Application Program

H20-0284-1

1130 Continuous System Modeling Program

(1130-CX-13X)

System Manual

This manual provides detailed information to assist the user in gaining a more thorough knowledge of the programming logic employed in the application.

The System Manual presents flowcharts, flowchart narratives, a list of switches, and program listings.

Kristofer Sweger

Second Edition

H20-0284-1 is a minor revision, incorporating TNL N20-1015, and does not obsolete H20-0284-0 with N20-1015.

Copies of this and other IBM publications can be obtained through IBM branch offices. Address comments concerning the contents of this publication to
IBM, Technical Publications Department, 112 East Post Road, White Plains, N.Y. 10601

SYSTEM MANUAL

CONTENTS

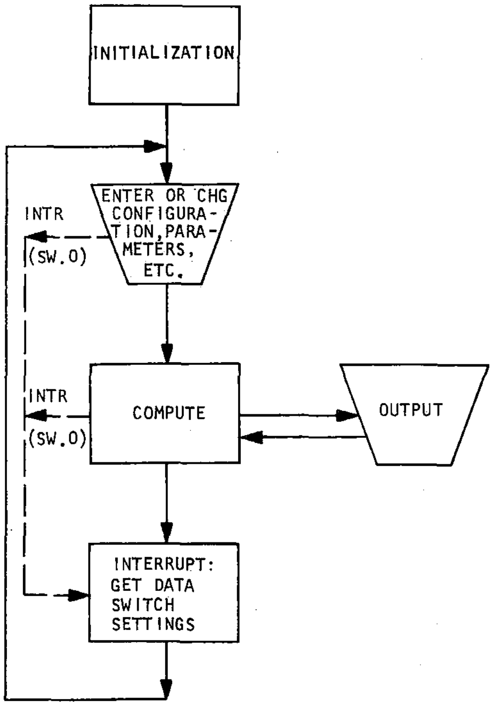
FLOWCHARTS

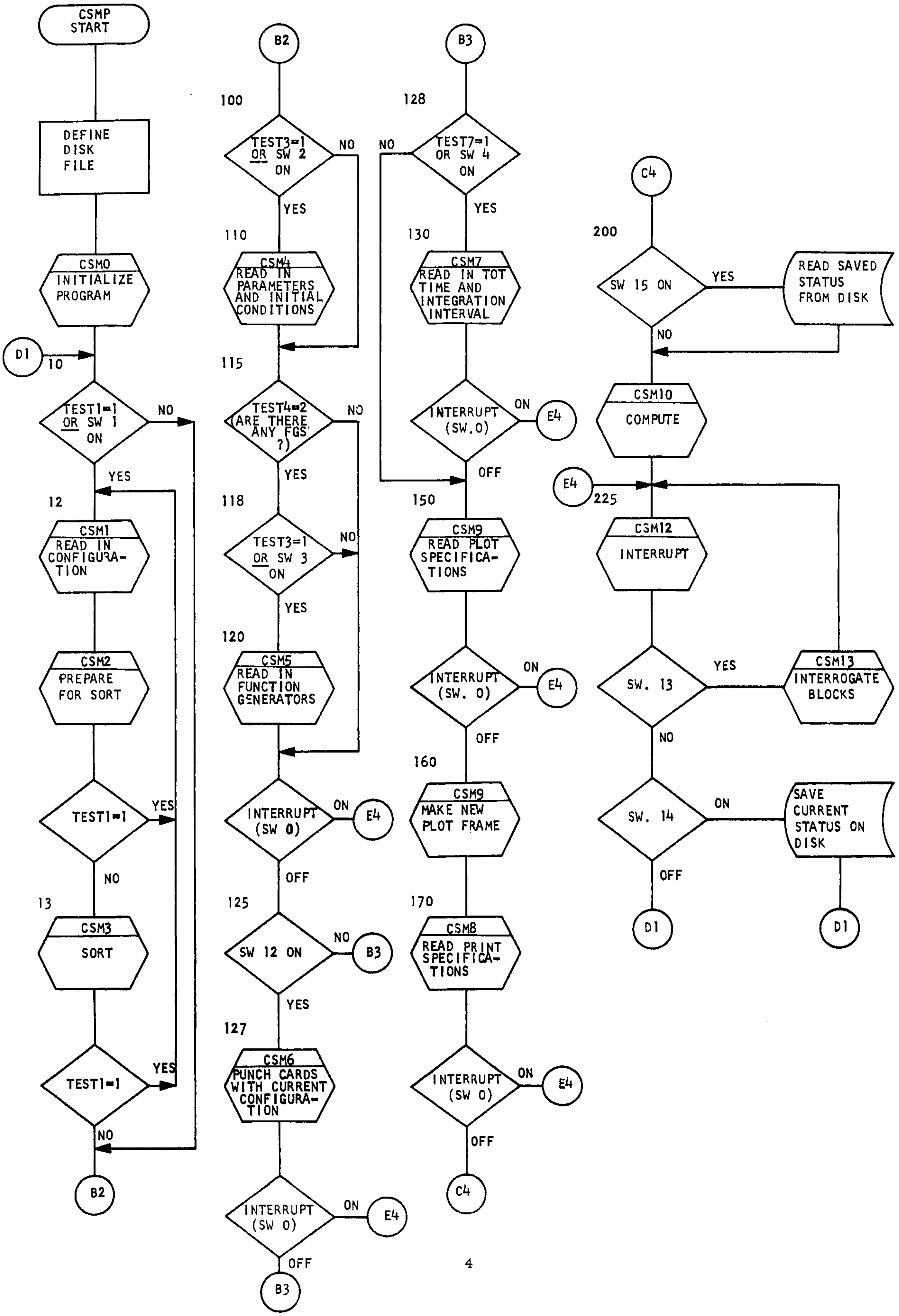
The flowcharts are in the following order:

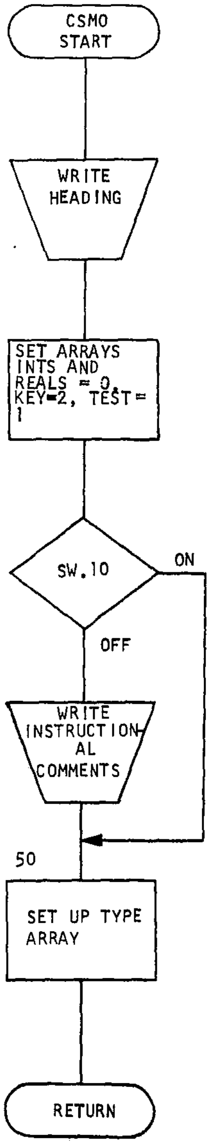
System	-	The overall system flowchart
Deck one (CSA)	-	Required Program Routines
CSMP	-	Main Program
CSM0	-	Initialization Routine
CSM1	-	Configuration Specification Routine
CSM2	-	Presort Routine
CSM3	-	Sort Routine
CSM4	-	Initial Conditions and Parameters Specification Routine
CSM5	-	Function Generator Intercept Specification Routine
CSM6	-	Punch Data Routine
CSM7	-	Request Timing Information Routine
CSM10	-	Control Computation and Output Routine
CSM11	-	Computation Routine
CSM12	-	Sense Switch Option Routine
CSM13	-	Block Interrogation Routine

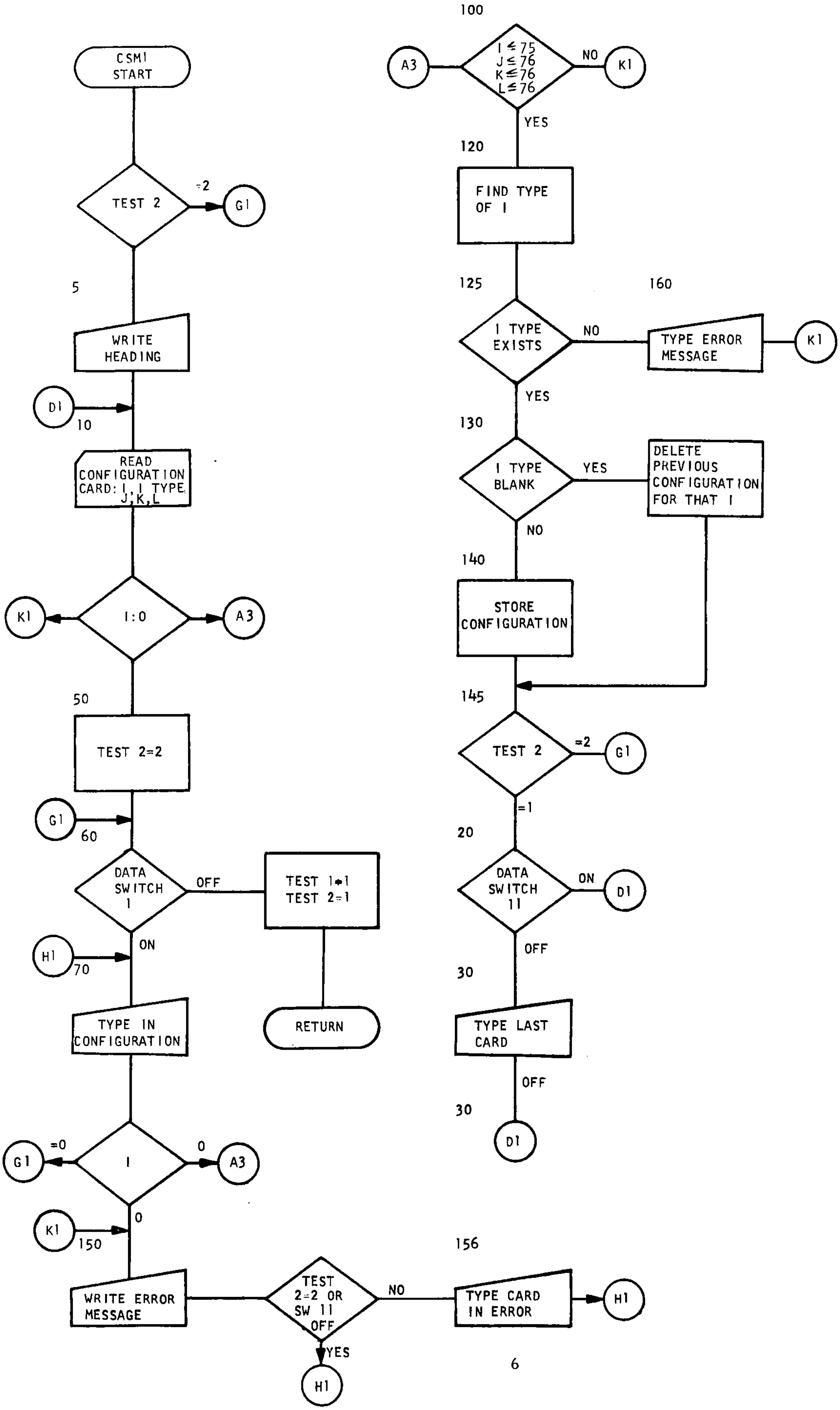
Deck two (CSB)	-	Routines for System with 1627 Plotter
CSM8	-	Printer Output Routine
CSM8A	-	Request Print Information Routine
CSM9	-	Plotter Control Routine
CSM9A	-	Plot Grid Routine
CSM9B	-	Request Plot Information Routine
Deck three (CSC)	-	Routines for System without 1627 Plotter
CSM8 Alternate	-	Alternate Printer Output Routine
CSM8A Alternate	-	Alternate Request Print Information Routine
CSM9 Alternate	-	Alternate (Dummy) Plotter Control Routine
Deck four (CSD)	-	Special Subroutines

There are no flowcharts for these routines. As supplied with the program, they are dummy routines (simple returns). If the user desires to use special purpose blocks, then these are the routines that will be replaced with user coded routines.

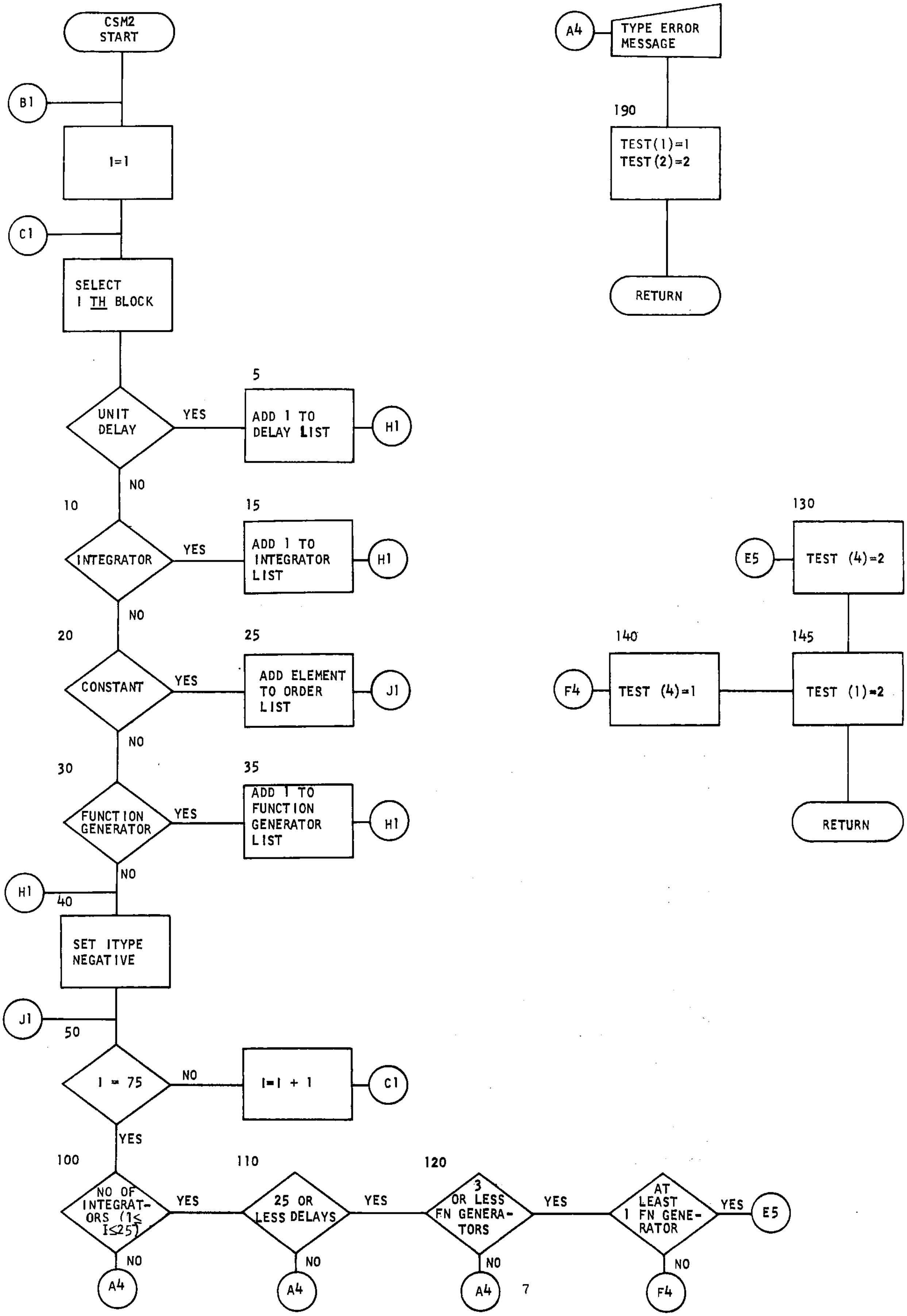

INITIALIZATION
ENTER OR CHG CONFIGURATION, PARAMETERS, ETC.
INTR
(SW.0)
COMPUTE
INTR
(SW.0)
OUTPUT
INTERRUPT: GET DATA SWITCH SETTINGS

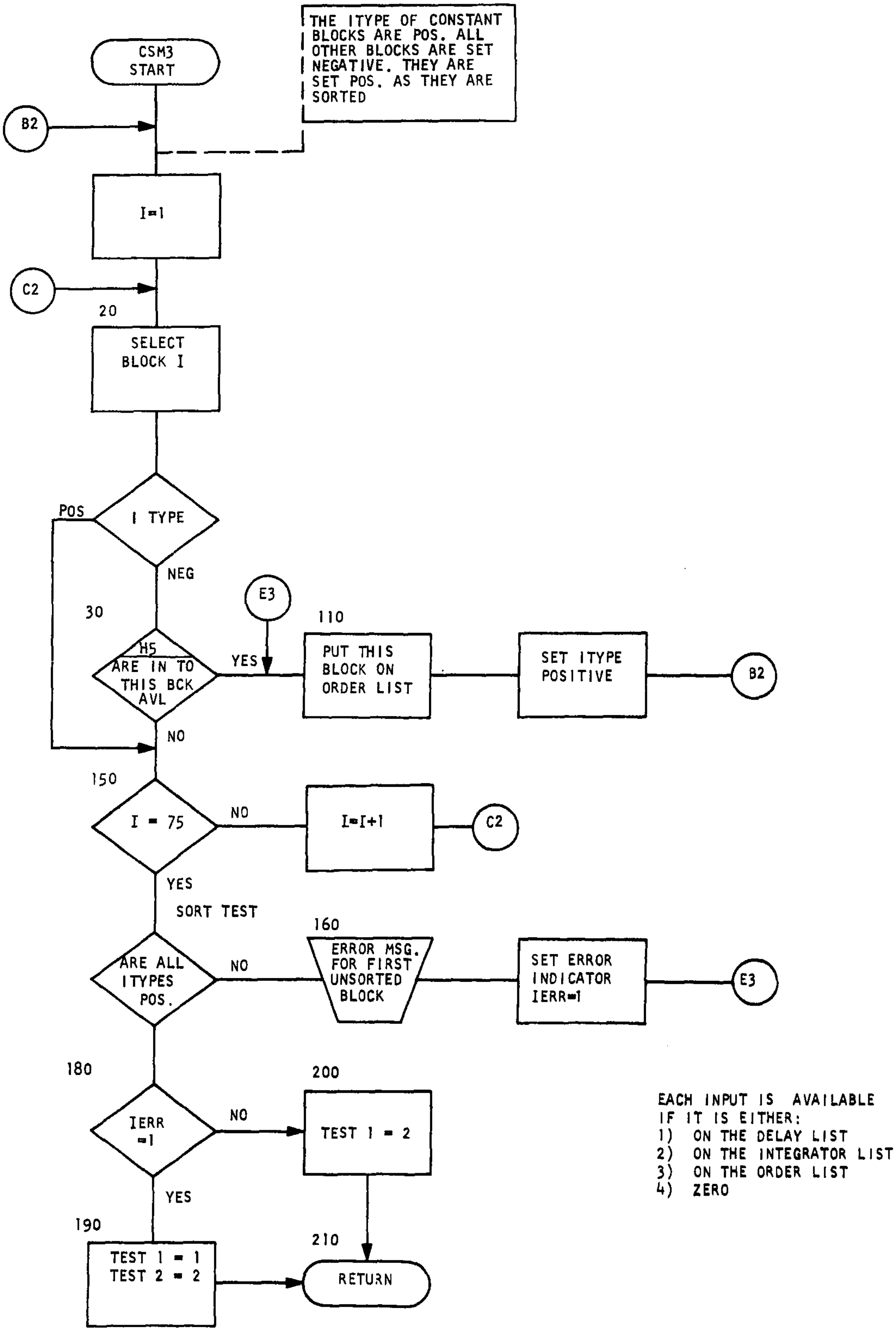

CSMP START
DEFINE DISK FILE
CSM0 INITIALIZE PROGRAM
D1
10
TEST1=1 OR SW 1 ON
NO
YES
12
CSM1 READ IN CONFIGURA-TION
CSM2 PREPARE FOR SORT
TEST1=1
YES
NO
13
CSM3 SORT
TEST1=1
YES
NO
B2
B2
100
TEST3=1 OR SW 2 ON
NO
YES
110
CSM4 READ IN PARAMETERS AND INITIAL CONDITIONS
115
TEST4=2 (ARE THERE ANY FGS' ?)
NO
YES
118
TEST3=1 OR SW 3 ON
NO
YES
120
CSM5 READ IN FUNCTION GENERATORS
INTERRUPT (SW 0)
ON
E4
OFF
125
SW 12 ON
NO
B3
YES
127
CSM6 PUNCH CARDS WITH CURRENT CONFIGURA-TION
INTERRUPT (SW 0)
ON
E4
OFF
B3
B3
128
TEST7=1 OR SW 4 ON
NO
YES
130
CSM7 READ IN TOT TIME AND INTEGRATION INTERVAL
INTERRUPT (SW.0)
ON
E4
OFF
150
CSM9 READ PLOT SPECIFICA-TIONS
INTERRUPT (SW. 0)
ON
E4
OFF
160
CSM9 MAKE NEW PLOT FRAME
170
CSM8 READ PRINT SPECIFICA-TIONS
INTERRUPT (SW 0)
ON
E4
OFF
C4
C4
200
SW 15 ON
YES
READ SAVED STATUS FROM DISK
NO
CSM10 COMPUTE
E4
225
CSM12 INTERRUPT
SW. 13
YES
CSM13 INTERROGATE BLOCKS
NO
SW. 14
ON
SAVE CURRENT STATUS ON DISK
OFF
D1
D1


CSMO
START
WRITE
HEADING
SET ARRAYS
INTS AND
REALS = 0,
KEY=2, TEST=
1
SW.10
ON
OFF
WRITE
INSTRUCTION-
AL
COMMENTS
50
SET UP TYPE
ARRAY
RETURN

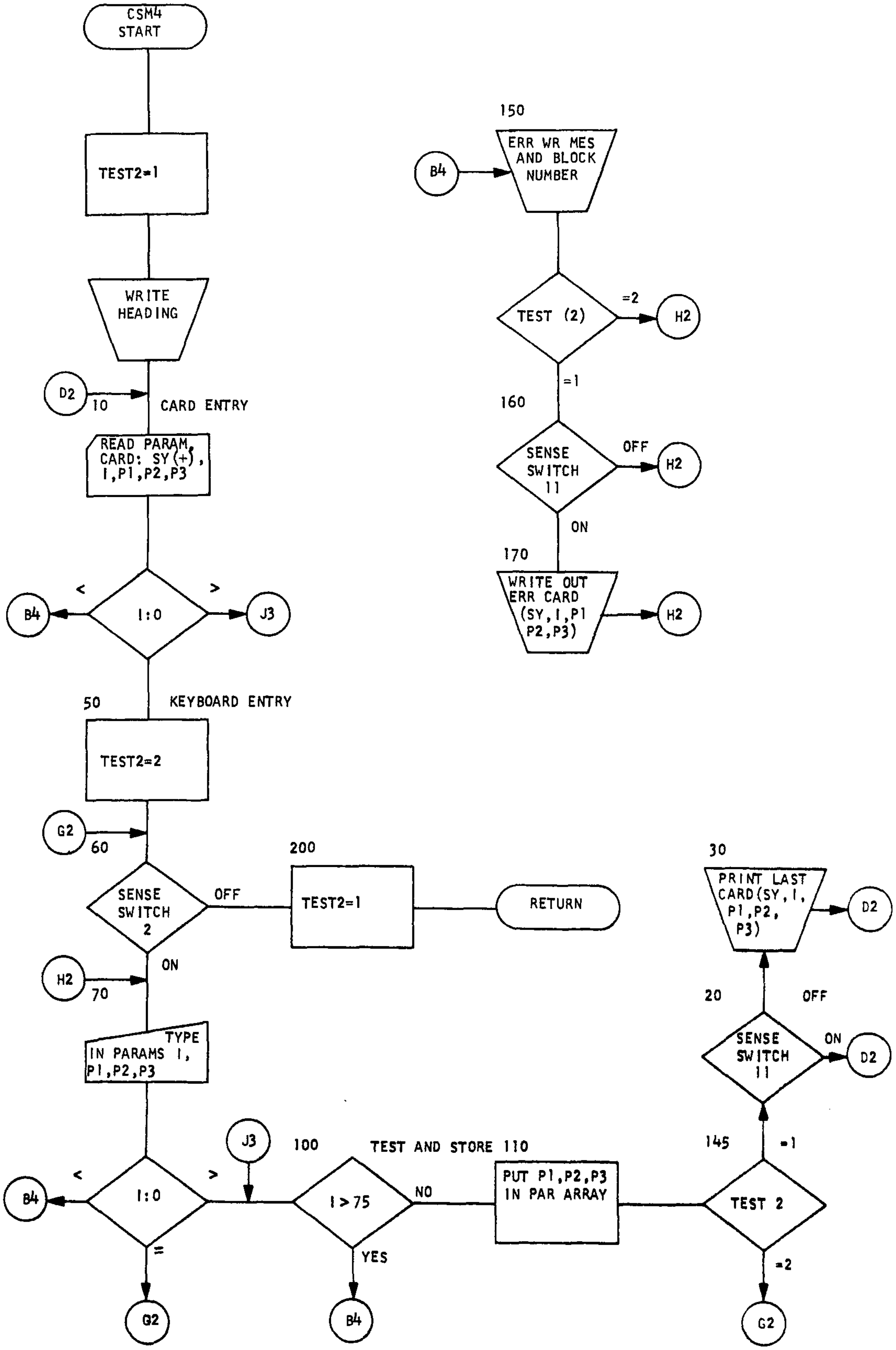

CSM1 START
TEST 2
=2
G1
5
WRITE HEADING
D1
10
READ CONFIGURATION CARD: I, I TYPE, J, K, L
K1
I:0
A3
50
TEST 2=2
G1
60
DATA SWITCH 1
OFF
TEST 1=1 TEST 2=1
RETURN
ON
H1
70
TYPE IN CONFIGURATION
G1
=0
I
0
A3
0
K1
150
WRITE ERROR MESSAGE
TEST 2=2 OR SW 11 OFF
NO
YES
H1
156
TYPE CARD IN ERROR
H1
100
A3
I≤75 J≤76 K≤76 L≤76
NO
K1
YES
120
FIND TYPE OF I
125
I TYPE EXISTS
NO
160
TYPE ERROR MESSAGE
K1
YES
130
I TYPE BLANK
YES
DELETE PREVIOUS CONFIGURATION FOR THAT I
NO
140
STORE CONFIGURATION
145
TEST 2
=2
G1
=1
20
DATA SWITCH 11
ON
D1
OFF
30
TYPE LAST CARD
OFF
30
D1

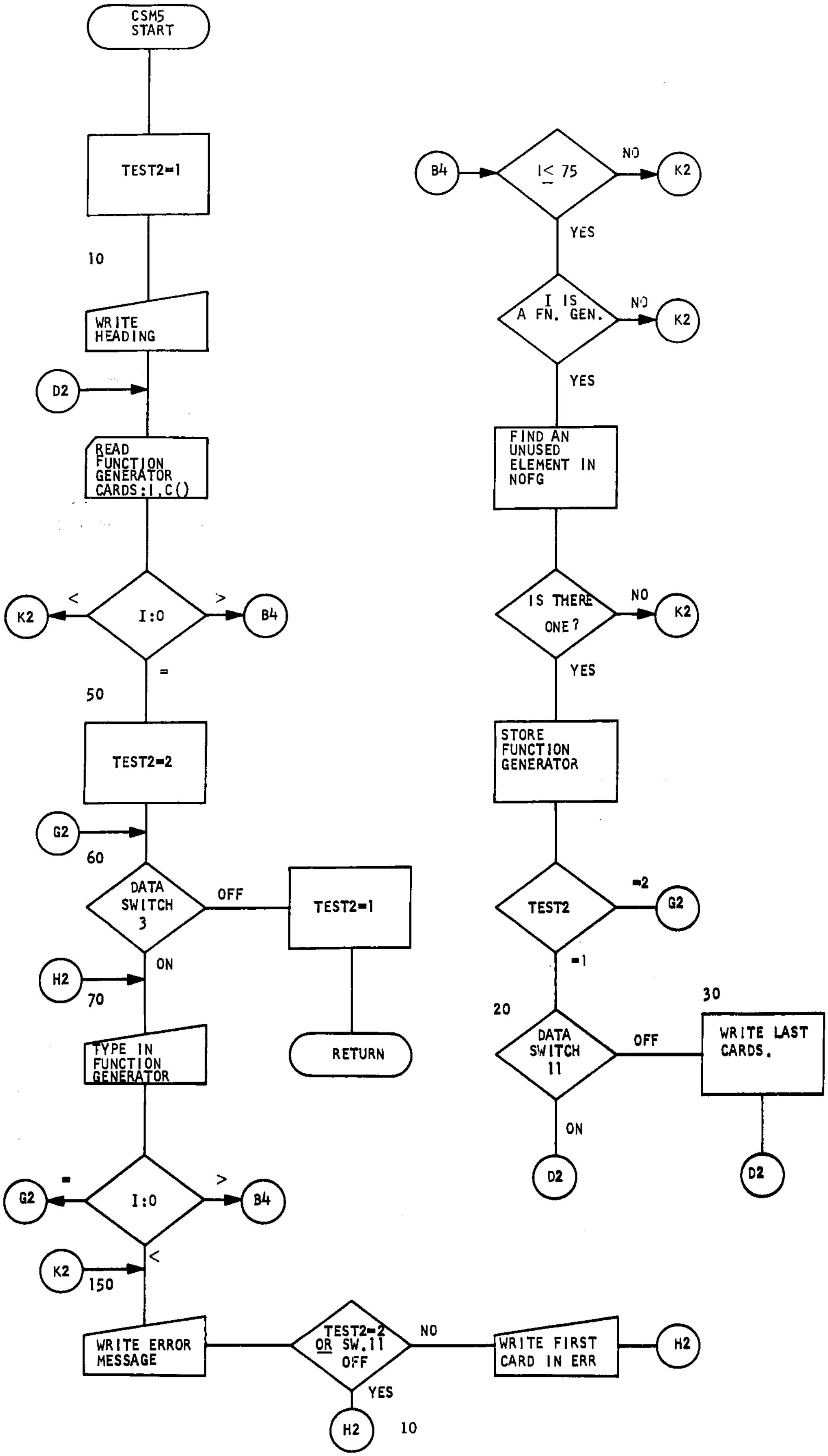
CSM2: PREPARE FOR SORT

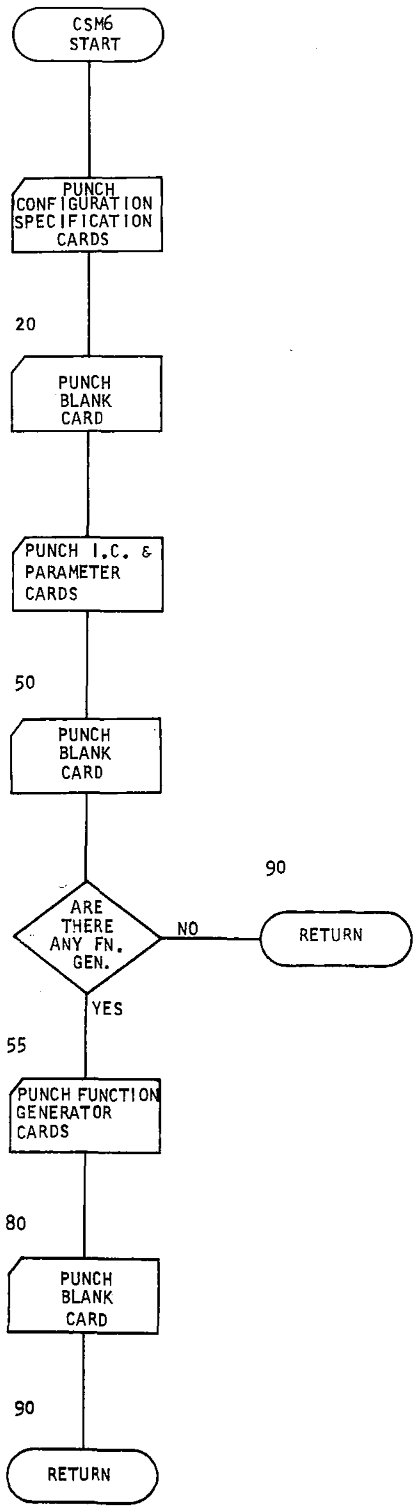


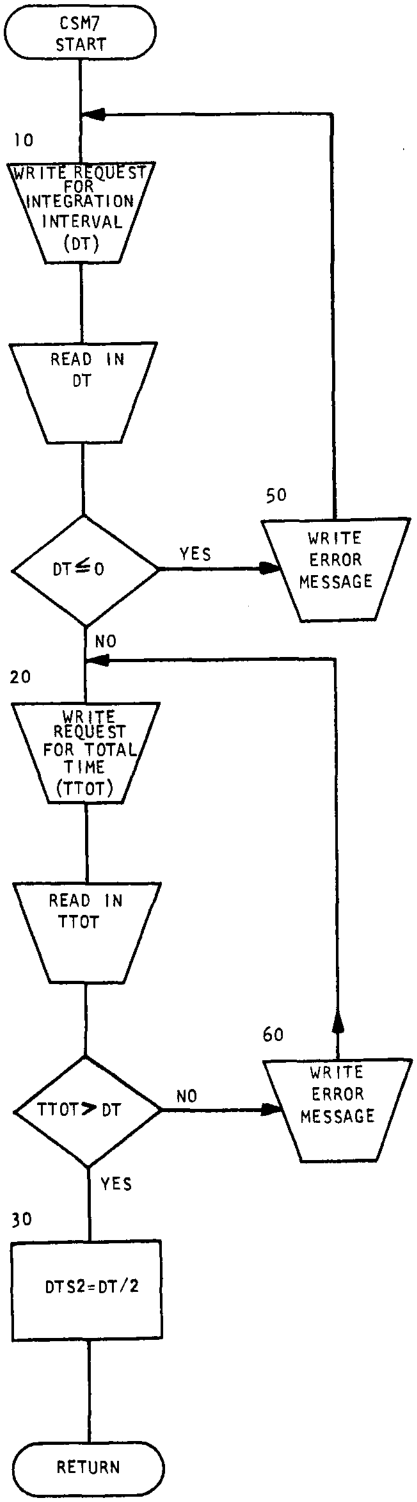

THE ITYPE OF CONSTANT BLOCKS ARE POS. ALL OTHER BLOCKS ARE SET NEGATIVE. THEY ARE SET POS. AS THEY ARE SORTED
CSM3 START
B2
I=1
C2
20
SELECT BLOCK I
POS
I TYPE
NEG
30
E3
110
H5 ARE IN TO THIS BCK AVL
YES
PUT THIS BLOCK ON ORDER LIST
SET ITYPE POSITIVE
B2
NO
150
I = 75
NO
I=I+1
C2
YES
SORT TEST
160
ARE ALL ITYPES POS.
NO
ERROR MSG. FOR FIRST UNSORTED BLOCK
SET ERROR INDICATOR IERR=1
E3
180
IERR =1
NO
200
TEST 1 = 2
YES
190
TEST 1 = 1
TEST 2 = 2
210
RETURN

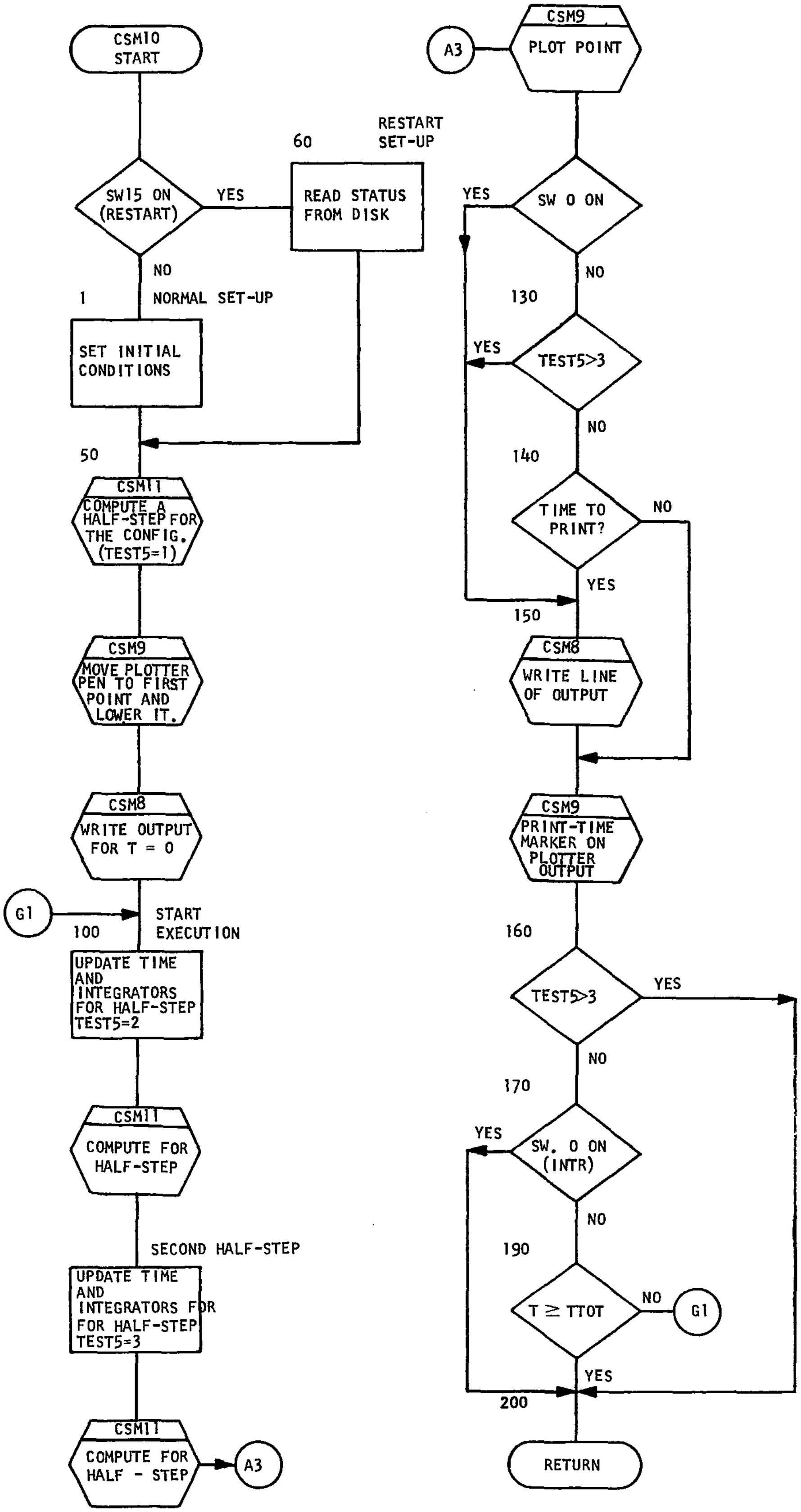
EACH INPUT IS AVAILABLE IF IT IS EITHER:
1) ON THE DELAY LIST
2) ON THE INTEGRATOR LIST
3) ON THE ORDER LIST
4) ZERO

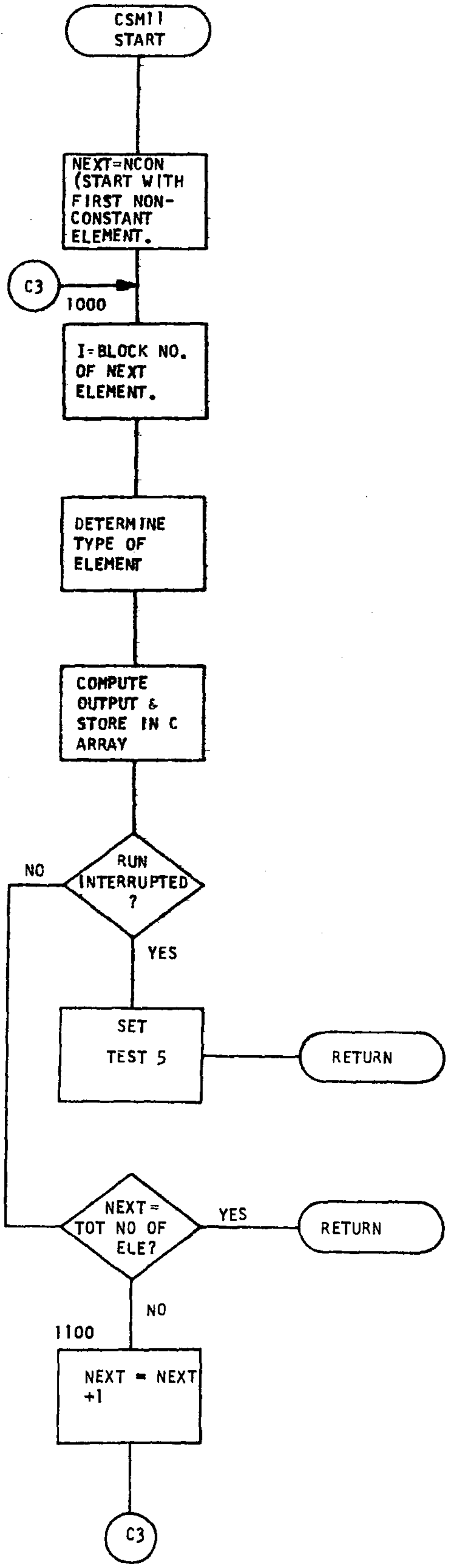

CSM4
START
TEST2=1
WRITE
HEADING
D2
10
CARD ENTRY
READ PARAM,
CARD: SY(+),
I,P1,P2,P3
<
>
B4
I:0
J3
50
KEYBOARD ENTRY
TEST2=2
G2
60
SENSE
SWITCH
2
OFF
200
TEST2=1
RETURN
ON
H2
70
TYPE
IN PARAMS I,
P1,P2,P3
<
>
B4
I:0
=
G2
J3
100
TEST AND STORE 110
I > 75
NO
YES
B4
PUT P1,P2,P3
IN PAR ARRAY
145
=1
TEST 2
=2
G2
20
OFF
SENSE
SWITCH
11
ON
D2
30
PRINT LAST
CARD(SY,I,
P1,P2,
P3)
D2
150
B4
ERR WR MES
AND BLOCK
NUMBER
TEST (2)
=2
H2
=1
160
SENSE
SWITCH
11
OFF
H2
ON
170
WRITE OUT
ERR CARD
(SY,I,P1
P2,P3)
H2

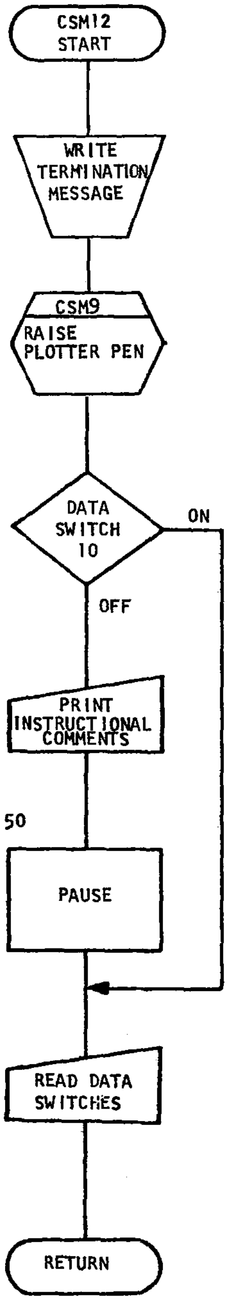

CSM5 START
TEST2=1
10
WRITE HEADING
D2
READ FUNCTION GENERATOR CARDS:I,C()
K2
<
I:0
>
B4
=
50
TEST2=2
G2
60
DATA SWITCH 3
OFF
TEST2=1
RETURN
ON
H2
70
TYPE IN FUNCTION GENERATOR
G2
=
I:0
>
B4
<
K2
150
WRITE ERROR MESSAGE
TEST2=2 OR SW.11 OFF
NO
WRITE FIRST CARD IN ERR
H2
YES
H2
B4
I≤ 75
NO
K2
YES
I IS A FN. GEN.
NO
K2
YES
FIND AN UNUSED ELEMENT IN NOFG
IS THERE ONE?
NO
K2
YES
STORE FUNCTION GENERATOR
TEST2
=2
G2
=1
20
DATA SWITCH 11
OFF
30
WRITE LAST CARDS.
ON
D2
D2

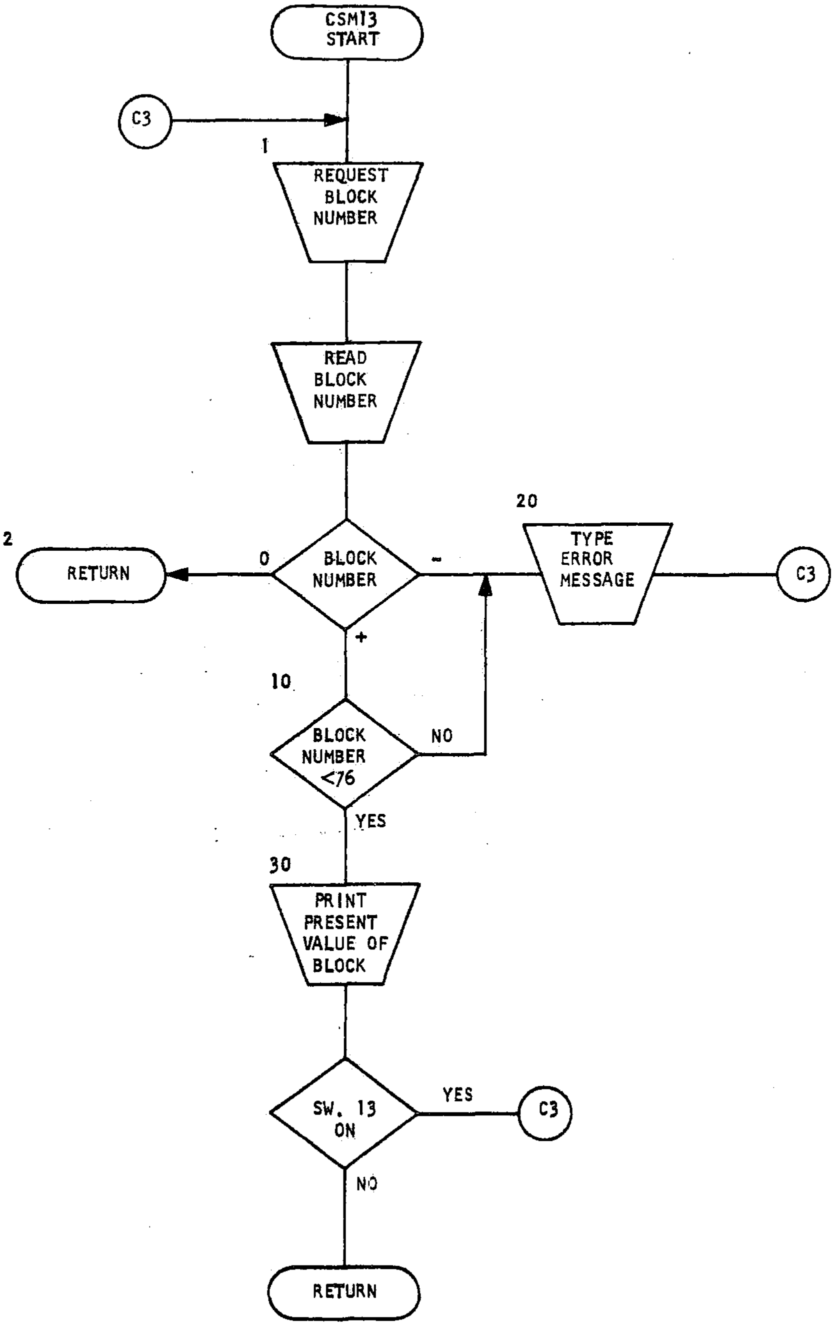

CSM6
START
PUNCH
CONFIGURATION
SPECIFICATION
CARDS
20
PUNCH
BLANK
CARD
PUNCH I.C. &
PARAMETER
CARDS
50
PUNCH
BLANK
CARD
ARE
THERE
ANY FN.
GEN.
NO
90
RETURN
YES
55
PUNCH FUNCTION
GENERATOR
CARDS
80
PUNCH
BLANK
CARD
90
RETURN

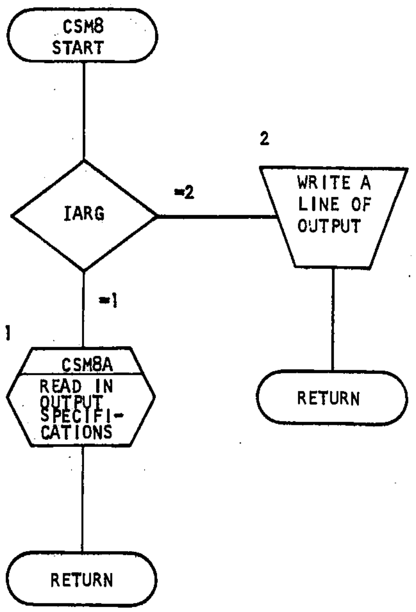

CSM7
START
10
WRITE REQUEST FOR INTEGRATION INTERVAL (DT)
READ IN DT
$DT \leq 0$
YES
50
WRITE ERROR MESSAGE
NO
20
WRITE REQUEST FOR TOTAL TIME (TTOT)
READ IN TTOT
TTOT > DT
NO
60
WRITE ERROR MESSAGE
YES
30
DTS2=DT/2
RETURN

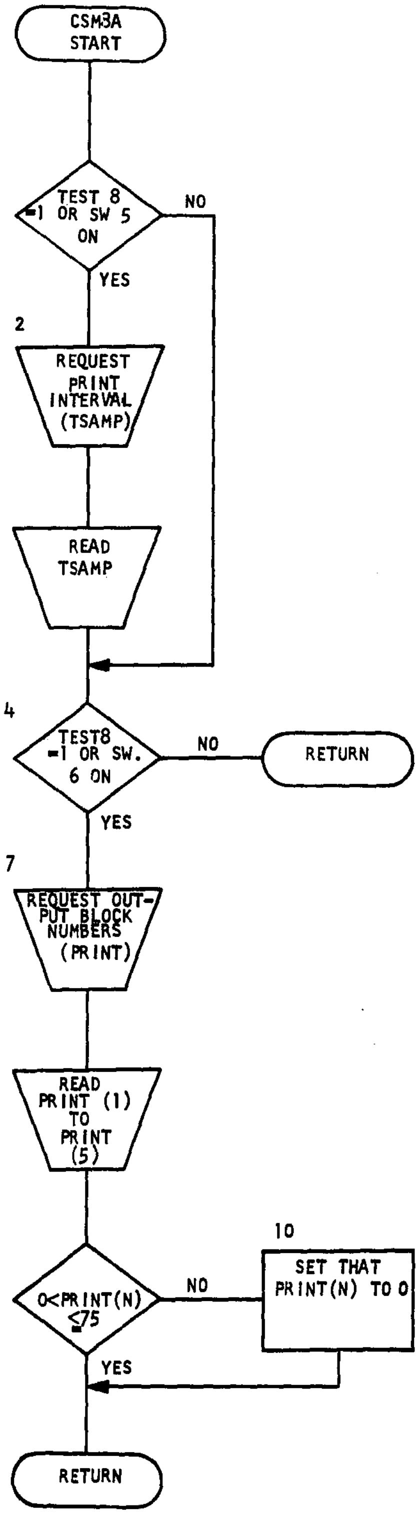

CSM10 START
SW15 ON (RESTART)
YES
60
RESTART SET-UP
READ STATUS FROM DISK
NO
1
NORMAL SET-UP
SET INITIAL CONDITIONS
50
CSM11
COMPUTE A HALF-STEP FOR THE CONFIG. (TEST5=1)
CSM9
MOVE PLOTTER PEN TO FIRST POINT AND LOWER IT.
CSM8
WRITE OUTPUT FOR T = 0
G1
100
START EXECUTION
UPDATE TIME AND INTEGRATORS FOR HALF-STEP TEST5=2
CSM11
COMPUTE FOR HALF-STEP
SECOND HALF-STEP
UPDATE TIME AND INTEGRATORS FOR FOR HALF-STEP TEST5=3
CSM11
COMPUTE FOR HALF - STEP
A3
A3
CSM9
PLOT POINT
YES
SW 0 ON
NO
130
YES
TEST5>3
NO
140
TIME TO PRINT?
NO
YES
150
CSM8
WRITE LINE OF OUTPUT
CSM9
PRINT-TIME MARKER ON PLOTTER OUTPUT
160
TEST5>3
YES
NO
170
YES
SW. 0 ON (INTR)
NO
190
T ≥ TTOT
NO
G1
YES
200
RETURN

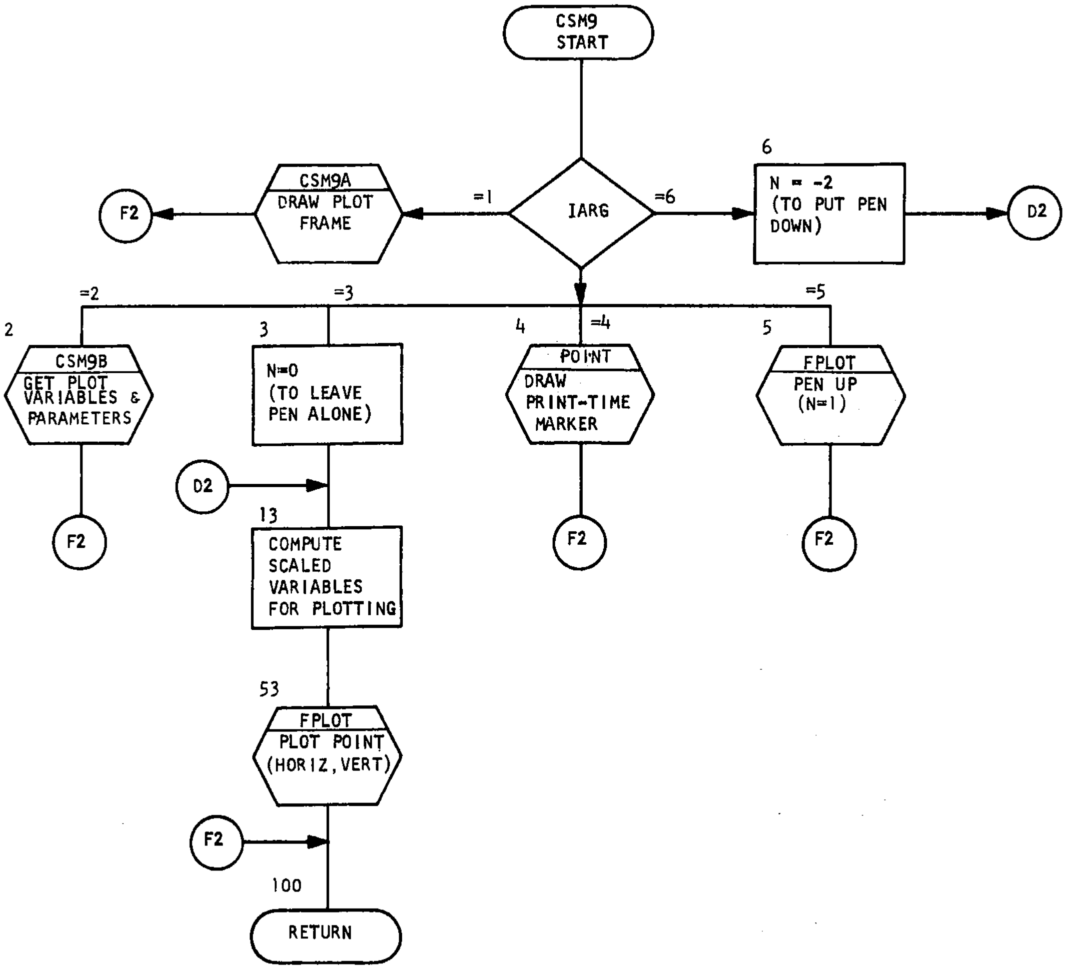

CSM11 START
NEXT=NCON (START WITH FIRST NON-CONSTANT ELEMENT.
C3
1000
I=BLOCK NO. OF NEXT ELEMENT.
DETERMINE TYPE OF ELEMENT
COMPUTE OUTPUT & STORE IN C ARRAY
RUN INTERRUPTED ?
NO
YES
SET TEST 5
RETURN
NEXT= TOT NO OF ELE?
YES
RETURN
NO
1100
NEXT = NEXT +1
C3

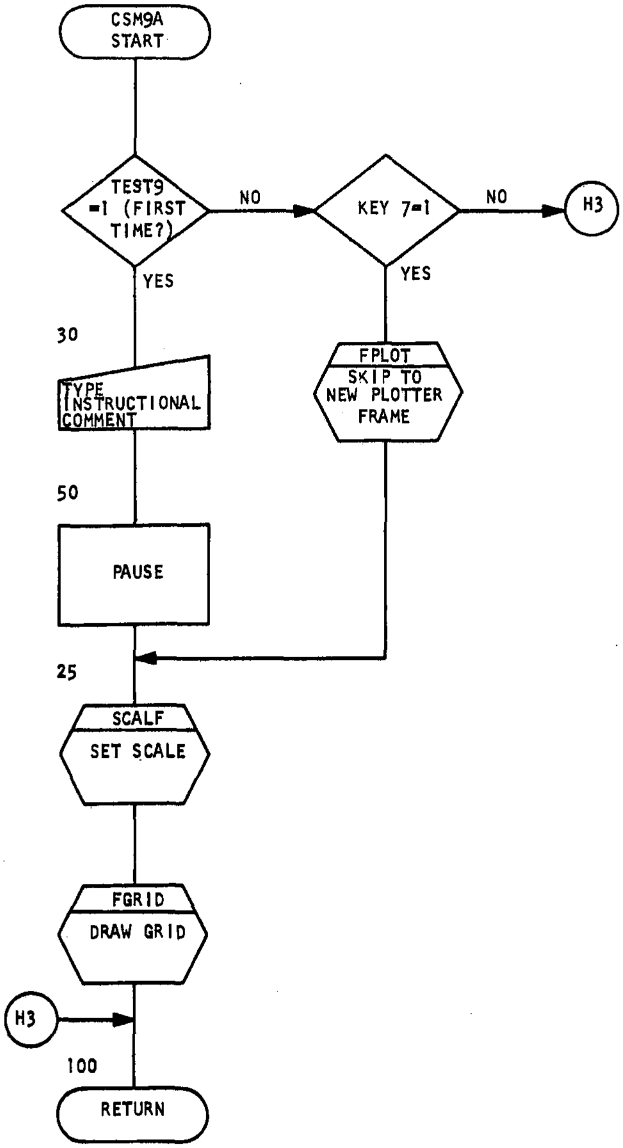

CSM12
START
WRITE
TERMINATION
MESSAGE
CSM9
RAISE
PLOTTER PEN
DATA
SWITCH
10
ON
OFF
PRINT
INSTRUCTIONAL
COMMENTS
50
PAUSE
READ DATA
SWITCHES
RETURN

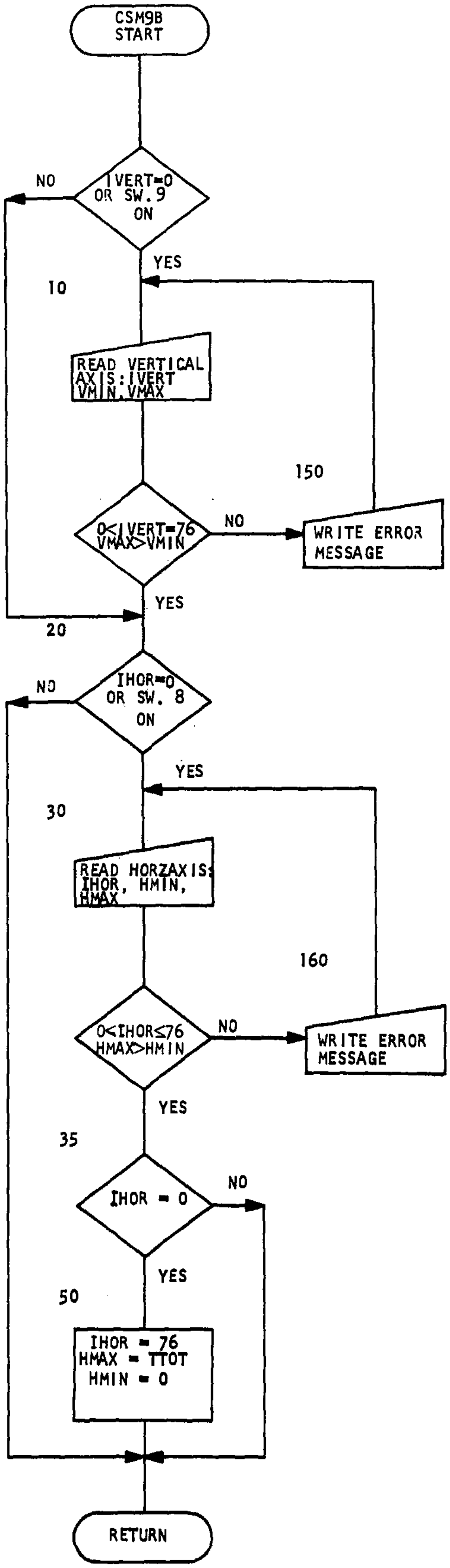

CSM13
START
C3
1
REQUEST
BLOCK
NUMBER
READ
BLOCK
NUMBER
20
2
RETURN
0
BLOCK
NUMBER
-
TYPE
ERROR
MESSAGE
C3
+
10
BLOCK
NUMBER
<76
NO
YES
30
PRINT
PRESENT
VALUE OF
BLOCK
SW. 13
ON
YES
C3
NO
RETURN

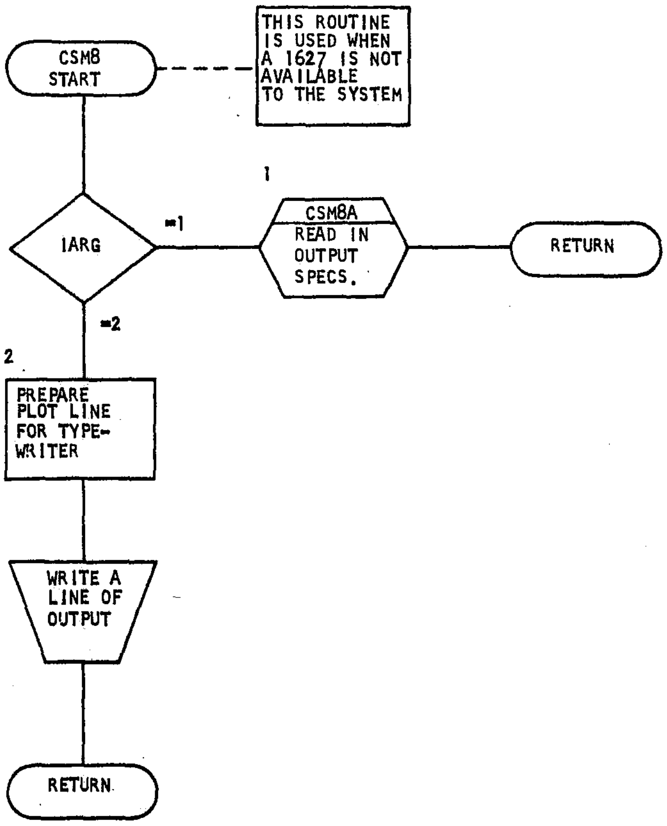

CSM8
START
IARG
=2
2
WRITE A
LINE OF
OUTPUT
=1
1
CSM8A
READ IN
OUTPUT
SPECIFI-
CATIONS
RETURN
RETURN

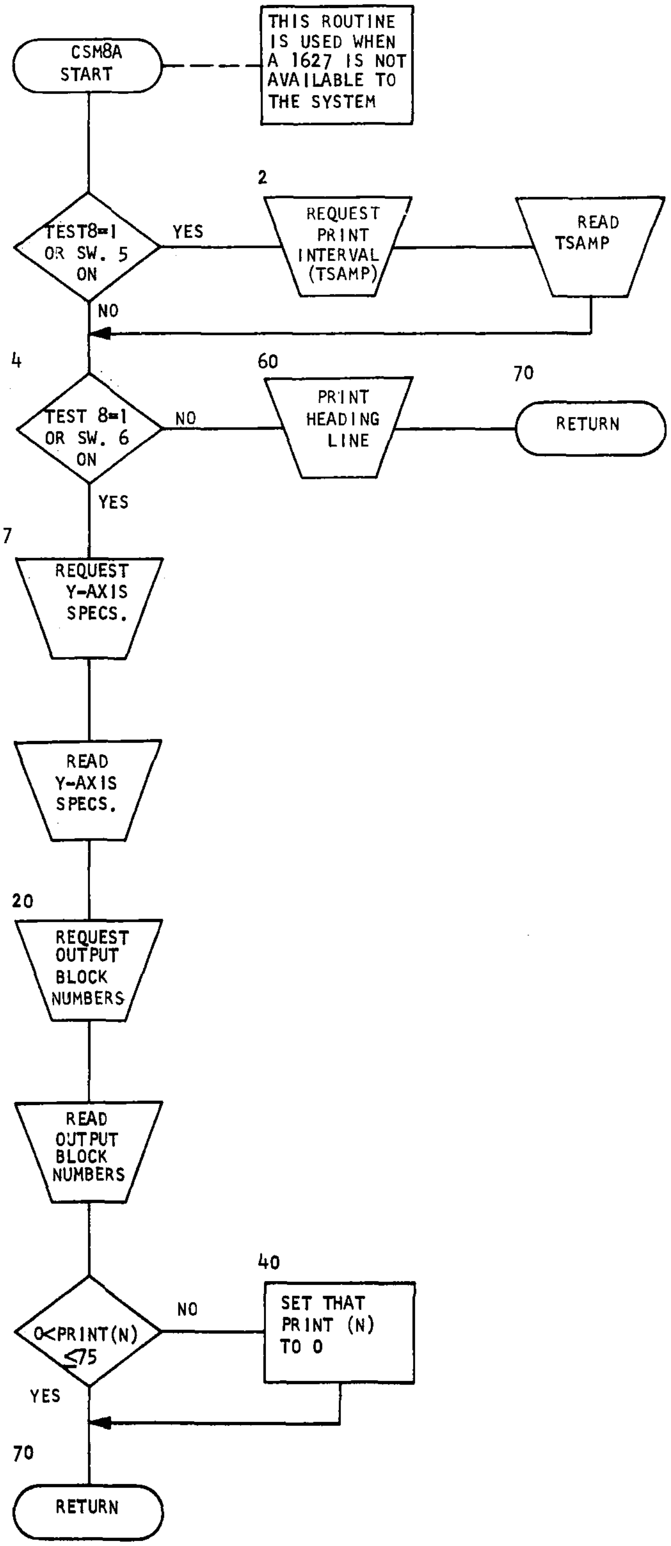

CSM3A START
TEST 8 =1 OR SW 5 ON
NO
YES
2
REQUEST PRINT INTERVAL (TSAMP)
READ TSAMP
4
TEST8 =1 OR SW. 6 ON
NO
RETURN
YES
7
REQUEST OUTPUT BLOCK NUMBERS (PRINT)
READ PRINT (1) TO PRINT (5)
0<PRINT(N) ≤75
NO
10
SET THAT PRINT(N) TO 0
YES
RETURN

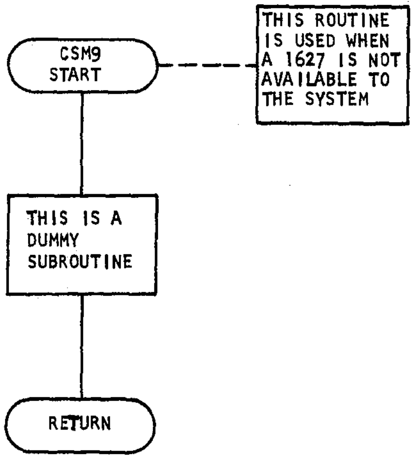

CSM9
START
IARG
=1
CSM9A
DRAW PLOT FRAME
F2
=6
6
N = -2
(TO PUT PEN DOWN)
D2
=2
=3
=4
=5
2
CSM9B
GET PLOT VARIABLES & PARAMETERS
F2
3
N=0
(TO LEAVE PEN ALONE)
D2
13
COMPUTE SCALED VARIABLES FOR PLOTTING
53
FPLOT
PLOT POINT (HORIZ,VERT)
F2
100
RETURN
4
POINT
DRAW PRINT-TIME MARKER
F2
5
FPLOT
PEN UP (N=1)
F2


CSM9A
START
TEST9
=1 (FIRST
TIME?)
NO
KEY 7=1
NO
H3
YES
YES
30
TYPE
INSTRUCTIONAL
COMMENT
FPLOT
SKIP TO
NEW PLOTTER
FRAME
50
PAUSE
25
SCALF
SET SCALE
FGRID
DRAW GRID
H3
100
RETURN


CSM9B START
IVERT=0 OR SW.9 ON
NO
YES
10
READ VERTICAL AXIS:IVERT VMIN,VMAX
150
0<IVERT=76 VMAX>VMIN
NO
WRITE ERROR MESSAGE
YES
20
IHOR=0 OR SW. 8 ON
NO
YES
30
READ HORZAXIS: IHOR, HMIN, HMAX
160
0<IHOR≤76 HMAX>HMIN
NO
WRITE ERROR MESSAGE
YES
35
IHOR = 0
NO
YES
50
IHOR = 76
HMAX = TTOT
HMIN = 0
RETURN


CSM8
START
THIS ROUTINE IS USED WHEN A 1627 IS NOT AVAILABLE TO THE SYSTEM
IARG
=1
1
CSM8A
READ IN OUTPUT SPECS.
RETURN
=2
2
PREPARE PLOT LINE FOR TYPE-WRITER
WRITE A LINE OF OUTPUT
RETURN


CSM8A START
THIS ROUTINE IS USED WHEN A 1627 IS NOT AVAILABLE TO THE SYSTEM
TEST8=1 OR SW. 5 ON
YES
2
REQUEST PRINT INTERVAL (TSAMP)
READ TSAMP
NO
4
TEST 8=1 OR SW. 6 ON
NO
60
PRINT HEADING LINE
70
RETURN
YES
7
REQUEST Y-AXIS SPECS.
READ Y-AXIS SPECS.
20
REQUEST OUTPUT BLOCK NUMBERS
READ OUTPUT BLOCK NUMBERS
0<PRINT(N) ≤75
NO
40
SET THAT PRINT (N) TO 0
YES
70
RETURN


CSM9
START
THIS ROUTINE
IS USED WHEN
A 1627 IS NOT
AVAILABLE TO
THE SYSTEM
THIS IS A
DUMMY
SUBROUTINE
RETURN

FLOWCHART NARRATIVES

The flowchart narratives are presented in the same order as the flowcharts.

CSMP

This is the main program of the IBM 1130 Continuous System Modeling Program. CSMP consists essentially of a series of tests on internal test values and the settings of console switches which determine a branch to one of the several subroutines. When the program is initiated, the first operation calls subroutine CSM0,which initializes all of the arrays in COMMON. The second function in the main program, CSMP, tests whether to branch to CSM1 for entry of configuration specifications or directly to CSM4 for entry of parameter specifications.

The subroutine CSM1 permits the entry or the modification of configuration specifications. Next subroutine CSM2 is called. It does a pre-sort test of the configuration statements to determine whether there are basic errors in the configuration. If errors in the configuration statements are noted in the subroutine CSM2, a branch is made back to CSM1 so that the user may correct these errors. If there are no errors in CSM2, the program then calls CSM3. CSM3 contains the sorting algorithm. It determines the proper order for evaluating the various block elements during each time interval. If the sorting is unsuccessful, a return is made for CSMP to again call CSM1, so that again the user has an opportunity to correct the configuration.

If the initial branch was to CSM4, or if the sorting operation was successful, then the program calls CSM4 for the entry of initial conditions or parameters associated with the various blocks. The program next checks to determine whether one or more Function Generators have been specified in the configuration. If so, there is a permissive branch to CSM5, the program which reads the Function Generator intercept points. Otherwise, CSMP tests the switch settings to determine whether or not to call subroutine CSM6. If called, it punches out an updated data card deck containing configuration, parameter, and Function Generator specifications. If CSM6 is not called, or after it is completed, the main program calls CSM7 for integration timing data and CSM8 and CSM9 for the plot and printing specifications.

The program finally calls CSM10 which handles the actual computation for the runs. At the completion of a run, or if it is interrupted by user intervention, there is a return from CSM10 to the main program. At that point, CSM12, the interrupt subroutine,is called. The user then specifies what to do next by means of the switches. On the basis of the switches that are set, there is a branch back to the appropriate portion of CSMP.

CSM0

This subroutine initializes data and control variables. The Type array is initialized by setting in the numerical equivalent for each of the symbols for the functional element types. In addition, this routine provides instructional messages which may be suppressed, if desired, by turning on Switch 10.

CSM1

This subroutine handles the entry or modification of configuration statements. The program has a branch at its input on internal switch TEST2. If it is equal to one, data entry is by card. If it is equal to two, data entry is by the keyboard. The latter would occur, for example, if the program had detected an earlier error. In CSM1 the data are read to determine the block number, the type of block, and the inputs. These data are checked for feasibility. If an error is found, error messages are provided. The statements are then stored in the appropriate locations of the array called MTRX.

CSM2

This subroutine performs a presort survey of the configuration statements. It counts to determine the number of Unit Delay elements, the number of Integrators, the number of Function Generators, and the number of Constants which have been specified. If any of these exceed the maximum permissible, error messages are provided. The maximum for Integrators and for Unit Delays is 25 blocks each. The maximum for Function Generators is three blocks. The program also checks to determine that each of the inputs required for a particular block has itself been specified as a block. For example, if Block 2 has Block 1 specified as one of its inputs, Block 1 must have been specified; otherwise, CSM2 will give an error message. Finally, this subroutine checks to determine if the number of Function Generators has been increased over the previous specification. If so, it sets a switch (TEST3) that will cause the calling of the Function Generator subroutine, CSM5, when operation is returned to the main program, CSMP.

CSM3

This subroutine performs the sorting algorithm. It checks to see that each block is capable of computation. At each time interval, there is a particular order in which computation is feasible. The outputs of Constant elements, Integrator elements, and Unit Delay elements are always available at each ΔT. With these available, it should be possible to compute the output of at least one other block. This block can then be added to the order list and array in the program. Its availability, in turn, should make at least one other computation permissible. The final result, assuming a successful sorting algorithm, is a list in which the block elements will be computed during each subsequent time interval. If there is a sort error in a configuration, such that a proper computational order is not permissible, an error message is provided. The main program, CSMP, then branches back to the configuration entry section CSM1.

CSM4

This subroutine handles the entry or modification of initial conditions and parameters associated with the specified block elements. Three parameters, at the most, can be associated with an individual block. In the case of memory elements, the first parameter is an initial condition. Entry of initial conditions and parameters can be either by card or via the keyboard. The selection depends on the value of the internal switch called TEST2. The program checks to see that the parameters specified are for a permissible block; that is, that the block numbers are between 1 and 75. If a block number has been specified negative or larger than 75, the user is provided a printer message requesting that he reenter the block number, initial conditions, and parameters properly via the keyboard.

CSM5

This subroutine handles data entry either from card or keyboard for each of the specified Function Generators. A maximum of three Function Generators is permissible. The program checks that the upper limit exceeds the value of the lower limit. If the limits have been specified incorrectly, the user is provided a printer message requesting that he enter the data properly via the keyboard. Error messages are provided if the user attempts to specify function intercepts for a block which is not a Function Generator.

CSM6

This subroutine is used to provide an updated or corrected data deck--a deck containing Configuration, Parameter, and Function Generator statements. This subroutine would be called by the user after he had made a number of modifications to his problem via the keyboard, or had entered his entire problem via the keyboard. This updated deck may be used for subsequent computer runs.

CSM7

This subroutine requests integration timing information. The console printer types:

(   ) INTEGRATION INTERVAL

This requests that the user type, within the parentheses, the value of the integration interval, ΔT, to be subsequently used in the second-order Runge-Kutta computation. Next the program prints:

(   ) TOTAL TIME

The user is expected to enter the duration of the run that is desired within the parentheses. If either of these entries is made improperly, e.g., if ΔT is specified as zero or a negative value, or if the total time is not in excess of the integration interval, an error message(s) is typed. The correct data can then be entered properly.

CSM10

This subroutine controls both the computation and the output during simulation runs. It contains the basic integration formula used with 1130 CSMP; this is a second-order Runge-Kutta method, also called a modified Euler integration method. This subroutine performs the integration calculations based upon values of the derivative vector DYDT which are obtained as needed by calling CSM11.

The integration procedure is initiated by first loading the integrator array, Y, with appropriate values of the state variables. For initiating runs at Time equal to zero, the initial condition stored as Parameter #1 of each Integrator is transferred to the block output array, C. For restarting runs from a previously saved condition (specified by use of Switches 14 and 15), this transfer is not required since the block outputs have been loaded in main program CSMP from data previously stored on the disk file. In either case, the starting values of all Integrators are next transferred from the C array to the corresponding locations in the Y array. As the integration proceeds, the outputs of the Y array are transferred to the corresponding C array locations after each integration half step. The error condition indicator, TEST 5, and the Console Switch 0 are checked at each step to determine if the run should be terminated. CSM8 and CSM9 are called as required for printing and plotting. If not prematurely terminated by an error condition or user intervention, the run terminates when the independent variable, Time, reaches the specified Total Time. Control is then returned to the main program, CSMP.

CSM11

This subroutine is called from CSM10 at every half step (T/2) to perform the computations required to evaluate the derivative vector. CSM11 contains the basic programming definition for the operation of each functional element. The program operates interpretively, first obtaining the block number of the next element to be evaluated from the ORDER array (which contains the block numbers of the configuration in properly sorted sequence as provided by CSM3). Next, from the MTRX matrix it determines the type of functional element and the block numbers of any inputs to the particular block. The routine then branches to the appropriate instructions for the type of functional element and computes the output of the block. This process is continued until all the blocks in the ORDER array have been serviced. Note that Integrators are serviced, not by computing their outputs, but rather, by computing their inputs which are stored in the DYDT array for subsequent use with the integration calculation; their outputs are set in CSM10. Upon completion, return is made to the calling program, CSM10.

CSM12

This subroutine prints a list of switch options for guidance of the user and requests that he set those appropriate for the options he chooses to exercise. Entry into this subroutine is caused by (1) TOTAL TIME being reached, (2) Quit element condition being reached, or (3) Switch 0 being turned ON. A pause occurs in order that the user may set the desired switches; after the switches have been set, he presses PROGRAM START. A return is then made to the main program, CSMP.

CSM13

This subroutine is provided primarily as a debugging aid. It permits the user to interrogate the value of various block outputs. He may continue to interrogate blocks so long as Switch 13 remains on. The subroutine causes the console printer to type:

```
OUTPUT INTERROGATION OPTION
(        ) BLOCK
```

The user then enters, within the parentheses, the block number for which he desires to know the value of the output variable. The program then prints:

```
OUTPUT OF BLOCK  nn  xxxxx.xxxx
```

where nn is the block number and xxxxx.xxxx is its value. If an incorrect block number (less than zero or greater than 75) is requested, the program prints:

```
ENTRY ERROR
```

and requests a new block number via the message:

```
(          ) BLOCK
```

If zeros or blanks are entered as the block number, then the program returns to the point of interruption. This subroutine permits the user to interrupt a run and check the block(s) whose performance he suspects is not as he intended.

CSM8

This subroutine has two functions. If internal switch IARG is equal to one, the subroutine calls subroutine CSM8A, which requests the initial information regarding printing. If IARG is equal to two, the subroutine provides a simple printer output of the status of the run. CSM8 is called during the run at every print interval.

CSM8A

This subroutine requests information regarding the block outputs to be printed. The printer types:

```
        (        ) PRINT INTERVAL
```

The user is expected to enter the desired print interval within the parentheses. The printer then types:

```
TIME OUTPUT (  ) OUTPUT(  ) OUTPUT(  ) OUTPUT(  ) OUTPUT(  )
```

The user is expected to enter the block numbers of the five block outputs desired within the five sets of parentheses. The program checks to ensure that the block numbers specified are between 1 and 75. If the block numbers(s) are improperly specified, the value zero will be printed during each print interval.

CSM9

This subroutine has six different functions depending upon the value of internal switch IARG. If IARG is equal to one, a call is made to CSM9A which moves the 1627 Plotter paper to a new position and draws a new frame. If IARG is equal to two, a call is made to CSM9B which asks the user to identify the plot variables and associated parameters. If IARG is equal to three, CSM9 plots the next point during the run; repeated entry provides the plot picture. If IARG is equal to four, a marker symbol is plotted on the output plot so that the user may later correlate the time of printing with the plot. This marker is drawn each time the tabular output occurs. If IARG is equal to five, the pen is raised. If IARG is equal to six, the pen is put down.

CSM9A

This subroutine sets up the scaling required by the plotter subroutines and draws a new frame or grid. On the initial entry to this subroutine, instructions are provided to the user on the preparation of the plotter.

CSM9B

This subroutine requests the identity of the dependent variable to be plotted on the vertical axis and the independent variable to be plotted on the horizontal axis, and the minimum and maximum values associated with each axis. The printer types:

```
(   ) BLOCK FOR Y-AXIS  (     ) MINIMUM VALUE (       ) MAXIMUM VALUE
```

The user is expected to enter the block number of the vertical axis variable, with its associated minimum and maximum values, within the parentheses. The printer then types:

```
(   ) BLOCK FOR X-AXIS (      ) MINIMUM VALUE (       ) MAXIMUM VALUE
```

The user is expected to enter the block number of the horizontal axis, with its associated minimum and maximum values, within the parentheses.

The program checks to ensure that the block numbers specified are between 1 and 75. If the block number(s) are improperly specified, an error message(s) is typed, and the correct data can then be entered.

CSM8 Alternate

This alternate subroutine is to be used in the place of CSM8 if the 1627 Plotter is unavailable. It provides a print-plot output on the console printer. Upon entry, a test is made to the value of the internal switch IARG. If the value is equal to one, a branch is made to subroutine CSM8A Alternate. If the value is equal to two, this subroutine prints the value of time, the three block output values previously specified in CSM8A Alternate, and a "print-plot" output of the fourth variable.

If the installation includes the 1132 Printer, the user might wish to modify this subroutine to output on the 1132 rather than the console printer. This would considerably reduce computer time for each simulation run.

CSM8A Alternate

This alternate subroutine is to be used in the place of CSM8A if the 1627 Plotter is unavailable. It is used in conjunction with CSM8 Alternate. This program requests the identification of the block outputs which are to be printed and the block which is designated for "print-plot". The printer types:

```
        (          ) PRINT INTERVAL
```

The user is expected to enter the desired print interval within the parentheses. The printer types:

```
(     ) BLOCK FOR Y-AXIS (      ) MINIMUM VALUE (      ) MAXIMUM VALUE
```

The user is expected to enter the block number for "print-plot" within the first set of parentheses. The user is expected to enter the minimum and maximum values of the variable to be plotted within the second and third sets of parentheses. The printer then types:

```
        TIME OUTPUT (    ) OUTPUT (    ) OUTPUT (    ) OUTPUT (    )
          nn  xxxx.xxxx      xxxx.xxxx
```

The user is expected to enter the numbers of the appropriate blocks within the three sets of parentheses. The nn indicates the block number of the variable to be both plotted and printed. The xxxx.xxxx indicate the minimum and maximum values of the plotted data.

CSM9 Alternate

This subroutine is to be used in place of CSM9, if the 1627 Plotter is unavailable. It is a dummy subroutine, as it provides no function.

SUB1 - SUB5

These dummy subroutines are provided with the program. As distributed, they do nothing. Whenever the user desires to use the "Special" elements feature of the program, he programs the desired operation in FORTRAN or Assembler Language. He deletes the previous SUBn subroutine from the disk, replacing it with his own. The calls to these routines are in CSM11 and are titled "SUB1, SUB2, ..., SUB5". The block types are "1, 2, ..., 5". Examples of possible special elements are found under "Definition of Special Elements" in the User's Manual.

PROGRAMMING NOTES AND SWITCH LISTINGS

Program switch settings and variable values are communicated between routines in the IBM 1130 Continuous System Modeling Program by means of the COMMON storage area. There are no arguments used in the calling sequence for the various routines. The COMMON area is divided into two portions: REAL area is named REALS and is subscripted from 1 to 587; INTEGER area is named INTS and is subscripted from 1 to 395. Locations in both arrays are given names more meaningful for programming by use of the EQUIVALENCE statement.

REALS Subscript	Normal Name	Alternate Name	Description of Use
1	BLANK		Constant of zero used as C(0).
2-76	C(1-75)		Current value of the output of blocks 1-75.
77	T	C(76)	Value of the independent variable time.
78	DT		Integration step interval.
79	DTS2		One half of integration step interval.
80	TTOT		Total time for run.
81-155	PAR(1-75, 1)	PAR1(1-75)	First parameter or initial condition of blocks 1-75.
156-230	PAR(1-75, 2)	PAR2(1-75)	Second parameter of blocks 1-75.
231-305	PAR(1-75, 3)	PAR3(1-75)	Third parameter of blocks 1-75.
306-340	F(1-3, 1-11)		The eleven intercept values for each of the three possible function generators.
341-365	Y(1-25)		The values of the integration output vector.

REALS Subscript	Normal Name	Alternate Name	Description of Use
366-390	DYDT(1-25)		The values of the integration input vector.
391	TSAMP		Print internal.
392	VDEL		Difference between minimum and maximum values of the variable to be plotted on the vertical scale.
393	HDEL		Difference between minimum and maximum values of the variable to be plotted on the horizontal scale.
394	VMIN		Minimum value of the variable to be plotted on the vertical scale.
395	HMIN		Minimum value of the variable to be plotted on the horizontal scale.

The INTS array is:

INTS Subscript	Normal Name	Alternate Name	Description of Use
1-75	MTRX(1-75, 1)	MTRX1(1-75)	Indication of block type for blocks 1-75.
76-150	MTRX(1-75, 2)	MTRX2(1-75)	Block number of first input to blocks 1-75.
151-225	MTRX(1-75, 3)	MTRX3(1-75)	Block number of second input to blocks 1-75.
226-300	MTRX(1-75, 4)	MTRX4(1-75)	Block number of third input to blocks 1-75.
301-375	MTRX(1-75, 5)	MTRX5(1-75)	The integrator or function generator number.
376	I		Index for the output variable.
377	J		Index for the first input variable.
378	K		Index for the second input variable.
379	L		Index for the third input variable.
380	KEY(1)	KEY1	Settings of Switches 1-15; 1 for on; 2 for off.
.	.	.	
.	.	.	
.	.	.	
394	KEY(15)	KEY15	
395	KEY(16)	KEY16	Setting of Switch 0; 1 for on; 2 for off.
396-420	INTG(1-25)		Block numbers of the integrator blocks (maximum is 25).

INTS Subscript	Normal Name	Alternate Name	Description of Use
421-423	NOFG(1-3)		Block numbers of the function generator blocks (maximum is 3).
424-448	DELAY(1-25)		Block numbers of the delay blocks (maximum is 25).
449-524	ORDER(1-76)		Computation order determined by the sorting algorithm.
525	TEST(1)	TEST1	Internal program switch 1. Equal to 1 after configuration specification or change, if pre-sort scan indicates an error, or if sort procedure is unsuccessful. Equal to 2 if pre-sort scan is successful, or if sort procedure is successful.
526	TEST(2)	TEST2	Internal program switch 2. Equal to 1 for card entry of data. Equal to 2 for keyboard entry of data.
527	TEST(3)	TEST3	Internal program switch 3. Equal to 1 for first time through set-up phase, or after any addition of a function generator element. Equal to 2 after the first time.
528	TEST(4)	TEST4	Internal program switch 4. Equal to 1 if there is no function generator. Equal to 2 if there are one or more function generators.

INTS Subscript	Normal Name	Alternate Name	Description of Use
529	TEST(5)	TEST5	Internal program switch 5. Equal to 1 upon first entry to CSM11 (T = TZERO). Equal to 2 for the first half step of each succeeding entry. Equal to 3 for the second half step of each succeeding entry. Equal to 4 if a run is terminated by an error in processing. Equal to 5 if run is terminated by Switch 0 during Wye element servicing. Equal to 6 if run is terminated by a Quit element.
530	TEST(6)	TEST6	Internal program switch 6. Unassigned.
531	TEST(7)	TEST7	Internal program switch 7. Equal to 1 before first pass through CSM7. Equal to 2 after first pass through CSM7.
532	TEST(8)	TEST8	Internal program switch 8. Equal to 1 before initial specification of printer variables. Equal to 2 after initial specifications of printer variables.
533	TEST(9)	TEST9	Internal program switch 9. Equal to 1 before initial specification of plotter variables. Equal to 2 after initial specification of plotter variables.
534	NLIST		The number of blocks used in the configuration.
535	PRINT(1)	K1	Block numbers for which the output is to be printed.
.	.	.	
.	.	.	
.	.	.	
539	PRINT(5)	K5	

INTS Subscript	Normal Name	Alternate Name	Description of Use
540	NCON		The number of constant elements.
541	NOD		The number of delay elements.
542	NEQ		The number of integrator elements.
543	NFG		The number of function generator elements.
544	IVERT		The block number specifying which output is to be the vertical axis of the plotter.
545	IHOR		The block number specifying which output is to be the horizontal axis of the plotter.
546	TYPE(1)		The numerical equivalent of the block types.
.	.		
.	.		
.	.		
585	TYPE(40)		
586	IR		Seed (internal starting number) for the random number generator.
587	IARG		Argument for the printing and plotting subroutines.

Flexibility in controlling the run is achieved by using the 16 Switches on the 1130 Console. The user manipulates these switches to stop and change the run at any desired time. Switch 0 is used to interrupt the run; the types of changes to be made are indicated by combinations of Switches 1-15. The individual switches are placed ON for indication as follows:

Sense Switch	Use
0	To interrupt the run.
1	To modify the configuration.
2	To modify parameters of initial condition.
3	To modify the function generator intercepts.
4	To modify the integration interval or run time.
5	To modify the print interval.
6	To modify the print variables.
7	To prepare for a new plot frame.
8	To modify plotter horizontal axis.
9	To modify plotter vertical axis.
10	To suppress console instructions.
11	To suppress typing of data read from cards.
12	To punch corrected data deck.
13	To interrogate block outputs.
14	To save status at interrupt time.
15	To restart at previous interrupt point.

The program variable (KEY(i)) is set to 1 if switch (i) is ON, and set to 2 if switch (i) is OFF. KEY(16) is used for Switch 0.

Legend

U - Used C - Changed or Set

I - Interrogated S - Set and Interrogated

Alternate Name	Normal Name	REALS Subscript	ROUTINE NAME																				
																					(Alt)		
			CSMP	CSMØ	CSM1	CSM2	CSM3	CSM4	CSM5	CSM6	CSM7	CSM8	CSM8A	CSM9	CSM9A	CSM9B	CSM1Ø	CSM11	CSM12	CSM13	CSM8	CSM8A	CSM9
	BLANK	1		C								U		U			U	U			U		
	C(1-75)	2-76	C	C					C			U		U			C	S		U	U		
	T	77		C								U					S				U		
	DT	78		C							S		U				U					U	
	DTS2	79		C							C						U	U					
	TTOT	80		C							S					U	I						
PAR1(1-75)	PAR(1-75, 1)	81-155		C				C	S	U							U	U					
PAR2(1-75)	PAR(1-75, 2)	156-230		C				C	S	U								S					
PAR3(1-75)	PAR(1-75, 3)	231-305		C				C		U								U					
	F(1-3, 1-11)	306-340		C					C	U								U					
	Y(1-75)	341-365		C													S	U					
	DYDT(1-75)	366-390		C													U	C					
	TSAMP	391		C									C				U					C	
	VDEL	392		C										U		S					U	S	
	HDEL	393		C										U		S							
	VMIN	394		C										U		C					U	S	
	HMIN	395		C										U		C							

Legend

U - Used C - Changed or Set
I - Interrogated S - Set and Interrogated

Alternate Name	Normal Name	INTS Subscript	ROUTINE NAME																		(Alt)		
			CSMP	CSMØ	CSM1	CSM2	CSM3	CSM4	CSM5	CSM6	CSM7	CSM8	CSM8A	CSM9	CSM9A	CSM9B	CSM1Ø	CSM11	CSM12	CSM13	CSM8	CSM8A	CSM9
MTRX1(1-75)	MTRX(1-75, 1)	1-75		C	S	C	S		S	U								U					
MTRX2(1-75)	MTRX(1-75, 2)	76-150		C	S	U	U			U								U					
MTRX3(1-75)	MTRX(1-75, 3)	151-225		C	S	U	U			U								U					
MTRX4(1-75)	MTRX(1-75, 4)	226-300		C	S	U	U			U								U					
MTRX5(1-75)	MTRX(1-75, 5)	301-375		C	S	C			C									S					
	I	376		C	S			S										S					
	J	377		C	S													S					
	K	378		C	S													S					
	L	379		C	S													S					
KEY1	KEY(1)	380	I	C	I														S				
KEY2	KEY(2)	381	I	C				I											S				
KEY3	KEY(3)	382	I	C					I										S				
KEY4	KEY(4)	383	I	C															S				
KEY5	KEY(5)	384		C									I						S			I	
KEY6	KEY(6)	385		C									I						S			I	
KEY7	KEY(7)	386		C											I				S				
KEY8	KEY(8)	387		C												I			S				
KEY9	KEY(9)	388		C												I			S				
KEY10	KEY(10)	389		S															S				

Legend

U - Used C - Changed or Set
I - Interrogated S - Set and Interrogated

ROUTINE NAME

Alternate Name	Normal Name	INTS Subscript	CSMP	CSMØ	CSM1	CSM2	CSM3	CSM4	CSM5	CSM6	CSM7	CSM8	CSM8A	CSM9	CSM9A	CSM9B	CSM1Ø	CSM11	CSM12	CSM13	(Alt) CSM8	(Alt) CSM8A	(Alt) CSM9
KEY11	KEY(11)	390		C	I			I	I										S				
KEY12	KEY(12)	391	I	C															S				
KEY13	KEY(13)	392	I	C															S	I			
KEY14	KEY(14)	393	I	C															S				
KEY15	KEY(15)	394	I	C													I		S				
KEY16	KEY(16)	395	I	C													I	I					
	INTG(1-25)	396-420		C		C	I										U	U					
	NOFG(1-3)	421-423		C					S	I													
	DELAY(1-25)	424-448		C		C	I																
	ORDER(1-76)	449-524		C		C	S										U	U					
TEST1	TEST(1)	525	I	C	S	C	C																
TEST2	TEST(2)	526		C	S	C	C	S	S														
TEST3	TEST(3)	527	C	C		C																	
TEST4	TEST(4)	528	I	C		C				I													
TEST5	TEST(5)	529		C													S	S					
TEST6	TEST(6)	530		C																			
TEST7	TEST(7)	531	I	C							S												
TEST8	TEST(8)	532		C									S									S	
TEST9	TEST(9)	533		C											S				I				
	NLIST	534		C			S										U	I					

Legend

U - Used C - Changed or Set
I - Interrogated S - Set and Interrogated

Alternate Name	Normal Name	INTS Subscript	ROUTINE																				
																					(Alt)		
			CSMP	CSMØ	CSM1	CSM2	CSM3	CSM4	CSM5	CSM6	CSM7	CSM8	CSM8A	CSM9	CSM9A	CSM9B	CSM1Ø	CSM11	CSM12	CSM13	CSM8	CSM8A	CSM9
K1...K5	PRINT(1-5)	535-539		C								U	S								U	S	
	NCON	540		C		C	I											U					
	NOD	541		C		S	I																
	NEQ	542		C		S	I										U	U					
	NFG	543		C		C													I				
	IVERT	544		C										U		S							
	IHOR	545		C										U		S							
	TYPE(1-40)	546-585		C	U					U													
	IR	586		C													C	S					
	IARG	587	C	C								I		I			C				I		

PROGRAM LISTINGS

This section contains the source program listings for the entire 1130 Continuous System Modeling Program. They are listed by decks in the same order that the flowcharts and flowchart narratives are listed.

```
// FOR                                                                  CSAA0010
*IOCS(CARD,TYPEWRITER,KEYBOARD,DISK)                                    CSAA0020
*ONE WORD INTEGERS                                                      CSAA0030
*NAME CSMP                                                              CSAA0040
C        CONTINUOUS SYSTEM MODELING PROGRAM                             CSAA0050
C      ***** MARCH 2, 1967 *****                 VERSION 1 MOD 1        CSAA0060
C                                                                       CSAA0070
C        THIS IS THE MAIN PROGRAM FOR CSMP                              CSAA0080
C                                                                       CSAA0090
      REAL    REALS(395)                                                CSAA0100
      INTEGER INTS(587), TEST1, TEST3, TEST4,TEST7                      CSAA0110
      DIMENSION C(76)                                                   CSAA0120
      COMMON  REALS, INTS                                               CSAA0130
      EQUIVALENCE  ( INTS(380),   KEY1      ) , ( REALS(  2),     C(1)  )CSAA0140
      EQUIVALENCE  ( INTS(381),   KEY2      )                           CSAA0150
      EQUIVALENCE  ( INTS(382),   KEY3      )                           CSAA0160
      EQUIVALENCE  ( INTS(383),   KEY4      )                           CSAA0170
      EQUIVALENCE  ( INTS(391),   KEY12     )                           CSAA0180
      EQUIVALENCE  ( INTS(392),   KEY13     )                           CSAA0190
      EQUIVALENCE  ( INTS(393),   KEY14     )                           CSAA0200
      EQUIVALENCE  ( INTS(394),   KEY15     )                           CSAA0210
      EQUIVALENCE  ( INTS(395),   KEY16     )                           CSAA0220
      EQUIVALENCE  ( INTS(525),  TEST1      )                           CSAA0230
      EQUIVALENCE  ( INTS(527),  TEST3      )                           CSAA0240
      EQUIVALENCE  ( INTS(528),  TEST4      )                           CSAA0250
      EQUIVALENCE  ( INTS(531),  TEST7      )                           CSAA0260
      EQUIVALENCE  ( INTS(587),  IARG       )                           CSAA0270
C                                                                       CSAA0280
C        THIS DEFINES DISK FILE 1 FOR SAVING STATUS                     CSAA0290
      DEFINE FILE 1 ( 1, 152, U, NXT )                                  CSAA0300
C                                                                       CSAA0310
C         INITIALIZATION SUBROUTINE                                     CSAA0320
      CALL CSM0                                                         CSAA0330
C                                                                       CSAA0340
C        CONFIGURATION SECTION                                          CSAA0350
C        PROGRAM WILL NOT BRANCH BEYOND THE CONFIGURATION SECTION       CSAA0360
C        UNTIL SUCCESSFUL SORT TEST IS ACHIEVED AT WHICH TIME THE       CSAA0370
C        SWITCH TEST1   IS SET TO 2                                     CSAA0380
C                                                                       CSAA0390
   10 GO TO ( 12, 11), TEST1                                            CSAA0400
   11 GO TO ( 12,100), KEY1                                             CSAA0410
C         GET CONFIG. SPECS                                             CSAA0420
   12 CALL CSM1                                                         CSAA0430
C         PREPARE FOR SORT                                              CSAA0440
      CALL CSM2                                                         CSAA0450
C            TEST1 = 1 IF PRE-SORT SCAN INDICATES ERROR                 CSAA0460
C            TEST1 = 2 IF PRE-SORT SCAN IS SUCCESSFUL                   CSAA0470
      GO TO (12,13), TEST1                                              CSAA0480
C         SORT                                                          CSAA0490
   13 CALL CSM3                                                         CSAA0500
C        TEST FOR SUCCESSFUL SORT                                       CSAA0510
C            TEST1 = 1 IF SORT PROCEDURE IS UNSUCCESSFUL                CSAA0520
C            TEST1 = 2 IF SORT PROCEDURE IS SUCCESSFUL                  CSAA0530
      GO TO ( 12,100) , TEST1                                           CSAA0540
  100 CONTINUE                                                          CSAA0550
C                                                                       CSAA0560
C         SET-UP SECTION                                                CSAA0570
C                                                                       CSAA0580
C         PARAMETERS AND INITIAL CONDITIONS                             CSAA0590
      GO TO (110,109), TEST3                                            CSAA0600
  109 GO TO (110,115), KEY2                                             CSAA0610
```

```
 110 CALL CSM4                                                          CSAA0620
 115 CONTINUE                                                           CSAA0630
C         FUNCTION GENERATORS     ******                                CSAA0640
      GO TO (121,118), TEST4                                            CSAA0650
 118 GO TO (120,119), TEST3                                             CSAA0660
 119 GO TO (120,121),KEY3                                               CSAA0670
 120 CALL CSM5                                                          CSAA0680
 121 CONTINUE                                                           CSAA0690
C        SET TEST3=2 TO INDICATE COMPLETION OF INITIAL SPECIFICATION    CSAA0700
C        OF CONFIGURATION, PARAMETERS, AND FUNCTION GENERATOR INTERCEPTSCSAA0710
             TEST3 = 2                                                  CSAA0720
C         (INTERRUPT POINT)                                             CSAA0730
      CALL DATSW (0,KEY16)                                              CSAA0740
      GO TO (225,125), KEY16                                            CSAA0750
 125 CONTINUE                                                           CSAA0760
C         PUNCH CARDS   ****************                                CSAA0770
      GO TO (127,126), KEY12                                            CSAA0780
 126 CALL DATSW(12,KEY12    )                                           CSAA0790
      GO TO (127,128), KEY12                                            CSAA0800
 127 CALL CSM6                                                          CSAA0810
C         (INTERRUPT POINT)                                             CSAA0820
      CALL DATSW (0,KEY16)                                              CSAA0830
      GO TO (225,128), KEY16                                            CSAA0840
 128 CONTINUE                                                           CSAA0850
C         TIMING   *********************                                CSAA0860
      GO TO (130,129), TEST7                                            CSAA0870
C        TEST7 = 1 UNTIL FIRST TIME THROUGH CSM7                        CSAA0880
C        TEST7 = 2 AFTER FIRST TIME THROUGH CSM7                        CSAA0890
 129 GO TO (130,135), KEY4                                              CSAA0900
 130 CALL CSM7                                                          CSAA0910
C         (INTERRUPT POINT)                                             CSAA0920
      CALL DATSW (0,KEY16)                                              CSAA0930
      GO TO (225,135), KEY16                                            CSAA0940
 135 CONTINUE                                                           CSAA0950
C         PLOT SPECS    ****************                                CSAA0960
            IARG = 2                                                    CSAA0970
 150 CALL CSM9                                                          CSAA0980
C         (INTERRUPT POINT)                                             CSAA0990
      CALL DATSW (0,KEY16)                                              CSAA1000
      GO TO (225,155), KEY16                                            CSAA1010
 155 CONTINUE                                                           CSAA1020
C         NEW PLOT FRAME     ***********                                CSAA1030
            IARG = 1                                                    CSAA1040
 160 CALL CSM9                                                          CSAA1050
 165 CONTINUE                                                           CSAA1060
C         OUTPUT SPECS  ****************                                CSAA1070
            IARG = 1                                                    CSAA1080
 170 CALL CSM8                                                          CSAA1090
C                                                                       CSAA1100
C         (INTERRUPT POINT)                                             CSAA1110
      CALL DATSW (0,KEY16)                                              CSAA1120
      GO TO (225,200), KEY16                                            CSAA1130
 200 CONTINUE                                                           CSAA1140
C                                                                       CSAA1150
      GO TO (210,220) , KEY15                                           CSAA1160
 210 CONTINUE                                                           CSAA1170
      READ (1-1) (C(N), N=1,76 )                                        CSAA1180
C         COMPUTE SECTION    ***********                                CSAA1190
 220 CALL CSM10                                                         CSAA1200
C                                                                       CSAA1210
C         CALLS INTERRUPT SUBROUTINE FOR NEW SENSE SWITCH SETTINGS      CSAA1220
```

```
  225 CALL CSM12                                                        CSAA1230
      GO TO (230,240) ,KEY13                                            CSAA1240
  230 CALL CSM13                                                        CSAA1250
      GO TO 225                                                         CSAA1260
  240 CONTINUE                                                          CSAA1270
      GO TO (250,10),KEY14                                              CSAA1280
C                                                                       CSAA1290
C             SAVE STATUS                                               CSAA1300
  250 CONTINUE                                                          CSAA1310
      WRITE (1-1) (C(N), N=1,76)                                        CSAA1320
      GO TO 10                                                          CSAA1330
      END                                                               CSAA1340
// DUP                                                                  CSAA1350
*STORE      WS  UA  CSMP                                                CSAA1360
// JOB                                                                  CSAA1370
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSMO
C       ***** APRIL 14, 1966 *****
C         ITS FUNCTION IS TO INITIALIZE DATA AND CONTROL VARIABLES AND TO
C         PROVIDE INSTRUCTIONAL MESSAGES FOR THE INEXPERIENCED USER
C
      REAL    REALS(395)
      INTEGER INTS(587),TYPE(40),TEST(9)
      DIMENSION KEY(16)
      COMMON  REALS, INTS
C
      EQUIVALENCE  ( INTS(380),   KEY(1)    )
      EQUIVALENCE  ( INTS(389),   KEY10     )
      EQUIVALENCE  ( INTS(525),  TEST(1)    )
      EQUIVALENCE  ( INTS(546),  TYPE(1)    )
C
      WRITE (1,22)
   22 FORMAT (26X,34HCONTINUOUS SYSTEM MODELING PROGRAM ,/
   221   18X,51HA DIGITAL ANALOG SIMULATOR PROGRAM FOR THE IBM 1130  //)
      DO 4 N=1,587
    4 INTS(N) = 0
      DO 5 N=1,395
    5 REALS(N) = 0.0
      DO 6 N=1,16
    6 KEY(N) = 2
      DO 7       N = 1,9
    7      TEST(N) = 1
C
C INSERT KEY SETTINGS IF NECESSARY
C          INSTRUCTIONAL MESSAGES
      CALL DATSW(10,KEY10    )
      GO TO ( 50,  1), KEY10
    1 WRITE (1,101)
  101 FORMAT ( - INSTRUCTIONAL COMMENTS MAY BE SUPPRESSED AT ANY TIME BY
     1 TURNING ON SWITCH 10- )
      CALL DATSW(10,KEY10    )
      GO TO ( 50,  2), KEY10
    2 WRITE (1,102)
  102 FORMAT ( - TURN ON SWITCH 1 TO ENTER OR MODIFY CONFIGURATION STATE
     1MENTS VIA THE KEYBOARD- )
      CALL DATSW(10,KEY10    )
      GO TO ( 50,  3), KEY10
    3 WRITE (1,103)
  103 FORMAT (- TURN ON SWITCH 2 TO ENTER OR MODIFY INITIAL CONDITIONS O
     1R ELEMENT PARAMETERS VIA THE KEYBOARD- )
      CALL DATSW(10,KEY10    )
      GO TO ( 50, 10), KEY10
   10 WRITE (1,110)
  110 FORMAT (- TURN ON SWITCH 3 TO ENTER OR MODIFY FUNCTION GENERATOR I
     1NTERCEPTS VIA THE KEYBOARD- ,//)
C
C          SET UP TYPE ARRAY
   50 CONTINUE
C        A         UNASSIGNED
C        B         BANG-BANG
      TYPE(02)  =  -15808
C        C         UNASSIGNED
C        D         DEAD SPACE
      TYPE(04)  =  -15296
C        E         UNASSIGNED
```

```
C        F         FUNCTION GENERATOR                                   CSAB0620
      TYPE(06)  =  -14784                                               CSAB0630
C        G         GAIN                                                 CSAB0640
      TYPE(07)  =  -14528                                               CSAB0650
C        H         HALF POWER (SQUARE ROOT)                             CSAB0660
      TYPE(08)  =  -14272                                               CSAB0670
C        I         INTEGRATOR   (MAXIMUM 25 ELEMENTS)                   CSAB0680
      TYPE(09)  =  -14016                                               CSAB0690
C        J         JITTER (RANDOM NUMBER GENERATOR BETWEEN + AND - 1)   CSAB0700
      TYPE(10)  =  -11968                                               CSAB0710
C        K         CONSTANT                                             CSAB0720
      TYPE(11)  =  -11712                                               CSAB0730
C        L         LIMITER                                              CSAB0740
      TYPE(12)  =  -11456                                               CSAB0750
C        M         MAGNITUDE  (ABSOLUTE VALUE)                          CSAB0760
      TYPE(13)  =  -11200                                               CSAB0770
C        N         NEGATIVE CLIPPER                                     CSAB0780
      TYPE(14)  =  -10944                                               CSAB0790
C        O         OFFSET                                               CSAB0800
      TYPE(15)  =  -10688                                               CSAB0810
C        P         POSITIVE CLIPPER                                     CSAB0820
      TYPE(16)  =  -10432                                               CSAB0830
C        Q         QUIT                                                 CSAB0840
      TYPE(17)  =  -10176                                               CSAB0850
C        R         RELAY                                                CSAB0860
      TYPE(18)  =  - 9920                                               CSAB0870
C        S         UNASSIGNED                                           CSAB0880
C        T         TIMING PULSE GENERATOR                               CSAB0890
      TYPE(20)  =  - 7360                                               CSAB0900
C        U         UNIT DELAY   (MAXIMUM 25 ELEMENTS)                   CSAB0910
      TYPE(21)  =  - 7104                                               CSAB0920
C        V         VACUOUS  (USED IN CONJUNCTION WITH WYE ELEMENT)      CSAB0930
      TYPE(22)  =  - 6848                                               CSAB0940
C        W         WEIGHTED SUMMER                                      CSAB0950
      TYPE(23)  =  - 6592                                               CSAB0960
C        X         MULTIPLIER                                           CSAB0970
      TYPE(24)  =  - 6336                                               CSAB0980
C        Y         WYE                 IMPLICIT FUNCTION TEST           CSAB0990
      TYPE(25)  =  - 6080                                               CSAB1000
C        Z         ZERO ORDER HOLD                                      CSAB1010
      TYPE(26)  =  - 5824                                               CSAB1020
C      PLUS        SUMMER          ( THREE INPUTS PERMITTED)            CSAB1030
      TYPE(27)  =   20032                                               CSAB1040
C        -         SIGN INVERTER  (LIMITED TO ONE INPUT CONNECTION)     CSAB1050
      TYPE(28)  =   24640                                               CSAB1060
C        /         DIVIDER                                              CSAB1070
      TYPE(29)  =   24896                                               CSAB1080
C  AMPERSAND       SUMMER          ( THREE INPUTS PERMITTED)            CSAB1090
      TYPE(30)  =  20544                                                CSAB1100
C        1         SPECIAL ELEMENT NUMBER 1                             CSAB1110
      TYPE(31)  =  - 3776                                               CSAB1120
C        2         SPECIAL ELEMENT NUMBER 2                             CSAB1130
      TYPE(32)  =  - 3520                                               CSAB1140
C        3         SPECIAL ELEMENT NUMBER 3                             CSAB1150
      TYPE(33)  =  - 3264                                               CSAB1160
C        4         SPECIAL ELEMENT NUMBER 4                             CSAB1170
      TYPE(34)  =  - 3008                                               CSAB1180
C        5         SPECIAL ELEMENT NUMBER 5                             CSAB1190
      TYPE(35)  =  - 2752                                               CSAB1200
C                  ASSIGNED BLANK                                       CSAB1210
      TYPE(40)  =   16448                                               CSAB1220
      RETURN                                                            CSAB1230
      END                                                               CSAB1240
// DUP                                                                  CSAB1250
*STORE      WS  UA  CSMO                                                CSAB1260
// JOB                                                                  CSAB1270
// FOR                                                                  CSAC0010
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS                                                      CSAC0020
      SUBROUTINE CSM1                                                   CSAC0030
C        ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2         CSAC0040
C         CONFIGURATION SPECIFICATIONS                                  CSAC0050
C                                                                       CSAC0060
      REAL     REALS(395)                                               CSAC0070
      INTEGER INTS(587), TYPE(40), TEST1, TEST2                         CSAC0080
      DIMENSION  MTRX(75,5), SY(4)                                      CSAC0090
      COMMON  REALS, INTS                                               CSAC0100
      EQUIVALENCE  ( INTS(  1),  MTRX(1,1) )                            CSAC0110
      EQUIVALENCE  ( INTS(376),   I        )                            CSAC0120
      EQUIVALENCE  ( INTS(377),   J        )                            CSAC0130
      EQUIVALENCE  ( INTS(378),   K        )                            CSAC0140
      EQUIVALENCE  ( INTS(379),   L        )                            CSAC0150
      EQUIVALENCE  ( INTS(380),   KEY1     )                            CSAC0160
      EQUIVALENCE  ( INTS(390),   KEY11    )                            CSAC0170
      EQUIVALENCE  ( INTS(525),  TEST1     )                            CSAC0180
      EQUIVALENCE  ( INTS(526),  TEST2     )                            CSAC0190
      EQUIVALENCE  ( INTS(546),  TYPE(1)   )                            CSAC0200
C                                                                       CSAC0210
      GO TO ( 5,70), TEST2                                              CSAC0220
C        SWITCH TEST2   = 1 FOR CARD ENTRY                              CSAC0230
C        SWITCH TEST2   = 2 FOR KEYBOARD ENTRY                          CSAC0240
    5 WRITE (1,201)                                                     CSAC0250
C                                                                       CSAC0260
C         CARD ENTRY SECTION                                            CSAC0270
C                                                                       CSAC0280
   10 READ (2,202) SY(1), SY(2), SY(3), SY(4), I, ITYPE, J, K, L        CSAC0290
      IF (I)  150, 50, 100                                              CSAC0300
C                                                                       CSAC0310
C         PRINT CARD                                                    CSAC0320
C                                                                       CSAC0330
   20 CALL DATSW (11, KEY11    )                                        CSAC0340
      GO TO (10,30), KEY11                                              CSAC0350
   30 WRITE (1,202) (SY(N),N=1,4), I, ITYPE, J, K, L                    CSAC0360
      GO TO 10                                                          CSAC0370
C                                                                       CSAC0380
C         KEYBOARD TEST SECTION                                         CSAC0390
C                                                                       CSAC0400
   50 CONTINUE                                                          CSAC0410
      TEST2 = 2                                                         CSAC0420
C                                                                       CSAC0430
C         TEST CONSOLE ENTRY SWITCH 1                                   CSAC0440
   60 CALL DATSW(1,KEY1  )                                              CSAC0450
      GO TO (70,200), KEY1                                              CSAC0460
C                                                                       CSAC0470
C         KEYBOARD ENTRY SECTION                                        CSAC0480
C                                                                       CSAC0490
   70 WRITE (1,203)                                                     CSAC0500
      READ (6,204) I,ITYPE,J,K,L                                        CSAC0510
      IF (I)   150, 75,100                                              CSAC0520
   75 GO TO ( 10, 60 ) , TEST2                                          CSAC0530
C        DECODE AND TEST CONFIGURATION STATEMENTS                       CSAC0540
C                                                                       CSAC0550
  100 CONTINUE                                                          CSAC0560
      IF ( I - 75 )   111, 111, 150                                     CSAC0570
  111 IF ( IABS(J) - 76 )    112, 112, 150                              CSAC0580
  112 IF ( IABS(K) - 76 )     113, 113, 150                             CSAC0590
  113 IF ( IABS(L) - 76 )     120, 120, 150                             CSAC0600
C        PROGRAM LIBRARY LIMITED TO 39 ELEMENT TYPES                    CSAC0610
C        ELEMENT TYPE 40 CORRESPONDS TO BLANK TYPE ENTRY                CSAC0620
  120 DO 125    ITEST = 1, 40                                           CSAC0630
       IF ( ITYPE - TYPE(ITEST) ) 125, 130, 125                         CSAC0640
  125 CONTINUE                                                          CSAC0650
      GO TO 160                                                         CSAC0660
  130 GO TO (150,132,150,132,150,132,132,132,143,131,131,132,132,132,   CSAC0670
 1301        132,132,133,143,150,132,132,143,143,133,133,133,146,132,   CSAC0680
 1302        133,135,143,143,143,143,143,150,150,150,150,134),ITEST     CSAC0690
  131 IF ( J )   132,132,150                                            CSAC0700
  132 IF ( K )   133,133,150                                            CSAC0710
  133 IF ( L )   143,143,150                                            CSAC0720
  134    MTRX(I,1) = 0                                                  CSAC0730
         MTRX(I,5) = 0                                                  CSAC0740
      WRITE (1,207)                                                     CSAC0750
      GO TO 149                                                         CSAC0760
  135        ITEST = 27                                                 CSAC0770
             ITYPE = 20032                                              CSAC0780
      GO TO 146                                                         CSAC0790
  143 IF ( J )              150,144,144                                 CSAC0800
  144 IF ( K )              150,145,145                                 CSAC0810
  145 IF ( L )              150,146,146                                 CSAC0820
C        STORE CONFIGURATION FOR CORRECT STATEMENT                      CSAC0830
  146 IF ( MTRX(I,1) )   148,148,147                                    CSAC0840
  147 WRITE (1,207)                                                     CSAC0850
  148    MTRX(I,1) = ITEST                                              CSAC0860
         MTRX(I,2) = J                                                  CSAC0870
         MTRX(I,3) = K                                                  CSAC0880
         MTRX(I,4) = L                                                  CSAC0890
  149 GO TO (20,60) , TEST2                                             CSAC0900
C                                                                       CSAC0910
C         ERROR SECTION                                                 CSAC0920
  150 WRITE (1,205) I                                                   CSAC0930
  151 GO TO (155, 70), TEST2                                            CSAC0940
  155 GO TO (70, 156), KEY11                                            CSAC0950
  156 WRITE (1,202) (SY(N),N=1,4), I, ITYPE, J, K, L                    CSAC0960
      GO TO 70                                                          CSAC0970
C        ERROR MESSAGE IF TYPE IS NOT FOUND IN THE LIBRARY              CSAC0980
  160 WRITE (1,206) I                                                   CSAC0990
      GO TO 151                                                         CSAC1000
C                                                                       CSAC1010
C         EXIT FROM THIS SUBROUTINE                                     CSAC1020
  200 CONTINUE                                                          CSAC1030
      TEST1 = 1                                                         CSAC1040
      TEST2 = 1                                                         CSAC1050
      RETURN                                                            CSAC1060
C                                                                       CSAC1070
  201 FORMAT(/26X,27HCONFIGURATION SPECIFICATION,//11HOUTPUT NAME, 5X,  CSAC1080
 2011              46HBLOCK      TYPE     INPUT 1    INPUT 2    INPUT 3 )CSAC1090
  202 FORMAT (4A4,2X,I2,9X,A1,3(7X,I3) )                                CSAC1100
  203 FORMAT (17X,4H(   ),7X,3H( ), 3(5X,5H(    )))                     CSAC1110
  204 FORMAT (18X,I2,9X,A1,3(7X,I3) )                                   CSAC1120
  205 FORMAT ( 28HTHE SPECIFICATION FOR BLOCK ,I2,13H IS INCORRECT  )   CSAC1130
  206 FORMAT ('ILLEGAL SYMBOL SPECIFIED FOR BLOCK',I3)                  CSAC1140
  207 FORMAT (31H PREVIOUS SPECIFICATION DELETED )                      CSAC1150
      END                                                               CSAC1160
```

```
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM2
C       ***** APRIL 15, 1966 *****
C
C         PREPARE FOR SORT OPERATION
C
      REAL    REALS(395)
      INTEGER INTS(587),TEST( 9),DELAY(25),ORDER(76)
      DIMENSION MTRX(75,5)  ,        INTG(25)
      COMMON  REALS, INTS
C
      EQUIVALENCE  ( INTS(   1),  MTRX(1,1) )
      EQUIVALENCE  ( INTS(396),  INTG(1)    )
      EQUIVALENCE  ( INTS(424), DELAY(1)    )
      EQUIVALENCE  ( INTS(449), ORDER(1)    )
      EQUIVALENCE  ( INTS(525),  TEST(1)    )
      EQUIVALENCE  ( INTS(540),  NCON       )
      EQUIVALENCE  ( INTS(541),   NOD       )
      EQUIVALENCE  ( INTS(542),   NEQ       )
      EQUIVALENCE  ( INTS(543),   NFG       )
C
C         RESET ERROR INDICATOR AND COUNTERS
              IERR = 2
             NOD    = 0
               NEQ = 0
               IFG = 0
              NCON = 2
          ORDER(1) = 76
C
C          TEST FOR SELECTED ELEMENTS
C
      DO   85    I = 1,75
             ITYPE = IABS( MTRX(I,1) )
      IF ( ITYPE ) 85,85,1
    1   MTRX(I,1) = ITYPE
C
C     TEST   IS ELEMENT A UNIT DELAY
      IF ( ITYPE - 21)  10,  5, 10
    5        NOD    = NOD + 1
       DELAY(NOD)   = I
      GO TO 45
   10 CONTINUE
C
C         TEST       IS ELEMENT AN INTEGRATOR
      IF ( ITYPE -  9 )  20, 15, 20
   15          NEQ = NEQ + 1
         INTG(NEQ) = I
         MTRX(I,5) = NEQ
      GO TO 45
   20 CONTINUE
C
C         TEST       IS ELEMENT A CONSTANT
      IF ( ITYPE - 11)  30, 25, 30
   25  ORDER( NCON ) = I
              NCON = NCON + 1
      GO TO 50
   30 CONTINUE
C
C     TEST     IS ELEMENT A FUNCTION GENERATOR
      IF ( ITYPE -  6)  40, 35, 40
```

```
  35      IFG = IFG + 1                                                 CSAD0620
  40 CONTINUE                                                           CSAD0630
C                                                                       CSAD0640
C     SET ELEMENT IDENTIFIER NEG UNTIL AFTER SORTING                    CSAD0650
  45  MTRX(I,1)  = -ITYPE                                               CSAD0660
  50 CONTINUE                                                           CSAD0670
     DO   75    M = 2,4                                                 CSAD0680
          LTEST = IABS( MTRX(I,M) )                                     CSAD0690
      IF (LTEST)   75,75,55                                             CSAD0700
  55 IF (LTEST - 76)   60,75,75                                         CSAD0710
  60 IF ( MTRX(LTEST,1) )   75,65,75                                    CSAD0720
C     TYPE ERROR MESSAGE AND SET ERROR INDICATOR                        CSAD0730
  65 WRITE (1,203) LTEST,I                                              CSAD0740
           IERR = 1                                                     CSAD0750
  75 CONTINUE                                                           CSAD0760
  85 CONTINUE                                                           CSAD0770
C                                                                       CSAD0780
C      TEST ON PROPER NUMBER OF ELEMENTS                                CSAD0790
C                                                                       CSAD0800
     IF ( NEQ      ) 150, 150, 100                                      CSAD0810
 100 IF (NEQ - 25)  110, 110, 155                                       CSAD0820
 110 IF (NOD - 25)  120, 120, 160                                       CSAD0830
 120 IF (IFG -  3)  130, 130, 170                                       CSAD0840
 130     TEST(4) = 2                                                    CSAD0850
     IF ( IFG )   140,140,141                                           CSAD0860
 140     TEST(4) = 1                                                    CSAD0870
             NFG = 0                                                    CSAD0880
     GO TO 300                                                          CSAD0890
 141 IF ( IFG - NFG )   143,143,142                                     CSAD0900
 142     TEST(3) = 1                                                    CSAD0910
C     SET TEST3 = 1 TO SIGNAL ADDITION OF FUNCTION GENERATOR BLOCK      CSAD0920
 143         NFG = IFG                                                  CSAD0930
     GO TO 300                                                          CSAD0940
C                                                                       CSAD0950
 150 WRITE (1,200)                                                      CSAD0960
     GO TO 190                                                          CSAD0970
 155 WRITE (1,204)                                                      CSAD0980
     GO TO 190                                                          CSAD0990
 160 WRITE (1,201)                                                      CSAD1000
     GO TO 190                                                          CSAD1010
 170 WRITE (1,202)                                                      CSAD1020
 190       IERR = 1                                                     CSAD1030
C                                                                       CSAD1040
 300 GO TO (310,320) ,IERR                                              CSAD1050
C      ERROR EXIT                                                       CSAD1060
 310     TEST(1) = 1                                                    CSAD1070
         TEST(2) = 2                                                    CSAD1080
     GO TO 350                                                          CSAD1090
C     SUCCESSFUL PRE-SORT EXIT                                          CSAD1100
 320     TEST(1) = 2                                                    CSAD1110
 350 RETURN                                                             CSAD1120
 200 FORMAT (-1130 CSMP REQUIRES AT LEAST 1 INTEGRATOR-)                CSAD1130
 201 FORMAT (-THE MAXIMUM OF 25 UNIT DELAYS HAS BEEN EXCEEDED-)         CSAD1140
 202 FORMAT (-THE MAXIMUM OF 3 FUNCTION GENERATORS HAS BEEN EXCEEDED-)  CSAD1150
 203 FORMAT (6H BLOCK,I3,28H, ONE OF THE INPUTS TO BLOCK,I3,            CSAD1160
 2031       24H, HAS NOT BEEN SPECIFIED )                               CSAD1170
 204 FORMAT ( -THE MAXIMUM OF 25 INTEGRATORS HAS BEEN EXCEEDED-)        CSAD1180
     END                                                                CSAD1190
// DUP                                                                  CSAD1200
*STORE      WS  UA  CSM2                                                CSAD1210
// JOB                                                                  CSAD1220
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM3
C       ***** MARCH 30, 1966 *****
C          SORT
C
      REAL    REALS(395)
      INTEGER INTS(587),TEST1,TEST2, DELAY(25), ORDER(76)
      DIMENSION MTRX(75,5), INTG(25)
      COMMON  REALS, INTS
C
      EQUIVALENCE  ( INTS(   1),  MTRX(1,1) )
      EQUIVALENCE  ( INTS(396),   INTG(1)   )
      EQUIVALENCE  ( INTS(424), DELAY(1)    )
      EQUIVALENCE  ( INTS(449), ORDER(1)    )
      EQUIVALENCE  ( INTS(525),   TEST1     )
      EQUIVALENCE  ( INTS(526),   TEST2     )
      EQUIVALENCE  ( INTS(534), NLIST       )
      EQUIVALENCE  ( INTS(540),   NCON      )
      EQUIVALENCE  ( INTS(541),    NOD      )
      EQUIVALENCE  ( INTS(542),    NEQ      )
C
C         RESET ERROR INDICATOR
               IERR = 2
      DO 10       N = NCON ,76
   10      ORDER(N) = 0
C
C                   SORT OPERATION
             NLIST = NCON - 1
   20 DO 150      I = 1,75
      IF (MTRX(I,1) )   30, 150, 150
   30 CONTINUE
      DO  100     M = 2,4
             LTEST = IABS (MTRX(I,M) )
      IF (LTEST)   40, 100, 40
   40 CONTINUE
      IF (NOD)   70, 70, 50
   50 DO 60   N     = 1,NOD
      IF (LTEST -  DELAY(N))  60, 100, 60
   60 CONTINUE
   70 CONTINUE
      DO  80      N = 1,NEQ
      IF (LTEST - INTG(N))  80, 100, 80
   80 CONTINUE
      DO  90      N = 1,NLIST
      IF (LTEST -  ORDER(N) )   90, 100, 90
   90 CONTINUE
      GO TO 150
  100 CONTINUE
C
  110        NLIST = NLIST + 1
       ORDER(NLIST) = I
         MTRX(I,1) = - MTRX(I,1)
      GO TO 20
  150 CONTINUE
C
C         SORT TEST
          DO 180 I  = 1,75
      IF(MTRX(I,1))  160, 180, 180
C
C         SET ERROR INDICATOR, TYPE ERROR MESSAGE, THEN PUT BLOCK
```

```
C         IN THE SORT ORDER LIST TO DETERMINE WHETHER THE REMAINDER
C         OF THE CONFIGURATION WOULD BE OK
  160          IERR = 1
        MTRX(I,1) = -MTRX(I,1)
      WRITE (1,199) I
             NLIST = NLIST + 1
      ORDER(NLIST) = I
      GO TO 20
  180 CONTINUE
      GO TO (190,200), IERR
C      UNSUCCESSFUL SORT
  190        TEST1 = 1
             TEST2 = 2
      GO TO 210
C         SUCCESSFUL SORT
  200        TEST1 = 2
  210 RETURN
C
  199 FORMAT (22H SORT FAILURE - BLOCK , I4)
      END
// DUP
*STORE      WS  UA  CSM3
// JOB
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM4
C          ***** AUGUST 13, 1966 *****
C            INITIAL CONDITIONS AND PARAMETERS
C
      REAL    REALS(395)
      INTEGER  INTS(587), TEST2
      DIMENSION PAR1(75),PAR2(75),PAR3(75),MTRX1(75)
      COMMON  REALS, INTS
      EQUIVALENCE  ( INTS(   1), MTRX1(1)    )
      EQUIVALENCE  ( INTS(376),    I         )
      EQUIVALENCE  ( INTS(381),    KEY2      ) , ( REALS( 81),   PAR1(1)  )
      EQUIVALENCE  ( INTS(390),    KEY11     ) , ( REALS(156),   PAR2(1)  )
      EQUIVALENCE  ( INTS(526),   TEST2      ) , ( REALS(231),   PAR3(1)  )
      TEST2 = 1
      WRITE (1,201)
C            CARD ENTRY SECTION
   10 READ  (2,206) SY1,SY2,SY3,SY4,I,P1,P2,P3
      IF (I)  150, 50, 100
C            PRINT CARD
   20 CALL DATSW (11, KEY11    )
      GO TO ( 10, 30), KEY11
   30 WRITE (1,202) SY1,SY2,SY3,SY4,I,P1,P2,P3
      GO TO 10
C            KEYBOARD TEST SECTION
   50 CONTINUE
      TEST2 = 2
C            TEST CONSOLE ENTRY SWITCH 2
   60 CALL DATSW(2,KEY2  )
      GO TO ( 70,200), KEY2
C            KEYBOARD ENTRY SECTION
   70 WRITE (1,203)
      READ  (6,205) I,P1, P2, P3
      IF (I) 150, 60, 100
C            TEST AND STORE PARAMETERS
  100 IF ( I - 75 )   101,101,150
  101          ITYPE = MTRX1(I)
      IF ( ITYPE )   111,102,103
  102 WRITE ( 1,207 )
      GO TO 111
  103 GO TO (111,104,111,108,111,108,106,104,111,104,106,108,104,104,
 1031          106,104,104,104,111,106,106,106,111,104,108,106,104,104,
 1032          104,104,111,111,111,111,111) , ITYPE
  104 IF ( P1 )   110,106,110
  106 IF ( P2 )   110,108,110
  108 IF ( P3 )   110,111,110
  110 WRITE (1,208)
  111        PAR1(I) = P1
             PAR2(I) = P2
             PAR3(I) = P3
  145 GO TO ( 20, 60), TEST2
C
C            ERROR SECTION
  150 GO TO (160,170), TEST2
  160 WRITE (1,202) SY1,SY2,SY3,SY4,I,P1,P2,P3
  170 WRITE (1,204)
      GO TO 70
C
  201 FORMAT (/25X,34HINITIAL CONDITIONS AND PARAMETERS ,//
 2011     -IC/PAR NAME-, 5X, -BLOCK-,
   2012 5X,7HIC/PAR1,10X,5H PAR2,10X,5H PAR3)                                 CSAF0620
   202 FORMAT (4A4, 2X, I2, 3F15.4 )                                          CSAF0630
   203 FORMAT (17X,4H(    ),3(3X,12H(            )))                          CSAF0640
   204 FORMAT(-THE ABOVE SPECIFICATION CONTAINS AN INVALID BLOCK NUMBER-)CSAF0650
   205 FORMAT (18X,I2,3F15.4)                                                 CSAF0660
   206 FORMAT (4A4, 2X, I2, 3E15.4 )                                          CSAF0670
   207 FORMAT (40HNO CORRESPONDING CONFIGURATION STATEMENT)                   CSAF0680
   208 FORMAT (44HIMPROPER PARAMETER SPECIFICATION FOR ELEMENT)               CSAF0690
   200 TEST2 = 1                                                              CSAF0700
       RETURN                                                                 CSAF0710
       END                                                                    CSAF0720
// DUP                                                                        CSAF0730
*STORE      WS   UA   CSM4                                                    CSAF0740
// JOB                                                                        CSAF0750
// FOR                                                                  CSAG0010
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS                                                      CSAG0020
      SUBROUTINE CSM5                                                   CSAG0030
C        ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2         CSAG0040
C         FUNCTION GENERATOR SPECIFICATIONS                             CSAG0050
C                                                                       CSAG0060
      REAL    REALS(395)                                                CSAG0070
      INTEGER INTS(587), TEST2                                          CSAG0080
      DIMENSION MTRX(75,5), NOFG(3), F(3,11), C(76),PAR1(75),PAR2(75)   CSAG0090
      COMMON  REALS, INTS                                               CSAG0100
C                                                                       CSAG0110
      EQUIVALENCE  ( INTS(  1),  MTRX(1,1) )                            CSAG0120
      EQUIVALENCE  ( INTS(382),   KEY3     ) , ( REALS(  2),     C(1)  )CSAG0130
      EQUIVALENCE  ( INTS(390),   KEY11    ) , ( REALS( 81),   PAR1(1) )CSAG0140
      EQUIVALENCE  ( INTS(421),  NOFG(1)   ) , ( REALS(156),   PAR2(1) )CSAG0150
      EQUIVALENCE  ( INTS(526),  TEST2     ) , ( REALS(306),     F(1,1))CSAG0160
C                                                                       CSAG0170
      TEST2 = 1                                                         CSAG0180
      WRITE (1,201)                                                     CSAG0190
C                                                                       CSAG0200
C         CARD ENTRY SECTION                                            CSAG0210
C                                                                       CSAG0220
   10 READ   (2,202)  SY1,SY2,SY3,SY4,I,(C(N),N=1,4)                    CSAG0230
      IF (I)  15, 50, 15                                                CSAG0240
   15 READ   (2,203) (C(N),N=5,11)                                      CSAG0250
      IF (I)  150, 50, 100                                              CSAG0260
C                                                                       CSAG0270
C         PRINT CARD                                                    CSAG0280
C                                                                       CSAG0290
   20 CALL DATSW (11, KEY11    )                                        CSAG0300
      GO TO ( 10, 30), KEY11                                            CSAG0310
   30 WRITE  (1,202)  SY1,SY2,SY3,SY4,I,(C(N),N=1,4)                    CSAG0320
      WRITE  (1,203) (C(N),N=5,11)                                      CSAG0330
      GO TO 10                                                          CSAG0340
C                                                                       CSAG0350
C         KEYBOARD TEST SECTION                                         CSAG0360
C                                                                       CSAG0370
   50 CONTINUE                                                          CSAG0380
      TEST2 = 2                                                         CSAG0390
C                                                                       CSAG0400
C         TEST CONSOLE ENTRY SWITCH 3                                   CSAG0410
   60 CALL DATSW(3,KEY3  )                                              CSAG0420
      GO TO ( 70,200), KEY3                                             CSAG0430
C                                                                       CSAG0440
C         KEYBOARD ENTRY SECTION                                        CSAG0450
C                                                                       CSAG0460
   70 WRITE (1,204)                                                     CSAG0470
      READ (6,202)    SY1,SY2,SY3,SY4,I,(C(N),N=1,4)                    CSAG0480
      IF (I) 150, 60, 80                                                CSAG0490
   80 CONTINUE                                                          CSAG0500
      WRITE (1,205)                                                     CSAG0510
      READ (6,203) (C(N), N=5,8)                                        CSAG0520
      WRITE (1,206)                                                     CSAG0530
      READ  (6,203) (C(N), N=9,11)                                      CSAG0540
      GO TO 100                                                         CSAG0550
C         TEST AND STORE FUNCTION GENERATOR                             CSAG0560
  100 IF (I-75) 110, 110, 150                                           CSAG0570
  110 IF (MTRX(I,1) - 6) 150, 111, 150                                  CSAG0580
  111 DO 112    I2 = 1,3                                                CSAG0590
      IF ( I - NOFG(I2) )   112,125,112                                 CSAG0600
  112 CONTINUE                                                          CSAG0610
  115 DO 120 I2=1,3                                                     CSAG0620
      M = NOFG(I2)                                                      CSAG0630
      IF (M)  119, 125, 119                                             CSAG0640
  119 IF (MTRX(M,1) - 6) 125, 120, 125                                  CSAG0650
  120 CONTINUE                                                          CSAG0660
      GO TO 150                                                         CSAG0670
  125 MTRX(I,5) = I2                                                    CSAG0680
      NOFG(I2) = I                                                      CSAG0690
  130 DO 140     M = 1,11                                               CSAG0700
  140      F(I2,M) = C(M)                                               CSAG0710
C        CHECK PAR1 AND PAR2                                            CSAG0720
  141 IF ( PAR1(I) - PAR2(I) )   190,190,145                            CSAG0730
  145 GO TO ( 20, 60), TEST2                                            CSAG0740
C                                                                       CSAG0750
C         ERROR SECTION                                                 CSAG0760
C                                                                       CSAG0770
  150 WRITE (1,207) I                                                   CSAG0780
      GO TO (170,70 ), TEST2                                            CSAG0790
  170 GO TO (180, 70), KEY11                                            CSAG0800
  180 WRITE (1,202)   SY1,SY2,SY3,SY4,I,(C(N),N=1,4)                    CSAG0810
      GO TO 70                                                          CSAG0820
  190 WRITE (1,208) I                                                   CSAG0830
      READ (6,209) PAR1(I),PAR2(I)                                      CSAG0840
      GO TO 141                                                         CSAG0850
C                                                                       CSAG0860
  201 FORMAT (/25X,34H FUNCTION GENERATOR SPECIFICATIONS )              CSAG0870
  202 FORMAT  (4A4,2X,I2,4(5X,F10.4))                                   CSAG0880
  203 FORMAT (25X,F10.4, 5X,F10.4, 5X,F10.4, 5X,F10.4)                  CSAG0890
  204 FORMAT (17X,4H(   ),4(3X,12H(           )))                       CSAG0900
  205 FORMAT (21X,       4(3X,12H(           )))                        CSAG0910
  206 FORMAT (21X,       3(3X,12H(           )))                        CSAG0920
  207 FORMAT (/'BLOCK',I3,' HAS NOT BEEN SPECIFIED AS A FUNCTION GENERATCSAG0930
  2071OR' )                                                             CSAG0940
  208 FORMAT ('SPECIFY UPPER AND LOWER LIMITS FOR F.G. BLOCK',I3 /      CSAG0950
  2081       '(         ) PAR1    (         ) PAR2')                    CSAG0960
  209 FORMAT (F10.0,10X,F9.0)                                           CSAG0970
C                                                                       CSAG0980
  200 TEST2 = 1                                                         CSAG0990
      RETURN                                                            CSAG1000
      END                                                               CSAG1010
```

```
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM6
C      ***** APRIL 14, 1966 *****
C     OPTION TO PUNCH CORRECTED DATA DECK  (SWITCH 12)
C
      REAL     REALS(395)
      INTEGER INTS(587),TEST4,TYPE(40)
      DIMENSION MTRX(75,5), NOFG(3), F(3,11), PAR(75,3)
      COMMON  REALS, INTS
      EQUIVALENCE  ( INTS(528),  TEST4      ) , ( REALS( 81),   PAR(1,1))
      EQUIVALENCE  ( INTS(  1),  MTRX(1,1) ) , ( REALS(306),     F(1,1))
      EQUIVALENCE  ( INTS(421),  NOFG(1)    )
      EQUIVALENCE  ( INTS(546),  TYPE(1)    )
C          PUNCH OUT CARDS FOR CONFIGURATION SPECIFICATIONS
C
    5 WRITE (1,6)
    6 FORMAT (50HLOAD BLANK CARDS IN CARD READ/PUNCH AND PUSH START)
      PAUSE
      DO 20 I=1,75
               ITYPE = MTRX(I,1)
      IF (ITYPE)   20,20,10
   10 WRITE (2,100) I,TYPE(ITYPE),MTRX(I,2),MTRX(I,3),MTRX(I,4)
   20 CONTINUE
C          PUNCH BLANK CARD
      WRITE (2,103)
C          PUNCH CARDS FOR INITIAL CONDITIONS AND PARAMETERS
      DO 50 I=1,75
      IF ( MTRX(I,1) ) 50, 50, 30
   30 DO 35 J=1,3
      IF ( PAR(I,J) ) 40, 35, 40
   35 CONTINUE
      GO TO 50
   40 WRITE (2,102) I, ( PAR(I,J), J=1,3 )
   50 CONTINUE
C
C          PUNCH BLANK CARD
      WRITE (2,103)
C          PUNCH FUNCTION GENERATOR CARDS
C
      GO TO (90,55), TEST4
   55 DO 80 I2=1,3
      N = NOFG(I2)
      IF (N) 80, 80, 60
C          CONFIRM THAT BLOCK IS A FUNCTION GENERATOR
   60 IF (MTRX(N,1) - 6 ) 80, 70, 80
   70 CONTINUE
      WRITE (2,104) N, (F(I2,K),K=1,4 )
      WRITE (2,105) (F(I2,K), K=5,11 )
   80 CONTINUE
C
C          PUNCH BLANK CARD
      WRITE (2,103)
C
   90 RETURN
  100 FORMAT (18X, I2, 9X, A1, 3(7X,I3) )
  102 FORMAT (18X, I2, 3F15.5)
  103 FORMAT (72X)
  104 FORMAT (18X, I2, 4( 5X,  F10.5 ) )
  105 FORMAT ( 25X, F10.5, 5X, F10.5, 5X, F10.5, 5X, F10.5 )
C
```

```
      END                                                               CSAH0620
// DUP                                                                  CSAH0630
*STORE      WS  UA  CSM6                                                CSAH0640
// JOB                                                                  CSAH0650
// FOR                                                                  CSAI0010
*ONE WORD INTEGERS                                                      CSAI0020
      SUBROUTINE CSM7                                                   CSAI0030
C      ***** APRIL 1, 1966 *****                                        CSAI0040
C     REQUEST TIMING INFORMATION                                        CSAI0050
C                                                                       CSAI0060
      REAL    REALS(395)                                                CSAI0070
      INTEGER INTS(587) ,TEST7                                          CSAI0080
      COMMON  REALS, INTS                                               CSAI0090
      EQUIVALENCE  ( INTS(531),  TEST7     ) , ( REALS( 78),    DT     )CSAI0100
      EQUIVALENCE                              ( REALS( 79),  DTS2     )CSAI0110
      EQUIVALENCE                              ( REALS( 80),  TTOT     )CSAI0120
C                                                                       CSAI0130
      GO TO (1,10) , TEST7                                              CSAI0140
C        TEST7 = 1 UNTIL FIRST TIME THROUGH CSM7                        CSAI0150
C        TEST7 = 2 AFTER FIRST TIME THROUGH CSM7                        CSAI0160
    1        TEST7 = 2                                                  CSAI0170
   10 WRITE (1,100)                                                     CSAI0180
      READ (6,101)  DT                                                  CSAI0190
      IF (DT) 50, 50, 20                                                CSAI0200
   20 WRITE (1,102)                                                     CSAI0210
      READ (6,101)  TTOT                                                CSAI0220
      IF (TTOT - DT) 60, 60, 30                                         CSAI0230
C                                                                       CSAI0240
  100 FORMAT (//32H(         ) INTEGRATION INTERVAL )                   CSAI0250
  101 FORMAT (F10.0)                                                    CSAI0260
  102 FORMAT (//22H(         ) TOTAL TIME )                             CSAI0270
  103 FORMAT (//-INTEGRATION INTERVAL MUST BE GREATER THAN ZERO-  )     CSAI0280
  104 FORMAT (//-TOTAL TIME MUST BE GREATER THAN INTEGRATION INTERVAL-) CSAI0290
C                                                                       CSAI0300
C         SET UP INTEGRATION AND TIME EQUATIONS                         CSAI0310
   30 CONTINUE                                                          CSAI0320
      DTS2 = DT*0.5                                                     CSAI0330
      RETURN                                                            CSAI0340
C                                                                       CSAI0350
C         DT ERROR                                                      CSAI0360
   50 WRITE (1,103)                                                     CSAI0370
      GO TO 10                                                          CSAI0380
C                                                                       CSAI0390
C         TTOT ERROR                                                    CSAI0400
   60 WRITE (1,104)                                                     CSAI0410
      GO TO 20                                                          CSAI0420
      END                                                               CSAI0430
// DUP                                                                  CSAI0440
*STORE      WS  UA  CSM7                                                CSAI0450
// JOB                                                                  CSAI0460
// FOR                                                                  CSAJ0010
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS                                                      CSAJ0020
      SUBROUTINE CSM10                                                  CSAJ0030
C        ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2         CSAJ0040
C         CONTROLS THE COMPUTATION AND OUTPUT                           CSAJ0050
C                                                                       CSAJ0060
      REAL    REALS(395)                                                CSAJ0070
      INTEGER  INTS(587), TEST5, ORDER(76),TEST6                        CSAJ0080
      DIMENSION INTG(25), C(76), PAR1(75) , Y(25), DYDT(25), YK(25)     CSAJ0090
      COMMON  REALS, INTS                                               CSAJ0100
      EQUIVALENCE  ( INTS(394),   KEY15     ) , ( REALS(  2),     C(1)  )CSAJ0110
      EQUIVALENCE  ( INTS(395),   KEY16     ) , ( REALS( 77),     T     )CSAJ0120
      EQUIVALENCE  ( INTS(396),  INTG(1)    ) , ( REALS( 78),    DT     )CSAJ0130
      EQUIVALENCE  ( INTS(449), ORDER(1)    ) , ( REALS( 79),  DTS2     )CSAJ0140
      EQUIVALENCE  ( INTS(529),   TEST5     ) , ( REALS( 80),  TTOT     )CSAJ0150
      EQUIVALENCE  ( INTS(530),   TEST6     ) , ( REALS( 81),   PAR1(1) )CSAJ0160
      EQUIVALENCE  ( INTS(534), NLIST       ) , ( REALS(341),     Y(1)  )CSAJ0170
      EQUIVALENCE  ( INTS(540), NCON        ) , ( REALS(366),  DYDT(1)  )CSAJ0180
      EQUIVALENCE  ( INTS(542),    NEQ      ) , ( REALS(391), TSAMP     )CSAJ0190
      EQUIVALENCE  ( INTS(586),     IR      )                           CSAJ0200
      EQUIVALENCE  ( INTS(587),   IARG      )                           CSAJ0210
C                                                                       CSAJ0220
      GO TO (60,1), KEY15                                               CSAJ0230
C                                                                       CSAJ0240
C         NORMAL SETUP                                                  CSAJ0250
C                                                                       CSAJ0260
    1 DO  10  NEXT = 2,NLIST                                            CSAJ0270
                 I =  ORDER(NEXT)                                       CSAJ0280
   10         C(I) = PAR1(I)                                            CSAJ0290
                 T = 0.0                                                CSAJ0300
             TZERO = 0.0                                                CSAJ0310
      GO TO 50                                                          CSAJ0320
C                                                                       CSAJ0330
C         RESTART SETUP                                                 CSAJ0340
C                                                                       CSAJ0350
   60            N = NCON - 1                                           CSAJ0360
      IF ( N - 2 )   66,66,61                                           CSAJ0370
   61 DO 65   NEXT = 2,N                                                CSAJ0380
                 I = ORDER(NEXT)                                        CSAJ0390
   65         C(I) = PAR1(I)                                            CSAJ0400
   66        TZERO = C(76)                                              CSAJ0410
C                                                                       CSAJ0420
   50 DO 70  INTNO = 1,NEQ                                              CSAJ0430
                 I =INTG(INTNO)                                         CSAJ0440
   70     Y(INTNO) = C(I)                                               CSAJ0450
C                                                                       CSAJ0460
C                                                                       CSAJ0470
                IR = 7243                                               CSAJ0480
      EPSLN = DTS2 / (TSAMP*2.0)                                        CSAJ0490
      TEST5 = 1                                                         CSAJ0500
      N = 1                                                             CSAJ0510
                NN = T/TSAMP + 1.0                                      CSAJ0520
      CALL CSM11                                                        CSAJ0530
             TEST6 = 1                                                  CSAJ0540
             IARG = 6                                                   CSAJ0550
      CALL CSM9                                                         CSAJ0560
             IARG = 2                                                   CSAJ0570
      GO TO 125                                                         CSAJ0580
   90 CALL CSM8
C
C          START EXECUTION
C
C          FIRST HALF-STEP
  100 TEST5 = 2
      DO 110     IX = 1,NEQ
         YK(IX)     = Y(IX)
  110        Y(IX) = Y(IX) + DTS2 * DYDT(IX)
             TNEXT = N*DT + TZERO
                 T = TNEXT - DTS2
      CALL CSM11
       GO TO 130
C          SECOND HALF-STEP
  115 TEST5 = 3
       DO 120  IX  = 1,NEQ
  120         Y(IX) = YK(IX) + DT*DYDT(IX)
                 T = TNEXT
                 N = N+1
      CALL CSM11
             IARG = 3
      CALL CSM9
  125 CALL DATSW (0, KEY16    )
      GO TO (150,130), KEY16
  130 GO TO ( 90,115,140,150,150,150) ,TEST5
  140 CONTINUE
C
C          IS IT TIME TO PRINT
                 M = T/TSAMP + EPSLN
      IF ( M - NN ) 160, 150, 150
  150 CONTINUE
C          PRINT
             IARG = 2
      CALL CSM8
C          PRINT-TIME MARKER ON PLOTTER OUTPUT
             IARG = 4
      CALL CSM9
                NN = M+1
C
C          IS RUN FINISHED
  160 GO TO (170,170,170,210,210,210), TEST5
  170 CALL DATSW (0, KEY16    )
      GO TO (200, 190), KEY16
  190 IF (T - TTOT + DTS2    ) 100, 210, 210
  200        TEST5 = 7
  210 RETURN
      END
```

```
// FOR
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS
      SUBROUTINE CSM11
C          ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2
C            DOES THE COMPUTATION FOR ONE HALF DT
C
      REAL     REALS(395)
      INTEGER INTS(587), TEST5, ORDER(76)
      INTEGER   TEST6
      DIMENSION INTG(25), C(76), F(3,11), Y(25), DYDT(25)
      DIMENSION MTRX1(75), MTRX2(75), MTRX3(75), MTRX4(75),MTRX5(75)
      DIMENSION  PAR1(75), PAR2(75), PAR3(75)
      COMMON  REALS, INTS
      EQUIVALENCE  ( INTS(  1),   MTRX1(1)   ) , ( REALS(  2),     C(1)   )
      EQUIVALENCE  ( INTS( 76),   MTRX2(1)   ) , ( REALS( 78),  DT        )
      EQUIVALENCE  ( INTS(151),   MTRX3(1)   ) , ( REALS( 79),  DTS2      )
      EQUIVALENCE  ( INTS(226),   MTRX4(1)   ) , ( REALS( 81),    PAR1(1) )
      EQUIVALENCE  ( INTS(301),   MTRX5(1)   ) , ( REALS(156),    PAR2(1) )
      EQUIVALENCE  ( INTS(376),       I      ) , ( REALS(231),    PAR3(1) )
      EQUIVALENCE  ( INTS(377),       J      ) , ( REALS(306),     F(1,1))
      EQUIVALENCE  ( INTS(378),       K      ) , ( REALS(341),     Y(1)   )
      EQUIVALENCE  ( INTS(379),       L      ) , ( REALS(366),  DYDT(1)   )
      EQUIVALENCE  ( INTS(395),    KEY16     )
      EQUIVALENCE  ( INTS(396),   INTG(1)    )
      EQUIVALENCE  ( INTS(449),  ORDER(1)    )
      EQUIVALENCE  ( INTS(529),   TEST5      )
      EQUIVALENCE  ( INTS(534),  NLIST       )
      EQUIVALENCE  ( INTS(540),  NCON        )
      EQUIVALENCE  ( INTS(542),    NEQ       )
      EQUIVALENCE  ( INTS(586),    IR        )
      EQUIVALENCE (INTS(530),TEST6),(INTS(394),KEY15)
C
  900 DO 901 INTNO = 1,NEQ
      N = INTG(INTNO)
  901         C(N) = Y(INTNO)
              NEXT = NCON
 1000  CONTINUE
                 I =  ORDER(NEXT)
             ITYPE = MTRX1(I)
      GO TO (140,56,140,53,140,53,54,56,51,10,11,53,56,56,54,56,55,52,
     1   140,53,53,22,51,55,53,53,27,56,55,140,31,32,33,34,35), ITYPE
C       THREE PARAMETERS
   51            P3 = PAR3(I)
C       THREE INPUTS
   52             L = MTRX4(I)
                 CL = C(L)
C       TWO PARAMETERS
   53            P2 = PAR2(I)
C       ONE PARAMETER
   54            P1 = PAR1(I)
C       TWO INPUTS
   55             K = MTRX3(I)
                 CK = C(K)
C       ONE INPUT
   56             J = MTRX2(I)
                 CJ = C(J)
      GO TO (140,2,140,4,140,6,7,8,9,10,11,12,13,14,15,16,17,18,140,20,
     1   21,140,23,24,25,26,27,28,29), ITYPE
```

```
C                                                                       CSAK0590
C        B         BANG-BANG                                            CSAK0600
    2 IF (CJ)     102,98,95                                             CSAK0610
  102           CI = -1.0                                               CSAK0620
      GO TO 99                                                          CSAK0630
C        D         DEAD SPACE                                           CSAK0640
    4 IF(CJ) 304, 98, 104                                               CSAK0650
  104         DIFF = CJ - P1                                            CSAK0660
      IF (DIFF) 98,98,204                                               CSAK0670
  204           CI = DIFF                                               CSAK0680
      GO TO 99                                                          CSAK0690
  304         DIFF = CJ - P2                                            CSAK0700
      IF (DIFF)  204,98,98                                              CSAK0710
C        F         FUNCTION GENERATOR                                   CSAK0720
    6           NF = MTRX5(I)                                           CSAK0730
                P3 = P1 - P2                                            CSAK0740
      IF (P3)   140,140,506                                             CSAK0750
  506           P1 = 10.0*(CJ - P2) / P3                                CSAK0760
             NSECT = P1                                                 CSAK0770
      IF ( P1 )   106,106,206                                           CSAK0780
  106           CI = F(NF,1)                                            CSAK0790
      GO TO 99                                                          CSAK0800
  206 IF ( NSECT - 10 )   406,306,306                                   CSAK0810
  306           CI = F(NF,11)                                           CSAK0820
      GO TO 99                                                          CSAK0830
  406           P2 = NSECT                                              CSAK0840
                P3 = P1 - P2                                            CSAK0850
                P1 = F(NF,NSECT+1)                                      CSAK0860
                P2 = F(NF,NSECT+2)                                      CSAK0870
                CI = P1 + P3*(P2 - P1)                                  CSAK0880
      GO TO 99                                                          CSAK0890
C        G         GAIN                                                 CSAK0900
   07           CI = P1 * CJ                                            CSAK0910
      GO TO 99                                                          CSAK0920
C        H         HALF POWER (SQUARE ROOT)                             CSAK0930
   08           CI = SQRT (CJ)                                          CSAK0940
      GO TO 99                                                          CSAK0950
C        I         INTEGRATOR   (MAXIMUM 25 ELEMENTS)                   CSAK0960
   09     INTNO    = MTRX5(I)                                           CSAK0970
       DYDT(INTNO) = CJ +P2*CK + P3*CL                                  CSAK0980
      GO TO 100                                                         CSAK0990
C        J         JITTER (RANDOM NUMBER GENERATOR BETWEEN + AND - 1)   CSAK1000
   10           IR = 259 * IR                                           CSAK1010
                CI = FLOAT(IR) / 32767.0                                CSAK1020
      GO TO 99                                                          CSAK1030
C        K         CONSTANT                                             CSAK1040
   11 GO TO 100                                                         CSAK1050
C        L         LIMITER                                              CSAK1060
   12        DIFF  = CJ - P1                                            CSAK1070
      IF (DIFF)    112,93,93                                            CSAK1080
  112         DIFF = CJ - P2                                            CSAK1090
      IF (DIFF)   94,97,97                                              CSAK1100
C        M         MAGNITUDE  (ABSOLUTE VALUE)                          CSAK1110
   13           CI = ABS (CJ)                                           CSAK1120
      GO TO 99                                                          CSAK1130
C        N         NEGATIVE CLIPPER                                     CSAK1140
   14 IF (CJ) 98,98,97                                                  CSAK1150
C        O         OFFSET                                               CSAK1160
```

```
   15           CI = CJ + P1
      GO TO 99
C        P          POSITIVE CLIPPER
   16 IF (CJ) 97,98,98
C        Q          QUIT
   17 IF (CJ - CK)   100,100,170
C        R          RELAY
   18 IF (CJ) 218, 118, 118
  118           CI = CK
      GO TO 99
  218           CI = CL
      GO TO 99
   20 GO TO ( 120,98,1620),TEST5
  120 GO TO ( 620,220),TEST6
  220 IF ( MTRX5(I) )   320,320,520
  320      PAR3(I) = 0.0
      GO TO 1120
  520      PAR3(I) = P2
  620 GO TO ( 720,1120),KEY15
  720 IF ( PAR3(I) )   820,1120,1120
  820      PAR2(I) = PAR3(I)
          MTRX5(I) = 1
      GO TO 100
 1120     MTRX5(I) = 0
 1220 IF( CJ )   98,1320,1320
 1320     MTRX5(I) = 1
 1420      PAR2(I) = -P1 + DTS2 + DT
      GO TO 95
 1620 IF ( MTRX5(I) )   1220,1220,1720
 1720 IF ( P2 )   1820,1420,1420
 1820      PAR2(I) = P2 + DT
      GO TO 98
C        U          UNIT DELAY   (MAXIMUM 25 ELEMENTS)
   21 GO TO (221,121,121), TEST5
  121         C(I) = P2
  221    PAR2(I)   = CJ
      GO TO 100
C        V          VACUOUS  (USED IN CONJUNCTION WITH WYE ELEMENT)
   22 GO TO (122,100,100), TEST5
  122    MTRX5(I)  = NEXT
      GO TO 100
C        W          WEIGHTED SUMMER
   23           CI =  CJ *P1 + CK*P2 + CL*P3
      GO TO 99
C        X          MULTIPLIER
   24           CI = CJ * CK
      GO TO 99
C        Y          WYE                    IMPLICIT FUNCTION TEST
   25       RELER  = ABS(1.-CK/CJ)
      IF ( RELLR  - P1)   97,97,125
  125 CALL DATSW (0, KEY16)
      GO TO (160,225), KEY16
  225        C(K) = (1.0 - P2)*CJ + P2*CK
            NEXT   = MTRX5(K)
        GO TO 1000
C        Z          ZERO ORDER HOLD
   26 GO TO (126,226,226), TEST5
  126     PAR2(I) = C(I)
```

```
                P2 = C(I)                                               CSAK1750
  226 IF (CK)   94,94,326                                               CSAK1760
  326    PAR2(I)   = CJ                                                 CSAK1770
      GO TO 97                                                          CSAK1780
C        +         SUMMER          ( THREE INPUTS PERMITTED)            CSAK1790
   27 J = MTRX2(I)                                                      CSAK1800
      IF (J) 127,927,227                                                CSAK1810
  127 J = -J                                                            CSAK1820
      CI = -C(J)                                                        CSAK1830
      GO TO 327                                                         CSAK1840
  227 CI = C(J)                                                         CSAK1850
  327 K = MTRX3(I)                                                      CSAK1860
      IF (K) 427,627,527                                                CSAK1870
  427 K = -K                                                            CSAK1880
      CI = CI - C(K)                                                    CSAK1890
      GO TO 627                                                         CSAK1900
  527 CI = CI + C(K)                                                    CSAK1910
  627 L = MTRX4(I)                                                      CSAK1920
      IF (L) 727,99,827                                                 CSAK1930
  727 L = -L                                                            CSAK1940
      CI = CI - C(L)                                                    CSAK1950
      GO TO 99                                                          CSAK1960
  827 CI = CI + C(L)                                                    CSAK1970
      GO TO 99                                                          CSAK1980
  927           CI = 0.0                                                CSAK1990
      GO TO 327                                                         CSAK2000
C        -         SIGN INVERTER  (LIMITED TO ONE INPUT CONNECTION)     CSAK2010
   28 GO TO 96                                                          CSAK2020
C        /         DIVIDER                                              CSAK2030
   29 IF (CK)   129,140,129                                             CSAK2040
  129           CI = CJ / CK                                            CSAK2050
      GO TO 99                                                          CSAK2060
C        1         SPECIAL ELEMENT NUMBER 1                             CSAK2070
   31 CALL SUB1                                                         CSAK2080
      GO TO 100                                                         CSAK2090
C        2         SPECIAL ELEMENT NUMBER 2                             CSAK2100
   32 CALL SUB2                                                         CSAK2110
      GO TO 100                                                         CSAK2120
C        3         SPECIAL ELEMENT NUMBER 3                             CSAK2130
   33 CALL SUB3                                                         CSAK2140
      GO TO 100                                                         CSAK2150
C        4         SPECIAL ELEMENT NUMBER 4                             CSAK2160
   34 CALL SUB4                                                         CSAK2170
      GO TO 100                                                         CSAK2180
C        5         SPECIAL ELEMENT NUMBER 5                             CSAK2190
   35 CALL SUB5                                                         CSAK2200
      GO TO 100                                                         CSAK2210
C                                                                       CSAK2220
   93           CI = P1                                                 CSAK2230
      GO TO 99                                                          CSAK2240
   94           CI = P2                                                 CSAK2250
      GO TO 99                                                          CSAK2260
   95           CI = 1.0                                                CSAK2270
      GO TO 99                                                          CSAK2280
   96           CI = -CJ                                                CSAK2290
      GO TO 99                                                          CSAK2300
   97           CI = CJ                                                 CSAK2310
      GO TO 99                                                          CSAK2320
   98           CI = 0.0                                                 CSAK2330
   99         C(I) = CI                                                  CSAK2340
  100 IF (NEXT - NLIST) 1100,1200,140                                    CSAK2350
 1100        NEXT = NEXT + 1                                             CSAK2360
        GO TO 1000                                                       CSAK2370
 1200  CONTINUE                                                          CSAK2380
      RETURN                                                             CSAK2390
C                                                                        CSAK2400
C         ERROR                                                          CSAK2410
  140 CONTINUE                                                           CSAK2420
  150 TEST5 = 4                                                          CSAK2430
      GO TO 180                                                          CSAK2440
  160 TEST5 = 5                                                          CSAK2450
      GO TO 180                                                          CSAK2460
  170 TEST5 = 6                                                          CSAK2470
  180 RETURN                                                             CSAK2480
      END                                                                CSAK2490
```

```
// FOR                                                                   CSAL0010
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS                                                       CSAL0020
      SUBROUTINE CSM12                                                   CSAL0030
C        ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2          CSAL0040
      REAL    REALS(395)                                                 CSAL0050
      INTEGER INTS(587),TEST5,TEST6,TEST9                                CSAL0060
      DIMENSION    IA(15)                                                CSAL0070
      COMMON  REALS, INTS                                                CSAL0080
      EQUIVALENCE  ( INTS(380),    KEY1     ) , (  INTS(381),    KEY2    )CSAL0090
      EQUIVALENCE  ( INTS(382),    KEY3     ) , (  INTS(383),    KEY4    )CSAL0100
      EQUIVALENCE  ( INTS(384),    KEY5     ) , (  INTS(385),    KEY6    )CSAL0110
      EQUIVALENCE  ( INTS(386),    KEY7     ) , (  INTS(387),    KEY8    )CSAL0120
      EQUIVALENCE  ( INTS(388),    KEY9     ) , (  INTS(389),    KEY10   )CSAL0130
      EQUIVALENCE  ( INTS(390),    KEY11    ) , (  INTS(391),    KEY12   )CSAL0140
      EQUIVALENCE  ( INTS(392),    KEY13    ) , (  INTS(393),    KEY14   )CSAL0150
      EQUIVALENCE  ( INTS(394),    KEY15    ) , (  INTS(395),    KEY16   )CSAL0160
      EQUIVALENCE  ( INTS(529),    TEST5    ) , (  INTS(530),    TEST6   )CSAL0170
      EQUIVALENCE  ( INTS(533),    TEST9    ) , (  INTS(543),    NFG     )CSAL0180
      EQUIVALENCE  ( INTS(587),   IARG      )                            CSAL0190
      GO TO ( 210,210,210,199,195,197,195), TEST5                        CSAL0200
  195 WRITE (1,196)                                                      CSAL0210
  196 FORMAT ('RUN TERMINATED BY SWITCH 0')                              CSAL0220
      GO TO 210                                                          CSAL0230
  197 WRITE (1,198)                                                      CSAL0240
  198 FORMAT ('RUN TERMINATED BY QUIT ELEMENT')                          CSAL0250
      GO TO 210                                                          CSAL0260
  199 WRITE (1,200)                                                      CSAL0270
  200 FORMAT  (/ 20H ERROR IN PROCESSING )                               CSAL0280
  210         TEST5 = 1                                                  CSAL0290
               IARG = 5                                                  CSAL0300
      CALL CSM9                                                          CSAL0310
      CALL DATSW(10,KEY10)                                               CSAL0320
      GO TO (50,10),KEY10                                                CSAL0330
   10 WRITE (1,110)                                                      CSAL0340
  110 FORMAT (/'TO SUPPRESS INSTRUCTIONS TURN ON SWITCH 10'/             CSAL0350
  1101          '     OPTION                              SWITCH')       CSAL0360
      CALL DATSW(10,KEY10)                                               CSAL0370
      GO TO (50, 1),KEY10                                                CSAL0380
    1 WRITE (1,101)                                                      CSAL0390
  101 FORMAT ('CONFIGURATION                            1')              CSAL0400
      CALL DATSW(10,KEY10)                                               CSAL0410
      GO TO (50, 2),KEY10                                                CSAL0420
    2 WRITE (1,102)                                                      CSAL0430
  102 FORMAT ('INITIAL CONDITIONS OR PARAMETERS         2')              CSAL0440
      IF (NFG)   204,204,203                                             CSAL0450
  203 CALL DATSW(10,KEY10)                                               CSAL0460
      GO TO (50, 3),KEY10                                                CSAL0470
    3 WRITE (1,103)                                                      CSAL0480
  103 FORMAT ('FUNCTION GENERATOR INTERCEPTS            3')              CSAL0490
  204 CALL DATSW(10,KEY10)                                               CSAL0500
      GO TO (50, 4),KEY10                                                CSAL0510
    4 WRITE (1,104)                                                      CSAL0520
  104 FORMAT ('INTEGRATION SPECIFICATIONS               4')              CSAL0530
      CALL DATSW(10,KEY10)                                               CSAL0540
      GO TO (50, 5),KEY10                                                CSAL0550
    5 WRITE (1,105)                                                      CSAL0560
  105 FORMAT ('PRINT INTERVAL                           5')              CSAL0570
      CALL DATSW(10,KEY10)                                               CSAL0580
      GO TO (50, 6),KEY10                                               CSAL0590
    6 WRITE (1,106)                                                     CSAL0600
  106 FORMAT ('PRINT VARIABLES                        6')               CSAL0610
      GO TO ( 202,201) , TEST9                                          CSAL0620
  201 CALL DATSW(10,KEY10)                                              CSAL0630
      GO TO (50, 7),KEY10                                               CSAL0640
    7 WRITE (1,107)                                                     CSAL0650
  107 FORMAT ('NEW PLOT FRAME                         7')               CSAL0660
      CALL DATSW(10,KEY10)                                              CSAL0670
      GO TO (50, 8),KEY10                                               CSAL0680
    8 WRITE (1,108)                                                     CSAL0690
  108 FORMAT ('PLOTTER X-AXIS                         8')               CSAL0700
      CALL DATSW(10,KEY10)                                              CSAL0710
      GO TO (50, 9),KEY10                                               CSAL0720
    9 WRITE (1,109)                                                     CSAL0730
  109 FORMAT ('PLOTTER Y-AXIS                         9')               CSAL0740
  202 CALL DATSW(10,KEY10)                                              CSAL0750
      GO TO (50,11),KEY10                                               CSAL0760
   11 WRITE (1,111)                                                     CSAL0770
  111 FORMAT ('SUPPRESS TYPING OF CARD INPUT DATA    11')               CSAL0780
      CALL DATSW(10,KEY10)                                              CSAL0790
      GO TO (50,12),KEY10                                               CSAL0800
   12 WRITE (1,112)                                                     CSAL0810
  112 FORMAT ('PUNCH UPDATED DATA DECK               12')               CSAL0820
      CALL DATSW(10,KEY10)                                              CSAL0830
      GO TO (50,13),KEY10                                               CSAL0840
   13 WRITE (1,113)                                                     CSAL0850
  113 FORMAT ('INTERROGATE BLOCK OUTPUTS             13')               CSAL0860
      CALL DATSW(10,KEY10)                                              CSAL0870
      GO TO (50,14),KEY10                                               CSAL0880
   14 WRITE (1,114)                                                     CSAL0890
  114 FORMAT ('SAVE STATUS AT INTERRUPT TIME         14')               CSAL0900
      CALL DATSW(10,KEY10)                                              CSAL0910
      GO TO (50,15),KEY10                                               CSAL0920
   15 WRITE (1,115)                                                     CSAL0930
  115 FORMAT ('RESTART AT PREVIOUS INTERRUPT POINT   15')               CSAL0940
   50 WRITE (1,150)                                                     CSAL0950
  150 FORMAT(/'AFTER SELECTING DESIRED OPTION PRESS START',//)          CSAL0960
      PAUSE                                                             CSAL0970
      CALL DATSW (1,KEY1)                                               CSAL0980
      CALL DATSW (2,KEY2)                                               CSAL0990
      CALL DATSW (3,KEY3)                                               CSAL1000
      CALL DATSW (4,KEY4)                                               CSAL1010
      CALL DATSW (5,KEY5)                                               CSAL1020
      CALL DATSW (6,KEY6)                                               CSAL1030
      CALL DATSW (7,KEY7)                                               CSAL1040
      CALL DATSW (8,KEY8)                                               CSAL1050
      CALL DATSW (9,KEY9)                                               CSAL1060
      CALL DATSW (11,KEY11)                                             CSAL1070
      CALL DATSW (12,KEY12)                                             CSAL1080
      CALL DATSW (13,KEY13)                                             CSAL1090
      CALL DATSW (14,KEY14)                                             CSAL1100
      CALL DATSW (15,KEY15)                                             CSAL1110
      IN = 0                                                            CSAL1120
      DO 170 I=1,15                                                     CSAL1130
      IF(INTS(I+379)-1) 170,160,170                                     CSAL1140
  160 IF ( I - 10)  161,170,161                                         CSAL1150
  161 IN = IN + 1                                                       CSAL1160
      IA(IN) = I                                                        CSAL1170
  170 CONTINUE                                                          CSAL1180
      WRITE (1,175) (IA(I), I = 1,IN)                                   CSAL1190
  175 FORMAT (22HSWITCHES SET ON WERE  ,15I3,//)                        CSAL1200
      GO TO ( 301, 304), TEST6                                          CSAL1210
  301 GO TO ( 302, 304), KEY14                                          CSAL1220
  302        TEST6 = 2                                                  CSAL1230
  304 RETURN                                                            CSAL1240
      END                                                               CSAL1250
```

```
// FOR                                                                  CSAM0010
*ONE WORD INTEGERS                                                      CSAM0020
      SUBROUTINE CSM13                                                  CSAM0030
C      ***** MARCH 30, 1966 *****                                       CSAM0040
C                                                                       CSAM0050
C        BLOCK OUTPUT INTERROGATION ROUTINE                             CSAM0060
C                                                                       CSAM0070
      REAL    REALS(395)                                                CSAM0080
      INTEGER INTS(587)                                                 CSAM0090
      DIMENSION C(76)                                                   CSAM0100
      COMMON REALS, INTS                                                CSAM0110
      EQUIVALENCE ( INTS(392),   KEY13    ) , ( REALS(  2),     C(1)   )CSAM0120
C                                                                       CSAM0130
      WRITE (1,100)                                                     CSAM0140
    1 WRITE (1,101)                                                     CSAM0150
      READ  (6,102) I                                                   CSAM0160
      IF (I )   20,2,10                                                 CSAM0170
   10 IF (I - 75)   30,30,20                                            CSAM0180
   20 WRITE (1,104)                                                     CSAM0190
      GO TO 1                                                           CSAM0200
   30 CONTINUE                                                          CSAM0210
      WRITE (1,103) I,C(I)                                              CSAM0220
      CALL DATSW(13,KEY13)                                              CSAM0230
      GO TO (1,2) ,KEY13                                                CSAM0240
    2 RETURN                                                            CSAM0250
  100 FORMAT (/-OUTPUT INTERROGATION OPTION- /)                         CSAM0260
  101 FORMAT (-(   ) BLOCK-)                                            CSAM0270
  102 FORMAT (I3)                                                       CSAM0280
  103 FORMAT (-OUTPUT OF BLOCK-,I3,5X,F10.4)                            CSAM0290
  104 FORMAT (-ENTRY ERROR-)                                            CSAM0300
      END                                                               CSAM0310
// DUP                                                                  CSAM0320
*STORE      WS  UA  CSM13                                               CSAM0330
// JOB                                                                  CSAM0340
// FOR                                                                  CSBA0010
*ONE WORD INTEGERS                                                      CSBA0020
      SUBROUTINE CSM8                                                   CSBA0030
C         MASTER CONTROL FOR PRINTER OUTPUT     VERSION 1 MOD 1         CSBA0040
      REAL    REALS(395)                                                CSBA0050
      INTEGER INTS(587)                                                 CSBA0060
      DIMENSION C(76)                                                   CSBA0070
      COMMON  REALS, INTS                                               CSBA0080
      EQUIVALENCE  ( INTS(535),    K1       ) , ( REALS(  2),     C(1)  )CSBA0090
      EQUIVALENCE  ( INTS(536),    K2       ) , ( REALS( 77),     T     )CSBA0100
      EQUIVALENCE  ( INTS(537),    K3       )                           CSBA0110
      EQUIVALENCE  ( INTS(538),    K4       )                           CSBA0120
      EQUIVALENCE  ( INTS(539),    K5       )                           CSBA0130
      EQUIVALENCE  ( INTS(587),  IARG       )                           CSBA0140
      GO TO ( 1, 2), IARG                                               CSBA0150
    1 CALL CSM8A                                                        CSBA0160
      RETURN                                                            CSBA0170
    2           TC = T + 0.0005                                         CSBA0180
               CK1 = C(K1) + SIGN(0.00005,C(K1))                        CSBA0190
               CK2 = C(K2) + SIGN(0.00005,C(K2))                        CSBA0200
               CK3 = C(K3) + SIGN(0.00005,C(K3))                        CSBA0210
               CK4 = C(K4) + SIGN(0.00005,C(K4))                        CSBA0220
               CK5 = C(K5) + SIGN(0.00005,C(K5))                        CSBA0230
      WRITE (1,201) TC,CK1,CK2,CK3,CK4,CK5                              CSBA0240
  201 FORMAT (5X, F10.3, 5F13.4 )                                       CSBA0250
      RETURN                                                            CSBA0260
      END                                                               CSBA0270
// DUP                                                                  CSBA0280
*STORE      WS  UA  CSM8                                                CSBA0290
// JOB                                                                  CSBA0300
// FOR                                                                  CSBB0010
*ONE WORD INTEGERS                                                      CSBB0020
      SUBROUTINE CSM8A                                                  CSBB0030
C      ***** APRIL 1, 1966 *****                                        CSBB0040
C     REQUEST PRINT INFORMATION                                         CSBB0050
C                                                                       CSBB0060
      REAL    REALS(395)                                                CSBB0070
      INTEGER INTS(587),PRINT(5),TEST8                                  CSBB0080
      COMMON  REALS, INTS                                               CSBB0090
      EQUIVALENCE                              ( REALS( 78),    DT     )CSBB0100
      EQUIVALENCE  ( INTS(384),   KEY5     ) , ( REALS(391), TSAMP     )CSBB0110
      EQUIVALENCE  ( INTS(385),   KEY6     )                            CSBB0120
      EQUIVALENCE  ( INTS(532),  TEST8     )                            CSBB0130
      EQUIVALENCE  ( INTS(535), PRINT(1)   )                            CSBB0140
C                                                                       CSBB0150
      GO TO (2,1), TEST8                                                CSBB0160
    1 GO TO (2 ,4 ), KEY5                                               CSBB0170
    2 WRITE (1,101)                                                     CSBB0180
      READ  (6,102) TSAMP                                               CSBB0190
      IF ( TSAMP - DT ) 80,4,4                                          CSBB0200
C                                                                       CSBB0210
C         FIND OUT WHICH BLOCKS ARE TO BE PRINTED                       CSBB0220
C                                                                       CSBB0230
    4 GO TO (6,5),TEST8                                                 CSBB0240
    5 GO TO (7 ,40), KEY6                                               CSBB0250
    6        TEST8 = 2                                                  CSBB0260
    7 WRITE (1,103)                                                     CSBB0270
      READ  (6,105)  (PRINT(N), N=1,5 )                                 CSBB0280
C                                                                       CSBB0290
C         IF BLOCK NUMBER IS NOT BETWEEN 1 AND 75, PRINT 0.0            CSBB0300
C                                                                       CSBB0310
      DO 30 N=1,5                                                       CSBB0320
      IF (  PRINT(N) )  10, 10, 20                                      CSBB0330
   10  PRINT(N) = 0                                                     CSBB0340
      GO TO 30                                                          CSBB0350
   20 IF ( PRINT(N) - 75 ) 30, 30, 10                                   CSBB0360
   30 CONTINUE                                                          CSBB0370
      GO TO 50                                                          CSBB0380
   40 WRITE (1,104) PRINT                                               CSBB0390
   50 RETURN                                                            CSBB0400
   80 WRITE (1,109)                                                     CSBB0410
      GO TO 2                                                           CSBB0420
C                                                                       CSBB0430
  101 FORMAT (// 26H(          ) PRINT INTERVAL )                       CSBB0440
  102 FORMAT (F10.0)                                                    CSBB0450
  103 FORMAT ( 9X, 4HTIME, 3X, 5(3X, 10HOUTPUT(   ) ) )                 CSBB0460
  104 FORMAT ( 9X, 4HTIME, 3X, 5(3X,  6HOUTPUT,I3,1X) )                 CSBB0470
  105 FORMAT (15X, 5(11X, I2) )                                         CSBB0480
  109 FORMAT (/-PRINT INTERVAL CAN NOT BE LESS THAN INTEGRATION INTERVALCSBB0490
  1091-   )                                                             CSBB0500
C                                                                       CSBB0510
      END                                                               CSBB0520
// DUP                                                                  CSBB0530
*STORE      WS  UA  CSM8A                                               CSBB0540
// JOB                                                                  CSBB0550
// FOR                                                                 CSBC0010
*ONE WORD INTEGERS                                                     CSBC0020
      SUBROUTINE CSM9                                                  CSBC0030
C      ***** APRIL 14, 1966 *****                                      CSBC0040
C         PLOTTER CONTROL SUBROUTINE                                   CSBC0050
C                                                                      CSBC0060
      REAL    REALS(395)                                               CSBC0070
      INTEGER INTS(587)                                                CSBC0080
      DIMENSION  C(76)                                                 CSBC0090
      COMMON  REALS, INTS                                              CSBC0100
      EQUIVALENCE                               ( REALS(  2),    C(1)  )CSBC0110
      EQUIVALENCE  ( INTS(544), IVERT       ) , ( REALS(392), VDEL     )CSBC0120
      EQUIVALENCE  ( INTS(545),  IHOR       ) , ( REALS(393), HDEL     )CSBC0130
      EQUIVALENCE  ( INTS(587),  IARG       ) , ( REALS(394),  VMIN    )CSBC0140
      EQUIVALENCE                               ( REALS(395),  HMIN    )CSBC0150
C                                                                      CSBC0160
      GO TO (1, 2, 3, 4, 5, 6),IARG                                    CSBC0170
C                                                                      CSBC0180
C         MOVE PLOTTER PAPER TO NEW POSITION AND DRAW PLOT FRAME       CSBC0190
    1 CALL CSM9A                                                       CSBC0200
      GO TO 100                                                        CSBC0210
C                                                                      CSBC0220
C         ENTER PLOT VARIABLES AND PARAMETERS                          CSBC0230
    2 CALL CSM9B                                                       CSBC0240
      GO TO 100                                                        CSBC0250
C                                                                      CSBC0260
C         PLOT POINT                                                   CSBC0270
    3 N = 0                                                            CSBC0280
   13         VERT = ( C(IVERT) - VMIN ) / VDEL                        CSBC0290
             HORIZ = ( C(IHOR ) - HMIN ) / HDEL                        CSBC0300
      IF ( VERT - 1.0 ) 23, 23, 63                                     CSBC0310
   23 IF ( VERT )   73, 33, 33                                         CSBC0320
   33 IF ( HORIZ - 1.0 )   43, 43, 83                                  CSBC0330
   43 IF ( HORIZ )   93, 53, 53                                        CSBC0340
   53 CALL FPLOT (N, HORIZ, VERT)                                      CSBC0350
      GO TO 100                                                        CSBC0360
   63         VERT = 1.0                                               CSBC0370
      GO TO 33                                                         CSBC0380
   73         VERT = 0.0                                               CSBC0390
      GO TO 33                                                         CSBC0400
   83        HORIZ = 1.0                                               CSBC0410
      GO TO 53                                                         CSBC0420
   93        HORIZ = 0.0                                               CSBC0430
      GO TO 53                                                         CSBC0440
C                                                                      CSBC0450
C         DRAW PRINT-TIME MARKER SYMBOL                                CSBC0460
    4 CALL POINT (0)                                                   CSBC0470
      GO TO 100                                                        CSBC0480
C                                                                      CSBC0490
C         PEN UP                                                       CSBC0500
    5 CALL FPLOT (1, HORIZ, VERT)                                      CSBC0510
      GO TO 100                                                        CSBC0520
C                                                                      CSBC0530
C         PEN DOWN                                                     CSBC0540
    6 N = -2                                                           CSBC0550
      GO TO 13                                                         CSBC0560
C                                                                      CSBC0570
  100 RETURN                                                           CSBC0580
      END                                                              CSBC0590
// DUP                                                                 CSBC0600
*STORE      WS  UA  CSM9                                               CSBC0610
```

```
// FOR                                                                  CSBD0010
*ONE WORD INTEGERS                                                      CSBD0020
      SUBROUTINE CSM9A                                                  CSBD0030
C      ***** MARCH 30, 1966 *****                                       CSBD0040
C         SETS UP PLOT SCALE AND DRAWS A NEW GRID                       CSBD0050
C                                                                       CSBD0060
      REAL    REALS(395)                                                CSBD0070
      INTEGER INTS(587), TEST9                                          CSBD0080
      COMMON  REALS, INTS                                               CSBD0090
      EQUIVALENCE  ( INTS(386),   KEY7      )                           CSBD0100
      EQUIVALENCE  ( INTS(533),  TEST9      )                           CSBD0110
C                                                                       CSBD0120
      GO TO (30, 10), TEST9                                             CSBD0130
C            TEST9 = 1 UNTIL INITIAL SPECIFICATION OF PLOTTER VARIABLES CSBD0140
   10 GO TO (20, 100), KEY7                                             CSBD0150
   20 CALL FPLOT(1,1.5,0.0)                                             CSBD0160
   25 CALL SCALF ( 10.0,10.0, 0.0, 0.0 )                                CSBD0170
      CALL FGRID( 0, 0.0, 0.0, 0.1, 10 )                                CSBD0180
      CALL FGRID( 1, 1.0, 0.0, 1.0,  1 )                                CSBD0190
      CALL FGRID( 2, 1.0, 1.0, 1.0,  1 )                                CSBD0200
      CALL FGRID( 3, 0.0, 1.0, 0.1, 10 )                                CSBD0210
      CALL FGRID( 1, 0.5, 0.0, 1.0,  1 )                                CSBD0220
      CALL FGRID( 2, 1.0, 0.5, 1.0,  1 )                                CSBD0230
  100 RETURN                                                            CSBD0240
   30 WRITE (1,130)                                                     CSBD0250
  130 FORMAT (- PREPARE PLOTTER AND PRESS START-/                       CSBD0260
  1301        - SET PEN ABOUT ONE INCH FROM RIGHT MARGIN- )             CSBD0270
             TEST9 = 2                                                  CSBD0280
C            TEST9 = 2 AFTER INITIAL SPECIFICATION OF PLOTTER VARIABLES CSBD0290
   50 PAUSE                                                             CSBD0300
      GO TO 25                                                          CSBD0310
      END                                                               CSBD0320
// DUP                                                                  CSBD0330
*STORE      WS  UA  CSM9A                                               CSBD0340
// JOB                                                                  CSBD0350
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM9B
C      ***** MARCH 30, 1966 *****
C     PRINT REQUEST AND FORMAT FOR PLOT INFORMATION
C
      REAL    REALS(395)
      INTEGER INTS(587)
      COMMON  REALS, INTS
      EQUIVALENCE   ( INTS(387),   KEY8      ) , ( REALS( 80),   TTOT     )
      EQUIVALENCE   ( INTS(388),   KEY9      ) , ( REALS(392), VDEL       )
      EQUIVALENCE                                ( REALS(393), HDEL       )
      EQUIVALENCE   ( INTS(544), IVERT       ) , ( REALS(394),   VMIN     )
      EQUIVALENCE   ( INTS(545),  IHOR       ) , ( REALS(395),   HMIN     )
C
      IF ( IVERT )    10,10,5
    5 GO TO (10,20), KEY9
   10 WRITE (1,100)
      READ (6,101) IVERT,VMIN,VMAX
              VDEL = VMAX - VMIN
      IF (IVERT) 150, 150, 14
   14 IF ( VDEL )    150,150,15
   15 IF (IVERT - 76) 20, 20, 150
   20 CONTINUE
      IF ( IHOR )     30,30,25
   25 GO TO (30,90), KEY8
   30 WRITE (1,102)
      READ (6,101)  IHOR,HMIN,HMAX
              HDEL = HMAX - HMIN
C
      IF (IHOR) 160,  50, 34
   34 IF( HDEL )    160,160,35
   35 IF ( IHOR  -76 )  60, 60,160
C
C          TIME IS THE HORIZONTAL AXIS
   50         HMIN = 0.0
              HDEL = TTOT
              IHOR = 76
   60 CONTINUE
C
   90 RETURN
C
  100 FORMAT (//-(   ) BLOCK FOR Y-AXIS    (           ) MINIMUM VALUE
  1001(           ) MAXIMUM VALUE-   )
  101 FORMAT ( I3,22X,F10.0,21X,F10.0 )
  102 FORMAT (//-(   ) BLOCK FOR X-AXIS    (           ) MINIMUM VALUE
  1021(           ) MAXIMUM VALUE-   )
  103 FORMAT (// - ERROR IN Y-AXIS SPECIFICATION- )
  104 FORMAT (// - ERROR IN X-AXIS SPECIFICATION- )
C
C          VERTICAL ERROR
  150 WRITE (1,103)
      GO TO 10
C
C          HORIZONTAL ERROR
  160 WRITE (1,104)
      GO TO 30
C
      END
// DUP
*STORE      WS  UA  CSM9B
```

```
// JOB                                                              CSBE0620
// FOR
* LIST SOURCE PROGRAM
*ONE WORD INTEGERS                                                      CSCA0020
      SUBROUTINE CSM8                                                   CSCA0030
C        ***** AUGUST 1, 1967 *****   VERSION 1, MODIFICATION 2         CSCA0040
C         MASTER CONTROL FOR PRINTER OUTPUT                             CSCA0050
C        THIS ALTERNATE SUBPROGRAM IS TO BE USED IF THE 1627 PLOTTER   CSCA0060
C        IS UNAVAILABLE  -- IT PROVIDES A PRINT-PLOT OUTPUT ON THE      CSCA0070
C        1131 CONSOLE PRINTER                                           CSCA0080
      REAL    REALS(395)                                                CSCA0090
      INTEGER INTS(587)                                                 CSCA0100
      DIMENSION C(76)                                                   CSCA0110
      DIMENSION IPLOT(51)                                               CSCA0120
      COMMON  REALS, INTS                                               CSCA0130
      EQUIVALENCE ( INTS(535),     K1       ) , ( REALS(  2),    C(1)  )CSCA0140
      EQUIVALENCE ( INTS(536),     K2       ) , ( REALS( 77),     T    )CSCA0150
      EQUIVALENCE ( INTS(537),     K3       ) , ( REALS(392),  VDEL    )CSCA0160
      EQUIVALENCE ( INTS(538),     K4       ) , ( REALS(394),  VMIN    )CSCA0170
      EQUIVALENCE ( INTS(587),   IARG       )                           CSCA0180
      GO TO ( 1, 2), IARG                                               CSCA0190
    1 CALL CSM8A                                                        CSCA0200
      RETURN                                                            CSCA0210
    2           TC = T + 0.0005                                         CSCA0220
               CK1 = C(K1) + SIGN(0.00005,C(K1))                        CSCA0230
               CK2 = C(K2) + SIGN(0.00005,C(K2))                        CSCA0240
               CK3 = C(K3) + SIGN(0.00005,C(K3))                        CSCA0250
               CK4 = C(K4) + SIGN(0.00005,C(K4))                        CSCA0260
                 N = 0.5 + 100.0*(C(K4)-VMIN) / VDEL                    CSCA0270
      IF (N)   10,60,20                                                 CSCA0280
   10            N = 0                                                  CSCA0290
      GO TO 60                                                          CSCA0300
   20 IF (N - 100)  40,40,30                                            CSCA0310
   30            N = 100                                                CSCA0320
   40            N = N/2                                                CSCA0330
C        DASHES TO LEFT OF PLOTTED POINT, BLANKS TO RIGHT               CSCA0340
   50 DO 51      IND = 1,N                                              CSCA0350
   51   IPLOT(IND) = 24640                                              CSCA0360
   60          NP1 = N + 1                                              CSCA0370
               NP2 = NP1 + 1                                            CSCA0380
      IF ( NP2 - 50 )   61,61,63                                        CSCA0390
   61 DO 62      IND = NP2,50                                           CSCA0400
   62   IPLOT(IND) = 16448                                              CSCA0410
C        INDICATE MARGINS BY LETTER I                                   CSCA0420
   63    IPLOT( 1) = -14016                                             CSCA0430
         IPLOT(51) = -14016                                             CSCA0440
C        INDICATE PLOTTED POINT BY PLUS SIGN                            CSCA0450
        IPLOT(NP1) = 20032                                              CSCA0460
      WRITE (1, 201) TC,CK1,CK2,CK3,CK4,IPLOT                           CSCA0470
  201 FORMAT (F11.3,4F13.4,5X,51A1 )                                    CSCA0480
      RETURN                                                            CSCA0490
      END                                                               CSCA0500
```

```
// FOR
*ONE WORD INTEGERS
      SUBROUTINE CSM8A
C      ***** APRIL 1, 1966 *****
C     REQUEST PRINT INFORMATION
C        THIS ALTERNATE SUBPROGRAM IS TO BE USED IF THE 1627 PLOTTER
C        IS UNAVAILABLE  -- IT PROVIDES A PRINT-PLOT OUTPUT ON THE
C        1131 CONSOLE PRINTER
C
      REAL    REALS(395)
      INTEGER INTS(587),PRINT(4),TEST8
      COMMON  REALS, INTS
      EQUIVALENCE                             ( REALS( 78),    DT      )
      EQUIVALENCE  ( INTS(384),   KEY5     ) , ( REALS(391), TSAMP     )
      EQUIVALENCE  ( INTS(385),   KEY6     ) , ( REALS(392),  VDEL     )
      EQUIVALENCE  ( INTS(532),  TEST8     ) , ( REALS(394),  VMIN     )
      EQUIVALENCE  ( INTS(535), PRINT(1)   )
C
      GO TO (2,1), TEST8
    1 GO TO (2 ,4 ), KEY5
    2 WRITE (1,101)
      READ  (6,102) TSAMP
      IF ( TSAMP - DT ) 80,4,4
C
C         FIND OUT WHICH BLOCKS ARE TO BE PRINTED AND WHICH PLOTTED
C
    4 GO TO (6,5),TEST8
    5 GO TO (7,60), KEY6
    6        TEST8 = 2
    7 CONTINUE
      WRITE (1,106)
      READ  (6,107) PRINT(4),VMIN,VMAX
              VDEL = VMAX - VMIN
C
C             TEST IF BLOCK NUMBER TO BE PLOTTED IS BETWEEN 1 AND 75
C                   AND IF VMAX IS GREATER THAN VMIN
C
      IF ( PRINT(4) ) 15,15,9
    9 IF ( VDEL ) 15,15,10
   10 IF ( PRINT(4) - 76 ) 20,20,15
C
C             Y-AXIS SPECIFICATION ERROR
C
   15 WRITE (1,108)
      GO TO 7
C
   20 CONTINUE
C
C             READ IN VARIABLES TO BE PRINTED
C
      WRITE (1,103) PRINT(4),VMIN,VMAX
      READ  (6,105) ( PRINT(N), N=1,3)
C
C         IF BLOCK NUMBER IS NOT BETWEEN 1 AND 75, PRINT 0.0
C
      DO 50 N=1,3
      IF ( PRINT(N) ) 40,40,30
   30 IF ( PRINT(N) - 75 ) 50,50,40
   40     PRINT(N) = 0.0
   50 CONTINUE
      GO TO 70
   60 CONTINUE                                                          CSCB0620
            VMAX = VMIN + VDEL                                          CSCB0630
      WRITE (1,104) PRINT,VMIN,VMAX                                     CSCB0640
   70 RETURN                                                            CSCB0650
   80 WRITE (1,109)                                                     CSCB0660
      GO TO 2                                                           CSCB0670
  101 FORMAT (// 26H(          ) PRINT INTERVAL )                       CSCB0680
  102 FORMAT (F10.0)                                                    CSCB0690
  103 FORMAT (/5X,-TIME-, 3X, 3(3X,-OUTPUT(   )-) 3X,-OUTPUT-,I3, 6X,   CSCB0700
  1031F10.4, 30X, F10.4   )                                             CSCB0710
  104 FORMAT (/5X, 4HTIME, 3X, 4(3X,  6HOUTPUT,I3,1X) ,5X,F10.4,        CSCB0720
  1041        30X,F10.4)                                                CSCB0730
  105 FORMAT (11X, 3(11X, I2) )                                         CSCB0740
  106 FORMAT (//-(   ) BLOCK FOR Y-AXIS    (          ) MINIMUM VALUE   CSCB0750
  1061(          ) MAXIMUM VALUE-   )                                   CSCB0760
  107 FORMAT ( I3,22X,F10.0,21X,F10.0 )                                 CSCB0770
  108 FORMAT (//-ERROR IN Y-AXIS SPECIFICATION- )                       CSCB0780
  109 FORMAT (/-PRINT INTERVAL CAN NOT BE LESS THAN INTEGRATION INTERVALCSCB0790
  1091-   )                                                             CSCB0800
C                                                                       CSCB0810
      END                                                               CSCB0820
// DUP                                                                  CSCB0830
*STORE      WS  UA   CSM8A                                              CSCB0840
// JOB                                                                  CSCB0850
// FOR                                                                  CSCC0010
*ONE WORD INTEGERS                                                      CSCC0020
      SUBROUTINE CSM9                                                   CSCC0030
C      ***** MARCH 30, 1966 *****                                       CSCC0040
C                                                                       CSCC0050
C        THIS ALTERNATE SUBPROGRAM IS TO BE USED IF THE 1627 PLOTTER    CSCC0060
C        IS UNAVAILABLE  -- IT PROVIDES A PRINT-PLOT OUTPUT ON THE      CSCC0070
C        1131 CONSOLE PRINTER                                           CSCC0080
      RETURN                                                            CSCC0090
      END                                                               CSCC0100
// DUP                                                                  CSCC0110
*STORE      WS  UA  CSM9                                                CSCC0120
// JOB                                                                  CSCC0130
// FOR                                                                  CSDA0010
*ONE WORD INTEGERS                                                      CSDA0020
      SUBROUTINE SUB1                                                   CSDA0030
      RETURN                                                            CSDA0040
      END                                                               CSDA0050
// DUP                                                                  CSDA0060
*STORE      WS  UA  SUB1                                                CSDA0070
// JOB                                                                  CSDA0080
// FOR                                                                  CSDB0010
*ONE WORD INTEGERS                                                      CSDB0020
      SUBROUTINE SUB2                                                   CSDB0030
      RETURN                                                            CSDB0040
      END                                                               CSDB0050
// DUP                                                                  CSDB0060
*STORE      WS  UA  SUB2                                                CSDB0070
// JOB                                                                  CSDB0080
// FOR                                                                  CSDC0010
*ONE WORD INTEGERS                                                      CSDC0020
      SUBROUTINE SUB3                                                   CSDC0030
      RETURN                                                            CSDC0040
      END                                                               CSDC0050
// DUP                                                                  CSDC0060
*STORE      WS  UA  SUB3                                                CSDC0070
// JOB                                                                  CSDC0080
// FOR                                                                  CSDD0010
*ONE WORD INTEGERS                                                      CSDD0020
      SUBROUTINE SUB4                                                   CSDD0030
      RETURN                                                            CSDD0040
      END                                                               CSDD0050
// DUP                                                                  CSDD0060
*STORE      WS  UA  SUB4                                                CSDD0070
// JOB                                                                  CSDD0080
// FOR                                                                  CSDE0010
*ONE WORD INTEGERS                                                      CSDE0020
      SUBROUTINE SUB5                                                   CSDE0030
      RETURN                                                            CSDE0040
      END                                                               CSDE0050
// DUP                                                                  CSDE0060
*STORE      WS  UA  SUB5                                                CSDE0070
// JOB                                                                  CSDE0080
```

Sample Problem Deck

```
// *    IF 1627 PLOTTER IS AVAILABLE TO THE SYSTEM, REMOVE CARD         CSE00010
// *           NUMBER 8  ****  *LOCALCSM13  ****                        CSE00020
// *    IF NO 1627 PLOTTER IS AVAILABLE TO THE SYSTEM, REMOVE CARD      CSE00030
// *           NUMBER 7  ****  *LOCALCSM13,CSM9A,CSM9B  ****            CSE00040
// XEQ CSMP     2                                                       CSE00050
*LOCALCSMP,CSM0,CSM1,CSM2,CSM3,CSM4,CSM5,CSM6,CSM7,CSM8A,CSM11,CSM12,   CSE00060
*LOCALCSM13,CSM9A,CSM9B                                                 CSE00070
*LOCALCSM13                                                             CSE00080
FORCE              17        W        48         9                      CSE00090
ACCELERATION        4        /        17         6                      CSE00100
MASS                6        K                                          CSE00110
VELOCITY           48        I         4                                CSE00120
DISPLACEMENT        9        I        48                                CSE00130
                                                                        CSE00140
C AND K            17        2.00000        1.00000                     CSE00150
- MASS              6       -5.00000                                    CSE00160
INITIAL DISP.       9      -10.00000                                    CSE00170
                                                                        CSE00180
                                                                        CSE00190
                                                                        CSE00200
                                                                        CSE00210
                                                                        CSE00220
                                                                        CSE00230
                                                                        CSE00240
```

Printed in U.S.A. H20-0284-1

International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, N.Y. 10601
(USA Only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

IBM Technical Newsletter

Re: Form No.	H20-0284-1
This Newsletter No.	N20-1044-0
Date	August 3, 1967
Previous Newsletter Nos.	None*

1130 CONTINUOUS SYSTEM MODELING PROGRAM (1130-CX-13X)
SYSTEM MANUAL

Please remove pages 13-14, 15-16, 35-36, 65-66, 67-68, 73-74, 75-76, 79 through 88, and 95-96 from H20-0284-1 and replace them with the attached pages bearing the same page numbers.

Please file this newsletter at the back of H20-0284-1. It will provide a reference to changes, a method of determining that all amendments have been received, and a check for determining if the bulletin contains the proper pages.

*If this newsletter is applied to H20-0284-0 instead of H20-0284-1, be sure to apply Technical Newsletter N20-1015-0 also.

IBM Corporation, Technical Publications Dept., 112 E. Post Road, White Plains, N. Y. 10601

Printed in U.S.A.