NASA
National Aeronautics and
Space Administration

Lyndon B. Johnson Space Center
Houston. Texas 77058

JSC-14804

PROGRAM DOCUMENTATION

DEVELOPMENT OF THE SINDA TEMPERATURE INITIALIZATION PROGRAM

CPD 909

N79-77458

(NASA-CR-160184) DEVELOPMENT OF THE SINDA
TEMPERATURE INITIALIZATION PROGRAM (Lockheed
Electronics Co.) 13 p

Unclas
00/16 22204

Prepared by

Lockheed Electronics Company, Inc.
Systems and Services Division
Houston, Texas

Contract NAS 9-15800

for

STRUCTURES AND MECHANICS DIVISION

April 1979

LEC-13264

13

JSC-14804
CPD 909

PROGRAM DOCUMENTATION

DEVELOPMENT OF THE SINDA TEMPERATURE INITIALIZATION PROGRAM

Job Order 52-309

Prepared by

P E Fogerson

P. E. Fogerson

D. Raznahan

Approved by

J M Allred

J. M. Allred, Supervisor
Thermal Technology Section

W J Reicks

W. J. Reicks, Manager
Applied Mechanics Department

Lockheed Electronics Company, Inc.
Systems and Services Division
Houston, Texas

April 1979

LEC-13264

1. Report No. JSC-14804	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle
Development of the SINDA Temperature Initialization Program

5. Report Date
April 1979

6. Performing Organization Code

7. Author(s)
P. E. Fogerson
D. Raznahan Lockheed Electronics Company, Inc.

8. Performing Organization Report No.
LEC-13264

9. Performing Organization Name and Address
Lockheed Electronics Company, Inc.
1830 NASA Rd. 1, Houston, Texas, 77058

Technical Monitor: R. G. Brown

10. Work Unit No.

11. Contract or Grant No.
NAS 9-15800

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
Houston, Texas 77058

13 Type of Report and Period Covered
Program Documentation

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This document describes a program which builds a steady state SINDA model from a transient model for the purpose of obtaining a set of initial temperatures for the correlation of predictions with test data. The new model converts to boundary nodes those diffusion and arithmetic nodes which correspond to measurement locations.

17. Key Words (Suggested by Author(s))
Thermal Analysis
SINDA

18. Distribution Statement

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 14	22. Price* PC A02 MF A01

*For sale by the National Technical Information Service, Springfield, Virginia 22161

1. INTRODUCTION

Obtaining a good set of initial temperatures is one of the problems which will be encountered in the correlation of the predictions of the Shuttle Thermal Models with the Orbital Flight Test (OFT) data. Of course, for nodes simulating an area which has a thermocouple, the actual data will be used. The rest of the temperature matrix may be obtained in one of several ways. Considering the large number of measurements which are available and the size of the models (also large), the most accurate temperature distribution will be obtained by fixing the temperatures of the nodes with measurements and performing a steady state solution on the rest of the model. This will require converting many diffusion and arithmetic nodes to boundary nodes, which, if done manually, would be tedious at best. Hence, the SINDA Temperature Initialization Program (STIP) was developed. Using the OFT data, a measurement node correlation list, and the SINDA transient thermal model, this program builds a steady state model which has the nodes from the transient model with measured temperatures converted to boundary nodes. After execution of the steady state solution, the temperature data are saved for subsequent input to the transient model.

2. DISCUSSION

The software package which provides the capability of initializing a thermal model from OFT data consists of two main parts: (1) the program and subroutines which build the new steady state model from the old transient model, and (2) two SINDA subroutines; SAVETS, which stores the final conditions from the steady state run, and LOADTS, which is used in the transient model to retrieve the steady state results. The entire process is shown schematically in figure 1.

2.1 SINDA TEMPERATURE INITIALIZATION PROGRAM

The temperature initialization program builds a SINDA thermal model which is designed to compute a set of initial temperatures for an OFT correlation effort. The program requires the following input:

a. A SINDA model for which a set of initial temperatures is desired
b. A list which specifies the relationship of the nodes of the thermal model to the OFT thermocouple data
c. A data tape containing the temperature data for the thermocouples
d. A namelist specifying user option parameters

The diffusion and arithmetic nodes of the original thermal model (item a) which are listed in the measurement node correlation list (item b) are converted into boundary nodes. They are assigned temperatures that were recorded on the OFT data tape (item c) for the corresponding thermocouple as specified in the measurement node correlation list. Since the OFT tape contains the temperature history for the thermocouples, the times at which to take the temperatures and other execution control parameters are specified in the namelist $NAME.

The input, or original, model must be a SINDA model and be in card image format on a logical unit. The measurement node correlation list must also be on a logical unit. This list is in the same format as that used as input for the Trend Monitoring System (TMS) prediction tape building program and the batch plotting program; i.e., a nine-character thermocouple identification, followed by an integer specifying the number of nodes that correspond to that

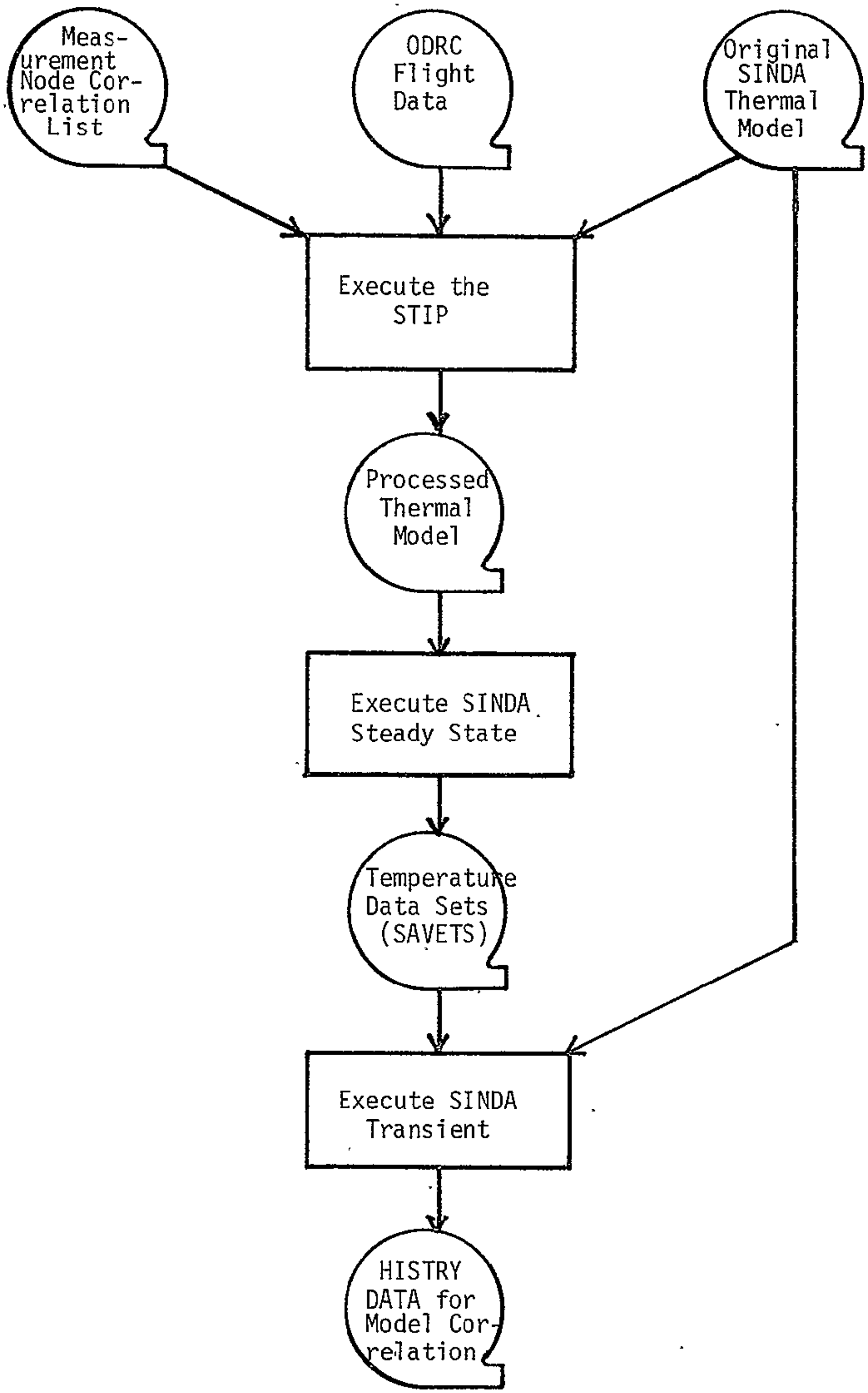

Measurement Node Correlation List
ODRC Flight Data
Original SINDA Thermal Model
Execute the STIP
Processed Thermal Model
Execute SINDA Steady State
Temperature Data Sets (SAVETS)
Execute SINDA Transient
HISTRY DATA for Model Correlation

Figure 1. - STIP operational flowchart.

thermocouple, and then the SINDA actual node numbers. All data are in free-field format; i.e., separated by commas. The node numbers are related to a particular thermal model by a six-character model name which separates the thermocouple, node correspondence cards. Figure 2 is a measurement-node correlation list.

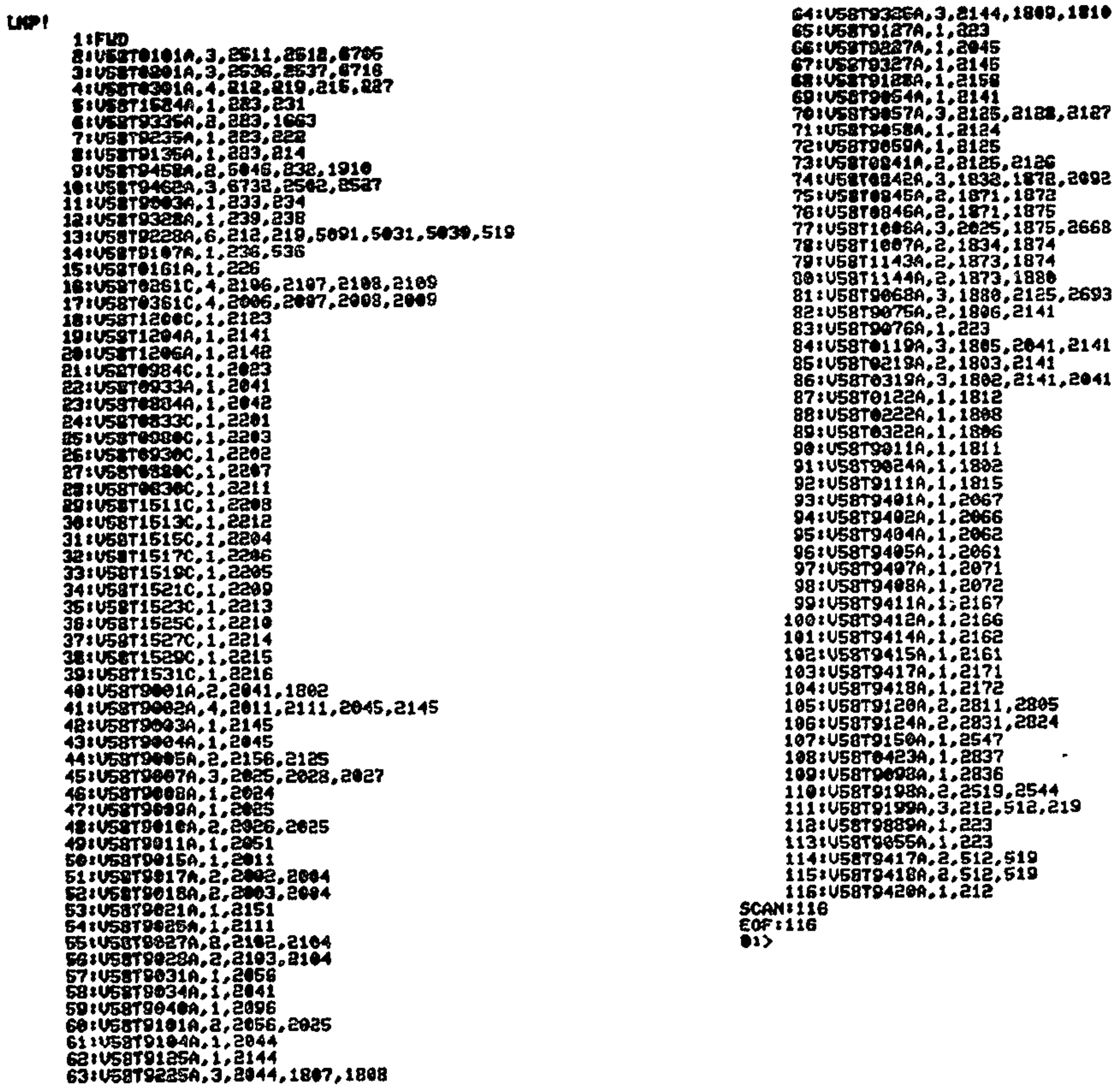
The thermocouple data tape must be a Computer-Compatible Tape (CCT) of the format normally output by the Orbital Data Reduction Center (ODRC). User input namelist $NAME provides control of the execution. Table I describes the input variables. The variable IMDOUT specifies the output unit for the processed model, and the variables IMODIN, IUNIT, and TMSNDC provide the program with the logical unit numbers for input of the original thermal model, the CCT from ODRC, and the measurement-node correlation list, respectively.

The variable ITIMEO is an ordered six-word array which determines the time at which the temperatures are to be taken. The input is six two-digit integers which specify year, month, day, hour, minute, and second in GMT. For example, for March 21, 1981, at 33 seconds past 11:45 a.m., input for ITIMEO would be 81, 03, 21, 11, 45, 33.

The six-character name specified by MODEL is used by the program to locate the section of the measurement-node correlation list which is to be used for this run. The list is searched until a card is found which has in its first six characters the same name specified by MODEL. Processing is terminated when a new model name or an end-of-file is encountered.

The variable PRINT controls the amount of printed output. For normal production runs, the default value of FALSE should be used. PRINT = .TRUE. is intended for checkout or diagnostic runs.

The processed model, i.e., the output model which has some nodes converted from diffusion and arithmetic to boundary, has the nodes in the same physical order; however, most multiple node cards will be unpacked. Thus, the node



```
LNP!
   1:FWD
   2:V58T0101A,3,2511,2512,0706
   3:V58T0201A,3,2536,2537,0716
   4:V58T0301A,4,212,219,216,227
   5:V58T1584A,1,283,231
   6:V58T9336A,2,283,1663
   7:V58T9235A,1,223,222
   8:V58T9135A,1,283,214
   9:V58T9458A,2,5046,232,1910
  10:V58T9462A,3,6732,2502,2527
  11:V58T9003A,1,233,234
  12:V58T9328A,1,239,238
  13:V58T9228A,6,212,219,5091,5031,5030,519
  14:V58T9107A,1,236,536
  15:V58T0161A,1,226
  16:V58T0261C,4,2106,2107,2108,2109
  17:V58T0361C,4,2006,2007,2008,2009
  18:V58T1200C,1,2123
  19:V58T1204A,1,2141
  20:V58T1206A,1,2142
  21:V58T0984C,1,2023
  22:V58T0933A,1,2041
  23:V58T0884A,1,2042
  24:V58T0833C,1,2201
  25:V58T0980C,1,2203
  26:V58T0930C,1,2202
  27:V58T0880C,1,2207
  28:V58T0830C,1,2211
  29:V58T1511C,1,2208
  30:V58T1513C,1,2212
  31:V58T1515C,1,2204
  32:V58T1517C,1,2206
  33:V58T1519C,1,2205
  34:V58T1521C,1,2209
  35:V58T1523C,1,2213
  36:V58T1525C,1,2210
  37:V58T1527C,1,2214
  38:V58T1529C,1,2215
  39:V58T1531C,1,2216
  40:V58T9001A,2,2041,1802
  41:V58T9002A,4,2011,2111,2045,2145
  42:V58T9003A,1,2145
  43:V58T9004A,1,2045
  44:V58T9005A,2,2156,2125
  45:V58T9007A,3,2025,2028,2027
  46:V58T9008A,1,2024
  47:V58T9009A,1,2025
  48:V58T9010A,2,2026,2025
  49:V58T9011A,1,2051
  50:V58T9015A,1,2011
  51:V58T9017A,2,2002,2004
  52:V58T9018A,2,2003,2004
  53:V58T9021A,1,2151
  54:V58T9025A,1,2111
  55:V58T9027A,2,2102,2104
  56:V58T9028A,2,2103,2104
  57:V58T9031A,1,2056
  58:V58T9034A,1,2041
  59:V58T9040A,1,2096
  60:V58T9101A,2,2056,2025
  61:V58T9104A,1,2044
  62:V58T9125A,1,2144
  63:V58T9225A,3,2044,1807,1808
  64:V58T9326A,3,2144,1809,1810
  65:V58T9127A,1,223
  66:V58T9227A,1,2045
  67:V58T9327A,1,2145
  68:V58T9128A,1,2158
  69:V58T9054A,1,2141
  70:V58T9057A,3,2125,2126,2127
  71:V58T9058A,1,2124
  72:V58T9059A,1,2125
  73:V58T0841A,2,2125,2126
  74:V58T0842A,3,1832,1878,2092
  75:V58T0845A,2,1871,1872
  76:V58T0846A,2,1871,1875
  77:V58T1006A,3,2025,1875,2668
  78:V58T1007A,2,1834,1874
  79:V58T1143A,2,1873,1874
  80:V58T1144A,2,1873,1880
  81:V58T9068A,3,1880,2125,2693
  82:V58T9075A,2,1806,2141
  83:V58T9076A,1,223
  84:V58T0119A,3,1805,2041,2141
  85:V58T0219A,2,1803,2141
  86:V58T0319A,3,1802,2141,2041
  87:V58T0122A,1,1812
  88:V58T0222A,1,1808
  89:V58T0322A,1,1806
  90:V58T9011A,1,1811
  91:V58T9024A,1,1802
  92:V58T9111A,1,1815
  93:V58T9401A,1,2067
  94:V58T9402A,1,2066
  95:V58T9404A,1,2062
  96:V58T9405A,1,2061
  97:V58T9407A,1,2071
  98:V58T9408A,1,2072
  99:V58T9411A,1;2167
 100:V58T9412A,1,2166
 101:V58T9414A,1,2162
 102:V58T9415A,1,2161
 103:V58T9417A,1,2171
 104:V58T9418A,1,2172
 105:V58T9120A,2,2811,2805
 106:V58T9124A,2,2831,2824
 107:V58T9150A,1,2547
 108:V58T0423A,1,2837
 109:V58T9098A,1,2836
 110:V58T9198A,2,2519,2544
 111:V58T9199A,3,212,512,219
 112:V58T9889A,1,223
 113:V58T9055A,1,223
 114:V58T9417A,2,512,519
 115:V58T9418A,2,512,519
 116:V58T9420A,1,212
SCAN:116
EOF:116
0:>
```

Figure 2. - Measurement node correlation list.

TABLE I. - $NAME INPUT VARIABLES

Name	Description	Range	Default
IMDOUT	Logical unit number for model output	1-29 except 5 and 6	8
IMODIN	Logical unit number for model input	1-29 except 6	7

ITIMEO(I)
I = 1, 6
	Time in Greenwich Mean Time (GMT) at which data is to be taken from CCT	integer (I2)	None

IUNIT	Logical unit number of the CCT	1-29 except 5 and 6	1
MODEL	Six-character model name that corresponds to name on measurement-node correlation list	6H______	None
PRINT	Printed output control =.TRUE. - full output =.FALSE. - partial output	.TRUE. or .FALSE.	.FALSE.
TMSNDC	Logical unit for measurement-node correlation list	1-29 except 6	11

data block might be somewhat longer physically. The relative actual dictionary will be ordered differently from the original model because the new boundary nodes will be shifted to follow the arithmetic nodes.

The remaining card images of the thermal model are unchanged from the original model except as follows:

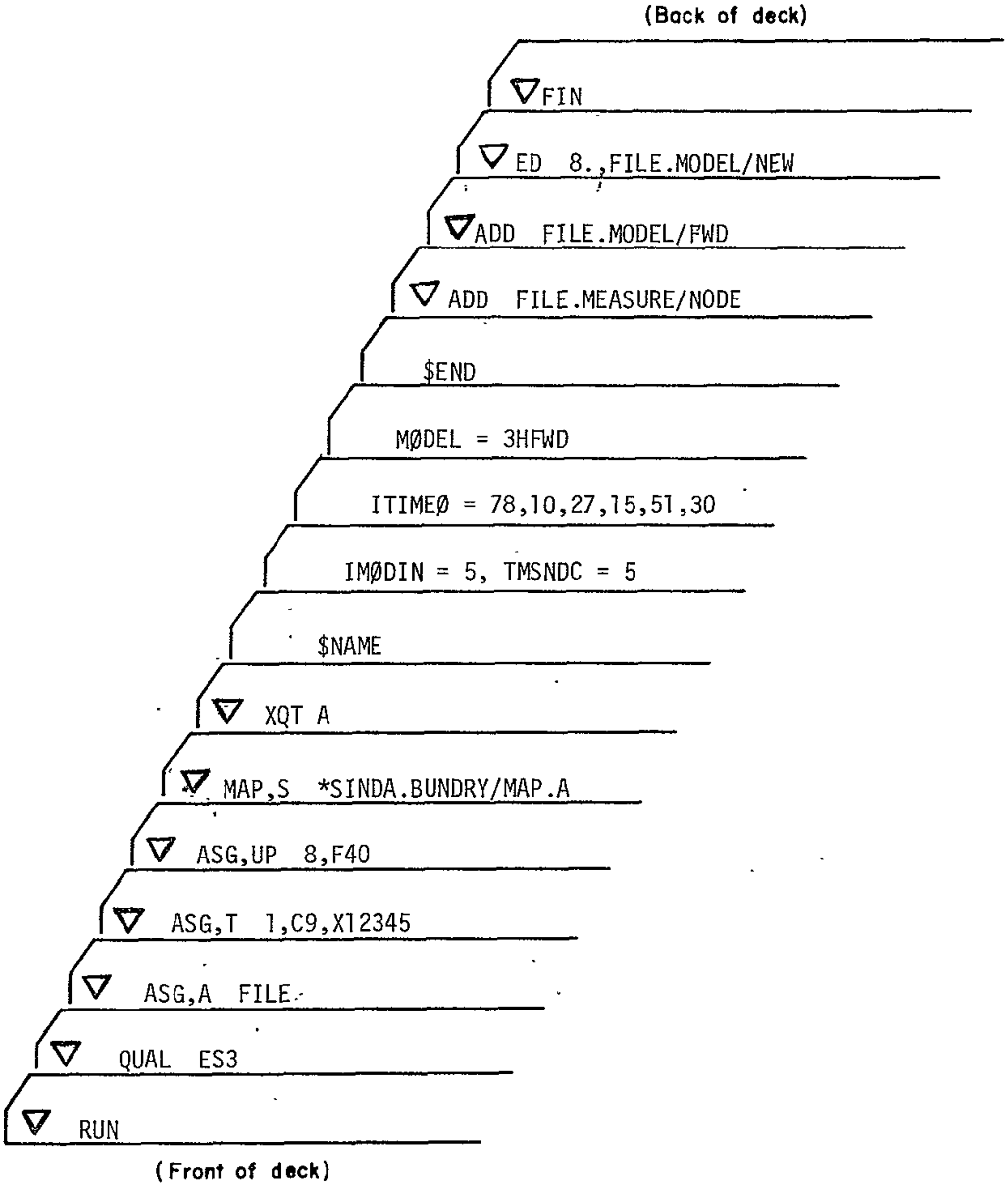
a. Any references to the converted nodes are deleted from the source data block.
b. The call to the transient execution routine in the execution block is deleted and replaced with these four cards:

```
F   ARLXCA = 0.01
F   DRLXCA = 0.01
            STDSTL
            SAVETS
```

c. All logic is deleted from variables 2
d. Any Q reference (heat input) to the converted nodes is deleted from the operations blocks.

The processed model is not expected to execute correctly without user modification, although there is no harm in making the attempt in the same run that does the conversion.

Figure 3 is an example of a runstream for the STIP. It assumes that the user has a secure file (FILE) which has a measurement-node correlation list and a thermal model as existing elements, and that the user wishes to store the processed model into FILE as an element. The ODRC tape is assigned to logical unit 1, the program default value, and a file is cataloged on logical unit 8 for the output model - also the default value. Since the thermal model and the measurement node correlation list are elements in FILE, the respective logical unit number entries in the Namelist IMODIN and TMSNDC are set to 5. Upon completion of namelist input, the measurement-node correlation list followed by the thermal model are made available to the program via the dynamic add. After the execution is completed, the text editor is used to store the processed model into FILE.



```
∇FIN
∇ED  8.,FILE.MODEL/NEW
∇ADD  FILE.MODEL/FWD
∇ ADD  FILE.MEASURE/NODE
 $END
  MØDEL = 3HFWD
   ITIMEØ = 78,10,27,15,51,30
    IMØDIN = 5, TMSNDC = 5
     $NAME
∇ XQT A
∇ MAP,S  *SINDA.BUNDRY/MAP.A
∇ ASG,UP  8,F40
∇ ASG,T  1,C9,X12345
∇ ASG,A  FILE.
∇ QUAL  ES3
∇ RUN
```

Figure 3. - Runstream for the SINDA temperature initialization program.

2.2 SINDA SUBROUTINES

Since the relative order of the nodes differs between the processed model and the original model, it is not possible to use the standard routine for saving temperatures, HSTFLO, and initializing a subsequent run, FLOTMP Hence, two new routines, SAVETS and LOADTS, were developed to perform these functions.

2.2.1 TEMPERATURE OUTPUT

After the call to the steady state execution routine, STDSTL, the program places a call to SAVETS for the purpose of saving the relative actual node dictionary and the temperatures which have just been calculated. The output of SAVETS is to unit 25, and is formatted in order that it may be examined by the text editor and/or stored in a file as an element. The dictionary is output first in a 10I8 format followed by the temperatures in 10F8.2. The file, unit 25, is rewound before output and terminated by a software EOF.

2.2.2 TEMPERATURE INPUT

Subroutine LOADTS was developed to allow model initialization to a set of temperatures which were stored by SAVETS. The user specifies the logical unit through an argument. The unit may be 5 if a corresponding dynamic add is placed correctly in the SINDA runstream. The routine searches the node dictionary from the logical unit for a correspondence to each node in the current model. If a match is not found, a message is printed but the initialization process is continued. This means that the model which had the SAVETS call does not have to be identical with the one being initialized. Thus SAVETS and LOADTS have applications beyond the STIP. LOADTS required dynamic storage equal to the total number of nodes of the current model (NNT_c) plus twice the total number of nodes of the larger of the current model or the input model (NNT_i), i.e.,

$$\text{dynamic storage} = NNT_c + 2*MAX(NNT_c,NNT_i)$$

The calling sequence is:

LOADTS (I)

where I is the logical unit containing the node dictionary and temperatures. The unit I is automatically released by the program.

3. SUMMARY

The STIP was not really expected to result in a model which, with some user changes, was completely satisfactory for a steady state execution. Perhaps the area which deserves most scrutiny is the heating rate inputs. The program deletes those references to converted nodes from the Source and Variable 1 blocks but leaves the other heating inputs unchanged. As this may not be acceptable for certain conditions, engineering evaluation is required. The user may also want to adjust the control parameters for the steady state routine, or change routines.

STIP will require less than 10 sups for normal applications. The runstream given in figure 3 when applied to a 5513 card image model of the forward fuselage required 00:03:12.65 sups. The program uses 10,595 words for instructions and 19,733 data locations, or total core requirements of 30,328. The program allows only 1200 nodes to be converted to boundary types.