

Internal Maintenance Specification

SYSTEM CHECKOUT PROGRAM 2.0

MSOS 4.0 Operating System

(C) COPYRIGHT CONTROL DATA CORP. 1973

Contained herein are software products copyrighted by Control Data Corporation. A reproduction of the copyright must appear on all complete or partial copies.

DOCUMENT CLASS TMS PAGE NO. 2
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012M1-E006M2 MACHINE SERIES 1700

TABLE OF CONTENTS

1.0	SCOPE	3
2.0	APPLICABLE DOCUMENTS	3
3.0	REQUIREMENTS	4
3.1	Definitions	4
3.1.1	Internal Symbols	4
3.1.2	Entry Symbol	6
3.1.3	External Symbols	6
3.1.4	Macro Assembler Definitions	7
3.1.5	Terms Used	8
3.2	Program Function	8
3.3	Hardware Requirements	8
3.4	Software Requirements	9
3.4.1	Computer Operator's Instructions	9
3.4.2	Relocatability	10
3.4.3	Analysis	10
3.4.4	Program Description	11
3.4.4.1	COBOP--Checkout Bootstrap Program	11
3.4.4.2	SYSCOP--System Checkout Program--Control Portion	12
3.4.4.3	SYSSEG--System Checkout Segments	15
3.4.4.4	Addition of New Segments	22

DOCUMENT CLASS IMS PAGE NO. 3
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. M012x2-E006x4 MACHINE SERIES 1700

1.0 SCOPE

This document describes the System Checkout Program, which is an on-line program that diagnoses failures in an E006x4 Operating System. The program performs the diagnosis on a failed image which has been written onto mass memory. The program operates in protected core; therefore, core size is a prime consideration.

2.0 APPLICABLE DOCUMENTS

2.1 Relevant Programs

E006x4 Mass Storage Operating System

2.2 Relevant Flow Diagrams and Figures

See attached flowcharts

2.3 Other Relevant Documents

CDC Publication 60362200- Mass Storage Operating System
Version 4.0 Reference Manual

DOCUMENT CLASS IMS PAGE NO. 4
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12M2-E006M4 MACHINE SERIES 1700

3.0 REQUIREMENTS

3.1 Definitions

3.1.1 Internal Symbols

SEGSIZ Length of segment {blocks} to be read from mass memory. Set to 96 words {1 sector}.

ASEC Word in sector zero which contains the hardware sector address of the system autoload image. Set to 15₁₀.

CP Priority level of the System Checkout Program. Set to 3.

HERE Current segment number in SEGMNT {must be at SEGMNT-1}.

SEGMNT Start of area used to exuecte each segment of the various system checkout programs {C01ST, C02ND, etc.}.

LOADA Subroutine used to fetch a specific message segment from the failed image.

AUTLDA Subroutine used to fetch a specific call from the autoload image.

GMEASG Subroutine used to fetch a specific message segment into core {MMBUF}.

MMOP Subroutine which outputs the main header messages and assures that they are output only once.

CMPENT Subroutine to compare two values.

CONVRT Subroutine which converts a one word hexadecimal value to two words of ASCII characters.

CONVST Subroutine which converts a specified buffer of hexadecimal values and stores the ASCII characters into a specific pattern.

LISTIT Subroutine which outputs a specific message {depending on the option selected} on the Print Logical Unit.

WRITIT Subroutine which outputs a specific message on the Comment Output Logical Unit.

DOCUMENT CLASS IMS PAGE NO. 5
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12M2-E006M4 MACHINE SERIES 1700

ANSWER Subroutine which inputs a line from the Comment Input Logical Unit.

EXITCO Routine which release the entire System Checkout Program.

XFERTB A transfer vector to the various subroutines consisting of the following. {NOTE: I-register points to XFERTB during execution}.

<u>NAME</u>	<u>VALUE</u>	<u>SUBROUTINE</u>
-	0	EXITCO
LDA	1	LOADA
CON	3	CONVRT
LST	5	LISTIT
WRT	7	WRITIT
ANS	9	ANSWER
ATA	11	AUTLDA
CNS	13	CONVST
GMS	15	GMSAG
MOP	17	MMOP
CMP	19	COMPAR

MMBUF Start of area used for storage of message segments or sectors from failed or autoloading images.

SEGCOR Number of locations between XFERTB and MMBUF.

NEXTN Segment number of the first sector of the next segment loaded {i.e., the first sector in CO2ND as defined in CO1ST}.

The following labels are defined for each executable segment. xx is replaced by either two or three letters to differentiate between segments.

xx First word of segment

xxN Segment Number

xxOP Location containing message option selected {first word after end of segment}.

xxIFG Location containing image flag {second word after the segment}.

xxMFG Location containing the main message {third word after the segment}.

DOCUMENT CLASS IMS PAGE NO. 6
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12x2-ED06x4 MACHINE SERIES 1700

xxC1
 xxC2 Cells for inter-segment communication {fourth
 xxC3 through seventh words after the segment}.
 xxC4

xxFIL Block to expand the segment to be SEGSIZ long

The following labels are defined for each message segment.

xxMx First word of the segment
 SxxMx Relative start of the segment
 xxMxN Segment number
 xxMxFL Block to expand segment to be SEGSIZ long

3.1.2 Entry Symbol

COBOP Start of BOOTSTRAP program to write failed image in disk.

3.1.3 External Symbols

SYSSEG The mass memory system director ordinal under which the segments are loaded {CO1ST, CO2ND, etc.}. This must be defined in an *YM statement.

The following may be set up as system initialization time through the use of *S records or as entry points in a core-resident program.

COBOPS Starting sector where failed image is to be written. SYSECT is assumed if unpatched.

COB0PL Length of core to be written by COBOP. The following are normal values.

3FFF	16K
4FFF	20K
5FFF	24K
6FFF	28K
7FFF	32K
9FFF	40K
BFFF	48K
DFFF	56K
FFFF	65K

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 7
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MO12M2-E006M4 MACHINE SERIES 1700

The following externals are referenced by various segments and therefore must be loaded prior to loading the segments.

FNR	INPROC	FMASK*	VOLEND	SPASW	LOADIN
COMPRQ	ALLIN	NDISP*	CALTHD	FILE1	SWTCH
MASKT	MAXSEC	INTSTK	SWAPON	FILE2	SABS
REQXT	LOG1A	CORALC	LVLSTR	FILE3	CABS
VOLR	LOG1	SCHLNG	MIB	FILE4	DISPXX
VOLA	LOG2	SCHTOP	MIBX	UNPTIM	MONI
LUABS	INVINT	VOLBLK	UNPIO	TRANV	NABS
					EXTBV4

* May not be patched, depending on system configuration.

The following externals are standard ordinal names for system modules.

JOBENT	PROTEC	BSUP
JOBPRO	LIBEDT	ODEBUG
JLOAD	RCOVER	LOGGER
JPCHGE	BRKPT	
JPT13	SELF	
JCRDV4	JPFLV4	
JLGOV4	NAMEV4	
JPSTV4	AFILV4	
RESTOR	DATEV4	

3.1.4 Macro Assembler Definitions

STRMSG A - Set up EQU's for executable segments and reserve word used for RTJ. $\nabla A \nabla$, $\nabla A \nabla N$, $\nabla A \nabla OP$, $\nabla A \nabla IFG$, $\nabla A \nabla MFG$, $\nabla A \nabla C1$, $\nabla A \nabla C2$, $\nabla A \nabla C3$, and $\nabla A \nabla C4$ are defined.

STRMSG A - Set up EQU's for message segments. $S \nabla A \nabla$, $\nabla A \nabla$, and $\nabla A \nabla N$ are defined.

CONSTR A, B, C, D, E - Sets up a call to CONVST which picks up $\nabla A \nabla$ words of data from a buffer $\nabla B \nabla$ in the segment starting at $\nabla D \nabla$ and places converted ASCII characters in buffer starting at relative location $\nabla C \nabla$ from XFERTB with $\nabla E \nabla$ words between each value [two characters per word and fours ASCII characters for each value].

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 7-a
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. MO12M2-ED06M4 MACHINE SERIES 1700

LISTIT A- Sets up a call to LISTIT to output the message labeled 'A'.

WRITIT A- Sets up a call to WRITIT.

ANSWER A- Sets up a call to ANSWER.

MESSAG A, B, C, D, E, F, G, H, I, J - Sets up a message starting at relative location 'B' within segment 'A' with an option number of 'C'. The alphanumeric data consists of 'D', 'F', 'H', and 'J'. If 'E', 'G', or 'I' are 1, then the relative {to XFERTB} address of the current word assigned to a label.

DOCUMENT CLASS IMS PAGE NO. 8
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12M2-E006M4 MACHINE SERIES 1700

VBV1, VBV2, or VBV3 respectively. The macro may be terminated with a zero in VE, VG or VI and labels new message VB.

3.2 Program Function

The basic function of the entire package is to assist the system programmer in isolating and correcting E006M4 system malfunctions. The package will provide a consistent and orderly diagnosis in a hard copy form of the pertinent facts at the time of system failure. The program may be run repeatedly on the same failed image, thus allowing the programmer to investigate the problem at several levels of detail.

3.3 Hardware Requirements

3.3.1 Memory Requirements

3.3.1.1 Core Memory

Bootstraps:	COBOP	393 ₁₀ M
	COBOPC	410 ₁₀ M
	COBOPD	61

Allocatable Core: 403₁₀ words required during the entire operation of system checkout.

*These values include a buffer area for a bad sector directory for 30 alternates.

3.3.1.2 Mass Memory

403₁₀ for the storage of SYSCOP program.

5185₁₀ for the storage of the segments - C01ST, etc.

32,768₁₀ for the storage of the failed image. This may be reduced to the system core size and is used only while the Checkout Program is running (i.e., may be part of system scratch).

DOCUMENT CLASS IMS PAGE NO. 9
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12M2-E006M4 MACHINE SERIES 1700

3.3.2 Interfacing Hardware

No special hardware is required, but minimum configuration is identical to E006M4.

3.4 Software Requirements

3.4.1 Computer Operator's Instructions

3.4.1.1 Writing the Failed Image on Mass Memory

When the operator desires to write the image on mass memory using COBOP, the machine must be stepped in the instruction mode, the P register set to the starting location of COBOP, and the M register cleared. Set the Selective Stop Switch and hit Run. When the program stops if $\Phi=0$, the write was okay. Otherwise, a hardware error was detected.

3.4.1.2 Executing SYSCOP

After normal restarting of the system, SYSCOP is called via a System Directory Scheduler Request. Upon entry, the package will request an option to be input. Responses are:

- 0 Go to Dump routine
- 1 Error messages only are to be input
- 2 Errors plus error supporting messages are to be output
- 3 Errors and all supporting messages are to be output
- MZ Exit from the package

When the package completes options 1, 2, or 3, it will again ask for a message selection. Any of the above responses may again be selected.

If 0 is selected DUMP is output and one of the following responses must be input:

- M R Restart package at Option Selection
- M Dxxxx,yyyy Dump cells from xxxx to yyyy from failed image
- M Z Exit from the package

DOCUMENT CLASS IMS PAGE NO. 10
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD122-ED064 MACHINE SERIES _____

3.4.1.3 Loading of Program

The following ordinals must be reserved:

MYM, SYSCOP, XX, SYSSEG, XX

The following externals may be patched:

MS, COBOPS, #XXXX Starting sector fro the failed image.
 MS, COBOPL, #XXXX Length of mass memory write for the failed
 image.
 MS, CRSZV4, #XXXX Physical core size

COBOP must be core resident and loaded prior to SYSSEG.

All of the segments, COLST . . . COLAST, must be loaded under the SYSSEG ordinal.

3.4.2 Relocatability

The program is run-anywhere relocatable.

3.4.3 Analysis

Design of the System Checkout Program was influenced by the following requirements and restrictions.

1. It would be necessary to run in protected core in order to facilitate monitor requests {especially mass memory read requests}.
2. Assembly language would help reduce the amount of core required and also allow more controlled checking of the possible errors.
3. Because of the possible destruction of system addressing, the program is linked to the system independent of pointers.
4. The program is to be run under the system it is checking; thus, emphasis is placed on possible errors which will be uncovered under given circumstances once a system is able to run to some extent {i.e., addition of application software, addition of a new piece of hardware, etc.}

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS TMS PAGE NO. 11
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012x2-ED06x4 MACHINE SERIES 1700

5. The program is to be modular and expandable, since new debugging techniques may be developed as the program is used.
6. A sector oriented mass memory device is assumed and word address requests are not used; thus, all mass memory reads for the program are in 96-word lengths.
7. In order to be able to compact more executable coding in each 96 word block, there are subroutines available to each segment which are core resident during the execution of the program as a part of the executive portion, SYSCOP. Also, all messages are read into the same mass memory buffer area which is used for reading in a sector of the failed or autoloading images.
8. The segments will call for all applicable messages to be output and the output routine will filter out the messages which are not to be output under the option selected.

<u>Option</u>	<u>Messages Output {Or Input}</u>
0	0 CONTROL and INPUT
1	0 CONTROL and INPUT 1 HEADER and ERROR
2	0 CONTROL and INPUT 1 HEADER and ERROR 2 SUPPORTING A SPECIFIC ERROR
3	0 CONTROL and INPUT 1 HEADER and ERROR 2 SUPPORTING A SPECIFIC ERROR 3 SUPPORTING

3.4.4 Program Description

3.4.4.1.1 COBOP--Checkout Bootstrap Program

Function

This program is used to write the current core image on a specified area of disk {853/854}.

DOCUMENT CLASS TMS PAGE NO. 12
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012x2-E006x4 MACHINE SERIES 1700

Requirements and Assumptions

Disk must be on equipment channel 3 {Q=1BX}. COBOPS, COBOP and CRSZV4 must be patched appropriately. COBOP does use system programs as the system status is unknown.

Description

COBOP first inhibits interrupts and saves the A and Q registers for later printout.

Entry is then made to the disk driver with the following register contents

A = address of the BSD buffer
 Q = number of words to read
 I = sector of the BSD

System Checkout initially reads in the bad sector directory {BSD} into BSDBUF and saves it for later reference.

System Checkout then writes the failed image starting at the system sector specified by COBOPS. If the write was successful, the program will loop with Q equal to zero. If COBOPS was undefined the program will loop with Q equal to minus one. If I/O was unsuccessful due to a hardware error the program will loop with Q equal to minus 0 {#FFFF}.

Disk writes are initiated in segments of #3FC0 words maximum.

COBOP requires its own internal disk driver. This is necessary because of sector biasing and the possibility of bad sectors.

The disk driver utilizes the bad sector directory which has been read into BSDBUF. On a write of the failed image the sector number is biased by the size of the autoload area {4 sectors} and the size of the bad sector directory {calculated by BSDSEC}.

Initially the disk driver checks to make sure that an attempt is not being made to write on the alternate area of the disk. If so, exit is made with Q set to -2.

The bad sector directory is checked to determine if the write starts or includes a bad sector. If a bad sector is found, the write is segmented, the alternate written, and then the remainder written until the request is satisfied.

DOCUMENT CLASS IMS PAGE NO. 12-A
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. MD122-EDD64 MACHINE SERIES 1700

3.4.4.1.2 COBOPC - Checkout Bootstrap Program

Function

This program is used to write the current core image on a specified area of a 1739 Cartridge Disk.

Requirements and Assumptions

The cartridge disk must be on equipment channel #D {Q=6BX}. COBOPS, COBOPPL must be patched appropriately. COBOPC does not use any system programs as their status is unknown.

Description

COBOPC writes the image basically the same as COBOPC. The bootstrap has the capability of writing on either the fixed or the removeable disk (one, not both). The starting sector number specified must allow for the image to fit entirely on the disk specified.

3.4.4.1.3 COBOPD - Checkout Bootstrap Program

Function

This program is used to write the current core image on a specified area of a 1751 drum.

Requirements and Assumptions

The drum must be on equipment 2 {Q=1DX}. COBOPS and COBOPPL must be patched appropriately.

Description

COBOPD first inhibits interrupts and saves the A and Q registers for later printout. The starting sector number is checked to see if it is defined, if not, exit is made with Q set to -1.

The core image is written starting at the system sector specified. If any hardware errors occur, the program will loop on a selective stop instruction with Q set to #FFFF. If the write was successful, the program will loop with Q equal to zero.

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 12-B
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. M012x2-E006x4 MACHINE SERIES 1700

3.4.4.2

SYSCOP--System Checkout Program--Control Portion

Function

SYSCOP reserves an area in core for segments which actually perform the checking and also perform the overlaying of the segments. It provides subroutines for use by the segments and an area for use as a mass memory buffer to store a sector of messages of words from an image.

Description

The Executive or Control portion initializes the I register which points to XFERTB and enters the segment. When the

CONTROL DATA CORPORATION
ANALOG-DIGITAL SYSTEMS _____ **DIVISION**

DOCUMENT CLASS IMS PAGE NO. 13
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M01241-EP0642 MACHINE SERIES 1700

segment returns control, the next segment is specified by the increment in the Q register and is read into the SEGMNT area from mass memory. The cycle is then repeated.

Initially, some one-shot calculations are set up by a segment called SETUP. This segment is read in as part of SYSCOP and the area it occupies is overlaid by future segments. When setting up absolute addresses, the autoloading image sector and failed image sector addresses are also initialized. The following messages are then output using LST.

SYSCOP START

IMAGE START SECTOR IS XXXX

The first checkout segment is then loaded and placed in execution.

The following subroutines are part of SYSCOP and are entered via an RTJ- N,I where N is an index in XFERTB. The I register is always restored and A and Q are passed as indicated for each routine.

<u>Name</u>	<u>Index</u>	<u>Description</u>
EXITCO	{#22}	This routine is entered via a JMP- {#22},I and releases the core allocated to SYSCOP. Terminates all checkout. NOTE: Cell #22 contains a zero.
LDA	1	This subroutine passes the caller the contents, from the failed image, of the address as specified in the A register. The value is passed in the A register and Q is restored. The sector length buffer MMBUF is used in reading from the image.
CON	3	This subroutine converts the hexadecimal value contained in the A register to four ASCII characters and passes them in A and Q. The A register contains the two MSB.

Example:

IN	OUT
A=1234	A=3132 Q=3334

CONTROL DATA CORPORATION
 ANALOG-DIGITAL SYSTEMS DIVISION

DOCUMENT CLASS IMS PAGE NO. 14
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012*1-E006*2 MACHINE SERIES 1700

<u>Name</u>	<u>Index</u>	<u>Description</u>
LST	5	This subroutine outputs a message {Format Write} on the standard print logical unit {specified in cell #FB of LOCORE}. The A register contains the length of the message and Q contains the starting location in MMBUF {relative to the XFERTB}. Before outputting, the first word {not part of output} is compared to the selected option. If the value is greater than the selected option, the output is not done. Neither A nor Q are restored.
WRT	7	This subroutine is identical to LST except the output, if any, is via the standard comment logical unit {#FC}.
ANS	9	This subroutine is identical to LST except a message is input {Format Read} via the standard comment input device {#FD}.
ATA	11	This subroutine is identical to LDA except the value is picked up from the autoloading image.
CNS	13	This subroutine converts a list of values starting at the address contained in the A register {relative to SEGMNT} and stores the ASCII {2 words per value} characters into a message area starting at the address specified in Q {relative to XFERTB}. The number of words and the number of spaces between the message words are in the two words following the RTJ. Control is returned to the third word. Neither A nor Q are restored.

IN
 A=LIST
 Q=MESSAGE

RTJ- CNS, I
 NUM N N=Number of values in LIST.
 NUM S+1 S=Number of words to be skipped
 between each value's characters.

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS TMS PAGE NO. 15
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012M2-E006M4 MACHINE SERIES 1700

<u>NAME</u>	<u>INDEX</u>	<u>DESCRIPTION</u>
GMS	15	This subroutine places the segment whose number is in the A register into MMBUF. This segment number is relative to the current segment in SEGMNT. This is used to bring message sectors into core for possible output. Neither A nor Q are restored.
MOP	17	This subroutine checks to see if the main error message for this area of checking has been output. The A register contains the length of the main message, and SEGIFG contains the length to be added to the length. This allows a distinction between images being checked {autoload or failed images}. SEGMFG is maintained to assure the message output only once. Neither A nor Q are restored. Before calling this routine, the main message to be output must be in core and must start at MMBUF {i.e., first in mass memory message sector now in core}.
CMP	19	This subroutine compares two values passed in A and Q. The values are assumed to be 16 bit. If A is greater than Q, exit is made with the difference in A. If A is equal to Q, exit is made with A set to zero. If A is less than Q, exit is made with A set to \$FFFF.

3.4.4.3

SYSSEG--System Checkout Segements

Functions

These segments actually perform the system checkout. There are also included message segments which contain the various messages.

General

The segments are broken into several problems: C01ST¹, C02ND, C03RD, and C0LAST. This was done to accomodate use of LIBEDT in loading, and for possible expansion. All of these programs are loaded under one system directory ordinal {SYSSEG} and are run in the order loaded {i.e., C01ST calls in the next sector after its end which is the first sector in C02ND}. C0LAST is a single sector program which causes control to be returned to the first sector in C01ST.

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 15-A
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. M012x2-E006x4 MACHINE SERIES 1700

The only ordering actually required is that COLIST must be first and COLAST must be last. Note the first sector of each program is an executable sector (i.e., not messages).

Description

The following is a description of the various segments. The description is divided by areas checked, as an area may

DOCUMENT CLASS IMS PAGE NO. 16
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012*1-E006*2 MACHINE SERIES 1700

require several sector overlays to complete. After each description is a list of the various messages which may be output by the segments. These messages are divided into various types {See 3.4.3, Item 8}.

1. Option Selection--OP

This segment initializes the common cells of the segments and requests the operator to select a mode of operation. {See section 3.4.1.2.} The requested mode is then started. If the segments are to be run, the registers' contents are output and segment WLC is called. If the Dump option was selected, DP is called.

Messages--OPMX

OPM1	SELECT OPTION	Control
OPM2	A Q I REGISTER XXXX XXXX XXXX	Support
OPM3	FINISH SYSCOP	Control
OPA1	XX	Input

2. LOCORE Analysis--WLC, XLC, YLC

This segment performs the following analysis on both the failed and autoloading images. A checksum of cells 2 through #46 is taken and compared to #1E. Certain communication addresses are compared to the loaded values. The core bounds are checked for reasonableness. MAXSEC is checked for negative addresses or non-zero MSB. Upon completion segment PL is called.

Messages--LCMX

LCM1	***LOCORE CONSTANT ERROR INITIALLY	Header
LCM2	***LOCORE CONSTANT ERROR	Header
LCM3	BIT TABLE CHECKSUM ERROR	Error
LCM4	ADDRESS IN XX WAS XXXX BUT SHOULD BE XXXX	Error
LCM5	MAX CORE WAS XXXX WITH XXXX TO XXXX UNPROT {ERROR}	Error

DOCUMENT CLASS IMS PAGE NO. 17
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M01,2x1-F006x2 MACHINE SERIES 1700

LCM6	MAX CORE WAS XXXX WITH XXXX TO XXXX UNPROT	Support
LCM7	MAXSEC WAS XXXXXXXX {ERROR}	Error
LCM8	MAXSEC WAS XXXXXXXX	Support

3. Priority Level Analysis--PL

This segment checks the priority level of the failed image. A legal level is -1 to #F. In addition, if the level is above 3, the operator is informed of a possible hang up. Upon completion, the next segment loaded immediately after COLST is initiated.

Messages--PLMX

PLM1	***POSSIBLE LEVEL HANGUP	Header
PLM2	PRI LVL WAS XXXX {ERROR}	Error
PLM3	PRI LVL WAS XXXX	Support

4. Core Dump--DP, WDP

This segment processes the inputs as explained in 3.4.1.2. The cells dumped are from the failed image. If the last address is less than the starting address, only the starting cell will be dumped. The output is 8 cells per line if teletype is used, otherwise 16 cells per line.

Messages--DPMX

DPM1	DUMP	Control
DPM2	XXXX XXXX XXXX . . . XXXX	Control
DPM3	FINISH SYSCOP	Control
DPA1	XXXX . . . X	Input

5. Interrupt Trap Analysis--IT, WIT, XIT, YIT

This segment checks both the failed images and autoloader images for the following errors: Unpatched response addresses in the Interrupt Trap Region, Line Zero not set up for Parity/Protect, Output of the location which was last interrupted {failed image only}, Invalid interrupt lines, Printout of line vs. level, FORTRAN

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 18
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12-2-E006-4 MACHINE SERIES 1700

and interrupt level interference, incorrect usage of NDISP.

Messages--ITMX, ITMY, ITMZ

ITM1	***INTERRUPT TRAP ERROR INITIALLY	Header
ITM2	***INTERRUPT TRAP ERROR	Header
ITM3	LINE XX RESPONSE IS UNPATCHED	Error
ITM4	LINE 0 IS NOT SETUP FOR PARITY/PROTECT	Error
ITM5	LINE XX LAST INTERRUPTED XXXX {INVALID}	Error
ITM6	LINE XX LAST INTERRUPTED XXXX	Support
ITM7	LINE 0 1 2 3 4 5 6 7 8 9 A B C D E F LEVEL X X X X X X X X X X X X X X X	Support
ITM8	LEVEL XX IS USED FOR INTERRUPTS AND IS RESERVED FOR FORTRAN	Error
ITM9	FORTRAN LEVELS X X X . . . X {ERROR}	Error
ITMA	FORTRAN LEVELS	Support
ITMB	SYSTEM USING NDISP WITH REENT FORTRAN {ERROR}	Error

.6 Interrupt Stack and Mask Table Analysis--IS, XIS, YIS

This segment performs the following checks and comparisons on the failed and autoload images. The Mask Table is set up in relationship to the Interrupt Trap Region. Comparison between failed and autoload Mask Tables. Output of levels in the Interrupt Stack from the failed image and assurance that length of stack is valid. At completion UC is called.

Messages--ISMX, ISMY

ISM1	***MASK TABLE ERROR INITIALLY	Header
ISM2	***MASK TABLE ERROR	Header
ISM3	LINE XX IS SET FOR LVL XXXX BUT XXABLE TO INTERRUPT XXXX	Error

DOCUMENT CLASS IMS PAGE NO. 19
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12M1-ED06M2 MACHINE SERIES 1700

ISM4	ENTRY FOR LVL XXXX INITIALLY XXXX CHANGED TO XXXX	Error
ISM5	INTERRUPT STACK LEVELS XXXX . . .	Error or Support

7. Logical Unit Capability--UC, WUC

This segment checks the standard devices as specified in LOCORE for specified capabilities {read/write} and alternate logical unit devices for at least as much capability as the original devices. Both autoloader and failed images are checked. LU is called at completion.

Messages--UCMX

UCM1	***LOGICAL UNIT CAPABILITY ERROR INITIALLY	Header
UCM2	***LOGICAL UNIT CAPABILITY ERROR	Header
UCM3	SXX IS NOT AN XXXX DEVICE	Error
UCM4	LU XX IS ALTERNATE FOR XX, BUT HAS LESS CAPABILITY	Error

8. Logical Unit Analysis--LU, WLU, XLU, YLU, ZLU

The following logical unit table analysis is performed for both the failed and autoloader images. The number of logical unit entries in each table is compared. Logical Unit One is checked to ensure that it is the Core Allocator. Each LOGIA entry is checked to assure it points to a physical device table. The marked down logical units are output for the failed image only. Shared devices are checked for match-up and non-shared devices are checked for non-matches. For the failed image, the devices which are assigned to a logical unit have the current parameter list output. The logical unit threads are output. The last location to use FNR and COMPRQ are listed. At completion, the next segment loaded after COEND is called.

DOCUMENT CLASS IMS PAGE NO. 20
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M01201-E00602 MACHINE SERIES 1700

Messages--LUMX, LUMY, LUMZ

LUM1	***LOGICAL UNIT TABLE ERROR INITIALLY	Header
LUM2	***LOGICAL UNIT TABLE ERROR	Header
LUM3	NUM OF LUS DO NOT AGREE, ASSUME XX	Error
LUM4	LU 1 NOT CORE ALLOCATOR	Error
LUM5	NO VALID PHYSTB FOR LU XX	Error
LUM6	LU XX WAS MARKED DOWN	Support
LUM7	LU XX IS SHARED BUT UNMATCHED	Error
LUM8	LU XX AND XX MATCH BUT SHARED BIT NOT SET	Error
LUM9	LU XX CURRENT PARA LIST AT XXXX	Support
LUMA	RC XXXX TH XXXX LU XXXX N XXXX S XXXX	Support
LUMB	I/O IN PROGRESS	Support
LUMC	LU XX THREAD	Support
LUMD	XXXX XXXX . . . XXXX	Support
LUME	LU XX THREAD MAY BE BROKEN	Error
LUMF	RETURN FOR XXR WAS XXXX	Support

9. Scheduler Stack Analysis--SS, WSS

This segment checks the failed image to assure the Scheduler Stack levels are valid and outputs the stack entries. The last scheduled program location, if available, is output. The amount of volatile available and in use is also output. AC is called at completion.

Messages--SSMX, SSMY

SSM1	***SCHEDULER STACK ERROR	Header
SSM2	NUM OF SCHEDL STACK ENTRIES WAS XX NUM OF SCHEDL CALLS STACKED WAS XX	Support

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 21
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. M012x2-E006x4 MACHINE SERIES 1700

SSM3	SCHEDL STACK ENTRIES	Support
SSM4	XXXX/ XXXX XXXX XXXX XXXX	Support
SSM5	LAST ENTRY TO BE SCHEDULED XXXX/ XXXX XXXX XXXX XXXX	Support
SSM6	THERE WERE XXXX OF THE XXXX VOLATILE WORDS ASSIGNED	Error or Support

10. Allocatable Core Analysis--AC, XAC, YAC

This segment outputs a map of Allocatable Core and checks the block length and threading for errors. It also checks the System Directory for correct request priority levels and assures that programs can fit into the requested level when scheduled. JP is called upon completion.

Messages--ACMX, ACMY

ACM1	***ALLOCATABLE CORE ERROR	Error
ACM2	ALLOCATABLE CORE MAP INDEX START LNTH THRD DUMP	Support
ACM3	XXXX XXXX XXXX XXXX XXXX . . . XXXX	Support
ACM4	***SYSTEM DIRECTORY ERROR	Header
ACM5	INDEX XXXX HAS INVALID REQ PRI XXXX	Error
ACM6	INDEX XXXX TOO LONG FOR REQ PRI XXXX	Error

11. Job Processor Analysis--JP, WJP

This segment analyzes the Core Swap and Job Processor status. The swap status is output from the failed image. File locations for the Job Processor are output. Pending requests are also output:

Messages--JPMX, JPMY, JPMZ

JPM1	CONSIDER UNPROTECTED I/O HANGUP	Header
JPM2	CONSIDER SWAP RATE TOO RAPID	Header

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 22
 PRODUCT NAME SYSTEM CHECKOUT PROGRAM
 PRODUCT MODEL NO. MD12x2-E006x4 MACHINE SERIES 1700

JPM3	CORE USAGE CAUSED SWAP WHILE JP IN	Header
JPM4	SYSTEM WAS SWAPPED	Support
JPM5	SYSTEM NOT SWAPPED	Support
JPM6	SYSTEM NOT SWAPPED BUT WAITING TO SWAP	Support
JPM7	JP NOT IN CORE	Support
JPM8	JP WAS IN CORE	Support
JPM9	FILE1 FILE2 FILE3 FILE4 LOADR BP XXXX XXXX XXXX XXXX :XXXX XXXX	Support
JPMA	JP LOCKED OUT FOR LIBEDT OR RECOVERY	Support
JPMB	SIGN OFF REQUESTED OF LIBEDT OR RECOVERY	Support
JPMC	XXXX UNPROT REQ WERE ACTIVE AND STACKED AT LOC XXXX	Support
JPMD	PENDING INPUT REQUEST FOR JP	Support
JPME	PENDING INPUT REQUEST FOR MIPRO	Support

12. Partition Core Analysis--XPR, YPR, PC

This segment checks the partition core assignments using the failed image. The partition addresses, threads, and use bits are printed as well as selected errors.

Messages--PARX, PCM

PAR1	PARTITION CORE ADDRESSES	Header
PAR2	PARTITION XXXX XXXX	Support
PAR3	PARTITION THREADS	Header
PAR5	PARTITONS IN USE	Header
PCM1	PARTITION CORE ERROR PARTITION 0 ABOVE 8000	Error
PCM2	PARTITION CORE ERROR PARTITION OUT OF ORDER	Error

CONTROL DATA CORPORATION

DIVISION

DOCUMENT CLASS IMS PAGE NO. 22-A
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. M01242-F0064 MACHINE SERIES 1700

PCMB PARTITION CORE ERROR Error
ILLEGAL BUSY INDICATOR

12. Restart of First Segment--COLAST

This segment calls the first segment loaded {OP}.

3.4.4.4 Addition of New Segments

It is desired to include additional segments to the System Checkout Program, the following requirements must be adhered to. Refer to a listing of COLAST.

1. The addition should be assembled as a separate program to be loaded after CO3RD and before COLAST under the ordinal SYSSEG.
2. The Macros and EQU's defined at the start of COLAST should be duplicated and used to set up segments, messages, and calling sequences. {See COLAST}.
3. Each executable segment must be a subroutine and may not make I/O requests without the I-register being restored at completion. The first segment in the program must be an executable segment.
4. Messages must reside in sectors separate from the executable code.
5. Each sector must be exactly 96 words long. {See the variable BZS blocks at the end of each sector}.
6. When the added segment has completed execution, it must relinquish control to the sector loaded immediately following it. {This is done by calling for NEXTN and setting up NEXTN at the end of the program}.

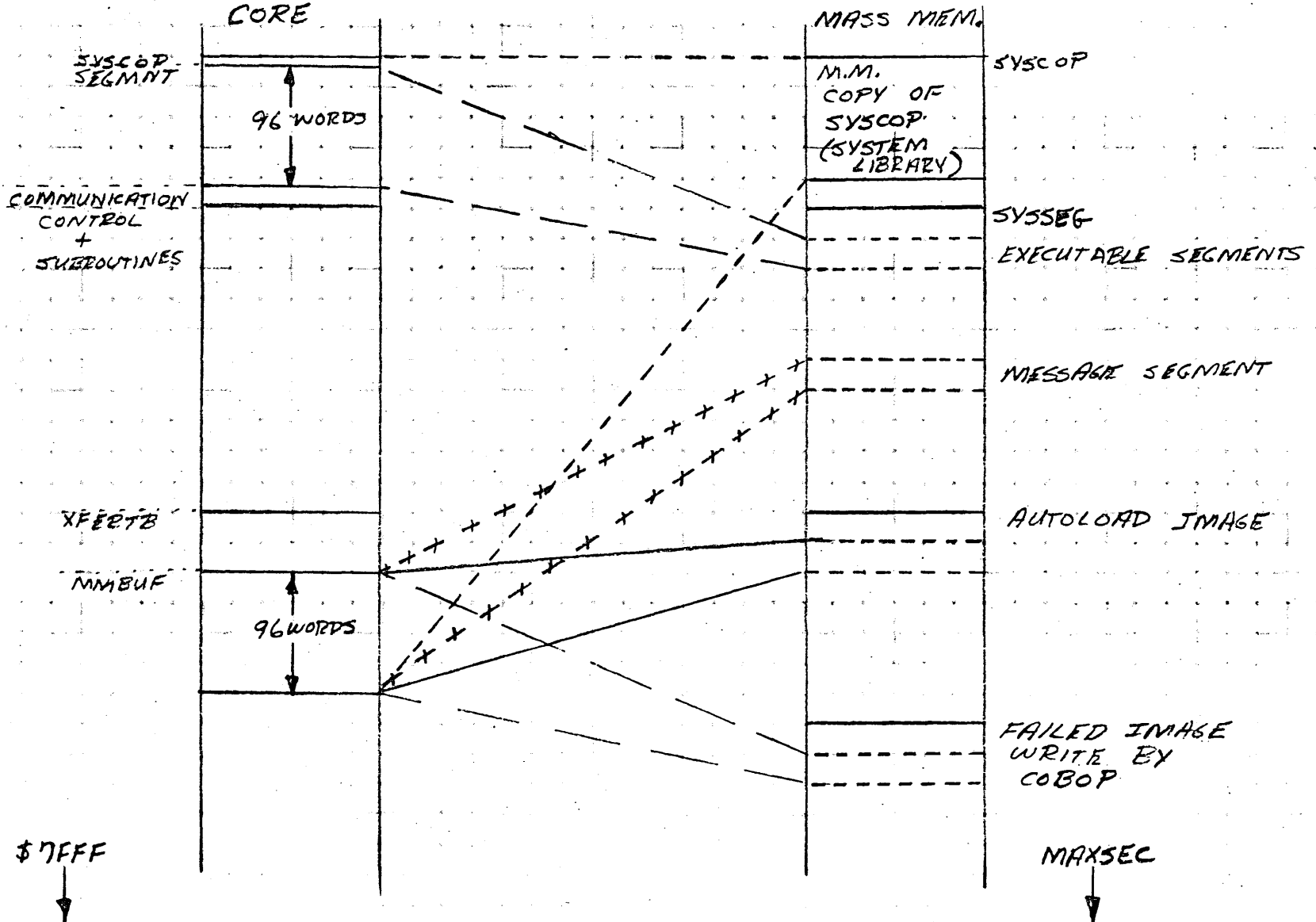
CONTROL DATA CORPORATION
ANALOG-DIGITAL SYSTEMS _____ **DIVISION**

DOCUMENT CLASS IMS PAGE NO. 23
PRODUCT NAME SYSTEM CHECKOUT PROGRAM
PRODUCT MODEL NO. MO12*1-ED06*2 MACHINE SERIES 1700

4. Messages must reside in sectors separate from the executable code.
5. Each sector must be exactly 96 words long. {See the variable BZS blocks at the end of each sector.}
6. When the added segment has completed execution, it must relinquish control to the sector loaded immediately following it. {This is done by calling for NEXTN and setting up NEXTN at the end of the program.}

1 2 3 4 5

MEMORY MAP OF SYSCOP WHILE RUNNING



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

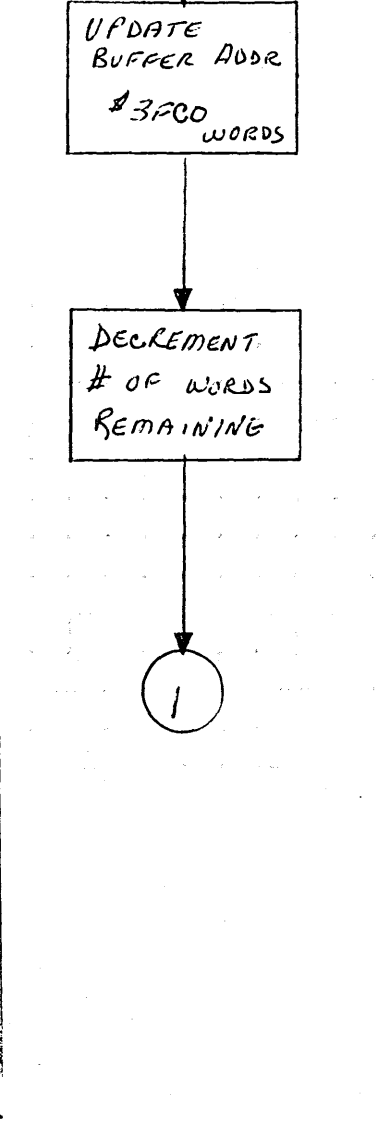
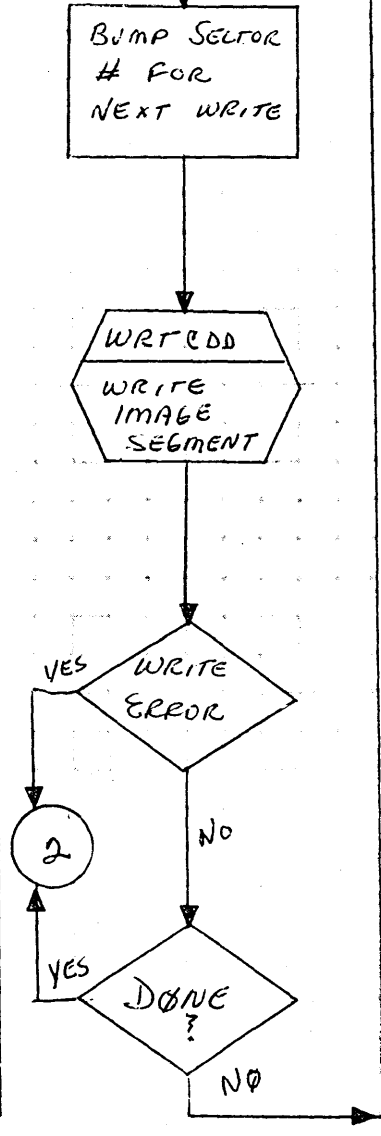
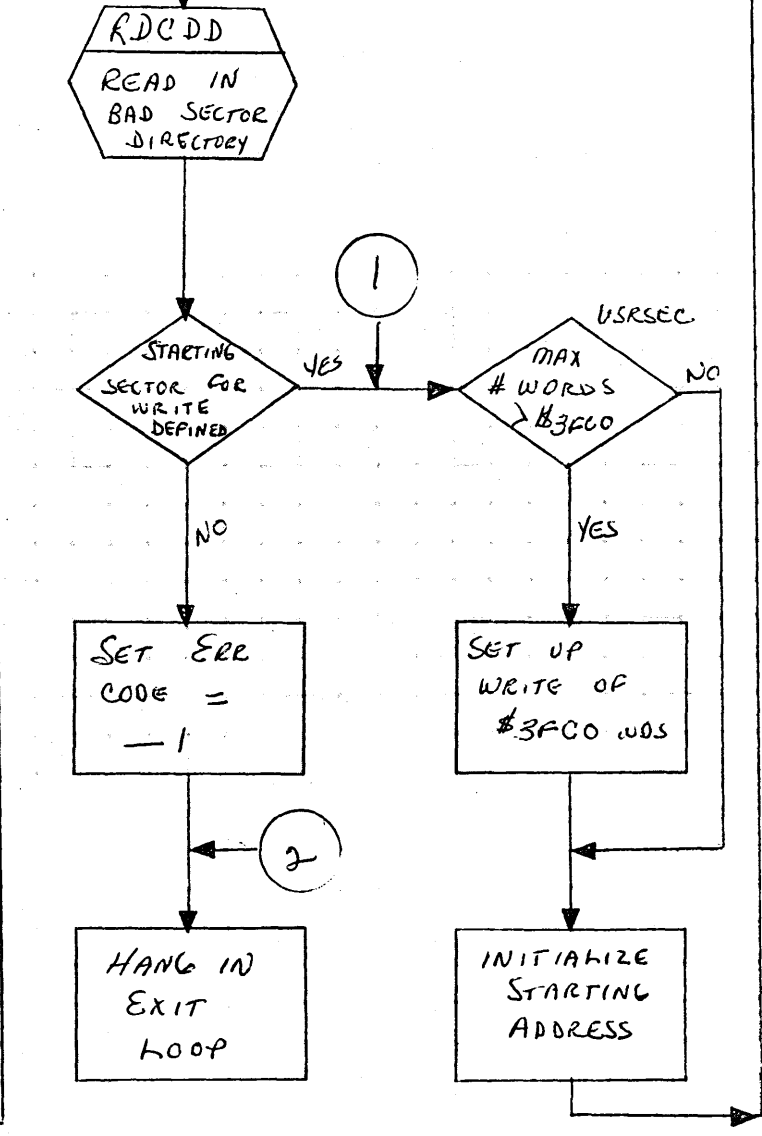
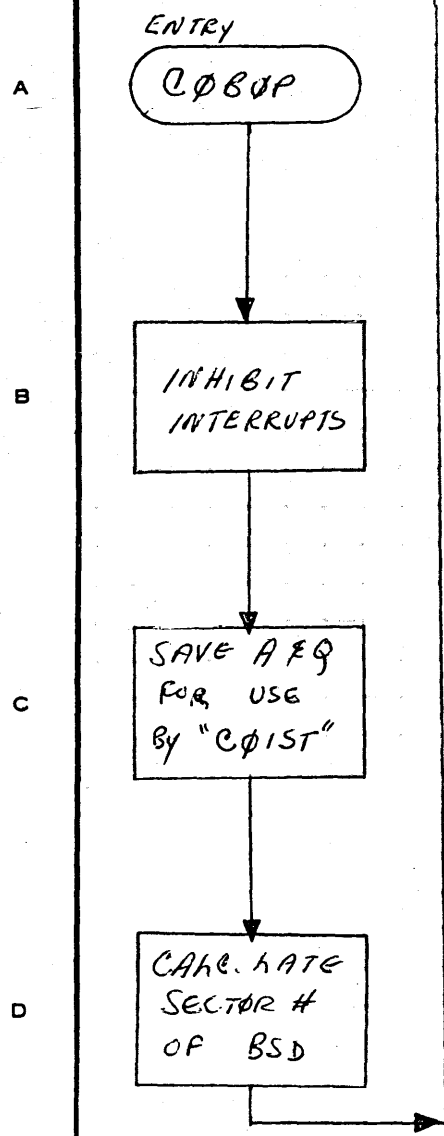
FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT			PROJECT MGR.			
PROGRAM	PAGE	OF		PROJECT NAME			
NUMBER	ISSUE DATE			TASK NO.			
DRAWN BY	DATE			TASK NAME			

CØBØPC (CARTRIDGE DISK BOOT)



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT		
	CØBØPC	PAGE	1 OF
NUMBER		ISSUE DATE	
DRAWN BY		DATE	

PROJECT NO.	
PROJECT MGR.	
PROJECT NAME	
TASK NO.	
TASK NAME	

REV	APPROVED	DATE

25

A
B
C
D

RDCDD

WRTDD

SET READ/
WRITE FLAG
FOR A READ

SET READ/
WRITE FLAG
FOR A WRITE

SAVE FWA
AND SECTOR
NUMBER

READ ?

BIAS SYSTEM
SECTOR
NUMBER

ATTEMPT
TO WRITE ON
ALTERNATE

SET 9
-2 AND
COOP WITH
ERROR

BIGHOP

INITIALIZE
FLAGS

USING
AN
ALTERNATE

ANY
BAD
SECTORS ?

SET
CLOSED
TO ZERO

GETHD
3-5

SECTOP
3-5

ALTERNATE
BAD ?

GET NEW
ALTERNATE

SEKMAX
3-5

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS *IMS* MACH. TYPE *1700*

DOCUMENT TITLE *SYSTEM CHECKOUT*

COBOL PAGE 2 OF

NUMBER ISSUE DATE

DRAWN BY DATE

PROJECT NO.

PROJECT MGR.

PROJECT NAME

TASK NO.

TASK NAME

REV	APPROVED	DATE

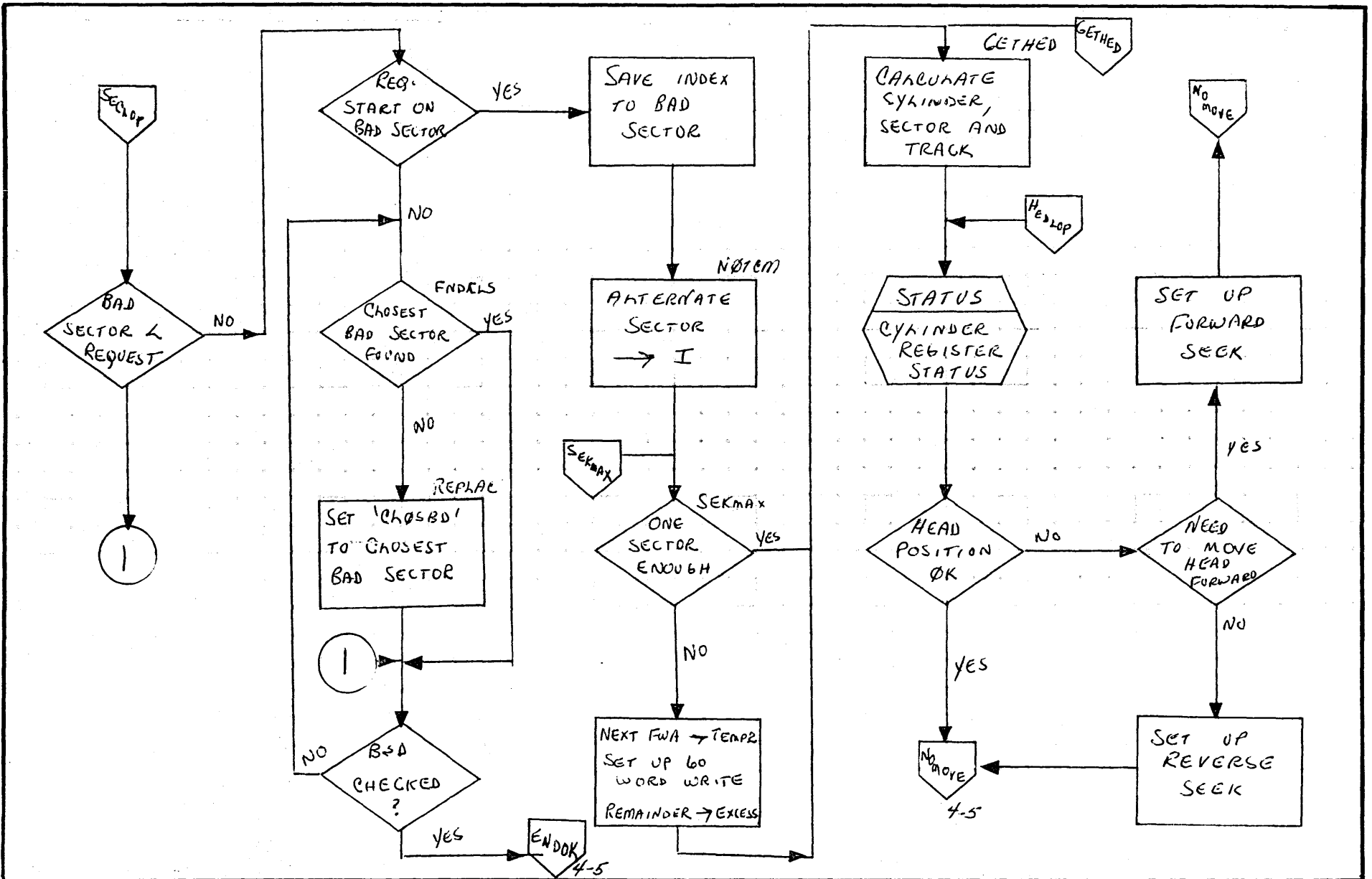
25-A

A

B

C

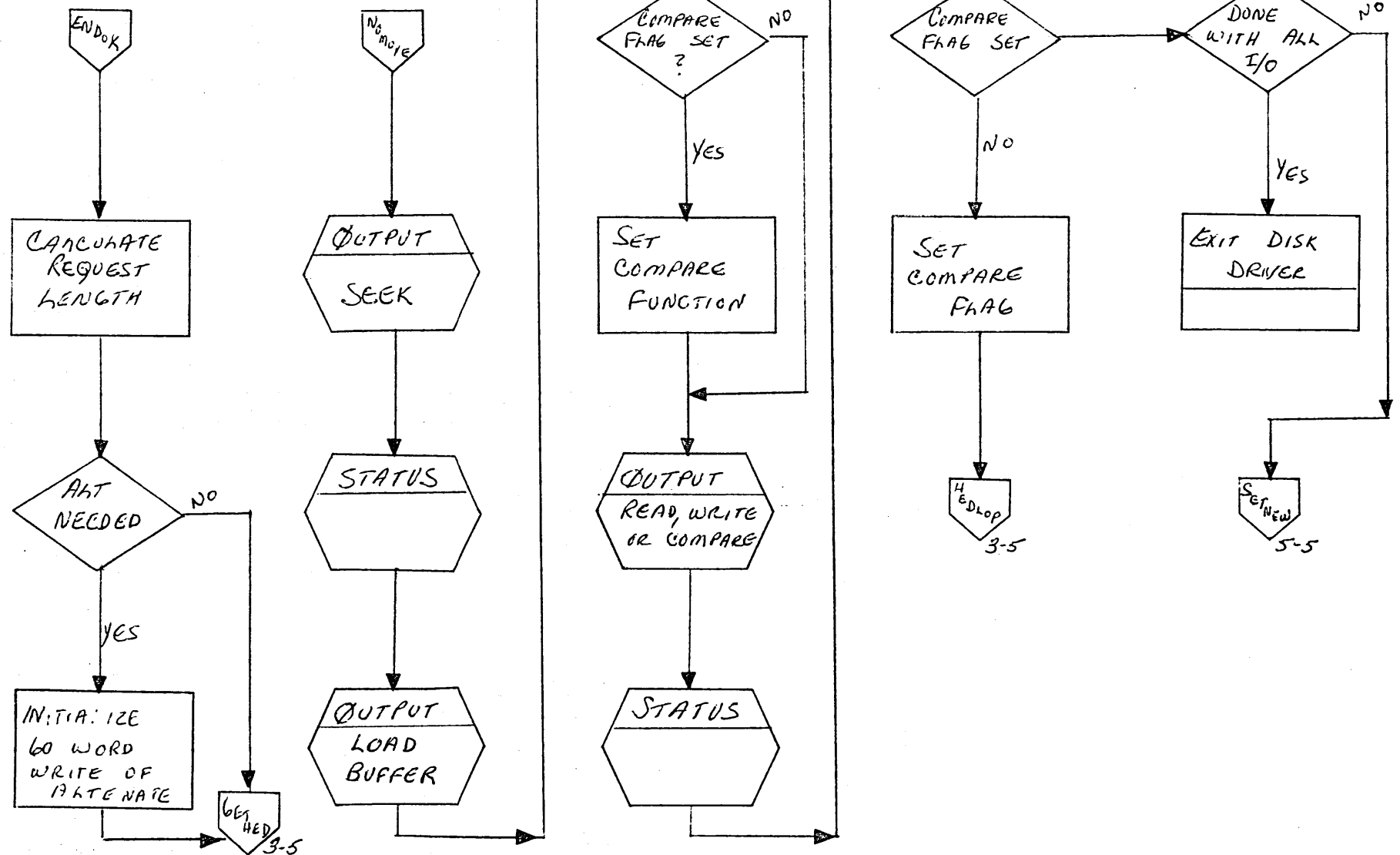
D



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
		COBOL	PAGE 3 OF 5	PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				

25-B

A
B
C
D



CONTROL DATA CORPORATION
 SOFTWARE DOCUMENT

SAMPLE CODE
 FLOWCHART
 DECISION TABLE
 OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT			PROJECT MGR.			
	COBOL	PAGE	4 OF 5	PROJECT NAME			
NUMBER		ISSUE DATE		TASK NO.			
DRAWN BY		DATE		TASK NAME			

25-2

CØBØP (853/854 BOOT)

A

B

C

D

ENTRY
CØBØP

INHIBIT
INTERRUPTS

SAVE A+B
FOR US,
BY CØDIST

1
RDISK
READ IN
BAD SECTOR
DIRECTOR

READ
ERRORS
Yes 1

START
SECTOR FOR
WRITE
DEFINED
Yes 2

SET ERR
CODE =
-1

3
HANG IN
EXIT
LOOP

USRSEC
MAX
WORDS >
#3FC0
NO

SET UP.
WRITE OF
#3FC0 WAS

INITIAHIZE
STARTING
ADDRESS

BUMP SECTOR
FOR
NEXT WRITE

WDISK
WRITE
IMAGE
SEGMENT

WRITE
ERRORS
Yes 3

NO
DONE
Yes NO

UPDATE
BUFFER
ADDRESS.
#3FC0 WDS

DECREMENT
WDS
REMAINING

2

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT
SAMPLE CODE
FLOWCHART
DECISION TABLE
OTHER

DOCUMENT CLASS	JMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE 1 OF 5
ISSUE DATE	CØBØP		
DRAWN BY	DATE		

PROJECT NO.	REV	APPROVED	DATE
PROJECT MGR.			
PROJECT NAME			
TASK NO.			
TASK NAME			

25-7

A

B

C

D

ENTRY

RDISK

WDISK

SET READ/ WRITE FLAG FOR READ

SET READ/ WRITE FLAG FOR A WRITE

SAVE START AND SECTOR #.

READ ?

BIAS SECTOR #

ATTEMPT TO WRITE ON ALT.

BIG LOOP

INITIALIZE FLAGS

USING AN ALTERNATE

ALTERNATE BAD

SET 9 - 2 AND LOOP WITH ERROR

ANY BAD SECTORS

SETNOV 3-5

GET NEW ALTERNATE

SET CLOSED TO ZERO

SEKOP 3-5

CONTROL DATA CORPORATION

SOFTWARE DOCUMENT

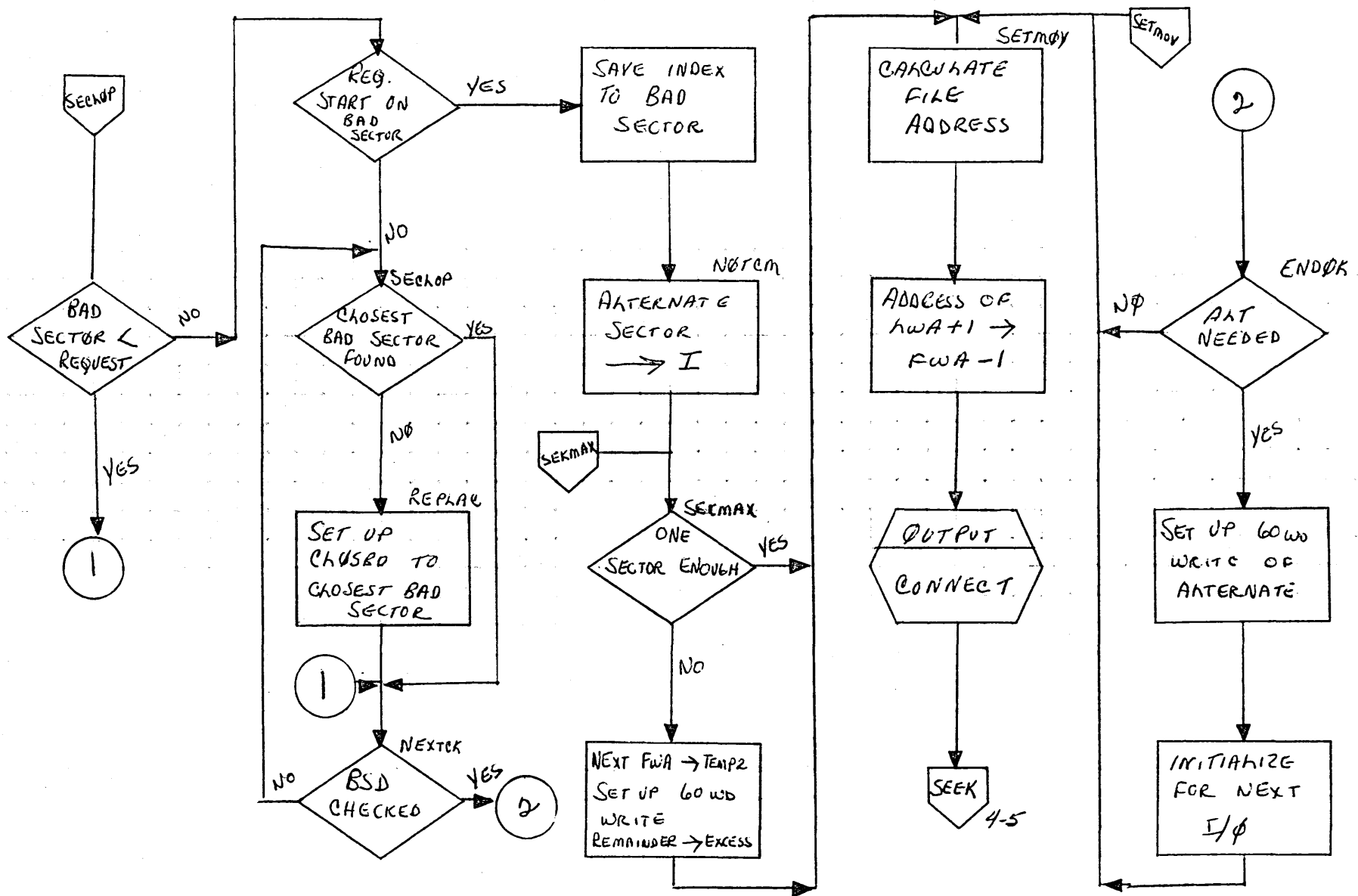
- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		
NUMBER	COBOP	ISSUE DATE	
DRAWN BY		DATE	

PROJECT NO.	REV	APPROVED	DATE
PROJECT MGR.			
PROJECT NAME			
TASK NO.			
TASK NAME			

25-1

A
B
C
D



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM	PAGE	3 OF 5
NUMBER	COBOP	ISSUE DATE	
DRAWN BY		DATE	

PROJECT NO.	REV	APPROVED	DATE
PROJECT MGR.			
PROJECT NAME			
TASK NO.			
TASK NAME			

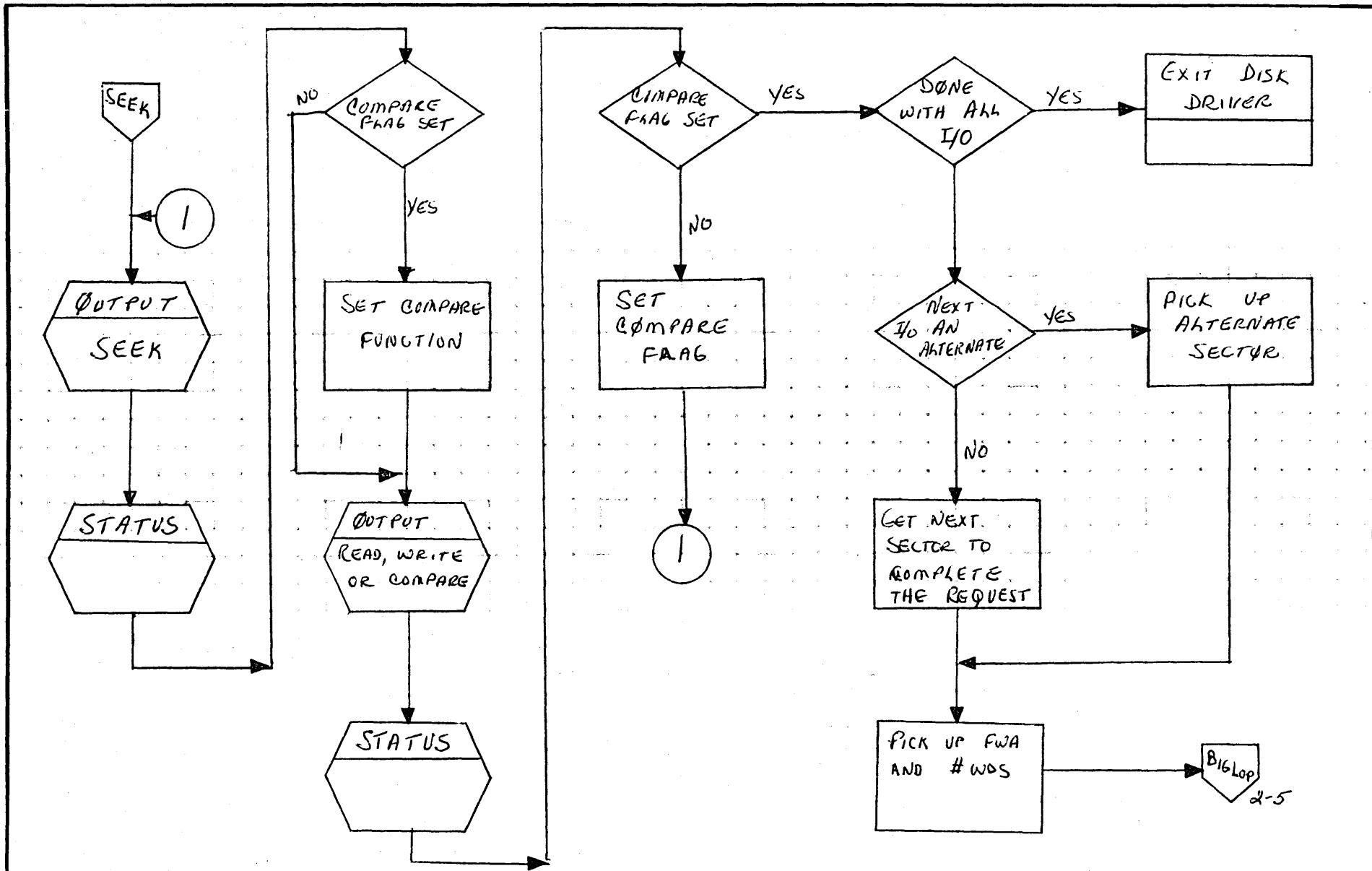
25-6

A

B

C

D



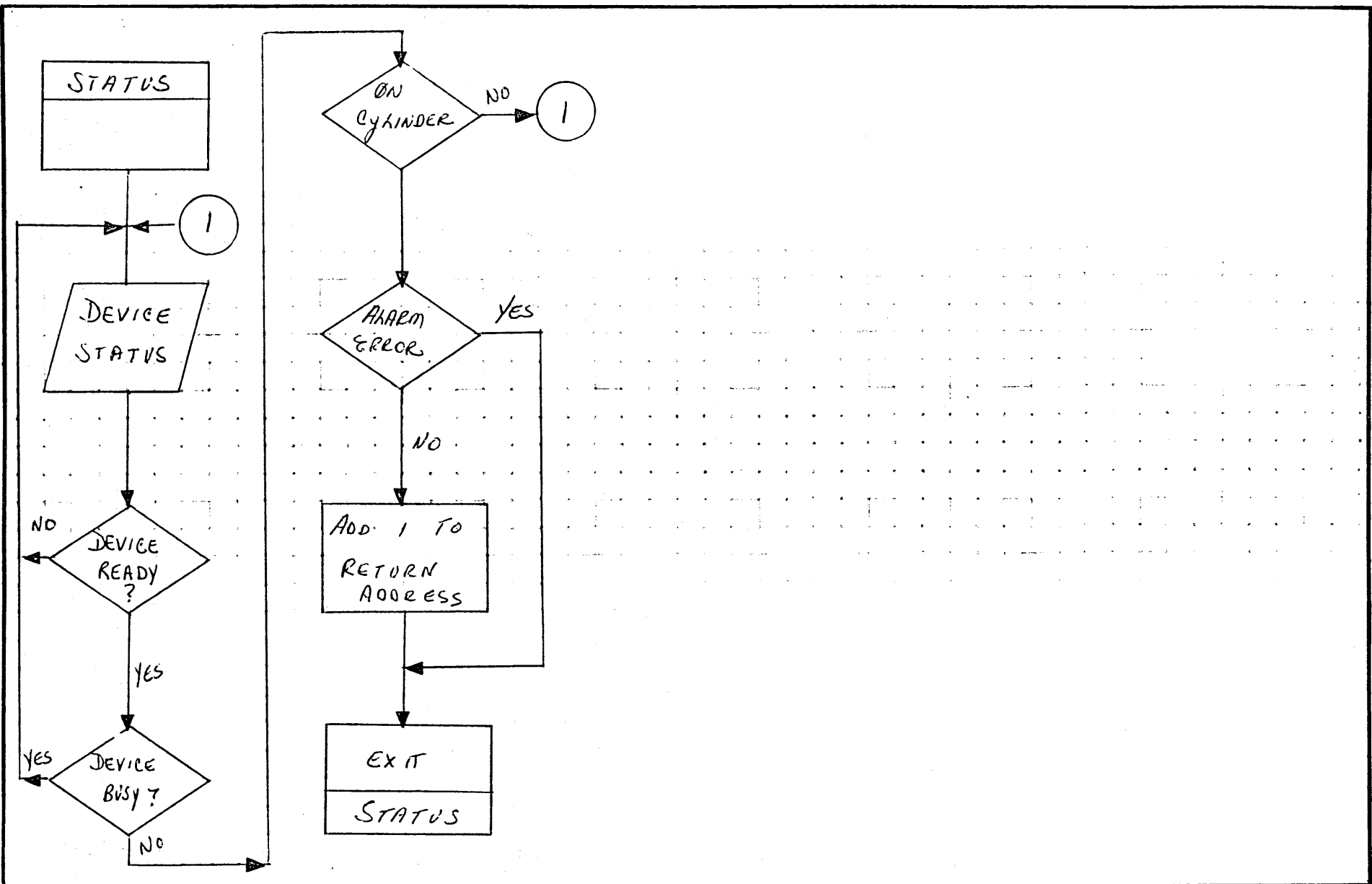
CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1706	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE 4 OF 5	PROJECT MGR.			
NUMBER	COBOP	ISSUE DATE		PROJECT NAME			
DRAWN BY		DATE		TASK NO.			
				TASK NAME			

95-14

A
B
C
D



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

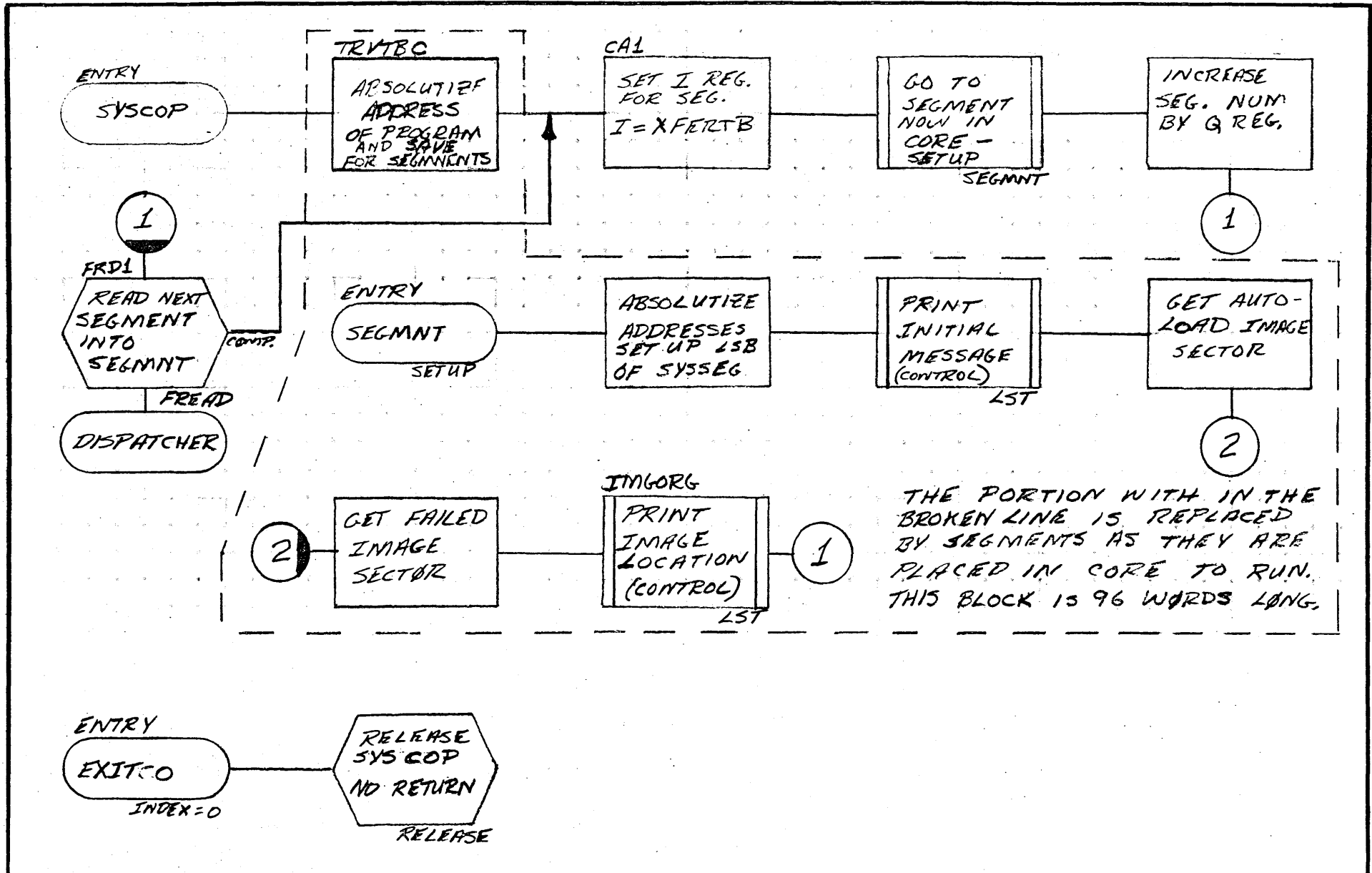
FLOWCHART

DECISION TABLE

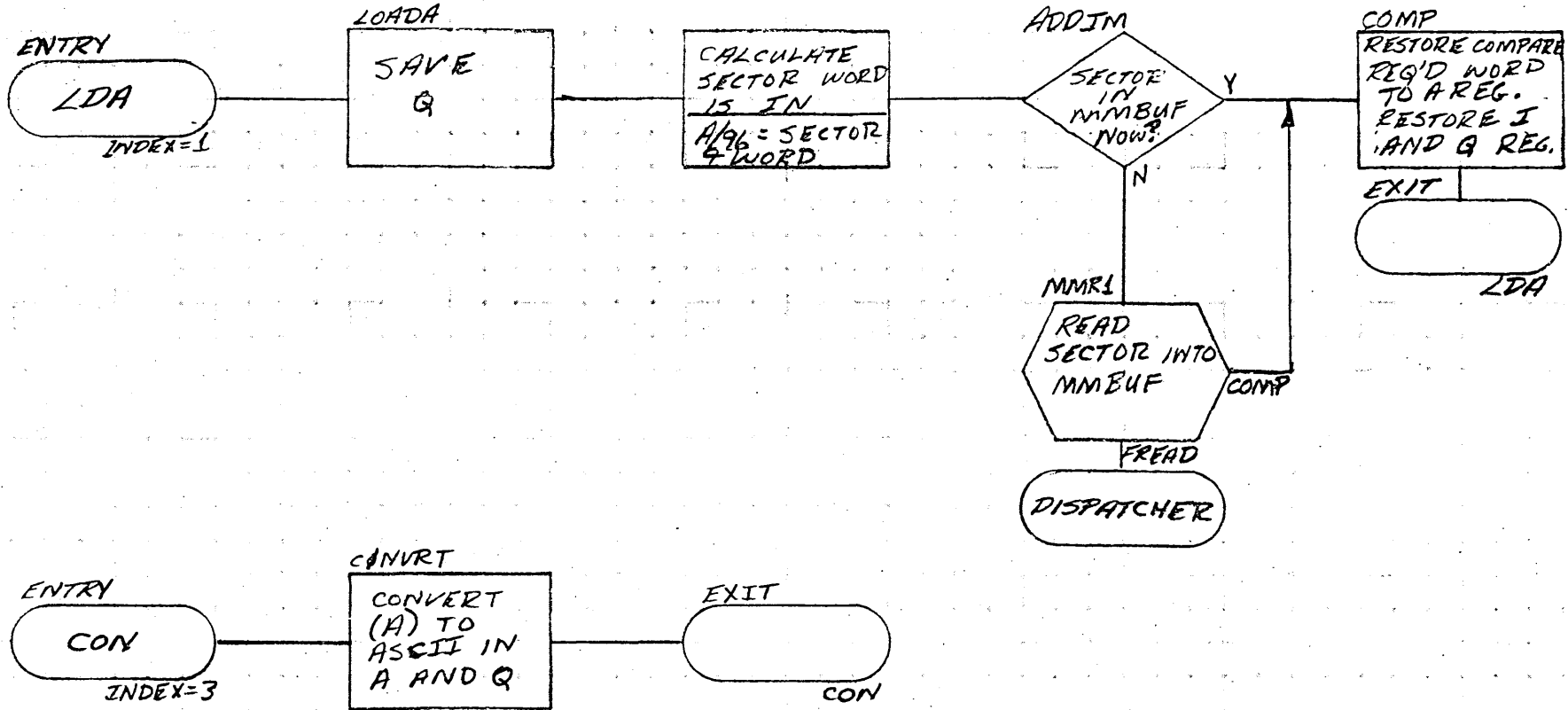
OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM	PAGES OF	5 OF 5	PROJECT MGR.			
NUMBER	COBOP	ISSUE DATE		TASK NO.			
DRAWN BY		DATE		TASK NAME			

25-1



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
	NUMBER		ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

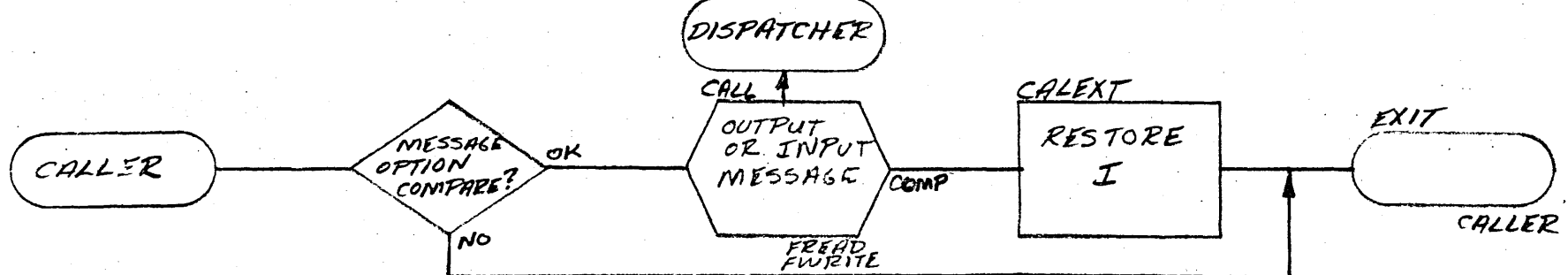
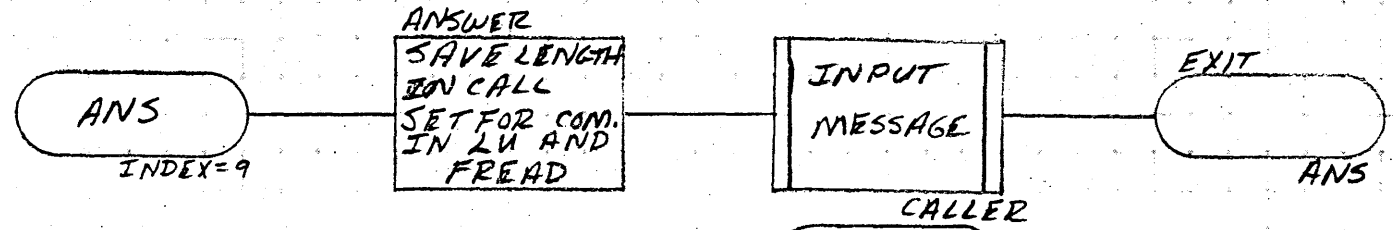
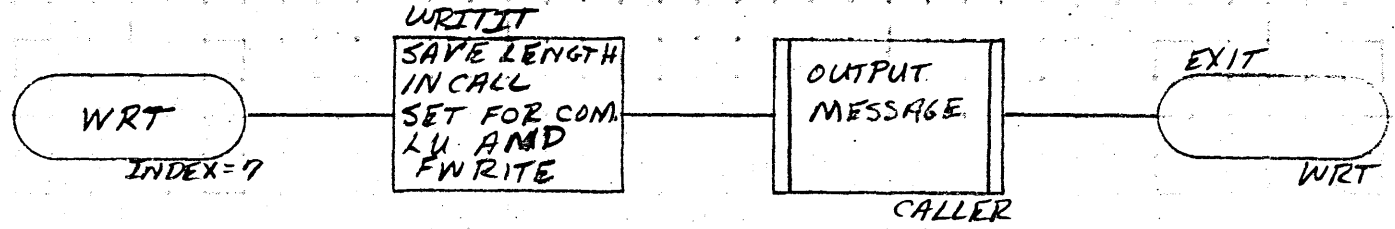
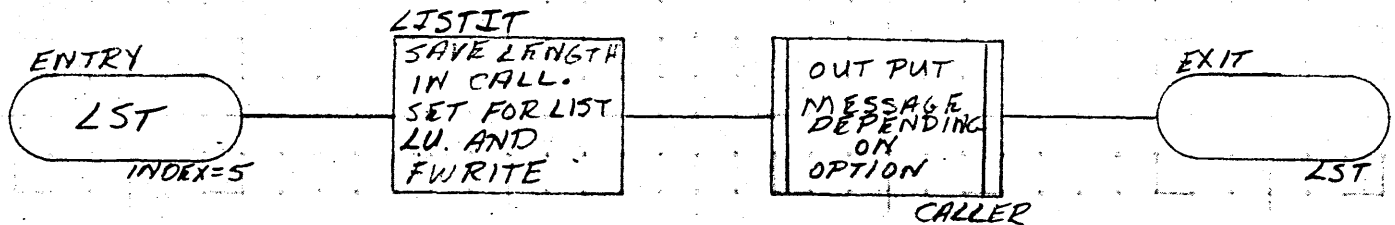
SAMPLE CODE

FLOWCHART

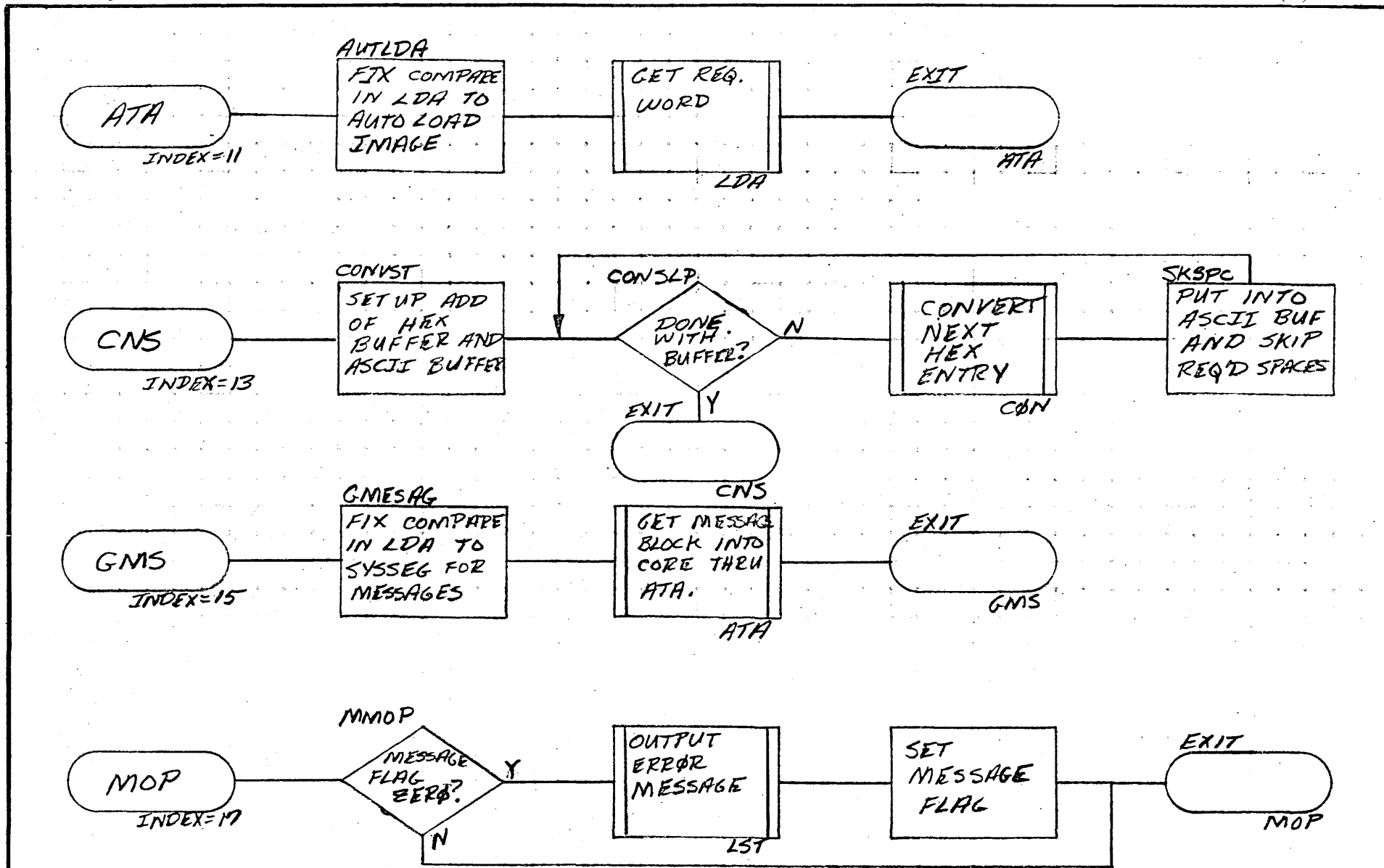
DECISION TABLE

OTHER

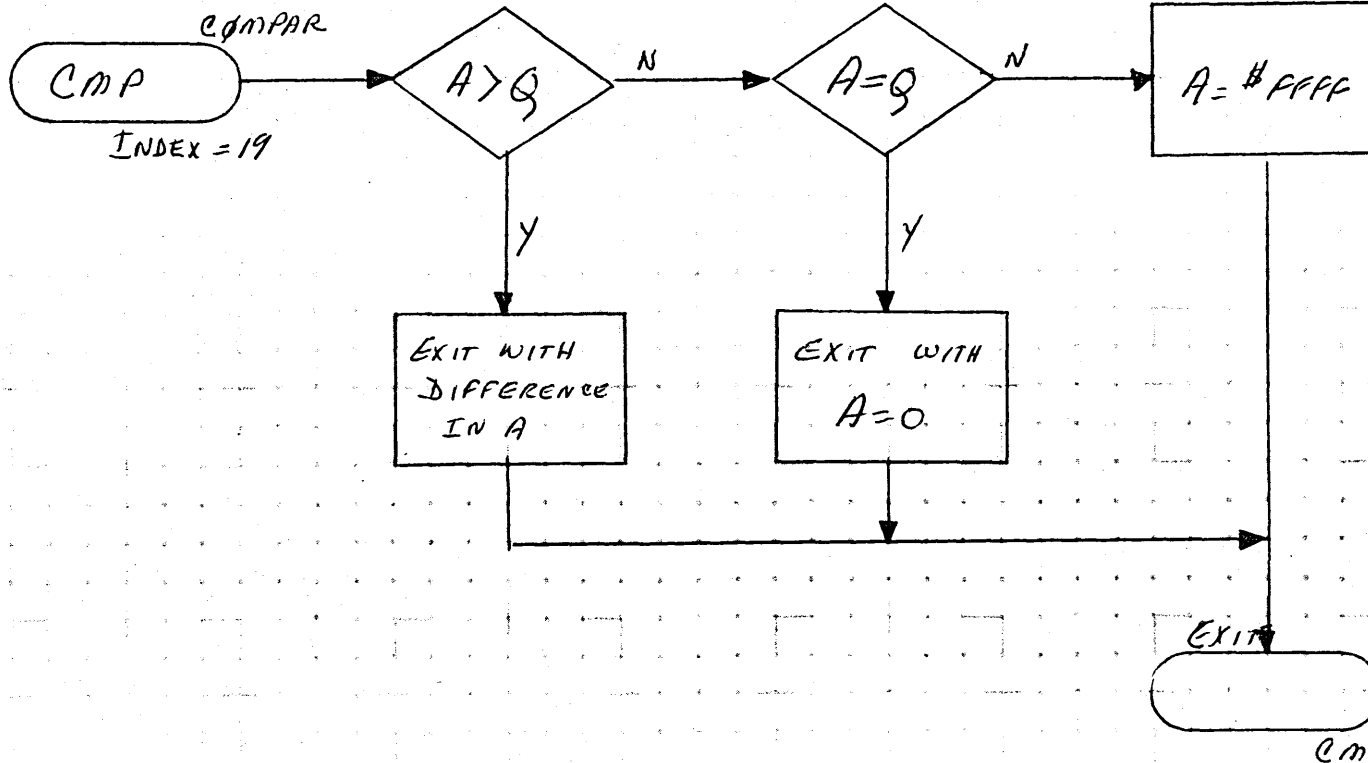
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
NUMBER		ISSUE DATE		PROJECT NAME			
DRAWN BY		DATE		TASK NO.			
				TASK NAME			



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	JMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
	PROGRAM	PAGE	OF	PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	JMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
	PROGRAM	PAGE	OF	PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

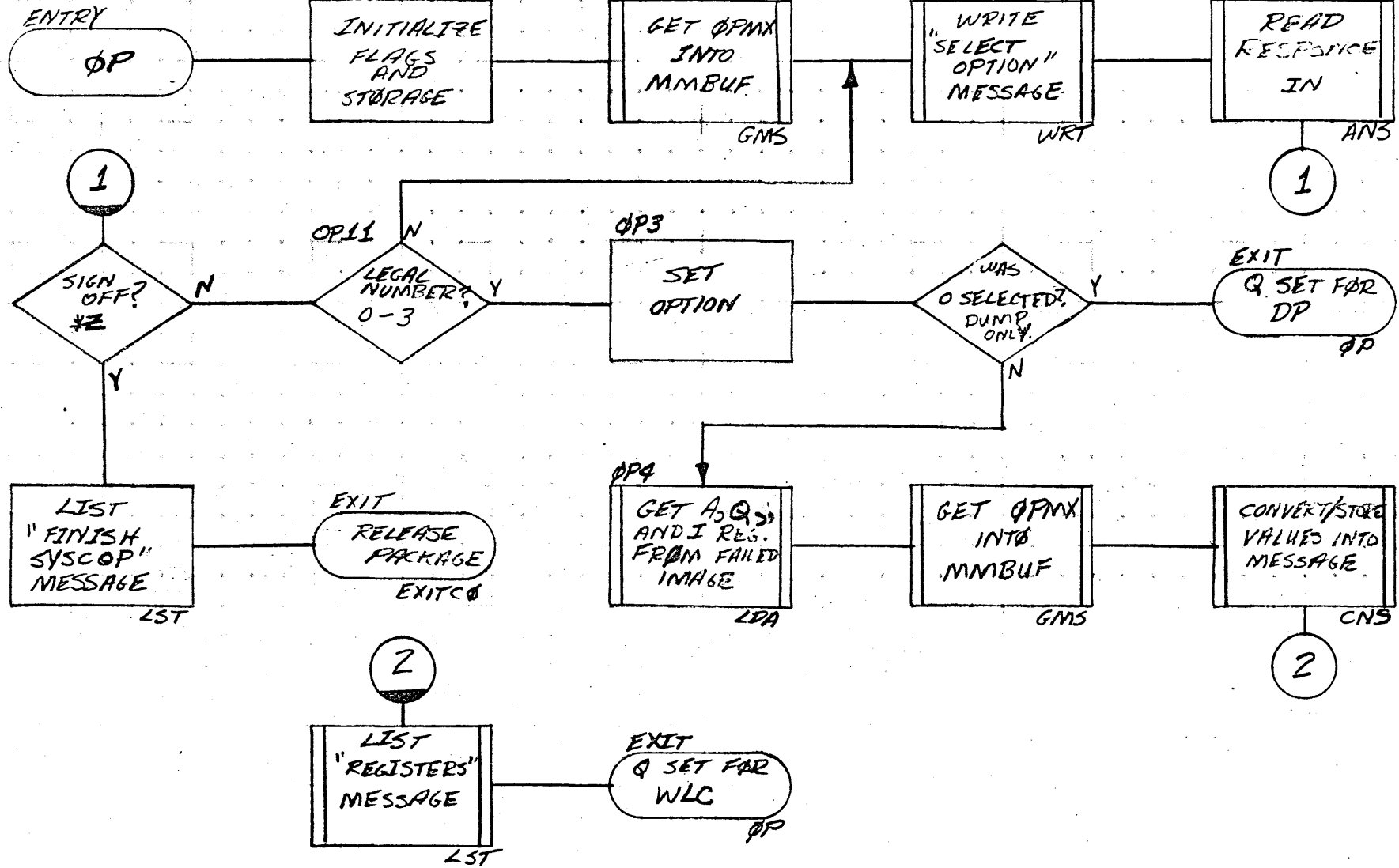
- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		
NUMBER		PAGE OF	
		ISSUE DATE	
DRAWN BY		DATE	

PROJECT NO.	REV	APPROVED	DATE
PROJECT MGR.			
PROJECT NAME			
TASK NO.			
TASK NAME			

H-60

OPTION SELECTION

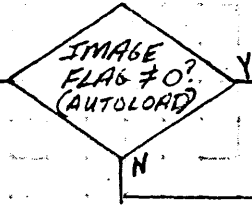


CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PAGE OF	PROJECT MGR.			
	NUMBER	PROGRAM	ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			

LOCORE ANALYSIS

ENTRY

WLC



CHANGE
CALLS FROM
FAILED TO
AUTOLOAD IMAGE
LDA → ATA

LC1

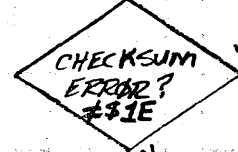
CHECK SUM
CELLS 7-746
ON IMAGE



LDA-ATA

EXIT
Q SET FOR
XLC

LC2



GET MESSAGE
INTO MMBUF
CHECK IF
MAIN MESSAGE
IS OUT

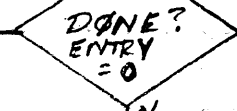
GMS-MQP

LIST
"BIT TABLE
CHECKSUM"
MESSAGE

LST

LC3

GET NEXT
LOCORE
ADDRESS
FROM TABLE



LC313

PROCESS ENTRY
NEW ADDRESS
OR
COMPARE WITH
TABLE ENTRY

LDA-ATA

LC311



LC33

GET LCMX
INTO MMBUF
CHECK IF
MAIN MESSAGE
IS OUT

GMS-MQP

CONVERT/STORE
VALUES INTO
MESSAGE

CON-CNS

LIST
"ADDRESS
ERROR"
MESSAGE.

LST

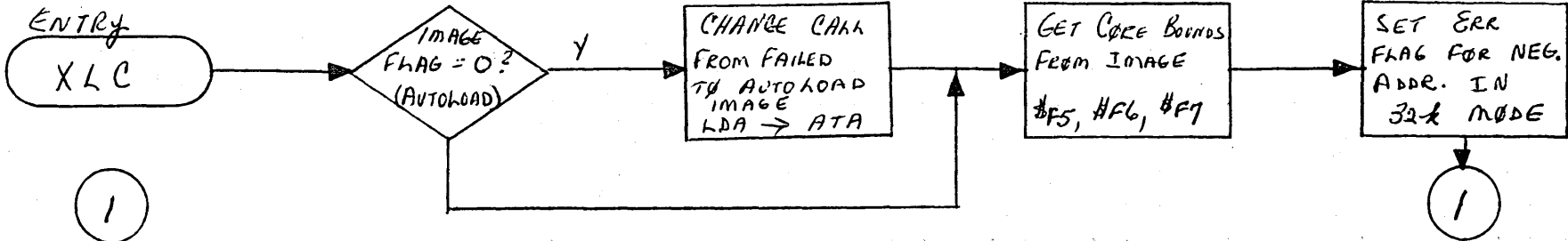
CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE
FLOWCHART
DECISION TABLE
OTHER

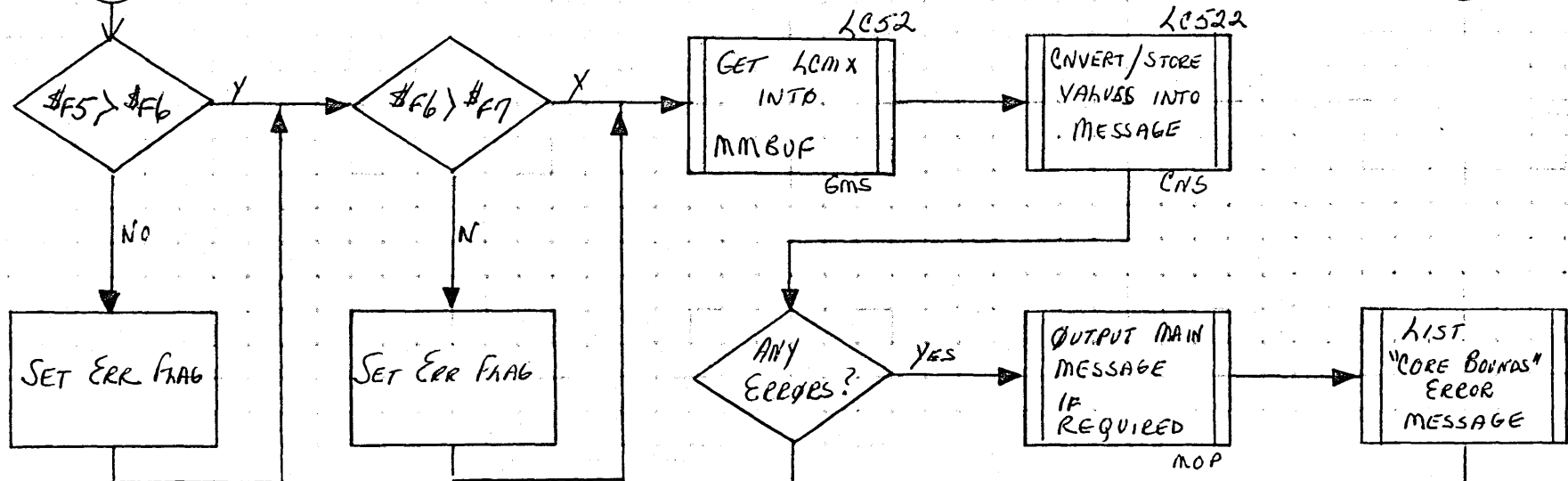
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT			PROJECT MGR.			
PROGRAM		PAGE OF		PROJECT NAME			
NUMBER		ISSUE DATE		TASK NO.			
DRAWN BY		DATE		TASK NAME			

LOADRE ANALYSIS - PART X

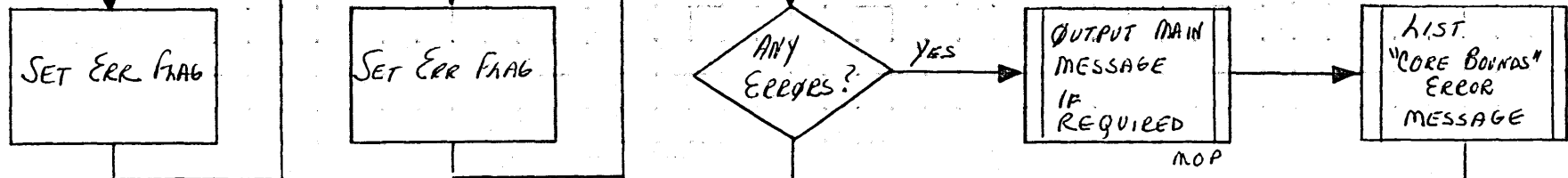
A



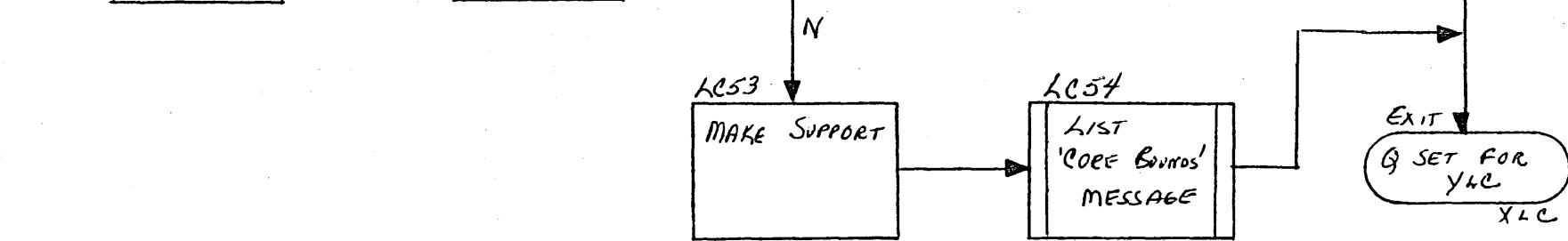
B



C



D



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	MACH. TYPE	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE		PROJECT MGR.			
		PAGE OF	PROJECT NAME			
	NUMBER	ISSUE DATE	TASK NO.			
	DRAWN BY	DATE	TASK NAME			

22

LC CORE ANALYSIS PART Y

ENTRY
YLC

IMAGE
FLAG #0?
(AUTOLOAD)
Y
N

CHANGE CALLS
FROM FAILED
TO AUTOLOAD
IMAGE
LDA → ATA

LC6
PICK TWO
WORDS
OF MAXSEC
LDA-ATA

GET CORRECT
MESSAGE
BLOCK INTO
CORE (MMBUP)
GMS

1

CONVERT/STORE
VALUES
INTO
MESSAGE
CNS

ERROR?
MSB #0
LSECO
Y
N

LC621
OUTPUT MAIN
MESSAGE IF
NOT ALREADY
OUT
MOP

OUTPUT
"MAXSEC
ERROR"
MESSAGE
LST

2

LC63
FIX MESSAGE
FOR SUPPORT
AND
RELATED
(2 OR 3)

LC65
OUTPUT
"MAXSEC"
MESSAGE
LST

2

LC7
AUTOLOAD
DONE?
Y
N
EXIT

LC74
EXIT
SET Q FOR
PL
YLC

EXIT
SET Q FOR
WLC
YLC

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

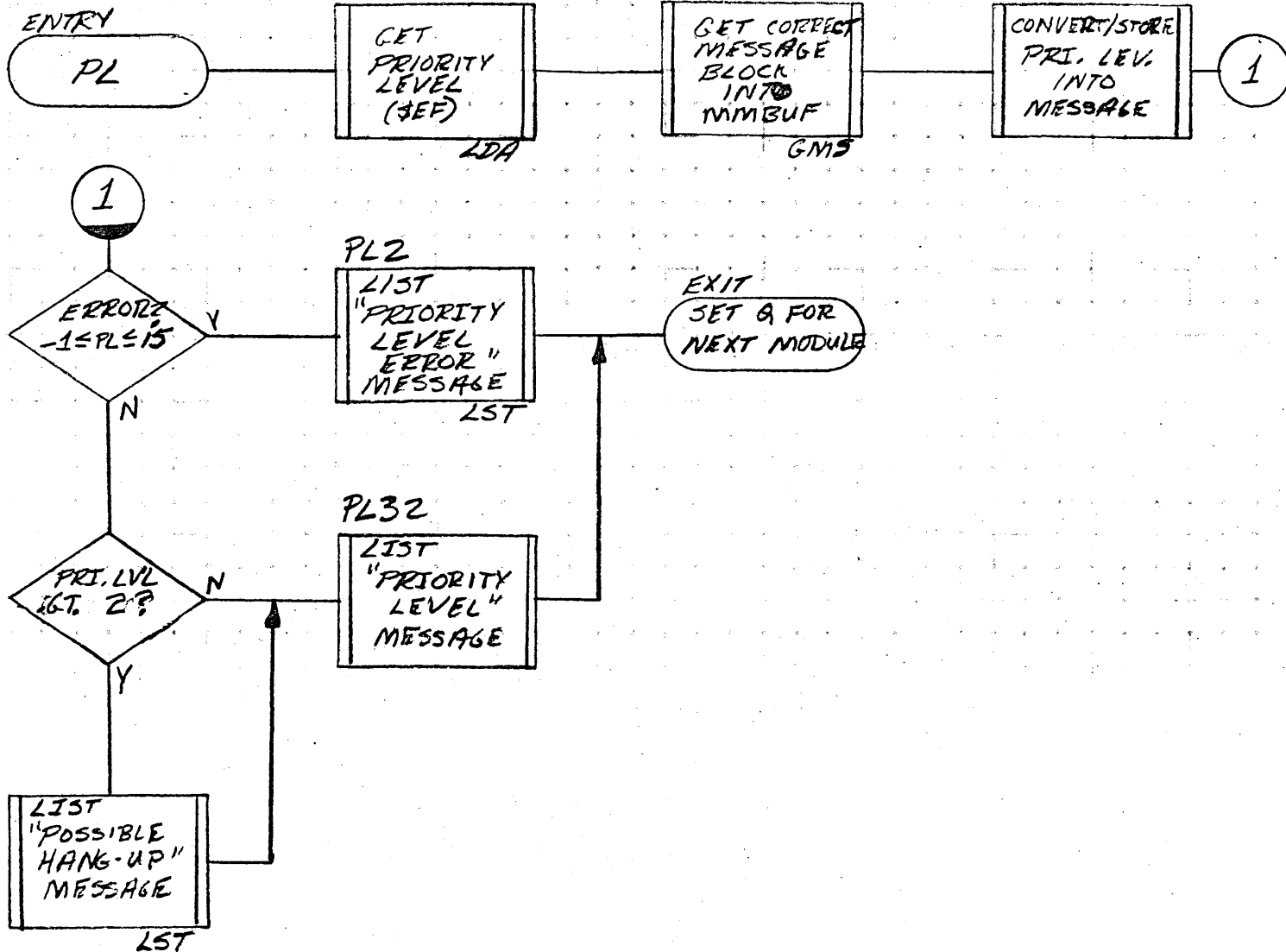
SAMPLE CODE
FLOWCHART
DECISION TABLE
OTHER

DOCUMENT CLASS **IMS** MACH. TYPE **1700**
DOCUMENT TITLE **SYSTEM CHECKOUT PROGRAM** PAGE OF
NUMBER ISSUE DATE
DRAWN BY DATE

PROJECT NO.
PROJECT MGR.
PROJECT NAME
TASK NO.
TASK NAME

REV APPROVED DATE

PRIORITY LEVEL ANALYSIS



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

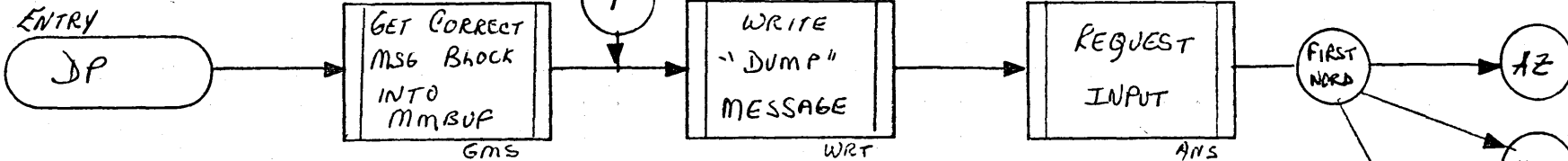
DECISION TABLE

OTHER

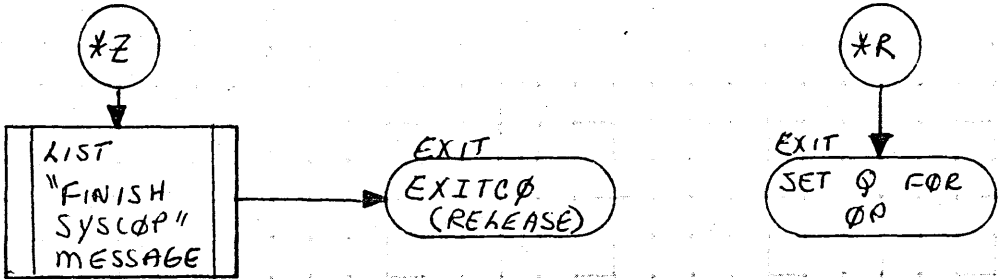
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT	PROJECT MGR.					
PROGRAM	PAGE OF	PROJECT NAME					
NUMBER	ISSUE DATE	TASK NO.					
DRAWN BY	DATE	TASK NAME					

CORE DUMP

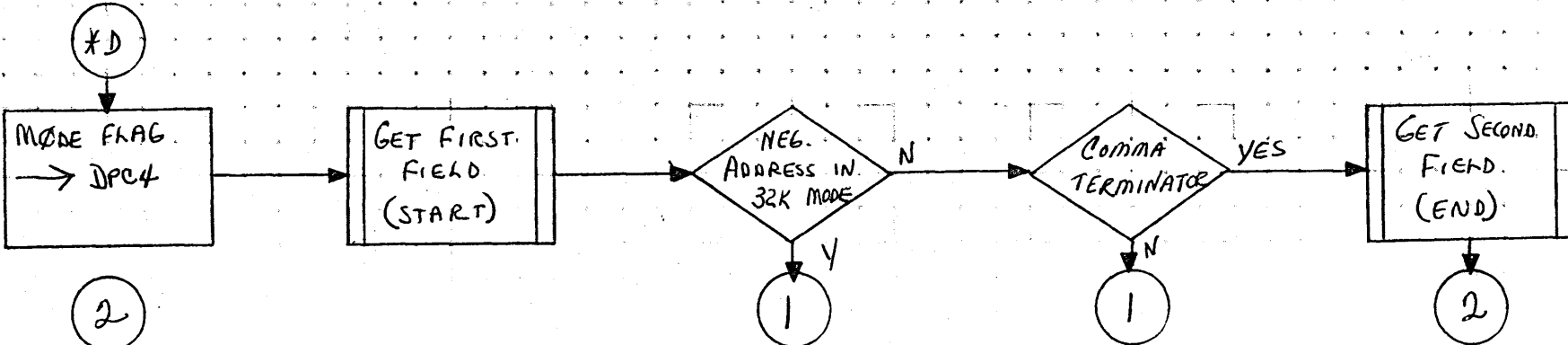
A



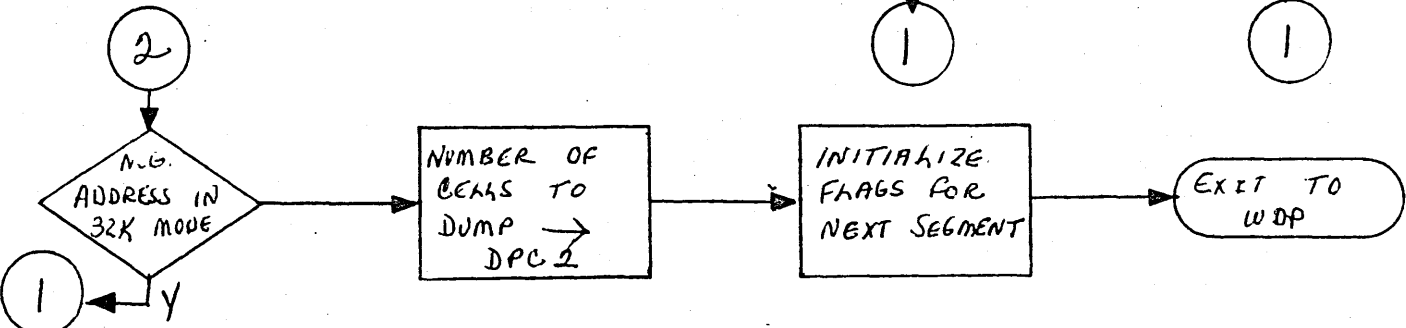
B



C



D



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	MACH. TYPE	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE		PROJECT MGR.			
	NUMBER	ISSUE DATE	PROJECT NAME			
			TASK NO.			
	DRAWN BY	DATE	TASK NAME			

35

ENTRY
DP46FD

DP46FI
GET NEXT
FIELD AND
SAVE

VALID
HEX CHAR.

CONVERT AND
COMBINE
VALUE

PASS VALUE
IN A

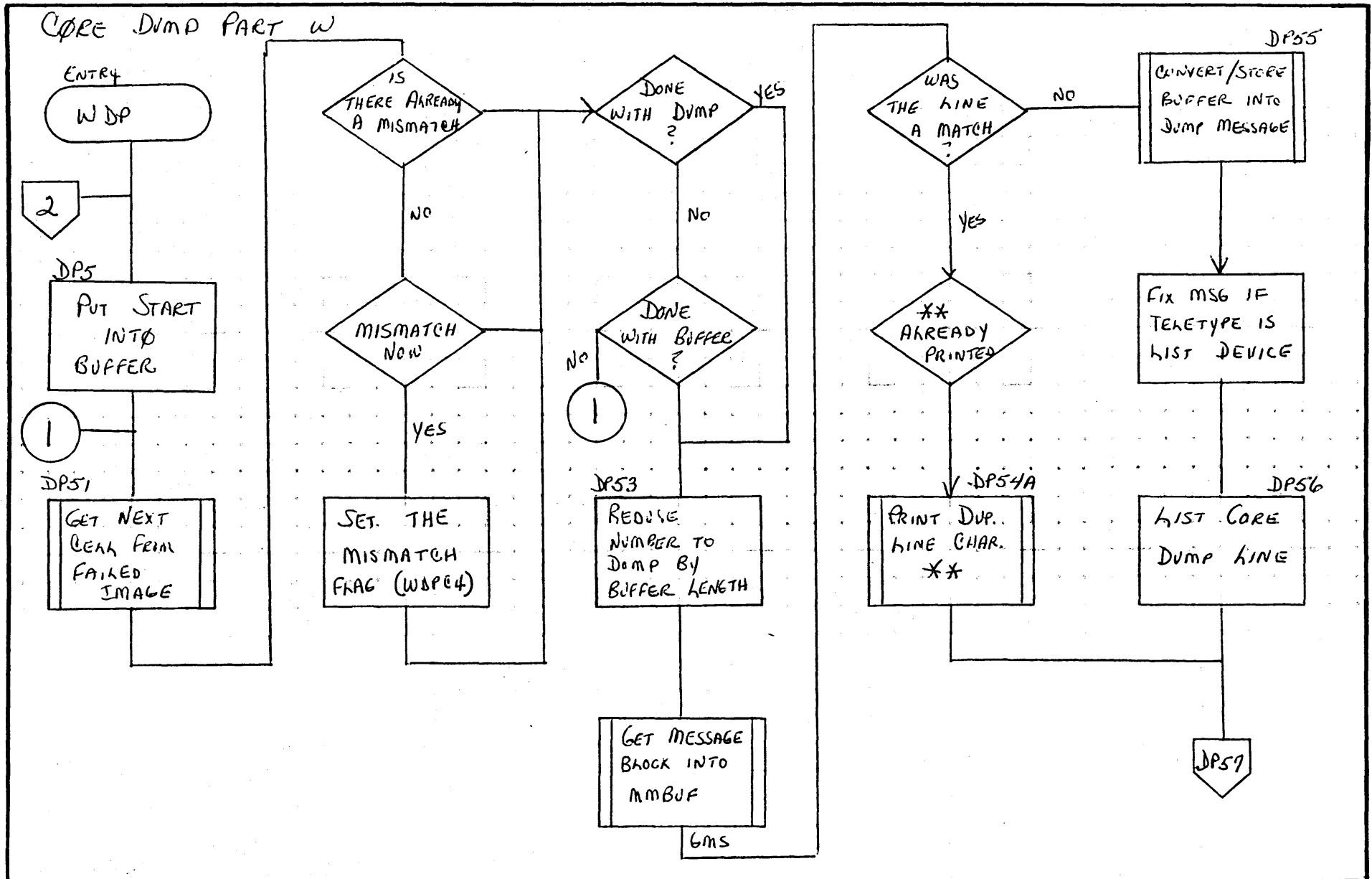
A
B
C
D

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE
 FLOWCHART
 DECISION TABLE
 OTHER

DOCUMENT CLASS <i>IMS</i>	MACH. TYPE <i>1700</i>	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE <i>SYSTEM CHECKOUT PROGRAM</i>	PAGE OF	PROJECT MGR.			
NUMBER	ISSUE DATE	TASK NO.			
DRAWN BY	DATE	TASK NAME			

35-A



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE
FLOWCHART
DECISION TABLE
OTHER

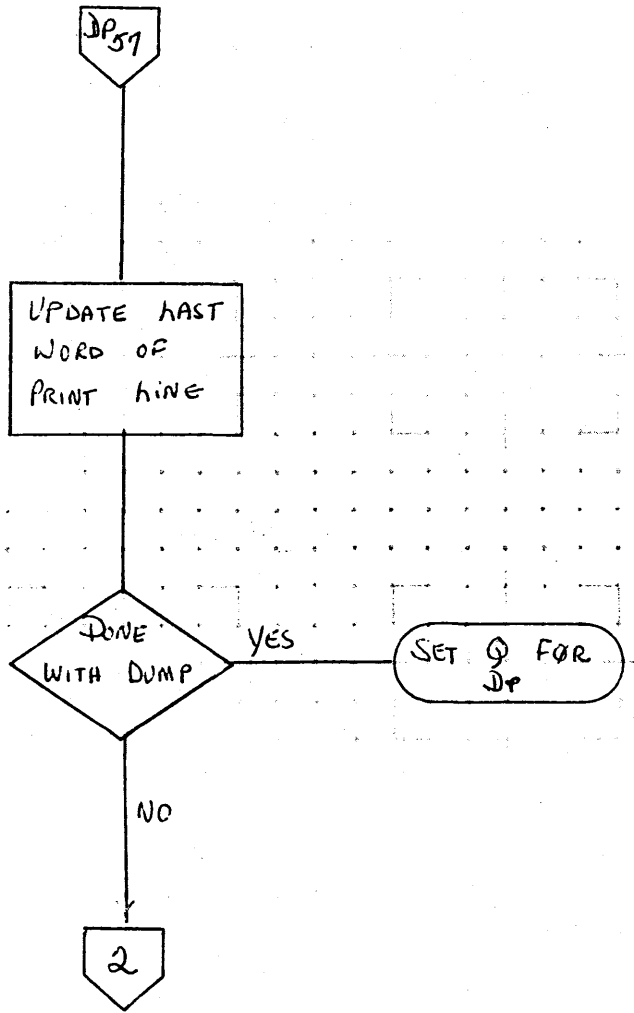
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
NUMBER	ISSUE DATE			PROJECT NAME			
DRAWN BY	DATE			TASK NO.			
				TASK NAME			

A

B

C

D



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

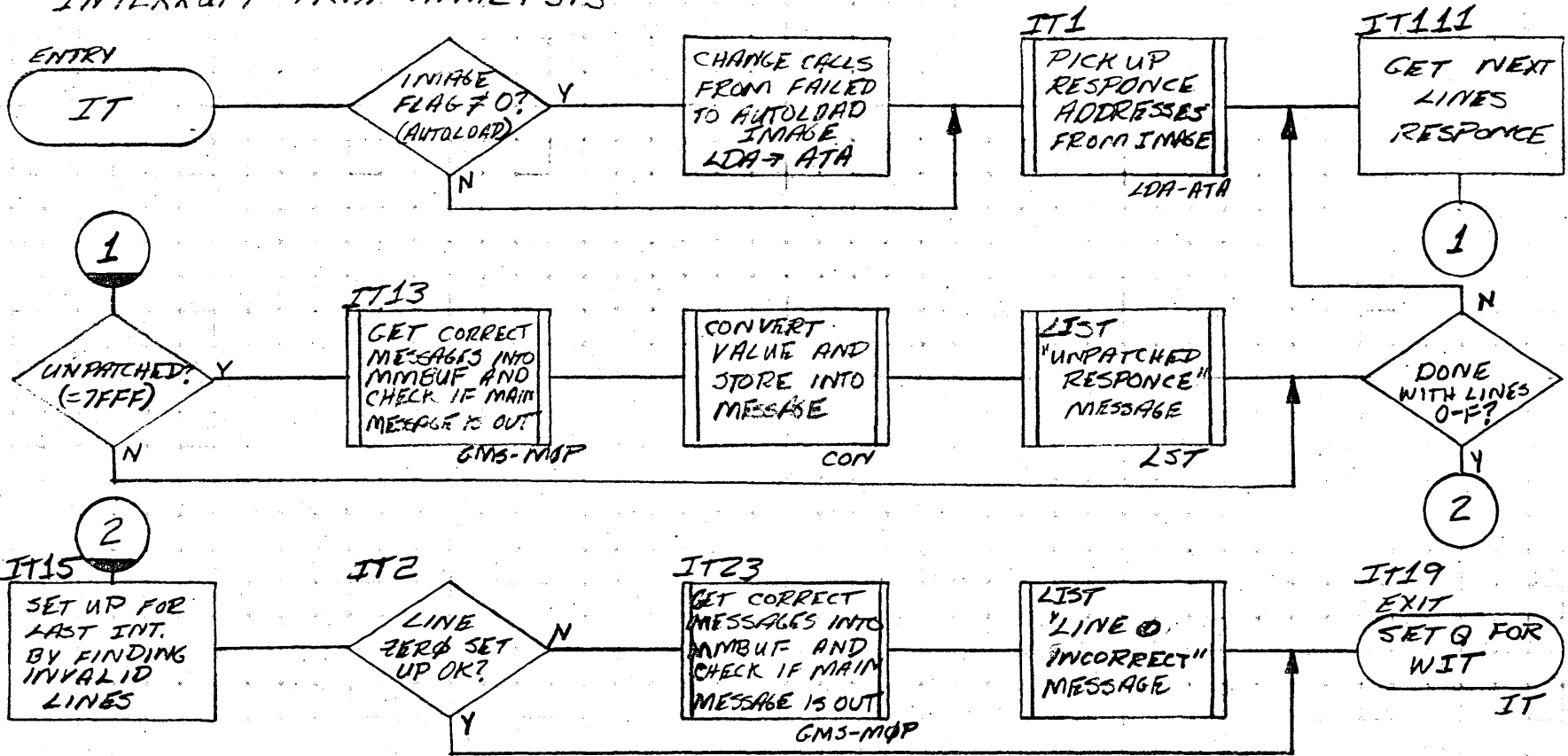
DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
NUMBER	ISSUE DATE	DATE	TASK NO.	PROJECT NAME			
DRAWN BY	DATE	DATE	TASK NAME				

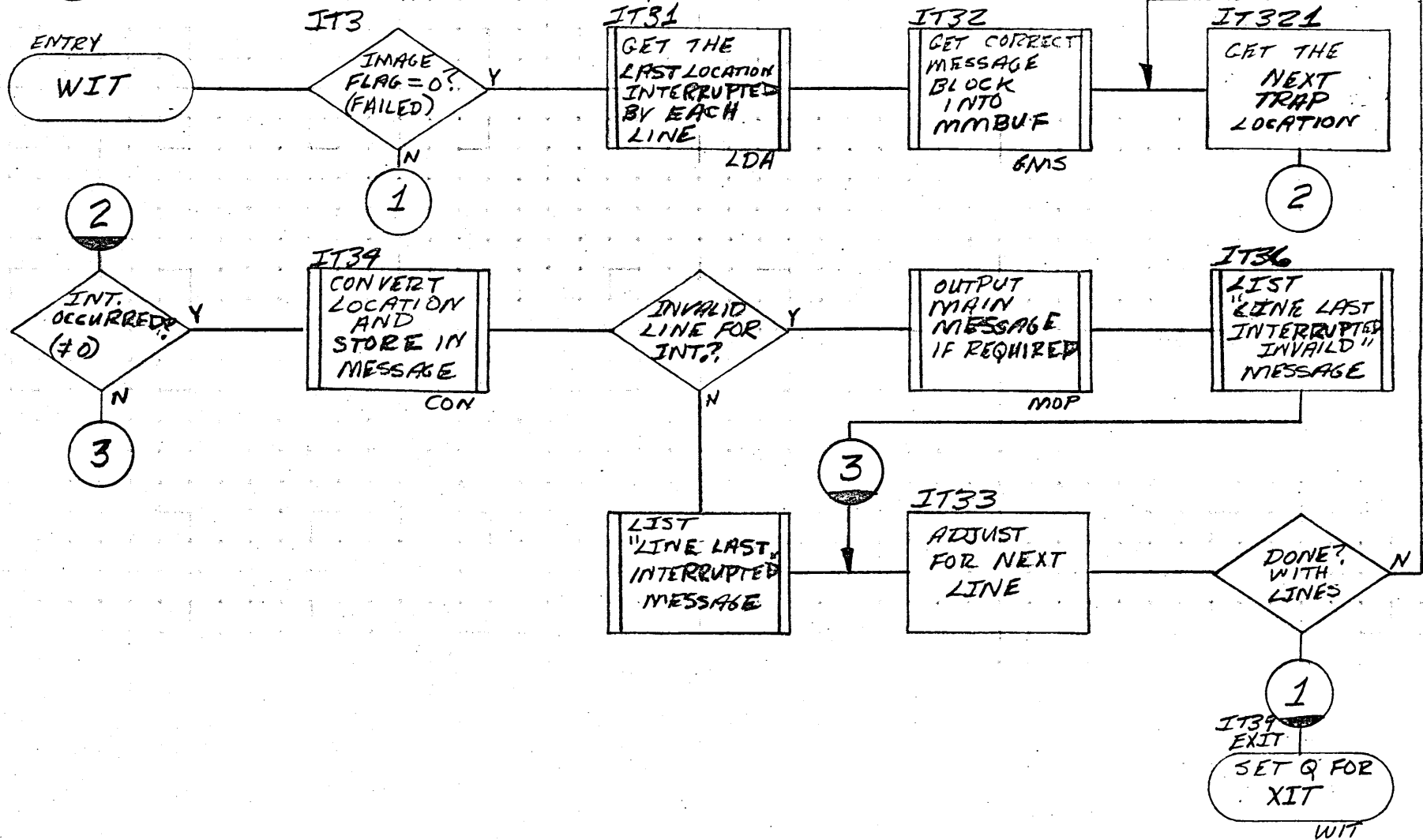
36-A

INTERRUPT TRAP ANALYSIS



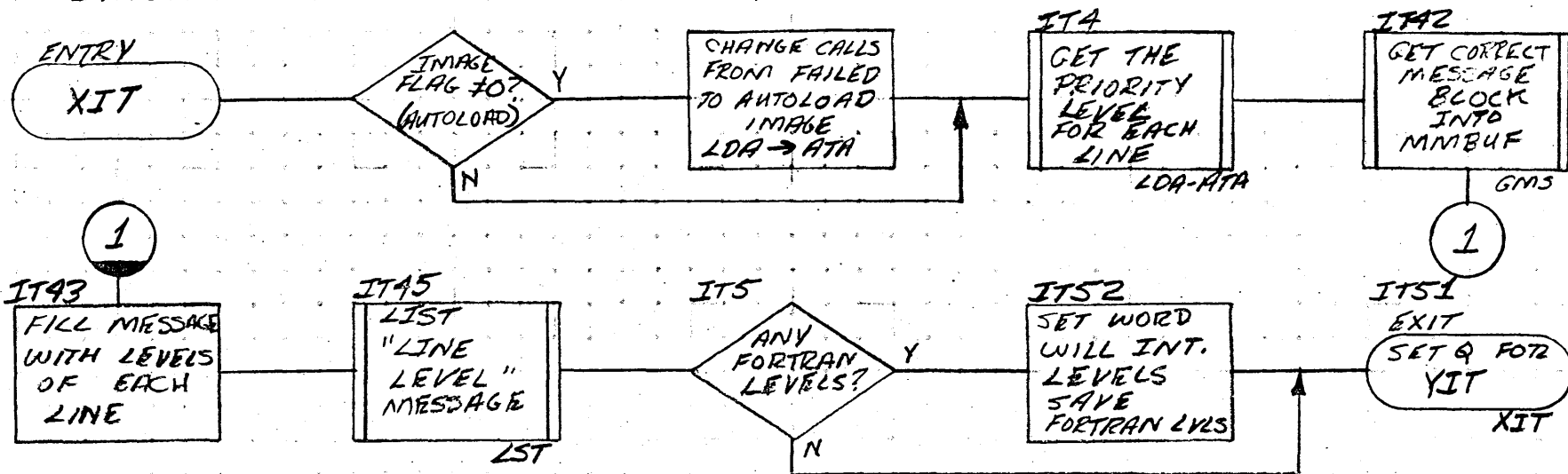
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
	PROGRAM	PAGE	OF	PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				

INTERRUPT TRAP ANALYSIS PART W



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				

INTERRUPT TRAP ANALYSIS PART X

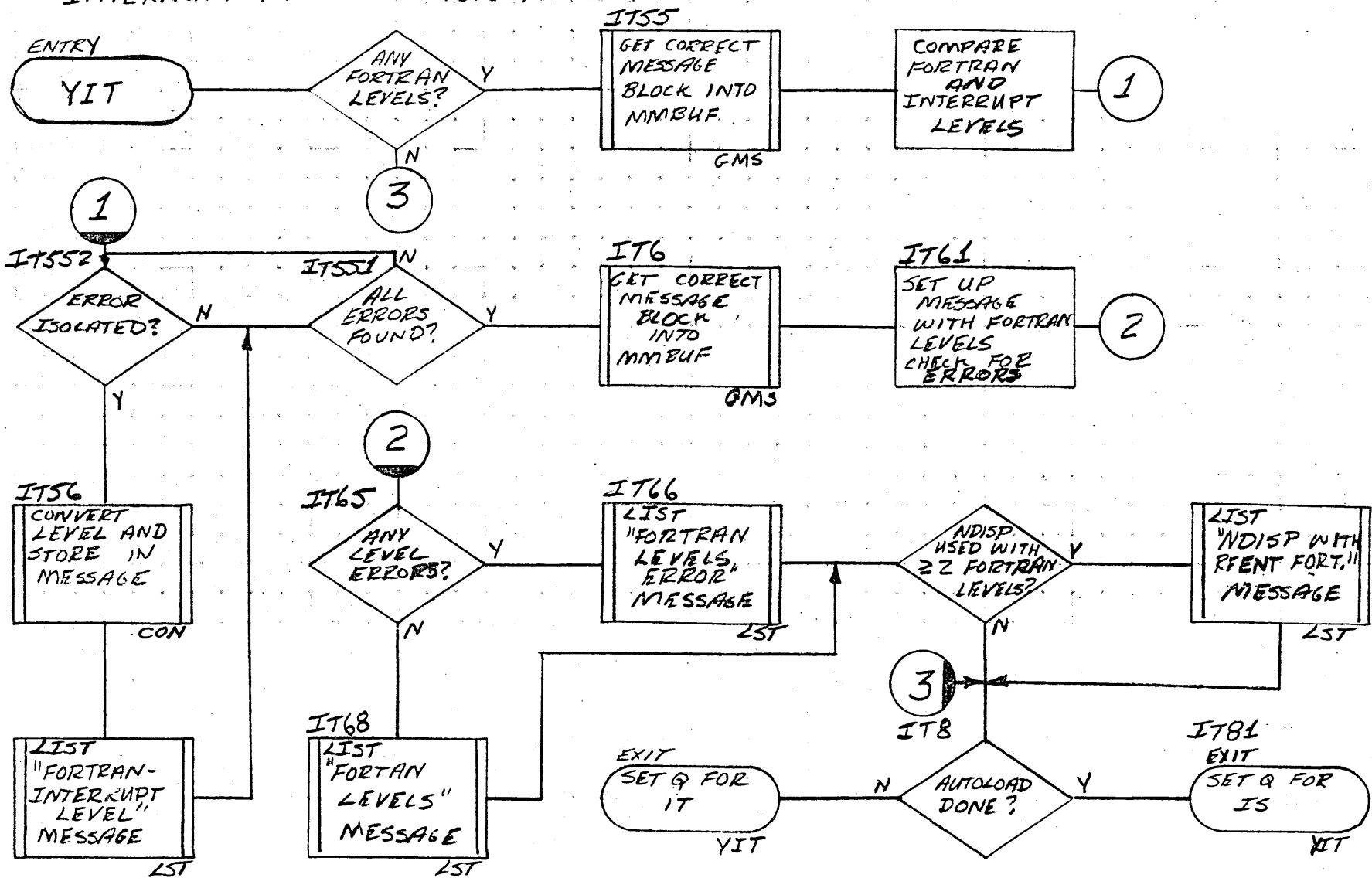


CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

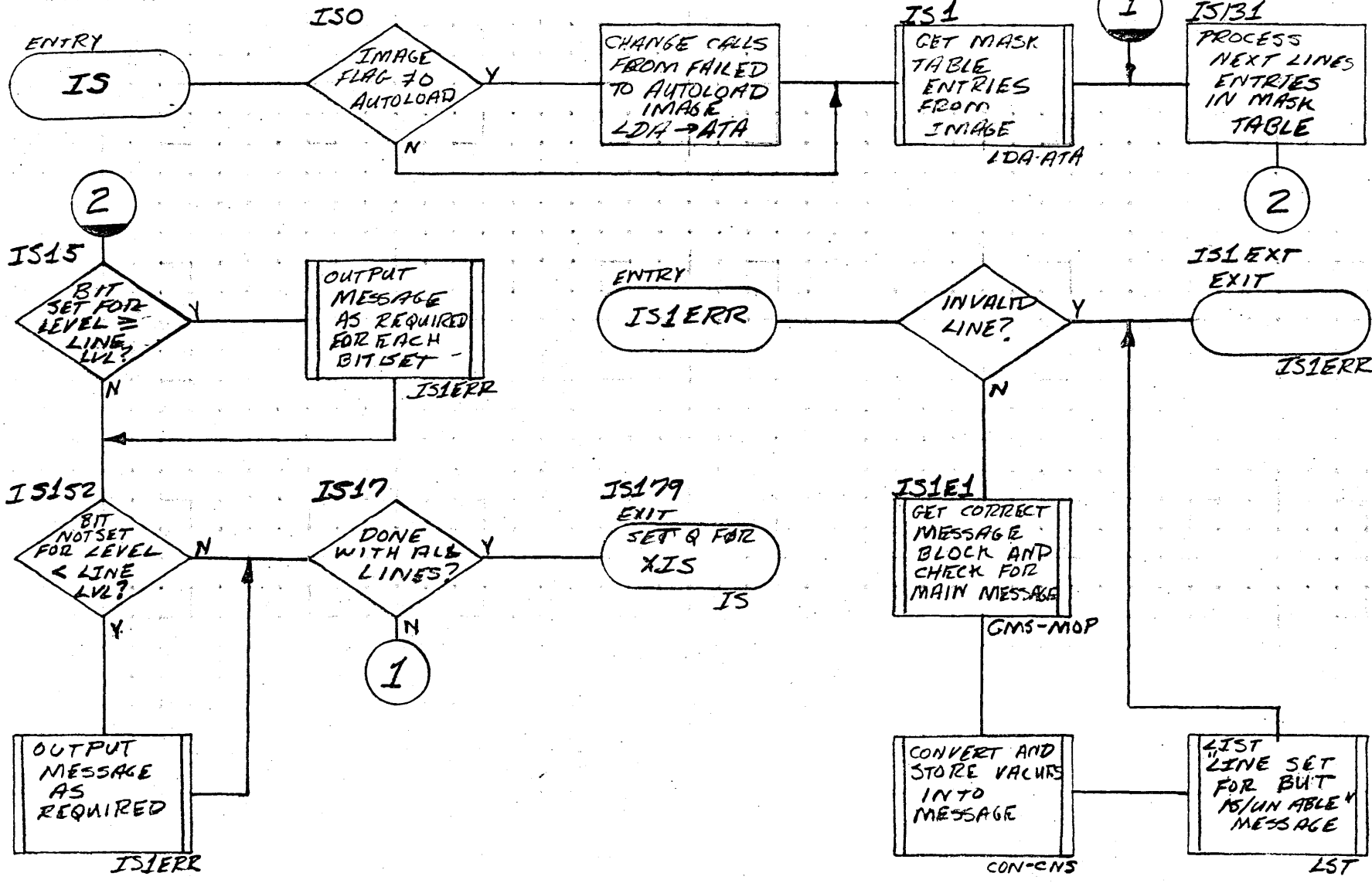
DOCUMENT CLASS	INIS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
NUMBER		PAGE OF		PROJECT NAME			
		ISSUE DATE		TASK NO.			
DRAWN BY		DATE		TASK NAME			

INTERRUPT TRAP ANALYSIS PART Y



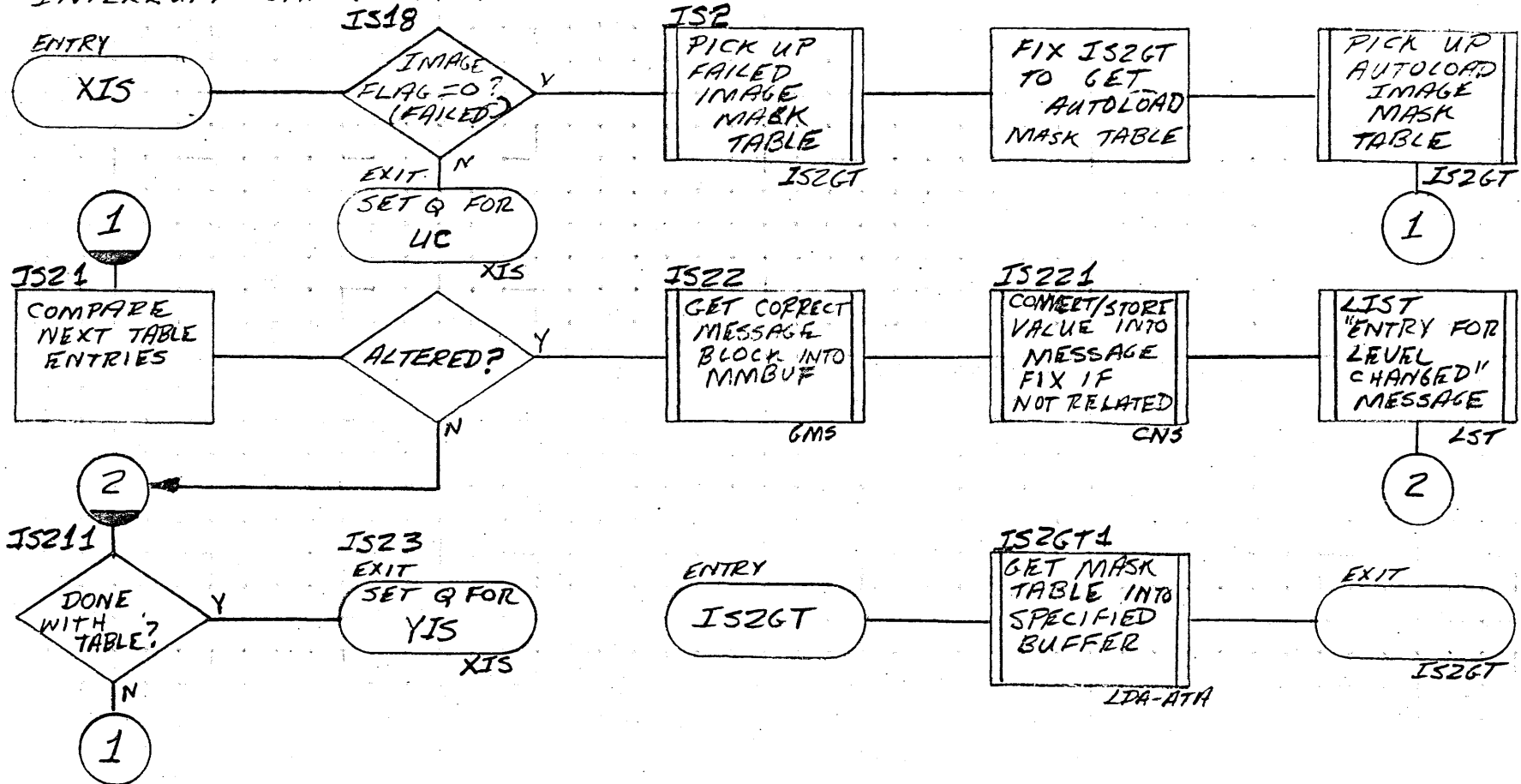
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER		ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			

INTERRUPT STACK AND MASK TABLE ANALYSIS



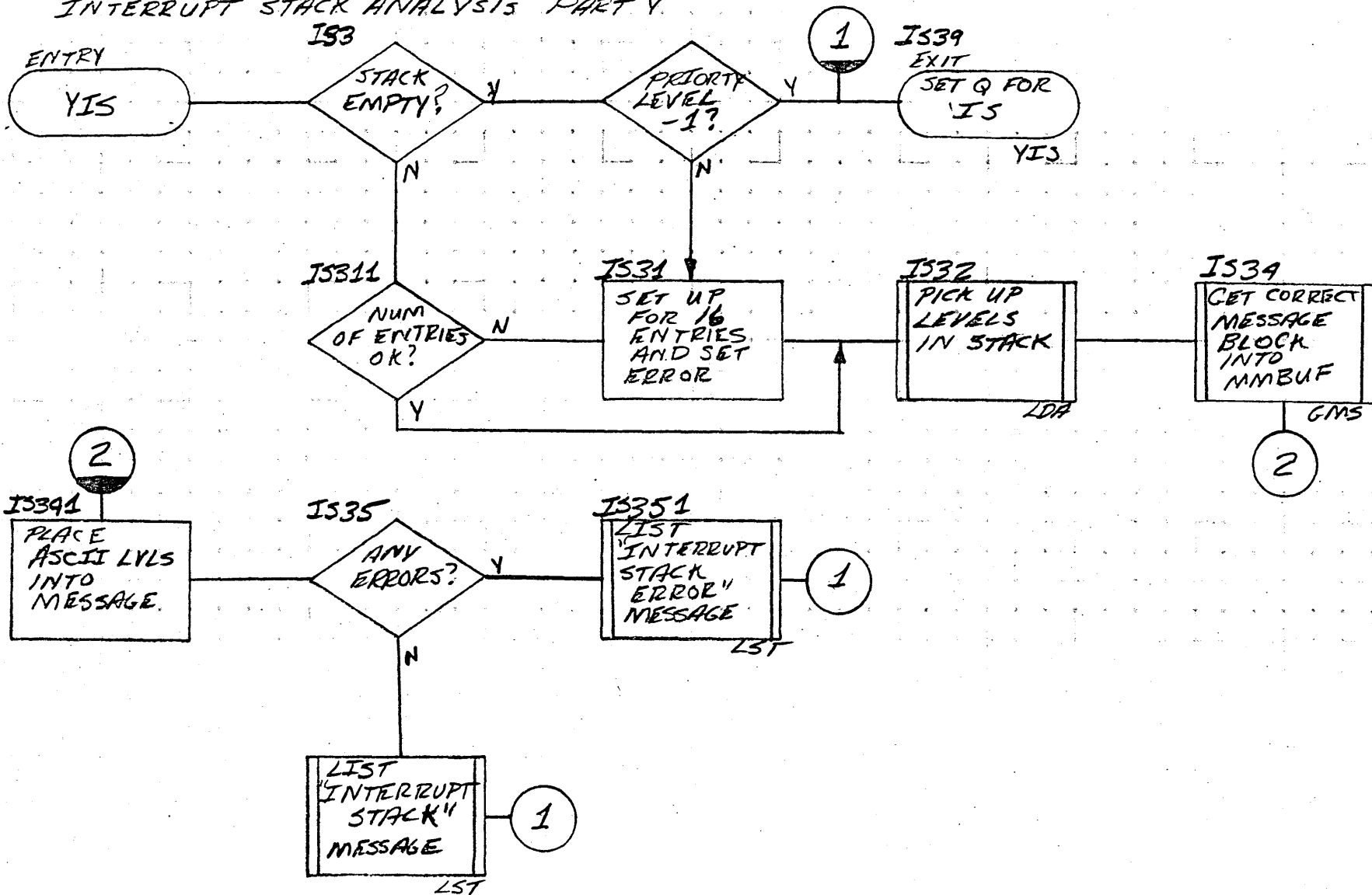
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
	PROGRAM	PAGE	OF	PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				

INTERRUPT STACK ANALYSIS PART X



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER		ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			

INTERRUPT STACK ANALYSIS PART V.



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

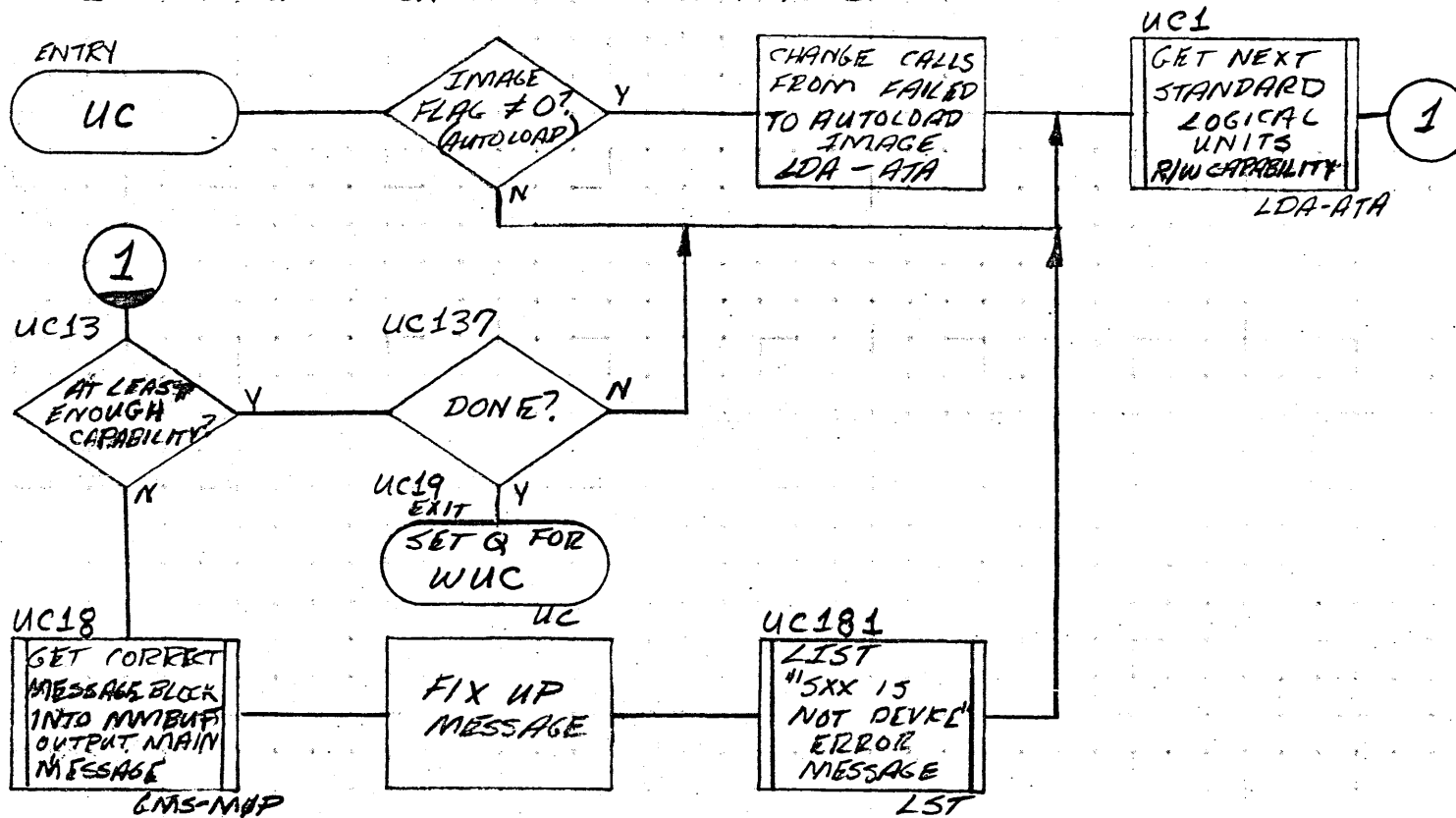
FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT			PROJECT MGR.			
PROGRAM		PAGE OF		PROJECT NAME			
NUMBER	ISSUE DATE			TASK NO.			
DRAWN BY	DATE			TASK NAME			

LOGICAL UNIT CAPABILITY ANALYSIS

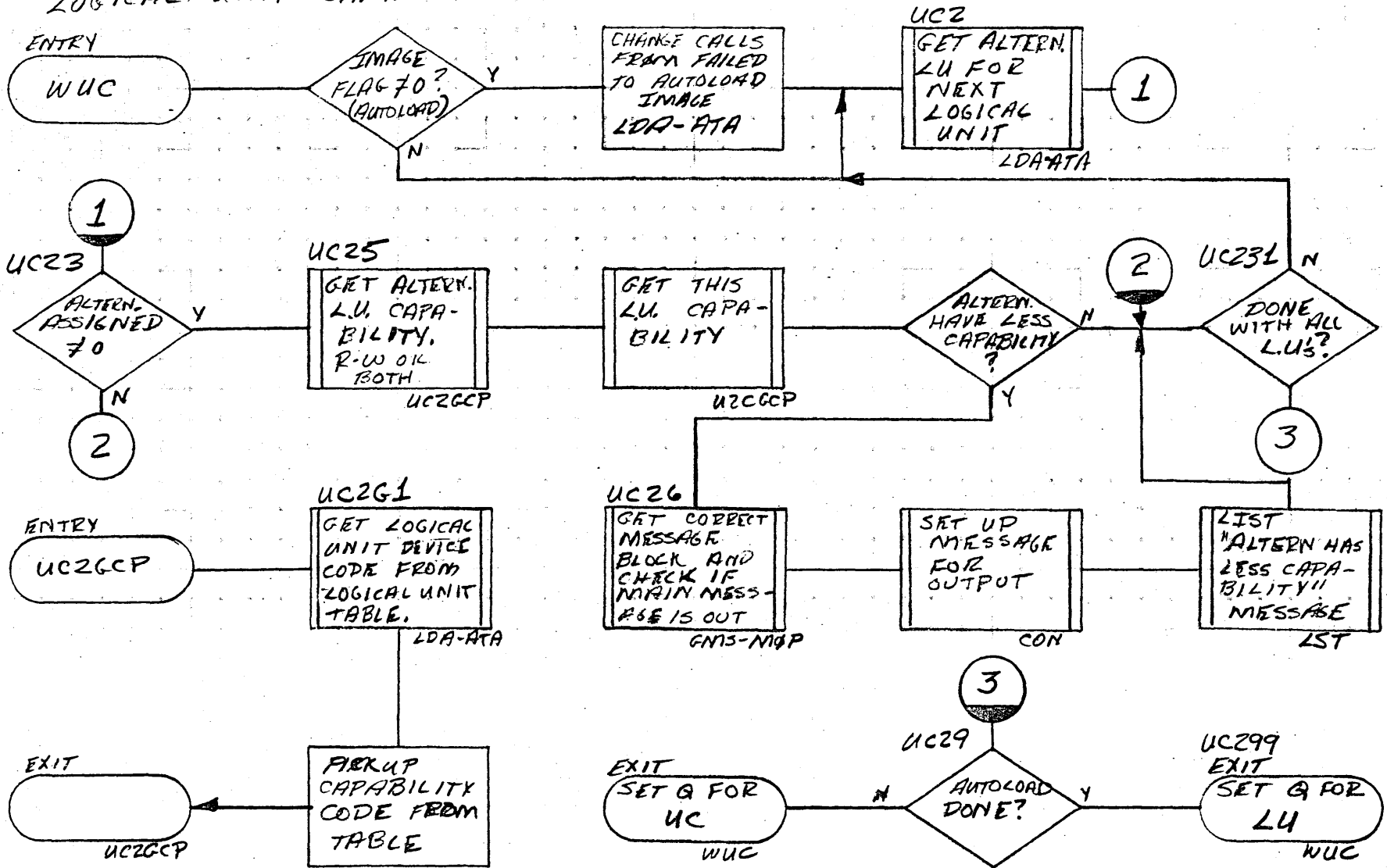


CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

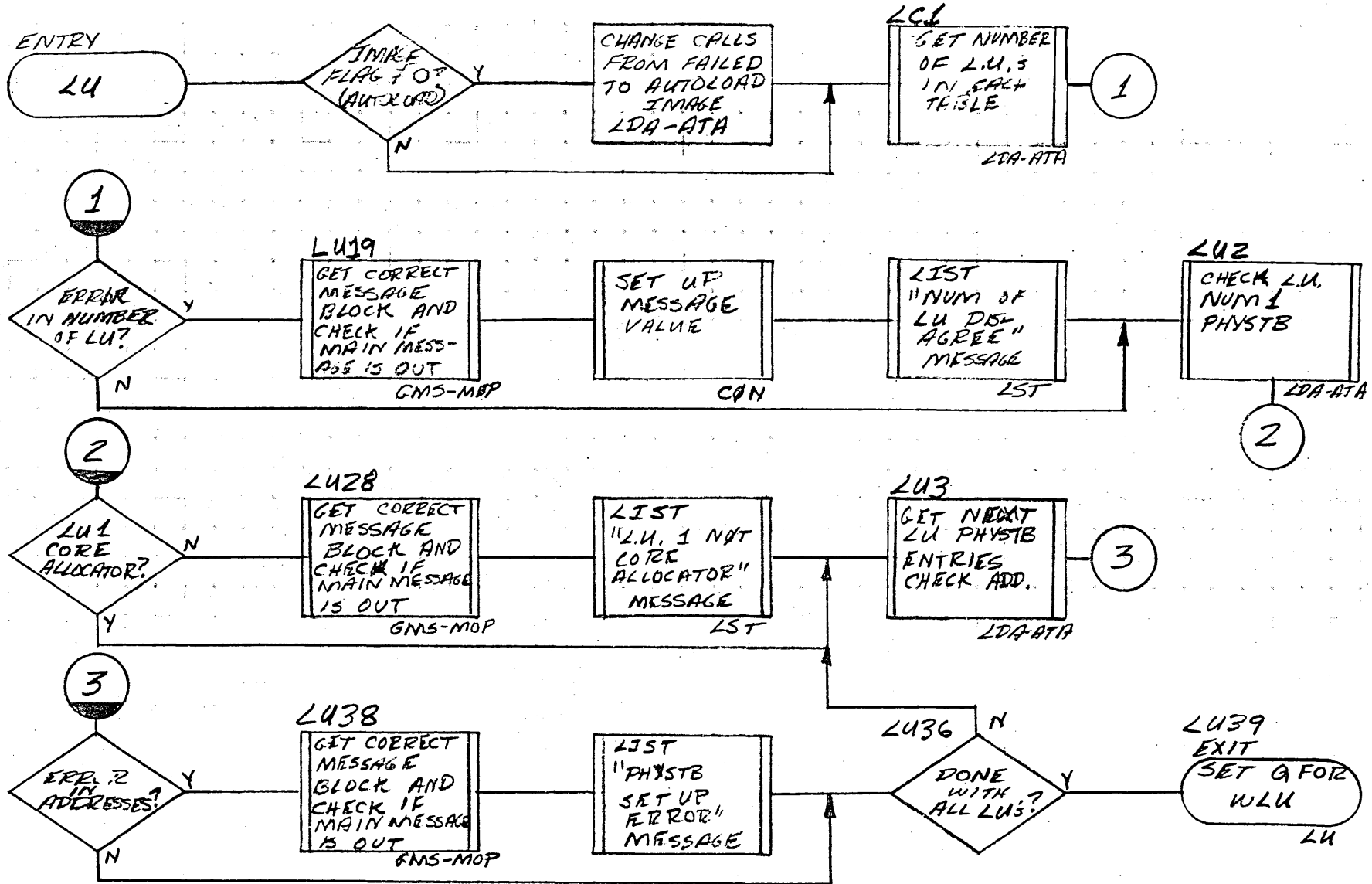
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
NUMBER	ISSUE DATE			PROJECT NAME			
DRAWN BY	DATE			TASK NO.			
				TASK NAME			

LOGICAL UNIT CAPABILITY ANALYSIS PART W



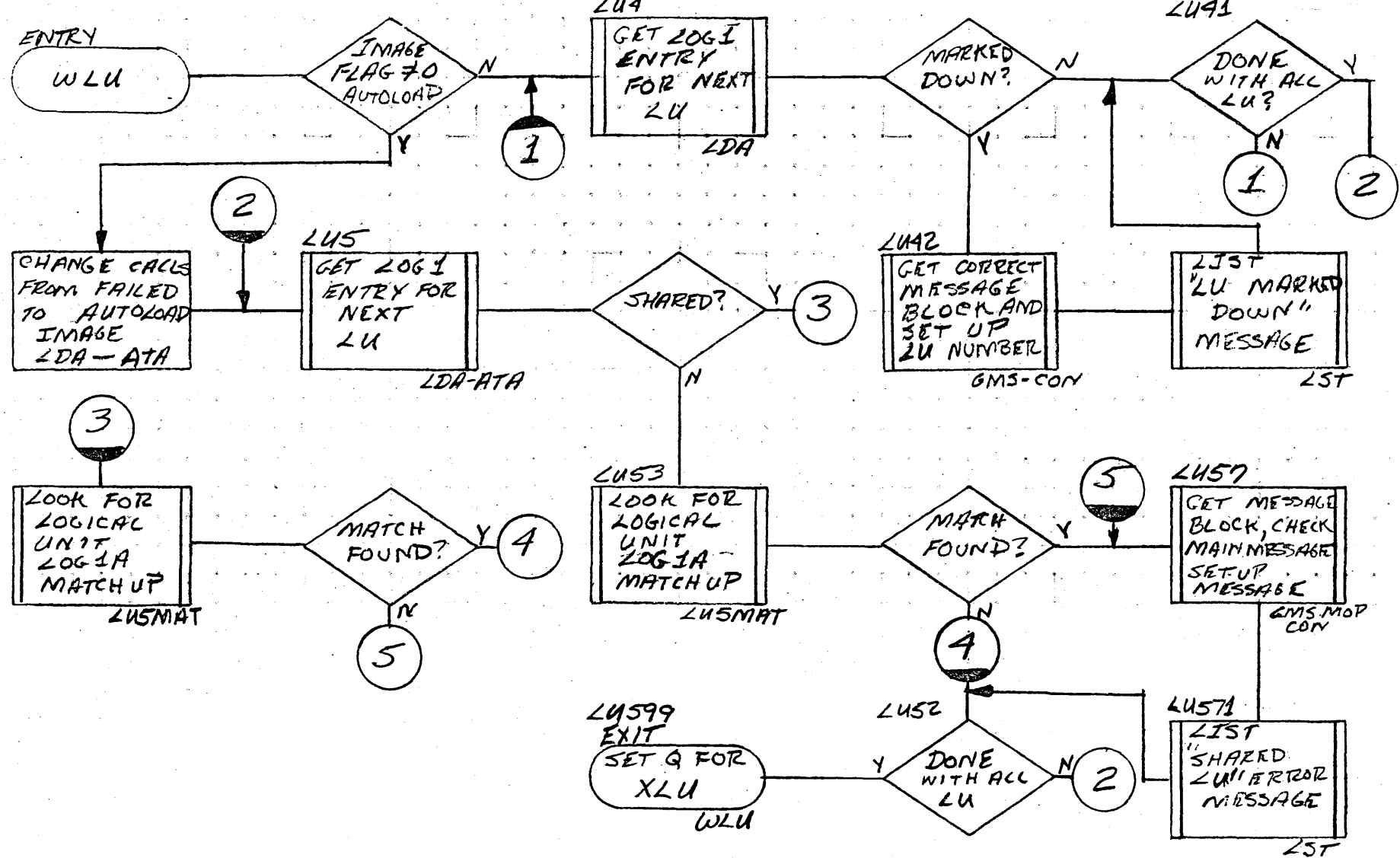
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS <i>IMS</i>	MACH. TYPE <i>1700</i>	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE <i>SYSTEM CHECKOUT PROGRAM</i>	PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE	PROJECT NAME			
	DRAWN BY	DATE	TASK NO.			
			TASK NAME			

LOGICAC UNIT ANALYSIS



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE			PROJECT NAME			
	DRAWN BY	DATE			TASK NO.			
					TASK NAME			

LOGICAL UNIT ANALYSIS PART W



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
	NUMBER		ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			

LOGICAL UNIT ANALYSIS PART X

ENTRY

XLU

IMAGE
FLAG 70
(AUTOLOAD)

EXIT
SET Q FOR
NEXT MODULE
XLU

LUG
CHECK WORD
5 OF NEXT
LUS PHYS-
TAB, BUSY
LU

1

LDA

1

DEVICE
ASSIGNED TO
THIS LU?

LUG5
PICK UP
THE CURRENT
PARAMETER
LIST.
ADD 15 WORD 6

LDA

LUG52
GET CORRECT
MESSAGE
BLOCK AND
SET UP
MESSAGE
VALUES

GMS-CON
CNS

L1ST
"HEADER AND
PARAMETER
L1ST"
MESSAGE

L1ST

2

LUG1
DONE
WITH ALL
LUS?

2

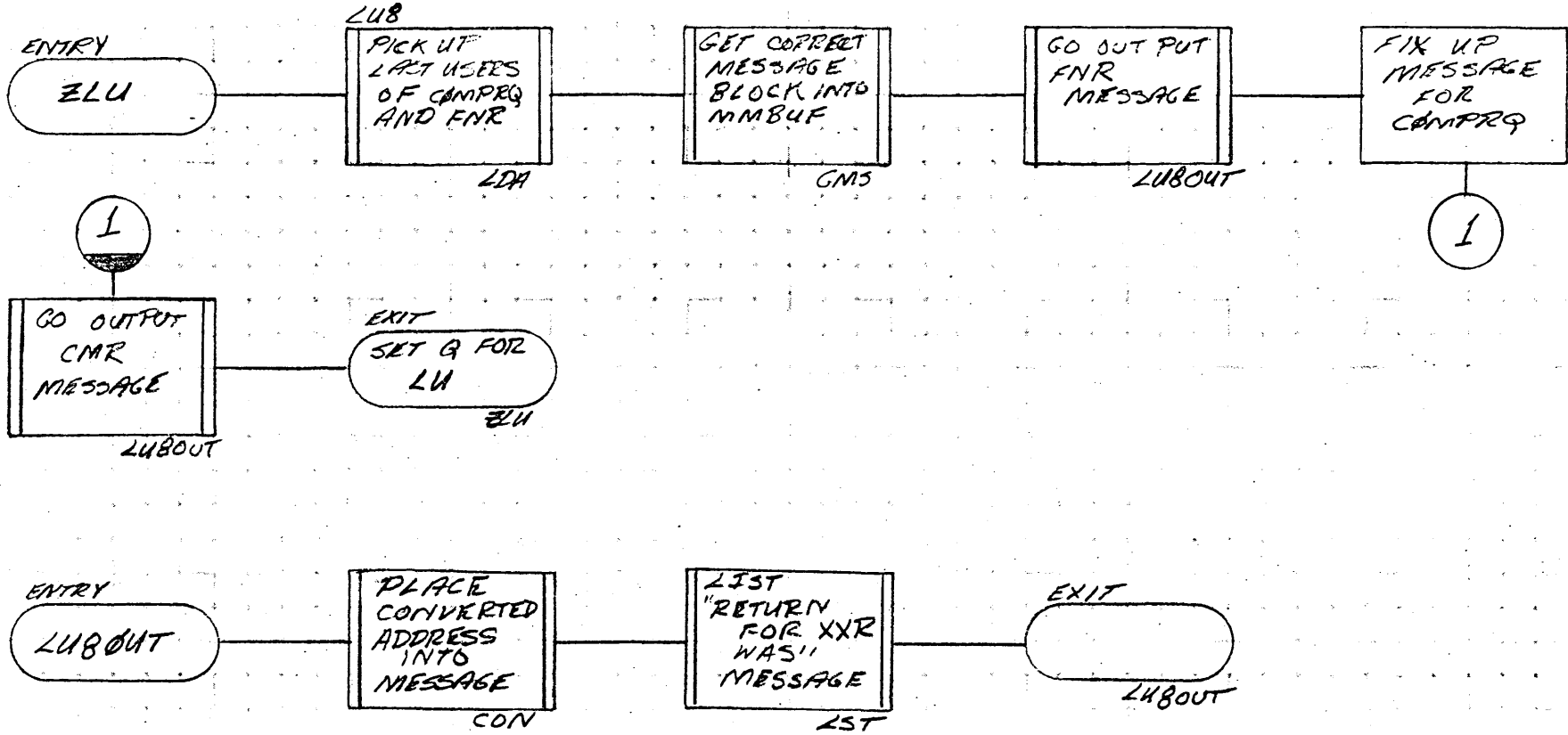
EXIT
SET Q FOR
VLU
XLU

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

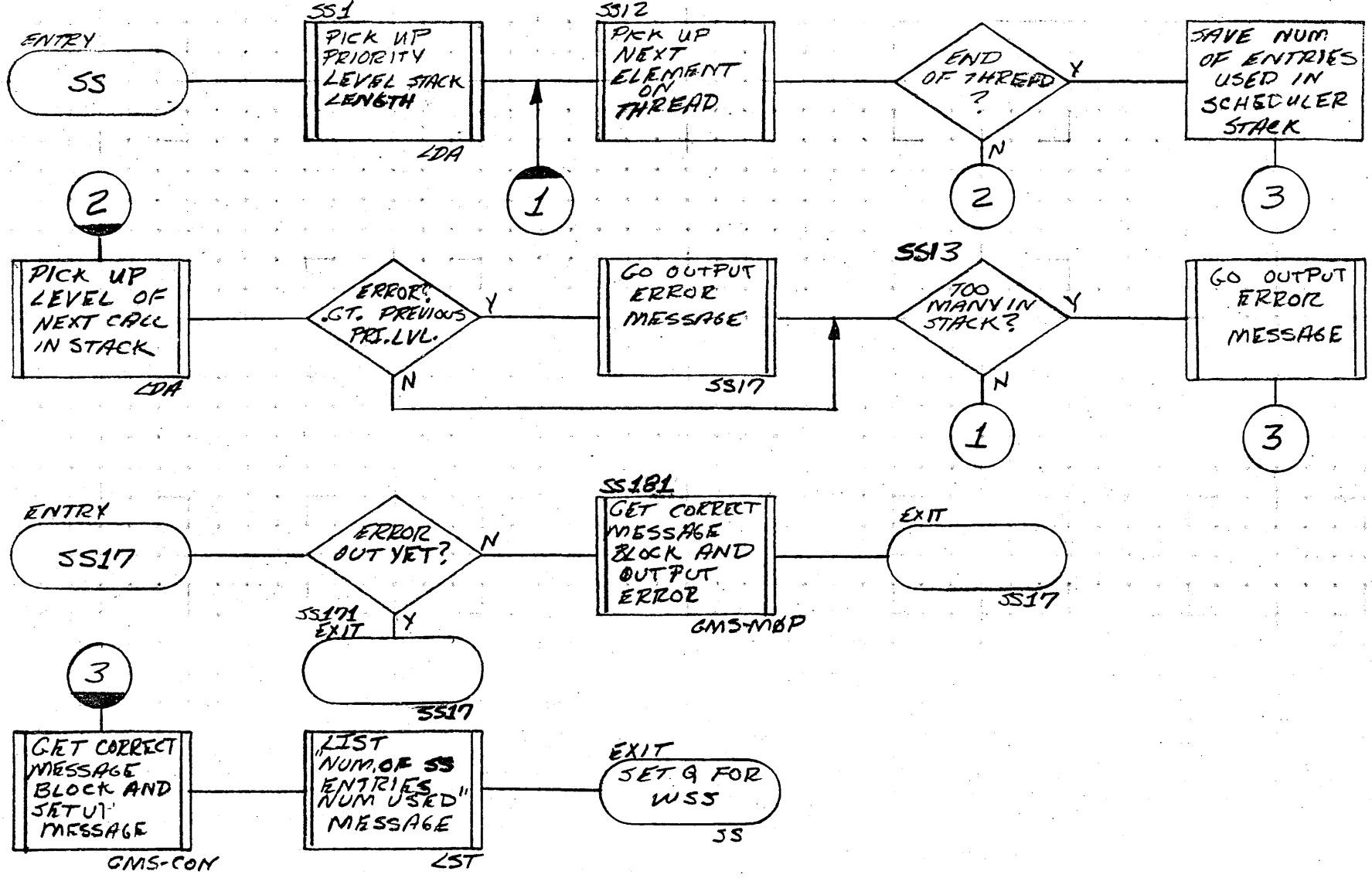
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT			PROJECT MGR.			
PROGRAM	PAGE	OF		PROJECT NAME			
NUMBER	ISSUE DATE			TASK NO.			
DRAWN BY	DATE			TASK NAME			

LOGICAL UNIT ANALYSIS PART 2



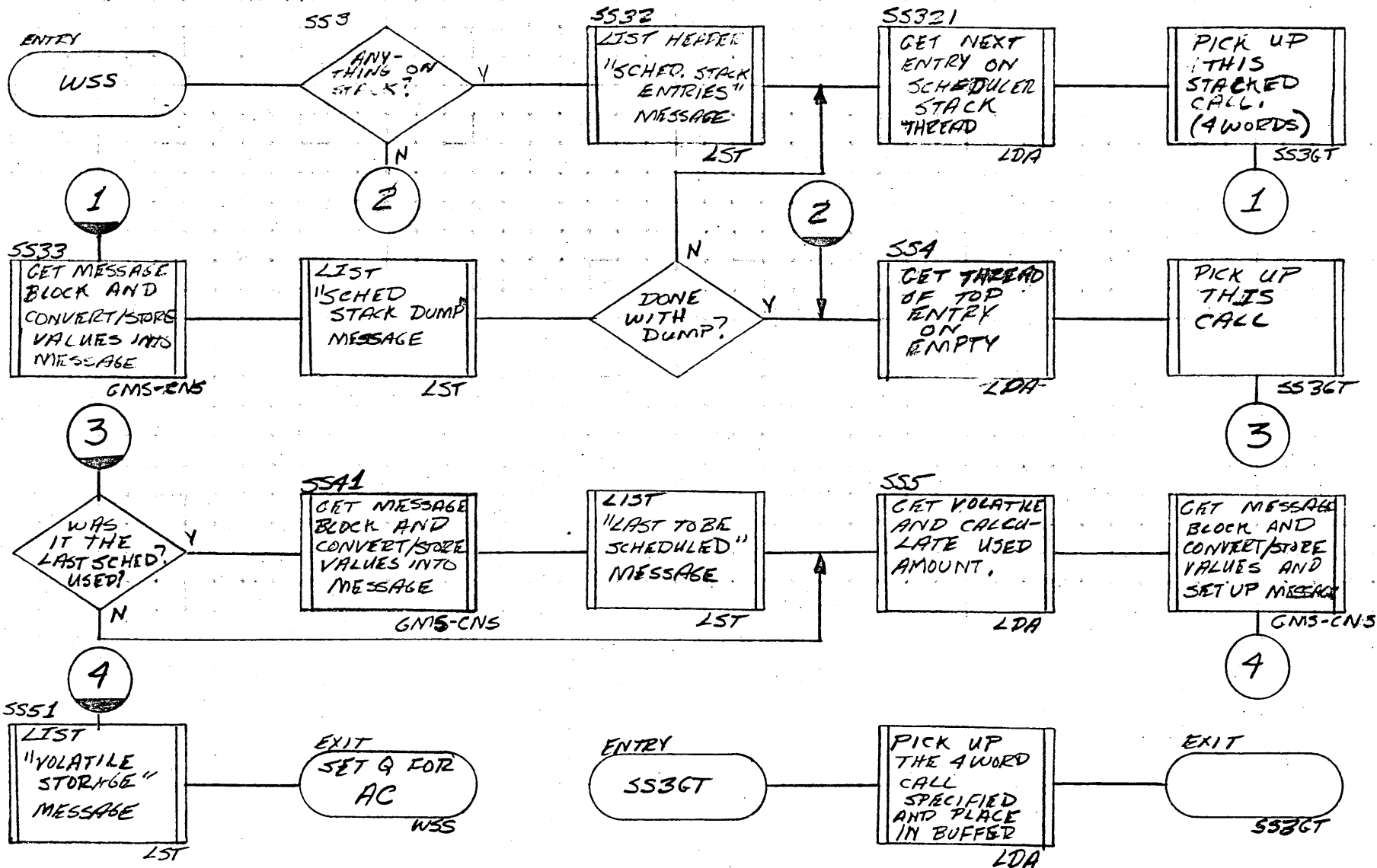
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS IMS	MACH. TYPE 1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE SYSTEM CHECKOUT PROGRAM	PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE	TASK NO.			
	DRAWN BY	DATE	TASK NAME			

SCHEDULER STACK ANALYSIS



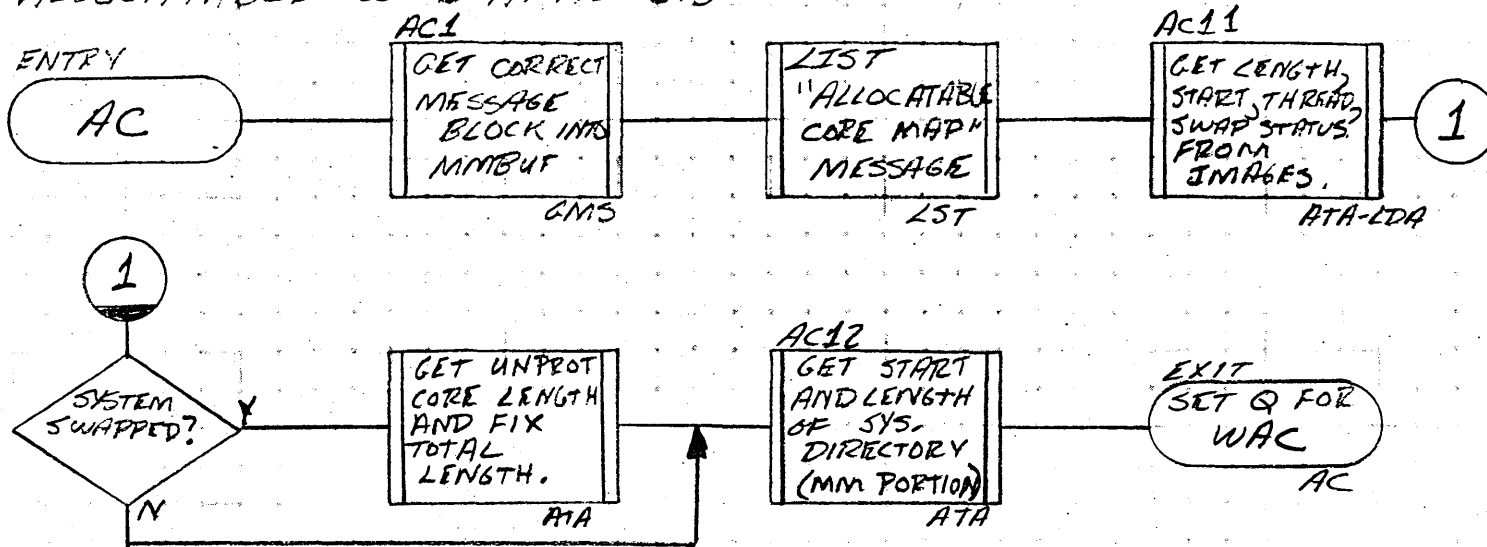
CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER		ISSUE DATE		PROJECT NAME			
	DRAWN BY		DATE		TASK NO.			
					TASK NAME			

SCHEDULER STACKS ANALYSIS PART W



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT		PROJECT MGR.				
	PROGRAM	PAGE OF		PROJECT NAME				
	NUMBER	ISSUE DATE		TASK NO.				
	DRAWN BY	DATE		TASK NAME				
	AA1385 (FORMERLY CA127-1)							

ALLOCATABLE CORE ANALYSIS



CONTROL DATA CORPORATION

SOFTWARE DOCUMENT

SAMPLE CODE

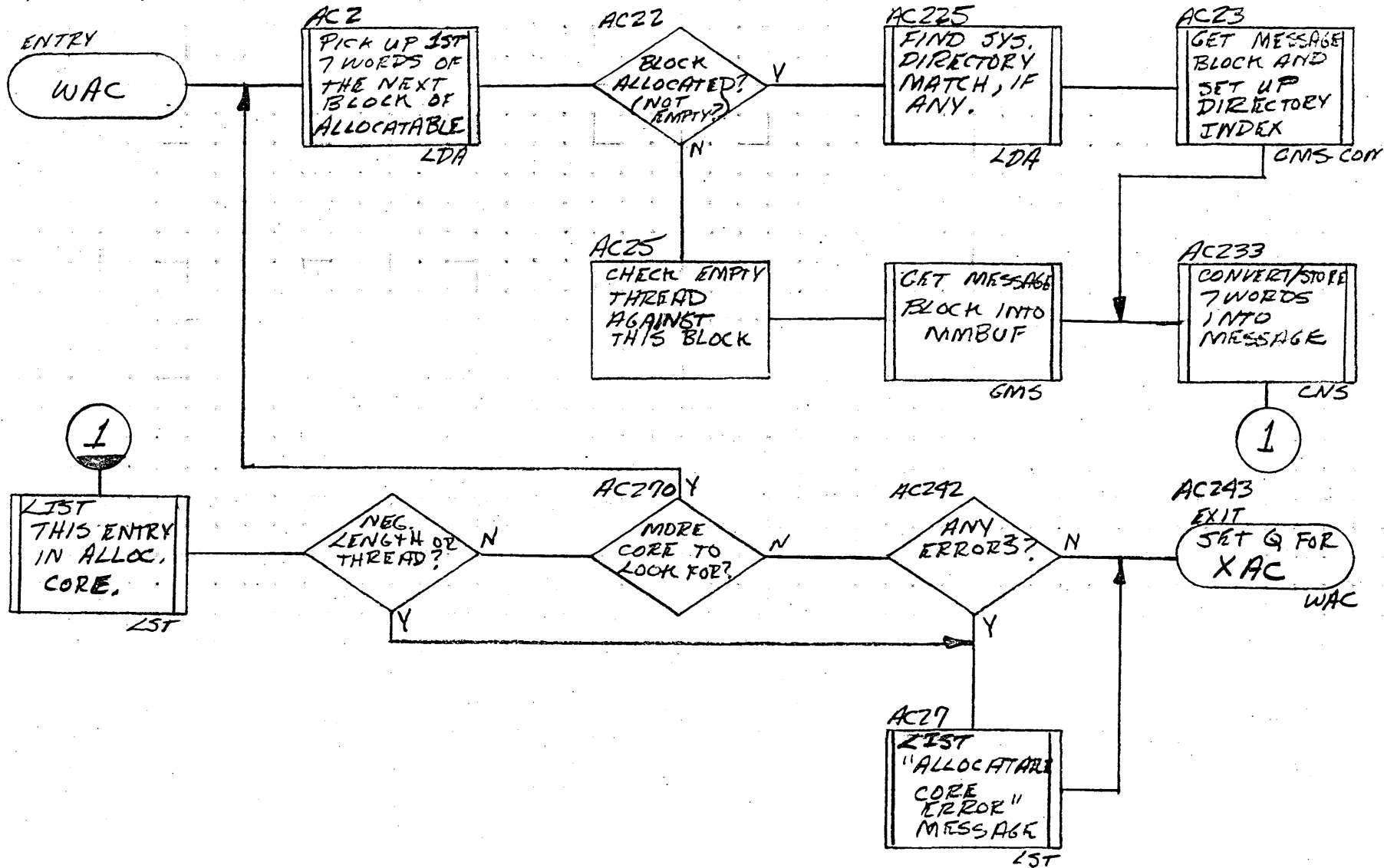
FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	<i>IMS</i>	MACH. TYPE	<i>1700</i>	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	<i>SYSTEM CHECKOUT PROGRAM</i>			PROJECT MGR.			
NUMBER		ISSUE DATE		TASK NO.			
DRAWN BY		DATE		TASK NAME			

ALLOCATABLE CORE ANALYSIS PART W



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

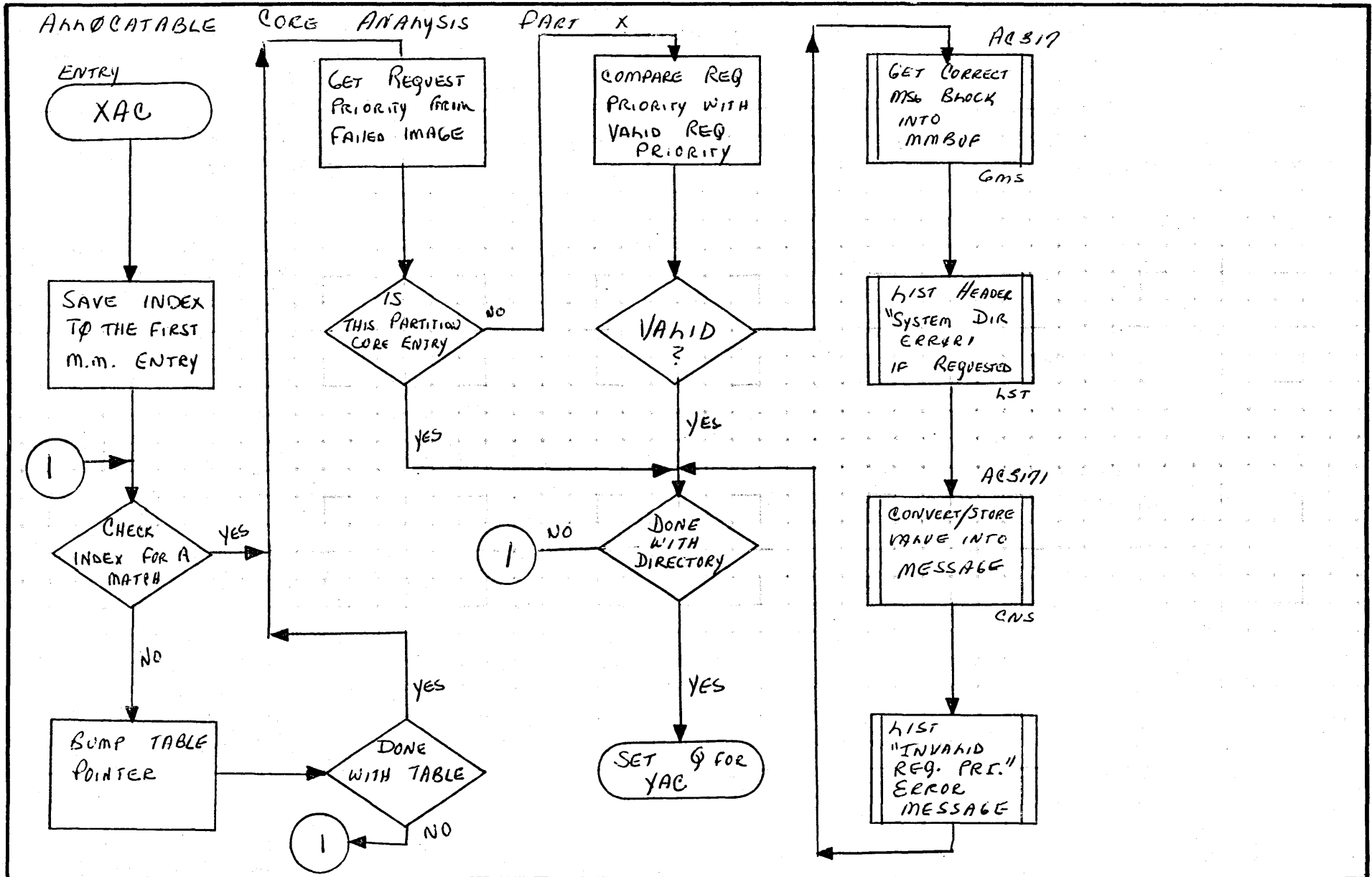
SAMPLE CODE

FLOWCHART

DECISION TABLE

OTHER

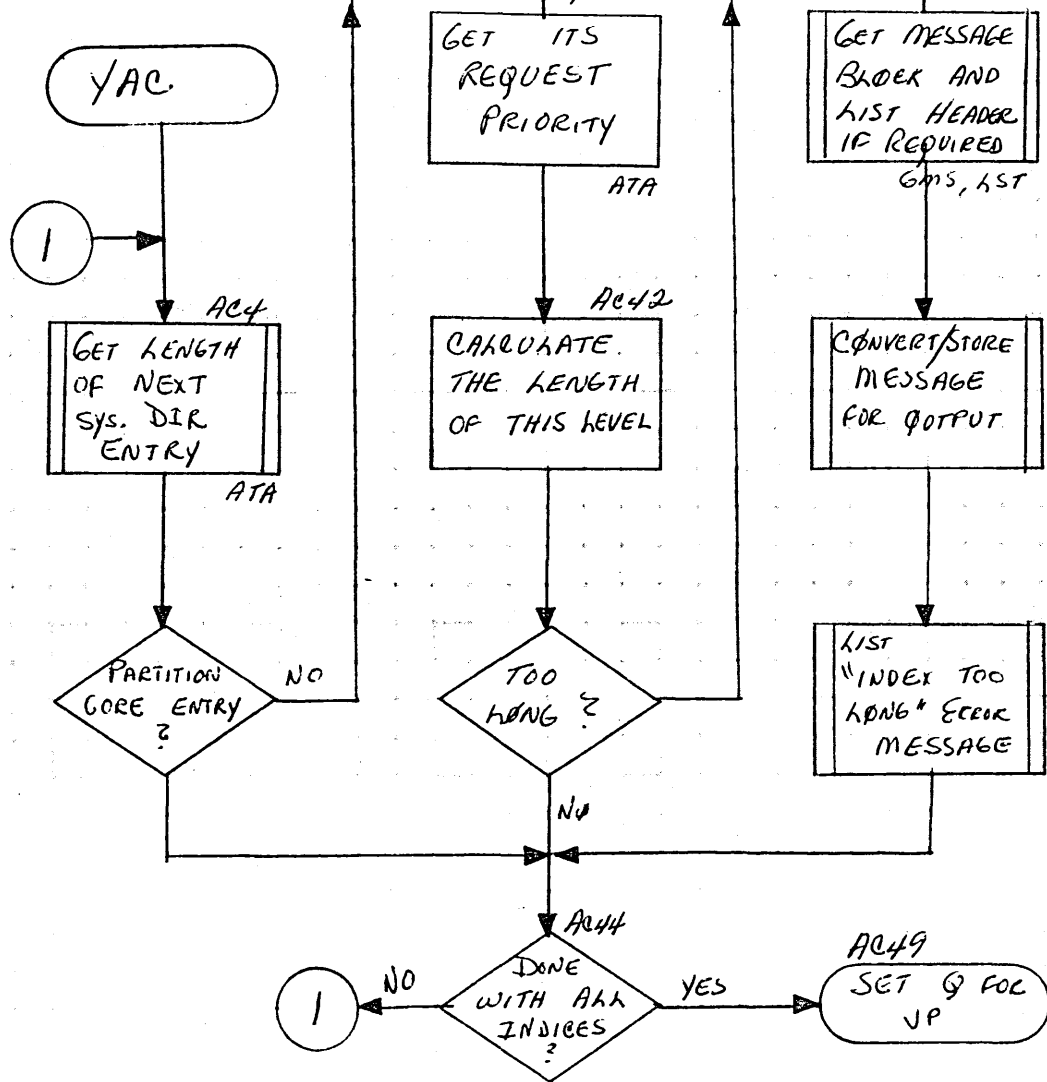
DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
NUMBER		ISSUE DATE		PROJECT NAME			
DRAWN BY		DATE		TASK NO.			
				TASK NAME			



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	MACH. TYPE	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE		PROJECT MGR.			
		PAGE OF	PROJECT NAME			
	NUMBER	ISSUE DATE	TASK NO.			
	DRAWN BY	DATE	TASK NAME			

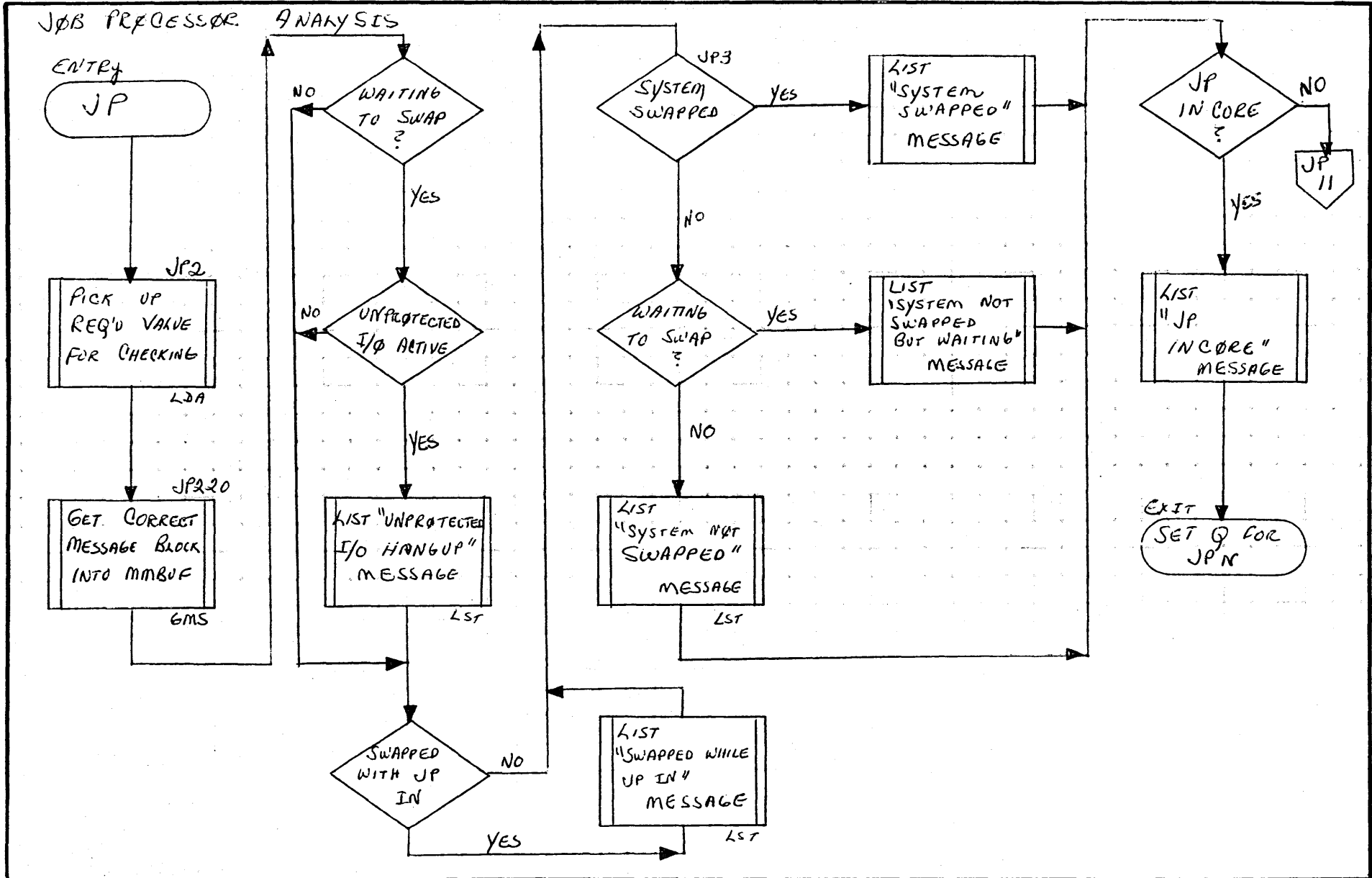
55

ALLOCATABLE CORE ANALYSIS PART 1



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS. <i>IMS</i>	MACH. TYPE <i>1700</i>	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE <i>PROGRAM</i>	PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE	PROJECT NAME			
	DRAWN BY	DATE	TASK NO.			
			TASK NAME			

56



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS IMS	MACH. TYPE 1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE SYSTEM CHECKOUT PROGRAM	PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE	PROJECT NAME			
	DRAWN BY	DATE	TASK NO.			
			TASK NAME			

57



1

2



3

4

5

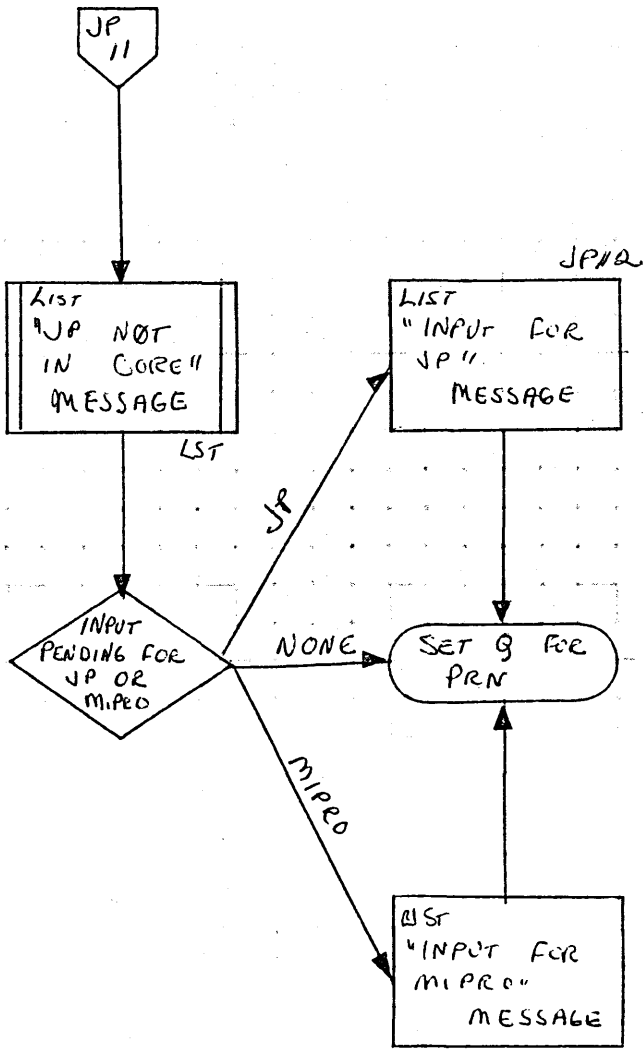


A

B

C

D



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

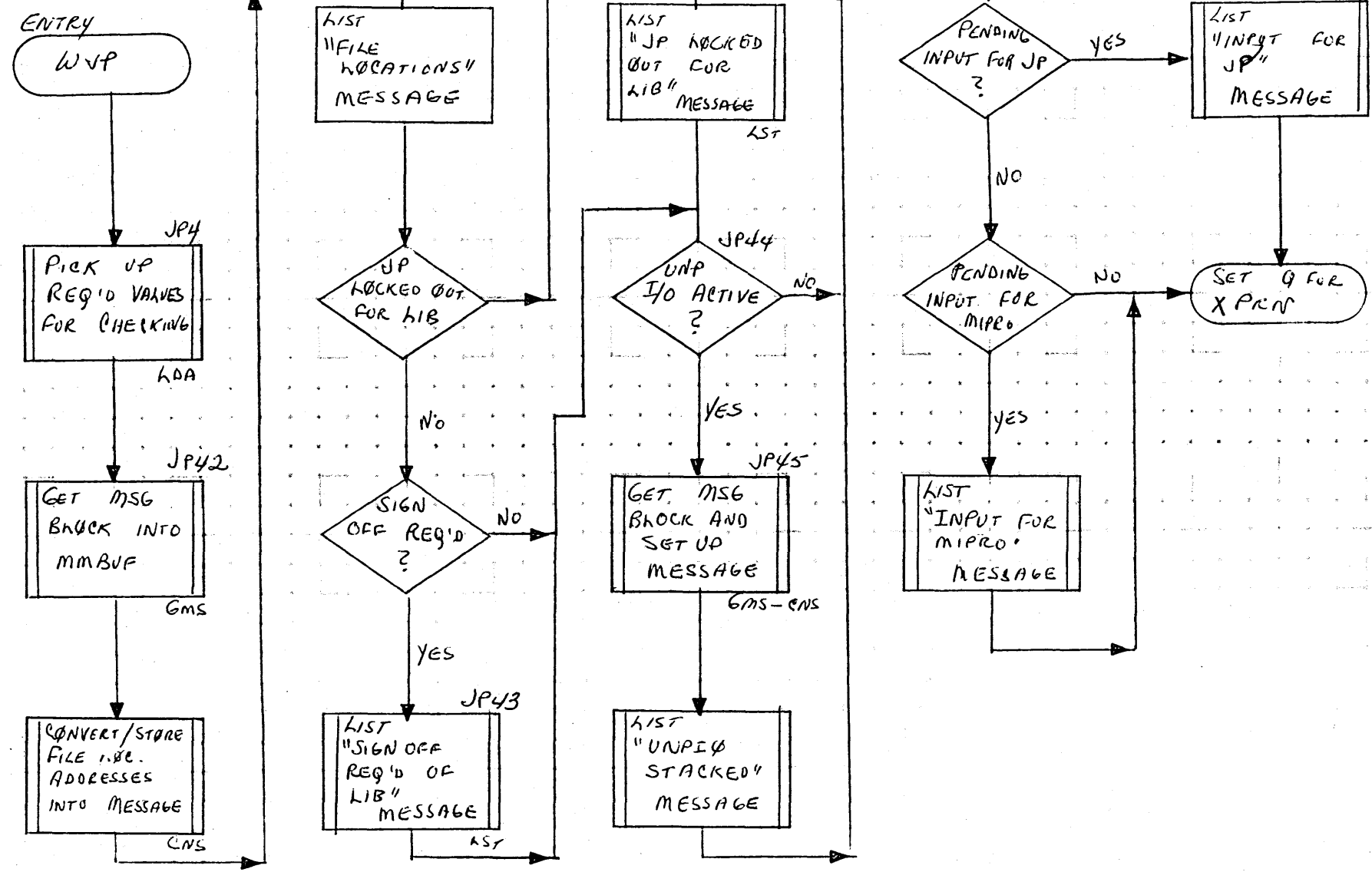
- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
NUMBER		ISSUE DATE		PROJECT NAME			
DRAWN BY		DATE		TASK NO.			
				TASK NAME			

57-A

JP ANALYSIS PART W

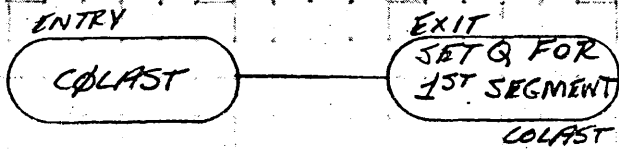
A
B
C
D



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input checked="" type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		PAGE OF	PROJECT MGR.			
	NUMBER	ISSUE DATE			PROJECT NAME			
	DRAWN BY	DATE			TASK NO.			
					TASK NAME			

85

COLAST



CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

- SAMPLE CODE
- FLOWCHART
- DECISION TABLE
- OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700	PROJECT NO.	REV	APPROVED	DATE
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM			PROJECT MGR.			
NUMBER		ISSUE DATE	PAGE OF	PROJECT NAME			
DRAWN BY		DATE		TASK NO.			
				TASK NAME			

PARTITION THREADS AND ADDRESSES

A

B

C

D

ENTRY
XPR

ANY PARTITIONS

EXIT
SET Q FOR NEXT

GET MSG BLOCK TO MMBUF

1

GET PARTITION ADDRESS AND NUMBER

RESTORE MESSAGE BLOCK

CONVERT/STORE VALUES IN OUTPUT BUFFER

LIST PARTITION # AND ADDRESS

DONE

1

LIST HEADER FOR THREADS

CALCULATE ADDR OF USE TABLE AND THDS TABLE

ANYTHING THREADED TO THIS PARTITION

LIST THREAD CONTENTS

DONE

SET Q FOR YPR

NO

YES

NO

YES

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE
FLOWCHART
DECISION TABLE
OTHER

DOCUMENT CLASS IMS MACH. TYPE 70
DOCUMENT TITLE SYSTEM CHECKOUT
PROGRAM PAGE OF
NUMBER ISSUE DATE
DRAWN BY DATE

PROJECT NO.
PROJECT MGR.
PROJECT NAME
TASK NO.
TASK NAME

REV APPROVED DATE

09

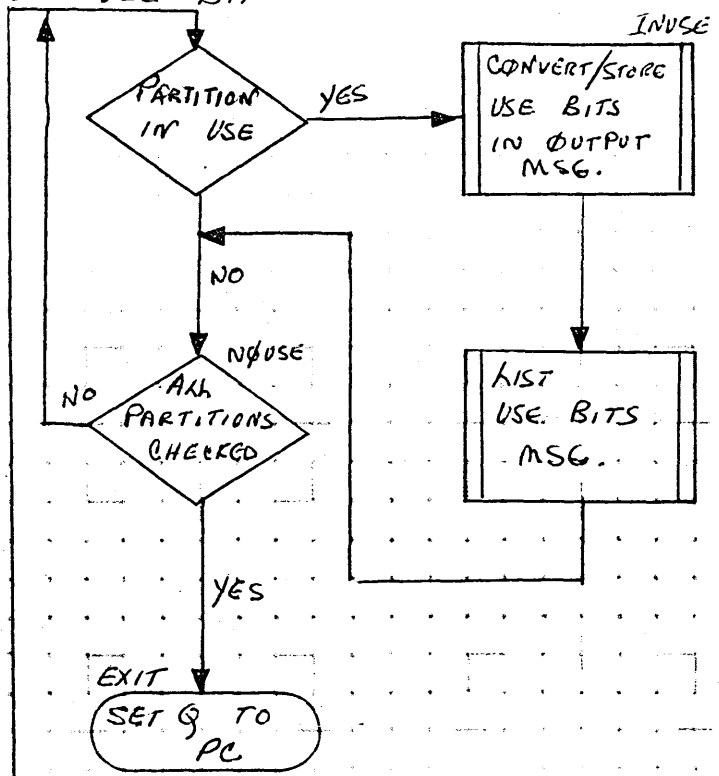
PRINT PARTITION USE BITS

A
B
C
D

YPR

GET MSG
BLOCK
TO MMBVC

HIST.
HEADER
MSG



CONTROL DATA CORPORATION SOFTWARE DOCUMENT SAMPLE CODE <input type="checkbox"/> FLOWCHART <input type="checkbox"/> DECISION TABLE <input type="checkbox"/> OTHER <input type="checkbox"/>	DOCUMENT CLASS	MACH. TYPE	PROJECT NO.	REV	APPROVED	DATE
	DOCUMENT TITLE		PROJECT MGR.			
		PAGE OF	PROJECT NAME			
	NUMBER	ISSUE DATE	TASK NO.			
	DRAWN BY	DATE	TASK NAME			

19

PARTITION CORE ANALYSIS

ENTRY

PC

START
PARTITION
0 > 8000

LIST ERROR
"ILLEGAL
PARTITION
0" MSG

LST

GET NEXT
PC
ADDRESS

ADDRESS
> THAN
PREVIOUS

LIST
"PARTITION
OUT OF
ORDER MSG"

LST

CALCULATE
OF USED
PARTITIONS

BUSY
WORD
OK

LIST
"ILLEGAL
BUSY
INDICATOR"

SET Q TO
NEXT

PC.12

NO

Yes

NO

Yes

A

B

C

D

CONTROL DATA CORPORATION
SOFTWARE DOCUMENT

SAMPLE CODE

FLOWCHART

DECISION TABLE

OTHER

DOCUMENT CLASS	IMS	MACH. TYPE	1700
DOCUMENT TITLE	SYSTEM CHECKOUT PROGRAM		
NUMBER	ISSUE DATE	PAGE OF	
DRAWN BY	DATE	TASK NO.	TASK NAME

PROJECT NO.	REV	APPROVED	DATE
PROJECT MGR.			
PROJECT NAME			
TASK NO.			
TASK NAME			

62