Price: $6.50

SDS PROJECT MANAGEMENT SYSTEM
TECHNICAL MANUAL

900 SERIES/9300 COMPUTERS

SDS 90 08 22A November 1965

SCIENTIFIC DATA SYSTEMS/1649 Seventeenth Street/Santa Monica, California (213)871-0960

Printed in U.S.A.

PREFACE

The SDS Project Management System consists of four programs: Schedule Spectrum Program (SSP), Detail Schedule Report Program (DSRP), Progress Evaluation Program (PEP), and Progress Evaluation Sort Program (PEPSORT). This document provides a technical reference for those interested in details of the organization and construction of these programs. Continuous improvements are being made to the system, and certain information in this document may become obsolete. Therefore, this information should be verified by the cognizant SDS representatives before it is used as a basis for changes.

The catalog numbers for these programs are

	Cover Number	SSP	DSRP	PEP	PEPSORT
910/925 Card Version	145001	145002	145003	145004	145005
910/925 Tape Version	145008	-	-	-	-
920/930 Card Version	245001	245002	245003	245004	245005
920/930 Tape Version	245008	-	-	-	-
9300 Card Version	645001	645002	645003	645004	645005
9300 Tape Version	645008	-	-	-	-

CONTENTS

Preface iii

1 SCHEDULE SPECTRUM PROGRAM 1
- Program Functions 1
- SSP Usage 2
  - Operating Procedures 2
  - SSP Output 2
  - Error List 2
- Program Organization Description 6
  - SSP Storage Arrangement 6
  - Tape Layouts 9
  - SSP Execution Description 9

2 DETAIL SCHEDULE REPORT PROGRAM 15
- Program Function 15
- DSRP Usage 15
  - Loading Procedures 15
  - DSRP Output 16
  - Error List 21
- Program Organization Description 21
  - DSRP Storage Arrangement 21
  - Tape Layouts 23
  - DSRP Execution Description 23

3 PROGRESS EVALUATION PROGRAM 27
- Program Function 27
- PEP Usage 27
  - Loading Procedures 29
  - PEP Output 29
  - Error List 33
- Program Organization Description 33
  - PEP Storage Arrangement 34
  - Tape Layouts 34
  - PEP Execution Description 34

4 PROGRESS EVALUATION SORT PROGRAM 40
- Program Functions 40
- PEPSORT Usage 40
  - Loading Procedures 40
  - PEPSORT Output 40
  - Error List 40
- Program Organization Description 40
  - PEPSORT Storage Arrangement 41
  - Tape Layouts 41
  - PEPSORT Execution Description 41

APPENDIX 44

SUBROUTINE ROSTERS 49
- Scheduled Spectrum Program 49
- Detail Schedule Report Program 50
- Progress Evaluation Program 50
- Progress Evaluation Sort Program 51

FLOWCHARTS 53

ILLUSTRATIONS

1 SSP Input/Output 3
2 Schedule Spectrum Summary Report 4
3 SSP Process Sequence Diagram 7
4 SSP Storage Arrangement 8
5 DSRP Parameter Page 16
6 Detail Schedule Report 17
7 DSRP Storage Arrangement 22
8 Project Status Evaluation Report 28
9 PEP Storage Arrangement 35
10 PEPSORT Storage Arrangement 42

1. SCHEDULE SPECTRUM PROGRAM

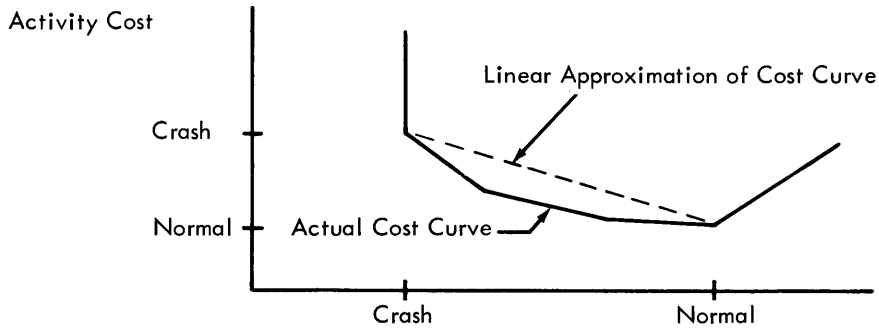
This section describes the SDS Schedule Spectrum Program (SSP), the first program in the SDS Project Management System Series. It is run during the process of determining a schedule for a project. The program reads data describing project networks and creates a spectrum of project schedules which give minimum costs over a range of feasible project durations.

PROGRAM FUNCTION

This program is used in the planning stages of a project as a tool for developing a project schedule. It is through this program that the activity network is first introduced into the system. The main function of the SSP is to analyze cost and time data in order to determine a spectrum of minimum cost schedules over a range of feasible activity durations. It performs this analysis based on four items of data which are input for each activity of the network:

1. Normal duration – The estimated time in which activity can be completed under a normal effort, where a normal effort is assumed to be the one which will accomplish the task at the least cost.
2. Normal Cost – The estimated cost of completing the activity in the normal duration.
3. Crash Duration – The minimum feasible time in which an activity can be completed by applying all practical means of reducing the duration (such as overtime, additional resources, etc.).
4. Crash Cost – The estimated cost of completing the activity in the crash duration.

These four figures are to be estimated based on the assumption that the cost-versus-time curve for each activity will be approximately as illustrated in this diagram:



With this data for each activity, there are many ways in which a project can be expedited to achieve a desired total project duration. It would be impossible through manual computations to determine the best way, even for the simplest cases. The SSP "crashing" routine is able to determine the least costly way of achieving every possible duration of the total project.

The main output of the SSP program is a schedule (on magnetic tape) for every change in the rate of increase of the total project cost per unit of decrease in the total project duration. The printed output from the program is a list of these schedules with the schedule number, total duration, approximate total cost, and, if requested, the completion date of the project for each schedule that was created. This listing is called the Schedule Spectrum Summary Report.

After examination of the summary report, the user is able to determine which schedules he would like to see in detail to help him in his development of a final project schedule. The selected Detailed Schedule Reports will be printed from the DSRP program. After examination of these detailed schedules, the user is able to make desired changes to network data and to make reruns of the Schedule Spectrum Program. This cycle can be repeated as often as necessary to develop a schedule that will be accepted as final.

As activity data is processed through this program, it is written onto magnetic tape. This tape may be used as the basic input to the next run of the SSP, with changes to the data coming in from cards or paper tape.

SSP USAGE

The minimum equipment configuration used by SSP consists of 8192 words of core, 2 magnetic tape units, a typewriter, paper tape or punched card input, and an off-line or on-line printer. The program automatically expands its operational capabilities as the configuration is increased. The maximum number of activities (NA) handled by the program is a function of the total amount of storage in the computer and of the ratio of activities to node points (RAE). The equation for an approximate value of NA, with calendar dating of project completion times on the spectrum summary report, is given below:

$$NA = \frac{\text{Total Storage} - 3150}{\left(3 + \frac{1}{RAE}\right)}$$

$$RAE = \frac{\text{Number of Activities}}{\text{Number of Nodes}}$$

If calendar dating is not requested, a slightly greater number of activities can be processed. An approximate value of NA is determined as

$$NA = \frac{\text{Total Storage} - 1500}{\left(3.5 + \frac{1.5}{RAE}\right)}$$

When RAE equals 1.8, the maximum number of activities handled by the program on an SDS 920 Computer without calendar dating is approximated in the following table:

Available Storage (words)	Maximum Number of Activities
8,192	1,545
12,288	2,490
16,384	3,435
20,480	4,095

These figures are slightly less for operation on an SDS 910 Computer due to the use of programmed operators. A maximum limit of 2047 event nodes can be handled by any of the above configurations.

Operating Procedures

The procedures for running SSP and instructions for filling out data forms for SSP are contained in the SDS Project Management System Reference Manual (publication number 90 08 18). Figure 1 is a diagram of the input/output process.

SSP Output

Output is always provided in off-line format on magnetic tape unit 2. A sample printout from SSP is presented in Figure 2.

An Updated Parameter File, Unscheduled Activity File, and Scheduled Event File are written on magnetic tape unit 1. This tape becomes the input for the Detailed Schedule Report Program (DSRP). It can also be used as the basic input file for subsequent SSP runs. A detailed layout of this tape is given in the appendix.

Error List

Upon detecting an error, SSP enters an error routine that produces an error message on the typewriter. The meaning of each error number is given in the Project Management System Reference Manual.

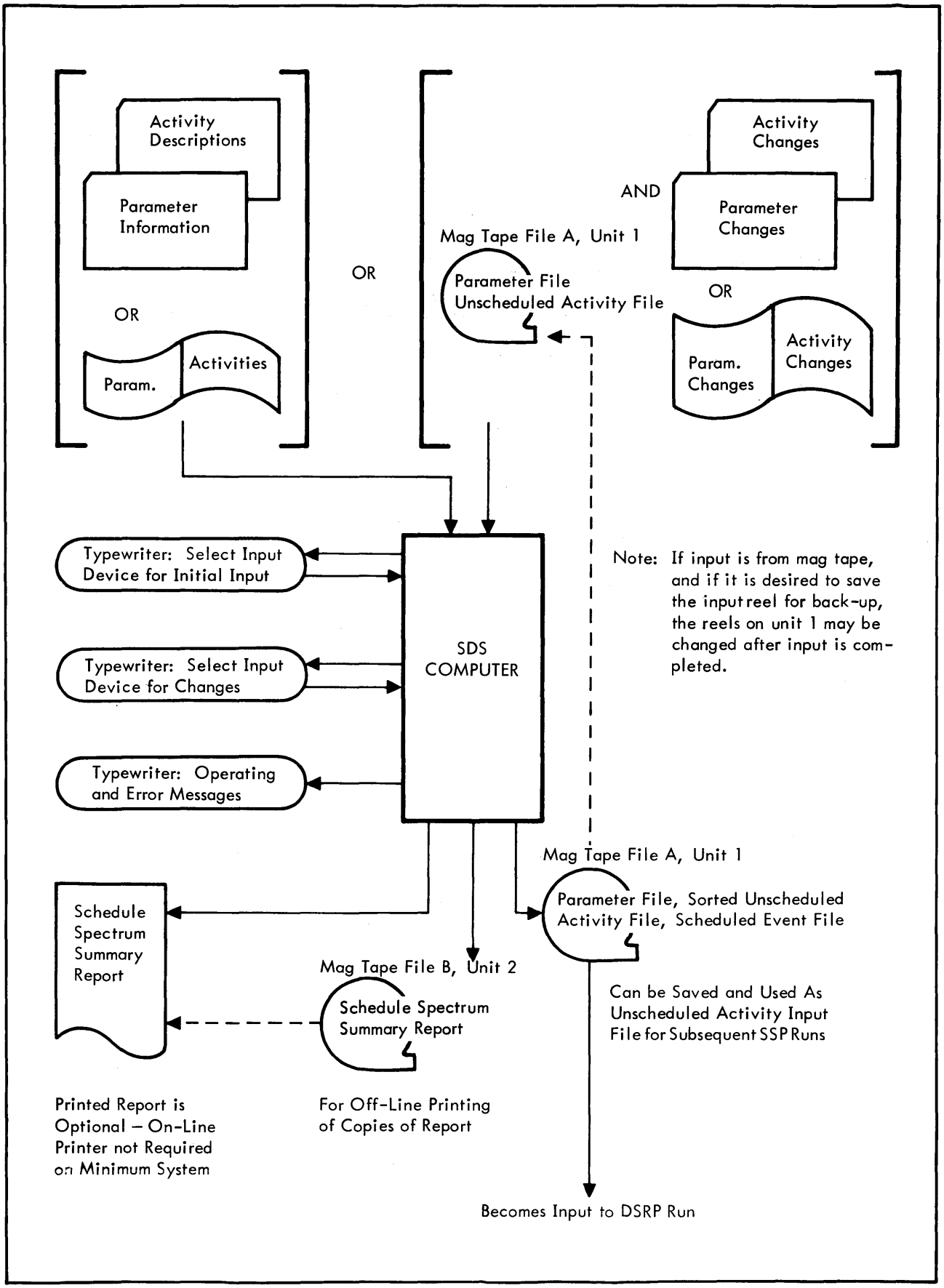


Figure 1. SSP Input/Output

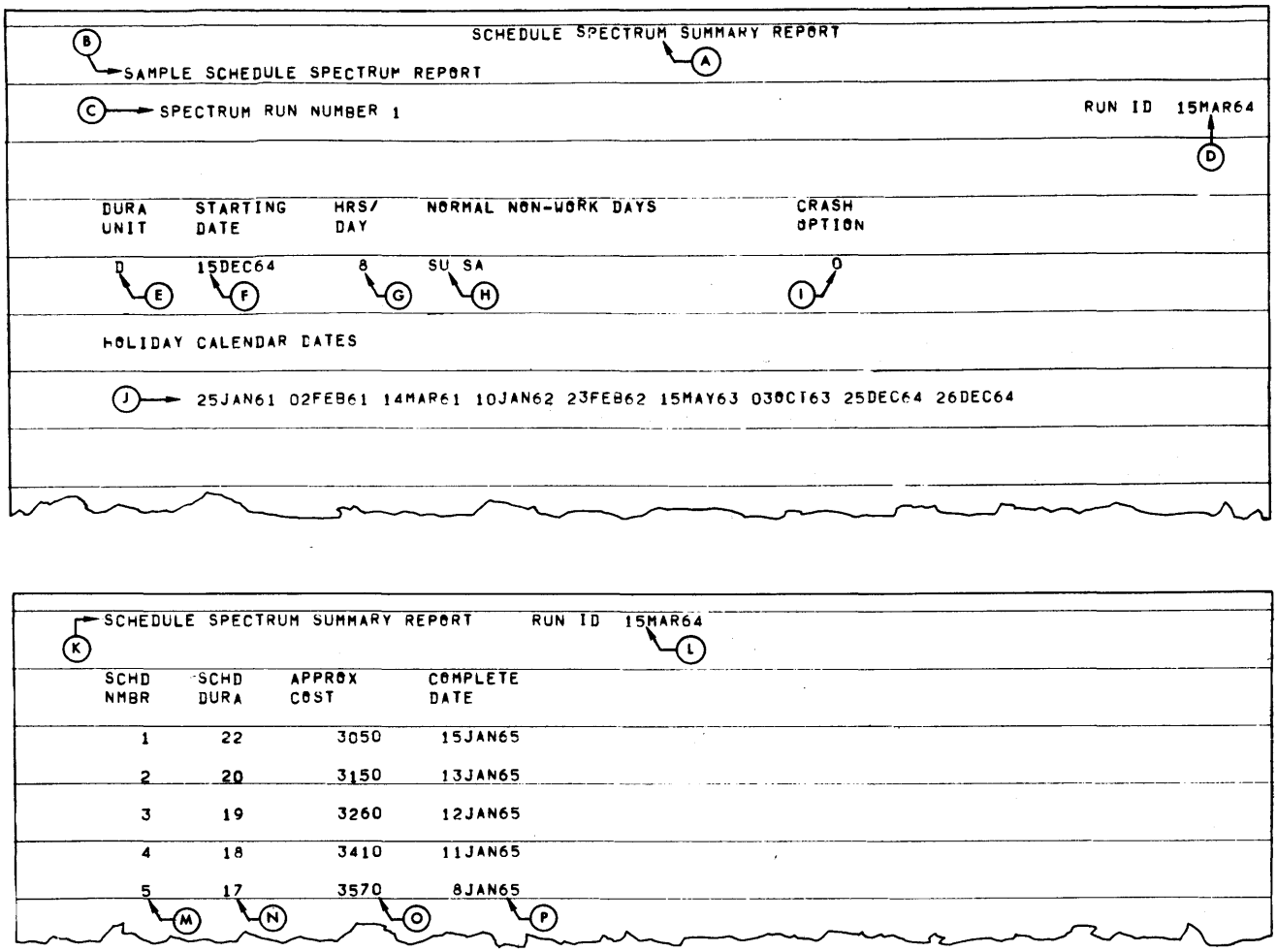


Figure 2. Schedule Spectrum Summary Report

Letters below in "Field" column refer to the sample printout.

Field	Contents	Description
A	Schedule Spectrum Summary Report	Report Heading
B	Any 79 alphanumeric characters	Project Heading From characters 2-80 of the type 1 input record.
C	Any 68 alphanumeric characters	Run Heading From characters 13-80 of the type 4 input record.
D	Any eight alphanumeric characters	Run Identification Code From characters 5-12 of the type 4 input record.
E	H = Hours D = Days W = Weeks M = Months	Type of Duration Units Code From character 2 of the type 2 input record. Shows type of time units in which durations are expressed.
F	Any legal date, blank, or zero. DDMMMYY DD = Day MMM = Month YY = Year	Starting Date From characters 3-9 of the type 2 input record. This is the calendar date (if any) that the program used as the start date of all starting activities.

Field	Contents	Description
G	00-24 or blank	Working Hours Per Day From characters 10-11 of the type 2 input record. Shows the number of hours in a work day.
H	From 0 to 7 day codes: SA = Saturday SU = Sunday MO = Monday TU = Tuesday WE = Wednesday TH = Thursday FR = Friday	Days Off Per Week From characters 12-25 of the type 2 input record. Shows which days of the week have been considered as nonworking days during the assignment of calendar dates.
I	C = Crash Blank = No Crash	Crash Option Code From character 4 of the type 4 input record. Shows if the crash analysis option was selected.
J	From 0 to 164 holiday dates, each of the form DDMMMYY: DD = Day MMM = Month YY = Year	Holiday Calendar Dates From the type 3 input records. Shows which dates have been considered as nonworking holidays during the assignment of calendar dates.
K	Schedule Spectrum Summary Report	Report Heading
L	Any eight alphanumeric characters	Run Identification Code From characters 5-12 of the type 4 input record. This code is printed at the top of each page of this report for identification.
M	1 - 9999	Schedule Number This item is a schedule identification number assigned sequentially by SSP.
N	1 - 4095	Schedule Duration Total project duration for the given schedule.
O	0 - 99999999	Approximate Schedule Cost Approximate total project cost associated with that schedule's duration. Note that this cost is not exact in this report. Due to memory limitations, the costs of each activity are automatically scaled and rounded to a 12-bit binary number, and sometimes significant figures are lost on the right end of this number. Therefore, this cost will not always agree with the one printed on the corresponding report from DSRP, since that one is exact.
P	DDMMMYY or blanks DD = Day MMM = Month YY = Year	Project Completion Date If a start date is given, the calendar date of the completion time of each schedule is determined and printed in this column. There is one exception to this statement: if the size of the network is so great that there is not enough storage space for the calendar routine (approximately 1400 activities with 8192 words of core storage), calendar dating of the total project duration is automatically inhibited.

PROGRAM ORGANIZATION DESCRIPTION

SSP is coded in relocatable form in the SDS META-SYMBOL language. It contains its own loader and, therefore, does not require the MONARCH system tape. The program is divided into many routines which are assembled separately. They are executed under control of the SSP Control Program, using the META-SYMBOL external reference provision. Most of the data used by the routines is located in a region called Common Data and Storage. References to this area also use the META-SYMBOL external reference provision.

The program is constructed in three segments. During execution of program segment I, parameters and the activities making up the network are read from the selected input device and edited. The activity records are written on magnetic tape unit 2, behind the program segments, as the unsorted activity file. Certain information is extracted to make up the abbreviated activity records which are retained in memory for further processing. Dummy tape activity records and abbreviated activity records are also created internally for milestones.

The first part of segment I is not overwritten when segments II and III are read and executed. This group of programs and subroutines, which are common to all three segments, consists of

Common Data and Storage,
SSP Program Control
Magnetic Tape Read Subroutine
Magnetic Tape Write Subroutine
Error Subroutine
Binary to Decimal Conversion Subroutine
Decimal to Binary Conversion Subroutine

Segment II does the network construction and schedule optimization and writes the updated parameter file, the sorted activity records, and the scheduled event file on magnetic tape unit 1.

Segment III reads the scheduled event file, determines costs and calendar dates for each schedule, and prints the Schedule Summary Report.

A summary of SSP processing sequences is given in Figure 3.

SSP Storage Arrangement

The storage arrangement of SSP for the SDS 920 Computer version is shown in Figure 4. The F, S, or T in front of the various routine numbers denotes first (F), second (S), or third (T) program segments.

Segment I (along with the common routines F01 through F09) is loaded from punched cards or paper tape into the computer first. Programs F05 (Magnetic Tape Write Subroutine) through F32 (Calendar Subroutine) are written on magnetic tape unit 2 under control of program F33 (End Segment I Load Routine). All external references in these programs have been fulfilled by the time they are written onto tape. Note that references to segments II and III in the SSP Control Program (F03) have not yet been fulfilled.

The End Segment I Load Routine then transfers control to the Binary Load Routine (F01) with the proper indications set up to retain the symbol table. It is important to retain the symbol table because the common routines (F02 through F09) are not loaded again with segments II and III. In fact, the Common Data and Storage (F02) and the SSP Control Program (F03) cannot be loaded again because they both contain references to segment I programs, which were set up by the Binary Loader during the reading of segment I.

Segment II routines (S1 through S9) are then read from punched cards or paper tape, under control of the Binary Loader. They load over part of segment I, starting at the origin of routine F10. All external references in segment II programs are fulfilled as they are read. Control is then transferred to the End Segment II Load Routine (S09), which writes segment II onto magnetic tape 2 immediately following segment I. The segment II that is written onto tape consists of common programs F05 through F09 as well as S01 through S08. Control is then returned to the Binary Load Routine, again retaining the symbol table.

Segment III routines (T1 through T11) are read from punched cards or paper tape. They are read over segment II, starting at the origin of routine S1. Control is then transferred to the End Segment III Load Routine (T11), which writes segment III onto magnetic tape unit 2, immediately following segment II. The segment III which is written onto tape is made up of program F05 through F09, as well as T01 through T10. Control is then transferred to the Binary Load Routine, again retaining the symbol table. At this point, all external references in the Common Data and Storage Area (F02) and the SSP Control Program (F03) have been satisfied.

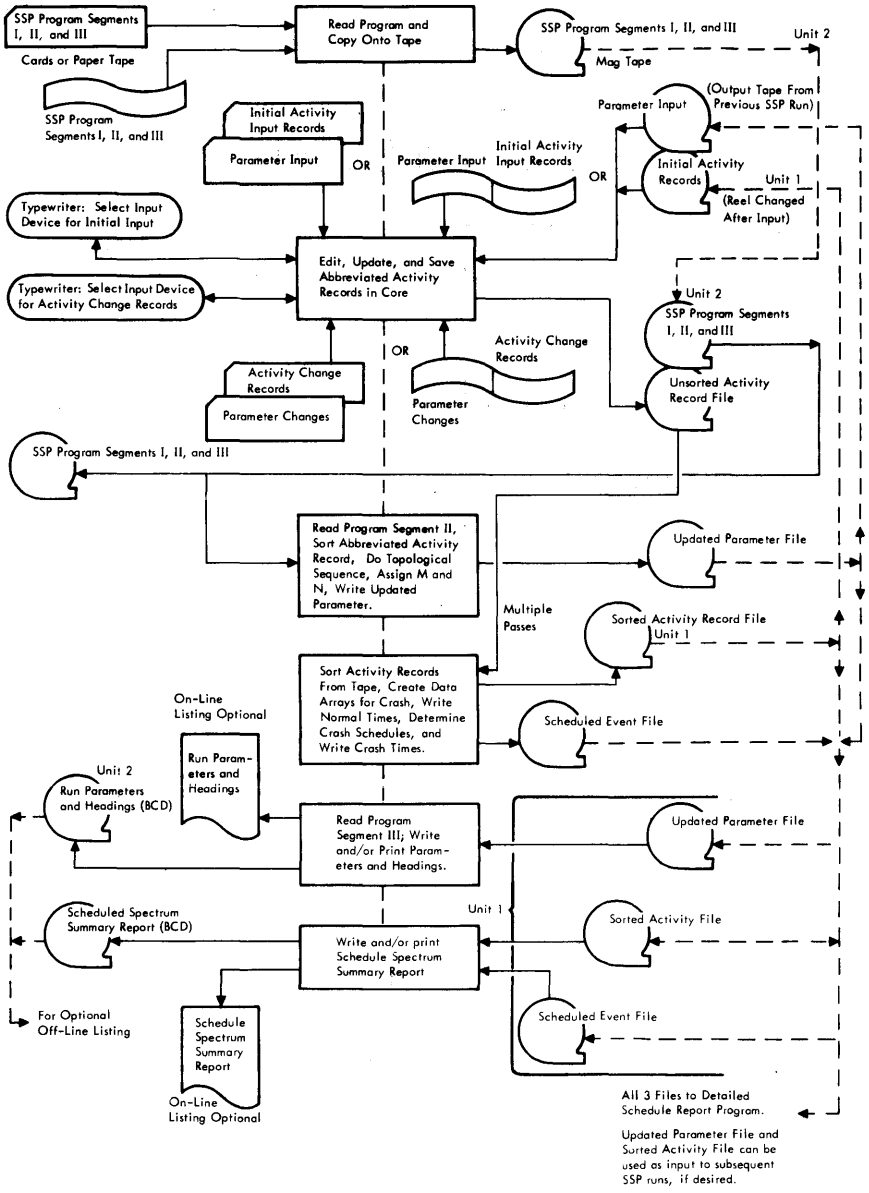


Figure 3. SSP Process Sequence Diagram

<table>
<tr><th>Approx-imate Location[t]</th><th>Pro-gram[tt]</th><th colspan="2">SEGMENT I</th><th colspan="2">SEGMENT II[ttt]</th><th colspan="2">SEGMENT III[ttt]</th></tr>
<tr><td>Variable
$\geq 17777_8$</td><td>F01</td><td colspan="2" rowspan="2">J abbreviated activity record

1/3 available storage</td><td rowspan="2">Sorted Tape Activity Records</td><td colspan="2">M and N node numbers

1 word per activity</td></tr>
<tr><td rowspan="2">Sequenced
Selected</td><td colspan="2" rowspan="2">Durations

1 word per activity</td></tr>
<tr><td>11152_8</td><td>T12</td><td colspan="2">I abbreviated activity record

1/3 available storage</td><td>M and N node numbers</td></tr>
<tr><td></td><td>F33
F32
F31</td><td colspan="2" rowspan="3">Miscellaneous abbreviated activity record

1/3 available storage</td><td rowspan="3">Activity successor</td><td>Node times

1/2 word per node</td><td colspan="2">Scaled slopes

1/2 word per activity</td></tr>
<tr><td></td><td>F30
F29
F28
F27
F26</td><td rowspan="2">Utility flows

1 word per activity</td><td colspan="2" rowspan="2">Scaled normal costs

1/2 word per activity</td></tr>
<tr><td>07643_8</td><td>F25
F24
F23</td></tr>
<tr><td>06547_8</td><td>F22</td><td colspan="2" rowspan="2"></td><td rowspan="2"></td><td rowspan="2">Node labels

1 word per node</td><td>T11
T10
T09
T08
T07</td><td>Scheduled times

1 word per node</td></tr>
<tr><td></td><td>F21
F20
F19
F18
F17</td><td>S09
S08
S07
S06
S05</td><td></td><td>T06

T05</td><td>Available for data arrays, without calendar dating.</td></tr>
<tr><td>06120_8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>04365_8</td><td>F16
F15
F14
F13
F12
F11</td><td colspan="2" rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td>T04

T03

T02</td><td rowspan="2"></td></tr>
<tr><td></td><td>S04
S03
S02</td><td></td><td></td></tr>
<tr><td>02716_8</td><td>F10</td><td>(I/O and parameter area)</td><td>S01</td><td colspan="2"></td><td>T01</td><td>Input/output area</td></tr>
<tr><td>00000</td><td>F09
F08
F07
F06
F05
F04
F03
F02</td><td colspan="6"></td></tr>
</table>

[t]The locations in this column are subject to change, and therefore, can only be regarded as approximations.

[tt]See Routine Roster for names of routines that correspond to these deck numbers.

[ttt]The data arrays in these columns are filled from the end of available storage towards the end of the program.

Figure 4. SSP Storage Arrangement

The SSP Start Routine (T12) is read from punched cards or paper tape into locations immediately following the end of segment I (behind routine F33). Control is transferred to the SSP Start Routine. It rewinds the tape and reads segment I (starting with routine F05). Then control is transferred to the SSP Control Program (F03), and the loading procedure is complete.

During execution of SSP, segments II and III are read from magnetic tape at the proper times under control of the SSP Control Program (F03). During the execution of the various segments, parts of the programs are overlayed with data, as shown in Figure 4.

Tape Layouts

There are three files on the output tape on unit 2: the Parameter File (1 BCD Record), followed by the Sorted Unscheduled Activity Record File (1 BCD record per activity), followed by the Scheduled Event File (1 binary record per schedule). This record can be used as input to DSRP or as input to subsequent SSP runs. The details of these record formats are presented in the appendix.

SSP Execution Description

The SSP Control Routine (F03) dictates the execution sequence of various routines that comprise SSP. After a routine is executed, control is returned to the Control Routine to determine which routine should be executed next. In the following discussion the statement that "control is returned to the Control Routine which in turn transfers control to routine X" is implied preceding the description of each routine. The routines are described in the order in which they are executed.

After the program has been loaded and written onto magnetic tape, the Non-Activity Oriented Initialization Routine (F17) determines the extent of core and sets up data words to define the locations of the abbreviated activity records, the activity successor records, the M and N node numbers, the selected activity area, the sequenced activity area and the sorted tape activity record area. Next the I/O Device Selection Routine(F18) requests the input and output configuration description from the typewriter. Then the Parameter and Holiday Input Routine (F19) reads the parameter records (see detailed input record description) from the selected input device into the parameter section of the Input/Output Area (F10). This program continues to read parameter cards until it encounters a type-5 or a type-9 record (which is saved for later processing).

Control is returned to the SSP Control Program which determines the input device. If input was from punched cards or paper tape, no changes to the parameters or activities can be input, and an indication of this fact is set up in storage. If the input is from magnetic tape, control is transferred to the Change Input Device Selection Routine (F20), which requests a type-in of an N for no changes, a C for changes on cards, or a P for changes from paper tape. If changes are to be input, control is transferred to the Parameter and Holiday Input Routine; if no changes are to be entered, control goes to the Parameter Conversion Routine.

The Parameter Holiday and Input Routine (F19) reads the changes and stores them over the original parameters of the given types. Changes are read and stored until an activity change or a type-9 record is read (the type-9 record is saved for later processing).

The Parameter Conversion Routine (F21) converts the parameters from their BCD codes into forms more suitable for internal processing and saves these new codes.

Next the Holiday Sort Routine (F22) sorts the holidays that were input with the parameters into chronological sequence, leaving them in the same area. Then the SSP Control Program determines whether or not a calendar start date was input with the parameters. If one was not input, control is transferred to the Activity Record Input Routine (F24). If one was input, the Calendar Initialization Routine (F23) is executed, which in turn transfers into the Calendar Routine (F32) at its setup entry location $CALE1. This entry sets up all data needed to compute calendar dates from basic working units or basic working units from a calendar date. Data calculated during calendar setup include:

1. A vector containing the number of working days in each of the years (maximum of 15) from the start date of the project.
2. A binary holiday vector containing the holiday dates in terms of total accumulated days since the start of the project.
3. A vector containing the subscript of the first holiday in each year.
4. A vector containing the number of holidays in the next 15 years.

5. The day of the week on which the project starts.
6. The type year in which the project starts (leap year, year after leap year, etc.).

The precalculation of the above data makes possible the rapid calculation of individual dates and durations. These calculated items are stored in the Common Data and Storage Area for later use. Control is returned to the Calendar Initialization Routine, which returns to the SSP Control Program.

The Activity Record Input Routine (F24) processes the first activity record from the area in which it was saved during the parameter input. The rest of the activities are read from the selected normal input device. As each record is read, validity checks are made on the costs and durations, and leading blanks are replaced by zeros wherever pertinent. The records are moved from the input area into the output area of the Input/Output Area (F10) and changed to the tape activity record format. One activity record is read during each entry of this routine. If this record is not an end-of-activity record, control is returned to the normal return in the SSP Control Program, where a test is made to determine if the record contains a milestone date or time. If it does contain one, the Milestone Time Setup Routine (F25) is entered. If the milestone was given as a date, it is converted to relative units by the Date to Basic Units Conversion Routine (F31), which executes the Calendar Routine. The milestone time, in relative units, is stored over M temporarily in the tape activity record.

The Write Activity Record (on Unit 2) Routine (F26) sets up the record number in the tape activity record and writes the record on magnetic tape unit 2.

This process of reading activity records from the selected input device, editing them, and writing them on magnetic tape unit 2 continues until an end-of-activity (input) record is read. At this point, if the input device was magnetic tape and if there are changes, the change records are read from the selected input change device, again using the Activity Record Input Routine (F24). They are processed, executing the same series of routines described for the normal activity input. The changes are written as part of the unsorted activity record file, immediately following the regular activities.

When an end-of-activity (input) record (type 9) is processed or if there were no changes, the End Activity File Record Routine (F13) is executed. An end-of-activity file record is written after the last activity change record on magnetic tape unit 2. This is followed by the erasure of an area of the tape, so that the tape can be repositioned for writing milestone dummy activity records at the end of the file later. A rewind is initialized on unit 2, and control is returned to the SSP Control Routine.

If the original activity input was from magnetic tape, the Type Reel Change Message Routine (F27) is executed to notify the operator that, if he wants to save the original input tape in order to be able to rerun, he should change reels on magnetic tape unit 1. There is no halt at this point, and processing continues while the reels are being changed. If the original input was not from magnetic tape, this program is not executed.

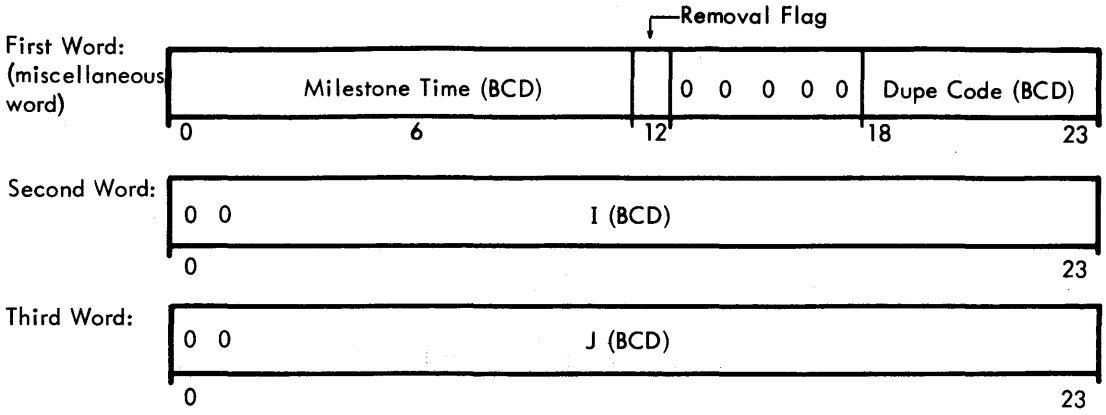
Next, the Skip Program Segments Routine (F28) delays until the rewind on magnetic tape unit 2 is completed and then positions the tape past the program segments to the beginning of the unsorted activity record file which was just written.

The Create Abbreviated Activity Records Routine (F11) reads the records, one at a time, from the unsorted activity record file on magnetic tape unit 2 into the output area of the Input/Output Area (F10). Certain information is extracted from each record and saved in core storage to create the abbreviated activity records. Each abbreviated activity record consists of three words, which are stored in three separate areas, as shown in Figure 4. One-third of the available core storage is assigned for each of these areas. The area formerly occupied by routines F22 through F33 is no longer required and is considered to be available for storage of abbreviated activity records.

The first word of the abbreviated activity record is referred to as the miscellaneous abbreviated activity record word. Initially, it is set up to contain the milestone time in binary in the first 12 bits (0-11), zeroes in bit positions 12 through 17, and the Duplicate I, J Differentiation Code in BCD in bits 18 through 23. The second word contains the I identification number of the activity in BCD, and the third word contains the J identification number of the activity in BCD. Note that the first and second bits of I and J are zero because of the restriction that the first character of I and J must be numeric. The contents of these words are changed throughout the processing.

Also during the execution of this routine, the transaction code of each activity record is tested to determine whether or not the removal flag should be turned on (bit 12 of the miscellaneous abbreviated activity record word). This flag is turned on in all abbreviated activity records that are to be deleted due to the input of replacements and deletions. It is also turned on when the input is a milestone dummy activity record that was created internally and written onto the tape on the previous SSP run.

The three words of the abbreviated activity records can be represented as follows:



This program is completed when the end-of-activity file record is read. At this time, the tape on unit 2 is backspaced so that it is positioned ahead of the end-of-activity file record in order that this record can be overwritten.

Control branches to the Set Up Milestone Dummy Activity Records Routine (F12), which makes a pass through all the abbreviated activity records, looking for those that contain relative milestone times. For each activity it finds that has a milestone, the program creates three dummy tape activity records and adds them to the unsorted activity record file on magnetic tape unit 2. It also creates the three corresponding abbreviated activity records and saves them in core storage. The purpose of this procedure is to present to the crash routine a set of activities that will cause the crashing to proceed in a manner which will force milestones to be met, if possible, before allowing crashing of any of the activities that follow the last milestone.

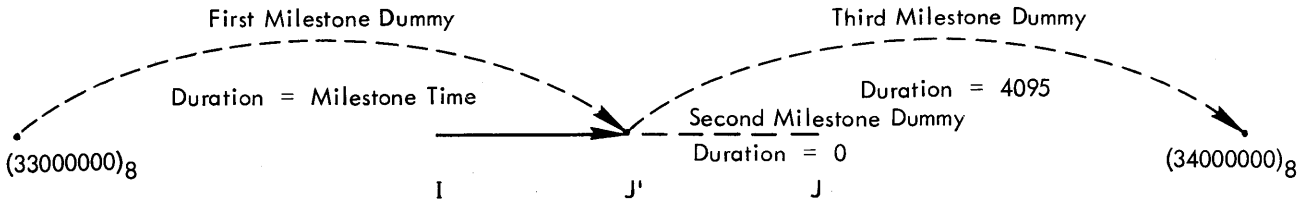
The first of the three milestone dummies for every activity has an I of 33000000_8, which is a dummy start node. J is equal to the J of the activity with milestone time, but with a bit merged into the second bit position. Its duration is the milestone time. This number is stored in the duration field of the dummy tape record and in the first 12 bits of the miscellaneous abbreviated activity record word.

The second dummy has an I equal to the J of the activity with the milestone time, but with the zero bit merged into the second bit position. Its J is actually the same as the J of the activity with the milestone time. Its duration is zero.

The third milestone dummy has an I equal to the J of the activity with the milestone time but with the 1 merged into the second bit position. Its J is always 34000000_8, which is a dummy ending node. Its duration is the maximum allowable duration $- (4095)_{10} = (7777)_8$.

After the completion of the above processing, one change is made in the abbreviated activity record of the activity with the milestone time. Its J is changed by merging the 1 bit into its second bit position.

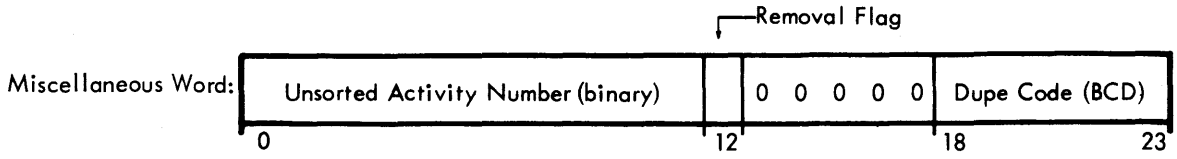
The changes made to the network for each activity found with a milestone time are summarized in the following diagram, where J' indicates the J of the activity with the milestone time but with the 1 merged into its second bit position.



In addition to setting up the milestone dummies, this routine changes I to 37777777_8 for each activity in which the delete flag is on. This procedure causes the deleted activities to sort at the end of the file when the abbreviated activity records are later sorted on I and J.

Control is returned to the SSP Control Routine, which branches again to the End Activity File Record Routine (F13). This program writes the end-of-activity file record (type 6000), initiates a rewind of magnetic tape unit 2, and returns to the SSP Control Program.

Control is then transferred to the Assign Unsorted Activity Numbers Routine (F14), which numbers the abbreviated activity records sequentially according to their current position in core storage. This unsorted activity number is stored in the first 12 bits (0-11) of the miscellaneous word of each abbreviated activity record (over-writing the relative milestone time, which is no longer required). The miscellaneous word now looks as follows:

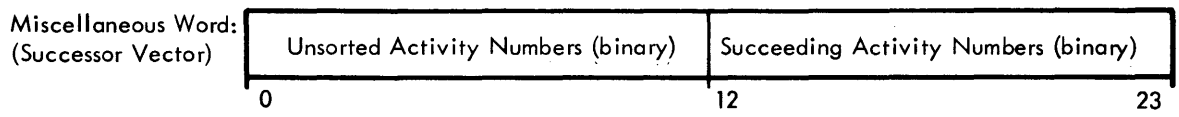


If the original input device was magnetic tape, the Reel Change Verification Routine (F15) is executed next. This program types a message notifying the operator that magnetic tape unit 1 is about to be used for output. Then the program halts. When the operator is ready for the run to continue, he clears the halt. Control is returned to the SSP Control Routine. If the original input device was not magnetic tape, this routine is not executed.

The SSP Control Program branches next to the Write Updated Parameters Routine (F16), which rewinds magnetic tape unit 1, erases an initial gap on that tape, and writes the updated parameter file as the first record on that tape. It also moves the number of activities and the number of events into the parameter area of the Input/Output Area (F10), before it writes the record from that area.

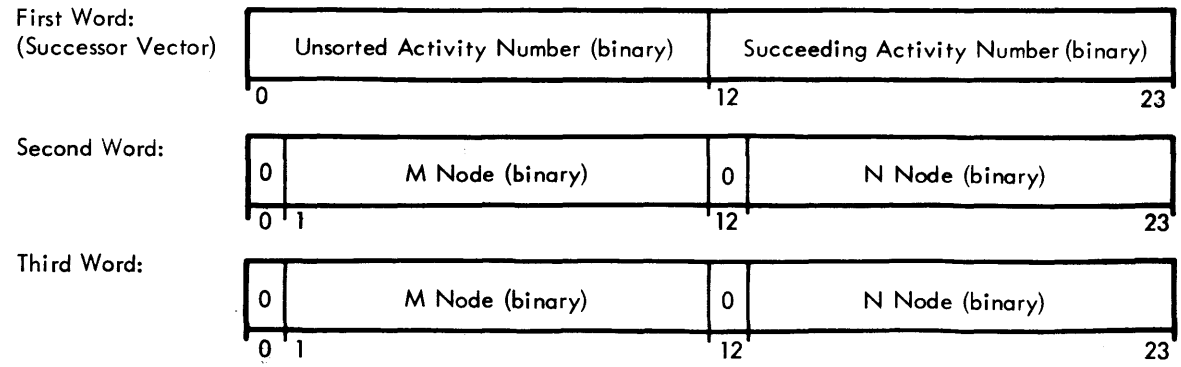
The Topological Sort Routine (S05) is executed next to assign each of the activities to a position in a thread such that every item on the thread follows all its predecessor activities. This is accomplished by creating an arbitrary initial thread, then moving along this thread, examining the I and J of each activity to find if the activity has successor activities that are assigned to the thread ahead of the given activity. When one is found, it is moved to the end of the thread, from which position it will later be examined to determine if it still follows all its predecessors. If it does not, it will be moved to the end again. This process continues until the last activity of the thread has been examined without finding any succeeding activities ahead of it. Activities are not physically moved from their I, J, Duplication Code sequence during this process. Instead, the last 12 bits (12-23) of each entry in the miscellaneous word of the abbreviated activity record is set to contain the number of the activity immediately succeeding it in the thread. These 12-bit items are referred to as the successor vector.

The miscellaneous word of the abbreviated activity record now appears as follows:



Once the topological thread has been determined, it is possible to assign new identification numbers M and N to the nodes, based on their order of appearance within the thread. This is accomplished by the execution of the M and N Assignment Routine (S06). It should be noted that all starting nodes (nodes with no activities leading into them) are assigned the number 1, and all ending nodes (nodes with no activities leading out of them) are assigned the highest N.

At the completion of this process, M and N are stored in the topological sequence in the I words of the abbreviated activity record and repeated again the J words of the abbreviated activity record. They are used only in the I words. The abbreviated activity record words now appear as follows:



The next task is to sort the activity status records on tape into the newly determined topological sequence, using the Topological Tape Sort Routine (S07). Since the program must operate using only two tape units, the sort is required to make multiple passes through the unsorted activity record file (on magnetic tape unit 2), pulling off as many records as possible (n records) each pass, and writing them on the sorted activity record file (magnetic tape unit 1).

This is accomplished by extracting the next n items from the successor vector (as defined by the thread) and moving them into the selected area (see Figure 4). The number of the activity in the sorted sequence is now inserted in the last 12 bits of the extracted words. The first 12 bits of these words retain the number of the activity in the original sequence. These extracted words are now sorted on the original sequence number and stored into the sequenced area. They are now in the same sequence as the records in the unsorted activity record file.

Since these records are not contiguous, a complete pass through the unsorted file is required to read each group of n activity records. As the records are read, their input locations are calculated, based on their new sequence number; thus, after each pass, storage will contain the next n records in the assigned topological sequence. They are now written on the sorted activity file. As many passes are made through the unsorted file as are required to place all the records in their new sequence in the sorted file.

The SSP Control Program then transfers to the Activity Oriented Initialization Routine (S08), which sets up all addresses required for locating those data arrays that vary in length depending on the number of activities or number of nodes in the network.

Next, control branches to the Create Crash Data Arrays Routine (S03). This routine reads each activity record from the sorted activity record file (magnetic tape unit 1), and sets up M and N, duration, and cost per unit of duration (slope) for each activity. Note that the slopes are set to a large number (4095) for every dummy activity and for every milestone dummy, and that the slopes are scaled so that the largest one does not exceed 12 bits. The routine then calculates the scheduled node times for each node, based on normal activity durations.

Note that when these normal node times are set up, tests are made on end nodes to determine if the time results from an end milestone dummy. The greatest milestone dummy node time is saved.

Now the Write Scheduled Event Time Record Routine (S01) writes a single binary record on magnetic tape unit 1, which contains one word for each node time. Also, at this time, milestone dummy end node times are reduced to the maximum allowable duration (4095).

The Control Program tests whether or not the crash option was selected. If it was, the Crash Routine (S02) is executed to determine which individual activity durations should be compressed in order to decrease the total project's duration at a minimum increase in cost. Once these activities have been determined, the schedule is compressed as far as possible without increasing the rate at which the cost of the total project is rising (constant slope).

Upon initial entry to the Crash Routine, all activities in the network are set at their normal durations. The activity durations to be compressed are determined and the node times affected by these duration changes are calculated. When an increase in the cost slope is reached, the routine returns to outer control and the new node times are recorded on tape using the Write Schedule Event Time Record Routine (S01). Outer control then re-enters the Crash Routine for the node times corresponding to the next slope change. This cyclic process is repeated until the shortest total project duration is obtained.

The shortest possible total project duration is reached when all activities along one of the current critical paths of the network have had their durations brought back to their crash durations. When this condition occurs, the Crash Routine sets an indicator for the outer control and returns.

The mathematical algorithm used to compress the project schedule along a path of minimum cost is a variation of a method developed by D. R. Fulkerson of RAND Corporation. This algorithm is described in detail in an article entitled "A Network Flow Computation for Project Cost Curves," which appeared in the January 1961 issue of Management Science, Volume 7, No. 2.

When the final schedule has been determined and written on magnetic tape, program segment III is read into core storage from magnetic tape unit 2. The first routine executed in segment III is the Write End of Time File Record Routine (T08), which writes a record with a 9000 record type to indicate the end of the file.

Next, the Read Parameters from Unit 1 Routine (T09) rewinds unit 1 and reads the updated parameter file back into core storage. The Parameter Print Routine (T10) is then executed to print the heading page of the summary report, including project and run heading, selected run option codes, starting date, and holidays. The page is printed using the Report Line Output Routine (T03). It is also written at the beginning of the tape on unit 2, with the required format digits for off-line listing.

Control branches to the Set Up Data Arrays for Summary Report Routine (T07) which reads the activity records from the sorted activity record file on magnetic tape unit 1 and stores the normal cost for each activity in an array in core storage. Each cost is shifted right as many bits as are required so that the maximum normal cost can be contained in 12 bits. If the activity number is odd, this scaled normal cost is stored in the left half of the word, and the scaled normal cost for the next even activity number is stored in the right 12 bits of the same word. Also in this routine, if the activity is noncontinuous, a 1 is stored in the first bit position (sign bit) of that activity's entry in the M and N node number array. If the activity is a milestone dummy, a 1 is stored in bit position 12 of that activity's array in the M and N node number array.

The Schedule Summary Report Output Routine (T02) prints the page headings and the Summary Report, using the Report Line Output Subroutine (T03), which also puts the line image with appropriate format control digits onto magnetic tape unit 2. The routine reads each scheduled event time record from magnetic tape unit 1 and outputs one line for each schedule. The program must return the normal cost and slope to their original scales. It then calculates the actual cost of each activity for a given schedule based on the normal cost, slope, node times, noncontinuous activity indicator, and milestone indicator for that activity. The calculated cost of all the activities in the network is accumulated, to double precision, in order to determine the approximate total cost of the schedule for output.

If a project starting date was given and if sufficient core storage is available to retain the Calendar Routine, the project completion date is determined from the project completion time, using the Date to Basic Units Conversion Routine (T06), which uses the Calendar Routine (T05). This date will also appear on the line of the report for the given schedule.

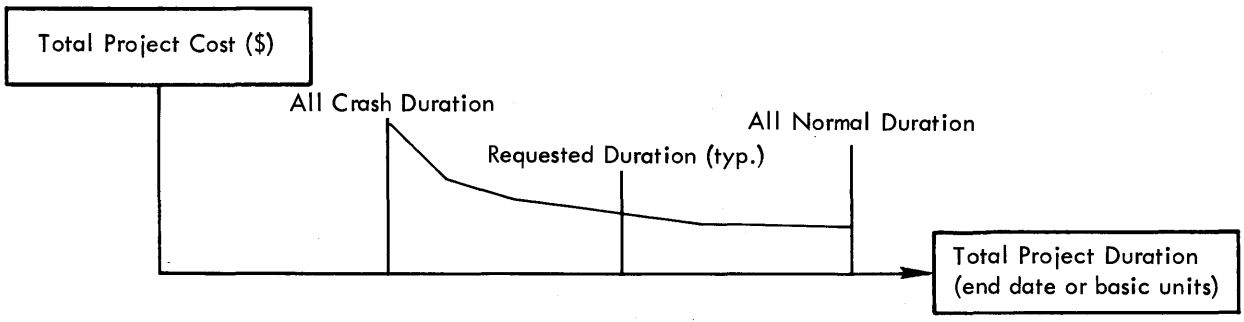
Control is finally returned to the SSP Control Program, which branches to the Error Routine (F06) to print the end of the program message.

2. DETAIL SCHEDULE REPORT PROGRAM

This section describes the SDS Detail Schedule Report Program (DSRP), the second program in the SDS Project Management System Series. It receives its input from the Schedule Spectrum Program (SSP) and provides the initial progress evaluation run input to the Progress Evaluation Program (PEP).

PROGRAM FUNCTION

The primary function of the Detail Schedule Report Program (DSRP) is to produce selected detail schedules from the information output on tape 1 by the Summary Spectrum Program (SSP). This tape contains system run parameter data, unscheduled activity information, and early node times at each slope change point on the Total Cost vs. Project Duration Curve. An example of the spectrum curve is shown below.



DSRP can produce detail schedules for any specified integer duration between the "All Normal" and "All Crash" durations of the project schedule spectrum. As many schedules as desired may be requested by the user. These schedules may be requested in terms of end duration in basic units of working time, by end of project data, or by schedule number. Likewise, all intermediate times within the detail schedule may be obtained in terms of a calendar date and/or in basic working units. If schedule numbers are used, no interpolation between SSP output schedule time is allowed.

The detail schedule report contains information defining each activity's early start and finish times, late start and finish times, free and total float, expected cost and duration, milestone status and other identifying comment fields.

A secondary function of DSRP is to provide the initial input (tape) for the Project Evaluation Program (PEP).

DSRP USAGE

The minimum configuration used by the DSRP consists of 8192 words of core, two magnetic tape units, paper tape reader, on-line typewriter, and an off-line printer. It is designed to expand its operational capabilities automatically as the configuration is increased. Provisions allow operation in conjunction with card reader input and on-line printer output. The number of activities processed by the program can be increased by the addition of more core memory.

The maximum number of activities handled by the program operating on an SDS 920 Computer are shown below.

Memory (words)	Activities
8, 192	1645
12, 288	3005
16, 384	4096

These figures are slightly less for the 910 Computer due to the use of programmed operators. A maximum limit of 2047 event nodes can be handled by any of the above configurations, independent of the number of activities.

Loading Procedures

Procedures for operating the DSRP and instructions for filling out data forms for the DSRP are contained in the SDS Project Management System Reference Manual (publication number 90 08 18).

DSRP Output

Output is always provided in off-line BCD format on tape 2, whether or not an on-line printer is attached to the system. The detail schedule report off-line provision allows the user to obtain multiple copies of the output without rerunning the program.

A sample printout from the DSRP is presented in Figures 5 and 6. For a detailed description of the printout, refer to the SDS Project Management Reference Manual.

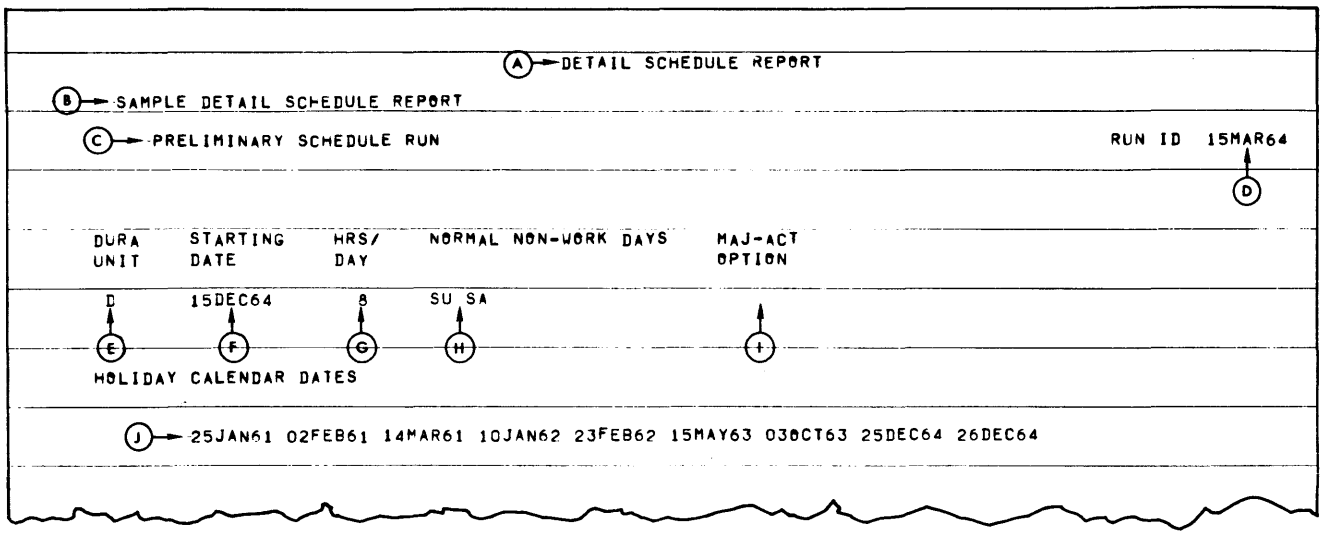


Figure 5. DSRP Parameter Page

The letters below in the "Field" column refer to the sample printout.

Field	Contents	Description
A	Detail Schedule Report	Report Heading
B	Any 79 alphanumeric characters	Project Heading From characters 2-80 of the type 1 input record.
C	Any 68 alphanumeric characters	Run Heading From characters 13-80 of the type 4 input record.
D	Any 8 alphanumeric characters	Run Identification Code From characters 5-12 of the type 4 input record.
E	H = Hours D = Days W = Weeks M = Months Blank	Type of Duration Units Code From character 2 of the type 2 input record. Shows type of time units in which durations are expressed.
F	Any legal data (DDMMMYY) or blank or zero	Starting date From characters 3-9 of the type 2 input record. This is the calendar date (if any) which the program used as the start date of all starting activities.
G	00-24 or blank	Working Hours per Day From characters 10-11 of the type 2 input record. Shows the number of hours in a work day.

Field	Contents	Description
H	From 0 to 7 day codes: SA = Saturday SU = Sunday MO = Monday TU = Tuesday WE = Wednesday TH = Thursday FR = Friday	Days Off per Week From characters 12-25 of the type 2 input record. Shows which days of the week have been considered as nonworking days during the assignment of calendar dates.
I	M = Major activity print Blank = All activity print	Major Activity Option Code From character 3 of the type 4 input record. An M indicates that only those activities previously defined on SSP activity input records as major activities are printed on the Detail Schedule Report. A blank or any other character indicates that all activities are included on the report.
J	From 0 to 164 holiday dates, each of the form DDMMMYY: DD = Day MMM = Month YY = Year	Holiday Calendar Dates From the type 3 input records. Shows which dates have been considered as nonworking holidays during the assignment of calendar dates.

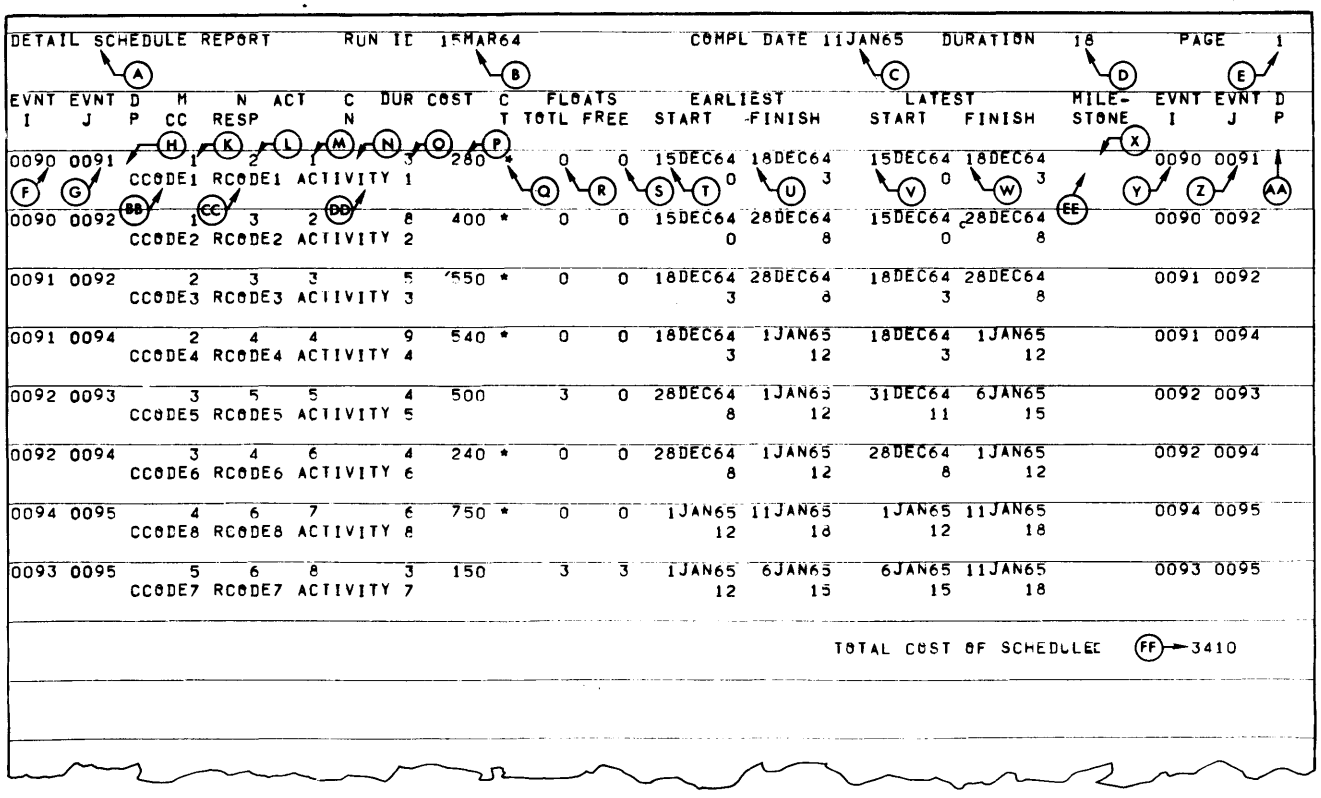


Figure 6. Detail Schedule Report

Field	Contents	Description
A	Detail Schedule Report	Report Heading
B	Any eight alphanumeric characters	Run Identification Code From characters 5-12 of the type 4 input record. This code is printed at the top of each page of this report for identification of the run.
C	A date of the form DDMMMYY	Requested Completion Date If the schedule was requested in terms of a calendar date, that date is printed in this field. If it was requested in terms of units of duration, this is the calendar date which corresponds to that number of units from the starting date specified on the type 2 record used in this DSRP run.
D	1-4095	Requested Duration If the schedule was requested in terms of a total project duration, that duration is printed in this field. If it was requested in terms of a calendar date, this field contains the duration corresponding to that date. The project start date specified on the type 2 record for this DSRP run is always used in this calculation. Note that the late start and finish times and total float calculations are based on this requested project duration.
C & D		If the schedule was requested by a schedule number, neither the requested completion date nor the requested duration is printed. Instead, those two items are replaced by: SCHEDULE NO. XXXX where XXXX is the number of the schedule requested on the type 8 input card.
E	1-4095	Page Number Assigned consecutively to each page of a schedule. Starts over from 1 for each requested schedule.
F	NAAA: N = any number AAA = any three alphanumeric characters	I Identification code of starting node of the activity. From characters 3-6 of the SSP type 5 record.
G	NAAA: N = any number AAA = any three alphanumeric characters	J Identification code of ending node of the activity. From characters 7-10 of the SSP type 5 record.
H	Any alphanumeric character	Duplicate I, J Differentiation Code From character 11 of the SSP type 5 record.
K	1-2047	M Identification number assigned internally to the starting node of the activity.
L	1-2047	N Identification number assigned internally to end node of the activity

Field	Contents	Description
M	1-4095	Activity Number Numbers assigned internally in increasing order to activities after they are resequenced internally into topological order (that is, an order such that no activity is found ahead of any activity which precedes it in the network diagram).
N	N = Noncontinuous Blank = Continuous	Cost Continuity Code From character 28 of the SSP type 5 input record. N in this field indicates a noncontinuous cost increase per unit of duration decrease (i.e., during the crashing procedure the activity could be considered at either its normal duration or its crash duration – never anywhere between these two durations). Any code other than N indicates a continuous linear cost/duration relationship. This field is meaningful only when using the crash option. (See discussion of schedule optimization.)
O	0-999	Scheduled Duration The duration used for the given activity in the determination of the given schedule. If the crash option (schedule optimization option) was not used, this duration is the normal duration from characters 15-17 of the SSP type 5 input record. If the crash option was used, this duration is determined internally to be the optimum duration for the given activity which can achieve the total project duration of the given schedule. It will be greater than or equal to the crash duration and less than or equal to the normal duration.
P	0-99999	Scheduled Cost This is the cost corresponding to the scheduled duration. If the crash option (schedule optimization option) was not used, this cost is the normal cost from characters 18-22 of the SSP type 5 input record. If the crash option was used, this figure is calculated from the scheduled duration, based on the assumption of a linear relationship between the normal and crash costs and durations. It is the minimum cost for the given activity which can achieve the total project duration of the given schedule. It will be greater than or equal to the normal cost and less than or equal to the crash cost.
Q	* = On critical path blank = Not on critical path	Critical Path Indicator Shows whether or not the given activity is on the critical path for the given schedule. An activity is on the critical path if its total float is zero or negative, which means that any delay in the completion of the activity would be expected to cause a delay in the completion of the total project.
R	-4095 to 4095	Total Float The amount by which the actual completion of the activity may be delayed beyond its earliest finish time before it causes a delay in the requested finish time of the entire project. It is calculated as the difference between the latest finish time and the earliest finish time of the activity. If the total float is zero or negative, the activity is on the critical path. If the latest finish time precedes the earliest finish time, so that the total float is negative, the activity is on a path which is preventing the requested project completion time from being met, by the specified number of units.

Field	Contents	Description
S	0-4095 (always positive)	Free Float The amount by which the completion of an activity may be allowed to slip beyond its earliest finish time before it causes a delay in the earliest start time of any of the activities following it. It is calculated as the difference between the earliest of the earliest start times of all activities immediately following the given activity and the earliest finish time of the given activity.
T	A date of the form DDMMMYY or blank and a number 0-4095	Earliest Start Time The earliest time at which the activity can start, based on its position in the network. It is calculated as the sum of the durations of all activities along the longest path from the start of the network to the start of the given activity. The time in units of duration is printed on the second line of activity information. Directly above this time is the corresponding date if a project starting date was specified on the type 2 input record for the run.
U	A date of the form DDMMMYY or blank and a number 0-4095	Earliest Finish Time The time at which a given activity should be finished if it is started at its earliest start time. It is calculated by adding the activity's duration to its earliest start time. The time in units of duration is printed on the second line of activity information. Directly above this time is the corresponding date, if a project starting date was specified on the type 2 input record for the run.
V	A date of the form DDMMMYY or blank and a number 0-4095	Latest Start Time The time by which the activity should start in order to be finished by its latest finish time. It is calculated by subtracting the activity's duration from the latest finish time. The time in units of duration is printed on the second line of activity information. Directly above this time is the corresponding date if a project starting date was specified on the type 2 input record for the run. If the latest start time is negative (i.e., precedes the project start time), the date in this field will be blank, and the time will be zero.
W	A date of the form DDMMMYY or blank and a number 0-4095	Latest Finish Time The time by which the activity should be finished in order to avoid extending the completion time of the project beyond the completion date entered at the start of the run. It is calculated as the requested project completion time minus the sum of the durations of all activities along the longest path from the completion of the given activity to the requested project completion date. Directly above this time is the corresponding date if a project starting date was specified on the type 2 input record for the run. If the latest finish time is negative (i.e., precedes the project start time), the date in this field will be blank, and the time will be zero.
X	1-4095, or a date (DDMMMYY), or zeros, or blanks	Milestone Time or Date From characters 30-36 of the SSP type 5 input record. If the crash option has been selected, an attempt has been made to meet this milestone in an optimum fashion before trying to reduce the total project duration. If the crash option has not been selected,

Field	Contents	Description
		this figure has had no effect on the schedule. (See discussion of schedule optimization.)
Y, Z, AA	Same as fields F, G, and H	Activity Identification Numbers These fields contain the same information as the corresponding fields on the left side of the listing. They are repeated here to facilitate reference to the listing.
BB	Any siz alphanumeric characters	Cost Code or Part of Activity Description From characters 37-42 of the SSP type 5 input record. May be considered as part of the activity description.
CC	Any six alphanumeric characters	Responsibility Code or Part of Activity Description From characters 43-48 of the SSP type 5 input record. May be considered as part of the activity description.
DD	Any 32 alphanumeric characters	Activity Description From characters 49-80 of the SSP type 5 input record.
EE	Blank, MET, or NOT MET	Milestone Analysis Field If no milestone time or date was entered for the activity on the SSP type 5 input record, this field will contain blanks. If one was entered and the earliest finish time is on or before the milestone time, the word MET is printed in this field. If the earliest finish time is after the milestone time, the words NOT MET are printed in this field.
FF	0-99999999	Total Cost of Schedule Sum of the costs for the given schedule of all activities making up the network.

Error List

When DSRP detects an error, it enters an error subroutine which types out a corresponding error number. The entire set of error numbers is defined in the Project Management System Reference Manual.

PROGRAM ORGANIZATION DESCRIPTION

The DSRP is a self-contained program, coded in relocatable form in the SDS META-SYMBOL language. It contains its own loader (card or paper tape) and, therefore, does not require a MONARCH system tape. Extensive use has been made of the META-SYMBOL external reference provision for defining data storage to the various subroutines in the program. The majority of data used by the program is located in a subroutine called COMMON.

The program is composed of 28 subroutines connected via a main control program. This modular construction using subroutines provides a high degree of flexibility and facilitates ease of system maintenance. The program itself can determine the type of configuration on which it is operating. Array data offsets are based on the amount of storage available. Automatic checks are made to determine if the problem being run will fit into the computer. To handle as many network activities as possible, the program incorporates a self overlay feature; data is allowed to extend over the initial setup portion of the program during the detail schedule report generation phase.

DSRP Storage Arrangement

The storage arrangement for the 920 Computer version of the DSRP is shown in Figure 7. The basic program organization is divided into three primary sections which are loaded in relocatable form into lower memory starting at location zero. The first section is composed of those routines which must be in the computer throughout the entire execution of the program. These are routines 1 through 20 (see Subroutine Roster) and require approximately 2813 words of storage.

<table>
<tr><th>Approximate Location</th><th colspan="2">PHASE I (Program Initialization)</th><th>PHASE II (Report Generation)</th></tr>
<tr><td>Variable</td><td colspan="2"></td><td>MN Vector</td></tr>
<tr><td>10762_8</td><td colspan="2">28. Transfer Address
27. Parameter Print Routine
26. Holiday Sort Routine
25. Parameter Input Routine
24. I/O Device Select Routine
23. Magnetic Tape Setup Routine
22. Parameter Control Routine
21. Main Control – Part 2</td><td>AB Vector</td></tr>
<tr><td>07072_8</td><td rowspan="4">COMMON</td><td rowspan="2">Parameter Region</td><td>Time Vector</td></tr>
<tr><td>06252_8</td><td>Binary Holidays</td></tr>
<tr><td>05777_8</td><td colspan="2">Input/Output Area</td></tr>
<tr><td>05651_8
20.</td><td colspan="2">Fixed Common</td></tr>
<tr><td>05375_8</td><td colspan="3">19. Double-Precision Binary to BCD Conversion Routine
18. Decimal to Binary Conversion Routine
17. Binary to BCD Calendar Data Conversion Routine
16. Binary to Decimal Integer Conversion Routine
15. Report Line Output Routine
14. Magnetic Tape Read Routine
13. Magnetic Tape Write Routine
12. Paper Tape Read Routine
11. Card Read Routine
10. Error Routine
9. Print Routine
8. Detail Schedule Report Routine
7. Calendar Routine
6. Time Vector Routine
5. Input Conversion Routine
4. Duration Read Routine
3. End Routine
2. Delay Routine</td></tr>
<tr><td>00000</td><td colspan="3">1. Main Control – Part 1</td></tr>
</table>

Figure 7. DSRP Storage Arrangement

The Common routine represents the second major section of the program. Three subsections are contained within Common to provide storage area for the following groups of data:

1. Fixed Common – contains permanent data constants, scalar temporary, and short vector temporary. Approximately 108 words.
2. Input/Output Area – buffer regions for I/O storage. Approximately 482 words.
3. Parameter region – temporary locations for BCD parameter data, such as output comments, days-off, holidays, etc. This data is later condensed and saved in fixed Common. Approximately 570 words.

Common is followed by another section of program which can be destroyed by data after the setup phase (Phase I) of execution is completed. This section consists of routines 21 through 28 and occupies 950 locations of storage.

During the detail report generation phase of execution (Phase II), the following three large vectors are maintained in core storage:

1. MN Vector – contains the internal node numbers at the beginning and end of each activity.
2. AB Vector – contains the normal and crash durations of each activity.
3. Time Vector – contains the scheduled early start times of each node. (Also used alternately to store activity durations).

The MN and AB vectors are each as long as the total number of activities. The time vector is as long as the total number of nodes; but since it is also used to store the durations of each of the activities, it must be as long as the number of activities. Therefore, the total amount of storage needed for the three vectors is 3 times the number of activities. These vectors are allowed to overlap the destructible portion of the program code and also a portion of the parameter region. The remainder of the parameter region is used to store holiday dates in accumulative day binary form. An internal check assures that the activity vectors do not overlap the binary holidays.

Tape Layouts

The format of the input tape (tape 1) is presented in detail in the appendix. This tape contains three files of information:

1. Parameter File (BCD)
2. Unscheduled Activity File (BCD) (one record per activity)
3. Time File (Binary) (one record per schedule)

DSRP Execution Description

After the program has been loaded, control is transferred to Part 1 of the Main Control Routine (F.D.[†] 1), which immediately branches to Part 2 to initiate the program's setup.

The Magnetic Tape Setup Routine (F.D.23) is then entered. This routine rewinds tapes 1 and 2 and prepares tape 2 for writing by erasing approximately 15 inches of tape following the load point.

Main Control next transfers to the Parameter Control (F.D.22), at PARCON, which in turn transfers to the necessary subroutines to set up parameter data. The first subroutine transferred to by PARCON is the Parameter Input Routine (F.D.25) at entry point $SCAP3. The parameter file on tape 1, prepared by SSP, is read into core. At this point, all parameter data in the DSRP is identical to that used by the SSP. Control is now returned to PARCON. The I/O Device Selection Routine (F.D.24) is then entered to request the input and output configuration data from the typewriter. A provision is incorporated in the program to place the configuration description in the permanent data region, if desired. When this latter provision is used, no typewriter request is made.

PARCON again returns control to the Parameter Input Routine, this time at entry $SCAP5. This entry causes parameter change data to be read from cards or paper tape. The type designation of each input record is examined, and the data on that record is stored in its assigned locations in memory. Holidays are separated into day, month, year parts and stored in unsorted input order. If any changes to the holidays used by SSP are made, a complete set must be input. The storing of parameter data is continued until a type-8 input record is read. This record contains the first selected duration for which a detailed schedule is to be constructed. The type-8 record is left in the input area in card image form, and control is returned to PARCON.

The Holiday Sort Routine (F.D.26) is next entered to sort the holidays in order of increasing date. After this is accomplished, control is returned to PARCON which transfers to the Run Heading and Parameter Print Routine (F.D.27) to output the parameter data page. Upon return, PARCON transfers control to Main Control, Part 2.

A check is made in Main Control to insure that the last record read was a type-8 record. If it was not, an error halt occurs. If the last record read was type-8, the run ID is moved from the input parameter region into Common. Control is then transferred to the Input Conversion Routine (F.D.5). This routine examines the variable field type-8 input record to determine if the selected schedule has been requested using an end date, by duration in basic working units, or by schedule number. If an end date was used, it is separated into day, month, year parts, and stored in Common. If basic working units or schedule numbers are used, they are stored in the year location, and the day and month locations are set to zero. The routine then returns to Main Control.

[†] F.D. denotes flow diagram number.

Main Control next converts various input parameters from BCD to binary and stores them in Common. A first pass flag, FIRPAS, is turned on to indicate that the first selected duration has not yet been processed. The BCD days-off are converted and stored in the binary day-off vector locations, and the number of days-off per week and number of working days per week are computed.

A test is made to determine the maximum location in the core memory. Working from the end of core, the offset address for the MN, AB, and Time vectors are calculated and stored in Common. An index bit is placed in these offset address instructions so that individual elements can be obtained from the various vectors by means of indirect addressing.

Next a test determines if a start date is given. If so, control is transferred to the Calendar Routine (F.D.7) at its setup entry location $CALE1. This entry sets up all data needed to compute calendar dates from basic working units or basic working units from a calendar date. Data calculated during calendar setup include:

1. A vector containing the number of working days in each year (up to 15 years) from the start date of the project.
2. A binary holiday vector containing the holiday dates in terms of total accumulative days since the start of the project.
3. A vector containing the subscript of the first holiday in each year.
4. A vector containing the number of holidays in the next years (up to 15 years).
5. The day of the week on which the project starts.
6. The type year in which the project starts (leap year, year after leap year, etc.).

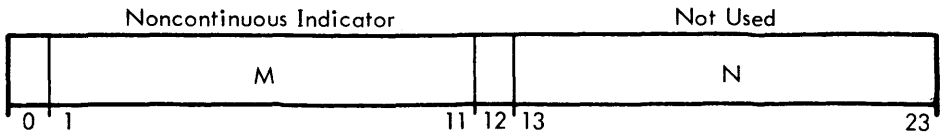
The precalculation of the above data makes possible the rapid calculation of dates and durations for calendar scheduling of the various detail schedules. After the completion of calendar setup, control returns to Main Control, Part 1.

At this point the storage area between the end of the binary holidays and the end of core storage is available for the MN, AB, and Time Vectors. It is of value to note that use of this entire area results in the destruction of a portion of the parameter region, as well as routines 22 through 28 (see Storage Arrangement). Although this does not affect the capability of the program to process the current desired set of detailed schedules, it does eliminate the ability to restart the problem without reloading the binary program deck. On the 920 Computer version, a complete restart can be made without reloading the program by a transfer to location zero, only if the total number of activities does not exceed the limits shown below.

Memory (words)	Activities
8, 192	1195
12, 288	2559
16, 384	3923

Main Control, Part 1 proceeds to generate the MN and AB vectors as follows: the input tape is rewound and positioned at the beginning of the unscheduled activity file; one activity record is read; and, the BCD M and N (internal node numbers) for that activity are converted to binary, merged into one word, and stored in the MN vector using the index loaded with the internal activity number, K. A similar sequence is used to store the crash and normal durations in the AB vector. The above vector element constructions are repeated for each activity in the unscheduled activity record file.

The sign of the MN vector entries are set minus if their associated activities are of the noncontinuous type. The resulting format of the MN entry is shown below.



Since the nodes are numbered continuously starting at 1, the N node of the last activity will be equal to the total number of nodes. Main Control, Part 1 picks up this node number and stores it in NE (number of events).

At this point the program enters the detail schedule generation loop. The general sequence of this loop is to read a selected schedule duration, develop an interpolated early node time vector, construct and print the detailed schedule, and repeat the loop until a type-9 input record is read.

The first routine entered in this loop is the Duration Read Routine (F.D.4). This routine reads the schedule duration or number select record from paper tape or cards, converts the variable field image using the Input Conversion Routine, and computes either the total number of working units in the project's total duration if an end date is given or the end date if total number of working units in the project are given. If no start date is given and project duration is specified in basic units, blanks are stored in the end date locations. Logic is provided to handle basic working units in hours, days, weeks, and months. Since the first project end date, duration, or schedule number will have already been read and converted prior to the first entry to this routine, these phases are bypassed when the FIRPAS indicator is on.

The next routine entered is the Time Vector Routine (F.D.6). At the time of entry to this routine, the input tape will be positioned just in front of the binary time vector file. This file contains the early times for each node at each break point in the Total Project Cost vs. Project Duration Curve. It is the function of the Time Vector Routine to generate a set of specific early node times for the selected project duration.

If the selected project duration is greater than the All Normal project duration, an error note is typed, and the program is allowed to continue, using the All Normal early node time vector. If the selected project duration is smaller than the All Crash project duration, an error note is typed, and the time vector for the completely crashed solution is used.

Normally, the selected duration falls within these limits. If the selected duration does not fall exactly on the duration time at one of the break points, it becomes necessary to interpolate for a time vector. This is accomplished by reading from tape the time vector above (toward normal) and below (toward crash) the selected project duration into the left and right halves of the time vector. The interpolated time vector is constructed by examining the above and below times at each node to determine if they are the same. If they are the same, this time is used for the current node in the interpolated time vector; if they are not, the node time for the lower vector is increased by an amount equal to the difference between the selected total project duration and the lower total project duration. This procedure is repeated for each node in the network. No interpolation is allowed if the schedule is selected by schedule number. The Duration Read Routine then returns control to Main Control.

After the time vector has been constructed, the program is ready to compute the various parameters needed to make up the detailed schedule report. The routine used for this function is the Detail Schedule Report Routine (F.D.8). This routine first constructs a vector in the left half of the time vector containing the durations of each activity. There are three possible forms these durations can take:

1. Normal Duration (B) – used when the time available between the activity's end nodes is greater than the normal duration.
2. Crash Duration (A) – used when an activity has been shortened as much as possible.
3. Semi-Crashed – used when the time available is greater than crash, but less than normal. In this case the duration is set equal to the time available.

If an activity is discontinuous, its duration is always set to the crash duration when the time available is less than the normal duration.

Using the vector of durations just developed, the routine next constructs a vector of late node times by working backward through the network, starting at the last node. This vector is stored over the AB vector, which will later be rebuilt.

When the early and late node times for the complete project have been computed and stored in memory, it is possible to construct the detailed schedule record for each activity by loading the unscheduled activity information for that activity alone into memory from tape. To initiate this operation, the input tape from SSP is rewound and positioned in front of the unscheduled activity file.

Next, the DSRP Print Routine (F.D.9) is entered at $SRPINIT. This entry initializes the line and page counters and returns.

The double-precision total cost accumulator is cleared, and the first unscheduled activity record is read from tape in the BCD mode. The type code for the unscheduled activity records is type 5. If the code read is not a type-5 code, it is assumed that all activities have been processed. The input tape is rewound in preparation for processing the next detail schedule, total cost is output using the DSRP Print Routine at entry $TOTCPR, and control is returned to Main Control, Part I.

If the record code is a type-5, an unscheduled activity record has just been brought into storage. The record is now ready for scheduling. A test determines if the activity is an internal or milestone dummy. If it is not, the routine proceeds to convert normal and crash cost and durations to binary. The internal activity number, K, is loaded into the index register and used to rebuild the AB vector entry for the current activity.

Free float and activity duration are next computed and stored ready for output. Late start and early finish in basic units are computed and stored ready for output along with early start and late finish in basic units. Total float is calculated and checked for criticality. If found critical, an indicator is stored ready for output. The cost of the activity is next computed and stored. This cost is added in double precision to the total cost of the project.

If a start date is given, early start, late start, early finish, and late finish are converted to calendar dates and also stored ready for output. Hour-, day-, week-, and month-type working units are considered in this conversion. If no start dates are given, the calendar dates for these four times are set to blanks.

A check determines if the end of this activity has a milestone date associated with it. If so, another check determines whether or not the milestone is met, and an indicator is set accordingly. If the milestone is given in basic units and a start date has been provided, the milestone time will be converted to a calendar date.

Before the output routine for the shceduled current activity is entered, two checks are made:

1. Should only major activities be output?
2. If so, is this a major activity?

Based on the results of this test, an activity is either output or not output. The entry to the DSRP Print Routine to produce the schedule activity record is $SOUTSR.

The program now reads the next activity and again checks for the 5 code as previously discussed.

After all activities have been processed for the current selected date, control is returned to Main Control, Part I, and a new selected schedule duration is brought in through the Duration Read Routine. When a type-9 record is read by Duration Read, an end statement is typed and the program halts.

3. PROGRESS EVALUATION PROGRAM

This section describes the SDS Progress Evaluation Program (PEP), the third program in the SDS Project Management System Series. To provide management with current project status information, this program should be run cyclically for the duration of a project.

PROGRAM FUNCTION

The Progress Evaluation Program (PEP) provides management with an effective means for determining the current status of a project at various times during its execution. The program points out those activities that are ahead or behind schedule, on or off the critical path, and how these activities affect the total status of the project. Also, summary information defining the rescheduling history of these activities is reported.

Other activity information provided includes activity descriptions, cost codes, and responsibility codes. The status of the entire project is reported in addition to the status of the individual activities. Such questions as: "When is the project scheduled to finish?", "When is the project expected to finish?", and "How much is it ahead or behind schedule?" are answered in the Project Status Evaluation Report generated by PEP.

PEP maintains an activity status file tape throughout the duration of the project. Initial information for this file is obtained from the output tape of the Detail Schedule Report Program (DSRP). With each progress evaluation run, the activity status file is updated to reflect the most current reporting information from the various activity managers. This updated tape is then used as input for the next progress evaluation run. The above procedure is repeated, cyclically, until the project is completed.

Provisions are incorporated in the program to accept two types of status changes: activity changes and network changes. An activity change refers to all data changes pertaining to individual activities, such as scheduled start and finish dates, actual start and finish dates, scheduled duration, etc. Network changes refer to the addition, deletion, and replacement of activities to the project network.

The assignment of the original activity scheduled start and finish dates is made automatically by PEP. These dates are obtained from the early start and early finish activity information on the DSRP output tape. The user has the option of overriding any of these automatic initial assignments during the various progress runs.

Since the DSRP output tape can contain more than one project schedule, the user must specify which schedule on the tape he desires to select for the project. This specification is accomplished by a type-in. Two methods are used to specify the selected schedule from the DSRP output tape.

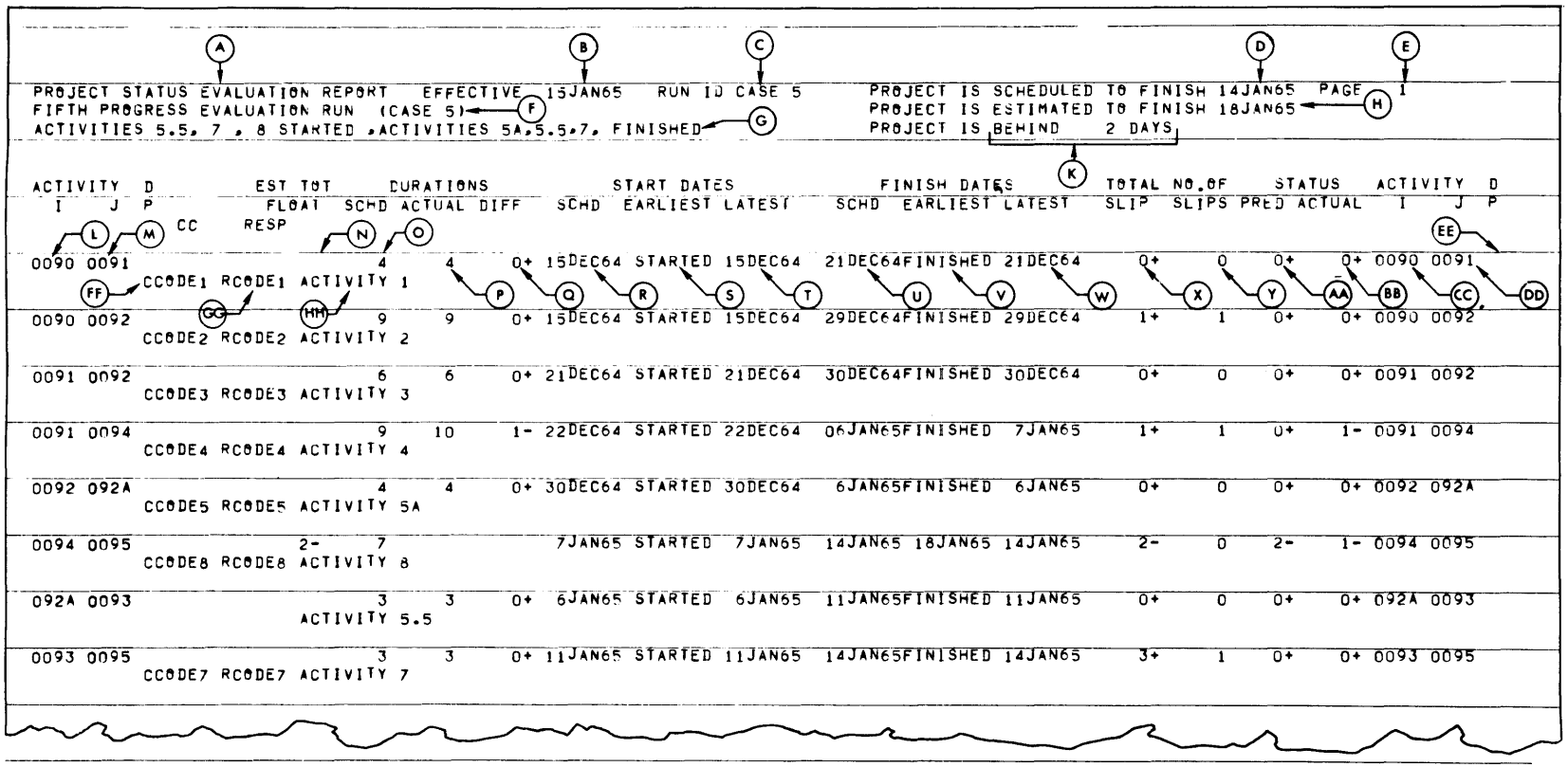
One method defines the selected project schedule by completion date and the other by schedule number. Completion date is used to specify the schedule for PEP when the schedules from DSRP have been requested by an end date or duration. A schedule number is used when the schedules have been requested from DSRP by schedule number. No schedule interpolations are made by PEP. Only those schedules on the DSRP output tape are available for assignment as the selected project shedule.

The Detailed Schedule Report used for initial PEP input must be generated with calendar dating. The basic units used by DSRP in developing this input schedule can be in terms of hours, days, weeks, or months.

PEP uses two schedules to evaluate the project's status. One schedule is constructed internally by PEP, based on the effective date of the progress evaluation run, actual reported start and finish information, and scheduled activity durations. This schedule is called the expected schedule and is compared against the "scheduled" schedule maintained on the activity status file. From this comparison the various activity status parameters are computed.

PEP USAGE

The minimum configuration used by PEP consists of 8192 words of core, two magnetic tape units, paper tape and typewriter input, and an off-line printer. Its operational capabilities automatically expand as the configuration is increased. Provisions are made for operating in conjunction with card reader input and on-line printer output. The number of activities processed by the program can be increased by adding more core storage to the configuration. The maximum number of activities (NA) handled by the program is a function of the total amount of storage in the computer and the ratio of activities to node points (R_{AE}). The equation for NA is given below.



ACTIVITY I	ACTIVITY J	P	CC	RESP	EST TOT FLOAT	DURATIONS SCHD	DURATIONS ACTUAL	DURATIONS DIFF	START DATES SCHD	START DATES EARLIEST	START DATES LATEST	FINISH DATES SCHD	FINISH DATES EARLIEST	FINISH DATES LATEST	TOTAL SLIP	NO. OF SLIPS	STATUS PRED	STATUS ACTUAL	ACTIVITY I	ACTIVITY J	P
0090	0091					4	4	0+	15DEC64	STARTED	15DEC64	21DEC64	FINISHED	21DEC64	0+	0	0+	0+	0090	0091	
			CODE1	RCODE1	ACTIVITY 1																
0090	0092					9	9	0+	15DEC64	STARTED	15DEC64	29DEC64	FINISHED	29DEC64	1+	1	0+	0+	0090	0092	
			CODE2	RCODE2	ACTIVITY 2																
0091	0092					6	6	0+	21DEC64	STARTED	21DEC64	30DEC64	FINISHED	30DEC64	0+	0	0+	0+	0091	0092	
			CODE3	RCODE3	ACTIVITY 3																
0091	0094					9	10	1-	22DEC64	STARTED	22DEC64	06JAN65	FINISHED	7JAN65	1+	1	0+	1-	0091	0094	
			CODE4	RCODE4	ACTIVITY 4																
0092	092A					4	4	0+	30DEC64	STARTED	30DEC64	6JAN65	FINISHED	6JAN65	0+	0	0+	0+	0092	092A	
			CODE5	RCODE5	ACTIVITY 5A																
0094	0095					2- 7			7JAN65	STARTED	7JAN65	14JAN65	18JAN65	14JAN65	2-	0	2-	1-	0094	0095	
			CODE8	RCODE8	ACTIVITY 8																
092A	0093					3	3	0+	6JAN65	STARTED	6JAN65	11JAN65	FINISHED	11JAN65	0+	0	0+	0+	092A	0093	
					ACTIVITY 5.5																
0093	0095					3	3	0+	11JAN65	STARTED	11JAN65	14JAN65	FINISHED	14JAN65	3+	1	0+	0+	0093	0095	
			CODE7	RCODE7	ACTIVITY 7																

Figure 8. Project Status Evaluation Report

$$NA = \frac{\text{Total Storage} - 4150}{\left(2 + \frac{1}{R_{AE}}\right)}$$

$$R_{AE} = \frac{\text{Number of Activites}}{\text{Number of Nodes}}$$

For a value of R_{AE} equal to 1.8, the maximum number of activities handled by the program on an SDS 920 Computer are

Total Core Storage	Number of Activity
8,192	1585
12,288	3200
16,384	4095

These figures are slightly less for SDS 910 Computer operation due to the use of programmed operators. A maximum limit of 2047 event nodes can be handled by any of the above configurations, independent of the number of activities.

Loading Procedures

The procedures for running PEP and instructions for filling out data forms for PEP are contained the the Project Management System Reference Manual (publication number 90 09 18).

PEP Output

Output is always provided in off-line format on tape 1, regardless of whether or not an on-line printer is attached to the system. Figure 8 shows a sample PEP printout. The letters below in the "Field" column refer to the sample printout.

Field	Contents	Description
A	Project Status Evaluation Report	Report Heading
B	Date: DDMMMYY	Effective Date of Run From characters 12 through 18 of the type-C record. This is the date as of which the input information is to be considered current. It is used in the calculation of the predicted and actual status of each activity.
C	Any alphanumeric characters or date	Run Identification From characters 4 through 11 of the type-C input record. This code is printed at the top of each page for identification of the run.
D	Date: DDMMMYY	Scheduled Project Finish Date Date on which the last activity of the network is currently scheduled to finish.
E	1-9999	Page Number A number assigned sequentially to each page of the report.
F	Any 71 alphanumeric characters	First Line of Project Heading From characters 2 through 72 of the type-A input record. Project identification information.
G	Any 71 alphanumeric characters	Second Line of Project Heading From characters 2 through 72 of the type-B input record. More project identification information.

Field	Contents	Description
H	Date: DDMMMYY	Estimated Project Finish Date Current value of the finish date of the last activity of the network.
K	AHEAD 0-4095 HOURS DAYS WEEKS MONTHS or BEHIND 1-4095 HOURS DAYS WEEKS MONTHS	Project Status Difference between the scheduled project finish date and the estimated project finish date in the type of units specified on the type-2 input record to SSP.
L	NAAA: N = Numeric AAA = Three alphanumeric characters	I Identification code of starting node of the activity.
M	NAAA: N = Numeric AAA = Three alphanumeric characters	J Identification code of ending node of the activity.
N	-4095 to +4095 or blank	Total Float The amount of time by which the actual completion of the activity can be delyed after its earliest finish time before it causes a delay in the scheduled project finish date. It is calculated as the difference between the latest finish date and the earliest finish date, in the units specified on the type-2 input record to SSP. If the latest finish date precedes the earliest finish date, this field will be negative. If the activity has been reported as finished, this field will contain a blank. A negative total float indicates that the activity is on a path that will prevent the scheduled project finish date from being achieved.
O	0-4095	Scheduled Activity Duration The duration for the given activity used in the determination of early and late start and finish dates and of total project duration. This amount is in the units specified on the type-2 input record to SSP. On the initial PEP run for the project, this field is initialized as the duration used for the given activity in the schedule that was selected from the DSRP output tape. It can be changed on any PEP run using characters 26 through 29 of the type E input record.
P	0-4095	Actual Activity Duration The difference between the activity's actual finish date and its actual start date, as specified on the type-E input records to PEP. The units are the same as those specified on the type-2 input record to SSP.
Q	0-4095	Duration Difference The difference between the activity's scheduled duration and its actual duration.

Field	Contents	Description
R	Date DDMMMYY	Schedule Start Date The date on which the activity is scheduled to start. On the initial PEP run, this field is initialized as the earliest start date from the selected schedule of the DSRP output, unless changed by input of a type-E record. It can be changed on any PEP run. Note that this field is not automatically changed as a result of duration changes, actual start and finish reports, etc. It must be changed specifically, using characters 12 through 18 of the type-E input record. Actual start dates are compared with this field to determine by how many units the activity was actually started ahead of or behind schedule.
S	Date (DDMMMYY) or STARTED	Earliest Start Date If an actual start date has been reported for an activity on either this run or on a previous PEP run (on a type-E PEP input record), The word STARTED is printed in this field. If not, this is the earliest date on which the activity can start, based on its position in the network and on the effective date of the run. It is the date corresponding to the time which is determined as follows: The sum of the durations (actual for completed activities and scheduled for noncompleted activities) along the longest path to each node is calculated. That sum is assigned as the time for that node, unless it corresponds to a date prior to the effective date of the run, in which case the time corresponding to the effective date is assigned as the time for the node. This means that no earliest start dates can precede the effective date of the run. This modified node time assignment is also used during the calculation of the longest paths to all succeeding nodes. The earliest start date may not be modified directly with input data. It is recomputed during every PEP run.
T	Date: DDMMMYY	Latest Start Date If an actual start date has been reported for the activity on either this run or a previous PEP run, that start date is printed in this field. If not, this is the date by which the activity should start in order to be finished by its latest finish date. It is calculated by subtracting the activity's scheduled duration from its latest finish date. The latest start date may not be modified directly with input data. It is recomputed during every PEP run. If the latest start time is negative (i.e., precedes the project start time), this field will be blank.
U	Date: DDMMMYY	Scheduled Finish Date The date on which the activity is scheduled to finish. On the initial PEP run, this field is initialized as the earliest finish date for the activity from the selected schedule of the DSRP output. After that, it can be changed on any PEP run. Note that this field is not automatically changed as a result of duration changes, actual start and finish reports, etc. It must be changed specifically using characters 19 through 25 of the type-E input record. Predicted finish dates (earliest finish) are compared with this field to determine by how many units the activity was actually finished ahead of or behind schedule.

Field	Contents	Description
V	Date (DDMMMYY) or FINISHED	Earliest Finish Date If an actual finish date has been reported for an activity on either this run or a previous PEP run (on a type-E PEP input record), the word FINISHED is printed in this field. If not, this is the date on which a given activity should be finished if it is started at its earliest start time. It is the date corresponding to the time that is calculated by adding the activity's scheduled duration to its earliest start time. The earliest finish date may not be modified directly with input data. It is recomputed during every PEP run.
W	Date: DDMMMYY	Latest Finish Date If an actual finish date has been reported for the activity on either this run or a previous PEP run, that finish date is printed in this field. If not, this is the date by which the activity should be finished in order to avoid exceeding the scheduled completion date of the project. It is the date that corresponds to the difference between the project scheduled completion time and the sum of the scheduled durations of all activities along the longest path from the finish of the given activity to the project scheduled completion time. If the latest finish time is negative (i.e., precedes the project start time), this field will be blank. If the latest finish date precedes the earliest finish date, a negative total float will result.
X	0-9999 (plus or minus)	Total Slip The accumulated amount of change in the scheduled finish date since the first PEP run. Each time an activity's scheduled finish date is changed (using characters 19 through 25 of the type-E input record), the difference between the old scheduled finish time and the new scheduled finish time is added to the total slip accumulated for that activity since the initial PEP run for the project. This new total slip is written onto the activity status output file on magnetic tape and is the input to the next PEP run. Changes to scheduled finish dates introduced on the initial PEP run for the project do not increase the total slip. Note that the total slip accumulation is reduced if the new scheduled finish date is earlier than the old scheduled finish date.
Y	0-9999 (always positive)	Number of Slips The accumulated number of extensions of the scheduled finish date since the first PEP run. Each time a scheduled finish date is changed to a later date (using characters 19 through 25 of the type-E input record), the number of slips accumulated for this activity since the initial PEP run is increased by 1. The number of slips is written onto the activity status output file on magnetic tape and is input to the next PEP run. Changes to scheduled finish dates introduced on the initial PEP run for the project do not increase the number of slips. If the scheduled finish date is changed to an earlier date, no change is made in the number of slips.
AA	0-4095 (plus or minus)	Predicted Status This number is a prediction of how far ahead of or behind schedule an unfinished activity will finish. For a finished activity, it will be zero. For an unfinished activity, it will be the difference between the scheduled finish date and the earliest finish

Field	Contents	Description
AA (cont.)		date. A plus sign indicates a prediction that it will finish ahead of schedule. A minus sign indicates a prediction that it will finish behind schedule.
BB	0-4095 (plus or minus)	Actual Status This number shows how far ahead of or behind schedule an activity was reported to have actually started or finished. It is set to zero when the activity is 1. started or finished on schedule. 2. not started and not scheduled to have started. 3. started, not finished, and not scheduled to have finished. If the activity has not started, but is scheduled to have started, its actual status is set to the difference between its scheduled start time and the effective time of the current run. If the activity has not finished, but is scheduled to have finished, the actual status is set to the difference between the scheduled finish time and the effective time of the current run. If the activity has finished, the actual status is set to the difference between the scheduled finish time and the actual finish time. In any of the above cases, a plus sign indicates that the activity is on or ahead of schedule. A minus sign indicates that it is behind schedule.
CC, DD, EE	Same as fields L, M, and N	Activity Identification Numbers These fields contain the same information as the corresponding fields on the left side of the listing. They are repeated here to facilitate reference to the listing.
FF	Any six alphanumeric characters	Cost Code or Part of Activity Description From characters 37 through 42 of the SSP type-5 input record, or from characters 44 through 49 of the PEP type-E input record. Can be used as a sort key for PEPSORT.
GG	Any six alphanumeric characters	Responsibility Code or Part of Activity Description From characters 43 through 48 of the SSP type-5 input record, or from characters 50 through 55 of the PEP type-E input record. Can be used as a sort key for PEPSORT.
HH	Any 32 alphanumeric characters	Activity Description From characters 49 through 80 of the SSP type-5 input record, or from characters 14 through 45 of the PEP type-F input record.

The updated activity status file is written on tape 2 in BCD. A detailed layout of this tape is in the appendix. This tape is used as input to PEP on the next progress evaluation, at which time it is mounted on unit 1. Tape 2 also provides the input data to the PEPSORT program. For input to PEPSORT this tape is mounted on unit 2.

Error List

Upon detection of an error, PEP enters an error subroutine that types an error number. The meaning of these error numbers is defined in the Project Management System Reference Manual.

PROGRAM ORGANIZATION DESCRIPTION

PEP is a self-contained program, coded in relocatable form in SDS META-SYMBOL language. It contains its own loader (card or paper tape) and, therefore, does not require a MONARCH system tape. Extensive usage has been made of the META-SYMBOL external reference provision for defining data storage to the various subroutines in the program. The majority of data used by the program is located in a subroutine called Common.

The program is constructed in two segments. A mutual Common area is shared by these two segments. Furthermore, each segment has an exclusive storage area that is not shared. Segment I contains the initial activity status file tape setup, parameter input processing, and network construction logic. It is made up of 32 subroutines. Segment II is made up of 30 subroutines and contains the necessary logic to accomplish activity posting, status evaluation, and progress evaluation report output.

The program determines the type of configuration on which it is being run and computes array data offsets based on the amount of storage available. Automatic checks are made to determine if the problem being run will fit into the computer storage available.

PEP Storage Arrangement

The storage arrangement for the SDS 920 Computer version of PEP is shown in Figure 9. The S or F in front of the various subroutine numbers denotes first (F) or second (S) segment.

Segment II is loaded into the computer first and written on tape 2 minus its Common (S01), Magnetic Tape Read (S02), and Segment II Load (S03) routines. At load time the relocatable binary addresses are converted to absolute addresses. All external references are also assigned absolute binary addresses. Note that although subroutines S01, S02, and S03 are not written on tape 2 with the remainder of the routines in segment II, all external references between the two groups of routines are already set up.

After segment II has been written on tape 2, segment I is loaded into the computer by the loader, which is in high core memory. Segment I is loaded at the same location as segment II and contains identical copies of subroutines S01, S02, and S03 (see Figure 9). An interface between segments I and II has now been set up, since all external references by segment II to routines S01, S02, and S03 are identical to external references to F01, F02, and F03.

Control is next transferred to Segment I Control Control (F04). After the execution of segment I, the Segment II Load Routine (F03) is entered. This routine loads segment II from tape 2 using the Magnetic Tape Read Routine (F02). Segment II is loaded just above F03 and, therefore, does not interfere with the loading routine's operation. After segment II has been loaded, transfer is made to Segment II Control (S04) and the remainder of the program is executed.

A number of data overlaps are made during various phases of both segments I and II. The data storage arrangements during these various phases are discussed under "PEP Execution Description."

Tape Layouts

The layout of the input file to PEP on the initial progress run is identical to the layout of the Detailed Scheduled Report written off-line by DSRP. On subsequent runs, the activity status file is used as input. This file is written on tape 2 by PEP during the previous progress run. Also on this tape is the project parameter file which contains total project data, such as start date, units type, days-off, holidays, etc. The parameter file precedes the activity status file.

Detailed layouts of both the parameter file and the activity status file are in the appendix.

PEP Execution Description

After segment II has been loaded on tape 2 and segment I into core memory, the loading program transfers control to Segment I Control (F. D. F04).

Segment I Control enters the I/O Device Selection Routine (F. D. F06) at entry $SCAP2. This routine requests the input and output configuration description from the typewriter. The program can place the configuration description in the permanent data region, if desired. When this provision is used, no typewriter request will be made.

Control is next transferred to the Segment I Setup Routin (F07), which determines the extent of core and calculates the vector offsets used by segment I. These vectors are illustrated at the top of the PEP storage map shown in Figure 9. During the different phases of segment I execution, the available storage area is assigned for various vector usage.

There are basically three major phases to segment I:

1. A parameter file preparation phase
2. An internal topological sort phase
3. A network update and tape sort phase

<table>
<tr><th rowspan="2">Approximate Location</th><th colspan="3">SEGMENT I</th><th colspan="3">SEGMENT II</th></tr>
<tr><th>Parameter Input Phase</th><th>Topological Sort Phase</th><th>Tape Sort Phase</th><th>Activity Data Post Phase</th><th>Status Evaluation Phase</th><th>Intermediate Buffer Phase</th></tr>
<tr><td>Variable</td><td rowspan="2"></td><td rowspan="2">J Abbreviated Activity Record</td><td>Sorted Record Area</td><td rowspan="3">Card Storage Buffer Area</td><td rowspan="2">MN Vector</td><td rowspan="2"></td></tr>
<tr><td>Sequence</td></tr>
<tr><td></td><td></td><td>Selected</td><td rowspan="2">Durtas Vector</td><td rowspan="3">Parameter Region
10027_8</td></tr>
<tr><td></td><td>I Abbreviated Activity Record</td><td>New MN Vector</td></tr>
<tr><td></td><td>Parameter Region</td><td>J Abbreviated Activity Record</td><td>Successor Vector</td><td>Time Vector</td></tr>
<tr><td>06000_8</td><td colspan="3"></td><td colspan="3">Binary Holidays
7560_8</td></tr>
<tr><td></td><td colspan="3">F32. Segment I Data Routine
F31. Decimal to Binary Conv.
F30. Date Comparison Routine
F29. Utility Package
F28. Error Routine
F27. Paper Tape Routine
F26. Card Read Routine
F25. Magnetic Tape Setup Routine
F24. Binary to Decimal Integer Conv.
F23. Magnetic Tape Write Routine
F22. Schedule Request Routine
F21. Read BCD Record Routine
F20. Tape Sort Routine
F19. M and N Assignment Routine
F18. Topological Sort Routine
F17. (Not used)
F16. Unsorted Activity Assign.
F15. Topological Sequence Control
F14. Blanks to Zero Routine
F13. Network Change Routine
F12. Initial Abbreviated Activity File
F11. Holiday Sort Routine
F10. Parameter and Holiday Input
F09. Initial Input Tape Routine
F08. Erase Routine
F07. Segment I Setup Routine
F06. I/O Device Selection Routine
F05. First Run Request Routine
F04. Segment I Control</td><td colspan="3">S30. Segment II Preparation Routine
S29. Binary to Decimal Integer Converion Routine
S28. Double-Precision Binary to BCD Conv.
S27. Decimal to Binary Conversion Routine
S26. Date Conversion Routine
S25. Utility Package
S24. Error Routine
S23. End Routine
S22. Paper Tape Read Routine
S21. Card Read Routine
S20. Magnetic Tape Write Routine
S19. Magnetic Tape Setup Routine
S18. Date to Basic Units Routine
S17. Calendar Routine
S16. Report Line Output Routine
S15. Print Evaluation Routine
S14. Print Initialization Routine
S13. Print Control Routine
S12. Status Evaluation Routine
S11. Reverse PERT Routine
S10. Forward PERT Routine
S09. Segment II Setup Routine
S08. Blanks to Zero Routine
S07. Post Routine
S06. Activity Change Routine
S05. Erase Routine
S04. Segment II Control</td></tr>
<tr><td>01101_8</td><td colspan="6"></td></tr>
<tr><td>00654_8</td><td colspan="6">F03 and S03. Segment II Load Routine
F02 and S02. Magnetic Tape Read Routine</td></tr>
<tr><td>00000</td><td>F01 and S01 Common Storage</td><td colspan="5">Input/Output Area
Fixed Common</td></tr>
</table>

Figure 9. PEP Storage Arrangement

During the initial progress evaluation run, there is an additional phase for preparation of the initial activity status file from the DSRP output tape.

During the topological sort phase, three large vectors are maintained in memory to define the project's network structure. These vectors are called the abbreviated activity records. The BCD I, J, and Duplication Codes are stored in these vectors, along with other brief indicator data. Each of these vectors occupies a third of the storage space available.

After completion of the topological sort, a vector (called the successor vector) is stored over the miscellaneous abbreviated activity record (Dublication Codes). A vector (called the new MN vector) is set up over the I abbreviated activity record. The J abbreviated activity record space is allocated to the storage of three groups of data. The areas for the storage of these groups are called the selected area, the sequenced area, and the record storage area. It is the function of the Segment I Setup Routine to set up addresses referring to the above data.

After the Segment I Setup Routine has been executed, tape 2 is position after its first record, which contains segment II. Subroutine First Run Request (F05) is entered at FRSTRUN to request (from the typewriter) it this is an initial progress evaluation run. If a yes (Y) answer is received, subroutine Initial Input Tape (F09) is entered to prepare the initial activity status and parameter files from data on the DSRP output tape. If a no (N) answer is given by the user, this subroutine is skipped.

Initial Input Tape reads parameter and holiday information from the DSRP tape and constructs a parameter file on tape 2, just behind segment II. Next, the Schedule Request Routine (F22) is entred to request the end date or number of the schedule from the DSRP tape to be used as the project schedule. After receiving an answer from the typewriter, Initial Input Tape searches the tape for the requested schedule.

When the requested schedule is found, it is read into memory, one activity at a time. Early start and finish dates are assigned as schedule start and finish dates for the current activity. The schedule duration for the activity is assigned the value for duration contained on DSRP output tape. This data is then written on tape 2 behind the parameter file as the activity status record for the current activity.

The above procedure is repeated for each activity in the project. The first word of each activity status record contains a type 2 identification code. A blank activity status record, containing a type-9 code, is written on tape after the last activity has been processed.

Next, a statement is typed to the user stating that if tape 1 is to be saved, it should be replaced with an available tape. The program halts after this request and waits for a run command. After returning to run status, tape 1 is rewound and the parameter and activity status files are copied onto it from tape 2; at this point the status of tapes 1 and 2 is identical to what it would be if this were a subsequent progress evaluation run. Initial Input Tape returns control to Segment I Control.

The program now executes the various routines required to update the project's network structure. The first of these routines is the Initial Abbreviated Activity File Routine (F12); it inserts the BCD I, J, and Dupe Code activity designations from the previous run's network configuration into the abbreviated activiy records (see Figure 9).

The next routine entered is the Network Change Routine (F13), which reads the type D network change records into memory from cards or paper tape. Each is tested to determine if it contains an addition (new), deletion, or replacement to the network. The I, J, and Dupe Codes of new activities are placed at the end of the abbreviated activity records. Deletions are flagged, and replacements inserted over the activities to be replaced. A count of the number of activities in the new configuration is generated during this posting loop.

When a non-network change input record is encountered, tape 1 is rewound and positioned at the beginning of the activity status file from the previous progress evaluation run. Tape 2 is already positioned for writing the activity status file for the new network configuration in unsorted order. The following general procedure is used in generating the new file on tape 2.

An activity status record is brought into memory from tape 1 and checked against its corresponding abbreviated activity entry to determine if it is to be replaced or deleted. If it is to be replaced, the I, J, Dupe Code from the abbreviated activity record is moved into the activity status record over the old I, J, Dupe Code, and the new record is written on tape 2. If the activity has been deleted, it is not written on tape 2.

After all retained and replaced activities have been written on tape 2, the new activities are posted. The activity status records for these new activities are set to blanks, except for the I, J, Dupe Code and activity number entries.

Data for these new activities is supplied during the activity change phase of segment II. A 9-code record is written after the last new activity record on tape 2 to flag the end of the activity status file.

A test is made to determine if the current run is an initial progress evaluation run. If it is not an initial run, the comment "SAVE TAPE 1 IF TAPE SAVE REQUIRED" is typed. The program halts and waits for a continue command. When this occurs, control is returned to Segment I Control.

The next routine entered is the Topological Sequence Control Routine (F15) which sets up addresses for the various vectors needed during the topological sort and tape sort phases. The Topological Sequence Control routine first sets up addresses of various data arrays, to take advantage of the amount of core storage available. This routine also computes the number of records that can be handled in the available core storage during each pass of the unsorted activity file. Topological Sequence Control next executes a number of subroutines. After all these subroutines are executed, the result is a magnetic tape file of activity records in a sequence such that no activity precedes any of its predecessors. This tape file contains internally assigned node numbers, M and N, and a new activity number K. All three of these numbers are assigned in an ascending sequence, based on the topologically ordered activity records.

The first of the sequencing routines entered is Unsorted Activity Number Assignment Routine (F16). This routine stores a record number in the first 12 bits of each entry in the successor vector. This number is assigned in ascending order, based on the orignal input sequence of the original activities and network changes.

The Topological Sort Routine (F18) is executed to assign each of the activities to a position in a thread such that every item on the thread follows all its predecessor activities. This is accomplished by creating an arbitrary initial thread, then moving along this thread, examining the I and J of each activity to find if the activity has successor activities that are assigned to the thread ahead of the given activitiy. When one is found, it is moved to the end of the thread, from which position it will later be examined to determine if it still follows all its predecessors. If it does not, it is moved to the end again. This process continues until the last activity of the thread has been examined without finding any succeeding activities ahead of it. Activities are not physically moved from their I, J, Duplication Code sequence during this process. Instead, the last 12 bits of each entry in the successor vector is set to contain the number of the activity immediately succeeding it in the thread.

Once the topological thread has been determined, it is possible to assign new identification numbers M and N to the nodes based on their order of appearance within the thread. This is accomplished by the execution of the M and N Assignment Routine (F19). It should be noted that all starting nodes (nodes with no activities leading into them) are assigned the number 1, and all ending nodes (nodes with no activities leading out of them) are assigned the highest N.

The next task is to sort the activity status records on tape into the newly determined topological sequence. Since the program must operate on a minimum configuration having only two tape units, the sort is required to make multiple passes through the unsorted activity status file, obtaining as many records as possible (n records) each pass, and writing them on the sorted activity status file tape. This is accomplished by extracting the next n items from the successor vector (as defined by the thread) and moving them into the selected area. The number of the activity in the sorted sequence is now inserted in the last 12 bits of the extracted words. The first 12 bits of these words retain the number of the activity in the original sequence. These extracted words are now sorted on the original sequence number and stored into the sequenced area shown in Figure 9. They are now in the same sequence as the records in the unsorted activity status file.

Since these records are not contiguous, a complete pass through the unsorted file is required to read each group of n activity status records. As the records are read, their input locations are calculated based on their new sequence number; thus, after each pass storage will contain the next n records in the assigned topological sequence. They are now written on the sorted activity status file tape. As many passes are made through the unsorted file as are required to place all the records in their new sequence in the sorted file.

Control is returned to Segment I Control which rewinds tape 2 and transfers to the Segment II Load Routine. This routine brings segment II into memory and transfers to the Segment II Control Routine (S04).

The first subroutine executed by segment II is the Activity Change Routine (S06). This routine updates the data contained in the activity status file with current reporting information entered on E- and F-type cards or paper tape records.

The method used for posting this information is

1. Tape 1 is assigned as the input tape, and tape 2 is assigned as the output tape.
2. The parameter file is read into core from the input tape and written on the output tape.

3. E and F records are read from cards or paper tape until the available buffer area for this data has been filled.
4. An activity status record is read into memory from the input tape.
5. The Post Change Routine (S07) is entered. This subroutine posts all E and F record information, contained in the buffer area, belonging to the activity status record currently in memory.
6. Items 4 and 5 are repeated until all activities on the input tape have been updated by the E- and F-type records in the buffer.
7. If there are more E and F records to be posted, they are stored in the buffer region until it is again filled.
8. The input tape is assigned as the output tape, and the output tape assigned as the the input tape.
9. Steps 1 through 6 are repeated.

The above sequence is repeated until all activity data is posted. A check is made to determine if the last pass through the posting loop left the updated activity status file on tape 1. If so, the routine returns to Segment I Control. If not, tape 2 is copied cnto tape 1 before the routine returns.

Segment II Setup Routine (S09) is entered next in order to set up various data used by segment II. The project's parameter file is brought into core from tape 1. Data contained in this file is converted from BCD to binary and moved to various working storage areas. Vector offsets used by segment II are computed and extent of storage checks are made. The Calendar Routine (S17) is entered at its initialization entry $CALEI. Upon return from the Calendar initialization, the effective date is converted to basic units and the routine returns to Segment II Control.

Tape 1 is now positioned at the beginning of the activity status file in preparation for the generation of the early start times by the Forward PERT Routine (S10). This routine constructs the vector of expected early start times based on current activity status, as defined by actual start and finish information, future scheduled activity durations, and the effective date of the current progress evaluation run.

Three vectors are generated by the Forward PERT Routine. These vectors are

1. MN vector – contains the node number at the beginning and end of each activity.
2. Durtas vector – contains information needed to compute late node times. If an activity is already started, its entry in this vector will be its actual start time in basic units. If the activity is not started, its scheduled duration is entered. When a duration is entered, a bit is merged into position zero.
3. Time vector – contains early node times in basic units in right 12 bits of word.

The activity status file is processed, one record at a time, in constructing the above records.

The Reverse PERT Routine (S11) is entered next. This routine works backward through the network to determine the latest time at which the various nodes can occur, without affecting the finish time of the total project. These late finish times are stored in the left 12 bits of the various elements of the time vector.

It is now possible to evaluate the current status of each activity in the project by reading the activity status record for one activity at a time from tape 1. The updated activity status records can then be written on tape 2. In preparation for this updating, Segment II Control transfers to the Magnetic Tape Setup Routine (S19) which rewinds tapes 1 and 2, and prepares tape 2 for writing.

Control is next transferred to the Status Evaluation Routine (S12). This routine reads one activity status record at a time from tape 1. The status of this activity is then computed by processing the various vectors, previously mentioned against the schedule information contained in the record. An updated record for the activity is then written on tape 2. The above is repeated for each activity in the network. After all activities have been evaluated, the routine computes an evaluation of the total project and stores parameters describing its status in Common. A return to Segment II Control is then executed.

Using the Utility Package (S25) at entry SETT1, tapes 1 and 2 are once again rewound, but this time tape 1 is prepared for writing. Control is then transferred to the Print Control Routine (S13). Print Control enters the Print Initialization Routine (S14) at entry PEPINT. This routine initializes the page and line counters and returns.

The first activity status record is brought into memory from tape 2. Checks are made to determine if "major activity" and/or "future activity" only printouts are desired. If so, the activity in memory is checked to determine if it meets these specifications. The Print Evaluation Routine (S15) is entered, and the status report lines for the current activity

are output on tape 1 and the line printer (if requested). The above sequence is repeated until all activities have been processed. The routine then returns to Segment II Control.

The End Routine (S23) is the last routine entered. This routine types the word END, places an end-of-file mark on tape 1, rewinds tapes 1 and 2, and halts to terminate the program.

4. PROGRESS EVALUATION SORT PROGRAM

This section describes the SDS Progress Evaluation Sort Program (PEPSORT), the fourth in the SDS Project Management Series. It receives its input from the Progress Evaluation Program (PEP).

PROGRAM FUNCTION

The Progress Evaluation Sort Program (PEPSORT) sorts the activity status file from the Progress Evaluation Program. The output of the program is a sorted off-line magnetic tape and, if an on-line printer is available, an on-line printout. The activity status file, described in detail in the appendix, may be sorted on any of 19 keys and up to 13 keys in a single run.

PEPSORT USAGE

The minimum configuration used by PEPSORT consists of 8192 words of core, two magnetic tape units, paper tape and typewriter input, and an off-line printer. It automatically expands its operational capabilities as the configuration increases. Provisions are also made for operation with card reader input and on-line printer output. The number of activities processed by the program can be increased by adding more core storage to the configuration.

The maximum number of activities handled by the PEPSORT program operating on an SDS 920 Computer is

Memory (words)	Activities
8, 192	2864
12, 228	3375
16, 384	5461

These figures are slightly less for SDS 910 Computer operation due to the use of programmed operators.

Loading Procedures

The procedures for PEPSORT operations are presented in the Project Management System Reference Manual (publication number 90 08 18). Instructions for filling out data forms for PEPSORT are also contained in the Reference Manual.

PEPSORT Output

Output always consists of an off-line BCD file on unit 1. If an on-line printer is available, an on-line printout will occur in addition to the off-line file. The off-line provision allows the user to obtain multiple copies of the output without rerunning the program.

The report produced by PEPSORT furnishes progress information in various selected sequences which facilitate analysis of the status of the project and of the reasons for delays. The contents of the fields on this report are similar to those of the Progress Status Evaluation Report (see Figure 8), except that this report includes total and earned cost fields that do not appear on the Progress Status Evaluation Report. Aside from these fields, only the sequence of the activities has been changed.

Error List

Upon internal detection of an error, the program enters an error routine which types an error number. The meanings of error numbers are defined in the Reference Manual. After the error number has been printed, in most cases a restart can be made by clearing the halt. A complete restart from the beginning of the program can always be made by pressing START, then STEP and RUN.

PROGRAM ORGANIZATION DESCRIPTION

PEPSORT is a self-contained program, coded in relocatable form in SDS META-SYMBOL language. It contains its own loader (card or paper tape) and, therefore, does not require a MONARCH system tape. Extensive use has been made of the META-SYMBOL external reference provision for defining data storage to the various subroutines in the program. The majority of data used by the program is located in a subroutine called COMMON.

The program is composed of 36 subroutines connected by a main control program. This subroutine module construction provides a high degree of flexibility and facilitates system maintenance. Logic is provided for the program to

determine the type of configuration on which it is being run. Array data offsets are computed based on the amount of storage available. Automatic checks are made to determine if the problem being run will fit into the computer.

PEPSORT Storage Arrangment

The storage arrangement for PEPSORT is shown in Figure 10. Core storage is divided into three basic regions. The first region is the Common region. This region contains 273 words of constants, and scalar and short vector temporary storage. Common is followed by the program area. This area contains 35 relocatable subroutines which occupy 2463 words of core. The third region is a data area that extends from the end of the program section to the end of the core. During the input phase (Phase I) of execution, this section contains only the 517-word parameter file. The internal sort phase (Phase II) requires three vectors:

1. S vector – S_i is the successor of the i-th activity.
2. PC vector – PC_i is the value of the current primary code for the i-th activity.
3. SC vector – SC_i is the value of the current secondary code for the i-th activity.

These three vectors occupy the data region during Phase II.

During the tape sort, Phase III, different vectors are required in addition to the S vector:

1. Select vector – contains selected elements of the S vector for the current pass.
2. Sequenced vector – contains the selected vector, sorted on original activity number.
3. Tape record region – contains one record for each element in the select vector.

Tape Layouts

The input tape format is presented in detail in the appendix. This tape contains two files:

1. Parameter file (BCD) – one record
2. Activity file (BCD) – one record per activity

PEPSORT Execution Description

After completion of loading, control is transferred to the Main Control Program MAINCON (F. D. 3); the Main Control then branches to Address Calculation Routine ADRES1 (F. D. 23), which determines the amount of storage available and calculates the addresses to be used in Phases I and II. Main Control next transfers to the Parameter Read Routine PARMRD (F. D. 4) which reads the parameter file from the input tape and moves the effective date into Common.

Main Control then transfers to the Input Control Routine CARDRD (F. D. 8). CARDRD requests the input device from the typewriter using the Input Device Selection Routine SCAP2 (F. D. 5). The data records are then read from either paper tape or cards. The data is moved into Common and control is returned to Main Control. Control is then transferred to Run Heading and Parameter Print Routine HEADPG (F. D. 31) to print the heading page.

Next, each requested sort key is converted from a BCD code to a numeric code, which is the subscript of the key in the input tape record. These numbers are stored in vector form in Common. Control is then transferred to the Main Sort Control Routine SORT (F. D. 13). This routine selects the numeric keys two at a time (starting with the last two), reads the activity status file, and then transfers to Store Routine STORE (F. D. 16), which stores the selected record elements in the PC and SC vectors. If this is the first pass, the S vector is initialized to

$$S_i = i + 1, \quad i = 0$$

Control is then transferred to the Internal Sort Routine SORT2K (F. D. 25), which sorts the PC and SC values by modifying the S vector. Upon return from SORT2K, a check is made to determine if all the requested keys have been selected. If they have not, the next two are selected and the process is repeated until the keys are exhausted. When all the keys have been used, control is returned to MAINCON. Main Control then transfers to the

<table>
<tr><th>Approximate Location</th><th>PHASE I (Input)</th><th>PHASE II (Internal Sort)</th><th>PHASE III (Tape Sort)</th></tr>
<tr><td>Variable</td><td rowspan="3">Not Used</td><td>Secondary Code</td><td rowspan="2">Scattered Tape Area</td></tr>
<tr><td rowspan="2">Primary Code</td></tr>
<tr><td>Sequenced</td></tr>
<tr><td></td><td rowspan="2">Parameter File</td><td rowspan="2">Successor Vector</td><td>Select</td></tr>
<tr><td>05637_8
04637_8</td><td></td><td>Successor Vector</td></tr>
<tr><td>00424_8</td><td colspan="3">1. Restart Routine
2. Common Package
3. Main Control Program
4. Parameter Read Routine
5. Input Device Selection Routine
6. Card Read Routine
7. Paper Tape Read Routine
8. Input Control Routine
9. Magnetic Tape Write Routine
10. Magnetic Tape Read Routine
11. Magnetic Tape Ready Routine
12. BCD Code to Numeric Routine
13. Main Sort Control Routine
14. Read Codes Routine
15. Transfer Index Routine
16. Store Routine
17. Blanks to Zeros Routine
18. Data Pack Routine
19. Binary to Decimal Conversion Routine
20. Decimal into Binary Routine
21. Decimal into Binary Control Routine
22. Error Routine
23. Address Calculation Routine
24. Initial Successor Routine
25. Internal Sort Routine
26. Tape Sort Control Routine
27. Original Sequence Number Routine
28. Move Select Routine
29. Sort Select Routine
30. Read Select Routine
31. Run Heading and Print Routine
32. Main Print Control Routine
33. Report Line Output Routine
34. Sort Output Routine
35. Double-Precision Binary to BCD Conv.
36. End Routine</td></tr>
<tr><td>00000</td><td colspan="3">Common Area</td></tr>
</table>

Figure 10. PEPSORT Storage Arrangement

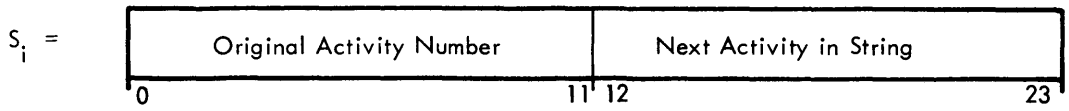
Address Calculation Routine ADRES2 (F. D. 23) to set up addresses for the tape sort. ADRES2 calculates the maximum string length (i.e., the maximum number of records that can be read into core) using the following relationship:

$$\text{STRINGM} = \frac{\text{Last location in core} - S_{(NA+1)}}{\text{Number of words per record} + 2}$$

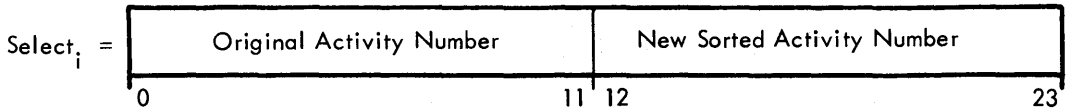
NA = Number of activities

After calculating the beginning of the select, sequenced, and tape areas, based on the maximum length of the string (STRINGM), control is returned to MAINCON. MAINCON then transfers to FINSORT (F. D. 26), the Main Tape Sort Control. FINSORT first goes to HEADPG (F. D. 31), which initializes the line and page counters for the output, prepares the off-line tape for writing prints the heading page on the line printer (if requested) and writes on the off-line tape. Next, Original Sequence Number Routine (F. D. 27) is entered. This routine merges the original sequence

numbers into the S vector. Each element of S now contains its subscript in the left half of its storage location and the next activity number in the right half:



Control is then transferred to Move Select MVESEL (F. D. 28), which moves through the S vector, assigns new activity numbers based on the sorted sequence, and then stores these numbers and the original activity numbers in one word and moves it to the select area:



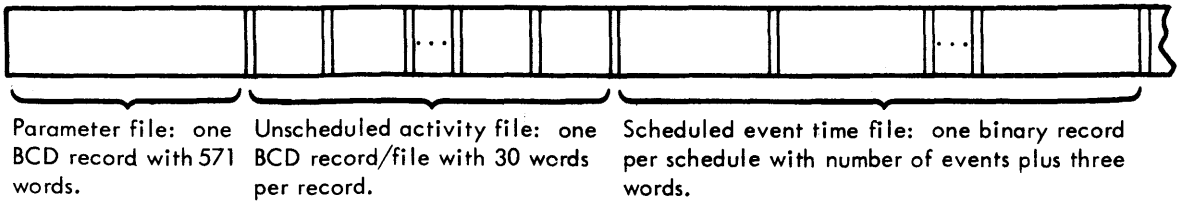
The Sort Select SORTSEL Routine (F. D. 29) sorts the select area on the original activity numbers and moves the sorted vector into the sequenced area. The sequenced area is now in the same sequence as the input tape. The Read Select READTP Routine (F. D. 30) reads the STRINGM records into the record area and positions them according to the new sequence. When this area is full, these records are printed on-line and written on tape. Next, a test is made to determine whether all activities have been processed; if they have not, the process is repeated for another block of STRINGM records. This is done until all records have been processed. At this time, control is returned to the Main Control routine. Main Control transfers to End (F. D. 36), which writes an end-of-file mark on tape 1 and types the word "END." End also prints total cost and earned cost for the project along with the ratio of earned to total expressed as a percent. On return from End, control is transferred to CARDRD to read input for the next case.

APPENDIX

SCHEDULE SPECTRUM PROGRAM OUTPUT TAPE LAYOUTS

The output tape from SSP (on magnetic tape unit 1) contains three files of information. These are pictured below.

Tape 1 Arrangement



Parameter file: one BCD record with 571 words.

Unscheduled activity file: one BCD record/file with 30 words per record.

Scheduled event time file: one binary record per schedule with number of events plus three words.

In the detailed layouts of each of these files, shown on the following pages,

B = blank
A = alphabetic character
N = numeric character
X = positions not used

Any other characters are actual representations.

Parameter File Layout

Word Number	Character				Field Description
1	1				Input Type Code (Type 1 – Run Heading)
		A	A	A	Run Heading, first 3 characters
2-20	A	A	A	A	Run Heading, characters 4 to end
21	2				Input Type Code (Type 2 – Run Parameters A)
21		A			Type of units of Duration Code (H=hours, D=days, W=weeks, M=months)
21			N	N	Day of starting date
22	A	A	A		Month of starting date
22				N	First character of starting date
23	N				Second character of starting date
23		N	N		Working hours/day
23				A1	First day-off, first character
24	A1				First day-off, second character
24		A2	A2		Second day-off
24				A3	Third day-off, first character
25	A3				Third day-off, second character
25		A4	A4		Fourth day-off
25				A5	Fifth day-off, first character
26	A5				Fifth day-off, second character
26		A6	A6		Sixth day-off
26				A7	Seventh day-off, first character
27	A7				Seventh day-off, second character
27		X	X	X	Not used
28-40	X	X	X	X	Not used
41	3	0	0	0	Input Type Code (Type 3 – Holiday Specifications)

Parameter File Layout (cont.)

Word Number	Character				Field Description
42	0	0	N	N	Day of first holiday
43	0	A	A	A	Month of first holiday
44	0	0	N	N	Year of first holiday
45-536	Repeats above format for remaining 165 holidays (3 words following last holiday contain all zeros)				
537	4				Input Type Code (Type 4 – Run Parameters B)
537		A			Not used
537			A		Major activity option code
537				A	Crash option code
538	A	A	A	A	First 4 characters of Run ID or Date
539	A	A	A	A	Last 4 characters of Run ID or Date
540-556	A	A	A	A	Run Heading
557	N	N	N	N	Number of activities
558-570					Not used
571	N	N	N	N	Checksum – For sync test and read test

Unscheduled Activity Record Layout

Word Number	Character				Field Description
1	5	0	0	0	Input Type Code (Type 5 – Activity Record)
2	N	A	A	A	I node
3	N	A	A	A	J node
4	0	0	0	A	IJ Dupe Code
5	0	N	N	N	Normal Duration
6	0	N	N	N	Crash Duration
7	N	N	N	N	Normal Cost, first 4 characters
8	N	0	0	0	Normal Cost, fifth character
9	N				Dummy Record Indicator (0 = not dummy; 1 = milestone dummy)
9		A			Transaction Code
9			A		Cost Continuity Indicator
9				A	Major Activity Code
10	0	N	N	A	First 3 characters of milestone date
11	A	A	N	N	Last 4 characters of milestone date
12	A	A	A	A	Part of Cost Center Code or Activity Description
13	A	A	0	0	Part of Cost Center Code or Activity Description
14	A	A	A	A	Part of Responsibility Code or Activity Description
15	A	A	0	0	Part of Responsibility Code or Activity Description
16-23	A	A	A	A	Remainder of Activity Description
24	N	N	N	N	Internal node, M
25	N	N	N	N	Internal node, N

Unscheduled Activity Record Layout (cont.)

Word Number	Character				Field Description
26	N	N	N	N	Internal Activity Number, K
27	N	N	N	N	Crash Cost, first 4 characters
28	N	0	0	0	Crash Cost, fifth character
29	N	N	N	N	Original Activity Sequence Number
30	N	N	N	N	Checksum

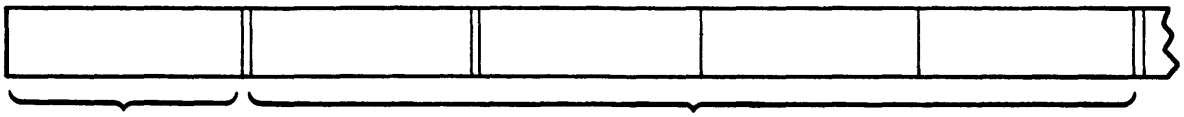
Scheduled Event Time File Record (binary)

Word Number	Character				BIN/BCD	Field Description
1	7	0	0	0	BCD	Input Type Code (Type 7 – Schedule File)
2	0	0	B	B	BIN	Total Project Duration Time – basic units
3	0	0	B	B	BIN	Time of Event 1 – basic units
4	0	0	B	B	BIN	Time Event 2 – basic units
		⋮				
NE+2	0	0	B	B	BIN	Time of Event NE (last event) – basic units
NE+3	B	B	B	B	BIN	Checksum

DETAILED SCHEDULE REPORT PROGRAM INPUT TAPE LAYOUTS

The input tape to DSRP is the output tape from SSP, described above.

PROJECT EVALUATION PROGRAM TAPE LAYOUTS



Project Parameter File: one BCD record containing 517 words.

Activity Status File: one 65-word BCD record for each activity in the project.

In the following detailed layouts of each of these files,

B = blank
A = alphabetic character
N = numeric character
X = positions not used

Any other characters are actual representations.

Project Parameter File Layout (BCD)

Word Number	Character				Field Description	
1	1	B	B	B	Input Type Code (Type 1 – PEP Parameter File)	
2	B	B	B	A	Units type, H, D, W, M	
3	B	B	N	N	Start Day	
4	B	A	A	A	Start Month	
5	B	B	N	N	Start Year	
6	B	B	N	N	Hours per Day	
7	B	B	A	A	First Nonworking Day	
8	B	B	A	A	Second Nonworking Day	
9	B	B	A	A	Third Nonworking Day	
10	B	B	A	A	Fourth Nonworking Day	
11	B	B	A	A	Fifth Nonworking Day	
12	B	B	A	A	Sixth Nonworking Day	
13	B	B	A	A	Seventh Nonworking Day	
14	B	B	B	A	Major Activities Only indicator	
15	B	B	N	N	Effective Date Day	
16	B	A	A	A	Effective Date Month	
17	B	B	N	N	Effective Date Year	
18	B	B	B	B	Not used	
19	B	B	B	B	Not used	
20	B	B	B	B	Not used	
21	B	B	B	B	Not used	
22	B	B	B	B	Not used	
23	B	B	B	B	Not used	
24	B	B	N	N	First Holiday Day	
25	B	A	A	A	First Holiday Month	
26	B	B	N	N	First Holiday Year	
27	B	B	N	N	Second Holiday Day	
28	B	A	A	A	Second Holiday Month	
29	B	B	N	N	Second Holiday Year	492 Words (zero if no holiday)
:	:	:	:	:		
513	B	B	N	N	164th Holiday Day	
514	B	A	A	A	164th Holiday Month	
515	B	B	N	N	164th Holiday Year	
516	0	0	0	0	Zero	
517	0	0	0	0	Zero	

Activity Status Record Layout (BCD)

Word Number	Character				Field Description
1	2	B	B	B	Type Code (Type 2 – Activity Status Record)
2	N	A	A	A	Activity I Node
3	N	A	A	A	Activity J Node
4	B	B	B	A	Activity Dupe Code
5	N	N	N	N	Internal Activity Number
6	N	N	N	N	Internal M Node
7	N	N	N	N	Internal N Node
8	N	N	N	N	Scheduled Activity Duration (Basic Units)
9	B	B	N	N	Scheduled Start Day
10	B	A	A	A	Scheduled Start Month
11	B	B	N	N	Scheduled Start Year
12	B	B	N	N	Scheduled Finish Day

Activity Status Record Layout (BCD) (cont.)

Word Number	Character				Field Description
13	B	A	A	A	Scheduled Finish Month
14	B	B	N	N	Scheduled Finish Year
15	N	N	N	N	Actual Duration (Basic Units)
16	B	B	N	N	Actual Start Day
17	B	A	A	A	Actual Start Month
18	B	B	N	N	Actual Start Year
19	B	B	N	N	Actual Finish Day
20	B	A	A	A	Actual Finish Month
21	B	B	N	N	Actual Finish Year
22	B	B	N	N	Original Scheduled Finish Day
23	B	A	A	A	Original Scheduled Finish Month
24	B	B	N	N	Original Scheduled Finish Year
25	B	B	N	N	Early Start Day
26	B	A	A	A	Early Start Month
27	B	B	N	N	Early Start Year
28	B	B	N	N	Late Start Day.
29	B	A	A	A	Late Start Month
30	B	B	N	N	Late Start Year
31	B	B	N	N	Early Finish Day
32	B	A	A	A	Early Finish Month
33	B	B	N	N	Early Finish Year
34	B	B	N	N	Late Finish Day
35	B	A	A	A	Late Finish Month
36	B	B	N	N	Late Finish Year
37	N	N	N	N	Predicted Status Criterion Magnitude
38	±	B	B	B	Predicted Status Criterion Sign
39	N	N	N	N	Estimated Total Float
40	N	N	N	N	Duration Difference Magnitude
41	±	B	B	±	Duration Difference Sign and Estimated Float Sign
42	N	N	N	N	Total Slip Magnitude
43	±	B	B	B	Total Slip Sign
44	N	N	N	N	Number of slips
45	0	0	0	A	Previous Finish Indicator (Y or N)
46	B	B	B	A	First Character of Cost Code
47	A	A	A	A	Second, Third, Fourth, and Fifth Characters of Cost Code
48	A	B	A	A	Sixth Character of Cost Code, Blank, First, and Second Characters of Responsibility Code
49	A	A	A	A	Third, Fourth, Fifth, and Sixth Characters of Responsibility Code
50	B	A	A	A	Activity Description
51-57	A	A	A	A	Activity Description
58	A	B	B	B	Activity Description
59	B	B	B	A	Major Activity Indicator
60	N	N	N	N	Actual Status Criterion Magnitude
61	±	B	B	B	Actual Status Criterion Sign
62	B	B	N	N	} Activity Total Cost
63	N	N	N	N	
64	B	B	B	N	} Percent Completed
65	N	N	N	N	

PROGRESS EVALUATION SORT PROGRAM INPUT TAPE LAYOUT

The Activity Status Tape, output on unit 2 by PEP and input to PEPSORT on unit 2, contains two files of information: The project parameter file and the activity status file. These files are illustrated in the preceding discussion, "Project Evaluation Program Tape Layouts."

SUBROUTINE ROSTER

A subroutine roster is given for each program in the Project Management System. Some routines are used by several programs, and their names appear in the individual subroutine rosters; however, only one flowchart is provided for each such subroutine.

In the SSP and PEP subroutine rosters the deck numbers are preceded by a letter indicating that the subroutine is used in the first (F), second (S), or third (T) segment of that program.

SCHEDULE SPECTRUM PROGRAM

Deck Number	Routine	Page
F01	Binary Card Load Routine	-
F02	Common Data and Storage Area (Listing)	53
F03	SSP Control Program	78
F04	Magnetic Tape Read	126
F05	Magnetic Tape Write	125
F06	Error	123
F07	Binary/Decimal Integer Conversion	167
F08	Decimal/Binary Conversion Control	81
F09	Decimal/Binary Conversion	167
	Segment I – Part B:	
F10	Input/Output Area (Listing)	57
F11	Create Abbreviated Activity Records	81
F12	Set Up Milestone Dummy Activity Records	82
F13	End of Activity File Record	83
F14	Assign Unsorted Activity Numbers	84
F15	Reel Change Verification	83
F16	Write Updated Parameter Record	84
	Segment I – Part A:	
F17	Non-Activity Oriented Initialization	84
F18	I/O Device Selection	129
F19	Parameter and Holiday Input	85
F20	Change Input Device Selection	86
F21	Parameter Conversion	86
F22	Holiday Sort	133
F23	Calendar Initialization	86
F24	Activity Record Input	87
F25	Milestone Time Set-Up	89
F26	Write Activity Record	89
F27	Type Reel Change Message	89
F28	Skip Program Segments	89
F29	Paper Tape Read	124
F30	Card Read	123
F31	Date to Basic Units Conversion	165
F32	Calendar	110
F33	End of Segment I Load	89
	Segment II – Part B:	
S01	Write Scheduled Event Time Record	90
S02	Crash	91
	Segment II – Part A:	
S03	Create Crash Data Arrays	95
S04	not used	
S05	Topological Sort	145
S06	M and N Assignment	147
S07	Topological Tape Sort	149
S08	Activity Oriented Initialization	99
S09	End of Segment II Load	100

Deck Number	Routine	Page
Segment III – Part B:		
T01	Input/Output Area (Listing)	57
T02	Schedule Summary Report	100
T03	Report Line Output	164
T04	Double-Precision Binary/Decimal Conversion	128
T05	Calendar	110
T06	Date to Basic Units Conversion	165
Segment III – Part A:		
T07	Set Up Data Arrays for Summary Report	102
T08	Write End of Time File Record	103
T09	Read Parameter – Unit 1	103
T10	Parameter Print	134
T11	End of Segment III Load	103
T12	SSP Start Routine	103
DETAIL SCHEDULE REPORT PROGRAM		
1	Main Control – Part 1	104
2	Delay	105
3	End	105
4	Duration Read	106
5	Input Conversion	108
6	Time Vector	109
7	Calendar	110
8	Detail Schedule Report	116
9	Detail Schedule Print	121
10	Error	123
11	Card Read	123
12	Paper Tape Read	124
13	Magnetic Tape Write	125
14	Magnetic Tape Read	126
15	Report Line Output	127
16	Binary/Decimal Integer Conversion	153
17	Binary/BCD Calendar Date Conversion	128
18	Decimal/Binary Conversion	128
19	Double-Precision Binary/BCD Conversion	128
20	Common Package (Listing)	59
21	Main Control – Part 2	104
22	Parameter Control	128
23	Magnetic Tape Set Up	128
24	I/O Device Selection	129
25	Parameter Input	130
26	Holiday Sort	133
27	Run Heading and Paramter Print	134
PROJECT EVALUATION PROGRAM		
Segment I:		
F01	Common Package (Listing)	64
F02	Magnetic Tape Read	126
F03	Segment II Load	136
F04	Segment I Control	137
F05	First Run Request	136
F06	I/O Device Selection	129
F07	Segment I Set Up	136
F08	Erase	153
F09	Initial Input Tape	138
F10	Parameter and Holiday Input	142
F11	Holiday Sort	133
F12	Initial Abbreviated Activity File	141
F13	Network Change	141

Deck Number	Routine	Page
Segment I (cont.)		
F14	Blanks to Zeros	153
F15	Topological Sequence Control	144
F16	Unsorted Activity Number Assignment	144
F17	not used	
F18	Topological Sort	145
F19	M and N Assignment	147
F20	Tape Sort	149
F21	Read BCD Record	153
F22	Schedule Request	151
F23	Magnetic Tape Write	125
F24	Binary/Decimal Integer Conversion	153
F25	Magnetic Tape Set Up	128
F26	Card Read	123
F27	Paper Tape Read	124
F28	Error	123
F29	Utility Package	152
F30	Date Comparison	152
F31	Decimal/Binary Conversion	167
F32	Segment I Data	152
Segment II:		
S01	Common Package (Listing)	64
S02	Magnetic Tape Read	126
S03	Segment II Load	136
S04	Segment II Control	153
S05	Erase	153
S06	Activity Change	154
S07	Post Change	156
S08	Blanks to Zeros	153
S09	Segment II Set Up	158
S10	Forward PERT	158
S11	Reverse PERT	159
S12	Status Evaluation	159
S13	Print Control	162
S14	Print Initialization	167
S15	Print Evaluation	163
S16	Report Line Output	164
S17	Calendar	110
S18	Date to Basic Units Conversion	165
S19	Magnetic Tape Set Up	128
S20	Magnetic Tape Write	125
S21	Card Read	123
S22	Paper Tape Read	124
S23	End	167
S24	Error	123
S25	Utility Package	152
S26	Date Comparison	152
S27	Decimal/Binary Conversion	167
S28	Double-Precision Binary/BCD Conversion	128
S29	Binary/Decimal Integer Conversion	153
S30	Segment II Preparation	153
PROGRESS EVALUATION SORT PROGRAM		
1	Restart	167
2	Common Package (Listing)	72
3	Main Control	168
4	Parameter Read	167
5	Input Device Selection	129
6	Card Read	123

Deck Number	Routine	Page
PEPSORT (cont.)		
7	Paper Tape Read	124
8	Input Control	169
9	Magnetic Tape Write	125
10	Magnetic Tape Read	126
11	Magnetic Tape Ready	170
12	BCD Code to Numeric	170
13	Main Sort Control	171
14	Read Codes	172
15	Transfer Index	173
16	Store	174
17	Blanks to Zeros	167
18	Date Pack	167
19	Binary/Decimal Conversion	167
20	Decimal/Binary Conversion	167
21	Decimal/Binary Control	167
22	Error	123
23	Address Calculation	175
24	Initial Successor	167
25	Sort on Two Keys	175
26	Tape Sort Control	176
27	Original Sequence Number	167
28	Move Select	176
29	Sort Select	177
30	Read Select	177
31	Run Heading and Parameter Print	134
32	Main Print Control	177
33	Report Line Output	127
34	Sort Output	178
35	Double-Precision Binary/BCD Conversion	128
36	End	167

```
       0000C                                1            RØRG  0                   CØMMØN DATA AND STØRAGE                   0000
*      00000  0 01 0 00000                  2            BRU   STPRØG              TØ READ SEG 1 FRØM TAPE AND EXECUTE       0000
                                            3 *                                                                              0000
       00001  0 32 0 00002                  4            WIM   02                  FØR PRØG READ FRØM CARDS                  0000
*      00002  0 01 0 00000                  5            BRU   WRSEG1              TØ WRITE SEG 1 ØN TAPE                    0001
                                            6 *                                                                              0002
*      00003  0 01 0 00000                  7            BRU   WRSEG2              TØ WRITE SEG 2 ØN TPAE                    0003
                                            8 *                                                                              0004
*      00004  0 01 0 00000                  9            BRU   WRSEG3              TØ WRITE SEG 3 ØN TAPE                    0005
                                           10 *                                                                              0006
                                           11 *                                                                              0008
       00005                               12 $ABØFF     RES   1             IND   DURATIØN VECTØR ØRIGIN MINUS 1            0840
       00006                               13 $ABRAC     RES   1             IND   ØRIGIN ØF 1ST WD ØF ABR ACT REC           0360
       00007                               14 $ABRACI    RES   1             IND   ØRIGIN ØF I ØF ABR ACT REC                0370
       00010                               15 $ABRACIL   RES   1             INDX  LAST +1 LØC ØF I REC NØT INCL REMØVAL     0000
       00011                               16 $ABRACIM1  RES   1             IND   ABRACI-1                                  0400
       00012                               17 $ABRACJ    RES   1             IND   ØRIGIN ØF J ØF ABR ACT REC                0380
       00013                               18 $ABRACJL   RES   1             IND   LAST+1 LØC ØF J REC NØT INCL REMØVAL      0420
                                           19 *                                                                              0430
       00014                               20 $ABRACJM1  RES   1             IND   ABRACJ-1                                  0410
       00015                               21 $ABRACM    RES   1                   M,N ARRAY ØRIGIN-1,ØVER ØLD I ARRAY)      0000
       00016                               22 $ABRACM1   RES   1             IND   ABRAC-1                                   0390
       00017                               23 $ACTH      RES   1                   START ØF PREVIØUS THREAD DURING RANK      0530
                                           24 *                                      START ØF NETWØRK AFTER RANK             0540
       00020                               25 $ACTK      RES   1                   LAST THREADED ACT NUM USED +1             0300
       00021  00000035                     26 $ACTNUMWD  DATA  29                  NUMBER ØF WØRDS IN ACTIVITY RECØRD        1290
       00022  01000000                     27 $BCD1000   DATA  '1000'                     BCD 1000                           1485
       00023  02000000                     28 $BCD2000   DATA  '2000'                     BCD 2000                           1486
       00024  06000000                     29 $BCD6000   DATA  '6000'              END ACTIVITIES CØDE                       1487
       00025  11000000                     30 $BCD9000   DATA  '9000'              END FILE Ø CØDE                           1488
       00026  60606060                     31 $BCDBLK    DATA  '    '              BCD BLANK                                 1285
       00027  60606033                     32 $BCDDP    DATA  '   .'               BCD DECIMAL PØINT                         1490
       00030  20606060                     33 $BCDPL     DATA  '+   '              BCD PLUS                                  1500
       00031  00006060                     34 $BLKDAY    DATA  '  '                BLANKS                                    1190
       00032                               35 $CABRAC    RES   1                   LØC NEXT 1ST WD ABR ACT TØ BE SET UP      0320
       00033                               36 $CABRACI   RES   1                   LØC NEXT I WD ABR ACT TØ BE SET UP        0330
       00034                               37 $CABRACJ   RES   1                   LØC NEXT J WD ABR ACT TØ BE SET UP        0340
*      00035  0 00 0 00000                 38 $CARST     HLT   SCAP21              1ST LØC CAN'T CLØBBER-SEG 2                000
       00036                               39 $CHINDEV   RES   1                   CØDE FØR DEVICE SELECTED FØR              0110
                                           40 *                                      CHANGES TØ INPUT FILES                  0120
                                           41 *                                      C=CDS P=PAPER TAPE N=NØNE               0130
       00037                               42 $CØFF      RES   1             IND   SLØPE VECTØR ØRIGIN MINUS 1               0830
       00040                               43 $CPBRA     RES   1                   CØNTRØL PRØG-BRANCH A  +PATH1,-PATH2      0210
       00041                               44 $CRASH     RES   1                   CØMPLETELY CRASHED IND = -1               0855
                                           45 *                                                                              0860
       00042                               46 $CRASHØP   RES   1             000C  CRASH ØPTIØN CØDE FRØM PARAMETER B        0660
       00043                               47 $CURHØL    RES   1                   LØC ØF NEXT HØLIDAY TØ BE STØRED          0140
       00044  37777777                     48 $DACTI     DATA  037777777           CØNST FØR I FØR DELETE + END ACT          0000
       00045                               49 $DATBASU   RES   1                   DATE BASIC UNITS                          1245
       00046                               50 $DAY       RES   1                                                             0606
       00047                               51 $DAYØFF    RES   7                   DAY ØFF VECTØR                            1150
       00056  00006264                     52 $DAYSWK    DATA  'SU','MØ','TU','WE','TH','FR','SA'                            1210
       00057  00004446
```

```
00060  00006364
00061  00006625
00062  00006330
00063  00002651
00064  00006221
       00000055          53 $DAYWK0   EQU  DAYSWK-1                                                    1200
00065                    54 $DS       RES  1          UNITS OF CONVERTED DATE                          0610
00066                    55 $DUR1     RES  1          DAY OF CONVERTED CALENDAR DATE                   0580
00067                    56 $DUR2     RES  1          MONTH OF CONVERTED CALENDAR DATE                 0590
00070                    57 $DUR3     RES  1          YEAR OF CONVERTED CALENDAR DATE                  0600
00071  2 00 0 00047      58 $DYOFF0   HLT  DAYOFF,2                                                    1170
00072  2 00 0 00046      59 $DYOFM1   HLT  DAYOFF-1,2                                                  1160
00073                    60 $ENDAC    RES  1          OUTAREA + ACTNUMWD                               1280
00074                    61 $ENDFILE  RES  1                                                           0935
                         62 *                                                                          0940
                         63 *                                                                          0990
00075  00000000          64 $ERD      DATA 0          DUPE IJ CODE FOR ERROR TYPE OUT                  0640
00076  00000000          65 $ERI      DATA 0          I STORE FOR ERROR TYPE OUT                       0620
00077  00000000          66 $ERJ      DATA 0          J STORE FOR ERROR TYPE OUT                       0630
00100                    67 $FEND     RES  1      IND LAST FLOW +1                                     0880
00101                    68 $FOFF     RES  1      IND FLOW VECTOR ORIGIN MINUS 1                       0810
00102                    69 $FSTCH    RES  1          1ST CHANGE ACT IND FOR ACT INPUT                 0435
00103                    70 $HNA      RES  1          HALF NUMBER OF ACT                               0740
00104                    71 $HNE      RES  1          HALF NUMBER OF EVENTS                            0750
00105                    72 $HNGNE    RES  1          HALF NEGATIVE NUMBER OF EVENTS                   0780
00106                    73 $HOLBIN   RES  165        BINARY HOLIDAY AREA                              0000
00353  2 00 0 00105      74 $HOLDM1   HLT  HOLBIN-1,2                                                  1145
*  00354  0 35 0 00000    75 $HOLINIT  STA  PARFD      DUMMY TO STORE IN 1ST WD CAL HOLDAS              1310
00355                    76 $HOLORG   RES  15                                                          1255
       00000354          77 $HOLOM1   EQU   HOLORG-1                                                   1259
00374                    78 $LABOFF   RES  1      IND LABEL VECTOR ORIGIN MINUS 1                      0800
00375                    79 $LINEC    RES  1          LINE COUNTER FOR OUTPUT ROUTINES                 0690
*  00376  0 00 0 00000    80 $LOCPRB   HLT  PARAMB                                                      0145
00377                    81 $LOSTDYS  RES  1           NEGATIVE OF LOST DAYS                           0000
00400                    82 $LSTLOC   RES  1          LAST LOCATION IN MEMORY                          0350
00401                    83 $MACC     RES  1          LAST NODE NUMBER USED                            0310
00402  77000000          84 $MASK1    DATA 077000000                                                   1320
00403  00770000          85 $MASK2    DATA 000770000                                                   1350
00404  00000077          86 $MASK4    DATA 000000077                                                   1360
00405  77770000          87 $MASK12   DATA 077770000                                                   1330
00406  00007777          88 $MASK34   DATA 000007777                                                   1340
00407  77777700          89 $MASK123  DATA 077777700                                                   1345
00410  00777777          90 $MASK234  DATA 00777777                                                    1365
00411  00037777          91 $MASKAD   DATA 000037777                                                   1370
00412  00077777          92 $MASKADR  DATA 000077777                                                   1380
00413                    93 $MAXABR   RES  1          MAXIMUM NUMBER ABREVIATED ACT REC                0490
00414  77776650          94 $MIN150   DATA  -600      CONSTANT TO MEASURE TAPE ERASE LNGTH             0000
00415                    95 $MNOFF    RES  1      IND NODE NUMBER VECTOR ORIGIN MINUS 1                0850
00416  40000000          96 $MONCONST DATA 040000000  CONST FOR NO HALT IN MONCARCH LOAD               0000
00417  77777777          97 $MONE     DATA -1         MINUS ONE = 77777777                             1450
00420                    98 $MONTH    RES  1                                                           0605
00421  00412145          99 $MONYR    DATA 'JAN','FEB','MAR','APR','MAY','JUN','JUL'                   1120
00422  00262522
00423  00442151
00424  00214751
```

```
00425  00442170
00426  00416445
00427  00416443
00430  00216427          100          DATA *AUG*,*SEP*,*OCT*,*NOV*,*DEC*                    1130
00431  00622547
00432  00462363
00433  00454665
00434  00242523
       00000420          101 $MONYRO  EQU  MONYR-1                                          1110
00435  00000000          102 $MSTEND  DATA 0          CURRENT NTWK END TIME INCL MSTN       0000
00436                    103 $NA      RES  1          NUMBER OF ACTIVITIES                  0720
00437                    104 $NANRM   RES  1          NEG OF NUM ACT NOT INCL TO BE REMOVD  0460
00440                    105 $NCMAX   RES  1          MAXIMUM NORMAL COST                   0000
00441                    106 $NCOFF   RES  1      IND NORMAL COST VECTOR ORIGIN MINUS 1      0870
00442                    107 $NCSCALE RES  1          NORMAL COST SCALE FACTOR              0000
00443                    108 $NDYOFF  RES  1          NUMBER DAYS OFF PER WEEK              1220
00444                    109 $NE      RES  1          NUMBER OF EVENTS                      0730
00445                    110 $NEGACWD RES  1          NEGATIVE OF NUMBER WDS IN ACT RES     1270
00446                    111 $NEP2    RES  1          NUMBER EVENTS +2                      0785
                         112 *                                                              0790
       00000437          113 $NGNA    EQU  NANRM      NEGATIVE NUMBER OF ACT IN FINAL NTWK  0760
00447                    114 $NGNE    RES  1          NEGATIVE NUMBER OF EVENTS             0770
00450                    115 $NHY     RES  15                                               1254
       00000447          116 $NHYM1   EQU   NHY-1                                          1258
00467                    117 $NUMAC   RES  1          NUMBER OF ACTIVITIES                  0440
       00000467          118 $NUMACT  EQU  NUMAC                                           0000
00470                    119 $NUMACNG RES  1          NEGATIVE NUMBER OF ACTIVITIES         0450
00471                    120 $NUMTWD  RES  1          NUMBER WORDS IN TIME TAPE RECORD      0715
00472                    121 $NWD     RES  1          NUMBER OF WORKING DAYAS TO DATE       1225
00473                    122 $NWDPW   RES  1          NUMBER WORKING DAYS PER WEEK          1230
00474                    123 $NWDY    RES  15                                               1253
       00000473          124 $NWDYM1  EQU   NWDY-1                                         1257
00513                    125 $NWHPD   RES  1          NUMBER WORKING HOURS PER DAY          1240
00514                    126 $NXTABR  RES  1          LOC OF NEXT ABR ACT REC OF 1ST MSTN   0470
                         127 *                          WHEN THRU SETTING UP NORMAL ABR RC  0480
00515                    128 $NXTABRI RES  1          LOC I WD OF NEXT ABR ACT REC          0000
00516                    129 $NXTMSTA RES  1          NEXT MSTN LMY LOC (LAST ABR REC +1)   0510
                         130 *                                                              0520
00517                    131 $NXTRN   RES  1          NEXT RECORD NUMBER TO BE ASSIGNED     0550
                         132 *                                                              0560
00520  00000001          133 $ONE     DATA 1                                                1410
*  00521  0 00 0 00354    134 $PARFDLC HLT  PARFD                                           0147
*  00522  0 00 0 00000    135 $PARFMLC HLT  PARFM                                           0146
00523  00001072          136 $PARNUMWD DATA 570        NUMBER OF WORDS IN PARAMETER RECORD   1300
00524  00000000          137 $PRNTCODE DATA 0          CODE FOR WHETHER OR NOT OUTPUT IS     0050
                         138 *                          DESIRED ON ON-LINE PRINTER          0060
                         139 *                          Y=YES, N=NO                         0070
                         140 * NOTE THAT ABOVE 2 WORDS ARE TESTED BEFORE TYPE-IN OF THE CODES
                         141 *   IS REQUESTED. IF CODES ARE THERE (FROM CARDS ADDED TO
                         142 *   MACHINE LANGUAGE PROGRAM DECK) THE REQUESTS ARE NOT EXECUTED
00525                    143 $RECIND  RES  1          INDICATES IF 1ST TIME THRU INPUT      0570
00526                    144 $RELMST  RES  1          CURRENT RELATIVE MSTN TIME            0500
00527  00000000          145 $RGINDEV DATA 0          CODE FOR DEVICE SELECTED FOR  REG     0030
                         146 *                          INPUT: C=CDS P=PAPER TP M=MAG TP    0040
00530  00000002          147 $RPUNO   DATA 2          UNIT NUMBER FOR RLOUTSR               0000
```

```
00531                    148 $RUNIDAS   RES  1              RUN ID FROM RUN PARAMETER CARD          0670
00532                    149 $RUNIDBS   RES  1                                                      0680
*  00533  0 00 0 00000   150 $SARST     HLT  SCAP27         1ST LOC CAN'T CLOBBER-SEG 3             0000
00534                    151 $SCALE     RES  1              PLACES TO SHIFT TO SCALE SLOPE          0650
00535                    152 $SCHED     RES  1              ORIGIN OF SCHEDULE FILE COVER LABELS    0920
00536                    153 $SCHEDOFF  RES  1              FIRST TIME OF SCHEDULE FILE +1          0930
00537                    154 $SCHIN     RES  1              ORIGIN FOR INPUT OF SCHED TIME FILE     0000
00540                    155 $SCHNUM    RES  1              SCHEDULE NUMBER FOR SUMMARY RPT LINE    0700
                         156 *                                                                      0710
00541                    157 $SDAY      RES  1              START DAY   BINARY                      1080
00542                    158 $SDTIY     RES  1                                                      1050
00543                    159 $SEG1NW    RES  1              NUMBER OF WORDS IN PROGRAM SEG 1        0000
00544                    160 $SEG2ENDC  RES  1              END SEG 2. ABR ACT REC START NEXT LOC   0000
00545                    161 $SEG2NW    RES  1              NUMBER OF WORDS IN PROGRAM SEG 2        1470
00546                    162 $SEG3NW    RES  1              NUMBER OF WORDS IN PROGRAM SEG 3        1480
00547                    163 $SELECT    RES  1              START OF SAVE FOR 1ST WD OF ABR ACT     0220
                         164 *                               RECORD FOR NEXT 50 REC- THREADEDSEQ    0230
00550                    165 $SELECTL   RES  1        IND   LAST +1 WD OF SELECT AREA               0280
00551                    166 $SEQ       RES  1              START OF SAVE FOR SAME AS SELECT        0240
                         167 *                               AREA,BUT IN TAPE REC NUM SEQ           0250
00552                    168 $SEQEND    RES  1              LAST+1 WD OF TOPOL SEQUENCED AREA       0000
00553                    169 $SEQL      RES  1        IND   LAST +1 WD OF SEQUENCED AREA            0290
         00000534        170 $SLSCALE   EQU  SCALE          SLOPE SCALE FACTOR                      0000
00554                    171 $SLMAX     RES  1              MAXIMUM SLOPE                           0000
00555                    172 $SLOPEIN   RES  1              SLOPE VECTOR ORIGIN                     0890
00556                    173 $SMONTH    RES  1              START MONTH-BINARY MONTH NUMBER         1090
00557  77777716          174 $STRINGM   DATA -50           MINUS NUMBER RECORDS IN SORT STRING      0000
00560                    175 $SYEAR     RES  1              START YEAR                              1100
00561                    176 $SYRTYP    RES  1             START YEAR TYPE                         1051
00562  20000000          177 $TAG       DATA 020000000      INDEX TAG BIT                           0000
         00000562        178 $TAGBIT    EQU  TAG            INDEX TAG BIT                           0000
00563  57777777          179 $TAGDEL    DATA 057777777      MASK TO DELETE INDEX TAG                0000
00564                    180 $TEND      RES  1        IND   LAST TIME +1                           0910
00565                    181 $THREDIN   RES  1              START OF STORAGE FOR SC COMPLETE        0260
                         182 *                               ACTIVITY RECORDS IN THREADED SEQUEN    0270
00566  00000003          183 $THREE     DATA 3                                                      1430
00567                    184 $TIME      RES  1              TIME VECTOR ORIGIN                      0900
00570  00000000          185 $TLAST     DATA 0              TIME OF LAST N NODE OF NTWRK SAVE       0000
00571                    186 $TOFF      RES  1        IND   TIME VECTOR ORIGIN MINUS 1              0820
00572                    187 $TOUTO     RES  1              ORIGIN OF TIME OUTPUT AREA              0000
00573  00000002          188 $TWO       DATA 2                                                      1420
00574                    189 $TYSDAY    RES  1              TYPE OF DAY PROJECT STARTS ON           0000
00575                    190 $UNITS     RES  1              BINARY DURATION UNIT TYPE CODE          1250
00576                    191 $WB1       RES  1              COMMON INTERMEDIATE STORAGE             0150
00577                    192 $WB2       RES  1                                                      0160
00600                    193 $WB3       RES  1                                                      0170
00601                    194 $WB4       RES  1                                                      0180
00602                    195 $WB5       RES  1                                                      0190
00603                    196 $YEAR      RES  1                                                      0604
00604                    197 $YMDFOR    RES  1              INDICATES IF CAL DATE ECD OF BIN        1180
*  00605  2 00 0 00000   198 $YMDHM1    HLT  PARHOL,2   IND                                         1263
*  00606  2 00 0 00000   199 $YMDH0     HLT  YMDH,2      IND                                        1265
*  00607  2 00 0 00522   200 $YMDHP1    HLT  PARFM,2    IND                                         1264
00610  00000000          201 $ZERO      DATA 0                                                      1400
                         202            END                                                         9995
```

```
00000                STPR0G
00002                WRSEG1
00003                WRSEG2
00004                WRSEG3
00035                SCAP21
00521                PARFD
00376                PARAMB
00607                PAREM
00533                SCAP27
00605                PARH0L
00606                YMDH
```

SSP INPUT/OUTPUT AREA

```
00000     1          R0RG  000                    2060
00000     2 $IA0     RES   1                      2070
00001     3 $IA1     RES   1                      2072
00002     4 $IA2     RES   1                      2074
00003     5 $IA3     RES   1                      2076
00004     6 $IA4     RES   1                      2078
00005     7 $IA5     RES   1                      2080
00006     8 $IA6     RES   1                      2082
00007     9 $IA7     RES   1                      2084
00010    10 $IA8     RES   1                      2086
00011    11 $IA9     RES   1                      2088
00012    12 $IA10    RES   1                      2090
00013    13 $IA11    RES   1                      2092
00014    14 $IA12    RES   1                      2094
00015    15 $IA13    RES   1                      2096
00016    16 $IA14    RES   1                      2098
00017    17 $IA15    RES   1                      2100
00020    18 $IA16    RES   1                      2102
00021    19 $IA17    RES   1                      2104
00022    20 $IA18    RES   1                      2106
00023    21 $IA19    RES   1                      2108
00024    22 $IA20    RES   1                      2110
00025    23 $IA21    RES   1                      2115
00026    24 $IA22    RES   1                      2120
00027    25 $IA23    RES   1                      2130
00030    26 $IA24    RES   1                      2140
00031    27 $IA25    RES   1                      2150
00032    28 $IA26    RES   1                      2160
00033    29 $IA27    RES   1                      2170
00034    30 $IA28    RES   1                      2180
00035    31 $IA29    RES   1                      2190
00036    32 $IA30    RES   1                      2200
00037    33 $IA31    RES   1                      2210
00040    34 $IA32    RES   1                      2220
00041    35 $IA33    RES   1                      2230
```

```
00042                    36 $IA34     RES  1                                    2240
00043                    37 $IA35     RES  1                                    2250
00044                    38 $IA36     RES  1                                    2260
00045                    39 $IA37     RES  1                                    2270
00046                    40 $IA38     RES  1                                    2280
00047                    41 $IA39     RES  1                                    2290
00050                    42 $IA40     RES  1                                    2300
00051                    43 $IA41     RES  1                                    2310
00052                    44 $IA42     RES  1                                    2320
00053                    45 $IA43     RES  1                                    2330
00054                    46 $IA44     RES  1                                    2340
00055                    47 $IA45     RES  1                                    2350
00056                    48 $IA46     RES  1                                    2360
00057                    49 $IA47     RES  1                                    2370
00060                    50 $IA48     RES  1                                    2380
00061                    51 $IA49     RES  1                                    2390
00062                    52 $IA50     RES  1                                    2400
00063                    53 $IA51     RES  1                                    2410
00064                    54 $IA52     RES  1                                    2420
00065                    55 $IA53     RES  1                                    2430
         00000001        56 $INPUTAR  EQU  IA1                                  2440
         00000063        57 $IAEND    EQU  INPUTAR+50                           2450
                         58 *                                                   2510
                         59 *                                                   2520
                         60 *                                                   2530
00066                    61 $OUTAREA  RES  33          OUTPUT AREA OF 132 CHARACTERS 2540
                         62 *                                                   2550
         00000067        63 $OAR2     EQU  OUTAREA+1                            2560
         00000070        64 $OAR3     EQU  OUTAREA+2                            2570
         00000071        65 $OAR4     EQU  OUTAREA+3                            2580
         00000072        66 $OAR5     EQU  OUTAREA+4                            2590
         00000073        67 $OAR6     EQU  OUTAREA+5                            2600
         00000074        68 $OAR7     EQU  OUTAREA+6                            2610
         00000075        69 $OAR8     EQU  OUTAREA+7                            2620
         00000076        70 $OAR9     EQU  OUTAREA+8                            2630
         00000077        71 $OAR10    EQU  OUTAREA+9                            2640
         00000100        72 $OAR11    EQU  OUTAREA+10                           2650
         00000101        73 $OAR12    EQU  OUTAREA+11                           2660
         00000102        74 $OAR13    EQU  OUTAREA+12                           2670
         00000103        75 $OAR14    EQU  OUTAREA+13                           2680
         00000104        76 $OAR15    EQU  OUTAREA+14                           2690
         00000105        77 $OAR16    EQU  OUTAREA+15                           2700
         00000106        78 $OAR17    EQU  OUTAREA+16                           2710
         00000107        79 $OAR18    EQU  OUTAREA+17                           2720
         00000110        80 $OAR19    EQU  OUTAREA+18                           2730
         00000111        81 $OAR20    EQU  OUTAREA+19                           2740
         00000112        82 $OAR21    EQU  OUTAREA+20                           2760
         00000113        83 $OAR22    EQU  OUTAREA+21                           2770
         00000114        84 $OAR23    EQU  OUTAREA+22                           2780
         00000115        85 $OAR24    EQU  OUTAREA+23                           2790
         00000116        86 $OAR25    EQU  OUTAREA+24                           2800
         00000117        87 $OAR26    EQU  OUTAREA+25                           2810
         00000120        88 $OAR27    EQU  OUTAREA+26                           2820
         00000121        89 $OAR28    EQU  OUTAREA+27                           2830
         00000122        90 $OAR29    EQU  OUTAREA+28                           2840
```

```
        00000123  $OAR30     EQU  OUTAREA+29                                            2850
        00000124  $OAR31     EQU  OUTAREA+30                                            2860
        00000125  $OAR32     EQU  OUTAREA+31                                            2870
        00000126  $OAR33     EQU  OUTAREA+32       LAST WORD OF OUTPUT AREA             2880
        00000127  $OUTAREAL  EQU  OUTAREA+33       LAST+1 WORD OF OUTPUT AREA           2890
                  *                           PARAMETER REGION                          3000
                  *                                                                     3010
00127             $PARAMET   RES  570          STORAGE AREA FOR PARAMETERS              3020
        00000127  $PARAHED   EQU  PARAMET          START OF RUN HEADING AREA            3030
        00000153  $PARAMA    EQU  PARAMET+20       START OF PARAMETER A AREA            3040
        00000154  $PARA2     EQU  PARAMET+21                                            3050
        00000155  $PARA3     EQU  PARAMET+22                                            3060
        00000157  $PARA5     EQU  PARAMET+24                                            3070
        00000160  $PARA6     EQU  PARAMET+25                                            3080
        00000161  $PARA7     EQU  PARAMET+26                                            3090
        00001157  $RUNIND    EQU  PARAMET+536      RUN OPTION INDICATORS                3100
        00000156  $PARA4     EQU  PARAMET+23                                            3110
        00000177  $PARHOL    EQU  PARAMET+40       START OF HOLIDAY AREA-TYPE CODE      3120
        00000200  $PARFD     EQU  PARAMET+41            1ST HOLIDAY DAY                 3130
        00000201  $PARFM     EQU  PARAMET+42            1ST HOLIDAY MONTH               3140
        00000202  $PARFY     EQU  PARAMET+43            1ST HOLIDAY YEAR                3150
        00001157  $PARAMB    EQU  PARAMET+536      START OF PARAMETER B AREA            3160
        00001160  $RUNIDA    EQU  PARAMET+537      1ST WD RUN ID                        3170
        00001161  $RUNIDB    EQU  PARAMET+538      2ND WD RUN ID                        3180
        00001203  $PARNA     EQU  PARAMET+556      NUMBER OF ACTIVITIES                 3185
        00001204  $PARNE     EQU  PARAMET+557      NEMBER OF EVENTS                     3186
        00001221  $PAREND    EQU  PARAMET+570      LOC OF LAST PARAMETER + 1            3190
        00000401  $LASTPRL   EQU  PARAMET+170                                           3200
        00000200  $YMDH      EQU  PARFD            1ST WD OF 3WD PER HOLI IN PARAM AREA 3210
                             END                                                        9999
```

DSRP COMMON DATA AND STORAGE

```
                  *                           COMMON STORAGE REGION                     1000
                  *                           PERMENANT DATA REGION                     1010
00000                        RORG 0                                                     1020
00000   00000000  $ZERO      DATA  0                                                    1025
00001   20000000  $BCDPL     DATA 020000000          BCD PLUS                           1030
00002   00000033  $BCDDP     DATA 000000033          BCD DECIMAL POINT                  1040
00003   01000000  $BCD1000   DATA '1000'             BCD 1000                           1044
00004   02000000  $BCD2000   DATA '2000'             BCD 2000                           1048
00005   05000000  $BCD5000   DATA '5000'             BCD 5000                           1050
00006   60606060  $BCDBLK    DATA '    '             BCD BLANK WORD                     1060
00007   00000054  $BCDASS    DATA '*'                BCD ASTERISK                       1070
00010   00000060  $BLK1CH   DATA ' '                 BCD 1 CHARACTER BLANK              1080
00011   00000030  $H         DATA 'H'                BCD 1 CHARACTER H                  1090
00012   00000024  $D         DATA 'D'                BCD 1 CHARACTER D                  1100
00013   00000066  $W         DATA 'W'                BCD 1 CHARACTER W                  1110
```

```
00014  00000044              16 $M        DATA *M*                    BCD 1 CHARACTER M                  1120
00015  00000023              17 $BCDC     DATA *C*                    BCD 1 CHARACTER C                  1130
00016  00000045              18 $BCDN     DATA *N*                    BCD 1 CHARACTER N                  1140
00017  00000070              19 $BCDY     DATA *Y*                    BCD 1 CHARACTER Y                  1150
00020  00006060              20 $BLKDAY   DATA *  *                    BCD 2 CHARACTER BLANK              1160
00021  60604560              21 $NAT3     DATA *  N *                 BCD N IN 3RD CHARACTER POSITION    1170
00022                        22 $DAYWK0   RES  1                      DAYS OF THE WEEK                   1180
00023  00006264              23 $DAYSWK   DATA *SU*,*M0*,*TU*,*WE*,*TH*,*FR*,*SA*                         1190
00024  00004446
00025  00006364
00026  00006625
00027  00006330
00030  00002651
00031  00006221
00032                        24 $M0NYR0   RES  1                      M0NTHS OF THE YEAR                 1200
00033  00412145              25 $M0NYR    DATA *JAN*,*FEB*,*MAR*,*APR*,*MAY*,*JUN*,*JUL*                 1210
00034  00262522
00035  00442151
00036  00214751
00037  00442170
00040  00416445
00041  00416443
00042  00216427              26           DATA *AUG*,*SEP*,*0CT*,*N0V*,*DEC*                             1220
00043  00622547
00044  00462363
00045  00454665
00046  00242523
                            27 *                          MASKS                                         1230
00047  77000000              28 $MASK1    DATA 077000000                                                 1240
00050  77770000              29 $MASK12   DATA 077770000                                                 1250
00051  77777700              30 $MASK123  DATA  077777700                                                1255
00052  00770000              31 $MASK2    DATA 000770000                                                 1260
00053  00777700              32 $MASK23   DATA 000777700                                                 1270
00054  00777777              33 $MAS234   DATA 000777777                                                 1280
00055  00007700              34 $MASK3    DATA 000007700                                                 1290
00056  00007777              35 $MASK34   DATA 000007777                                                 1300
00057  00000077              36 $MASK4    DATA 000000077                                                 1310
00060  37777777              37 $MASKN    DATA 037777777                                                 1320
00061  00003777              38 $MASK11R  DATA 000003777                                                 1340
00062  40007777              39 $MASK11L  DATA 040007777                                                 1350
00063  00000017              40 $MASK9    DATA 000000017                                                 1360
00064  00037777              41 $MASKAD   DATA  037777                                                   1370
00065  40000000              42 $MASKNG   DATA 040000000                                                 1380
00066  00010000              43 $0NE11    DATA 000010000                 0NE AT BM 0F 11                 1390
                            44 *                        INSTRUCTION REGION                              1400
00067  2 00 0 00244          45 $DY0FM1   HLT  DAY0FF-1,2                                                1410
00070  2 00 0 00453          46 $YMDHM1   HLT  YMDH-1,2                                                  1422
00071  2 00 0 00455          47 $YMDHP1   HLT  YMDH+1,2                                                  1424
00072  2 00 0 00454          48 $YMDH0    HLT  YMDH,2                                                    1426
00073  2 00 0 00000          49 $TAG      HLT  0,2                                                       1430
00074                        50 $MAXL0C   RES  1                                                         1440
00075  0 00 0 01433          51 $L0CPRB   HLT  PARAMB                                                    1452
00076  0 00 0 00455          52 $PARFMLC  HLT  PARFM                                                     1454
00077  0 00 0 00454          53 $PARFDLC  HLT  PARFD                                                     1456
00100  00000002              54 $RPUN0    DATA 2                                                         1457
```

```
00101                    55 $TVM1      RES  1                    NO TAG TIME VECTOR - 1                1460
00102                    56 $TVM2      RES  1                    NO TAG TIME VECTOR - 2                1470
00103                    57 $MNOFF     RES  1                                                          1480
00104                    58 $TOFF      RES  1                                                          1490
00105                    59 $ABOFF     RES  1                                                          1500
00106  2 00 0 00402      60 $HOLDM1    HLT  HOLBIN-1,2                                                 1505
00107  0 35 0 00454      61 $HOLINIT   STA  PARHOL+1             DUMMY FOR 1ST WD CAL HOLDAS           1510
                         62 *                          REGULAR SCALAR TEMP. REGION                   1520
00110  00000035          63 $ACTNUMWD  DATA 29                    NO) WDS) IN ACTIVITY RECORD           1530
00111                    64 $CHINDEV   RES  1                                                          1540
00112                    65 $CURHOL    RES  1                                                          1550
00113                    66 $DUR1      RES  1                    DURATION DAY - BCD                  1560
00114                    67 $DUR2      RES  1                    DURATION MONTH - BCD                1570
00115                    68 $DUR3      RES  1                    DURATION YEAR OR BASIC UNIT-BCD     1580
00116                    69 $DS        RES  1                    DURATION OF SCHEDULE                1590
00117                    70 $ENDFILE   RES  1                                                          1595
00120  00000000          71 $ERI       DATA 0                    FOR STORAGE OF                      1600
00121  00000000          72 $ERJ       DATA 0                    I,J, AND DUPE CODE                  1610
00122  00000000          73 $ERD       DATA 0                    FOR ERROR PRINTOUT                  1620
00123                    74 $ENDAC     RES  1                    OUTAREA + ACTNUMWD                  1630
00124                    75 $FIRPAS    RES  1                    FIRST PASS INDICTOR                 1640
00125                    76 $LINCNT    RES  1                    LINE COUNTER                        1650
00126                    77 $LINEC     RES   1                                                       1655
00127                    78 $LOSTDYS   RES   1                                                       1658
00130                    79 $MAJOR     RES  1                    MAJOR ACTIVITY INDICTOR             1660
00131                    80 $NA        RES  1                    NUMBER OF ACTIVITIES                1670
00132                    81 $NE        RES  1                    NUMBER OF EVENTS                    1680
00133                    82 $NEP2      RES  1                    NUMBER OF EVENTS + 2                1685
00134                    83 $NEGACWD   RES  1                    NEG. OF NO. WDS IN ACT. RES.        1688
00135                    84 $NDYOFF    RES  1                    NO. DAYS-OFF PER WK                 1690
00136                    85 $NWD       RES  1                    NUMBER OF WORKING DAYS              1700
00137                    86 $NWDPW     RES  1                    NUMBER OF WORKING DAYS PER WEEK     1710
00140                    87 $NWHPD     RES  1                    NUMBER OF WORKING HRS PER DAY       1720
00141                    88 $SYRTYP    RES   1                    START YEAR TYPE                    1726
00142  00000000          89 $PRNTCODE  DATA   0                                                       1730
00143  00001072          90 $PARNUMWD  DATA 570                                                       1735
00144  00000000          91 $RGINDEV   DATA   0                                                       1740
00145                    92 $RUNIDAS   RES   1                                                       1745
00146                    93 $RUNIDBS   RES   1                                                       1748
00147                    94 $SDAY      RES  1                    START DAY                           1750
00150                    95 $SMONTH    RES  1                    START MONTH                         1760
00151                    96 $SDTIY     RES   1                    START DAY TYPE                     1764
00152                    97 $SYEAR     RES  1                    START YEAR                          1770
00153                    98 $TYSDAY    RES  1                    TYPE DAY PROJ. STARTS ON            1775
00154                    99 $TCOSA     RES  1                    DOUBLE PRECISION                    1780
00155                   100 $TCOSB     RES  1                    TOTAL COST                          1790
00156                   101 $UNITS     RES  1                    BASIC UNITS TYPE INDICATOR          1800
00157                   102 $DAY       RES  1                                                        1810
00160                   103 $MONTH     RES  1                                                        1820
00161                   104 $YEAR      RES  1                                                        1830
00162                   105 $YMDFOR    RES  1                    YMD FORMAT INDICATOR                1840
     00000110           106 $NWREC     EQU  ACTNUMWD                                                 1850
00163                   107 $WB1       RES  1                                                        1860
00164                   108 $WB2       RES  1                                                        1870
00165                   109 $WB3       RES  1                                                        1880
```

```
00166  110 $WB4      RES  1                                                    1890
00167  111 $WB5      RES  1                                                    1900
       112 *                                                                  2000
00170  113 $NWDY     RES   15   REG. VECTOR TEMP. REGION                       2002
00207  114 $NHY      RES 15           NUMBER OF WORKING DAYS PER YEAR          2004
00226  115 $HOLORG   RES   15         NUMBER OF HOLIDAYS PER YEAR              2006
00245  116 $DAYOFF   RES  7          HOLIDAY ORGINS                           2010
       117 *                        DAY-OFF VECTOR                           2060
00254  118 $IA0      RES  1    INPUT REGION                                   2070
00255  119 $IA1      RES  1           CODE                                    2072
00256  120 $IA2      RES  1           I                                       2074
00257  121 $IA3      RES  1           J                                       2076
00260  122 $IA4      RES  1           DUP                                     2078
00261  123 $IA5      RES  1           B                                       2080
00262  124 $IA6      RES  1           A                                       2082
00263  125 $IA7      RES  1           COSTB - 1                               2084
00264  126 $IA8      RES  1           COSTB - 2                               2086
00265  127 $IA9      RES  1           INDICATORS                              2088
00266  128 $IA10     RES  1           MILESTONE - 1ST WORD                    2090
00267  129 $IA11     RES  1           MILESTONE - 2ND WORD                    2092
00270  130 $IA12     RES  1           COST CENTER                             2094
00271  131 $IA13     RES  1           CODE                                    2096
00272  132 $IA14     RES  1           RESPONSIBILITY                          2098
00273  133 $IA15     RES  1                                                   2100
00274  134 $IA16     RES  1           ACTIVITY DESCRIPTION                    2102
00275  135 $IA17     RES  1           ACTIVITY DESCRIPTION                    2104
00276  136 $IA18     RES  1           ACTIVITY DESCRIPTION                    2106
00277  137 $IA19     RES  1           ACTIVITY DESCRIPTION                    2108
00300  138 $IA20     RES  1           ACTIVITY DESCRIPTION                    2110
00301  139 $IA21     RES  1           ACTIVITY DESCRIPTION                    2115
00302  140 $IA22     RES  1           ACTIVITY DESCRIPTION                    2120
00303  141 $IA23     RES  1           M                                       2130
00304  142 $IA24     RES  1           N                                       2140
00305  143 $IA25     RES  1           K                                       2150
00306  144 $IA26     RES  1           COSTA- 1                                2160
00307  145 $IA27     RES  1           COSTA- 2                                2170
00310  146 $IA28     RES  1           MILESTONE FLAG -BCD                     2180
00311  147 $IA29     RES  1           DURATION-BINARY                         2190
00312  148 $IA30     RES  1           COST                                    2200
00313  149 $IA31     RES  1           FREE FLOAT                              2210
00314  150 $IA32     RES  1           TOTAL FLOAT                             2220
00315  151 $IA33     RES  1           EARLY START -BASIC UNITS                2230
00316  152 $IA34     RES  1           EARLY FINISH - BASIC UNITS              2240
00317  153 $IA35     RES  1           LATE START - BASIC UNITS                2250
00320  154 $IA36     RES  1           LATE FINISH - BASIC UNITS               2260
00321  155 $IA37     RES  1           CRITICAL INDICATOR - BCD                2270
00322  156 $IA38     RES  1           EARLY START - DAY UNITS                 2280
00323  157 $IA39     RES  1           EARLY FINISH - DAY UNITS                2290
00324  158 $IA40     RES  1           LATE START - DAY UNITS                  2300
00325  159 $IA41     RES  1           LATE FINISH - DAY UNITS                 2310
00326  160 $IA42     RES  1           EARLY START DAY                         2320
00327  161 $IA43     RES  1           EARLY FINISH DAY                        2330
00330  162 $IA44     RES  1           LATE START DAY                          2340
00331  163 $IA45     RES  1           LATE FINISH DAY                         2350
00332  164 $IA46     RES  1           EARLY START MONTH                       2360
```

```
00333                  165 $IA47      RES  1                    EARLY FINISH MONTH                2370
00334                  166 $IA48      RES  1                    LATE START MONTH                  2380
00335                  167 $IA49      RES  1                    LATE FINISH MONTH                 2390
00336                  168 $IA50      RES  1                    EARLY START YEAR                  2400
00337                  169 $IA51      RES  1                    EARLY FINISH YEAR                 2410
00340                  170 $IA52      RES  1                    LATE START YEAR                   2420
00341                  171 $IA53      RES  1                    LATE FINISH YEAR                  2430
       00000255        172 $INPUTAR   EQU  IA1                                                    2440
       00000337        173 $IAEND     EQU  INPUTAR+50                                           2450
                       174 *                                                                     2510
                       175 *                                                                     2520
                       176 *                                                                     2530
00342                  177 $OUTAREA   RES  33             OUTPUT AREA OF 132 CHARACTERS          2540
                       178 *                                                                     2550
       00000343        179 $OAR2      EQU  OUTAREA+1                                            2560
       00000344        180 $OAR3      EQU  OUTAREA+2                                            2570
       00000345        181 $OAR4      EQU  OUTAREA+3                                            2580
       00000346        182 $OAR5      EQU  OUTAREA+4                                            2590
       00000347        183 $OAR6      EQU  OUTAREA+5                                            2600
       00000350        184 $OAR7      EQU  OUTAREA+6                                            2610
       00000351        185 $OAR8      EQU  OUTAREA+7                                            2620
       00000352        186 $OAR9      EQU  OUTAREA+8                                            2630
       00000353        187 $OAR10     EQU  OUTAREA+9                                            2640
       00000354        188 $OAR11     EQU  OUTAREA+10                                           2650
       00000355        189 $OAR12     EQU  OUTAREA+11                                           2660
       00000356        190 $OAR13     EQU  OUTAREA+12                                           2670
       00000357        191 $OAR14     EQU  OUTAREA+13                                           2680
       00000360        192 $OAR15     EQU  OUTAREA+14                                           2690
       00000361        193 $OAR16     EQU  OUTAREA+15                                           2700
       00000362        194 $OAR17     EQU  OUTAREA+16                                           2710
       00000363        195 $OAR18     EQU  OUTAREA+17                                           2720
       00000364        196 $OAR19     EQU  OUTAREA+18                                           2730
       00000365        197 $OAR20     EQU  OUTAREA+19                                           2740
       00000366        198 $OAR21     EQU  OUTAREA+20                                           2760
       00000367        199 $OAR22     EQU  OUTAREA+21                                           2770
       00000370        200 $OAR23     EQU  OUTAREA+22                                           2780
       00000371        201 $OAR24     EQU  OUTAREA+23                                           2790
       00000372        202 $OAR25     EQU  OUTAREA+24                                           2800
       00000373        203 $OAR26     EQU  OUTAREA+25                                           2810
       00000374        204 $OAR27     EQU  OUTAREA+26                                           2820
       00000375        205 $OAR28     EQU  OUTAREA+27                                           2830
       00000376        206 $OAR29     EQU  OUTAREA+28                                           2840
       00000377        207 $OAR30     EQU  OUTAREA+29                                           2850
       00000400        208 $OAR31     EQU  OUTAREA+30                                           2860
       00000401        209 $OAR32     EQU  OUTAREA+31                                           2870
       00000402        210 $OAR33     EQU  OUTAREA+32     LAST WORD OF OUTPUT AREA              2880
       00000403        211 $OUTAREAL  EQU  OUTAREA+33     LAST+1 WORD OF OUTPUT AREA            2890
                       212 *                        PARAMETER REGION                          3000
                       213 *                                                                  3010
00403                  214 $PARAMET   RES  570            STORAGE AREA FOR PARAMETERS         3020
       00000403        215 $PARAHED   EQU  PARAMET        START OF RUN HEADING AREA           3030
       00000427        216 $PARAMA    EQU  PARAMET+20     START OF PARAMETER A AREA           3040
       00000430        217 $PARA2     EQU  PARAMET+21                                         3050
       00000431        218 $PARA3     EQU  PARAMET+22                                         3060
       00000433        219 $PARA5     EQU  PARAMET+24                                         3070
```

```
         00000434          220 $PARA6     EQU   PARAMET+25                                              3080
         00000435          221 $PARA7     EQU   PARAMET+26                                              3090
         00001433          222 $RUNIND    EQU   PARAMET+536      RUN OPTION INDICATORS                  3100
         00000432          223 $PARA4     EQU   PARAMET+23                                              3110
         00000453          224 $PARHOL    EQU   PARAMET+40       START OF HOLIDAY AREA-TYPE CODE        3120
         00000454          225 $PARFD     EQU   PARAMET+41            1ST HOLIDAY DAY                   3130
         00000455          226 $PARFM     EQU   PARAMET+42            1ST HOLIDAY MONTH                 3140
         00000456          227 $PARFY     EQU   PARAMET+43            1ST HOLIDAY YEAR                  3150
         00001433          228 $PARAMB    EQU   PARAMET+536      START OF PARAMETER B AREA              3160
         00001457          229 $PARNA     EQU    PARAMB+20                                              3165
         00001434          230 $RUNIDA    EQU   PARAMET+537      1ST WD RUN ID                          3170
         00001435          231 $RUNIDB    EQU   PARAMET+538      2ND WD RUN ID                          3180
         00001475          232 $PAREND    EQU   PARAMET+570      LOC OF LAST PARAMETER + 1              3190
01475    0 00 0 00655      233 $LASTPRL   HLT   PARAMET+170                                             3200
         00000403          234 $HOLBIN    EQU   PARAMET                                                 3210
         00000454          235 $YMDH     EQU    PARFD                                                   3220
         00000167          236 $NWDYM1    EQU    NWDY-1                                                 3230
         00000206          237 $NHYM1     EQU    NHY-1                                                  3240
         00000225          238 $HOLOM1    EQU    HOLORG-1                                               3250
         00001457          239 $BCDNA    EQU    PARAMET+556                                             3270
                           240            END                                                           9999
```

PEP COMMON DATA AND STORAGE

```
                           1 *                            COMMON STORAGE REGION                1000
                           2 *                            PERMENANT DATA REGION                1010
00000                      3            RORG 0                                                 1020
00000   20000000           4 $BCDPL     DATA 020000000         BCD PLUS                        1030
00001   20606060           5 $BCDPLBK   DATA  '+   '                                           1031
00002   40606060           6 $BCDNEG    DATA  '-   '                                           1034
00003   00000033           7 $BCDDP     DATA 000000033         BCD DECIMAL POINT               1040
00004   01000000           8 $BCD1000   DATA '1000'            BCD 1000                        1044
00005   02000000           9 $BCD2000   DATA '2000'            BCD 2000                        1048
00006   05000000          10 $BCD5000   DATA '5000'            BCD 5000                        1050
00007   06000000          11 $BCD6000   DATA  '6000'                                           1051
00010   00000001          12 $ONE       DATA  1                                                1052
00011   00000000          13 $ZERO      DATA  0                                                1053
00012   00000002          14 $TWO       DATA  2                                                1054
00013   77777777          15 $MONE      DATA  -1                                               1055
00014   37777777          16 $DACTI     DATA  037777777                                        1056
00015   60606060          17 $BCDBLK    DATA '    '            BCD BLANK WORD                  1060
00016   00000054          18 $BCDASS    DATA '*'               BCD ASTERISK                    1070
00017   00000060          19 $BLK1CH   DATA ' '                BCD 1 CHARACTER BLANK           1080
00020   00000030          20 $H         DATA 'H'               BCD 1 CHARACTER H               1090
00021   00000024          21 $D         DATA 'D'               BCD 1 CHARACTER D               1100
00022   00000066          22 $W         DATA 'W'               BCD 1 CHARACTER W               1110
00023   00000023          23 $BCDC      DATA 'C'               BCD 1 CHARACTER C               1130
00024   00000045          24 $BCDN      DATA 'N'               BCD 1 CHARACTER N               1140
```

```
00025  00000070          25 $BCDY     DATA 'Y'                  BCD 1 CHARACTER Y                    1150
00026  00006060          26 $BLKDAY   DATA '  '                  BCD 2 CHARACTER BLANK                1160
00027  60604560          27 $NAT3     DATA '  N '                BCD N IN 3RD CHARACTER POSITION      1170
00030                    28 $DAYWK0   RES  1                     DAYS OF THE WEEK                     1180
00031  00006264          29 $DAYSWK   DATA 'SU','M0','TU','WE','TH','FR','SA'                         1190
00032  00004446
00033  00006364
00034  00006625
00035  00006330
00036  00002651
00037  00006221
00040                    30 $M0NYR0   RES  1                     M0NTHS OF THE YEAR                   1200
00041  00412145          31 $M0NYR    DATA 'JAN','FEB','MAR','APR','MAY','JUN','JUL'                   1210
00042  00262522
00043  00442151
00044  00214751
00045  00442170
00046  00416445
00047  00416443
00050  00216427          32           DATA 'AUG','SEP','0CT','N0V','DEC'                               1220
00051  00622547
00052  00462363
00053  00454665
00054  00242523
                         33 *                        MASKS                                            1230
00055  77000000          34 $MASK1    DATA 077000000                                                   1240
00056  77770000          35 $MASK12   DATA 077770000                                                   1250
00057  77777700          36 $MASK123  DATA  077777700                                                  1255
00060  00770000          37 $MASK2    DATA 000770000                                                   1260
00061  00777700          38 $MASK23   DATA 000777700                                                   1270
00062  00777777          39 $MAS234   DATA 000777777                                                   1280
       00000062          40 $MASK234  EQU   MAS234                                                     1281
00063  00007700          41 $MASK3    DATA 000007700                                                   1290
00064  00007777          42 $MASK34   DATA 000007777                                                   1300
00065  00000077          43 $MASK4    DATA 000000077                                                   1310
00066  77777777          44 $MASK1234 DATA   077777777                                                 1311
00067  37777777          45 $MASKN    DATA 037777777                                                   1320
00070  00003777          46 $MASK11R  DATA 000003777                                                   1340
00071  40007777          47 $MASK11L  DATA 040007777                                                   1350
00072  00000017          48 $MASK9    DATA 000000017                                                   1360
00073  00037777          49 $MASKAD   DATA  037777                                                     1370
00074  00077777          50 $MASKADR  DATA 000077777                                                   1371
00075  40000000          51 $MASKNG   DATA 040000000                                                   1380
00076  00010000          52 $0NE11    DATA 000010000          0NE AT BM 0F 11                          1390
00077                    53 $NWSEG2   RES   1                NO. 0F WORDS IN SEG. TWO                 1391
00100  00000041          54 $RLNUMWD  DATA  33        NUMBER 0F WDS. IN DSRP OUTPUT REC.               1392
00101  00001005          55 $NWPARF   DATA  517       NUMBER 0F WDS. IN PARAMETER FILE REGION          1393
00102  77776773          56 $NNWPARF  DATA  -517      NEGATIVE 0F NWPARF                               1394
00103  77777022          57 $NNWHEP   DATA  -494      NEG. 0F NO. WDS. 1ST HOL. TO END PAR. REG.       1395
00104  00000101          58 $NWBAR    DATA  65        NO. WDS. IN BCD ACTIVITY RECORD                  1396
00105  77777677          59 $NWBARN   DATA -65        NEG. 0F NWBAR                                    1397
00106  00000001          60 $RPUN0    DATA  1                 REPORT UNIT NUMBER                       1398
                         61 *                        INSTRUCTION REGION                               1400
00107  2 00 0 00361      62 $DY0FM1   HLT   DAY0FF-1,2                                                 1410
00110  2 35 0 00413      63 $UC1ST0   STA    USCR1+18,2                                                1422
```

```
00111  2 35 0 00435          64 $UC2ST0   STA   USCR2+18,2                                                    1424
00112  2 C0 0 00000          65 $TAG      HLT   0,2                                                          1430
00113  57777777              66 $TAGDEL   DATA  057777777      DELETE TAG MASK
00114                        67 $MN0FF    RES   1                    WITH TAB                                1440
00115                        68 $DRTS0FF  RES   1                    WITH TAB                                1445
00116                        69 $T0FF     RES   1                    WITH TAB                                1448
00117                        70 $ABRACM   RES   1
00120                        71 $ABRAC    RES   1              TAG   0RG. MISC. WD. ABR. ACT. REC.           1450
00121                        72 $ABRACM1  RES   1              TAG   0RG.-1 MISC. WD. ABR ACT. REC.        1455
00122                        73 $ABRACI   RES   1              TAG   0RG. I WD. 0F ABRV. ACT. REC.          1460
00123                        74 $ABRACIM1 RES   1              TAG   0RG.-1 I WD 0F ABRV. ACT REC.         1465
00124                        75 $ABRACJ   RES   1              TAG   0RG. J WD. 0F ABRV. ACT. REC.         1470
00125                        76 $ABRACJM1 RES   1              TAG   0RG.-1 J WD 0F ABRV. ACT REC.        1475
00126                        77 $ABRACIL  RES   1              TAG   LAST+1 I WD 0F ABR. ACT. REC.         1480
00127                        78 $ABRACJL  RES   1              TAG   LAST+1 J WD 0F ABR. ACT. REC.         1485
00130                        79 $NXTMSTA  RES   1                    LAST+1 MISC. WD. (INC. DELETS)        1490
00131                        80 $SEQ      RES   1                    START SELECTED INF0,0RG. SEQ          1492
00132                        81 $THREDIN  RES   1                    START SORT TAPE OUTPUT AREA           1494
00133                        82 $SELECT   RES   1                BEGINNING 0F SELECT AREA N0 TAG           1495
00134                        83 $SELECTL  RES   1              TAG   LAST+1 SELECTE INF0 IN T0P0L SEQ      1496
00135                        84 $SEQL     RES   1              TAG   LAST+1 SELECTED INF0, 0RG SEQ         1498
00136                        85 $SEGEND   RES   1                    LAST+1 SELECTED INF0, 0RG SEQ         1500
                             86 *                             REGULAR SCALAR TEMP. REGION                  1520
00137                        87 $ACTK     RES   1                    N0. LAST ACT.+1 0F THREAD             1522
00140                        88 $ACTH     RES   1                    N0. 0F STARTING ACT. 0F NETW0RK       1524
00141  0 00 0 00000          89 $CARDSTI  HLT   0           INIT L0C T0 ST0 ACT. CRD CHANGES (TAG)          1526
00142  0 00 0 00000          90 $CARDSTMX HLT   0           MAX. VALUE 0F CARDSTI                           1527
00143                        91 $BRANCHB3 RES   1            INPUT DEVICE INDICATOR - BINARY                1530
00144  00000000              92 $CHINDEV  DATA  0                                                           1540
00145                        93 $CURH0L   RES   1                                                           1550
00146                        94 $CURACT   RES   1                    CURRENT ACTIVITY                      1555
00147                        95 $DUR1     RES   1                  DURATION DAY - BCD                      1560
00150                        96 $DUR2     RES   1                  DURATION M0NTH - BCD                    1570
00151                        97 $DUR3     RES   1                  DURATION YEAR 0R BASIC UNIT-BCD         1580
00152                        98 $DUR      RES   1                                                          1590
00153                        99 $ENDFILE  RES   1                                                          1595
00154                       100 $DATBASU  RES   1                                                          1596
00155                       101 $DA       RES   1                    ACTUAL DURATION                       1597
00156                       102 $DIFA     RES   1                    ACTUAL DIFFERENTIAL DURATION          1598
00157  00000000             103 $ERI      DATA  0                  F0R STORAGE 0F                          1600
00160  00000000             104 $ERJ      DATA  0                  I,J, AND DUPE C0DE                      1610
00161  00000000             105 $ERD      DATA  0                  F0R ERR0R PRINT0UT                      1620
00162                       106 $EXPFDY   RES   1                    EXP. PR0J. FIN. DAY                   1630
00163                       107 $EXPFMN   RES   1                    EXP. PR0J. FIN. M0NTH                 1632
00164                       108 $EXPFYR   RES   1                    EXP. PR0J. FIN. YEAR                  1634
00165                       109 $EFFDAY   RES   1            EFFECTIVE DATE - 3 PART , BCD                 1636
00166                       110 $EFFM0N   RES   1                                                          1637
00167                       111 $EFFYER   RES   1                                                          1638
00170                       112 $FPI      RES   1            FIRST PR0GRESS RUN INDICATOR                  1639
00171                       113 $KMAT     RES   1                                                          1640
00172                       114 $K        RES   1                                                          1643
00173                       115 $M        RES   1                                                          1644
00174                       116 $N        RES   1                                                          1645
00175                       117 $LINCNT   RES   1                    LINE C0UNTER                          1650
00176  00000000             118 $L0STDYS  DATA  0
```

```
00177  00000000          119 $LINEC     DATA  0                        MAJOR ACTIVITY INDICTOR                1660
00200                    120 $MAJOR     RES   1                         MAX. LOC. IN COMPUTER                 1662
00201                    121 $MAXLOC    RES    1                        MAX. LENGTH ABR. ACT. REC.            1664
00202                    122 $MAXLAAR   RES    1
       00000202          123 $MAXABR    EQU    MAXLAAR
00203                    124 $MACC      RES    1                         M ACCUMULATOR                         1666
00204                    125 $NA        RES   1                        NUMBER OF ACTIVITIES                   1670
00205                    126 $NE        RES   1                        NUMBER OF EVENTS                       1680
00206                    127 $NANRM     RES    1                         NEGATIVE OF NA                        1685
       00000206          128 $NGNA      EQU    NANRM
00207                    129 $NUMAC     RES    1        NO. ACTS. INCLUDING ADDS AND DELETES                  1686
00210                    130 $NUMACNG   RES    1        NEGATIVE OF NUMAC                                     1687
00211                    131 $NDYOFF    RES   1                         NO. DAYS-OFF PER WK                   1690
00212                    132 $NWD       RES   1                        NUMBER OF WORKING DAYS                 1700
00213                    133 $NWDPW     RES   1                        NUMBER OF WORKING DAYS PER WEEK        1710
00214                    134 $NWHPD     RES   1                        NUMBER OF WORKING HRS PER DAY          1720
00215                    135 $SYRTYP    RES    1                         START YEAR TYPE                       1726
00216  00000000          136 $PAGEC     DATA  0
00217                    137 $PFPRINT   RES    1        PAST-FUTURE ACTIVITY PRINT INDICATOR                  1727
00220                    138 $PSTATUS   RES    1                                                              1729
00221  00000000          139 $PRNTCODE  DATA     0                                                             1730
00222                    140 $SCHPFDY   RES    1                         SCHD. PROJ. FIN. DAY                  1731
00223                    141 $SCHPFMN   RES    1                         SCHD. PROJ. FIN. MONTH                1732
00224                    142 $SCHPFYR   RES    1                         SCHD. PROJ. FIN. YEAR                 1733
00225                    143 $STATWD1   RES    1                         PROJ. STATUS WORD 1                   1734
00226                    144 $STATWD2   RES    1                         PROJ. STATUS WORD2                    1735
00227                    145 $SCR       RES    1                         SCHD. CRIT. IND. IN BINARY            1736
00230                    146 $STASG1    RES    1                         STORAGE AVAILABLE IN SEG. 1           1737
00231                    147 $STRING    RES    1                         NO.REC.IN INPUT STRING                1738
00232                    148 $STRINGM   RES    1                         NEGATIVE OF STRING                    1739
00233  00000000          149 $RGINDEV   DATA     0                                                             1740
00234                    150 $RUNID     RES    1        RUN ID                                                1744
00235                    151 $RUNIDP1   RES    1                                                              1748
00236                    152 $SDAY      RES   1                        START DAY                              1750
00237                    153 $SMONTH    RES   1                        START MONTH                            1760
00240                    154 $SYEAR     RES   1                        START YEAR                             1763
00241                    155 $SDTIY     RES    1                         START DAY TYPE                        1764
00242                    156 $TYSDAY   RES    1                        TYPE DAY PROJ. STARTS ON               1775
00243                    157 $TEMP      RES     1
00244                    158 $TEMP1     RES    1                                                              1776
00245                    159 $TEMP2     RES    1                                                              1777
00246                    160 $TEMP3     RES    1                                                              1778
00247                    161 $TEMP4     RES    1                                                              1779
00250                    162 $TSLP      RES    1                                                              1780
00251                    163 $TLS       RES    1                         LATE START IN BASIC UNITS             1781
00252                    164 $TEFINS    RES    1                         PROJ. EXP. FIN. IN BASIC UNITS        1782
00253                    165 $TSS       RES    1                         SCHD. START IN BASIC UNITS            1783
00254                    166 $TAS       RES   1                         ACT. START IN BASIC UNITS             1784
00255                    167 $TAC       RES    1                         CURRENT TIME IN BASIC UNITS           1785
00256                    168 $TAF       RES    1                         ACT. FINISH IN BASIC UNITS            1786
00257                    169 $TPSF      RES    1                         TOTAL PROJ. SCHD. FINISH              1787
00260                    170 $TSF       RES    1                         SCHD. FIN. IN BASIC UNITS             1788
00261                    171 $TMR       RES    1                         T AT NODE M FROM RT. SIDE T VEC.      1789
00262                    172 $TSLCAL    RES    1                         TIME OF END OF EVENT                  1790
00263                    173 $TM        RES    1                                                              1791
```

```
00264                    174 $TN       RES   1                                                    1792
00265                    175 $TNP      RES   1                                                    1793
00266                    176 $TMP      RES   1                                                    1794
00267                    177 $TEF      RES   1        EARLY FINISH IN BASIC UNITS                 1795
00270                    178 $TLF      RES   1        LATE FINISH IN BASIC UNITS                  1796
00271                    179 $TFLOAT   RES   1        TOTAL FLOAT IN BINARY                       1797
00272                    180 $TORG     RES   1        ORG. SCHD. FIN. IN BASIC UNITS              1800
00273                    181 $UNITS    RES   1      BASIC UNITS TYPE INDICATOR                    1810
00274                    182 $DAY      RES   1                                                    1820
00275                    183 $MONTH    RES   1                                                    1830
00276                    184 $YEAR     RES   1                                                    1840
00277                    185 $YMDFOR   RES   1      YMD FORMAT INDICATOR                        1860
00300                    186 $WB1      RES   1                                                    1870
00301                    187 $WB2      RES   1                                                    1880
00302                    188 $WB3      RES   1                                                    1890
00303                    189 $WB4      RES   1                                                    1900
00304                    190 $WB5      RES   1                                                    2000
                         191 *      REG. VECTOR TEMP. REGION                                      2002
00305                    192 $NWDY     RES   15         NUMBER OF WORKING DAYS PER YEAR          2004
00324                    193 $NHY      RES 15           NUMBER OF HOLIDAYS PER YEAR              2006
00343                    194 $HOLORG   RES   15         HOLIDAY ORGINS                           2010
00362                    195 $DAYOFF   RES   7        DAY-OFF VECTOR                             2020
00371                    196 $USCR1    RES   18     USER'S COMMENT REGION 1                      2030
00413                    197 $USCR2    RES   18     USER'S COMMENT REGION 2                      2040
00435                    198 $CKDUR    RES   3      CHECK DURATION VECTOR                        2060
                         199 *      INPUT REGION                                                  2070
00440                    200 $IA       RES   70                                                   2075
         00000440        201 $IAO      EQU   IA                                                   2080
         00000440        202 $CODE     EQU   IA                                                   2085
         00000441        203 $IA1      EQU   IA+1                                                 2090
         00000441        204 $IBCD     EQU   IA+1       I IN BCD                                  2095
         00000442        205 $IA2      EQU   IA+2                                                 2100
         00000442        206 $JBCD     EQU   IA+2       J IN BCD                                  2105
         00000443        207 $IA3      EQU   IA+3                                                 2110
         00000443        208 $IJDUP    EQU   IA+3       IJ DUP IN BCD                             2115
         00000444        209 $IA4      EQU   IA+4                                                 2120
         00000444        210 $KBCD     EQU   IA+4       K IN BCD                                  2125
         00000445        211 $IA5      EQU   IA+5                                                 2130
         00000445        212 $MBCD     EQU   IA+5       M IN BCD                                  2135
         00000446        213 $IA6      EQU   IA+6                                                 2140
         00000446        214 $NBCD     EQU   IA+6       N IN BCD                                  2145
         00000447        215 $IA7      EQU   IA+7                                                 2150
         00000447        216 $SCHDUR   EQU   IA+7       SCHEDULED DURATION (BASIC UNITS)          2155
         00000450        217 $IA8      EQU   IA+8                                                 2160
         00000450        218 $SCHSDAY  EQU   IA+8       SCHEDULED START DAY                       2165
         00000451        219 $IA9      EQU   IA+9                                                 2170
         00000451        220 $SCHSMON  EQU   IA+9       SCHEDULED START MONT (ALPHA)              2175
         00000452        221 $IA10     EQU   IA+10                                                2180
         00000452        222 $SCHSYER  EQU   IA+10      SCHEDULED START YEAR (2 DIGITS)           2185
         00000453        223 $IA11     EQU   IA+11                                                2190
         00000453        224 $SCHFDAY  EQU   IA+11      SCHEDULED FINISH DAY                      2195
         00000454        225 $IA12     EQU   IA+12                                                2200
         00000454        226 $SCHFMON  EQU   IA+12      SCHEDULED FINISH MONTH (ALPHA)            2210
         00000455        227 $IA13     EQU   IA+13                                                2215
         00000455        228 $SCHFYER  EQU   IA+13      SCHEDULED FINISH YEAR
```

```
00000456          229 $IA14      EQU   IA+14
00000456          230 $ACTDUR    EQU   IA+14    ACTUAL DURATION (BASIC UNITS)
00000457          231 $IA15      EQU   IA+15
00000457          232 $ACTSDAY   EQU   IA+15    ACTUAL START DAY
00000460          233 $IA16      EQU   IA+16
00000460          234 $ACTSMON   EQU   IA+16    ACTUAL START MONTH
00000461          235 $IA17      EQU   IA+17
00000461          236 $ACTSYER   EQU   IA+17    ACTUAL START YEAR
00000462          237 $IA18      EQU   IA+18
00000462          238 $ACTFDAY   EQU   IA+18    ACTUAL FINISH DAY
00000463          239 $IA19      EQU   IA+19
00000463          240 $ACTFMON   EQU   IA+19    ACTUAL FINISH MONTH
00000464          241 $IA20      EQU   IA+20
00000464          242 $ACTFYER   EQU   IA+20    ACTUAL FINISH YEAR
00000465          243 $IA21      EQU   IA+21
00000465          244 $ORGFDAY   EQU   IA+21    ORIGINAL FINISH DAY
00000466          245 $IA22      EQU   IA+22
00000466          246 $ORGFMON   EQU   IA+22    ORIGINAL FINISH MONTH
00000467          247 $IA23      EQU   IA+23
00000467          248 $ORGFYER   EQU   IA+23    ORIGINAL FINISH YEAR
00000470          249 $IA24      EQU   IA+24
00000470          250 $EARSDY    EQU   IA+24    EARLY START DAY
00000471          251 $IA25      EQU   IA+25
00000471          252 $EARSMN    EQU   IA+25    EARLY START MONTH
00000472          253 $IA26      EQU   IA+26
00000472          254 $EARSYR    EQU   IA+26    EARLY START YEAR
00000473          255 $IA27      EQU   IA+27
00000473          256 $LATSDY    EQU   IA+27    LATE START DAY
00000474          257 $IA28      EQU   IA+28
00000474          258 $LATSMN    EQU   IA+28    LATE START MONTH
00000475          259 $IA29      EQU   IA+29
00000475          260 $LATSYR    EQU   IA+29    LATE START YEAR
00000476          261 $IA30      EQU   IA+30
00000476          262 $EARFDY    EQU   IA+30    EARLY FINISH DAY
00000477          263 $IA31      EQU   IA+31
00000477          264 $EARFMN    EQU   IA+31    EARLY FINISH MONTH
00000500          265 $IA32      EQU   IA+32
00000500          266 $EARFYR    EQU   IA+32    EARLY FINISH YEAR
00000501          267 $IA33      EQU   IA+33
00000501          268 $LATFDY    EQU   IA+33    LATE FINISH DAY
00000502          269 $IA34      EQU   IA+34
00000502          270 $LATFMN    EQU   IA+34    LATE FINISH MONTH
00000503          271 $IA35      EQU   IA+35
00000503          272 $LATFYR    EQU   IA+35    LATE FINISH YEAR
00000504          273 $IA36      EQU   IA+36
00000504          274 $SCHCRIT   EQU   IA+36    SCHEDULE CRITERION
00000505          275 $IA37      EQU   IA+37
00000505          276 $SCHCRIS   EQU   IA+37    SCHEDULE CRITERION SIGN
00000506          277 $IA38      EQU   IA+38
00000506          278 $ESTTFL    EQU   IA+38    ESTIMATED TOTAL FLOAT
00000507          279 $IA39      EQU   IA+39
00000507          280 $DIFFDUR   EQU   IA+39    DIFFERENTIAL DURATION
00000510          281 $IA40      EQU   IA+40
00000510          282 $DIFFDUS   EQU   IA+40    DIFFERENTIAL DURATION SIGN
00000511          283 $IA41      EQU   IA+41
```

```
00000511              284 $TOTSLIP   EQU   IA+41          TOTAL SLIP                    2461
00000512              285 $IA42      EQU   IA+42                                        2462
00000512              286 $TOTSLIS   EQU   IA+42          TOTAL SLIP SIGN               2463
00000513              287 $IA43      EQU   IA+43                                        2464
00000513              288 $NOSLIPS   EQU   IA+43          NUMBER OF SLIPS               2465
00000514              289 $IA44      EQU   IA+44          PREVIOUS SLIP INDICATOR       2466
00000515              290 $IA45      EQU   IA+45                                        2467
00000516              291 $IA46      EQU   IA+46          COST CODE                     2468
00000517              292 $IA47      EQU   IA+47                                        2469
00000520              293 $IA48      EQU   IA+48          RESPONSIBILITY CODE           2470
00000521              294 $IA49      EQU   IA+49          ACTIVITY DESCRIPTION          2471
00000522              295 $IA50      EQU   IA+50                                        2472
00000523              296 $IA51      EQU   IA+51                                        2473
00000524              297 $IA52      EQU   IA+52                                        2474
00000525              298 $IA53      EQU   IA+53                                        2475
00000526              299 $IA54      EQU   IA+54                                        2476
00000527              300 $IA55      EQU   IA+55                                        2477
00000530              301 $IA56      EQU   IA+56                                        2478
00000531              302 $IA57      EQU   IA+57                                        2479
00000532              303 $IA58      EQU   IA+58                                        2480
00000532              304 $MAJAIN    EQU   IA+58          MAJOR ACTIVITY INDICATOR      2481
00000533              305 $IA59      EQU   IA+59                                        2482
00000533              306 $MILDAY    EQU   IA+59          MILESTONE DAY                 2483
00000534              307 $IA60      EQU   IA+60                                        2484
00000534              308 $MILMON    EQU   IA+60          MILESTONE MONTH               2485
00000535              309 $IA61      EQU   IA+61                                        2486
00000535              310 $MILYER    EQU   IA+61          MILESTONE YEAR                2487
00000536              311 $IA62      EQU   IA+62                                        2488
00000537              312 $IA63      EQU   IA+63                                        2489
00000540              313 $IA64      EQU   IA+64                                        2490
00000541              314 $IA65      EQU   IA+65                                        2491
00000542              315 $IA66      EQU   IA+66                                        2492
00000543              316 $IA67      EQU   IA+67                                        2493
00000544              317 $IA68      EQU   IA+68                                        2494
00000545              318 $IA69      EQU   IA+69                                        2495
00000441              319 $INPUTAR   EQU   IA1                                         2496
00000523              320 $IAEND     EQU   INPUTAR+50                                  2497
                      321 *                                                            2510
                      322 *                                                            2520
                      323 *                                                            2530
00546                 324 $OUTAREA   RES   70                                          2540
                      325 *                                                            2550
00000546              326 $OAR1      EQU   OUTAREA                                     2560
00000547              327 $OAR2      EQU   OUTAREA+1                                   2570
00000550              328 $OAR3      EQU   OUTAREA+2                                   2580
00000551              329 $OAR4      EQU   OUTAREA+3                                   2590
00000552              330 $OAR5      EQU   OUTAREA+4                                   2600
00000553              331 $OAR6      EQU   OUTAREA+5                                   2610
00000554              332 $OAR7      EQU   OUTAREA+6                                   2620
00000555              333 $OAR8      EQU   OUTAREA+7                                   2630
00000556              334 $OAR9      EQU   OUTAREA+8                                   2640
00000557              335 $OAR10     EQU   OUTAREA+9                                   2650
00000560              336 $OAR11     EQU   OUTAREA+10                                  2660
00000561              337 $OAR12     EQU   OUTAREA+11                                  2670
00000562              338 $OAR13     EQU   OUTAREA+12
```

```
00000563          339 $0AR14    EQU  0UTAREA+13                                  2680
00000564          340 $0AR15    EQU  0UTAREA+14                                  2690
00000565          341 $0AR16    EQU  0UTAREA+15                                  2700
00000566          342 $0AR17    EQU  0UTAREA+16                                  2710
00000567          343 $0AR18    EQU  0UTAREA+17                                  2720
00000570          344 $0AR19    EQU  0UTAREA+18                                  2730
00000571          345 $0AR20    EQU  0UTAREA+19                                  2740
00000572          346 $0AR21    EQU  0UTAREA+20                                  2760
00000573          347 $0AR22    EQU  0UTAREA+21                                  2770
00000574          348 $0AR23    EQU  0UTAREA+22                                  2780
00000575          349 $0AR24    EQU  0UTAREA+23                                  2790
00000576          350 $0AR25    EQU  0UTAREA+24                                  2800
00000577          351 $0AR26    EQU  0UTAREA+25                                  2810
00000600          352 $0AR27    EQU  0UTAREA+26                                  2820
00000601          353 $0AR28    EQU  0UTAREA+27                                  2830
00000602          354 $0AR29    EQU  0UTAREA+28                                  2840
00000603          355 $0AR30    EQU  0UTAREA+29                                  2850
00000604          356 $0AR31    EQU  0UTAREA+30                                  2860
00000605          357 $0AR32    EQU  0UTAREA+31                                  2870
00000606          358 $0AR33    EQU  0UTAREA+32   LAST WORD OF OUTPUT AREA       2880
00000607          359 $0UTAREAL EQU  0UTAREA+33   LAST+1 WORD OF OUTPUT AREA     2890
00000607          360 $0AR34    EQU   0UTAREA+33                                 3000
00000610          361 $0AR35    EQU   0UTAREA+34                                 3010
00000611          362 $0AR36    EQU   0UTAREA+35                                 3020
00000612          363 $0AR37    EQU   0UTAREA+36                                 3030
00000613          364 $0AR38    EQU   0UTAREA+37                                 3040
00000614          365 $0AR39    EQU   0UTAREA+38                                 3050
00000615          366 $0AR40    EQU   0UTAREA+39                                 3060
00000616          367 $0AR41    EQU   0UTAREA+40                                 3070
00000617          368 $0AR42    EQU   0UTAREA+41                                 3080
00000620          369 $0AR43    EQU   0UTAREA+42                                 3090
00000621          370 $0AR44    EQU   0UTAREA+43                                 3100
00000622          371 $0AR45    EQU   0UTAREA+44                                 3110
00000623          372 $0AR46    EQU   0UTAREA+45                                 3120
00000624          373 $0AR47    EQU   0UTAREA+46                                 3130
00000625          374 $0AR48    EQU   0UTAREA+47                                 3140
00000626          375 $0AR49    EQU   0UTAREA+48                                 3150
00000627          376 $0AR50    EQU   0UTAREA+49                                 3160
00000630          377 $0AR51    EQU   0UTAREA+50                                 3170
00000631          378 $0AR52    EQU   0UTAREA+51                                 3180
00000632          379 $0AR53    EQU   0UTAREA+52                                 3190
00000633          380 $0AR54    EQU   0UTAREA+53                                 3200
00000634          381 $0AR55    EQU   0UTAREA+54                                 3210
00000635          382 $0AR56    EQU   0UTAREA+55                                 3220
00000636          383 $0AR57    EQU   0UTAREA+56                                 3230
00000637          384 $0AR58    EQU   0UTAREA+57                                 3240
00000640          385 $0AR59    EQU   0UTAREA+58                                 3250
00000641          386 $0AR60    EQU   0UTAREA+59                                 3260
00000642          387 $0AR61    EQU   0UTAREA+60                                 3270
00000643          388 $0AR62    EQU   0UTAREA+61                                 3280
00000644          389 $0AR63    EQU   0UTAREA+62                                 3290
00000645          390 $0AR64    EQU   0UTAREA+63                                 3300
00000646          391 $0AR65    EQU   0UTAREA+64                                 3310
00000647          392 $0AR66    EQU   0UTAREA+65                                 3320
00000650          393 $0AR67    EQU   0UTAREA+66                                 3330
```

```
00000651              394 $OAR68    EQU   OUTAREA+67                          3340
00000652              395 $OAR69    EQU   OUTAREA+68                          3350
00000653              396 $OAR70    EQU   OUTAREA+69                          3360
00000644              397 $OARAP1   EQU   OUTAREA+62                          4000
00000370              398 $USCM1L   EQU   USCR1-1                             4010
00000412              399 $USCM2L   EQU   USCR2-1                             4020
00000435              400 $CKDUR1   EQU   CKDUR                               4030
00000436              401 $CKDUR2   EQU   CKDUR+1                             4040
00000437              402 $CKDUR3   EQU   CKDUR+2                             4050
00000440              403 $CKDURL   EQU   CKDUR+3                             4060
00000104              404 $ACTNUMWD EQU   NWBAR                               4070
00000170              405 $EFFDATE  EQU   EFFDAY+3                            4080
00000304              406 $NWDYM1   EQU   NWDY-1                              4090
00000323              407 $NHYM1    EQU   NHY-1                               4100
00000342              408 $HOLDM1   EQU   HOLDRG-1                            4110
                      409           END                                       9999
```


PEPSORT COMMON

```
00000                   1           RORG  0          COMMON DATA  AREA          0000
00000                   2 $ACTAA    RES   1              ACT NO. OF CURRENT ACT 0020
00001                   3 $ACTBB    RES   1              -(NO. OF RECORDS TO WRITE) 0030
00002                   4 $ACTK     RES   1              CURRENT NO OF ACT PROCESSED 0040
00003 00000101          5 $ACTNUMWD DATA  65             NO WORDS IN ACT RECORD 0050
00004 60606060          6 $BCDBLK   DATA  '    '             BCD BLANK
00005                   7 $BCDCODE  RES   13
      00000004          8 $BCDM1    EQU   BCDCODE-1
00022 20000000          9 $BCDPL    DATA  '+000'
00023 00000033         10 $BCDDP    DATA  '000.'
00024 37777777         11 $DACTI    DATA  037777777
00025                  12 $EFFDAY   RES   1
00026                  13 $EFFMON   RES   1
00027                  14 $EFFYER   RES   1
00030                  15 $ENDFILE  RES   1
00031 00000000         16 $ERD      DATA  0
00032 00000000         17 $ERI      DATA  0
00033 00000000         18 $ERJ      DATA  0
00034 00000000         19 $EXPFMN   DATA  0
00035 00000000         20 $EXPFDY   DATA  0
00036 00000000         21 $EXPFYR   DATA  0
00037 60606024         22 $UNITS    DATA  '   D'
00040                  23 $IAO      RES     70            INPUT AREA
      00000040         24 $IA       EQU   IAO
      00000040         25 $CODE     EQU   IA                                  2080
      00000041         26 $IA1      EQU   IA+1                                2085
      00000041         27 $IBCD     EQU   IA+1              I IN BCD          2090
```

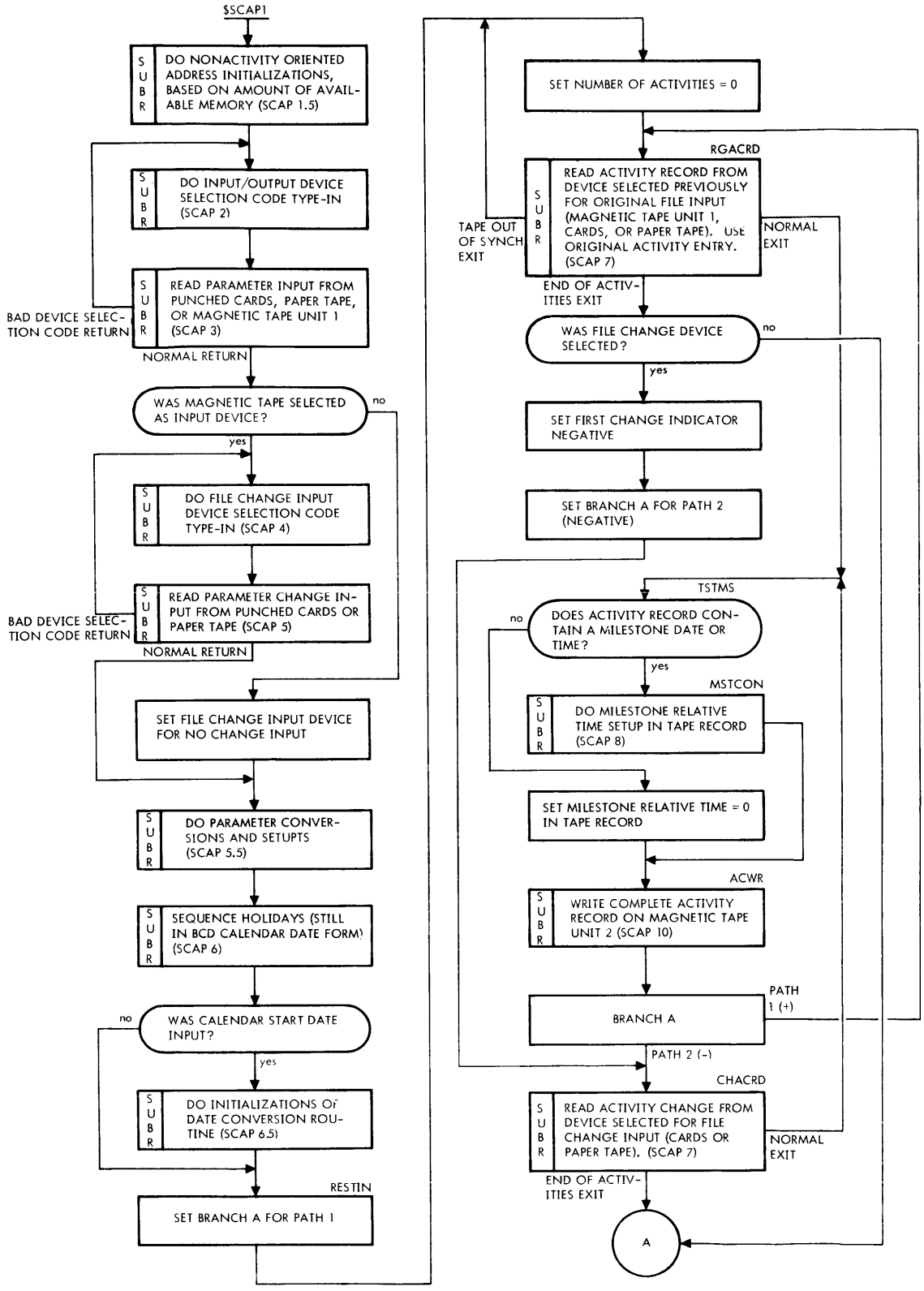
```
00000042  28 $IA2      EQU   IA+2                                        2095
00000042  29 $JBCD     EQU   IA+2     J IN BCD                           2100
00000043  30 $IA3      EQU   IA+3                                        2105
00000043  31 $IJDUP    EQU   IA+3     IJ DUP IN BCD                      2110
00000044  32 $IA4      EQU   IA+4                                        2115
00000044  33 $KBCD     EQU   IA+4     K IN BCD                           2120
00000045  34 $IA5      EQU   IA+5                                        2125
00000045  35 $MBCD     EQU   IA+5     M IN BCD                           2130
00000046  36 $IA6      EQU   IA+6                                        2135
00000046  37 $NBCD     EQU   IA+6     N IN BCD                           2140
00000047  38 $IA7      EQU   IA+7                                        2145
00000047  39 $SCHDUR   EQU   IA+7     SCHEDULED DURATION (BASIC UNITS)   2150
00000050  40 $IA8      EQU   IA+8                                        2155
00000050  41 $SCHSDAY  EQU   IA+8     SCHEDULED START DAY                2160
00000051  42 $IA9      EQU   IA+9                                        2165
00000051  43 $SCHSMON  EQU   IA+9     SCHEDULED START MONT (ALPHA)       2170
00000052  44 $IA10     EQU   IA+10                                       2175
00000052  45 $SCHSYER  EQU   IA+10    SCHEDULED START YEAR (2 DIGITS)    2180
00000053  46 $IA11     EQU   IA+11                                       2185
00000053  47 $SCHFDAY  EQU   IA+11    SCHEDULED FINISH DAY               2190
00000054  48 $IA12     EQU   IA+12                                       2195
00000054  49 $SCHFMON  EQU   IA+12    SCHEDULED FINISH MONTH (ALPHA)     2200
00000055  50 $IA13     EQU   IA+13                                       2210
00000055  51 $SCHFYER  EQU   IA+13    SCHEDULED FINISH YEAR              2215
00000056  52 $IA14     EQU   IA+14                                       2220
00000056  53 $ACTDUR   EQU   IA+14    ACTUAL DURATION (BASIC UNITS)      2225
00000057  54 $IA15     EQU   IA+15                                       2230
00000057  55 $ACTSDAY  EQU   IA+15    ACTUAL START DAY                   2235
00000060  56 $IA16     EQU   IA+16                                       2240
00000060  57 $ACTSMON  EQU   IA+16    ACTUAL START MONTH                 2250
00000061  58 $IA17     EQU   IA+17                                       2255
00000061  59 $ACTSYER  EQU   IA+17    ACTUAL START YEAR                  2260
00000062  60 $IA18     EQU   IA+18                                       2265
00000062  61 $ACTFDAY  EQU   IA+18    ACTUAL FINISH DAY                  2270
00000063  62 $IA19     EQU   IA+19                                       2275
00000063  63 $ACTFMON  EQU   IA+19    ACTUAL FINISH MONTH                2280
00000064  64 $IA20     EQU   IA+20                                       2285
00000064  65 $ACTFYER  EQU   IA+20    ACTUAL FINISH YEAR                 2290
00000065  66 $IA21     EQU   IA+21                                       2295
00000065  67 $ORGFDAY  EQU   IA+21    ORIGINAL FINISH DAY                2300
00000066  68 $IA22     EQU   IA+22                                       2305
00000066  69 $ORGFMON  EQU   IA+22    ORIGINAL FINISH MONTH              2310
00000067  70 $IA23     EQU   IA+23                                       2315
00000067  71 $ORGFYER  EQU   IA+23    ORIGINAL FINISH YEAR               2320
00000070  72 $IA24     EQU   IA+24                                       2325
00000070  73 $EARSDY   EQU   IA+24    EARLY START DAY                    2330
00000071  74 $IA25     EQU   IA+25                                       2335
00000071  75 $EARSMN   EQU   IA+25    EARLY START MONTH                  2340
00000072  76 $IA26     EQU   IA+26                                       2345
00000072  77 $EARSYR   EQU   IA+26    EARLY START YEAR                   2350
00000073  78 $IA27     EQU   IA+27                                       2355
00000073  79 $LATSDY   EQU   IA+27    LATE START DAY                     2360
00000074  80 $IA28     EQU   IA+28                                       2365
00000074  81 $LATSMN   EQU   IA+28    LATE START MONTH                   2370
00000075  82 $IA29     EQU   IA+29                                       2375
```

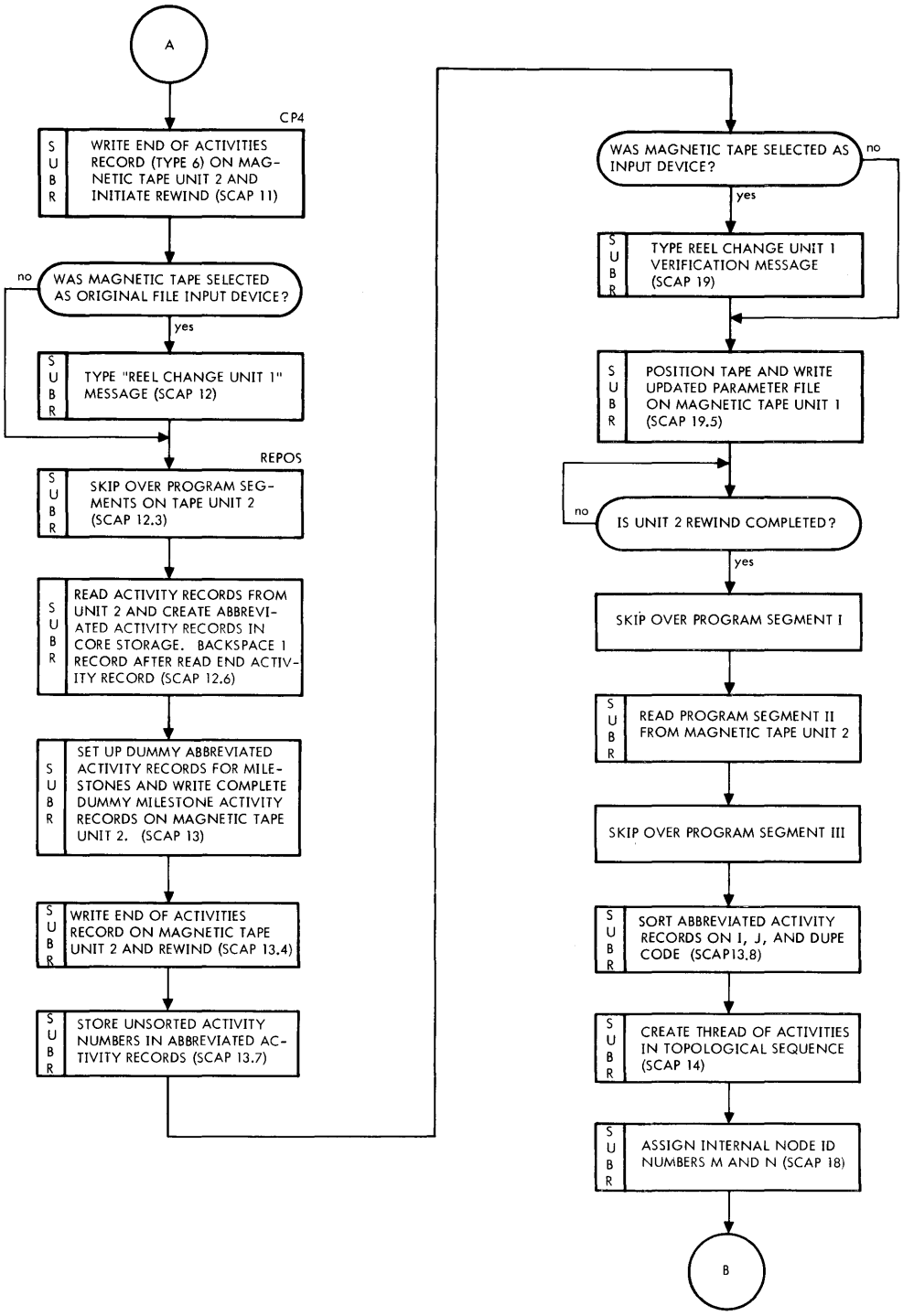
```
00000075          83 $LATSYR   EQU   IA+29          LATE START YEAR                2380
                                                                                    2385
00000076          84 $IA30     EQU   IA+30
00000076          85 $EARFDY   EQU   IA+30          EARLY FINISH DAY               2390
                                                                                    2395
00000077          86 $IA31     EQU   IA+31
00000077          87 $EARFMN   EQU   IA+31          EARLY FINISH MONTH             2400
                                                                                    2410
00000100          88 $IA32     EQU   IA+32
00000100          89 $EARFYR   EQU   IA+32          EARLY FINISH YEAR              2415
                                                                                    2420
00000101          90 $IA33     EQU   IA+33
00000101          91 $LATFDY   EQU   IA+33          LATE FINISH DAY                2425
                                                                                    2430
00000102          92 $IA34     EQU   IA+34
00000102          93 $LATFMN   EQU   IA+34          LATE FINISH MONTH              2435
                                                                                    2440
00000103          94 $IA35     EQU   IA+35
00000103          95 $LATFYR   EQU   IA+35          LATE FINISH YEAR               2445
                                                                                    2450
00000104          96 $IA36     EQU   IA+36
00000104          97 $SCHCRIT  EQU   IA+36          SCHEDULE CRITERION             2451
                                                                                    2452
00000105          98 $IA37     EQU   IA+37
00000105          99 $SCHCRIS  EQU   IA+37          SCHEDULE CRITERION SIGN        2453
                                                                                    2454
00000106         100 $IA38     EQU   IA+38
00000106         101 $ESTTFL   EQU   IA+38          ESTIMATED TOTAL FLOAT          2455
                                                                                    2456
00000107         102 $IA39     EQU   IA+39
00000107         103 $DIFFDUR  EQU   IA+39          DIFFERENTIAL DURATION          2457
                                                                                    2458
00000110         104 $IA40     EQU   IA+40
00000110         105 $DIFFDUS  EQU   IA+40          DIFFERENTIAL DURATION SIGN     2459
                                                                                    2460
00000111         106 $IA41     EQU   IA+41
00000111         107 $TOTSLIP  EQU   IA+41          TOTAL SLIP                     2461
                                                                                    2462
00000112         108 $IA42     EQU   IA+42
00000112         109 $TOTSLIS  EQU   IA+42          TOTAL SLIP SIGN                2463
                                                                                    2464
00000113         110 $IA43     EQU   IA+43
00000113         111 $NOSLIPS  EQU   IA+43          NUMBER OF SLIPS                2465
00000114         112 $IA44     EQU   IA+44          PREVIOUS SLIP INDICATOR        2466
00000115         113 $IA45     EQU   IA+45                                         2467
00000116         114 $IA46     EQU   IA+46          COST CODE                      2468
00000117         115 $IA47     EQU   IA+47                                         2469
00000120         116 $IA48     EQU   IA+48          RESPONSIBILITY CODE            2470
00000121         117 $IA49     EQU   IA+49          ACTIVITY DESCRIPTION           2471
00000122         118 $IA50     EQU   IA+50                                         2472
00000123         119 $IA51     EQU   IA+51                                         2473
00000124         120 $IA52     EQU   IA+52                                         2474
00000125         121 $IA53     EQU   IA+53                                         2475
00000126         122 $IA54     EQU   IA+54                                         2476
00000127         123 $IA55     EQU   IA+55                                         2477
00000130         124 $IA56     EQU   IA+56                                         2478
00000131         125 $IA57     EQU   IA+57                                         2479
00000132         126 $IA58     EQU   IA+58                                         2480
00000132         127 $MAJAIN   EQU   IA+58          MAJOR ACTIVITY INDICATOR       2481
00000133         128 $IA59     EQU   IA+59                                         2482
00000133         129 $MILDAY   EQU   IA+59          MILESTONE DAY                  2483
00000134         130 $IA60     EQU   IA+60                                         2484
00000134         131 $MILMON   EQU   IA+60          MILESTONE MONTH                2485
00000135         132 $IA61     EQU   IA+61                                         2486
00000135         133 $MILYER   EQU   IA+61          MILESTONE YEAR                 2487
00000136         134 $IA62     EQU   IA+62                                         2488
00000136         135 $IAOL     EQU    IAO+62
00000137         136 $IA63     EQU   IA+63                                         2489
00000140         137 $IA64     EQU   IA+64                                         2490
```

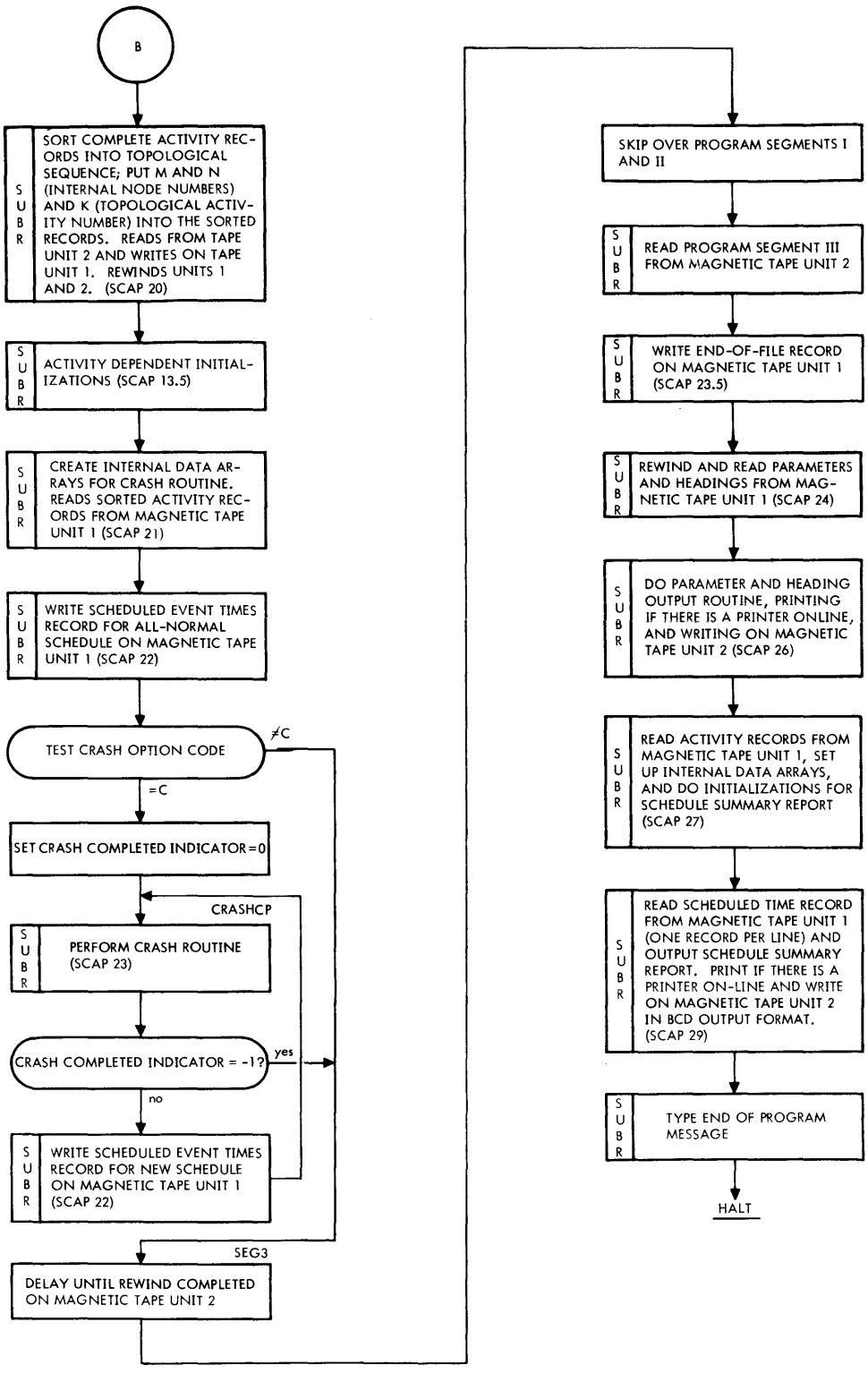
```
00000141          138 $IA65      EQU   IA+65                                    2491
00000142          139 $IA66      EQU   IA+66                                    2492
00000143          140 $IA67      EQU   IA+67                                    2493
00000144          141 $IA68      EQU   IA+68                                    2494
00000145          142 $IA69      EQU   IA+69                                    2495
00000041          143 $INPUTAR   EQU   IA1                                      2496
00000123          144 $IAEND     EQU   INPUTAR+50                               2497
                  145 *                                                         2510
                  146 *                                                         2520
                  147 *                                                         2530
00146             148 $OUTAREA   RES   70                                       2540
                  149 *                                                         2550
00000147          150 $OAR2      EQU   OUTAREA+1                                2560
00000150          151 $OAR3      EQU   OUTAREA+2                                2570
00000151          152 $OAR4      EQU   OUTAREA+3                                2580
00000152          153 $OAR5      EQU   OUTAREA+4                                2590
00000153          154 $OAR6      EQU   OUTAREA+5                                2600
00000154          155 $OAR7      EQU   OUTAREA+6                                2610
00000155          156 $OAR8      EQU   OUTAREA+7                                2620
00000156          157 $OAR9      EQU   OUTAREA+8                                2630
00000157          158 $OAR10     EQU   OUTAREA+9                                2640
00000160          159 $OAR11     EQU   OUTAREA+10                               2650
00000161          160 $OAR12     EQU   OUTAREA+11                               2660
00000162          161 $OAR13     EQU   OUTAREA+12                               2670
00000163          162 $OAR14     EQU   OUTAREA+13                               2680
00000164          163 $OAR15     EQU   OUTAREA+14                               2690
00000165          164 $OAR16     EQU   OUTAREA+15                               2700
00000166          165 $OAR17     EQU   OUTAREA+16                               2710
00000167          166 $OAR18     EQU   OUTAREA+17                               2720
00000170          167 $OAR19     EQU   OUTAREA+18                               2730
00000171          168 $OAR20     EQU   OUTAREA+19                               2740
00000172          169 $OAR21     EQU   OUTAREA+20                               2760
00000173          170 $OAR22     EQU   OUTAREA+21                               2770
00000174          171 $OAR23     EQU   OUTAREA+22                               2780
00000175          172 $OAR24     EQU   OUTAREA+23                               2790
00000176          173 $OAR25     EQU   OUTAREA+24                               2800
00000177          174 $OAR26     EQU   OUTAREA+25                               2810
00000200          175 $OAR27     EQU   OUTAREA+26                               2820
00000201          176 $OAR28     EQU   OUTAREA+27                               2830
00000202          177 $OAR29     EQU   OUTAREA+28                               2840
00000203          178 $OAR30     EQU   OUTAREA+29                               2850
00000204          179 $OAR31     EQU   OUTAREA+30                               2860
00000205          180 $OAR32     EQU   OUTAREA+31                               2870
00000206          181 $OAR33     EQU   OUTAREA+32     LAST WORD OF OUTPUT AREA  2880
00000207          182 $OUTAREAL  EQU   OUTAREA+33     LAST+1 WORD OF OUTPUT AREA 2890
00000207          183 $OAR34     EQU    OUTAREA+33                              3000
00000210          184 $OAR35     EQU    OUTAREA+34                              3010
00000211          185 $OAR36     EQU    OUTAREA+35                              3020
00000212          186 $OAR37     EQU    OUTAREA+36                              3030
00000213          187 $OAR38     EQU    OUTAREA+37                              3040
00000214          188 $OAR39     EQU    OUTAREA+38                              3050
00000215          189 $OAR40     EQU    OUTAREA+39                              3060
00000217          190 $OAR42     EQU    OUTAREA+41                              3080
00000220          191 $OAR43     EQU    OUTAREA+42                              3090
00000221          192 $OAR44     EQU    OUTAREA+43                              3100
```

```
00000222          193 $0AR45    EQU   0UTAREA+44                                  3110
00000223          194 $0AR46    EQU   0UTAREA+45                                  3120
00000224          195 $0AR47    EQU   0UTAREA+46                                  3130
00000225          196 $0AR48    EQU   0UTAREA+47                                  3140
00000226          197 $0AR49    EQU   0UTAREA+48                                  3150
00000227          198 $0AR50    EQU   0UTAREA+49                                  3160
00000230          199 $0AR51    EQU   0UTAREA+50                                  3170
00000231          200 $0AR52    EQU   0UTAREA+51                                  3180
00000232          201 $0AR53    EQU   0UTAREA+52                                  3190
00000233          202 $0AR54    EQU   0UTAREA+53                                  3200
00000234          203 $0AR55    EQU   0UTAREA+54                                  3210
00000235          204 $0AR56    EQU   0UTAREA+55                                  3220
00000236          205 $0AR57    EQU   0UTAREA+56                                  3230
00000237          206 $0AR58    EQU   0UTAREA+57                                  3240
00000240          207 $0AR59    EQU   0UTAREA+58                                  3250
00000241          208 $0AR60    EQU   0UTAREA+59                                  3260
00000242          209 $0AR61    EQU   0UTAREA+60                                  3270
00000243          210 $0AR62    EQU   0UTAREA+61                                  3280
00000244          211 $0ARAP1   EQU   0UTAREA+62                                  4000
00000244          212 $0AR63    EQU   0UTAREA+62                                  3290
00000245          213 $0AR64    EQU   0UTAREA+63                                  3300
00000246          214 $0AR65    EQU   0UTAREA+64                                  3310
00000247          215 $0AR66    EQU   0UTAREA+65                                  3320
00000250          216 $0AR67    EQU   0UTAREA+66                                  3330
00000251          217 $0AR68    EQU   0UTAREA+67                                  3340
00000252          218 $0AR69    EQU   0UTAREA+68                                  3350
00000253          219 $0AR70    EQU   0UTAREA+69                                  3360
00254             220 $LINEC    RES   1
00255             221 $LPK      RES   1          NO OF FIRST ACT IN CURRENT BLOCK 0060
00256             222 $MAJOR    RES   1
00257 77000000    223 $MASK1    DATA  077000000
00260 77770000    224 $MASK12   DATA  077770000
00261 00777700    225 $MASK23   DATA  000777700
00262 00770000    226 $MASK2    DATA  00770000
00263 00007700    227 $MASK3    DATA  00007700
00264 00000077    228 $MASK4 DATA     00000077
00265 00007777    229 $MASK34   DATA  000007777                                   0070
00266 77777700    230 $MASK123  DATA  077777700
00267 00777777    231 $MASK234  DATA  00777777
00270 00077777    232 $MASKADR  DATA  000077777
00271 00077777    233 $MASKADA  DATA  000077777                                   0080
00272             234 $MAXABR   RES   1          MAX LENGTH OF S VECTOR
00273 77777777    235 $MONE     DATA  -1                                          0090
00274             236 $NA       RES   1          TOTAL NO OF ACT                  0100
00275             237 $NCODES   RES   1
00276 77777677    238 $NEGNW    DATA  -65        NEG NO OF WORDS IN ACT RECORD
00277 00000001    239 $ONE      DATA  1
00300             240 $PAGEC    RES   1
00301             241 $PC       RES   1          LOC OF PRIMARY CODES ,TAGED
00302             242 $PCODE    RES   1
00303             243 $PFPRINT  RES   1
00304 00000000    244 $PRNTCODE DATA  0        OUTPUT ON PRINTE Y=YES, N=NO
00305 60606060    245 $PSTATUS  DATA  '    '
00306 00000000    246 $QT    DATA    0
00307 00000000    247 $RGINDEV  DATA  0        DEVICE SEL, CODE, C=CDS,P=PAPER,M=MAGTAPE
```

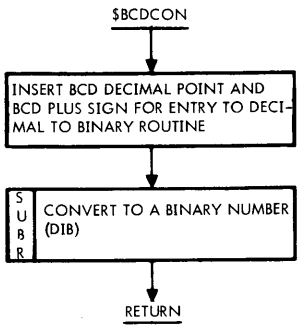
```
00310                    248 $RUNID     RES   1
00311                    249 $RUNIDP1   RES   1
00312  00000001          250 $RPUN0     DATA  1
00313                    251 $S         RES   1              LOC OF THE ABRIVIATED ACT REC.          0010
00314                    252 $SC        RES   1                   LOC OF SECONDARY CODES.TAGED
00315                    253 $SCODE     RES   1
00316                    254 $SECKEY    RES   1
00317                    255 $SELECT    RES   1                   LOC OF START OF SELECT AREA
00320                    256 $SEQ       RES   1              LOC OF FIRST WORD IN SEQ ARRAY          0110
00321                    257 $SEQEND    RES   1              LOC OF LAST WORD+1  SEQ ARRAY           0120
00322                    258 $SEQL      RES   1        INDEX LOC OF LAST WORD+1  SEQ ARRAY           0130
00323                    259 $SELECTL   RES   1        INDEX LOC OF LAST WORD+1  SELECT ARRAY        0140
00324                    260 $SRTCDN    RES   13           VECTOR OF NUMERIC SORT CODES
00341  00000000          261 $SCHPFDY   DATA  0
00342  00000000          262 $SCHPFMN   DATA  0
00343  00000000          263 $SCHPFYR   DATA  0
       00000323          264 $SRTCD     EQU   SRTCDN-1
00344  60606060          265 $STATWD1   DATA  '    '
00345  60606060          266 $STATWD2   DATA  '    '
00346                    267 $STRINGM   RES   1              NO. OF RECORDS IN A BLOCK               0150
00347  57777777          268 $TAGDEL    DATA  057777777        DELETE TAG MASK
00350  20000000          269 $TAGBIT    DATA  020000000
       00000350          270 $TAG       EQU   TAGBIT
00351                    271 $THREDIN   RES   1              LOC OF RECORD BLOCK                     0160
00352  00000003          272 $THREE     DATA  3
00353  00000002          273 $TWO       DATA  2
00354                    274 $USECA     RES   18
00376                    275 $USECB     RES   18
00420  2 35 0 00376      276 $UC1ST0    STA   USECA+18,2
00421  2 35 0 00420      277 $UC2ST0    STA   USECB+18,2
00422  00000000          278 $ZERO      DATA  0                                                       0170
       00000422          279 $Z         EQU   ZERO
                         280            END                                                           3000
```


$SCAP1
SUBR DO NONACTIVITY ORIENTED ADDRESS INITIALIZATIONS, BASED ON AMOUNT OF AVAILABLE MEMORY (SCAP 1.5)
SUBR DO INPUT/OUTPUT DEVICE SELECTION CODE TYPE-IN (SCAP 2)
SUBR READ PARAMETER INPUT FROM PUNCHED CARDS, PAPER TAPE, OR MAGNETIC TAPE UNIT 1 (SCAP 3)
BAD DEVICE SELECTION CODE RETURN
NORMAL RETURN
WAS MAGNETIC TAPE SELECTED AS INPUT DEVICE?
no
yes
SUBR DO FILE CHANGE INPUT DEVICE SELECTION CODE TYPE-IN (SCAP 4)
SUBR READ PARAMETER CHANGE INPUT FROM PUNCHED CARDS OR PAPER TAPE (SCAP 5)
BAD DEVICE SELECTION CODE RETURN
NORMAL RETURN
SET FILE CHANGE INPUT DEVICE FOR NO CHANGE INPUT
SUBR DO PARAMETER CONVERSIONS AND SETUPTS (SCAP 5.5)
SUBR SEQUENCE HOLIDAYS (STILL IN BCD CALENDAR DATE FORM) (SCAP 6)
WAS CALENDAR START DATE INPUT?
no
yes
SUBR DO INITIALIZATIONS OF DATE CONVERSION ROUTINE (SCAP 6.5)
RESTIN
SET BRANCH A FOR PATH 1
SET NUMBER OF ACTIVITIES = 0
RGACRD
SUBR READ ACTIVITY RECORD FROM DEVICE SELECTED PREVIOUSLY FOR ORIGINAL FILE INPUT (MAGNETIC TAPE UNIT 1, CARDS, OR PAPER TAPE). USE ORIGINAL ACTIVITY ENTRY. (SCAP 7)
TAPE OUT OF SYNCH EXIT
NORMAL EXIT
END OF ACTIVITIES EXIT
WAS FILE CHANGE DEVICE SELECTED?
no
yes
SET FIRST CHANGE INDICATOR NEGATIVE
SET BRANCH A FOR PATH 2 (NEGATIVE)
TSTMS
DOES ACTIVITY RECORD CONTAIN A MILESTONE DATE OR TIME?
no
yes
MSTCON
SUBR DO MILESTONE RELATIVE TIME SETUP IN TAPE RECORD (SCAP 8)
SET MILESTONE RELATIVE TIME = 0 IN TAPE RECORD
ACWR
SUBR WRITE COMPLETE ACTIVITY RECORD ON MAGNETIC TAPE UNIT 2 (SCAP 10)
BRANCH A
PATH 1 (+)
PATH 2 (-)
CHACRD
SUBR READ ACTIVITY CHANGE FROM DEVICE SELECTED FOR FILE CHANGE INPUT (CARDS OR PAPER TAPE). (SCAP 7)
NORMAL EXIT
END OF ACTIVITIES EXIT
A

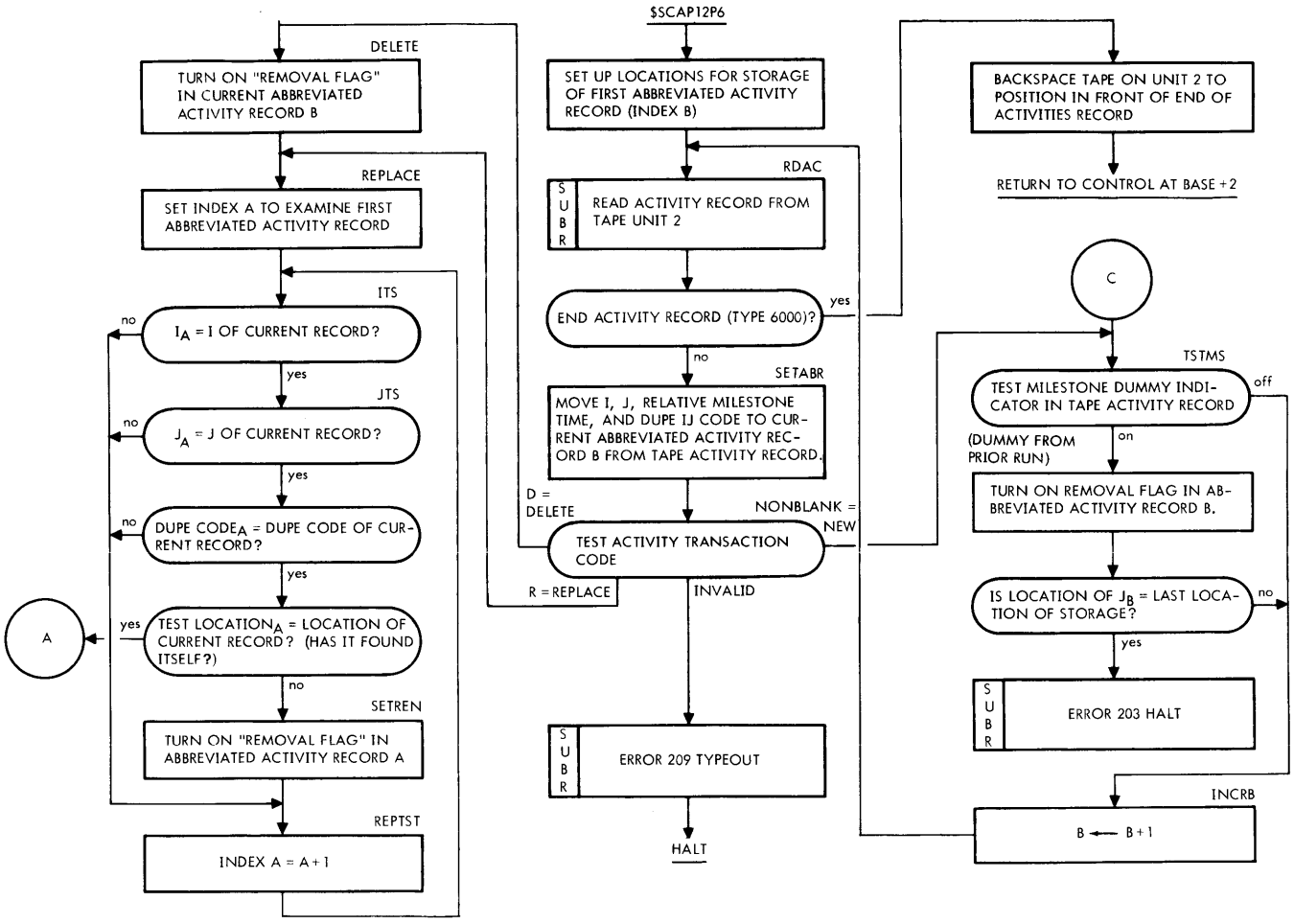

A
CP4
SUBR
WRITE END OF ACTIVITIES RECORD (TYPE 6) ON MAGNETIC TAPE UNIT 2 AND INITIATE REWIND (SCAP 11)
WAS MAGNETIC TAPE SELECTED AS ORIGINAL FILE INPUT DEVICE?
no
yes
TYPE "REEL CHANGE UNIT 1" MESSAGE (SCAP 12)
REPOS
SKIP OVER PROGRAM SEGMENTS ON TAPE UNIT 2 (SCAP 12.3)
READ ACTIVITY RECORDS FROM UNIT 2 AND CREATE ABBREVIATED ACTIVITY RECORDS IN CORE STORAGE. BACKSPACE 1 RECORD AFTER READ END ACTIVITY RECORD (SCAP 12.6)
SET UP DUMMY ABBREVIATED ACTIVITY RECORDS FOR MILESTONES AND WRITE COMPLETE DUMMY MILESTONE ACTIVITY RECORDS ON MAGNETIC TAPE UNIT 2. (SCAP 13)
WRITE END OF ACTIVITIES RECORD ON MAGNETIC TAPE UNIT 2 AND REWIND (SCAP 13.4)
STORE UNSORTED ACTIVITY NUMBERS IN ABBREVIATED ACTIVITY RECORDS (SCAP 13.7)
WAS MAGNETIC TAPE SELECTED AS INPUT DEVICE?
no
yes
TYPE REEL CHANGE UNIT 1 VERIFICATION MESSAGE (SCAP 19)
POSITION TAPE AND WRITE UPDATED PARAMETER FILE ON MAGNETIC TAPE UNIT 1 (SCAP 19.5)
IS UNIT 2 REWIND COMPLETED?
no
yes
SKIP OVER PROGRAM SEGMENT I
READ PROGRAM SEGMENT II FROM MAGNETIC TAPE UNIT 2
SKIP OVER PROGRAM SEGMENT III
SORT ABBREVIATED ACTIVITY RECORDS ON I, J, AND DUPE CODE (SCAP13.8)
CREATE THREAD OF ACTIVITIES IN TOPOLOGICAL SEQUENCE (SCAP 14)
ASSIGN INTERNAL NODE ID NUMBERS M AND N (SCAP 18)
B


B
SUBR SORT COMPLETE ACTIVITY RECORDS INTO TOPOLOGICAL SEQUENCE; PUT M AND N (INTERNAL NODE NUMBERS) AND K (TOPOLOGICAL ACTIVITY NUMBER) INTO THE SORTED RECORDS. READS FROM TAPE UNIT 2 AND WRITES ON TAPE UNIT 1. REWINDS UNITS 1 AND 2. (SCAP 20)
SUBR ACTIVITY DEPENDENT INITIALIZATIONS (SCAP 13.5)
SUBR CREATE INTERNAL DATA ARRAYS FOR CRASH ROUTINE. READS SORTED ACTIVITY RECORDS FROM MAGNETIC TAPE UNIT 1 (SCAP 21)
SUBR WRITE SCHEDULED EVENT TIMES RECORD FOR ALL-NORMAL SCHEDULE ON MAGNETIC TAPE UNIT 1 (SCAP 22)
TEST CRASH OPTION CODE
≠C
=C
SET CRASH COMPLETED INDICATOR=0
CRASHCP
SUBR PERFORM CRASH ROUTINE (SCAP 23)
CRASH COMPLETED INDICATOR = -1?
yes
no
SUBR WRITE SCHEDULED EVENT TIMES RECORD FOR NEW SCHEDULE ON MAGNETIC TAPE UNIT 1 (SCAP 22)
SEG3
DELAY UNTIL REWIND COMPLETED ON MAGNETIC TAPE UNIT 2
SKIP OVER PROGRAM SEGMENTS I AND II
SUBR READ PROGRAM SEGMENT III FROM MAGNETIC TAPE UNIT 2
SUBR WRITE END-OF-FILE RECORD ON MAGNETIC TAPE UNIT 1 (SCAP 23.5)
SUBR REWIND AND READ PARAMETERS AND HEADINGS FROM MAGNETIC TAPE UNIT 1 (SCAP 24)
SUBR DO PARAMETER AND HEADING OUTPUT ROUTINE, PRINTING IF THERE IS A PRINTER ONLINE, AND WRITING ON MAGNETIC TAPE UNIT 2 (SCAP 26)
SUBR READ ACTIVITY RECORDS FROM MAGNETIC TAPE UNIT 1, SET UP INTERNAL DATA ARRAYS, AND DO INITIALIZATIONS FOR SCHEDULE SUMMARY REPORT (SCAP 27)
SUBR READ SCHEDULED TIME RECORD FROM MAGNETIC TAPE UNIT 1 (ONE RECORD PER LINE) AND OUTPUT SCHEDULE SUMMARY REPORT. PRINT IF THERE IS A PRINTER ON-LINE AND WRITE ON MAGNETIC TAPE UNIT 2 IN BCD OUTPUT FORMAT. (SCAP 29)
SUBR TYPE END OF PROGRAM MESSAGE
HALT

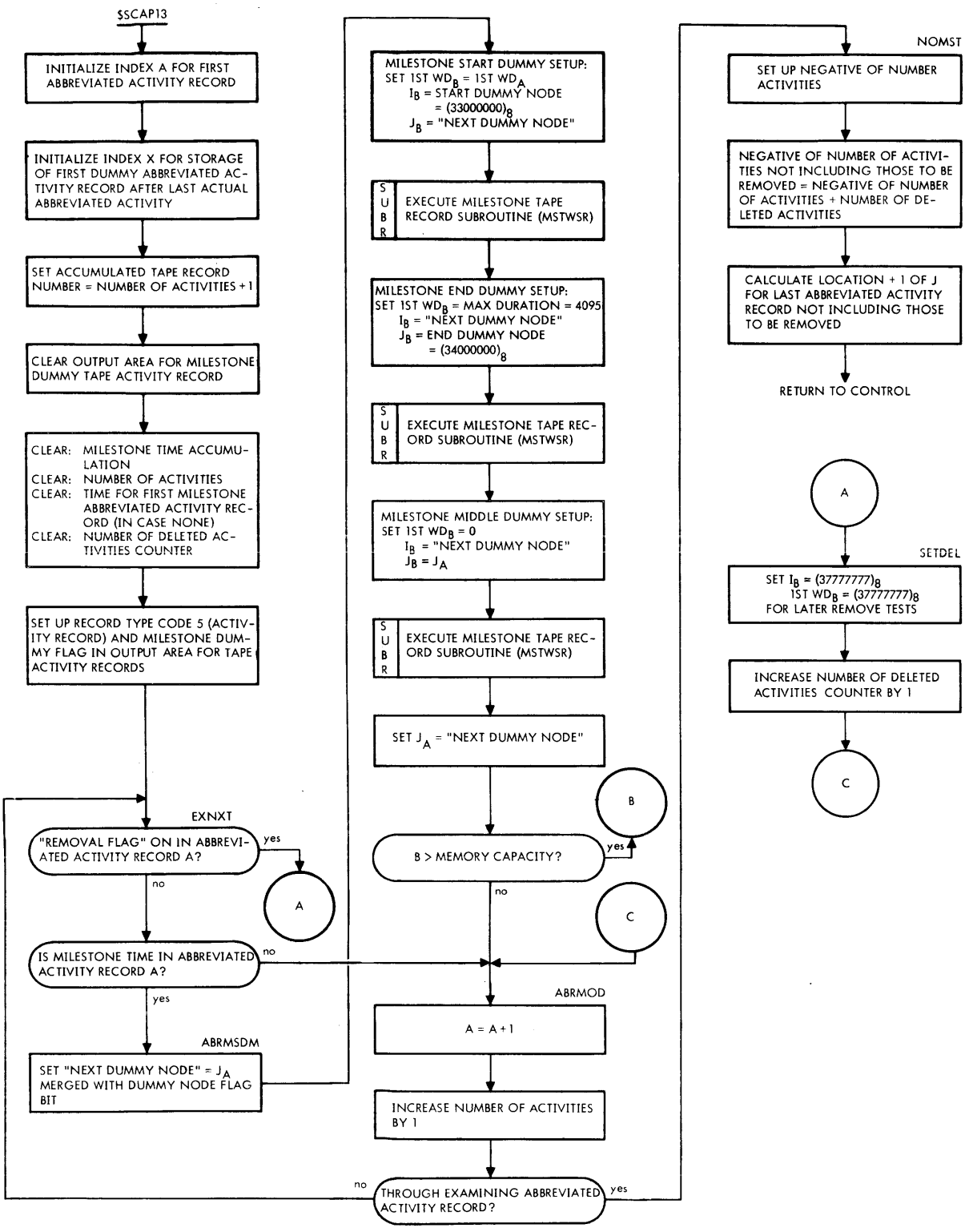
DECIMAL TO BINARY CONVERSION CONTROL



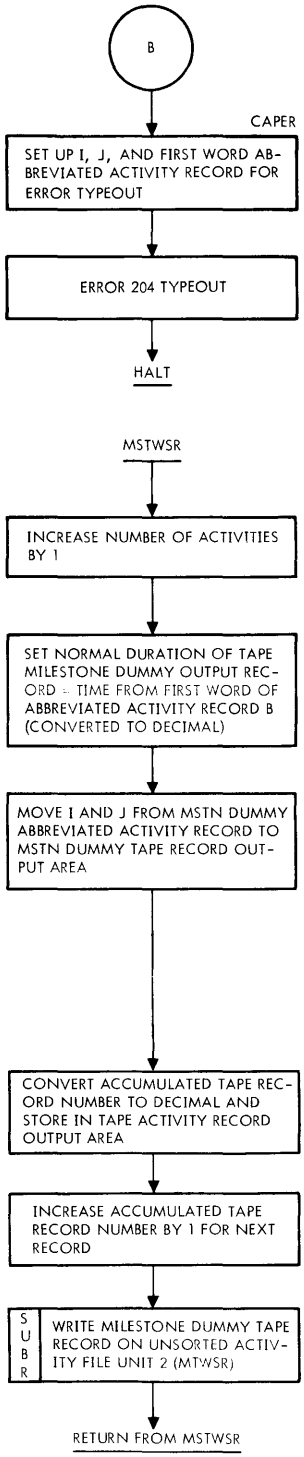
CREATE ABBREVIATED ACTIVITY RECORDS



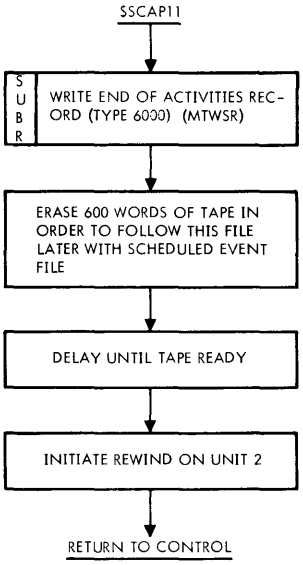
SET UP MILESTONE DUMMY ACTIVE RECORDS



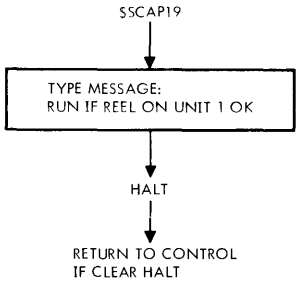
SET UP MILESTONE DUMMY ACTIVITY RECORDS (cont.)



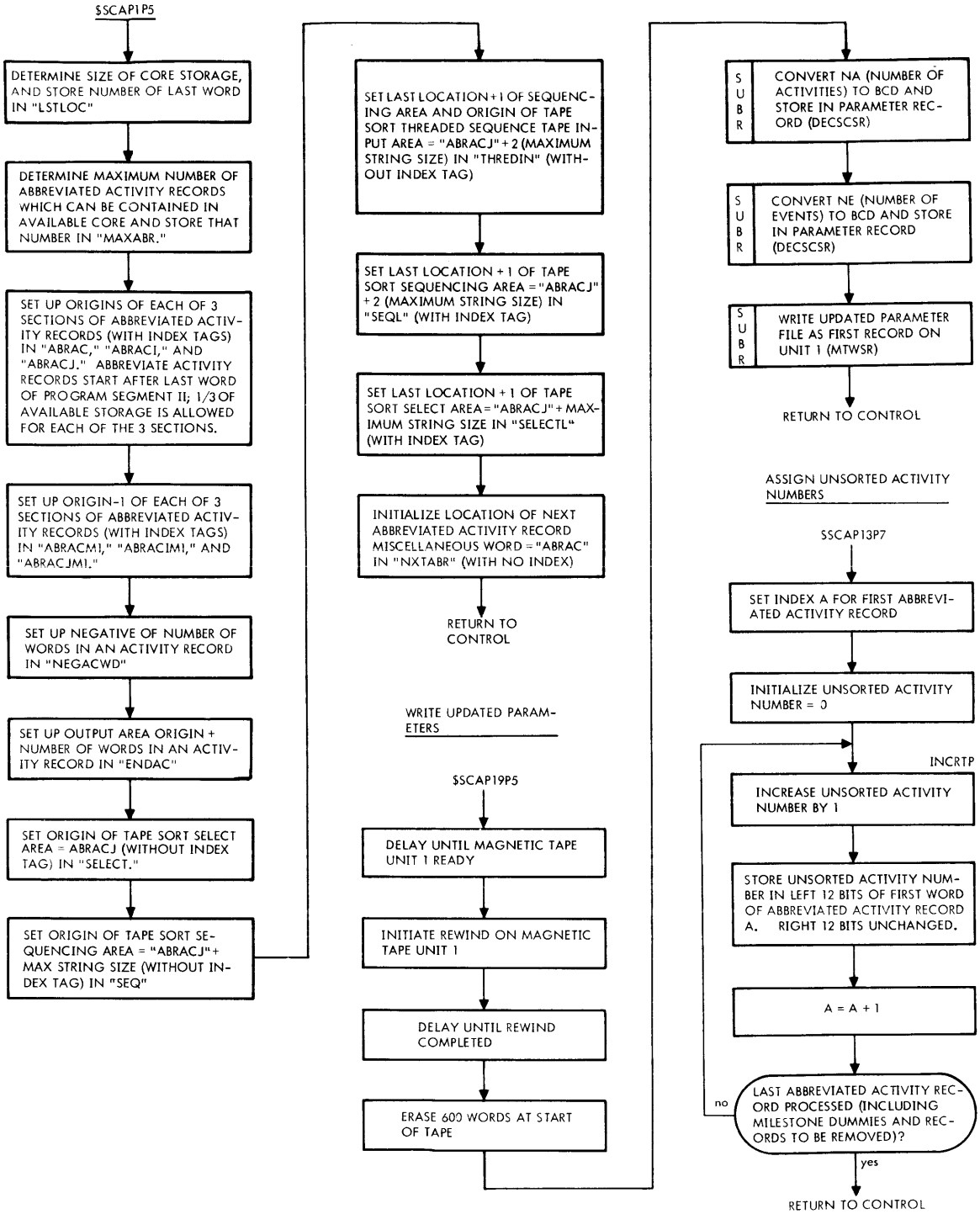
END ACTIVITY FILE AND REWIND UNIT 2



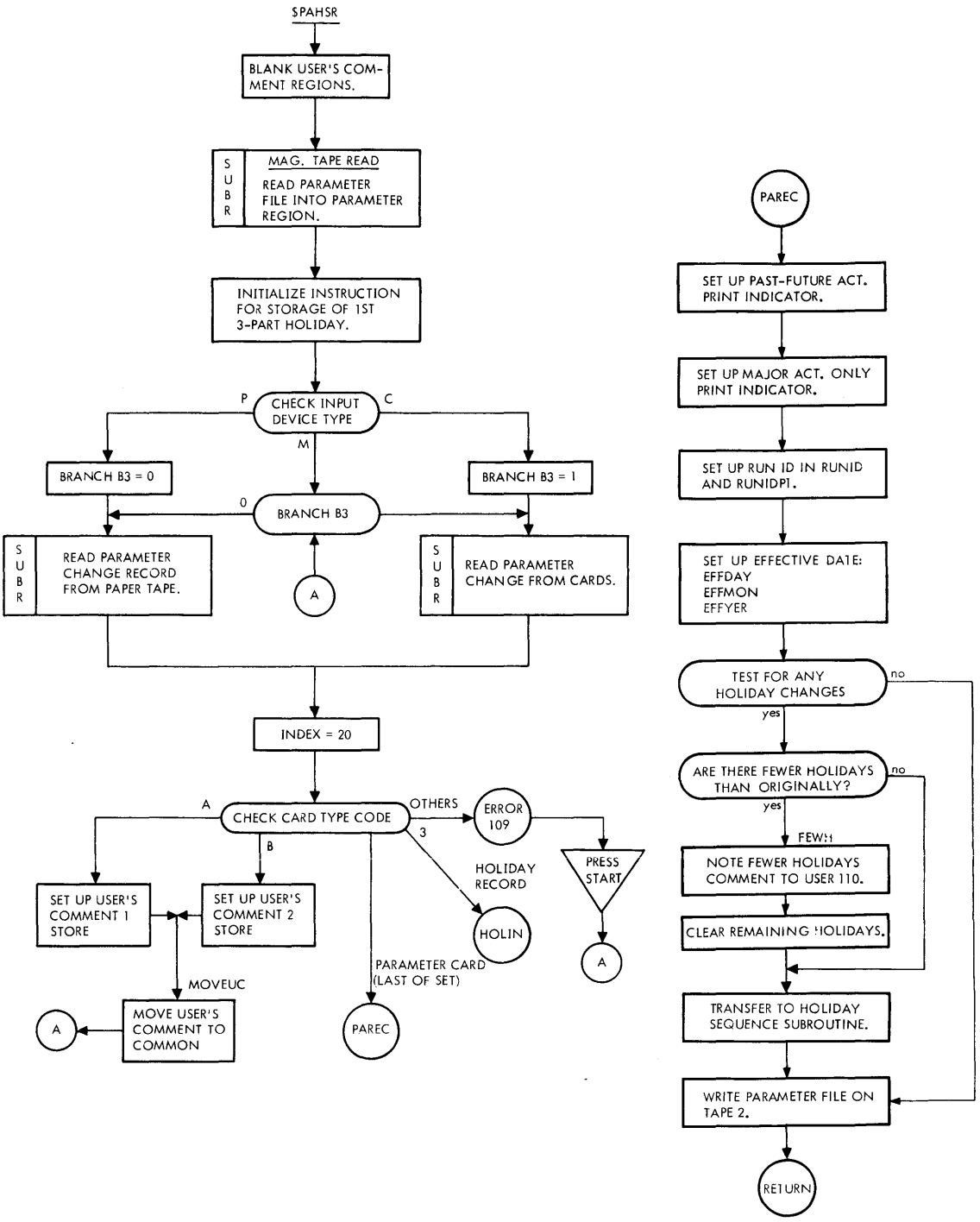
REEL CHANGE VERIFICATION (F.D. F15)

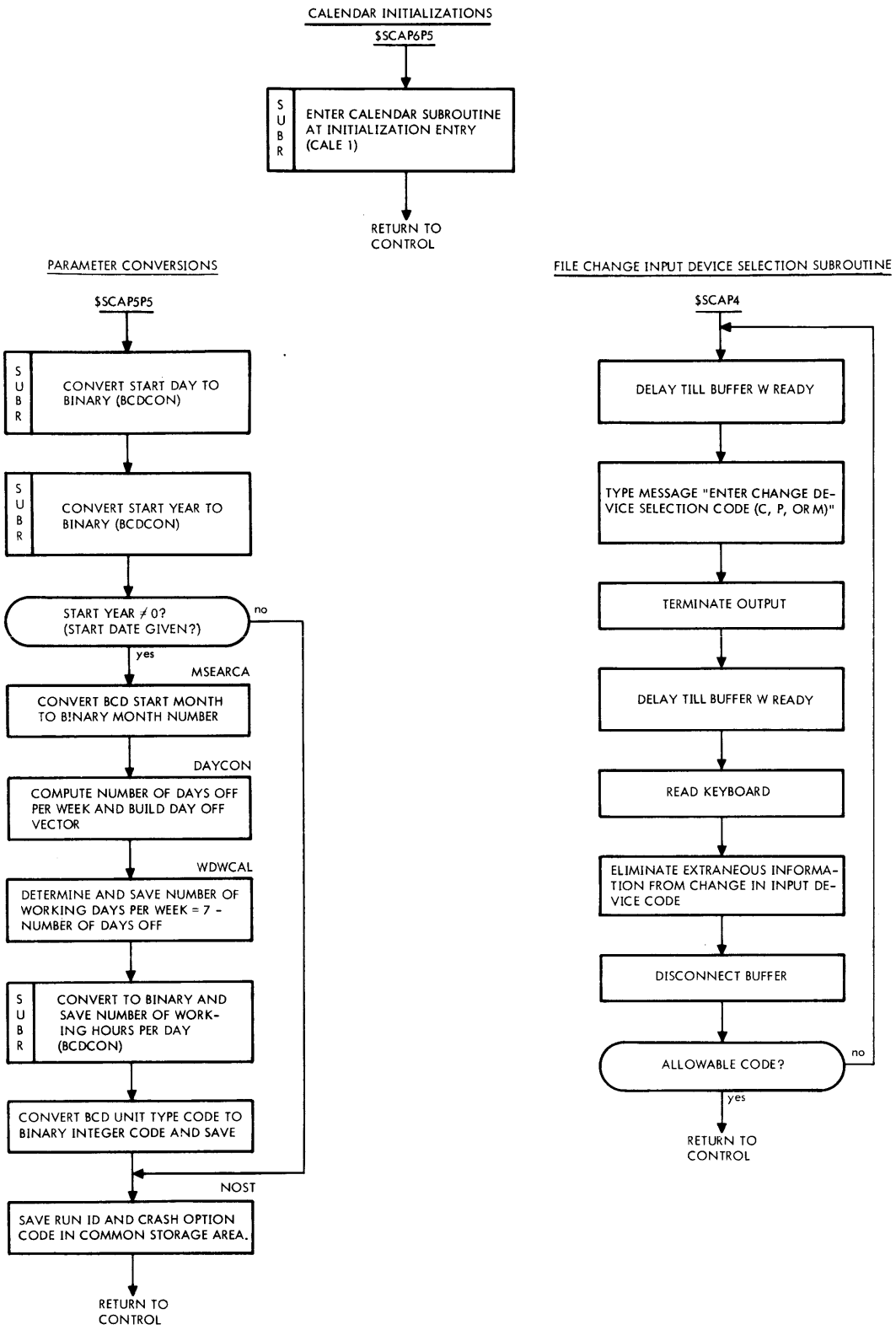


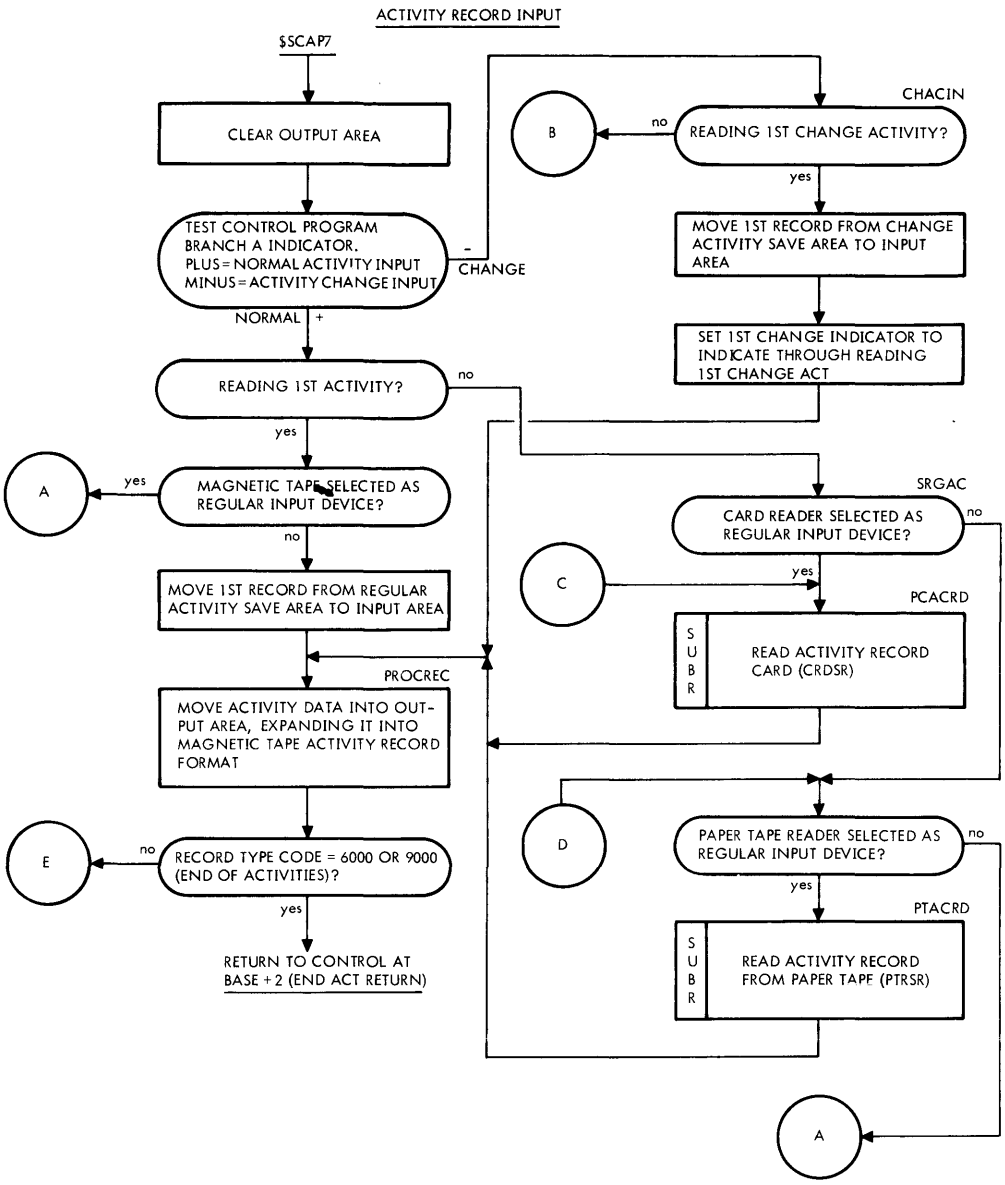
NONACTIVITY ORIENTED INITIALIZATIONS



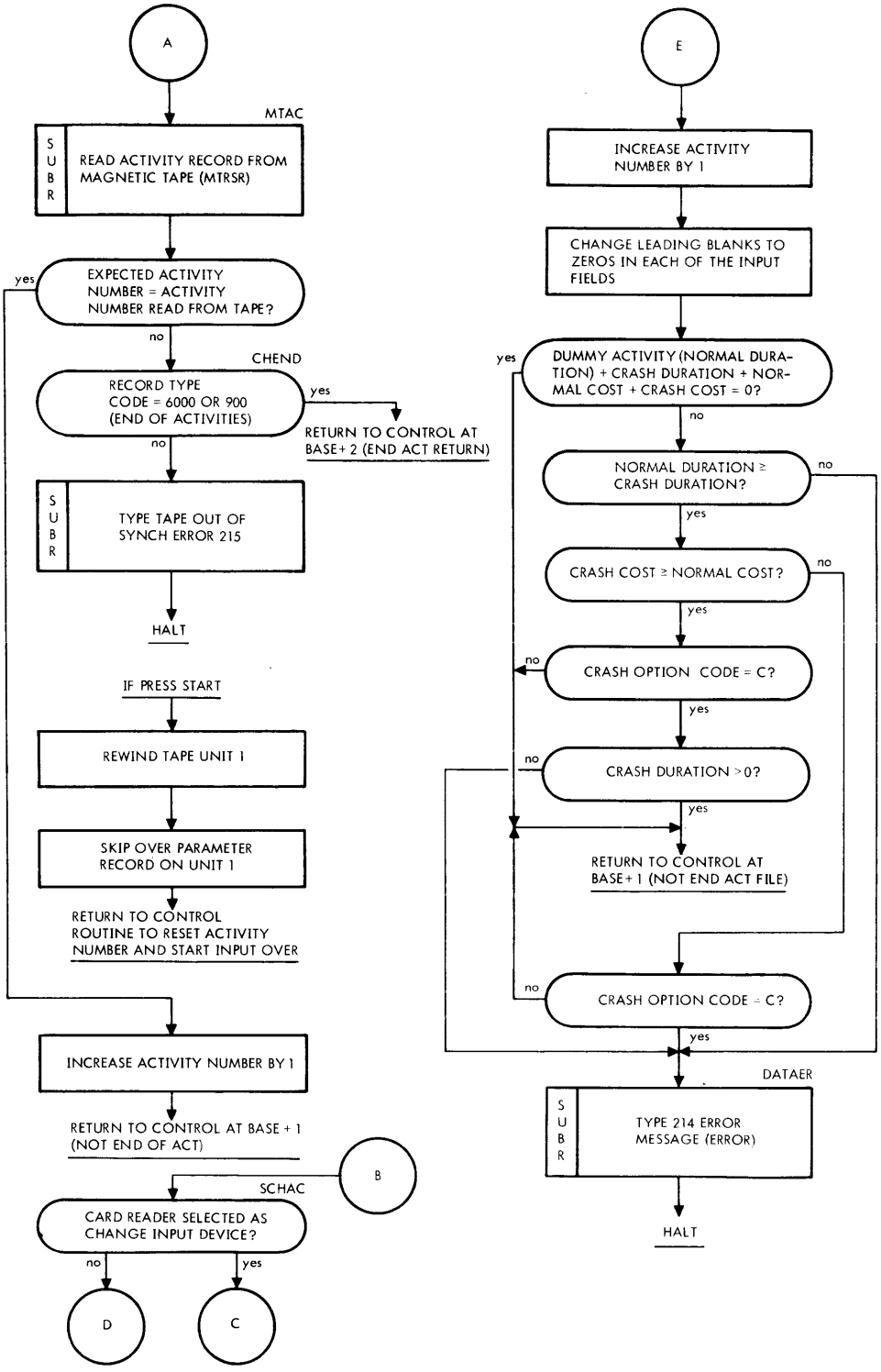
PARAMETER AND HOLIDAY SUBROUTINE (SSP only)

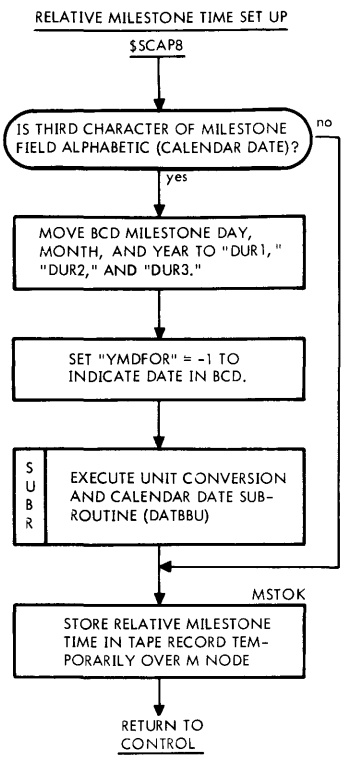


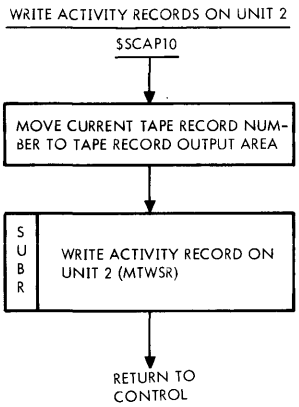

CALENDAR INITIALIZATIONS
$SCAP6P5
SUBR
ENTER CALENDAR SUBROUTINE AT INITIALIZATION ENTRY (CALE 1)
RETURN TO CONTROL
PARAMETER CONVERSIONS
$SCAP5P5
SUBR
CONVERT START DAY TO BINARY (BCDCON)
SUBR
CONVERT START YEAR TO BINARY (BCDCON)
START YEAR ≠ 0? (START DATE GIVEN?)
no
yes
MSEARCA
CONVERT BCD START MONTH TO BINARY MONTH NUMBER
DAYCON
COMPUTE NUMBER OF DAYS OFF PER WEEK AND BUILD DAY OFF VECTOR
WDWCAL
DETERMINE AND SAVE NUMBER OF WORKING DAYS PER WEEK = 7 - NUMBER OF DAYS OFF
SUBR
CONVERT TO BINARY AND SAVE NUMBER OF WORKING HOURS PER DAY (BCDCON)
CONVERT BCD UNIT TYPE CODE TO BINARY INTEGER CODE AND SAVE
NOST
SAVE RUN ID AND CRASH OPTION CODE IN COMMON STORAGE AREA.
RETURN TO CONTROL
FILE CHANGE INPUT DEVICE SELECTION SUBROUTINE
$SCAP4
DELAY TILL BUFFER W READY
TYPE MESSAGE "ENTER CHANGE DEVICE SELECTION CODE (C, P, OR M)"
TERMINATE OUTPUT
DELAY TILL BUFFER W READY
READ KEYBOARD
ELIMINATE EXTRANEOUS INFORMATION FROM CHANGE IN INPUT DEVICE CODE
DISCONNECT BUFFER
ALLOWABLE CODE?
no
yes
RETURN TO CONTROL

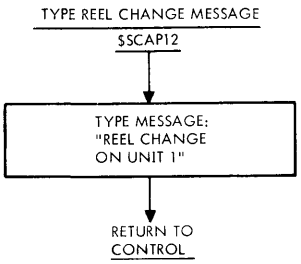

ACTIVITY RECORD INPUT
$SCAP7
CLEAR OUTPUT AREA
TEST CONTROL PROGRAM BRANCH A INDICATOR. PLUS = NORMAL ACTIVITY INPUT MINUS = ACTIVITY CHANGE INPUT
-
CHANGE
NORMAL +
READING 1ST ACTIVITY?
no
yes
A
yes
MAGNETIC TAPE SELECTED AS REGULAR INPUT DEVICE?
no
MOVE 1ST RECORD FROM REGULAR ACTIVITY SAVE AREA TO INPUT AREA
PROCREC
MOVE ACTIVITY DATA INTO OUTPUT AREA, EXPANDING IT INTO MAGNETIC TAPE ACTIVITY RECORD FORMAT
E
no
RECORD TYPE CODE = 6000 OR 9000 (END OF ACTIVITIES)?
yes
RETURN TO CONTROL AT BASE + 2 (END ACT RETURN)
CHACIN
B
no
READING 1ST CHANGE ACTIVITY?
yes
MOVE 1ST RECORD FROM CHANGE ACTIVITY SAVE AREA TO INPUT AREA
SET 1ST CHANGE INDICATOR TO INDICATE THROUGH READING 1ST CHANGE ACT
SRGAC
CARD READER SELECTED AS REGULAR INPUT DEVICE?
no
C
yes
PCACRD
SUBR
READ ACTIVITY RECORD CARD (CRDSR)
D
PAPER TAPE READER SELECTED AS REGULAR INPUT DEVICE?
no
yes
PTACRD
SUBR
READ ACTIVITY RECORD FROM PAPER TAPE (PTRSR)
A

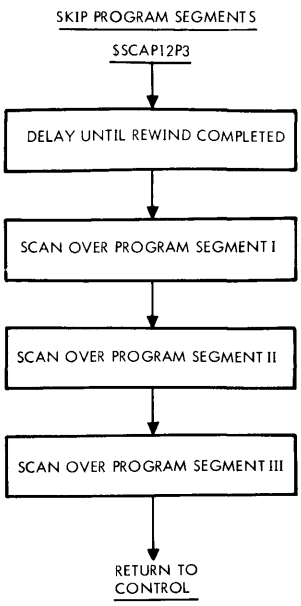
ACTIVITY RECORD INPUT (cont.)

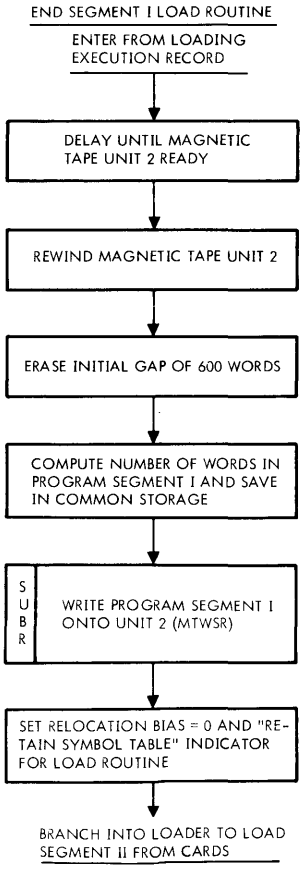



RELATIVE MILESTONE TIME SET UP
$SCAP8
IS THIRD CHARACTER OF MILESTONE FIELD ALPHABETIC (CALENDAR DATE)?
no
yes
MOVE BCD MILESTONE DAY, MONTH, AND YEAR TO "DUR1," "DUR2," AND "DUR3."
SET "YMDFOR" = -1 TO INDICATE DATE IN BCD.
SUBR
EXECUTE UNIT CONVERSION AND CALENDAR DATE SUB-ROUTINE (DATBBU)
MSTOK
STORE RELATIVE MILESTONE TIME IN TAPE RECORD TEMPORARILY OVER M NODE
RETURN TO CONTROL

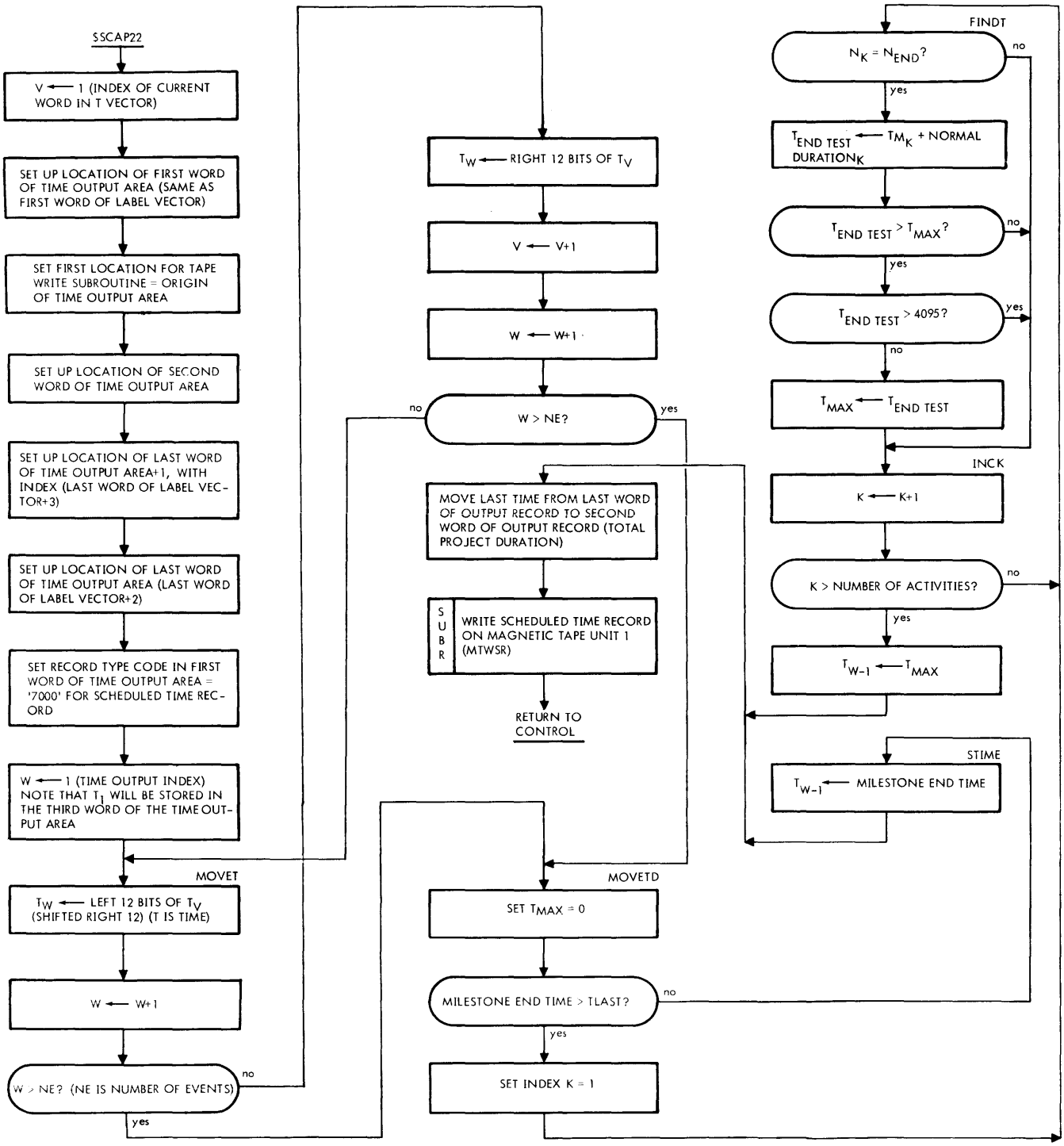

WRITE ACTIVITY RECORDS ON UNIT 2
$SCAP10
MOVE CURRENT TAPE RECORD NUMBER TO TAPE RECORD OUTPUT AREA
SUBR
WRITE ACTIVITY RECORD ON UNIT 2 (MTWSR)
RETURN TO CONTROL


TYPE REEL CHANGE MESSAGE
$SCAP12
TYPE MESSAGE: "REEL CHANGE ON UNIT 1"
RETURN TO CONTROL


SKIP PROGRAM SEGMENTS
$SCAP12P3
DELAY UNTIL REWIND COMPLETED
SCAN OVER PROGRAM SEGMENT I
SCAN OVER PROGRAM SEGMENT II
SCAN OVER PROGRAM SEGMENT III
RETURN TO CONTROL

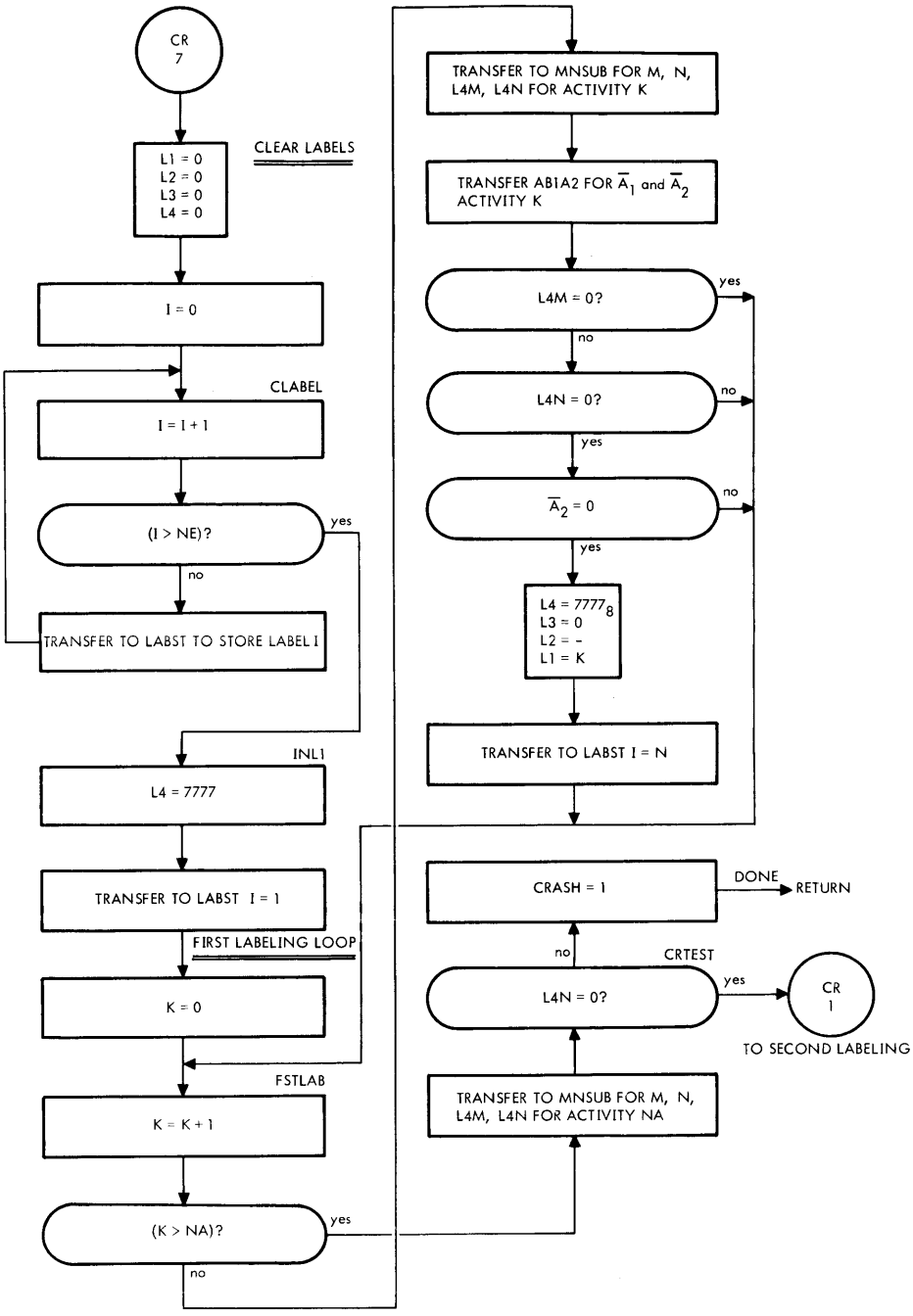

END SEGMENT I LOAD ROUTINE
ENTER FROM LOADING EXECUTION RECORD
DELAY UNTIL MAGNETIC TAPE UNIT 2 READY
REWIND MAGNETIC TAPE UNIT 2
ERASE INITIAL GAP OF 600 WORDS
COMPUTE NUMBER OF WORDS IN PROGRAM SEGMENT I AND SAVE IN COMMON STORAGE
SUBR
WRITE PROGRAM SEGMENT I ONTO UNIT 2 (MTWSR)
SET RELOCATION BIAS = 0 AND "RETAIN SYMBOL TABLE" INDICATOR FOR LOAD ROUTINE
BRANCH INTO LOADER TO LOAD SEGMENT II FROM CARDS

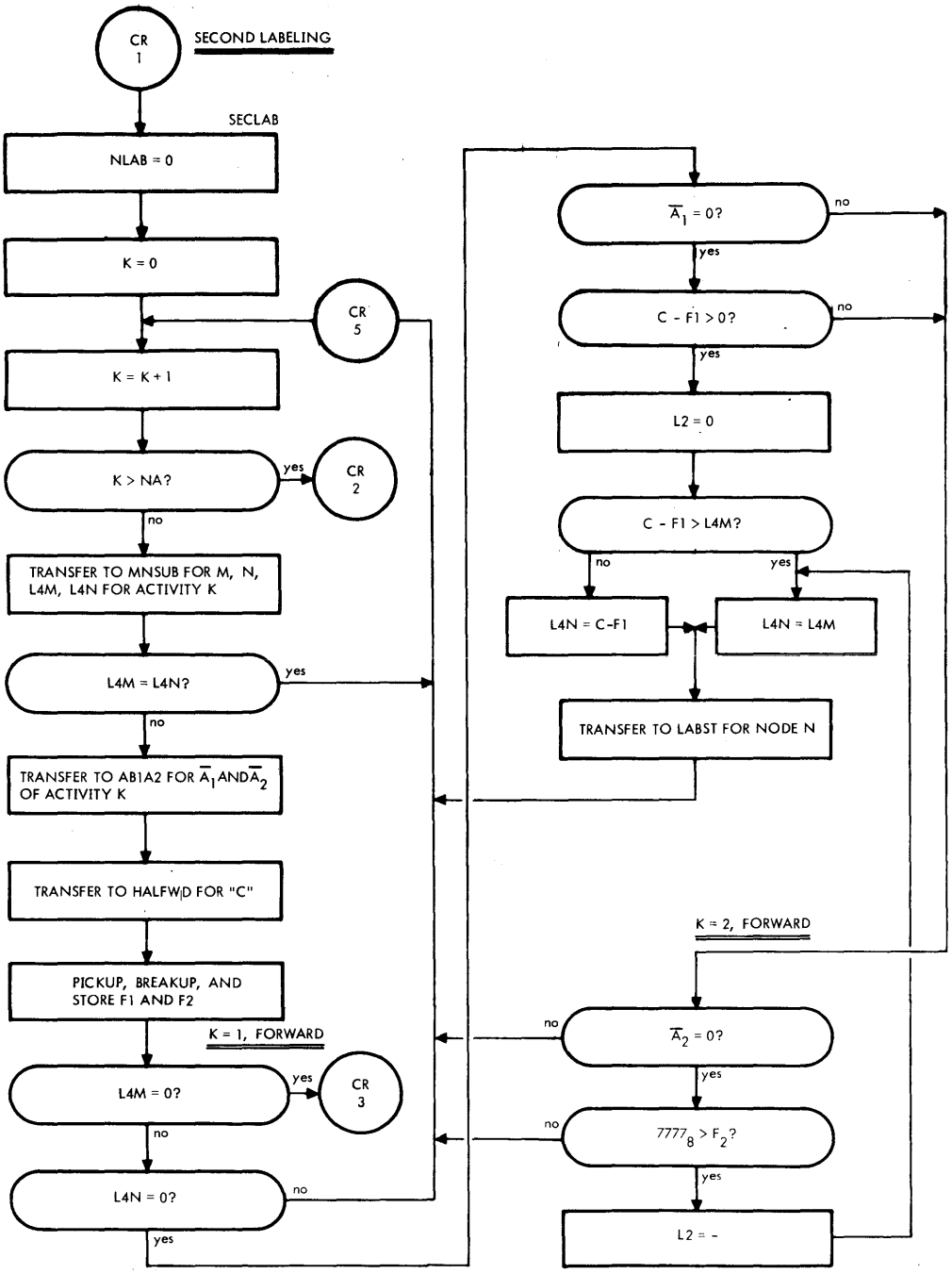
WRITE SCHEDULED EVENT TIME RECORD

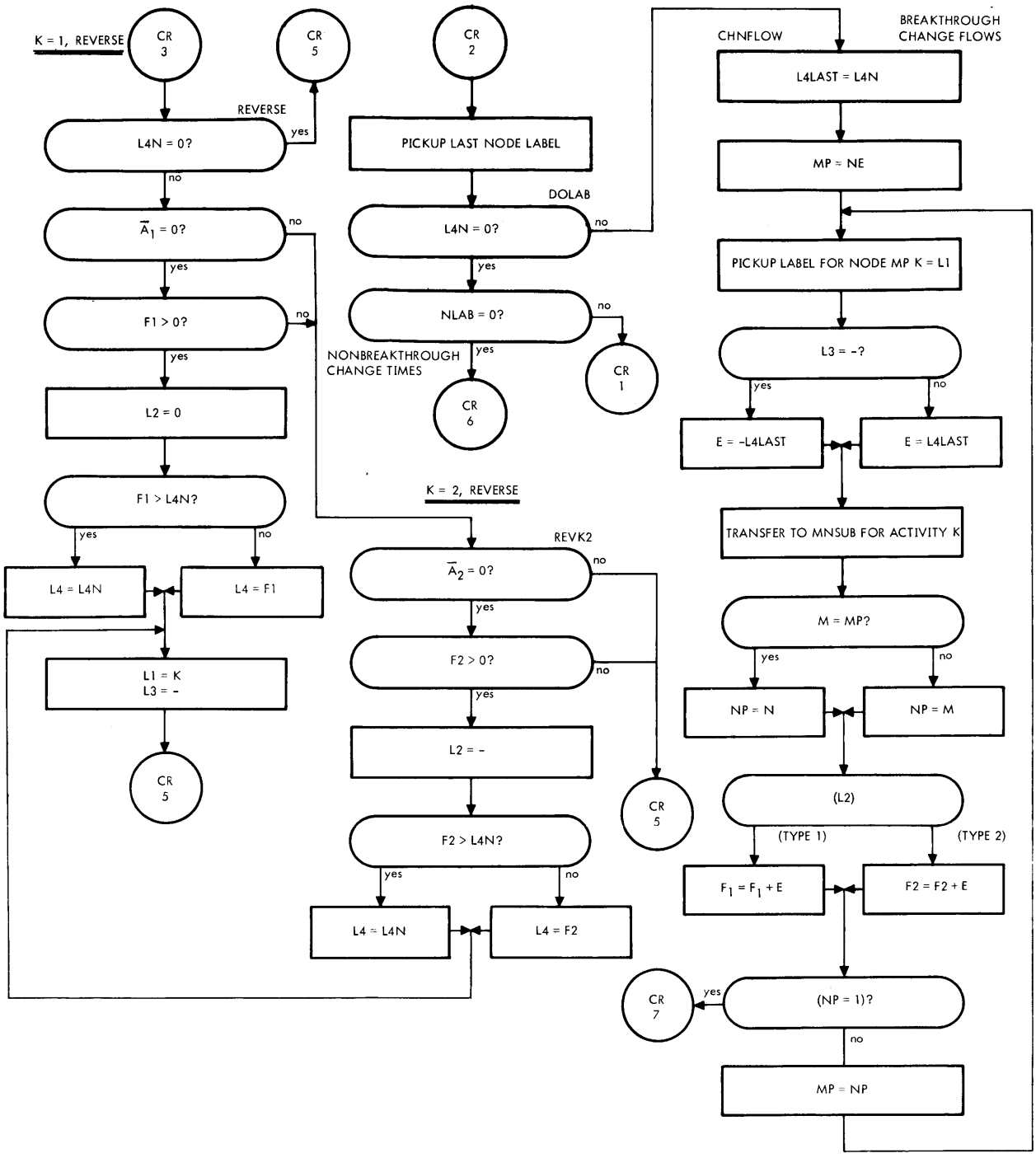


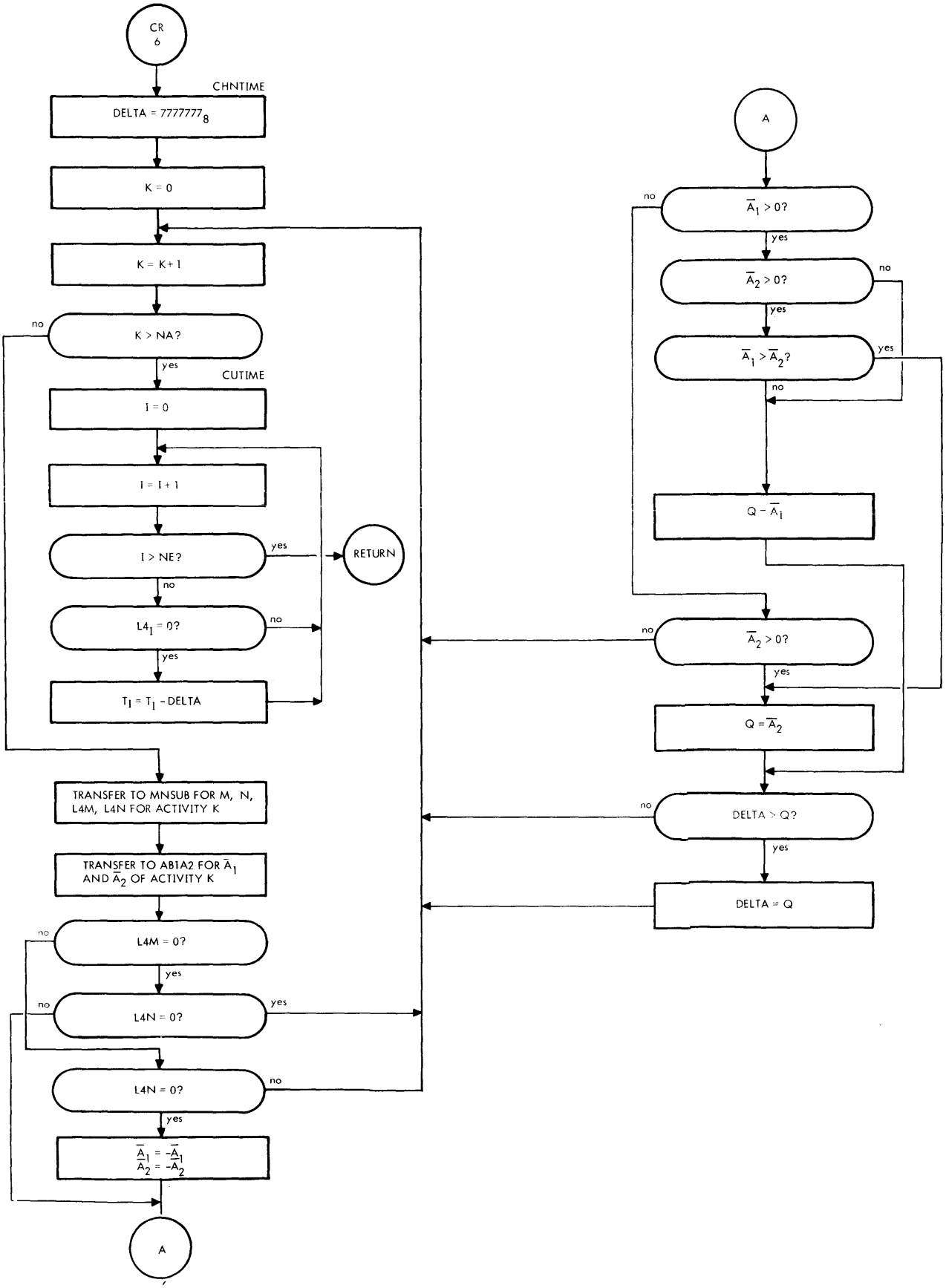
CPM CRASH SUBROUTINE

$CRASHSR

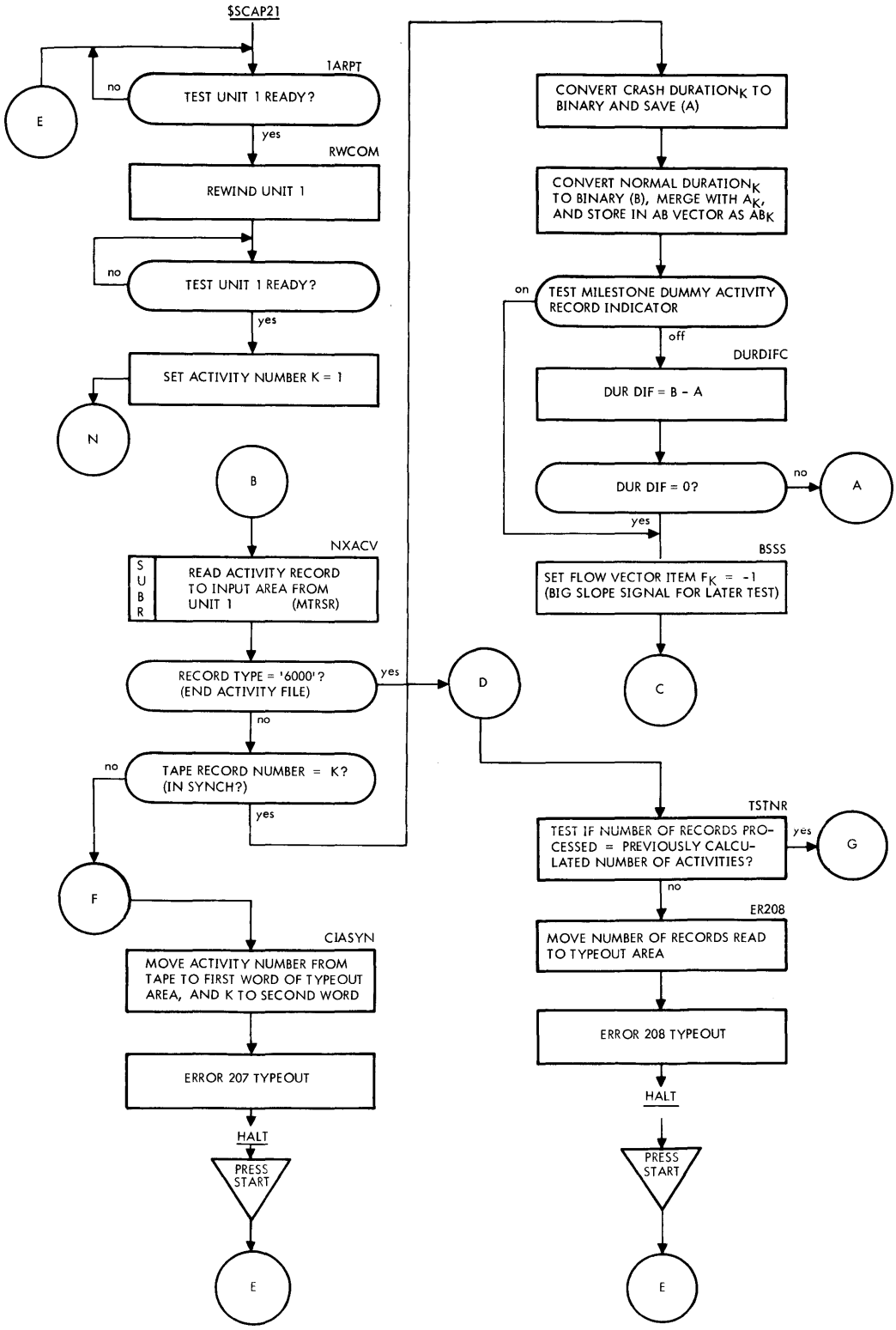


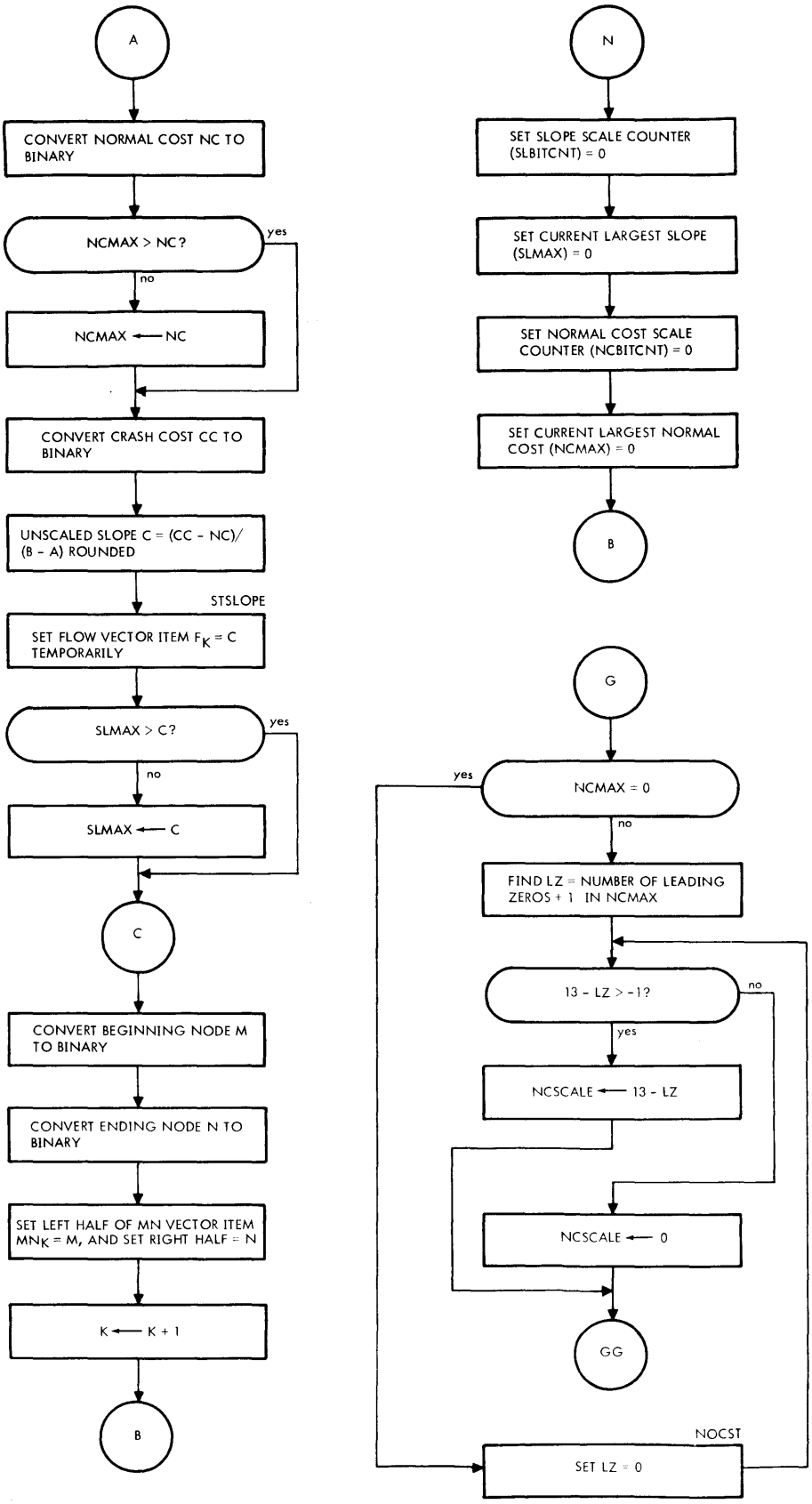

CR 1
SECOND LABELING
SECLAB
NLAB = 0
K = 0
CR 5
K = K + 1
K > NA?
yes
CR 2
no
TRANSFER TO MNSUB FOR M, N, L4M, L4N FOR ACTIVITY K
L4M = L4N?
yes
no
TRANSFER TO AB1A2 FOR $\overline{A}_1$ AND $\overline{A}_2$ OF ACTIVITY K
TRANSFER TO HALFWD FOR "C"
PICKUP, BREAKUP, AND STORE F1 AND F2
K = 1, FORWARD
L4M = 0?
yes
CR 3
no
L4N = 0?
no
yes
$\overline{A}_1$ = 0?
no
yes
C - F1 > 0?
no
yes
L2 = 0
C - F1 > L4M?
no
yes
L4N = C-F1
L4N = L4M
TRANSFER TO LABST FOR NODE N
K = 2, FORWARD
$\overline{A}_2$ = 0?
no
yes
$7777_8 > F_2$?
no
yes
L2 = -

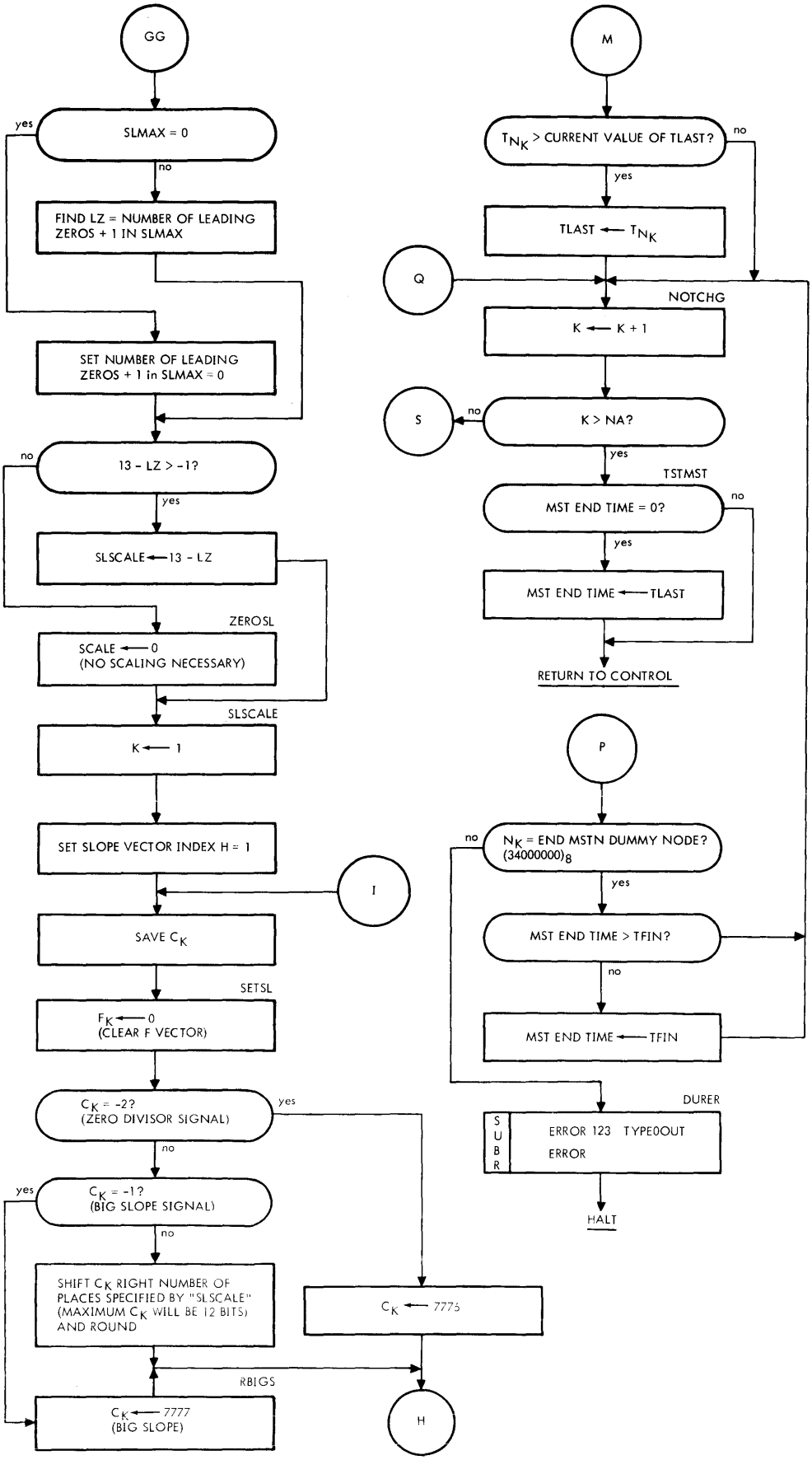

K = 1, REVERSE
CR 3
CR 5
CR 2
REVERSE
L4N = 0?
yes
no
PICKUP LAST NODE LABEL
DOLAB
$\bar{A}_1$ = 0?
L4N = 0?
yes
no
F1 > 0?
NLAB = 0?
NONBREAKTHROUGH CHANGE TIMES
CR 1
CR 6
L2 = 0
F1 > L4N?
K = 2, REVERSE
REVK2
L4 = L4N
L4 = F1
$\bar{A}_2$ = 0?
L1 = K
L3 = -
F2 > 0?
CR 5
L2 = -
F2 > L4N?
L4 = L4N
L4 = F2
CHNFLOW
BREAKTHROUGH CHANGE FLOWS
L4LAST = L4N
MP = NE
PICKUP LABEL FOR NODE MP K = L1
L3 = -?
E = -L4LAST
E = L4LAST
TRANSFER TO MNSUB FOR ACTIVITY K
M = MP?
NP = N
NP = M
(L2)
(TYPE 1)
(TYPE 2)
$F_1 = F_1 + E$
F2 = F2 + E
CR 7
(NP = 1)?
MP = NP

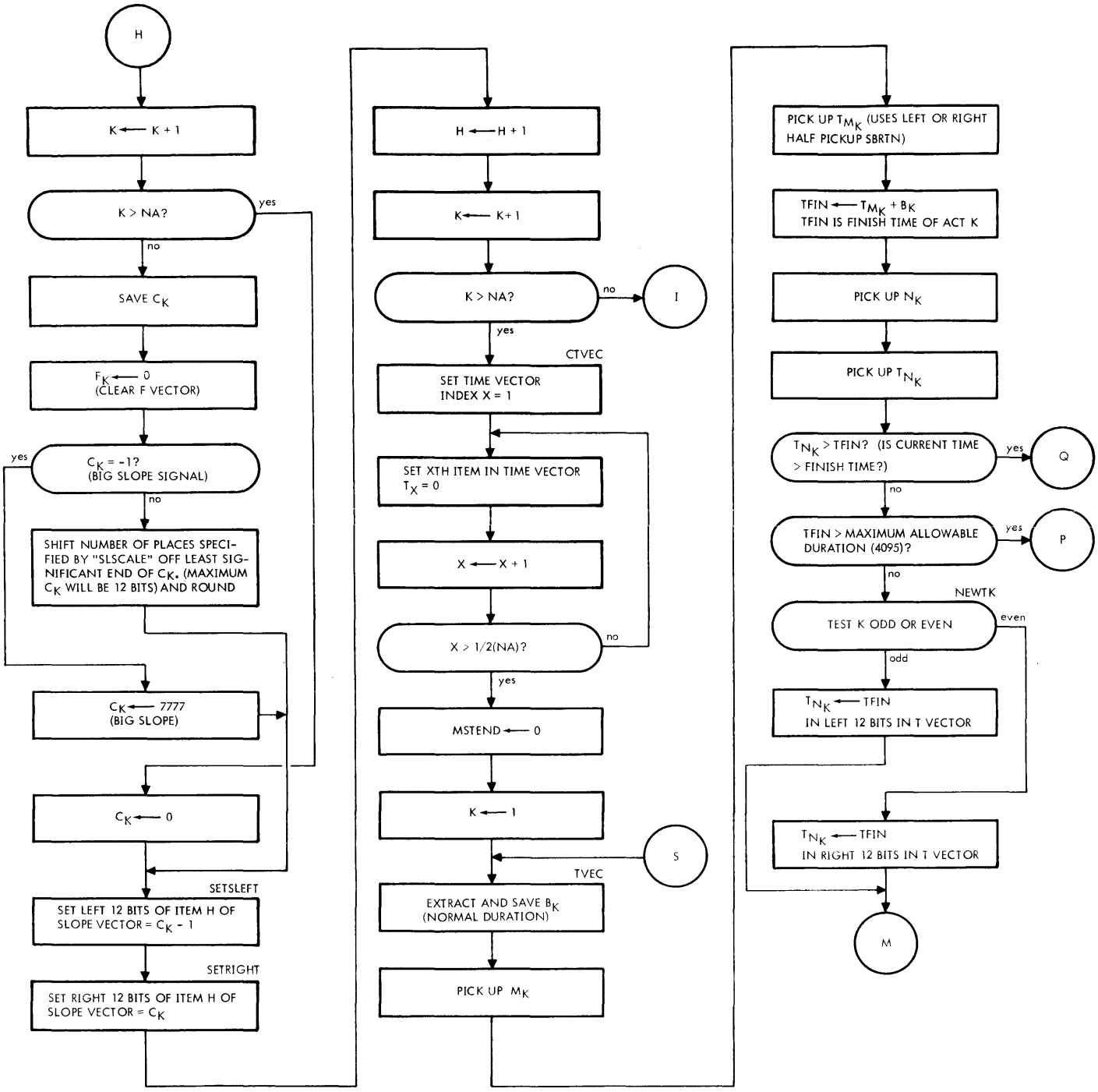

CR 6
CHNTIME
DELTA = 7777777₈
K = 0
K = K+1
no
K > NA?
yes
CUTIME
I = 0
I = I + 1
I > NE?
yes
RETURN
no
L4I = 0?
no
yes
TI = TI - DELTA
TRANSFER TO MNSUB FOR M, N, L4M, L4N FOR ACTIVITY K
TRANSFER TO AB1A2 FOR Ā1 AND Ā2 OF ACTIVITY K
no
L4M = 0?
yes
no
L4N = 0?
yes
L4N = 0?
no
yes
Ā1 = -Ā1
Ā2 = -Ā2
A
A
no
Ā1 > 0?
yes
Ā2 > 0?
no
yes
Ā1 > Ā2?
yes
no
Q = Ā1
no
Ā2 > 0?
yes
Q = Ā2
no
DELTA > Q?
yes
DELTA = Q

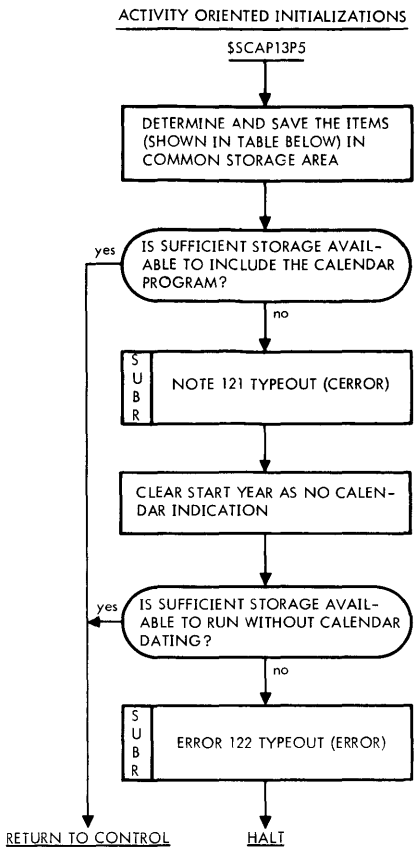
CREATE CRASH DATA ARRAYS




A
CONVERT NORMAL COST NC TO BINARY
NCMAX > NC?
yes
no
NCMAX ← NC
CONVERT CRASH COST CC TO BINARY
UNSCALED SLOPE C = (CC - NC)/ (B - A) ROUNDED
STSLOPE
SET FLOW VECTOR ITEM F_K = C TEMPORARILY
SLMAX > C?
yes
no
SLMAX ← C
C
CONVERT BEGINNING NODE M TO BINARY
CONVERT ENDING NODE N TO BINARY
SET LEFT HALF OF MN VECTOR ITEM MN_K = M, AND SET RIGHT HALF = N
K ← K + 1
B
N
SET SLOPE SCALE COUNTER (SLBITCNT) = 0
SET CURRENT LARGEST SLOPE (SLMAX) = 0
SET NORMAL COST SCALE COUNTER (NCBITCNT) = 0
SET CURRENT LARGEST NORMAL COST (NCMAX) = 0
B
G
yes
NCMAX = 0
no
FIND LZ = NUMBER OF LEADING ZEROS + 1 IN NCMAX
13 - LZ > -1?
no
yes
NCSCALE ← 13 - LZ
NCSCALE ← 0
GG
NOCST
SET LZ = 0

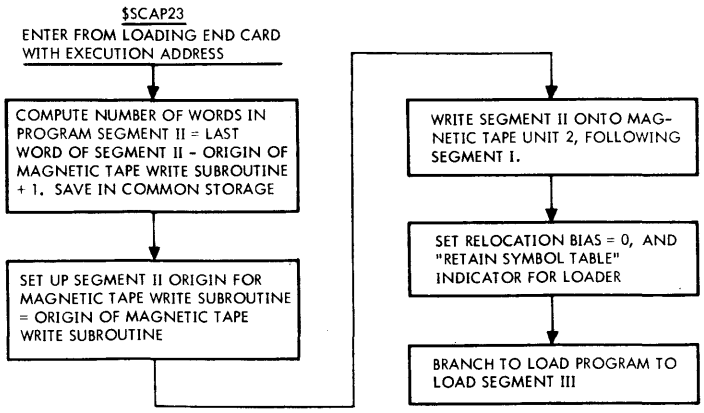

GG
yes
SLMAX = 0
no
FIND LZ = NUMBER OF LEADING ZEROS + 1 IN SLMAX
SET NUMBER OF LEADING ZEROS + 1 in SLMAX = 0
no
13 - LZ > -1?
yes
SLSCALE ← 13 - LZ
ZEROSL
SCALE ← 0
(NO SCALING NECESSARY)
SLSCALE
K ← 1
SET SLOPE VECTOR INDEX H = 1
I
SAVE C_K
SETSL
F_K ← 0
(CLEAR F VECTOR)
C_K = -2?
(ZERO DIVISOR SIGNAL)
yes
no
yes
C_K = -1?
(BIG SLOPE SIGNAL)
no
SHIFT C_K RIGHT NUMBER OF PLACES SPECIFIED BY "SLSCALE" (MAXIMUM C_K WILL BE 12 BITS) AND ROUND
C_K ← 7775
RBIGS
C_K ← 7777
(BIG SLOPE)
H
M
T_{N_K} > CURRENT VALUE OF TLAST?
no
yes
TLAST ← T_{N_K}
Q
NOTCHG
K ← K + 1
S
no
K > NA?
yes
TSTMST
MST END TIME = 0?
no
yes
MST END TIME ← TLAST
RETURN TO CONTROL
P
no
N_K = END MSTN DUMMY NODE?
$(34000000)_8$
yes
MST END TIME > TFIN?
no
MST END TIME ← TFIN
DURER
SUBR
ERROR 123 TYPEOOUT
ERROR
HALT


H
K ← K + 1
K > NA?
yes
no
SAVE C_K
F_K ← 0
(CLEAR F VECTOR)
C_K = -1?
(BIG SLOPE SIGNAL)
yes
no
SHIFT NUMBER OF PLACES SPECIFIED BY "SLSCALE" OFF LEAST SIGNIFICANT END OF C_K. (MAXIMUM C_K WILL BE 12 BITS) AND ROUND
C_K ← 7777
(BIG SLOPE)
C_K ← 0
SETSLEFT
SET LEFT 12 BITS OF ITEM H OF SLOPE VECTOR = C_K - 1
SETRIGHT
SET RIGHT 12 BITS OF ITEM H OF SLOPE VECTOR = C_K
H ← H + 1
K ← K + 1
K > NA?
no
I
yes
CTVEC
SET TIME VECTOR INDEX X = 1
SET XTH ITEM IN TIME VECTOR T_X = 0
X ← X + 1
X > 1/2(NA)?
no
yes
MSTEND ← 0
K ← 1
S
TVEC
EXTRACT AND SAVE B_K (NORMAL DURATION)
PICK UP M_K
PICK UP T_{M_K} (USES LEFT OR RIGHT HALF PICKUP SBRTN)
TFIN ← T_{M_K} + B_K
TFIN IS FINISH TIME OF ACT K
PICK UP N_K
PICK UP T_{N_K}
T_{N_K} > TFIN? (IS CURRENT TIME > FINISH TIME?)
yes
Q
no
TFIN > MAXIMUM ALLOWABLE DURATION (4095)?
yes
P
no
NEWTK
TEST K ODD OR EVEN
even
odd
T_{N_K} ← TFIN
IN LEFT 12 BITS IN T VECTOR
T_{N_K} ← TFIN
IN RIGHT 12 BITS IN T VECTOR
M

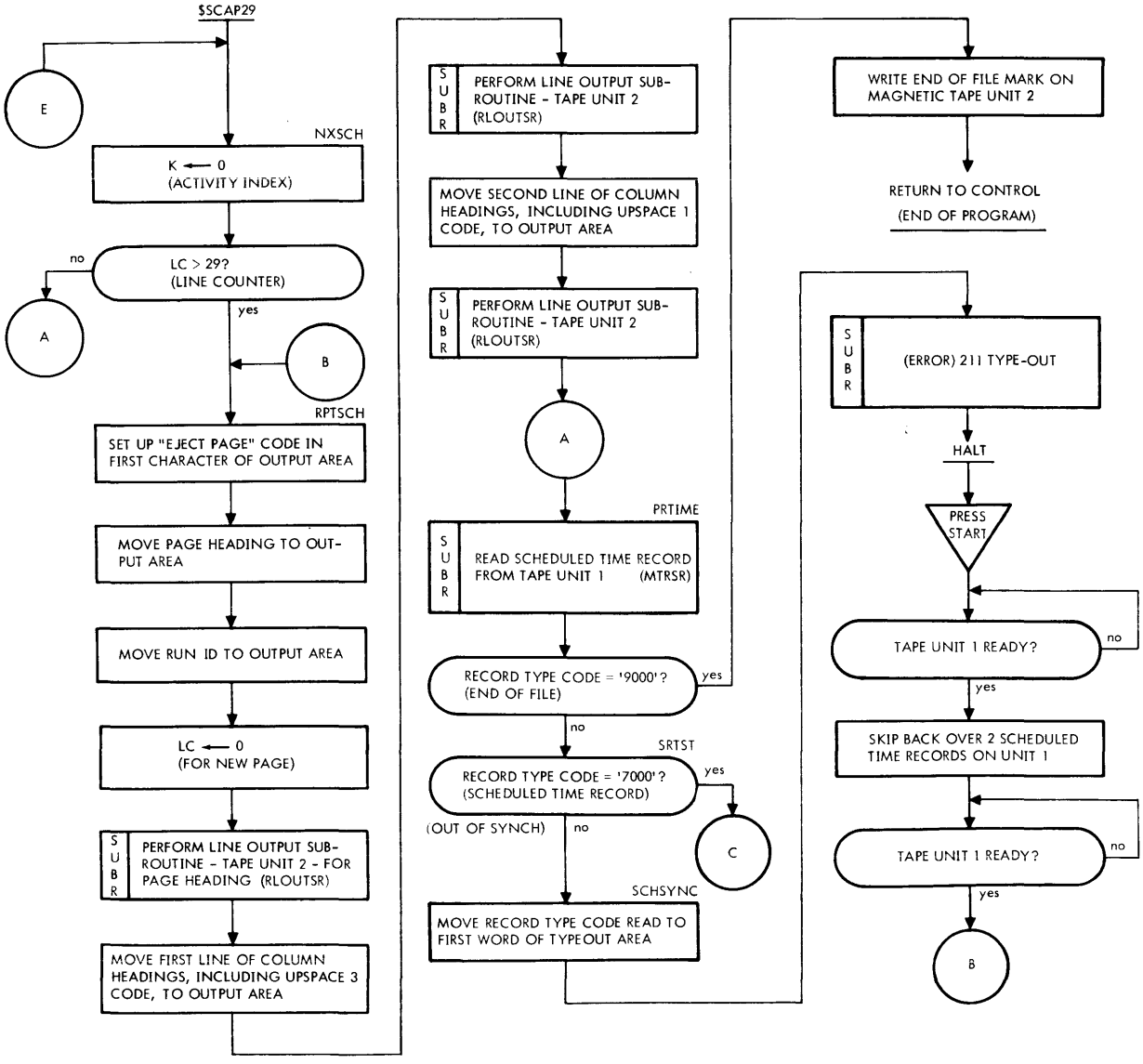


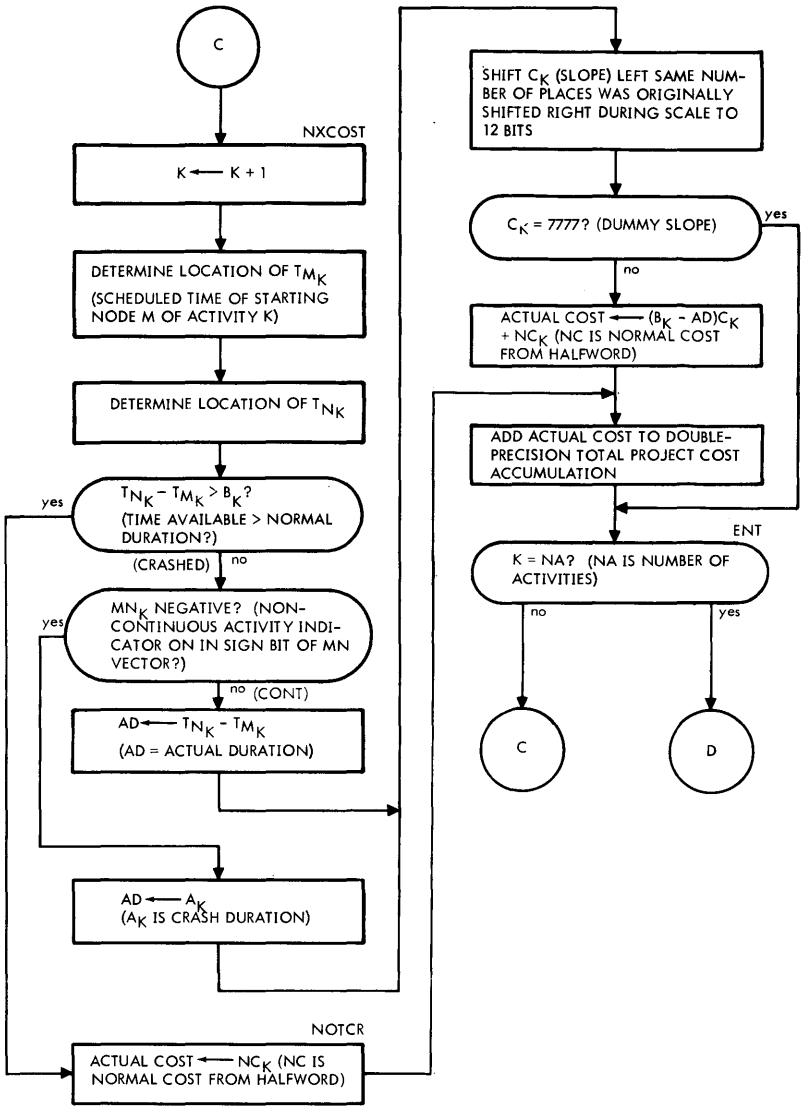
HNA	One-Half Number of Activities Rounded	
NE	Number of Events	
HNE	One-Half Number of Events Rounded	
NEP2	Number of Events Plus 2	
HNGNE	Negative of HNE	
NGNE	Negative of NE	
NGNA	Negative of Number of Activities	
MNOFF	Origin-1 of MN Vector	With Index Tag
ABOFF	Origin-1 of Normal and Crash Duration Vector	With Index Tag
COFF	Origin-1 of Slope Vector	With Index Tag
TOFF	Origin-1 of Node Time Vector	With Index Tag
FOFF	Origin-1 of Utility Flow Vector	With Index Tag
LABOFF	Origin-1 of Label Vector	With Index Tag
FEND	Last+1 Location of Utility Flow Vector	With Index Tag
TEND	Last+1 Location of Node Time Vector	With Index Tag
SLOPEIN	Origin of Slope Vector - No Index Tab	
TIME	Origin of Node Time Vector – No Index Tag	
NCOFF	Origin-1 of Normal Cost Vector – With Index Tag	
NCORG	Origin of Normal Cost Vector – No Index Tag	
SCHED	Origin of Scheduled Time Vector – No Index Tag	
SCHEDOFF	Origin-1 of Scheduled Time Vector – With Index Tag	
SCHIN	Origin of Magnetic Tape Scheduled Time File	

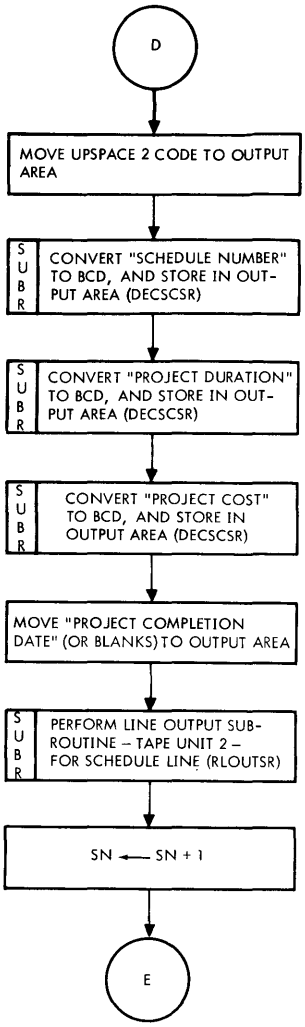
END SEGMENT II LOAD PROGRAM

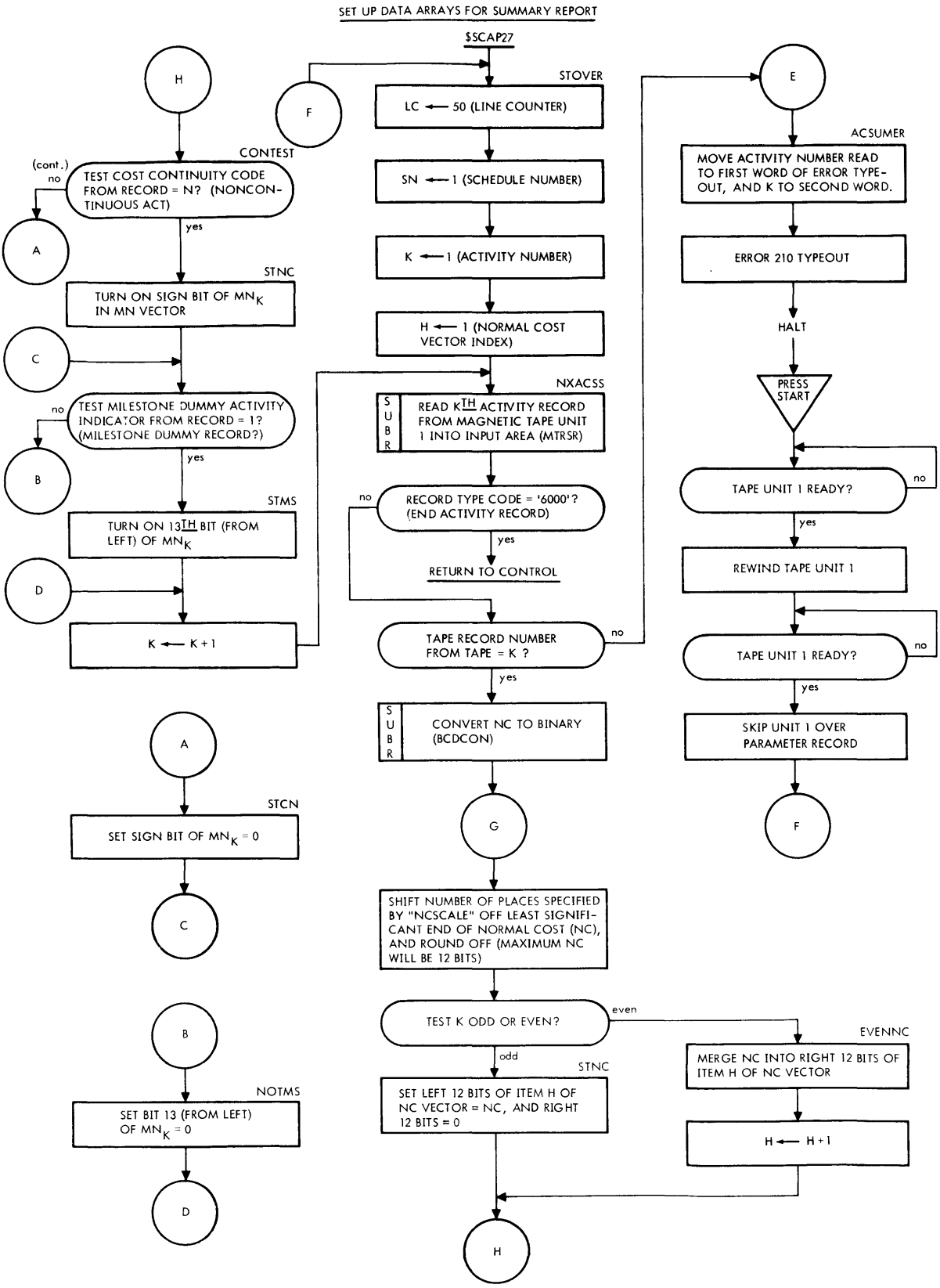


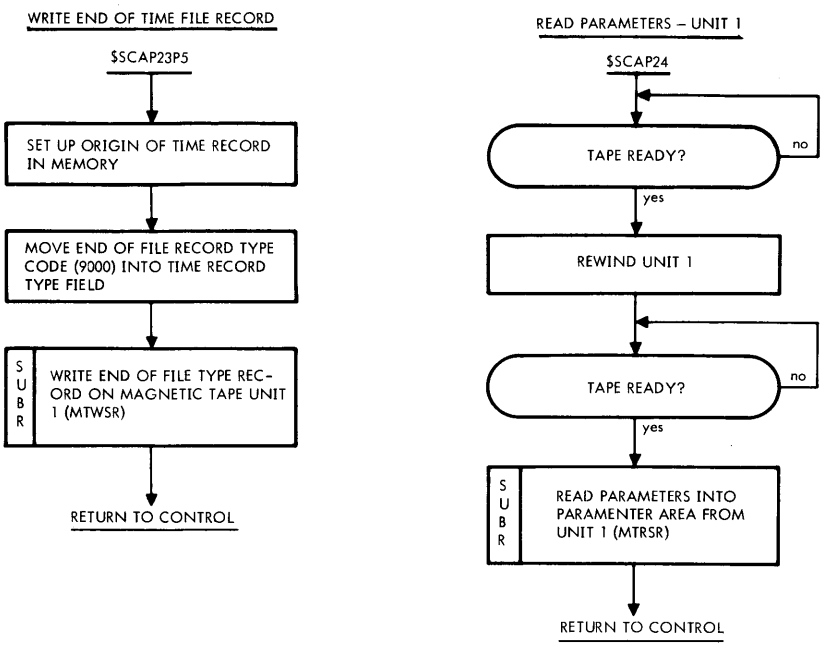
SCHEDULE SUMMARY REPORT OUTPUT

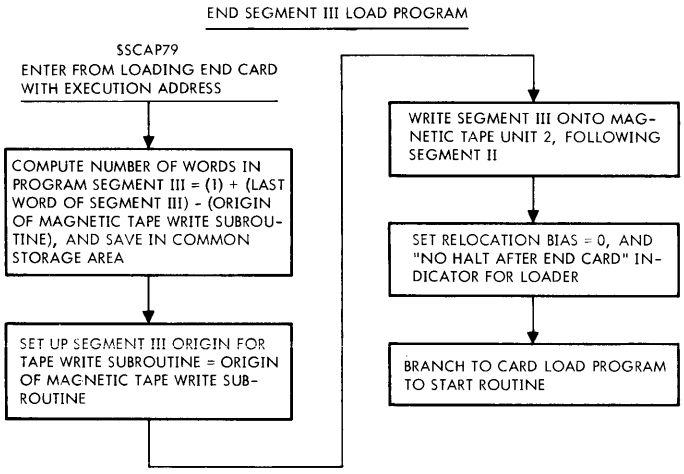


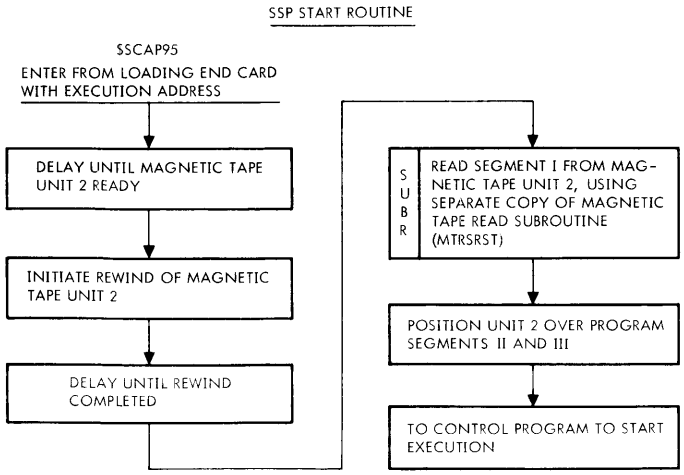

C
NXCOST
K ← K + 1
DETERMINE LOCATION OF T_{M_K} (SCHEDULED TIME OF STARTING NODE M OF ACTIVITY K)
DETERMINE LOCATION OF T_{N_K}
$T_{N_K} - T_{M_K} > B_K$? (TIME AVAILABLE > NORMAL DURATION?)
yes
(CRASHED) no
MN_K NEGATIVE? (NON-CONTINUOUS ACTIVITY INDICATOR ON IN SIGN BIT OF MN VECTOR?)
yes
no (CONT)
AD ← $T_{N_K} - T_{M_K}$ (AD = ACTUAL DURATION)
AD ← A_K (A_K IS CRASH DURATION)
NOTCR
ACTUAL COST ← NC_K (NC IS NORMAL COST FROM HALFWORD)
SHIFT C_K (SLOPE) LEFT SAME NUMBER OF PLACES WAS ORIGINALLY SHIFTED RIGHT DURING SCALE TO 12 BITS
C_K = 7777? (DUMMY SLOPE)
yes
no
ACTUAL COST ← $(B_K - AD)C_K + NC_K$ (NC IS NORMAL COST FROM HALFWORD)
ADD ACTUAL COST TO DOUBLE-PRECISION TOTAL PROJECT COST ACCUMULATION
ENT
K = NA? (NA IS NUMBER OF ACTIVITIES)
no
yes
C
D


D
MOVE UPSPACE 2 CODE TO OUTPUT AREA
SUBR
CONVERT "SCHEDULE NUMBER" TO BCD, AND STORE IN OUTPUT AREA (DECSCSR)
SUBR
CONVERT "PROJECT DURATION" TO BCD, AND STORE IN OUTPUT AREA (DECSCSR)
SUBR
CONVERT "PROJECT COST" TO BCD, AND STORE IN OUTPUT AREA (DECSCSR)
MOVE "PROJECT COMPLETION DATE" (OR BLANKS) TO OUTPUT AREA
SUBR
PERFORM LINE OUTPUT SUBROUTINE – TAPE UNIT 2 – FOR SCHEDULE LINE (RLOUTSR)
SN ← SN + 1
E

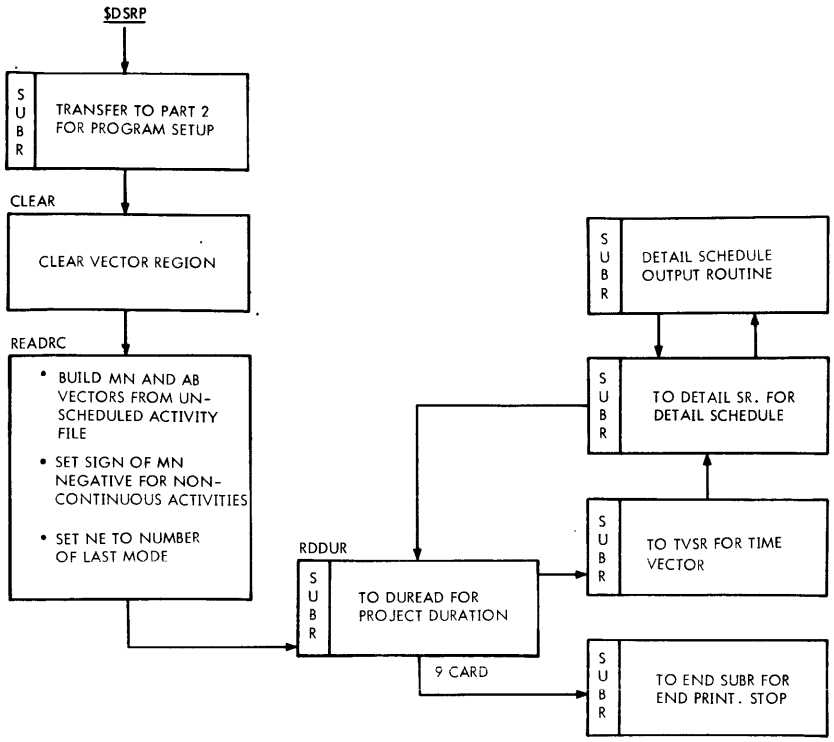

SET UP DATA ARRAYS FOR SUMMARY REPORT
$SCAP27
STOVER
LC ← 50 (LINE COUNTER)
SN ← 1 (SCHEDULE NUMBER)
K ← 1 (ACTIVITY NUMBER)
H ← 1 (NORMAL COST VECTOR INDEX)
NXACSS
SUBR
READ K TH ACTIVITY RECORD FROM MAGNETIC TAPE UNIT 1 INTO INPUT AREA (MTRSR)
RECORD TYPE CODE = '6000'? (END ACTIVITY RECORD)
no
yes
RETURN TO CONTROL
TAPE RECORD NUMBER FROM TAPE = K ?
no
yes
SUBR
CONVERT NC TO BINARY (BCDCON)
SHIFT NUMBER OF PLACES SPECIFIED BY "NCSCALE" OFF LEAST SIGNIFICANT END OF NORMAL COST (NC), AND ROUND OFF (MAXIMUM NC WILL BE 12 BITS)
TEST K ODD OR EVEN?
even
odd
STNC
SET LEFT 12 BITS OF ITEM H OF NC VECTOR = NC, AND RIGHT 12 BITS = 0
EVENNC
MERGE NC INTO RIGHT 12 BITS OF ITEM H OF NC VECTOR
H ← H+1
H
F
E
ACSUMER
MOVE ACTIVITY NUMBER READ TO FIRST WORD OF ERROR TYPE-OUT, AND K TO SECOND WORD.
ERROR 210 TYPEOUT
HALT
PRESS START
TAPE UNIT 1 READY?
no
yes
REWIND TAPE UNIT 1
TAPE UNIT 1 READY?
no
yes
SKIP UNIT 1 OVER PARAMETER RECORD
F
H
CONTEST
(cont.)
no
TEST COST CONTINUITY CODE FROM RECORD = N? (NONCONTINUOUS ACT)
yes
A
STNC
TURN ON SIGN BIT OF MN K IN MN VECTOR
C
no
TEST MILESTONE DUMMY ACTIVITY INDICATOR FROM RECORD = 1? (MILESTONE DUMMY RECORD?)
yes
B
STMS
TURN ON 13 TH BIT (FROM LEFT) OF MN K
D
K ← K+1
A
STCN
SET SIGN BIT OF MN K = 0
C
B
NOTMS
SET BIT 13 (FROM LEFT) OF MN K = 0
D


WRITE END OF TIME FILE RECORD
$SCAP23P5
SET UP ORIGIN OF TIME RECORD IN MEMORY
MOVE END OF FILE RECORD TYPE CODE (9000) INTO TIME RECORD TYPE FIELD
SUBR
WRITE END OF FILE TYPE RECORD ON MAGNETIC TAPE UNIT 1 (MTWSR)
RETURN TO CONTROL
READ PARAMETERS – UNIT 1
$SCAP24
TAPE READY?
no
yes
REWIND UNIT 1
TAPE READY?
no
yes
SUBR
READ PARAMETERS INTO PARAMENTER AREA FROM UNIT 1 (MTRSR)
RETURN TO CONTROL

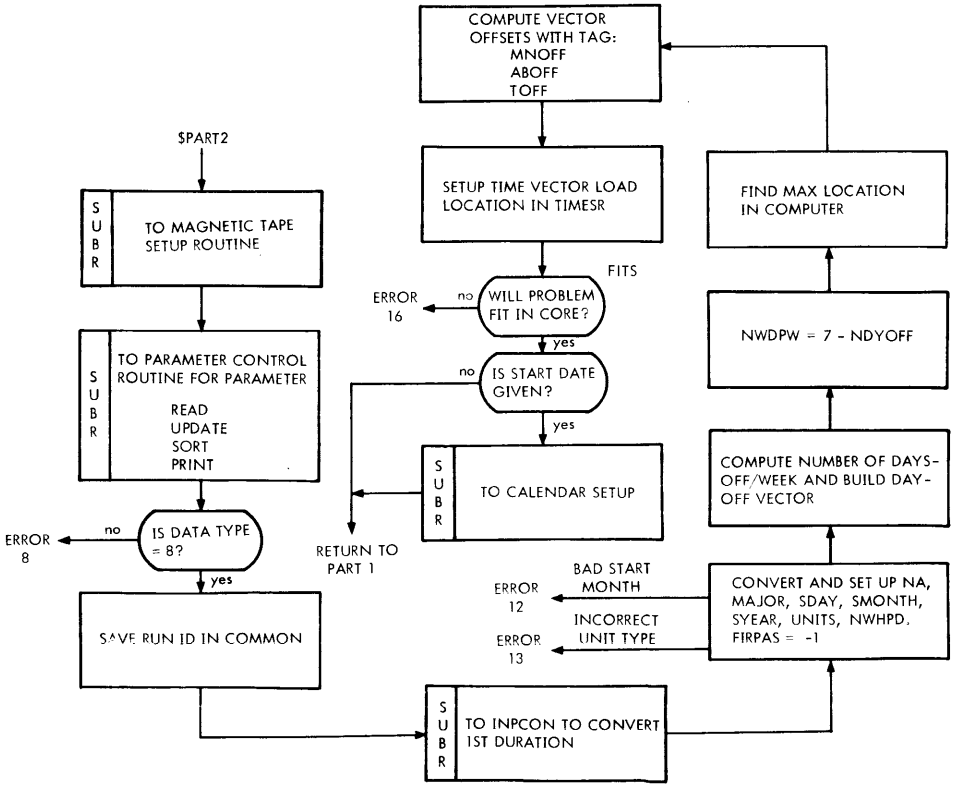

END SEGMENT III LOAD PROGRAM
$SCAP79
ENTER FROM LOADING END CARD WITH EXECUTION ADDRESS
COMPUTE NUMBER OF WORDS IN PROGRAM SEGMENT III = (1) + (LAST WORD OF SEGMENT III) - (ORIGIN OF MAGNETIC TAPE WRITE SUBROUTINE), AND SAVE IN COMMON STORAGE AREA
SET UP SEGMENT III ORIGIN FOR TAPE WRITE SUBROUTINE = ORIGIN OF MAGNETIC TAPE WRITE SUBROUTINE
WRITE SEGMENT III ONTO MAGNETIC TAPE UNIT 2, FOLLOWING SEGMENT II
SET RELOCATION BIAS = 0, AND "NO HALT AFTER END CARD" INDICATOR FOR LOADER
BRANCH TO CARD LOAD PROGRAM TO START ROUTINE


SSP START ROUTINE
$SCAP95
ENTER FROM LOADING END CARD WITH EXECUTION ADDRESS
DELAY UNTIL MAGNETIC TAPE UNIT 2 READY
INITIATE REWIND OF MAGNETIC TAPE UNIT 2
DELAY UNTIL REWIND COMPLETED
SUBR
READ SEGMENT I FROM MAGNETIC TAPE UNIT 2, USING SEPARATE COPY OF MAGNETIC TAPE READ SUBROUTINE (MTRSRST)
POSITION UNIT 2 OVER PROGRAM SEGMENTS II AND III
TO CONTROL PROGRAM TO START EXECUTION

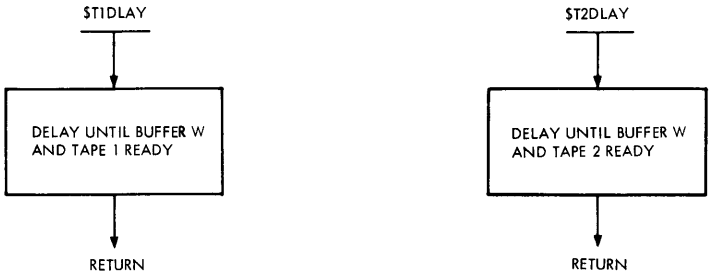
PART 1


$DSRP
SUBR
TRANSFER TO PART 2 FOR PROGRAM SETUP
CLEAR
CLEAR VECTOR REGION
READRC
• BUILD MN AND AB VECTORS FROM UN-SCHEDULED ACTIVITY FILE
• SET SIGN OF MN NEGATIVE FOR NON-CONTINUOUS ACTIVITIES
• SET NE TO NUMBER OF LAST MODE
RDDUR
SUBR
TO DUREAD FOR PROJECT DURATION
9 CARD
SUBR
TO END SUBR FOR END PRINT. STOP
SUBR
TO TVSR FOR TIME VECTOR
SUBR
TO DETAIL SR. FOR DETAIL SCHEDULE
SUBR
DETAIL SCHEDULE OUTPUT ROUTINE

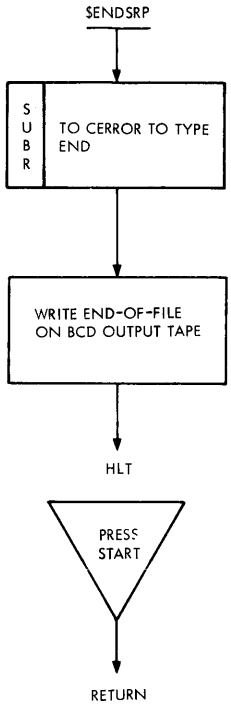
PART 2


$PART2
SUBR
TO MAGNETIC TAPE SETUP ROUTINE
SUBR
TO PARAMETER CONTROL ROUTINE FOR PARAMETER
READ
UPDATE
SORT
PRINT
IS DATA TYPE = 8?
no
ERROR 8
yes
SAVE RUN ID IN COMMON
SUBR
TO INPCON TO CONVERT 1ST DURATION
RETURN TO PART 1
COMPUTE VECTOR OFFSETS WITH TAG:
MNOFF
ABOFF
TOFF
SETUP TIME VECTOR LOAD LOCATION IN TIMESR
WILL PROBLEM FIT IN CORE?
no
ERROR 16
FITS
yes
IS START DATE GIVEN?
no
yes
SUBR
TO CALENDAR SETUP
BAD START MONTH
ERROR 12
INCORRECT UNIT TYPE
ERROR 13
CONVERT AND SET UP NA, MAJOR, SDAY, SMONTH, SYEAR, UNITS, NWHPD, FIRPAS = -1
COMPUTE NUMBER OF DAYS-OFF/WEEK AND BUILD DAY-OFF VECTOR
NWDPW = 7 - NDYOFF
FIND MAX LOCATION IN COMPUTER

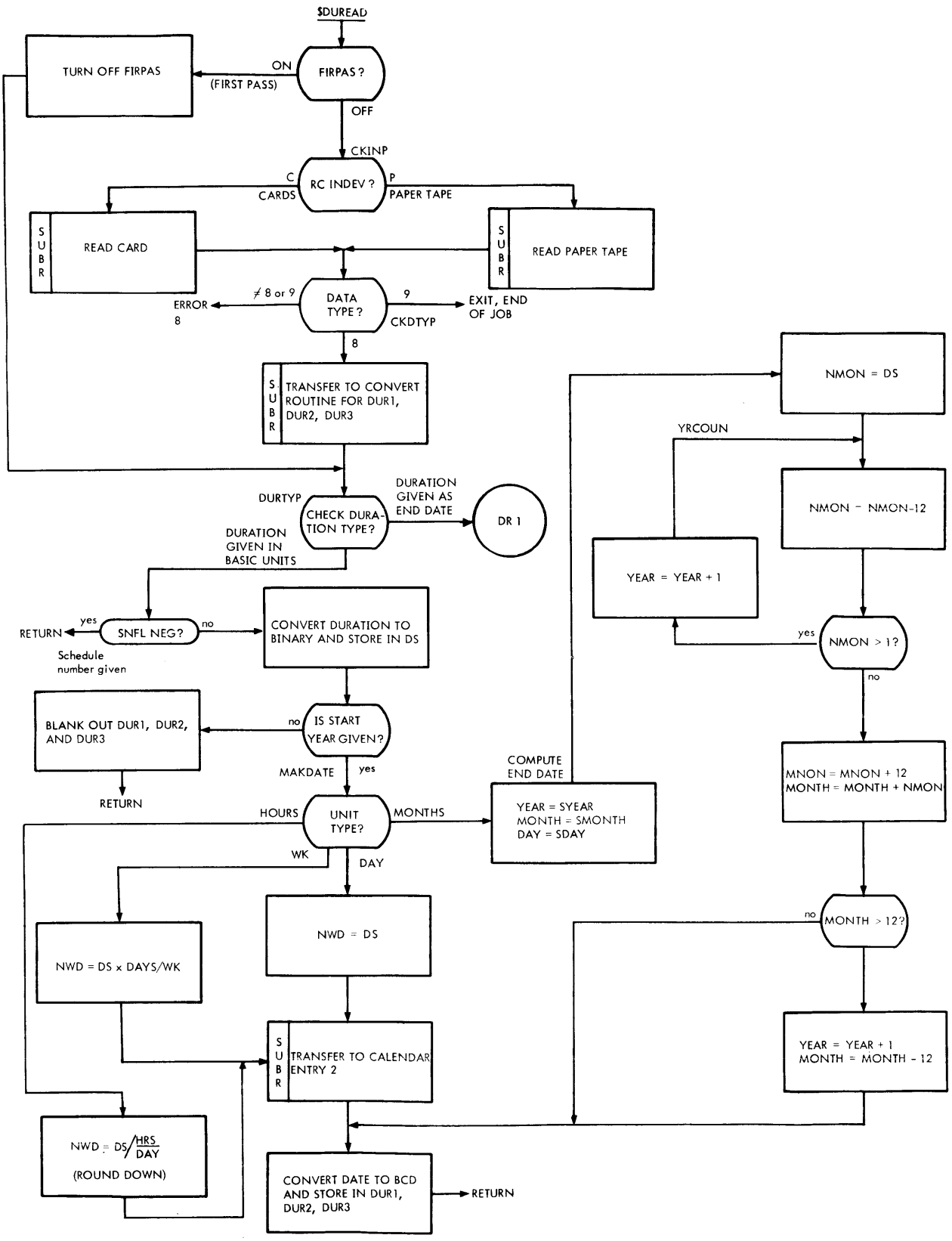
DELAY ROUTINE



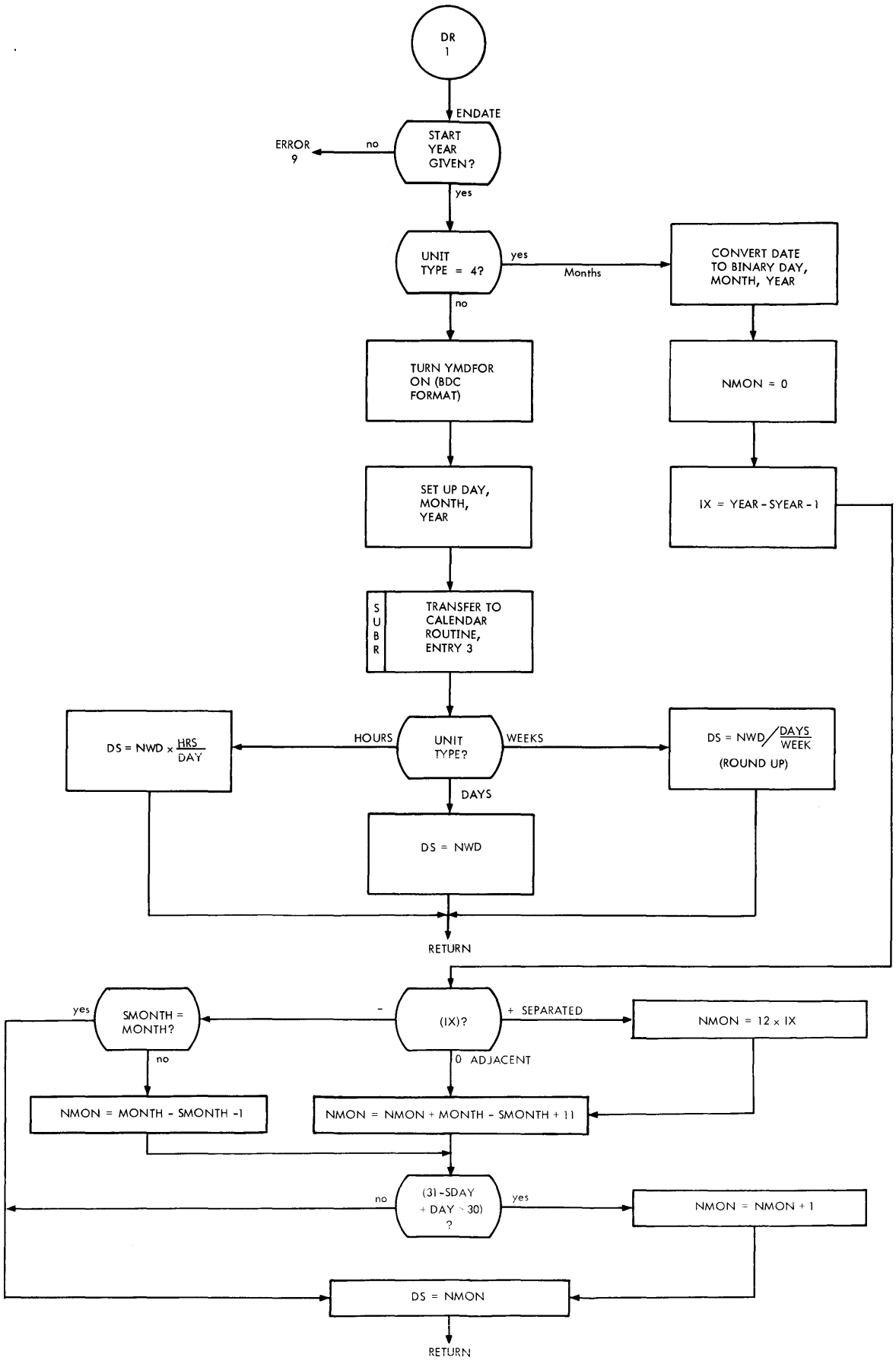
END ROUTINE



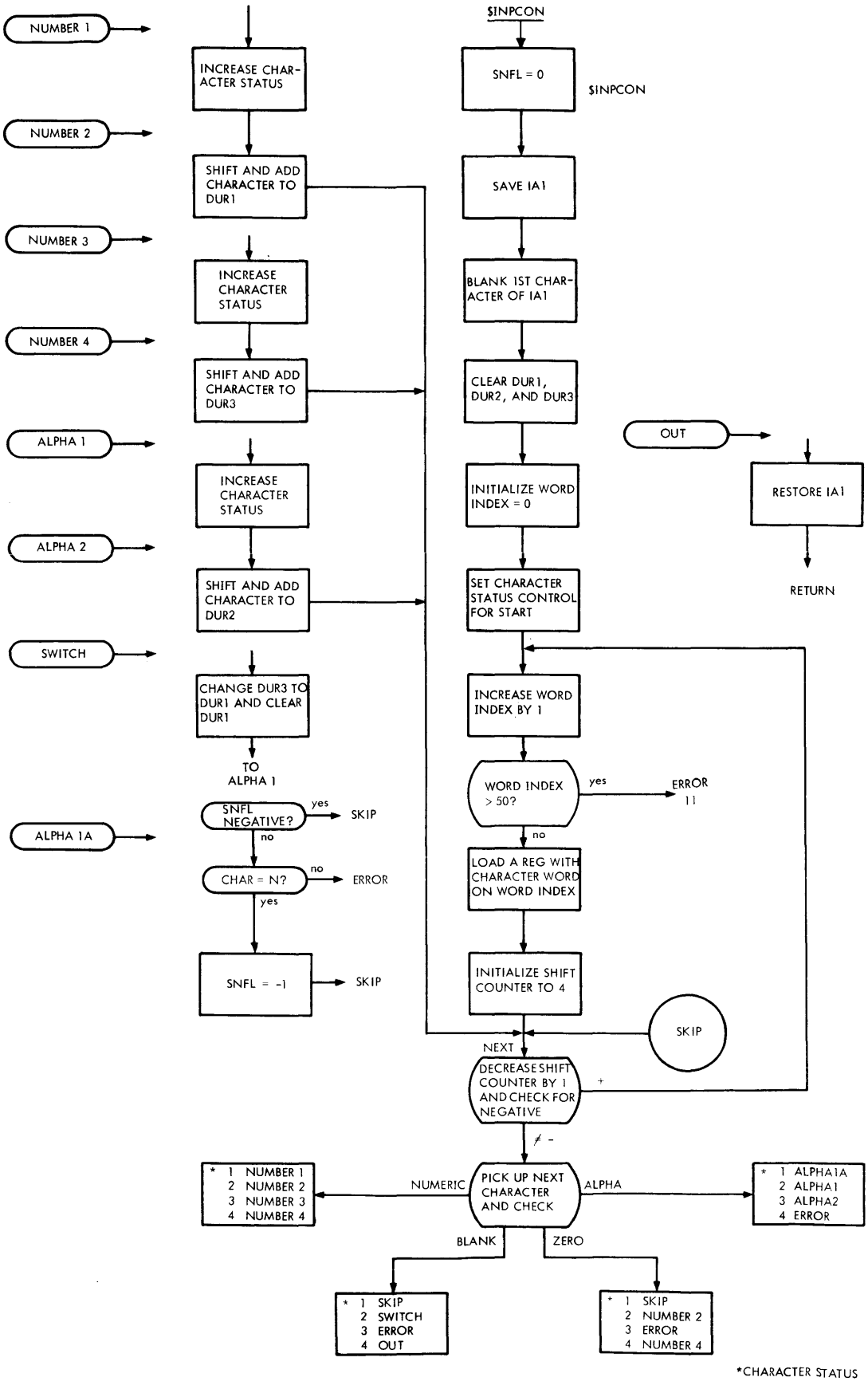
DURATION READ ROUTINE



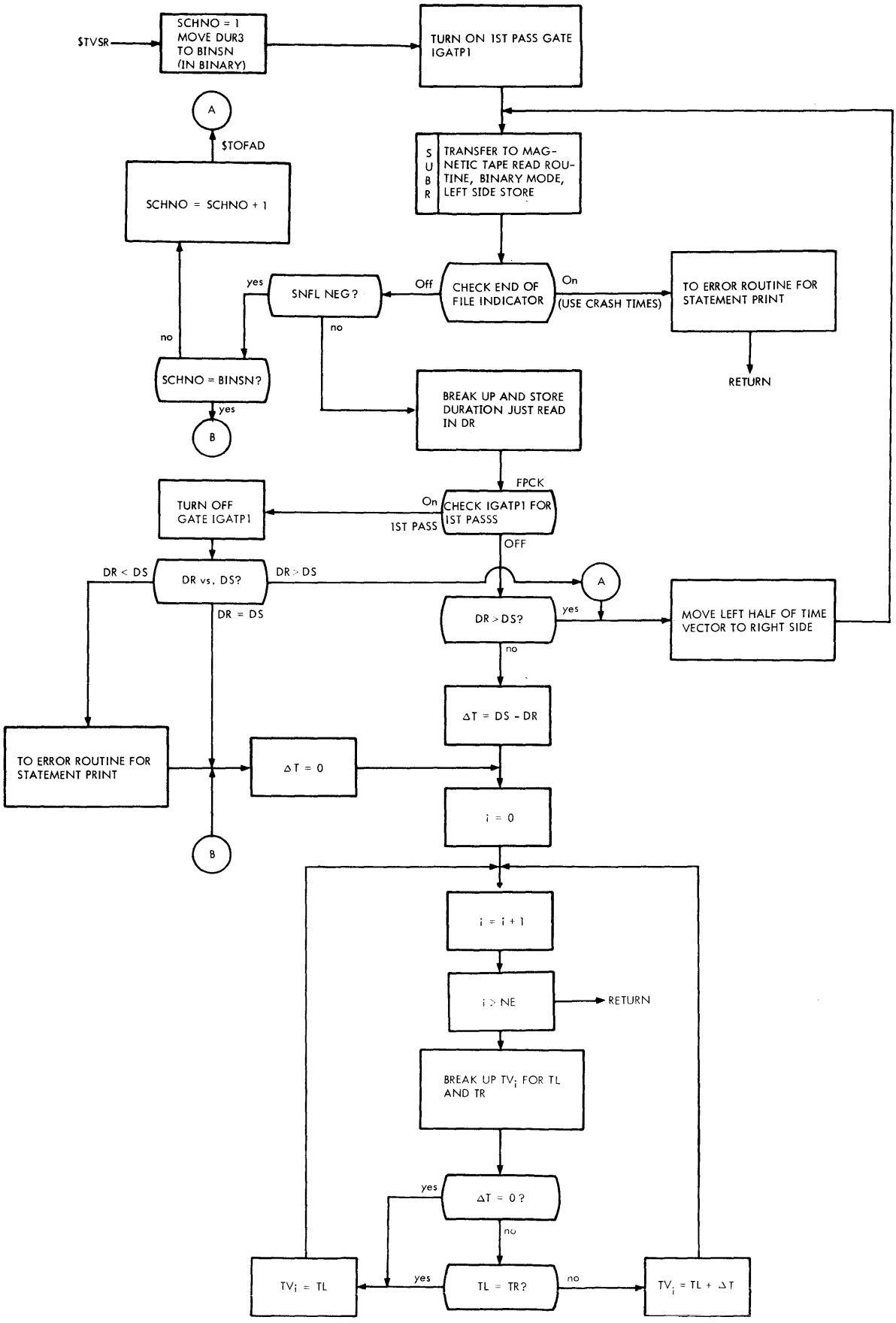
DURATION READ ROUTINE (cont.)

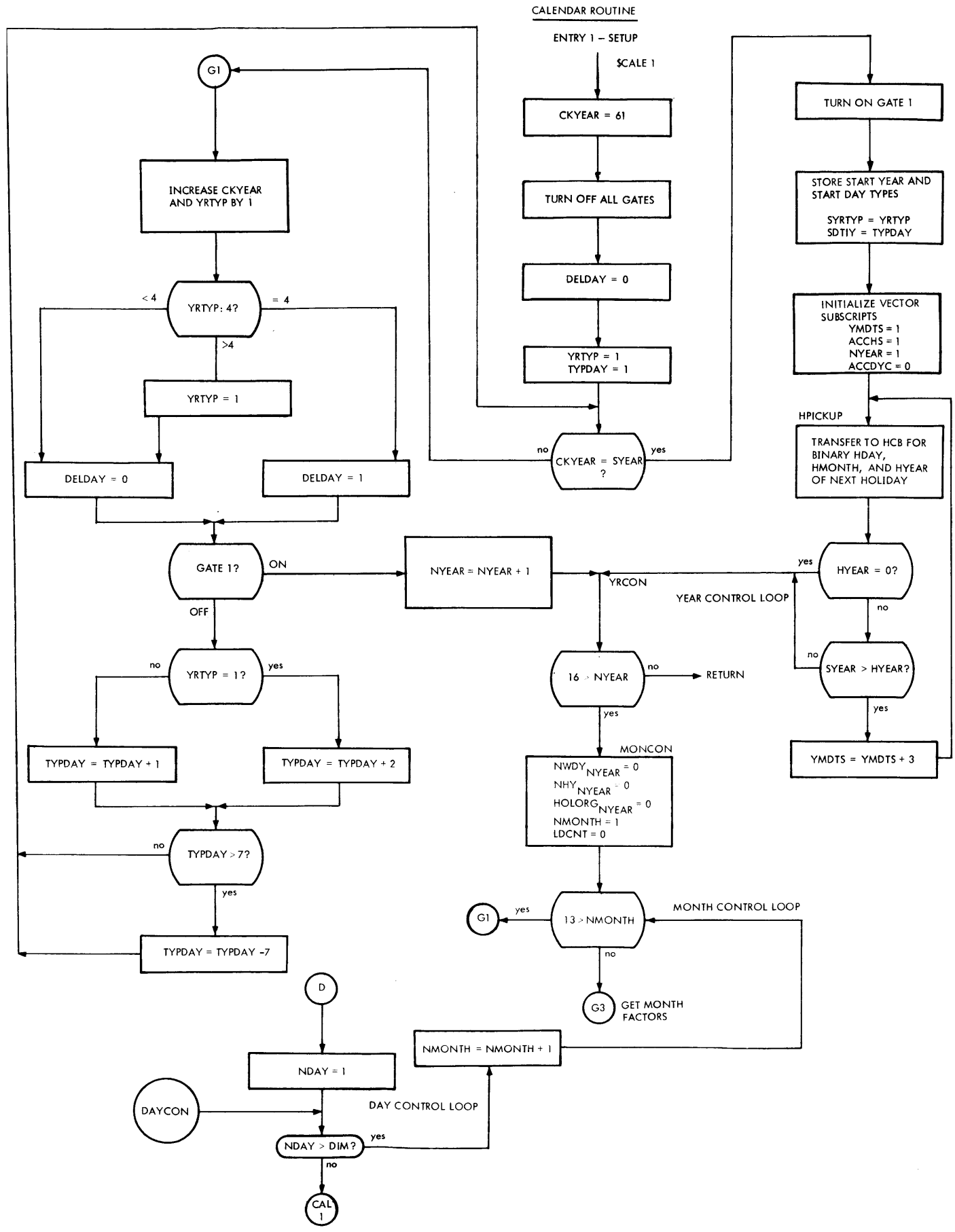


INPUT CONVERSION ROUTINE

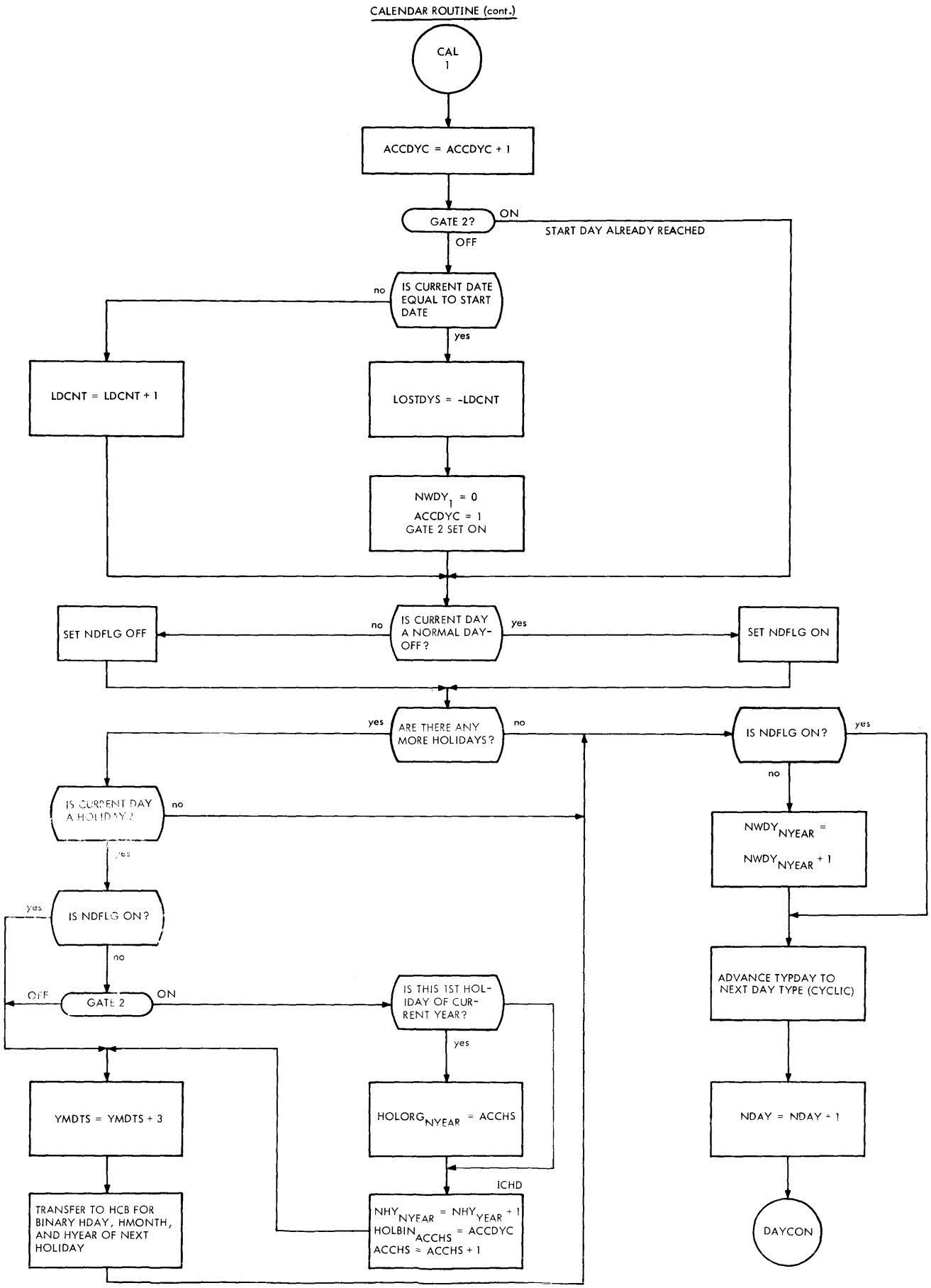


TIME VECTOR ROUTINE

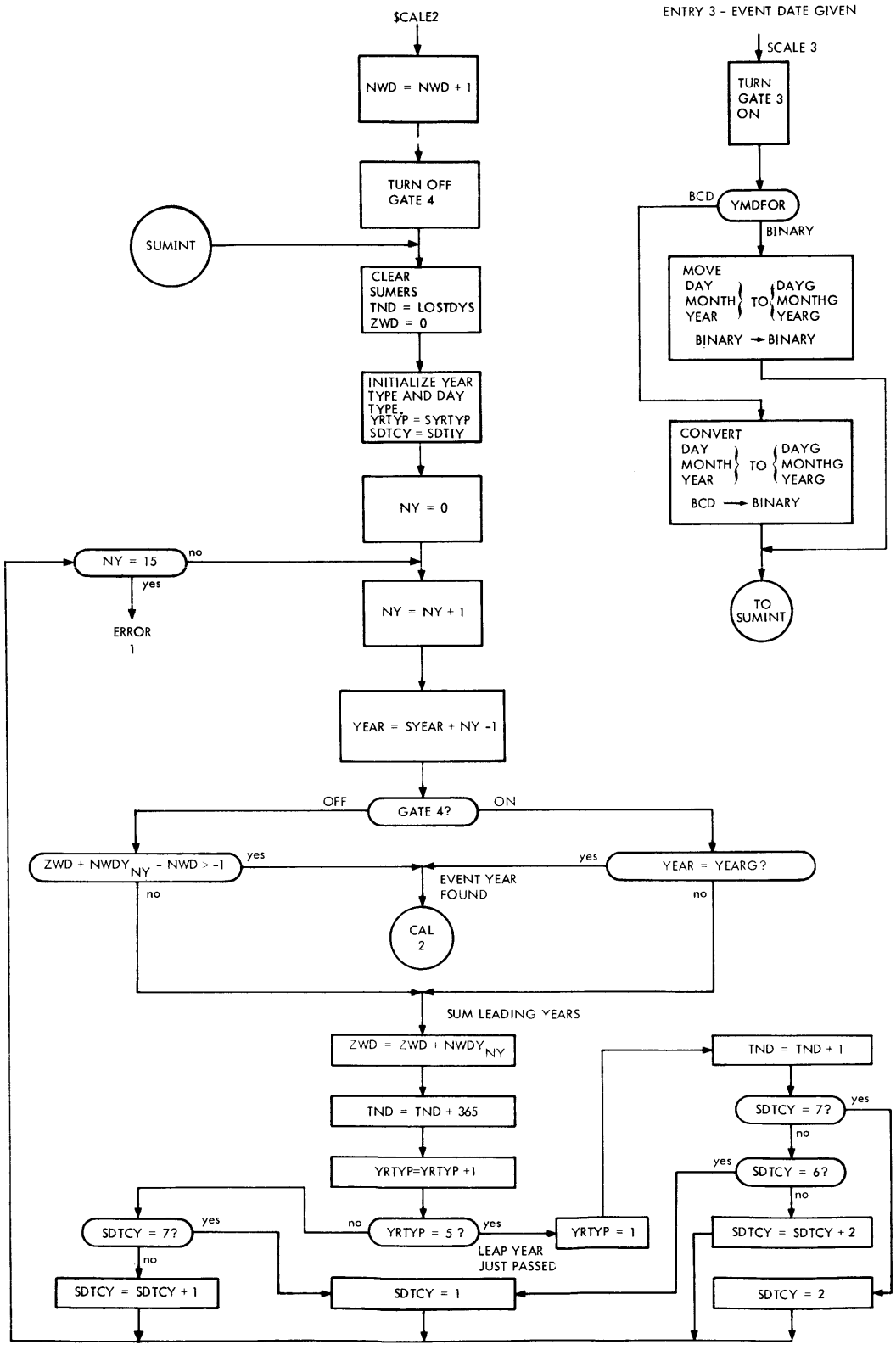


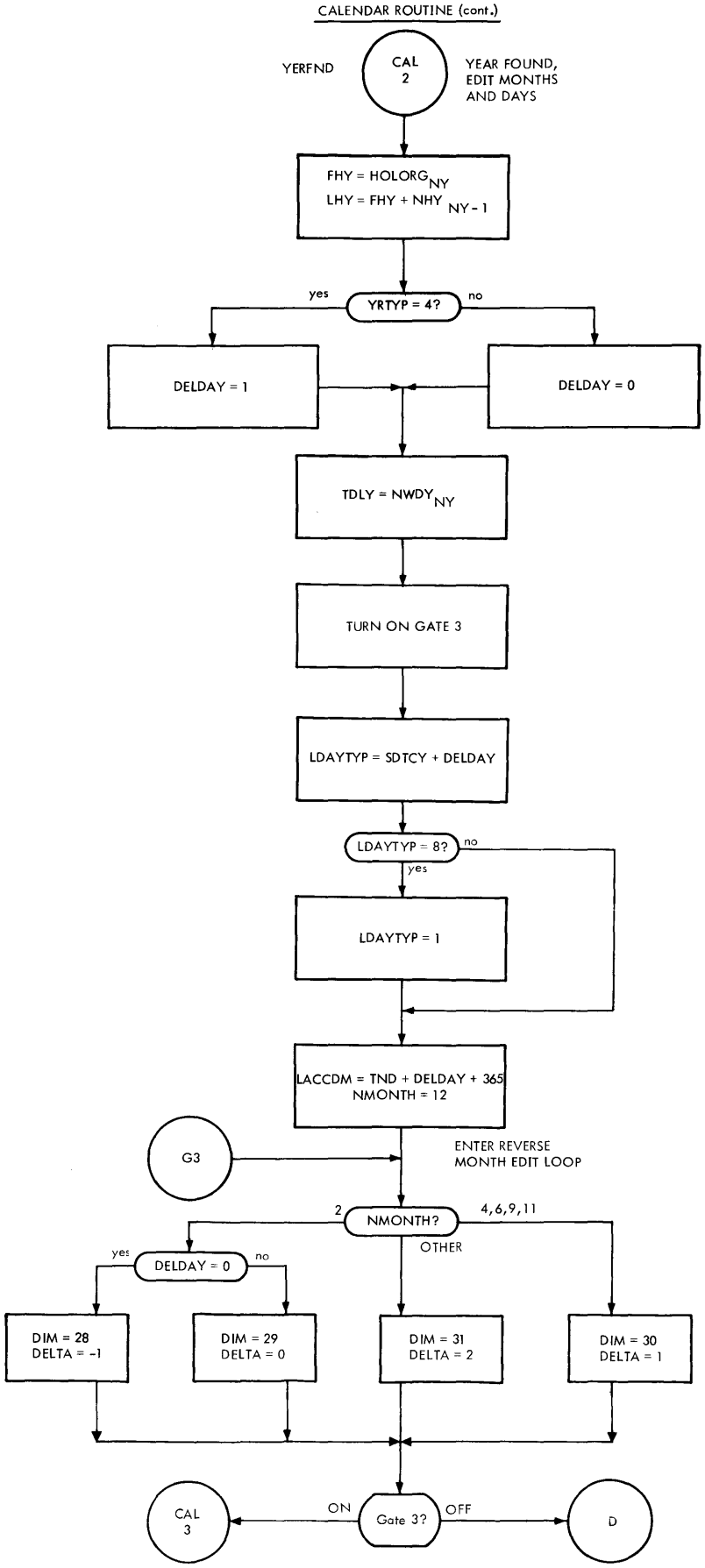

CALENDAR ROUTINE
ENTRY 1 – SETUP
$CALE 1
CKYEAR = 61
TURN OFF ALL GATES
DELDAY = 0
YRTYP = 1
TYPDAY = 1
CKYEAR = SYEAR ?
no
yes
TURN ON GATE 1
STORE START YEAR AND START DAY TYPES
SYRTYP = YRTYP
SDTIY = TYPDAY
INITIALIZE VECTOR SUBSCRIPTS
YMDTS = 1
ACCHS = 1
NYEAR = 1
ACCDYC = 0
HPICKUP
TRANSFER TO HCB FOR BINARY HDAY, HMONTH, AND HYEAR OF NEXT HOLIDAY
HYEAR = 0?
yes
no
SYEAR > HYEAR?
no
yes
YMDTS = YMDTS + 3
YEAR CONTROL LOOP
YRCON
16 > NYEAR
no
RETURN
yes
MONCON
NWDY NYEAR = 0
NHY NYEAR = 0
HOLORG NYEAR = 0
NMONTH = 1
LDCNT = 0
G1
yes
13 > NMONTH
MONTH CONTROL LOOP
no
G3
GET MONTH FACTORS
NMONTH = NMONTH + 1
G1
INCREASE CKYEAR AND YRTYP BY 1
YRTYP: 4?
< 4
= 4
>4
YRTYP = 1
DELDAY = 0
DELDAY = 1
GATE 1?
ON
OFF
NYEAR = NYEAR + 1
YRTYP = 1?
no
yes
TYPDAY = TYPDAY + 1
TYPDAY = TYPDAY + 2
TYPDAY > 7?
no
yes
TYPDAY = TYPDAY -7
D
NDAY = 1
DAYCON
DAY CONTROL LOOP
NDAY > DIM?
yes
no
CAL 1

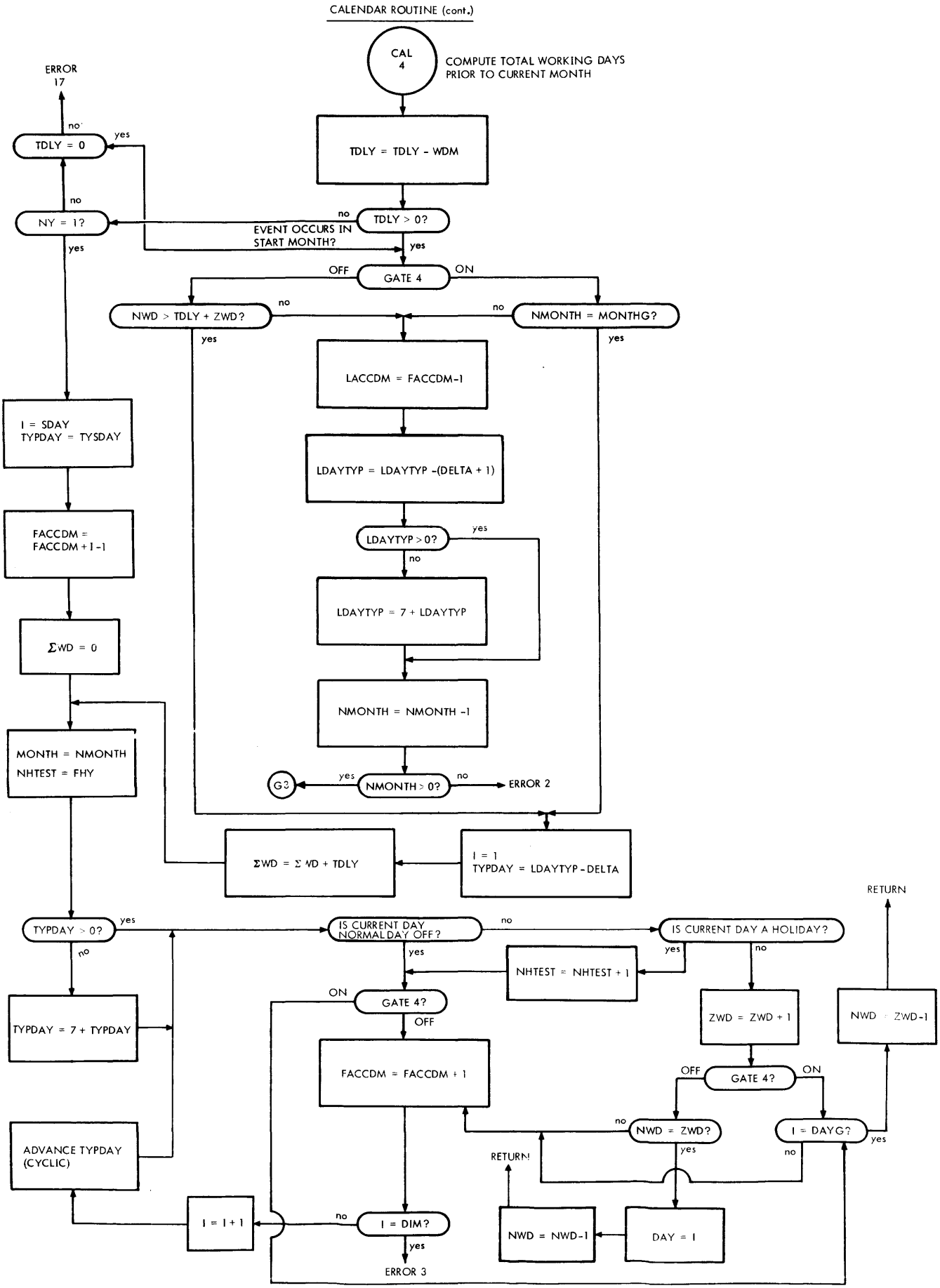
CALENDAR ROUTINE (cont.)



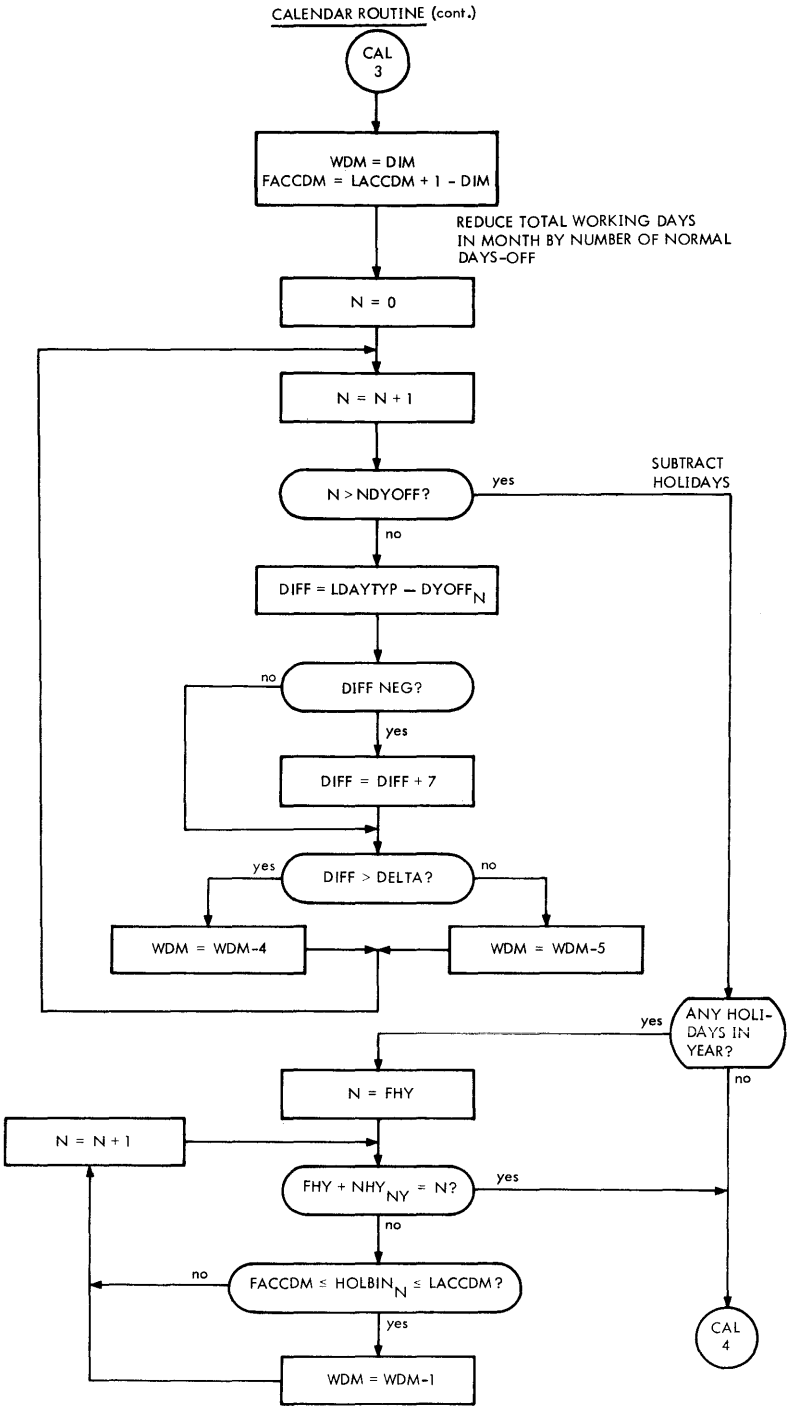
ENTRY 2 – WORKING DAYS GIVEN



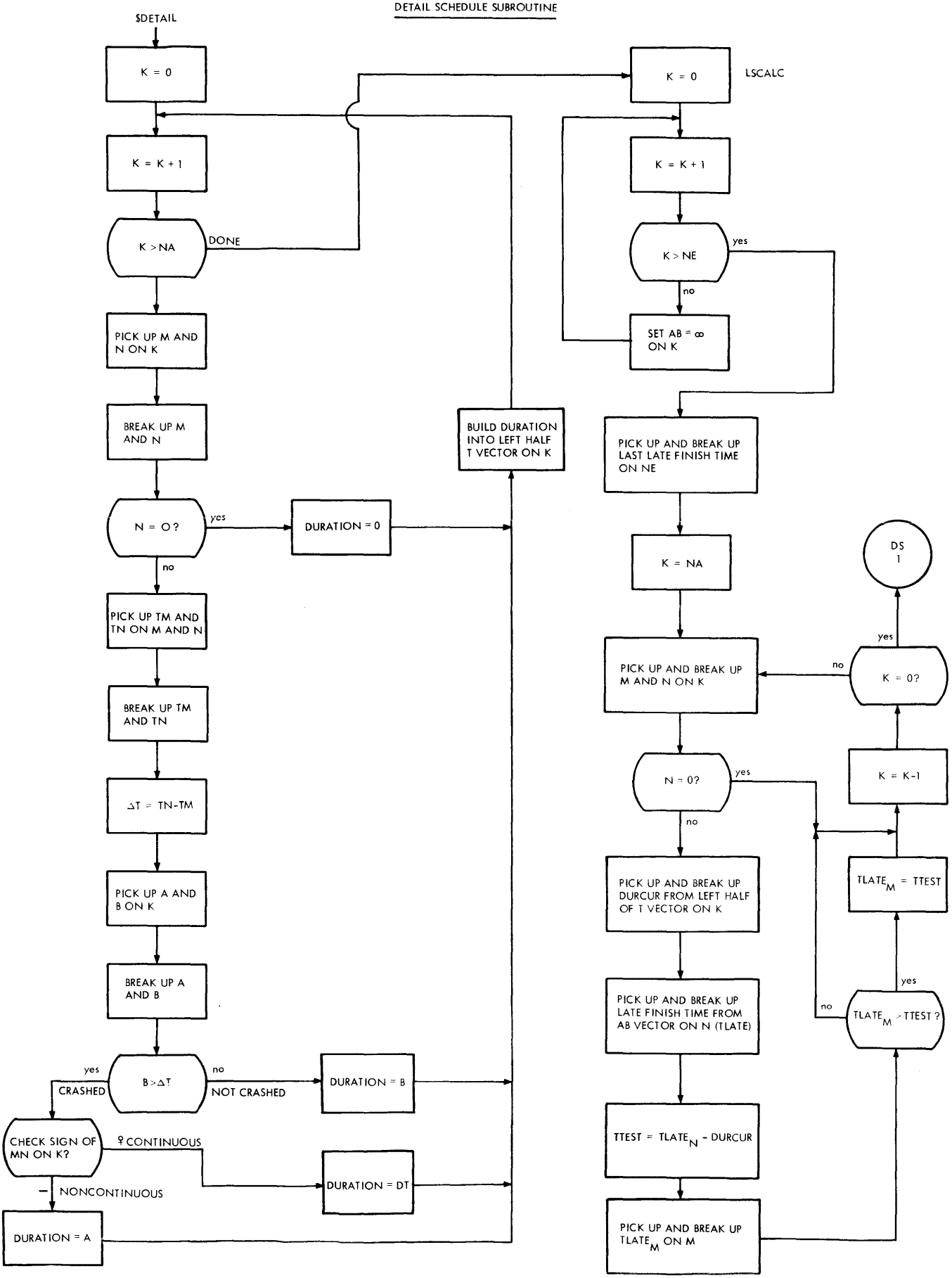

YERFND
CAL 2
YEAR FOUND, EDIT MONTHS AND DAYS
FHY = HOLORG NY
LHY = FHY + NHY NY - 1
yes
YRTYP = 4?
no
DELDAY = 1
DELDAY = 0
TDLY = NWDY NY
TURN ON GATE 3
LDAYTYP = SDTCY + DELDAY
LDAYTYP = 8?
no
yes
LDAYTYP = 1
LACCDM = TND + DELDAY + 365
NMONTH = 12
G3
ENTER REVERSE MONTH EDIT LOOP
2
NMONTH?
4,6,9,11
OTHER
yes
DELDAY = 0
no
DIM = 28
DELTA = -1
DIM = 29
DELTA = 0
DIM = 31
DELTA = 2
DIM = 30
DELTA = 1
CAL 3
ON
Gate 3?
OFF
D

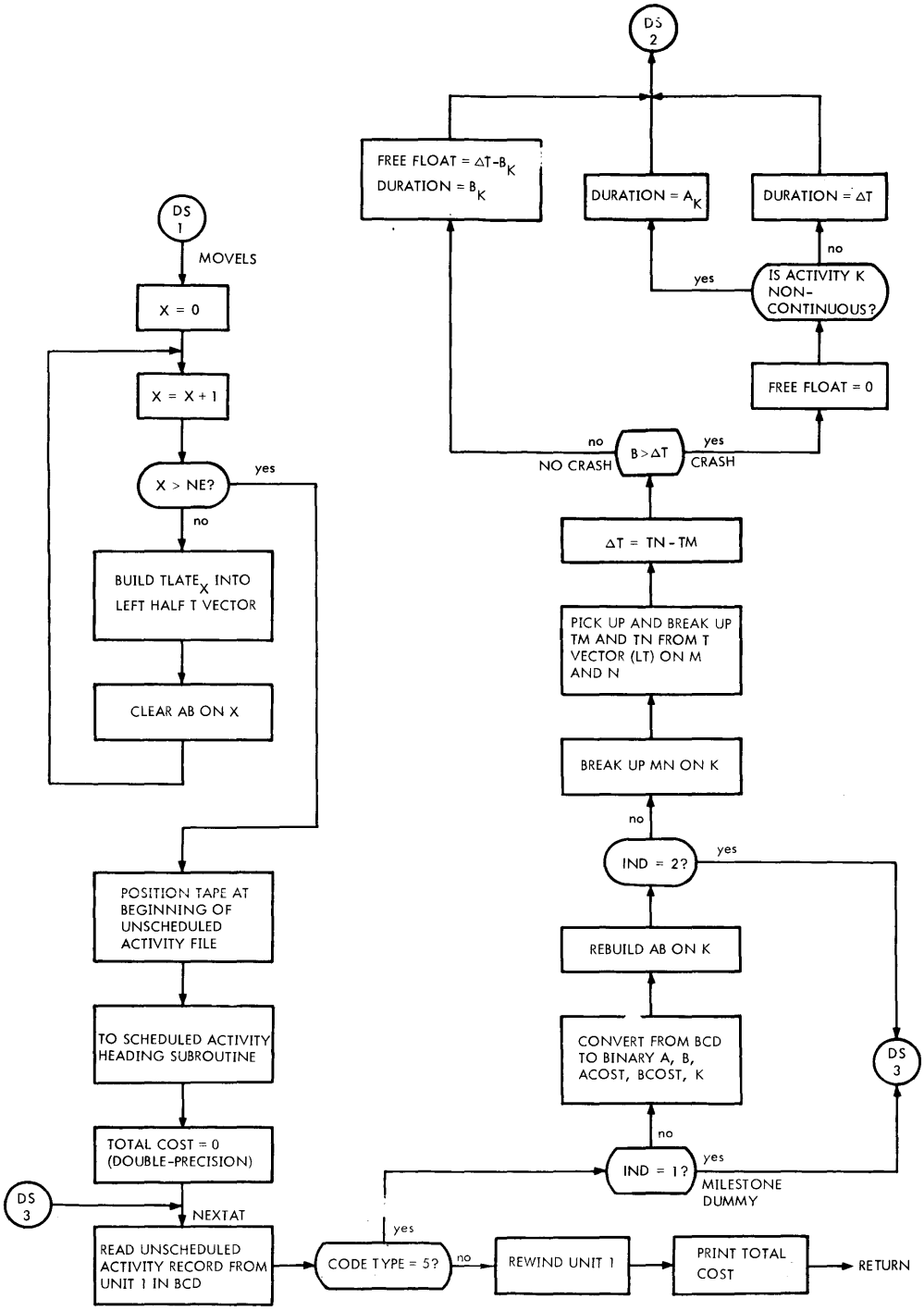

CALENDAR ROUTINE (cont.)
CAL 4
COMPUTE TOTAL WORKING DAYS PRIOR TO CURRENT MONTH
TDLY = TDLY - WDM
TDLY > 0?
no
yes
EVENT OCCURS IN START MONTH?
GATE 4
OFF
ON
NWD > TDLY + ZWD?
NMONTH = MONTHG?
LACCDM = FACCDM-1
LDAYTYP = LDAYTYP -(DELTA + 1)
LDAYTYP > 0?
LDAYTYP = 7 + LDAYTYP
NMONTH = NMONTH -1
NMONTH > 0?
ERROR 2
G3
I = 1
TYPDAY = LDAYTYP - DELTA
ΣWD = Σ WD + TDLY
NY = 1?
ERROR 17
TDLY = 0
I = SDAY
TYPDAY = TYSDAY
FACCDM = FACCDM + I - 1
ΣWD = 0
MONTH = NMONTH
NHTEST = FHY
TYPDAY > 0?
TYPDAY = 7 + TYPDAY
ADVANCE TYPDAY (CYCLIC)
IS CURRENT DAY NORMAL DAY OFF?
IS CURRENT DAY A HOLIDAY?
NHTEST = NHTEST + 1
GATE 4?
FACCDM = FACCDM + 1
I = I + 1
I = DIM?
ERROR 3
ZWD = ZWD + 1
NWD = ZWD?
I = DAYG?
NWD = NWD-1
DAY = 1
NWD = ZWD-1
RETURN
RETURN

CALENDAR ROUTINE (cont.)

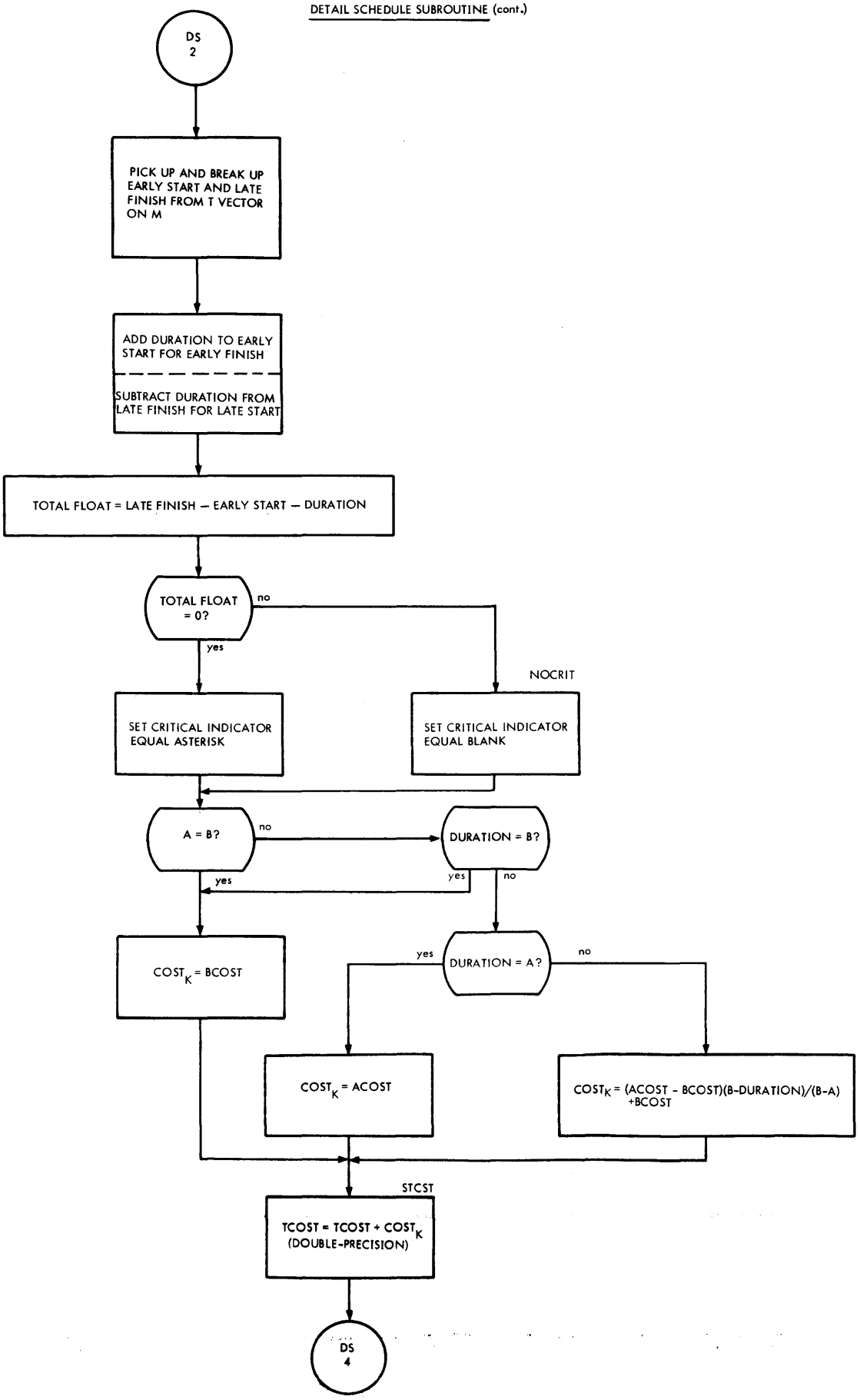


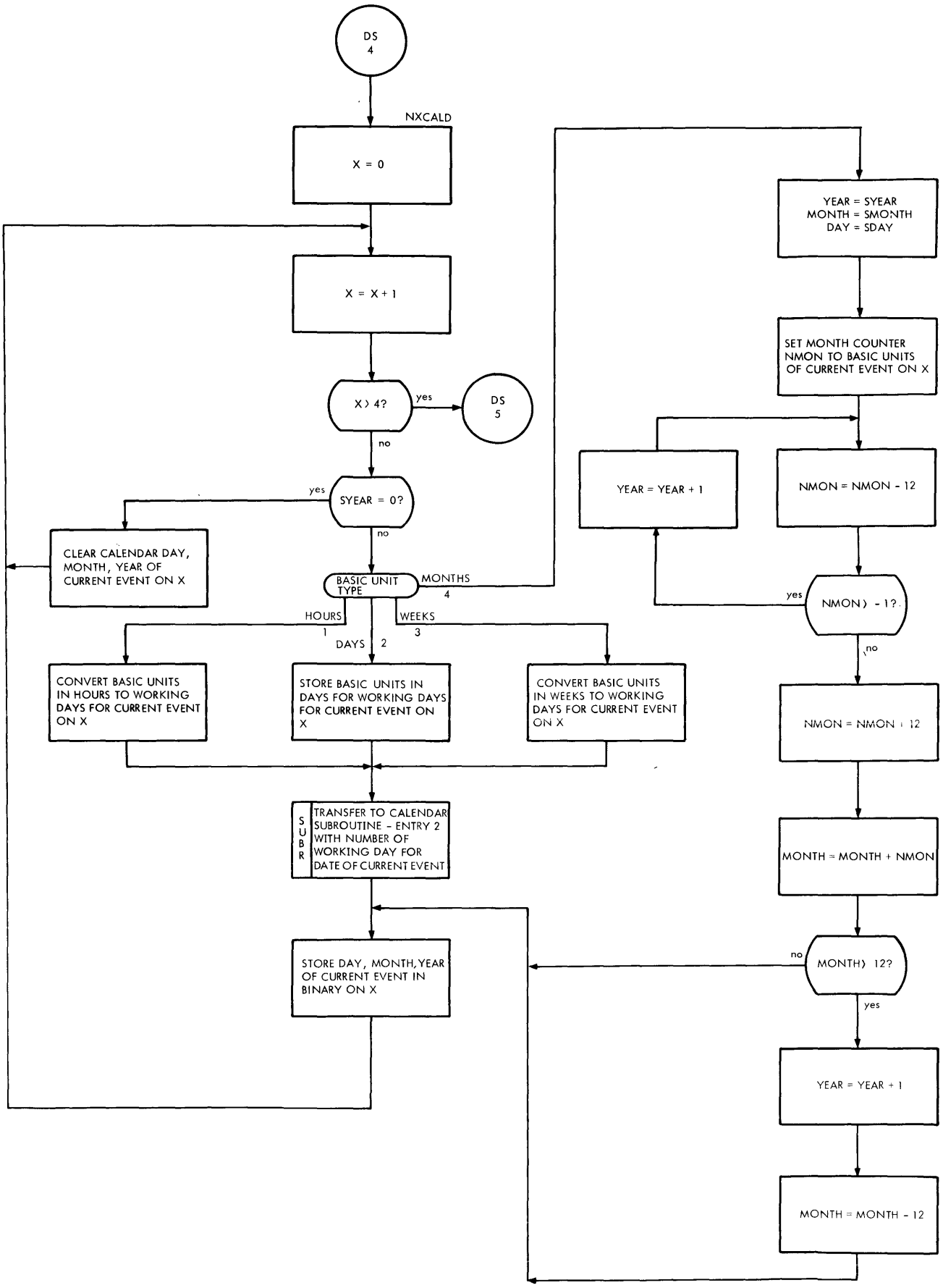
DETAIL SCHEDULE SUBROUTINE



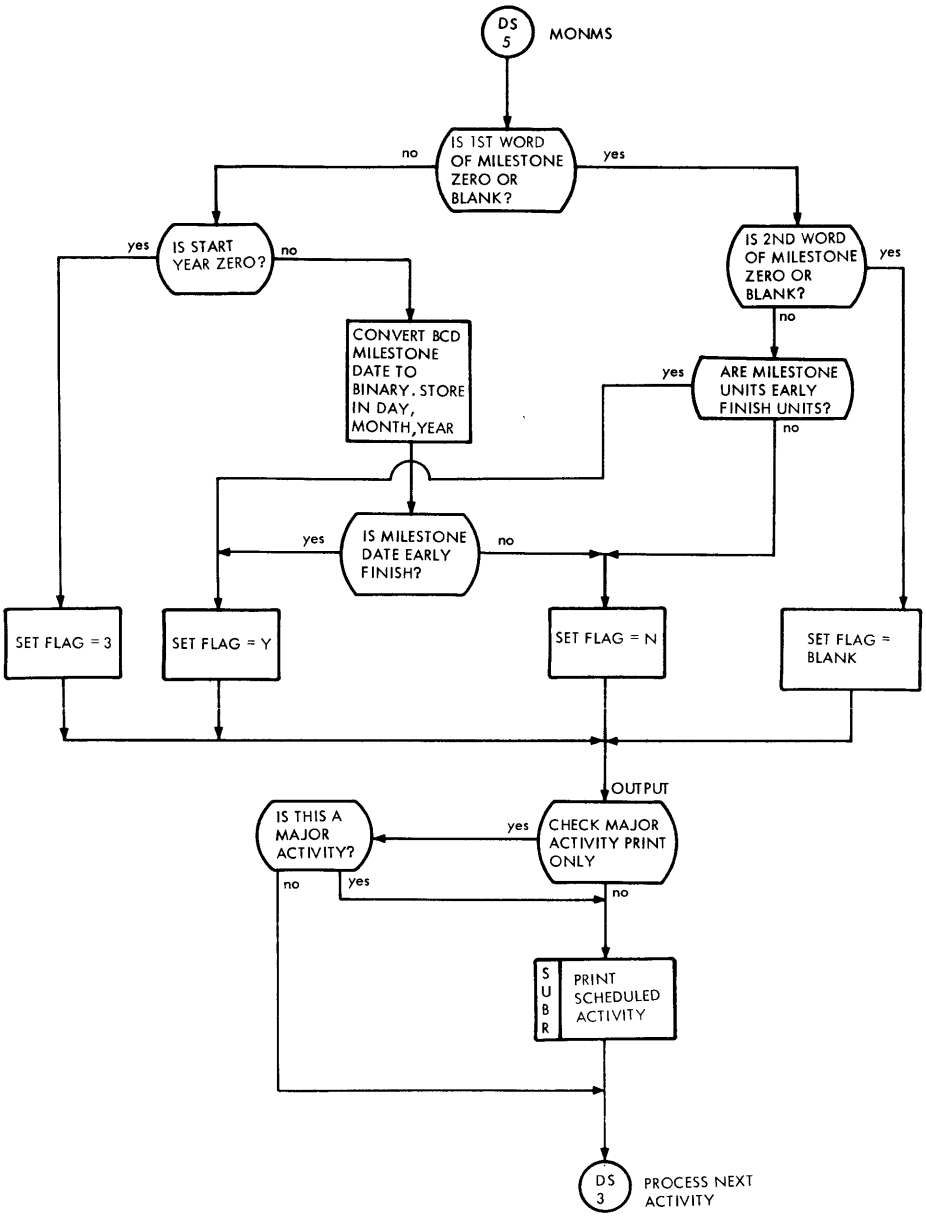

DS 1
MOVELS
X = 0
X = X + 1
X > NE?
yes
no
BUILD TLATE$_X$ INTO LEFT HALF T VECTOR
CLEAR AB ON X
POSITION TAPE AT BEGINNING OF UNSCHEDULED ACTIVITY FILE
TO SCHEDULED ACTIVITY HEADING SUBROUTINE
TOTAL COST = 0 (DOUBLE-PRECISION)
DS 3
NEXTAT
READ UNSCHEDULED ACTIVITY RECORD FROM UNIT 1 IN BCD
CODE TYPE = 5?
yes
no
REWIND UNIT 1
PRINT TOTAL COST
RETURN
IND = 1?
yes
MILESTONE DUMMY
no
CONVERT FROM BCD TO BINARY A, B, ACOST, BCOST, K
DS 3
REBUILD AB ON K
IND = 2?
yes
no
BREAK UP MN ON K
PICK UP AND BREAK UP TM AND TN FROM T VECTOR (LT) ON M AND N
ΔT = TN - TM
B>ΔT
no
NO CRASH
yes
CRASH
FREE FLOAT = ΔT-B$_K$
DURATION = B$_K$
DS 2
DURATION = A$_K$
DURATION = ΔT
yes
IS ACTIVITY K NON-CONTINUOUS?
no
FREE FLOAT = 0

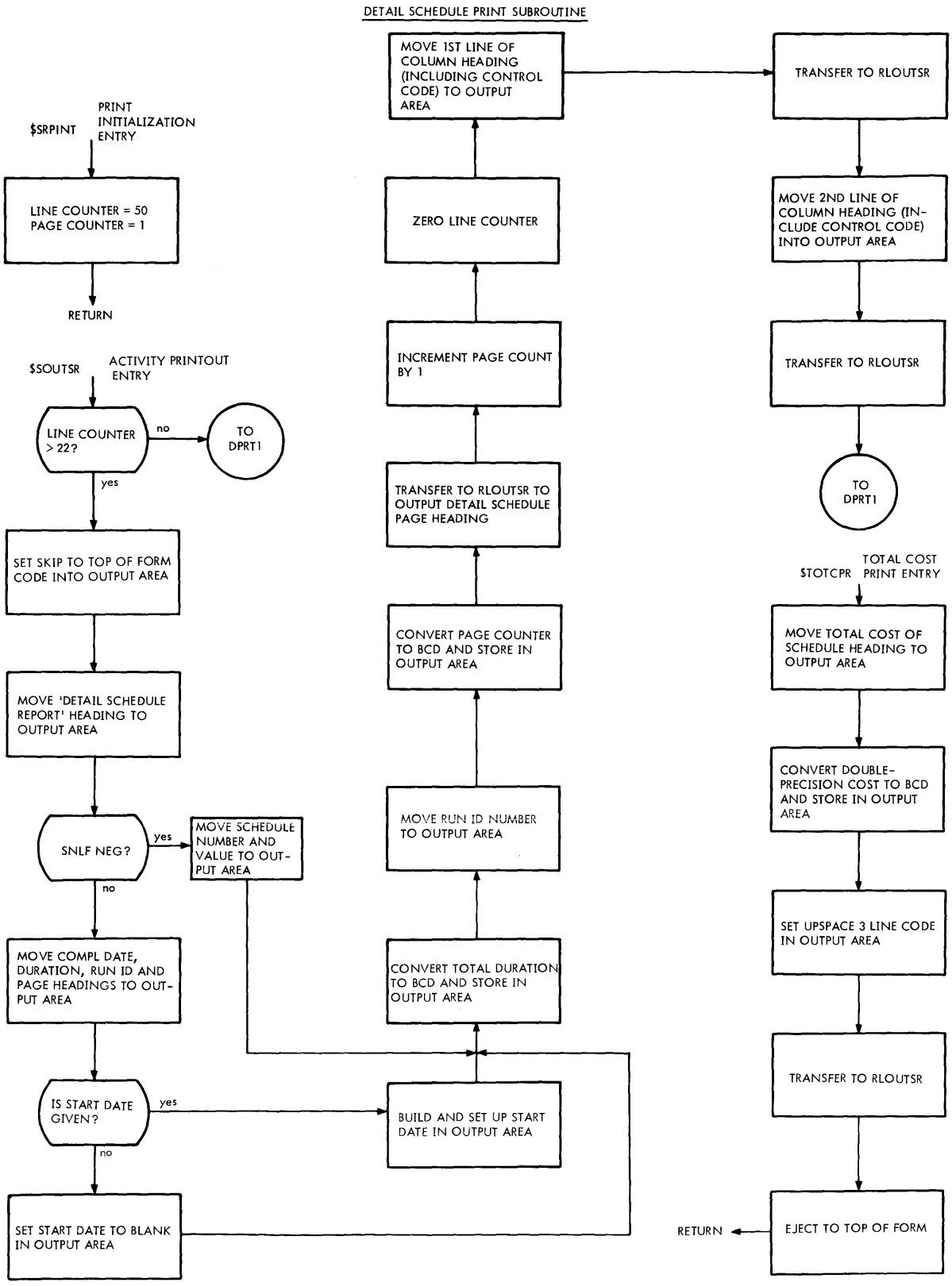
DETAIL SCHEDULE SUBROUTINE (cont.)

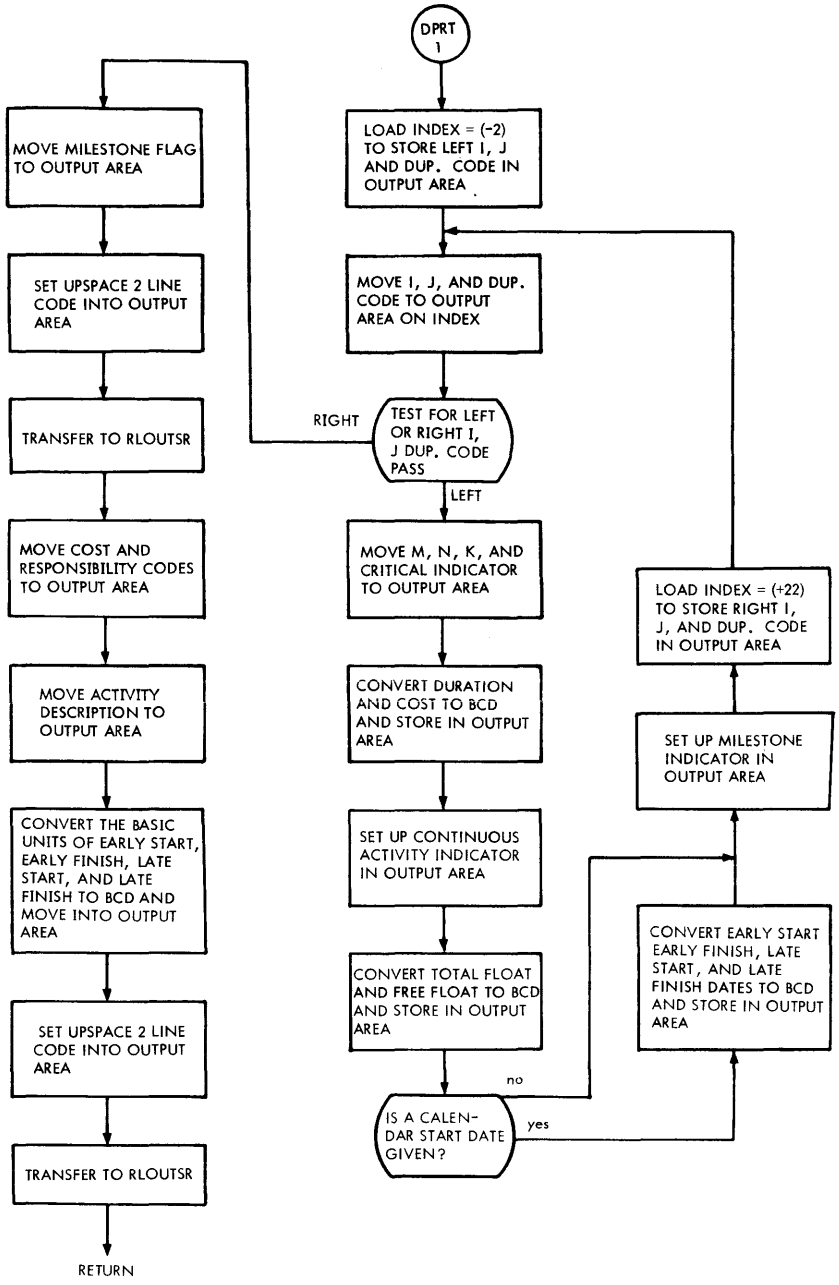



DS 4
NXCALD
X = 0
X = X + 1
X > 4?
yes
DS 5
no
yes
SYEAR = 0?
no
CLEAR CALENDAR DAY, MONTH, YEAR OF CURRENT EVENT ON X
BASIC UNIT TYPE
MONTHS 4
HOURS 1
DAYS 2
WEEKS 3
CONVERT BASIC UNITS IN HOURS TO WORKING DAYS FOR CURRENT EVENT ON X
STORE BASIC UNITS IN DAYS FOR WORKING DAYS FOR CURRENT EVENT ON X
CONVERT BASIC UNITS IN WEEKS TO WORKING DAYS FOR CURRENT EVENT ON X
SUBR
TRANSFER TO CALENDAR SUBROUTINE - ENTRY 2 WITH NUMBER OF WORKING DAY FOR DATE OF CURRENT EVENT
STORE DAY, MONTH, YEAR OF CURRENT EVENT IN BINARY ON X
YEAR = SYEAR
MONTH = SMONTH
DAY = SDAY
SET MONTH COUNTER NMON TO BASIC UNITS OF CURRENT EVENT ON X
YEAR = YEAR + 1
NMON = NMON - 12
yes
NMON > 12?
no
NMON = NMON + 12
MONTH = MONTH + NMON
no
MONTH > 12?
yes
YEAR = YEAR + 1
MONTH = MONTH - 12

DETAIL SCHEDULE SUBROUTINE (cont.)

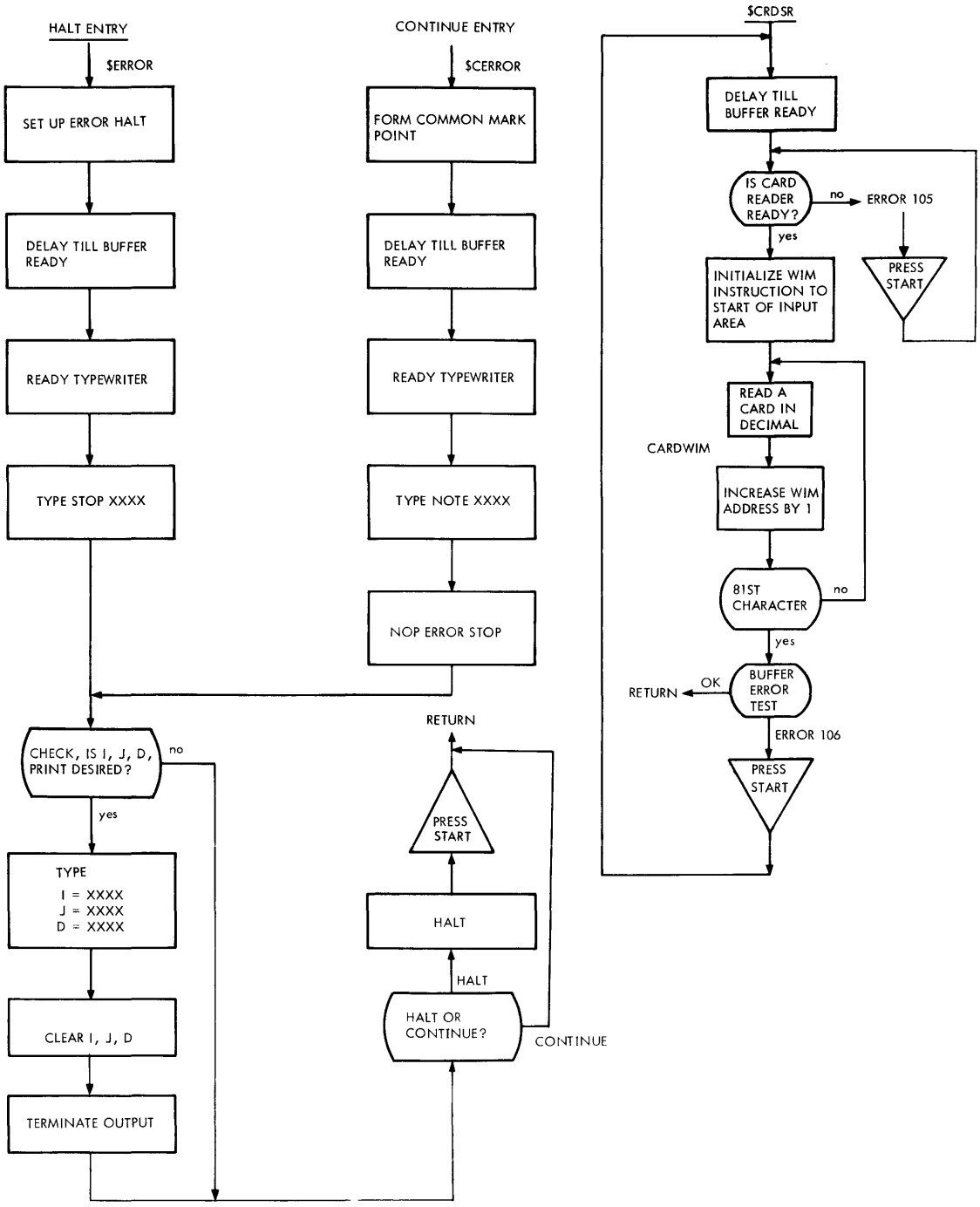



DETAIL SCHEDULE PRINT SUBROUTINE
$SRPINT
PRINT INITIALIZATION ENTRY
LINE COUNTER = 50
PAGE COUNTER = 1
RETURN
$SOUTSR
ACTIVITY PRINTOUT ENTRY
LINE COUNTER > 22?
no
TO DPRT1
yes
SET SKIP TO TOP OF FORM CODE INTO OUTPUT AREA
MOVE 'DETAIL SCHEDULE REPORT' HEADING TO OUTPUT AREA
SNLF NEG?
yes
MOVE SCHEDULE NUMBER AND VALUE TO OUTPUT AREA
no
MOVE COMPL DATE, DURATION, RUN ID AND PAGE HEADINGS TO OUTPUT AREA
IS START DATE GIVEN?
yes
no
SET START DATE TO BLANK IN OUTPUT AREA
BUILD AND SET UP START DATE IN OUTPUT AREA
CONVERT TOTAL DURATION TO BCD AND STORE IN OUTPUT AREA
MOVE RUN ID NUMBER TO OUTPUT AREA
CONVERT PAGE COUNTER TO BCD AND STORE IN OUTPUT AREA
TRANSFER TO RLOUTSR TO OUTPUT DETAIL SCHEDULE PAGE HEADING
INCREMENT PAGE COUNT BY 1
ZERO LINE COUNTER
MOVE 1ST LINE OF COLUMN HEADING (INCLUDING CONTROL CODE) TO OUTPUT AREA
TRANSFER TO RLOUTSR
MOVE 2ND LINE OF COLUMN HEADING (INCLUDE CONTROL CODE) INTO OUTPUT AREA
TRANSFER TO RLOUTSR
TO DPRT1
$TOTCPR
TOTAL COST PRINT ENTRY
MOVE TOTAL COST OF SCHEDULE HEADING TO OUTPUT AREA
CONVERT DOUBLE-PRECISION COST TO BCD AND STORE IN OUTPUT AREA
SET UPSPACE 3 LINE CODE IN OUTPUT AREA
TRANSFER TO RLOUTSR
EJECT TO TOP OF FORM
RETURN

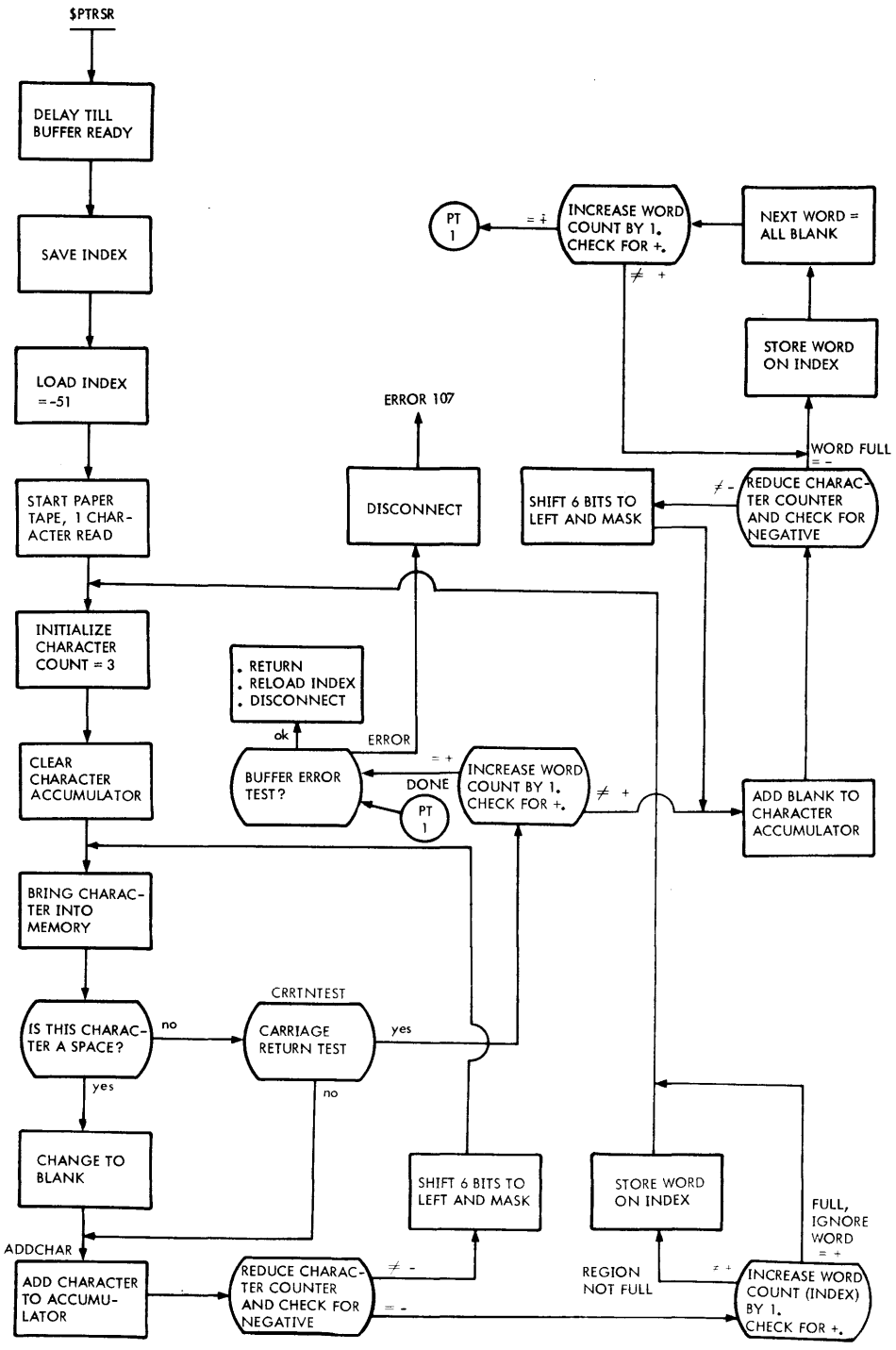

DPRT 1
LOAD INDEX = (-2) TO STORE LEFT I, J AND DUP. CODE IN OUTPUT AREA
MOVE I, J, AND DUP. CODE TO OUTPUT AREA ON INDEX
TEST FOR LEFT OR RIGHT I, J DUP. CODE PASS
RIGHT
LEFT
MOVE M, N, K, AND CRITICAL INDICATOR TO OUTPUT AREA
CONVERT DURATION AND COST TO BCD AND STORE IN OUTPUT AREA
SET UP CONTINUOUS ACTIVITY INDICATOR IN OUTPUT AREA
CONVERT TOTAL FLOAT AND FREE FLOAT TO BCD AND STORE IN OUTPUT AREA
IS A CALENDAR START DATE GIVEN?
no
yes
CONVERT EARLY START EARLY FINISH, LATE START, AND LATE FINISH DATES TO BCD AND STORE IN OUTPUT AREA
SET UP MILESTONE INDICATOR IN OUTPUT AREA
LOAD INDEX = (+22) TO STORE RIGHT I, J, AND DUP. CODE IN OUTPUT AREA
MOVE MILESTONE FLAG TO OUTPUT AREA
SET UPSPACE 2 LINE CODE INTO OUTPUT AREA
TRANSFER TO RLOUTSR
MOVE COST AND RESPONSIBILITY CODES TO OUTPUT AREA
MOVE ACTIVITY DESCRIPTION TO OUTPUT AREA
CONVERT THE BASIC UNITS OF EARLY START, EARLY FINISH, LATE START, AND LATE FINISH TO BCD AND MOVE INTO OUTPUT AREA
SET UPSPACE 2 LINE CODE INTO OUTPUT AREA
TRANSFER TO RLOUTSR
RETURN

ERROR SUBROUTINE

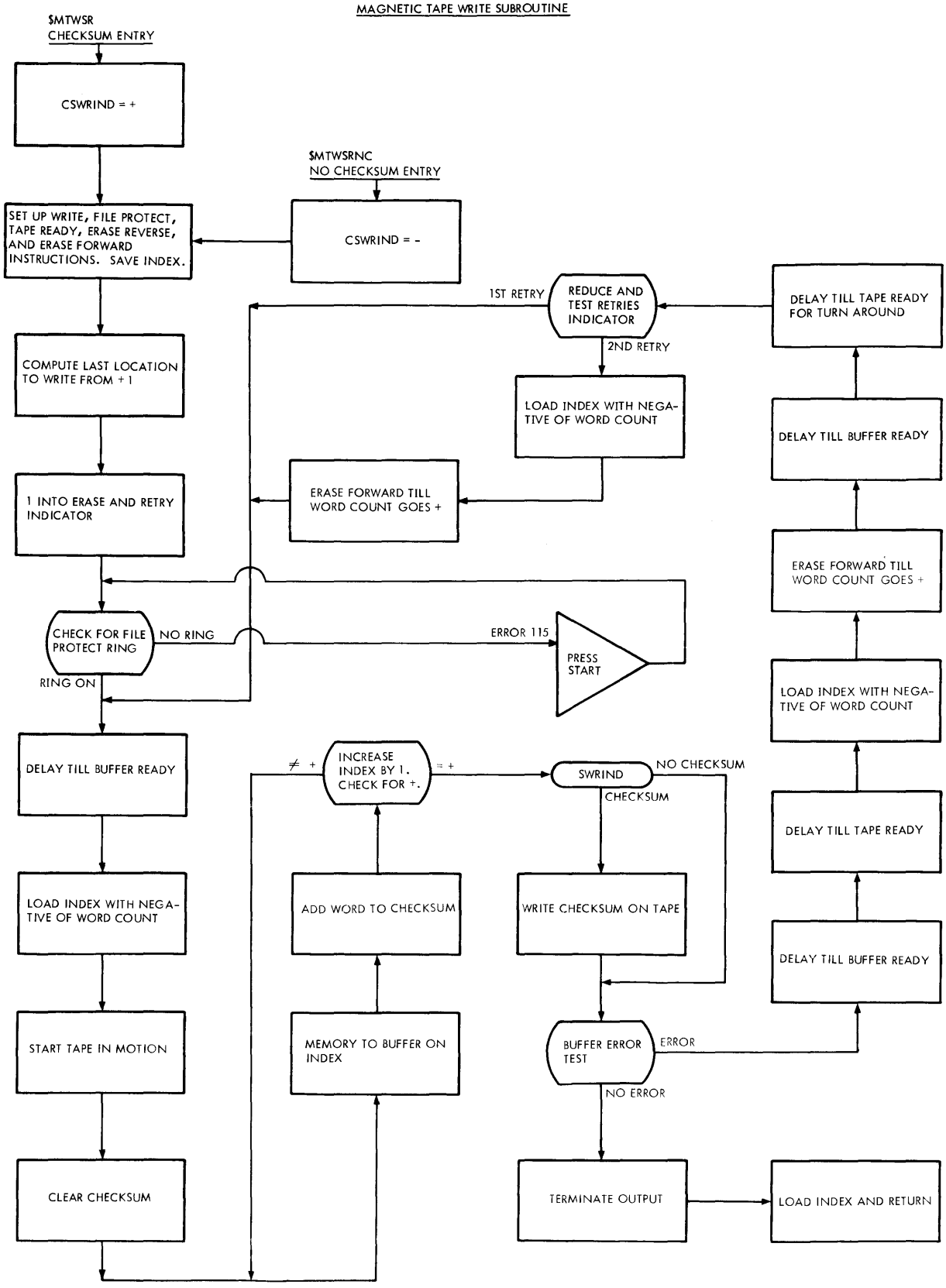
CARD READ SUBROUTINE

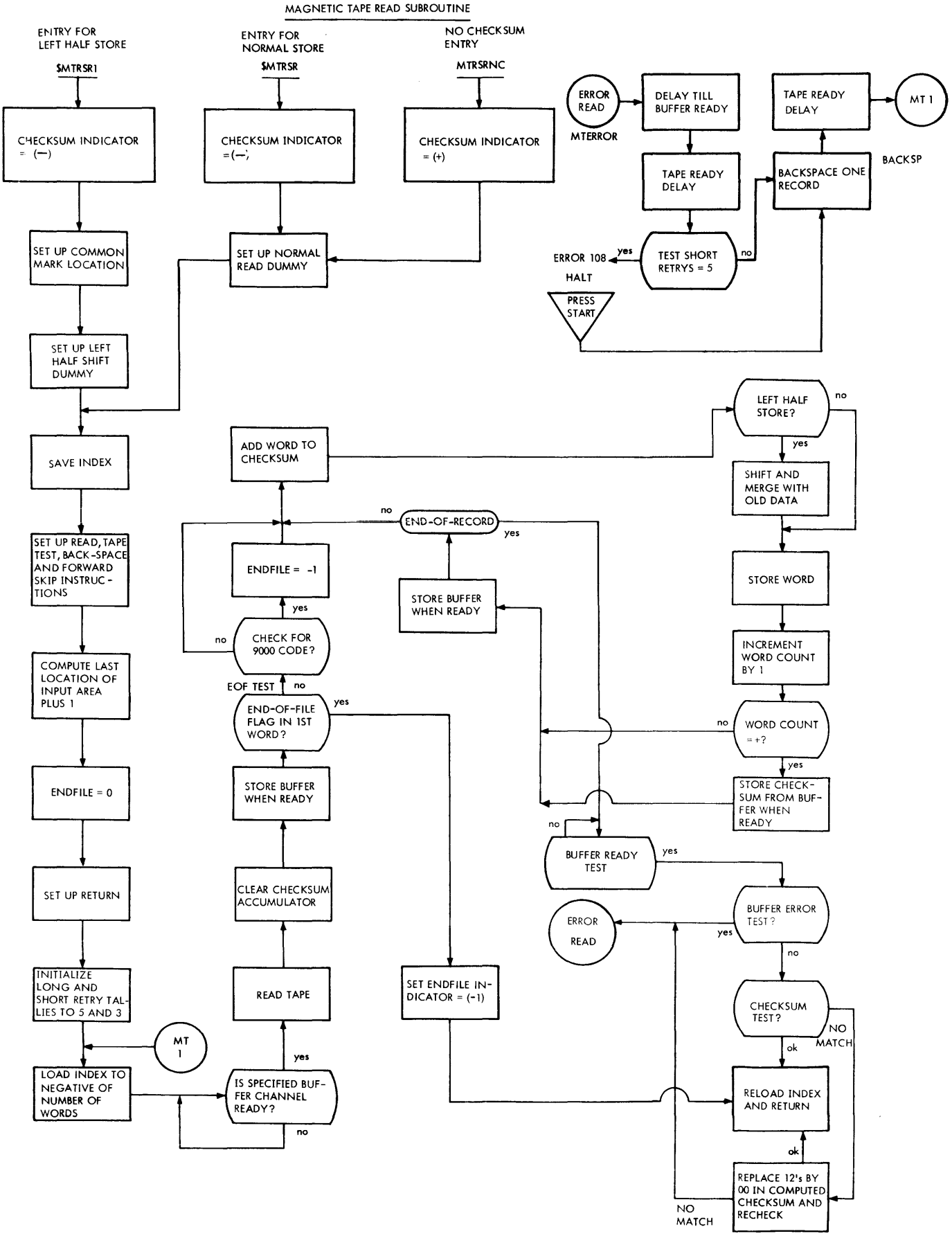


PAPER TAPE READ SUBROUTINE

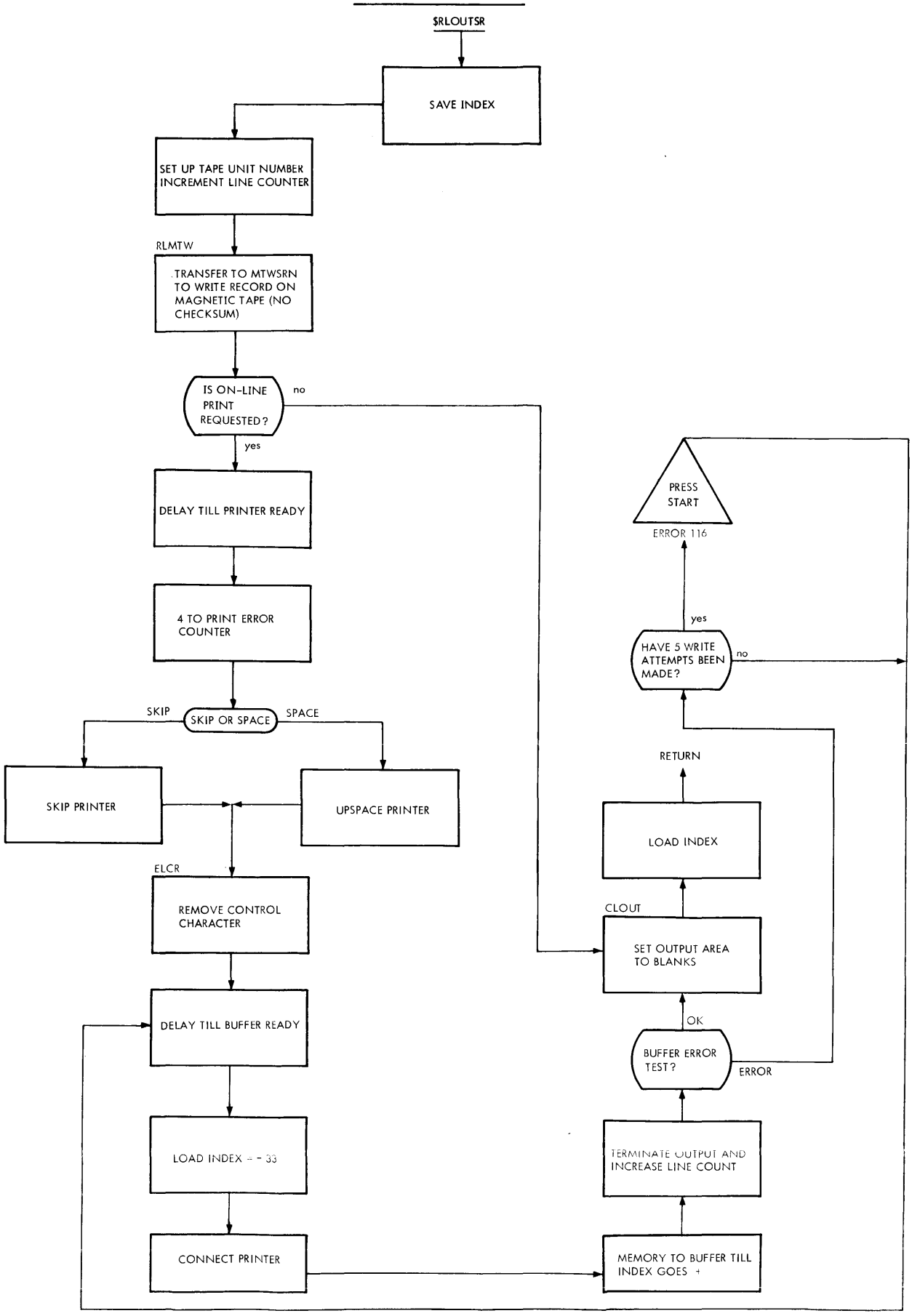


MAGNETIC TAPE WRITE SUBROUTINE

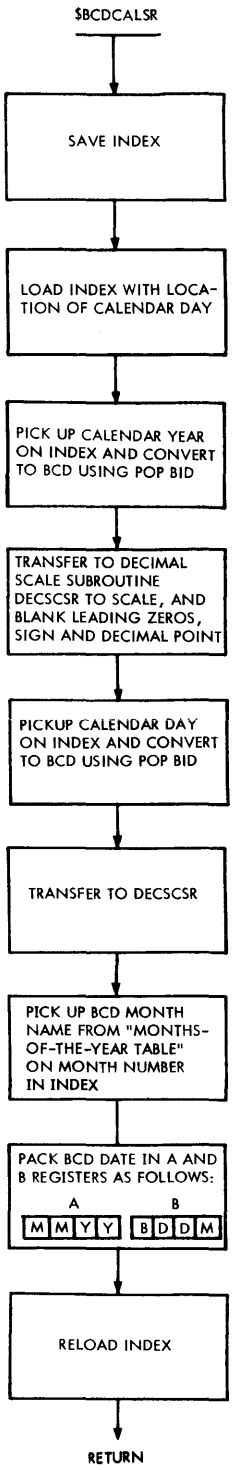



MAGNETIC TAPE READ SUBROUTINE
ENTRY FOR LEFT HALF STORE
$MTRSR1
ENTRY FOR NORMAL STORE
$MTRSR
NO CHECKSUM ENTRY
MTRSRNC
CHECKSUM INDICATOR = (—)
CHECKSUM INDICATOR =(—)
CHECKSUM INDICATOR = (+)
SET UP COMMON MARK LOCATION
SET UP NORMAL READ DUMMY
SET UP LEFT HALF SHIFT DUMMY
SAVE INDEX
SET UP READ, TAPE TEST, BACK-SPACE AND FORWARD SKIP INSTRUC-TIONS
COMPUTE LAST LOCATION OF INPUT AREA PLUS 1
ENDFILE = 0
SET UP RETURN
INITIALIZE LONG AND SHORT RETRY TAL-LIES TO 5 AND 3
LOAD INDEX TO NEGATIVE OF NUMBER OF WORDS
MT 1
IS SPECIFIED BUF-FER CHANNEL READY?
yes
no
READ TAPE
CLEAR CHECKSUM ACCUMULATOR
STORE BUFFER WHEN READY
END-OF-FILE FLAG IN 1ST WORD?
yes
no
EOF TEST
CHECK FOR 9000 CODE?
yes
no
ENDFILE = -1
ADD WORD TO CHECKSUM
END-OF-RECORD
no
yes
STORE BUFFER WHEN READY
SET ENDFILE IN-DICATOR = (-1)
LEFT HALF STORE?
no
yes
SHIFT AND MERGE WITH OLD DATA
STORE WORD
INCREMENT WORD COUNT BY 1
WORD COUNT = +?
no
yes
STORE CHECK-SUM FROM BUF-FER WHEN READY
BUFFER READY TEST
no
yes
BUFFER ERROR TEST?
yes
no
ERROR READ
CHECKSUM TEST?
NO MATCH
ok
RELOAD INDEX AND RETURN
ok
REPLACE 12's BY 00 IN COMPUTED CHECKSUM AND RECHECK
NO MATCH
ERROR READ
MTERROR
DELAY TILL BUFFER READY
TAPE READY DELAY
TEST SHORT RETRYS = 5
yes
no
ERROR 108 HALT
PRESS START
BACKSPACE ONE RECORD
TAPE READY DELAY
MT 1
BACKSP

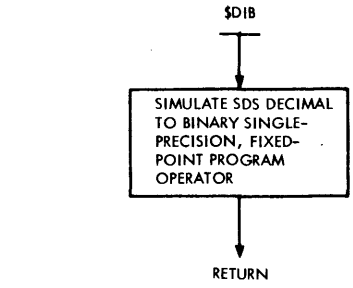
REPORT LINE OUT SUBROUTINE



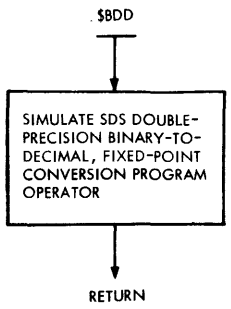
BINARY TO BCD CALENDAR DATE CONVERSION SUBROUTINE



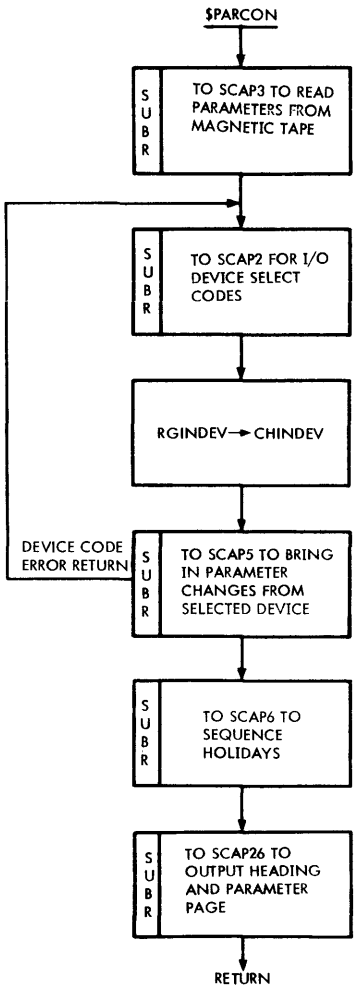
DECIMAL TO BINARY CONVERSION SUBROUTINE



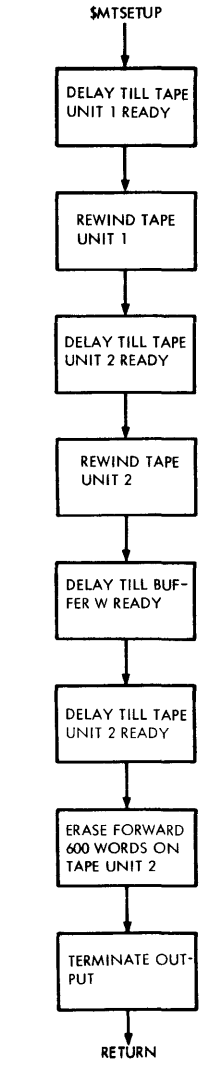
DOUBLE-PRECISION BINARY TO BCD CONVERSION SUBROUTINE



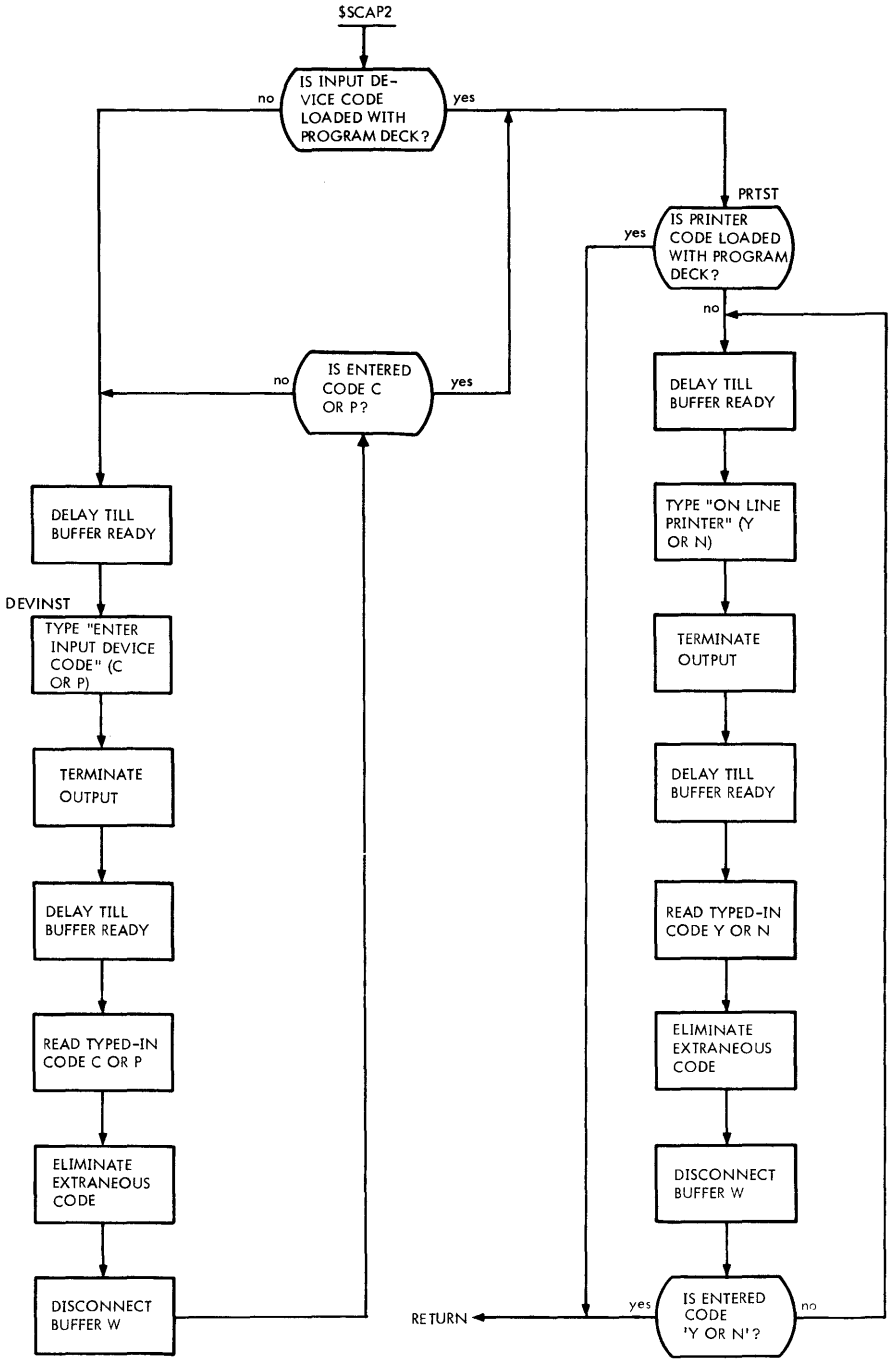
PARAMETER CONTROL SUBROUTINE



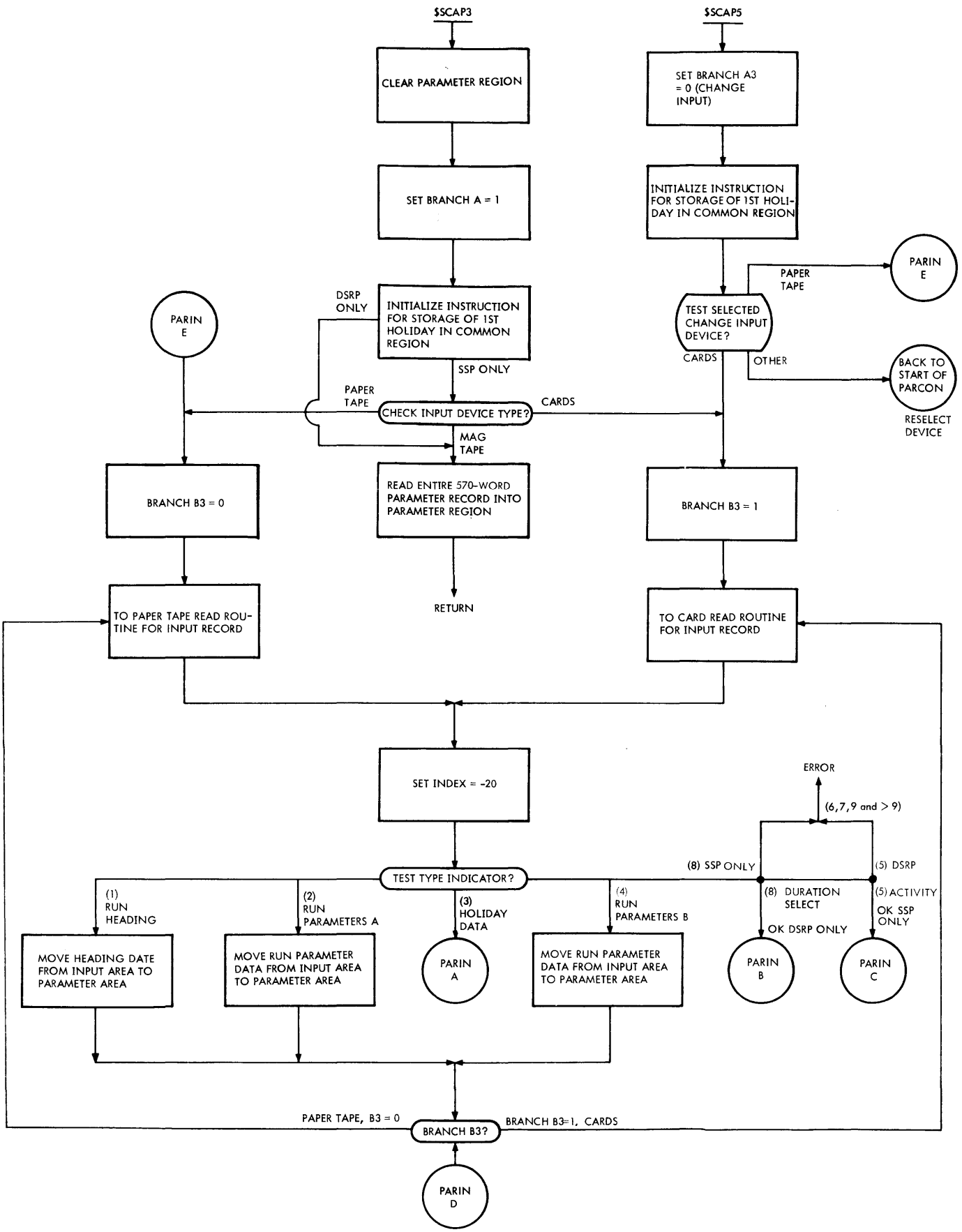
MAGNETIC TAPE SETUP ROUTINE



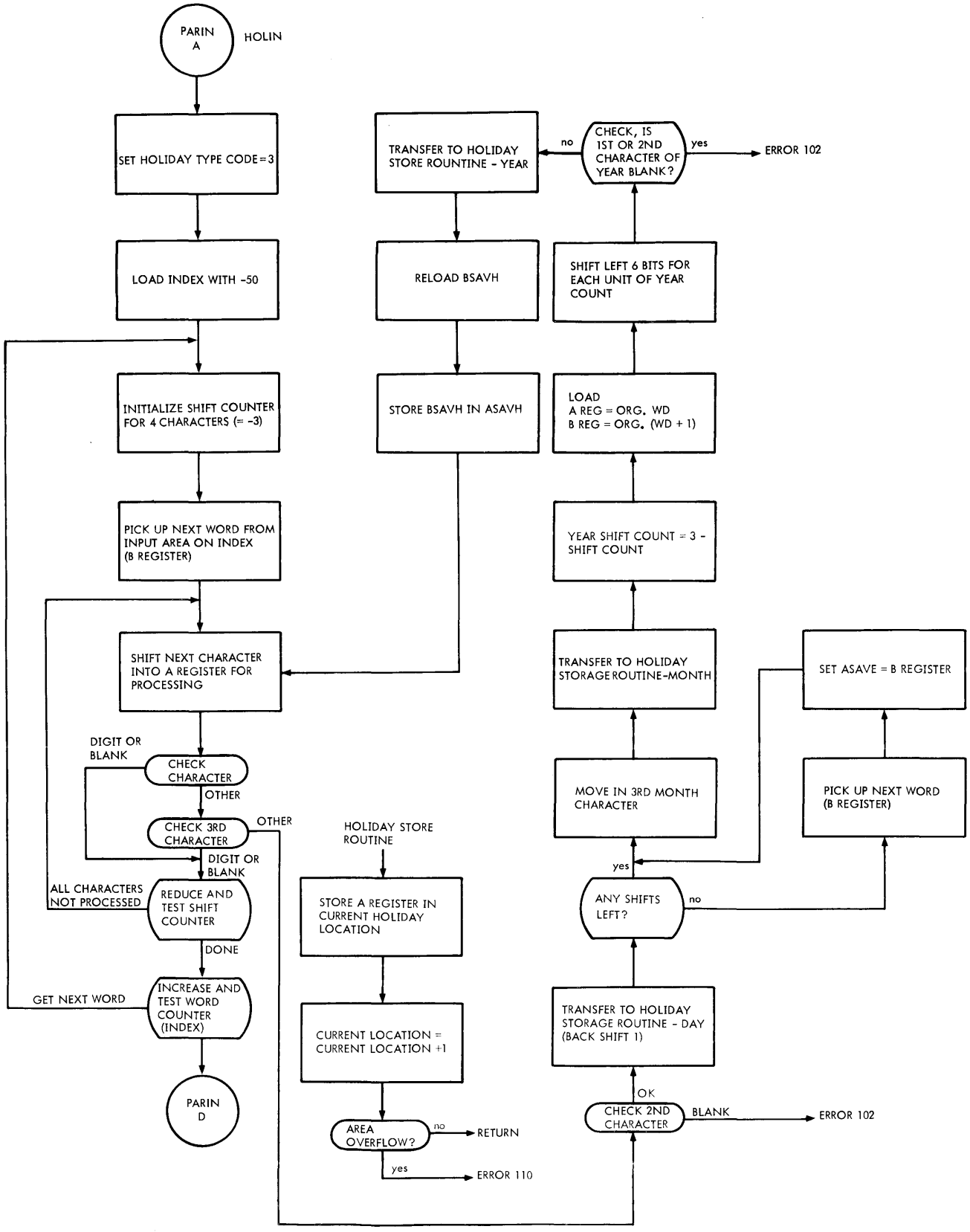
I/O DEVICE SELECTION ROUTINE



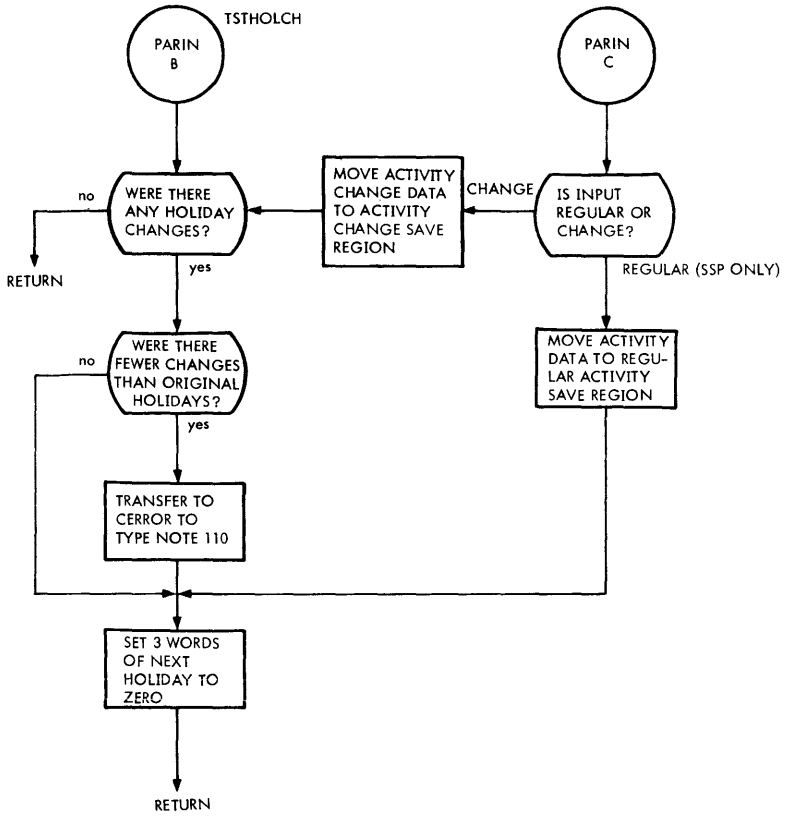
PARAMETER INPUT ROUTINE



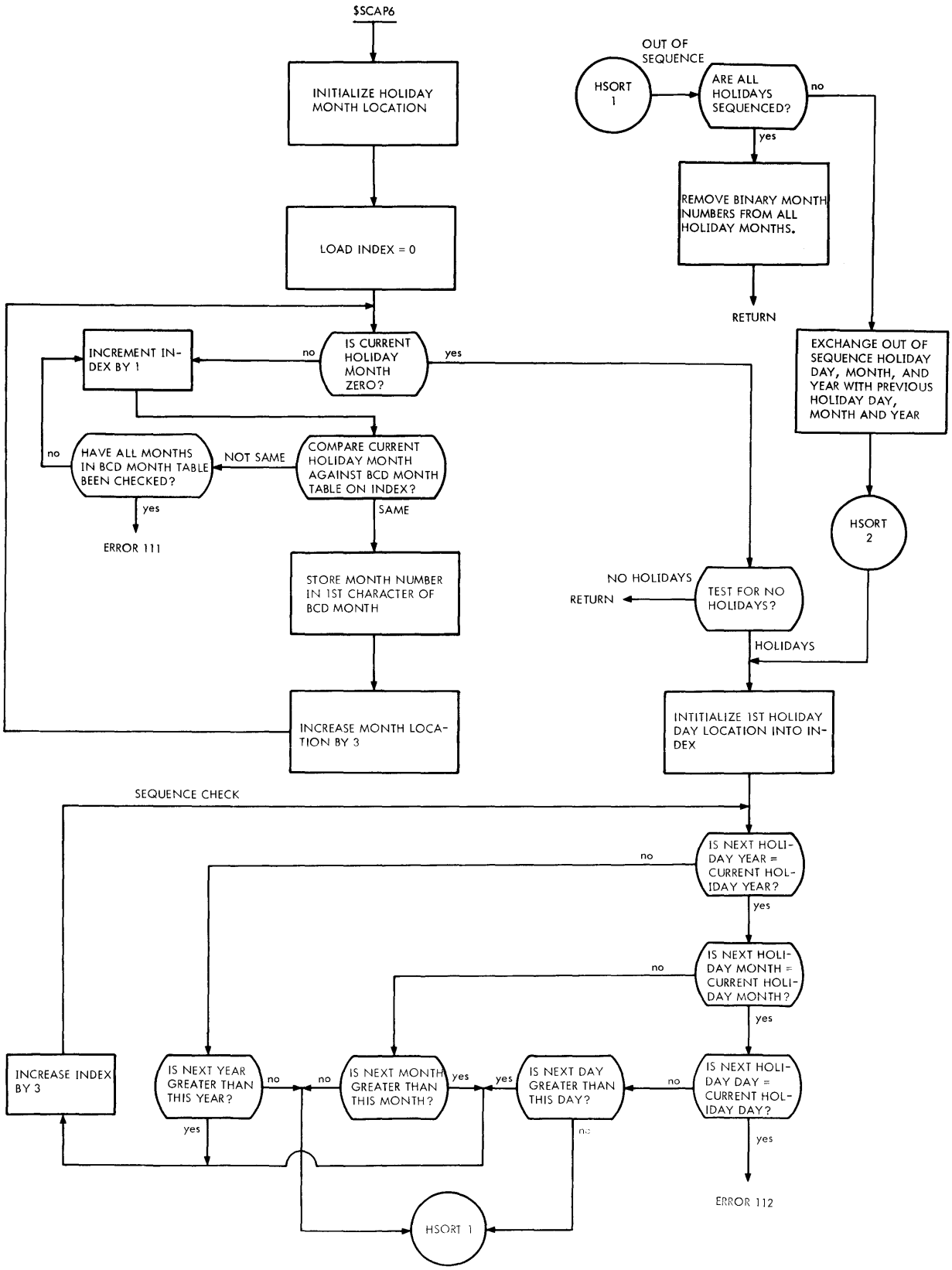
PARAMETER INPUT ROUTINE (cont.)



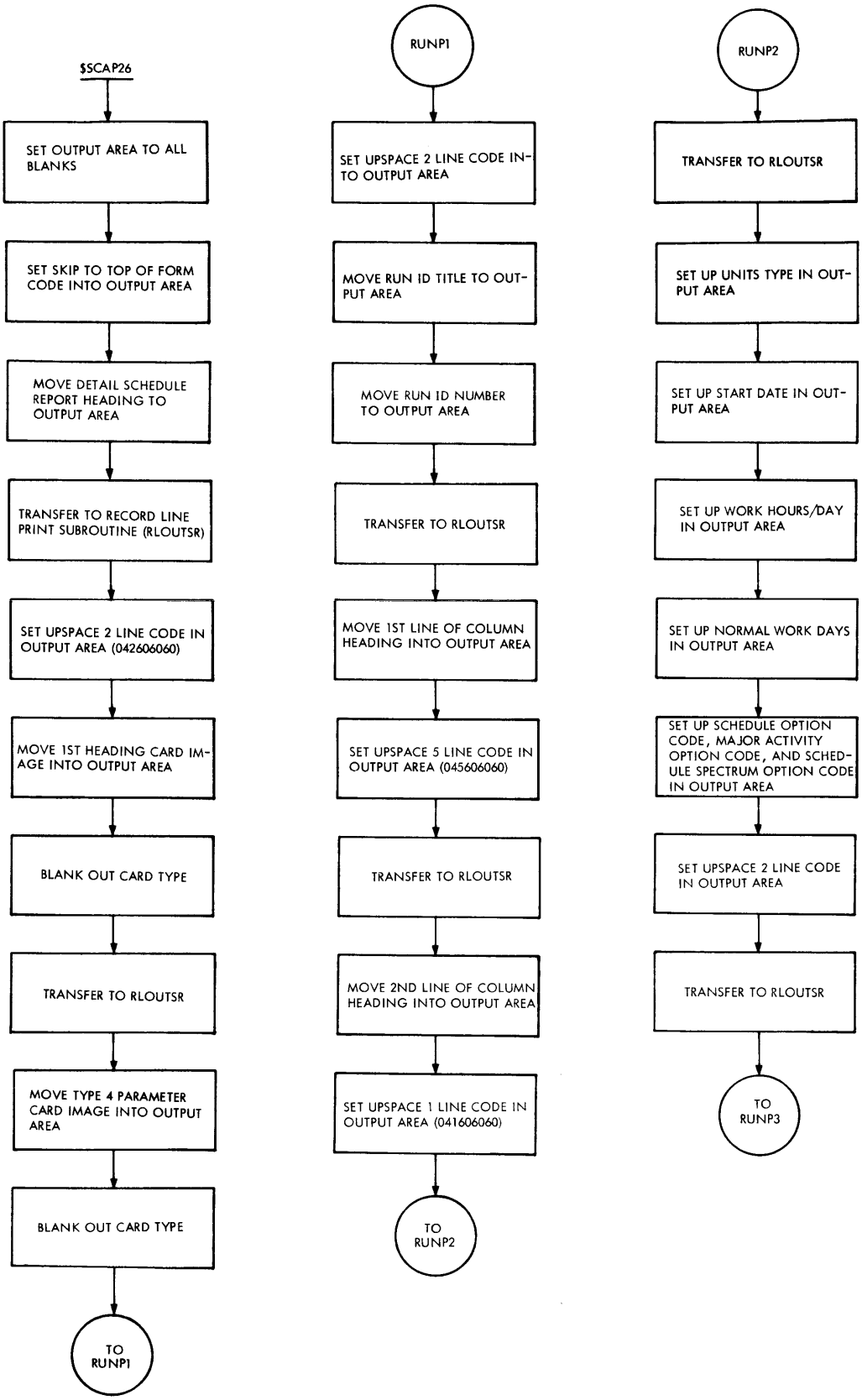
PARAMETER INPUT ROUTINE (cont.)

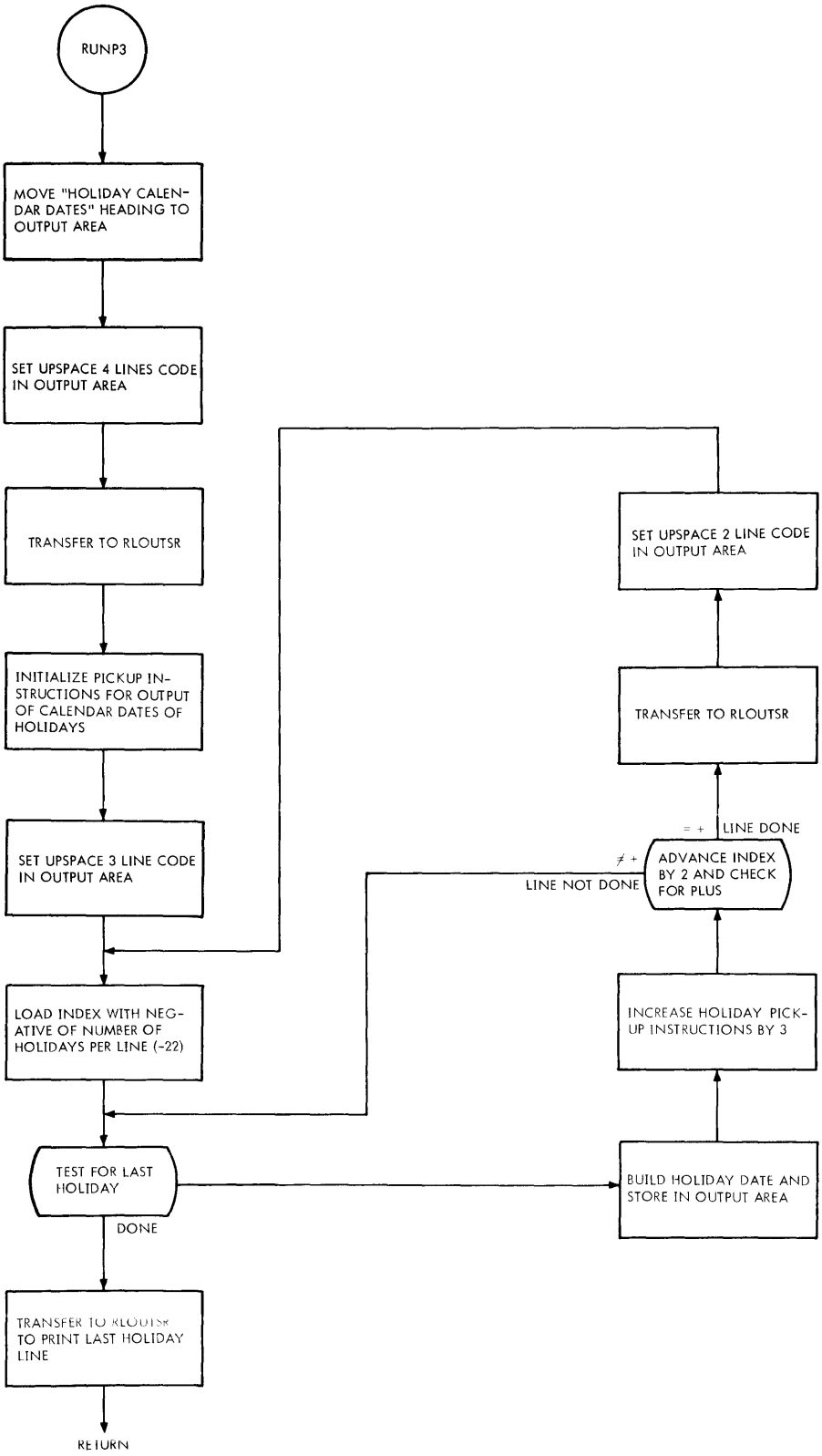


HOLIDAY SORT SUBROUTINE

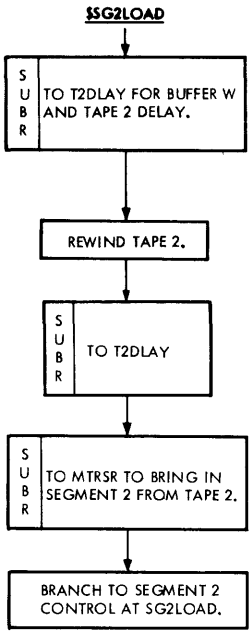


RUN HEADING AND PARAMETER PRINT SUBROUTINE

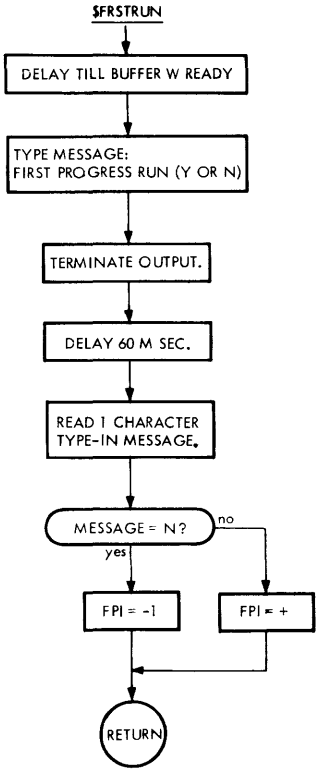



RUNP3
MOVE "HOLIDAY CALENDAR DATES" HEADING TO OUTPUT AREA
SET UPSPACE 4 LINES CODE IN OUTPUT AREA
TRANSFER TO RLOUTSR
INITIALIZE PICKUP INSTRUCTIONS FOR OUTPUT OF CALENDAR DATES OF HOLIDAYS
SET UPSPACE 3 LINE CODE IN OUTPUT AREA
LOAD INDEX WITH NEGATIVE OF NUMBER OF HOLIDAYS PER LINE (-22)
TEST FOR LAST HOLIDAY
DONE
TRANSFER TO RLOUTSR TO PRINT LAST HOLIDAY LINE
RETURN
BUILD HOLIDAY DATE AND STORE IN OUTPUT AREA
INCREASE HOLIDAY PICK-UP INSTRUCTIONS BY 3
ADVANCE INDEX BY 2 AND CHECK FOR PLUS
= +
LINE DONE
≠ +
LINE NOT DONE
TRANSFER TO RLOUTSR
SET UPSPACE 2 LINE CODE IN OUTPUT AREA

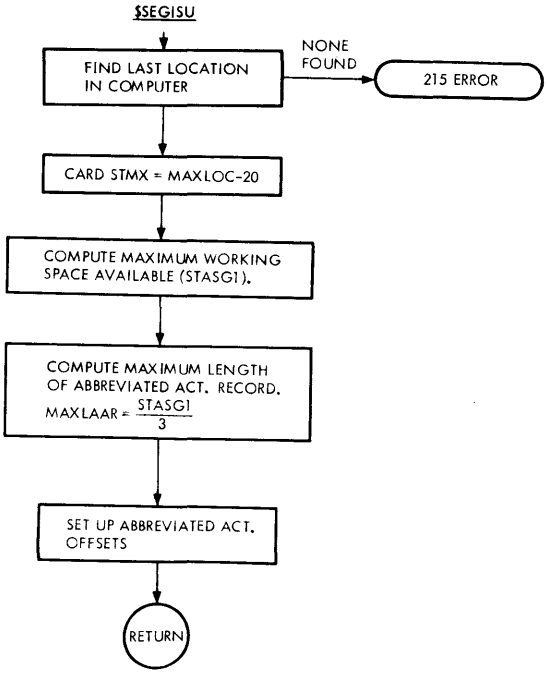
SEGMENT 2 LOAD SUBROUTINE

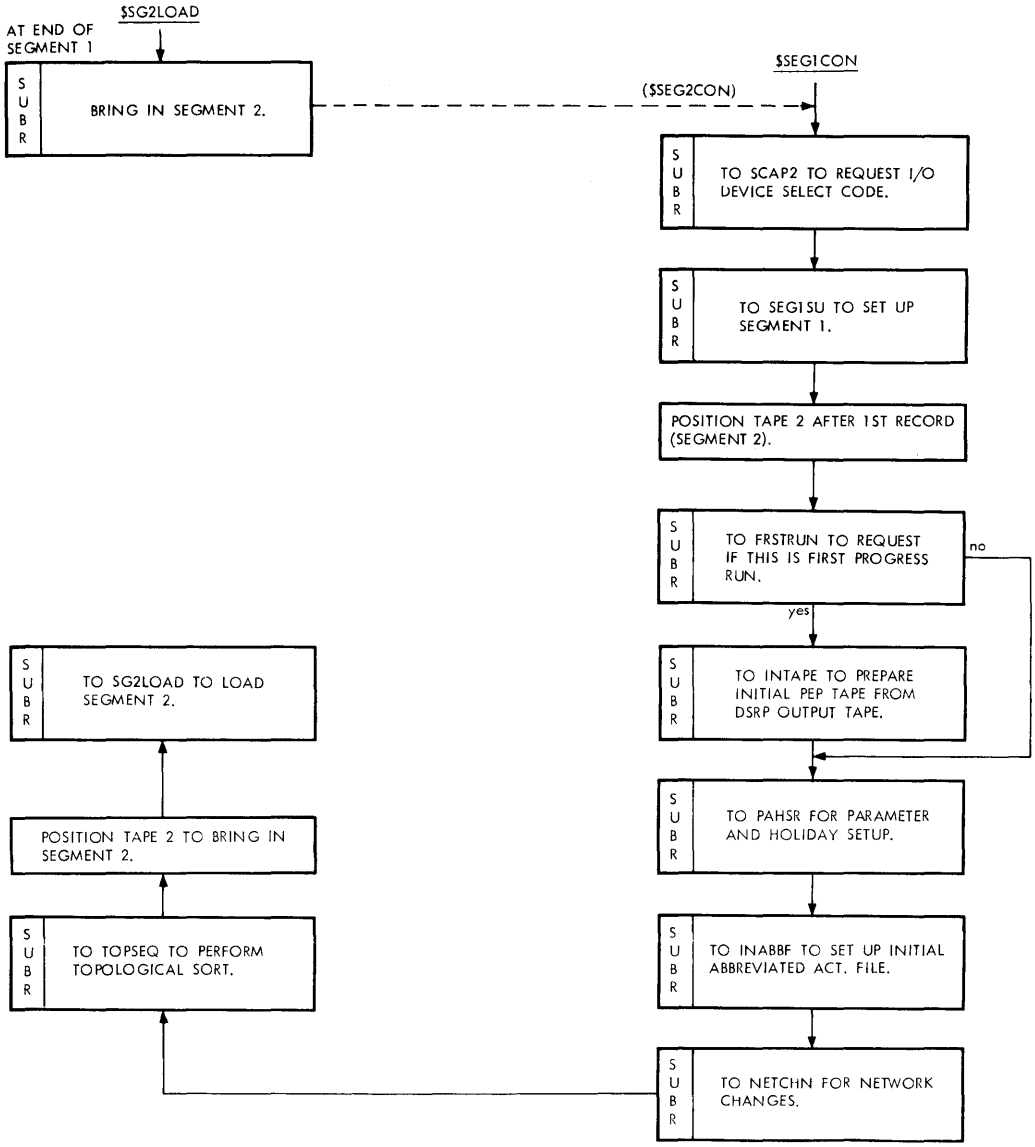


FIRST RUN REQUEST SUBROUTINE (F. D. F05)

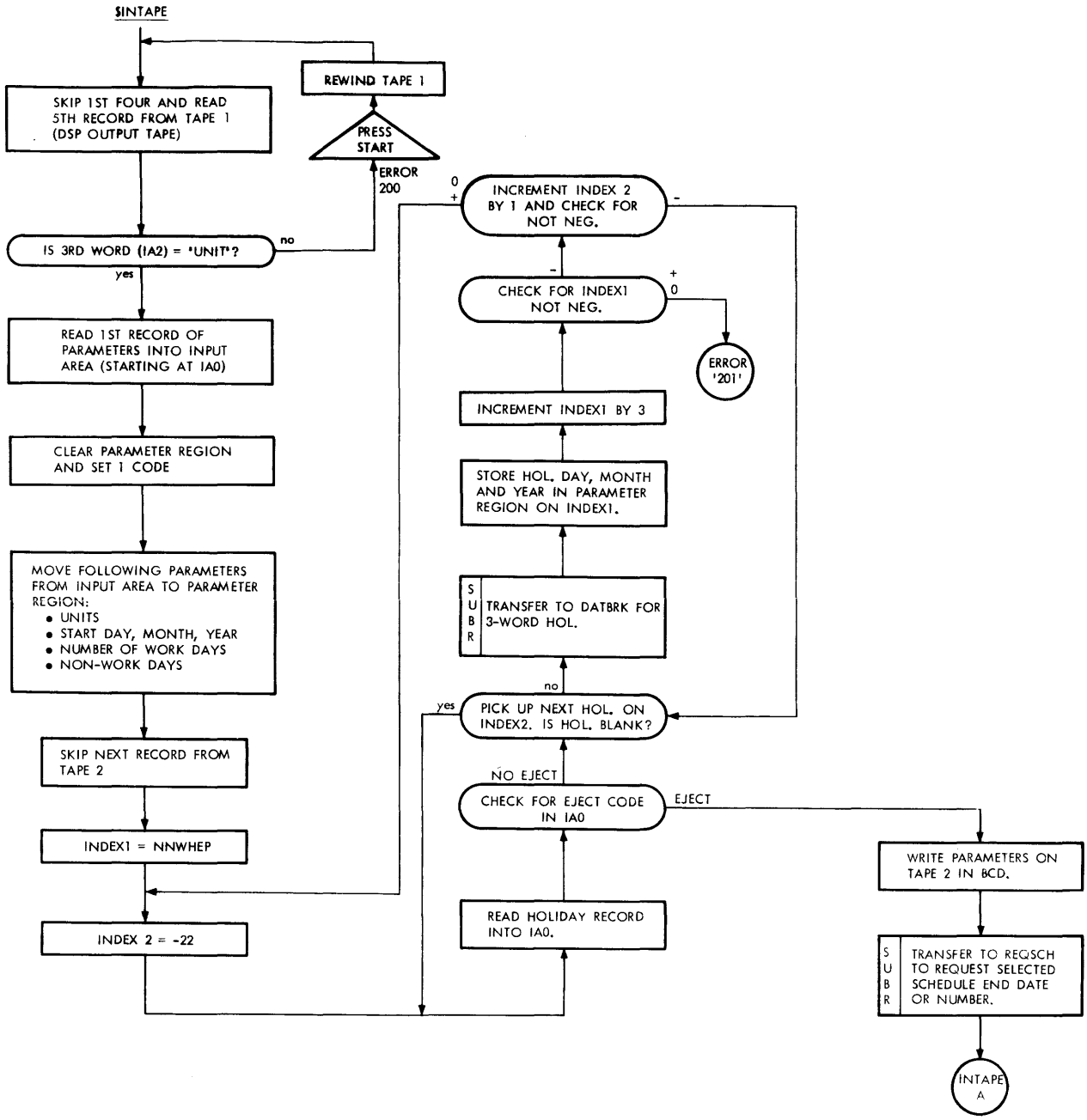


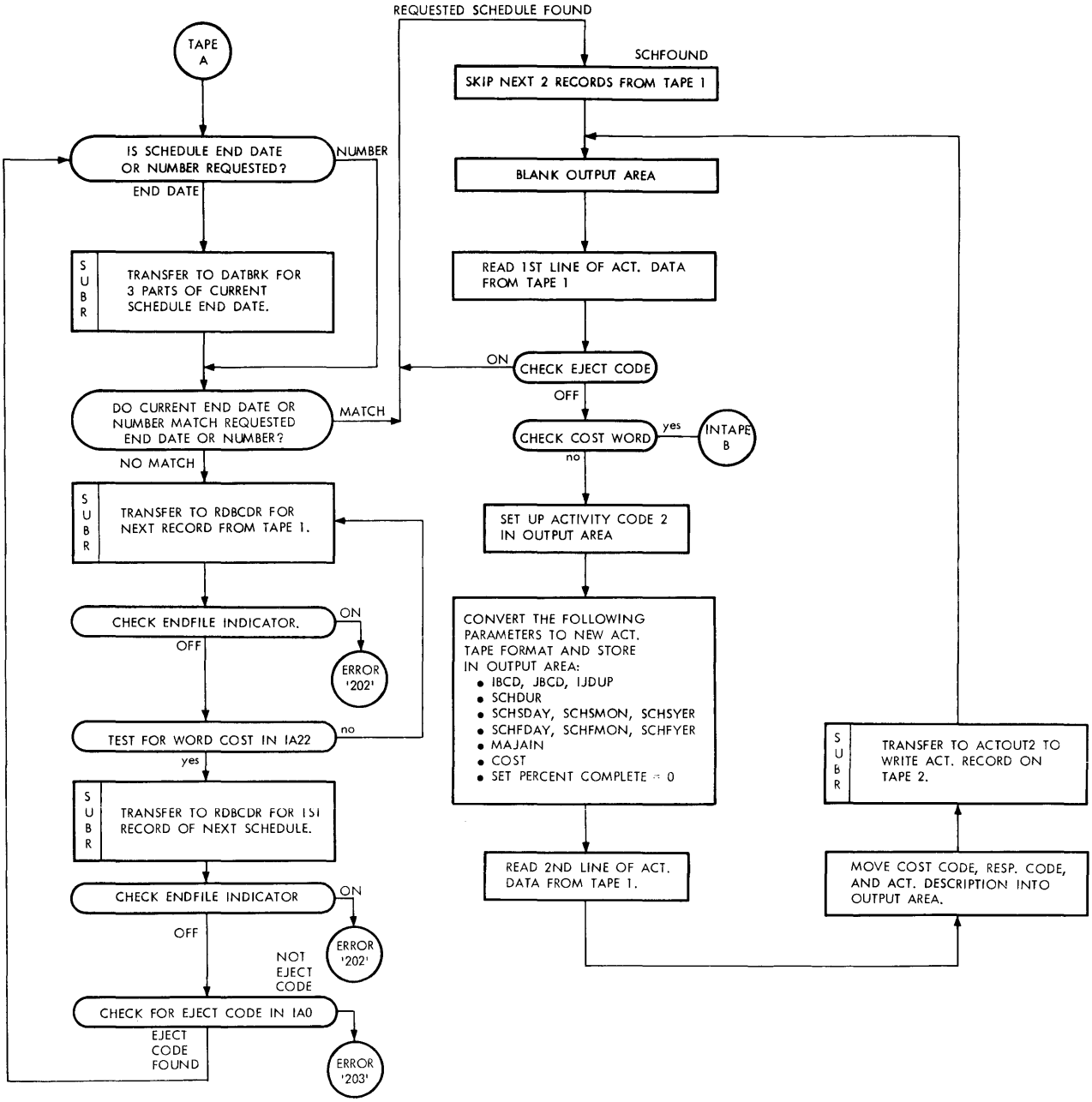
SEGMENT 1 SETUP SUBROUTINE (F. D. F07)

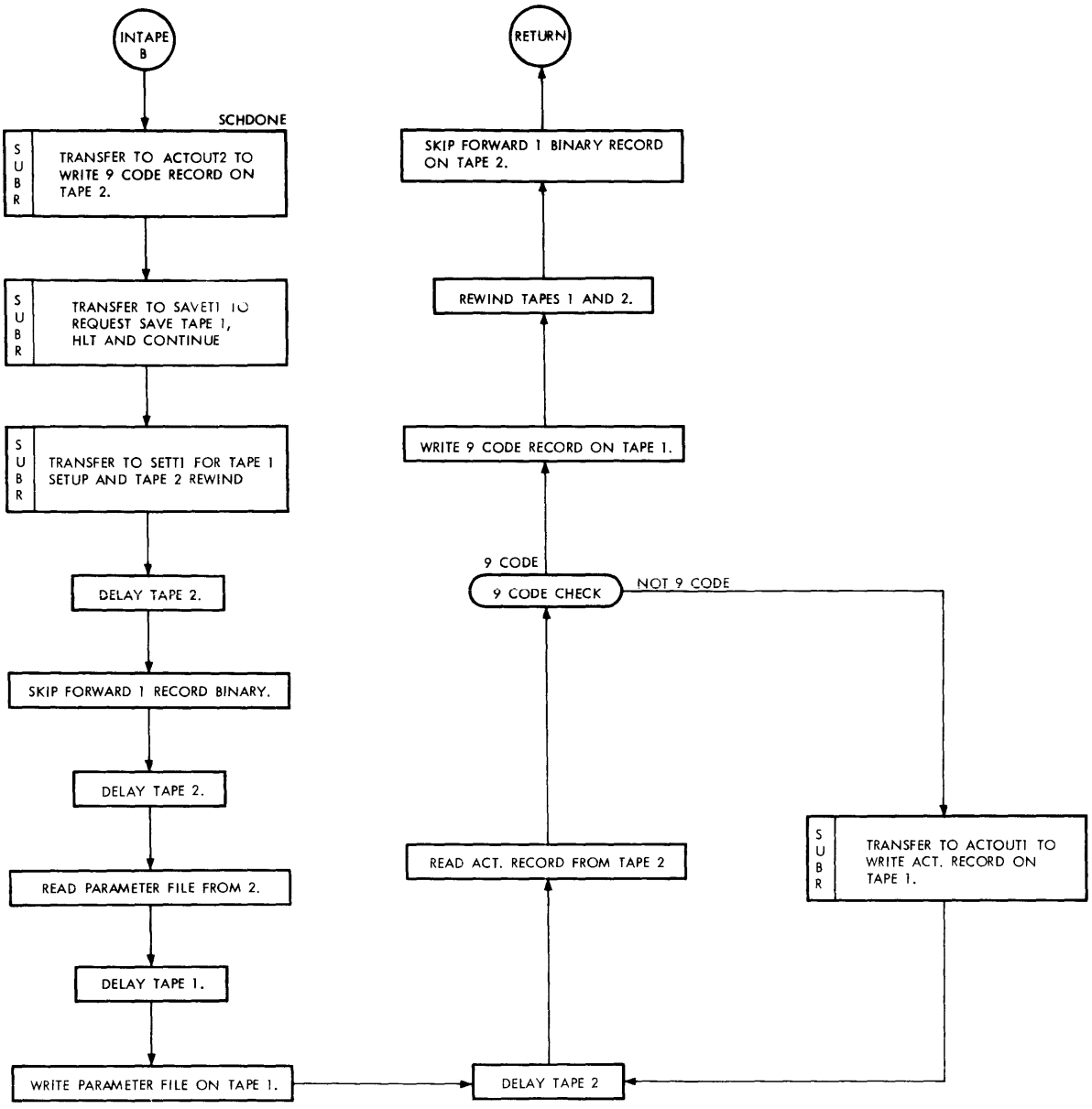



$SG2LOAD
AT END OF SEGMENT 1
SUBR
BRING IN SEGMENT 2.
($SEG2CON)
$SEG1CON
SUBR
TO SCAP2 TO REQUEST I/O DEVICE SELECT CODE.
SUBR
TO SEG1SU TO SET UP SEGMENT 1.
POSITION TAPE 2 AFTER 1ST RECORD (SEGMENT 2).
SUBR
TO FRSTRUN TO REQUEST IF THIS IS FIRST PROGRESS RUN.
no
yes
SUBR
TO INTAPE TO PREPARE INITIAL PEP TAPE FROM DSRP OUTPUT TAPE.
SUBR
TO PAHSR FOR PARAMETER AND HOLIDAY SETUP.
SUBR
TO INABBF TO SET UP INITIAL ABBREVIATED ACT. FILE.
SUBR
TO NETCHN FOR NETWORK CHANGES.
SUBR
TO TOPSEQ TO PERFORM TOPOLOGICAL SORT.
POSITION TAPE 2 TO BRING IN SEGMENT 2.
SUBR
TO SG2LOAD TO LOAD SEGMENT 2.

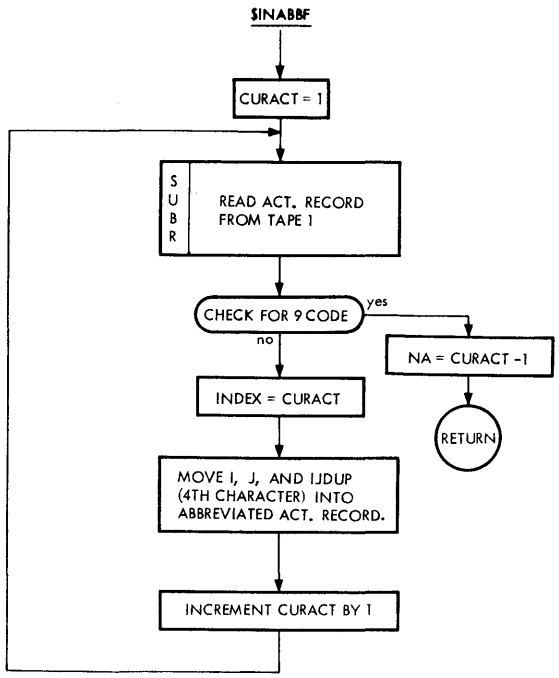
INITIAL INPUT TAPE PREPARATION SUBROUTINE



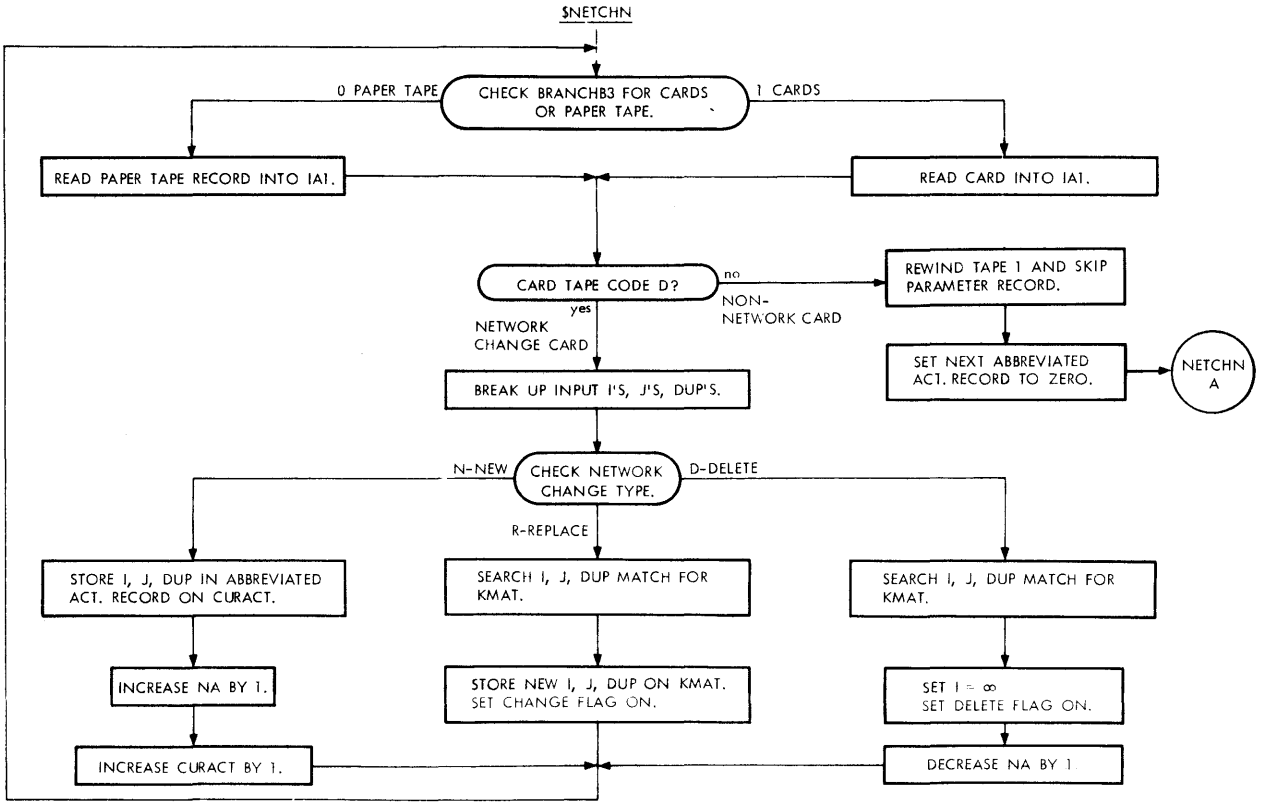

TAPE A
IS SCHEDULE END DATE OR NUMBER REQUESTED?
NUMBER
END DATE
SUBR
TRANSFER TO DATBRK FOR 3 PARTS OF CURRENT SCHEDULE END DATE.
DO CURRENT END DATE OR NUMBER MATCH REQUESTED END DATE OR NUMBER?
MATCH
NO MATCH
SUBR
TRANSFER TO RDBCDR FOR NEXT RECORD FROM TAPE 1.
CHECK ENDFILE INDICATOR.
ON
OFF
ERROR '202'
TEST FOR WORD COST IN IA22
no
yes
SUBR
TRANSFER TO RDBCDR FOR 1ST RECORD OF NEXT SCHEDULE.
CHECK ENDFILE INDICATOR
ON
OFF
ERROR '202'
NOT EJECT CODE
CHECK FOR EJECT CODE IN IA0
EJECT CODE FOUND
ERROR '203'
REQUESTED SCHEDULE FOUND
SCHFOUND
SKIP NEXT 2 RECORDS FROM TAPE 1
BLANK OUTPUT AREA
READ 1ST LINE OF ACT. DATA FROM TAPE 1
ON
CHECK EJECT CODE
OFF
CHECK COST WORD
yes
INTAPE B
no
SET UP ACTIVITY CODE 2 IN OUTPUT AREA
CONVERT THE FOLLOWING PARAMETERS TO NEW ACT. TAPE FORMAT AND STORE IN OUTPUT AREA:
• IBCD, JBCD, IJDUP
• SCHDUR
• SCHSDAY, SCHSMON, SCHSYER
• SCHFDAY, SCHFMON, SCHFYER
• MAJAIN
• COST
• SET PERCENT COMPLETE = 0
READ 2ND LINE OF ACT. DATA FROM TAPE 1.
MOVE COST CODE, RESP. CODE, AND ACT. DESCRIPTION INTO OUTPUT AREA.
SUBR
TRANSFER TO ACTOUT2 TO WRITE ACT. RECORD ON TAPE 2.

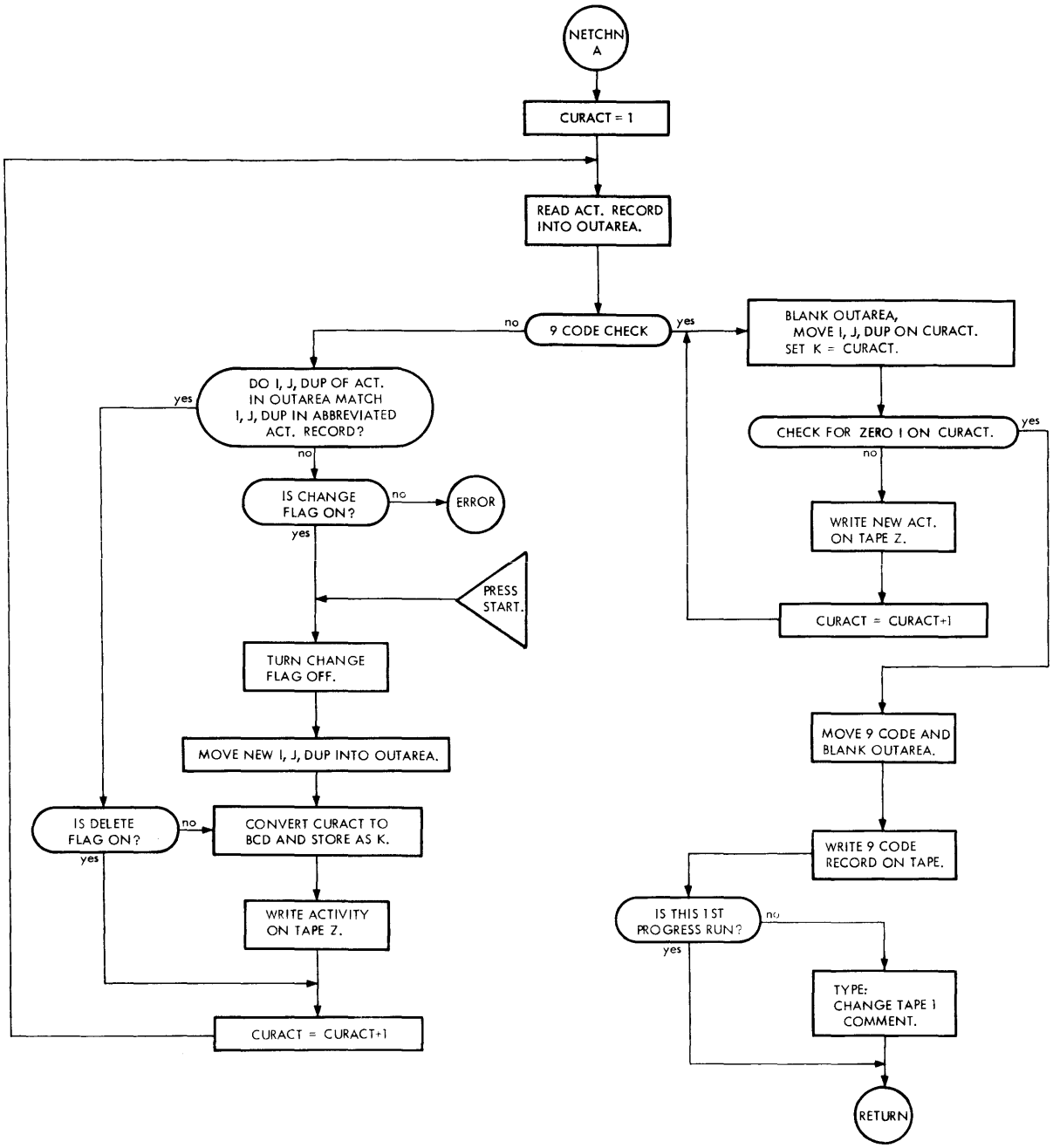

INTAPE B
SCHDONE
SUBR
TRANSFER TO ACTOUT2 TO WRITE 9 CODE RECORD ON TAPE 2.
SUBR
TRANSFER TO SAVETI TO REQUEST SAVE TAPE 1, HLT AND CONTINUE
SUBR
TRANSFER TO SETTI FOR TAPE 1 SETUP AND TAPE 2 REWIND
DELAY TAPE 2.
SKIP FORWARD 1 RECORD BINARY.
DELAY TAPE 2.
READ PARAMETER FILE FROM 2.
DELAY TAPE 1.
WRITE PARAMETER FILE ON TAPE 1.
RETURN
SKIP FORWARD 1 BINARY RECORD ON TAPE 2.
REWIND TAPES 1 AND 2.
WRITE 9 CODE RECORD ON TAPE 1.
9 CODE
9 CODE CHECK
NOT 9 CODE
READ ACT. RECORD FROM TAPE 2
DELAY TAPE 2
SUBR
TRANSFER TO ACTOUTI TO WRITE ACT. RECORD ON TAPE 1.

INITIAL ABBREVIATED ACTIVITY FILE SUBROUTINE

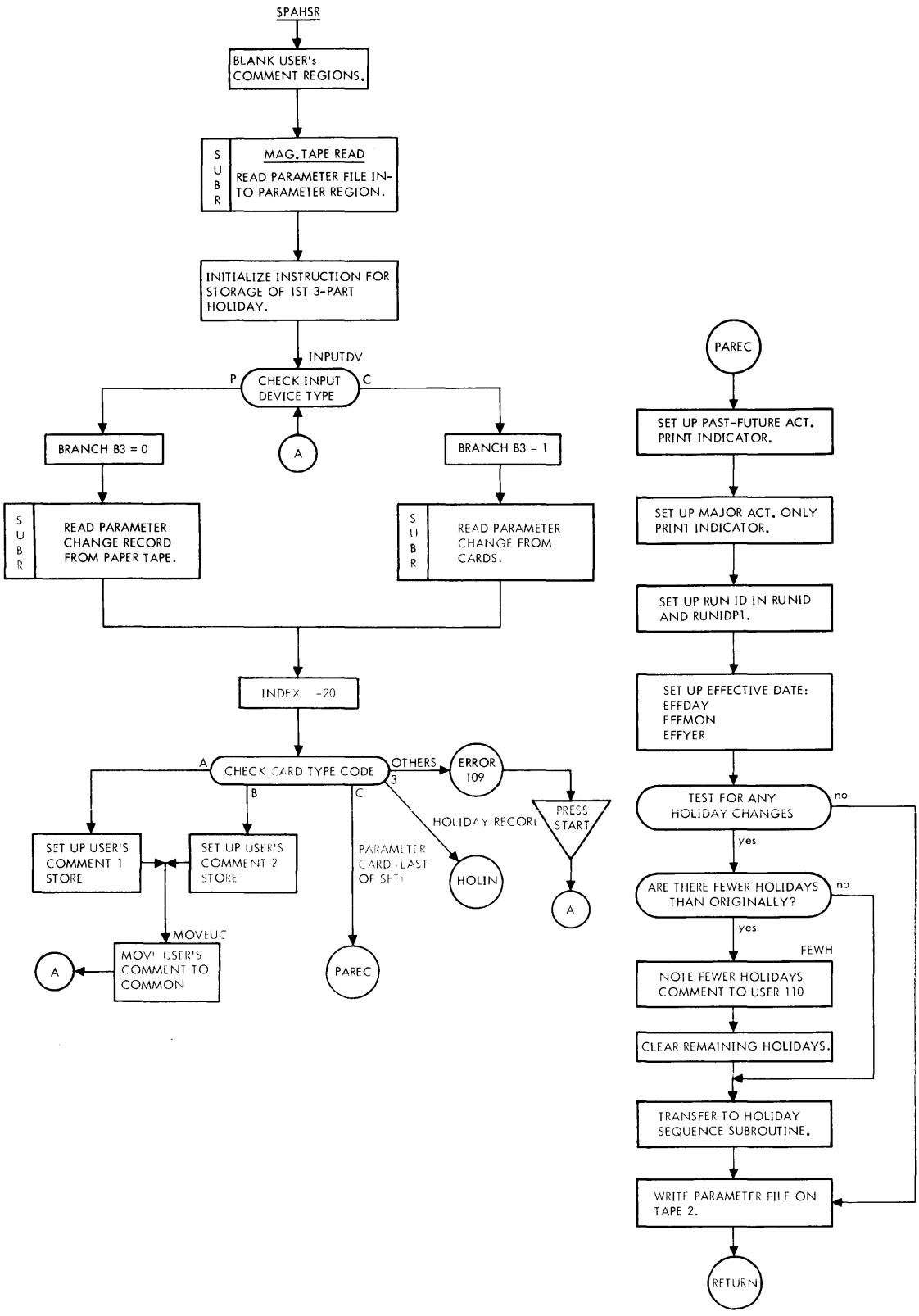


NETWORK CHANGE ROUTINE

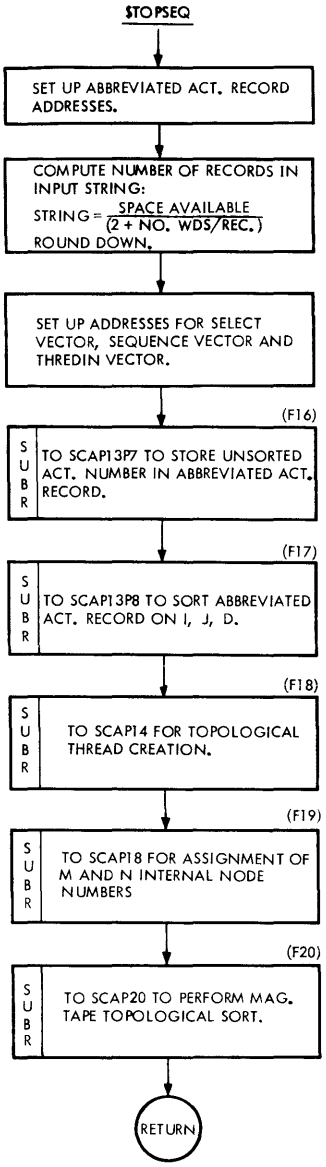



NETCHN A
CURACT = 1
READ ACT. RECORD INTO OUTAREA.
9 CODE CHECK
no
yes
BLANK OUTAREA, MOVE I, J, DUP ON CURACT. SET K = CURACT.
CHECK FOR ZERO 1 ON CURACT.
yes
no
WRITE NEW ACT. ON TAPE Z.
CURACT = CURACT+1
MOVE 9 CODE AND BLANK OUTAREA.
WRITE 9 CODE RECORD ON TAPE.
IS THIS 1ST PROGRESS RUN?
no
yes
TYPE: CHANGE TAPE 1 COMMENT.
RETURN
DO I, J, DUP OF ACT. IN OUTAREA MATCH I, J, DUP IN ABBREVIATED ACT. RECORD?
yes
no
IS CHANGE FLAG ON?
no
ERROR
yes
PRESS START.
TURN CHANGE FLAG OFF.
MOVE NEW I, J, DUP INTO OUTAREA.
IS DELETE FLAG ON?
no
yes
CONVERT CURACT TO BCD AND STORE AS K.
WRITE ACTIVITY ON TAPE Z.
CURACT = CURACT+1

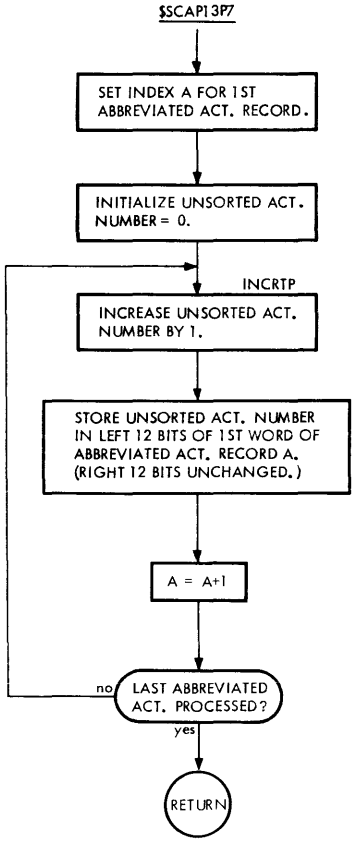
PARAMETER AND HOLIDAY SUBROUTINE (PEP only)

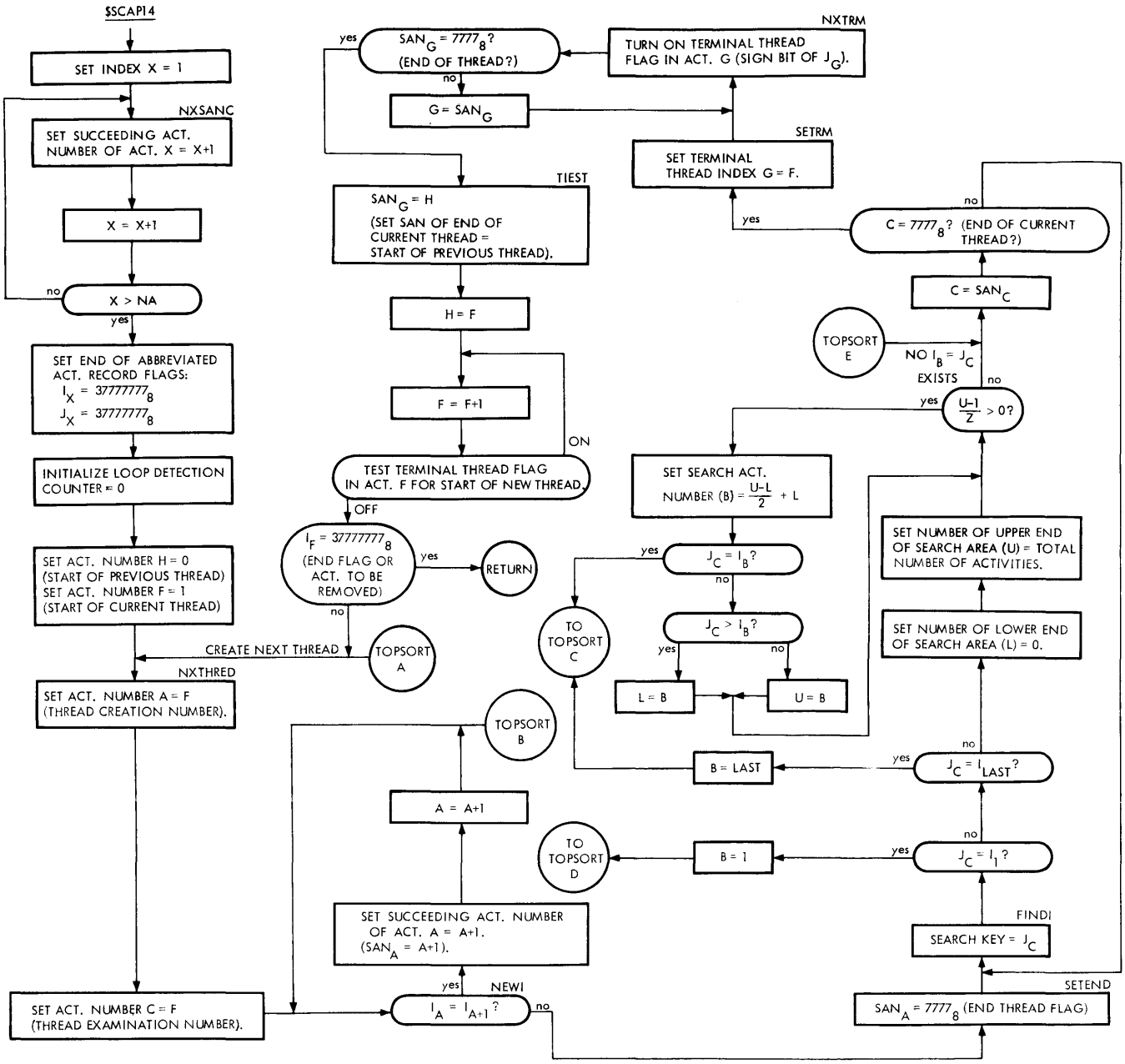


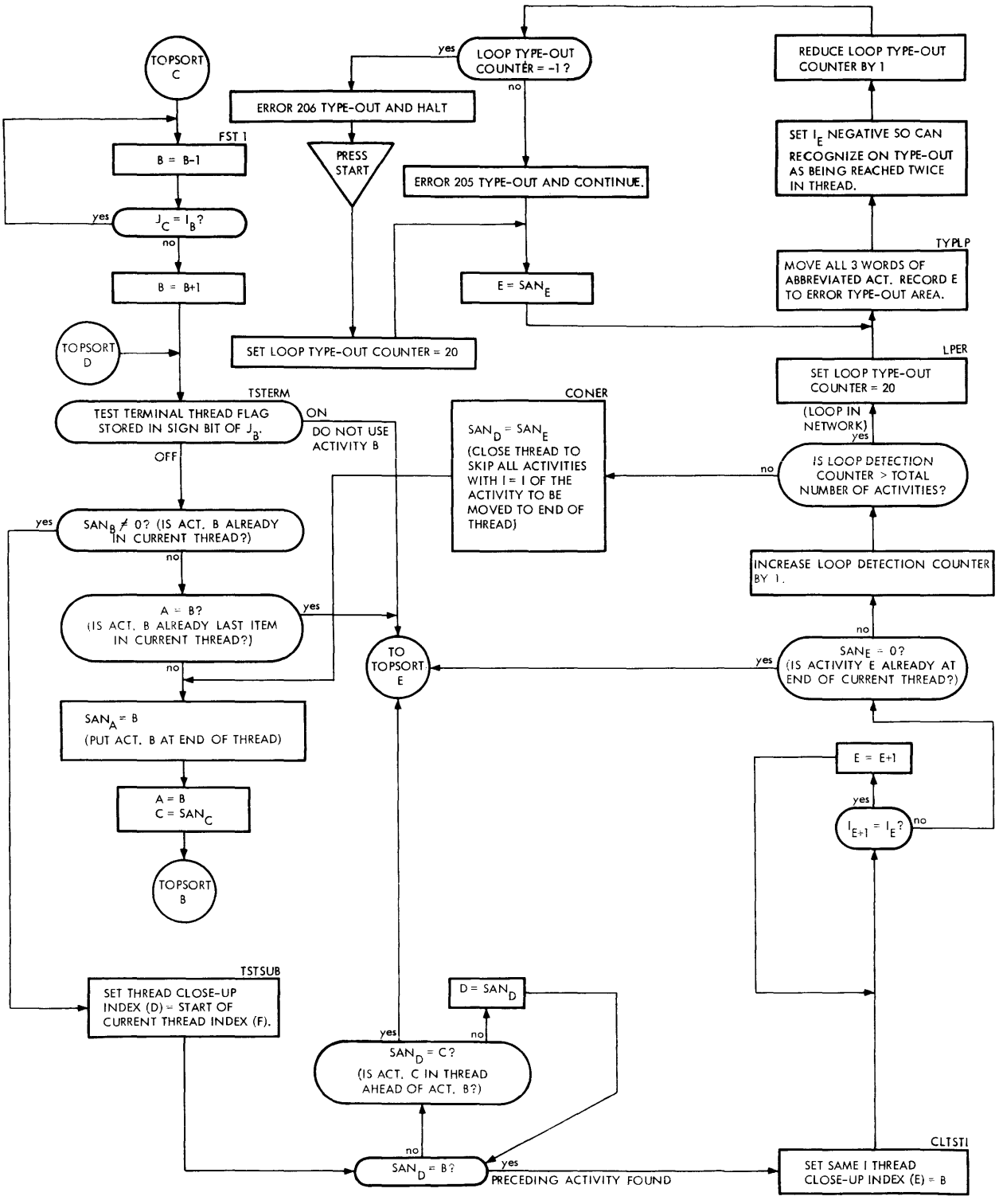
TOPOLOGICAL SEQUENCING CONTROL PROGRAM



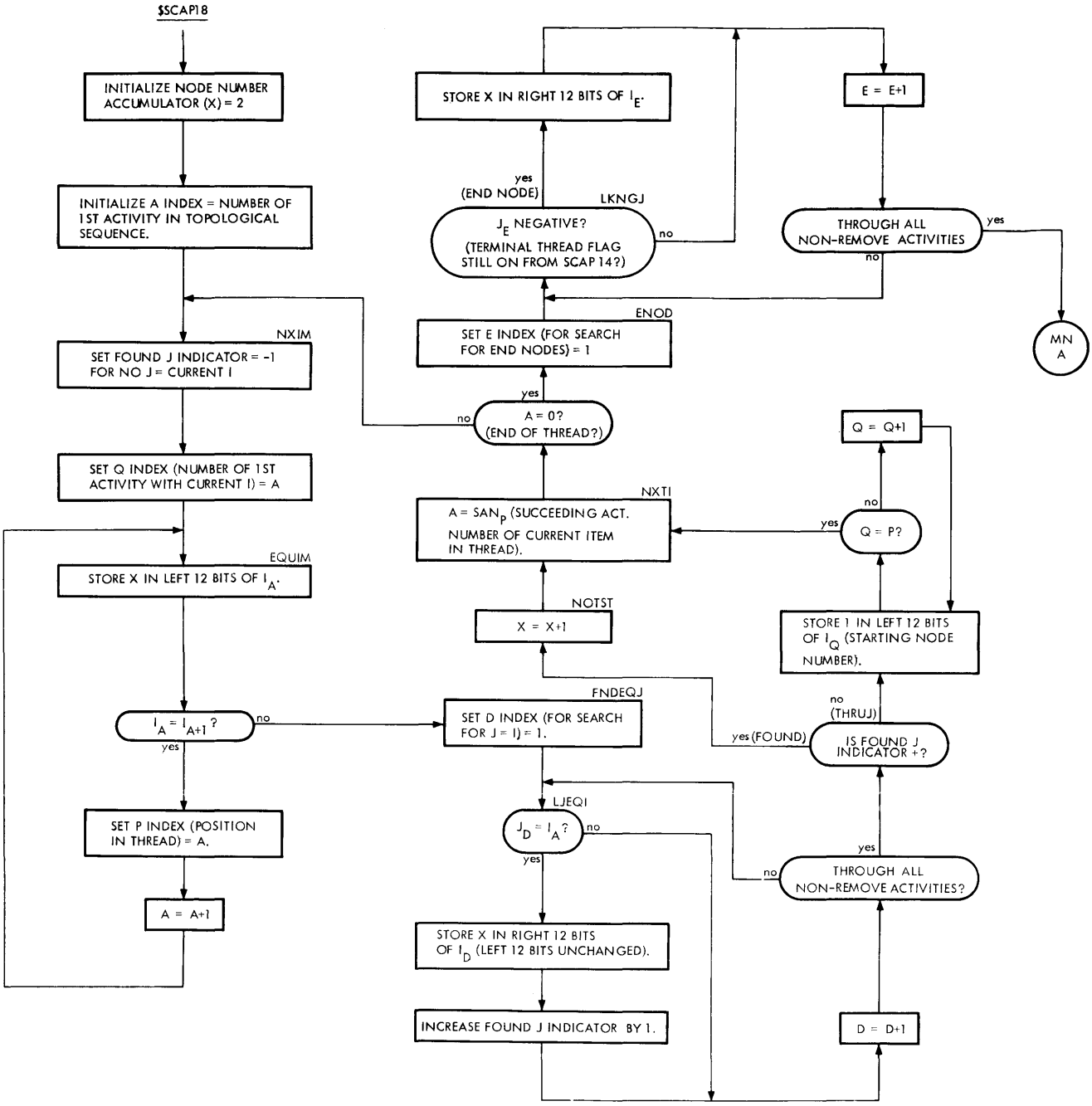
UNSORTED ACTIVITY NUMBER ASSIGNMENT SUBROUTINE

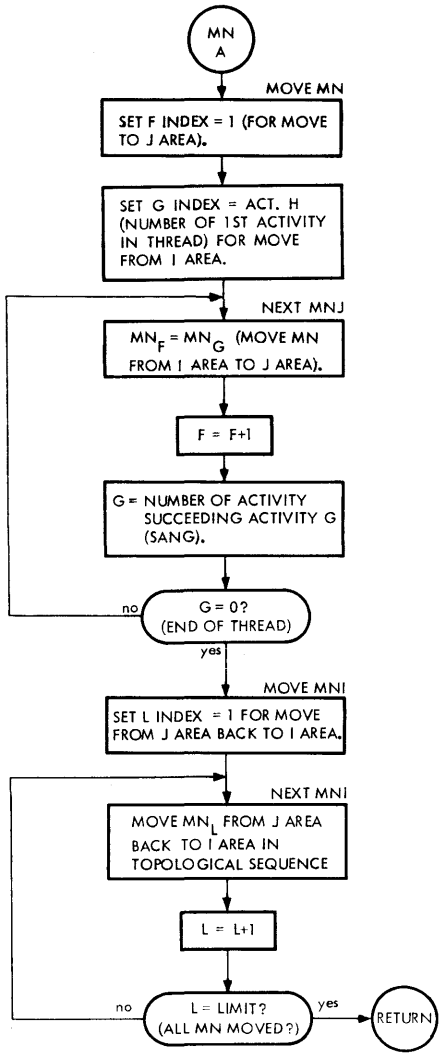


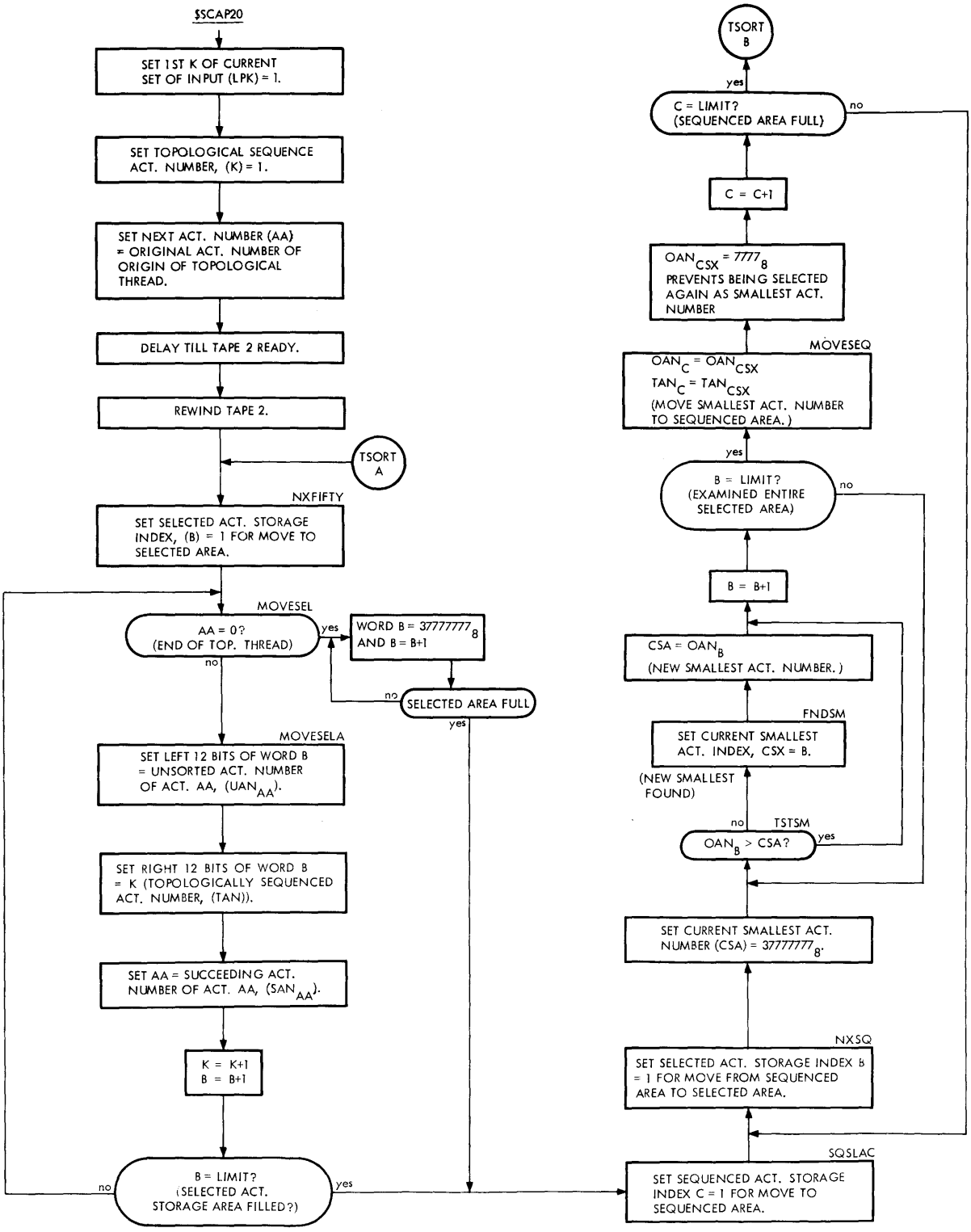

$SCAPI4
SET INDEX X = 1
NXSANC
SET SUCCEEDING ACT. NUMBER OF ACT. X = X+1
X = X+1
no
X > NA
yes
SET END OF ABBREVIATED ACT. RECORD FLAGS:
$I_X = 37777777_8$
$J_X = 37777777_8$
INITIALIZE LOOP DETECTION COUNTER = 0
SET ACT. NUMBER H = 0 (START OF PREVIOUS THREAD)
SET ACT. NUMBER F = 1 (START OF CURRENT THREAD)
NXTHRED
SET ACT. NUMBER A = F (THREAD CREATION NUMBER).
SET ACT. NUMBER C = F (THREAD EXAMINATION NUMBER).
yes
$SAN_G = 7777_8$? (END OF THREAD?)
no
G = SAN_G
TIEST
SAN_G = H (SET SAN OF END OF CURRENT THREAD = START OF PREVIOUS THREAD).
H = F
F = F+1
ON
TEST TERMINAL THREAD FLAG IN ACT. F FOR START OF NEW THREAD.
OFF
$I_F = 37777777_8$ (END FLAG OR ACT. TO BE REMOVED)
yes
RETURN
no
CREATE NEXT THREAD
TOPSORT A
NXTRM
TURN ON TERMINAL THREAD FLAG IN ACT. G (SIGN BIT OF J_G).
SETRM
SET TERMINAL THREAD INDEX G = F.
yes
C = 7777_8? (END OF CURRENT THREAD?)
no
C = SAN_C
TOPSORT E
NO $I_B = J_C$ EXISTS
yes
$\frac{U-1}{Z} > 0$?
no
SET SEARCH ACT. NUMBER (B) = $\frac{U-L}{2}$ + L
yes
$J_C = I_B$?
no
$J_C > I_B$?
yes
no
L = B
U = B
SET NUMBER OF UPPER END OF SEARCH AREA (U) = TOTAL NUMBER OF ACTIVITIES.
SET NUMBER OF LOWER END OF SEARCH AREA (L) = 0.
TO TOPSORT C
B = LAST
yes
no
$J_C = I_{LAST}$?
TOPSORT B
A = A+1
TO TOPSORT D
B = 1
yes
no
$J_C = I_1$?
SET SUCCEEDING ACT. NUMBER OF ACT. A = A+1. (SAN_A = A+1).
FINDI
SEARCH KEY = J_C
yes
NEWI
$I_A = I_{A+1}$?
no
SETEND
$SAN_A = 7777_8$ (END THREAD FLAG)

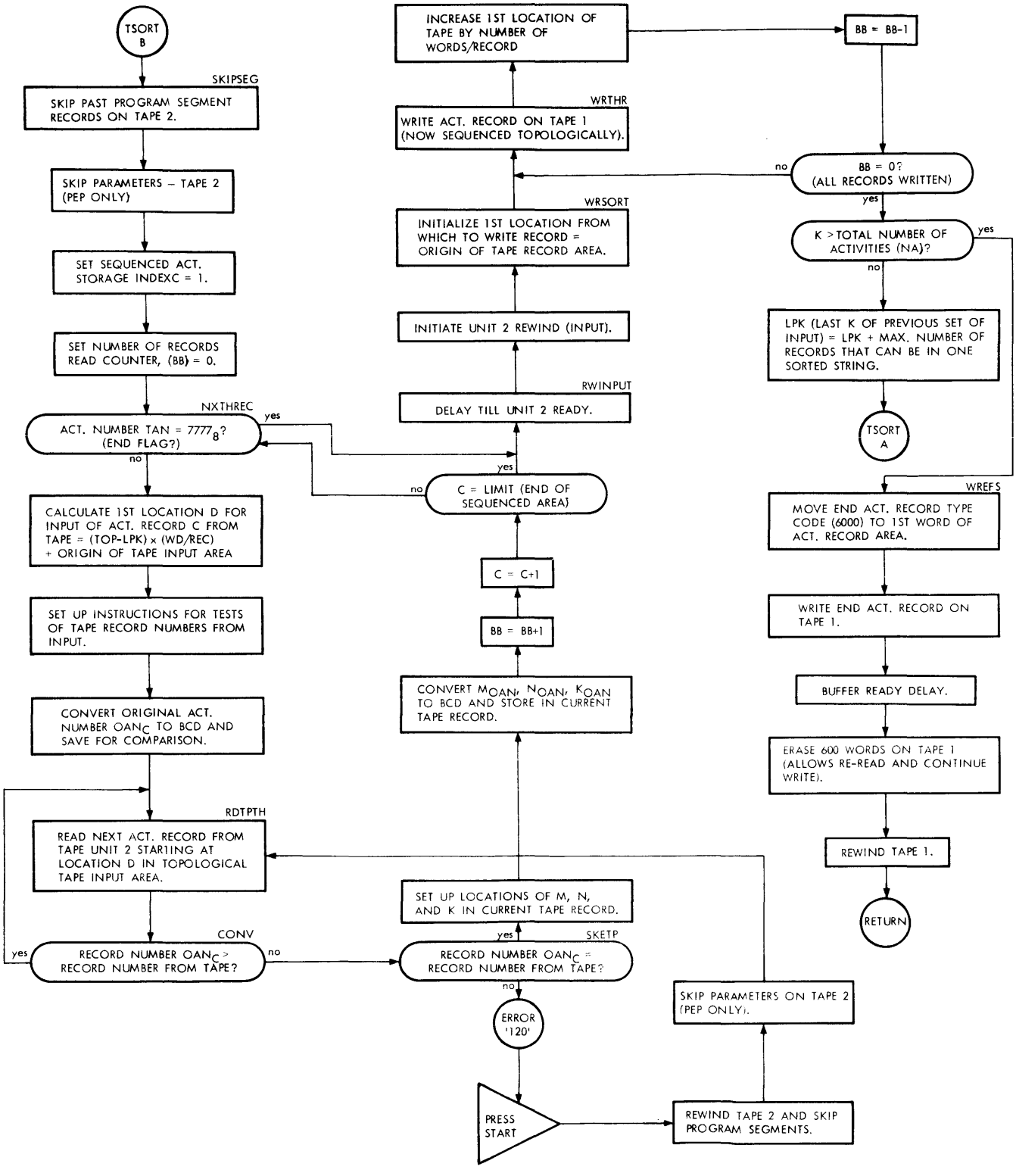

TOPSORT C
FST 1
B = B-1
yes
$J_C = I_B$?
no
B = B+1
TOPSORT D
TSTERM
TEST TERMINAL THREAD FLAG STORED IN SIGN BIT OF J_B.
ON
DO NOT USE ACTIVITY B
OFF
yes
$SAN_B \neq 0$? (IS ACT. B ALREADY IN CURRENT THREAD?)
no
A = B? (IS ACT. B ALREADY LAST ITEM IN CURRENT THREAD?)
yes
no
SAN_A = B (PUT ACT. B AT END OF THREAD)
A = B
C = SAN_C
TOPSORT B
TSTSUB
SET THREAD CLOSE-UP INDEX (D) = START OF CURRENT THREAD INDEX (F).
yes
LOOP TYPE-OUT COUNTER = -1 ?
no
ERROR 206 TYPE-OUT AND HALT
PRESS START
ERROR 205 TYPE-OUT AND CONTINUE.
E = SAN_E
SET LOOP TYPE-OUT COUNTER = 20
CONER
SAN_D = SAN_E (CLOSE THREAD TO SKIP ALL ACTIVITIES WITH I = I OF THE ACTIVITY TO BE MOVED TO END OF THREAD)
TO TOPSORT E
D = SAN_D
yes
SAN_D = C? (IS ACT. C IN THREAD AHEAD OF ACT. B?)
no
no
SAN_D = B?
yes
PRECEDING ACTIVITY FOUND
REDUCE LOOP TYPE-OUT COUNTER BY 1
SET I_E NEGATIVE SO CAN RECOGNIZE ON TYPE-OUT AS BEING REACHED TWICE IN THREAD.
TYPLP
MOVE ALL 3 WORDS OF ABBREVIATED ACT. RECORD E TO ERROR TYPE-OUT AREA.
LPER
SET LOOP TYPE-OUT COUNTER = 20
(LOOP IN NETWORK)
yes
IS LOOP DETECTION COUNTER > TOTAL NUMBER OF ACTIVITIES?
no
INCREASE LOOP DETECTION COUNTER BY 1.
no
yes
SAN_E = 0? (IS ACTIVITY E ALREADY AT END OF CURRENT THREAD?)
E = E+1
yes
$I_{E+1} = I_E$?
no
CLTSTI
SET SAME I THREAD CLOSE-UP INDEX (E) = B

M AND N ASSIGNMENT SUBROUTINE

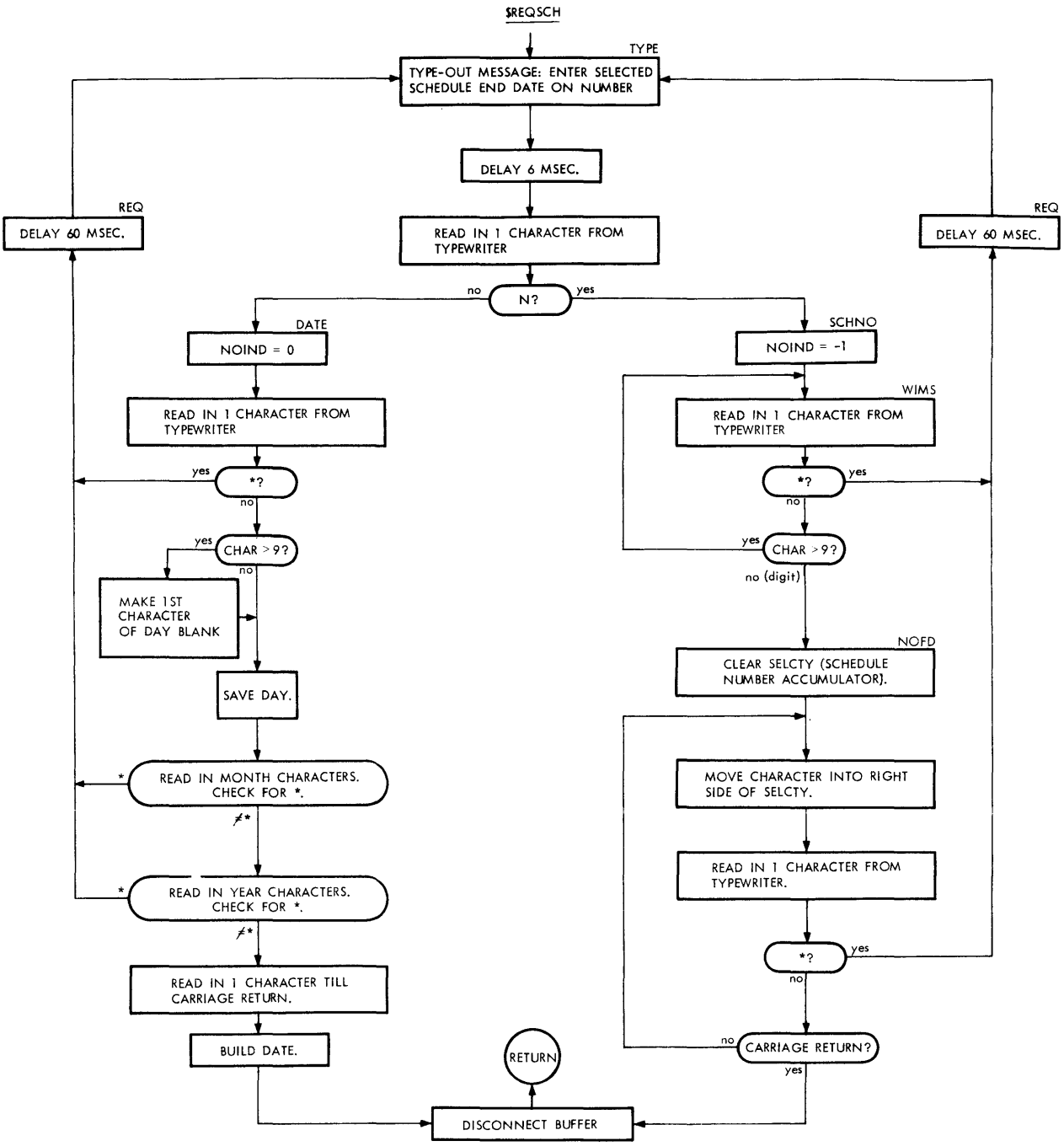



MN
A
MOVE MN
SET F INDEX = 1 (FOR MOVE TO J AREA).
SET G INDEX = ACT. H (NUMBER OF 1ST ACTIVITY IN THREAD) FOR MOVE FROM I AREA.
NEXT MNJ
$MN_F = MN_G$ (MOVE MN FROM I AREA TO J AREA).
F = F+1
G = NUMBER OF ACTIVITY SUCCEEDING ACTIVITY G (SANG).
no
G = 0? (END OF THREAD)
yes
MOVE MNI
SET L INDEX = 1 FOR MOVE FROM J AREA BACK TO I AREA.
NEXT MNI
MOVE MN_L FROM J AREA BACK TO I AREA IN TOPOLOGICAL SEQUENCE
L = L+1
no
L = LIMIT? (ALL MN MOVED?)
yes
RETURN


$SCAP20
SET 1ST K OF CURRENT SET OF INPUT (LPK) = 1.
SET TOPOLOGICAL SEQUENCE ACT. NUMBER, (K) = 1.
SET NEXT ACT. NUMBER (AA) = ORIGINAL ACT. NUMBER OF ORIGIN OF TOPOLOGICAL THREAD.
DELAY TILL TAPE 2 READY.
REWIND TAPE 2.
TSORT A
NXFIFTY
SET SELECTED ACT. STORAGE INDEX, (B) = 1 FOR MOVE TO SELECTED AREA.
MOVESEL
AA = 0? (END OF TOP. THREAD)
yes
WORD B = 37777777 8 AND B = B+1
no
SELECTED AREA FULL
yes
no
MOVESELA
SET LEFT 12 BITS OF WORD B = UNSORTED ACT. NUMBER OF ACT. AA, (UAN AA).
SET RIGHT 12 BITS OF WORD B = K (TOPOLOGICALLY SEQUENCED ACT. NUMBER, (TAN)).
SET AA = SUCCEEDING ACT. NUMBER OF ACT. AA, (SAN AA).
K = K+1
B = B+1
B = LIMIT? (SELECTED ACT. STORAGE AREA FILLED?)
no
yes
TSORT B
yes
C = LIMIT? (SEQUENCED AREA FULL)
no
C = C+1
OAN CSX = 7777 8 PREVENTS BEING SELECTED AGAIN AS SMALLEST ACT. NUMBER
MOVESEQ
OAN C = OAN CSX
TAN C = TAN CSX
(MOVE SMALLEST ACT. NUMBER TO SEQUENCED AREA.)
yes
B = LIMIT? (EXAMINED ENTIRE SELECTED AREA)
no
B = B+1
CSA = OAN B
(NEW SMALLEST ACT. NUMBER.)
FNDSM
SET CURRENT SMALLEST ACT. INDEX, CSX = B.
(NEW SMALLEST FOUND)
no
TSTSM
OAN B > CSA?
yes
SET CURRENT SMALLEST ACT. NUMBER (CSA) = 37777777 8.
NXSQ
SET SELECTED ACT. STORAGE INDEX B = 1 FOR MOVE FROM SEQUENCED AREA TO SELECTED AREA.
SQSLAC
SET SEQUENCED ACT. STORAGE INDEX C = 1 FOR MOVE TO SEQUENCED AREA.


TSORT B
SKIPSEG
SKIP PAST PROGRAM SEGMENT RECORDS ON TAPE 2.
SKIP PARAMETERS – TAPE 2 (PEP ONLY)
SET SEQUENCED ACT. STORAGE INDEXC = 1.
SET NUMBER OF RECORDS READ COUNTER, (BB) = 0.
NXTHREC
ACT. NUMBER TAN = 7777$_8$? (END FLAG?)
yes
no
CALCULATE 1ST LOCATION D FOR INPUT OF ACT. RECORD C FROM TAPE = (TOP-LPK) x (WD/REC) + ORIGIN OF TAPE INPUT AREA
SET UP INSTRUCTIONS FOR TESTS OF TAPE RECORD NUMBERS FROM INPUT.
CONVERT ORIGINAL ACT. NUMBER OAN$_C$ TO BCD AND SAVE FOR COMPARISON.
RDTPTH
READ NEXT ACT. RECORD FROM TAPE UNIT 2 STARTING AT LOCATION D IN TOPOLOGICAL TAPE INPUT AREA.
CONV
yes
RECORD NUMBER OAN$_C$ > RECORD NUMBER FROM TAPE?
no
SET UP LOCATIONS OF M, N, AND K IN CURRENT TAPE RECORD.
SKETP
yes
RECORD NUMBER OAN$_C$ = RECORD NUMBER FROM TAPE?
no
ERROR '120'
PRESS START
REWIND TAPE 2 AND SKIP PROGRAM SEGMENTS.
SKIP PARAMETERS ON TAPE 2 (PEP ONLY).
CONVERT M$_{OAN}$, N$_{OAN}$, K$_{OAN}$ TO BCD AND STORE IN CURRENT TAPE RECORD.
BB = BB+1
C = C+1
C = LIMIT (END OF SEQUENCED AREA)
no
yes
RWINPUT
DELAY TILL UNIT 2 READY.
INITIATE UNIT 2 REWIND (INPUT).
WRSORT
INITIALIZE 1ST LOCATION FROM WHICH TO WRITE RECORD = ORIGIN OF TAPE RECORD AREA.
WRTHR
WRITE ACT. RECORD ON TAPE 1 (NOW SEQUENCED TOPOLOGICALLY).
INCREASE 1ST LOCATION OF TAPE BY NUMBER OF WORDS/RECORD
BB = BB-1
no
BB = 0? (ALL RECORDS WRITTEN)
yes
K > TOTAL NUMBER OF ACTIVITIES (NA)?
yes
no
LPK (LAST K OF PREVIOUS SET OF INPUT) = LPK + MAX. NUMBER OF RECORDS THAT CAN BE IN ONE SORTED STRING.
TSORT A
WREFS
MOVE END ACT. RECORD TYPE CODE (6000) TO 1ST WORD OF ACT. RECORD AREA.
WRITE END ACT. RECORD ON TAPE 1.
BUFFER READY DELAY.
ERASE 600 WORDS ON TAPE 1 (ALLOWS RE-READ AND CONTINUE WRITE).
REWIND TAPE 1.
RETURN

SCHEDULE REQUEST SUBROUTINE



DATE COMPARISON SUBROUTINE

$DATCOM

THIS ROUTINE COMPARES A DATE STORED IN DUR1, DUR2, AND DUR3 WITH A CHECK DATE STORED IN CKDUR1, CKDUR2, CKDUR3.

IF THE DATE IN THE DUR LOCATIONS OCCURS AFTER THE DATE IN THE CKDUR REGION, THE ROUTINE FORWARDS TO ENTRY+2; OTHERWISE, IT FORWARDS TO ENTRY+1.

RETURN

SEGMENT 1 DATA ROUTINE

$SEG2ENT

BRANCH TO $SEG1CON

NOTE: THIS ROUTINE CONTAINS THE TRANSFER ADDRESS FOR ENTERING THE PROGRESS EVALUATION PROGRAM. IT IS ENTERED BY THE UNIVERSAL LOADER.

ALSO CONTAINED IN THIS ROUTINE ARE EQUIVALENCE INSTRUCTIONS DEFINING THE ARRANGEMENT OF THE PARAMETER REGION DURING SEGMENT 1.

UTILITY PACKAGE

$ACTOUT2

SUBR MTWSR

WRITE ACTIVITY STATUS RECORD ON TAPE 2 (BCD) FROM OUTPUT AREA.

RETURN

$DATBRK

ENTERS WITH DATE IN COMPACT FORM IN A AND B REGISTERS

DDMM (A REG) MMYY (B REG)

BREAKS DATE UP AS FOLLOWS:
BBDD IN DUR1 AND A REGISTER.
BMMM IN DUR2 AND B REGISTER.
BBYY IN DUR3.

RETURN

$BUILDL

MERGE RIGHTMOST 11 BITS OF A REGISTER INTO BITS 1-11 OF B REGISTER.
MOVE B REGISTER TO A REGISTER.

RETURN

$T1DLAY

DELAY TILL BUFFER W AND TAPE 1 READY.

RETURN

$ACTOUT1

SUBR MTWSR

WRITE ACTIVITY STATUS RECORD ON TAPE 1 (BCD) FROM OUTPUT AREA

RETURN

$BCDCON

CONVERT BCD 4-DIGIT NUMBER IN A REGISTER TO BINARY.
PLACE RESULT IN A REGISTER.

RETURN

$BRKUP

REMOVE BIT ZERO OF A REG.

BREAK WORD IN A REGISTER INTO 2 PARTS:
BITS 0-11 TO A REGISTER (RIGHT JUSTIFY)
BITS 12-23 TO B REGISTER (RIGHT JUSTIFY)

RETURN

$SAVET1

TYPE MESSAGE: REPLACE TAPE 1 IF TAPE SAVE REQUIRED.

HLT

PRESS START

RETURN

$SETT1

REWIND TAPES 1 AND 2

ERASE 1SO WORDS ON TAPE 1.

RETURN

$BUILDL1

MERGE RIGHTMOST 12 BITS OF A REGISTER INTO BITS 0-11 OF B REGISTER.
MOVE B REGISTER TO A REGISTER.

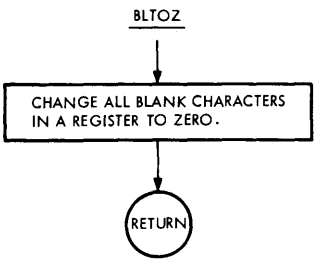
RETURN

T2DLAY

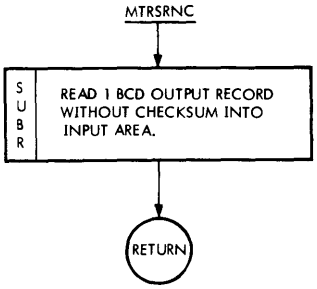
DELAY TILL BUFFER W AND TAPE 2 READY

RETURN

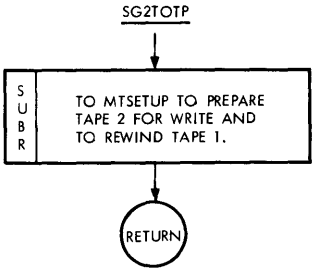
BLANKS TO ZERO SUBROUTINE



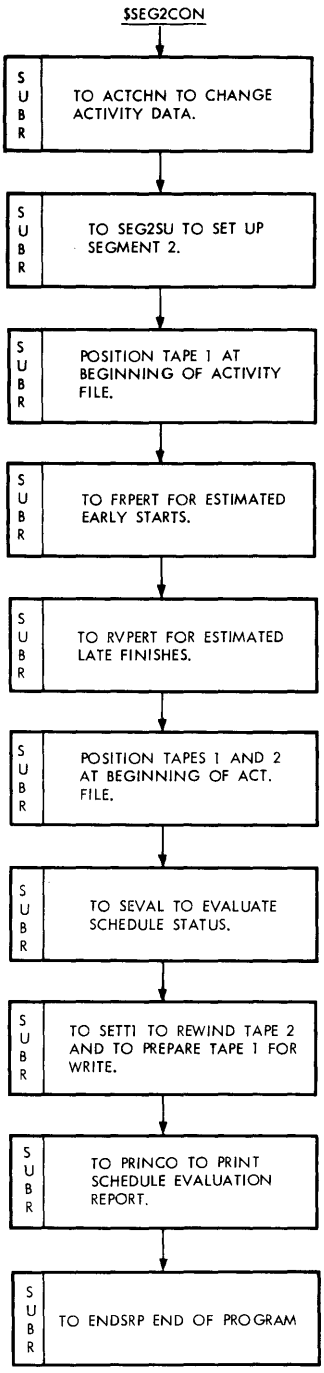
READ BCD RECORD SUBROUTINE



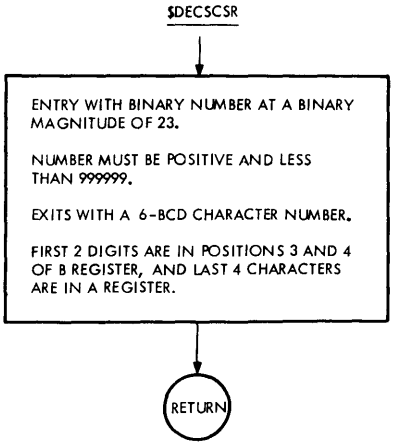
SEGMENT 2 TAPE PREP.



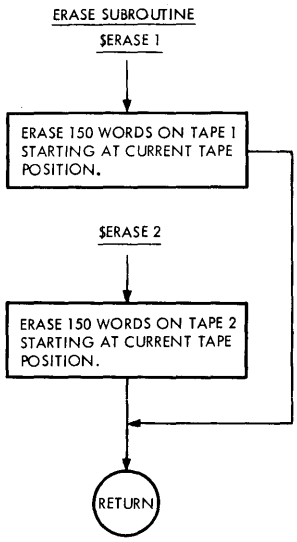
SEGMENT 2 CONTROL



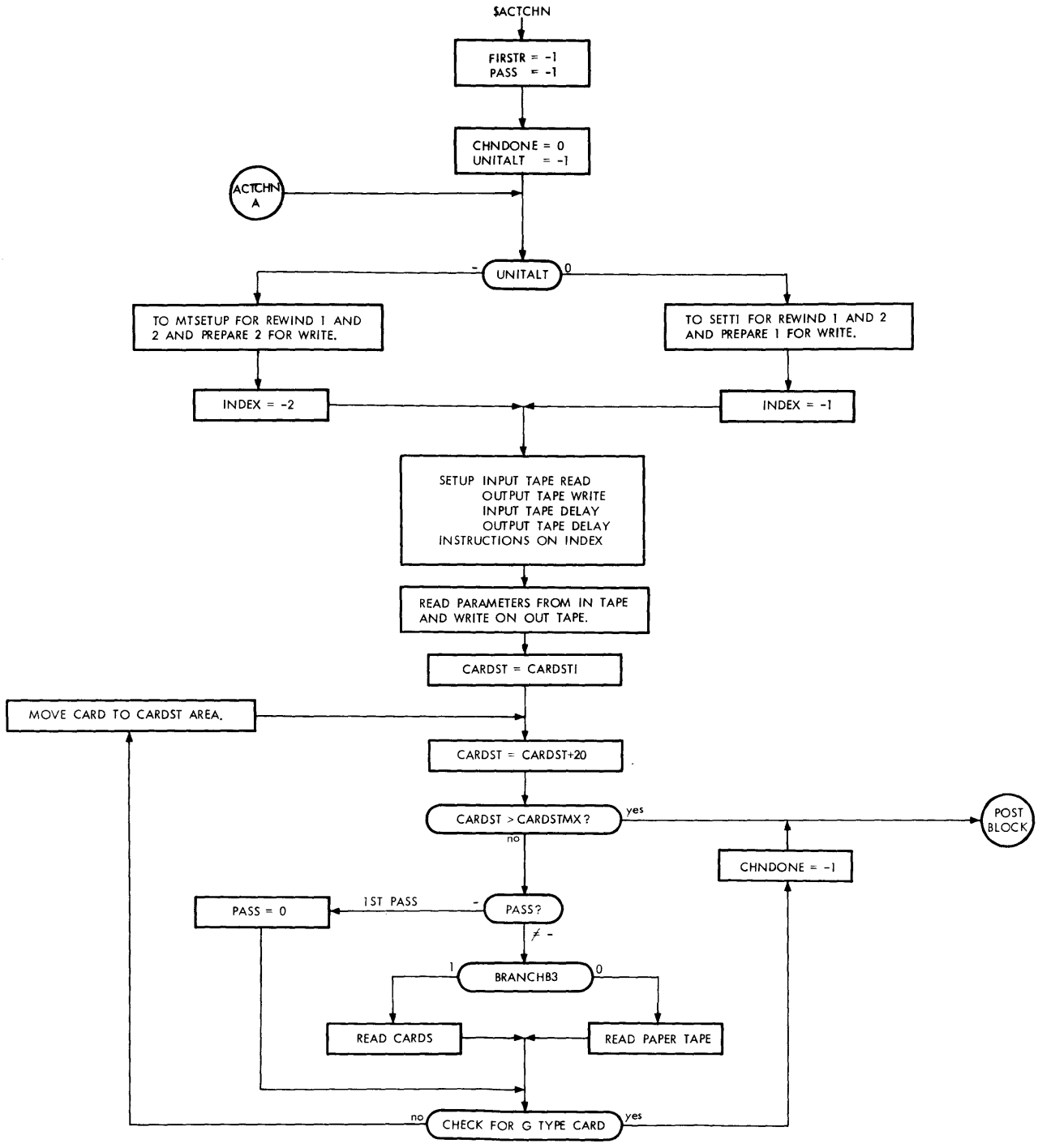
BINARY TO DECIMAL INTEGER CONVERSION SUBROUTINE

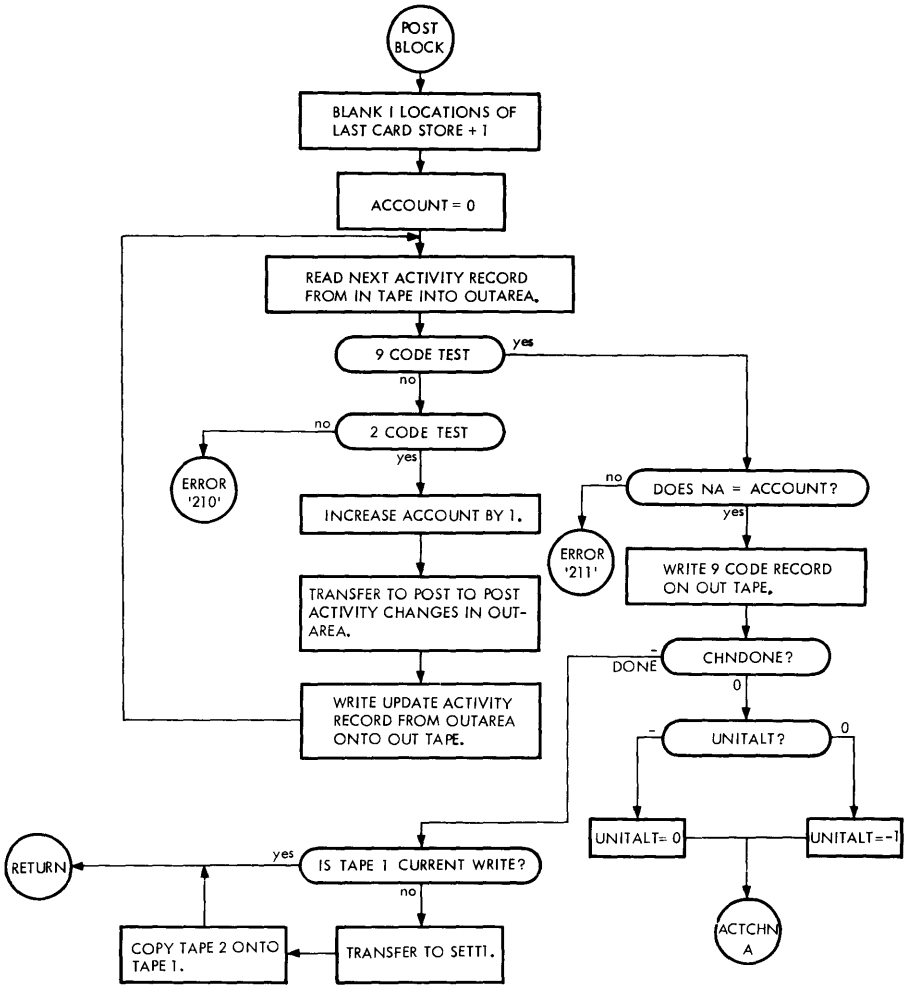


ERASE SUBROUTINE

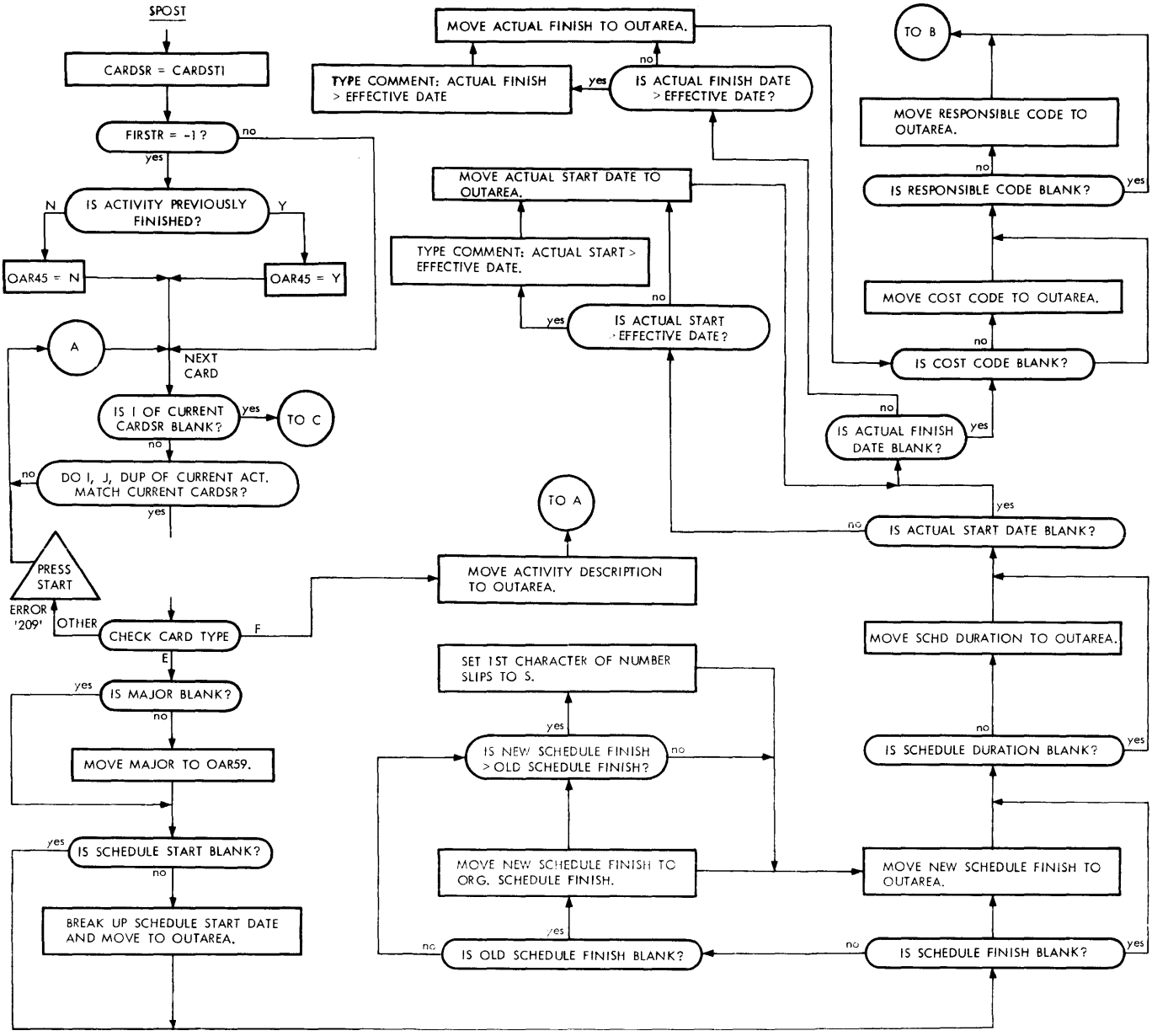


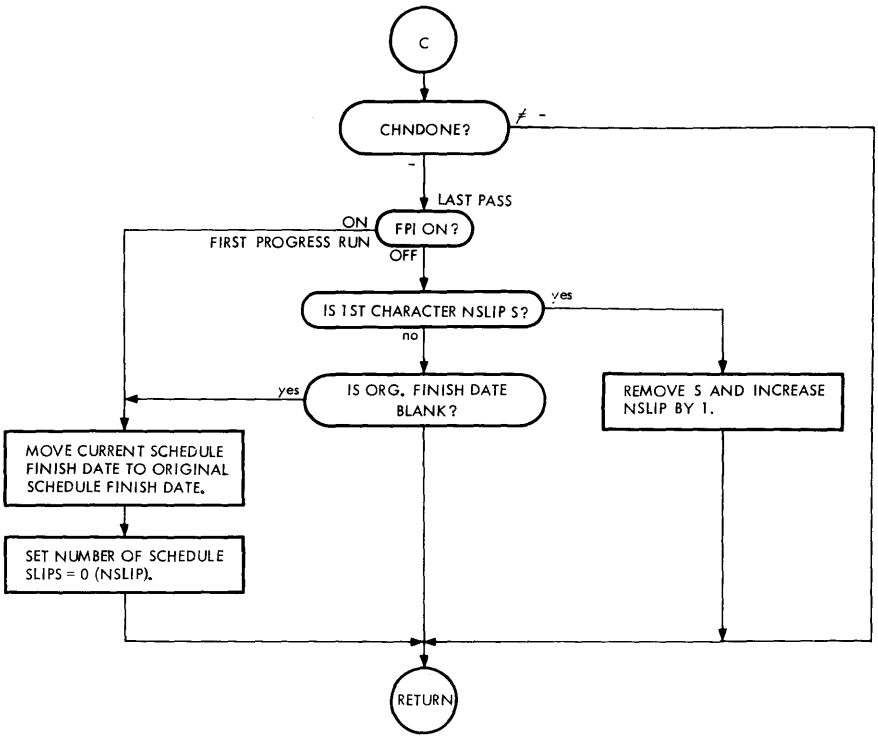
ACTIVITY CHANGE ROUTINE

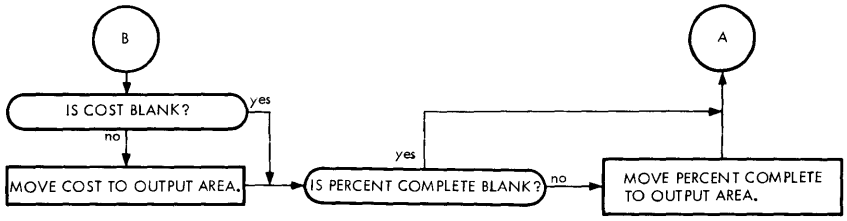


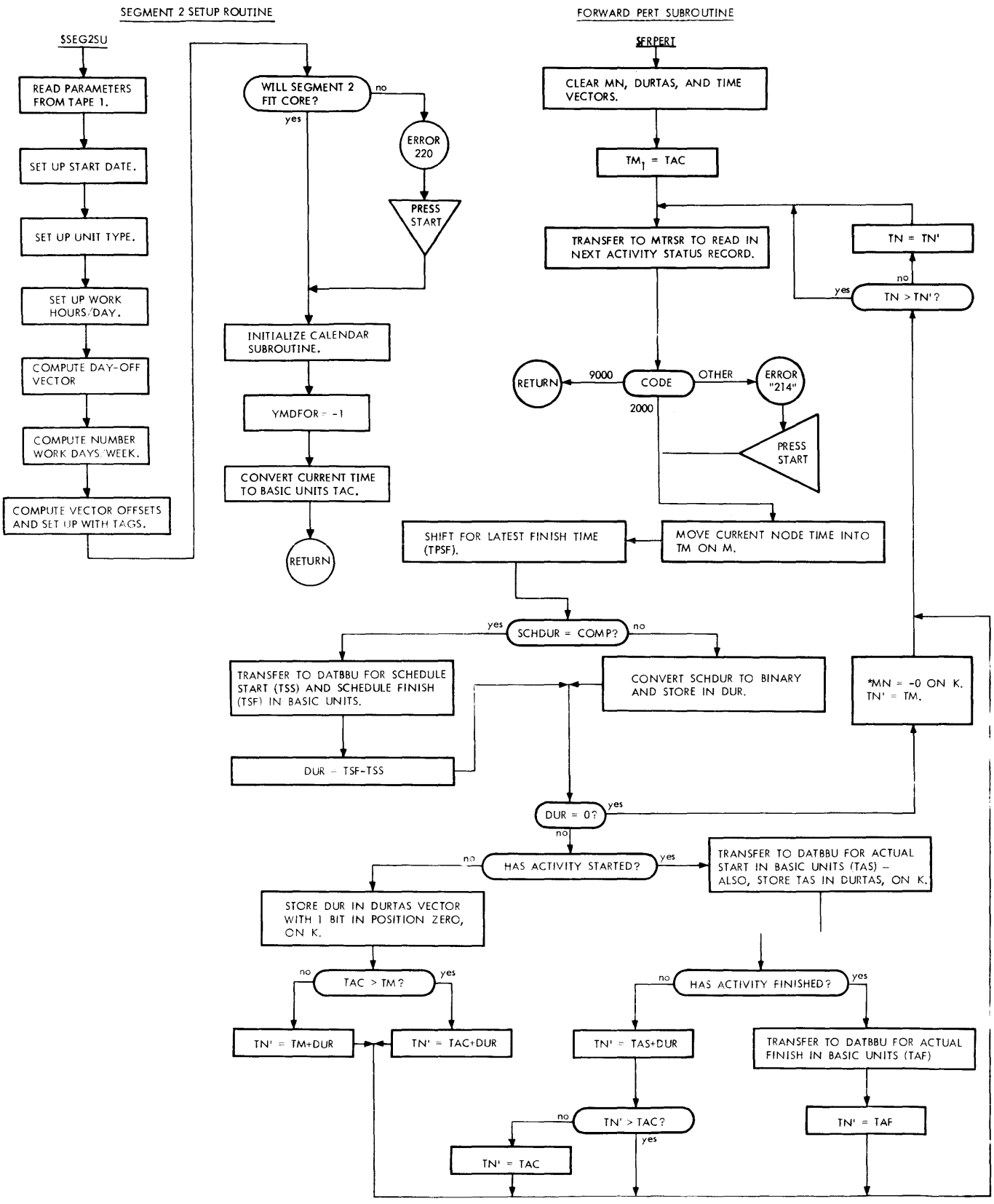

POST BLOCK
BLANK 1 LOCATIONS OF LAST CARD STORE + 1
ACCOUNT = 0
READ NEXT ACTIVITY RECORD FROM IN TAPE INTO OUTAREA.
9 CODE TEST
yes
no
2 CODE TEST
no
yes
ERROR '210'
INCREASE ACCOUNT BY 1.
TRANSFER TO POST TO POST ACTIVITY CHANGES IN OUT-AREA.
WRITE UPDATE ACTIVITY RECORD FROM OUTAREA ONTO OUT TAPE.
DOES NA = ACCOUNT?
no
yes
ERROR '211'
WRITE 9 CODE RECORD ON OUT TAPE.
CHNDONE?
DONE
0
UNITALT?
-
0
UNITALT= 0
UNITALT=-1
ACTCHN A
IS TAPE 1 CURRENT WRITE?
yes
no
RETURN
COPY TAPE 2 ONTO TAPE 1.
TRANSFER TO SETT1.

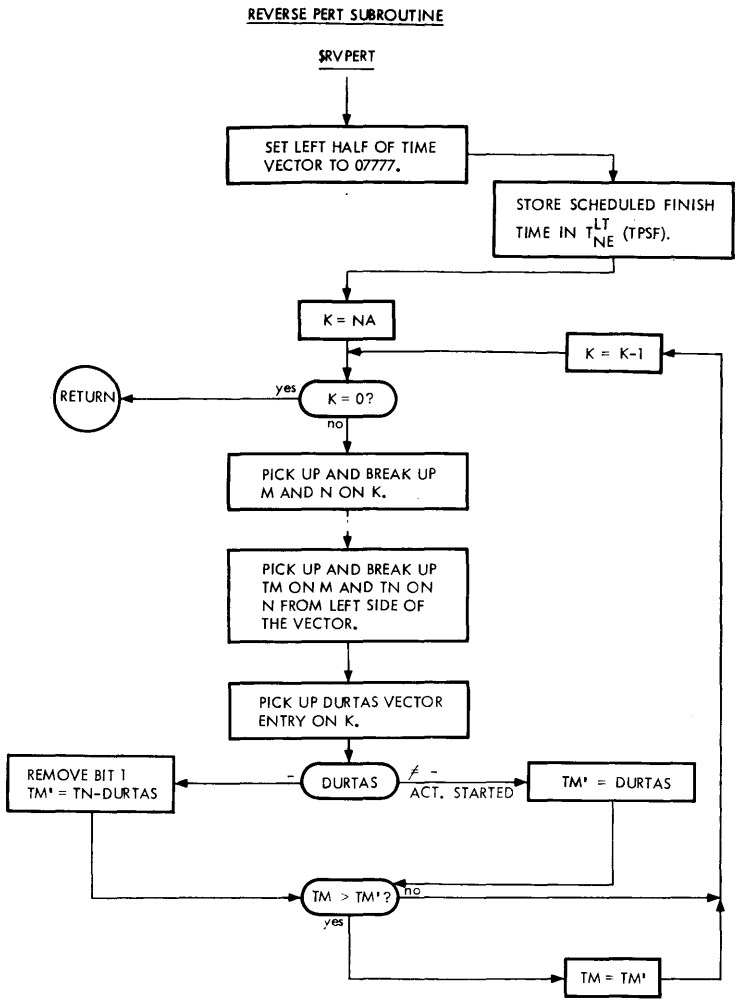
POST CHANGE SUBROUTINE

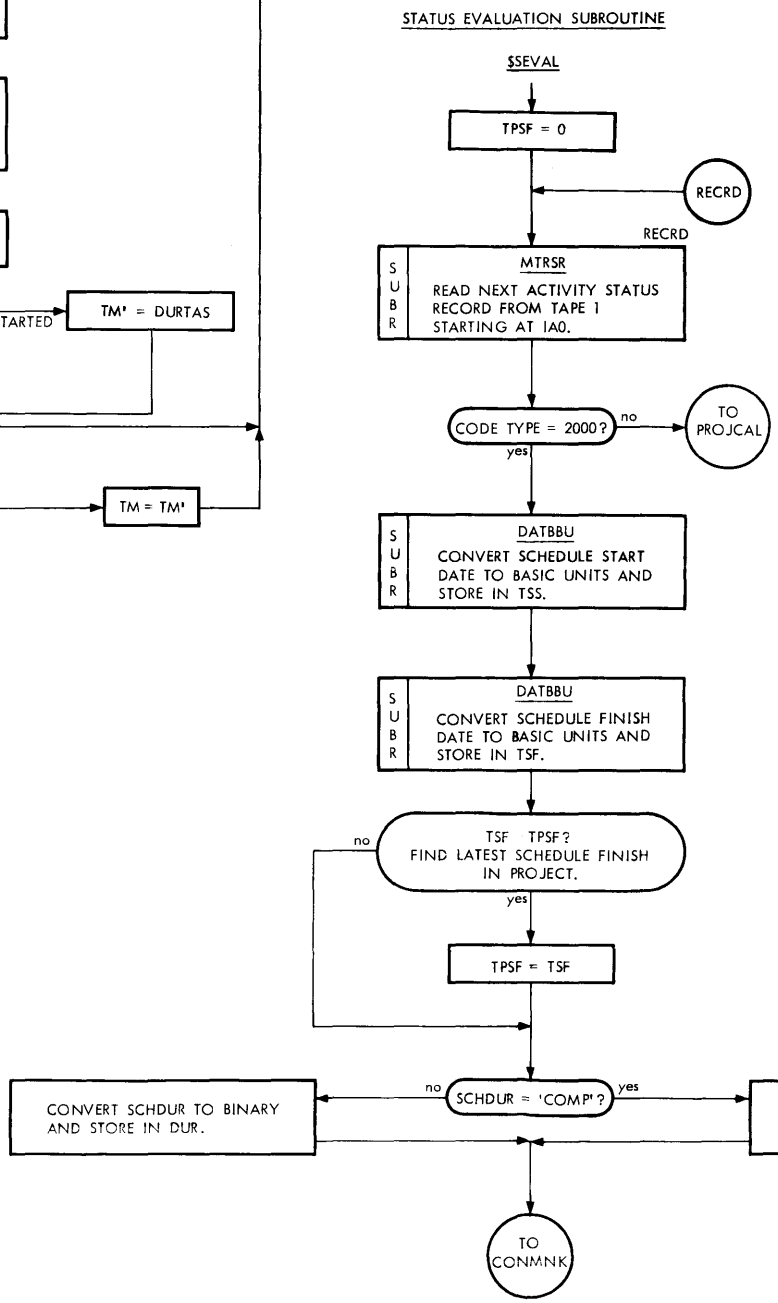


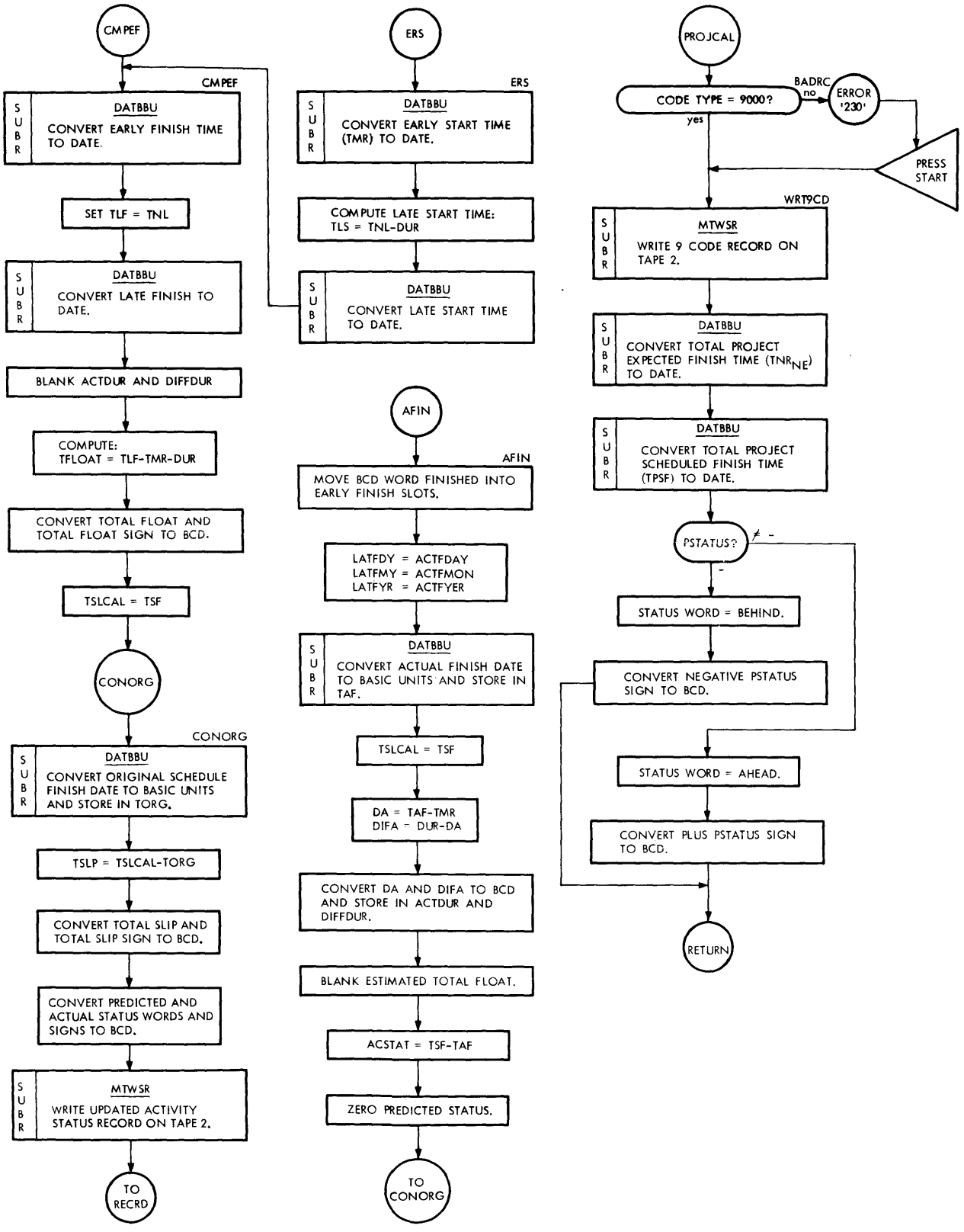

C
CHNDONE?
≠ -
-
LAST PASS
ON
FIRST PROGRESS RUN
FPI ON?
OFF
IS 1ST CHARACTER NSLIP S?
yes
no
yes
IS ORG. FINISH DATE BLANK?
REMOVE S AND INCREASE NSLIP BY 1.
MOVE CURRENT SCHEDULE FINISH DATE TO ORIGINAL SCHEDULE FINISH DATE.
SET NUMBER OF SCHEDULE SLIPS = 0 (NSLIP).
RETURN

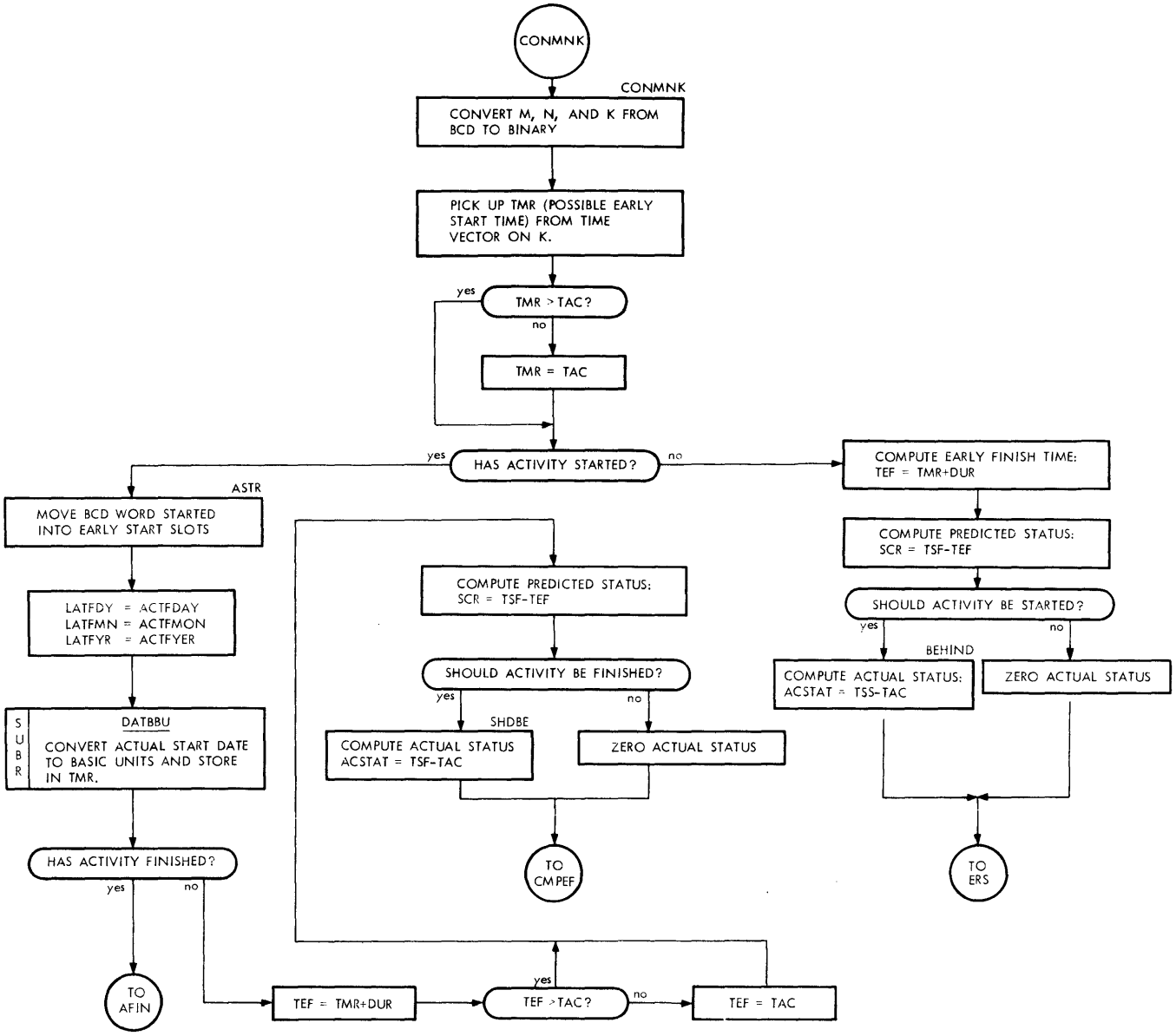

B
A
IS COST BLANK?
yes
no
yes
MOVE COST TO OUTPUT AREA.
IS PERCENT COMPLETE BLANK?
no
MOVE PERCENT COMPLETE TO OUTPUT AREA.


SEGMENT 2 SETUP ROUTINE
$SEG2SU
READ PARAMETERS FROM TAPE 1.
SET UP START DATE.
SET UP UNIT TYPE.
SET UP WORK HOURS/DAY.
COMPUTE DAY-OFF VECTOR
COMPUTE NUMBER WORK DAYS/WEEK.
COMPUTE VECTOR OFFSETS AND SET UP WITH TAGS.
WILL SEGMENT 2 FIT CORE?
no
yes
ERROR 220
PRESS START
INITIALIZE CALENDAR SUBROUTINE.
YMDFOR = -1
CONVERT CURRENT TIME TO BASIC UNITS TAC.
RETURN
FORWARD PERT SUBROUTINE
$FRPERT
CLEAR MN, DURTAS, AND TIME VECTORS.
TM1 = TAC
TRANSFER TO MTRSR TO READ IN NEXT ACTIVITY STATUS RECORD.
TN = TN'
no
yes
TN > TN'?
RETURN
9000
CODE
OTHER
ERROR "214"
2000
PRESS START
SHIFT FOR LATEST FINISH TIME (TPSF).
MOVE CURRENT NODE TIME INTO TM ON M.
yes
SCHDUR = COMP?
no
TRANSFER TO DATBBU FOR SCHEDULE START (TSS) AND SCHEDULE FINISH (TSF) IN BASIC UNITS.
CONVERT SCHDUR TO BINARY AND STORE IN DUR.
*MN = -0 ON K.
TN' = TM.
DUR = TSF-TSS
DUR = 0?
yes
no
no
HAS ACTIVITY STARTED?
yes
TRANSFER TO DATBBU FOR ACTUAL START IN BASIC UNITS (TAS) – ALSO, STORE TAS IN DURTAS, ON K.
STORE DUR IN DURTAS VECTOR WITH 1 BIT IN POSITION ZERO, ON K.
no
TAC > TM?
yes
TN' = TM+DUR
TN' = TAC+DUR
no
HAS ACTIVITY FINISHED?
yes
TN' = TAS+DUR
TRANSFER TO DATBBU FOR ACTUAL FINISH IN BASIC UNITS (TAF)
no
TN' > TAC?
yes
TN' = TAC
TN' = TAF

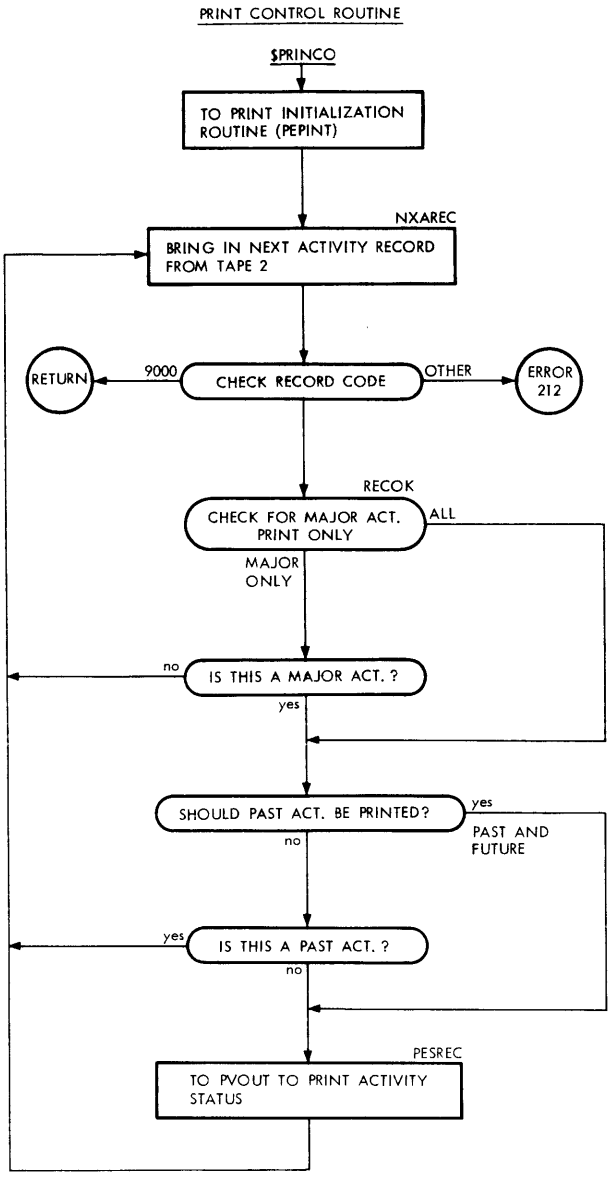

REVERSE PERT SUBROUTINE
$RVPERT
SET LEFT HALF OF TIME VECTOR TO 07777.
STORE SCHEDULED FINISH TIME IN T_{NE}^{LT} (TPSF).
K = NA
K = K-1
RETURN
yes
K = 0?
no
PICK UP AND BREAK UP M AND N ON K.
PICK UP AND BREAK UP TM ON M AND TN ON N FROM LEFT SIDE OF THE VECTOR.
PICK UP DURTAS VECTOR ENTRY ON K.
REMOVE BIT 1 TM' = TN-DURTAS
-
DURTAS
≠ -
ACT. STARTED
TM' = DURTAS
TM > TM'?
no
yes
TM = TM'


STATUS EVALUATION SUBROUTINE
$SEVAL
TPSF = 0
RECRD
RECRD
SUBR
MTRSR
READ NEXT ACTIVITY STATUS RECORD FROM TAPE 1 STARTING AT IA0.
CODE TYPE = 2000?
no
TO PROJCAL
yes
SUBR
DATBBU
CONVERT SCHEDULE START DATE TO BASIC UNITS AND STORE IN TSS.
SUBR
DATBBU
CONVERT SCHEDULE FINISH DATE TO BASIC UNITS AND STORE IN TSF.
no
TSF TPSF?
FIND LATEST SCHEDULE FINISH IN PROJECT.
yes
TPSF = TSF
CONVERT SCHDUR TO BINARY AND STORE IN DUR.
no
SCHDUR = 'COMP'?
yes
DUR = TSF-TSS
CONVERT DUR TO BCD AND STORE IN SCHDUR.
TO CONMNK

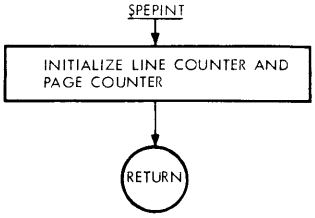

CMPEF
CMPEF
SUBR
DATBBU
CONVERT EARLY FINISH TIME TO DATE.
SET TLF = TNL
SUBR
DATBBU
CONVERT LATE FINISH TO DATE.
BLANK ACTDUR AND DIFFDUR
COMPUTE:
TFLOAT = TLF-TMR-DUR
CONVERT TOTAL FLOAT AND TOTAL FLOAT SIGN TO BCD.
TSLCAL = TSF
CONORG
CONORG
SUBR
DATBBU
CONVERT ORIGINAL SCHEDULE FINISH DATE TO BASIC UNITS AND STORE IN TORG.
TSLP = TSLCAL-TORG
CONVERT TOTAL SLIP AND TOTAL SLIP SIGN TO BCD.
CONVERT PREDICTED AND ACTUAL STATUS WORDS AND SIGNS TO BCD.
SUBR
MTWSR
WRITE UPDATED ACTIVITY STATUS RECORD ON TAPE 2.
TO RECRD
ERS
ERS
SUBR
DATBBU
CONVERT EARLY START TIME (TMR) TO DATE.
COMPUTE LATE START TIME:
TLS = TNL-DUR
SUBR
DATBBU
CONVERT LATE START TIME TO DATE.
AFIN
AFIN
MOVE BCD WORD FINISHED INTO EARLY FINISH SLOTS.
LATFDY = ACTFDAY
LATFMY = ACTFMON
LATFYR = ACTFYER
SUBR
DATBBU
CONVERT ACTUAL FINISH DATE TO BASIC UNITS AND STORE IN TAF.
TSLCAL = TSF
DA = TAF-TMR
DIFA = DUR-DA
CONVERT DA AND DIFA TO BCD AND STORE IN ACTDUR AND DIFFDUR.
BLANK ESTIMATED TOTAL FLOAT.
ACSTAT = TSF-TAF
ZERO PREDICTED STATUS.
TO CONORG
PROJCAL
CODE TYPE = 9000?
BADRC
no
ERROR '230'
yes
PRESS START
WRT9CD
SUBR
MTWSR
WRITE 9 CODE RECORD ON TAPE 2.
SUBR
DATBBU
CONVERT TOTAL PROJECT EXPECTED FINISH TIME (TNR$_{NE}$) TO DATE.
SUBR
DATBBU
CONVERT TOTAL PROJECT SCHEDULED FINISH TIME (TPSF) TO DATE.
PSTATUS?
≠ -
-
STATUS WORD = BEHIND.
CONVERT NEGATIVE PSTATUS SIGN TO BCD.
STATUS WORD = AHEAD.
CONVERT PLUS PSTATUS SIGN TO BCD.
RETURN


CONMNK
CONMNK
CONVERT M, N, AND K FROM BCD TO BINARY
PICK UP TMR (POSSIBLE EARLY START TIME) FROM TIME VECTOR ON K.
yes
TMR > TAC?
no
TMR = TAC
yes
HAS ACTIVITY STARTED?
no
ASTR
MOVE BCD WORD STARTED INTO EARLY START SLOTS
LATFDY = ACTFDAY
LATFMN = ACTFMON
LATFYR = ACTFYER
SUBR
DATBBU
CONVERT ACTUAL START DATE TO BASIC UNITS AND STORE IN TMR.
HAS ACTIVITY FINISHED?
yes
no
TO AFIN
TEF = TMR+DUR
yes
TEF > TAC?
no
TEF = TAC
COMPUTE PREDICTED STATUS: SCR = TSF-TEF
SHOULD ACTIVITY BE FINISHED?
yes
no
SHDBE
COMPUTE ACTUAL STATUS ACSTAT = TSF-TAC
ZERO ACTUAL STATUS
TO CMPEF
COMPUTE EARLY FINISH TIME: TEF = TMR+DUR
COMPUTE PREDICTED STATUS: SCR = TSF-TEF
SHOULD ACTIVITY BE STARTED?
yes
no
BEHIND
COMPUTE ACTUAL STATUS: ACSTAT = TSS-TAC
ZERO ACTUAL STATUS
TO ERS

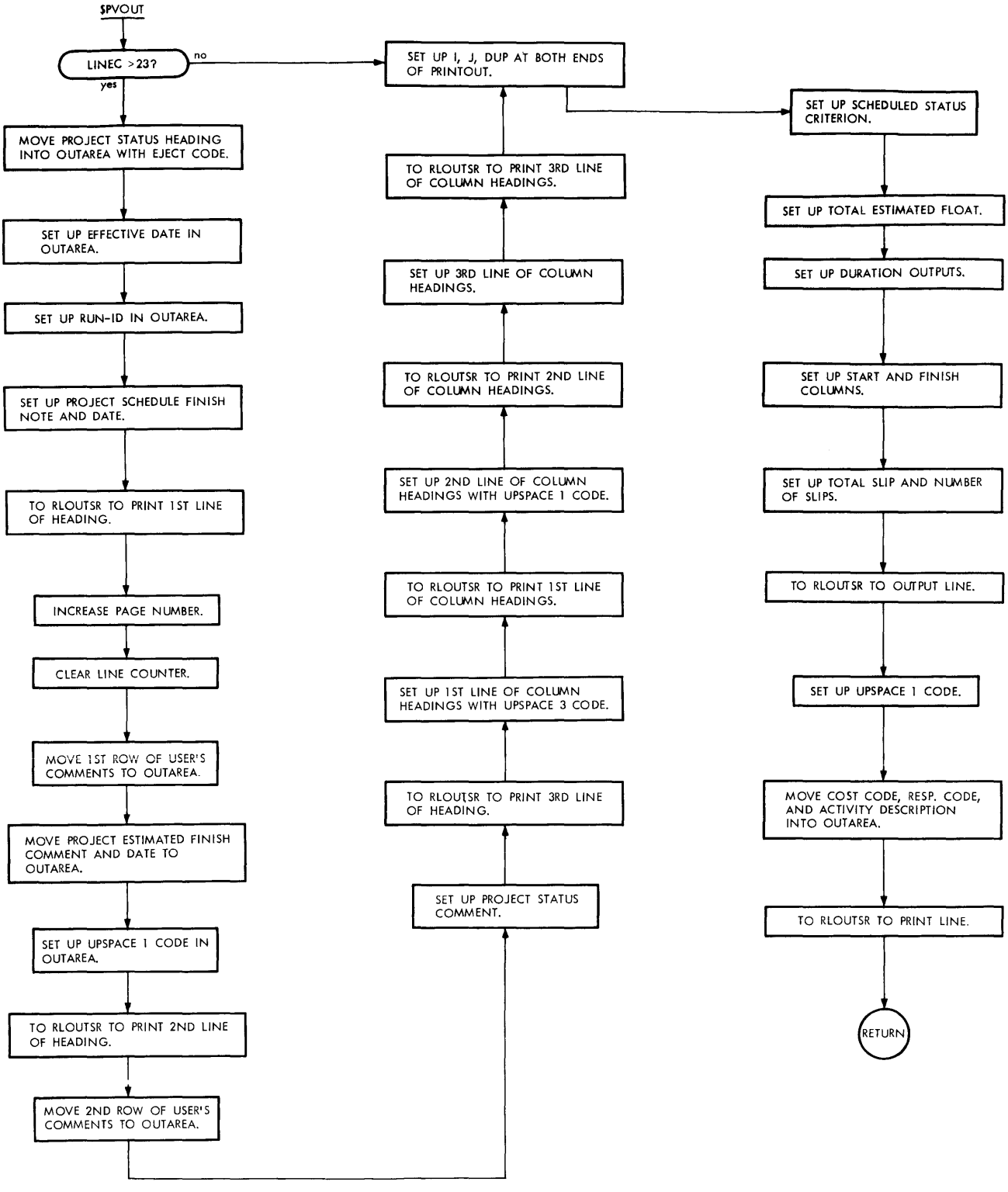
PRINT CONTROL ROUTINE



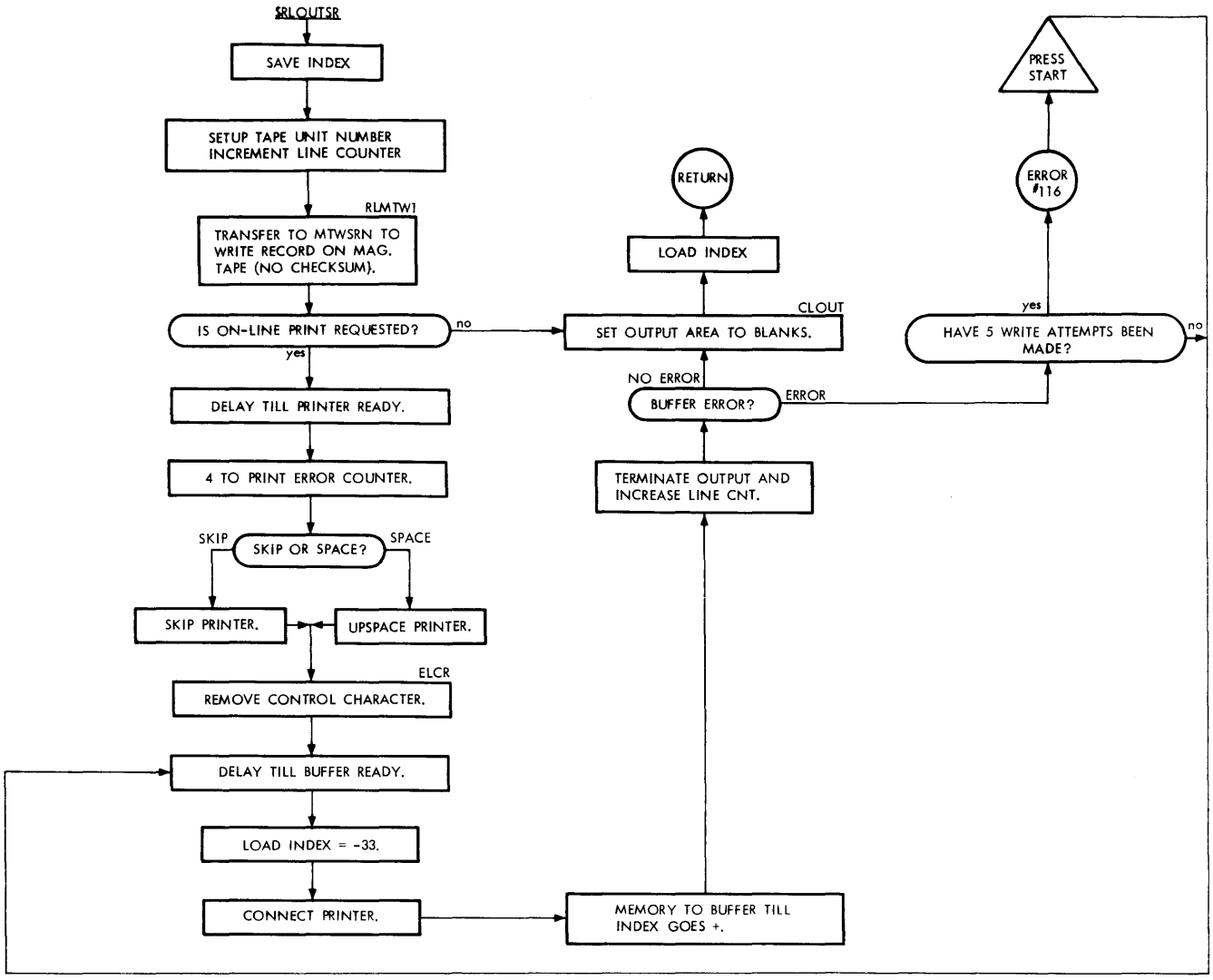
PRINT INITIALIZATION ROUTINE



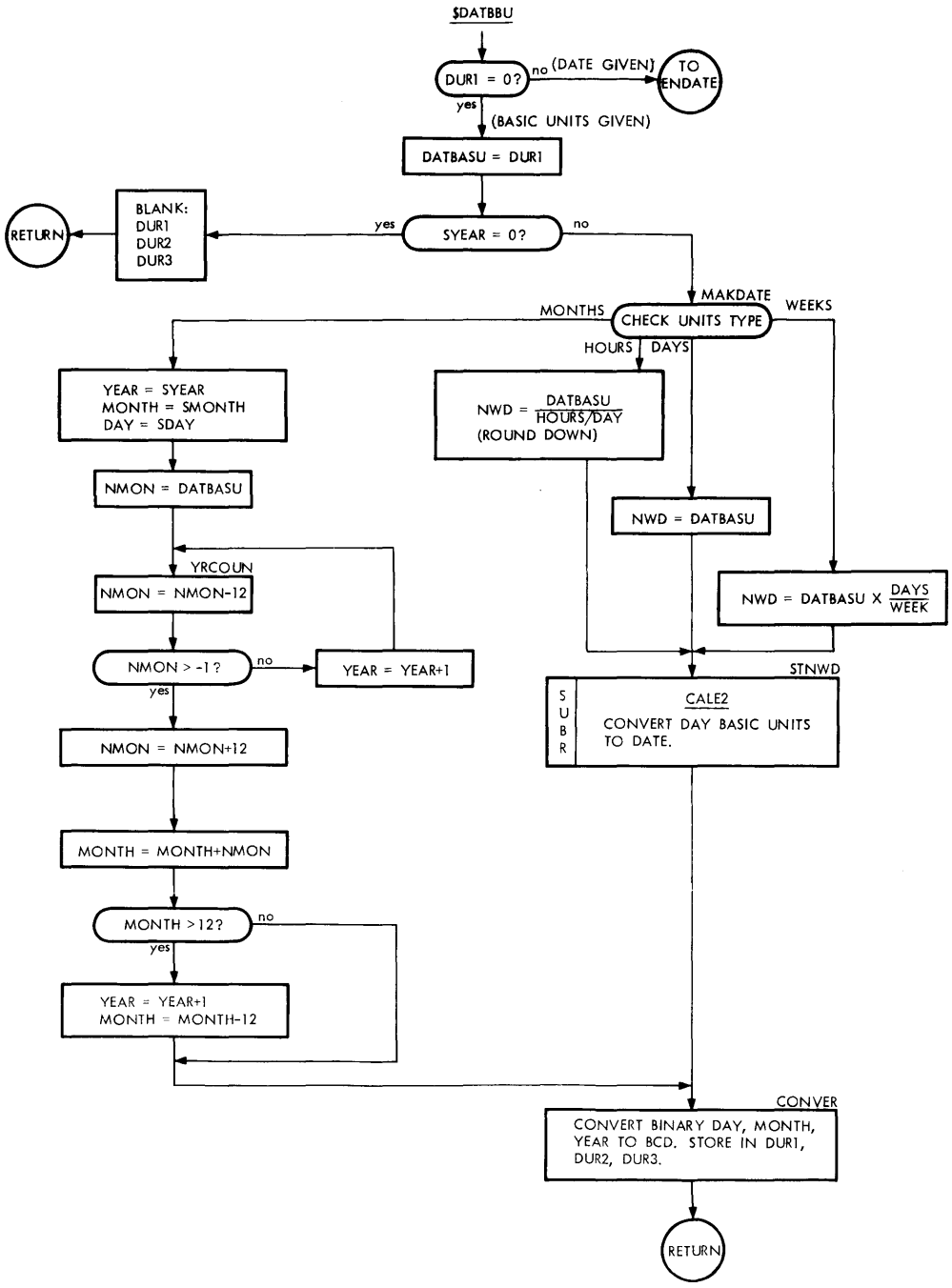
PRINT EVALUATION SUBROUTINE

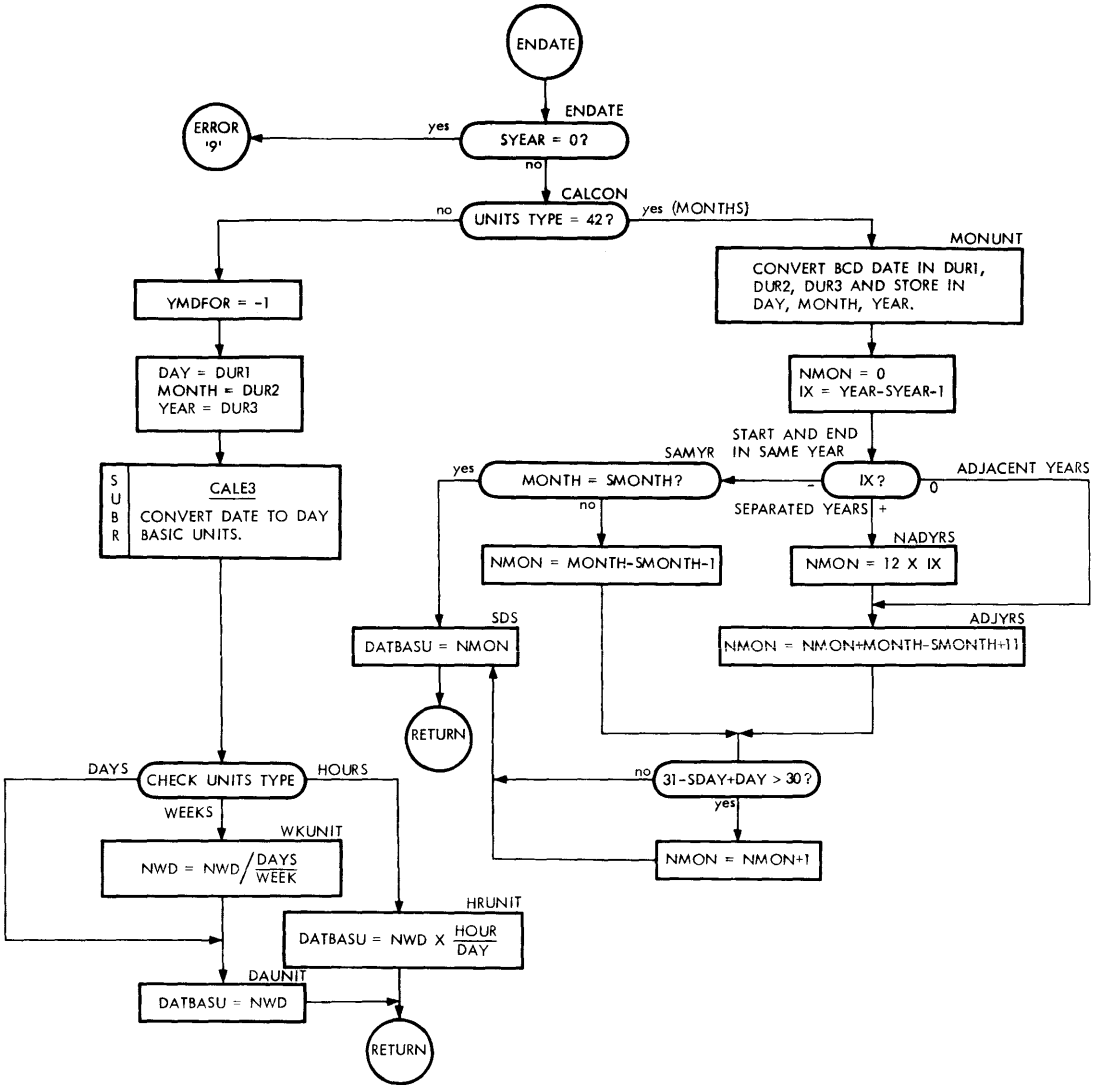


REPORT LINE OUT SUBROUTINE



DATE TO BASIC UNITS SUBROUTINE




ENDATE
ENDATE
SYEAR = 0?
yes
ERROR '9'
no
CALCON
UNITS TYPE = 42?
no
yes (MONTHS)
MONUNT
CONVERT BCD DATE IN DUR1, DUR2, DUR3 AND STORE IN DAY, MONTH, YEAR.
YMDFOR = -1
DAY = DUR1
MONTH = DUR2
YEAR = DUR3
NMON = 0
IX = YEAR-SYEAR-1
START AND END IN SAME YEAR
SAMYR
IX?
ADJACENT YEARS
0
SEPARATED YEARS
+
-
SUBR
CALE3
CONVERT DATE TO DAY BASIC UNITS.
yes
MONTH = SMONTH?
no
NMON = MONTH-SMONTH-1
NADYRS
NMON = 12 X IX
SDS
DATBASU = NMON
RETURN
ADJYRS
NMON = NMON+MONTH-SMONTH+11
DAYS
CHECK UNITS TYPE
HOURS
WEEKS
WKUNIT
NWD = NWD / DAYS/WEEK
no
31-SDAY+DAY > 30?
yes
NMON = NMON+1
HRUNIT
DATBASU = NWD X HOUR/DAY
DAUNIT
DATBASU = NWD
RETURN

RESTART SUBROUTINE

$RESTART

BRANCH TO MAINCON

PARAMETER READ SUBROUTINE

$PARMRD

READ PARAMETER FILE FROM TAPE 2. STORE EFFECTIVE DATE IN COMMON.

RETURN

BLANKS TO ZEROS SUBROUTINE

$BLTOZ

CHANGE ALL BLANK CHARACTERS IN A REGISTER TO ZEROS

RETURN

DATE PACK SUBROUTINE

$DATEPCK

PACK A 3-WORD DATE (D, M, Y) INTO ONE WORD.

RETURN

BINARY TO DECIMAL CONVERSION SUBROUTINE

$DECSCSR

ENTRY WITH A BINARY NUMBER AT A BINARY MAGNITUDE OF 23.
NUMBER MUST BE POSITIVE AND LESS THAN 999999.
EXITS WITH A 6-BCD-CHARACTER NUMBER. FIRST 2 DIGITS ARE IN POSITIONS 3 AND 4 OF B REGISTER, AND LAST 4 CHARACTERS ARE IN A REGISTER.

RETURN

DECIMAL TO BINARY CONVERSION SUBROUTINE

$DIB

SIMULATE SDS DECIMAL TO BINARY, SINGLE-PRECISION, FIXED-POINT PROGRAM OPERATOR.

RETURN

DECIMAL INTO BINARY CONTROL SUBROUTINE

$BCDCON

CONVERT BCD 4-DIGIT NUMBER IN A REGISTER TO BINARY. PLACE RESULT IN A REGISTER.

RETURN

INITIAL SUCCESSOR SUBROUTINE

$BLDST

CHECK FOR FIT IN CORE.
INITIALIZE S VECTOR.
S_i = L+1 FOR i = 0, NA.

RETURN

ORIGINAL SEQUENCE NUMBER SUBROUTINE

$ORGSEQ

MERGE THE ORIGINAL ACT. NUMBERS INTO THE RIGHT HALF OF THE S VECTOR.

RETURN

INITIALIZE PRINT SUBROUTINE

$PRINTIN

SET LINE COUNTER TO 50.
SET PAGE COUNTER TO 1.
REWIND 1 AND ERASE A GAP.

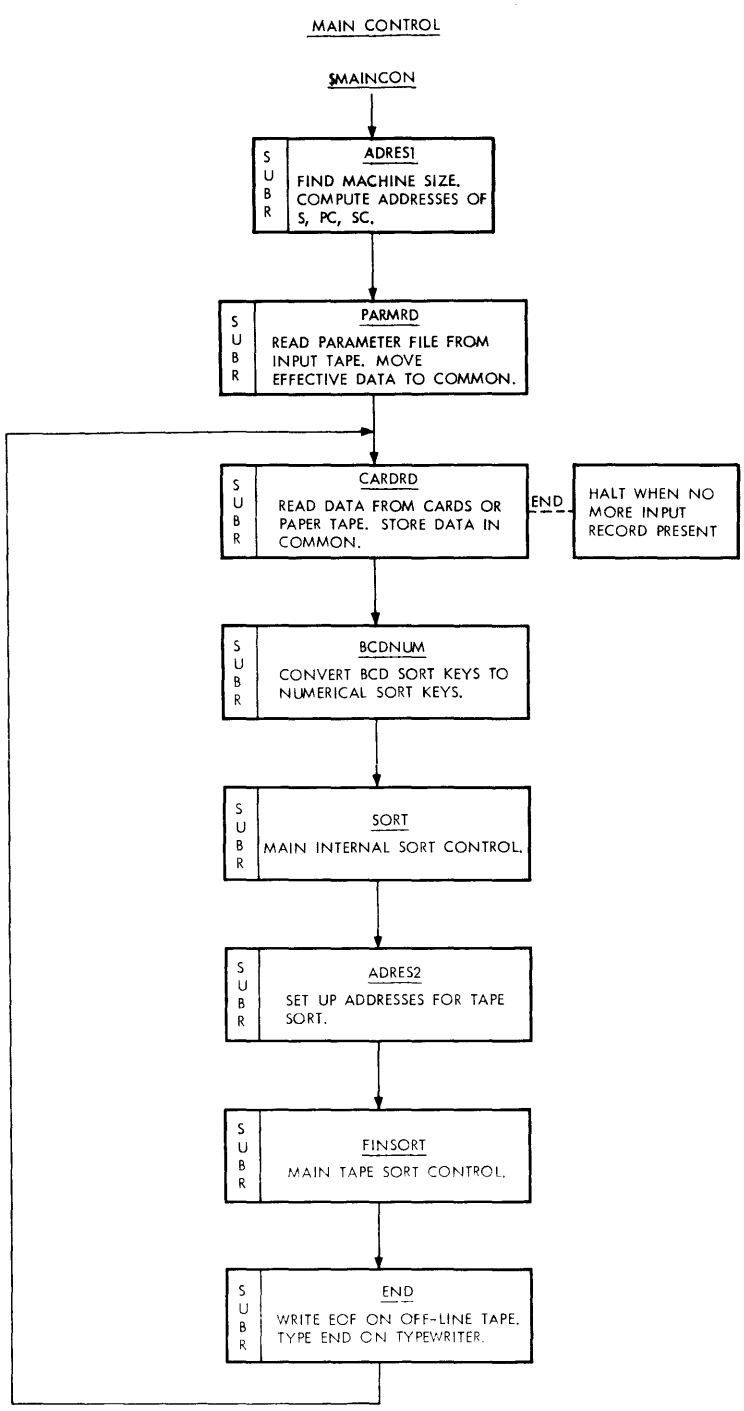
RETURN

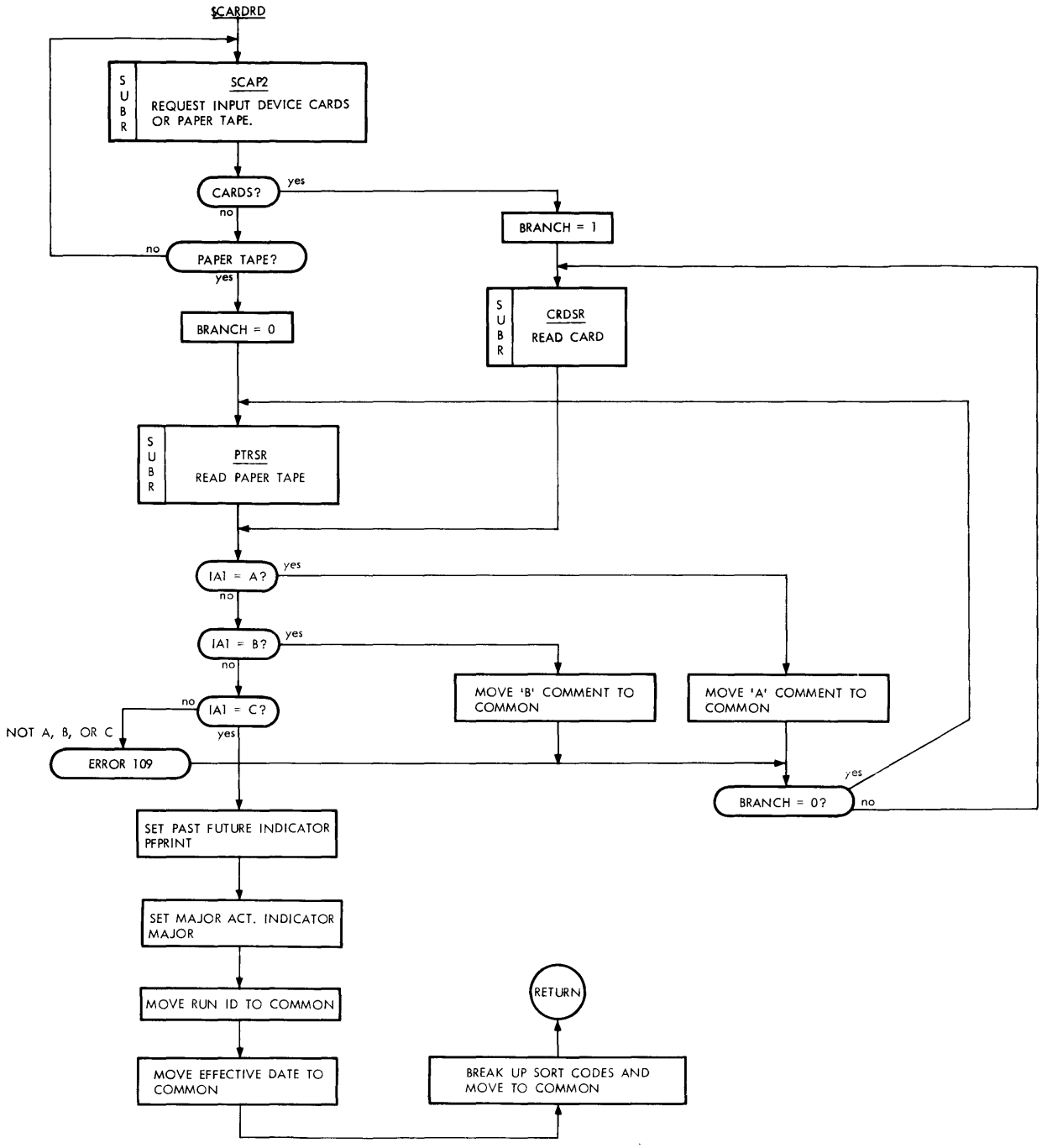
END SUBROUTINE

$END

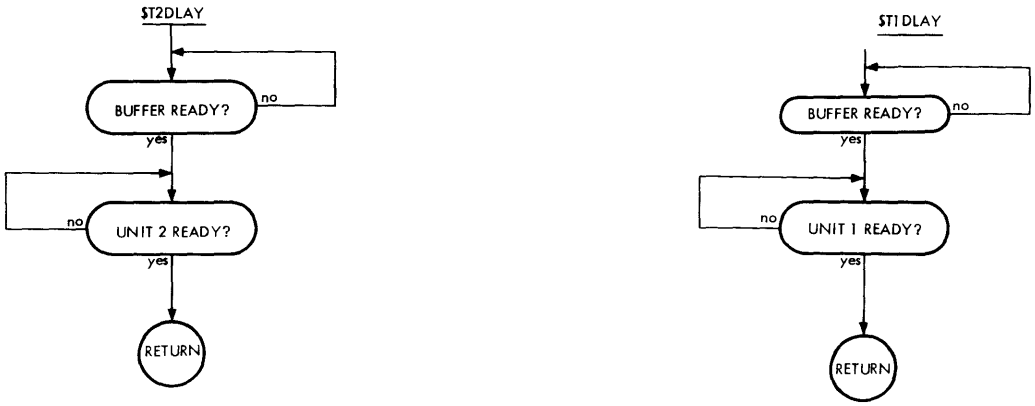
WRITE END OF FILE ON TAPE 1.
TYPE THE WORD 'END' ON TYPEWRITER.

RETURN

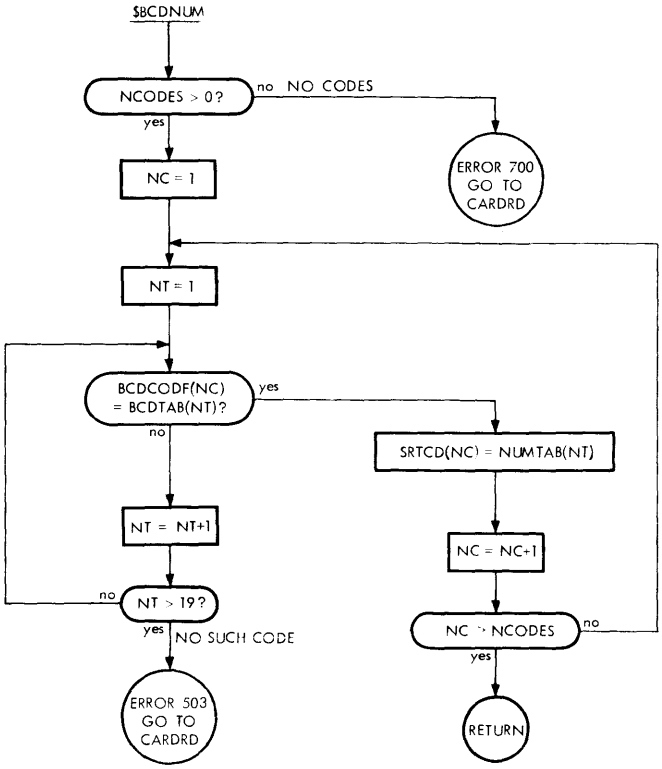

MAIN CONTROL
$MAINCON
SUBR
ADRES1
FIND MACHINE SIZE. COMPUTE ADDRESSES OF S, PC, SC.
SUBR
PARMRD
READ PARAMETER FILE FROM INPUT TAPE. MOVE EFFECTIVE DATA TO COMMON.
SUBR
CARDRD
READ DATA FROM CARDS OR PAPER TAPE. STORE DATA IN COMMON.
END
HALT WHEN NO MORE INPUT RECORD PRESENT
SUBR
BCDNUM
CONVERT BCD SORT KEYS TO NUMERICAL SORT KEYS.
SUBR
SORT
MAIN INTERNAL SORT CONTROL.
SUBR
ADRES2
SET UP ADDRESSES FOR TAPE SORT.
SUBR
FINSORT
MAIN TAPE SORT CONTROL.
SUBR
END
WRITE EOF ON OFF-LINE TAPE. TYPE END ON TYPEWRITER.

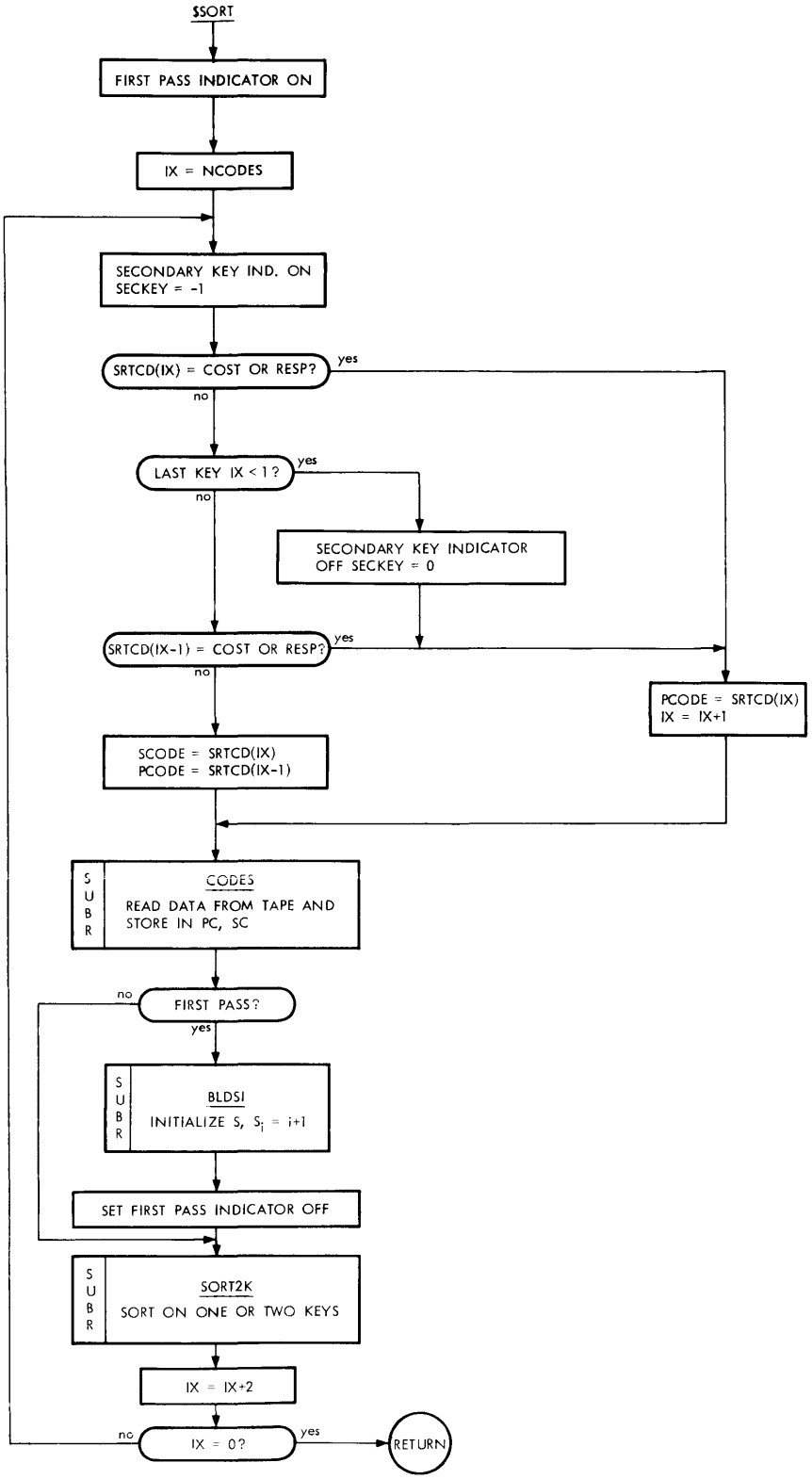

$CARDRD
SUBR
SCAP2
REQUEST INPUT DEVICE CARDS OR PAPER TAPE.
CARDS?
yes
no
BRANCH = 1
no
PAPER TAPE?
yes
BRANCH = 0
SUBR
CRDSR
READ CARD
SUBR
PTRSR
READ PAPER TAPE
IA1 = A?
yes
no
IA1 = B?
yes
no
MOVE 'B' COMMENT TO COMMON
MOVE 'A' COMMENT TO COMMON
no
IA1 = C?
NOT A, B, OR C
yes
ERROR 109
yes
BRANCH = 0?
no
SET PAST FUTURE INDICATOR PFPRINT
SET MAJOR ACT. INDICATOR MAJOR
MOVE RUN ID TO COMMON
MOVE EFFECTIVE DATE TO COMMON
RETURN
BREAK UP SORT CODES AND MOVE TO COMMON

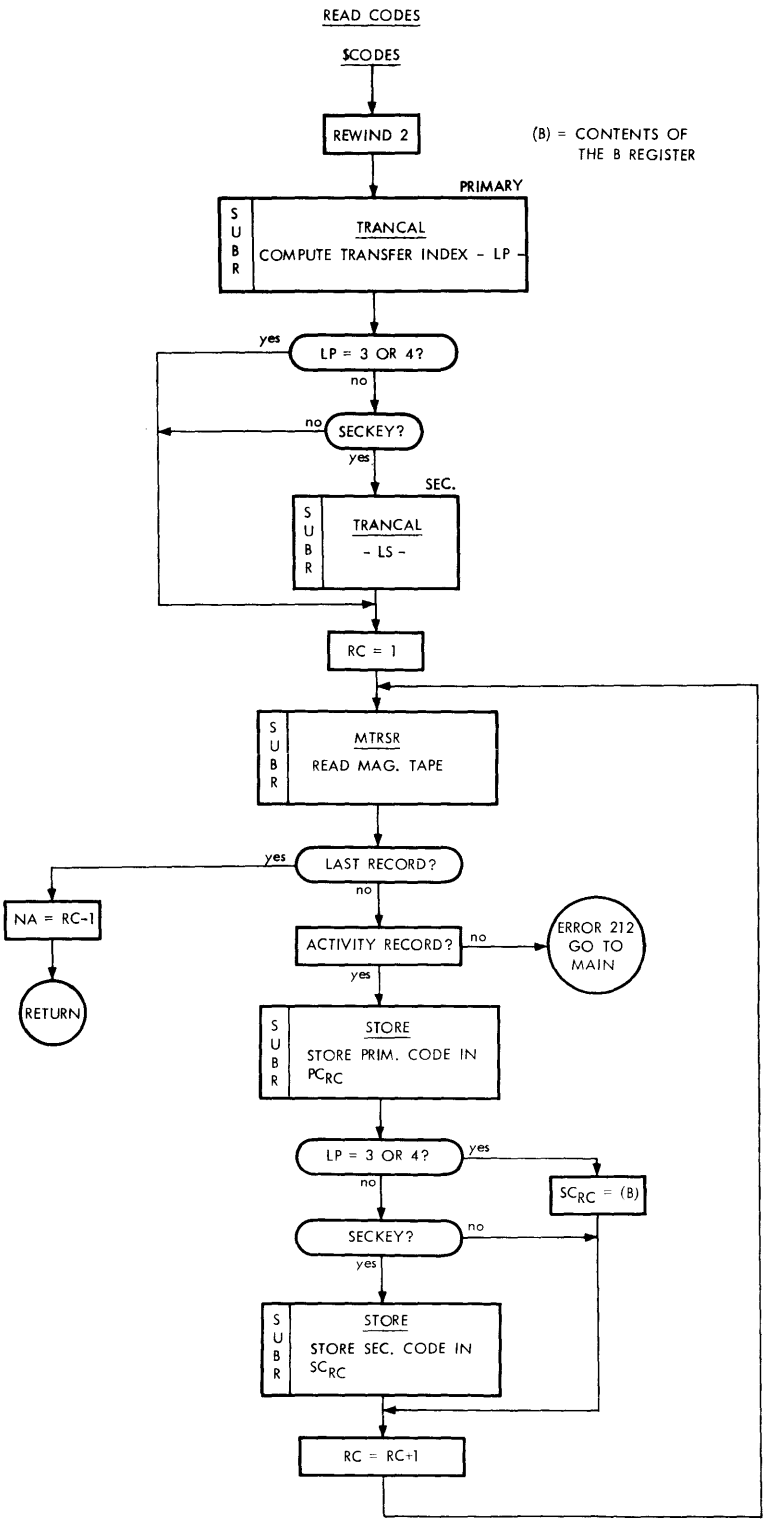
MAGNETIC TAPE READY



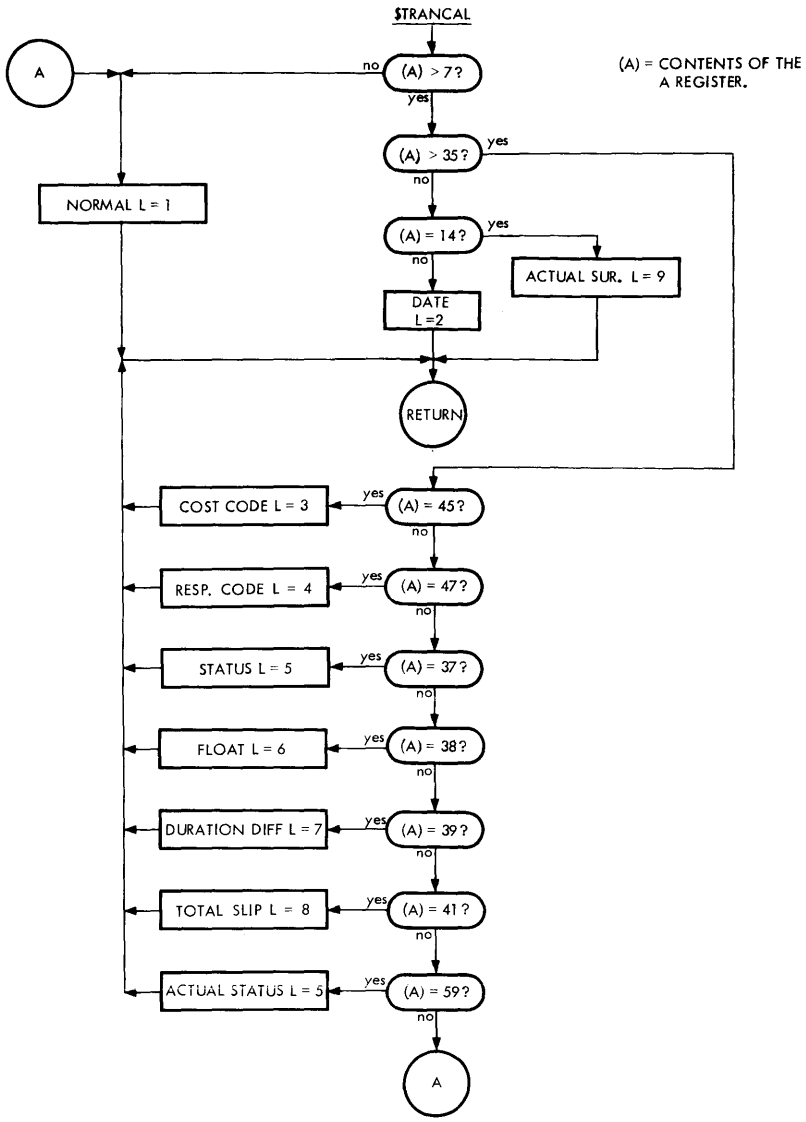
BCD CODE TO NUMERIC

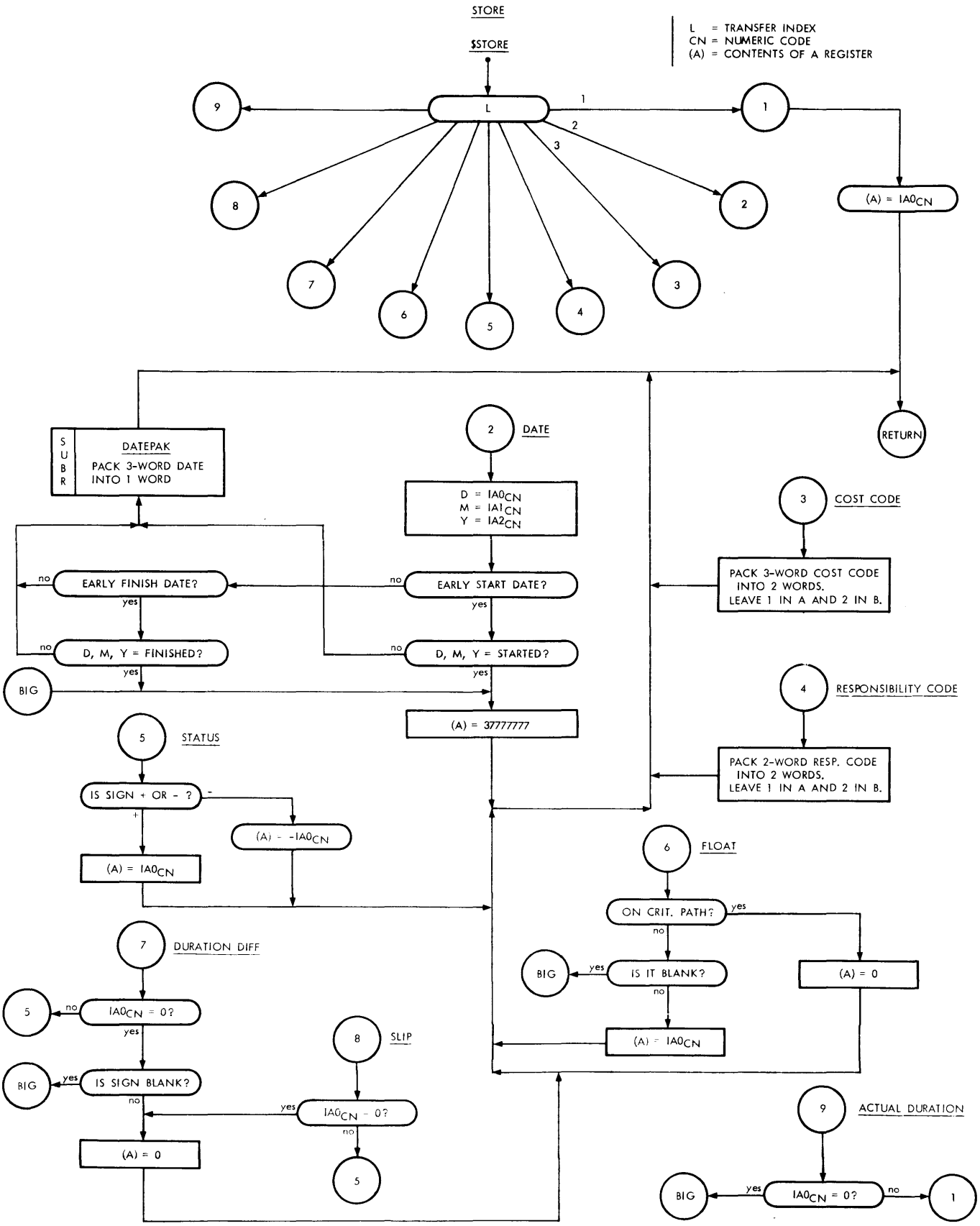



$SORT
FIRST PASS INDICATOR ON
IX = NCODES
SECONDARY KEY IND. ON
SECKEY = -1
SRTCD(IX) = COST OR RESP?
yes
no
LAST KEY IX < 1 ?
yes
no
SECONDARY KEY INDICATOR
OFF SECKEY = 0
SRTCD(IX-1) = COST OR RESP?
yes
no
PCODE = SRTCD(IX)
IX = IX+1
SCODE = SRTCD(IX)
PCODE = SRTCD(IX-1)
SUBR
CODES
READ DATA FROM TAPE AND
STORE IN PC, SC
no
FIRST PASS?
yes
SUBR
BLDSI
INITIALIZE S, S_i = i+1
SET FIRST PASS INDICATOR OFF
SUBR
SORT2K
SORT ON ONE OR TWO KEYS
IX = IX+2
no
IX = 0?
yes
RETURN

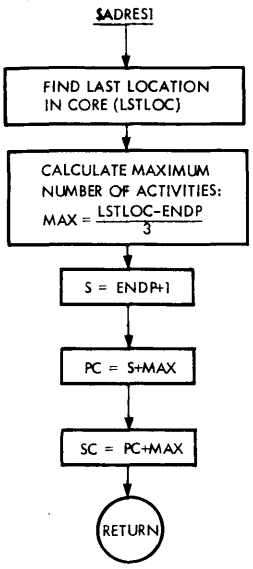

READ CODES
$CODES
REWIND 2
(B) = CONTENTS OF THE B REGISTER
PRIMARY
SUBR
TRANCAL
COMPUTE TRANSFER INDEX - LP -
yes
LP = 3 OR 4?
no
no
SECKEY?
yes
SEC.
SUBR
TRANCAL
- LS -
RC = 1
SUBR
MTRSR
READ MAG. TAPE
yes
LAST RECORD?
no
NA = RC-1
RETURN
ACTIVITY RECORD?
no
ERROR 212 GO TO MAIN
yes
SUBR
STORE
STORE PRIM. CODE IN PCRC
LP = 3 OR 4?
yes
no
SCRC = (B)
SECKEY?
no
yes
SUBR
STORE
STORE SEC. CODE IN SCRC
RC = RC+1

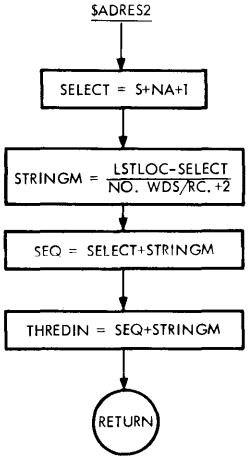
TRANSFER INDEX



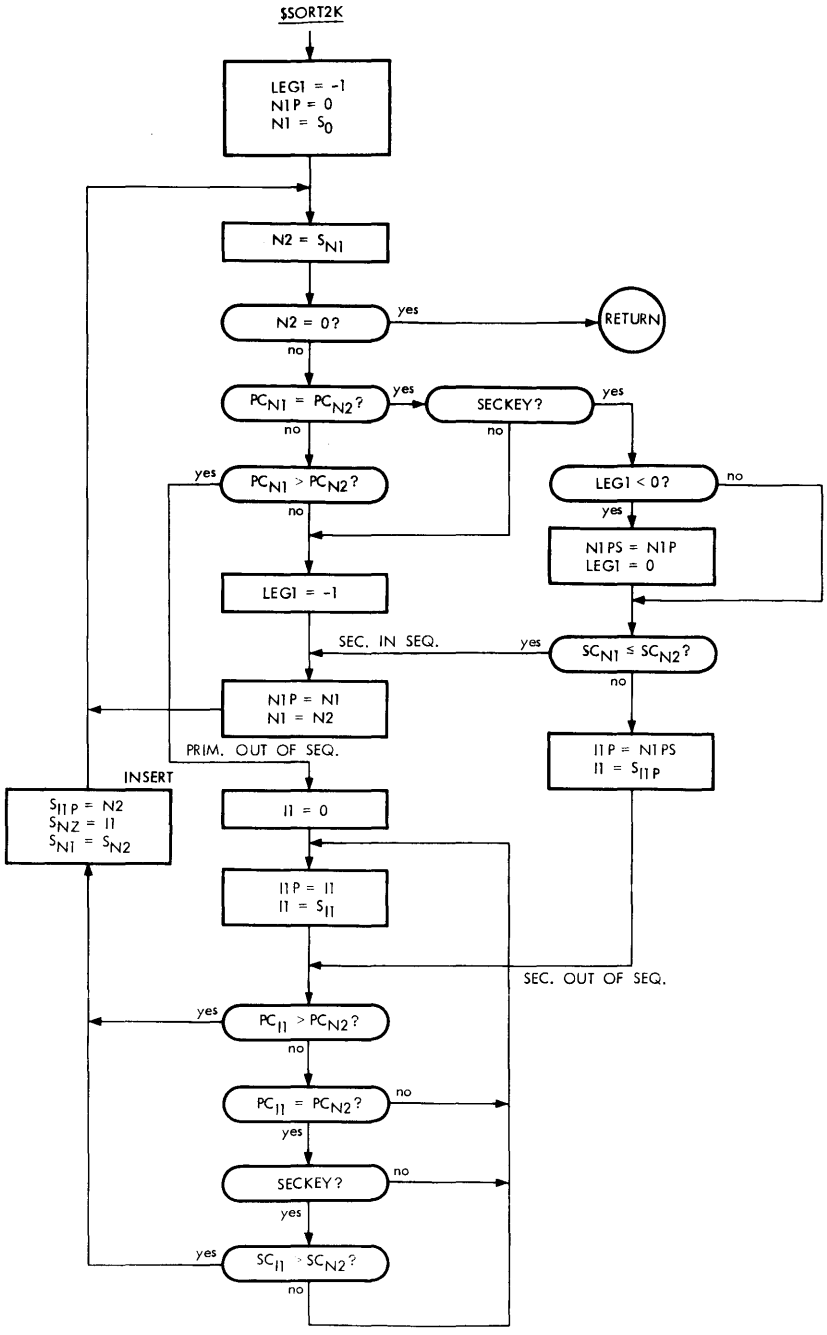

STORE
$STORE
L = TRANSFER INDEX
CN = NUMERIC CODE
(A) = CONTENTS OF A REGISTER
L
1
2
3
9
8
7
6
5
4
(A) = IA0CN
RETURN
SUBR
DATEPAK
PACK 3-WORD DATE INTO 1 WORD
2 DATE
D = IA0CN
M = IA1CN
Y = IA2CN
3 COST CODE
PACK 3-WORD COST CODE INTO 2 WORDS. LEAVE 1 IN A AND 2 IN B.
EARLY FINISH DATE?
EARLY START DATE?
D, M, Y = FINISHED?
D, M, Y = STARTED?
BIG
(A) = 37777777
4 RESPONSIBILITY CODE
PACK 2-WORD RESP. CODE INTO 2 WORDS. LEAVE 1 IN A AND 2 IN B.
5 STATUS
IS SIGN + OR - ?
(A) = -IA0CN
(A) = IA0CN
6 FLOAT
ON CRIT. PATH?
BIG
IS IT BLANK?
(A) = 0
(A) = IA0CN
7 DURATION DIFF
IA0CN = 0?
IS SIGN BLANK?
(A) = 0
8 SLIP
IA0CN = 0?
9 ACTUAL DURATION
IA0CN = 0?
yes
no

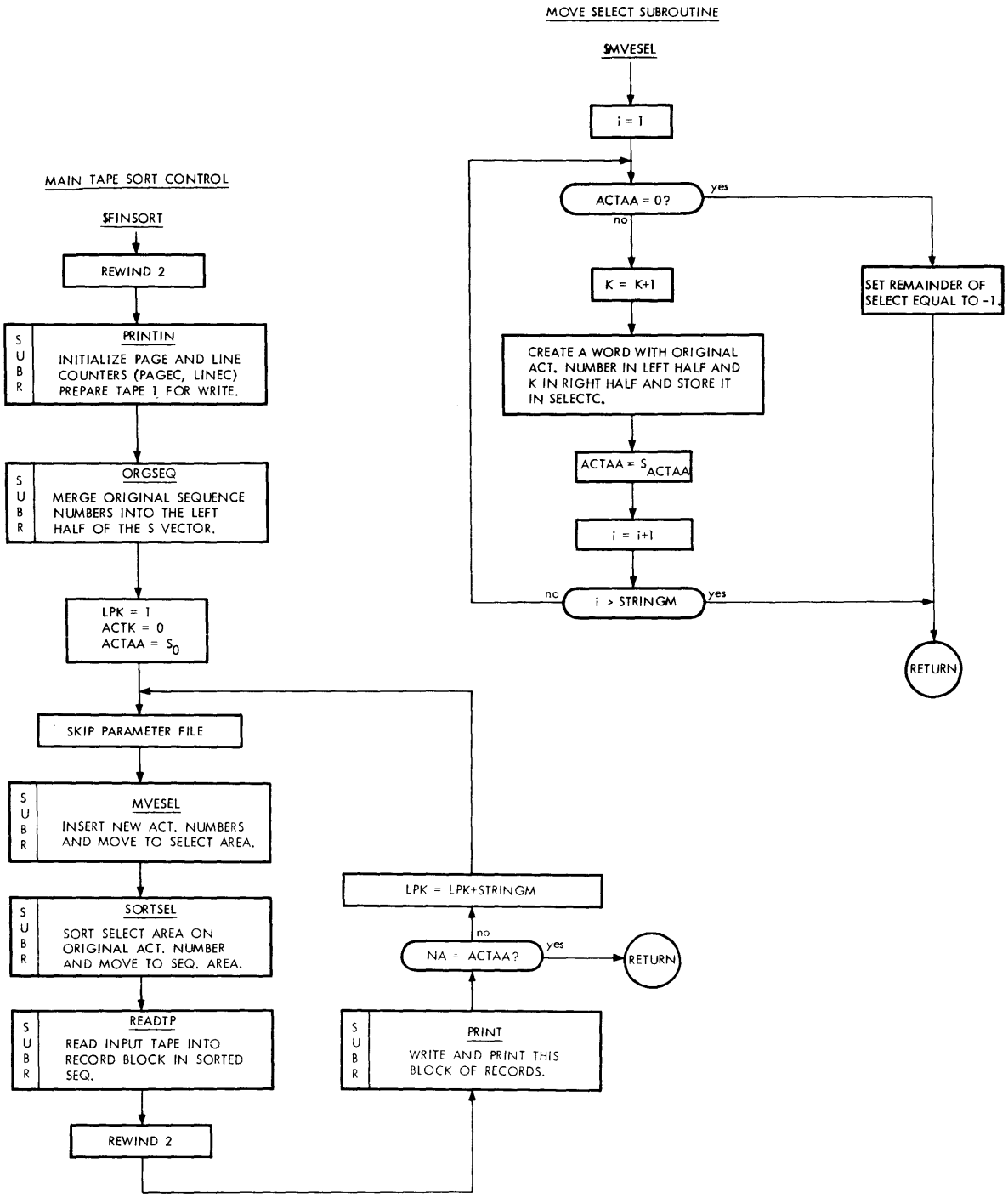
ADDRESS CALCULATION

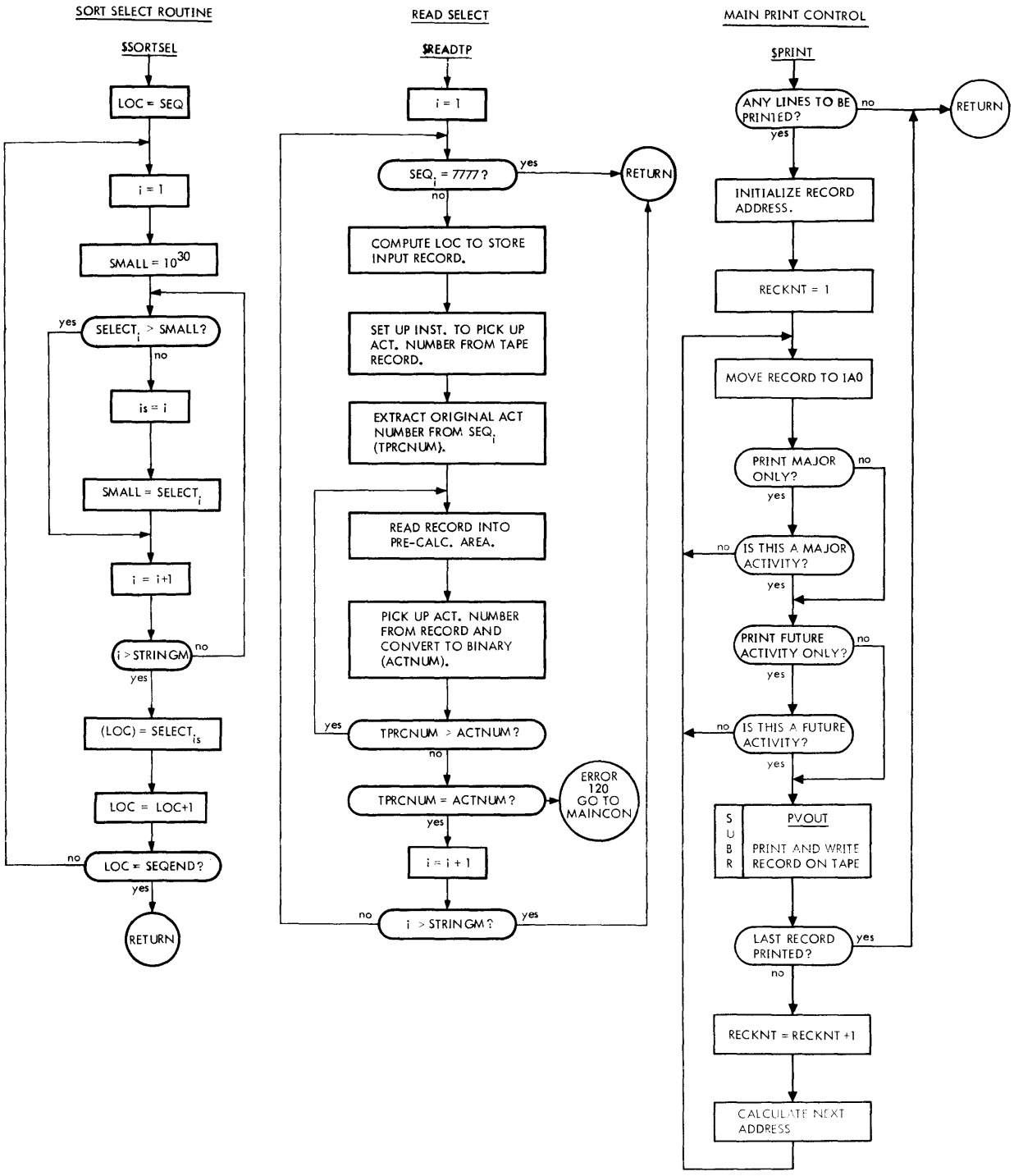


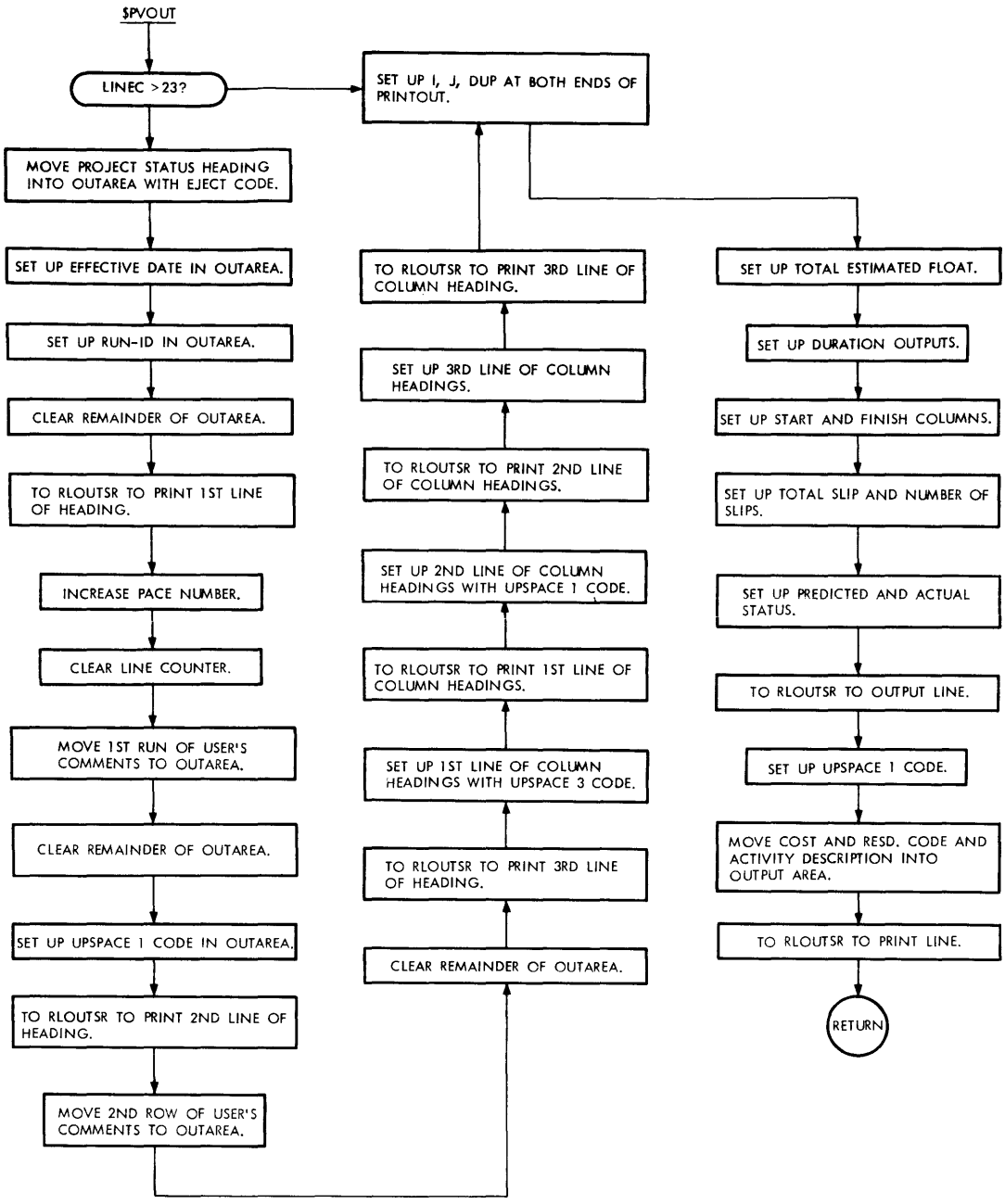


SORT ON TWO KEYS




MOVE SELECT SUBROUTINE
$MVESEL
i = 1
ACTAA = 0?
yes
no
K = K+1
SET REMAINDER OF SELECT EQUAL TO -1
CREATE A WORD WITH ORIGINAL ACT. NUMBER IN LEFT HALF AND K IN RIGHT HALF AND STORE IT IN SELECTC.
ACTAA = S ACTAA
i = i+1
no
i > STRINGM
yes
RETURN
MAIN TAPE SORT CONTROL
$FINSORT
REWIND 2
SUBR
PRINTIN
INITIALIZE PAGE AND LINE COUNTERS (PAGEC, LINEC) PREPARE TAPE 1 FOR WRITE.
SUBR
ORGSEQ
MERGE ORIGINAL SEQUENCE NUMBERS INTO THE LEFT HALF OF THE S VECTOR.
LPK = 1
ACTK = 0
ACTAA = S0
SKIP PARAMETER FILE
SUBR
MVESEL
INSERT NEW ACT. NUMBERS AND MOVE TO SELECT AREA.
SUBR
SORTSEL
SORT SELECT AREA ON ORIGINAL ACT. NUMBER AND MOVE TO SEQ. AREA.
SUBR
READTP
READ INPUT TAPE INTO RECORD BLOCK IN SORTED SEQ.
REWIND 2
LPK = LPK+STRINGM
no
NA = ACTAA?
yes
RETURN
SUBR
PRINT
WRITE AND PRINT THIS BLOCK OF RECORDS.


SORT SELECT ROUTINE
$SORTSEL
LOC = SEQ
i = 1
SMALL = 10^30
SELECT_i > SMALL?
yes
no
is = i
SMALL = SELECT_i
i = i+1
i > STRINGM
no
yes
(LOC) = SELECT_is
LOC = LOC+1
LOC = SEQEND?
no
yes
RETURN
READ SELECT
$READTP
i = 1
SEQ_i = 7777?
yes
RETURN
no
COMPUTE LOC TO STORE INPUT RECORD.
SET UP INST. TO PICK UP ACT. NUMBER FROM TAPE RECORD.
EXTRACT ORIGINAL ACT NUMBER FROM SEQ_i (TPRCNUM).
READ RECORD INTO PRE-CALC. AREA.
PICK UP ACT. NUMBER FROM RECORD AND CONVERT TO BINARY (ACTNUM).
TPRCNUM > ACTNUM?
yes
no
TPRCNUM = ACTNUM?
ERROR 120 GO TO MAINCON
yes
i = i + 1
i > STRINGM?
no
yes
MAIN PRINT CONTROL
$PRINT
ANY LINES TO BE PRINTED?
no
RETURN
yes
INITIALIZE RECORD ADDRESS.
RECKNT = 1
MOVE RECORD TO 1A0
PRINT MAJOR ONLY?
no
yes
IS THIS A MAJOR ACTIVITY?
no
yes
PRINT FUTURE ACTIVITY ONLY?
no
yes
IS THIS A FUTURE ACTIVITY?
no
yes
SUBR
PVOUT
PRINT AND WRITE RECORD ON TAPE
LAST RECORD PRINTED?
yes
no
RECKNT = RECKNT +1
CALCULATE NEXT ADDRESS


$PVOUT
LINEC >23?
SET UP I, J, DUP AT BOTH ENDS OF PRINTOUT.
MOVE PROJECT STATUS HEADING INTO OUTAREA WITH EJECT CODE.
SET UP EFFECTIVE DATE IN OUTAREA.
SET UP RUN-ID IN OUTAREA.
CLEAR REMAINDER OF OUTAREA.
TO RLOUTSR TO PRINT 1ST LINE OF HEADING.
INCREASE PACE NUMBER.
CLEAR LINE COUNTER.
MOVE 1ST RUN OF USER'S COMMENTS TO OUTAREA.
CLEAR REMAINDER OF OUTAREA.
SET UP UPSPACE 1 CODE IN OUTAREA.
TO RLOUTSR TO PRINT 2ND LINE OF HEADING.
MOVE 2ND ROW OF USER'S COMMENTS TO OUTAREA.
TO RLOUTSR TO PRINT 3RD LINE OF COLUMN HEADING.
SET UP 3RD LINE OF COLUMN HEADINGS.
TO RLOUTSR TO PRINT 2ND LINE OF COLUMN HEADINGS.
SET UP 2ND LINE OF COLUMN HEADINGS WITH UPSPACE 1 CODE.
TO RLOUTSR TO PRINT 1ST LINE OF COLUMN HEADINGS.
SET UP 1ST LINE OF COLUMN HEADINGS WITH UPSPACE 3 CODE.
TO RLOUTSR TO PRINT 3RD LINE OF HEADING.
CLEAR REMAINDER OF OUTAREA.
SET UP TOTAL ESTIMATED FLOAT.
SET UP DURATION OUTPUTS.
SET UP START AND FINISH COLUMNS.
SET UP TOTAL SLIP AND NUMBER OF SLIPS.
SET UP PREDICTED AND ACTUAL STATUS.
TO RLOUTSR TO OUTPUT LINE.
SET UP UPSPACE 1 CODE.
MOVE COST AND RESD. CODE AND ACTIVITY DESCRIPTION INTO OUTPUT AREA.
TO RLOUTSR TO PRINT LINE.
RETURN