Bill Wilkins

DISTRIBUTION

Eastern Region

Paramus

Tom Strasser - Regional Data Products Sales Mgr.
Larry Lotito - Data Center Mgr.
Charles Chernack - Data Center Training Mgr.
Don Barkley - District Mgr.
George Tibaldi - Systems Analyst

Norwalk

Dave Denman - Systems Analyst

Lexington

Ted Doyle - District Mgr.
Steve Rosen - Systems Analyst

King of Prussia

Stroud Custer - Systems Analyst

Rockville

Ed Hayes - District Mgr.
Ken Kelley - Systems Analyst

Midwestern

Skokie

Ed Pulsifer - Regional Data Products Sales Mgr.
Lloyd Kusak - Data Center Mgr.
Thurman Gladden - Systems Analyst
Larry Mills - Systems Analyst
Bill Alexander - Systems Analyst
Stan Baker - Systems Analyst
Rick Walsh - Systems Analyst

Neely

No. Hollywood

Jim Schmidt - Regional Data Products Sales Mgr.
Jerry Allen - Systems Analyst
Tom Winker - Systems Analyst

Englewood

Geo. Taylor - Systems Analyst

Fullerton

Rick Connella - District Mgr.
Olen Morain - Data Center Mgr.
Diane Miller - Systems Analyst
Lee Kesky - Systems Analyst

Neely (Cont'd.)

Palo Alto

Bill Krause - District Mgr.
Frank Jackson - Systems Analyst
Larry Wear - Systems Analyst
Mike Leavell - Systems Analyst

Southern

Atlanta

Neil Fisk - Regional Data Products Sales Mgr.
Phil Carter - Data Center Mgr.
Dave McClellan - Systems Analyst
Frank Berman — " "

Richardson

Bill Swanson - Systems Analyst (D.P.)
Jack Olamar — " " (SYSTEMS)

Cupertino Division

Janet Baran
Dave Blecki
Bob Bond
Joe Dietzgen
~~Marc Cumucio~~
Dave Harris
George Moore
John Pavone
Dick Peake
Larry Walsh
Galen Wampler
Bill Williams

HEWLETT PACKARD

CUPERTINO DIVISION • 11000 Wolfe Road, Cupertino, California 95014, Telephone 408 257 7000

C. Ackerman

Distribution

October 19, 1970

Systems Analyst DOS-M Course in Paramus and Atlanta

This memo confirms the dates of December 8 - 10, 1970 and December 15 - 17, 1970 for the DOS-M Seminars to be held in Paramus and Atlanta respectively.

Bill Williams, who in conjunction with Joe Dietzgen is providing factory DOS-M support, will be presenting these seminars. Bill's work schedule at each Data Center will be as follows:

Monday, Dec. 7 - Paramus Monday, Dec.14 - Atlanta	Arrive at Data Center, taking care of last minute preparations as necessary for a Tuesday, 8:00 A.M. Seminar starting time.
Tues.-Thur. Dec.8-10 Par. Tues.-Thur. Dec.15-17 Atl.	Conduct the DOS-M Seminar Lecture 8:00 A.M. - 2:30 P.M. Lab 2:30 P.M. - 8:00 P.M. where a DOS-M system would be available.
Fri. Dec. 11 - Paramus Fri. Dec. 18 - Atlanta	Remain at Data Center providing additional system training & assistance as requested by regional personnel.

I am sure you will find these seminars the answer to many of your DOS-M system questions.

Chuck

DOS-M SEMINAR - PARAMUS: December 8-10, 1970 ATLANTA: December 15 - 17, 1970

TIME	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURD.
8:00	Arrive at Data Making Necessary Preparations.	Introduction - Hardware Req. - Adv./Disadv. - IOMEC - System Software	Installation (Cond.) - Starting Gen. - Formatting Disc - DOS-M Bootstrap	Internal Sys.Oper. - Bootstrap - I/O - Drivers	Provide Additional training & assistance as requested	
9:00						
		COFFEE				
10:00		DOS/DOS-M DIFF. - Starting Sys. - New Directives - New Exec. Calls - LOADR - Libraries - I/O		- On-line Mods & patches - System bugs		
11:00						
12:00						
1:00		Installation - Generation - DSGEN - Prelim. Consid.	Internal Sys. Org. - Overview Core Alloc. Disc Alloc.			
2:00						
		COFFEE				
3:00		DOS-M Lab	DOS-M Lab	DOS-M Lab		
4:00						
5:00						
6:00						

SYSTEMS ANALYST DOS-M COURSE OUTLINE

I INTRODUCTION

A. Minimum Hardware Requirements
   1. DOS and DOS-M
   2. Cost Comparison
      a. DOS to DOS-M
      b. DOS (HP) to Competitor (1130)

B. Advantages and Disadvantages

C. IOMEC Disc Overall Description
   1. Controller
      a. Connection to Drives
      b. Disc Protect Override Switch
   2. Drive
      a. Fixed Disc
      b. Cartridge (Removable Pack)
      c. Power ON/OFF and READY
      d. Cartridge LOCK/UNLOCK
      e. Maximum of Four Per Controller
   3. Discs
      a. Physical Layout (Heads Etc.)
      b. Addressing (Subchannel, Cylinder, Track, Sector)
      c. Sector Address Field
      d. Storage Capacity Breakdown
      e. Difference Between Physical and Logical Track
      f. Programming Commands
      g. Diagnostic Program

D. System Software
   1. Components
   2. Minimum Core 16K For ALGOL
   3. Disc Space Used by System
   4. Concept of SYSTEM and USER Discs
      a. Labels, Sys. Gen. Code, Sys. Proprietary Code
      b. Allocation (System, Work, JBIN, "USER")

II OPERATIONAL DIFFERENCES FROM DOS

A. Starting System
   1. Bootstrap Rather Than BBDL
   2. No "FRESH", "CONTINUATION" Requirement

B. New Directives
   1. Change User Disc (:UD)
   2. Disc to Disc Dump (:DD)
   3. System Search (:SY)
   4. Initialize (:IN)
   5. Abort without Job Termination (:OFF)

C. Minor Changes to Some DOS Directives

III PROGRAMMING DIFFERENCES FROM DOS

A. New EXEC Call (change user disc)
B. Minor Changes to Some DOS EXEC Calls
C. Use of LOADR
D. Libraries
C. I/O

IV INSTALLATION

A. Introduction to System Generation
   1. Tapes Needed & Program Names
   2. Preliminary Consideration
      a. Core Size
      b. Speed of System Operation Desired
      c. System and User Disc Placement
      d. Particular Needs For Application Used
      e. Whether Another System is Around
      f. Medium For Input of System Modules
      g. Hardware Required
   3. Preparation of DSGEN (SIO Environment)
   4. Starting Generator
      a. All Equipment on, Disc Drive "ON" and "READY"
      b. Disc Protect Override "ON"
      c. Load DSGEN to Memory and Start at 1ØØ Octal
      d. Description of the Four PHASES and Ability to Restart Each in Case of Error

B. System Generation Procedure (and EXAMPLE)
   1. Initialization Phase
   2. Program Input Phase
   3. Parameter Input Phase
   4. Disc Loading Phase
      a. Parallel Memory Block Diagram
      b. Parallel Memory Octal Dump

C. Formatting User Discs or Cartridges
   1. Why and When Necessary
   2. What System Actually Does
   3. Operation Procedure

D. Configuring DOS-M Bootstrap
   1. Function Performed
   2. Clear up Confusion Between
      a. Prepare Configured Bootstrap
      b. Configured Bootstrap
      c. Disc Resident Bootstrap
   3. Hardware Required
   4. Operation Procedure

V INTERNAL SYSTEM ORGANIZATION

A. Brief Review of Core Layout
   1. Block Diagram
   2. Core Dump (Octal)

B. Disc Layout
   1. Block Diagram
   2. Disc Dump (Octal)

C. DOS-M Drivers

VI INTERNAL SYSTEM OPERATION

A. Prepare Configured Bootstrap Program Study
   1. Listing
   2. Execution of Configured Bootstrap

B. Disc Resident Bootstrap Study
   1. Listing
   2. Execution

C. Trace I/O Operation Through Flowcharts Involving the Following Programs:
   1. Disc Monitor (DISCM)
   2. Input Output Control ($EX18)
   3. Example Driver (DVRØ1)

D. On-Line System Modification and Patches
   1. In Case Line Printer Goes Down
   2. Memory Protect Considerations

E. Known System Bugs in Version Being Shipped
   1. Symptoms
   2. Patches

OCTAL TO DECIMAL CONVERSION

TEN-THOUSD.		THOUSAND		HUND.		TEN		UNIT	
OCT.	DEC.	OCT.	DEC.	OCT.	DEC.	OCT.	DEC.	OCT.	DEC.
0	0	0	0	0	0	0	0	0	0
1	4,096	1	512	1	64	1	8	1	1
2	8,192	2	1,024	2	128	2	16	2	2
3	12,288	3	1,536	3	192	3	24	3	3
4	16,384	4	2,048	4	256	4	32	4	4
5	20,480	5	2,560	5	320	5	40	5	5
6	24,576	6	3,072	6	384	6	48	6	6
7	28,672	7	3584	7	448	7	56	7	7

"A" VERSION IOMEC

SYSTEM ANALYST DOS-M COURSE

DECEMBER 1969

or

JAN. 1970

SYSTEM ANALYST DOS-M COURSE OUTLINE

I. INTRODUCTION
- A. Minimum Hardware Requirements
- B. Advantages and Disadvantages (Compared to DOS)
- C. IOMEC Overall Description
- D. DOS-M Software
- E. System Startup Description
- F. DOS-M I/O Request Processing

II. OPERATIONAL DIFFERENCES FROM DOS
- A. System Startup
- B. New Directives
- C. Operational Difference Summary

III. PROGRAMMING DIFFERENCES FROM DOS
- A. New EXEC Call
- B. Negative Request Codes
- C. EXEC Calls Difference Summary
- D. Other Important Points

IV. INSTALLATION
- A. Introduction to Generation
- B. System Generation Procedure and Example
- C. Formatting User Discs and Cartridges

V. INTERNAL SYSTEM ORGANIZATION
- A. Disc File(s) Format
- B. Disc Dump of Generation Example
- D. System Base Page Communication Area Description

VI. INTERNAL SYSTEM OPERATION
- A. Iomec Command Sequences
- B. Supplied DOS-M Bootstrap
- C. Disc Resident Bootstrap
- D. DOS-M System HALTS
- E. I/O Request Processing Example

VII. DOS-M FLOWCHARTS

I. INTRODUCTION

A. Minimum Hardware Requirements

1. Why DOSM?
2. DOS/DOSM minimum hardware [SLIDE 1]
3. Cost Comparison
4. Comparison to competitor - (IBM 1130) [SLIDE 1A]
5. DOSM Hardware options [SLIDE 1B] 12K with C.I Comp.

B. Advantages and Disadvantages [SLIDE 2]

1. ADVANTAGES (Special Points)

   a. Another cabinet and power supply needed when number of drives is expanded to 3 or 4.

   b. User could operate in his own instrument driver environment if no MP (Memory Protect).

   c. If User does not want EAU or clock, he is not forced to have it. Gains one more I/O channel without TBG option.

   d. Easy creation of System Backup (which will not be hardware protected) on Cartridge.

   e. Multiple System Discs with different configurations on separate drives.

   f. Exchange of user files between systems even though systems may be configured differently.

   g. Hardware protection scheme using DISC PROTECT OVERRIDE SWITCH and PCI (Protected Cylinder Indicator).

   h. Operation with USER DISCS Labeled to avoid using incorrect cartridge.

   i. Minimum core resident system reduced from DOS (DVRØ5 and DVR31 changes).

2. DISADVANTAGES (Special Points)

a. Three bootstraps (cover details later). Method of System Start-up.

b. No plans at present for moving head RTE or TSB. Some talk for ISS system (ISS disc cost about $ 35,000 and has about 12 million word storage).

c. Better to lose sale rather than deliver an 8K system that will "strangle" customer's programs (during loading or execution). Just because JOBPR will fit is not any indication that system will be adequate.

C. IOMEC Overall Description

1. CONTROLLER [SLIDE 3]

a. Interface between computer and drive(s).

b. Interface cards on computer side.

(1) Identical electronically except for positions of jumper wires.

(2) Signals inverted from positive-true/ ground-false logic to ground-true/ positive-false to be compatable with controller.

(3) DATA CHANNEL - Transfers data, status, and addressing information. DMA controls data; program controls status and addressing.

(4) COMMAND CHANNEL - Transfers commands, drive selection, and drive attention bits (LSB). All under program control.

2. DRIVE (Maximum of 4 per controller)
   a. Fixed Disc and Removable Cartridge.
   b. Movable heads and their numbers.
   c. Power ON/OFF.
   d. Cartridge LOCK/UNLOCK and light.
   e. READY light.
   f. Physical description of PACK.
      (1) Opening at rear for heads entry.

      (2) Opening underneath for forced air.

      (3) Rim markers for TRACK/SECTOR origins.

3. DISC
   a. Physical storage capacity breakdowns [SLIDE 4].
   b. Physical layout [SLIDE 5].
   c. Addressing [SLIDE 6].
      (1) Physical (Drive, Cylinder, Head, Sector).
      (2) Logical (Subchannel, Track, Sector).
          Software inverts (complements)lower order bit of subchannel # for higher order bit of head #.
   d. Sector Address and Data Fields [SLIDE 7].

      (1) INITIALIZE WRITE COMMAND is used for controller to construct and write the Sector Address Field with PCI=DCI=Ø (only disc formatting seciton of generator does this).
      (2) WRITE PROTECTED COMMAND is used for controller to set PCI=1 in Sector address field (only generator does this).
      (3) WRITE DEFECTIVE COMMAND is used for controller to set DCI=1 in Sector Address Field Only $EX2Ø makes this call (DVR31 will accept under system operation).

(Continued)

(4) ~~Controller does not send back any status on PCI or DCI if~~ DISC PROTECT OVERRIDE ~~SWITCH is "ON" (i.e. UP)~~. This switch must be "ON" when executing (1), (2), or (3) on the preceding page!

D. DOSM Software and Relationship to Discs

1. Components [SLIDES 8A, 8B, 8C, 8D].
2. Disc to Memory Transfers [SLIDE 9].
3. DOSM General Core Layout [SLIDE 10A].
   a. EQT Format [SLIDE 10B].
   b. DRT Format [SLIDE 10C].
   c. INT Table Format [SLIDE 10D]
4. Discs Layout
   a. Concept of "SYSTEM" and "USER" Discs.
   b. Oversimplified "SYSTEM " disc layout [SLIDE 11A
   c. More detailed "SYSTEM" disc layout [SLIDE 11B].
   d. User disc layout [SLIDE 11C].
   e. Label Sectors
      (1) System Disc [SLIDE 12A].
      (2) User Disc [SLIDE 12B].

E. System Startup Operation Example with block diagrams.

1. Execution of Configured Supplied Bootstrap [SLIDE 13A.1].
   a. Loads Disc Resident Bootstrap into high core, relocating it as necessary.
   b. Transfers control to start of DRB just loaded.

E. SYSTEM Startup Operation Example with block diagrams.

2. Execution of Disc Resident Bootstrap [SLIDE 13A.2].

   a. Loads Core Resident System from System Disc in four parts.

   b. Configures continuator section of DVRØ5 and DVR31.

   c. Transfers control to $STRT in DISC MONITOR by JMP 3,I.

3. Disc Monitor First Entry [SLIDE 13A.3].

   a. $STRT calls $MDLD to transfer control to $EXI2 (System Startup).

   b. $MDLD makes decision whether to load $EXI2 from Disc (if Disc Resident) then transfers control to $EXI2.

4. Execution of $EXI2 [SLIDE 13A.4].

   a. Sets (MP FENCE ADDRESS = UMFWA) with OTA 5.

   b. Reads System Buffer Sector to Base Page.

   c. Builds new System Buffer Sector and writes back on disc if not valid one (i.e., does not end with "SB").

   d. Calls $SYIO to output "INPUT :DATE,XXXXXX,H,M" on System TTY.

   e. Sets input request code = "DA" and transfers control to $TYPE for System K.B. Input.

5. Execution of $TYPE for System K.B. Input [SLIDE 13A.5].

   a. Calls $SYIO to output (CR) (LF) @.

   b. Calls $SYIO to input 72 characters into JOB INPUT BUFFER.

   c. $SYIO calls $TEST routine to force :DA input.

   d. If :DA not inputted, calls $SYIO to output "IGNORED" and goes to b. above.

   e. If :DA inputted, transfers control to $JLOD to load and branch to JOBPR.

6. Execution of $JLOD to load JOBPR [SLIDE 13A.6].

   a. $JLOD calls DISCX twice to read in JOBPR (MAIN and Base Page) and then transfers control to JOBPR main entry point.

   b. Each call to DISCX results in call to $DISC which in turn calls $SYIO to read DISC.

7. Execution of JOBPR to Update System Buffer Sector [SLIDE 13A.7].

   a. Date routine in JOBPR reads System Buffer Sector from Disc to its own internal buffer by JSB EXEC call.

   b. Updates DATE, LU TABLE entries, and Default User Label in System Buffer Sector.

   c. Writes updated System Buffer Sector back on Disc by JSB EXEC call.

8. Execution of JOBPR to report Default USER DISC SUBCHANNEL # and LABEL [SLIDE 13A.8].

   a. Date Routine continues by executing EXEC call to request CURRENT USER DISC SUBCHANNEL # and LABEL.

   b. $EXI7 is used just as if :UD directive had been entered.

c. Point out that if GENERATION CODE or PROPRIETARY CODE do not agree with System Disc, ERROR MESSAGE may be printed here.

d. At end JOBPR then calls $TYPE for System TTY to output (CR) (LF) and @ and to input from Keyboard.

9. Summary of 1-8 above.

a. JOBPR is a USER PROGRAM and must do all I/O by JSB EXEC.

b. At end of 8 above, the JOBPR remains in memory until :PROG,X entered or :OFF given in response to * (here $CLER loads JOBPR fresh).

c. No part of Core Resident System ever does any I/O by JSB EXEC; always does by $SYIO.

d. MP and Interrupt System "ON" when in USER AREA. MP and Interrupt System "OFF" when in System Area.

f. DOSM I/O REQUEST PROCESSING [LARGE CHART].

DOS/DOSM MINIMUM HARDWARE COST COMPARISON

DOS

PROVIDES MINIMUM HARDWARE CONFIGURATION CONSISTING OF:

		PRICE 115V 60Hz	PRICE 230V 50Hz
1.	2116B Computer with 8192 Word Memory	$20,000	$20,000
2.	Direct Memory Access, Accessory No. 12578A	3,000	3,000
3.	Extended Arithmetic Unit, Accessory No 12579A	3,000	3,000
4.	Memory Parity Check, Accessory No. 12591A	1,000	1,000
5.	Time Base Generator, Accessory Kit No. 12539A	1,000	1,000
6.	Memory Protect, Accessory Kit No. 12581A	2,000	2,000
7.	Teleprinter Input/Output consisting of:		
	HP 2752A Teleprinter (Modified Teletype ASR-33) with	1,250	1,450
	HP 12531B Teleprinter Input/Output Interface Kit	750	750
8.	Teleprinter Input/Output, consisting of:		
	HP 2754B Heavy-Duty Teleprinter (modified Teletype ASR-35) with	3,850	4,250
	HP 12531B Teleprinter Input/Output Interface Kit	750	750
9.	Disc Memory consisting of:		
	HP 2770A Disc Memory (368,640 words non-expandable)	17,000	17,200
	HP 2772A Disc Memory Power Supply	2,500	2,700
	HP 12606A Disc Memory Interface Kit	4,000	4,000
10.	2886A Single-Bay Cabinet	900	900
	TOTAL COST	$61,000	$62,000

(Four-Year Lease @ $1,690/month)

DOSM

PROVIDES MINIMUM HARDWARE CONFIGURATION CONSISTING OF:

		PRICE 115V 60 Hz	PRICE 230V 50 Hz
1.	2114B Computer with 8K memory	$13,000	$13,100
2.	Direct Memory Access, Accessory Kit No. 12607A	1,500	1,500
3.	Memory Parity Check with Interrupt, Accessory Kit No. 12598A	1,000	1,000
4.	System Console, consisting of:		
	HP 2752A Teleprinter (Modified Teletype ASR-33) with	1,250	1,450
	HP 12531B Teleprinter Input/Output Interface Kit	750	750
5.	System Input, consisting of:		
	HP 2748A Punched Tape Reader with	1,500	1,600
	HP 12597A-002 Punched Tape Interface Kit	600	600
6.	Cartridge Disc Memory System, consisting of:		
	HP 2870A Disc Drive (includes HP 12536A Disc Cartridge)		
	HP 2871A Disc Controller	13,500	13,650
	HP 2881A Power Supply		
	HP 2882A Cabinet		
	HP 12557A Disc Interface Kit	2,500	2,500
	TOTAL COST	$35,600	$36,150

(Five-Year Lease @ $765/Month)

HP DOS-M/IBM 1130 COST COMPARISON

HP DOS-M		IBM 1130	
MINIMUM SYSTEM (WITH PAPER TAPE)	PURCHASE	MINIMUM SYSTEM (WITH PAPER TAPE)	PURCHASE
2114B (2.0 MICROSEC)	$ 8,500	1131-2A (3.6 MICROSEC)	$ 34,610
OPTION 4 (8K MEMORY TOTAL)	4,500	TTY/PRINTER CONSOLE	
12591A MEMORY PARITY CHECK	1,000	4K CORE, 500K DISC	
12067A DIRECT MEMORY ACCESS	1,500	1134 PAPER TAPE READER	1,270
2870A CARTRIDGE DISC DRIVE	8,700	(60 CHARACTERS/SEC)	
2871A DISC CONTROLLER	2,800	3623 P.T. READER ATTACHMENT	450
12557A DISC INTERFACE	2,500	1055 PAPER TAPE PUNCH	900
2882A DISC CABINET	600	(14.8 CHARACTER/SEC)	
2881A DISC POWER SUPPLY	1,400	7923 P.T. PUNCH ATTACHMENT	900
2752A TELEPRINTER ASR-33	1,250		$ 38,130
12531B TELEPRINTER INTERFACE	750	(WITH CARD I/O)	
2784A PAPER TAPE READER	1,500	1131-2A AS SHOWN ABOVE	34,610
(500 CHARACTERS/SEC)		1442 CARD READER/PUNCH	12,750
12597A P.T. READER INTERFACE	600	(160 COLUMNS/SEC)	
		4419 CARD READER/PUNCH ATTACHMENT	1,525
		3630 1442 INTERFACE	225
	$ 35,600		$ 49,110
TYPICAL SYSTEM	PURCHASE	TYPICAL SYSTEM	PURCHASE
2114B (2.0 MICROSEC)	$ 8,500	1131-3B (2.2 MICROSEC)	$ 58,050
OPTION 4 (8K MEMORY TOTAL)	4,500	TTY/PRINTER CONSOLE	
12591 MEMORY PARITY CHECK	1,000	8K CORE, 500K DISC	
2870A CARTRIDGE DISC DRIVE	8,700	1134 PAPER TAPE READER	1,270
2871A DISC CONTROLLER	2,800	(60 CHAR/SEC)	
12557A DISC INTERFACE	2,500	3623 P.T. READER ATTACHMENT	450
2882A DISC CABINET	600		
2881A DISC POWER SUPPLY	1,400		
2752A TELEPRINTER ASR-33	1,250	1055 PAPER TAPE PUNCH	900
12531B TELEPRINTER INTERFACE	750	(14.8 CHAR/SEC)	
2748A PAPER TAPE READER	1,500	7923 P.T. PUNCH ATTACHMENT	900
(500 CHAR/SEC)		1132 LINE PRINTER	11,350
12597A P.T. READER INTERFACE	600	82 LPM ALPHAMERIC	
2753A PAPER TAPE PUNCH	3,300	110 LPM NUMERIC	
(120 CHAR/SEC)			
12597A P.T. PUNCH INTERFACE	600	2310 DISC DRIVE	12,150
2767A LINE PRINTER	10,000	500K WORDS	
(80 COL, 356-1110 LMP)			
12653A LINE PRINTER INTERFACE	2,500		
	$ 52,000		$ 85,070

HP DOSM / IBM 1130 COST COMPARISON

HP DOS-M		IBM 1130	
MINIMUM SYSTEM (with paper tape)	**PURCHASE**	**MINIMUM SYSTEM (with paper tape)**	**PURCHASE**
2114B (2.0 microsec)	$8,500	1131-2A (3.6 microsec) TTY/Printer Console 4K core, 500K disc	$34,610
Option 4 (8K memory total)	4,500		
12591A Memory Parity Check	1,000	1134 Paper Tape Reader (60 characters/sec)	1,270
12067A Direct Memory Access	1,500	3623 P.T. Reader Attachment	450
2870A Cartridge Disc Drive	8,700	1055 Paper Tape Punch (14.8 character/sec)	900
2871A Disc Controller	2,800		
12557A Disc Interface	2,500	7923 P.T. Punch Attachment	900
2882A Disc Cabinet	600		
2881A Disc Power Supply	1,400		$38,130
2752A Teleprinter ASR-33	1,250	**(with card I/O)**	
12531B Teleprinter Interface	750	1131-2A as shown above	$34,610
2748A Paper Tape Reader (500 characters /sec)	1,500	1442 Card Reader/Punch (160 columns/sec)	12,750
		4419 Card Reader/Punch Attachment	1,525
12597A P.T. Reader Interface	600	3630 1442 Interface	225
	$35,600		$49,110

TYPICAL SYSTEM	PURCHASE	TYPICAL SYSTEM	PURCHASE
2114B (2.0 microsec)	$8,500	1131-3B (2.2 microsec) TTY/Printer Console 8K core, 500K disc	$58,050
Option 4 (8K memory total)	4,500		
12591A Memory Parity Check	1,000	1134 Paper Tape Reader (60 char/sec)	1,270
12067A Direct Memory Access	1,500		
2870A Cartridge Disc Drive	8,700	3623 P.T. Reader Attachment	450
2871A Disc Controller	2,800		
12557A Disc Interface	2,500	1055 Paper Tape Punch (14.8 char/sec)	900
2882A Disc Cabinet	600		
2881A Disc Power Supply	1,400	7923 P.T. Punch Attachment	900
2752A Teleprinter ASR-33	1,250		
12531B Teleprinter Interface	750	1132 Line Printer 82 LPM Alphameric 110 LPM Numeric	11,350
2748A Paper Tape Reader (500 char/sec)	1,500		
12597A P.T. Reader Interface	600	2310 Disc Drive 500K words	12,150
2753A Paper Tape Punch (120 char/sec)	3,300		
12597A P.T. Punch Interface	600		
2767A Line Printer (80 col. 356-1110 lpm)	10,000		
12653A Line Printer Interface	2,500		
	$52,000		$85,070

[SLIDE 4A]

FROM WALL STREET JOURNAL OCT. 29, 1970

IBM Enters the Mini-Computer Market As It Unveils 2 New Models, Its Cheapest

By a WALL STREET JOURNAL *Staff Reporter*

NEW YORK — International Business Machines Corp. introduced two small computers for office and industrial applications.

The two are the least expensive that the world's largest computer-maker has offered, and one of them, at least, puts IBM into competition for the first time with makers of what have come to be called mini-computers.

Digital Equipment Corp. of Maynard, Mass., has been the dominant manufacturer of mini-computers, which generally are priced under $20,000 and as low as $5,000. Such computers have been used largely for calculations by scientists and engineers, but they are being applied increasingly to industrial processes and some small-business uses. Other leading manufacturers include Honeywell Inc., Varian Associates Inc., Hewlett-Packard Co. and Data General Corp.

Trading in Digital Equipment's stock on the American Stock Exchange yesterday reacted sharply to IBM's announcement. Digital Equipment closed at $61.875, down 37½ cents a share. During the day it traded as low as $57.50. IBM, traded on the New York Stock Exchange, closed at $296.50, up $5.50 a share.

The new IBM model that can be considered a mini-computer is the System 7, designed specifically to monitor and control industrial and laboratory processes. It may be purchased for a minimum of $16,060 or rented for $352 a month and up. First deliveries to customers are scheduled for November 1971.

In Maynard, a Digital Equipment spokesman said the System 7 appeared to be priced too high to be a mini-computer by his company's standard. He added that Digital Equipment products had competed very well with IBM's previous process-control model and that "we think we will continue to do well."

The System 7, operating unattended, can measure, test, analyze or control processes in petroleum and chemical plants, electric-utility substations, steel mills and laboratories and in a variety of other industries. It can take as many as 250,000 readings a second from instruments, analyze them, and, if desired, forward the data to a larger central computer.

As do several mini-computers already on the market, the System 7 has a main memory made of integrated electronic circuits, rather than of the usual magnetic cores.

The System 7 isn't designed for business data processing or for direct use by an engineer or scientist, IBM said. However, F. G. Rodgers, president of IBM's Data Processing division, was asked if the System 7 would be available for purchase as a component by other assemblers of control systems, who currently represent a large share of the market for mini-computers. "We'll be delighted to sell this to anybody," Mr. Rodgers replied.

IBM's other new computer is the System 3 Model 6, an extension of the System 3 small-scale computer, since designated the Model 10, that was introduced in 1969. System 3 computers are aimed for the most part at businesses that haven't previously used computers. IBM has said that more than 1,000 System 3 computers have been delivered this year.

Designed for business data processing, a typical Model 6 can be purchased for $48,250 or rented for $1,015 a month. First deliveries of it, in contrast to those of the System 7, are scheduled within 60 days. IBM demonstrated the first production units of the Model 6 in more than 40 cities yesterday.

The Model 6 can be used for accounting functions, including the processing of standard ledger cards, and also for engineering calculations in a "conversational" mode, IBM said. It stores data on magnetic disks and can communicate with other IBM computers.

The two new computers aren't part of the System 370 family of medium and large, general-purpose computers announced by IBM in June. Both small computers are being produced at a plant in Boca Raton, Fla. The System 3 Model 6 also is in production in Vimercate, Italy.

MODEL 6
SYSTEM 3 (small business)
(used Discs)
Typical $48,250

SYSTEM 7 for mainly scientific applications
$16,060

weekly Newspaper – Second-class postage paid at Chicago, Illinois

November 4, 1970 Price: $9/ye

See Page 4 on DAS

Keyboard Oriented

System/3 Model 6 Includes Disk, Ledger Card, Basic Language

Operator's Console and Memory Cabinet

Model 6 Ledger Card Unit

By Frank Piasta
CW Staff Writer

WHITE PLAINS, N.Y. – The newest version of the IBM System/3, the Model 6, is designed to appeal to a variety of users:

- The business-oriented user who has been doing his data processing on bookkeeping machines and desk calculators.
- The scientific-oriented problem solvers, such as engineers and scientists, who can use the system to perform computations that would not justify a larger machine.
- The large computer installation that needs programmable terminals that can tolerate the restricted I/O facilities of the Model 6.

Priced at about $1,000/mo for a typical configuration, the small system will be available for first deliveries in "less than 60 days," IBM said.

Restricted Card-Handling

One group that will not find its requirements met are the card-oriented users. The restricted card handling capability of the new system will force these users to look to the original System/3, now called the Model 10, to fulfill their needs.

Card handling equipment is limited to the System/3 Data Recorder, which can read, punch and print 96-column cards at 22 card/min, on-line as an optional feature.

IBM has joined GE, Burroughs Honeywell, and others in supplying a keyboard-oriented computer with disk capability.

The system can process ledge cards similar to those used in bookkeeping machine applications, using the serial printer as its principal output device.

The software packages supplied with the Model 6, RPG II and Basic compilers, are used in conjunction with the keyboard to achieve an interactive mode of operation, with the operator keying in the data. Fortran, IBM said, will not be available.

A typical System/3 Model 6 keyboard entry configuration would consist of a central processing unit with an 8K byte main memory, a 2.45 million character disk file and an 85-char/sec serial printer.

Core capacities can range from 8K to 16K bytes. Memory cycle time is 1.52 μsec. Add time is 26 μsec for two 5-digit numbers.

The primary method of entering information into the Model 6 is the console keyboard, an integral part of the CPU. The keyboard is similar to that of a typewriter. A standard 10-key adding machine cluster is used for numerical data entry.

Model 6 Peripherals

Printers available with System/3 Model 6 include three 13-in.-wide models (132-print positions) available for conventional printing and two 22-in.-wide models (220-print posi-

(Continued on Page 4)

IBM Adds Small Process Control Unit, System/7

WHITE PLAINS, N.Y. – System/7, a low-cost process control computer announced by IBM, provides sensor-based data processing related to either relay or solid-state multiplexer points. The company said the smallest configuration of the System/7 would rent for $352/mo or sell for $16,000.

The system can be used as part of a network or as a stand-alone unit. The type and number of sensing points is determined by the configuration of input-output modules attached to the central processor, IBM said.

The system's all-monolithic memory can be expanded at the user's site, IBM said.

A "host" computer attachment allows System/7 to operate with an IBM 1130, 1800, System/370 or models 25 and up of the System/360 for additional I/O and computational capability.

(Continued on Page 4)

rd of Hospitals Reported ied With Vendor Support

on re-
hospi-
ey has
ird of
ate the
n their

tals al-
us de-
with

ospit
ssoc
ate of
in the

e

Health Care Industry," was compiled by Harris, Kerr, Chervenak, and Co.

The survey is based on questionnaire and interview responses from 2,800 of the 7,137 members of the American Hospital Association in the 48 contiguous states.

The survey revealed many hospitals where costs were clearly out of proportion to results and suggested: "Too often the hospital seems to have wandered into data processing with insufficient expertise; without first defining its requirements fully and realistically, and without establishing its short- and long-range objectives."

Of the respondents, 503 had installed computers, 75 had computers on order, and 1,106 an integrated manner.

None of the computer manufacturers has provided such packages to cover all – or even most – of the major hospital reporting areas. All of them con-

(Continued on Page 4)

Hearings to Probe Status

Are Programmers Professionals?

By Edward J. Bride
CW Staff Writer

WASHINGTON, D.C. – Computer programmers who insist they should be considered "professionals" will have their day in court soon, when the Labor Department opens two-pronged hearings to define the term "professional."

An offspring of achieving such official status dustry," and have been trying to achieve that status "through their various societies like DPMA and ACM."

"I think we would want to upgrade them in that area, too," Dreyer commented.

"Problem Is Inconsistency"

Adapso will appear at the December hearings.

3/6 Appeals to Different Users

(Continued from Page 1)

tions) that accommodate ledger cards.

Model 6 printers use a serial wire matrix printhead with a speed of 85 char/sec. One 13-in. and one 22-in. printer offer bi-directional capability. As one line is completed with the printhead moving from left to right, the next line is printed on the return, right to left.

The 22-in. printers equipped with ledger card device allow a single card to be positioned, posted and stacked in about four seconds.

A Data Recorder can be attached to the Model 6 for batch input of programs and data using the 96-column punched card. It can be used both on-line and off-line. During the on-line use, the data recorder can operate at 22 card/min – reading, punching, or punching and printing.

Ledger cards can be handled in sizes up to 11 in. by 14 in., and can be used on both sides. The ledger card stacker can accommodate up to 100 cards.

An additional output device, an optional IBM 2265 CRT display unit, can be be used to provide interaction with the system and access to stored information.

To provide additional functional flexibility, particularly for small banks, an optional Serial Input/Output Channel (Sioc) is offered for System/3 Model 6. The Sioc allows attachment of input-output units such as the IBM 1255 magnetic character reader.

System/3 Model 6 can be used as a terminal/processor. A Binary Synchronous Communications Adapter (BSCA) is available that allows Model 6 users to exchange data over switched, leased or private lines with remotely located System/3s, and with 360 and 370 models.

A typical System/3 Model 6 will rent for $1,015/mo, or may be purchased for $48,250.

Model / Features	GE 58	H 115	Singer System 10	Burroughs L 4000	IBM Sys Mod. 10	IBM Sys./3 Model 6
Main Mem. (K Words)	5 to 10 (bytes)	16 to 32 (6-bit)	10 to 110 (bytes)	1,024 (8-byte) (fixed disk)	8 to 32 (bytes)	8 to 16 (bytes)
Cycle Time (μsec)	1.2	2.75	3.3	5 msec (av. access time)	1.52	1.52
Auxiliary Random Access Storage (millions)	11.5 (bytes)	36.8 (char)	10 to 100 (char) (bytes)	No	2.5 to 9.8 (bytes)	2 to 9.8 (bytes)
Ledger Card Handling	No	No	Yes	Yes	No	Yes
Languages	Gesal, Gecol	Easy-coder, Cobol, Fortran	Assembler, Fortran, Simulator	Cobol (sub-set), Assembler	RPG II	RPG II Basic
Rental ($K)	0.9 to 2.2	1.9 to 6.8	1.4 to 6.5	0.4 to 0.6	0.9 to 4.0	0.9 to 2.7
First Delivery	10/70	6/70	9/70	7/70	1/70	12/70

Chart compares new System/3 Model 6 with other small business computers.

Basic as Well as RPG II Offered With Model 6

By Don Leavitt
CW Staff Writer

WHITE PLAINS, N.Y. – The Model 6 user has a choice of programming languages. RPG II is available for commercial data processing and Basic can be used for problem-solving.

The user is not faced with an "either/or" choice: both language processors can reside on the same disk cartridge.

RPG II, the only language that was available on the earlier S/3, Model 10, has been modified to make use of the hardware capabilities of the Model 6.

The Basic compiler includes software paging techniques which allow the Model 6 to operate as if it had 64K bytes in its main memory.

According to IBM, this paging allows programs containing up to 990 statements to run on a Model 6 with only 8K bytes of main memory.

Model 10 Batch-Oriented

The S/3 Basic, accessible to only one user at a time, is a version of Call/360 Basic. Basic for the Model 6 is not adaptable to the Model 10, since that is essentially a batch-oriented machine, IBM noted.

RPG II handles the new I/O features, including ledger card processing and the display station output.

The Basic compiler will be licensed at $110/mo, available with first customer shipments in December. The RPG II compiler for the Model 6 will be licensed at $35/mo, and will be available

COMPUTERWORLD
THE NEWSWEEKLY FOR THE COMPUTER COMMUNITY
TM Reg. U.S. Pat. Off.

Process Control Unit Added to IBM Line

(Continued from Page 1)

IBM System/7 includes a 16-bit processor module, I/O modules, and an operator station. The processor module and from one to 11 input-output modules are housed in enclosures that also provide internal power and signal distribution.

The processor module features 2,048 to 16,384 words of monolithic storage in 2,048-word increments.

Although the System/7 has a 400 nsec cycle time and its input modules can scan 128 relay or solid-state multiplexer points, it is limited to four priority interrupt levels.

By contrast, IBM's 1800 process control system has 12 levels of interrupt but operates, depending on the model, at speeds of only 2 μsec or 4 μsec.

There are two types of input-output modules available with System/7, the IBM 5014 analog input module and the 5012 multifunction module. In each, the external sensor wiring connects by plug-in termination cards.

The operator station consists of a typewriter-like keyboard and request key, printer, paper tape reader, and paper tape punch. Operation is under program control. Switches permit the system to be used as a source program preparation device, the company said.

Programming for users who do not have access to a host computer is accomplished through the IBM System/7 Assembler.

The IBM 2790 data communication system devices supported by System/7 include the 2791 and 2793 area stations, 2795 and 2796 data entry units, 1053 printers and 1035 badge readers.

Monthly rental is $352 for the smallest IBM System/7, including the processor, 2,048 words of monolithic memory and one I/O module. The purchase price is $16,060. Additional memory increments of 2,048 words rent for $105.

The required IBM 5028 operator station is available, purchase only, at $2,240.

DOSM HARDWARE OPTIONS

1. ADDITIONAL MEMORY
   16,384 OR 32,768 WORDS ON 2116B

2. ADDITIONAL I/O CHANNELS
   EXTENDERS ARE AVAILABLE FOR 2114B OR 2116B

3. TIME BASE GENERATOR (TBG)

4. EXTENDED ARITHMETIC UNIT (EAU)
   AVAILABLE ONLY ON 2116B

5. MEMORY PROTECT (MP)
   AVAILABLE ONLY ON 2116B

6. PHOTOREADER

7. PAPER TAPE PUNCH

8 LINE PRINTER (2778A CDC OR 80 COLUMN D.P.)

9. MARK SENSE CARD READER

10. MAGNETIC TAPE (3030A OR 7970A)

11. CALCOMP PLOTTER

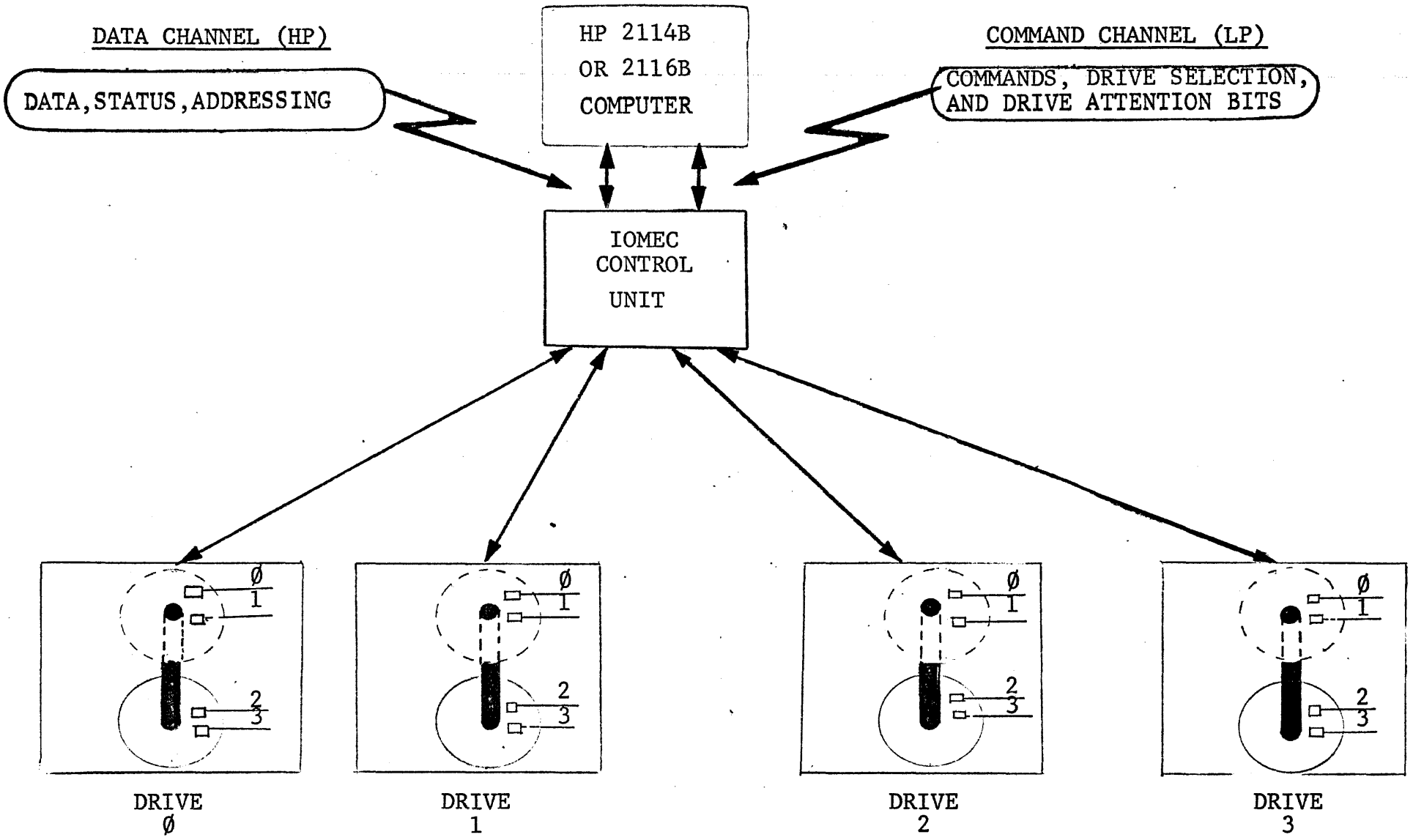
12. UP TO THREE ADDITIONAL DRIVES

DOSM ADVANTAGES AND DISADVANTAGES WITH RESPECT TO DOS

ADVANTAGES	DISADVANTAGES
1. Lower initial system cost.	1. Slower Disc average access time (about 100 milliseconds).
2. Lower Disc expandable cost.	2. Cumbersome and somewhat confusing system bootstrap.
3. Flexibility in options (EAU, TBG, MP) selection.	3. No other HP Disc based system (RTE, TSB).
4. Use of 2114B or 2116B.	4. Overselling 8K feature.
5. Flexibility in Disc Storage medium with removeable cartridge.	
6. Disc content protection.	
7. Disc labeling capability.	
8. Lower core resident system.	
9. Interdisc file(s) transfer.	

[SLIDE 2]

COMPUTER/DISC CONTROLLER/DISC DRIVES LAYOUT



[SLIDE 3]

PHYSICAL ALLOCATION FOR EACH DRIVE

203 CYLINDERS
4 TRACKS PER CYLINDER
12 SECTORS PER TRACK
128 WORDS PER SECTOR

∴ TOTAL WORD CAPACITY PER DRIVE IS
203 X 4 X 12 X 128 = 1,247,232 WORDS

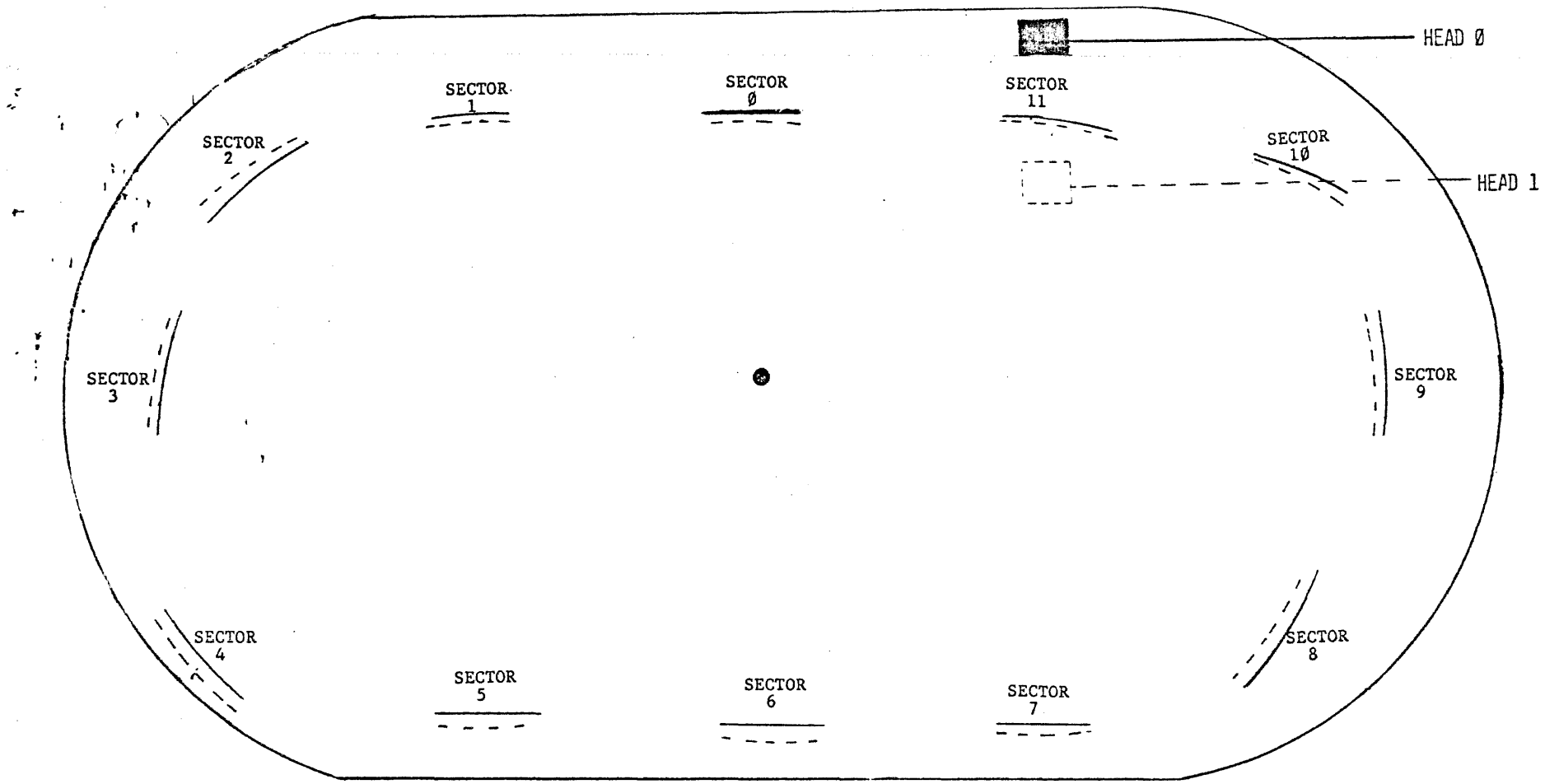
PHYSICAL ALLOCATION FOR EACH DISC

203 CYLINDERS
2 TRACKS PER CYLINDER
12 SECTORS PER TRACK
128 WORDS PER SECTOR

∴ TOTAL WORD CAPACITY PER DISC IS
203 X 2 X 12 X 128 = 623,616 WORDS

[SLIDE 4]

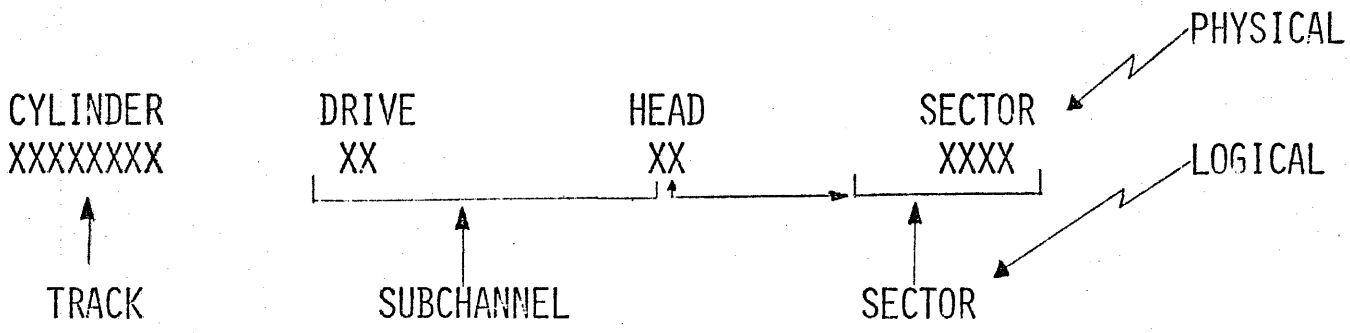
TWO PHYSICAL TRACKS ON CARTRIDGE DISC



"TOP" SURFACE IS ONE PHYSICAL TRACK (12 SECTORS) FOR THIS POSITION OF HEAD Ø
"BOTTOM" SURFACE IS ONE PHYSICAL TRACK (12 SECTORS) FOR THIS POSITION OF HEAD 1
THERE ARE 202 OTHER POSITIONS (CYLINDERS) THESE HEADS MAY BE MOVED TO!

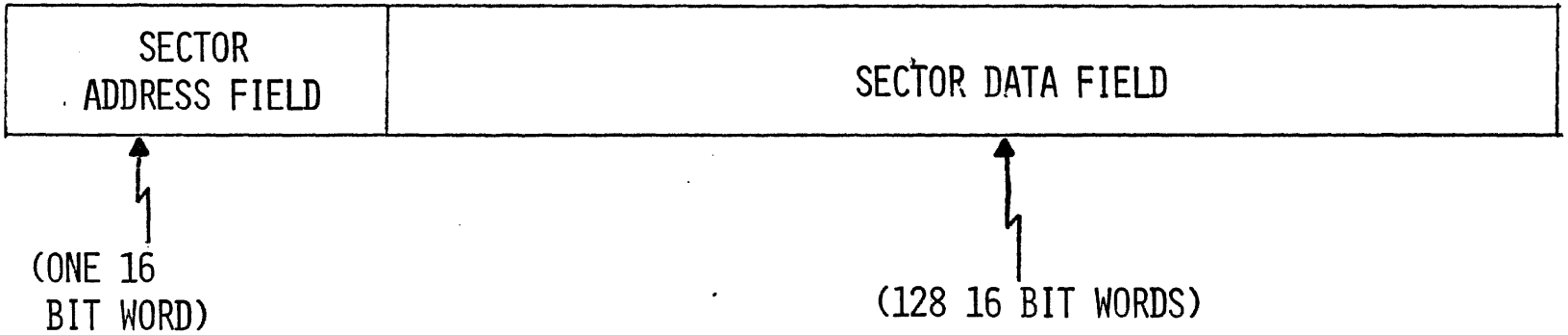
[SLIDE 5]

DISC ADDRESSING



SUBCHANNEL	DRIVE	HEADS	ADDRESSED DISC
000	00	10 11	FIXED DISC DRIVE Ø HEADS 2 & 3
001		00 01	CARTRIDGE DISC DRIVE Ø HEADS Ø & 1
010	01	10 11	FIXED DISC DRIVE 1 HEADS 2 & 3
011		00 01	CARTRIDGE DISC DRIVE 1 HEADS Ø & 1
100	10	10 11	FIXED DISC DRIVE 2 HEADS 2 & 3
101		00 01	CARTRIDGE DISC DRIVE 2 HEADS Ø & 1
110	11	10 11	FIXED DISC DRIVE 3 HEADS 2 & 3
111		00 01	CARTRIDGE DISC DRIVE 3 HEADS Ø & 1

CONTENTS OF EACH SECTOR



1. SECTOR ADDRESS FIELD CONTAINS:
   - 8 BITS FOR CYLINDER #
   - 2 BITS FOR HEAD #
   - 4 BITS FOR ITS SECTOR #
   - 1 BIT USED FOR DCI (DEFECTIVE CYLINDER INDICATOR)
   - 1 BIT USED FOR PCI (PROTECTED CYLINDER INDICATOR)

2. SECTOR DATA FIELD CONTAINS DATA TRANSFERRED TO AND FROM COMPUTER

NOTE: BOTH FIELDS ARE CYCLIC CHECKED BY CONTROLLER

DOSM SOFTWARE COMPONENTS (PART 1)

PROGRAM	NAME(S)	GENERAL FUNCTION(S)
System Generator ⊛	DSGEN	DOSM System Generation User Disc and Cartridge Formatting
System Bootstrap ⊛	BOOTSTRAP	Preparation of configured System bootstrap
Disc Monitor	DISCM	Interrupt Processing ($CIC) Executive Processor (EXEC) I/O Processor ($IORQ)
Executive Modules	$EXØ1	Disc Work Track Status
	$EXØ2	Disc Work Track Limits
	$EXØ3	Program Completion
	$EXØ4	Program Suspension
	$EXØ5	Program Segment Load
	$EXØ6	User File Name Search
	$EXØ7	Current Time Processor
	$EXØ8	Real-Time Disc Allocation
	$EXØ9	Execution Time :EQ Processor
	$EX1Ø	Load and Execute Program
	$EX11	System File Name Search
	$EX12	System Startup
	$EX13	Error Message Processor
	$EX14	Execution Time :UP, :DN, :LU Processor
	$EX15	Abort and Post Mortem Dump
	$EX16	:GO Parameter Processor
	$EX17	:UD Processor
	$EX18	I/O Initiation Processor
	$EX19	:IN Processor
	$EX2Ø	Disc Parity Error Processor
Executive Module Subroutines	$LBL	Service Routines for Label Checking
	$SRCH	Search System or User Directory
	$ADDR	Buffer Address Validity Check
	ASCII	Convert Binary to ASCII
	DUMRX	RTE simulation routines
Special DOSM Drivers	DVRØ5	System Teleprinter Driver
	DVR31	Moving Head Disc Driver

⊛ This is an ABSOLUTE program executed in a separate process from the DOSM system

[SLIDE 8A]

DOSM SOFTWARE COMPONENTS (PART 2)

PROGRAM	NAME(S)	GENERAL FUNCTION(S)
Job Processor	JOBPR	Directive Processing File Management
Relocating Loader	LOADR	Relocates relocatable binary code created by Assembler or Compilers.
Assembler	ASMB ASMBD ASMB1 ASMB2 ASMB3 ASMB4 ASMB5	Translates Assembly language source code into binary. EAU or NON-EAU options included. (MAIN SECTION) (6 SEGMENTS)
HP Basic FORTRAN Compiler	FTN FTNØ1 FTNØ2 FTNØ3 FTNØ4	Translates HP Basic FORTRAN source code into NON-EAU relocatable binary. (MAIN SECTION) (4 SEGMENTS)
ALGOL Compiler ⊛	ALGOL ALGL1	Translates HP ALGOL source code into Non-EAU relocatable binary. (MAIN SECTION) (1 SEGMENT)
FORTRAN IV Compiler (4K user)	FTN4	Translates ASA FORTRAN IV source code into Non-EAU relocatable binary. Consists of one MAIN section and 18 SEGMENTS.
FORTRAN IV Compiler (10K user)	FTN4	Translates ASA FORTRAN IV source code into NON-EAU relocatable binary. Consists of one MAIN section and 2 SEGMENTS.
CROSS REFERENCE TABLE GENERATOR	XREF	Generates Cross Reference Table for Assembly Language Source Code.

⊛ Requires minimum 16K environment

DOSM SOFTWARE COMPONENTS (PART 3)

DOS AND DOSM DRIVERS

EQUIPMENT TYPE CODE (DVR__)	DEVICE	DOS ONLY	DOSM ONLY	BOTH	DMA
ØØ	Teleprinter			X	
Ø1	Photoreader			X	
Ø2	Punch			X	
Ø5	Teleprinter		X		
1Ø	Plotter			X	
12	2778A CDC Line Printer			X	
15	Mark Sense Card Reader			X	X
~~16~~ 12	Data Products Line Printer 2767A (8Ø column)			X	
22	3Ø3Ø Mag. Tape			X	X
23	797Ø Mag. Tape			X	X
3Ø	Fixed Head Disc	X			X
31	Moving Head Disc (IOMEC)		X		X
~~32~~ 31	Moving Head Disc (ISS)		X		X

Equipment Type Code Numbering Convention

ØØ - Ø7 Paper Tape Devices } Odd # for INPUT, Even # for OUTPUT

1Ø - 17 Unit Record Devices

2Ø - 37 Mass Storage Devices

[SLIDE 8C]

DOSM SOFTWARE COMPONENTS (PART 4)

LIBRARIES

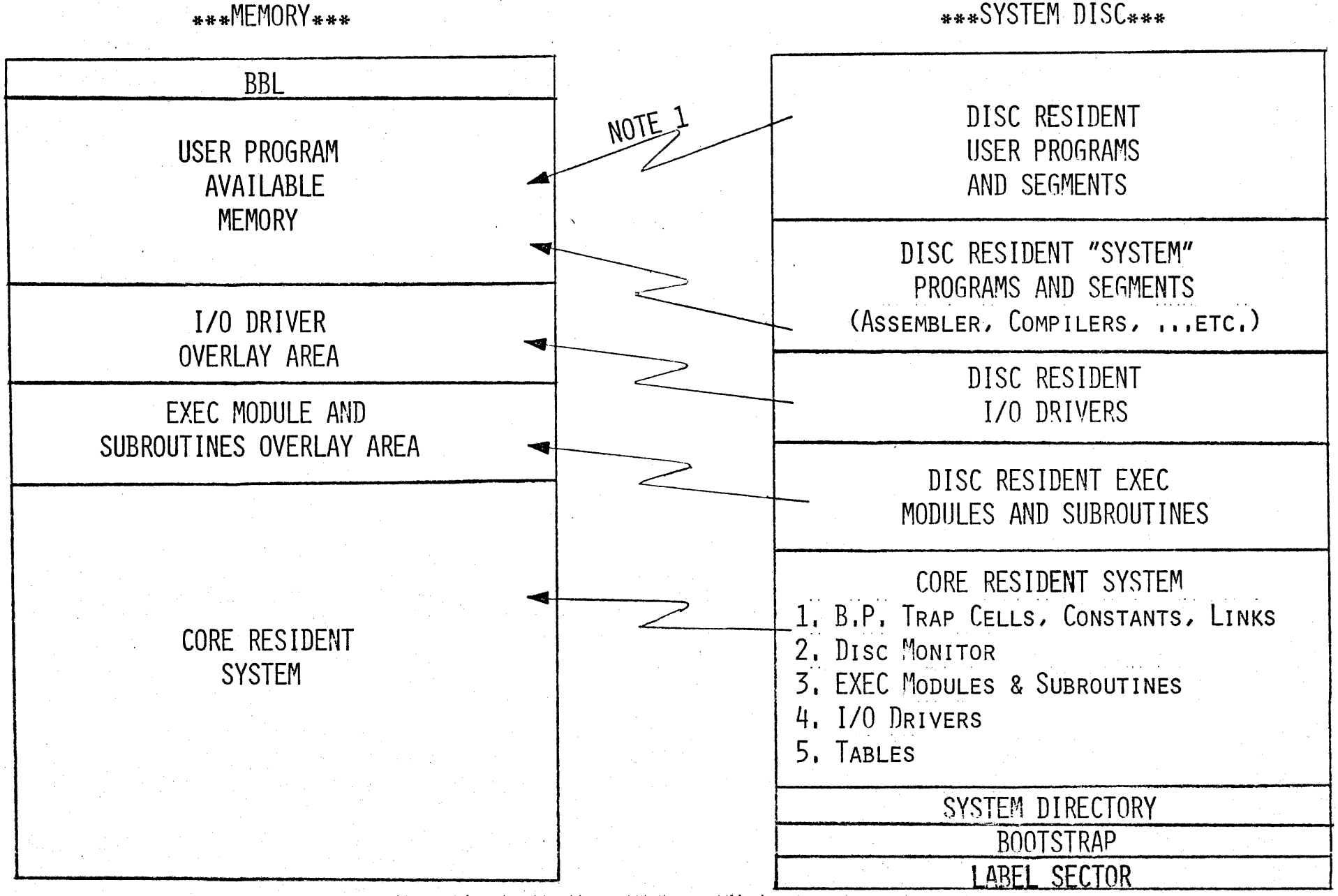
NAME	RELOCATABLE LIBRARY TYPE
F2N.V	NON-EAU RTE/DOS/DOSM (no Formatter)
F2E.V	EAU RTE/DOS/DOSM (no Formatter)
F4D.V	RTE/DOS/DOSM FORTRAN IV with FORTRAN IV Formatter (Double Precision)
_____	RTE/DOS/DOSM HP FORTRAN Formatter (no Double Precision)
_____	RTE/DOS/DOSM Plotter

where V = the revision letter (A, B, C ....)

NOTES:

1. System must include F2N.V or F2E.V even if FORTRAN IV library (F4D.V) is to be included. This is because the FORTRAN IV library references routines whose entry points are in F2N.V and F2E.V libraries.
2. RTE/DOS/DOSM HP FORTRAN Formatter is seperate from F2N.V and F2E.V due to FORTRAN IV library (F4D.V) containing a formatter.

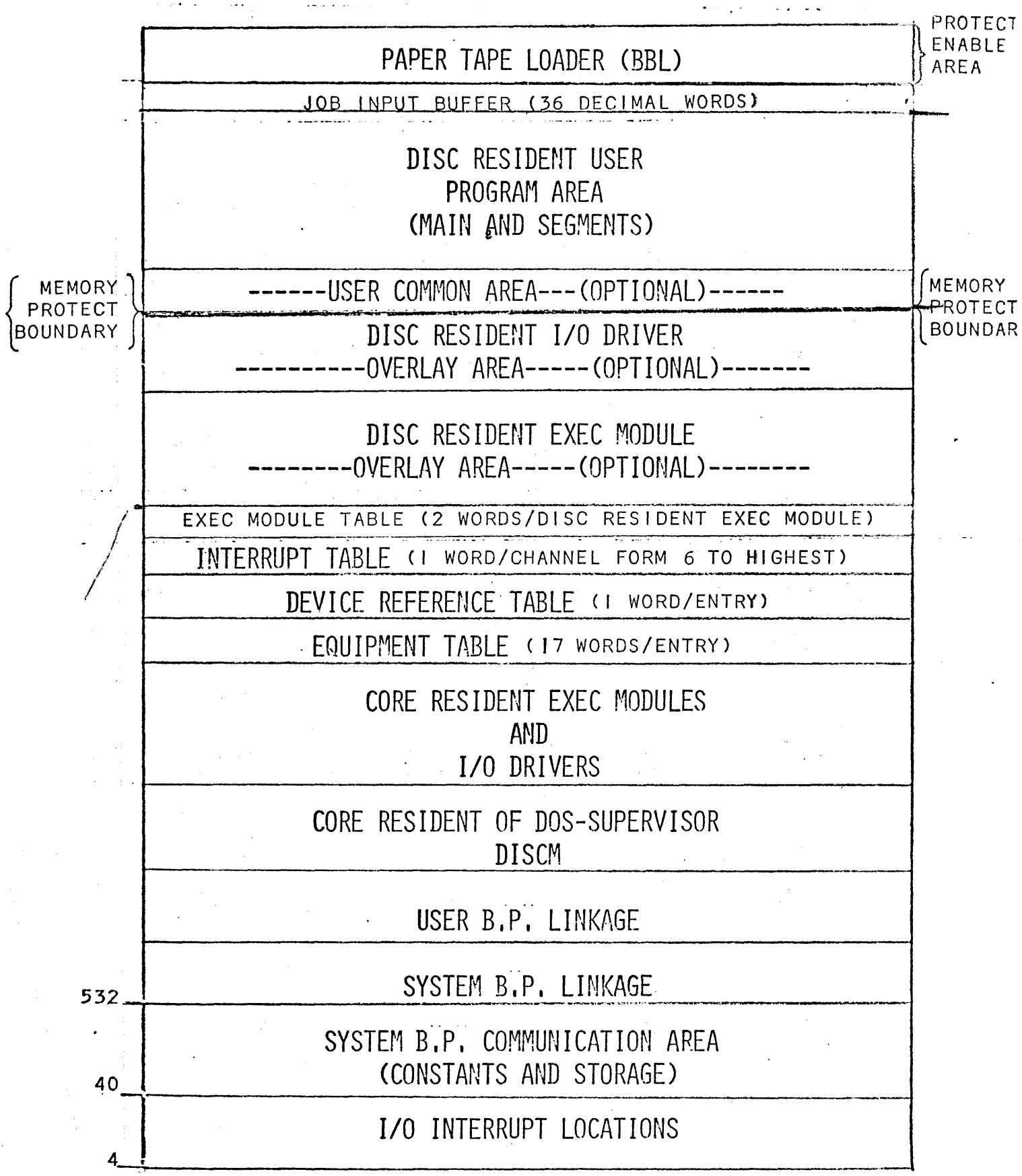
DOSM DISC TO MEMORY TRANSFERS (GENERAL)



NOTE 1: MAY BE FROM "SYSTEM" DISC OR USER DISC(s)

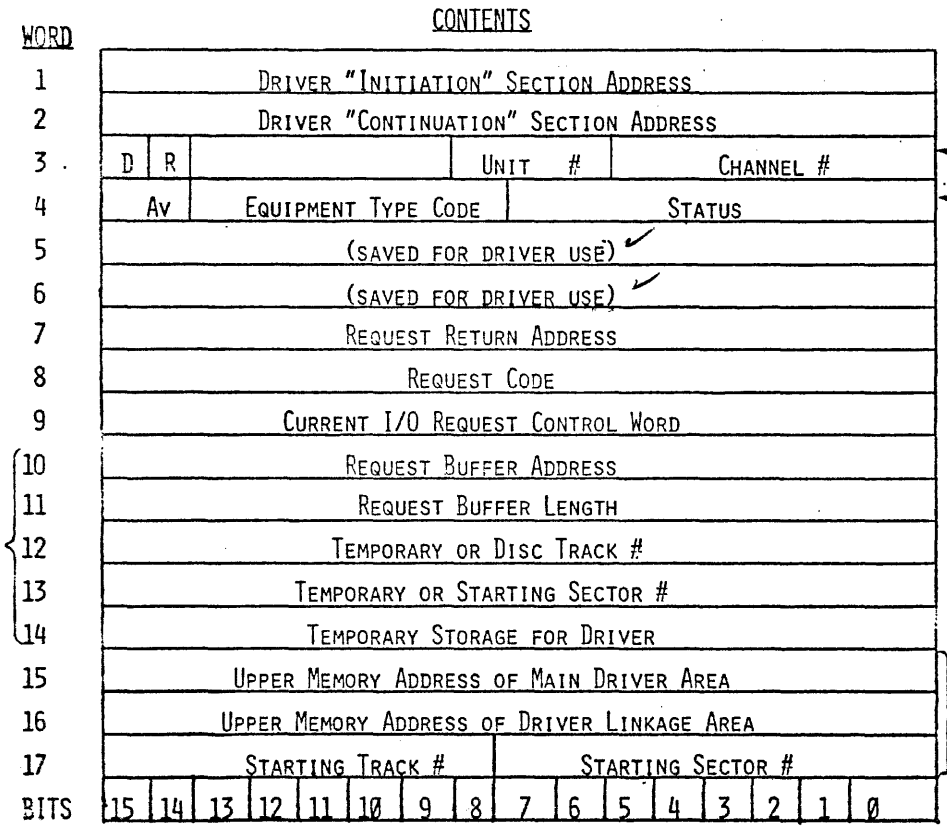
[SLIDE 9]

DOSM GENERAL CORE LAYOUT



[SLIDE 10A]

EQUIPMENT TABLE ENTRY FORMAT



D = 1 if DMA channel required.
R = 1 if driver type is core-resident.
Unit # May be used for sub-channel addressing.
Channel # I/O select code for device (lower number if multiboard interface.)

Av = Ø - Unit not busy and available
= 1 - Unit disabled (down)
= 2 - Unit busy
= 3 - Unit waiting for an available DMA channel
(This field set by system)

Status - Actual or simulated unit status at end of operation. (Driver must set this field)

Equipment Type Code - Identifies type of device and associated software driver. Assigned equipment type codes in octal are:

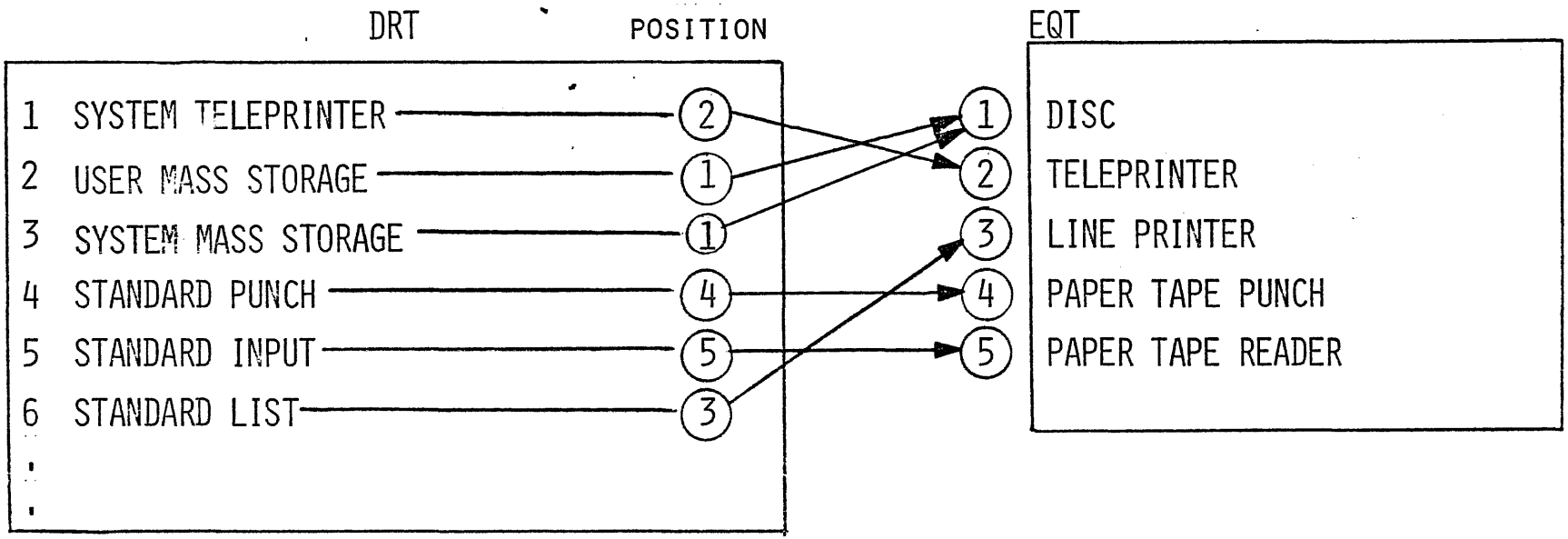
Ø0-07	Papter Type Devices
Ø0	Teleprinter
Ø1	Punched Tape Reader
Ø2	High Speed Punch
Ø5	Teletype (System)
10-17	Unit Record Devices
10	Reserved for Plotter
12	Line Printer
15	Mark Sense Card Reader
20-37	Magnetic Tape/Mass Storage and other devices capable of both input and output.
22	3030 Magnetic Tape
31	Moving-Head Disc

For equipment type codes 1 through 17, odd number indicate input devices and even number indicate output devices (except Ø5, which is both input and output).

[SLIDE 10B]

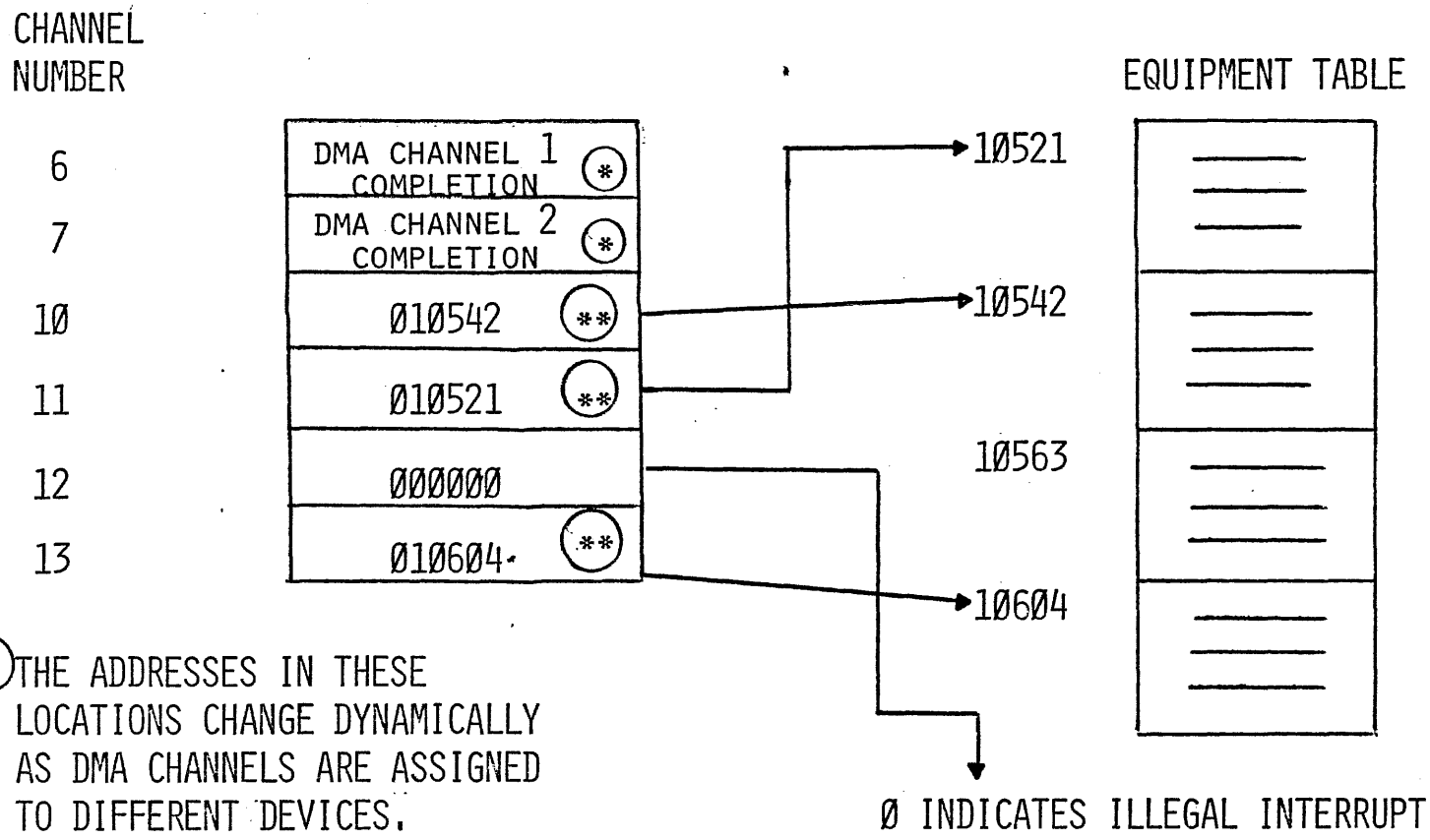
THE DEVICE REFERENCE TABLE

THE DEVICE REFERENCE TABLE PROVIDES FOR LOGICAL ADDRESSING OF PHYSICAL UNITS DEFINED IN THE EQUIPMENT TABLE. THE DRT CONSISTS OF ONE WORD ENTRIES CORRESPONDING TO THE RANGE OF USER-SPECIFIED LOGICAL UNITS (1 TO N, WHERE N ≤ 63). THE CONTENTS OF THE WORD CORRESPONDING TO A LOGICAL UNIT IS THE RELATIVE POSITION OF THE EQT ENTRY DEFINING THE PHYSICAL UNIT.



THE INTERRUPT TABLE

THE INTERRUPT TABLE CONTAINS A ONE WORD ENTRY FOR EACH I/O DEVICE. THESE ENTRIES CONTAIN THE ADDRESSES OF EQUIPMENT TABLE ENTRIES FOR DEVICES ASSOCIATED WITH THESE CHANNELS.

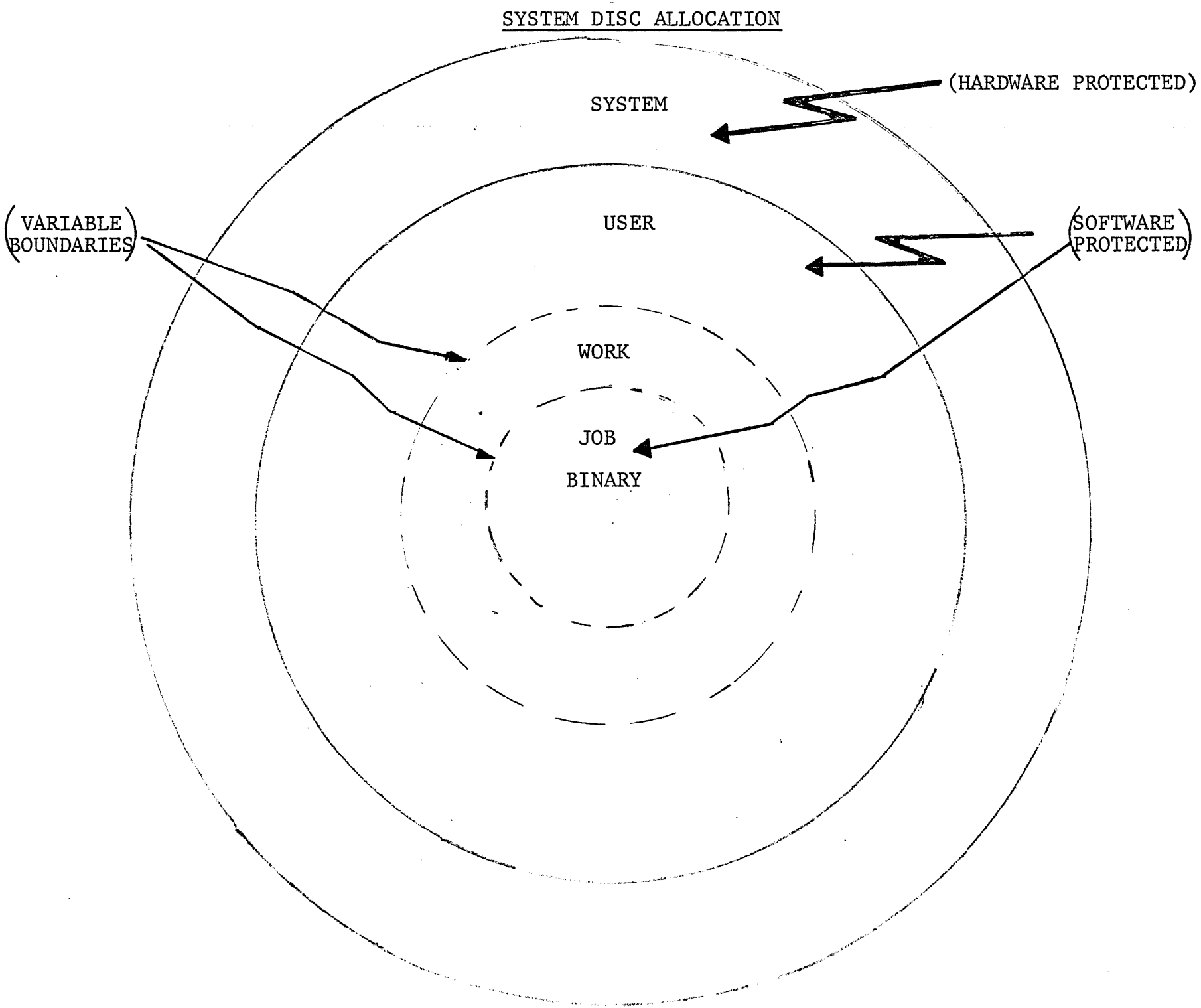


(*) THE ADDRESSES IN THESE LOCATIONS CHANGE DYNAMICALLY AS DMA CHANNELS ARE ASSIGNED TO DIFFERENT DEVICES.

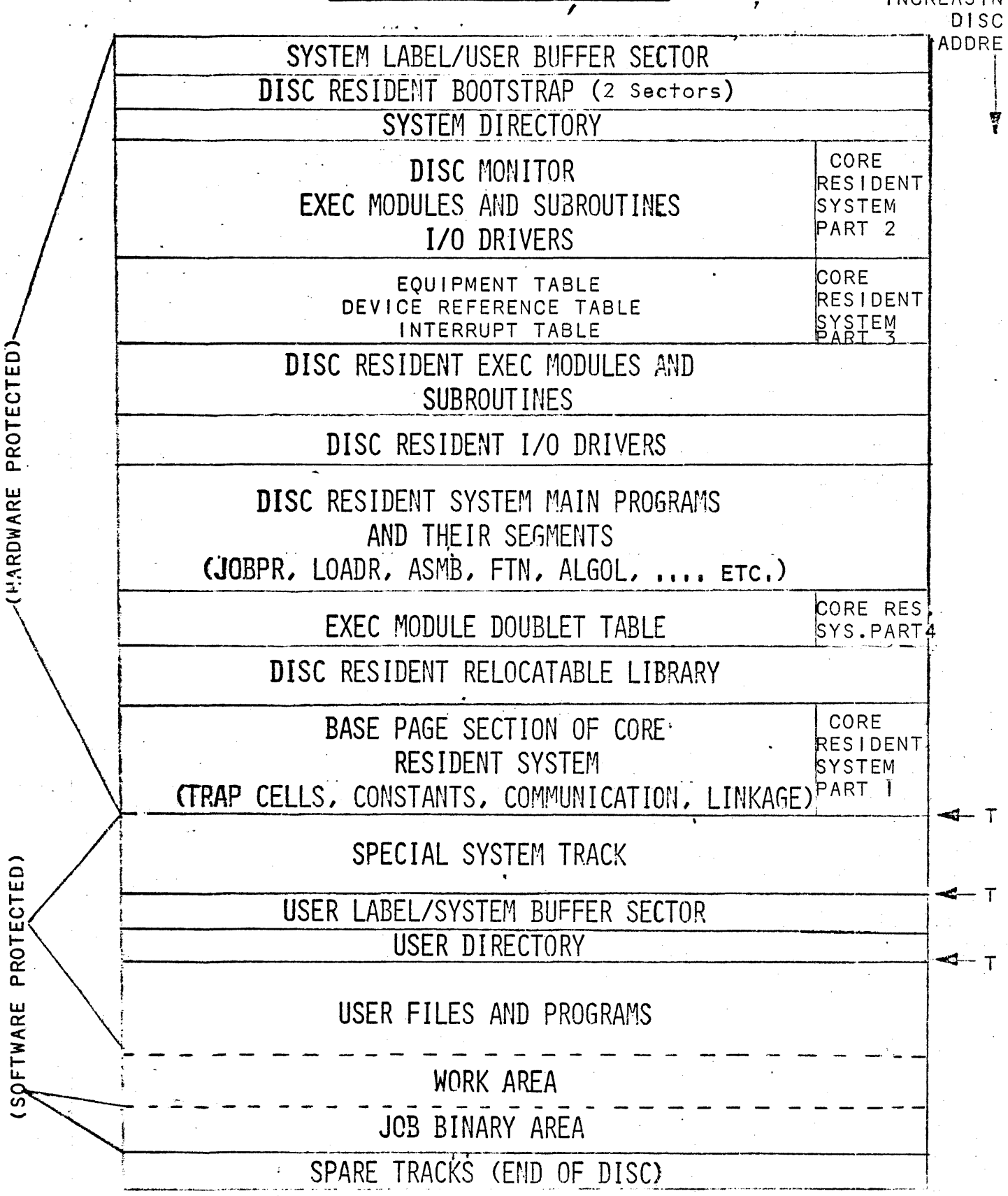
(**) THE ADDRESSES IN THESE LOCATIONS CHANGE DYNAMICALLY AS DIFFERENT EQT ENTRIES ARE USED FOR GIVEN I/O CHANNEL.

[SLIDE 10D]

SYSTEM DISC ALLOCATION



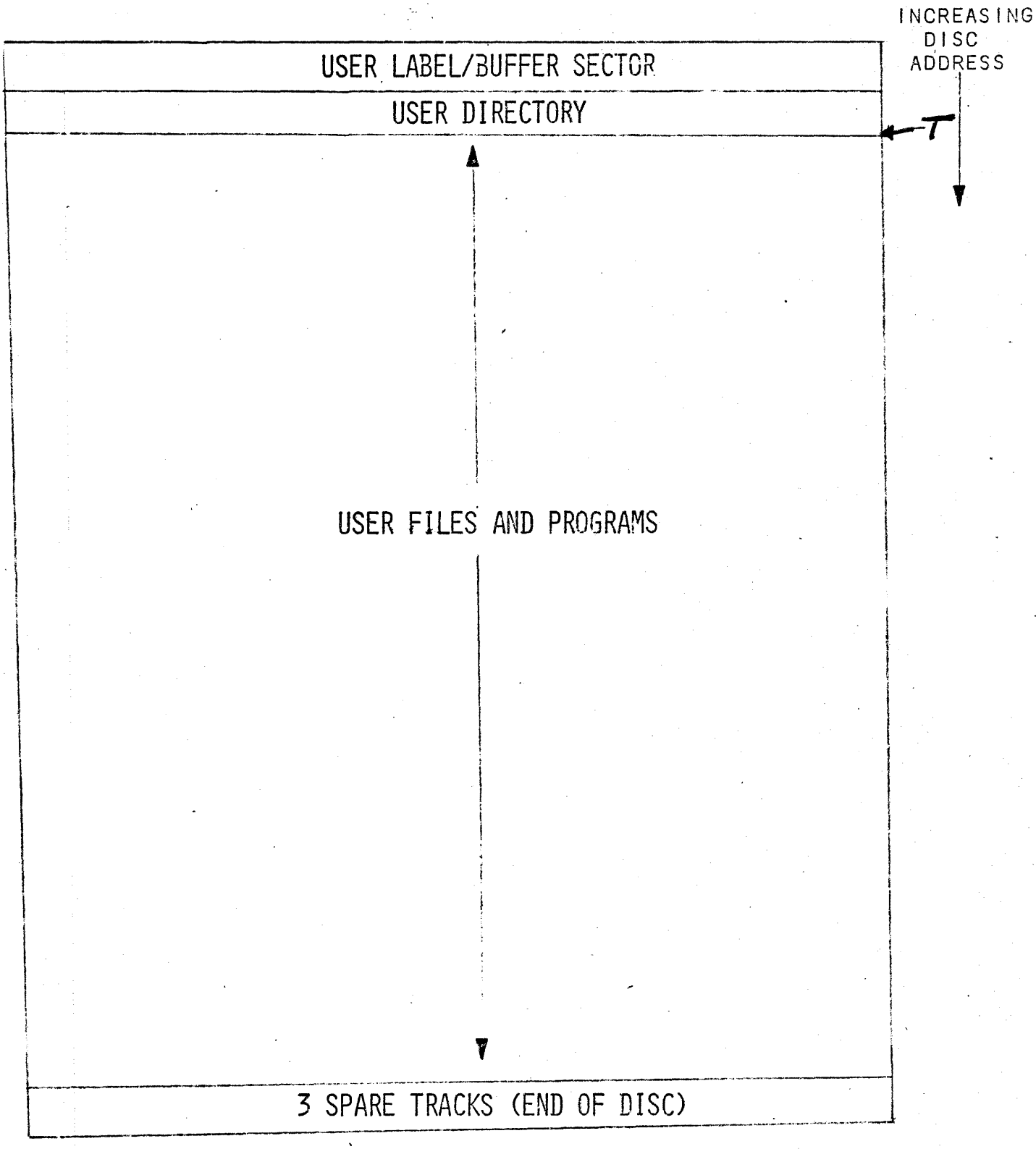
DOSM "SYSTEM" DISC LAYOUT



T = STARTS ON TRACK BOUNDARY

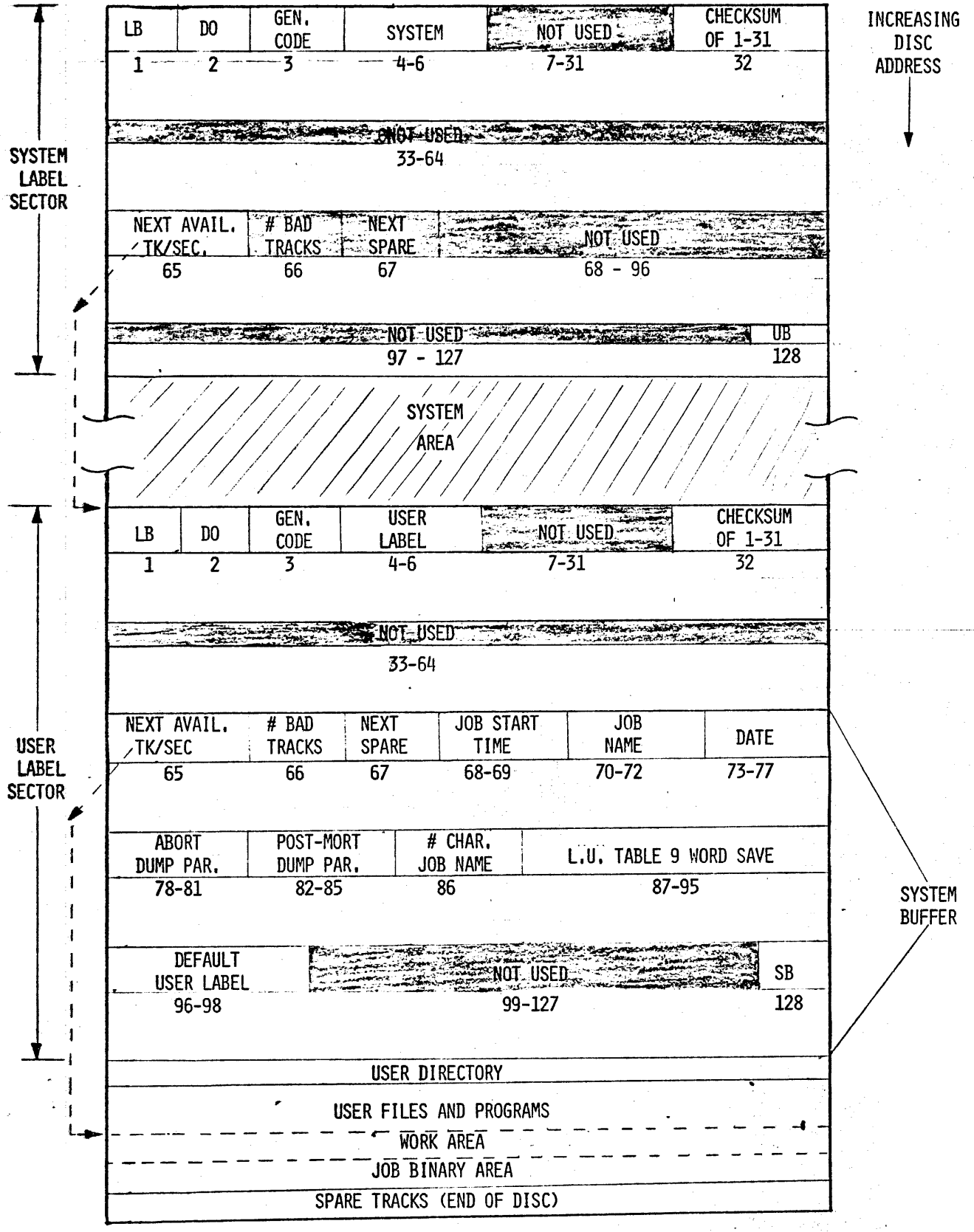
[SLIDE 11B]

DOSM "USER" DISC LAYOUT



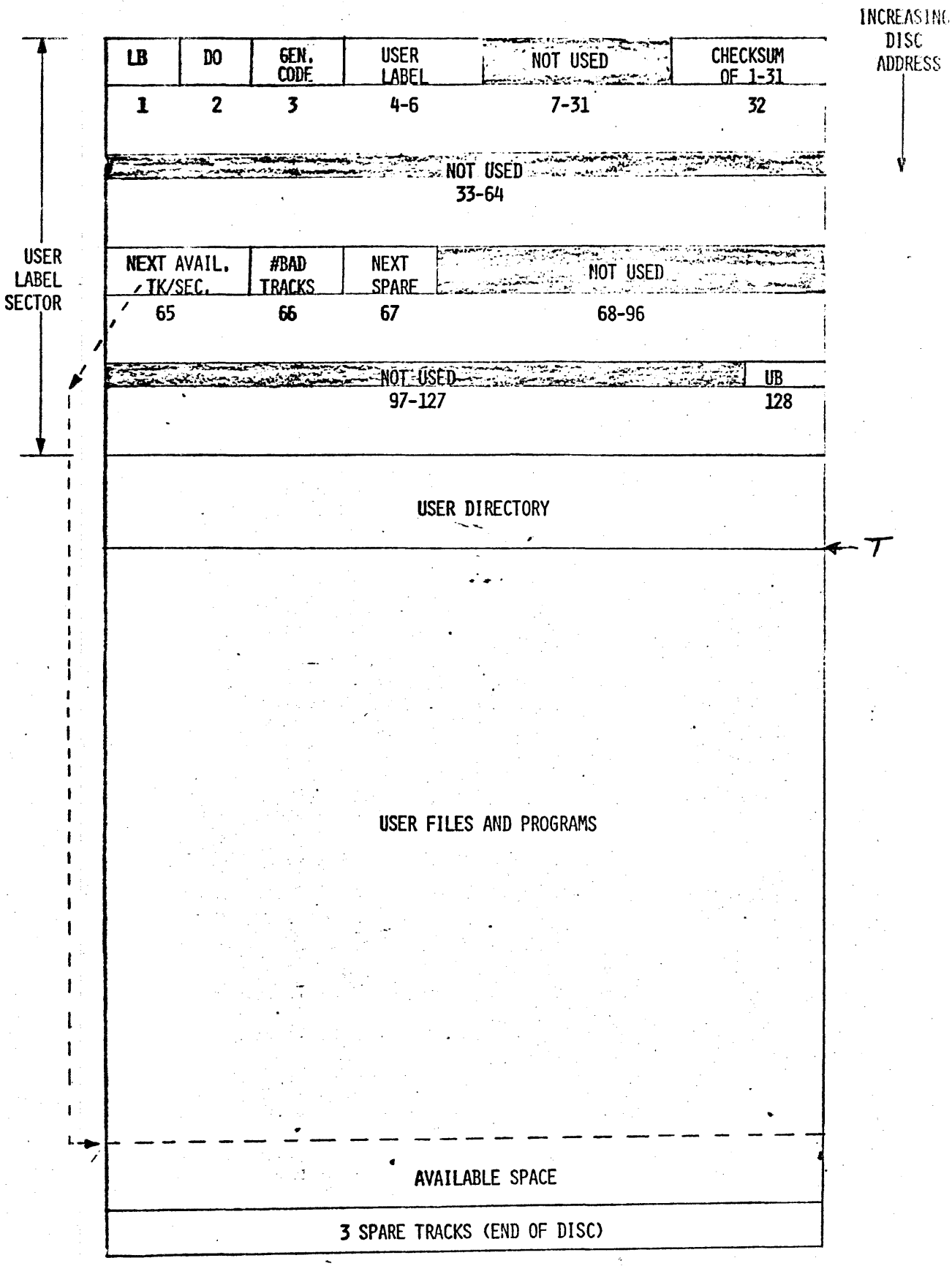
[SLIDE 11C]

"SYSTEM" DISC (LABEL SECTORS DESCRIPTION)

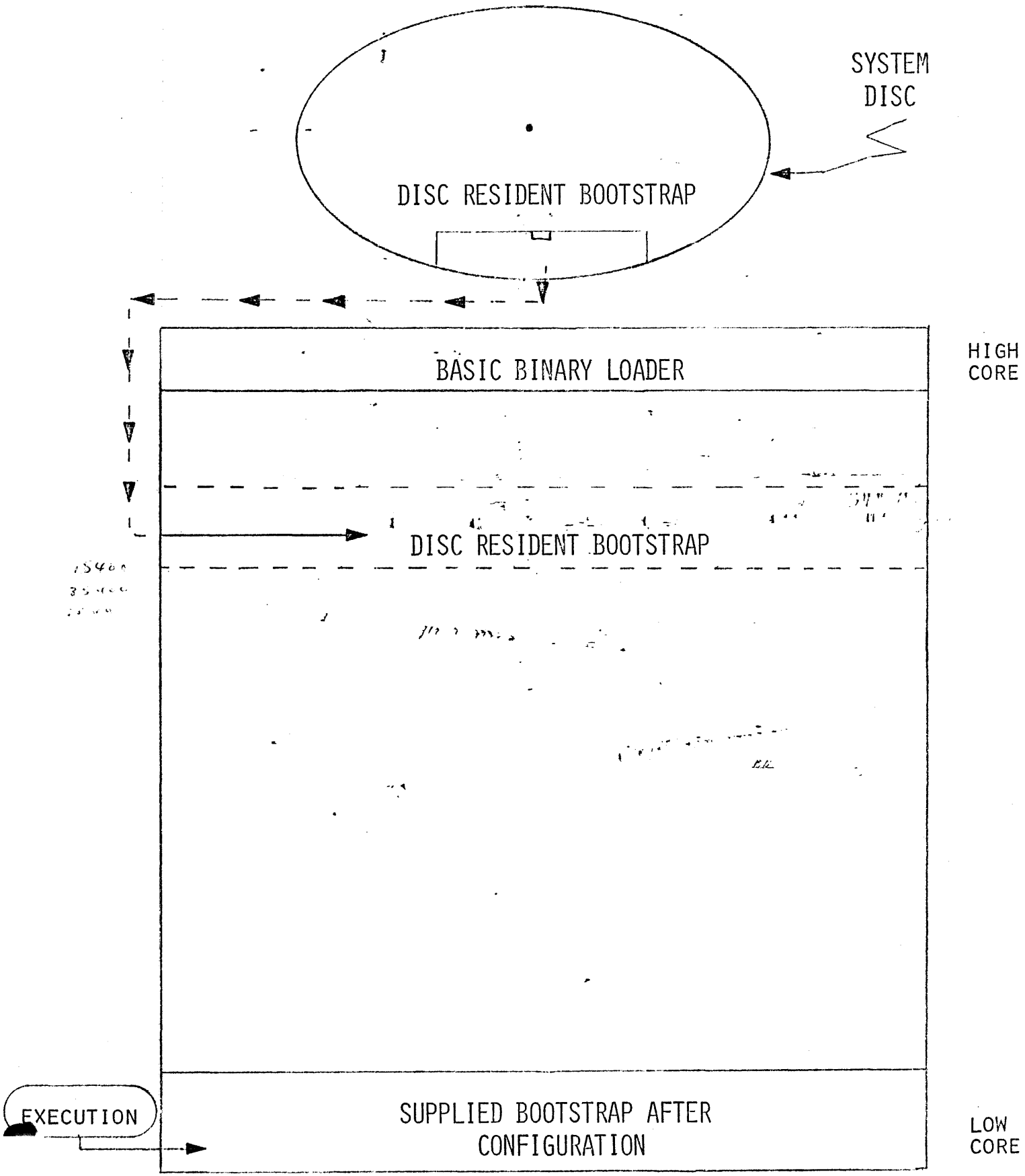


[SLIDE 12A]

"USER" DISC (LABEL SECTOR DESCRIPTION)

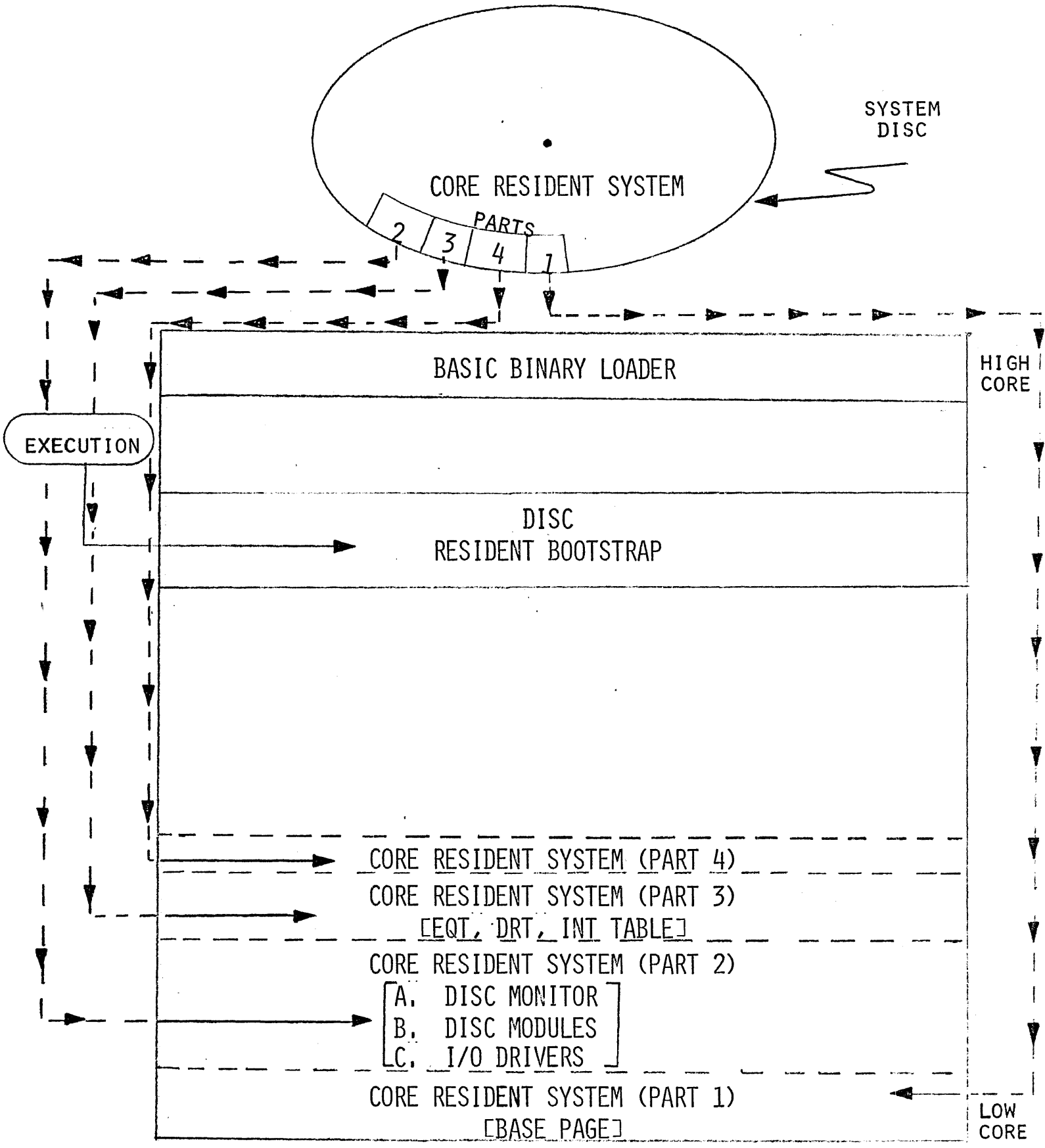


SYSTEM STARTUP (PART 1)
(EXECUTION OF CONFIGURED BOOTSTRAP)



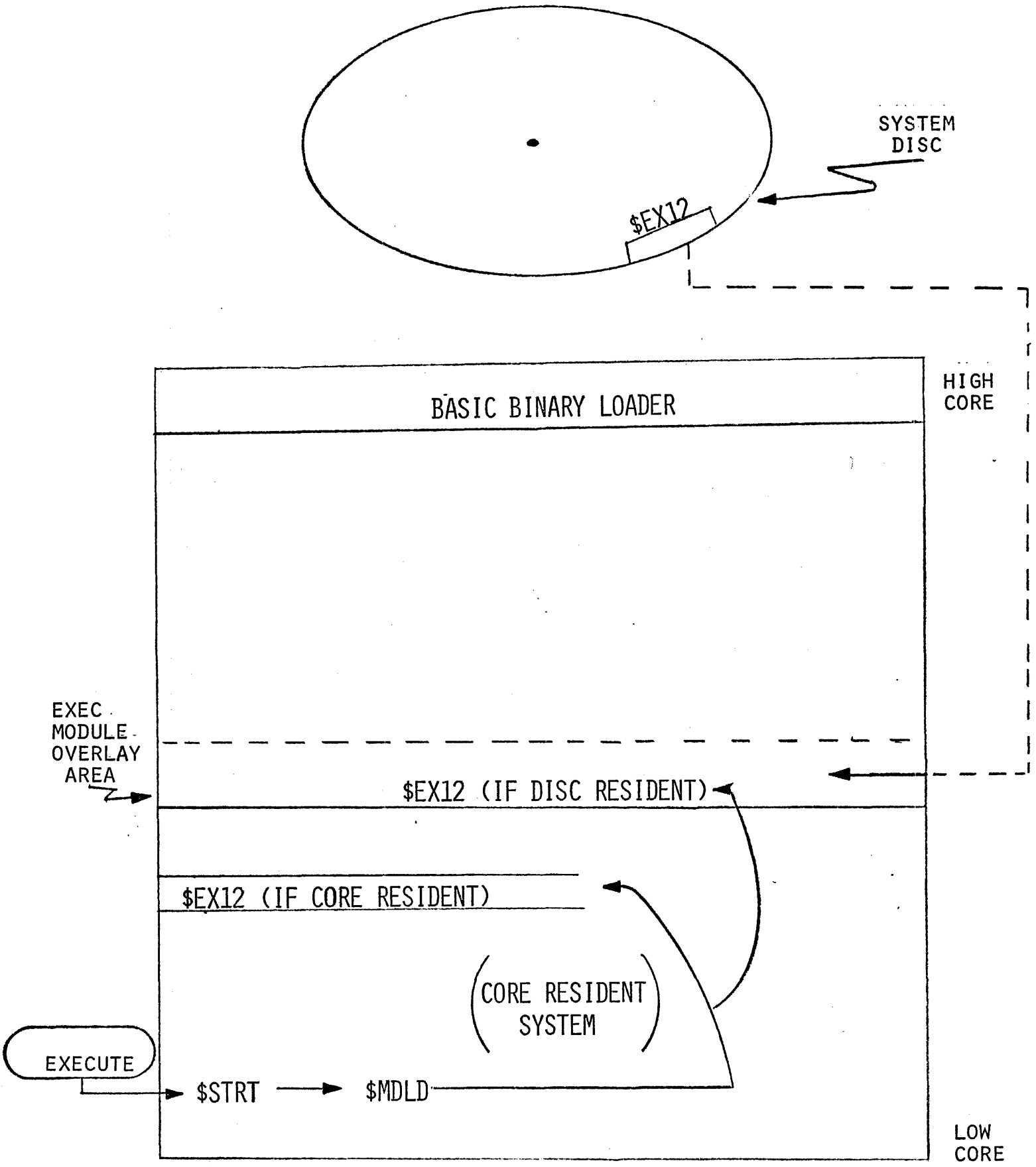
[SLIDE 13A.1]

SYSTEM STARTUP (PART 2)
(EXECUTION OF DISC RESIDENT BOOTSTRAP)



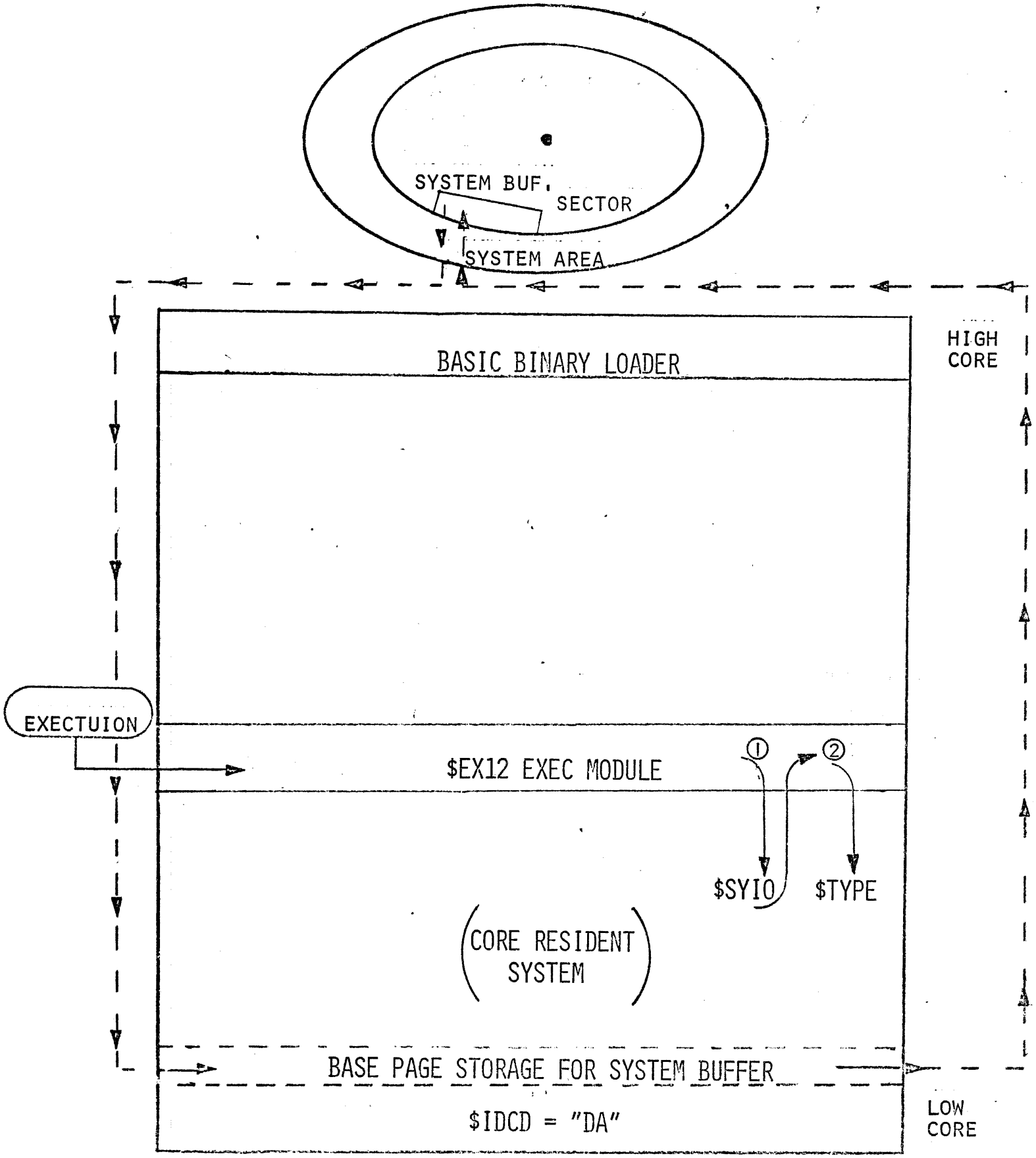
[SLIDE 13A.2]

SYSTEM STARTUP (PART 3)
(DISC MONITOR FIRST ENTRY)



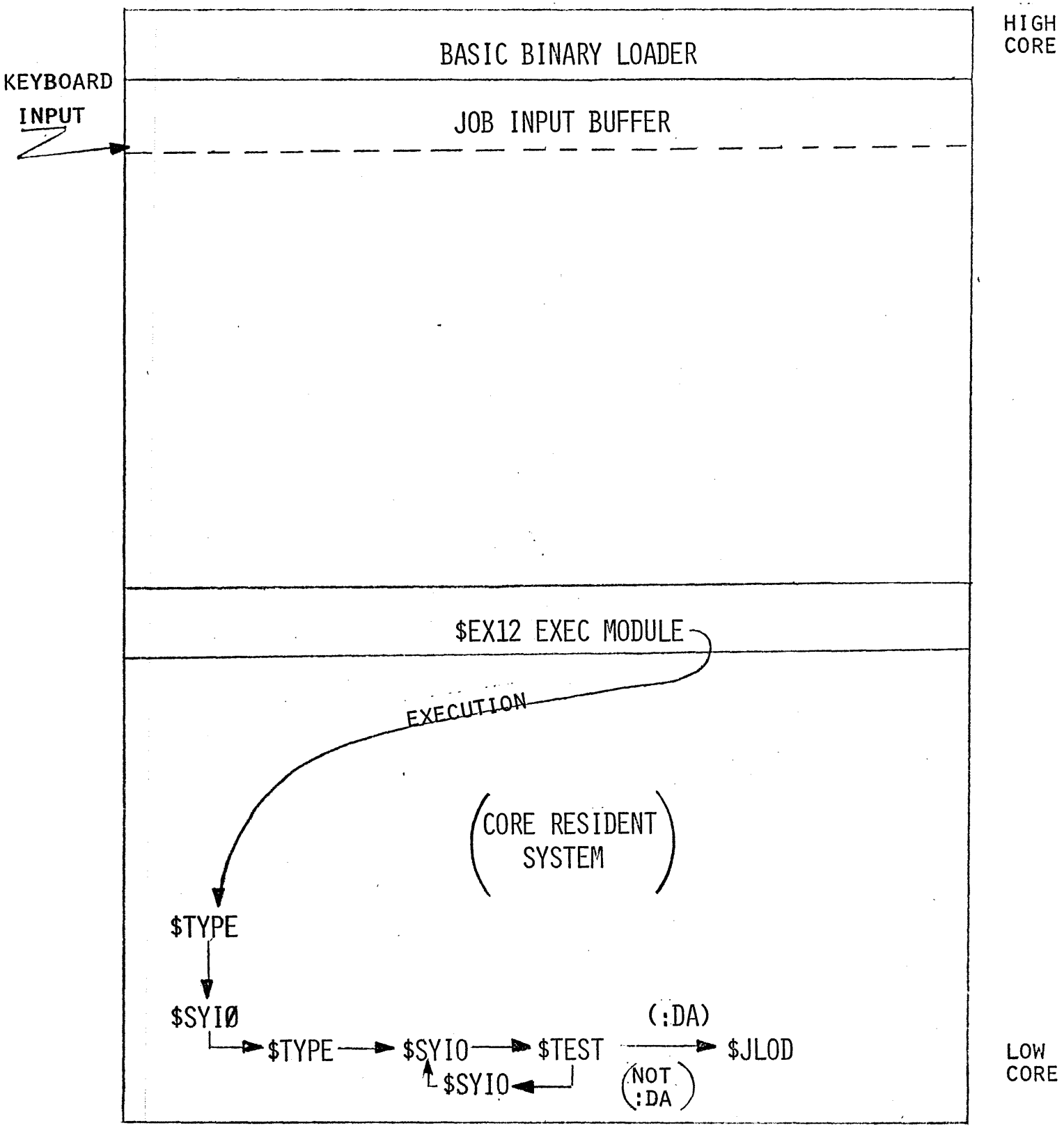
[SLIDE 13A.3]

SYSTEM STARTUP (PART 4)
(EXECUTION OF $EX12 EXEC MODULE)



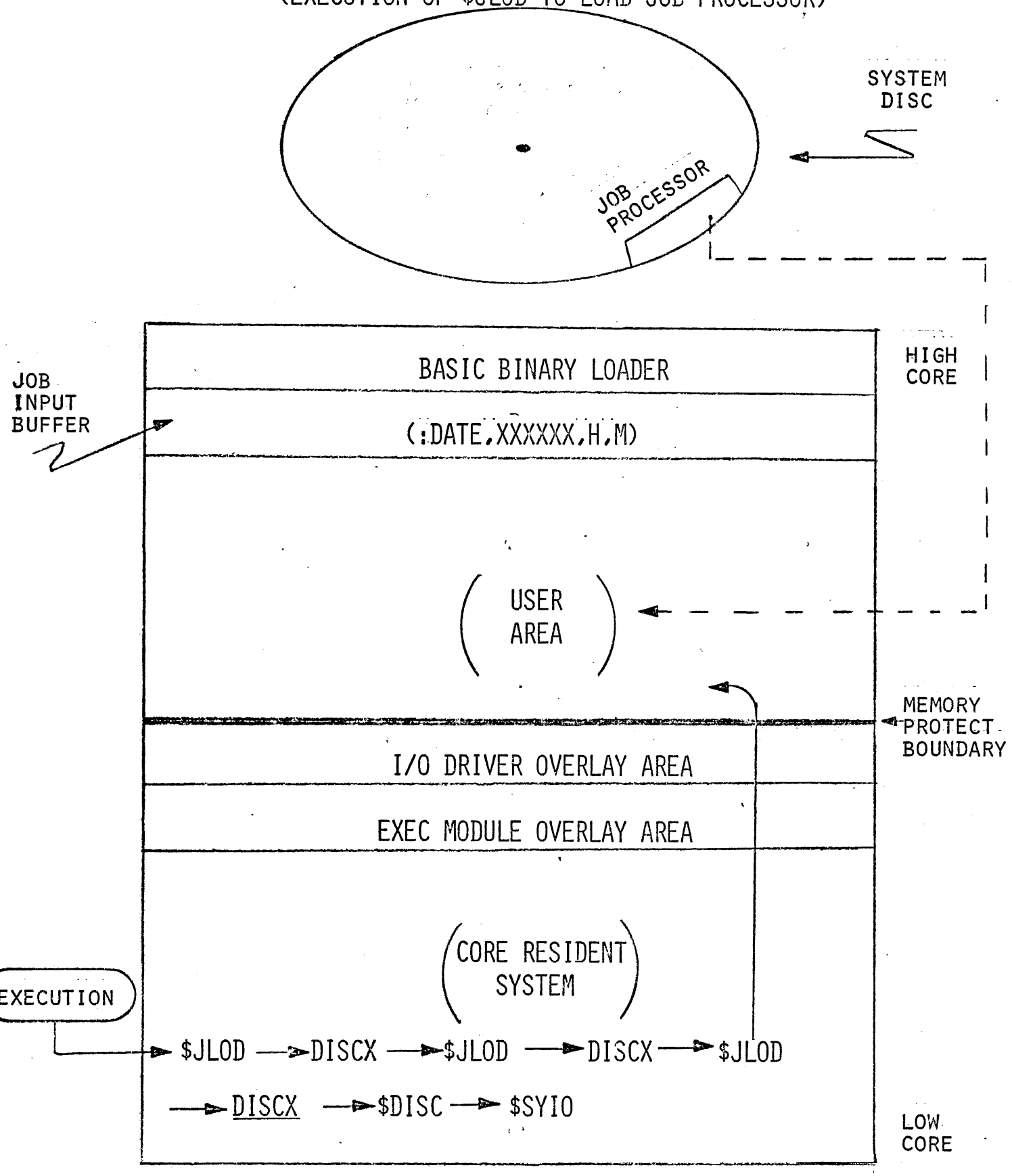
[SLIDE 13A.4]

SYSTEM STARTUP (PART 5)
(EXECUTION OF $TYPE FOR ":DATE" INPUT)



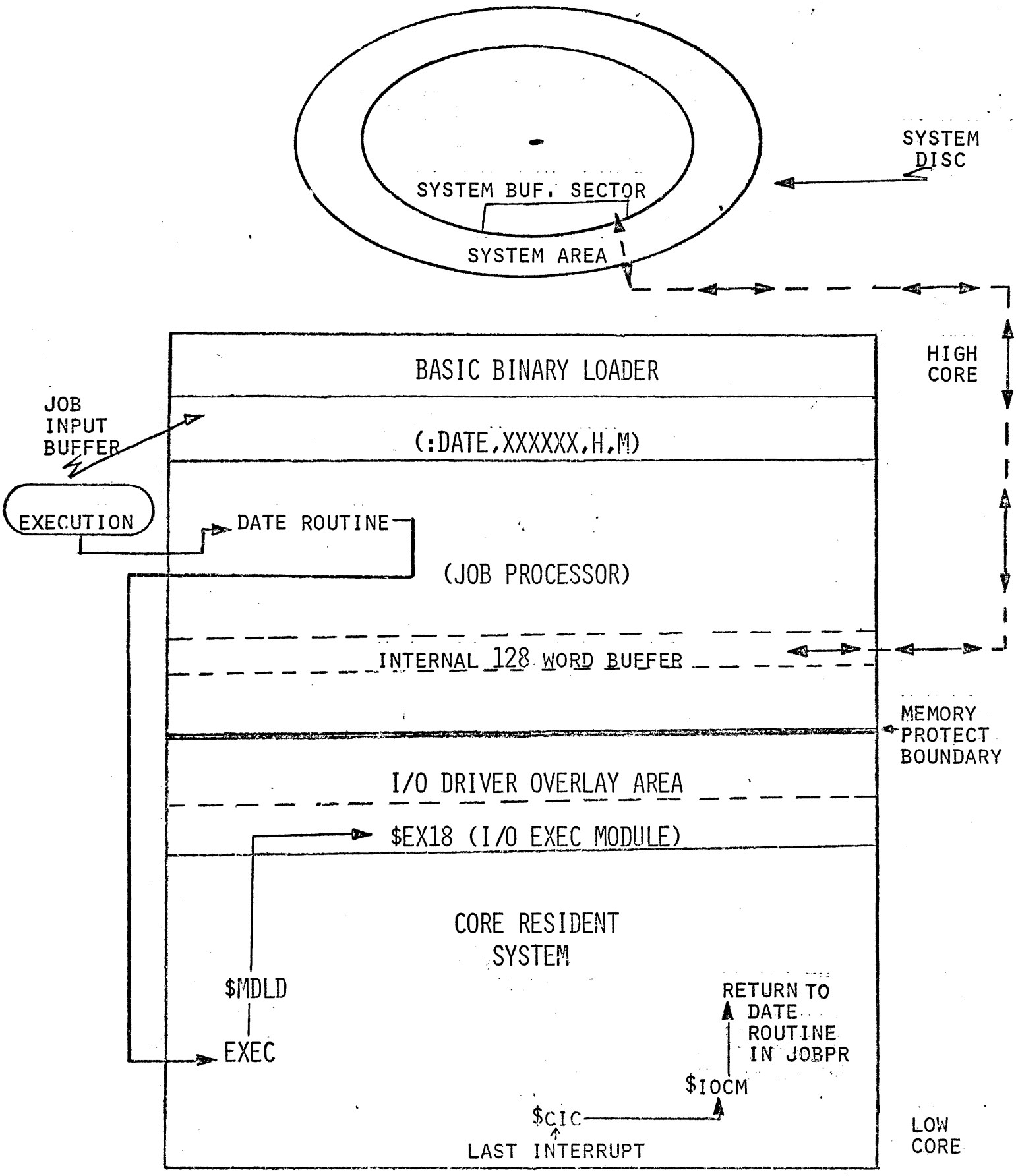
[SLIDE 13A.5]

SYSTEM STARTUP (PART 6)
(EXECUTION OF $JLOD TO LOAD JOB PROCESSOR)



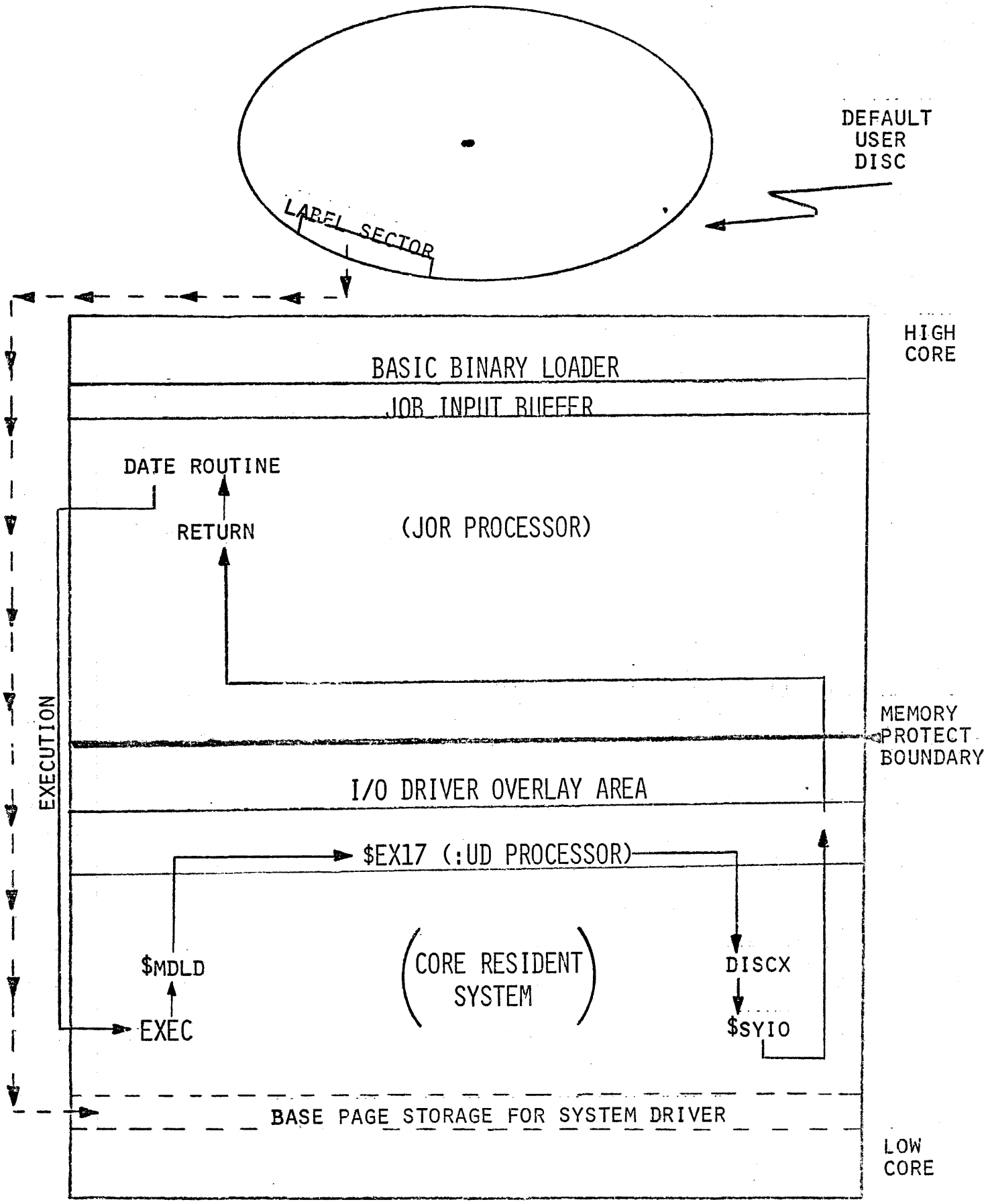
[SLIDE 13A.6]

SYSTEM STARTUP (PART 7)
(EXECUTION OF JOBPR TO UPDATE SYSTEM BUFFER SECTOR)



[SLIDE 13A.7]

SYSTEM STARTUP (PART 8)
(REPORTING OF DEFAULT USER DISC SUBCHANNEL AND LABEL)



[SLIDE 13A.8]

II. OPERATIONAL DIFFERENCES FROM DOS

A. System Startup

1. Bootstrap rather than BBDL (at X776Ø).

   a. DOS procedure using BBDL.

   b. DOSM procedure using Bootstrap [SLIDES 14A & 14B].

      (1) Configure and execute.

      or (2) Configure, punch configured bootstrap, load configured bootstrap.

   c. No "FR" or "CO" entry statement as in DOS.

B. NEW DIRECTIVES

1. :OFF [SLIDE 15].

   a. Does not clear Job Binary Area.

2. :IN [SLIDE 16A & 16B]

   a. Can prepare discs for use in DOSM that were formatted by other software

   b. DISC PROTECT OVERRIDE SWITCH must be "ON" to purge a protected Disc (PCI=1).

3. :UD [SLIDES 17A, 17B].

4. :DD [SLIDE 18].

   a. No SDUMP with DOSM; do not need!

   b. Source Disc for User Area is current user disc.

5. :SS [SLIDES 19A, 19B].

   a. Duplicate file handling

6. Example Slides using the above directives. [SLIDES 20A, 20B, 20C].

C. Operational Difference Summary [SLIDES 21A, 21B, 21C, 21D].

III. PROGRAMMING DIFFERENCES FROM DOS

A. New EXEC call to change user disc [SLIDES 22A, 22B, 22C].

   1. :EJOB resets changes made.

   2. Error messages if incorrect SYSTEM GENERATION or SYSTEM PROPRIETARY CODE, but assignment still made.

B. DOS/DOSM EXEC calls with negative request codes [SLIDE 23A].

C. DOSM Disc I/O with EXEC calls [SLIDE 23B].

D. EXEC calls difference summary [SLIDE 24].

E. Other important points.

   1. If MP option not used the following are valid.

      a. All I/O instructions and HALT

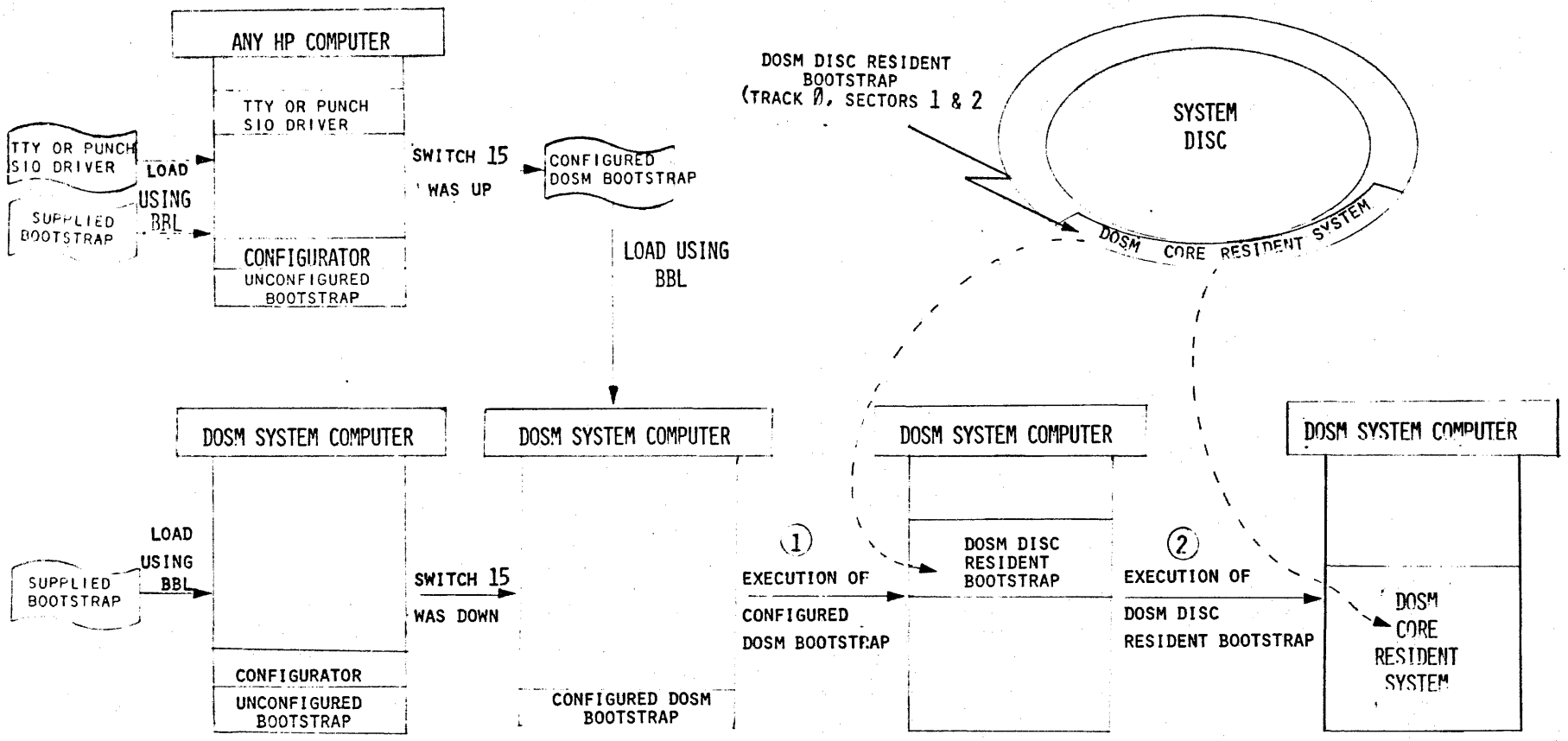
      b. Base page modifications.

      c. Special interaction with DISC MONITOR (DISCM).

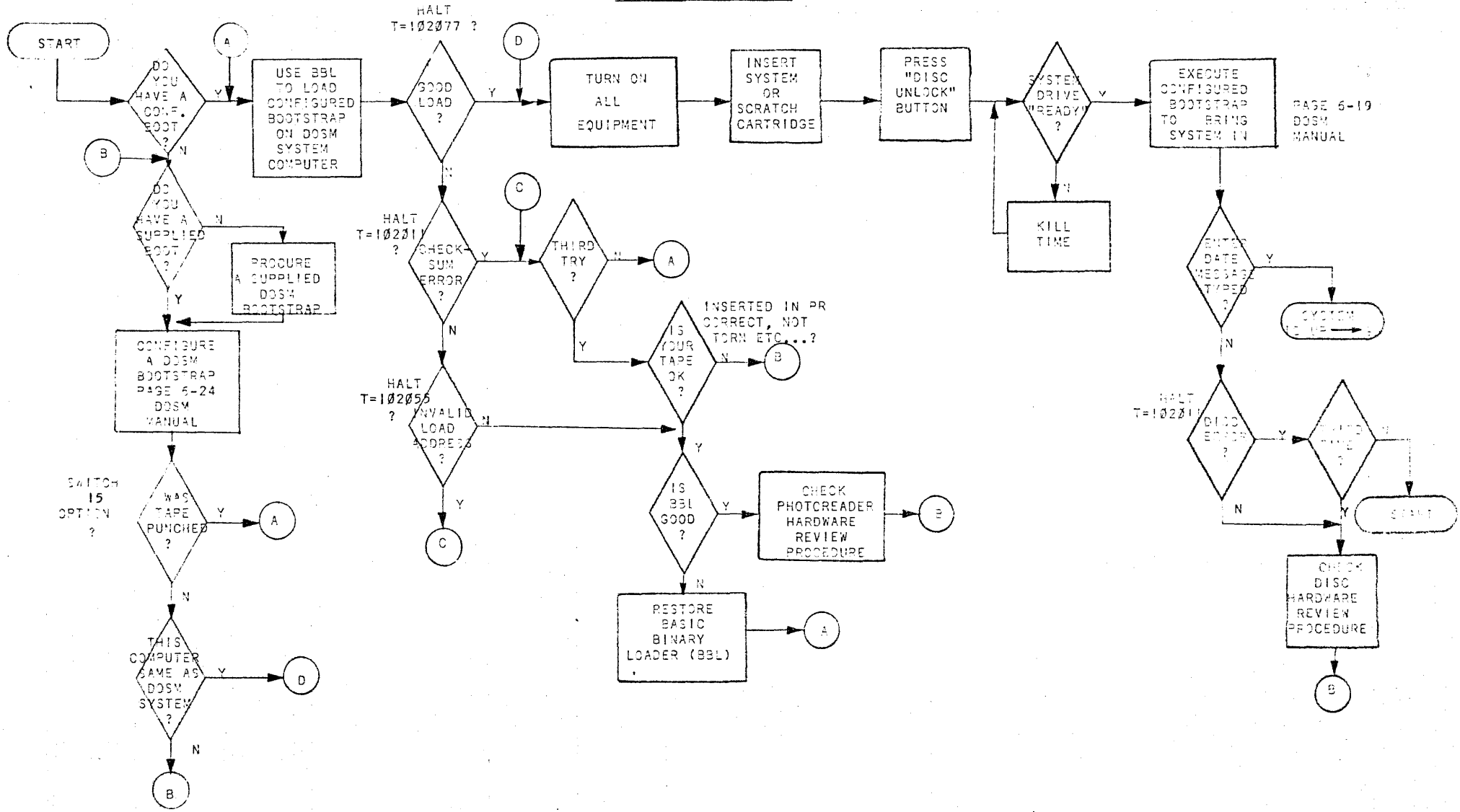
   2. LOADR is unaffected by :SS condition. File searches it initiates are for only current user disc. Order of scanning is:

      a. JOB BINARY (if any programs in it)
      b. USER FILES (if any given)
      c. PAPER TAPE (if specified)
      d. DISC RESIDENT RELOCATABLE LIBRARY

BOOTSTRAPPING DOSM UP FROM DISC



DOSM SYSTEM START-UP



OFF DIRECTIVE

PURPOSE: TO ABORT CURRENTLY EXECUTING USER PROGRAM OF SYSTEM OPERATION WITHOUT TERMINATING THE JOB.

FORMAT: :OFF

NOTES:

1. RETURNS SYSTEM TO KEYBOARD MODE.
2. CAN BE USED TO TERMINATE UNDESIRED LISTS, EDITS, DISC-TO-DISC DUMPS, PROGRAM LOOPS, LOADER OPERATIONS, ASSEMBLIES, AND COMPILATIONS.
3. CANCELS ANY :DD, :AD, OR :PD DIRECTIVES, UNLESS A PROGRAM IS RUNNING, IN WHICH CASE A PENDING :AD IS EXECUTED.
4. MUST NEVER BE GIVEN DURING A PURGE (:PURGE DIRECTIVE) OR FOLLOWING /E IN AN EDIT LIST.

INITIALIZE DIRECTIVE

PURPOSE: TO LABEL OR UNLABEL THE CURRENT USER DISC.

FORMAT: :IN,LABEL

WHERE THE LABEL IS A SIX-CHARACTER NAME TO BE WRITTEN IN WORDS 4-6 OF THE LABEL SECTOR ON THE CURRENT USER DISC. A "*" IS ENTERED TO UNLABEL THE DISC. THE FIRST LETTER OF LABEL MUST NOT BE "CONTROL@".

NOTES:

1. IF THE CURRENT USER DISC IS ALREADY LABELED, DOSM PRINTS THE FOLLOWING MESSAGE:

   {TSB / DOS / ???} LABEL XXXXXX (WHERE XXXXXX IS EXISTING LABEL)

   OK TO PURGE?

   THE OPERATOR THEN ANSWERS "YES" OR "NO".

2. :IN,* EXECUTION PURGES ALL FILES ON THE CURRENT USER DISC AS FOLLOWS BELOW:

   A. LABEL PRESENCE CODE SET = Ø.
   B. FIRST AVAILABLE TK/SEC SET TO START OF USER AREA (IF USER DISC) OR START OF SYSTEM DIRECTORY TRACK +1 (IF SYSTEM DISC).
   C. SETS FIRST WORD IN DIRECTORY = Ø TO INDICATE END-OF-DIRECTORY.
   D. * SET IN FIRST CHARACTER OF LABEL FIELD.
   E. SETS SYSTEM GENERATION CODE AND PROPRIETARY CODE = TO THAT ON SYSTEM LABEL SECTOR.
   F. GENERATES NEW CHECKSUM.

3. :IN,NEWLB PURGES UNPROTECTED SYSTEM AND MOVES ANY USER FILES DOWN TO LOW DISC IF THE CURRENT USER DISC IS LABELED "SYSTEM" AND IS NOT HARDWARE PROTECTED (I.E., IT WAS CREATED WITH :DD,X DIRECTIVE).

4. SYSTEM GENERATION CODE AND SYSTEM PROPRIETARY CODE ARE SET EQUAL TO THOSE IN THE CURRENT SYSTEM.

CHANGE USER DISC DIRECTIVE (PART 1)

PURPOSE: TO CHANGE SUBCHANNEL ASSIGNMENT FOR THE CURRENT USER DISC.

FORMAT: :UD [, [LABEL] [,N]]

WHERE LABEL IS A SIX-CHARACTER LABEL OR
*IF UNLABELED DISC.

AND N IS THE SUBCHANNEL NUMBER.

NOTES:

1. SIX BASIC FORMS ARE POSSIBLE (PART 2).
2. IF THE DISC ON SUBCHANNEL #N HAS A SYSTEM PROPRIETARY CODE NOT EQUAL TO "DO", THE ASSIGNMENT IS STILL MADE AND THE SYSTEM PRINTS THE FOLLOWING:

   {TSB / ???} DISC

3. IF THE DISC ON SUBCHANNEL #N HAS A SYSTEM GENERATION CODE NOT EQUAL TO THAT OF THE CURRENT SYSTEM, THE ASSIGNMENT IS STILL MADE AND THE SYSTEM PRINTS THE FOLLOWING:

   DISC GEN CODE XXXX NOT SYS GEN CODE YYYY ERR POSS

4. USER DISC SUBCHANNEL ASSIGNMENTS MADE BY THIS DIRECTIVE ARE ONLY TEMPROARY; THE USER DISC SUBCHANNEL ASSIGNMENT IS RESET TO THAT SPECIFIED DURING SYSTEM GENERATION AT THE END OF EACH JOB.
5. USED IMMEDIATELY FOLLOWING :DD (DISC DUMP) DIRECTIVE TO SPECIFY DESTINATION DISC.

[SLIDE 17A]

CHANGE USER DISC DIRECTIVE (PART 2)

EXAMPLE	ACTION
:UD (WITHOUT LABEL OR SUBCHANNEL)	INTERROGATES THE CURRENT USER DISC SUBCHANNEL AND PRINTS ITS LABEL ON THE SYSTEM TELEPRINTER: SUBCHAN = n LBL = label (or UNLBL)
:UD,,n (NO LABEL)	IF n IS LABELED, DOS-M PRINTS: LBL = label (OR UNLBL) NO ASSIGNMENT IS MADE.
:UD,label (NO SUBCHANNEL)	DOS-M SEARCHES FOR THE label, STARTING WITH THE HIGHEST NUMBER SUBCHANNEL (DETERMINED AT SYSTEM GENERATION). IF label IS FOUND, DOS-M MAKES IT THE USER DISC AND PRINTS: SUBCHAN = n IF label IS NOT FOUND, DOS-M PRINTS: DISC NOT ON SYS
:UD,label,n	IF n IS LABELED WITH THE SPECIFIED label, DOS-M ASSIGNS n AS THE USER DISC. IF n IS UNLABELED OR HAS A DIFFERENT label, DOS-M PRINTS: LBL = label (OR UNLBL) OPERATOR CAN THEN REISSURE :UD,label,n WITH THE CORRECT LABEL.
:UD,*,n	IF n IS UNLABELED, DOS-M ASSIGNS n AS THE USER DISC. IF n IS LABELED, DOS-M MAKES NO ASSIGNMENT AND PRINTS: LBL = label
:UD,*	ASSIGNS THE HIGHEST NUMBER UNLABELED DISC AS THE USER DISC AND PRINTS: SUBCHAN = n IF THERE ARE NO UNLABELED DISC, DOS-M PRINTS: DISC NOT ON SYS

DISC-TO-DISC DUMP DIRECTIVE

PURPOSE: To dump onto another subchannel

1. An entire Disc using :DD
2. The System Area (including system buffer) using :DD,X.
3. All or specified files of the User Area (optionally assigning some new file names) using

   :DD,U[,FILE 1 [,(FILE A)],FILE 2[,(FILE B)],...,]

where X specifies the System Area
U specifies the User Area
FILE 1, FILE 2, ... specify the files to be dumped
FILE A, FILE B, ... specify the optional new names for FILE 1, FILE 2, ...

NOTES:

A. Renamed files may be intermixed with unchanged files in 3. above.

B. The destination disc must be specified by the :UD directive immediately following the :DD directive. For :DD,:DD,X the following :UD directive must be :UD,*,n where n is not the System Disc.

C. When the destination disc for a :DD,U is a System Disc (other than current system), the user files are dumped in the user area following the System files.

D. If files of the source disc will not completely fit on the destination disc, the system will transfer as many WHOLE FILES as possible and print: TRAC # TOO BIG.

[SLIDE 18]

SYSTEM SEARCH DIRECTIVE (PART 1)

PURPOSE: TO SPECIFY A LIST OF DISC SUBCHANNELS TO BE SEARCHED BY SYSTEM FOR FILE NAMES OTHER THAN THE CURRENT USER DISC.

FORMATS:

```
:SS
```

ALL ACTIVE SUBCHANNELS ARE SEARCHED IN THE FOLLOWING ORDER:

1. CURRENT USER DISC SUBCHANNEL
2. HIGHEST ACTIVE SUBCHANNEL IN SYSTEM
3. NEXT HIGHEST ACTIVE SUBCHANNEL IN SYSTEM

. . .

LOWEST ACTIVE SUBCHANNEL IN SYSTEM

```
:SS,N1,N2,N3,...
```

ALL ACTIVE SUBCHANNELS (WITHIN N1,N2,N3,...LIST) ARE SEARCHED IN THE FOLLOWING ORDER:

1. CURRENT USER DISC SUBCHANNEL
2. LOWEST NUMBERED ACTIVE SUBCHANNEL SPECIFIED IN N1,N2,N3,...LIST.
3. NEXT LOWEST NUMBERED ACTIVE SUBCHANNEL SPECIFIED IN N1,N2,N3,...LIST.

. . .

HIGHEST NUMBERED ACTIVE SUBCHANNEL SPECIFIED IN N1,N2,N3,...LIST.

```
:SS,99
```

ONLY THE CURRENT USER DISC SUBCHANNEL IS SEARCHED. THIS IS THE DEFAULT CONDITION. EVERY JOB STARTS OUT IN THIS CONDITION.

SYSTEM SEARCH DIRECTIVE (PART 2)

NOTES:

1. THIS IS AN OPTIONAL DIRECTIVE VALID ONLY IF "YES" WAS RESPONSE TO ALLOW :SS? QUESTION DURING SYSTEM GENERATION.

2. THE :SS CONDITION SET APPLIES TO ALL EXEC CALLS AND DIRECTIVES THAT REQUIRE A FILE SEARCH.

3. CURRENT USER DISC SUBCHANNEL NUMBER IS CHANGED TO THE SUBCHANNEL CONTAINING THE FILE THAT INITIATED THE FILE SEARCH. THIS IS REPORTED BY SYSTEM EACH TIME IT CHANGES WITH TTY PRINTOUT, SUBCHAN = n IF THE JOB PROCESSOR IS IN CORE (I.E. NO OTHER USER PROGRAM EXECUTING).

4. IF SEARCH DOES NOT FIND THE DESIRED FILE, THE CURRENT USER DISC SUBCHANNEL NUMBER IS RESTORED TO ITS VALUE BEFORE SEARCH.

5. IF SEARCH IS INTERRUPTED BEFORE COMPLETION, THE CURRENT USER DISC SUBCHANNEL NUMBER WILL BE ON WHATEVER SUBCHANNEL NUMBER THE SYSTEM WAS SEARCHING WHEN INTERRUPTION OCCURRED.

6. :LIST,U DIRECTIVE DOES NOT STOP ON DUPLICATE FILE NAMES, BUT CONTINUES SEARCHING AND PRINTING USER DIRECTORY. AT COMPLETION, THE CURRENT USER DISC SUBCHANNEL IS RESTORED TO NUMBER BEFORE THIS DIRECTIVE ENTERED.

7. MORE THAN ONE :SS CONDITION MAY BE SET DURING A JOB. EACH ONE SET REMAINS IN EFFECT UNTIL A NEW ONE IS ENTERED OR THE JOB IS ENDED.

8. :SS CONDITIONS SET ARE NOT FOLLOWED BY RELOCATING LOADER (LOADR) OR TO DISC DUMPS INITIATED BY :DD DIRECTIVE.

:UD, :DD, :OFF, :SS, :IN DIRECTIVE EXAMPLE

```
INPUT :DATE,XXXXXXXXXX,H,M          <-- Brought up System from PACK

@:DA,19.OCT.70,14,0
SUBCHAN=1                           <-- System reports default User Disc Subchannel #
LBL=QQQQQ                               and Label
@
:JOB,EXMP1
JOB EXMP1 19.OCT.70  TIME=0840 MIN. 13.4 SECS.
@
:UD
SUBCHAN=1
LBL=QQQQQ
@
:DD                                 <-- Declared entire Disc to Disc dump
@
:OFF                                <-- Changed mind - bailed out of DD condition
@
:DD,X                               <-- Declared System Area Only Disc dump
@
:UD,*,0                             <-- Declared fixed Disc as destination disc
LBL=SYSTEM
DISC GEN CODE 1013 NOT SYS GEN CODE 9000 ERR POSS
RE-ENTER STATEMENT ON TTY.
@
:UD
MISSING PARAMETER                   <-- System still waiting for
RE-ENTER STATEMENT ON TTY.              Destination Disc
@
:OFF                                <-- Bailed out of DD,X Condition
@
:UD
SUBCHAN=1
LBL=QQQQQ
@
:UD,SYSTEM,0                        <-- Changed Current User Disc to fixed Disc
DISC GEN CODE 1013 NOT SYS GEN CODE 9000 ERR POSS
@
:UD                                 <-- Checked to see if assignment was made
SUBCHAN=0
LBL=SYSTEM                          --> YES !
DISC GEN CODE 1013 NOT SYS GEN CODE 9000 ERR POSS
@
:IN,*                               <-- Entry to unlabel the fixed Disc
DOS LABEL SYSTEM
OK TO PURGE?
YES                                 <-- Told System to purge "SYSTEM" label, Gen. Code
@
:UD
SUBCHAN=0                           <-- Now System has unlabeled fixed Disc
UNLBL
@
```

```
:UD,QQQQQ,1
@
:UD
SUBCHAN=1
LBL=QQQQQ
@
:DD,X
@
:UD,*,0
@
:UD
SUBCHAN=0
LBL=SYSTEM
@
:UD,,0
LBL=SYSTEM
@
:UD
SUBCHAN=0
LBL=SYSTEM
@
:LI,U
MISSING PARAMETER
RE-ENTER STATEMENT ON TTY.
@
:LI,U,1
NAME TYPE SCTRS DISC ORG PROG LIMITS   B.P. LIMITS   ENTRY LIBR.  P-BIT
SUBCHAN=0
@
:SS
@
:LI,U,1
NAME TYPE SCTRS DISC ORG PROG LIMITS   B.P. LIMITS   ENTRY LIBR.  P-BIT
SUBCHAN=0
SUBCHAN=1
XREF  UM  0013  T023 000 12000 14750   01002 01036   12000 14071
ET1   SS  0001  T023 013
WEOT  UM  0002  T023 014 12000 12013   01002 01003   12000 12013

*:OFF
@
:SS,99
@
:LI,U,1
NAME TYPE SCTRS DISC ORG PROG LIMITS   B.P. LIMITS   ENTRY LIBR.  P-BIT
SUBCHAN=1
XREF  UM  0013  T023 000 12000 14750   01002 01036   12000 14071

*:OFF
@
:UD
SUBCHAN=1
LBL=QQQQQ
@
```

:UD,QQQQQ,1 ← Reassigned PACK as current user Disc

:UD ← Checked to make sure assignment made

:DD,X ← Declared System area only Disc dump

:UD,*,0 ← Declared Destination Disc

:UD,,0 ← checked Label on Subchannel # ∅ (Fixed Disc)

:LI,U ← Incorrect Statement Entered

:LI,U,1 ← List USER Directory

NOTHING ON Fixed Disc to be listed

:SS ← Set System Search for all active subchannels

*:OFF ← Bailed out of long listing

:SS,99 ← Reset default System Search

:LI,U,1 ← List User Directory

*:OFF ← Bailed out

Note how Current user Disc is still 1

[SLIDE 20B]

```
SUBCHAN=1
LBL=QQQQQ
@
:DD,U
@
:UD,SYSTEM,0
XREF
EOT1
WEOT
XREFR
DISCM
EXECS
DVR05
DVR31
LIBRY
DVR02
DVR01
DVR22
LODR
JOBP
ASMBL
ASMD
ASM3
ASM4
ASM5
FRTN
FTN1
FTN2
FTN3
FTN4
ASM1
ASM2
SIO1
BASC1
BOOT
FTNH
EOF
FSPCE
RWIND
D.00S
TSRTS
TSRTR
CLEAR
@
:UD
SUBCHAN=0
LBL=SYSTEM
@
:LI,U,1
NAME TYPE SCTRS DISC ORG PROG LIMITS   B.P. LIMITS   ENTRY LIBR.  P-BIT
SUBCHAN=0
XREF   UM  0013  T024 000 12000 14750   01002 01036   12000 14071
EOT1   SS  0001  T024 013
WEOT   UM  0002  T024 014 12000 12013   01002 01003   12000 12013

*:ABORT
JOB ABORTED!
END JOB EXMP1 RUN=0011 MIN. 52.2 SEC.  EXEC=0000 MIN. 00.0 SEC.

@
```

Declared User Area only Disc Dump

Destination Disc

System Reported Each User File Name Transfered as transfer done

List user Directory

Bailed out and aborted

[SLIDE 20C]

OPERATIONAL DIFFERENCE SUMMARY (PART 1)

CONDITION	DOS	DOSM
SYSTEM START-UP	Outputs the following: INPUT FR = FRESH; CO = CONTINUATION	Does not output this message. Outputs INPUT :DATE,XXXXXXXXXX,H,M (H and M are omitted if system does not have Time Base Generation)
:OFF Directive	Does not exit. Must use :ABORT to terminate the current job.	New Directive to abort without terminating current job.
:DD Directive	Does not exist. SDUMP program must be used to create backup copies on Mag. Tape	New Directive to perform disc to disc dumps. Backup copy may be put on cartridge disc.
:SS Directive	Does not exist.	New Directive to enable multi-disc file searching.
:IN Directive	Does not exist. Discs are not labeled.	New Directive to label or unlabel discs.
:UD Directive	Does not exist.	New Directive to change current user disc.
System Recognition of Operator Attention by outputting *	Following are valid entries at this time: :ABORT, :DN, :EQ, :LU, :TYPE, :UP	Following are valid entries at this time: :OFF, :PAUSE, :ABORT, :DN, :EQ, :LU, :TYPE, :UP
:JOB Directive	Current time is always printed on the System Teleprinter and List device along with the job name and date. TBG is System requirement.	Current Time is only printed on the system Teleprinter and list device when Time Base Generator is in system. TBG is an option.

OPERATIONAL DIFFERENCE SUMMARY (PART 2)

CONDITION	DOS	DOSM
:EJOB Directive	Not Applicable	System resets :SS condition to be only standard user disc.
	System condenses User file. Only one User File Area.	System condenses all user discs following :SS condition.
	Not Applicable	User Disc subchannel assignment reset to standard subchannel # unless standard is "NOT READY" or new cartridge has been inserted with different label.
	Message is printed on System Teleprinter and standard List device with job name, execution and run times. TBG is a System Requirement.	Execution and Run times are not printed if the Time Base Generator is not in the system. TBG is optional.
:PROG Directive	Not Applicable Only one user area in system.	File Search for program specified follows :SS condition. User files are searched first, then system files.
:RUN Directive	Optional "time parameter" always used. TBG is System requirement.	"Time parameter" is ignored if Time Base Generator is not in system. TBG is optional.
	Not Applicable. Only one user area in system.	File search for User program follows :SS condition.

OPERATIONAL DIFFERENCE SUMMARY (PART 3)

CONDITION	DOS	DOSM
:TRACKS DIRECTIVE	REQUIRES THAT THE OPERATOR INFORM SYSTEM OF THE FAULTY TRACKS ON A FRESH START-UP FOLLOWING THE DATE DIRECTIVE.	DOES NOT INCLUDE THIS OPTION BECAUSE A RECORD IS MAINTAINED IN THE LABEL SECTOR ON EACH DISC FOR NUMBER OF FAULTY TRACKS, THE ADDRESS OF NEXT SPARE TRACK, ETC...
	REPORTS TRACK NUMBERS THAT ARE FAULTY.	REPORTS TOTAL NUMBER OF TRACKS THAT HAVE BEEN REPLACED BY SPARES.
:STORE DIRECTIVE	CHECKS USER AREA FOR DUPLICATE FILE NAMES.	CHECKS ALL ACTIVE SUBCHANNELS (ACCORDING TO :SS CONDITION) FOR DUPLICATE FILE NAME. STORE ACUALLY DONE ON CURRENT USER DISC.
	ONE SECTOR = 64 WORDS	ONE SECTOR = 128 WORDS
:JFILE DIRECTIVE	SOURCE FILE SPECIFIED IS IN ONE USER AREA.	SOURCE FILE SPECIFIED MAY BE ON ANY ACTIVE SUBCHANNEL (ACCORDING TO :SS CONDITION).
:EDIT DIRECTIVE	SOURCE FILE SPECIFIED IS IN ONE USER AREA.	SOURCE FILE SPECIFIED MAY BE ON ANY ACTIVE SUBCHANNEL (ACCORDING TO :SS CONDITION).
	UPDATED OR NEW SOURCE FILE IS STORED IN ONLY ONE USER AREA.	IF NEW FILE NAME IS SPECIFIED, THIS FILE IS STORED ON SAME SUBCHANNEL AS OLD FILE.
:PURGE DIRECTIVE	FILES SPECIFIED ARE ONLY IN ONE USER	FILES SPECIFIED MAY BE ON ANY ACTIVE SUBCHANNEL (ACCORDING TO :SS CONDITION). ALL ASSOCIATED USER DISCS ARE REPACKED FOR EFFICIENTY. USE :IN,* TO PURGE ALL USER FILES ON A GIVEN USER DISC.

OPERATIONAL DIFFERENCE SUMMARY (PART 4)

CONDITION	DOS	DOSM
:LIST Directive	Dose not have P-BIT field for directory listings.	Has all fields of DOS with additional field, P-BIT. Entry under this field will be "T" to indicate that the associated file is temporary and will be purged at :EJOB if not stored with :STORE.
	User directory is only on one disc.	User directory listing has subchannel numbers preceeding user files on that subchannel.
	Not Applicable	Current user disc subchannel number is restored following :LIST,U.
	Source file specified is on one user area.	Source file specified may be on any active subchannel (according to :SS condition).
:DUMP Directive	User files only on one disc.	File specified may be on any active subchannel (according to :SS condition)
:SA or :SO Directives	Called DISC DUMP	Called SECTOR DUMP to distinguish from :DD (DISC DUMP).
	Dump is to System Teleprinter (LU # 1)	Dump is to standard list device (LU #6).
	Any portion of disc(s) on system may be dumped.	Dump any portion of Current User disc even if user area is on System Disc.
:DATE Directive	HOURS and MINUTES entries are always meaningful. TBG is system requirement	If Time Base Generator is not present in system, HOURS and MINUTES are set to zero.

CHANGE USER DISC EXEC CALL

(GENERAL FORMAT)

PURPOSE

To change the subchannel assignment for the user disc.

ASSEMBLY LANGUAGE

```
      EXT EXEC
       .
       .
       .
      JSB EXEC                  (Transfer control to DOS-M)
      DEF *+3 (or 4)            (Point of return from DOS-M)
      DEF RCODE                 (Request Code)
      DEF LABEL                 (Disc Label)
      DEF SUBCH                 (Disc Subchannel; optional)
      RETURN POINT
       .
       .
       .
RCODE DEC 23                    (Request Code = 23)
LABEL ASC 3, xxxxxx             (Label = xxxxxx)
SUBCH DEC (Ø to 7)
```

FORTRAN

```
      IRCDE = 23
      DIMENSION LABEL (3)
      LABEL (1) = xx
      LABEL (2) = xx
      LABEL (3) = xx
      ICHNL = M      (Ø THROUGH 7)
      CALL EXEC (IRCDE, LABEL, ICHNL)
```

CHANGE USER DISC EXEC CALL -- FORM # 1

(LABEL AND SUBCHANNEL SPECIFIED)

CALLING SEQUENCE:

```
      JSB EXEC          TRANSFER CONTROL TO EXEC
      DEF *+4           DEFINE RETURN POINT
      DEF RCODE         DEFINE REQUEST CODE LOCATION
      DEF LABEL         DEFINE LABEL LOCATION
      DEF SUBCH         DEFINE SUBCHANNEL LOCATION
      (RETURN POINT)
       .
       .
       .
RCODE DEC 23            23 FOR REQUEST CODE
LABEL ASC 3,xxxxxx      6 CHARACTER DISC LABEL OR "*"
SUBCH DEC N             N = Ø-7 FOR SUBCHANNEL #
```

SYSTEM ACTION	OPERATOR ACTION
CHECKS IF SUBCHANNEL N IS LABELED AS SPECIFIED IN CALL (LABEL NAME OR "*")	NONE REQUIRED
MATCH - MAKES ASSIGNMENT AND RETURNS	NONE REQUIRED
NO MATCH - PRINTS MESSAGE: LBL = (LABEL NAME FOUND ON SUBCHANNEL N) OR UNLBL IF "*" xxxxx SUSP WHERE xxxxx IS NAME OF EXECUTING PROGRAM.	1. IF CORRECTLY LABELED DISC ON HAND: MOUNT IN DRIVE AND "READY" DRIVE. THEN ENTER :GO FOR SYSTEM TO EXECUTE AT START OF EXEC CALL. OR 2. IF NO PROPERLY LABELED DISC ON HAND: ENTER: :ABORT OR :OFF

CHANGE USER DISC EXEC CALL -- FORM # 2

(ONLY LABEL SPECIFIED)

CALLING SEQUENCE:

```
      JSB EXEC          TRANSFER CONTROL TO EXEC
      DEF *+3           DEFINE RETURN POINT
      DEF RCODE         DEFINE REQUEST CODE LOCATION
      DEF LABEL         DEFINE LABEL LOCATION
      (RETURN POINT)
       .
       .
       .
RCODE DEC 23            23 FOR REQUEST CODE
LABEL ASC 3,xxxxxx      6 CHARACTER LABEL OR "*"
```

SYSTEM ACTION	OPERATOR ACTION
SEARCHES FOR LABEL OR "*" DISC STARTING WITH THE HIGHEST SUBCHANNEL NUMBER.	NONE REQUIRED
MATCH - MAKES ASSIGNMENT AND RETURNS	NONE REQUIRED
NO MATCH - PRINTS MESSAGE: DISC NOT ON SYST xxxxx SUSP WHERE xxxxx IS NAME OF EXECUTING PROGRAM.	1. IF PROPERLY LABELED DISC ON HAND: MOUNT IN DRIVE AND "READY" DRIVE. THEN ENTER :GO FOR SYSTEM TO EXECUTE AT START OF EXEC CALL. OR 2. IF NO APPROPRIATELY LABELED DISC ON HAND: ENTER: :ABORT OR :OFF

DOS/DOSM GENERAL PURPOSE EXEC CALLS WITH NEGATIVE REQUEST CODES

REQUEST CODE	FUNCTION	CALLING SEQUENCE
-19	BASE PAGE STORE (STA B,I)	LDA "VALUE TO STORE" LDB "DESTINATION ADDRESS" JSB EXEC DEF *+2 DEF RCODE (RETURN WITH B = FORMER VALUE +1) . . . RCODE DEC - 19
-20	TO LOAD AND START EXECUTION OF A PROGRAM WHOSE DIRECTORY ENTRY IS IN LOCATIONS 141-153 OCTAL	STORE DIRECTORY ENTRY OF DESIRED PROGRAM IN LOCATIONS 141-153B JSB EXEC DEF *+2 DEF RCODE (RETURN) . . . RCODE DEC - 20
-21	TO INITIALIZE TBG (IF IN SYSTEM) FOR .1 SECOND TIMED INTERRUPTS	JSB EXEC DEF *+2 DEF RCODE (RETURN) . . . RCODE DEC - 21
-22	TO EXECUTE AN I/O INSTRUCTION	LDA "I/O INSTRUCTION" JSB EXEC DEF *+2 DEF RCODE (RETURN) . . . RCODE DEC - 22

DOSM DISC I/O WITH EXEC CALLS

(ABSOLUTE DISC ADDRESSING)

GENERAL CALLING SEQUENCE:

```
      JSB EXEC   (TRANSFER TO EXEC)
      DEF RTN    (DEFINE RETURN ADDRESS)
      DEF RCODE  (SEE BELOW)
      DEF CNTLW  (SEE BELOW)
      DEF BUFFR  (DEFINE BUFFER ADDRESS)
      DEF BUFFL  (DEFINE BUFFER LENGTH)
      DEF TRCK   (DEFINE TRACK #)
      DEF SECT   (DEFINE SECTOR #)
RTN   (RETURN POINT)
```

RCODE	CNTLW	DISC AREA ADDRESSED
+1 (READ) +2 (WRITE)	2	"WORK AREA" ON SYSTEM DISC ONLY. SYSTEM CHECKS FOR LEGALITY OF TRCK/SECT IN CALL.
	3 → Avail. Area on CUD.	
-1 (READ) -2 (WRITE)	2	"ANY AREA" ON SYSTEM DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.
	3	"ANY AREA" ON CURRENT USER DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.
	-3	"ANY AREA" ON CURRENT JOB FILE (JFILE) DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.

starting with next avail JK/SEC on CUD. on ISS ...

NOTE: SYSTEM WILL HALT (WITH T-REG. =102031 OCTAL) IF DISC PROTECT OVERRIDE SWITCH IS "OFF" (DOWN) AND REQUEST MADE TO WRITE ON A SECTOR THAN IS FLAGGED PROTECTED (PCI=1).

More on NEG. REQUEST CODES [Relating to Disc]

RCODE = (+1, +2) [USER]

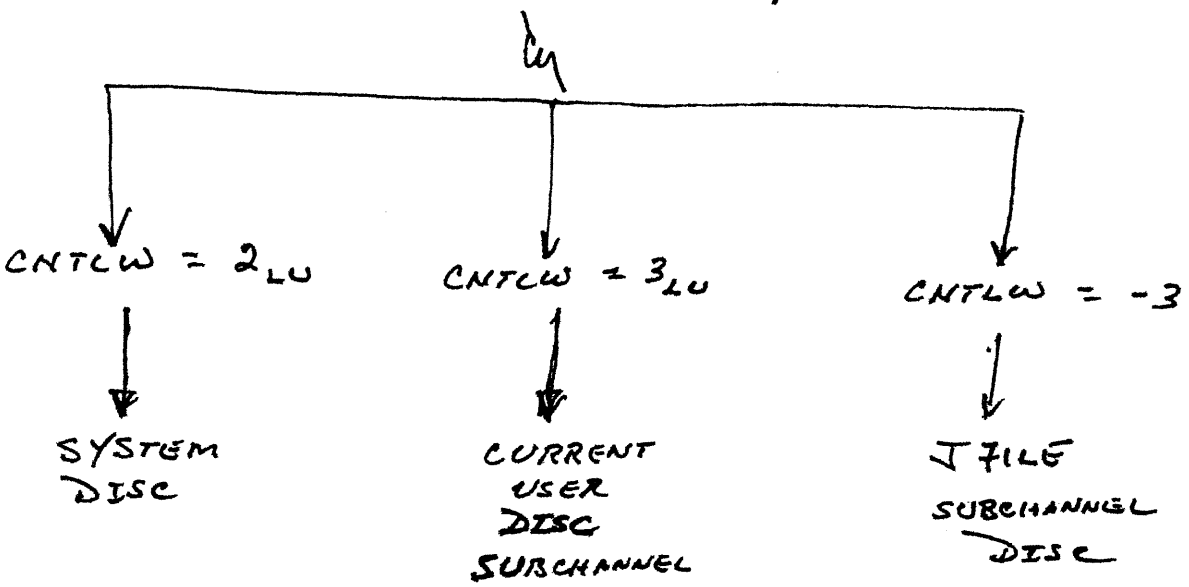
positive

with LU = 2 (DISC) or CNTLW

user may write or read WORK AREA

on System Disc.

RCODE (-1, -2) [SYSTEM]



calling sequence →

```
JSB EXEC
DEF RTN
DEF RCODE   (Request Code)
DEF CNTLW   (Control Word)
DEF BUFFR   (Buffer Address)
DEF BUFFL   (Buffer Length) — words
DEF DTRK    (Disc TRACK ADDRESS)
DEF DSECT   (Disc SECTOR ADDRESS)
RTN
```

[SLIDE 23B]

LARGE TYPE

DOSM DISC I/O WITH EXEC CALLS

(ABSOLUTE DISC ADDRESSING)

GENERAL CALLING SEQUENCE:

CENTER

```
JSB EXEC      (Transfer to EXEC)
DEF RTN       (Define Return Address)
DEF RCODE     (See Below)
DEF CNTLW     (See Below)
DEF BUFFR     (Define Buffer Address)
DEF BUFFL     (Define Buffer Length)
DEF TRCK      (Define Track #)
DEF SECT      (Define Sector #)
RTN (Return Point)
```

RCODE	CNTLW	DISC AND AREA ADDRESSED
+1 (READ) +2 (WRITE)	2	"WORK AREA" ON SYSTEM DISC ONLY. SYSTEM CHECKS FOR LEGALITY OF TRCK/SECT IN CALL
~~-1 (READ)~~ ~~-2 (WRITE)~~ -1 (READ) -2 (WRITE)	2	"ANY AREA" ON SYSTEM DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.
	3	"ANY AREA" ON CURRENT USER DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.
	-3	"ANY AREA" ON CURRENT JOB FILE (JFILE) DISC. SYSTEM DOES NOT CHECK FOR LEGALITY OF TRCK/SECT IN CALL.

NOTE: SYSTEM WILL HALT (WITH T-REG. = 102031 octal) IF DISC PROTECT OVERRIDE SWITCH IS "OFF" (DOWN) AND REQUEST MADE TO WRITE ON A SECTOR THAT IS FLAGGED PROTECTED (PCI = 1).

EXEC CALLS DIFFERENCE SUMMARY

EXEC CALL	DOS	DOSM
EXECUTION OF "JSB EXEC" INSTRUCTION	CAUSES MEMORY PROTECT VIOLATION INTERRUPT. LOCATION 5 EXECUTED WHICH IS JSB $CIC. $CIC THEN TRANSFERS CONTROL TO EXEC.	SAME AS DOS WHEN MEMORY PROTECT OPTION IS USED. IF NO MP OPTION, EXEC IS ENTERED DIRECTLY WHEN JSB EXEC IS EXECUTED.
FILE READ/WRITE	FILE IS REFERENCED WITH RESPECT TO ONLY ONE USER AREA ON DISC(S).	:SS CONDITION IS FOLLOWED IN SEARCHING FOR REFERENCED FILE. IF UNSURE AS TO PRESENCE OF CORRECT USER DISC, SHOULD FIRST USE CHANGE USER DISC EXEC CALL WITH ONLY LABEL SPECIFIED.
WORK AREA LIMITS	WORK AREA ALWAYS ON SAME DISC.	WORK AREA IS ON SYSTEM DISC THAT IS ACTIVE (I.E., BOOTSTRAPPED IN)
	PROGRAM MUST FIRST CALL WORK AREA STATUS EXEC CALL IF CONSECUTIVE TRACKS ARE REQUIRED BECAUSE SOME TRACKS WITHIN WORK AREA MAY BE FAULTY.	PROGRAM DOES NOT HAVE TO WORRY ABOUT FAULTY TRACKS IN THE WORK AREA; SYSTEM HANDLES AUTOMATICALLY. NEED ONLY USE WORK AREA STATUS CALL TO CHECK IF ENOUGH ARE AVAILABLE
PROGRAM SEGMENT LOAD	USER AND SYSTEM DIRECTORY ONE AREA OF DISC(S).	CURRENT USER DISC AND SYSTEM DIRECTORIES SEARCHED FOR REFERENCED SEGMENT.
SEARCH FILE NAMES	ONE USER AND SYSTEM DIRECTORY TO SEARCH.	:SS CONDITION FOLLOWED WHEN USER DIRECTORIES ARE SEARCHED
TIME REQUEST	TIME SET IN TIME ARRAY IS ALWAYS AVAILABLE.	IF TIME BASE GENERATOR NOT IN SYSTEM, ALL VALUES IN TIME ARRAY ARE SET TO Ø.

IV. INSTALLATION

A. Introduction to System Generation.

1. Binary tapes needed [SLIDES 3ØA, 3ØB].

2. Preliminary Considerations

   a. Medium of Input

      (1) Paper tape.

      (2) Magnetic tape (restrictions with FORTRAN IV in 8K).

      (3) Combination

   b. Core size [SLIDE 31] -- GENERAL (Projecture #1)

      (1) Speed of System Operation needed.

      (2) Core Resident versus Disc Resident EXEC Modules and I/O Drivers.

      (3) System Modules size breakdown [SLIDE 32] (Projecture #2).

         a. Minimum System Analysis using slides 31 and 32.

   c. Particular needs for given application.

   d. System and User Disc subchannel declaration.

      (1) More efficient (time wise) if System and User Discs are on different drives, depending on what System is doing.

   e. Other System Discs considerations.

      (1) If both generated so that linkage and DISCM are in same place, then user main programs LOADED on one system would "RUN" and be compatable with the other. Location of EXEC entry in DISCM must be same; # links must be enough; Etc...

f. Hardware required for Generation.

3. Starting System Generator

a. SIO Configuration - loading Generator.

b. All equipment to be used "ON".

c. Disc Drive for Generated system "READY".

d. Disc Protect Override Switch "ON".

e. Starting Generator (S.A. = 100 octal).

(1) Switch 15 DOWN for Straight Generation.

(2) Switch 15 UP only for User Disc Formatting.

4. Brief description of the four PHASES and ability to restart at any one at 100 octal.

B. System Generation Procedure and Example

1. Initialization Phase [SLIDE 40A].

a. Responses to questions about the System (in general) to be generated.

b. System Generation Code - maximum 4 decimal positive digits. Written in Label Sector of System Disc.

c. # Sectors/Track - Actually # Sectors/Physical track which is 12 for low density disc.

d. System Disc Size - actually # cylinders.

e. First System Sector - System uses first 3 sectors on track 0 of System Disc.

f. 2114 question - only for DMA considerations (only one DMA Channel available on 2114B).

g. Program Input, Library Input questions - Unimportant whether PT or MT entered here
(1) MT may not be used for FORTRAN IV in 8K.
(2) DF also valid entry for disc file input. (SIO driver for IOMEC available later).

h. Parameter Input question - only applies to PARAMETER INPUT PHASE.

3. Program Input and Parameter Input Phases [SLIDE 40B].

a. Input device selected via S.R. switches Ø-1.

$ØØ_2$ - PROGRAM INPUT
$1Ø_2$ - LIBRARY INPUT
$Ø1_2$ - TERMINATE LOADING

b. Restrictions

(1) DISCM should be loaded first for intersystem compatability.

(2) Main Programs (like FTN) must be loaded prior to segments (like FTNØ1, FTNØ2, ... etc.)

(3) If generating 8K system with FORTRAN IV no Compilers, or Assembler may be loaded at this time. (Must be loaded using LOADR during System operation).

c. If undefined externals exist (message printed), may load module forgot by setting S.R. accordingly as in a. above and pressing "RUN".

d. During PARAMETER INPUT PHASE, be sure to declare other routines ($SRCH, $LBL, etc...) core resident too if certain EXEC modules are declared core resident. Generator will not flag if omitted.

e. LINKAGE QUESTIONS

(1) #SYSTEM LINKS - only used by Core Resident System.

(2) #USER LINKS - only used by User Programs

(3) To make DISCM start at 2ØØØ octal (page boundary) respond with 177 and 5ØØ respectively.

f. Switch 15 must be up for Subroutines (indented two spaces) and entry points (preceeded by "*") to be printed in memory allocation listing.

3. Disc Loading Phase (class follows Xerox of Generation).

TOPIC	[SLIDE] MAIN PROJECTOR	[SLIDE] AUXILIARY PROJECTO
a. Links	AP-1	
b. Loc. 4-Start of Links	AP-1	AP-6
c. Core Res Prog & Links	AP-1	AP6-13
d. Equip. Table	AP-1,AP-2	AP-14
e. DRT + Int. Tables	AP-1,AP-3	AP-14
f. Disc Res. Exec Mod.---TABLE	AP-1,AP-4	AP-14
g. Disc Res Exec. Mod.	AP-1	AP-5
h. Disc Res. I/O Drivers	AP-1	AP-5
i. Disc Res User Prog.	AP-1	AP-5

j. Value of A-Reg. at end (do on slide 40J)

k. Listing of :EQ & :LU [SLIDE AP-5]

C. Formatting User Discs or Cartridges [SLIDE 41].

1. Example printout [SLIDE 42].

BINARY TAPES NEEDED FOR SYSTEM GENERATION (PART 1)

PROGRAM(S)	# TAPES	COMMENTS
SYSTEM GENERATOR	1	OPERATES IN SIO ENVIRONMENT, THEREFORE THE FOLLOWING SIO DRIVERS MAY BE NEEDED: TELETYPE, PHOTOREADER, PUNCH (IF SIO DUMP TO BE USED), AND MAGNETIC TAPE. GENERATOR CONTAINS DISC I/O DRIVER INTERNAL TO ITSELF, THEREFORE, NO SIO DISC DRIVER NEEDED.
DISC MONITOR (CORE RESIDENT SYSTEM)	1	ALWAYS MADE CORE RESIDENT BY GENERATOR. GOOD PRACTICE TO LOAD AS FIRST PROGRAM FOR SYSTEM COMPATABILITY BETWEEN PROGRAMS LOADED ON OTHER SYSTEMS.
EXECUTIVE MODULES AND SUBROUTINES	1	MUST BE INCLUDED IN SYSTEM GENERATION. CONTAINS $EX01 - $EX20 EXEC MODULES AND SUBROUTINES $LBL, $SRCH, $ADDR, ASCII, DUMRX. ALL EXEC MODULES ARE PROGRAM TYPE 1 (SYSTEM DISC RESIDENT), BUY MAY BE MADE SYSTEM CORE RESIDENT DURING GENERATION (PROGRAM TYPE 0). CAUTION: IF CERTAIN EXEC MODULES ARE MADE SYSTEM CORE RESIDENT, THEIR ASSOCIATED SUBROUTINES MUST ALSO BE DECLARED CORE RESIDENT.
I/O DRIVERS	1 PER DRIVER	DVR31 (IOMEC DISC DRIVER) MUST ALWAYS BE INCLUDED. IT IS DECLARED PROGRAM TYPE 0 (SYSTEM CORE RESIDENT) AND MUST NOT BE REDECLARED. DVR05 DVR00 ONE OF THESE DRIVERS MUST BE INCLUDED. BOTH ARE DECLARED PROGRAM TYPE 0 (SYSTEM CORE RESIDENT). THE ONE TO BE USED AS SYSTEM TELETYPE MUST NOT BE REDECLARED AS DISC RESIDNET. DVR05 IS SHORTER IN CORE REQUIREMENTS. ALL OTHER DRIVERS ARE DECLARED PROGRAM TYPE 4 (DISC RESIDENT I/O DRIVER) AND MAY BE REDECLARED PROGRAM TYPE 0 IF DESIRED.
JOB PROCESSOR	1	MUST ALWAYS BE INCLUDED IN GENERATION. MUST ALWAYS BE DISC RESIDENT.
RELOCATING LOADER	1	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION, BUT IF NOT INCLUDED NO PROGRAMS COULD BE RELOCATED INTO CORE IMAGE ABSOLUTE FORM BY THE SYSTEM THAT IS GENERATED. DECLARED PROGRAM TYPE 3 (USER MAIN) AND MAY NOT BE MADE CORE RESIDENT. MUST ALWAYS BE DISC RESIDENT.

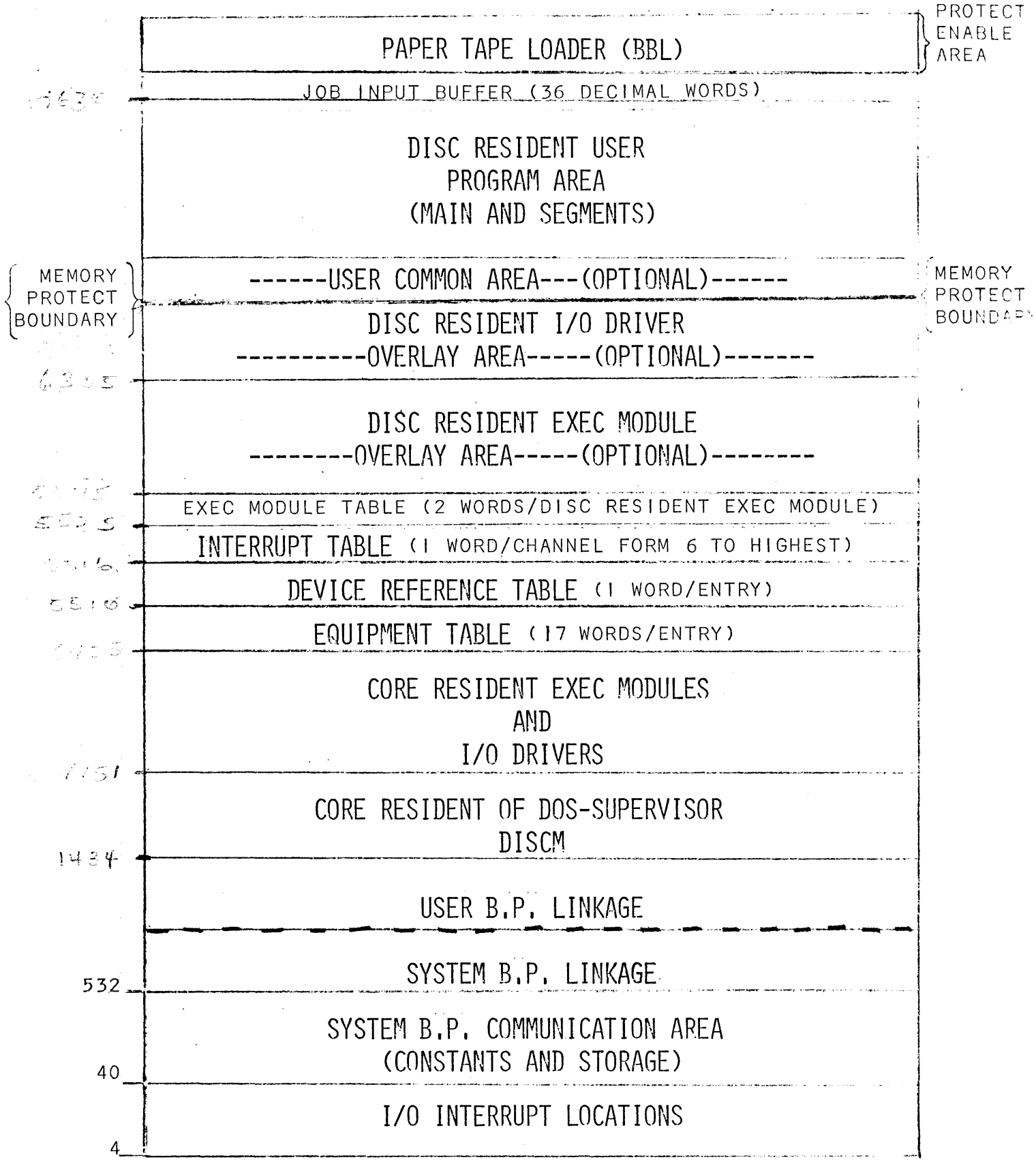
BINARY TAPES NEEDED FOR SYSTEM GENERATION (PART 2)

PROGRAM(S)	# TAPES	COMMENTS
EXTENDED ASSEMBLER	7	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION. ONE TAPE IS THE MAIN PROGRAM (TYPE 3) AND SIX TAPES ARE SEGMENTS (PROGRAM TYPE 5). THE MAIN PROGRAM (ASMB) MUST BE LOADED PRIOR TO ITS SEGMENTS. MUST ALWAYS BE DISC RESIDENT.
HP BASIC FORTRAN COMPILER	5	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION. ONE TAPE IS THE MAIN PROGRAM (TYPE 3) AND FOUR TAPES ARE SEGMENTS (PROGRAM TYPE 5). THE MAIN PROGRAM (FTN) MUST BE LOADED PRIOR TO ITS SEGMENTS. MUST ALWAYS BE DISC RESIDENT.
HP ALGOL COMPILER	2	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION. ONE TAPE IS THE MAIN PROGRAM (TYPE 3) AND ONE SMALL TAPE IS THE ONLY SEGMENT (PROGRAM TYPE 5). THE MAIN PROGRAM (ALGOL) MUST BE LOADED PRIOR TO THE SEGMENT. MUST ALWAYS BE DISC RESIDENT. REQUIRES 16K MINIMUM CORE.
HP FORTRAN IV COMPILER	19	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION. ONE TAPE IS THE MAIN PROGRAM (TYPE 3) AND 18 OTHER TAPES ARE ITS SEGMENTS (PROGRAM TYPE 5). THE MAIN PROGRAM (FTN4) MUST BE LOADED PRIOR TO THE 18 SEGMENTS. MUST ALWAYS BE DISC RESIDENT.
CROSS REFERENCE TABLE GENERATOR	1	DOES NOT HAVE TO BE INCLUDED IN SYSTEM GENERATION. DELARED PROGRAM TYPE 3 (USER DISC RESIDENT MAIN). MUST BE DISC RESIDENT.
LIBRARIES	5	THE LIBRARIES INCLUDED DURING SYSTEM GENERATION WILL DEPEND ON THE PARTICULAR SYSTEM THAT IS BEING GENERATED AND WILL VARY ACCORDINGLY. FACTORS THAT WILL HELP DETERMINE ARE: 1. IS EAU TO BE USED. 2. IS FORTRAN IV COMPILER TO BE INCORPORATED INTO SYSTEM. 3. IS PLOTTING EQUIPMENT TO BE USED.
ANY USER PROGRAMS TO BE MADE A PERMANENT PART OF SYSTEM	?	SAME CONVENTIONS MUST BE FOLLOWED IN SEGMENTATION. USER MAIN MUST BE LOADED PRIOR TO SEGMENTS ETC. LIBRARY PROGRAMS MUST BE DECLARED TYPE 6 OR 7.

NOTE: IF THE FORTRAN IV LIBRARY IS TO BE INCLUDED IN AN 8K SYSTEM, CERTAIN RULES MUST BE FOLLOWED:

1. THE SYSTEM MUST BE GENERATED WITHOUT ANY COMPILERS OR AN ASSEMBLER.
2. A MAGNETIC TAPE SIO DRIVER CANNOT BE USED WITH DSGEN.
3. THE COMPILERS AND ASSEMBLER MUST BE LOADED INTO THE SYSTEM DURING OPERATION (USING THE LOADER).

DOSM GENERAL CORE LAYOUT



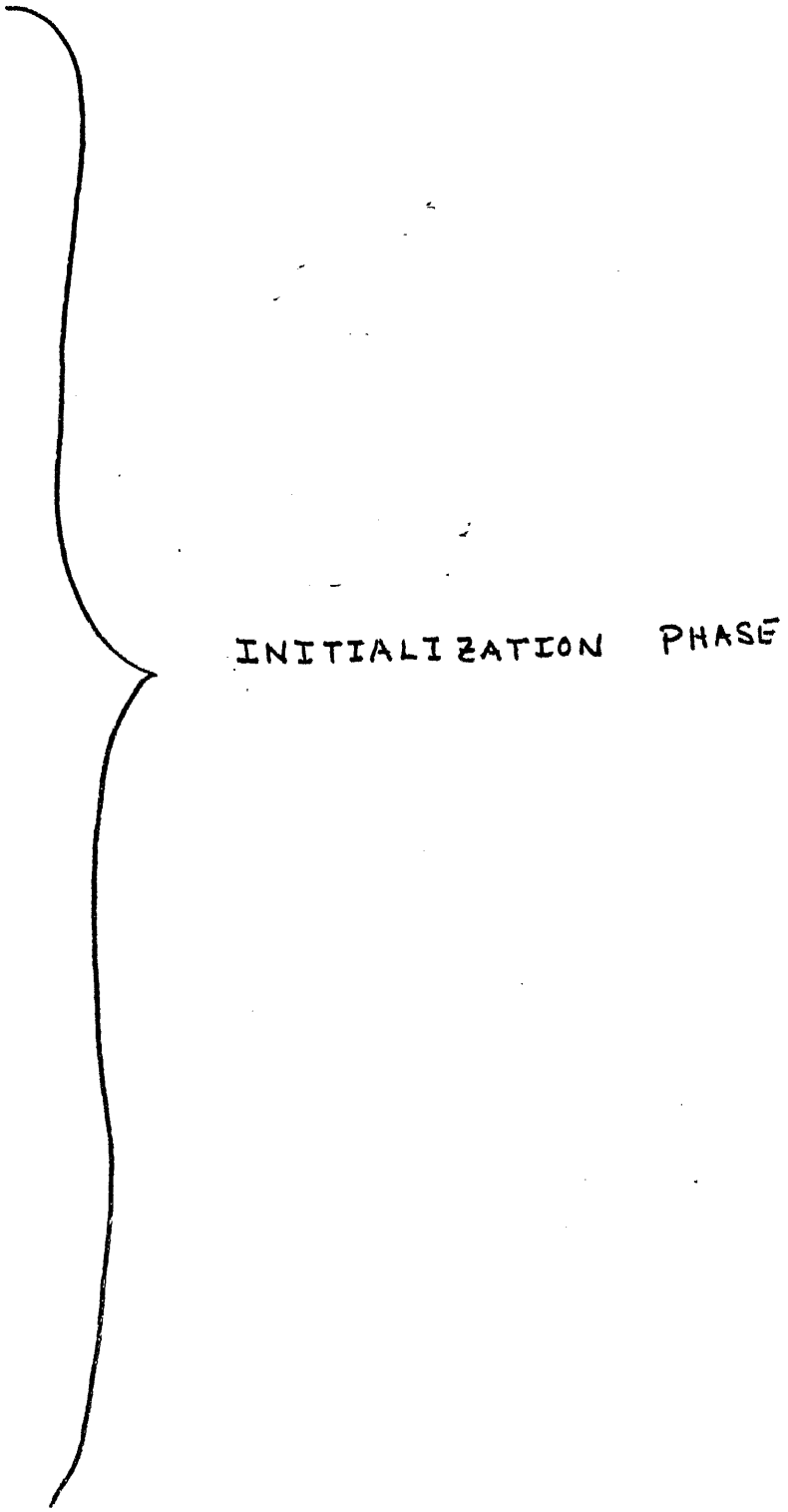
[SLIDE 31]

SYSTEM SOFTWARE SIZE BREAKDOWN ("A" VERSIONS)

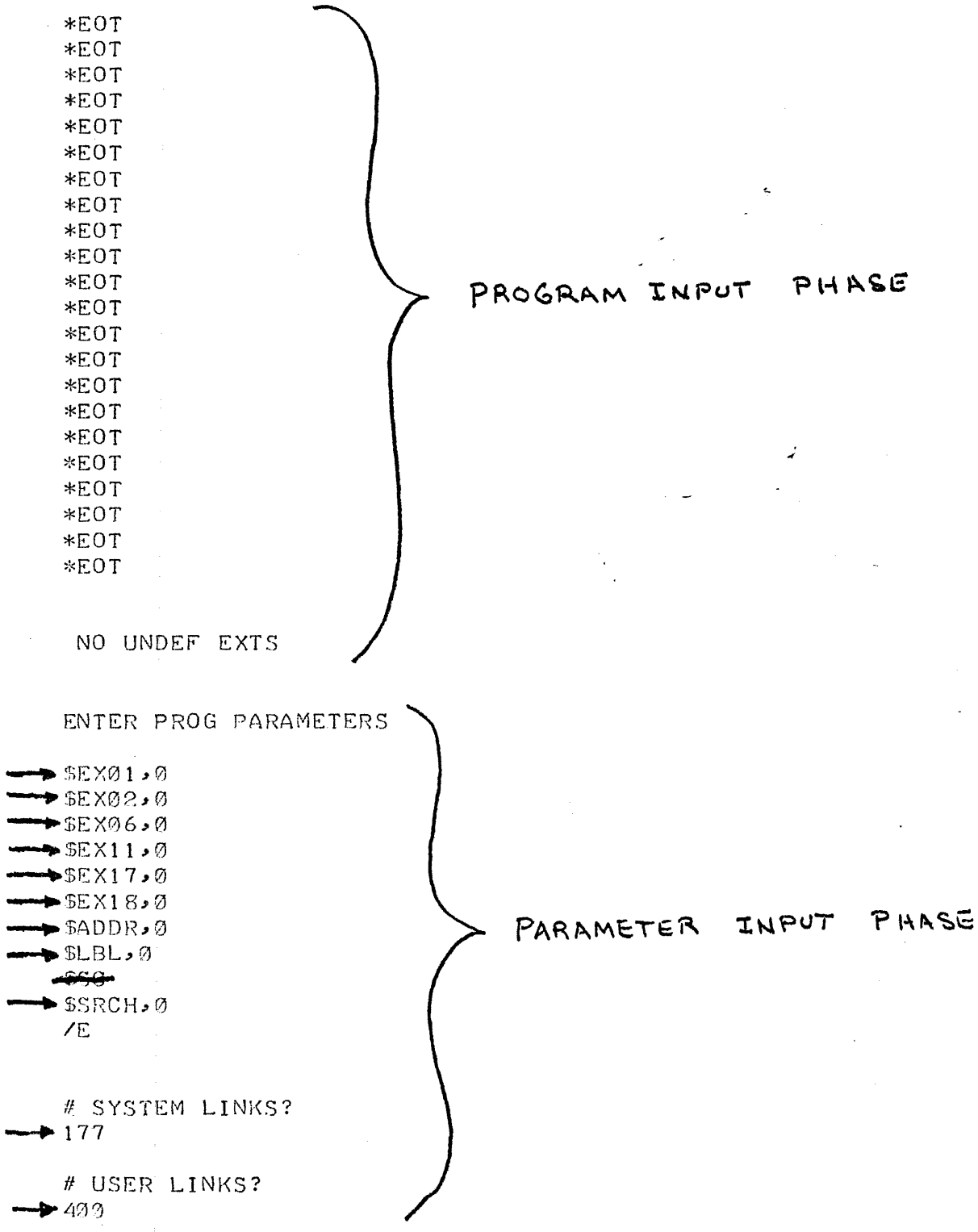
PROGRAM NAME	LENGTH (OCTAL)	LENGTH (DECIMAL)	EXTERNAL ROUTINES
DISCM	2515	1357	-----
$EXØ1	62	5Ø	$ADDR
$EXØ2	5Ø	4Ø	$ADDR
$EXØ3	35	29	-----
$EXØ4	315	2Ø5	ASCII
$EXØ5	156	11Ø	$SRCH
$EXØ6	37	31	$ADDR, $SRCH
$EXØ7	157	111	$ADDR
$EXØ8	143	99	$ADDR
$EXØ9	261	177	ASCII
$EX1Ø	156	11Ø	-----
$EX11	164	116	$SRCH
$EX12	172	122	-----
$EX13	342	226	ASCII
$EX14	36Ø	24Ø	ASCII
$EX15	272	186	ASCII
$EX16	133	91	-----
$EX17	373	251	$LBL
$EX18	51Ø	328	-----
$EX19	32Ø	2Ø8	$LBL
$EX2Ø	3Ø6	198	ASCII
$LBL	73	59	
$SRCH	3Ø4	196	
$ADDR	15	13	
ASCII	72	58	
DUMRX	64	52	
DVRØØ	553	363	
DVRØ1	314	2Ø4	
DVRØ2	2Ø2	13Ø	
DVRØ5	25Ø	168	
DVR1Ø	135	93	
DVR12	527	343	
DVR15	325	213	
DVR22	634	412	
DVR23	566	374	
DVR3Ø	252	17Ø	
DVR31	5Ø1	321	
JOBPR	1Ø463	4403	
LOADR	7Ø32	3610	

DOSM SYSTEM GENERATION EXAMPLE
("A" VERISION TAPES USED)

```
    SYS GEN CODE?
--> 9000

    SYS DISC CHNL?
--> 14

    # SECTORS/TRACK?
--> 12

    SYS DISC SIZE?
--> 200

    # DRIVES?
--> 1

    FIRST SYSTEM TRACK?
--> 0
    FIRST SYSTEM SECTOR?
--> 3

    SYS DISC SUBCHNL?
--> 1

    USER DISC SUBCHNL?
--> 1

    TIME BASE GEN CHNL?
--> 12

    IS 2114?
--> NO

    LWA MEM?
--> 37677

    ALLOW :SS?
--> YES

    PRGM INPT?
--> PT

    LIBR INPT?
--> PT

    PRAM INPT?
--> TY
```



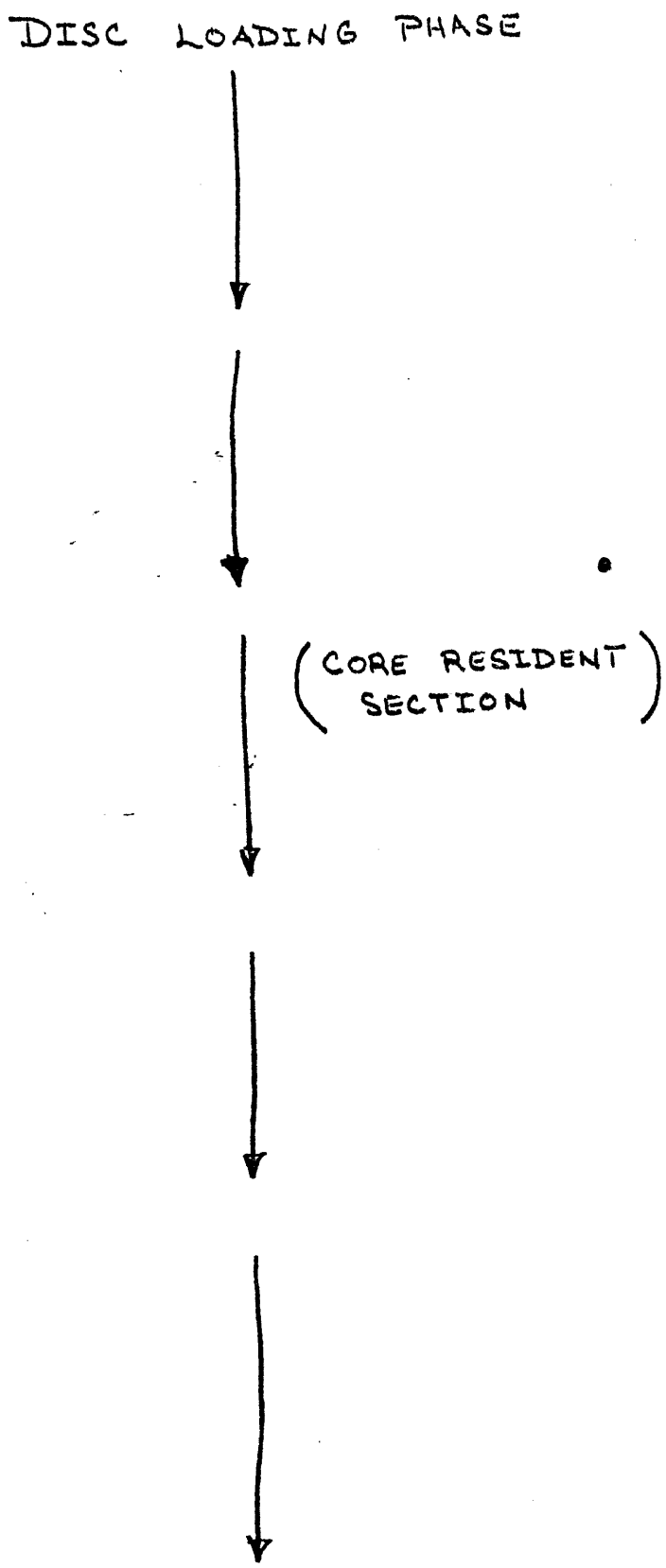
[SLIDE 40A]


*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
*EOT
NO UNDEF EXTS
PROGRAM INPUT PHASE
ENTER PROG PARAMETERS
$EX01,0
$EX02,0
$EX06,0
$EX11,0
$EX17,0
$EX18,0
$ADDR,0
$LBL,0
$SG
$SRCH,0
/E
SYSTEM LINKS?
177
USER LINKS?
400
PARAMETER INPUT PHASE

[SLIDE 40B]

SYSTEM

DISCM	01634
*$CIC	01634
*$STRT	04160
*$LDEX	04154
*EXEC	02214
*$DISC	03400
*$IDL1	03511
*$MDLD	03020
*$RQER	02635
*$JLOD	03744
*$MOVE	03361
*$TYPE	03426
*$SYIO	04165
*$BFND	04267
*$EFAD	02507
*$ABRT	04010
*$WAIT	03244
*SETEQ	03345
*$BLOP	03177
*$CIC3	01714
*SAVE$	03563
*$CLER	04044
*$OPER	02212
*ERR01	02653
*ERR03	02657
*ERR04	02661
*ERR05	02663
*ERR06	02665
*LUCHK	02553
*$DMA	03214
*$MBSY	03207
*$LDVR	03143
*RQEQT	02576
*DRIVR	02700
*ERRTN	02631
*IO.40	02521
*$GDTK	03130
*DISCX	03117
*..RRL	04243
*DEF04	02445
*DEF19	04302
*DEF20	04303
$EX01	04314
*$EX01	04314
$EX02	04376
*$EX02	04376
$EX06	04446
*$EX06	04446
$EX11	04505
*$EX11	04505
$EX17	04671



[SLIDE 40C]

DISC LOADING PHASE (CON'T)

```
   *$EX17   04671

$EX18       05264
   *$EX18   05264

$ADDR       05774
   *$ADDR   05774

$SRCH       06011
   *$SRCH   06011
   *$CMPR   06266

$LBL        06315
   *LBLIO   06354
   *ISLBL   06336
   *LBLMV   06330
   *CHSUM   06315
   *MESSG   06372

DVR05       06410
   *I.05    06410
   *C.05    06464

DVR31       06660
   *I.31    06660
   *C.31    06743
```

(CORE RESIDENT SECTION)

```
* EQUIPMENT TABLE ENTRY

11,DVR05,R
13,DVR01
14,DVR31,R,D
16,DVR02
22,DVR22,D
/E
```

BUILD EQT TABLE

```
* DEVICE REFERENCE TABLE

  1 = EQT #?
1
  2 = EQT #?
3
  3 = EQT #?
3
  4 = EQT #?
4
  5 = EQT #?
2
  6 = EQT #?
1
  7 = EQT #?
5
  8 = EQT #?
/E
```

BUILD DRT TABLE

```
* INTERRUPT TABLE

11,1
13,2
15,3
16,4
23,5
/E
```

BUILD INTERRUPT TABLE

EXEC SUPERVISOR MODULES

```
$EX03       07567
   *$EX03   07567

$EX04       07567
   *$EX04   07567
  ASCII     10104
   *CNDEC   10104
   *CNOCT   10110

$EX05       07567
   *$EX05   07567

$EX07       07567
   *$EX07   07567

$EX08       07567
   *$EX08   07567

$EX09       07567
   *$EX09   07567
  ASCII     10050
   *CNDEC   10050
   *CNOCT   10054

$EX10       07567
   *$EX10   07567

$EX12       07567
   *$EX12   07567

$EX13       07567
   *$EX13   07567
  ASCII     10131
   *CNDEC   10131
   *CNOCT   10135

$EX14       07567
   *$EX14   07567
  ASCII     10147
   *CNDEC   10147
   *CNOCT   10153

$EX15       07567
   *$EX15   07567
  ASCII     10061
   *CNDEC   10061
   *CNOCT   10065
```

DISC LOADING PHASE (CON'T)

(DISC RESIDENT SECTION)

[SLIDE 40E]

```
$EX16       07567
   *$EX16   07567

$EX19       07567
   *$EX19   07567

$EX20       07567
   *$EX20   07567




  ASCII     10075
   *CNDEC   10075
   *CNOCT   10101


I/O DRIVER MODULES

DVR01       10241
   *I.01    10241
   *C.01    10313

DVR02       10241
   *I.02    10241
   *C.02    10320

DVR22       10241
   *I.22    10241
   *C.22    11001


LWA SYS  11075


FWA USER?
→12000

USER SYSTEM PROGRAMS

LOADR       12000
   *LOADR   12000

JOBPR       12000
   *JOBPR   12000

ASMB        12000
   *ASMB    16522
   *?ASCN   13700
   *?ASMB   12554
   *?BNCN   14510
   *?BPKU   15326
   *?CHOP   12646
   *?CHPI   15610
   *?DCOD   15616
   *?ENDS   15230
   *?ERPR   15150
```

DISC LOADING PHASE (CON'T)

(DISC RESIDENT SECTION)

[SLIDE 40F]

```
*?MSYS   15667
*?GETC   15654
*?MOVE   13437
*?MSYM   14775
*?RLUN   16375
*?AFLG   16430
*?LSTL   14717
*?LUNI   16436
*?RFLG   16425
*?Z      16446
*?ASM1   13371
*?LABE   13407
*?OKOL   15307
*?ORRP   14603
*?PNLE   16443
*?SETM   15674
*?SUP    15303
*?LPER   15306
*?PERL   15271
*?LOUT   15336
*?LTFL   15275
*?DRFL   16433
*?LTSA   15560
*?LTSB   15561
*?ORGS   15301
*?CNTR   15370
*?TSTR   16434
*?ASII   16452
*?ICSA   15146
*?FLGS   16422
*?BFLG   16423
*?LFLG   16424
*?TFLG   16426
*?X      16445
*?MESX   12505
*?ASCI   16451
*?LINC   15110
*?LINS   14765
*?LIST   14653
*?LUNP   16440

*?OPLK   12600
*?OPER   15640
*?PKUP   15321
*?PLIT   15406
*?PNCH   13632
*?PRNT   15033
*?RSTA   13105
*?LWA    16444
*?RDSC   16401
*?WEOF   16021
*?WRIF   16102
*?LGFL   16432
*?SEGM   12541
*?SYMK   13506
*?V      15633
```

DISC LOADING PHASE (CON'T)

(DISC RESIDENT SECTION)

[SLIDE 40G]

```
    *?ARTL   15472
    *?LST    15274
    *?PLIN   16435
    *?PCOM   15112
    *?SECT   16420
    *?NEAU   12443
    *?HA38   15347
    *?XRFI   12540

ASMBD        17120
    *ASMBD   17442

ASMB1        17120
    *ASMB1   17366
    *?LITI   20030
    *?CMQ    17560
    *?INSR   17726
    *?HA3Z   17527
    *?ENP    17662
    *?EXP    17645

ASMB2        17120
    *ASMB2   17351
    *?ART    20021
    *?BREC   17475
    *?LKLI   20535
    *?SKPR   17441
    *?SPCR   17444

ASMB3        17120
    *ASMB3   17630

ASMB4        17120
    *ASMB4   17366
    *?INS?   17541

ASMB5        17120
    *ASMB5   17351

FTN          12000
    *%WLIC   13042
    *%FTN0   12000
    *%WPRN   12735
    *%ERRR   12701
    *%RDIS   12557

    *%WDIS   12244
    *%SEGN   12224
    *%WTRA   12236
    *%WSEC   12237
    *%RTRA   12347
    *%RSEC   12350
    *%RBFA   12352
    *%LUNO   12203
    *%LUNI   12204
```

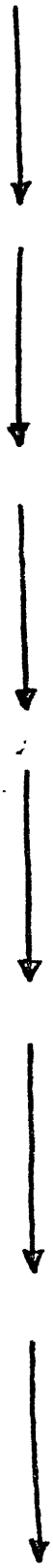
DISC LOADING PHASE (CON'T)

(DISC RESIDENT SECTION)

[SLIDE 40H]

```
   *%LUNP   12205
   *%NAMA   12226
   *%RTYP   12227
   *%WLIN   13017
   *%WPAG   13043
   *%TILT   13074
   *%RDSI   13044
   *%WDSI   13056
   *%WOUT   12334
   *%RBFW   12623
   *%LABL   12733
   *%CONA   12734
   *%ENDP   13105
   *%WDLU   12241
   *%RDLU   12346
   *%RFLG   12554
   *%WBFW   12341
   *%WBFA   12232
   *%HEDN   13010
   *%DUP8   12373
   *%NXDV   12402
   *%NELM   12352
   *%STYP   12433
   *%LGO    12202

FTN01       13127
   *%FTN1   16550
  SREAD     21241
   *%READ   21241
   *%JFIL   21707
   *%RDSC   21663
  .OPSY     21774
   *.OPSY   21774
  DUMRX     22034
   *$LIBR   22034
   *$LIBX   22062

FTN02       13127
   *%FTN2   13741

FTN03       13127
   *%FTN3   15117

FTN04       13127
   *%FTN4   13702
  %WRIT     17515
   *%WRIT   17700
   *%WRIF   17577
   *%WBUF   17777
  FADSB     20213
   *.FAD    20213
```

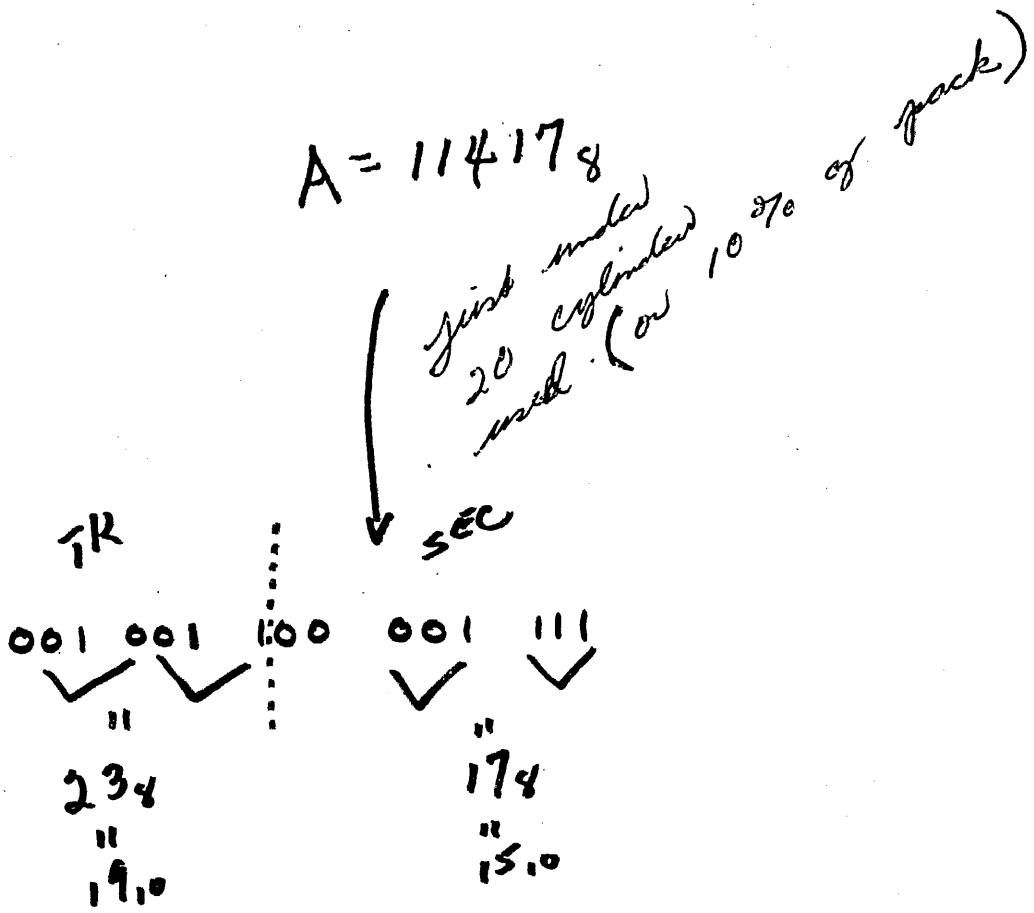
DISC LOADING PHASE (CON'T)



(DISC RESIDENT SECTION)

[SLIDE 40]

```
  *.FSB    20222
 .OPSY     20363
  *.OPSY   20363
 .FLUN     20423
  *.FLUN   20423
 .PACK     20444
  *.PACK   20444
 DUMRX     20560
  *$LIBR   20560
  *$LIBX   20606
 .ZRLB     20644
  *.ZRLB   20644
 DLDST     20705
  *.DLD    20705
  *.DST    20715

*SYSTEM STORED ON DISC
```



[SLIDE 40J]

FORMATTING USER DISCS OR CARTRIDGES

PURPOSE: To format a USER DISC or CARTRIDGE anytime a new disc is added or an old SYSTEM DISC is to be reused as a USER DISC.

WHAT SYSTEM DOES: Creates an Unlabeled Disc ready for use in DOS/M System by

1. Writing new Label Sector on sector Ø with
   a. First two words as Ø,DO
   b. Generation Code # entered by operator
   c. Three label words as *Y,ST,EM
   d. # bad tracks as Ø

2. Writing new bootstrap on sectors 1 and 2

3. Clearing all PCI and DCI on all sectors

OPERATION PROCEDURE:

1. All equpment on. "READY" Drive.
2. DISC PROTECT OVERRIDE SWITCH "ON".
3. Load Configured System Generator (DSGEN) into memory using BBL.
4. LOAD ADDRESS 1ØØ octal.
5. Switch 15 "UP".
6. PRESET and RUN.
7 Answer requests printed on TTY.
8. System Generator halts with T=1Ø2ØØ7 at end. Press "RUN" to do another Disc (with switch 15 still "UP") OR put switch 15 down and press "RUN" to begin SYSTEM GENERATION proper.

FORMATTING USER DISCS EXAMPLE

```
    SYS GEN CODE?
→9000

    SYS DISC CHNL?
→14

    # SECTORS/TRACK?
→12

    USER DISC SUBCHNL?
→0
    TURN ON DISC PROTECT OVERRIDE - PRESS RUN

    USER DISC SUBCHNL?
→1
```

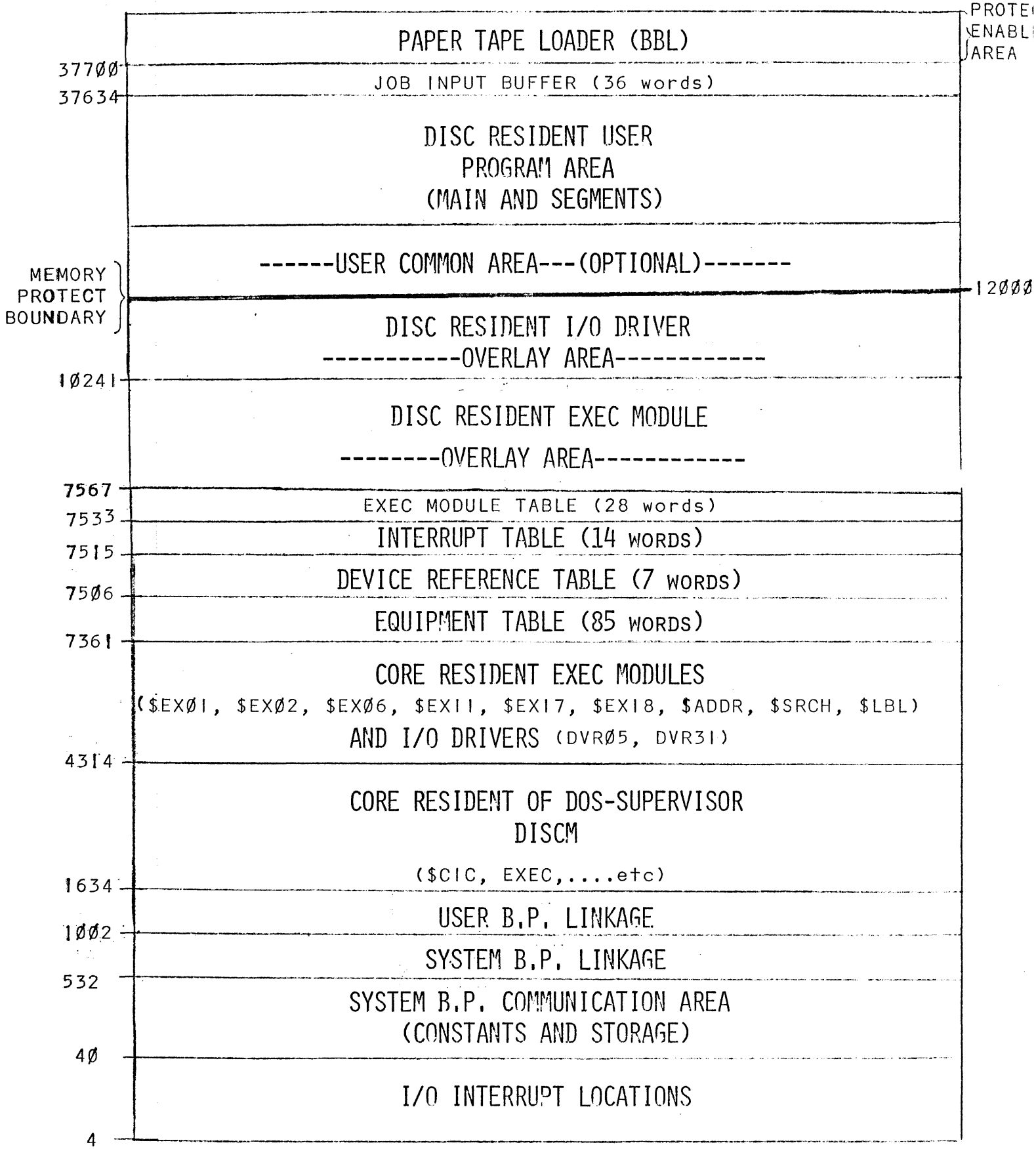
FORMATTED FIXED DISC

FORMATTED CARTRIDGE

```
    SYS GEN CODE?
→9000

    SYS DISC CHNL?
→14

    # SECTORS/TRACK?
→12

    SYS DISC SIZE?
→200

    # DRIVES?
→1

    FIRST SYSTEM TRACK?
→0
    FIRST SYSTEM SECTOR?
→3

    SYS DISC SUBCHNL?
→1

    USER DISC SUBCHNL?
→1

    TIME BASE GEN CHNL?
→12

    IS 2114?
→NO

    LWA MEM?
→37677

    ALLOW :SS?
→YES

    PRGM INPT?
→PT

    LIBR INPT?
→PT
```

WENT DIRECTLY INTO STANDARD SYSTEM GENERATION BY PUTING SWITCH 15 DOWN AND PRESSING "RUN"

CORE MAP FOR DOSM SYSTEM GENERATION EXAMPLE (16K)



[SLIDE AP-1]

EQUIPMENT TABLE ENTRY FORMAT

WORD	CONTENTS															
1	DRIVER "INITIATION" SECTION ADDRESS															
2	DRIVER "CONTINUATION" SECTION ADDRESS															
3	D	R					UNIT #		CHANNEL #							
4	AV		EQUIPMENT TYPE CODE						STATUS							
5	(SAVED FOR DRIVER USE)															
6	(SAVED FOR DRIVER USE)															
7	REQUEST RETURN ADDRESS															
8	REQUEST CODE															
9	CURRENT I/O REQUEST CONTROL WORD															
10	REQUEST BUFFER ADDRESS															
11	REQUEST BUFFER LENGTH															
12	TEMPORARY OR DISC TRACK #															
13	TEMPORARY OR STARTING SECTOR #															
14	TEMPORARY STORAGE FOR DRIVER															
15	UPPER MEMORY ADDRESS OF MAIN DRIVER AREA															
16	UPPER MEMORY ADDRESS OF DRIVER LINKAGE AREA															
17	STARTING TRACK #								STARTING SECTOR #							
BITS	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Ø'S IF CORE-RESIDENT

AVAILABLE FOR DRIVER TEMPORARY

D = 1 IF DMA CHANNEL REQUIRED.
R = 1 IF DRIVER TYPE IS CORE-RESIDENT.
UNIT # MAY BE USED FOR SUB-CHANNEL ADDRESSING.
CHANNEL # I/O SELECT CODE FOR DEVICE (LOWER NUMBER IF MULTIBOARD INTERFACE.)

AV = Ø - UNIT NOT BUSY AND AVAILABLE
= 1 - UNIT DISABLED (DOWN)
= 2 - UNIT BUSY
= 3 - UNIT WAITING FOR AN AVAILABLE DMA CHANNEL

(THIS FIELD SET BY SYSTEM)

STATUS - ACTUAL OR SIMULATED UNIT STATUS AT END OF OPERATION. (DRIVER MUST SET THIS FIELD)

EQUIPMENT TYPE CODE - IDENTIFIES TYPE OF DEVICE AND ASSOCIATED SOFTWARE DRIVER. ASSIGNED EQUIPMENT TYPE CODES IN OCTAL ARE:

ØØ-Ø7	PAPTER TYPE DEVICES
ØØ	TELEPRINTER
Ø1	PUNCHED TAPE READER
Ø2	HIGH SPEED PUNCH
Ø5	TELETYPE (SYSTEM)
1Ø-17	UNIT RECORD DEVICES
1Ø	RESERVED FOR PLOTTER
12	LINE PRINTER
15	MARK SENSE CARD READER
2Ø-37	MAGNETIC TAPE/MASS STORAGE AND OTHER DEVICES CAPABLE OF BOTH INPUT AND OUTPUT.
22	3030 MAGNETIC TAPE
31	MOVING-HEAD DISC

FOR EQUIPMENT TYPE CODES 1 THROUGH 17, ODD NUMBER INDICATE INPUT DEVICES AND EVEN NUMBER INDICATE OUTPUT DEVICES (EXCEPT Ø5, WHICH IS BOTH INPUT AND OUTPUT).

DEVICE REFERENCE TABLE FORMAT

EACH ENTRY IN THIS TABLE REQUIRES ONLY ONE WORD IN MEMORY. THE VALUE OF EACH ENTRY (DECIMAL NUMBER, 1-63) ASSOCIATES A LOGICAL UNIT NUMBER WITH AN EQUIPMENT TABLE ENTRY FOR THE SYSTEM IN THE FOLLOWING MANNER:

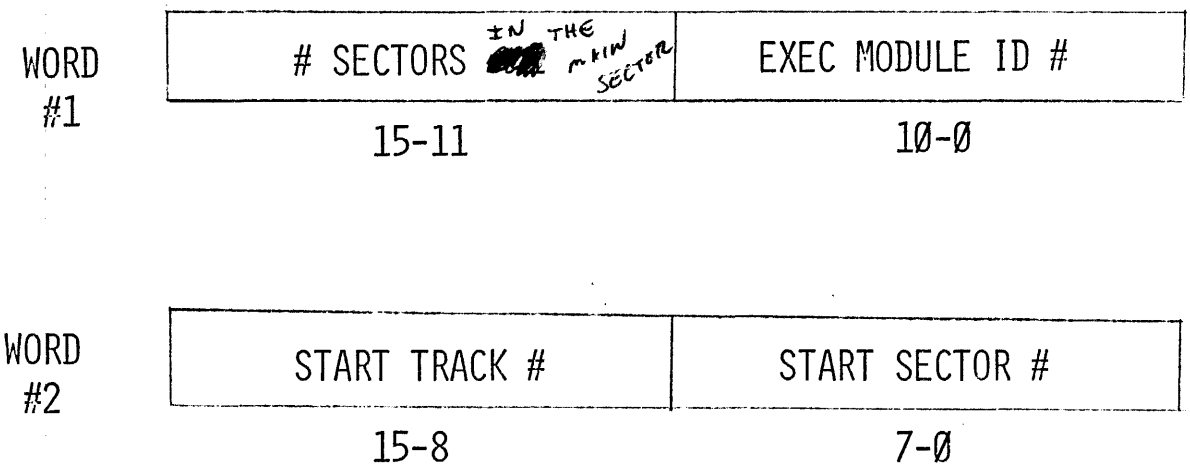
SEQUENCE IN MEMORY TABLE	LOGICAL UNIT #	FUNCTION
1	1	SYSTEM TELEPRINTER
2	2	USER MASS STORAGE
3	3	SYSTEM MASS STORAGE
4	4	STANDARD PUNCH DEVICE
5	5	STANDARD INPUT DEVICE
6	6	STANDARD LIST DEVICE
7-63	7-63	ANY DEVICE

INTERRUPT TABLE FORMAT

EACH ENTRY IN THIS TABLE REQUIRES ONLY ONE WORD IN MEMORY AND IS ASSOCIATED WITH EACH I/O CHANNEL IN THE COMPUTER (STARTING WITH LOCATION 6) WHICH CAN CAUSE AN INTERRUPT. EACH LOCATION IN THIS TABLE HAS AN ENTRY VALUE. MEMORY, LOCATIONS ARE ASSOCIATED IN CONSECUTIVE INCREASING ORDER WITH AN I/O CHANNEL. TABLE VALUES ARE ZERO FOR AN I/O CHANNEL NOT REQUIRING INTERRUPT. I/O CHANNELS REQUIRING INTERRUPT CONTAIN THE START ADDRESS OF THE EQUIPMENT TABLE ENTRY OF THE ASSOCIATED DEVICE.

EXEC MODULE DOUBLET TABLE FORMAT

(TWO WORDS PER DISC RESIDENT EXEC MODULE)



SYSTEM DIRECTORY LISTING FOR DOSM GENERATION EXAMPLE

NAME	TYPE	SCTRS	DISC	ORG	PROG LIMITS		B.P. LIMITS		ENTRY LIBR.		P-BIT
SUBCHAN=1											
$EX03	XS	0002	T001	011	07567	07624	00732	00733	07567	07624	
$EX04	XS	0004	T001	013	07567	10176	00732	00741	07567	10176	
$EX05	XS	0002	T001	017	07567	07745	00732	00733	07567	07745	
$EX07	XS	0002	T001	019	07567	07746	00732	00733	07567	07746	
$EX08	XS	0002	T001	021	07567	07732	00732	00733	07567	07732	
$EX09	XS	0003	T001	023	07567	10142	00732	00763	07567	10142	
$EX10	XS	0002	T002	002	07567	07745	00732	00733	07567	07745	
$EX12	XS	0002	T002	004	07567	07761	00732	00733	07567	07761	
$EX13	XS	0004	T002	006	07567	10223	00732	00754	07567	10223	
$EX14	XS	0004	T002	010	07567	10241	00732	00751	07567	10241	
$EX15	XS	0003	T002	014	07567	10153	00732	00763	07567	10153	
$EX16	XS	0002	T002	017	07567	07722	00732	00733	07567	07722	
$EX19	XS	0003	T002	019	07567	10107	00732	01000	07567	10107	
$EX20	XS	0003	T002	022	07567	10167	00732	00761	07567	10167	
DVR01	DR	0003	T003	001	10241	10555	01000	01002	10241	10555	
DVR02	DR	0003	T003	004	10241	10443	01000	01002	10241	10443	
DVR22	DR	0005	T003	007	10241	11075	01000	01002	10241	11075	
LOADR	UM	0032	T003	012	12000	21032	01002	01425	12000	21032	
JOBPR	UM	0038	T004	020	12000	22463	01002	01414	12000	22463	
ASMB	UM	0023	T006	010	12000	17120	01002	01362	16522	17120	
ASMBD	US	0004	T007	009	17127	17647	01362	01363	17442	17647	
ASMB1	US	0006	T007	013	17366	20542	01362	01424	17366	20542	
ASMB2	US	0007	T007	019	17345	20550	01362	01410	17351	20550	
ASMB3	US	0003	T008	002	17473	17771	01362	01363	17630	17771	
ASMB4	US	0004	T008	005	17366	20027	01362	01371	17366	20027	
ASMB5	US	0006	T008	009	17345	20425	01362	01404	17351	20425	
FTN	UM	0006	T008	015	12000	13127	01002	01047	12000	13127	
FTN01	US	0031	T008	021	13254	22120	01047	01502	16550	22120	
FTN02	US	0025	T010	004	13254	21027	01047	01356	13741	21027	
FTN03	US	0024	T011	005	13254	20600	01047	01277	15117	20600	
FTN04	US	0025	T012	005	13254	20750	01047	01360	13702	20750	
LIBRY	LB	0147	T013	007							

EQUIPMENT TABLE LISTING

```
:EQ
EQT 01 CH 11 DVR05 0 R U0 S0
EQT 02 CH 13 DVR01 0 0 U0 S0
EQT 03 CH 14 DVR31 D R U0 S0
EQT 04 CH 16 DVR02 0 0 U0 S0
EQT 05 CH 22 DVR22 D 0 U0 S0
@
```

LOGICAL UNIT TABLE LISTING

```
:LU
LU01 EQT01
LU02 EQT03
LU03 EQT03
LU04 EQT04
LU05 EQT02
LU06 EQT01
LU07 EQT05
@
```

MEMORY DUMP FOR DOSM SYSTEM GENERATION EXAMPLE

CORE DUMP: 000004-007566

000000:					114532	114532	114532	114532	TRAP CEL
00[illegible]0:	114532	114532	114532	114532	114532	114532	114532	114532	JSB $CIC
00[illegible]20:	114532	114532	114532	114532	114532	114532	114532	114532	
000030:	114532	114532	114532	114532	114532	114532	114532	114532	
000040:	177700	177766	177767	177770	177771	177772	177773	177774	
000050:	177775	177776	177777	000000	000001	000002	000003	000004	
000060:	000005	000006	000007	000010	000011	000012	000021	000140	CONSTANTS
000070:	000017	000037	000077	000177	000377	177400	003777	177700	
000100:	037633	000000	000000	000012	000000	000000	000030	000000	
000110:	000000	000001	000001	000000	000003	013000	000030	007361	
000120:	000005	007506	000007	037634	000000	000000	000000	000000	
000130:	000000	000000	000000	000000	000000	000000	000000	000000	
000140:	000000	000000	000000	000000	000000	000000	000000	000000	
000150:	000000	000000	000000	000000	060307	000001	000001	033403	
000160:	033403	000001	000000	000377	000026	000000	001005	000000	
000170:	000046	000000	000000	000000	013001	000001	321450	000000	CURRENT
000200:	013000	007515	000016	007361	007362	007363	007364	007365	EQT TABL
000210:	007366	007367	007370	007371	007372	007373	007374	007375	ADDRESSES
000220:	007376	007377	007400	007401	000000	000000	000000	000000	
000230:	000000	000000	000000	000000	000000	000000	000000	177777	
000240:	177777	077777	003327	007533	000016	000000	007567	010241	
000250:	000732	001000	010241	001000	012000	001002	001633	000006	
000260:	000000	000001	000000	002024	010463	000412	000000	000000	
000270:	046102	042117	021450	050521	050521	053421	044456	031461	
000300:	020000	036173	000723	041456	031461	020000	036173	000724	
000310:	044456	031062	020000	036164	000000	041456	031062	020000	
000320:	035164	000000	046117	040504	051000	036155	000000	163252	SYSTEM
00[illegible]0:	041120	051000	036146	000000	040523	046502	020000	036137	I/O
000340:	000000	037501	051503	047000	036137	000000	037501	051515	BUFFER
000350:	041000	036137	000000	037502	047103	047000	036137	000000	
000360:	037502	050113	052400	036137	000000	037503	044117	050000	
000370:	033403	000000	000312	001534	177610	020040	020040	020040	
000400:	030471	027117	041524	027067	030040	000000	000000	000000	
000410:	000000	000000	000000	000000	000400	000000	000001	000003	
000420:	000003	000004	000002	000001	000005	000000	000000	050521	
000430:	050521	050421	044456	031461	020000	037515	051531	046400	
000440:	036137	000000	037522	046125	047000	036137	000000	037501	
000450:	043114	043400	036137	000000	037514	051524	046000	036137	
000460:	000000	037514	052516	044400	036137	000000	037522	051502	
000470:	000000	042101	000000	000000	000000	000000	000000	007552	
000500:	177771	000000	000000	000000	000000	003463	300000	000000	
000510:	000000	177750	000000	000000	000000	177777	177777	077777	
000520:	003327	000000	006407	000120	000001	000000	000000	000000	
000530:	114532	000000	001634	004160	004154	002214	003400	003511	
000540:	003020	002635	003744	003361	003426	004165	004267	002507	DISCM ENT
000550:	004010	003244	003345	003177	001714	003563	004044	002212	POINT
000560:	002653	002657	002661	002663	002665	002553	003214	003207	LINKS
000570:	003143	002576	002700	002631	002521	003130	003117	004243	
000600:	002445	004302	004303	004314	004376	000000	000000	000000	EXEC ENT
000610:	004446	000000	000000	000000	000000	004505	000000	000000	POINTS
000620:	000000	000000	000000	004671	005264	000000	000000	002322	
000630:	002222	002637	004262	004151	001746	004034	004125	004304	
00[illegible]0:	004305	001736	001726	001723	004300	004301	004307	004272	
000650:	004312	004313	004310	004311	004274	004150	004126	004127	DISCM
000660:	004135	004124	004005	004024	004136	004035	004137	004277	PAGE
000670:	004131	004132	004133	004134	004270	004275	004130	004145	LINKS
000700:	004146	004147	003607	002213	003743	103575	003325	002463	
000710:	003204	003213	003220	003236	003003	005774	006011	006266	

DISC RESIDENT I/O DRIVER B.P. LINKAGE

EXEC SUBROUTINES LINKS

EXEC MOD DISC RES LINKAGE

USER BASE PAGE LINKAGE

DISCM

000720:	006354	006336	006372	105774	006330	006315	006410	006464
000730:	006660	006743	007567	000000	000000	000000	000000	000000
000740:	000000	000000	000000	000000	000000	000000	000000	000000
000750:	000000	000000	000000	000000	000000	000000	000000	000000
000760:	000000	000000	000000	000000	000000	000000	000000	000000
000770:	000000	000000	000000	000000	000000	000000	000000	000000
001000:	010241	011001	012000	017624	020315	022175	022202	015126
001010:	021023	015342	015465	014302	014733	022213	115037	014623
001020:	015442	015754	020307	015124	015216	015221	014035	014711
001030:	020431	020432	014364	020433	020434	020435	015101	015557
001040:	021237	015623	014027	021051	022017	020443	020444	020445
001050:	020446	020447	020450	020451	020452	021015	014124	115101
001060:	014645	020453	015217	015256	017674	017377	017621	017661
001070:	022263	015753	021173	017625	020427	014435	014370	014353
001100:	015041	014636	014463	017314	015337	015513	014145	014044
001110:	014054	014055	014056	014052	014050	014022	014051	014057
001120:	014060	014061	015040	014045	012262	020524	013761	012725
001130:	013157	012063	012636	013152	016551	013435	013626	013147
001140:	013627	013140	013166	016242	013142	013433	013434	013170
001150:	020426	013161	021647	020757	021016	017427	021001	013126
001160:	020624	017412	016276	016410	016267	016561	016026	016273
001170:	013141	016010	013163	015427	012645	015214	020330	020327
001200:	015755	020326	115427	015341	015317	012103	015466	015445
001210:	015570	022214	022052	015266	015443	015675	015430	015462
001220:	015432	015433	015434	021026	015227	015431	021042	013164
001230:	015756	015343	022326	015624	020323	020324	020325	022264
001240:	022265	022266	022325	022274	022273	022276	022275	022270
001250:	022271	022267	022272	022305	022314	022320	022323	022324
001260:	014432	115442	014434	015131	020274	020015	022176	016415
001270:	017622	016416	015127	016376	016763	016720	017026	017023
001300:	015104	016422	017606	022463	022464	016411	016414	017600
001310:	017157	017156	022203	015444	022205	016417	016420	017632
001320:	016426	015722	013151	013133	016424	015037	015102	115102
001330:	016372	013174	016423	015720	022201	022204	022174	022036
001340:	022177	022207	022062	012722	022212	022210	022206	022211
001350:	022200	022215	022001	022003	021767	016241	021651	021064
001360:	013104	013135	012447	013010	016575	021543	021534	021542
001370:	021541	021540	021461	021474	016721	116721	013154	020527
001400:	013155	020465	012473	020436	120436	012723	020437	020440
001410:	020441	020442	014351	020314	012214	012215	012216	021000
001420:	020777	020653	020757	012213	021023	020043	020057	020206
001430:	020353	113273	020007	017433	117751	017774	017100	017150
001440:	017661	017542	017624	113276	017421	017111	017112	017571
001450:	017077	117112	017423	017107	017424	017422	017110	017106
001460:	017103	017113	117110	017306	117107	017102	017114	013271
001470:	013267	017101	013275	017305	117436	021707	021663	021774
001500:	022034	022062	147722	146701	152240	143317	151240	120302
001510:	144716	140722	154640	125215	105252	120311	147320	152724
001520:	120315	140731	120301	146323	147640	141305	120323	142714
001530:	142703	152305	142240	143317	151240	140640	151711	147307
001540:	146305	120322	142701	142240	120240	120240	120252	106612
001550:	125240	147720	142722	140724	144717	147256	120240	120240
001560:	120240	120240	120240	120240	120240	120240	120240	120240
001570:	120240	120240	120240	120240	120240	120240	120240	120240
001600:	120240	120240	125215	105252	120240	120240	120240	120240
001610:	120240	120240	120240	120240	120240	120240	120240	120240
001620:	120240	120240	120240	120240	120240	120240	120240	120240
001630:	120240	120240	120240	120240	003327	103100	017736	063034
001640:	070242	163634	170627	102504	033635	073646	103111	106504
001650:	054057	102004	054060	124630	054103	027754	044045	074000

DISCM

001660:	002020	027714	003000	040202	002020	027714	044201	160001
001670:	002003	027714	114552	102504	164204	114001	027776	060200
001700:	002003	027714	060261	002002	027714	160557	164201	006003
001[illegible]0:	002002	027714	074260	124544	063634	017746	002001	027722
001720:	017726	127634	017726	102100	102705	127634	001721	060241
001730:	103101	000036	102101	060237	064240	127726	001637	070237
001740:	074240	001520	102201	002004	070241	127736	002106	003000
001750:	040254	002020	037746	127746	034105	027761	063775	070105
001760:	034104	060106	002020	027714	034107	027714	063775	070107
001770:	034106	060106	050110	124631	027714	176650	170217	070531
002000:	174220	017331	060513	003004	050203	070513	060262	000010
002010:	026070	060141	002003	026070	160212	002020	003004	050054
002020:	002001	026070	007400	044111	044121	003400	140001	114632
002030:	040117	050203	002001	026070	160214	160000	010075	150633
002040:	002001	026070	160211	114634	026070	060123	072052	160214
002050:	066207	017361	002052	160220	164215	006021	026061	002004
002060:	001100	070530	160206	012211	170206	160211	070242	026551
002070:	160217	002002	026152	160206	012211	170206	160213	012210
002100:	002002	026163	160211	070505	164220	114634	026150	164635
002110:	154636	026204	060260	002003	026143	003400	140121	114632
002120:	040117	040056	160000	002020	026143	062212	002002	026141
002130:	160206	070237	164220	074240	160211	070242	006400	074260
002140:	027426	006400	074260	164220	160206	102100	102705	124505
002150:	160206	124505	050056	002301	026641	160206	110637	150640
002160:	026644	060056	026641	063213	002003	026171	060513	002021
002170:	027256	160205	010072	053204	027244	063220	002003	124554
002200:	017003	027271	124554	027271	060203	072213	027244	177734
002210:	020000	037777	000000	000000	000000	103100	114641	062214
002220:	040052	002001	102505	070512	070242	072424	036424	160000
002230:	012473	052475	002001	026633	160512	012474	160000	052476
002240:	002001	026315	066424	160001	114634	026635	160001	070225
002250:	007004	040001	002003	026635	040052	070224	002020	026635
002260:	040042	002021	026635	064043	002400	017361	000226	064224
002270:	007004	036424	062266	070505	162424	016507	170505	034505
002300:	036424	006006	026274	160226	052500	002001	026325	114642
002310:	170001	006004	102100	102705	124225	052477	026320	026633
002320:	114642	114577	007400	134532	124643	002003	026635	052501
002330:	124534	052502	026346	052503	026340	052504	026425	026362
002340:	060237	072342	000000	026312	034105	026312	060103	002003
002350:	026344	032505	072357	060103	032506	072360	060056	000000
002360:	000000	026312	002021	003004	042461	002020	026635	160226
002370:	002020	003004	070514	050064	026400	050065	002001	026402
002400:	060063	070514	050060	026312	042436	160000	050052	026635
002410:	052437	026431	070474	062462	070475	002404	064514	154475
002420:	027020	002004	034475	026417	000000	160644	070474	060066
002430:	027020	006400	160645	070474	062470	027020	002436	002431
002440:	002431	002431	100612	177777	100605	100606	100607	100607
002450:	100607	100611	177777	002431	002431	002431	100603	100604
002460:	100617	000022	002463	000020	000021	000006	000007	000010
002470:	000022	000013	000004	176000	001777	114000	002214	004243
002500:	177755	177754	177753	177752	000027	102600	103700	000000
002510:	002003	026635	050054	026635	001275	002001	126507	160000
002520:	026510	140646	070514	060232	070472	066552	074473	160647
002530:	070474	062471	027020	006003	026665	060514	050055	002001
002540:	026432	060001	040055	160000	010071	050065	026432	052471
002550:	026432	026631	002533	004172	010072	002003	026653	007400
002560:	044000	003004	040122	002020	026653	044121	160001	002003
002570:	026655	040052	114632	040117	017345	126553	000000	164226
002600:	174212	006020	007004	060225	170211	160227	050050	003004
002610:	170213	060230	054056	160230	170214	160231	170215	060232

002620:	154650	026625	154651	026625	160232	170216	160233	170217	
002630:	126576	006400	026670	064054	026670	064055	026670	064056	
002640:	026670	064057	070473	026670	007400	170206	016776	164652	
002650:	026670	164653	026670	064060	026670	064061	026670	064062	
002660:	026670	064063	026670	064064	026670	064065	060512	002001	
002670:	060242	070472	002400	070245	160654	070474	160653	027020	
002700:	004241	160205	002021	026723	060061	070257	017003	026721	
002710:	026717	160206	110637	150640	026717	034257	026721	060160	
002720:	026774	060203	170001	160213	010077	070001	160212	002020	
002730:	003004	150650	026742	150651	026742	030001	170213	026744	
002740:	002404	026735	060055	026735	160205	010072	164203	114001	
002750:	070510	002102	026766	160205	001222	000010	026762	060513	
002760:	003004	070513	160206	130655	170206	126700	050056	002300	
002770:	017331	006440	016776	060510	036700	126700	000000	060526	
003000:	002003	126776	124000	002707	064201	002400	150001	027017	
003010:	050165	002002	027016	006004	150001	037003	037003	127003	
003020:	073116	077115	060243	070477	060244	002003	124474	003004	
003030:	070500	060245	002020	027222	053116	027110	160477	010074	
003040:	053116	027047	034477	034477	034500	027036	124474	003004	
003050:	070245	064246	074157	007004	044247	074170	160477	001722	
003060:	010071	034477	164477	074166	036212	002003	027106	001722	
003070:	001222	070001	003004	040170	002021	074170	017117	060174	
003100:	070166	064250	074167	007004	044251	074170	017117	072212	
003110:	067115	063116	003004	070245	124246	000001	000014	003107	
003120:	060166	017130	063127	006404	017400	002400	127117	000166	
003130:	003122	006400	114577	177770	060154	010074	003004	044000	
003140:	006021	026663	127130	000000	070513	064252	074167	007004	
003150:	140650	144000	074170	040055	160000	070166	017117	060174	
003160:	070166	064253	007004	060513	140651	144000	006003	127143	
003170:	060253	070167	036212	074170	017117	072212	127143	073204	
003200:	077205	060206	073205	027244	000000	000000	000000	000000	
003210:	060203	073213	027244	000000	073221	060203	073220	027244	
003220:	000000	000000	037236	063115	073240	063116	073237	060474	
003230:	073241	060472	073242	060473	073243	027244	000000	000000	
003240:	000000	000000	000000	000000	063204	002003	027256	163205	
003250:	001222	010056	002002	027256	073204	127206	063213	002003	
003260:	027271	060513	002020	027271	063213	017345	002400	073213	
003270:	127207	063220	002003	027304	017003	027277	027304	063220	
003300:	017345	002400	073220	127221	063236	002003	027325	060245	
003310:	002020	027325	002400	073236	063242	070472	063243	070473	
003320:	063241	070474	063237	067240	027020	102100	003400	007400	
003330:	027325	002002	064201	060203	150001	027342	006004	150001	
003340:	002001	127331	002400	170001	127331	002575	007360	074474	
003350:	067357	170001	002004	006004	034474	027351	127345	000203	
003360:	177757	000000	070503	074505	167361	074504	037361	006400	
003370:	002002	164503	174504	034503	034504	034505	027367	127361	
003400:	007675	077412	064203	077423	073424	060055	073413	063424	
003410:	073414	114545	000001	000002	007726	000000	003420	027244	
003420:	063423	017345	127400	000000	007726	000000	077561	017563	
003430:	034261	067561	063556	006002	063553	073441	114545	000002	
003440:	000001	003554	000002	003463	063561	002002	027244	060520	
003450:	114634	027244	060517	103101	000036	102101	000515	064516	
003460:	102100	102705	124520	017563	060123	073471	114545	000001	
003470:	000401	037634	000044	003475	027444	077562	017563	002400	
003500:	070261	067562	017575	060471	150656	027430	150657	027430	
003510:	027444	060111	050054	027544	073532	064123	077533	016553	
003520:	160205	001222	000010	027530	060203	050513	002001	017143	
003530:	114545	000001	003532	003533	000044	003537	027244	160123	
003540:	010075	150633	017575	027511	060261	002003	027551	034260	
003550:	027244	007400	027426	003554	006412	040137	003557	006412	

DT SCM

```
003560: 025137 177777 000000 003464 060237 070515 060240 070516
003570: 060241 070517 060242 070520 127563 000000 074530 074472
003600: 067575 074473 064123 160001 010075 150633 027631 034261
003610: 114545 000002 000001 004140 177770 003625 064471 154656
003620: 027244 060520 114634 124554 027452 002400 070261 017563
003630: 127575 160001 006004 164001 114577 177770 170635 064471
003640: 154660 027677 150661 124662 150636 124663 150664 124665
003650: 150666 124662 154656 002001 027702 150656 006401 027714
003660: 074471 062213 002002 027670 160667 070474 062463 027020
003670: 076213 150656 027447 017345 160211 070505 026141 150660
003700: 027743 027607 154657 027736 006002 027607 064262 004010
003710: 027714 064141 006003 027743 150670 027731 150671 027725
003720: 150672 027731 150673 027731 027607 160674 070474 060064
003730: 027020 160675 070474 160650 064530 027020 150657 027743
003740: 150676 027743 027607 002400 114556 002404 070262 160677
003750: 070141 160700 070142 160701 070143 060263 070166 060254
003760: 070167 060264 070170 017117 060174 070166 060255 070167
003770: 060265 070170 002400 070471 017117 002400 070261 070260
004000: 064530 070530 102100 102705 124254 060112 070111 006400
004010: 074475 060106 032150 070106 016044 070262 070245 062276
004020: 070474 064475 062313 124540 064242 074470 062126 050471
004030: 124702 070471 170703 124544 000000 060141 002003 124704
004040: 060262 032150 070262 124705 000000 060120 003004 070474
004050: 074475 064117 044055 160001 010072 032144 072076 006004
004060: 160001 001265 012304 052305 002041 026073 062072 044056
004070: 170001 124706 004045 052153 032152 170001 107700 144707
004100: 034474 026053 102106 107706 102107 107707 064201 002400
004110: 170001 070513 170710 170711 170712 170713 070261 070200
004120: 170703 006004 170001 126044 040502 050101 043517 045117
004130: 042512 046125 042521 052520 042116 042101 052131 047506
004140: 044507 047117 051105 042040 107700 045117 041120 051040
004150: 100000 035000 000040 000400 062271 070474 060065 124540
004160: 106700 062273 070474 062310 124540 003474 062165 002004
004170: 160000 114565 160206 001222 010056 002003 026204 007400
004200: 046165 160205 010072 124553 160206 012304 052305 002001
004210: 026217 114714 026217 003401 003400 042165 124566 162165
004220: 002021 003004 170212 036165 162165 170213 036165 162165
004230: 170214 036165 162165 170215 036165 162165 170211 036165
004240: 114572 126165 026240 003134 072250 162243 072261 062260
004250: 036243 000066 005600 002040 002004 036261 026251 126243
004260: 001005 000000 002573 070001 001700 040001 126262 000470
004270: 100613 100614 100615 100616 100617 100620 100621 100622
004300: 100623 100624 100625 100626 037400 014400 177764 177763
004310: 000014 000015 000016 000017 060224 040047 002002 026373
004320: 060231 114715 160226 002020 026353 060160 006400 114577
004330: 177770 002002 006004 007004 144230 006021 026341 006400
004340: 026365 006400 060154 010074 003000 140227 140230 002003
004350: 026353 002021 026365 164230 144227 007004 060154 010074
004360: 002004 040001 002020 026337 164230 174231 002400 070245
004370: 102100 102705 124225 002400 070245 124541 060224 040047
004400: 002002 026443 060227 114715 060230 114715 060231 114715
004410: 060160 006400 114577 177770 002002 006004 174227 060102
004420: 002003 026431 006400 114577 177770 002002 044052 074000
004430: 026433 060154 010074 170230 060116 170231 002400 070245
004440: 102100 102705 124225 002400 070245 124541 060050 040224
004450: 002002 026502 060230 114715 064227 034162 060200 002004
004460: 114716 026477 064227 060114 114716 026477 002400 170230
004470: 002400 070245 044052 160001 102100 102705 124225 044057
004500: 160001 026467 002400 070245 124541 060472 072604 060473
004510: 072665 060245 052666 026537 060262 002002 026601 060525
```

DISCM

$EX01

$EX02

$EX06

$EX11

```
004520:  166663  056670  002003  026601  066647  076641  062664  114717
004530:  026614  066641  044063  076641  056663  026605  026526  062664
004540:  066577  114717  026556  062664  066573  114717  026562  034162
004550:  060200  002004  066664  114716  026567  006401  066577  002400
004560:  070245  126665  062573  064046  114543  000270  066647  076664
004570:  062577  064046  114543  000133  062664  064046  114543  000126
004600:  026556  002400  066667  114543  000270  034162  060200  002004
004610:  066664  114716  026630  026555  066641  044060  160001  070161
004620:  006004  160001  070157  006004  160001  070200  006641  026567
004630:  076664  062663  040043  052647  026644  072641  040043  064043
004640:  114543  000000  062641  026632  062664  064046  114543  000270
004650:  060161  070275  060157  070276  060200  070277  062670  172663
004660:  034525  066647  026567  000460  000000  000000  177765  177607
004670:  022124  026707  062771  002020  026772  062704  006002  062705
004700:  070473  017250  064057  027051  047122  050105  004672  060161
004710:  073261  060200  073262  064224  044047  006024  026724  160231
004720:  050054  027074  002024  027120  062706  070526  060156  003000
004730:  006021  160230  072771  002020  027013  070161  017215  017126
004740:  063030  114721  026747  064227  017200  027025  027101  160227
004750:  002003  027043  010075  053000  027101  027043  062771  003000
004760:  070161  017215  063030  114721  027001  064227  017200  026772
004770:  027006  000000  036771  026756  063162  066777  027045  177756
005000:  025000  160227  010075  053000  027006  026772  060262  002002
005010:  017230  017126  027101  160227  002002  026756  072771  017230
005020:  017215  017126  063030  114721  027043  063031  064041  027045
005030:  005034  005032  046102  046075  000000  000000  000000  005040
005040:  052516  046102  046040  063037  064046  114722  017250  064055
005050:  047154  002400  073154  070526  070245  160600  070474  060057
005060:  124540  006400  074245  074526  063154  002002  027050  063263
005070:  102100  102705  124225  005046  160230  070161  063073  070526
005100:  017215  006400  077253  074126  074133  067177  017200  027115
005110:  006404  060370  070200  063227  114720  064370  074157  027061
005120:  070245  160601  070474  063125  124540  000023  000000  060176
005130:  050272  027136  070472  060272  070473  037154  000271  053155
005140:  127126  067161  053156  067157  057161  063160  073164  077165
005150:  063163  064043  114722  127126  000000  042117  052123  041040
005160:  037477  037440  005166  005164  000000  000000  042111  051503
005170:  020116  047524  020117  047040  051531  051524  042515  005174
005200:  000000  160001  050273  006005  127200  160001  050274  006005
005210:  127200  160001  050275  037200  127200  000000  006400  060161
005220:  050155  064115  074200  063227  006404  114720  127215  000200
005230:  000000  060161  033241  001727  073247  063242  064042  114722
005240:  127230  000060  005243  051525  041103  044101  047075  000000
005250:  000000  063261  070161  063262  070200  060074  070163  073263
005260:  127250  000000  000000  000000  006002  027700  160227  064161
005270:  077314  050050  003004  114565  060514  053313  002001  027335
005300:  017766  160206  170230  064224  054056  027310  164220  174231
005310:  102100  102705  124225  000015  000000  000000  040201  040045
005320:  073766  164000  006003  027331  044055  160001  010072  053334
005330:  002001  037315  060203  127315  000000  160205  010072  073334
005340:  017315  027356  063334  002004  017315  027356  060061  017315
005350:  027356  060062  017315  027356  102000  027354  006004  160001
005360:  064514  054056  027411  001222  010056  002003  027411  050054
005370:  027402  050055  027376  063375  124566  005266  017766  063334
005400:  064512  124553  060512  070470  063410  070471  017766  124544
005410:  043517  063334  017315  173766  063334  002004  017315  173766
005420:  160205  001222  000010  027427  064513  006020  114567  060514
005430:  050054  027435  050055  027435  027443  160227  050050  003004
005440:  010072  050056  034162  114571  160226  002021  027462  160206
005450:  013456  053457  027565  027667  017766  124574  037400  014400
```

$EX11

$EX17

$EX18

[SLIDE AP-11]

$EX18

```
005460:  000016  000017  060514  050056  027667  064230  007000  044254
005470:  006021  124561  017756  044230  002040  124562  007004  044100
005500:  006020  124562  160206  013456  053457  002001  027667  060514
005510:  053460  027454  053461  027454  060102  002003  027523  001727
005520:  010074  040052  027525  060154  010076  070505  060160  006400
005530:  114577  177770  002002  006004  160232  010074  170232  003000
005540:  040001  002021  124563  063755  006400  114577  177767  002002
005550:  006004  144233  002400  044511  006020  027560  002004  027553
005560:  140232  003004  040505  002020  124563  160233  040511  002021
005570:  124563  160232  001727  130233  070171  060230  070172  160231
005600:  070173  060224  040045  002002  124560  060055  114565  114571
005610:  064177  160227  050050  074161  063753  170214  114572  027654
005620:  067314  074161  050060  027642  050055  027650  050056  002001
005630:  027645  060203  070472  160600  070474  017766  067641  060057
005640:  124540  000022  063644  124566  005616  050057  002001  124573
005650:  017766  160206  006400  027310  017766  063314  070161  160227
005660:  050050  003004  013666  002003  124551  027310  020000  160205
005670:  001222  000010  027616  060203  050513  027616  114570  027616
005700:  160233  003004  067754  074504  044057  140001  070505  017756
005710:  060161  073314  006400  063755  114577  177747  006002  002004
005720:  003004  040505  002020  124564  064504  044056  160001  006400
005730:  114577  177770  077756  001727  140233  070505  040511  002020
005740:  027744  037756  070505  027736  067756  005727  060505  030001
005750:  070171  034162  027575  000171  000126  000000  000000  164231
005760:  006121  027764  005100  007004  077755  127756  000000  060245
005770:  002020  003004  070245  127766  000000  003000  070001  040254
006000:  002021  026006  044100  006020  026006  124723  002400  070245
006010:  124541  000000  072125  076126  060161  072107  060162  072123
006020:  002400  070162  072130  060115  040075  064055  016252  016136
006030:  062123  002003  026114  064527  056057  026111  006020  026060
006040:  004065  006043  026111  076131  002041  026054  062130  052107
006050:  026054  016202  026054  016136  036130  056131  026040  047117
006060:  060156  003000  072130  062130  003000  052107  026072  016202
006070:  026072  016136  036130  026063  026111  062124  066110  114543
006100:  006123  060161  052107  026115  002400  070126  026115  000000
006110:  177765  062107  016202  000000  036011  060115  040075  006404
006120:  016252  066100  126011  000000  000000  000000  000000  000000
006130:  000000  000000  000000  000000  000000  000000  000000  062125
006140:  016244  062264  072124  062126  066124  016266  026075  066124
006150:  044055  160001  010073  040045  002020  044061  044056  076124
006160:  007000  144546  006020  026172  162124  002003  126136  050052
006170:  002001  026143  036125  062125  010073  003004  040116  002002
006200:  026137  126136  000000  070161  060074  070163  062237  070526
006210:  002400  070200  016244  062241  050273  060274  052242  060275
006220:  052243  002001  026226  060370  070200  016244  060370  070157
006230:  060200  002004  072125  036202  002400  070526  126202  006240
006240:  026234  051531  051524  042515  000000  066123  074162  006404
006250:  016252  126244  000000  070166  062264  070167  062263  070170
006260:  062265  114536  126252  000200  000270  000166  000000  072127
006270:  160001  152127  002001  026313  006004  036127  160001  152127
006300:  002001  026313  006004  036127  162127  010075  072127  160001
006310:  010075  052127  002001  036266  126266  000000  062327  072334
006320:  003400  066370  120001  006004  036334  026322  126315  177741
006330:  000000  076334  064050  114543  000000  126330  000000  064000
006340:  062352  016330  060270  052353  002001  126336  016315  050327
006350:  036336  126336  000273  046102  000000  070170  076334  064200
006360:  074166  062370  070167  062371  066334  034162  114536  126354
006370:  000270  000166  000000  072401  076402  034261  114545  000002
006400:  000001  000000  000000  006005  124551  002400  070261  126372
006410:  002750  160213  010056  050056  026431  032633  002311  001425
```

$ADDR

$SRCH

$LBL

DVRØS

DVR05

```
006420:  102611  016446  000015  016464  102711  002400  170220  126410
006430:  006424  120213  052577  026436  060057  126410  164214  006021
006440:  007004  174217  005300  174214  060055  026415  006422  164214
006450:  005200  174216  164215  007324  005010  007004  001310  026462
006460:  006003  007400  174217  126446  001676  062464  052430  026476
006470:  160206  002220  026476  034260  036464  126464  164213  005332
006500:  026611  102511  010073  050073  026541  052422  026537  050065
006510:  026562  006020  026537  050054  026544  164216  004065  134216
006520:  002041  001727  072446  060074  002340  001727  110001  032446
006530:  170001  160213  001421  134217  000040  001425  170213  030464
006540:  026606  016446  160213  026535  016553  044052  174216  160217
006550:  040052  170217  026537  000000  160214  001000  150216  026537
006560:  164216  126553  005421  174213  016553  004065  160001  010075
006570:  032534  002040  170001  160217  164215  002004  006021  001100
006600:  006020  007300  044000  106711  001521  102611  103711  002400
006610:  126464  062464  052430  026621  102511  010074  050074  026621
006620:  070260  002400  150216  026574  150217  026645  160216  134216
006630:  000065  004010  062630  160000  006051  001727  010074  134217
006640:  026643  052657  026574  102611  026537  062422  150220  060065
006650:  170220  052422  026643  006400  174215  026643  004000  000137
006660:  002753  064162  060155  006002  060161  073360  000065  073347
006670:  002441  063344  073352  060164  006002  060163  073353  063347
006700:  017211  000400  010067  002102  026737  160213  010056  001510
006710:  026714  067145  063336  026727  063337  002341  026726  160213
006720:  010077  001225  053340  026726  060057  126660  067332  077062
006730:  073346  060041  170220  062736  072743  026764  006737  060056
006740:  006400  074162  126660  001676  063347  106515  006003  026755
006750:  002400  004033  026755  002004  026751  017211  027115  067333
006760:  106606  164217  124001  006764  164214  006004  160001  002003
006770:  027071  073354  006004  160001  002021  003005  001100  002003
007000:  003400  073355  044051  160001  010075  073350  120001  073351
007010:  003004  040116  001727  001300  043355  002020  027034  001727
007020:  001200  010074  003004  040116  050116  062701  043350  070174
007030:  002400  006004  170001  027051  044055  003004  170001  043355
007040:  073355  044052  003004  043354  170001  044052  063350  042701
007050:  170001  017126  017241  067354  063346  053336  047343  106702
007060:  106602  067355  103714  017162  017241  063346  053337  017172
007070:  026764  060200  002302  000040  060162  067353  002041  002003
007100:  074164  002041  002002  074163  002400  070162  164215  006020
007110:  007004  126743  007171  063112  170217  002400  036743  126743
007120:  007167  033347  106715  102615  103715  127120  007052  063351
007130:  043335  002021  032701  002020  043334  033352  070001  063350
007140:  001767  017143  127126  007142  102614  103714  063353  002300
007150:  073353  063341  001225  017120  102314  027154  106614  103714
007160:  017112  127143  007064  102702  106602  103706  017120  106706
007170:  017112  127162  000000  017126  063355  003004  040073  001727
007200:  001200  010071  102614  103714  063342  017120  017112  017241
007210:  127172  006757  102106  103714  106715  102615  103715  102314
007220:  027217  102514  073357  010074  073356  063357  001226  002440
007230:  027237  160206  010075  033356  170206  037211  127211  070525
007240:  127211  007065  063357  002111  127241  001422  001727  000010
007250:  027262  001723  000312  027305  057340  127241  134220  027276
007260:  001200  002021  002405  060056  006400  074162  067360  074126
007270:  067351  074130  067350  074127  005727  126743  063252  006500
007300:  017143  002400  006500  017143  027051  001332  027315  067346
007310:  057336  027254  027313  102031  027104  017126  067345  106702
007320:  106602  063336  007400  103714  017162  063350  010075  073350
007330:  034525  027051  102114  120014  000014  177764  020000  010000
007340:  110400  030000  060000  100000  001000  107350  020000  000000
007350:  013000  000000  000000  000026  000270  177600  000000  100000
```

DVR31

007360:	000001	006410	006464	040011	102400	000000	000000	003475	DVR05
007370:	177777	000401	037634	000044	077470	177670	000000	000000	
007400:	000000	000000	010241	010313	000013	001440	000002	000000	DVR01
007410:	000000	000000	000000	000000	000000	000000	000000	000000	
007420:	010555	001002	001401	006660	006743	140014	014400	000000	DVR31
007430:	000000	003420	177777	000001	007726	000000	000000	000000	
007440:	000000	000000	000000	000000	010241	010320	000016	001000	DVR02
007450:	000000	000000	000000	000000	000000	000000	000000	000000	
007460:	000000	000000	010443	001002	001404	010241	011001	100022	DVR22
007470:	011000	000000	000000	000000	000000	000000	000000	000000	
007500:	000000	000000	000000	011075	001002	001407	000001	000043	DRT →
007510:	000003	000004	000002	000001	000205	000000	000000	000000	INT. TABLE
007520:	007361	000000	007402	000000	007423	007444	000000	000000	
007530:	000000	000000	007465	004003	000413	014004	000415	004005	EXEC MOD.
007540:	000421	004007	000423	004010	000425	010011	000427	004012	DOUBLET
007550:	001002	004014	001004	014015	001006	014016	001012	010017	TABLE
007560:	001016	004020	001021	010023	001023	010024	001026		

EQ
TABL

DRT
CONTINUED

$EX16 Entry

$EX03 Entry

$EX19 Entry

$EX20 Entry

$EX04 Entry

[SLIDE AP-14]

V. INTERNAL SYSTEM ORGANIZATION

A. Format for Disc Files

(1) Absolute (Core Image) [SLIDE 43].

(2) Relocatable [SLIDE 44].

(3) ASCII Source Statements [SLIDES 45A, 45B].

(4) ASCII or Binary Data.

a. System simply reserves space - does not set initial file contents to any value(s).

B. Disc Layout for Generation Example

TOPIC	MAIN PROJECTURE	AUX. PROJECTURE
Overall Disc Layout	5Ø	-----
System Label Sector	51	AD-1
Disc Resident Bootstrap	5Ø	AD-1
System Directory	5Ø,52	AD-1→AD-3
Core Res. Sys. (#2)	5Ø,53	AD-3→AD-1Ø
Core Res. Sys. (#3)	5Ø,54,55	AD-1Ø
Disc Res. Programs	5Ø,56	AD-1Ø→AD-24
Core Res. Sys. (#4)	5Ø,57,58	AD-25
Core Res. Sys. (#1)	5Ø,58	AD-26→AD-27
User Label Sector	5Ø,59	AD-28
User Directory	5Ø,6Ø,61	AD-28→AD-29

C. Detailed Description of System Base Page Communication Area. (Found in Appendix A of Operators' Manual). Description starts on next page. (Slides AP-1) (General Core Layout) and (AP-6) (Low Core of Memory Dump) will be used to relate values where possible.

DOS-M BASE PAGE LOCATIONS

LOCATION (S)	TYPE	CONTENTS
3	--	Start address for System Start-up (branched to indirect by Disc Resident Bootstrap following loading of Core Resident System).
4-37	--	JSB N,I where N is a Base Page Location containing the Central Interrupt Controller ($CIC) address.
4Ø	DEC	-64 (177ØØ)
41	DEC	-1Ø (17776)
42	DEC	-9 (17767)
43	DEC	-8 (17777Ø)
44	DEC	-7 (177771)
45	DEC	-6 (177772)
46	DEC	-5 (177773)
47	DEC	-4 (177774)
5Ø	DEC	-3 (177775)
51	DEC	-2 (177776)
52	DEC	-1 (177777)
53	DEC	Ø (Ø)
54	DEC	1 (1)
55	DEC	2 (2)
56	DEC	3 (3)
57	DEC	4 (4)
6Ø	DEC	5 (5)
61	DEC	6 (6)
62	DEC	7 (7)
63	DEC	8 (1Ø)

LOCATION	TYPE	CONTENTS	
64	DEC	9	(11)
65	DEC	10	(12)
66	DEC	17	(21)
67	DEC	64	(100)
70	OCT	17	(17)
71	OCT	37	(37)
72	OCT	77	(77)
73	OCT	177	(177)
74	OCT	377	(377)
75	OCT	177400	(177400)
76	OCT	3777	(3777)
77	OCT	177700	(177700)

LOCATION	LABEL	CONTENTS
100	UMLWA	Last word address of user available memory. Will always be one less than contents of location 123.
101	JBINS	Start TRACK/SECTOR of job binary area. =0 if job binary area not assigned. =-1 if this area overflows during compilation or assembly. = TRACK/SECTOR at end-of-disc for area assigned.
102	JBINC	Current TRACK/SECTOR of job binary area. Only set by compilers or assembler using this area.
103	TBG	Time Base Generator I/O Channel address. Will be 0 if TBG not on system.
104	CLOCK	Minutes part of System Time Clock.
105	CLOCK+1	Tenths of seconds part of System Time Clock.
106	CLEX	Minutes part of execution Time Clock. Bit 15 is set "ON" to turn this clock off.
107	CLEX+1	Tenths of seconds part of Execution Time Clock.
110	CXMX	Maximum allowable execution time. Set by :RUN Directive time parameter or to 5 if not given.

set to 6 in Athena system

LOCATION	LABEL	CONTENTS
111	BATCH	Logical Unit # of Batch Input Device. Set by :BATCH Directive.
112	SYSTY	Logical Unit # of System Teletype.
113	DUMPS	Abort/Post Mortem dump flags. Bit 15 --- Abort dump flag. Bit Ø --- Post mortem dump flag. Bit will be on if condition set. These bits will be set by :ADUMP and :PDUMP Directives and cleared by either their execution, :OFF Directive, or new :JOB Directive.
114	SYSDR	System Directory start TRACK/SECTOR. Set to where system is declared as starting during generation.
115	SYSBF	System Buffer TRACK/SECTOR. Since always on track boundary, sector part will always be Ø.
116	SECTR	Number of logical sectors per disc track.
117	EQTAB	Start Address of Equipment Table.
120	EQT#	Number of entries in entire Equipment Table. Each entry is 17 words.
121	LUTAB	Start address of Logical Unit Table.
122	LUT#	Number of entries in Logical Unit Table.
123	JBUF	Start address of Job Input Buffer.
124	JFILS	Start TRACK/SECTOR address of source file. Set by execution of :JFILE Directive.

LOCATION	LABEL	CONTENTS
125	JFILC	Current TRACK/SECTOR address of source file. Updated as Compiler or Assembler accesses the source file.
126-14Ø	RONBF RONBF+1 ⋮ RONBF+1Ø	Multi-purpose 11 word buffer used by system when user program is executing. Some uses are: (1) Saving of two 5 word user File directory entries to increase system efficiency when user program is running on only one subchannel. System looks here first for Directory entry before searching Disc Directory. (2) Contains actual parameter valves (P1,P2,...) following :PROG, and :GO directives. (3) Information is passed to $EX20 (Parity Error Processor) by these locations.
141-153	EXPG ⋮ EXPG+1Ø	Directory Entry for currently executing USER program. For MAIN programs having segments: The first 2 1/2 words will always be those of the MAIN program's Directory entry (File Name in ASCII) with the remaining 8 1/2 words equal to the segment currently executing Directory entry information.
154	DISCO	Bits 11-15 (Disc Data Channel select code). Bits Ø-1Ø (Last Track on System). just 6 (as 13+ spin
155	SYSSC	System Disc Subchannel number. Will always be equal to S.C. bootstrapped down from.

LOCATION	LABEL	CONTENTS
* 156	SCCNT	Number of Subchannels on System -1.
157	UDNTS	Next TRACK/SECTOR address on Current User Disc.
16Ø	SYNTS	Next TRACK/SECTOR address on System Disc. Will always equal the start of Work Area.
* 161	CUDSC	Current User Disc subchannel number.
162	CRFLG	Current Disc request flag.(Ø for System Disc; ≠Ø for Current User Disc). DVR31 always clears on completion of Disc request and examines on entry to see what disc to access.
163	CUDLA	Current User Disc TRACK/SECTOR address last accessed. Only used by DVR31.
164	SDLA	System Disc TRACK/SECTOR address last accessed. Only used by DVR31.
165	CUMID	Computer identification code. (≠Ø if computer is 2114B thus only having one DMA channel).
166-17Ø	DBUFR	System Disc Request Parameter Buffer. DBUFR = TRACK/SECTOR DBUFR+1 = BUFFER ADDRESS DBUFR+2 = NUMBER OF WORDS (Set by System prior to Disc I/O for DVR31 to use).

LOCATION	LABEL	CONTENTS
171-173	UBUFR	Current User Disc Request Paramenter Buffer. UBUFR = TRACK/SECTOR UBUFR+1 = BUFFER ADDRESS UBUFR+2 = NUMBER OF WORDS (set by System prior to Disc I/O for DVR31 to use).
174	TSONE	Last referenced TRACK/SECTOR address +1. Set by DVR31. Could be used by User program accessing the WORK AREA to see what next available TRACK/SECTOR address is.
175	GUDSC	Default User Disc Subchannel number. Always follows System Disc Subchannel number when Default User Disc is on same subchannel as System. (like when :DD executed), otherwise it stays where started W.R.T. Bootstrapped System.
176	SYSCD	System Generation Code.
* 177	JFLSC	Current Source File Subchannel number. Set by :JFILE Directive.
200	DISCL	User label TRACK/SECTOR address. =Ø if Current User Disc is not on System Disc. If Current User Disc is on System Disc this Disc address = System Buffer Sector address. Incrementing this Disc address by one sector always gives the start of the User Directory TRACK/SECTOR address on the Current User Disc.

LOCATION	LABEL	CONTENTS
201	INTAB	Start address of Interrupt Table.
202	INT#	Number of Interrupt Table entries.
203-223	EQT1 EQT2 ⋮ EQT17	Addresses of Current Equipment Table Entry
224	RQCNT	Number of request parameters in current EXEC call. JSB EXEC and DEF RTN are not counted.
225	RQRTN	Request return address in current EXEC call.
226-235	RQP1 RQP2 ⋮ RQP8	Addresses of current request parameters. RQP1 is for the request code address etc.
236	NABRT	Illegal request code abort/no abort option parameter. ≠Ø if set. Set by N parameter in :RUN Directive. Stays in effect in current job.
237	XA	A Register contents at time of interrupt.
240	XB	B Register contents at time of interrupt.
241	XEO	E (Bit 15) and Ø (Bit Ø) Register contents at time of interrupt.
242	XSUSP	Address at time of interrupt (P-Register)
243	EXLOC	Start address of EXEC MODULE DOUBLET TABL

LOCATION	LABEL	CONTENTS
244	EX#	Number of entries in EXEC MODULE DOUBLET TABLE.
245	EXMOD	EXEC MODULE currently in EXEC MODULE overlay area. = Ø if none resident. = +N if module #N resident and available. = -N if module #N resident and BUSY.
246	EXMAN	EXEC MODULE overlay area low Main Core Address.
247	EXMAN+1	EXEC MODULE overlay area high Main Core Address.
25Ø	EXBAS	EXEC MODULE Base Page linkage low address.
251	EXBAS+1	EXEC MODULE Base Page linkage high address.
252	IODMN	START ADDRESS OF I/O Driver Main overlay area.
253	IODBS	Start address of I/O Driver Base Page overlay area.
254	UMFWA	Start address of User Main Area.
255	UBFWA	Start address of User Base Page Linkage Area.
256	UBLWA	Last word address of User Base Page Linkage Area.
257	CHAN	Current DMA channel number assigned. = Ø if no DMA in use (not really true; last value remains)

LOCATION	LABEL	CONTENTS
26Ø	OPATN	Operator attention flag. = Ø for not set. ≠ Ø if desired. Set by System TTY Driver.
261	OPFLG	System TTY busy flag. = Ø if not busy. ≠ Ø if busy.
262	SWAP	Job Processor resident flag. BIT 15 = 1 if System TTY is Batch Device. BIT Ø = 1 if Job Processor is in core.
263	JOBPM	Job Processor start TRACK/SECTOR address.
264	JOBPM+1	# of words in MAIN section of Job Processor.
265	JOBPB	# of words in Base Page Linkage for Job Processor.
266	EJOBF	End-of-Job flag used only by Job Processor. = "blanks" if re-entry of :DATE allowed = Ø if in a job. = 1 if between jobs. = -1 if end-of-job.
267	RTRK	Real Time simulation track #.
27Ø-467	$BUF $BUF+1 ⋮ $BUF+127	128 Word System I/O Buffer. Used only by Monitor and EXEC modules.
47Ø	$GOPT	Point of suspension return address. Contains return address when $IDCD (location 471 below) = GO.

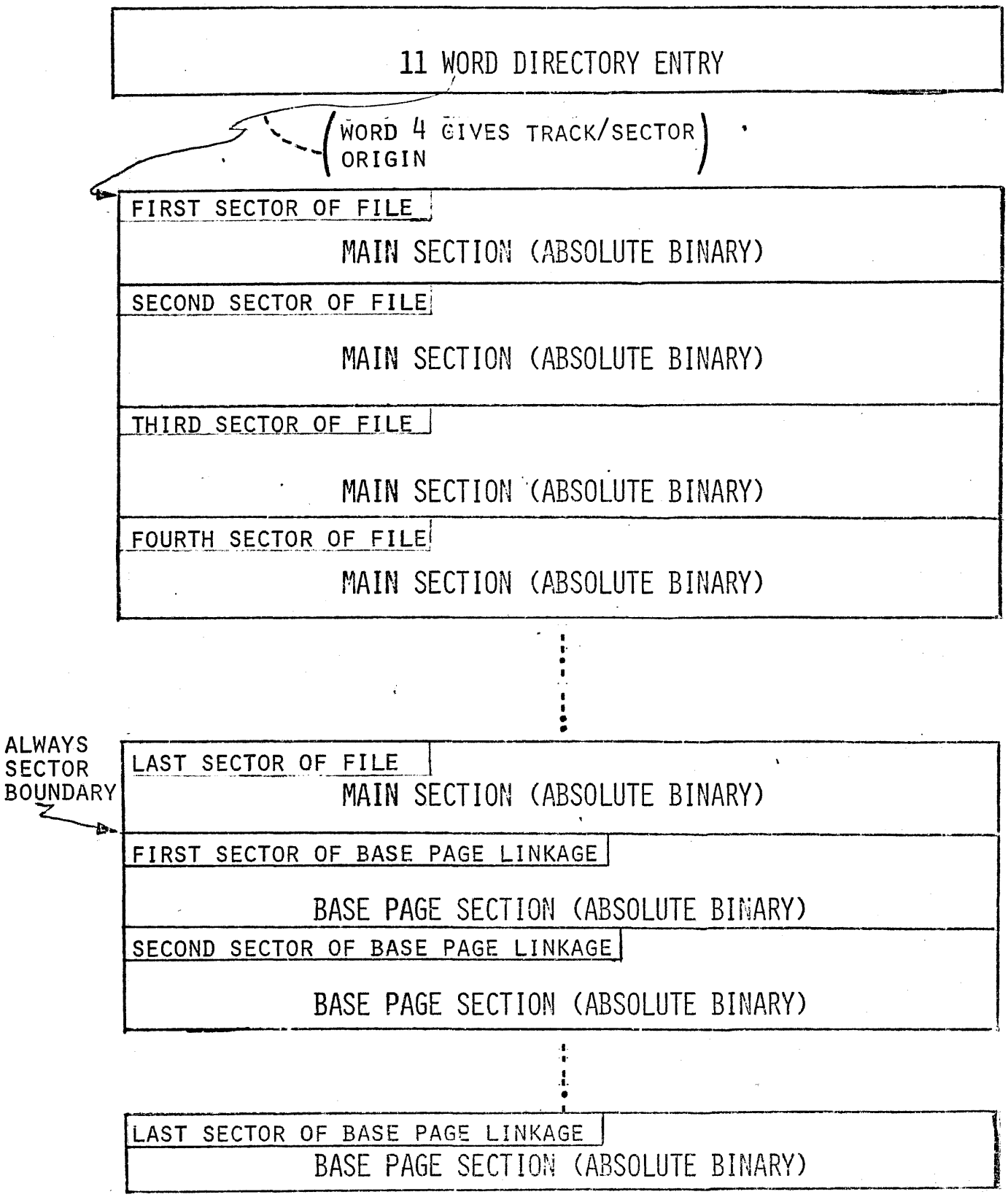
LOCATION	LABEL	CONTENTS
471	$IDCD	Input request code check characters. = Ø for no special restrictions. ≠ Ø for special restrictions placed on what can be entered via system TTY keyboard (like DA, GO, etc...).
472-473	$MDBF	2 Word EXEC Module Data Buffer.
474-5Ø2	TEMP TEMP+1 ⋮ TEMP+6	System Temporary.
5Ø3	TEMPØ	System Temporary.
5Ø4	TEMP1	System Temporary.
5Ø5	TEMP2	System Temporary.
5Ø6	TEMP3	System Temporary.
5Ø7	TEMP4	System Temporary.
51Ø	TEMP5	System Temporary.
511	MSECT	Negative # of logical SECTORS per TRACK.
512	VADR	Address of last instruction that caused a memory protect violation.
513	IODMD	I/O Driver Overlay Area resident flag. = Ø if no I/O Driver in this area. ≠ Ø if an I/O Driver is in this area. The value (if not Ø) will be: + (Address of resident Driver's first EQT entry) if area is available OR - (Address of resident Driver's first EQT entry) if this area is not available

LOCATION	LABEL	CONTENTS
514	RCODE	Current request code value. Will always be positive.
515	SXA	Operator attention A Register save.
516	SXB	Operator attention B Register save.
517	SXEO	Operator attention E (Bit 15) and O (Bit Ø) Register save.
52Ø	SXSUS	Operator attention return address save (P-Resister).
521	SEQTI	Operator attention EQT Table address save.
522	DSCLB	Disc TRACK/SECTOR Address of Disc Resident Relocatable Library. Used by Relocating Loader.
523	DSCL#	Number of sectors in Disc Resident Relocatable Library.
524	LSTCH	Last Disc referenced flag. = Ø if current user program (to be executed by :PROG or :RUN) is on System Disc. ≠ Ø if current user program (to be executed by :PROG or :RUN) is on Current User Disc.
	NOTE:	$EXIO (Program Load) uses to see how to set CRFLG flag (location 162).
525	FLFLG	User file table validity flag. (= Ø if invalid; ≠ Ø if valid). $EXII uses to see if OK to use $BUF area for user file directory entry storage.

LOCATION	LABEL	CONTENTS
526	XFLG	Transfer Address for Disc Not Ready condition. = Ø to process Not Ready condition normally. ≠ Ø to transfer to this address if Not Ready condition present. A good use is to ignore "NOT READY" Drives when doing multiple Drive System Searches.
527	SSFLG	System Search Flag Values it can have are: a. ASCII "NO" if :SS Directive not allowed. b. Ø for only current user Disc (:SS,99 condition). c. -X for full System Search (:SS) where X= # subchannels on system -1. d. +X for Selected System Search Bits Ø-7 are used to represent Subchannels Ø-7 respectively. Bit ON=OK, Bit OFF=not OK.
53Ø-531	CHARC	System Temporary.

DOSM ABSOLUTE DISC FILE FORMAT

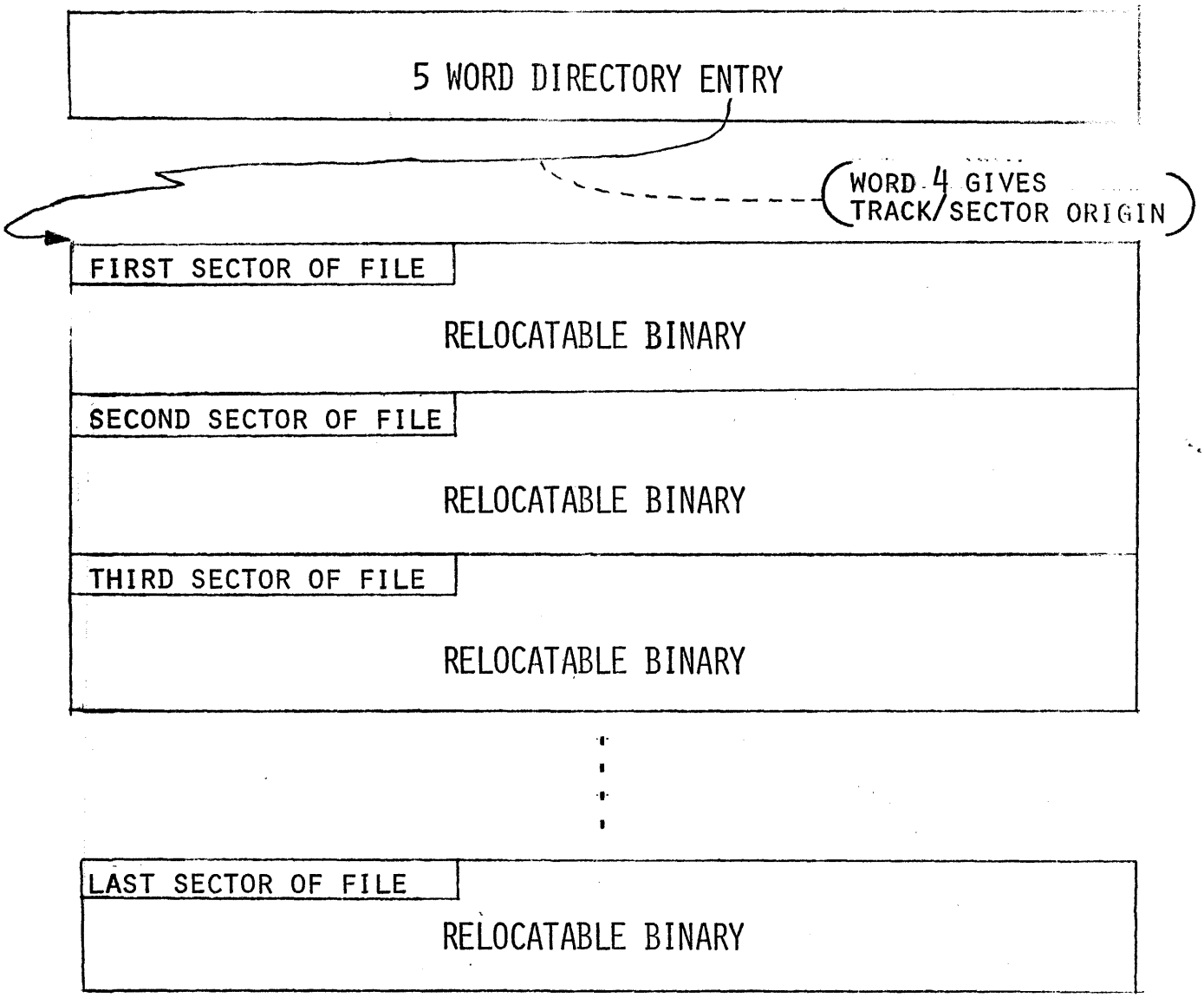
(ENTRY TYPES 1, 2, 3, 4, AND 5)



[SLIDE 43]

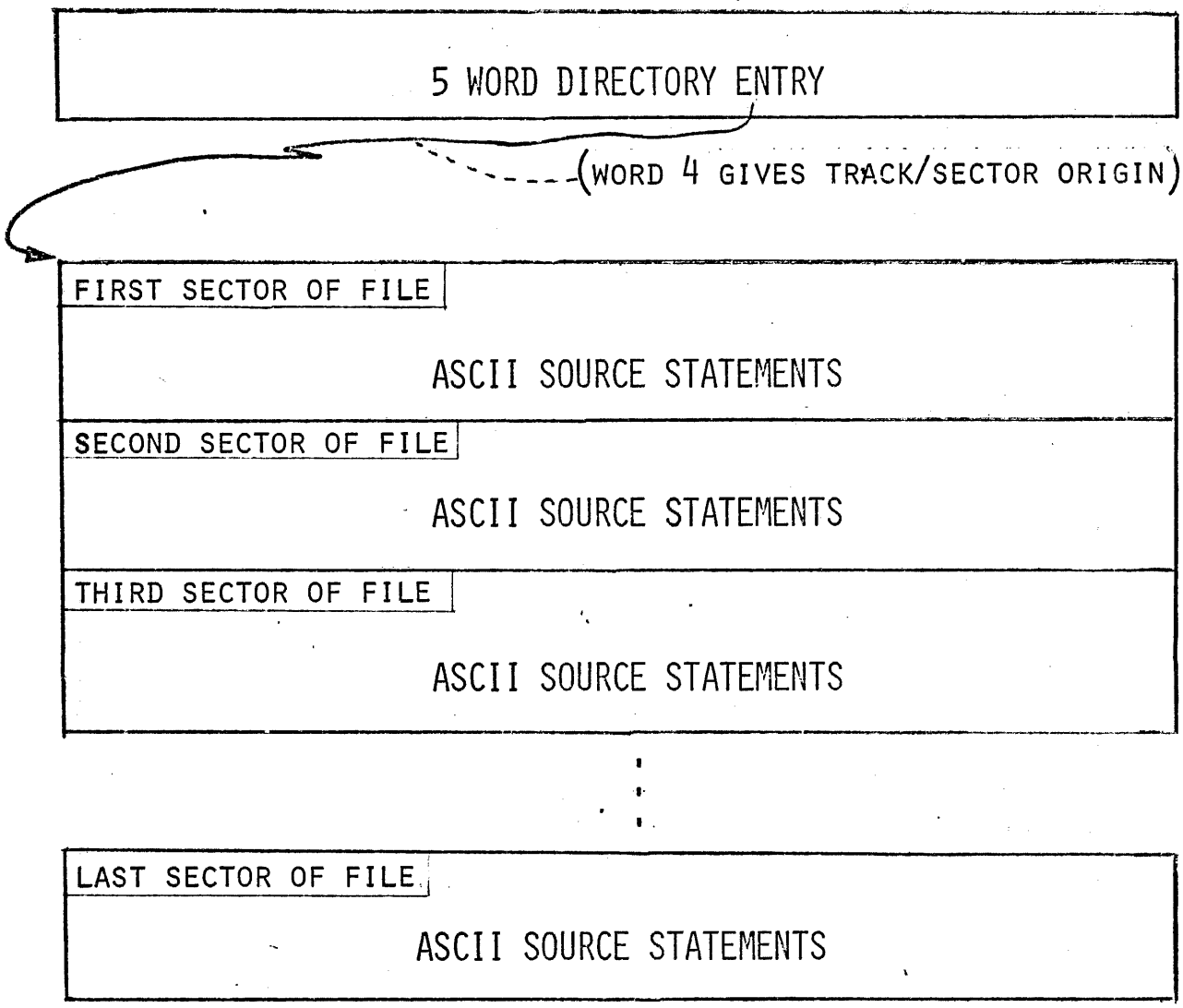
DOSM RELOCATABLE DISC FILE FORMAT

(ENTRY TYPES 6, 7 AND 8)

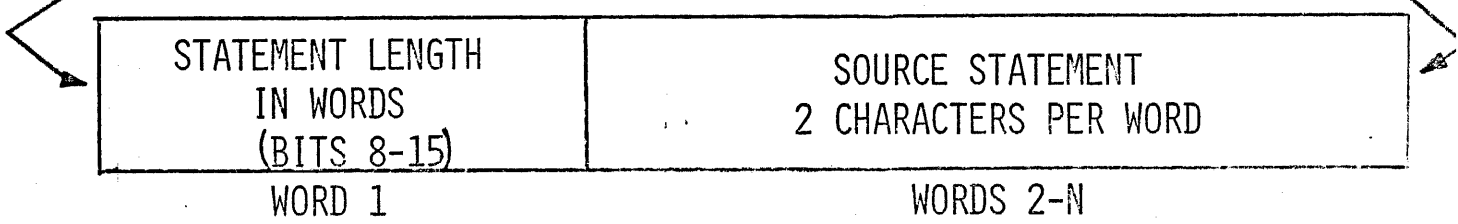


NOTE: "NAM" RECORD LENGTH FOR RTE/DOS/DOSM SYSTEMS IS 17 WORDS IN LENGTH WHICH IS INCOMPATABLE TO "NAM" RECORD LENGTH OF 9 WORDS OF BCS SYSTEMS.

DOSM ASCII SOURCE STATEMENT DISC FILE FORMAT
(ENTRY TYPE 9)



FORMAT FOR EACH SOURCE STATEMENT



[SLIDE 45A]

ASCII SOURCE FILE FORMAT EXAMPLE

```
INPUT :DATE,XXXXXXXXXX,H,M

@:DA,27.OCT.70,9,15
SUBCHAN=1
LBL=QQQQQ
@
:JOB,ASCII
JOB ASCII 27.OCT.70  TIME=0555 MIN. 16.0 SECS.
@
:ST,S,SORSE,1

AAAA
BBBBB
CCCCCC
DDDDDDD
::
0004 LINES
@
:LIST,U,1,SORSE
NAME TYPE SCTRS DISC ORG PROG LIMITS   B.P. LIMITS   ENTRY LIBR.  P-BIT
USCHAN=1
SORSE SS  0001  T055 003
@
:SO,55,3
001 001000 040501 040501 001400 041102 041102 041040 001400
    041503 041503 041503 002000 042104 042104 042104 042040
    000000 177777 020000 036164 000000 041456 031062 020000
    036164 000000 046117 040504 051000 036155 000000 163252
    041120 051000 036146 000000 040523 046502 020000 036137
    000000 037501 051503 047000 036137 000000 037501 051515
    041000 036137 000000 037502 047103 047000 036137 000000
    037502 050113 052400 036137 000000 037503 044117 050000
    033403 000000 000312 000024 177767 040523 041511 044440
    031067 027117 041524 027067 030040 000000 000000 000000
    000000 000000 000000 000000 000000 000005 000001 000003
    000003 000004 000002 000001 000005 000000 000000 020473
    050521 050421 044456 031461 020000 037515 051531 046400
    036137 000000 037522 046125 047000 036137 000000 037501
    043114 043400 036137 000000 037514 051524 046000 036137
    000000 037514 052516 044400 036137 000000 037522 051502

@
```

SOURCE FILE CREATED

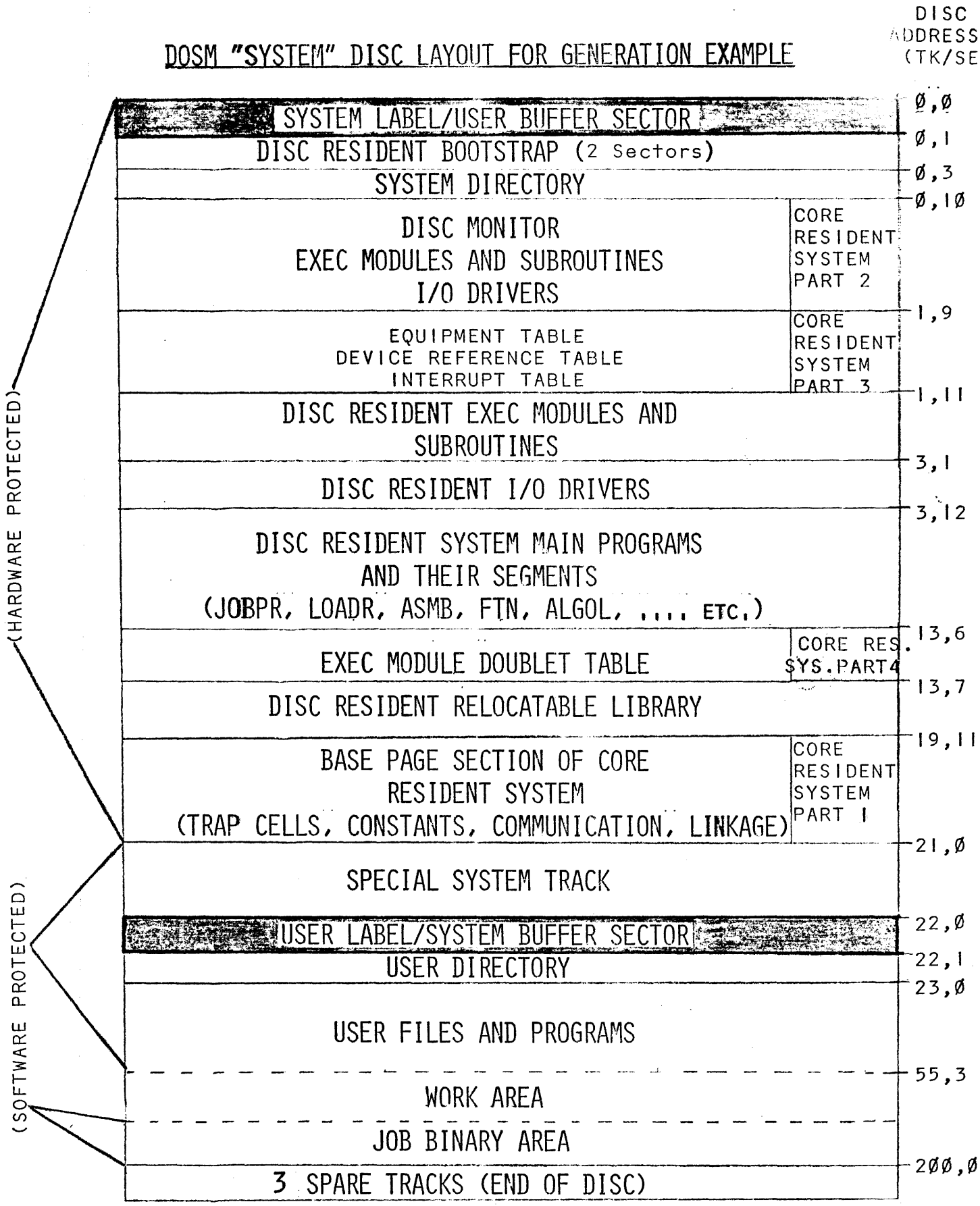
3 WORDS

3 WORDS

2 WORDS

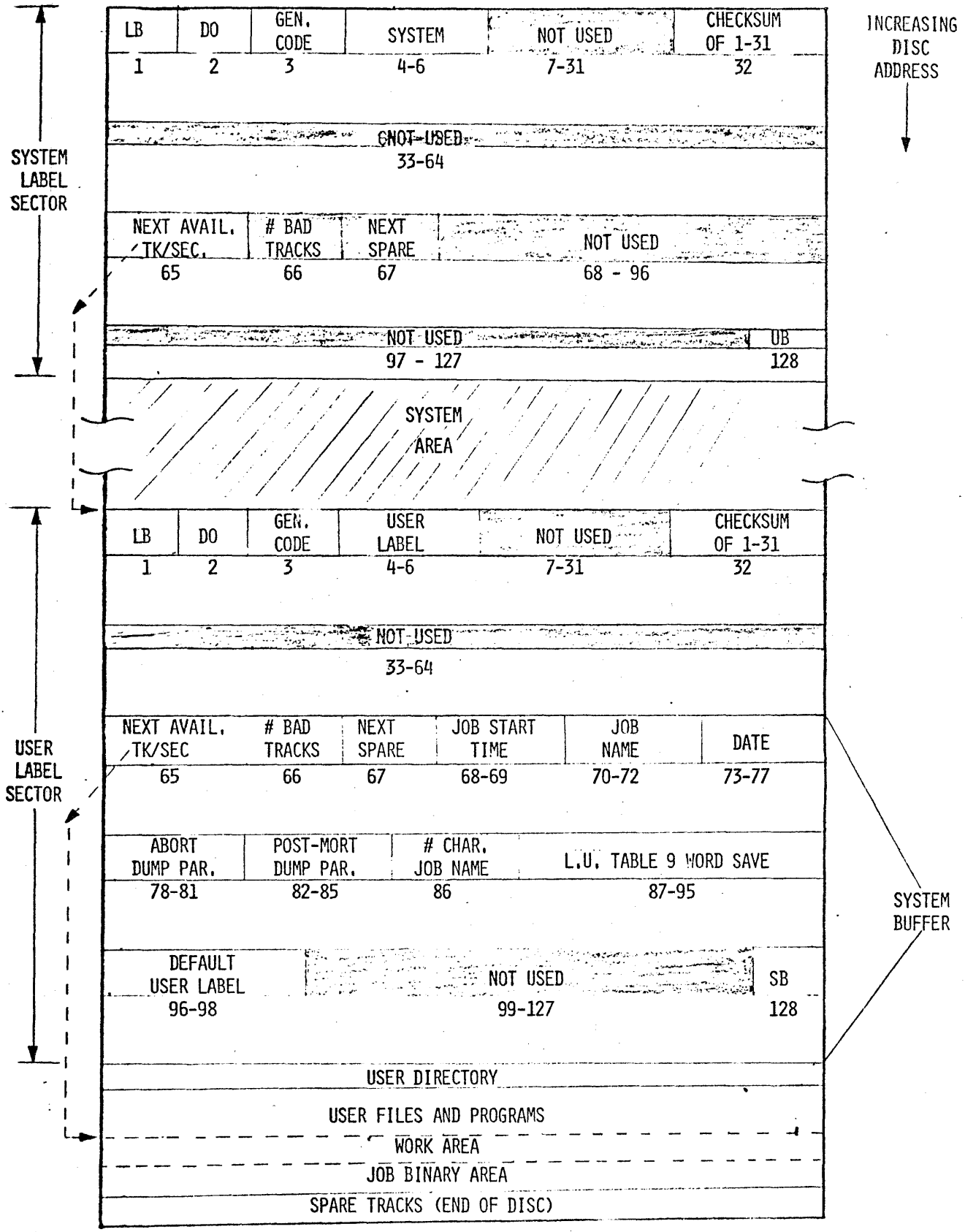
EOF

DOSM "SYSTEM" DISC LAYOUT FOR GENERATION EXAMPLE



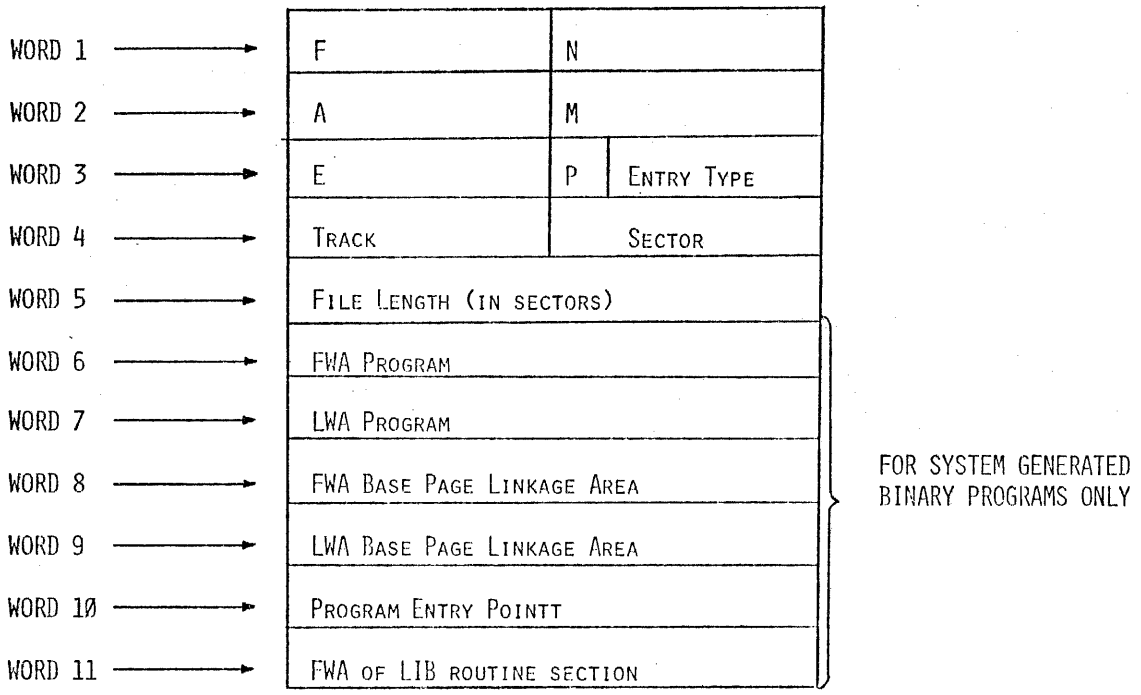
[SLIDE 50]

"SYSTEM" DISC (LABEL SECTORS DESCRIPTION)



[SLIDE 51]

DIRECTORY ENTRY FORMAT



ENTRY TYPE	FILE
0	SYSTEM RESIDENT
1	DISC RESIDENT EXECUTIVE SUPERVISOR MODULE
2	CURRENTLY UNUSED
*3	USER PROGRAM, MAIN
4	DISC RESIDENT DEVICE DRIVER
*5	USER PROGRAM, SEGMENT
6,7	LIBRARY
10_8	RELOCATABLE BINARY
11_8	ASCII SOURCE STATEMENTS
12_8	BINARY DATA
13_8	ASCII DATA

'P' BIT

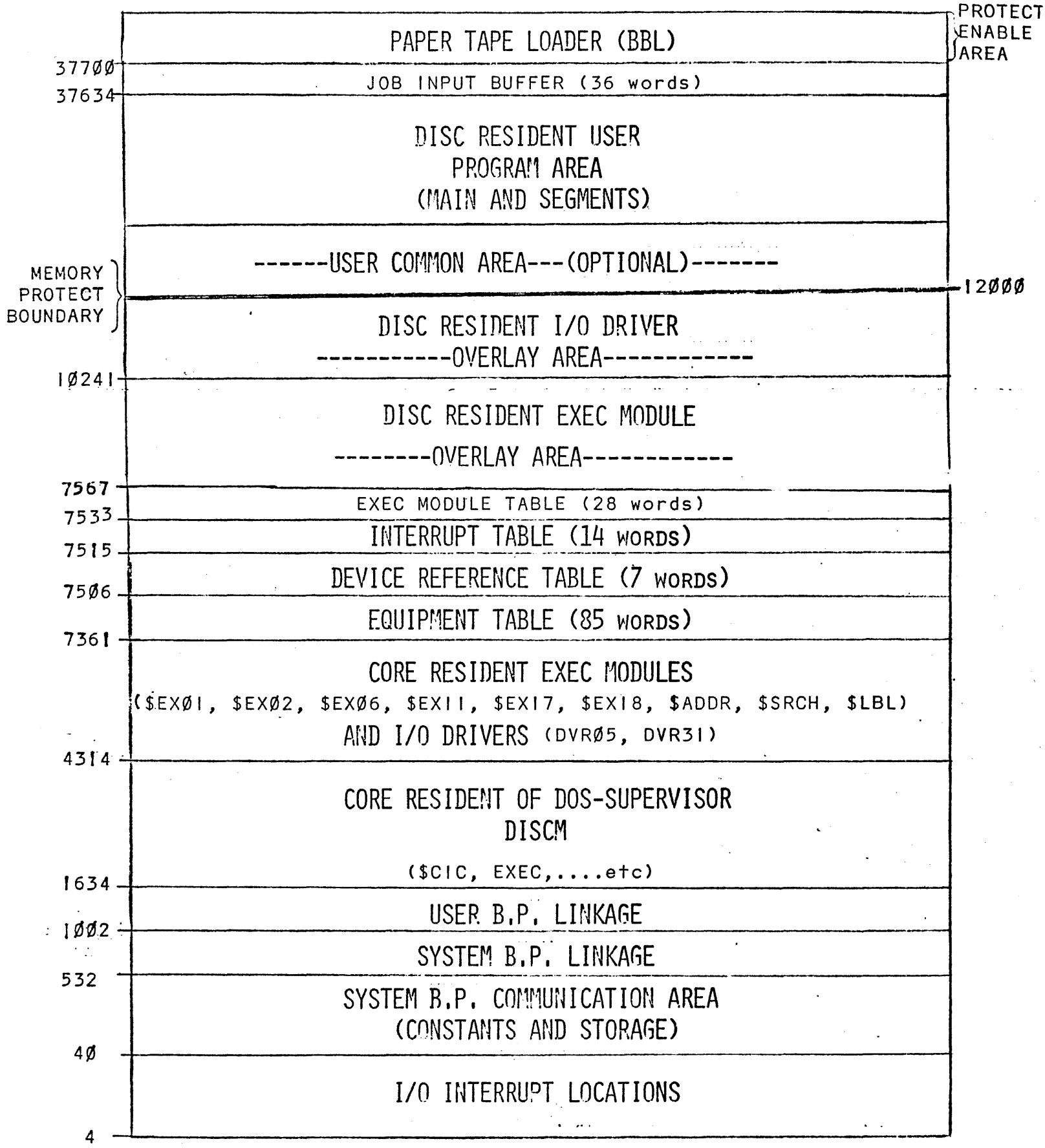
0 = NO ACTION

1 = PURGE THIS ENTRY AT THE END OF THE JOB OR FOLLOWING ANY EXECUTION OF :PU DIRECTIVE. THIS BIT IS SET BY THE LOADER AND CLEARED BY A :STORE,P,[file-name] REQUEST.

THE LAST DIRECTORY ENTRY IN EACH SECTOR IS FOLLOWED BY A WORD CONTAINING '-1' UNLESS THE GIVEN SECTOR IS EXACTLY FILLED WITH ENTRIES.

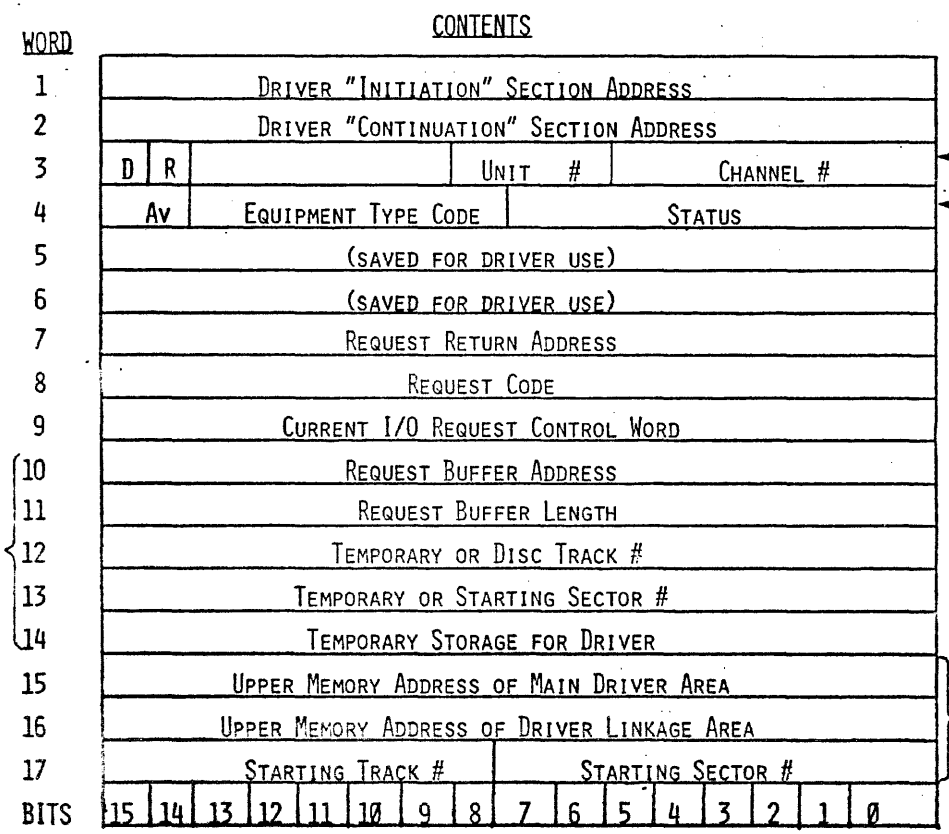
THE LAST ENTRY IN THE DIRECTORY IS FOLLOWED BY A WORD CONTAINING ZERO.

CORE MAP FOR DOSM SYSTEM GENERATION EXAMPLE (16K)



[SLIDE 53]

EQUIPMENT TABLE ENTRY FORMAT



D = 1 if DMA channel required.
R = 1 if driver type is core-resident.
Unit # May be used for sub-channel addressing.
Channel # I/O select code for device (lower number if multiboard interface.)

Av = Ø - Unit not busy and available
= 1 - Unit disabled (down)
= 2 - Unit busy
= 3 - Unit waiting for an available DMA channel

(This field set by system)

Status - Actual or simulated unit status at end of operation. (Driver must set this field)

Equipment Type Code - Identifies type of device and associated software driver. Assigned equipment type codes in octal ar

Ø0-07	Papter Type Devices
Ø0	Teleprinter
Ø1	Punched Tape Reader
Ø2	High Speed Punch
Ø5	Teletype (System)
10-17	Unit Record Devices
10	Reserved for Plotter
12	Line Printer
15	Mark Sense Card Reader
20-37	Magnetic Tape/Mass Storage and other devices capable of both input and output.
22	3030 Magnetic Tape
31	Moving-Head Disc

For equipment type codes 1 through 17, odd number indicate input device and even number indicate output devices (except Ø5, which is both input and output).

[SLIDE 54]

DEVICE REFERENCE TABLE FORMAT

EACH ENTRY IN THIS TABLE REQUIRES ONLY ONE WORD IN MEMORY. THE VALUE OF EACH ENTRY (DECIMAL NUMBER, 1-63) ASSOCIATES A LOGICAL UNIT NUMBER WITH AN EQUIPMENT TABLE ENTRY FOR THE SYSTEM IN THE FOLLOWING MANNER:

SEQUENCE IN MEMORY TABLE	LOGICAL UNIT #	FUNCTION
1	1	SYSTEM TELEPRINTER
2	2	USER MASS STORAGE
3	3	SYSTEM MASS STORAGE
4	4	STANDARD PUNCH DEVICE
5	5	STANDARD INPUT DEVICE
6	6	STANDARD LIST DEVICE
7-63	7-63	ANY DEVICE

INTERRUPT TABLE FORMAT

EACH ENTRY IN THIS TABLE REQUIRES ONLY ONE WORD IN MEMORY AND IS ASSOCIATED WITH EACH I/O CHANNEL IN THE COMPUTER (STARTING WITH LOCATION 6) WHICH CAN CAUSE AN INTERRUPT. EACH LOCATION IN THIS TABLE HAS AN ENTRY VALUE. MEMORY LOCATIONS ARE ASSOCIATED IN CONSECUTIVE INCREASING ORDER WITH AN I/O CHANNEL. TABLE VALUES ARE ZERO FOR AN I/O CHANNEL NOT REQUIRING INTERRUPT. I/O CHANNELS REQUIRING INTERRUPT CONTAIN THE START ADDRESS OF THE EQUIPMENT TABLE ENTRY OF THE ASSOCIATED DEVICE.

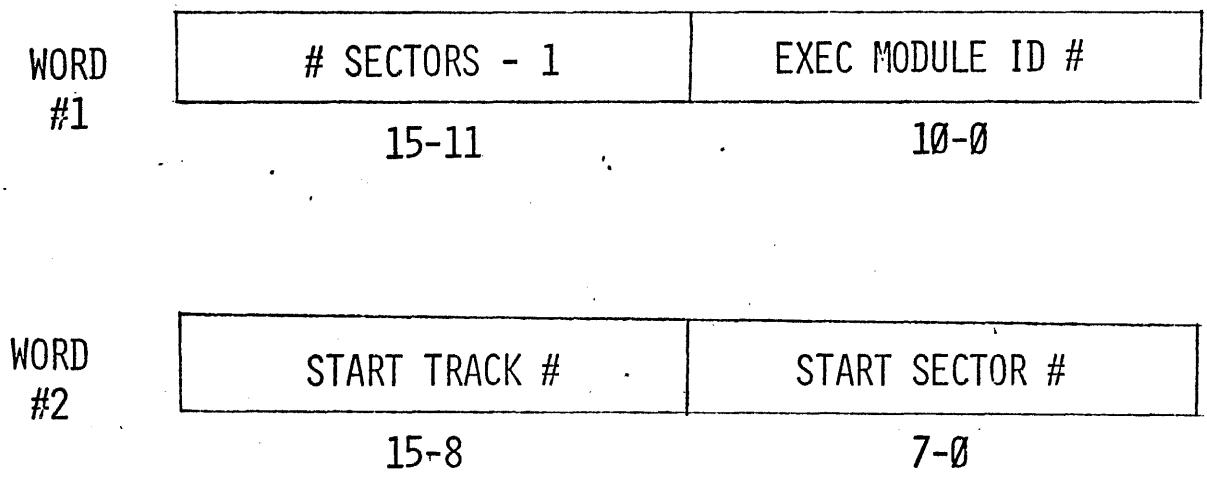
SYSTEM DIRECTORY LISTING FOR GENERATION EXAMPLE

NAME	TYPE	SCTRS	DISC	ORG	PROG LIMITS		B.P. LIMITS		ENTRY	LIB.	P-BIT
SUBCHAN=1											
$EX03	XS	0002	T001	011	07567	07624	00732	00733	07567	07624	
$EX04	XS	0004	T001	013	07567	10176	00732	00741	07567	10176	
$EX05	XS	0002	T001	017	07567	07745	00732	00733	07567	07745	
$EX07	XS	0002	T001	019	07567	07746	00732	00733	07567	07746	
$EX08	XS	0002	T001	021	07567	07732	00732	00733	07567	07732	
$EX09	XS	0003	T001	023	07567	10142	00732	00763	07567	10142	
$EX10	XS	0002	T002	002	07567	07745	00732	00733	07567	07745	
$EX12	XS	0002	T002	004	07567	07761	00732	00733	07567	07761	
$EX13	XS	0004	T002	006	07567	10223	00732	00754	07567	10223	
$EX14	XS	0004	T002	010	07567	10241	00732	00751	07567	10241	
$EX15	XS	0003	T002	014	07567	10153	00732	00763	07567	10153	
$EX16	XS	0002	T002	017	07567	07722	00732	00733	07567	07722	
$EX19	XS	0003	T002	019	07567	10107	00732	01000	07567	10107	
$EX20	XS	0003	T002	022	07567	10167	00732	00761	07567	10167	
DVR01	DR	0003	T003	001	10241	10555	01000	01002	10241	10555	
DVR02	DR	0003	T003	004	10241	10443	01000	01002	10241	10443	
DVR22	DR	0005	T003	007	10241	11275	01000	01002	10241	11275	
LOADR	UM	0038	T003	012	12000	21932	01002	01405	12000	21932	
JOBPR	UM	0038	T004	020	12000	22463	01002	01414	12000	22463	
ASMB	UM	0023	T006	010	12000	17120	01002	01362	16522	17120	
ASMB0	US	0004	T007	009	17127	17647	01362	01363	17442	17647	
ASMB1	US	0006	T007	013	17366	20542	01362	01424	17366	20542	
ASMB2	US	0007	T007	019	17345	20550	01362	01410	17351	20550	
ASMB3	US	0003	T008	002	17473	17771	01362	01363	17630	17771	
ASMB4	US	0004	T008	005	17366	20027	01362	01371	17366	20027	
ASMB5	US	0006	T008	009	17345	20425	01362	01404	17351	20425	
FTN	UM	0006	T008	015	12000	13127	01002	01047	12000	13127	
FTN01	US	0031	T008	021	13254	22120	01047	01502	16550	22120	
FTN02	US	0025	T010	004	13254	21027	01047	01356	13741	21027	
FTN03	US	0024	T011	005	13254	20600	01047	01277	15117	20600	
FTN04	US	0025	T012	005	13254	20750	01047	01360	13702	20750	
LIBRY	LR	0147	T013	007							

[SLIDE 56]

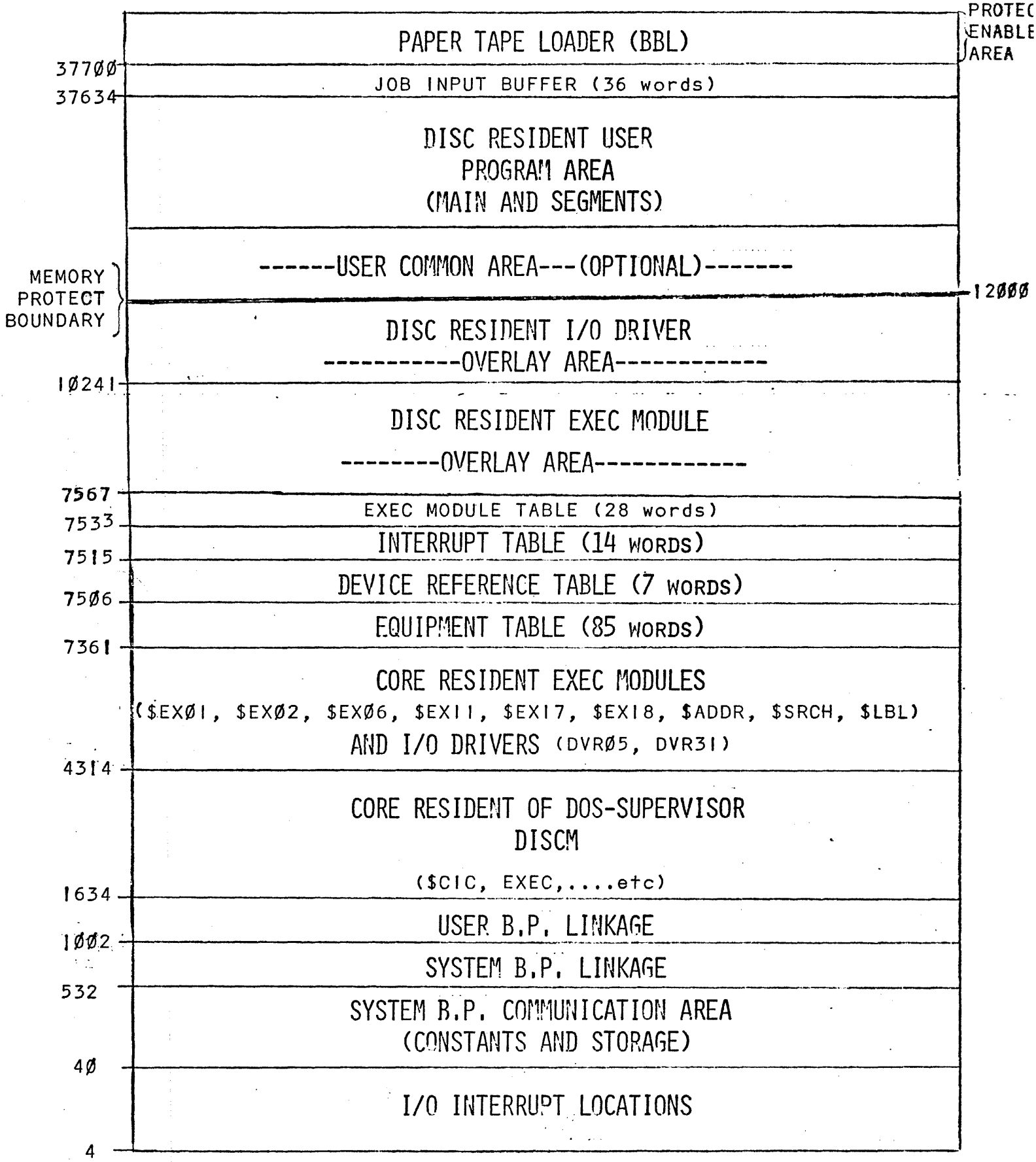
EXEC MODULE DOUBLET TABLE FORMAT

(TWO WORDS PER DISC RESIDENT EXEC MODULE)



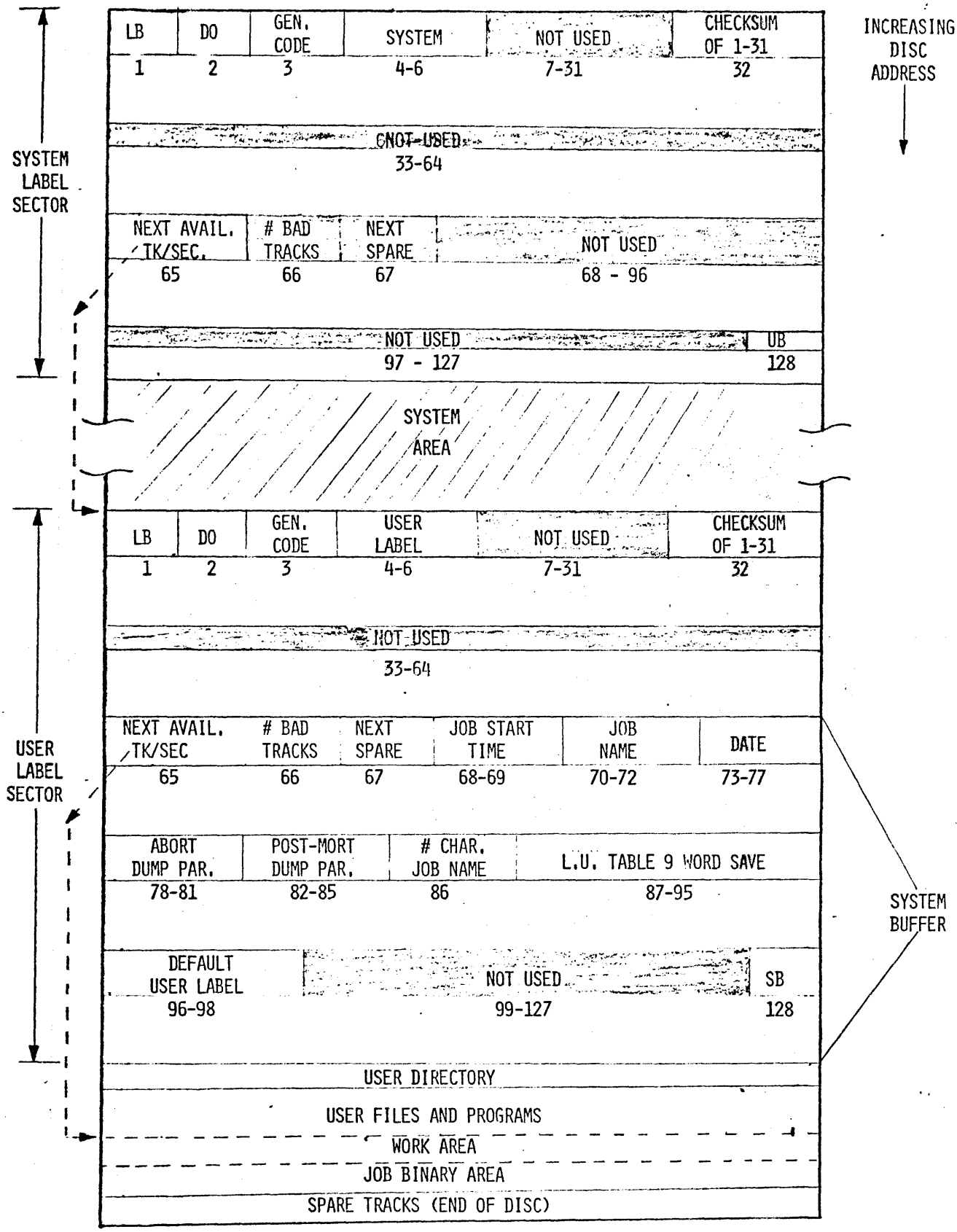
[SLIDE 57]

CORE MAP FOR DOSM SYSTEM GENERATION EXAMPLE (16K)



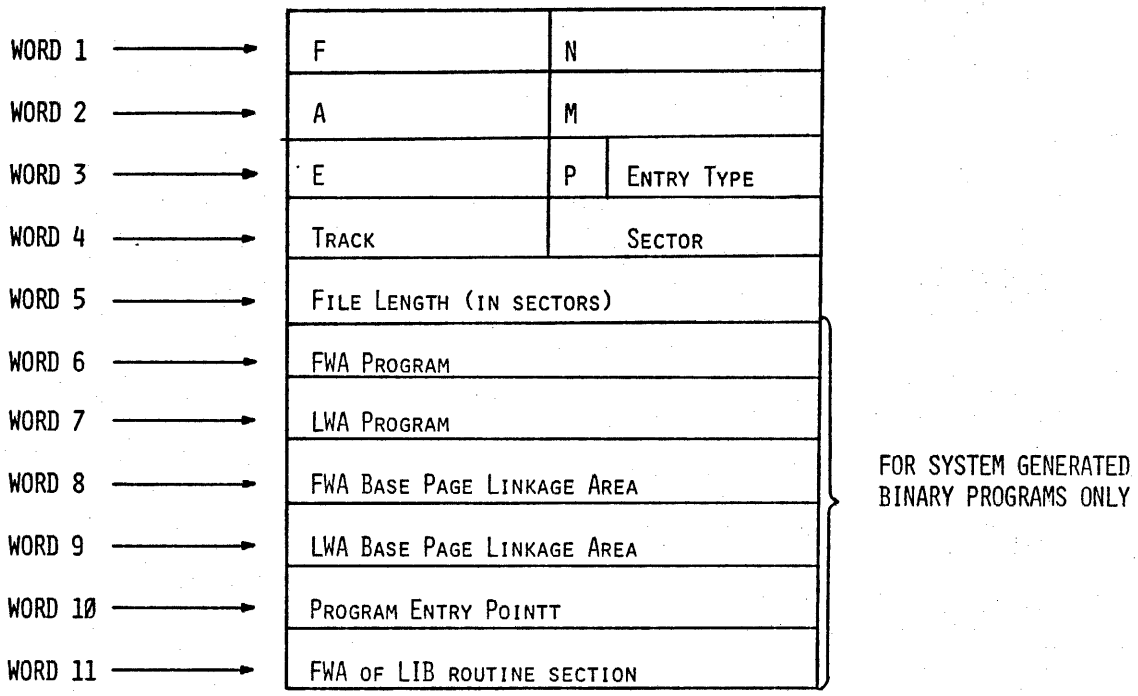
[SLIDE 58]

"SYSTEM" DISC (LABEL SECTORS DESCRIPTION)



[SLIDE 59]

DIRECTORY ENTRY FORMAT



ENTRY TYPE	FILE
0	SYSTEM RESIDENT
1	DISC RESIDENT EXECUTIVE SUPERVISOR MODULE
2	CURRENTLY UNUSED
3	USER PROGRAM, MAIN
4	DISC RESIDENT DEVICE DRIVER
5	USER PROGRAM, SEGMENT
6,7	LIBRARY
10_8	RELOCATABLE BINARY
11_8	ASCII SOURCE STATEMENTS
12_8	BINARY DATA
13_8	ASCII DATA

'P' BIT

0 = NO ACTION

1 = PURGE THIS ENTRY AT THE END OF THE JOB OR FOLLOWING ANY EXECUTION OF :PU DIRECTIVE. THIS BIT IS SET BY THE LOADER AND CLEARED BY A :STORE,P,[file-name] REQUEST.

THE LAST DIRECTORY ENTRY IN EACH SECTOR IS FOLLOWED BY A WORD CONTAINING '-1' UNLESS THE GIVEN SECTOR IS EXACTLY FILLED WITH ENTRIES.

THE LAST ENTRY IN THE DIRECTORY IS FOLLOWED BY A WORD CONTAINING ZERO.

USER DIRECTORY LISTING IN GENERATION EXAMPLE SYSTEM

NAME	TYPE	SCTRS	DISC	ORG	PROG LIMITS		B.P. LIMITS		ENTRY	LIBR.	P-BIT
SUBCHAN=1											
XREF	UM	0013	T023	000	12000	14750	01002	01036	12000	14071	
EOT1	SS	0001	T023	013							
WEOT	UM	0002	T023	014	12000	12013	01002	01003	12000	12013	
XREFR	RB	0016	T023	016							
DISCM	RB	0020	T024	008							
EXECS	RB	0063	T025	004							
DVR05	RB	0003	T027	019							
DVR31	RB	0005	T027	022							
LIBRY	RB	0143	T028	003							
DVR02	RB	0002	T034	002							
DVR01	RB	0003	T034	004							
DVR22	RB	0007	T034	007							
LODR	RB	0049	T034	014							
JOBP	RB	0065	T036	015							
ASMBL	RB	0040	T039	008							
ASMD	RB	0004	T041	000							
ASM3	RB	0004	T041	004							
ASM4	RB	0006	T041	008							
ASM5	RB	0010	T041	014							
FRTN	RB	0008	T042	000							
FTN1	RB	0048	T042	008							
FTN2	RB	0045	T044	008							
FTN3	RB	0042	T046	005							
FTN4	RB	0031	T047	023							
ASM1	RB	0012	T049	006							
ASM2	RB	0011	T049	018							
SIO1	SS	0005	T050	005							
BASC1	SS	0009	T050	010							
BOOT	SS	0021	T050	019							
FTNH	SS	0001	T051	016							
EOF	UM	0001	T051	017	12000	12013	01002	01002	12000	12013	
FSPCE	UM	0001	T051	018	12000	12013	01002	01002	12000	12013	
RWIND	UM	0001	T051	019	12000	12013	01002	01002	12000	12013	
D.00S	SS	0067	T051	020							
TSRTS	SS	0006	T054	015							
TSRTR	RB	0005	T054	021							
CLEAR	BD	0001	T055	002							

[SLIDE 61]

DISC DUMP FOR DOSM GENERATION EXAMPLE

Φ,Φ	LB	DO	ΦΦΦ	SY	ST	EM		
001	046100	042117	021450	051531	051524	042515	044456	031461
	020001	036173	000723	041456	031461	020000	036173	000724
	044456	031062	020000	036164	000000	041455	031062	020000
	036164	000000	046117	040504	051000	036155	000000	171373 C.S.
	041120	051000	036146	000000	040523	046502	020000	036137
	000000	037501	051503	047000	036137	000000	037501	051515
	041000	036137	000000	037502	047103	047000	036137	000000
	037502	050113	052400	036137	000000	037503	044117	050000
TK 322 SEC Φ	013000	000000	000312	044120	044400	036137	000000	037504
	041517	042000	036137	000000	037505	047104	051400	036137
	000000	037505	051120	051000	036137	000000	037515	051531
	051400	036137	000000	037507	042524	041400	036137	000000
	037515	047526	042400	036137	000000	037515	051531	046400
	036137	003000	037522	046125	047100	036137	000000	037501
	043114	043400	036137	000000	037514	051524	046000	036137
	000000	037514	052516	044400	036137	000000	037522	052502 UB
Φ,1								
002	000000	067731	017570	067732	077762	017613	017613	017613
	017613	064120	007004	077310	064117	044055	160001	044051
	010072	073773	053774	077763	053775	077764	077304	044056
	160001	001727	013752	073305	050060	027460	053765	027445
	067304	044066	037310	027415	027505	044052	160001	023773
	033774	170001	063773	073302	002004	073303	063774	073773
	067304	160001	073730	164000	017570	063305	050060	027440
	006004	160001	033773	170001	006004	063747	170001	006004
	003004	170001	067304	077311	027440	060154	001722	013767
	033774	001727	001723	070154	063763	067302	017606	063764
	067303	017605	002400	067774	017606	063311	067775	017606
	067763	006003	027540	044055	160001	023774	033302	170001
	067764	006003	027546	023775	033303	170001	063776	001200
	067772	006003	002004	064155	070155	054175	070175	006400
	050175	064115	074200	047757	074157	064175	074161	124003
	000000	057730	127570	037730	163730	002021	027571	013766
Φ,2								
003	002002	027571	163730	043773	173730	027571	000000	044045
	044201	170001	127606	000000	167762	037762	163762	037762
	003304	040001	005225	106702	106602	167762	037762	002021
	127613	043754	073761	002020	002400	043753	102702	102602
	060001	001767	013752	102600	103700	063755	033776	106701
	102601	103701	102300	027652	060001	013752	043751	002021
	033757	002020	043747	033772	102600	103700	102301	027666
	017714	063756	033776	102601	103700	106701	103706	103701
	102301	027700	017714	060001	013752	053750	002001	002405
	063760	044003	063761	027627	000000	103700	063776	106701
	102601	103701	102300	027722	102500	000010	102031	127714
	015642	015724	037733	000002	000732	011413	001634	007361
	000012	007361	007567	000411	007533	007567	006406	000014
	000027	177764	000377	177600	000200	030000	020000	000400
	000351	000030	000000	000000	000000	000031	070036	177740
	015771	045511	000000	000015	000015	000016	000000	177733
Φ,3								
004	022105	054060	031461	000413	000002	007567	007624	000732
	000733	007567	007624	022105	054060	032001	000415	000004
	007567	011176	000732	000741	007567	011176	022105	054060
	032401	000421	000002	007567	007745	000732	000733	007567
	007745	022105	054060	033401	000423	000002	007567	007746
	000732	000733	007567	007746	022105	054060	034001	000425
	000002	007567	007732	000732	000733	007567	007732	022105
	054060	034401	000427	000003	007567	011142	000732	000763

SYSTEM LABEL/ USER BUFFER SECTOR

DISC RESIDENT BOOTSTRA

MEMORY BOUNDS & DISC ADDRESSES FOR C.R. SYSTEM

$EXØ3 $EXØ4 $EXØ5 $EXØ7 $EXØ8 $EXØ9

SYSTE DIRECTO

[SLIDE AD-1]

SYSTEM DIRECTORY

	010142	022105	054061	031401	001002	000002	007567	$EX10
	000733	007567	007745	022105	054061	031401	001004	$EX12
001004	000732	007567	007751	000732	000733	007567	007751	
022105	054061	031401	001006	000004	007567	010223	000732	$EX13
000754	007567	010223	022105	054061	032001	001012	000004	$EX14
007567	010241	000732	000751	007567	010241	022105	054061	
032001	001016	000003	007567	011153	000732	000763	007567	$EX15
010153	177777	000001	000000	000000	000000	000000	000000	

LAST DIRECTORY ENTRY THIS SECTOR

Ø,4
005 022105 054061 033001 001021 000002 007567 007722 000732 $EX16
000733 007567 007722 022105 054061 034401 001023 000003 $EX19
007567 010107 000732 001000 007567 010107 022105 054062 $EX2Ø
030001 001026 000003 007567 010167 000732 000761 007507
010167 042126 051063 030004 001401 010003 010241 010555 DVRØ1
001000 001002 010241 010555 042126 051060 031004 001004 DVRØ2
000003 010241 010443 041004 001002 010241 010443 042126 DVR22
051062 031004 001027 000005 010241 011025 001000 001002
010241 011375 045117 040504 051003 001414 000040 012000 LOADR
021032 001002 001425 012000 021032 045117 041120 051003 JOBPR
002024 000240 012000 022463 001002 001414 012000 022463
040523 046502 020003 003411 000027 012000 017120 001002 ASMB
001362 016522 017120 040523 046502 042005 003411 000004 ASMBD
017127 017547 001362 001363 017442 017647 040523 046502 ASM81
030405 003415 000026 017356 020542 001362 001424 017356
020542 177777 000000 000000 000000 000000 000000 000000

LAST DIRECTORY ENTRY THIS SECTOR

Ø,5
0 040523 046502 031005 003423 000007 017345 020550 001362 ASMB2
001410 017351 020550 040523 046502 031405 004002 000003 ASMB3
017473 017771 001362 001363 017630 017771 040523 046502
032005 004005 000004 017366 020027 001362 001371 017365 ASMB4
020027 040523 046532 032405 004011 000006 017345 020425 ASMB5
001362 001404 017351 020425 043124 047040 020003 000017 FTN
000005 012000 013127 001002 001007 012000 013127 043124
047060 030405 000025 000027 013254 022120 001047 001502 FTNØ1
016552 022120 043124 047060 031005 005004 000031 013254 FTNØ2
021027 001047 001356 013741 021027 043124 047050 031405
000405 001033 013254 022630 001047 001277 015117 020500 FTNØ3
043124 047060 032005 000405 000041 013254 020750 001047 FTNØ4
001360 013702 020756 046111 041122 054400 000407 000023 LIBRY
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000

LAST DIRECTORY ENTRY

Ø,6
007 000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000

not used

[SLIDE AD-2]

φ,7

008	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

φ,8

009	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

NOT USED
"B" VERSION
GENERATOR
WILL NOT
WASTE

φ,9

010	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

φ,10

011	000000	103100	017735	063634	070242	163634	170627	102504	DISCM
	033635	073646	000000	106504	054057	102004	054060	124630	
	054103	027754	044045	074000	002020	027714	003000	040202	
	002020	027714	044201	160001	002003	027714	114552	102504	
	164204	114001	027776	050260	002003	027714	060261	002002	CRS.
	027714	160557	164201	006003	002002	027714	074260	124544	
	063634	017746	002001	027722	017726	127634	017726	102100	
	102705	127634	000000	060241	103101	000036	102101	060237	
	064240	127726	000000	070237	074240	001520	102201	002004	

[SLIDE AD-3]

CRS$_2$

DISCM

	070241	127736	000000	003000	040254	002020	037746	127746
	034105	227761	063775	070105	034104	060106	002020	027714
	034107	027714	063775	070107	034106	060106	050110	124631
	027714	176650	170217	070531	174220	017331	060513	003004
	050203	070513	060262	000010	026070	060141	002003	026070
	160212	002020	003004	050254	002001	026070	007400	044111
	044121	003400	140001	114632	040117	050203	002001	026070
Ø,11								
012	160214	160000	010075	153633	002001	026070	160211	114634
	026070	060123	072052	160214	066207	017361	002052	160220
	164215	006021	025061	002004	001100	070530	160206	012211
	170206	160211	170242	026651	160217	002002	026152	160206
	012211	170206	160213	012210	002002	026163	160211	070505
	164220	114634	026150	164635	154636	026204	060260	002003
	026143	003400	140121	114632	040117	040056	160000	002020
	026143	052212	002002	026141	160206	070237	164220	074240
	160211	070242	006400	074260	027426	006400	074260	164220
	160206	102100	102705	124505	160206	124505	050056	002001
	026641	160206	110637	150640	026644	060056	026641	053213
	002003	026171	060513	002021	027256	160205	010072	053204
	027244	053223	002003	124554	017003	027271	124554	027271
	060203	072213	027244	177734	020000	037777	000000	000000
	000000	103100	114641	062214	040052	002001	102505	070512
	070242	072424	036424	160000	012473	052475	002001	026633
Ø,12								
013	160512	012474	160000	052476	002001	026315	066424	160001
	114634	026636	160001	070225	007004	040001	002003	026635
	040052	070224	002020	026535	040042	002021	026635	064043
	002400	017361	000225	061204	007004	036424	062266	070505
	162424	016507	170505	034505	036424	006006	026274	160226
	052500	002001	026325	114642	170001	006004	102100	102705
	124225	052477	026320	026633	114642	114577	000000	134532
	124643	002003	026635	052501	124534	052502	026346	052503
	026340	052504	026425	026362	060237	072342	000000	026312
	034105	026312	060103	002003	026344	032505	072357	060103
	032506	072350	060056	000000	000000	026312	002021	003004
	042461	002020	026635	160226	002020	003004	070514	050064
	026400	050065	002001	026402	060063	070514	050060	026312
	042436	160000	050052	026635	052437	026431	070474	062462
	070475	002004	064514	154475	027020	002004	034475	026417
	000000	160644	070474	060066	027020	006400	160645	070474
Ø,13								
014	062470	027020	002436	002431	002431	002431	100612	177777
	100605	100606	100607	100607	100607	100611	177777	002431
	002431	002431	100603	100604	100610	000022	002463	000020
	000021	000006	000007	000010	000022	000013	000004	176000
	001777	114000	002214	004243	177755	177754	177753	177752
	000027	102500	103700	000000	002003	026635	050054	026635
	001275	002001	126507	160000	025510	140646	070514	060232
	070472	066552	074473	160647	070474	062471	027020	006003
	026665	060514	050055	002001	025432	060001	040055	160000
	010071	050065	026432	052471	026432	026631	002533	000000
	010072	002003	026653	007400	044121	003004	040122	002020
	026653	044121	160001	002003	026665	040052	114632	040117
	017345	126553	000000	164226	174212	006020	007004	060225
	170211	160227	050050	003004	170213	060230	054056	160230
	170214	160231	170215	060232	154650	026625	154651	026625
	160232	170216	160233	170217	126576	006400	026670	064054
Ø,14								
015	026670	054055	026670	064056	026670	064057	070473	026670

CRS$_2$

[SLIDE AD-4]

CRS2

DISCM

	007400	170205	016776	164652	026670	164653	026670	064060
	026670	064061	026670	064062	026670	064063	026670	064064
	026670	064065	060512	000001	060242	070472	002400	070245
	160654	070474	160653	027020	000000	160205	002021	026723
	060061	070257	017003	026721	026717	160206	110637	150640
	026717	034257	026721	050060	026774	060203	170001	150213
	010077	070001	160212	002020	003004	150650	026740	150651
	026742	030001	170213	026744	002404	026735	060055	026735
	160205	010072	164203	114001	070510	002102	026766	160205
	001222	000010	026762	060513	003004	070513	160206	130655
	170206	126703	050056	002300	017331	006440	016776	060510
	036700	126701	000000	060526	002003	126776	124000	000000
	064201	002400	150001	027017	060165	002002	027016	006004
	150001	037003	037003	127003	073116	077115	060243	070477
	060244	002003	124474	003004	070500	060245	002020	027222
Ø,15								
016	053116	027110	160477	010074	053116	027047	034477	034477
	034500	027036	124474	003004	070245	064246	074167	007004
	044247	074170	160477	001722	010071	034477	164477	074166
	036212	002003	027106	001722	001222	070001	003004	040170
	002021	074170	017117	060174	070166	064250	074167	007004
	044251	074170	017117	072212	067115	063116	003004	070245
	124246	000000	000000	000000	060166	017130	063127	006404
	017400	002400	127117	000166	000000	006400	114577	177770
	060154	010074	003004	044000	006021	026663	127130	000000
	070513	064252	074167	007004	140650	144000	074170	040055
	160000	070166	017117	060174	070166	064253	007004	060513
	140651	144000	006003	127143	060253	070167	036212	074170
	017117	072212	127143	073204	077206	060206	073205	027244
	000000	000000	000000	000000	060203	073213	027244	000000
	073221	060203	073220	027244	000000	000000	037236	063115
	073240	053116	073237	060474	073241	060472	073242	060473
Ø,16								
017	073243	027244	000000	000000	000000	000000	000000	000000
	063204	002003	027256	163205	001222	010056	002002	027256
	073204	127206	063213	002003	027271	060513	002020	027271
	063213	017345	002400	073213	127207	063220	002003	027304
	017003	027277	027304	063220	017345	002400	073220	127221
	063236	002003	027325	060245	002020	027325	002400	073236
	063242	070472	063243	070473	063241	070474	063237	067240
	027020	102100	003400	007400	027325	000000	064201	060203
	150001	027342	006004	150001	002001	127331	002400	170001
	127331	000003	067360	074474	067357	170001	002004	006004
	034474	027351	127345	000203	177757	000000	070503	074505
	167361	074504	037351	006400	002002	164503	174504	034503
	034504	034505	027367	127361	000000	077412	064203	077423
	073424	060055	073413	063424	073414	114545	000000	000000
	003414	000000	003420	027244	063423	017345	127400	000000
	000000	000000	077561	017563	034261	067561	063556	006002
Ø,17								
018	063553	073441	114545	000002	000001	003441	000002	003463
	063561	002002	027244	060520	114634	027244	060517	103101
	000036	102101	060515	064516	102100	102705	124520	017563
	060123	073471	114545	000001	000401	003471	000044	003475
	027441	077562	017563	002400	070261	067562	017575	060471
	150056	027430	150657	027430	027444	060111	050054	027544
	073532	064123	077533	016553	160205	001222	000010	027530
	060203	053513	002001	017143	114545	000001	003532	003533
	000044	003537	027244	160123	010075	150633	017575	027511
	060261	002003	027551	034260	027244	007400	027426	003554

CRS2

[SLIDE AD-5]

DISCM

```
006412  040137  003557  006412  025137  000000  000000  000000
060237  070515  063240  070516  060241  070517  060242  070520
127563  000000  074530  074472  062575  074473  064123  160001
010075  150633  027631  034261  114545  000002  000001  004140
177770  003625  064471  154656  027244  060520  114634  124554
027452  002430  070261  017563  127575  160001  006004  164001

114577  177770  170635  064471  154660  027677  150661  124662
150636  124663  150664  124665  150666  124662  154656  002001
027702  150656  005401  027714  074471  062213  002002  027670
160667  070474  062463  027720  076213  150656  027447  017345
160211  070535  026141  150660  027743  027607  154657  027736
006002  027607  064262  004010  027714  064141  006003  027743
150670  027731  150671  027725  150672  027731  150673  027731
027607  160674  070474  060064  027720  150675  070474  160650
064530  027020  150657  027743  150676  027743  027607  002400
114556  002404  070262  160677  070141  160700  070142  160701
070143  060263  070166  060254  070167  060264  070170  017117
060174  070166  060255  070167  060265  070170  002400  070471
017117  002400  070261  070260  064530  070530  102100  102705
124254  060112  070111  006400  074475  060106  032150  070106
016044  070252  070245  062276  070474  054475  062313  124540
064242  074470  062126  050471  124702  070471  170703  124544

020 000000  060141  002003  124704  062262  032150  070262  124705
000060  060120  003004  070474  074475  064117  044055  160001
010072  032144  072076  006004  160001  001265  012304  052305
002041  026073  062072  044056  170001  124706  004045  052153
032152  170001  107700  144707  034474  026053  102106  107706
102107  107707  064201  002400  170001  070513  170710  170711
170712  170713  070261  070260  170703  006004  170001  126044
040502  050101  043517  045117  042512  046125  042521  052520
042116  042101  052131  047506  044507  047117  051105  042040
107700  045117  041120  051040  100000  035000  000040  000400
062271  070474  060065  124540  105700  062273  070474  062310
124540  000000  052165  002004  160000  114565  160206  001222
010056  002003  026204  007400  046165  160205  010072  124553
160206  012304  052305  002001  026217  114714  026217  003401
003400  042165  124566  162165  002021  003004  170212  036165
162165  170213  036165  162165  170214  036165  162165  170215

021 036165  162165  170211  036165  114572  126165  026240  000000
072260  162243  072261  062260  036243  000066  005600  002040
002004  035261  525251  126243  000000  000000  000000  070001
001700  040001  126252  000470  100613  100614  100615  100616
100617  100620  100621  100622  100623  100624  100625  100626
037400  014430  177764  177763  000014  000015  000016  000017
060224  040047  002002  026373  060231  114715  160226  002020
026353  060163  006400  114577  177770  002002  006004  007004
144230  000021  026341  006400  026365  006400  060154  010074
003000  140227  140230  002003  026353  002021  026365  164230
144227  007004  060154  010074  002004  040001  002020  026337
164230  174231  002400  070245  102100  102705  124225  002400
070245  124541  060224  040047  002002  026443  060227  114715
060230  114715  060231  114715  060160  006400  114577  177770
002002  005004  174227  060102  002003  026431  006400  114577
177770  002002  044052  074000  026433  060154  010074  170230

022 060116  170231  002400  070245  102100  102705  124225  002400
070245  124541  060050  040224  002002  026502  060230  114715
```

CRS_2

$EX01

$EX02

$EX06

[SLIDE AD-6]

CRS2

```
        064227  034162  050200  002004  114716  026477  064227  060114
        114716  026477  002400  170230  002400  070245  044052  160001
        102100  102705  124225  044057  160001  026467  002400  070245
        124541  050472  072664  060473  072665  060245  052566  026537
        060262  002002  026501  060525  166663  056670  002003  026601
        066647  076641  062654  114717  026614  066641  044063  076641
        056663  026505  026526  062664  066577  114717  026556  062664
        066573  114717  026562  034162  050200  002004  066664  114716
        026567  006401  066577  002400  070245  126665  062573  064046
        114543  000270  066647  076664  062577  064046  114543  000133
        062664  064046  114543  000126  026556  002400  066667  114543
        000270  034162  050200  002004  066664  114716  026630  026555
        066641  044060  160001  070161  006004  160001  070157  006004
        160001  070230  066641  026567  076664  062663  040043  052647
Ø,22
023     026644  072641  040043  064043  114543  000000  062641  026632
        062664  064046  114543  000270  060161  070275  060157  070276
        060200  070277  062670  172663  034525  066647  026567  000460
        000000  000000  177755  177607  022124  025707  062771  002020
        026772  062704  006002  062705  070473  017250  064057  027351
        047122  050105  004672  060161  073261  060200  073262  064224
        044047  006024  026724  160231  050054  027074  002024  027120
        062706  070526  060156  003000  006021  160230  072771  002020
        027013  070161  017215  017126  053030  114721  026747  064227
        017200  027025  027101  160227  002003  027043  010075  053000
        027101  027043  062771  003000  070161  017215  063030  114721
        027001  064227  017200  026772  027006  000000  036771  026756
        063162  056777  027045  177756  025000  160227  010075  053000
        027006  026772  060262  002002  017230  017126  027101  160227
        002002  025756  072771  017230  017215  017126  063030  114721
        027043  053031  064041  027045  005034  005032  046102  046075
Ø,23
024     000000  000000  000000  005040  052516  046102  046040  063037
        064046  114722  017250  064055  047154  002400  073154  070526
        070245  160600  070474  060057  124540  006400  074245  074526
        063154  002002  027050  063263  102100  102705  124225  005046
        160230  070161  063073  070525  017215  006400  077263  074126
        074133  067177  017200  027115  006404  060370  070200  063227
        114720  064373  074157  027061  070245  160601  070474  063125
        124540  000023  000000  060176  050272  027136  070472  060272
        070473  037154  060271  053155  127126  067161  053156  067157
        057161  063160  073164  077165  063163  064043  114722  127126
        000000  042117  052123  041040  037477  037440  005166  005164
        000000  000000  042111  051503  020116  047524  020117  047040
        051531  051524  042515  005174  000000  160001  050273  006005
        127200  160001  050274  006005  127200  160001  050275  037200
        127200  000020  006400  060161  050155  064115  074200  063227
        006404  114720  127215  000200  000000  060161  033241  001727
1,Ø
025     073247  063242  064042  114722  127230  000060  005243  051525
        041103  044101  047075  000000  000000  063261  070161  063262
        070200  060074  070163  073263  122250  000000  000000  000000
        006002  027700  150227  064161  077314  050050  003004  114565
        060514  053313  002001  027335  017766  160206  170230  064224
        054056  027310  164220  174231  102100  102705  124225  000015
        000000  000000  040231  040045  073766  164000  006003  027331
        044055  160001  010072  053334  002001  037315  060203  127315
        000000  160205  010072  073334  017315  027356  063334  002004
        017315  027356  060061  017315  027356  060062  017315  027356
        102000  027354  006004  160001  064514  054056  027411  001222
```

$EX10

$EX11

$EX17

$EX18

CRS2

$EX18

	010056	002003	027411	050054	027402	050055	027376	063375
	124566	005266	017766	063334	064512	124553	060512	070470
	063410	070471	017766	124544	043517	063334	017315	173766
	063334	002004	017315	173766	160205	001222	000010	027427
	064513	006020	114567	060514	050054	027435	050055	027435
1,4								
026	027443	160227	050050	003004	010072	050056	034162	114571
	160226	002021	027462	160206	013456	053457	027565	027667
	017766	124574	037400	014400	000016	000017	060514	050056
	027667	064230	007000	044254	006021	124561	017756	044230
	002040	124562	007004	044100	006020	124562	160206	013456
	053457	002001	027667	060514	053460	027454	053461	027454
	060102	002003	027523	001727	010074	040052	027525	060154
	010076	070535	060160	006430	114577	177770	002002	006004
	160232	010074	170232	003060	040001	002021	124563	063755
	006400	114577	177767	002002	006004	144233	002400	044511
	006020	027563	002004	027553	140232	003004	040505	002020
	124563	160233	044511	002021	124563	160232	001727	130233
	070171	060230	070172	160231	070173	060224	040045	002002
	124560	050055	114565	114571	064177	160227	050050	074161
	063753	170214	114572	027654	067314	074161	050060	027642
	050055	027650	050056	002001	027645	060203	070472	150600
1,2								
027	070474	017765	067641	060057	124540	000022	063644	124566
	005616	050057	002001	124573	017766	160206	006400	027310
	017766	063314	070161	160227	050050	003004	013666	002003
	124551	027310	020000	160205	001222	000010	027616	060203
	050513	027616	114570	027615	160233	003004	067754	074504
	044057	140001	070505	017756	060161	073314	006400	063755
	114577	177747	006002	002004	003004	040505	002020	124564
	064504	044056	160001	006400	114577	177770	077756	001727
	140233	070505	040511	002020	027744	037756	070505	027736
	067756	005727	060505	030001	070171	034162	027575	000171
	000126	000000	000000	164231	006121	027764	005100	007004
	077755	127756	004000	060245	002020	003004	070245	127766
	000000	003000	070001	040254	002021	026006	044100	006420
	026006	124723	002400	070245	124541	000000	072125	076126
	060161	072107	060162	072123	002400	076162	072130	060115
	040075	064055	016252	016136	062123	002003	026114	064527
1,3								
028	056057	026111	006020	026060	004065	006043	026111	076131
	002041	026054	362130	052107	026054	016202	026054	016136
	036130	065131	026040	047117	060156	003000	072130	062130
	003000	052107	026072	016202	026072	016136	036130	026063
	026111	062124	066110	114543	006123	060161	052107	026115
	002400	070126	026115	000000	177765	052107	016202	000000
	036011	052115	040075	006404	016252	066100	126011	000000
	000400	000000	000000	000000	000000	000000	000000	000000
	000000	000003	000000	062125	016244	062264	072124	062126
	066124	016266	026075	066124	044055	160001	010073	040045
	002020	044061	044056	076124	007000	144546	006020	026172
	162124	002003	126136	050052	002101	026143	036125	062125
	010073	003004	040116	002002	026137	126136	000000	070161
	060074	070163	062237	070526	002400	070200	016244	062241
	050273	060274	052242	060275	052243	002001	026226	060370
	070200	016244	060370	070157	060200	002004	072125	036202
1,4								
029	002400	070526	126202	006240	026234	051531	051524	042515
	000000	066123	074162	006404	016252	126244	000000	070166
	062264	070167	062263	070170	062265	114536	126252	000200

$ADDR

$SRCH

CRS2

$SRCH

$LBL

DVR05

DVR31

1,5

1,6

1,7

	000270	000166	000000	072127	160001	152127	002001	026313
	006004	036127	160001	152127	002001	026313	006004	036127
	162127	010075	172127	160001	010075	052127	002001	036266
	126265	000000	062327	072334	003400	066370	120001	006004
	036334	026322	126315	177741	000000	076334	064050	114543
	000000	126330	000000	054000	062352	016330	060270	052353
	002001	126336	016315	050327	036336	126336	000273	046102
	000000	070170	076334	064200	074166	062370	070167	062371
	066334	034162	114536	126354	000270	000166	000000	072401
	076402	034261	114545	000002	000001	000000	000000	006405
	124551	002400	070261	126372	000043	160213	010056	050456
	026431	032633	002311	001425	102600	016446	000015	016464
	102700	002400	170220	126410	006424	120213	052577	026436
030	060057	126410	164214	006021	007004	174217	005300	174214
	060055	026415	000000	164214	005200	174216	164215	007324
	005010	007004	001310	026462	006003	007400	174217	126446
	000000	062464	052430	026476	163206	002020	026476	034260
	036464	126464	164213	005332	026611	102500	010073	050073
	026541	052422	026537	050065	026562	006020	026537	050054
	026544	164216	004065	134216	002041	001727	072446	060074
	002340	001727	110001	032446	170001	160213	001421	134217
	000040	001425	170213	036464	026606	016446	160213	026535
	016553	044052	174216	160217	040052	170217	026537	000000
	160214	001000	150216	026537	164216	126553	005421	174213
	016553	004065	160001	010475	032534	002040	170001	160217
	164215	002004	006021	001100	006020	007300	044000	106700
	001521	102600	103700	002400	126464	062464	052430	026621
	102500	010074	050074	026621	070260	002400	150216	026574
	150217	026645	150216	134216	000065	044010	062630	160000
031	006051	001727	010074	134217	026643	052657	026574	102600
	026537	062422	150220	060065	170220	052422	026643	006400
	174216	026643	004000	002132	007332	064162	060155	006402
	060161	073360	000065	073347	002441	063344	073352	060164
	006002	060163	073353	063347	017211	000400	010067	002102
	026737	160213	010056	001510	025714	067145	063336	026727
	063337	002341	026726	160213	010077	001225	053340	026726
	060057	126660	067332	077062	073346	060041	170220	062736
	072743	026764	006737	060056	006400	074162	126660	000000
	063347	106501	006003	026755	002400	004033	026755	002004
	026751	017211	027115	067333	106606	164217	124001	006764
	164214	006004	160001	002003	027071	073354	006004	160001
	002021	003005	001100	002003	003400	073355	044051	160001
	010075	073350	120001	073351	003004	040116	001727	001300
	043355	002020	027034	001727	001200	010074	003004	040116
	050116	062701	043350	070174	002400	006004	170001	027051
032	044055	003004	170001	043355	073355	044052	003004	043354
	170001	044052	063350	042701	174001	017126	017241	067354
	063346	053336	047343	106702	106602	067355	000000	017162
	017241	063346	053337	017172	026764	060200	002302	000040
	060162	067353	002041	002003	074164	002041	002002	074163
	002400	070162	164215	006020	007004	126743	000000	063112
	170217	002400	035743	126743	000000	033347	106701	102601
	103701	127124	000000	063351	043335	002021	032701	002020
	043334	033352	070001	063350	001767	017143	127126	000000
	102600	103700	053353	002300	073353	063341	001225	017120
	102300	027154	106600	103700	017112	127143	000000	102702
	106602	103746	017120	106706	017112	127162	000000	017126

CRS_2 & CRS_3

DVR31

Addr								
	063355	003004	040073	001727	001200	010071	102600	103700
	063342	017120	017112	017241	127172	000000	102106	103700
	106701	102601	103701	102300	027217	102500	073357	010074
	073356	063357	001226	002440	027237	160206	010075	033356

CRS_2

Addr								
033	170206	037211	127211	070525	127211	000000	063357	002111
	127241	001422	001727	000010	027262	001723	000312	027365
	057340	127241	134220	027276	001200	002021	002405	060056
	006400	074152	067360	074125	067351	074130	067350	074127
	005727	126743	063252	006500	017143	002400	006500	017143
	027051	001332	027315	067345	057336	027254	027313	102031
	027104	017126	067345	106702	106602	063336	007400	103700
	017162	053350	010075	073350	034525	027051	102100	120000
	000000	000000	020000	010000	110400	030000	060000	100000
	001000	107350	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

NOT USED

1,9

Addr								
034	006410	006464	040011	002400	000000	000000	000000	000000
	000000	000000	000030	000000	000000	000000	000000	000000
	000000	010241	010313	000013	000440	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	010555
	001002	001401	006660	006743	140014	014400	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	012241	010320	000316	001000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	010443	001002	001404	010241	011001	100022	011000
	000000	000000	000002	000000	000000	000000	000000	000000
	000000	000000	011075	001002	001407	000001	000003	000003
	000004	000002	000001	000005	000000	000000	000000	007361
	000004	007402	000000	007423	007444	000000	000000	000000
	000000	007465	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

EQUIPMENT TABLE CRS_3

DEVICE REF. & INT. TABLES CRS_3

NOT USED

1,10

Addr								
035	030400	060135	050221	001260	000130	160001	002003	002004
	072000	001657	042000	001277	133330	072000	001647	062000
	001300	066000	001276	016000	000363	052002	133320	072000
	001432	162000	001250	072000	001443	026000	000000	177760
	002000	000400	001301	020102	042507	044516	000000	020047
	042105	041125	043447	020117	000000	050105	051101	052111
	047516	030030	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

NOT USED

START DRS

11

Addr								
036	060105	002023	027604	033622	070106	060113	002011	027604
	002400	070245	070530	006404	124550	067623	002400	070245
	070471	114543	000141	114556	006400	060262	074262	002321
	124537	007400	124544	100000	177765	000000	000000	000000

$EX03

DRS

	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	003000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000400	000000

1,12 ← B.P. LINKAGE FOR $EXØ3

037	007567	000000	000020	000000	000000	000000	000000	000000
	000000	000000	000000	000400	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
038	054062	027575	057574	027633	027711	000022	060225	070470
	060141	073626	060142	073627	060143	010075	033620	073630
	063625	064041	114735	063624	070471	002400	070245	007400
	124544	000240	040000	037777	000031	043517	007626	020040
	020040	020043	051525	051520	060472	114552	160206	001727
	010073	053623	027704	160206	013622	033621	170206	060512
	070470	002444	064117	054203	027657	002004	044066	027552
	114733	073702	063671	067746	114735	060472	114552	027612
	000012	030050	007672	044457	047440	042522	051040	047122
	020105	050524	021440	020040	020040	063676	073773	063766
	027662	000000	054057	027755	077710	004010	114736	063710
	001110	027726	006400	074245	102100	102705	124225	060262
	002002	027721	114737	160740	114734	073754	160001	073752
	006004	160001	073753	063747	064041	114735	027575	177754
	007750	052504	020040	020040	020040	020040	060473	073773
	063766	067745	114735	060262	002002	027721	027726	007767

$EXØ4

1,14

039	044457	047440	042522	051040	020040	020125	051505	051040
	042111	051503	000000	060472	114733	072047	006004	160001
	072046	060473	114733	072034	006004	160001	072033	062023
	066022	016066	126001	177716	010024	042111	051503	020107
	042516	020103	047504	042440	000000	000000	020116	047524
	020123	054523	020107	042516	020103	047504	042440	000000
	000000	020105	051122	020120	047523	051440	000000	060224
	002004	003000	040225	070225	072065	126055	000000	000000
	072075	076075	034261	114545	000002	000001	000000	000000
	010101	124551	002400	070261	126066	000000	064041	016114
	126104	000000	064043	016114	126110	000000	076175	066153
	076147	076150	076151	066146	076152	016154	146152	176152
	002003	026140	016154	005727	146152	176152	036152	002002

ASCII ↓

DRS

ASCII ↓

	026124	066151	062147	072151	076147	066146	126114	010147
	000000	000000	000000	000000	030060	000000	006400	076174
	070001	042175	036174	002021	026157	006020	026160	060175
	007004	014007	062174	040052	126154	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
1,16								
041	007567	010104	010110	010065	010001	012055	010055	000400
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
1,17								
042	060051	040224	002020	027725	003004	073741	064227	034102
	060200	002004	114716	027656	064227	060114	114716	027654
	064227	160001	073730	006004	160001	073731	006004	160001
	010075	073732	034261	114545	000002	000001	007730	177702
	007631	124551	002400	070261	070245	060106	002020	027646
	033737	070106	060113	002021	027646	006400	124550	063740
	064111	054054	002400	070471	124537	002400	027657	003400
	070524	160226	050063	027670	060001	067744	114543	000141
	027675	044056	060001	064043	114543	000144	002400	070126
	070127	070130	070131	070132	063741	002003	027720	067742
	063743	073730	160001	160000	173730	037730	006004	037741
	027711	060245	002020	003004	070245	124534	002400	070245
	124541	000004	000000	000000	045511	051523	044516	043440
	100000	045117	000000	000230	000126	177765	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
043	007567	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

BASE PAGE LINKS FOR $EX04

$EX05

← B.P. LINK FOR $EX05

NOTE: $EX05 uses $SRC but not needed here because CORE RES.

[SLIDE AD-12]

DRS

1,19

044 060051 040224 002002 027647 060227 070475 114715 006400

063744 040105 002002 027607 170475 034475 170475 027622

017552 000065 070474 074000 001723 040001 040001 170475

034475 050474 170475 034475 006400 060104 002002 027633

170475 034475 170475 027640 017652 007745 174475 034475

170475 034475 002400 170475 070245 102100 102705 124225

002400 070245 124541 000000 077735 167652 164001 037652

006003 027733 006120 007204 077736 007004 077737 067740

077741 064051 077742 077743 057735 006021 027703 037743

007200 002002 003005 006004 002040 037742 047737 006021

027733 047736 000066 005600 047737 006021 002005 047736

037741 027711 003006 002020 002001 027733 103101 037742

003004 037743 127652 007005 102101 127652 000000 000000

000000 177750 000000 000000 000000 001130 000074 000000

000000 000000 000000 000000 000000 000000 000000 000100

000000 000103 000000 000000 000000 000000 000000 000000

$EXØ7

1,20 B.P. LINK FOR $EXØ7

007567 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

NOTE: $EXØ7 uses $ADDR but since $ADDR is Core Resident not needed here

1,21

046 060224 040046 002002 027714 060230 114715 060231 114715

060232 114715 060267 002002 027613 060160 006400 114577

177770 002002 006004 074267 160227 001665 040267 070001

060102 001727 010074 002003 053154 010076 007004 040001

002021 027701 034261 114545 000002 000001 007720 177754

007641 124551 002400 075261 160227 002021 027655 003400

170230 002400 070245 102100 102705 124225 060113 006400

074245 002020 027677 063717 064111 054054 002400 070471

114555 006400 060262 074262 002021 124537 007400 124544

006400 124550 060267 170234 164227 005665 040001 070267

060055 170231 060116 170232 027650 002400 070245 124541

045117 021440 052122 040503 045523 020125 047101 053101

044514 040502 046105 000000 000000 000000 000000 000000

000000 000000 000000 000000 000000 000000 000000 000000

$EXØ8

DRS

1,22 ← BASE PAGE LINK FOR $EX08

1,23

2,0

21

$EX09

ASCII

BASE PAGE LINKAGES FOR $EX09

DRS

```
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
```

2,2

```
051 134557 050254 003004 040146 002020 027660 060147 003004
    040100 002020 027660 060255 003004 040150 002020 027660
    060151 003004 040256 002020 027660 060146 070167 060144
    070166 060146 003004 040147 070170 002400 070262 064524
    006002 034162 114576 060174 070166 060150 003004 040151
    002003 027650 070170 060150 070167 064524 006002 034162
    114576 002430 070245 170557 070260 067724 102100 102705
    124152 034261 114545 000002 000001 007725 177746 007670
    124551 002400 070261 060106 002020 027677 033743 070106
    002400 057744 114543 000141 063742 070471 002400 070245
    006400 050252 074262 002021 027720 002400 070471 007400
    124544 060112 050111 027714 124537 000126 044514 046105
    043501 046040 050122 047507 051101 046440 051125 047440
    046111 046511 052123 045117 100000 177765 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
```

$EX10

2,3 ← BASE PAGE LINKAGE FOR $EX10

```
052 007567 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
```

2,4

```
053 134557 050116 003004 070511 060254 102605 060115 073725
    063751 073727 063754 073730 017671 060467 053747 027634
    002400 067757 114543 000270 063160 070370 060115 017676
    070371 074372 063747 070467 060115 073726 063751 073727
    063755 073730 063731 064051 114536 060200 002002 027641
    060370 027643 060370 070157 070160 057756 060103 002003
    077655 034261 114545 000002 000001 007732 177746 007660
    124551 002400 070245 170557 070260 070261 063750 070471
    007400 124544 000000 063731 000424 114536 127671 000000
    073726 001727 043753 073725 063726 043752 073726 063751
    073727 060052 073730 017671 037725 027703 002400 070270
    067760 060525 003004 044000 060525 127676 000000 000000
    000000 000000 007726 044516 050125 052040 035104 040524
    042454 054130 054130 054130 054130 054130 026110 026115
    051502 042101 000270 000400 177770 000200 000201 177752
```

$EX12

[SLIDE AD-15]

DRS

	177577	000313	000000	000000	000000	000000	000000	000000

2,5 ← BASE PAGE LINK FOR $EX12

054	007567	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000004	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	010000	000000	000000

2,6

055	060472	170735	060473	073723	060203	073722	074270	057732
	027626	002400	070126	054057	027736	114556	064270	047700
	160001	067717	114736	064270	054055	002001	027634	060236
	002003	027634	002400	070245	102100	102705	124225	160602
	070474	063733	006400	074245	124540	006400	060106	002020
	027645	033721	070106	060113	002020	124737	002400	067734
	114543	000141	063720	070471	002400	070245	064270	057731
	002001	027663	054530	124542	006400	060262	074262	002021
	027674	002400	070471	007400	124544	060112	050111	027670
	124537	007721	047522	046520	051121	052115	000000	046125
	046116	044502	041527	044524	043111	042104	000000	044505
	020040	045117	100000	000000	000000	000000	000000	037777
	040000	000031	000015	000014	000024	177765	177757	063723
	140740	160000	170741	160206	001727	010072	164742	053730
	027770	160206	013726	033727	170206	064117	007004	044203
	002404	006303	027765	002004	047735	027760	114733	170743

2,7

056	164744	060046	073724	160745	073725	160001	173725	006004
	037725	134746	124747	160750	050055	002001	026011	060202
	000010	026022	114545	000002	000001	010027	177756	010020
	124551	160751	114552	002400	070245	160206	164220	124752
	044457	047440	042522	051040	000000	020040	020040	020040
	020040	020040	020040	020040	010034	010045	020105	050524
	021440	000000	020040	010053	020125	051505	051040	042111
	051503	010063	047122	042524	050105	062076	066077	015100
	006400	074245	060270	150753	002001	074530	124550	040502
	051124	000000	072034	076035	062130	114734	072041	160001
	072037	006034	160001	072040	034261	114545	000002	000001
	010034	177762	010123	124551	002400	070261	160751	114552
	126100	000000	000000	064041	015141	126131	000000	064043
	016141	126135	000000	076222	066200	076174	076175	076176
	066173	076177	016201	146177	176177	002003	026165	016201
	005727	146177	176177	036177	002002	026151	066176	062174

2,8

057	072175	076174	066173	126141	010174	000000	000000	000000
	000000	030060	000000	006400	076221	070001	042222	036221
	002021	025204	006020	026205	066222	007004	044000	062221
	040052	126201	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

$EX13

ASCII

[SLIDE AD-16]

DRS

	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
2,9 058	007567	010131	010135	010139	010100	010064	010050	010033
	010052	010053	010044	010043	007724	007774	007723	007722
	007623	007731	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
2,10 059	064123	004066	077745	060473	073751	002400	073753	073754
	017725	017725	001727	073752	017725	033752	073752	017725
	053756	002001	027632	017725	010070	073753	017725	053755
	027632	053756	124734	010070	073754	063753	001723	043753
	043753	043754	073753	063752	053757	027766	053760	124735
	063753	002002	027650	002044	073747	064122	007404	077750
	027653	073747	003400	073750	063747	003004	040122	002020
	124734	053747	114733	073762	063747	040052	040121	160000
	114733	073765	034261	114545	000002	000001	007761	177766
	007716	060520	003000	040254	002021	124554	060517	103101
	000036	102101	060515	064516	102100	102705	124520	002400
	070261	114555	037747	037750	027653	124736	000000	067745
	004065	160021	002041	001727	010074	005600	006004	077745
	127725	000000	070001	001700	040001	127740	000000	000000
	000000	000000	000000	000000	000000	000000	000040	000054
	052520	042116	046125	000000	020105	050524	000000	063753
2,11 060	114737	040052	017740	040117	040056	073746	160000	001222
	010055	050054	002001	026007	160740	012104	170740	026136
	060120	003004	170741	002404	170742	064117	060001	044066
	174743	040056	160000	001222	010056	050054	026033	160743
	134742	134741	026015	026136	160742	114733	072056	034261
	114545	000002	000001	010052	177761	010046	124744	002400
	070261	114555	026026	042105	053111	041505	020043	054130
	020104	047527	047040	160745	016110	150121	026123	040052
	114746	040117	040056	170743	160000	012106	052107	026123
	160746	012104	032105	170740	026136	037777	040000	037400
	014400	000000	002003	026123	002020	026123	064120	007000
	144745	006021	026123	126110	034261	114545	000002	000001
	010141	177765	010133	124744	002400	070261	114555	002400
	070245	124747	044516	050125	052040	042522	051117	051040
	000000	064041	016157	126147	000000	064043	016157	126153
	000000	076240	066216	076212	076213	076214	066211	076215

BASE PAGE LINKAGES FOR $EX13

$EX14

ASCII ↓

[SLIDE AD-17]

DRS

2,12

061 016217 146215 176215 002003 026203 016217 005727 146215
176215 036215 002002 026157 066214 062212 072214 075212
066211 126157 010212 000000 000000 000000 000000 030060
000000 006400 076237 070001 042240 036237 002021 026222
006020 026223 066240 007004 044000 052237 000052 126217
000000 000000 000000 000000 000000 000000 000000 000000
000000 000001 100000 000000 000000 000000 000000 000000
000000 000000 000000 010000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000

2,13

062 007567 010147 010123 010062 010136 010110 107746 007750
007747 007746 007709 007753 007740 107751 010153 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000
000000 000000 000000 000000 000000 000000 000000 000000

BASE PAGE LINKAGES FOR $EX14

2,14

063 174734 060115 170735 160736 170737 160740 170741 160735
114575 160742 006004 114536 160734 002020 002400 164743
002002 164744 174745 006004 160746 002002 027622 164001
006003 027644 027624 050052 027625 040254 027627 060254
110747 170750 134745 160746 050052 027637 040254 027640
060153 170751 114752 017744 002001 134745 134745 160746
002021 027719 060255 110747 170750 060151 170751 114752
060047 170753 063776 170754 160750 114733 160001 010074
130755 170001 114756 160757 114733 114756 160757 001265
160000 114733 114756 134759 134753 027672 017772 002021
027657 134745 163745 002021 027723 060153 110747 170750
060147 170751 114752 017744 002400 070245 070113 070471
164760 114543 000141 002400 066530 006002 124542 060262
074262 002021 124537 007400 124544 000000 060043 170753
063776 170754 160750 114733 160001 010074 130755 170001
114756 160757 114733 114756 134750 134753 027760 017772

$EX15

2,15

064 002021 027745 127744 000003 114545 000002 000006 000333
177674 010002 124551 062047 003004 042052 124761 000000
062037 172051 136051 160001 172051 036051 006004 160001
172051 036051 006004 160001 172051 036051 126006 000000
114545 000003 001106 177777 000000 010036 124551 126026
020040 020000 000000 000000 000000 000000 010042 000000
000000 000000 000000 000004 177765 000200 000274 000405
000411 177774 000000 064041 016071 126061 000000 064545

ASCII

ASCII

DRS

ASCII ↓

016071	125065	000000	076152	066130	076124	076125	076126
066123	076127	010131	146127	176127	002003	026115	016131
005727	145127	176127	036127	002002	026101	066126	062124
072126	076124	066123	126071	010124	000000	000000	000000
000000	030060	000000	006100	076151	070001	042152	036151
002021	026134	006020	026135	066152	007004	044000	062151
040052	126131	000000	000000	000000	000000	000000	000000
000000	040003	000000	000004	000000	000000	000000	000000

2,16 — BASE PAGE LINKAGES FOR $EX15

065	007567	010065	010041	010042	010055	010043	010054	010044
	010045	010056	010057	010046	110046	010060	010047	010052
	010026	010050	010051	010040	010006	110047	010053	107772
	010061	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

2,17 — $EX16

06	000000	077721	063710	073716	060046	073717	060123	000066
	073714	060472	001200	003044	073715	017672	017672	017672
	017672	002400	173716	017672	002002	027623	037715	037717
	002001	027645	173716	027615	053711	027642	053713	027612
	010074	043712	073720	163716	070001	041723	040001	040001
	043720	173716	027612	037716	037717	027610	060470	073716
	006400	074470	074471	074245	007400	042003	124544	003000
	040254	002021	124537	063721	002002	114552	002400	067710
	102100	102705	127716	000000	063715	002003	127672	067714
	004065	160001	002041	001727	010074	037714	037715	127672
	127672	000126	000054	177724	000040	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

2,18 ← BASE PAGE LINK FOR $EX16

067	007567	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

DRS

2,19

068	060200	170733	064161	002400	170734	170735	054155	160733
	070200	160736	006404	114720	160737	114721	002001	027705
	164740	160227	010075	150741	006400	074270	160742	050273
	060274	150743	060275	150744	027742	160745	070271	060176
	070272	060227	164746	114724	060470	170747	114725	070327
	060200	140750	064270	006002	060370	070370	070157	064161
	054155	070160	160736	064270	006003	160751	074470	054055
	114720	060161	050175	006405	124752	060155	070161	050115
	070200	160736	114720	050227	164753	114724	160736	064055
	114720	060175	070161	160733	070200	124752	060271	170754
	164755	150745	164756	150757	164760	154755	160761	174762
	160763	164764	114722	160765	164766	114722	114545	000001
	000401	010106	177776	007734	124551	160767	150770	027607
	150771	124752	027722	006003	027624	160733	170772	070200
	001727	170773	170735	160736	006404	114720	060370	170774
	060511	002004	170775	134772	134734	160772	070200	160736

$EX19

2,20

069	006404	114720	160734	070200	160736	064055	114720	160735
	140773	072035	036036	124775	002400	070200	062046	006404
	114720	062030	003004	042033	070370	002400	070273	072030
	124777	062105	070470	062035	006400	074245	102100	102765
	124225	000000	000000	000000	000000	000000	000000	000000
	000273	003427	020040	025000	000400	177764	177760	000200
	000201	037477	037440	054505	047117	046102	052123	041040
	042117	051440	051531	051524	042515	010073	013066	000000
	000000	045101	041105	046040	000000	000000	000000	010077
	047513	020124	047440	050125	051107	042477	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000030	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

2,21

070	007567	010030	010032	010035	010046	010064	010054	010042
	010061	010062	010063	010057	010037	010105	010043	010047
	010020	010040	010066	010051	010060	010055	010056	010050
	010067	010065	010045	010076	010044	010106	010052	010053
	010031	010034	010033	010036	007762	007607	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

BASE PAGE LINKAGES FOR $EX19

2,22

071	060127	073775	001727	114733	170734	006004	160001	010074
	130735	170736	060126	070161	114733	130735	170737	060130
	114733	170740	160741	164742	017737	160743	164744	017737
	160745	170746	160747	067777	017737	102011	002400	070200
	006404	017755	160750	050273	060274	150751	060275	150752
	002001	027645	060370	070200	006404	017755	064161	060370
	070157	054155	074150	060154	010074	064372	054000	027733
	005727	077774	044075	074372	034371	064055	017755	064511
	077776	067767	074526	063775	006404	017755	063774	064055

$EX20

[SLIDE AD-20]

DRS

$EX2Ø

```
017755 063775 070166 063773 070167 002404 073170 067770
074526 063772 164753 034162 114536 037774 037775 037776
027670 160754 170746 160747 067777 017737 102077 006400
074245 074526 074525 124550 160755 164756 017737 027726
000000 073746 077747 034261 114545 000002 000001 000000
000000 007752 124551 002402 070261 127737 000000 070166
063771 073167 160757 070170 063772 034162 114536 127755
```

2,25

```
072 007766 007714 000270 000166 007774 000000 000000 000000
    177732 177759 177756 177764 000200 010403 036400 047040
    043106 051531 051524 042515 010014 050101 051111 052131
    020105 051122 047522 010023 051543 020040 026043 020124
    051113 020040 020040 026040 020123 041524 051075 000060
    010040 052125 051115 020117 020040 020104 044523 041440
    050122 047524 042503 052040 047526 042522 051111 042105
    020123 053511 052103 044040 010064 051520 040522 042440
    052122 045440 047525 042522 043114 047527 000000 064041
    016105 126075 000000 064043 016105 126101 000000 076166
    066144 076140 076141 076142 065137 076143 016145 146143
    176143 002003 026131 016145 005727 146143 176143 036143
    002002 026115 066142 062140 072142 076140 066137 126105
    010140 000000 000000 000000 000000 030060 000000 006400
    076165 070001 042166 036155 002021 026150 006020 026151
    066166 007004 044000 062155 040052 126145 000000 000000
```

ASCII

3,Ø

```
073 007567 010375 010031 010005 010030 010024 010036 010013
    010002 010022 010000 010006 010043 010037 010010 010011
    010012 010004 010007 010063 010001 010003 010101 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000300 000000 000002 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 006000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
    000000 000000 000000 000000 000000 000000 000000 000000
```

BASE PAGE LINKAGES FOR $EX2Ø

3,1

```
074 000000 016530 160213 010056 050054 026267 006404 050455
    026265 160213 001727 001222 010074 050062 002001 026264
    160206 032542 170206 006004 060001 126241 160214 001200
    170216 160215 002020 026277 001000 003004 002003 003400
    170217 062547 170220 160213 012541 032545 170213 102700
    002400 126241 000000 016530 160213 001200 072550 001727
    072551 000266 066550 102500 004010 002002 026340 134220
    026425 160206 012542 002002 026425 006400 026516 002040
    026355 010073 050073 026430 052553 026425 050065 026464
    006020 026425 050054 026451 026376 006011 026376 066551
    006011 026376 070001 001000 140217 002021 026375 060001
    170215 001000 003004 170217 060001 164216 004065 134216
    002041 001727 072552 060074 002040 001727 110001 032552
    170001 160213 012541 170213 134217 026425 062551 002020
    026503 160213 032546 170213 102700 036313 126313 160214
    001200 170216 160215 002020 026440 001000 003004 002003
```

DVR01

3,2

```
075 003400 170217 062547 170220 160213 032546 170213 026425
```

[SLIDE AD-21]

DRS

DVRØ1

	160214	001200	150216	026425	160216	040052	170216	160217
	040052	170217	026425	160213	012541	032545	170213	160214
	001200	150215	026425	164216	004065	160001	010075	032542
	002040	170001	160217	164215	006020	026514	005000	044000
	004031	006004	025516	007004	044000	160206	010075	006003
	032542	170206	002400	006003	064055	106700	126313	000000
	032554	072324	042543	072310	072425	032544	072526	125530
	037777	000040	001200	004000	100000	040000	177742	000000
	000000	000000	000015	102500	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

3,3 — BASE PAGE LINKS FOR DVRØ1

076	010241	010313	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

3,4 — DVRØ2

077	000000	016420	160213	010056	050054	126241	050055	026300
	160213	001727	001222	010073	050063	026263	050064	026267
	060055	126241	062442	170216	002400	026276	160214	002021
	003004	002003	003400	170216	060065	170217	026313	160214
	001200	170216	160215	002020	026310	001000	003004	170217
	002400	170220	062441	072320	026322	002400	126241	000000
	016420	160213	070001	001727	001200	072440	002400	150216
	026402	004010	026360	150217	026364	164216	134216	004065
	160001	002041	001727	010074	066440	134217	026354	006020
	026354	052436	026402	102600	102700	036320	126320	160217
	134216	000000	026354	062440	002020	026402	164220	062435
	056435	060065	170220	056435	026377	026354	006400	174216
	026354	102500	070001	160206	010075	030001	170205	002400
	006002	060055	164215	006020	007004	106700	126320	000000
	032437	072402	040067	072354	042433	072355	032434	072416
	126420	000040	001100	004000	000015	000137	102500	000000

3,5

078	010315	177610	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

DRS

	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

3,6 ← BASE PAGE LINKS FOR DVR02

079	010241	013327	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

3,7

DVR22

080	000000	164237	005600	064257	005623	174207	016700	073001
	002001	015536	160213	010056	050056	026401	004033	026252
	000031	026346	005310	026546	016524	016550	076550	007007
	026376	044045	160214	006121	026330	072536	060045	072524
	072612	060257	067037	050062	067046	076521	160213	001727
	001226	002541	063055	170001	006004	036612	026314	066521
	162536	170001	036536	006004	036550	026321	062521	016633
	065261	063030	016657	010266	016557	067056	062345	016657
	010336	064064	063034	016657	010342	160215	026372	016612
	062050	170217	016550	160214	016633	067057	063035	016657
	010353	016557	134217	002001	026567	067056	062361	016657
	010366	160213	010067	002002	026440	060057	126241	016536
	160213	013050	053061	026376	004033	026400	053062	026422
	053063	026440	053064	026445	053065	026452	050067	026476
	026531	016625	067056	063031	016657	010423	002400	170207
	006401	006404	063001	002020	026376	002400	127001	016524

3,8

081	064056	063036	016657	010441	016625	067066	063031	016657
	010446	016521	010067	002003	026462	064056	062475	016657
	010456	067067	107627	016521	001323	001310	026472	102727
	026427	067063	062475	016657	010462	005310	026546	016612
	064064	062510	016657	010501	067070	063033	016657	010505
	016521	013071	002003	026432	067056	063032	016657	010515
	000000	102527	126521	000000	016521	013072	002103	126524
	067001	060055	006021	127001	126241	000000	005727	006020
	026546	067063	062536	016657	110536	060056	126241	000000
	164215	006321	007005	005100	076524	126550	003000	016521
	013071	002002	126557	160213	002011	026673	016521	001727
	002020	026606	102502	002002	033373	003004	016550	044000
	007004	160215	002020	005000	026436	164215	006020	007004
	026436	000000	016521	013072	002303	126612	160207	001623
	170207	002340	026531	126612	020000	016521	010067	002002
	026427	126625	000000	036635	177777	146702	001225	102602

3,9

082	062557	033074	102606	102702	062524	142602	126633	003400
	072635	162657	001275	160000	067063	002001	000000	170216

DRS

DVR22

	107627	016521	001323	001310	026650	002400	052635	103726
	102727	036635	103706	063001	002007	126241	124000	000000
	072557	064000	062706	016775	072706	106726	002004	072711
	106727	052670	016775	072670	006004	062463	016775	072463
	072661	062470	016775	072470	072671	062522	016775	072522
	064257	062643	016775	072643	073013	040047	072640	072646
	062673	016775	072673	062747	016775	072747	106706	040047
	044047	072636	062573	016775	072573	062644	016775	072644
	016521	010074	070001	160206	010075	030001	170206	016521
	070001	003400	072635	126700	000000	010077	030001	126775
	000000	164207	005265	006002	050001	027022	074257	040052
	016700	002400	102606	160216	004050	002405	124000	006400
	127001	070001	063026	016775	073026	106700	026674	010335
	010427	010500	010511	010265	010362	010431	011040	020040
3,10								
083	020040	020040	020040	020040	020040	011047	020040	020040
	020040	020040	020040	020040	020040	000041	000023	017700
	000600	000200	000300	000400	000500	000201	000101	000035
	000022	000040	140000	060000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
3,11			BASE PAGE LINKS FOR DVR22					
084	010241	011201	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
3,12								
085	160001	002002	131003	171004	006004	160001	002005	026012
	131005	171006	006004	160001	171007	006004	160001	171010
	006004	160001	171011	002400	171012	171013	171014	060527
	151015	026036	171016	002400	165017	115020	060255	171021
	171022	003004	171023	040256	171024	141025	171026	002004
	171027	171030	161025	003004	040255	171031	060045	040100
	171032	171033	161024	003004	171034	165025	002400	170001
	006004	135034	026067	114535	012101	000066	020677	015171
	020771	161035	003004	141036	070001	040051	002020	027307
	161036	002004	001727	073037	161035	003004	141036	001100
	141035	001727	171037	001727	064155	054161	026132	060154

LOADR

FOR 32 SECTORS

[SLIDE AD-24]

13,6

EXEC MODULE DOU... TA... CRS 4

001	004003	000413	014004	000415	004005	000421	004007	000423
	004010	000425	010011	000427	004012	001002	004014	001004
	014015	001006	014016	001012	010017	001016	004020	001021
	010023	031023	010024	001026	000400	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	124112	037236
	037567	037135	037405	016000	036514	036515	001633	007777
	006000	002266	004134	004000	004150	005010	005035	005074
	005152	003123	003151	002135	006000	000471	000513	000534
	010132	010144	010072	010030	010000	011135	011210	000765
	011005	011114	001461	010021	005177	003136	007660	010755
	015400	015600	015732	006004	005160	005324	003525	005334
	001160	001163	001177	001211	001223	012246	002151	000522
	000523	000673	007726	012000	000207	001350	015747	015750

13,7

RELOCATABLE LIBRARY

002	010400	020000	167137	040523	041511	044440	000072	000000
	000000	000006	000143	000000	000000	000000	000000	000000
	000000	005400	040002	170452	041516	042105	041400	000000
	041516	047503	052000	000004	036000	060135	140462	000000
	001320	000000	064041	016000	000010	126000	000000	000000
	013212	064043	016000	000010	126000	000004	000000	076000
	000071	133332	066000	000047	076000	000043	076000	000044
	076000	000045	066000	000042	133320	076000	000046	016000
	000050	146000	000046	176000	000046	002003	132132	026000
	000034	016000	000050	005727	146000	000046	176000	000046
	121320	036000	000046	002042	026000	000020	066000	000045
	010400	060106	121765	000035	133332	062000	000043	072000
	000045	076000	000043	066000	000042	126000	000010	020000
	000043	021000	060122	052445	000046	000012	000000	030000
	000000	006400	076000	000070	013212	070001	042000	000071
	036000	000070	002021	026000	000053	013200	006020	026000

13,8

70R TOTAL OF 147 SECTOR

003	000054	065000	000071	007004	044000	121200	062000	000070
	040052	126000	000050	003000	060101	060172	000071	000000
	000000	002000	120000	120000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

13,9

004	010400	020000	005144	042125	045522	054040	000064	000000
	000000	000006	000143	000000	000000	000000	000000	000000
	000000	005400	040002	142464	022114	044502	051000	000000
	022114	044502	054000	000026	035400	050136	033267	000000
	013320	000000	072000	000056	162000	000000	036000	000000
	002002	133200	026000	000010	036000	000061	026000	000024
	002004	002004	121332	076000	000057	005500	076000	000060
	066000	000000	046000	000062	001212	164001	174000	066000

[SLIDE AD-25]

CRS,

19,11

```
001  124003  004167  114532  114532  114532  114532  114532  114532
     114532  114532  114532  114532  114532  114532  114532  114532
     114532  114532  114532  114532  114532  114532  114532  114532
     114532  114532  114532  114532  114532  114532  177700  177766
     177767  177770  177771  177772  177773  177774  177775  177776
     177777  000000  000001  000002  000003  000004  000005  000006
     000007  000013  000011  000012  000021  000100  000017  000037
     000077  003177  000377  177400  003777  177700  037633  000000
     000000  000012  000000  000000  000000  000000  000000  000001
     000001  000000  000003  013004  000030  007361  000005  007506
     000007  037634  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  003000  000000  000000  000000
     000000  000000  060307  000001  000001  013400  013400  000001
     000000  000377  000377  000000  000000  000000  000000  000000
     000000  000000  000003  000001  021450  000000  013000  007515
```

19,12

```
002  000016  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  007533  000016  000000  007567  010241  000732  001000
     010241  001000  012000  001002  001633  000000  000000  000000
     000000  002024  010453  000412  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
```

19,13

```
003  000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  000000
     000000  000000  000000  000000  000000  000000  000000  177750
     000000  000000  000000  000000  000000  000000  000000  000000
     006407  000120  000021  000000  000000  000000  114532  114532
     001634  004160  004154  002214  001400  003511  003020  002035
     003744  003361  003426  004165  004267  002507  004010  003244
     003345  003177  001714  003563  004044  002212  002653  002657
     002661  002663  002665  002553  003214  003207  003143  002576
     002700  002631  002521  003130  003117  004243  002445  004302
```

19,14

```
004  004303  004314  004376  000000  004400  000000  004446  000000
     000000  000000  000000  004505  000000  000000  000000  000000
     000000  004671  005264  000000  000000  002322  002222  002657
     004262  004151  001746  004034  004125  004304  004305  001756
     001725  001723  004300  004301  004307  004272  004312  004313
     004310  004311  004274  004150  004126  004127  004135  004124
     004005  004024  004136  004035  004137  004277  004131  004132
     004133  004134  004270  004275  004130  004145  004146  004147
```

JSB * CIC

CONSTANTS

BASE PAGE COMMUNICATION AREA

DISCM ENTRY POINTS LINKAGE

EXEC MODULES ENTRY LINKS

DISCM PAGE BOUNDARY LINKS

[SLIDE AD-26]

CRS1

DISCM PAGE LINKS

EXEC MODULE LINKS

	003607	002213	003743	103575	003325	002463	003204	003213
	003220	003236	003003	005774	006411	006266	006354	006336
	006372	105774	006330	006315	006413	006464	006660	006743
	000001	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000
	000000	000000	000000	000000	000000	000000	000000	000000

19,15

005	023776	154025	117770	060413	115154	041036	102074	012676
	025574	053370	126760	064446	151114	130736	070402	161004
	150516	127742	066412	155024	140556	110042	026612	055424
	133050	074626	000162	000344	000710	001620	003440	007100
	016200	034400	071000	162000	152506	133722	076352	003432
	007064	015150	034320	070640	161500	151706	132322	073352
	166724	164356	157442	145612	122132	052772	125764	062456
	145134	123776	050502	121204	051116	122234	053176	126374
	063476	147174	125076	060702	141604	112116	032742	065704
	153610	136126	102762	014452	031124	062250	144520	117746
	046422	115044	040616	101434	011576	023374	046770	115760
	042446	105114	020736	041674	103570	016066	034154	070330
	160660	150246	127222	065152	152324	133356	075442	001612
	003424	007050	016120	034240	070500	161200	151106	130722
	070352	160724	150356	127442	065612	153424	135556	102042
	012612	025424	053050	126120	062746	145714	122336	103216

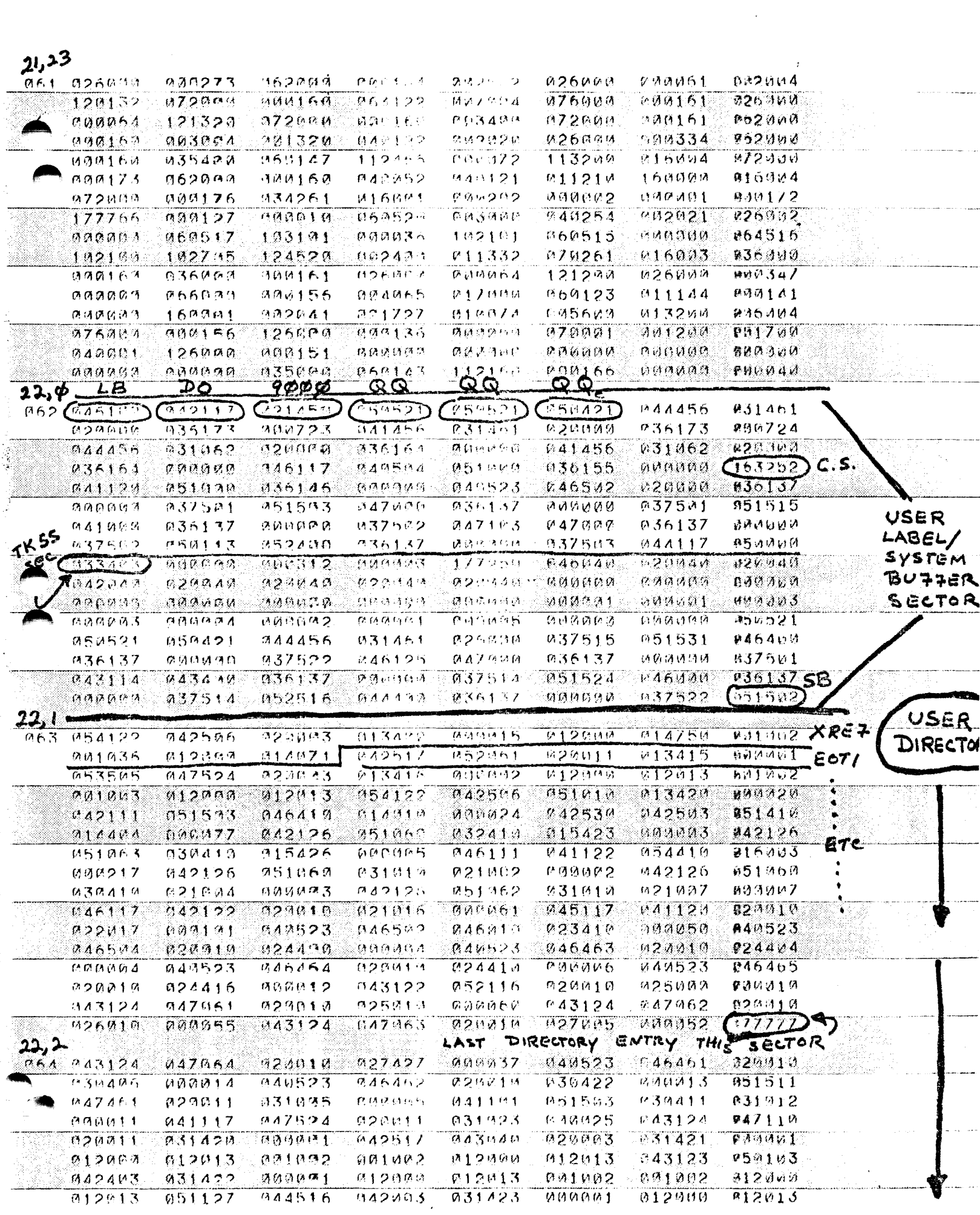
19,16

0	002502	175321	012410	165774	052040	124100	056706	135614
	102136	013002	026004	054010	130020	066546	155314	141336
	111402	031512	063224	146450	123626	056162	134344	077416
	005542	013304	026610	055420	133040	074606	000122	000244
	000510	001220	002440	005100	012200	024400	051000	122000
	052506	125214	061136	142274	113276	035302	072604	165410
	161526	151752	132452	073632	167464	165656	162242	153212
	135132	100772	010472	021164	042350	104720	020346	040714
	101630	012166	024354	050730	121660	052246	124514	057736
	137674	106276	023302	046604	115410	041526	103254	015236
	032474	065170	152360	133446	075622	002152	004324	010650
	021520	043240	106500	023706	047614	117430	045566	113354
	035436	073074	166170	163066	154662	140252	107232	025172
	052364	124750	060426	141054	110636	030202	060404	141010
	110526	027762	057744	137710	106326	023362	046744	115710
	042326	104654	020236	040474	101170	011066	022154	170601

19,17

007	110660	067143	060514	117671	111166	031062	062144	144310
	117326	045362	112744	034416	071034	162070	152666	134262
	077252	005232	012464	025150	052320	124640	060206	140414
	107536	026002	054004	130010	066526	155254	141236	111202
	031112	062224	144450	117626	046162	114344	037416	077034
	004576	011374	022770	045760	113740	036406	075014	000536
	001274	002570	005360	012740	025700	053600	127400	065506
	153214	135136	101002	010512	021224	042450	105120	020746
	041714	103630	016166	034354	070730	161660	152246	133222
	075152	001032	002064	004150	010320	020640	041500	103200
	015106	032214	064430	151060	134046	070222	160444	147616
	126142	063012	146024	122556	054042	130104	066716	155634
	142176	113102	034712	071624	163450	155626	142162	113052
	034632	071464	163150	155026	140562	110052	026632	055464
	133150	075026	000562	001344	002710	005620	013440	027100
	056200	134400	077506	005722	013644	027510	057220	127110

REMAINDER OF TRACK 19 NOT USED


21,23
22,0
LB
DO
QQ
QQ
QQ
C.S.
USER LABEL/ SYSTEM BUFFER SECTOR
TK55 SEC
SB
22,1
XREF
EOT1
USER DIRECTORY
ETC
LAST DIRECTORY ENTRY THIS SECTOR
22,2

END OF USER DIRECTORY
NOT LAST DIRECTORY ENTRY THIS SECTOR
177777
22,3
22,4
22,5
REST OF TRACK 22 IS RESERVED FOR USER DIRECTORY ENTRIES

VI. INTERNAL SYSTEM OPERATION

A. IOMEC Command Sequences

1. Seek Record [SLIDE 69A]
2. Read Data [SLIDE 69B]
3. Write Data [SLIDE 69C]
4. Check Data [SLIDE 69D]
5. Status Check [SLIDE 69E]
6. Constants and Storage [SLIDE 69F] --(Second Projector)

B. Supplied DOS-M Bootstrap Listing Study ("A" Version)

1. Configuration Section (Start Address =2) [SLIDES 70D - 70F]
   a. Data Channel in Switches Ø-5
   b. Configure DMA Control Word
   c. Loop to Configure all D.C. and C.C. Instructions
   d. Switch 15 Down -- HALT
      Switch 15 UP -- Punch Configured Bootstrap in absolute tape format.
2. Execution Section (Start Address =1ØØB or 5) [SLIDES 70A-70C]
   a. Loads Disc Resident Bootstrap form Track Ø Sectors 1 & 2 on Subchannel specified to memory locations 154ØØ - 15777 octal.
   b. Sets locations in DRB as follows:

      15771 ← Ø.
      15772 ← Head # (Bits 8-9).
      15776 ← System Disc Drive #.
      15773, 15774 ← Data Channel I/O select code for Run Time Disc.
      15775 ← Command Channel I/O select code for Run Time Disc.

   c. Uses address in ASPBF of DRB (which was set earlier during generation) to see what page LWAM declared in adjusts for one page lower for relocation of Disc Resident Bootstrap.
   d. Adjusts DEF's in DRB for correct page (ASPBF, DEFDY, and DVADR).
   e. Relocates entire DRB to new page -- only does page relative move.

f. Transfers control to relocated DRB to have it bring in Core Resident DOS-M System from Disc.

C. Disc Resident Bootstrap (DRB) Listing Study [SLIDES 71A - 71F]

1. Configures all Disc I/O instructions (within locations 15643-15724 octal) according to Data Channel already setup in location 15773 by supplied bootstrap.
2. Calls PLOAD routine four times to load in the four sections of Core Resident System defined in ASPBF through ASPBF+11 (3 words per load). SLOAD reads a sector at a time. Note how DMA Control Word does not have to be output again here because it was already outputted by Supplied Bootstrap in loading DRB.
3. Examine Equipment Table of CRS just loaded to accomplish the following:
   a. For DVRØ5 - configure all I/O instructions in this System TTY driver.
   b. For DVR31 - modify its EQT entry for "RUN TIME DISC" channels. Configure all I/O instructions in this driver. Configure DMA control word in this driver. Set + and - # sectors per track locations in this driver.
4. Set RUN TIME DISC channel in DISCO (base page location 154B; bits 15-11).
5. Set RUN TIME DISC Interrupt Table entries in correct interrupt table entry and also put correct entry in interrupt table where Generator Disc Chahnel entries were.
6. Set new I/O channels in Equipment Table for devices swapped with RUN TIME DISC.
7. Set RUN TIME SYSTEM Subchannel in Base Page location 155 octal from information passed by Supplied Bootstrap.
8. Set User Label TRACK/SECTOR Disc Address (Base Page location 2ØØ octal) according to Base Page locations 155 and 175 octal.
9. Set Next TRACK/SECTOR Address on Current User Disc (Base Page Location 157 octal).
10. Set Current User Disc Subchannel # (location 161) equal to Default User Disc Subchannel # (location 175).
11. Branch to location 3 indirect to start DOS-M System.

D. DOS-M System Halts [SLIDE 72]

These Halts are only during System Operation following successful bootstrap.

E. DOS-M I/O Request Processing Example

1. Materials used
   a. Tape Recording of step by step execution
   b. Large chart of DOS-M I/O Request Processing
   c. Foldout flowcharts on DOS-M (next section)
   d. Slides
      (1) Equipment Table Format; # 10B
      (2) Device Reference Table Format; # 10C
      (3) Interrupt Table Format; # 10D

2. Procedure
   a. Using the above tape recording and materials, trace an example I/O Request Operation (b below) through all major steps that occur from the initial EXEC call until the I/O operation is complete.
   b. The example I/O operation will be:

```
      JSB EXEC          (Call to Executive Supervisor)
      DEF RTN           (Define Return Address)
      DEF RCODE         (Define Request Code Address)
      DEF CONWD         (Define Control Word Address)
      DEF BUFA          (Define Buffer Start Address)
      DEF BUFL          (Define Buffer Length Address)
RTN (Return Point)
        .
        .
        .
RCODE DEC 1             (Read Operation R.C.)
CONWD OCT 5             (Logical Unit 5)
BUFA  BSS 36            (36 word or 72 char. Buffer)
BUFL  DEC -72           (72 character length)
```

```
0004*
0005***** SEEK RECORD *****
0006*
0007  00002 000000  SEEK  NOP
0008  00003 054155       LDB CYL        CYLINDER (TRACK) NUMBER
0009  00004 106510       OTB DC         OUTPUT CYL# TO DATA CHNL      NOTE 1
0010  00005 103710       STC DC,C
0011  00006 000136       LDA SKCMD      LOAD SEEK COMMAND
0012  00007 054154       CPB LSTSK      * IF CYL# = LAST CYL# ACCESSED,
0013  00010 030151       IOR MSIGN      *   CHANGE CMND TO ADDRS RECORD
0014  00011 074154       STB LSTSK      * UPDATE LAST SEEK INDICATOR
0015  00012 030157       IOR DRV        INCLUDE DRIVE#
0016  00013 100711       CLC CC                                       NOTE 2
0017  00014 102611       OTA CC         OUTPUT SEEK/ADDRESS COMMAND
0018  00015 103711       STC CC,C         TO CMND CHNL
0019  00016 102310       SFS DC   <-- WAIT FOR CYL# ACCEPTANCE
0020  00017 024016       JMP *-1
0021  00020 060156       LDA HDSCT      HEAD:BITS 15-6; SECTOR:BITS 7-0
0022  00021 102610       OTA DC         OUTPUT HEAD/SECTOR TO DATA CHNL
0023  00022 103710       STC DC,C                    NOTE 3
0024  00023 102311       SFS CC         * IN INTERRUPT MODE, EXIT HERE
0025  00024 024023       JMP *-1        *   AND RETURN ON INTERRUPT
0026  00025 014121       JSB STAT <-- (CHECK STATUS)
0027  00026 124002       JMP SEEK,I
0028*
0029* NOTE: EITHER A SEEK (03020XXB) OR AN ADDRESS RECORD (13000XXB)
0030*       COMMAND MUST BE ISSUED PRIOR TO ANY OTHER DISC COMMAND
0031*       EXCEPT STATUS CHECK. ADDRESS RECORD MAY BE ISSUED ONLY
0032*       IF THE HEAD (ON THE SELECTED DRIVE) IS ALREADY IN POSITION
0033*       FOR THE CURRENT ACCESS; IT WILL EXECUTE FASTER THAN A
0034*       SEEK UNDER THESE CONDITIONS. THE CODING IN THE EXAMPLE
0035*       FOR CHECKIN THIS CONDITION IS APPLICABLE ONLY TO A SINGLE
0036*       DRIVE SYSTEM. IN A MULTI-DRIVE ENVIRONMENT, PROVISION
0037*       MUST BE MADE TO MAINTAIN AND CHECK THE HEAD POSITION
0038*       INDICATOR FOR EACH DRIVE.
0039*
```

NOTE 1 : BITS 0-7 = CYLINDER # ; BITS 8-15 = 0

NOTE 2 : BITS 12-15 = COMMAND CODE ; BITS 0-1 = DRIVE #

NOTE 3 : BITS 8-9 = HEAD # ; BITS 0-3 = SECTOR #

```
0041*
0042***** READ DATA *****
0043*
0044  00027 000000  READ  NOP
0045  00030 060152        LDA DMACW      LOAD DMA CONTROL WORD,
0046  00031 030153        IOR DCHNL        INCLUDE DATA CHNL#,
0047  00032 102606        OTA 6            AND OUTPUT TO DMA CHNL
0048  00033 102702        CLC 2
0049  00034 060160        LDA CORAD      LOAD CORE BUFFER ADDRESS,
0050  00035 030151        IOR MSIGN        INCLUDE DIRECTION BIT,
0051  00036 102602        OTA 2            AND SET IN MEMORY ADDRESS REG
0052  00037 102702        STC 2
0053  00040 060161        LDA WDCNT      LOAD NEG # OF WORDS
0054  00041 102602        OTA 2            AND SET IN WORD COUNT REG
0055  00042 103710        STC DC,C       SET DATA CHNL FOR INPUT      NOTE 1
0056  00043 060140        LDA RDCMD      LOAD READ COMMAND
0057  00044 030157        IOR DRV        INCLUDE DRIVE#
0058  00045 106711        CLC CC                                      NOTE 2
0059  00046 102611        OTA CC         OUTPUT READ COMMAND TO CMND CHNL
0060  00047 103706        STC 6,C        START DMA
0061  00050 103711        STC CC,C       START DATA TRANSFER
0062  00051 106706        CLC 6          INHIBIT DMA INTERRUPT
0063  00052 102311        SFS CC         * IN INTERRUPT MODE, EXIT HERE
0064  00053 024052        JMP *-1        *   AND RETURN ON INTERRUPT
0065  00054 014121        JSB STAT       (CHECK STATUS)
0066  00055 124027        JMP READ,I
```

NOTE 1 : PREPARES DATA CHANNEL TO RECEIVE DATA FROM CONTROLLER

NOTE 2 : BITS 12-15 = READ COMMAND ; BITS 0-1 = DRIVE #

```
0068*
0069***** WRITE *****
0070*
0071  00056 000000  WRITE NOP
0072  00057 060152        LDA DMACW      LOAD DMA CONTROL WORD,
0073  00060 030153        IOR DCHNL        INCLUDE DATA CHNL#,
0074  00061 102606        OTA 6            AND OUTPUT TO DMA CHNL
0075  00062 106702        CLC 2
0076  00063 060160        LDA CORAD      LOAD CORE BUFFER ADDRESS
0077  00064 102602        OTA 2            AND SET IN MEMORY ADDRESS REG
0078  00065 102702        STC 2
0079  00066 060161        LDA WDCNT      LOAD NEG # OF WORDS
0080  00067 102602        OTA 2            AND SET IN WORD COUNT REG
0081  00070 102110        STF DC         SET DATA CHANNEL FOR OUTPUT
0082  00071 060144        LDA WRCMD      LOAD WRITE COMMAND
0083  00072 030157        IOR DRV        INCLUDE DRIVE#
0084  00073 106711        CLC CC
0085  00074 102611        OTA CC         OUTPUT WRITE CMND TO CMND CHNL
0086  00075 105706        STC 6,C        START DMA
0087  00076 105711        STC CC,C       START DATA TRANSFER
0088  00077 106706        CLC 6          INHIBIT DMA INTERRUPT
0089  00100 102311        SFS CC         * IN INTERRUPT MODE, EXIT HERE
0090  00101 024100        JMP *-1        *   AND WAIT FOR INTERRUPT
0091  00102 014121        JSB STAT <---(CHECK STATUS)
0092  00103 124056        JMP WRITE,I
0093*
0094* NOTE: THE *WRITE* SEQUENCE ABOVE MAY BE USED,
0095*       WITH THE APPROPRIATE COMMAND, AS FOLLOWS:
0096*
0097*         COMMAND (OCTAL)      FUNCTION
0098*
0099*              010000          WRITE DATA
0100*              110000          INITIALIZE DATA
0101*                                (USED TO  INITIALIZE ADDRESS
0102*                                 FIELDS OF A NEW DISC)
0103*              110400          FLAG DEFECTIVE CYLINDER
0104*              111000          FLAG PROTECTED CYLINDER
0105*
0106*       ALL EXCEPT WRITE DATA REQUIRE THAT THE
0107*         DISC PROTECT OVERRIDE SWITCH BE TURNED ON
0108*
0109*       ALL OF THESE COMMANDS ACTUALLY WRITE DATA IN
0110*         THE SECTOR(S) BEING PROCESSED. IF THE WCR
0111*         GOES TO ZERO BEFORE THE END OF THE SECTOR
0112*         IS REACHED, THE REMAINDER OF THE SECTOR
0113*         WILL BE FILLED WITH ZEROS; THUS IF ZERO
0114*         WORDS ARE SPECIFIED, THE ENTIRE SECTOR IS
0115*         WRITTEN WITH ZEROS.
0116*
0117*       FLAG CYLINDER PROTECTED OR DEFECTIVE ARE SUBSETS
0118*         OF THE INITIALIZE DATA COMMAND, AND WRITE THE
0119*         ADDRESS FIELD(S) OF THE SECTOR(S) BEING PROCESSED.
0120*
```

```
0122*
0123***** CHECK DATA *****
0124*
0125  00104 000000  CHECK NOP
0126  00105 060162        LDA SCTRS     LOAD SECTOR COUNT TO BE CHECKED
0127  00106 102010        OTA DC          AND OUTPUT TO DATA CHANNEL
0128  00107 103710        STC DC,C                                  NOTE
0129  00110 060141        LDA CHCMD     LOAD CHECK DATA COMMAND       1
0130  00111 030157        IOR DRV       INCLUDE DRIVE#
0131  00112 106711        CLC CC
0132  00113 103611        OTA CC,C      OUTPUT CHECK COMMAND          NOTE 2
0133  00114 103711        STC CC,C        TO CMND CHNL
0134  00115 102311        SFS CC        * IN INTERRUPT MODE, EXIT HERE
0135  00116 024115        JMP *-1       *   AND RETURN ON INTERRUPT
0136  00117 014121        JSB STAT <--(CHECK STATUS)
0137  00120 124104        JMP CHECK,I
```

NOTE 1: A = +(# SECTORS TO BE CHECKED) BITS 0-4

NOTE 2: BITS 12-15 = CHECK DATA COMMAND
BITS 0-1 = DRIVE #

THE CONTROLLER EXECUTES THIS COMMAND MUCH AS IT DOES READ DATA; HOWEVER, NO TRANSFER OF DATA OCCURS. RESULTS OF CHECK MAY BE OBTAINED WITH STATUS COMMAND CALL.

```
0139*
0140***** STATUS CHECK *****
0141*
0142  00121 000000  STAT  NOP
0143  00122 103100        CLF 0          (TURN OFF INTERR SYS IF IT'S ON)
0144  00123 103710        STC DC,C       SET DATA CHANNEL FOR INPUT
0145  00124 060157        LDA DRV        LOAD DRIVE#
0146  00125 106711        CLC CC
0147  00126 102611        OTA CC         OUTPUT STATUS COMMAND
0148  00127 103711        STC CC,C         TO CMND CHANNEL
0149  00130 102310        SFS DC
0150  00131 024130        JMP *-1
0151  00132 106711        CLC CC
0152  00133 102510        LIA DC         GET STATUS FROM DATA CHNL
0153  00134 102100        STF 0          (RESET INTERR SYS IF IT WAS ON)
0154  00135 124121        JMP STAT,I
0155*
0156*  NOTE: IOMEC STATUS BITS
0157*
0158*        15 - ATTENTION - OPERATION COMPLETED
0159*        14 - FIRST SEEK - DRIVE HAS GONE FROM NOT READY TO READY
0160*        13 - OVERRUN - LATE DATA TRANSFER - HARDWARE FAILURE
0161*        12 - READ/WRITE UNSAFE - HARDWARE FAILURE
0162*        11 - ACCESS UNSAFE - HARDWARE FAILURE
0163*        10 - ACCESS HUNTING - HARDWARE FAILURE
0164*         9 - SEEK INCOMPLETE - HARDWARE FAILURE
0165*         8 - SEEK CHECK - SOFTWARE ERROR (E.G., CYL# > 202)
0166*         7 - (NOT USED)
0167*         6 - NOT READY (ALSO SET WHEN BITS 11 AND/OR 12 SET)
0168*         5 - END OF CYLINDER - SOFTWARE ERROR - ATTEMPTED TO
0169*               WRITE PAST THE END OF A CYLINDER
0170*         4 - ADDRESS ERROR - ADDRESS ISSUED DOES NOT AGREE WITH
0171*               DISC ADDRESS - HARDWARE FAILURE OR DISC NOT
0172*               INITIALIZED - OR - IF BIT 3 IS ALSO ON, THEN
0173*               THE CYLINDER BEING PROCESSED HAS BEEN FLAGGED
0174*               DEFECTIVE.
0175*         3 - FLAGGED CYLINDER - SET IF CYLINDER BEING PROCESSED
0176*             HAS BEEN FLAGGED PROTECTED OR (IF BIT 4 IS ALSO SET )
0177*             DEFECTIVE - OR - INITIALIZE DATA COMMAND HAS BEEN
0178*             ISSUED WITH DISC PROTECT OVERRIDE SWITCH OFF
0179*         2 - DRIVE BUSY - SEEK IN PROCESS
0180*         1 - DATA ERROR - CYCLIC CHECK INCORRECT
0181*         0 - ANY ERROR - TURNED ON WHEN ANY OF THE ABOVE EXCEPT
0182*               BIT 15 OR, ON A READ OR CHECK DATA, BIT 3, IS SET.
0183*
0184* NOTE: ANY HARDWARE FAILURE WHICH DOES NOT SET BIT 6 MAY BE
0185*       RECOVERABLE ON RETRY.
0186*
```

STF 6

[SLIDE 69E]

```
0188*
0189***** DATA CONSTANT AND STORAGE AREA *****
0190*
0191  00010            DC     EQU 10B          DATA CHANNEL (HIGH PRIORITY)
0192  00011            CC     EQU 11B          COMMAND CHANNEL (LOW PRIORTY)
0193*
0194  00136 030000  SKCMD OCT 030000          SEEK RECORD COMMAND
0195  00137 130000  ADCMD OCT 130000          ADDRESS RECORD COMMAND
0196  00140 020000  RDCMD OCT 020000          READ DATA COMMAND
0197  00141 060000  CHCMD OCT 060000          CHECK DATA COMMAND
0198  00142 000000  STCMD OCT 000000          STATUS CHECK COMMAND
0199  00143 050000  RFCMD OCT 050000          REFINE SECTOR COMMAND
0200*
0201  00144 000000  WRCMD NOP                 STORAGE FOR CURRENT WRITE CMND
0202*
0203  00145 010000  WDCMD OCT 010000          WRITE DATA COMMAND
0204  00146 110000  INCMD OCT 110000          INITIALIZE DATA COMMAND
0205  00147 111000  PCCMD OCT 111000          FLAG PROTECTED CYLINDER CMND
0206  00150 110400  DCCMD OCT 110400          FLAG DEFECTIVE CYLINDER CMND
0207*
0208  00151 100000  MSIGN OCT 100000          BIT 15
0209  00152 120000  DMACW OCT 120000          DMA CONTROL WORD
0210*
0211  00153 000010  DCHNL OCT 10              DISC DATA CHNL# (HP)
0212  00154 000313  LSTSK DEC 203             LAST SEEK IND. (INIT. > 202)
0213*
0214  00155 000000  CYL    NOP               CYLINDER#
0215  00156 000000  HDSCT  NOP               HEAD#(15-8), SCTR#(7-0)
0216  00157 000000  DRV    NOP               DRIVE#
0217  00160 000000  CCRAD  NOP               CORE BUFFER ADDRESS
0218  00161 000000  WDCNT  NOP               NEG #WORDS TO BE TRANSFERRED
0219  00162 000000  SCTRS  NOP               POS #SCTRS TO BE CHECKED
0220*
0221                       END
** NO ERRORS*
```

[SLIDE 69F]

DOS-M SUPPLIED BOOTSTRAP ("A" VERSION)

```
0001                    ASMB,L,A
0002*
0003  00002                   ORG 2B
0004  00002 024223            JMP CONFG       GO CONFIGURE BOOTSTRAP
0005*
0006  00005                   ORG 5B
0007  00005 102501  BOOT      LIA 1           GET SYSTEM SUBCHNL
0008  00006 010200            AND M7
0009  00007 000065            CLE,ERA
0010  00010 070166            STA DRV#        SET DRIVE#
0011  00011 006400            CLB
0012  00012 106600            OTB DC          OUTPUT TRK#
0013  00013 103700            STC DC,C          FOR SEEK
0014  00014 030204            IOR SEEK
0015  00015 102601            OTA CC          OUTPUT SEEK CMND
0016  00016 103701            STC CC,C          TO COMMAND CHNL
0017  00017 102300            SFS DC          <-- WAIT FOR TRACK # RECEIVED
0018  00020 024017            JMP *-1
0019  00021 060206            LDA HDSCT
0020  00022 002040            SEZ             SUBCHNL ON REMOVABLE PACK?
0021  00023 010200            AND M7            -YES, SET HEAD# = 0
0022  00024 102600            OTA DC          OUTPUT HEAD/SCTR
0023  00025 103700            STC DC,C          TO DATA CHNL
0024  00026 010203            AND M1774
0025  00027 070170            STA HDMSK
0026  00030 060207            LDA DMACW
0027  00031 102606            OTA 6           OUTPUT DMA CNTRL WORD
0028  00032 106702            CLC 2
0029  00033 060210            LDA MEMAD
0030  00034 102602            OTA 2           OUTPUT BFR ADDRS
0031  00035 102702            STC 2
0032  00036 060173            LDA N256
0033  00037 102602            OTA 2           OUTPUT WORD COUNT
0034  00040 102301            SFS CC          WAIT FOR SEEK
0035  00041 024040            JMP *-1           TO COMPLETE
0036  00042 014056            JSB STAT        <-- CHECK STATUS
0037  00043 060205            LDA READ
0038  00044 030166            IOR DRV#
0039  00045 106701            CLC CC
0040  00046 102601            OTA CC          OUTPUT READ COMMAND
0041  00047 103700            STC DC,C        SET DATA CHNL FOR READ
0042  00050 103706            STC 6,C         START DMA
0043  00051 103701            STC CC,C        START READ OPERATION
0044  00052 102301            SFS CC          WAIT FOR READ
0045  00053 024052            JMP *-1           TO COMPLETE
0046  00054 014056            JSB STAT        <-- CHECK STATUS
0047*
0048  00055 024110            JMP RELOC       GO RELOCATE BOOTSTRAP
0049*
0050  00056 000000  STAT      NOP
0051  00057 103700            STC DC,C
0052  00060 060166            LDA DRV#
0053  00061 106701            CLC CC
0054  00062 102601            OTA CC          OUTPUT STATUS CMND
0055  00063 103701            STC CC,C          TO COMMAND CHNL
0056  00064 102300            SFS DC
```

PROPG BOOTSTRAP

SET-UP DMA

```
0057  00065 024064          JMP *-1
0058  00066 102500          LIA DC  <-------- GET STATUS
0059  00067 000010          SLA               ANY ERROR?
0060  00070 102011          HLT 11B            -YES
0061  00071 124056          JMP STAT,I
0062*
0063  00100                 ORG 100B          BOOTSTRAP START ADDRESS
0064  00100 024005          JMP BOOT
0065  00105                 ORG 105B
0066  00105 000222  DFEND   DEF CHSUM
0067*
0068  00110                 ORG 110B
0069*
0070  00110 002400  RELOC   CLA
0071  00111 170212          STA CLER1,I
0072  00112 170213          STA CLER2,I
0073  00113 060170          LDA HDMSK
0074  00114 170214          STA ABHDM,I       SAVE HEAD# FOR DISC-RES. BOOT
0075  00115 060166          LDA DRV#
0076  00116 170220          STA ABDRV,I
0077  00117 060165          LDA CHAN
0078  00120 170215          STA ACHNL,I       SAVE DISC I/O CHNLS
0079  00121 170216          STA ADCHN,I
0080  00122 002004          INA  .              FOR DISC-RESIDENT
0081  00123 170217          STA ACCHN,I         BOOTSTRAP
0082*
0083  00124 064221          LDB SPPNT
0084  00125 144221          ADB SPPNT,I                          ASPBF
0085  00126 160001          LDA B,I           GET ADDRESS OF [redacted]
0086  00127 010202          AND M76K          ISOLATE PAGE BITS
0087  00130 040176          ADA N2KB          SUBTRACT 1 PAGE
0088  00131 070171          STA PGMSK
0089  00132 160001          LDA B,I
0090  00133 010201          AND M1777
0091  00134 030171          IOR PGMSK                  ASPBF
0092  00135 170001          STA B,I           ADJUST [redacted] ADDRESS
0093  00136 044175          ADB N1
0094  00137 160001          LDA B,I
0095  00140 010201          AND M1777
0096  00141 030171          IOR PGMSK
0097  00142 170001          STA B,I           ADJUST DEFDY
0098  00143 044175          ADB N1
0099  00144 160001          LDA B,I
0100  00145 010201          AND M1777
0101  00146 030171          IOR PGMSK
0102  00147 170001          STA B,I           ADJUST DVADR
0103  00150 060211          LDA DBOOT
0104  00151 010201          AND M1777
0105  00152 030171          IOR PGMSK
0106  00153 070172          STA RELBT         SET TRANSFER ADDRESS
0107  00154 064173          LDB N256
0108  00155 074167          STB WDCNT
0109  00156 164211  MVMOR   LDB DBOOT,I       *
0110  00157 174000          STB A,I           *
0111  00160 002004          INA               * RELOCATE
0112  00161 034211          ISZ DBOOT         * BOOTSTRAP
```

BOOTSTRAP PTAPER

[SLIDE 70B]

```
0113  00162 034167          ISZ WDCNT      *
0114  00163 024156          JMP MVMOR      *
0115*
0116  00164 124172          JMP RELBT,I    TRANSFER TO DISC-RES. BOOTSTRAP
0117*
0118  00000           DC    EQU 0          DISC DATA CHANNEL
0119  00001           CC    EQU 1          DISC CMND CHANNEL
0120  00165 000000    CHAN  NOP
0121  00166 000000    DRV#  NOP
0122  00167 000000    WDCNT NOP
0123  00170 000000    HDMSK NOP
0124  00171 000000    PGMSK NOP
0125  00172 000000    RELBT NOP
0126  00173 177400    N256  DEC -256
0127  00174 177405    N251  DEC -251
0128  00175 177777    N1    DEC -1
0129  00176 176000    N2KB  OCT -2000
0130  00177 177700    N100  OCT -100
0131  00200 000007    M7    OCT 7
0132  00201 001777    M1777 OCT 1777
0133  00202 076000    M76K  OCT 76000
0134  00203 177400    M1774 OCT 177400
0135  00204 030000    SEEK  OCT 030000
0136  00205 020000    READ  OCT 020000
0137  00206 001001    HDSCT OCT 001001
0138  00207 120000    DMACW OCT 120000
0139  00210 115400    MEMAD OCT 115400
0140  00211 015400    DBOOT OCT 15400
0141  00212 015771    CLER1 OCT 15771
0142  00213 015772    CLER2 OCT 15772
0143  00214 015772    ABHDM OCT 15772
0144  00215 015773    ACHNL OCT 15773
0145  00216 015774    ADCHN OCT 15774
0146  00217 015775    ACCHN OCT 15775
0147  00220 015776    ABDRV OCT 15776
0148  00221 015777    SPPNT OCT 15777
0149*
```

BOOTSTRAP PROPER

[SLIDE 70C]

BOOTSTRAP CONFIGURATOR

```
0151*
0152  00222 000000  CHSUM NOP
0153*
0154  00223 102501  CONFG LIA 1          GET DISC DATA CHANNEL
0155  00224 010337        AND B77
0156  00225 070165        STA CHAN
0157  00226 030207        IOR DMACW
0158  00227 070207        STA DMACW      CONFIGURE DMA CNTRL WORD
0159  00230 060337        LDA B77
0160  00231 003000        CMA
0161  00232 040327        ADA DEFBT
0162  00233 070330        STA CNTR
0163  00234 064327        LDB DEFBT
0164  00235 006004  CLOOP INB            *
0165  00236 160001        LDA B,I        *
0166  00237 002021        SSA,RSS        * CONFIGURE ALL
0167  00240 024247        JMP CNEXT      *
0168  00241 010340        AND MASK       *   DISC I/O
0169  00242 002002        SZA            *
0170  00243 024247        JMP CNEXT      *   INSTRUCTIONS
0171  00244 160001        LDA B,I        *
0172  00245 040165        ADA CHAN       *   IN BOOTSTRAP
0173  00246 170001        STA B,I        *
0174  00247 034330  CNEXT ISZ CNTR
0175  00250 024235        JMP CLOOP
0176*
0177  00251 102501        LIA 1
0178  00252 002020        SSA            PUNCH CONFIGURED BOOTSTRAP?
0179  00253 024256        JMP *+3         -YES
0180*
0181  00254 102077        HLT 77B         -NO (HALT IRRECOVERABLE)
0182  00255 024254        JMP *-1
0183*
```

[SLIDE 70D]

PUNCH CONFIGURATED BOOTSTRAP

```
0185  00256 060334          LDA .2
0186  00257 070326          STA ABSAD      SET BOOTSTRAP START ADDRESS
0187  00260 003004          CMA,INA
0188  00261 040105          ADA DFEND
0189  00262 070331          STA TEMP       SAVE BOOTSTRAP END ADDRESS
0190  00263 001727          ALF,ALF
0191  00264 010203          AND M1774
0192  00265 070325          STA RCLNG      SET PUNCH RECORD LENGTH
0193*
0194  00266 002400          CLA
0195  00267 070002          STA 2B
0196  00270 070003          STA 3B
0197  00271 070004          STA 4B
0198  00272 060331          LDA TEMP
0199  00273 003004          CMA,INA
0200  00274 070330          STA CNTR       SET COUNTER FOR CHECKSUM
0201  00275 060326          LDA ABSAD
0202  00276 064334          LDB .2
0203  00277 140001  KLOOP   ADA B,I        *
0204  00300 006004          INB            * GENERATE
0205  00301 034330          ISZ CNTR       *   CHECKSUM
0206  00302 024277          JMP KLOOP      *   FOR BBL
0207  00303 070222          STA CHSUM      *
0208*
0209  00304 060332  PMORE   LDA N50
0210  00305 064341          LDB AFDFR
0211  00306 114103          JSB HSPDR,I    PUNCH LEADER
0212*
0213  00307 060333          LDA N2
0214  00310 064324          LDB SHREC
0215  00311 114103          JSB HSPDR,I    PUNCH RECORD LENGTH, ABS ADDRS
0216*
0217  00312 060331          LDA TEMP
0218  00313 003004          CMA,INA
0219  00314 002004          INA
0220  00315 064336          LDB .4
0221  00316 114103          JSB HSPDR,I    PUNCH BOOTSTRAP
0222*
0223  00317 060332          LDA N50
0224  00320 064341          LDB AFDFR
0225  00321 114103          JSB HSPDR,I    PUNCH TRAILER
0226*
0227  00322 102077          HLT 77B
0228  00323 024304          JMP PMORE
0229*
0230  00324 000325  SHREC   DEF *+1
0231  00325 000000  RCLNG   NOP
0232  00326 000000  ABSAD   NOP
0233  00000         A       EQU 0B
0234  00001         B       EQU 1B
0235  00103         HSPDR   EQU 103B
0236  00327 000005  DEFBT   DEF BOOT
0237  00330 000000  CNTR    NOP
0238  00331 000000  TEMP    NOP
0239  00332 177716  N50     DEC -50
0240  00333 177776  N2      DEC -2
```

```
0241   00334 000002   .2      OCT 2
0242   00335 000003   .3      OCT 3
0243   00336 000004   .4      OCT 4
0244   00337 000077   B77     OCT 77
0245   00340 070036   MASK    OCT 070036
0246*
0247   00341 000342   AFDFR   DEF *+1
0251                          LST
0252*
0253                          END
**   NO ERRORS*
```

DOS-M DISC RESIDENT BOOTSTRAP


```
0350  15400                ORG 15400B
0351*
0352*  THE FOLLOWING LOADER PERMITS LOADING OF THE RESIDENT PORTIONS
0353*  OF THE DISC MONITOR SYSTEM. THE LOADER IS LOCATED ON SECTORS 1 & 2
0354*  TRACK 0 OF THE SYSTEM DISC.  IT IS GENERATED BY THE SYSTEM
0355*  GENERATOR AND CONSISTS OF:
0356*
0357*  (1) THE INSTRUCTIONS REQUIRED FOR LOADING THE SYSTEM
0358*  (2) THE DISK AND CORE ADDRESSES SPECIFYING LOADING
0359*
0360*
0361*  THE ADDRESSES REQUIRED FOR LOADING ARE THE FOLLOWING:
0362*
0363*  (A) BASE PAGE LINKAGES
0364*      (1) LOW CORE ADDRESS
0365*      (2) HIGH CORE ADDRESS
0366*      (3) DISK ADDRESS OF ABSOLUTE CODE
0367*
0368*  (B) SYSTEM, RT RESIDENT MAIN
0369*      (1) LOW CORE ADDRESS
0370*      (2) HIGH CORE ADDRESS
0371*      (3) DISK ADDRESS OF ABSOLUTE CODE
0372*
0373*  RELEASED "A" VERSION 10-70
0374*
0375*
0376*
0377  15400 000000  START NOP
0378  15401 067731        LDB DEFDY
0379  15402 017570        JSB CNFGR     CONFG. BOOTSTRAP I/O INSTR.
0380  15403 067732        LDB ASPBF     GET APPRS OF DISC SPEC. BFR
0381  15404 077762        STB SPCAD     SET CURRENT SPBUF ADDRESS
0382  15405 017613        JSB PLOAD     LOAD BP LINKAGES
0383  15406 017613        JSB PLOAD     LOAD MAIN SYSTEM
0384  15407 017613        JSB PLOAD     LOAD I/O TABLES
0385  15410 017613        JSB PLOAD     LOAD EXEC DOUBLETS
0386*
0387  15411 064120        LDB BEQT#     GET # OF EQUIPMENT
0388  15412 007004        CMB,INB         TABLE ENTRIES AND
0389  15413 077310        STB CNTR        STORE NEGATIVE
0390  15414 064117        LDB BEQTB     GET FWA OF EQUIPMENT TABLE
0391  15415 044055  CNFG1  ADB .2
0392  15416 160001        LDA B,I       A = 3RD WORD OF EQT entry
0393  15417 044051        ADB .2N
0394  15420 010072        AND M.77
0395  15421 073773        STA CHANL     SAVE I/O CHANNEL#
0396  15422 053774        CPA RUND1      =RUN TIME DISC DATA CHNL?
0397  15423 077763        STB SWP1       -YES, SAVE
0398  15424 053775        CPA RUND2      =RUN TIME DISC CMND CHNL?
0399  15425 077764        STB SWP2       -YES, SAVE
0400  15426 077304        STB EQCUR     SAVE CURRENT EQPT TABLE ADDRESS
0401  15427 044056        ADB .3
0402  15430 160001        LDA B,I       A = 4 TH WORD OF EQT Entry
0403  15431 001727        ALF,ALF
0404  15432 013752        AND M.377
0405  15433 073305        STA EQPCD     SAVE EQPT TYPE CODE
```

LOAD CORE RESIDENT SYSTEM

DOS-M DISC RESIDENT BOOTSTRAP


```
0406  15434 050060          CPA .5          =SYSTEM TELETYPE?
0407  15435 027460          JMP CNFG4        -YES
0408  15436 053765          CPA DISK        =DISC ?
0409  15437 027445          JMP CNFG3        -YES
0410  15440 067304 CNFG2    LDB EQCUR
0411  15441 044066          ADB .17D        INCR TO NEXT EQPMT TABLE ENTRY
0412  15442 037310          ISZ CNTR        CHECKED ALL ENTRIES?
0413  15443 027415          JMP CNFG1        -NO
0414  15444 027505          JMP CNFG7        -YES
0415  15445 044052 CNFG3    ADB .1N
0416  15446 160001          LDA B,I
0417  15447 023773          XOR CHANL
0418  15450 033774          IOR RUND1
0419  15451 170001          STA B,I         SET DISC EQPT TABLE AT RUN TIME
0420  15452 063773          LDA CHANL
0421  15453 073302          STA GEND1       SAVE DISC I/O CHN2 AT GEN. TIME
0422  15454 002004          INA
0423  15455 073303          STA GEND2        SAVE GEN. DISC CMND CHNL
0424  15456 063774          LDA RUND1
0425  15457 073773          STA CHANL       SET CHAN= RUN TIME DISC CHNL
0426  15460 067304 CNFG4    LDB EQCUR
0427  15461 160001          LDA B,I
0428  15462 073730          STA DVADR       SAVE DRIVER ENTRY POINT
0429  15463 164000          LDB A,I         GET CONFIGURATION STOP POINT
0430  15464 017570          JSB CNFGR  <-- CONF. ALL I/O INST. IN DRIVER
0431*
0432  15465 063305          LDA EQPCD       GET EQPMT TYPE CODE
0433  15466 050060          CPA .5           =SYSTEM TELETYPE?
0434  15467 027440          JMP CNFG2        -YES
0435  15470 006004          INB              -NO, MUST BE DISC
0436  15471 160001          LDA B,I
0437  15472 033773          IOR CHANL       CONFIGURE DMA CNTRL WORD
0438  15473 170001          STA B,I          AND STORE
0439  15474 006004          INB
0440  15475 063747          LDA #SPTK       * SET + AND -
0441  15476 170001          STA B,I         *  SECTORS/TRACK
0442  15477 006004          INB             *    IN
0443  15500 003004          CMA,INA         *  DISC DRIVER
0444  15501 170001          STA B,I         *
0445  15502 067304          LDB EQCUR       SAVE ADDRESS OF
0446  15503 077311          STB EQDSK         DISC EQPMT TABLE
0447  15504 027440          JMP CNFG2
0448*                                       CONTINUE
0449  15505 060154 CNFG7    LDA BDSC0       *
0450*
0451  15506 001722          ALF,RAL         *
0452  15507 013767          AND M.740       * SET RUN TIME
0453  15510 033774          IOR RUND1       *   DISC CHANNEL
0454  15511 001727          ALF,ALF         *   IN *DISC0*
0455  15512 001723          ALF,RAR         *
0456  15513 070154          STA BDSC0       *
0457*
0458  15514 063763          LDA SWP1        *
0459  15515 067302          LDB GEND1       * SET NEW ENTRIES
0460  15516 017606          JSB INSWP       *   IN SYS GEN. TIME
0461  15517 063764          LDA SWP2        *   DISC CHANNEL
```

[SLIDE 71B]

DOS-M DISC RESIDENT BOOTSTRAP


```
0462  15520 067303          LDB GEND2      *   INTERRUPT TABLE LOCATIONS
0463  15521 017606          JSB INSWP      *
0464*
0465  15522 002400          CLA            *
0466  15523 067774          LDB RUND1      * SET RUN TIME
0467  15524 017606          JSB INSWP      *   DISC CHANNELS
0468  15525 063311          LDA EQDSK      *     IN
0469  15526 067775          LDB RUND2      *   INTERRUPT TABLE
0470  15527 017606          JSB INSWP      *
0471*
0472  15530 067763          LDB SWP1       *
0473  15531 006003          SZB,RSS        *
0474  15532 027540          JMP SWAP2      * SET NEW I/O CHANNELS
0475  15533 044055          ADB .2         *
0476  15534 160001          LDA B,I        *   IN EQUIPMENT TABLE ENTRIES
0477  15535 023774          XOR RUND1      *
0478  15536 033302          IOR GEND1      *   OF DEVICES
0479  15537 170001          STA B,I        *
0480  15540 067764 SWAP2    LDB SWP2       *   SWAPPED
0481  15541 006003          SZB,RSS        *
0482  15542 027546          JMP SWPSC      *   WITH RUN TIME DISC
0483  15543 023775          XOR RUND2      *
0484  15544 033303          IOR GEND2      *
0485  15545 170001          STA B,I        *
0486*
0487  15546 063776 SWPSC    LDA BDRV#
0488  15547 001200          RAL
0489  15550 067772          LDB BHMSK
0490  15551 006003          SZB,RSS  <-- PACK BOOTSTRAPPED UP ?
0491  15552 002004          INA      <-- YES! SET S.C. ADD...
0492  15553 064155          LDB BSYSC
0493  15554 070155          STA BSYSC      SET RUN TIME SYS SUBCHNL
0494  15555 054175          CPB BUDSC
0495  15556 070175          STA BUDSC
0496  15557 006400          CLB
0497  15560 050175          CPA BUDSC      SYS SC = USER SC?
0498  15561 064115          LDB BSYBF       -YES,
0499  15562 074200          STB BDSCL         SET DISCL
0500  15563 047757          ADB .400
0501  15564 074157          STB BUNTS        = SYSTEM NEXT TRK/SCTR
0502  15565 064175          LDB BUDSC
0503  15566 074161          STB BCDSC
0504*                                        GO START SYSTEM
0505  15567 124003          JMP 3B,I       GO START DOS
0506*
0507  15570 000000 CNFGR    NOP
0508  15571 057730          CPB DVADR      DONE CONFIGURING THIS DRIVER?
0509  15572 127570          JMP CNFGR,I     -YES
0510  15573 037730          ISZ DVADR       -NO, INCR TO NEXT INSTRUCTION
0511  15574 163730          LDA DVADR,I    LOAD INSTRUCTION
0512  15575 002021          SSA,RSS        *
0513  15576 027571          JMP CNFGR+1    * CHECK IF INSTRUCTION
0514  15577 013766          AND MASK       *   IS I/O
0515  15600 002002          SZA            *   FOR DEVICE (EXCLUDING DMA)
0516  15601 027571          JMP CNFGR+1    *
0517  15602 163730          LDA DVADR,I     -YES,
```

DOS-M DISC RESIDENT BOOTSTRAP


```
0518  15603 043773          ADA CHANL       CONFIGURE INSTRUCTION
0519  15604 173730          STA DVADR,I     AND STORE
0520  15505 027571          JMP CNFGR+1
0521*
0522  15606 000000  INSWP   NOP             *
0523  15607 044045          ADB .6N         * THIS SUBROUTINE
0524  15610 044201          ADB BINTB       *   IS USED TO SWAP
0525  15611 170001          STA B,I         *   INTERRUPT TABLE ENTRIES
0526  15612 127606          JMP INSWP,I     *
0527*
0528  15613 000000  PLOAD   NOP                      DISC READ ROUTINE
0529  15614 167762          LDB SPCAD,I     GET LOW CORE ADDRESS
0530  15615 037762          ISZ SPCAD       INCR CURRENT SPBUF ADDRESS
0531  15616 163762          LDA SPCAD,I     GET HIGH CORE ADDRESS
0532  15617 037762          ISZ SPCAD       INCR CURRENT SPBUF ADDRESS
0533  15620 003304          CMA,CCE,INA     COMPLEMENT, SET DIRECTION BIT
0534  15621 040001          ADA B           SET A = TOTAL WORD COUNT
0535  15622 005225          RBL,ERB         SET DIRECTION BIT IN CORE ADDR
0536  15623 106702          CLC 2
0537  15624 106602          OTB 2           SET MEMORY ADDRESS REGISTER
0538  15625 167762          LDB SPCAD,I     GET DISK ADDRESS OF ABSOLUTE COD
0539  15626 037762          ISZ SPCAD       INCR CURRENT SPBUF ADDRESS
0540*
0541  15627 002021  SLOAD   SSA,RSS         SKIP - MORE SECTORS TO LOAD
0542  15630 127613          JMP PLOAD,I     RETURN - THIS SECTION LOADED
0543  15631 043754          ADA P.128       ADJUST FOR NEXT COUNT
0544  15632 073761          STA RECNT       SET REMAINING COUNT
0545  15633 002020          SSA             SKIP - LESS THAN 128 WORDS
0546  15634 002400          CLA
0547  15635 043753          ADA N.128       SET A = CURRENT SECTOR COUNT
0548  15636 102702          STC 2
0549  15637 102602          OTA 2           SET WORD COUNT REGISTER
0550  15640 060001          LDA B           LOAD CURRENT DISK ADRS INTO A,
0551  15641 001767          ALF,CLE,ALF       ROTATE TO LO BITS,
0552  15642 013752          AND M.377         AND ISOLATE TRK#
0553  15643 102600  LSKA    OTA 0           OUTPUT TRK#
0554  15644 103700          STC 0,C           TO DATA CHANNEL
0555  15645 063755          LDA SKCMD       LOAD SEEK COMMAND
0556  15646 033776          IOR BDRV#       INCLUDE DRIVE #
0557  15647 106701          CLC 1
0558  15650 102601          OTA 1           OUTPUT SEEK/ADDRESS CMND
0559  15651 103701          STC 1,C           TO COMMAND CHANNEL
0560  15652 102300          SFS 0           CHECK DATA CHNL FLAG,
0561  15653 027652          JMP *-1           LOOP UNTIL SET
0562  15654 060001          LDA B           LOAD CURRENT DISK ADDRESS INTO A
0563  15655 013752          AND M.377         ISOLATE SECTOR#
0564  15656 043751          ADA #SPTN       ADD NEG #SCTRS/TRK
0565  15657 002021          SSA,RSS         CHECK IF SCTR# > #SCTRS/TRK
0566  15660 033757          IOR .400          -YES, SET LOWER HEAD#
0567  15661 002020          SSA
0568  15662 043747          ADA #SPTK         -NO, ADD #SCTRS/TRK BACK IN
0569  15663 033772          IOR BHMSK       INCLUDE SYS HEAD# MASK
0570  15664 102600          OTA 0           OUTPUT HEAD/SECTOR
0571  15665 103700          STC 0,C           TO DATA CHANNEL
0572  15666 102301          SFS 1           CHECK CMND CHNL FLAG,  > WAIT FOR SEEK
0573  15667 027666          JMP *-1           WAIT UNTIL SET
```

DOS-M DISC RESIDENT BOOTSTRAP


```
0574  15670 017714          JSB BSTAT <-
0575  15671 063756          LDA RDCMD     LOAD READ COMMAND
0576  15672 033776          IOR BDRV#     INCLUDE DRIVE#
0577  15673 102601          OTA 1         OUTPUT COMMAND FOR READ
0578  15674 103700          STC 0,C
0579  15675 106701          CLC 1
0580  15676 103706          STC 6,C       INITIATE DMA
0581  15677 103701          STC 1,C       INITIATE DATA TRANSFER
0582  15700 102301          SFS 1         CHECK CMND CHNL FLAG,
0583  15701 027700          JMP *-1         WAIT UNTIL SET
0584  15702 017714          JSB BSTAT <-
0585  15703 060001          LDA B         LOAD CURRENT DISC ADDRESS INTO A
0586  15704 013752          AND M.377       AND ISOLATE
0587  15705 053750          CPA #SPCY     CHECK IF LAST SECTOR ON CYL
0588  15706 002001          RSS             -YES
0589  15707 002405          CLA,INA,RSS     -NO
0590  15710 063760          LDA #MASK
0591  15711 044000          ADB A         INCR TO NEXT DISC ADDRESS (TRACK)
0592  15712 063761          LDA RECNT     GET REMAINING COUNT
0593  15713 027627          JMP SLOAD     LOAD NEXT SECTOR
0594*
0595  15714 000000  BSTAT   NOP
0596  15715 103700          STC 0,C
0597  15716 063776          LDA BDRV#
0598  15717 106701          CLC 1         OUTPUT STATUS COMMAND
0599  15720 102601          OTA 1
0600  15721 103701          STC 1,C
0601  15722 102300          SFS 0
0602  15723 027722          JMP *-1
0603  15724 102500  DSKY    LIA 0 <- GET STATUS
0604  15725 000010          SLA
0605  15726 102031          HLT 31B
0606  15727 127714          JMP BSTAT,I
0607*
0608*
0609  15730 015642  DVADR   DEF DSKA-1
0610  15731 015724  DEFDY   DEF DSKY
0611*
0612  15732 015733  ASPBF   DEF *+1
0613  15733 000000          BSS 12
0614*
0615  15747 000000  #SPTK   NOP           #SCTR/TRK (physical)
0616  15750 000000  #SPCY   NOP           #SCTR/CYL - 1
0617  15751 000000  #SPTN   NOP           NEG # SCTRS/TRK (physical)
0618  15752 000377  M.377   OCT 377
0619  15753 177600  N.128   DEC -128
0620  15754 000200  P.128   DEC 128
0621  15755 030000  SKCMD   OCT 030000    SEEK COMMAND
0622  15756 020000  RDCMD   OCT 020000    READ COMMAND
0623  15757 000400  .400    OCT 400       LOWER HEAD# BIT
0624  15760 000351  #MASK   OCT 351       INCR. TRK# MASK
0625  15761 000000  RECNT   OCT 0         CURRENT REMAINING COUNT
0626  15762 000000  SPCAD   OCT 0         CURRENT DISK SPEC. BUFFER ADDR
0627*
0628  15300         SPBF    EQU 153008
0629  15302         GEND1   EQU SPBF+2
```

STATUS SUBRTN.

DATA AREA

CONTAINS 4 Entries (3 words each) for loading CRS.

[SLIDE 71E]

DOS-M DISC RESIDENT BOOTSTRAP


```
0630   15303           GEND2 EQU SPBF+3
0631   15304           EGCUR EQU SPBF+4
0632   15305           EGPCD EQU SPBF+5
0633   15310           CNTR  EQU SPBF+8
0634   15311           EGDSK EQU SPBF+9
0635*
0636   00053           #     EQU 53B
0637   00045           .6N   EQU #-6
0638   00051           .2N   EQU #-2
0639   00052           .1N   EQU #-1
0640   00055           .2    EQU #+2
0641   00056           .3    EQU #+3
0642   00058           .5    EQU #+5
0643   00066           .17D  EQU #+11
0644   00071           M.37  EQU #+14
0645   00072           M.77  EQU #+15
0646*
0647   00100           @     EQU 100B
0648   00115           BSYBF EQU @+13
0649   00117           BEQIB EQU @+15
0650   00122           BEQT# EQU @+16
0651   00154           BDSCU EQU @+44
0652   00155           BSYSC EQU @+45
0653   00157           BUNTS EQU @+47
0654   00160           BSNTS EQU @+48
0655   00161           BCDSC EQU @+49
0656   00175           BUDSC EQU @+61
0657   00200           BDSCL EQU @+64
0658   00201           BINTB EQU @+65
0659*
0660   15763  000000    SWP1  NOP
0661   15764  000000    SWP2  NOP
0662   15765  000031    DISK  OCT 31
0663   15766  070036    MASK  OCT 070036
0664   15767  177740    M.740 OCT 177740
0665*
0666   15777           END   EQU 15777B
0667   15772           BFMSK EQU END-5
0668   15773           CHANL EQU END-4
0669   15774           RUND1 EQU END-3
0670   15775           RUND2 EQU END-2
0671   15776           BDRV# EQU END-1
0672*
0673   15777                 ORG 15777B
0674   15777  177733         ABS ASPBF-*
0675*
0676                         END DSGEN
**  NO ERRORS*
```

BASE PAGE COMMUNICATION AREA LOCATIONS

[SLIDE 71F]

HALTS IN DOS-M DURING SYSTEM OPERATION

T-REGISTER CONTENTS	PROGRAM LOCATION	CAUSE OF HALT	RECOVERY ACTION
102000	$EX18	System was unable to use Interrupt Table to match channel # in Equipment Table for given I/O request.	Check Interrupt Table entries and patch if possible. Regenerate correct system. Irrecoverable halt.
102004	DISCM	Power UP or DOWN with DOS-M System in core with P.F. option present	Bootstrap System back up from Disc and Restart.
102011	$EX20	Disc Parity Error. Halt occurs after printing messages on System TTY to inform operator where error occurred. (TRACK #, SECTOR #, and Sub-channel #).	Turn ON "Disc Protect Override Switch" and press "RUN" for System to assign next spare track.
102077	$EX20	Follows message telling operator to turn OFF "Disc Protect Override Switch" after spare track assignment.	Turn OFF "Disc Protect Override Switch" and press "Run". System aborts job that was running.
102031	DVR31	Trying to write on cylinder that has been flagged protected with "Dics Protect Override Switch" OFF.	Press "Run" to exit driver with no action taken on Disc.

DOS-M FLOWCHARTS TABLE OF CONTENTS

DISC MONITOR

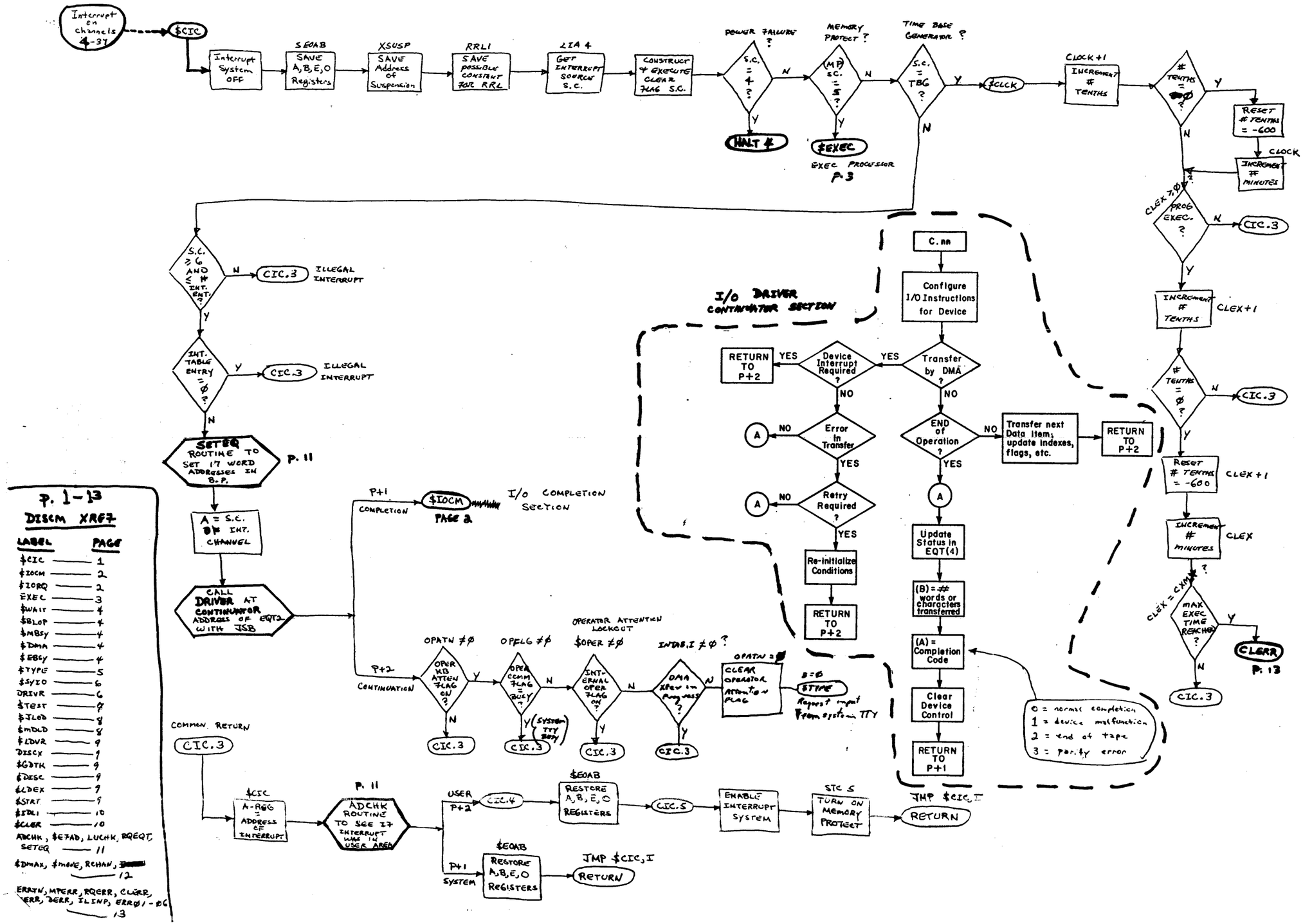
EXECUTIVE MODULES

EXECUTIVE MODULES (continued)

SYSTEM SUBROUTINES

DRIVERS

CENTRAL INTERRUPT CONTROL PROCESSOR ($CIC)



P. 1-13

DISCM XREF

LABEL	PAGE
$CIC	1
$IOCM	2
$IORQ	2
EXEC	3
$WAIT	4
$BLOP	4
$MBSY	4
$DMA	4
$EBSY	4
$TYPE	5
$SYIO	6
DRIVR	6
$TEST	7
$JLOD	8
$MDLD	8
$LDVR	9
DISCX	9
$GDTK	9
$DESC	9
$LDEX	9
$STRT	9
$IDLI	10
$CLER	10
ADCHK, $E7AD, LUCHK, RQEQT, SETEQ	11
$DMAX, $MOVE, RCHAN, [redacted]	12
ERATN, MPERR, RQERR, CLERR, TERR, BERR, ILINP, ERRØ1-ØC	13

INPUT/OUTPUT COMPLETION SECTION ($IOCM)

ENTERED FROM $EXEC WITH
A = STATUS FROM DRIVER
B = # CHAR./WORDS XMITTED

$IOCM

SAVE
EQT13,I ← A
CHAR#1 ← A
EQT14,I ← B

RCHAN
RECTINE TO RELEASE DMA CHAN
C,T,E. ← B
P. 12

IOCDMD = EQT1 ?
Y → SET IOCDMD = EQT1 (+ FOR AVAIL.)
N

LSB SWAP ≠ Ø
JOB PROG. IN CORE ?
Y → IOCØ4
N

NO PROGRAM RUNNING IF EXPG = Ø
DIR. ENTRY FOR CUR. PROG. (WORD 1 = Ø ?)
Y → IOCØ4
N

EQT13,I = 1
R.Q. (PCB) = ENPUT ?
Y → ICCØ4
N

CONVERT BATCH L.U. TO EQT TABLE ADDRESS

BATCH DEVICE = CURRENT DEVICE ?
THIS ADDR. = EQT1 ?
Y → 1ST CHAR. OF REQ. BUF. = : ?
Y → P. 11 ADCHK ROUTINE TO SEE IF REQUESTED BUFF IS Sys/USER
N → IOCØ4
1ST CHAR. OF REQ. BUF. = : ?
N → IOCØ4

Pt2 USER → MOVE ENPUT DATA FROM USR BUFFER TO JOB ENPUT BUF. (MOVE) P. 12
Pt1 SYSTEM → IOCØR

CONVERT XMISSION LOG TO CHAR. IF WORD REQ. + SAVE IN CHARC

SET AV FIELD IN EQT TABLE = Ø#

SAVE REQ. RETURN ADDRESS AS SUSP. ADDR
XSUSP ← EQT9,I

ILINP → ILLEGAL USER INPUT DATA
P. 13

1. JOBPR IN CORE
2. NO PROGRAM RUNNING
3. Request ≠ 1
4. Batch device ≠ current device
5. 1st character ≠ :
6. Return is to system

IOCØ4

EQT13,I ≠ Ø
DID MALFUNCTION OCCUR ?
Y → IOC1Ø
N

EQT4,I
SET AV FIELD IN EQT TABLE = Ø#

CONTROL WORD BIT OFF ?
WAIT TILL DONE REQ. ?
Y → IOC2Ø
N I/O WITHOUT WAIT

TEMP2
SAVE REQUESTED RETURN ADDRESS

P. 11
ADCHK ROUTINE TO SEE IF REQ. RETURN IS EXEC/Sys.

Pt1 SYSTEM → ICC48
EQT4,I
A = DRIVER STATUS
RETURN REQ.
TEMP2,I
IS = TLCC

Pt2 USER

PARITY ERROR ?
= 3 ?
Y → IOC1Ø
N → IEAR
I/O ERROR
P. 13

PARITY = "PA"
PARITY FLAG = "PA" ?
Y → ICC68
EQT1 → SLEQT
SAVE EQT TABLE pointer to return ← in : GO
$WAIT
P. 4.
N

OPATN ≠ Ø ?
OPER ATTEN FLAG ON ?
N → ICC47
Y

SYSTEM TTY BUSY
WORD 4 BIT 15 OF EQT = Ø ?
Y → ICC47
N

OPERATOR ATTEN REQ'T CODE
ENT. KEYBOARD OPER ATTEN FLAG ON ?
$OPER ?
Y → ICC4G
OPATN = Ø
CLEAR OPER ATTEN FLAG
EQT4,I
A = DRIVER STATUS
B = XMISSION LOG
ENTERR,RT SYSTEM CN
ST7 Ø
MEMORY PROTECT ON
ST6 5
TEMP3,I
RETURN REQ.
from EXEC C211
N

SET RETURN A & B REGISTER VALUES
XA = DRIVER STATUS
XB = XMISSION LOG

SET REQ RETURN ADDR AS SUSP ADDR
X SUSP ← EQT7,I

CLEAR OPER. ATTEN. FLAG
OPATN = Ø
B = Ø
$TYPE
P. 5
REQUEST INPUT FROM SYSTEM TTY

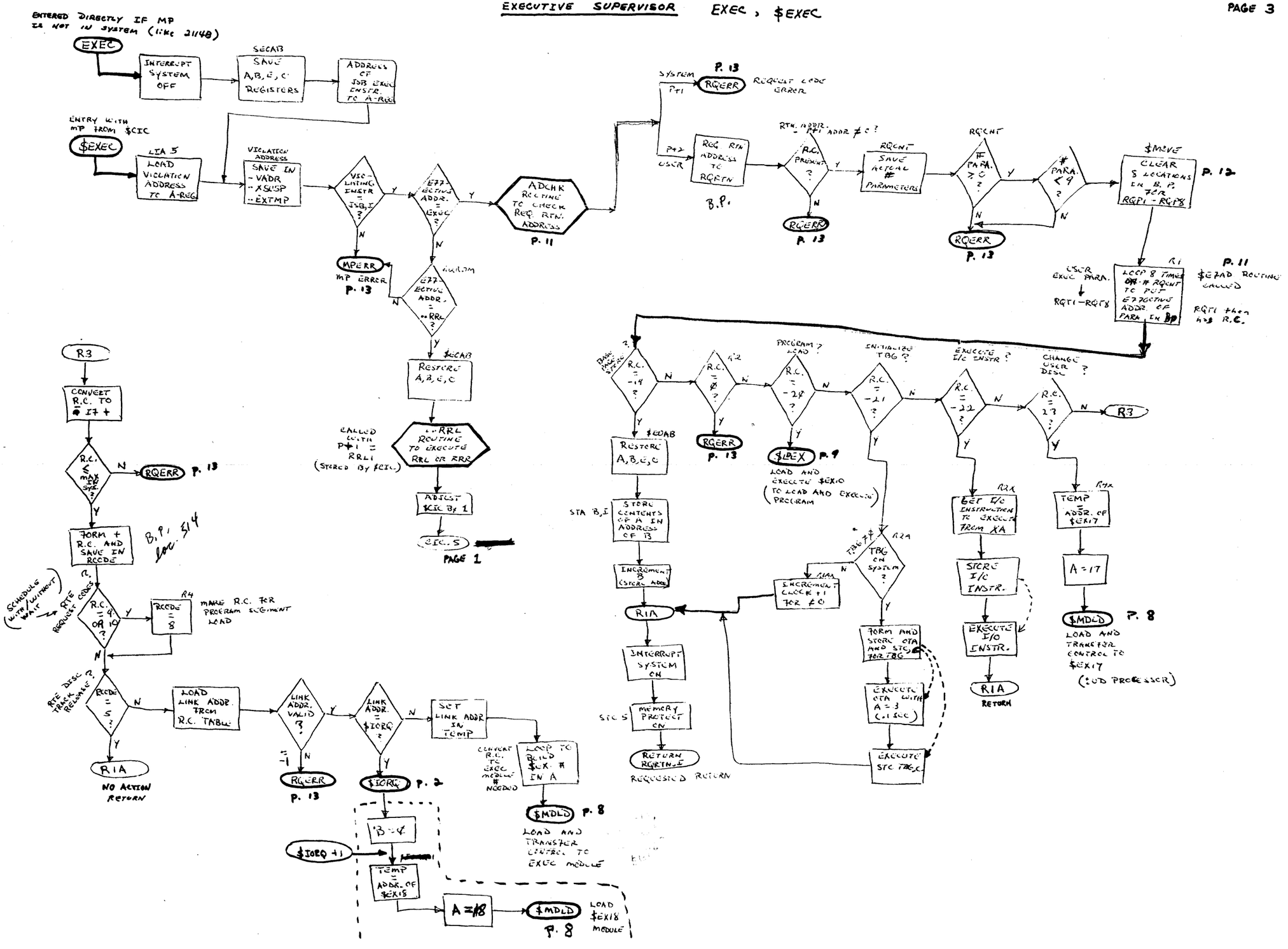
$IORQ
B = Ø
$IORQ+1
TEMP = DCB18
A = 18
$MDLD
P. 8
LOAD $EXIØ (I/O INITIATION PROCESSER)

IOC1Ø
PARITY ERROR ?
= 3 ?
Y → DISK ?
Y → DERR
P. 13
SPECIAL ERROR
N → IERR
A = 3
I/O ERROR
P. 13

IOC2Ø
MUST BE I/O DRIVER WAITING ?
Y → I/O DRIVER AREA BUSY ?
IOCDMD FLAG < Ø ?
N → WAIØ
P. 4
GO LOAD EN MODULE
Y

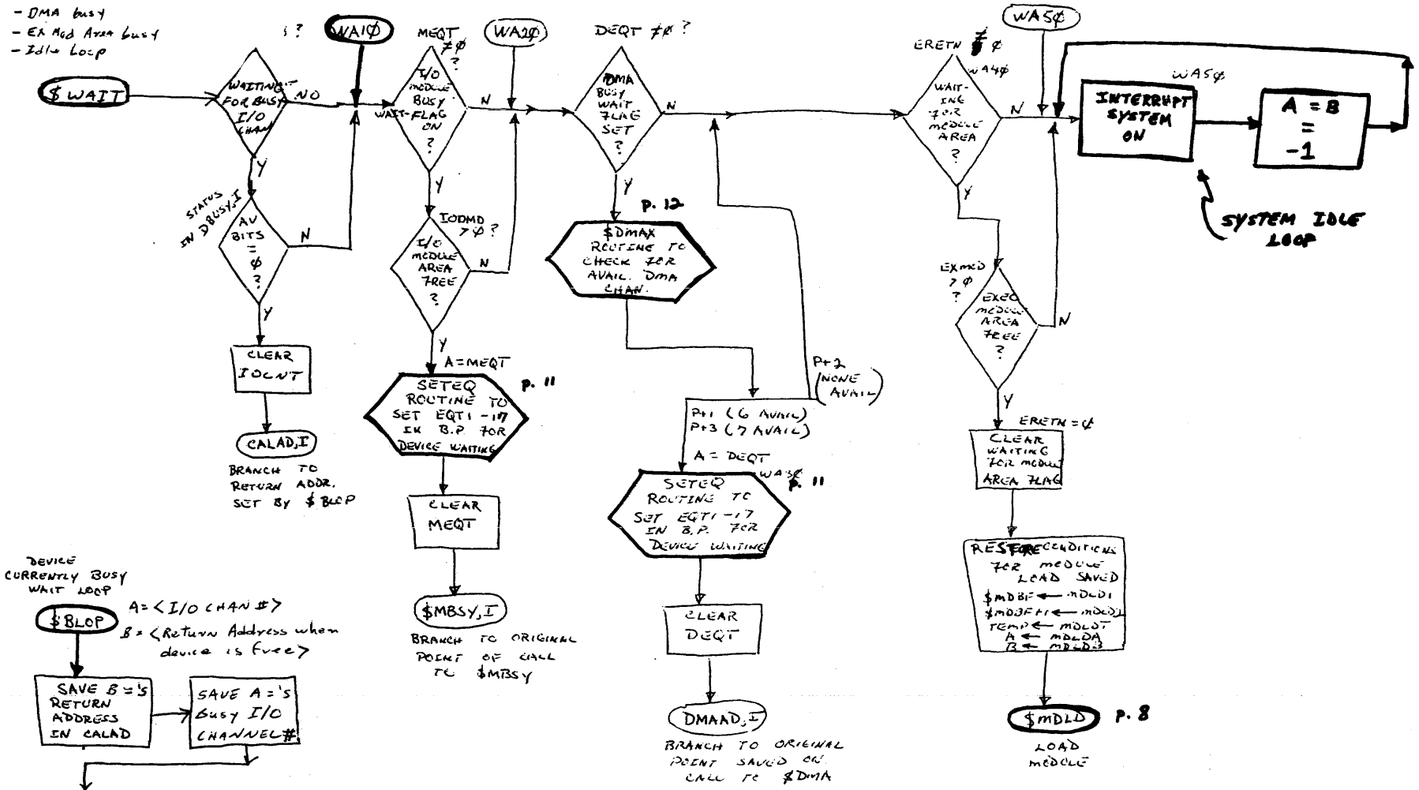
ICC3W
COMPLETE CHANNEL = WAITING REQUEST CODE ?
Y → $WAIT
P. 4
TRANS BACK TO CT
N

ICC4Ø
DCB18 = Ø ?
DMA WAITING FOR DMA ?
Y → $DMAX
P. 12
ROUTINE TO SEE IF DMA CHAN. AVAIL
Pt2 BUSY → WAIØ P. 4
Pt1 C AVAIL
Pt3 7 AVAIL
N → CIC3
Page 1

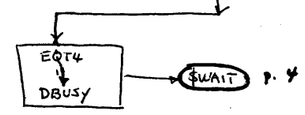

ENTERED DIRECTLY IF MP IS NOT IN SYSTEM (like 21148)
EXEC
INTERCEPT SYSTEM OFF
SECAB
SAVE A,B,E,C REGISTERS
ADDRESS OF JSB EXEC ENSTR. TO A-REG
ENTRY WITH MP FROM $CIC
$EXEC
LIA 5
LOAD VIOLATION ADDRESS TO A-REG
VIOLATION ADDRESS
SAVE IN -VADR -XSUSP -EXTMP
VIOLATING ENTER = JSB,I ?
Y
N
MPERR
MP ERROR
P. 13
EFFECTIVE ADDR. = EXEC ?
Y
N
EFFECTIVE ADDR. = -RRL ?
N
Y
ADCHR
ROUTINE TO CHECK REQ. RTN. ADDRESS
P. 11
$ECAB
RESTORE A,B,E,C
CALLED WITH P+1 = RRLI (STORED BY $CIC)
-RRL ROUTINE TO EXECUTE RRL OR RRR
ADJST $CIC BY 1
$CIC, 5
PAGE 1
SYSTEM P+1
RQERR
REQUEST CODE ERROR
P. 13
P+2 USER
REQ. RTN ADDRESS TO RQRTN
B.P.
RTN. ADDR. - PRT ADDR ≠ 0 ?
R.C. PRESENT ?
Y
N
RQERR
P. 13
RQCNT
SAVE ACTUAL # PARAMETERS
RQCNT
PARA. ≠ 0 ?
Y
N
PARA. < 9 ?
Y
N
RQERR
P. 13
$MOVE
CLEAR $ LOCATIONS IN B.P. FOR RQP1 - RQP8
P. 12
R1
LOOP 8 TIMES OR # RQCNT TO PUT EFFECTIVE ADDR. OF PARA IN BP
P. 11
$EZAD ROUTINE CALLED
USER EXEC PARA.
RQP1 - RQP8
RQP1 then has R.C.
R3
CONVERT R.C. TO @17+
R.C. ≤ MAX RQ SVC ?
N
RQERR
P. 13
Y
FORM + R.C. AND SAVE IN RCODE
B, P, $14 loc.
SCHEDULE (WITH/WITHOUT WAIT
RTE REQUEST CODES
R.C. = 0 OR 10 ?
Y
RCODE = 8
MAKE R.C. FOR PROGRAM SEGMENT LOAD
N
RTE DESC. TRACK/RELEASE ?
RCODE = 5 ?
N
Y
RIA
NO ACTION RETURN
LOAD LINK ADDR. FROM R.C. TABLE
LINK ADDR. VALID ?
Y
N
-1
RQERR
P. 13
LINK ADDR. = $IORQ ?
N
Y
$IORQ
P. 2
SET LINK ADDR IN TEMP
CONVERT R.C. TO EXEC MODULE # NEEDED
LOOP TO BUILD $CX.# IN A
$MDLD
P. 8
LOAD AND TRANSFER CONTROL TO EXEC MODULE
B=4
$IOEQ+1
TEMP = ADDR. OF $EXIS
A=18
$MDLD
P. 8
LOAD $EXIS MODULE
PASS PASSWORD ?
R.C. = -19 ?
N
Y
$EDAB
RESTORE A,B,E,C
STA B,I
STORE CONTENTS OF A IN ADDRESS OF B
INCREMENT B (STORE ADDR)
RIA
INTERRUPT SYSTEM ON
STC 5
MEMORY PROTECT ON
RETURN RQRTN,I
REQUESTED RETURN
R2
R.C. = 0 ?
N
Y
RQERR
P. 13
PROGRAM ? LOAD
R.C. = -28 ?
N
Y
$LDEX
P. 9
LOAD AND EXECUTE $EXIO (TO LOAD AND EXECUTE PROGRAM)
INITIALIZE TBG ?
R.C. = -21 ?
N
Y
TBG ?
R29
TBG ON SYSTEM ?
N
Y
R29A
INCREMENT CLOCK +1 FOR TBG
FORM AND STORE OTA AND STC FOR TBG
EXECUTE OTA WITH A = 3 (.1 SEC)
EXECUTE STC TBG,C
EXECUTE I/C ENSTR. ?
R.C. = -22 ?
N
Y
R2X
GET I/C INSTRUCTION TO EXECUTE FROM XA
STORE I/C INSTR.
EXECUTE I/O ENSTR.
RIA
RETURN
CHANGE USER DESC ?
R.C. = 23 ?
N
Y
R3
R2X
TEMP = ADDR. OF $EXIT
A = 17
$MDLD
P. 8
LOAD AND TRANSFER CONTROL TO $EXIT
(I/O PROCESSOR)

$WAIT, $BLOP, $MBSY, $DMA, $EBSY

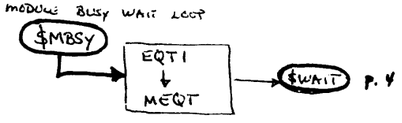
System Idling
- Busy I/O CHAN
- I/O Driver Module Busy
- DMA busy
- EX Mod Area Busy
- Idle Loop



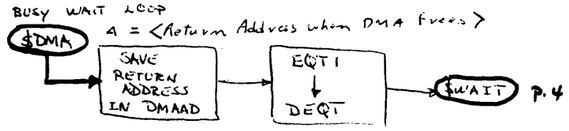
DEVICE CURRENTLY BUSY WAIT LOOP



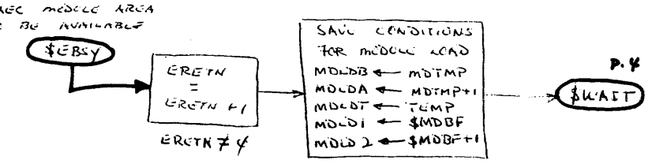
I/O MODULE BUSY WAIT LOOP



DMA BUSY WAIT LOOP



WAIT FOR EXEC MODULE AREA TO BE AVAILABLE



$TYPE

REQUEST INPUT FROM SYSTEM TTY

$TYPE

B VALUE ON ENTRY FOR RETURN
= Ø TO RETURN TO POINT OF SUSP.
≠ Ø TO WAIT UNTIL INPUT IN

I7LAG
SAVE B FOR RETURN OR WAIT

$SAVE$
SAVE POINT OF SUSPENSION + REGISTERS
SXA ← XA
SXB ← XB
SXEO ← XEO
SXSUS ← XSUSP

OP7LG ≠ Ø
INCREMENT SYSTEM TTY BUSY FLAG

$TYP7L

I7LAG = ?

Ø
STORE BUF. ADDR. FOR $SYEO WITH CR-L7-* ADDRESS

≠ Ø
STORE BUF. ADDR FOR $SYEC WITH CR-L7-@ ADDRESS

$SYEO ROUTINE TO OUTPUT CR-L7-* OR CR-L7-@ ON SYS TTY

P+5
COMPLETION RETURN
TYØ15

$SAVE$
SAVE POINT OF SUSPENSION + REGISTERS
SXA ← XA
SXB ← XB
SXEO ← XEO
SXSUS ← XSUSP

STORE JOB INPUT BUF ADDRESS (JBUF) FOR $SYEO RTN.

$SYEO ROUTINE TO READ UP TO 72 CHAR. FROM SYS. TTY KB
P. 6
INTO JOB BUF

P+5 OPERATION COMPLETE
P+6 OPERATION INITIATED

SAVE B IN SAVEB
TYØ2Ø

$SAVE$
SAVE POINT OF SUSPENSION + REGISTERS
SXA ← XA
SXB ← XB
SXEO ← XEO
SXSUS ← XSUSP

OP7LG = Ø
CLEAR SYSTEM TTY BUSY FLAG

RESTORE B FROM SAVEB

$TEST ROUTINE TO TEST INPUT
P. 7

P+1 ERROR OR WAITING (NO OTHER RETURNS HERE)

$IDCD
INPUT REQ. CODE CHECK = ?

OTHER

GO OR JO

(REQUEST MORE INPUT)

OPERATION INITIATED RETURN
P+6

TY≠Ø5

I7LAG = ?

TYØ1Ø
= Ø
TYØ1Ø

A = SXSUS

≠ Ø
$WAIT
P. 4

ADCHK ROUTINE TO CHECK RETURN SUSP ADDRESS
P. 11

TYØ12

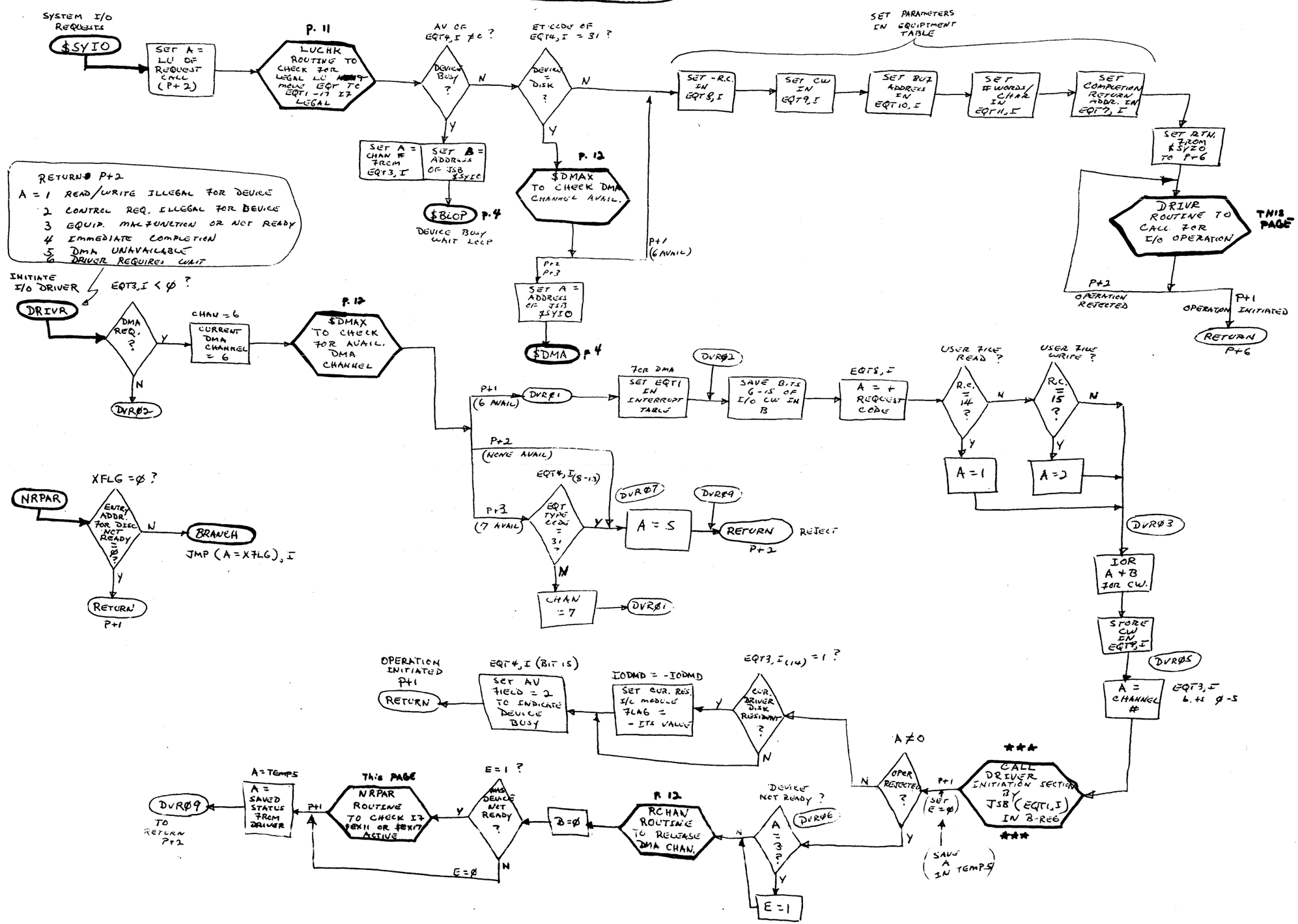
P+2 USER
TYØ12
RESTORE A ← SXA B ← SXB E + O FROM SXEO

INTERRUPT SYSTEM ON

SXSUS,I
RETURN VIA POINT OF SUSPENSION

P+1 SYSTEM
$WAIT
P. 4

$SYIO, DRIVR


SYSTEM I/O REQUESTS
$SYIO
SET A = LU OF REQUEST CALL (P+2)
P. 11
LUCHK ROUTINE TO CHECK FOR LEGAL LU ... NEW EQT TO EQT1,-17 IF LEGAL
AV OF EQT4,I ≠ 0 ?
DEVICE BUSY ?
N
Y
ET CODE OF EQT4,I = 31 ?
DEVICE = DISK ?
N
Y
SET A = CHAN # FROM EQT3,I
SET B = ADDRESS OF JSB $SYIO
$BLOP P.4
DEVICE BUSY WAIT LOOP
P. 12
$DMAX TO CHECK DMA CHANNEL AVAIL.
P+1 (6 AVAIL)
P+2
P+3
SET A = ADDRESS OF JSB $SYIO
$DMA P.4
SET PARAMETERS IN EQUIPMENT TABLE
SET -R.C. IN EQT8,I
SET CW IN EQT9,I
SET BUF ADDRESS IN EQT10,I
SET # WORDS/CHAR IN EQT11,I
SET COMPLETION RETURN ADDR. IN EQT7,I
SET RTN. FROM $SYIO TO P+6
DRIVR ROUTINE TO CALL FOR I/O OPERATION
THIS PAGE
P+2 OPERATION REJECTED
P+1 OPERATION INITIATED
RETURN P+6
RETURNS P+2
A = 1 READ/WRITE ILLEGAL FOR DEVICE
2 CONTROL REQ. ILLEGAL FOR DEVICE
3 EQUIP. MALFUNCTION OR NOT READY
4 IMMEDIATE COMPLETION
5 DMA UNAVAILABLE
6 DRIVER REQUIRES WAIT
INITIATE I/O DRIVER
DRIVR
EQT3,I < 0 ?
DMA REQ. ?
Y
N
DVR02
CHAN = 6
CURRENT DMA CHANNEL = 6
P. 12
$DMAX TO CHECK FOR AVAIL. DMA CHANNEL
P+1 (6 AVAIL)
DVR01
P+2 (NONE AVAIL)
P+3 (7 AVAIL)
EQT4,I(8-13)
EQT TYPE CODE = 31 ?
DVR07
A = 5
DVR09
RETURN P+2
REJECT
CHAN = 7
DVR01
FOR DMA
SET EQT1 IN INTERRUPT TABLE
DVR01
SAVE BITS 6-15 OF I/O CW IN B
EQT5,I
A = + REQUEST CODE
USER FILE READ ?
USER FILE WRITE ?
R.C. = 14 ?
R.C. = 15 ?
A = 1
A = 2
DVR03
IOR A+B FOR CW.
STORE CW IN EQT9,I
DVR05
A = CHANNEL #
EQT3,I bits 0-5
CALL DRIVER INITIATION SECTION BY JSB (EQT1,I) IN B-REG
A ≠ 0
OPER REJECTED ?
N
Y
P+1
(SET E = 0)
(SAVE A IN TEMP5)
DEVICE NOT READY ?
A = 3 ?
DVR06
E = 1
N
Y
B = 0
RCHAN ROUTINE TO RELEASE DMA CHAN.
P. 12
E = 1 ?
THIS DEVICE NOT READY ?
Y
N
E = 0
THIS PAGE
NRPAR ROUTINE TO CHECK IF $EXIT OR $EXIT ACTIVE
P+1
A
A = TEMP5
A = SAVED STATUS FROM DRIVER
DVR09
TO RETURN P+2
EQT4,I (BIT 15)
SET AV FIELD = 2 TO INDICATE DEVICE BUSY
IODMD = -IODMD
SET CUR. RES. I/O MODULE FLAG = - ITS VALUE
EQT3,I(14) = 1 ?
CUR. DRIVER DISK RESIDENT ?
Y
N
OPERATION INITIATED P+1
RETURN
XFLG = 0 ?
NRPAR
ENTRY ADDR. FOR DISC NOT READY = 0 ?
N
Y
BRANCH
JMP (A = XFLG), I
RETURN P+1

$TEST

ENTRY: B = # WORDS INPUT
TO TEST IF DESIRED INPUT HAS BEEN INPUT

$TEST

RETURN TO CALLER I7 NOT CORRECT INPUT

SAVE # WORDS INPUT IN
- $MDB7
- CHARC

SET RTN ADDRESS ($TEST) IN $MDB7+1

JOB INPUT BUF

FIRST CHAR = ?

TE#2

SET A & B = THE FIRST 2 WORDS IN JOB INPUT BUF

DO RRL ROUTINE TO DOUBLE REGISTER SHIFT A AND B TO LEFT

PA7LG

FIRST TWO CHAR (A) TO PAUSE FLAG

OPFLG ≠ ∅

SET SYS TTY BUSY FLG ≠ ∅

$SYIO ROUTINE TO OUTPUT "IGNORED" ON SYS. TTY — p.6

OPERATION INITIATED — P+6

OPERATION COMPLETE — P+5

$IDCD

INPUT REQ. CODE = GO ?

B ≠ ∅

$TYPE — REQUEST INPUT FROM SYS. TTY

ADCHR ROUTINE TO CHECK SAVED SUS. ADDRESS (SXSUS) — p. 11

P+1 (system) — PAGE 1 $CIC8 $CIC.3

P+2 (USER) — TY$12 PAGE 5

CLEAR SYS. TTY BUSY FLAG

CPFLG

SAVE$

SAVE
SXA ← XA
SXB ← XB
SXEO ← XEO
SXSUS ← XSUSP

RETURN P+1

$IDCD

INPUT REQ. CODE = DA ?

INPUT FIRST TWO CHAR = DA ? — DATE ?

TE#2

TE6∅

A = 0

LOAD JOB PROCESSOR

$JLOD — p. 8

ABRT ?

INPT = AB ?

TE9∅

ASSIGN BATCH DEVICE AS SYSTEM TTY — BATCH = SYSTY

B = ∅

ABORT

$ABRT

TEMP+1 = B

EXECUTION TIME CLOCK OFF — CLEX BIT IS ON

$CLER ROUTINE TO CLEAR ALL I/C, RECTANG. DMA, FLAGS... — p. 10

JOBPR IN CORE FLAG = ∅ — SCHAP = ∅

EXEC MODULE REQ. FLAG = ∅ — EXMOD = ∅

TEMP = DE715

RESTORE B WITH TEMP+1

A = 15

$MDLD — p. 8

LOAD EXEC MODULE 15 (ABRT AND PRT MORTEM DUMP PROCESSOR)

PAUSE ?

INPUT = PA ?

$GOPT = XSUSP

$IDCD

INPUT REQ. CODE = GO

$TYPE — p. 5

TYPE ?

INPUT = TY ?

EXPG

DIR. ENTRY FOR CLR PRG. ≠ ∅ ?

BIT 15 OF SWAP = 1

RETURN P+1

TE6∅

OFF ?

INPUT = OF ?

TE9∅

$IDCD

INPUT REQ. CODE = GO ?

TE22

GO ?

INPUT = GO ?

TE25

CLEAR INPUT REQ. CODE

$IDCD = ∅

GO PROCESS POSSIBLE GO PARA.

SVEQT = ∅ ?

TEMP = DE716

A = 16

$MDLD — LOAD $EXIG — p. 8

TE14

SVEQT = B

GO ?

INPUT = GO ?

TY$10 p. 5

SETEQ ROUTINE TO SET I7 WITH EQT ADDRESS IN B — P.11

TEMP2 = EQT7,I

IOC$6 — p. 2

TE22

$IDCD

INPUT REQ. CODE = JO ?

JOB ?

INPUT = JO ?

TE6∅

ETQR ?

INPUT = EJ ?

TE#2

WAITING FOR ANY INPUT ?

$IDCD

INPUT REQ. CODE = ∅ ?

TE#2 WAITING FOR INPUT

BIT 8 OF SWAP = 1 ?

JOB PREC. IN CORE ?

EXPG

DIR. ENTRY FOR CLR. PREC ≠ ∅ ?

TE6X

USER PROG RUNNING

TE25

LU ?

INPUT = LU ?

UP ?

INPUT = UP ?

DN ?

INPUT = DN ?

EQ ?

INPUT = EQ ?

TE#2

TEMP = DE714

A = 14

B = CHARC

$MDLD — p. 8

LOAD EXEC MODULE 14 (:UP, :DN, :LU PROCESSOR)

TEMP = DE709

A = 9

$MDLD — p. 8

LOAD EXEC MODULE 9 (:EQ PROCESSOR)

$JLOD , $MDLD IO.4Ø

give example where would be true for I/O Request

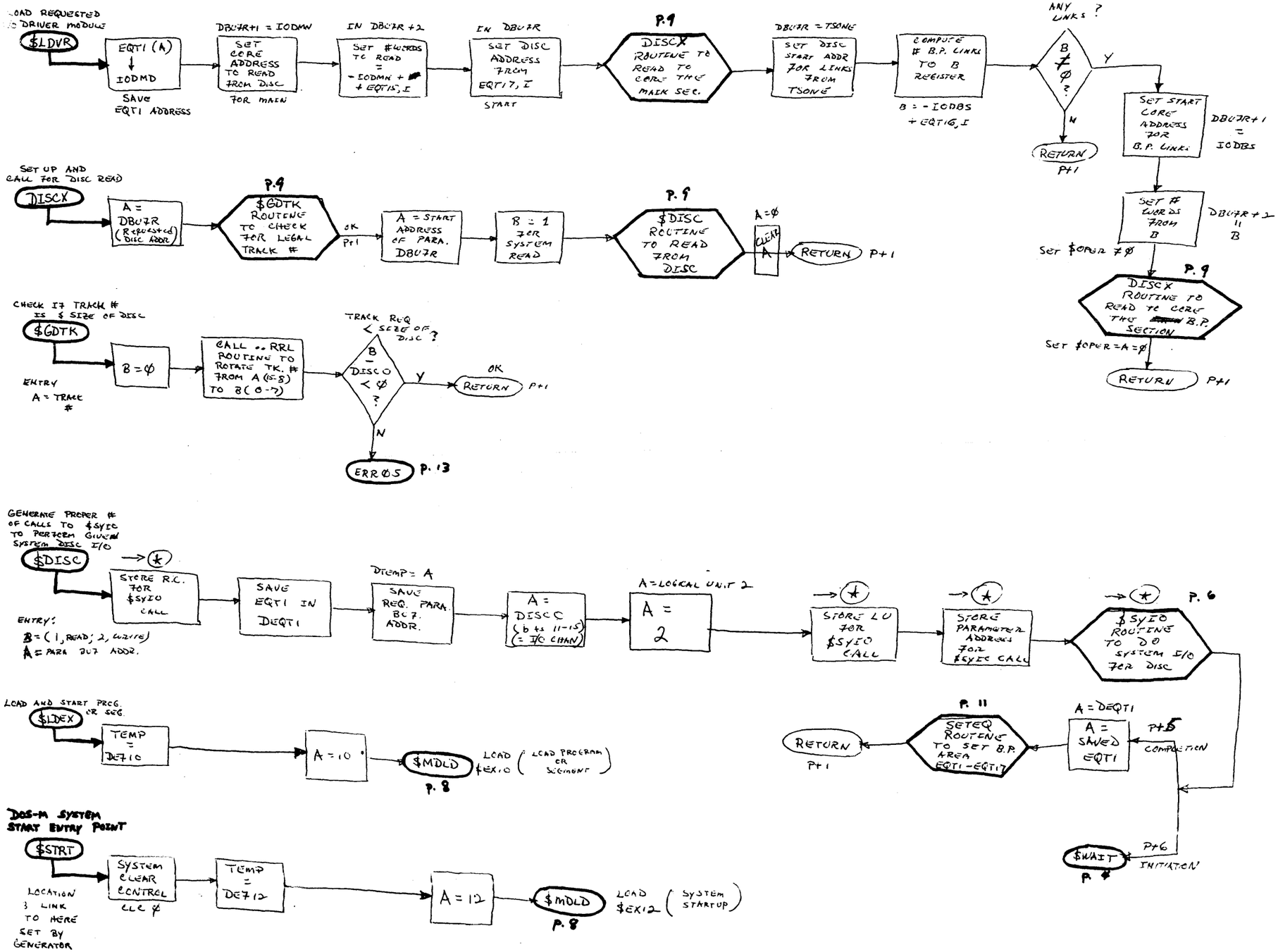
LOAD AND START JOB PROCESSOR
$JLOD
→ $CLER ROUTINE TO CLEAR ALL I/O, RELEASE DMA, CLEAR FLAGS SET P.8
→ SWAP = 1 JOB PROC IN 7LAG ON
→ EXPC, EXPC+1, EXPC+2 SET FIRST 3 WORDS OF CURRENT ENTRY OF DIRECTORY = "JOBPR"
→ SET DISK ADDRESS OF JOB PROCESSOR IN DBU7R
→ SET CORE START ADDRESS IN DBU7R+1
→ SET # WORDS IN DBU7R+2
→ DISCX ROUTINE TO READ MAIN OF JOBPR TO CORE P.9
→ DBU7R = TSCNE SET DISK ADDRESS TO READ IN LINKS
→ DBU7R+1 = UB7WA SET CORE ADDRESS START FOR USER B.P.
→ SET # WORDS IN DBU7R+2 = JOBPB DBU7Z+2
→ CLEAR INPUT REQ. CODE $IDCD = Ø
→ DISCX ROUTINE TO READ B.P. SECTION OF JOBPR TO CORE P.9
→ CLEAR OPER. ATT + SYS. TTY BUSY FLAGS OPATN = Ø, OP7LG = Ø
→ SAVE CHAR. COUNT IN B B = CHARC
→ CLEAR CHAR. COUNT CHARC = Ø
→ INTERRUPT SYSTEM ON
→ MEMORY PROTECT ON STC 5
→ UM7WA,I
BRANCH TO START OF JOB PROCESSOR

EXEC MODULE LOAD
$MDLD
ENTRY
TEMP = TRANSFER ADDRESS
A = MODULE #
B = INFO. NEEDED B/ MODULE
→ SAVE MDTMP+1 ← A, MDTMP ← B
→ DOUBLET START ADDR. → TEMP+3
→ ANY DOUBLETS ? EX# ≠ Ø ?
  Y → EX# TO TEMP+4
  N → TEMP, I BRANCH TO START OF CORE RESIDENT EXEC MODULE (ALL ARE CORE RESIDENT)
→ EXMOD EXEC MODULE FLAG < Ø ?
  Y → $EBSY EXEC MODULE AREA BUSY P. 4
  N → CUR. MODULE = DESIRED MOD. ? EXMOD = MDTMP+1 ?
    Y → MDY
    N → GET NEXT MOD. 1ST WORD ENTRY → SEARCH DOUBLET TABLE
→ ENTRY = DESIRED ? TEMP+3, I = MDTMP+1 ?
  Y → SET EXEC MODULE FLAG = -THIS MOD EXMOD = -THIS MODULE #
    → DBU7R+1 ← EXMAN SET CORE START ADDR. FOR EXEC MODULE IN DBU7R+1
    → (EXMAN+1) - (EXMAN) COMPUTE # WORDS AND STORE IN DBU7R+2
    → SET A = BITS 11-15 OF 1ST ENTRY LET = X
    → PUT 2ND DOUBLET ENTRY IN DBU7R SET DISK ADDRES AND SET $OPER ≠ Ø
    → ANY B.P. WORDS ? A ≠ Ø
      Y → A = A*128
        → B ← A SAVE
        → A = # WORDS IN MAIN -128(X)
        → A < Ø ?
          Y → SET DBU7R+2 = B New # WORDS
          N → DISCX ROUTINE TO READ IN DISK TO CORE P.9
            → DBU7R = TSCNE SET DISK ADDRESS = NEXT LAST REF. +1
            → DBU7R+1 = EXBAS SET CORE ADDR. FOR B.P. LINKS
            → DBU7R+2 = (EXBAS+1) - EXBAS COMPUTE # WORDS AND STORE IN DBU7R+2
            → RETURN TO $EXIB B ≠ Ø P.2
      N → DISCX ROUTINE TO READ IN DISK TO CORE P.9
        ON RETURN SET $OPER = A = Ø
        → RESTORE B TO MDTMP
        → EXMOD = -(MDTMP+1) SET EXEC MOD. FLAG = NEG. THIS MOD. EFFECT IS TO SET EXEC MODULE AREA BUSY
        → EXMAN,I BRANCH DIRECTLY TO EXEC MODULE OVERLAY AREA START
→ TEMP+4 = Ø ? TABLE SEARCH FINISHED ?
  N → INCREMENT TEMP+3 BY 2 FOR NEXT INCREMENT TEMP+4
  Y → TEMP, I MUST BE CORE RESIDENT (BRANCH TO IT)

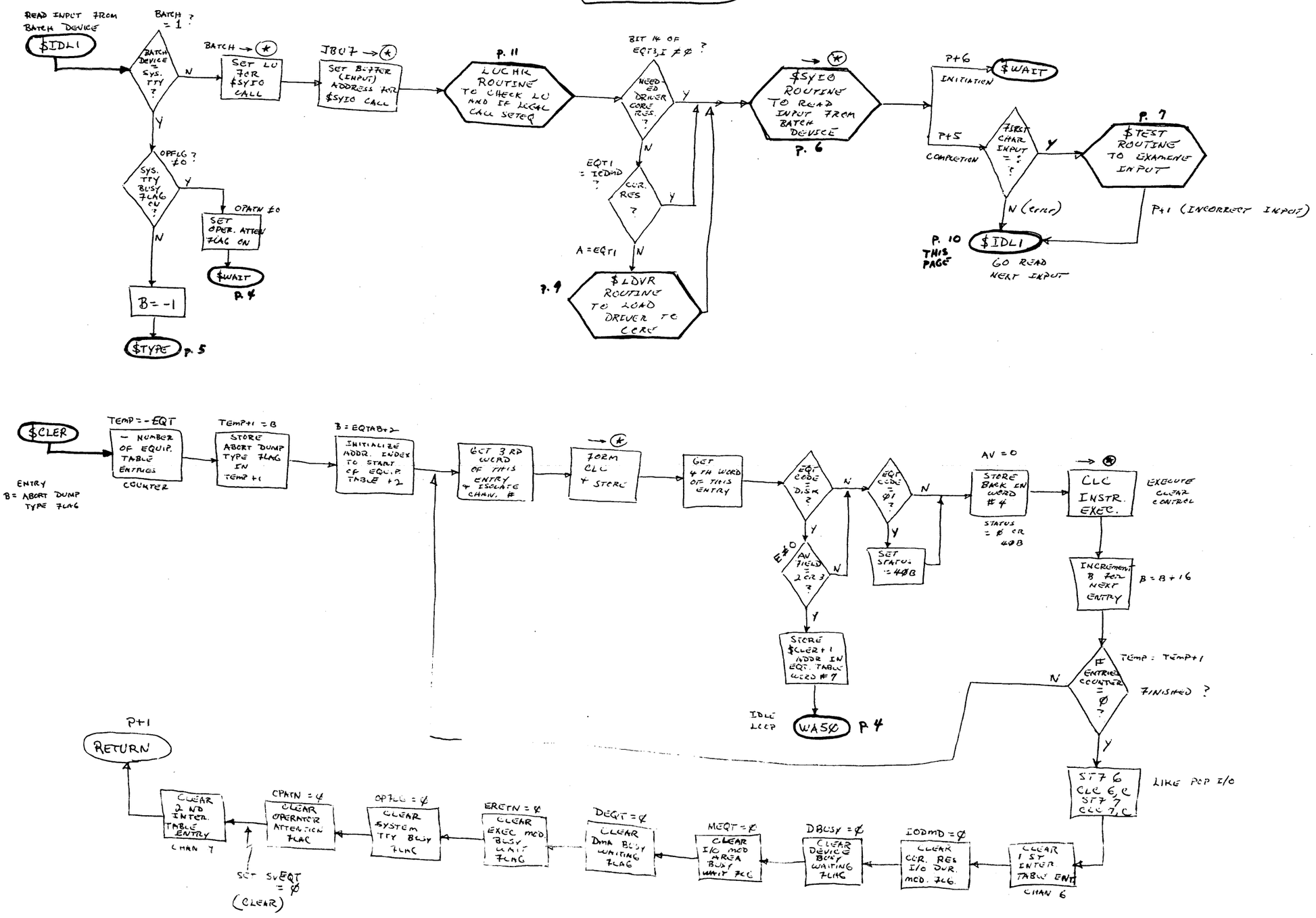
ENTERED FROM $EXIB TO DO A FILE SEARCH
IO.4Ø
→ RCODE = A - 13 To make 14 → 1 (READ), 15 → 2 (WRITE)
→ SET FILE NAME ADDR. IN $MDBF $MDBF = RQPS
→ SET RETURN ADDR. FROM $EXII IN $MDBF+1 $MDBF+1 = IO.FL
→ TEMP = DEF11
→ A = 11
P.8 → $MDLD ROUTINE TO LOAD AND START $EXII

RETURN FROM $EXII
IO.FL
→ B ≠ Ø ?
  → FILE FOUND ?
    Y → RCODE = ?
      1 READ → $IORQ+1 P.2 RETURN TO $EXIB B ≠ Ø
      2 WRITE → GET 3RD WORD OF FILE DIRECTORY
        → FILE TYPE = BD ? TYPE = 1Ø ?
          Y → $IORQ+1 P.2 RETURN TO $EXIB WITH B ≠ Ø
          N → FILE TYPE = AD ? TYPE = 11 ?
            Y → $IORQ+1 P.2 RETURN TO $EXIB WITH B ≠ Ø
            N → ERRTN P. 13 OPERATION REJECT
    N → ERR16 P. 13

($LDVR, DISCX, $GDTK, $DISC) $LDEX, $STRT

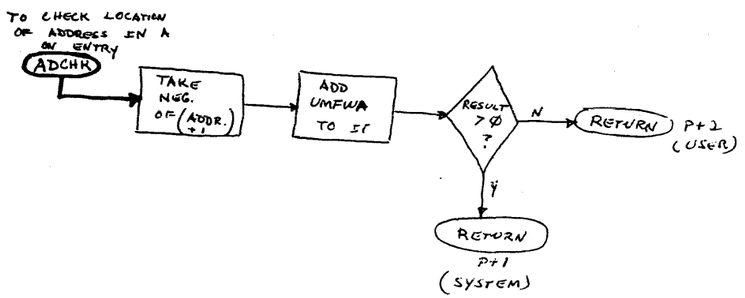


$IDLI, $CLER

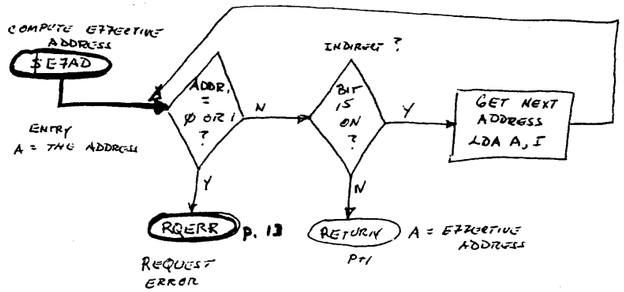

READ INPUT FROM BATCH DEVICE
$IDLI
BATCH ? = 1
BATCH DEVICE = SYS. TTY ?
N
Y
BATCH → ⊛
SET LU FOR $SYIO CALL
JBUF → ⊛
SET BUFFER (INPUT) ADDRESS FOR $SYIO CALL
P. 11
LUCHK ROUTINE TO CHECK LU AND IF LEGAL CALL SETEQ
BIT 14 OF EQT3,I ≠ Ø ?
NEEDED DRIVER CORE RES. ?
Y
N
EQT1 = IDMD ?
CUR. RES ?
Y
A = EQT1
N
P. 9
$LDVR ROUTINE TO LOAD DRIVER TO CORE
⊛
$SYIO ROUTINE TO READ INPUT FROM BATCH DEVICE
P. 6
P+6
INITIATION
$WAIT
P+5
COMPLETION
FIRST CHAR INPUT = ! ?
Y
N (error)
P. 7
$TEST ROUTINE TO EXAMINE INPUT
P+1 (INCORRECT INPUT)
P. 10 THIS PAGE
$IDLI
GO READ NEXT INPUT
OPFLG ? ≠0
SYS. TTY BUSY FLAG ON ?
Y
N
OPATN ≠0
SET OPER. ATTEN FLAG ON
$WAIT
P. 4
B = -1
$TYPE
P. 5
$CLER
ENTRY
B = ABORT DUMP TYPE FLAG
TEMP = -EQT
- NUMBER OF EQUIP. TABLE ENTRIES
COUNTER
TEMP+1 = B
STORE ABORT DUMP TYPE FLAG IN TEMP+1
B = EQTA+2
INITIALIZE ADDR. INDEX TO START OF EQUIP. TABLE +2
GET 3 RD WORD OF THIS ENTRY & ESCAPE CHAN. #
FORM CLC & STORE
GET 4 TH WORD OF THIS ENTRY
EQT CODE = DISK ?
Y
N
AV FIELD = 2 OR 3 ?
Y
N
STORE $CLER+1 ADDR IN EQT. TABLE WORD #7
IDLE LOOP
WAIT
P 4
EQT CODE = 01 ?
Y
N
SET STATUS = 40B
AV = 0
STORE BACK EQT WORD #4
STATUS = Ø OR 40B
CLC INSTR. EXEC.
EXECUTE CLEAR CONTROL
INCREMENT B FOR NEXT ENTRY
B = B + 16
# ENTRIES COUNTER = Ø ?
N
Y
TEMP = TEMP+1
FINISHED ?
ST 7 6
CLC 6,C
ST 7 7
CLC 7,C
LIKE POP I/O
CLEAR 1 ST INTER TABLE ENT
CHAN 6
IODMD = Ø
CLEAR CUR. RES I/O DUR. MOD. FLG.
DBUSY = Ø
CLEAR DEVICE BUSY WAITING FLAG
MEQT = Ø
CLEAR I/O MOD AREA BUSY WAIT FLG
DEQT = Ø
CLEAR DMA BUSY WAITING FLAG
ERETN = Ø
CLEAR EXEC MOD BUSY WAIT FLAG
OPFLG = Ø
CLEAR SYSTEM TTY BUSY FLAG
OPATN = Ø
CLEAR OPERATOR ATTENTION FLAG
CLEAR 2 ND INTER. TABLE ENTRY
CHAN 7
SET SVEQT = Ø (CLEAR)
P+1
RETURN

UTILITY SUBROUTINES

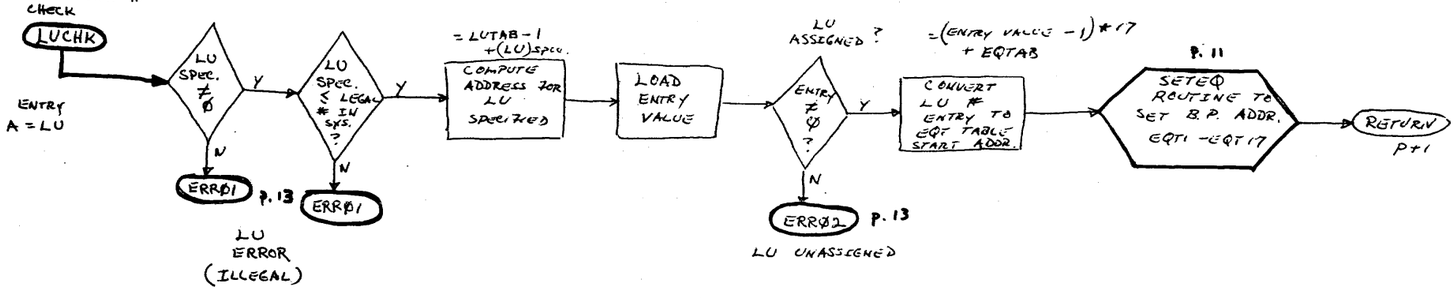
TO CHECK LOCATION OF ADDRESS IN A ON ENTRY



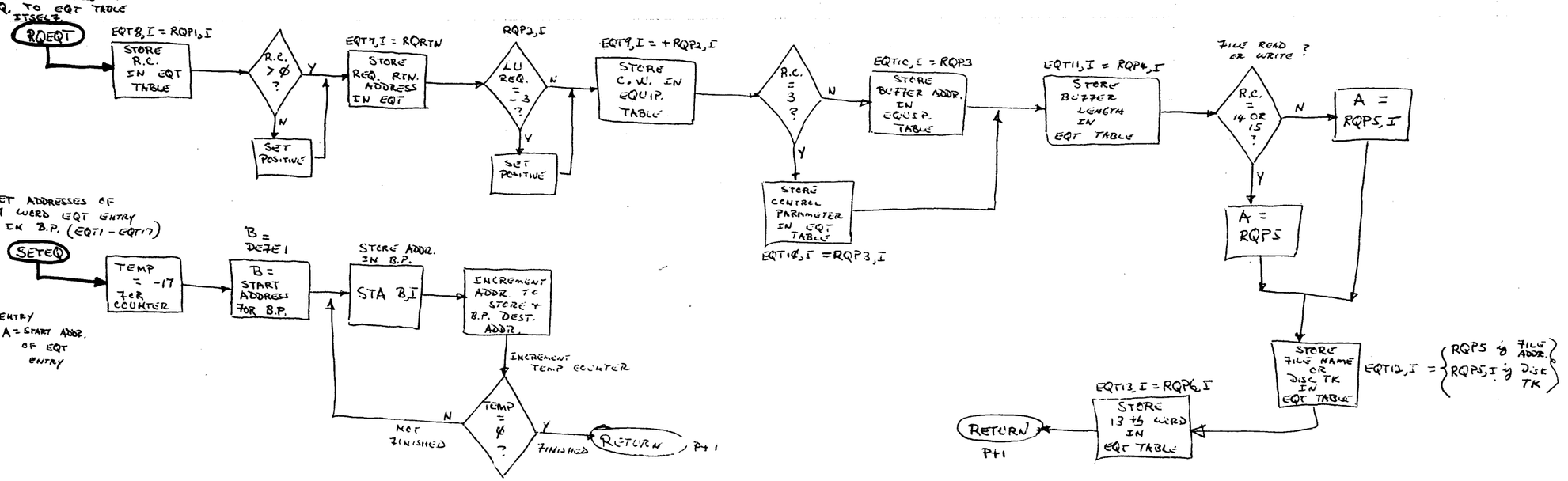
COMPUTE EFFECTIVE ADDRESS



LOGICAL UNIT # CHECK

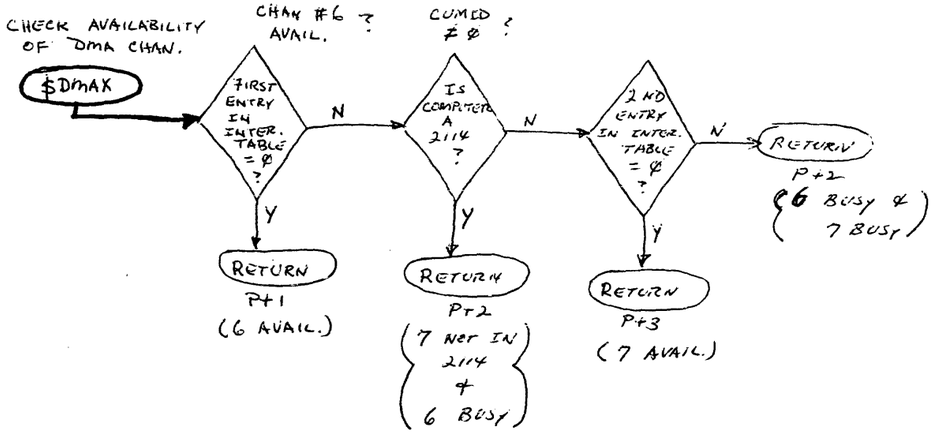


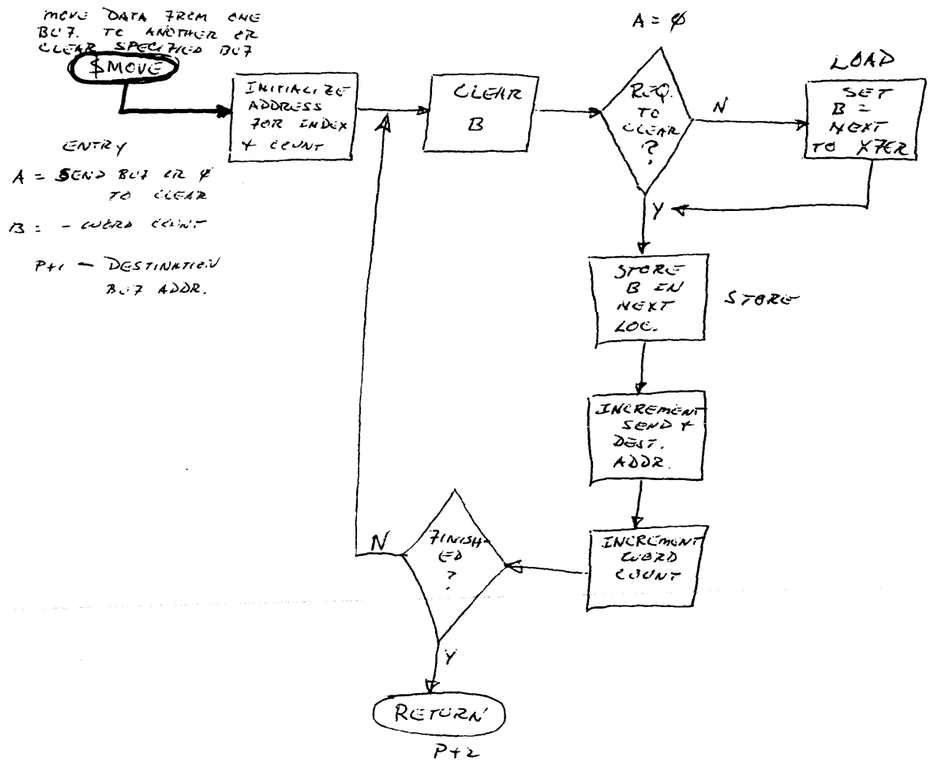
TRANSFER PARAMETERS + RETURN ADDRESS OF REQ. TO EQT TABLE

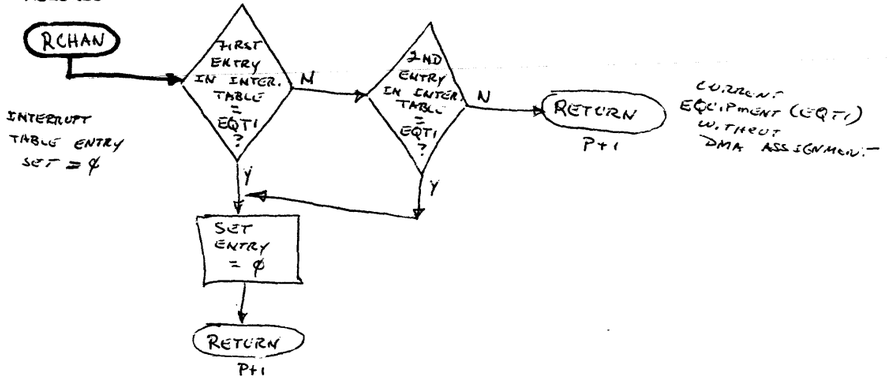


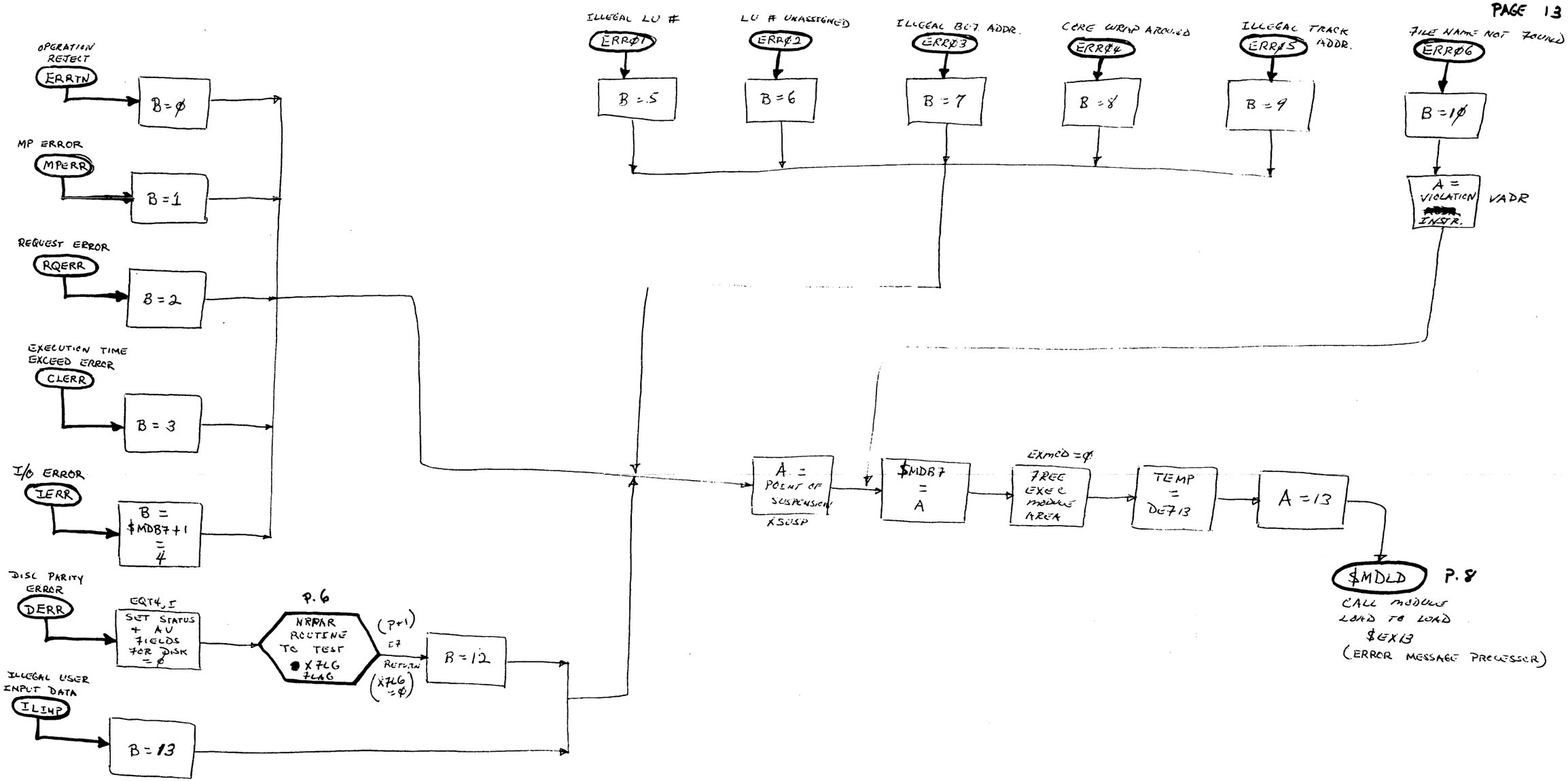
SET ADDRESSES OF 17 WORD EQT ENTRY IN B.P. (EQT1 - EQT17)

UTILITY SUBROUTINES



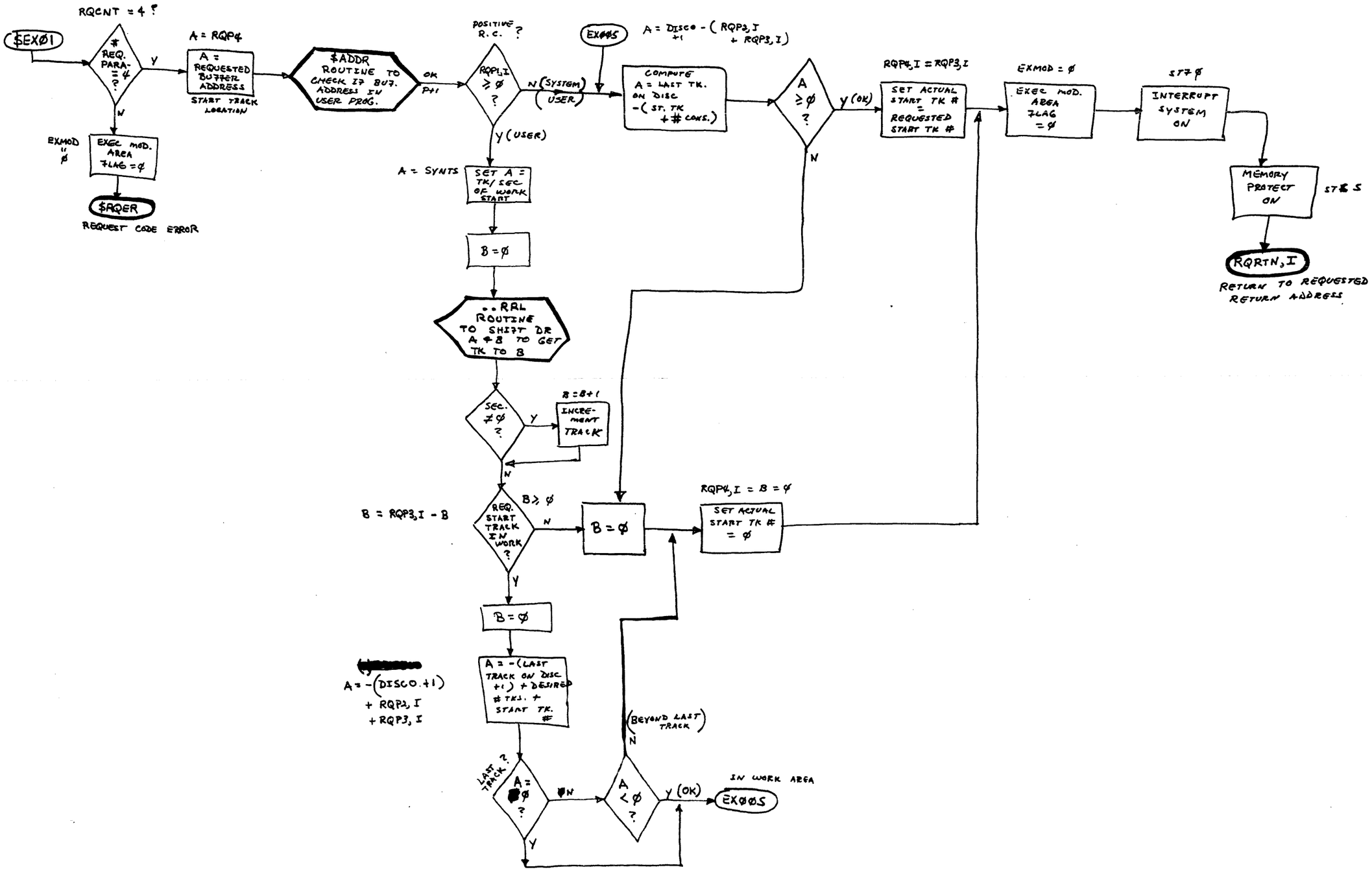




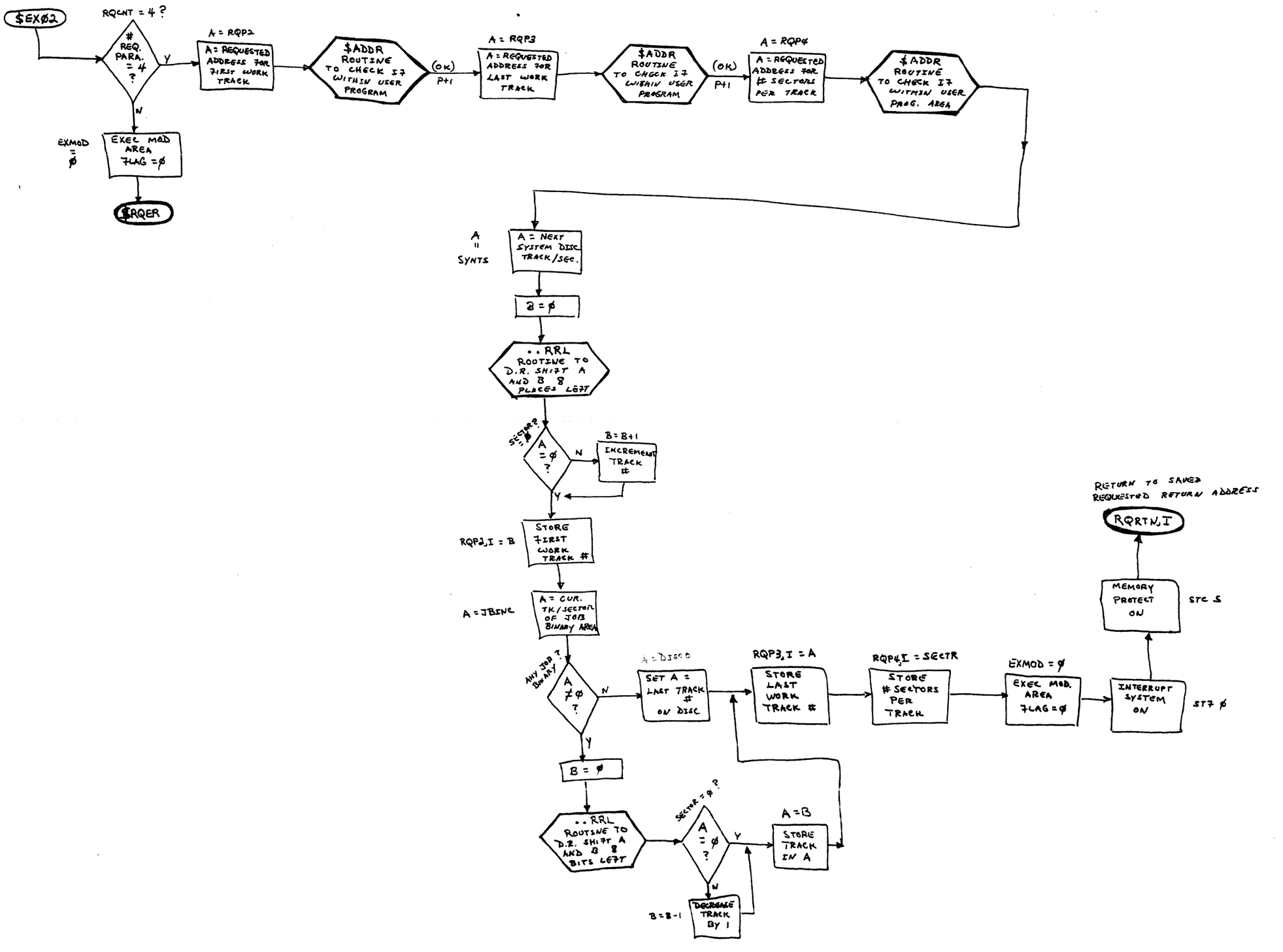


ERROR ENTRY POINTS

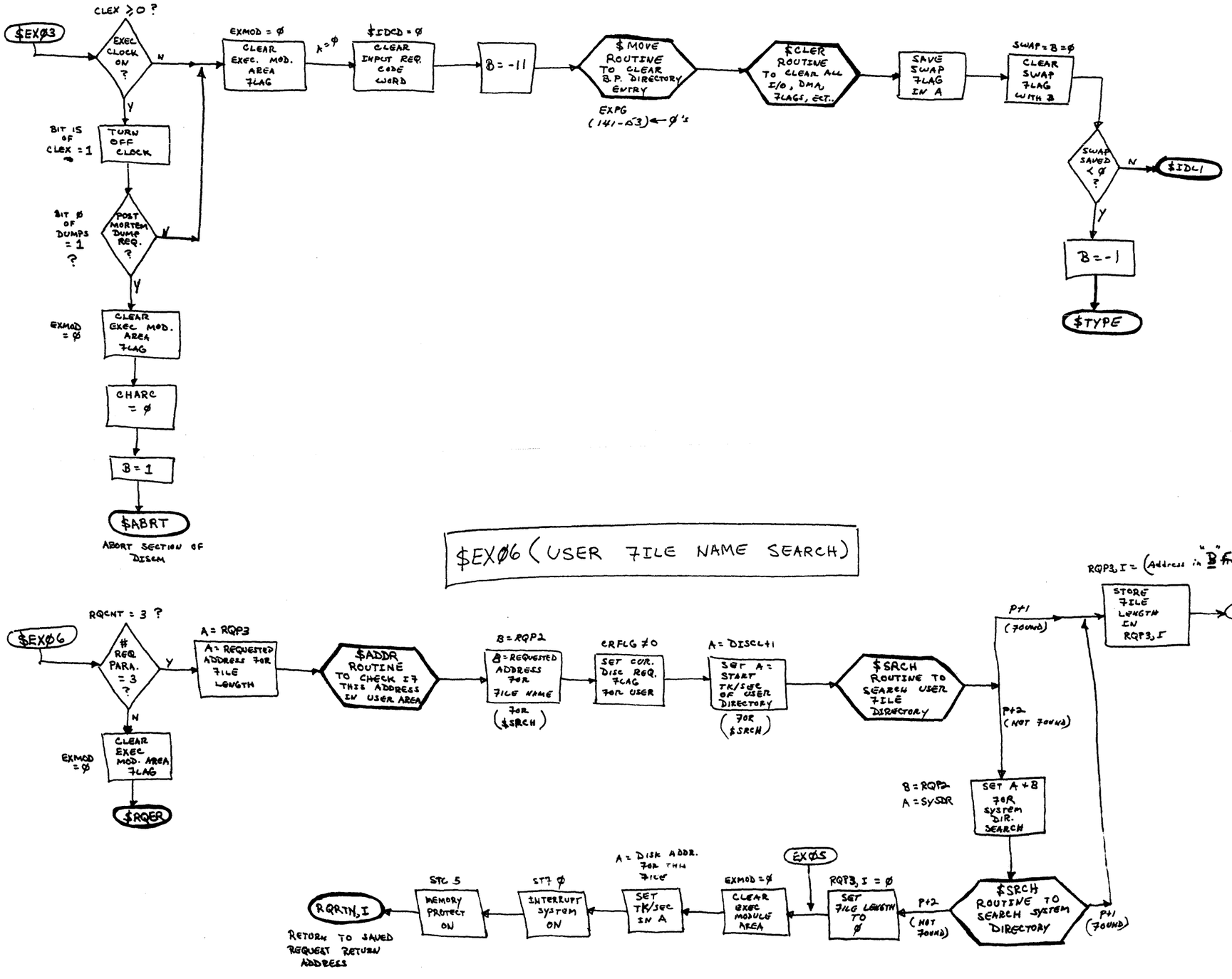
$EXØ1 (DISC TRACK STATUS)



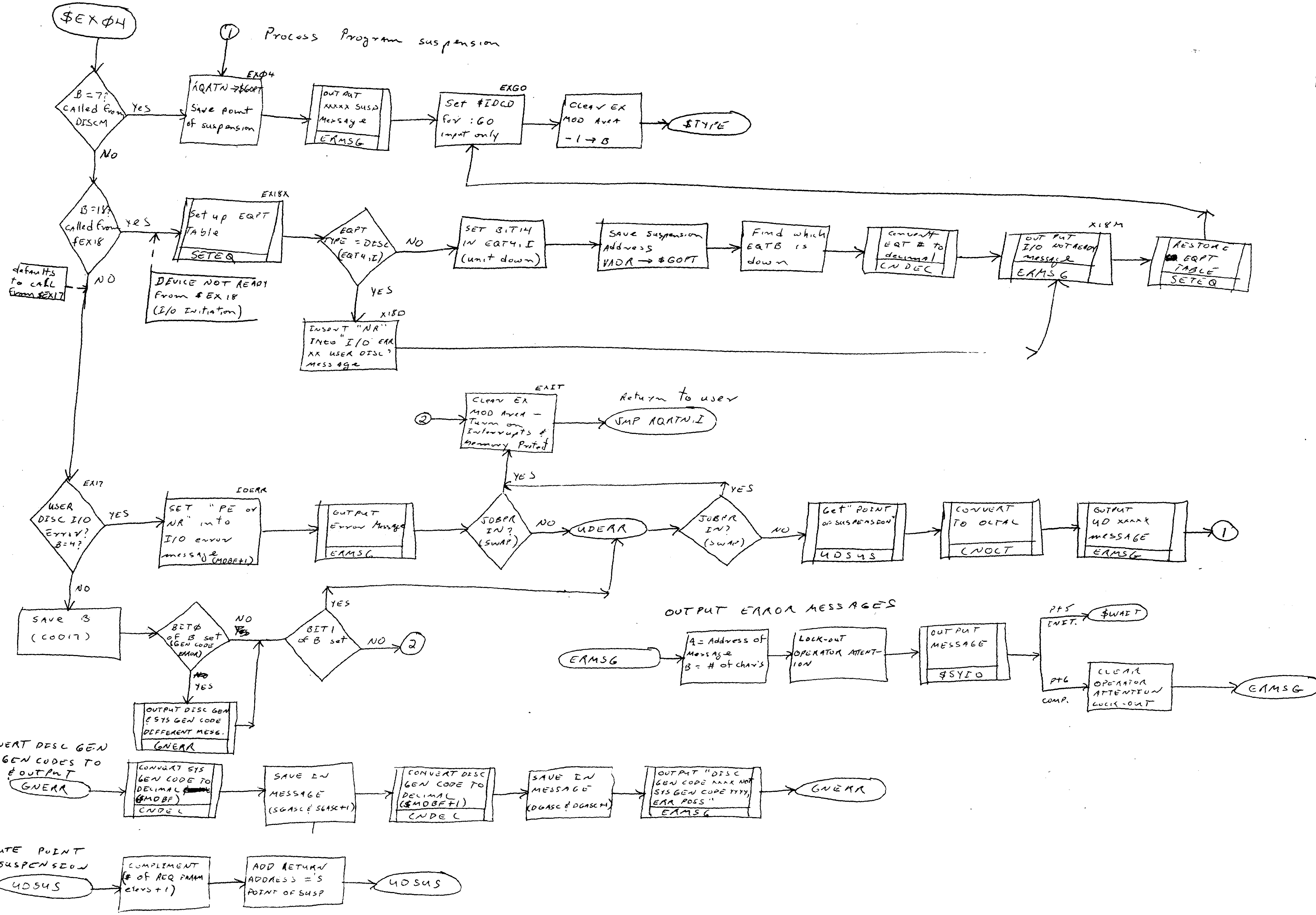
$EX02 (DISC WORK TRACK LIMITS)

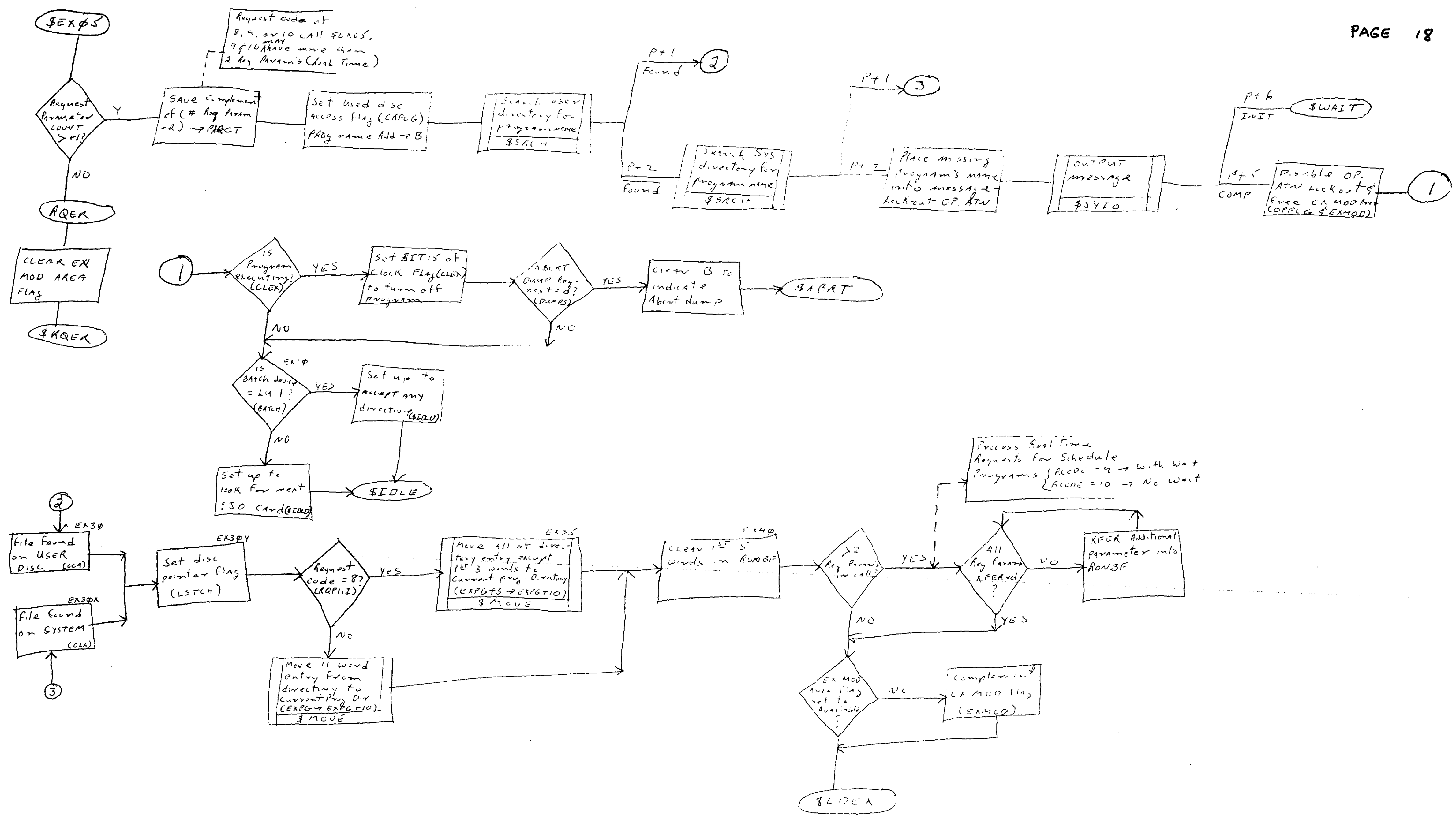


$EXØ3 (PROGRAM COMPLETION)

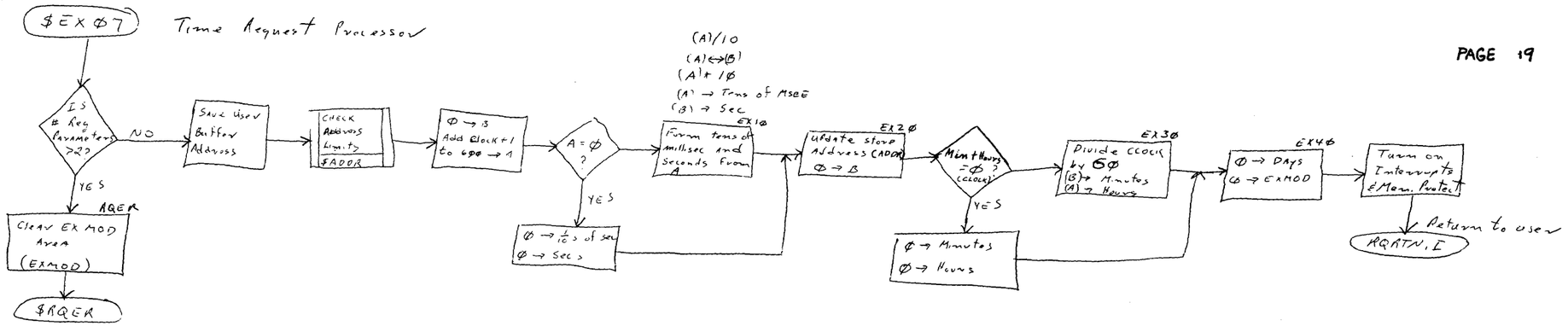


$EXØ6 (USER FILE NAME SEARCH)

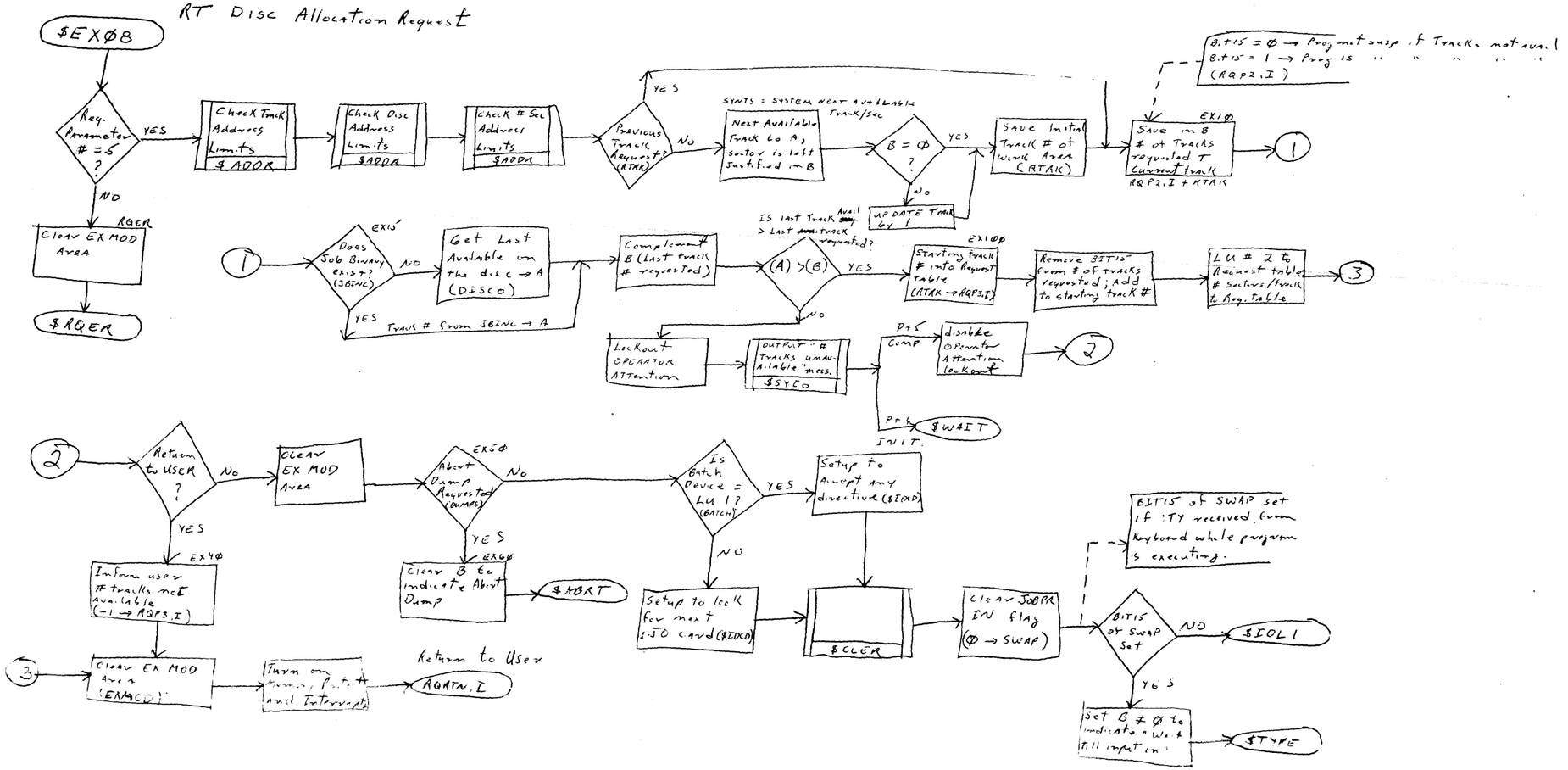

$EXØ4
Process Program suspension
B=7? called from DISCM
YES
NO
EXØ4
AQATN → $GOPT
Save point of suspension
OUTPUT XXXXX SUSP Message
ERMSG
EXGO
Set $IDCD for :GO input only
Clear EX MOD Area
-1 → B
$TYPE
B=18? called from $EX18
defaults to call from $EX17
EX18X
Set up EQPT Table
SETEQ
DEVICE NOT READY from $EX18 (I/O Initiation)
EQPT TYPE = DISC (EQT4,I)
SET BIT14 IN EQT4,I (unit down)
Save Suspension Address VADR → $GOPT
Find which EQTB is down
Convert EQT # to decimal
CNDEC
X18M
OUTPUT I/O NOT READY message
ERMSG
RESTORE EQPT TABLE
SETEQ
X18D
INSERT "NR" INTO "I/O ERR XX USER DISC" message
EXIT
Clear EX MOD Area — Turn on Interrupts & memory Protect
Return to user
JMP AQATN,I
EX17
USER DISC I/O Error? B=4?
IOERR
SET "PE or NR" into I/O error message (MOBF+1)
OUTPUT Error Message
ERMSG
JOBPR IN? (SWAP)
UDERR
Get POINT OF SUSPENSION
4DSUS
CONVERT TO OCTAL
CNOCT
OUTPUT 4D XXXXX MESSAGE
ERMSG
SAVE B (C0017)
BIT Ø of B set (GEN CODE ERROR)
OUTPUT DISC GEN & SYS GEN CODE DIFFERENT MESG.
GNERR
BIT 1 of B set
OUTPUT ERROR MESSAGES
ERMSG
A = Address of Message
B = # of char's
Lock-out OPERATOR ATTENTION
OUTPUT MESSAGE
$SYIO
P+5 INIT.
$WAIT
P+6 COMP.
CLEAR OPERATOR ATTENTION LOCK-OUT
ERMSG
CONVERT DISC GEN & SYS GEN CODES TO ASCII & OUTPUT
GNERR
CONVERT SYS GEN CODE TO DECIMAL ($MOBF)
CNDEC
SAVE IN MESSAGE (SGASC & SGASC+1)
CONVERT DISC GEN CODE TO DECIMAL ($MOBF+1)
CNDEC
SAVE IN MESSAGE (DGASC & DGASC+1)
OUTPUT "DISC GEN CODE XXXX NOT SYS GEN CODE YYYY, ERR POSS"
ERMSG
GNERR
COMPUTE POINT OF SUSPENSION
4DSUS
COMPLIMENT (# of REQ PARAM elems+1)
ADD RETURN ADDRESS = 'S POINT OF SUSP
4DSUS

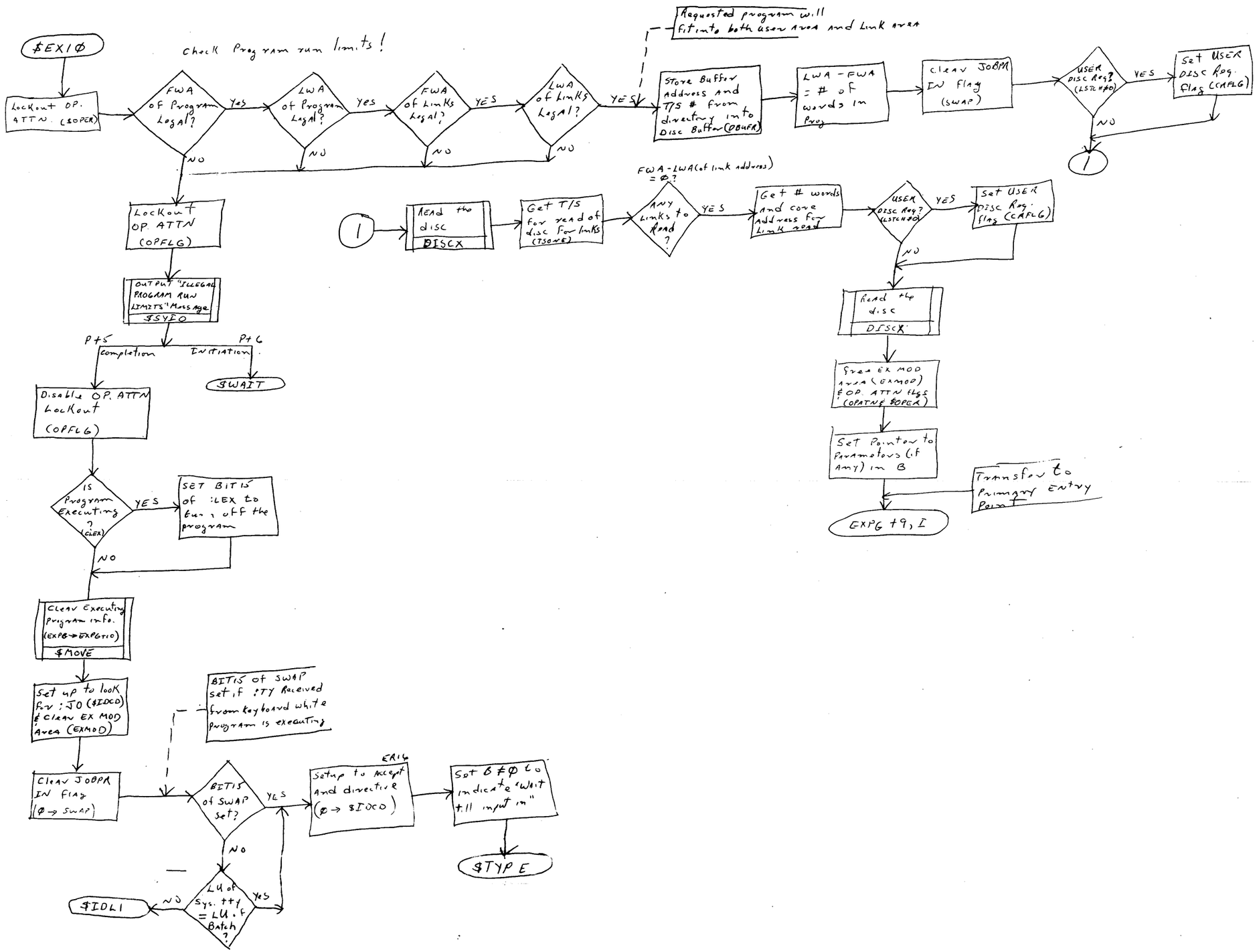

$EXØ5
Request code at 8, 9, or 10 call $EXO5. 9 & 10 may have more than 2 Req Param's (Real Time)
Request Parameter count > 1?
Y
NO
AQER
CLEAR EX MOD AREA Flag
$AQER
Save Complement of (# Req Param -2) → PARCT
Set Used disc access flag (EXFLG)
Prog name Add → B
Search user directory for program name
$SRCH
P+1
Found
P+2
Found
Search Sys directory for program name
$SRCH
P+1
P+2
Place missing program's name into message & Lockout OP ATN
Output message
$SYIO
P+6
INIT
$WAIT
P+5
COMP
Disable OP ATN Lockout & Free EX MOD Area (EXFLG & $EXMOD)
Is Program Executing? (CLEX)
YES
NO
Set BIT 15 of Clock Flag (CLEX) to turn off program
Abort Dump Requested? (DUMPS)
YES
NO
Clear B to indicate Abort dump
$ABRT
EX1Ø
Is BATCH device = LU 1? (BATCH)
YES
NO
Set up to accept any directive (SIDCO)
Set up to look for next :JO card (SIDCO)
$IDLE
EX3Ø
File found on USER DISC (CCA)
EX3ØX
File found on SYSTEM (CCA)
EX3ØY
Set disc pointer flag (LSTCH)
Request code = 8? (RQP1,I)
YES
NO
EX35
Move all of directory entry except 1st 3 words to Current prog. Directory (EXPGT5 → EXPGT10)
$MOVE
Move 11 word entry from directory to Current Prog Dir (EXPG → EXPGT10)
$MOVE
EX4Ø
Clear 1st 5 words in RQBUF
>2 Req Params in call?
YES
NO
Process Real Time Requests for Schedule Programs {RCODE = 9 → with wait; RCODE = 10 → No wait}
All Req Params XFERed?
NO
YES
XFER Additional parameter into RQBUF
Ex MOD Area Flag set to Available?
NO
Complement EX MOD Flag (EXMOD)
$IDLE

Time Request Processor

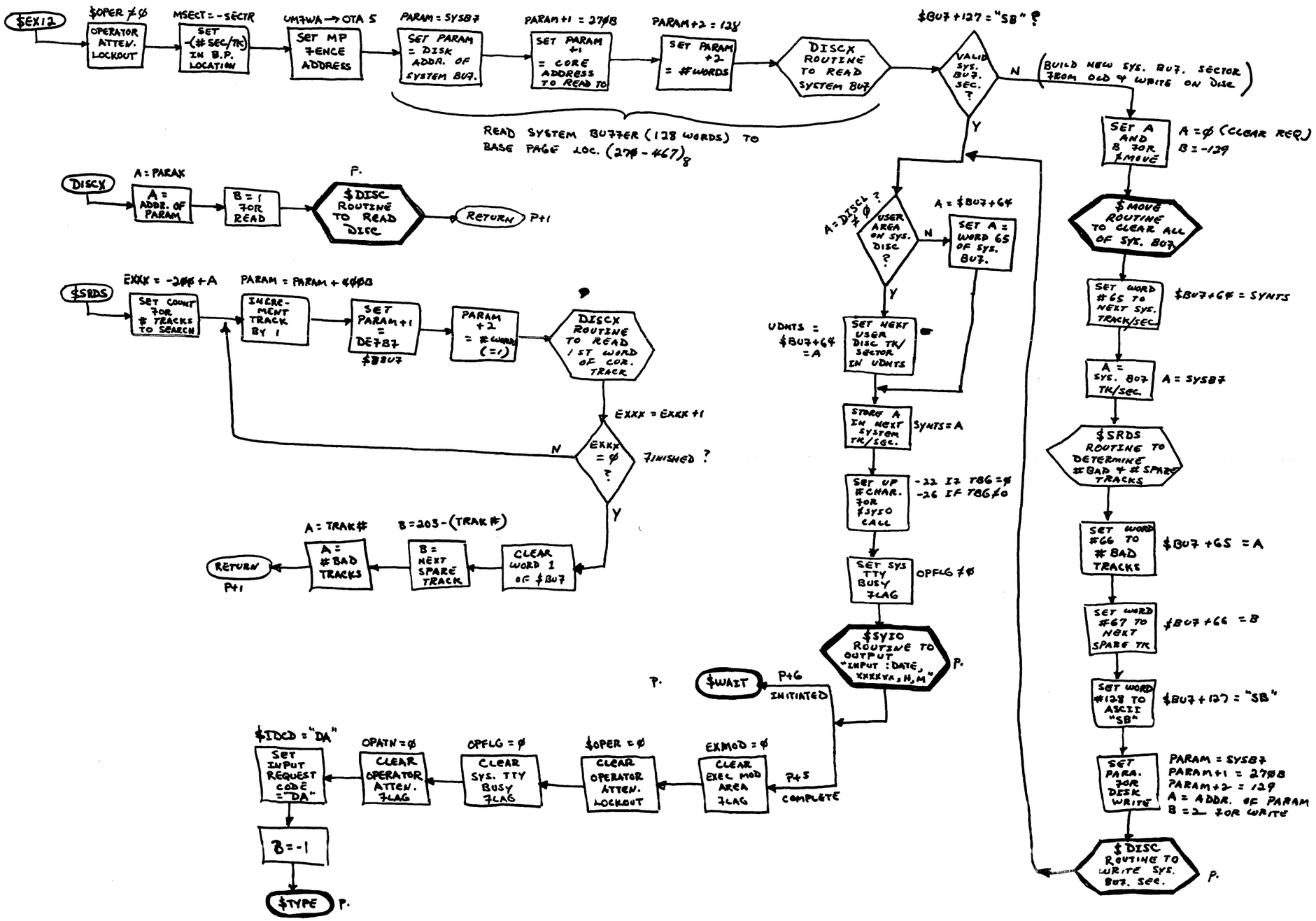


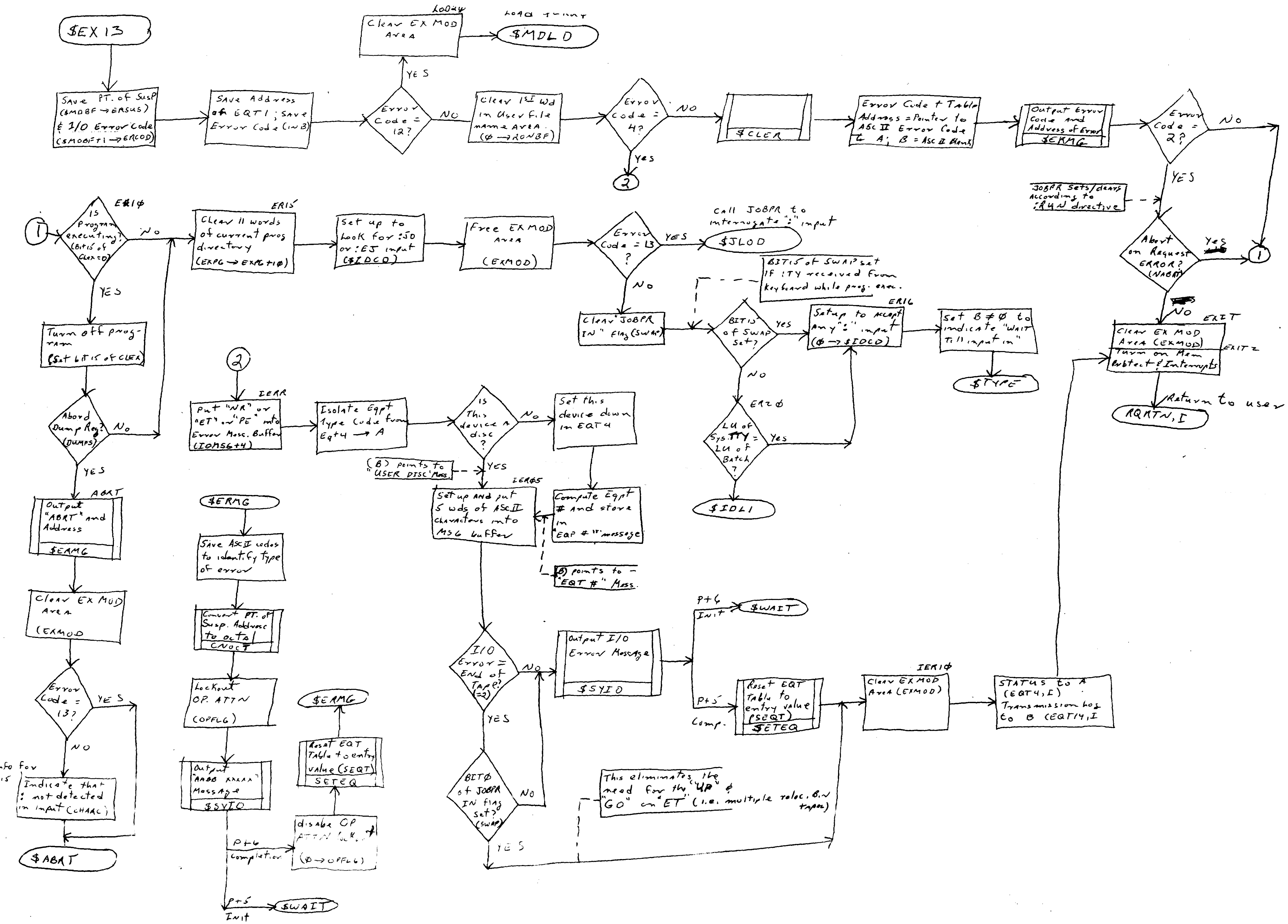
RT Disc Allocation Request

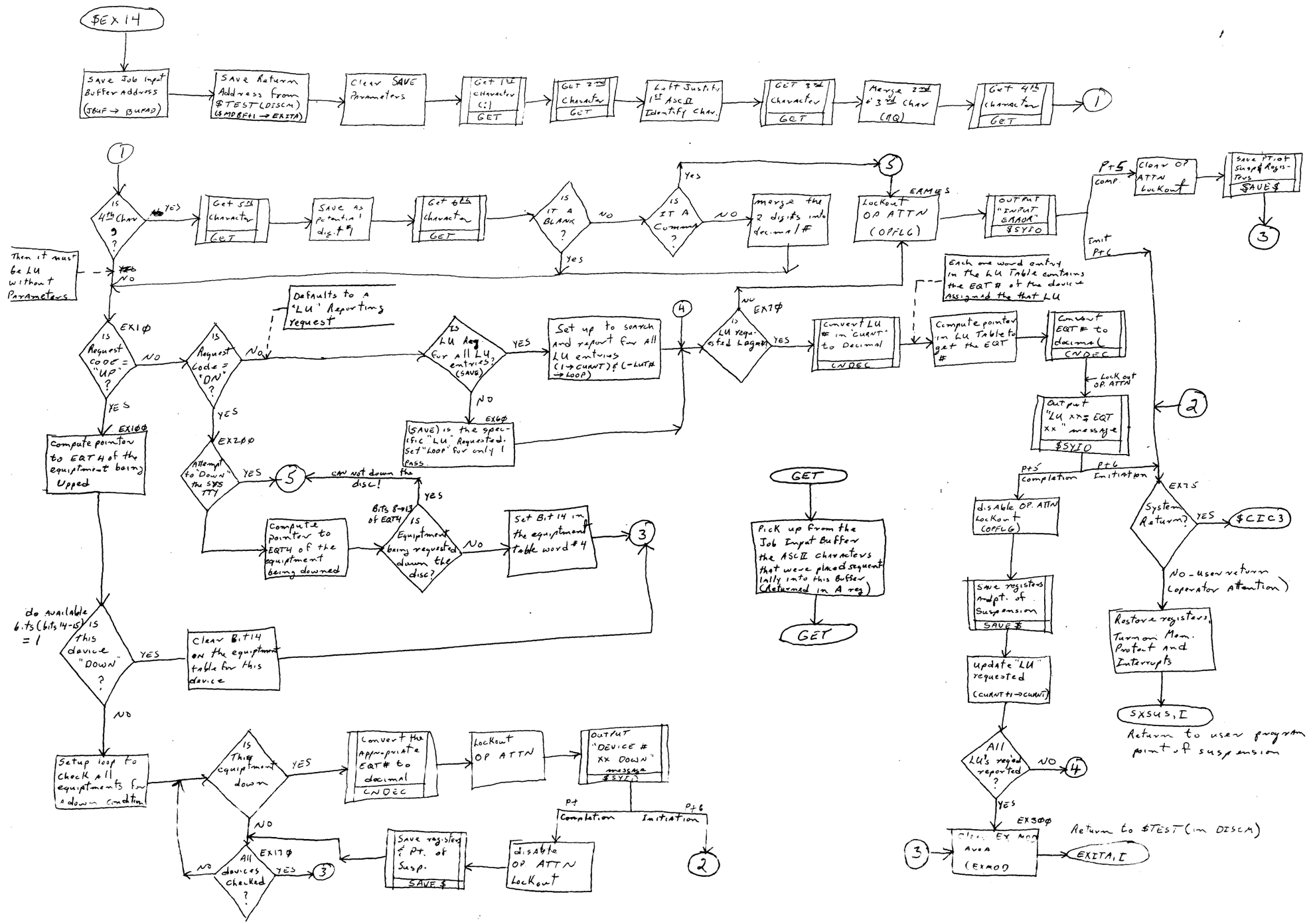


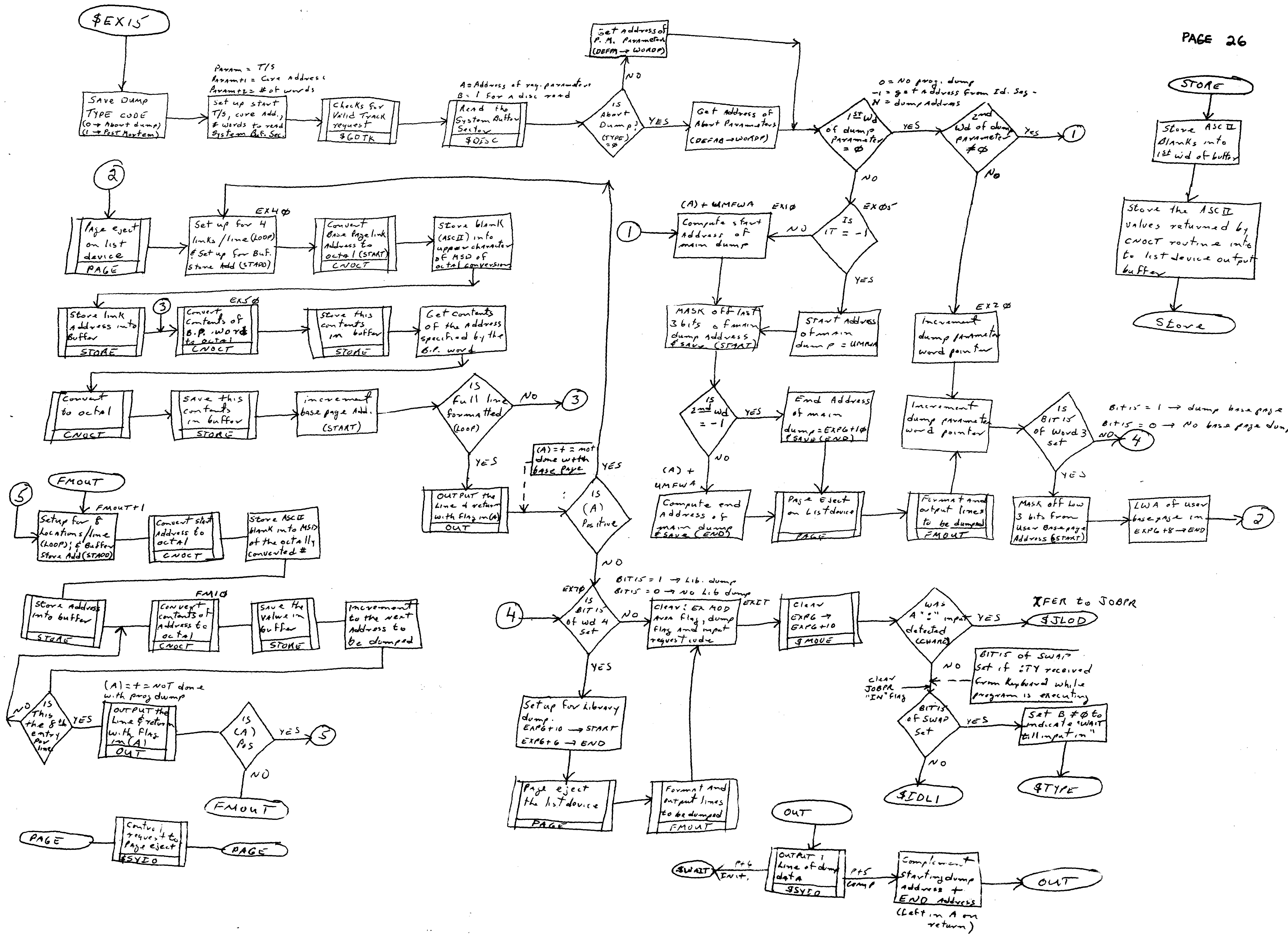

$EXIØ
Lockout OP. ATTN (SOPER)
Check Program run limits!
FWA of Program Legal?
Yes
NO
LWA of Program Legal?
Yes
NO
FWA of Links Legal?
YES
NO
LWA of Links Legal?
YES
NO
Requested program will fit into both user area and Link area
Store Buffer Address and T/S # from directory into Disc Buffer (DBUFR)
LWA - FWA = # of words in Prog
Clear JOBPM IN flag (SWAP)
USER DISC Req? (LSTCMD)
YES
NO
Set USER DISC Req. flag (CAFLG)
1
Lockout OP. ATTN (OPFLG)
OUTPUT "ILLEGAL PROGRAM RUN LIMITS" Message
$SYEO
P+5 completion
P+6 INITIATION
$WAIT
Disable OP. ATTN Lockout (OPFLG)
IS Program Executing? (CLEX)
YES
SET BIT15 of :LEX to turn off the program
NO
Clear Executing Program info. (EXPB→EXPB10)
$MOVE
Set up to look for :JO ($IDCO) & Clear EX MOD Area (EXMOD)
Clear JOBPM IN Flag (Ø → SWAP)
BIT15 of SWAP set if :TY Received from keyboard while program is executing
BIT15 of SWAP set?
YES
NO
LU of Sys. tty = LU of Batch?
YES
NO
$IDLI
Setup to Accept ERIG And directive ($ → $IDCO)
Set B ≠ Ø to indicate "Wait till input in"
$TYPE
1
Read the disc
DISCX
Get T/S for read of disc for links (TSOVE)
FWA - LWA(of link address) = Ø?
ANY Links to Read?
YES
Get # words and core address for Link read
USER DISC Req? (LSTCMD)
YES
NO
Set USER DISC Req. flag (CAFLG)
Read the disc
DISCX
Free ER MOD Area (EAMOD) & OP. ATTN flgs (OPATNE $OPER)
Set Pointer to Parameters (if Any) in B
Transfer to Primary Entry Point
EXPG +9, I

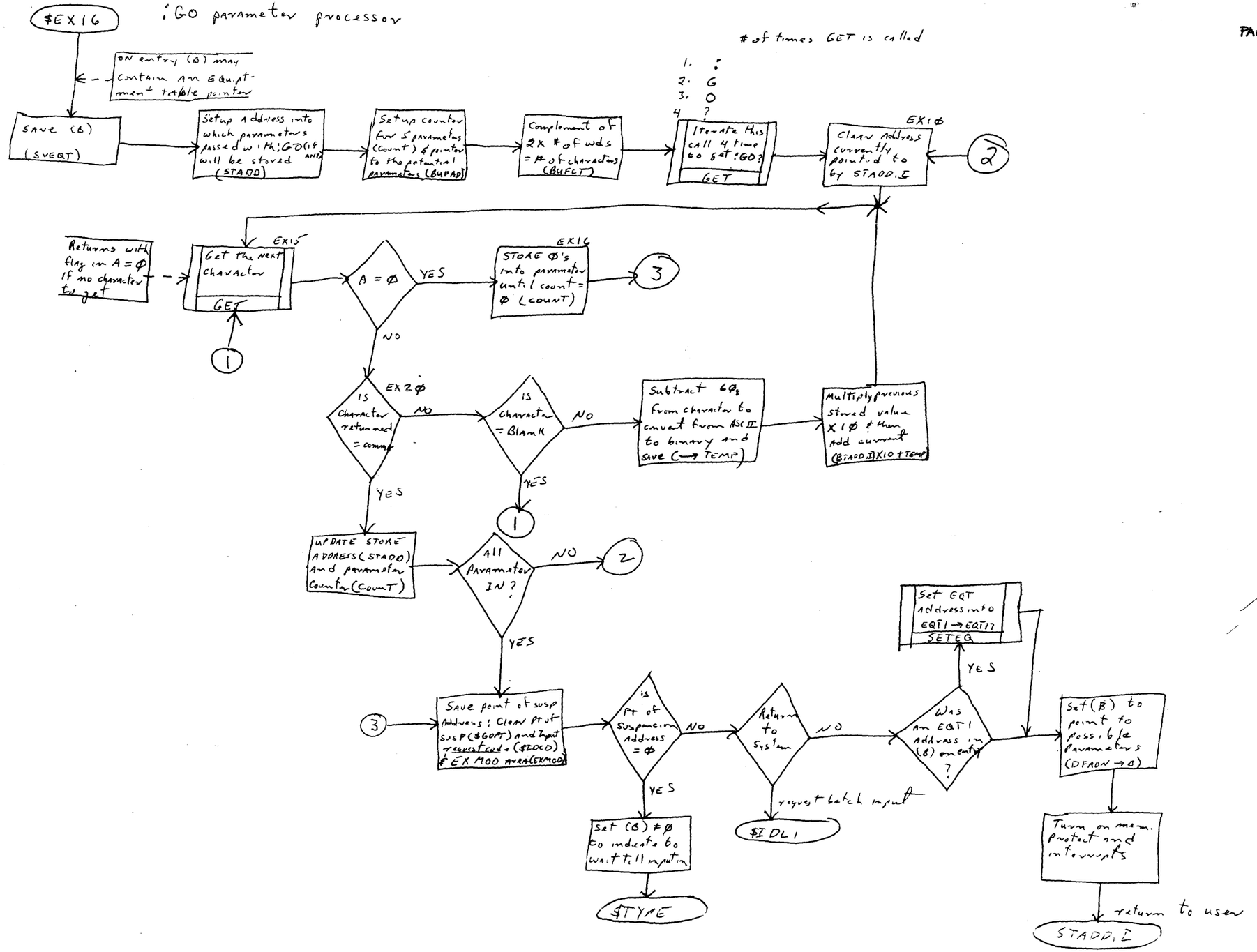
$EX12 (SYSTEM STARTUP)



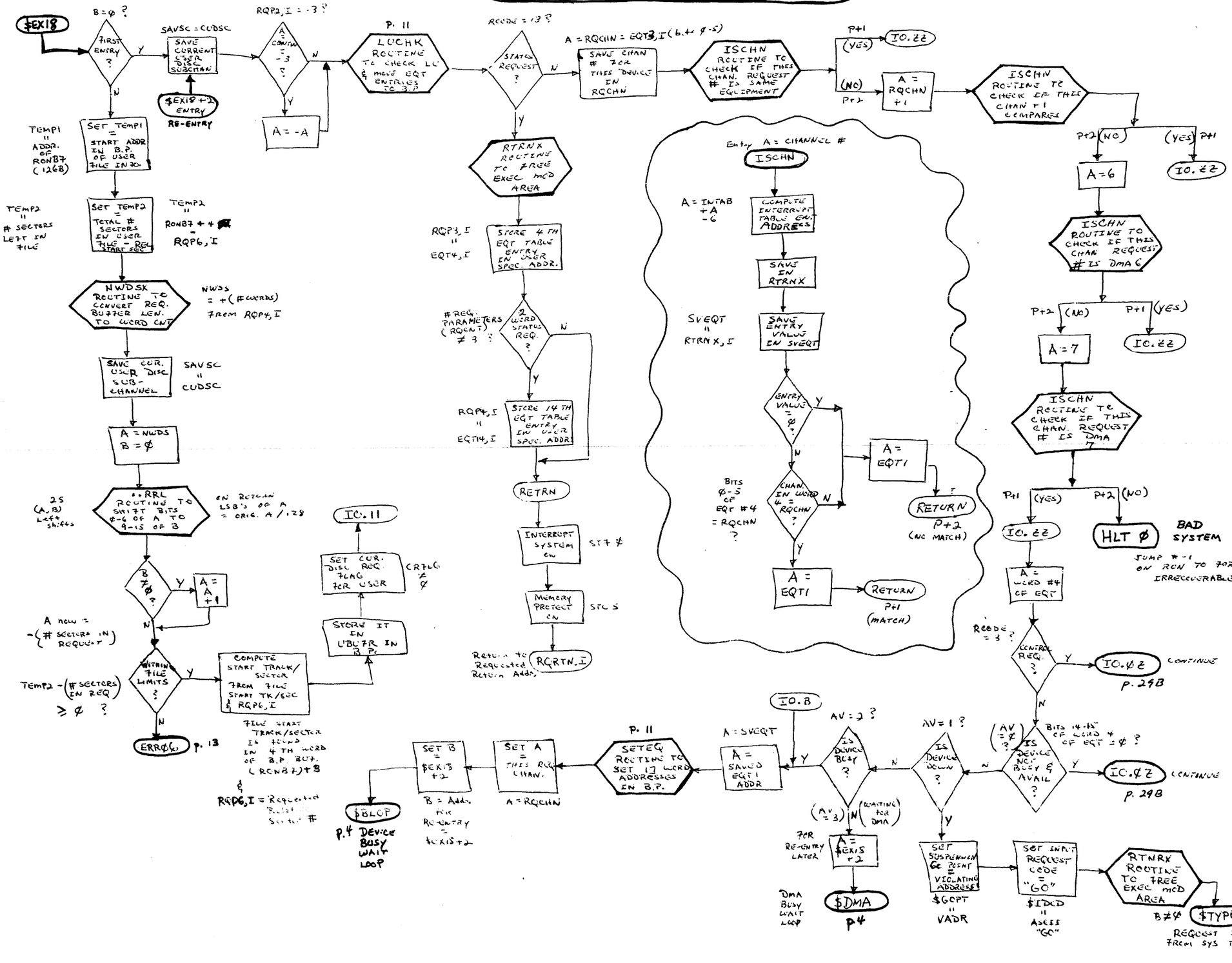

$EX13
Save PT. of Susp ($MOBF → ERSUS) & I/O Error Code ($MOBFT1 → ERCOD)
Save Address of EQT1; Save Error Code (in B)
Error Code = 12?
YES
Clear EX MOD Area
$MDLD
NO
Clear 1st wd in User file name area. (∅ → RONBF)
Error Code = 4?
Yes
2
$CLER
Error Code + Table Address = Pointer to ASC II Error Code to A; B = Asc II Blank
Output Error Code and Address of Error
$ERMG
Error Code = 2?
YES
JOBPR Sets/clears According to :RUN directive
Abort on Request ERROR? (NABR)
Yes
1
No
EXIT
Clear EX MOD Area (EXMOD)
Turn on Mem Protect & Interrupts
Return to user
RQRTN, I
1
ER10
Is Program executing? (Bit 15 of CLEX=0)
YES
Turn off program ($et bit 15 of CLEX)
Abort Dump Req? (DUMPS)
YES
ABRT
Output "ABRT" and Address
$ERMG
Clear EX MOD Area (EXMOD)
Error Code = 13?
YES
NO
Info for $ER15
Indicate that : not detected in input (CHARC)
$ABRT
ER15
Clear 11 words of current prog directory (EXPG → EXPG+10)
Set up to Look for :SO or :EJ input ($IDCD)
Free EXMOD Area (EXMOD)
Error Code = 13 ?
YES
Call JOBPR to interrogate ":" input
$JLOD
No
Clear JOBPR IN" Flag (SWAP)
BIT 15 of SWAP set IF :TY received from keyboard while prog. exec.
BIT 15 of SWAP Set?
Yes
ER16
Set up to accept any ":" input (∅ → $IDCD)
Set B ≠ ∅ to indicate "WAIT Till input in"
$TYPE
No
ER20
LU of Sys. TTY = LU of Batch?
Yes
$IDLI
2
IERR
Put "WR" or "ET" or "PE" into Error Msg. Buffer (IOMSG+4)
Isolate Eqpt Type Code from Eqt4 → A
Is This device a disc?
No
Set this device down in EQT4
(B) points to "USER DISC" Mess.
Yes
IER05
Set up and put 5 wds of ASC II characters into MSG buffer
Compute Eqpt # and store in "EQP #" message
(B) points to "EQT #" Mess.
I/O Error = End of Tape? (=2)
NO
Output I/O Error Message
$SYIO
P+6 Init
$WAIT
P+5 Comp.
Reset EQT Table to entry value (SEQT)
$ETEQ
IER10
Clear EXMOD Area (EXMOD)
STATUS to A (EQT4, I) Transmission log to B (EQT14, I
YES
BIT∅ of JOBPR IN flag Set? (SWAP)
NO
YES
This eliminates the need for the "UP" & "GO" on "ET" (i.e. multiple reloc. bin tapes)
$ERMG
Save ASC II codes to identify type of error
Convert PT. of Susp. Address to octal
CNOCT
Lockout OP. ATTN (OPFLG)
Output "ABBB XXXXX" Message
$SYIO
P+6 Completion
disable OP ATTN Lockout (∅ → OPFLG)
Reset EQT Table to entry value (SEQT)
SETEQ
$ERMG
P+5 Init
$WAIT


$EX14
Save Job Input Buffer Address (JBUF → BUFAD)
Save Return Address from $TEST (DISCM) (GMDBF+1 → EXITA)
Clear SAVE Parameters
Get 1st character (:)
GET
Get 2nd Character
GET
Left Justify, 1st ASCII Identify char.
GET 3rd character
GET
Merge 2nd & 3rd Char (INQ)
Get 4th character
GET
1
Is 4th Char 3 ?
YES
Get 5th character
GET
Save as potential digit #1
Get 6th character
GET
Is IT A BLANK ?
NO
YES
Is IT A Comma ?
YES
NO
merge the 2 digits into decimal I#
Then it must be LU without Parameters
NO
Defaults to a "LU" Reporting request
Is Request code = "UP" ?
EX1Φ
NO
YES
Is Request Code = "DN" ?
NO
YES
EX2ΦΦ
Is LU Req. for all LU entries (SAVE)
YES
NO
Set up to search and report for all LU entries (1→CURNT) & (-LUT#→LOOP)
EX6Φ
(SAVE) is the specific "LU" Requested. Set "LOOP" for only 1 pass.
4
Is LU requested Logged
NO
EX7Φ
YES
Convert LU # in "CURNT" to Decimal
CNDEC
Each one word entry in the LU Table contains the EQT # of the device Assigned the that LU
Compute pointer in LU Table to get the EQT #
Convert EQT # to decimal
CNDEC
Lock out OP. ATTN
Output "LU xx≠ EQT xx" message
$SYIO
2
Pt 5 completion
Pt 6 Initiation
Compute pointer to EQT 4 of the equipment being Upped
Attempt to "Down" the SYS TTY
YES
5
can not down the disc!
YES
Bits 8→13 of EQT4 IS Equipment being requested down the disc?
NO
Compute pointer to EQT4 of the equipment being downed
Set Bit 14 in the equipment table word #4
3
do available bits (bits 14-15) = 1
IS this device "DOWN" ?
YES
Clear Bit 14 on the equipment table for this device
NO
Setup loop to check all equipments for a down condition
Is This equipment down
YES
NO
Convert the appropriate EQT# to decimal
CNDEC
LockOut OP ATTN
OUTPUT "DEVICE # XX DOWN" message
$SYIO
Pt 5 Completion
Pt 6 Initiation
disable OP ATTN LockOut
Save registers & Pt. of Susp.
SAVE $
All devices checked ?
EX17Φ
NO
YES
3
LockOut OP ATTN
OUTPUT "INPUT ERROR"
$SYFO
Pt 5 COMP
Clear OP ATTN LockOut
Save Pt. of Susp & Registers
SAVE $
3
Init Pt 6
2
GET
Pick up from the Job Input Buffer the ASCII characters that were placed sequentially into this Buffer (Returned in A reg)
GET
Pt 5 Completion
Pt 6 Initiation
disable OP. ATTN LocKout (COFFLG)
Save registers & pt. of Suspension
SAVE $
update "LU" requested (CURNT+1→CURNT)
All LU's req'ed reported ?
NO
4
YES
EX15
System Return?
YES
$CIC3
No - user return (operator Attention)
Restore registers, Turn on Mem. Protect and Interrupts
SXSUS,I
Return to user program point of suspension
EX3ΦΦ
Return to $TEST (in DISCM)
3
EXITA,I

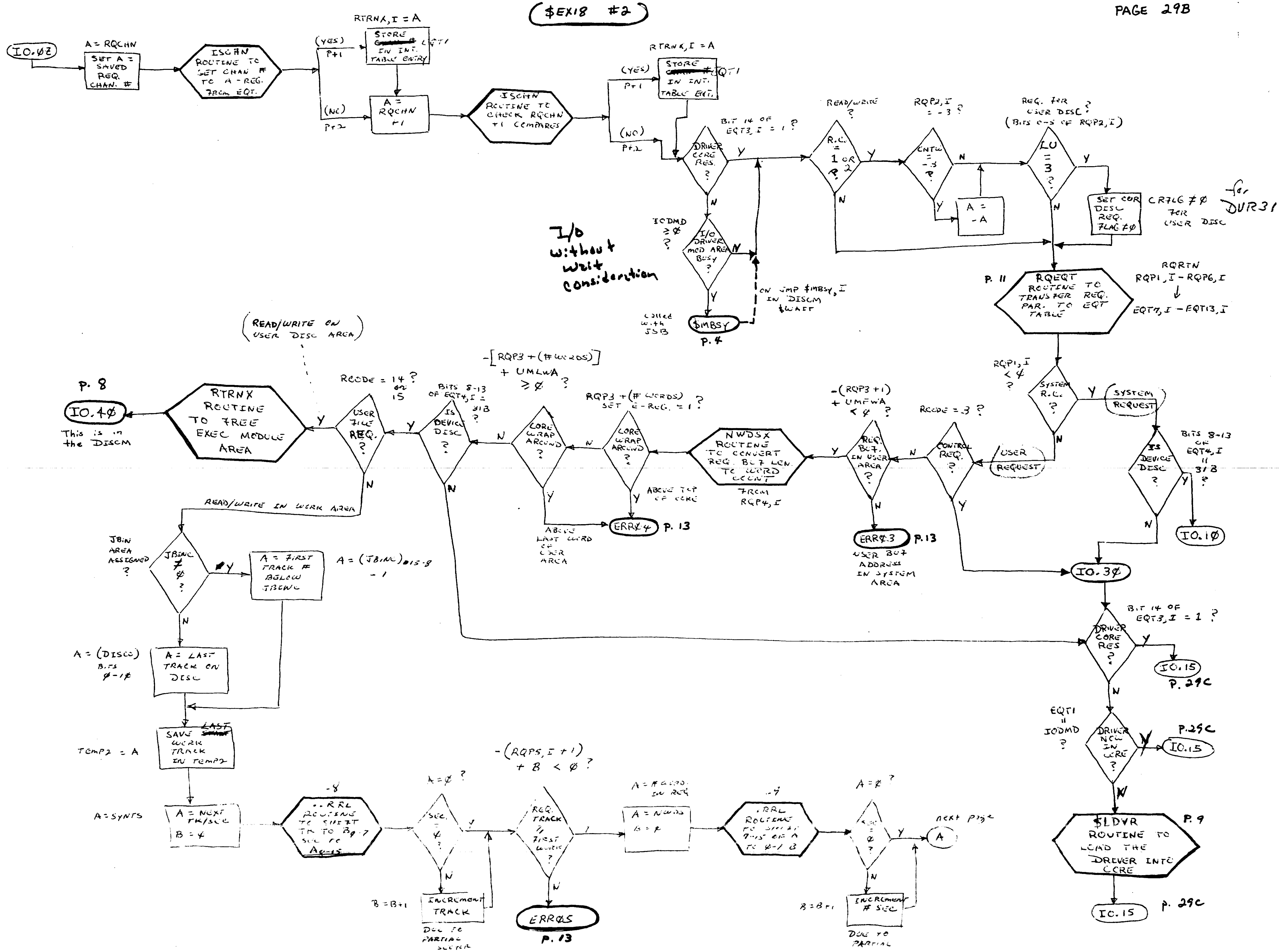

$EXIS
Save Dump TYPE CODE (0 → Abort dump) (1 → Post Mortem)
Param = T/S
Param+1 = Core Address
Param+2 = # of words
Set up start T/S, core Add., # words to read System Buf. Sec.
Checks for Valid Track request
$GOTK
A = Address of req. parameter
B = 1 for a disc read
Read the System Buffer Sector
$DISC
Is Abort Dump? (TYPE) = Ø
NO
YES
Set Address of P. M. Parameters (DEFPM → WORDP)
Get Address of Abort Parameters (DEFAB → WORDP)
0 = No prog. dump
-1 = get Address from Id. Seg.
N = dump Address
1st wd of dump parameter = Ø
2nd wd of dump parameter ≠ Ø
Yes
1
No
STORE
Store ASC II Blanks into 1st wd of buffer
Store the ASC II values returned by CNOCT routine into list device output buffer
Store
2
Page eject on list device
PAGE
EX4Ø
Set up for 4 links/line (LOOP) & Set up for Buf. Store Add (STADD)
Convert Base Page link Address to octal (START)
CNOCT
Store blank (ASC II) into upper character of MSD of octal conversion
(A) + UMFWA
EX1Ø
Compute start Address of main dump
EXØ5
Is IT = -1
Store link Address into buffer
STORE
3
EX5Ø
Convert Contents of B.P. word to octal
CNOCT
Store this contents in buffer
STORE
Get contents of the Address specified by the B.P. word
MASK off last 3 bits of main dump address & save (START)
Start Address of main dump = UMFWA
EX2Ø
Increment dump parameter word pointer
Convert to octal
CNOCT
Save this contents in buffer
STORE
increment base page Add. (START)
Is full line formatted (LOOP)
Is 2nd wd = -1
End Address of main dump = EXPG+1Ø & SAVE (END)
Increment dump parameter word pointer
Is BIT 15 of Word 3 set
Bit 15 = 1 → dump base page
Bit 15 = 0 → No base page dump
4
(A) = + = not done with base page
OUTPUT the Line + return with Flag in (A)
OUT
Is (A) Positive
(A) + UMFWA
Compute end Address of main dump & SAVE (END)
Page Eject on List device
PAGE
Format and output lines to be dumped
FMOUT
MASK off Low 3 bits from User Base page Address (START)
LWA of user base page in EXPG+8 → END
FMOUT
5
FMOUT+1
Set up for 8 Locations/line (LOOP); & Buffer Store Add (STADD)
Convert start Address to octal
CNOCT
Store ASC II blank into MSD of the octally converted #
EXTØ
BIT 15 = 1 → Lib. dump
BIT 15 = 0 → No Lib dump
Is BIT 15 of wd 4 set
EXIT
Clear: EX MOD Area Flag, dump Flag and input request code
Clear EXPG → EXPG+1Ø
$MOVE
Was A ":" input detected (CHAR)
XFER to JOBPR
$JLOD
Store Address into buffer
STORE
FM1Ø
Convert contents of Address to octal
CNOCT
Save the value in buffer
STORE
Increment to the next Address to be dumped
Set up for Library dump. EXPG+10 → START EXPG+6 → END
BIT 15 of SWAP set if :TY received from Keyboard while program is executing
Clear JOBPR "IN" Flag
BIT 15 of SWAP Set
Set B ≠ Ø to indicate "WAIT till input"
$TYPE
$IDLI
Is This the 8th entry per line
(A) = + = NOT done with prog dump
OUTPUT the Line & return with Flag in (A)
OUT
Is (A) Pos
FMOUT
Page eject the list device
PAGE
Format and output lines to be dumped
FMOUT
OUT
PAGE
Control request to Page eject
$SYIO
PAGE
$SWAIT
P+4 Init.
OUTPUT 1 Line of dump data
$SYIO
P+5 Comp
Complement starting dump address + END Address (Left in A on return)
OUT


$EX16
:GO parameter processor
ON entry (B) may contain an Equipment table pointer
SAVE (B) ($VEQT)
Setup Address into which parameters passed with :GO if will be stored (STADD)
Setup counter for 5 parameters (Count) & pointer to the potential parameters (BUPAD)
Complement of 2x # of wds = # of characters (BUFCT)
# of times GET is called
1. :
2. G
3. O
4. ?
Iterate this call 4 time to get :GO?
GET
EX10
Clear Address currently pointed to by STADD,I
2
Returns with flag in A = Ø if no character to get
EX15
Get the next character
GET
1
A = Ø
YES
EX16
STORE Ø's into parameter until (count = Ø (COUNT)
3
NO
EX20
IS Character returned = comma
NO
IS character = Blank
NO
Subtract 60₈ From character to convert from ASCII to binary and save (→ TEMP)
Multiply previous stored value X10 & then Add current (BTADD,I)X10 + TEMP
YES
1
YES
UPDATE STORE ADDRESS (STADD) And parameter Counter (COUNT)
All Parameters IN?
NO
2
YES
3
Save point of susp Address; Clear Prot susp ($GOAT) and Input request word ($IOCO) & EX MOD AREA (EXMOD)
Is Pt of Suspension Address = Ø
NO
Return to System
NO
Was an EQT1 Address in (B) on entry?
YES
Set EQT Address into EQT1 → EQT17
SETEQ
Set (B) to point to possible Parameters (DFADN → A)
Turn on mem. Protect and interrupts
return to user
STADD,I
YES
Set (B) ≠ Ø to indicate to wait till input
$TYPE
request batch input
$IDL1

$EX18 (INPUT/OUTPUT CONTROL PROCESSOR) #1



($EX18 #2)


IO.02
A = RQCHN
SET A = SAVED REQ. CHAN. #
ISCHN ROUTINE TO GET CHAN # TO A-REG. FROM EQT.
(YES) P+1
(NO) P+2
RTRNX,I = A
STORE IN INT. TABLE ENTRY
A = RQCHN +1
ISCHN ROUTINE TO CHECK RQCHN +1 COMPARES
STORE IN INT. TABLE EQT.
BIT 14 OF EQT3,I = 1 ?
DRIVER CORE RES. ?
IODMD ≥ 0 ?
I/O DRIVER MOD AREA BUSY ?
I/O without wait consideration
ON JMP $MBSY,I IN DISCM $WAIT
called with JSB
$MBSY
P. 4
READ/WRITE ?
R.C. = 1 OR 2
RQP2,I = -3 ?
CNTW = -3 ?
A = -A
REQ. FOR USER DISC ? (BITS 0-5 OF RQP2,I)
LU = 3 ?
SET COR DISC REQ. FLAG ≠ 0
CR7LG ≠ 0 FOR USER DISC
for DVR31
RQRTN RQP1,I - RQP6,I → EQT9,I - EQT13,I
P. 11
RQEQT ROUTINE TO TRANSFER REQ. PAR. TO EQT TABLE
RQP1,I < 4 ?
SYSTEM R.C. ?
SYSTEM REQUEST
USER REQUEST
BITS 8-13 OF EQT4,I = 31B ?
IS DEVICE DISC ?
IO.10
RCODE = 3 ?
CONTROL REQ. ?
-(RQP3 +1) + UMFWA < 0 ?
REQ. BUF. IN USER AREA ?
ERR03 P.13
USER BUF ADDRESS IN SYSTEM AREA
NWDSX ROUTINE TO CONVERT REQ. BUF. LEN. TO WORD COUNT FROM RQP4,I
RQP3 + (# WORDS) SET E-REG. = 1 ?
CORE WRAP AROUND ?
ABOVE TOP OF CORE
-[RQP3 + (# WORDS)] + UMLWA ≥ 0 ?
ABOVE LAST WORD OF USER AREA
ERR04 P. 13
RCODE = 14 OR 15 ?
USER FILE REQ. ?
(READ/WRITE ON USER DISC AREA)
RTRNX ROUTINE TO FREE EXEC MODULE AREA
IO.40
P. 8
This is in the DISCM
IO.30
DRIVER CORE RES ?
IO.15
P. 29C
EQT1 = IODMD ?
DRIVER NOW IN CORE ?
$LDVR ROUTINE TO LOAD THE DRIVER INTO CORE
P. 9
READ/WRITE IN WORK AREA
JBIN AREA ASSIGNED ?
JBINC ≠ 0 ?
A = FIRST TRACK # BELOW JBINC
A = (JBINC)15-8 - 1
A = (DISCO) BITS 0-10
A = LAST TRACK ON DISC
TEMP2 = A
SAVE LAST WORK TRACK IN TEMP2
A = SYNTS
A = NEXT TK/SEC B = 4
-8
RRL ROUTINE TO SHIFT TR TO B
A = 0 ?
SEC. = 0 ?
B = B+1
INCREMENT TRACK
DUE TO PARTIAL SECTOR
-(RQP5,I +1) + B < 0 ?
REQ. TRACK ≥ FIRST WORK ?
ERR05
P. 13
A = # WORDS IN REQ
A = NWDS B = 0
-4
RRL ROUTINE
A = 0 ?
INCREMENT # SEC
DUE TO PARTIAL
next page
A

$EX18 #3

A

B = B + RQP6, I

ADD REQ. SECTOR ADDR TO #SEC. DUE TO # WORDS

A = B / #SEC/TK + RQP5, I

LOOP TO GET LAST TRACK USED IN REQ. TO A-REG.

-A + TEMP2 ≥ 0 ?

LAST REQ. TK. ≤ LAST WORK ?

N → ERR#5 (P. 13)

Y

RQP6, I + MSECT < 0 ?

REQ. SEC. # < #SEC/TK LOCAL ?

N → ERR#5 (P. 13)

Y

SET UP USER DESC TRIPLET PARAMETER BUF. IN BASE PAGE LOC. 171-173 UBUFR UBUFR+1, +2

171 = (RQP5, I)8-15 (RQP6, I)0-7

172 = RQP3 (CORE ADD.)

173 = RQP4, I (WORD COUNT)

RQCNT = 6 ?

REQ. PARA. = 6 ?

N → ERR#1 (P. 13)

Y

A = 2

LUCHK ROUTINE TO PUT EQT1 - EQT17 FOR LU 2 IN B.P. (P. 11)

RQEQT ROUTINE TO TRANSFER REQ PARAMETERS TO EQUIP. TABLE (P. 11)

CNTLW WORD = -3 ?

RQP2, I = -3 ?

Y

SET COR. USER DESC SUBCHAN = SOURCE FILE SUBCHANNEL

CUDSC 11 JFLSC

N

A = ADDR. OF USER DESC TRIPLET PARAMETER BUFFER

A = 171₈

IO.15

STORE IN TENTH EQUIP. TABLE ENTRY

EQT10, I = A

★★★ P.6 ★★★

DRIVR ROUTINE TO CALL DRIVER INITIATOR SECTION

OPERATION INITIATED P+1

RTRNX ROUTINE TO FREE EXEC MODULE AREA

CUDSC = SAVSC

RESTORE CURRENT USER DESC. SUBCHANNEL

CNTLW WORD = -3 ?

RQP2, I = -3

Y

A = +3

N

BIT 13 OF RQP2, I = 1 ?

I/O WITHOUT WAIT BIT ON ?

N (WAIT) → $WAIT (P. 4)

Y (NO WAIT) → RETRN (P. $EX18 #1)

CONVERT REQUEST BUFFER LENGTH TO WORD COUNT

NWDSK

RQP4, I

REQ. BUF. LEN. < 0 ?

Y → DIVIDE BY 2 AND 2's COMP

N

STORE AS + # WORDS

NWDS = +(# WORDS)

RETURN P+1

A = Reason OTHER OPERATION

- REJECT
- IMMEDIATE COMP.
- NO DMA AVAIL
- MALFUNCTION

...etc...

CUDSC = SAVSC

RESTORE CURRENT USER DESC SUBCHANNEL

P+2

WAITING FOR DMA ?

Y → A = ADDR. OF IO.15 FOR RETRY → $DMA (P. 4)

N

CONTL OPER. REJECT ?

Y → IO.19A → RTRNX ROUTINE TO FREE EXEC MOD AREA → A = EQT4, I → SET A = STATUS FROM EQT TABLE → B = 0 → SET TRANSMISSION LOG. = 0 → RETRN (P. $EX18 #1)

N

EQUIP MALFUNC OR NOT READY ?

Y → SET ADDR. OF CUR. EQT TABLE START IN EXEC MOD DATA BUF → $MDB7 = EQT1 → TEMP 11 DE704 → RTRNX ROUTINE TO FREE EXEC MOD AREA → B = 18, A = 4 → $MDLD (P. 8) To load $EX04

N

A = 6 ?

Y → DRIVER REQUIRE A WAIT ? → Y → IO.B

N

A = 4 ?

IMMED COMP ? → Y → IO.19A

N → ERRTN (P. 13) operation reject

A = 3 ?

A = 2 ?

A = 5 ?

FREE EXEC MODULE AREA

RTRNX

SET A = EXEC MODULE CORE FLAG → A = EXMOD

A < 0 ?

TIMER FLAG < 0 ?

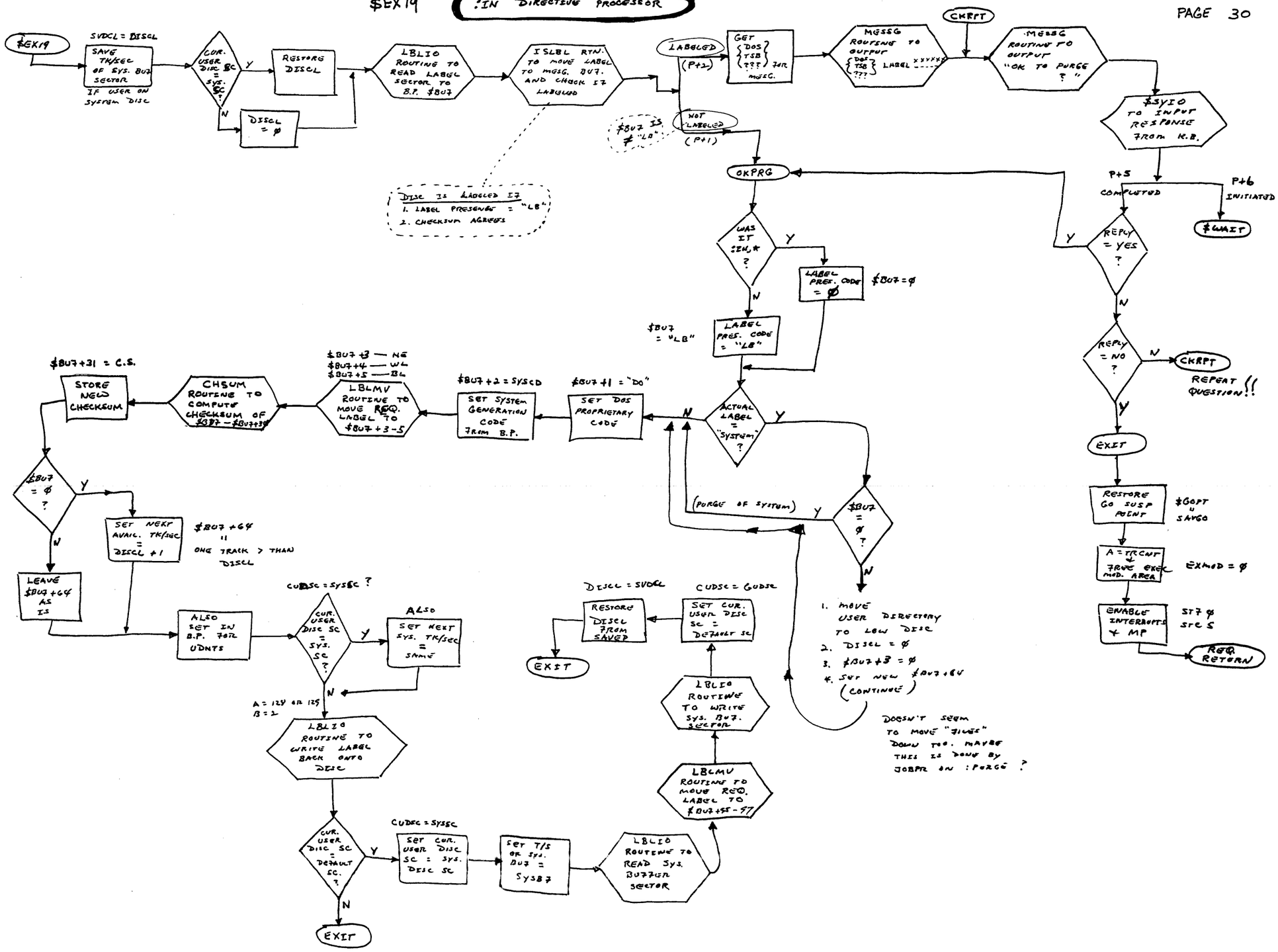
Y → A = -A

N

STORE BACK IN EXMOD

RETURN P+1

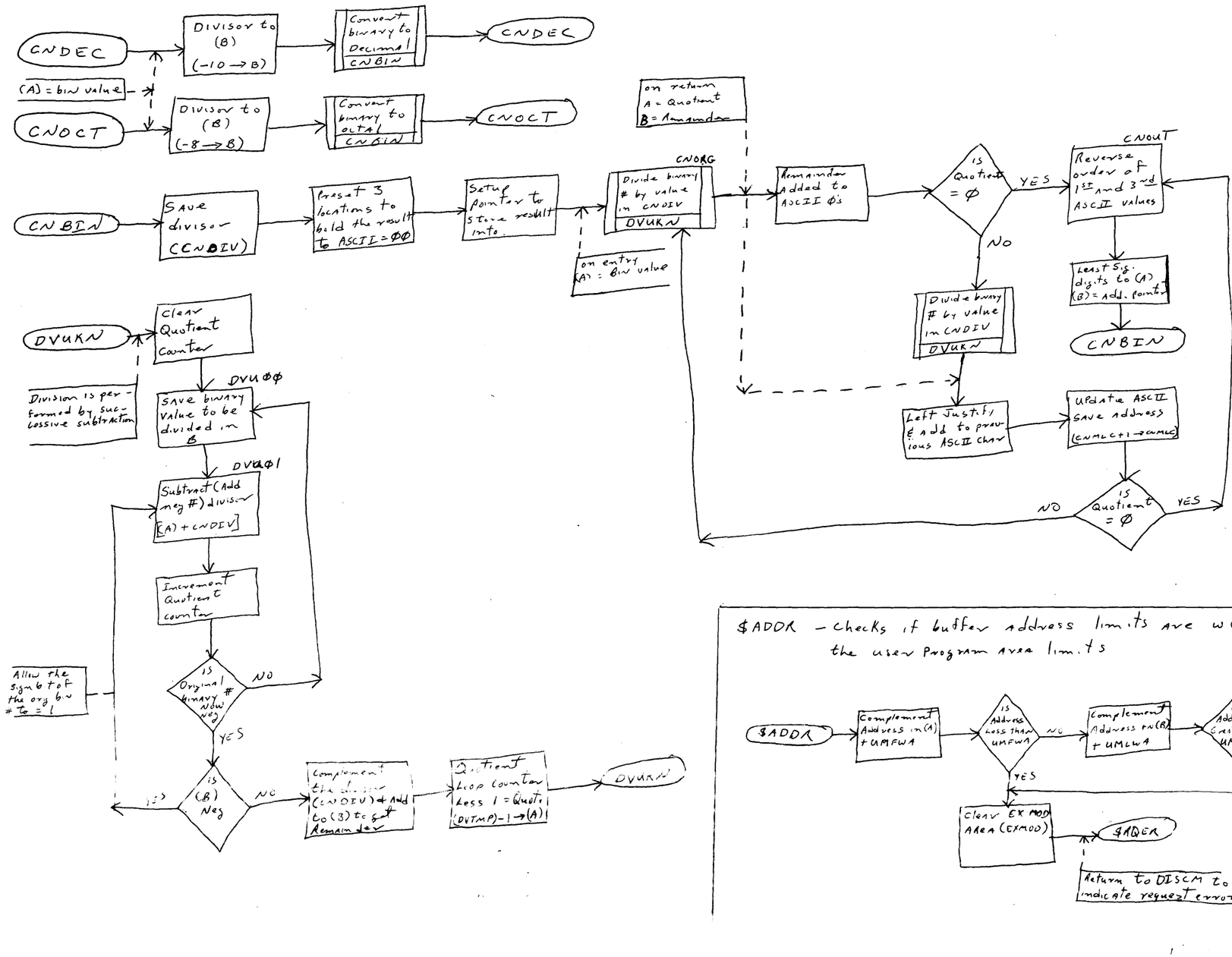
$EX19 'IN DIRECTIVE PROCESSOR



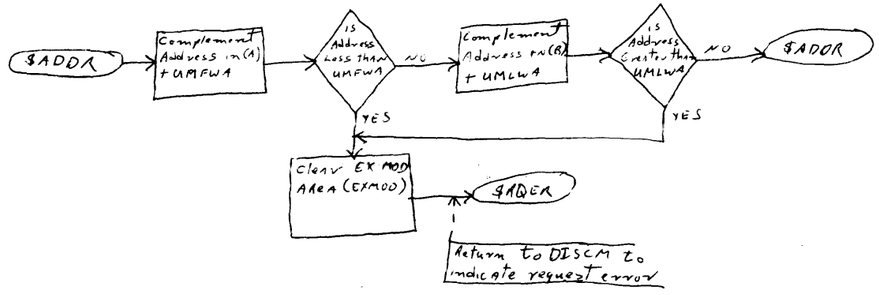
ASCII → Convert Binary to ASCII Octal or Decimal

Calling sequence: LDA <Value in binary>
JSB CNDEC/CNOCT

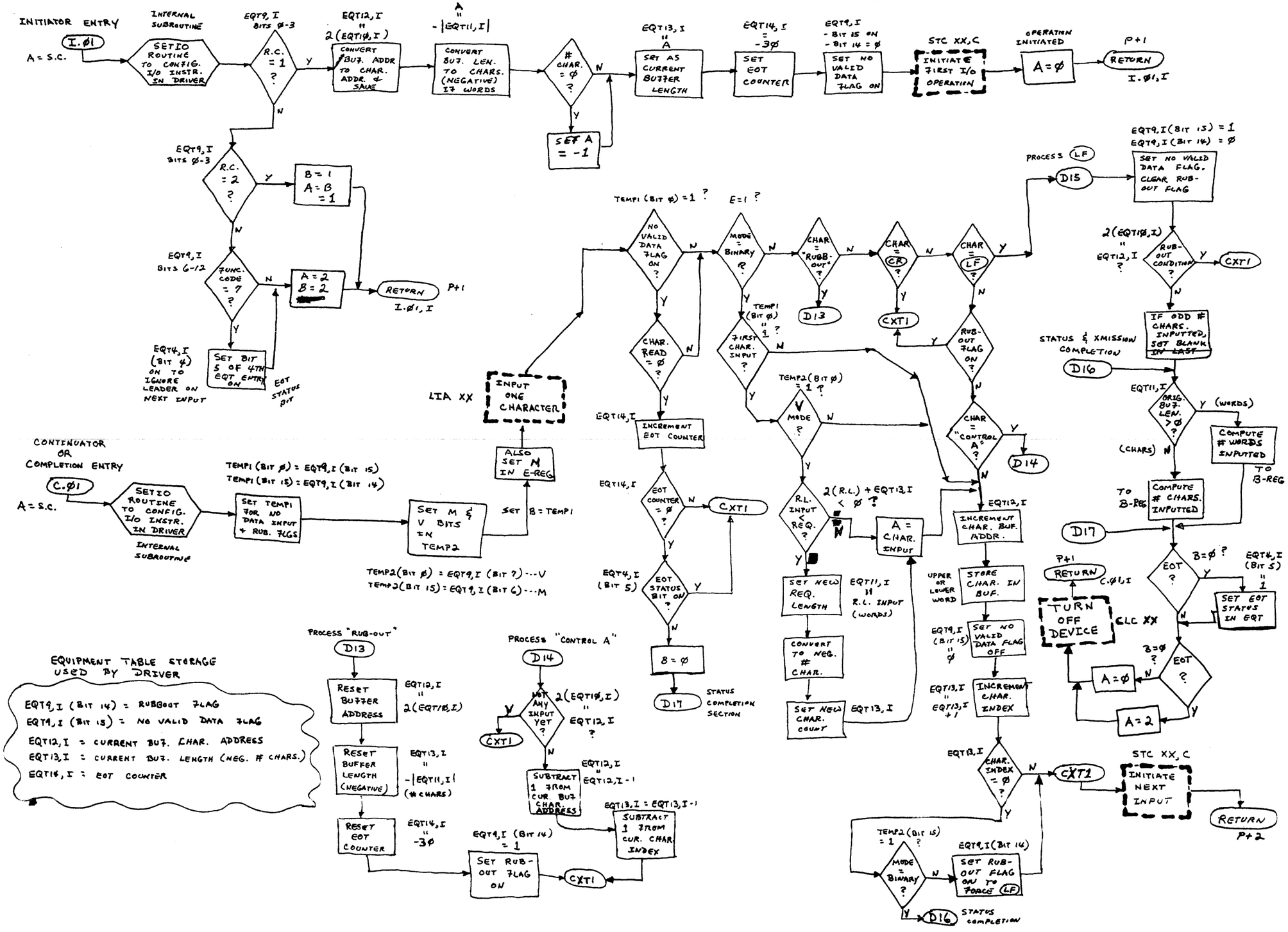
Return: (A) Least significant 2 digits
(B) Address of most significant digits



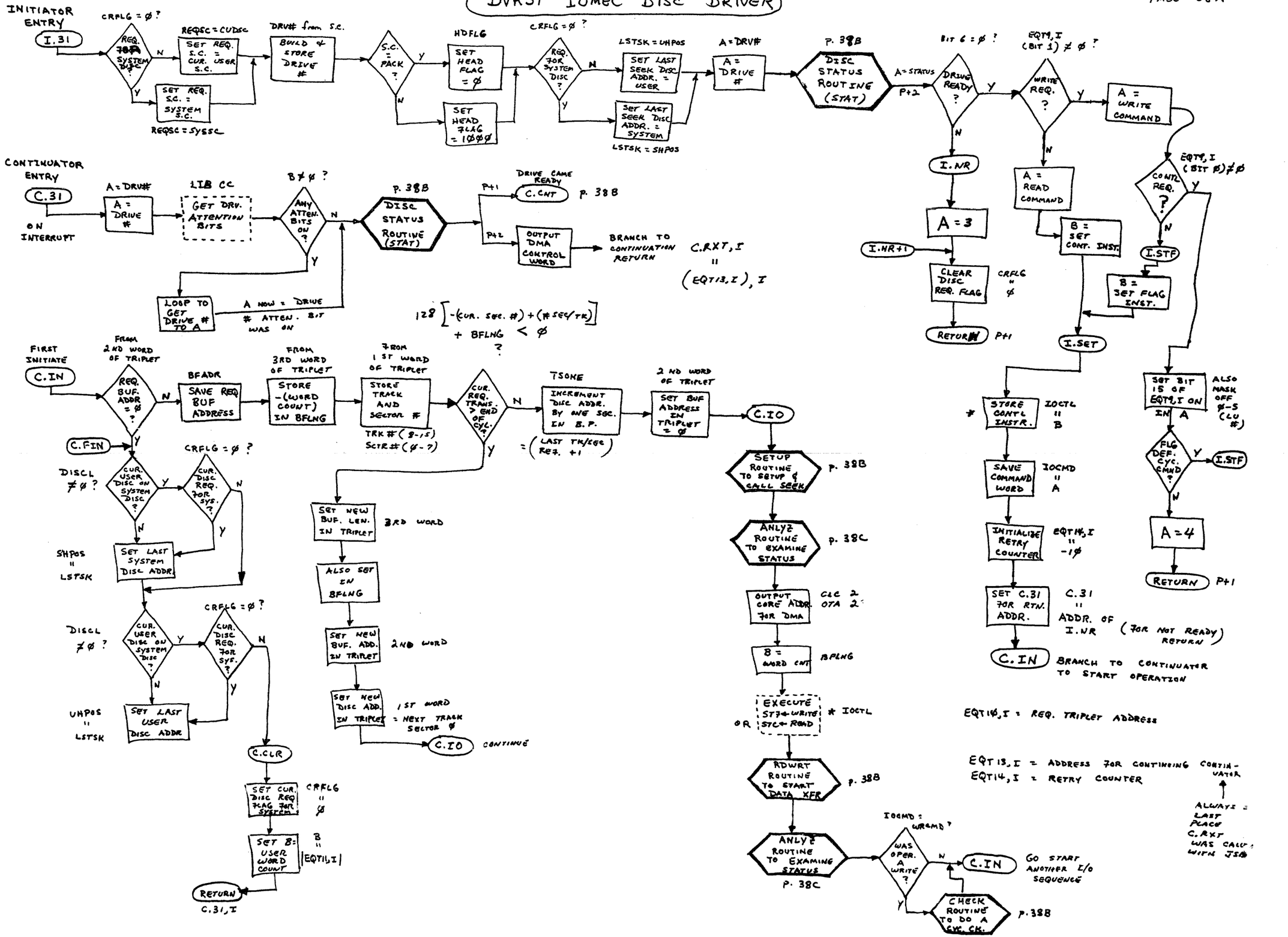
$ADDR - checks if buffer address limits are within the user Program Area limits

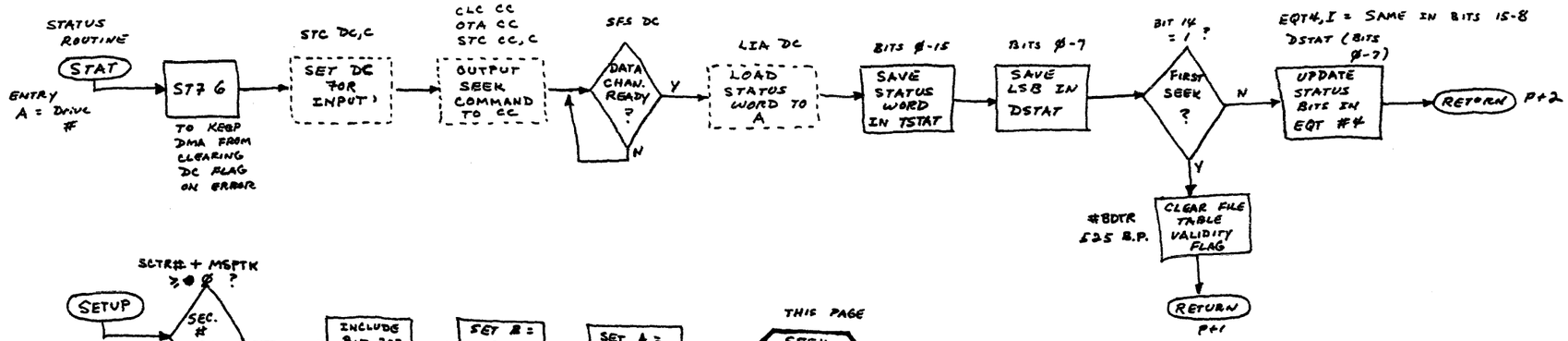


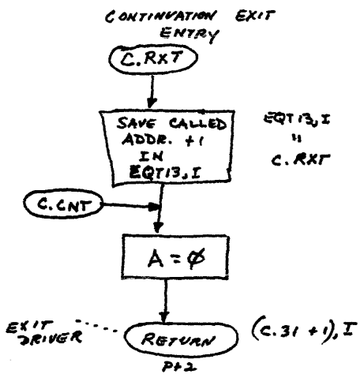
DOS/DOS-M READER DRIVER (DVR01) "A"-VERSION

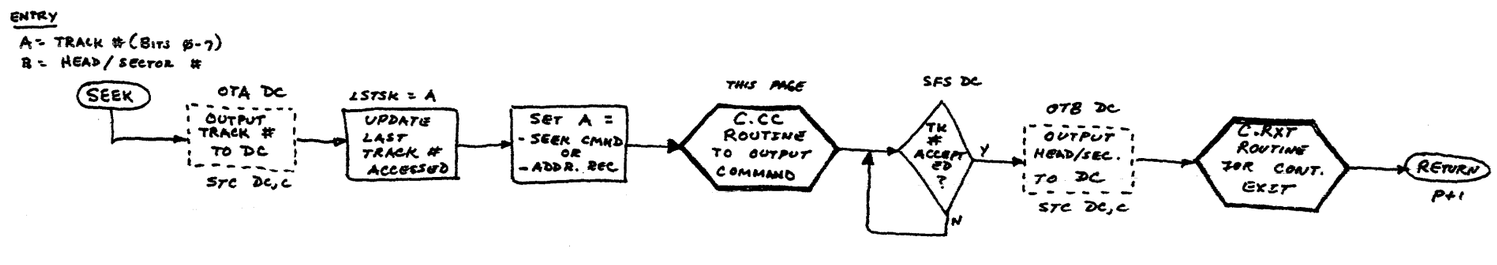


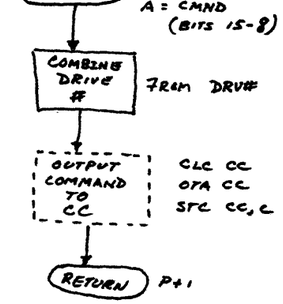
DVR31 IOMEC DISC DRIVER

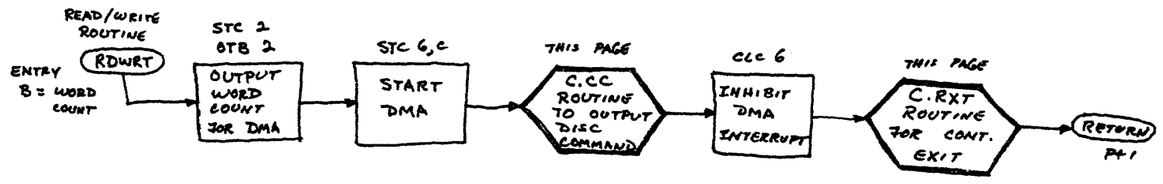


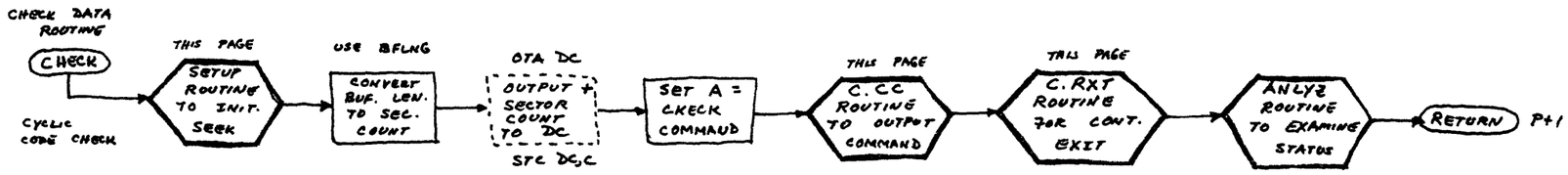

STATUS ROUTINE
STAT
ENTRY
A = DRIVE #
S77 6
TO KEEP DMA FROM CLEARING DC FLAG ON ERROR
STC DC,C
SET DC FOR INPUT
CLC CC
OTA CC
STC CC,C
OUTPUT SEEK COMMAND TO CC
SFS DC
DATA CHAN. READY ?
Y
N
LIA DC
LOAD STATUS WORD TO A
BITS Ø-15
SAVE STATUS WORD IN TSTAT
BITS Ø-7
SAVE LSB IN DSTAT
BIT 14 = 1 ?
FIRST SEEK ?
N
Y
EQT4,I = SAME IN BITS 15-8
DSTAT (BITS Ø-7)
UPDATE STATUS BITS IN EQT #4
RETURN
P+2
#BDTR
525 B.P.
CLEAR FILE TABLE VALIDITY FLAG
RETURN
P+1
SETUP
SECTR# + MSPTR ≥ Ø ?
SEC. # Ø > 11 ?
Y
SET BOTTOM HEAD BIT
INCLUDE BIT FOR S.C. HEAD #
SET B = CORRECT HEAD/SEC
SET A = TRACK # IN BITS Ø-7
THIS PAGE
SEEK ROUTINE TO INITIATE SEEK
RETURN
P+1

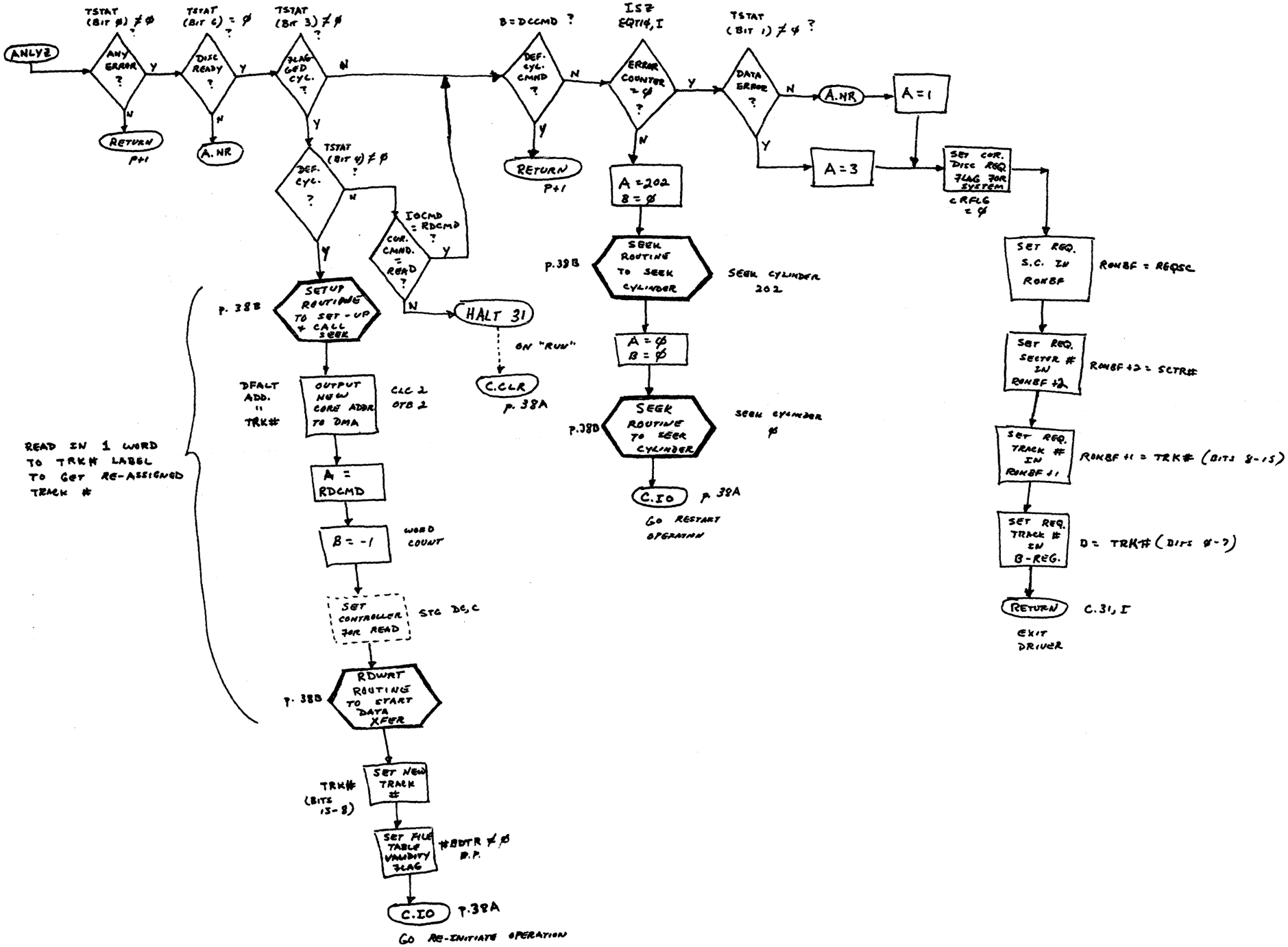

CONTINUATION EXIT ENTRY
C.RXT
SAVE CALLED ADDR. +1 IN EQT13,I
EQT13,I = C.RXT
C.CNT
A = Ø
EXIT DRIVER
RETURN
P+2
(C.31 +1),I

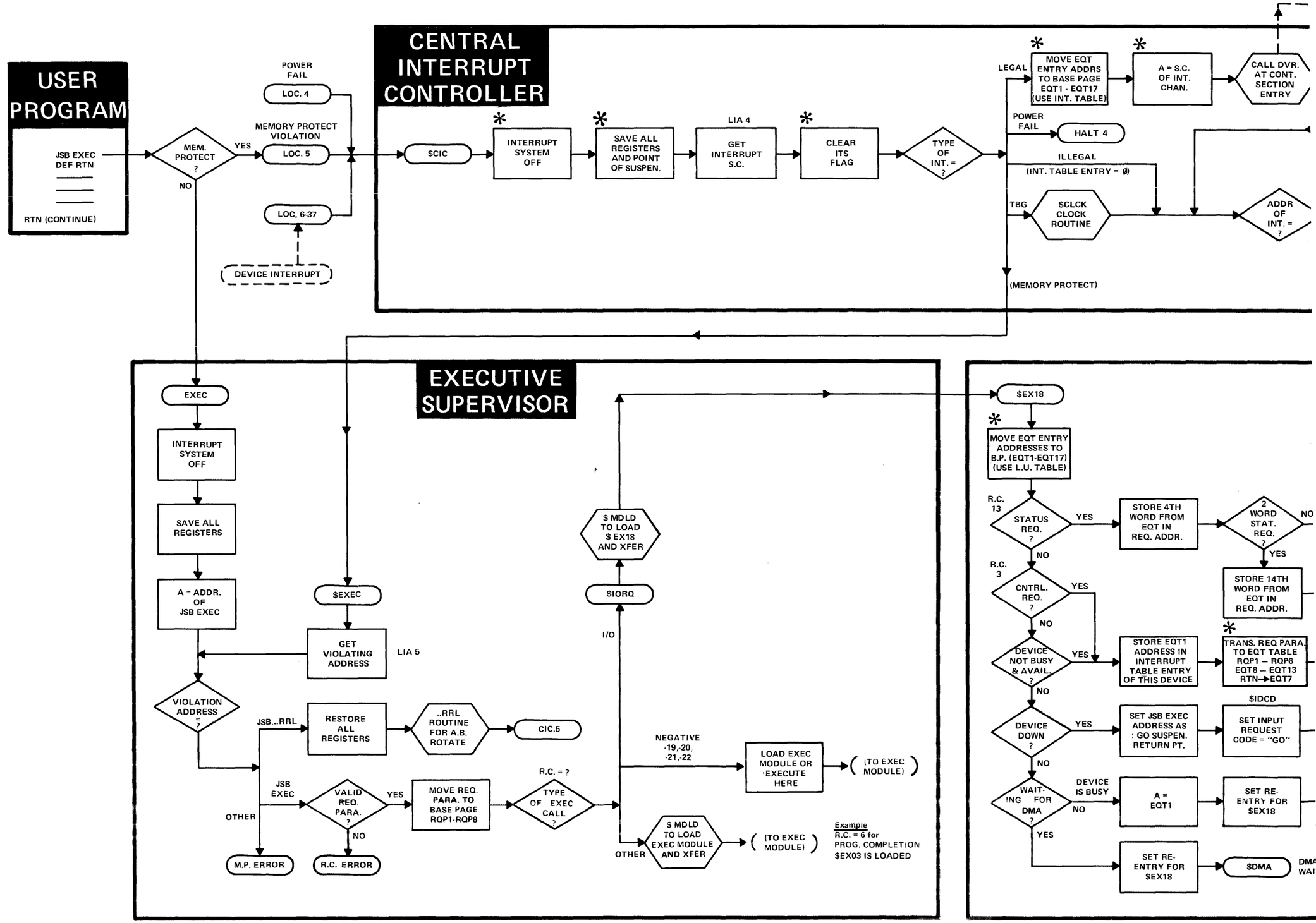

ENTRY
A = TRACK # (BITS Ø-7)
B = HEAD/SECTOR #
SEEK
OTA DC
OUTPUT TRACK # TO DC
STC DC,C
LSTSK = A
UPDATE LAST TRACK # ACCESSED
SET A = - SEEK CMND OR - ADDR. REC
THIS PAGE
C.CC ROUTINE TO OUTPUT COMMAND
SFS DC
TK # ACCEPTED ?
Y
N
OTB DC
OUTPUT HEAD/SEC. TO DC
STC DC,C
C.RXT ROUTINE FOR CONT. EXIT
RETURN
P+1


OUTPUT DISC COMMAND
C.CC
ENTRY
A = CMND (BITS 15-8)
COMBINE DRIVE #
FROM DRV#
OUTPUT COMMAND TO CC
CLC CC
OTA CC
STC CC,C
RETURN
P+1


READ/WRITE ROUTINE
RDWRT
ENTRY
B = WORD COUNT
STC 2
OTB 2
OUTPUT WORD COUNT FOR DMA
STC 6,C
START DMA
THIS PAGE
C.CC ROUTINE TO OUTPUT DISC COMMAND
CLC 6
INHIBIT DMA INTERRUPT
THIS PAGE
C.RXT ROUTINE FOR CONT. EXIT
RETURN
P+1


CHECK DATA ROUTINE
CHECK
Cyclic code check
THIS PAGE
SETUP ROUTINE TO INIT. SEEK
USE BFLNG
CONVERT BUF. LEN. TO SEC. COUNT
OTA DC
OUTPUT + SECTOR COUNT TO DC
STC DC,C
SET A = CHECK COMMAUD
THIS PAGE
C.CC ROUTINE TO OUTPUT COMMAND
THIS PAGE
C.RXT ROUTINE FOR CONT. EXIT
ANLYZ ROUTINE TO EXAMINE STATUS
RETURN
P+1


ANLYZ
TSTAT (BIT 0) ≠ 0 ?
ANY ERROR ?
Y
N
RETURN P+1
TSTAT (BIT 6) = 0 ?
DISC READY ?
Y
N
A.NR
TSTAT (BIT 3) ≠ 0 ?
FLAGGED CYL. ?
N
Y
TSTAT (BIT 4) ≠ 0 ?
DEF. CYL. ?
N
Y
IOCMD = RDCMD ?
COR. CMND. = READ ?
Y
N
HALT 31
ON "RUN"
C.CLR
p. 38A
p. 38B
SETUP ROUTINE TO SET-UP & CALL SEEK
DFALT ADD. = TRK#
OUTPUT NEW CORE ADDR TO DMA
CLC 2
OTB 2
A = RDCMD
B = -1
WORD COUNT
SET CONTROLLER FOR READ
STC DC,C
RDWRT ROUTINE TO START DATA XFER
p. 38D
SET NEW TRACK #
TRK# (BITS 15-8)
SET FILE TABLE VALIDITY FLAG
#BDTR ≠ 0 B.P.
C.IO
p. 38A
Go RE-INITIATE OPERATION
READ IN 1 WORD TO TRK# LABEL TO GET RE-ASSIGNED TRACK #
B = DCCMD ?
DEF. CYL. CMND ?
Y
N
RETURN P+1
ISZ EQT14,I
ERROR COUNTER = 0 ?
Y
N
A = 202
B = 0
SEEK ROUTINE TO SEEK CYLINDER
p. 38B
SEEK CYLINDER 202
A = 0
B = 0
SEEK ROUTINE TO SEEK CYLINDER
p. 38D
SEEK CYLINDER 0
C.IO
p. 38A
Go RESTART OPERATION
TSTAT (BIT 1) ≠ 0 ?
DATA ERROR ?
N
Y
A.NR
A = 1
A = 3
SET COR. DISC REQ FLAG FOR SYSTEM
CRFLG = 0
SET REQ. S.C. IN RQNBF
RQNBF = REQSC
SET REQ. SECTOR # IN RQNBF+2
RQNBF+2 = SCTR#
SET REQ. TRACK # IN RQNBF+1
RQNBF+1 = TRK# (BITS 8-15)
SET REQ. TRACK # IN B-REG.
B = TRK# (BITS 8-7)
RETURN
C.31, I
EXIT DRIVER


USER PROGRAM
JSB EXEC DEF RTN
RTN (CONTINUE)
MEM. PROTECT ?
YES
NO
POWER FAIL
LOC. 4
MEMORY PROTECT VIOLATION
LOC. 5
LOC. 6-37
DEVICE INTERRUPT
CENTRAL INTERRUPT CONTROLLER
$CIC
INTERRUPT SYSTEM OFF
SAVE ALL REGISTERS AND POINT OF SUSPEN.
LIA 4
GET INTERRUPT S.C.
CLEAR ITS FLAG
TYPE OF INT. = ?
LEGAL
MOVE EQT ENTRY ADDRS TO BASE PAGE EQT1 - EQT17 (USE INT. TABLE)
A = S.C. OF INT. CHAN.
CALL DVR. AT CONT. SECTION ENTRY
POWER FAIL
HALT 4
ILLEGAL
(INT. TABLE ENTRY = 0)
TBG
$CLCK CLOCK ROUTINE
ADDR OF INT. = ?
(MEMORY PROTECT)
EXECUTIVE SUPERVISOR
EXEC
INTERRUPT SYSTEM OFF
SAVE ALL REGISTERS
A = ADDR. OF JSB EXEC
$EXEC
GET VIOLATING ADDRESS
LIA 5
VIOLATION ADDRESS = ?
JSB..RRL
RESTORE ALL REGISTERS
..RRL ROUTINE FOR A.B. ROTATE
CIC.5
JSB EXEC
OTHER
VALID REQ. PARA. ?
YES
NO
MOVE REQ. PARA. TO BASE PAGE RQP1-RQP8
R.C. = ?
TYPE OF EXEC CALL ?
M.P. ERROR
R.C. ERROR
$ MDLD TO LOAD $ EX18 AND XFER
$IORQ
I/O
NEGATIVE -19,-20, -21,-22
LOAD EXEC MODULE OR EXECUTE HERE
(TO EXEC MODULE)
$ MDLD TO LOAD EXEC MODULE AND XFER
OTHER
(TO EXEC MODULE)
Example
R.C. = 6 for PROG. COMPLETION $EX03 IS LOADED
$EX18
MOVE EQT ENTRY ADDRESSES TO B.P. (EQT1-EQT17) (USE L.U. TABLE)
R.C. 13
STATUS REQ. ?
YES
NO
STORE 4TH WORD FROM EQT IN REQ. ADDR.
2 WORD STAT. REQ. ?
NO
YES
STORE 14TH WORD FROM EQT IN REQ. ADDR.
R.C. 3
CNTRL. REQ. ?
YES
NO
DEVICE NOT BUSY & AVAIL. ?
YES
NO
STORE EQT1 ADDRESS IN INTERRUPT TABLE ENTRY OF THIS DEVICE
TRANS. REQ PARA. TO EQT TABLE RQP1 – RQP6 EQT8 – EQT13 RTN→EQT7
$IDCD
DEVICE DOWN ?
YES
NO
SET JSB EXEC ADDRESS AS : GO SUSPEN. RETURN PT.
SET INPUT REQUEST CODE = "GO"
WAITING FOR DMA ?
DEVICE IS BUSY
NO
YES
A = EQT1
SET RE-ENTRY FOR $EX18
SET RE-ENTRY FOR $EX18
$DMA
DMA WAI

CENTRAL INTERRUPT CONTROLLER

$CIC → * INTERRUPT SYSTEM OFF → * SAVE ALL REGISTERS AND POINT OF SUSPEN. → LIA 4 GET INTERRUPT S.C. → * CLEAR ITS FLAG → TYPE OF INT. = ?

TYPE OF INT. = ?
- POWER FAIL → HALT 4
- ILLEGAL (INT. TABLE ENTRY = 0) → (MEMORY PROTECT)
- TBG → $CLCK CLOCK ROUTINE
- LEGAL → * MOVE EQT ENTRY ADDRS TO BASE PAGE EQT1 - EQT17 (USE INT. TABLE) → * A = S.C. OF INT. CHAN. → CALL DVR. AT CONT. SECTION ENTRY

$CLCK CLOCK ROUTINE → ADDR OF INT. = ?

ADDR OF INT. = ?
- SYSTEM AREA → RESTORE ALL REGISTERS → RETURN → JMP SCIC,I
- USER AREA → RESTORE ALL REGISTERS → CIC.5 → ENABLE MP & INTERRUPT SYSTEM → RETURN

CALL DVR. AT CONT. SECTION ENTRY → CONTINUATION P+2 → OPER. ATT. ON ? (YES / NO) → OPER. ATT. ALLOWED ? (YES / NO) → REQ. INPUT FROM SYSTEM TTY K.B. → $TYPE

COMPLETION P+1 (TO COMPLETION SECTION)

EXECUTIVE SUPERVISOR

$EXEC → GET VIOLATING ADDRESS (LIA 5) → RESTORE ALL REGISTERS → ..RRL ROUTINE FOR A.B. ROTATE → CIC.5

VALID REQ. PARA. ? (YES / NO)
- YES → MOVE REQ. PARA. TO BASE PAGE RQP1-RQP8 → R.C. = ? TYPE OF EXEC CALL ?
- NO → R.C. ERROR

R.C. = ? TYPE OF EXEC CALL ?
- NEGATIVE -19,-20, -21,-22 → LOAD EXEC MODULE OR EXECUTE HERE → (TO EXEC MODULE)
- OTHER → $ MDLD TO LOAD EXEC MODULE AND XFER → (TO EXEC MODULE)

Example
R.C. = 6 for PROG. COMPLETION
$EX03 IS LOADED

I/O → $IORQ → $ MDLD TO LOAD $ EX18 AND XFER → $EX18

INPUT/OUTPUT CONTROL EXECUTIVE MODULE

$EX18 → * MOVE EQT ENTRY ADDRESSES TO B.P. (EQT1-EQT17) (USE L.U. TABLE)

R.C. 13 — STATUS REQ. ? (YES / NO)
- YES → STORE 4TH WORD FROM EQT IN REQ. ADDR. → 2 WORD STAT. REQ. ? (NO / YES)
  - NO → ENABLE MP & INTERRUPT SYSTEM → REQ. RETURN
  - YES → STORE 14TH WORD FROM EQT IN REQ. ADDR. → ENABLE MP & INTERRUPT SYSTEM → REQ. RETURN

R.C. 3 — CNTRL. REQ. ? (YES / NO)
- YES → STORE EQT1 ADDRESS IN INTERRUPT TABLE ENTRY OF THIS DEVICE

DEVICE NOT BUSY & AVAIL. ? (YES / NO)
- YES → * TRANS. REQ PARA. TO EQT TABLE RQP1 – RQP6 EQT8 – EQT13 RTN→EQT7 → $LDVR READ I/O DRIVER TO CORE IF DISC RES. → DRIVR ROUTINE TO CALL I/O DRIVER INITIATOR

DRIVR ROUTINE TO CALL I/O DRIVER INITIATOR
- P + 1 (A = 0) — OPERATION INITIATED → I/O WITHOUT WAIT REQ. ? (YES / NO)
  - YES → ENABLE MP & INTERRUPT SYSTEM → REQ. RETURN
  - NO → SYSTEM IDLE LOOP $WAIT
- P + 2 (A ≠ 0) → HANDLE REJECTS, MALFUNCTIONS IMMEDIATE COMPLETION NO DMA AVAILABLE

DEVICE DOWN ? (YES / NO)
- YES → SET JSB EXEC ADDRESS AS : GO SUSPEN. RETURN PT. → $IDCD SET INPUT REQUEST CODE = "GO" → $TYPE REQUEST INPUT FROM SYSTEM TTY

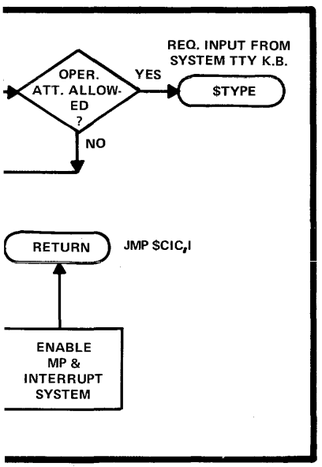
WAITING FOR DMA ? (DEVICE IS BUSY / NO)
- DEVICE IS BUSY → A = EQT1 → SET RE-ENTRY FOR $EX18 → $BLOF DEVICE BUSY WAIT LOOP
- YES → SET RE-ENTRY FOR $EX18 → $DMA DMA BUSY WAIT LOOP

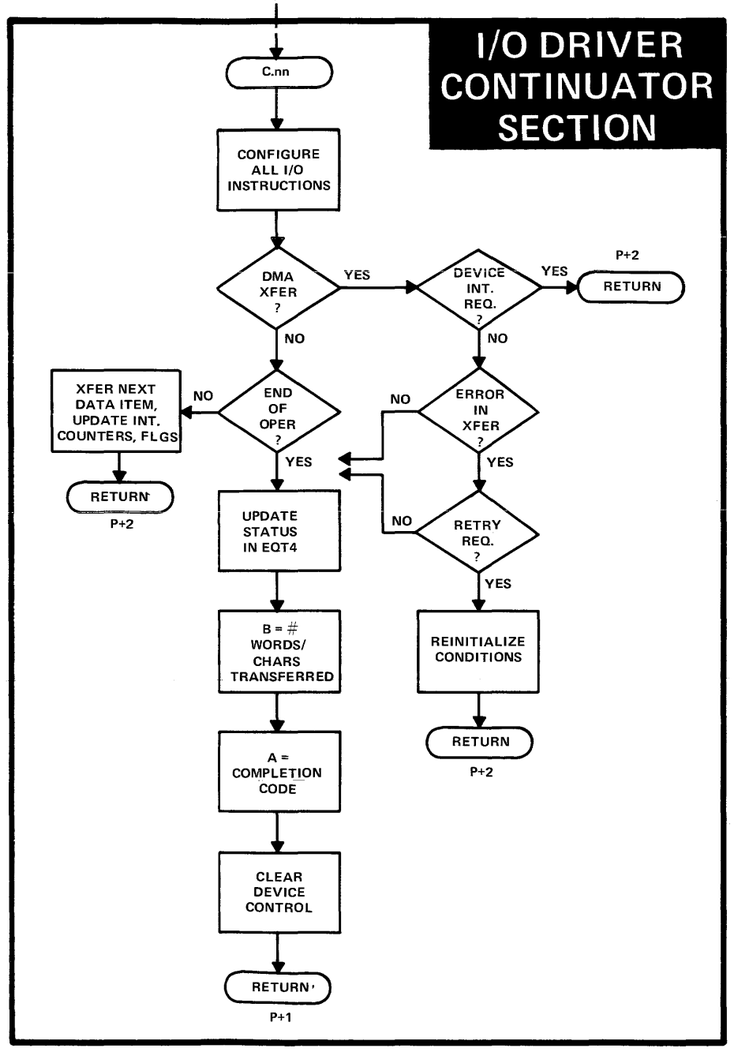
DRIVR → $DMAX

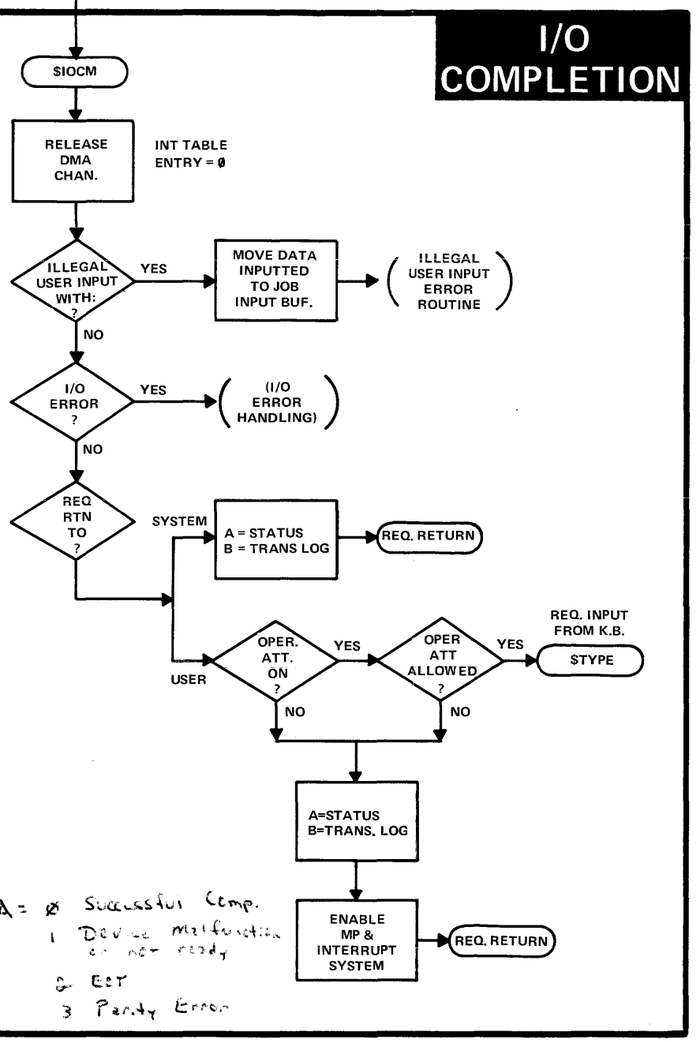
XFER NEXT DATA ITEM, UPDATE INT. COUNTERS, FLGS (NO) → RETURN P+2

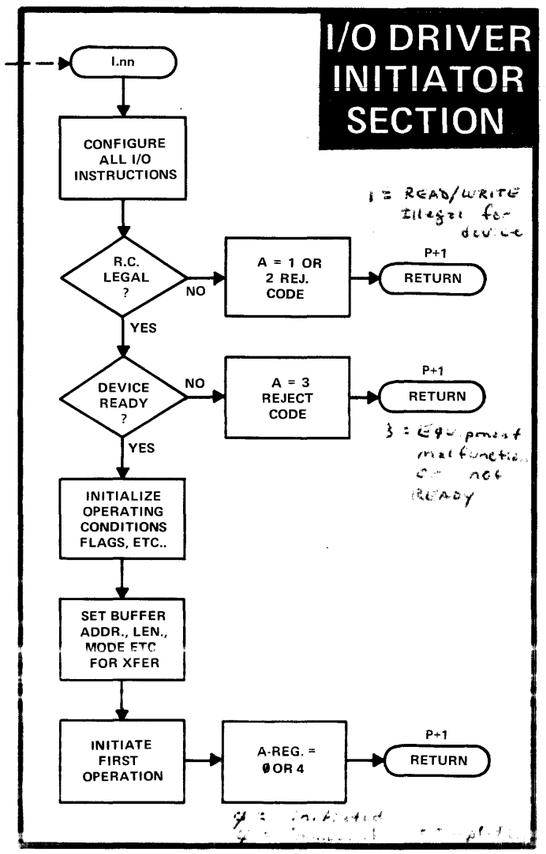
DOS-M I/O REQUEST PROCESSING

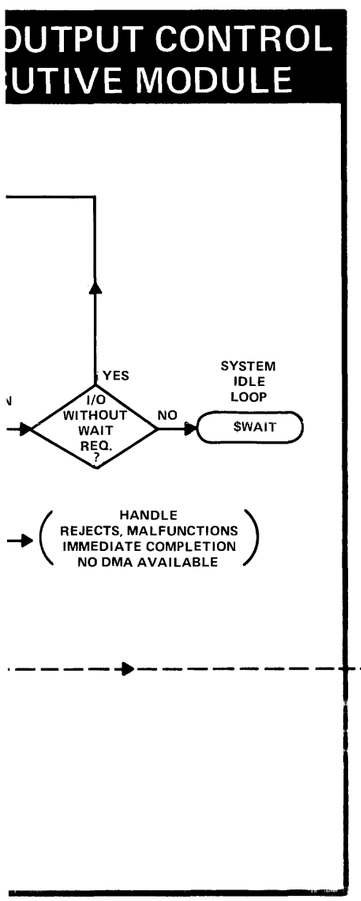
* INDICATES FUNCTIONS SYSTEM DOES FOR DRIVER

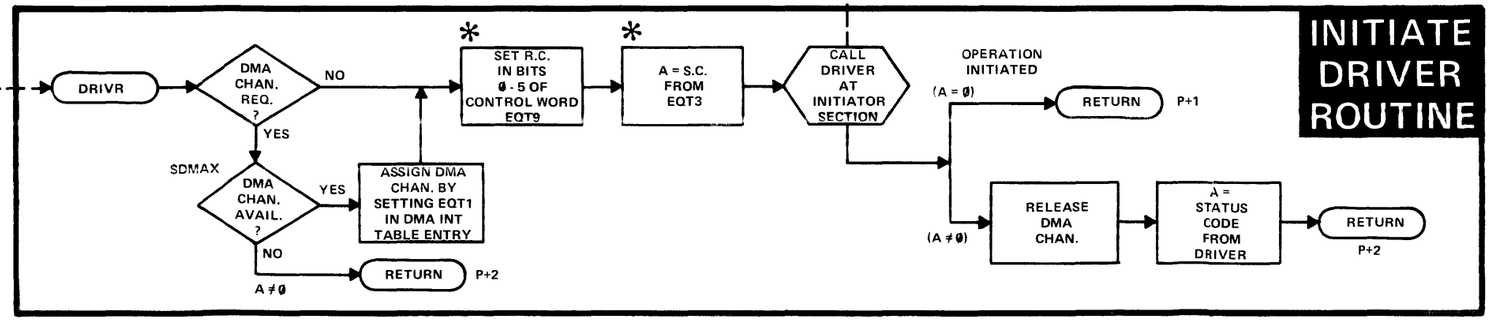












DOS-M OPERATIONAL LAB PROBLEM

PURPOSE:

To acquaint user that is already familiar with operation of Fixed-Head DOS System with operation of Moving-Head DOS-M System.

TASKS TO PERFORM:

I Bootstrap system up by two different methods:

A. Using supplied bootstrap without punching a configured bootstrap. This will be a two step operation on the DOS-M System Computer.

B. Using supplied bootstrap to punch a configured bootstrap tape on a computer system different than DOS-M System Computer. Then use this tape on DOS-M System to start system.

QUESTIONS AND NOTES:

1. Review figures 14A and 14B before starting.
2. Try specifying a subchannel number not on system or an invalid one on system drive. What happens and why?
3. How could you modify the Core Resident System before it starts executing in memory once loaded from disc? Assume that it is not possible to alter the Core Resident System on the disc before bringing into core with bootstrap(s).
4. Try bringing the system up with TTY turned "OFF" or the cable disconnected. What happens and why?
5. Try bringing the system up with Disc DRIVE not ready (Power off or Cartridge UNLOCKED). What happens and why?

II Once the system is up and running, try to get around having to enter the DATE DIRECTIVE. Why will the system not let you do this?

III Use the system to do some of the operations you did with Fixed Head DOS (store source file, compile, list, edit, load, RUN, ... etc.). Use :OFF directive to bail out of some operations (like :LIST etc.).

QUESTIONS AND NOTES:

1. Why does it take longer in long Edit or Purge operations than it does in Fixed Head DOS?
2. You will be using the DEFAULT User Disc during these operations which is on the system disc.
3. Try listing the User and System Directories. What differences are noted from Fixed-Head DOS?

IV Exercise the other New Directives (:UD, :SS, :DD, :IN) that do not exist in the Fixed-Head DOS. Here it is best to first use :UD and :IN directives to label the other disc on the system.

IV Continued

A. Try labeling the other disc with your name (6 characters maximum).
B. Store various source files on both user discs using:
   (1) :UD to assign which disc (current user disc)
   (2) :ST to store source then-
C. Exercise :SS directive in Listing (:LI), Editing (:ED) and Purging (:PU) these files. Note how it is possible to store <u>DUPLICATE FILE NAMES</u> on different discs!

<u>QUESTIONS AND NOTES:</u>

1. Is it possible to purge (:PU) files that are on different User Discs with one command? If not,why? (Review :SS directive if confused)
2. Try using :DD,U directive to transfer some of your above files to the other User Disc, changing some of their names in the process. Verify results using :SS and :LI,U directives
3. If and when error messages occur with :UD and :IN directives, try to reason why according to lecture regarding labeling.
4. Try entering :PROG,JOBPR. What happens and why? How could you get out of this situation?

DOS/DOS-M MEMORY DISPLAY - MODIFY LAB PROGRAM

PROBLEM: Write a user program (and needed subroutines) that will allow the operator to perform the following tasks once loaded and started by :PROG or :RUN directive:

1. Display (in the B-REGISTER) the contents of any memory location whose address is currently in the Switch Register. This section should loop to give the operator a REAL-TIME display in B-REGISTER until switch 15 of the Switch Register is set "ON" (UP). At such time, 2 below should be entered.

2. Allow the operator to modify any memory location contents simply by entering a VALUE & ADDRESS (both in octal) via the TTY keyboard in response to a question typed out by the program (like for example, VALUE/ADDR =?).

NOTE: Following execution of 2 above, the program should go back to Section 1 above to see if Switch Register has been changed (Bit 15).

PRELIMINARY CONSIDERATIONS:

A. Exec call with Request Code of -19 may be used in step 2 above.
B. One possible method that may be used to read an octal number from the keyboard is by using K or @ format specification. Let the Fortran Formatter do the dirty work!
C. Put in some feature to "bail out" of this program and return to DOS/DOS-M system control.

TEST OUT:

A. Obviously a program such as this has extensive power to WIPE OUT THE SYSTEM. It should be O.K. to modify locations 2,3, or a trap cell that is higher than any used in the particular system being used (like location 37B).
B. Once you believe your program is faultless, use it to disable the functioning of the System Clock (Time Base Generator) feature. Also use it to display all the SYSTEM BASE PAGE constants and storage locations that we discussed in class.
C. What good is a program like this?
D. When would this program be useless?

DOS-M OPERATIONAL LAB PROBLEM

PURPOSE:

To acquaint user that is already familiar with operation of Fixed-Head DOS System with operation of Moving-Head DOS-M System.

TASKS TO PERFORM:

I Bootstrap system up by two different methods:

A. Using supplied bootstrap without punching a configured bootstrap. This will be a two step operation on the DOS-M System Computer.

B. Using supplied bootstrap to punch a configured bootstrap tape on a computer system different than DOS-M System Computer. Then use this tape on DOS-M System to start system.

QUESTIONS AND NOTES:

1. Review figures 14A and 14B before starting.
2. Try specifying a subchannel number not on system or an invalid one on system drive. What happens and why?
3. How could you modify the Core Resident System before it starts executing in memory once loaded from disc? Assume that it is not possible to alter the Core Resident System on the disc before bringing into core with bootstrap(s).
4. Try bringing the system up with TTY turned "OFF" or the cable disconnected. What happens and why?
5. Try bringing the system up with Disc DRIVE not ready (Power off or Cartridge UNLOCKED). What happens and why?

II Once the system is up and running, try to get around having to enter the DATE DIRECTIVE. Why will the system not let you do this?

III Use the system to do some of the operations you did with Fixed Head DOS (store source file, compile, list, edit, load, RUN, ... etc.). Use :OFF directive to bail out of some operations (like :LIST etc.).

QUESTIONS AND NOTES:

1. Why does it take longer in long Edit or Purge operations than it does in Fixed Head DOS?
2. You will be using the DEFAULT User Disc during these operations which is on the system disc.
3. Try listing the User and System Directories. What differences are noted from Fixed-Head DOS?

IV Exercise the other New Directives (:UD, :SS, :DD, :IN) that do not exist in the Fixed-Head DOS. Here it is best to first use :UD and :IN directives to label the other disc on the system.

IV Continued

A. Try labeling the other disc with your name (6 characters maximum).
B. Store various source files on both user discs using:
   (1) :UD to assign which disc (current user disc)
   (2) :ST to store source then-
C. Exercise :SS directive in Listing (:LI), Editing (:ED) and Purging (:PU) these files. Note how it is possible to store DUPLICATE FILE NAMES on different discs!

QUESTIONS AND NOTES:

1. Is it possible to purge (:PU) files that are on different User Discs with one command? If not,why? (Review :SS directive if confused)
2. Try using :DD,U directive to transfer some of your above files to the other User Disc, changing some of their names in the process. Verify results using :SS and :LI,U directives
3. If and when error messages occur with :UD and :IN directives, try to reason why according to lecture regarding labeling.
4. Try entering :PROG,JOBPR. What happens and why? How could you get out of this situation?

DOS/DOS-M MEMORY DISPLAY - MODIFY LAB PROGRAM

PROBLEM: Write a user program (and needed subroutines) that will allow the operator to perform the following tasks once loaded and started by :PROG or :RUN directive:

1. Display (in the B-REGISTER) the contents of any memory location whose address is currently in the Switch Register. This section should loop to give the operator a REAL-TIME display in B-REGISTER until switch 15 of the Switch Register is set "ON" (UP). At such time, 2 below should be entered.

2. Allow the operator to modify any memory location contents simply by entering a VALUE & ADDRESS (both in octal) via the TTY keyboard in response to a question typed out by the program (like for example, VALUE/ADDR =?).

NOTE: Following execution of 2 above, the program should go back to Section 1 above to see if Switch Register has been changed (Bit 15).

PRELIMINARY CONSIDERATIONS:

A. Exec call with Request Code of -19 may be used in step 2 above.
B. One possible method that may be used to read an octal number from the keyboard is by using K or @ format specification. Let the Fortran Formatter do the dirty work!
C. Put in some feature to "bail out" of this program and return to DOS/DOS-M system control.

TEST OUT:

A. Obviously a program such as this has extensive power to WIPE OUT THE SYSTEM. It should be O.K. to modify locations 2,3, or a trap cell that is higher than any used in the particular system being used (like location 37B).
B. Once you believe your program is faultless, use it to disable the functioning of the System Clock (Time Base Generator) feature. Also use it to display all the SYSTEM BASE PAGE constants and storage locations that we discussed in class.
C. What good is a program like this?
D. When would this program be useless?

MODIFY / DISPLAY LAB PROBLEM

21'6

```
FTN,L
      PROGRAM LAB7
  100 CALL DSPLY(IDUM)
C
      WRITE(1,101)
  101 FORMAT(/,"VAL/ADDR ?  ←")
      READ(1,*) IV,IA
      IF(IA-0) 500,500,300
C
  300 CALL MODFY(IV,IA)
      GO TO 100
C
  500 CALL EXEC(6)
      END
```

```
 PAGE 0002 #01

0001                    ASMB,L
0002  00000                     NAM DSPLY,6
0003                            ENT DSPLY
0004                            EXT .ENTR
0005*
0006  00000 000000  IDUM  NOP
0007  00001 000000  DSPLY NOP
0008  00002 016001X       JSB .ENTR       LINK
0009  00003 000000R       DEF IDUM           PARAMETERS
0010*
0011  00004 102501  LOOP  LIA 1
0012  00005 002020       SSA
0013  00006 126001R       JMP DSPLY,I    .....EXIT SW 15 IS UP....
0014  00007 164000       LDB 0,I
0015  00010 026004R       JMP LOOP
0016*
0017                            END
**   NO ERRORS*
```

```
 PAGE 0002 #01

0001                    ASMB,L
0002  00000                     NAM MODFY,6
0003                            ENT MODFY
0004                            EXT .ENTR,EXEC
0005*
0006  00000 000000  IV     NOP
0007  00001 000000  IA     NOP
0008  00002 000000  MODFY NOP
0009  00003 016001X       JSB .ENTR
0010  00004 000000R       DEF IV
0011*
0012  00005 162000R       LDA IV,I
0013  00006 166001R       LDB IA,I
0014*
0015  00007 016002X       JSB EXEC
0016  00010 000012R       DEF *+2
0017  00011 000013R       DEF RCODE
0018  00012 126002R       JMP MODFY,I
0019*
0020  00013 177755  RCODE DEC -19
0021                            END
**   NO ERRORS*
```

```
0001   FTN.I
0002         PROGRAM LAB7
0003     100 CALL DSPLY(IDUM.ITEST)
0004         IF(ITEST-1) 900.900.800
0005   C
0006     800 WRITE(1.801) IDUM
0007     801 FORMAT(K20)
0008         GO TO 100
0009   C
0010     900 WRITE(1.101)
0011     101 FORMAT(/."VAL/ADDR ?  ←")
0012         READ(1.*) IV.IA
0013         IF(IA-0) 500.500.300
0014   C
0015     300 CALL MODFY(IV.IA)
0016         GO TO 100
0017   C
0018     500 CALL EXEC(6)
0019         END
0020         END$
**** ITST END ****
```

DISPLAY / MODIFY
FOR 2114

IOMEC DISC MODIFY LAB PROBLEM

PROBLEM: Write a STANDALONE IOMEC Disc Modify program that will enable a competent user to observe and modify the contents of any sector within any track on any disc (fixed or removable) in a maximum of up to four drive disc system by following the instructions below.

OPERATING INSTRUCTIONS FOR USING THIS PROGRAM

1. Load absolute tape using paper tape binary loader. Turn Disc Protect Override Switch "ON".
2. Initialize by performing the following:
   A. LOAD ADDRESS 2.
   B. Set Switch Register as follows:

SWITCH #'s	ACTION OR CONTENT
3-7	Select Code of Data Channel.
Ø-2	Subchannel number.

   C. Press "PRESET" and "RUN". If 1Ø2Ø77 Halt follows proceed to 3 below, else restart at B above.
3. Use Step 4 below to READ or WRITE one sector from or to the Disc selected in 2 above. The 128 word buffer (starting at location 1ØØ octal in memory) may be examined, modified, and written back onto the disc as desired. To restart Step 4 following buffer display or modify, load address 3ØØ octal. To reinitialize (i.e. change disc subchannel #, etc..) restart at step 2 above.
4. Disc Read/Write Step
   A. Set Switch Register as follows:

SWITCH #	ACTION OR CONTENT
15	"UP" to write one sector on Disc "DOWN" to read one sector from Disc.
6-13	Track # (Ø-312 octal).
Ø-4	Sector # (Ø-27 octal).

4. Disc Read/Write Step (continued)

   B. Press "RUN" for Read/Write to occur. If 102077 HALT follows the operation was performed successfully. HALT 102011 means an error occurred (the DISC PROTECT OVERRIDE SWITCH being "OFF" can cause this). Press "RUN" for automatic retry of the same operation following 102011 HALT.

PRELIMINARY CONSIDERATIONS

1. Follow Disc Command sequences of slides 69A-69F.
2. Write configuration section so that it may be used again without reloading the program.
3. Declare Read/Write Buffer starting at 100 octal.
4. Turn off interrupt system to avoid trap cell execution etc...
5. Write Disc Read/Write section such that previous operation may be retried on "RUN" following 102011 error HALT.

NOTES:

1. Applications for this program include:
   A. Patching protected areas of System Disc to correct known bugs or incorrect System Generation.
   B. Purging portions of Protected System Directory.
   C. Modifying System Generated I/O Tables for a completely different environment.
   D. Simple file modification without re-entering data or using Edit Directive etc...
2. Try some of the above applications once you are positive your program is debugged.
3. How many other useful applications can you think of?

Section II—Disk Cartridge Handling and Operation

2.1 GENERAL

Magnetic disk technology provides computer users the advantages of random recovery of data, and overall increases in speed and efficiency in information storage.

The disk is a precision instrument; thus, it requires more careful handling than other media such as tape. This is most clearly seen in its relationship with the Disk Cartridge Drive. The read/write heads on the Disk Drive float above the disk surfaces at 80 to 125 millionths of an inch. There is no actual physical contact between the heads and the disk. This means that any deviation from an ultra-flat, uniform disk (or any particle larger than about 100 millionths of an inch, which could be a cigarette ash or any other kind of contaminant) could cause head-to-disk interference and possible damage to the Disk Drive and disk.

Care must be exercised to avoid contaminants entering the disk cartridge and to prevent a bent disk due to mishandling. To achieve maximum usage of a magnetic disk, it is necessary to become familiar with the proper methods of handling and operation. The life of the cartridge and disk can be extended indefinitely by observing the following procedures:

- Replace cracked, chipped or defective cartridges.

- Clean the cartridge periodically to remove dust and lint from the exterior of the housing, using

a soft, lint-free cloth dampened with 91% isopropyl alcohol. CAUTION: Do not use medicinal isopropyl alcohols from a drug store. They often contain additives harmful to the disk cartridge and Disk Drive.

- A disk suspected of being damaged should be removed from operation until it can be inspected.

- Keep all foods, beverages and objects off the drive, and away from the disk cartridge. Any of these items can cause permanent damage to either the disk or the Disk Drive, or both.

- If a cartridge has been dropped or is visibly damaged, do not put it into operation.

- When a cartridge is not in use and the Drive is inoperable, remove the cartridge and close the head access door on the cartridge to ensure that dust and contaminants do not enter it.

- Store the cartridge in an environment of 60° to 90° F with a relative humidity of 10 to 80 per cent.

- Do not store disk cartridges close to intense magnetic fields.

- Use a storage cabinet made of fire-resistent material with a metal door. The cabinet should be kept clean and free of dirt and other contaminants.

CAUTION

Improper handling of the disk cartridge not only can cause disk damage but can cause extensive damage to the Disk Drive.

2.2 ACCLIMATIZATION

The disks are made of precisely machined aluminum with a magnetic oxide coating. They will expand and contract with significant changes in temperature. If the ambient temperature at the point-of-use is less than 60° or more than 90°, the disk must be conditioned to the point-of-use temperature for a minimum of two hours before mounting it on the Drive. This prevents loss of data due to a shift in track location.

CAUTION

Disk cartridges must be acclimatized to room temperature for a minimum of two hours before being placed on the Disk Drive to prevent drive damage.

2.3 DISK CARTRIDGE LOADING

The procedure for loading the disk cartridge is:

a. Open the tinted cover of the Disk Cartridge Drive to obtain access to the cartridge receiver handle.

NOTE

The cartridge receiver handle will not be operable until 30 seconds after the Disk

		BIT 4	BIT 3	BIT Ø
OVERIDE SWITCH ON PROTECT BIT SET				
	Read	0	1	0
	Write	0	1	0
	Check Data	0	1	0
	Initialize Data	0	0	0
DEFECTIVE BIT SET				
	Read	1	1	1
	Write	1	1	1
	Check Data	1	1	1
	Initialize Data	0	0	0
OVERIDE SWITCH OFF PROTECT BIT SET				
	Read	0	1	0
	Write	0	1	1
	Check Data	0	1	0
	Initialize Data	0	1	1
DEFECTIVE BIT SET				
	Read	1	1	1
	Write	1	1	1
	Check Data	1	1	1
	Initialize Data	0	1	1

BIT 4 - ADDRESS ERROR

BIT 3 - PROTECT/DEFECTIVE

BIT Ø - ERROR

FIGURE 17.

Bill Williams

HEWLETT hp PACKARD

CUPERTINO DIVISION • 11000 Wolfe Road, Cupertino, California 95014, Telephone 408-257-7000,

Larry Walsh

Distribution

January 5, 1971

Enclosed are notes and diagrams describing interesting facts about the HP 2870/1 Disc Controller and Drive (IOMEC). I hope they are useful to you.

Please notify me if you discover errors or if you are able to supply additional facts.

Regards,

larry walsh

BASIC IOMEC DISC DESCRIPTION

The basic physical and operational specifications of the IOMEC disc are well known and not repeated here. A few items of interest are, however, discussed.

The advantage of using a system with one fixed and one removable disc is shown in Figure 1. The fixed disc is used for storage of "system" programs and of the non-changing data base. The removable disc is then used to provide an unlimited file and program capability. There are two alternatives to this configuration:

A. Non-removable discs allow only limited storage capability - expansion of disc capacity can be accomplished only through purchase of more drives.

B. Removable discs allow unlimited storage capacity, but require that every disc also contain the non-changing data base and"system" programs (see Figure 2).

Inspection of a disc pack shows a metal ring on the outside and at the bottom. The thin slots cut into this ring are physical indications of sector boundaries. All of the slots are equi-spaced from adjacent slots, except for two - these two are close together and designate the start-of-track boundary. The disc drive senses these slot markers and is thus able to physically determine which sector is currently passing beneath the heads.

A rectangular hole, covered by a spring loaded metal plate, can be seen on the bottom of a disc cartridge. When the cartridge is loaded, filtered air is forced into this opening, thus providing a positive pressure inside the cartridge. Thus, only filtered air is allowed into the cartridge and no dust particles can drift in through the opening for the heads.

Figure 3 shows an IOMEC disc cross-section. Since data is recorded at a constant bit rate, inside tracks will be recorded more densely. In order to provide contant levels of magnetization across all tracks, it is desirable to have a slightly thinner coating of magnetic material near the center of the disc. This explains the reason for the slight tapering of the magnetic coating.

The "attention" bits are used to signal command completion for all commands except the STATUS command, and appear in three situations:

A. An "LIA COMMAND CHANNEL" inputs the attention bits for each of the drives (up to four). Bit N (N=0,1,2,3) corresponds to drive unit N; bit N true means that the command last issued to drive N is complete. This bit will be reset by issuance of a new command. The only exception to this is in the SEEK command, and is explained later.

B. The attention bit also appears as bit 15 of each drive unit's STATUS word. Bit 15 true signifies that the last previous command has been completed. Bit 15 may be reset by loading of the STATUS word into a computer register through the Data Channel and will not be set again until another command is issued to, and completed by, that drive unit.

C. The attention bit also appears as the Command Channel Flag. The four drive units' attention bits are OR'ed together to set this flag. This flag may be reset and otherwise handled in the normal ways.

When commands are issued, and "STC" issued to the Command Channel, the interface card will respond only if the Control Flip-Flop was previously cleared - i.e., it responds only to the changing level. Therefore, "CLC" to the Command Channel is usually executed just prior to the "STC" Command.

The error correction technique used is illustrated in Figures 4 and 5. 16 bit words are transferred one at a time (bit parallel, word serial) to the disc controller. The controller converts these bits into a serial stream, passes the resultant bit stream through a cyclic code generator, and onto the disc. The cyclic code is recorded as the last word in the sector. During a read operation, the serial bit stream is picked up by the read heads, passed to the controller, sent through the cyclic code generator, formed into 16 bit words and passed word-by-word to the computer interface. The last word to be read in a sector is the recorded Cyclic Code and this is compared with the one generated during the read operation - if they are not equal, a Data Error Status condition will be generated. Note that data error detection concerns itself only with errors introduced between the controller and the disc surface - there is no "automatic" check on errors that might be introduced during the transfer between the computer and the controller. This subject will be covered further in the section on the CHECK DATA Command.

Two things happen to the magnetic disc surface during a data write operation (see Figure 6). First the data is written. The disc surface then passes beneath two tunnel erase heads which narrow the actual data track width, and help to magnetically isolate that track from adjacent tracks.

Data is recorded using a Manchester Code (see Figure 7). The advantage of this code is that the Read/Write heads of the disc need respond only to two frequencies - the clock frequency, and twice the clock frequency. Thus, the bandwidth requirements in the heads and associated circuitry are reduced, and more reliable data recovery is achieved.

Format of the disc is shown in Figures 7 and 8. Sector gaps, or unrecorded areas, exist between each sector to allow an area for "slop" during the recording process. The recorded area of a

sector is a continuous bit stream, all recorded contiguously, and beginning with a preamble of synchronization bits. This sync bit preamble is a series of zero bits, a one bit, and some more zero bits. This allows the reading clock to adjust to the exact frequency of the recording clock, and to identify the correct bit (of the two possible) in each bit cell as the clock bit. The preamble is followed by address information for sector N+1 - this consists of cylinder number, head number, sector number, and the protected or defective status of this cylinder. The address area is followed immediately by the data area for sector N, which consists of 128 words. The cyclic code is the 129th word, and ends the sector. The cyclic code is generated from the bit stream which is the concatenation of the address and data fields. The reasons for the one sector separation of data areas for each sector are explained in other sections of these notes.

SEEK RECORD

The SEEK RECORD Command initiates head positioning on a particular drive unit. One word of (cylinder) address information is set into the data channel, the Control FF is set, and the Flag FF is cleared. The command word is then set into the Command Channel Interface Card, the Control FF set, and the Flag FF cleared. The disc controller accepts the command and looks to the Data Channel Interface Card for the cylinder number. If the Data Channel Control FF is set, the controller reads the buffer. Note that if the cylinder number is not placed into the Data Channel Interface Card (IFC) prior to command issuance, the program must ensure that the Control FF in the Data Channel is initially cleared.

The controller accepts the cylinder number, and sets the Data Channel Flag FF. The program responds by placing the head/sector number into the Data Channel IFC, setting the Control FF, and clearing the Flag FF. The controller responds by accepting this final address information and setting the Data Channel Flag FF. Note that the program may wait for the Data Channel Flag before proceeding, but that this is not required (except for perhaps diagnostic purposes).

The controller now has all of the address information required, combines it into the Record Address Register (RAR), and the drive proceeds to "SEEK" to the track and sector of interest. The head-servo mechanism moves the head over the desired track. The drive then waits until the sector of interest approaches the heads. (This is determined by counting the physical sector slots). When the address information of the sector of interest is 3.3 msec away (or approximately one sector away) from the heads, a flag is set on the Command Channel IFC. Note that no data transfer can actually take place until 6.6 msec after that flag. Thus, in Figure 8, "A" designates the approximate point of flag setting for sector N+1. This 3.3 msec is provided as a known time interval to allow the program to set up the next operation (e.g., READ,WRITE, etc.). If the Flag were to be set without regard to sector position (i.e., as soon as the heads had settled over the track), then the period

between issuing a read or write command, and commencement of reading or writing could be a random variable as large as 43.3 msec -- the variability and the possibly long delays are often undesirable to a programmer interested in optimizing programming and data transfer speeds and interactions.

Each controller may handle up to four disc drives. Not more than one SEEK may be issued to a particular drive, but SEEK's to different drives may be overlapped. SEEK is the only command that may be executed in this overlapping mode. Completion of each of the SEEK's (up to four) will generate a flag on the Command Channel IFC. The program may then determine which drive generated the flag (and completed its SEEK) by an "LIA/B COMMAND CHANNEL" to obtain the "attention" bits. The low order four bits of the loaded word specify which drive unit has completed its SEEK. If bit N=1 (N=0,1,2,3) then drive number N is the one that set the Command Channel Flag. This attention bit will remain true for 2.6 msec, then go false; it will then become true again when the address information of the sector of interest is 3.3 msec away from the heads. The attention bits for a SEEK, loaded into a register through the Command Channel, are periodic with period 40 msec and duration 2.6 msec. The Flag on the Command Channel is set by the first of these attention bits--but if the flag is cleared, subsequent attention bits will not set the flag again. Similarly, the attention bit appears as bit 15 of each drive's status word, is reset by reading the status, and is not set again by the subsequent periodic attention bits resulting from a SEEK.

It is not necessary to wait for the SEEK completion flag on the Command Channel before issuing another command on that channel. The SEEK will be completed before the READ, WRITE, etc. operation commences and the only Command Channel Flag that appears will signal the completion of the entire operation. Similarly, if a SEEK is issued to one drive, READ,WRITE,etc., Commands may be issued to other drives without waiting for the SEEK completion.

If a second SEEK command is issued to a single drive before the first has been completed, head positioning does not occur, the Command Channel Flag is set, and the SEEK-CHECK error bit is set in the status word. The next SEEK command will clear the error condition and be executed properly.

If a cylinder number greater than 202 (decimal) is issued to a drive, no head movement occurs, the SEEK-CHECK bit is set and a Command Channel Flag is set.

Mechanical movement errors may be detected in the drive servo system. These will set the appropriate error bit, and set the Flag on the Command Channel.

When the Command Channel Flag is set by any kind of SEEK error condition, the attention bit for that drive is also set, but will not be periodic.

The IOMEC SEEK-time specifications are met on all units tested so far. Note that the sales sheet refers to access times - however, track to track movement does not include rotational delay time, and

so on. For example, a data transfer preceded by a SEEK command to an adjacent track would involve the following delays: 30 msec for track to track switching, 20 msec average rotational delay time, and 6.6 msec from the time the SEEK flag sets and actual data transfer begins -- thus, the average delay in these circumstances is 56.6 msec.

The SEEK command may precede READ, WRITE, CHECK, REFINE, AND INITIALIZE Commands. In certain circumstances, the ADDRESS RECORD command may be used alternatively.

ADDRESS RECORD

The ADDRESS RECORD Command is used to change the contents of the Record Address Register (RAR) for a particular drive unit. Issuance of the command proceeds like that of a SEEK command, with three exceptions:

A. The command code is different.

B. The command code does not require a drive unit designation. This is one of the two exceptions to the standard command word format.

C. A Flag will be returned through the Command Channel as soon as the new RAR has been formed. There is no waiting for the addressed sector to be 3.3 msec away before setting that Flag.

This command may be useful in decreasing latency (rotational) delays, if the heads are already positioned above the desired cylinder. If the track number in the RAR is changed, no head positioning takes place, and any attempt at READing, etc., will fail. However, if only the head and/or sector number is changed, no head repositioning is required, and READing, WRITEing, etc may then proceed onto the addressed sector. Note that the SEEK command does not have to be issued.

If the SEEK command is used to change the RAR, an average rotational delay of 20 msec occurs before setting of the Command Channel Flag, and an additional 6.6 msec will pass before data transfer begins. However, if an ADDRESS RECORD Command is issued, followed immediately by a READ Command, an average of only 20 msec will pass before data transfer begins. If a WRITE Command follows the ADDRESS RECORD Command, an average delay of 23.3 msec would occur. The reasons for these delays will become clearer in the sections dealing with the READ and WRITE Commands.

The ADDRESS RECORD Command may be used in lieu of the SEEK Command in all cases, subject to the restrictions explained above.

If a multiple SEEK has been performed on several (up to four) drive units, it will be necessary to make the RAR conform to the address of the drive to be processed (e.g., READ) first (i.e., the drive which is first to complete its SEEK). The ADDRESS RECORD Command is the most efficient way to accomplish this.

WRITE DATA

The WRITE DATA Command usually follows a SEEK or ADDRESS RECORD Command, but it may follow any other command as long as the RAR contains the correct address information, and the heads are positioned properly. Because of the timing considerations during data transfer DMA is always used.

Programming of the WRITE Command proceeds as shown in Figure 11. Note that the Data Channel Flag must be set prior to initiation of data transfer, to ensure that no spurious data is transferred. If this is not done, one bad word will be output, and the last word of the desired data will be missed.

Execution of the WRITE Command by the controller proceeds with the controller comparing the RAR with the sector position as established by the physical sector slots. When the address information pertaining to the sector to be written arrives beneath the head, it is read. If writing is to proceed, the following conditions must be satisfied:

A. The address info read must compare with the RAR;

B. The address plus data field is cylic checked and must be okay;

C. The defective cylinder bit must not be set; and

D. The protected cylinder bit must not be set, unless the OVERIDE SWITCH is ON.

If any of the above conditions are not met, writing will be inhibited, appropriate STATUS bits set, and the Command Channel Flag set.

If writing is to proceed, the sector address in the RAR is incremented by one. This address information, along with the protected bit status of the previous sector, will be written into the next sector address field. The data is appended and written immediately after the address information.

These are two reasons for writing the address information for a sector in an area disconnected from the data for that sector:

A. The address information must be read and checked before writing is allowed to begin; and

B. The data is contiguous to the address information, and thus, when the data is written the address information must be rewritten. If address and data were to be kept together, it would not be possible to read and check the address, and then rewrite it, unless there were two sets of heads - the alternative is to separate the address and data areas.

When the DMA word count has been satisfied, and the current sector has completely passed beneath the heads, the Flag will be set on the Command Channel. The controller senses the completion of DMA data transfer by a "time-out." If the next sequential data word is not transferred to the controller within a windowed limit (approximately 19 to 26 microseconds), the controller concludes that data transfer is complete, and writing will halt at the end of the present sector. If the DMA attempts to transfer another word, or if the Control FF is set on the Data Channel IFC, after the window has passed and before end-of-sector is encountered, then the Overrun Error status bit will be set, and the Command Channel Flag set at the end of the present sector.

If some error is detected in the generation of the cylic code, the Data Error status bit is set, writing halts, and the Command Channel Flag is set.

If a DMA word count of zero is specified, one sector will be written with all zeros. If the word count is less than 128, the required number of words will be written into one sector, zeros used to fill the remaining spaces, and the cyclic code written at the end. If exactly 128 words are specified, one sector with cylic code will be written. If more than 128 words are specified, writing will automatically proceed into sequential sectors.

Data transfer must be suspended as the address information is written into each new sector. Writing will continue until the word count is satisfied, or until the end of a disc-cylinder is encountered. The drive will automatically switch from the top disc-head to the bottom disc-head and continue writing, with no rotational delay. Switching will not, however, occur from the head of one disc to the head of another. Thus, a maximum of 24 sectors may be transferred in a continuous stream. If an end of disc-cylinder is encountered before the DMA word count has been satisfied, data transfer will halt, the End-of-Cylinder status error bit set, and the Command Channel Flag set.

READ DATA

As in the WRITE Command, the READ Command utilizes DMA and must be preceded by the proper addressing and positioning commands.

Programming of the READ Command proceeds as shown in Figure 12. Note that the Data Channel Control FF must be set, and the Flag FF cleared prior to starting the DMA. This ensures that only valid data is transferred into core. If these conditions do not exist when DMA is begun, one bad word will precede the disc data, and the last word of interest from the disc will be missed.

Execution of the READ Command begins with the RAR automatically incrementing its sector address by one. The drive then counts physical sector slots until the sector of interest is beneath the heads. The address and data information is then read - they are both read because there is no separation between their respective recorded bit streams, and thus, no time to check for address validity prior to accepting the data.

For a read operation to be considered successful, the following conditions must be satisfied:

A. The address read must compare with the RAR;

B. The defective cylinder bit must not be set; and

C. The cyclic code written at the end of the address plus data field must compare with the one generated during the READ operation.

If any of the above conditions are not properly satisfied, reading continues to the end of the current sector and halts, the appropriate error bit is set, and the Command Channel Flag is set.

Data transfer completion and timing errors are handled by the controller as during the WRITE Command. If the word count is not a multiple of 128, data transfer ends at the satisfaction of the DMA word count, the rest of the sector passes beneath the heads and is cyclic checked by the controller, and the Command Channel Flag is set.

The RAR sector address is automatically incremented until the DMA word count has been satisfied, subject to the same restrictions described in the WRITE Command. If multiple sectors are being read, data transfer must, of course, be suspended while the address information in each sector is read.

CHECK DATA

The CHECK DATA Command checks the recoverability of data already written onto the disc. Only errors occurring in the transfer of data from the controller to the disc will be detected - no detection of errors occurring in the transfer of data from the computer to the controller will be found. The only method of detecting the latter type of error is to read the data of interest immediately after writing, and to compare them word-by-word - a core and time consuming procedure. The CHECK DATA Command, therefore, is a technique that checks against only some kinds of errors,but in an efficient manner.

Programming of the CHECK DATA Command proceeds as in Figure 13. Note that, in addition to cylinder/head/sector information, a sector count is also required - this is expressed as a positive number and is output through the Data Channel.

Execution of the CHECK DATA Command by the controller and drive proceeds exactly as the READ Command, except that no data is transferred over the Data Channel. Each sector is cyclic checked - this, along with the address checking, is the full extent of the check.

The Command Channel Flag will be set when the sector count

has been satisfied, or when errors are detected. No data transfer timing error can occur.

REFINE SECTOR

The REFINE SECTOR Command is an error recovery operation. However, it should be used only after other error recovery techniques have been attempted - e.g., re-read, move heads and re-read, etc.

Programming of the REFINE SECTOR Command is illustrated in Figure 14. Note that no provision is made for specifying a word count or a sector count. Only one sector may be "refined" per command, and that will be the sector specified in the RAR.

Refining a sector consists of a tunnel erase over the sector specified. This has the effect of (perhaps!) lessening "cross-talk" from adjacent tracks. Note that it may also have the effect of partially erasing adjacent tracks, and should be used cautiously and only as a last resort.

During a REFINE SECTOR, no transfers through the data channel occur. Because no address check is made, no address error can occur.

The Flag will be set on the Command Channel when the tunnel erase of the single sector is complete.

STATUS CHECK

The STATUS CHECK Command allows a 15-bit status word for one of the disc drive units to be loaded in through the Data Channel.

Programming of the command is illustrated in Figure 15. The Data Channel is prepared for acceptance of the status word by setting the Control FF. The Data Channel Flag is cleared in preparation for controller response. The STATUS CHECK Command is the only command that signals its completion by setting the Data Channel Flag. The Command Channel Flag will not be set. When the Data Flag is set, the status may be loaded into a computer register through the Data Channel.

The STATUS CHECK Command may be used at any time, and is often used following commands involving data transfer. Note that if an error is detected during data transfer, the Command Channel Flag will be set, and the DMA may be "hung-up" with an unsatisfied word count. Attempting to obtain the status through the Data Channel will result in the interception of the Device Response Flag by the DMA, resetting of the Control FF, and reclearing of the Flag FF. Thus, although the status will be properly loaded into the Data Channel Interface Register, no Flag will ever be seen as "set" by the checking program. To allow the Flag to come through and be seen in these circumstances, it may be necessary to clear the DMA (e.g., STF 6) before proceeding with the STATUS CHECK Command.

The step of ensuring that DMA is cleared is not shown in the STATUS Command programming example. However, any driver written for this disc must consider the DMA "hang-up" problem when checking status.

INITIALIZE DATA

The INITIALIZE DATA Command must be used before any other reading from or writing onto a particular disc cartridge may occur. It may also be used for subsequent protecting/unprotecting, or flagging as defective of particular disc cylinders.

Programming of this command proceeds much like the WRITE Command. Note, however, that this is one of two commands which vary in command word format. If bit 8 is set, the sectors to be initialized will be flagged as defective. If bit 9 is set, those sectors will be flagged as protected. If both bits 8 and 9 are set in the command word, only the defective bit will be written onto the disc. Entire cylinders must be flagged if any sector within that cylinder is to be flagged. Cylinder is defined here, as in the interface manual, as two tracks, both on the same disc; cylinder is not defined in the conventional manner as the 4 tracks on both discs.

Command execution is performed by the controller much as it executes the WRITE Command, but with one notable exception. Since the disc may be blank before the execution of this command, there is no written address information to check. Therefore, the RAR sector address is incremented by one, and writing of addresses, cylinder flags, and data proceeds.

The OVERIDE SWITCH must be ON, or the command will not execute.

A good programming technique for defective cylinders is to rewrite those cylinders, using the INITIALIZE DATA Command, as those cylinders are discovered as defective. The defective bit must be set in each of the sector addresses in that cylinder, and the address of the alternate cylinder written as the first word of each sector. A program looking for data in a given cylinder will discover the defective bit set, but should be able to successfully read at least one of the sectors, thus obtaining the alternate cylinder address.

STATUS WORD BIT DEFINITIONS

These notes on Status word bit definitions mention only exceptions or clarifications of the Interface Manual descriptions.

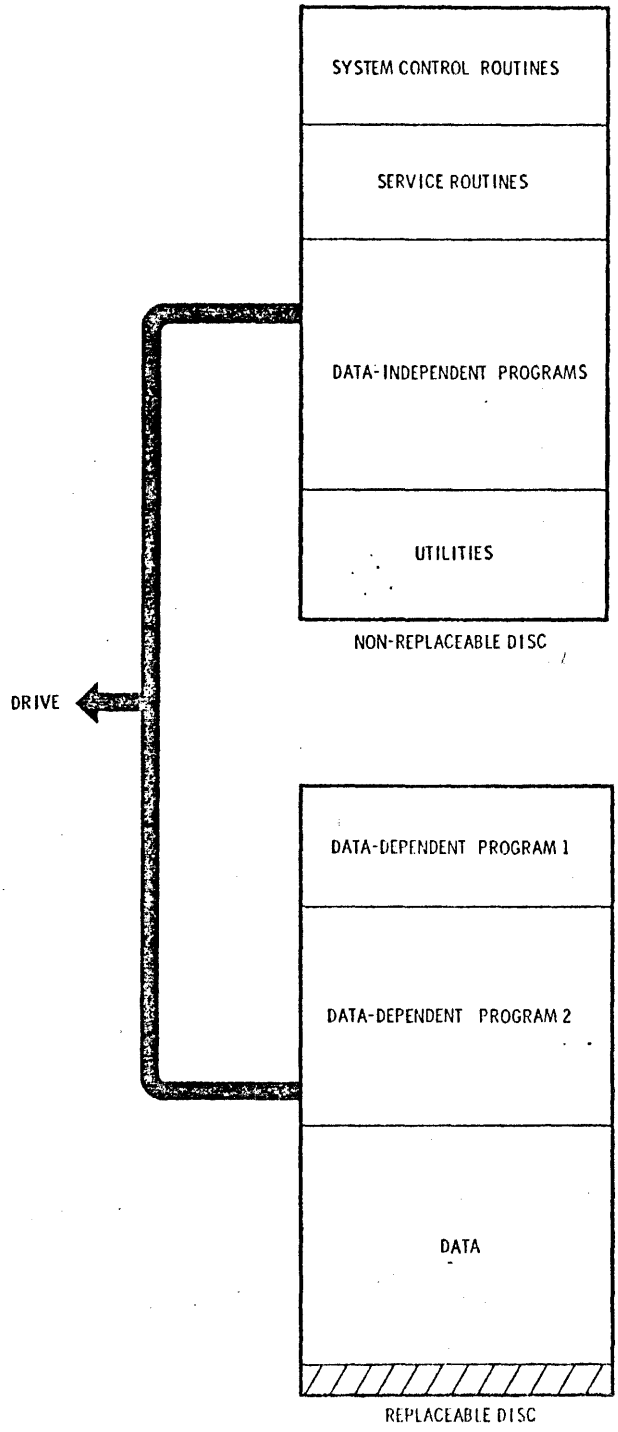
BIT Ø

ANY ERROR - "any error" is somewhat of a misnomer. This bit is set when any other bit in the Status word is set, with the exception of bits 2,7, and 15, and sometimes 3.

BIT 1
DATA ERROR - Can be set by cyclic code checks.

BIT 3
FLAGGED
CYLINDER - Set if cylinder being processed is protected or defective; check bit 4 to determine which. See Figure 17 for clarification of whether bit Ø may be set or not set.

BIT 4
ADDRESS ERROR - See Figure 17.



Typical Multiple Disc Capacity File Organization

FIGURE 1.

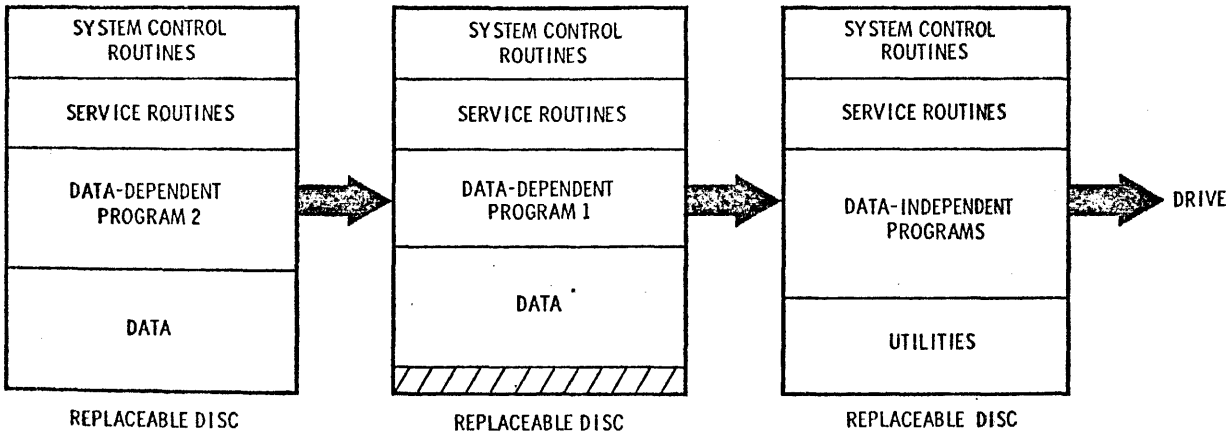
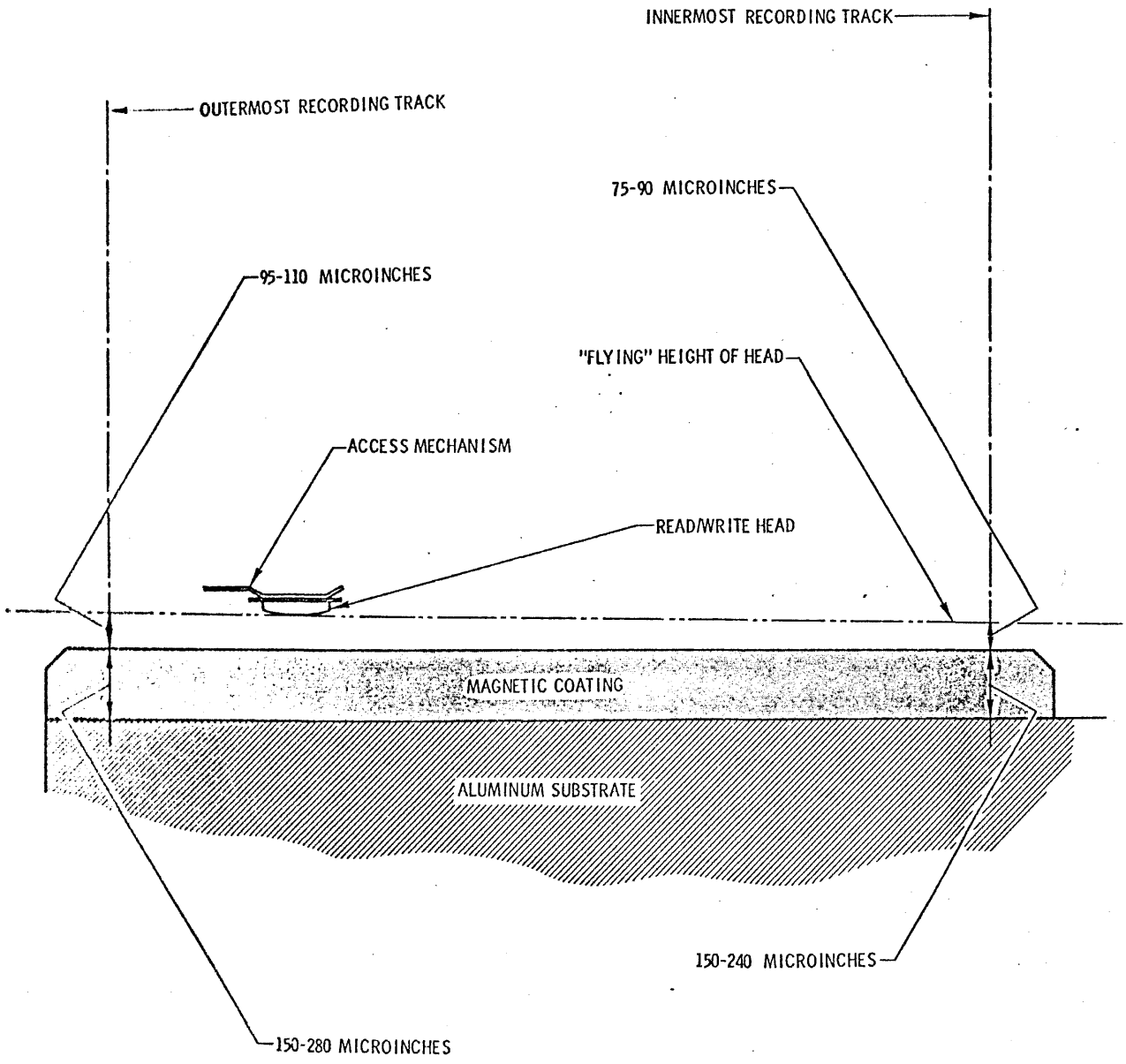


Figure 2. Typical Single Disc Capacity File Organization



Disc Recording Surface Characteristics
(2200 BPI Recording)

FIGURE 3.

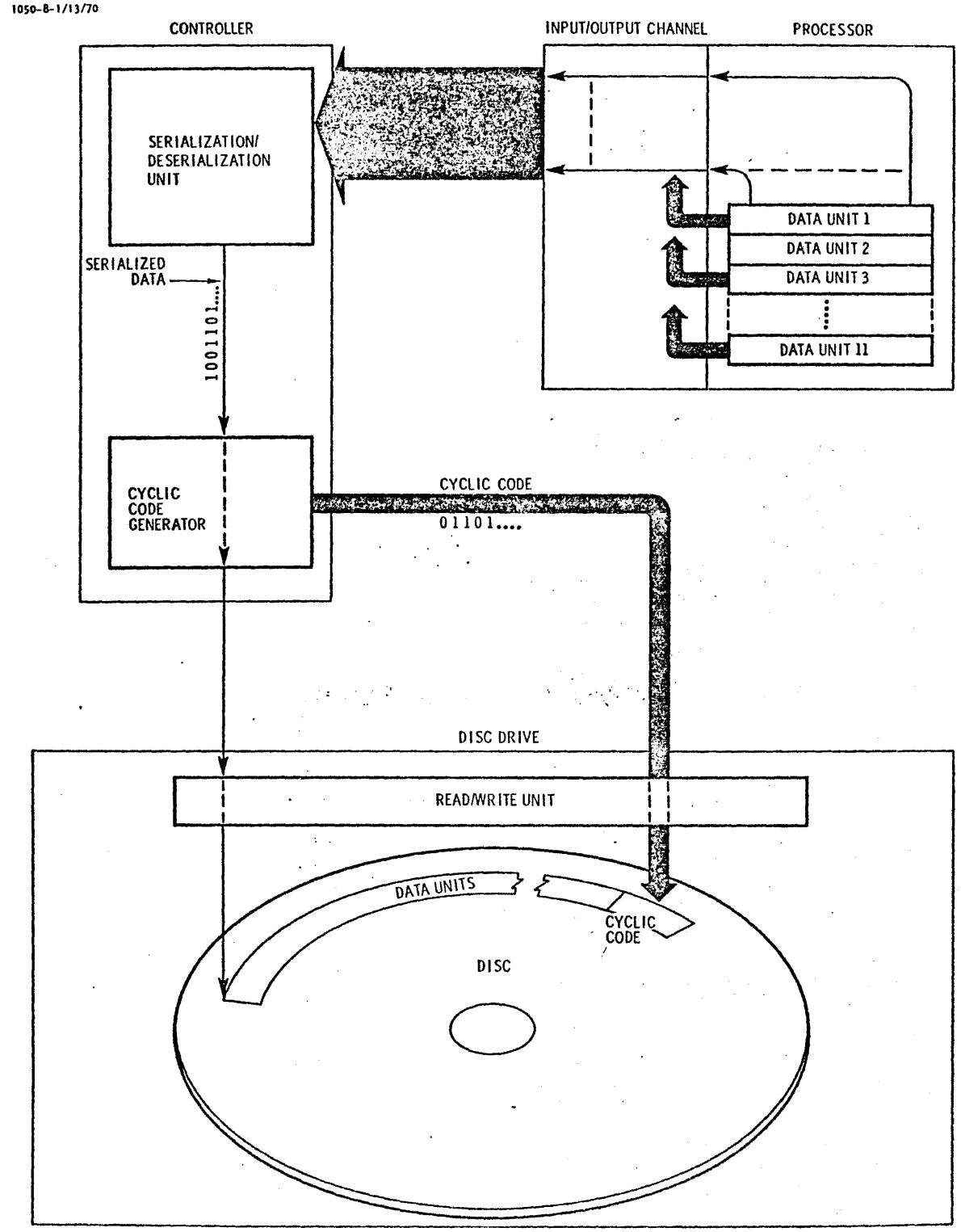


Figure 4. Generation of Cyclic Code (Write Operation)

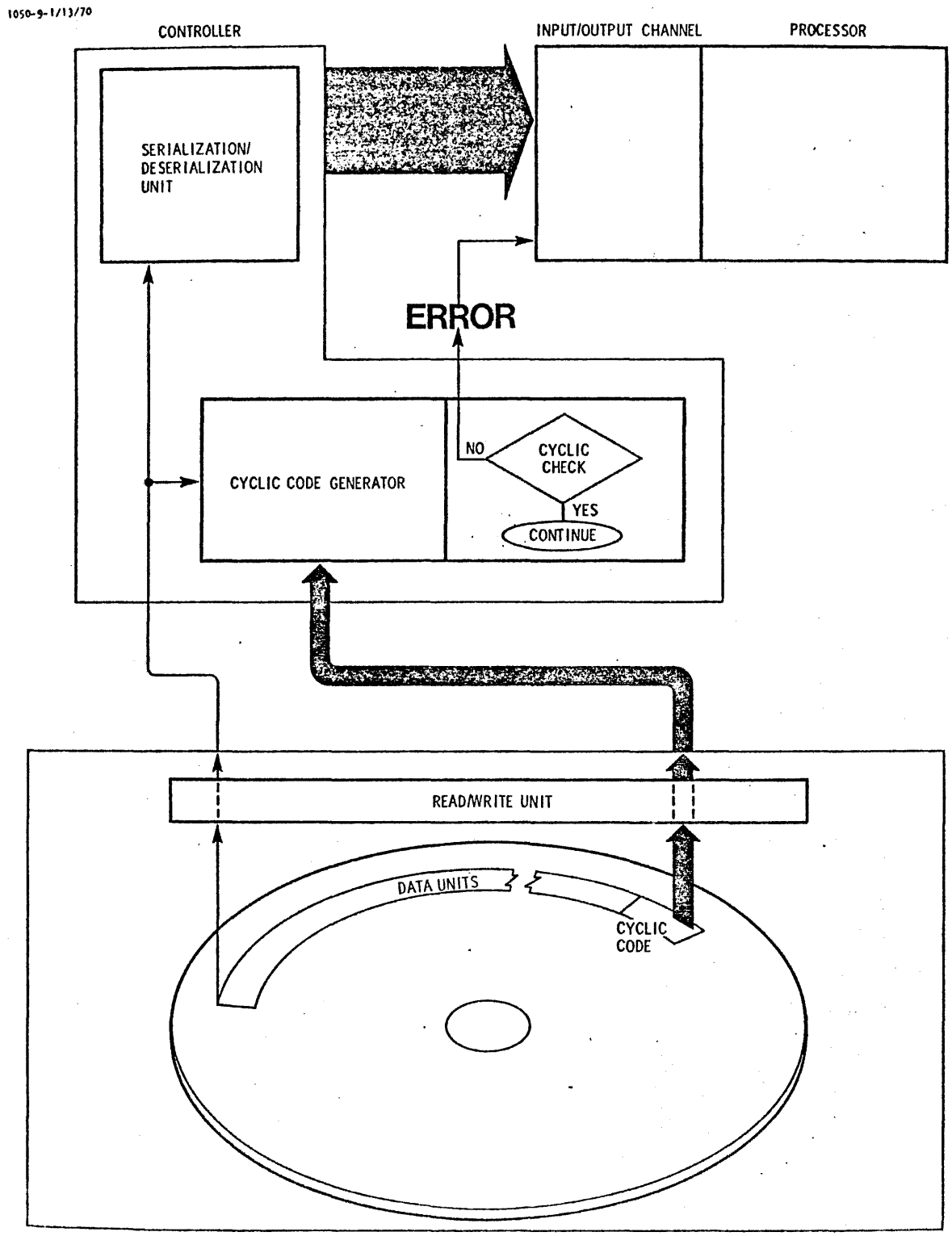


Figure 5. Cyclic Code Error Detection (Read Operation)

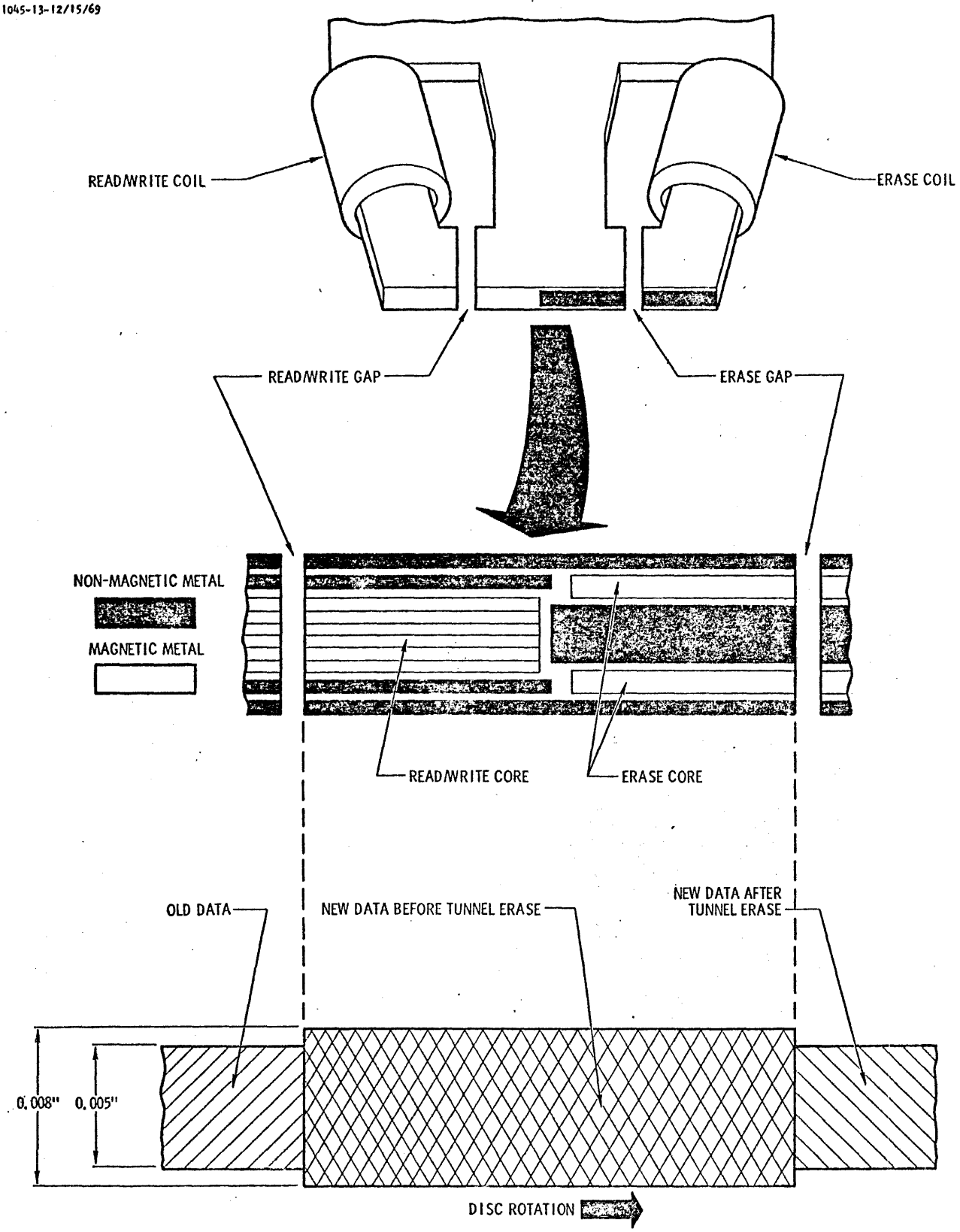


Figure 6. Read/Write-Erase Structure

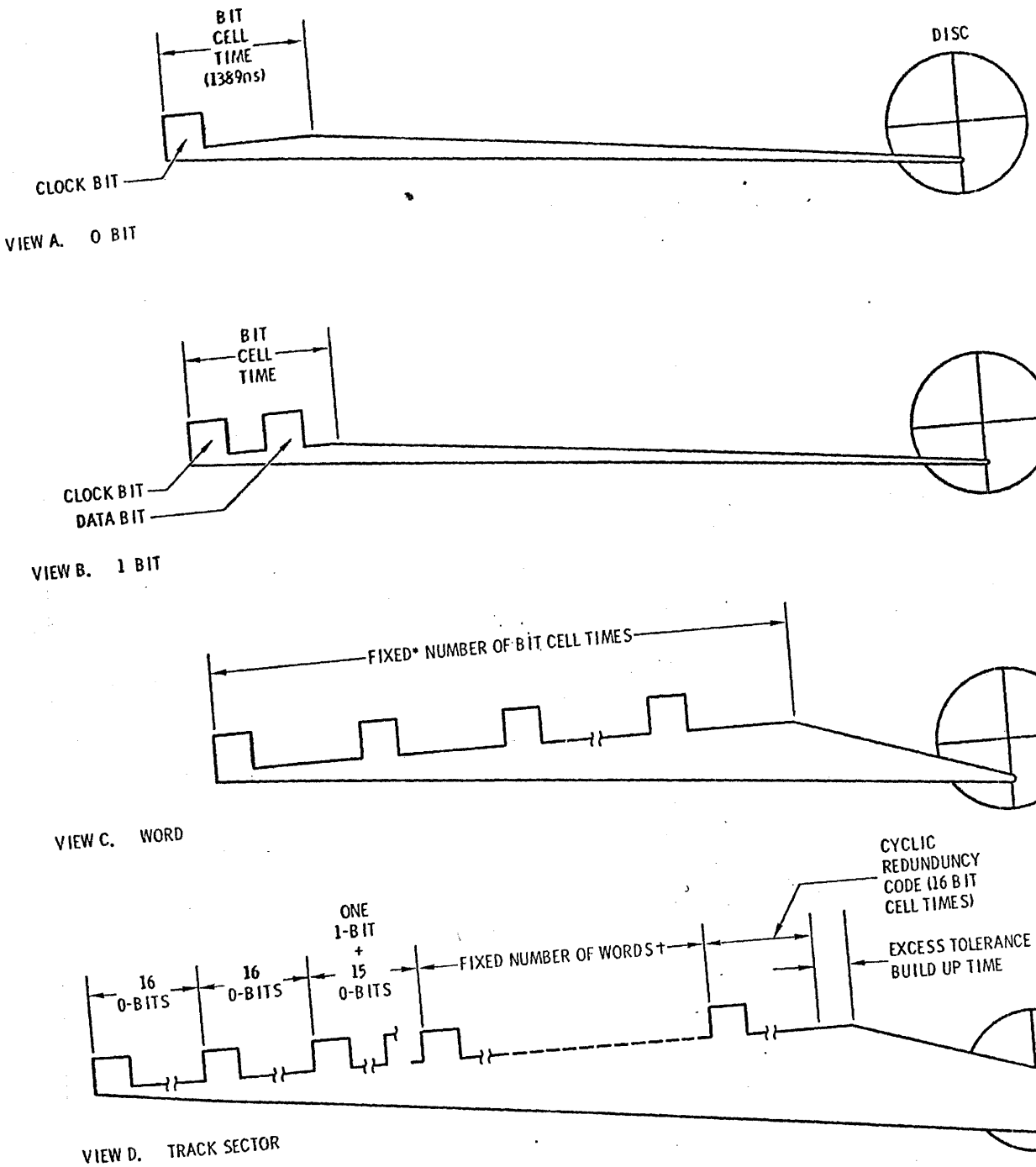


FIGURE 7. DATA TIMING

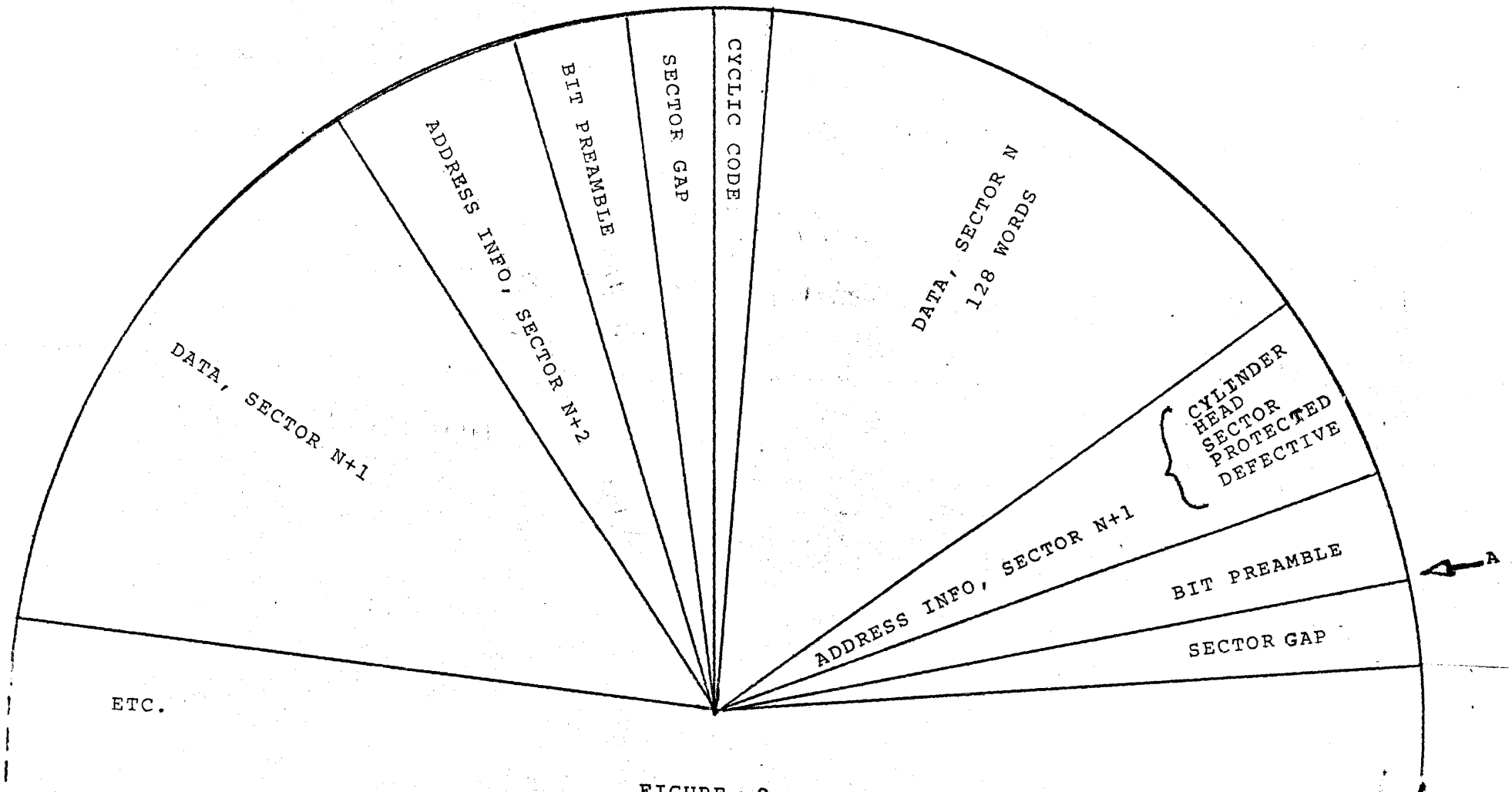

DATA, SECTOR N+1
ADDRESS INFO, SECTOR N+2
BIT PREAMBLE
SECTOR GAP
CYCLIC CODE
DATA, SECTOR N
128 WORDS
CYLINDER
HEAD
SECTOR
PROTECTED
DEFECTIVE
ADDRESS INFO, SECTOR N+1
BIT PREAMBLE
SECTOR GAP
A
ETC.

FIGURE 8.

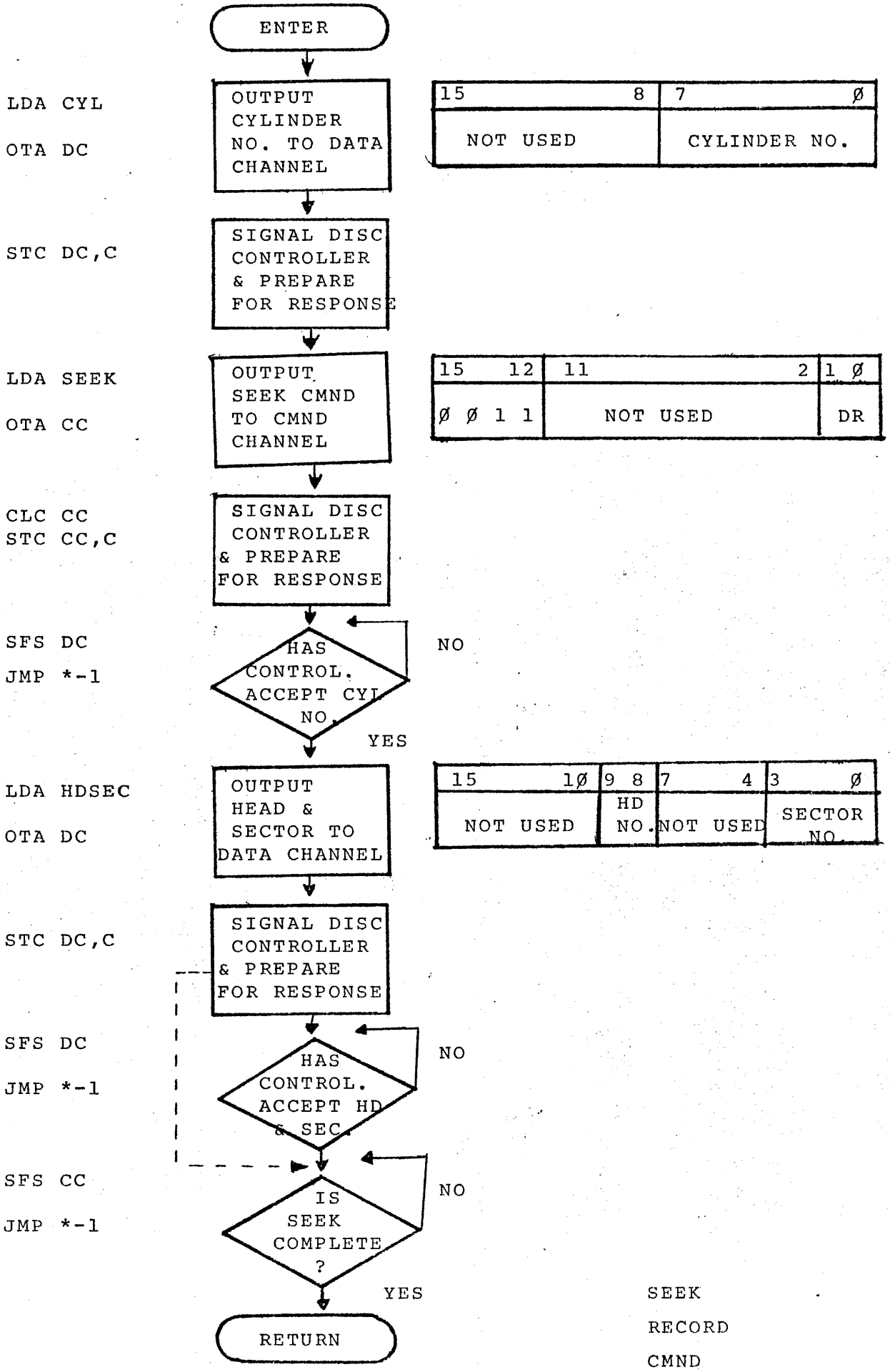

ENTER
LDA CYL
OTA DC
OUTPUT CYLINDER NO. TO DATA CHANNEL
15 8
NOT USED
7 Ø
CYLINDER NO.
STC DC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
LDA SEEK
OTA CC
OUTPUT SEEK CMND TO CMND CHANNEL
15 12
Ø Ø 1 1
11 2
NOT USED
1 Ø
DR
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS DC
JMP *-1
HAS CONTROL. ACCEPT CYL NO.
NO
YES
LDA HDSEC
OTA DC
OUTPUT HEAD & SECTOR TO DATA CHANNEL
15 1Ø
NOT USED
9 8
HD NO.
7 4
NOT USED
3 Ø
SECTOR NO.
STC DC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS DC
JMP *-1
HAS CONTROL. ACCEPT HD & SEC.
NO
SFS CC
JMP *-1
IS SEEK COMPLETE ?
NO
YES
RETURN
SEEK
RECORD
CMND

FIGURE 9.

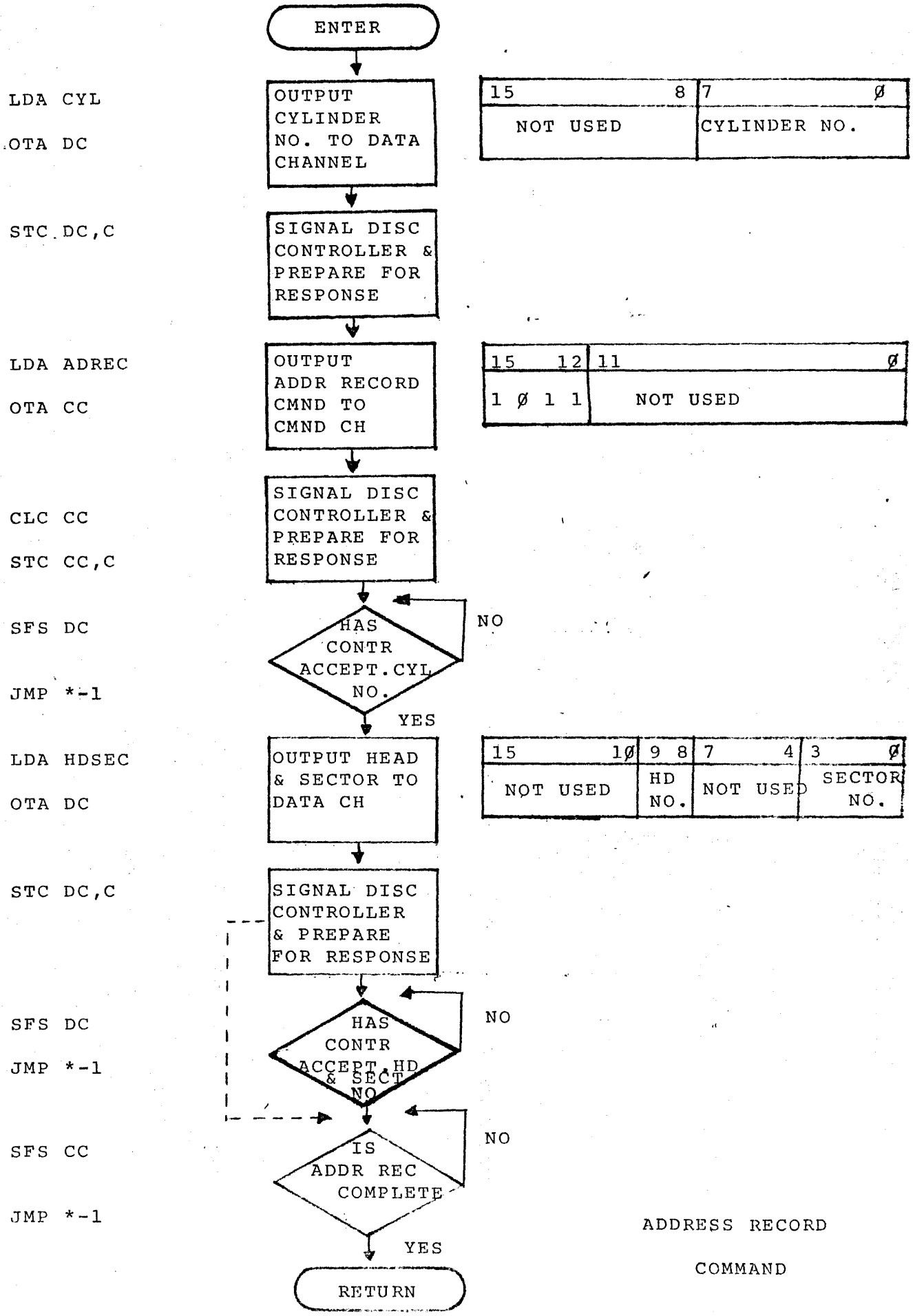

ENTER
LDA CYL
OTA DC
OUTPUT CYLINDER NO. TO DATA CHANNEL
15 8 7 Ø
NOT USED
CYLINDER NO.
STC DC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
LDA ADREC
OTA CC
OUTPUT ADDR RECORD CMND TO CMND CH
15 12 11 Ø
1 Ø 1 1
NOT USED
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS DC
JMP *-1
HAS CONTR ACCEPT.CYL NO.
NO
YES
LDA HDSEC
OTA DC
OUTPUT HEAD & SECTOR TO DATA CH
15 1Ø 9 8 7 4 3 Ø
NOT USED
HD NO.
NOT USED
SECTOR NO.
STC DC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS DC
JMP *-1
HAS CONTR ACCEPT.HD & SECT NO.
NO
SFS CC
JMP *-1
IS ADDR REC COMPLETE
NO
YES
RETURN
ADDRESS RECORD
COMMAND

FIGURE 10.

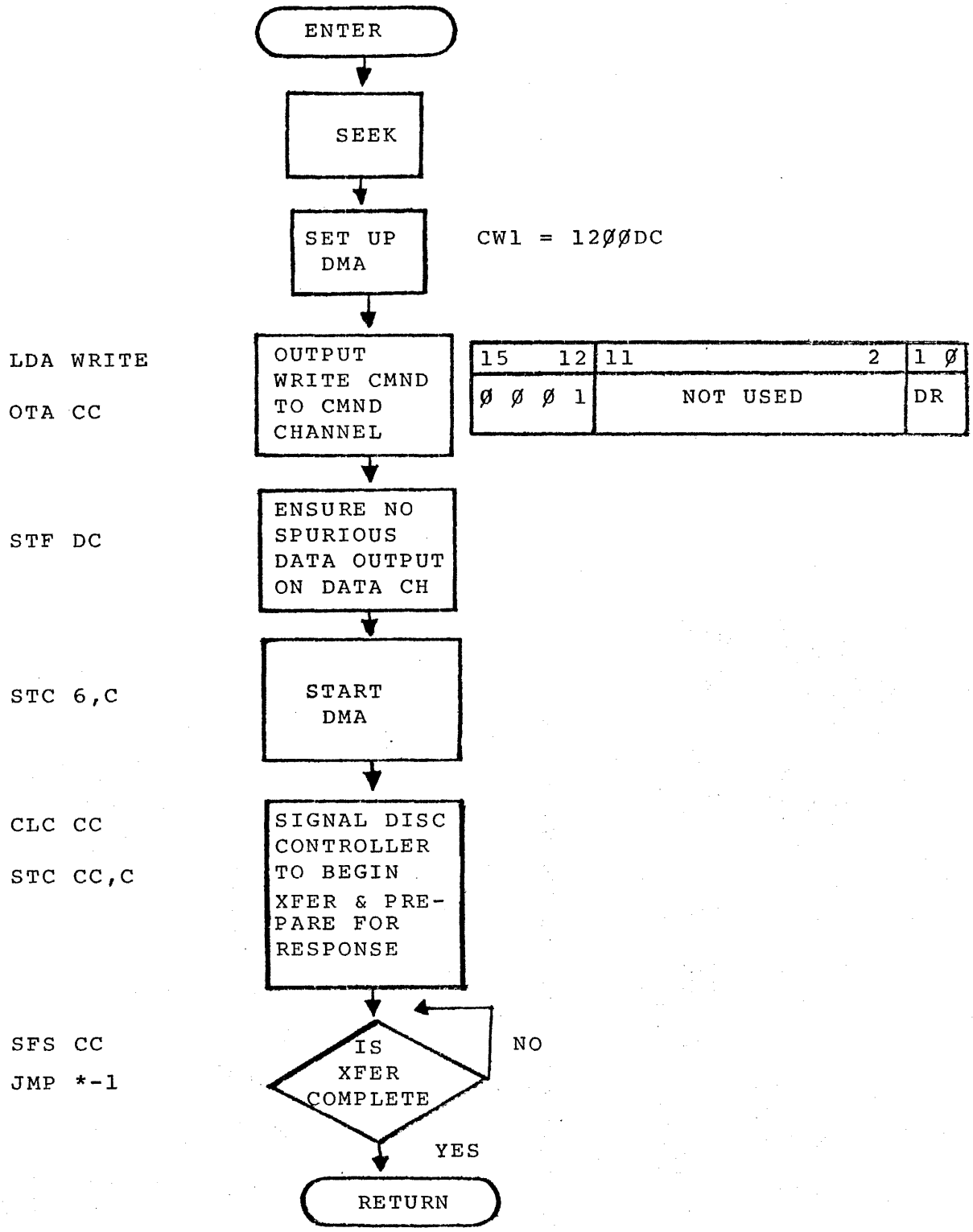

ENTER
SEEK
SET UP DMA
CW1 = 12ØØDC
LDA WRITE
OTA CC
OUTPUT WRITE CMND TO CMND CHANNEL
15 12
11 2
1 Ø
Ø Ø Ø 1
NOT USED
DR
STF DC
ENSURE NO SPURIOUS DATA OUTPUT ON DATA CH
STC 6,C
START DMA
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER TO BEGIN XFER & PREPARE FOR RESPONSE
SFS CC
JMP *-1
IS XFER COMPLETE
NO
YES
RETURN
WRITE
CMND

FIGURE 11.

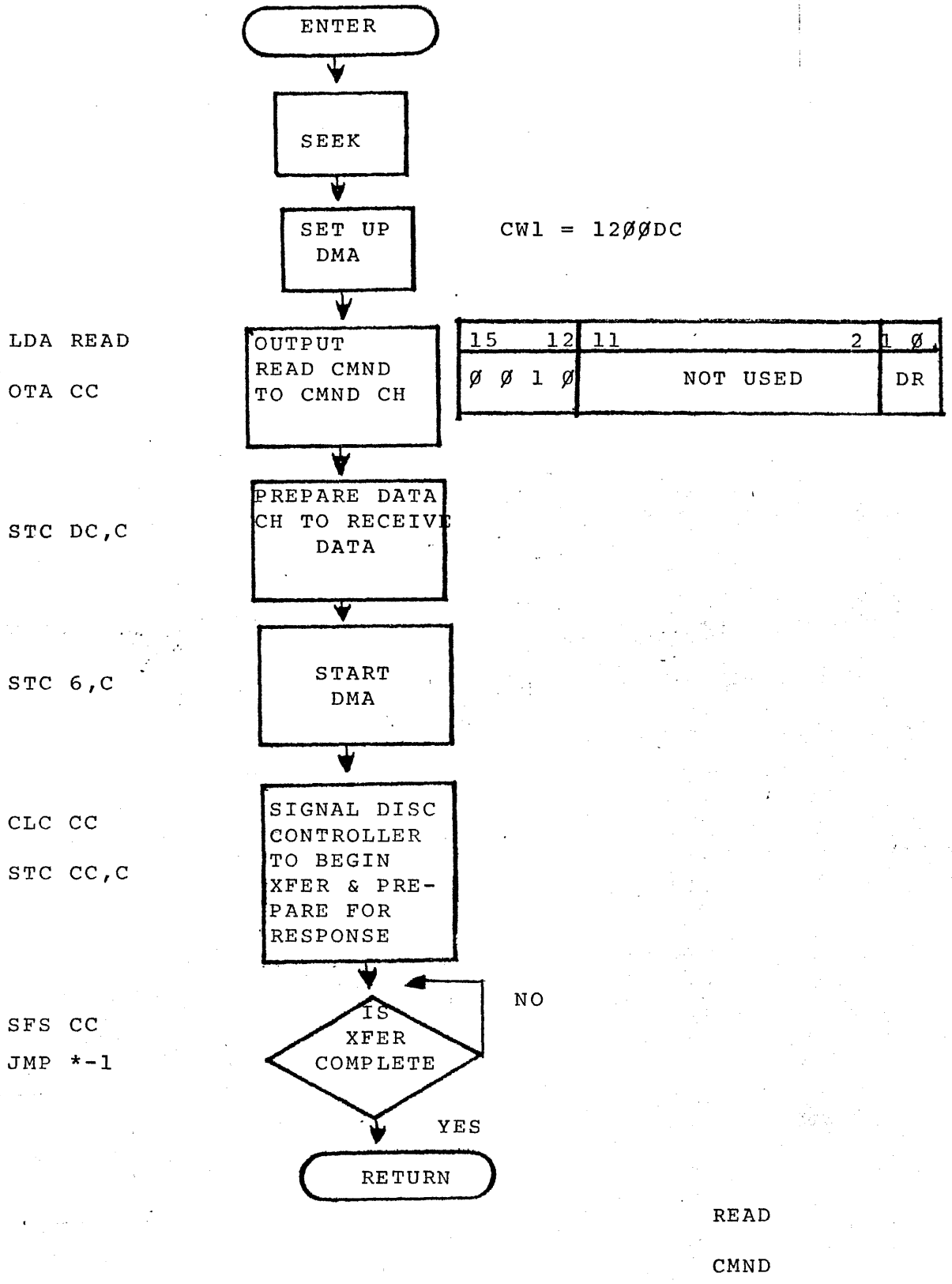

ENTER
SEEK
SET UP DMA
CW1 = 12ØØDC
LDA READ
OTA CC
OUTPUT READ CMND TO CMND CH
15 12
11 2
1 Ø
Ø Ø 1 Ø
NOT USED
DR
STC DC,C
PREPARE DATA CH TO RECEIVE DATA
STC 6,C
START DMA
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER TO BEGIN XFER & PREPARE FOR RESPONSE
SFS CC
JMP *-1
IS XFER COMPLETE
NO
YES
RETURN
READ
CMND

FIGURE 12.

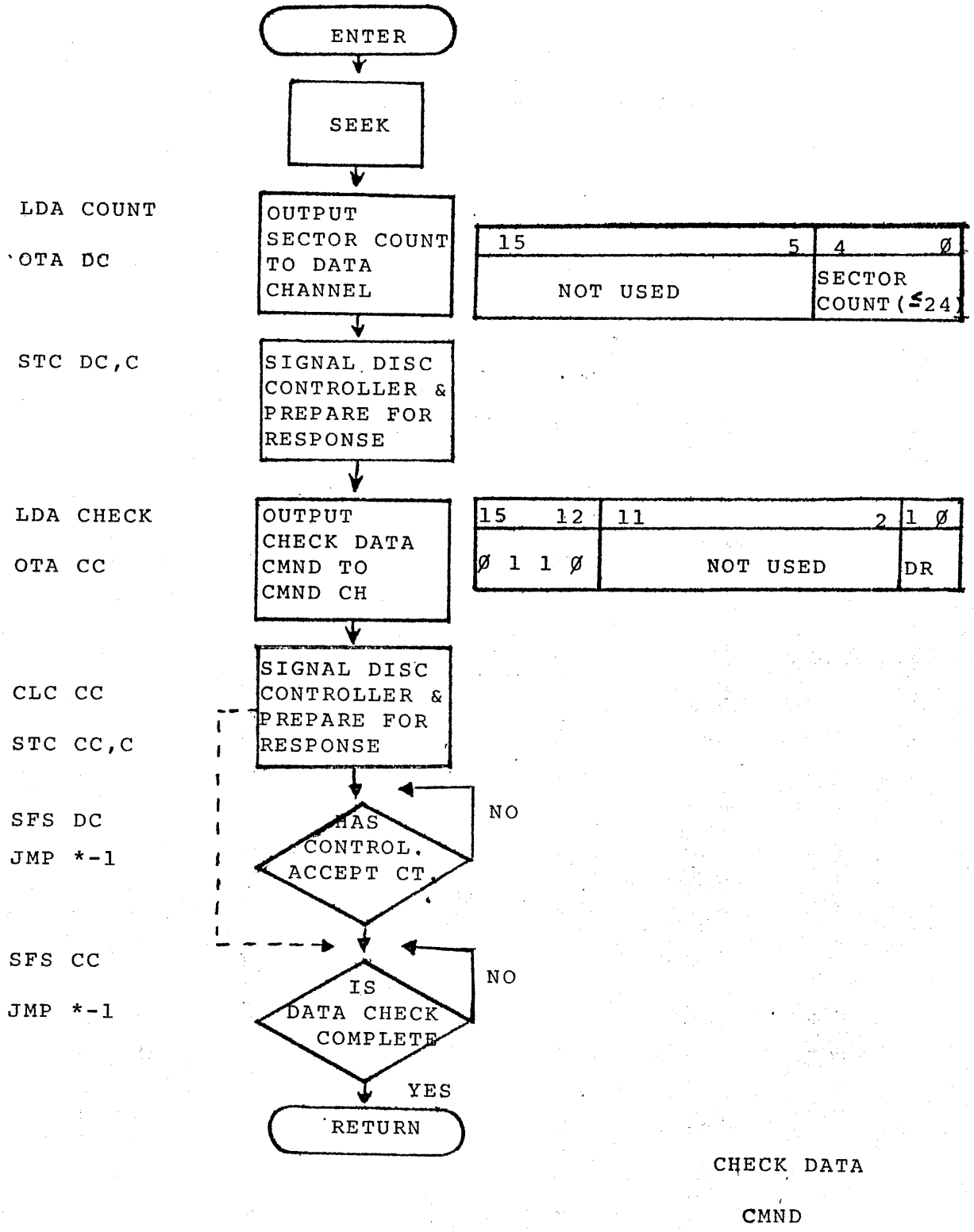

ENTER
SEEK
LDA COUNT
OTA DC
OUTPUT SECTOR COUNT TO DATA CHANNEL
15 5
NOT USED
4 Ø
SECTOR COUNT (≤24)
STC DC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
LDA CHECK
OTA CC
OUTPUT CHECK DATA CMND TO CMND CH
15 12
Ø 1 1 Ø
11 2
NOT USED
1 Ø
DR
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS DC
JMP *-1
HAS CONTROL, ACCEPT CT.
NO
SFS CC
JMP *-1
IS DATA CHECK COMPLETE
NO
YES
RETURN
CHECK DATA
CMND

FIGURE 13.

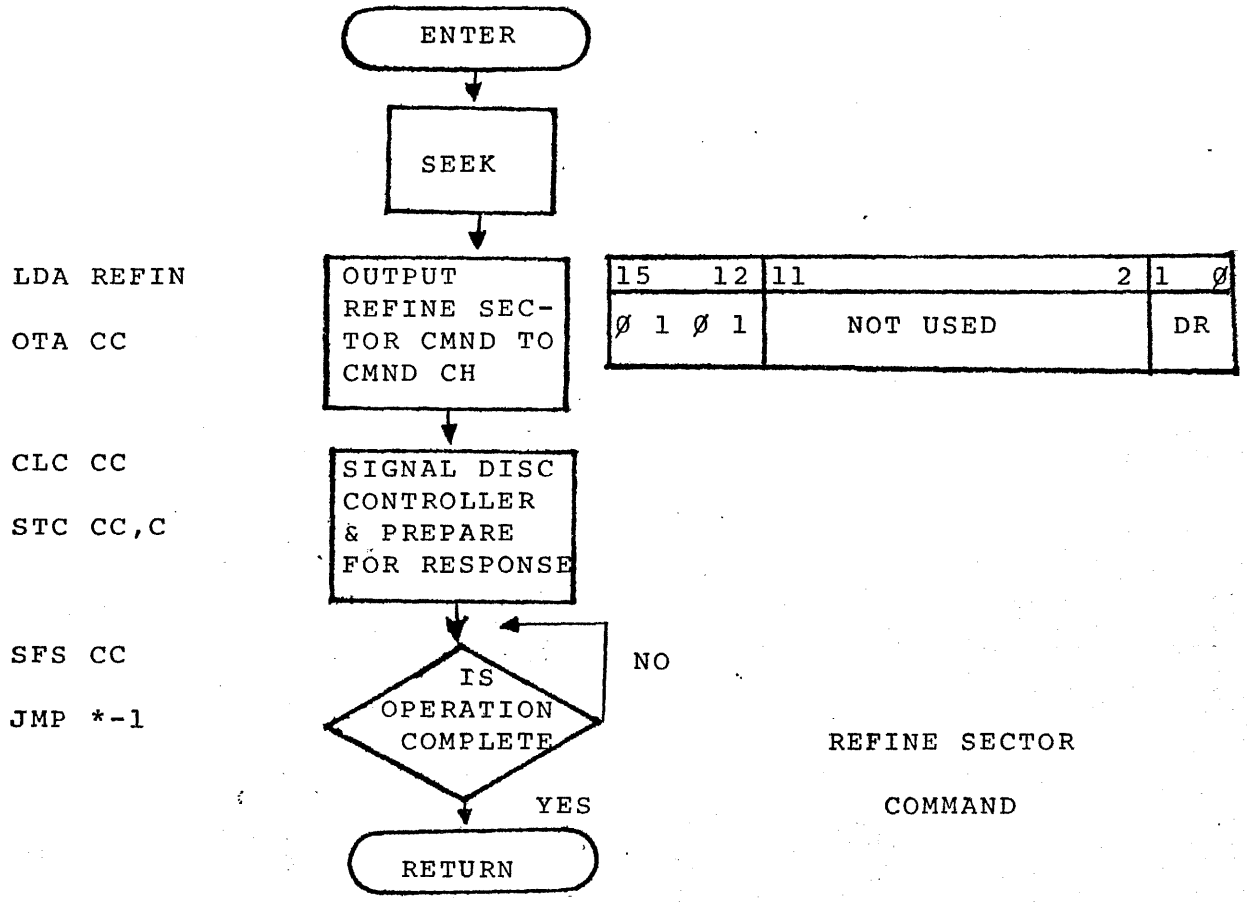

ENTER
SEEK
LDA REFIN
OTA CC
OUTPUT REFINE SECTOR CMND TO CMND CH
15 12
11 2
1 Ø
Ø 1 Ø 1
NOT USED
DR
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS CC
JMP *-1
IS OPERATION COMPLETE
NO
YES
RETURN
REFINE SECTOR
COMMAND

FIGURE 14.

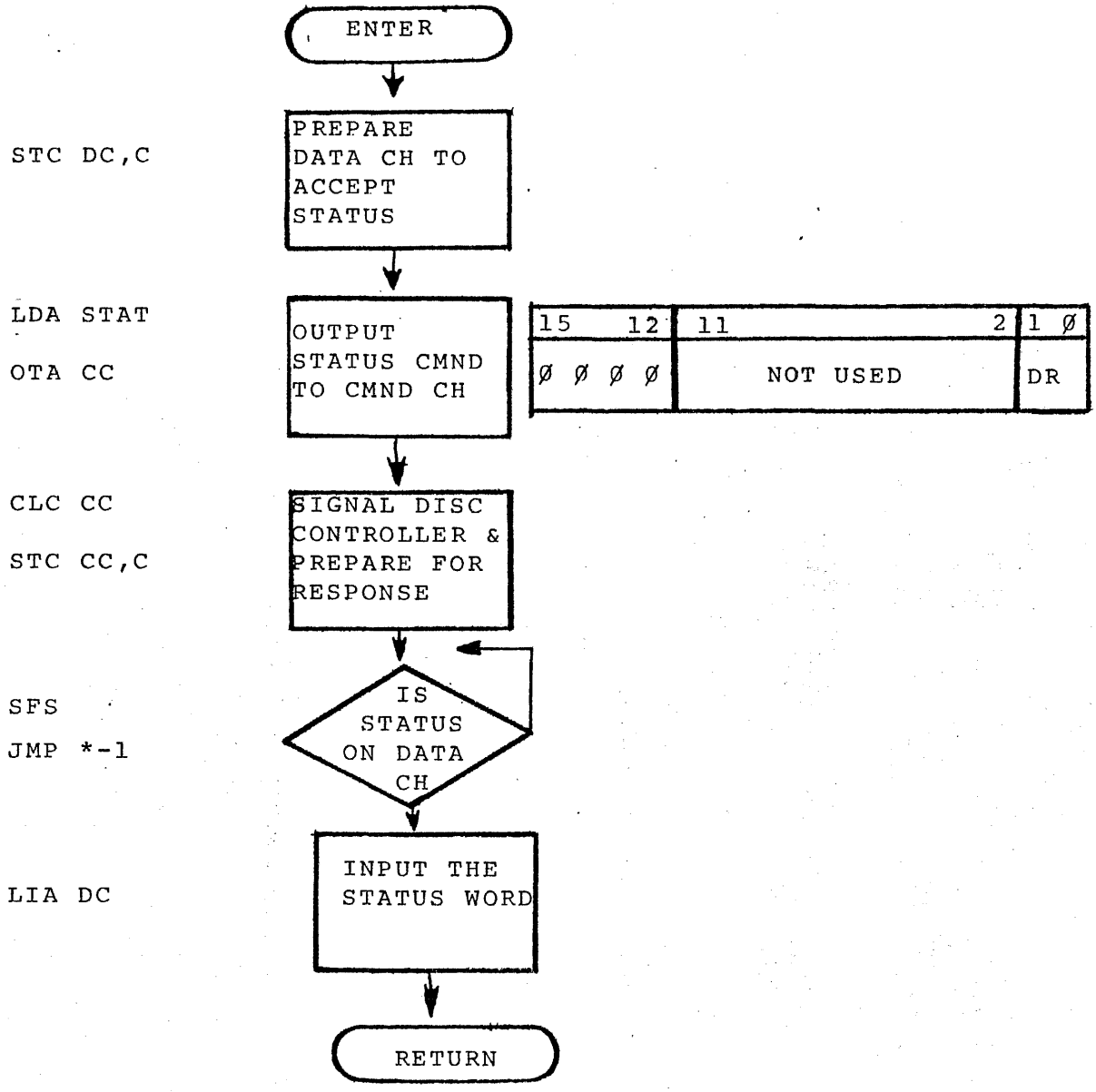

ENTER
STC DC,C
PREPARE DATA CH TO ACCEPT STATUS
LDA STAT
OTA CC
OUTPUT STATUS CMND TO CMND CH
15 12
11 2
1 Ø
Ø Ø Ø Ø
NOT USED
DR
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS
JMP *-1
IS STATUS ON DATA CH
LIA DC
INPUT THE STATUS WORD
RETURN
STATUS CHECK
CMND

FIGURE 15.

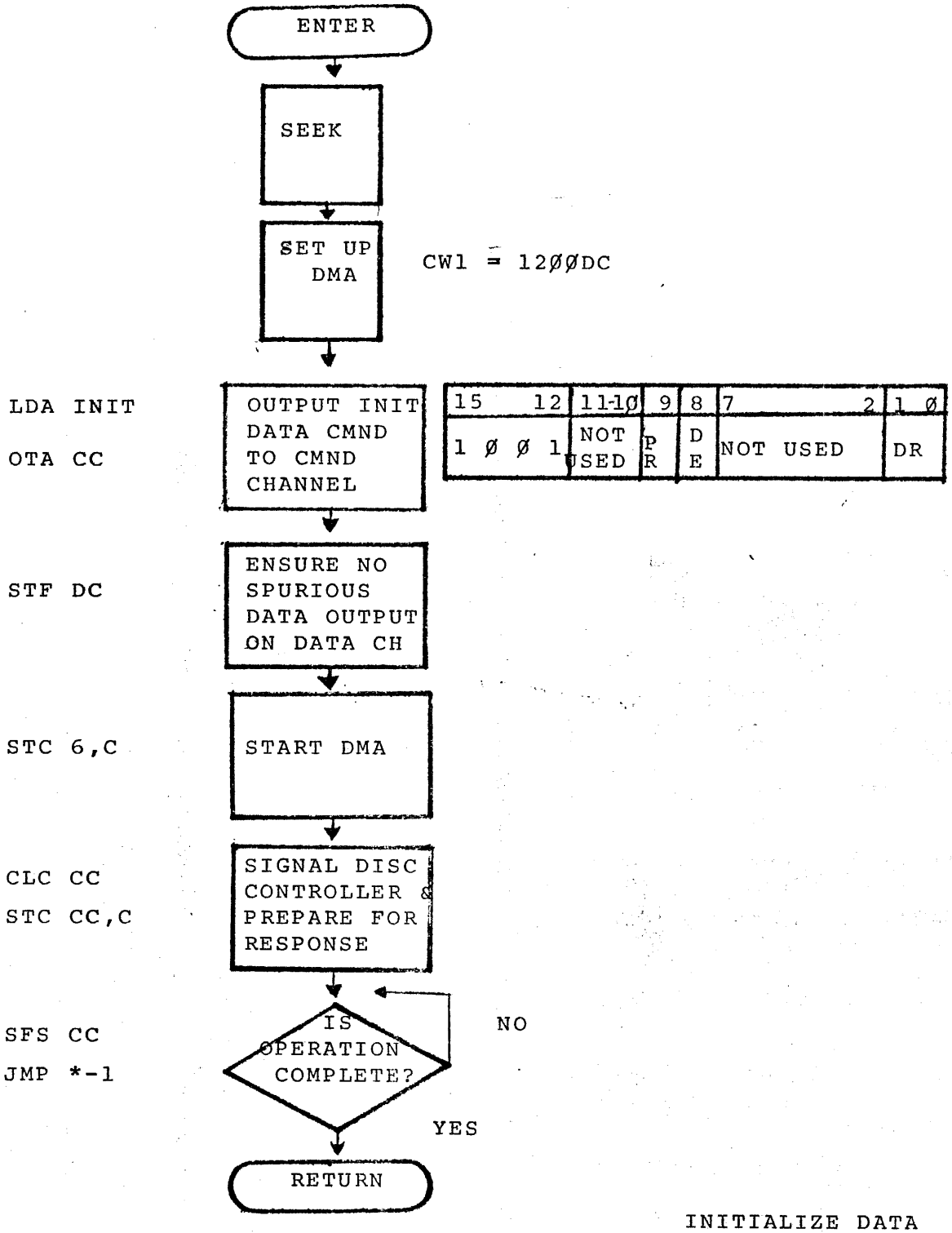

ENTER
SEEK
SET UP DMA
CW1 = 12ØØDC
LDA INIT
OTA CC
OUTPUT INIT DATA CMND TO CMND CHANNEL
15 12
11-1Ø
9
8
7 2
1 Ø
1 Ø Ø 1
NOT USED
PR
DE
NOT USED
DR
STF DC
ENSURE NO SPURIOUS DATA OUTPUT ON DATA CH
STC 6,C
START DMA
CLC CC
STC CC,C
SIGNAL DISC CONTROLLER & PREPARE FOR RESPONSE
SFS CC
JMP *-1
IS OPERATION COMPLETE?
NO
YES
RETURN
INITIALIZE DATA COMMAND

FIGURE 16.

		BIT 4	BIT 3	BIT Ø
OVERIDE SWITCH ON PROTECT BIT SET				
	READ	0	1	0
	WRITE	0	1	0
	CHECK DATA	0	1	0
	INITIALIZE DATA	0	0	0
DEFECTIVE BIT SET				
	READ	1	1	1
	WRITE	1	1	1
	CHECK DATA	1	1	1
	INITIALIZE DATA	0	0	0
OVERIDE SWITCH OFF PROTECT BIT SET				
	READ	0	1	0
	WRITE	0	1	1
	CHECK DATA	0	1	0
	INITIALIZE DATA	0	1	1
DEFECTIVE BIT SET				
	READ	1	1	1
	WRITE	1	1	1
	CHECK DATA	1	1	1
	INITIALIZE DATA	0	1	1

BIT 4 - ADDRESS ERROR

BIT 3 - PROTECT/DEFECTIVE

BIT Ø - ERROR