NASA Contractor Report 2858

Random Harmonic Analysis Program, L221 (TEV156)

Volume II: Supplemental System Design and Maintenance Document

M. L. Graham, R. E. Clemmons, and R. D. Miller

CONTRACT NAS1-13918
SEPTEMBER 1979

NASA

NASA Contractor Report 2858

Random Harmonic Analysis Program, L221 (TEV156)

Volume II: Supplemental System Design and Maintenance Document

M. L. Graham, R. E. Clemmons, and R. D. Miller
Boeing Commercial Airplane Company
Seattle, Washington

Prepared for
Langley Research Center
under Contract NAS1-13918

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1979

CONTENTS

Page

1.0 SUMMARY 1

2.0 INTRODUCTION 1

3.0 PROGRAM STRUCTURE AND DESCRIPTION 3
- 3.1 Overlay (L221,0,0) - L221vc 5
- 3.2 Overlay (L221,1,0) - FINDRMS 10
- 3.3 Overlay (L221,2,0) - SORTQLS 24
- 3.4 Overlay (L221,3,0) - PLOTQLS (Available Only at Boeing) 32
- 3.5 Data Bases 42
  - 3.5.1 Input Data 42
  - 3.5.2 Output Data 42
  - 3.5.3 Internal Data 42

4.0 EXTENT OF CHECKOUT 105

REFERENCES 108

TABLES

No.		Page
1	Routines Called by L221vc	9
2	Routines Called by FINDRMS	19
3	Routines Called by SORTQLS	30
4	Routines Called by PLOTQLS	38
5	Common Blocks Used in Each Overlay	45
6	Common Blocks Used in the Routines of L221vc	46
7	Common Blocks Used in the Routines of FINDRMS	47
8	Common Blocks Used in the Routines of SORTQLS	48
9	Common Blocks Used in the Routines of PLOTQLS	49
10	Description of the Labeled Common Blocks	50
11	Input Constants and Frequencies of Block I	95
12	Arrays Contained in Block II	96
13	Arrays Contained in Block III	98
14	Options Used in Checkout Cases	106

FIGURES

No.		Page
1	DYLOFLEX Flow Chart	2
2	Overlay Structure of L221 (TEV156)	4
3.	External Input/Output of L221 (TEV156)	4
4.	Macro Flow Chart of Overlay (L221,0,0) – L221vc	6
5	Macro Flow Chart of Overlay (L221,1,0) – FINDRMS	11
6	Overlay (L221,1,0) FINDRMS Input/Output	23
7	Macro Flow Chart of Overlay (L221,2,0) – SORTQLS	26
8	Overlay (L221,2,0) SORTQLS Input/Output	31
9	Macro Flow Chart of Overlay (L221,3,0) – PLOTQLS	33
10	Overlay (L221,3,0) PLOTQLS Input/Output	41
11	Contents of SCRATCH – A Temporary Scratch File	43
12	Contents of SCRAT2 – A Temporary Scratch File	44

1.0 SUMMARY

Program L221 (TEV156) is structured as four overlays, one main and three primary. Input into the program is made via cards and magnetic files (tapes or disks). Output from the program consists of printed results and magnetic files containing vectors suitable for plotting.

Although L221 serves as a module of the DYLOFLEX system, it can be operated as a standalone program. Subroutines used by L221 include routines embedded in the program code, routines obtained from the standard FORTRAN library, and routines obtained from the DYLOFLEX alternate library.

2.0 INTRODUCTION

Program L221 (TEV156) was developed for use as either a standalone program or as a module of a program system called DYLOFLEX (see fig. 1), developed for NASA under contract NAS1-13918 (ref. 1). Because of the DYLOFLEX contract requirements developed in reference 2, a program was needed to calculate power spectral density (PSD) gust load parameters for equations of motion and load equations that have frequency-dependent coefficients. An existing program[1] that calculated PSD gust load parameters for equations with constant coefficients was modified according to the DYLOFLEX specifications[2] to also use nonconstant coefficients matrices for the equations of motion and load equations.

The objective of this volume is to aid those persons who will maintain and/or modify the program in the future. To meet this objective, the following items are defined in some detail:

- Program design and structure
- Overlay purpose and description
- Input, output, and internal data base descriptions
- Test cases and procedures

---

[1]Clemmons, R. E.: *A Power Spectral Digital Computer Program to Determine Dynamic Loads Due to Random Gusts–PSDSYS (TEV156) – Users Guide.* BCS-G025, June 1973.
[2]Clemmons, R. E.: *Programming Specifications for Modules of the Dynamic Loads System to Interface with FLEXSTAB.* NASA contract NAS1-13918. BCS-G0701, September 1975.

(Internal Documents.)

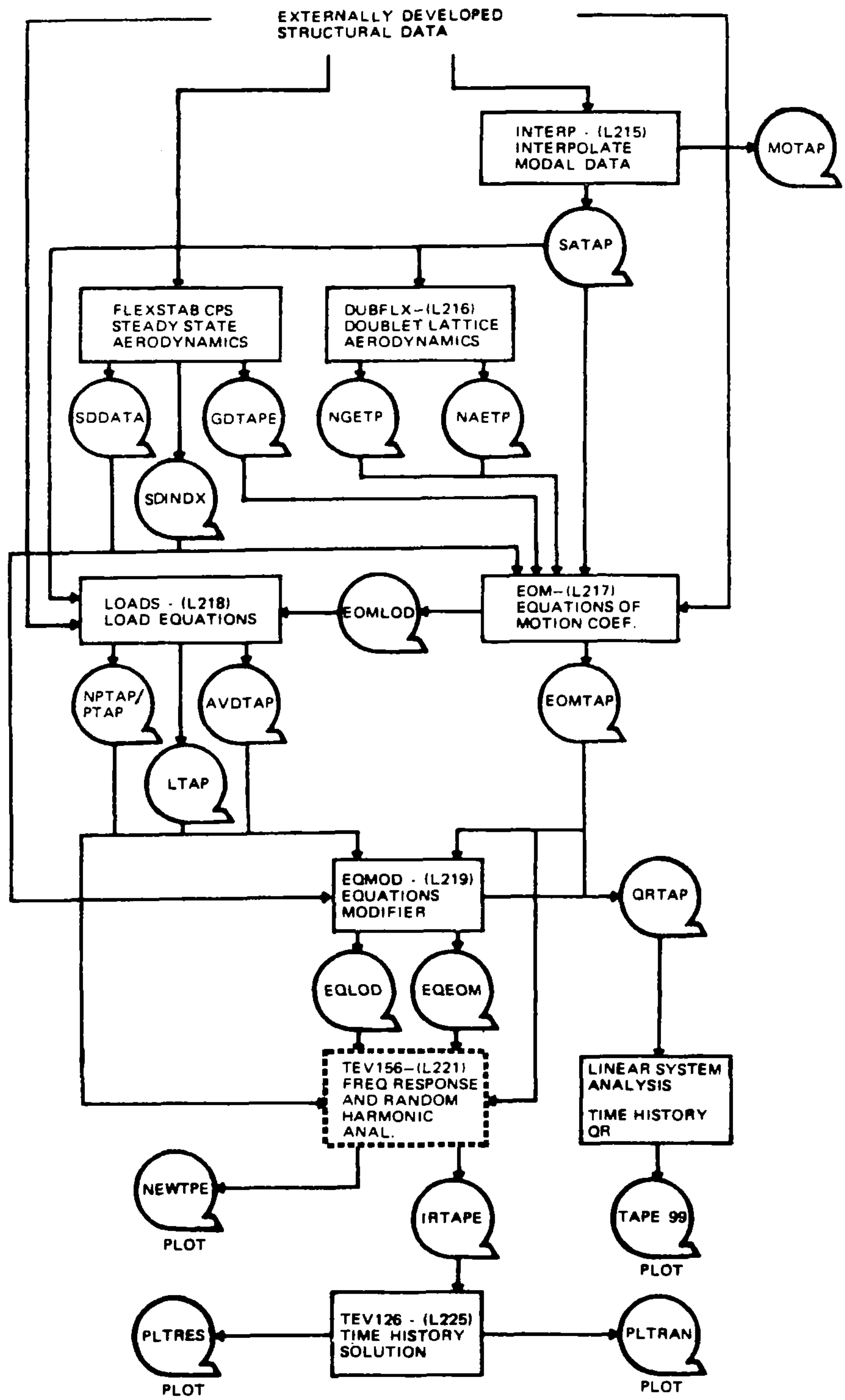


Figure 1.–DYLOFLEX Flow Chart

3.0 PROGRAM DESIGN AND STRUCTURE

The program is structured as a system of four overlays (fig. 2).

Main overlay (L221,0,0) L221vc
Primary overlay (L221,1,0) FINDRMS
Primary overlay (L221,2,0) SORTQLS
Primary overlay (L221,3,0) PLOTQLS

The main overlay L221 reads cards that direct the execution of the primary overlays. It also aids communication between the primary overlays via labelled common blocks.

The 1,0 primary overlay FINDRMS performs all of the analytical calculations of the program. Optionally, its results are written onto magnetic files (tape or disk) to be processed by the two other primary overlays.

The 2,0 primary overlay SORTQLS sorts the FINDRMS results (responses – Q's, loads, output spectrum, and $\bar{A}$ values) and prepares a magnetic file (tape or disk) of data vectors suitable for plotting.

The 3,0 primary overlay PLOTQLS sorts the FINDRMS results exactly the same as SORTQLS, but writes a magnetic file of plot data and plotting instructions geared specifically to the COMp80 plotter.

The input and output data for L221 (TEV156) is displayed in figure 3. Each overlay is discussed in detail in succeeding sections. Included for each overlay are:

1. The overlay's purpose
2. The overlay's analytical steps
3. The input/output devices used
4. A macro flow chart
5. Table of subroutines called
   (Note: All subroutines have only one entry point)

Special symbols are used on the tables of subroutines called to indicate routines that are loaded from the operating system library and from the DYLOFLEX alternate subroutine library (DYLIB). All other subroutines are local to L221 (TEV156).

This document does not contain a description of each local subroutine. Please see the program listing, where each routine contains "internal documentation," comments describing its purpose, operation, input, output, modification history, etc.

The storage and handling of data is discussed in section 3.5, Data Bases. All magnetic file (tape or disk) communication outside the program is done via READTP/ WRTETP[1] To achieve the most efficient use of core, a scheme to allow variably dimensioned arrays (dynamic storage allocation) was used. Section 3.5 also displays the variably dimensioned arrays (matrices) and describes the subroutines that keep track of them.

---

[1]Clemmons, R. E.: *Programming Specifications for Modules of the Dynamic Loads System to Interface with FLEXSTAB*. NASA contract NAS1-13918, BCS-G0701, September 1975. (Internal Document.)

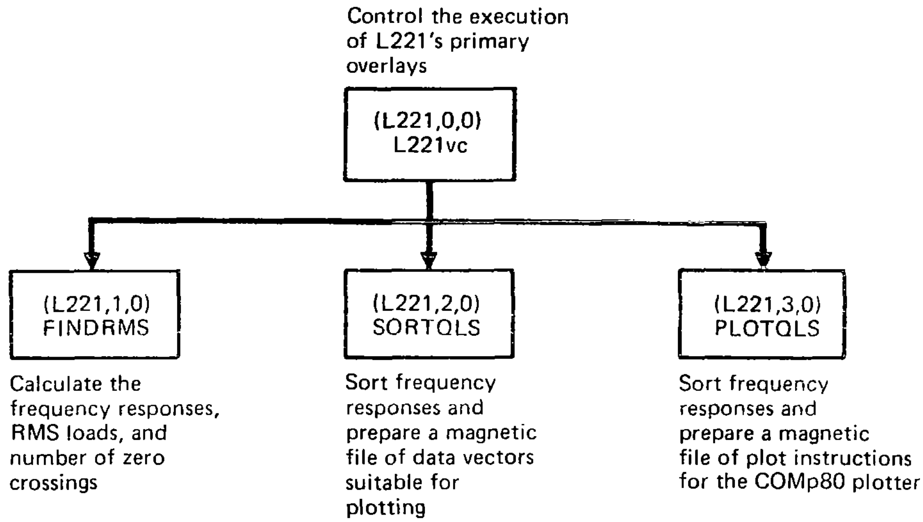


Figure 2.—Overlay Structure of L221 (TEV156)

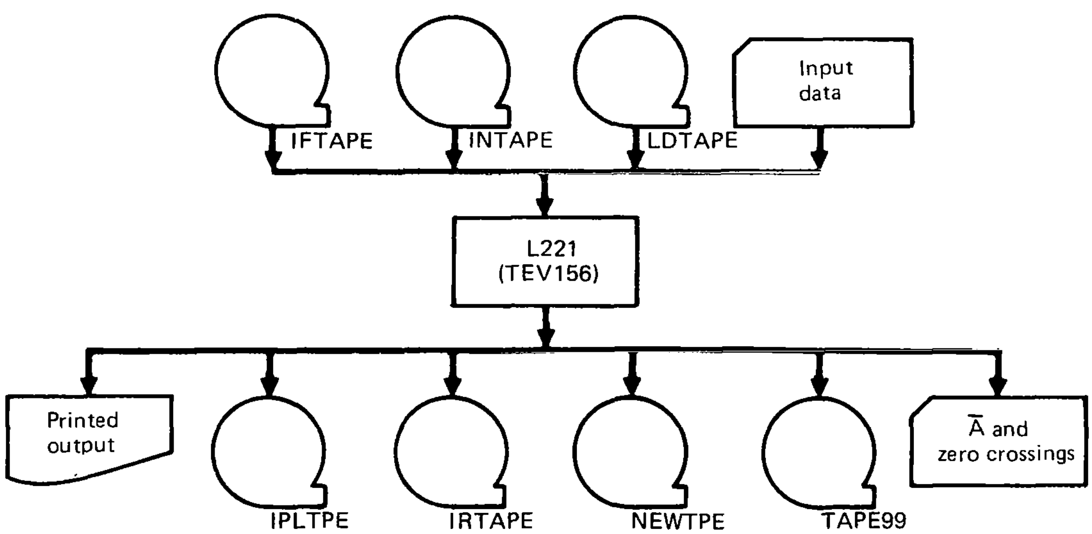


Figure 3.—External Input/Output of L221 (TEV156)

3.1 OVERLAY (L221,0,0) - L221vc

The main overlay of L221 (TEV156) is L221vc, where v is a letter indicating the program version, and c is an integer number indicating the correction that applies to the v version.

Purpose of L221vc

L221vc performs certain bookkeeping tasks, directs the execution of the primary overlays, and aids communication between primary overlays via labeled common blocks.

Analytical Steps of L221vc

L221vc performs its task in the following steps:

1. The subroutine PRGBEG is called to place the program header on the printed output.

2. A data card is read. It must begin with $FREQ to assure that the card input file is correctly positioned. If it does not contain $FREQ, execution is terminated.

3. A program directive card is read, printed, interpreted, and acted upon according to the following logic:

   a. If the keyword is $TITLE jump to step 3 again.
   b. If the keyword is $CHECKOUT, set the checkout switch and jump to step 3 again
   c. If the keyword is $FIND, jump to step 4.
   d. If the keyword is $SORT, jump to step 5.
   e. If the keyword is $PLOT, jump to step 6.
   f. If the keyword is $QUIT jump to step 7.

4. Overlay (L221,1,0) is called. When it is finished program control returns to step 3.

5. Overlay (L221,2,0) is called. When it is finished program control returns to step 3.

6. Overlay (L221,3,0) is called. When it is finished program control returns to step 3.

7. Subroutine PRGEND is called. This subroutine places the program termination message on the printed output.

If fatal errors are discovered, the program prints a diagnostic and jumps to step 7. The macro flow chart of this overlay is shown in figure 4. The subroutines called are displayed in table 1.

I/O Devices of L221vc

L221vc reads program directive cards and writes them on the printed output file. All other I/O accomplished by L221 (TEV156) is done in the primary overlays.

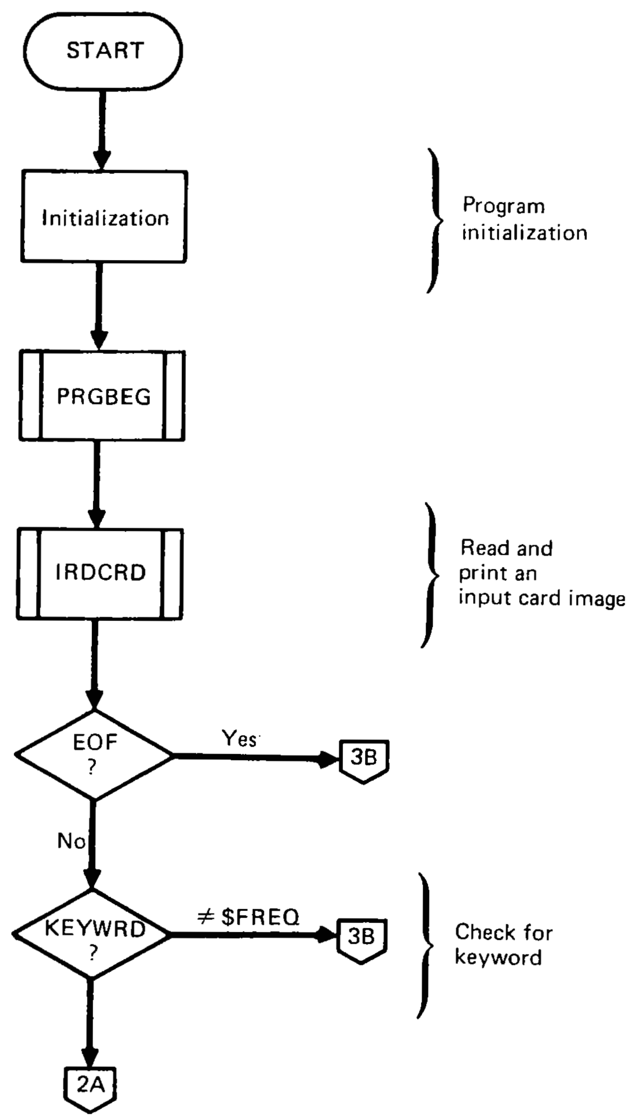


Figure 4.—Macro Flow Chart of Overlay (L221,0,0)—L221vc

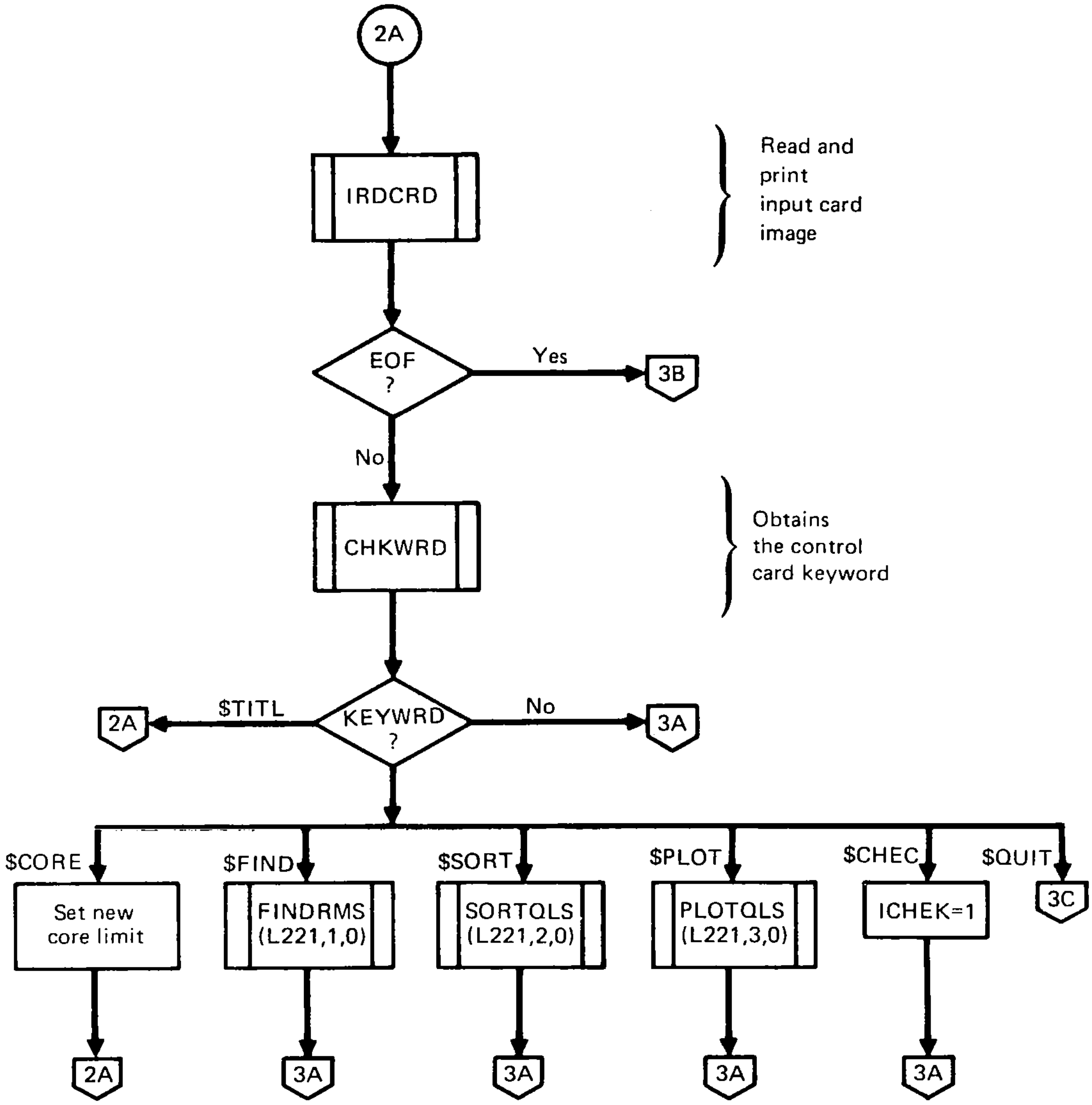


Figure 4.—(Continued)

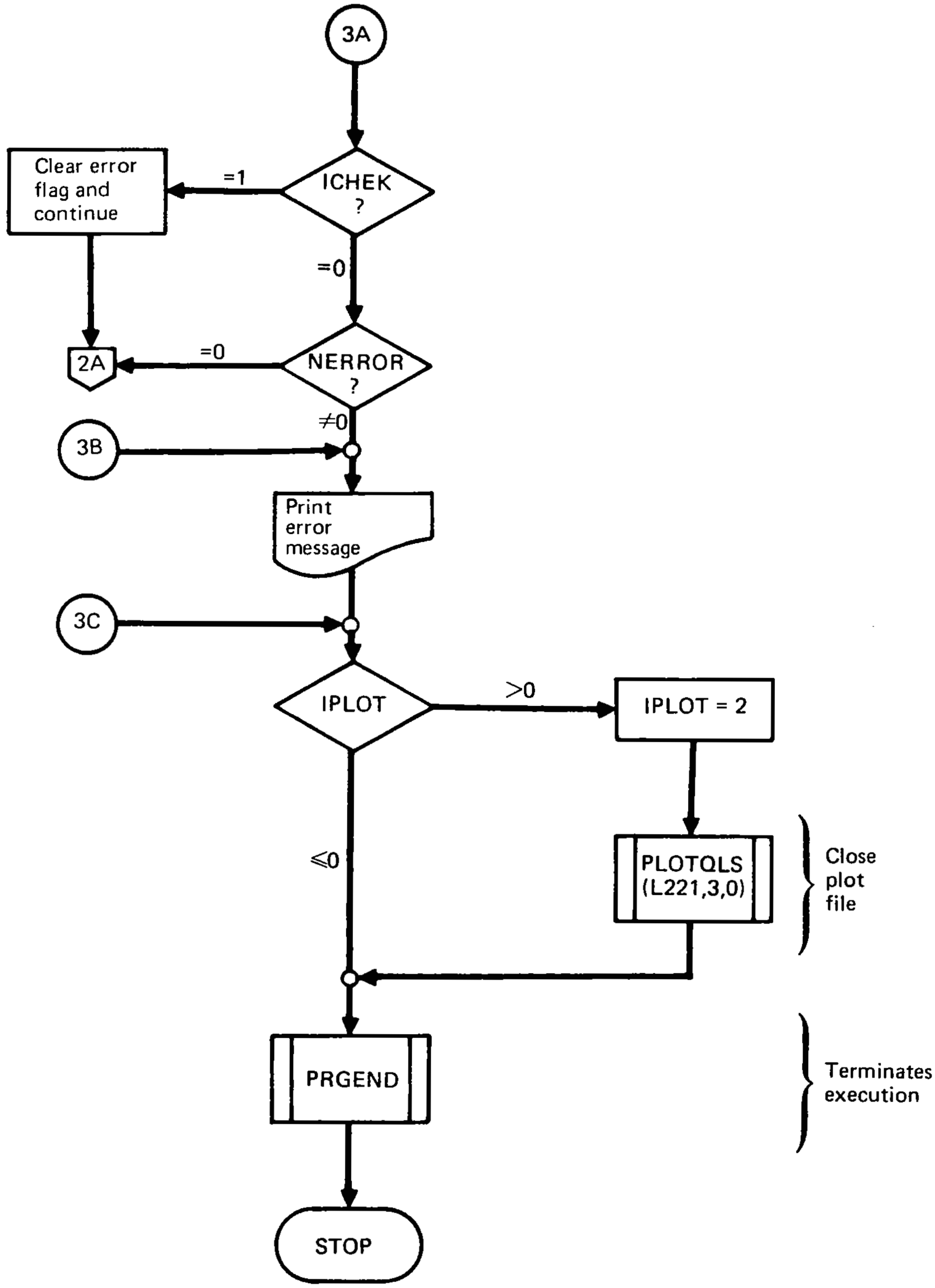


Figure 4.– (Concluded)

Table 1.—Routines Called by L221vc

```
OVERLAY (L221,0,0)
PROGRAM L221vc
-----------------------------------------------------------------

CHKWRD
FINDRMS - OVERLAY(L221,1,0)
IRDCRD+
PLOTQLS - OVERLAY(L221,3,0)
PRGBEG+
PRGEND+
REQFL+
SORTQLS - OVERLAY(L221,2,0)

---------------
+  indicates a routine on the DYLOFLEX alternate
   subroutine library DYLIB.
All others are local to L221 (TEV156).
```

3.2 OVERLAY (L221,1,0) - FINDRMS

Purpose of FINDRMS

The first primary overlay of L221 (TEV156) is FINDRMS. FINDRMS performs all of the analytical tasks of L221. Given an input power spectrum $\phi(\Omega)$ over a range of frequencies (Ω_1, Ω_2, ..., Ω_{NFREQ}), matrices of equations of motion and load equations, and excitation forces, FINDRMS will find:

$\{Q\}$	The generalized coordinates at each frequency
$\{SUMC\}$	The load transfer function at each frequency
	and integrate over the range of frequencies to find:
$\{\bar{A}\}$	The RMS load values
$\{N_0\}$	The number of zero crossings per unit length

Analytical Steps of FINDRMS

The FINDRMS operations are divided into the following steps:

1. Read card input of options and constants.
2. Read card or tape input of equations of motion ($[M_1]$, $[M_2]$, ..., $[M_6]$ and $[S_1]$, $[S_2]$, ..., $[S_6]$ for feedback) plus the excitation forces ($\{C_2\}$ and $\{C_3\}$ or $\{f_\ell\}$ and $[\tilde{\Phi}]$). This includes $\{FREQM\}$ frequencies if NKVAL > 0.
3. For each of the NFREQ frequencies, solve for $\{Q\}$, the generalized coordinates, save $\{Q\}$ on the scratch file, and optionally print $\{Q\}$.
4. (Optional) - Modify the equations of motion for a static elastic solution and repeat step 3. Chosen columns of the matrices $[M_2]$, $[M_3]$, $[M_5]$ and $[M_6]$ are set equal to zero for the static elastic solution.
5. Read card or tape input of load equations ($[\bar{M}_1]$, $[\bar{M}_2]$, ..., $[\bar{M}_6]$ and $[\bar{S}_1]$, $[\bar{S}_2]$, $[\bar{S}_3]$, ..., $[\bar{S}_6]$ for feedback) plus the matrices $\{\bar{C}_2\}$ and $\{\bar{C}_3\}$ or $[\bar{\Phi}]$.
6. For each frequency read $\{Q\}$ (from step 3), calculate $\{\overline{SUMC}\}$, the input spectrum $\phi(\Omega)$, and the output spectrum SPEC, optionally write data on files IPLTPE and IRTAPE, optionally print $\{\overline{SUMC}\}$, and keep a running integration over the frequencies of $\{\bar{A}\}$ and $\{N_0\}$.
7. Print $\{\bar{A}]$ and $\{N_0\}$ and optionally punch them on cards.
8. (Optional) - Check the correlation between the different load responses requested by card input data.
9. (Optional) - Modify the load equations for a static elastic solution (set chosen columns of $[\bar{M}_2]$, $[\bar{M}_3]$, $[\bar{M}_5]$ and $[\bar{M}_6]$ to zero) and repeat steps 6, 7, and 8.
10. (Optional) - For additional sets of load equations repeat steps 5 through 9.

Figure 5 contains a macro flow chart of FINDRMS. The subroutines called by FINDRMS are displayed in table 2.

I/O Devices of FINDRMS

The possible I/O devices for FINDRMS are shown in figure 6. For a complete description of the input data cards and magnetic files used, see sections 6.3 and 6.4 in volume I of this document.

The two scratch files, SCRATCH and SCRAT2, are described in section 3.5.3.

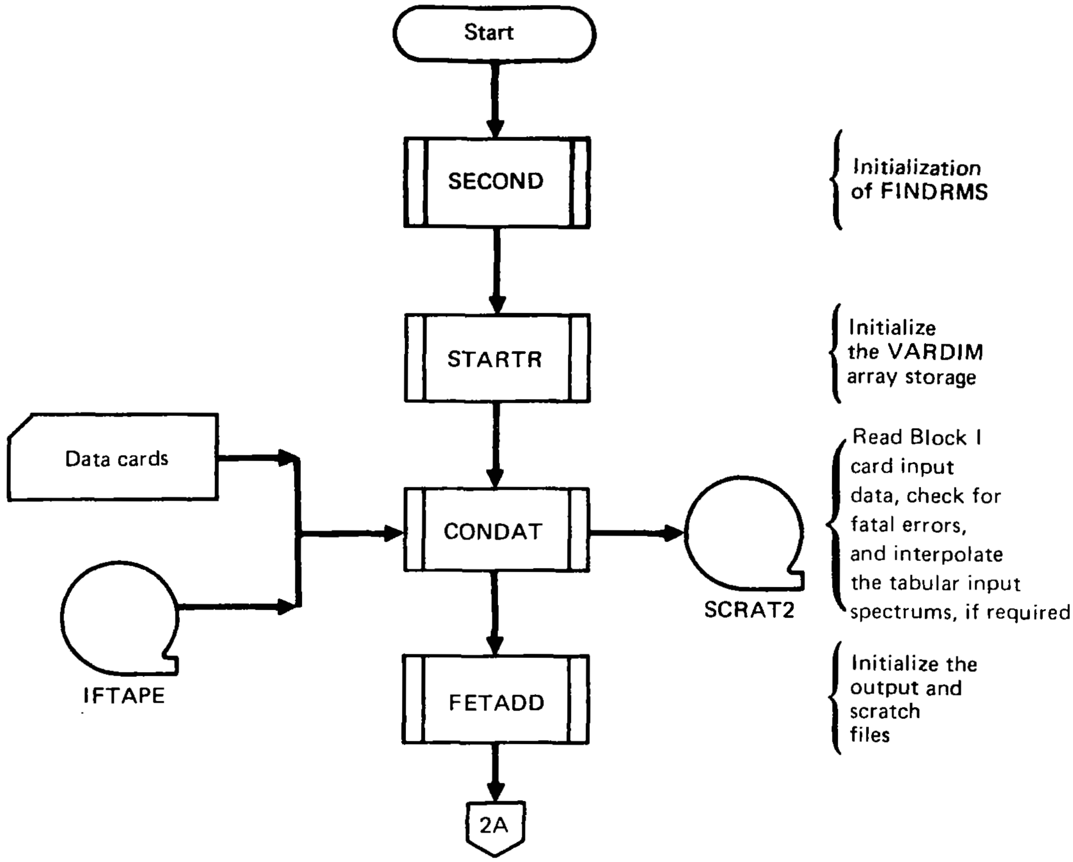


Figure 5.—Macro Flow Chart of Overlay (L221,1,0)—FINDRMS

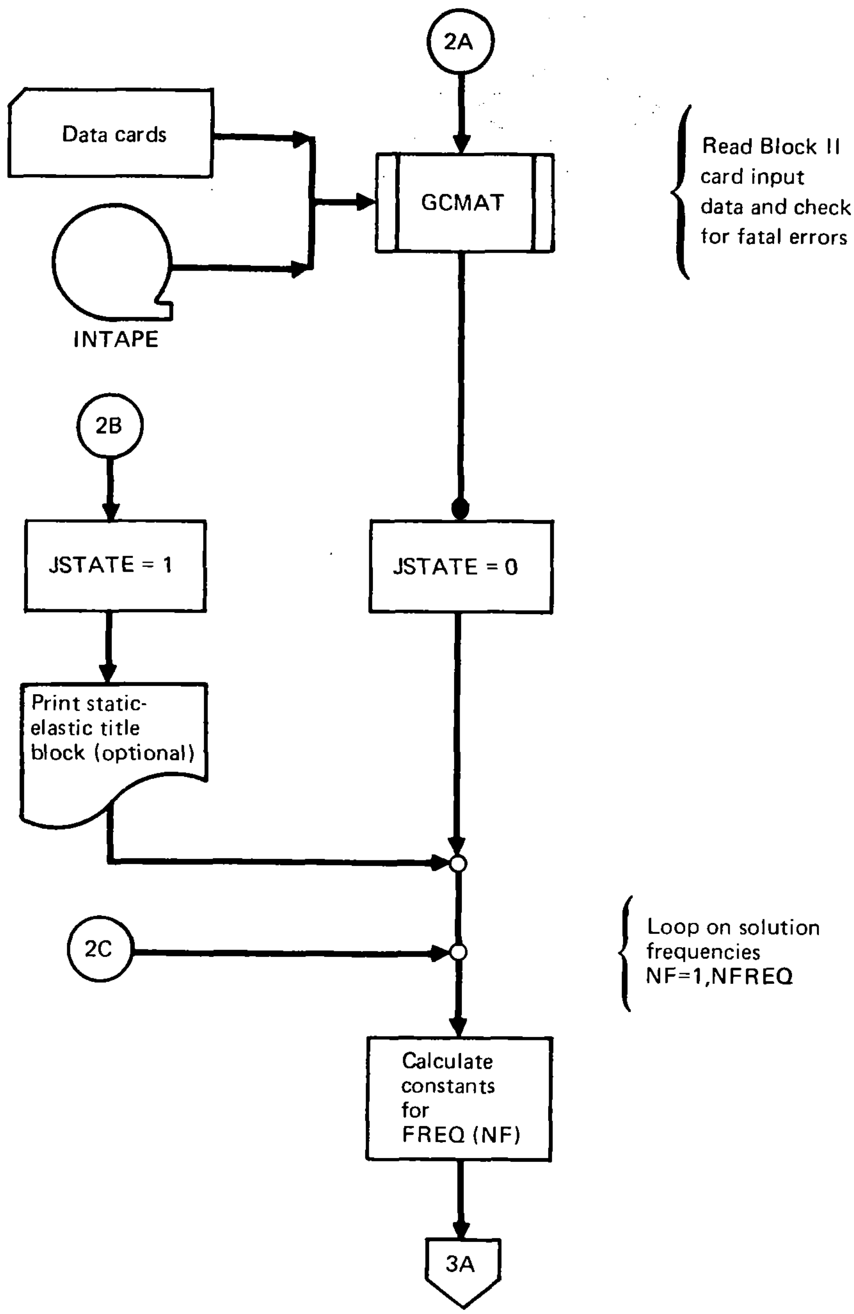


Figure 5.—(Continued)

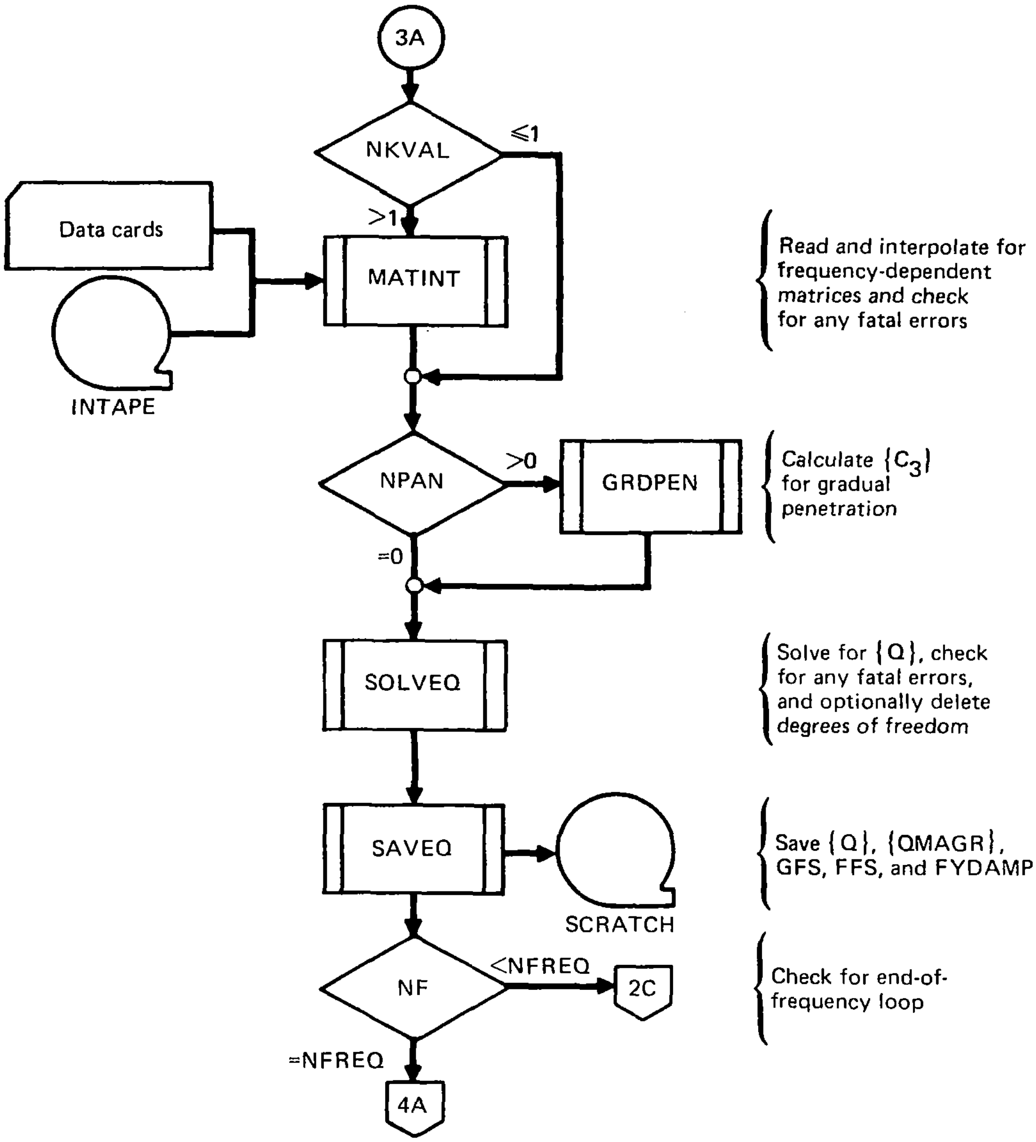


Figure 5.—(Continued)

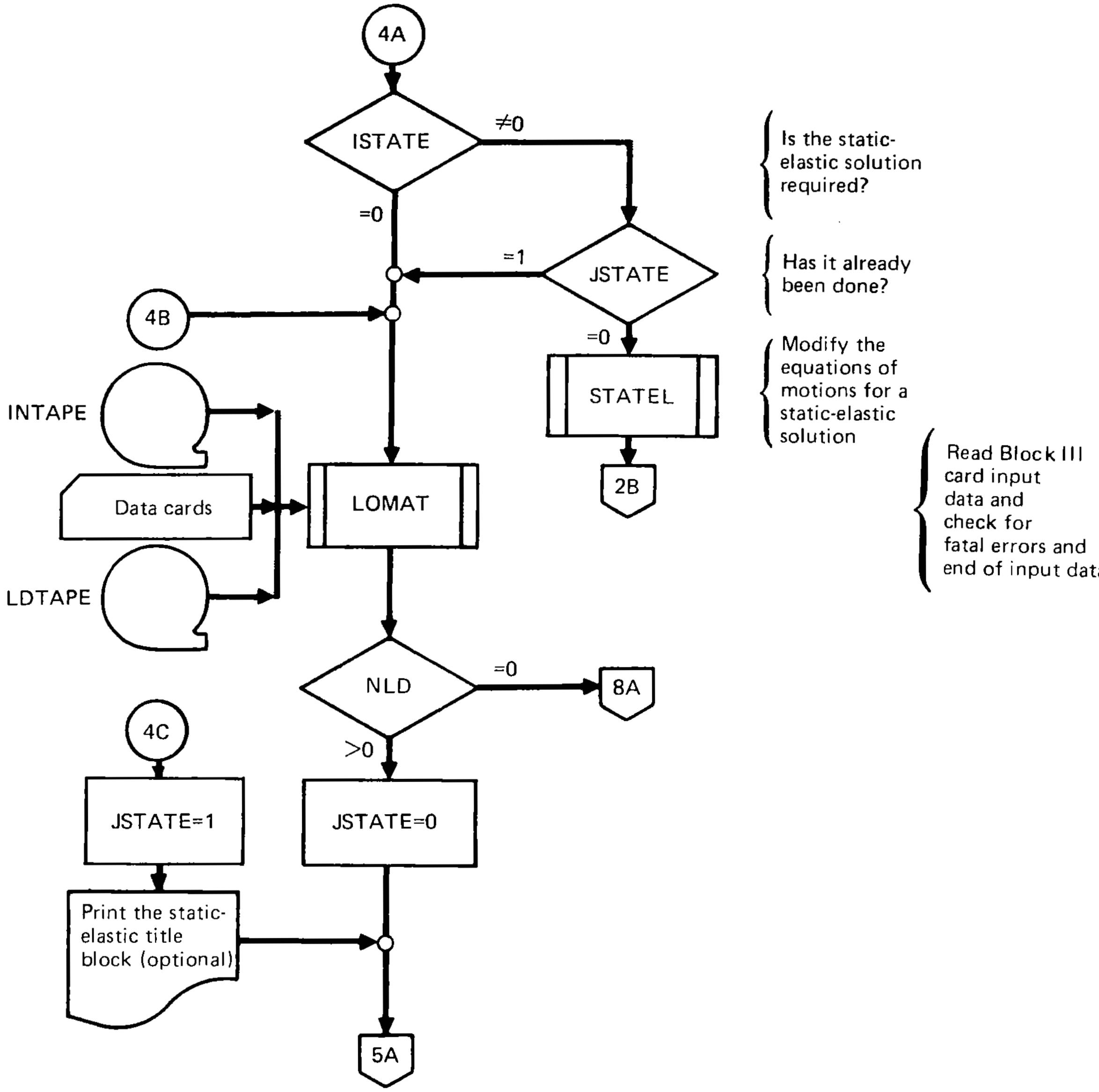


Figure 5.—(Continued)

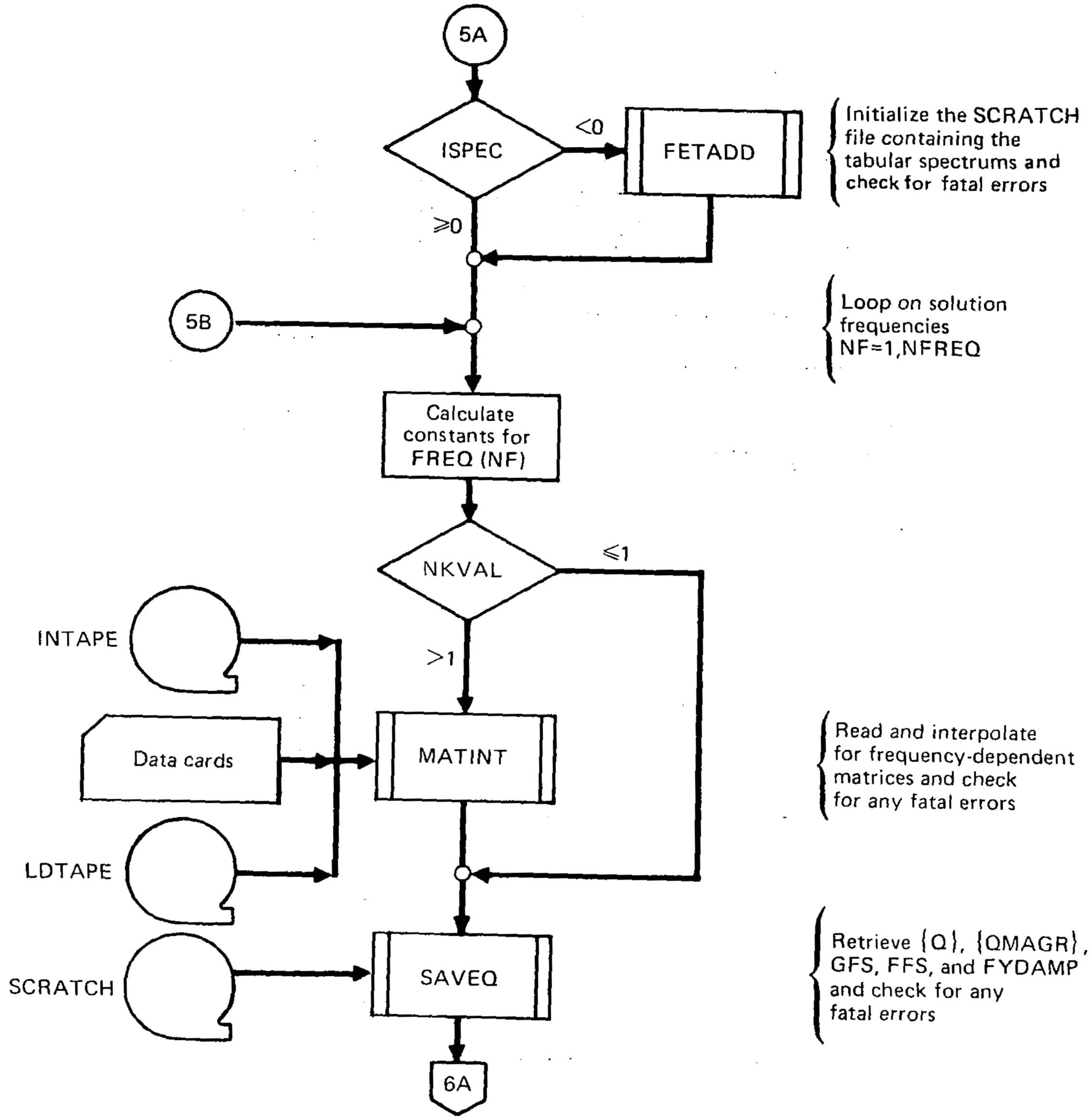


Figure 5.—(Continued)

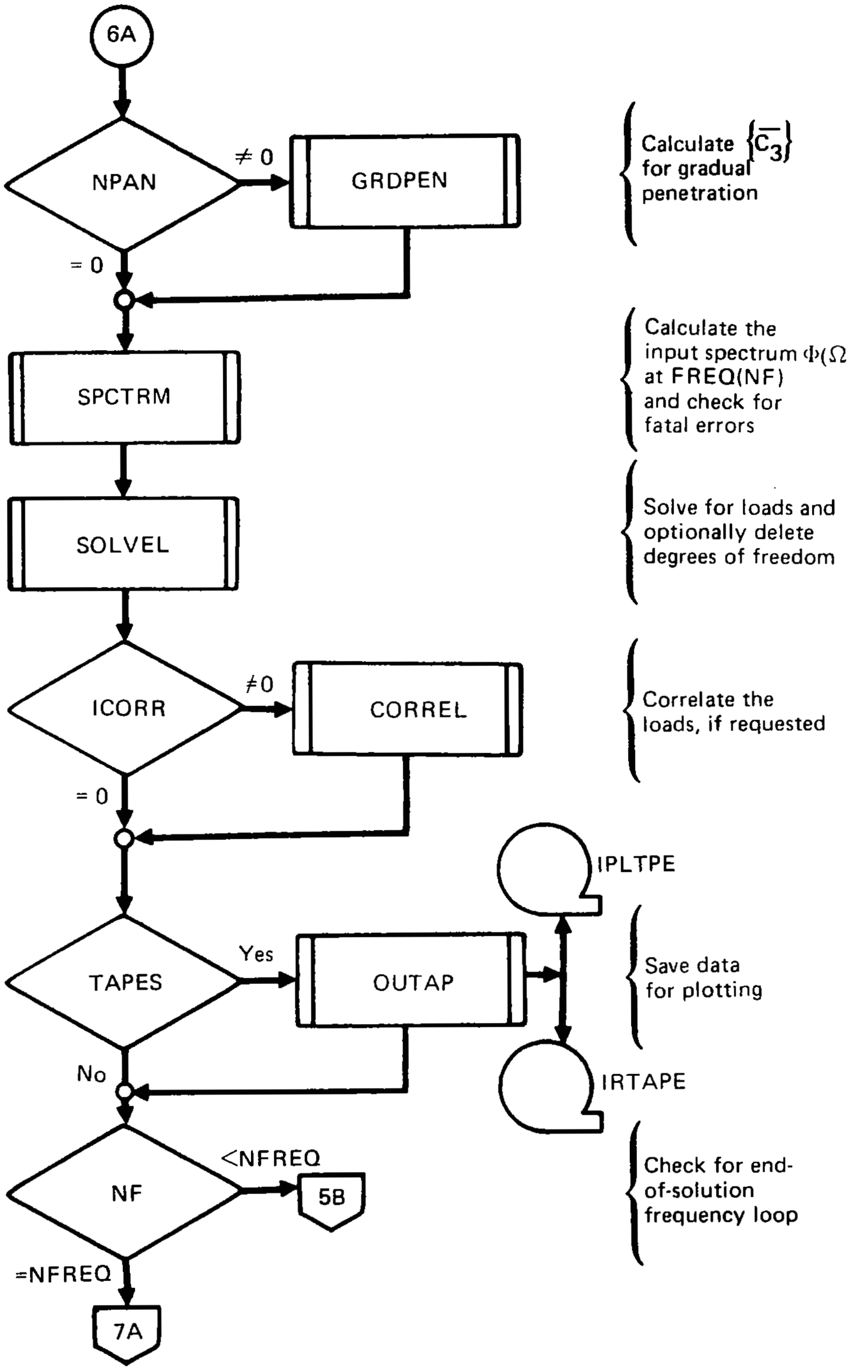


Figure 5.—(Continued)

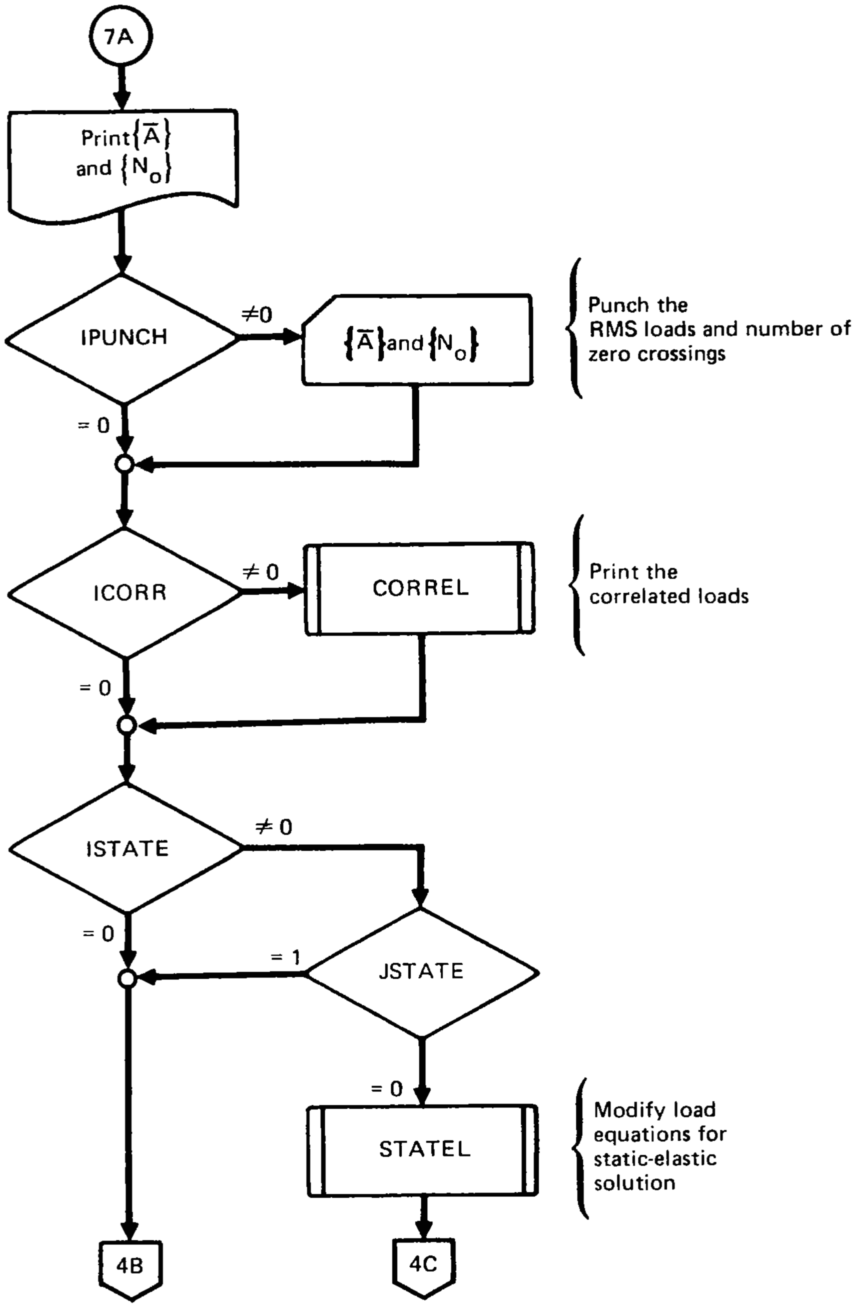


Figure 5.—(Continued)

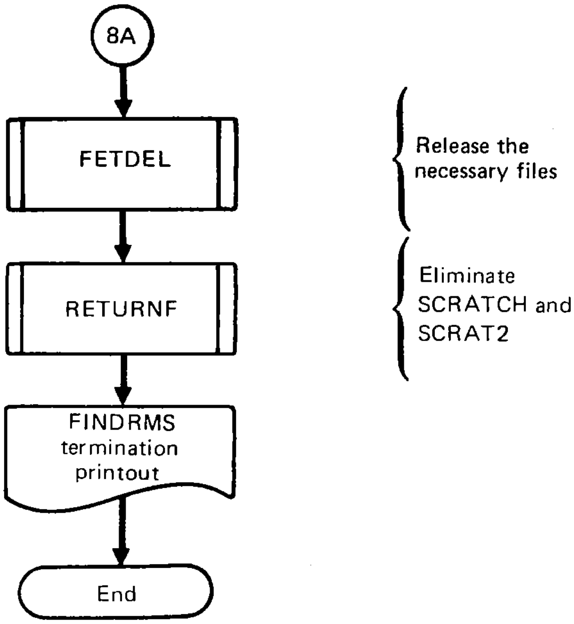


Figure 5.—(Concluded)

Table 2.—Routines Called by FINDRMS

```
OVERLAY(L221,1,0)

PROGRAM FINDRMS

------------------------------------------------------------
                 | DELETR+
                 | FETADD+
                 | FETDEL+
                 | FRCALC
CONDAT          <  INITIR+
                 | LOCF*
                 | NAMFIL+
                 | READTP+
                 | REQFL+
                             | ENDFIL*
                   SINTRP   <  FETADD+
                             | FETDEL+

---------------
* indicates a routine in the FORTRAN subroutine library.
+ indicates a routine in the DYLOFLEX alternate subroutine library DYLIB.
All others are local to L221 (TEV156).
```

Table 2.—(Continued)

OVERLAY (L221, 1, 0)

PROGRAM FINDRMS

- CORREL
- FETADD+
- FETDEL+
- GCMAT
  - DELETR+
  - FETADD+
  - INITIR+
  - IRDCRD+
  - LOCF*
  - MATRED
    - EOF*
    - INITIR+
    - PRINTM+
    - PRNTCM+
    - READTP+
  - REQFL+
  - READTP+
- GRDPEN

Table 2.—(Continued)

```
OVERLAY(L221,1,0)
PROGRAM FINDRMS

-------------------------------------------------------------
                   ┌ DELETR+
                   │ EOF*
                   │ FETADD+
                   │ FETDEL+
                   │ FSF+
                   │ IRDCRD+
LOMAT             <
                   │ INITIR+
                   │ LOCF*
                   │              ┌ EOF*
                   │              │ INITIR+
                   │ MATRED      <
                   │              │ PRINTM+
                   │              │ PRINTCM+
                   │              └ READTP+
                   │ NAMFIL+
                   │ REQFL+
                   └ READTP+
```

Table 2.—(Concluded)

```
OVERLAY(L221,1,0)

PROGRAM FINDRMS

-------------------------------------------------------------

                                     ┌ EOF*
                                     │ INITIR+
MATINT          { MATRED            <  PRINTM+
                                     │ PRNTCM+
                                     └ READTP+
OUTAP           { ENDFIL*
                  WRTETP+
RETURNF+
SECOND*
SAVEQ           { ENDFIL*
                  FSF+
SOLVEL          { FEDBAK
                  SUMATS
                ┌ CGLESM+
SOLVEQ         <  FEDBAK
                └ SUMATS
SPCTRM
STARTR+
STATEL
```

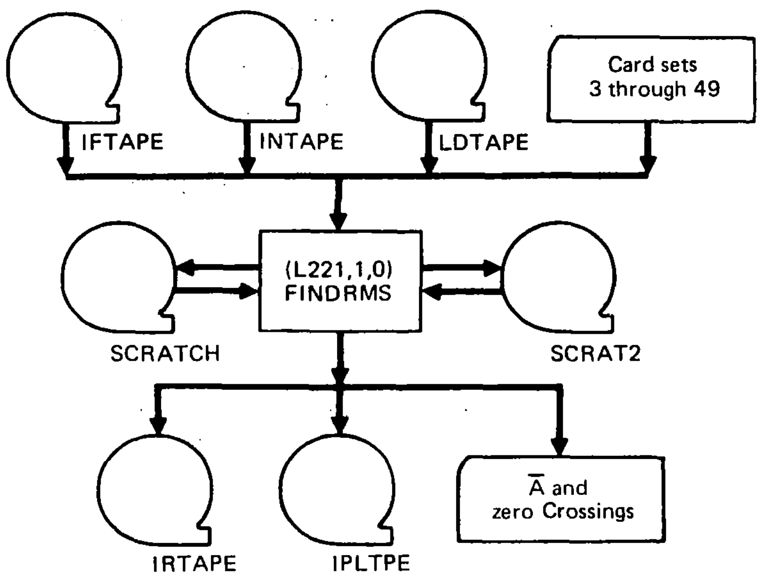


Figure 6.—Overlay (L221,1,0) FINDRMS Input/Output

3.3 OVERLAY (L221,2,0) - SORTQLS

Purpose of SORTQLS

SORTQLS is called to process data written on the magnetic file IPLTPE by FINDRMS, and write a magnetic file, NEWTPE, which contains pairs of matrices, independent and dependent variables, to be plotted by a subsequent program.

IPLTPE contains one file of data for each FINDRMS solution. Each file begins with the frequency array $\{FREQ\}$ (a 1 x NFREQ matrix), which is followed by NFREQ groups of three or four of the following arrays ($\{\overline{A}\}$ will be present only if IPLRMS = 1 on FINDRMS card set 4.3).

o	$\{QMAG_R\}$	1 x NDOF	The magnitude of Q
o	$\{QMAG_L\}$	1 x NLD	The magnitude of $\overline{SUMC}$
o	$\{SPEC\}$	1 x NLD	The output spectrum
o	$\{\overline{A}\}$	1 x NLD	The RMS loads, through frequency i

If the static elastic solution was requested in FINDRMS, every second file on IPLTPE is from a static elastic solution.

Analytical Steps of SORTQLS

The SORTQLS operations are divided into the following logic:

1. Read a set of data cards defining the SORTQLS options, the FINDRMS load set to be processed, and the item ($QMAG_R$, $QMAG_L$, SPEC, or $\overline{A}$) to be sorted.

2. Position IPLTPE at the beginning of the FINDRMS solution to be processed.

3. Read and store the frequency array, $\{FREQ\}$.

4. Read and store the NFREQ matrices of the desired item ($QMAG_R$, $QMAG_L$, SPEC, or $\overline{A}$) as the rows of the NFREQ x NR matrix [SOL1] shown below.

SOL1

$$\text{NFREQ}\left\{\begin{bmatrix} \text{Row 1} \\ \text{Row 2} \\ \cdot \\ \\ \cdot \\ \text{Row NFREQ} \end{bmatrix}\right.$$

$\underbrace{\qquad\qquad\qquad}_{\text{NR}}$

where NR = NDOF for item = $QMAG_R$

and NR=NLD for item $QMAG_L$,

SPEC, or $\overline{A}$.

5. (Optional) – When pairing of standard and static elastic solutions is requested, step 4 will be repeated for the same item of the next solution, which is the static elastic solution in the subsequent file. The new solution is stored in the array [SOL2], which is identical to [SOL1] in size.

6. (Optional) – Scale the frequencies and/or [SOL1] (and [SOL2]) as requested.

7. Write on NEWTPE the frequency array {FREQ} and the column of [SOL1] corresponding to the chosen degree of freedom (for $QMAG_R$) or load (for $QMAG_L$, or SPEC, or $\bar{A}$).

8. (Optional) – Repeat step 7 for the array [SOL2] if pairing was requested.

9. Repeat steps 7 and 8 for all generalized coordinates, loads, spectra, or $\bar{A}$ values requested by the current data set.

10. Repeat steps 1 through 9 for as many data sets as needed, but terminate SORTQLS execution after reading the $END card (see card set 61).

Figure 7 contains a macro flow chart of SORTQLS. The subroutines called by SORTQLS are displayed in table 3.

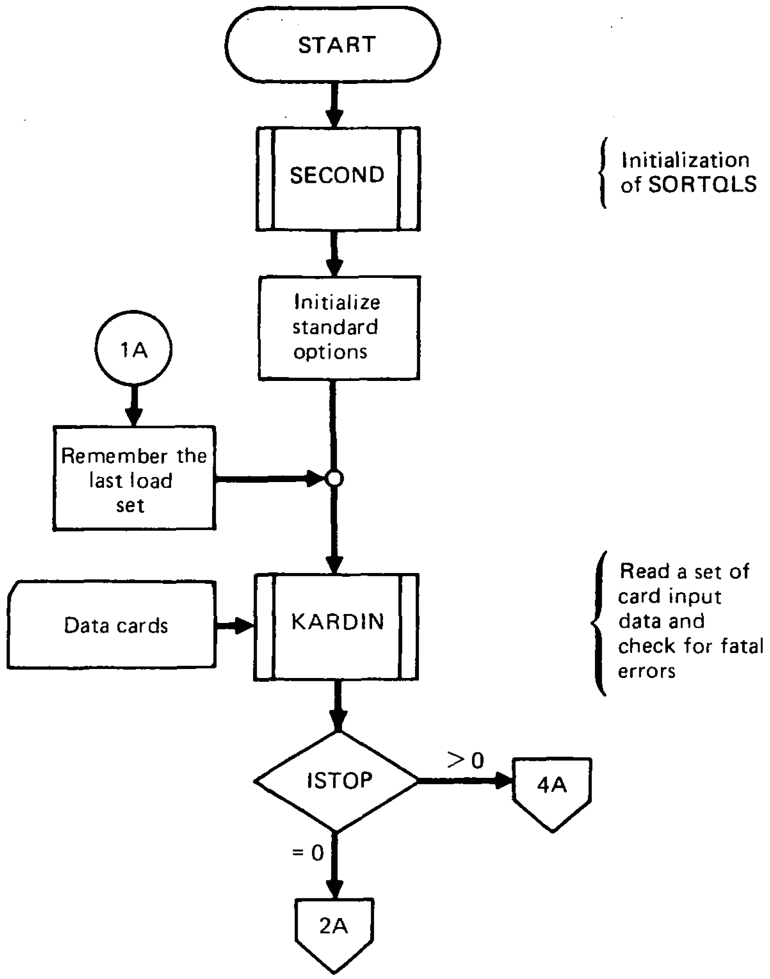


Figure 7.—Macro Flow Chart of Overlay (L221,2,0)—SORTQLS

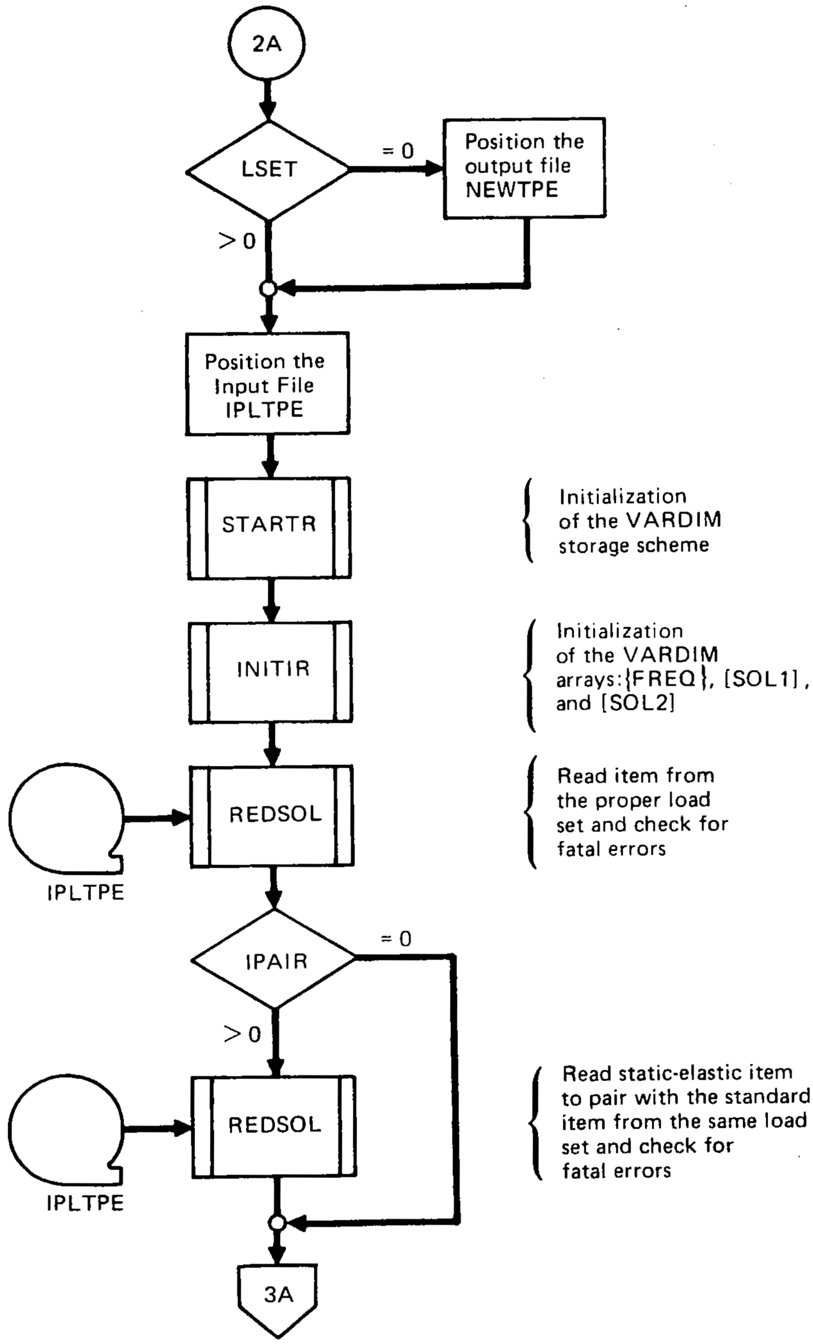


Figure 7.—(Continued)

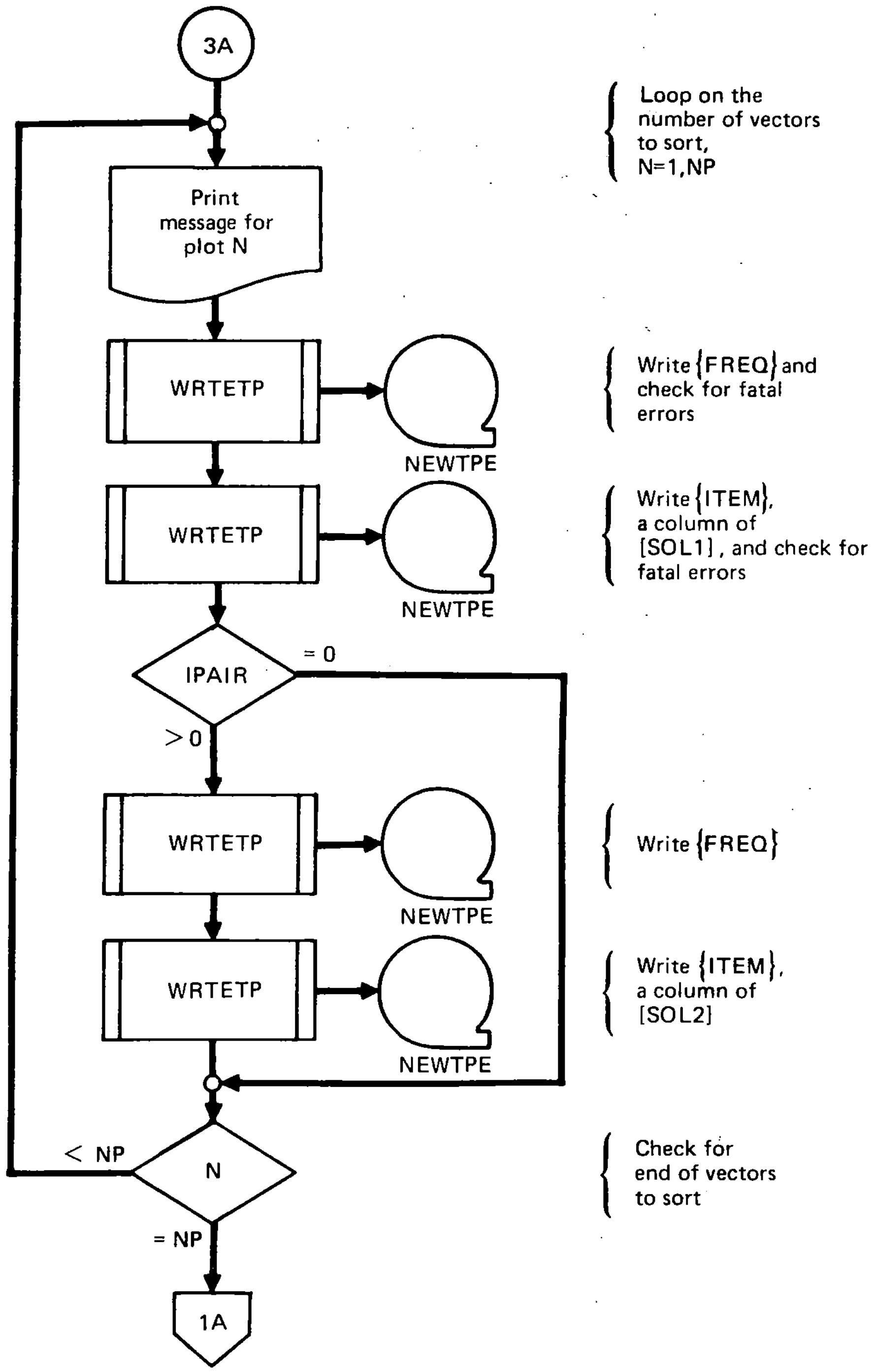


Figure 7.—(Continued)

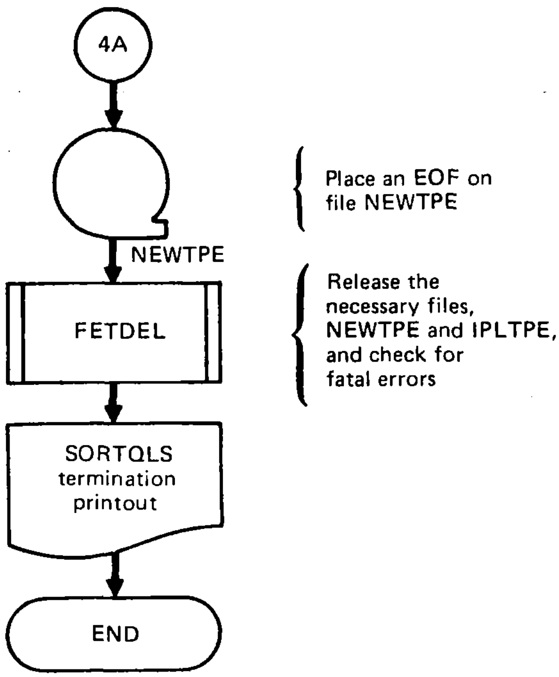


Figure 7.—(Concluded)

Table 3.—Routines Called by SORTQLS

```
OVERLAY(L221,2,0)
PROGRAM SORTQLS
-------------------------------------------------------------
DELETR+
ENDFIL*
FETADD+
FETDEL+
INITIR+
         { CHKWRD
KARDIN   { EOF*
         { IRDCRD+
         { NAMFIL+
LOCF*
REDSOL   { READTP+
REQFL+
SECOND*
STARTR+
WRTETP+
```

----

* indicates a routine in the FORTRAN subroutine library.

\+ indicates a routine in the DYLOFLEX alternate subroutine library.

All others are local to L221 (TEV156).

I/O Devices of SORTQLS

The possible I/O devices for SORTQLS are shown in figure 8. For a complete description of the input data cards and magnetic files, see sections 6.3 and 6.4 in volume I of this document.

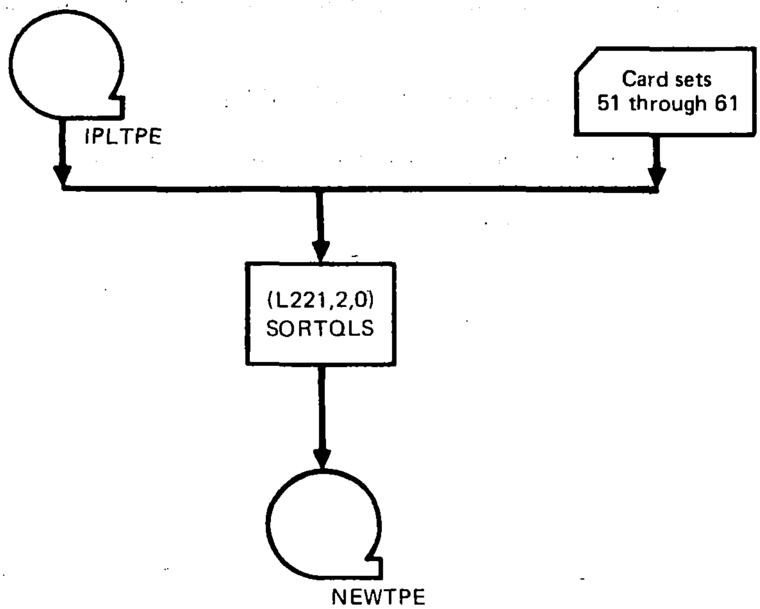


Figure 8.—Overlay (L221,2,0) SORTQLS Input/Output

3.4 OVERLAY (L221,3,0) - PLOTQLS

Purpose of PLOTQLS

The Overlay (L221,3,0), PLOTQLS, is called to process the data written on file IPLTPE by FINDRMS and write an output file TAPE99, which contains the plotting instructions for the COMp80 plotter.

Analytical Steps of PLOTQLS

PLOTQLS sorts the independent and dependent variables in exactly the same manner as program SORTQLS (see section 3.3). Steps 7 and 8 are replaced by calls to NPS subroutines,[1] which generate the plotting instructions.

PLOTQLS has additional data instructions not applicable to SORTQLS; the establishment of grid limits and plot labeling information are two examples.

Figure 9 contains a macro flow chart of PLOTQLS. The subroutines called by PLOTQLS are displayed in table 4.

Note: The automatic plotting overlay PLOTQLS in this program requires subroutines that are proprietary to The Boeing Company.

[1] *Numerical Plotting System - Users Manual*. BCS-G0509, March 1976. (Internal Document.)

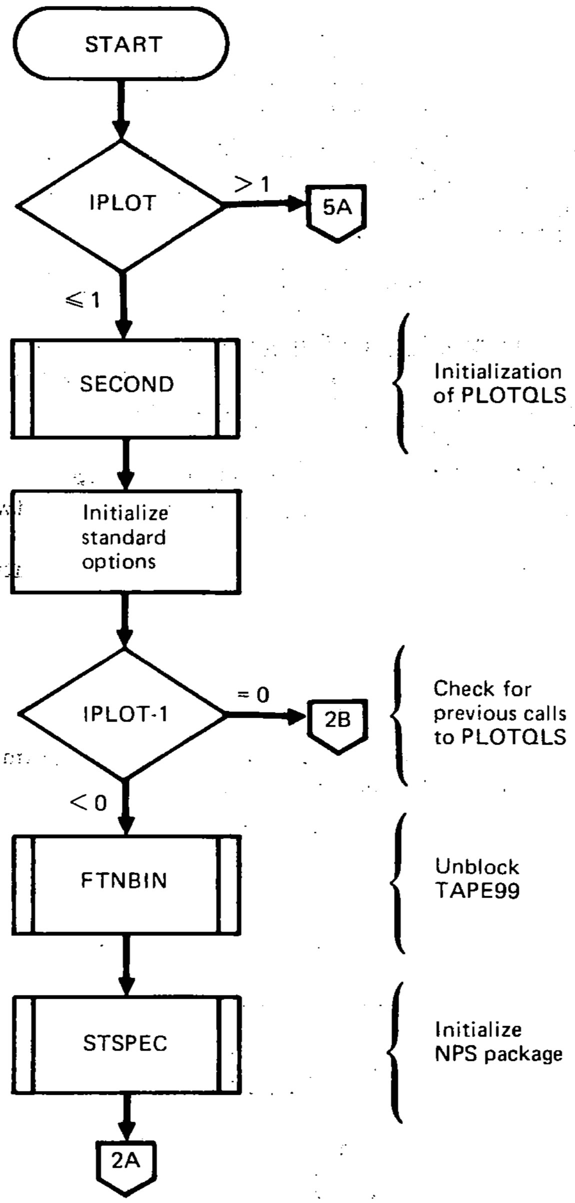


Figure 9.—Macro Flow Chart of Overlay (L221,3,0)—PLOTQLS

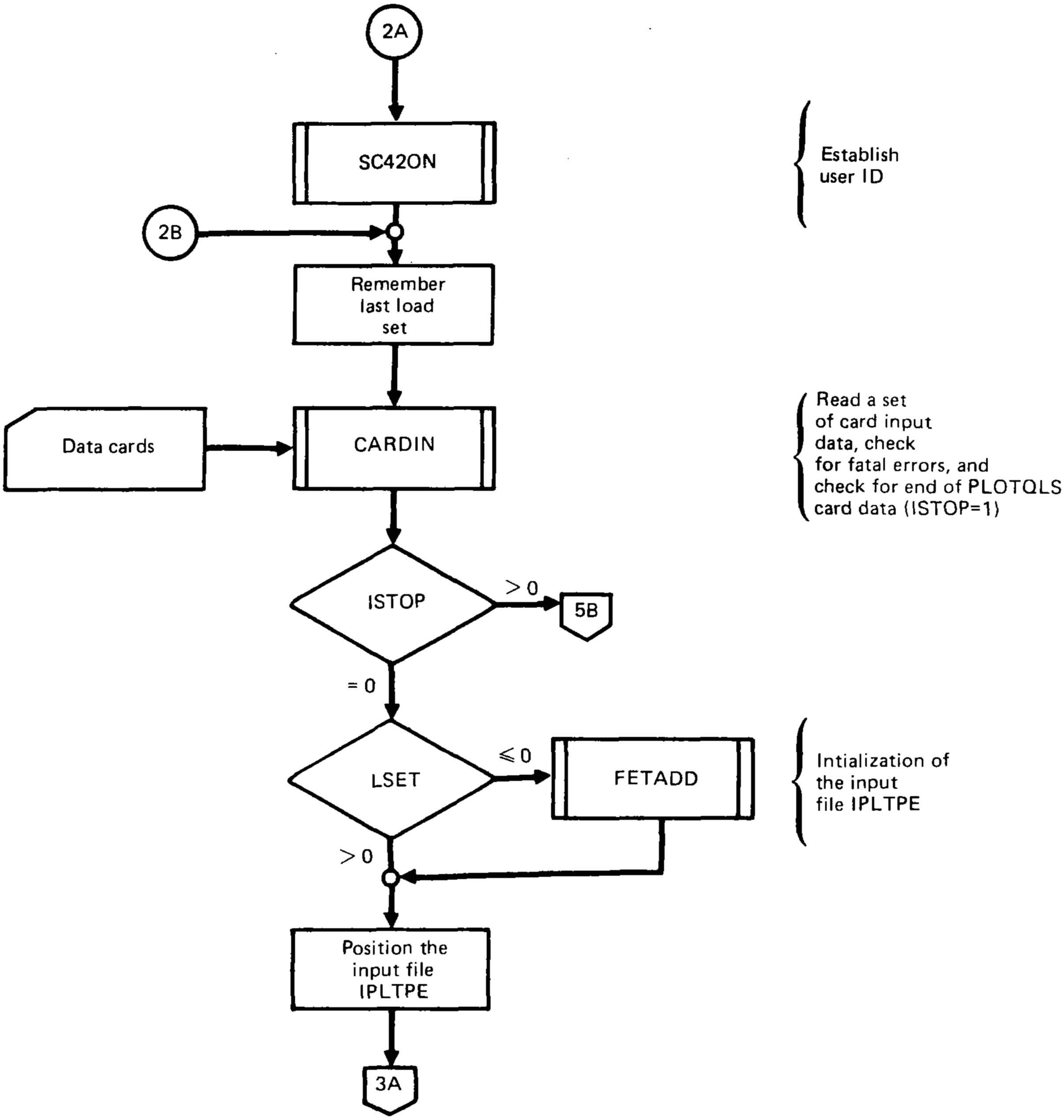


Figure 9.—(Continued)

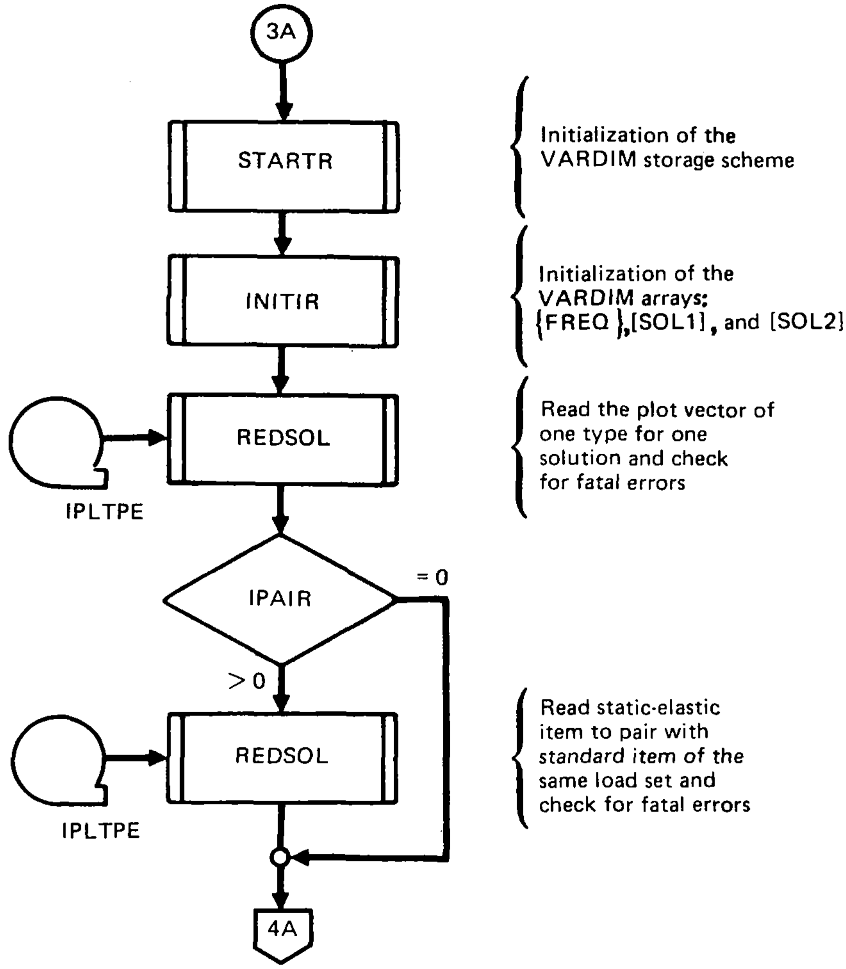


Figure 9.—(Continued)

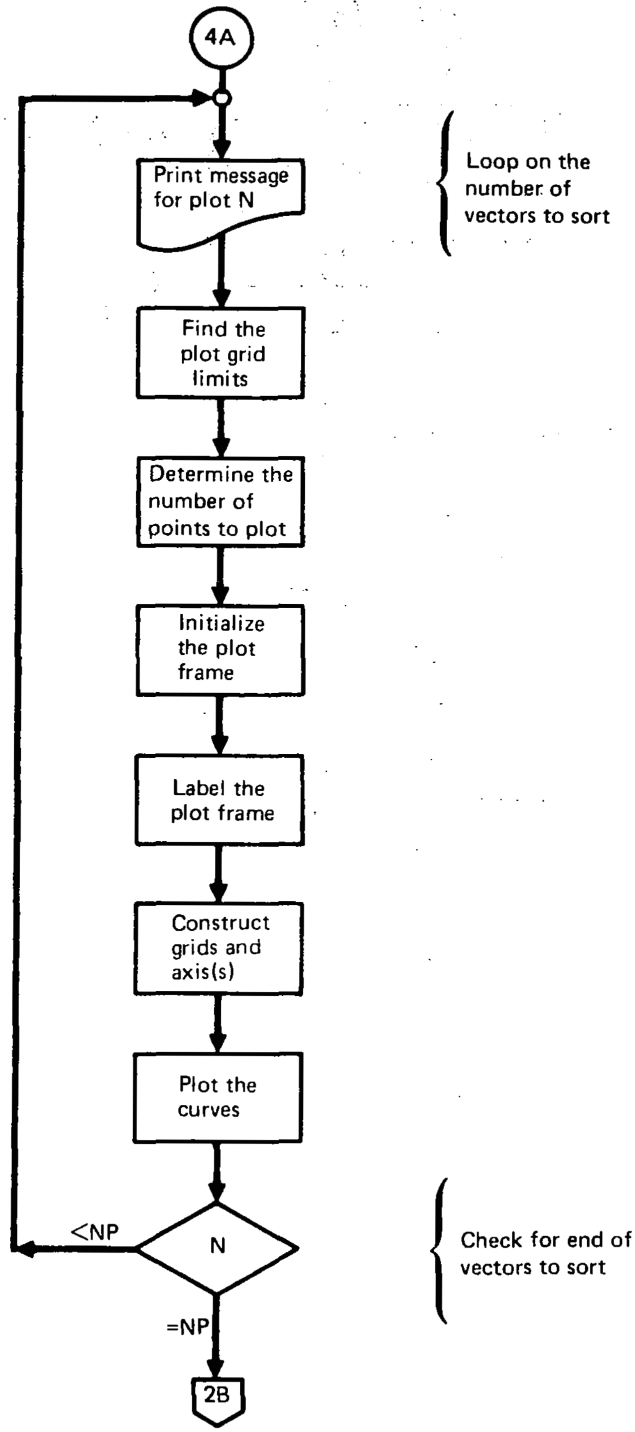


Figure 9.–(Continued)

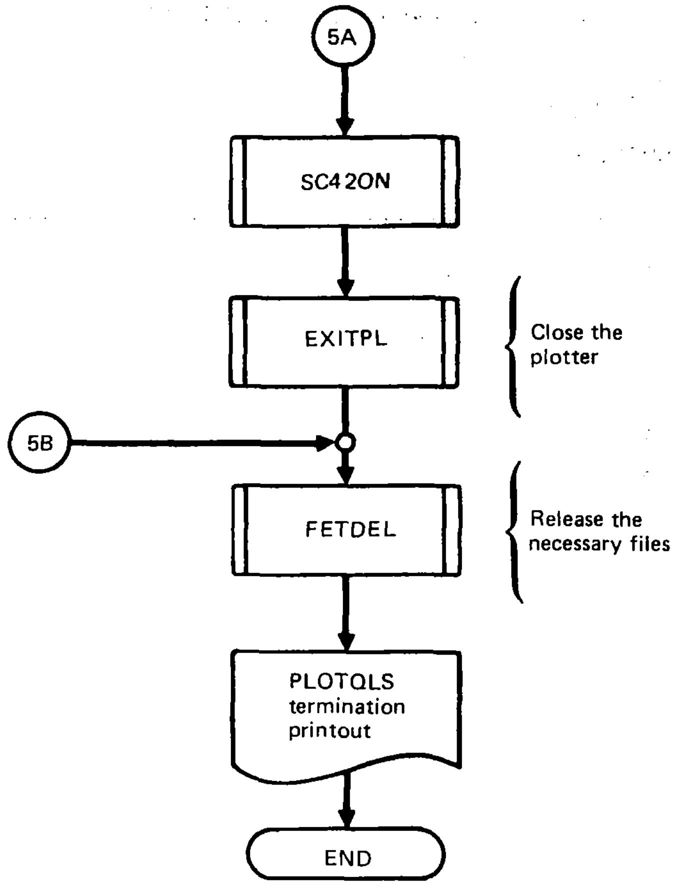


Figure 9.—(Concluded)

Table 4.—Routines Called by PLOTQLS

OVERLAY(L221,3,0)

PROGRAM PLOTQLS

ADJSMX

ADVANC**

AXLGLG**

AXLILI**

CARDIN { CHKWRD, EOF*, IRDCRD+, NAMFIL+ }

EXITPL**

FETADD+

FETDEL+

FORM**

FTNBIN*

---

* indicates a routine in the FORTRAN subroutine library.

** indicates an NPS routine in the BCS FORTRAN subroutine library.

\+ indicates a routine in the DYLOFLEX alternate subroutine library.

All others are local to L221 (TEV156).

Table 4.—(Continued)

```
OVERLAY(L221,3,0)

PROGRAM PLOTQLS

-------------------------------------------------------------

GDLGLG**
GDLILI**
GMNMX
INITIR+
LOCF*
NOLGB**
NOLGL**
NOSLIB**
NOSLILI**
PFLGLG**
PFLILI**
REDSOL   {READTP+
REQFL+
SC42ON**
SECOND*
STARTR+
```

Table 4.—(Concluded)

```
OVERLAY(L221,3,0)
PROGRAM PLOTQLS
------------------------------------------------------------

STCHSZ**
STFONT**
STLNOR**
STLNST**
STNDIV**
STNCHR**
STNPTS**
STSIGF**
STSPEC**
STSUBJ**
STS2OB**
STTOOL**
STTXTR**
TITLEG**
```

I/O Devices of PLOTQLS

The I/O devices used in PLOTQLS are shown in figure 10. For a complete description of the input data cards and files, see sections 6.3 and 6.4 in volume I of this document.

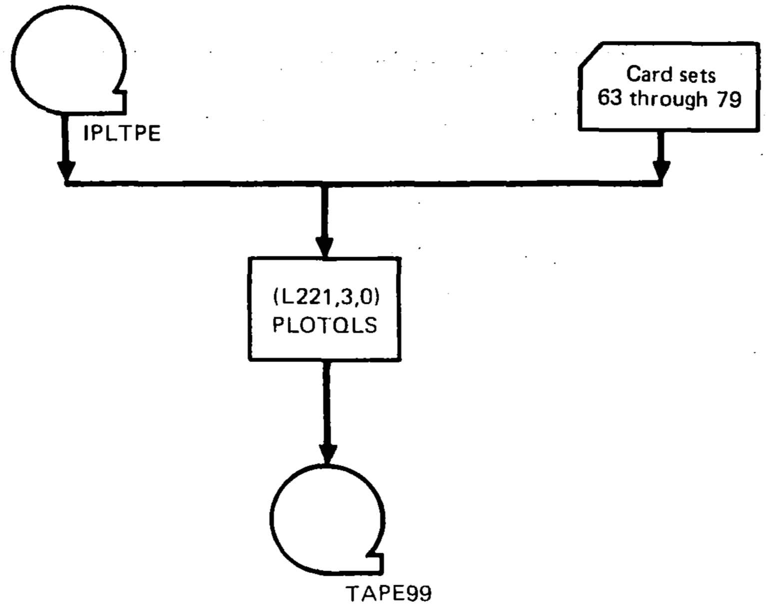


Figure 10.—Overlay (L221,3,0) PLOTQLS Input/Output

3.5 DATA BASES

The L221 (TEV156) data bases include I/O magnetic files (either tape or disk), internal scratch (temporary) storage files, and labeled common blocks.

3.5.1 INPUT DATA

The input data for L221 (TEV156) is in two forms; data cards and magnetic files (either tape or disk).

Card Input Data

For a complete description of the L221 (TEV156) card input data, see section 6.3 in volume I of this document.

Magnetic Files (Tape or Disk)

For a complete description of the L221 (TEV156) disk or tape input data, see section 6.4 in volume I of this document. The possible files read are IFTAPE, INTAPE, and LDTAPE.

3.5.2 OUTPUT DATA

The output results of L221 (TEV156) may be of three types: printed results, magnetic files, and/or punched cards.

Printed Output Data

For a complete description of the printed output data, see section 6.5.1 in volume I of this document.

Magnetic Files (Tape or Disk)

For a complete description of the magnetic file output data, see section 6.5.2 in volume I of this document. The possible output files are IPLTPE, IRTAPE, and NEWTPE.

Punched Card Output Data

For a complete description of the punched card output, see section 6.5.3 in volume I of this document.

3.5.3 INTERNAL DATA

L221 (TEV156) uses two methods to pass data between sections of the program: labelled common blocks and scratch (temporary) magnetic files (either tape or disk).

Magnetic Files (Scratch Disk Files)

L221 (TEV156) uses two disk files, SCRATCH and SCRAT2, for the temporary storage of data. Both files, SCRATCH and SCRAT2, will be returned to the system through the use of subroutine RETURNF[1] when L221 (TEV156) execution is terminated.

---

[1] Clemmons, R. E.: *Programming Specifications for Modules of the Dynamic Loads System to Interface with FLEXSTAB*. NASA Contract NAS1-13918, BCS-G0701, September 1975. (Internal Document.)

SCRATCH

SCRATCH is a temporary scratch file written and read by the 1,0 primary overlay, FINDRMS. The file contains the power spectral density (PSD) solution frequencies and responses required to calculate loads. The data for each frequency is written as a single record with a standard FORTRAN binary WRITE statement. The contents of SCRATCH are displayed in figure 11.

Array size	Contents	
NDOF x 1*	$\{Q\}_i$	Repeat for $i = 1$, NFREQ
NDOF x 1	$\{QMAGR\}_i$	
1*	GFS_i	
1*	FFS_i	
1*	$FYDAMP_i$	
	End-of-File	

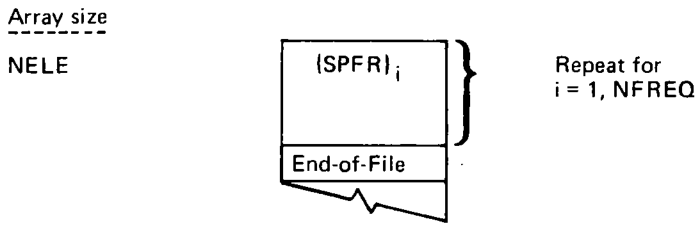
where:

FFS	=	The computed Küssner lift growth function for frequency i
FYDAMP	=	The computed feedback transfer function for frequency i
GFS	=	The computed Wagner lift growth function for frequency i
NDOF	=	The number of degrees of freedom
NFREQ	=	The number of frequencies
Q	=	The array of generalized coordinates for frequency i
QMAGR	=	The array containing the magnitudes of {Q}'s elements for frequency i
*		Signifies that arrays are complex and twice the size indicated

Figure 11.—Contents of SCRATCH—A Temporary Scratch File

SCRAT2

SCRAT2 is a temporary scratch file written and read by the 1,0 primary overlay, FINDRMS. The file contains the interpolated tabular input spectrums to be used in the calculation of $\{\overline{A}\}$ and $\{N_0\}$. The arrays are written onto SCRAT2 with standard FORTRAN binary WRITE statements. The contents of SCRAT2 are displayed in figure 12.



where:

NELE	=	The number of unique spectrum elements per frequency, (NFORC* (NFORC+1))/2
NFREQ	=	The number of frequencies
NFORC	=	The number of excitation forces

Figure 12.—Contents of SCRAT2—A Temporary Scratch File

Common Blocks

Table 5 displays the common blocks used in the program and the programs in which they are used.

The labeled common blocks are used for communication between the main and primary overlays, and for communication between routines in a primary overlay. The block names and contents are described on the following pages.

Blank common is used in the primary overlays of L221 (TEV156) as a variable length working storage area. Each overlay calculates the core required to process the problem being analyzed, and calls subroutines of VARDIM (variable dimensioning storage) subroutines to perform the bookkeeping of array allocation. Pointers to the array locations are stored in labeled common blocks.

Table 5.—Common Blocks Used in Each Overlay

Common Blocks

	ALOVER	BUFFR1	BUFFR3	BUFFR8	BUFF11	BUFR12	CASLOC	CHKPRT	CNSTNT	COMLOC	CORSIZ	GNCLOC	INOUT	KDEPEN	LODLOC	NPSCMN	NPSCOM	NPSZQ1	NUMERR	PLTBND	PLTDTA	RWBUFF	SIZOPT	SRTDTA	TAPBUF	TAPNUM	VARDIM	blank
OVERLAY(L221,0,0)																												
PROGRAM L221vc	*							*			*		*			*	*	*	*	*								
OVERLAY(L221,1,0)																												
PROGRAM FINDRMS	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*				*			*	*			*	*	*
OVERLAY(L221,2,0)																												
PROGRAM SORTQLS	*							*		*	*		*						*			*		*	*		*	*
OVERLAY(L221,3,0)																												
PROGRAM PLOTQLS	*							*		*	*		*			*	*	*	*	*	*	*			*		*	*

The tables on the following pages display all of the common blocks within L221 (TEV156). Table 5 indicates which blocks are used by each of the overlays. Tables 6 through 9 show (per overlay) which routines require the common blocks. Finally, the contents of each labeled common block are described in table 10.

Description of the Labeled Common Blocks

Table 10 describes each variable contained in all of the labeled common blocks of L221 (TEV156). The common blocks are ordered alphabetically, with each variable being described according to its location within the common blocks.

Table 6.—Common Blocks Used in the Routines of L221vc

	Common Blocks								
	ALOVER	CHKPRT	CORSIZ	INOUT	NPSCMN	NPSCOM	NPSZQ1	NUMERR	PLTBND
OVERLAY(L221,0,0) PROGRAM L221vc	*	*	*	*	*	*	*	*	*
CHKWRD									
IRDCRD				*					
PRGBEG				•					
PRGEND				*					

Table 7.—Common Blocks Used in the Routines of FINDRMS

Common Blocks

	Global[a]				Local to (L221,1,0) - FINDRMS																
	ALOVER	CORSIZ	INOUT	NUMERR	BUFFR1	BUFFR3	BUFFR8	BUFF11	BUFR12	CASLOC	CHKPRT	CNSTNT	COMLOC	GNCLOC	KDEPEN	LODLOC	RWBUFF	SIZOPT	TAPNUM	VARDIM	blank
OVERLAY(L221,1,0)																					
PROGRAM FINDRMS	*	*	*	*	*	*	*	*	*	*	*	*		*	*	*	*	*			*
CONDAT		*	*				*			*	*	*						*		*	*
CORREL			*																		
DELETR			*								*		*							*	*
FEDBAK																					
FRCALC			*	*																	
GCMAT		*	*				*			*	*	*		*	*			*	*	*	*
GRDPEN																					
INITIR			*	*							*									*	*
LOCATR			*								*		*							*	*
LOMAT		*	*				*			*	*	*			*	*		*	*	*	*
MATINT																			*		
MATRED			*	*															*		*
OUTAP			*	*																	
READTP																	*				
SAVEQ			*	*																	
SINTRP			*						*												
SOLVEL			*																		
SOLVEQ			*	*																	
SPCTRM			*	*																	
STARTR			*	*							*									*	*
STATEL																					
SUMATS																					
WRTETP																	*				
XFER																					

[a]Global means the block is used in more than one overlay

Table 8.—Common Blocks Used in the Routines of SORTQLS

Common Blocks

	Global[a]				Local to (L221,2,0) - SORTQLS						
	ALOVER	CORSIZ	INOUT	NUMERR	CHKPRT	COMLOC	RWBUFF	SRTDTA	TAPBUF	VARDIM	blank
OVERLAY(L221,2,0)											
PROGRAM SORTQLS	*	*	*	*			*	*	*	*	*
CHKWRD											
DELETR			*		*	*				*	*
INITIR			*	*	*					*	*
KARDIN	*		*					•			
LOCATR			*		*	*				*	*
READTP							*				
REDSOL			*								
STARTR			*	*	*					*	*
WRTETP							•				
XFER											

[a]Global means the block is used in more than one overlay

Table 9.—Common Blocks Used in the Routines of PLOTQLS

Common Blocks

	Global[a]							Local to (L221,3,0) - PLOTQLS							
	ALOVER	CORSIZ	INOUT	NPSCMN	NPSCOM	NPSZG1	NUMERR	CHKPRT	COMLOC	PLTBND	PLTDTA	RWBUFF	TAPBUF	VARDIM	blank
OVERLAY(L221,3,0)															
PROGRAM PLOTQLS	*	*	*				*			*	*	*	*	*	*
ADJSMX															
CARDIN	*		*								*				
CHKWRD															
DELETR			*					*	*					*	*
GMNMX															
INITIR			*				*	*						*	*
LOCATR								*	*					*	*
READTP			*									*			
REDSOL			*												
STARTR			*				*	*						*	*
XFER															

[a]Global means the block is used in more than one overlay.

Table 10.—Description of the Labeled Common Blocks

LABELLED COMMON NAME: ALOVER

DESCRIPTION: Variables used between the primary overlays

NO.	VARIABLE	TYPE[a]	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	MDOF	I			Contains the number of degrees of freedom for SORTQLS and PLOTQLS.
2	MFREQ	I			Contains the number of frequencies for use in SORTQLS and PLOTQLS.
3	MSTATE	I			Indicates which static-elastic solution type will be used.
4	MPLTPE	I			Contains the name of the plot file for use in SORTQLS and PLOTQLS.
5	MRTAPE	I			Contains the magnetic file name whose contents are the load transfer functions to be used in SORTQLS and PLOTQLS.
6	NSETS	I			Number of sets of load equations.
7	ISET	I	10		Array containing the number of loads in each set.
8	NGROUP	I			Contains the number of items to be contained on the magnetic file defined by MRTAPE for each frequency to be used by SORTQLS and PLOTQLS.

[a]Variable types are as follows: C = complex, H = hollerith, I = integer, L = logical, O = octal, R = real

Table 10.—(Continued)

LABELLED COMMON NAME: BUFFR1

DESCRIPTION: Buffer Area for the Generalized Coordinate Scratch File.

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA1	I			Length of ARAY1
2	ARAY1	R	1000		Buffer area used for the generalized coordinates (scratch).

Table 10.—(Continued)

LABELLED COMMON NAME: BUFFR3

DESCRIPTION: Buffer Area for the Generalized Coordinates and Loads.

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA3	I			Length of ARAY3
2	ARAY3	R	1000		Buffer area used for the Generalized Coordinates and Loads.

Table 10.—(Continued)

LABELLED COMMON NAME: BUFFR8

DESCRIPTION: Buffer Area used for the Input Matrices

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA8	I			Length of ARAY8
2	ARAY8	R	540		Buffer area used for the input matrices.

Table 10.—(Continued)

LABELLED COMMON NAME: BUFF11

DESCRIPTION: Buffer Area used for Plot Data

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA11	I			Length of ARA11.
2	ARA11	R	1000		Buffer area used for plot data.

Table 10.–(Continued)

LABELLED COMMON NAME: BUFR12

DESCRIPTION: Scratch Area used for Tabular Input Spectrum

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA12	I			Length of ARA12.
2	ARA12	R	540		Scratch area used for the Tabular Input Spectrum.
3	ISCRT2	I			Name of the scratch file, (SCRAT2).

Table 10.—(Continued)

LABELLED COMMON NAME: **CASLOC**

DESCRIPTION: Contains the Locations of the Case-dependent arrays in blank common (VARDIM storage).

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LCSO	I			Pointer to {CSO}, special feedback coefficient matrix.
2	LCS1	I			Pointer to {CS1}, special feedback coefficient matrix.
3	LCS2	I			Pointer to {CS2}, special feedback coefficient matrix.
4	LDAMP	I			Pointer to {DAMP}, the damping coefficient matrix.
5	LFLL	I			Pointer to {FLL}, the distance between gradual penetration panels. Used to calculate the time log.
6	LFREQ	I			Pointer to {FREQ}, the frequency solution matrix.
7	LINSP	I			Pointer to {INSP}, the tabular input spectrum matrix.
8	LNDEL	I			Pointer to {NDEL}, the degrees of freedom deletion matrix.

Table 10.—(Continued)

LABELLED COMMON NAME: CASLOC (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
9	LNSTEL	I			Pointer to {NSTEL}, the columns to be zeroed for a static-elastic solution.
10	LQ	I			Pointer to {Q}, the frequency response matrix.
11	LQMAGR	I			Pointer to {QMAGR}, the magnitude of {Q}

Table 10.—(Continued)

LABELLED COMMON NAME: CHKPRT

DESCRIPTION: Problem Printout Option

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	ICKPRT	I			Problem Printout Option =0 The input matrices are not printed. =1 The input matrices will be printed as read. =2 The input matrices plus the intermediate data will be printed for the first 3 frequencies. =5 Special program checkout feature. It prints 1 and 2 above plus a record of all the VARDIM routines called.

Table 10.—(Continued)

LABELLED COMMON NAME: CNSTNT

DESCRIPTION: Input Constants for FINDRMS

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	A1	R		a_1	Wagner indicial lift growth function coefficients.
2	B1	R		b_1	
3	C1	R		c_1	
4	D1	R		d_1	
5	ALPHA1	R		α_1	
6	BETA1	R		β_1	
7	GAMMA1	R		γ_1	
8	A2	R		a_2	Küssner indicial lift growth function coefficients.
9	B2	R		b_2	
10	C2F	R		c_2	
11	D2	R		d_2	
12	ALPHA2	R		α_2	
13	BETA2	R		β_2	
14	GAMMA2	R		γ_2	
15	Z	R		Z	The scalar constant for the equation of motions and loads equation forcing function coefficient matrices.
16	VEL	R		V_T	The velocity (true air speed).
17	OCOEF	R		κ	Exponential Input Spectrum power constant.

Table 10.—(Continued)

LABELLED COMMON NAME: CNSTNT (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
18	OEX	R		η	Exponential Input Spectrum power constant.
19	TSCALE	R		L	The turbulence scalar constant.
20	GPSCAL	R		$\bar{c}$	The gradual penetration scalar constant.
21	TABFS	R			The tabular input spectrum frequency scalar.
22	TABS	R			The tabular input spectrum power scalar constant.
23	FDSCAL	R			Contains the scalar constant used to scale the frequencies in the Frequency-Dependent Frequency Matrix {FREQM}.
24	GPSCAL	R		c	Contains the gradual penetration scalar constant of $\{C_3\}$.
25	G	R		G	The feedback gain coefficient of the feedback transfer function, FYDAMP.
26	RLIMIT	R		$\delta_{r_{lim}}$	Special feedback rudder limit.

Table 10.—(Continued)

LABELLED COMMON NAME: CNSTNT (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
27	P	R	7	{P}	Coefficients of the feedback transfer function, FYDAMP.
28	U	R	7	{U}	
29	R	R	7	{R}	
30	T	R	7	{T}	

Table 10.—(Continued)

LABELLED COMMON NAME: COMLOC

DESCRIPTION: Specifies the last array entry found by LØCATR.

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	JTHENT	I			Specifies the catalog (KATLOG) entry of the last array located by routine LOCATR, (LOCATR is a VARDIM routine).

Table 10.—(Continued)

LABELLED COMMON NAME: CORSIZ

DESCRIPTION: Core size

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	MAXCOR	O			Contains the maximum allowable core size. The default of 200000_8 may be changed via card input.

Table 10.—(Continued)

LABELLED COMMON NAME: GNCLOC

DESCRIPTION: Locations of the Generalized Coordinate Arrays

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LPHIT	I			Pointer to [$\tilde{\phi}$], gradual penetration summation matrix used to calculate {C_3}.
2	LC2	I			Pointer to {C_2}.
3	LC3	I			Pointer to {C_3}.
4	LSUMM	I			Pointer to [SUMM], the equations of motion summation matrix.
5	LSUMC	I			Pointer to {SUMC}, the forcing function matrix.
6	LM1	I			Pointer to [M_1], an equations of motion matrix.
7	LM2	I			Pointer to [M_2], an equations of motion matrix.
8	LM3	I			Pointer to [M_3], an equations of motion matrix.
9	LM4	I			Pointer to [M_4], an equations of motion matrix.
10	LM5	I			Pointer to [M_5], an equations of motion matrix.

Table 10.—(Continued)

LABELLED COMMON NAME: **GNCLOC** (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
11	LM6	I			Pointer to [M_6], an equations of motion matrix.
12	LS1	I			Pointer to [S_1], a feedback matrix.
13	LS2	I			Pointer to [S_2], a feedback matrix.
14	LS3	I			Pointer to [S_3], a feedback matrix.
15	LS4	I			Pointer to [S_4], a feedback matrix.
16	LS5	I			Pointer to [S_5], a feedback matrix.
17	LS6	I			Pointer to [S_6], a feedback matrix.

Table 10.—(Continued)

LABELLED COMMON NAME: **INOUT**

DESCRIPTION: Input/Output File Names

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	INFIL	I			Name of the file used for card input data, (TAPE5).
2	IUTFIL	I			Name of the file used for printed output, (TAPE6).
3	IPFIL	I			Name of the file used for punched output, (TAPE7).

Table 10.—(Continued)

LABELLED COMMON NAME: KDEPEN

DESCRIPTION: Locations of the arrays needed for multiple Frequency-Dependent Matrix Problems.

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	OMEG1	R		Ω_1	Frequency at which input matrices are defined.
2	LM41	I			Pointer to [M41], $[M_4]$ at OMEG1.
3	LM51	I			Pointer to [M51], $[M_5]$ at OMEG1.
4	LM61	I			Pointer to [M61], $[M_6]$ at OMEG1.
5	LC31	I			Pointer to {C31}, $\{C_3\}$ at OMEG1.
6	LPHIT1	I			Pointer to [PHIT1], $[\tilde{\phi}]$ or $[\tilde{\tilde{\phi}}]$ at OMEG1.
7	LFREQM	I			Pointer to {FREQM}, the frequencies (in CPS) at which $[M_4]$, $[M_5]$, $[M_6]$, and $\{C_3\}$ or $[\tilde{\phi}]$ defined.
8	OMEG2	R		Ω_2	Frequency at which input matrices are defined.
9	LM42	I			Pointer to [M42], $[M_4]$ at OMEG2.
10	LM52	I			Pointer to [M52], $[M_5]$ at OMEG2.

Table 10.—(Continued)

LABELLED COMMON NAME: KDEPEN (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
11	LM62	I			Pointer to [M62], $[M_6]$ at OMEG2.
12	LC32	I			Pointer to {C32}, $\{C_3\}$ at OMEG2.
13	LPHIT2	I			Pointer to [PHIT2], $[\tilde{\phi}]$ or $[\tilde{\tilde{\phi}}]$ at OMEG2.

Table 10.–(Continued)

LABELLED COMMON NAME: LODLOC

DESCRIPTION: Locations of the Load Set-Dependent Arrays

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LLOAD	I			Pointer to {LOAD}, a correlation array.
2	LLOADC	I			Pointer to [LOADC], a correlation array.
3	LNLDC	I			Pointer to {NLDC}, number of loads to be correlated with each load.
4	LCORSU	I			Pointer to [CORSUM], a correlation array.
5	LCORLS	I			Pointer to [CORLST], a correlation array.
6	LQMAGL	I			Pointer to {QMAGR}, the magnitude of {Q}.
7	LSPEC	I			Pointer to {SPEC}, the output spectrum.
8	LA	I			Pointer to $\{\bar{A}\}$, the RMS values.
9	LPM	I			Pointer to $\{N_o\}$, the zero crossings.
10	LPHITB	I			Pointer to $[\bar{\bar{\phi}}]$, the summation matrix for gradual penetration calculation of $\{\bar{C}_3\}$.
11	LC2B	I			Pointer to $\{\bar{C}_2\}$.
12	LC3B	I			Pointer to $\{\bar{C}_3\}$.

Table 10.–(Continued)

LABELLED COMMON NAME: **LODLOC** (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
13	LSUMMB	I			Pointer to [$\overline{SUMM}$], the load summation matrix.
14	LSUMCB	I			Pointer to {$\overline{SUMC}$}, the load transfer function.
15	LPHIOM	I			Pointer to {PHIØME}, the spectrums at Ω.
16	LMB1	I			Pointer to [$\overline{M}_1$], the load matrix.
17	LMB2	I			Pointer to [$\overline{M}_2$], the load equation matrix.
18	LMB3	I			Pointer to [$\overline{M}_3$], the load equation matrix.
19	LMB4	I			Pointer to [$\overline{M}_4$], the load equation matrix.
20	LMB5	I			Pointer to [$\overline{M}_5$], the load equation matrix.
21	LMB6	I			Pointer to [$\overline{M}_6$], the load equation matrix.
22	LSB1	I			Pointer to [$\overline{S}_1$], a load feedback matrix.
23	LSB2	I			Pointer to [$\overline{S}_2$], a load feedback matrix.
24	LSB3	I			Pointer to [$\overline{S}_3$], a load feedback matrix.

Table 10.–(Continued)

LABELLED COMMON NAME: **LODLOC** (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
25	LSB4	I			Pointer to [$\bar{S}_4$], a load feedback matrix.
26	LSB5	I			Pointer to [$\bar{S}_5$], a load feedback matrix.
27	LSB6	I			Pointer to [$\bar{S}_6$], a load feedback matrix.

Table 10.—(Continued)

LABELLED COMMON NAME: NPSCMN

DESCRIPTION: NPS routines buffer area

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	DUM2	R	132		Scratch storage area used by the NPS plotting subroutines.*

* Boeing Computer Services, Inc., "Numerical Plotting System - Users Manual"; BCS Document BCS-G0509; March, 1976. (Internal document)

Table 10.—(Continued)

LABELLED COMMON NAME: **NPSCOM**

DESCRIPTION: NPS routines

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	DUM1	R	1157		Scratch storage area used by the NPS plotting subroutines.*

* Boeing Computer Services, Inc., "Numerical Plotting System - Users Manual"; BCS Document BCS-G0509; March, 1976. (Internal document)

Table 10.—(Continued)

LABELLED COMMON NAME: NPSZQ1

DESCRIPTION: NPS routines

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	DUM3	R	3		Scratch storage area used by the NPS plot subroutines.*

* Boeing Computer Services, Inc.; "Numerical Plotting System - Users Manual"; BCS Document BCS-G0509; March, 1976. (Internal document)

Table 10.–(Continued)

LABELLED COMMON NAME: NUMERR

DESCRIPTION: Number of Diagnosed Errors

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	NERROR	I			Contains the number of fatal errors diagnosed during the execution of L221 (TEV156).

Table 10.—(Continued)

LABELLED COMMON NAME: PLTBND

DESCRIPTION: Plot Identification

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	IPLOT	I			Indicates the status of the plots. =0 plot has not been previously called. =1 plotter has been previously called. =2 close the plotter.
2	IDPLOT	H	5		Array containing the user's plotting information (name,phone no., mail stop, and organization).
3	SPECS	R	75		Array containing the internal NPS plotting specifications.
4	BUFFER	R	126		Buffer area for NPS plotting routines.

Table 10.—(Continued)

LABELLED COMMON NAME: PLTDTA

DESCRIPTION: Plot data

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	JSET	I			Contains the load set from which to plot.
2	ITYPE	I			Option indicating the variable plotted. 1=Q's 2=Loads 3=Spectrums 4=Root Mean Square of Loads.
3	IPAIR	I			Denotes whether the standard and the static-elastic solutions will be paired for plotting. =0 separate =1 pair
4	JSTATE	I			Index on the solution to be plotted. =0 standard =1 static-elastic
5	NP	I			Number of responses to plot.
6	IRRAY	R	100		Array containing the numbers of vectors to be plotted.
7	SCALX	R			Scalar constant for the frequency array.
8	SCALY	R			Scalar constant for the output responses.

Table 10.—(Continued)

LABELLED COMMON NAME: PLTDTA (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
9	ITTLE	H	12		Misc. title blocks for labeling the plot frames.
10	IGRID	I			Grid limit input =0 set to data limits =1 input on cards.
11	GRIDL	I	4		Contains the X and Y grid limits GRIDL(1)=X min.lmts. GRIDL(2)=X max.lmts. GRIDL(3)=Y min.lmts. GRIDL(4)=Y max.lmts.
12	LINLOG	I			Axis scale type =0 Linear-linear =1 log-log
13	LABELX	H	2		Array containing the X-axis plotting label.
14	LABELY	H	4		Array containing the Y-axis plotting label.

Table 10.–(Continued)

LABELLED COMMON NAME: RWBUFF

DESCRIPTION: Buffer area for Routines READTP/WRTETP

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	ITELL	H			Contains 8HBUFFSIZE
2	ISIZE	I			Length of DECRS.
3	DECRS		10000		Buffer area used by the routines READTP/ WRTETP.

Table 10.–(Continued)

LABELLED COMMON NAME: SIZOPT

DESCRIPTION: Contains the problems , dimensions, options, and tape numbers to be used in FINDRMS.

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	FREQL	R	10		Frequencies at which LOAD/DOF will be printed.
2	ICORR	I			Number of loads to be correlated with others.
3	IDAMP	I			Option which indicates the number of different structural damping factors. =0 the same factor for all freedoms. =1 a different damping factor for each frequency.
4	IFDBAK	I			Option which chooses the feedback loop to be used. =0 no feedback loop used. =1 standard feedback loop used. =2 special feedback loop used.
5	IFREQ	I			Option which chooses the frequency matrix input. =-1 Frequencies will be read from cards in cycles/second.

Table 10.—(Continued)

LABELLED COMMON NAME: SIZOPT (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
					=0 Frequencies will be read from cards in radians per sec. =1,2,3,4,5 The frequency matrix will be generated according to Appendix A.
6	IFTAPE	H			Denotes the location of the Tabular Input Spectrums. They will be read from cards if blank or will be read from tape "IFTAPE" if defined.
7	INTAPE	H			Denotes the location of Input Matrices. They will be read from cards if blank or will be read from tape "INTAPE" if defined.
8	INTZRO	I			Option which chooses the starting point for the RMS integration. =0 start at zero. =1 start at $FREQ_1$.

Table 10.—(Continued)

LABELLED COMMON NAME: **SIZOPT** (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
9	IPLTPE	H			Indicates whether or not a plotting file will be created. Plotting information will not be written onto a file if IPLTPE is blank, otherwise it will be written onto tape "IPLTPE" if defined.
10	IPLRMS	I			Option requesting that the $\bar{A}$ values be written on the plot tape. =0 $\bar{A}$ values not saved. =1 $\bar{A}$ values saved on "IPLTPE" for each frequency.
11	IPRINT	I			Option which chooses the type of printing to take place. =0 print only the final $\bar{A}$ values and the number of zero crossings. =1 print the generalized coordinates and load transfer functions for each frequency as well as "0" above.

Table 10.—(Continued)

LABELLED COMMON NAME: **SIZOPT** (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
12	IPUNCH	I			Option which chooses whether or not the final $\bar{A}$ values and the number of zero crossings will be punched on cards. =0 no punched output =1 punched output.
13	IREDUC	I			Not used.
14	IRMSPR	I			Option which chooses the type of printing for the $\bar{A}$ values. =0 print only the final $\bar{A}$ values. >0 print the $\bar{A}$ value for each of the last IRMSPR frequencies.
15	IRTAPE	H			Denotes whether or not the Q's and the load transfer functions will be written. The information will not be saved if left blank, otherwise the information will be written onto tape "IRTAPE".
16	ISPEC	I			Option which chooses the type of input spectrum to be used.

Table 10.—(Continued)

LABELLED COMMON NAME: SIZOPT (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
					=0 Dryden =1 Von Karman =2 Expoential =-NSP Tabular input spectrum defined at the NSP point.
17	ISTATE	I			Option which chooses the static-elastic solution. =0 no static-elastic solution. =1 static-elastic solution found with rigid body degrees of freedom in col.1 & 2. =2 static-elastic solution found after zeroing specified columns.
18	LABEL	H	7		Contains miscellaneous labeling information for labeling the data case.
19	LDTAPE	H			Denotes the location of the load equations. If the load equations and/or frequency-dependent load equations are contained on tape "INTAPE" or if they

Table 10.—(Continued)

LABELLED COMMON NAME: SIZOPT (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
					are to be read from cards the variable LDTAPE is to be left blank.
20	LPDOF	I			Option which indicates whether or not the loads per degrees of freedom will be found. =0 will not be found. >0 loads due to each degree of freedom will be found and printed.
21	NDOF	I			Contains the number of degrees of freedom.
22	NDOFD	I			Contains the number of degrees of freedom to be deleted.
23	NFORC	I			Contains the number of excitation forces for the handling of multiple forcing functions.
24	NFREQ	I			Contains the number of frequencies over which the solution will be found.

Table 10.—(Continued)

LABELLED COMMON NAME: SIZOPT (Continued)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
25	NKVAL	I			Contains the number of K-values to be used in an unsteady aerodynamic analysis. If NKVAL is greater than one, then the frequency-dependent matrices will be read and interpolated over for each solution frequency.
26	NLD	I			Contains the number of load equations in the current load set.
27	NPAN	I			Contains the number of panels used for gradual penetration.
28	NULMAT	I	16		An array used to specify the non-null matrices for the Equation of Motions and Load Equations to be read from either input cards or magnetic files. =0 matrix <u>not</u> read. =1 matrix will be read. (see the next page)

Table 10.–(Continued)

LABELLED COMMON NAME: SIZOPT (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION

Nulmat	Matrix to be Read: EOM		Matrix to be Read: Loads
1	$[M_1]$	or	$[\bar{M}_1]$
2	$[M_2]$	or	$[\bar{M}_2]$
3	$[M_3]$	or	$[\bar{M}_3]$
4	$[M_4]$	or	$[\bar{M}_4]$
5	$[M_5]$	or	$[\bar{M}_5]$
6	$[M_6]$	or	$[\bar{M}_6]$
7	$\{C_2\}$	or	$\{\bar{C}_2\}$
8	$\{C_3\}$	or	$\{\bar{C}_3\}$
9	$\{f_\ell\}$		
10	$[\tilde{\Phi}]$	or	$[\bar{\tilde{\Phi}}]$
11	$[S_1]$	or	$[\bar{S}_1]$
12	$[S_2]$	or	$[\bar{S}_2]$
13	$[S_3]$	or	$[\bar{S}_3]$
14	$[S_4]$	or	$[\bar{S}_4]$
15	$[S_5]$	or	$[\bar{S}_5]$
16	$[S_6]$	or	$[\bar{S}_6]$

Table 10.–(Continued)

LABELLED COMMON NAME: SRTDTA

DESCRIPTION: SORTQLS data constants

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	JSET	I			Index of the set of loads to be sorted.
2	ITYPE	I			Option indicating the variable to be sorted. 1 = Q's 2 = Loads 3 = Spectrums 4 = RMS load values
3	IPAIR	I			Denotes whether the standard and the static-elastic solutions will be paired. =0 separate =1 pair
4	JSTATE	I			Index on the solution to be sorted. =0 standard =1 static-elastic
5	NP	I			Number of responses to sort.
6	IRRAY	I	100		Array containing the response numbers to be sorted.
7	SCALX	R			Scalar constant for the output frequencies.
8	SCALY	R			Scalar constant for the output responses.

Table 10.–(Continued)

LABELLED COMMON NAME: SRTDTA (Concluded)

DESCRIPTION:

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
9	NEWTPE	H			Name of the magnetic file which will contain the sorted plot vectors.

Table 10.—(Continued)

LABELLED COMMON NAME: TAPBUF

DESCRIPTION: File Buffer areas for SORTQLS

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	LA1	I			Length of ARAY1.
2	ARAY1	R	1000		Buffer area used for "INTAPE" within SORTQLS.
3	LA2	I			Length of ARAY2.
4	ARAY2	R	1000		Buffer area used for "NEWTPE" within SORTQLS .

Table 10.–(Continued)

LABELLED COMMON NAME: TAPNUM

DESCRIPTION: File Constants

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	ITAPE	H			Contains the magnetic file name (either "INTAPE" or "LDTAPE", if defined) from which MATRED is to read a matrix.
2	IPR	I			Variable which contains the special trouble-shooting printout option.
3	IRLCMP	I			Indicates the matrix type. =0 Real matrix =1 Complex matrix

Table 10.–(Continued)

LABELLED COMMON NAME: VARDIM

DESCRIPTION: Variable Dimensioning Control Variables

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	NMAX	I			Maximum number of arrays possible in VARDIM storage.
2	NENTRY	I			Number of arrays currently defined in VARDIM storage.
3	LWAVAL	I			Last word available in blank common.
4	LWUSED	I			Last word currently being used in blank common by a VARDIM array.
5	LKAT	I			First word address of KATLOG in blank common.
6	MAXUSD	I			Maximum core used by VARDIM since calling routine STARTR.

Table 10.—(Concluded)

LABELLED COMMON NAME: blank

DESCRIPTION: Blank common

NO.	VARIABLE	TYPE	DIM.	ENG. NOMENCLATURE	DESCRIPTION
1	D	R	1000		The blank common will be used to store the matrices under a variable dimensioning storage scheme (VARDIM).

Description of Blank Common Used

The array or matrix sizes required in the modules of L221 (TEV156) vary widely from one problem to another. In fact, in most cases more than 50% of the matrices are null and need not be stored. For these reasons the VARDIM routines described later in this section are used to dynamically allocate core, compact storage, and keep track of most array locations and sizes.

The arrays independent of problem size are still defined in the standard FORTRAN fashion. Only the arrays using VARDIM will be discussed in this section.

Program L221vc

No VARDIM array storage is required.

Program FINDRMS

Three different blocks of arrays are defined in VARDIM storage as FINDRMS is executed. The arrays will be defined only if the statement under the heading option-dependent is true.

- Block I is defined in the subroutine CONDAT and contains the arrays of input constants and frequencies (table 11). Block I is made up of card sets 3 through 11.

- Block II is defined in the subroutine GCMAT and contains the arrays necessary to calculate the generalized coordinates (table 12). Block II is made up of card sets 12 through 30.

- Block III is defined in the subroutine LOMAT and contains the arrays necessary to calculate the load frequency response and the RMS values (table 13). All arrays of Block II except Q, QMAGR, and FLL are deleted before Block III is defined. Block III is made up of card sets 31 through 49.

Table 11.—Input Constants and Frequencies of Block I

Array Name	Size	Option-dependent
NSTEL	NDOF	ISTATE>0
NDEL	NDOFD	NDOFD>1
DAMP	NDOF	--
CS2	NDOF	IFDBAK=2
CS1	NDOF	IFDBAK=2
CS0	NDOF	IFDBAK=2
FREQ	NFREQ	--
FREQS	NSP	
SPECIN	NSP*2*NELE	ISPEC<0
SPFR	2*NELE	ISPEC<0

where NELE = (NFORC*(NFORC+1))/2

NSP = ABS(ISPEC)

Table 12.—Arrays Contained in Block II

Array Name	Size	Option-dependent
Q	2*NDOF*NFORC	--
QMAGR	NDOF*NFORC	--
FREQM	NKVAL	NKVAL>1
FLL	NPAN	NPAN>0
PHIT	2*NDOF*NPAN	NPAN>0
C2	NDOF*NFORC	--
C3	2*NDOF*NFORC	--
SUMM	2*NDOF*NDOF	--
SUMC	2*NDOF*NFOC	--
M1	NDOF*NDOF	NULMAT(1)>0
M2	↓	NULMAT(2)>0
M3	↓	NULMAT(3)>0
M4	↓	NULMAT(4)>0
M5	↓	NULMAT(5)>0
M6	NDOF*NDOF	NULMAT(6)>0
		IFDBAK>0, and
S1	NDOF*NDOF	NULMAT(7)>0
S2	↓	NULMAT(8)>0
S3	↓	NULMAT(9)>0
S4	↓	NULMAT(10)>0
S5	↓	NULMAT(11)>0
S6	NDOF*NDOF	NULMAT(12)>0

Table 12.—(Concluded)

Array Name	Size	Option-dependent
		NKVAL>1, and
M41	NDOF*NDOF	NULMAT(4)>0
M42	↓	
M51	↓	NULMAT(5)>0
M52	↓	
M61	↓	NULMAT(6)>0
M62	NDOF*NDOF	
C31	2*NDOF*NFORC	NPAN<0
C32	2*NDOF*NFORC	
PHIT1	2*NDOF*NPAN	NPAN>0
PHIT2	2*NDOF*NPAN	

Table 13.—Arrays Contained in Block III

Array Name	Size	Option-dependent
LOAD	ICORR	
LOADC	ICORR*10	
NLDC	ICORR	ICORR>0
CORSUM	ICORR*10	
CORLST	ICORR*10	
QMAGL	NLD*NFORC	--
SPEC	↓	--
A	↓	--
PM	↓	--
C2B	NLD*NFORC	--
PHITB	2*NLD*NPAN	NPAN>0
C3B	2*NLD*NFORC	--
SUMMB	2*NLD*NDOF	--
SUMCB	2*NLD	--
PHIOME	(NFORC*(NFORC+1))/2	--
MB1	NLD*NDOF	NULMAT(1)>0
MB2	↓	NULMAT(2)>0
MB3	↓	NULMAT(3)>0
MB4	↓	NULMAT(4)>0
MB5	↓	NULMAT(5)>0
MB6	NLD*NDOF	NULMAT(6)>0

Table 13.—(Concluded)

Array Name	Size	Option-dependent
		IFDBAK>0, and
SB1	NLD*NDOF	NULMAT(7)>0
SB2		NULMAT(8)>0
SB3		NULMAT(9)>0
SB4		NULMAT(10)>0
SB5		NULMAT(11)>0
SB6	NLD*NDOF	NULMAT(12)>0
		NKVAL>1, and
M41	NLD*NDOF	NULMAT(4)>0
M42		
M51		NULMAT(5)>0
M52		
M61		NULMAT(6)>0
M62	NLD*NDOF	
PHIT1	2*NLD*NPAN	NPAN>0
PHIT2	2*NLD*NPAN	
C31	2*NLD*NFORC	NPAN<0
C32	2*NLD*NFORC	

Program SORTQLS

Two or three arrays will be defined in VARDIM for SORTQLS.

Array name	Size	Option-dependent
FREQ	NFREQ	-
SOL1	(NFREQ*NR)	-
SOL2	(NFREQ*NR)	IPAIR>0

where NR is NDOF or NLD, depending upon the response being plotted (NDOF for QMAGR and NLD for QMAGL and/or SPEC), and IPAIR is 1 if pairing of standard and static elastic solutions has been requested.

Program PLOTQLS

The required VARDIM storage is the same as described in program SORTQLS above.

Variable Dimensioning Storage Scheme (VARDIM)

Analyses requiring the storage of matrices can lead to inefficient large core programs (i.e., coded for the maximum matrix sizes) unless some method is used to make the storage required dependent upon the individual problem's size. A series of five subroutines (STARTR, INITIR, LOCATR, DELETR, and XFER) collectively known as VARDIM, has been written to handle the allocation of matrix storage during program execution. VARDIM uses blank common for all array storage. This method is possible on operating systems that place blank common at the end of all other program storage. The user must request enough field length (core) to provide sufficient array storage between the beginning of blank common and the end of the declared field length.

VARDIM is not a matrix language. It does not provide matrix operations, only array storage allocations and the necessary bookkeeping. The subroutines were designed to be called by FORTRAN programs that do their own matrix storage (input/output), perform their own matrix operations, and calculate their own array subscripts.

Some general features of the VARDIM scheme follow:

1. Array storage may be variable within a single program run as well as between runs. Arrays may be defined or deleted at any time. It is not necessary to define them all at the beginning of the program.

2. Each array will be identified by a six-character hollerith name (left-justified), may have from one to three dimensions, and may be a real or integer variable type.

3. Newly defined arrays will be null.

4. The array storage is always compacted to use the first words of blank common. A newly defined array is always located after pre-existing arrays. If array i is deleted, then arrays i+1 through n are moved forward to positions i through n-1.

5. When any VARDIM routine is called, it checks to see that there are no duplicate array names or illegal dimensions and that core and/or the array catalog has not been exceeded.

   The bookkeeping performed by VARDIM is stored in a catalog array KATLOG and the labeled common block /VARDIM/.

6. The VARDIM array catalog, KATLOG, is itself an array in VARDIM storage with the dimensions (6, NMAX, 1). Each array will have a six-word entry in KATLOG (i.e., one column). The six words contain:

   a. Name–6 hollerith characters, left-justified and blank filled
   b. Location–the first word address of the array in blank common
   c. Type–0 for integer and 1 for real
   d. Row dimension size
   e. Column dimension size
   f. Level dimension size

7. All VARDIM routines except XFER contain the labeled common block /VARDIM/, which has the bookkeeping variables:

NMAX	Maximum number of arrays which may be defined for this program
NENTRY	Number of arrays currently defined
LWAVAL	Last word available in blank common (length of blank common)
LWUSED	Last word currently in use by VARDIM in blank common
LKAT	The first word address in blank common of the array named KATLOG
MAXUSD	Maximum core used by VARDIM since calling routine STARTR

8. All VARDIM arrays except XFER contain the blank common definition

```
COMMON D(1)
DIMENSION ID(1), KATLOG (6,1)
EQUIVALENCE (D,ID,KATLOG)
```

9. A program checkout feature is available. The VARDIM routines contain the following labeled common block.

COMMON / CHKPRT / ICKPRT

If ICKPRT$\geq$5, each VARDIM routine except XFER will print a message giving information about the array and core locations it is manipulating. There are two print lines for each message.

NOTES:

- For a description of the VARDIM routines used by the overlays of L221 (TEV156), see the discussion of the VARDIM routines beginning on the next page.
- It will be necessary to use the REDUCE(-) control card to prevent the loader from reducing the program's executable field length to dimensioned blank common size of 1.

Discussion of VARDIM Routines

The purpose and usage of each of the VARDIM routines will be briefly described. The variables below appear repeatedly in their argument lists.

- NAMEA — An array name of up to 6 hollerith characters (left-justified)
- LA — First dimension of array NAMEA
- MA — Second dimension of array NAMEA
- NA — Third dimension of array NAMEA
- INTRA — A key to the VARDIM routines indicating NAMEA's variable type (0 for integer and 1 for real)
- LOCA — The location of array NAMEA-the first word address in blank common

Subroutine STARTR

STARTR must be called before any other VARDIM routine. It creates the array catalog, sets the maximum number of arrays, and determines the maximum array storage available.

Calling sequence of STARTR:

CALL STARTR (IMAX,LENGTH)

Input

IMAX — The maximum number of arrays to be defined at any one time; if IMAX = 0, STARTR will assume 100

LENGTH — The number of words available in blank common for use by the VARDIM routines; if length is input as zero, STARTR will find the length available (difference between the beginning of blank common and end of field length)

Output

LENGTH — The number of words available in blank common; if LENGTH is returned as zero, there was not enough blank common to create the catalog array, KATLOG

Subroutine INITIR

INITIR is called to initialize or define a new array. It must be called before any other VARDIM routine refers to the array. INITIR allocates the array storage, zeros the area, and makes an entry in the array catalog. When the array being defined already exists, the elements are simply set equal to zero if the dimensions are to remain the same. If the array size is to be changed, the old array is deleted and a new one defined.

Calling sequence of INITIR:

CALL INITIR (NAMEA,LA,MA,NA,INTRA,LOCA,IRI)

Input

NAMEA, LA, MA, NA, INTRA — See subroutine argument definitions in discussion of VARDIM routines

Output

LOCA — See subroutine argument definitions

IRI — Error code

- = 0, no error detected
- = -1, previous array of same name deleted before defining a new one
- = 1, maximum number of catalog entries was exceeded
- = 2, one of the arrays dimensions is 0
- = 3, blank common storage exceeded
- = 4, routine STARTR was not called beforehand.

Subroutine DELETR

DELETR is called to eliminate an array from VARDIM storage. The array's entry in the VARDIM catalog will disappear and both the catalog and array storage compacted.

Calling sequence of DELETR:

CALL DELETR (NAMEA,IRD)

Input

NAMEA — See subroutine argument definitions

Output

IRD	Error code	
	=	0, no error detected
	=	-2, NAMEA is not in the VARDIM catalog

Subroutine LOCATR

LOCATR is called to determine an array's size, type, and location. LOCATR should be called just before handling the array because the array's location changes as other arrays are deleted and added to VARDIM storage.

Calling sequence of LOCATR:

CALL LOCATR (NAMEA,LA,MA,NA,INTRA,LOCA,IRL)

Input

NAMEA	See subroutine argument definitions

Output

LA	See subroutine argument definitions	
MA		
NA		
INTRA		
LOCA		
IRL	Error code	
	=	0, no error detected
	=	-1, no entries in catalog
	=	-2, NAMEA is not in the catalog

Subroutine XFER

XFER is a subroutine called by DELETR to quickly move blocks of core.

4.0 EXTENT OF CHECKOUT

Nine different data cases were used to verify that L221 (TEV156) runs correctly. The results of the nine cases were compared against answers produced by hand calculations and previous versions of the program (see footnote 1 in sec. 2).

Checkout Problems

No.	Type of problem
1	Standard execution with input equations of motion and load equations from cards
2	Standard execution with equations input on a magnetic file; SORTQLS and PLOTQLS also
3	Execution with frequency-dependent equations of motion and load equations input on cards
4	Same as 3 except equations input via a magnetic file
5	Multiple forcing functions input on cards
6	Same as 5 except equations input via a magnetic file
7	Same as 1 except freedoms physically deleted from cards; serves as a check case for number 8
8	Same as 1 except freedoms deleted through use of a program option
9	Magnetic file input of matrices (same as 4) with DYLOFLEX header arrays on the file

The options used in the checkout problems are displayed in table 14.

Boeing Commercial Airplane Company
P.O. Box 3707
Seattle, Washington 98124
May 1977

Table 14.—Options Used in Checkout Cases

DATA CASES

OPTIONS	1	2	3	4	5	6	7	8	9
NDOF	7	7;7	7	8	2;7	2;7	5	7	8
NFREQ	10	10;10	10	210	3;72	3;72	10	10	210
NPAN	0	5;0	0	0	0;0	0;0	0	0	0
NKVAL	0	0;0	10	20	0;0	0;0	0	0	20
NFORC	0	0;0	0	0	2;6	2;6	0	0	0
NDOFD	0	0;0	0	0	0;0	0;0	0	2	0
IFREQ	0	0;0	0	0	-1;0	-1;0	0	0	0
ISPEC	1	1;-10	1	1	-3;-26	-3;-26	1	1	1
ISTATE	0	2;0	0	0	0;0	0;0	0	0	0
IFDBAK	0	1;0	0	0	0;0	0;0	0	0	0
IDAMP	0	1;0	0	0	0;0	0;0	0	0	0
INTZRO	1	0;1	1	1	1;1	1;1	1	1	1
IPRINT	1	1;0	0	0	1;1	1;1	1	1	0
IRMSPR	0	3;0	0	0	3;72	3;72	0	0	0
IPUNCH	0	0;0	0	0	0;0	0;0	0	1	0
ICKPRT	2	2;0	0	2	5;1	5;1	5	5	2
IPLRMS	0	1;0	0	0	0;0	0;0	0	0	0
LPDOF	0	2;0	0	0	0;0	0;0	0	0	0
INTAPE		INTAPE;-		EOMTPE					DYFEOM
IPLTPE	ONE11	TWO11;-	THRE11						
IRTAPE		TAPE3;-		IRTAPE					
IFTAPE						;IFTAPE			
Z	12.	12.;12.	12.	1.	1.;1.	1.;1.	12.	12.	1.
VEL	2613.83	2613.83; 2613.83	2613.83	829.53	1.;1.	1.;1.	2613.83	2613.83	829.53
L(TSCALE)	2500.	2500;2500	2500.	2500.	1.;1.	1.;1.	2500.	2500.	2500.
C(GPSCAL)	0	1.;0.	0.	0.	0.;0.	0.;0.	0.	0.	0.
TABFS	0.	0.;0.	0.	0.	0.;0.	0.;0.	0.	0.	0.
TABS	0.	0.;0.	0.	0.	0.;0.	0.;0.	0.	0.	0.

Table 14.—(Concluded)

DATA CASES

OPTIONS	1	2	3	4	5	6	7	8	9
FDSCAL	0	0.;0.	0.	0.	0.;0.	0.;0.	0.	0.	0.
NLD	6	6,9;6	6	8	9;7	9;7	6	6	8
ICØRR	0	0,2;0	0	0	0;0	0;0	0	0	0
$\overline{C}$(GPSCLB)	0	1;1;0.	0.	0.	0.;0.	0.;0.	0	0	0
LDTAPE				LØDTPE					DYFLØD
DYLØFLX									✓
SØRTQLS		-;✓							
NEWTPE		-;TAPE12							
PLØTQLS		✓;-							

REFERENCES

1. Miller, R. D.; Kroll, R. I.; and Clemmons, R. E.: Dynamic Loads Analysis System (DYLOFLEX) Summary. NASA CR-2846-1, 1979.

2. Miller, Ronald D.; Richard, Michael; and Rogers, John T.: Feasibility of Implementing Unsteady Aerodynamics Into the FLEXSTAB Computer Program System. NASA CR-132530, 1974.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-2858		

4. Title and Subtitle	5. Report Date
Random Harmonic Analysis Program, L221 (TEV156) Volume II: Supplemental System Design and Maintenance Document	SEPTEMBER 1979
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
M. L. Graham, R. E. Clemmons, and R. D. Miller	D6-44467
9. Performing Organization Name and Address	10. Work Unit No.
Boeing Commercial Airplane Company P.O. Box 3707 Seattle, Washington	11. Contract or Grant No. NAS1-13918
12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, DC 20546	Contractor Report May 1975 to May 1977
	14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitors: Robert C. Goetz and Boyd Perry III
Topical Report

16. Abstract

L221 (TEV156) is a digital computer program available for execution on the CDC 6600. The program is capable of calculating steady-state solutions for linear second-order differential equations due to sinusoidal forcing functions. From this steady-state solution, generalized coordinates and load frequency responses may be determined. Statistical characteristics of loads for the forcing function spectral shape may also be calculated using random harmonic analysis techniques.

The particular field of application of the program is the analysis of airplane response and loads due to continuous random air turbulence. Optional capabilities include frequency dependent input matrices, feedback damping, gradual gust penetration, multiple excitation forcing functions, and a static elastic solution.

Program usage and a brief description of the analysis used are presented in volume I of this document. Volume II contains a description of the design and structure of the program to aid those persons who will maintain and/or modify the program in the future.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
PSD – Power Spectral Density Frequency Response Dynamic Loads	Unclassified – unlimited Subject Category 02

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	110	$6.50

*For sale by the National Technical Information Service, Springfield, Virginia 22151