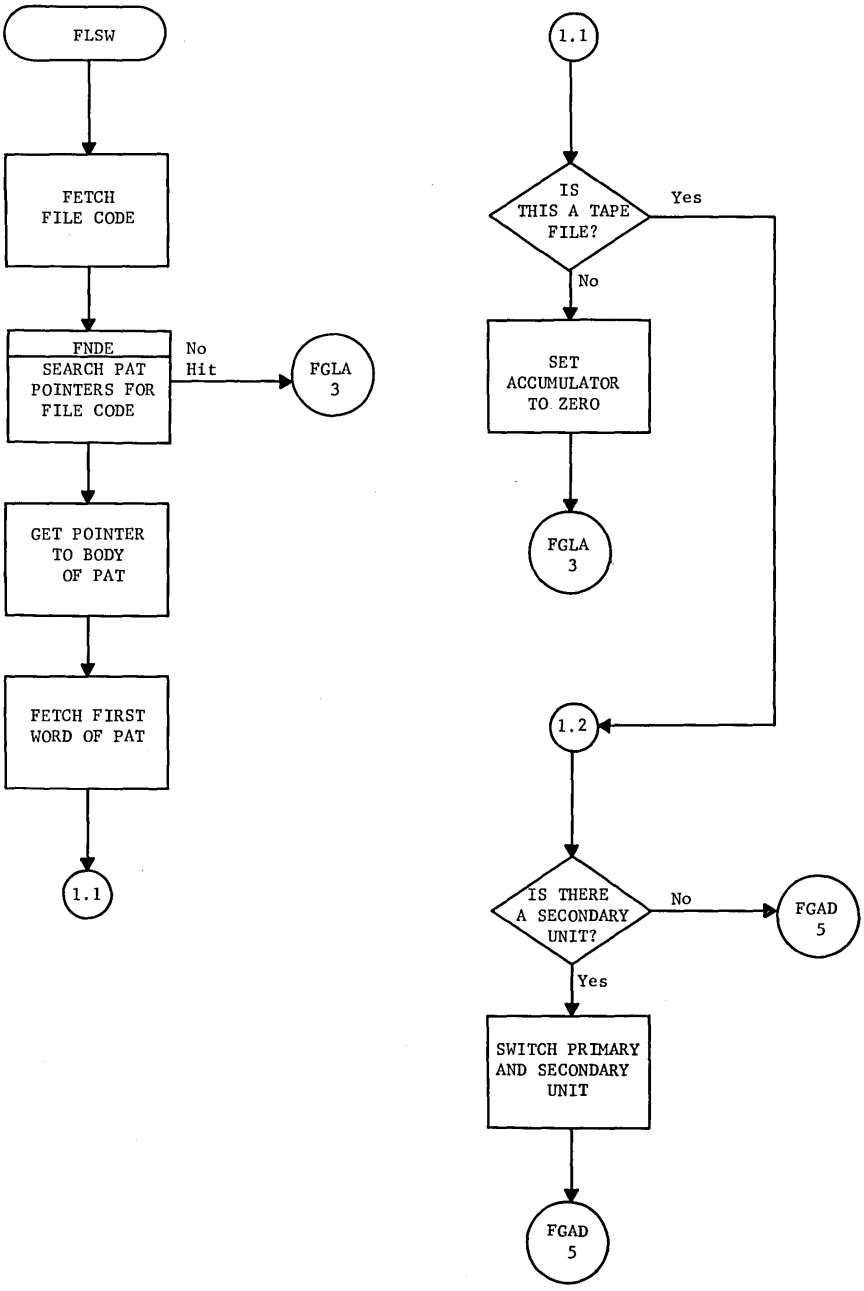

FCON B
FETCH PAT POINTER WORD
IS FILE ALLOCATED?
No
FCON N
Yes
SET FILE PRESENT BIT IN LOCSYM -5 OF FCB
REMOVE ALL DEVICE DEPENDENT FLAGS FROM WORD 0
DOES FILE HAVE REAL DEVICE?
Yes
FCON E
No
2.1
2.1
SET BIT FOR SYSOUT IN LOCSYM OF FCB
FCON N
DO WE HAVE A POINTER TO NEXT FCB?
No
FCON X
Yes
DOES IT POINT TO THE ORIGINAL FCB?
No
FCON M
Yes
FCON X
RESTORE AQ-REGISTER
.EXIT


FCON (EP2)
.MFLT1
FCON
E
X2=ABS PAT
POINTER
X1=SCT PNTR
CLEAR OLD
PHYSICAL DEVICE
ADDR IN LOCSYM
-1 OF FCB
EXTRACT IOC
AND CHANNEL
FROM SCT PUT
IN LOCSYM -1
OF FCB
IS IT
A MULTIDEVICE
CHANNEL?
No
FCON
H
Yes
ENTER DEVICE
NUMBER IN
LOCSYM -1 OF
FCB
FCON
H
FCON
H
CONVERT DEVICE
TYPE TO SHORT
FORM AND STORE
IT IN LOCSYM
OF FCB
DISC
OR
DRUM?
Yes
FCON
D
No
TAPE?
No
FCON
N
Yes
FILE
SERIAL NUMBER
GIVEN?
No
4.0
Yes
FETCH FILE
SERIAL NUMBER
FROM PAT
4.1

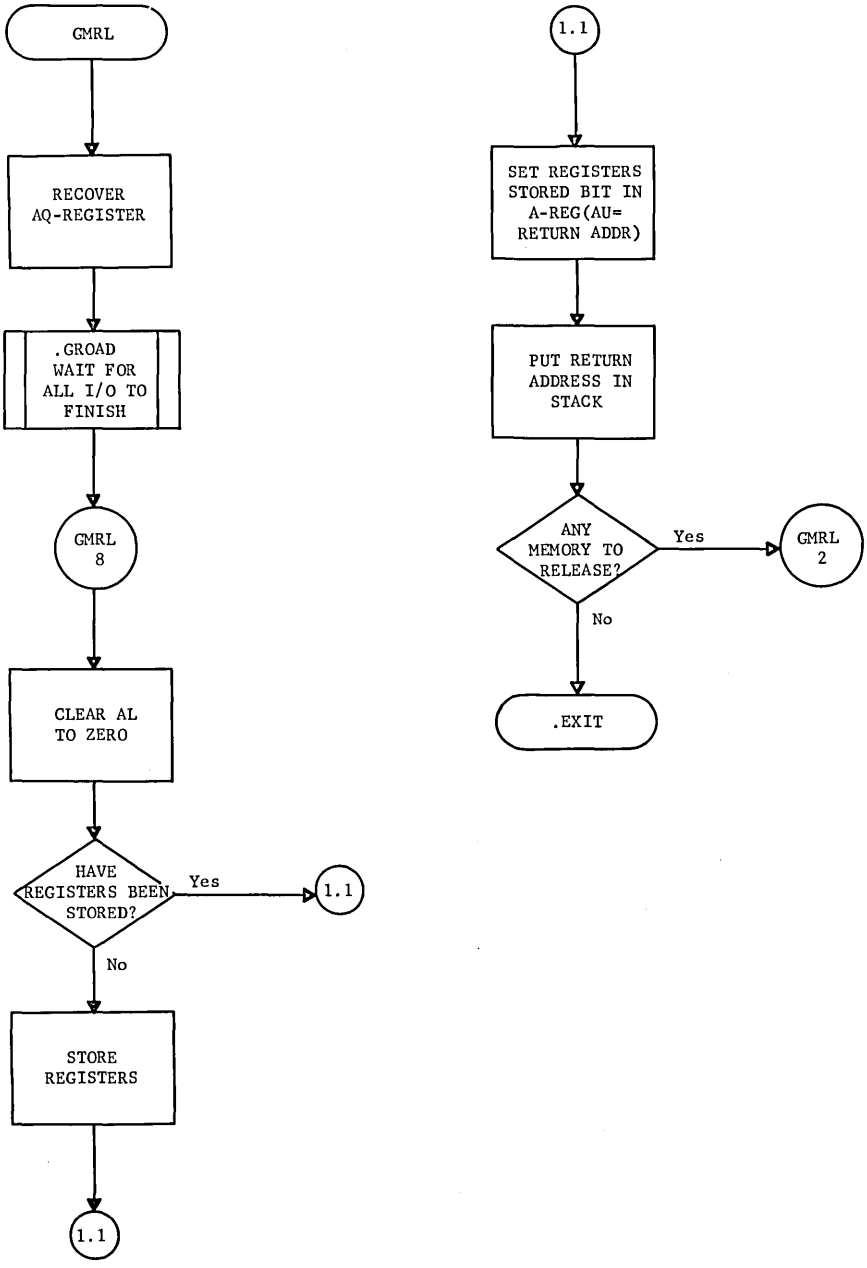

4.0
USE 99999 AS FILE SERIAL NUMBER
4.1
PUT FILE SERIAL NUMBER IN LOCSYM -7 OF FCB
FETCH REEL SEQ NUMBER FROM PAT
DOES IT EQUAL ZERO?
No
4.2
Yes
MAKE REEL SEQ NUMBER EQUAL TO 1
4.2
4.2
PUT REEL SEQ NUMBER IN LOCSYM -8 OF FCB
DISP CODE EQUAL CORD?
No
FCON N
Yes
SET BIT 29 IN LOCSYM -5 OF FCB
FCON N
FCON D
SET LINKED/ RANDOM BIT IN LOCSYM OF FCB
PUT 1ST LINK NUMBER IN UPPER PART OF LOCSYM -7 OF FCB
PUT FICTICIOUS LAST LINK NUMBER IN LOWER HALF
FCON N

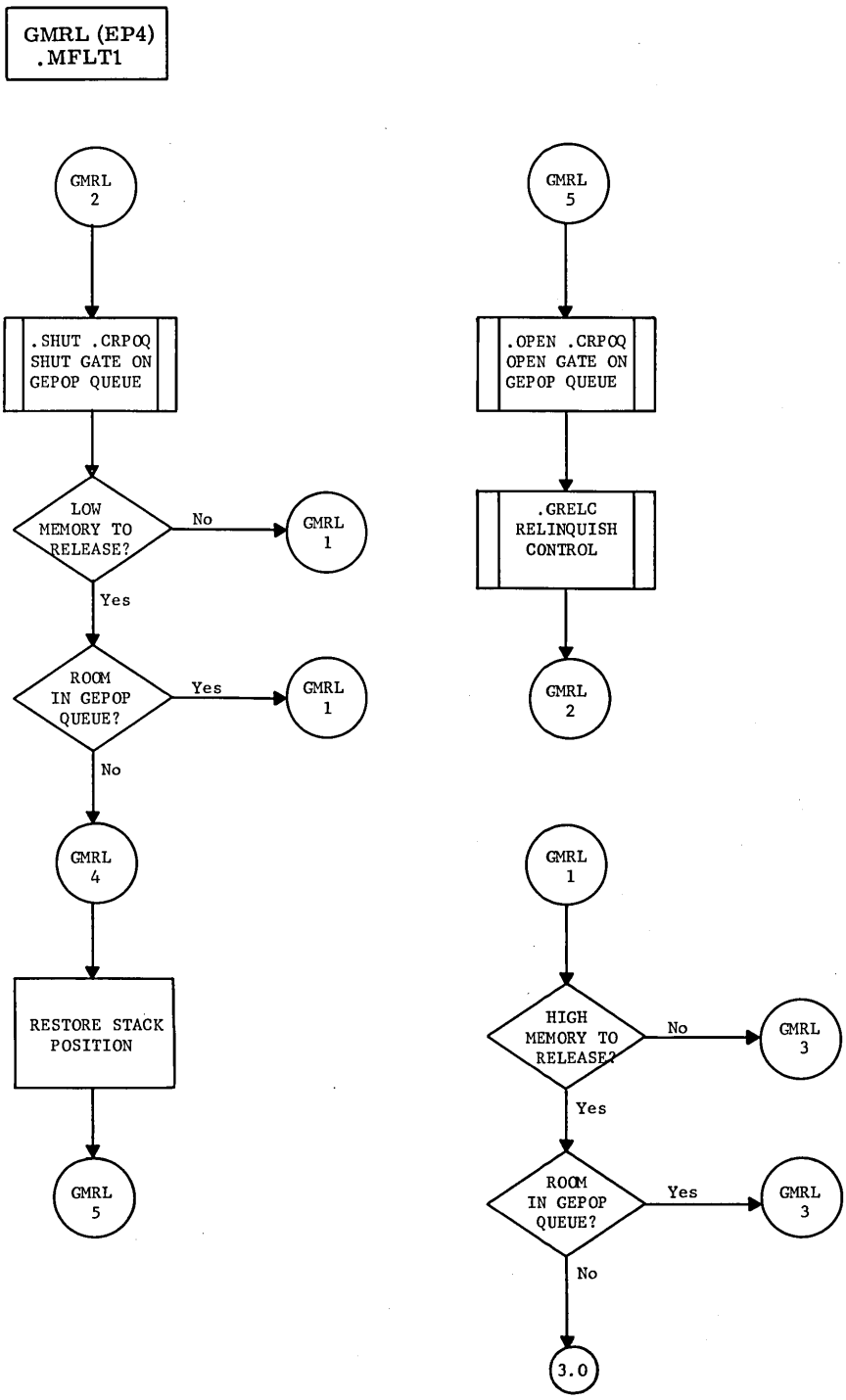
FLSW (EP3)
.MFLT1

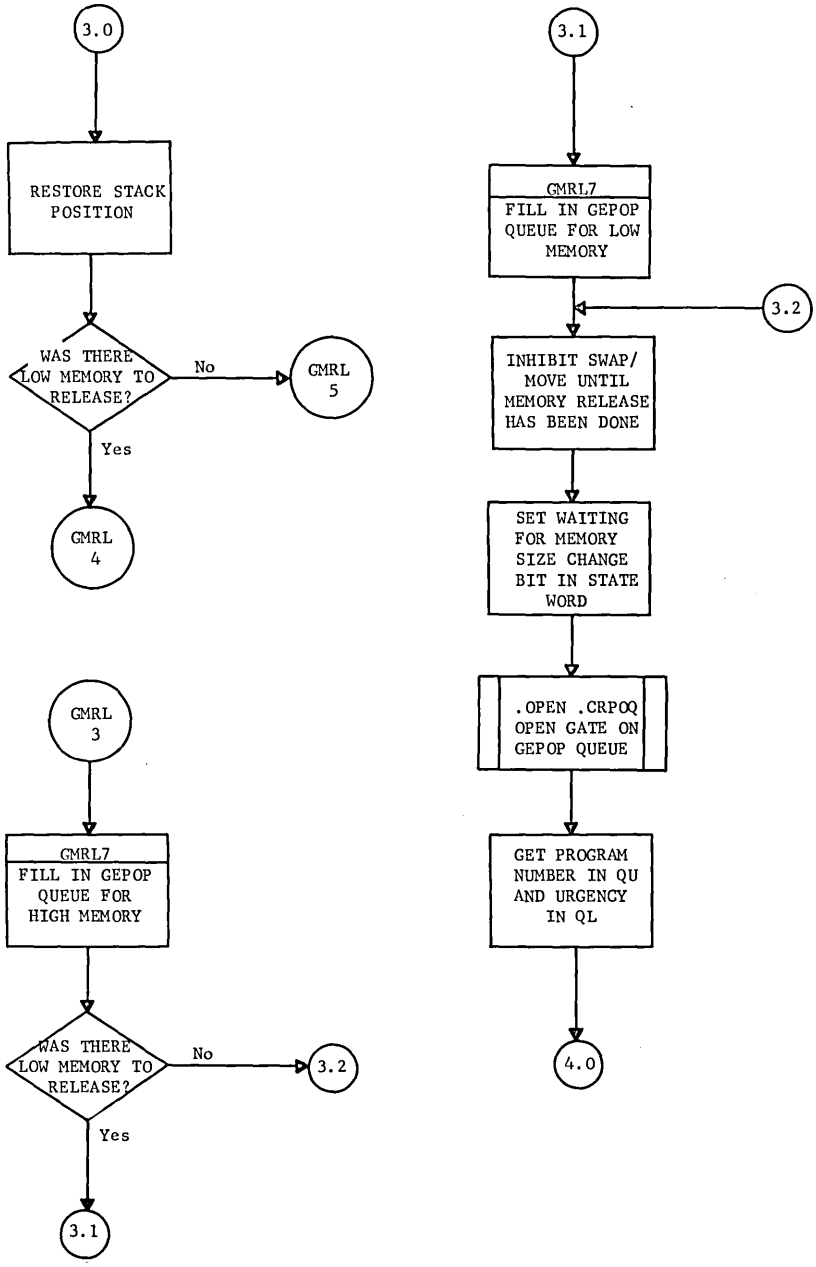
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS

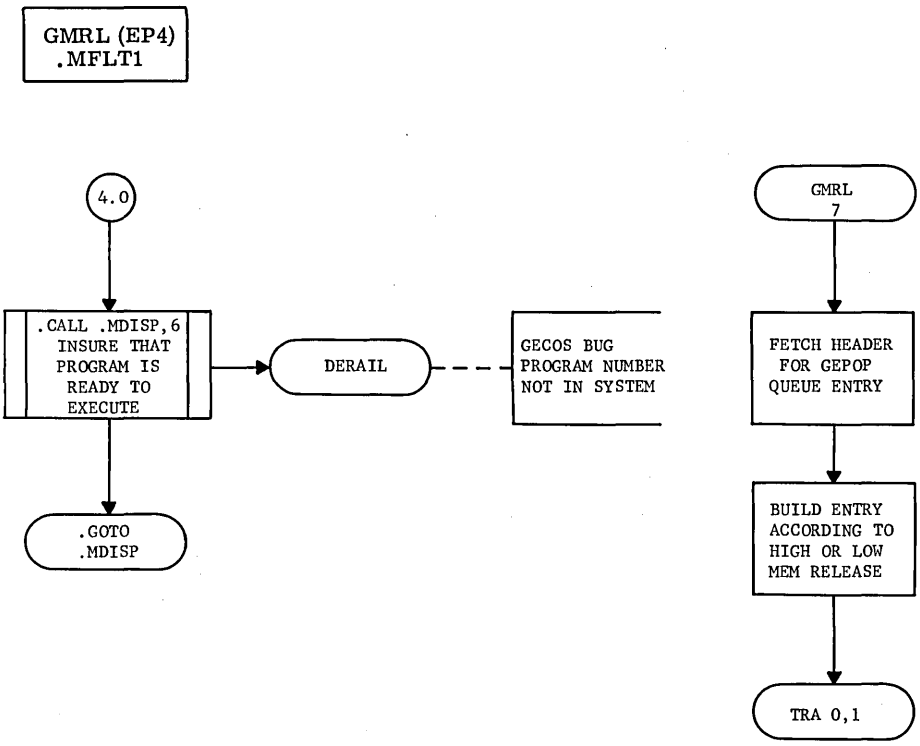


DEALLOCATE MEMORY

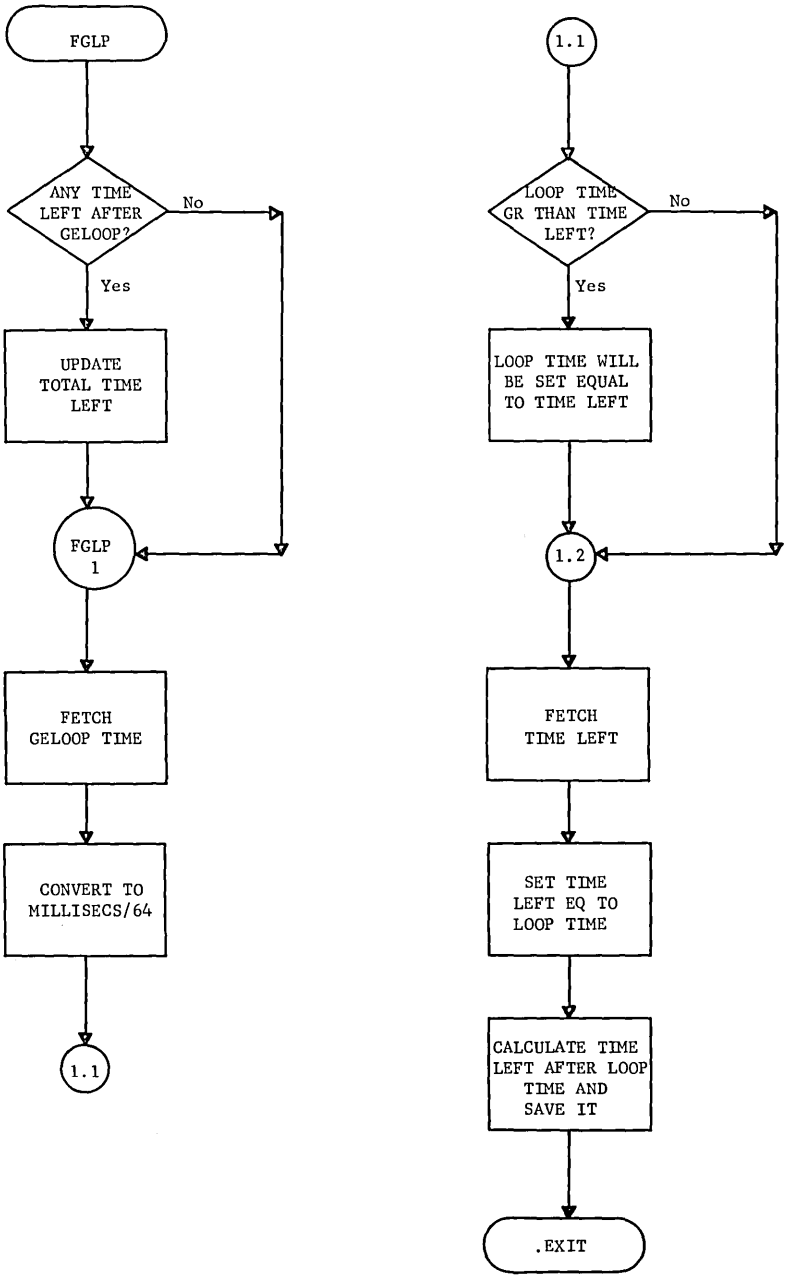



GMRL (EP4)
.MFLT1
GMRL 2
.SHUT .CRPOQ SHUT GATE ON GEPOP QUEUE
LOW MEMORY TO RELEASE?
No
GMRL 1
Yes
ROOM IN GEPOP QUEUE?
Yes
GMRL 1
No
GMRL 4
RESTORE STACK POSITION
GMRL 5
GMRL 5
.OPEN .CRPOQ OPEN GATE ON GEPOP QUEUE
.GRELC RELINQUISH CONTROL
GMRL 2
GMRL 1
HIGH MEMORY TO RELEASE?
No
GMRL 3
Yes
ROOM IN GEPOP QUEUE?
Yes
GMRL 3
No
3.0


3.0
RESTORE STACK POSITION
WAS THERE LOW MEMORY TO RELEASE?
No
GMRL 5
Yes
GMRL 4
GMRL 3
GMRL7
FILL IN GEPOP QUEUE FOR HIGH MEMORY
WAS THERE LOW MEMORY TO RELEASE?
No
3.2
Yes
3.1
3.1
GMRL7
FILL IN GEPOP QUEUE FOR LOW MEMORY
3.2
INHIBIT SWAP/ MOVE UNTIL MEMORY RELEASE HAS BEEN DONE
SET WAITING FOR MEMORY SIZE CHANGE BIT IN STATE WORD
.OPEN .CRPOQ OPEN GATE ON GEPOP QUEUE
GET PROGRAM NUMBER IN QU AND URGENCY IN QL
4.0

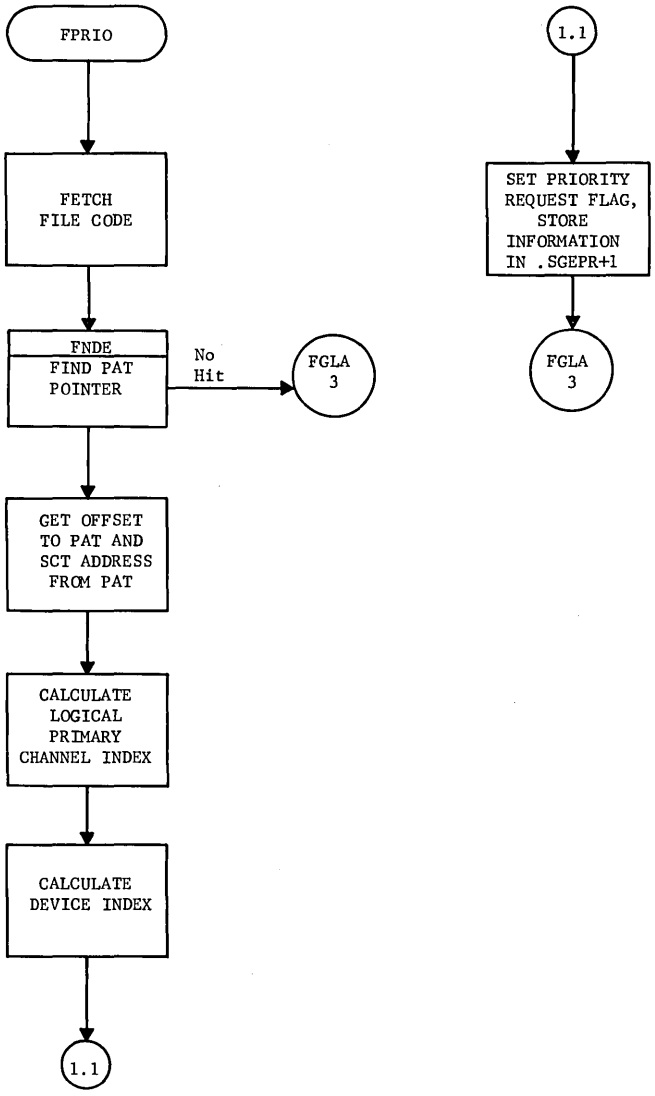

GMRL (EP4)
.MFLT1
4.0
.CALL .MDISP,6
INSURE THAT
PROGRAM IS
READY TO
EXECUTE
DERAIL
GECOS BUG
PROGRAM NUMBER
NOT IN SYSTEM
.GOTO
.MDISP
GMRL
7
FETCH HEADER
FOR GEPOP
QUEUE ENTRY
BUILD ENTRY
ACCORDING TO
HIGH OR LOW
MEM RELEASE
TRA 0,1

DIVIDE EXECUTION TIME INTO SECONDS

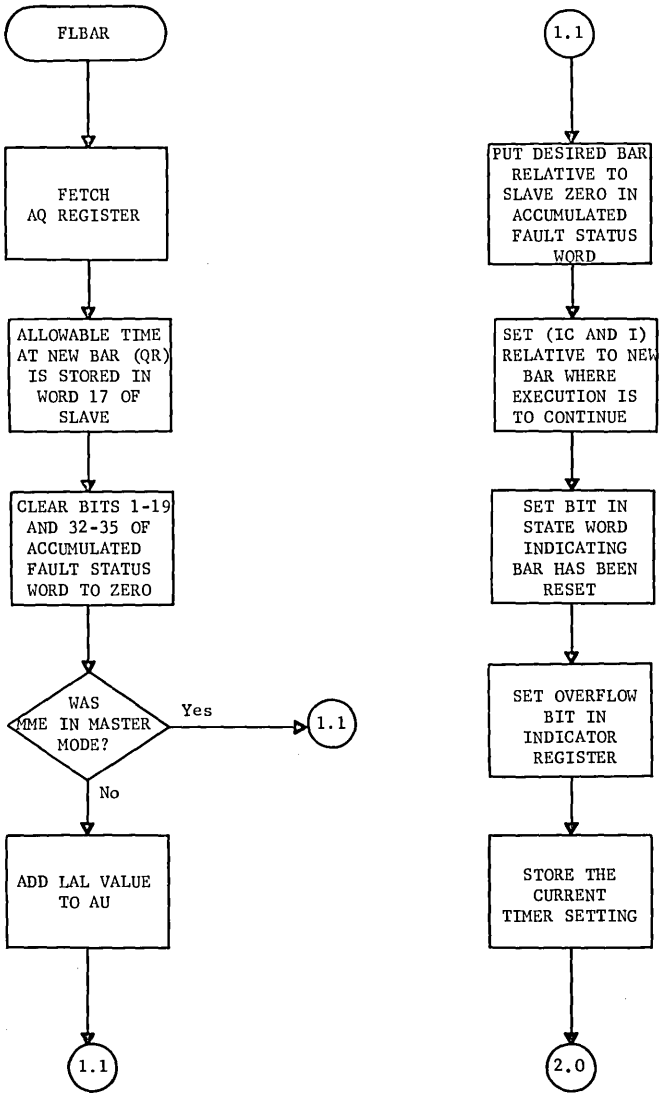


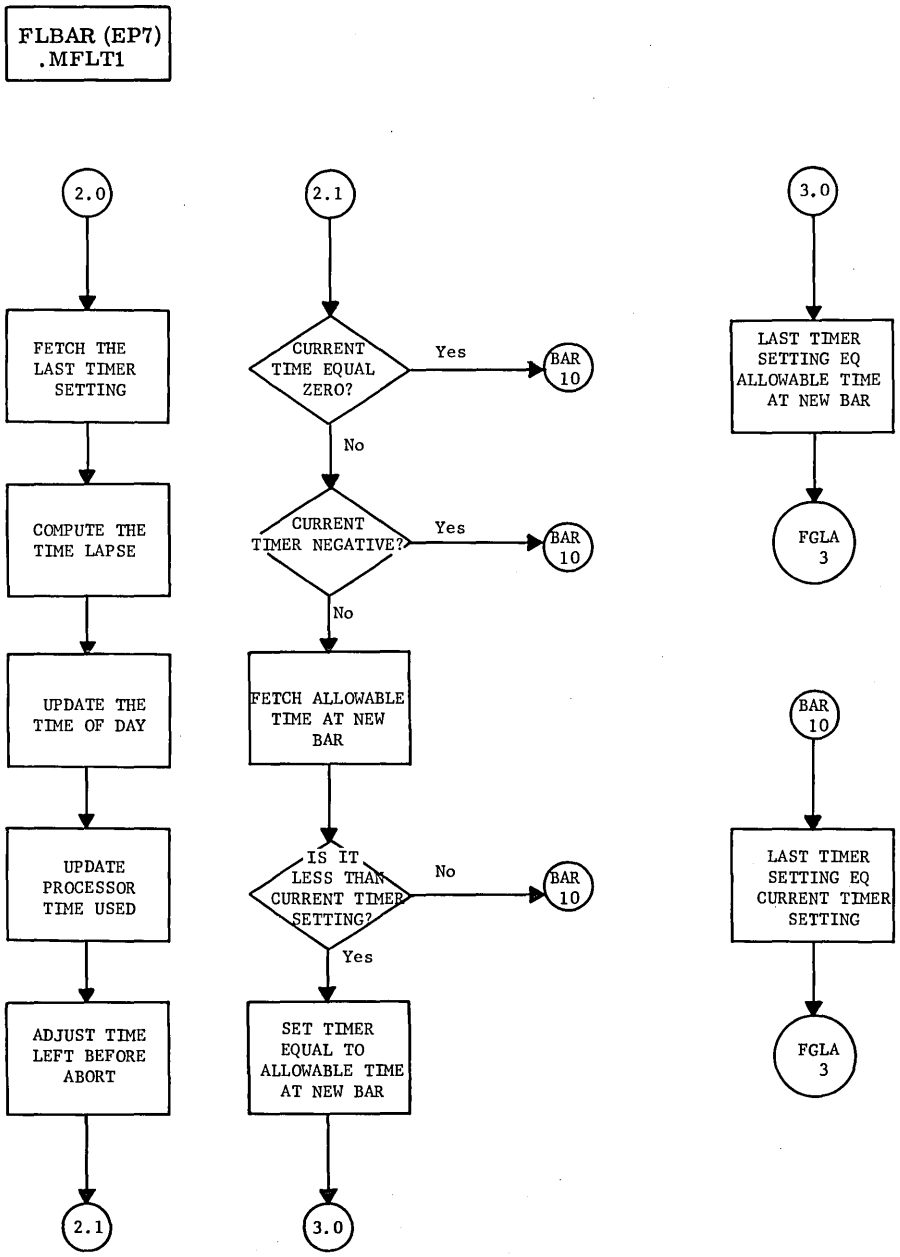
FPRIO (EP6)
.MFLT1

GIVE I/O ACTION PRIORITY

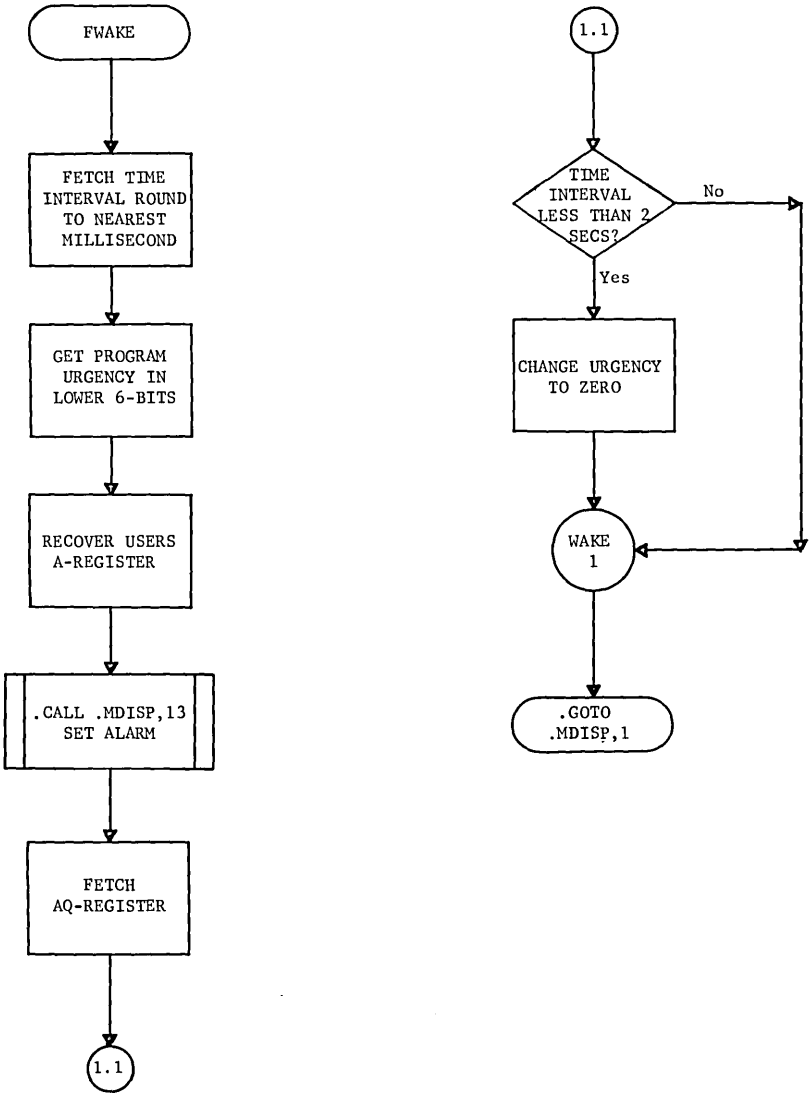


RESET BAR TO SMALLER AREA




FLBAR (EP7)
.MFLT1
2.0
FETCH THE LAST TIMER SETTING
COMPUTE THE TIME LAPSE
UPDATE THE TIME OF DAY
UPDATE PROCESSOR TIME USED
ADJUST TIME LEFT BEFORE ABORT
2.1
2.1
CURRENT TIME EQUAL ZERO?
Yes
BAR 10
No
CURRENT TIMER NEGATIVE?
Yes
BAR 10
No
FETCH ALLOWABLE TIME AT NEW BAR
IS IT LESS THAN CURRENT TIMER SETTING?
No
BAR 10
Yes
SET TIMER EQUAL TO ALLOWABLE TIME AT NEW BAR
3.0
3.0
LAST TIMER SETTING EQ ALLOWABLE TIME AT NEW BAR
FGLA 3
BAR 10
BAR 10
LAST TIMER SETTING EQ CURRENT TIMER SETTING
FGLA 3

ALLOW A PROGRAM TO DELAY



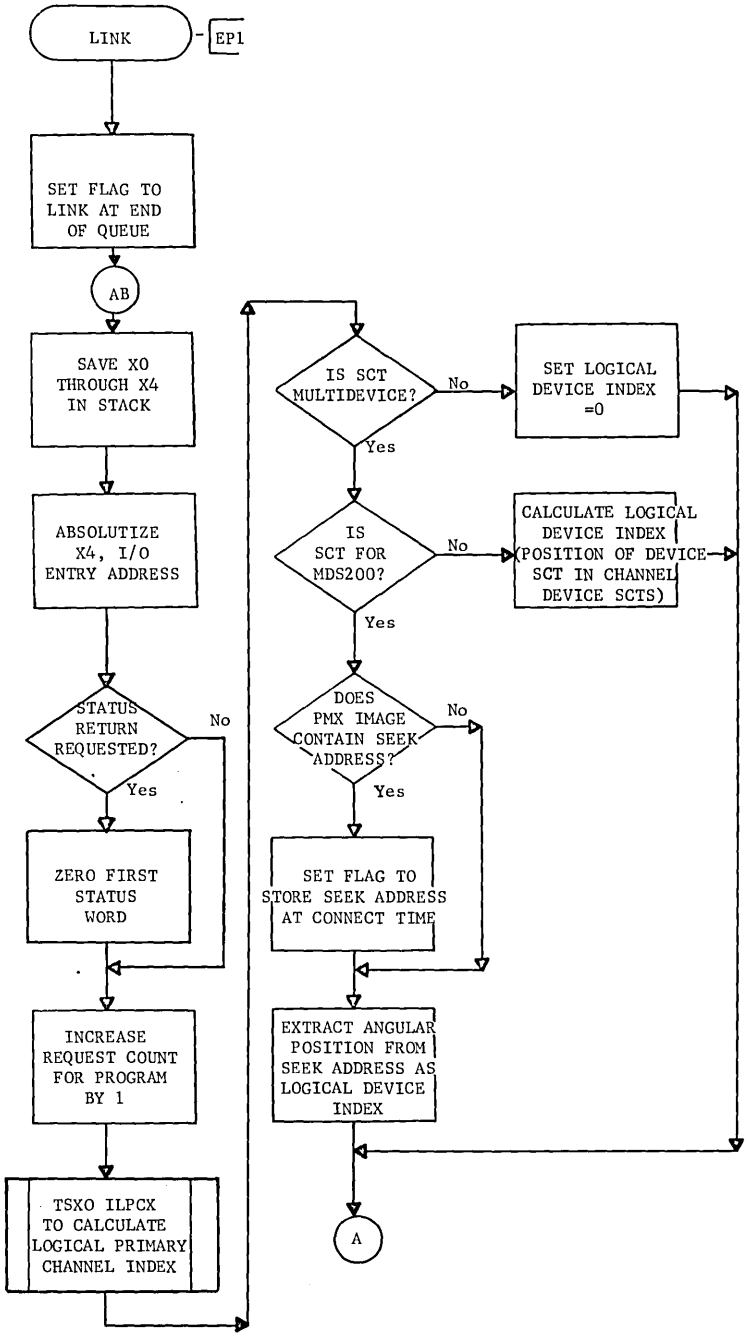
GMRLM (EP9)
.MFLT1

MEMORY RELEASE ENTRY FOR TIME-SHARING

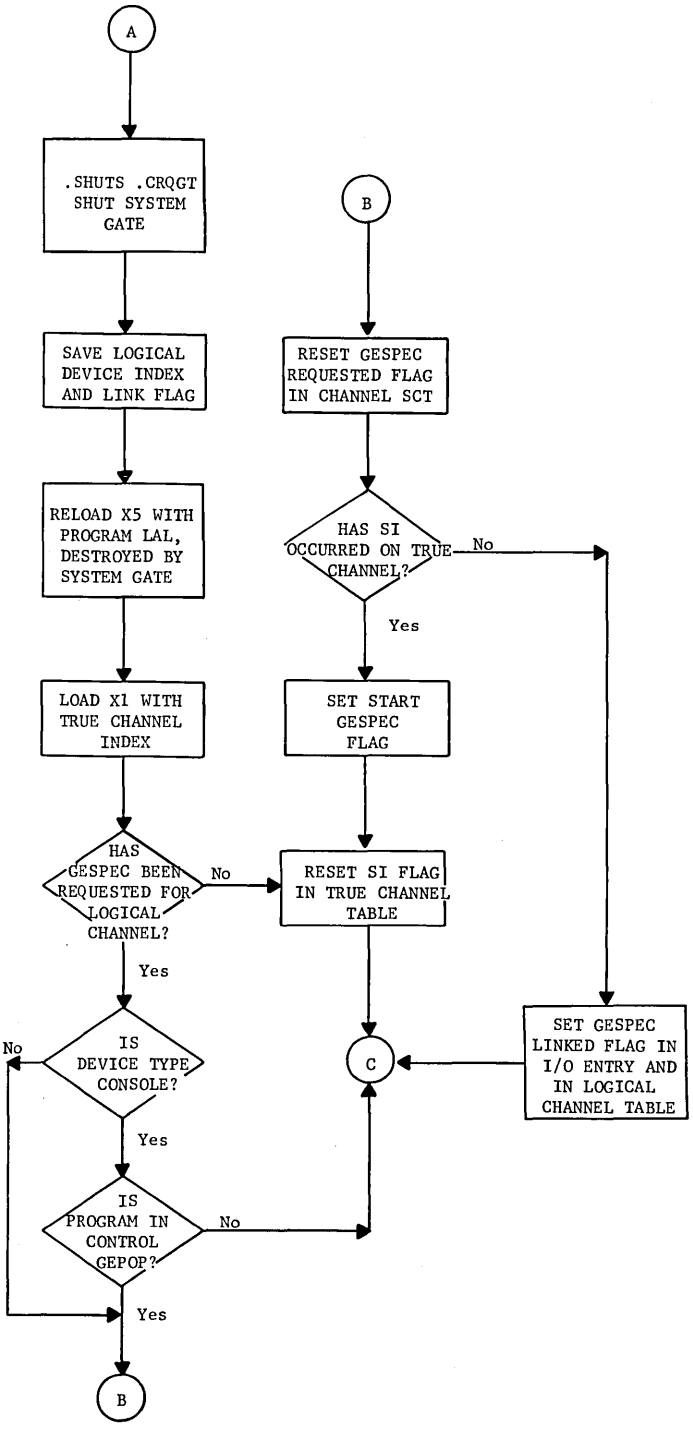
3. I/O SUPERVISION

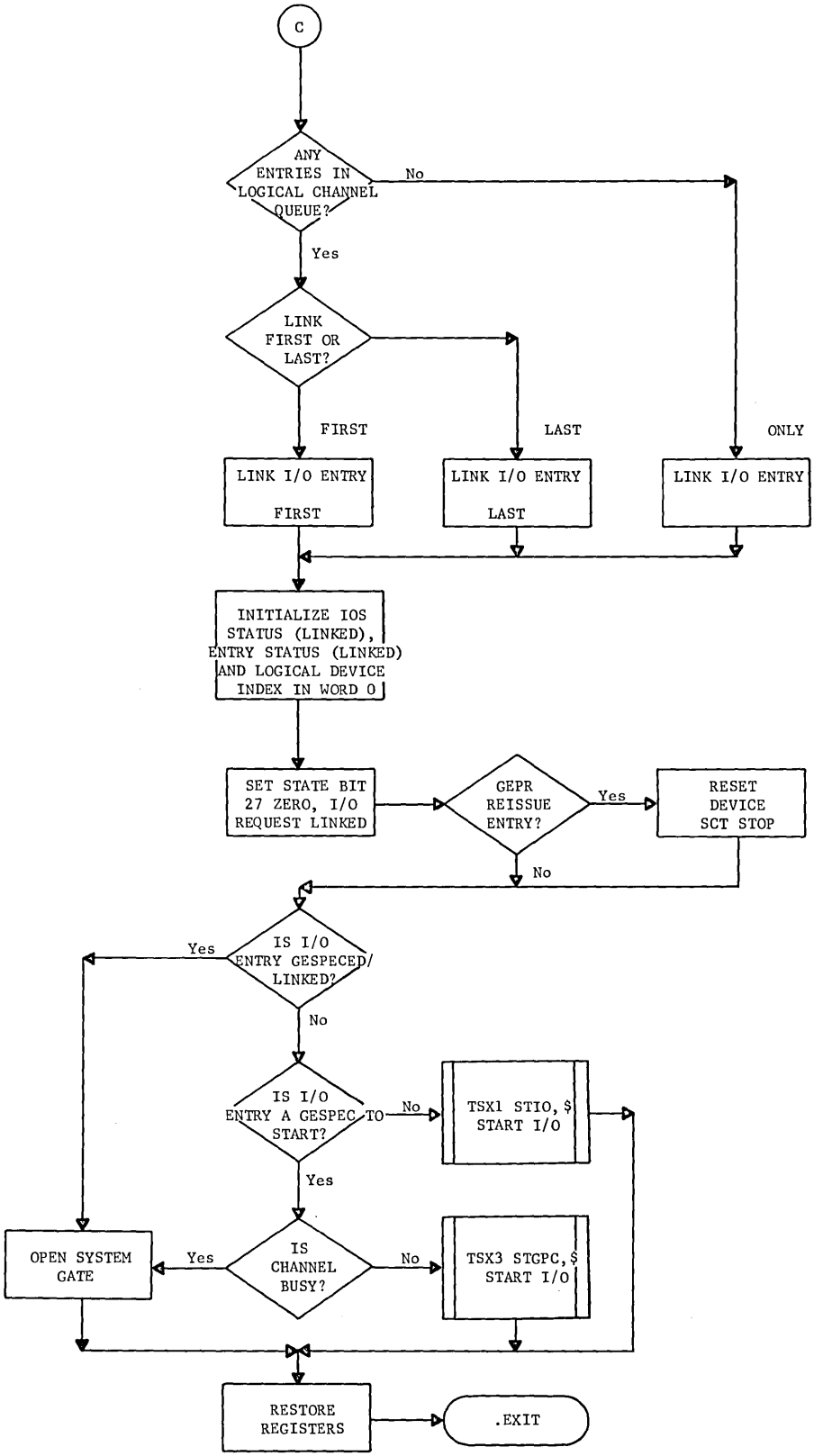
(Reference CPB-1494)

LINK I/O TO END OF QUEUE



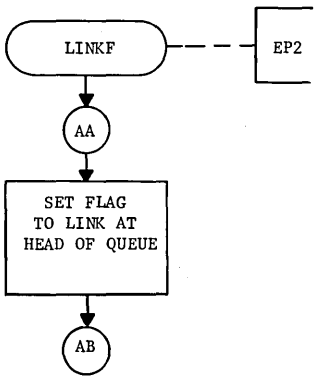
LINK (EP1)
.MIOS



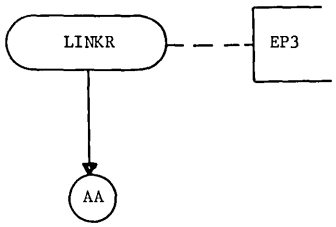

C
ANY ENTRIES IN LOGICAL CHANNEL QUEUE?
No
Yes
LINK FIRST OR LAST?
FIRST
LAST
ONLY
LINK I/O ENTRY FIRST
LINK I/O ENTRY LAST
LINK I/O ENTRY
INITIALIZE IOS STATUS (LINKED), ENTRY STATUS (LINKED) AND LOGICAL DEVICE INDEX IN WORD 0
SET STATE BIT 27 ZERO, I/O REQUEST LINKED
GEPR REISSUE ENTRY?
Yes
RESET DEVICE SCT STOP
No
IS I/O ENTRY GESPECED/ LINKED?
Yes
No
IS I/O ENTRY A GESPEC TO START?
No
TSX1 STIO, $ START I/O
Yes
OPEN SYSTEM GATE
Yes
IS CHANNEL BUSY?
No
TSX3 STGPC, $ START I/O
RESTORE REGISTERS
.EXIT

LINKF (EP2)
.MIOS

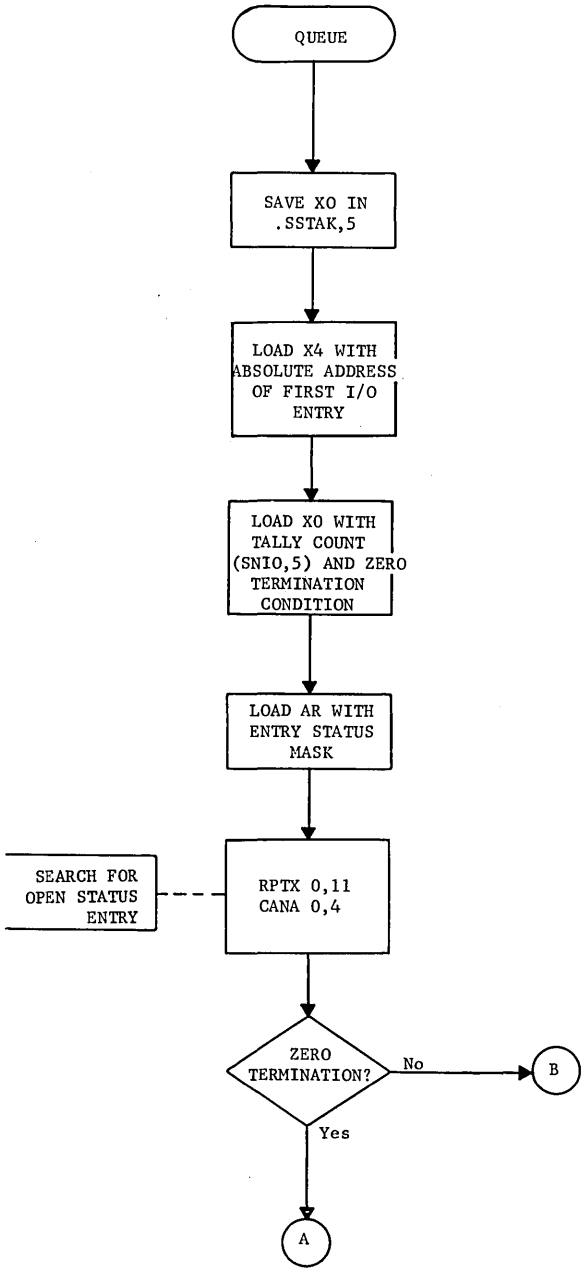
LINK I/O TO FRONT OF QUEUE

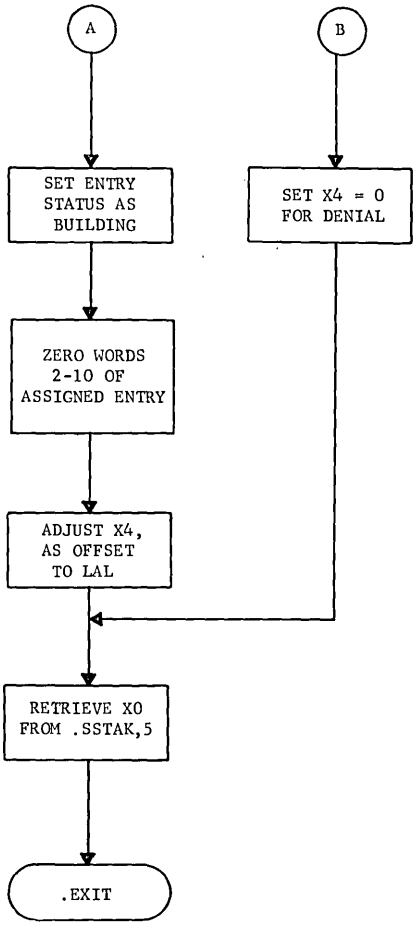


LINK REISSUED I/O TO FRONT OF QUEUE



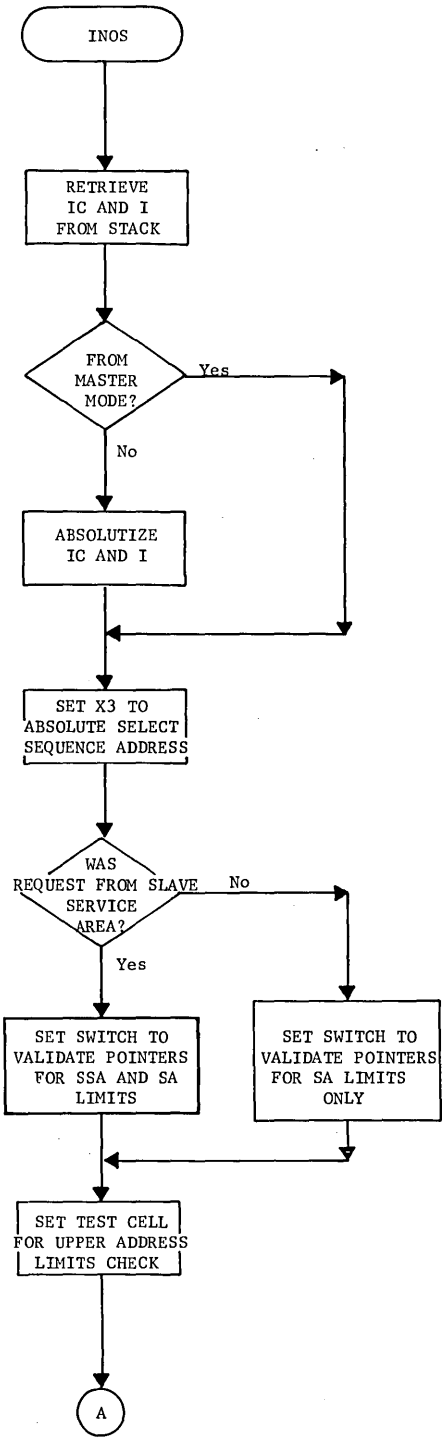
ASSIGN AN I/O ENTRY

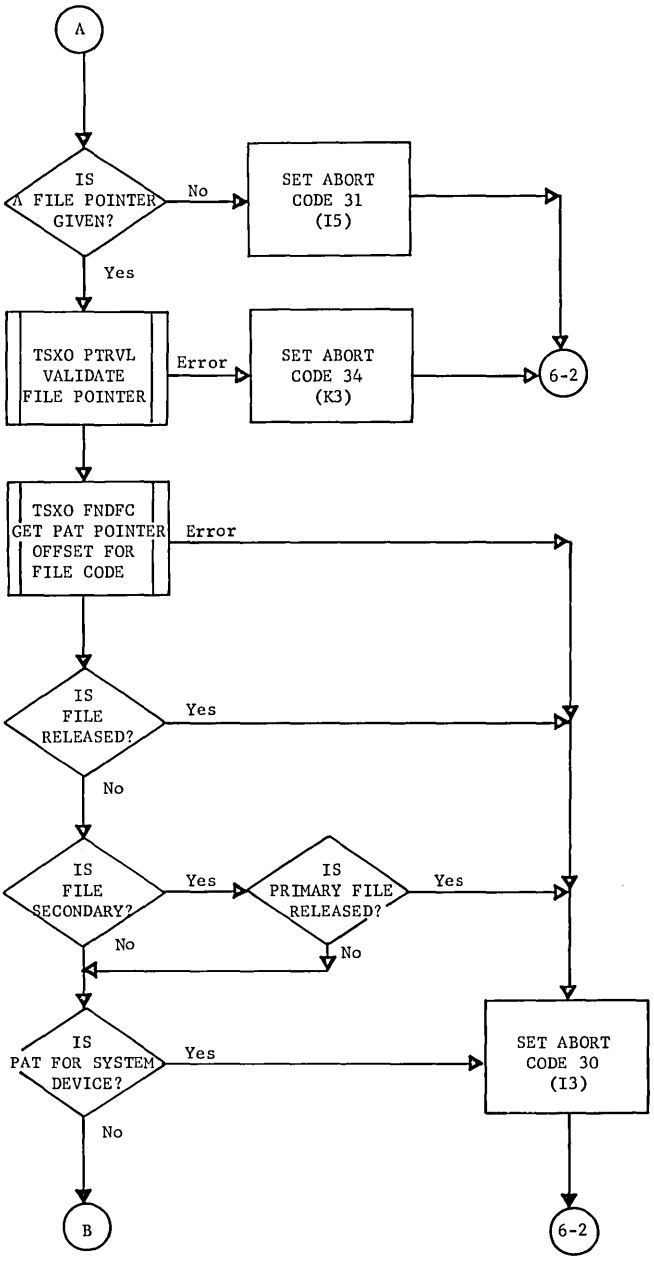



A
B
SET ENTRY STATUS AS BUILDING
SET X4 = 0 FOR DENIAL
ZERO WORDS 2-10 OF ASSIGNED ENTRY
ADJUST X4, AS OFFSET TO LAL
RETRIEVE X0 FROM .SSTAK,5
.EXIT

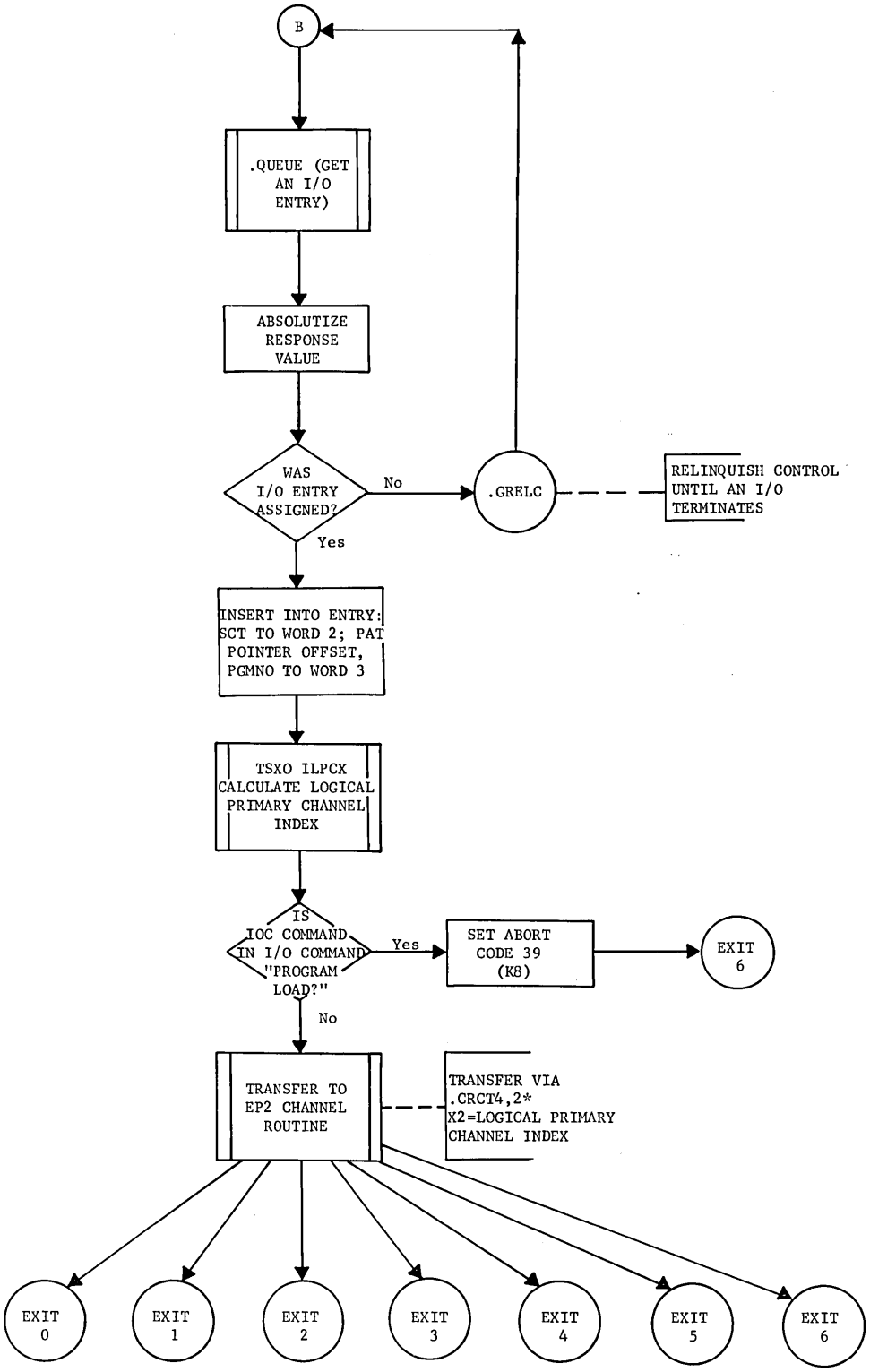
INOS (EP5)
.MIOS

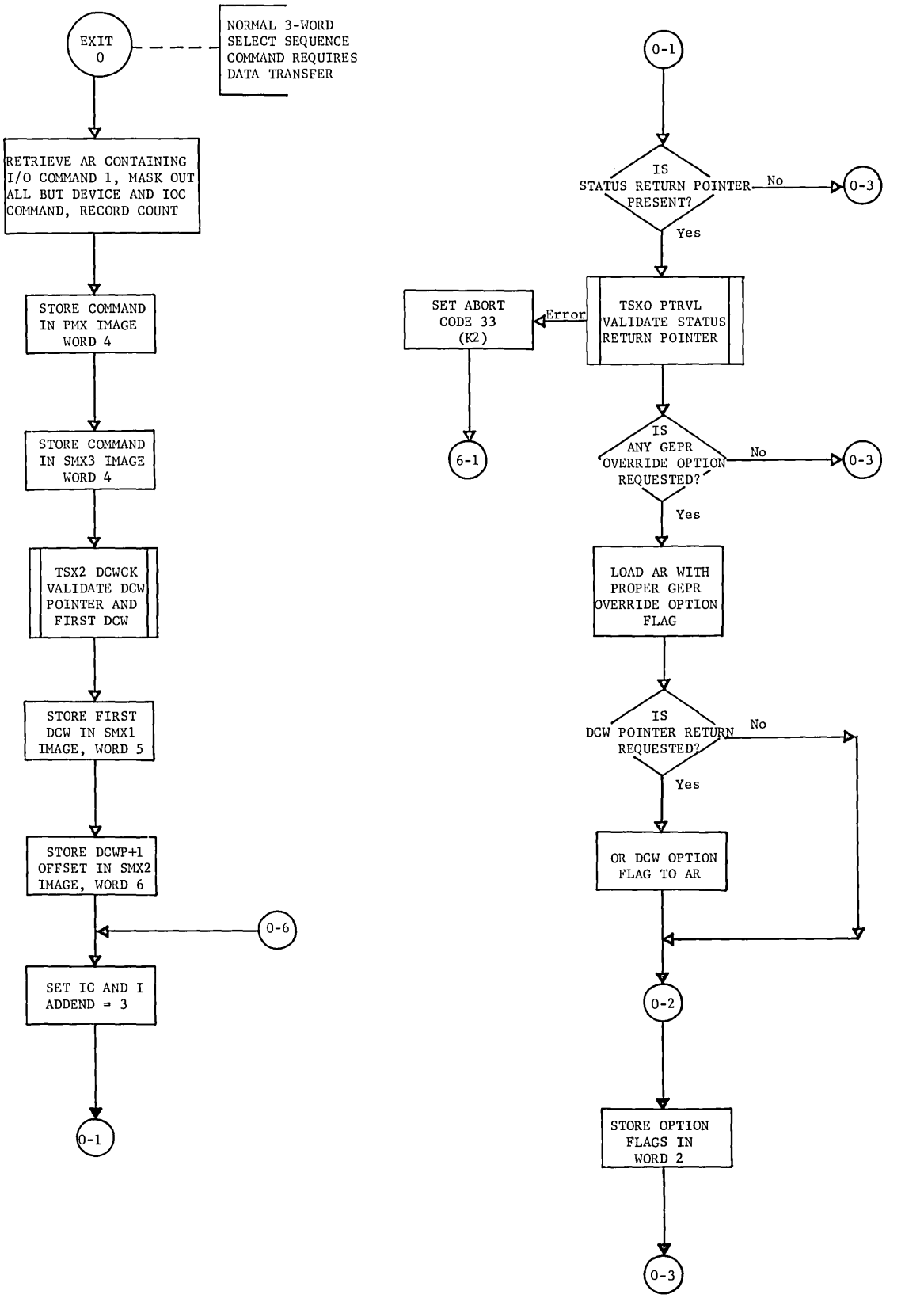
MME GEINOS PROCESSOR

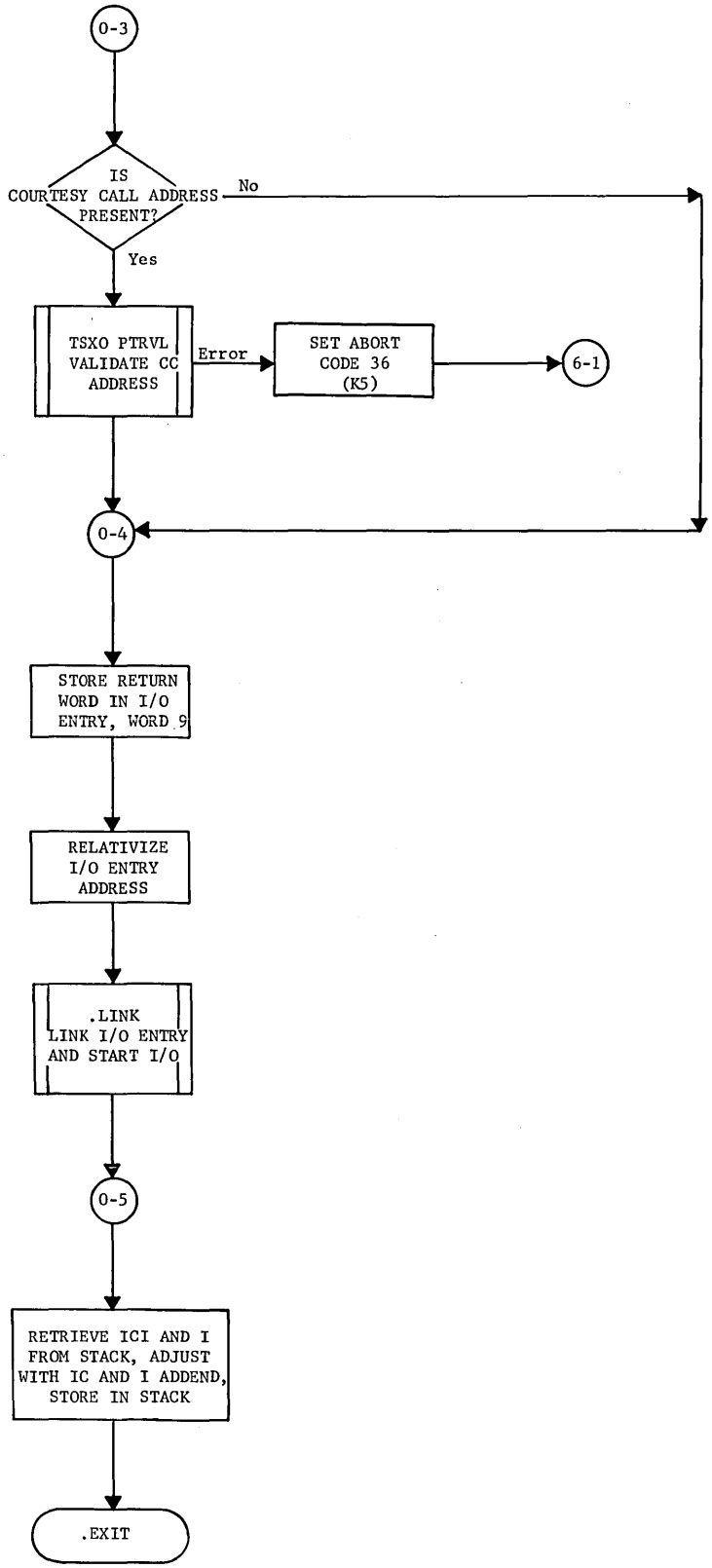


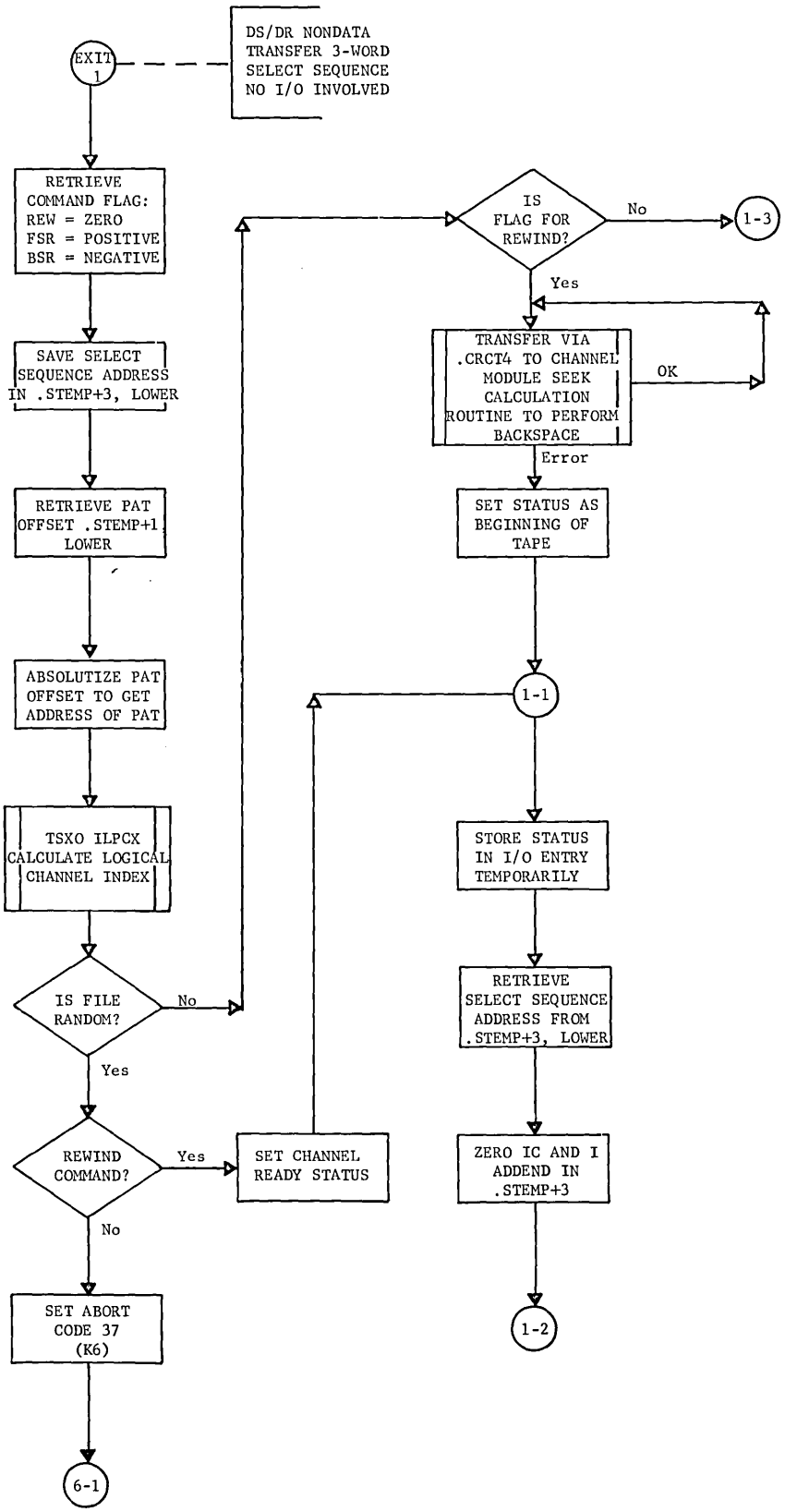

A
IS A FILE POINTER GIVEN?
No
SET ABORT CODE 31 (I5)
Yes
TSXO PTRVL VALIDATE FILE POINTER
Error
SET ABORT CODE 34 (K3)
6-2
TSXO FNDFC GET PAT POINTER OFFSET FOR FILE CODE
Error
IS FILE RELEASED?
Yes
No
IS FILE SECONDARY?
Yes
IS PRIMARY FILE RELEASED?
Yes
No
No
IS PAT FOR SYSTEM DEVICE?
Yes
SET ABORT CODE 30 (I3)
No
B
6-2

INOS (EP5)
.MIOS

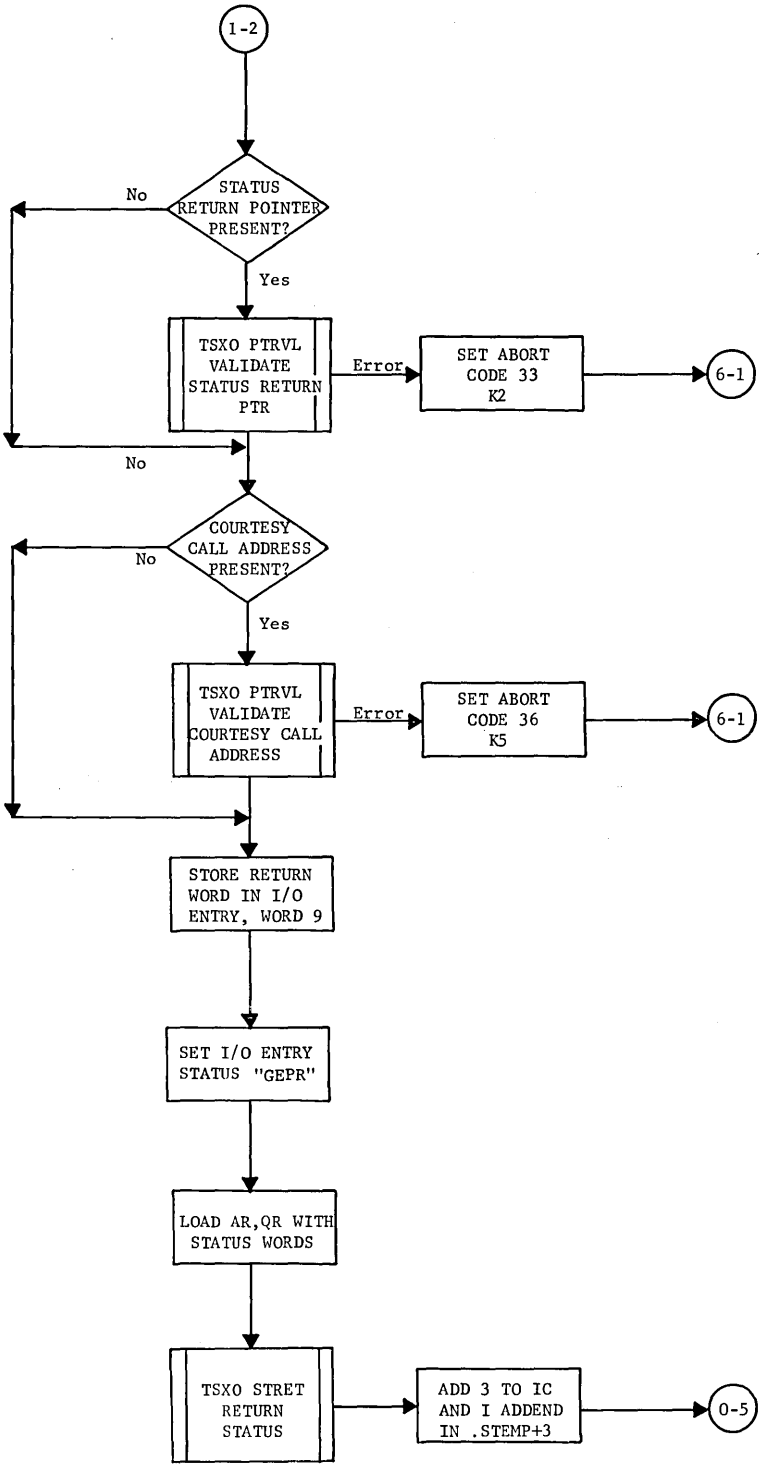


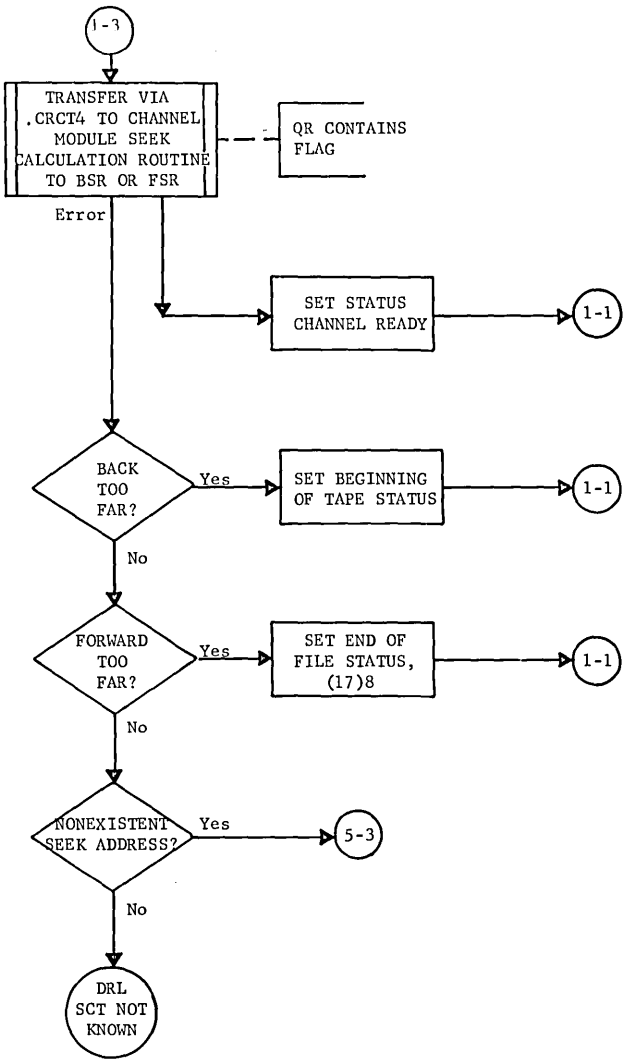

EXIT 0
NORMAL 3-WORD SELECT SEQUENCE COMMAND REQUIRES DATA TRANSFER
RETRIEVE AR CONTAINING I/O COMMAND 1, MASK OUT ALL BUT DEVICE AND IOC COMMAND, RECORD COUNT
STORE COMMAND IN PMX IMAGE WORD 4
STORE COMMAND IN SMX3 IMAGE WORD 4
TSX2 DCWCK VALIDATE DCW POINTER AND FIRST DCW
STORE FIRST DCW IN SMX1 IMAGE, WORD 5
STORE DCWP+1 OFFSET IN SMX2 IMAGE, WORD 6
0-6
SET IC AND I ADDEND = 3
0-1
0-1
IS STATUS RETURN POINTER PRESENT?
No
0-3
Yes
TSXO PTRVL VALIDATE STATUS RETURN POINTER
Error
SET ABORT CODE 33 (K2)
6-1
IS ANY GEPR OVERRIDE OPTION REQUESTED?
No
0-3
Yes
LOAD AR WITH PROPER GEPR OVERRIDE OPTION FLAG
IS DCW POINTER RETURN REQUESTED?
No
Yes
OR DCW OPTION FLAG TO AR
0-2
STORE OPTION FLAGS IN WORD 2
0-3

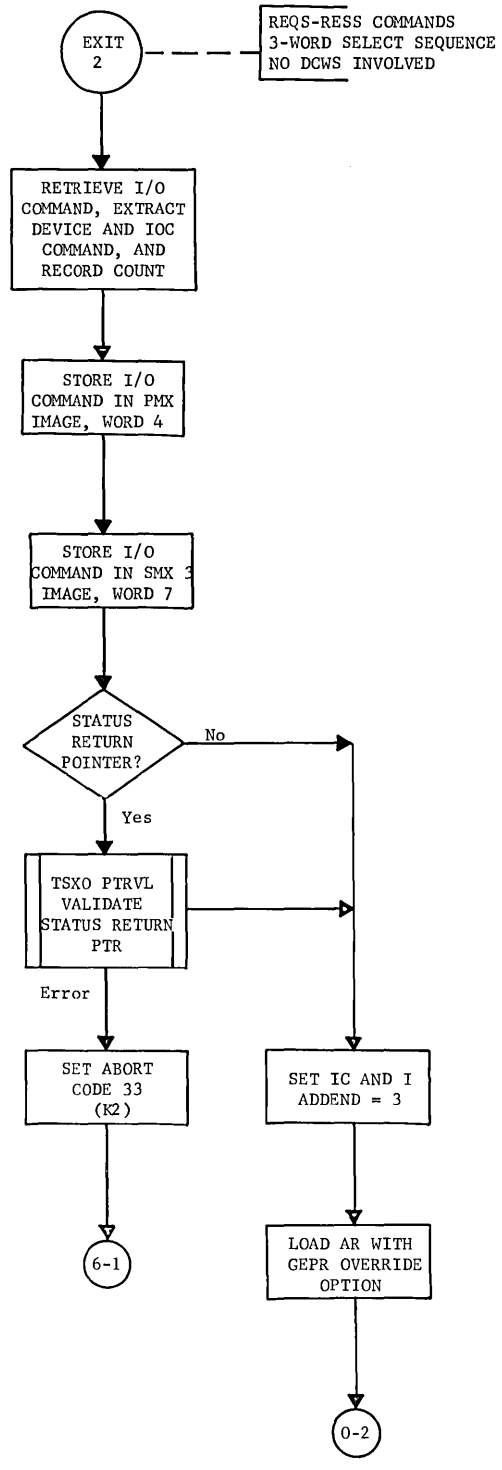

0-3
IS COURTESY CALL ADDRESS PRESENT?
No
Yes
TSXO PTRVL VALIDATE CC ADDRESS
Error
SET ABORT CODE 36 (K5)
6-1
0-4
STORE RETURN WORD IN I/O ENTRY, WORD 9
RELATIVIZE I/O ENTRY ADDRESS
.LINK LINK I/O ENTRY AND START I/O
0-5
RETRIEVE ICI AND I FROM STACK, ADJUST WITH IC AND I ADDEND, STORE IN STACK
.EXIT

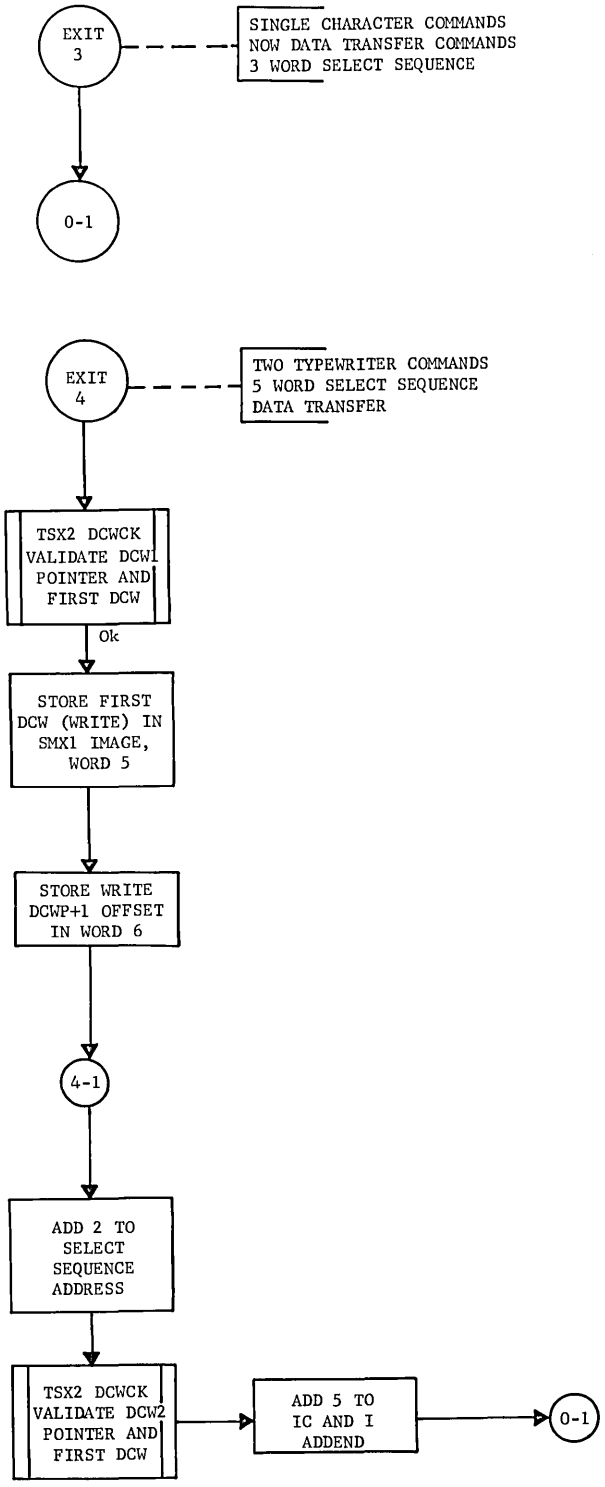

EXIT 1
DS/DR NONDATA TRANSFER 3-WORD SELECT SEQUENCE NO I/O INVOLVED
RETRIEVE COMMAND FLAG: REW = ZERO FSR = POSITIVE BSR = NEGATIVE
SAVE SELECT SEQUENCE ADDRESS IN .STEMP+3, LOWER
RETRIEVE PAT OFFSET .STEMP+1 LOWER
ABSOLUTIZE PAT OFFSET TO GET ADDRESS OF PAT
TSXO ILPCX CALCULATE LOGICAL CHANNEL INDEX
IS FILE RANDOM?
No
Yes
REWIND COMMAND?
Yes
SET CHANNEL READY STATUS
No
SET ABORT CODE 37 (K6)
6-1
IS FLAG FOR REWIND?
No
1-3
Yes
TRANSFER VIA .CRCT4 TO CHANNEL MODULE SEEK CALCULATION ROUTINE TO PERFORM BACKSPACE
OK
Error
SET STATUS AS BEGINNING OF TAPE
1-1
STORE STATUS IN I/O ENTRY TEMPORARILY
RETRIEVE SELECT SEQUENCE ADDRESS FROM .STEMP+3, LOWER
ZERO IC AND I ADDEND IN .STEMP+3
1-2

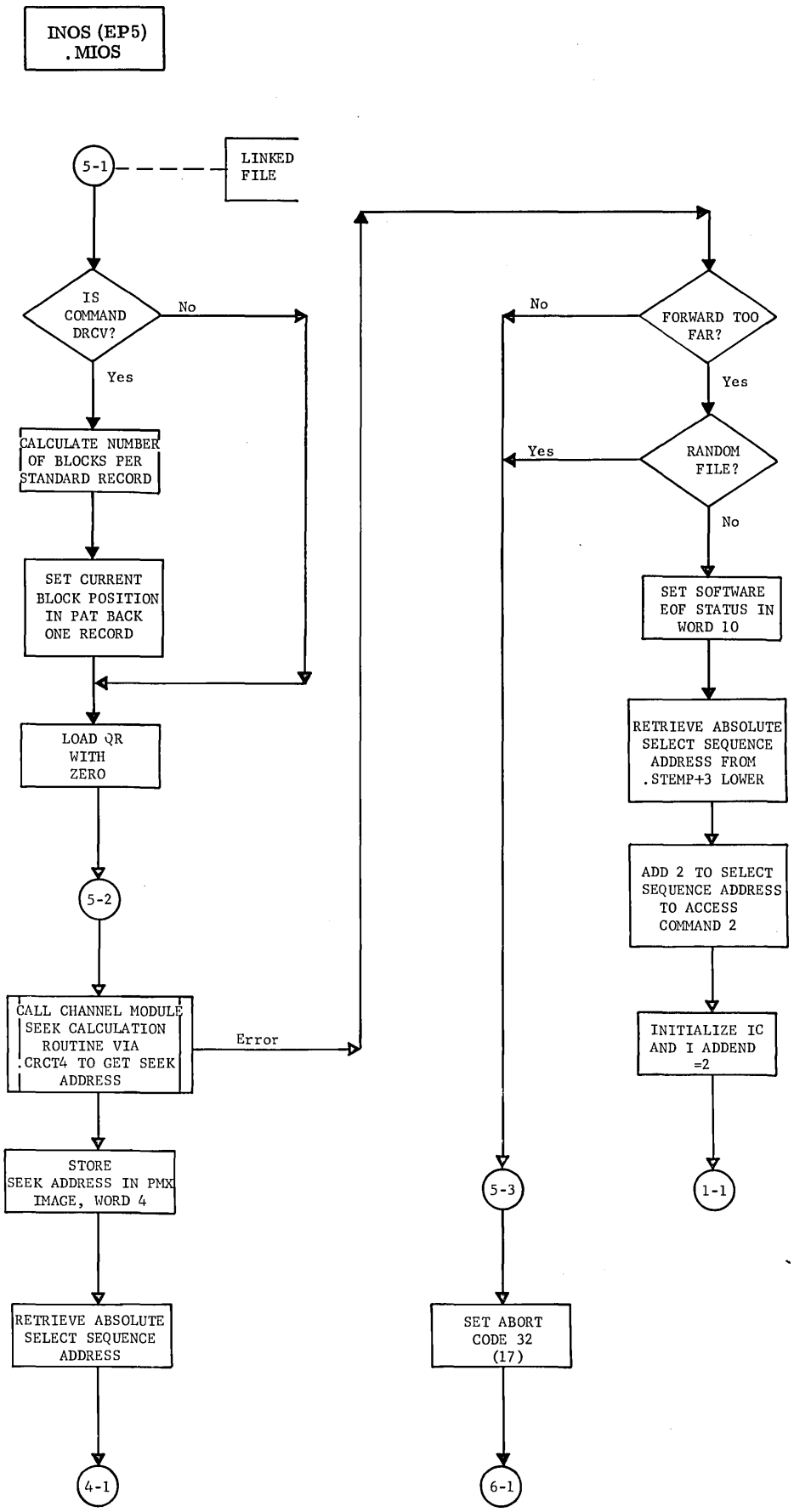
INOS (EP5)
.MIOS

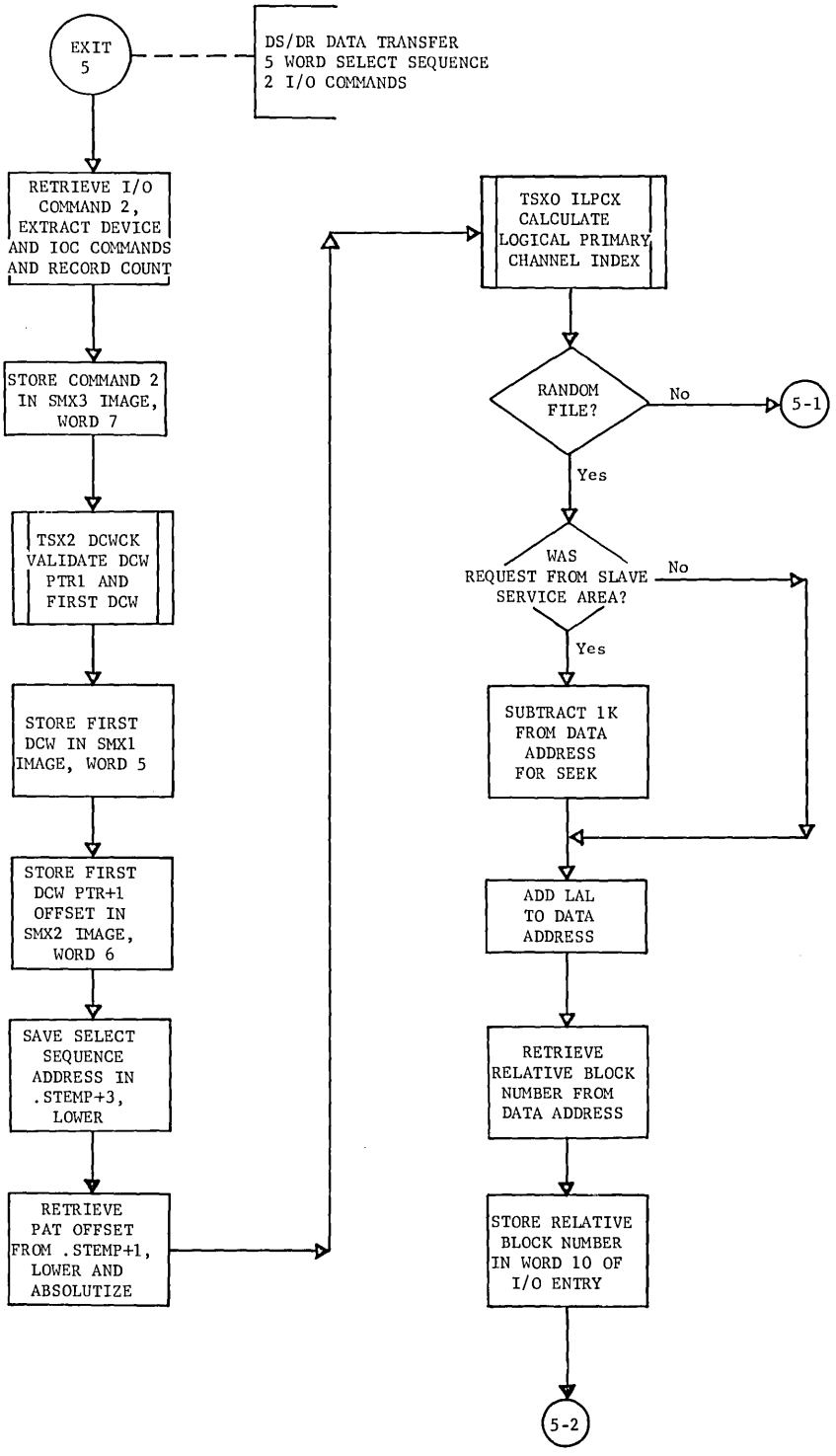



1-3
TRANSFER VIA .CRCT4 TO CHANNEL MODULE SEEK CALCULATION ROUTINE TO BSR OR FSR
QR CONTAINS FLAG
Error
SET STATUS CHANNEL READY
1-1
BACK TOO FAR?
Yes
SET BEGINNING OF TAPE STATUS
1-1
No
FORWARD TOO FAR?
Yes
SET END OF FILE STATUS, (17)8
1-1
No
NONEXISTENT SEEK ADDRESS?
Yes
5-3
No
DRL SCT NOT KNOWN

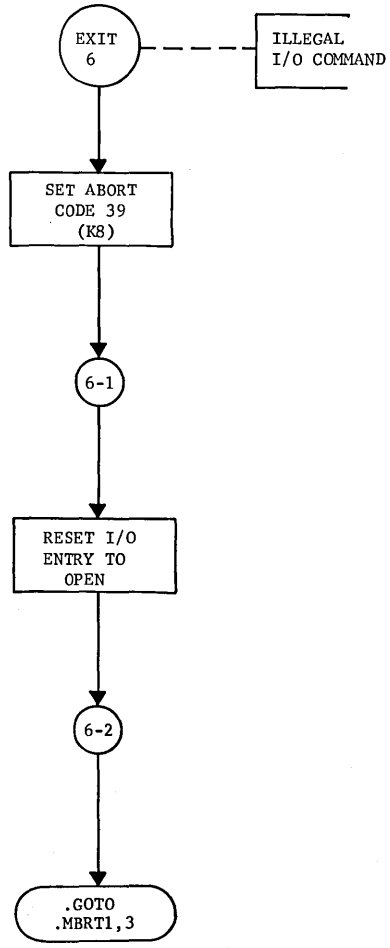

EXIT
2
REQS-RESS COMMANDS
3-WORD SELECT SEQUENCE
NO DCWS INVOLVED
RETRIEVE I/O COMMAND, EXTRACT DEVICE AND IOC COMMAND, AND RECORD COUNT
STORE I/O COMMAND IN PMX IMAGE, WORD 4
STORE I/O COMMAND IN SMX 3 IMAGE, WORD 7
STATUS RETURN POINTER?
No
Yes
TSXO PTRVL VALIDATE STATUS RETURN PTR
Error
SET ABORT CODE 33 (K2)
SET IC AND I ADDEND = 3
6-1
LOAD AR WITH GEPR OVERRIDE OPTION
0-2


EXIT 3
SINGLE CHARACTER COMMANDS
NOW DATA TRANSFER COMMANDS
3 WORD SELECT SEQUENCE
0-1
EXIT 4
TWO TYPEWRITER COMMANDS
5 WORD SELECT SEQUENCE
DATA TRANSFER
TSX2 DCWCK
VALIDATE DCW1
POINTER AND
FIRST DCW
Ok
STORE FIRST
DCW (WRITE) IN
SMX1 IMAGE,
WORD 5
STORE WRITE
DCWP+1 OFFSET
IN WORD 6
4-1
ADD 2 TO
SELECT
SEQUENCE
ADDRESS
TSX2 DCWCK
VALIDATE DCW2
POINTER AND
FIRST DCW
ADD 5 TO
IC AND I
ADDEND
0-1

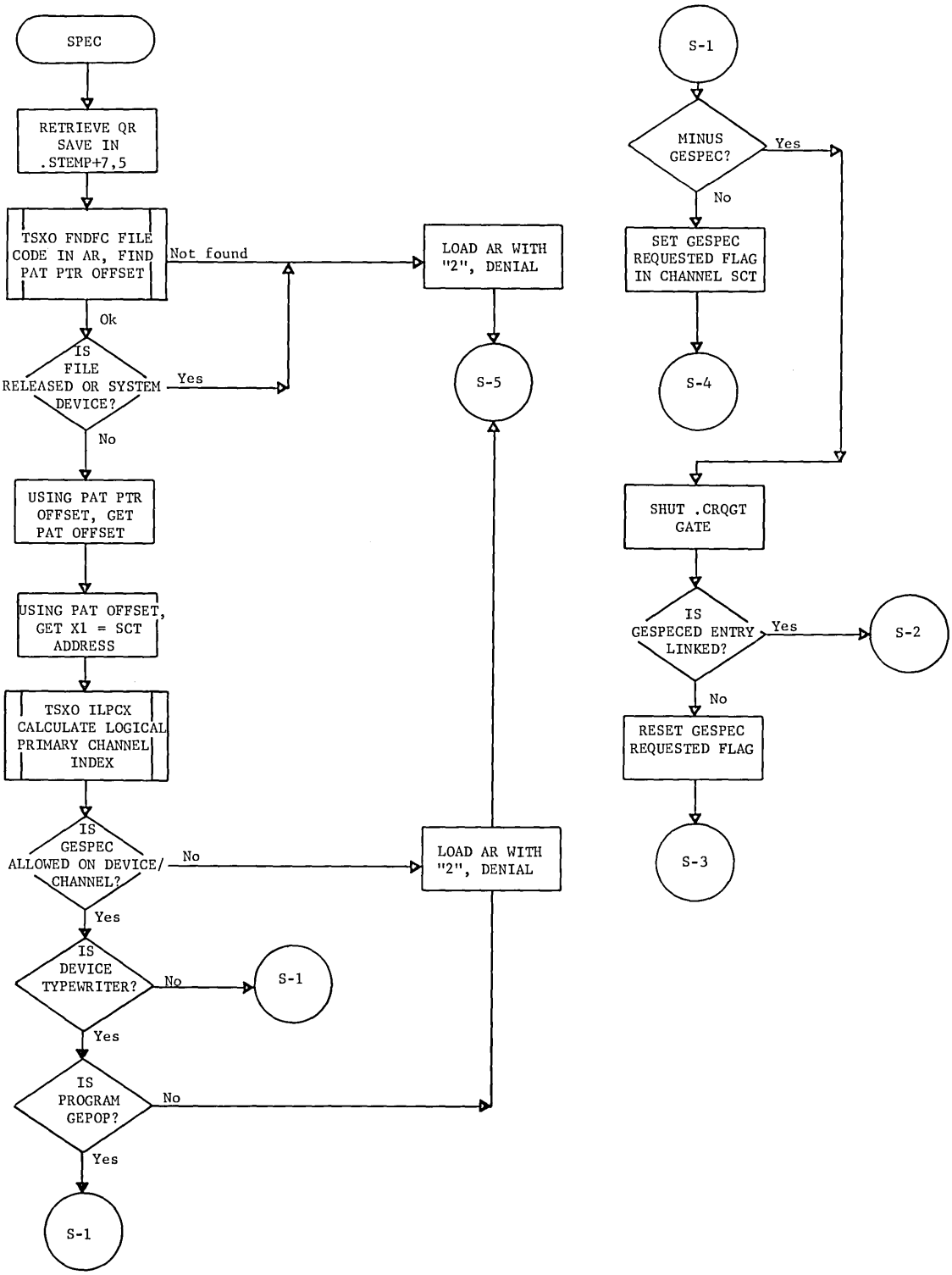

INOS (EP5)
.MIOS
5-1
LINKED FILE
IS COMMAND DRCV?
No
Yes
CALCULATE NUMBER OF BLOCKS PER STANDARD RECORD
SET CURRENT BLOCK POSITION IN PAT BACK ONE RECORD
LOAD QR WITH ZERO
5-2
CALL CHANNEL MODULE SEEK CALCULATION ROUTINE VIA .CRCT4 TO GET SEEK ADDRESS
Error
STORE SEEK ADDRESS IN PMX IMAGE, WORD 4
RETRIEVE ABSOLUTE SELECT SEQUENCE ADDRESS
4-1
FORWARD TOO FAR?
No
Yes
RANDOM FILE?
Yes
No
SET SOFTWARE EOF STATUS IN WORD 10
RETRIEVE ABSOLUTE SELECT SEQUENCE ADDRESS FROM .STEMP+3 LOWER
ADD 2 TO SELECT SEQUENCE ADDRESS TO ACCESS COMMAND 2
INITIALIZE IC AND I ADDEND =2
1-1
5-3
SET ABORT CODE 32 (17)
6-1


EXIT 5
DS/DR DATA TRANSFER
5 WORD SELECT SEQUENCE
2 I/O COMMANDS
RETRIEVE I/O COMMAND 2, EXTRACT DEVICE AND IOC COMMANDS AND RECORD COUNT
STORE COMMAND 2 IN SMX3 IMAGE, WORD 7
TSX2 DCWCK VALIDATE DCW PTR1 AND FIRST DCW
STORE FIRST DCW IN SMX1 IMAGE, WORD 5
STORE FIRST DCW PTR+1 OFFSET IN SMX2 IMAGE, WORD 6
SAVE SELECT SEQUENCE ADDRESS IN .STEMP+3, LOWER
RETRIEVE PAT OFFSET FROM .STEMP+1, LOWER AND ABSOLUTIZE
TSX0 ILPCX CALCULATE LOGICAL PRIMARY CHANNEL INDEX
RANDOM FILE?
No
5-1
Yes
WAS REQUEST FROM SLAVE SERVICE AREA?
No
Yes
SUBTRACT 1K FROM DATA ADDRESS FOR SEEK
ADD LAL TO DATA ADDRESS
RETRIEVE RELATIVE BLOCK NUMBER FROM DATA ADDRESS
STORE RELATIVE BLOCK NUMBER IN WORD 10 OF I/O ENTRY
5-2

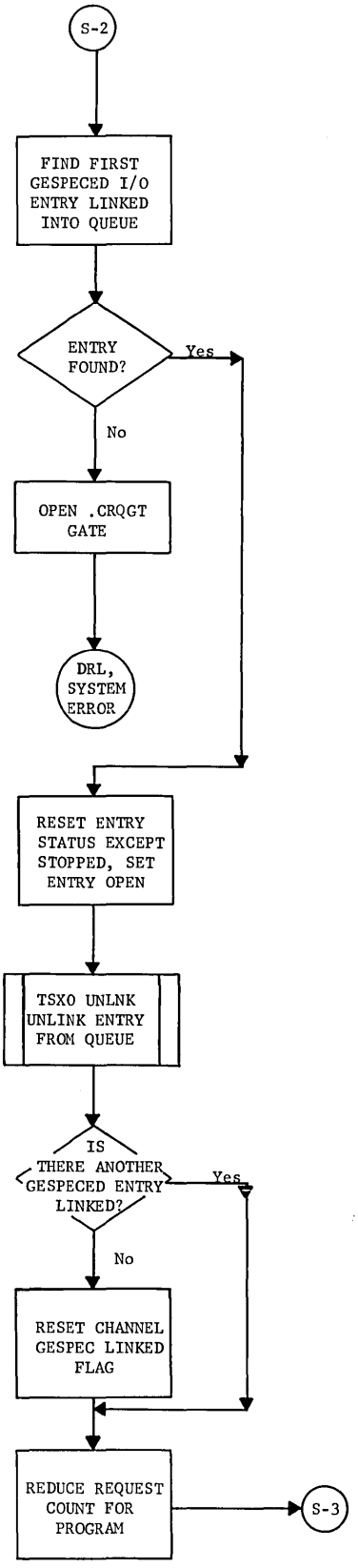
INOS (EP5)
.MIOS

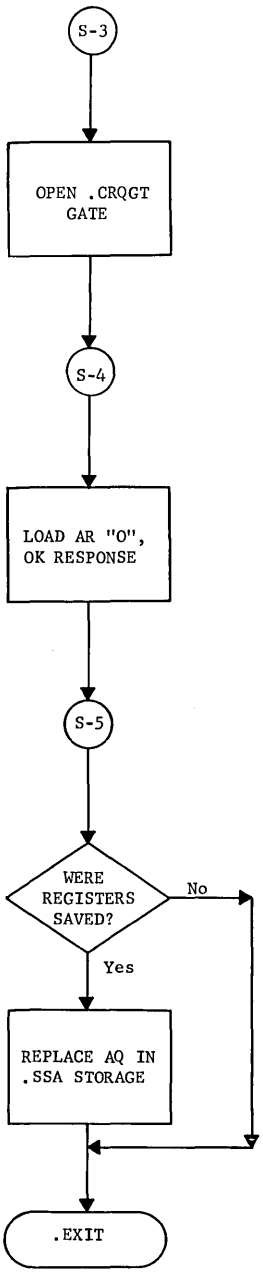


MME GESPEC PROCESSOR

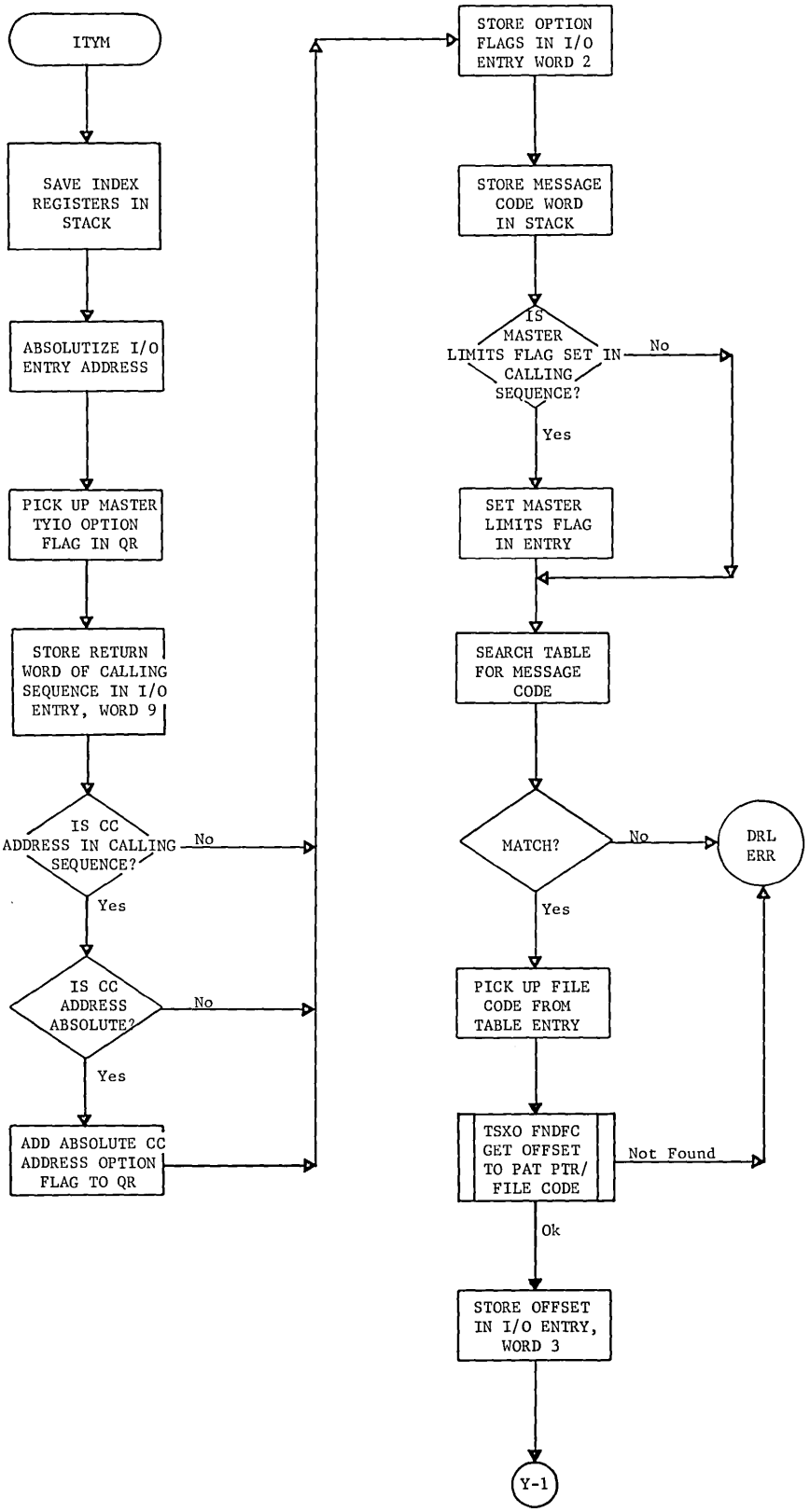


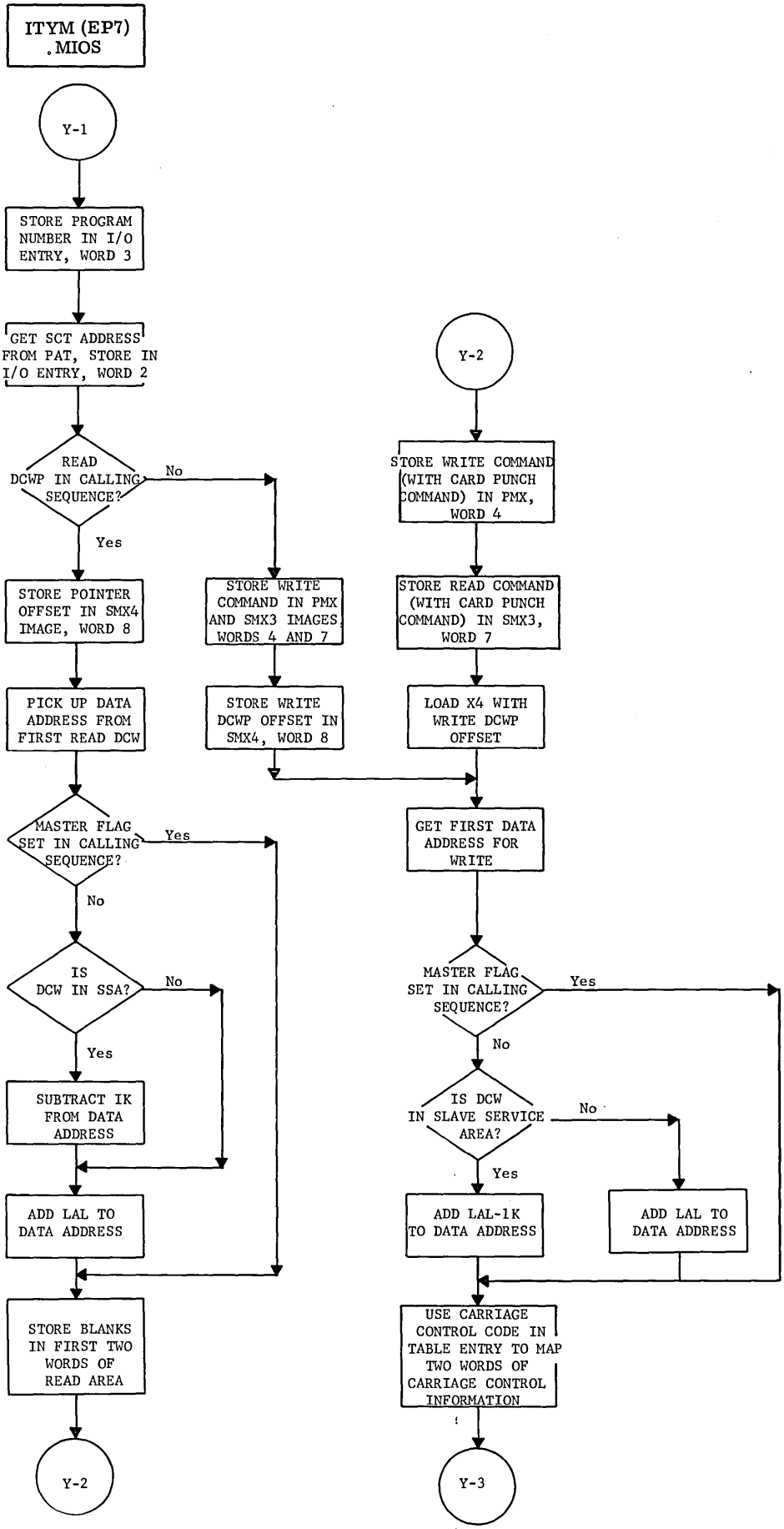
SPEC (EP6)
.MIOS

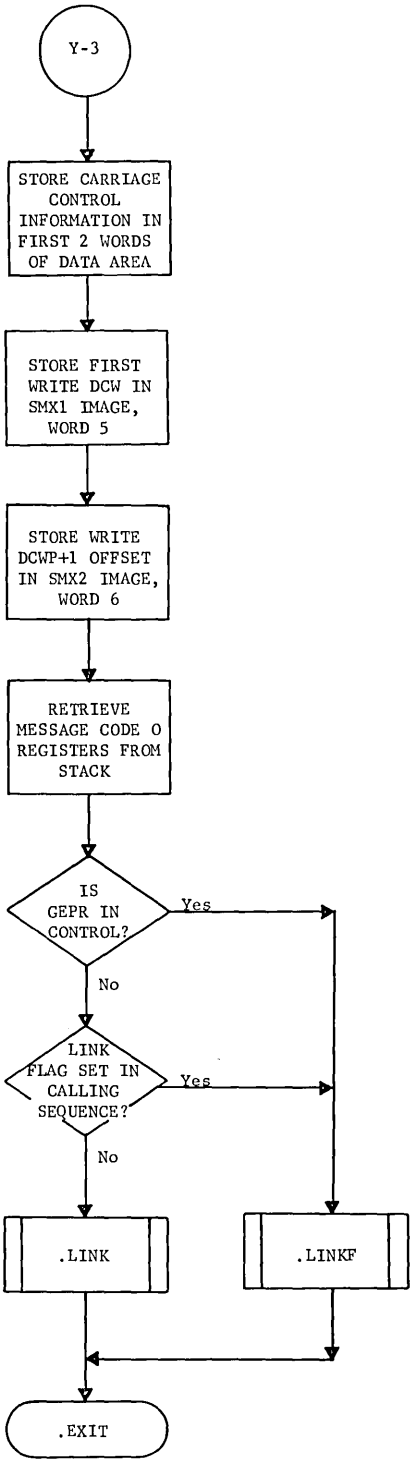




MASTER MESSAGE PROCESSOR

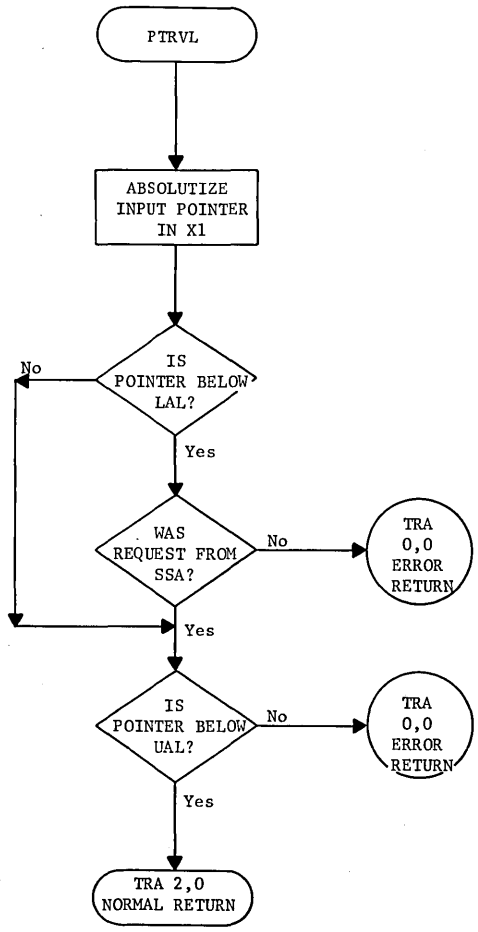



ITYM (EP7)
.MIOS
Y-1
STORE PROGRAM NUMBER IN I/O ENTRY, WORD 3
GET SCT ADDRESS FROM PAT, STORE IN I/O ENTRY, WORD 2
READ DCWP IN CALLING SEQUENCE?
No
Yes
STORE POINTER OFFSET IN SMX4 IMAGE, WORD 8
PICK UP DATA ADDRESS FROM FIRST READ DCW
MASTER FLAG SET IN CALLING SEQUENCE?
Yes
No
IS DCW IN SSA?
No
Yes
SUBTRACT IK FROM DATA ADDRESS
ADD LAL TO DATA ADDRESS
STORE BLANKS IN FIRST TWO WORDS OF READ AREA
Y-2
STORE WRITE COMMAND IN PMX AND SMX3 IMAGES WORDS 4 AND 7
STORE WRITE DCWP OFFSET IN SMX4, WORD 8
Y-2
STORE WRITE COMMAND (WITH CARD PUNCH COMMAND) IN PMX, WORD 4
STORE READ COMMAND (WITH CARD PUNCH COMMAND) IN SMX3, WORD 7
LOAD X4 WITH WRITE DCWP OFFSET
GET FIRST DATA ADDRESS FOR WRITE
MASTER FLAG SET IN CALLING SEQUENCE?
Yes
No
IS DCW IN SLAVE SERVICE AREA?
No
Yes
ADD LAL-1K TO DATA ADDRESS
ADD LAL TO DATA ADDRESS
USE CARRIAGE CONTROL CODE IN TABLE ENTRY TO MAP TWO WORDS OF CARRIAGE CONTROL INFORMATION
Y-3

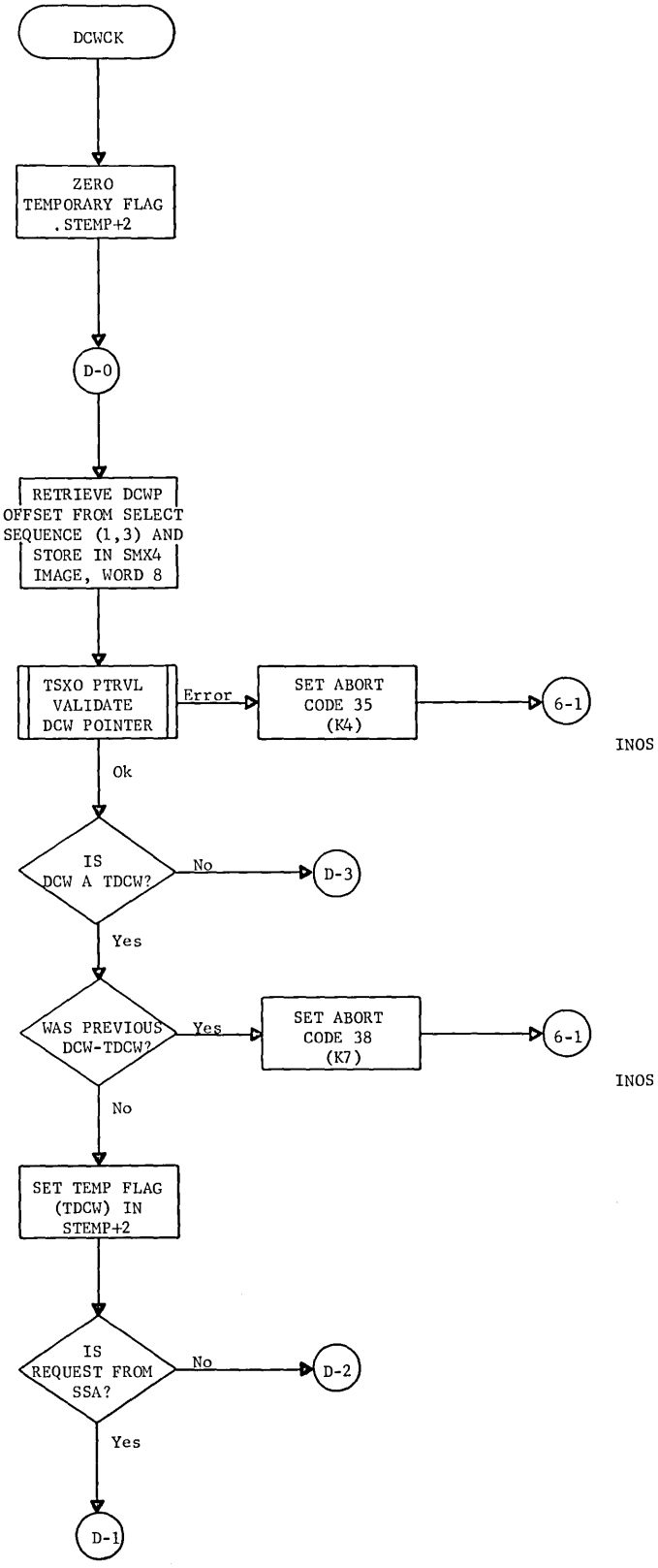

Y-3
STORE CARRIAGE CONTROL INFORMATION IN FIRST 2 WORDS OF DATA AREA
STORE FIRST WRITE DCW IN SMX1 IMAGE, WORD 5
STORE WRITE DCWP+1 OFFSET IN SMX2 IMAGE, WORD 6
RETRIEVE MESSAGE CODE 0 REGISTERS FROM STACK
IS GEPR IN CONTROL?
Yes
No
LINK FLAG SET IN CALLING SEQUENCE?
Yes
No
.LINK
.LINKF
.EXIT

PTRVL
.MIOS

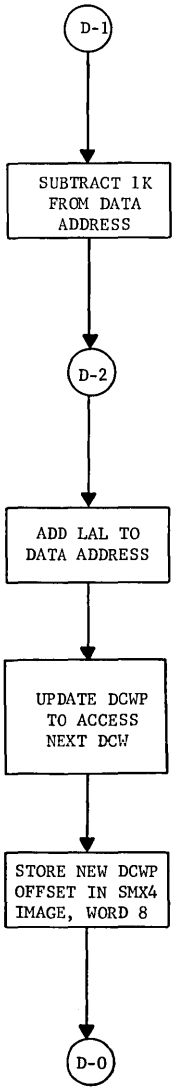
POINTER VALIDATION

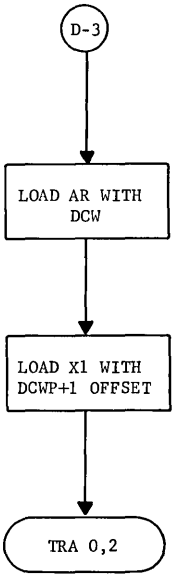


DCW POINTER VALIDATION

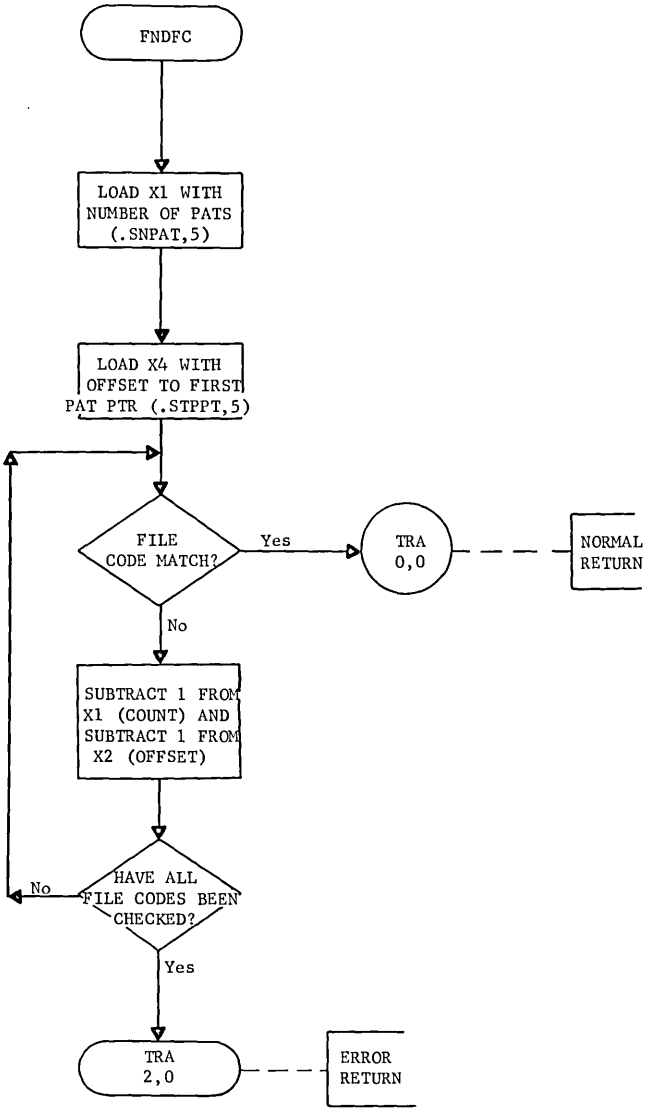


DCWCK
.MIOS


D-1
SUBTRACT 1K FROM DATA ADDRESS
D-2
ADD LAL TO DATA ADDRESS
UPDATE DCWP TO ACCESS NEXT DCW
STORE NEW DCWP OFFSET IN SMX4 IMAGE, WORD 8
D-0

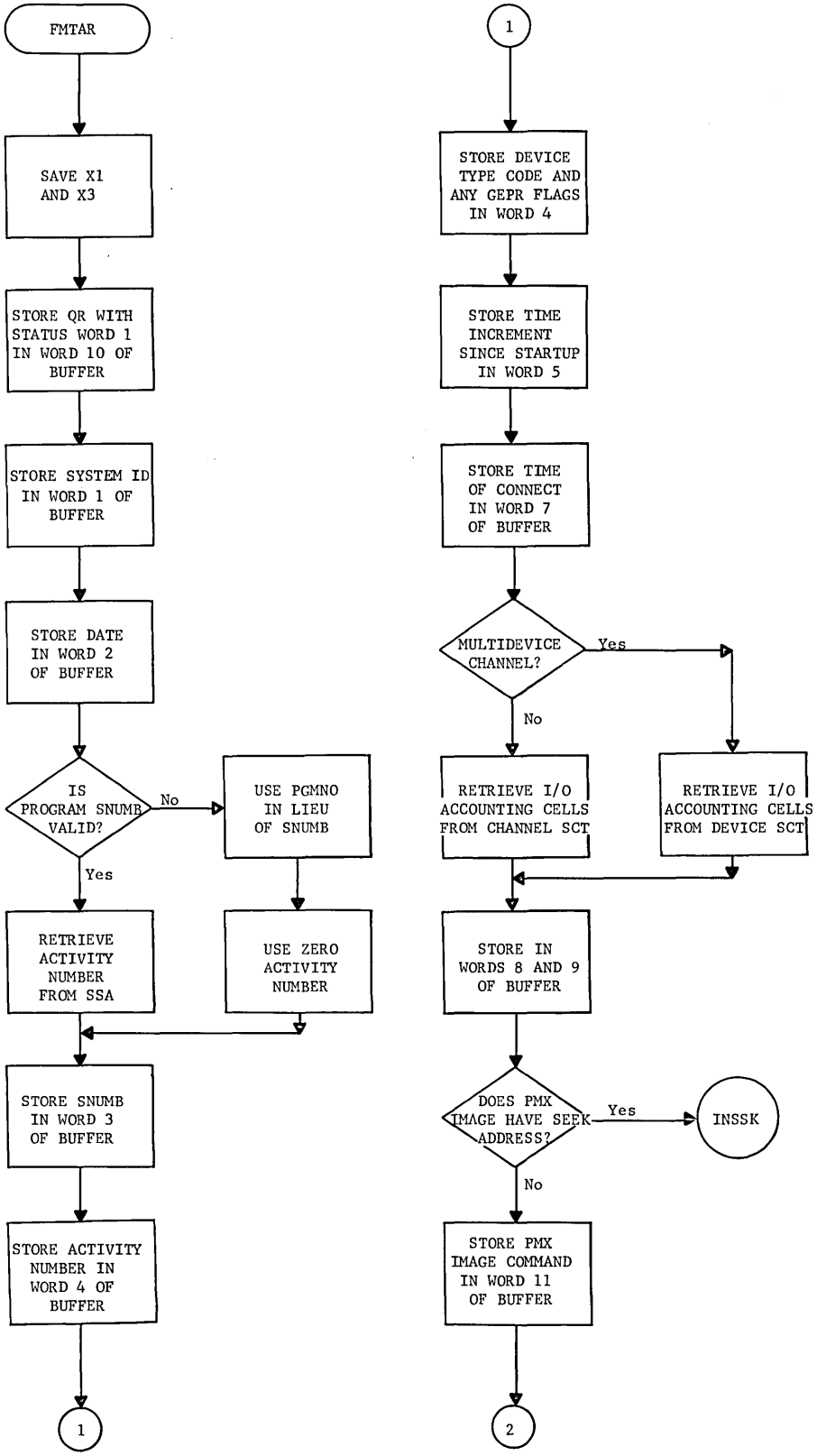

D-3
LOAD AR WITH DCW
LOAD X1 WITH DCWP+1 OFFSET
TRA 0,2

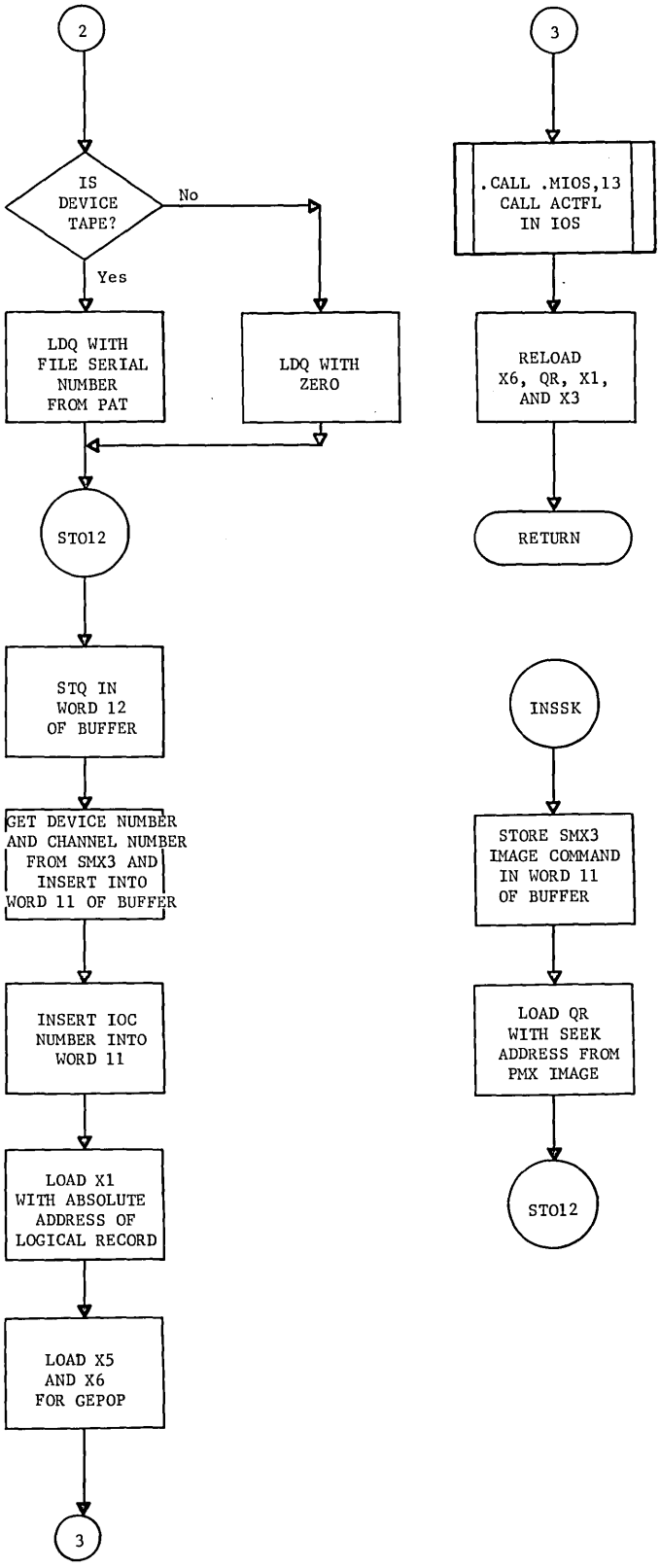
FIND FILE CODE



FMTAR
.MIOS

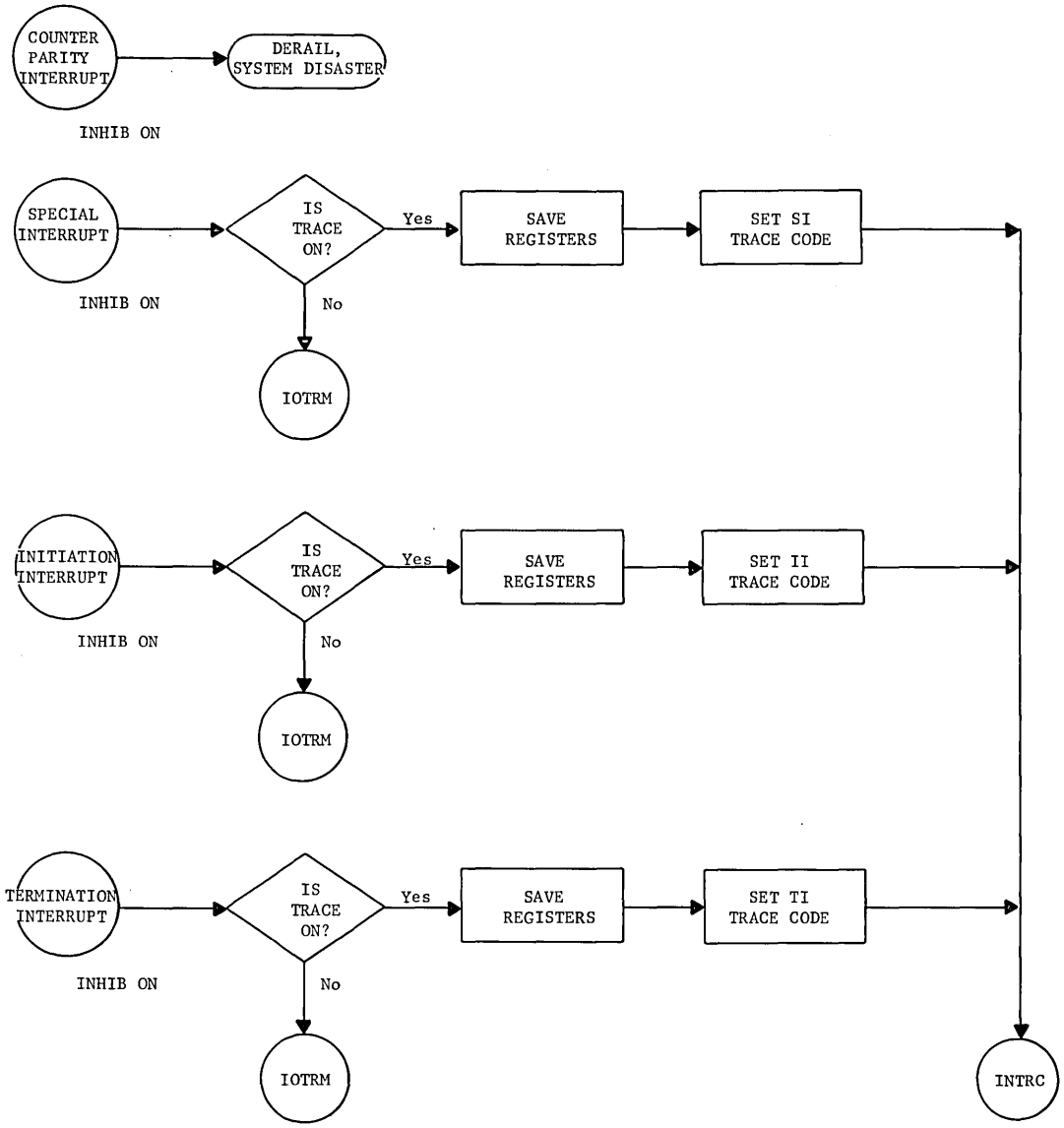
FORMAT ERROR ACCOUNTING RECORD

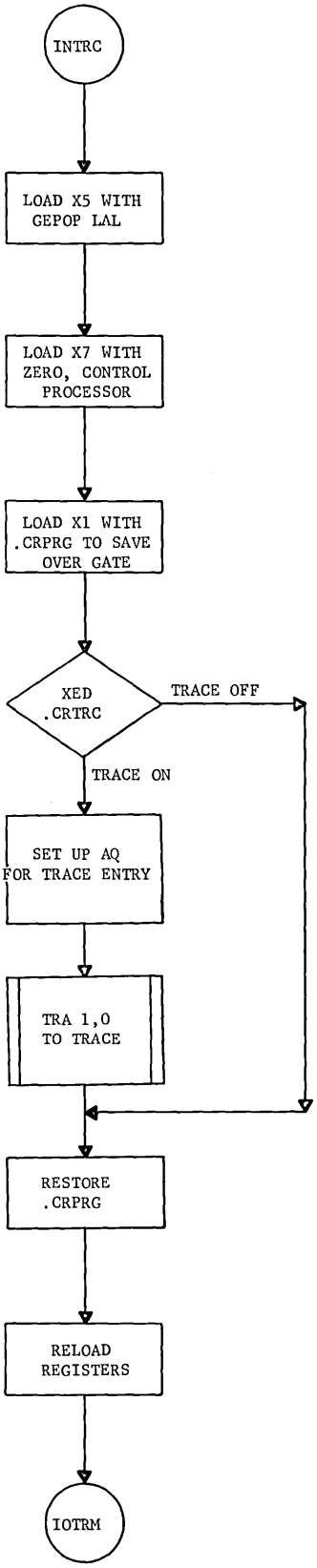


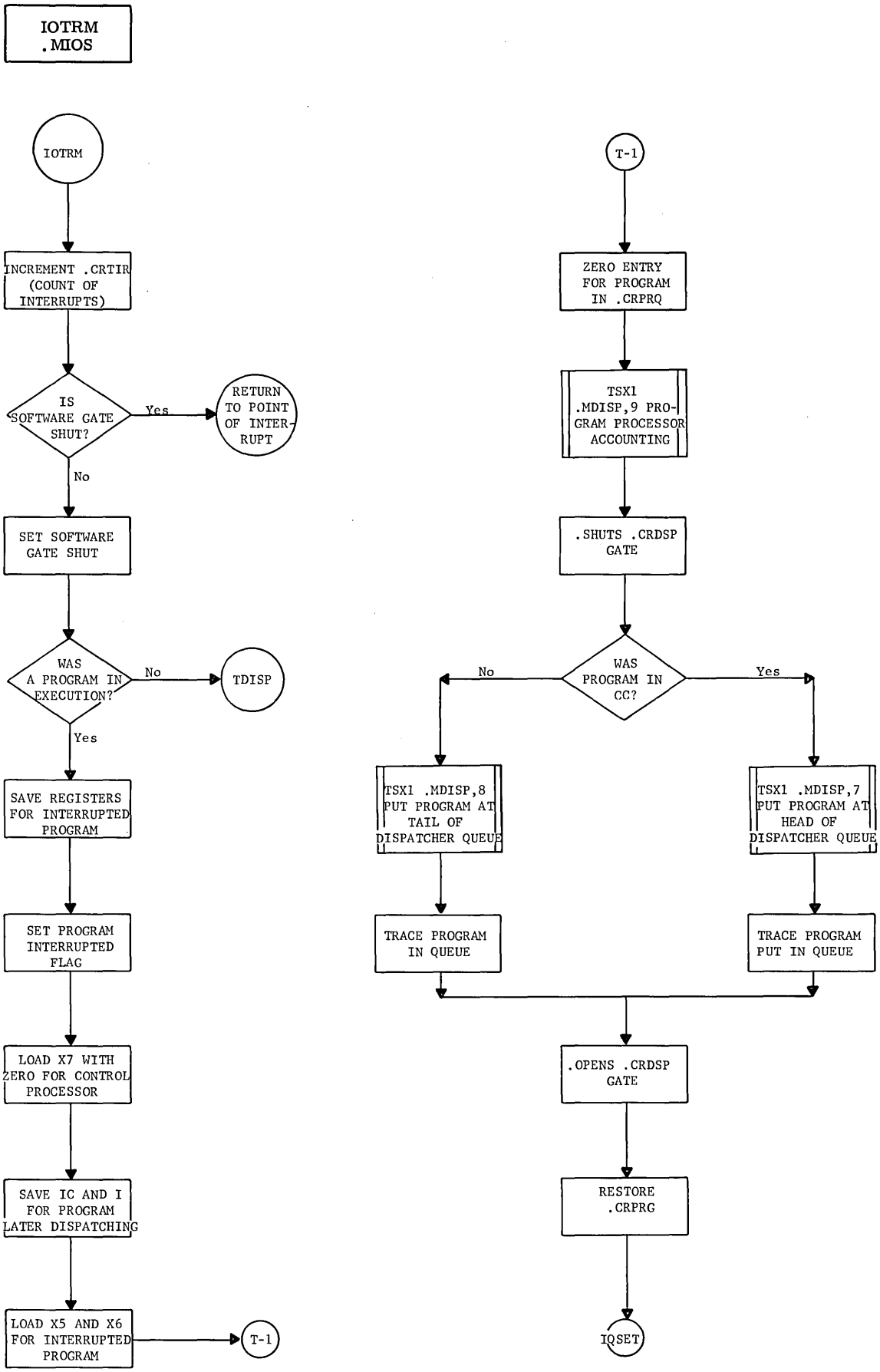

2
IS DEVICE TAPE?
No
Yes
LDQ WITH FILE SERIAL NUMBER FROM PAT
LDQ WITH ZERO
STO12
STQ IN WORD 12 OF BUFFER
GET DEVICE NUMBER AND CHANNEL NUMBER FROM SMX3 AND INSERT INTO WORD 11 OF BUFFER
INSERT IOC NUMBER INTO WORD 11
LOAD X1 WITH ABSOLUTE ADDRESS OF LOGICAL RECORD
LOAD X5 AND X6 FOR GEPOP
3
3
.CALL .MIOS,13
CALL ACTFL
IN IOS
RELOAD X6, QR, X1, AND X3
RETURN
INSSK
STORE SMX3 IMAGE COMMAND IN WORD 11 OF BUFFER
LOAD QR WITH SEEK ADDRESS FROM PMX IMAGE
STO12

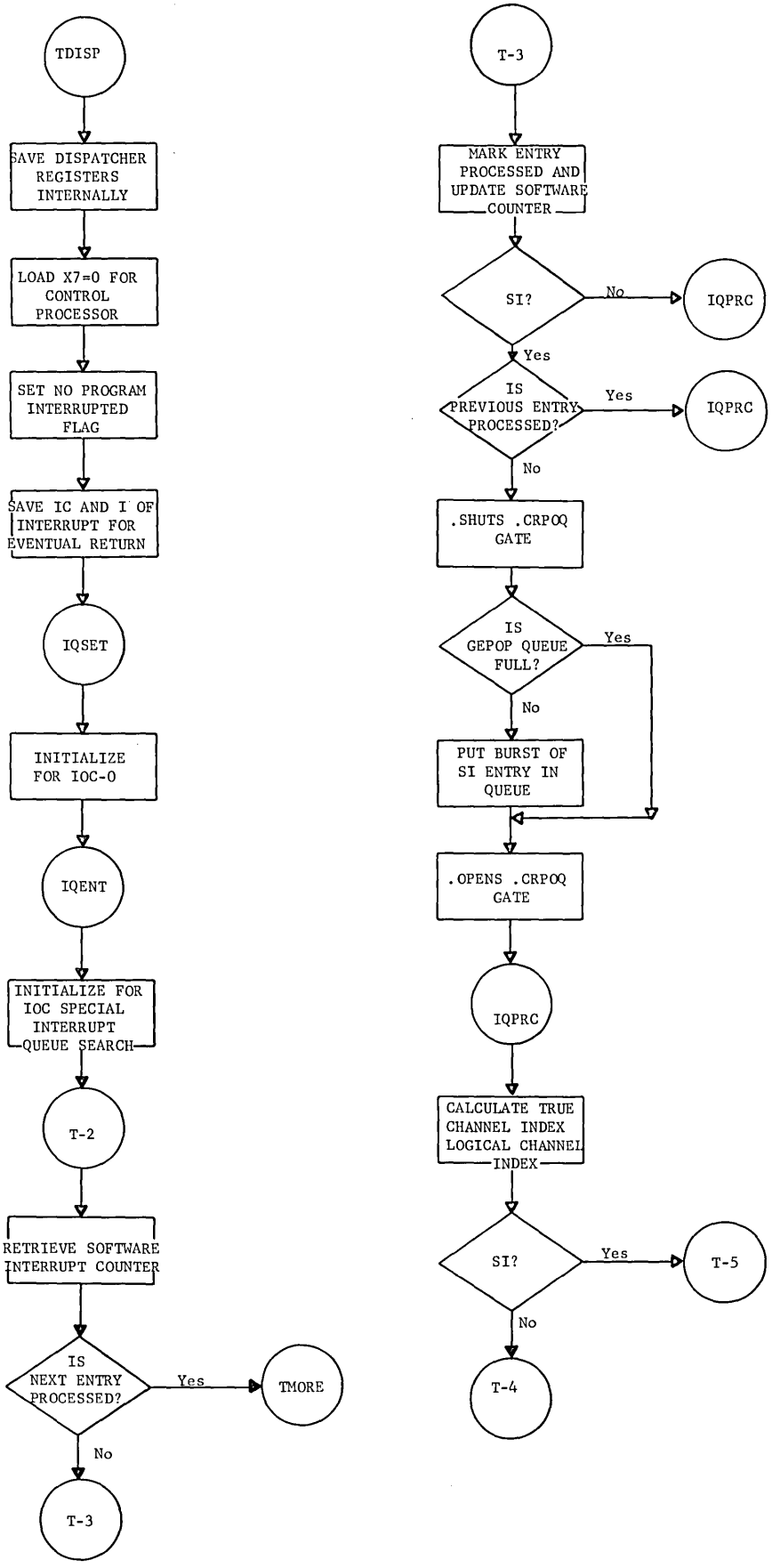
IOTRM
.MIOS

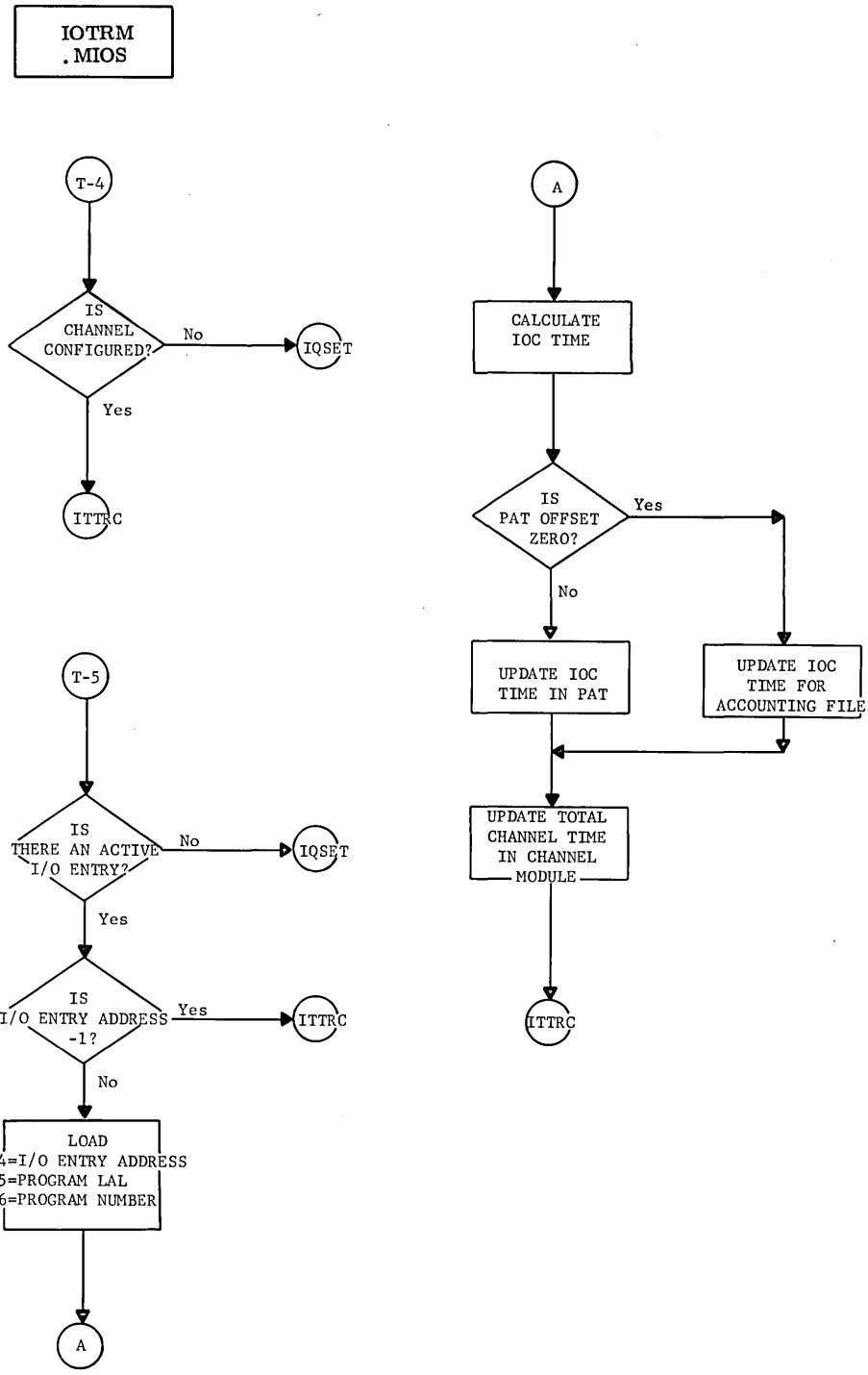
INTERRUPT HANDLER

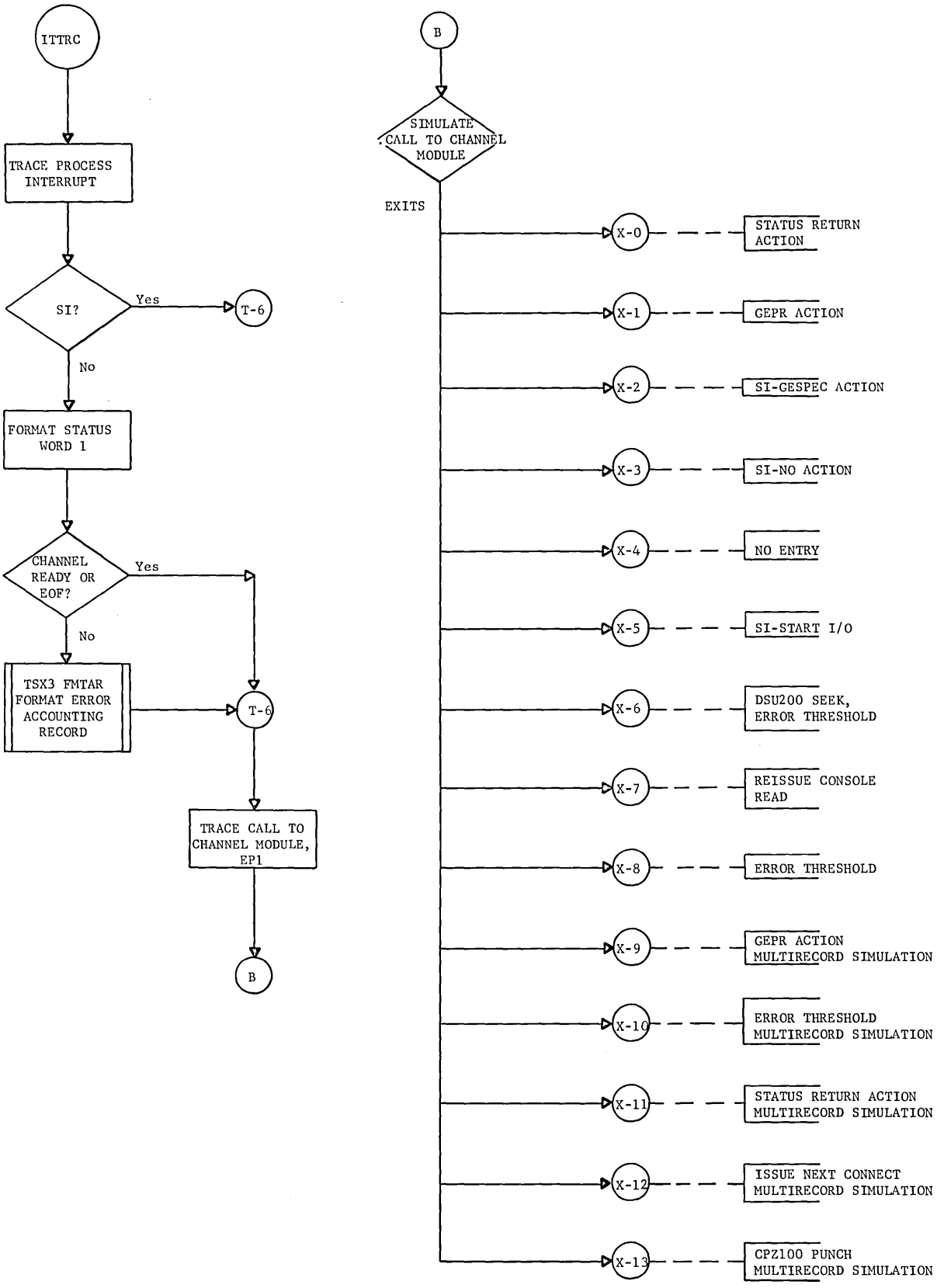


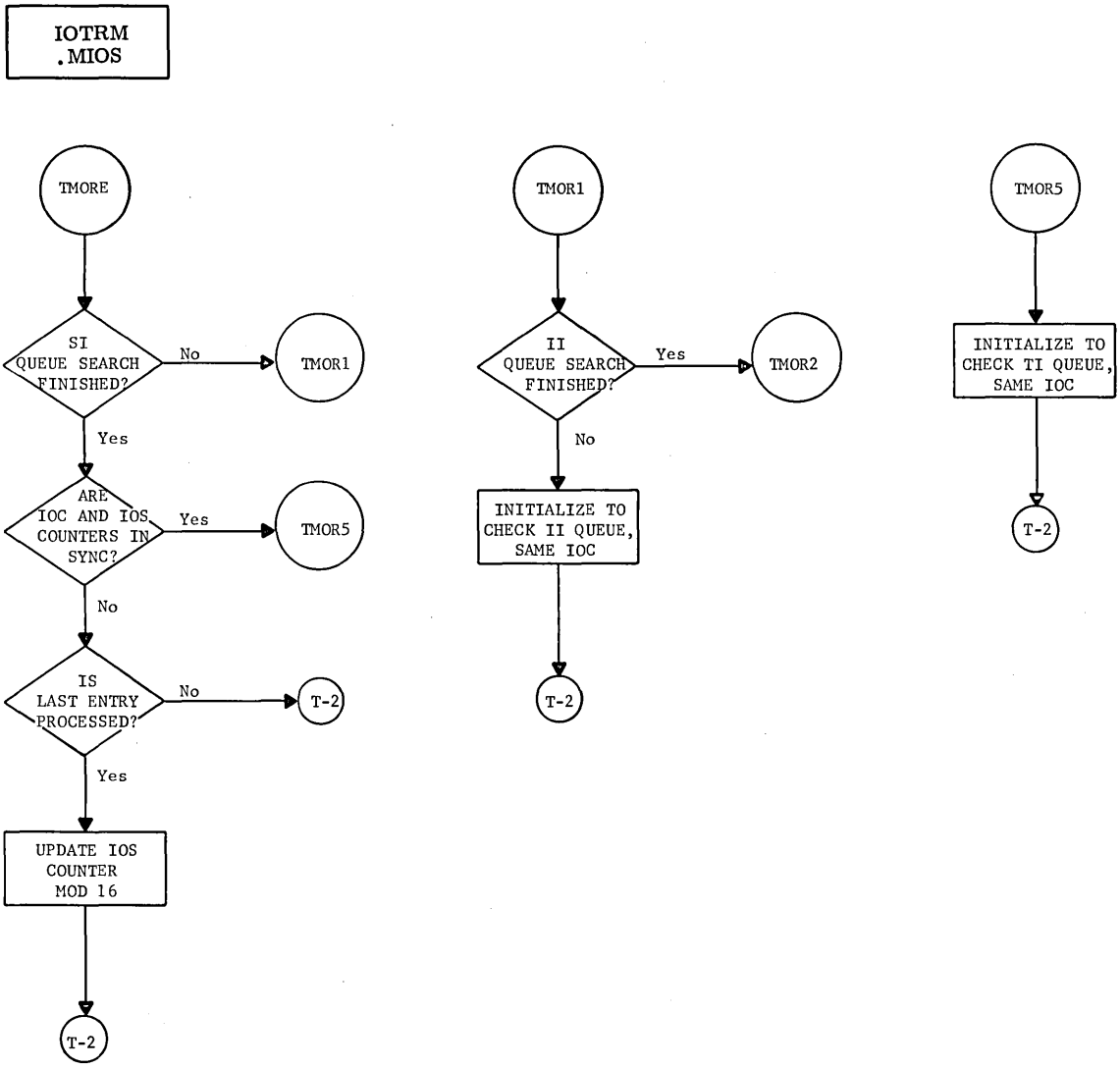

INTRC
LOAD X5 WITH GEPOP LAL
LOAD X7 WITH ZERO, CONTROL PROCESSOR
LOAD X1 WITH .CRPRG TO SAVE OVER GATE
XED .CRTRC
TRACE OFF
TRACE ON
SET UP AQ FOR TRACE ENTRY
TRA 1,0 TO TRACE
RESTORE .CRPRG
RELOAD REGISTERS
IOTRM

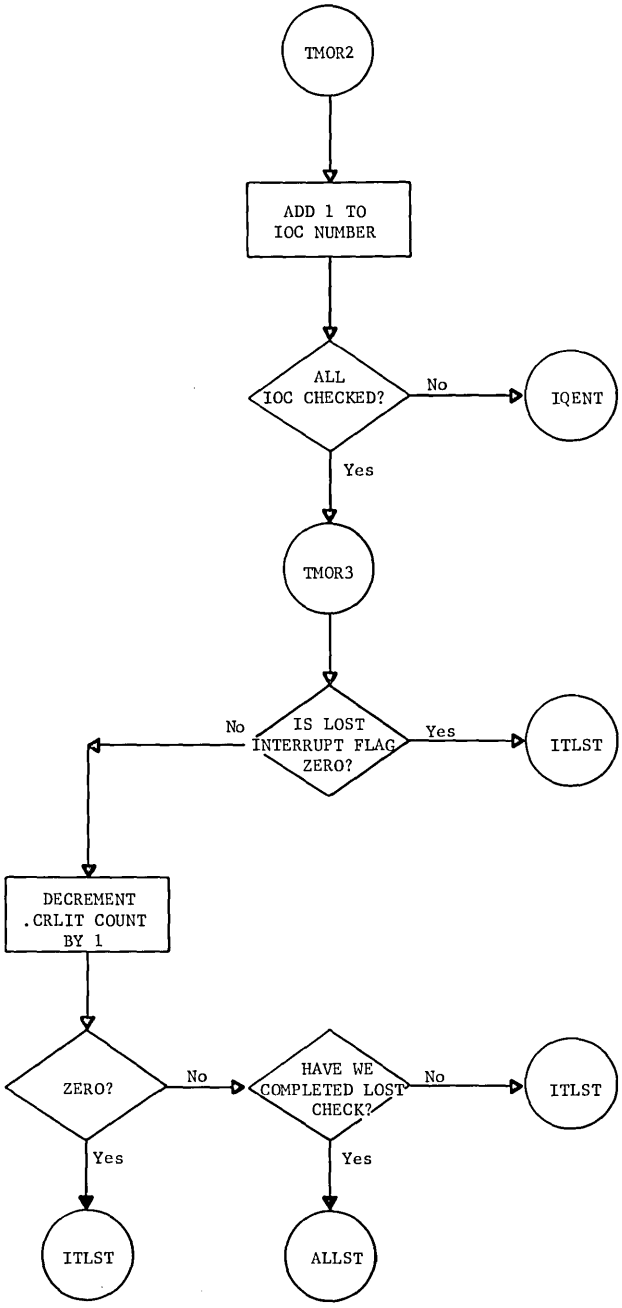

IOTRM .MIOS
IOTRM
INCREMENT .CRTIR (COUNT OF INTERRUPTS)
IS SOFTWARE GATE SHUT?
Yes
RETURN TO POINT OF INTER-RUPT
No
SET SOFTWARE GATE SHUT
WAS A PROGRAM IN EXECUTION?
No
TDISP
Yes
SAVE REGISTERS FOR INTERRUPTED PROGRAM
SET PROGRAM INTERRUPTED FLAG
LOAD X7 WITH ZERO FOR CONTROL PROCESSOR
SAVE IC AND I FOR PROGRAM LATER DISPATCHING
LOAD X5 AND X6 FOR INTERRUPTED PROGRAM
T-1
T-1
ZERO ENTRY FOR PROGRAM IN .CRPRQ
TSX1 .MDISP,9 PRO-GRAM PROCESSOR ACCOUNTING
.SHUTS .CRDSP GATE
WAS PROGRAM IN CC?
No
Yes
TSX1 .MDISP,8 PUT PROGRAM AT TAIL OF DISPATCHER QUEUE
TSX1 .MDISP,7 PUT PROGRAM AT HEAD OF DISPATCHER QUEUE
TRACE PROGRAM IN QUEUE
TRACE PROGRAM PUT IN QUEUE
.OPENS .CRDSP GATE
RESTORE .CRPRG
IQSET

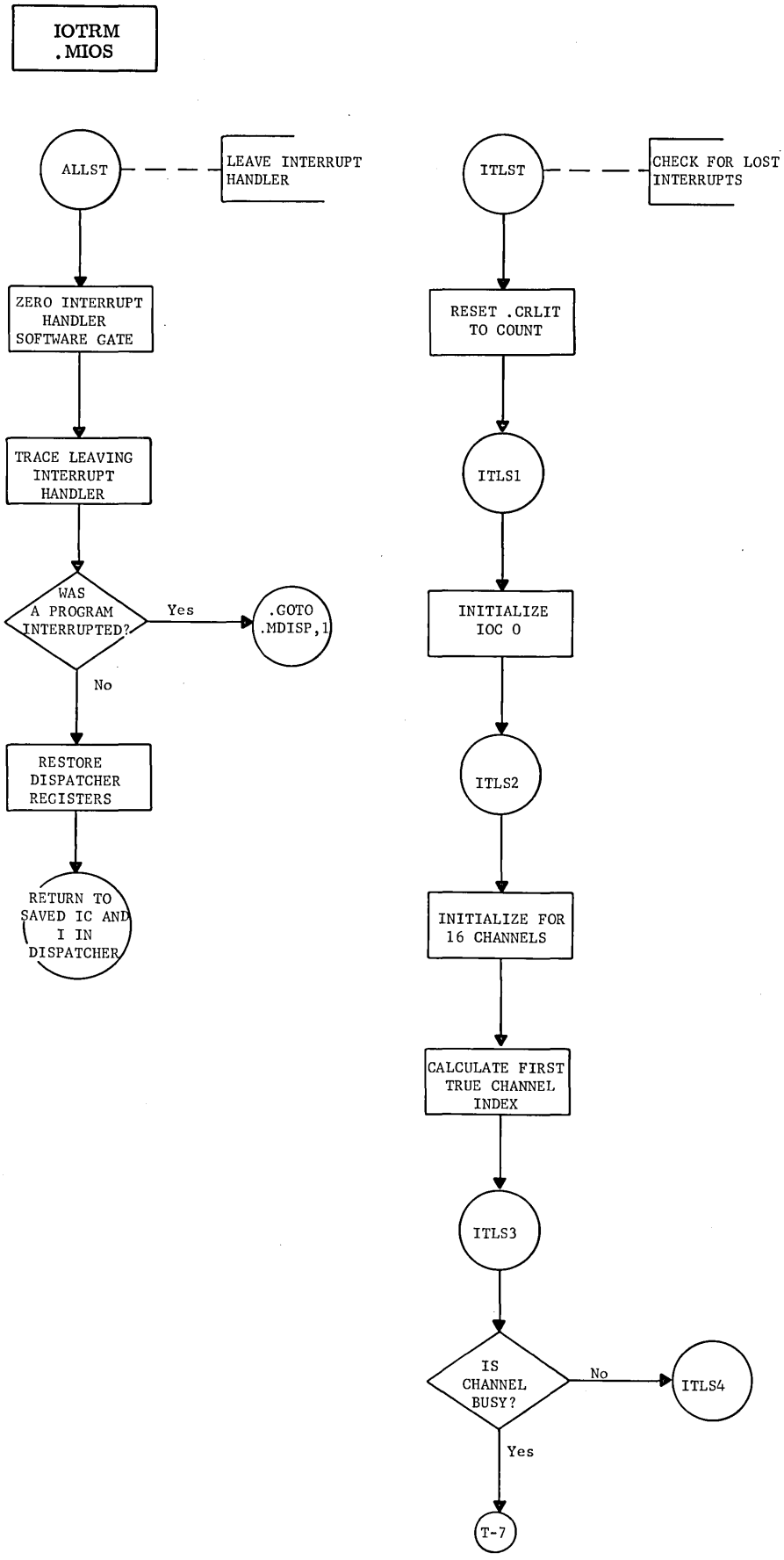

TDISP
SAVE DISPATCHER REGISTERS INTERNALLY
LOAD X7=0 FOR CONTROL PROCESSOR
SET NO PROGRAM INTERRUPTED FLAG
SAVE IC AND I OF INTERRUPT FOR EVENTUAL RETURN
IQSET
INITIALIZE FOR IOC-0
IQENT
INITIALIZE FOR IOC SPECIAL INTERRUPT QUEUE SEARCH
T-2
RETRIEVE SOFTWARE INTERRUPT COUNTER
IS NEXT ENTRY PROCESSED?
Yes
TMORE
No
T-3
T-3
MARK ENTRY PROCESSED AND UPDATE SOFTWARE COUNTER
SI?
No
IQPRC
Yes
IS PREVIOUS ENTRY PROCESSED?
Yes
IQPRC
No
.SHUTS .CRPOQ GATE
IS GEPOP QUEUE FULL?
Yes
No
PUT BURST OF SI ENTRY IN QUEUE
.OPENS .CRPOQ GATE
IQPRC
CALCULATE TRUE CHANNEL INDEX LOGICAL CHANNEL INDEX
SI?
Yes
T-5
No
T-4

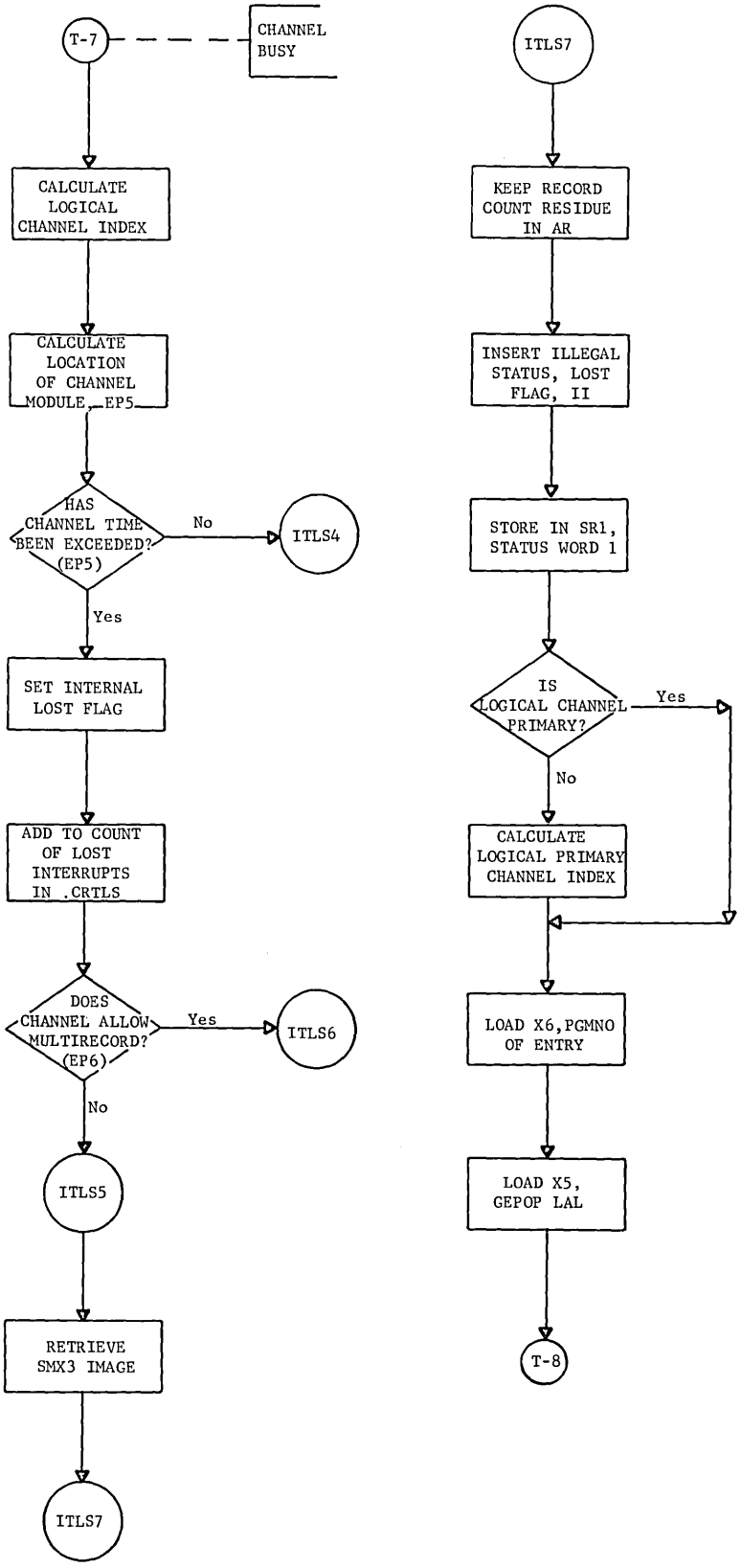

IOTRM
. MIOS
T-4
IS CHANNEL CONFIGURED?
No
IQSET
Yes
ITTRC
T-5
IS THERE AN ACTIVE I/O ENTRY?
No
IQSET
Yes
IS I/O ENTRY ADDRESS -1?
Yes
ITTRC
No
LOAD
X4=I/O ENTRY ADDRESS
X5=PROGRAM LAL
X6=PROGRAM NUMBER
A
A
CALCULATE IOC TIME
IS PAT OFFSET ZERO?
Yes
No
UPDATE IOC TIME IN PAT
UPDATE IOC TIME FOR ACCOUNTING FILE
UPDATE TOTAL CHANNEL TIME IN CHANNEL MODULE
ITTRC


ITTRC
TRACE PROCESS INTERRUPT
SI?
Yes
T-6
No
FORMAT STATUS WORD 1
CHANNEL READY OR EOF?
Yes
No
TSX3 FMTAR FORMAT ERROR ACCOUNTING RECORD
T-6
TRACE CALL TO CHANNEL MODULE, EP1
B
B
SIMULATE CALL TO CHANNEL MODULE
EXITS
X-0
STATUS RETURN ACTION
X-1
GEPR ACTION
X-2
SI-GESPEC ACTION
X-3
SI-NO ACTION
X-4
NO ENTRY
X-5
SI-START I/O
X-6
DSU200 SEEK, ERROR THRESHOLD
X-7
REISSUE CONSOLE READ
X-8
ERROR THRESHOLD
X-9
GEPR ACTION MULTIRECORD SIMULATION
X-10
ERROR THRESHOLD MULTIRECORD SIMULATION
X-11
STATUS RETURN ACTION MULTIRECORD SIMULATION
X-12
ISSUE NEXT CONNECT MULTIRECORD SIMULATION
X-13
CPZ100 PUNCH MULTIRECORD SIMULATION

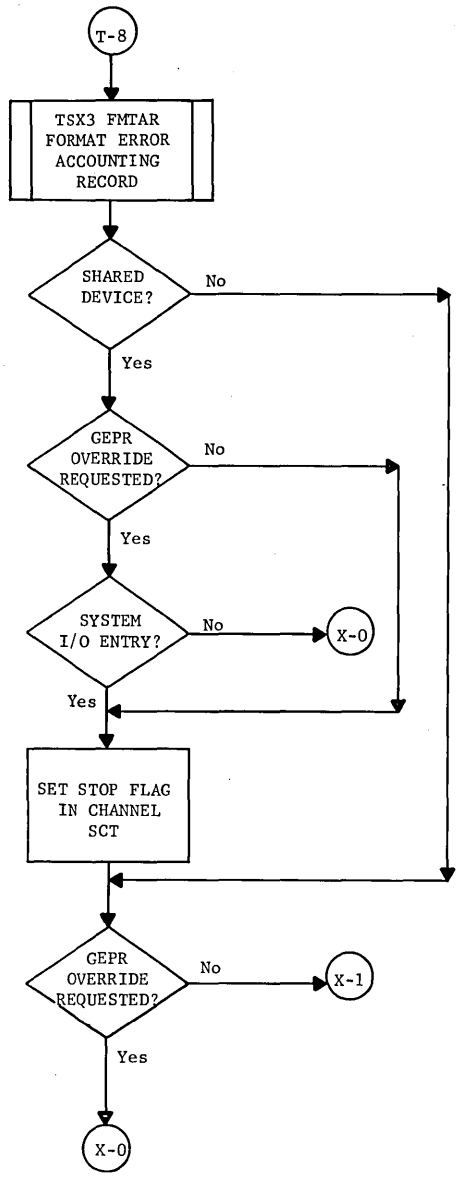

IOTRM
.MIOS
TMORE
SI
QUEUE SEARCH
FINISHED?
No
TMOR1
Yes
ARE
IOC AND IOS
COUNTERS IN
SYNC?
Yes
TMOR5
No
IS
LAST ENTRY
PROCESSED?
No
T-2
Yes
UPDATE IOS
COUNTER
MOD 16
T-2
TMOR1
II
QUEUE SEARCH
FINISHED?
Yes
TMOR2
No
INITIALIZE TO
CHECK II QUEUE,
SAME IOC
T-2
TMOR5
INITIALIZE TO
CHECK TI QUEUE,
SAME IOC
T-2

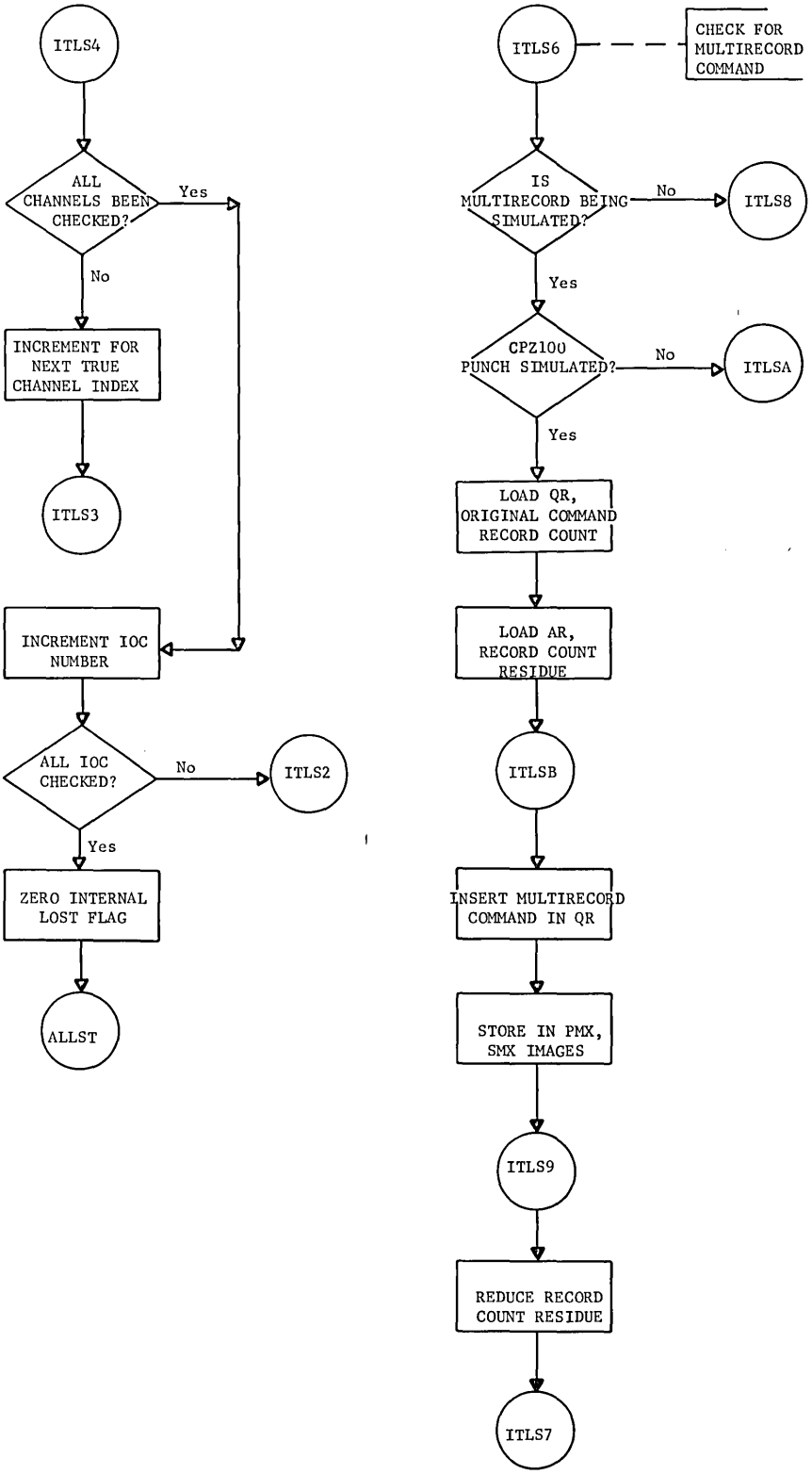

TMOR2
ADD 1 TO IOC NUMBER
ALL IOC CHECKED?
No
IQENT
Yes
TMOR3
IS LOST INTERRUPT FLAG ZERO?
No
Yes
ITLST
DECREMENT .CRLIT COUNT BY 1
ZERO?
No
HAVE WE COMPLETED LOST CHECK?
No
ITLST
Yes
Yes
ITLST
ALLST


IOTRM
.MIOS
ALLST
LEAVE INTERRUPT HANDLER
ZERO INTERRUPT HANDLER SOFTWARE GATE
TRACE LEAVING INTERRUPT HANDLER
WAS A PROGRAM INTERRUPTED?
Yes
.GOTO .MDISP,1
No
RESTORE DISPATCHER REGISTERS
RETURN TO SAVED IC AND I IN DISPATCHER
ITLST
CHECK FOR LOST INTERRUPTS
RESET .CRLIT TO COUNT
ITLS1
INITIALIZE IOC 0
ITLS2
INITIALIZE FOR 16 CHANNELS
CALCULATE FIRST TRUE CHANNEL INDEX
ITLS3
IS CHANNEL BUSY?
No
ITLS4
Yes
T-7

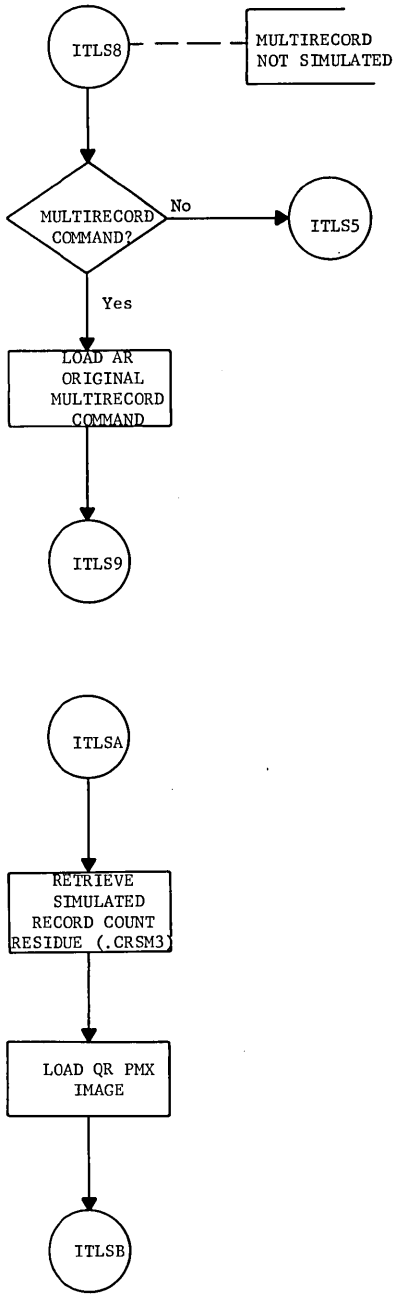

T-7
CHANNEL BUSY
CALCULATE LOGICAL CHANNEL INDEX
CALCULATE LOCATION OF CHANNEL MODULE, EP5
HAS CHANNEL TIME BEEN EXCEEDED? (EP5)
No
ITLS4
Yes
SET INTERNAL LOST FLAG
ADD TO COUNT OF LOST INTERRUPTS IN .CRTLS
DOES CHANNEL ALLOW MULTIRECORD? (EP6)
Yes
ITLS6
No
ITLS5
RETRIEVE SMX3 IMAGE
ITLS7
ITLS7
KEEP RECORD COUNT RESIDUE IN AR
INSERT ILLEGAL STATUS, LOST FLAG, II
STORE IN SR1, STATUS WORD 1
IS LOGICAL CHANNEL PRIMARY?
Yes
No
CALCULATE LOGICAL PRIMARY CHANNEL INDEX
LOAD X6, PGMNO OF ENTRY
LOAD X5, GEPOP LAL
T-8

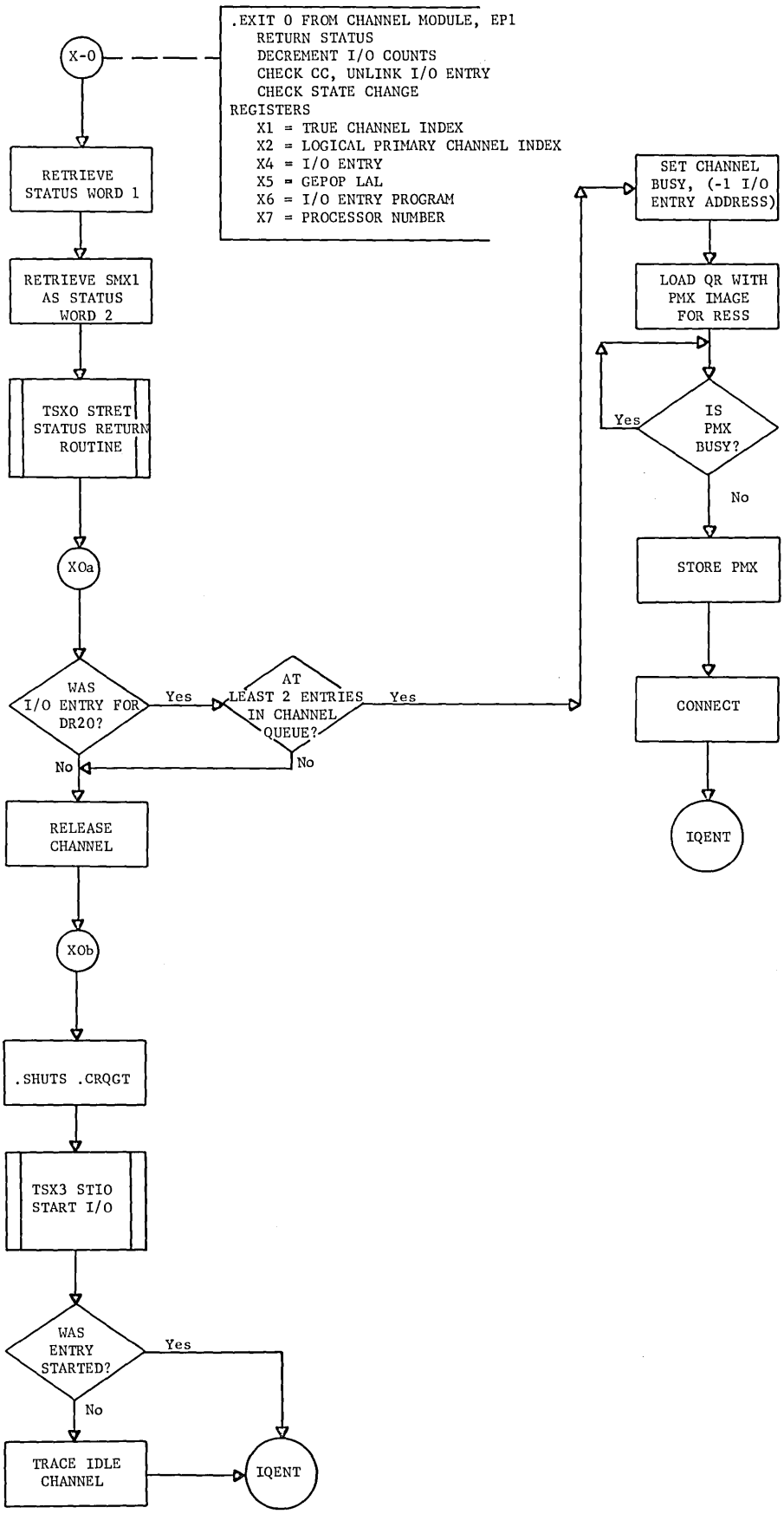
IOTRM
.MIOS


T-8
TSX3 FMTAR
FORMAT ERROR
ACCOUNTING
RECORD
SHARED
DEVICE?
No
Yes
GEPR
OVERRIDE
REQUESTED?
No
Yes
SYSTEM
I/O ENTRY?
No
X-0
Yes
SET STOP FLAG
IN CHANNEL
SCT
GEPR
OVERRIDE
REQUESTED?
No
X-1
Yes
X-0

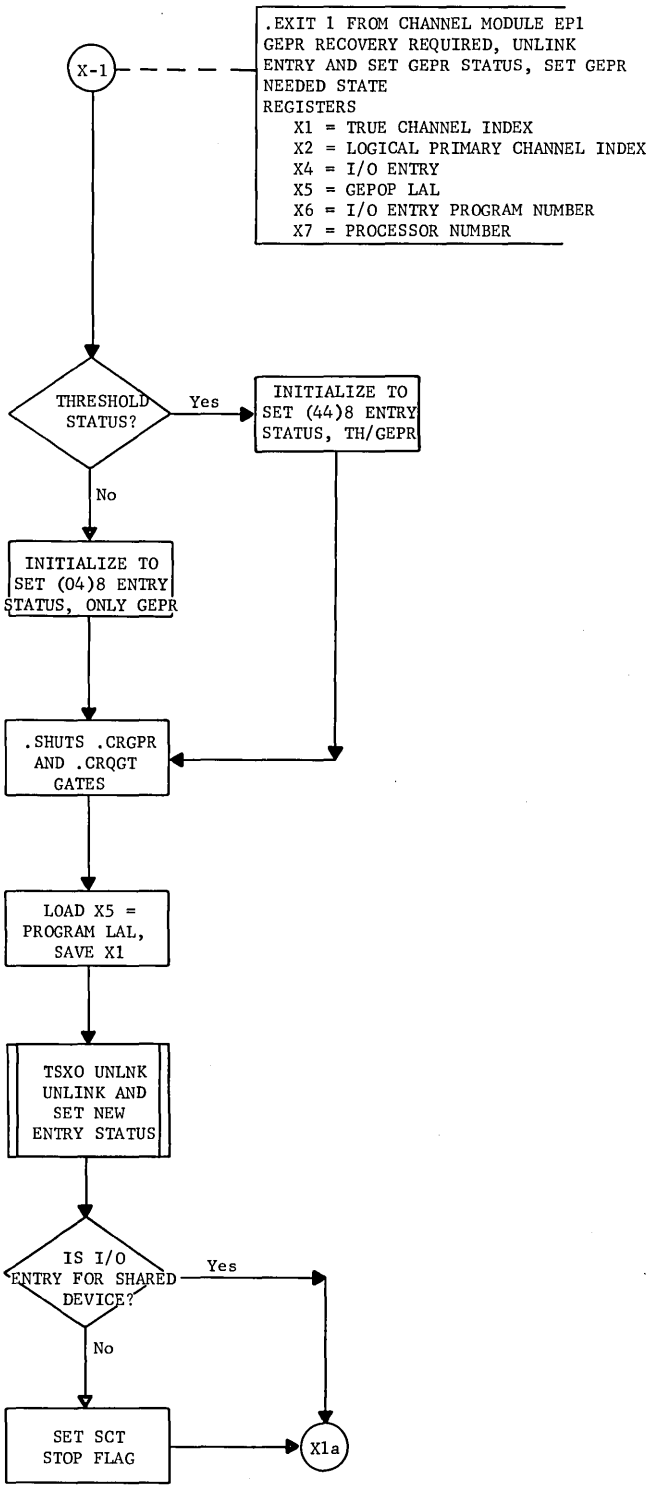

ITLS4
ALL CHANNELS BEEN CHECKED?
Yes
No
INCREMENT FOR NEXT TRUE CHANNEL INDEX
ITLS3
INCREMENT IOC NUMBER
ALL IOC CHECKED?
No
ITLS2
Yes
ZERO INTERNAL LOST FLAG
ALLST
ITLS6
CHECK FOR MULTIRECORD COMMAND
IS MULTIRECORD BEING SIMULATED?
No
ITLS8
Yes
CPZ100 PUNCH SIMULATED?
No
ITLSA
Yes
LOAD QR, ORIGINAL COMMAND RECORD COUNT
LOAD AR, RECORD COUNT RESIDUE
ITLSB
INSERT MULTIRECORD COMMAND IN QR
STORE IN PMX, SMX IMAGES
ITLS9
REDUCE RECORD COUNT RESIDUE
ITLS7

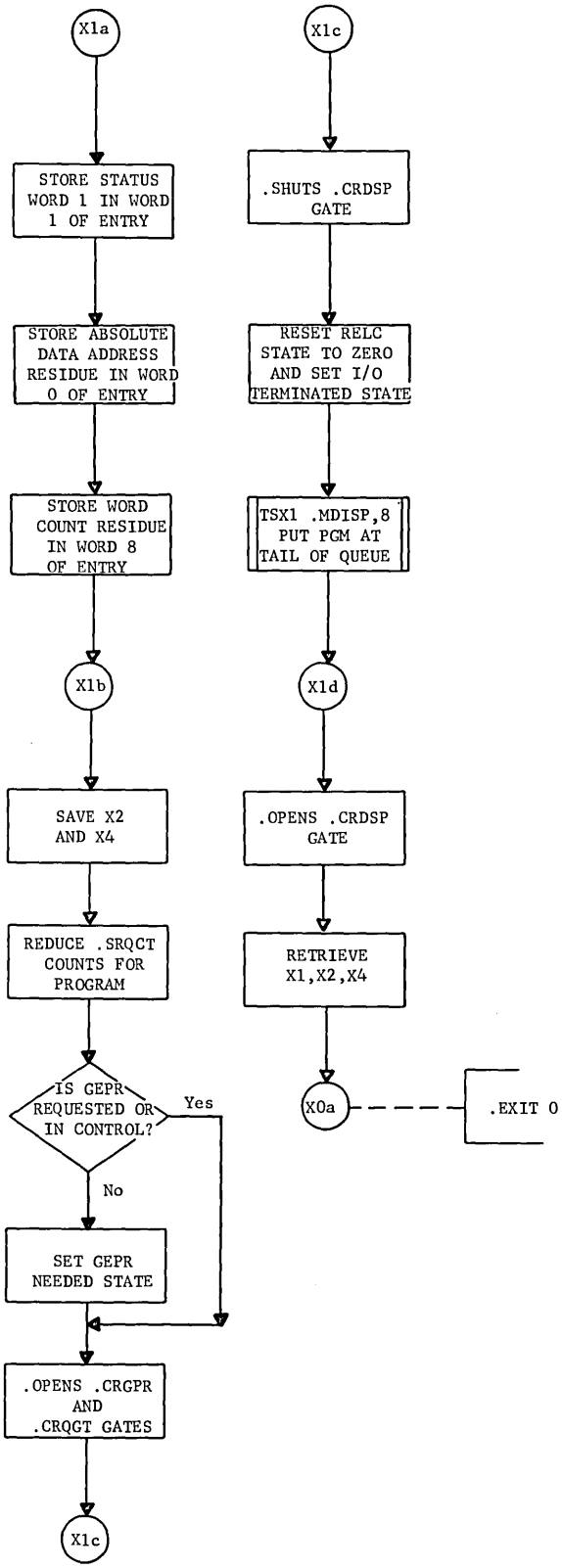
IOTRM
.MIOS

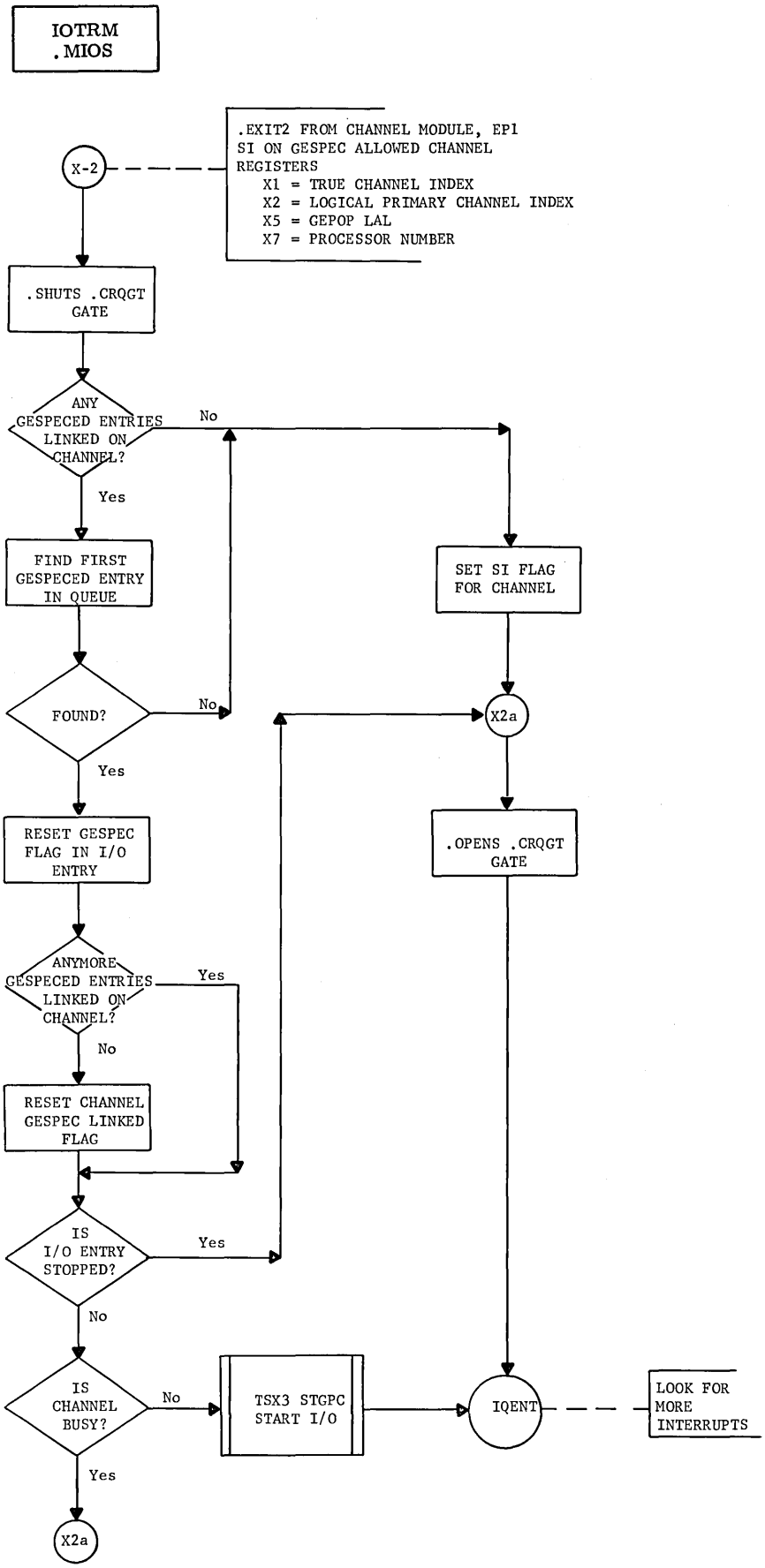

ITLS8
MULTIRECORD NOT SIMULATED
MULTIRECORD COMMAND?
No
ITLS5
Yes
LOAD AR ORIGINAL MULTIRECORD COMMAND
ITLS9
ITLSA
RETRIEVE SIMULATED RECORD COUNT RESIDUE (.CRSM3)
LOAD QR PMX IMAGE
ITLSB

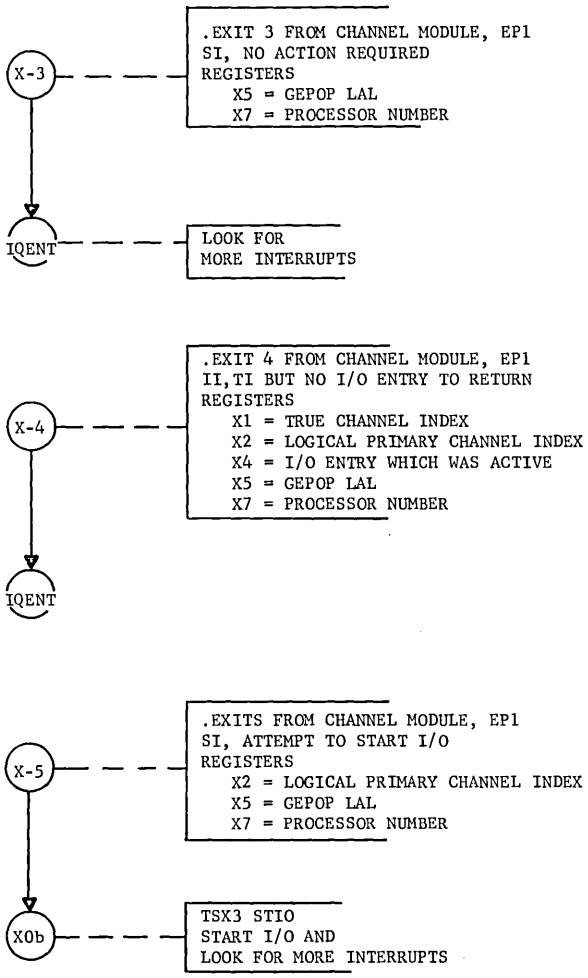

.EXIT 0 FROM CHANNEL MODULE, EP1
RETURN STATUS
DECREMENT I/O COUNTS
CHECK CC, UNLINK I/O ENTRY
CHECK STATE CHANGE
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = I/O ENTRY PROGRAM
X7 = PROCESSOR NUMBER
X-0
RETRIEVE STATUS WORD 1
RETRIEVE SMX1 AS STATUS WORD 2
TSXO STRET STATUS RETURN ROUTINE
XOa
WAS I/O ENTRY FOR DR20?
Yes
AT LEAST 2 ENTRIES IN CHANNEL QUEUE?
Yes
No
No
RELEASE CHANNEL
XOb
.SHUTS .CRQGT
TSX3 STIO START I/O
WAS ENTRY STARTED?
Yes
No
TRACE IDLE CHANNEL
IQENT
SET CHANNEL BUSY, (-1 I/O ENTRY ADDRESS)
LOAD QR WITH PMX IMAGE FOR RESS
Yes
IS PMX BUSY?
No
STORE PMX
CONNECT
IQENT

IOTRM
.MIOS

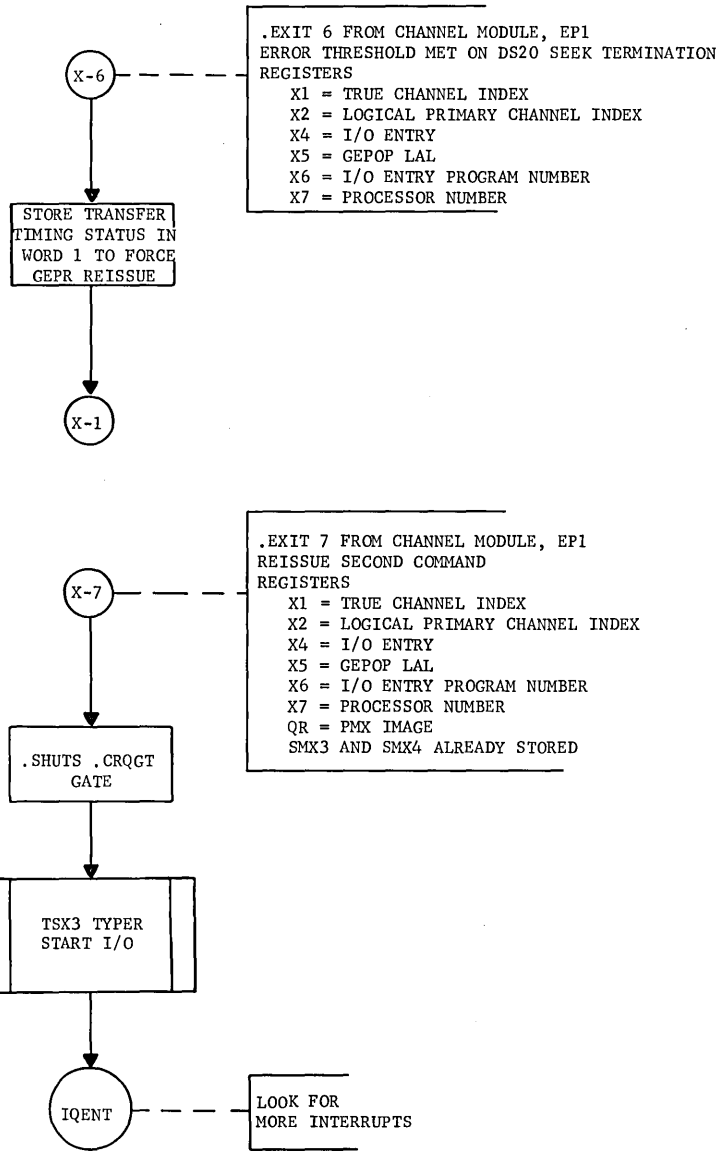

.EXIT 1 FROM CHANNEL MODULE EP1
GEPR RECOVERY REQUIRED, UNLINK
ENTRY AND SET GEPR STATUS, SET GEPR
NEEDED STATE
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = I/O ENTRY PROGRAM NUMBER
X7 = PROCESSOR NUMBER
X-1
THRESHOLD STATUS?
Yes
No
INITIALIZE TO SET (44)8 ENTRY STATUS, TH/GEPR
INITIALIZE TO SET (04)8 ENTRY STATUS, ONLY GEPR
.SHUTS .CRGPR AND .CRQGT GATES
LOAD X5 = PROGRAM LAL, SAVE X1
TSXO UNLNK UNLINK AND SET NEW ENTRY STATUS
IS I/O ENTRY FOR SHARED DEVICE?
Yes
No
SET SCT STOP FLAG
X1a

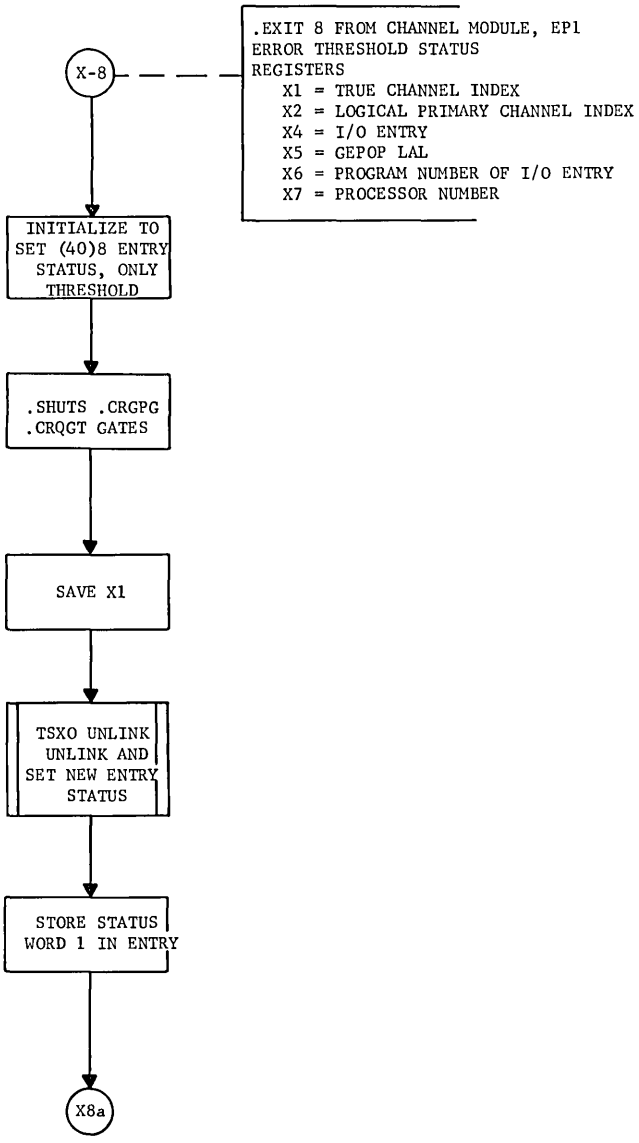

X1a
STORE STATUS WORD 1 IN WORD 1 OF ENTRY
STORE ABSOLUTE DATA ADDRESS RESIDUE IN WORD 0 OF ENTRY
STORE WORD COUNT RESIDUE IN WORD 8 OF ENTRY
X1b
SAVE X2 AND X4
REDUCE .SRQCT COUNTS FOR PROGRAM
IS GEPR REQUESTED OR IN CONTROL?
Yes
No
SET GEPR NEEDED STATE
.OPENS .CRGPR AND .CRQGT GATES
X1c
X1c
.SHUTS .CRDSP GATE
RESET RELC STATE TO ZERO AND SET I/O TERMINATED STATE
TSX1 .MDISP,8 PUT PGM AT TAIL OF QUEUE
X1d
.OPENS .CRDSP GATE
RETRIEVE X1,X2,X4
X0a
.EXIT 0


IOTRM
.MIOS
X-2
.EXIT2 FROM CHANNEL MODULE, EP1
SI ON GESPEC ALLOWED CHANNEL
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X5 = GEPOP LAL
X7 = PROCESSOR NUMBER
.SHUTS .CRQGT GATE
ANY GESPECED ENTRIES LINKED ON CHANNEL?
No
Yes
FIND FIRST GESPECED ENTRY IN QUEUE
SET SI FLAG FOR CHANNEL
FOUND?
No
Yes
X2a
RESET GESPEC FLAG IN I/O ENTRY
.OPENS .CRQGT GATE
ANYMORE GESPECED ENTRIES LINKED ON CHANNEL?
Yes
No
RESET CHANNEL GESPEC LINKED FLAG
IS I/O ENTRY STOPPED?
Yes
No
IS CHANNEL BUSY?
No
TSX3 STGPC START I/O
IQENT
LOOK FOR MORE INTERRUPTS
Yes
X2a

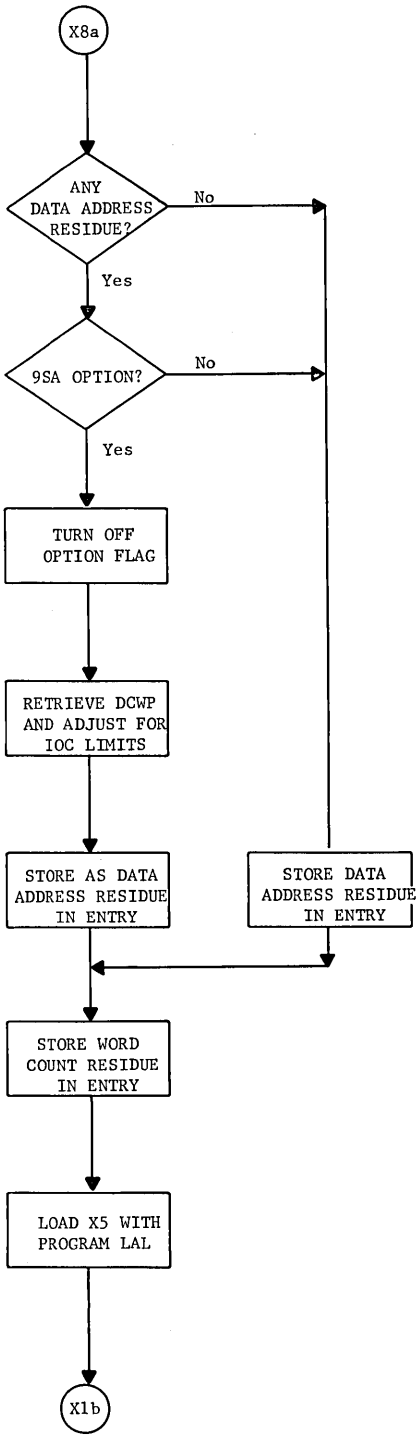

X-3
.EXIT 3 FROM CHANNEL MODULE, EP1
SI, NO ACTION REQUIRED
REGISTERS
X5 = GEPOP LAL
X7 = PROCESSOR NUMBER
IQENT
LOOK FOR
MORE INTERRUPTS
X-4
.EXIT 4 FROM CHANNEL MODULE, EP1
II,TI BUT NO I/O ENTRY TO RETURN
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY WHICH WAS ACTIVE
X5 = GEPOP LAL
X7 = PROCESSOR NUMBER
IQENT
X-5
.EXITS FROM CHANNEL MODULE, EP1
SI, ATTEMPT TO START I/O
REGISTERS
X2 = LOGICAL PRIMARY CHANNEL INDEX
X5 = GEPOP LAL
X7 = PROCESSOR NUMBER
X0b
TSX3 STIO
START I/O AND
LOOK FOR MORE INTERRUPTS

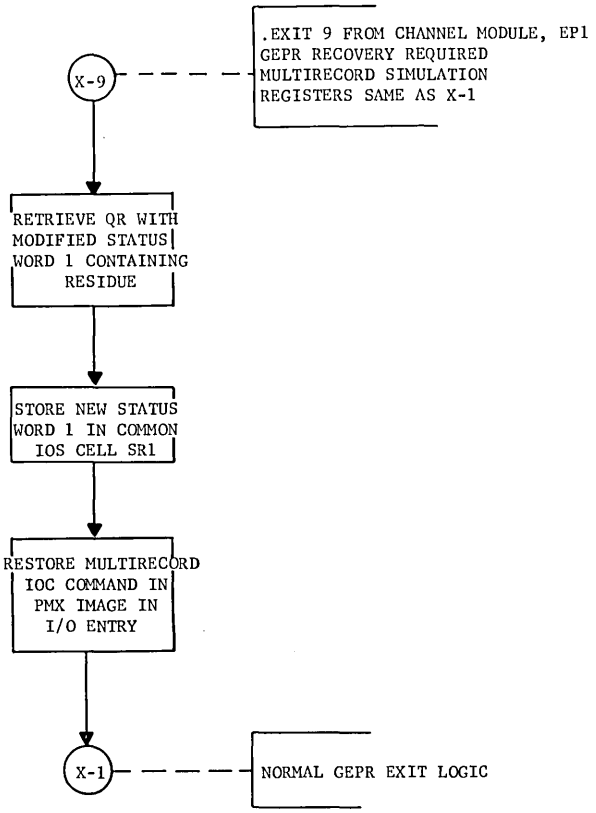
IOTRM
.MIOS


X-6
.EXIT 6 FROM CHANNEL MODULE, EP1
ERROR THRESHOLD MET ON DS20 SEEK TERMINATION
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = I/O ENTRY PROGRAM NUMBER
X7 = PROCESSOR NUMBER
STORE TRANSFER TIMING STATUS IN WORD 1 TO FORCE GEPR REISSUE
X-1
X-7
.EXIT 7 FROM CHANNEL MODULE, EP1
REISSUE SECOND COMMAND
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = I/O ENTRY PROGRAM NUMBER
X7 = PROCESSOR NUMBER
QR = PMX IMAGE
SMX3 AND SMX4 ALREADY STORED
.SHUTS .CRQGT GATE
TSX3 TYPER START I/O
IQENT
LOOK FOR MORE INTERRUPTS

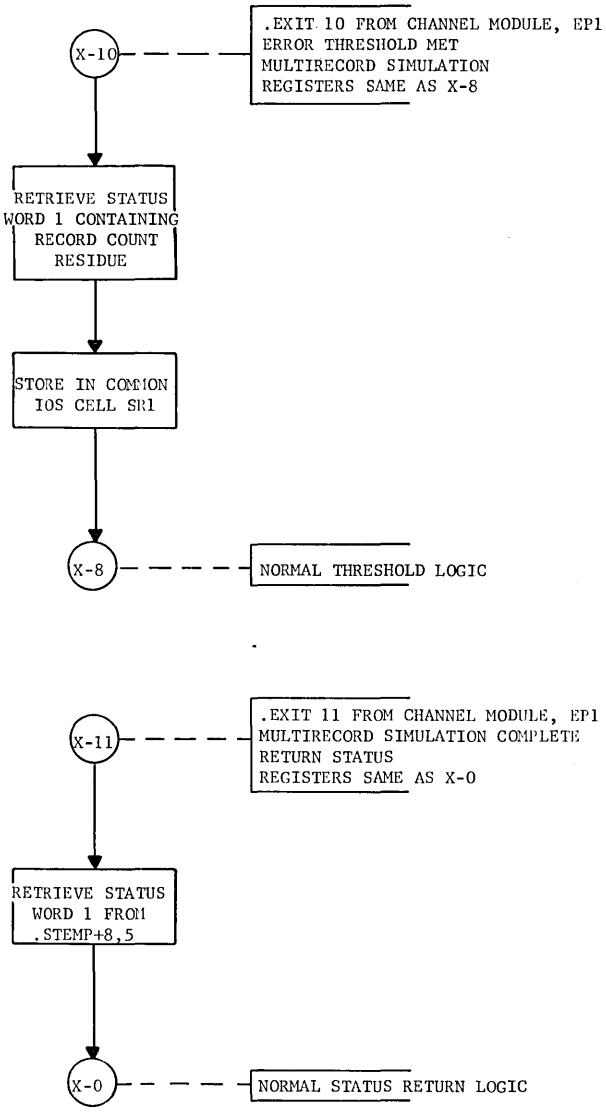

.EXIT 8 FROM CHANNEL MODULE, EP1
ERROR THRESHOLD STATUS
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = PROGRAM NUMBER OF I/O ENTRY
X7 = PROCESSOR NUMBER
X-8
INITIALIZE TO SET (40)8 ENTRY STATUS, ONLY THRESHOLD
.SHUTS .CRGPG .CRQGT GATES
SAVE X1
TSXO UNLINK UNLINK AND SET NEW ENTRY STATUS
STORE STATUS WORD 1 IN ENTRY
X8a

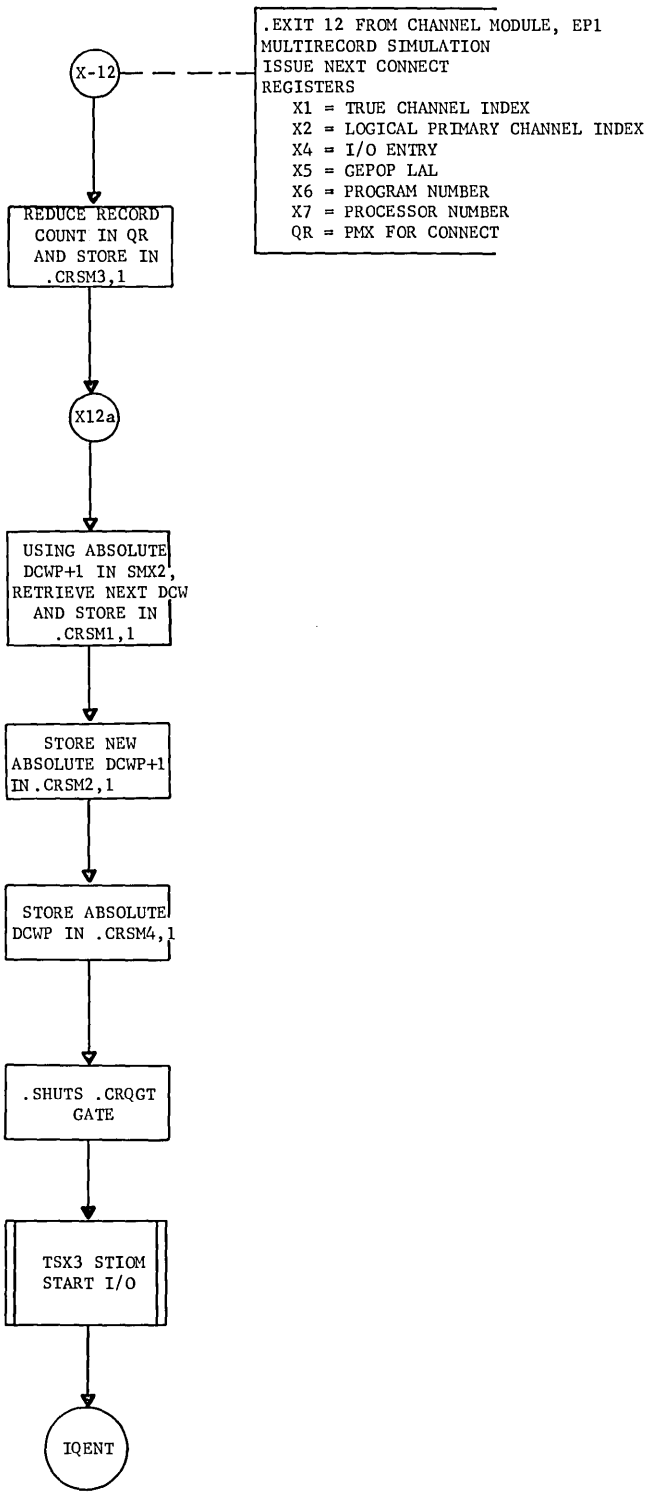
IOTRM
.MIOS

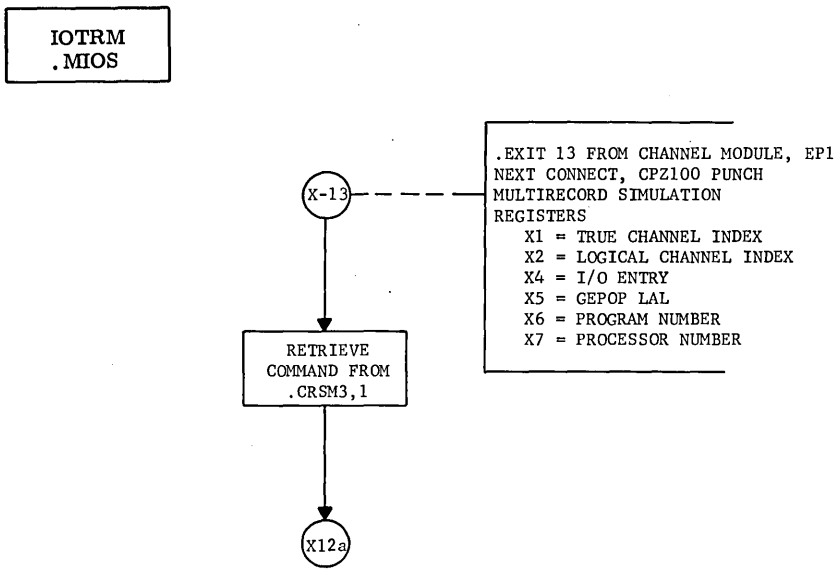

X8a
ANY DATA ADDRESS RESIDUE?
No
Yes
9SA OPTION?
No
Yes
TURN OFF OPTION FLAG
RETRIEVE DCWP AND ADJUST FOR IOC LIMITS
STORE AS DATA ADDRESS RESIDUE IN ENTRY
STORE DATA ADDRESS RESIDUE IN ENTRY
STORE WORD COUNT RESIDUE IN ENTRY
LOAD X5 WITH PROGRAM LAL
X1b


.EXIT 9 FROM CHANNEL MODULE, EP1
GEPR RECOVERY REQUIRED
MULTIRECORD SIMULATION
REGISTERS SAME AS X-1
X-9
RETRIEVE QR WITH MODIFIED STATUS WORD 1 CONTAINING RESIDUE
STORE NEW STATUS WORD 1 IN COMMON IOS CELL SR1
RESTORE MULTIRECORD IOC COMMAND IN PMX IMAGE IN I/O ENTRY
X-1
NORMAL GEPR EXIT LOGIC

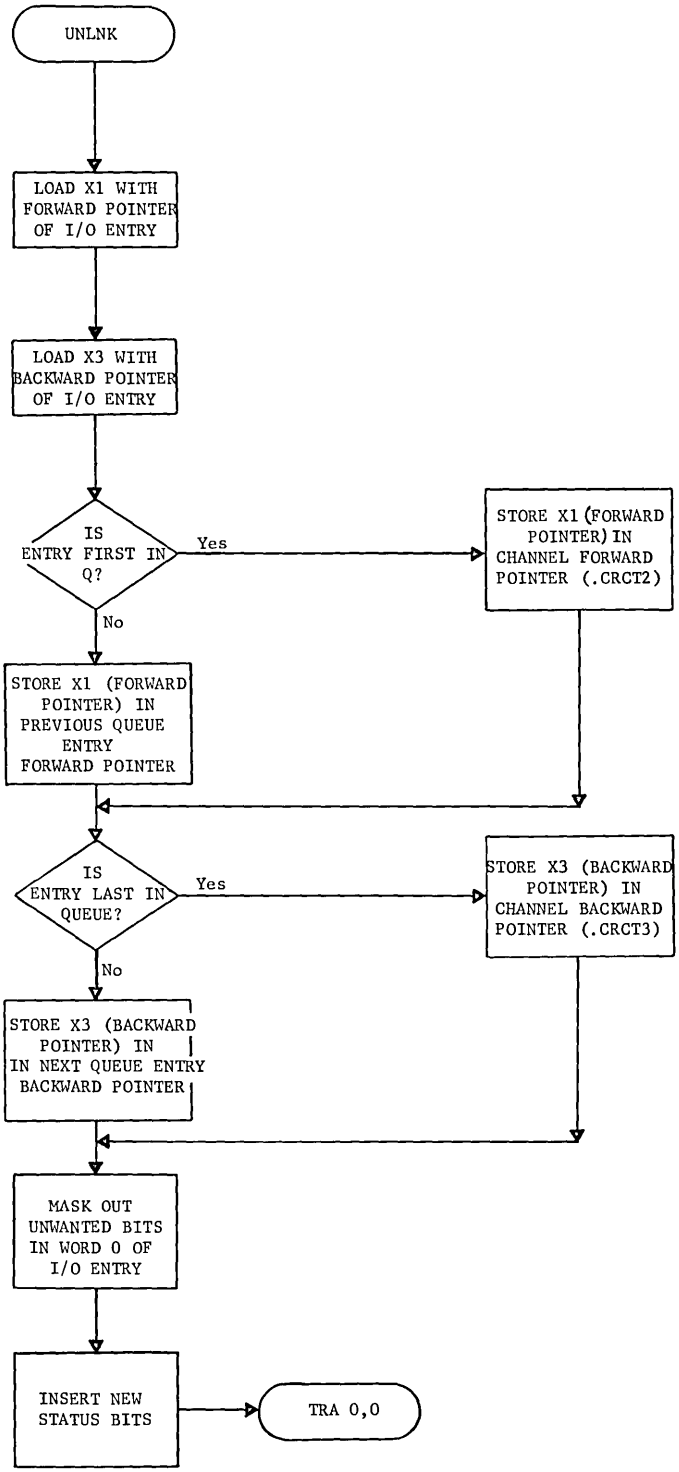
IOTRM
.MIOS


X-10
.EXIT 10 FROM CHANNEL MODULE, EP1
ERROR THRESHOLD MET
MULTIRECORD SIMULATION
REGISTERS SAME AS X-8
RETRIEVE STATUS WORD 1 CONTAINING RECORD COUNT RESIDUE
STORE IN COMMON IOS CELL SR1
X-8
NORMAL THRESHOLD LOGIC
X-11
.EXIT 11 FROM CHANNEL MODULE, EP1
MULTIRECORD SIMULATION COMPLETE
RETURN STATUS
REGISTERS SAME AS X-0
RETRIEVE STATUS WORD 1 FROM .STEMP+8,5
X-0
NORMAL STATUS RETURN LOGIC


.EXIT 12 FROM CHANNEL MODULE, EP1
MULTIRECORD SIMULATION
ISSUE NEXT CONNECT
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL PRIMARY CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = PROGRAM NUMBER
X7 = PROCESSOR NUMBER
QR = PMX FOR CONNECT
X-12
REDUCE RECORD COUNT IN QR AND STORE IN .CRSM3,1
X12a
USING ABSOLUTE DCWP+1 IN SMX2, RETRIEVE NEXT DCW AND STORE IN .CRSM1,1
STORE NEW ABSOLUTE DCWP+1 IN .CRSM2,1
STORE ABSOLUTE DCWP IN .CRSM4,1
.SHUTS .CRQGT GATE
TSX3 STIOM START I/O
IQENT

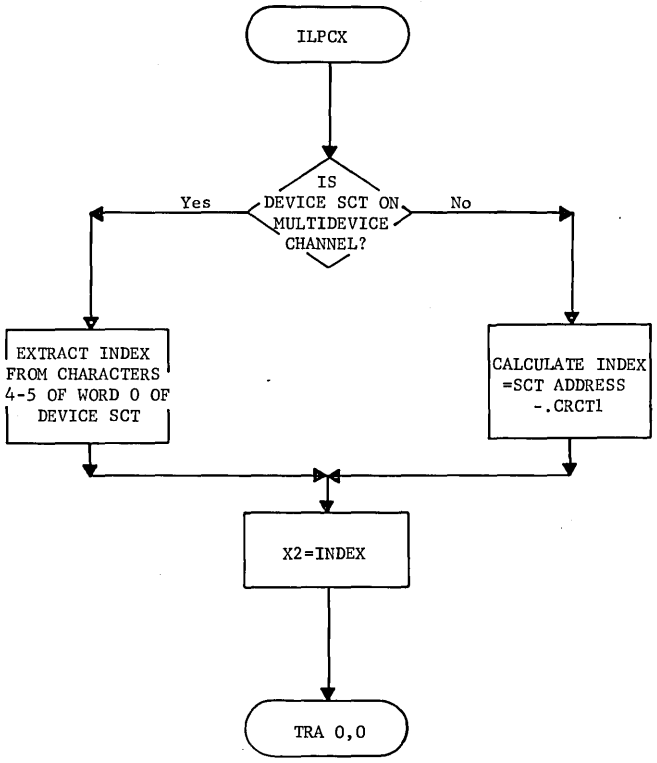

IOTRM
.MIOS
X-13
.EXIT 13 FROM CHANNEL MODULE, EP1
NEXT CONNECT, CPZ100 PUNCH
MULTIRECORD SIMULATION
REGISTERS
X1 = TRUE CHANNEL INDEX
X2 = LOGICAL CHANNEL INDEX
X4 = I/O ENTRY
X5 = GEPOP LAL
X6 = PROGRAM NUMBER
X7 = PROCESSOR NUMBER
RETRIEVE
COMMAND FROM
.CRSM3,1
X12a

UNLINK I/O ENTRY

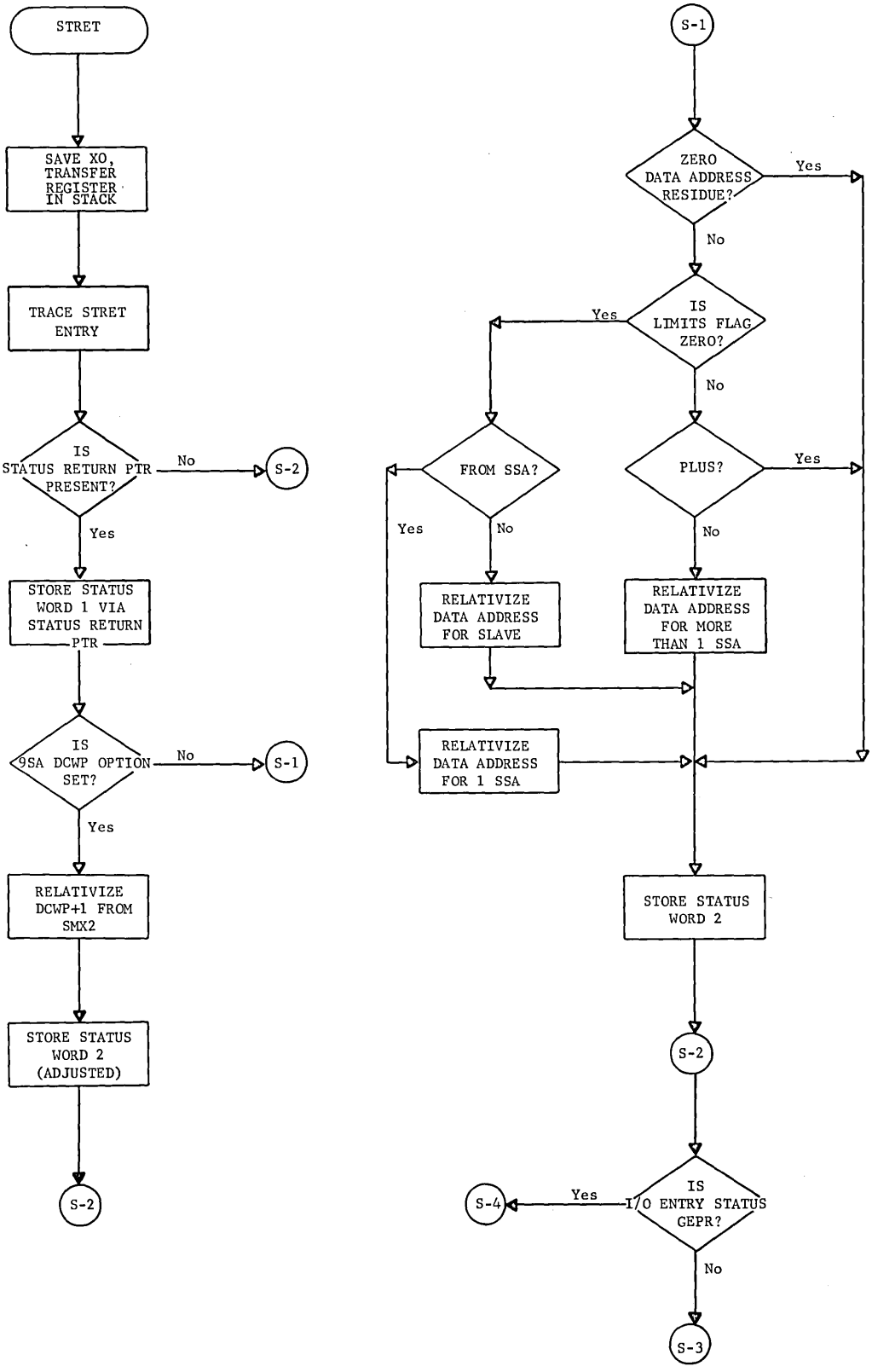


ILPCX
.MIOS

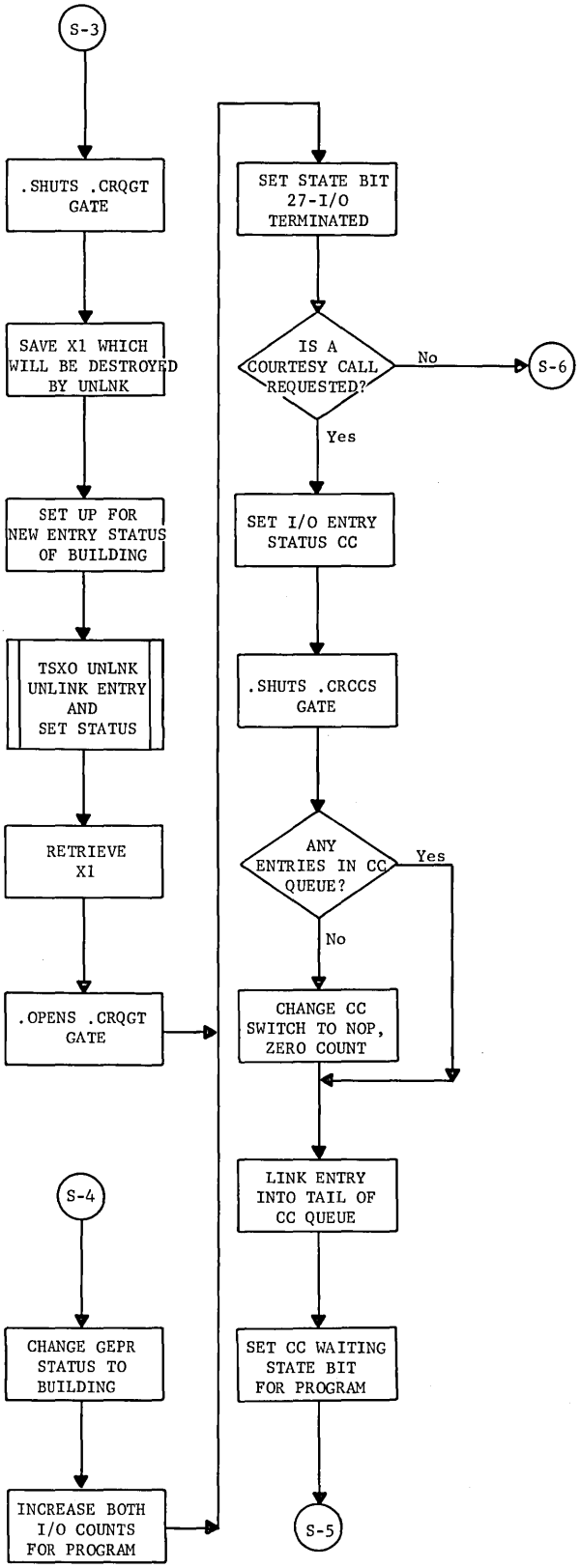
CALCULATE LOGICAL PRIMARY CHANNEL INDEX

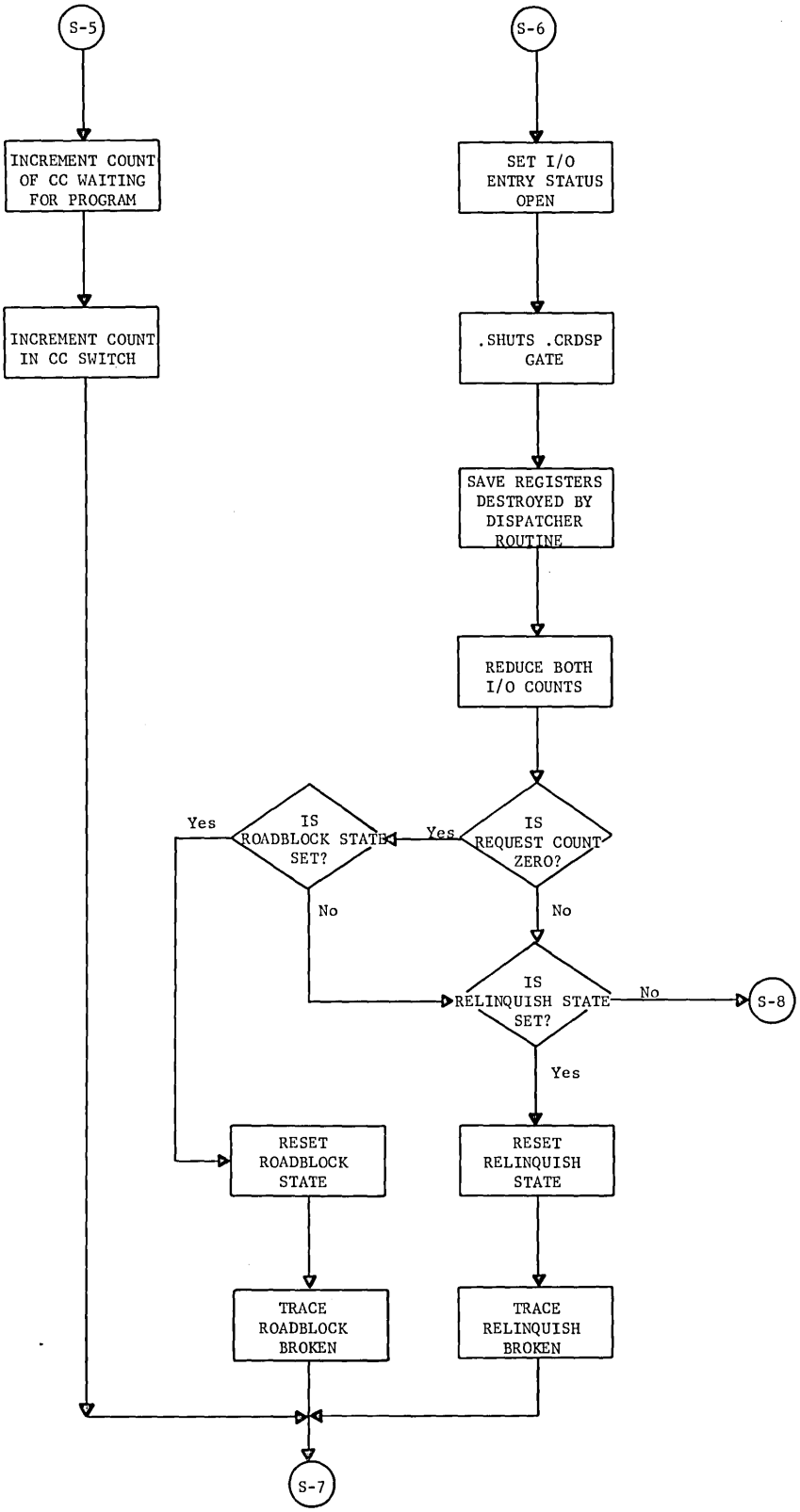


STATUS RETURN

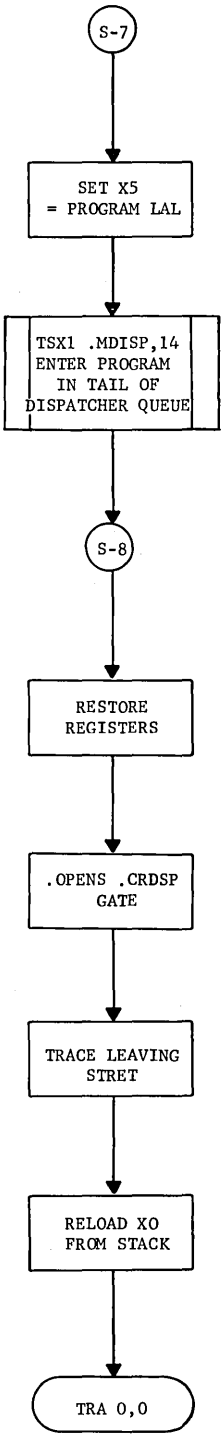


STRET
.MIOS

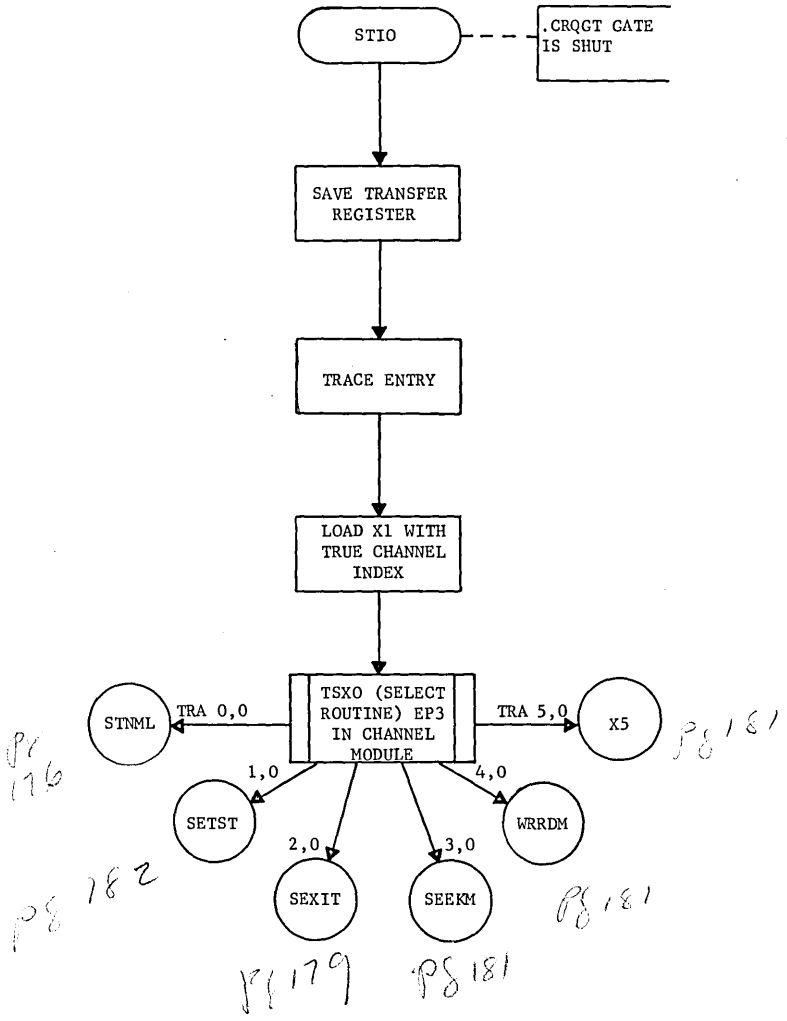

S-3
.SHUTS .CRQGT GATE
SAVE X1 WHICH WILL BE DESTROYED BY UNLNK
SET UP FOR NEW ENTRY STATUS OF BUILDING
TSXO UNLNK UNLINK ENTRY AND SET STATUS
RETRIEVE X1
.OPENS .CRQGT GATE
S-4
CHANGE GEPR STATUS TO BUILDING
INCREASE BOTH I/O COUNTS FOR PROGRAM
SET STATE BIT 27-I/O TERMINATED
IS A COURTESY CALL REQUESTED?
No
S-6
Yes
SET I/O ENTRY STATUS CC
.SHUTS .CRCCS GATE
ANY ENTRIES IN CC QUEUE?
Yes
No
CHANGE CC SWITCH TO NOP, ZERO COUNT
LINK ENTRY INTO TAIL OF CC QUEUE
SET CC WAITING STATE BIT FOR PROGRAM
S-5


S-5
S-6
INCREMENT COUNT OF CC WAITING FOR PROGRAM
INCREMENT COUNT IN CC SWITCH
SET I/O ENTRY STATUS OPEN
.SHUTS .CRDSP GATE
SAVE REGISTERS DESTROYED BY DISPATCHER ROUTINE
REDUCE BOTH I/O COUNTS
IS REQUEST COUNT ZERO?
Yes
IS ROADBLOCK STATE SET?
Yes
No
No
IS RELINQUISH STATE SET?
No
S-8
Yes
RESET ROADBLOCK STATE
RESET RELINQUISH STATE
TRACE ROADBLOCK BROKEN
TRACE RELINQUISH BROKEN
S-7

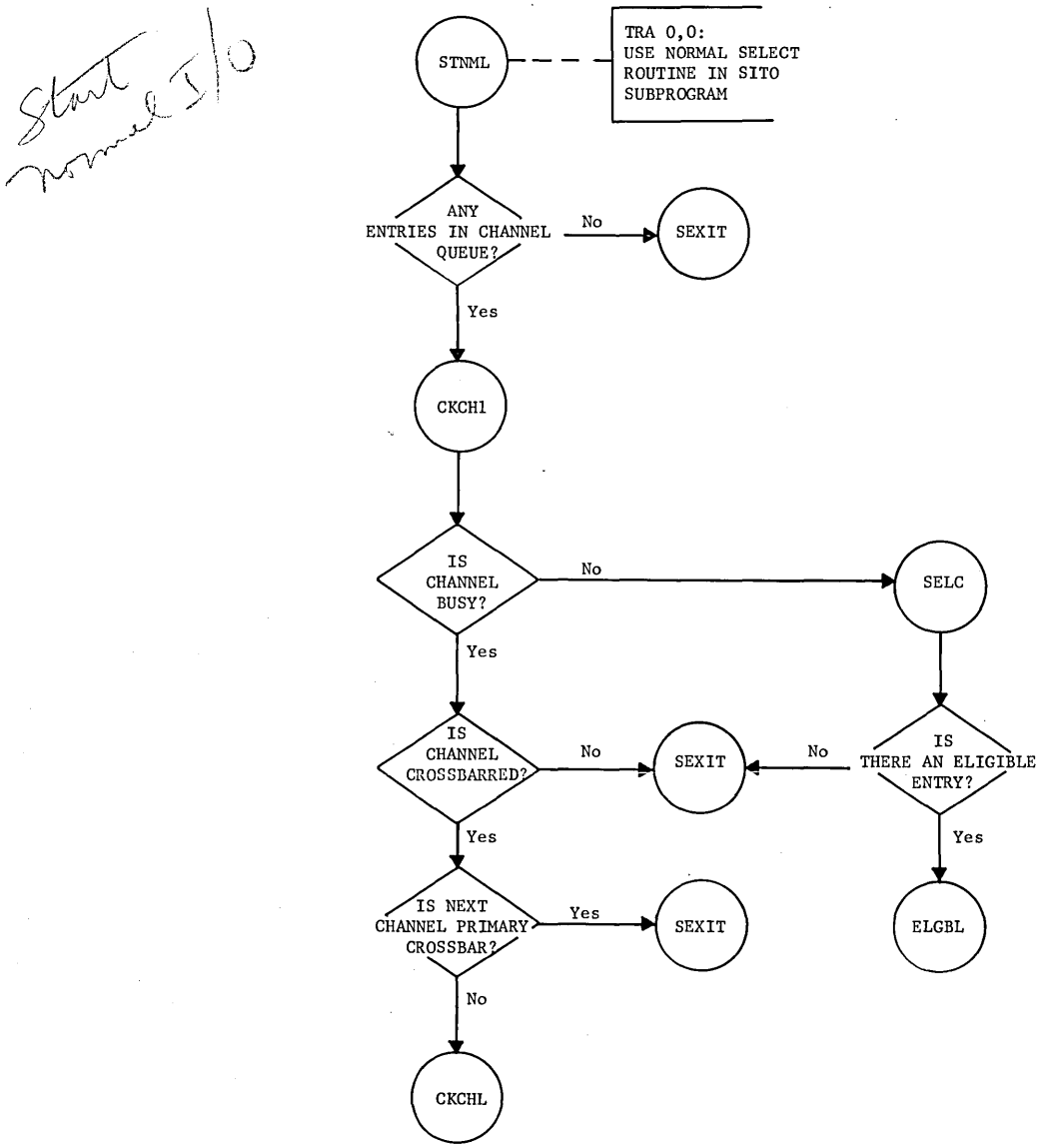
STRET
.MIOS

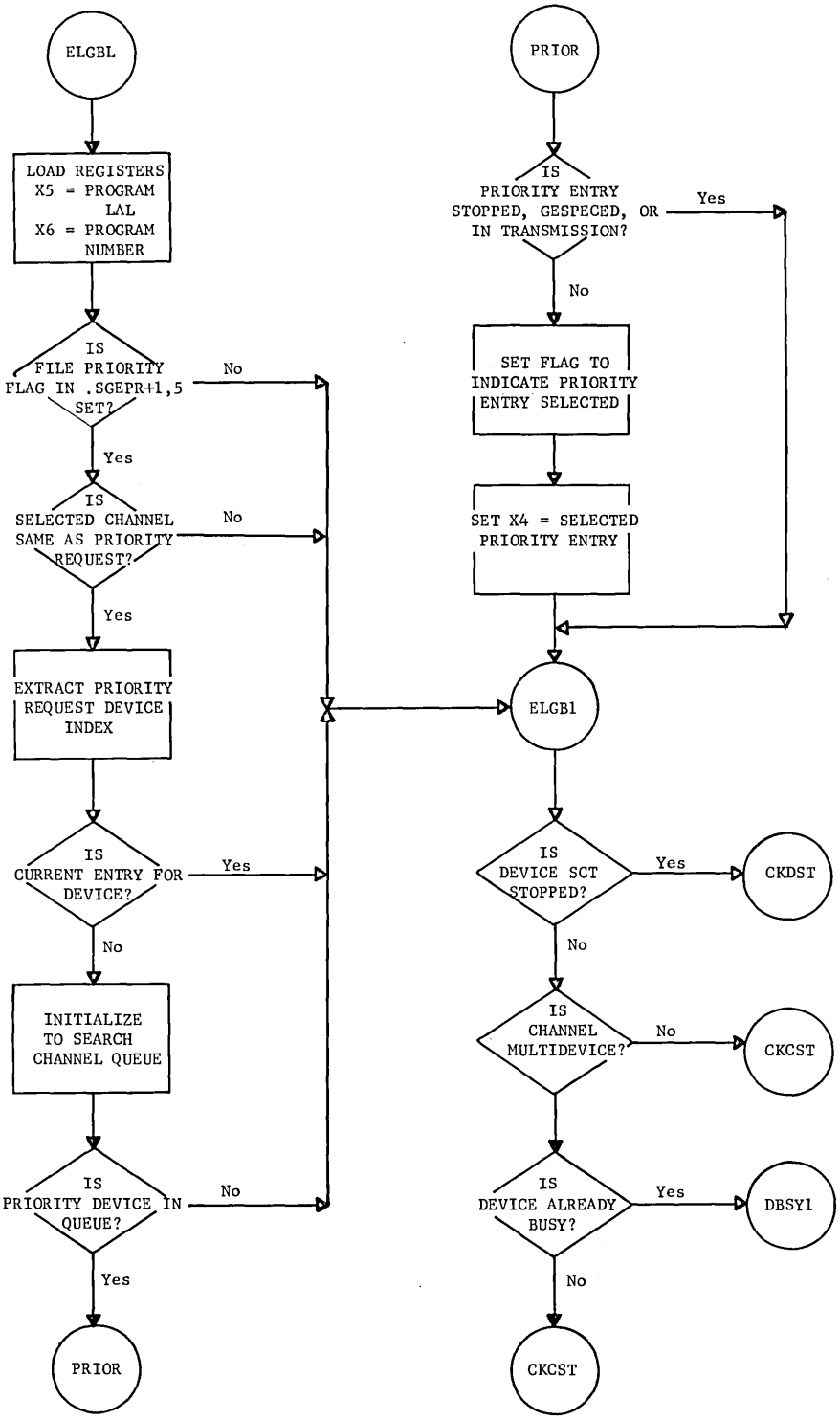

S-7
SET X5
= PROGRAM LAL
TSX1 .MDISP,14
ENTER PROGRAM
IN TAIL OF
DISPATCHER QUEUE
S-8
RESTORE
REGISTERS
.OPENS .CRDSP
GATE
TRACE LEAVING
STRET
RELOAD XO
FROM STACK
TRA 0,0

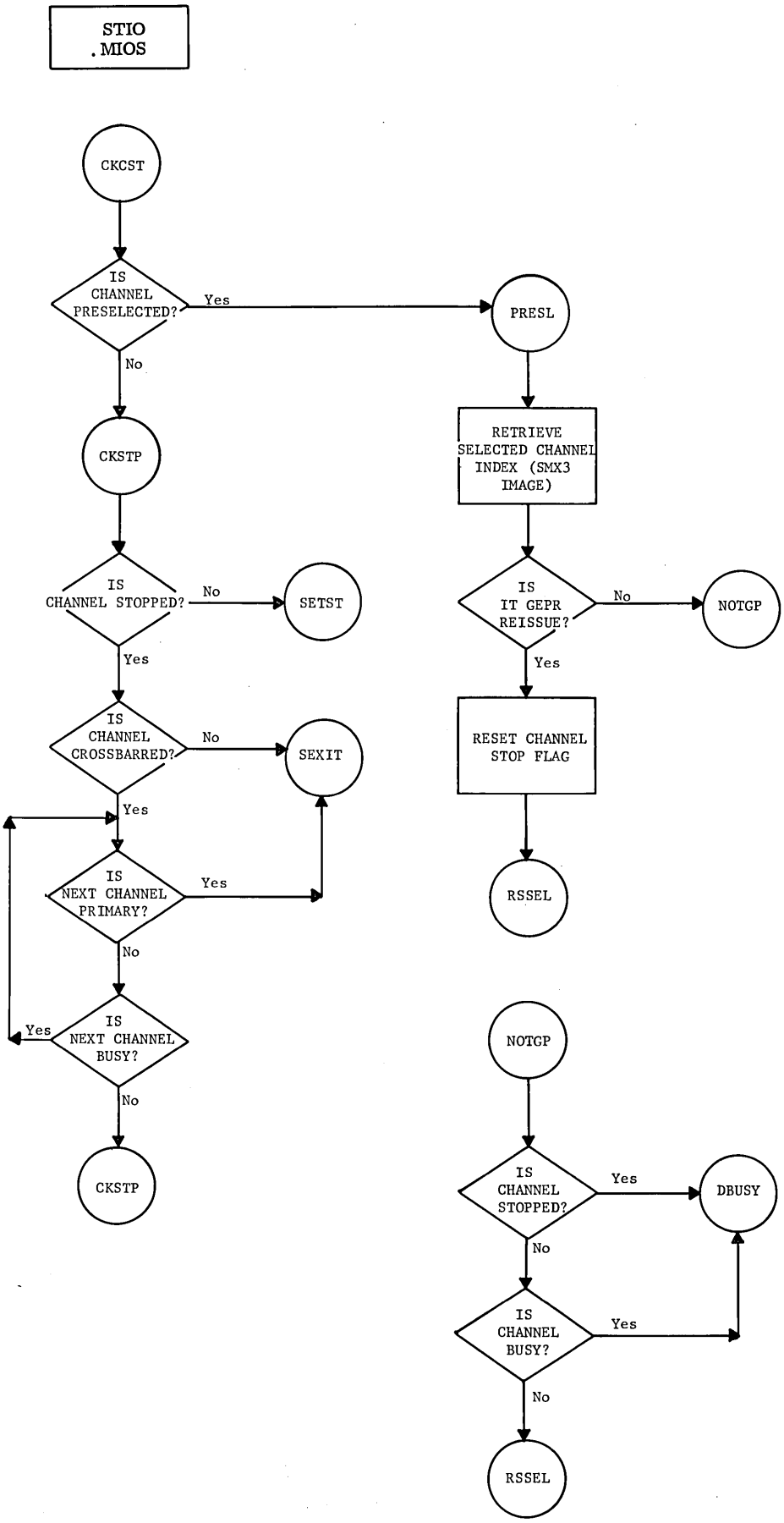
START I/O

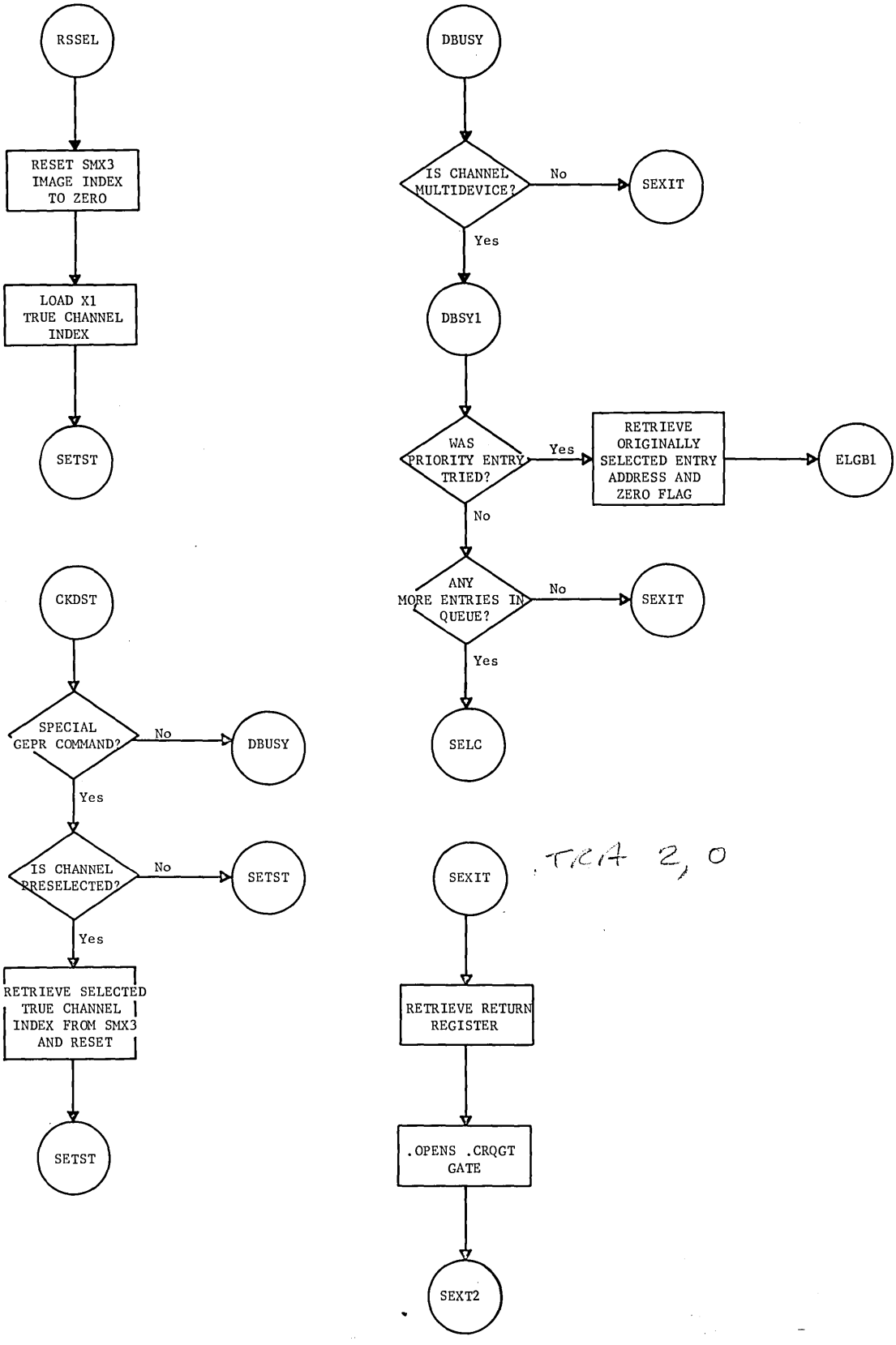

STIO
.CRQGT GATE IS SHUT
SAVE TRANSFER REGISTER
TRACE ENTRY
LOAD X1 WITH TRUE CHANNEL INDEX
TSXO (SELECT ROUTINE) EP3 IN CHANNEL MODULE
STNML
TRA 0,0
TRA 5,0
X5
1,0
SETST
2,0
SEXIT
3,0
SEEKM
4,0
WRRDM

STIO
.MIOS


STNML
TRA 0,0:
USE NORMAL SELECT
ROUTINE IN SITO
SUBPROGRAM
ANY
ENTRIES IN CHANNEL
QUEUE?
No
SEXIT
Yes
CKCH1
IS
CHANNEL
BUSY?
No
SELC
Yes
IS
CHANNEL
CROSSBARRED?
No
SEXIT
No
IS
THERE AN ELIGIBLE
ENTRY?
Yes
Yes
IS NEXT
CHANNEL PRIMARY
CROSSBAR?
Yes
SEXIT
ELGBL
No
CKCHL

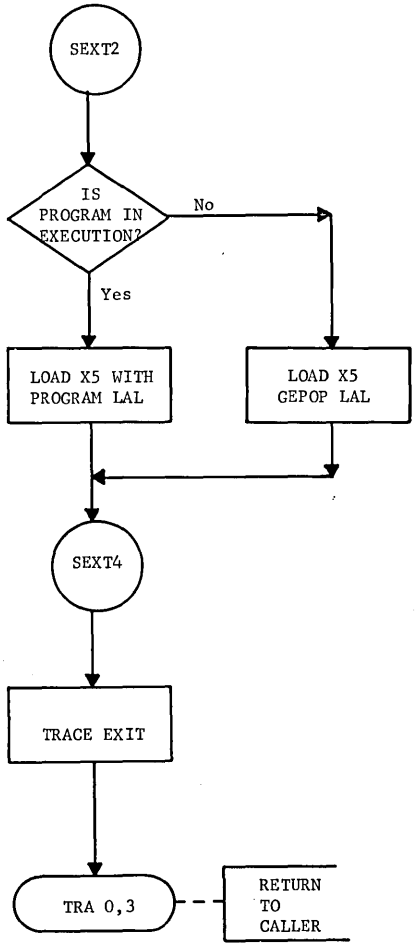

ELGBL
LOAD REGISTERS
X5 = PROGRAM LAL
X6 = PROGRAM NUMBER
IS FILE PRIORITY FLAG IN .SGEPR+1,5 SET?
No
Yes
IS SELECTED CHANNEL SAME AS PRIORITY REQUEST?
No
Yes
EXTRACT PRIORITY REQUEST DEVICE INDEX
IS CURRENT ENTRY FOR DEVICE?
Yes
No
INITIALIZE TO SEARCH CHANNEL QUEUE
IS PRIORITY DEVICE IN QUEUE?
No
Yes
PRIOR
PRIOR
IS PRIORITY ENTRY STOPPED, GESPECED, OR IN TRANSMISSION?
Yes
No
SET FLAG TO INDICATE PRIORITY ENTRY SELECTED
SET X4 = SELECTED PRIORITY ENTRY
ELGB1
IS DEVICE SCT STOPPED?
Yes
CKDST
No
IS CHANNEL MULTIDEVICE?
No
CKCST
IS DEVICE ALREADY BUSY?
Yes
DBSY1
No
CKCST

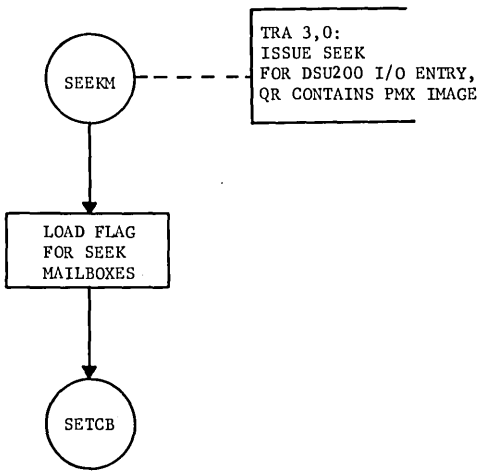

STIO
.MIOS
CKCST
IS CHANNEL PRESELECTED?
Yes
No
PRESL
CKSTP
RETRIEVE SELECTED CHANNEL INDEX (SMX3 IMAGE)
IS CHANNEL STOPPED?
No
SETST
IS IT GEPR REISSUE?
No
NOTGP
Yes
Yes
IS CHANNEL CROSSBARRED?
No
SEXIT
RESET CHANNEL STOP FLAG
Yes
IS NEXT CHANNEL PRIMARY?
Yes
RSSEL
No
Yes
IS NEXT CHANNEL BUSY?
NOTGP
No
IS CHANNEL STOPPED?
Yes
DBUSY
CKSTP
No
IS CHANNEL BUSY?
Yes
No
RSSEL

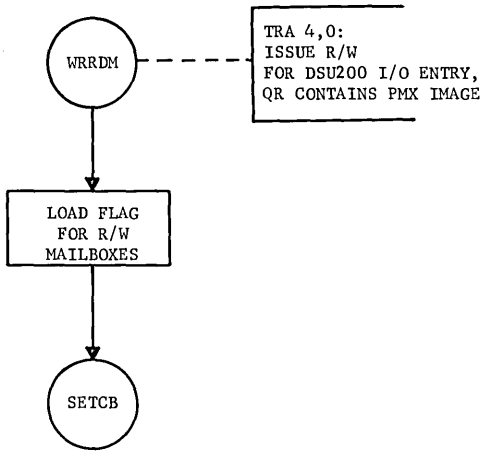

RSSEL
RESET SMX3 IMAGE INDEX TO ZERO
LOAD X1 TRUE CHANNEL INDEX
SETST
CKDST
SPECIAL GEPR COMMAND?
No
DBUSY
Yes
IS CHANNEL PRESELECTED?
No
SETST
Yes
RETRIEVE SELECTED TRUE CHANNEL INDEX FROM SMX3 AND RESET
SETST
DBUSY
IS CHANNEL MULTIDEVICE?
No
SEXIT
Yes
DBSY1
WAS PRIORITY ENTRY TRIED?
Yes
RETRIEVE ORIGINALLY SELECTED ENTRY ADDRESS AND ZERO FLAG
ELGB1
No
ANY MORE ENTRIES IN QUEUE?
No
SEXIT
Yes
SELC
SEXIT
RETRIEVE RETURN REGISTER
.OPENS .CRQGT GATE
SEXT2

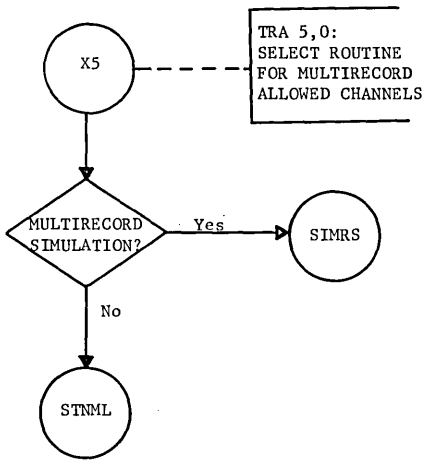
TCA 2,0

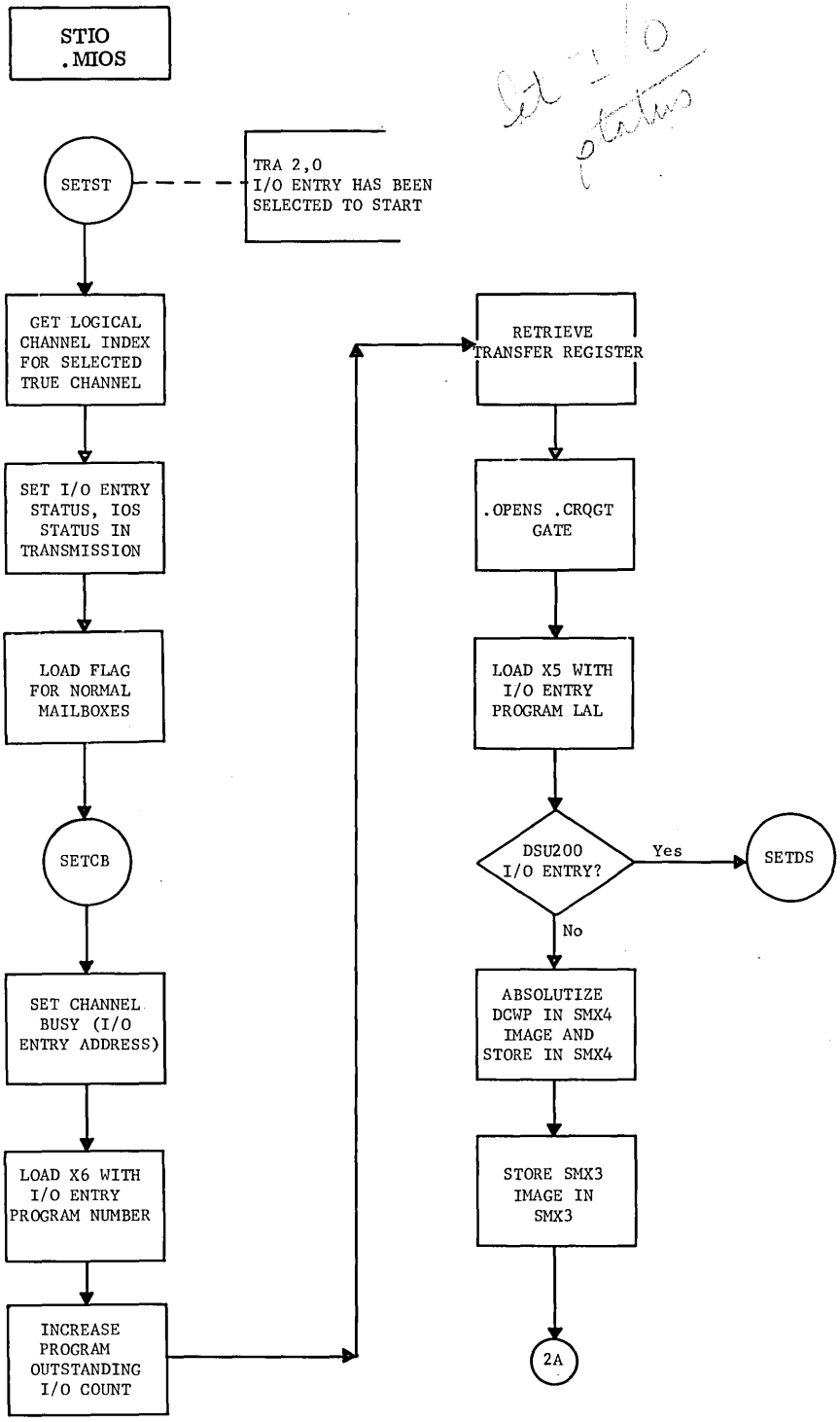
STIO
.MIOS

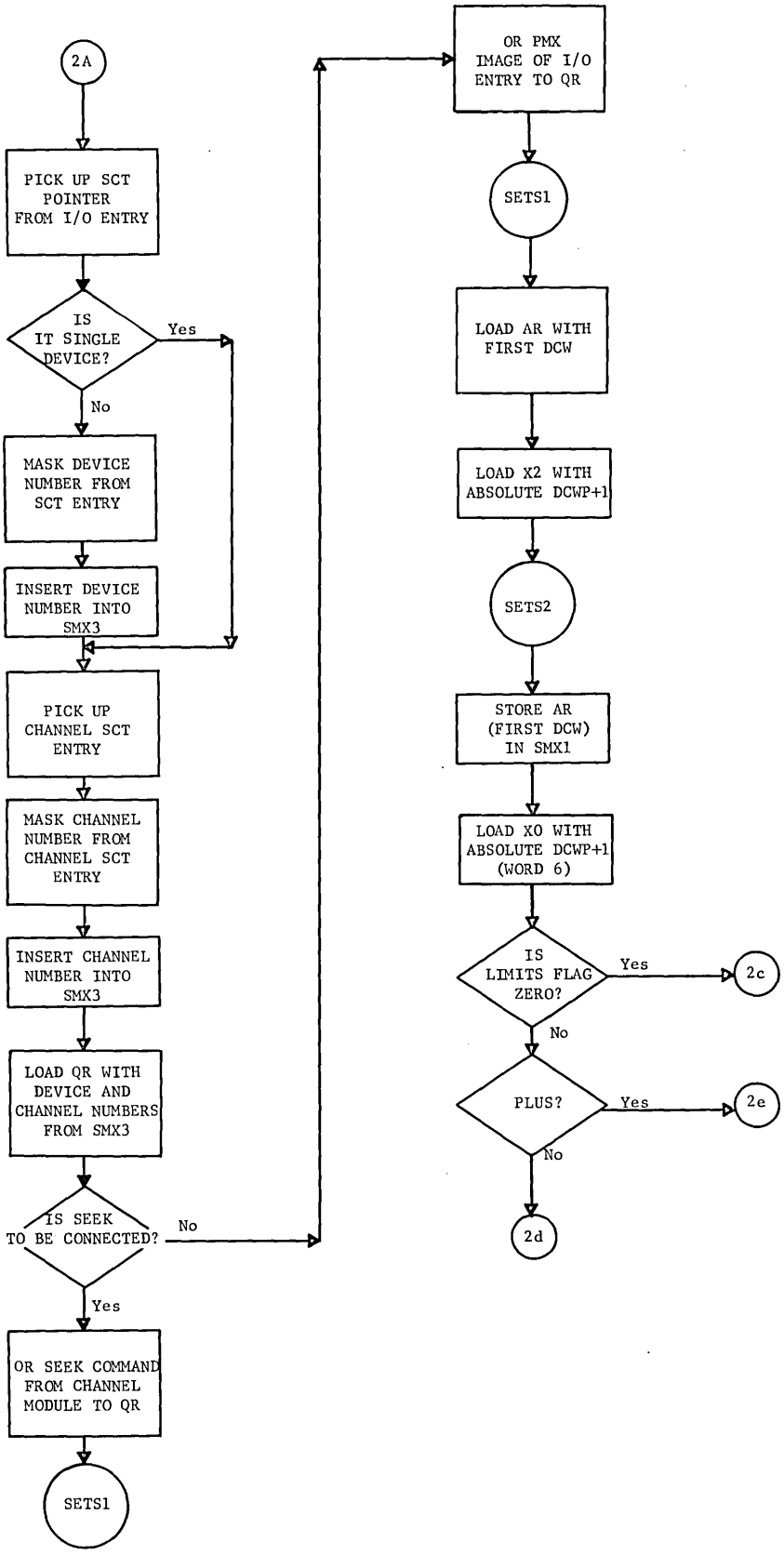

SEXT2
IS PROGRAM IN EXECUTION?
No
Yes
LOAD X5 WITH PROGRAM LAL
LOAD X5 GEPOP LAL
SEXT4
TRACE EXIT
TRA 0,3
RETURN TO CALLER

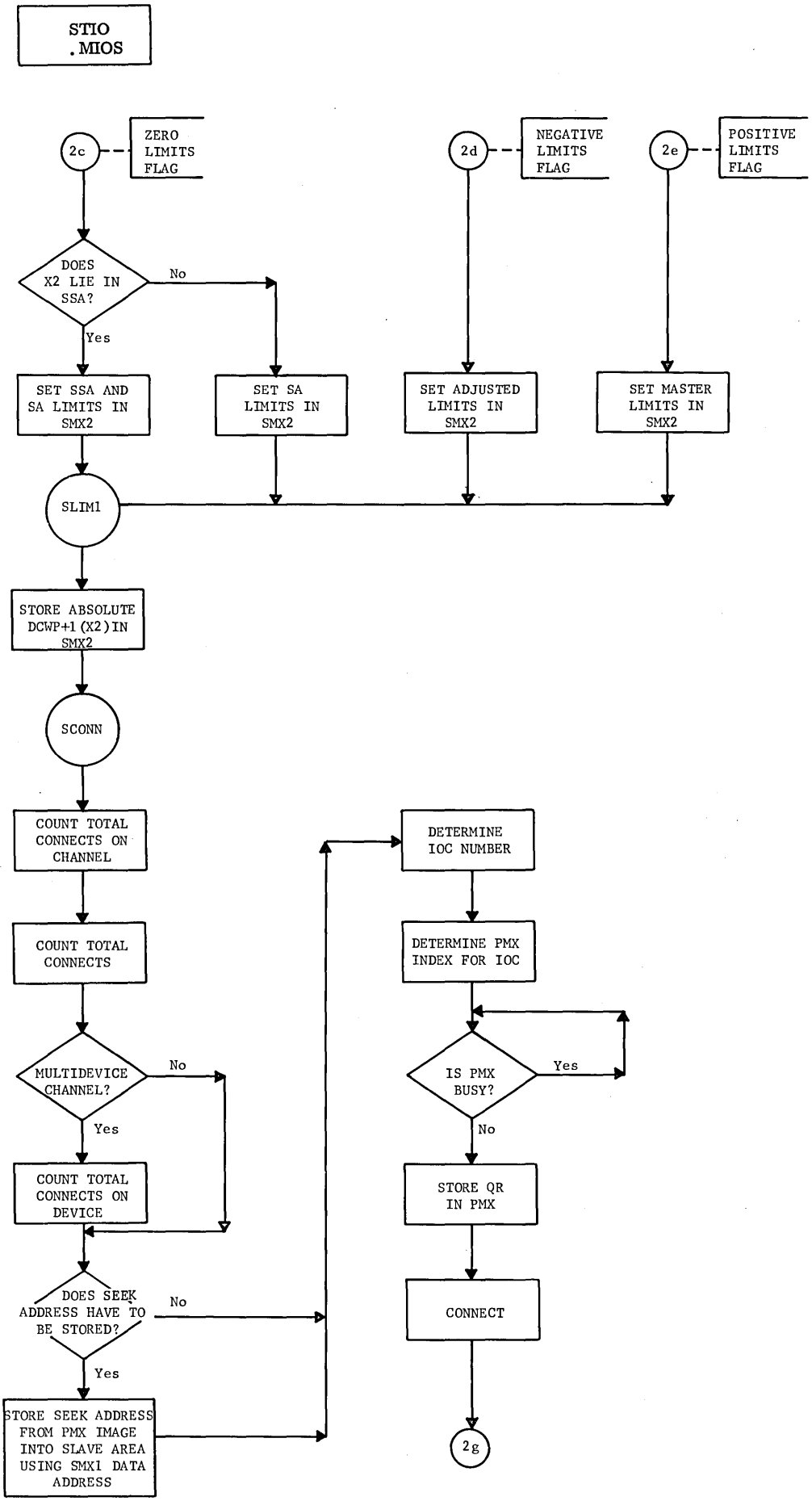

SEEKM
TRA 3,0:
ISSUE SEEK
FOR DSU200 I/O ENTRY,
QR CONTAINS PMX IMAGE
LOAD FLAG
FOR SEEK
MAILBOXES
SETCB

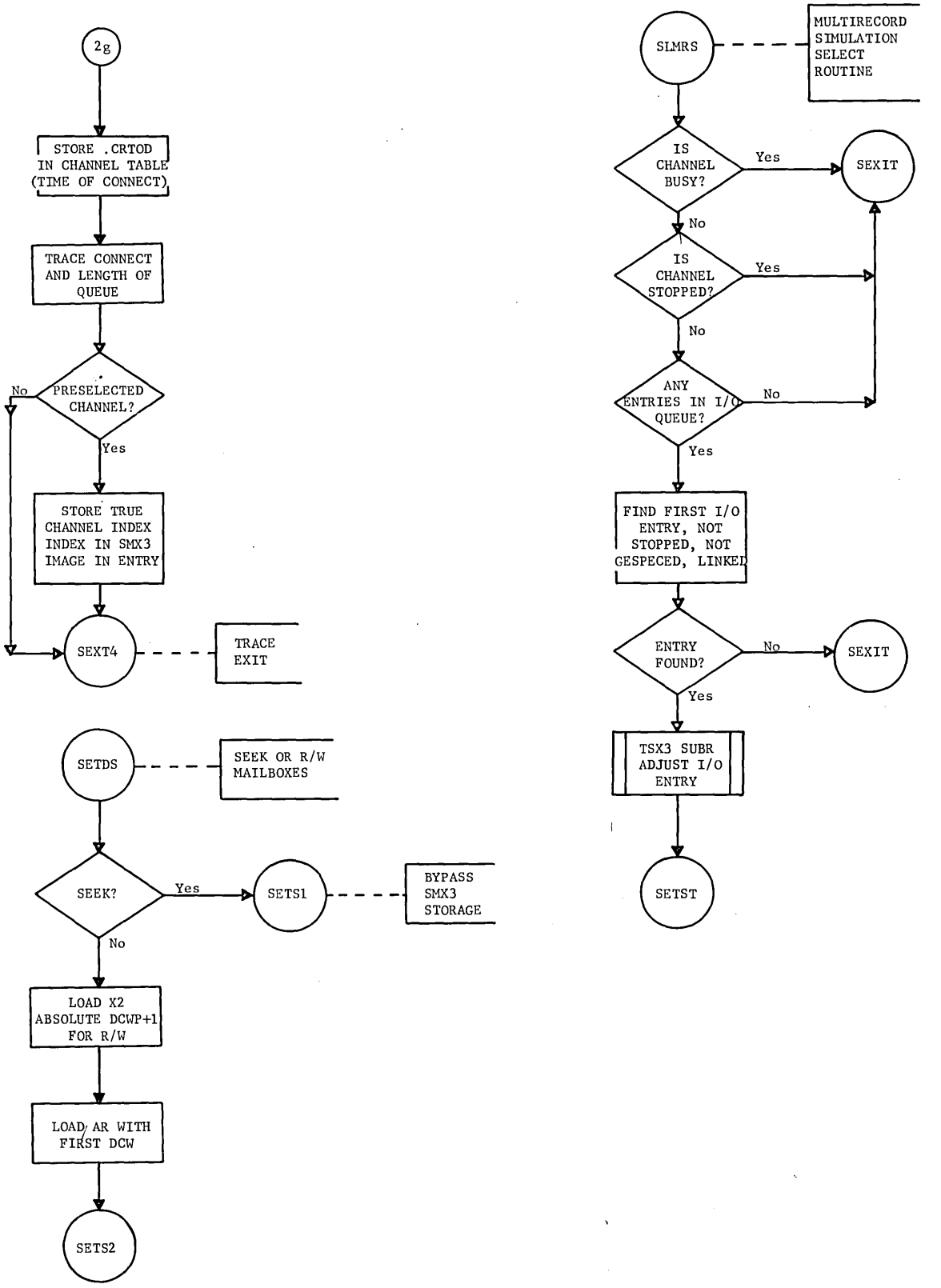

WRRDM
TRA 4,0:
ISSUE R/W
FOR DSU200 I/O ENTRY,
QR CONTAINS PMX IMAGE
LOAD FLAG
FOR R/W
MAILBOXES
SETCB


X5
TRA 5,0:
SELECT ROUTINE
FOR MULTIRECORD
ALLOWED CHANNELS
MULTIRECORD
SIMULATION?
Yes
SIMRS
No
STNML

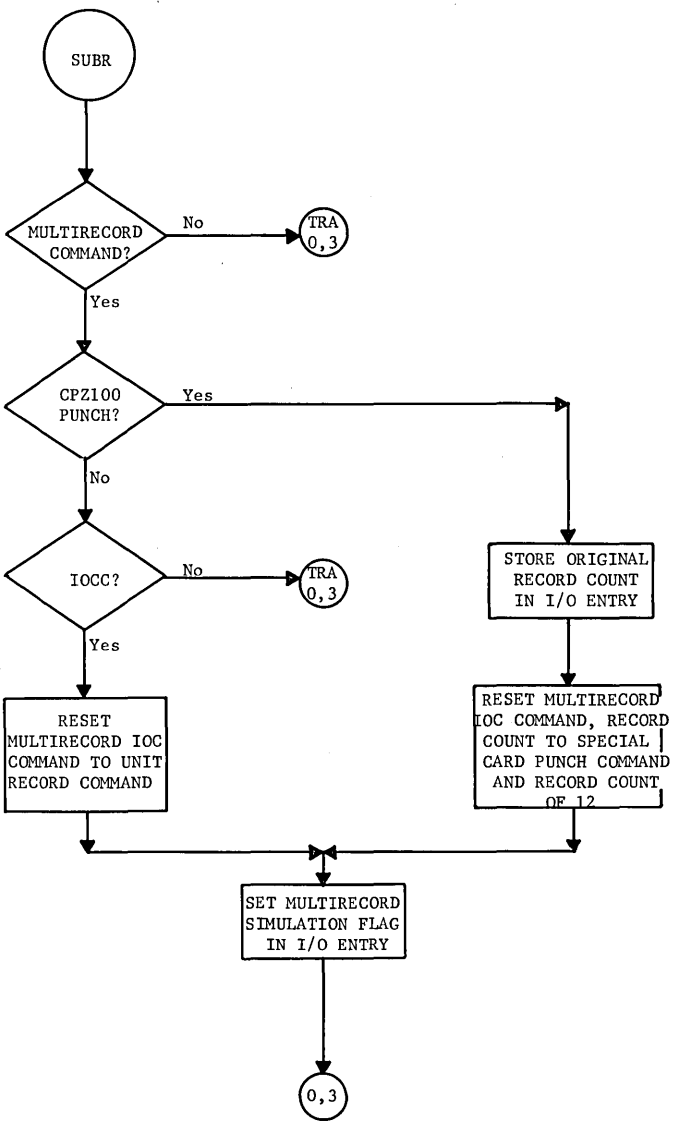

STIO
.MIOS
SETST
TRA 2,0
I/O ENTRY HAS BEEN
SELECTED TO START
GET LOGICAL
CHANNEL INDEX
FOR SELECTED
TRUE CHANNEL
SET I/O ENTRY
STATUS, IOS
STATUS IN
TRANSMISSION
LOAD FLAG
FOR NORMAL
MAILBOXES
SETCB
SET CHANNEL
BUSY (I/O
ENTRY ADDRESS)
LOAD X6 WITH
I/O ENTRY
PROGRAM NUMBER
INCREASE
PROGRAM
OUTSTANDING
I/O COUNT
RETRIEVE
TRANSFER REGISTER
.OPENS .CRQGT
GATE
LOAD X5 WITH
I/O ENTRY
PROGRAM LAL
DSU200
I/O ENTRY?
Yes
SETDS
No
ABSOLUTIZE
DCWP IN SMX4
IMAGE AND
STORE IN SMX4
STORE SMX3
IMAGE IN
SMX3
2A


2A
PICK UP SCT POINTER FROM I/O ENTRY
IS IT SINGLE DEVICE?
Yes
No
MASK DEVICE NUMBER FROM SCT ENTRY
INSERT DEVICE NUMBER INTO SMX3
PICK UP CHANNEL SCT ENTRY
MASK CHANNEL NUMBER FROM CHANNEL SCT ENTRY
INSERT CHANNEL NUMBER INTO SMX3
LOAD QR WITH DEVICE AND CHANNEL NUMBERS FROM SMX3
IS SEEK TO BE CONNECTED?
No
Yes
OR SEEK COMMAND FROM CHANNEL MODULE TO QR
SETS1
OR PMX IMAGE OF I/O ENTRY TO QR
SETS1
LOAD AR WITH FIRST DCW
LOAD X2 WITH ABSOLUTE DCWP+1
SETS2
STORE AR (FIRST DCW) IN SMX1
LOAD X0 WITH ABSOLUTE DCWP+1 (WORD 6)
IS LIMITS FLAG ZERO?
Yes
2c
No
PLUS?
Yes
2e
No
2d

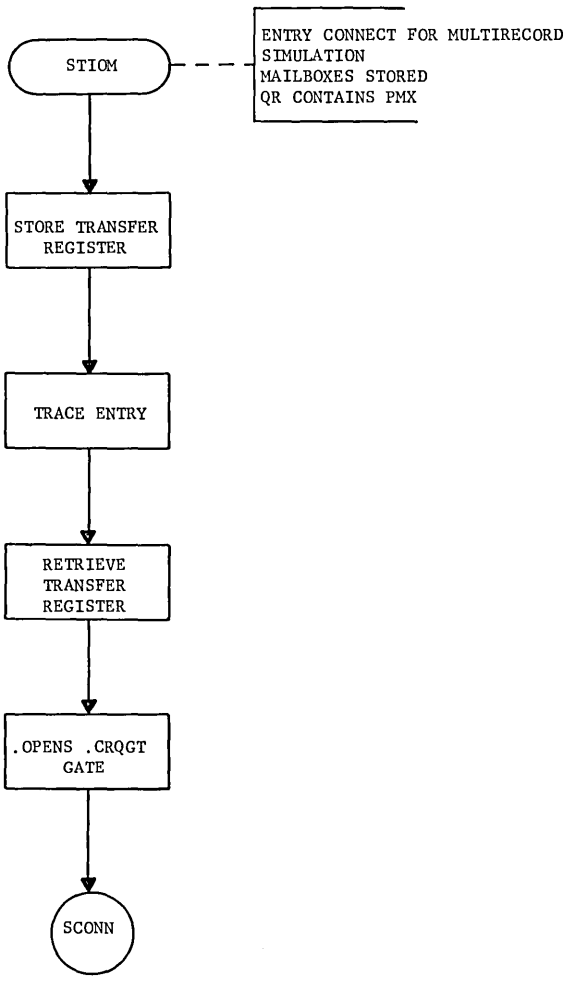

STIO
.MIOS
2c
ZERO LIMITS FLAG
2d
NEGATIVE LIMITS FLAG
2e
POSITIVE LIMITS FLAG
DOES X2 LIE IN SSA?
No
Yes
SET SSA AND SA LIMITS IN SMX2
SET SA LIMITS IN SMX2
SET ADJUSTED LIMITS IN SMX2
SET MASTER LIMITS IN SMX2
SLIM1
STORE ABSOLUTE DCWP+1 (X2) IN SMX2
SCONN
COUNT TOTAL CONNECTS ON CHANNEL
COUNT TOTAL CONNECTS
MULTIDEVICE CHANNEL?
No
Yes
COUNT TOTAL CONNECTS ON DEVICE
DOES SEEK ADDRESS HAVE TO BE STORED?
No
Yes
STORE SEEK ADDRESS FROM PMX IMAGE INTO SLAVE AREA USING SMX1 DATA ADDRESS
DETERMINE IOC NUMBER
DETERMINE PMX INDEX FOR IOC
IS PMX BUSY?
Yes
No
STORE QR IN PMX
CONNECT
2g


2g
STORE .CRTOD IN CHANNEL TABLE (TIME OF CONNECT)
TRACE CONNECT AND LENGTH OF QUEUE
PRESELECTED CHANNEL?
No
Yes
STORE TRUE CHANNEL INDEX INDEX IN SMX3 IMAGE IN ENTRY
SEXT4
TRACE EXIT
SETDS
SEEK OR R/W MAILBOXES
SEEK?
Yes
SETS1
BYPASS SMX3 STORAGE
No
LOAD X2 ABSOLUTE DCWP+1 FOR R/W
LOAD AR WITH FIRST DCW
SETS2
SLMRS
MULTIRECORD SIMULATION SELECT ROUTINE
IS CHANNEL BUSY?
Yes
SEXIT
No
IS CHANNEL STOPPED?
Yes
No
ANY ENTRIES IN I/O QUEUE?
No
Yes
FIND FIRST I/O ENTRY, NOT STOPPED, NOT GESPECED, LINKED
ENTRY FOUND?
No
SEXIT
Yes
TSX3 SUBR ADJUST I/O ENTRY
SETST

STIO
.MIOS

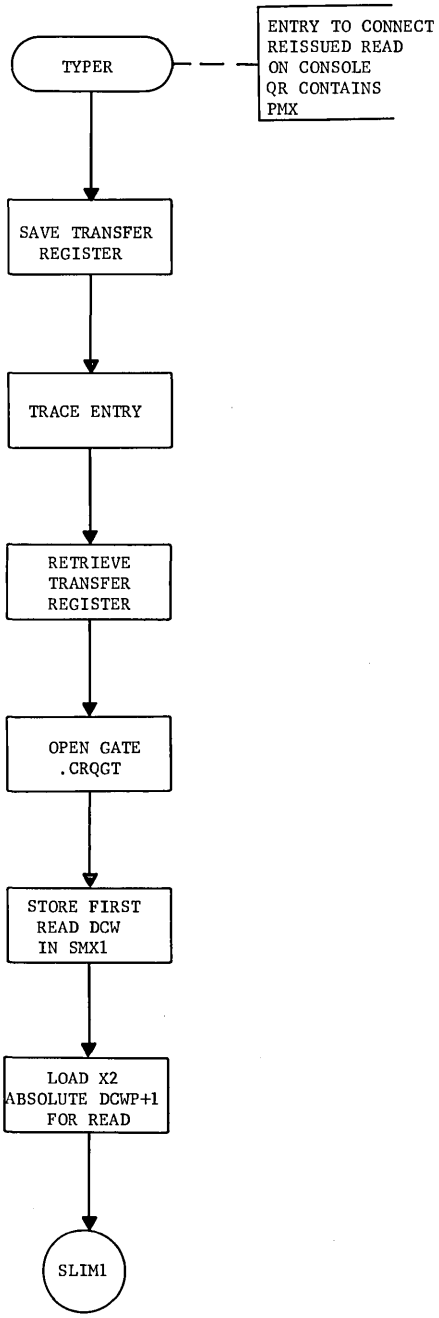

SUBR
MULTIRECORD COMMAND?
No
TRA 0,3
Yes
CPZ100 PUNCH?
Yes
No
IOCC?
No
TRA 0,3
Yes
STORE ORIGINAL RECORD COUNT IN I/O ENTRY
RESET MULTIRECORD IOC COMMAND TO UNIT RECORD COMMAND
RESET MULTIRECORD IOC COMMAND, RECORD COUNT TO SPECIAL CARD PUNCH COMMAND AND RECORD COUNT OF 12
SET MULTIRECORD SIMULATION FLAG IN I/O ENTRY
0,3

CONNECT MULTIRECORD SIMULATION DCW

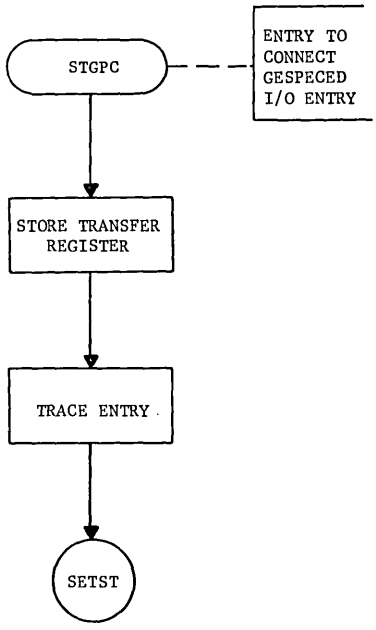


TYPER
.MIOS

CONNECT REISSUE OF SECOND TYPEWRITER COMMAND

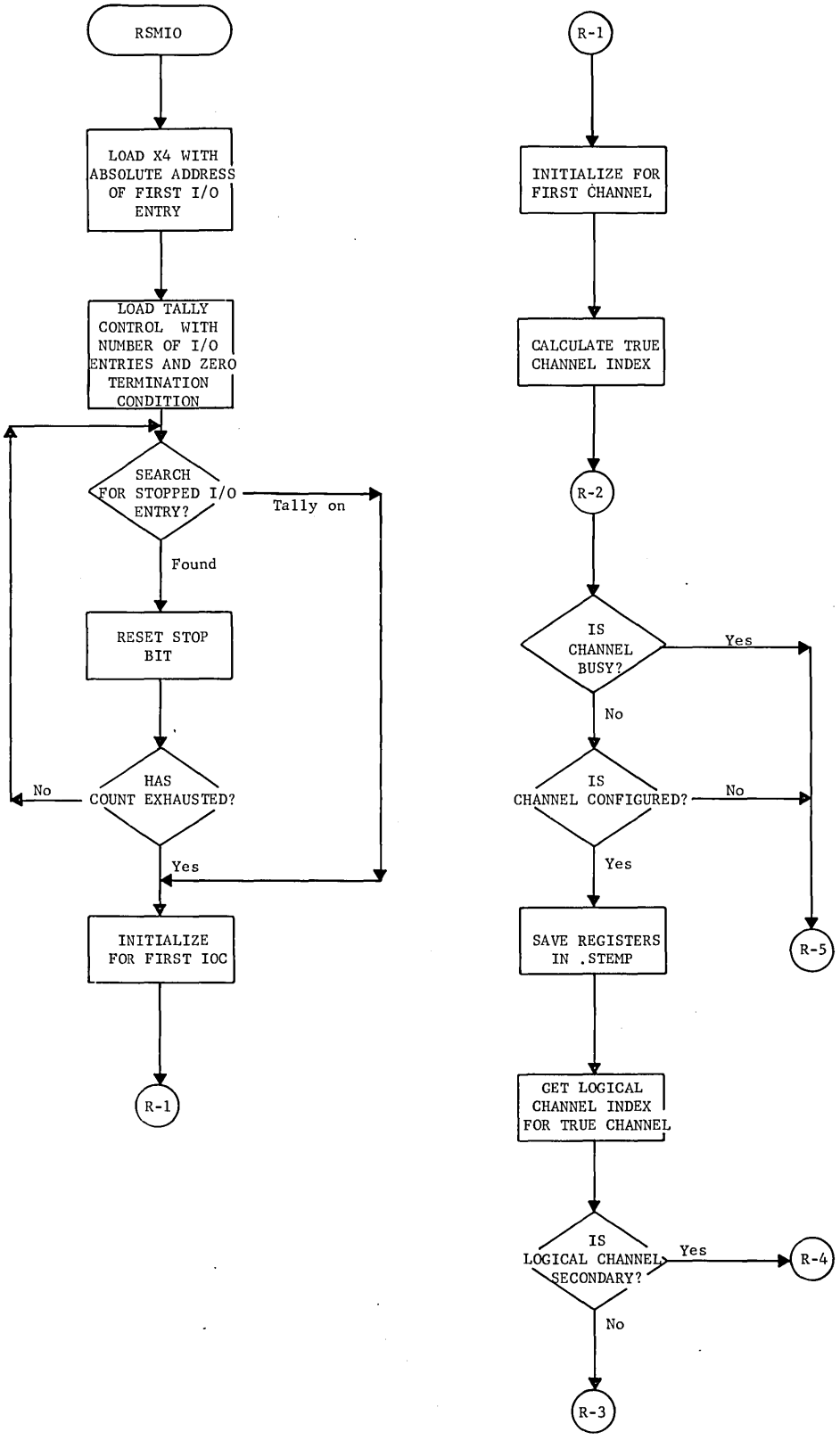


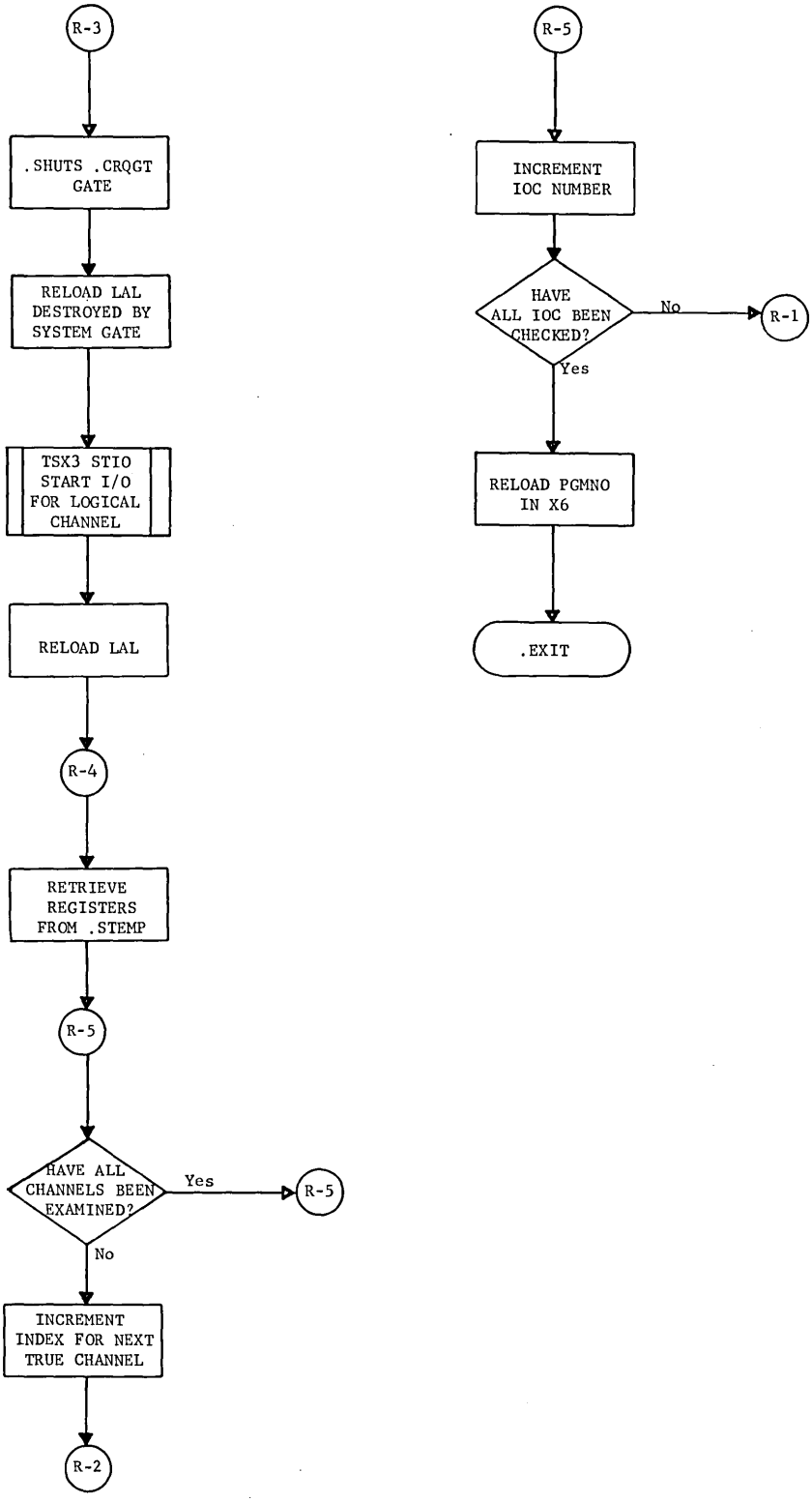
CONNECT SELECTED GESPECED ENTRY



RSMIO (EP8)
.MIOS

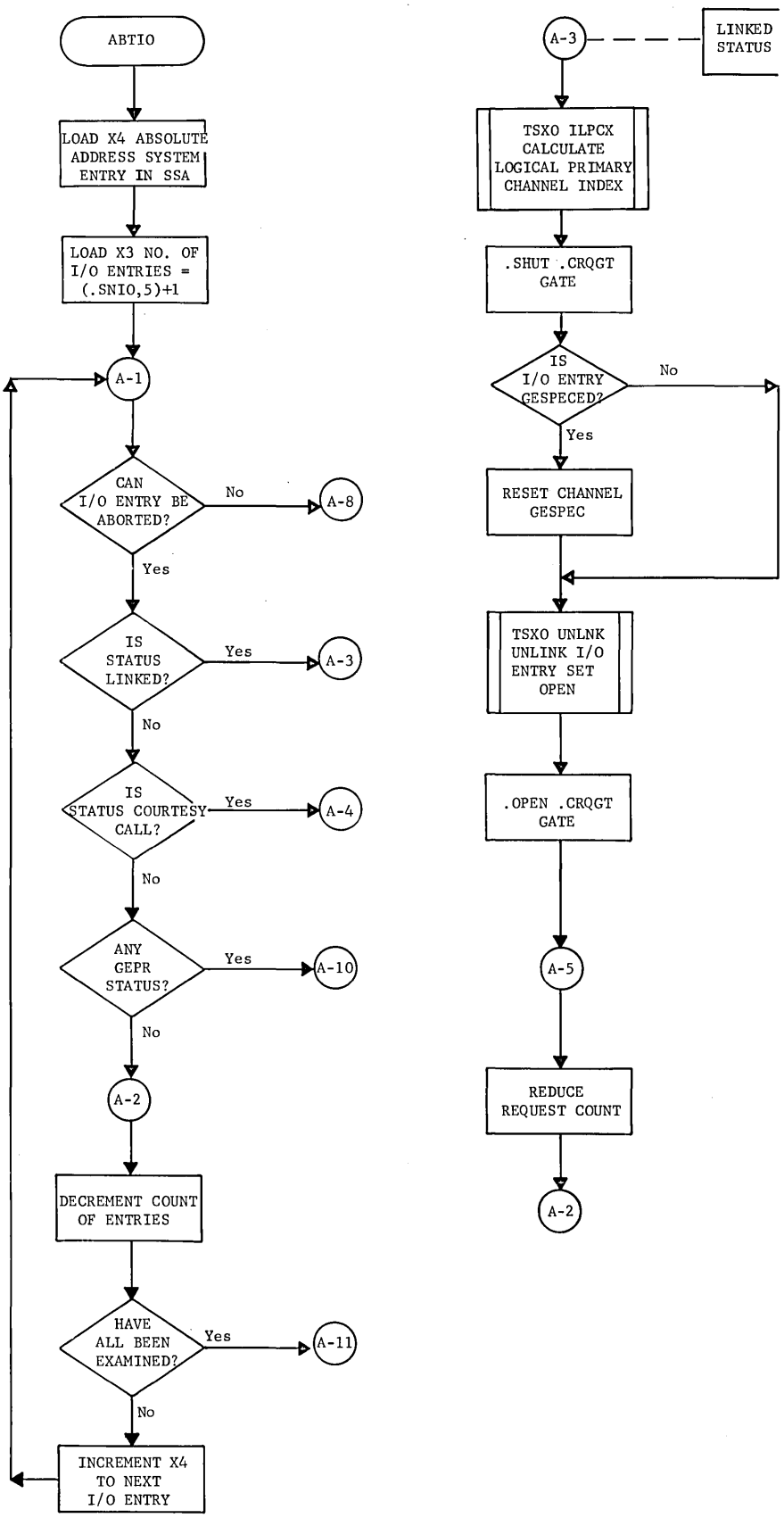
RESUME I/O FOR PROGRAM

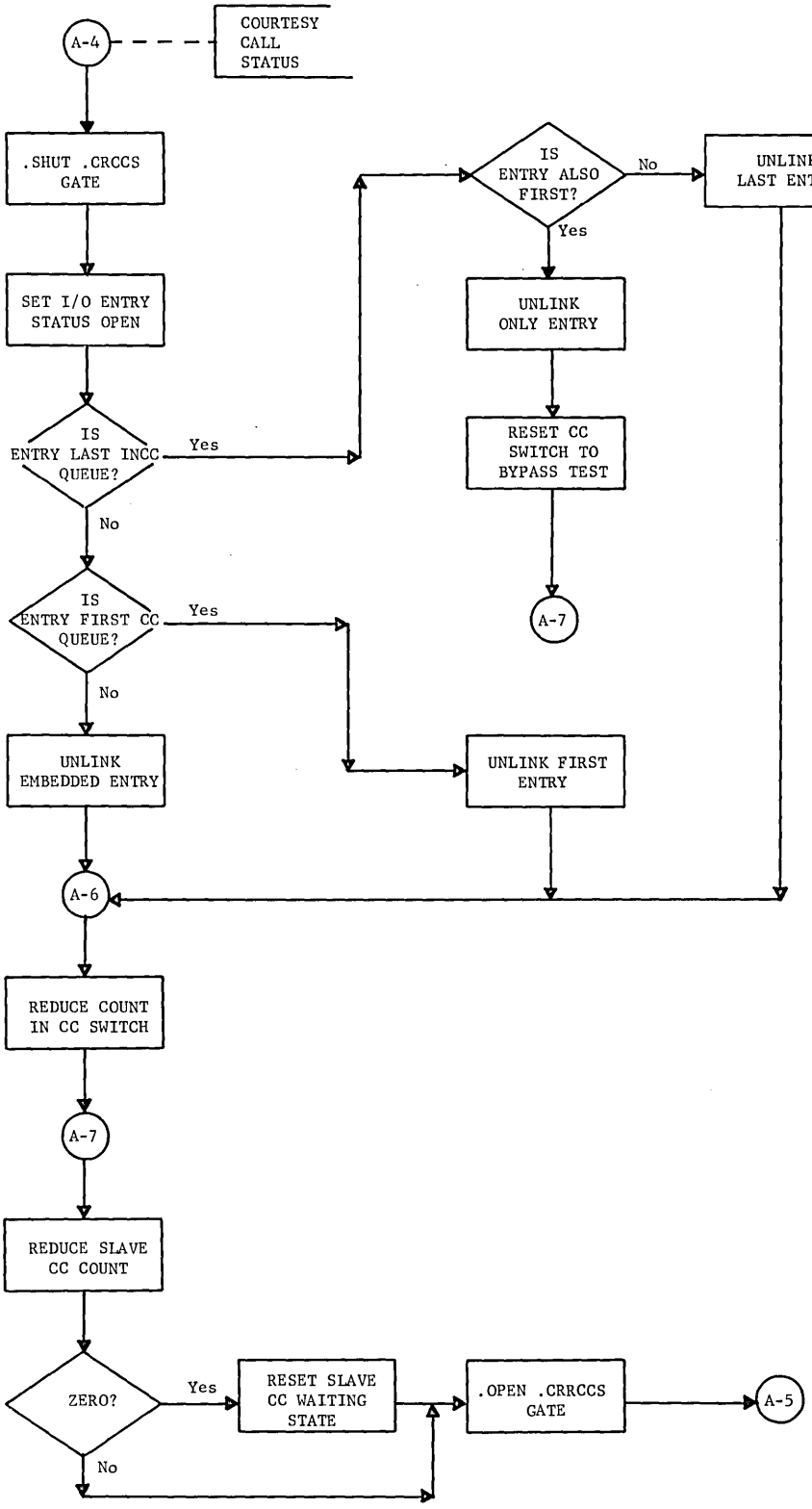


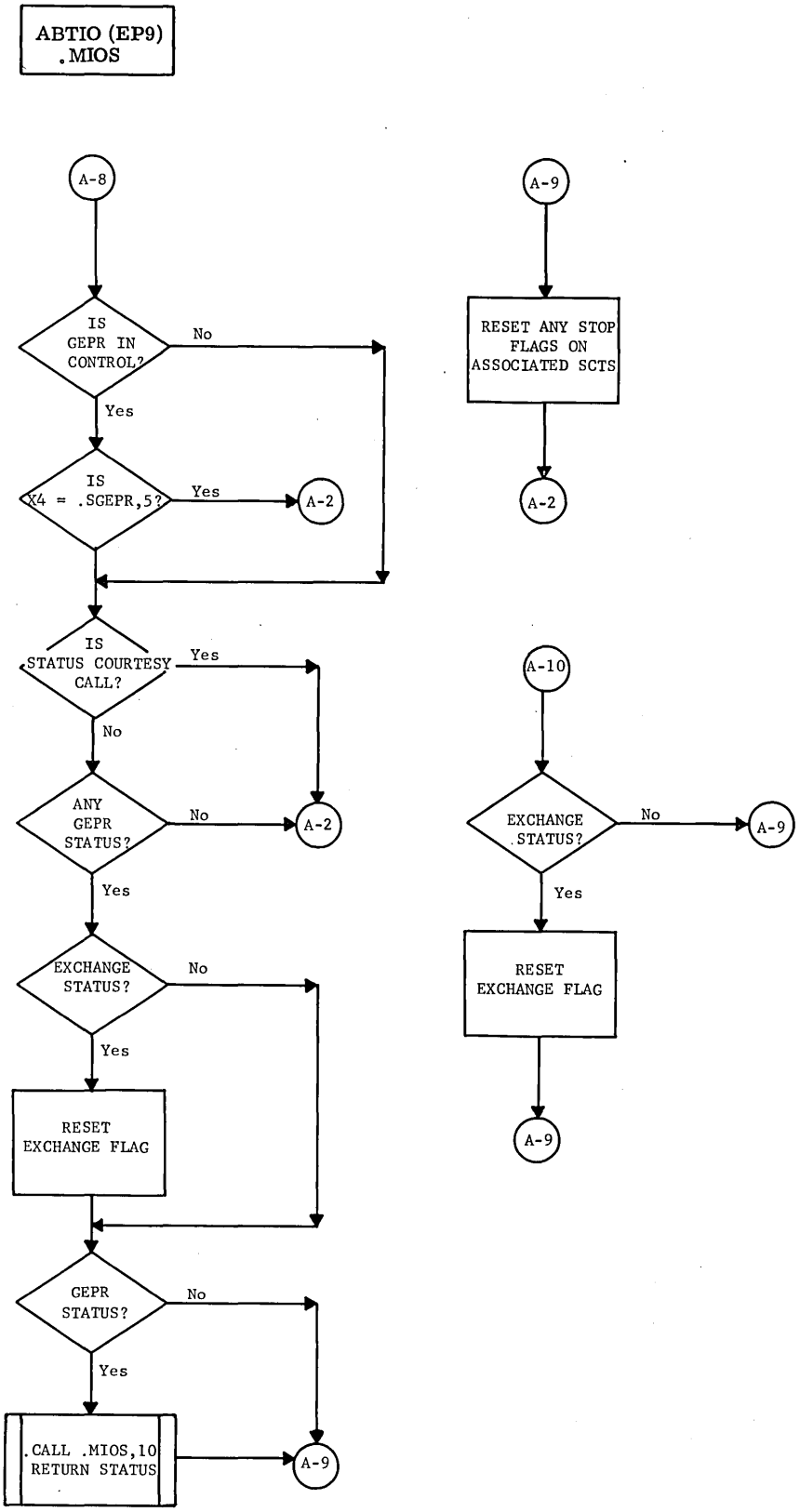

R-3
.SHUTS .CRQGT GATE
RELOAD LAL DESTROYED BY SYSTEM GATE
TSX3 STIO START I/O FOR LOGICAL CHANNEL
RELOAD LAL
R-4
RETRIEVE REGISTERS FROM .STEMP
R-5
HAVE ALL CHANNELS BEEN EXAMINED?
Yes
R-5
No
INCREMENT INDEX FOR NEXT TRUE CHANNEL
R-2
R-5
INCREMENT IOC NUMBER
HAVE ALL IOC BEEN CHECKED?
No
R-1
Yes
RELOAD PGMNO IN X6
.EXIT

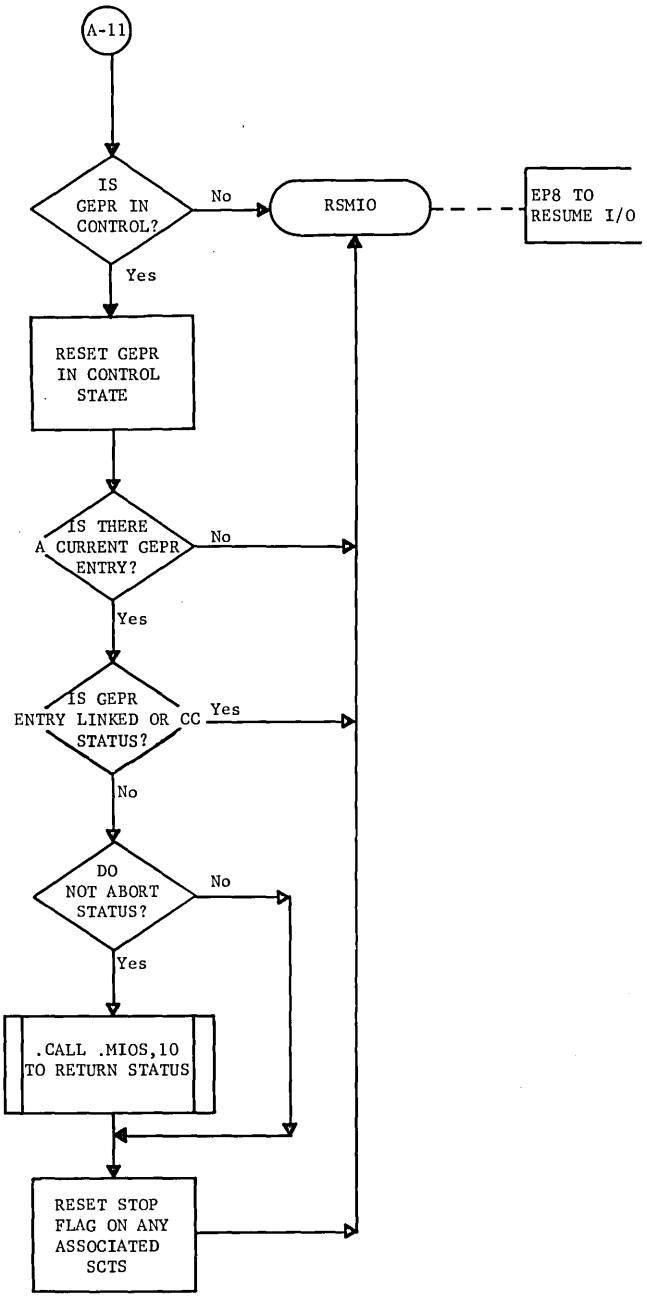
ABTIO (EP9)
.MIOS

ABORT I/O FOR PROGRAM



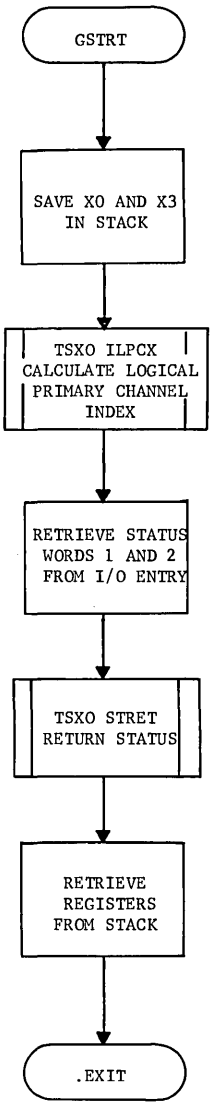

A-4
COURTESY CALL STATUS
.SHUT .CRCCS GATE
SET I/O ENTRY STATUS OPEN
IS ENTRY LAST INCC QUEUE?
Yes
No
IS ENTRY FIRST CC QUEUE?
Yes
No
UNLINK EMBEDDED ENTRY
IS ENTRY ALSO FIRST?
No
Yes
UNLINK LAST ENTRY
UNLINK ONLY ENTRY
RESET CC SWITCH TO BYPASS TEST
A-7
UNLINK FIRST ENTRY
A-6
REDUCE COUNT IN CC SWITCH
A-7
REDUCE SLAVE CC COUNT
ZERO?
Yes
No
RESET SLAVE CC WAITING STATE
.OPEN .CRRCCS GATE
A-5


ABTIO (EP9)
.MIOS
A-8
IS
GEPR IN
CONTROL?
No
Yes
IS
X4 = .SGEPR,5?
Yes
A-2
IS
STATUS COURTESY
CALL?
Yes
No
ANY
GEPR
STATUS?
No
A-2
Yes
EXCHANGE
STATUS?
No
Yes
RESET
EXCHANGE FLAG
GEPR
STATUS?
No
Yes
.CALL .MIOS,10
RETURN STATUS
A-9
A-9
RESET ANY STOP
FLAGS ON
ASSOCIATED SCTS
A-2
A-10
EXCHANGE
STATUS?
No
A-9
Yes
RESET
EXCHANGE FLAG
A-9

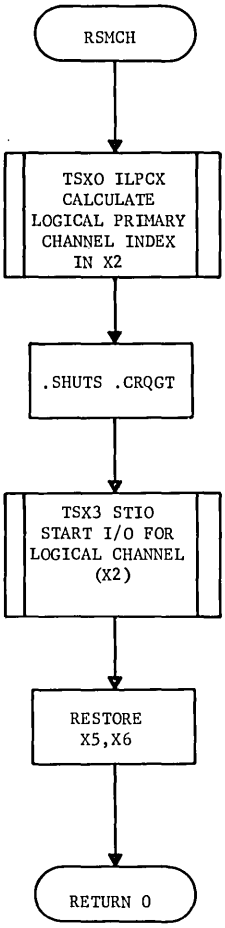

A-11
IS GEPR IN CONTROL?
No
RSMIO
EP8 TO RESUME I/O
Yes
RESET GEPR IN CONTROL STATE
IS THERE A CURRENT GEPR ENTRY?
No
Yes
IS GEPR ENTRY LINKED OR CC STATUS?
Yes
No
DO NOT ABORT STATUS?
No
Yes
.CALL .MIOS,10 TO RETURN STATUS
RESET STOP FLAG ON ANY ASSOCIATED SCTS

GSTRT (EP10)
.MIOS

FORMAT I/O STATUS WORDS AND RETURN STATUS

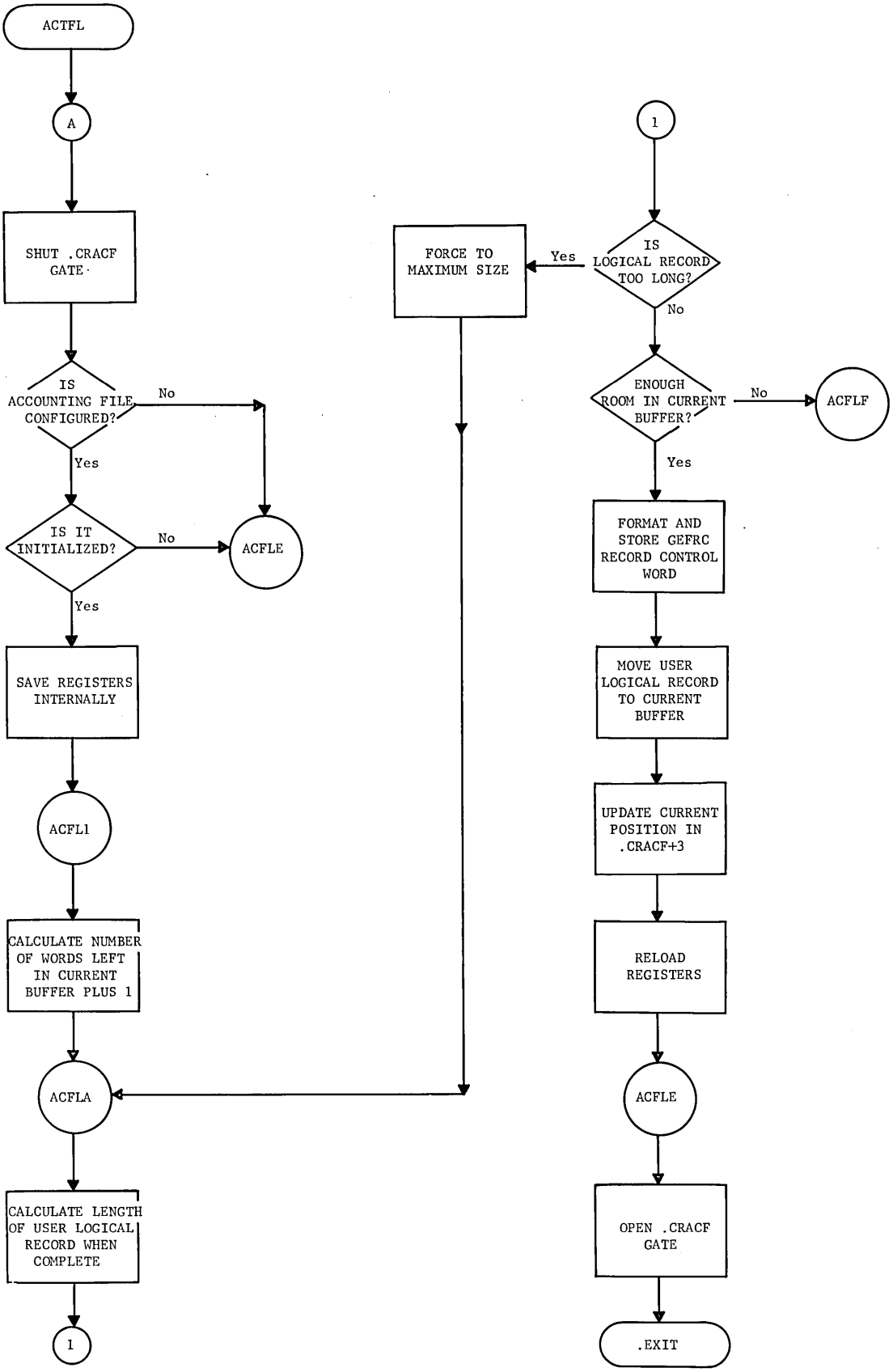


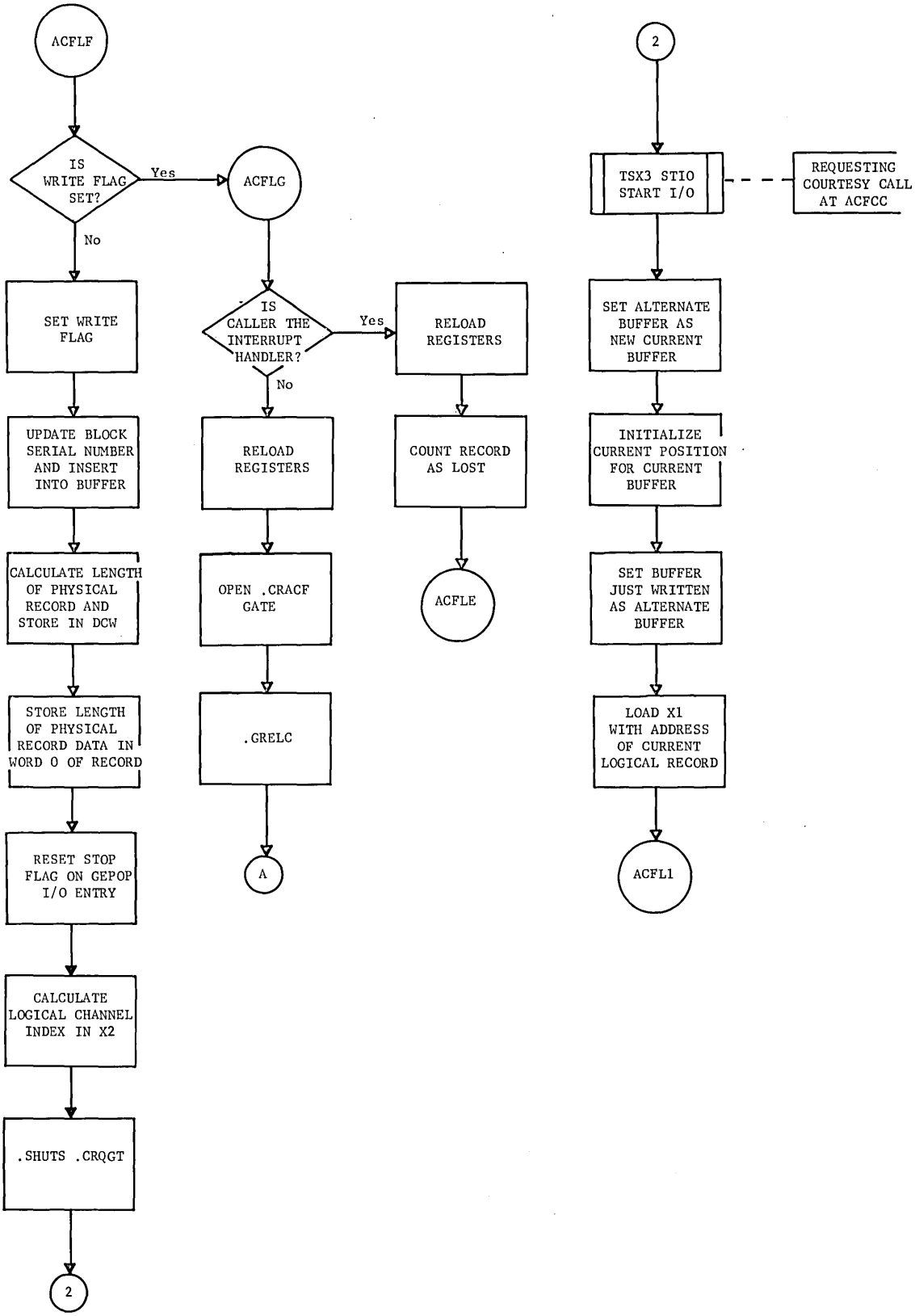
RESUME I/O ON CHANNEL

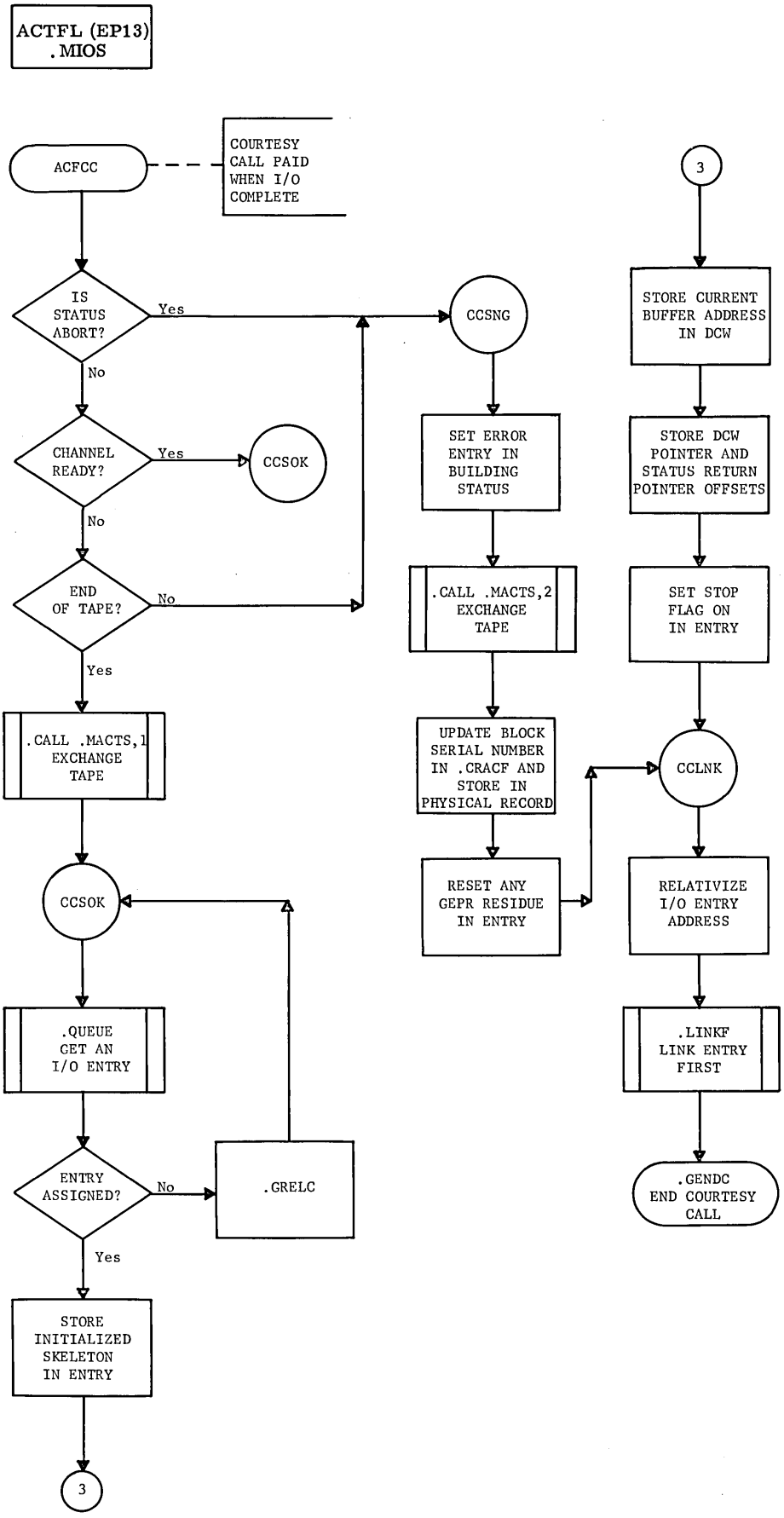


ACTFL (EP13)
.MIOS

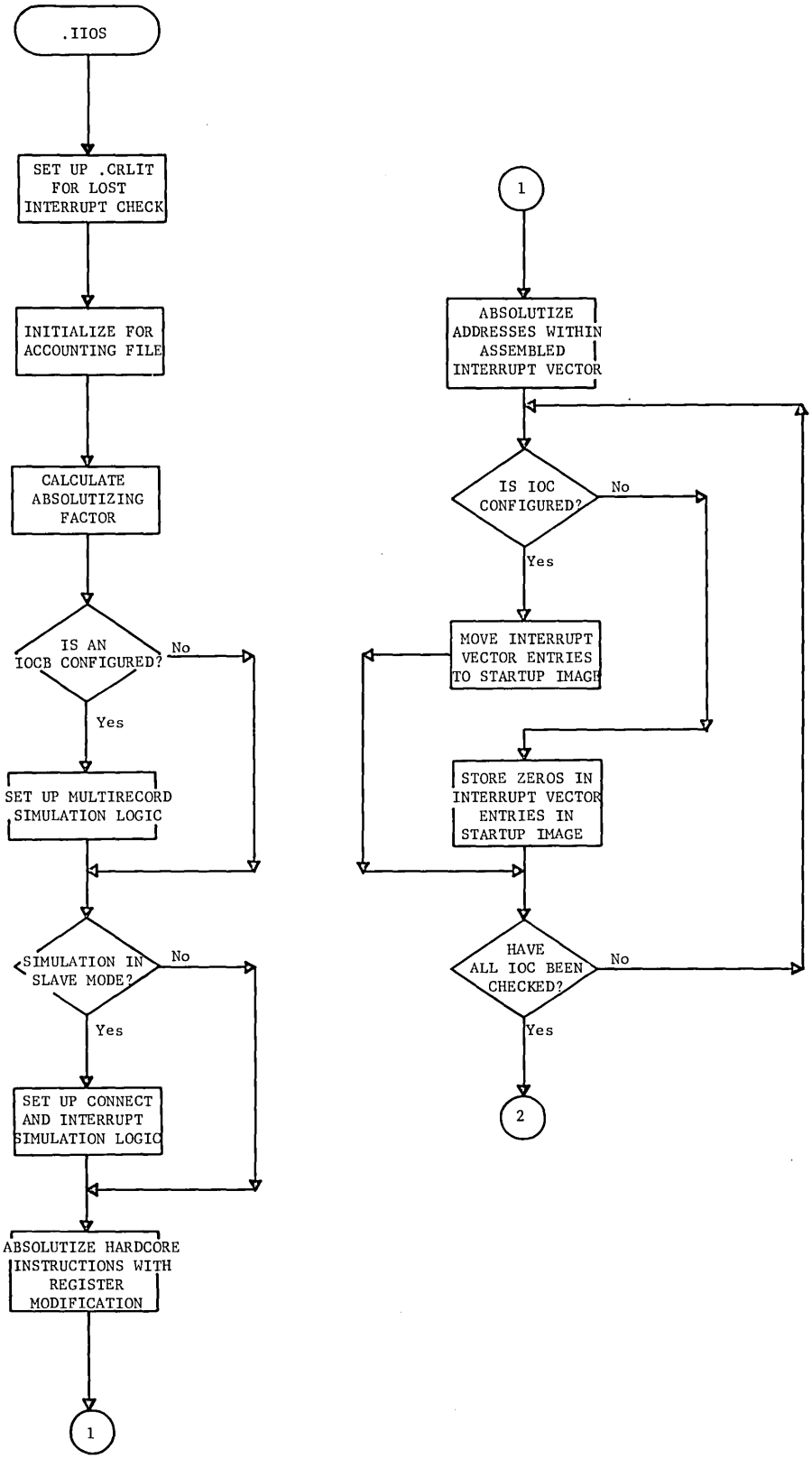
ACCOUNTING FILE REQUEST

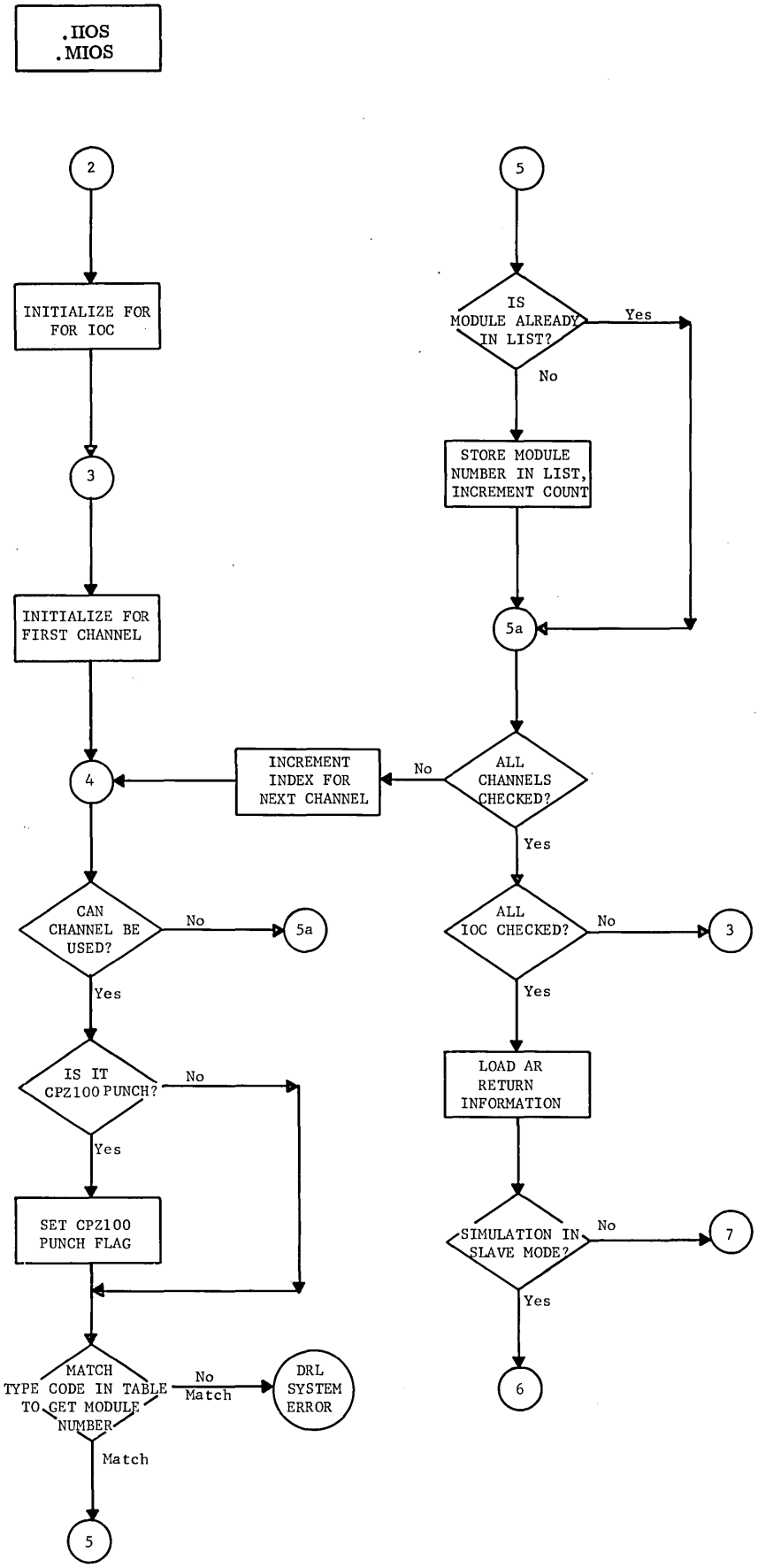


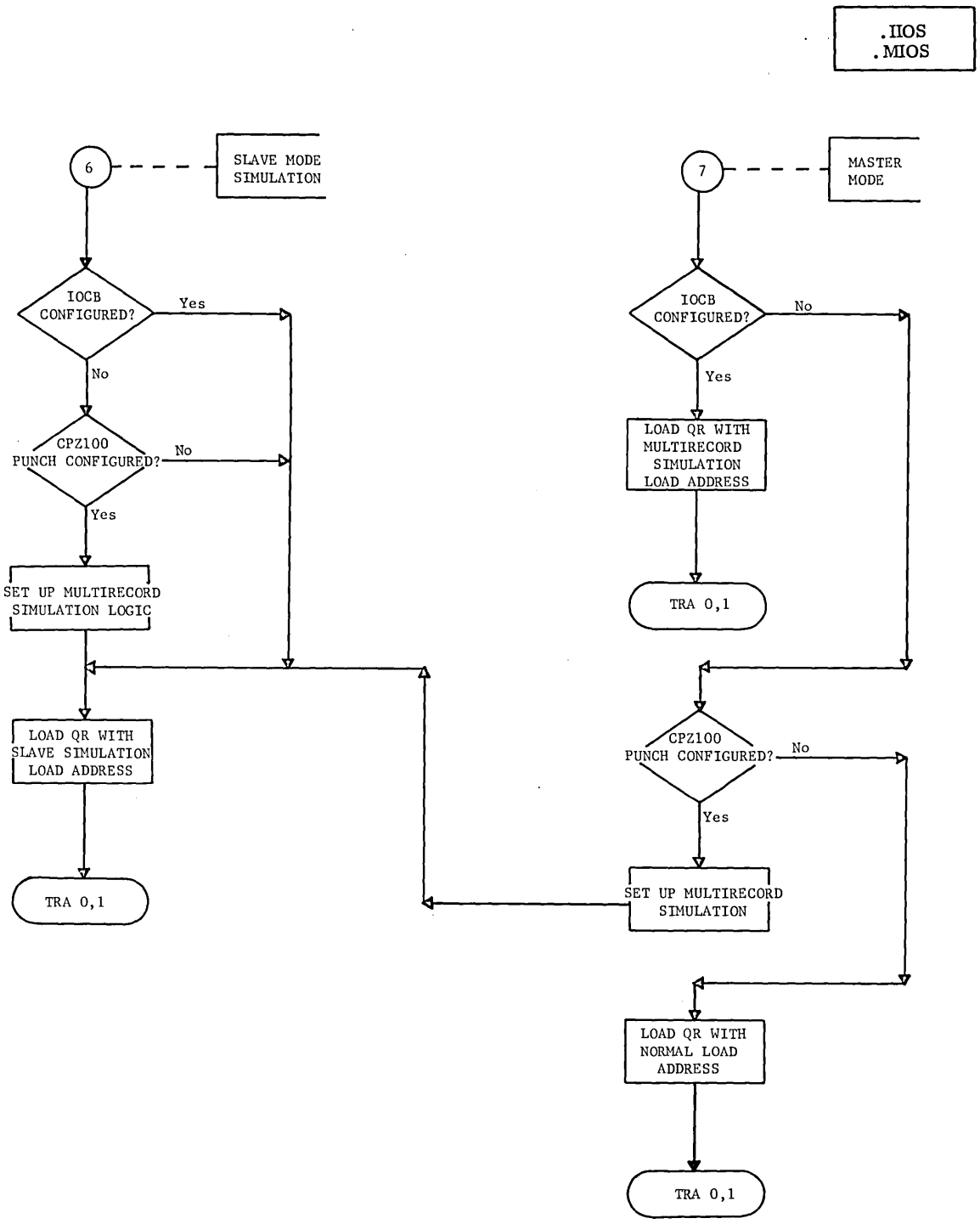

ACFLF
IS WRITE FLAG SET?
Yes
No
ACFLG
SET WRITE FLAG
IS CALLER THE INTERRUPT HANDLER?
Yes
No
RELOAD REGISTERS
UPDATE BLOCK SERIAL NUMBER AND INSERT INTO BUFFER
RELOAD REGISTERS
COUNT RECORD AS LOST
CALCULATE LENGTH OF PHYSICAL RECORD AND STORE IN DCW
OPEN .CRACF GATE
ACFLE
STORE LENGTH OF PHYSICAL RECORD DATA IN WORD 0 OF RECORD
.GRELC
RESET STOP FLAG ON GEPOP I/O ENTRY
A
CALCULATE LOGICAL CHANNEL INDEX IN X2
.SHUTS .CRQGT
2
2
TSX3 STIO START I/O
REQUESTING COURTESY CALL AT ACFCC
SET ALTERNATE BUFFER AS NEW CURRENT BUFFER
INITIALIZE CURRENT POSITION FOR CURRENT BUFFER
SET BUFFER JUST WRITTEN AS ALTERNATE BUFFER
LOAD X1 WITH ADDRESS OF CURRENT LOGICAL RECORD
ACFL1


ACTFL (EP13)
.MIOS
ACFCC
COURTESY CALL PAID WHEN I/O COMPLETE
IS STATUS ABORT?
Yes
No
CHANNEL READY?
Yes
CCSOK
No
END OF TAPE?
No
Yes
.CALL .MACTS,1 EXCHANGE TAPE
CCSOK
.QUEUE GET AN I/O ENTRY
ENTRY ASSIGNED?
No
.GRELC
Yes
STORE INITIALIZED SKELETON IN ENTRY
3
CCSNG
SET ERROR ENTRY IN BUILDING STATUS
.CALL .MACTS,2 EXCHANGE TAPE
UPDATE BLOCK SERIAL NUMBER IN .CRACF AND STORE IN PHYSICAL RECORD
RESET ANY GEPR RESIDUE IN ENTRY
3
STORE CURRENT BUFFER ADDRESS IN DCW
STORE DCW POINTER AND STATUS RETURN POINTER OFFSETS
SET STOP FLAG ON IN ENTRY
CCLNK
RELATIVIZE I/O ENTRY ADDRESS
.LINKF LINK ENTRY FIRST
.GENDC END COURTESY CALL

MAIN IOS MODULE INITIALIZATION

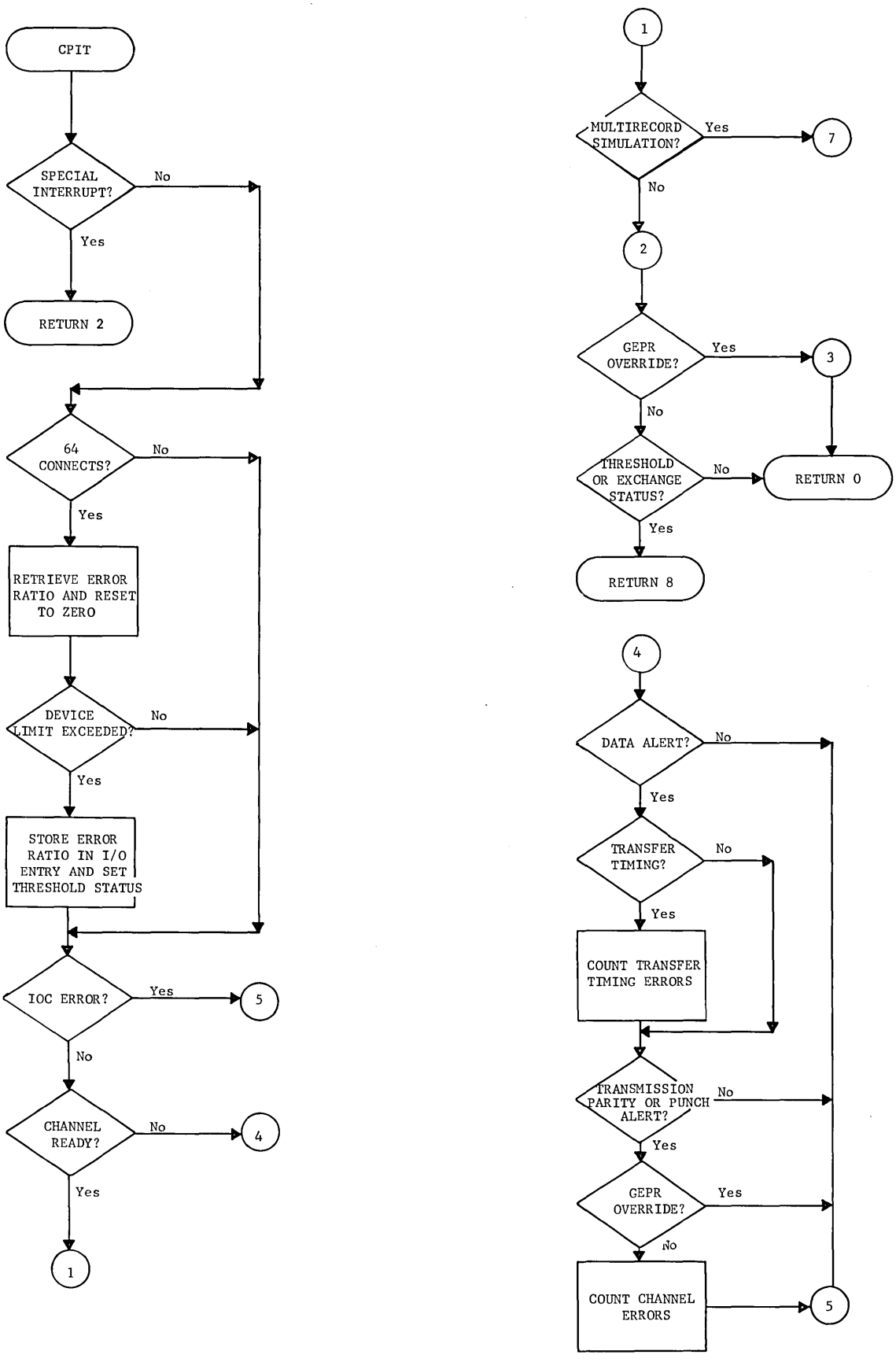


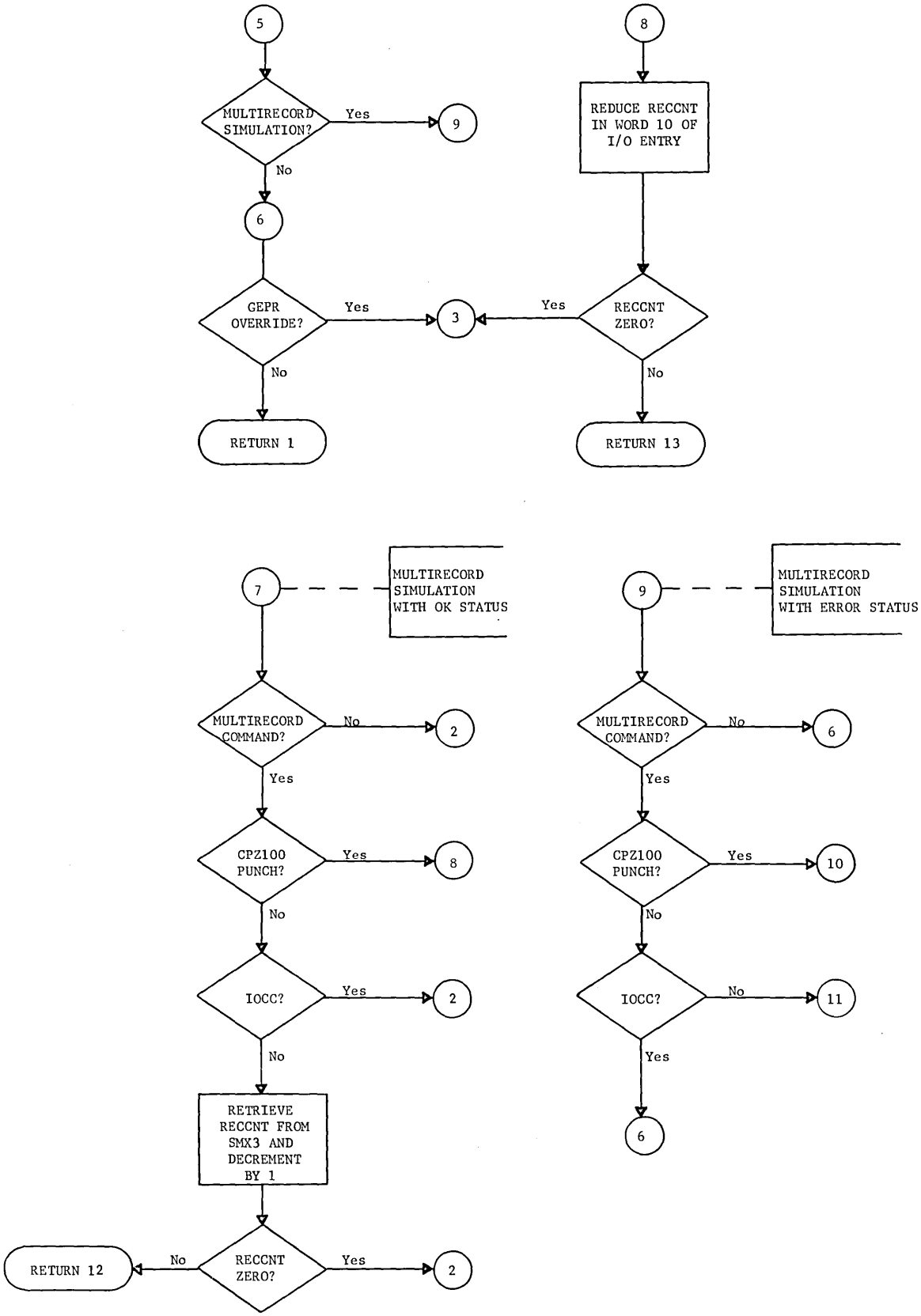

.IIOS
.MIOS
2
INITIALIZE FOR FOR IOC
3
INITIALIZE FOR FIRST CHANNEL
4
CAN CHANNEL BE USED?
No
5a
Yes
IS IT CPZ100 PUNCH?
No
Yes
SET CPZ100 PUNCH FLAG
MATCH TYPE CODE IN TABLE TO GET MODULE NUMBER
No Match
DRL SYSTEM ERROR
Match
5
INCREMENT INDEX FOR NEXT CHANNEL
No
5
IS MODULE ALREADY IN LIST?
Yes
No
STORE MODULE NUMBER IN LIST, INCREMENT COUNT
5a
ALL CHANNELS CHECKED?
Yes
ALL IOC CHECKED?
No
3
Yes
LOAD AR RETURN INFORMATION
SIMULATION IN SLAVE MODE?
No
7
Yes
6


.IIOS
.MIOS
6
SLAVE MODE SIMULATION
IOCB CONFIGURED?
Yes
No
CPZ100 PUNCH CONFIGURED?
No
Yes
SET UP MULTIRECORD SIMULATION LOGIC
LOAD QR WITH SLAVE SIMULATION LOAD ADDRESS
TRA 0,1
7
MASTER MODE
IOCB CONFIGURED?
No
Yes
LOAD QR WITH MULTIRECORD SIMULATION LOAD ADDRESS
TRA 0,1
CPZ100 PUNCH CONFIGURED?
No
Yes
SET UP MULTIRECORD SIMULATION
LOAD QR WITH NORMAL LOAD ADDRESS
TRA 0,1

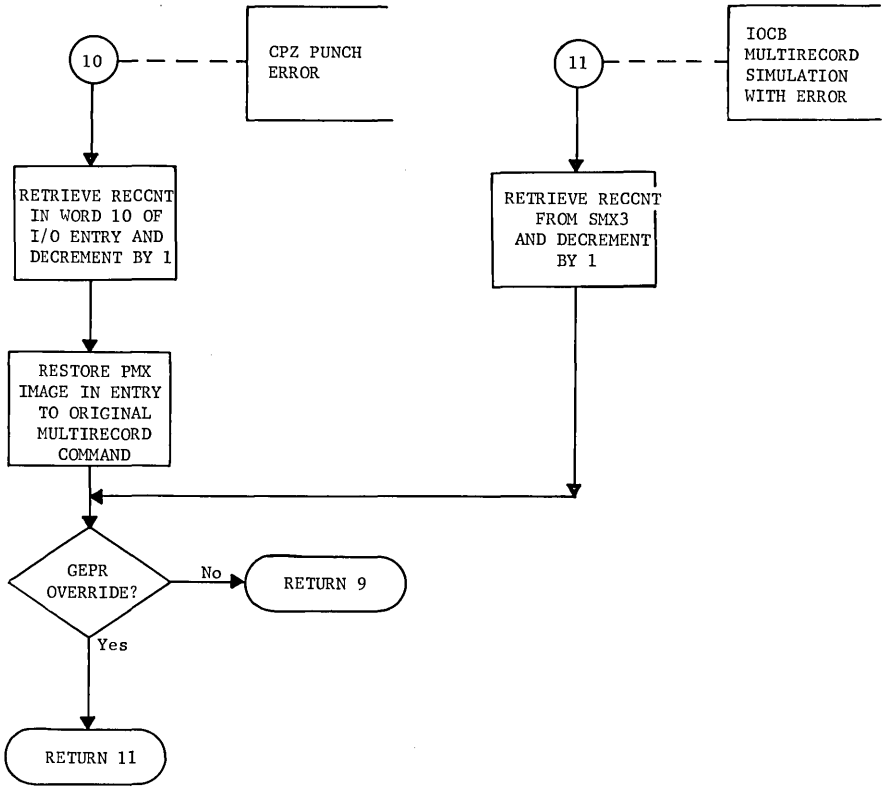
CPIT (EP1)
.MCPIO

CARD PUNCH INTERRUPT HANDLER

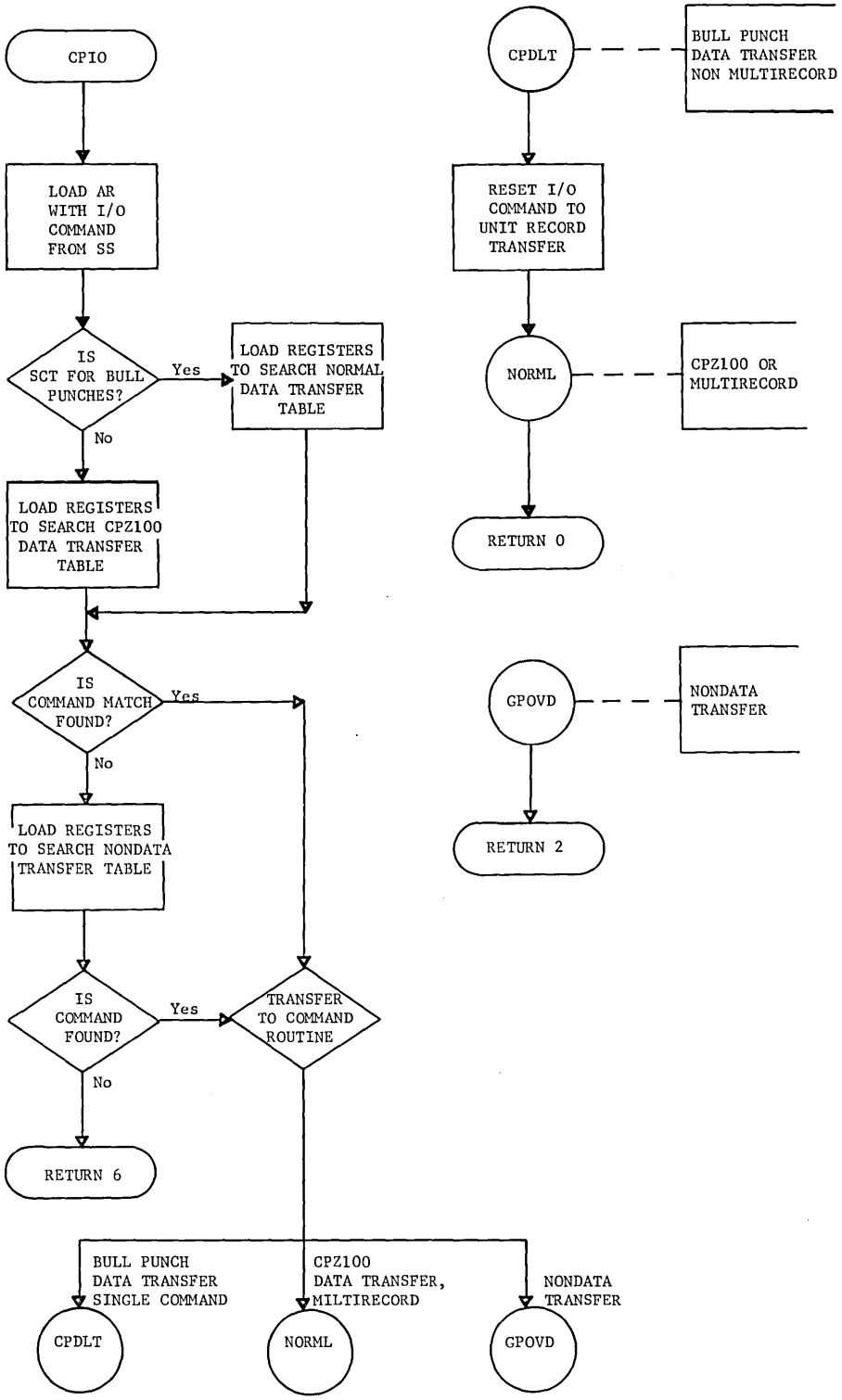



5
MULTIRECORD SIMULATION?
Yes
9
No
6
GEPR OVERRIDE?
Yes
3
No
RETURN 1
8
REDUCE RECCNT IN WORD 10 OF I/O ENTRY
Yes
RECCNT ZERO?
No
RETURN 13
7
MULTIRECORD SIMULATION WITH OK STATUS
MULTIRECORD COMMAND?
No
2
Yes
CPZ100 PUNCH?
Yes
8
No
IOCC?
Yes
2
No
RETRIEVE RECCNT FROM SMX3 AND DECREMENT BY 1
RECCNT ZERO?
Yes
2
No
RETURN 12
9
MULTIRECORD SIMULATION WITH ERROR STATUS
MULTIRECORD COMMAND?
No
6
Yes
CPZ100 PUNCH?
Yes
10
No
IOCC?
No
11
Yes
6

CPIT (EP1)
.MCPIO

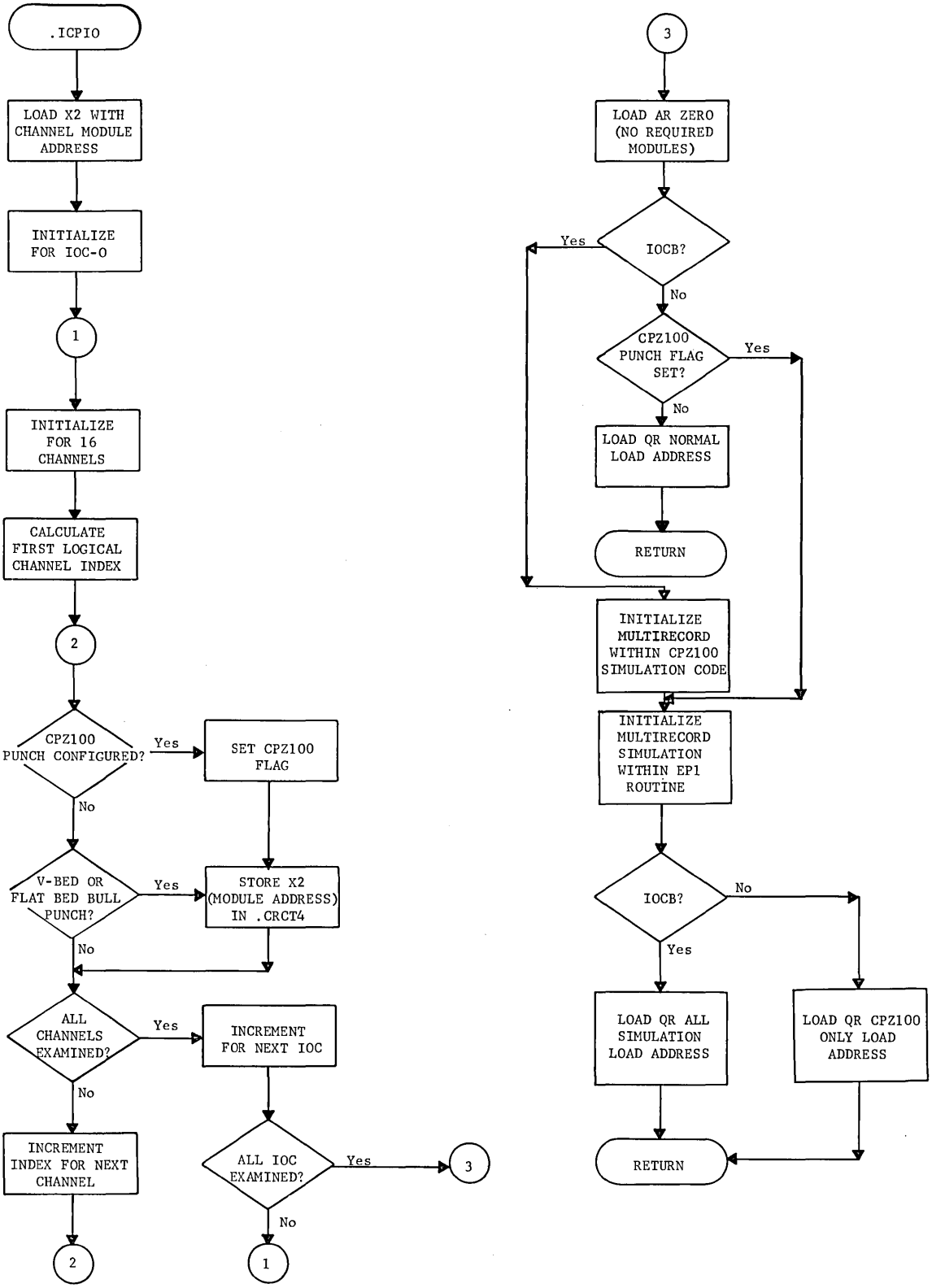


CARD PUNCH REQUEST

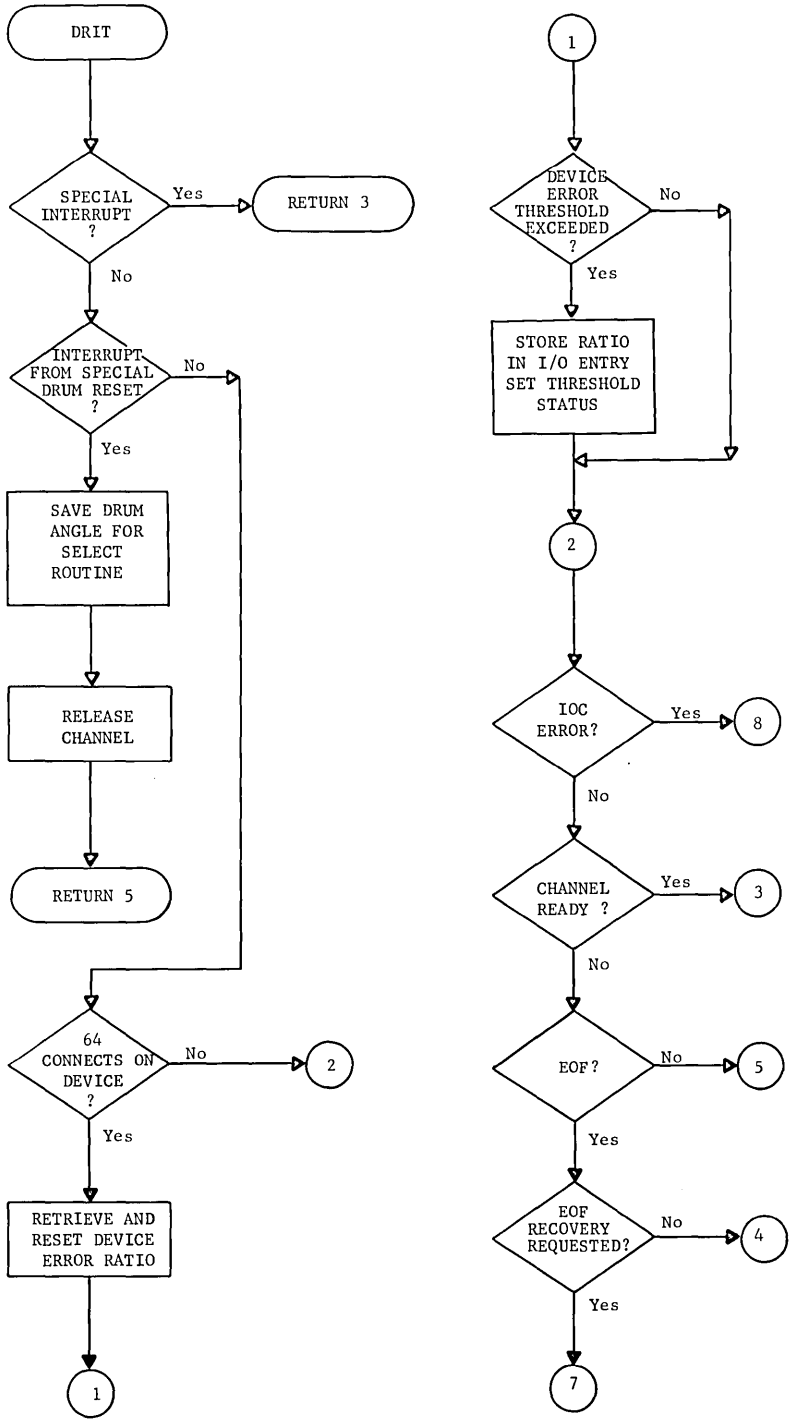


.ICPIO
.MCPIO

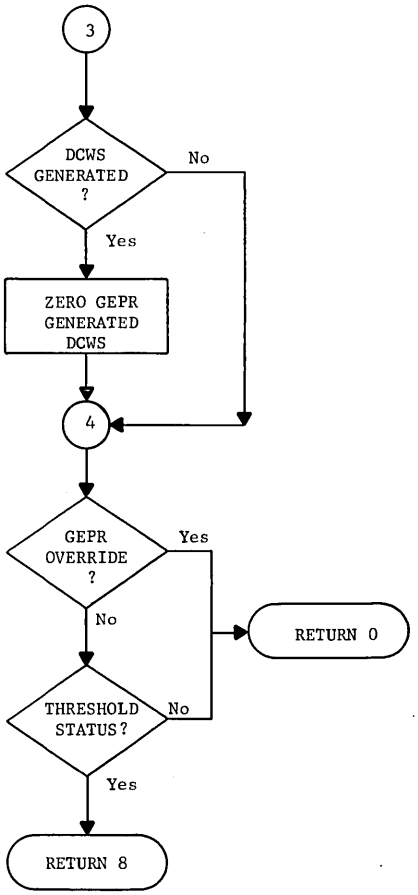
CARD PUNCH INITIALIZATION

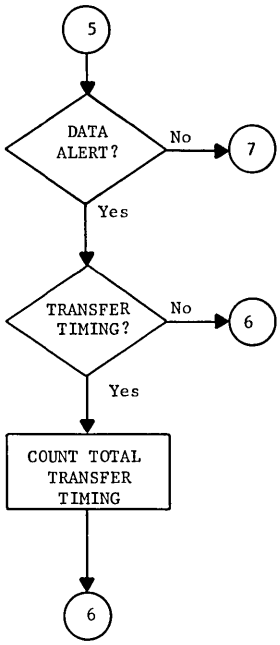


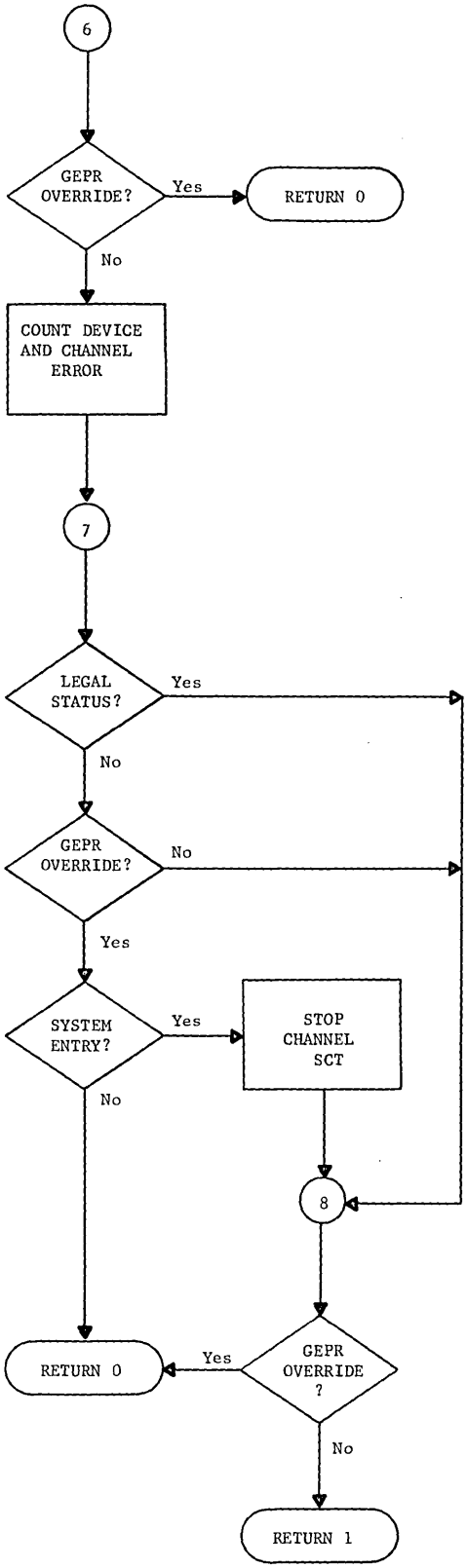
MDS200 INTERRUPT HANDLER



DRIT (EP1)
. MDR20

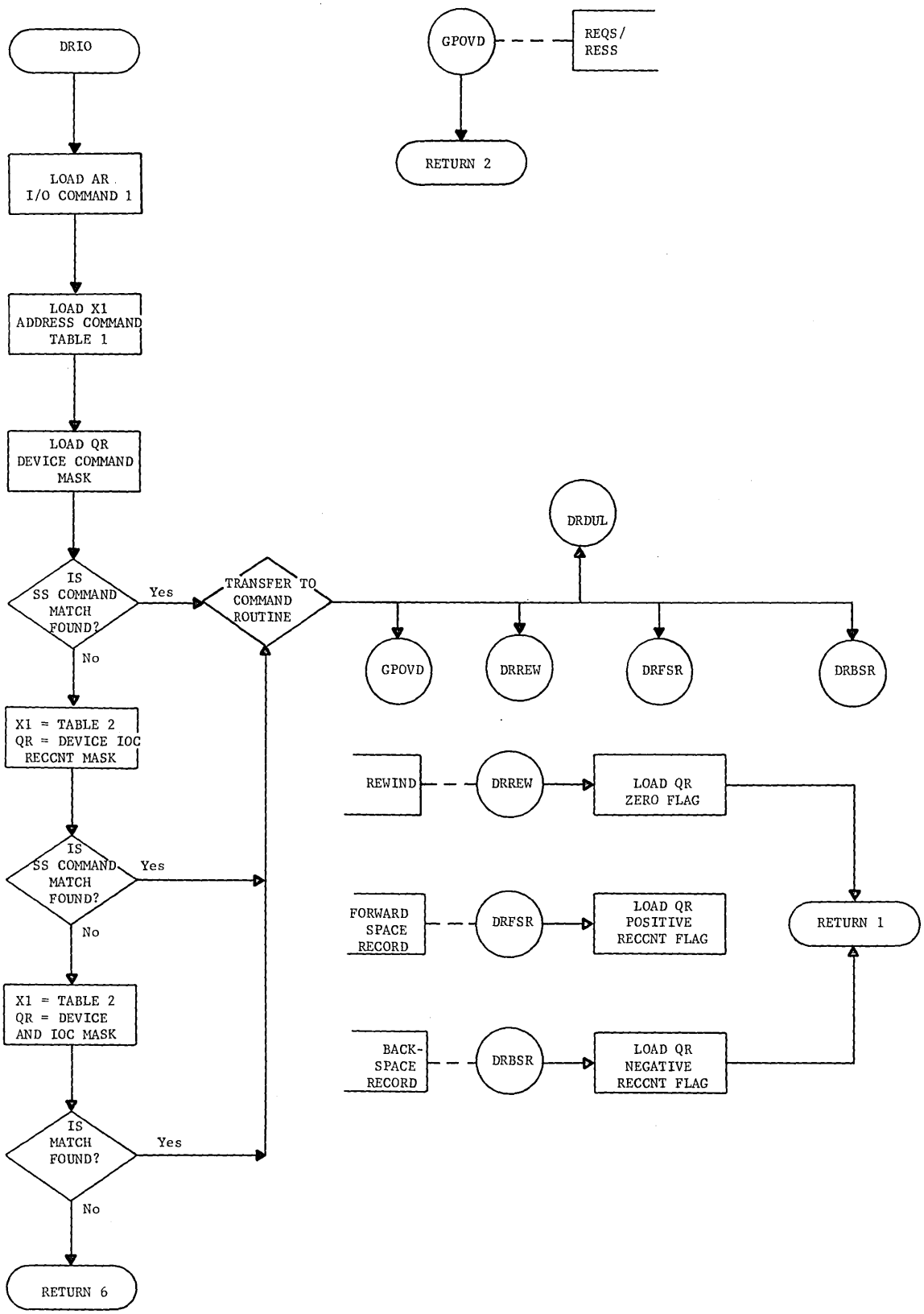


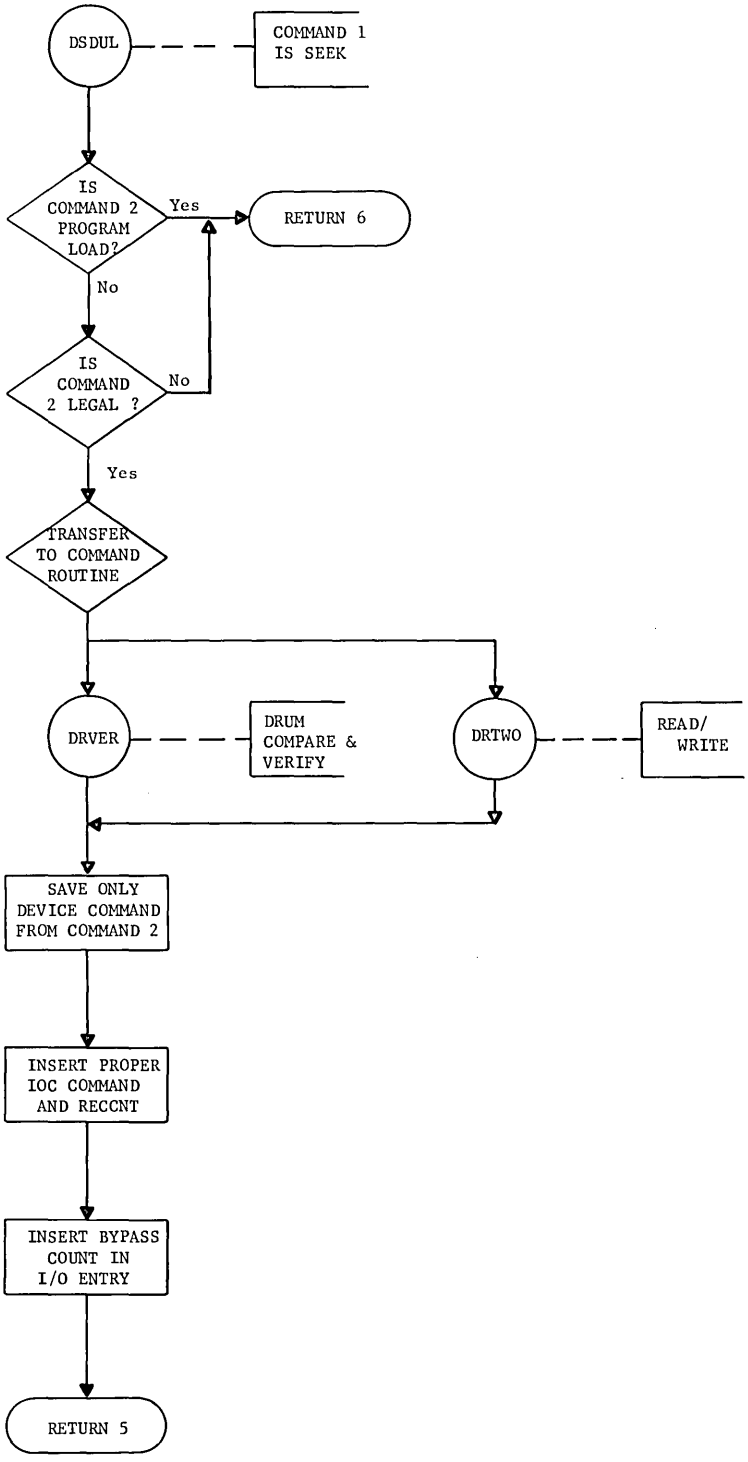



6
GEPR OVERRIDE?
Yes
RETURN 0
No
COUNT DEVICE AND CHANNEL ERROR
7
LEGAL STATUS?
Yes
No
GEPR OVERRIDE?
No
Yes
SYSTEM ENTRY?
Yes
STOP CHANNEL SCT
No
8
RETURN 0
Yes
GEPR OVERRIDE ?
No
RETURN 1

DRIO (EP2)
.MDR20

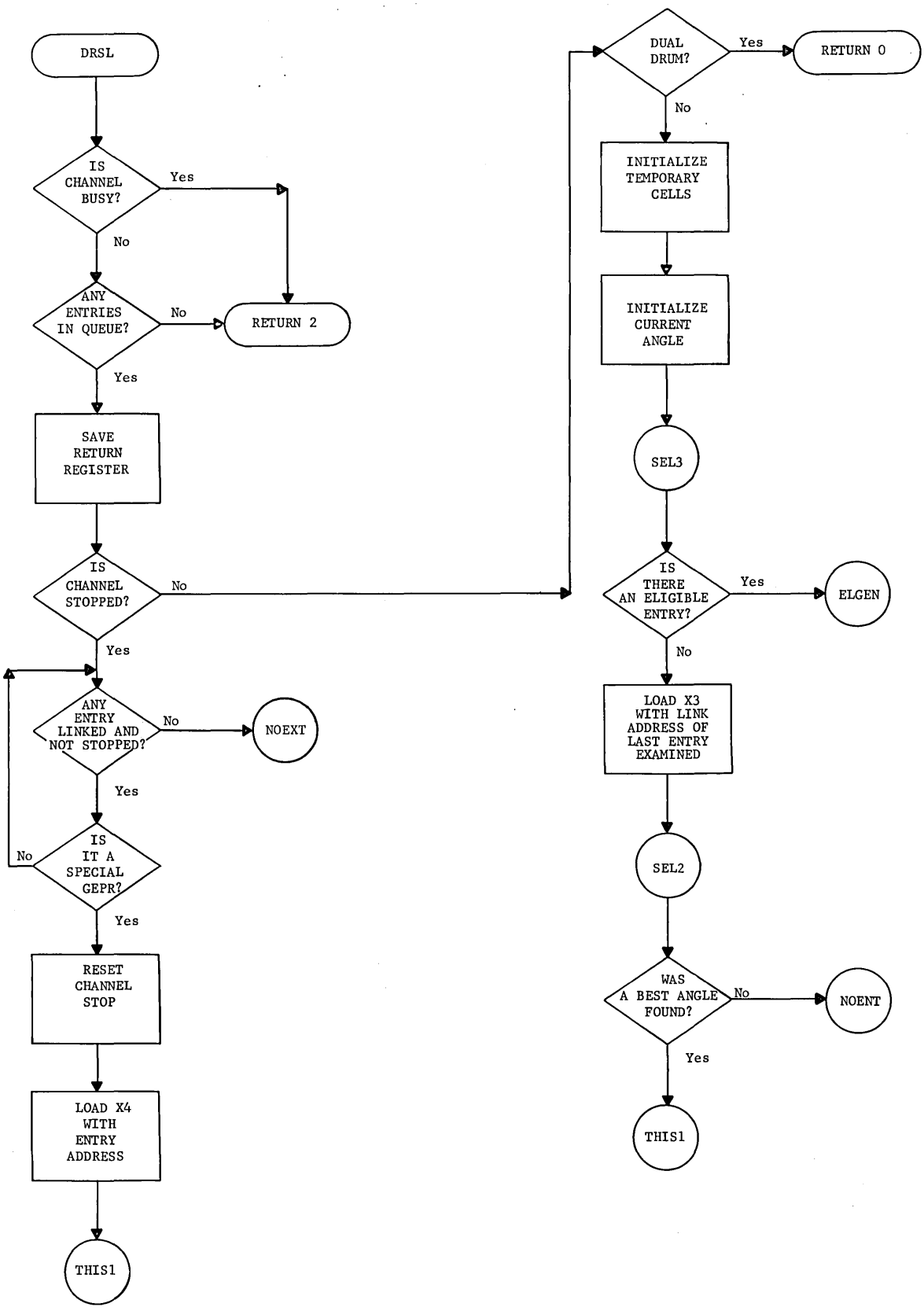
MDS200 REQUEST

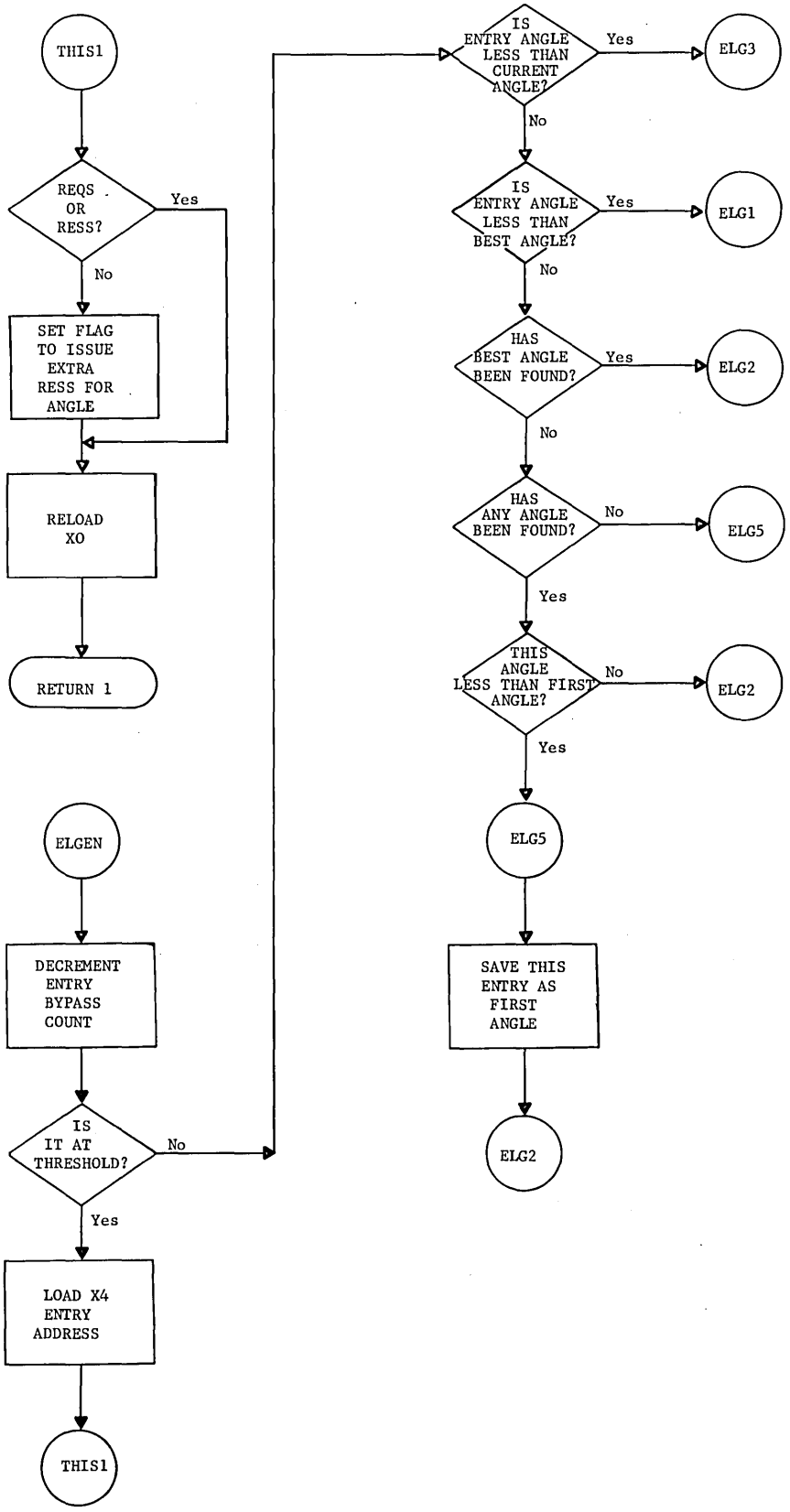


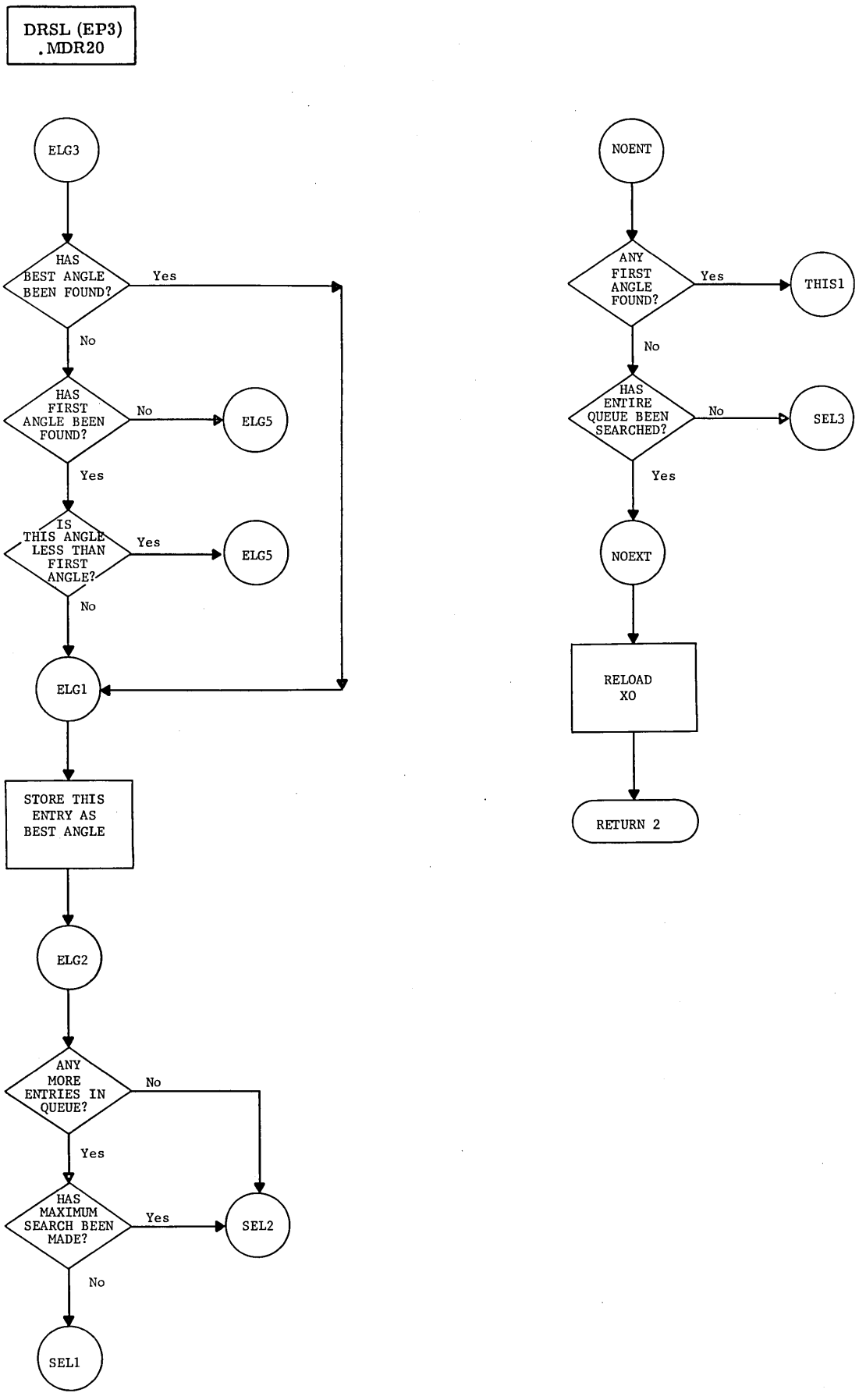

DSDUL
COMMAND 1 IS SEEK
IS COMMAND 2 PROGRAM LOAD?
Yes
RETURN 6
No
IS COMMAND 2 LEGAL ?
No
Yes
TRANSFER TO COMMAND ROUTINE
DRVER
DRUM COMPARE & VERIFY
DRTWO
READ/ WRITE
SAVE ONLY DEVICE COMMAND FROM COMMAND 2
INSERT PROPER IOC COMMAND AND RECCNT
INSERT BYPASS COUNT IN I/O ENTRY
RETURN 5

DRSL (EP3)
.MDR20

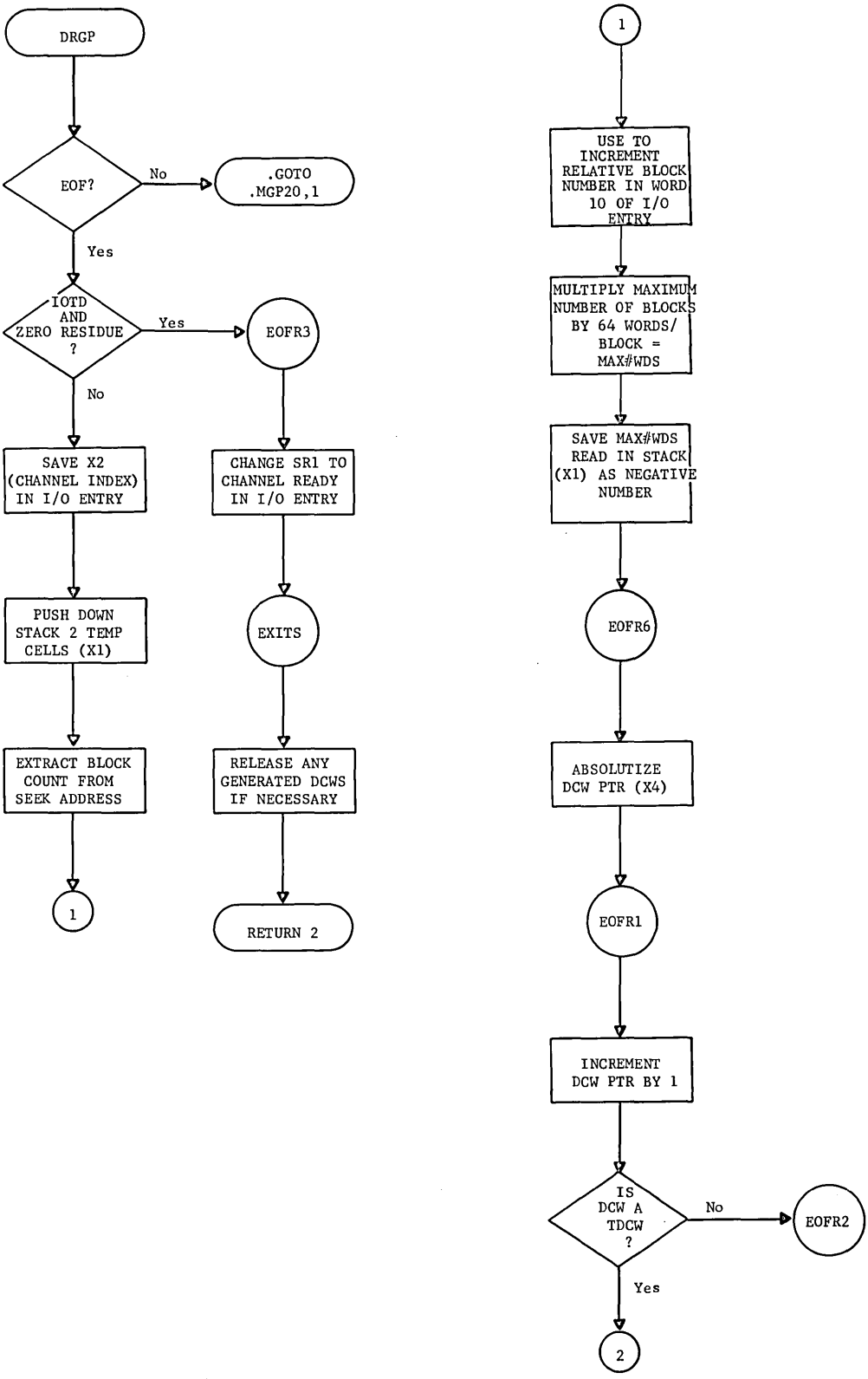
MDS200 SELECT

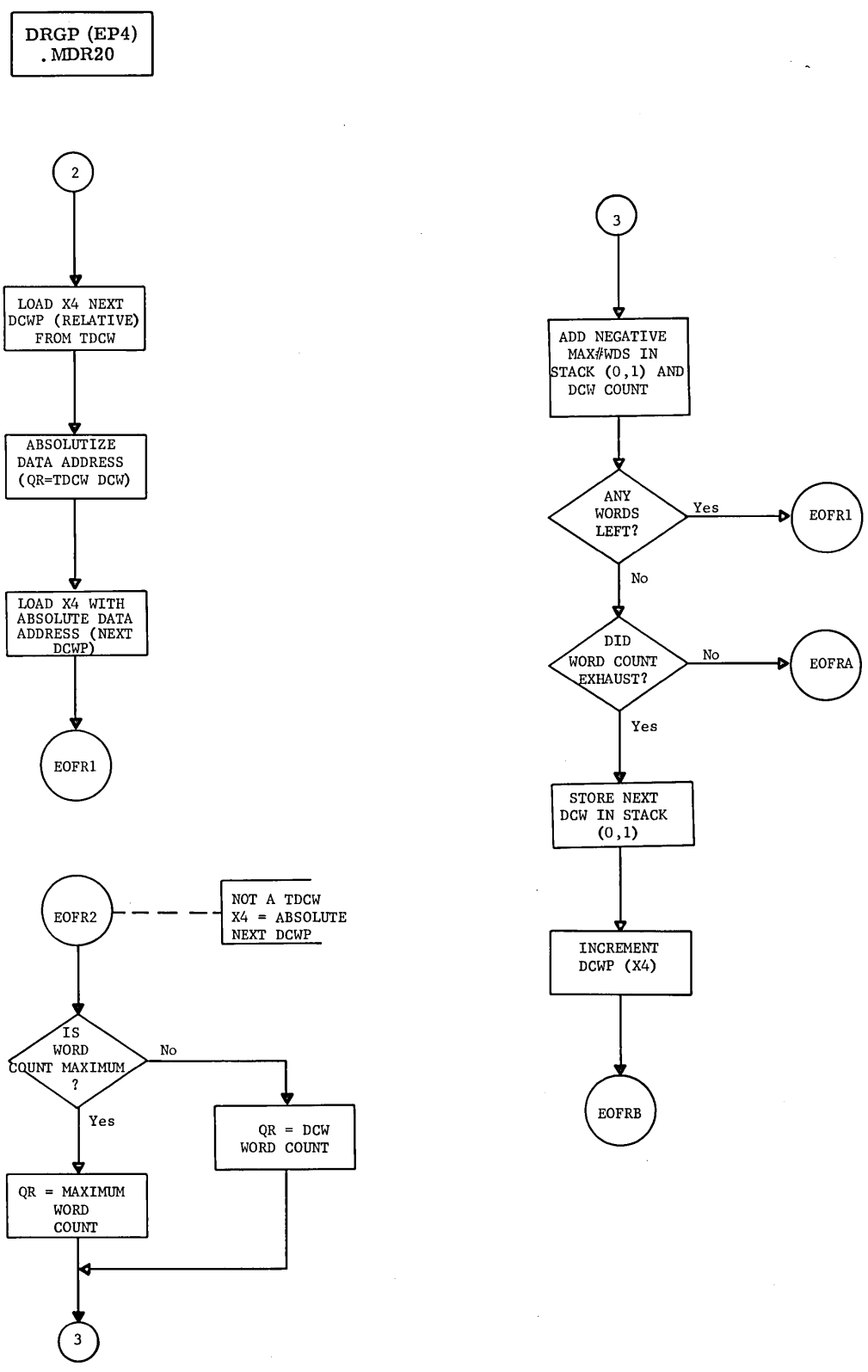


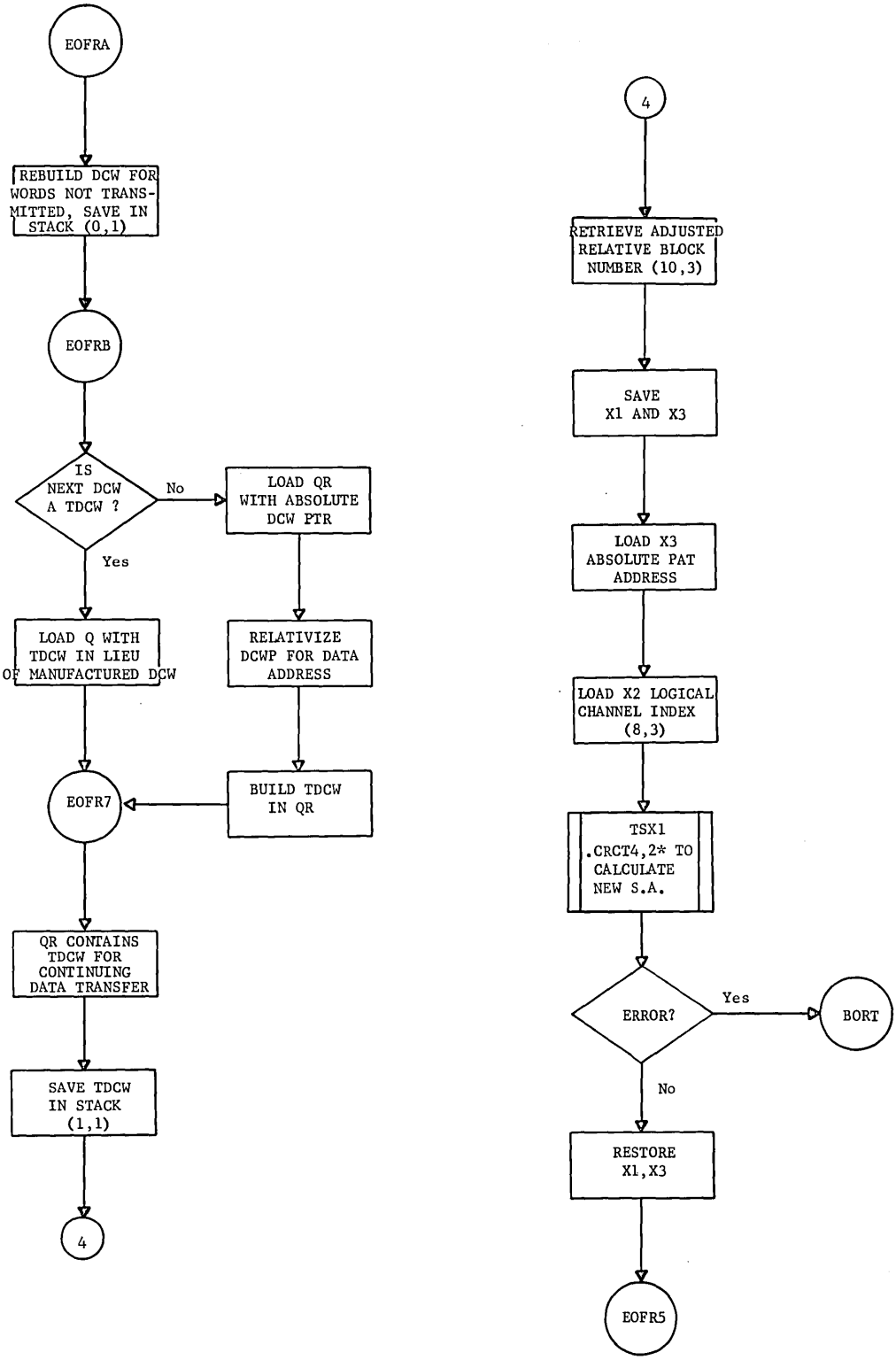

THIS1
REQS OR RESS?
Yes
No
SET FLAG TO ISSUE EXTRA RESS FOR ANGLE
RELOAD XO
RETURN 1
ELGEN
DECREMENT ENTRY BYPASS COUNT
IS IT AT THRESHOLD?
No
Yes
LOAD X4 ENTRY ADDRESS
THIS1
IS ENTRY ANGLE LESS THAN CURRENT ANGLE?
Yes
ELG3
No
IS ENTRY ANGLE LESS THAN BEST ANGLE?
Yes
ELG1
No
HAS BEST ANGLE BEEN FOUND?
Yes
ELG2
No
HAS ANY ANGLE BEEN FOUND?
No
ELG5
Yes
THIS ANGLE LESS THAN FIRST ANGLE?
No
ELG2
Yes
ELG5
SAVE THIS ENTRY AS FIRST ANGLE
ELG2

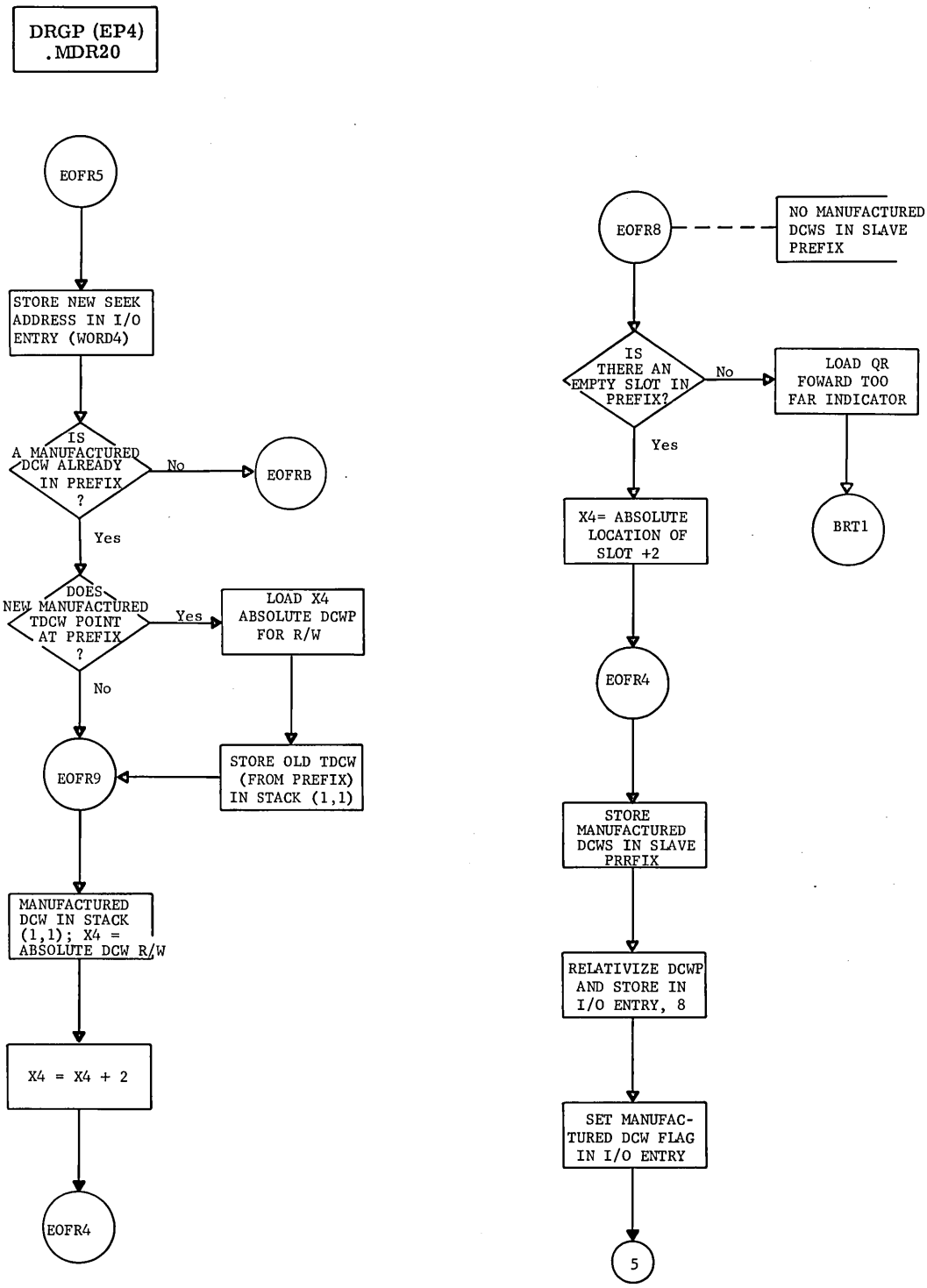

DRSL (EP3)
.MDR20
ELG3
HAS BEST ANGLE BEEN FOUND?
Yes
No
HAS FIRST ANGLE BEEN FOUND?
No
ELG5
Yes
IS THIS ANGLE LESS THAN FIRST ANGLE?
Yes
ELG5
No
ELG1
STORE THIS ENTRY AS BEST ANGLE
ELG2
ANY MORE ENTRIES IN QUEUE?
No
Yes
HAS MAXIMUM SEARCH BEEN MADE?
Yes
SEL2
No
SEL1
NOENT
ANY FIRST ANGLE FOUND?
Yes
THIS1
No
HAS ENTIRE QUEUE BEEN SEARCHED?
No
SEL3
Yes
NOEXT
RELOAD XO
RETURN 2

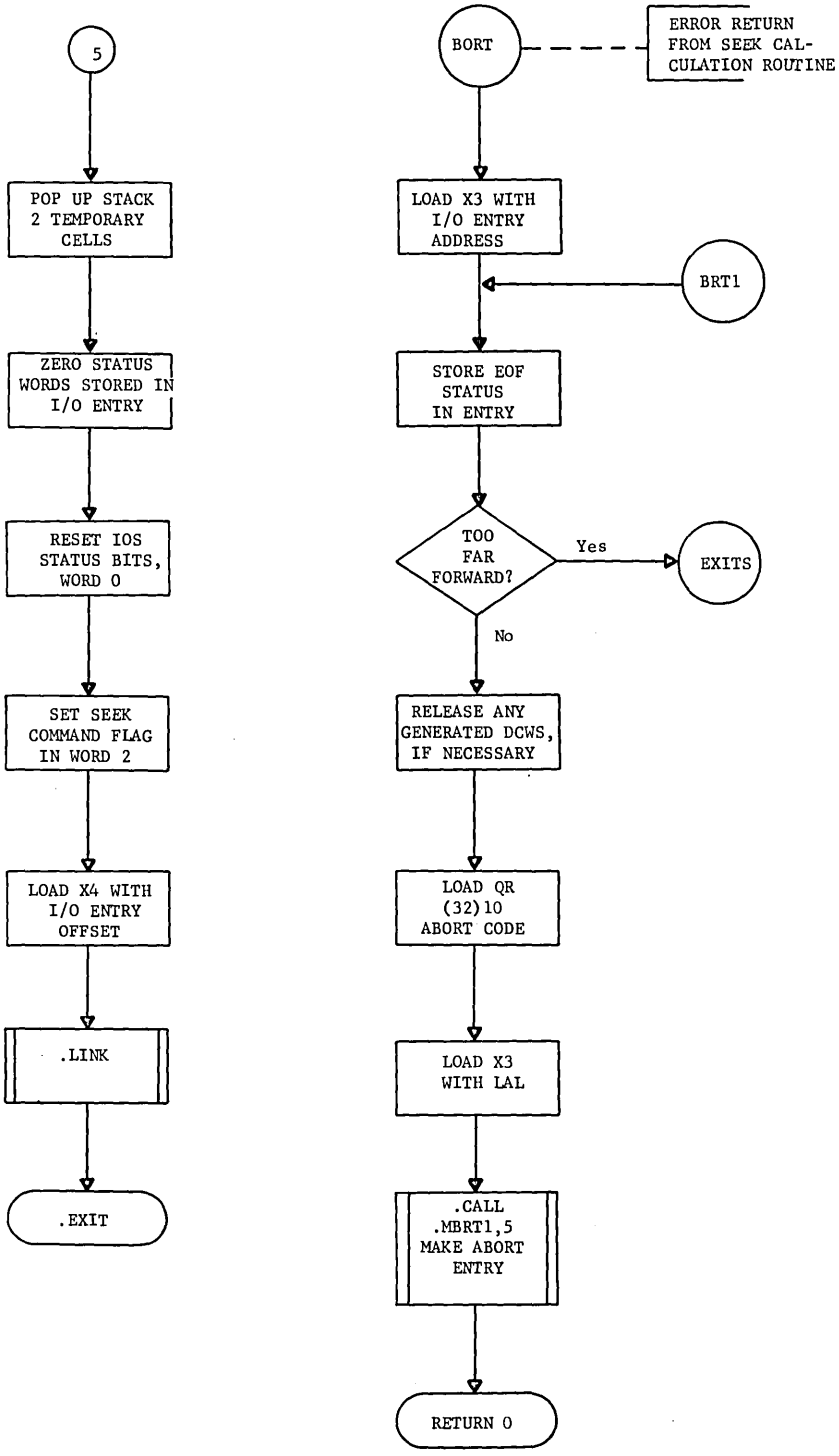
MDS200 ERROR AND EOF RECOVERY




DRGP (EP4)
.MDR20
2
LOAD X4 NEXT DCWP (RELATIVE) FROM TDCW
ABSOLUTIZE DATA ADDRESS (QR=TDCW DCW)
LOAD X4 WITH ABSOLUTE DATA ADDRESS (NEXT DCWP)
EOFR1
EOFR2
NOT A TDCW X4 = ABSOLUTE NEXT DCWP
IS WORD COUNT MAXIMUM ?
No
Yes
QR = DCW WORD COUNT
QR = MAXIMUM WORD COUNT
3
3
ADD NEGATIVE MAX#WDS IN STACK (0,1) AND DCW COUNT
ANY WORDS LEFT?
Yes
EOFR1
No
DID WORD COUNT EXHAUST?
No
EOFRA
Yes
STORE NEXT DCW IN STACK (0,1)
INCREMENT DCWP (X4)
EOFRB

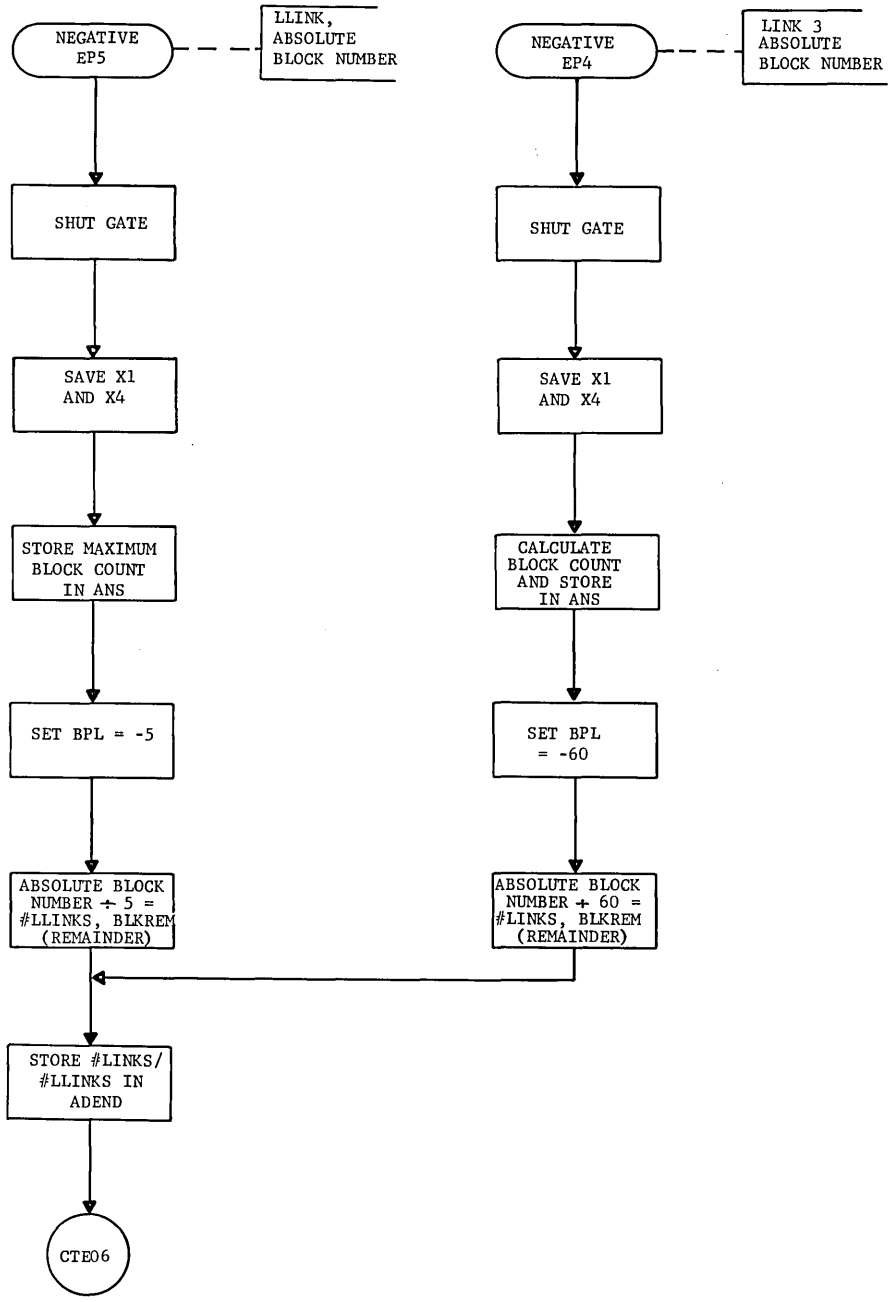

EOFRA
REBUILD DCW FOR WORDS NOT TRANS-MITTED, SAVE IN STACK (0,1)
EOFRB
IS NEXT DCW A TDCW ?
No
Yes
LOAD QR WITH ABSOLUTE DCW PTR
RELATIVIZE DCWP FOR DATA ADDRESS
BUILD TDCW IN QR
LOAD Q WITH TDCW IN LIEU OF MANUFACTURED DCW
EOFR7
QR CONTAINS TDCW FOR CONTINUING DATA TRANSFER
SAVE TDCW IN STACK (1,1)
4
4
RETRIEVE ADJUSTED RELATIVE BLOCK NUMBER (10,3)
SAVE X1 AND X3
LOAD X3 ABSOLUTE PAT ADDRESS
LOAD X2 LOGICAL CHANNEL INDEX (8,3)
TSX1 .CRCT4,2* TO CALCULATE NEW S.A.
ERROR?
Yes
BORT
No
RESTORE X1,X3
EOFR5


DRGP (EP4)
.MDR20
EOFR5
STORE NEW SEEK ADDRESS IN I/O ENTRY (WORD4)
IS A MANUFACTURED DCW ALREADY IN PREFIX ?
No
EOFRB
Yes
DOES NEW MANUFACTURED TDCW POINT AT PREFIX ?
Yes
LOAD X4 ABSOLUTE DCWP FOR R/W
No
EOFR9
STORE OLD TDCW (FROM PREFIX) IN STACK (1,1)
MANUFACTURED DCW IN STACK (1,1); X4 = ABSOLUTE DCW R/W
X4 = X4 + 2
EOFR4
EOFR8
NO MANUFACTURED DCWS IN SLAVE PREFIX
IS THERE AN EMPTY SLOT IN PREFIX?
No
LOAD QR FOWARD TOO FAR INDICATOR
BRT1
Yes
X4= ABSOLUTE LOCATION OF SLOT +2
EOFR4
STORE MANUFACTURED DCWS IN SLAVE PRRFIX
RELATIVIZE DCWP AND STORE IN I/O ENTRY, 8
SET MANUFAC-TURED DCW FLAG IN I/O ENTRY
5

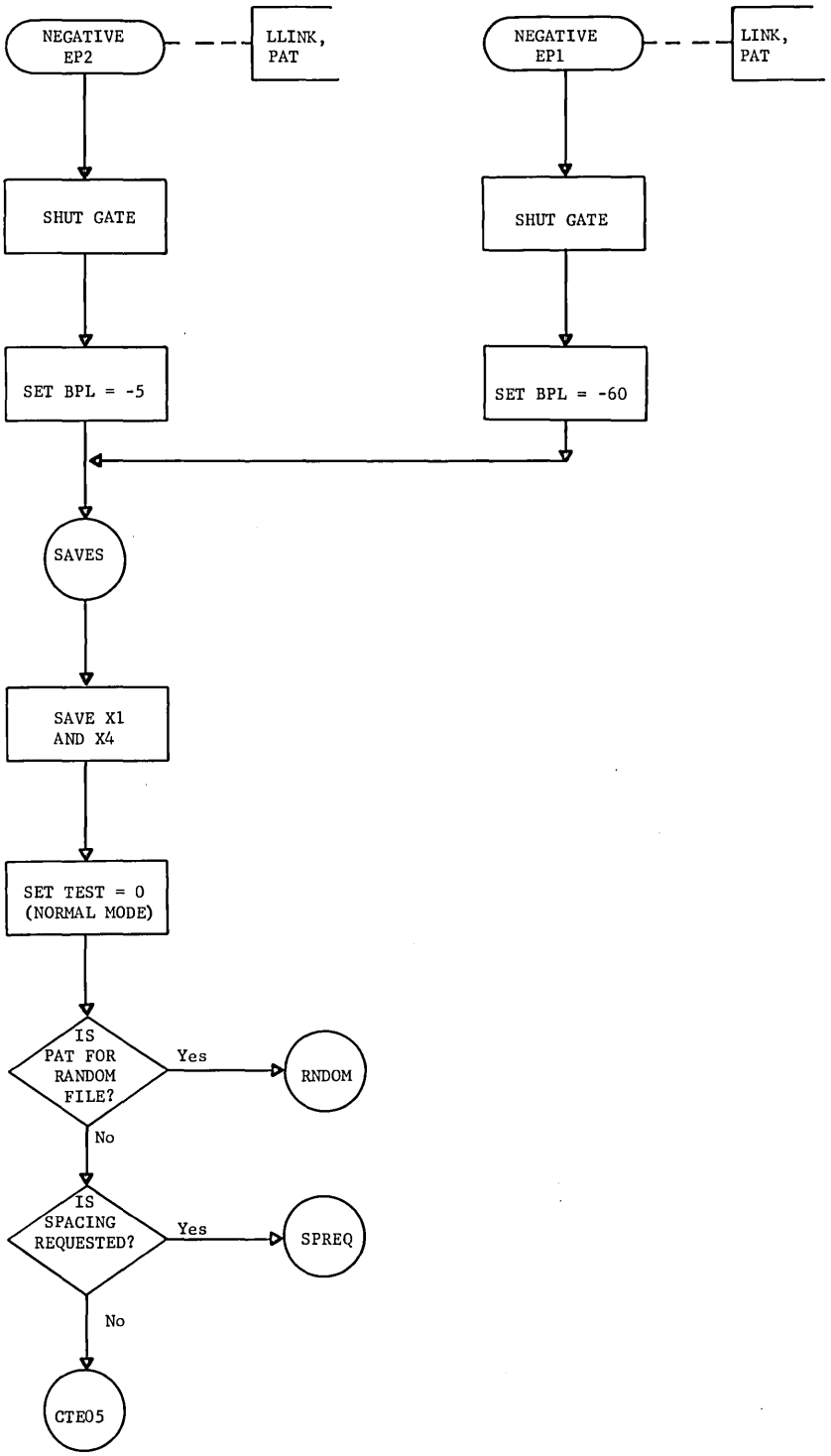

5
POP UP STACK 2 TEMPORARY CELLS
ZERO STATUS WORDS STORED IN I/O ENTRY
RESET IOS STATUS BITS, WORD 0
SET SEEK COMMAND FLAG IN WORD 2
LOAD X4 WITH I/O ENTRY OFFSET
.LINK
.EXIT
BORT
ERROR RETURN FROM SEEK CALCULATION ROUTINE
LOAD X3 WITH I/O ENTRY ADDRESS
BRT1
STORE EOF STATUS IN ENTRY
TOO FAR FORWARD?
Yes
EXITS
No
RELEASE ANY GENERATED DCWS, IF NECESSARY
LOAD QR (32)10 ABORT CODE
LOAD X3 WITH LAL
.CALL .MBRT1,5 MAKE ABORT ENTRY
RETURN 0

NEGATIVE EP
.MDR20

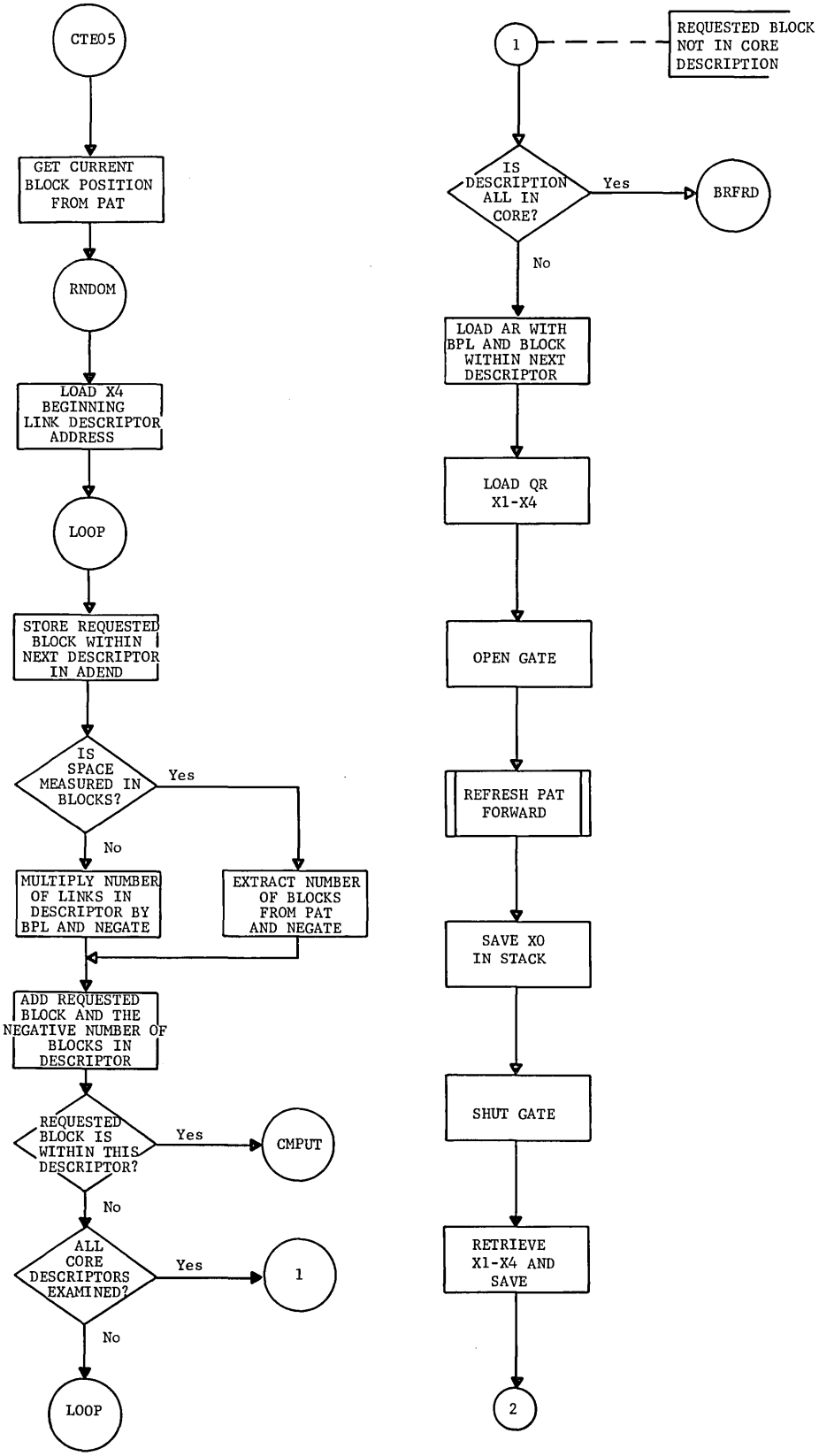
MDS200 NEGATIVE ENTRY POINTS



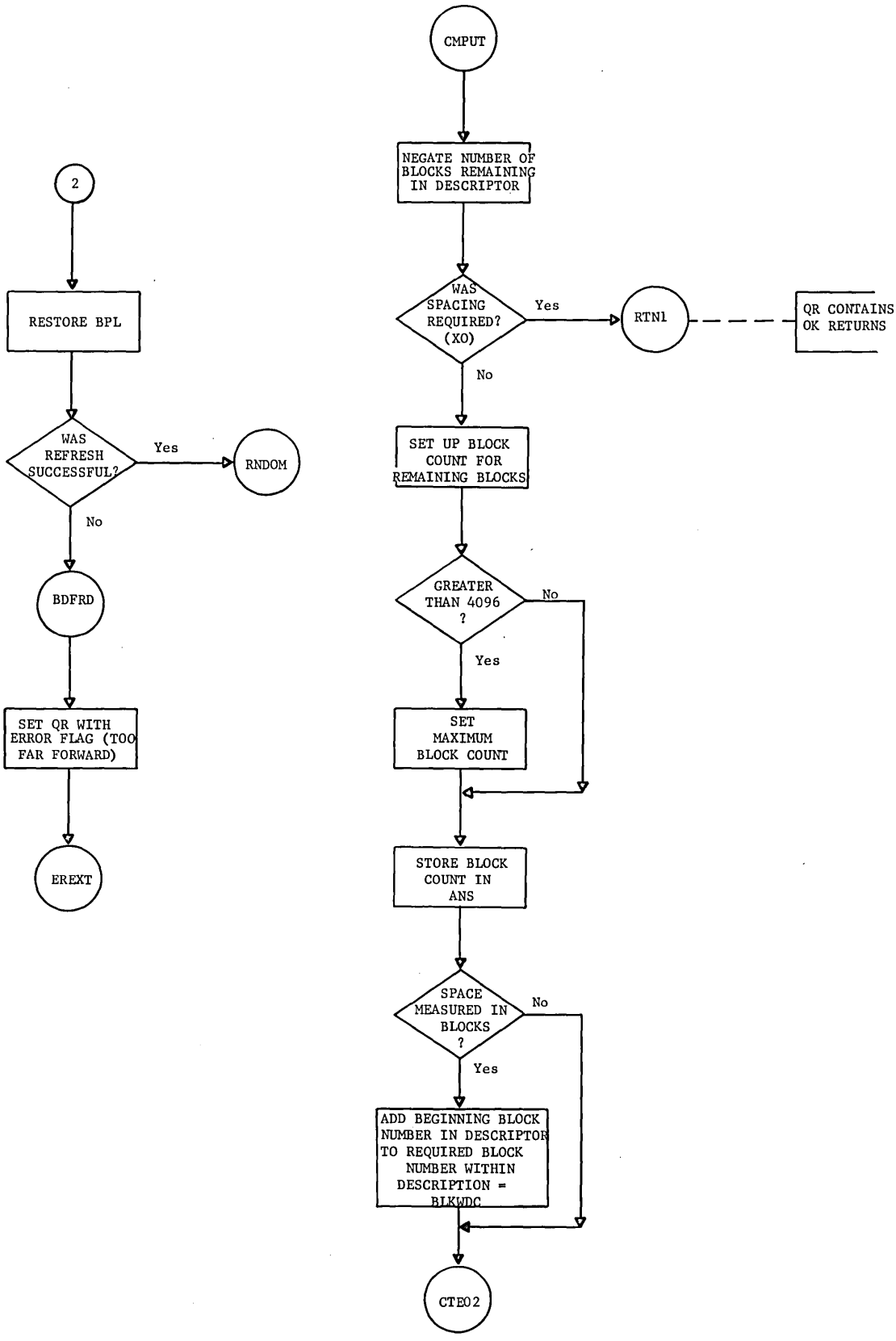
NEGATIVE EP
.MDR20

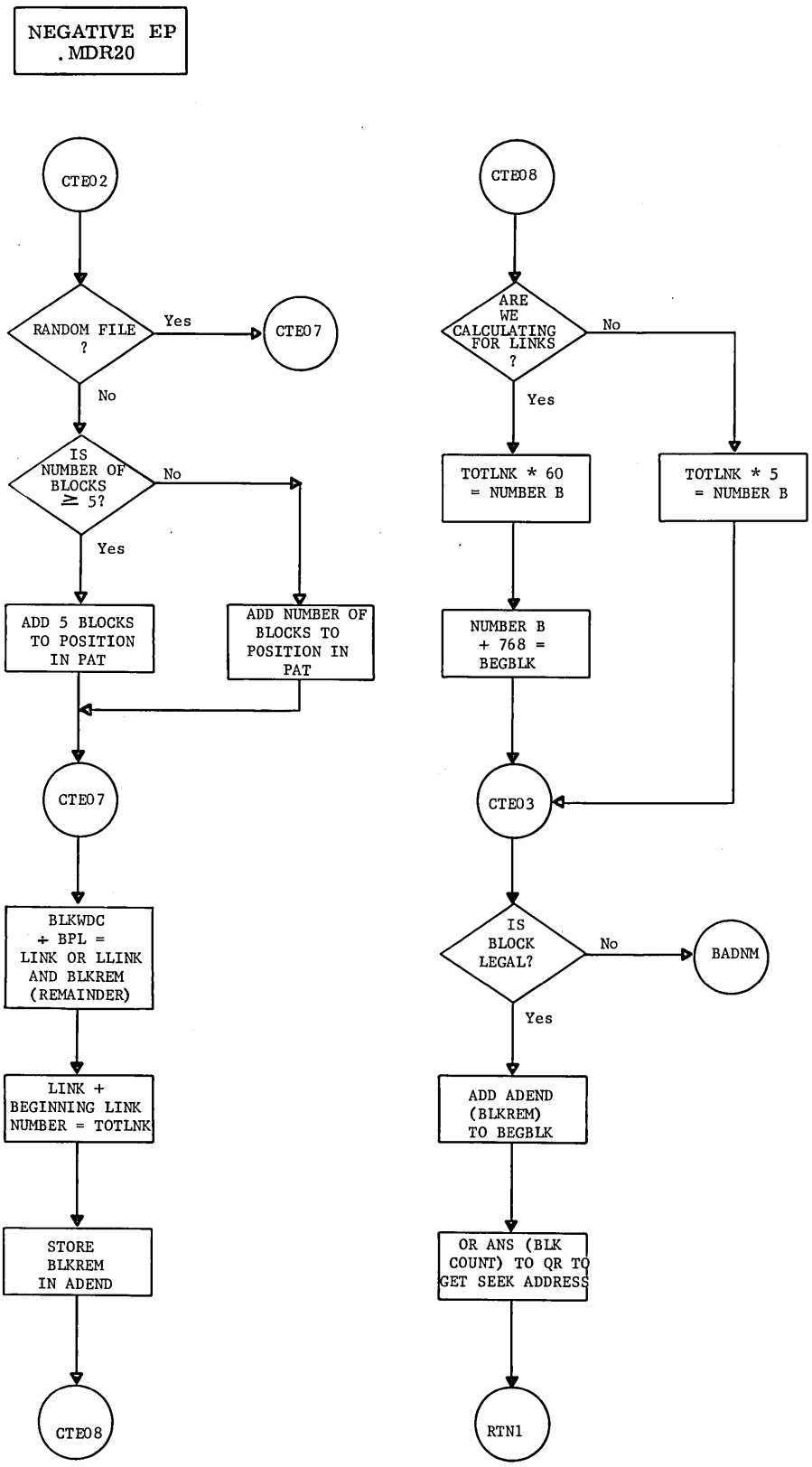


NEGATIVE EP
.MDR20

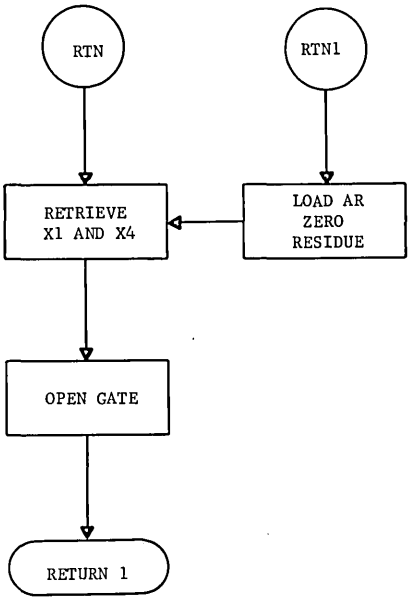


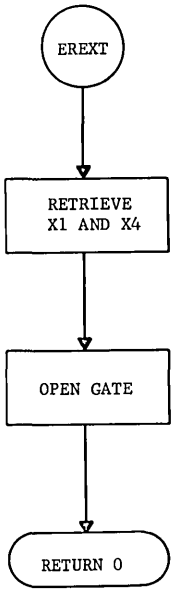
NEGATIVE EP
.MDR20

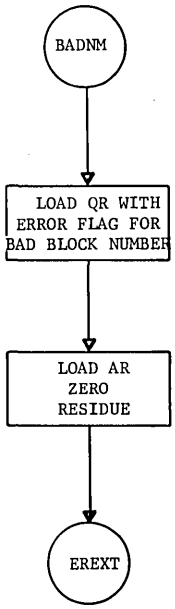



NEGATIVE EP
.MDR20
CTE02
RANDOM FILE ?
Yes
CTE07
No
IS NUMBER OF BLOCKS ≥ 5?
No
Yes
ADD 5 BLOCKS TO POSITION IN PAT
ADD NUMBER OF BLOCKS TO POSITION IN PAT
CTE07
BLKWDC ÷ BPL = LINK OR LLINK AND BLKREM (REMAINDER)
LINK + BEGINNING LINK NUMBER = TOTLNK
STORE BLKREM IN ADEND
CTE08
CTE08
ARE WE CALCULATING FOR LINKS ?
No
Yes
TOTLNK * 60 = NUMBER B
TOTLNK * 5 = NUMBER B
NUMBER B + 768 = BEGBLK
CTE03
IS BLOCK LEGAL?
No
BADNM
Yes
ADD ADEND (BLKREM) TO BEGBLK
OR ANS (BLK COUNT) TO QR TO GET SEEK ADDRESS
RTN1

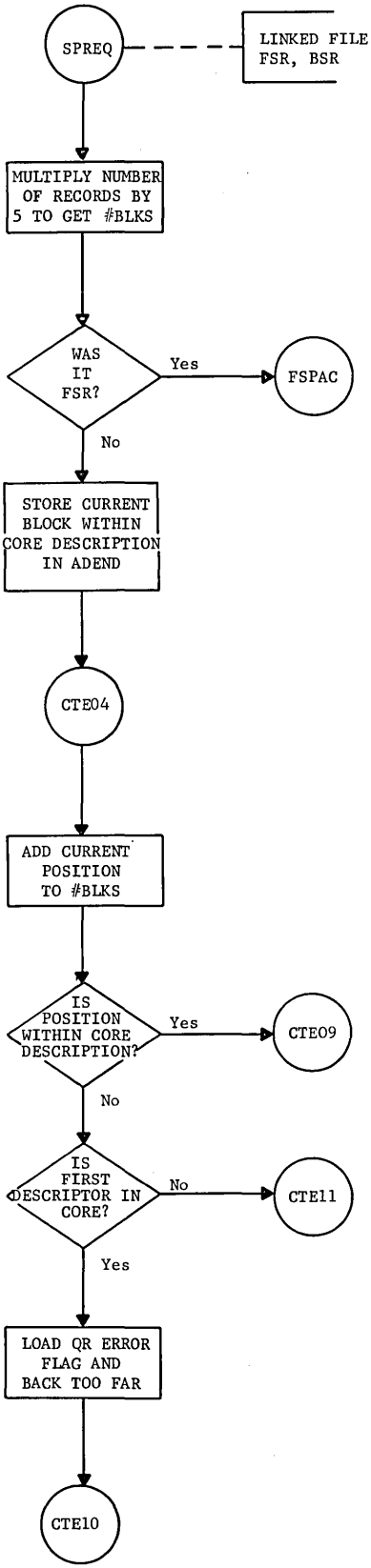
NEGATIVE EP
.MDR20



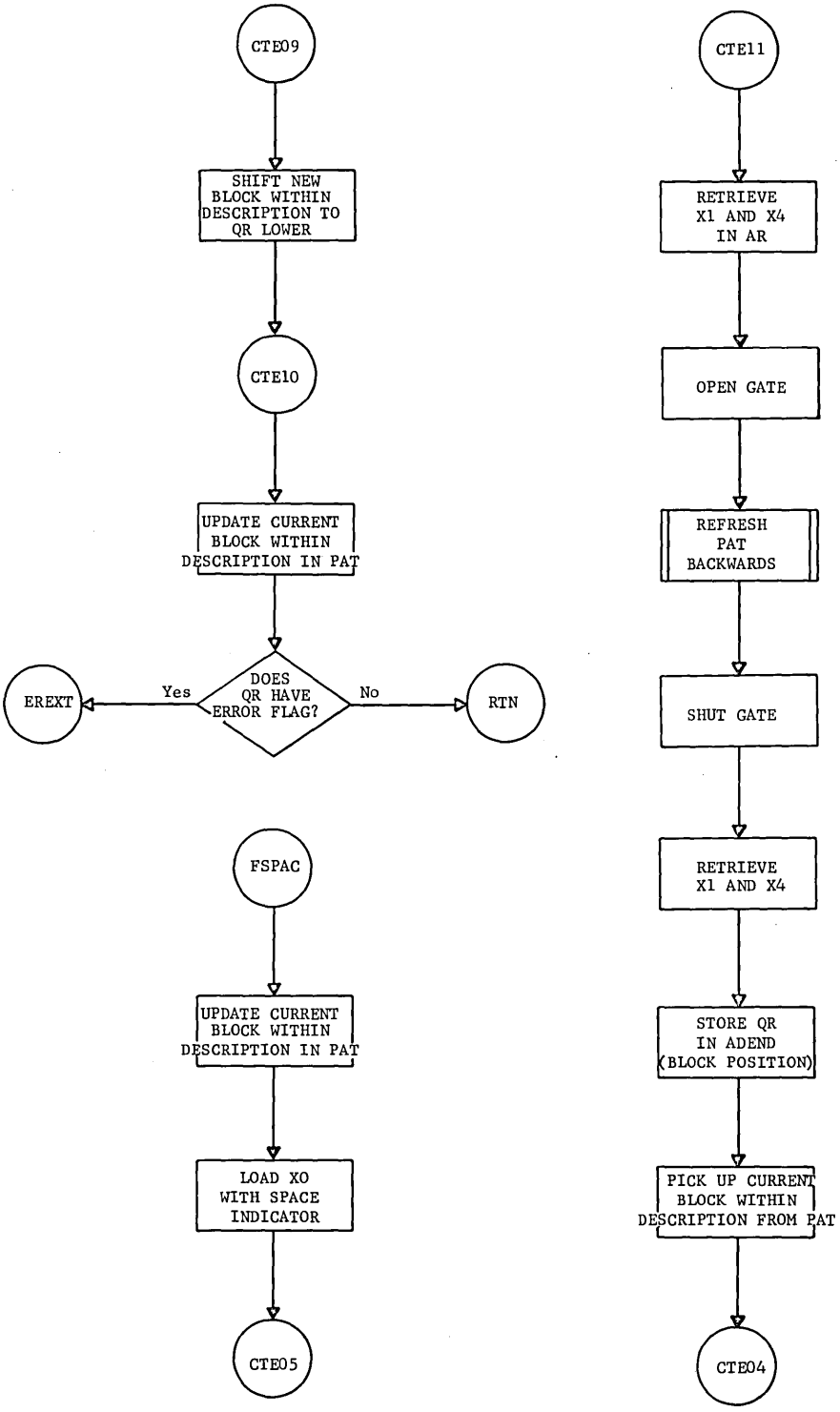




NEGATIVE EP
.MDR20

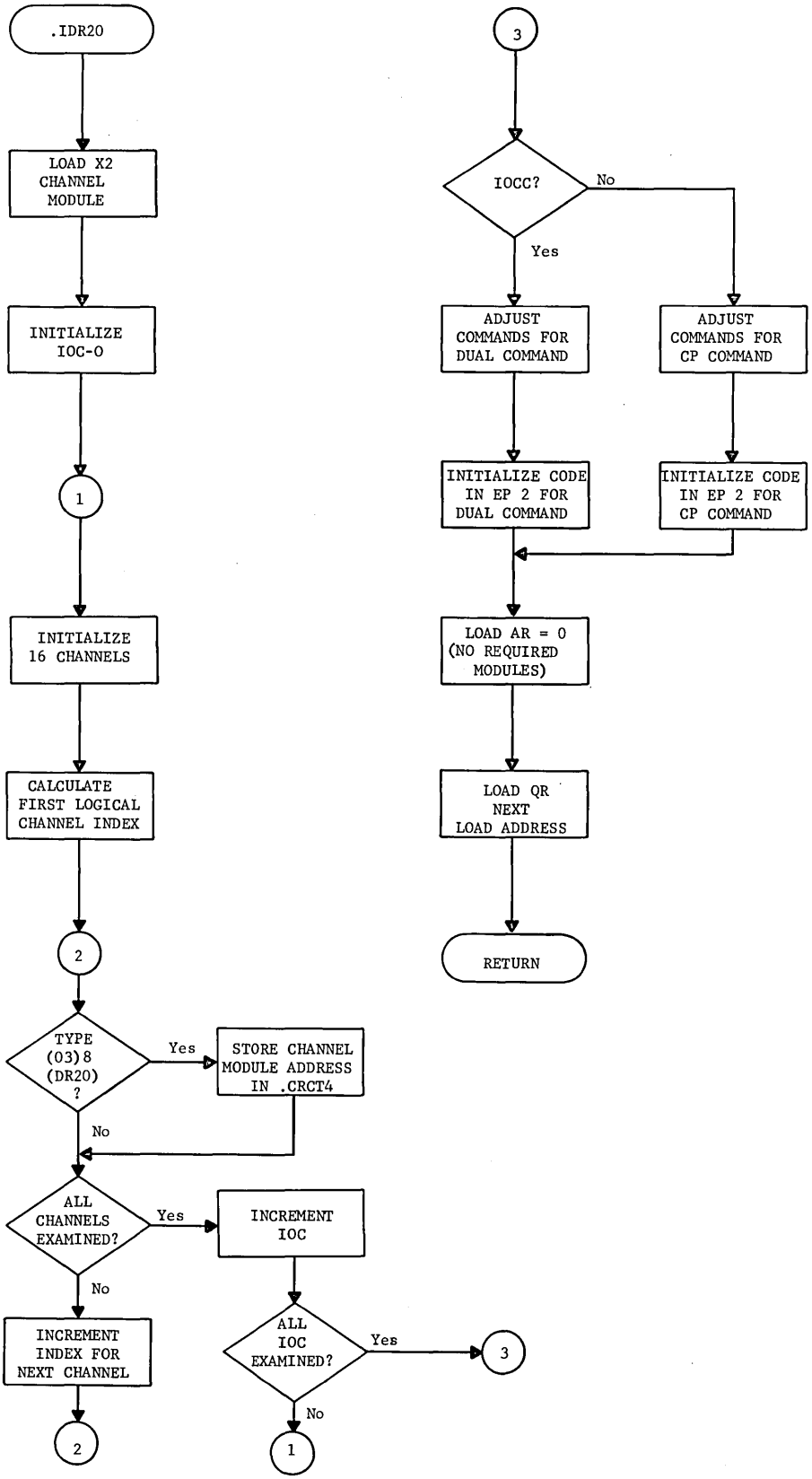


NEGATIVE EP
.MDR20

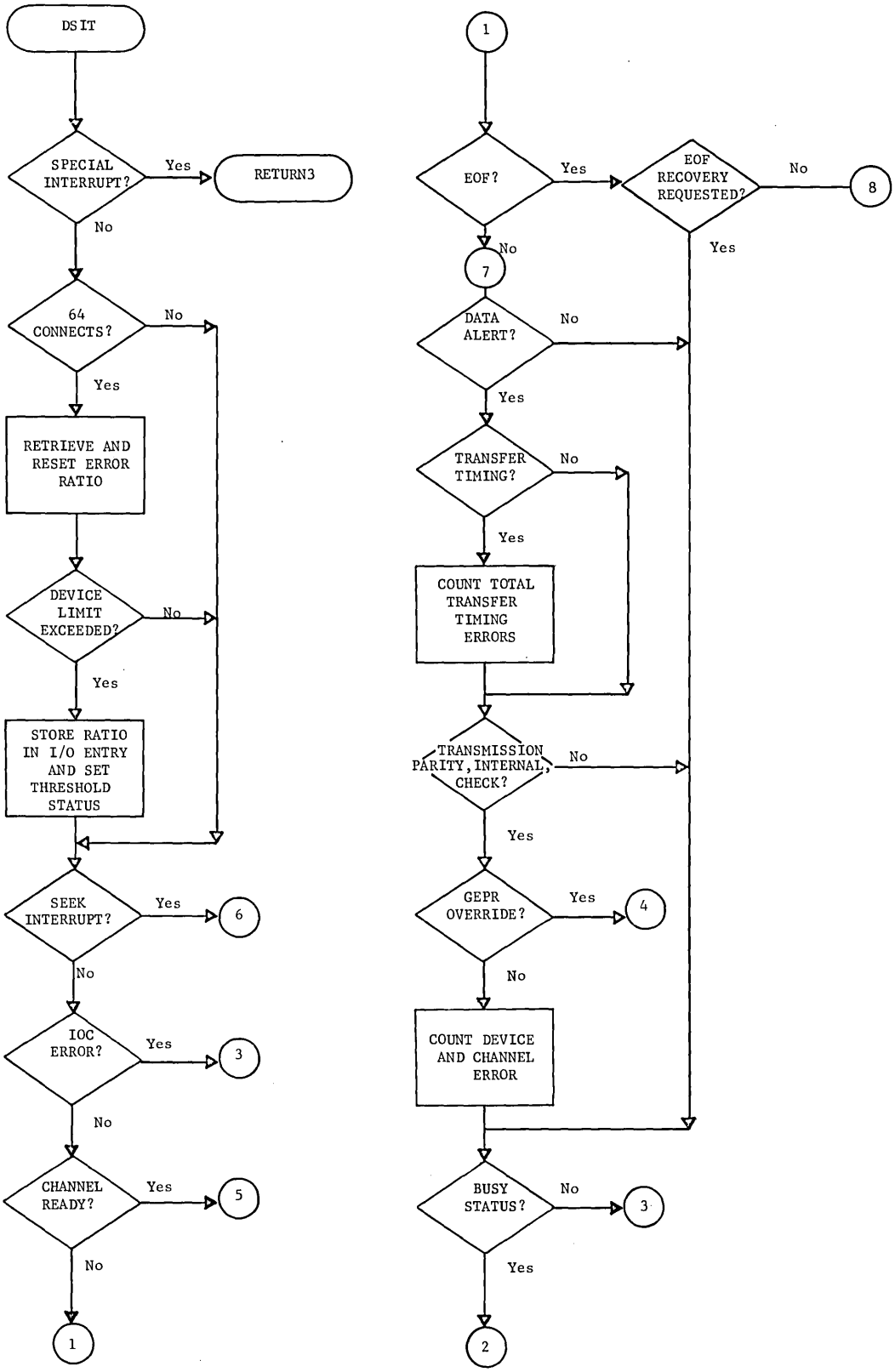


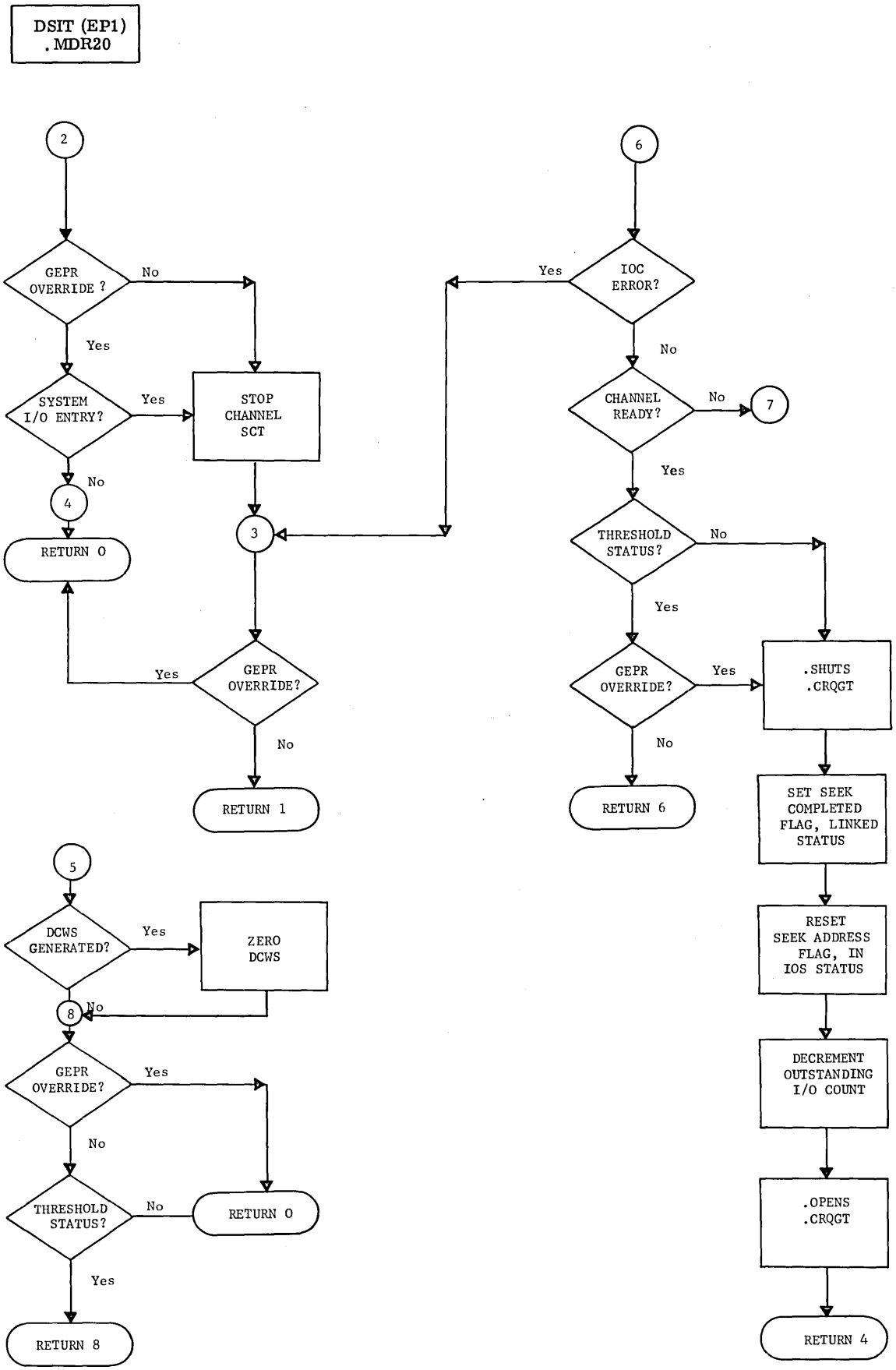
.IDR20
.MDR20

MDS200 INITIALIZATION



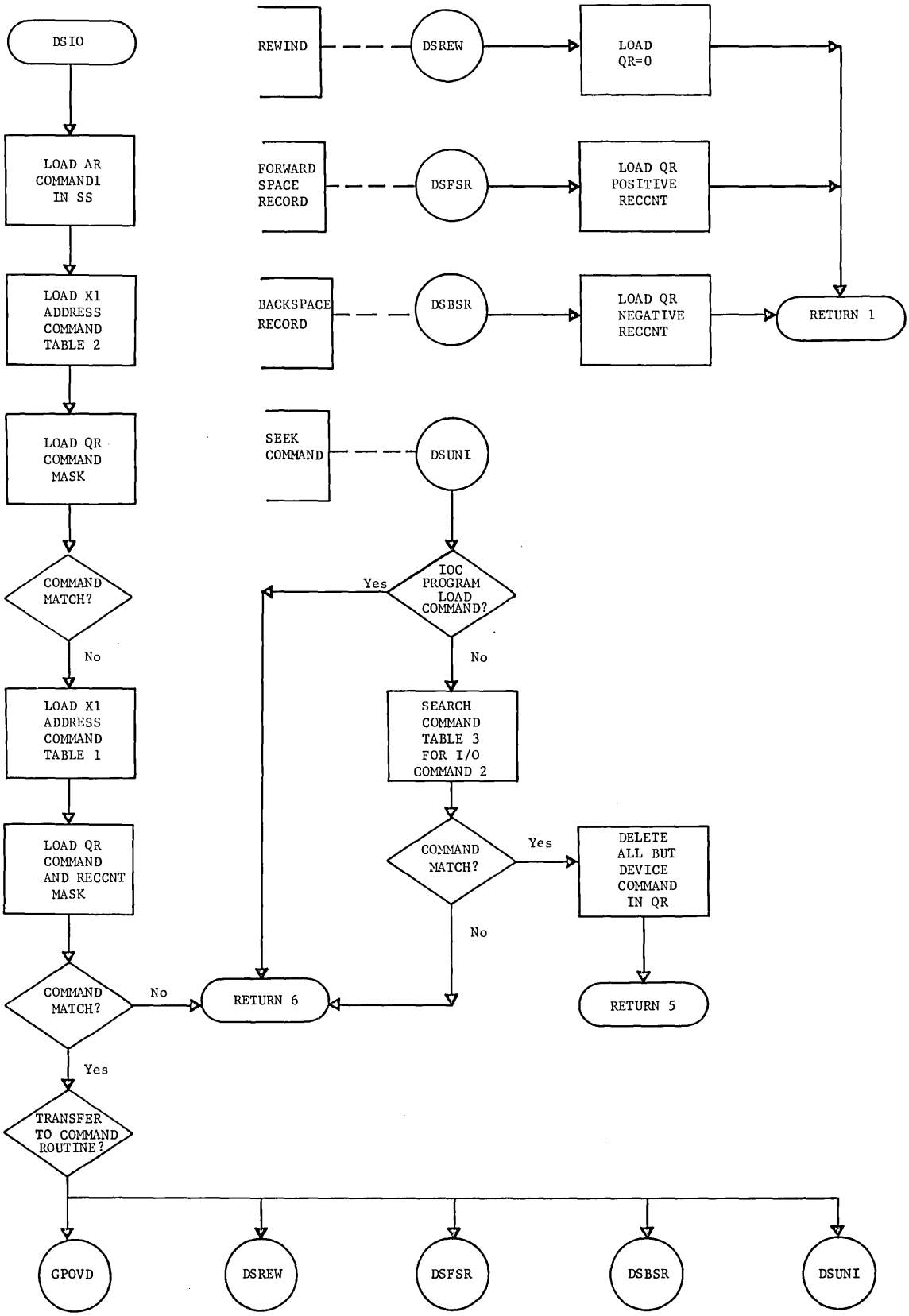
DSU200 INTERRUPT HANDLER




DSIT (EP1)
.MDR20
2
GEPR OVERRIDE ?
No
Yes
SYSTEM I/O ENTRY?
Yes
STOP CHANNEL SCT
No
4
RETURN 0
3
Yes
GEPR OVERRIDE?
No
RETURN 1
6
Yes
IOC ERROR?
No
CHANNEL READY?
No
7
Yes
THRESHOLD STATUS?
No
Yes
GEPR OVERRIDE?
Yes
.SHUTS .CRQGT
No
RETURN 6
SET SEEK COMPLETED FLAG, LINKED STATUS
RESET SEEK ADDRESS FLAG, IN IOS STATUS
DECREMENT OUTSTANDING I/O COUNT
.OPENS .CRQGT
RETURN 4
5
DCWS GENERATED?
Yes
ZERO DCWS
No
8
GEPR OVERRIDE?
Yes
No
THRESHOLD STATUS?
No
RETURN 0
Yes
RETURN 8

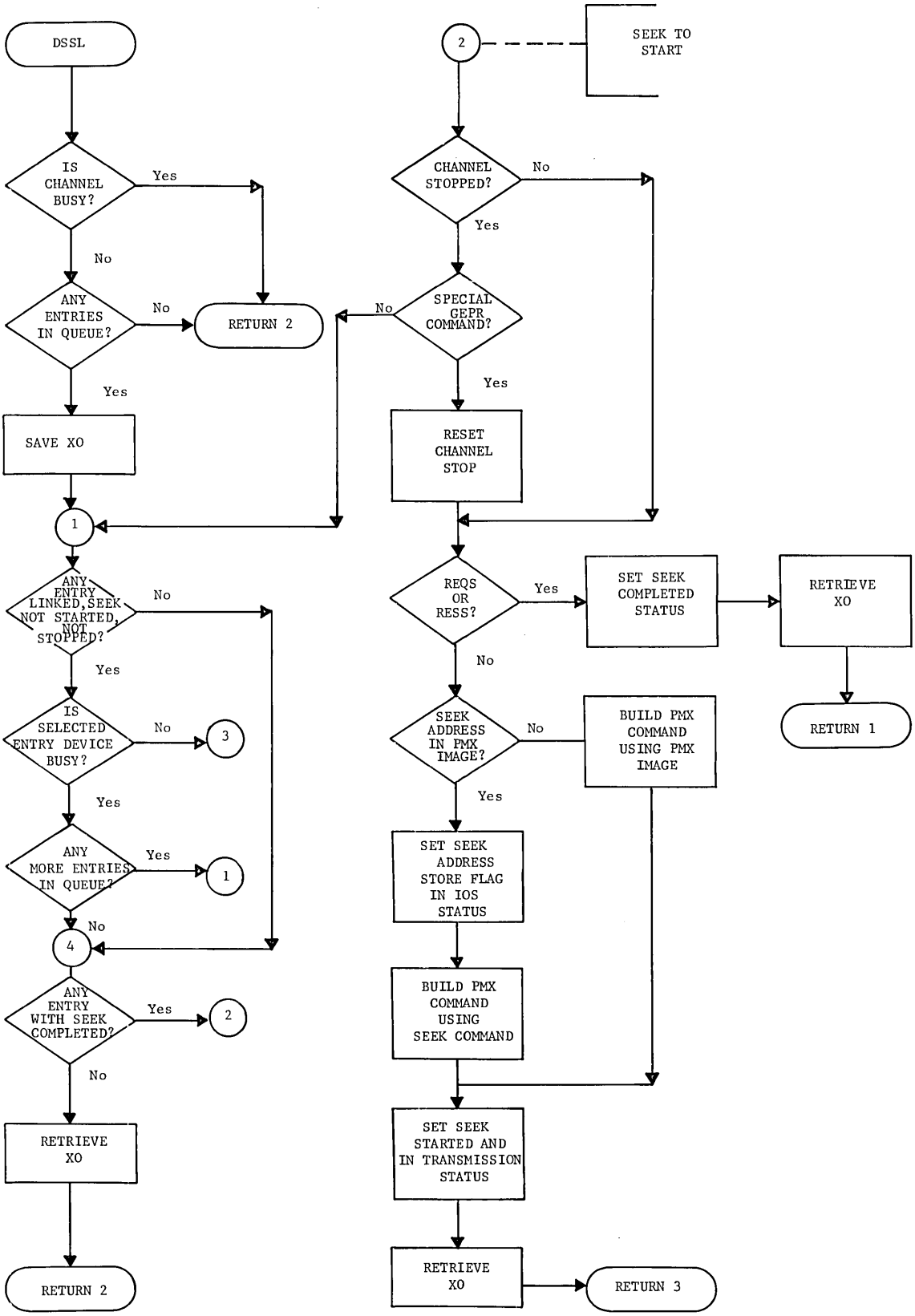
DSIO (EP2)
.MDS20

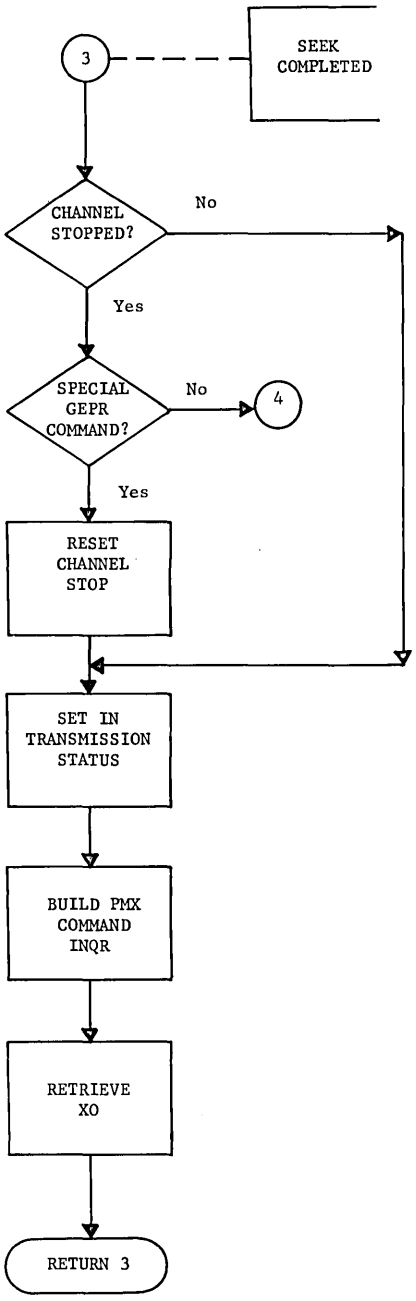
DSU200 REQUEST



DSSL (EP3)
.MDS20

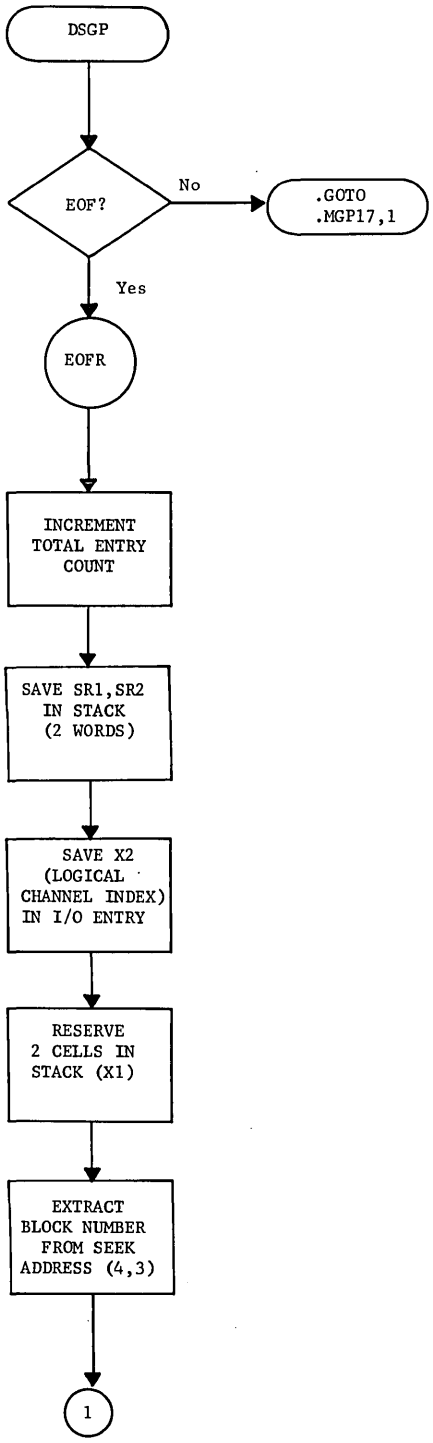
DSU200 SELECT

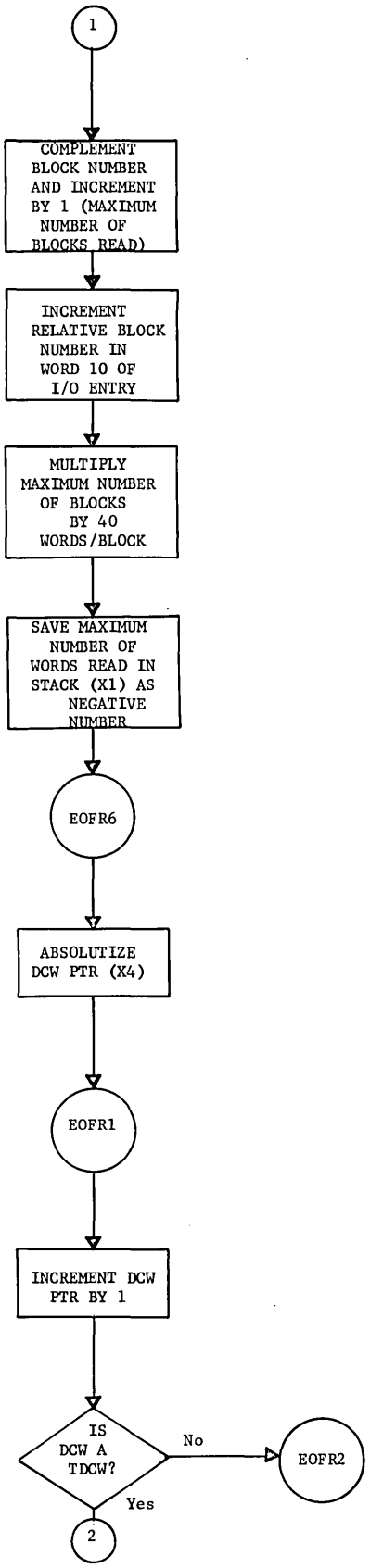


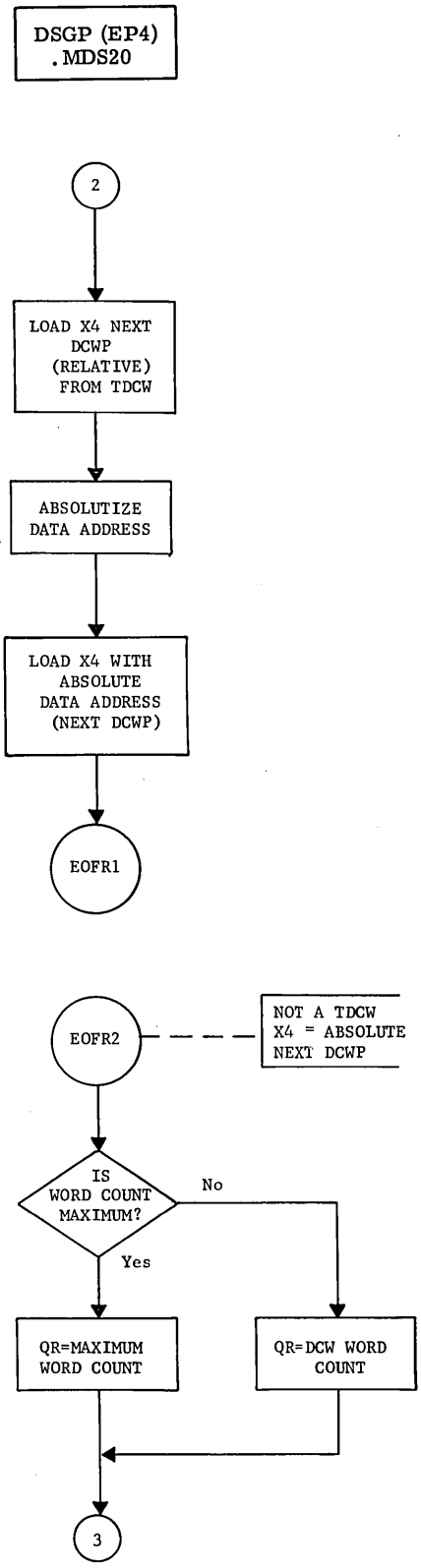

3
SEEK COMPLETED
CHANNEL STOPPED?
No
Yes
SPECIAL GEPR COMMAND?
No
4
Yes
RESET CHANNEL STOP
SET IN TRANSMISSION STATUS
BUILD PMX COMMAND INQR
RETRIEVE XO
RETURN 3

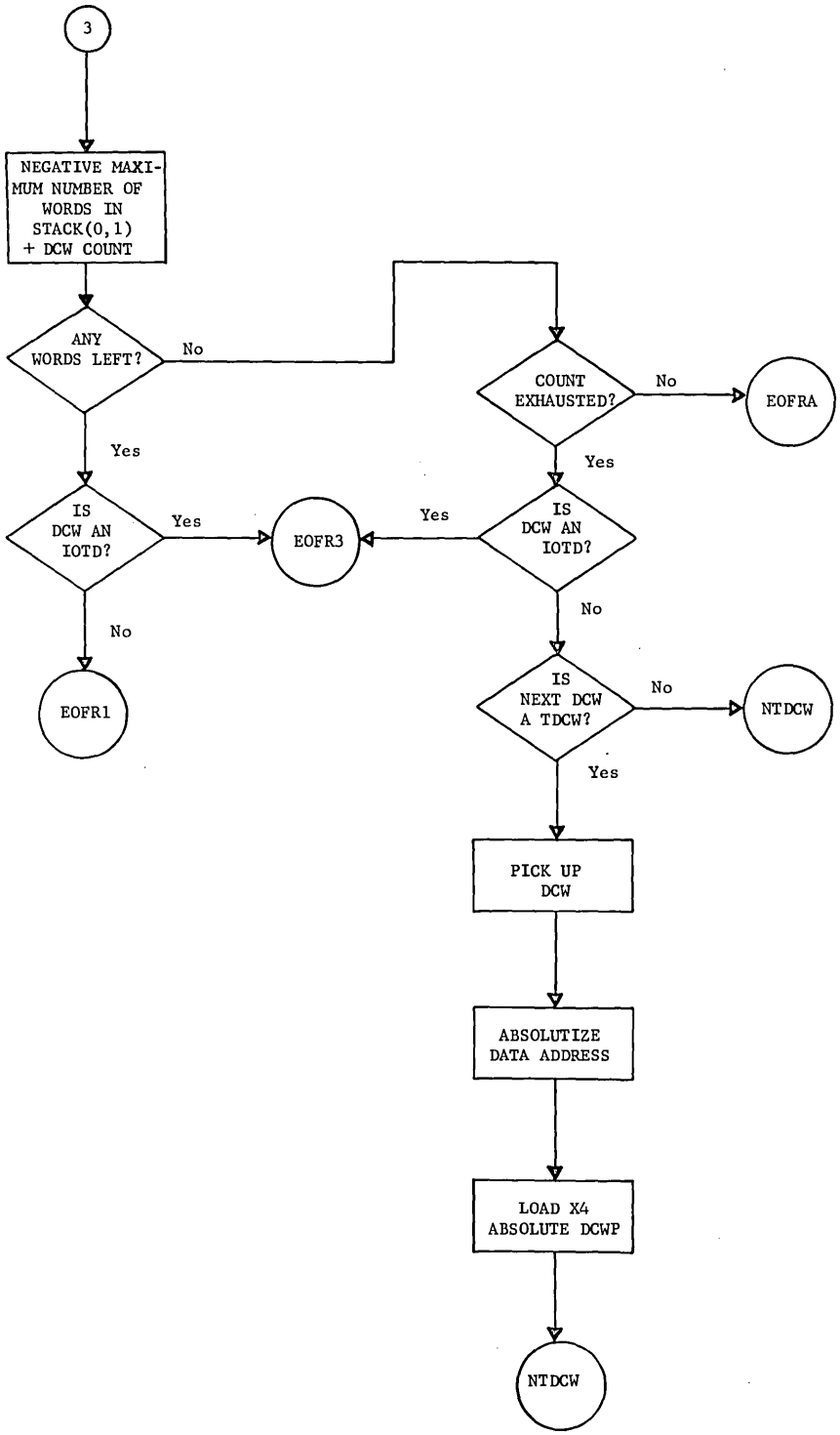
DSGP (EP4)
.MDS20

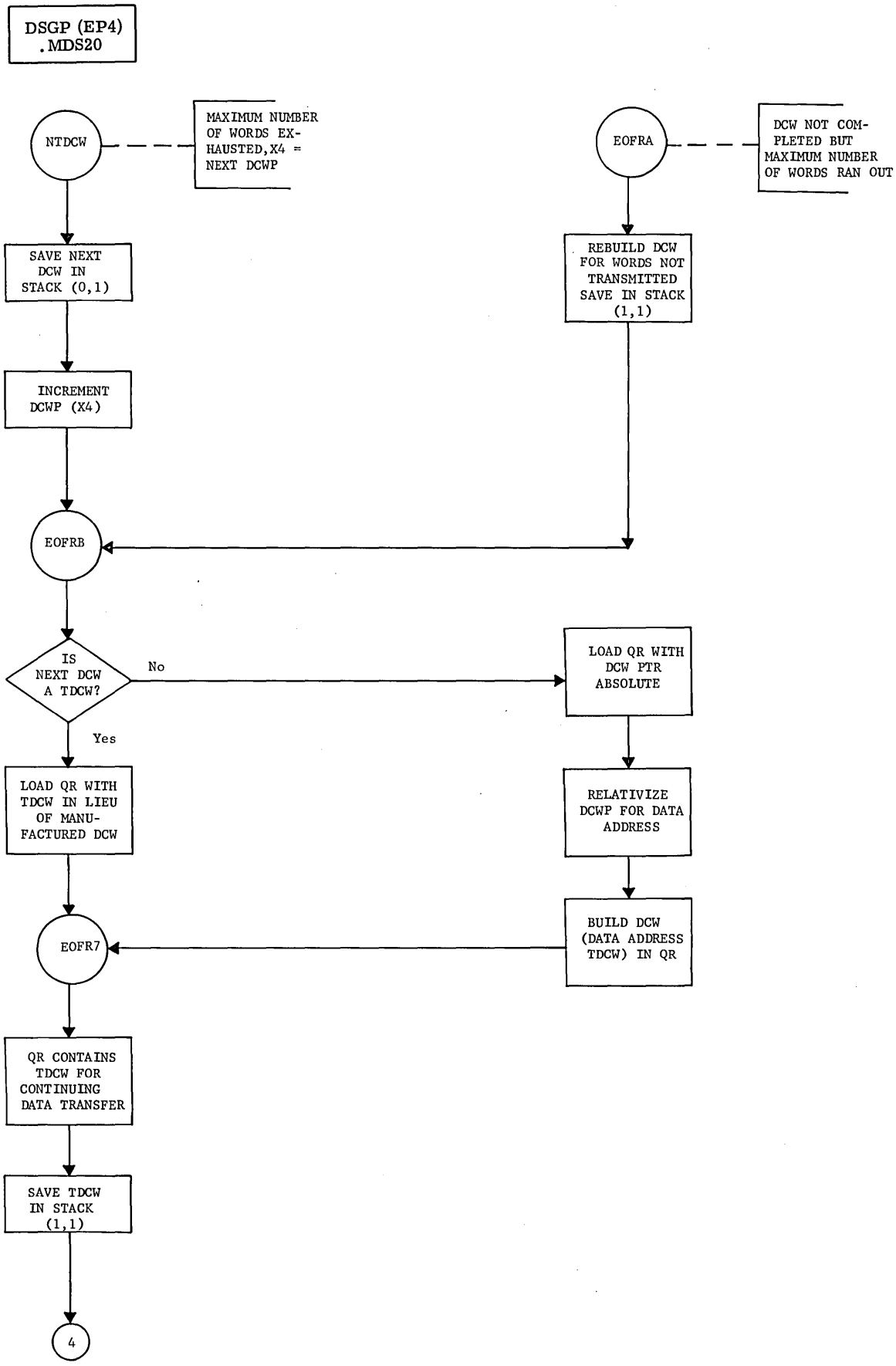
DSU200 ERROR AND EOF RECOVERY

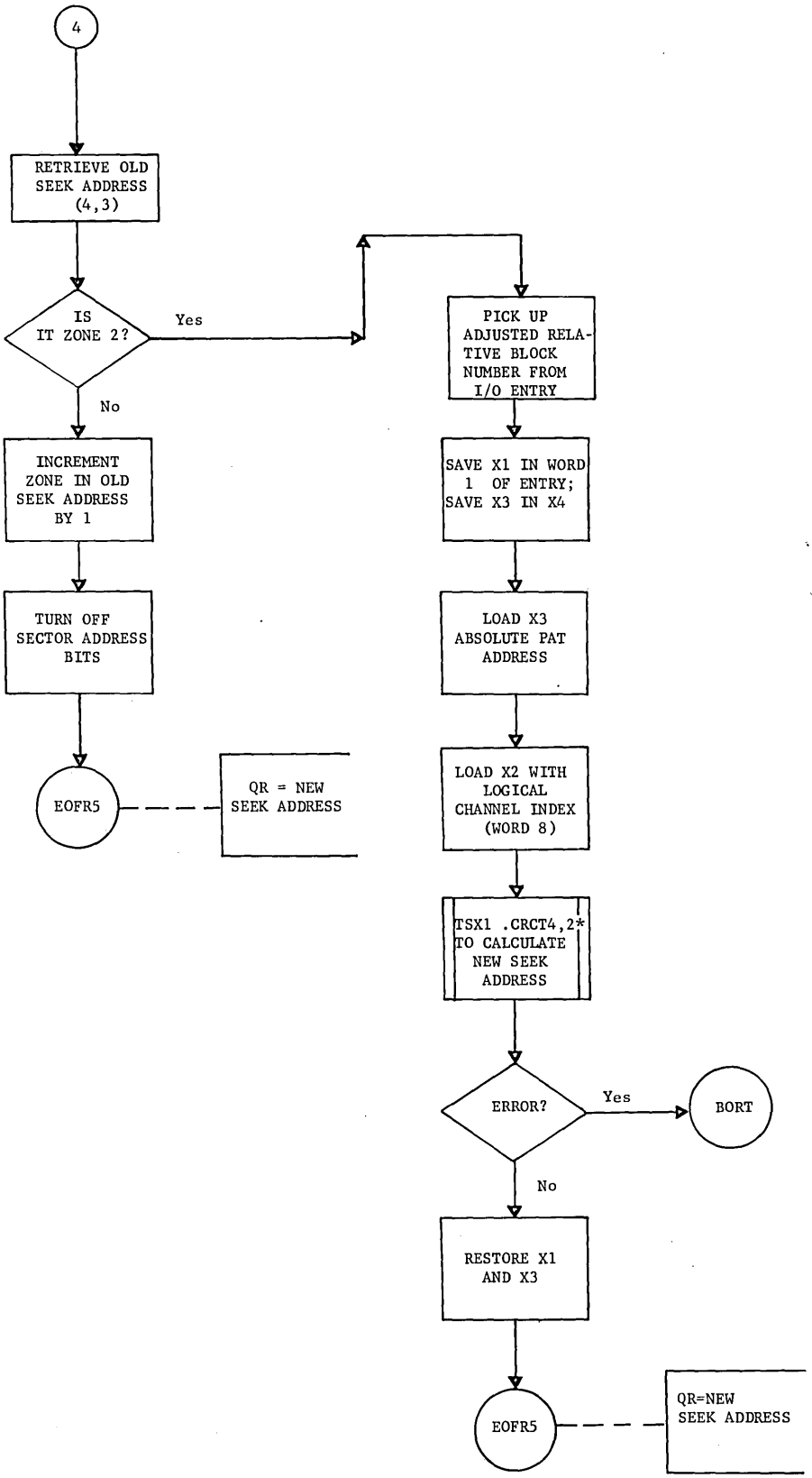


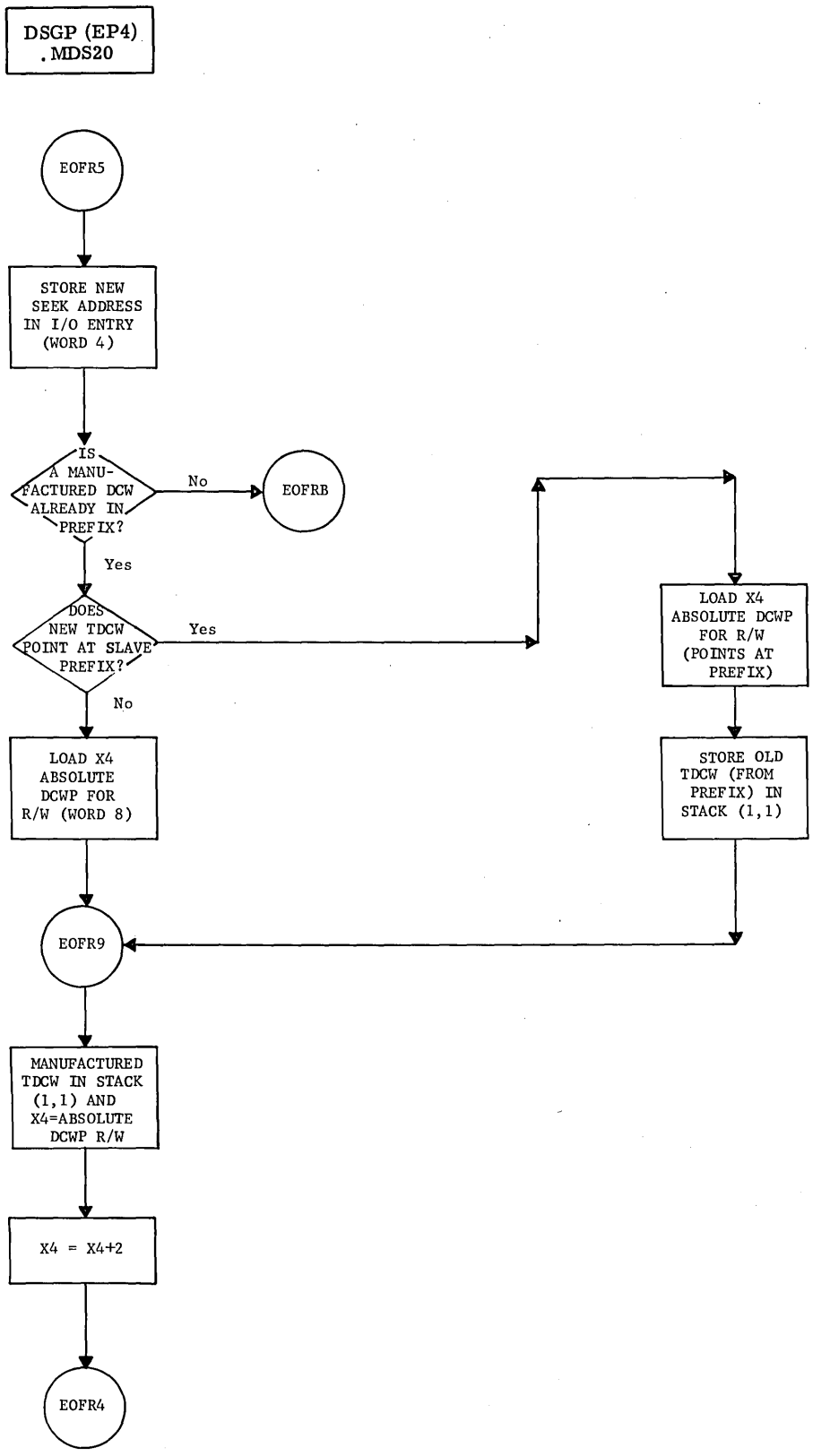

1
COMPLEMENT BLOCK NUMBER AND INCREMENT BY 1 (MAXIMUM NUMBER OF BLOCKS READ)
INCREMENT RELATIVE BLOCK NUMBER IN WORD 10 OF I/O ENTRY
MULTIPLY MAXIMUM NUMBER OF BLOCKS BY 40 WORDS/BLOCK
SAVE MAXIMUM NUMBER OF WORDS READ IN STACK (X1) AS NEGATIVE NUMBER
EOFR6
ABSOLUTIZE DCW PTR (X4)
EOFR1
INCREMENT DCW PTR BY 1
IS DCW A TDCW?
No
EOFR2
Yes
2

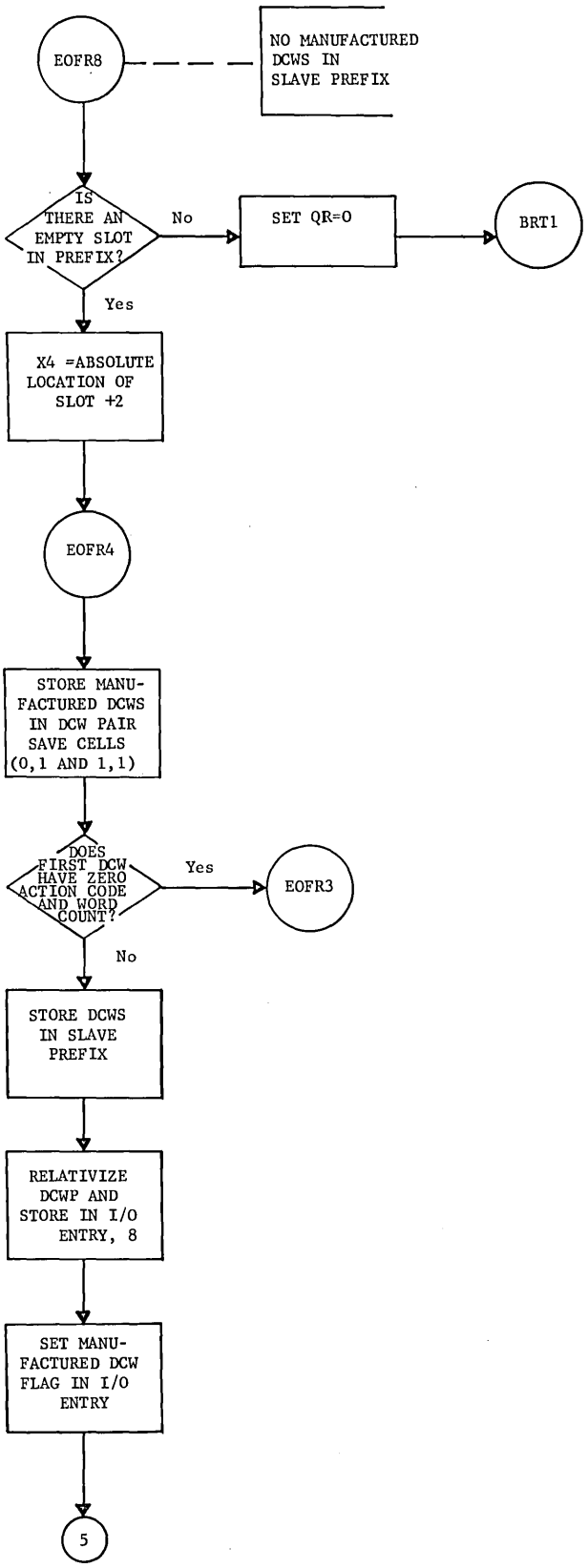

DSGP (EP4)
.MDS20
2
LOAD X4 NEXT
DCWP
(RELATIVE)
FROM TDCW
ABSOLUTIZE
DATA ADDRESS
LOAD X4 WITH
ABSOLUTE
DATA ADDRESS
(NEXT DCWP)
EOFR1
EOFR2
NOT A TDCW
X4 = ABSOLUTE
NEXT DCWP
IS
WORD COUNT
MAXIMUM?
No
Yes
QR=MAXIMUM
WORD COUNT
QR=DCW WORD
COUNT
3


3
NEGATIVE MAXI-MUM NUMBER OF WORDS IN STACK(0,1) + DCW COUNT
ANY WORDS LEFT?
No
Yes
COUNT EXHAUSTED?
No
EOFRA
Yes
IS DCW AN IOTD?
Yes
EOFR3
Yes
IS DCW AN IOTD?
No
EOFR1
No
IS NEXT DCW A TDCW?
No
NTDCW
Yes
PICK UP DCW
ABSOLUTIZE DATA ADDRESS
LOAD X4 ABSOLUTE DCWP
NTDCW

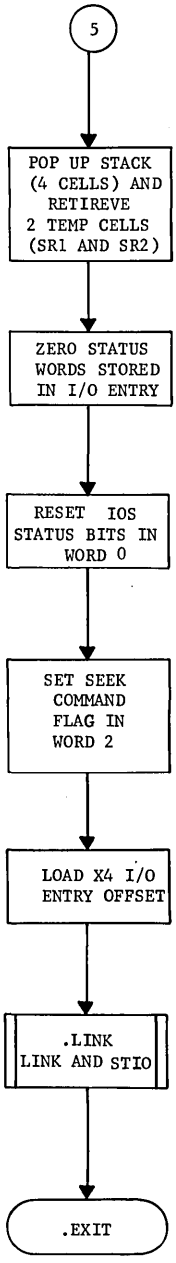

DSGP (EP4)
.MDS20
NTDCW
MAXIMUM NUMBER OF WORDS EXHAUSTED, X4 = NEXT DCWP
EOFRA
DCW NOT COMPLETED BUT MAXIMUM NUMBER OF WORDS RAN OUT
SAVE NEXT DCW IN STACK (0,1)
REBUILD DCW FOR WORDS NOT TRANSMITTED SAVE IN STACK (1,1)
INCREMENT DCWP (X4)
EOFRB
IS NEXT DCW A TDCW?
No
LOAD QR WITH DCW PTR ABSOLUTE
Yes
LOAD QR WITH TDCW IN LIEU OF MANUFACTURED DCW
RELATIVIZE DCWP FOR DATA ADDRESS
EOFR7
BUILD DCW (DATA ADDRESS TDCW) IN QR
QR CONTAINS TDCW FOR CONTINUING DATA TRANSFER
SAVE TDCW IN STACK (1,1)
4

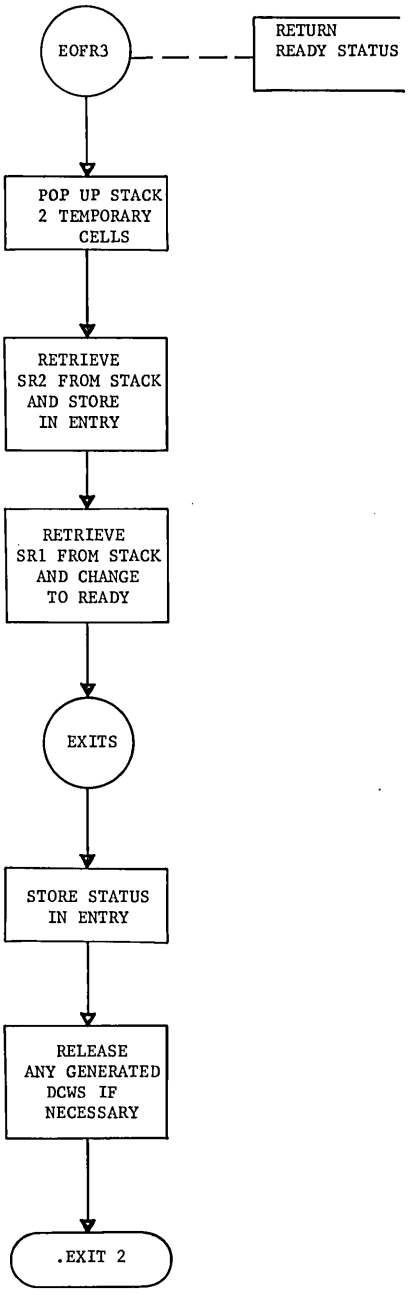

4
RETRIEVE OLD SEEK ADDRESS (4,3)
IS IT ZONE 2?
Yes
No
INCREMENT ZONE IN OLD SEEK ADDRESS BY 1
TURN OFF SECTOR ADDRESS BITS
EOFR5
QR = NEW SEEK ADDRESS
PICK UP ADJUSTED RELATIVE BLOCK NUMBER FROM I/O ENTRY
SAVE X1 IN WORD 1 OF ENTRY; SAVE X3 IN X4
LOAD X3 ABSOLUTE PAT ADDRESS
LOAD X2 WITH LOGICAL CHANNEL INDEX (WORD 8)
TSX1 .CRCT4,2* TO CALCULATE NEW SEEK ADDRESS
ERROR?
Yes
BORT
No
RESTORE X1 AND X3
EOFR5
QR=NEW SEEK ADDRESS


DSGP (EP4)
.MDS20
EOFR5
STORE NEW SEEK ADDRESS IN I/O ENTRY (WORD 4)
IS A MANU-FACTURED DCW ALREADY IN PREFIX?
No
EOFRB
Yes
DOES NEW TDCW POINT AT SLAVE PREFIX?
Yes
No
LOAD X4 ABSOLUTE DCWP FOR R/W (POINTS AT PREFIX)
STORE OLD TDCW (FROM PREFIX) IN STACK (1,1)
LOAD X4 ABSOLUTE DCWP FOR R/W (WORD 8)
EOFR9
MANUFACTURED TDCW IN STACK (1,1) AND X4=ABSOLUTE DCWP R/W
X4 = X4+2
EOFR4

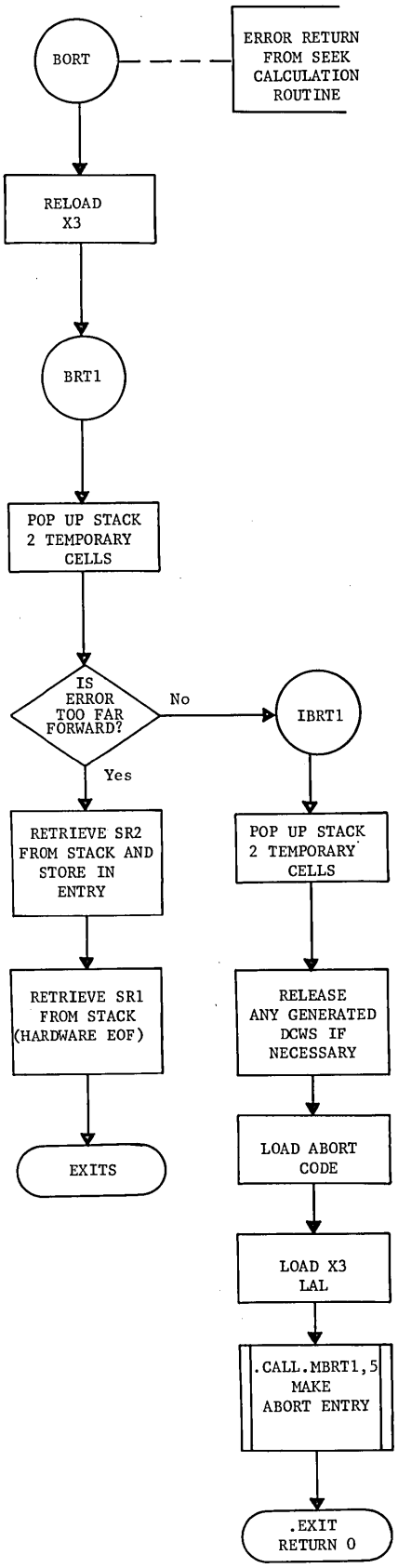

EOFR8
NO MANUFACTURED DCWS IN SLAVE PREFIX
IS THERE AN EMPTY SLOT IN PREFIX?
No
SET QR=0
BRT1
Yes
X4 =ABSOLUTE LOCATION OF SLOT +2
EOFR4
STORE MANU-FACTURED DCWS IN DCW PAIR SAVE CELLS (0,1 AND 1,1)
DOES FIRST DCW HAVE ZERO ACTION CODE AND WORD COUNT?
Yes
EOFR3
No
STORE DCWS IN SLAVE PREFIX
RELATIVIZE DCWP AND STORE IN I/O ENTRY, 8
SET MANU-FACTURED DCW FLAG IN I/O ENTRY
5

DSGP (EP4)
.MDS20



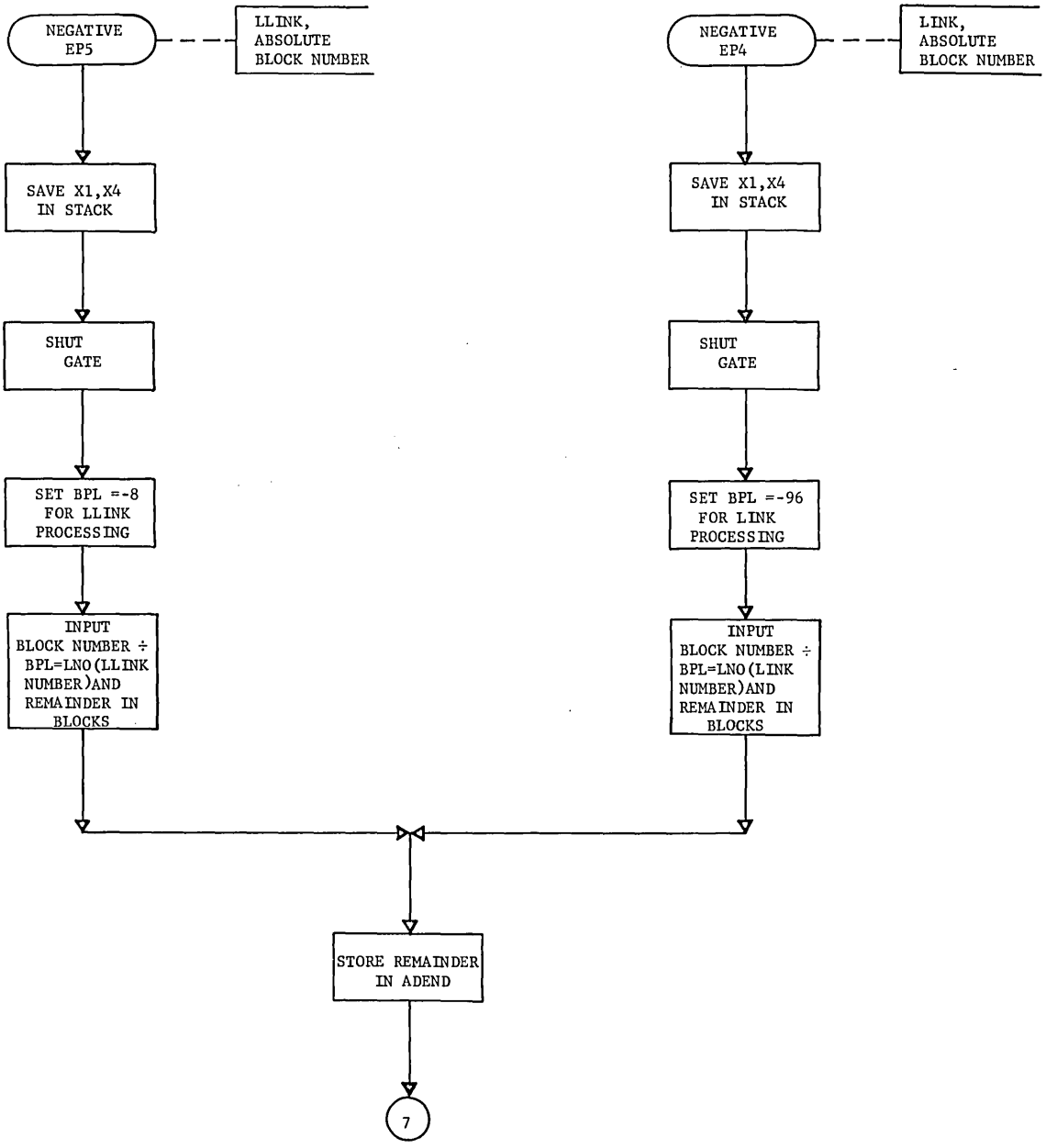

EOFR3
RETURN
READY STATUS
POP UP STACK
2 TEMPORARY
CELLS
RETRIEVE
SR2 FROM STACK
AND STORE
IN ENTRY
RETRIEVE
SR1 FROM STACK
AND CHANGE
TO READY
EXITS
STORE STATUS
IN ENTRY
RELEASE
ANY GENERATED
DCWS IF
NECESSARY
.EXIT 2

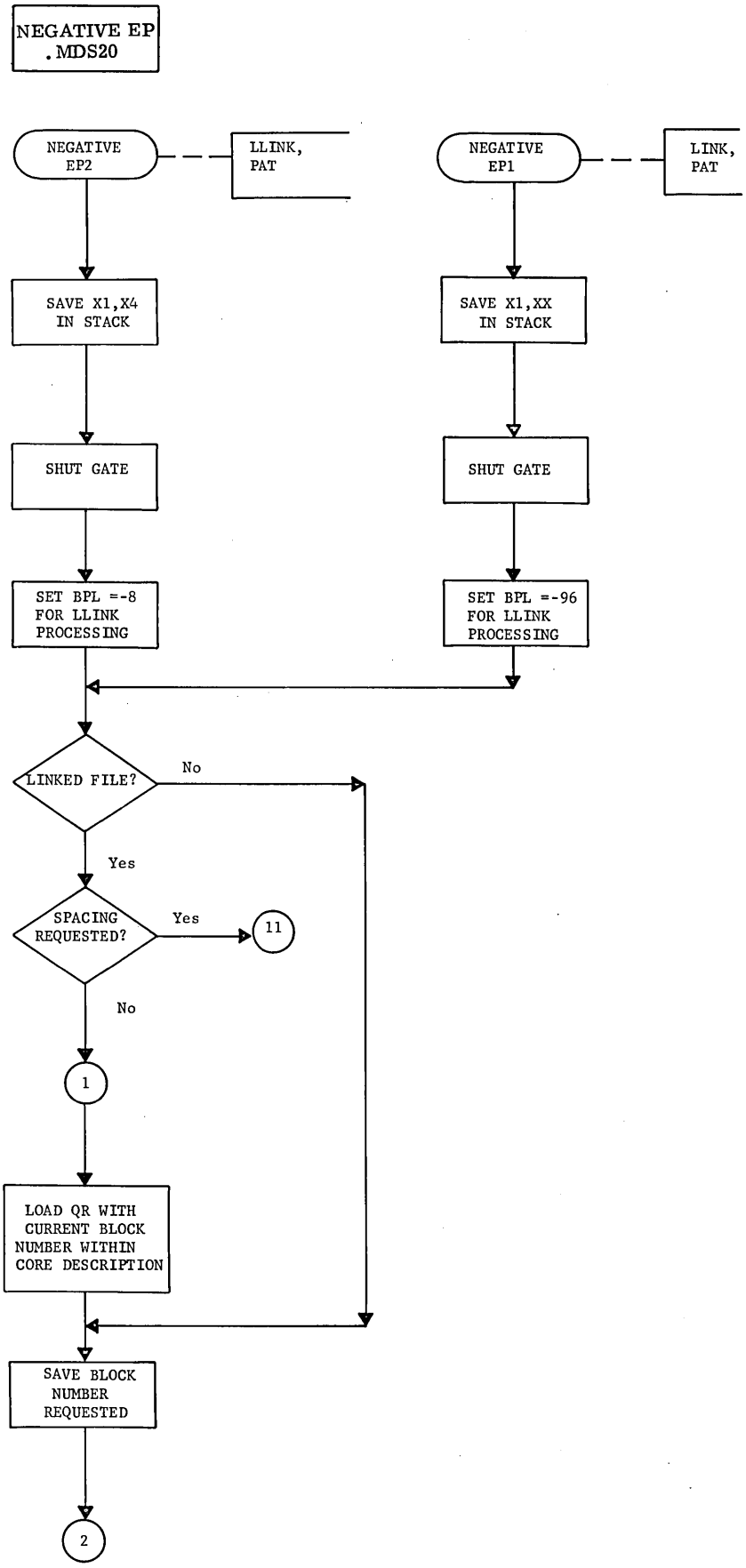
DSGP (EP4)
.MDS20

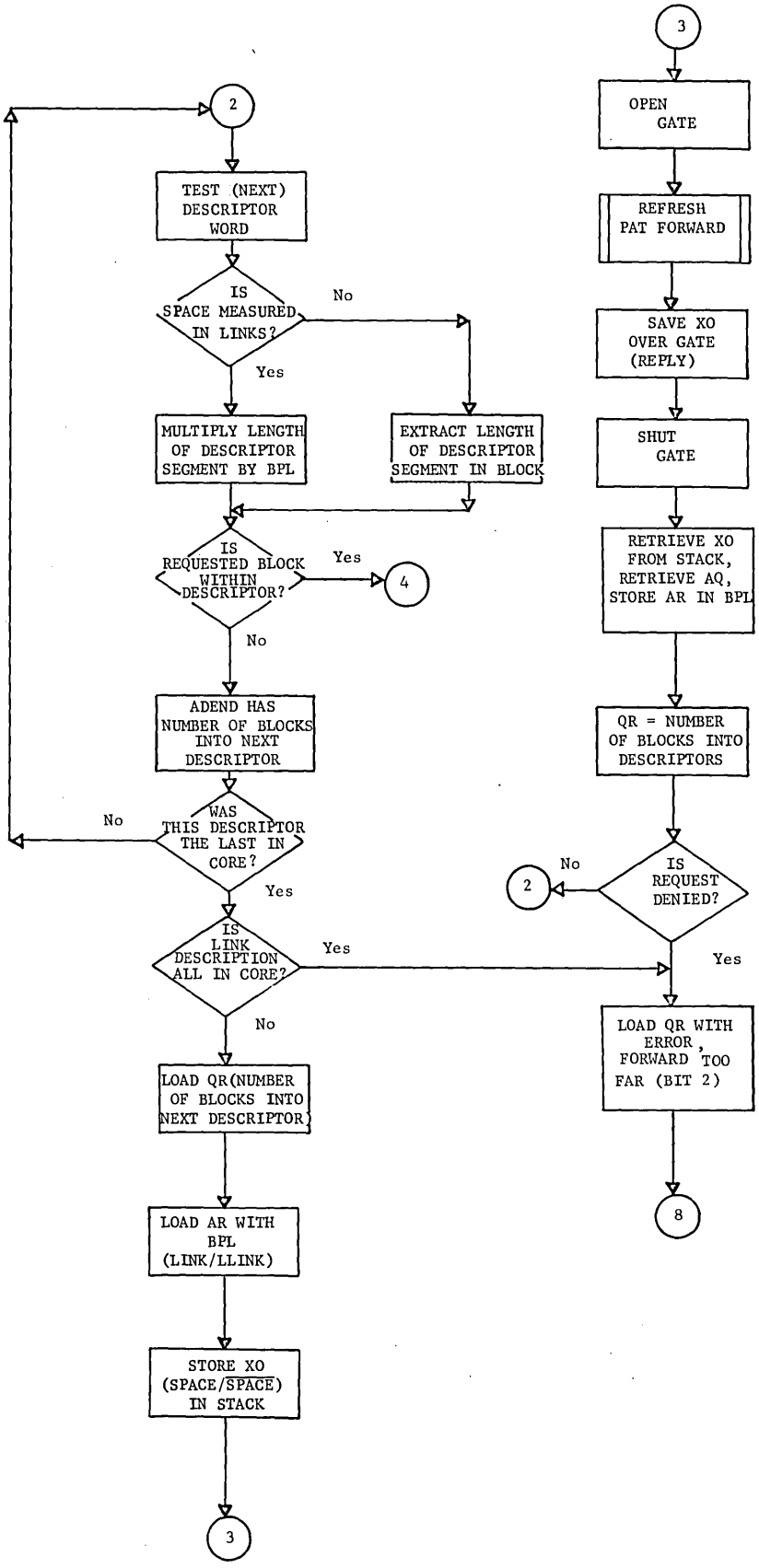


NEGATIVE EP
.MDS20

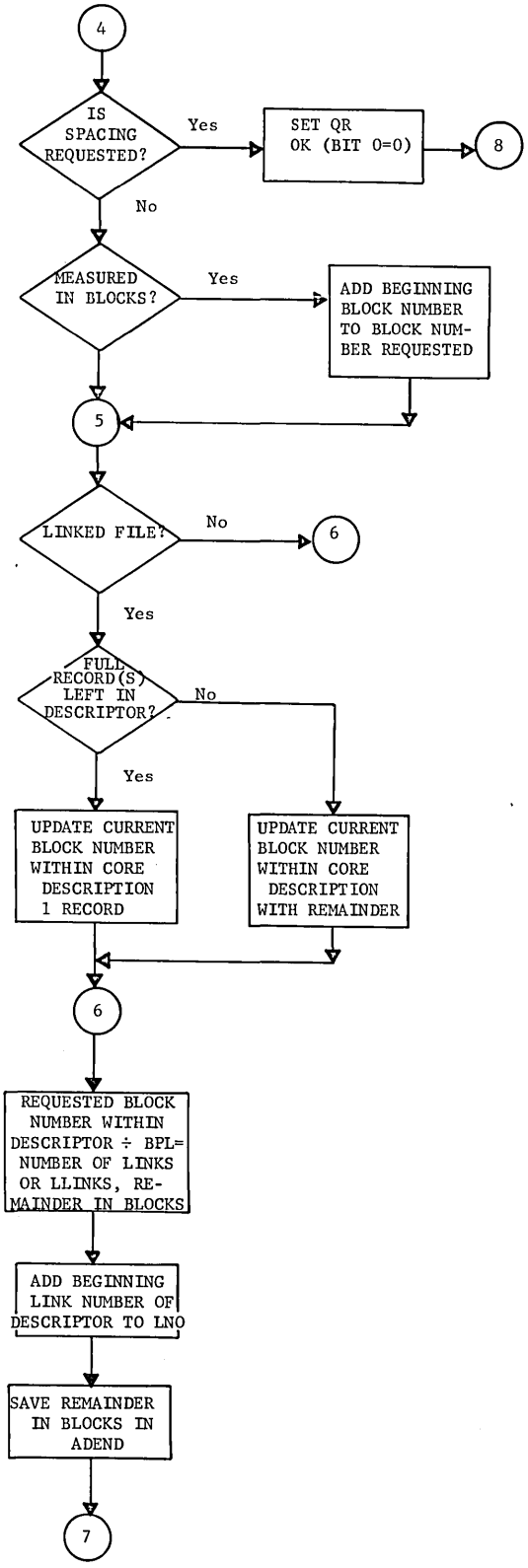
DSU200 NEGATIVE ENTRY POINTS

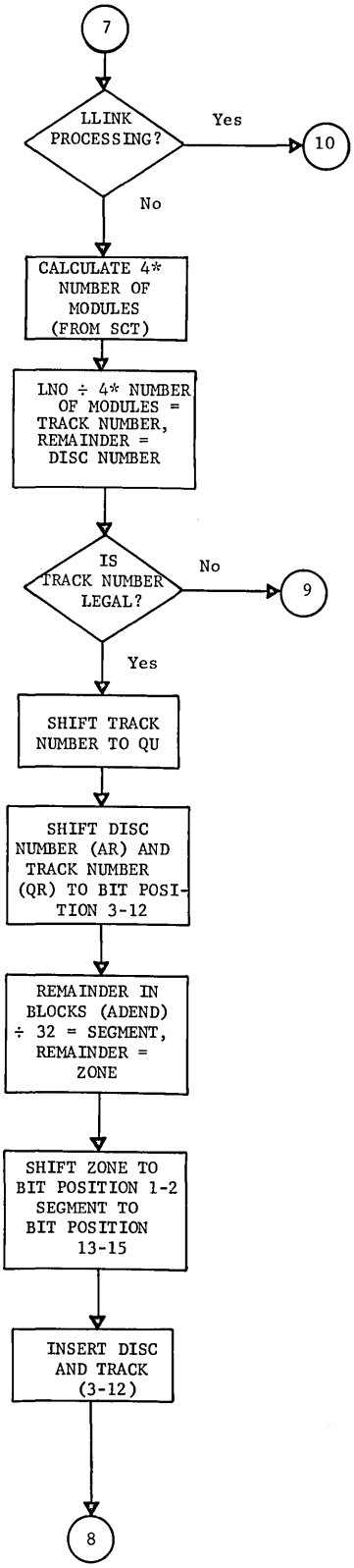


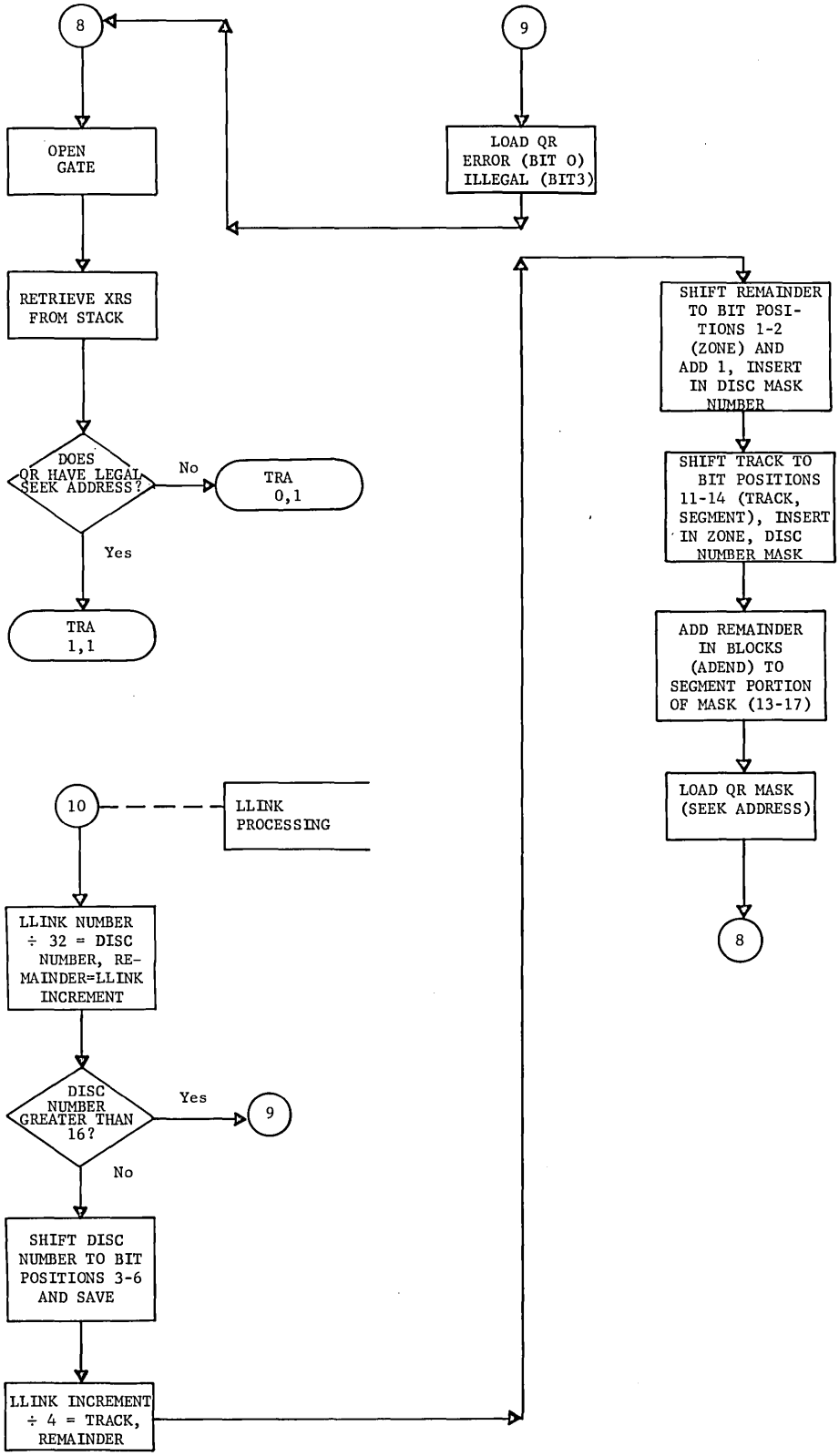

NEGATIVE EP
.MDS20
NEGATIVE
EP2
LLINK,
PAT
NEGATIVE
EP1
LINK,
PAT
SAVE X1,X4
IN STACK
SAVE X1,XX
IN STACK
SHUT GATE
SHUT GATE
SET BPL =-8
FOR LLINK
PROCESSING
SET BPL =-96
FOR LLINK
PROCESSING
LINKED FILE?
No
Yes
SPACING
REQUESTED?
Yes
11
No
1
LOAD QR WITH
CURRENT BLOCK
NUMBER WITHIN
CORE DESCRIPTION
SAVE BLOCK
NUMBER
REQUESTED
2

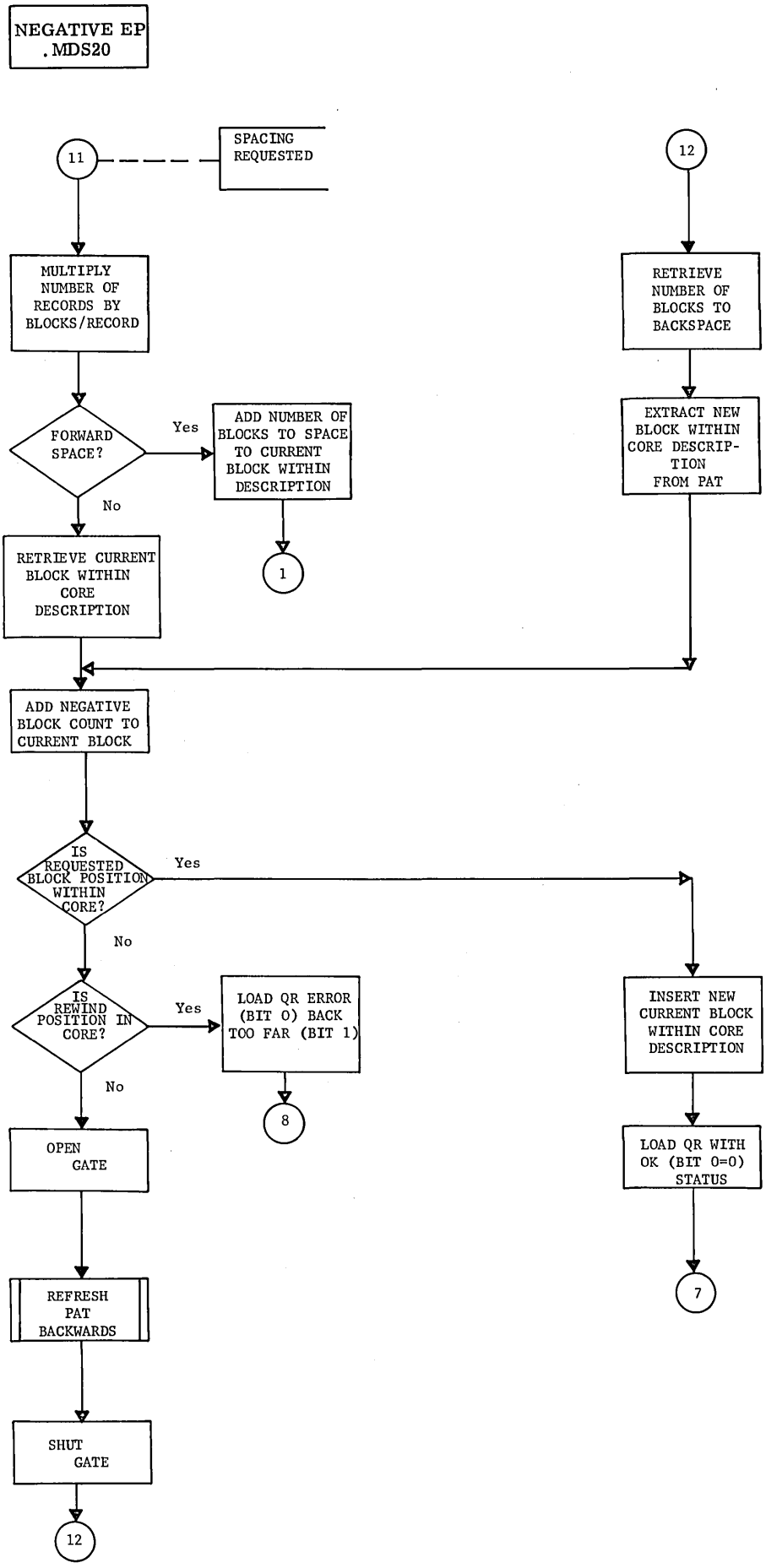

2
TEST (NEXT) DESCRIPTOR WORD
IS SPACE MEASURED IN LINKS?
No
Yes
MULTIPLY LENGTH OF DESCRIPTOR SEGMENT BY BPL
EXTRACT LENGTH OF DESCRIPTOR SEGMENT IN BLOCK
IS REQUESTED BLOCK WITHIN DESCRIPTOR?
Yes
4
No
ADEND HAS NUMBER OF BLOCKS INTO NEXT DESCRIPTOR
WAS THIS DESCRIPTOR THE LAST IN CORE?
No
Yes
IS LINK DESCRIPTION ALL IN CORE?
Yes
No
LOAD QR(NUMBER OF BLOCKS INTO NEXT DESCRIPTOR)
LOAD AR WITH BPL (LINK/LLINK)
STORE XO (SPACE/SPACE) IN STACK
3
3
OPEN GATE
REFRESH PAT FORWARD
SAVE XO OVER GATE (REPLY)
SHUT GATE
RETRIEVE XO FROM STACK, RETRIEVE AQ, STORE AR IN BPL
QR = NUMBER OF BLOCKS INTO DESCRIPTORS
IS REQUEST DENIED?
No
2
Yes
LOAD QR WITH ERROR, FORWARD TOO FAR (BIT 2)
8

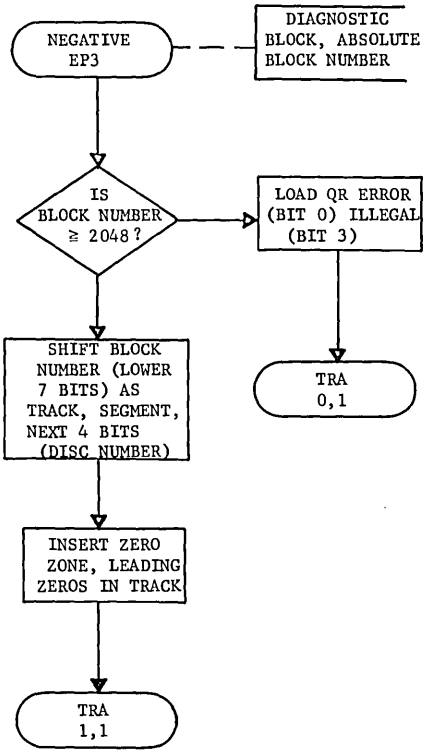
NEGATIVE EP
.MDS20





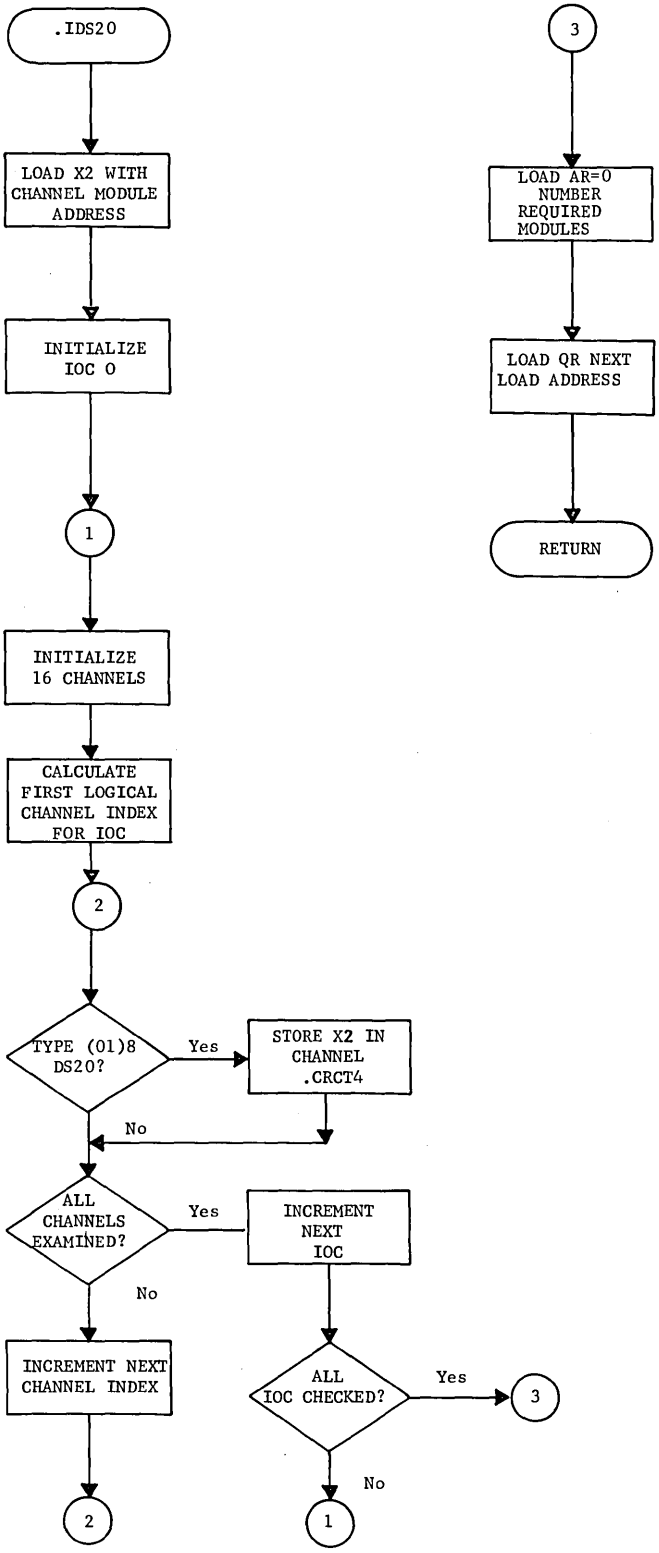

8
9
OPEN GATE
LOAD QR ERROR (BIT 0) ILLEGAL (BIT3)
RETRIEVE XRS FROM STACK
SHIFT REMAINDER TO BIT POSITIONS 1-2 (ZONE) AND ADD 1, INSERT IN DISC MASK NUMBER
DOES QR HAVE LEGAL SEEK ADDRESS?
No
TRA 0,1
Yes
TRA 1,1
SHIFT TRACK TO BIT POSITIONS 11-14 (TRACK, SEGMENT), INSERT IN ZONE, DISC NUMBER MASK
ADD REMAINDER IN BLOCKS (ADEND) TO SEGMENT PORTION OF MASK (13-17)
LOAD QR MASK (SEEK ADDRESS)
8
10
LLINK PROCESSING
LLINK NUMBER ÷ 32 = DISC NUMBER, REMAINDER=LLINK INCREMENT
DISC NUMBER GREATER THAN 16?
Yes
9
No
SHIFT DISC NUMBER TO BIT POSITIONS 3-6 AND SAVE
LLINK INCREMENT ÷ 4 = TRACK, REMAINDER


NEGATIVE EP
.MDS20
11
SPACING REQUESTED
MULTIPLY NUMBER OF RECORDS BY BLOCKS/RECORD
FORWARD SPACE?
Yes
ADD NUMBER OF BLOCKS TO SPACE TO CURRENT BLOCK WITHIN DESCRIPTION
1
No
RETRIEVE CURRENT BLOCK WITHIN CORE DESCRIPTION
12
RETRIEVE NUMBER OF BLOCKS TO BACKSPACE
EXTRACT NEW BLOCK WITHIN CORE DESCRIP-TION FROM PAT
ADD NEGATIVE BLOCK COUNT TO CURRENT BLOCK
IS REQUESTED BLOCK POSITION WITHIN CORE?
Yes
No
IS REWIND POSITION IN CORE?
Yes
LOAD QR ERROR (BIT 0) BACK TOO FAR (BIT 1)
8
No
OPEN GATE
REFRESH PAT BACKWARDS
SHUT GATE
12
INSERT NEW CURRENT BLOCK WITHIN CORE DESCRIPTION
LOAD QR WITH OK (BIT 0=0) STATUS
7


NEGATIVE EP3
DIAGNOSTIC BLOCK, ABSOLUTE BLOCK NUMBER
IS BLOCK NUMBER ≥ 2048?
LOAD QR ERROR (BIT 0) ILLEGAL (BIT 3)
TRA 0,1
SHIFT BLOCK NUMBER (LOWER 7 BITS) AS TRACK, SEGMENT, NEXT 4 BITS (DISC NUMBER)
INSERT ZERO ZONE, LEADING ZEROS IN TRACK
TRA 1,1

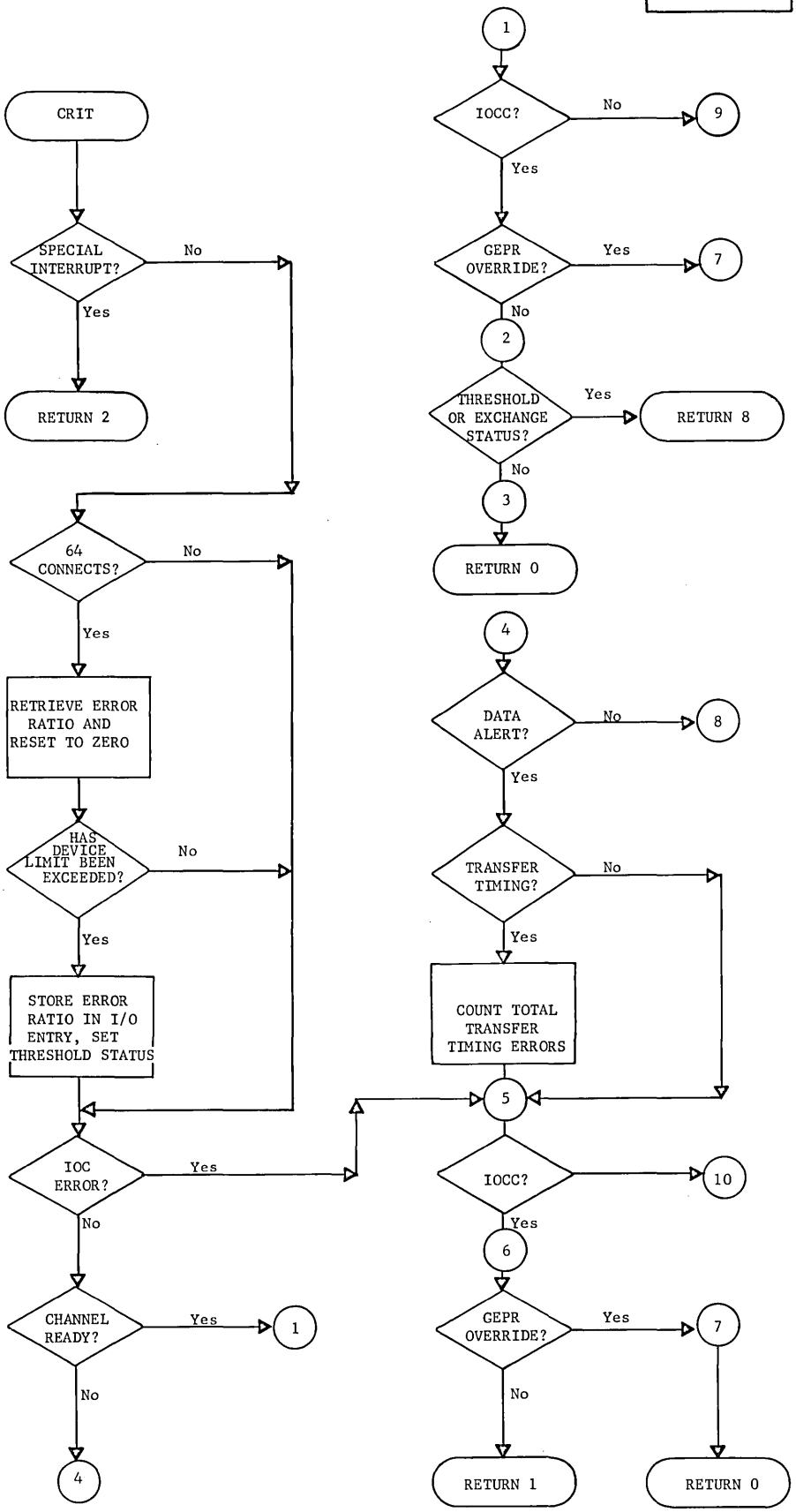
.IDS20
.MDS20

DSU200 INITIALIZATION

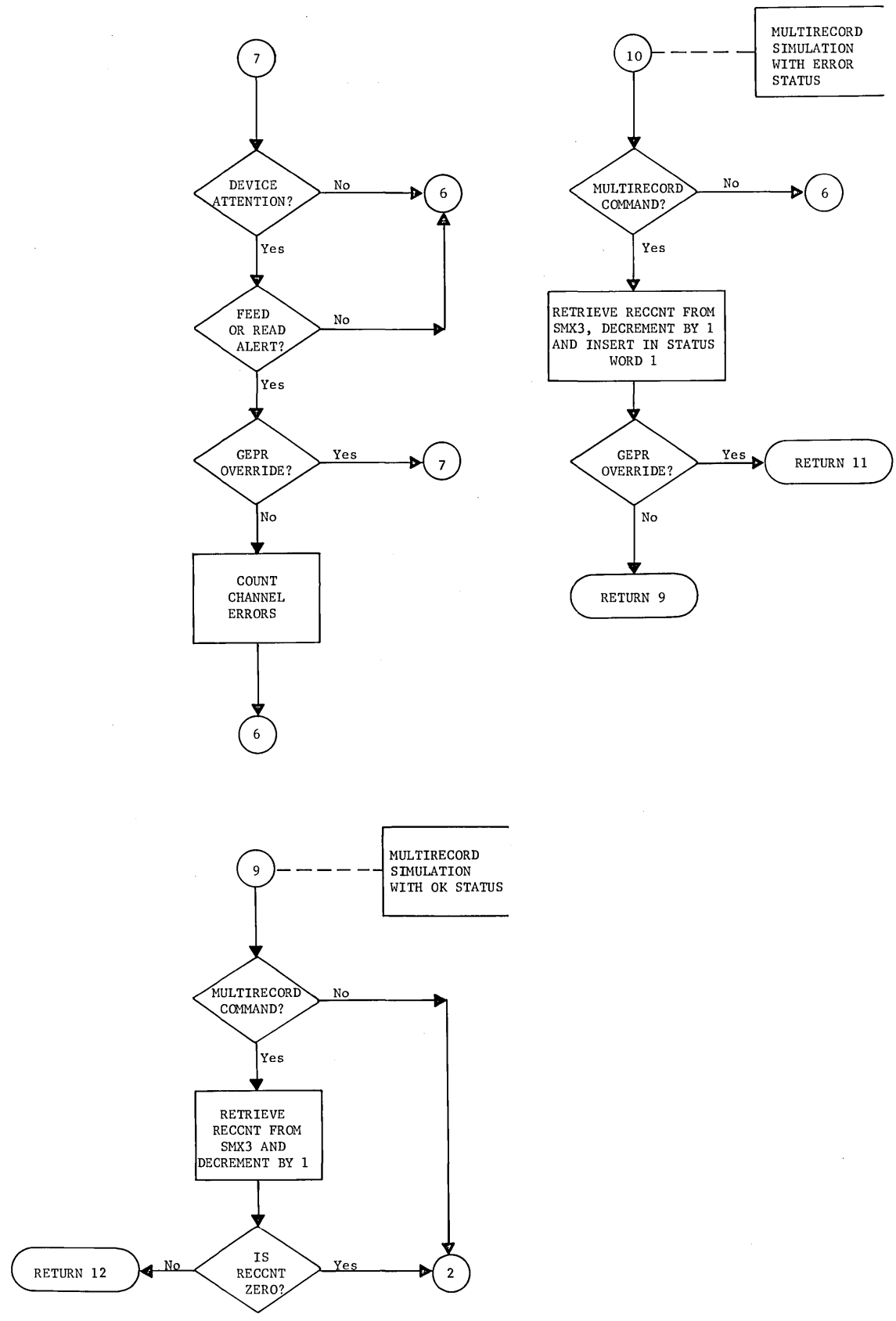


CARD READER INTERRUPT HANDLER

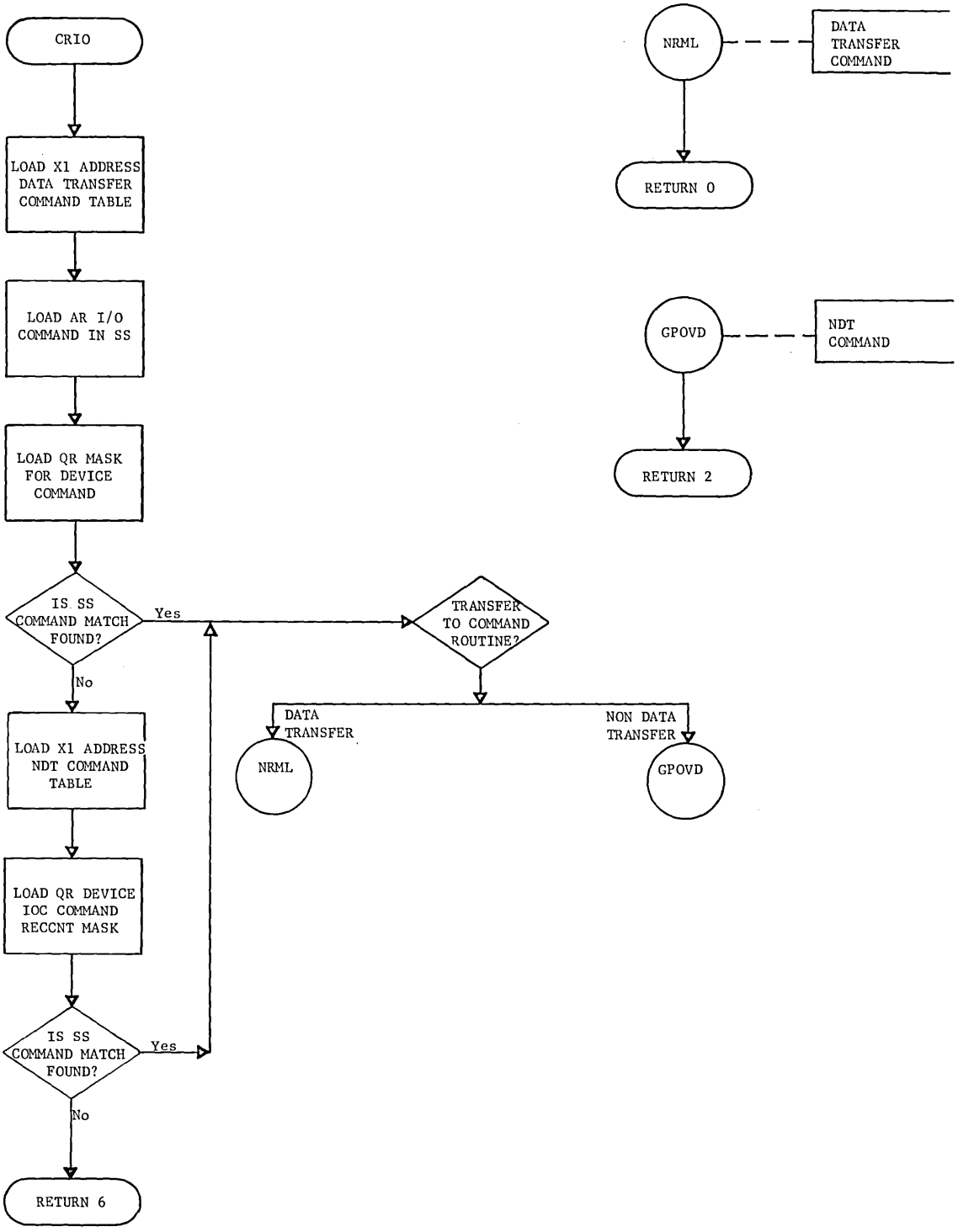
CRIT (EP1)
.MGPIO



CRIT (EP1)
.MGPIO

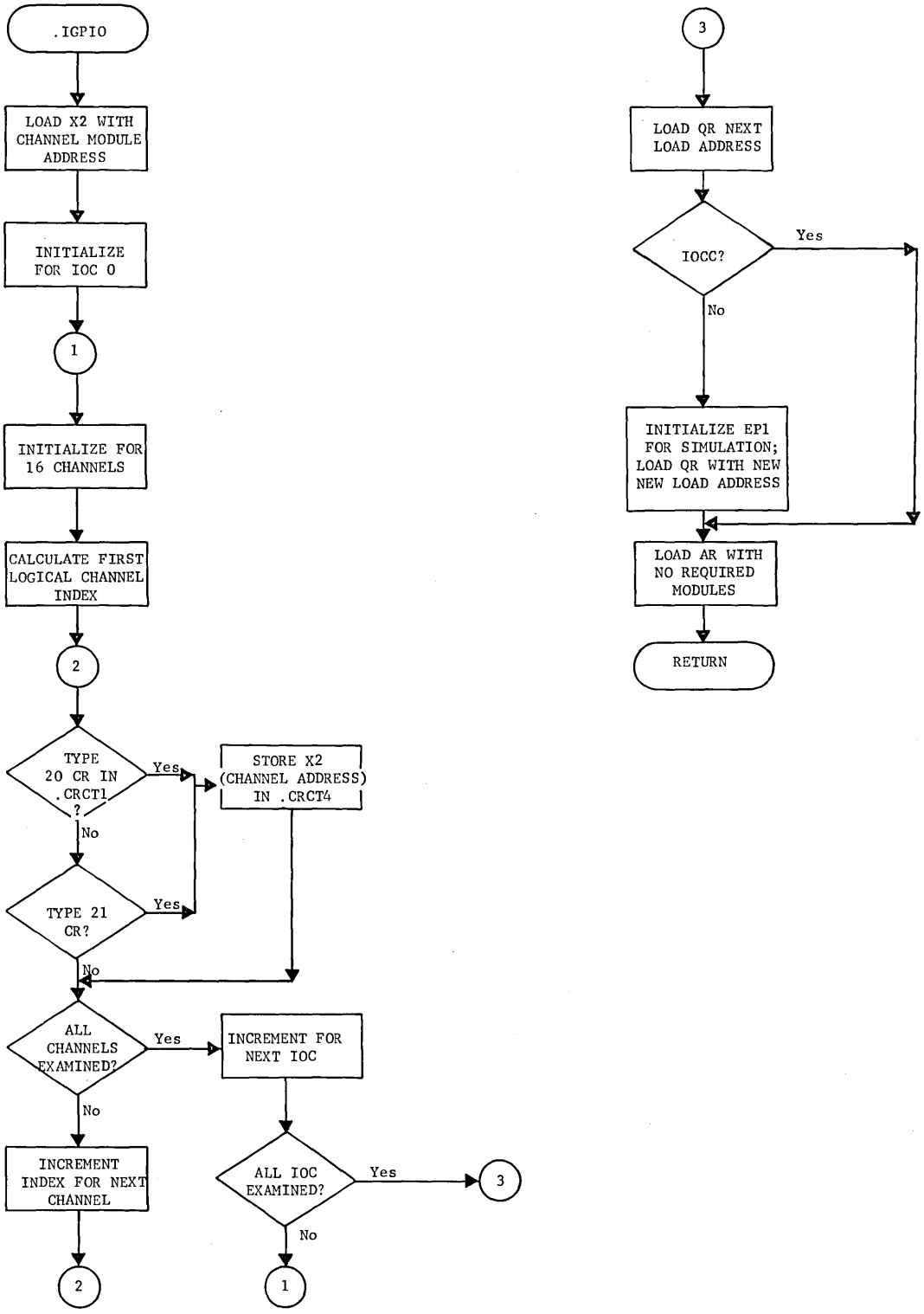


CARD READER REQUEST



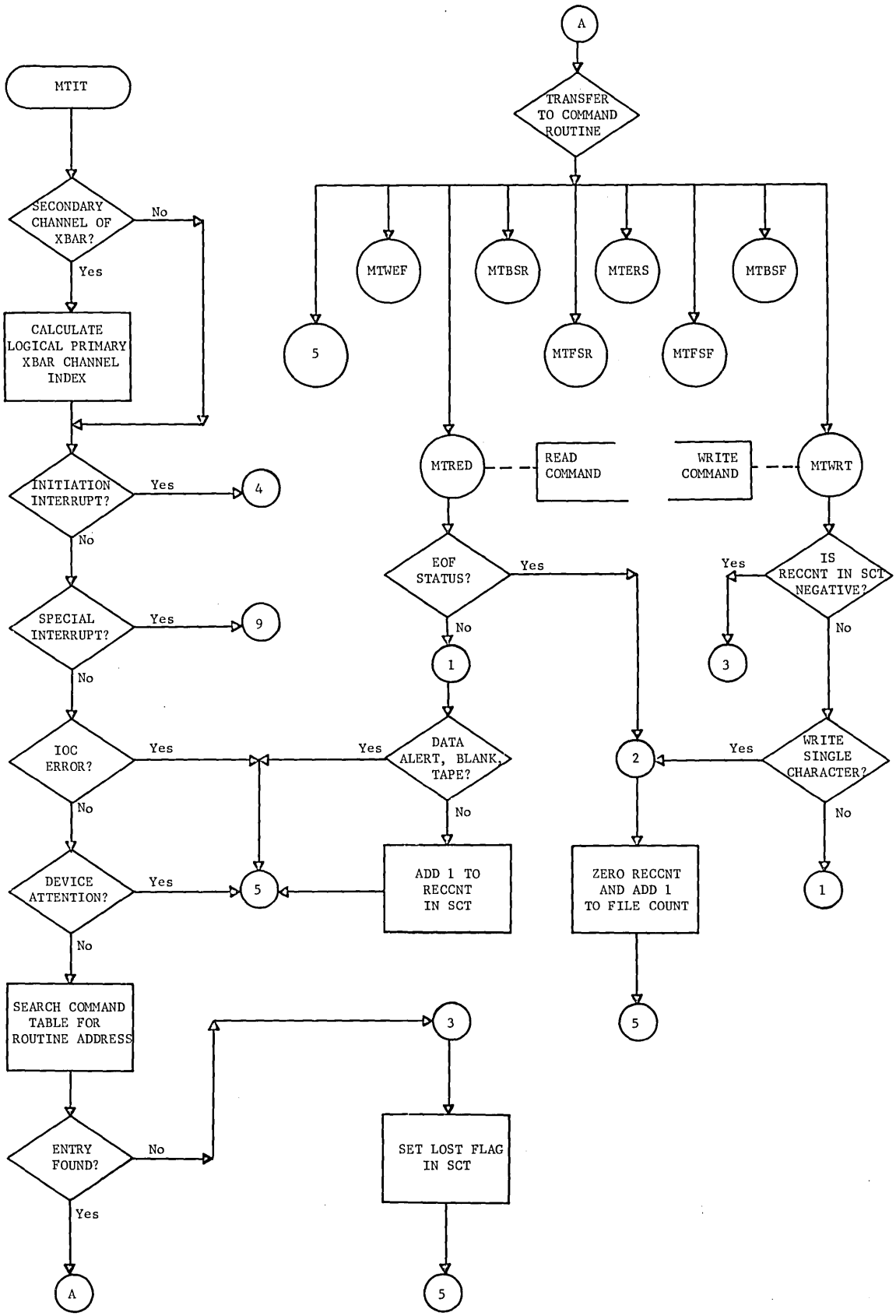
.IGPIO
.MGPIO

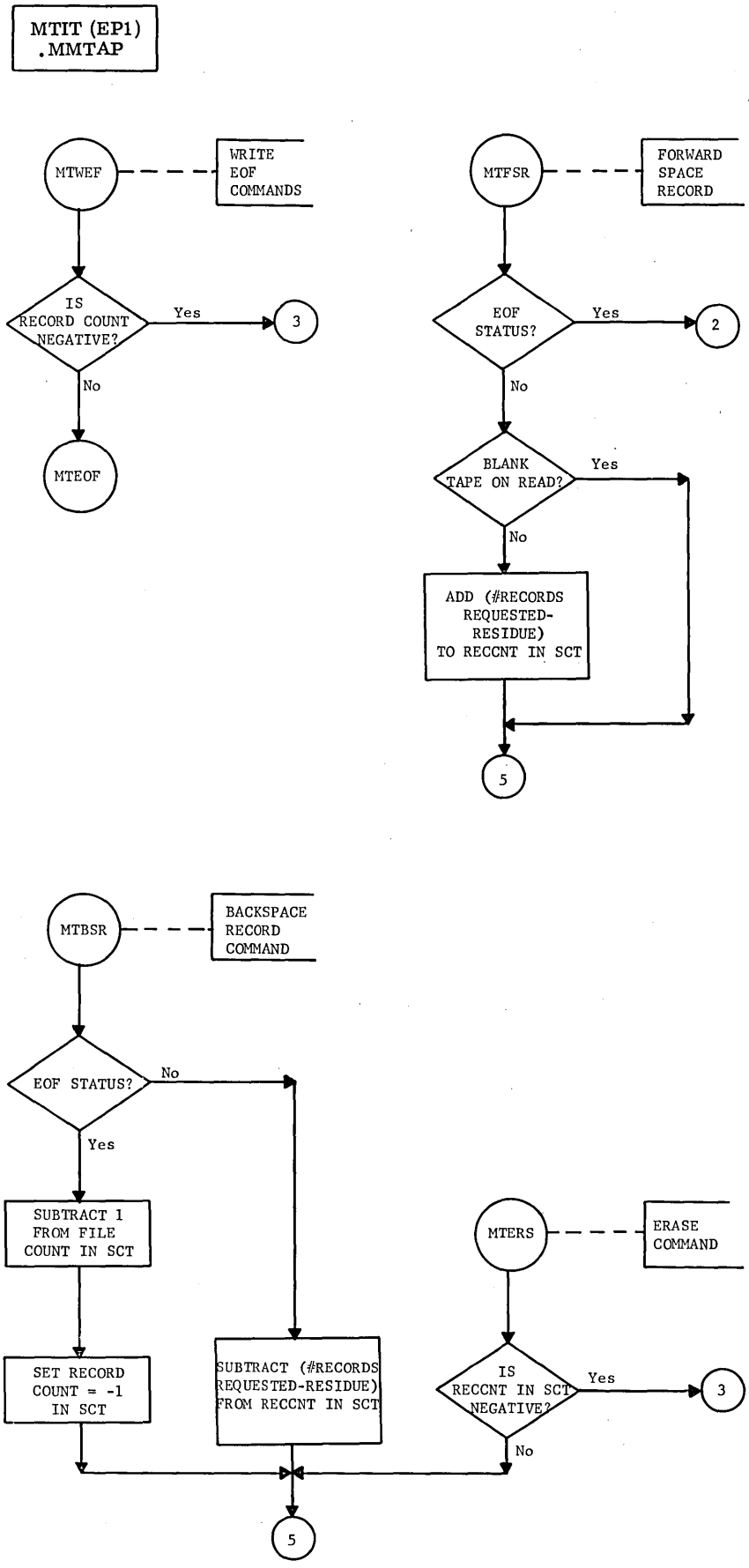
CARD READER INITIALIZATION

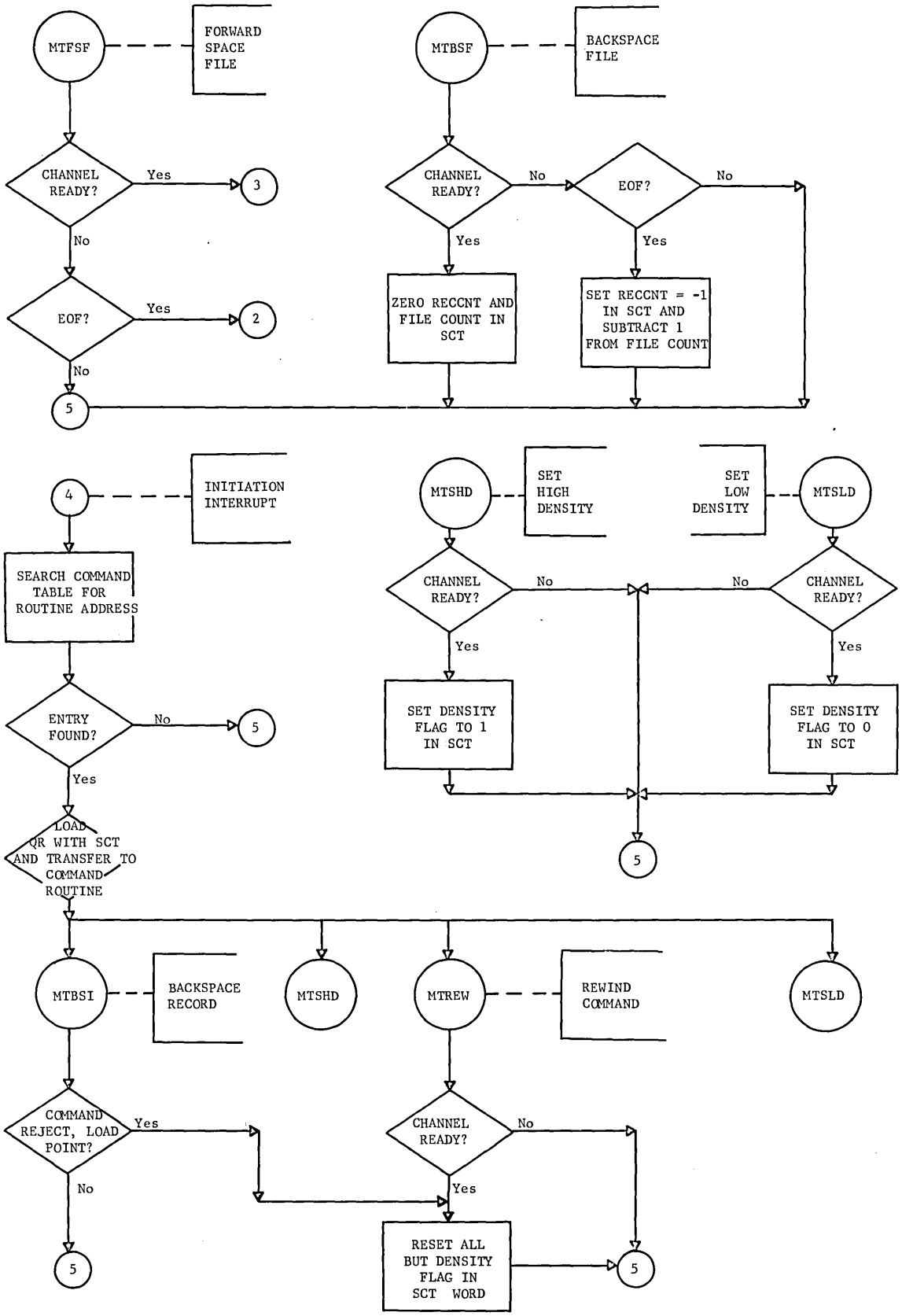


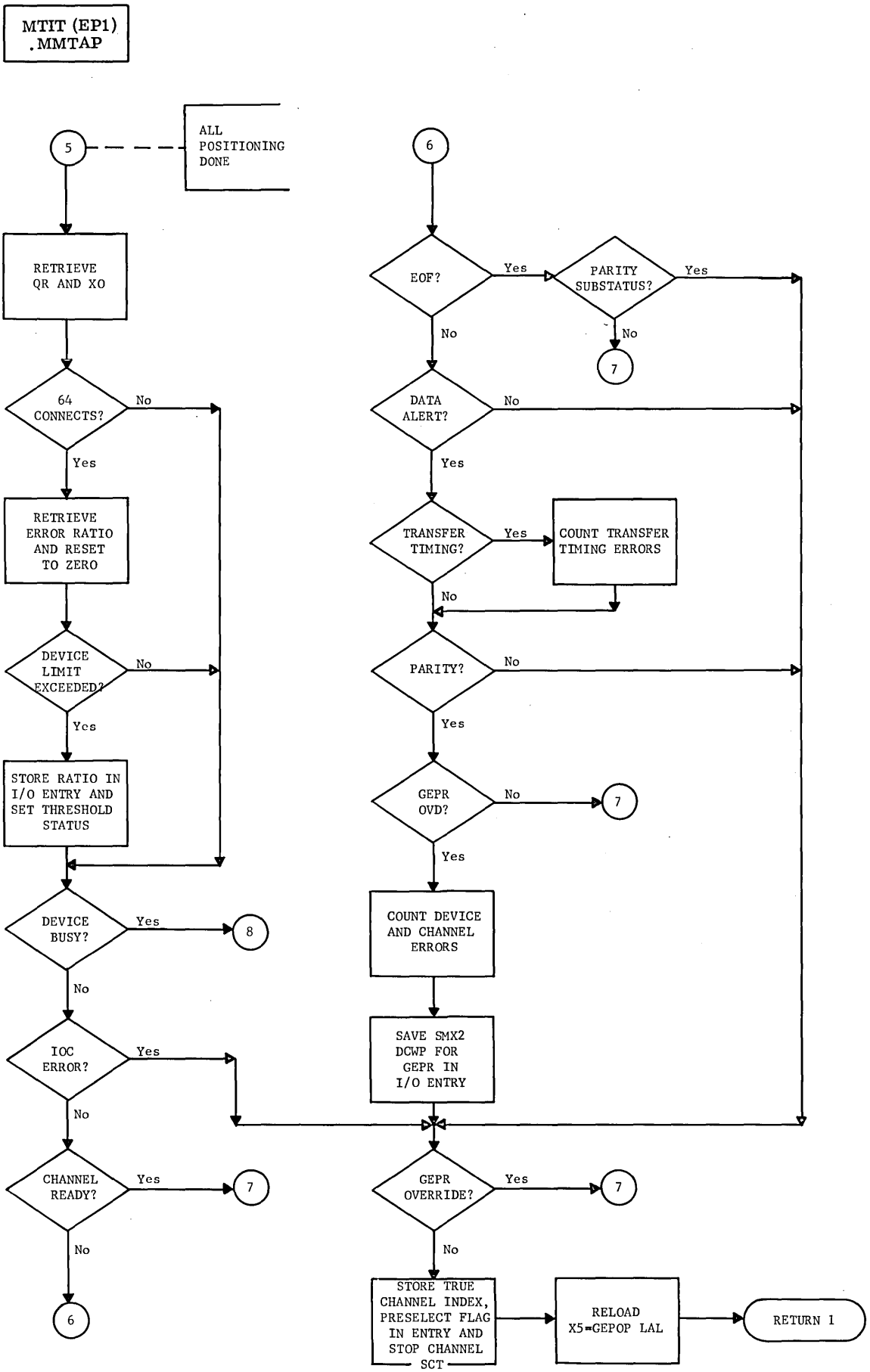
MAGNETIC TAPE INTERRUPT HANDLER

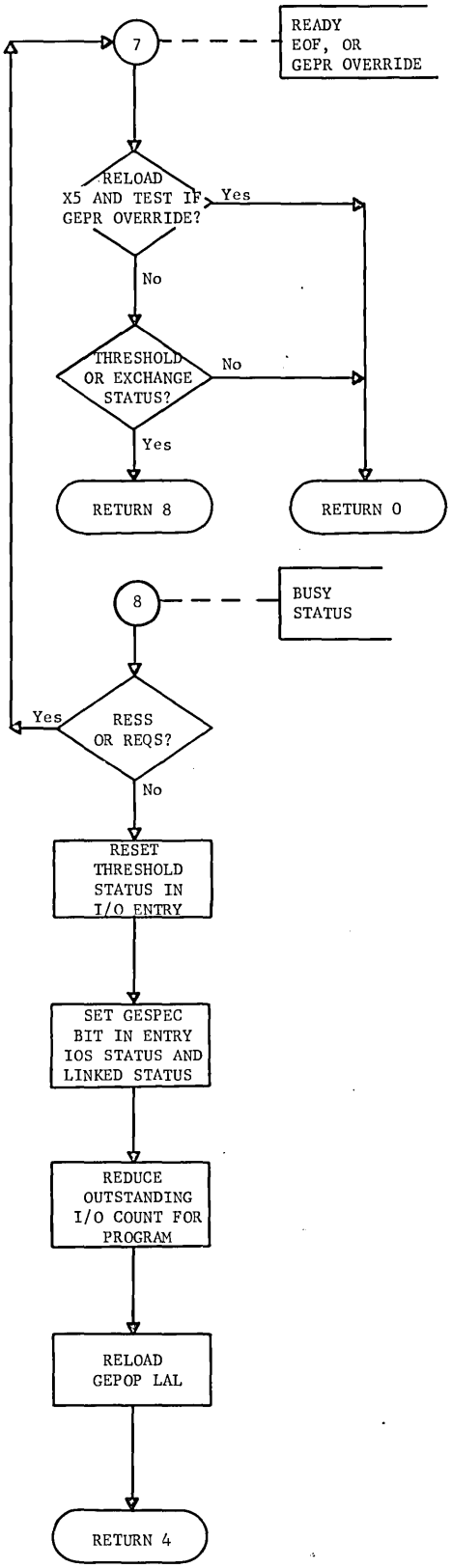
MTIT (EP1)
.MMTAP

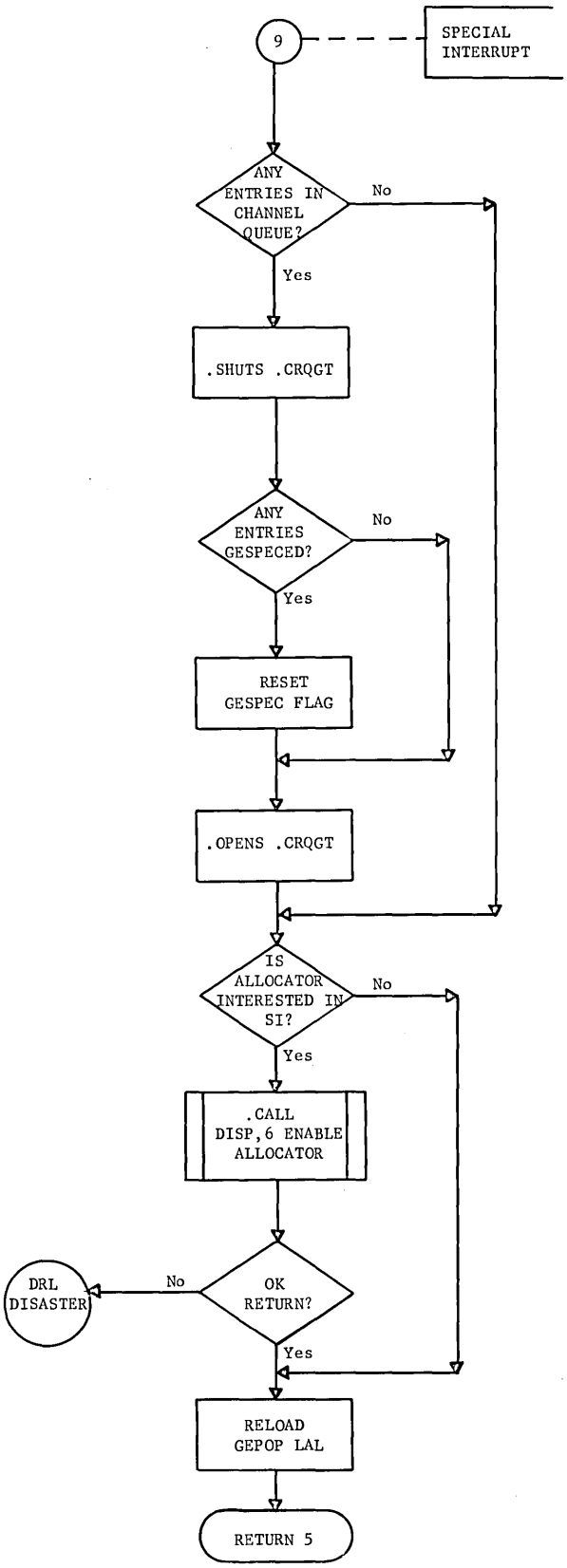



MTIT (EP1)
.MMTAP
MTWEF
WRITE
EOF
COMMANDS
IS
RECORD COUNT
NEGATIVE?
Yes
3
No
MTEOF
MTFSR
FORWARD
SPACE
RECORD
EOF
STATUS?
Yes
2
No
BLANK
TAPE ON READ?
Yes
No
ADD (#RECORDS
REQUESTED-
RESIDUE)
TO RECCNT IN SCT
5
MTBSR
BACKSPACE
RECORD
COMMAND
EOF STATUS?
No
Yes
SUBTRACT 1
FROM FILE
COUNT IN SCT
SET RECORD
COUNT = -1
IN SCT
SUBTRACT (#RECORDS
REQUESTED-RESIDUE)
FROM RECCNT IN SCT
5
MTERS
ERASE
COMMAND
IS
RECCNT IN SCT
NEGATIVE?
Yes
3
No

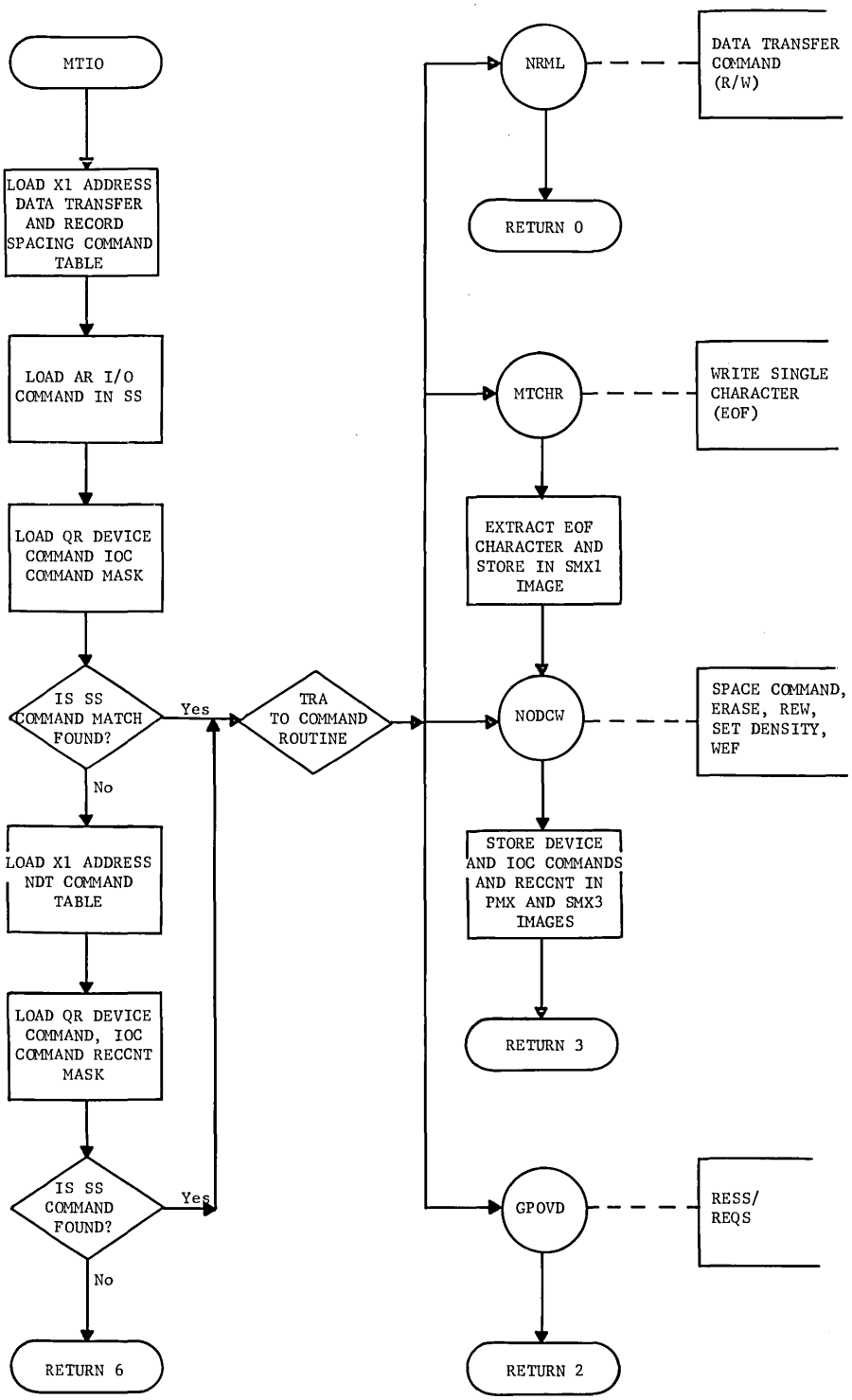

MTFSF
FORWARD SPACE FILE
CHANNEL READY?
Yes
3
No
EOF?
Yes
2
No
5
MTBSF
BACKSPACE FILE
CHANNEL READY?
No
EOF?
No
Yes
Yes
ZERO RECCNT AND FILE COUNT IN SCT
SET RECCNT = -1 IN SCT AND SUBTRACT 1 FROM FILE COUNT
4
INITIATION INTERRUPT
SEARCH COMMAND TABLE FOR ROUTINE ADDRESS
ENTRY FOUND?
No
5
Yes
LOAD QR WITH SCT AND TRANSFER TO COMMAND ROUTINE
MTSHD
SET HIGH DENSITY
SET LOW DENSITY
MTSLD
CHANNEL READY?
No
No
CHANNEL READY?
Yes
Yes
SET DENSITY FLAG TO 1 IN SCT
SET DENSITY FLAG TO 0 IN SCT
5
MTBSI
BACKSPACE RECORD
MTSHD
MTREW
REWIND COMMAND
MTSLD
COMMAND REJECT, LOAD POINT?
Yes
No
5
CHANNEL READY?
No
Yes
RESET ALL BUT DENSITY FLAG IN SCT WORD
5


MTIT (EP1)
.MMTAP
5
ALL POSITIONING DONE
RETRIEVE QR AND XO
64 CONNECTS?
No
Yes
RETRIEVE ERROR RATIO AND RESET TO ZERO
DEVICE LIMIT EXCEEDED?
No
Yes
STORE RATIO IN I/O ENTRY AND SET THRESHOLD STATUS
DEVICE BUSY?
Yes
8
No
IOC ERROR?
Yes
No
CHANNEL READY?
Yes
7
No
6
6
EOF?
Yes
PARITY SUBSTATUS?
Yes
No
No
7
DATA ALERT?
No
Yes
TRANSFER TIMING?
Yes
COUNT TRANSFER TIMING ERRORS
No
PARITY?
No
Yes
GEPR OVD?
No
7
Yes
COUNT DEVICE AND CHANNEL ERRORS
SAVE SMX2 DCWP FOR GEPR IN I/O ENTRY
GEPR OVERRIDE?
Yes
7
No
STORE TRUE CHANNEL INDEX, PRESELECT FLAG IN ENTRY AND STOP CHANNEL
SCT
RELOAD X5=GEPOP LAL
RETURN 1

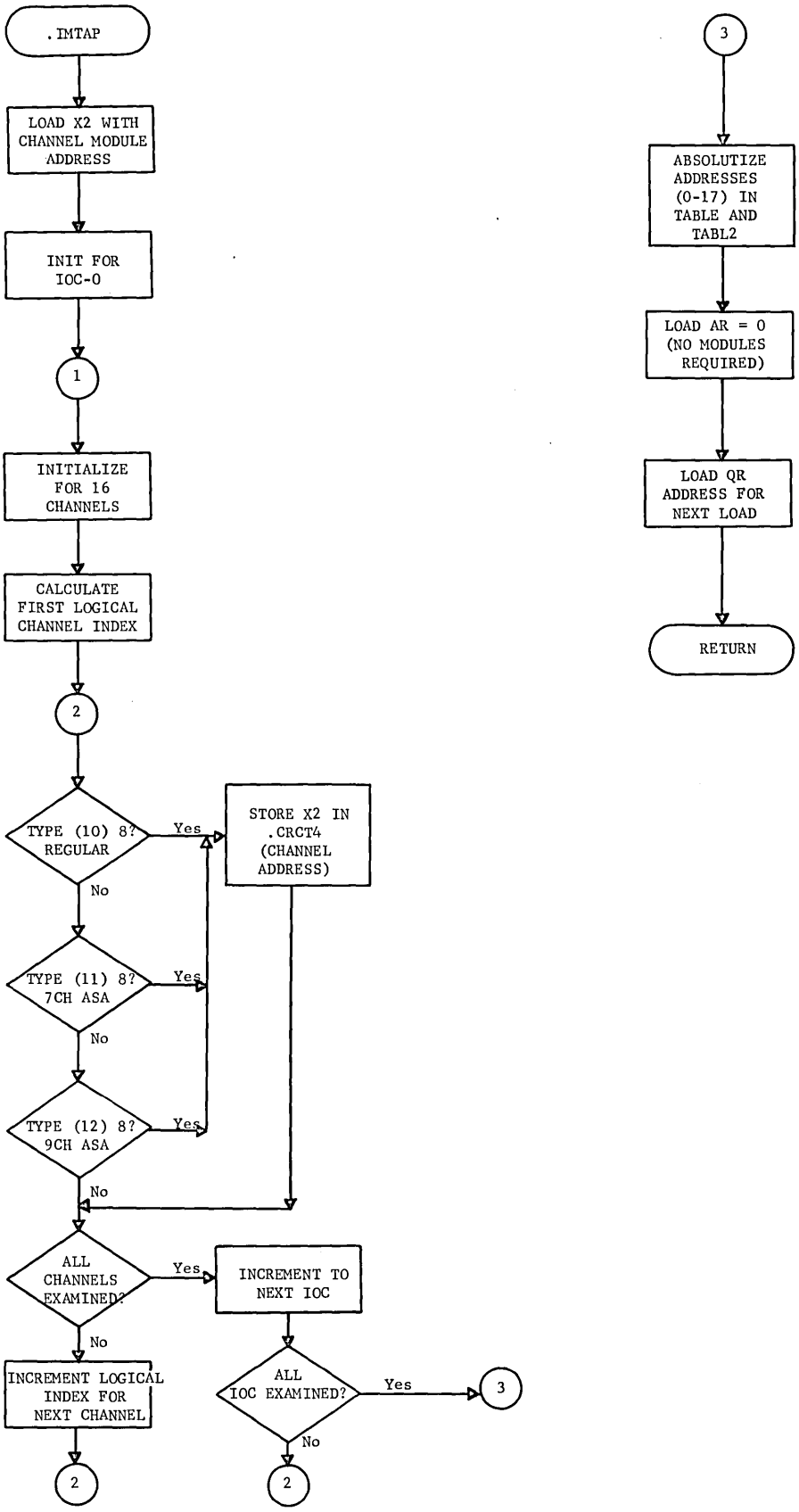

7
READY
EOF, OR
GEPR OVERRIDE
RELOAD X5 AND TEST IF GEPR OVERRIDE?
Yes
No
THRESHOLD OR EXCHANGE STATUS?
No
Yes
RETURN 8
RETURN 0
8
BUSY
STATUS
RESS OR REQS?
Yes
No
RESET THRESHOLD STATUS IN I/O ENTRY
SET GESPEC BIT IN ENTRY IOS STATUS AND LINKED STATUS
REDUCE OUTSTANDING I/O COUNT FOR PROGRAM
RELOAD GEPOP LAL
RETURN 4


9
SPECIAL
INTERRUPT
ANY ENTRIES IN CHANNEL QUEUE?
No
Yes
.SHUTS .CRQGT
ANY ENTRIES GESPECED?
No
Yes
RESET GESPEC FLAG
.OPENS .CRQGT
IS ALLOCATOR INTERESTED IN SI?
No
Yes
.CALL DISP,6 ENABLE ALLOCATOR
DRL DISASTER
No
OK RETURN?
Yes
RELOAD GEPOP LAL
RETURN 5

MAGNETIC TAPE REQUEST

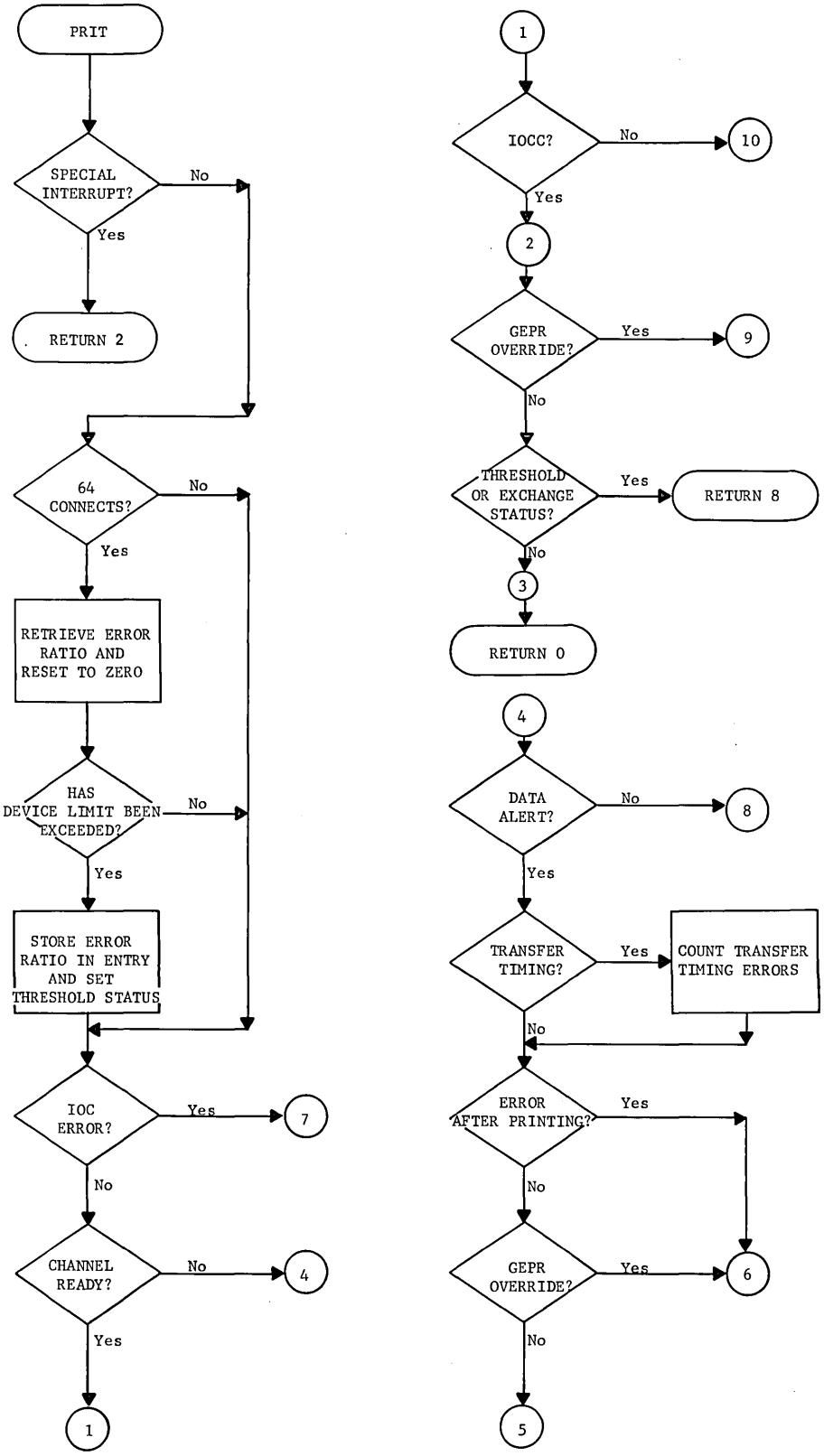


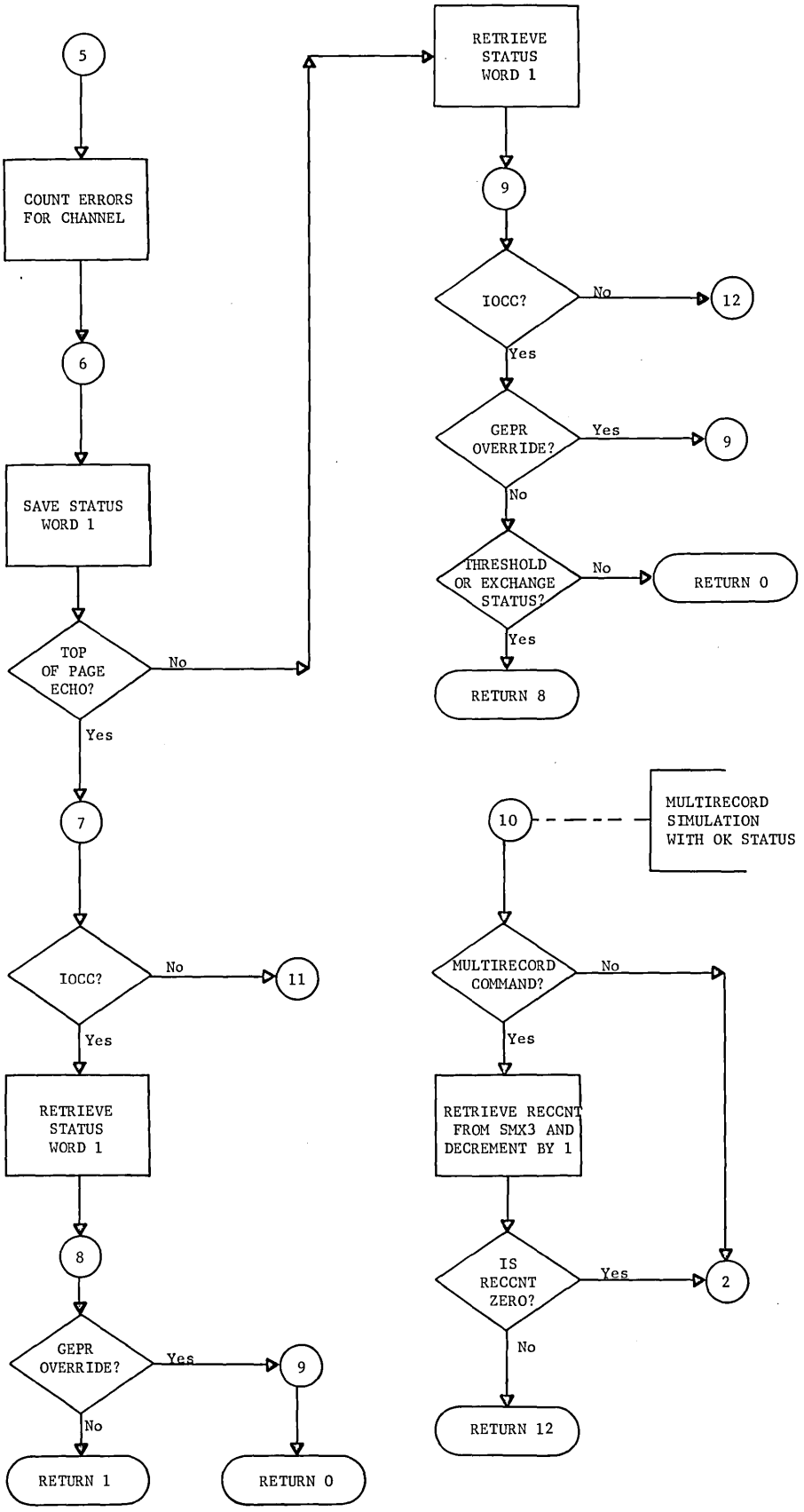
MAGNETIC TAPE INITIALIZATION

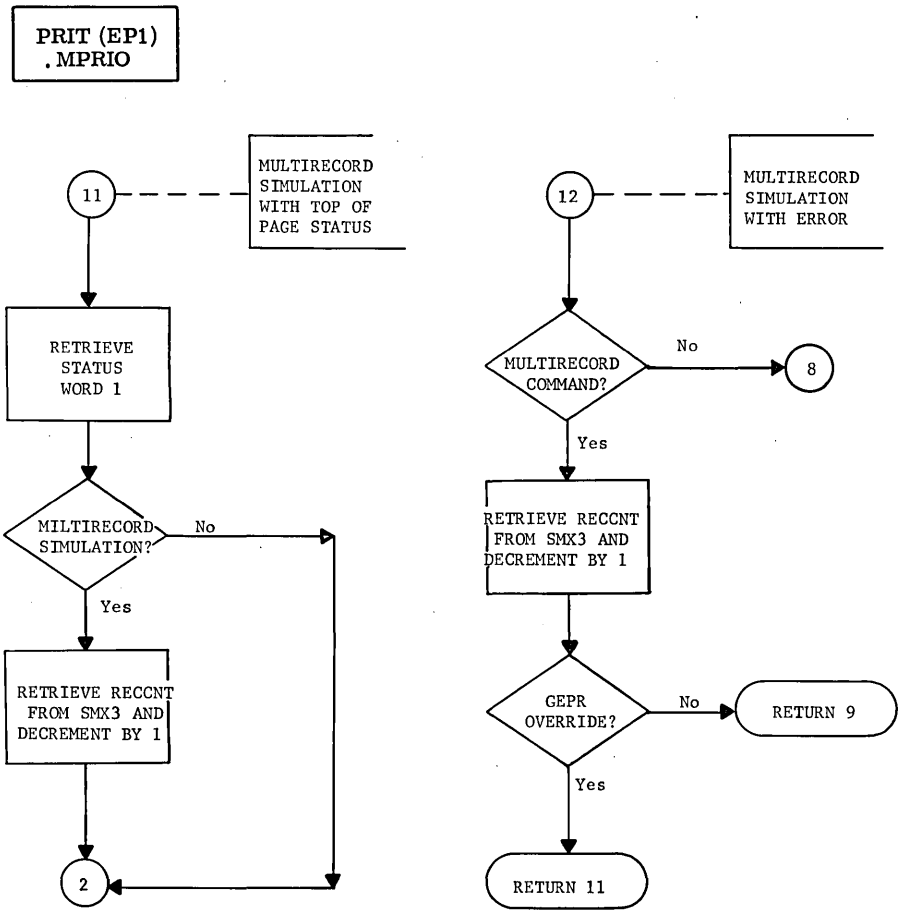


PRIT (EP1)
.MPRIO

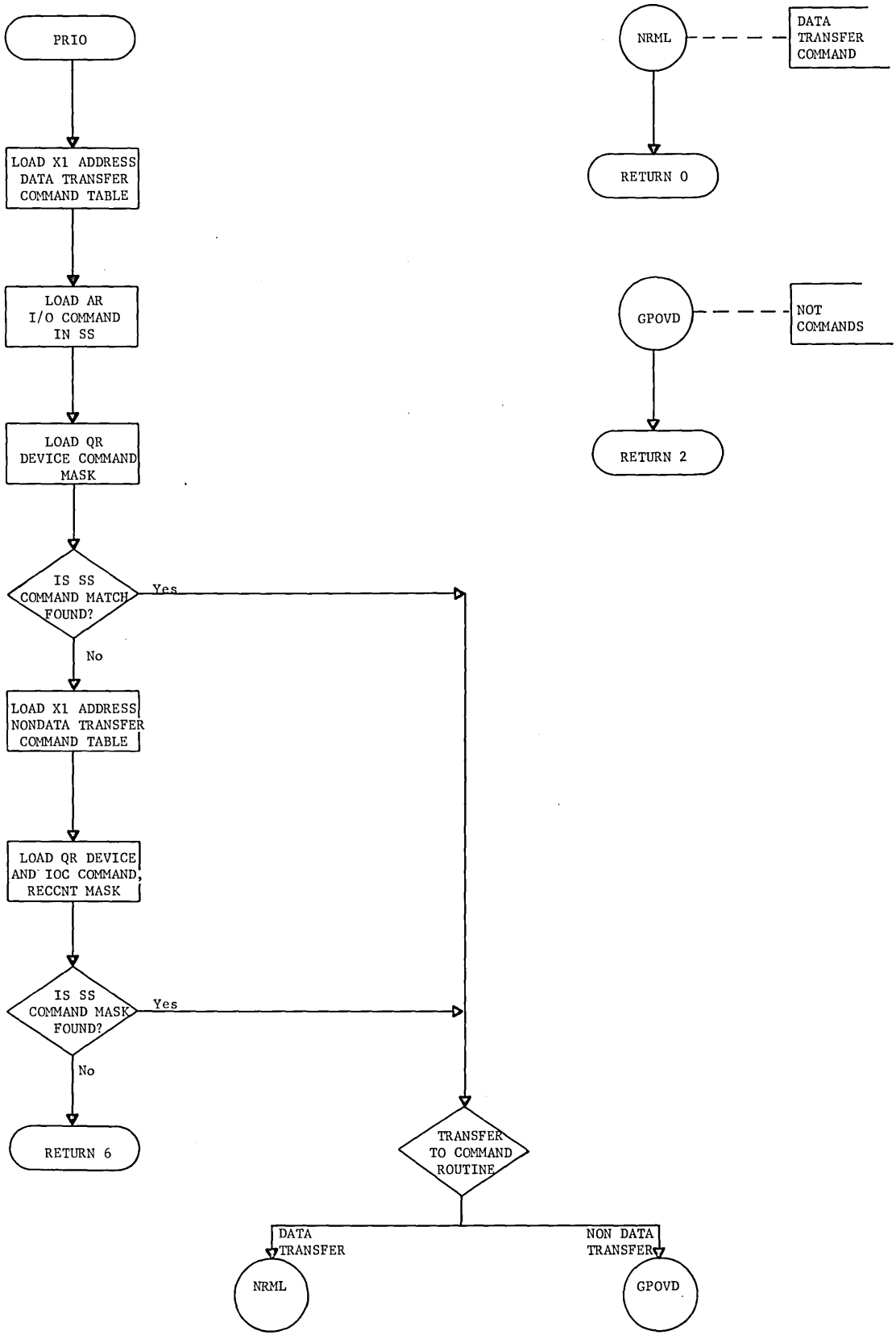
PRINTER INTERRUPT HANDLER




5
COUNT ERRORS FOR CHANNEL
6
SAVE STATUS WORD 1
TOP OF PAGE ECHO?
No
Yes
7
IOCC?
No
11
Yes
RETRIEVE STATUS WORD 1
8
GEPR OVERRIDE?
Yes
9
No
RETURN 1
RETURN 0
RETRIEVE STATUS WORD 1
9
IOCC?
No
12
Yes
GEPR OVERRIDE?
Yes
9
No
THRESHOLD OR EXCHANGE STATUS?
No
RETURN 0
Yes
RETURN 8
10
MULTIRECORD SIMULATION WITH OK STATUS
MULTIRECORD COMMAND?
No
Yes
RETRIEVE RECCNT FROM SMX3 AND DECREMENT BY 1
IS RECCNT ZERO?
Yes
2
No
RETURN 12

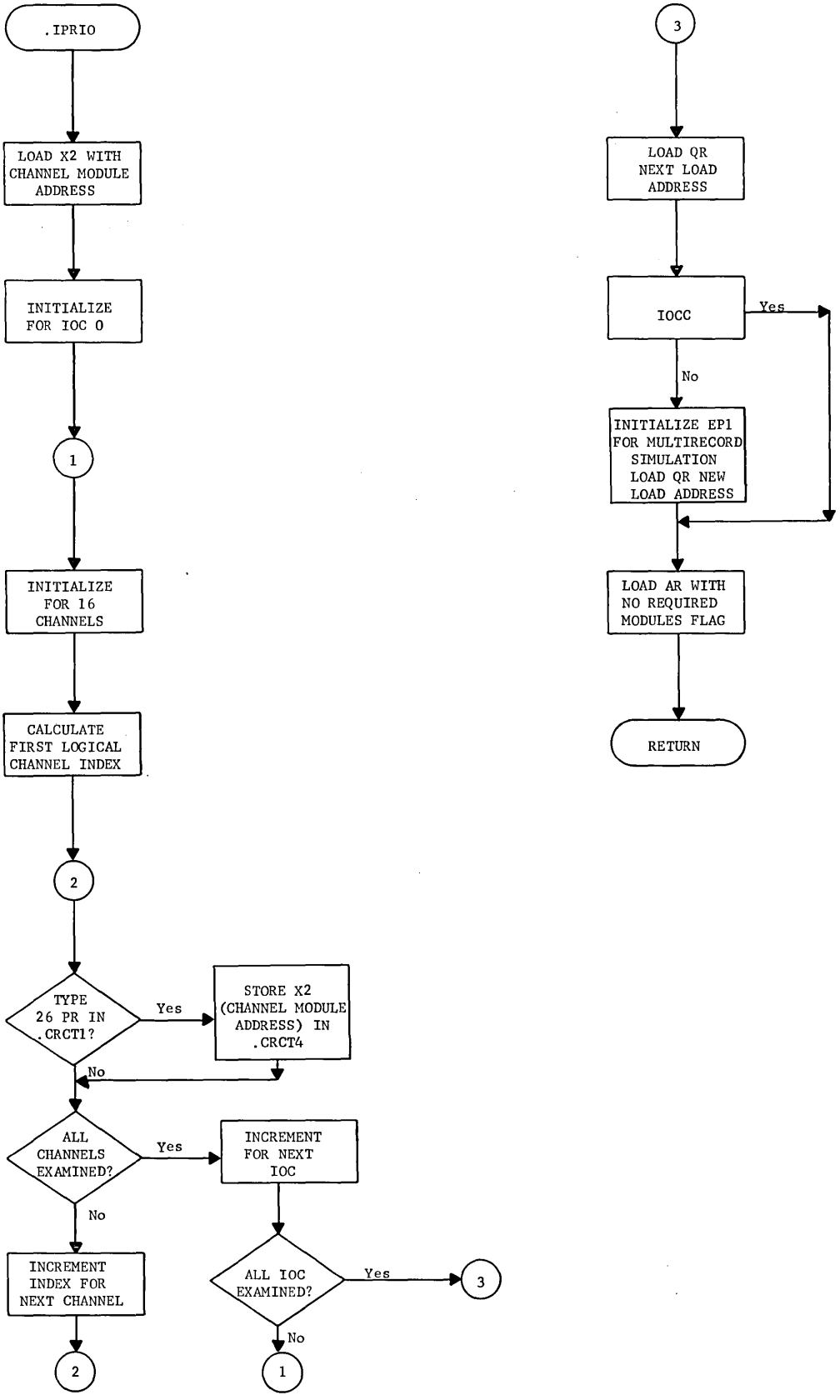

PRIT (EP1)
.MPRIO
11
MULTIRECORD SIMULATION WITH TOP OF PAGE STATUS
RETRIEVE STATUS WORD 1
MILTIRECORD SIMULATION?
No
Yes
RETRIEVE RECCNT FROM SMX3 AND DECREMENT BY 1
2
12
MULTIRECORD SIMULATION WITH ERROR
MULTIRECORD COMMAND?
No
8
Yes
RETRIEVE RECCNT FROM SMX3 AND DECREMENT BY 1
GEPR OVERRIDE?
No
RETURN 9
Yes
RETURN 11

PRINTER REQUEST



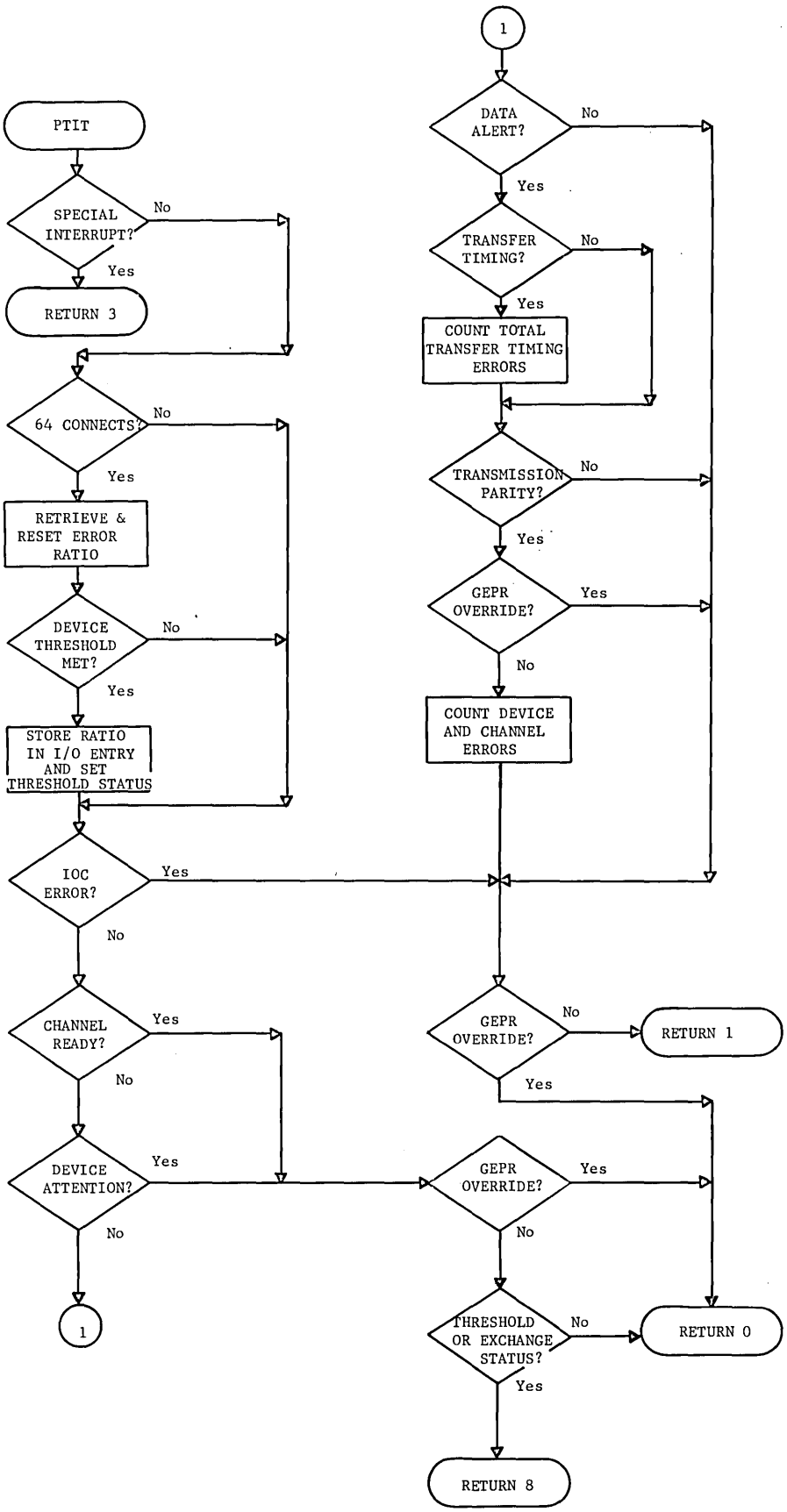
.IPRIO
.MPRIO

PRINTER INITIALIZATION



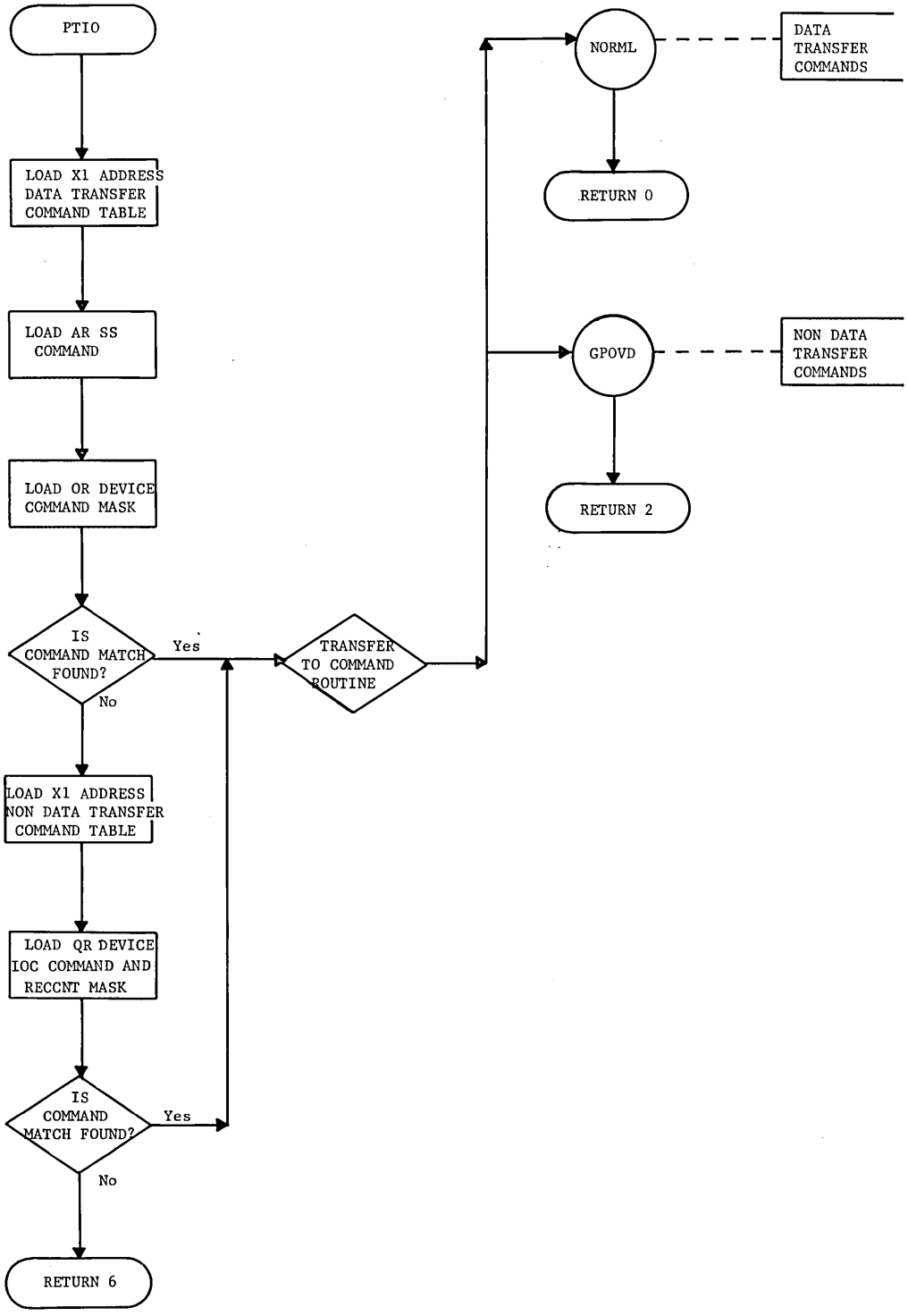
PAPER TAPE INTERRUPT HANDLER

PTIT (EP1)
.MPTAP

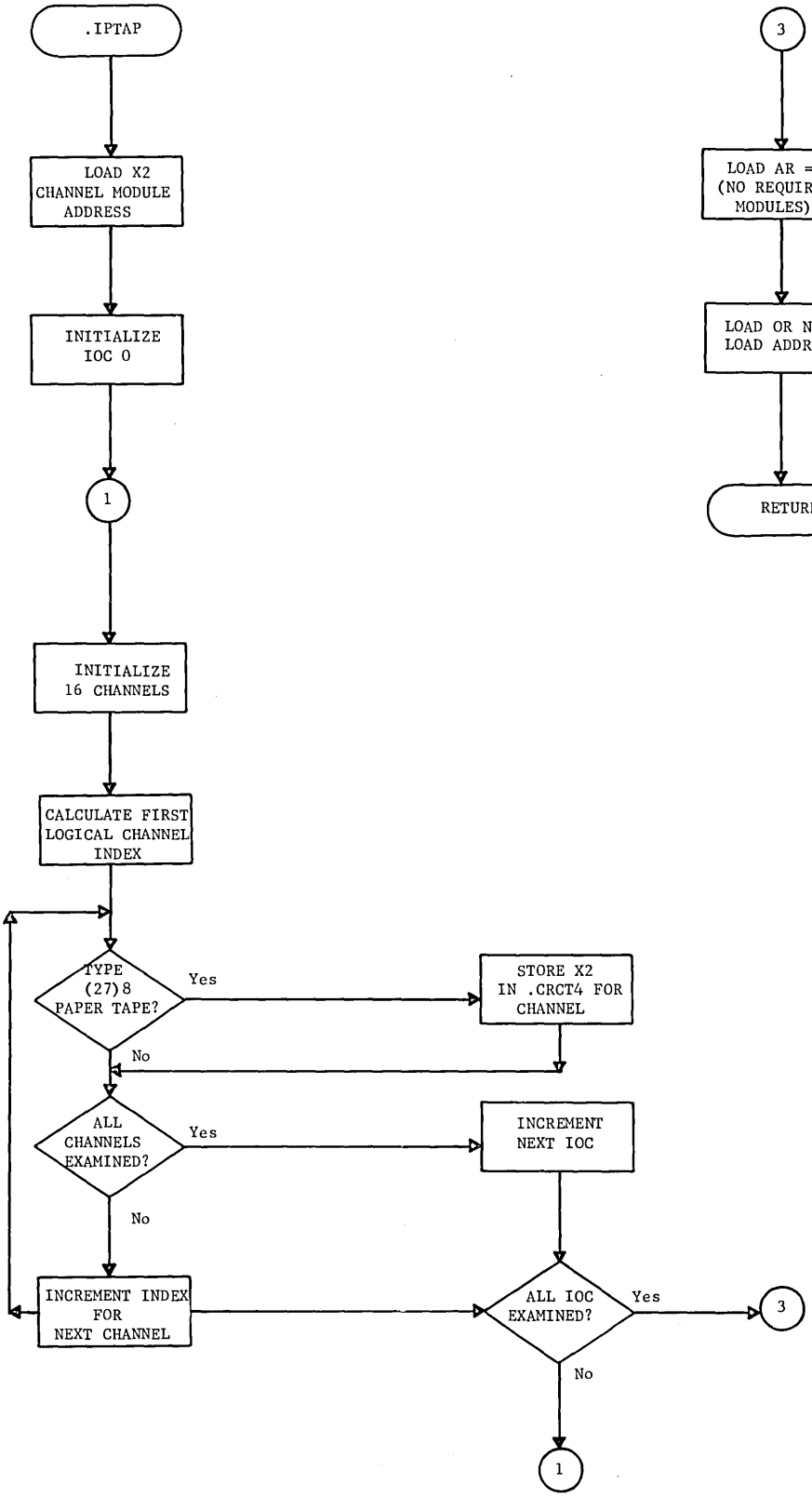


PTIO (EP2)
.MPTAP

PAPER TAPE REQUEST

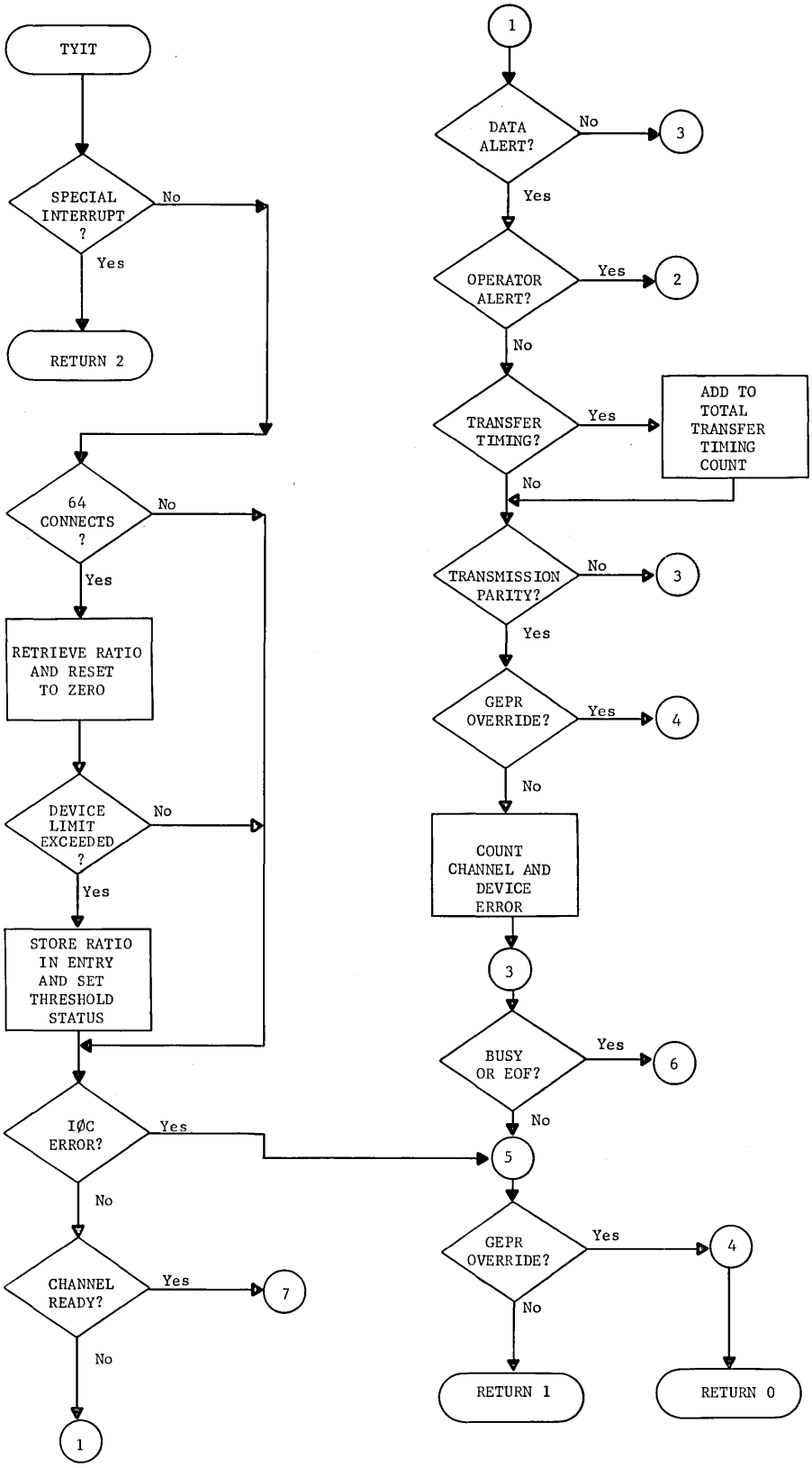


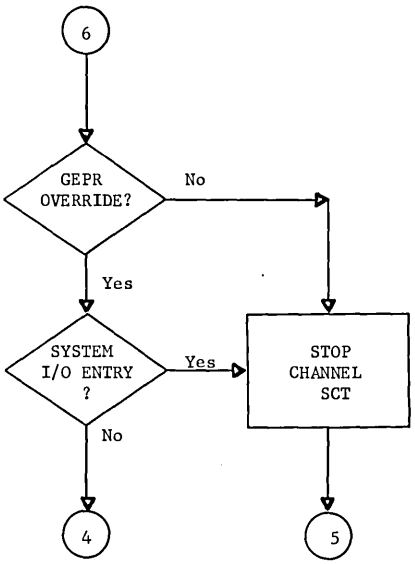
PAPER TAPE INITIALIZATION

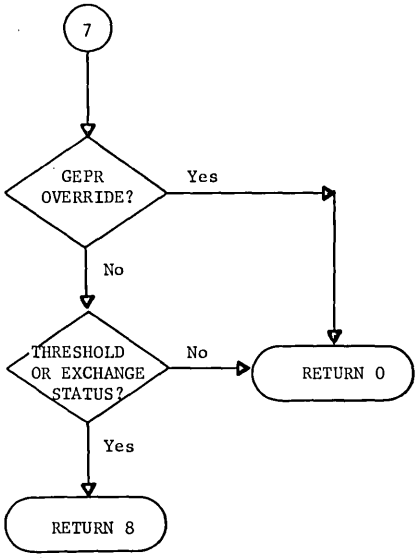


TYIT (EP1)
.MTYPE

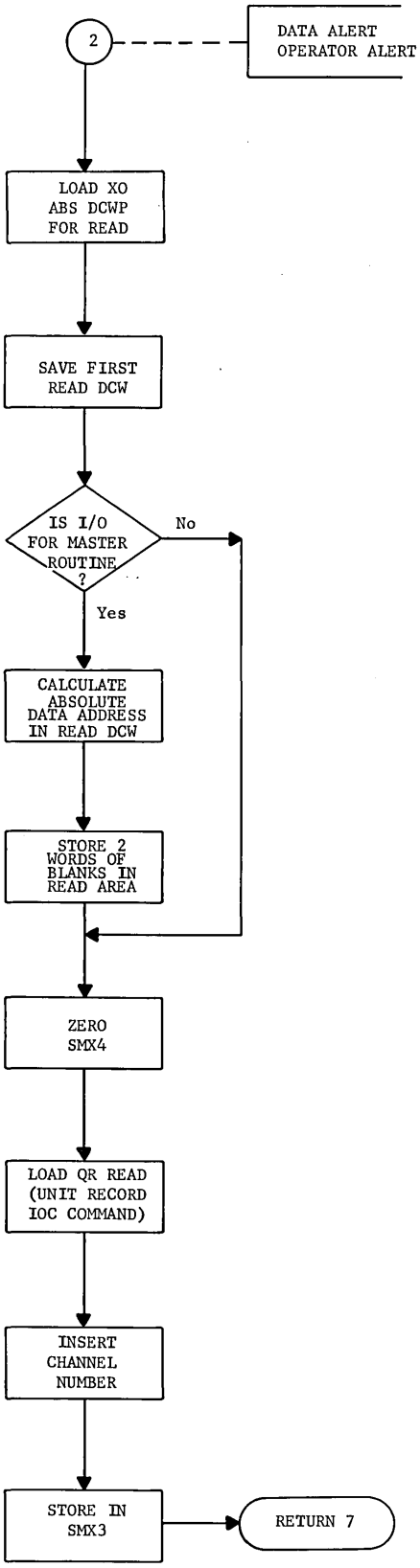
TYPEWRITER INTERRUPT HANDLER




6
GEPR OVERRIDE?
No
Yes
SYSTEM I/O ENTRY ?
Yes
STOP CHANNEL SCT
No
4
5

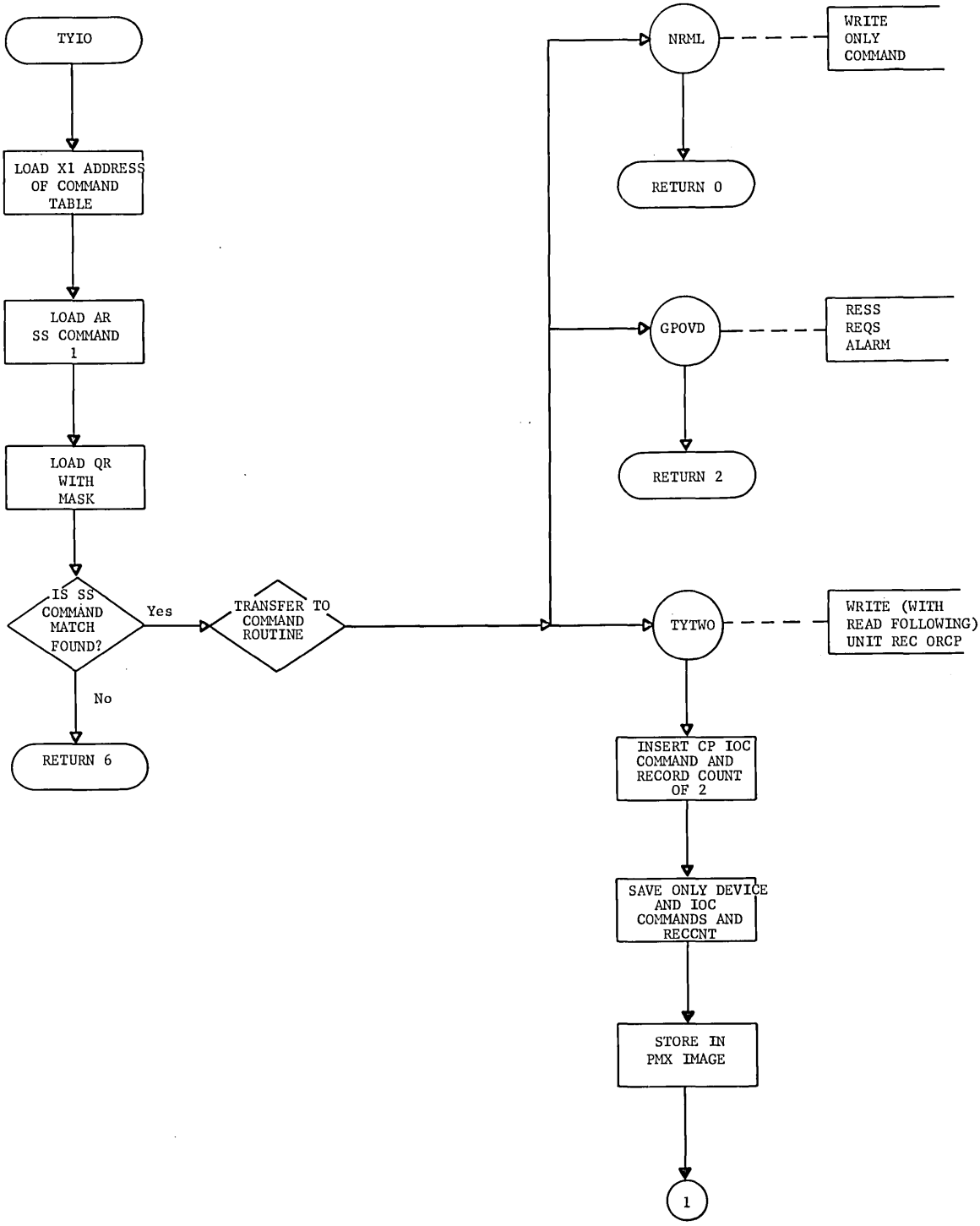

7
GEPR OVERRIDE?
Yes
No
THRESHOLD OR EXCHANGE STATUS?
No
RETURN 0
Yes
RETURN 8

TYIT (EP1)
.MTYPE

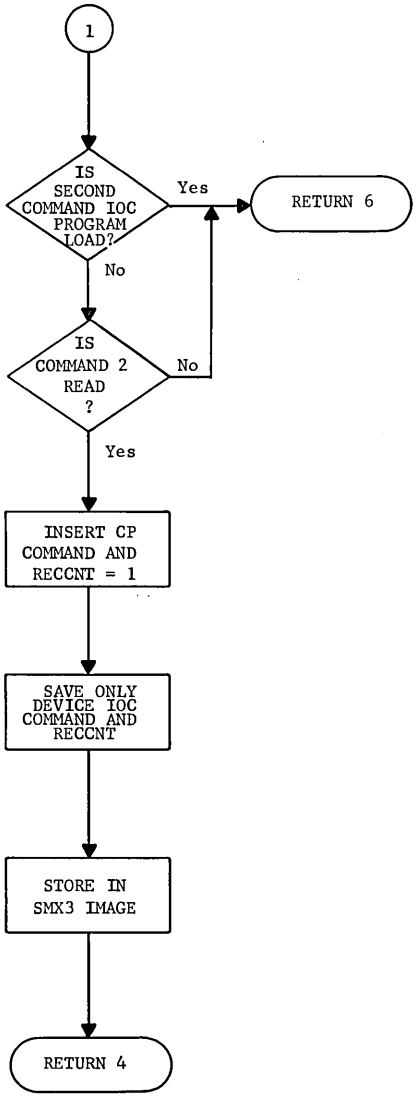


TYIO (EP2)
.MTYPE

TYPEWRITER REQUEST

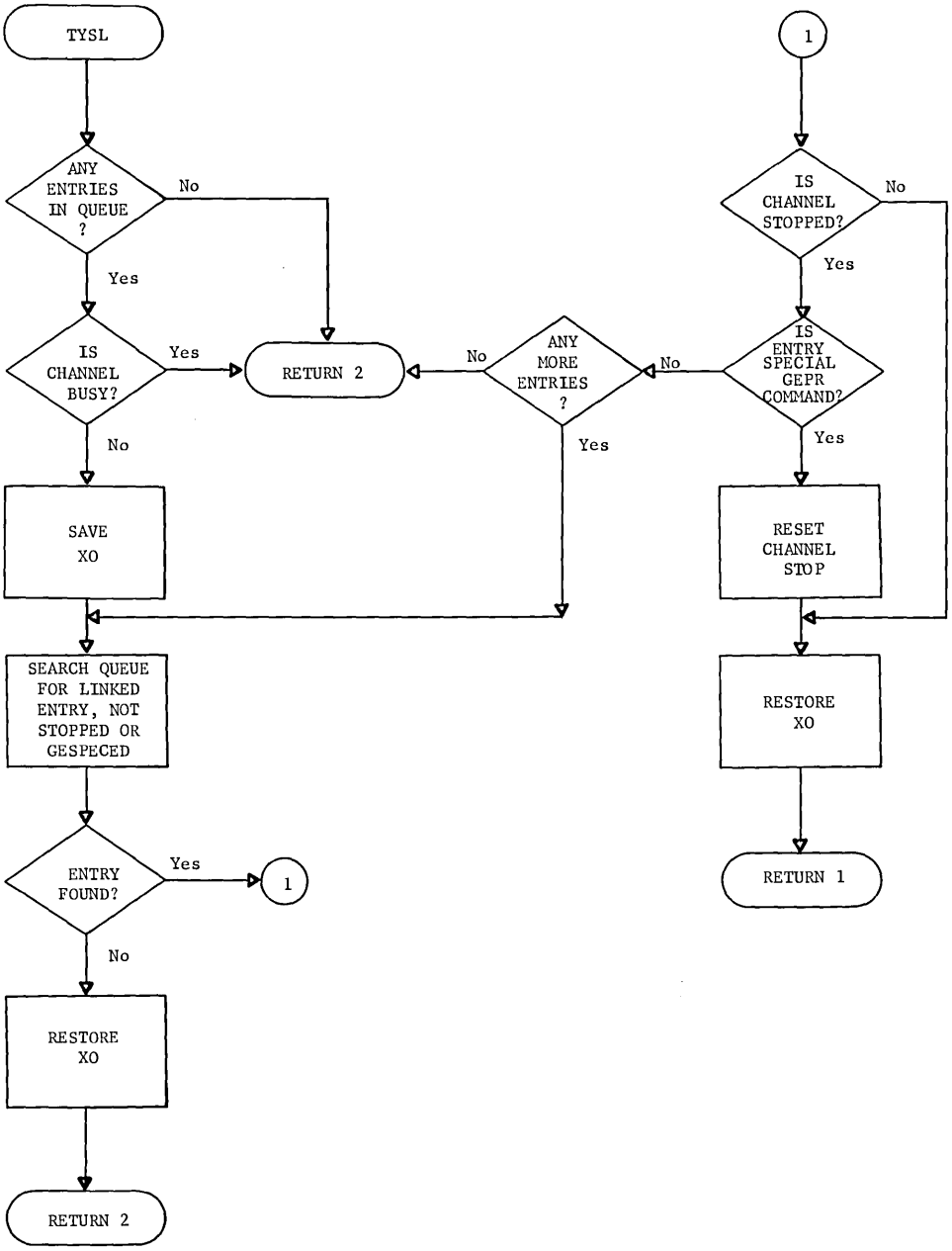


TYIO (EP2)
.MTYPE



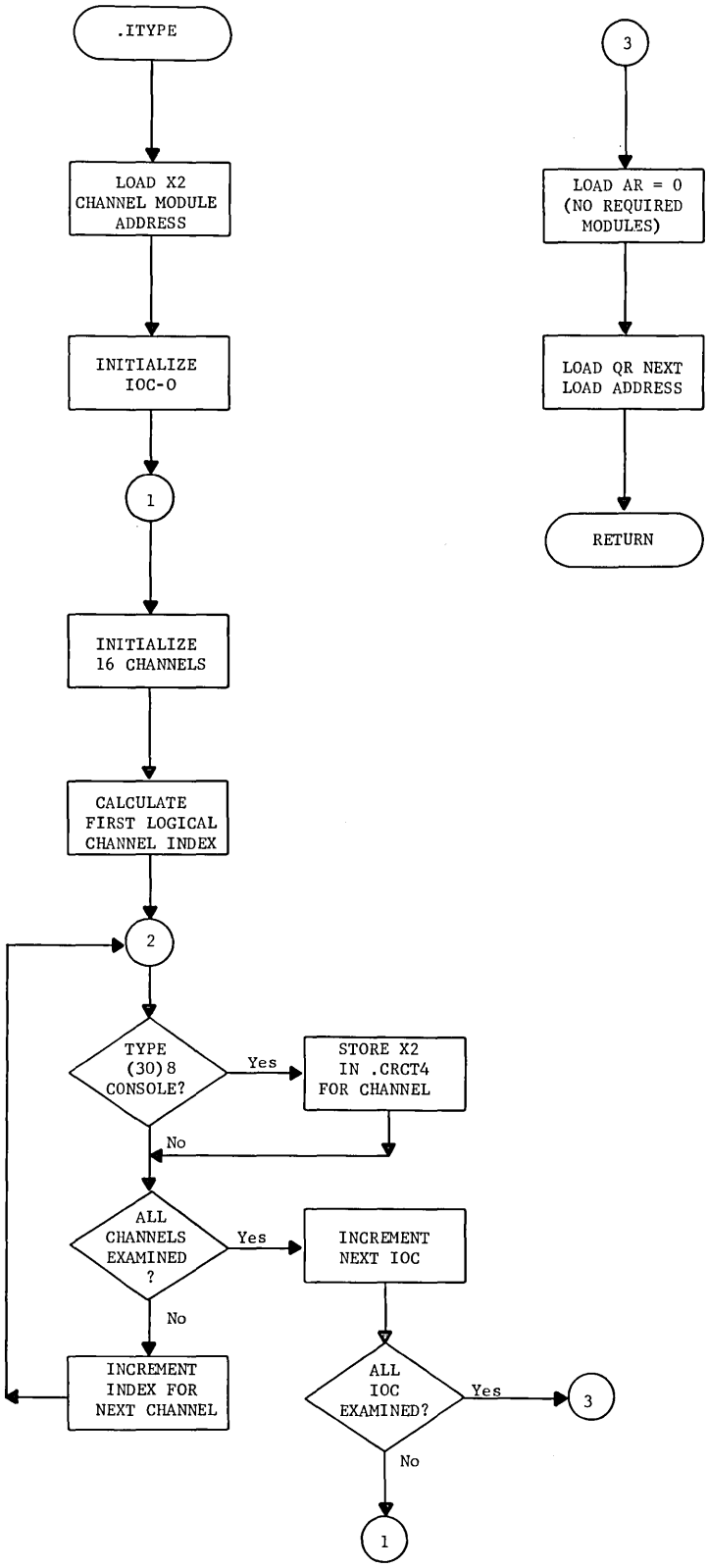
TYSL (EP3)
.MTYPE

TYPEWRITER SELECT

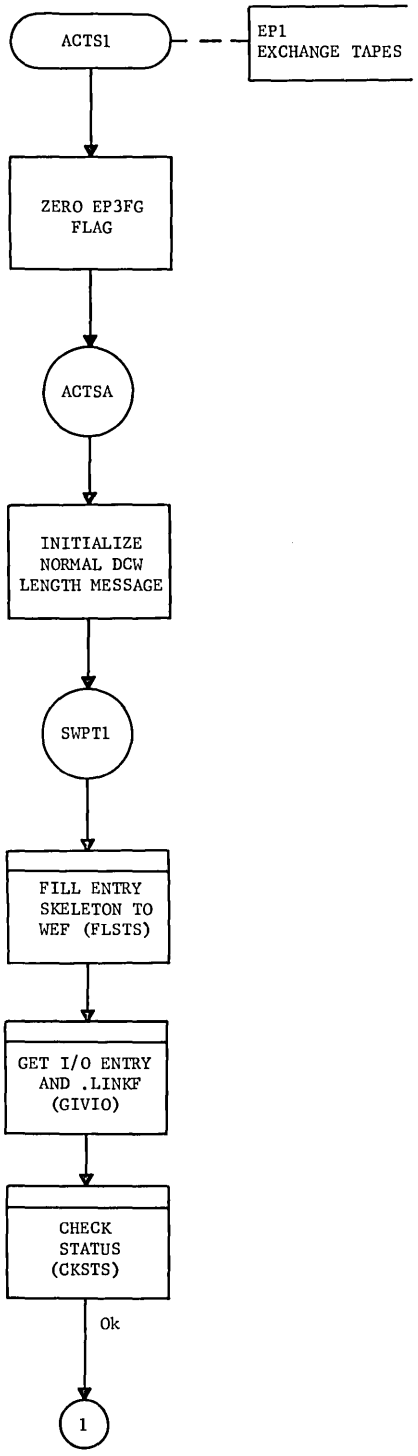


.ITYPE
.MTYPE

TYPEWRITER INITIALIZATION

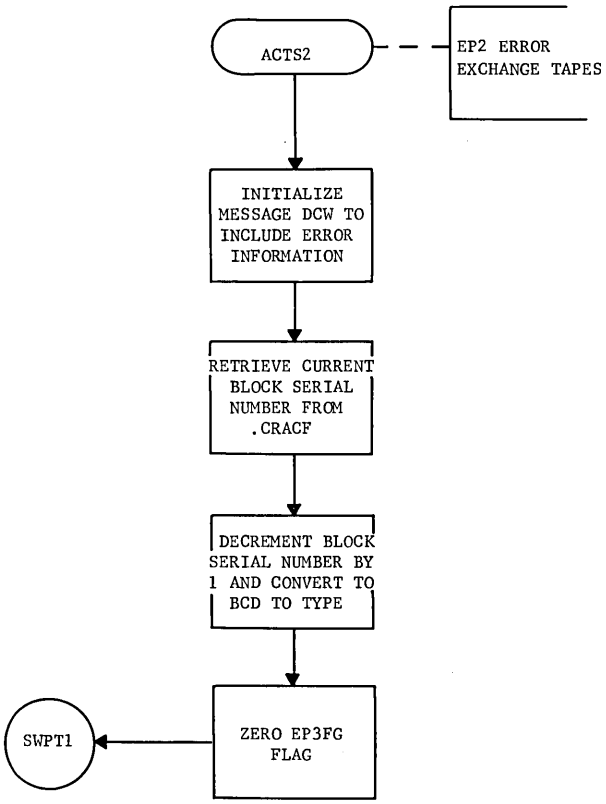


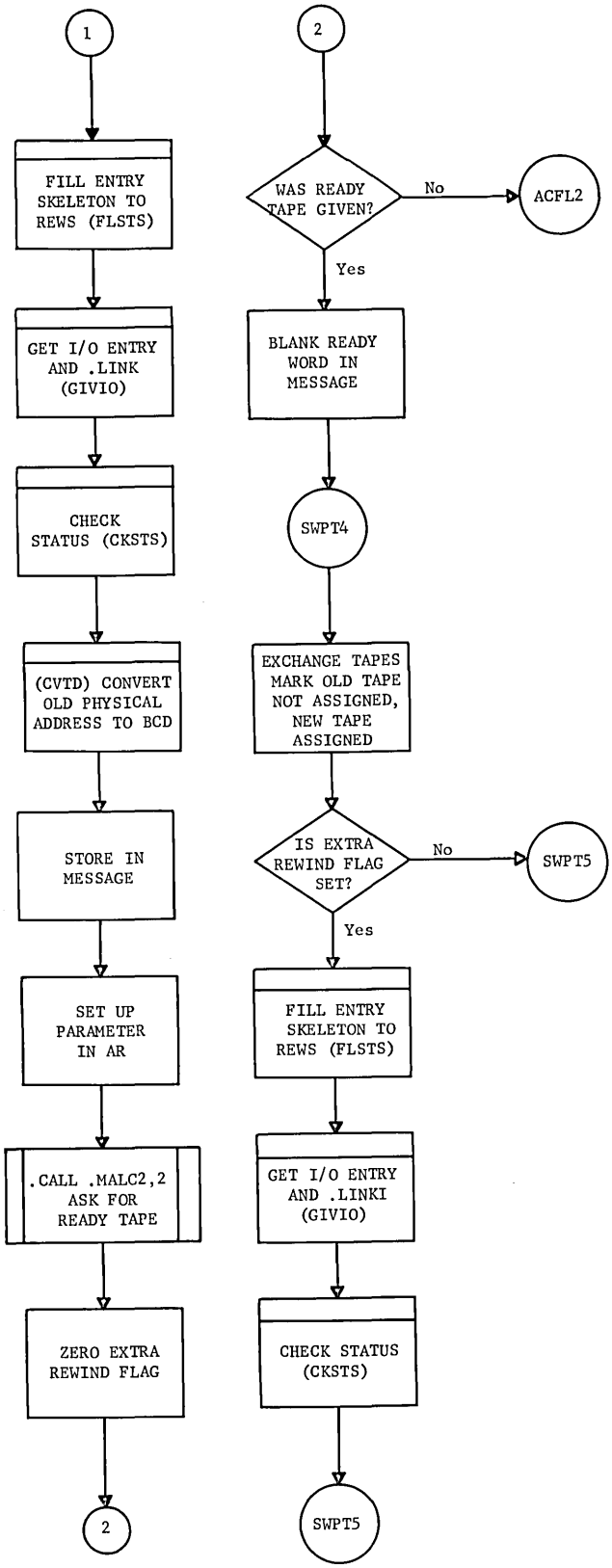
NORMAL CLOSE OF ACCOUNTING FILE

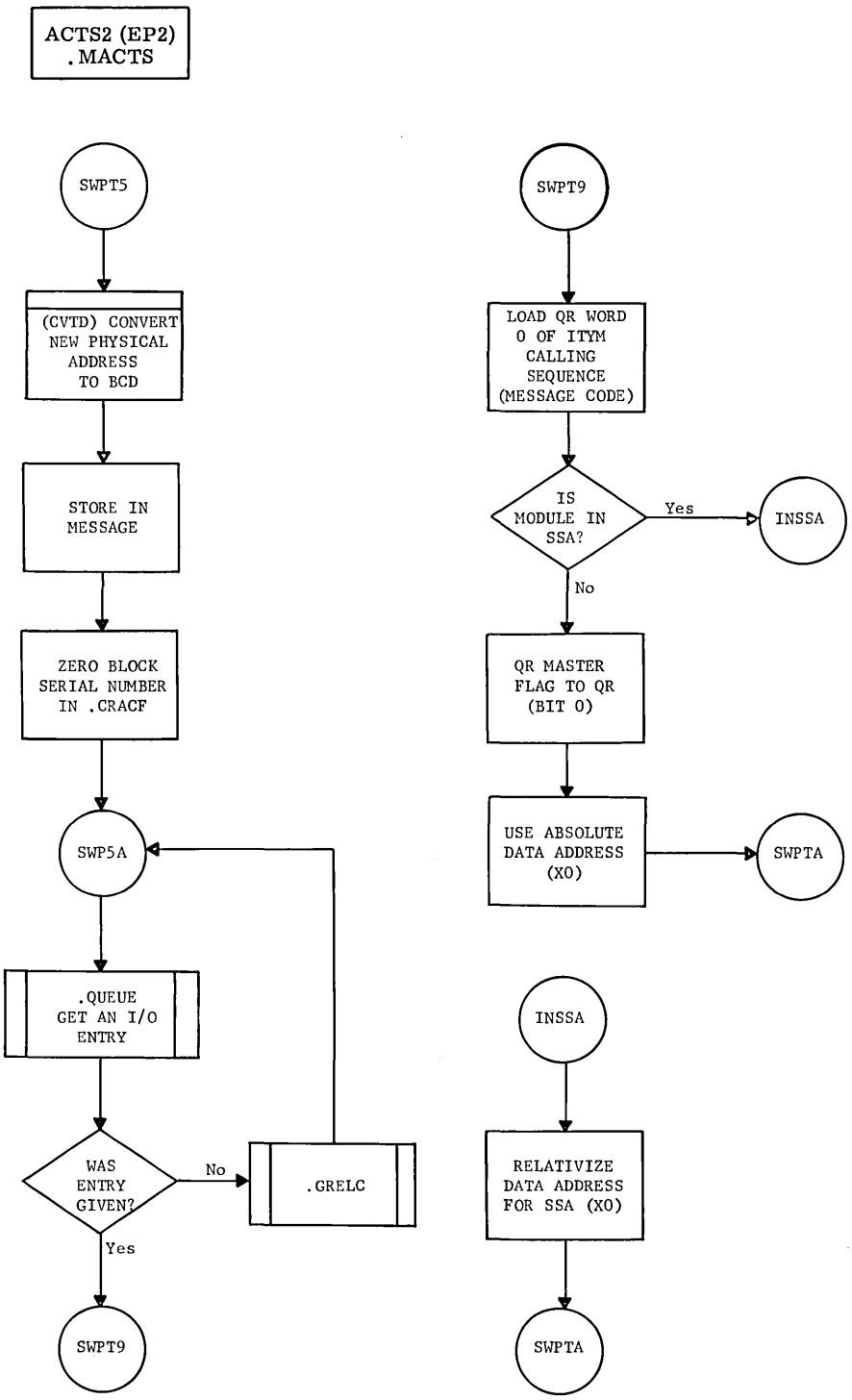


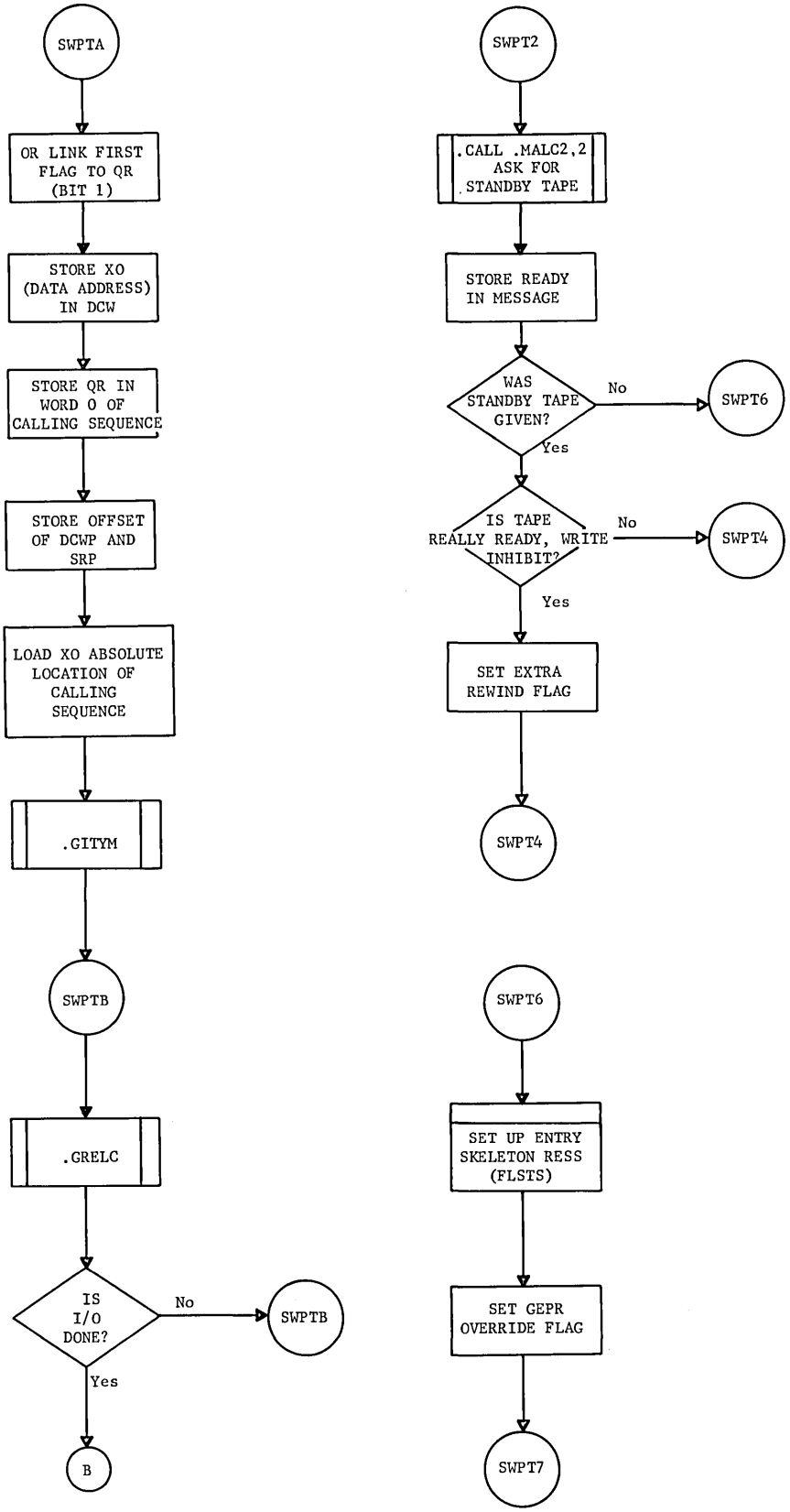
ACTS2 (EP2)
.MACTS

ERROR CLOSE OF ACCOUNTING FILE

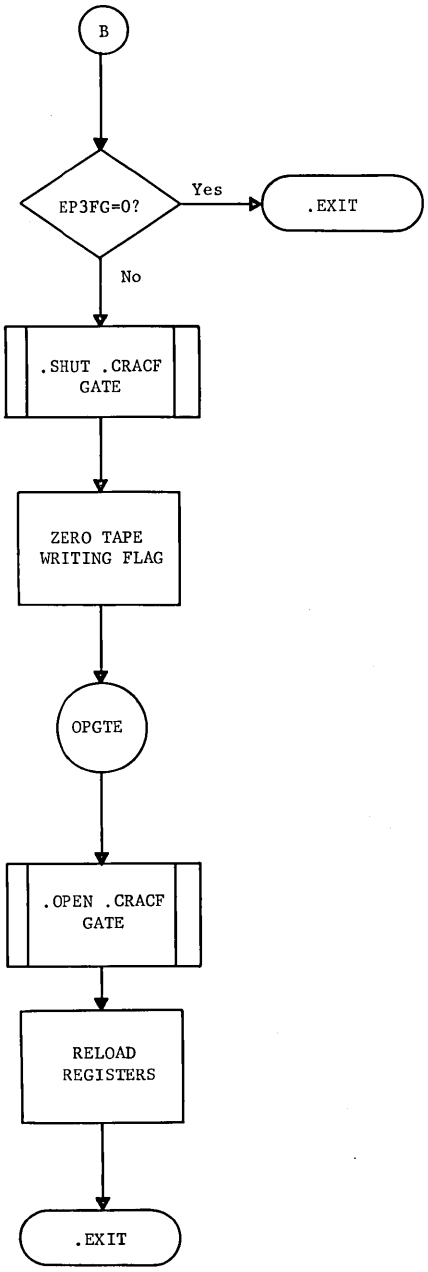


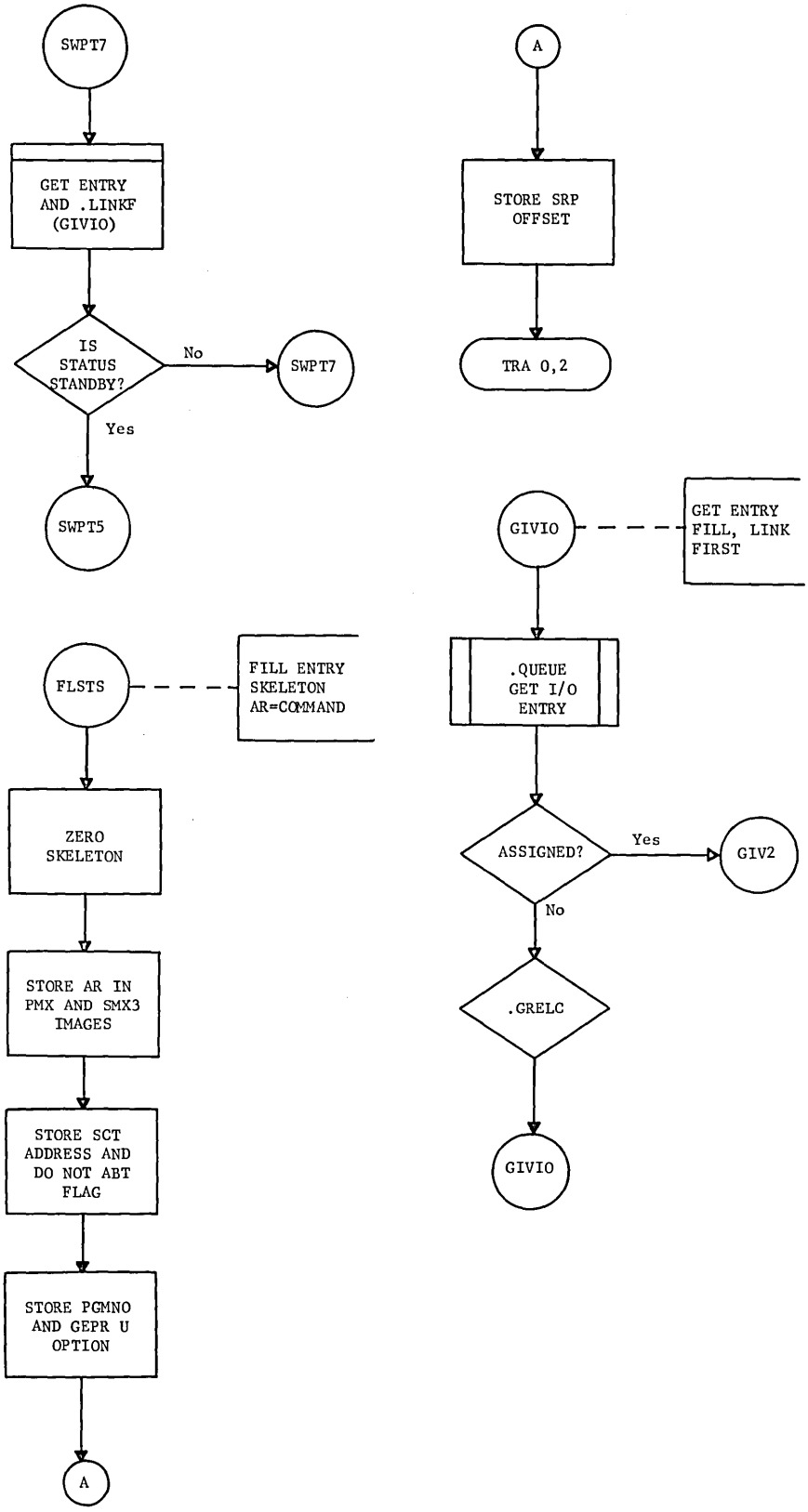

1
FILL ENTRY SKELETON TO REWS (FLSTS)
GET I/O ENTRY AND .LINK (GIVIO)
CHECK STATUS (CKSTS)
(CVTD) CONVERT OLD PHYSICAL ADDRESS TO BCD
STORE IN MESSAGE
SET UP PARAMETER IN AR
.CALL .MALC2,2 ASK FOR READY TAPE
ZERO EXTRA REWIND FLAG
2
2
WAS READY TAPE GIVEN?
No
ACFL2
Yes
BLANK READY WORD IN MESSAGE
SWPT4
EXCHANGE TAPES MARK OLD TAPE NOT ASSIGNED, NEW TAPE ASSIGNED
IS EXTRA REWIND FLAG SET?
No
SWPT5
Yes
FILL ENTRY SKELETON TO REWS (FLSTS)
GET I/O ENTRY AND .LINKI (GIVIO)
CHECK STATUS (CKSTS)
SWPT5

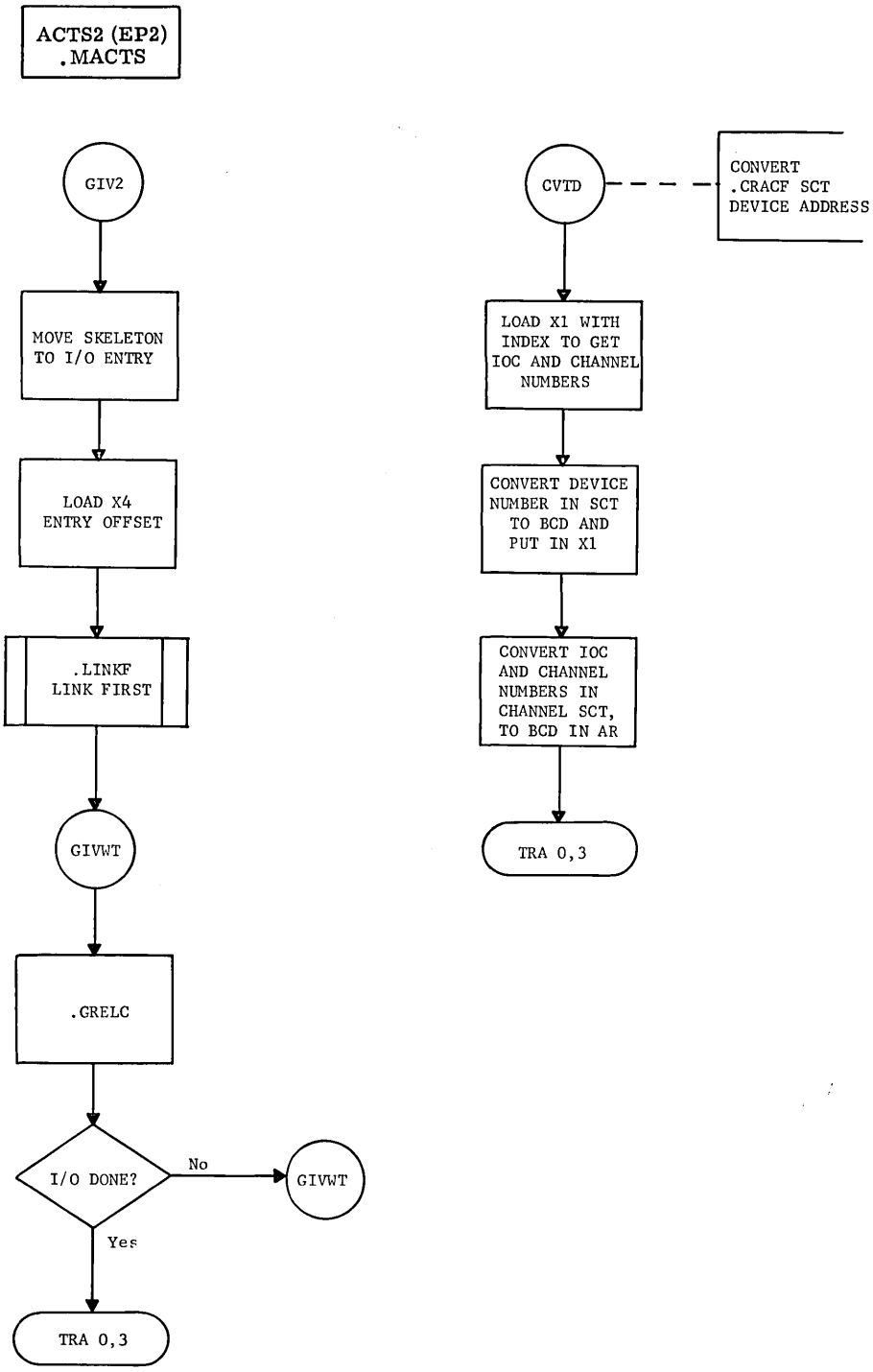

ACTS2 (EP2)
.MACTS
SWPT5
(CVTD) CONVERT NEW PHYSICAL ADDRESS TO BCD
STORE IN MESSAGE
ZERO BLOCK SERIAL NUMBER IN .CRACF
SWP5A
.QUEUE GET AN I/O ENTRY
WAS ENTRY GIVEN?
No
.GRELC
Yes
SWPT9
SWPT9
LOAD QR WORD 0 OF ITYM CALLING SEQUENCE (MESSAGE CODE)
IS MODULE IN SSA?
Yes
INSSA
No
QR MASTER FLAG TO QR (BIT 0)
USE ABSOLUTE DATA ADDRESS (X0)
SWPTA
INSSA
RELATIVIZE DATA ADDRESS FOR SSA (X0)
SWPTA


SWPTA
OR LINK FIRST FLAG TO QR (BIT 1)
STORE XO (DATA ADDRESS) IN DCW
STORE QR IN WORD 0 OF CALLING SEQUENCE
STORE OFFSET OF DCWP AND SRP
LOAD XO ABSOLUTE LOCATION OF CALLING SEQUENCE
.GITYM
SWPTB
.GRELC
IS I/O DONE?
No
SWPTB
Yes
B
SWPT2
.CALL .MALC2,2 ASK FOR STANDBY TAPE
STORE READY IN MESSAGE
WAS STANDBY TAPE GIVEN?
No
SWPT6
Yes
IS TAPE REALLY READY, WRITE INHIBIT?
No
SWPT4
Yes
SET EXTRA REWIND FLAG
SWPT4
SWPT6
SET UP ENTRY SKELETON RESS (FLSTS)
SET GEPR OVERRIDE FLAG
SWPT7

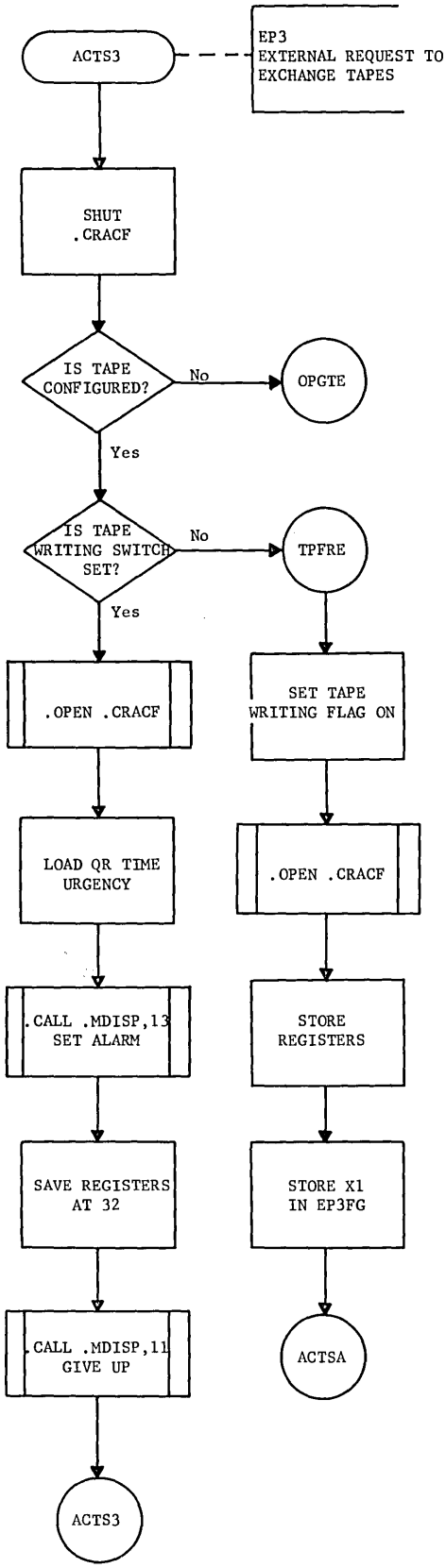
ACTS2 (EP2)
.MACTS




SWPT7
GET ENTRY AND .LINKF (GIVIO)
IS STATUS STANDBY?
No
SWPT7
Yes
SWPT5
FLSTS
FILL ENTRY SKELETON AR=COMMAND
ZERO SKELETON
STORE AR IN PMX AND SMX3 IMAGES
STORE SCT ADDRESS AND DO NOT ABT FLAG
STORE PGMNO AND GEPR U OPTION
A
A
STORE SRP OFFSET
TRA 0,2
GIVIO
GET ENTRY FILL, LINK FIRST
.QUEUE GET I/O ENTRY
ASSIGNED?
Yes
GIV2
No
.GRELC
GIVIO


ACTS2 (EP2)
.MACTS
GIV2
MOVE SKELETON
TO I/O ENTRY
LOAD X4
ENTRY OFFSET
.LINKF
LINK FIRST
GIVWT
.GRELC
I/O DONE?
No
GIVWT
Yes
TRA 0,3
CVTD
CONVERT
.CRACF SCT
DEVICE ADDRESS
LOAD X1 WITH
INDEX TO GET
IOC AND CHANNEL
NUMBERS
CONVERT DEVICE
NUMBER IN SCT
TO BCD AND
PUT IN X1
CONVERT IOC
AND CHANNEL
NUMBERS IN
CHANNEL SCT,
TO BCD IN AR
TRA 0,3

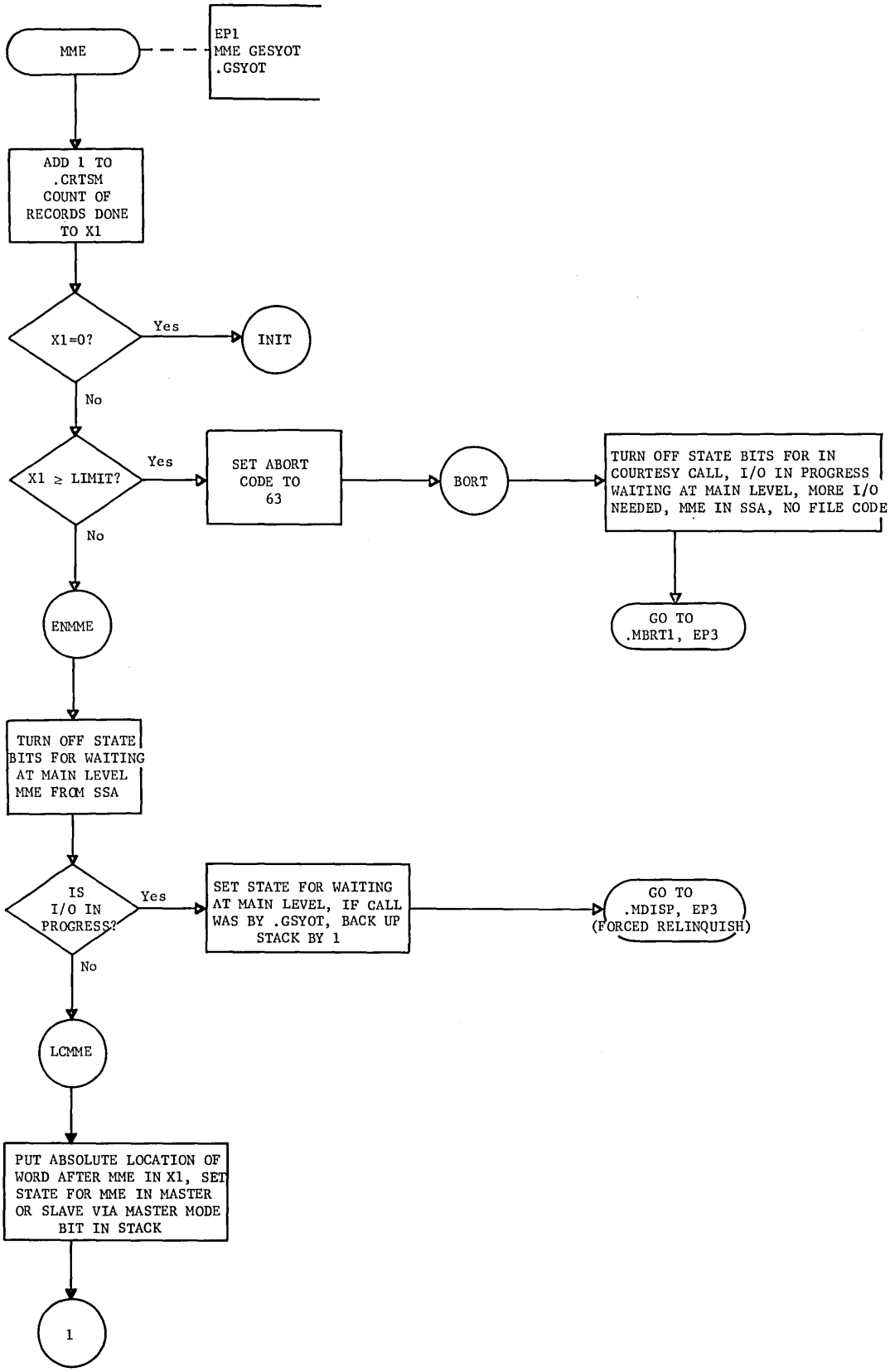
EXTERNAL REQUEST FOR ACCOUNTING FILE

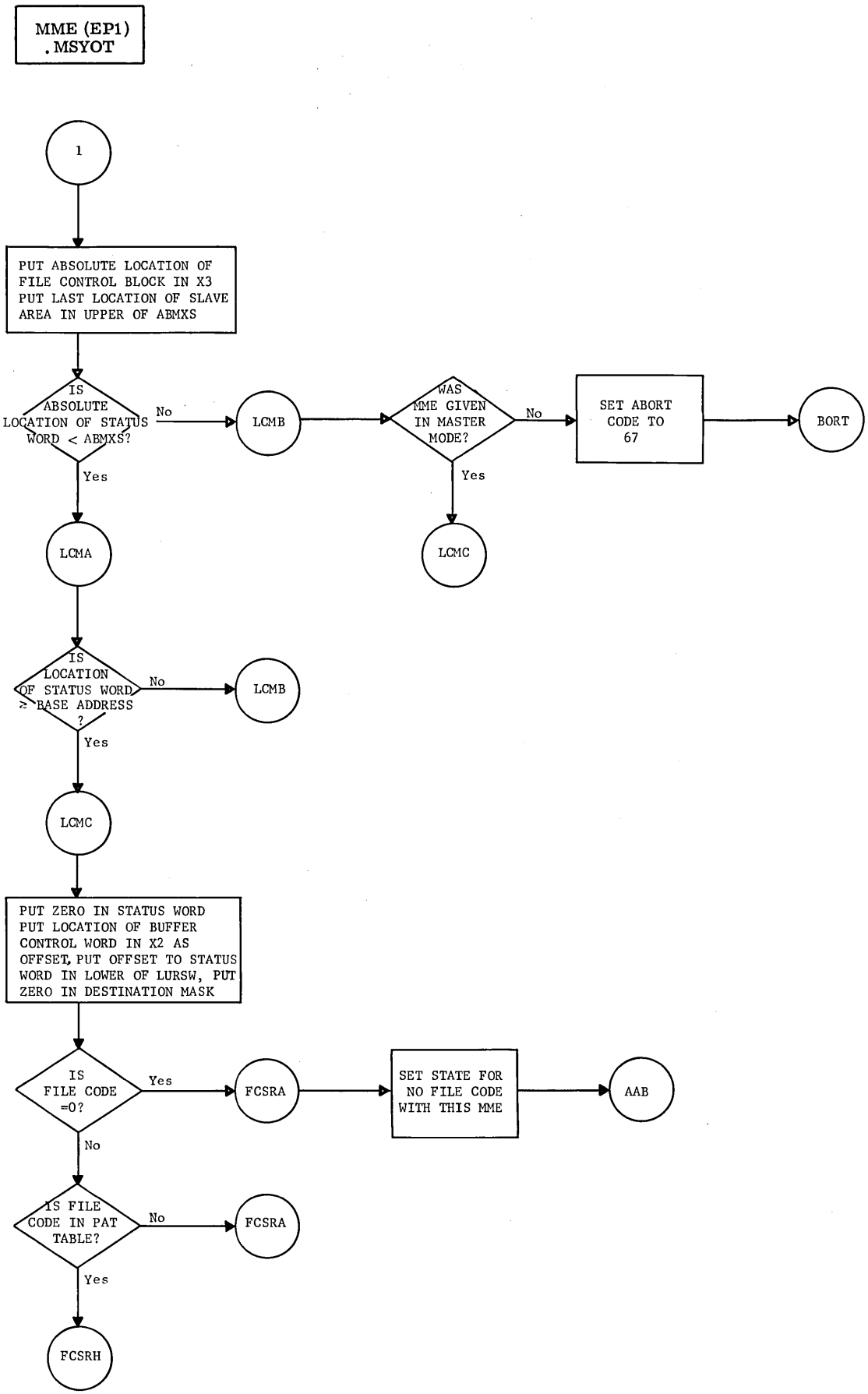


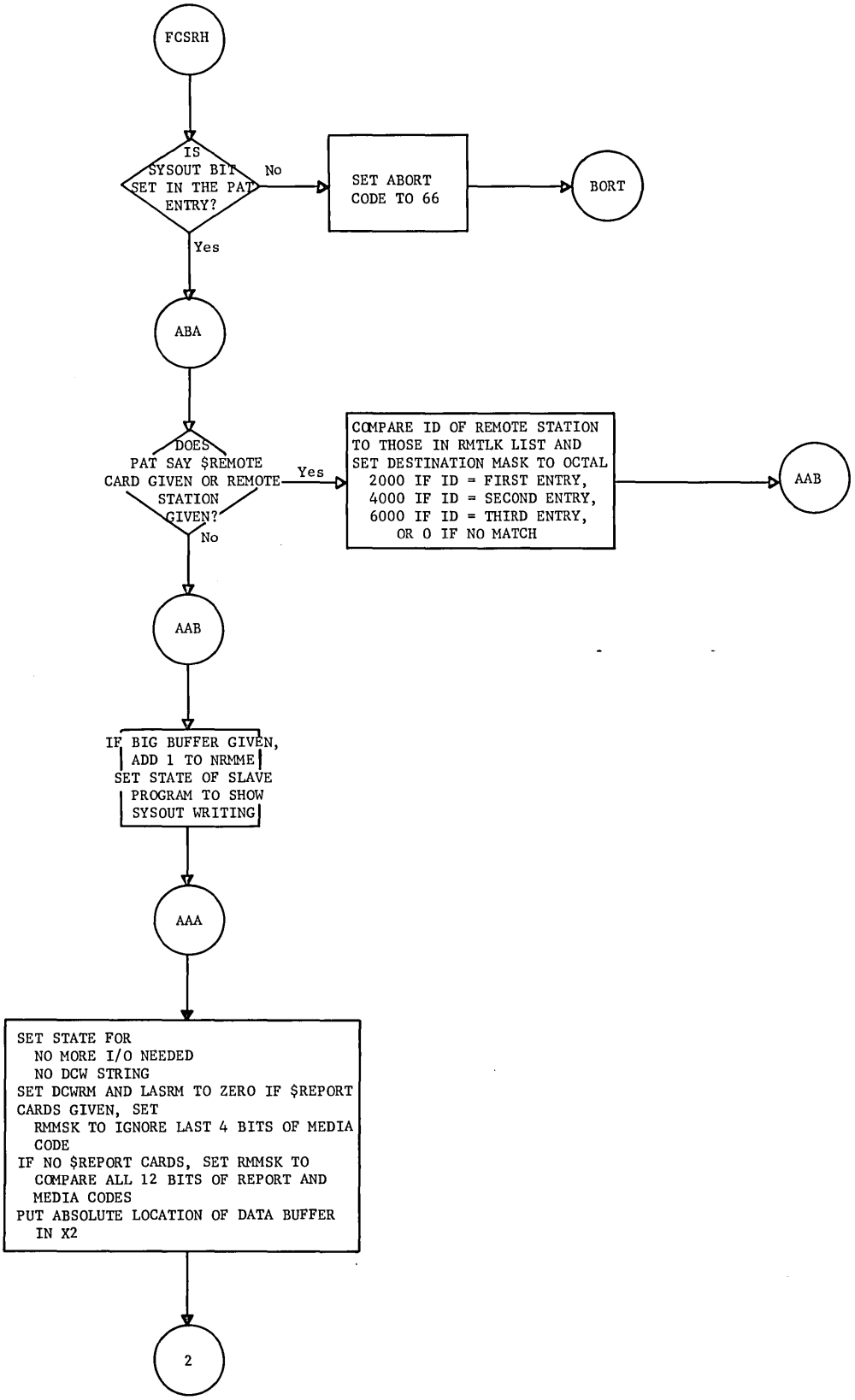
4. TERMINATION AND SYSTEM OUTPUT

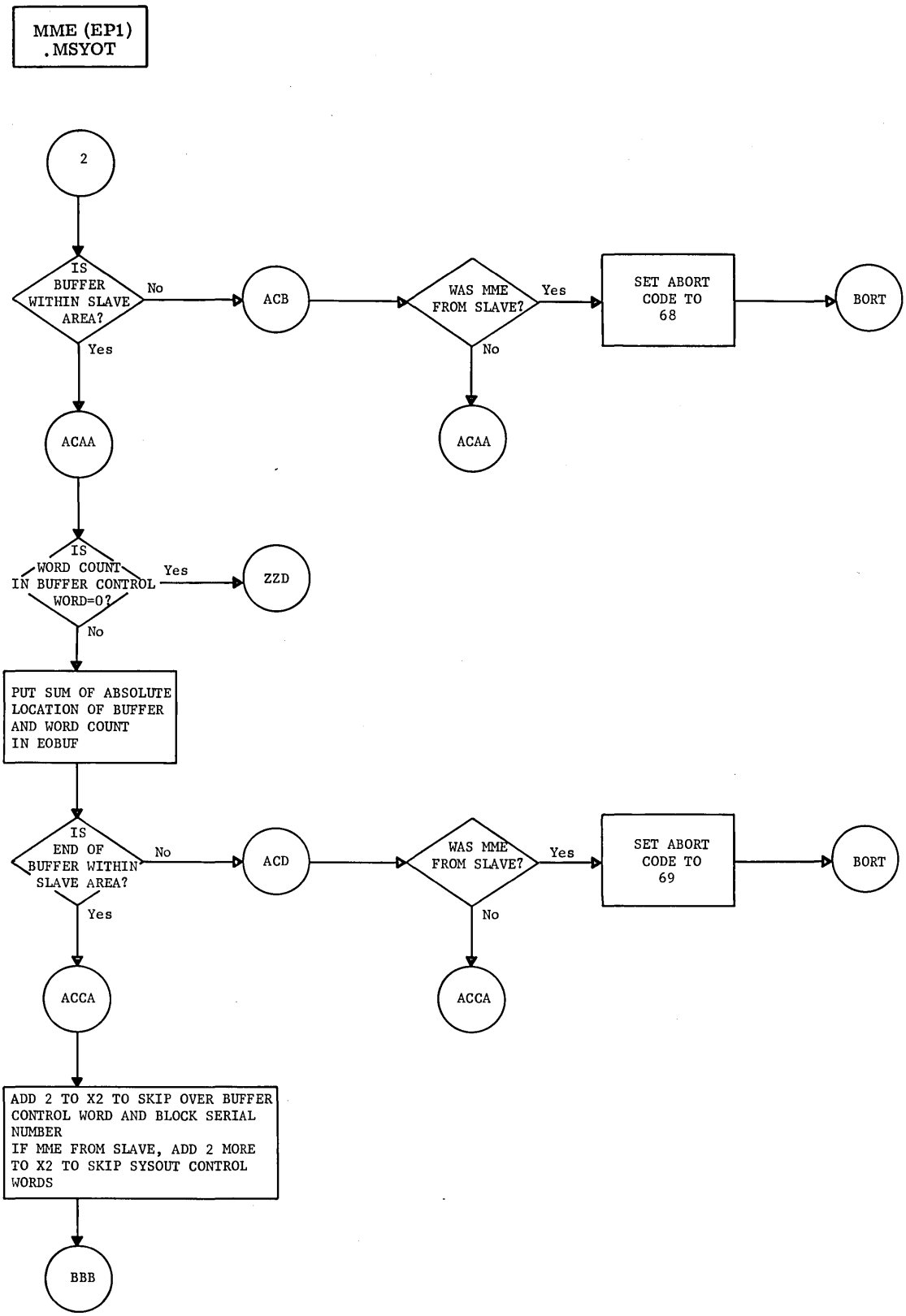
(Reference CPB-1496)

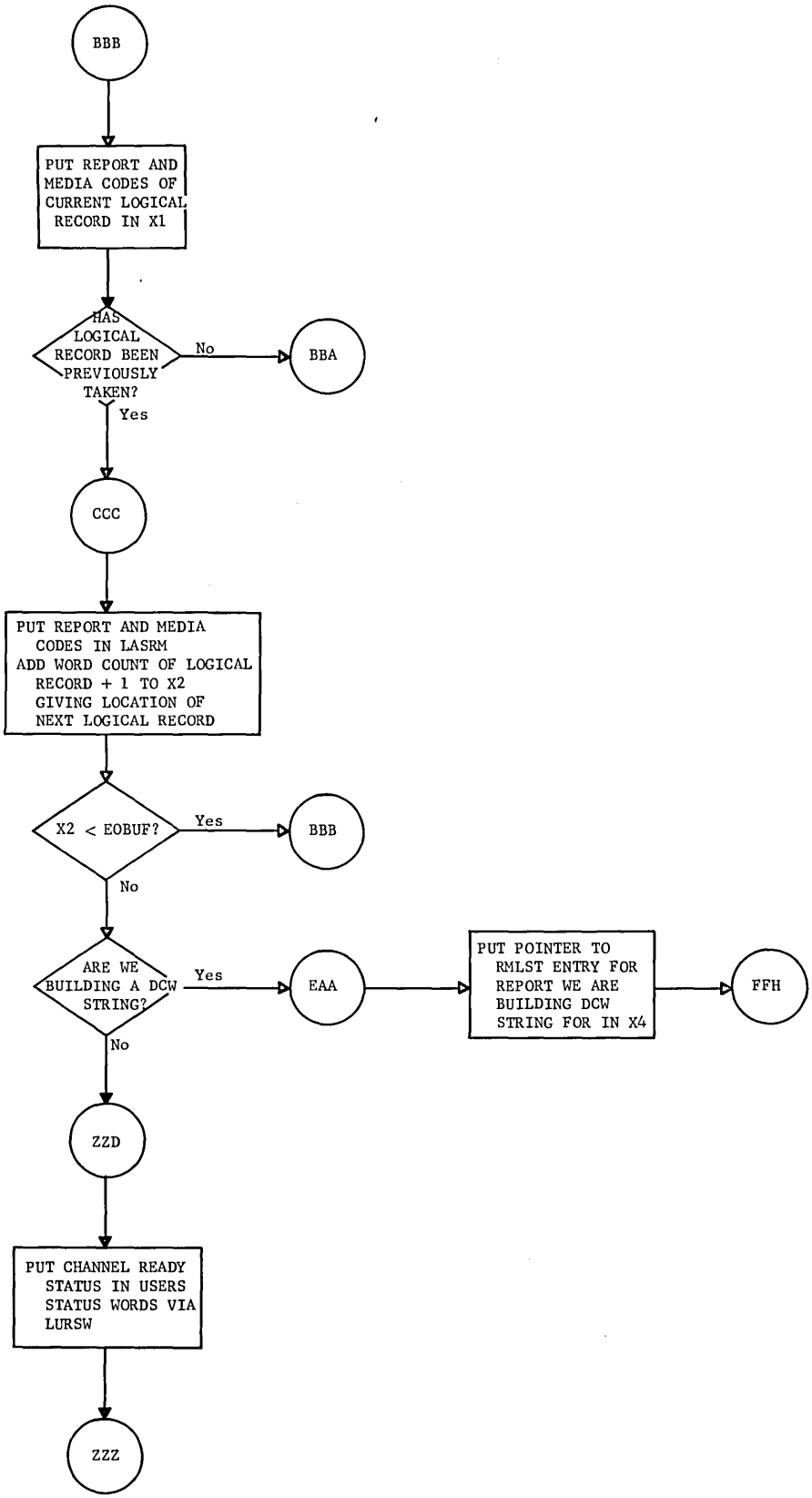
PROCESS MME GESYOT

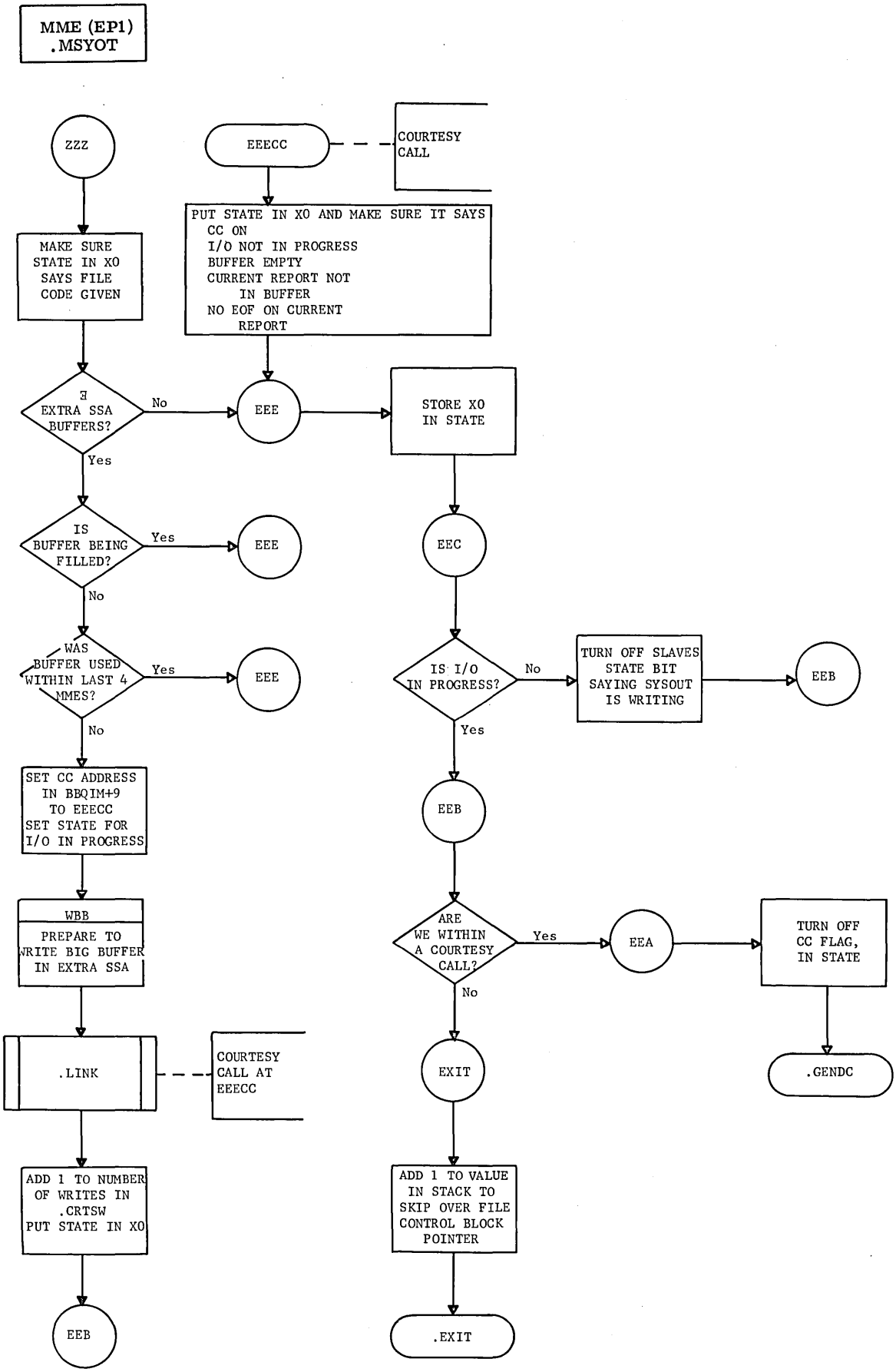


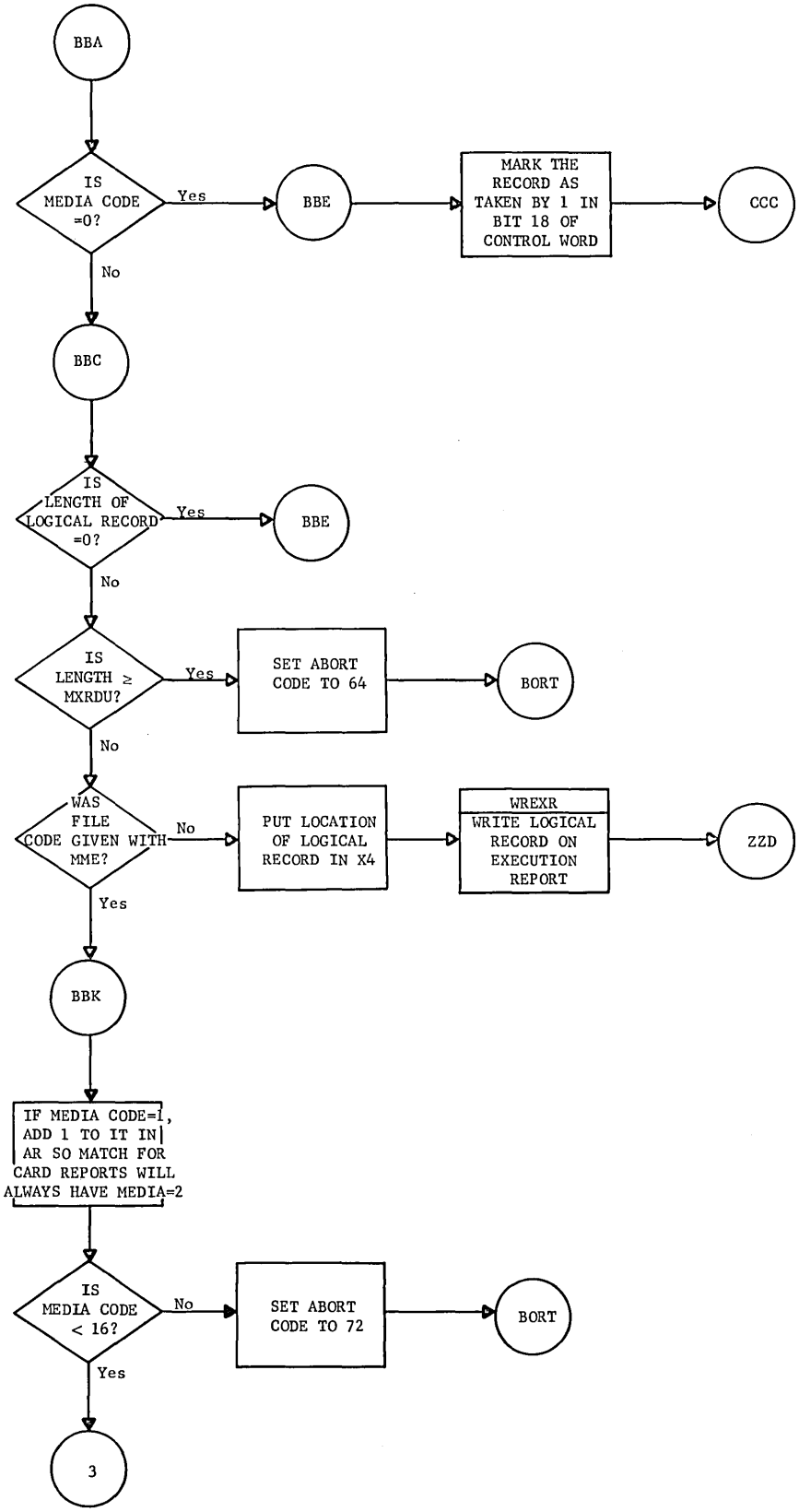

MME (EP1)
.MSYOT
1
PUT ABSOLUTE LOCATION OF FILE CONTROL BLOCK IN X3 PUT LAST LOCATION OF SLAVE AREA IN UPPER OF ABMXS
IS ABSOLUTE LOCATION OF STATUS WORD < ABMXS?
No
LCMB
WAS MME GIVEN IN MASTER MODE?
No
SET ABORT CODE TO 67
BORT
Yes
LCMC
Yes
LCMA
IS LOCATION OF STATUS WORD ≥ BASE ADDRESS ?
No
LCMB
Yes
LCMC
PUT ZERO IN STATUS WORD PUT LOCATION OF BUFFER CONTROL WORD IN X2 AS OFFSET, PUT OFFSET TO STATUS WORD IN LOWER OF LURSW, PUT ZERO IN DESTINATION MASK
IS FILE CODE =0?
Yes
FCSRA
SET STATE FOR NO FILE CODE WITH THIS MME
AAB
No
IS FILE CODE IN PAT TABLE?
No
FCSRA
Yes
FCSRH

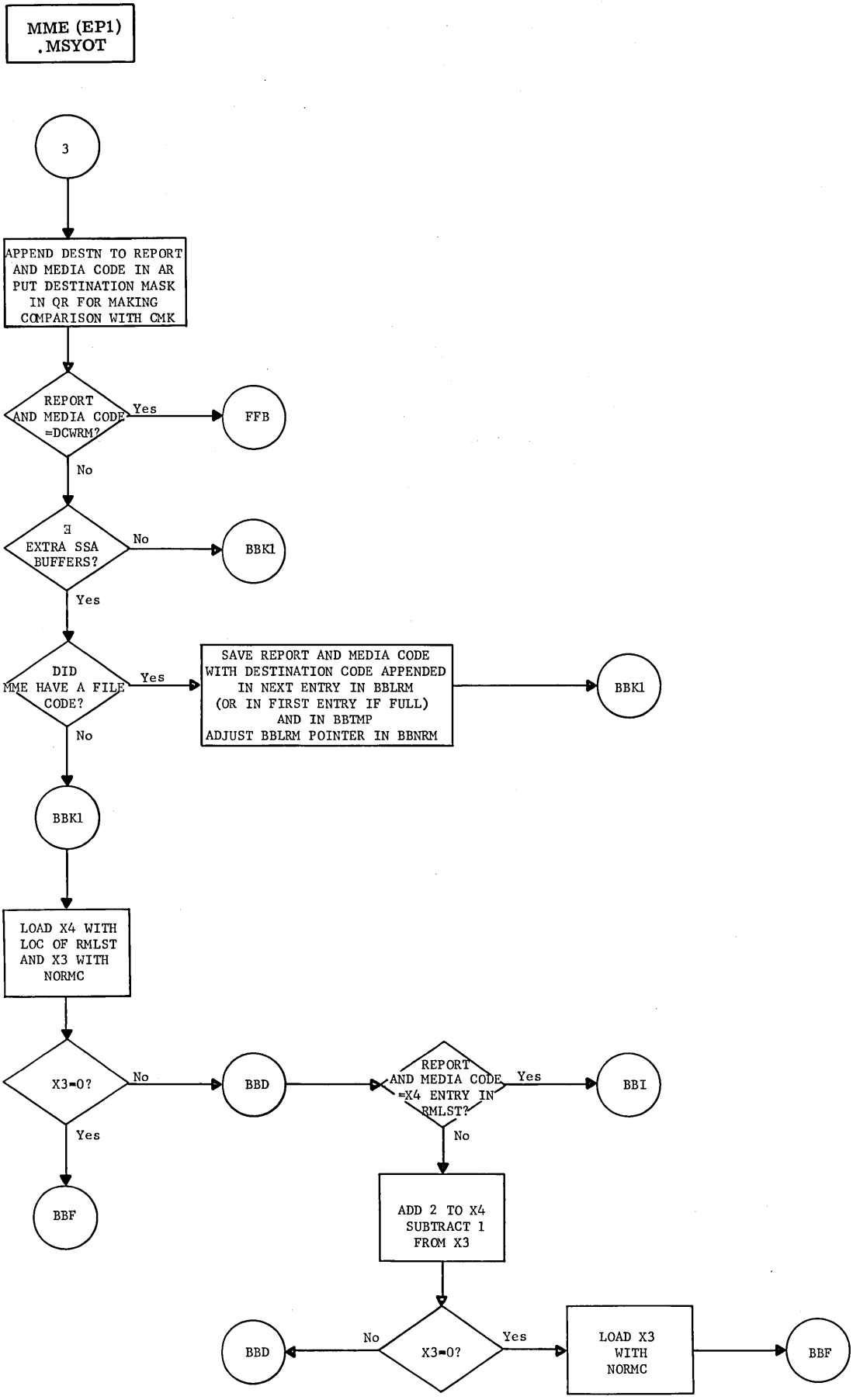

FCSRH
IS SYSOUT BIT SET IN THE PAT ENTRY?
No
SET ABORT CODE TO 66
BORT
Yes
ABA
DOES PAT SAY $REMOTE CARD GIVEN OR REMOTE STATION GIVEN?
Yes
COMPARE ID OF REMOTE STATION TO THOSE IN RMTLK LIST AND SET DESTINATION MASK TO OCTAL
2000 IF ID = FIRST ENTRY,
4000 IF ID = SECOND ENTRY,
6000 IF ID = THIRD ENTRY,
OR 0 IF NO MATCH
AAB
No
AAB
IF BIG BUFFER GIVEN, ADD 1 TO NRMME
SET STATE OF SLAVE PROGRAM TO SHOW SYSOUT WRITING
AAA
SET STATE FOR
NO MORE I/O NEEDED
NO DCW STRING
SET DCWRM AND LASRM TO ZERO IF $REPORT CARDS GIVEN, SET
RMMSK TO IGNORE LAST 4 BITS OF MEDIA CODE
IF NO $REPORT CARDS, SET RMMSK TO
COMPARE ALL 12 BITS OF REPORT AND MEDIA CODES
PUT ABSOLUTE LOCATION OF DATA BUFFER IN X2
2

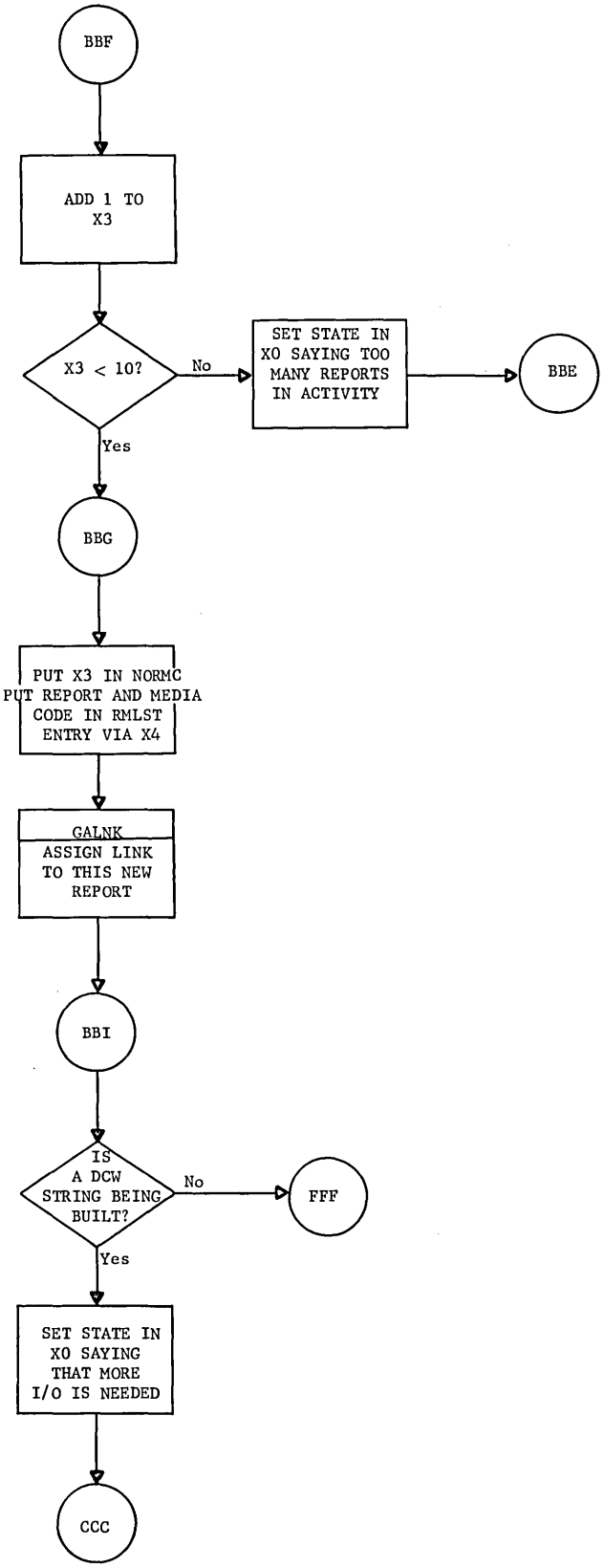

MME (EP1)
.MSYOT
2
IS BUFFER WITHIN SLAVE AREA?
No
ACB
WAS MME FROM SLAVE?
Yes
SET ABORT CODE TO 68
BORT
Yes
No
ACAA
ACAA
IS WORD COUNT IN BUFFER CONTROL WORD=0?
Yes
ZZD
No
PUT SUM OF ABSOLUTE LOCATION OF BUFFER AND WORD COUNT IN EOBUF
IS END OF BUFFER WITHIN SLAVE AREA?
No
ACD
WAS MME FROM SLAVE?
Yes
SET ABORT CODE TO 69
BORT
Yes
No
ACCA
ACCA
ADD 2 TO X2 TO SKIP OVER BUFFER CONTROL WORD AND BLOCK SERIAL NUMBER
IF MME FROM SLAVE, ADD 2 MORE TO X2 TO SKIP SYSOUT CONTROL WORDS
BBB

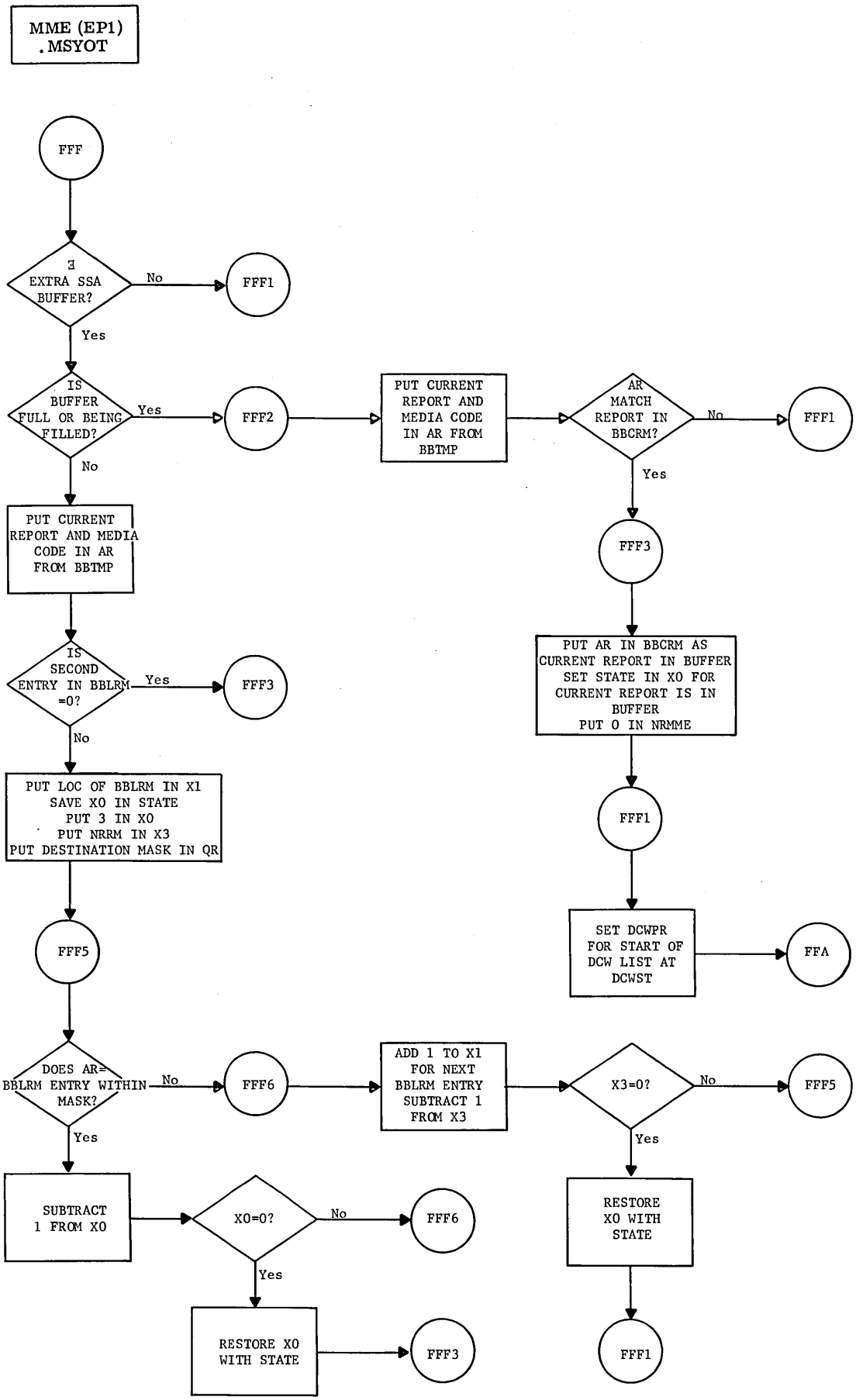

BBB
PUT REPORT AND MEDIA CODES OF CURRENT LOGICAL RECORD IN X1
HAS LOGICAL RECORD BEEN PREVIOUSLY TAKEN?
No
BBA
Yes
CCC
PUT REPORT AND MEDIA CODES IN LASRM
ADD WORD COUNT OF LOGICAL RECORD + 1 TO X2 GIVING LOCATION OF NEXT LOGICAL RECORD
X2 < EOBUF?
Yes
BBB
No
ARE WE BUILDING A DCW STRING?
Yes
EAA
PUT POINTER TO RMLST ENTRY FOR REPORT WE ARE BUILDING DCW STRING FOR IN X4
FFH
No
ZZD
PUT CHANNEL READY STATUS IN USERS STATUS WORDS VIA LURSW
ZZZ

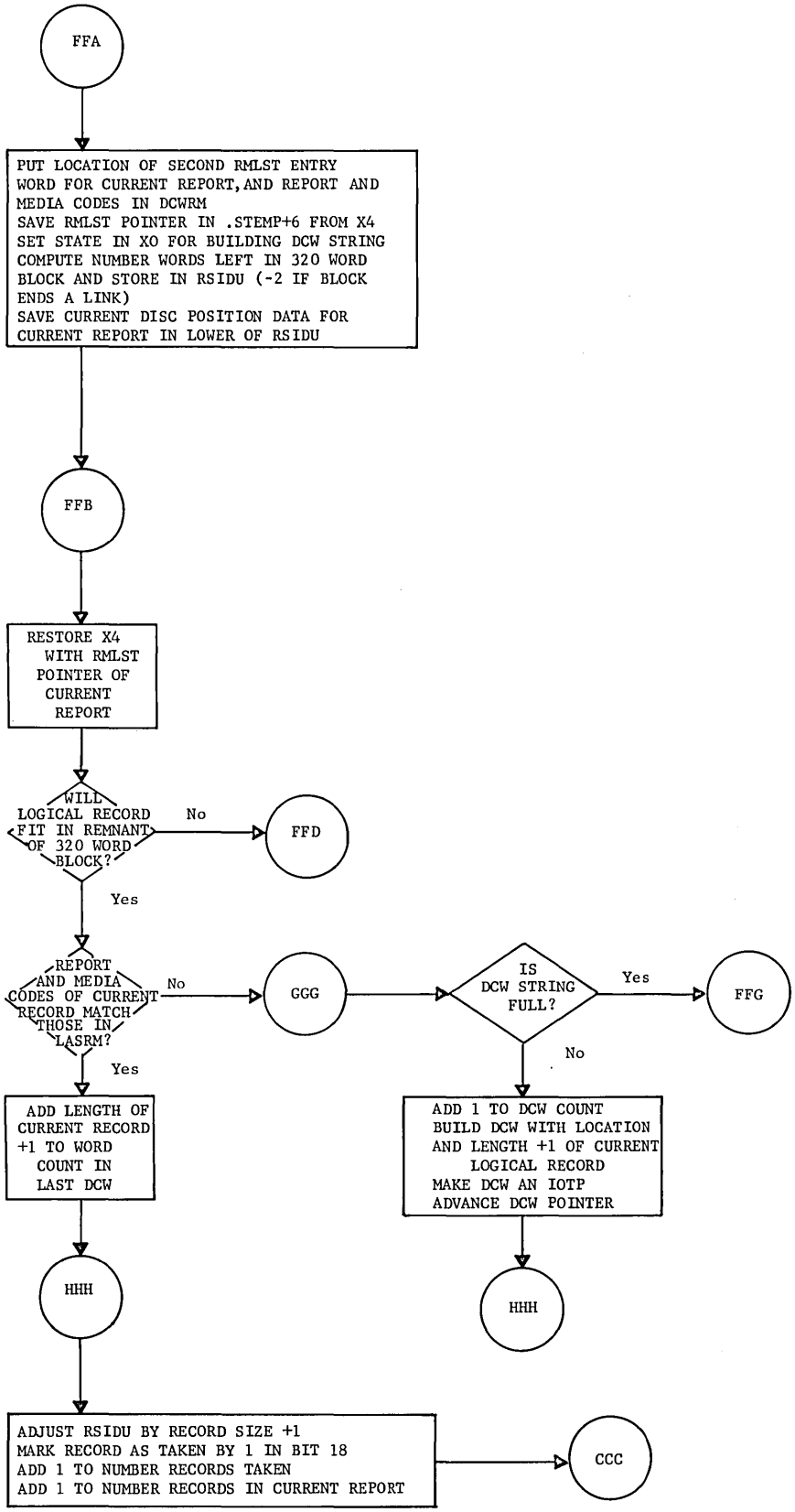

MME (EP1)
.MSYOT
ZZZ
EEECC
COURTESY CALL
PUT STATE IN XO AND MAKE SURE IT SAYS
CC ON
I/O NOT IN PROGRESS
BUFFER EMPTY
CURRENT REPORT NOT IN BUFFER
NO EOF ON CURRENT REPORT
MAKE SURE STATE IN XO SAYS FILE CODE GIVEN
EXTRA SSA BUFFERS?
No
EEE
STORE XO IN STATE
Yes
IS BUFFER BEING FILLED?
Yes
EEE
EEC
No
WAS BUFFER USED WITHIN LAST 4 MMES?
Yes
EEE
IS I/O IN PROGRESS?
No
TURN OFF SLAVES STATE BIT SAYING SYSOUT IS WRITING
EEB
No
Yes
SET CC ADDRESS IN BBQIM+9 TO EEECC
SET STATE FOR I/O IN PROGRESS
EEB
WBB
PREPARE TO WRITE BIG BUFFER IN EXTRA SSA
ARE WE WITHIN A COURTESY CALL?
Yes
EEA
TURN OFF CC FLAG, IN STATE
No
.LINK
COURTESY CALL AT EEECC
EXIT
.GENDC
ADD 1 TO NUMBER OF WRITES IN .CRTSW
PUT STATE IN XO
ADD 1 TO VALUE IN STACK TO SKIP OVER FILE CONTROL BLOCK POINTER
EEB
.EXIT

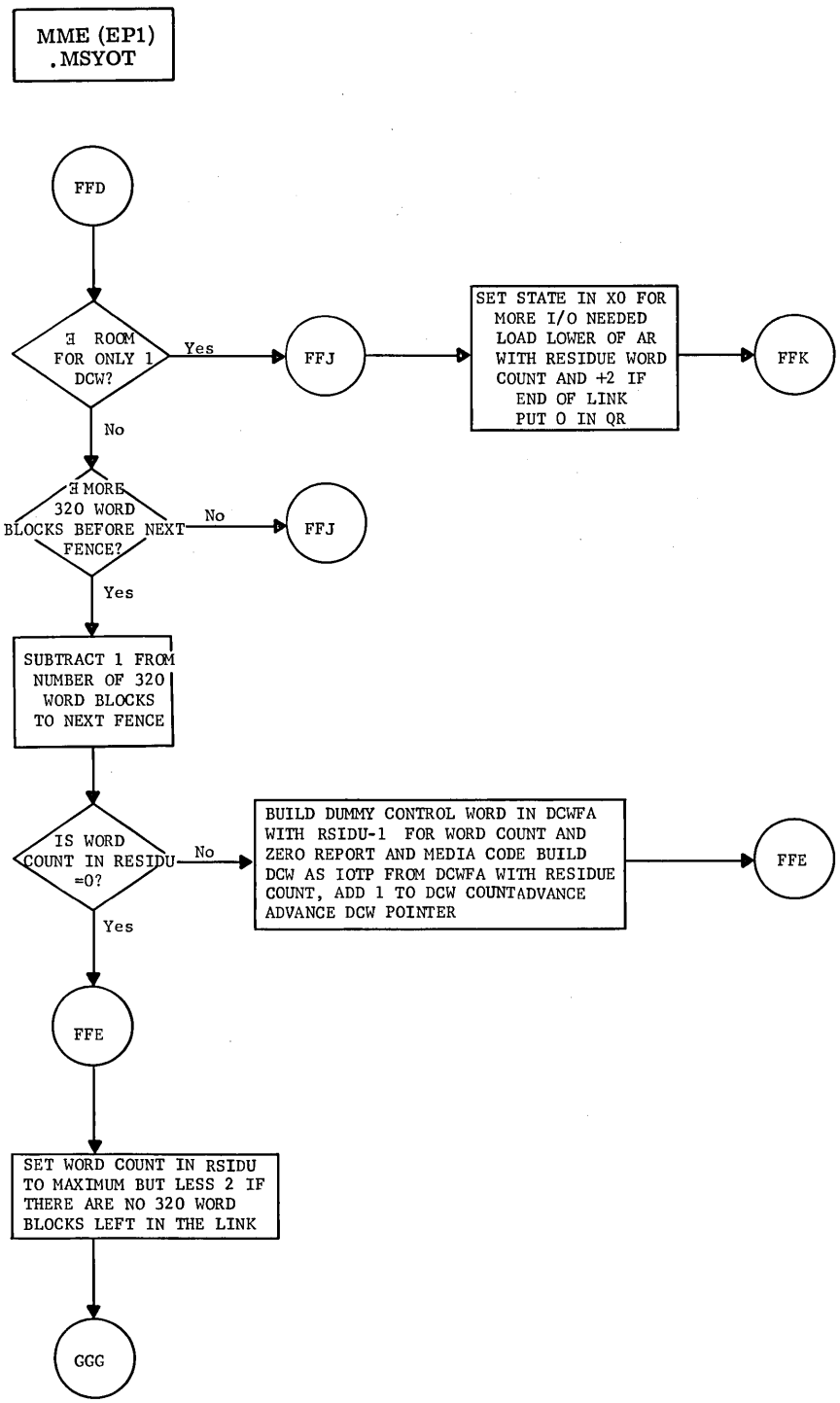

BBA
IS MEDIA CODE =0?
Yes
BBE
MARK THE RECORD AS TAKEN BY 1 IN BIT 18 OF CONTROL WORD
CCC
No
BBC
IS LENGTH OF LOGICAL RECORD =0?
Yes
BBE
No
IS LENGTH ≥ MXRDU?
Yes
SET ABORT CODE TO 64
BORT
No
WAS FILE CODE GIVEN WITH MME?
No
PUT LOCATION OF LOGICAL RECORD IN X4
WREXR
WRITE LOGICAL RECORD ON EXECUTION REPORT
ZZD
Yes
BBK
IF MEDIA CODE=1, ADD 1 TO IT IN AR SO MATCH FOR CARD REPORTS WILL ALWAYS HAVE MEDIA=2
IS MEDIA CODE < 16?
No
SET ABORT CODE TO 72
BORT
Yes
3

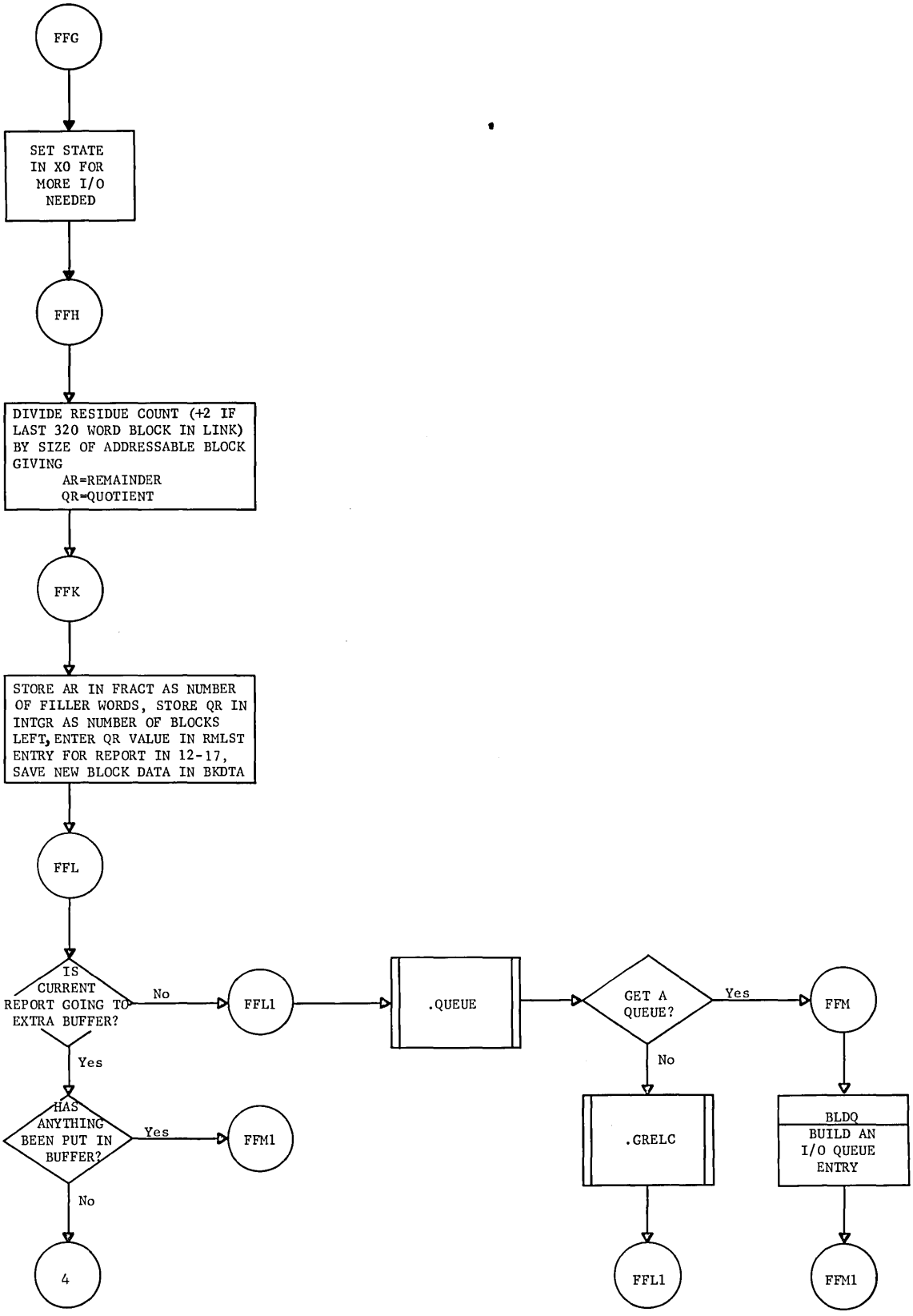

MME (EP1)
.MSYOT
3
APPEND DESTN TO REPORT AND MEDIA CODE IN AR PUT DESTINATION MASK IN QR FOR MAKING COMPARISON WITH CMK
REPORT AND MEDIA CODE =DCWRM?
Yes
FFB
No
∃ EXTRA SSA BUFFERS?
No
BBK1
Yes
DID MME HAVE A FILE CODE?
Yes
SAVE REPORT AND MEDIA CODE WITH DESTINATION CODE APPENDED IN NEXT ENTRY IN BBLRM (OR IN FIRST ENTRY IF FULL) AND IN BBTMP ADJUST BBLRM POINTER IN BBNRM
BBK1
No
BBK1
LOAD X4 WITH LOC OF RMLST AND X3 WITH NORMC
X3=0?
No
BBD
REPORT AND MEDIA CODE =X4 ENTRY IN RMLST?
Yes
BBI
Yes
BBF
No
ADD 2 TO X4 SUBTRACT 1 FROM X3
BBD
No
X3=0?
Yes
LOAD X3 WITH NORMC
BBF

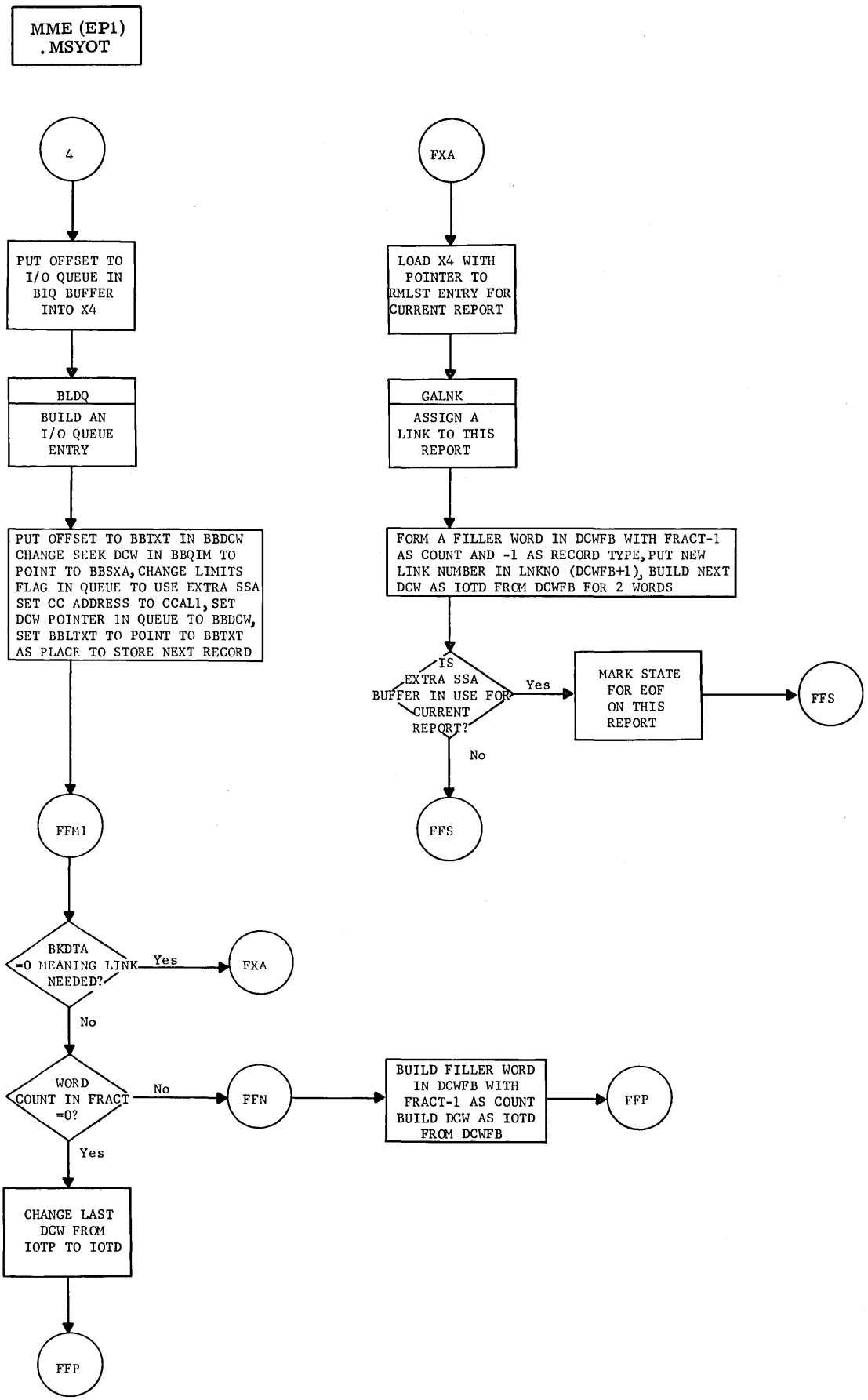

BBF
ADD 1 TO X3
X3 < 10?
No
SET STATE IN X0 SAYING TOO MANY REPORTS IN ACTIVITY
BBE
Yes
BBG
PUT X3 IN NORMC PUT REPORT AND MEDIA CODE IN RMLST ENTRY VIA X4
GALNK
ASSIGN LINK TO THIS NEW REPORT
BBI
IS A DCW STRING BEING BUILT?
No
FFF
Yes
SET STATE IN X0 SAYING THAT MORE I/O IS NEEDED
CCC

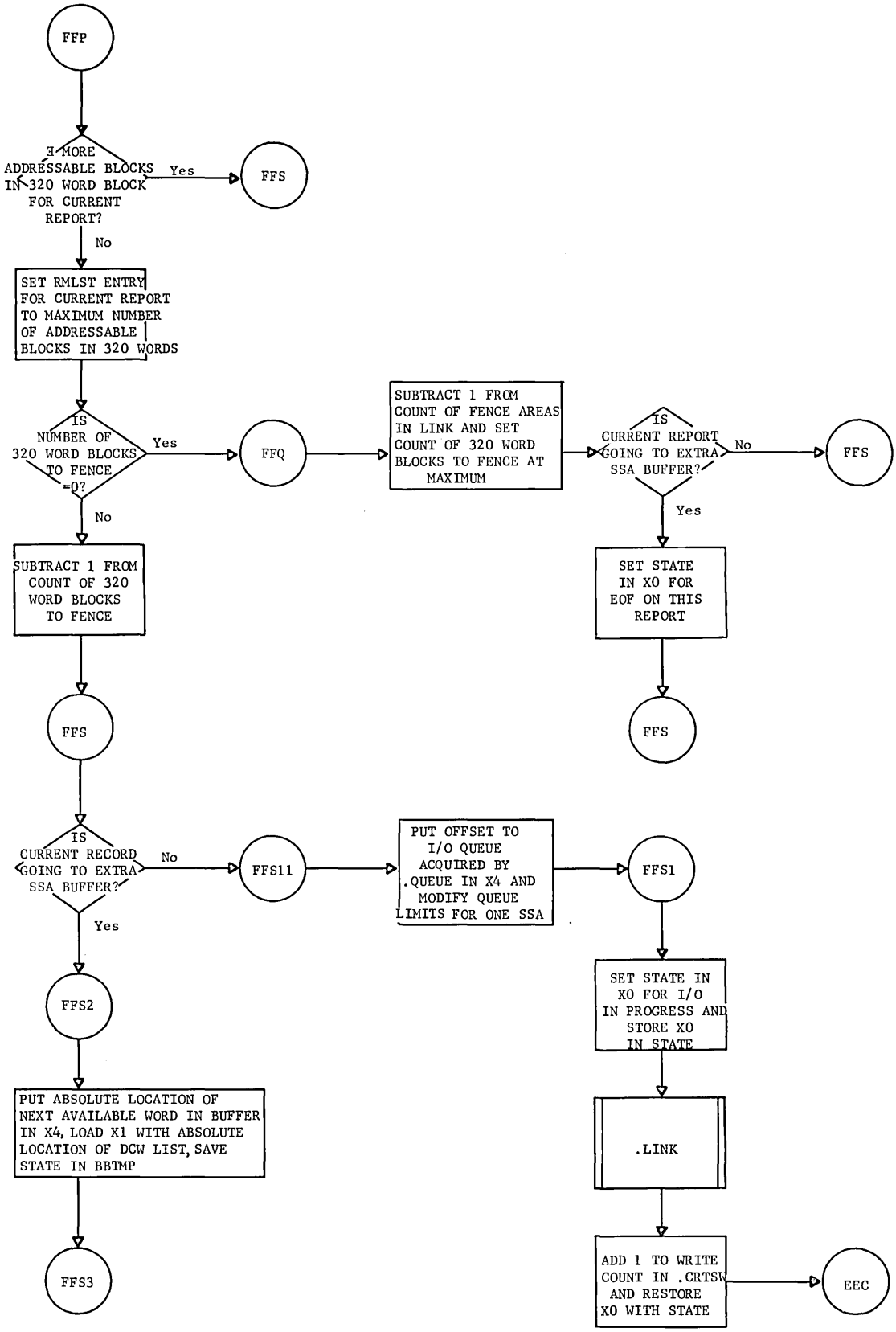

MME (EP1)
.MSYOT
FFF
∃ EXTRA SSA BUFFER?
No
FFF1
Yes
IS BUFFER FULL OR BEING FILLED?
Yes
FFF2
PUT CURRENT REPORT AND MEDIA CODE IN AR FROM BBTMP
AR MATCH REPORT IN BBCRM?
No
FFF1
Yes
FFF3
No
PUT CURRENT REPORT AND MEDIA CODE IN AR FROM BBTMP
IS SECOND ENTRY IN BBLRM =0?
Yes
FFF3
No
PUT AR IN BBCRM AS CURRENT REPORT IN BUFFER SET STATE IN X0 FOR CURRENT REPORT IS IN BUFFER PUT 0 IN NRMME
FFF1
SET DCWPR FOR START OF DCW LIST AT DCWST
FFA
PUT LOC OF BBLRM IN X1 SAVE X0 IN STATE PUT 3 IN X0 PUT NRRM IN X3 PUT DESTINATION MASK IN QR
FFF5
DOES AR= BBLRM ENTRY WITHIN MASK?
No
FFF6
ADD 1 TO X1 FOR NEXT BBLRM ENTRY SUBTRACT 1 FROM X3
X3=0?
No
FFF5
Yes
Yes
SUBTRACT 1 FROM X0
X0=0?
No
FFF6
Yes
RESTORE X0 WITH STATE
FFF3
RESTORE X0 WITH STATE
FFF1

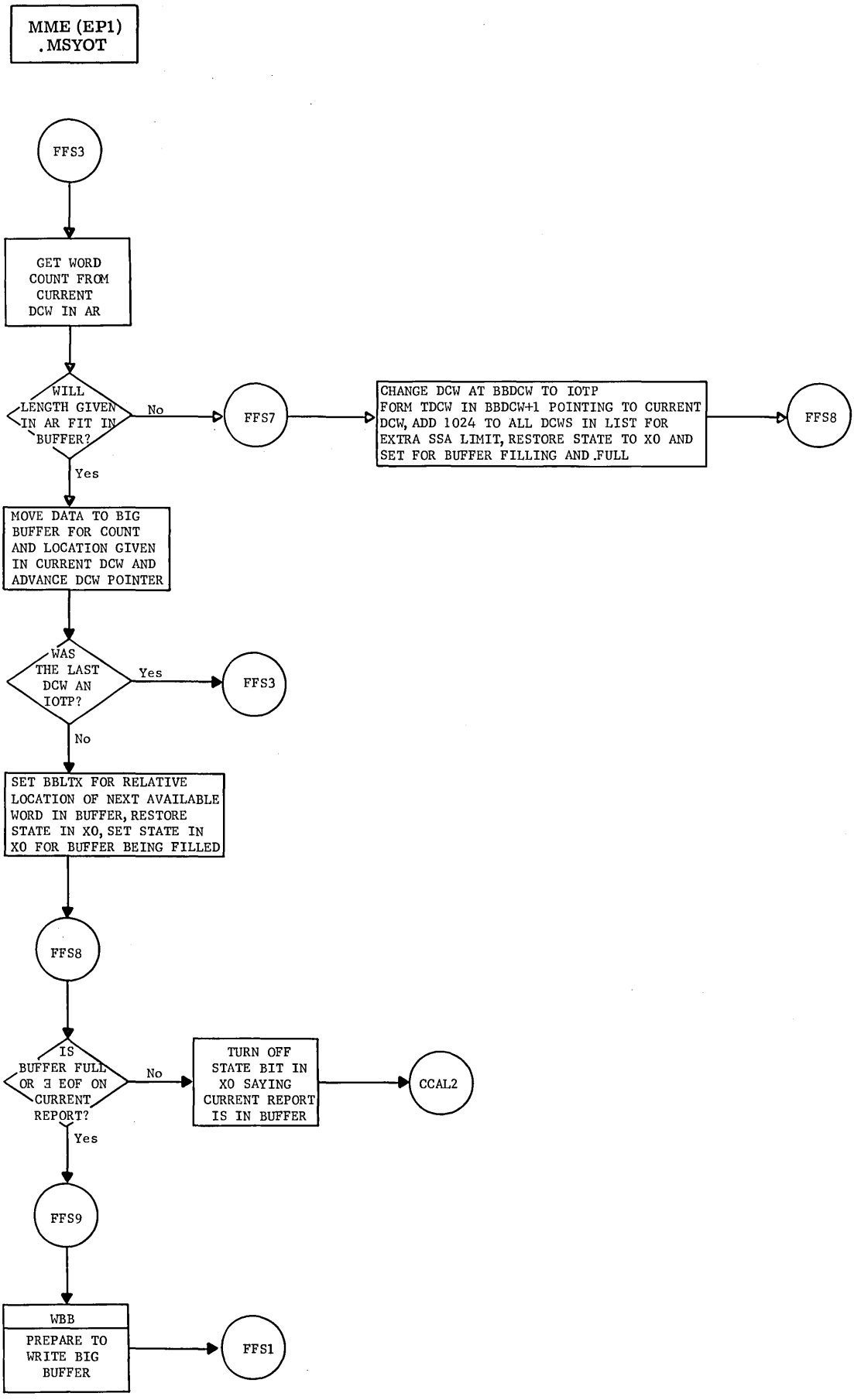

FFA
PUT LOCATION OF SECOND RMLST ENTRY WORD FOR CURRENT REPORT, AND REPORT AND MEDIA CODES IN DCWRM
SAVE RMLST POINTER IN .STEMP+6 FROM X4
SET STATE IN XO FOR BUILDING DCW STRING
COMPUTE NUMBER WORDS LEFT IN 320 WORD BLOCK AND STORE IN RSIDU (-2 IF BLOCK ENDS A LINK)
SAVE CURRENT DISC POSITION DATA FOR CURRENT REPORT IN LOWER OF RSIDU
FFB
RESTORE X4 WITH RMLST POINTER OF CURRENT REPORT
WILL LOGICAL RECORD FIT IN REMNANT OF 320 WORD BLOCK?
No
FFD
Yes
REPORT AND MEDIA CODES OF CURRENT RECORD MATCH THOSE IN LASRM?
No
GGG
IS DCW STRING FULL?
Yes
FFG
No
Yes
ADD LENGTH OF CURRENT RECORD +1 TO WORD COUNT IN LAST DCW
ADD 1 TO DCW COUNT
BUILD DCW WITH LOCATION AND LENGTH +1 OF CURRENT LOGICAL RECORD
MAKE DCW AN IOTP
ADVANCE DCW POINTER
HHH
HHH
ADJUST RSIDU BY RECORD SIZE +1
MARK RECORD AS TAKEN BY 1 IN BIT 18
ADD 1 TO NUMBER RECORDS TAKEN
ADD 1 TO NUMBER RECORDS IN CURRENT REPORT
CCC

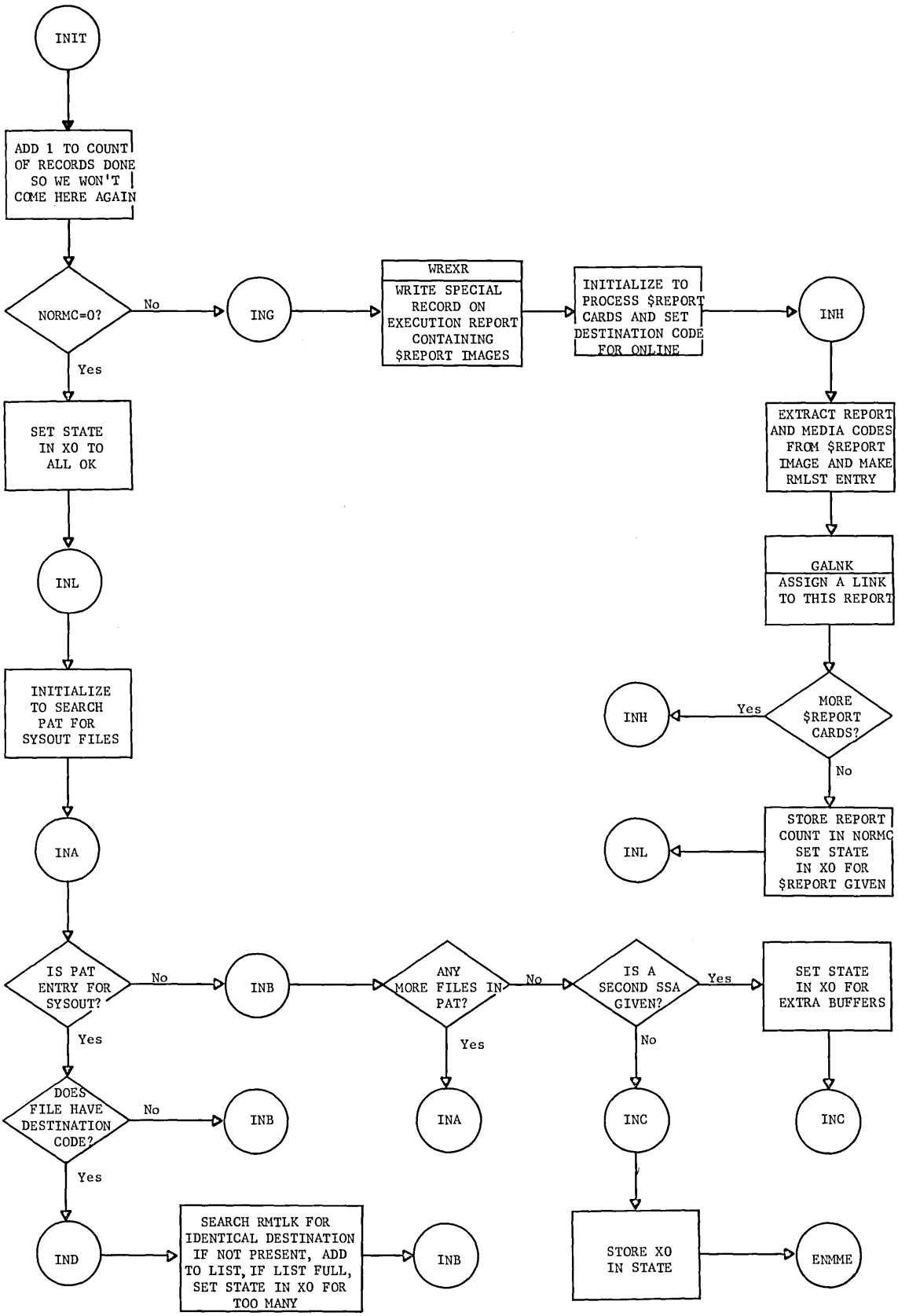

MME (EP1)
.MSYOT
FFD
∃ ROOM FOR ONLY 1 DCW?
Yes
FFJ
SET STATE IN XO FOR MORE I/O NEEDED LOAD LOWER OF AR WITH RESIDUE WORD COUNT AND +2 IF END OF LINK PUT 0 IN QR
FFK
No
∃ MORE 320 WORD BLOCKS BEFORE NEXT FENCE?
No
FFJ
Yes
SUBTRACT 1 FROM NUMBER OF 320 WORD BLOCKS TO NEXT FENCE
IS WORD COUNT IN RESIDU =0?
No
BUILD DUMMY CONTROL WORD IN DCWFA WITH RSIDU-1 FOR WORD COUNT AND ZERO REPORT AND MEDIA CODE BUILD DCW AS IOTP FROM DCWFA WITH RESIDUE COUNT, ADD 1 TO DCW COUNTADVANCE ADVANCE DCW POINTER
FFE
Yes
FFE
SET WORD COUNT IN RSIDU TO MAXIMUM BUT LESS 2 IF THERE ARE NO 320 WORD BLOCKS LEFT IN THE LINK
GGG


FFG
SET STATE IN XO FOR MORE I/O NEEDED
FFH
DIVIDE RESIDUE COUNT (+2 IF LAST 320 WORD BLOCK IN LINK) BY SIZE OF ADDRESSABLE BLOCK GIVING
AR=REMAINDER
QR=QUOTIENT
FFK
STORE AR IN FRACT AS NUMBER OF FILLER WORDS, STORE QR IN INTGR AS NUMBER OF BLOCKS LEFT, ENTER QR VALUE IN RMLST ENTRY FOR REPORT IN 12-17, SAVE NEW BLOCK DATA IN BKDTA
FFL
IS CURRENT REPORT GOING TO EXTRA BUFFER?
No
FFL1
.QUEUE
GET A QUEUE?
Yes
FFM
No
Yes
HAS ANYTHING BEEN PUT IN BUFFER?
Yes
FFM1
No
.GRELC
BLDQ
BUILD AN I/O QUEUE ENTRY
4
FFL1
FFM1


MME (EP1)
.MSYOT
4
PUT OFFSET TO I/O QUEUE IN BIQ BUFFER INTO X4
BLDQ
BUILD AN I/O QUEUE ENTRY
PUT OFFSET TO BBTXT IN BBDCW CHANGE SEEK DCW IN BBQIM TO POINT TO BBSXA, CHANGE LIMITS FLAG IN QUEUE TO USE EXTRA SSA SET CC ADDRESS TO CCAL1, SET DCW POINTER IN QUEUE TO BBDCW, SET BBLTXT TO POINT TO BBTXT AS PLACE TO STORE NEXT RECORD
FFM1
BKDTA =0 MEANING LINK NEEDED?
Yes
FXA
No
WORD COUNT IN FRACT =0?
No
FFN
BUILD FILLER WORD IN DCWFB WITH FRACT-1 AS COUNT BUILD DCW AS IOTD FROM DCWFB
FFP
Yes
CHANGE LAST DCW FROM IOTP TO IOTD
FFP
FXA
LOAD X4 WITH POINTER TO RMLST ENTRY FOR CURRENT REPORT
GALNK
ASSIGN A LINK TO THIS REPORT
FORM A FILLER WORD IN DCWFB WITH FRACT-1 AS COUNT AND -1 AS RECORD TYPE, PUT NEW LINK NUMBER IN LNKNO (DCWFB+1), BUILD NEXT DCW AS IOTD FROM DCWFB FOR 2 WORDS
IS EXTRA SSA BUFFER IN USE FOR CURRENT REPORT?
Yes
MARK STATE FOR EOF ON THIS REPORT
FFS
No
FFS

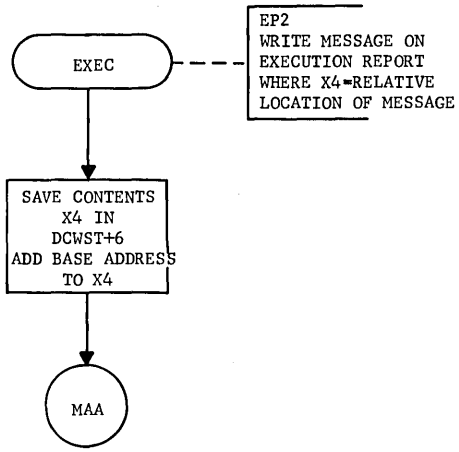

FFP
IS MORE ADDRESSABLE BLOCKS IN 320 WORD BLOCK FOR CURRENT REPORT?
Yes
FFS
No
SET RMLST ENTRY FOR CURRENT REPORT TO MAXIMUM NUMBER OF ADDRESSABLE BLOCKS IN 320 WORDS
IS NUMBER OF 320 WORD BLOCKS TO FENCE =0?
Yes
FFQ
SUBTRACT 1 FROM COUNT OF FENCE AREAS IN LINK AND SET COUNT OF 320 WORD BLOCKS TO FENCE AT MAXIMUM
IS CURRENT REPORT GOING TO EXTRA SSA BUFFER?
No
FFS
Yes
SET STATE IN XO FOR EOF ON THIS REPORT
FFS
No
SUBTRACT 1 FROM COUNT OF 320 WORD BLOCKS TO FENCE
FFS
IS CURRENT RECORD GOING TO EXTRA SSA BUFFER?
No
FFS11
PUT OFFSET TO I/O QUEUE ACQUIRED BY .QUEUE IN X4 AND MODIFY QUEUE LIMITS FOR ONE SSA
FFS1
Yes
SET STATE IN XO FOR I/O IN PROGRESS AND STORE XO IN STATE
FFS2
.LINK
PUT ABSOLUTE LOCATION OF NEXT AVAILABLE WORD IN BUFFER IN X4, LOAD X1 WITH ABSOLUTE LOCATION OF DCW LIST, SAVE STATE IN BBTMP
ADD 1 TO WRITE COUNT IN .CRTSW AND RESTORE XO WITH STATE
EEC
FFS3


MME (EP1)
.MSYOT
FFS3
GET WORD COUNT FROM CURRENT DCW IN AR
WILL LENGTH GIVEN IN AR FIT IN BUFFER?
No
FFS7
CHANGE DCW AT BBDCW TO IOTP
FORM TDCW IN BBDCW+1 POINTING TO CURRENT DCW, ADD 1024 TO ALL DCWS IN LIST FOR EXTRA SSA LIMIT, RESTORE STATE TO X0 AND SET FOR BUFFER FILLING AND .FULL
FFS8
Yes
MOVE DATA TO BIG BUFFER FOR COUNT AND LOCATION GIVEN IN CURRENT DCW AND ADVANCE DCW POINTER
WAS THE LAST DCW AN IOTP?
Yes
FFS3
No
SET BBLTX FOR RELATIVE LOCATION OF NEXT AVAILABLE WORD IN BUFFER, RESTORE STATE IN X0, SET STATE IN X0 FOR BUFFER BEING FILLED
FFS8
IS BUFFER FULL OR ∃ EOF ON CURRENT REPORT?
No
TURN OFF STATE BIT IN X0 SAYING CURRENT REPORT IS IN BUFFER
CCAL2
Yes
FFS9
WBB
PREPARE TO WRITE BIG BUFFER
FFS1


INIT
ADD 1 TO COUNT OF RECORDS DONE SO WE WON'T COME HERE AGAIN
NORMC=0?
No
ING
WREXR
WRITE SPECIAL RECORD ON EXECUTION REPORT CONTAINING $REPORT IMAGES
INITIALIZE TO PROCESS $REPORT CARDS AND SET DESTINATION CODE FOR ONLINE
INH
Yes
SET STATE IN XO TO ALL OK
EXTRACT REPORT AND MEDIA CODES FROM $REPORT IMAGE AND MAKE RMLST ENTRY
GALNK
ASSIGN A LINK TO THIS REPORT
INL
INITIALIZE TO SEARCH PAT FOR SYSOUT FILES
INH
Yes
MORE $REPORT CARDS?
No
INA
INL
STORE REPORT COUNT IN NORMC SET STATE IN XO FOR $REPORT GIVEN
IS PAT ENTRY FOR SYSOUT?
No
INB
ANY MORE FILES IN PAT?
No
IS A SECOND SSA GIVEN?
Yes
SET STATE IN XO FOR EXTRA BUFFERS
Yes
Yes
No
DOES FILE HAVE DESTINATION CODE?
No
INB
INA
INC
INC
Yes
IND
SEARCH RMTLK FOR IDENTICAL DESTINATION IF NOT PRESENT, ADD TO LIST, IF LIST FULL, SET STATE IN XO FOR TOO MANY
INB
STORE XO IN STATE
ENMME

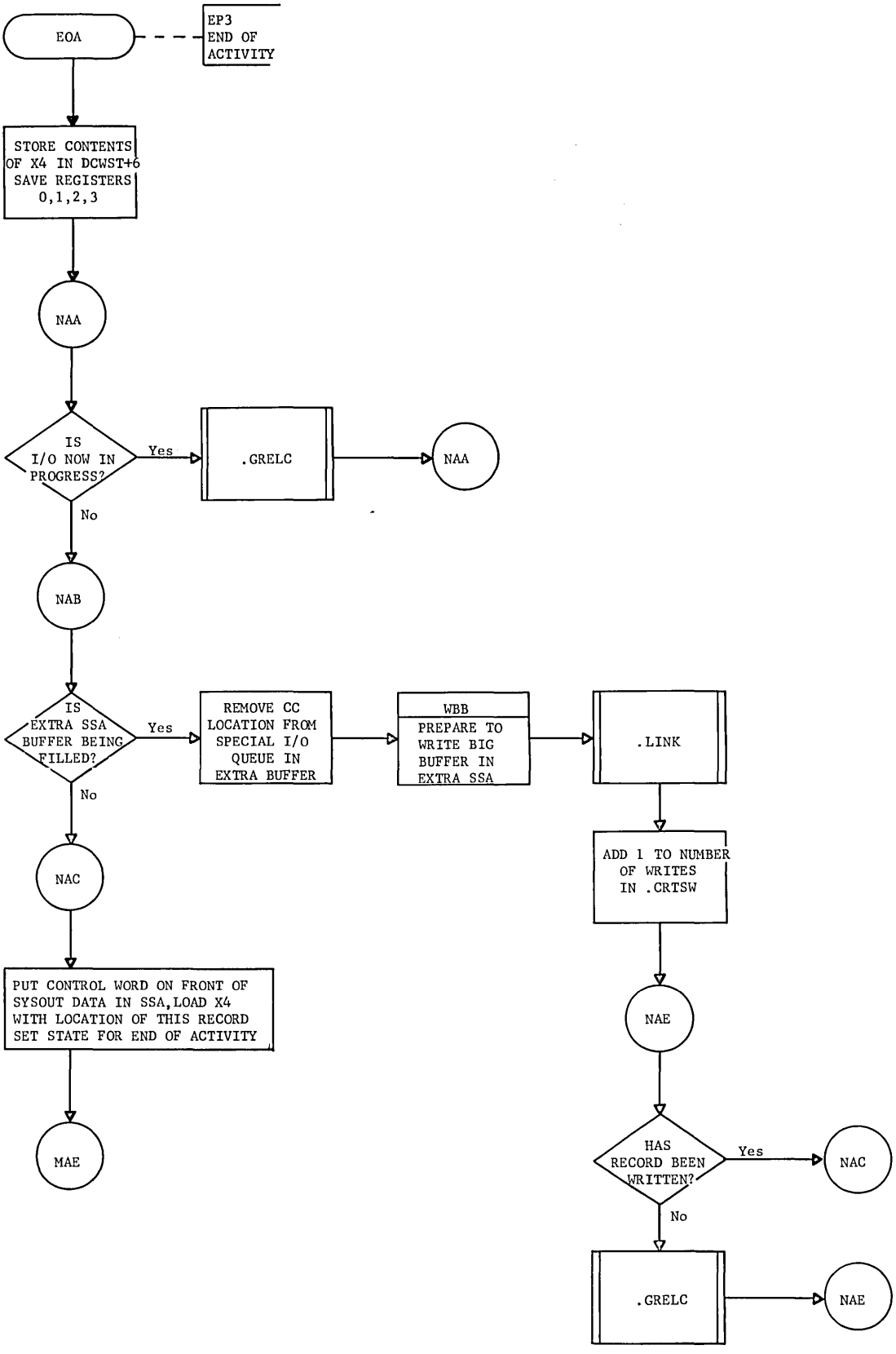
EXEC (EP2)
.MSYOT

WRITE EXECUTION REPORT MESSAGE (RELATIVE LOCATION)

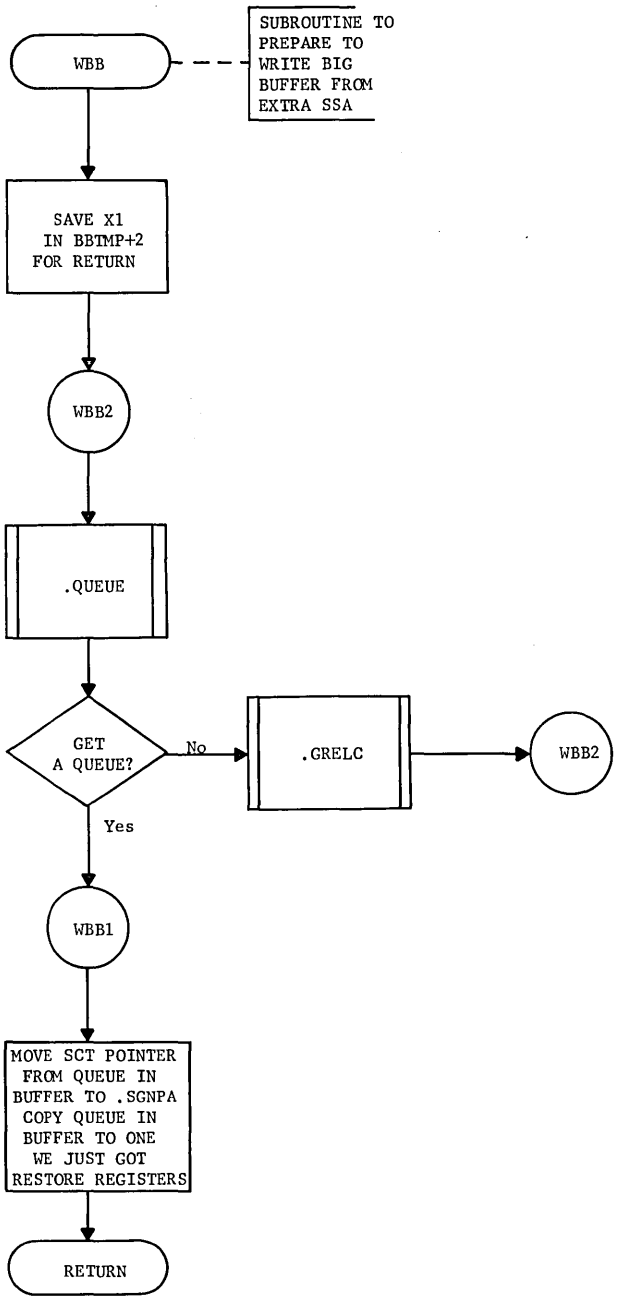


EOA (EP3)
.MSYOT

PROCESS END OF ACTIVITY

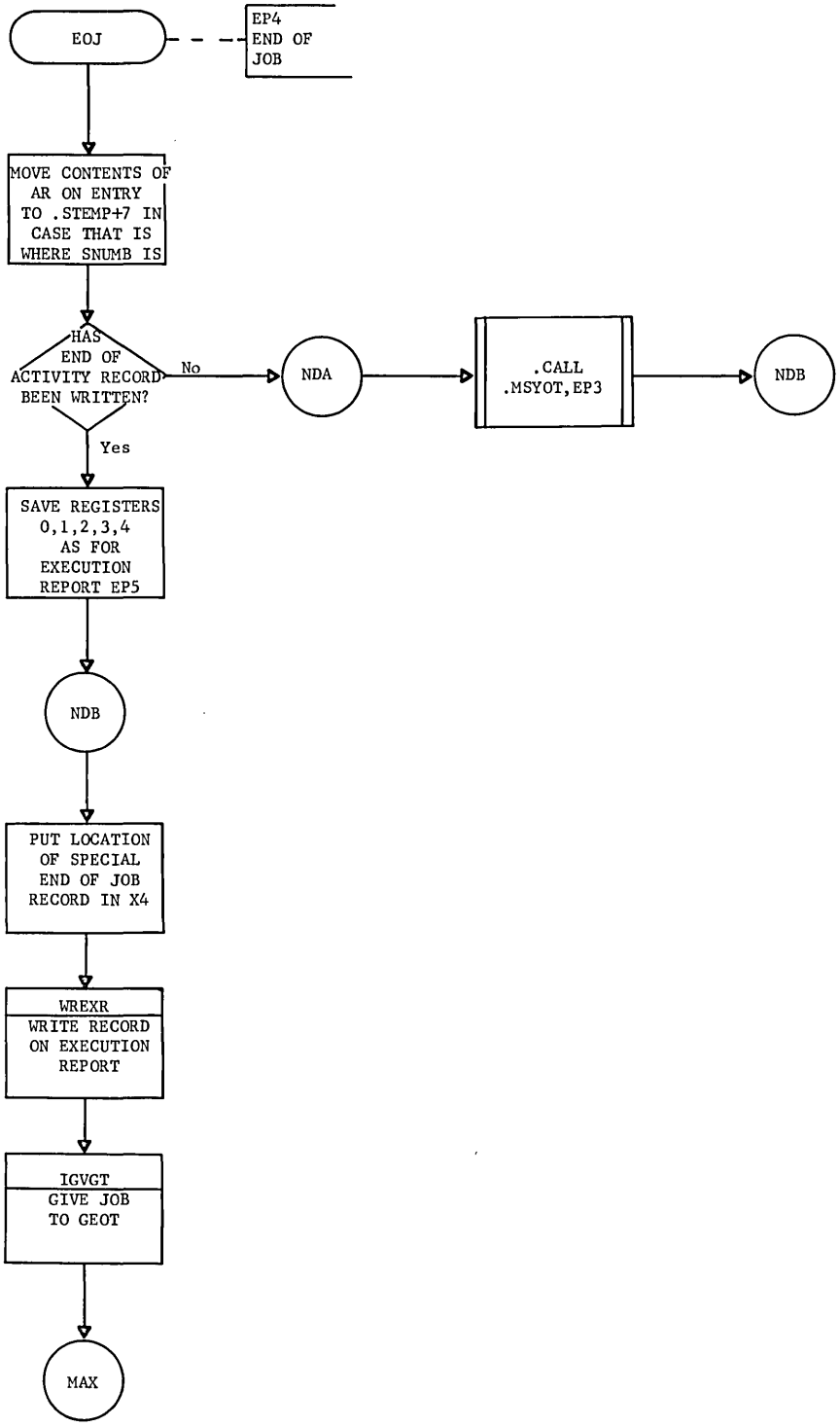


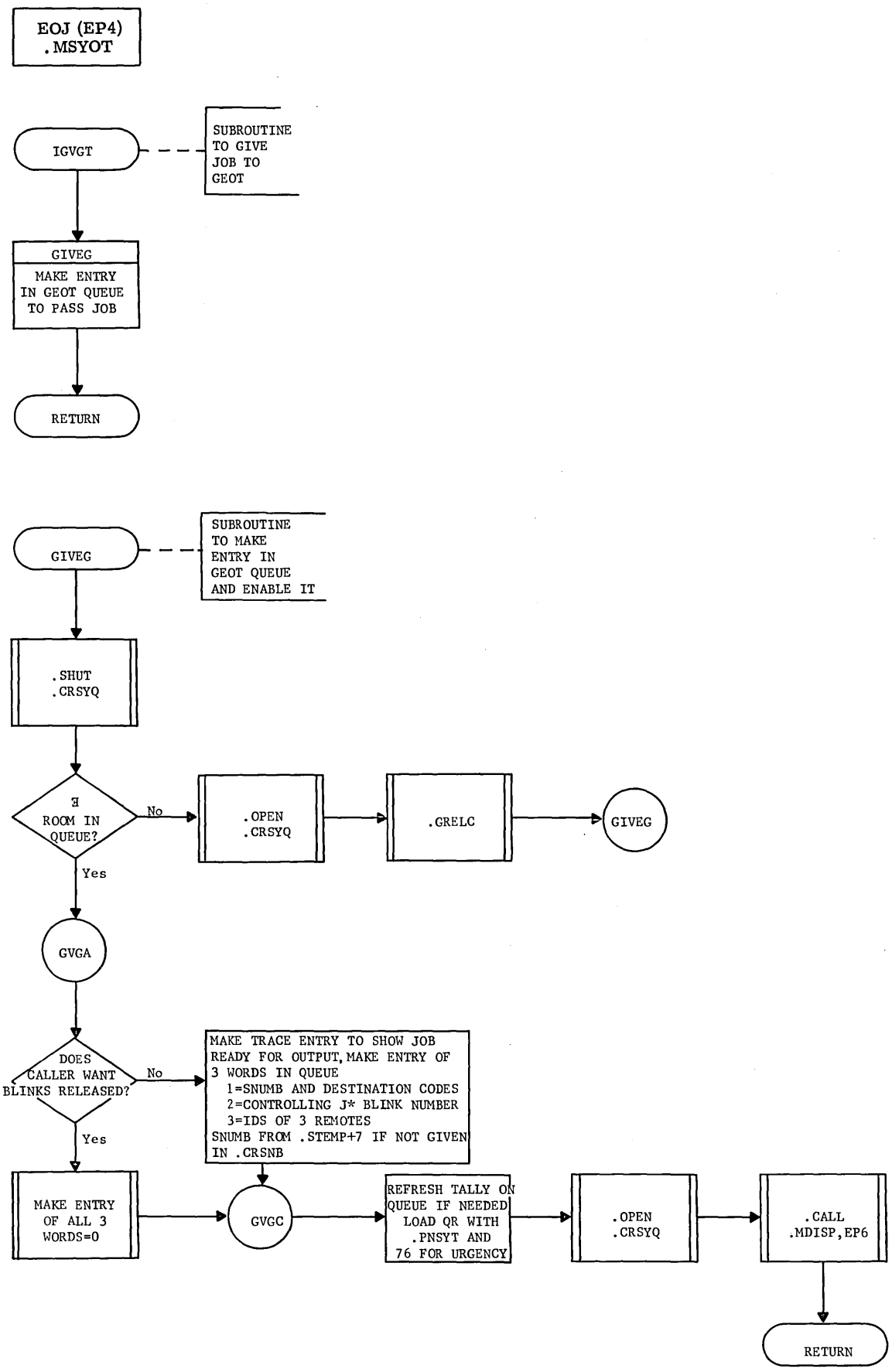
EOA (EP3)
.MSYOT

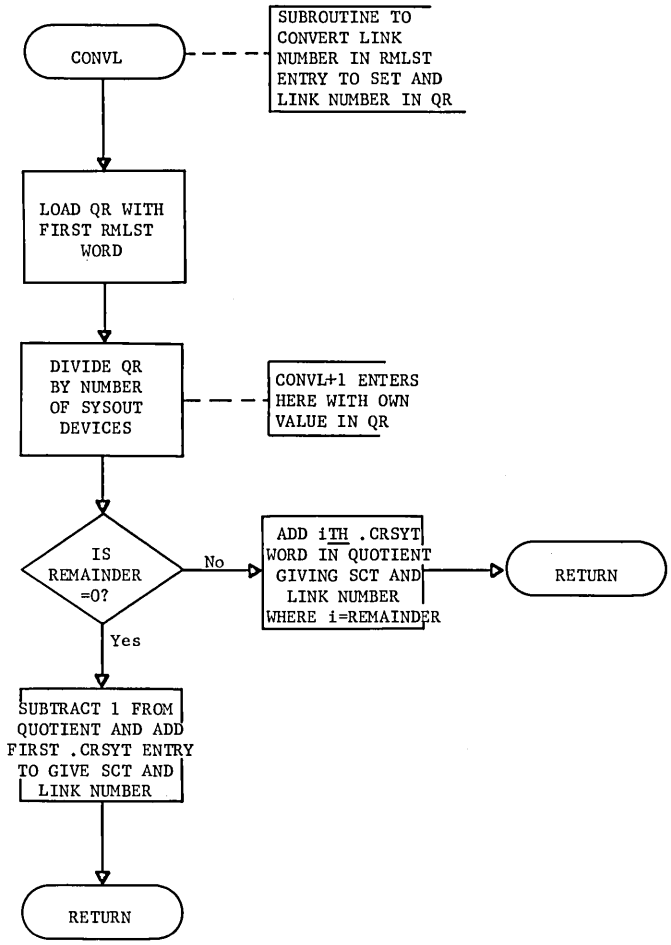


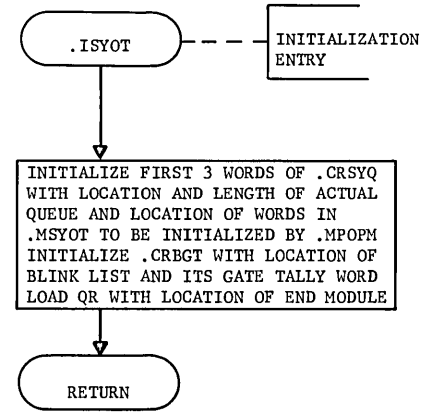
EOJ (EP4)
.MSYOT

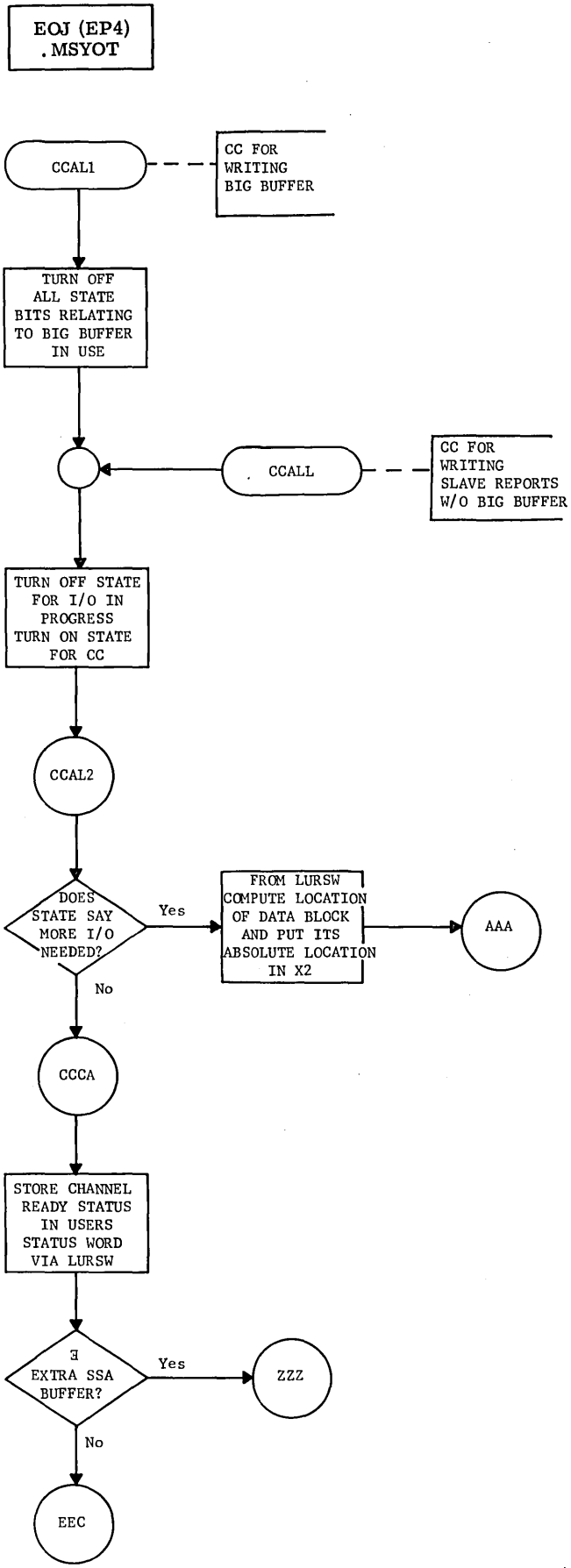
PROCESS END OF JOB

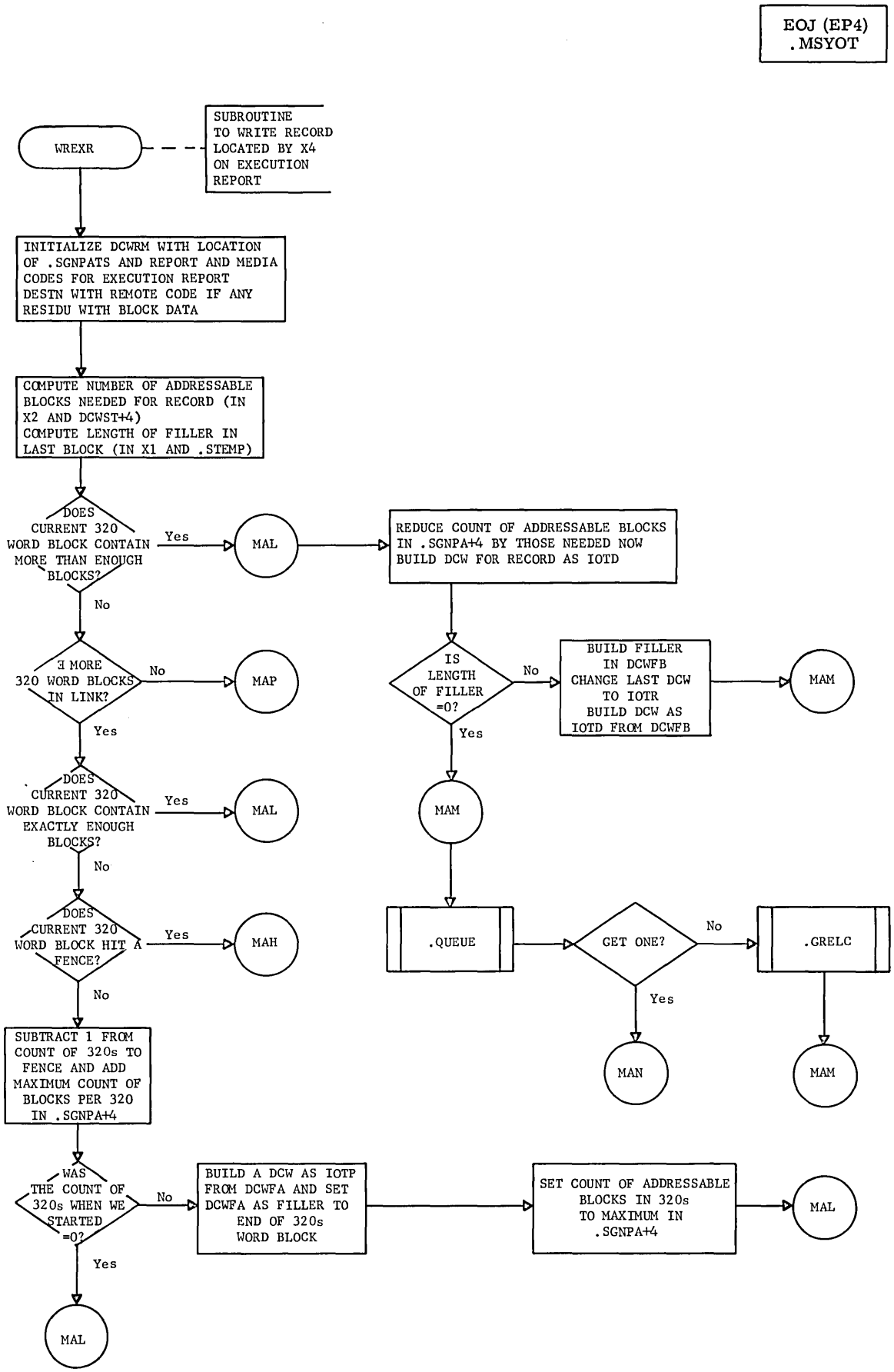


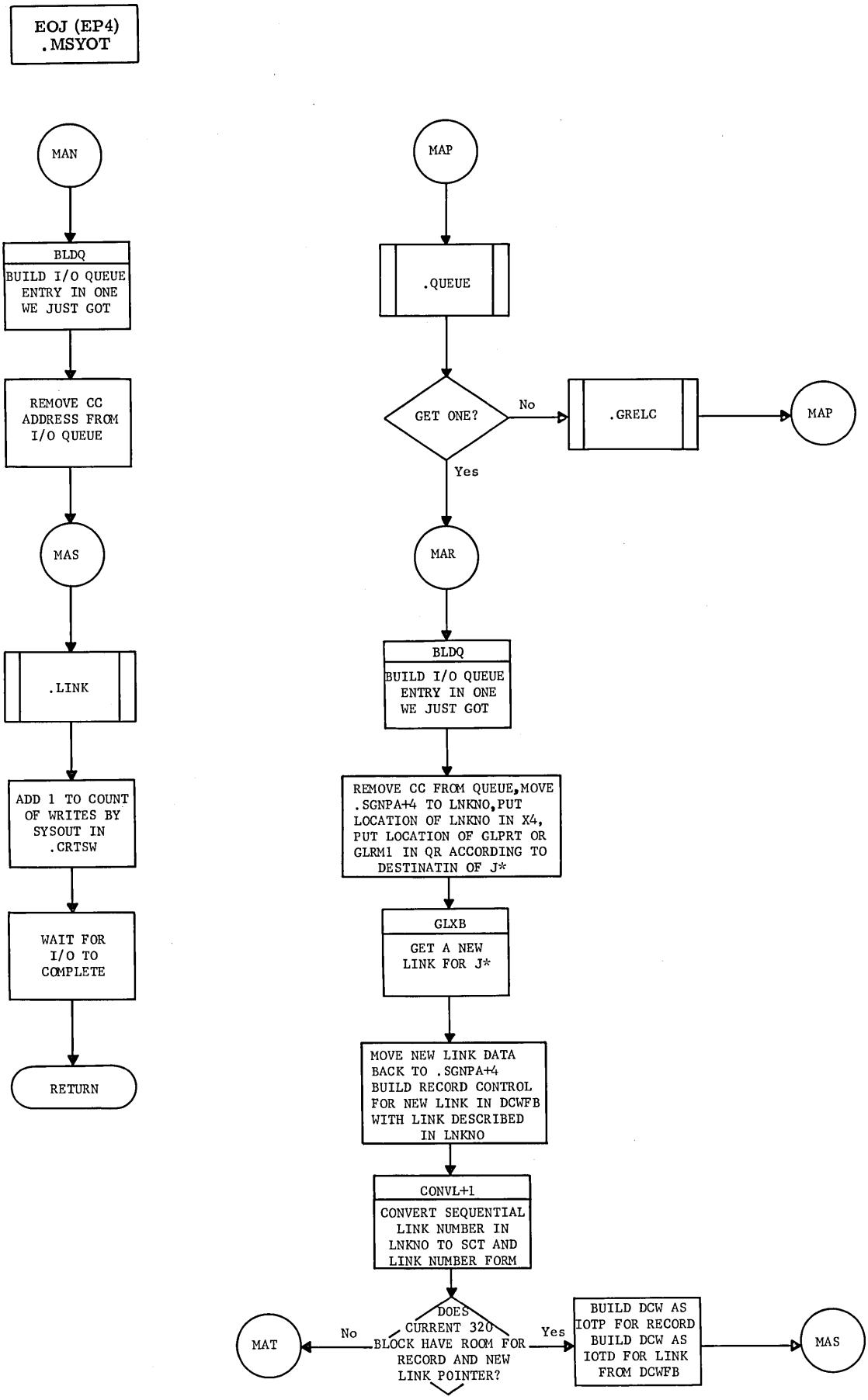

EOJ (EP4)
.MSYOT
IGVGT
SUBROUTINE TO GIVE JOB TO GEOT
GIVEG
MAKE ENTRY IN GEOT QUEUE TO PASS JOB
RETURN
GIVEG
SUBROUTINE TO MAKE ENTRY IN GEOT QUEUE AND ENABLE IT
.SHUT
.CRSYQ
∃ ROOM IN QUEUE?
No
.OPEN
.CRSYQ
.GRELC
GIVEG
Yes
GVGA
DOES CALLER WANT BLINKS RELEASED?
No
MAKE TRACE ENTRY TO SHOW JOB READY FOR OUTPUT, MAKE ENTRY OF 3 WORDS IN QUEUE
1=SNUMB AND DESTINATION CODES
2=CONTROLLING J* BLINK NUMBER
3=IDS OF 3 REMOTES
SNUMB FROM .STEMP+7 IF NOT GIVEN IN .CRSNB
Yes
MAKE ENTRY OF ALL 3 WORDS=0
GVGC
REFRESH TALLY ON QUEUE IF NEEDED LOAD QR WITH .PNSYT AND 76 FOR URGENCY
.OPEN
.CRSYQ
.CALL
.MDISP,EP6
RETURN

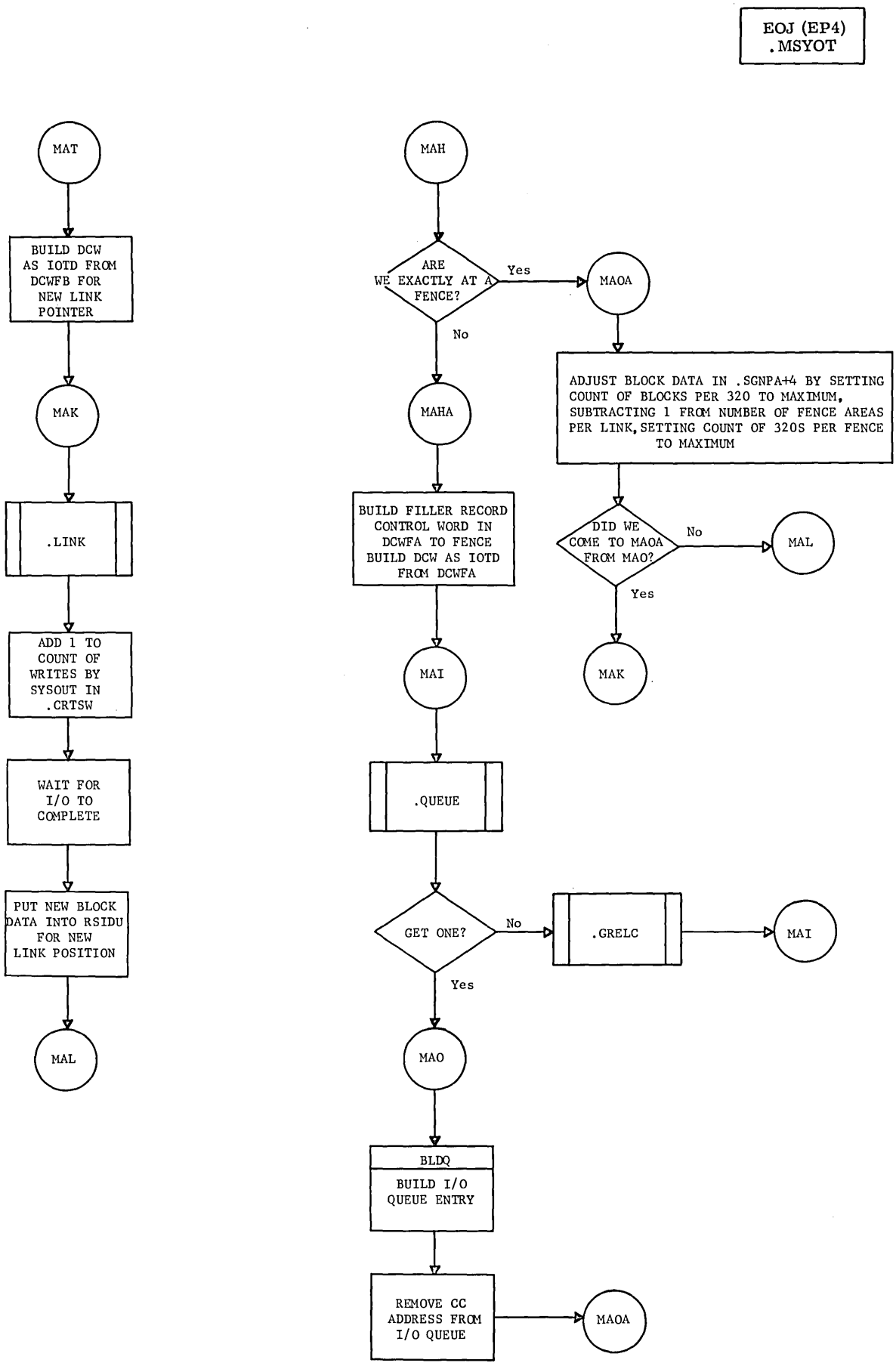

CONVL
SUBROUTINE TO CONVERT LINK NUMBER IN RMLST ENTRY TO SET AND LINK NUMBER IN QR
LOAD QR WITH FIRST RMLST WORD
DIVIDE QR BY NUMBER OF SYSOUT DEVICES
CONVL+1 ENTERS HERE WITH OWN VALUE IN QR
IS REMAINDER =0?
No
ADD iTH .CRSYT WORD IN QUOTIENT GIVING SCT AND LINK NUMBER WHERE i=REMAINDER
RETURN
Yes
SUBTRACT 1 FROM QUOTIENT AND ADD FIRST .CRSYT ENTRY TO GIVE SCT AND LINK NUMBER
RETURN


.ISYOT
INITIALIZATION ENTRY
INITIALIZE FIRST 3 WORDS OF .CRSYQ WITH LOCATION AND LENGTH OF ACTUAL QUEUE AND LOCATION OF WORDS IN .MSYOT TO BE INITIALIZED BY .MPOPM
INITIALIZE .CRBGT WITH LOCATION OF BLINK LIST AND ITS GATE TALLY WORD
LOAD QR WITH LOCATION OF END MODULE
RETURN

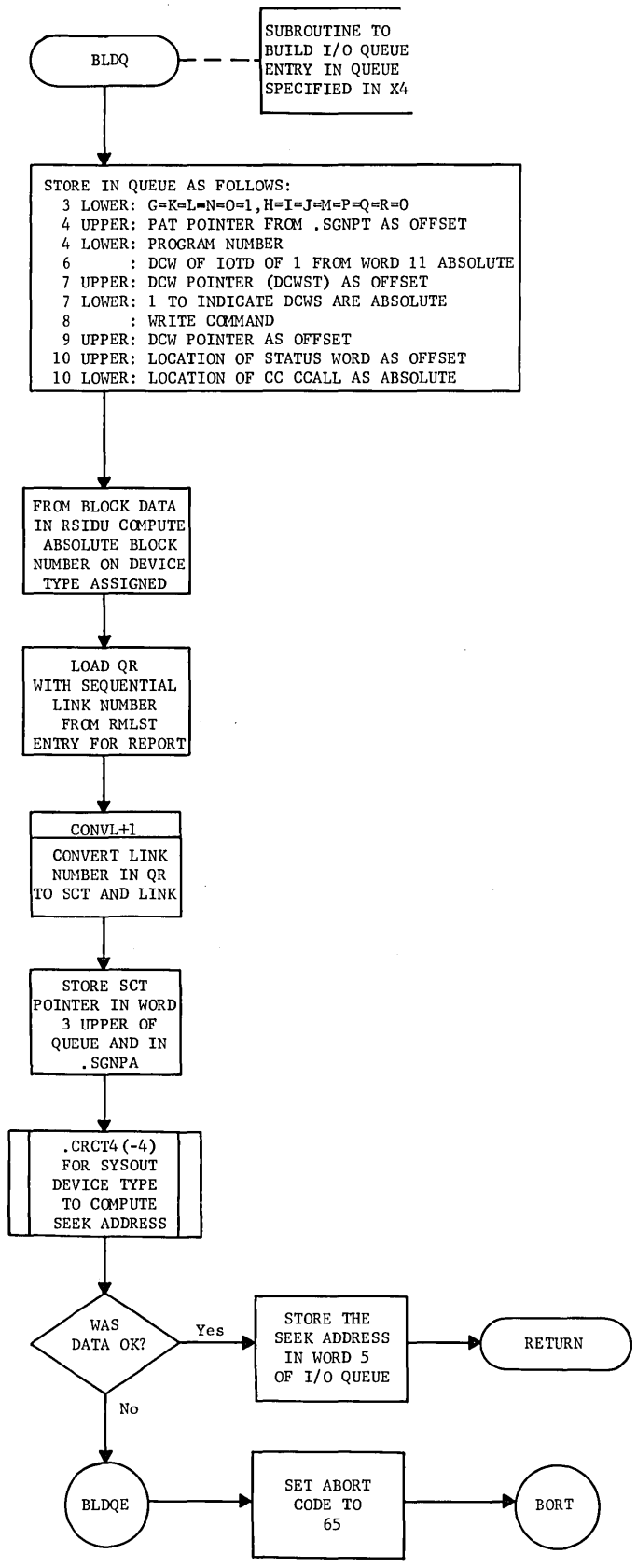

EOJ (EP4)
.MSYOT
CCAL1
CC FOR
WRITING
BIG BUFFER
TURN OFF
ALL STATE
BITS RELATING
TO BIG BUFFER
IN USE
CCALL
CC FOR
WRITING
SLAVE REPORTS
W/O BIG BUFFER
TURN OFF STATE
FOR I/O IN
PROGRESS
TURN ON STATE
FOR CC
CCAL2
DOES
STATE SAY
MORE I/O
NEEDED?
Yes
FROM LURSW
COMPUTE LOCATION
OF DATA BLOCK
AND PUT ITS
ABSOLUTE LOCATION
IN X2
AAA
No
CCCA
STORE CHANNEL
READY STATUS
IN USERS
STATUS WORD
VIA LURSW
∃
EXTRA SSA
BUFFER?
Yes
ZZZ
No
EEC

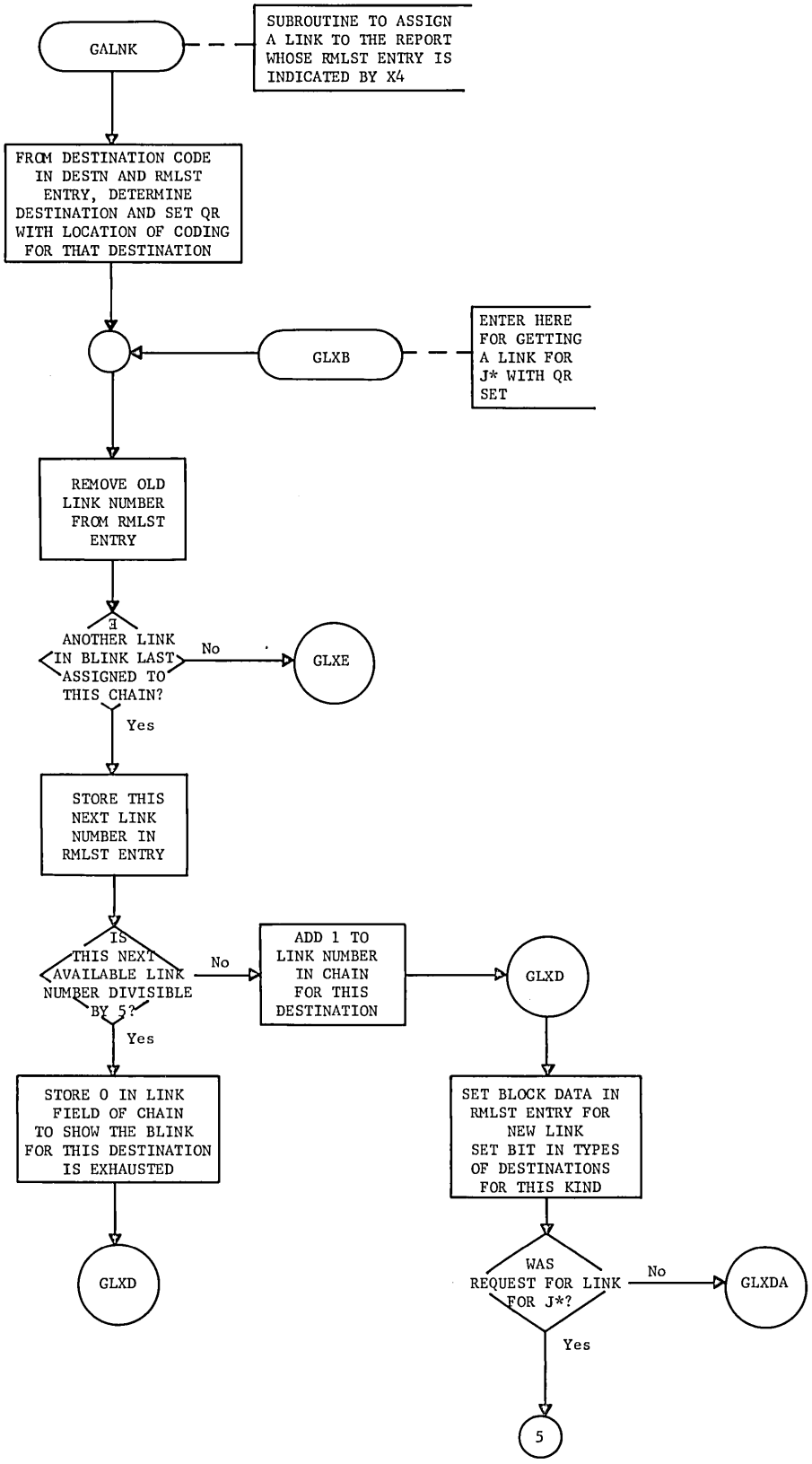

EOJ (EP4)
.MSYOT
WREXR
SUBROUTINE TO WRITE RECORD LOCATED BY X4 ON EXECUTION REPORT
INITIALIZE DCWRM WITH LOCATION OF .SGNPATS AND REPORT AND MEDIA CODES FOR EXECUTION REPORT DESTN WITH REMOTE CODE IF ANY RESIDU WITH BLOCK DATA
COMPUTE NUMBER OF ADDRESSABLE BLOCKS NEEDED FOR RECORD (IN X2 AND DCWST+4)
COMPUTE LENGTH OF FILLER IN LAST BLOCK (IN X1 AND .STEMP)
DOES CURRENT 320 WORD BLOCK CONTAIN MORE THAN ENOUGH BLOCKS?
Yes
MAL
REDUCE COUNT OF ADDRESSABLE BLOCKS IN .SGNPA+4 BY THOSE NEEDED NOW
BUILD DCW FOR RECORD AS IOTD
No
∃ MORE 320 WORD BLOCKS IN LINK?
No
MAP
Yes
IS LENGTH OF FILLER =0?
No
BUILD FILLER IN DCWFB
CHANGE LAST DCW TO IOTR
BUILD DCW AS IOTD FROM DCWFB
MAM
Yes
MAM
DOES CURRENT 320 WORD BLOCK CONTAIN EXACTLY ENOUGH BLOCKS?
Yes
MAL
No
DOES CURRENT 320 WORD BLOCK HIT A FENCE?
Yes
MAH
.QUEUE
GET ONE?
No
.GRELC
Yes
MAN
MAM
No
SUBTRACT 1 FROM COUNT OF 320s TO FENCE AND ADD MAXIMUM COUNT OF BLOCKS PER 320 IN .SGNPA+4
WAS THE COUNT OF 320s WHEN WE STARTED =0?
No
BUILD A DCW AS IOTP FROM DCWFA AND SET DCWFA AS FILLER TO END OF 320s WORD BLOCK
SET COUNT OF ADDRESSABLE BLOCKS IN 320s TO MAXIMUM IN .SGNPA+4
MAL
Yes
MAL


EOJ (EP4)
.MSYOT
MAN
BLDQ
BUILD I/O QUEUE ENTRY IN ONE WE JUST GOT
REMOVE CC ADDRESS FROM I/O QUEUE
MAS
.LINK
ADD 1 TO COUNT OF WRITES BY SYSOUT IN .CRTSW
WAIT FOR I/O TO COMPLETE
RETURN
MAP
.QUEUE
GET ONE?
No
.GRELC
MAP
Yes
MAR
BLDQ
BUILD I/O QUEUE ENTRY IN ONE WE JUST GOT
REMOVE CC FROM QUEUE,MOVE .SGNPA+4 TO LNKNO,PUT LOCATION OF LNKNO IN X4, PUT LOCATION OF GLPRT OR GLRM1 IN QR ACCORDING TO DESTINATIN OF J*
GLXB
GET A NEW LINK FOR J*
MOVE NEW LINK DATA BACK TO .SGNPA+4 BUILD RECORD CONTROL FOR NEW LINK IN DCWFB WITH LINK DESCRIBED IN LNKNO
CONVL+1
CONVERT SEQUENTIAL LINK NUMBER IN LNKNO TO SCT AND LINK NUMBER FORM
DOES CURRENT 320 BLOCK HAVE ROOM FOR RECORD AND NEW LINK POINTER?
No
MAT
Yes
BUILD DCW AS IOTP FOR RECORD BUILD DCW AS IOTD FOR LINK FROM DCWFB
MAS

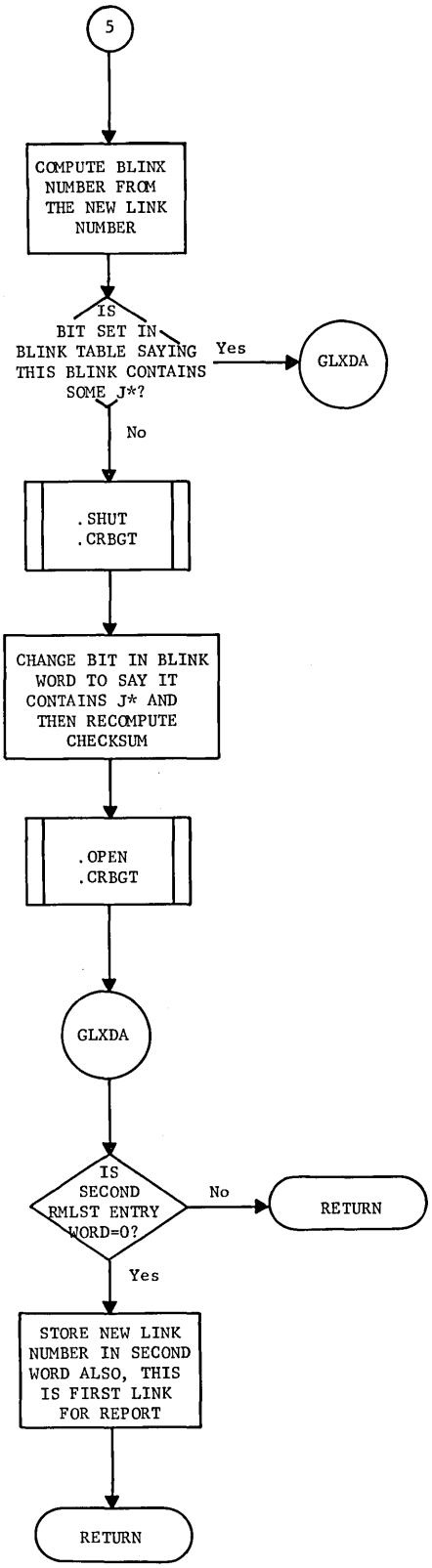

EOJ (EP4)
.MSYOT
MAT
BUILD DCW AS IOTD FROM DCWFB FOR NEW LINK POINTER
MAK
.LINK
ADD 1 TO COUNT OF WRITES BY SYSOUT IN .CRTSW
WAIT FOR I/O TO COMPLETE
PUT NEW BLOCK DATA INTO RSIDU FOR NEW LINK POSITION
MAL
MAH
ARE WE EXACTLY AT A FENCE?
Yes
No
MAOA
ADJUST BLOCK DATA IN .SGNPA+4 BY SETTING COUNT OF BLOCKS PER 320 TO MAXIMUM, SUBTRACTING 1 FROM NUMBER OF FENCE AREAS PER LINK, SETTING COUNT OF 320S PER FENCE TO MAXIMUM
MAHA
BUILD FILLER RECORD CONTROL WORD IN DCWFA TO FENCE BUILD DCW AS IOTD FROM DCWFA
DID WE COME TO MAOA FROM MAO?
No
MAL
Yes
MAK
MAI
.QUEUE
GET ONE?
No
.GRELC
MAI
Yes
MAO
BLDQ
BUILD I/O QUEUE ENTRY
REMOVE CC ADDRESS FROM I/O QUEUE
MAOA

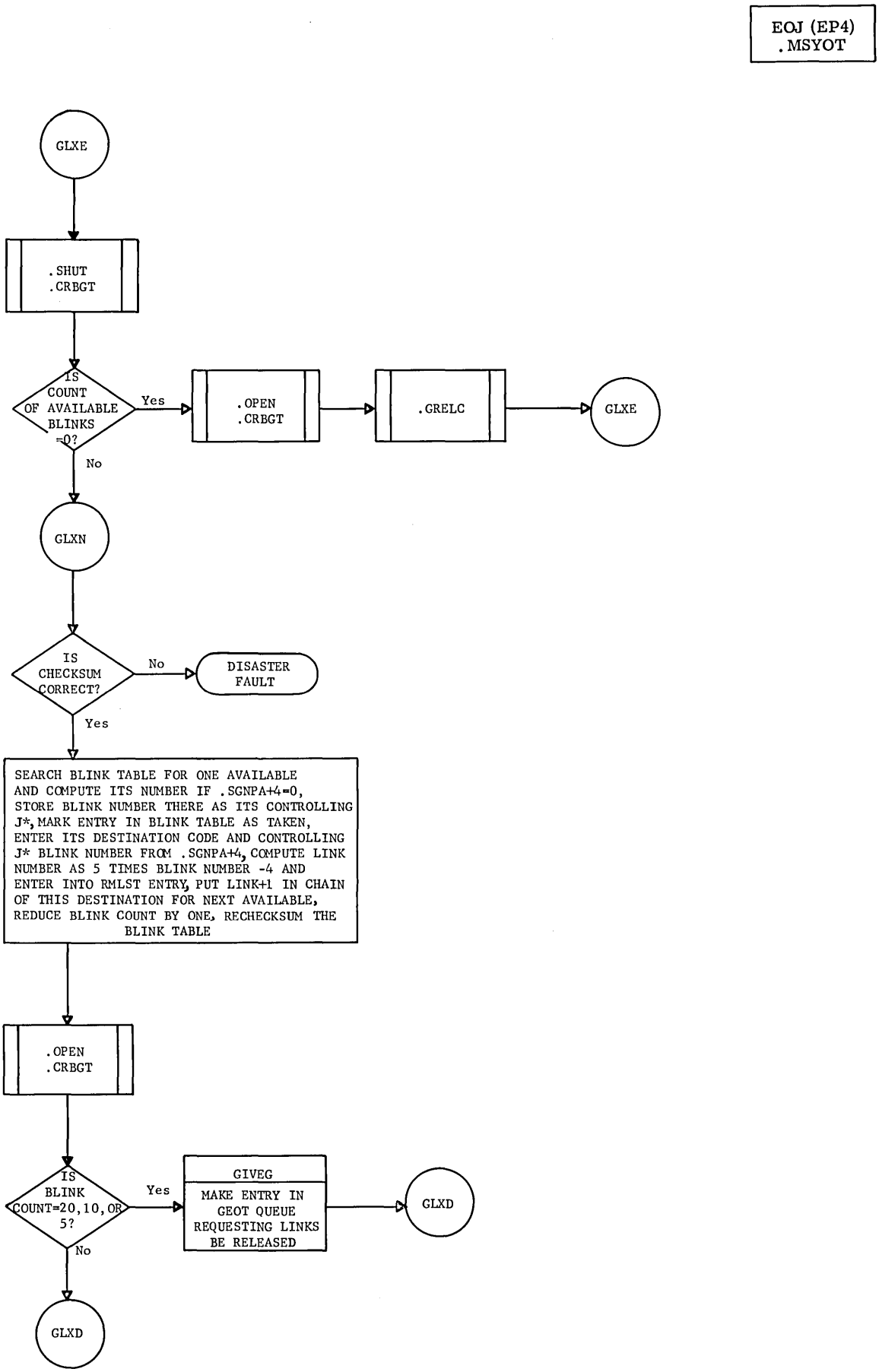
EOJ (EP4)
.MSYOT




GALNK
SUBROUTINE TO ASSIGN A LINK TO THE REPORT WHOSE RMLST ENTRY IS INDICATED BY X4
FROM DESTINATION CODE IN DESTN AND RMLST ENTRY, DETERMINE DESTINATION AND SET QR WITH LOCATION OF CODING FOR THAT DESTINATION
GLXB
ENTER HERE FOR GETTING A LINK FOR J* WITH QR SET
REMOVE OLD LINK NUMBER FROM RMLST ENTRY
IS ANOTHER LINK IN BLINK LAST ASSIGNED TO THIS CHAIN?
No
GLXE
Yes
STORE THIS NEXT LINK NUMBER IN RMLST ENTRY
IS THIS NEXT AVAILABLE LINK NUMBER DIVISIBLE BY 5?
No
ADD 1 TO LINK NUMBER IN CHAIN FOR THIS DESTINATION
GLXD
Yes
STORE 0 IN LINK FIELD OF CHAIN TO SHOW THE BLINK FOR THIS DESTINATION IS EXHAUSTED
GLXD
SET BLOCK DATA IN RMLST ENTRY FOR NEW LINK
SET BIT IN TYPES OF DESTINATIONS FOR THIS KIND
WAS REQUEST FOR LINK FOR J*?
No
GLXDA
Yes
5

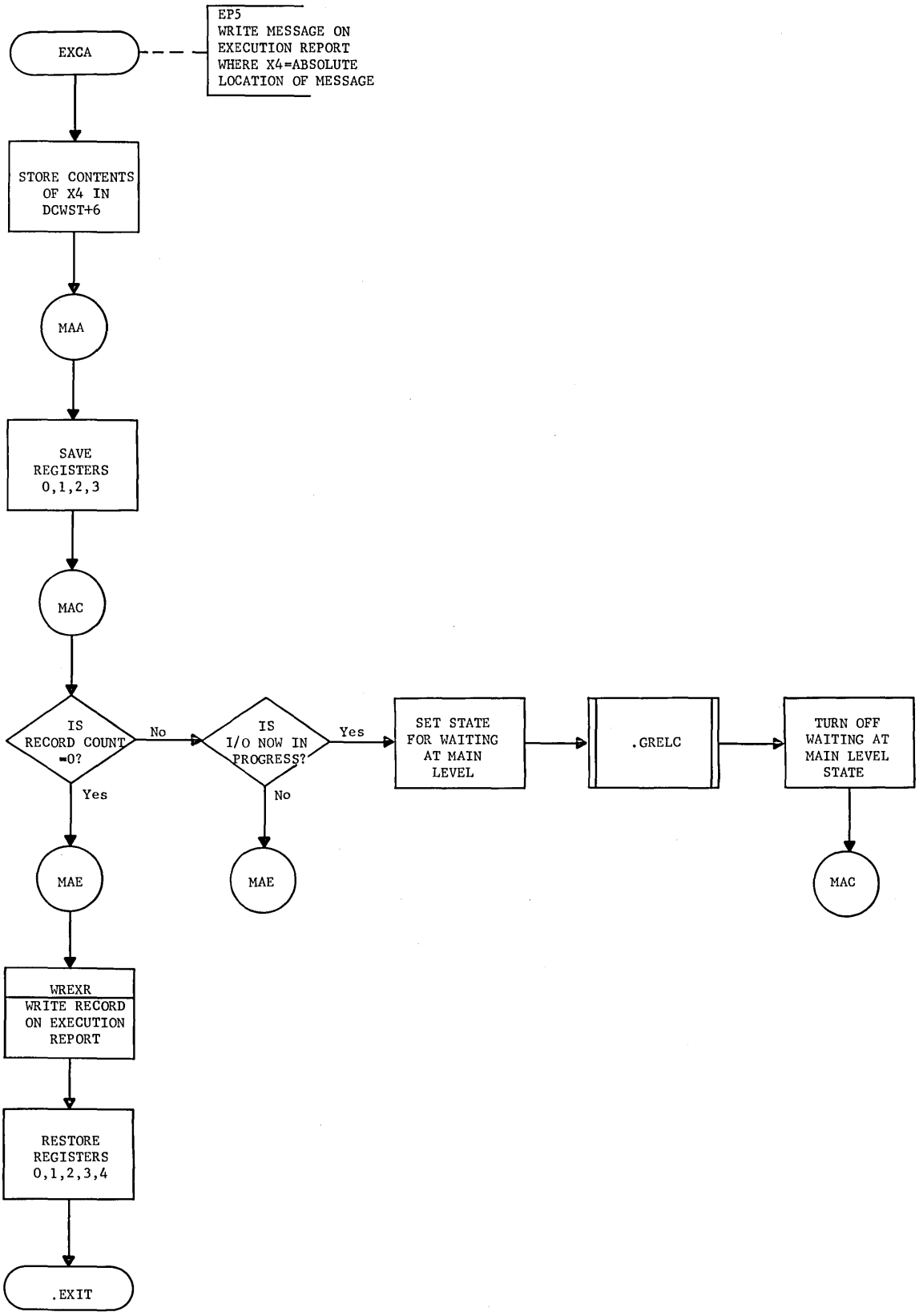
EOJ (EP4)
.MSYOT




EOJ (EP4)
.MSYOT
GLXE
.SHUT
.CRBGT
IS COUNT OF AVAILABLE BLINKS =0?
Yes
.OPEN
.CRBGT
.GRELC
GLXE
No
GLXN
IS CHECKSUM CORRECT?
No
DISASTER FAULT
Yes
SEARCH BLINK TABLE FOR ONE AVAILABLE AND COMPUTE ITS NUMBER IF .SGNPA+4=0, STORE BLINK NUMBER THERE AS ITS CONTROLLING J*, MARK ENTRY IN BLINK TABLE AS TAKEN, ENTER ITS DESTINATION CODE AND CONTROLLING J* BLINK NUMBER FROM .SGNPA+4, COMPUTE LINK NUMBER AS 5 TIMES BLINK NUMBER -4 AND ENTER INTO RMLST ENTRY, PUT LINK+1 IN CHAIN OF THIS DESTINATION FOR NEXT AVAILABLE, REDUCE BLINK COUNT BY ONE, RECHECKSUM THE BLINK TABLE
.OPEN
.CRBGT
IS BLINK COUNT=20,10,OR 5?
Yes
GIVEG
MAKE ENTRY IN GEOT QUEUE REQUESTING LINKS BE RELEASED
GLXD
No
GLXD

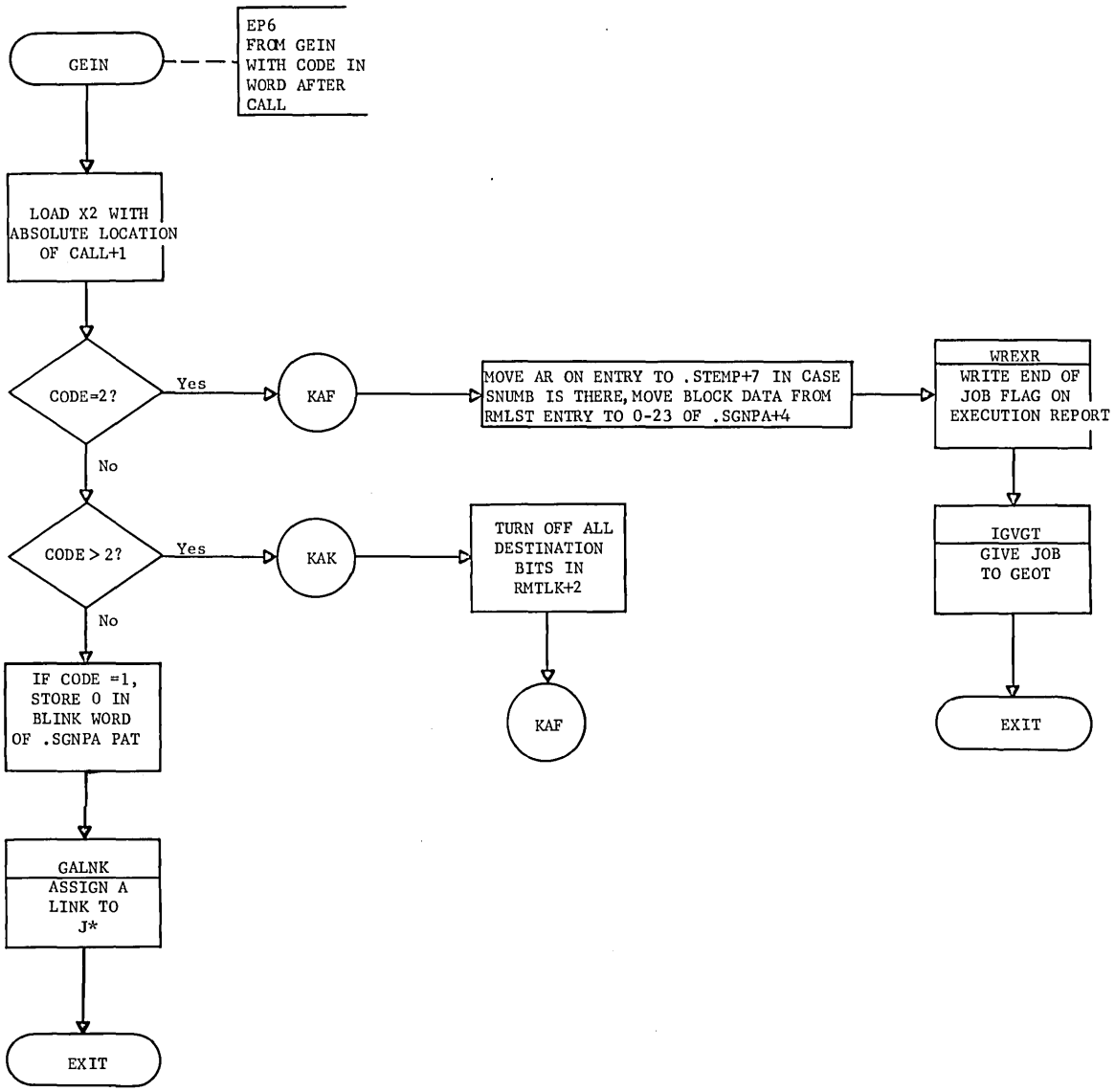
EXCA (EP5)
.MSYOT

WRITE EXECUTION REPORT MESSAGE (ABSOLUTE LOCATION)



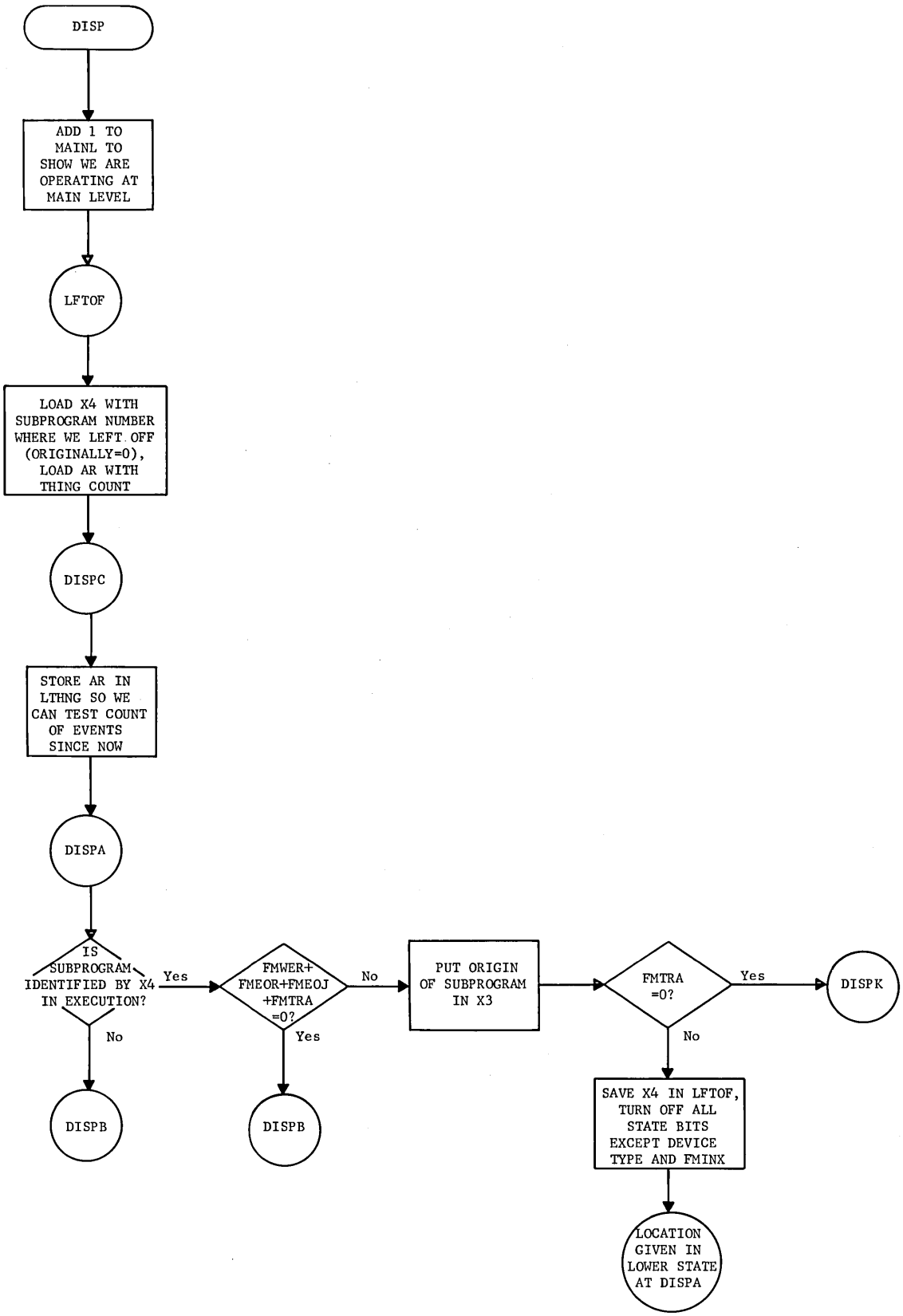
GEIN (EP6)
.MSYOT

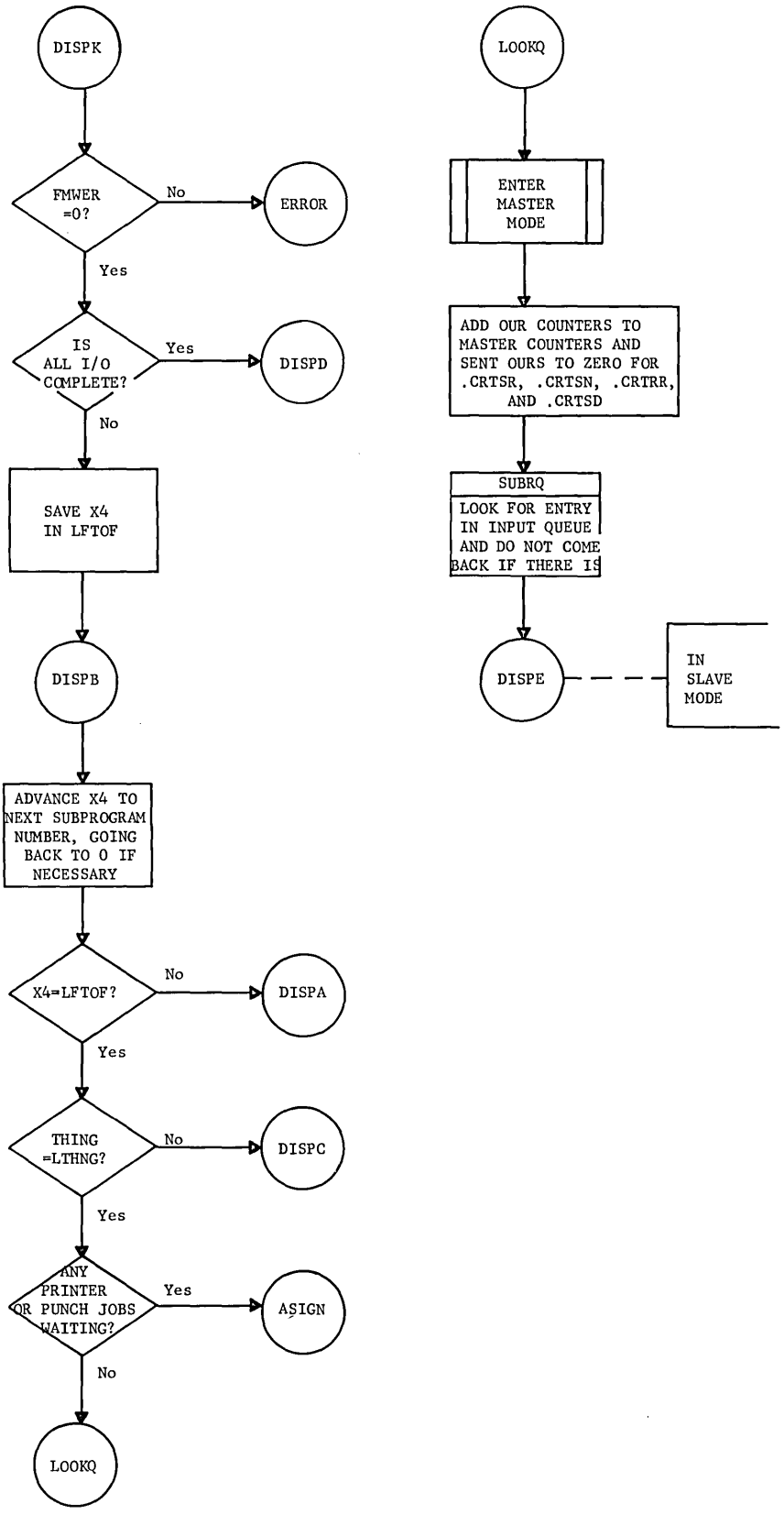
PROCESS CALL FROM GEIN

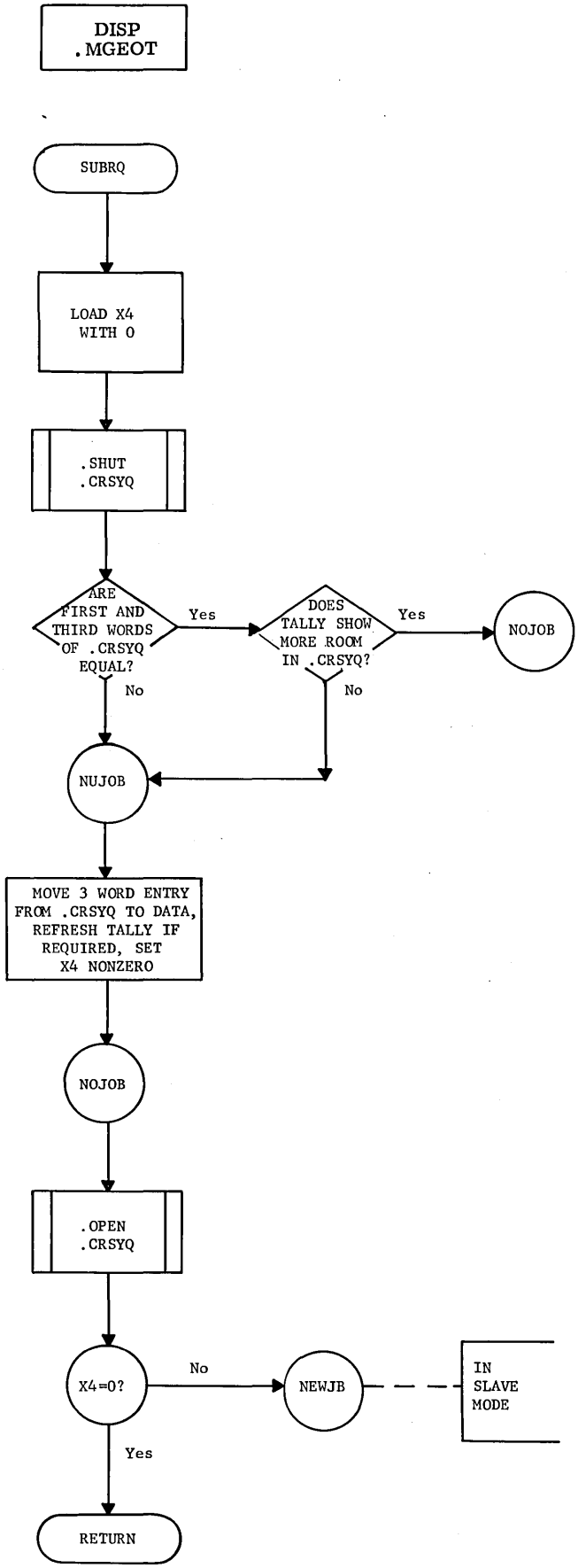


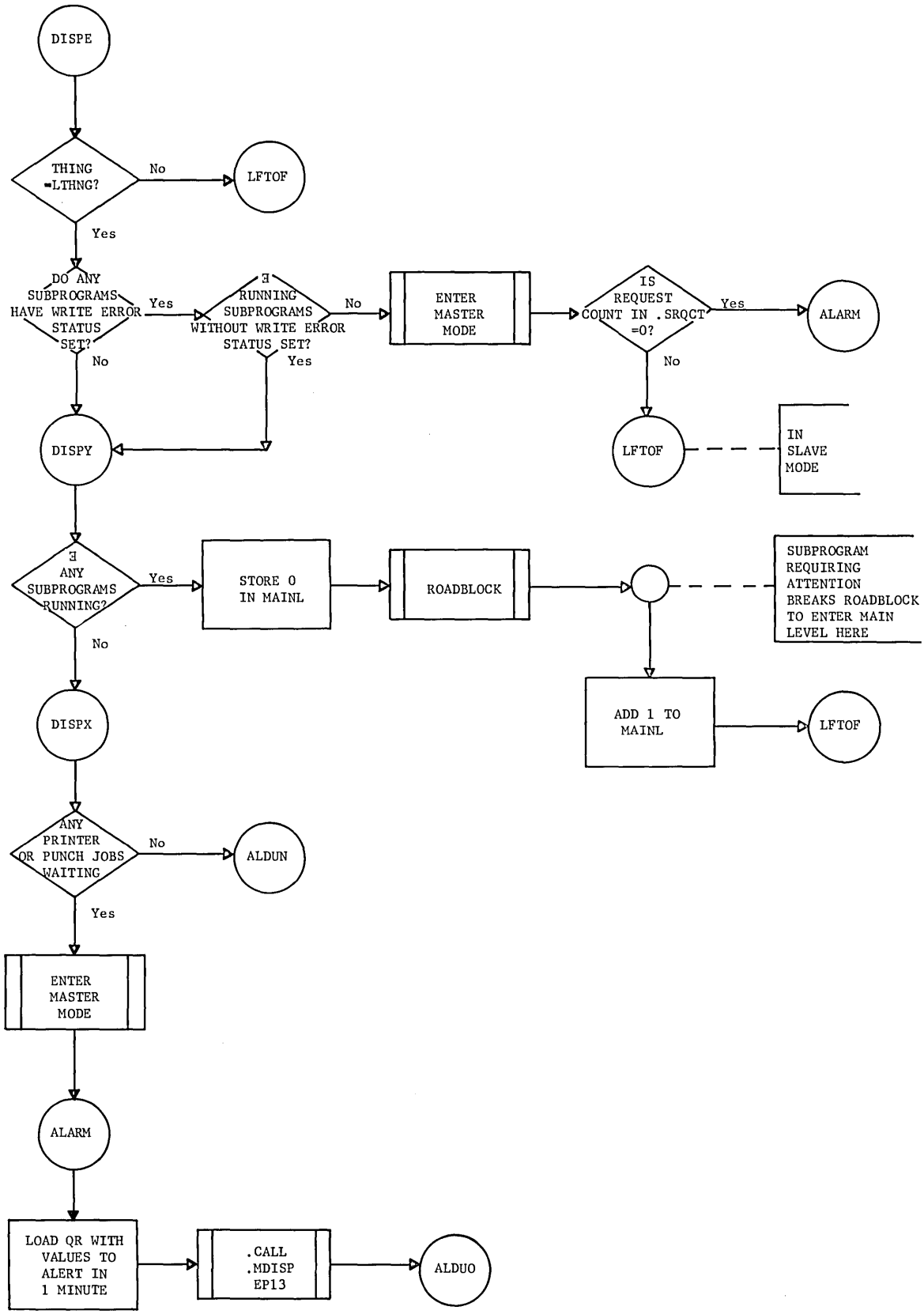
DISP
.MGEOT

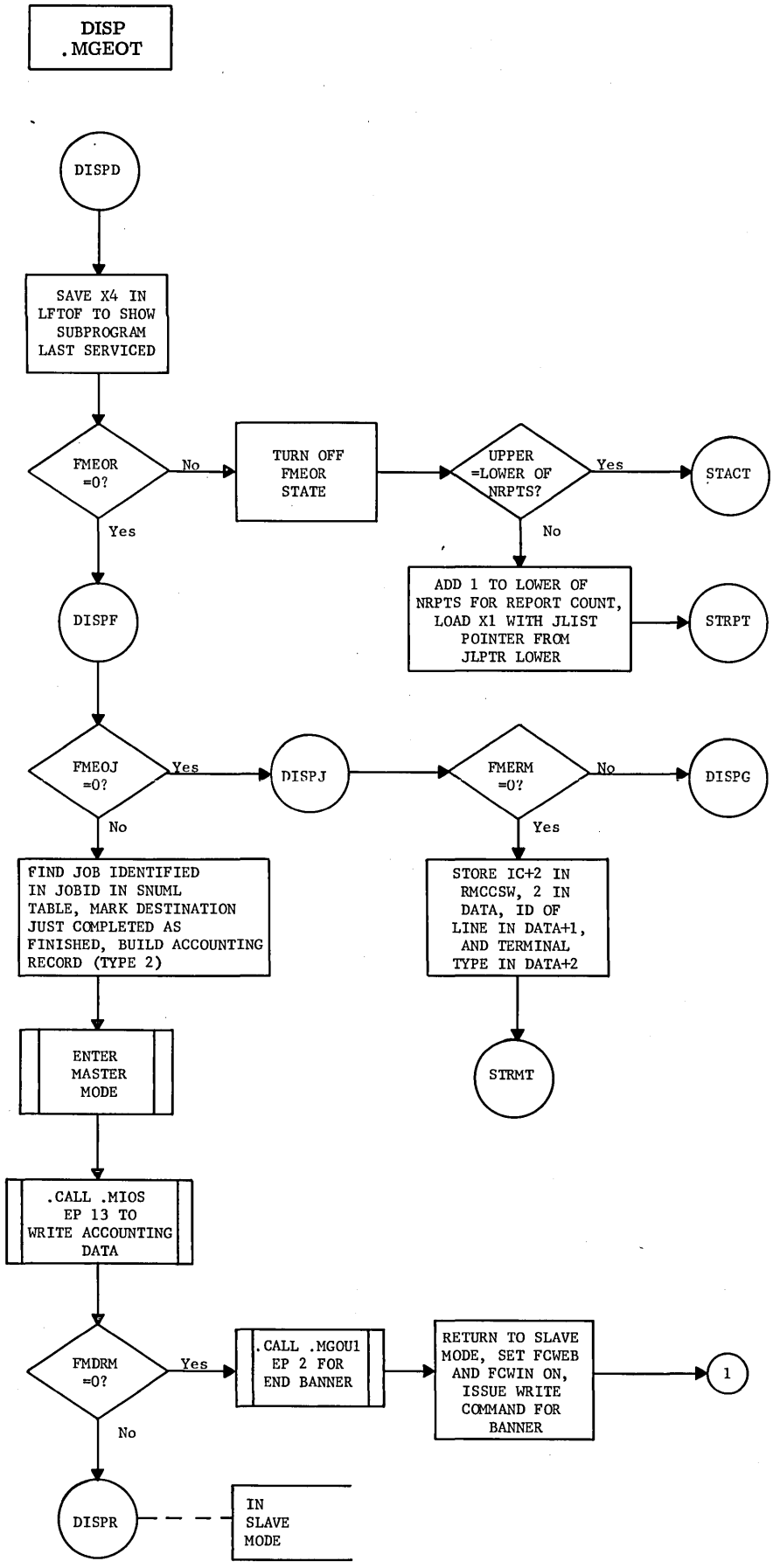
DISPATCHER

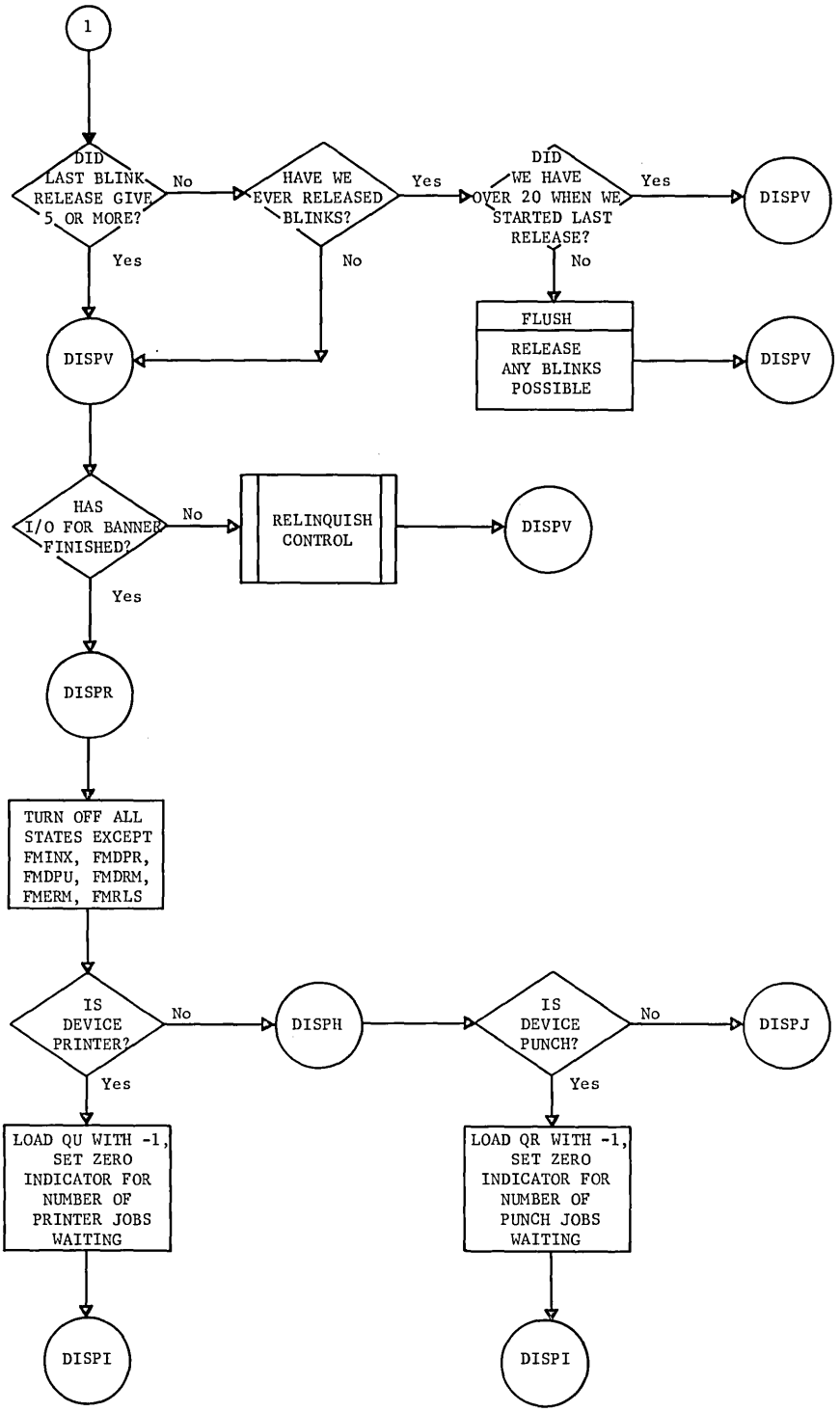


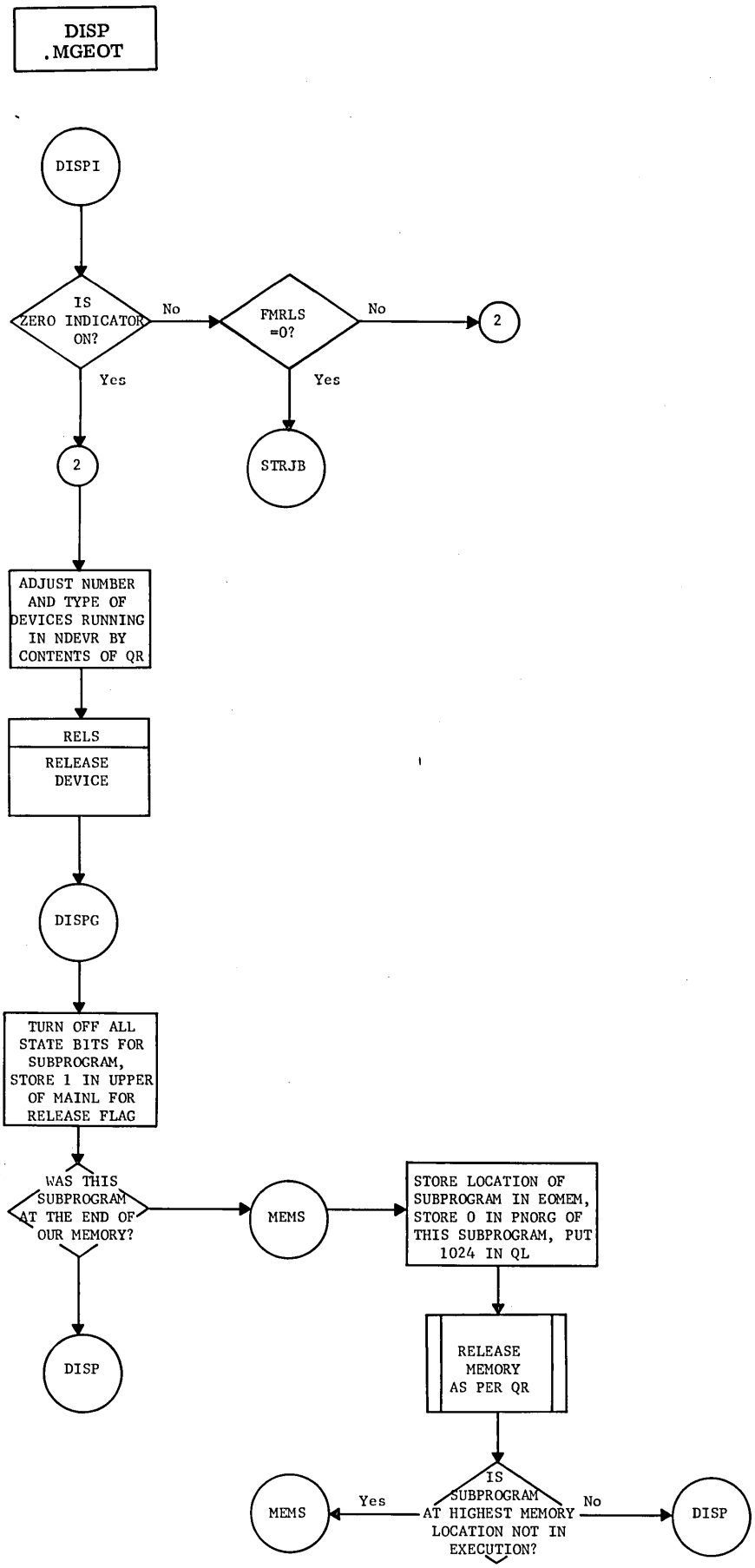

DISPK
FMWER =0?
No
ERROR
Yes
IS ALL I/O COMPLETE?
Yes
DISPD
No
SAVE X4 IN LFTOF
DISPB
ADVANCE X4 TO NEXT SUBPROGRAM NUMBER, GOING BACK TO 0 IF NECESSARY
X4=LFTOF?
No
DISPA
Yes
THING =LTHNG?
No
DISPC
Yes
ANY PRINTER OR PUNCH JOBS WAITING?
Yes
ASIGN
No
LOOKQ
LOOKQ
ENTER MASTER MODE
ADD OUR COUNTERS TO MASTER COUNTERS AND SENT OURS TO ZERO FOR .CRTSR, .CRTSN, .CRTRR, AND .CRTSD
SUBRQ
LOOK FOR ENTRY IN INPUT QUEUE AND DO NOT COME BACK IF THERE IS
DISPE
IN SLAVE MODE

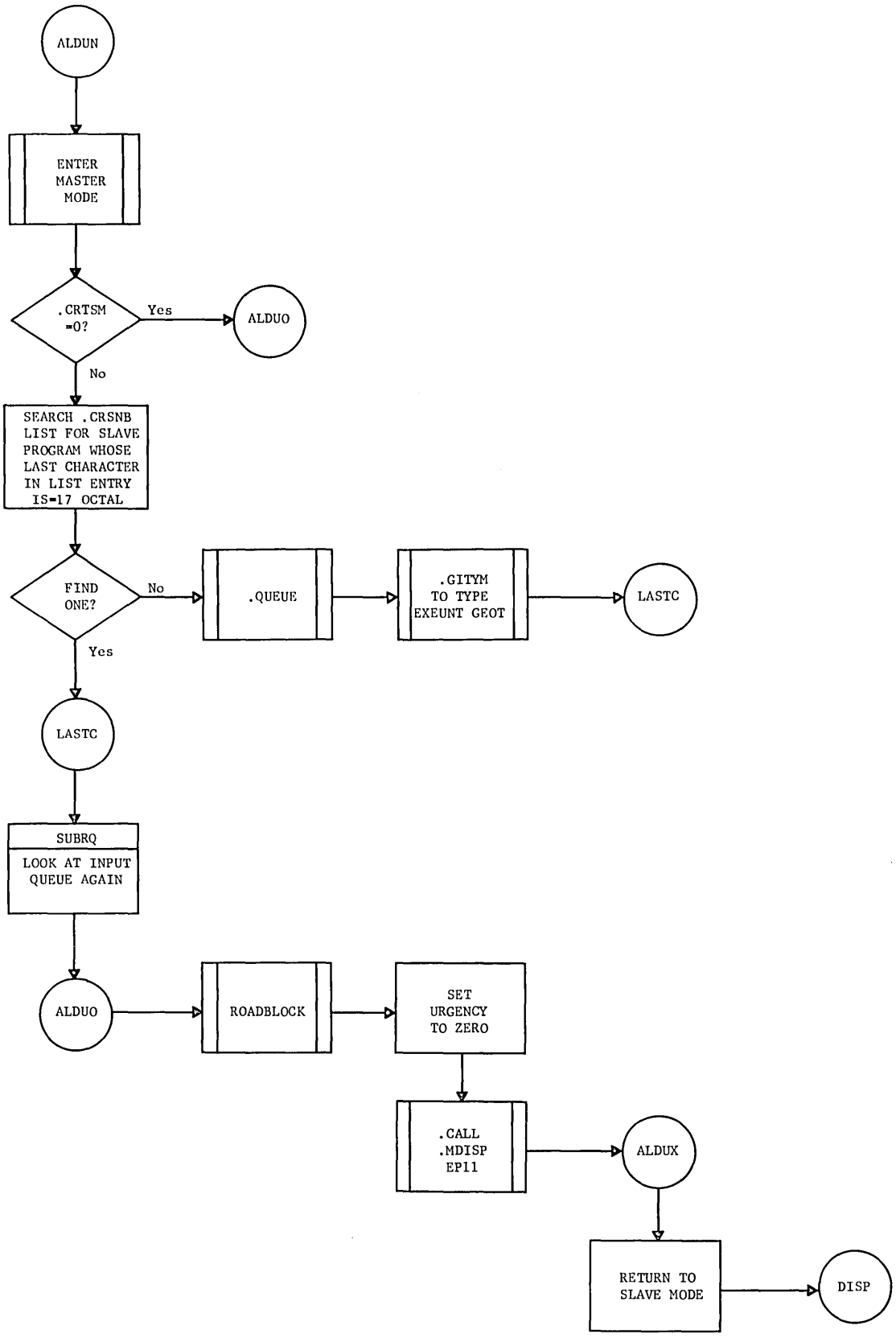

DISP
.MGEOT
SUBRQ
LOAD X4
WITH 0
.SHUT
.CRSYQ
ARE
FIRST AND
THIRD WORDS
OF .CRSYQ
EQUAL?
Yes
DOES
TALLY SHOW
MORE ROOM
IN .CRSYQ?
Yes
NOJOB
No
No
NUJOB
MOVE 3 WORD ENTRY
FROM .CRSYQ TO DATA,
REFRESH TALLY IF
REQUIRED, SET
X4 NONZERO
NOJOB
.OPEN
.CRSYQ
X4=0?
No
NEWJB
IN
SLAVE
MODE
Yes
RETURN

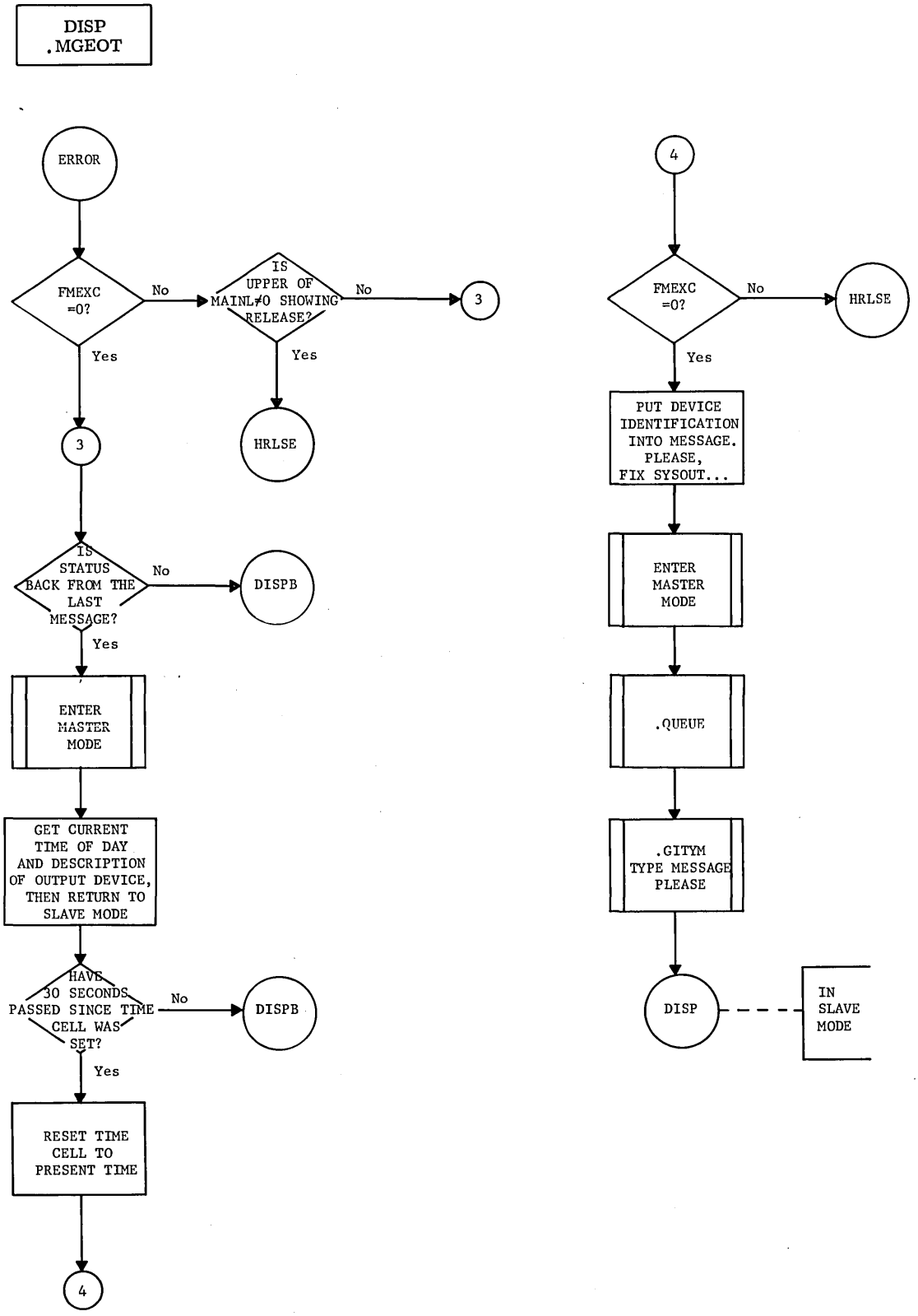

DISPE
THING =LTHNG?
No
LFTOF
Yes
DO ANY SUBPROGRAMS HAVE WRITE ERROR STATUS SET?
Yes
E RUNNING SUBPROGRAMS WITHOUT WRITE ERROR STATUS SET?
No
ENTER MASTER MODE
IS REQUEST COUNT IN .SRQCT =0?
Yes
ALARM
No
Yes
No
DISPY
LFTOF
IN SLAVE MODE
E ANY SUBPROGRAMS RUNNING?
Yes
STORE 0 IN MAINL
ROADBLOCK
SUBPROGRAM REQUIRING ATTENTION BREAKS ROADBLOCK TO ENTER MAIN LEVEL HERE
No
ADD 1 TO MAINL
LFTOF
DISPX
ANY PRINTER OR PUNCH JOBS WAITING
No
ALDUN
Yes
ENTER MASTER MODE
ALARM
LOAD QR WITH VALUES TO ALERT IN 1 MINUTE
.CALL .MDISP EP13
ALDUO

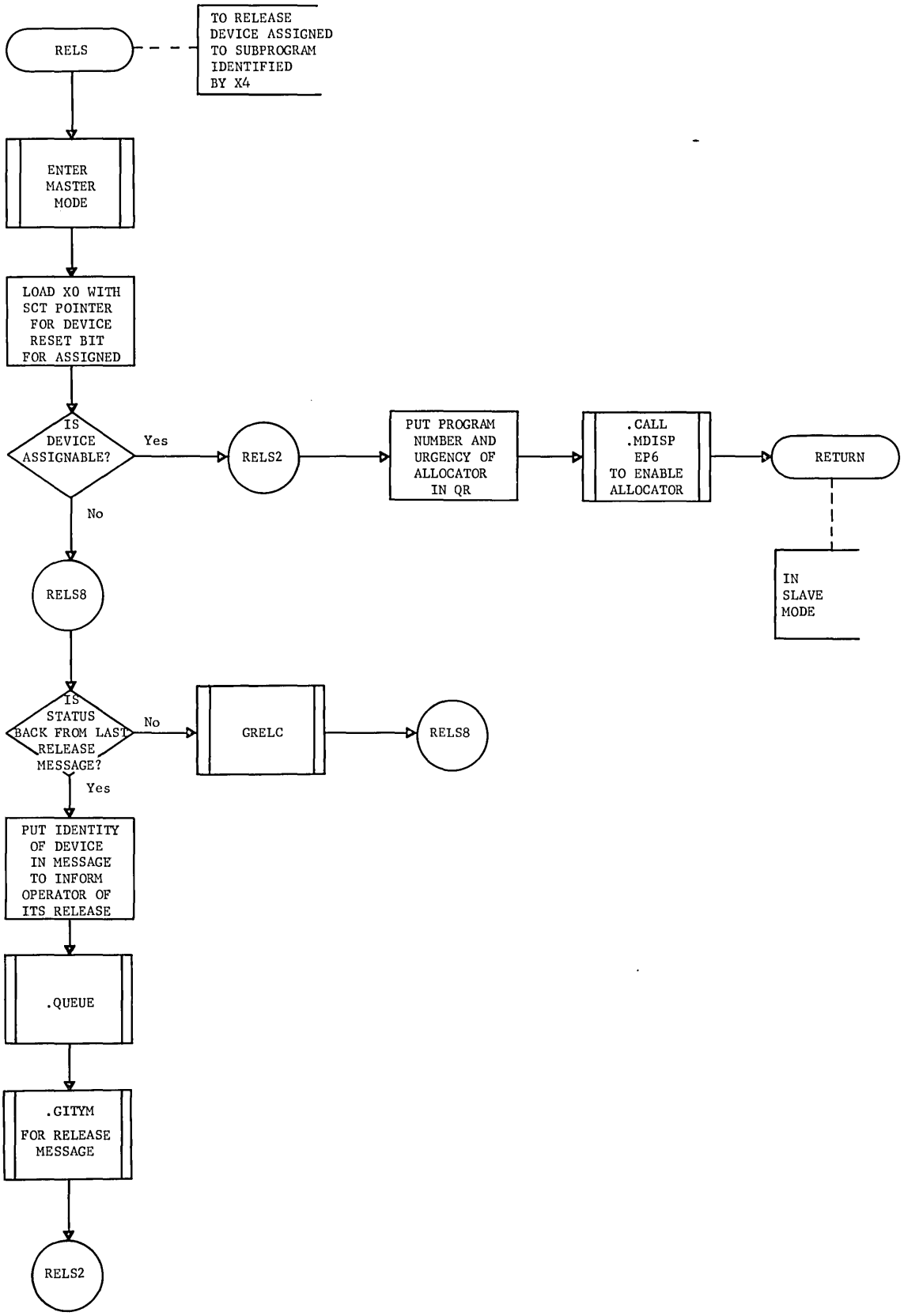

DISP
.MGEOT
DISPD
SAVE X4 IN LFTOF TO SHOW SUBPROGRAM LAST SERVICED
FMEOR =0?
No
TURN OFF FMEOR STATE
UPPER =LOWER OF NRPTS?
Yes
STACT
No
ADD 1 TO LOWER OF NRPTS FOR REPORT COUNT, LOAD X1 WITH JLIST POINTER FROM JLPTR LOWER
STRPT
Yes
DISPF
FMEOJ =0?
Yes
DISPJ
FMERM =0?
No
DISPG
No
Yes
FIND JOB IDENTIFIED IN JOBID IN SNUML TABLE, MARK DESTINATION JUST COMPLETED AS FINISHED, BUILD ACCOUNTING RECORD (TYPE 2)
STORE IC+2 IN RMCCSW, 2 IN DATA, ID OF LINE IN DATA+1, AND TERMINAL TYPE IN DATA+2
ENTER MASTER MODE
STRMT
.CALL .MIOS EP 13 TO WRITE ACCOUNTING DATA
FMDRM =0?
Yes
.CALL .MGOU1 EP 2 FOR END BANNER
RETURN TO SLAVE MODE, SET FCWEB AND FCWIN ON, ISSUE WRITE COMMAND FOR BANNER
1
No
DISPR
IN SLAVE MODE


1
DID LAST BLINK RELEASE GIVE 5 OR MORE?
No
HAVE WE EVER RELEASED BLINKS?
Yes
DID WE HAVE OVER 20 WHEN WE STARTED LAST RELEASE?
Yes
DISPV
Yes
No
No
DISPV
FLUSH
RELEASE ANY BLINKS POSSIBLE
DISPV
HAS I/O FOR BANNER FINISHED?
No
RELINQUISH CONTROL
DISPV
Yes
DISPR
TURN OFF ALL STATES EXCEPT FMINX, FMDPR, FMDPU, FMDRM, FMERM, FMRLS
IS DEVICE PRINTER?
No
DISPH
IS DEVICE PUNCH?
No
DISPJ
Yes
Yes
LOAD QU WITH -1, SET ZERO INDICATOR FOR NUMBER OF PRINTER JOBS WAITING
LOAD QR WITH -1, SET ZERO INDICATOR FOR NUMBER OF PUNCH JOBS WAITING
DISPI
DISPI


DISP
.MGEOT
DISPI
IS
ZERO INDICATOR
ON?
No
FMRLS
=0?
No
2
Yes
Yes
2
STRJB
ADJUST NUMBER
AND TYPE OF
DEVICES RUNNING
IN NDEVR BY
CONTENTS OF QR
RELS
RELEASE
DEVICE
DISPG
TURN OFF ALL
STATE BITS FOR
SUBPROGRAM,
STORE 1 IN UPPER
OF MAINL FOR
RELEASE FLAG
WAS THIS
SUBPROGRAM
AT THE END OF
OUR MEMORY?
MEMS
STORE LOCATION OF
SUBPROGRAM IN EOMEM,
STORE 0 IN PNORG OF
THIS SUBPROGRAM, PUT
1024 IN QL
DISP
RELEASE
MEMORY
AS PER QR
MEMS
Yes
IS
SUBPROGRAM
AT HIGHEST MEMORY
LOCATION NOT IN
EXECUTION?
No
DISP

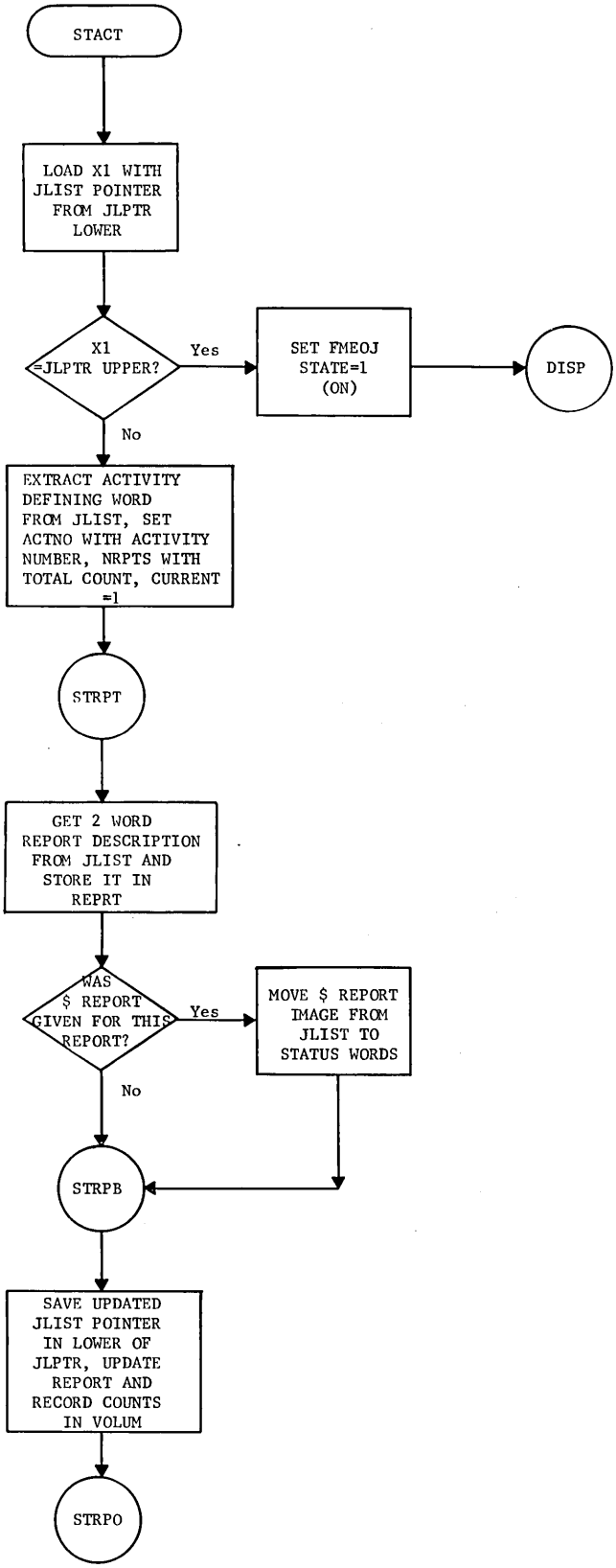

ALDUN
ENTER MASTER MODE
.CRTSM =0?
Yes
ALDUO
No
SEARCH .CRSNB LIST FOR SLAVE PROGRAM WHOSE LAST CHARACTER IN LIST ENTRY IS=17 OCTAL
FIND ONE?
No
.QUEUE
.GITYM TO TYPE EXEUNT GEOT
LASTC
Yes
LASTC
SUBRQ
LOOK AT INPUT QUEUE AGAIN
ALDUO
ROADBLOCK
SET URGENCY TO ZERO
.CALL .MDISP EP11
ALDUX
RETURN TO SLAVE MODE
DISP

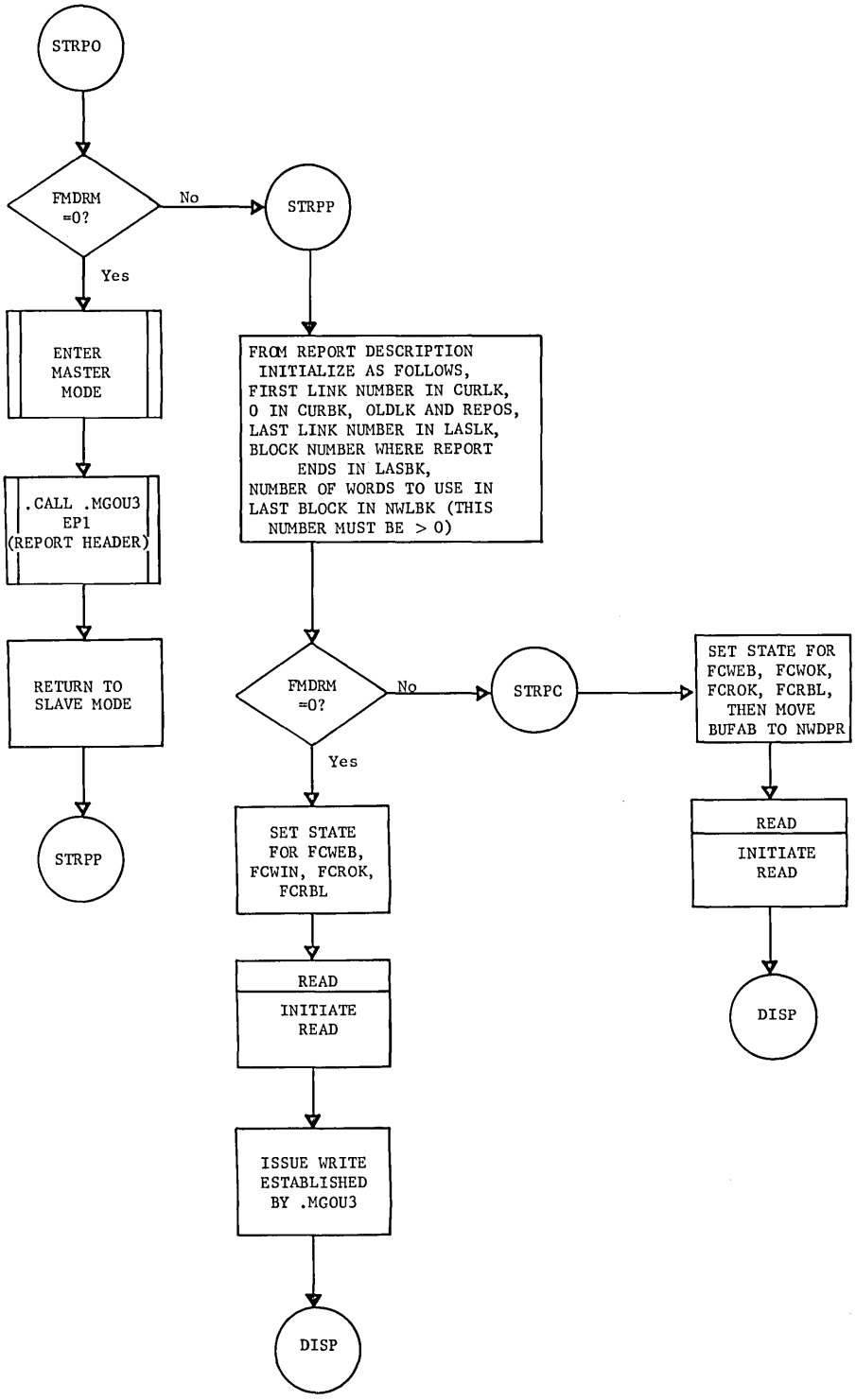

DISP
.MGEOT
ERROR
FMEXC
=0?
No
IS
UPPER OF
MAINL≠0 SHOWING
RELEASE?
No
3
Yes
Yes
3
HRLSE
IS
STATUS
BACK FROM THE
LAST
MESSAGE?
No
DISPB
Yes
ENTER
MASTER
MODE
GET CURRENT
TIME OF DAY
AND DESCRIPTION
OF OUTPUT DEVICE,
THEN RETURN TO
SLAVE MODE
HAVE
30 SECONDS
PASSED SINCE TIME
CELL WAS
SET?
No
DISPB
Yes
RESET TIME
CELL TO
PRESENT TIME
4
4
FMEXC
=0?
No
HRLSE
Yes
PUT DEVICE
IDENTIFICATION
INTO MESSAGE.
PLEASE,
FIX SYSOUT...
ENTER
MASTER
MODE
.QUEUE
.GITYM
TYPE MESSAGE
PLEASE
DISP
IN
SLAVE
MODE


RELS
TO RELEASE DEVICE ASSIGNED TO SUBPROGRAM IDENTIFIED BY X4
ENTER MASTER MODE
LOAD X0 WITH SCT POINTER FOR DEVICE RESET BIT FOR ASSIGNED
IS DEVICE ASSIGNABLE?
Yes
RELS2
PUT PROGRAM NUMBER AND URGENCY OF ALLOCATOR IN QR
.CALL .MDISP EP6 TO ENABLE ALLOCATOR
RETURN
IN SLAVE MODE
No
RELS8
IS STATUS BACK FROM LAST RELEASE MESSAGE?
No
GRELC
RELS8
Yes
PUT IDENTITY OF DEVICE IN MESSAGE TO INFORM OPERATOR OF ITS RELEASE
.QUEUE
.GITYM FOR RELEASE MESSAGE
RELS2

STACT
.MGEOT

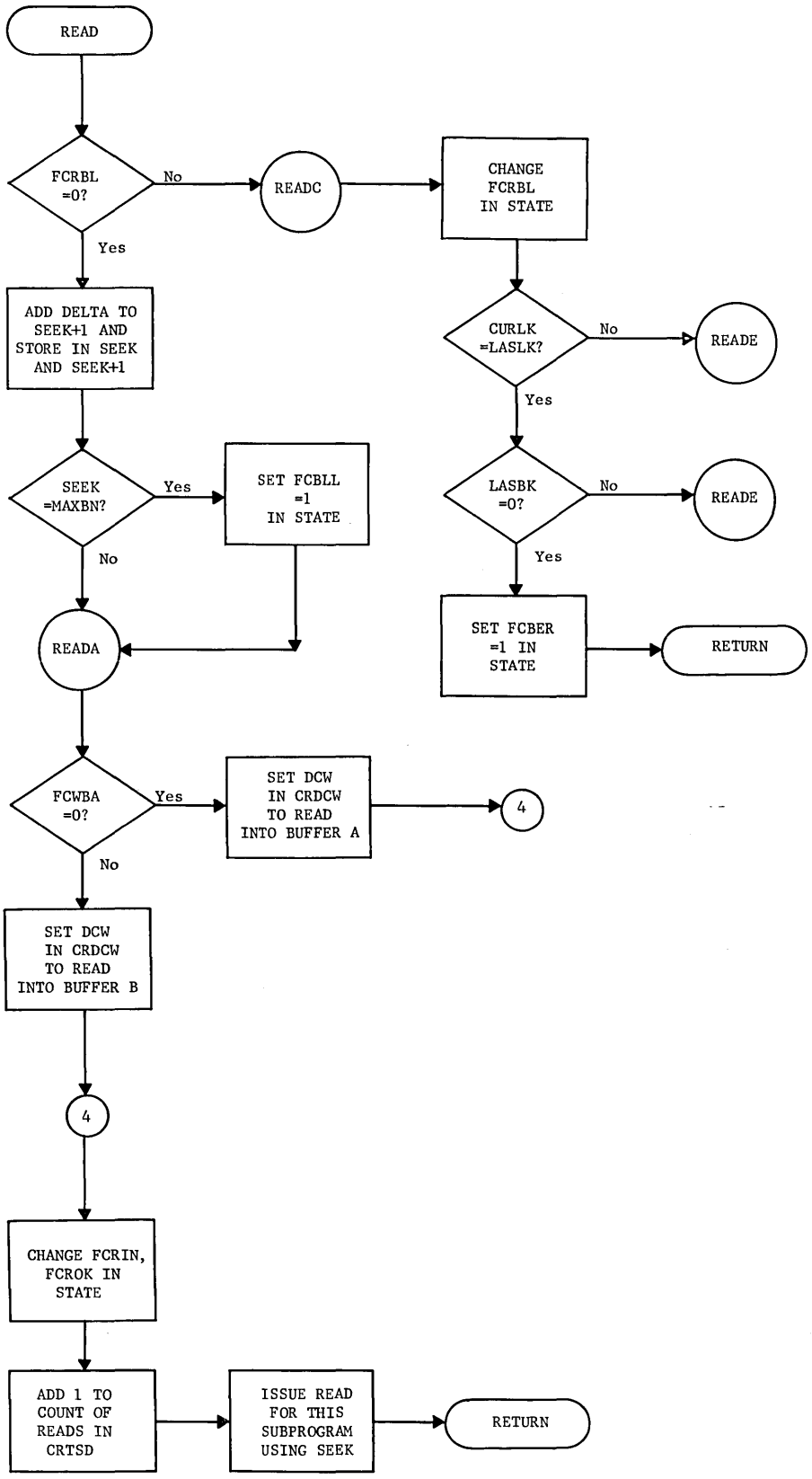
START REPORT/ACTIVITY

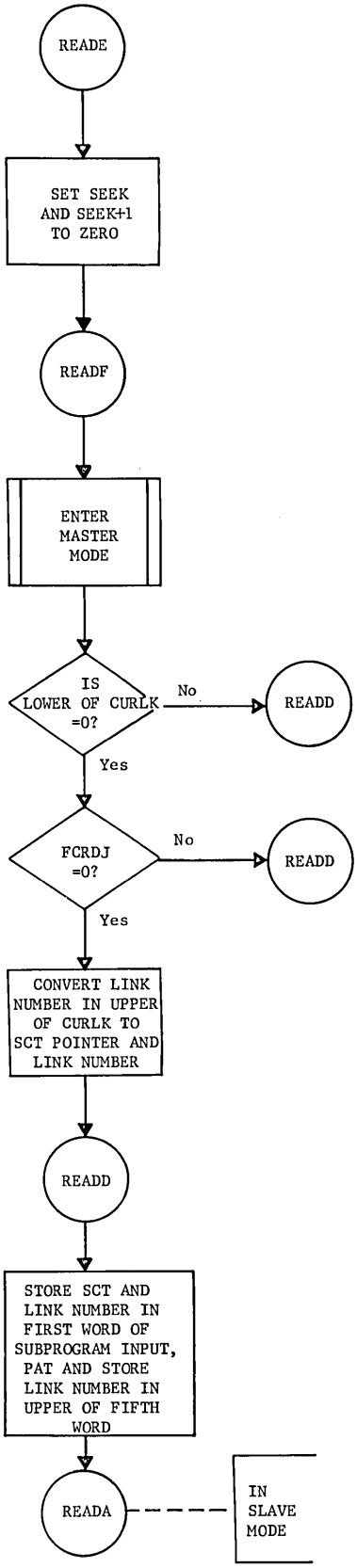



STRPO
FMDRM =0?
No
STRPP
Yes
ENTER MASTER MODE
.CALL .MGOU3 EP1 (REPORT HEADER)
RETURN TO SLAVE MODE
STRPP
FROM REPORT DESCRIPTION INITIALIZE AS FOLLOWS, FIRST LINK NUMBER IN CURLK, 0 IN CURBK, OLDLK AND REPOS, LAST LINK NUMBER IN LASLK, BLOCK NUMBER WHERE REPORT ENDS IN LASBK, NUMBER OF WORDS TO USE IN LAST BLOCK IN NWLBK (THIS NUMBER MUST BE > 0)
FMDRM =0?
No
STRPC
SET STATE FOR FCWEB, FCWOK, FCROK, FCRBL, THEN MOVE BUFAB TO NWDPR
READ
INITIATE READ
DISP
Yes
SET STATE FOR FCWEB, FCWIN, FCROK, FCRBL
READ
INITIATE READ
ISSUE WRITE ESTABLISHED BY .MGOU3
DISP

READ
.MGEOT

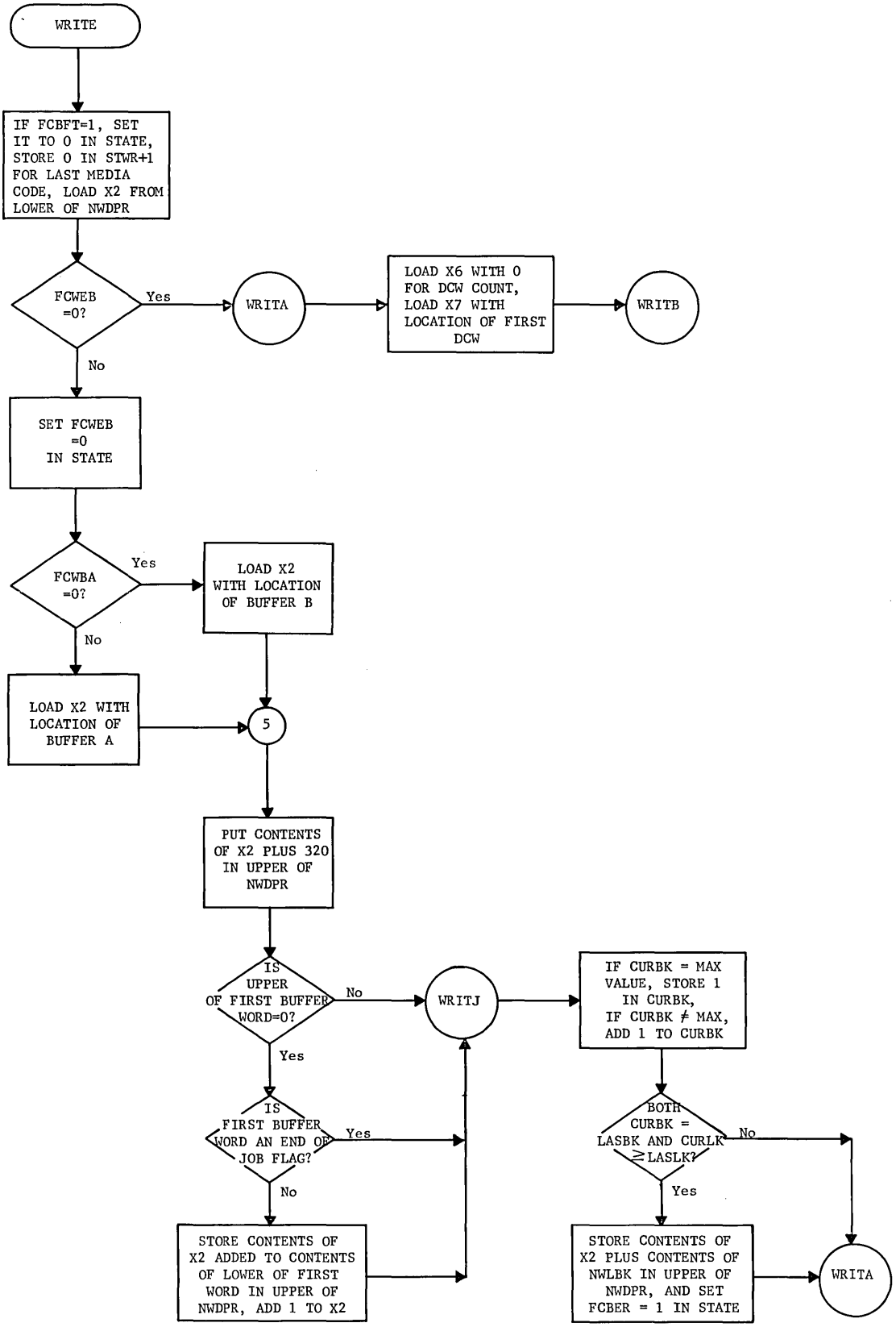
READ INTO BUFFER

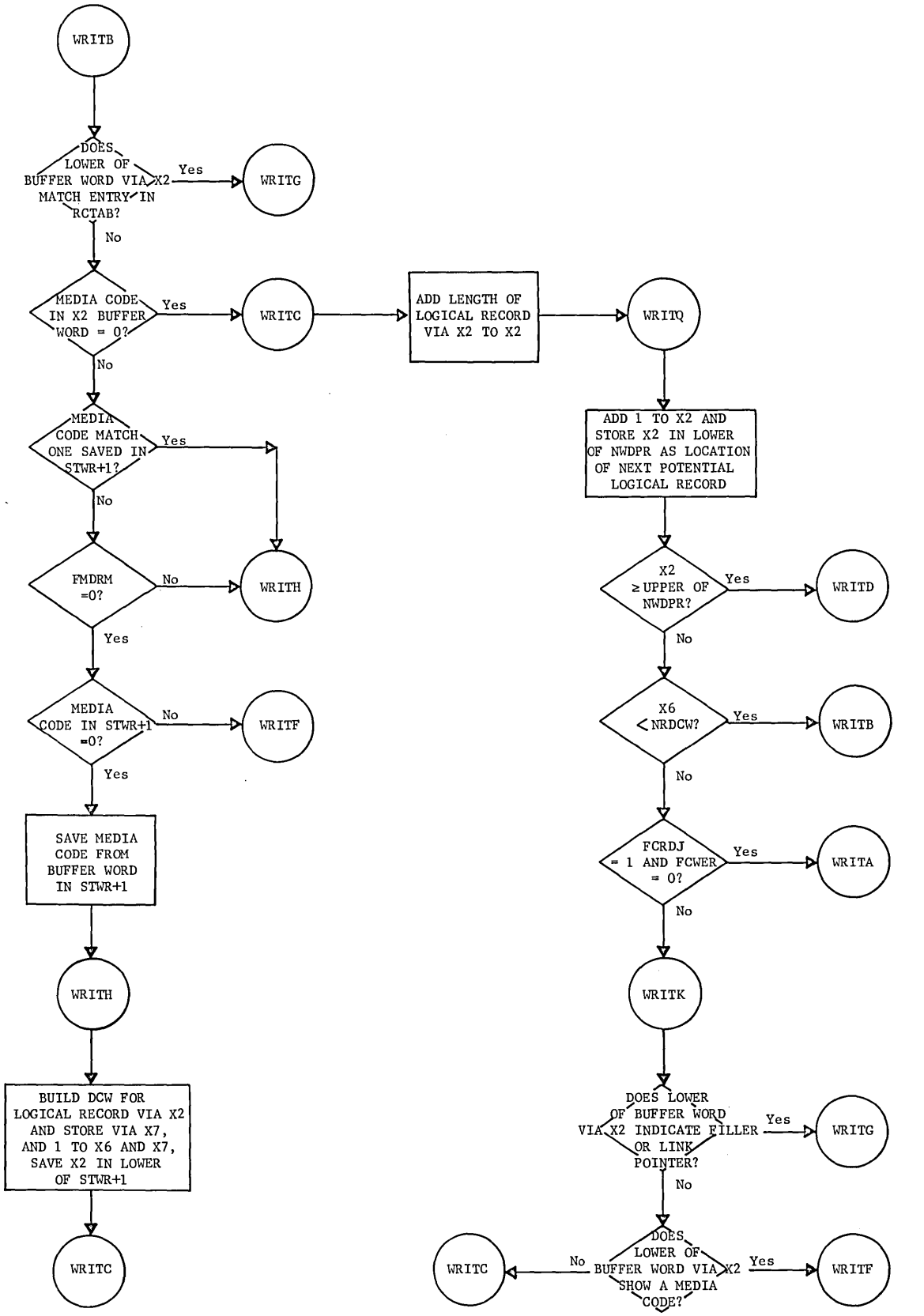


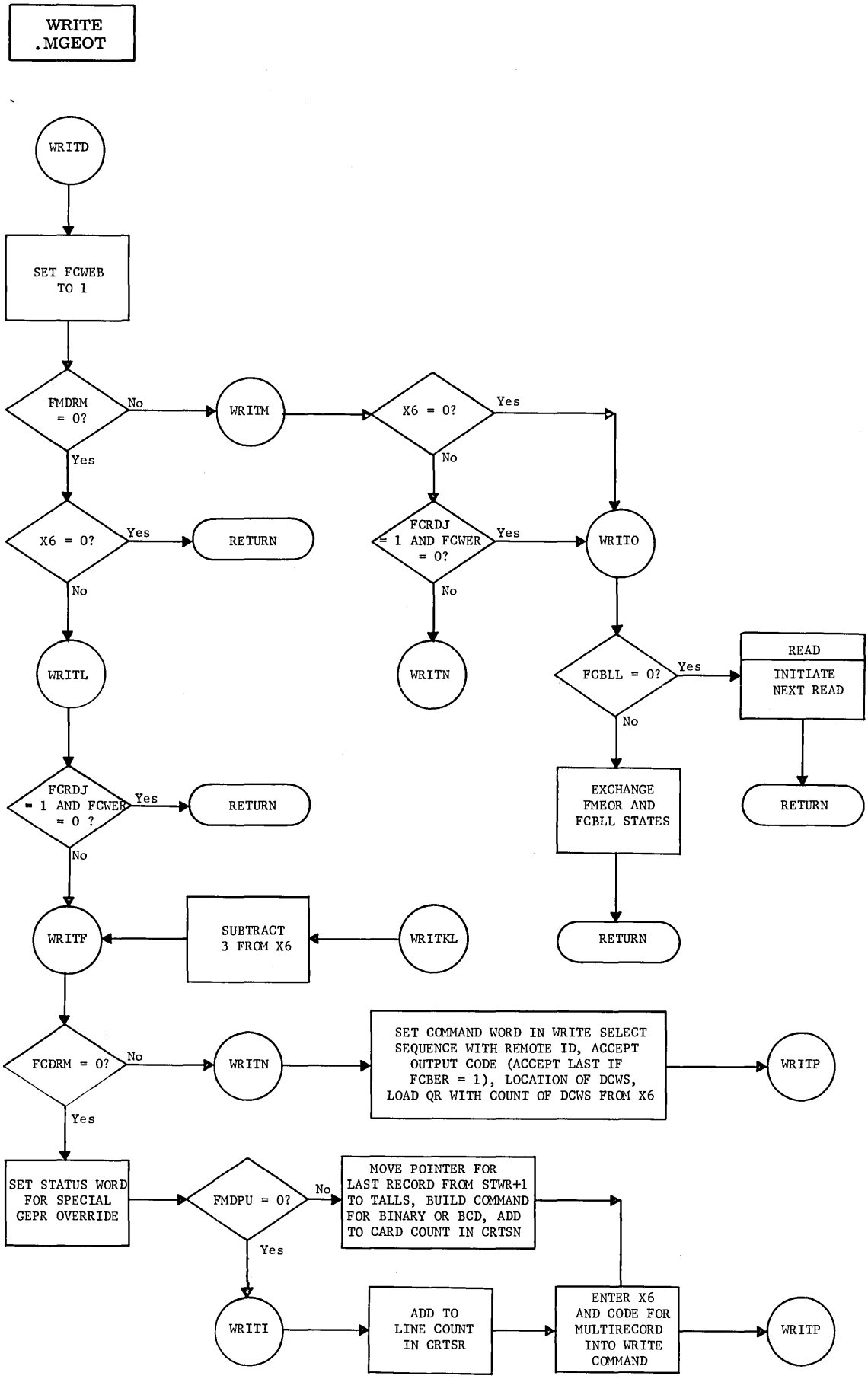

READE
SET SEEK AND SEEK+1 TO ZERO
READF
ENTER MASTER MODE
IS LOWER OF CURLK =0?
No
READD
Yes
FCRDJ =0?
No
READD
Yes
CONVERT LINK NUMBER IN UPPER OF CURLK TO SCT POINTER AND LINK NUMBER
READD
STORE SCT AND LINK NUMBER IN FIRST WORD OF SUBPROGRAM INPUT, PAT AND STORE LINK NUMBER IN UPPER OF FIFTH WORD
READA
IN SLAVE MODE

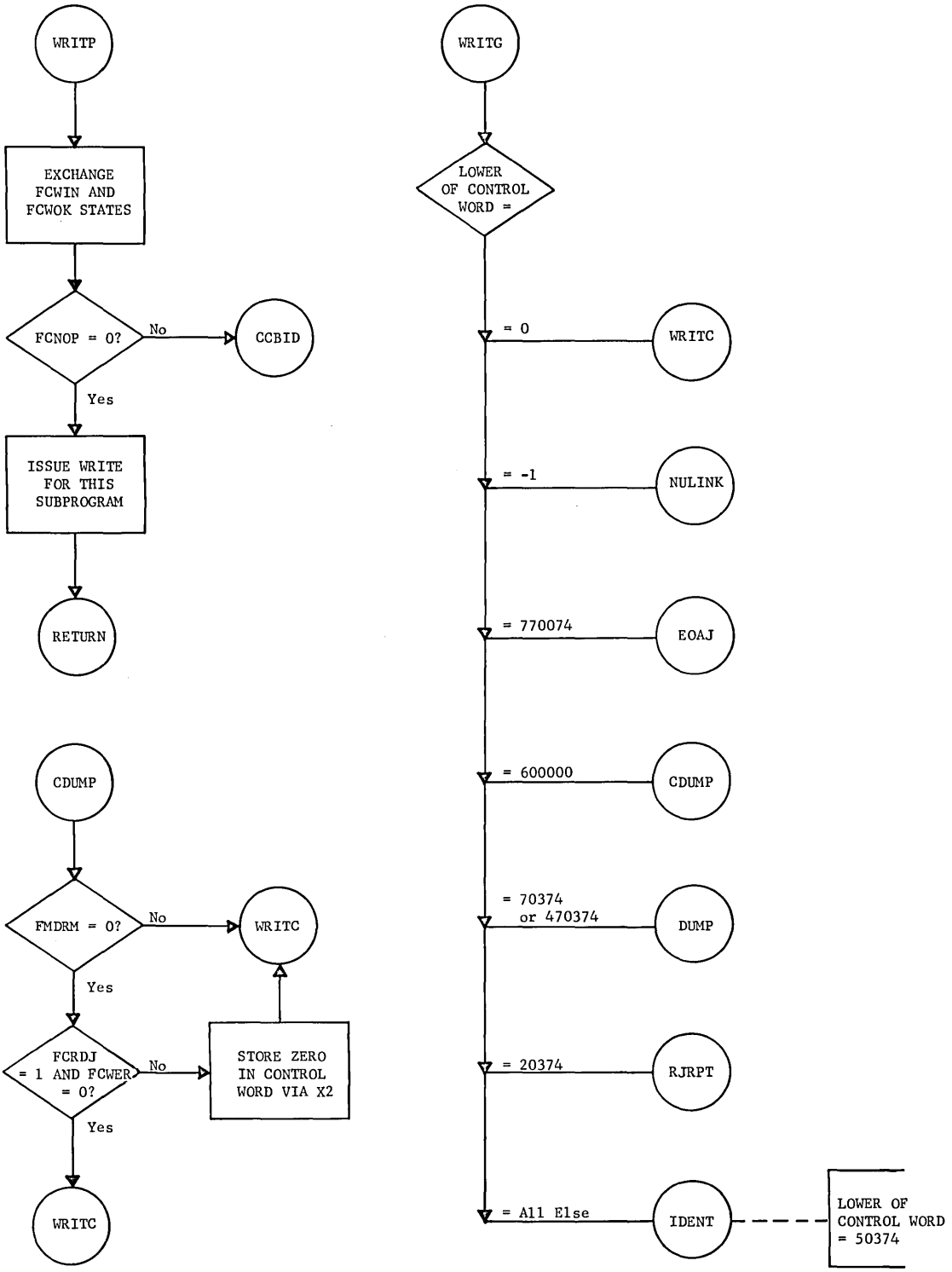
WRITE
.MGEOT

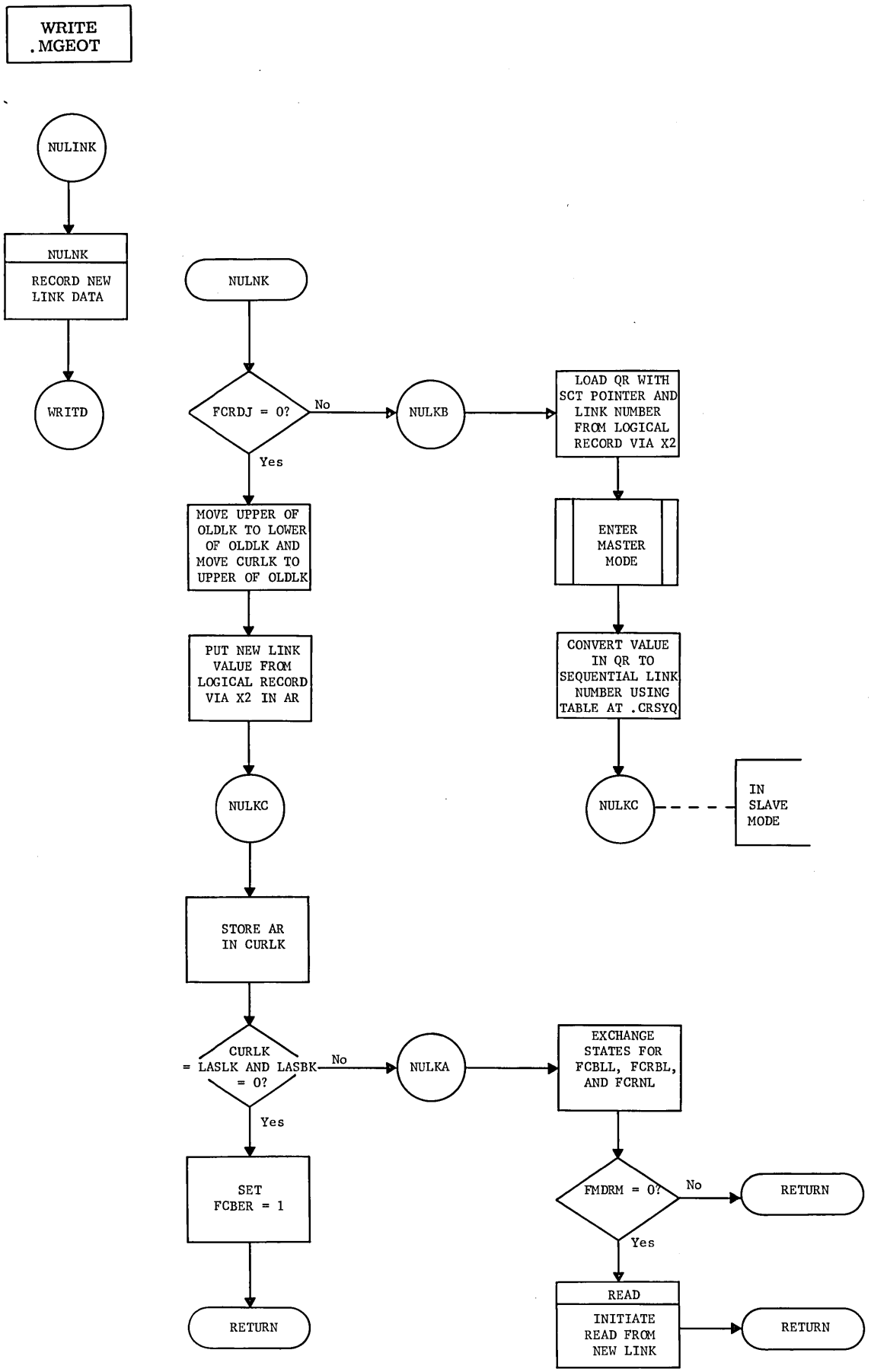
WRITE FROM BUFFER

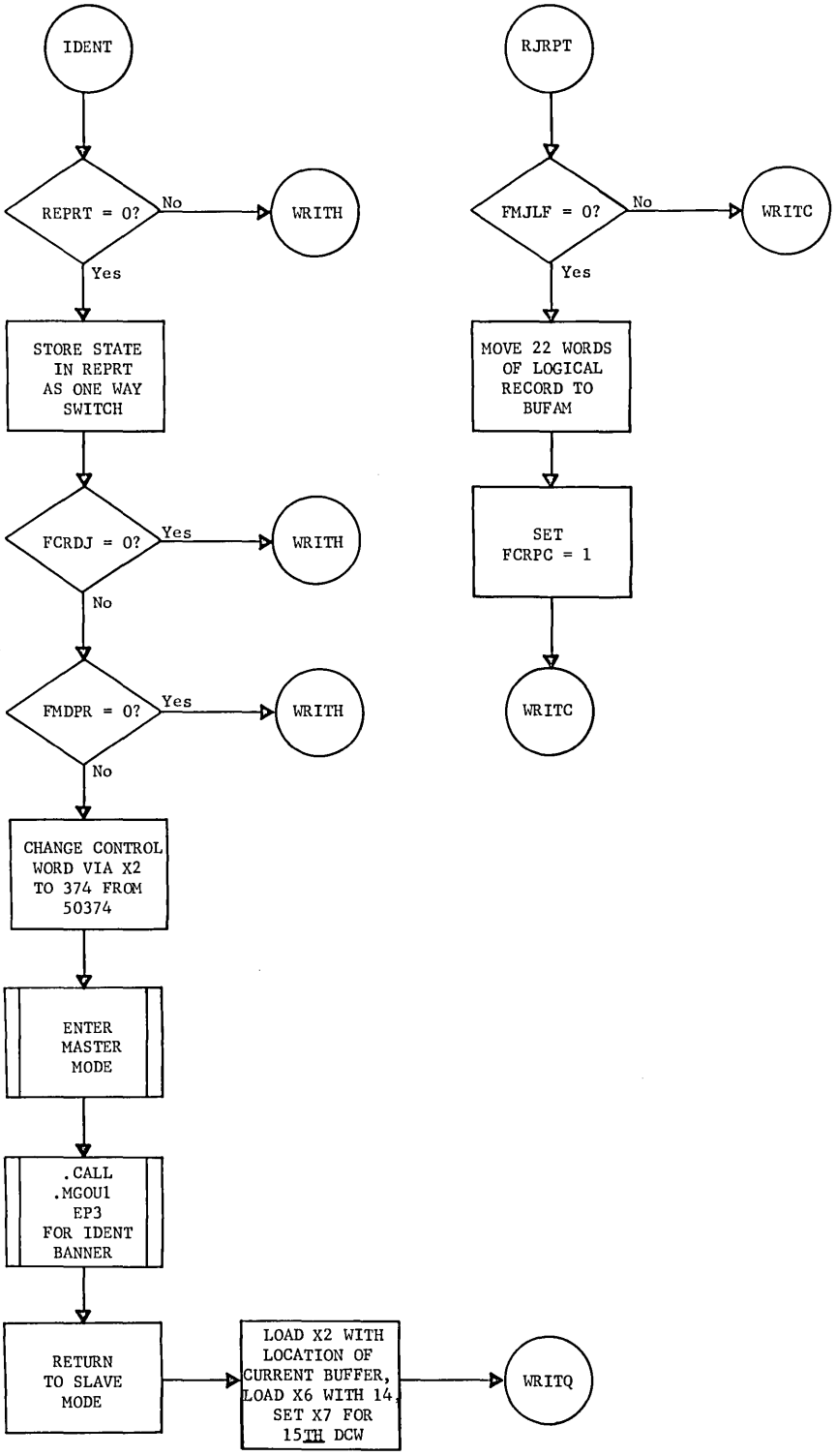



WRITB
DOES LOWER OF BUFFER WORD VIA X2 MATCH ENTRY IN RCTAB?
Yes
WRITG
No
MEDIA CODE IN X2 BUFFER WORD = 0?
Yes
WRITC
No
MEDIA CODE MATCH ONE SAVED IN STWR+1?
Yes
No
FMDRM =0?
No
WRITH
Yes
MEDIA CODE IN STWR+1 =0?
No
WRITF
Yes
SAVE MEDIA CODE FROM BUFFER WORD IN STWR+1
WRITH
BUILD DCW FOR LOGICAL RECORD VIA X2 AND STORE VIA X7, AND 1 TO X6 AND X7, SAVE X2 IN LOWER OF STWR+1
WRITC
ADD LENGTH OF LOGICAL RECORD VIA X2 TO X2
WRITQ
ADD 1 TO X2 AND STORE X2 IN LOWER OF NWDPR AS LOCATION OF NEXT POTENTIAL LOGICAL RECORD
X2 ≥ UPPER OF NWDPR?
Yes
WRITD
No
X6 < NRDCW?
Yes
WRITB
No
FCRDJ = 1 AND FCWER = 0?
Yes
WRITA
No
WRITK
DOES LOWER OF BUFFER WORD VIA X2 INDICATE FILLER OR LINK POINTER?
Yes
WRITG
No
DOES LOWER OF BUFFER WORD VIA X2 SHOW A MEDIA CODE?
No
WRITC
Yes
WRITF

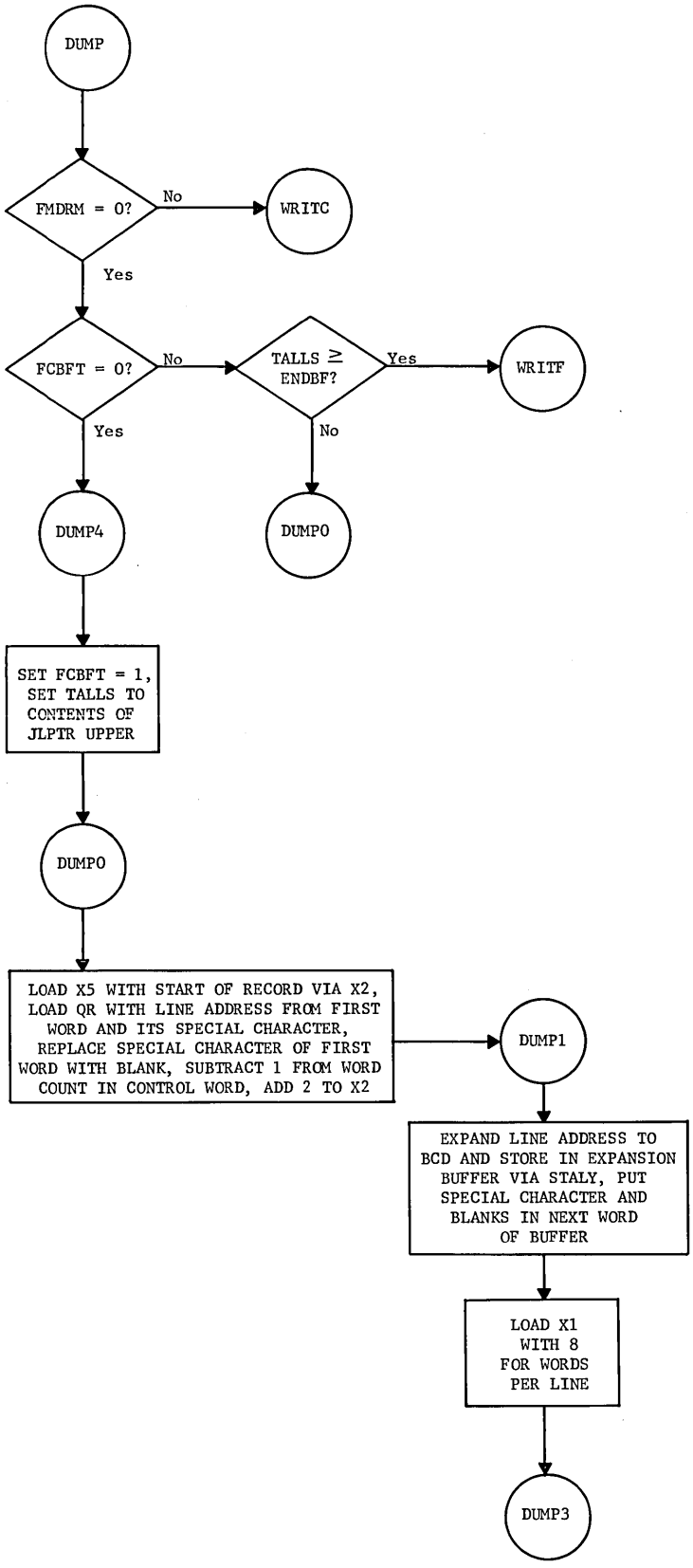

WRITE
.MGEOT
WRITD
SET FCWEB
TO 1
FMDRM
= 0?
No
WRITM
X6 = 0?
Yes
No
Yes
X6 = 0?
Yes
RETURN
FCRDJ
= 1 AND FCWER
= 0?
Yes
WRITO
No
No
WRITL
WRITN
FCBLL = 0?
Yes
READ
INITIATE
NEXT READ
No
FCRDJ
= 1 AND FCWER
= 0 ?
Yes
RETURN
EXCHANGE
FMEOR AND
FCBLL STATES
RETURN
No
WRITF
SUBTRACT
3 FROM X6
WRITKL
RETURN
FCDRM = 0?
No
WRITN
SET COMMAND WORD IN WRITE SELECT
SEQUENCE WITH REMOTE ID, ACCEPT
OUTPUT CODE (ACCEPT LAST IF
FCBER = 1), LOCATION OF DCWS,
LOAD QR WITH COUNT OF DCWS FROM X6
WRITP
Yes
SET STATUS WORD
FOR SPECIAL
GEPR OVERRIDE
FMDPU = 0?
No
MOVE POINTER FOR
LAST RECORD FROM STWR+1
TO TALLS, BUILD COMMAND
FOR BINARY OR BCD, ADD
TO CARD COUNT IN CRTSN
Yes
WRITI
ADD TO
LINE COUNT
IN CRTSR
ENTER X6
AND CODE FOR
MULTIRECORD
INTO WRITE
COMMAND
WRITP

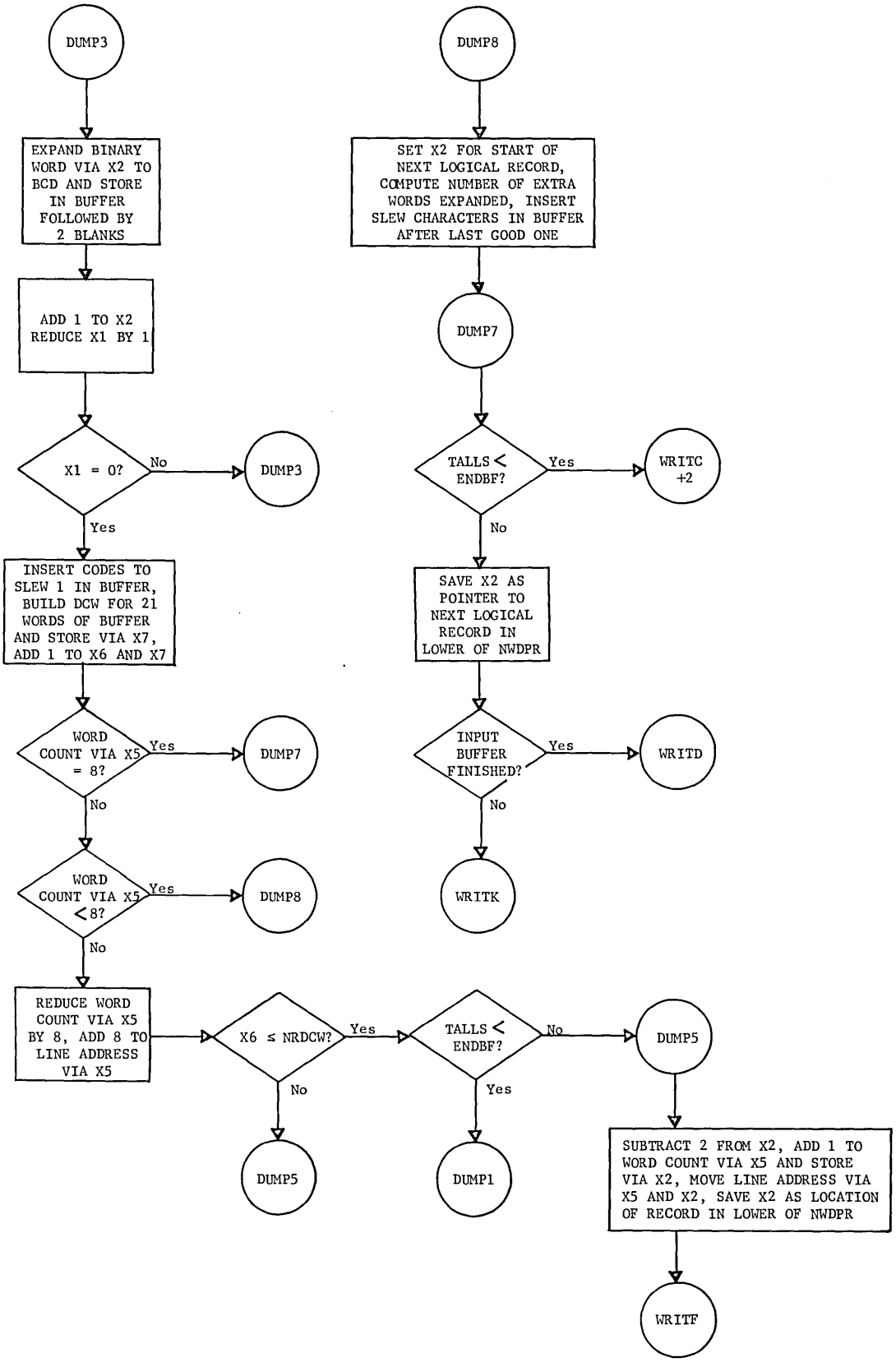

WRITP
EXCHANGE FCWIN AND FCWOK STATES
FCNOP = 0?
No
CCBID
Yes
ISSUE WRITE FOR THIS SUBPROGRAM
RETURN
CDUMP
FMDRM = 0?
No
WRITC
Yes
FCRDJ = 1 AND FCWER = 0?
No
STORE ZERO IN CONTROL WORD VIA X2
Yes
WRITC
WRITG
LOWER OF CONTROL WORD =
= 0
WRITC
= -1
NULINK
= 770074
EOAJ
= 600000
CDUMP
= 70374 or 470374
DUMP
= 20374
RJRPT
= All Else
IDENT
LOWER OF CONTROL WORD = 50374

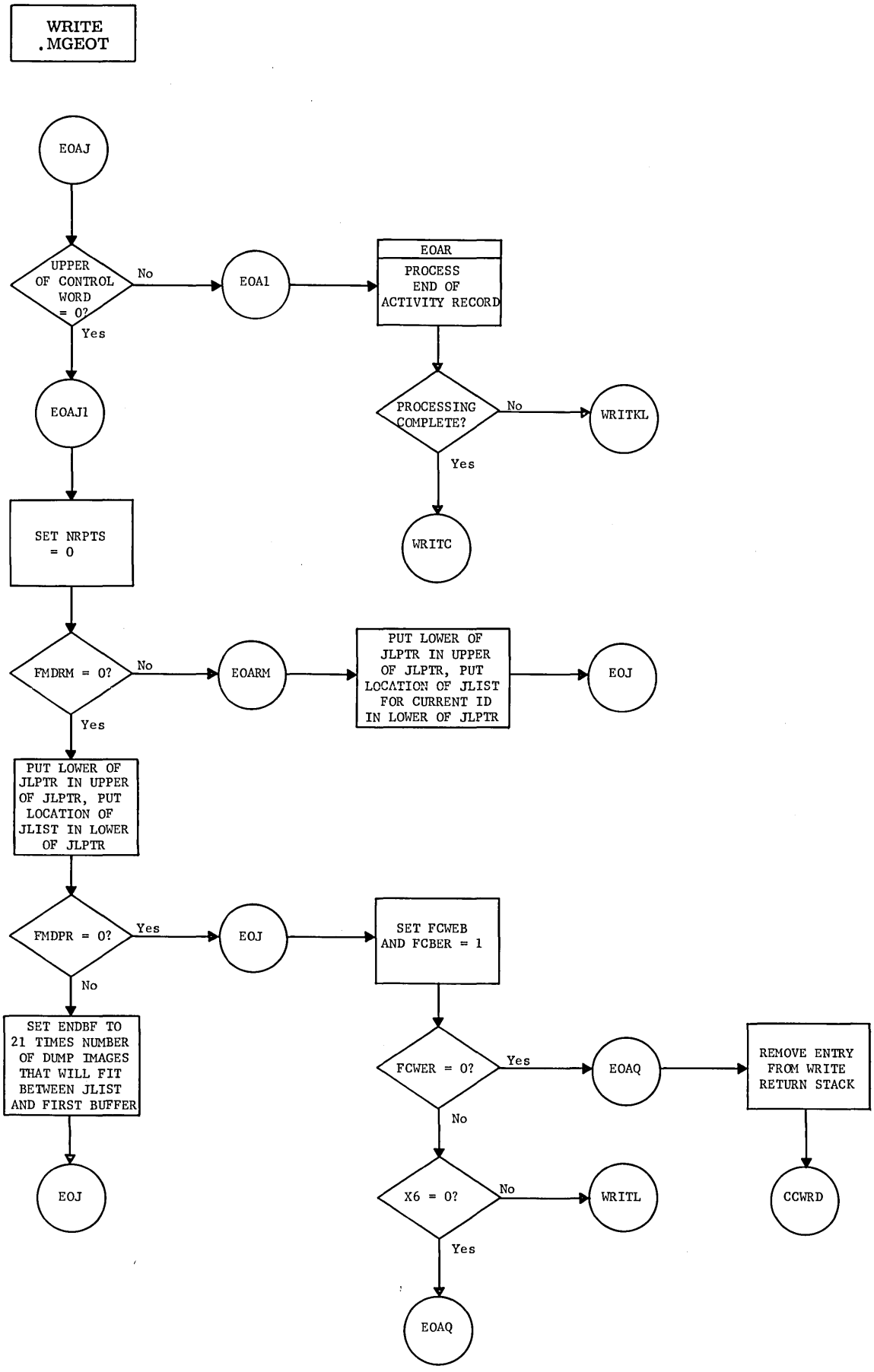

WRITE
.MGEOT
NULINK
NULNK
RECORD NEW
LINK DATA
WRITD
NULNK
FCRDJ = 0?
No
Yes
NULKB
LOAD QR WITH
SCT POINTER AND
LINK NUMBER
FROM LOGICAL
RECORD VIA X2
ENTER
MASTER
MODE
CONVERT VALUE
IN QR TO
SEQUENTIAL LINK
NUMBER USING
TABLE AT .CRSYQ
NULKC
IN
SLAVE
MODE
MOVE UPPER OF
OLDLK TO LOWER
OF OLDLK AND
MOVE CURLK TO
UPPER OF OLDLK
PUT NEW LINK
VALUE FROM
LOGICAL RECORD
VIA X2 IN AR
NULKC
STORE AR
IN CURLK
CURLK
= LASLK AND LASBK
= 0?
No
Yes
NULKA
EXCHANGE
STATES FOR
FCBLL, FCRBL,
AND FCRNL
SET
FCBER = 1
FMDRM = 0?
No
Yes
RETURN
READ
INITIATE
READ FROM
NEW LINK
RETURN
RETURN


IDENT
REPRT = 0?
No
WRITH
Yes
STORE STATE IN REPRT AS ONE WAY SWITCH
FCRDJ = 0?
Yes
WRITH
No
FMDPR = 0?
Yes
WRITH
No
CHANGE CONTROL WORD VIA X2 TO 374 FROM 50374
ENTER MASTER MODE
.CALL .MGOU1 EP3 FOR IDENT BANNER
RETURN TO SLAVE MODE
LOAD X2 WITH LOCATION OF CURRENT BUFFER, LOAD X6 WITH 14, SET X7 FOR 15TH DCW
WRITQ
RJRPT
FMJLF = 0?
No
WRITC
Yes
MOVE 22 WORDS OF LOGICAL RECORD TO BUFAM
SET FCRPC = 1
WRITC

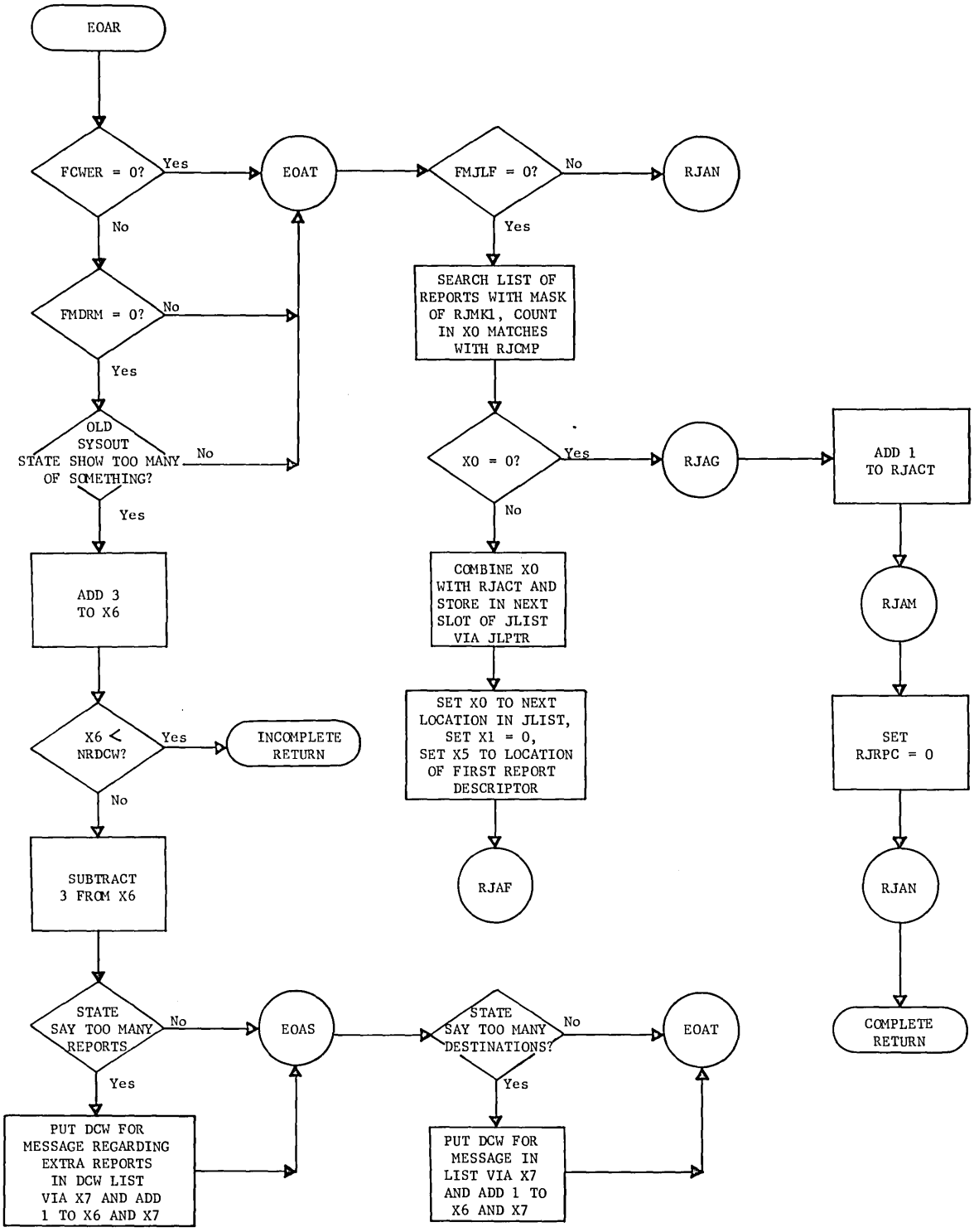
WRITE
.MGEOT

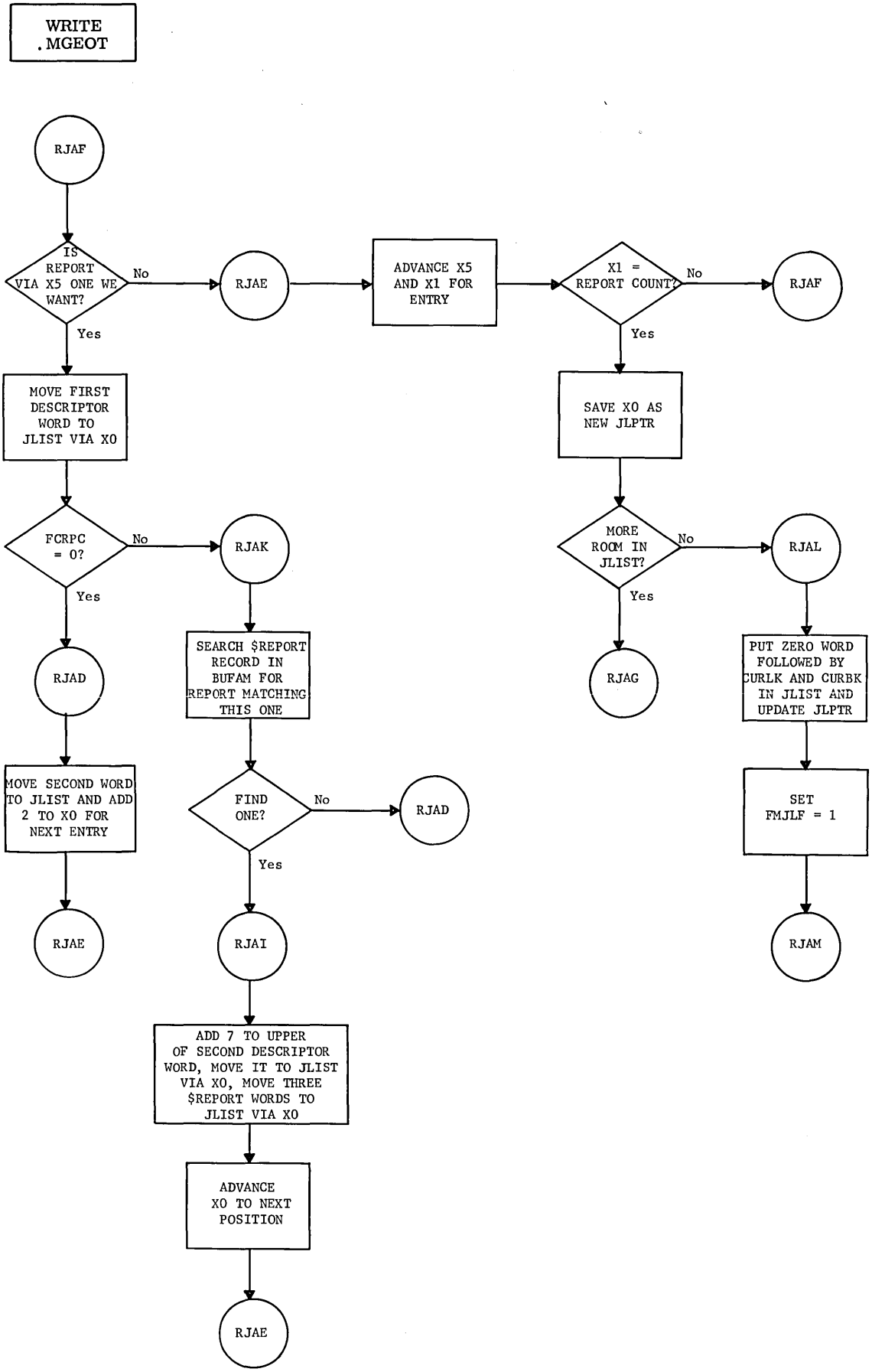

DUMP
FMDRM = 0?
No
WRITC
Yes
FCBFT = 0?
No
TALLS ≥ ENDBF?
Yes
WRITF
Yes
No
DUMP4
DUMP0
SET FCBFT = 1, SET TALLS TO CONTENTS OF JLPTR UPPER
DUMP0
LOAD X5 WITH START OF RECORD VIA X2, LOAD QR WITH LINE ADDRESS FROM FIRST WORD AND ITS SPECIAL CHARACTER, REPLACE SPECIAL CHARACTER OF FIRST WORD WITH BLANK, SUBTRACT 1 FROM WORD COUNT IN CONTROL WORD, ADD 2 TO X2
DUMP1
EXPAND LINE ADDRESS TO BCD AND STORE IN EXPANSION BUFFER VIA STALY, PUT SPECIAL CHARACTER AND BLANKS IN NEXT WORD OF BUFFER
LOAD X1 WITH 8 FOR WORDS PER LINE
DUMP3


DUMP3
EXPAND BINARY WORD VIA X2 TO BCD AND STORE IN BUFFER FOLLOWED BY 2 BLANKS
ADD 1 TO X2 REDUCE X1 BY 1
X1 = 0?
No
DUMP3
Yes
INSERT CODES TO SLEW 1 IN BUFFER, BUILD DCW FOR 21 WORDS OF BUFFER AND STORE VIA X7, ADD 1 TO X6 AND X7
WORD COUNT VIA X5 = 8?
Yes
DUMP7
No
WORD COUNT VIA X5 < 8?
Yes
DUMP8
No
REDUCE WORD COUNT VIA X5 BY 8, ADD 8 TO LINE ADDRESS VIA X5
X6 ≤ NRDCW?
Yes
No
DUMP5
TALLS < ENDBF?
No
Yes
DUMP1
DUMP5
SUBTRACT 2 FROM X2, ADD 1 TO WORD COUNT VIA X5 AND STORE VIA X2, MOVE LINE ADDRESS VIA X5 AND X2, SAVE X2 AS LOCATION OF RECORD IN LOWER OF NWDPR
WRITF
DUMP8
SET X2 FOR START OF NEXT LOGICAL RECORD, COMPUTE NUMBER OF EXTRA WORDS EXPANDED, INSERT SLEW CHARACTERS IN BUFFER AFTER LAST GOOD ONE
DUMP7
TALLS < ENDBF?
Yes
WRITC +2
No
SAVE X2 AS POINTER TO NEXT LOGICAL RECORD IN LOWER OF NWDPR
INPUT BUFFER FINISHED?
Yes
WRITD
No
WRITK


WRITE
.MGEOT
EOAJ
UPPER OF CONTROL WORD = 0?
No
Yes
EOA1
EOAR
PROCESS END OF ACTIVITY RECORD
PROCESSING COMPLETE?
No
Yes
WRITKL
WRITC
EOAJ1
SET NRPTS = 0
FMDRM = 0?
No
Yes
EOARM
PUT LOWER OF JLPTR IN UPPER OF JLPTR, PUT LOCATION OF JLIST FOR CURRENT ID IN LOWER OF JLPTR
EOJ
PUT LOWER OF JLPTR IN UPPER OF JLPTR, PUT LOCATION OF JLIST IN LOWER OF JLPTR
FMDPR = 0?
Yes
No
EOJ
SET FCWEB AND FCBER = 1
SET ENDBF TO 21 TIMES NUMBER OF DUMP IMAGES THAT WILL FIT BETWEEN JLIST AND FIRST BUFFER
EOJ
FCWER = 0?
Yes
No
EOAQ
REMOVE ENTRY FROM WRITE RETURN STACK
CCWRD
X6 = 0?
No
Yes
WRITL
EOAQ

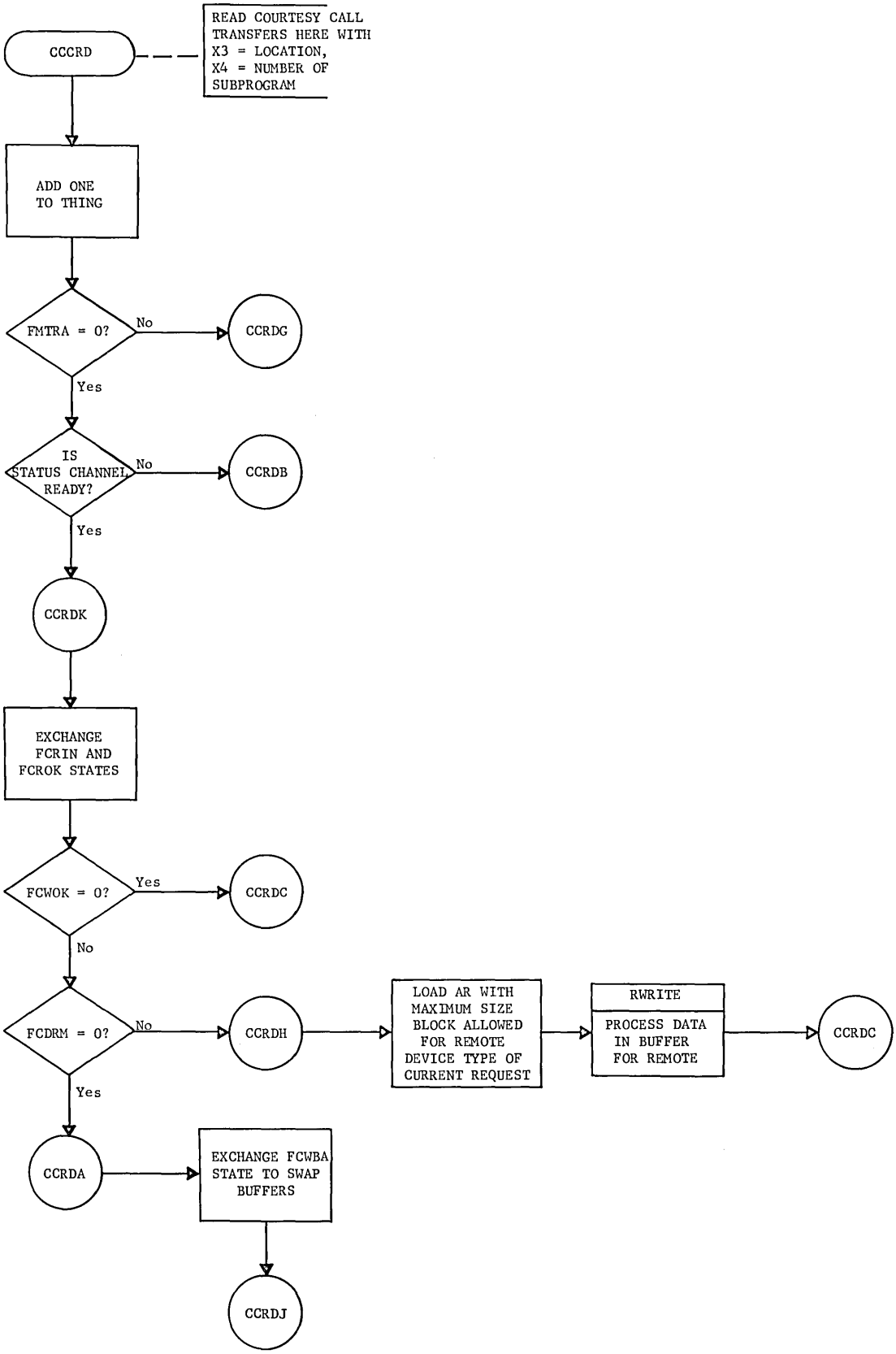
WRITE
.MGEOT

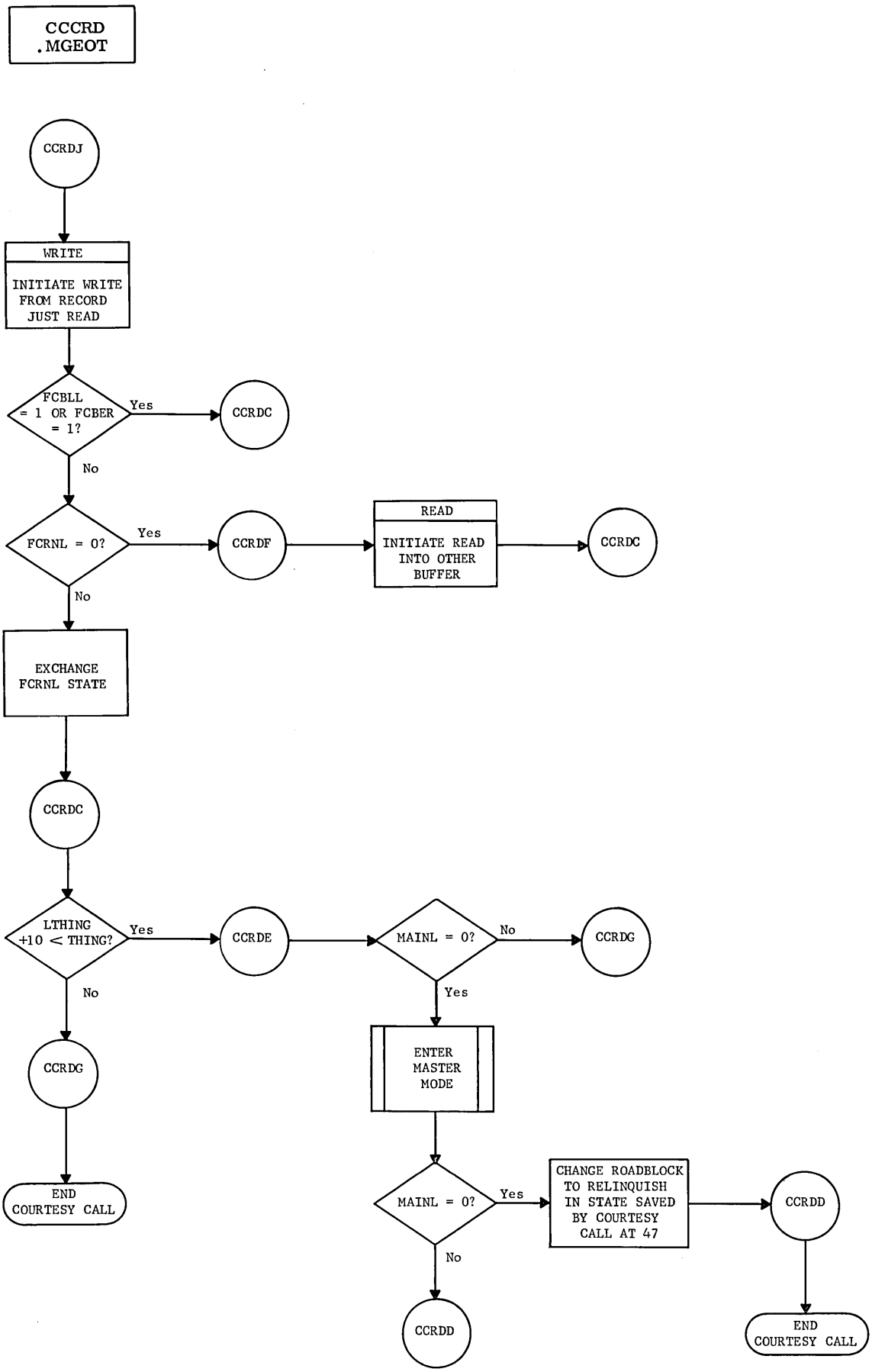

EOAR
FCWER = 0?
Yes
EOAT
No
FMDRM = 0?
No
Yes
OLD SYSOUT STATE SHOW TOO MANY OF SOMETHING?
No
Yes
ADD 3 TO X6
X6 < NRDCW?
Yes
INCOMPLETE RETURN
No
SUBTRACT 3 FROM X6
STATE SAY TOO MANY REPORTS
No
EOAS
Yes
PUT DCW FOR MESSAGE REGARDING EXTRA REPORTS IN DCW LIST VIA X7 AND ADD 1 TO X6 AND X7
FMJLF = 0?
No
RJAN
Yes
SEARCH LIST OF REPORTS WITH MASK OF RJMK1, COUNT IN XO MATCHES WITH RJCMP
XO = 0?
Yes
RJAG
No
COMBINE XO WITH RJACT AND STORE IN NEXT SLOT OF JLIST VIA JLPTR
SET XO TO NEXT LOCATION IN JLIST, SET X1 = 0, SET X5 TO LOCATION OF FIRST REPORT DESCRIPTOR
RJAF
STATE SAY TOO MANY DESTINATIONS?
No
EOAT
Yes
PUT DCW FOR MESSAGE IN LIST VIA X7 AND ADD 1 TO X6 AND X7
ADD 1 TO RJACT
RJAM
SET RJRPC = 0
RJAN
COMPLETE RETURN

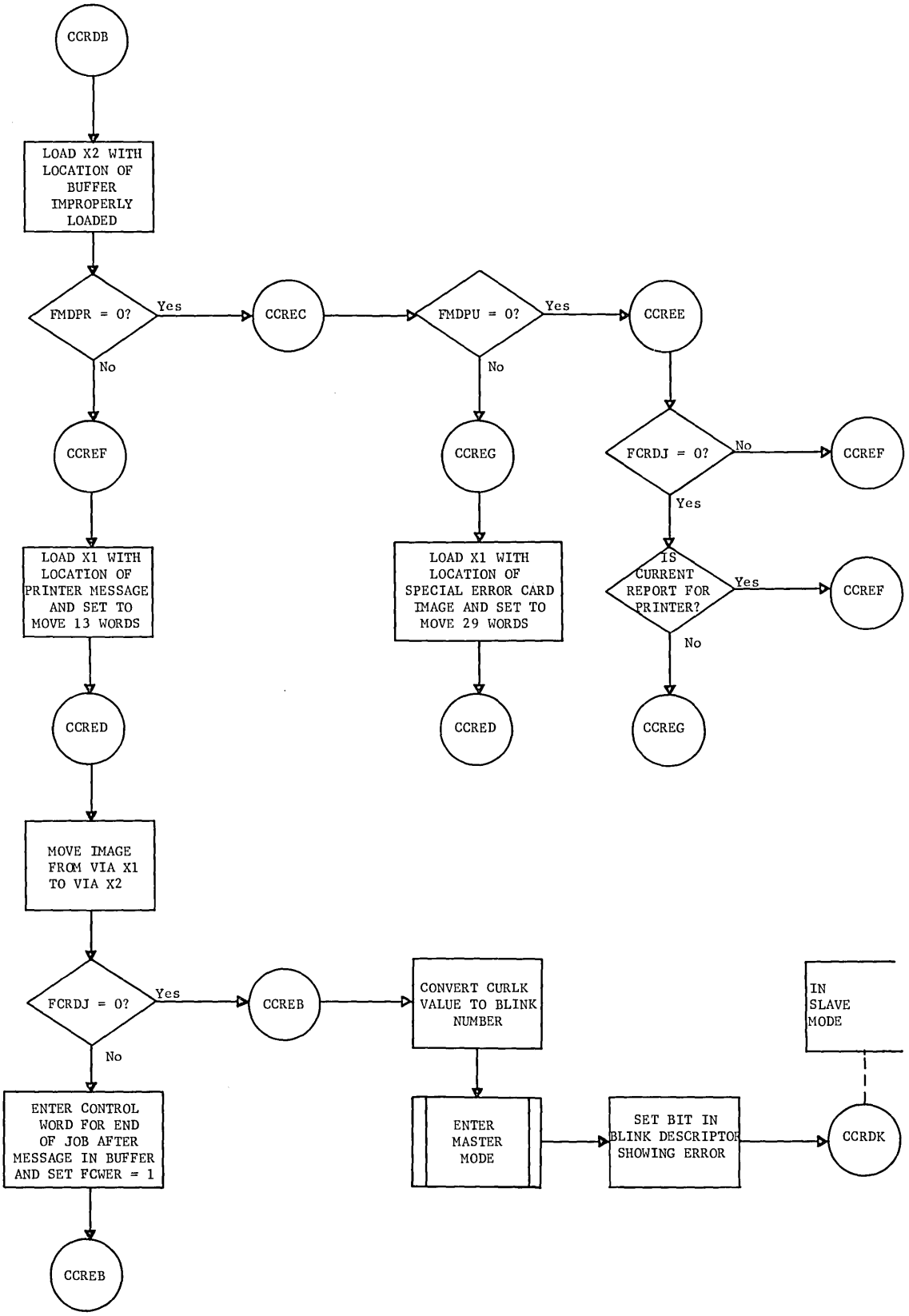

WRITE
.MGEOT
RJAF
IS REPORT VIA X5 ONE WE WANT?
No
RJAE
ADVANCE X5 AND X1 FOR ENTRY
X1 = REPORT COUNT?
No
RJAF
Yes
MOVE FIRST DESCRIPTOR WORD TO JLIST VIA X0
SAVE X0 AS NEW JLPTR
FCRPC = 0?
No
RJAK
MORE ROOM IN JLIST?
No
RJAL
Yes
Yes
RJAD
SEARCH $REPORT RECORD IN BUFAM FOR REPORT MATCHING THIS ONE
RJAG
PUT ZERO WORD FOLLOWED BY CURLK AND CURBK IN JLIST AND UPDATE JLPTR
MOVE SECOND WORD TO JLIST AND ADD 2 TO X0 FOR NEXT ENTRY
FIND ONE?
No
RJAD
SET FMJLF = 1
Yes
RJAE
RJAI
RJAM
ADD 7 TO UPPER OF SECOND DESCRIPTOR WORD, MOVE IT TO JLIST VIA X0, MOVE THREE $REPORT WORDS TO JLIST VIA X0
ADVANCE X0 TO NEXT POSITION
RJAE

CCCRD
.MGEOT

COURTESY CALL FOR READ

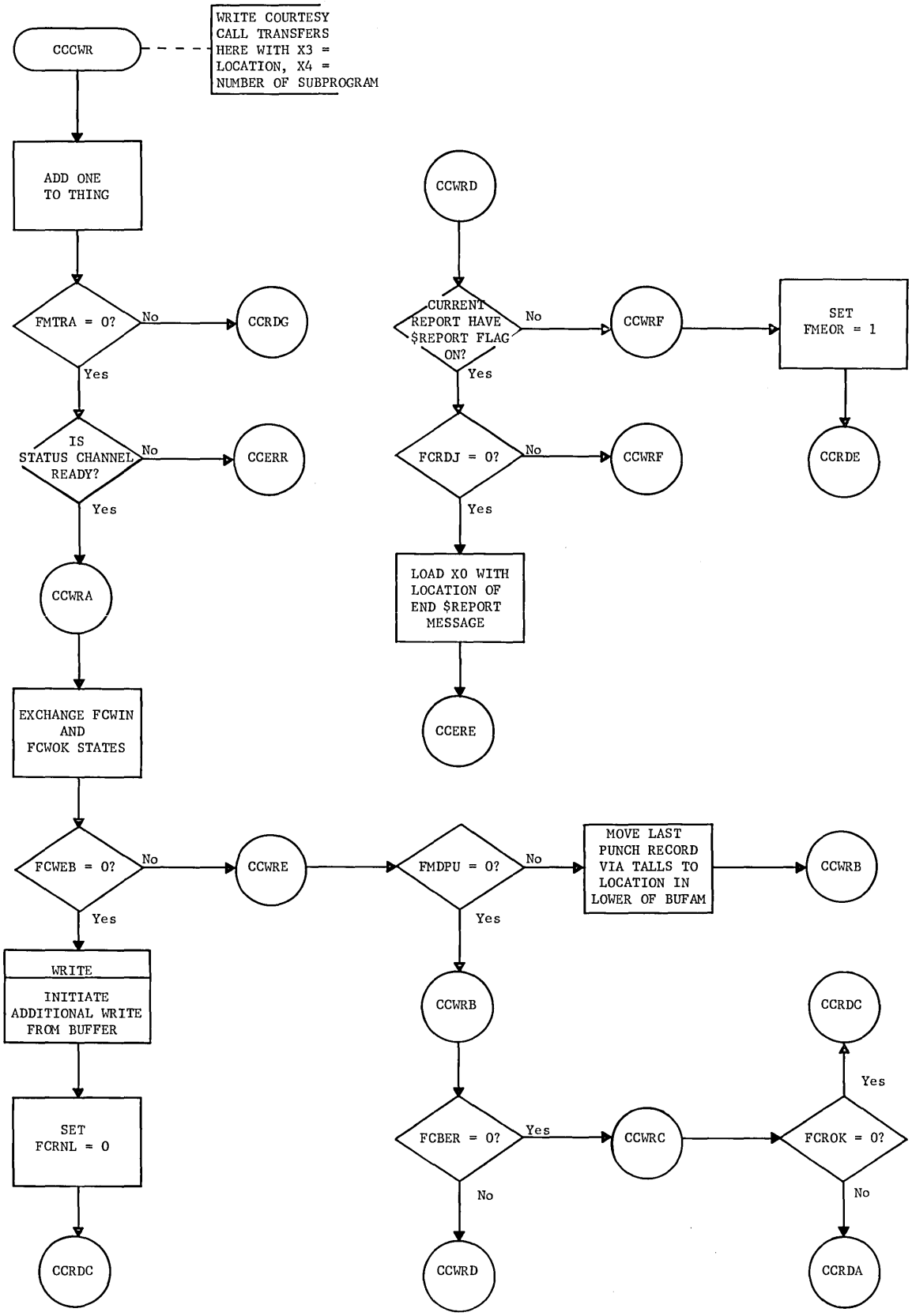



CCCRD
.MGEOT
CCRDJ
WRITE
INITIATE WRITE FROM RECORD JUST READ
FCBLL = 1 OR FCBER = 1?
Yes
CCRDC
No
FCRNL = 0?
Yes
CCRDF
READ
INITIATE READ INTO OTHER BUFFER
CCRDC
No
EXCHANGE FCRNL STATE
CCRDC
LTHING +10 < THING?
Yes
CCRDE
MAINL = 0?
No
CCRDG
Yes
No
CCRDG
ENTER MASTER MODE
END COURTESY CALL
MAINL = 0?
Yes
CHANGE ROADBLOCK TO RELINQUISH IN STATE SAVED BY COURTESY CALL AT 47
CCRDD
No
CCRDD
END COURTESY CALL

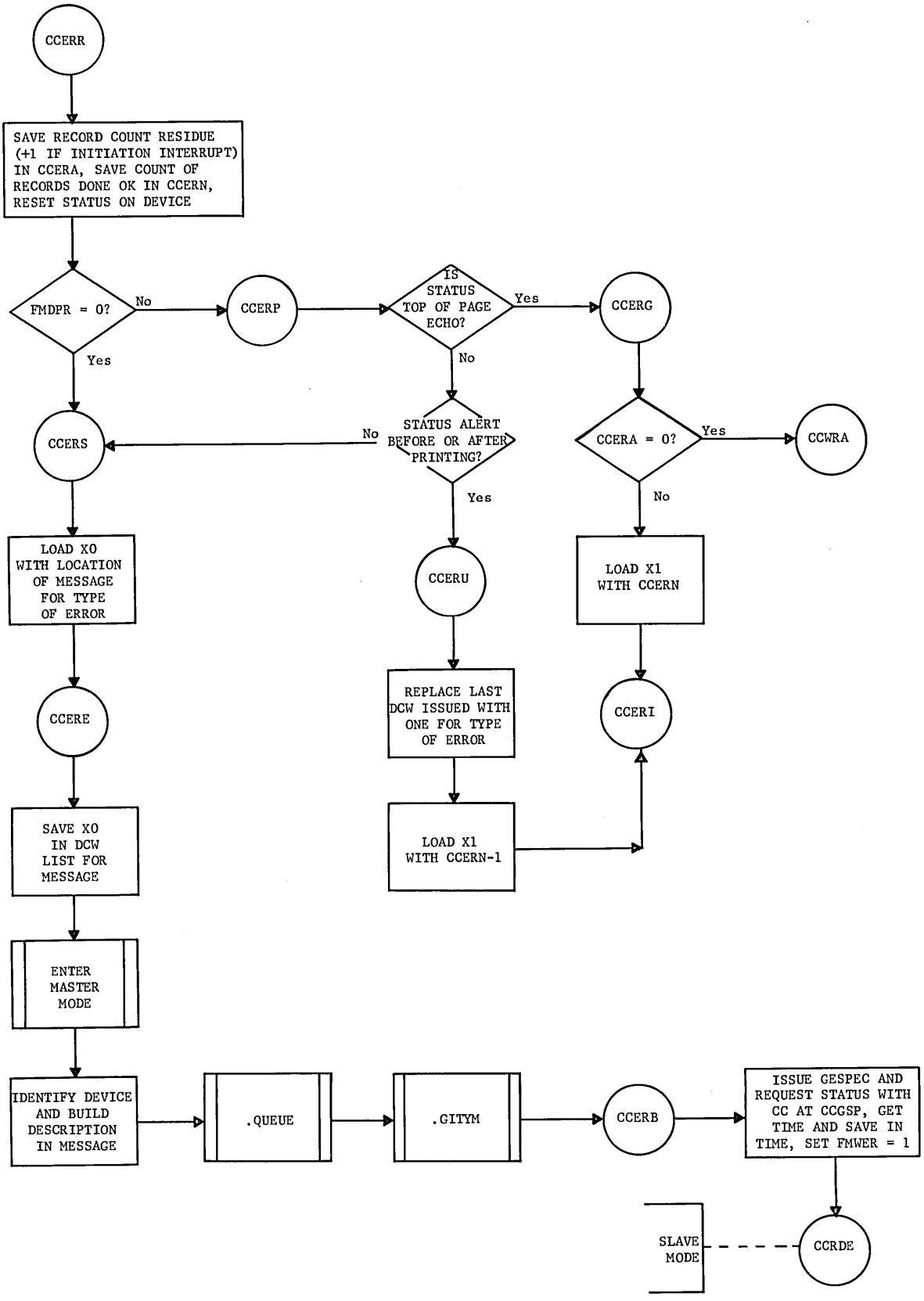

CCRDB
LOAD X2 WITH LOCATION OF BUFFER IMPROPERLY LOADED
FMDPR = 0?
Yes
No
CCREC
FMDPU = 0?
Yes
No
CCREE
CCREF
CCREG
FCRDJ = 0?
No
Yes
CCREF
LOAD X1 WITH LOCATION OF PRINTER MESSAGE AND SET TO MOVE 13 WORDS
LOAD X1 WITH LOCATION OF SPECIAL ERROR CARD IMAGE AND SET TO MOVE 29 WORDS
IS CURRENT REPORT FOR PRINTER?
Yes
No
CCREF
CCRED
CCRED
CCREG
MOVE IMAGE FROM VIA X1 TO VIA X2
FCRDJ = 0?
Yes
No
CCREB
CONVERT CURLK VALUE TO BLINK NUMBER
IN SLAVE MODE
ENTER CONTROL WORD FOR END OF JOB AFTER MESSAGE IN BUFFER AND SET FCWER = 1
ENTER MASTER MODE
SET BIT IN BLINK DESCRIPTOR SHOWING ERROR
CCRDK
CCREB

CCCWR
.MGEOT

COURTESY CALL FOR WRITE

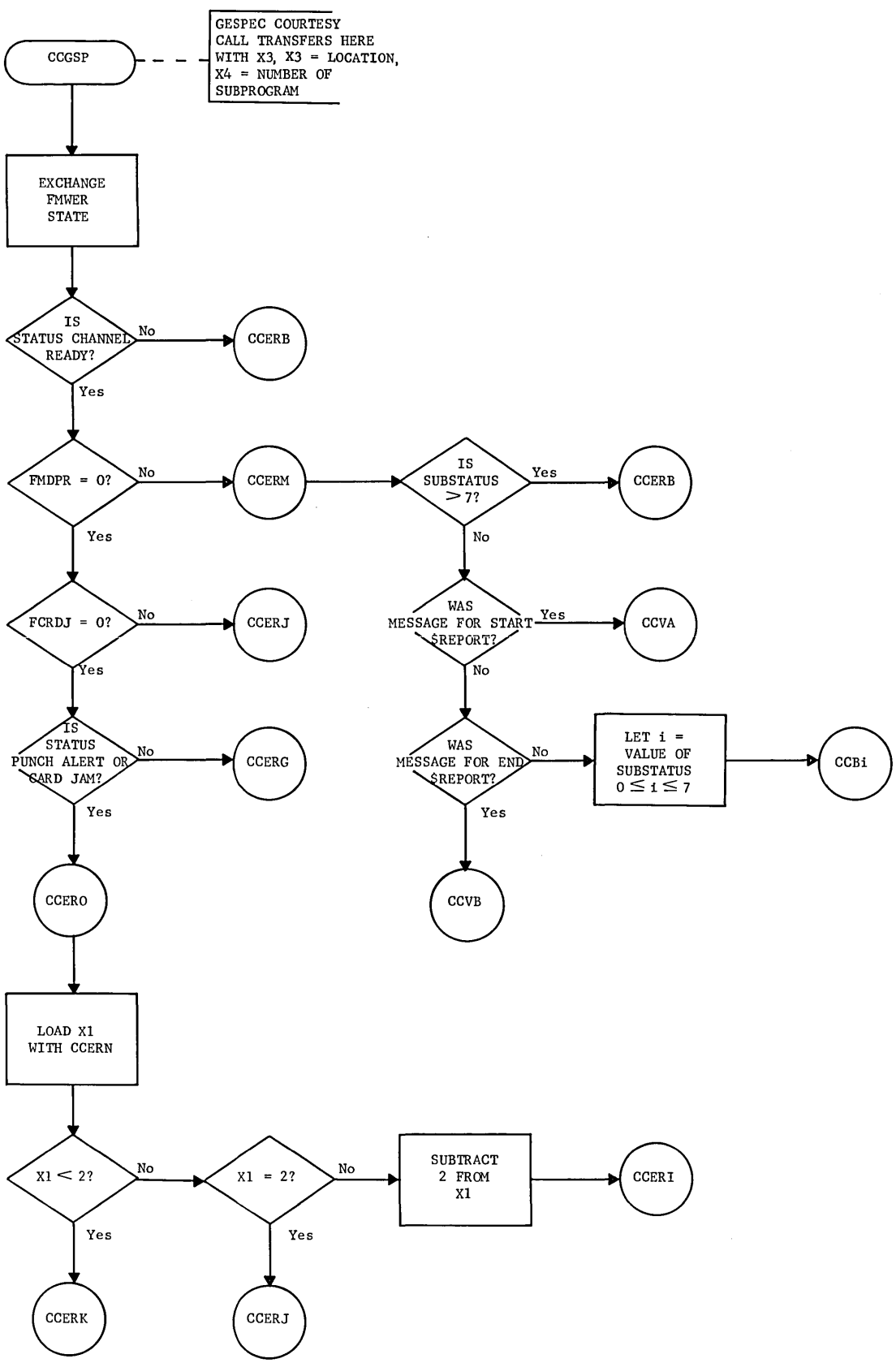


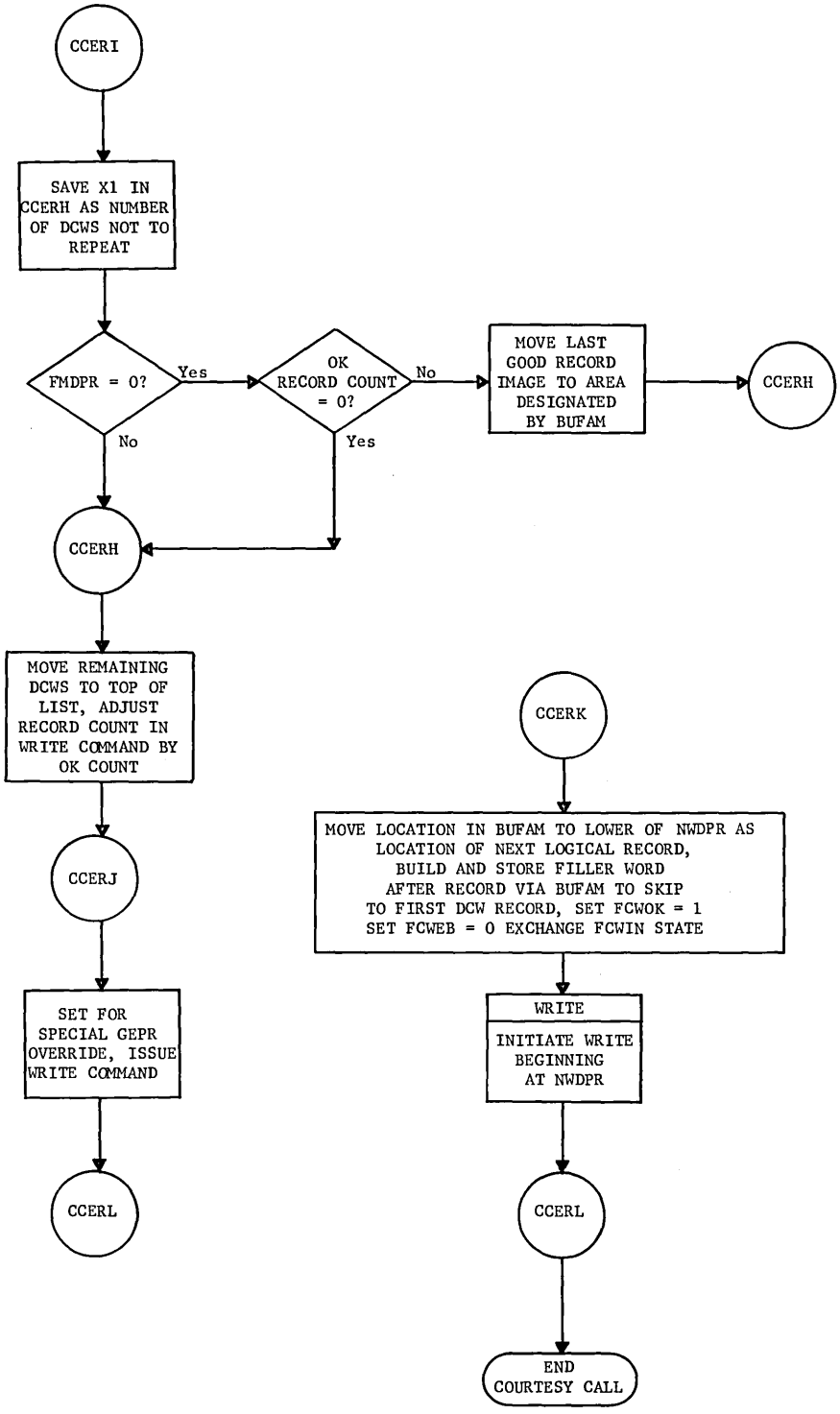
CCCWR
.MGEOT

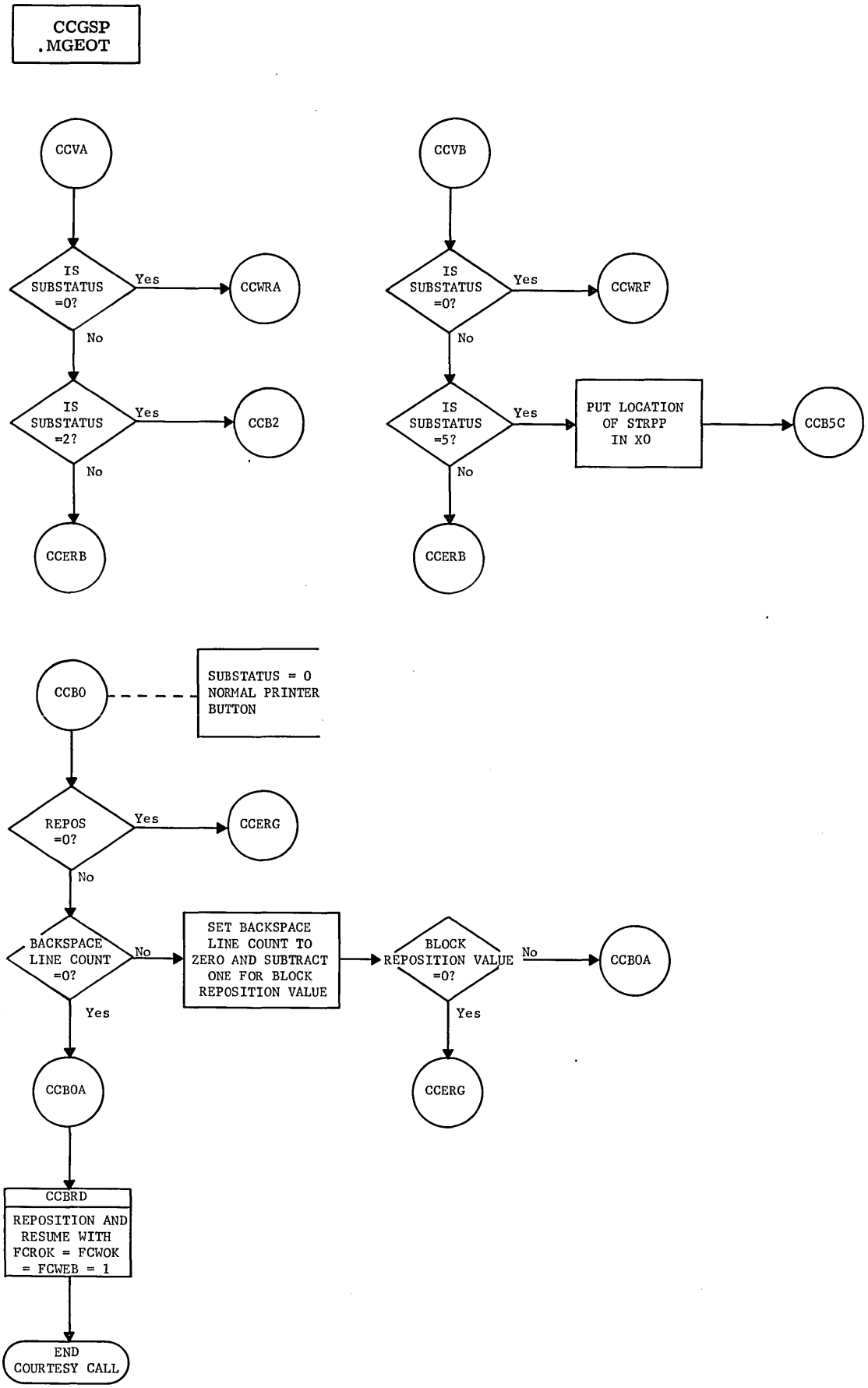

CCERR
SAVE RECORD COUNT RESIDUE (+1 IF INITIATION INTERRUPT) IN CCERA, SAVE COUNT OF RECORDS DONE OK IN CCERN, RESET STATUS ON DEVICE
FMDPR = 0?
No
Yes
CCERP
IS STATUS TOP OF PAGE ECHO?
Yes
No
CCERG
CCERS
No
STATUS ALERT BEFORE OR AFTER PRINTING?
Yes
CCERA = 0?
Yes
No
CCWRA
LOAD X0 WITH LOCATION OF MESSAGE FOR TYPE OF ERROR
CCERU
LOAD X1 WITH CCERN
CCERE
REPLACE LAST DCW ISSUED WITH ONE FOR TYPE OF ERROR
CCERI
SAVE X0 IN DCW LIST FOR MESSAGE
LOAD X1 WITH CCERN-1
ENTER MASTER MODE
IDENTIFY DEVICE AND BUILD DESCRIPTION IN MESSAGE
.QUEUE
.GITYM
CCERB
ISSUE GESPEC AND REQUEST STATUS WITH CC AT CCGSP, GET TIME AND SAVE IN TIME, SET FMWER = 1
SLAVE MODE
CCRDE

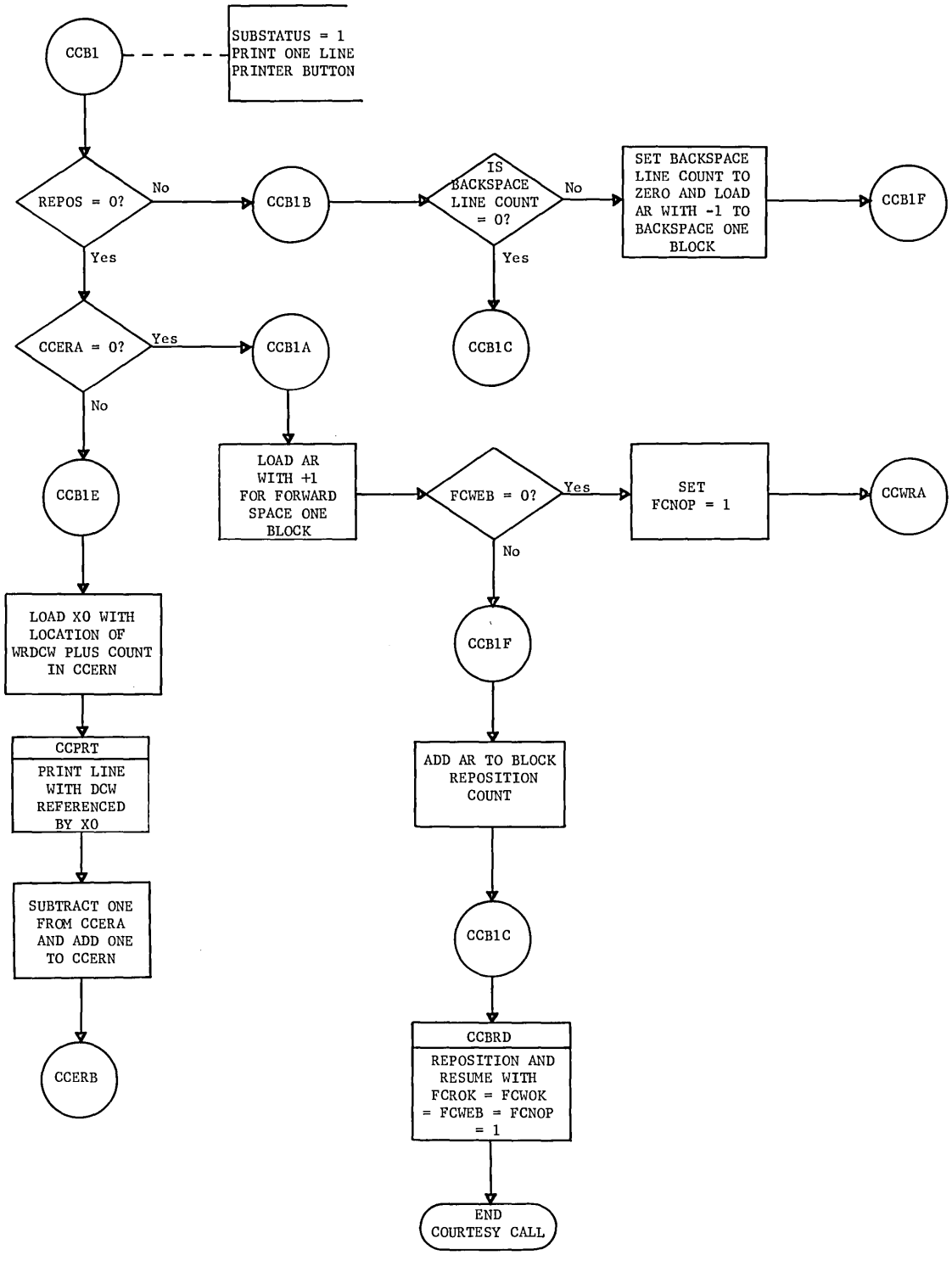
CCGSP
.MGEOT

COURTESY CALL FOR GESPEC

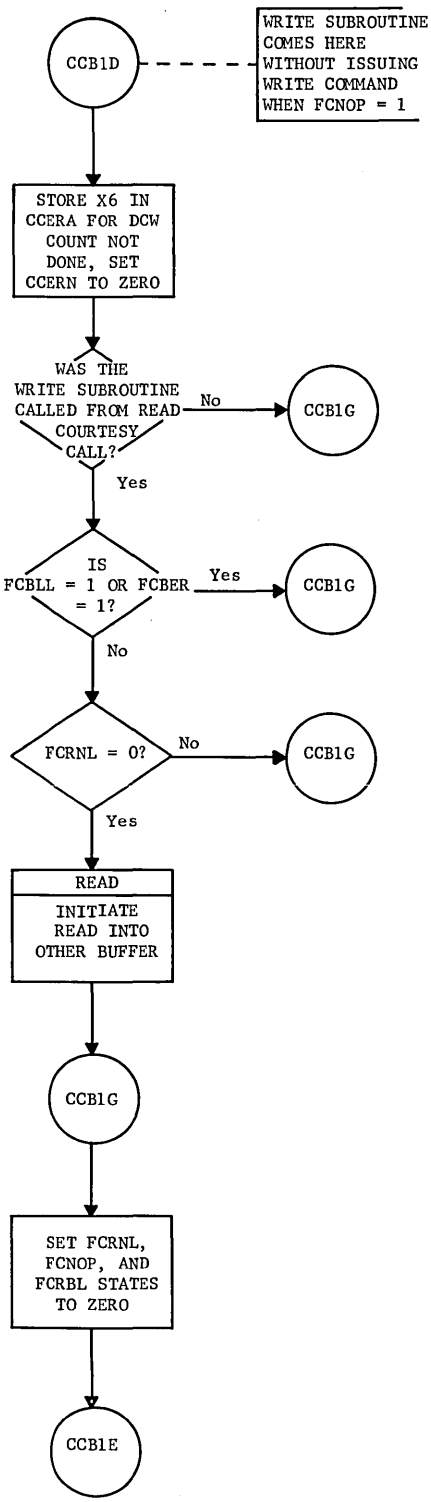


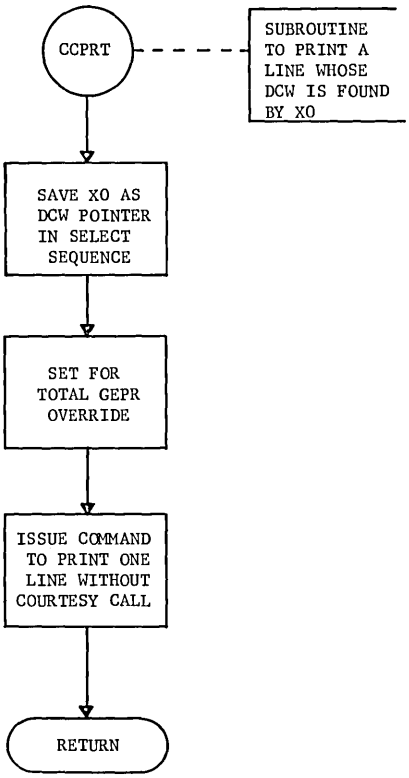

CCERI
SAVE X1 IN CCERH AS NUMBER OF DCWS NOT TO REPEAT
FMDPR = 0?
Yes
OK RECORD COUNT = 0?
No
MOVE LAST GOOD RECORD IMAGE TO AREA DESIGNATED BY BUFAM
CCERH
No
Yes
CCERH
MOVE REMAINING DCWS TO TOP OF LIST, ADJUST RECORD COUNT IN WRITE COMMAND BY OK COUNT
CCERJ
SET FOR SPECIAL GEPR OVERRIDE, ISSUE WRITE COMMAND
CCERL
CCERK
MOVE LOCATION IN BUFAM TO LOWER OF NWDPR AS LOCATION OF NEXT LOGICAL RECORD, BUILD AND STORE FILLER WORD AFTER RECORD VIA BUFAM TO SKIP TO FIRST DCW RECORD, SET FCWOK = 1 SET FCWEB = 0 EXCHANGE FCWIN STATE
WRITE
INITIATE WRITE BEGINNING AT NWDPR
CCERL
END COURTESY CALL

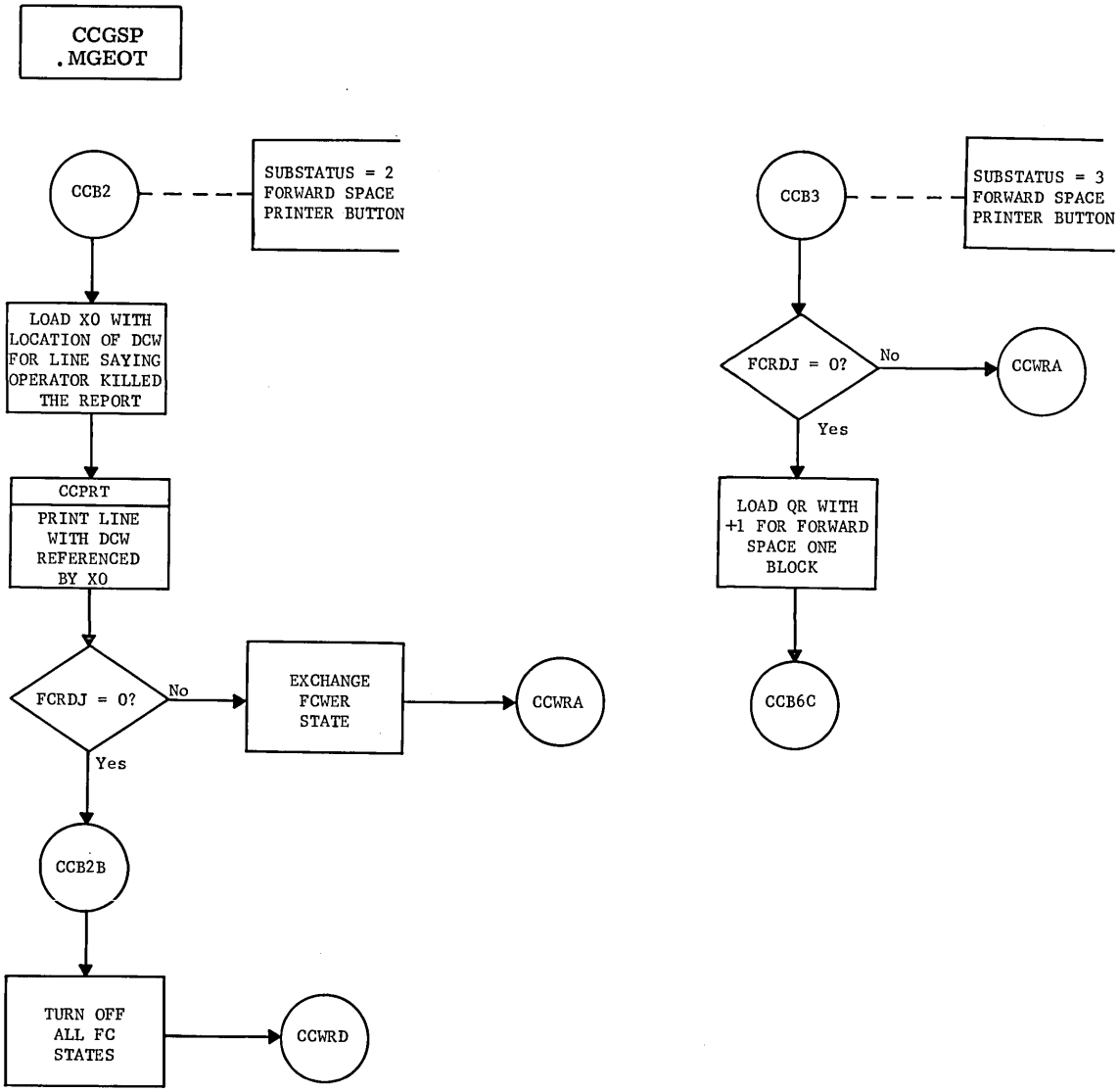

CCGSP
.MGEOT
CCVA
IS SUBSTATUS =0?
Yes
CCWRA
No
IS SUBSTATUS =2?
Yes
CCB2
No
CCERB
CCVB
IS SUBSTATUS =0?
Yes
CCWRF
No
IS SUBSTATUS =5?
Yes
PUT LOCATION OF STRPP IN XO
CCB5C
No
CCERB
CCBO
SUBSTATUS = 0 NORMAL PRINTER BUTTON
REPOS =0?
Yes
CCERG
No
BACKSPACE LINE COUNT =0?
No
SET BACKSPACE LINE COUNT TO ZERO AND SUBTRACT ONE FOR BLOCK REPOSITION VALUE
BLOCK REPOSITION VALUE =0?
No
CCBOA
Yes
Yes
CCBOA
CCERG
CCBRD
REPOSITION AND RESUME WITH FCROK = FCWOK = FCWEB = 1
END COURTESY CALL

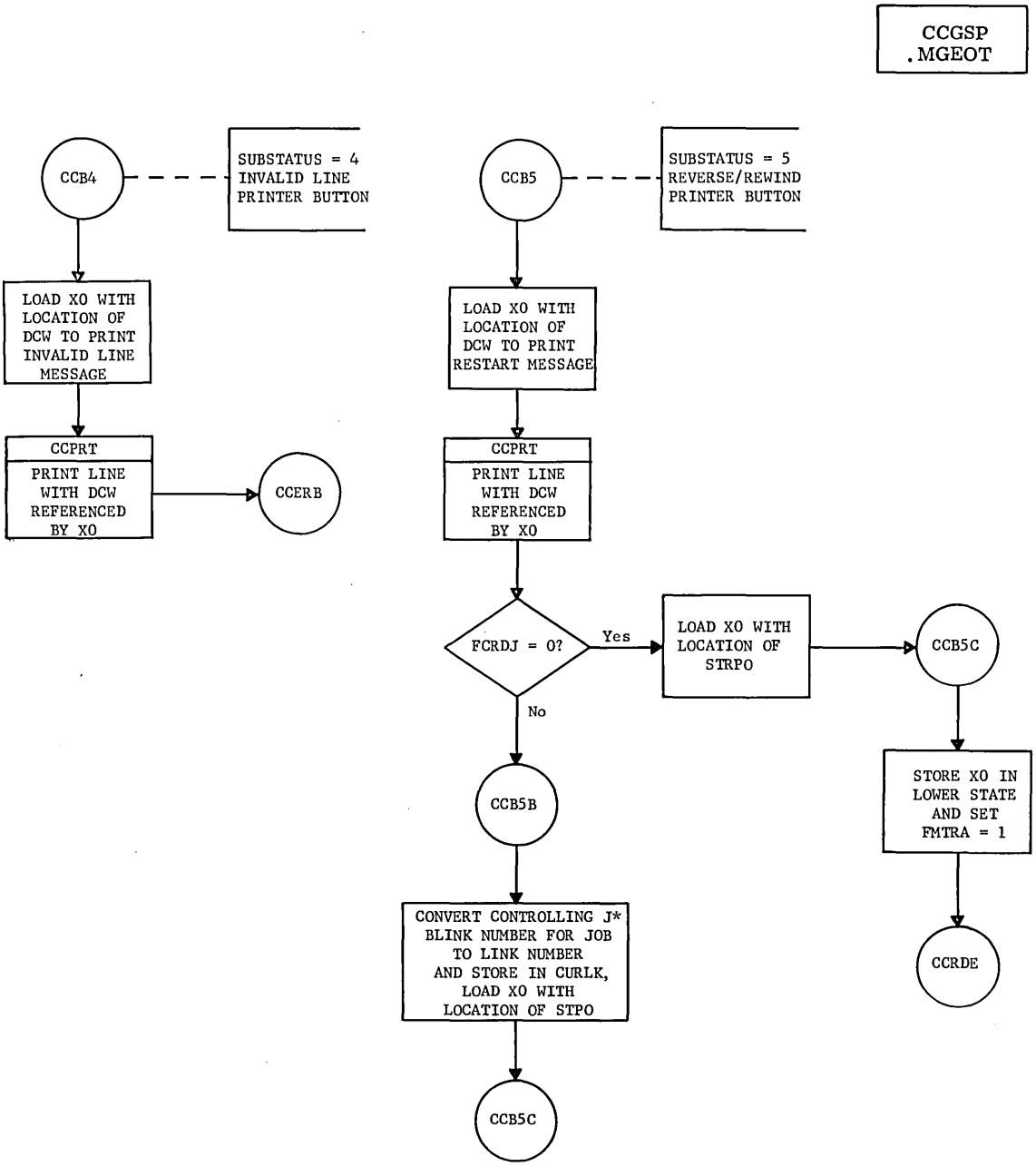

CCB1
SUBSTATUS = 1
PRINT ONE LINE
PRINTER BUTTON
REPOS = 0?
No
CCB1B
IS BACKSPACE LINE COUNT = 0?
No
SET BACKSPACE LINE COUNT TO ZERO AND LOAD AR WITH -1 TO BACKSPACE ONE BLOCK
CCB1F
Yes
Yes
CCERA = 0?
Yes
CCB1A
CCB1C
No
CCB1E
LOAD AR WITH +1 FOR FORWARD SPACE ONE BLOCK
FCWEB = 0?
Yes
SET FCNOP = 1
CCWRA
No
LOAD X0 WITH LOCATION OF WRDCW PLUS COUNT IN CCERN
CCB1F
CCPRT
PRINT LINE WITH DCW REFERENCED BY X0
ADD AR TO BLOCK REPOSITION COUNT
SUBTRACT ONE FROM CCERA AND ADD ONE TO CCERN
CCB1C
CCERB
CCBRD
REPOSITION AND RESUME WITH FCROK = FCWOK = FCWEB = FCNOP = 1
END
COURTESY CALL

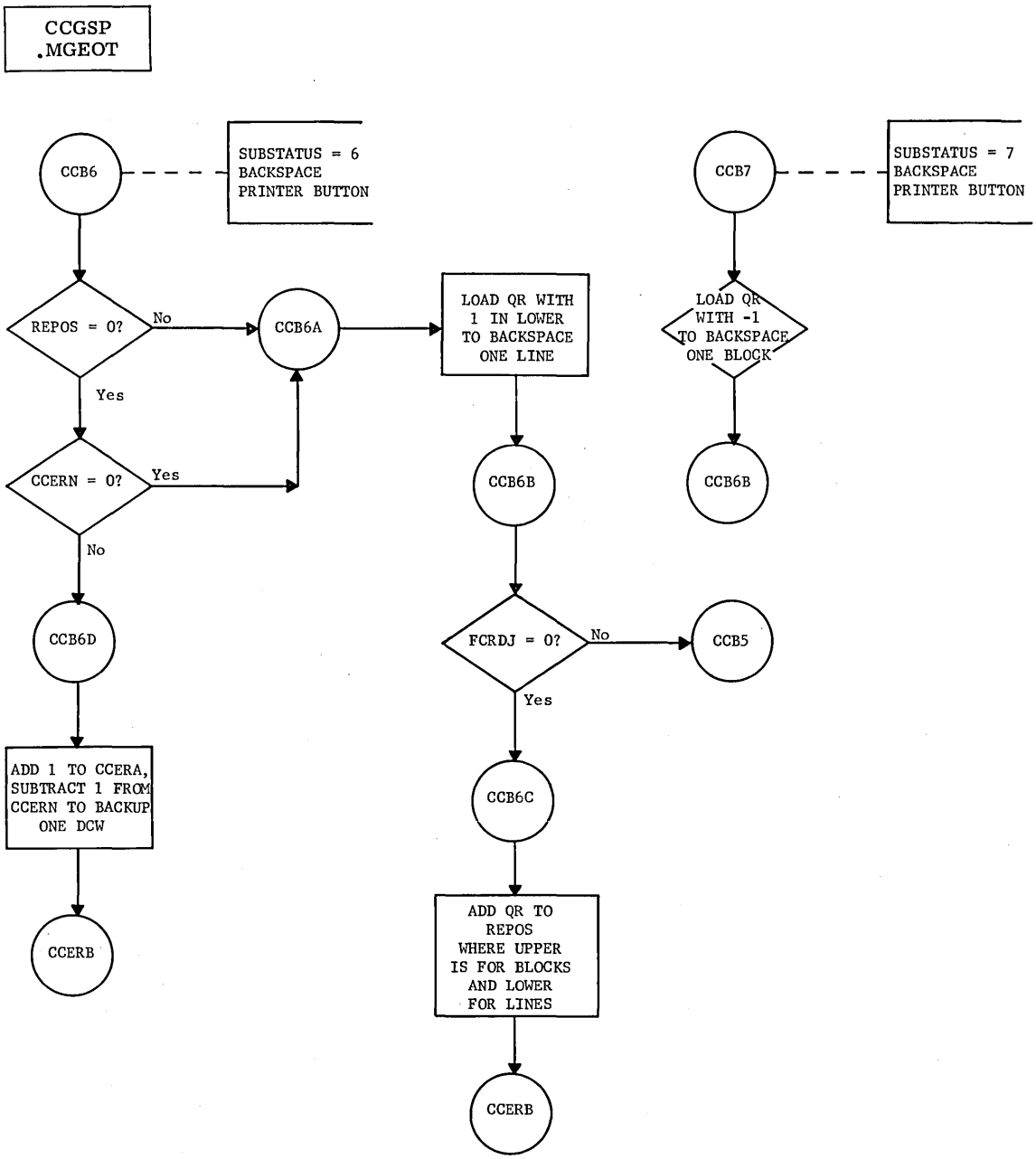
CCGSP
.MGEOT

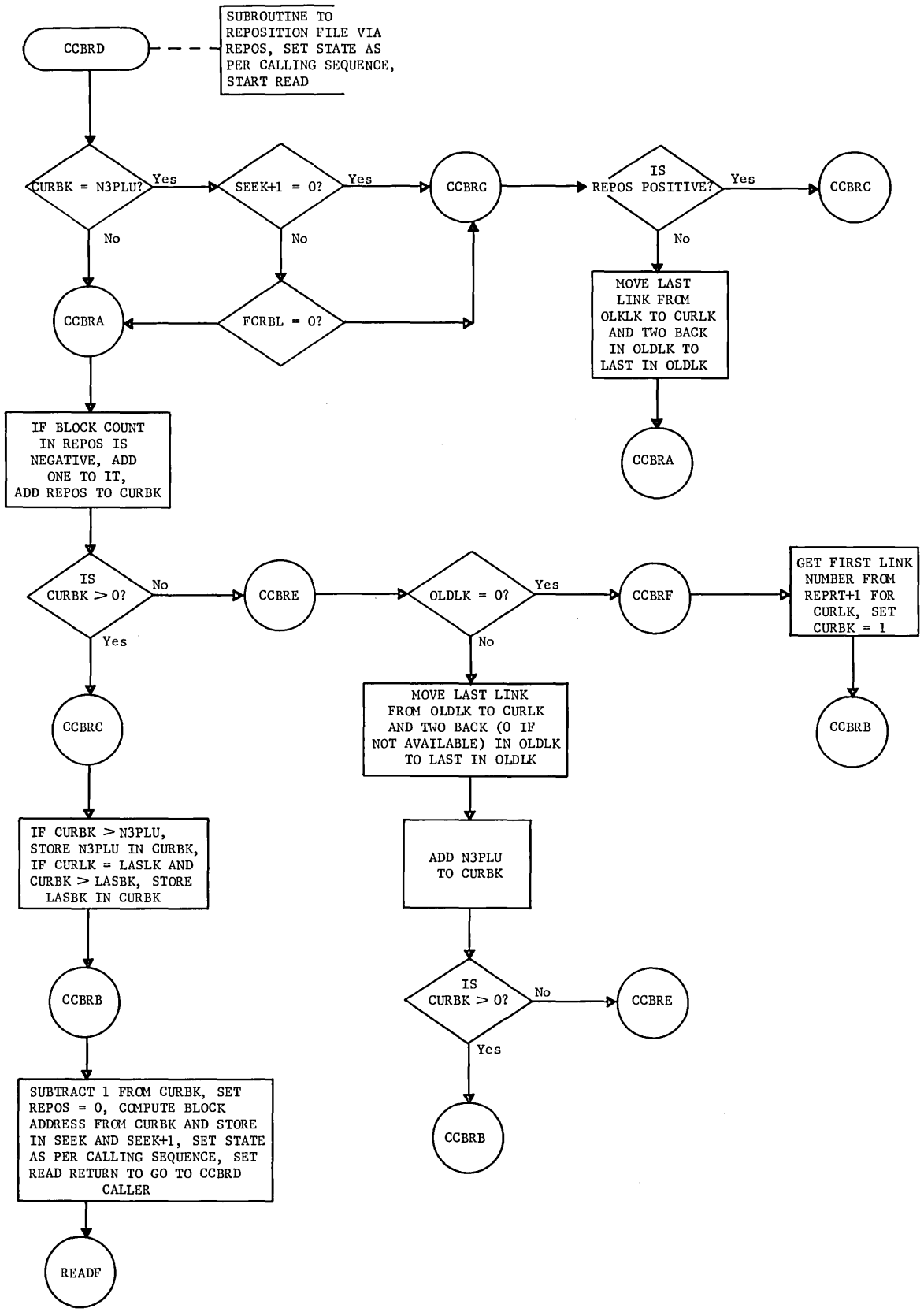

CCB1D
WRITE SUBROUTINE COMES HERE WITHOUT ISSUING WRITE COMMAND WHEN FCNOP = 1
STORE X6 IN CCERA FOR DCW COUNT NOT DONE, SET CCERN TO ZERO
WAS THE WRITE SUBROUTINE CALLED FROM READ COURTESY CALL?
No
CCB1G
Yes
IS FCBLL = 1 OR FCBER = 1?
Yes
CCB1G
No
FCRNL = 0?
No
CCB1G
Yes
READ
INITIATE READ INTO OTHER BUFFER
CCB1G
SET FCRNL, FCNOP, AND FCRBL STATES TO ZERO
CCB1E


CCPRT
SUBROUTINE TO PRINT A LINE WHOSE DCW IS FOUND BY XO
SAVE XO AS DCW POINTER IN SELECT SEQUENCE
SET FOR TOTAL GEPR OVERRIDE
ISSUE COMMAND TO PRINT ONE LINE WITHOUT COURTESY CALL
RETURN


CCGSP
.MGEOT
CCB2
SUBSTATUS = 2
FORWARD SPACE
PRINTER BUTTON
LOAD XO WITH
LOCATION OF DCW
FOR LINE SAYING
OPERATOR KILLED
THE REPORT
CCPRT
PRINT LINE
WITH DCW
REFERENCED
BY XO
FCRDJ = 0?
No
EXCHANGE
FCWER
STATE
CCWRA
Yes
CCB2B
TURN OFF
ALL FC
STATES
CCWRD
CCB3
SUBSTATUS = 3
FORWARD SPACE
PRINTER BUTTON
FCRDJ = 0?
No
CCWRA
Yes
LOAD QR WITH
+1 FOR FORWARD
SPACE ONE
BLOCK
CCB6C

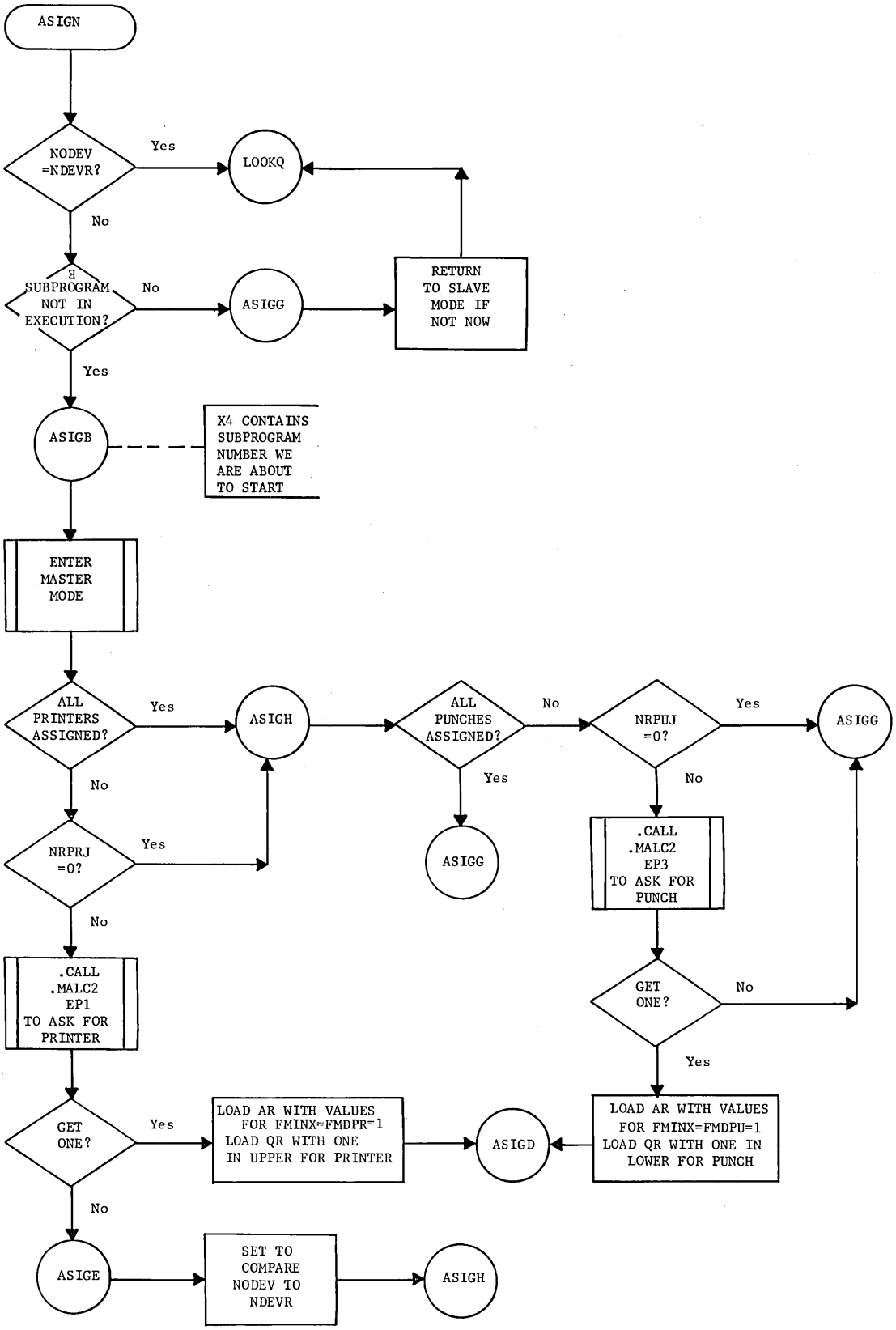

CCGSP
.MGEOT
CCB4
SUBSTATUS = 4
INVALID LINE
PRINTER BUTTON
LOAD X0 WITH
LOCATION OF
DCW TO PRINT
INVALID LINE
MESSAGE
CCPRT
PRINT LINE
WITH DCW
REFERENCED
BY X0
CCERB
CCB5
SUBSTATUS = 5
REVERSE/REWIND
PRINTER BUTTON
LOAD X0 WITH
LOCATION OF
DCW TO PRINT
RESTART MESSAGE
CCPRT
PRINT LINE
WITH DCW
REFERENCED
BY X0
FCRDJ = 0?
Yes
No
LOAD X0 WITH
LOCATION OF
STRPO
CCB5C
STORE X0 IN
LOWER STATE
AND SET
FMTRA = 1
CCRDE
CCB5B
CONVERT CONTROLLING J*
BLINK NUMBER FOR JOB
TO LINK NUMBER
AND STORE IN CURLK,
LOAD X0 WITH
LOCATION OF STPO
CCB5C

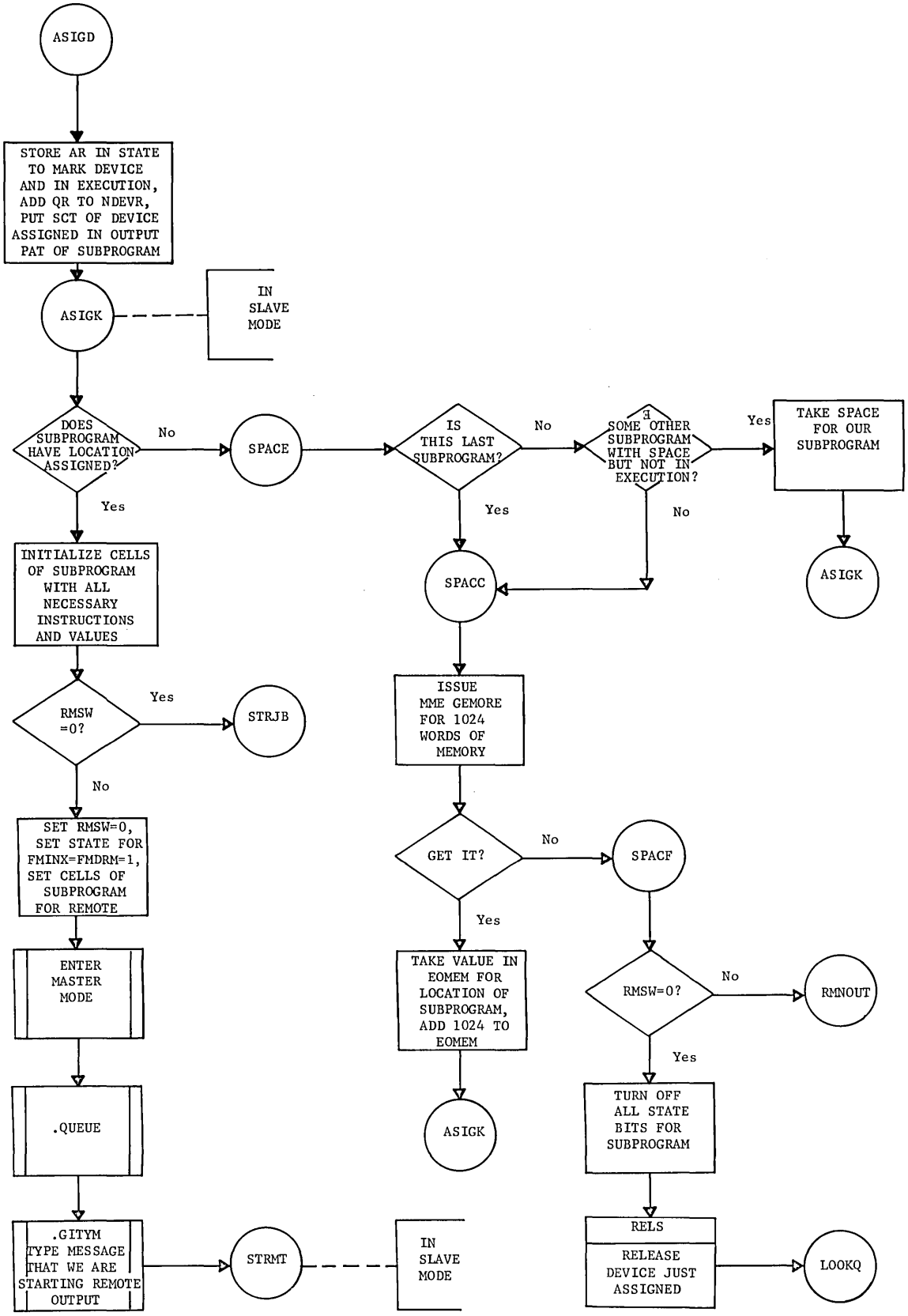

CCGSP
.MGEOT
CCB6
SUBSTATUS = 6
BACKSPACE
PRINTER BUTTON
CCB7
SUBSTATUS = 7
BACKSPACE
PRINTER BUTTON
REPOS = 0?
No
CCB6A
LOAD QR WITH
1 IN LOWER
TO BACKSPACE
ONE LINE
LOAD QR
WITH -1
TO BACKSPACE
ONE BLOCK
Yes
CCERN = 0?
Yes
CCB6B
CCB6B
No
CCB6D
FCRDJ = 0?
No
CCB5
Yes
ADD 1 TO CCERA,
SUBTRACT 1 FROM
CCERN TO BACKUP
ONE DCW
CCB6C
ADD QR TO
REPOS
WHERE UPPER
IS FOR BLOCKS
AND LOWER
FOR LINES
CCERB
CCERB


CCBRD
SUBROUTINE TO REPOSITION FILE VIA REPOS, SET STATE AS PER CALLING SEQUENCE, START READ
CURBK = N3PLU?
Yes
No
SEEK+1 = 0?
Yes
No
CCBRG
IS REPOS POSITIVE?
Yes
No
CCBRC
MOVE LAST LINK FROM OLKLK TO CURLK AND TWO BACK IN OLDLK TO LAST IN OLDLK
CCBRA
FCRBL = 0?
CCBRA
IF BLOCK COUNT IN REPOS IS NEGATIVE, ADD ONE TO IT, ADD REPOS TO CURBK
IS CURBK > 0?
No
Yes
CCBRE
OLDLK = 0?
Yes
No
CCBRF
GET FIRST LINK NUMBER FROM REPRT+1 FOR CURLK, SET CURBK = 1
CCBRB
CCBRC
MOVE LAST LINK FROM OLDLK TO CURLK AND TWO BACK (0 IF NOT AVAILABLE) IN OLDLK TO LAST IN OLDLK
IF CURBK > N3PLU, STORE N3PLU IN CURBK, IF CURLK = LASLK AND CURBK > LASBK, STORE LASBK IN CURBK
ADD N3PLU TO CURBK
CCBRB
IS CURBK > 0?
No
Yes
CCBRE
SUBTRACT 1 FROM CURBK, SET REPOS = 0, COMPUTE BLOCK ADDRESS FROM CURBK AND STORE IN SEEK AND SEEK+1, SET STATE AS PER CALLING SEQUENCE, SET READ RETURN TO GO TO CCBRD CALLER
CCBRB
READF

ASIGN
.MGEOT

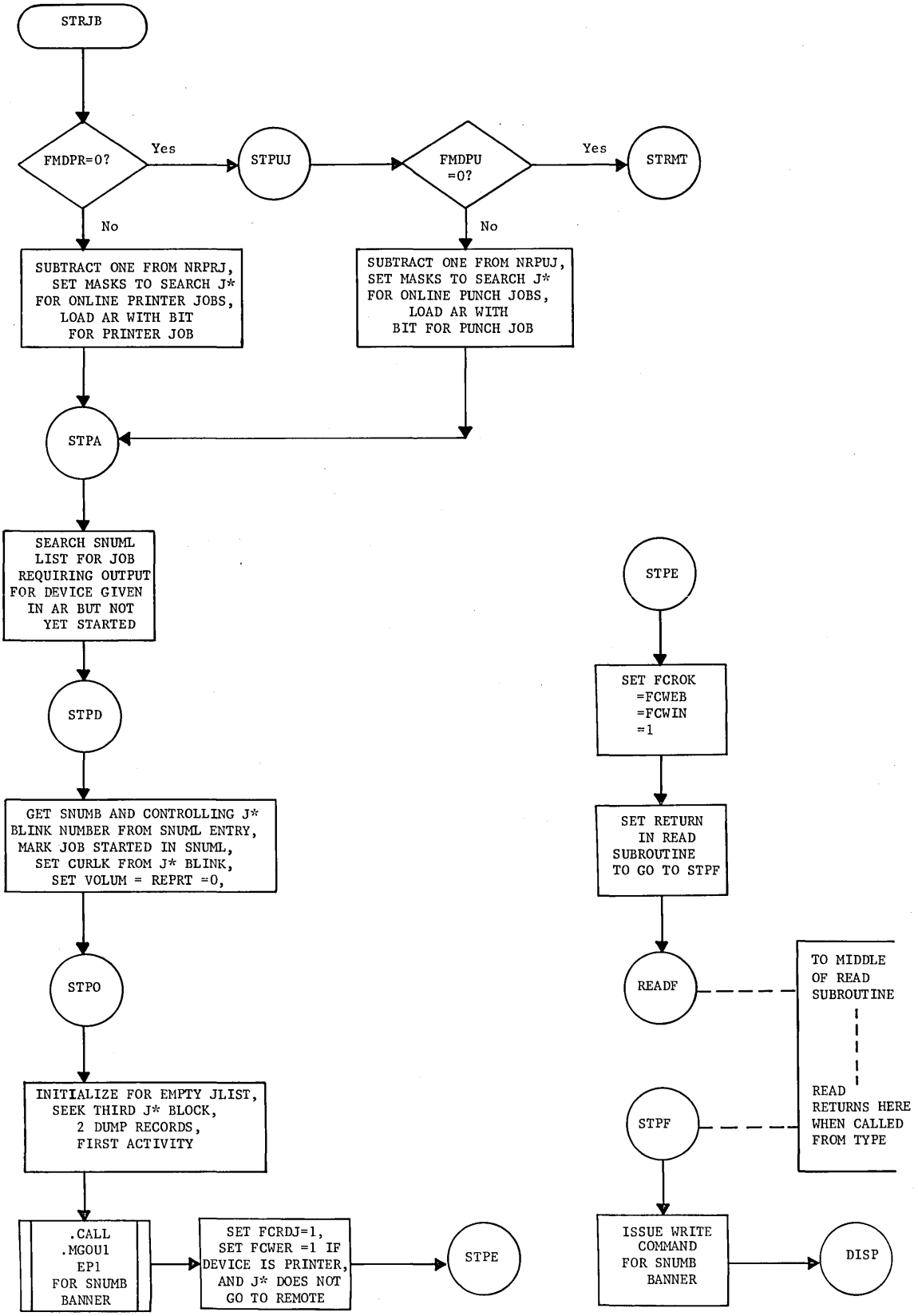
ASSIGN DEVICE AND JOB




ASIGD
STORE AR IN STATE TO MARK DEVICE AND IN EXECUTION, ADD QR TO NDEVR, PUT SCT OF DEVICE ASSIGNED IN OUTPUT PAT OF SUBPROGRAM
ASIGK
IN SLAVE MODE
DOES SUBPROGRAM HAVE LOCATION ASSIGNED?
No
SPACE
IS THIS LAST SUBPROGRAM?
No
SOME OTHER SUBPROGRAM WITH SPACE BUT NOT IN EXECUTION?
Yes
TAKE SPACE FOR OUR SUBPROGRAM
ASIGK
Yes
No
Yes
INITIALIZE CELLS OF SUBPROGRAM WITH ALL NECESSARY INSTRUCTIONS AND VALUES
SPACC
RMSW =0?
Yes
STRJB
ISSUE MME GEMORE FOR 1024 WORDS OF MEMORY
No
SET RMSW=0, SET STATE FOR FMINX=FMDRM=1, SET CELLS OF SUBPROGRAM FOR REMOTE
GET IT?
No
SPACF
Yes
ENTER MASTER MODE
TAKE VALUE IN EOMEM FOR LOCATION OF SUBPROGRAM, ADD 1024 TO EOMEM
RMSW=0?
No
RMNOUT
Yes
.QUEUE
ASIGK
TURN OFF ALL STATE BITS FOR SUBPROGRAM
.GITYM TYPE MESSAGE THAT WE ARE STARTING REMOTE OUTPUT
STRMT
IN SLAVE MODE
RELS
RELEASE DEVICE JUST ASSIGNED
LOOKQ

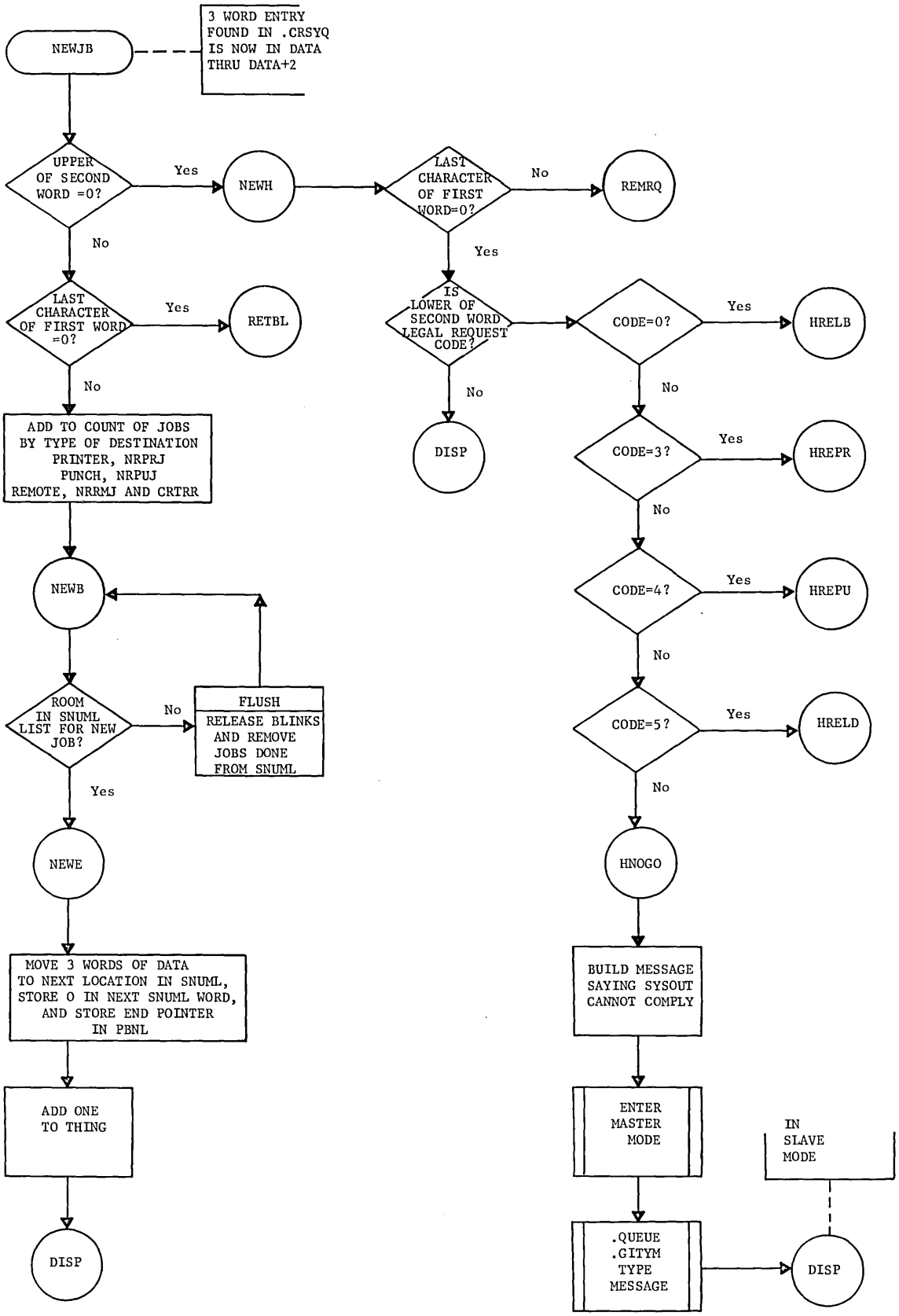
STRJB
.MGEOT

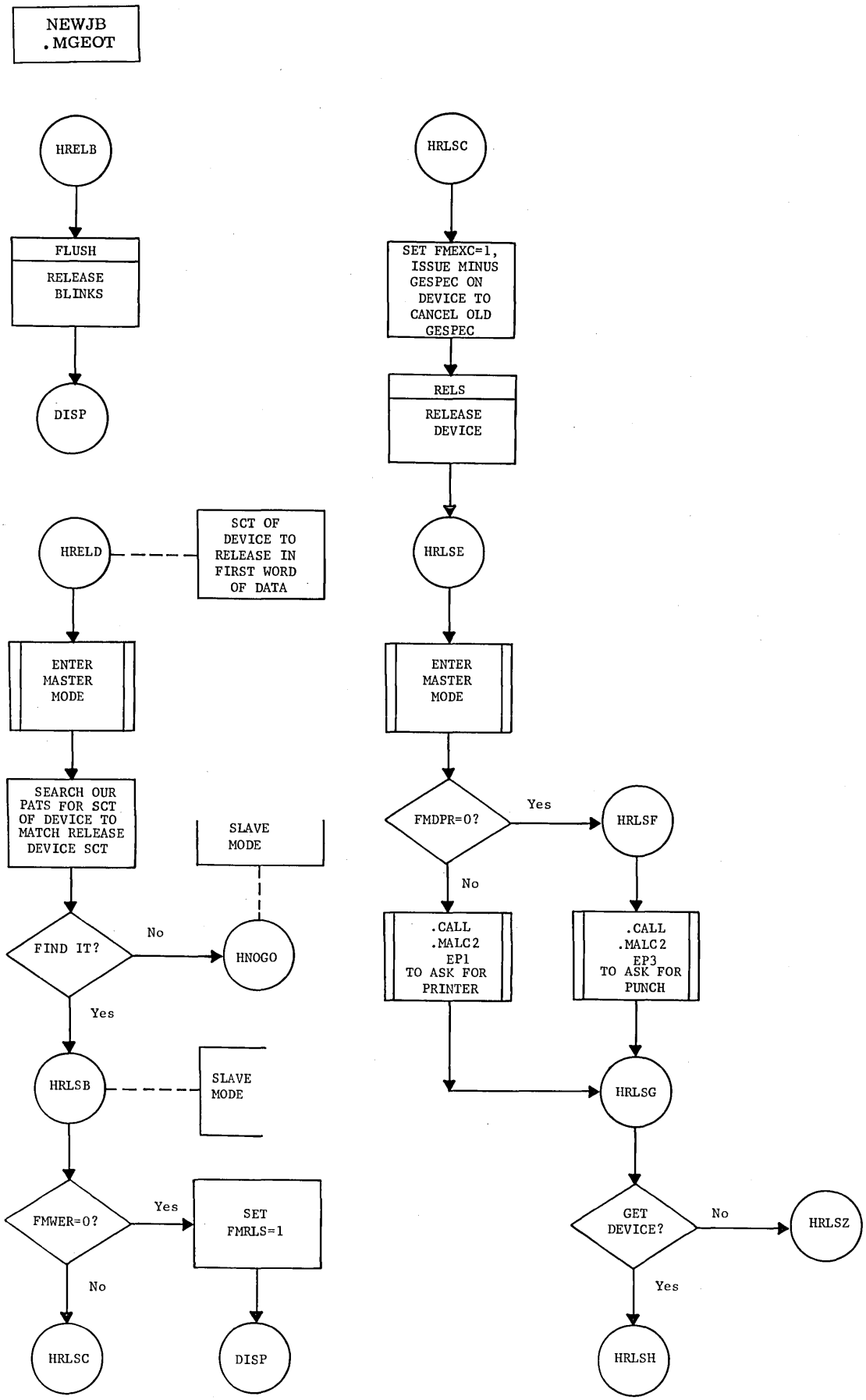
START JOB

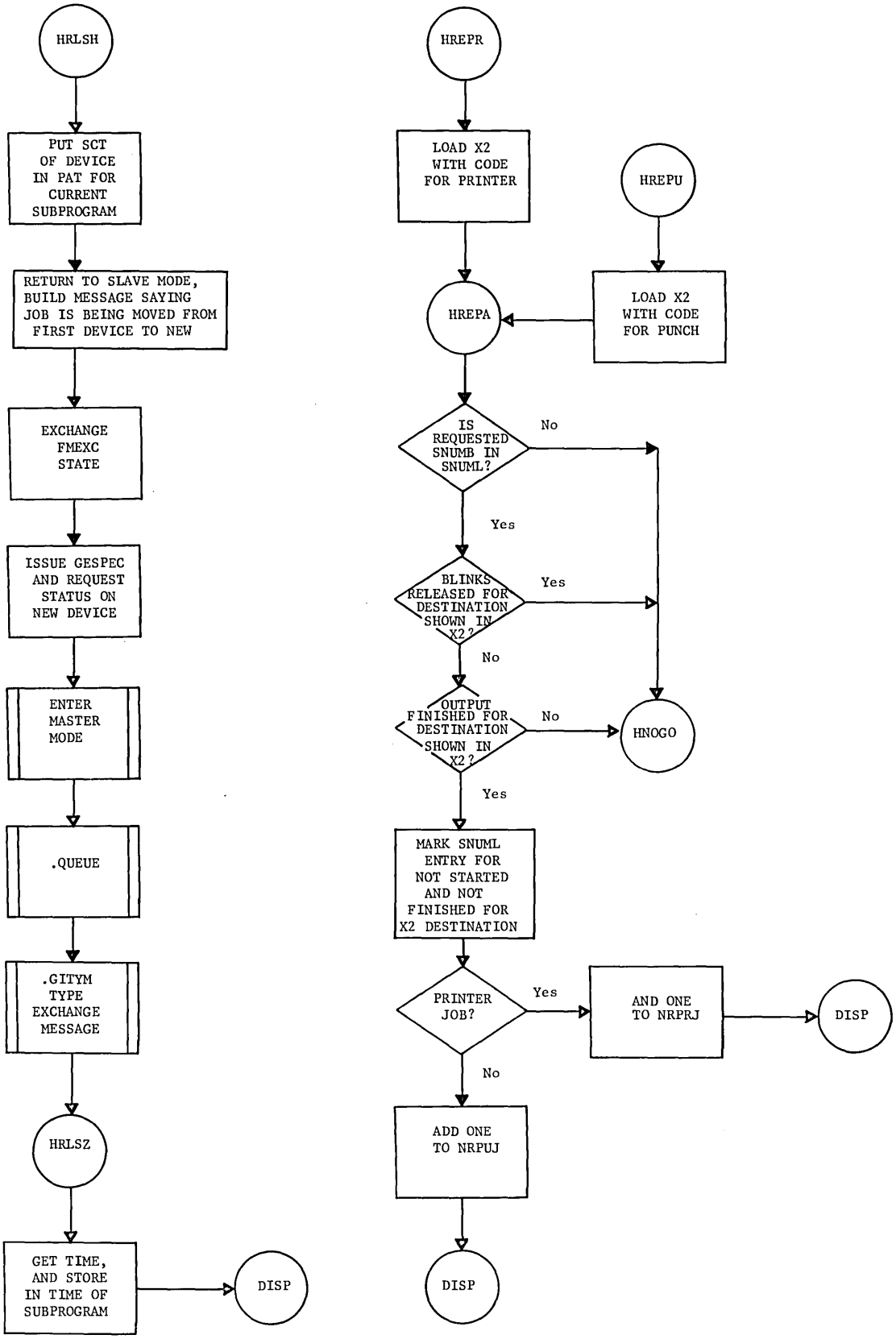


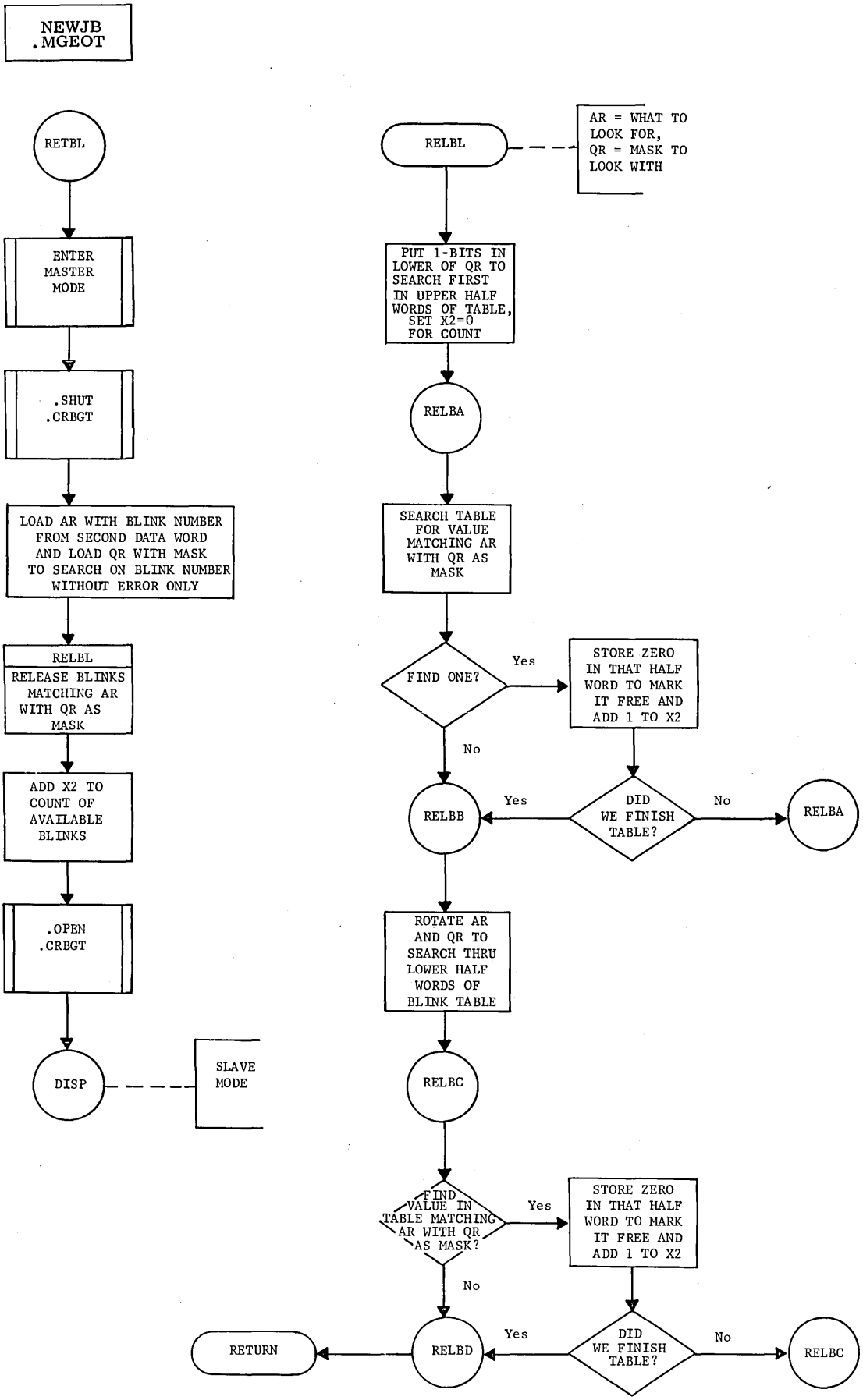
NEWJB
.MGEOT

ANALYZE INPUT REQUEST

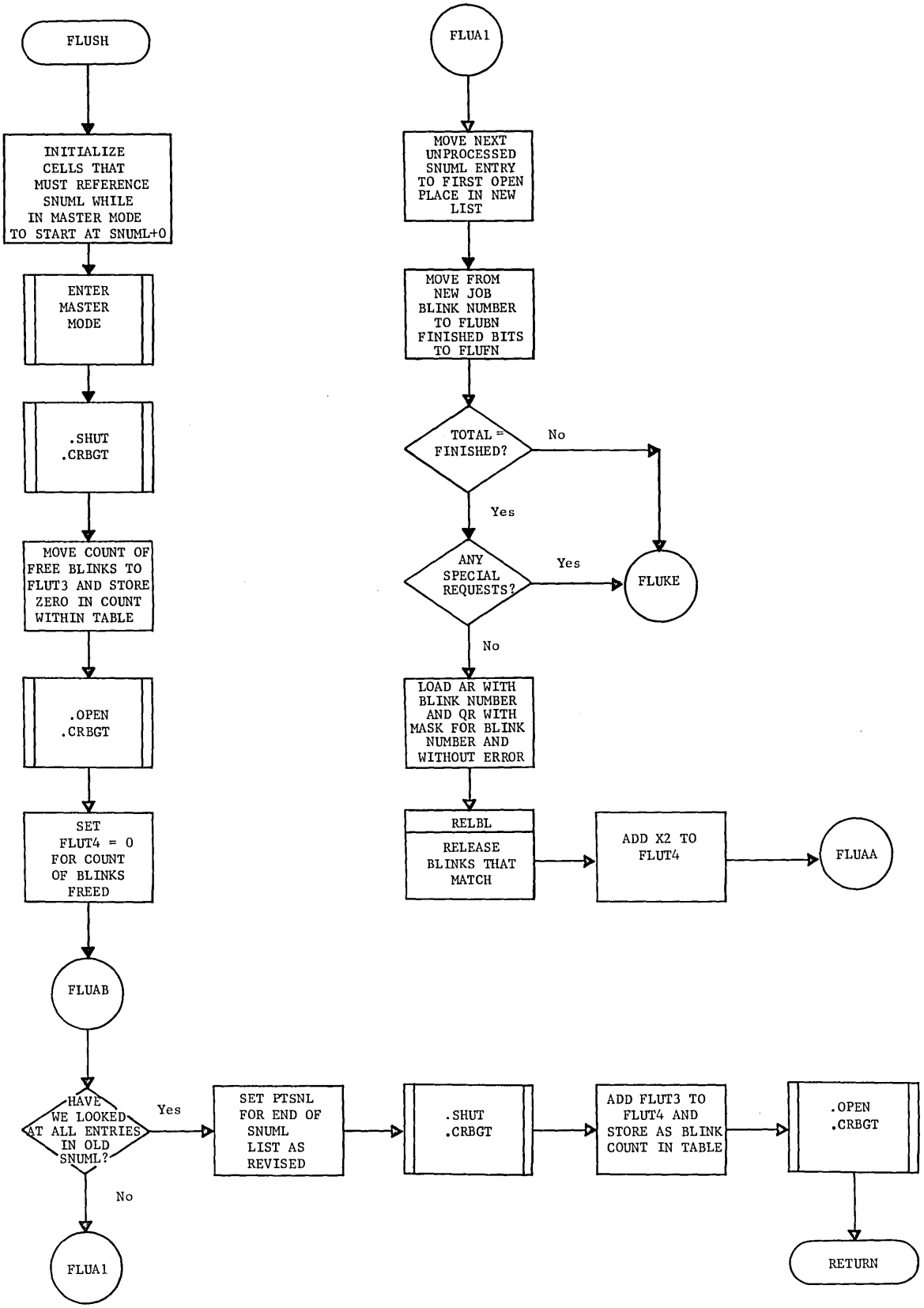


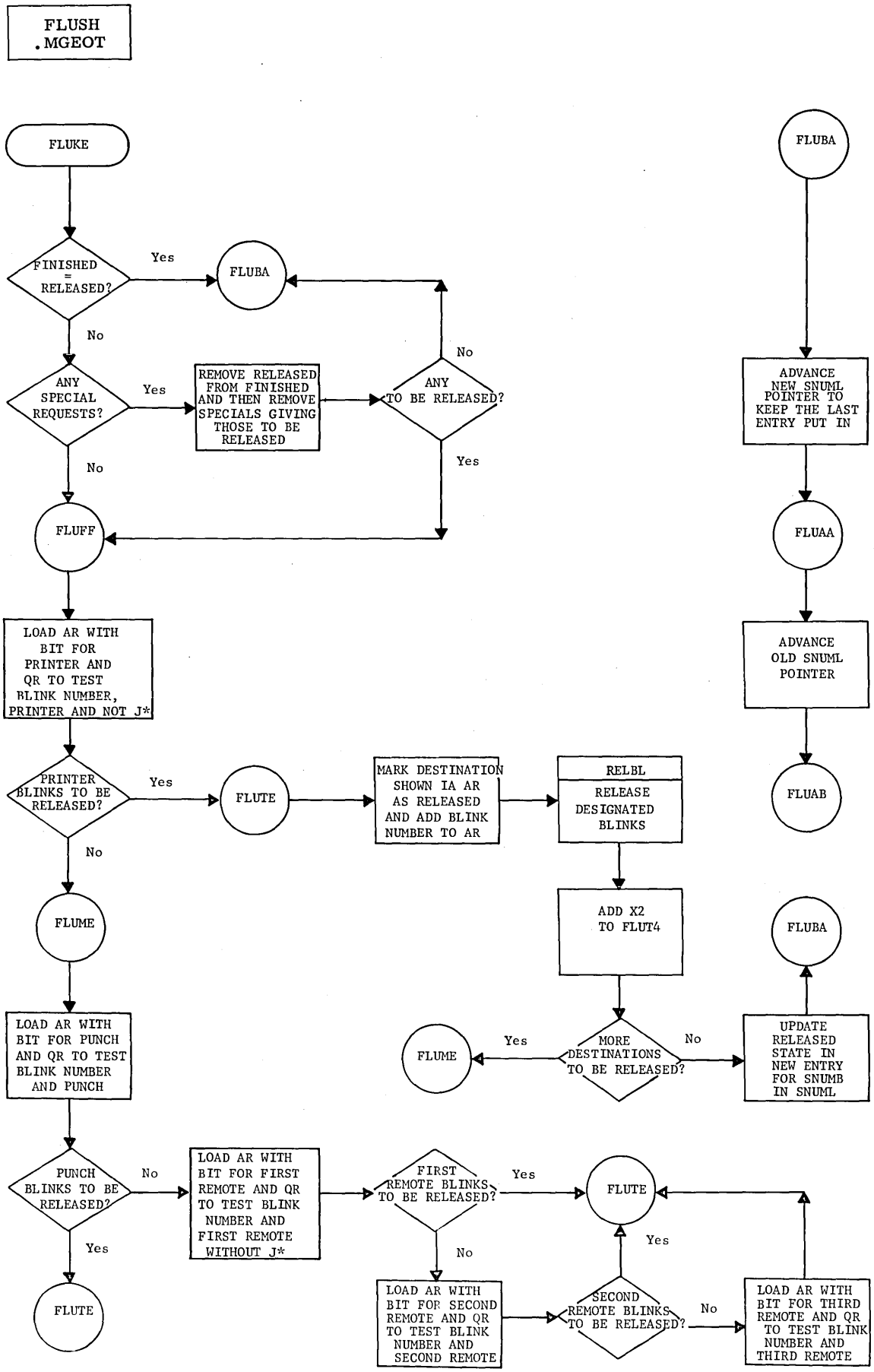

NEWJB
.MGEOT
HRELB
FLUSH
RELEASE BLINKS
DISP
HRELD
SCT OF DEVICE TO RELEASE IN FIRST WORD OF DATA
ENTER MASTER MODE
SEARCH OUR PATS FOR SCT OF DEVICE TO MATCH RELEASE DEVICE SCT
SLAVE MODE
FIND IT?
No
HNOGO
Yes
HRLSB
SLAVE MODE
FMWER=0?
Yes
SET FMRLS=1
No
HRLSC
DISP
HRLSC
SET FMEXC=1, ISSUE MINUS GESPEC ON DEVICE TO CANCEL OLD GESPEC
RELS
RELEASE DEVICE
HRLSE
ENTER MASTER MODE
FMDPR=0?
Yes
HRLSF
No
.CALL .MALC2 EP1 TO ASK FOR PRINTER
.CALL .MALC2 EP3 TO ASK FOR PUNCH
HRLSG
GET DEVICE?
No
HRLSZ
Yes
HRLSH


HRLSH
PUT SCT OF DEVICE IN PAT FOR CURRENT SUBPROGRAM
RETURN TO SLAVE MODE, BUILD MESSAGE SAYING JOB IS BEING MOVED FROM FIRST DEVICE TO NEW
EXCHANGE FMEXC STATE
ISSUE GESPEC AND REQUEST STATUS ON NEW DEVICE
ENTER MASTER MODE
.QUEUE
.GITYM TYPE EXCHANGE MESSAGE
HRLSZ
GET TIME, AND STORE IN TIME OF SUBPROGRAM
DISP
HREPR
LOAD X2 WITH CODE FOR PRINTER
HREPU
LOAD X2 WITH CODE FOR PUNCH
HREPA
IS REQUESTED SNUMB IN SNUML?
No
Yes
BLINKS RELEASED FOR DESTINATION SHOWN IN X2?
Yes
No
OUTPUT FINISHED FOR DESTINATION SHOWN IN X2?
No
HNOGO
Yes
MARK SNUML ENTRY FOR NOT STARTED AND NOT FINISHED FOR X2 DESTINATION
PRINTER JOB?
Yes
AND ONE TO NRPRJ
DISP
No
ADD ONE TO NRPUJ
DISP

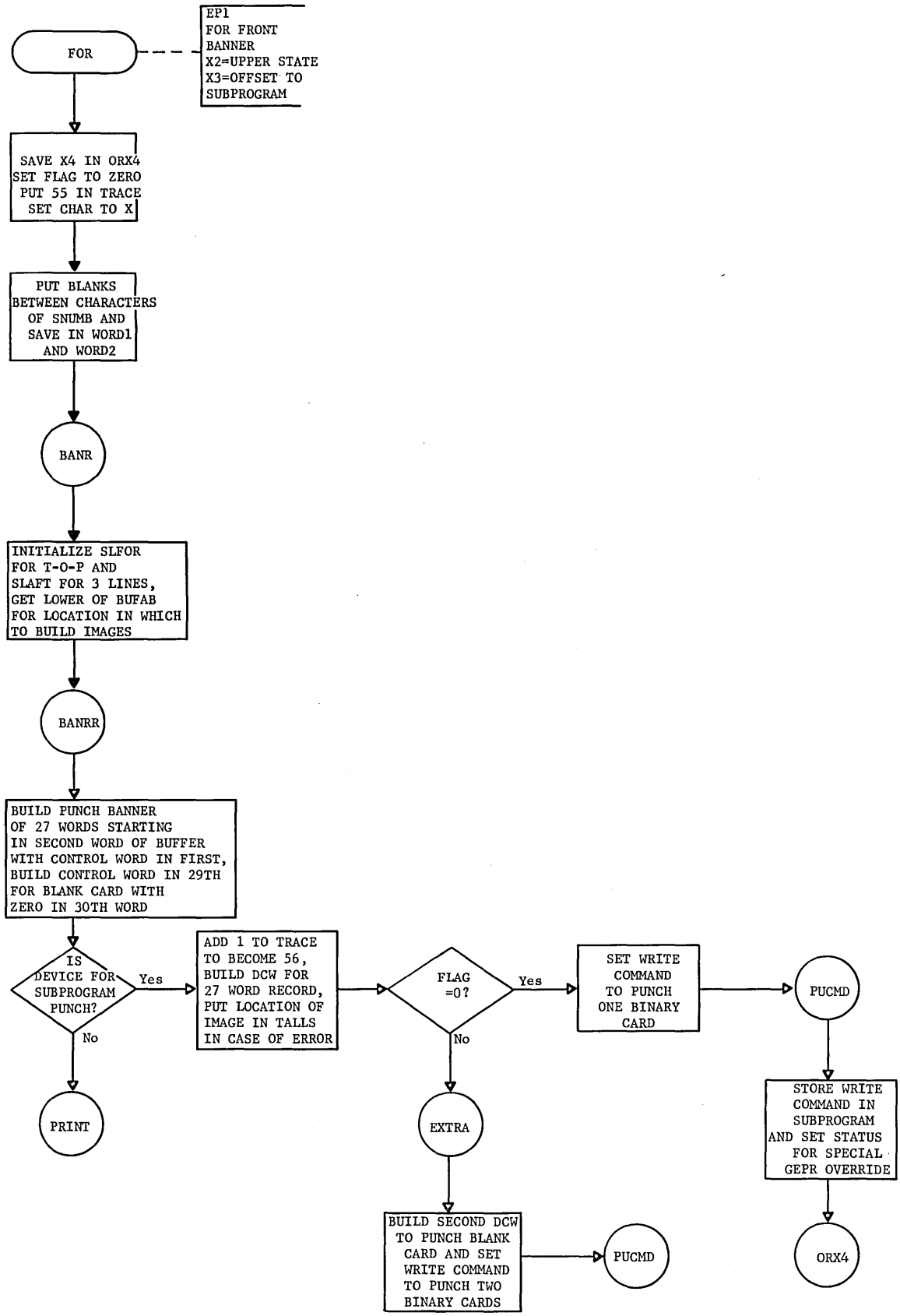

NEWJB
.MGEOT
RETBL
ENTER MASTER MODE
.SHUT
.CRBGT
LOAD AR WITH BLINK NUMBER FROM SECOND DATA WORD AND LOAD QR WITH MASK TO SEARCH ON BLINK NUMBER WITHOUT ERROR ONLY
RELBL
RELEASE BLINKS MATCHING AR WITH QR AS MASK
ADD X2 TO COUNT OF AVAILABLE BLINKS
.OPEN
.CRBGT
DISP
SLAVE MODE
RELBL
AR = WHAT TO LOOK FOR, QR = MASK TO LOOK WITH
PUT 1-BITS IN LOWER OF QR TO SEARCH FIRST IN UPPER HALF WORDS OF TABLE, SET X2=0 FOR COUNT
RELBA
SEARCH TABLE FOR VALUE MATCHING AR WITH QR AS MASK
FIND ONE?
Yes
STORE ZERO IN THAT HALF WORD TO MARK IT FREE AND ADD 1 TO X2
No
RELBB
Yes
DID WE FINISH TABLE?
No
RELBA
ROTATE AR AND QR TO SEARCH THRU LOWER HALF WORDS OF BLINK TABLE
RELBC
FIND VALUE IN TABLE MATCHING AR WITH QR AS MASK?
Yes
STORE ZERO IN THAT HALF WORD TO MARK IT FREE AND ADD 1 TO X2
No
RETURN
RELBD
Yes
DID WE FINISH TABLE?
No
RELBC

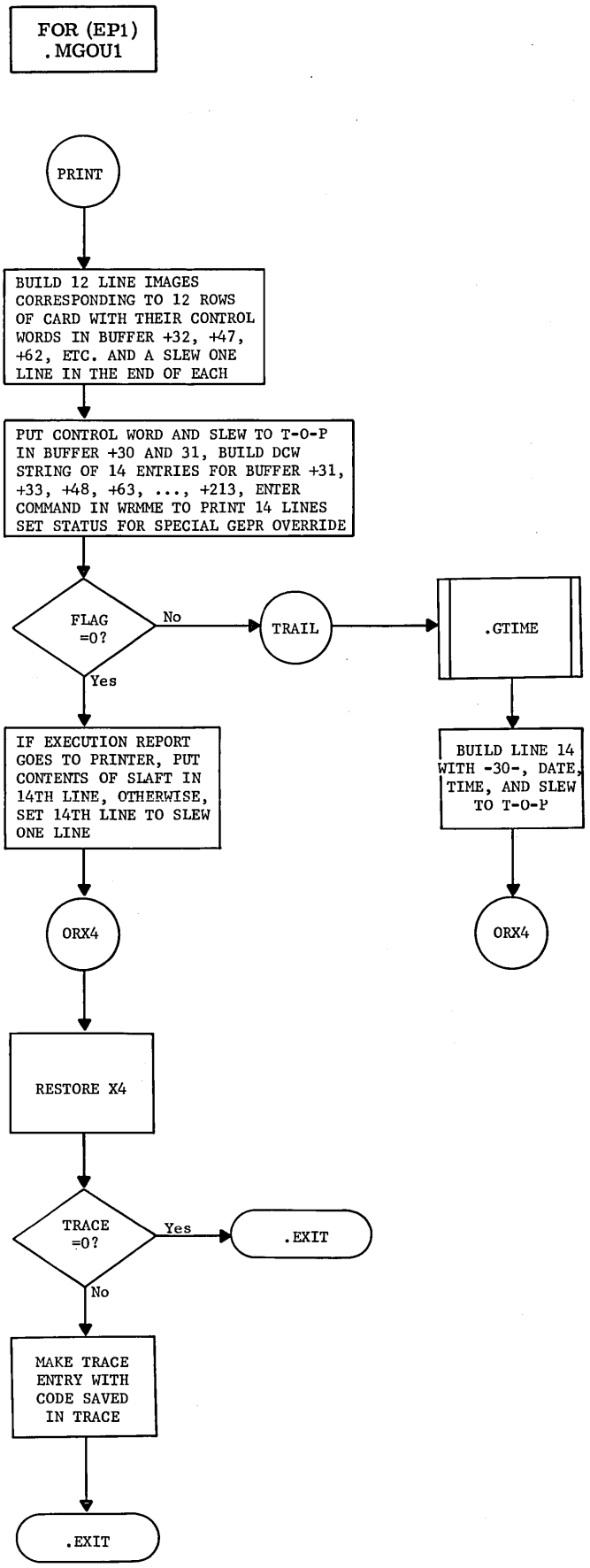
RELEASE BLINKS



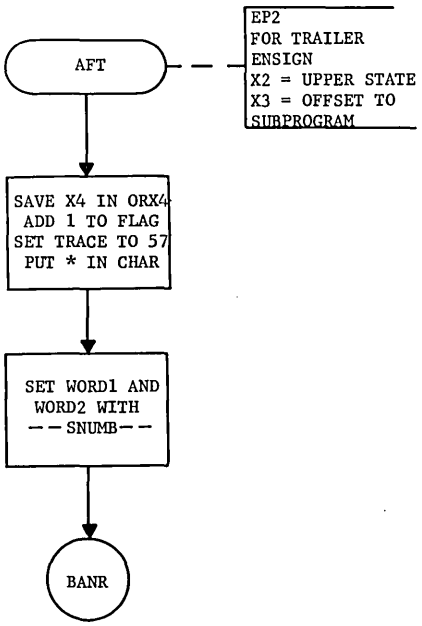

FLUSH
.MGEOT
FLUKE
FINISHED = RELEASED?
Yes
FLUBA
No
ANY SPECIAL REQUESTS?
Yes
REMOVE RELEASED FROM FINISHED AND THEN REMOVE SPECIALS GIVING THOSE TO BE RELEASED
ANY TO BE RELEASED?
No
Yes
No
FLUFF
LOAD AR WITH BIT FOR PRINTER AND QR TO TEST BLINK NUMBER, PRINTER AND NOT J*
PRINTER BLINKS TO BE RELEASED?
Yes
FLUTE
MARK DESTINATION SHOWN IA AR AS RELEASED AND ADD BLINK NUMBER TO AR
RELBL
RELEASE DESIGNATED BLINKS
ADD X2 TO FLUT4
No
FLUME
LOAD AR WITH BIT FOR PUNCH AND QR TO TEST BLINK NUMBER AND PUNCH
PUNCH BLINKS TO BE RELEASED?
No
LOAD AR WITH BIT FOR FIRST REMOTE AND QR TO TEST BLINK NUMBER AND FIRST REMOTE WITHOUT J*
FIRST REMOTE BLINKS TO BE RELEASED?
Yes
FLUTE
Yes
FLUTE
No
LOAD AR WITH BIT FOR SECOND REMOTE AND QR TO TEST BLINK NUMBER AND SECOND REMOTE
SECOND REMOTE BLINKS TO BE RELEASED?
Yes
No
LOAD AR WITH BIT FOR THIRD REMOTE AND QR TO TEST BLINK NUMBER AND THIRD REMOTE
FLUME
Yes
MORE DESTINATIONS TO BE RELEASED?
No
UPDATE RELEASED STATE IN NEW ENTRY FOR SNUMB IN SNUML
FLUBA
FLUBA
ADVANCE NEW SNUML POINTER TO KEEP THE LAST ENTRY PUT IN
FLUAA
ADVANCE OLD SNUML POINTER
FLUAB

CREATE FRONT BANNER



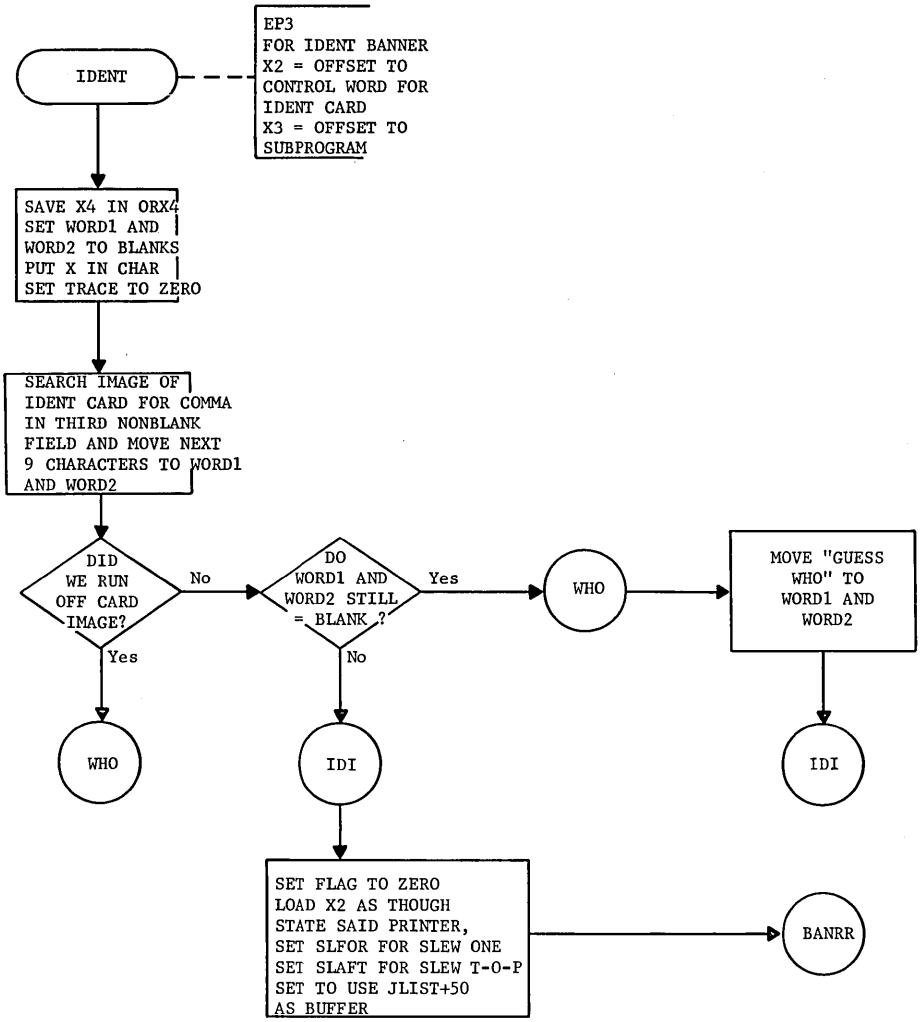

FOR (EP1)
.MGOU1
PRINT
BUILD 12 LINE IMAGES CORRESPONDING TO 12 ROWS OF CARD WITH THEIR CONTROL WORDS IN BUFFER +32, +47, +62, ETC. AND A SLEW ONE LINE IN THE END OF EACH
PUT CONTROL WORD AND SLEW TO T-O-P IN BUFFER +30 AND 31, BUILD DCW STRING OF 14 ENTRIES FOR BUFFER +31, +33, +48, +63, ..., +213, ENTER COMMAND IN WRMME TO PRINT 14 LINES SET STATUS FOR SPECIAL GEPR OVERRIDE
FLAG =0?
No
TRAIL
.GTIME
Yes
IF EXECUTION REPORT GOES TO PRINTER, PUT CONTENTS OF SLAFT IN 14TH LINE, OTHERWISE, SET 14TH LINE TO SLEW ONE LINE
BUILD LINE 14 WITH -30-, DATE, TIME, AND SLEW TO T-O-P
ORX4
ORX4
RESTORE X4
TRACE =0?
Yes
.EXIT
No
MAKE TRACE ENTRY WITH CODE SAVED IN TRACE
.EXIT

CREATE TRAILER ENSIGN



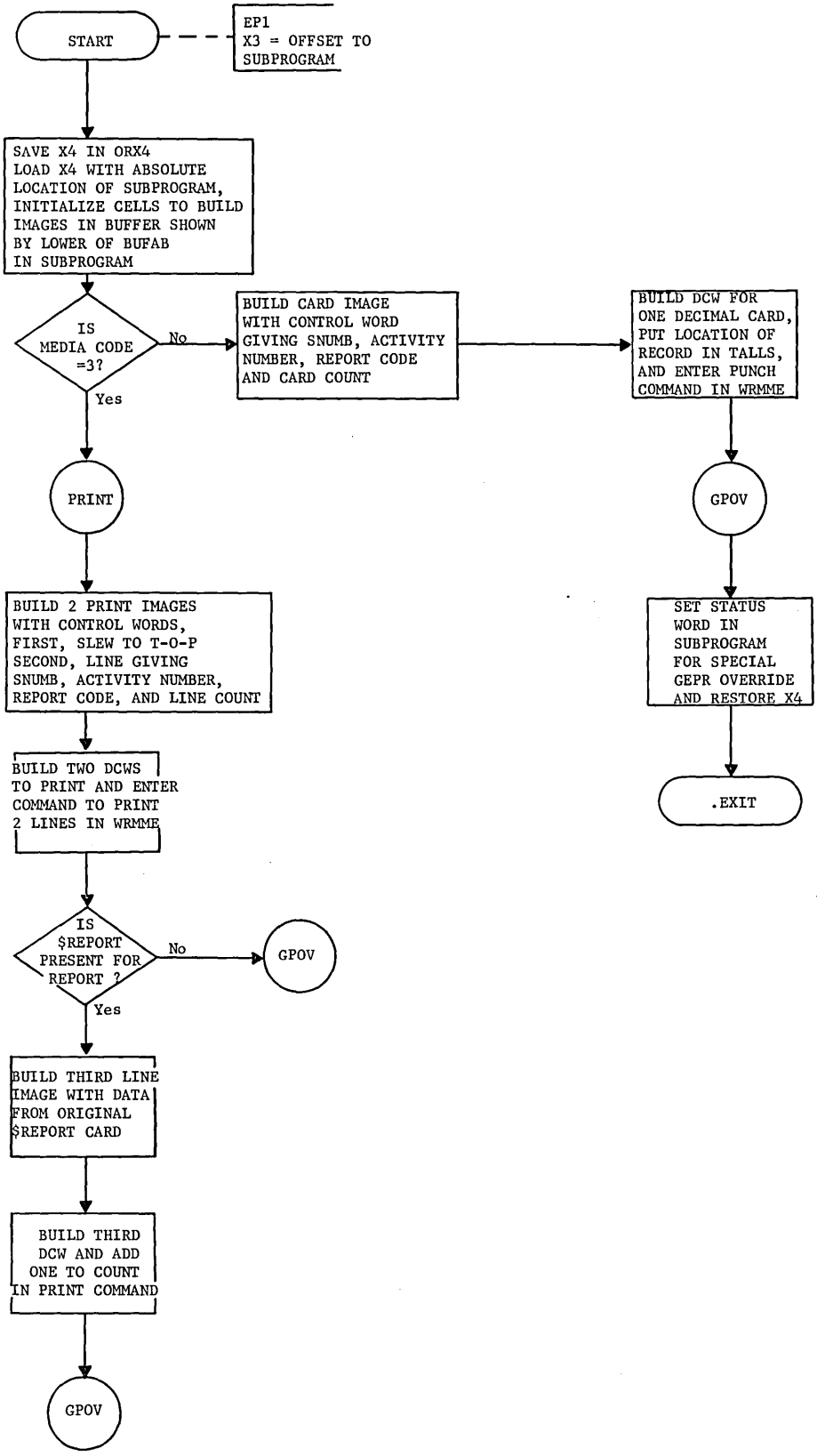
IDENT (EP3)
.MGOU1

CREATE IDENT BANNER



START (EP1)
.MGOU3

CREATE REPORT HEADER



INDEX

ABORT
   TAKE DUMP FOR SYSTEM ABORT, FSB 77
   ABORT I/O FOR PROGRAM, ABTIO 192

ABSOLUTE
   WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324

ABTIO
   ABORT I/O FOR PROGRAM, ABTIO 192
   ABTIO (EP9) .MIOS 192

ACCOUNTING
   FORMAT ERROR ACCOUNTING RECORD, FMTAR 142
   ACCOUNTING FILE REQUEST, ACTFL 198
   NORMAL CLOSE OF ACCOUNTING FILE, ACTS1 281
   ERROR CLOSE OF ACCOUNTING FILE, ACTS2 282
   EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289

ACCUMULATE
   ACCUMULATE PROCESSOR TIME, DACNB 26

ACTFL
   ACCOUNTING FILE REQUEST, ACTFL 198
   ACTFL (EP13) .MIOS 198

ACTION
   GIVE I/O ACTION PRIORITY, FPRIO 106

ACTIVITY
   PROCESS END OF ACTIVITY, EOA 311

ACTS1
   NORMAL CLOSE OF ACCOUNTING FILE, ACTS1 281
   ACTS1 (EP1) .MACTS 281

ACTS2
   ERROR CLOSE OF ACCOUNTING FILE, ACTS2 282
   ACTS2 (EP2) .MACTS 282

ACTS3
   EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289
   ACTS3 (EP3) .MACTS 289

ADDRESS
   GET ADDRESS FOR FILE CODE, FGAD (.MFALT) 78
   GET ADDRESS FOR FILE CODE, FGAD (.MFLT1) 95

AFT
   CREATE TRAILER ENSIGN, AFT 377
   AFT (EP2) .MGOU1 377

ALARM
   SET ALARM, SCK 32

ALLOCATION
1. DISPATCHER AND PERIPHERAL ALLOCATION 1

ALLOW
ALLOW A PROGRAM TO DELAY, FWAKE 109

ANALYZE
ANALYZE INPUT REQUEST, NEWJB 369

ASIGN
ASSIGN DEVICE AND JOB, ASIGN 366
ASIGN .MGEOT 366

ASSIGN
ASSIGN AN I/O ENTRY, QUEUE 118
ASSIGN DEVICE AND JOB, ASIGN 366

BANNER
CREATE FRONT BANNER, FOR 375
CREATE IDENT BANNER, IDENT 378

BAR
RESET BAR TO SMALLER AREA, FLBAR 107

BITS
RESET PROGRAM SWITCH WORD BITS, FGRET 82
SET PROGRAM SWITCH WORD BITS, FGSET 83

BLINKS
RELEASE BLINKS, FLUSH 373

BOOT
CALL .MDUMP AT EP2, BOOT 89
BOOT (EP13) .MFALT 89

BUFFER
SAVE TRACE BUFFER AND WRITE OUT, TRACF 34
READ INTO BUFFER, READ 338
WRITE FROM BUFFER, WRITE 340

CALCULATE
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170

CALL
CALL .MDUMP AT EP2, BOOT 89
PROCESS CALL FROM GEIN, GEIN 325

CARD PUNCH
CARD PUNCH INTERRUPT HANDLER, CPIT 204
CARD PUNCH REQUEST, CPIO 207
CARD PUNCH INITIALIZATION, .ICPIO 208

CARD READER
CARD READER INTERRUPT HANDLER, CRIT 255
CARD READER REQUEST, CRIO 257
CARD READER INITIALIZATION, .IGPIO 258

CCCRD
COURTESY CALL FOR READ, CCCRD 351
CCCRD .MGEOT 351

CCCWR
COURTESY CALL FOR WRITE, CCCWR 354
CCCWR .MGEOT 354

CCGSP
COURTESY CALL FOR GESPEC, CCGSP 356
CCGSP .MGEOT 356

CHANNEL
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170
RESUME I/O ON CHANNEL, RSMCH 197

CLOSE
CLOSE SYSTEM GATES, SHUG 52
NORMAL CLOSE OF ACCOUNTING FILE, ACTS1 281
ERROR CLOSE OF ACCOUNTING FILE, ACTS2 282
EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289

COMMAND
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER 188

CONNECT
CONNECT MULTIRECORD SIMULATION DCW, STIOM 187
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER 188
CONNECT SELECTED GESPECED ENTRY, STGPC 189

CONTROL
RELINQUISH CONTROL, RLC 20
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT 28

COURTESY CALL
END COURTESY CALL, ENCC 21
COURTESY CALL FOR READ, CCCRD 351
COURTESY CALL FOR WRITE, CCCWR 354
COURTESY CALL FOR GESPEC, CCGSP 356

CPIO
CARD PUNCH REQUEST, CPIO 207
CPIO (EP2) .MCPIO 207

CPIT
CARD PUNCH INTERRUPT HANDLER, CPIT 204
CPIT (EP1) .MCPIO 204

CREATE
CREATE FRONT BANNER, FOR 375
CREATE TRAILER ENSIGN, AFT 377
CREATE IDENT BANNER, IDENT 378
CREATE REPORT HEADER, START 379

CRIO
CARD READER REQUEST, CRIO 257
CRIO (EP2) .MGPIO 257

CRIT
CARD READER INTERRUPT HANDLER, CRIT 255
CRIT (EP1) .MGPIO 255

DACNB
ACCUMULATE PROCESSOR TIME, DACNB 26
DACNB (EP9) .MDISP 26

DATA
PROVIDE DATA AND TIME OF DAY, FGTIM 84

DCW
DCW POINTER VALIDATION, DCWCK 139
CONNECT MULTIRECORD SIMULATION DCW, STIOM 187

DCWCK
DCW POINTER VALIDATION, DCWCK 139
DCWCK .MIOS 139

DEALLOCATE
DEALLOCATE MEMORY, GMRL (.MFALT) 88
DEALLOCATE MEMORY, GMRL (.MFLT1) 101

DELAY
ALLOW A PROGRAM TO DELAY, FWAKE 109

DEVICE
ASSIGN DEVICE AND JOB, ASIGN 366

DISC
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29

DISP
DISPATCHER, DISP 326
DISP .MGEOT 326

DISPATCHER
1. DISPATCHER AND PERIPHERAL ALLOCATION 1
DISPATCHER, DISP 326

DIVIDE
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105

DMIOA
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29
DMIOA (EP12) .MDISP 29

DO
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29

DRGP
MDS200 ERROR AND EOF RECOVERY, DRGP 217
DRGP (EP4) .MDR20 217

DRIO
MDS200 REQUEST, DRIO 212
DRIO (EP2) .MDR20 212

DRIT
MDS200 INTERRUPT HANDLER, DRIT 209
DRIT (EP1) .MDR20 209

DRSL
MDS200 SELECT, DRSL 214
DRSL (EP3) .MDR20 214

DRSTR
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27
DRSTR (EP10) .MDISP 27

DSCNT
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT 28
DSCNT (EP11) .MDISP 28

DSGP
DSU200 ERROR AND EOF RECOVERY, DSGP 236
DSGP (EP4) .MDS20 236

DSIO
DSU200 REQUEST, DSIO 233
DSIO (EP2) .MDS20 233

DSIT
DSU200 INTERRUPT HANDLER, DSIT 231
DSIT (EP1) .MDS20 231

DSP
REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP 3
DSP (EP1) .MDISP 3

DSPQH
PROGRAM NUMBER AT FRONT OF QUEUE, DSPQH 24
DSPQH (EP7) .MDISP 24

DSPQM
PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT, DSPQM 33
DSPQM (EP14) .MDISP 33

DSPQT
PROGRAM NUMBER AT END OF QUEUE, DSPQT 25
DSPQT (EP8) .MDISP 25

DSSL
DSU200 SELECT, DSSL 234
DSSL (EP3) .MDS20 234

DSU200
DSU200 INTERRUPT HANDLER, DSIT 231
DSU200 REQUEST, DSIO 233
DSU200 SELECT, DSSL 234
DSU200 ERROR AND EOF RECOVERY, DSGP 236
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP 247
DSU200 INITIALIZATION, .IDS20 254

DUMP
TAKE DUMP FOR SYSTEM ABORT, FSB 77

ENABLE
ENABLE PROGRAM, ENB 22

ENABLED
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT 28

ENB
ENABLE PROGRAM, ENB 22
ENB (EP6) .MDISP 22

ENCC
END COURTESY CALL, ENCC 21
ENCC (EP5) .MDISP 21

END
END COURTESY CALL, ENCC 21
PROGRAM NUMBER AT END OF QUEUE, DSPQT 25
LINK I/O TO END OF QUEUE, LINK 113
PROCESS END OF ACTIVITY, EOA 311
PROCESS END OF JOB, EOJ 313

ENSIGN
CREATE TRAILER ENSIGN, AFT 377

ENTER
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86

ENTRY
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM 110
ASSIGN AN I/O ENTRY, QUEUE 118
UNLINK I/O ENTRY, UNLNK 169
CONNECT SELECTED GESPECED ENTRY, STGPC 189
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP 222
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP 247

EOA
PROCESS END OF ACTIVITY, EOA 311
EOA (EP3) .MSYOT 311

EOF
MDS200 ERROR AND EOF RECOVERY, DRGP 217
DSU200 ERROR AND EOF RECOVERY, DSGP 236

EOJ
PROCESS END OF JOB, EOJ 313
EOJ (EP4) .MSYOT 313

ERROR
FORMAT ERROR ACCOUNTING RECORD, FMTAR 142
MDS200 ERROR AND EOF RECOVERY, DRGP 217
DSU200 ERROR AND EOF RECOVERY, DSGP 236
ERROR CLOSE OF ACCOUNTING FILE, ACTS2 282

EXCA
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324
EXCA (EP5) .MSYOT 324

EXEC
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310
EXEC (EP2) .MSYOT 310

EXECUTION
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324

EXTERNAL
EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289

FAULT
2. FAULT PROCESSING AND SERVICE MME'S 53
SECONDARY FAULT VECTOR, FPRC 55
RECOGNIZE FAULT TYPE AND PROCESS, FLT 56
FAULT PROCESSOR INITIALIZATION, .IFALT 91

FCB
PUT INFORMATION IN FCB, FGCON 79
PUT INFORMATION IN FCB, FCON 96

FCON
PUT INFORMATION IN FCB, FCON 96
FCON (EP2) .MFLT1 96

FEMM
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86
FEMM (EP10) .MFALT 86

FGAD
GET ADDRESS FOR FILE CODE, FGAD (.MFALT) 78
FGAD (EP2) .MFALT 78
GET ADDRESS FOR FILE CODE, FGAD (.MFLT1) 95
FGAD (EP1) .MFLT1 95

FGCON
PUT INFORMATION IN FCB, FGCON 79
FGCON (EP3) .MFALT 79

FGFIL
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL 80
FGFIL (EP4) .MFALT 80

FGLAP
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81
FGLAP (EP5) .MFALT 81

FGLP
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
FGLP (EP9) .MFALT 85
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105
FGLP (EP5) .MFLT1 105

FGRET
RESET PROGRAM SWITCH WORD BITS, FGRET 82
FGRET (EP6) .MFALT 82

FGSET
SET PROGRAM SWITCH WORD BITS, FGSET 83
FGSET (EP7) .MFALT 83

FGTIM
PROVIDE DATA AND TIME OF DAY, FGTIM 84
FGTIM (EP8) .MFALT 84

FILE
GET ADDRESS FOR FILE CODE, FGAD (.MFALT) 78
FIND PAT FROM FILE CODE, FNDE 87
GET ADDRESS FOR FILE CODE, FGAD (.MFLT1) 95
FIND FILE CODE, FNDFC 141
ACCOUNTING FILE REQUEST, ACTFL 198
NORMAL CLOSE OF ACCOUNTING FILE, ACTS1 281
ERROR CLOSE OF ACCOUNTING FILE, ACTS2 282
EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289

FIND
FIND PAT FROM FILE CODE, FNDE 87
FIND FILE CODE, FNDFC 141

FLBAR
RESET BAR TO SMALLER AREA, FLBAR 107
FLBAR (EP7) .MFLT1 107

FLSW
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW 100
FLSW (EP3) .MFLT1 100

FLT
RECOGNIZE FAULT TYPE AND PROCESS, FLT 56
FLT .MFALT 56

FLUSH
RELEASE BLINKS, FLUSH 373
FLUSH .MGEOT 373

FMTAR
FORMAT ERROR ACCOUNTING RECORD, FMTAR 142
FMTAR .MIOS 142

FNDE
FIND PAT FROM FILE CODE, FNDE 87
FNDE (EP11) .MFALT 87

FNDFC
FIND FILE CODE, FNDFC 141
FNDFC .MIOS 141

FORCED
FORCED RELINQUISH, FRLC 19

FORMAT
FORMAT ERROR ACCOUNTING RECORD, FMTAR 142
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196

FPRC
SECONDARY FAULT VECTOR, FPRC 55
FPRC .MFALT 55

FPRIO
GIVE I/O ACTION PRIORITY, FPRIO 106
FPRIO (EP6) .MFLT1 106

FRLC
FORCED RELINQUISH, FRLC 19
FRLC (EP3) .MDISP 19

FRONT
PROGRAM NUMBER AT FRONT OF QUEUE, DSPQH 24
LINK I/O TO FRONT OF QUEUE, LINKF 116
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117
CREATE FRONT BANNER, FOR 375

FSB
TAKE DUMP FOR SYSTEM ABORT, FSB 77
FSB (EP1) .MFALT 77

FWAKE
ALLOW A PROGRAM TO DELAY, FWAKE 109
FWAKE (EP8) .MFLT1 109

GATES
OPEN SYSTEM GATES, OPGAT 51
CLOSE SYSTEM GATES, SHUG 52

GEIN
PROCESS CALL FROM GEIN, GEIN 325
PROCESS CALL FROM GEIN, GEIN 325
GEIN (EP6) .MSYOT 325

GEINOS
MME GEINOS PROCESSOR, INOS 120

GEPR
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27

GESPEC
MME GESPEC PROCESSOR, SPEC 133
COURTESY CALL FOR GESPEC, CCGSP 356

GESPECED
CONNECT SELECTED GESPECED ENTRY, STGPC 189

GESYOT
PROCESS MME GESYOT, MME 293

GET
GET ADDRESS FOR FILE CODE, FGAD (.MFALT) 78
GET ADDRESS FOR FILE CODE, FGAD (.MFLT1) 95

GIVE
GIVE I/O ACTION PRIORITY, FPRIO 106

GMRL
DEALLOCATE MEMORY, GMRL (.MFALT) 88
GMRL (EP12) .MFALT 88
DEALLOCATE MEMORY, GMRL (.MFLT1) 101
GMRL (EP4) .MFLT1 101

GMRLM
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM 110
GMRLM (EP9) .MFLT1 110

GRD
ROADBLOCK, GRD 17
GRD (EP2) .MDISP 17

GSTRT
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196
GSTRT (EP10) .MIOS 196

HCL
.CALL MACRO, HCL 36
HCL .MDISP 36

HEADER
CREATE REPORT HEADER, START 379

HEX
.EXIT MACRO, HEX 38
HEX .MDISP 38

HGT
.GOTO MACRO, HGT 40
HGT .MDISP 40

IDENT
CREATE IDENT BANNER, IDENT 378
CREATE IDENT BANNER, IDENT 378
IDENT (EP3) .MGOU1 378

ILPCX
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170
ILPCX .MIOS 170

INDEX
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170
INDEX 381

INFORMATION
PUT INFORMATION IN FCB, FGCON 79
PUT INFORMATION IN FCB, FCON 96

INITIALIZATION
INITIALIZATION, .IDISP 35
FAULT PROCESSOR INITIALIZATION, .IFALT 91
MAIN IOS MODULE INITIALIZATION, .IIOS 201
CARD PUNCH INITIALIZATION, .ICPIO 208
MDS200 INITIALIZATION, .IDR20 230
DSU200 INITIALIZATION, .IDS20 254
CARD READER INITIALIZATION, .IGPIO 258

INITIALIZATION (continued)
MAGNETIC TAPE INITIALIZATION, .IMTAP 265
PRINTER INITIALIZATION, .IPRIO 270
PAPER TAPE INITIALIZATION, .IPTAP 273
TYPEWRITER INITIALIZATION, .ITYPE 280

INOS
MME GEINOS PROCESSOR, INOS 120
INOS (EP5) .MIOS 120

INPUT
ANALYZE INPUT REQUEST, NEWJB 369

INTERRUPT
REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP 3
PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT, DSPQM 33

INTERRUPT HANDLER
INTERRUPT HANDLER, IOTRM 144
CARD PUNCH INTERRUPT HANDLER, CPIT 204
MDS200 INTERRUPT HANDLER, DRIT 209
DSU200 INTERRUPT HANDLER, DSIT 231
CARD READER INTERRUPT HANDLER, CRIT 255
MAGNETIC TAPE INTERRUPT HANDLER, MTIT 259
PRINTER INTERRUPT HANDLER, PRIT 266
PAPER TAPE INTERRUPT HANDLER, PTIT 271
TYPEWRITER INTERRUPT HANDLER, TYIT 274

IOS
MAIN IOS MODULE INITIALIZATION, .IIOS 201

IOTRM
INTERRUPT HANDLER, IOTRM 144
IOTRM .MIOS 144

ITYM
MASTER MESSAGE PROCESSOR, ITYM 135
ITYM (EP7) .MIOS 135

I/O
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29
GIVE I/O ACTION PRIORITY, FPRIO 106
3. I/O SUPERVISION 111
LINK I/O TO END OF QUEUE, LINK 113
LINK I/O TO FRONT OF QUEUE, LINKF 116
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117
ASSIGN AN I/O ENTRY, QUEUE 118
UNLINK I/O ENTRY, UNLNK 169
START I/O, STIO 175
RESUME I/O FOR PROGRAM, RSMIO 190
ABORT I/O FOR PROGRAM, ABTIO 192
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196
RESUME I/O ON CHANNEL, RSMCH 197

JOB
PROCESS END OF JOB, EOJ 313
ASSIGN DEVICE AND JOB, ASIGN 366
START JOB, STRJB 368

LET
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86

LINK
LINK I/O TO END OF QUEUE, LINK 113
LINK I/O TO END OF QUEUE, LINK 113
LINK (EP1) .MIOS 113
LINK I/O TO FRONT OF QUEUE, LINKF 116
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117

LINKF
LINK I/O TO FRONT OF QUEUE, LINKF 116
LINKF (EP2) .MIOS 116

LINKR
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117
LINKR (EP3) .MIOS 117

LOGICAL
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170

LOGICAL UNITS
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL 80
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW 100

MACRO
.CALL MACRO, HCL 36
.EXIT MACRO, HEX 38
.GOTO MACRO, HGT 40

MAGNETIC TAPE
MAGNETIC TAPE INTERRUPT HANDLER, MTIT 259
MAGNETIC TAPE REQUEST, MTIO 264
MAGNETIC TAPE INITIALIZATION, .IMTAP 265

MAIN
MAIN IOS MODULE INITIALIZATION, .IIOS 201

MASTER
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86
MASTER MESSAGE PROCESSOR, ITYM 135

MDS200
MDS200 INTERRUPT HANDLER, DRIT 209
MDS200 REQUEST, DRIO 12
MDS200 SELECT, DRSL 214
MDS200 ERROR AND EOF RECOVERY, DRGP 217
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP 222
MDS200 INITIALIZATION, .IDR20 230

MEMORY
DEALLOCATE MEMORY, GMRL (.MFALT) 88
DEALLOCATE MEMORY, GMRL (.MFLT1) 101
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM 110

MESSAGE
MASTER MESSAGE PROCESSOR, ITYM 135
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324

MME
MME GEINOS PROCESSOR, INOS 120
MME GESPEC PROCESSOR, SPEC 133
PROCESS MME GESYOT, MME 293
PROCESS MME GESYOT, MME 293
MME (EP1) .MSYOT 293

MODE
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86

MODULE
MAIN IOS MODULE INITIALIZATION, .IIOS 201

MOVE
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27

MTIO
MAGNETIC TAPE REQUEST, MTIO 264
MTIO (EP2) .MMTAP 264

MTIT
MAGNETIC TAPE INTERRUPT HANDLER, MTIT 259
MTIT (EP1) .MMTAP 259

MULTIRECORD
CONNECT MULTIRECORD SIMULATION DCW, STIOM 187

NEGATIVE
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP 222
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP 247

NEGATIVE EP
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP 222
NEGATIVE EP .MDR20 222
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP 247
NEGATIVE EP .MDS20 247

NEWJB
ANALYZE INPUT REQUEST, NEWJB 369
NEWJB .MGEOT 369

NORMAL
NORMAL CLOSE OF ACCOUNTING FILE, ACTS1 281

OPEN
OPEN SYSTEM GATES, OPGAT 51

OPGAT
OPEN SYSTEM GATES, OPGAT 51
OPGAT .MDISP 51

OUTPUT
4. TERMINATION AND SYSTEM OUTPUT 291

PAPER TAPE
PAPER TAPE INTERRUPT HANDLER, PTIT 271
PAPER TAPE REQUEST, PTIO 272
PAPER TAPE INITIALIZATION, .IPTAP 273

PAT
FIND PAT FROM FILE CODE, FNDE 87

PERIPHERAL
1. DISPATCHER AND PERIPHERAL ALLOCATION 1

POINTER
POINTER VALIDATION, PTRVL 138
DCW POINTER VALIDATION, DCWCK 139

POINTS
MDS200 NEGATIVE ENTRY POINTS, NEGATIVE EP 222
DSU200 NEGATIVE ENTRY POINTS, NEGATIVE EP 247

PRIMARY
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL 80
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW 100
CALCULATE LOGICAL PRIMARY CHANNEL INDEX, ILPCX 170

PRINTER
PRINTER INTERRUPT HANDLER, PRIT 266
PRINTER REQUEST, PRIO 269
PRINTER INITIALIZATION, .IPRIO 270

PRIO
PRINTER REQUEST, PRIO 269
PRIO (EP2) .MPRIO 269

PRIOR
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81

PRIORITY
GIVE I/O ACTION PRIORITY, FPRIO 106

PRIT
PRINTER INTERRUPT HANDLER, PRIT 266
PRIT (EP1) .MPRIO 266

PROCESS
RECOGNIZE FAULT TYPE AND PROCESS, FLT 56
PROCESS MME GESYOT, MME 293
PROCESS END OF ACTIVITY, EOA 311
PROCESS END OF JOB, EOJ 313
PROCESS CALL FROM GEIN, GEIN 325

PROCESSING
2. FAULT PROCESSING AND SERVICE MME'S 53

PROCESSOR
ACCUMULATE PROCESSOR TIME, DACNB 26
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81
FAULT PROCESSOR INITIALIZATION, .IFALT 91
MME GEINOS PROCESSOR, INOS 120
MME GESPEC PROCESSOR, SPEC 133
MASTER MESSAGE PROCESSOR, ITYM 135

PROGRAM
ENABLE PROGRAM, ENB 22
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT 28
RESET PROGRAM SWITCH WORD BITS, FGRET 82
SET PROGRAM SWITCH WORD BITS, FGSET 83
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86
ALLOW A PROGRAM TO DELAY, FWAKE 109
RESUME I/O FOR PROGRAM, RSMIO 190
ABORT I/O FOR PROGRAM, ABTIO 192

PROGRAM NUMBER
PROGRAM NUMBER AT FRONT OF QUEUE, DSPQH 24
PROGRAM NUMBER AT END OF QUEUE, DSPQT 25
PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT, DSPQM 33

PROVIDE
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81
PROVIDE DATA AND TIME OF DAY, FGTIM 84

PTIO
PAPER TAPE REQUEST, PTIO 272
PTIO (EP2) .MPTAP 272

PTIT
PAPER TAPE INTERRUPT HANDLER, PTIT 271
PTIT (EP1) .MPTAP 271

PTRVL
POINTER VALIDATION, PTRVL 138
PTRVL .MIOS 138

PUT
PUT INFORMATION IN FCB, FGCON 79
PUT INFORMATION IN FCB, FCON 96

QUEUE
PROGRAM NUMBER AT FRONT OF QUEUE, DSPQH 24
PROGRAM NUMBER AT END OF QUEUE, DSPQT 25
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29
PROGRAM NUMBER IN QUEUE FOLLOWING INTERRUPT, DSPQM 33
LINK I/O TO END OF QUEUE, LINK 113
LINK I/O TO FRONT OF QUEUE, LINKF 116
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117
ASSIGN AN I/O ENTRY, QUEUE 118
QUEUE (EP4) .MIOS 118

READ
READ INTO BUFFER, READ 338
READ INTO BUFFER, READ 338
READ .MGEOT 338

READ (continued)
COURTESY CALL FOR READ, CCCRD 351

RECOGNIZE
RECOGNIZE FAULT TYPE AND PROCESS, FLT 56

RECORD
FORMAT ERROR ACCOUNTING RECORD, FMTAR 142

RECOVERY
MDS200 ERROR AND EOF RECOVERY, DRGP 217
DSU200 ERROR AND EOF RECOVERY, DSGP 236

REDISPATCH
REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP 3

REISSUE
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER 188

REISSUED
LINK REISSUED I/O TO FRONT OF QUEUE, LINKR 117

RELATIVE
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310

RELEASE
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM 110
RELEASE BLINKS, FLUSH 373

RELINQUISH
FORCED RELINQUISH, FRLC 19
RELINQUISH CONTROL, RLC 20
RELINQUISH CONTROL UNTIL PROGRAM ENABLED, DSCNT 28

REPORT
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324
CREATE REPORT HEADER, START 379

REPORT/ACTIVITY
START REPORT/ACTIVITY, STACT 336

REQUEST
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81
ACCOUNTING FILE REQUEST, ACTFL 198
CARD PUNCH REQUEST, CPIO 207
MDS200 REQUEST, DRIO 212
DSU200 REQUEST, DSIO 233
CARD READER REQUEST, CRIO 257
MAGNETIC TAPE REQUEST, MTIO 264
PRINTER REQUEST, PRIO 269
PAPER TAPE REQUEST, PTIO 272
TYPEWRITER REQUEST, TYIO 277
EXTERNAL REQUEST FOR ACCOUNTING FILE CLOSE, ACTS3 289
ANALYZE INPUT REQUEST, NEWJB 369

RESET
RESET PROGRAM SWITCH WORD BITS, FGRET 82
RESET BAR TO SMALLER AREA, FLBAR 107

RESTORE
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27

RESUME
RESUME I/O FOR PROGRAM, RSMIO 190
RESUME I/O ON CHANNEL, RSMCH 197

RETURN
STATUS RETURN, STRET 171
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196

RLC
RELINQUISH CONTROL, RLC 20
RLC (EP4) .MDISP 20

ROADBLOCK
ROADBLOCK, GRD 17

RSMCH
RESUME I/O ON CHANNEL, RSMCH 197
RSMCH (EP12) .MIOS 197

RSMIO
RESUME I/O FOR PROGRAM, RSMIO 190
RSMIO (EP8) .MIOS 190

RUNOUT
REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP 3

SAVE
SAVE TRACE BUFFER AND WRITE OUT, TRACF 34

SCK
SET ALARM, SCK 32
SCK (EP13) .MDISP 32

SECONDARY
SECONDARY FAULT VECTOR, FPRC 55
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL 80
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW 100

SECONDS
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105

SELECT
MDS200 SELECT, DRSL 214
DSU200 SELECT, DSSL 234
TYPEWRITER SELECT, TYSL 279

SELECTED
CONNECT SELECTED GESPECED ENTRY, STGPC 189

SERVICE
2. FAULT PROCESSING AND SERVICE MME'S 53

SET
SET ALARM, SCK 32
SET PROGRAM SWITCH WORD BITS, FGSET 83

SHUG
CLOSE SYSTEM GATES, SHUG 52
SHUG .MDISP 52

SIMULATION
CONNECT MULTIRECORD SIMULATION DCW, STIOM 187

SLAVE
LET SLAVE PROGRAM ENTER MASTER MODE, FEMM 86

SPEC
MME GESPEC PROCESSOR, SPEC 133
SPEC (EP6) .MIOS 133

STACT
START REPORT/ACTIVITY, STACT 336
STACT .MGEOT 336

START
START I/O, STIO 175
START REPORT/ACTIVITY, STACT 336
START JOB, STRJB 368
CREATE REPORT HEADER, START 379
START (EP1) .MGOU3 379

STATE
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27

STATUS
STATUS RETURN, STRET 171
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196

STATUS WORDS
FORMAT I/O STATUS WORDS AND RETURN STATUS, GSTRT 196

STGPC
CONNECT SELECTED GESPECED ENTRY, STGPC 189
STGPC .MIOS 189

STIO
START I/O, STIO 175
STIO .MIOS 175

STIOM
CONNECT MULTIRECORD SIMULATION DCW, STIOM 187
STIOM .MIOS 187

STRET
STATUS RETURN, STRET 171
STRET .MIOS 171

STRJB
START JOB, STRJB 368
STRJB .MGEOT 368

SUPERVISION
3. I/O SUPERVISION 111

SWAP
RESTORE STATE AFTER SWAP, MOVE, GEPR, DRSTR 27

SWITCH
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FGFIL 80
RESET PROGRAM SWITCH WORD BITS, FGRET 82
SET PROGRAM SWITCH WORD BITS, FGSET 83
SWITCH PRIMARY AND SECONDARY LOGICAL UNITS, FLSW 100

SYSTEM
DO DISC I/O USING SYSTEM I/O QUEUE, DMIOA 29
SYSTEM TRACE, TRACE 49
OPEN SYSTEM GATES, OPGAT 51
CLOSE SYSTEM GATES, SHUG 52
TAKE DUMP FOR SYSTEM ABORT, FSB 77
4. TERMINATION AND SYSTEM OUTPUT 291

TAKE
TAKE DUMP FOR SYSTEM ABORT, FSB 77

TERMINATION
4. TERMINATION AND SYSTEM OUTPUT 291

TIME
ACCUMULATE PROCESSOR TIME, DACNB 26
PROVIDE PROCESSOR TIME PRIOR TO REQUEST, FGLAP 81
PROVIDE DATA AND TIME OF DAY, FGTIM 84
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105

TIMER
REDISPATCH AFTER INTERRUPT OR TIMER RUNOUT, DSP 3

TIME-SHARING
MEMORY RELEASE ENTRY FOR TIME-SHARING, GMRLM 110

TRACE
SAVE TRACE BUFFER AND WRITE OUT, TRACF 34
SYSTEM TRACE, TRACE 49
SYSTEM TRACE, TRACE 49
TRACE .MDISP 49

TRACF
SAVE TRACE BUFFER AND WRITE OUT, TRACF 34
TRACF (EP15) .MDISP 34

TRAILER
CREATE TRAILER ENSIGN, AFT 377

TYIO
TYPEWRITER REQUEST, TYIO 277
TYIO (EP2) .MTYPE 277

TYIT
TYPEWRITER INTERRUPT HANDLER, TYIT 274
TYIT (EP1) .MTYPE 274

TYPE
RECOGNIZE FAULT TYPE AND PROCESS, FLT 56

TYPER
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER 188
TYPER .MIOS 188

TYPEWRITER
CONNECT REISSUE OF SECOND TYPEWRITER COMMAND, TYPER 188
TYPEWRITER INTERRUPT HANDLER, TYIT 274
TYPEWRITER REQUEST, TYIO 277
TYPEWRITER SELECT, TYSL 279
TYPEWRITER INITIALIZATION, .ITYPE 280

TYSL
TYPEWRITER SELECT, TYSL 279
TYSL (EP3) .MTYPE 279

UNLINK
UNLINK I/O ENTRY, UNLNK 169

UNLNK
UNLINK I/O ENTRY, UNLNK 169
UNLNK .MIOS 169

VALIDATION
POINTER VALIDATION, PTRVL 138
DCW POINTER VALIDATION, DCWCK 139

VECTOR
SECONDARY FAULT VECTOR, FPRC 55

WORD
RESET PROGRAM SWITCH WORD BITS, FGRET 82
SET PROGRAM SWITCH WORD BITS, FGSET 83

WRITE
SAVE TRACE BUFFER AND WRITE OUT, TRACF 34
WRITE EXECUTION REPORT MESSAGE (RELATIVE), EXEC 310
WRITE EXECUTION REPORT MESSAGE (ABSOLUTE), EXCA 324
WRITE FROM BUFFER, WRITE 340
WRITE FROM BUFFER, WRITE 340
WRITE .MGEOT 340
COURTESY CALL FOR WRITE, CCCWR 354

.CALL
.CALL MACRO, HCL 36

.EXIT
.EXIT MACRO, HEX 38

.GOTO
.GOTO MACRO, HGT 40

.ICPIO
CARD PUNCH INITIALIZATION, .ICPIO 208
.ICPIO .MCPIO 208

.IDISP
INITIALIZATION, .IDISP 35
.IDISP .MDISP 35

.IDR20
MDS200 INITIALIZATION, .IDR20 230
.IDR20 .MDR20 230

.IDS20
DSU200 INITIALIZATION, .IDS20 254
.IDS20 .MDS20 254

.IFALT
FAULT PROCESSOR INITIALIZATION, .IFALT 91
.IFALT .MFALT 91

.IGPIO
CARD READER INITIALIZATION, .IGPIO 258
.IGPIO .MGPIO 258

.IIOS
MAIN IOS MODULE INITIALIZATION, .IIOS 201
.IIOS .MIOS 201

.IMTAP
MAGNETIC TAPE INITIALIZATION, .IMTAP 265
.IMTAP .MMTAP 265

.IPRIO
PRINTER INITIALIZATION, .IPRIO 270
.IPRIO .MPRIO 270

.IPTAP
PAPER TAPE INITIALIZATION, .IPTAP 273
.IPTAP .MPTAP 273

.ITYPE
TYPEWRITER INITIALIZATION, .ITYPE 280
.ITYPE .MTYPE 280

.MACTS
ACTS1 (EP1) .MACTS 281
ACTS2 (EP2) .MACTS 282
ACTS3 (EP3) .MACTS 289

.MCPIO
CPIT (EP1) .MCPIO 204
CPIO (EP2) .MCPIO 207
.ICPIO .MCPIO 208

.MDISP
DSP (EP1) .MDISP 3
GRD (EP2) .MDISP 17
FRLC (EP3) .MDISP 19
RLC (EP4) .MDISP 20
ENCC (EP5) .MDISP 21
ENB (EP6) .MDISP 22
DSPQH (EP7) .MDISP 24
DSPQT (EP8) .MDISP 25
DACNB (EP9) .MDISP 26
DRSTR (EP10) .MDISP 27
DSCNT (EP11) .MDISP 28
DMIOA (EP12) .MDISP 29
SCK (EP13) .MDISP 32
DSPQM (EP14) .MDISP 33
TRACF (EP15) .MDISP 34
.IDISP .MDISP 35
HCL .MDISP 36
HEX .MDISP 38
HGT .MDISP 40
TRACE .MDISP 49
OPGAT .MDISP 51
SHUG .MDISP 52

.MDR20
DRIT (EP1) .MDR20 209
DRIO (EP2) .MDR20 212
DRSL (EP3) .MDR20 214
DRGP (EP4) .MDR20 217
NEGATIVE EP .MDR20 222
.IDR20 .MDR20 230

.MDS20
DSIT (EP1) .MDS20 231
DSIO (EP2) .MDS20 233
DSSL (EP3) .MDS20 234
DSGP (EP4) .MDS20 236
NEGATIVE EP .MDS20 247
.IDS20 .MDS20 254

.MDUMP
CALL .MDUMP AT EP2, BOOT 89

.MFALT
FPRC .MFALT 55
FLT .MFALT 56
FSB (EP1) .MFALT 77
GET ADDRESS FOR FILE CODE, FGAD (.MFALT) 78
FGAD (EP2) .MFALT 78
FGCON (EP3) .MFALT 79
FGFIL (EP4) .MFALT 80
FGLAP (EP5) .MFALT 81
FGRET (EP6) .MFALT 82
FGSET (EP7) .MFALT 83
FGTIM (EP8) .MFALT 84
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFALT) 85
FGLP (EP9) .MFALT 85
FEMM (EP10) .MFALT 86
FNDE (EP11) .MFALT 87
DEALLOCATE MEMORY, GMRL (.MFALT) 88

.MFALT (continued)
GMRL (EP12) .MFALT 88
BOOT (EP13) .MFALT 89
.IFALT .MFALT 91

.MFLT1
GET ADDRESS FOR FILE CODE, FGAD (.MFLT1) 95
FGAD (EP1) .MFLT1 95
FCON (EP2) .MFLT1 96
FLSW (EP3) .MFLT1 100
DEALLOCATE MEMORY, GMRL (.MFLT1) 101
GMRL (EP4) .MFLT1 101
DIVIDE EXECUTION TIME INTO SECONDS, FGLP (.MFLT1) 105
FGLP (EP5) .MFLT1 105
FPRIO (EP6) .MFLT1 106
FLBAR (EP7) .MFLT1 107
FWAKE (EP8) .MFLT1 109
GMRLM (EP9) .MFLT1 110

.MGEOT
DISP .MGEOT 326
STACT .MGEOT 336
READ .MGEOT 338
WRITE .MGEOT 340
CCCRD .MGEOT 351
CCCWR .MGEOT 354
CCGSP .MGEOT 356
ASIGN .MGEOT 366
STRJB .MGEOT 368
NEWJB .MGEOT 369
FLUSH .MGEOT 373

.MGOU1
FOR (EP1) .MGOU1 375
AFT (EP2) .MGOU1 377
IDENT (EP3) .MGOU1 378

.MGOU3
START (EP1) .MGOU3 379

.MGPIO
CRIT (EP1) .MGPIO 255
CRIO (EP2) .MGPIO 257
.IGPIO .MGPIO 258

.MIOS
LINK (EP1) .MIOS 113
LINKF (EP2) .MIOS 116
LINKR (EP3) .MIOS 117
QUEUE (EP4) .MIOS 118
INOS (EP5) .MIOS 120
SPEC (EP6) .MIOS 133
ITYM (EP7) .MIOS 135
PTRVL .MIOS 138
DCWCK .MIOS 139
FNDFC .MIOS 141
FMTAR .MIOS 142
IOTRM .MIOS 144
UNLNK .MIOS 169
ILPCX .MIOS 170

.MIOS (continued)
STRET .MIOS 171
STIO .MIOS 175
STIOM .MIOS 187
TYPER .MIOS 188
STGPC .MIOS 189
RSMIO (EP8) .MIOS 190
ABTIO (EP9) .MIOS 192
GSTRT (EP10) .MIOS 196
RSMCH (EP12) .MIOS 197
ACTFL (EP13) .MIOS 198
.IIOS .MIOS 201

.MMTAP
MTIT (EP1) .MMTAP 259
MTIO (EP2) .MMTAP 264
.IMTAP .MMTAP 265

.MPRIO
PRIT (EP1) .MPRIO 266
PRIO (EP2) .MPRIO 269
.IPRIO .MPRIO 270

.MPTAP
PTIT (EP1) .MPTAP 271
PTIO (EP2) .MPTAP 272
.IPTAP .MPTAP 273

.MSYOT
MME (EP1) .MSYOT 293
EXEC (EP2) .MSYOT 310
EOA (EP3) .MSYOT 311
EOJ (EP4) .MSYOT 313
EXCA (EP5) .MSYOT 324
GEIN (EP6) .MSYOT 325

.MTYPE
TYIT (EP1) .MTYPE 274
TYIO (EP2) .MTYPE 277
TYSL (EP3) .MTYPE 279
.ITYPE .MTYPE 280

DOCUMENT REVIEW SHEET

TITLE: GE-625/635 GECOS-III Flowcharts

CPB #: 1500

FROM:

Name: _____

Position: _____

Address: _____

_____

Comments concerning this publication are solicited for use in improving future editions. Please provide any recommended additions, deletions, corrections, or other information you deem necessary for improving this manual. The following space is provided for your comments.

COMMENTS: _____

Please cut along this line

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
Fold on two lines shown on reverse
side, staple, and mail.

STAPLE STAPLE

FOLD

FIRST CLASS
PERMIT, No 4332
PHOENIX, ARIZONA

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

GENERAL ELECTRIC COMPANY
COMPUTER EQUIPMENT DEPARTMENT
13430 NORTH BLACK CANYON HIGHWAY
PHOENIX, ARIZONA - 85029

ATTENTION: DOCUMENTATION STANDARDS AND PUBLICATIONS B-90

FOLD

INFORMATION SYSTEMS

GENERAL ELECTRIC

LITHO U.S.A.