UNCLASSIFIED

AD NUMBER

AD838826

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; MAR 1968. Other requests shall be referred to Air Force Rocket Propulsion Lab., Edwards AFB, CA 93523.

AUTHORITY

AFRPL ltr, 27 Oct 1971

THIS PAGE IS UNCLASSIFIED

AD838826

AFRPL-TR-67-314
Volume II

THE SENSITIVE TIME LAG THEORY AND ITS APPLICATION TO LIQUID ROCKET COMBUSTION INSTABILITY PROBLEMS

Volume II: Programmer's Manual

A. J. Smith, Jr.
F. H. Reardon
et al.

TECHNICAL REPORT AFRPL-TR-67-314, VOLUME II

March 1968

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of AFRPL (RPPR/STINFO), Edwards, California, 93523.

AIR FORCE ROCKET PROPULSION LABORATORY
Research and Technology Division
Air Force Systems Command
Edwards Air Force Base, California 93523

AFRPL-TR-67-314
Volume II

THE SENSITIVE TIME LAG THEORY AND ITS APPLICATION TO LIQUID ROCKET COMBUSTION INSTABILITY PROBLEMS

Volume II: Programmer's Manual

A. J. Smith, Jr.
F. H. Reardon
et al.

Aerojet-General Corporation

March 1968

TECHNICAL REPORT AFRPL-TR-67-314, VOLUME II

This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with prior approval of AFRPL (RPPR/STINFO), Edwards, California, 93523.

AIR FORCE ROCKET PROPULSION LABORATORY
Research and Technology Division
Air Force Systems Command
Edwards Air Force Base, California 93523

FOREWORD

This is the final report submitted in fulfillment of Contract F04611-67-C-0019. The program had two Air Force Project Engineers, Lt. James D. Cox/RPRRC and, later, Capt. J. F. Ensminger/RPRRC. Volume I of this report contains the background, development, and critique of the Sensitive Time Lag Theory of combustion instability, a description of the computer program that is used to provide a solution of the analytical model, and the use of the model in the analysis and design of liquid rocket engines based on empirical correlations. Volume II contains the flowcharts and listings of the computer program. Therefore, Volume I is referred to as the Engineer's Handbook while Volume II is referred to as the Programmer's Handbook.

The contract effort was conducted by the Thrust Chamber Engineering Division of Liquid Rocket Operations under Dr. C. B. McGough, Program Manager; Dr. R. J. Hefner and, during the final phases of the contract, by Mr. J. M. McBride, Project Managers; Mr. A. J. Smith, Jr., Project Engineer; Dr. F. H. Reardon, of Sacramento State College, and Dr. L. M. Crocco, Dr. W. A. Sirignano, and Mr. D. T. Harrje, of Princeton University, Consultants. This report has been prepared in accordance with MIL-STD-847 (USAF) dated 25 February 1965. The period of this report is 2 September 1966 through 31 November 1967.

It is essential to give special acknowledgement to Dr. L. M. Crocco, originator of the Sensitive Time Lag Theory, and his co-workers whose individual contributions to this discertation are reflected throughout. In addition to contributing significant portions of the Background, Theory, and Conclusion sections, as well as Appendices I through III, they have reviewed the entire text and have made many helpful suggestions during the entire course of the contract.

This technical report has been reviewed and is approved.

Capt. J. F. Ensminger
USAF Project Engineer

ABSTRACT

The main objective of this report is to include, under one cover, all of the work concerned with the development of the Sensitive Time Lag Theory of liquid rocket engine combustion instability. This work includes all the aspects of the theory from the mathematical formulation of the analytical model to the application of the model to actual engine problems. The initial section of the report reviews the logical considerations of the instability phenomenon and relates how the time lag concept conforms analytically as well as experimentally to the problem. Thereafter, the mathematics of the model are developed with the major emphasis placed on the linearized model; however, various aspects of the nonlinear model are also discussed. The mathematical analysis gives rise to a computer program, which is presented in Volume I in the form of an Engineer's instruction manual and in Volume II in the form of a detailed description for the Programmer. The report then focuses its attention on the designer and instructs him in not only how to use the model in practical situations but also how to interpret and correlate test data. The main body of the text concludes with a critique of the time lag concept and outlines the kind of research that is needed in order to improve the time lag model.

TABLE OF CONTENTS

			PAGE
I.	PROGRAM LISTINGS		1
	A. LINK ZERO (SYSTEM ORIGIN)		1
	DECK NAME	ENTRY NAME	
	BLOK	---	1
	V8050	--- (MAIN PROGRAM)	1
	OUT	OUTAS8	1
	IN	INAS58	1
	*KTRAN	KOVFLO	1
		KUNFLO	
		KDATEA	
		KERROR	
		KFINIS	
		KINDX4	
		KINDX1	
		KINDX2	
		KOPINP	
		KISORG	
		KDATEZ	
	*AAS58	AS58	2
		AS138	
	*INTG	INTGR	5
		INTGS	
	*KDATE	DATE	6
	*NT4	INT4	6
	NT4D*	INT4D	6
	PAGER	PAGE	7
	B. LINK 1 (ORIGIN ALODE)		8
	DECK NAME	ENTRY NAME	
	CHAMB	CHAMBR	8
	WGEN	GENMEG	10
	C. LINK 2 (ORIGIN BLODE)		11
	DECK NAME	ENTRY NAME	
	HYMN	TRANS	11
	HTINT	DDD	18

TABLE OF CONTENTS (cont.)

DECK NAME	ENTRY NAME		PAGE
*ADMIN	ADMSET	(SET)	20
	ADMINT	(INTEGRATE)	
	ADMCOR	(CORRECT)	
	ADMPAR	(PARTIAL STEP INTEGRATION)	
	ADMRES	(RESET)	
NOZMIT	CCC		23
VELPOT	TBLCAL		27
LONGIT	LONGL		29
NTAU	FFF		30
QUADR	QUAD		30
KORE	CORE		30

D. LINK 3 (ORIGIN ALODE) 31

DECK NAME	ENTRY NAME	
INJ	INJCTR	31
JECT	JJJ	31
INJDS	INJDIS	35
BESS	BESSEL	37

II. PROGRAM FLOW CHARTS 44

A. LINK ZERO 44
1. V8050 44

B. LINK 1
1. CHAMB 45
2. WGEN 48

C. LINK 2 49
1. HYMN 49
2. HTINT 51
3. NOZMIT 52
4. VELPOT 53
5. LONGIT 54
6. NTAU 55

D. LINK 3 56
1. INJ 56
2. JECT 57
3. INJDS 58

```
$EXECUTE       IBJOB                                                    BLOK


$IBJOB         MAP,ALTIO                                                BLOK  10

$IBFTC BLOK    LIST,N94                                                 BLOK  20
      BLOCK DATA                                                       BLOK  30
C                                                                      BLOK  40
      COMMON  /PROLOG/  LOGIK(50), SL1, SL2, EORJ                      BLOK  50
      LOGICAL  LOGIK,  SL1,  SL2,  EORJ                                BLOK  60
C                                                                      BLOK  70
      DATA  SL1, SL2  /  .FALSE., .FALSE.  /                           BLOK  80
      END                                                              BLOK  90


$IBFTC V8050   LIST,N94                                                 ....   0
      COMMON  /PROLOG/  LOGIK(50), SL1, SL2, EORJ                      ....  10
      LOGICAL  LOGIK,  SL1,  SL2,  EORJ                                ....  20
C                                                                      ....  30
   10 CALL  CHAMBR                                                     ....  40
C                                                                      ....  50
C         RETURNS FROM MAJOR PROGRAM IF AND ONLY IF AN INJECTOR PROGRAM ....  60
C          IS TO BE CALLED..., INJCTR DETERMINES WHICH, WRITES SCRATCH ....  70
C          TAPE, AND CALLS PROPER ROUTINE.                             ....  80
C                                                                      ....  90
      CALL  INJCTR                                                     .... 100
C                                                                      .... 110
      GO TO 10                                                         .... 120
         END                                                           .... 130


$IBFTC OUT     LIST,N94                                                 OUT
      SUBROUTINE OUTASS(A,K)                                           OUT   10
      DIMENSION A(12)                                                  OUT   20
      WRITE(6,10)(A(I),I=1,K)                                          OUT   30
   10 FORMAT(1X,12A6)                                                  OUT   40
      RETURN                                                           OUT   50
      END                                                              OUT   60


$IBFTC IN      LIST,N94                                                 IN
      SUBROUTINE INASS0(A)                                             IN    10
      DIMENSION A(12)                                                  IN    20
      READ(5,10)A                                                      IN    30
   10 FORMAT(12A6)                                                     IN    40
      RETURN                                                           IN    50
      END                                                              IN    60


$IBMAP *KTRAN  50,LIST,REF,DECK,N/94,RELMOD                            *KTR
*                                                                      *KTR  10
*      KISMET TRANSFER VECTOR AS APPLIED TO IBM 7094 IBSYS    9/   /64 *KTR  20
*                                                                      *KTR  30
       ENTRY   KOVFLO                                                  *KTR  40
       ENTRY   KUNFLO                                                  *KTR  50
       ENTRY   KDATEA                                                  *KTR  60
       ENTRY   KERROR                                                  *KTR  70
       ENTRY   KFINIS                                                  *KTR  80
       ENTRY   KINDX4                                                  *KTR  90
       ENTRY   KINDX1                                                  *KTR 100
       ENTRY   KINDX2                                                  *KTR 110
       ENTRY   KOPINP                                                  *KTR 120
       ENTRY   KISORG                                                  *KTR 130
       ENTRY   KDATEZ                                                  *KTR 140
KSTCHA EQU     29                                                      *KTR 150
KPRINT EQU     10                                                      *KTR 160
KPUNCH EQU     15                                                      *KTR 170
SYSDAT EQU     65                                                      *KTR 180
KOPINP TRA     1,4                 LEFT FROM 704 DAYS OF OPTIONAL INPUT*KTR 190
KFINAL TSX     KFINIS,4                                                *KTR 200
       TTR     KERROR              AT KFINAL+1                         *KTR 210
KOVFLO TTR     KERROR              INOPERATIVE IN THE KISMET 9 SENSE   *KTR 220
KUNFLO TTR     KERROR              INOPERATIVE IN THE KISMET 9 SENSE   *KTR 230
KINDX1 PZE                                                             *KTR 240
KINDX2 PZE                                                             *KTR 250
KINDX4 PZE                                                             *KTR 260
KDATEA BCI     2,                  NOT AVAILABLE W/OUT PROGRAMMER HELP *KTR 270
KISORG PZE     1486,,1486          PROTECT NUCLEUS                     *KTR 280
KERROR TSX     KFINIS,4                                                *KTR 290
KFINIS CALL    EXIT                                                    *KTR 300
*                                                                      *KTR 310
*      KDATEZ, TODAY-S DATE FOR KISMET-MAP OR FORTRAN IV PROGRAMS      *KTR 320
*                                                                      *KTR 330
KDATEZ TRA     1,4                                                     *KTR 340
       END                                                             *KTR 350
```

```
$IBMAP *AAS58                                                          *AAS0010
*  AS58K9                                                              *AAS0020
*                                                                      *AAS0030
        ENTRY   AS58                                                   *AAS0040
        ENTRY   AS138                                                  *AAS0050
KISORG  EQU     1480                                                   *AAS0060
 AS138  SAVE                                                           *AAS0070
        CAL     4,4                                                    *AAS0080
        ALS     18                                                     *AAS0090
        ORA     3,4                                                    *AAS0100
        SLW     BOGUS                                                  *AAS0110
        CAL     5,4                                                    *AAS0120
        LXA     GETOUT-2,4                                             *AAS0130
        TXH     BADGUY,4,0          CHECK FOR PREVIOUS EOF             *AAS0140
        STA     GETOUT                                                 *AAS0150
        TSX     AS58,4                                                 *AAS0160
 BOGUS  EQU     *                                                      *AAS0170
        PZE     **,,**              DATA,,TCARD                        *AAS0180
        TTR     GETOUT-2                                               *AAS0190
        TTR     GETOUT-2                                               *AAS0200
        PXA     **,0                EOF ENCOUNTERED FLAG IF ADDR UNZERO *AAS0210
        ADD     58ONE                                                  *AAS0220
GETOUT  STO     **                                                     *AAS0230
        RETURN  AS138                                                  *AAS0240
BADGUY  EQU     *                                                      *AAS0250
        SXA     GETOUT-2,0          RESET EOF FLAG                     *AAS0260
        TTR     KFINIS              WRITE A NOTE....THEN FLUSH HIM     *AAS0270
EOFMES  BCI     9,1READING PAST END-OF-FILE OR INPUT TAPE...JOB FLUSHED *AAS0280
AS58    CLS     1,4                                                    *AAS0290
        SXA     58IR1,1                                                *AAS0300
        SXA     58IR2,2                                                *AAS0310
        SXD     58IR4,4                                                *AAS0320
AS58X4  LXD     58IR4,4                                                *AAS0330
        CAL     1,4                                                    *AAS0340
        STA     *+5                                                    *AAS0350
        STT     *+4                                                    *AAS0360
        ARS     18                                                     *AAS0370
        STA     *+6                                                    *AAS0380
        STT     *+5                                                    *AAS0390
        PXA     **,**                                                  *AAS0400
        SBM     *-1                                                    *AAS0410
        STA     AS58D               SETD,T FOR DATA                    *AAS0420
        STA     AS58X                                                  *AAS0430
        PXA     **,**                                                  *AAS0440
        SBM     *-1                                                    *AAS0450
        STA     AS58BZ              SET B,T FOR                        *AAS0460
        STA     AS58B               SET B,TAG FOR BCD HEADER           *AAS0470
        STO     AS58C2-3            SET TAPE ERROR FLAG REG.           *AAS0480
        SBM     AS58E+9             L(12) HEAD+12                      *AAS0490
        STA     AS58C1                                                 *AAS0500
        LXA     KOVFLO,2                                               *AAS0510
        SXA     AS58C+9,2                                              *AAS0520
        AXT     AS58Q,2                                                *AAS0530
        SXA     KOVFLO,2                                               *AAS0540
        STZ     58YYAS+2            0 TO CARD ERROR IND.               *AAS0550
        LXA     GETOUT-2,4          EOF CHECK                          *AAS0560
        TXH     BADGUY,4,0          READING PAST EOF                   *AAS0570
        LXA     AS58A,4                                                *AAS0580
        TXH     AS58A+4,KISORG      STORAGE INTO MONITOR               *AAS0590
        CALL    OUTAS8(58BCD,7)                                        *AAS0600
        TRA     KERROR                                                 *AAS0610
 58BCD  BCI     7, B,TAG TOO LOW IN MEMORY USING AS58 INPUT            *AAS0620
        BCI     2, DATA    D,T                                         *AAS0630
AS58Z   TXH     AS58A,4,0                                              *AAS0640
        TSX     OUTAS8,4                                               *AAS0650
        TRA     AS58A                                                  *AAS0660
        TTR     119                                                    *AAS0670
 AS58B  PZE     **                                                     *AAS0680
        CLA     =12                                                    *AAS0690
 AS58A  TSX     INAS58,4                                               *AAS0700
        TRA     58IR4                                                  *AAS0710
        TTR     119                                                    *AAS0720
AS58BZ  PZE     **                                                     *AAS0730
 58IR4  TXL     *+9,,**             NORMAL RETURN                      *AAS0740
 58ERR  CLS     AS58C2-1                                               *AAS0750
        STO     AS58C2-3        ERROR FLAG                             *AAS0760
        CAL*    AS58B                                                  *AAS0770
        LRS     30                                                     *AAS0780
        SUB     AS38J                                                  *AAS0790
        TNZ     AS58C2                                                 *AAS0800
        CLA     AS58C2-2                                               *AAS0810
        LLS     30                                                     *AAS0820
AS58C   SLW*    AS58B                                                  *AAS0830
        LXD     58IR4,4                                                *AAS0840
        LDQ     AS58C2-3                                               *AAS0850
        CAL     58YYAS+2                                               *AAS0860
 58IR2  AXT     **,2                                                   *AAS0870
        AXT     **0,1               RESTORE TRA VECTOR                 *AAS0880
        SXA     KOVFLO,1                                               *AAS0890
 58IR1  AXT     **,1                                                   *AAS0900
        TQP     *+2                                                    *AAS0910
        TZE     4,4             NORMAL OR EOF EXIT                     *AAS0920
        LXD     58IR4,4                                                *AAS0930
        TRA     2,4             ERROR EXIT FROM AS58                   *AAS0940
58EOF   LF      58IR4,4                                                *AAS0950
        STA     GETOUT-2,4          EOF IND                            *AAS0960
        TXI     AS58C+2,4,1                                            *AAS0970
        TNX     AS58C+2,1       SKIP IF END OF CARD                    *AAS0980
AS58C1  LDQ     **0,2                                                  *AAS0990
        TXI     AS58E+1,2,500       RETURN WITH NEXT BCD WORD IN MQ    *AAS1000
 AS58P  CALL    OUTAS8(58BCD+9,1)                                      *AAS1010
        CALL    OUTAS8(58BCD+1,6)                                      *AAS1020
AS58Q   AXT     0,4                 TRA HERE IF ERROR                  *AAS1030
        CLA     58YYAS+2                                               *AAS1040
        ALS     1                                                      *AAS1050
        TXH     *+2,4,0                                                *AAS1060
        ORA     58ONE               FX1                                *AAS1070
        STO     58YYAS+2                                               *AAS1080
        ORS     58YYAS+2                                               *AAS1090
        TXL     AS58Z,0,32      TO READ NEXT CARD                      *AAS1100
AS58C3  TXL     AS58Q+1,2,0                                            *AAS1110
        LXD     *-2,1                                                  *AAS1120
        TXI     AS58E,2,-1                                             *AAS1130
 AS58J  BCT     63,21                                                  *AAS1140
        BCT     3000000,63,21                                          *AAS1150
        DEC     9000000,900000,90000,9000,900,90,9,294                 *AAS1160
        DEC     8000000,800000,80000,8000,800,80,8,128                 *AAS1170
        DEC     7000000,700000,70000,7000,700,70,7,216                 *AAS1180
        DEC     6000000,600000,60000,6000,600,60,6,163                 *AAS1190
        DEC     5000000,500000,50000,5000,500,50,5,280                 *AAS1200
        DEC     4000000,400000,40000,4000,400,40,4,136                 *AAS1210
```

```
        DEC 3000000,300000,30000,3000,300,30,3,344                      *AAS1220
        DEC 2000000,200000,20000,2000,200,20,2,13383                    *AAS1230
        DEC 1000000,100000,10000                                        *AAS1240
58MIL   DEC 1000            FX1000                                      *AAS1250
58CENT  DEC 100             FX100                                       *AAS1260
58TEN   DEC 10              FX10                                        *AAS1270
58ONE   DEC 1               FX1                                         *AAS1280
        DEC 1E9,1E8,1E7,1E6,1E5,1E4,1E3,1E2,1E1                         *AAS1290
        DEC 0,48                                                        *AAS1300
        SVN   0-1,7,-1                                                  *AAS1310
AS58C2  AXT    0,1                                                      *AAS1320
        AXT    0,2                                                      *AAS1330
        STP   AS58G+3                                                   *AAS1340
        STZ   AS58Y1                ZERO FIRST NUMBER FLAG              *AAS1350
        TXI   AS58D1+1,2,13                                             *AAS1360
        LXA   *+9,2                                                     *AAS1370
        TXL    AS58P,2,KISOR6       STORAGE INTO MONITOR                *AAS1380
        TXI   *+1,2,1               BUMP STORAGE ADDR.                  *AAS1390
AS58D   STO   **0                                                       *AAS1400
        SXA   *-1,2                                                     *AAS1410
        SXA   AS58Y1,2                                                  *AAS1420
        PXD   1,0                                                       *AAS1430
        TXL   *+2,1,56                                                  *AAS1440
        SSM                                                             *AAS1450
        STO   58YYAS+1                                                  *AAS1460
        LXD   AS58F1+5,2             RELOAD                             *AAS1470
AS58D1  LDQ   58YYAS+3                                                  *AAS1480
        SXD   AS58F,0                                                   *AAS1490
        SXD   AS58V,1           SAVE IR1                                *AAS1500
        STZ   58YYAS                                                    *AAS1510
        LXD   AS58F+1,4         7 TO IR4                                *AAS1520
        SXD    AS58K,4                                                  *AAS1530
        PXD   0,0                                                       *AAS1540
AS58E   TNX   AS58C1-1,2,100        TRA IF NO EXHAUSTED                 *AAS1550
        LGL   6                                                         *AAS1560
        ALS   3                                                         *AAS1570
        PAX    12,1                                                     *AAS1580
        TXH   AS58F1,1,72                                               *AAS1590
        CAL   58YYAS                                                    *AAS1600
        TXL   AS58F2,1,0                                                *AAS1610
AS58K   TXI    *+1,1,**                                                 *AAS1620
        ADD    AS58C2-4,1                                               *AAS1630
        SLW   58YYAS                                                    *AAS1640
        TIX    AS58E-2,4,1                                              *AAS1650
        STU   58YYAS+3                                                  *AAS1660
        ORA   58MIL-4                                                   *AAS1670
        FAD   58MIL-4                                                   *AAS1680
        LDQ   58YYAS+1                                                  *AAS1690
        LLS   0                                                         *AAS1700
        LXD   *+2,4             7 TO IR4                                *AAS1710
AS58F   TXI   *+1,4,***0                                                *AAS1720
        TXL   *+4,0,7                                                   *AAS1730
        STO   58YYAS                                                    *AAS1740
        FMP   AS58C2-10         1E7                                     *AAS1750
        FAD   58YYAS                                                    *AAS1760
        STO   58YYAS+1                                                  *AAS1770
        LDQ   58YYAS+3                                                  *AAS1780
        SXD   AS58F,4                                                   *AAS1790
        TXL    AS58E-4,,64                                              *AAS1800
AS58F1  LXD   *-1,1                                                     *AAS1810
        CAS   58ONE+1,1                                                 *AAS1820
        TIX   *-1,1,8                                                   *AAS1830
        TXH   AS58G-2,1,8                                               *AAS1840
        TIX   *-3,1,0                                                   *AAS1850
        TXL   AS58Q,0,***0              BCD POSITION                    *AAS1860
AS58F2  ZET   AS58Y1                                                    *AAS1870
        TRA    AS58E+10                                                 *AAS1880
        AXT   56,1                                                      *AAS1890
        SXA   AS58Y1,1                                                  *AAS1900
        TXL    AS58E-2,1,16                                             *AAS1910
AS58G   STO   58YYAS+3                                                  *AAS1920
        SXD   AS58F1+5,2                                                *AAS1930
        LXD   AS58F,2                                                   *AAS1940
        NOP   AS58X-1                                                   *AAS1950
        TXH   AS58H,4,6                                                 *AAS1960
        CLA   58YYAS                                                    *AAS1970
        ORA   58MIL-4                                                   *AAS1980
        FAD   58MIL-4                                                   *AAS1990
        FDP   AS58C2-3,4                                                *AAS2000
        CLA   58YYAS+1                                                  *AAS2010
        LRS   0                                                         *AAS2020
        STQ   58YYAS                                                    *AAS2030
        SXD   *+2,4                                                     *AAS2040
        LXD   AS58F+1,4         7                                       *AAS2050
        TNX   AS58H+1,4,***0                                            *AAS2060
        SXD   *+4,4                                                     *AAS2070
        TXL   *+3,2,0                                                   *AAS2080
        LDQ   58YYAS+1                                                  *AAS2090
        FMP   AS58C2-3,4                                                *AAS2100
        TXI   *+2,2,***0                                                *AAS2110
AS58H   CLA   58YYAS+1                                                  *AAS2120
        FAD   58YYAS                                                    *AAS2130
        STO   58YYAS+1                                                  *AAS2140
        LXD   AS58V,4                                                   *AAS2150
        TXH   **4,4,48                                                  *AAS2160
        TXH   AS58V+1,4,40                                              *AAS2170
        TXL   *+3,1,48                                                  *AAS2180
        TXL   AS58D+3,2,0                                               *AAS2190
        TXH   AS58D-3,1,48          TO STORE RESULT                     *AAS2200
        TXL   AS58W,1,40                                                *AAS2210
        SXD   AS58F,2                                                   *AAS2220
        SXD   *+4,2                                                     *AAS2230
        LXD   AS58F1+5,2                                                *AAS2240
        LDQ   58YYAS+3                                                  *AAS2250
AS58V   TXL   AS58D1+2,0,***0           CODE                             *AAS2260
        TNX   AS58W-1,2,***0                                            *AAS2270
        CLA   58YYAS+1                                                  *AAS2280
        TXL   *+4,2,9                                                   *AAS2290
        FDP   AS58C2-12         1E9                                     *AAS2300
        STQ   58YYAS+1                                                  *AAS2310
        TXI   *-4,2,-9                                                  *AAS2320
        TXL   *+4,2,0                                                   *AAS2330
        FDP   AS58C2-3,2                                                *AAS2340
        STO   58YYAS+1                                                  *AAS2350
        CLA   58YYAS+1                                                  *AAS2360
        SXD   AS58V+1,0                                                 *AAS2370
        TXH   AS58D-3,1,40          STORE IF , OR - OR +                *AAS2380
AS58W   LDQ   **7               NEXT IS E,L,A                           *AAS2390
        SLQ   AS58G+3           SET FIXED                               *AAS2400
        SLQ   AS58Y             SET POSITIVE                            *AAS2410
        TXH   AS58D1-1,1,92         NEXT CODE E                         *AAS2420
```

```
        [illegible]   [illegible]        [illegible] STORE IF PREV. CODE [illegible]      [illegible]
        [illegible]   [illegible]        [illegible] GO [illegible]                       [illegible]
        [illegible]   [illegible]        [illegible] STORE IF [illegible] MODE [illegible] [illegible]
        [illegible]   [illegible]        [illegible] READ [illegible] NE CODE             [illegible]
        [illegible]   [illegible]        ERROR                                            [illegible]
[illegible] [illegible]  [illegible]                                                      [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        SET [illegible]                                  [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        SET FIRST                                        [illegible]
[illegible] [illegible]  [illegible]                                                      [illegible]
        [illegible]                                                                       [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        ERROR                                            [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        L OR A                                           [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        SET NEW STORAGE LOCATION                         [illegible]
        [illegible]   [illegible]                                                         [illegible]
        [illegible]   [illegible]        ERROR                                            [illegible]
        [illegible]   [illegible]                                                         [illegible]
[illegible] [illegible]  [illegible]                                                      [illegible]
[illegible] [illegible]  COMMON                                                           [illegible]
COMMON BSS   32                                                                           [illegible]
ZZZZZZ [illegible]                                                                        [illegible]
COMMON [illegible]  COMMON.ZZZZZZ                                                         [illegible]
        END                                                                               [illegible]
```

```
SIMMAP 0:NT6   LIST.REF.DECK                                          INT00010
*      SLAUZ - DOBLES INTERSATION - SIMPSONS ERROR CONTROL (F.I.E.M)  INT00020
       ENTRY  INTGP                                                   INT00030
       ENTRY  INTGS                                                   INT00040
INTGP  SAVE   1.2.4                                                   INT00050
       CLA*   4.4                  B                                  INT00060
       SXA    IR4.4                                                   INT00070
       STO    B                                                       INT00080
       LDQ*   3.4                  A                                  INT00090
       STO    A                                                       INT00100
       FSB    A                    B-.                                INT00110
       STO    IVAL                 B-A IS INTERVAL                    INT00120
       CLA-   6.4                  NUMBER OF INTERVALS ESTIMATED      INT00130
       LDQ    ZERO                                                    INT00140
       ALS    18                                                      INT00150
       STA    IRT                                                     INT00160
       ANA    MASK                 MAKES MULTIPLE OF 4                INT00170
       TRZ    *+2                                                     INT00180
       CLA    FR                   4 IS DECREMENT                     INT00190
DBL    STO    DECA                 SETS DECREMENTS AT AP              INT00200
       STO    DECB                                  AP               INT00210
       STZ    IRT                  ZERO OUT INTEGRAL AND              INT00220
       STZ    ERR                            ERROR SUMS               INT00230
       AXT    4.4                                                     INT00240
       SXA    I1.4                 PRESET IR1 AT 4                    INT00250
       PDX    0.4                  AP TO IR4                          INT00260
       TXI    *+1.4.1              UP TO AP+1                         INT00270
       SXA    I2.4                 PRESET I2 TO AP+1                  INT00280
       ANA    MASK                 MASK OFF EVERYTHING EXCEPT DECREMENTINT00290
       ARS    18                                                      INT00300
       ADD    MAS                  ADD 255    FLOAT N TO GET NUMBER OF INT00310
       FAD    MAS                  NORMALIZE  INTERVALS               INT00320
       STO    XN                   FLOATED NUMBER OF INTERVALS        INT00330
       CLA    B                                                       INT00340
IR4    AXT    *+4.4                                                   INT00350
       STO*   5.4                                                     INT00360
       RETURN INTGP                EXIT WITH X                        INT00370
INTGS  SAVE   1.2.4                                                   INT00380
       SXA    IR1.1                                                   INT00390
       SXA    IR2.2                                                   INT00400
I1     AXT    **.1                 COEFFICIENT COUNTER                INT00410
I2     AXT    **.2                 INTERVAL COUNTER                   INT00420
       LDQ    FRE                                                     INT00430
       TXL    *+3.2.1                                                 INT00440
DECA   TXH    *+2.2.**             SET AT AP                          INT00450
       LDQ    TABI+4.1                                                INT00460
       FMP*   3.4                  F                                  INT00470
       FAD    IRT                                                     INT00480
       STO    IRT                  NEW INTEGRAL SUM                   INT00490
       LDQ    ONE                                                     INT00500
       TXL    *+3.2.1                                                 INT00510
DECB   TXH    *+2.2.**             SET AT AP                          INT00520
       LDQ    TABE+4.1                                                INT00530
       FMP*   3.4                                                     INT00540
       FAD    ERR                                                     INT00550
       STO    ERR                  NEW ERROR SUM                      INT00560
       TIX    *+2.1.1              REDUCE IR1 BY 1                    INT00570
       AXT    4.1                  RESET IR1                          INT00580
       SXA    I1.1                 STORE COEFFICIENT COUNTER          INT00590
IR1    AXT    **.1                                                    INT00600
       TIX    INC+2.1              OUT TO GET NEW VALUE OF X          INT00610
       PDP    IRT                  ALL THROUGH SUMMING                INT00620
       STO    TEMPA                                                   INT00630
       STQ    TEMPQ                                                   INT00640
       LDQ    TEMPA                                                   INT00650
       CLA    TEMPQ                                                   INT00660
       SSP                                                            INT00670
       SUB*   5.4                  MAXIMUM RELATIVE ERROR             INT00680
       TMI    FIN                  OUT IF FINISHED                    INT00690
IRT    AXT    **.2                                                    INT00700
       TXI    *+1.2.1                                                 INT00710
       TXH    FIN+2.7                                                 INT00720
       SXA    IRT+2                                                   INT00730
       LXA    IR2.2                                                   INT00740
       CLA    DECA                 DOUBLE NUMBER                      INT00750
       ALS    1                       OF                              INT00760
       TPA    DBL                     INTERVALS                       INT00770
INC    SXA    I2.2                 STORE NEW INTERVAL COUNTER         INT00780
       TXI    *+1.2.-1             REDUCE TO NUMBER OF INTERVALS      INT00790
       PXA    0.2                  TO ACCUMULATOR                     INT00800
       ADD    MAS                  FLOAT                              INT00810
       FAD    MAS                    NUMBER                           INT00820
       FDP    XN                   /TOTAL NUMBER IFTERVALS            INT00830
       FMP    IVAL                                                    INT00840
       FAD    A                    NEW X                              INT00850
IR2    AXT    **.2                 RESTORE IR2                        INT00860
       TRA    IR4                                                     INT00870
FIN    CLA    INT                                                     INT00880
       FDP    XN                   IVAL/XN*H                          INT00890
       FMP    IVAL                                                    INT00900
       FDP    FRF                                                     INT00910
       STO*   4.4                  INTEGRAL                           INT00920
       STZ*   6.4                                                     INT00930
       CLA    DECA                 NUMBER OF                          INT00940
       ARS    18                        INTERVALS                     INT00950
       STA*   6.4                                                     INT00960
       LXA    IR2.2                                                   INT00970
       RETURN INTGS                                                   INT00980
MASK   OCT    077774000000                                            INT00990
FR     OCT    000604000000                                            INT01000
MAS    OCT    233000000000                                            INT01010
FRE    DEC    14.                                                     INT01020
TABI   DEC    28..64..24..64.                                         INT01030
ONE    DEC    1.                                                      INT01040
TABE   DEC    2..-4..6..-4.                                           INT01050
FRF    DEC    45.                                                     INT01060
A      BSS    1                                                       INT01070
B      BSS    1                                                       INT01080
ERR    BSS    1                                                       INT01090
INT    BSS    1                                                       INT01100
IVAL   BSS    1                                                       INT01110
XN     BSS    1                                                       INT01120
TEMPA  BSS    1                                                       INT01130
TEMPQ  BSS    1                                                       INT01140
ZERO   DEC    0                                                       INT01150
       END                                                            INT01160
```

```
$IBLAP *KDATE  SD,LIST,REF,DECK,M/94,RELMOD                          *KDA
*                                                                    *KDA  10
*      DATE -- TODAY-S DATE IN THE FORM DD MMMM YYYY                 *KDA  20
*      CALL DATE (BCDATE), WHERE BCDATE IS 2 CELLS WHICH WILL CONTAIN *KDA  30
*                          THE BCD FORM OF TODAY-S DATE              *KDA  40
*                                                                    *KDA  50
       ENTRY  DATE                                                   *KDA  60
 SYSDAT EQU   65                                                     *KDA  70
 DATE   TXH   0,,0                                                   *KDA  80
        SXA   DATE1,1                                                *KDA  90
        SXA   DATE4,4                                                *KDA 100
        LXA   DATE,1                                                 *KDA 110
        TXH   TSL,1,0                                                *KDA 120
 PXA    PXA   1,4                 TSX ENTRY                         *KDA 130
        PAC   ,1                                                     *KDA 140
        TXI   *+1,1,2                                                *KDA 150
        SXA   DATE,1              BUILD RETURN ADDRESS               *KDA 160
        TRA   *+4                                                    *KDA 170
 TSL    TXI   *+1,1,-1            TSL ENTRY                          *KDA 180
        PXA   ,1                                                     *KDA 190
        PAC   ,4                  LOAD UP XR4 FOR CALL SEQ.          *KDA 200
        CAL   3,4                                                    *KDA 210
        ADD   PAX                                                    *KDA 220
        STA   D1                                                     *KDA 230
        PXA   ,0                                                     *KDA 240
        LDQ   SYSDAT              MMDDYY                             *KDA 250
        LGL   12                  MM/DDYY00                          *KDA 260
 PAX    PAX   1,1                                                    *KDA 270
        LGL   12                  00MMDD/YY0000                      *KDA 280
        ALS   24                  DD0000/YY0000                      *KDA 290
        SLW*  3,4                 DD0000/0000YY                      *KDA 300
        LGL   12                  000 YY/000000                      *KDA 310
        ORA   D19                 0019YY/000000                      *KDA 320
 D1     SLW   **                                                     *KDA 330
        TNX   *+2,1,63                                               *KDA 340
        TXI   *+1,1,9                                                *KDA 350
        CAL   MOTABL+1,1                                             *KDA 360
        LGR   12                  0/BAAA/AB0000                      *KDA 370
        ORA*  3,4                 DD0000+BAAA IN 3,4 (INDIRECT)      *KDA 380
        SLW*  3,4                                                    *KDA 390
        LGL   36                  AB0000/000000                      *KDA 400
        ORA*  D1                  AB0000+19YY                        *KDA 410
        SLW*  D1                                                     *KDA 420
 DATE4  AXT   **,4                                                   *KDA 430
 DATE1  AXT   **,1                                                   *KDA 440
        TRA*  DATE                ESCAPE                             *KDA 450
 D19    BCI   1,001900                                               *KDA 460
        BCI   9, DEC.  NOV.  OCT.  SEPT  AUG.  JULY  JUNE  MAY  APR. *KDA 470
        BCI   2, MAR.  FEB.                                          *KDA 480
MOTABL  BCI   1, JAN.                                                *KDA 490
        END                                                          *KDA 500
```

```
$IBFTC *NT4    LIST,M94                                              INT4
      SUBROUTINE INT4(X,Y,XI,YO)                                     INT4  10
      DIMENSION X(9),Y(9),XC(4),YC(4)                                INT4  20
      EQUIVALENCE (XC(1),X1),(XC(2),X2),(XC(3),X3),(XC(4),X4),(YC(1),Y1)INT4  30
     1,(YC(2),Y2),(YC(3),Y3),(YC(4),Y4)                              INT4  40
   10 ASSIGN 90 TO NA                                                INT4  50
      J=2                                                            INT4  60
      B=XI                                                           INT4  70
   20 IF(X(J))30,40,30                                               INT4  80
   30 GO TO NA,(90,160)                                              INT4  90
   40 IF(Y(J))130,50,30                                              INT4 100
   50 IF(J-2)160,60,70                                               INT4 110
   60 YE=0.                                                          INT4 120
      GO TO 180                                                      INT4 130
   70 ASSIGN 150 TO NB                                               INT4 140
      J=J-1                                                          INT4 150
   80 X1=X(J)                                                        INT4 160
      X2=X(J-1)                                                      INT4 170
      X3=X(J-2)                                                      INT4 180
      Y1=Y(J)                                                        INT4 190
      Y2=Y(J-1)                                                      INT4 200
      Y3=Y(J-2)                                                      INT4 210
      GO TO NB,(150,170)                                             INT4 220
   90 IF(X(J)-B)120,100,100                                          INT4 230
  100 IF(J-2)130,130,110                                             INT4 240
  110 ASSIGN 160 TO NA                                               INT4 250
  120 J=J+1                                                          INT4 260
      GO TO 20                                                       INT4 270
  130 DO 140 J=1,4                                                   INT4 280
      XC(J)=X(J)                                                     INT4 290
  140 YC(J)=Y(J)                                                     INT4 300
  150 D=X2-X1                                                        INT4 310
      A1=B-X1                                                        INT4 320
      A2=B-X2                                                        INT4 330
      YE=A1*A2/2.0/D*((Y3-Y2)/(X3-X2)-(Y2-Y1)/D)-A2/D*Y1+A1/D*Y2     INT4 340
      GO TO 180                                                      INT4 350
  160 ASSIGN 170 TO NB                                               INT4 360
      GO TO 80                                                       INT4 370
  170 X4=X(J-3)                                                      INT4 380
      Y4=Y(J-3)                                                      INT4 390
      D=X3-X2                                                        INT4 400
      A1=B-X2                                                        INT4 410
      A2=B-X3                                                        INT4 420
      XM12=(Y2-Y1)/(X2-X1)                                           INT4 430
      XM23=(Y3-Y2)/D                                                 INT4 440
      XM34=(Y4-Y3)/(X4-X3)                                           INT4 450
      YE=A1*A2**2/2.0/D**2*(XM12-XM23)+A2*A1**2/2.0/D**2*(XM34-XM23)-A2*INT4 460
     1Y2/D+A1*Y3/D                                                   INT4 470
  180 YO=YE                                                          INT4 480
      RETURN                                                         INT4 490
      END                                                            INT4 500
```

```
$IBFTC NT4D*   LIST,M94                                              INT4D
      SUBROUTINE INT4D(X,Y,XI,YO,DY)                                 INT4D 10
      DIMENSION X(9),Y(9),XC(4),YC(4)                                INT4D 20
      EQUIVALENCE (XC(1),X1),(XC(2),X2),(XC(3),X3),(XC(4),X4),(YC(1),Y1)INT4D 30
     1,(YC(2),Y2),(YC(3),Y3),(YC(4),Y4)                              INT4D 40
   10 ASSIGN 90 TO NA                                                INT4D 50
      J=2                                                            INT4D 60
      B=XI                                                           INT4D 70
   20 IF(X(J))30,40,30                                               INT4D 80
   30 GO TO NA,(90,160)                                              INT4D 90
   40 IF(Y(J))130,50,30                                              INT4D100
```

```
   50 IF(J-2)60,60,70                                                   INT4D110
   60 YF=0.0                                                            INT4D120
      GO TO 180                                                         INT4D130
   70 ASSIGN 150 TO MB                                                  INT4D140
      J=J-1                                                             INT4D150
   80 X1=X(J)                                                           INT4D160
      X2=X(J-1)                                                         INT4D170
      X3=X(J-2)                                                         INT4D180
      Y1=Y(J)                                                           INT4D190
      Y2=Y(J-1)                                                         INT4D200
      Y3=Y(J-2)                                                         INT4D210
      GO TO MB,(150,170)                                                INT4D220
   90 IF(X(J)-B)120,100,100                                             INT4D230
  100 IF(J-2)130,130,110                                                INT4D240
  110 ASSIGN 160 TO MA                                                  INT4D250
  120 J=J+1                                                             INT4D260
      GO TO 20                                                          INT4D270
  130 DO 140 J=1,3                                                      INT4D280
      XC(J)=X(J)                                                        INT4D290
  140 YC(J)=Y(J)                                                        INT4D300
  150 D=X2-X1                                                           INT4D310
      A1=B-X1                                                           INT4D320
      A2=B-X2                                                           INT4D330
      XM23=(Y3-Y2)/(X3-X2)                                              INT4D340
      XM12=(Y2-Y1)/(X2-X1)                                              INT4D350
      XM2B=(XM23-XM12)/2.0/D                                            INT4D360
      YO=A1*A2*XM2B-A2*Y1/D+A1*Y2/D                                     INT4D370
      DY=XM2B*(A1+A2)+XM12                                              INT4D380
      GO TO 180                                                         INT4D390
  160 ASSIGN 170 TO MB                                                  INT4D400
      GO TO 80                                                          INT4D410
  170 X4=X(J-3)                                                         INT4D420
      Y4=Y(J-3)                                                         INT4D430
      D=X3-X2                                                           INT4D440
      A1=B-X2                                                           INT4D450
      A2=B-X3                                                           INT4D460
      XM12=(Y2-Y1)/(X2-X1)                                              INT4D470
      XM23=(Y3-Y2)/D                                                    INT4D480
      XM34=(Y4-Y3)/(X4-X3)                                              INT4D490
      AM2=A2*(XM12-XM23)                                                INT4D500
      AM1=A1*(XM34-XM23)                                                INT4D510
      YO=(A1*A2/2.0/D*(AM2+AM1)-A2*Y2+A1*Y3)/D                          INT4D520
      DY=(AM2*(2.0*A1+A2)+AM1*(2.0*A2+A1))/2.0/D**2+XM23                INT4D530
  180 RETURN                                                            INT4D540
      END                                                               INT4D550
```

```
$IBFTC PAGER   LIST,M94                                                 PAGE
      SUBROUTINE PAGE(LINES)                                            PAGE  10
C                                                                       PAGE  20
C     HEAD MOVED TO /PROLOG/ AND PAGE MODIFIED TO PRINT HEAD 25JUL 67   PAGE  30
C                                                                       PAGE  40
      COMMON  /PROLOG/  LOGIK(38), HEAD(12), SL1, SL2, EORJ             PAGE  50
      DIMENSION  TODAY (2)                                              PAGE  60
      DATA  KPG / 0 /                                                   PAGE  70
C                                                                       PAGE  80
      IF(LINES-60)20,10,10                                              PAGE  90
   10 L=2                                                               PAGE 100
      GO TO 60                                                          PAGE 110
   20 K=L+LINES                                                         PAGE 120
      IF(K-60)30,20,30                                                  PAGE 130
   30 L=K                                                               PAGE 140
   40 RETURN                                                            PAGE 150
C                                                                       PAGE 160
   50 L=LINES+2                                                         PAGE 170
   60 IF ( KPG .EQ. 0 )   CALL  DATE (TODAY)                            PAGE 180
   70 KPG = KPG +                                                       PAGE 190
      WRITE (6,80)  TODAY, HEAD, KPG                                    PAGE 200
   80 FORMAT ( 1H1 5( 6HDATE   2A6, 12X 12A6, 11X 5HPAGE  I5 )          PAGE 210
      GO TO 40                                                          PAGE 220
      END                                                               PAGE 230
```

```
$ORIGIN        ALODE
$INCLUDE       ARCDF
$IBFTC CHAMR   LIST,NOG                                                 CHAM
      SUBROUTINE  CHAMBR                                                CHAM  10
C                                                                       CHAM  20
C           20 SEP 67  MODIFIED FOR TABULAR INJECTOR COEFFICIENTS       CHAM  30
C                                                                       CHAM  40
      LOGICAL  LOGIK, ARUN,BRUN,CRUN,DRUN,ERUN,FRUN,GRUN,HRUN,IRUN,JRUN CHAM  50
      LOGICAL  SL1, SL2, EORJ                                           CHAM  60
      REAL  MACH                                                        CHAM  70
      COMMON  / /                                                       CHAM  80
     1   GAM, NWI, WIT(30), AVN(30), BVN(30), CVNR(30), CVNI(30), EE,DSCCHAM  90
      COMMON  /PROLOG/  LOGIK(38), HEAD(12), SL1, SL2, EORJ             CHAM 100
      COMMON/ARCDF/ DINP,STOW                                           CHAM 110
      DIMENSION  EXTRA(100),                          WC(75)            CHAM 120
      DIMENSION DINP(4300),A(1),B(1),C(1),D(1)                          CHAM 130
      DIMENSION  X(133),Y(133),Q(134),STOW(222),STODAT(4607)            CHAM 140
      DIMENSION ZZ(205)                                                 CHAM 150
      DIMENSION G(1),DISTL(20),DISTM(20)                                CHAM 160
      DIMENSION  AMIT(96)                                               CHAM 170
      EQUIVALENCE ( EXTRA, DINP, STODAT )                               CHAM 180
      EQUIVALENCE                                                       CHAM 190
     1   (LOGIK(1) , ARUN),    (LOGIK(2) , BRUN),    (LOGIK(3) , CRUN),CHAM 200
     2   (LOGIK(4) , DRUN),    (LOGIK(5) , ERUN),    (LOGIK(6) , FRUN),CHAM 210
     3   (LOGIK(7) , GRUN),    (LOGIK(8) , HRUN),    (LOGIK(9) , IRUN),CHAM 220
     4   (LOGIK(10), JRUN)                                              CHAM 230
      EQUIV LENCE (EXTRA(1),CA),(EXTRA(2),CB),(EXTRA(3),CC),(EXTRA(4),CHAM 240
     1CD),(EXTRA(5),CE), EXTRA(6),CF),(EXTRA(7),CG),(EXTRA(8),CH)       CHAM 250
      EQUIVALENCE (EXTRA(9),C1)                                         CHAM 260
      EQUIVALENCE (DINP(1),A),(DINP(3001),B),(DINP(3401),D),            CHAM 270
     1(DINP(3501),G),(DINP(3801),C),(EXTRA(21),WC) , (B(4),UE)          CHAM 280
      EQUIVALENCE  (DINP(601),AMIT),(DINP(2013),Q),(DINP(3501),X),      CHAM 290
     1       (DINP(3651)),Y)                                            CHAM 300
      EQUIVALENCE                                                       CHAM 310
     1  (STOW(1),ZZ), (STOW(213),YM)                                    CHAM 320
     2  ,  (STOW(214),J), (STOW(215),NE), (STOW(216),YL), (STOW(217),K1)CHAM 330
     3  ,  (STOW(218),N), (STOW(219),KER), (STOW(220),XM)               CHAM 340
     4  ,  (STOW(221),KQUAD), (STOW(222),XL)                            CHAM 350
     5  ,   (EXTRA(51), DISTL), (EXTRA(71), DISTM )                     CHAM 360
     6  ,  (EXTRA(10),CJ), (EXTRA(21),SNH ), (EXTRA(12), MACH )         CHAM 370
     7  ,   (DINP(107),UIBAR)                                           CHAM 380
C                                                                       CHAM 390
C*********************************************************************** CHAM 400
C                                                                       CHAM 410
   10 FORMAT(1H0,60H      A      B      C      D      E      F      G      H CHAM 420
     1 I      J      // 3X,(10F6.0))                                   CHAM 430
   20 FORMAT(////,9X,108H************************* THE FOLLOWING MAIN CONTCHAM 440
     1ROL DATA WILL BE USED IN THIS CASE ************************  ,///,CHAM 450
     245X,33HRATIO OF SPECIFIC HEAT (GAMMA) = ,F7.4,///,45X,22HDESIRED MACHAM 460
     3CH NUMBER = ,E12.5,20H (=0 IF BEING CALCULATED) //45X        17HCHCHAM 470
     5AMBER RADIUS = ,F7.3, 9H (INCHES),///,45X,17HCHAMBER LENGTH = ,F7.3CHAM 480
     6, 9H (INCHES),///,45X, 17HSPEED OF SOUND = ,F10.3,  9H (FT/SEC),///,CHAM 490
     745X, 27HCHAMBER MODE DESCRIPTION = ,F8.5, 20H (=0 FOR LONGITUDINALCHAM 500
     8 MODES),/,)                                                       CHAM 510
   30 FORMAT(////,33X,48H********** CHAMBER FREQUENCIES (WC) **********  CHAM 520
     1   ,///,)                                                         CHAM 530
   40 FORMAT(5F20.5,/ )                                                 CHAM 540
   50 FORMAT(30X, 64H*****J MACH DISTRIBUTION IN CHAMBER AS A FUNCTION OFCHAM 550
     1 LENGTH *****,//,11X,7HCHAMBER,15X,4HMACH,14X,7HCHAMBER,15X,4HMACHCHAM 560
     2,14X,7HCHAMBER,15X,4HMACH,/,12X,6HLENGTH,11X,12HDISTRIBUTION,11X, CHAM 570
     36HLENGTH,11X,12HDISTRIBUTION,11X,6HLENGTH,11X,12HDISTRIBUTION,/,) CHAM 580
   60 FORMAT(6(10X,F10.5))                                              CHAM 590
C*********************************************************************** CHAM 600
C*********************************************************************** CHAM 610
C                                                                       CHAM 620
      IF ( SL2 )              GO TO 170                                 CHAM 630
C                                                                       CHAM 640
      SL2 = .TRUE.                                                      CHAM 650
      DO 70 I = 1, 4300                                                 CHAM 660
   70    DINP (I) = 0.0                                                 CHAM 670
      GO TO 110                                                         CHAM 680
C                                                                       CHAM 690
C*********************************************************************** CHAM 700
C                                                                       CHAM 710
   80 WRITE (6,100)   NE                                                CHAM 720
   90 CALL EXIT                                                         CHAM 730
  100 FORMAT (1H010X17HINPUT ERROR, NE = I3, 19H, HENCE TERMINATION )   CHAM 740
C                                                                       CHAM 750
C*********************************************************************** CHAM 760
C*********************************************************************** CHAM 770
C                                                                       CHAM 780
  110 KER = 0                                                           CHAM 790
      CALL  DVCHK  (KCHK)                                               CHAM 800
C                                                                       CHAM 810
      DO 120 I = 1, 10                                                  CHAM 820
  120    DINP(I) = 0.0                                                  CHAM 830
      CALL  A5130  ( DINP(1), HEAD(1), NE )                             CHAM 840
      IF ( NE .NE. 1 )   GO TO 80                                       CHAM 850
      ARUN = CA .NE. 0.0                                                CHAM 860
      BRUN = CB .NE. 0.0                                                CHAM 870
      CRUN = CC .NE. 0.0                                                CHAM 880
      DRUN = CD .NE. 0.0                                                CHAM 890
      ERUN = CE .NE. 0.0                                                CHAM 900
      FRUN = CF .NE. 0.0                                                CHAM 910
      IRUN = CI .NE. 0.0                                                CHAM 920
      JRUN = CJ .NE. 0.0                                                CHAM 930
     GGRUN =        DINP(22) .LE. 0.0                                  CHAM 940
     1        .AND. ( ARUN .OR. BRUN .OR. CRUN .OR. IRUN )             CHAM 950
C                                                                       CHAM 960
C     PRINT NEW MAIN CONTROL DATA                                       CHAM 970
C*********************************************************************** CHAM 980
      CALL  PAGE  ( 60 )                                                CHAM 990
      WRITE (6,10)  (DINP(I), I=1,10)                                   CHAM1000
      WRITE (6,20)  DINP(11), MACH, DINP(14), DINP(15), DINP(16), SNH   CHAM1010
C*********************************************************************** CHAM1020
C     ARE FREQUENCIES TO BE CALCULATED... IF SO, CALCULATE AND PRINT.   CHAM1030
C*********************************************************************** CHAM1040
      IF ( .NOT. GRUN )        GO TO 130                                CHAM1050
      CALL  GENMEG  ( WC(1)                                             CHAM1060
      IF ( BRUN .OR. ARUN )             CRUN = .TRUE.                   CHAM1070
      IF ( CRUN )     CC = CC + 11.0                                    CHAM1080
  130 JOMEGA=ABS(EXTRA(22))+22.0001                                     CHAM1090
      WRITE (6,30)                                                      CHAM1100
      WRITE (6,40)(EXTRA(I),I=23,JOMEGA)                                CHAM1110
      WRITE (6,50)                                                      CHAM1120
      WRITE (6,60)( DISTL(I), DISTM(I), DISTL(I+7),           DISTM(I+7)CHAM1130
     1, DISTL(I+14), DISTM(I+14), I=1,6 ), DISTL(7),          DISTM(7), CHAM1140
     2DISTL(16), DISTM(16)                                              CHAM1150
C*********************************************************************** CHAM1160
C                                                                       CHAM1170
      EORJ = ERUN .OR. JRUN .OR. IRUN                                   CHAM1180
```

```
      IF ( CRUN .OR. EORJ )   CALL  PAGE  ( 60 )                        CHAN1190
  140 IF ( .NOT. EORJ )          GO TO 200                              CHAN1200
      WRITE (14)       STODAT                                           CHAN1210
      BACKSPACE  14                                                     CHAN1220
      EE  = CE                                                          CHAN1230
      DSC = CI                                                          CHAN1240
      GAM = DINP(11)                                                    CHAN1250
      IF ( .NOT. IRUN )       GO TO 160                                 CHAN1260
      NWI = ( JOMEGA-22 )/2 +1                                          CHAN1270
      DO 150 I = 1, NWI                                                 CHAN1280
  150    WIT(I) = DINP( 2*I+21 )                                        CHAN1290
      WIT(NWI) = DINP( JOMEGA )                                         CHAN1300
C                                                                       CHAN1310
  160 RETURN                                                            CHAN1320
C     ******                                                            CHAN1330
C          PROGRAM RETURNS IF AND ONLY IF INJECTOR PROGRAM IS TO BE RUN CHAN1340
C                                                                       CHAN1350
C******************************************************************* CHAN1360
C******************************************************************* CHAN1370
C                                                                       CHAN1380
C                                                                       CHAN1390
  170 READ (14)  STODAT                                                 CHAN1400
      BACKSPACE  14                                                     CHAN1410
      IF ( .NOT. ERUN )       GO TO 200                                 CHAN1420
C                                                                       CHAN1430
      DINP(3400) = NWI                                                  CHAN1440
      DO 180 I = 1, NWI                                                 CHAN1450
         DINP(I+4522) = WIT(I)                                          CHAN1460
         DINP(I+4539) = AVR (I)                                         CHAN1470
         DINP(I+4556) = BVR (I)                                         CHAN1480
         DINP(I+4573) = CVR(I)                                          CHAN1490
  180    DINP(I+4590) = CVRI(I)                                         CHAN1500
      DO 190 I = NWI, 84, 17                                            CHAN1510
  190    DINP(I+4529) = 0.0                                             CHAN1520
C                                                                       CHAN1530
C******************************************************************* CHAN1540
C                                                                       CHAN1550
  200 WC(2) = ABS( WC(2) )                                              CHAN1560
C******************************************************************* CHAN1570
C     FIX NUMBER OF FREQUENCIES AND TEST DESIRE FOR INTERNAL C FLAG     CHAN1580
C******************************************************************* CHAN1590
  210 NOMEG=WC(2)+.0001                                                 CHAN1600
      IF(EXTRA(12))230,220,230                                          CHAN1610
  220 DINP(3002)=3.0                                                    CHAN1620
      GO TO 240                                                         CHAN1630
  230 U1BAR = MACH                                                      CHAN1640
      UE = MACH                                                         CHAN1650
  240 IF ( .NOT. IRUN )              GO TO 390                          CHAN1660
C******************************************************************* CHAN1670
C     SET UP DATA FOR PROGRAM  C.                                       CHAN1680
C******************************************************************* CHAN1690
  250 DINP(3001)=EXTRA(11)                                              CHAN1700
      NNMAP = NOMEG/2 + 1                                               CHAN1710
      ZZ(1)=EXTRA(14)                                                   CHAN1720
      ZZ(2)=EXTRA(15)                                                   CHAN1730
      ZZ(3)=EXTRA(16)                                                   CHAN1740
       DINP(3005) = EXTRA(14)                                           CHAN1750
      DINP(3003)=NNMAP                                                  CHAN1760
      IF ( DINP(3009) .EQ. 0.0 )        DINP(3009) = 101.0              CHAN1770
      K = 0                                                             CHAN1780
      ZAVE=((EXTRA(11)+1.0)/2.0)*SQRT(DINP(3006)*DINP(3007))            CHAN1790
      IF(EXTRA(21))270,260,270                                          CHAN1800
  260 SMOZ = 0.0                                                        CHAN1810
C******************************************************************* CHAN1820
C     USING HALF OF CHAMBER FREQUENCIES, CALCULATE NOZZLE FREQUENCIES   CHAN1830
C       FOR USE IN PROG C.                                              CHAN1840
C******************************************************************* CHAN1850
      RORL = EXTRA(15)                                                  CHAN1860
      GO TO 280                                                         CHAN1870
  270 RORL = EXTRA(14)                                                  CHAN1880
      GRAD = SQRT((2.0/(EXTRA(11)+1.0))*(DINP(3006)/DINP(3007)))        CHAN1890
      SMOZ = SRM/GRAD                                                   CHAN1900
  280 DO 290 I=1,NNMAP                                                  CHAN1910
      KN=(2*I)+1                                                        CHAN1920
      NN=6210+K                                                         CHAN1930
      DINP(NN) = ZAVE*WC(KN)/RORL                                       CHAN1940
      DINP(NN+1) = SMOZ                                                 CHAN1950
      DINP(NN+2) = MACH                                                 CHAN1960
      K=K+3                                                             CHAN1970
  290 CONTINUE                                                          CHAN1980
      DINP(NN)=ZAVE*(WC(NOMEG+2)/EXTRA(14))                             CHAN1990
  300 CALL CCC(C(1),ZZ(1),WC(1),CC,KER)                                 CHAN2000
C     SET UP NOZZ ADM FOR A,B                                           CHAN2010
      IF(KER) 310,330,310                                               CHAN2020
  310 WRITE (6,320)                                                     CHAN2030
  320 FORMAT (1H0 30X39H ERROR PROGRAM C.  ALL CASES TERMINATED )       CHAN2040
      CALL CORE(C(1),429,CC)                                            CHAN2050
      GO TO 90                                                          CHAN2060
  330 IF ( .NOT. ARUN )          GO TO 360                              CHAN2070
C******************************************************************* CHAN2080
C     MOVE OUTPUT FROM C INTO INPUT BLOCK TO A                          CHAN2090
C******************************************************************* CHAN2100
  340 NP = C(5)                                                         CHAN2110
      IWO = 0                                                           CHAN2120
      DO 350 I = 1, NP                                                  CHAN2130
         IWO = IWO + 2                                                  CHAN2140
         AMIT(I    ) = ZZ(IWO     )                                     CHAN2150
         AMIT(I+30) = ZZ(IWO+  1)                                       CHAN2160
         AMIT(I+60) = ZZ(IWO+101)                                       CHAN2170
  350    CONTINUE                                                       CHAN2180
C                    ZEROES TO END TABLES                               CHAN2190
      AMIT(NP+ 1) = 0.0                                                 CHAN2200
      AMIT(NP+31) = 0.0                                                 CHAN2210
      AMIT(NP+61) = 0.0                                                 CHAN2220
      IF ( MACH .LE. 0.0 )   U1BAR = ZZ(205)                            CHAN2230
      GO TO 410                                                         CHAN2240
C                                                                       CHAN2250
  360 IF ( .NOT. BRUN .AND. SRM .EQ. 0.0 )      GO TO 340               CHAN2260
C******************************************************************* CHAN2270
C     MOVE OUTPUT FROM PROG C INTO INPUT BLOCK FOR PROG B.              CHAN2280
C******************************************************************* CHAN2290
  370 DO 380 I=18,200                                                   CHAN2300
  380 B(I)=ZZ(I)                                                        CHAN2310
      B(4)=ZZ(205)                                                      CHAN2320
C                                                                       CHAN2330
  390 CONTINUE                                                          CHAN2340
      HRUN = (CH.GT. 0.0) .OR. BRUN .AND.(UE.GE.0.1) .AND.(CH.EQ.0.0)   CHAN2350
C     NOZZLE ADMITTANCE IS INPUT TO PROGS A, B.                         CHAN2360
C******************************************************************* CHAN2370
C                                                                       CHAN2380
  400 IF ( .NOT. ARUN )            GO TO 490                            CHAN2390
```

```
C                                                                       CHAM2400
  410 CALL  LONGL                                                       CHAM2410
C                                                                       CHAM2420
      IF(KER)420,440,420                                                CHAM2430
  420 WRITE (6,430)                                                     CHAM2440
  430 FORMAT(1H0,60X,18H ERROR PROGRAM A  )                             CHAM2450
      CALL  CORE  ( A(1), 700, CA )                                     CHAM2460
  440 CONTINUE                                                          CHAM2470
      GO TO 110                                                         CHAM2480
C*********************************************************************** CHAM2490
C     SET UP MAIN CONTROL DATA FOR USE IN B                             CHAM2500
C*********************************************************************** CHAM2510
  450 IF ( .NOT. BRUN )         GO TO 530                               CHAM2520
      B(1)=WC(1)                                                        CHAM2530
      B(2)=EXTRA(15)/EXTRA(14)                                          CHAM2540
      B(3)=EXTRA(11)                                                    CHAM2550
      B(5)=EXTRA(16)                                                    CHAM2560
      B(6) = DINP(17)                                                   CHAM2570
      B(7) = DINP(18)                                                   CHAM2580
      B(8) = DINP(19)                                                   CHAM2590
      B(169)=WC(2)                                                      CHAM2600
      DO 460 I=1,NOMEG                                                  CHAM2610
      B(I+169)=WC(I+2)                                                  CHAM2620
  460 CONTINUE                                                          CHAM2630
      B(200) = DISTL(1) / DINP(14)                                      CHAM2640
      B(220) = DISTM(1)                                                 CHAM2650
      B(17) = 1.0                                                       CHAM2660
      DO 480 I = 2, 20                                                  CHAM2670
      IF(DISTM(I))490,490,470                                           CHAM2680
  470 B(I+199)=DISTL(I)/EXTRA(14)                                       CHAM2690
      B(I+219)=DISTM(I)                                                 CHAM2700
      B(17)=B(17)+1.0                                                   CHAM2710
  480 CONTINUE                                                          CHAM2720
  490 CONTINUE                                                          CHAM2730
      CALL  TRANS  ( B, X, CB, KER )                                    CHAM2740
      DO 500 I = 9, 100                                                 CHAM2750
  500     D(I) = X(I)                                                   CHAM2760
      IF(KER)530,510,530                                                CHAM2770
  510 WRITE (6,520)                                                     CHAM2780
  520 FORMAT (1H0,60X,18H ERROR PROGRAM B  )                            CHAM2790
      CALL CORE(B(1),400,CB)                                            CHAM2800
      GO TO 110                                                         CHAM2810
C*********************************************************************** CHAM2820
  530 IF ( .NOT. BRUN )         GO TO 620                               CHAM2830
  540 DO 550 I=1,9                                                      CHAM2840
      X(I)=D(I)                                                         CHAM2850
  550 CONTINUE                                                          CHAM2860
  560 X(9)=EXTRA(11)                                                    CHAM2870
      DO 570 I = 9, 100                                                 CHAM2880
  570     X(I) = D(I)                                                   CHAM2890
      CALL  DDO  ( X(1), V(1), CD, KER, ERR )                           CHAM2900
      IF ( ERR )                    110,580,580                         CHAM2910
  580 IF ( KER )                    610,590,610                         CHAM2920
  590 WRITE (6,600)                                                     CHAM2930
  600 FORMAT (1H0,60X,18H ERROR PROGRAM D  )                            CHAM2940
      CALL CORE(X(1),100,CD)                                            CHAM2950
      GO TO 110                                                         CHAM2960
  610 CONTINUE                                                          CHAM2970
  620 IF ( .NOT. FRUN )         GO TO 670                               CHAM2980
C*********************************************************************** CHAM2990
C     SET UP DATA FOR PROG F    (W, TAU )                               CHAM3000
C*********************************************************************** CHAM3010
  630 V(1)=EXTRA(16)                                                    CHAM3020
      G(101) = EXTRA(16)/V(1) * 12.0/6.2831853                          CHAM3030
  640 V(2)=EXTRA(16)                                                    CHAM3040
      CALL  FFF  ( V(1), G(1), CF, KER, WC(1) )                         CHAM3050
      IF ( KER )                    670,650,670                         CHAM3060
  650 WRITE (6,660)                                                     CHAM3070
  660 FORMAT (1H0,60X,18H ERROR PROGRAM F  )                            CHAM3080
      CALL CORE(V(1),100,CF)                                            CHAM3090
  670 GO TO 110                                                         CHAM3100
C                                                                       CHAM3110
C                                                                       CHAM3120
C*********************************************************************** CHAM3130
C*********************************************************************** CHAM3140
C*********************************************************************** CHAM3150
      END                                                               CHAM3160
```

```
$IBFTC WGEN    LIST,MAP                                                 WGEN
      SUBROUTINE GENWEG(W)                                              WGEN  10
C                                                                       WGEN  20
C*********************************************************************** WGEN  30
      DIMENSION W(1)                                                    WGEN  40
C                                                                       WGEN  50
C   LONGITUDINAL WILL HAVE NEG SAW WHICH WILL BE SET TO ZERO BEFOR RETURWGEN  60
C  IF SAW 0 LONGITUDINAL AND GENERATE + OR - 10 PERCENT OF PIE          WGEN  70
C*********************************************************************** WGEN  80
      IF(W(1))40,50,10                                                  WGEN  90
C*********************************************************************** WGEN 100
C   FOR TRANSVERSE GENERATE 10 VALUES AROUND SAW 9 PERCEN. BELOW AND    WGEN 110
C   11 PERCENT ABOVE                                                    WGEN 120
C*********************************************************************** WGEN 130
   10 SAW=W(1)                                                          WGEN 140
   20 DELWEG=.1*SAW                                                     WGEN 150
      W(9)=SAW-DELWEG                                                   WGEN 160
      DELWEG=DELWEG/9.0                                                 WGEN 170
      W(9)=W(9)+DELWEG                                                  WGEN 180
      DO 30 I=6,12                                                      WGEN 190
      W(I)=W(I-1)+DELWEG                                                WGEN 200
   30 CONTINUE                                                          WGEN 210
C*********************************************************************** WGEN 220
C  NEGITIVE 10 INDICATES TO PROGRAM THAT FREQUENCES ARE GENERATED INTERWGEN 230
C*********************************************************************** WGEN 240
      W(2)=-10.0                                                        WGEN 250
      RETURN                                                            WGEN 260
C*********************************************************************** WGEN 270
C   IF NEG GENERATE + OR - 10 PERCENT OF POSITIVE INITIAL GUESS         WGEN 280
C*********************************************************************** WGEN 290
   40 SAW=-W(1)                                                         WGEN 300
      W(1)=0.0                                                          WGEN 310
      GO TO 20                                                          WGEN 320
   50 SAW=3.141592                                                      WGEN 330
      GO TO 20                                                          WGEN 340
      END                                                               WGEN 350
```

```
$ORIGIN        BLCOE                                                    HYMN

$IBFTC HYMN    LIST,N94                                                 HYMN  10
      SUBROUTINE  TRANS  ( BIN, XOUT, CB, KER )                         HYMN  20
C                                                                       HYMN  30
C     CALCULATES COMBUSTION PARAMETERS  HR, HI                          HYMN  40
C        PROGRAM BY  LW VERNON  FROM  11MAY67 ANALYSIS OF  AJ SMITH, JR HYMN  50
C           INCORPORATES CORRECTIONS TO ANALYSIS THRU  1 JUN 67         HYMN  60
C           BOOLE INTEGRATION AS OF  12 JUNE 67                         HYMN  70
C           RHO, RHOL CONVENTIONS CORRECTED  22JUN 67                   HYMN  80
C           20 SEP 67  MODIFIED FOR TABULAR INJECTOR COEFFICIENTS       HYMN  90
C                                                                       HYMN 100
C*********************************************************************** HYMN 110
C                                                                       HYMN 120
      REAL  IAR, IAI, IBR, IBI                                          HYMN 130
      LOGICAL  LOGIK, HRUN, CKOUT, SIMPL, KNOT, LIMIT, TABLR            HYMN 140
C                                                                       HYMN 150
      COMMON  /PROLOS/  LOGIK(30), MEAD(12), SL1, SL2, ECRJ             HYMN 160
      COMMON  /ABCDF / DIN (4500)                                       HYMN 170
     1    , SPACE(222), WIT(17), AVN(17), BVN(17), CVNR(17), CVNI(17)   HYMN 180
      DIMENSION          B    ( 240 ) ,  GS    ( 6, 102 ) ,             HYMN 190
     1   ZZ    ( 102 ) ,  STAB ( 102 ) ,  GL    ( 12, 3 )   ,            HYMN 200
     2   U     ( 102 ) ,  CTAB ( 102 ) ,  SLAM  ( 12 )      ,            HYMN 210
     3   DU    ( 102 ) ,  D2U  ( 102 ) ,  ZAP   ( 12 )      ,            HYMN 220
     4   UL    ( 102 ) ,  DUL  ( 102 ) ,  G     ( 12 )      ,            HYMN 230
     5   RHO   ( 102 ) ,  DRHO ( 102 ) ,  GLD   ( 12 )      ,            HYMN 240
     6   RHOL  ( 102 ) ,  DRGL ( 102 ) ,  ZING  (  6 )      ,            HYMN 250
     7   QBAR  ( 102 ) ,  DQ   ( 102 ) ,  SIMP  (  6 )      ,            HYMN 260
     8   ZIP3  ( 102 ) ,  V1   ( 102 ) ,  ERRZ  (  6 )      ,            HYMN 270
     9   ZIP5  ( 102 ) ,  V2   ( 102 ) ,  ZP    (  6 )                  HYMN 280
      DIMENSION                                                         HYMN 290
     1   THR   (  30 ) ,  THI  (  30 ) ,  THTR (  30 ) ,  THTI (  30 )  HYMN 300
     ODIMENSION           AL   (  6 ) ,  DFEK (  4 )      ,             HYMN 310
     1   BIN   ( 240 ) ,  XOUT ( 100 ) ,  WC    ( 28 )      ,            HYMN 320
     2   WET   (  30 ) ,  ERT  (  30 ) ,  EIT   ( 30 )      ,            HYMN 330
     3   ZDIST(  20 ) ,  CRT  (  30 ) ,  CIT   ( 30 )      ,            HYMN 340
     4   DISTN(  20 )                                                   HYMN 350
      EQUIVALENCE                                                       HYMN 360
     1   ( HRUN , LOGIK( 8)),( CKOUT, LOGIK(13)),( SIMPL, LOGIK(24)),   HYMN 370
     2   ( KNOT , LOGIK(15)),( LIMIT, LOGIK(16)),( TABLR, LOGIK(17))    HYMN 380
      EQUIVALENCE                                                       HYMN 390
     1   ( B(1) , SNH    )),( B( 10) , ZINC   )),( DIN( 14) , RCH    )), HYMN 400
     2   ( B(2) , ZE     )),( B( 18) , XNE    )),( UL(1)    , ULO    )), HYMN 410
     3   ( B(3) , GAM    )),( B( 19) , XNW    )),( RHOL(1)  , RHOLO  )), HYMN 420
     4   ( B(4) , UE     )),( B(170) , WC     )),( B(20)    , WET    )), HYMN 430
     5   ( B(5) , SOUND  )),( B(200) , ZDIST  )),( B(50)    , ERT    )), HYMN 440
     6   ( B(6) , ULN    )),( B(220) , DISTN  )),( B(80)    , EIT    )), HYMN 450
     7   ( B(7) , XK     )),( B(110) , CRT    )),                       HYMN 460
     8   ( B(8) , XCMPL  )),( B(140) , CIT    )                         HYMN 470
      EQUIVALENCE                                                       HYMN 480
     1   ( DIN(3406), RLON )),( DIN(3407), TLON )                       HYMN 490
      EQUIVALENCE                                                       HYMN 500
     1   ( A1R  , AL   ( 1) )),( A2R  , AL   ( 3) )),( A4R  , AL   ( 5) )), HYMN 510
     2   ( A1I  , AL   ( 2) )),( A2I  , AL   ( 4) )),( A4I  , AL   ( 6) )), HYMN 520
     3   ( G20R , SLAM( 9) )),( DG20R , SLAM(10) )),                    HYMN 530
     4   ( G20I , SLAM(11) )),( DG20I , SLAM(12) )                      HYMN 540
C*********************************************************************** HYMN 550
C                                                                       HYMN 560
   10 FORMAT ( / 65H TRANSVERSE STABILITY PROGRAM... CALCULATES  HR, HI HYMN 570
     1 OR  HTR, HTI )                                                   HYMN 580
   20 FORMAT ( // 11H INPUT DATA// 9X9HSNH12X2HZE10X5HGAMMA9X2HUE9X     HYMN 590
     1   14HSOUND (FT/SEC) 2X13H ULN (FT/SEC) 5X 9HXK  (DRAG)4X11HRCOMPL HYMN 600
     2(IN)  //   F15.8, F14.8, F12.5, F14.8, F16.2, 2X 3F15.6   //     HYMN 610
     3 5X10HINCREMENTS 8X4HLR/N 11X4HLT/N / 5XF8.1, 2F19.7 // )         HYMN 620
   30 FORMAT ( // 29H NOZZLE ADMITTANCES INPUT//  19X9HOMEGA(CM) 12X    HYMN 630
     1   5HERT 17X5HEIT 17X5HCRT 17X5HCIT // )                          HYMN 640
   40 FORMAT ( ////22H CALCULATED RESULTS... 10X 29HFIRST-ORDER SOLN ( UHYMN 650
     1NIF INJ ) 15X21HSECOND-ORDER SOLUTION // 17X5HOMEGA 12X6HN REAL   HYMN 660
     2 14X6HN IMAG 13X7HHT REAL 13X7HHT IMAG  )                         HYMN 670
   50 FORMAT ( 9XF15.7,2E20.8 )                                         HYMN 680
   60 FORMAT ( 9XF15.7,4E20.8 )                                         HYMN 690
   70 FORMAT ( // 31H INJECTOR COEFFICIENTS INPUT... // )               HYMN 700
   80 FORMAT ( 11X3HARM ,7X3HBNM 16X6HCNM RE 14X6HCNM IM / 4E20.8 )     HYMN 710
   90 FORMAT (10X5HOMEGA16X3HANM17X3HBNM16X6HCNM RE14X6HCNM IM/(5E20.8)) HYMN 720
  100 FORMAT ( 12H0INTEGRALS   6E18.8 )                                 HYMN 730
  110 FORMAT ( 6H ERROR 6X6E18.8 )                                      HYMN 740
  120 FORMAT ( 29H0DUMP OF TABULAR FUNCTIONS... // 6X2HZZ12X1HU11X2HDU  HYMN 750
     1   11X2HUL11X3HRHO9X4HRHOL9X4HQBAR9X4HZIP39X4HZIP5 // (1H 9E13.5)) HYMN 760
  130 FORMAT ( 18H0TABLES FOR HIMACH//6X2HZZ11X5HD2U11X5HDUL9X4HDRHO9X  HYMN 770
     1 4HDROL10X2HDQ11X2HV111X2HV2//)                                   HYMN 780
  140 FORMAT ( 1H 8E13.5 )                                              HYMN 790
  150 FORMAT (/10X1HW16X5HIAR15X5HIAI15X5HIBR15X5HIBI14X5HERCO X13X5HEICOHYMN 800
     1N/8X4HG20R14X4HG21R14X5HDG20R13X5HDG21R14X5HN1R15X5HN2R15X2HCR    HYMN 810
     2 /8X4HG20I14X4HG21I14X5HDG20I13X5HDG21I14X5HN1I15X5HN2I15X2HCI//) HYMN 820
  160 FORMAT ( 7E18.8 )                                                 HYMN 830
C                                                                       HYMN 840
C*********************************************************************** HYMN 850
C*********************************************************************** HYMN 860
      HRUN = DIN(8) .GE. 0.0                                            HYMN 870
      CALL  DVCHK  ( KCHK )                                             HYMN 880
      NWI = DIN(3408) + 0.01                                            HYMN 890
      TABLR = NWI .GE. 9                                                HYMN 900
      DO 170 I = 1, 240                                                 HYMN 910
  170    B(I) = BIN(I)                                                  HYMN 920
  180 IF ( CB-10.0 )            220,190,190                             HYMN 930
  190 CALL  PAGE  (  70 )                                               HYMN 940
      KOUT = 2                                                          HYMN 950
      NET = XNE                                                         HYMN 960
      WRITE (6,10)                                                      HYMN 970
      IF ( CB-100.0 )           220,200,200                             HYMN 980
  200 WRITE (6,20)  ( B(I), I = 1, 8 ),  ZINC, RLON, TLON               HYMN 990
      WRITE (6,30)                                                      HYMN1000
      WRITE (6,40)  ( WET(I), ERT(I),EIT(I), CRT(I),CIT(I), I=1,NET )   HYMN1010
      CKOUT = ( CB .GT. 200.0 )                                         HYMN1020
      IF ( .NOT. HRUN )       GO TO 220                                 HYMN1030
      WRITE (6,70)                                                      HYMN1040
      IF ( TABLR )            GO TO 210                                 HYMN1050
      WRITE (6,80)     AVN(1), BVN(1), CVNR(1), CVNI(1)                 HYMN1060
      GO TO 220                                                         HYMN1070
  210 WRITE (6,90)  ( WIT(I), AVN(I), BVN(I), CVNR(I), CVNI(I),         HYMN1080
     1                I = 1, NWI  )                                     HYMN1090
C                                                                       HYMN1100
C     GENERATE NORMALIZED DISTRIBUTION AND OTHER TABULAR FUNCTIONS      HYMN1110
C                                                                       HYMN1120
  220 ZC  = XCMPL / RCH                                                 HYMN1130
      ZC = ZE                                                           HYMN1140
      IDZ = ZINC + 0.001                                                HYMN1150
      CALL  INT4  ( ZDIST, DISTN, ZC, UENRM )                           HYMN1160
      SCALE = UE / UENRM                                                HYMN1170
      SIMPL = IDZ .LE. 0                                                HYMN1180
      IF ( SIMPL )            GO TO 290                                 HYMN1190
```

```
      INT = 2                                                           HYMN1200
      IDZ = IDZ/4 * 4                                                   HYMN1210
C            MUST BE POSITIVE MULTIPLE OF FOUR.  LESS THAN  101         HYMN1220
      GO TO 240                                                         HYMN1230
C                                                                       HYMN1240
  230 INT = 1                                                           HYMN1250
      IDZ = -IDZ/2 * 2                                                  HYMN1260
C                                                                       HYMN1270
  240 IF ( ( IDZ .EQ. 0 ) .OR. ( IDZ .GT. 100 ) )  IDZ = 80             HYMN1280
C                                                                       HYMN1290
C          IDZ  IS NUMBER OF Z-INCREMENTS.                              HYMN1300
C                                                                       HYMN1310
      IDZP = IDZ + 1                                                    HYMN1320
      DZ  = ZC / FLOAT(IDZ)                                             HYMN1330
C                                                                       HYMN1340
      ZZ(1) = 0.0                                                       HYMN1350
      U(1)  = 1.0E-10                                                   HYMN1360
      DU(1) = 0.0                                                       HYMN1370
      RHO(1)= 1.0                                                       HYMN1380
      ULO   = ULM/SCUND                                                 HYMN1390
      GF1 = -1.0 / (GAM-1.0)                                            HYMN1400
      GF2 = (GAM-1.0) / 2.0                                             HYMN1410
      RHOZE = ( 1.0 + GF2*UE*UE ) **GF1                                 HYMN1420
      RHOLO = RHOZE * UE / ULO                                          HYMN1430
      OBAR(1) = 0.0                                                     HYMN1440
      ZIP3(1) = 0.0                                                     HYMN1450
      ZIP5(1) = RHOLO                                                   HYMN1460
C                     ABOVE ARE FIRST TABULAR ENTRIES.                  HYMN1470
C                                                                       HYMN1480
      Z = 0.0                                                           HYMN1490
C                                                                       HYMN1500
      DO 250 IZ = 2, IDZP                                               HYMN1510
         Z = Z + DZ                                                     HYMN1520
         CALL  INT4D  ( ZDIST, DISTW, Z, U(IZ), DU(IZ) )                HYMN1530
         ZZ(IZ) = Z                                                     HYMN1540
         U (IZ) =  U(IZ)*SCALE                                          HYMN1550
         DU(IZ) = DU(IZ)*SCALE                                          HYMN1560
         UL(IZ) = UL(IZ-1) + XK*DZ*( U(IZ-1)-UL(IZ-1) )/UL(IZ-1)        HYMN1570
         TEMP = 1.0 + GAM*( U(IZ)-UL(IZ) )*U(IZ)                        HYMN1580
         RHO(IZ) = ( 1.0 + GF2*U(IZ)*U(IZ) ) **GF1                      HYMN1590
         RHOL(IZ) = ( RHOZE*UE - RHO(IZ)*U(IZ) ) / UL(IZ)               HYMN1600
     0   OBAR(IZ) = ( ( 1.0 - GAM*U(IZ)*U(IZ) )*RHO(IZ)*DU(IZ)          HYMN1610
     1              - GAM*U(IZ)*RHOL(IZ)*XK*( U(IZ)-UL(IZ) ) ) / TEMP   HYMN1620
         ZIP3(IZ) = DU(IZ) + DU(IZ)                                     HYMN1630
         ZIP5(IZ) = RHOL(IZ) / RHO(IZ)                                  HYMN1640
  250    CONTINUE                                                       HYMN1650
      ZZ(IDZP) = ZC                                                     HYMN1660
      U (IDZP) = UE                                                     HYMN1670
      DU(IDZP) = 0.0                                                    HYMN1680
      RHO(IDZP) = RHOZE                                                 HYMN1690
      RHOL(IDZP) = 0.0                                                  HYMN1700
      OBAR(IDZP) = 0.0                                                  HYMN1710
      ZIP3(IDZP) = 0.0                                                  HYMN1720
      ZIP5(IDZP) = 0.0                                                  HYMN1730
C                                                                       HYMN1740
C     OBAR, ZIP3, ZIP5  AS TABULATED ABOVE ARE FREQ-INDEPENDENT PARTS   HYMN1750
C        OF INTEGRANDS.                                                 HYMN1760
C                                                                       HYMN1770
C*********************************************************************** HYMN1780
      IF ( .NOT. HRUN )        GO TO 270                                HYMN1790
C                                                                       HYMN1800
C          INITIAL VALUES FOR  HI-ORDER TABLES                          HYMN1810
      HDZ = 0.5 * DZ                                                    HYMN1820
      D2U (1) = 0.0                                                     HYMN1830
      DRHO(1) = 0.0                                                     HYMN1840
      DROL(1) = 0.0                                                     HYMN1850
      DQ  (1) = 0.0                                                     HYMN1860
      DUL (1) = -XK                                                     HYMN1870
      V1  (1) = 0.0                                                     HYMN1880
      V2  (1) = 1.0                                                     HYMN1890
C                                                                       HYMN1900
      HDZR = 0.5/DZ                                                     HYMN1910
      A1 = HDZ/ULO                                                      HYMN1920
C                              HI-ORDER TABLES                          HYMN1930
      DO 260 IZ = 2, IDZP                                               HYMN1940
         A2 = HDZ/UL(IZ)                                                HYMN1950
         V1(IZ) = V1(IZ-1) + A1 + A2                                    HYMN1960
         V2(IZ) = EXP( XK*V1(IZ) )                                      HYMN1970
         A1 = A2                                                        HYMN1980
         DUL (IZ) = ( UL  (IZ+1) - UL  (IZ-1) )*HDZR                    HYMN1990
         DRHO(IZ) = ( RHO (IZ+1) - RHO (IZ-1) )*HDZR                    HYMN2000
         DROL(IZ) = ( RHOL(IZ+1) - RHOL(IZ-1) )*HDZR                    HYMN2010
         DQ  (IZ) = ( OBAR(IZ+1) - OBAR(IZ-1) )*HDZR                    HYMN2020
         D2U (IZ) = ( DU  (IZ+1) - DU  (IZ-1) )*HDZR                    HYMN2030
  260    CONTINUE                                                       HYMN2040
C                       FINAL VALUES                                    HYMN2050
      DUL (IDZP) = XK*(UE-UL(IDZP))/UL(IDZP)                            HYMN2060
      DROL(IDZP) = 0.0                                                  HYMN2070
      DRHO(IDZP) = 0.0                                                  HYMN2080
      DQ  (IDZP) = 0.0                                                  HYMN2090
      D2U (IDZP) = 0.0                                                  HYMN2100
      D2U(2) = 0.5*D2U(3)                                               HYMN2110
      DQ (2) = 0.5*DQ (3)                                               HYMN2120
      D2U(IDZ) = 0.5*D2U(IDZ-1)                                         HYMN2130
      DQ (IDZ) = 0.5*DQ (IDZ-1)                                         HYMN2140
C                                                                       HYMN2150
      KNOT = XK .EQ. 0.0                                                HYMN2160
      XI2R = 0.0                                                        HYMN2170
      XI3R = 0.0                                                        HYMN2180
      XI4R = 0.0                                                        HYMN2190
      VK1 = 0.0                                                         HYMN2200
      VK2 = 0.0                                                         HYMN2210
      IF ( TABLR )             GO TO 270                                HYMN2220
      AHH  = AVN(1)                                                     HYMN2230
      BNH  = RVN (1) * RLOW                                             HYMN2240
      CHHR = CVNR(1) * TLOW                                             HYMN2250
      CHHI = CVNI(1) * TLOW                                             HYMN2260
C                                                                       HYMN2270
  270 CONTINUE                                                          HYMN2280
      NW  = XNW                                                         HYMN2290
C                                                                       HYMN2300
C*********************************************************************** HYMN2310
C                                                                       HYMN2320
C     CALCULATIONS FOR EACH FREQUENCY, W                                HYMN2330
C                                                                       HYMN2340
C*********************************************************************** HYMN2350
C                                                                       HYMN2360
      DO 320 IW = 1, NW                                                 HYMN2370
         W = WC(IW)                                                     HYMN2380
         OMEG2 = SNH*SNH   W*W                                          HYMN2390
         OMEG = SQRT(ABS(OMEG2))                                        HYMN2400
```

```
      LIMIT = .FALSE.
C
      CALL  INT4  ( WET, ERT, W, ER )
      CALL  INT4  ( WET, EIT, W, EI )
      IF ( OMEG2 )          280,290,300
  280 IM = 1
      GO TO 310
C
  290 IM = 2
      OMEG = 0.5E-10
      OMEG2 = 0.25E-20
C
  300 IM = 3
C
C     EVALUATION OF SIX INTEGRALS  BY  BOOLE  FORMULA  ( OR SIMPSON )
C      BOOLE INTEGRATION IF  ZING INPUT POSITIVE,  ELSE SIMPSON RULE
C        ( BOOLE IS SIMPSON RULE WITH ERROR FORMULA )
C
C        IM  IS  ( 1, 2, 3 )  AS  OMEG2  IS  ( -, 0, + ),
C        THEN USE  ( CIRCULAR, LIMIT, HYPERBOLIC )  FUNCTIONS
C
  310 DO 320 I = 1, 6
  320    ZING(I) = 0.0
      NOD = 3
      DO 430 IZ = 1, IDZP, INT
         Z = ZZ(IZ)
         GO TO (350,340,330), NOD
C
  330    WT = 1.0
         NOD = 1
         GO  TO 360
C
  340    WT = 2.0
         IF ( IZ .EQ. IDZP )   GO TO 350
         NOD = 1
         GO TO 360
C
  350    WT = 4.0
         NOD = 2
C
  360    PSI = OMEG * (ZE-Z)
         GO TO (390,370,380), IM
C
  370    ZEZ = ZE - Z
         ZF(1) = QBAR(IZ) * ZEZ
         ZF(2) = QBAR(IZ)
         ZF(3) = ZIP5(IZ) * ZEZ
         ZF(4) = ZIP5(IZ)
         ZF(5) = ZIP5(IZ) * ZEZ
         ZF(6) = ZIP5(IZ)
         GO TO 410
C
  380    CF = COSH(PSI)
         SF = SINH(PSI)
         GO TO 400
C
  390    CF = COS (PSI)
         SF = SIN (PSI)
C
  400    ZF(1) = QBAR(IZ) * SF
         ZF(2) = QBAR(IZ) * CF
         ZF(3) = ZIP5(IZ) * SF
         ZF(4) = ZIP5(IZ) * CF
         ZF(5) = ZIP5(IZ) * SF
         ZF(6) = ZIP5(IZ) * CF
C       ZF ARE COMPLETE INTEGRANDS
C
  410    DO 420 I = 1, 6
            ZING(I) = ZING(I) + WT*ZF(I)
  420       CONTINUE
  430    CONTINUE
C
      GO TO (460,440), INT
C
  440 INT = 1
      DO 450 I = 1, 6
  450    SIMP(I) = ZING(I)*DZ/1.5
C              SIMP IS SIMPSON INTEGRAL FOR IDZ/2 INCREMENTS
      GO TO 310
C
  460 DO 470 I = 1, 6
  470    ZING(I) = ZING(I)*DZ/3.0
C              ZING IS SIMPSON INTEGRAL FOR  IDZ  INCREMENTS
C
      IF ( SIMPL )          GO TO 490
    INT = 2
      DO 480 I = 1, 6
         ERRZ(I) = ( ZING(I)-SIMP(I) ) / 15.0
         IF ( ABS(ERRZ(I)/ZING(I) ) .GT. 0.05 )  CKOUT = .TRUE.
C    ZING, SIMP ARE SIMPSON INTEGRALS WITH  IDZ, IDZ/2  INCREMENTS
         ZING(I) = ZING(I) + ERRZ(I)
C    CORRECTED  ZING  IS NOW  BOOLE INTEGRAL
  480    CONTINUE
C
C     ABOVE  ZING(I)  ARE THE REQUIRED INTEGRALS
C
  490 IF ( .NOT. CKOUT )    GO TO 500
      WRITE (6,100)  ZING
      IF ( .NOT. SIMPL )    WRITE (6,110)  ERRZ
C
  500 OZE = OMEG * ZE
      WKSQ = XK*XK + W*W
      T1 = XK*W / WKSQ
      T2 = W*W / WKSQ
C
      Y1R = GAM*ZING(2)-ZING(2) + ZING(4) + W*T1*ZING(6)
      Y1I = XK*T1*ZING(6)
C
      GO TO (530,510,520), IM
C
  510 Y2R = -SNH*ER/W
      Y2I = -SNH*EI/W
      ERCOM = SNH*ER
      EICOM = SNH*EI
      GO TO 550
C
  520 COZE = COSH(OZE)
      SOZE = SINH(OZE)
      GO TO 540
C
  530 COZE =  COS(OZE)
```

```
      SOZE = -SIN(OZE)                                                  HYMN3620
C                                                                       HYMN3630
  540 ERCON = SNH*ER/OMEG                                               HYMN3640
      EICON = SNH*EI/OMEG                                               HYMN3650
      Y2R = -SNH*ER/W*COZE                                              HYMN3660
      Y2I = -SNH*EI/W*COZE - OMEG/W*SOZE                                HYMN3670
C                                                                       HYMN3680
  550 TEMP = GAM*ZING(1)-ZING(1) + ZING(3) + W*T1*ZING(5)               HYMN3690
      Y3R = ERCON*XK*T1*ZING(5) + EICON*TEMP                            HYMN3700
      Y3I = EICON*XK*T1*ZING(5) - ERCON*TEMP                            HYMN3710
      Y6R =  EICON*ZING(1) + ZING(2)                                    HYMN3720
      Y6I = -ERCON*ZING(1)                                              HYMN3730
C                                                                       HYMN3740
C                                                                       HYMN3750
C*********************************************************************** HYMN3760
C                                                                       HYMN3770
C****  CALCULATION OF FIRST-ORDER SOLUTION...                           HYMN3780
C                                                                       HYMN3790
C                                                                       HYMN3800
C                                                                       HYMN3810
  560 H1R = ( Y1R+Y2R+Y3R )/GAM                                         HYMN3820
      H1I = ( Y1I+Y2I+Y3I )/GAM                                         HYMN3830
      HD = Y6R*Y6R + Y6I*Y6I                                            HYMN3840
C                                                                       HYMN3850
      HR = ( H1R*Y6R + H1I*Y6I ) / HD                                   HYMN3860
      HI = ( H1I*Y6R - H1R*Y6I ) / HD                                   HYMN3870
C                                                                       HYMN3880
      THR(IW) = HR                                                      HYMN3890
      THI(IW) = HI                                                      HYMN3900
C                                                                       HYMN3910
      IF ( .NOT. HRUN )     GO TO 820                                   HYMN3920
C                                                                       HYMN3930
C*********************************************************************** HYMN3940
C                                                                       HYMN3950
C     TERMS FROM HIGH-ORDER ANALYSIS   ( STILL IN  W-LOOP )             HYMN3960
C        INITIALIZE  INTEGRALS, INTEGRANDS FOR  Z=0.0                   HYMN3970
C                                                                       HYMN3980
      DO 570 I = 1, 6                                                   HYMN3990
         AL(I) = 0.0                                                    HYMN4000
         DO 570 IZ = 1, IDZP                                            HYMN4010
  570       GS(I,IZ) = 0.0                                              HYMN4020
      DO 580 I = 1, 12                                                  HYMN4030
         OLD(I) = 0.0                                                   HYMN4040
         ZAP(I) = 0.0                                                   HYMN4050
         GL(I,1) = 0.0                                                  HYMN4060
  580    SLAM(I) = 0.0                                                  HYMN4070
      OLD(3) = -XK*OMEG2                                                HYMN4080
      IF ( IM .EQ. 1 )     SOZE = -SOZE                                 HYMN4090
      NOD = 1                                                           HYMN4100
      TEMP = -T1*RHOLO                                                  HYMN4110
      GL( 9,1) = TEMP*XK*W*SOZE                                         HYMN4120
      GL(10,1) = TEMP*XK*W*COZE                                         HYMN4130
      GL(11,1) = TEMP*OMEG2*SOZE                                        HYMN4140
      GL(12,1) = TEMP*OMEG2*COZE                                        HYMN4150
C                                                                       HYMN4160
      IF ( .NOT. TABLR )   GO TO 585                                    HYMN4161
      CALL  INT4  ( WIT, AVN , W, ANH  )                                HYMN4162
      CALL  INT4  ( WIT, BVN , W, BNH  )                                HYMN4163
      CALL  INT4  ( WIT, CVNR, W, CNHR )                                HYMN4164
      CALL  INT4  ( WIT, CVNI, W, CNHI )                                HYMN4165
C                                                                       HYMN4166
      BNH  = RLON * BNH                                                 HYMN4167
      CNHR = TLON * CNHR                                                HYMN4168
      CNHI = TLON * CNHI                                                HYMN4169
C                                                                       HYMN4170
  585 YE = ANH - CNHI/(W*GAM)                                           HYMN4171
      YF = ( BNH + CNHR ) / (W*GAM)                                     HYMN4180
      H2R = Y6R*YE - Y6I*YF                                             HYMN4190
      H2I = Y6R*YF + Y6I*YE                                             HYMN4200
      HSSQ = H2R*H2R + H2I*H2I                                          HYMN4210
      HA = ( H1R*H2R + H1I*H2I ) / HSSQ                                 HYMN4220
      HB = ( H1I*H2R - H1R*H2I ) / HSSQ                                 HYMN4230
C                                                                       HYMN4240
      CALL  INT4  ( WET, CRT, W, CR )                                   HYMN4250
      CALL  INT4  ( WET, CIT, W, CI )                                   HYMN4260
C                                                                       HYMN4270
      GO TO (590,610,630),  IM                                          HYMN4280
  590 DO 600 IZ = 1, IDZP                                               HYMN4290
         ARG = OMEG*ZZ(IZ)                                              HYMN4300
         STAB(IZ) = SIN (ARG)                                           HYMN4310
  600    CTAB(IZ) = COS (ARG)                                           HYMN4320
      SIGN = -1.0                                                       HYMN4330
      GO TO 650                                                         HYMN4340
C                                                                       HYMN4350
  610 DO 620 IZ = 1, IDZP                                               HYMN4360
         STAB(IZ) = ZZ(IZ)                                              HYMN4370
  620    CTAB(IZ) = 1.0                                                 HYMN4380
      GL(11,1) = TEMP * ZE                                              HYMN4390
      GL(12,1) = TEMP                                                   HYMN4400
      GO TO 650                                                         HYMN4410
C                                                                       HYMN4420
  630 DO 640 IZ = 1, IDZP                                               HYMN4430
         ARG = OMEG*ZZ(IZ)                                              HYMN4440
         STAB(IZ) = SINH(ARG)                                           HYMN4450
  640    CTAB(IZ) = COSH(ARG)                                           HYMN4460
      SIGN = 1.0                                                        HYMN4470
C                                                                       HYMN4480
C                                                                       HYMN4490
  650 DO 780 IZ = 2, IDZP                                               HYMN4500
         NOD = NOD + 1                                                  HYMN4510
         V3 = XK*RHOL(IZ)/RHO(IZ)                                       HYMN4520
         V4 = DU(IZ) + V3                                               HYMN4530
         V5 = QBAR(IZ)*V4 + RHO(IZ)*U(IZ)*DQ(IZ)                        HYMN4540
         V6 = ( RHOL(IZ)/UL(IZ)*DUL(IZ) - DROL(IZ) ) / RHO(IZ)          HYMN4550
         V7 = ( 2.0*(D2U(IZ)-DU(IZ)) + V3 - XK*V6 )                     HYMN4560
         V8  = XK*ZIP5(IZ) - U(IZ) - 3.0*DU(IZ)                         HYMN4570
     0   V9  = V3*( DU(IZ) - DUL(IZ) - UL(IZ)/RHOL(IZ)*DROL(IZ) )       HYMN4580
     1         + DU(IZ)*DU(IZ)                                          HYMN4590
     0   V10 = ( GAM*DQ(IZ) - DQ(IZ) + RHO(IZ)*D2U(IZ) + D2U(IZ) )      HYMN4600
     1         * U(IZ)/RHO(IZ)                                          HYMN4610
         V11 = U(IZ)/RHO(IZ)*DROL(IZ)/RHO(IZ)                           HYMN4620
         V12 = ( GAM*QBAR(IZ) - QBAR(IZ) )*DU(IZ)                       HYMN4630
         V13 = ( V12 + D2U(IZ) )/RHO(IZ)                                HYMN4640
         V16 = QBAR(IZ)/RHO(IZ)                                         HYMN4650
         V17 = (GAM-1.0) * ( V16*V16 + U(IZ)/RHO(IZ)*DQ(IZ) )           HYMN4660
     0   V18 = (U(IZ)-UL(IZ))*DQ(IZ) + QBAR(IZ)*(DU(IZ)-DUL(IZ)))       HYMN4670
     1         * RHO(IZ)                                                HYMN4680
         V19 = ( UL(IZ)-XK-DUL(IZ) ) / ( UL(IZ)*UL(IZ) ) * RHOL(IZ)     HYMN4690
         V20 = V19 + DROL(IZ)/UL(IZ)                                    HYMN4700
         V21 = V16 - GAM*V16 - DU(IZ) - DU(IZ)/RHO(IZ)                  HYMN4710
            V22 = GAM - 1.0                                             HYMN4720
```

```
      V23 = RHO(IZ) + 1.0                                               HYMN4730
      V24 = RHO(IZ) - 1.0                                               HYMN4740
      V25 = V22 * (QBAR(IZ)*DU(IZ) + U(IZ)*DQ(IZ))                      HYMN4750
      V26 = DU(IZ) * (V22*QBAR(IZ) + (GAM + V23)*DU(IZ))                HYMN4760
      V27 = RHO(IZ)*V10/T2                                              HYMN4770
     0V28 = V22 * (QBAR(IZ)*(QBAR(IZ)/RHO(IZ) + DU(IZ))                 HYMN4780
     1          + 2.0*U(IZ)*DQ(IZ))                                     HYMN4790
     0V29 = RHO(IZ) * (D2U(IZ)*(U(IZ  - UL(IZ))                        HYMN4800
     1                  + DU(IZ)*(DU(IZ) - DUL(IZ)))                   HYMN4810
      V30 = U(IZ)*D2U(IZ)                                               HYMN4820
C                                                                       HYMN4830
C     THE FUNCTIONS  V1 ... V20  ARE INDEPENDENT OF W.  HENCE WE COULD  HYMN4840
C        TRADE CORE FOR TIME BY TAKING ALL OR SOME OUT OF W-LOOP AS     HYMN4850
C        SUBSCRIPTED VARIABLES.  LET S SEE FIRST WHAT CORE WE HAVE.     HYMN4860
C                                                                       HYMN4870
      ARG = W * V1(IZ)                                                  HYMN4880
      CWV = COS(ARG)                                                    HYMN4890
      SWV = SIN(ARG)                                                    HYMN4900
      XI10R = CWV*V2(IZ)/RHO(IZ)                                        HYMN4910
      XI10I = SWV*V2(IZ)/RHO(IZ)                                        HYMN4920
      ZETAR = CWV/V2(IZ)                                                HYMN4930
      ZETAI = SWV/V2(IZ)                                                HYMN4940
      XI14R = XI10R*DUL(IZ)                                             HYMN4950
      XI14I = XI10I*DUL(IZ)                                             HYMN4960
      OZEZ = OMEG * ( ZE-ZZ(IZ) )                                       HYMN4970
      GO TO (660,670,680),  IM                                          HYMN4980
C                                                                       HYMN4990
  660 P0  = CTAB(IZ)                                                    HYMN5000
      PP0 = -OMEG*STAB(IZ)                                              HYMN5010
      ST = SIN ( OZEZ )                                                 HYMN5020
      CT = COS ( OZEZ )                                                 HYMN5030
      GO TO 690                                                         HYMN5040
C                                                                       HYMN5050
  670 P0  = 1.0                                                         HYMN5060
      PP0 = 0.0                                                         HYMN5070
      ST = ZE-ZZ(IZ)                                                    HYMN5080
      CT = 1.0                                                          HYMN5090
      GO TO 690                                                         HYMN5100
C                                                                       HYMN5110
  680 P0  = CTAB(IZ)                                                    HYMN5120
      PP0 = OMEG*STAB(IZ)                                               HYMN5130
      ST = SINH( OZEZ )                                                 HYMN5140
      CT = COSH( OZEZ )                                                 HYMN5150
C                                                                       HYMN5160
  690  IJK = 0                                                          HYMN5170
       DO 700 JZ = IZ, IDZP                                             HYMN5180
          IJK = IJK + 1                                                 HYMN5190
     0    GS(1,JZ) = (QBAR(IZ)*STAB(IJK)+QBAR(IZ-1)*STAB(IJK+1))        HYMN5200
     1                 * HDZ + GS(1,JZ)                                 HYMN5210
     0    GS(2,JZ) = (QBAR(IZ)*CTAB(IJK)+QBAR(IZ-1)*CTAB(IJK+1))        HYMN5220
     1                 * HDZ + GS(2,JZ)                                 HYMN5230
     0    GS(3,JZ) = (ZIP5(IZ)*STAB(IJK)+ZIP5(IZ-1)*STAB(IJK+1))        HYMN5240
     1                 * HDZ + GS(3,JZ)                                 HYMN5250
     0    GS(4,JZ) = (ZIP5(IZ)*CTAB(IJK)+ZIP5(IZ-1)*CTAB(IJK+1))        HYMN5260
     1                 * HDZ + GS(4,JZ)                                 HYMN5270
     0    GS(5,JZ) = (ZIP5(IZ)*STAB(IJK)+ZIP5(IZ-1)*STAB(IJK+1))        HYMN5280
     1                 * HDZ + GS(5,JZ)                                 HYMN5290
     0    GS(6,JZ) = (ZIP5(IZ)*CTAB(IJK)+ZIP5(IZ-1)*CTAB(IJK+1))        HYMN5300
     1                 * HDZ + GS(6,JZ)                                 HYMN5310
C                                                                       HYMN5320
  700     CONTINUE                                                      HYMN5330
C                                                                       HYMN5340
      P10R = -XK*W*GS(5,IZ) * T1                                        HYMN5350
      P10I = ( GAM*GS(1,IZ)-GS(1,IZ)+GS(3,IZ)+W*T1*GS(5,IZ) ) * W       HYMN5360
C     ( P11R = 0.0 )                                                    HYMN5370
      P11I = -W*GS(1,IZ)                                                HYMN5380
      IF ( LIMIT )       GO TO 710                                      HYMN5390
      P10R = P10R/OMEG                                                  HYMN5400
      P10I = P10I/OMEG                                                  HYMN5410
      P11I = P11I/OMEG                                                  HYMN5420
  710 PP10R = -W*XK*T1 * GS(6,IZ)                                       HYMN5430
     0PP10I = (GAM*GS(2,IZ) - GS(2,IZ) + GS(4,IZ)                       HYMN5440
     1                                 + W*T1*GS(6,IZ) ) * W            HYMN5450
C     ( PP11R = 0.0 )                                                   HYMN5460
      PP11I = -W*GS(2,IZ)                                               HYMN5470
C                                                                       HYMN5480
      IF ( KNOT )        GO TO 730                                      HYMN5490
C       EVALUATE INTEGRALS IN EXPRESSIONS FOR LAMDAS  ( AL(I) )         HYMN5500
      XI16R = ( W*RHOL(IZ)*ZETAR - V19*ZETAI ) * T1                     HYMN5510
      XI16I = ( W*RHOL(IZ)*ZETAI - V19*ZETAR ) * T1                     HYMN5520
      T3 = ( W*T1*DROL(IZ)/RHO(IZ) + D2U(IZ) ) * P0                     HYMN5530
      G(1) = T3 + XI10R                                                 HYMN5540
      G(2) = T3 + XI10I                                                 HYMN5550
      T4 = OMEG2*P0 + PP0                                               HYMN5560
      G(3) = XI14R * T4                                                 HYMN5570
      G(4) = XI14I * T4                                                 HYMN5580
      G(5) = PP0 * XI10R                                                HYMN5590
      G(6) = PP0 * XI10I                                                HYMN5600
      G( 7) = XI14R * PP11I                                             HYMN5610
      G( 8) = XI14I * PP11I                                             HYMN5620
      G( 9) = XI14R * PP10R                                             HYMN5630
      G(10) = XI14I * PP10R                                             HYMN5640
      G(11) = XI14R * PP10I                                             HYMN5650
      G(12) = XI14I * PP10I                                             HYMN5660
C                                                                       HYMN5670
      DO 720 I = 1, 12                                                  HYMN5680
         ZAP(I) = ( G(I) + OLD(I) )*HDZ + ZAP(I)                        HYMN5690
  720    OLD(I) = G(I)                                                  HYMN5700
C                 ( TRAPEZOIDAL RULE )                                 HYMN5710
     0    VK1 = XK * (  T2*RHOL(IZ)/RHO(IZ)*(V24*DU(IZ) - V3)          HYMN5720
     1                 + RHOL(IZ) * (W*T1*RHOL(IZ)/RHO(IZ)+DU(IZ)      HYMN5730
     2                                 - DUL(IZ) - XK*ZETAR)            HYMN5740
     3                 - UL(IZ)*DROL(IZ) - U(IZ)/RHO(IZ)*DROL(IZ)      HYMN5750
     4              - RHOL(IZ)/RHO(IZ)*(V22*QBAR(IZ) + V23*DU(IZ)))    HYMN5760
     0    VK2 = XK * (  T2*U(IZ)/RHO(IZ )DROL(IZ)                      HYMN5770
     1                 + XK*GAM*RHOL(IZ)*ZETAI/W)                      HYMN5780
C                                                                       HYMN5790
  730 XI1R = T2/RHO(IZ) * (V25 - V26 - V27 - V28 + V29 + VK1)           HYMN5800
      XI1I = T2 * (VK2 + V30)                                           HYMN5810
      XI2I = ( GAM*V16 - V16 + DU(IZ) + T2*V4 )*W                       HYMN5820
      XI3I = W*U(IZ) - T2/W*V7                                          HYMN5830
      XI4I = T1*V6 + 2.0*W*U(IZ)                                        HYMN5840
      XI5I = ( T2*V8 + DU(IZ)/RHO(IZ) )/W                               HYMN5850
C     XI2R = XI3R = XI4R = XI5R = 0                                      HYMN5860
C                                                                       HYMN5870
C                                                                       HYMN5880
C                                                                       HYMN5890
      IF ( KNOT )        GO TO 740                                      HYMN5900
      A1R = -XI16R*ZAP(8) - XI16I*ZAP(7)                                HYMN5910
      A1I =  XI16R*ZAP(7) - XI16I*ZAP(8) + GAM*P0*V5*T1                 HYMN5920
      A2R = T1/W*(V16*V5-V5/RHO(IZ))*P0                                 HYMN5930
```

```
      T? = [illegible]
      T6 = A??ZETAI - T?*ZETAR
      T7 = A??ZETAR - T??ZETAI
      T8 = [illegible] ( [illegible] )
C
      AAR = T1 * ( T8*( ZETAI*ZAP(5) + ZETAR*ZAP(6) )
     1          - [illegible]*( ZETAR*ZAP(1) + ZETAI*ZAP(2) )
     2          - T6*ZAP(3) + T7*ZAP(4) )
     3        + [illegible]
C
      AAI = T1 * ( T8*( ZETAI*ZAP(6) - ZETAR*ZAP(5) )
     1          - [illegible]*( ZETAR*ZAP(2) - ZETAI*ZAP(1) )
     2          + T7*ZAP(3) - T6*ZAP(4) )
     3        + [illegible]
C
 767  T9  = [illegible] * P111
      T12 = [illegible] * P111
      TEMP = -[illegible]
      A10 = A10 + TEMP*(T9-P1??) + [illegible]
     1         + [illegible]
      A11 = A11 + TEMP*(T19-P10?)
      A20 = A20 + V10*(T19-P10?)
      A21 = [illegible]
     1      [illegible]
      AAR = AAR + [illegible]    + [illegible]       + [illegible]
      AAI = AAI + [illegible]
     1           - [illegible]
C
C*****************************************************************
C
C     SET UP INTEGRANDS  ( SIMPSON RULE HERE  )
          DO 7?0 I = 1, 6
             SL(2*I-1,NDD) = AL(I)*ST
             GL(2*I  ,NDD) = AL(I)*CT
             AL(I) = 0.0
 7?0         CONTINUE
C
          GO TO (7?0,7?3,7?0), NDD
C
 760      DO 77? I = 1, 12
             SLAM(I) = SLAM(I) + GL(I,1) + 4.0*GL(I,2) + GL(I,3)
C                               WEIGHTED SUM FOR THREE ORDINATES
             GL(I,1) = GL(I,3)
 77?         CONTINUE
          NDD = 1
C
 78?      CONTINUE
C
C********************   ENDS  PRINCIPAL  Z-LOOP   ***************
C                          STILL DOING FOR  IW
C
C*****************************************************************
C
C       COMPLETE INTEGRATION
        DO 79C I = 1, 12
           SLAM(I) = SLAM(I) * DZ/3.0
 79C       CONTINUE
      IF ( LIMIT )            GO TO 80C
C     CALCULATE DEFECT  ( CON  IBUTION OF INTEGRALS FROM  Z=ZC TO Z=ZE )
        TEMP = 2.0*W*UE
        SLAM( 1) = SLAM( 1) + TEMP*P111*(CF-1.0)/OMEG2
        SLAM( 2) = SLAM( 2) + TEMP*P111*SF/OMEG
        SLAM( 9) = SLAM( 9) + TEMP*P101*(CF-1.0)/OMEG2
        SLAM(10) = SLAM(10) + TEMP*P101*SF/OMEG
        SLAM(11) = SLAM(11) - TEMP*P10R*(CF-1.0)/OMEG2
     1           -0.5*W*UE*( PSI*COZE-CTAB(IDZP)*SF )/OMEG
     2           + 0.5*T2/W*UE*( PSI*SOZE-STAB(IDZP)*SF )*SIGN
        SLAM(12) = SLAM(12) - TEMP*P10R*SF/OMEG
     1           - 0.5*W*UE*( PSI*SOZE-STAB(IDZP)*SF )*SIGN
     2           + 0.5*T2/W*UE*( PSI*COZE+CTAB(IDZP)*SF )*OMEG*SIGN
C
 800    G21R  = ANN*SLAM(1) + BNN*SLAM(5) + CNNR*SLAM(5) - CNNI*SLAM(7)
        G21I  = ANN*SLAM(3) + BNN*SLAM(7) + CNNR*SLAM(7) + CNNI*SLAM(5)
        DG21R = ANN*SLAM(2) + BNN*SLAM(6) + CNNR*SLAM(6) - CNNI*SLAM(8)
        DG21I = ANN*SLAM(4) + BNN*SLAM(8) + CNNR*SLAM(8) + CNNI*SLAM(6)
C
C         THE FOLLOWING AVAILABLE BY EQUIVALENCE...
C       G20R     IS  SLAM ( 9)
C       G20I     IS  SLAM (11)
C       DG20R    IS  SLAM (10)
C       DG20I    IS  SLAM (12)
C
C     FOR NONZERO OMEG, DIVISION BY OMEG IS IMPLICIT IN ERCON, EICON.
C      IAR, IAI, IBR, IBI  ARE TYPE REAL.
        TEMP = UE * W * ZING(2)
C     IAR = H1R - DG20R - ERCON*G20I - EICON*G20R
     1             + ( ( CI*Y1I - CR*Y1R )*W - CR*Y1I + CI*Y1R )*W*UE
C     IAI = H1I - DG20I + ERCON*G20R - EICON*G20I
     1             - ( ( CR*Y1I + CI*Y1R )*W + CR*Y1R - CI*Y1I )*W*UE
        IBR = H2R + DG21R + ERCON*G21I + EICON*G21R + TEMP*( CI-W*CR )
        IBI = H2I + DG21I - ERCON*G21R + EICON*G21I - TEMP*( CR+W*CI )
C
        HTD = IBR*IBR + IBI*IBI
        HTR = ( IAR*IBR + IAI*IBI ) / HTD
        HTI = ( IAI*IBR - IAR*IBI ) / HTD
C
        THTR(IW) = HTR
        THTI(IW) = HTI
C
      IF ( .NOT. CKOUT )      GO TO 820
      WRITE (6,150)
      WRITE (6,160)  W  ,  IAR,  IAI ,  IBR ,  IBI , ERCON, EICON,
     1               G20R, G21R, DG20R, DG21R,  H1R ,  H2R ,  CR  ,
     2               G20I, G21I, DG20I, DG21I,  H1I ,  H2I ,  CI
C
 820    CONTINUE
C                         END  OF  W-LOOP
C*****************************************************************
C
      KEP = NW
      KOUT(9) = XNW
C
      IF ( NR NN )      GO TO 850
C
      DO 830 IW = 1, NW
         KOUT(IW+ 9) = WC (IW)
         KOUT(IW+39) = THR(IW)
         KOUT(IW+69) = THI(IW)
 830     CONTINUE
      GO TO (880,840), KOUT
 840  IF ( CKOUT )      CALL PAGE  ( 70 )
```

```
      WRITE (6,60)                                                      HYWR7250
      WRITE (6,90)  ( WC(I), THR(I), THI(I),  I = 1, NW  )              HYWR7260
      GO TO 890                                                         HYWR7270
C                                                                       HYWR7280
  850 DO 860 IW = 1, NW                                                 HYWR7290
          ROUT(IW+ 9) = WC (IW)                                         HYWR7300
          ROUT(IW+39) = THTR(IW)                                        HYWR7310
          ROUT(IW+69) = THTI(IW)                                        HYWR7320
  860     CONTINUE                                                      HYWR7330
      GO TO (880,870),  KOUT                                            HYWR7340
  870 IF ( CKOUT )      CALL PAGE  ( 70 )                               HYWR7350
      WRITE (6,60)                                                      HYWR7360
      WRITE (6,60)  ( WC(I), THR(I), THI(I), THTR(I), THTI(I),          HYWR7370
     1                I = 1, NW  )                                      HYWR7380
C                                                                       HYWR7390
  880 IF ( .NOT. CKOUT )       GO TO 890                                HYWR7400
      CALL  PAGE ( 70 )                                                 HYWR7410
      WRITE (6,125)  ( ZZ(I), UU(I), BU(I), UL(I), BHO(I), BHOL(I),     HYWR7420
     1  BBAR(I), ZIP3(I), ZIP5(I),  I = 1, IBZP )                       HYWR7430
      IF  ( .NOT. HRUN )      GO TO 890                                 HYWR7440
      CALL  PAGE ( 70 )                                                 HYWR7450
      WRITE (6,135)                                                     HYWR7460
      WRITE (6,140)  ( ZZ(I),DZU(I),DUL(I),DBHO(I),DBHOL(I),DD(I),V1(I), HYWR7470
     1    V2(I),  I = 1, IBZP )                                         HYWR7480
C                                                                       HYWR7490
C                                                                       HYWR7500
C                                                                       HYWR7510
C     I.E.  HTR, HTI  ARE FINAL RESULTS OF THIS SUBROUTINE WHEN PLACED  HYWR7520
C           IN ROUT.   HTR, HTI INCLUDE INJECTOR EFFECTS, WHILE  HR, HI HYWR7530
C           DO NOT.  HENCE IF SUBROUTINE  500  IS ENTERED WITH HTR, HTI,HYWR7540
C           500  WILL EXPAND TABLE (BY INTERPOLATION), PRINT,           HYWR7550
C           AND RETURN.                                                 HYWR7560
C                                                                       HYWR7570
C                                                                       HYWR7580
C                                                                       HYWR7590
  890 RETURN                                                            HYWR7600
          END                                                           HYWR7610
```

```
$IBFTC MTINT   LIST,MAP                                                 HTNT
      SUBROUTINE DOD(DIN,DOUT,CD,NER,ERR)                               HTNT  10
C                                                                       HTNT  20
C     PROGRAM  D  COMPUTE HTR,HTI + INTERPOLATE 40 POINTS               HTNT  30
C           20 SEP 67  MODIFIED FOR TABULAR INJECTOR COEFFICIENTS       HTNT  40
C                                                                       HTNT  50
      LOGICAL LOGIK,HRUN,TABLR                                          HTNT  60
      COMMON /PROLOG/ LOGIK(50)                                         HTNT  70
      COMMON  /ABCDF /  EXTRA(100),  ABLOK(600), A   , B , SPACE(3556), HTNT  80
     1          WIT(17), AVN(17), BVN(17),CVNR(17), CVNI(17)            HTNT  90
      DIMENSION DIN(1),DOUT(1),A(133),B(193),OMEGA(1),HR(1),HI(1)       HTNT 100
     1,HTR(30),HTI(30),HTRINT(1),HTIINT(1), OMGA(1)                     HTNT 110
     2,DOM(30)                                                          HTNT 120
      EQUIVALENCE (LOGIK(8),HRUN),(LOGIK(17),TABLR)                     HTNT 130
      EQUIVALENCE (A(1),ANH),(A(2),BNH),(A(3),CNHRE),(A(4),CNHIM),      HTNT 140
     1(A(5),GAMMA),(A(6),XLRN),(A(7),XLON),(A(9),XNW),(A(10),OMEGA),    HTNT 150
     2(A(40),HR),(A(70),HI),(B(9),ONW),(B(10),OMGA),(B(51),HTRINT),     HTNT 160
     3(B(92),HTIINT) ,  ( OMEGA, DOM )                                  HTNT 170
C                                                                       HTNT 180
   10 FORMAT ( 21HO INPUT TO PROGRAM  D // 10X8H GAMMA = F8.4,5X6HLR/N =HTNT 190
     1  F12.8, 5X6HLT/N = F12.8 )                                       HTNT 200
   20 FORMAT ( 18X3HANH12X3HBNH11X6HCNH RE 9X6HCNH IM // 9X 4F15.7 // ) HTNT 210
   30 FORMAT ( // 38H INJECTOR DISTRIBUTION COEFFICIENTS... //          HTNT 220
     117X5HOMEGA 11X3HANH 12X3HBNH 11X6HCNH RE 9X6HCNH IM // (9X5F15.7)HTNT 230
   40 FORMAT (1H0, 19X,8HOMEGA(C),9X3H HR,13X3H HI  )                   HTNT 240
   50 FORMAT (1H0,19X, 8HOMEGA(C),9X3HHTR 13X3HHTI )                    HTNT 250
   60 FORMAT(1H ,10X,3F16.6)                                            HTNT 260
   70 FORMAT (1H0 ,20H PROGRAM  D  OUTPUT  // 19X,8HOMEGA(C) 9X         HTNT 270
     16H   HTR ,10X,6H   HTI  // )                                      HTNT 280
   80 FORMAT (91HOALL VALUES OF HTR ARE NEGATIVE- ( I.E. OUT OF RANGE OFHTNT 290
     1 INTEREST- WILL PROCEED TO NEXT CASE))                            HTNT 300
   90 FORMAT(//,30X,69H   FOLLOWING WILL BE INTERPOLATION WITHIN HTR HTIHTNT 310
     1 TABLE GIVEN ABOVE   )                                            HTNT 320
  100 FORMAT (19X,8H OMEGA  ,9X6HHTRINT,10X,6HHTIINT  )                 HTNT 330
  110 FORMAT (11X,F10.5,10X,F10.5,10X,F10.5)                            HTNT 340
C                                                                       HTNT 350
      ERR=0.0                                                           HTNT 360
      C=0.0                                                             HTNT 370
      CALL DVCHK (K000FX)                                               HTNT 380
  120 DO 130 I=1,133                                                    HTNT 390
      A(I)= D[illegible]                                                HTNT 400
  130 B(I)=0.0                                                          HTNT 410
      NER=IFIX(XNW)                                                     HTNT 420
      IF(NER)140,140,160                                                HTNT 430
  140 WRITE (6,150)NER                                                  HTNT 440
  150 FORMAT (1H0,10X,31H NUMBER OF OMEGAS IN ERROR  =  ,3X,I4   )      HTNT 450
      GO TO 510                                                         HTNT 460
  160 IF(NER-29)170,170,140                                             HTNT 470
  170 ONW = 40.0                                                        HTNT 480
      IF ( HRUN )              GO TO 190                                HTNT 490
      NWI = A(8) + 0.01                                                 HTNT 500
      IF ( NWI .GT. 2 )        GO TO 180                                HTNT 510
      ASSIGN 280 TO  NT                                                 HTNT 520
      TABLR = .FALSE.                                                   HTNT 530
      ANH = AVN (1)                                                     HTNT 540
      BNH = BVN (1)                                                     HTNT 550
      CNHRE = CVNR(1)                                                   HTNT 560
      CNHIM = CVNI(1)                                                   HTNT 570
      GO TO 190                                                         HTNT 580
  180 ASSIGN 270 TO  NT                                                 HTNT 590
      TABLR = .TRUE.                                                    HTNT 600
  190 IF(CD-99.0)260,260,200                                            HTNT 610
  200 CALL  PAGE  ( 60 )                                                HTNT 620
      WRITE (6,10)  GAMMA, XLRN, XLON                                   HTNT 630
      IF ( HRUN )              GO TO 230                                HTNT 640
      IF ( TABLR )             GO TO 210                                HTNT 650
      WRITE (6,20)  ANH, BNH, CNHRE, CNHIM                              HTNT 660
      GO TO 220                                                         HTNT 670
  210 WRITE (6,30)  ( WIT(I), AVN(I), BVN(I), CVNR(I), CVNI(I),         HTNT 680
     1                I = 1, NWI )                                      HTNT 690
  220 WRITE (6,40)                                                      HTNT 700
      GO TO 240                                                         HTNT 710
  230 WRITE (6,50)                                                      HTNT 720
  240 DO 250 I = 1,NER                                                  HTNT 730
  250 WRITE (6,60)OMEGA(I),HR(I),HI(I),                                 HTNT 740
C                                                                       HTNT 750
  260 IF ( HRUN )              GO TO 350                                HTNT 760
      DO 340 I=1,NER                                                    HTNT 770
      W = OMEGA(I)                                                      HTNT 780
      GO TO NT,(270,280)                                                HTNT 790
  270 CALL  INT4  ( WIT, AVN , W, ANH   )                               HTNT 800
      CALL  INT4  ( WIT, BVN , W, BNH   )                               HTNT 810
      CALL  INT4  ( WIT, CVNR, W, CNHRE )                               HTNT 820
      CALL  INT4  ( WIT, CVNI, W, CNHIM )                               HTNT 830
  280 DEN = GAMMA*W                                                     HTNT 840
      X = ANH - CNHIM/DEN*XLON                                          HTNT 850
      Y = BNH / DEN + XLRN + CNHRE * XLON / DEN                         HTNT 860
      SQD = X * X + Y * Y                                               HTNT 870
      HTR(I) = (X * HR(I) + Y * HI(I) ) /SQD                            HTNT 880
      HTI(I) = (X * HI(I) - Y * HR(I) ) / SQD                           HTNT 890
      CALL DVCHK (K000FX)                                               HTNT 900
       GO TO (310,290),K000FX                                           HTNT 910
  290 IF (CD-10.0) 340,300,300                                          HTNT 920
  300 IF (I-1) 330,320,330                                              HTNT 930
  310 C=10.0                                                            HTNT 940
  320 LIN=NER+6                                                         HTNT 950
      CALL PAGE (LIN)                                                   HTNT 960
      WRITE (6,70)                                                      HTNT 970
  330 WRITE (6,60)DOM(I),HTR(I),HTI(I)                                  HTNT 980
  340 CONTINUE                                                          HTNT 990
      GO TO 370                                                         HTNT1000
C                                                                       HTNT1010
  350 DO 360 I = 1,NER                                                  HTNT1020
      HTI(I) = HI(I)                                                    HTNT1030
  360 HTR(I) = HR(I)                                                    HTNT1040
  370 JOMEG1 = 0                                                        HTNT1050
      JOMEG2=0                                                          HTNT1060
      IF ( CD .LE. 99.0 )      CALL  PAGE  (60)                         HTNT1065
      DO 390 I =1,NER                                                   HTNT1070
      IF (HTR(I)) 390,390,380                                           HTNT1080
  380 JOMEG1 = I                                                        HTNT1090
      GO TO 410                                                         HTNT1100
  390 CONTINUE                                                          HTNT1110
      IF (JOMEG1)400,400,410                                            HTNT1120
  400 WRITE (6,80)                                                      HTNT1130
      ERR= -1.0                                                         HTNT1140
      GO TO 520                                                         HTNT1150
  410 DO 430 J =I,NER                                                   HTNT1160
      IF (HTR(J)) 420,430,430                                           HTNT1170
  420 JOMEG2 = J                                                        HTNT1180
      GO TO 440                                                         HTNT1190
```

```
430 CONTINUE                                                          MTRT1200
    JONE62 =NER                                                       MTRT1210
440 DLTME6 = (DOM(JONE62) -DOM(JONE61))/ 39.0                         MTRT1220
    L=NER+1                                                           MTRT1230
    MTI(L)=0.0                                                        MTRT1240
    MTR(L)=0.0                                                        MTRT1250
    DOM(L)=0.0                                                        MTRT1260
    OMGA(1) =DOM(JONE61)                                              MTRT1270
    DO 490 I=1,40                                                     MTRT1280
    CALL INT4 (DOM(1),MTR(1),OMGA(1),MTRINT(1) )                      MTRT1290
    CALL INT4 (DOM(1) MTI(1),OMGA(1),MTIINT(1) )                      MTRT1300
    IF (CD-10.6)480,4>3,450                                           MTRT1310
450 IF ( I-1 )   470,460,470                                          MTRT1320
460 IF ( .NOT. MRUN )    CALL  PAGE  ( 60 )                           MTRT1330
    WRITE (6,90)                                                      MTRT1340
    WRITE (6,100)                                                     MTRT1350
470 WRITE (6,60) OMGA(I),MTRINT(I),MTIINT(I)                          MTRT1360
480 OMGA(I+1) = OMGA(I) + DLTME6                                      MTRT1370
490 CONTINUE                                                          MTRT1380
    DO 900 I=1,135                                                    MTRT1390
900 DOUT (I) = B(I)                                                   MTRT1400
    IF(C) 910,920,910                                                 MTRT1410
910 NER=0                                                             MTRT1420
920 RETURN                                                            MTRT1430
    END                                                               MTRT1440
```

```
*TITLE *ADMIN                                                          *ADM0010
* ALGZ - ADAMS METHOD OF INTEGRATION                                   *ADM0020
* FAP TO MAP CONVERSION MODIFIED DEC 1963 . FORTRAN 4 PATTERSON JULY 64 *ADM0030
        ENTRY   ADMSET                                                 *ADM0040
        ENTRY   ADMINT                                                 *ADM0050
        ENTRY   ADMCOR                                                 *ADM0060
        ENTRY   ADMPAR                                                 *ADM0070
        ENTRY   ADMRES                                                 *ADM0080
*ADMSET SAVE    1                                                      *ADM0090
        LDQ     1.4                                                    *ADM0100
        STA     RZ                      N TO ADD  OF RZ                *ADM0110
        ALS     7                       4N                             *ADM0120
        STA     BP18            SAVE 4N.  SET FLAG FOR RETURN          *ADM0130
        ADD*    3.4                     4N                             *ADM0140
        STA     CF                      5N                             *ADM0150
        ADC     7.4                             PUT IN                 *ADM0160
        ADD     BP18            4N                                     *ADM0170
        STA     CC                              PUT IN    DIFF TABLE LOC *ADM0180
        CLA     6.4                     F FUNCTION LOCATION            *ADM0190
        ADD*    3.4                     4N                             *ADM0200
        STA     LF                      FUNCTION VALUE LOC             *ADM0210
        CLA     4.4                     DERIV LOC                      *ADM0220
        ADD*    3.4                     4N                             *ADM0230
        STA     LD                      DERIV VALUE LOC                *ADM0240
        CLA     6.4                     PARTIAL STEP LOC               *ADM0250
        ADD*    3.4                     4N                             *ADM0260
        STA     CP                      PARTIAL STEP LOC               *ADM0270
        LDQ     7.4                                                    *ADM0280
        ADD*    3.4                     4N                             *ADM0290
        STA     LG                      TEST VALUE LOC.     7.4 WAS PUT IN *ADM0300
        ADD*    3.4                     4N                             *ADM0310
        STA     CA                      ACCURACY EXPONENT LOC          *ADM0320
        ADD*    3.4                     4N                             *ADM0330
        STA     CF                      OLD FCN. LOC.                  *ADM0340
        ADD*    3.4                                                    *ADM0350
        STA     CD                                                     *ADM0360
        CLA     8.4                                                    *ADM0370
        STA     CX                      X LOC                          *ADM0380
        CLA     9.4                                                    *ADM0390
        STA     CH                      H LOC                          *ADM0400
        CLA     10.4                                                   *ADM0410
        ADD*    3.4                     4N                             *ADM0420
        STA     CD                      ACCURACY INPUT TABLE           *ADM0430
        TRA     BP20                                                   *ADM0440
*ADMPAR SAVE    1                                                      *ADM0450
        STZ     BP18                                                   *ADM0460
BP20    STZ     FLAG                                                   *ADM0470
        LXA     RZ.4                    4N                             *ADM0480
CD      CAL     .4.4                    ACCURACY INPUT.4               *ADM0490
        ANA     =0377000000000                                         *ADM0500
        TNZ     4.2                     TEST FOR 0 ACCURACY INPUT      *ADM0510
        CLA     =0152000000000          ACCURACY SET= +152             *ADM0520
        ADD     =0022000000000                                         *ADM0530
CA      STO     .4.4                    ACCURACY EXPONET.4             *ADM0540
        TIX     CD.4.2                                                 *ADM0550
        LXA     CF.1                    4N                             *ADM0560
BP      STZ*    CC                      DIFF TABLE.1                   *ADM0570
        TIX     BP.1.1                                                 *ADM0580
        AXT     5.1                                                    *ADM0590
        STT     CNT.1                                                  *ADM0600
        CLA     BP18                                                   *ADM0610
        TNZ     BP19                                                   *ADM0620
        RETURN  ADMRES                                                 *ADM0630
BP19    RETURN  ADMSET                                                 *ADM0640
BP18    PZE                                                            *ADM0650
ADMINT  SAVE                                                           *ADM0660
        CLA     FLAG                                                   *ADM0670
        TNZ     RZ                                                     *ADM0680
        RETURN  ADMINT                                                 *ADM0690
RZ      AXT     .4.4                    4N                             *ADM0700
BP2     CLA*    LF                      FUNCTION.4                     *ADM0710
CF      STO     .4.4                    OLD FUNCTION.4                 *ADM0720
        TIX     BP2.4.1                                                *ADM0730
RF      AXT     COEFP.4                 PRED TABLE LOC                 *ADM0740
        STA     CO.4                                                   *ADM0750
        TIX     COMP.4                                                 *ADM0760
CX      CLA     .                       X                              *ADM0770
CH      FAD     .                       H                              *ADM0780
        STO*    CX                      STORE IN X                     *ADM0790
        RETURN  ADMINT                                                 *ADM0800
COMP    SXA     IR1.1                                                  *ADM0810
        SXA     IR2.2                                                  *ADM0820
        SXA     IR4.4                                                  *ADM0830
        LXA     RZ.4                    N TO IN 4                      *ADM0840
CF      AXT     .4.1                    5N   IN 1                      *ADM0850
CR      AXT     5.2                                                    *ADM0860
        STZ     TEMP                                                   *ADM0870
CC      LDQ     .4.1                    DIFF TABLE.1                   *ADM0880
        FMP     .4.2                     OFF.2                         *ADM0890
        FAD     TEMP                                                   *ADM0900
        STO     TEMP                                                   *ADM0910
        TXI     BP3.1.-1                NEXT DIFF                      *ADM0920
BP3     TIX     CC.2.1                  NEXT COEF                      *ADM0930
        AXT     0.2                                                    *ADM0940
LD      LDQ     .4.4                    DERIV.4                        *ADM0950
        FMP*    CO                      COEF.2                         *ADM0960
        FAD     TEMP                                                   *ADM0970
        LRS     35                                                     *ADM0980
        FMP*    CH                      H                              *ADM0990
LG      STO     .4.4                    TEST VALUE                     *ADM1000
CCM     FAD*    CX                      OLD FUNCTION                   *ADM1010
LM      STO*    LF                      FUNCTION.4                     *ADM1020
        TIX     CR.4.1                  NEXT EQUATION                  *ADM1030
IR4     AXT     .4.4                                                   *ADM1040
IR2     AXT     .4.2                                                   *ADM1050
IR1     AXT     .4.1                                                   *ADM1060
        TNA     1.4                                                    *ADM1070
ADMCOR  SAVE    1.2                                                    *ADM1080
        CLA     FLAG                                                   *ADM1090
        TZF     FIRST                                                  *ADM1100
        STO     TT                      0 TO DECR TT                   *ADM1110
        LXA     RZ.4                    N IN 4                         *ADM1120
DA      CLA*    LG                      TEST VALUE.4                   *ADM1130
CO      STO     .4.4                    PREDICTOR TEST VALUE.4         *ADM1140
        TIX     DA.4.1                                                 *ADM1150
        AXT     COFFC.4                 CORRECTOR TABLE LOC            *ADM1160
        STA     CO.4                                                   *ADM1170
        TIX     COMP.4                                                 *ADM1180
        LXA     RZ.4                    N TO IN 4                      *ADM1190
LF      CAL     .4.4                    FUNCTION.4                     *ADM1200
        ANA     =0377000000000          KEEP EXP ONLY                  *ADM1210
```

```
        SXD     -0002000000000      SHIFT EXP                       *ADM1220
        UFA*    L6                  TEST VALUE.4                    *ADM1230
        UFS*    C0                  DEL YC - DEL YP                 *ADM1240
        SXP                                                         *ADM1250
        UFA*    CA                                                  *ADM1260
        ARS     -0000777777776      DOUBLING TEST 0 DOUBLE          *ADM1270
        TZE     BP4                                                 *ADM1280
        SXD     TT.4                NO DOUBLE FLAG OVER 0           *ADM1290
BP4     ARA     -0000777777600      HAVING TEST 0 NO HALVE          *ADM1300
        TNZ     HALV                NO ZERO HALVE                   *ADM1310
DE      TIX     LF.4.1                                              *ADM1320
        LXA     BZ.4                .R                              *ADM1330
        LXD     CNT.1                                               *ADM1340
        TRL     BP5.1.0                                             *ADM1350
        TXI     BP6.1.-1            REDUCE CNT BY 1                 *ADM1360
BP6     SXD     CNT.1                                               *ADM1370
BP5     LXA     CF.1                SN                              *ADM1380
LP      AXT     5.2                                                 *ADM1390
        LDQ*    LD                  DERIV.4                         *ADM1400
CPR     CLS*    CC                  DIFF TABLE.1                    *ADM1410
        STQ*    CC                  DIFF TABLE.1                    *ADM1420
        FAD*    CC                                                  *ADM1430
        LRS     35                                                  *ADM1440
CNT     TIX     BP7.2.+                                             *ADM1450
        LDQ     COEFP               0.                              *ADM1460
BP7     TXI     BP8.1.-1            NEXT DIFF                       *ADM1470
BP8     TIX     CPR.2.1                                             *ADM1480
        TIX     LP.4.1                                              *ADM1490
TT      TRL     OUT.4.+             ** ZERO DOUBLE                  *ADM1500
        AXT     TBBL.4                                              *ADM1510
        SXA     TV.4                                                *ADM1520
        TSX     MAT.4                                               *ADM1530
        LDQ*    CH                  H                               *ADM1540
        FMP     THAL.1              2.0                             *ADM1550
        STQ*    CH                  H                               *ADM1560
OUT     RETURN  ADMCOR                                              *ADM1570
HALV    AXT     THAL.4              HAVING TABLE                    *ADM1580
        SXA     TV.4                                                *ADM1590
        TSX     MAT.4                                               *ADM1600
        CLA*    CX                  X                               *ADM1610
        FSB*    CH                  H                               *ADM1620
        STQ*    CX                  X BACKED UP                     *ADM1630
        LDQ*    CH                  H                               *ADM1640
        FMP     TPAL                .5                              *ADM1650
        STQ*    CH                  NEW H                           *ADM1660
        TSX     IR2.4               RESTORE IR1.IR2 DESTROYED BY MAT *ADM1670
        TRA     BF                                                  *ADM1680
FIRST   STL     FLAG                FLAG NOT = ZERO                 *ADM1690
        SXA     IR1F.1                                              *ADM1700
        LXA     BZ.4                N                               *ADM1710
        LXA     CF.1                SN                              *ADM1720
BP10    CLA*    LD                  DERIV.4                         *ADM1730
        STQ*    CC                  DIFF TABLE.1                    *ADM1740
        TXI     *+1.1.-5                                            *ADM1750
BP9     TIX     BP10.4.1                                            *ADM1760
IR1F    AXT     *.1                                                 *ADM1770
        TRA     BZ                                                  *ADM1780
MAT     LXA     CF.1                SN                              *ADM1790
        SXA     IR4M.4                                              *ADM1800
TW      AXT     4.4                 4                               *ADM1810
        AXT     9.2                 9                               *ADM1820
TV      SXD     TU.4                STORE IR4 IRSHIFT LOC           *ADM1830
        TXI     BP11.1.-1           MOVE PAST V OR PAST DIFF.       *ADM1840
BP11    STZ     TEMP                                                *ADM1850
TX      LDQ*    CC                  DIFF TABLE.1                    *ADM1860
TY      FMP     *.2                 MAT COEF.2                      *ADM1870
        FAD     TEMP                                                *ADM1880
        STO     TEMP                                                *ADM1890
        TXI     BP12.1.-1           MOVE TO NEXT DIFF               *ADM1900
BP12    TXI     BP14.2.-1           MOVE TO NEXT COEF               *ADM1910
BP14    TIX     TX.4.1              GO FOR MORE TERMS               *ADM1920
TU      TXI     TZ.1.+              MOVE BACK TO ORIG DIFF          *ADM1930
TZ      STQ*    CC                  DIFF TABLE.1                    *ADM1940
        LXD     TU.4                                                *ADM1950
        TIX     TV.4.1                                              *ADM1960
        TIX     TW.1.1              MOVE TO V TERM OR ALL THROUGH   *ADM1970
IR4M    AXT     *.4                                                 *ADM1980
        TRA     1.4                                                 *ADM1990
ADMPAR  SAVE                                                        *ADM2000
        SXA     IR2.2                                               *ADM2010
        SXA     IR1.1                                               *ADM2020
        CLA*    3.4                                                 *ADM2030
        FSB*    CX                  XP-X                            *ADM2040
        FDP*    CH                  XP-X /H =P                      *ADM2050
        STO     TEMPA-5             P TO TEMPA-5                    *ADM2060
        STZ     TEMPA                                               *ADM2070
        AXT     -4.4                GENERATE POWERS OF P            *ADM2080
BP16    FMP     TEMPA-5                                             *ADM2090
        STO     TEMPB-4.4                                           *ADM2100
        LRS     35                                                  *ADM2110
        TXI     BP15.4.1                                            *ADM2120
BP15    TXH     BP16.4.0                                            *ADM2130
        AXT     5.1                                                 *ADM2140
        AXT     CPP.4                                               *ADM2150
        SXA     TX.4                                                *ADM2160
        SXA     TZ.4                                                *ADM2170
        AXT     TPBL.4                                              *ADM2180
        SXA     TV.4                                                *ADM2190
        TSX     MAT+1.4             AVOIDS SN TO IN1                *ADM2200
        AXT     CC.4                                                *ADM2210
        SXA     TX.4                                                *ADM2220
        SXA     TZ.4                                                *ADM2230
        AXT     1.1                                                 *ADM2240
        AXT     4.2                                                 *ADM2250
CPP     CLA     TEMPB+1.1                                           *ADM2260
        STO     TEMPA.2                                             *ADM2270
        TXI     BP17.1.1                                            *ADM2280
BP17    TIX     CPP.2.1                                             *ADM2290
        AXT     TEMPA.4                                             *ADM2300
        SXA     CO.4                                                *ADM2310
        AXT     LF.4                                                *ADM2320
        SXA     CCN.4                                               *ADM2330
        AXT     CP.4                                                *ADM2340
        SXA     CM.4                                                *ADM2350
        TSX     COMP+2.4                                            *ADM2360
        AXT     LF.4                                                *ADM2370
        SXA     CN.4                                                *ADM2380
        AXT     CK.4                                                *ADM2390
        SXA     CCN.4                                               *ADM2400
        RETURN  ADMPAR                                              *ADM2410
CP      PZE     0.4                                                 *ADM2420
```

```
[illegible]  [illegible]                                                    *AD=2630
             [illegible]   1.                  1                            *AD=2640
             [illegible]   .5                  1/2                          *AD=2650
             DEC           [illegible]         [illegible]   5/12           *AD=2660
             DEC           .[illegible]        1/8                          *AD=2670
             DEC           [illegible]         .[illegible]  291/720        *AD=2680
[illegible]  DEC           0.                                               *AD=2690
             DEC           [illegible]         .[illegible]  [illegible]    *AD=2700
             DEC           [illegible]         .[illegible]  [illegible]    *AD=2710
             DEC           [illegible]                       [illegible]    *AD=2720
             DEC           [illegible]         .[illegible]  19/720         *AD=2730
             DEC           0.                  .00                          *AD=2740
[illegible]  DEC           [illegible]         .36[illegible]  291/720      *AD=2750
             DEC           .[illegible]                      64/720         *AD=2760
             DEC           .0625                             [illegible]    *AD=2770
             DEC           .[illegible]                      110/720        *AD=2780
             DEC           .125                              90/720         *AD=2790
             DEC           .[illegible]                      1/24           *AD=2800
             DEC           .[illegible]                      1/6            *AD=2810
             DEC           .[illegible]                      1/6            *AD=2820
             DEC           .[illegible]                      1/6            *AD=2830
             DEC           .25                               1/6            *AD=2840
TMPL         DEC           .5                               *AD=2850
             DEC           .125                             *AD=2860
             DEC           .2625                            *AD=2870
             DEC           .0390625                         *AD=2880
             DEC           .25                              *AD=2890
             DEC           .125                             *AD=2900
             DEC           .078125                          *AD=2910
             DEC           .375                             *AD=2920
             DEC           .09375                           *AD=2930
TMAL         DEC           .0625                            *AD=2940
             DEC           1.                               *AD=2950
             DEC           -1.                              *AD=2960
             DEC           0                                *AD=2970
             DEC           0                                *AD=2980
             DEC           4.                               *AD=2990
             DEC           -4.                              *AD=3000
             DEC           1.                               *AD=3010
             DEC           8.                               *AD=3020
             DEC           -12.                             *AD=3030
TMAL         DEC           16.                              *AD=3040
TEMP         BSS           11                               *AD=3050
TEMPA        SYN           TEMP+6                           *AD=3060
TEMPB        SYN           TEMP+10                          *AD=3070
             END                                            *AD=3080
```

```
$IBFTC NOZWIT  LIST,MAP                                                 NOZW
      SUBROUTINE CCC(DIN,DOUT,WC,CODE,NER)                              NOZW  10
C*****************************************************************NOZW  20
C***** THIS PROGRAM WAS WRITTEN FROM A REPORT ON NOZZLE ADMITTANCE******NOZW  30
C***** THEORY FROM PRINCETON UNIVERSITY.  THE ANALYSIS WAS DONE BY******NOZW  40
C***** CARL LUNDELIUS AND THE PROGRAMMING BY JERRY HOWARD.JOB 8052******NOZW  50
C*************                                                         NOZW  60
C     NOZWIT MODIFIED 29 JUL 67  TO SUPPLY CR1,CI1 IN PLACE OF BR1,BI1. NOZW  70
C*****************************************************************NOZW  80
C                                                                       NOZW  90
      LOGICAL  LOGIK,  SL1,  SL2,  EORJ                                 NOZW 100
      COMMON  /ABCDF/  EXTRA(100),  ABLOK(600)                          NOZW 110
     1      , XX      , U2TBL   , DESIRE  , RAT     , RAC     , RCC     NOZW 120
     2      , RCT     , ALFA    , G       , KN      , Y       , YP      NOZW 130
     3      , YOUT    , TEMP    , E       , XTABLE  , YTABLE  , A       NOZW 140
     4      , R       , AM      , AP      , AMP     , ZZ      , AMN     NOZW 150
     5      , AMP2    , CALFA   , CTALFA  , DELAM   , DELTZ   , FKN     NOZW 160
     6      , G1      , G2      , G3      , G4      , JPLAG1  , KRN1    NOZW 170
     7      , K       , RN      , PI      , PROD    , RSTA1   , RSTA2   NOZW 180
     8      , SALFA   , T1      , T2      , T3      , XINT    , XK      NOZW 190
     9      , ZZ1     , ZZ2     , ZZ3     , A1      , ABC     , ABD     NOZW 200
      COMMON  /ABCDF/                                                   NOZW 210
     1        AI1     , ALPHAI  , ALPHAR  , AR1     , BI01    , BI02    NOZW 220
     2      , BI0     , B1      , B2      , B3      , B4      , B5      NOZW 230
     3      , B6      , B7      , B8      , BR1     , BR2     , B9      NOZW 240
     4      , BI1     , BR1     , C2      , C3      , CNII    , CNIR    NOZW 250
     5      , CI1     , CR1     , C       , D10     , D11     , D1      NOZW 260
     6      , D2      , D3      , D4      , D5      , D6      , D7      NOZW 270
     7      , D8      , D9      , DC2     , D       , DU2     , EI      NOZW 280
     8      , ER      , F31     , F3R     , FI      , FR      , H1      NOZW 290
     9      , H       , I       , IWO     , IW      , IWW     , J       NOZW 300
      COMMON  /ABCDF/                                                   NOZW 310
     1        NDESIR  , NK      , NP      , S2      , S       , TT      NOZW 320
     2      , U2      , U       , W2      , W       , XI01    , XI0R    NOZW 330
     3      , XI2I    , XI2R    , XI      , XJI     , XJR     , XNNEW   NOZW 340
     4      , XNOLD   , XNEW    , XOLD    , XPT     , X       , ZI      NOZW 350
     5      , ZR                                                       NOZW 360
      COMMON  /PROLOG/  LOGIK(50), SL1, SL2, EORJ                       NOZW 370
C                                                                       NOZW 380
      DIMENSION XX(200),U2TBL(200),XTABLE(200),YTABLE(200),ZZ(200)      NOZW 390
      DIMENSION  Y(8),YP(8),YOUT(8),TEMP(72),E(8)                       NOZW 400
      DIMENSION A(200),R(200),AM(200),AP(200),AMP(200)                  NOZW 410
      DIMENSION   DIN(1), DOUT(1),WC(1)                                 NOZW 420
C                                                                       NOZW 430
C*****************************************************************NOZW 440
C                                                                       NOZW 450
C                                                                       NOZW 460
      NER  =  0                                                         NOZW 470
C ***********                                                           NOZW 480
C READ INPUT*                                                           NOZW 490
C ***********                                                           NOZW 500
      G  =  DIN(1)                                                      NOZW 510
      NOMEG=WC(2)+2.0001                                                NOZW 520
      NDESIR  =  DIN(2)  +  .0001                                       NOZW 530
      GEO=.5*(G+1.0)*SQRT(DIN(4)*DIN(7))                                NOZW 540
      IF  (CODE)   10,20,30                                             NOZW 550
   10 NER =  1                                                          NOZW 560
   20 RETURN                                                            NOZW 570
   30 IF(WC(1))40,50,40                                                 NOZW 580
   40 WCONST=DOUT(3)/(6.2831853*DOUT(1))                                NOZW 590
      GEO=GEO/DOUT(1)                                                   NOZW 600
      GO TO 60                                                          NOZW 610
   50 WCONST=DOUT(3)/(6.2831853*DOUT(2))                                NOZW 620
      GEO=GEO/DOUT(2)                                                   NOZW 630
   60 IF(CODE-99.0)100,100,70                                           NOZW 640
C ****************                                                      NOZW 650
C PRINT OUT INPUT*                                                      NOZW 660
C ****************                                                      NOZW 670
   70 CALL PAGE(5)                                                      NOZW 680
      WRITE (6,80)                                                      NOZW 690
   80 FORMAT (1H0,75H PROGRAM  C  INPUT - CALCULATES NOZZLE ADMITTANCE  NOZW 700
     1COEFFICIENTS USING 8052   )                                       NOZW 710
      WRITE (6,90)G,  NDESIR                                            NOZW 720
   90 FORMAT (1H0,5X,3HG =,F9.3,11H , NDESIR =,I2  )                    NOZW 730
  100 CONTINUE                                                          NOZW 740
C ***************************************************************       NOZW 750
C NDESIR=1,INPUT TABLE,INPUT DESIRE  NDESIR=2,CALCULATE TABLE,INPUT*    NOZW 760
C DESIRE  NDESIR=3,CALCULATE TABLE,CALCULATE DESIRE (AT LAST POINT)*    NOZW 770
C ***************************************************************       NOZW 780
      GO TO (130,110,110),NDESIR                                        NOZW 790
  110 RAT   =  DIN(4)                                                   NOZW 800
      RAC   =  DIN(5)                                                   NOZW 810
      RCC   =  DIN(6)                                                   NOZW 820
      RCT   =  DIN(7)                                                   NOZW 830
      ALFA  =  DIN(8)                                                   NOZW 840
      KN    =  DIN(9)  +  .0001                                         NOZW 850
      IF   (CODE  -  99.0)   140,140,120                                NOZW 860
  120 CALL  PAGE (1)                                                    NOZW 870
      WRITE (6,130)RAT, RAC, RCC, RCT, ALFA, KN                         NOZW 880
  130 FORMAT  (1H ,5X5HRAT =,F7.3,6H,RAC =,F7.3,6H,RCC =,F7.3,6H,RCT =, NOZW 890
     1   F7.3, 7H,ALFA = F7.3,5H,KN = I4   )                            NOZW 900
  140 CONTINUE                                                          NOZW 910
      CALL TBLCAL                                                       NOZW 920
      KN = KN/2 + 1                                                     NOZW 930
      GO TO 190                                                         NOZW 940
C ***************************************************                   NOZW 950
C READ VELOCITY POTENTIAL TABLE, FIRST POINT IS (0,1)*                  NOZW 960
C ***************************************************                   NOZW 970
  150 J = 10                                                            NOZW 980
      DO 170 I = 1,200                                                  NOZW 990
      I = I                                                             NOZW1000
      XX(I) =  DIN(J)                                                   NOZW1010
      U2TBL(I)  =  DIN(J+1)                                             NOZW1020
      J = J+2                                                           NOZW1030
      IF (I)  170,170,160                                               NOZW1040
  160 IF ( XX(I) )   170,180,180                                        NOZW1050
  170 CONTINUE                                                          NOZW1060
C ***************************************************                   NOZW1070
C KN = COUNT OF TOTAL NO. OF POINTS IN THE TABLE.*                      NOZW1080
C ***************************************************                   NOZW1090
  180 KN = I - 1                                                        NOZW1100
  190 IF(CODE-199.0)240,240,200                                         NOZW1110
  200 CALL PAGE(70)                                                     NOZW1120
      WRITE (6,220)                                                     NOZW1130
      KKKN=KN/2+1                                                       NOZW1140
      DO 210 I=1,KKKN                                                   NOZW1150
      KKK=KKKN+I                                                        NOZW1160
      VOL1=SQRT(U2TBL(I))                                               NOZW1170
      VOL2=SQRT(U2TBL(KKK))                                             NOZW1180
      WRITE (6,230)I,ZZ(2*I-1),XX(I),VOL1,KKK,ZZ(2*KKK-1),XX(KKK),VOL2  NOZW1190
  210 CONTINUE                                                          NOZW1200
```

```
220 FORMAT(///,60X, 28HVELOCITY POTENTIAL TABLE ,//,39X, 46H  AXIAL P
   1OSITION IN N        (INCHES)          ,//,39X,32H    VELO ITY POTEN
   2TIAL (NONDIMENSIONAL) ,//,39X, 49HV  AXIAL VELO ITY  NONDIMENSIONAL
   3IZED BY SPEED OF SOUND AT THROAT)
   4 ,///,9X,5HPOINT,11X,1HZ,16X,1HR,15X,1HV,16X,5HPOINT,11X,1HZ,16X,1H
   59X,16X,1HV,//,)
230 FORMAT(6X,I3,6X,3(E12.5,4X),6X,I3,6X,2(E12.5,5X),E12.5,)
240 CONTINUE
C ***************************************************
C REVERSE ORDER OF TABLE VALUES AND SET  P FOR INTAD*
250 NK = KN+1
    XTABLE(NK) = 0.0
    YTABLE(NK) = 0.0
    DO 260 I = 1,KN
    NK = NK - 1
    XTABLE(I) = XX(NK)
    YTABLE(I) =  ZTBL(NK)
260 CONTINUE
C
C ************************
C READ ONE CASE AT A TIME*
C ************************
    IWW  = 410
    IW   =  2
    NP   = DIN(3) + .0001
    IF   (CODE   -  99.1)  290,290,270
270 NPR  =  NP + 3
    NPT  =  3 * NP  + 409
    CALL  PAGE (NPR)
    WRITE (6,280)( DIN(I), I =  410, NPT  )
280 FORMAT(1H0,39X, 6X,4H  WN,12X,6H(SNM),10X,4H DES //
   1          (40X,3E16.6 ))
290 CONTINUE
300 IF  (CODE -  9.0)   340,340,310
C *********************************
C PRINT HEADER AND OUTPUT SYMBOLS*
C *********************************
310 CALL PAGE(70)
    KPAGE=48
    KCOUNT=0
    WRITE (6,320)
320 FORMAT (1H0, 20H PROGRAM  C  OUTPUT    )
    WRITE (6,330)
330 FORMAT(//4X,6H(SNMIC,5X2HW ,6X,7HMACH NO,9X,2HAR,14X,2HAI,14X,2HBR
   1,14X,2HBI,14X,2HCR,14X,2HCI/4X,6H(SNMIN,5X2HWN,9X,1HG,6X,13H-AR/(M
   2ACH NO),3X,13H-AI/(MACH NO),9X,2HT1,14X,2HT2,8X,13H-CR/(MACH NO),3
   3X,13H-CI/(MACH NO)//)
340 CONTINUE
    DO 660 IW =  1,NP
    W  =   DIN(IWW)
    S  =   DIN(IWW+1)
    GO  TO (350,350,360), MDESIR
350 DESIRE    =  DIN(IWW+2)
360 IWW =  IWW + 3
C
    XNOLD=0.0
    XNNEW=0.0
    YOLD=0.0
    XPT = 0.0
    X = 0.0
    XNEW=0.0
    XI = -.001
    H=XI
    W2=W*W
    S2=S*S
    G2=G*G
C ***********************************
C      INITIALIZE 8 EQUATIONS AT X=0.*
C ***********************************
    DO 370 I = 1,8
370 E(I) = 1.E-05
    I = 0
    CALL ADMSET(8,Y(1),YP(1),YOUT(1),TEMP(1),X,H,E(1))
    A1=.5*(G+1.0)
    C2=1.0
    C3=(G+1.0)*(G+1.0)+4.0*W2
    B6=A1*A1+W2
    B7=.25*(A1*(W2-S2)-W2*(G-1.0))/B6
    B8=-.125*W*(G2-1.0+2.0*(W2-S2))/B6
    B91=B7*(A1*A1*A1+W2)-A1*B8*(1.0-A1*W)+.125*(G2-1.0)*(W2-S2)
    B92=A1*A1*(B7*B7-B8*B8)+2.0*A1*W*B7*B8
    B9=(B91+B92)/B6
    B101=.5*A1*(3.0-G)*W*B7+B8*(A1*A1*A1+W2)+.125*(G-1.0)*W*(A1*A1-S2)
    B102=-A1*W*(B7*B7-B8*B8)+2.0*A1*A1*B7*B8
    B10=(B101+B102)/B6
C
    Y(1)=0.0
    Y(2)=0.0
    Y(3)=0.0
    Y(4)=0.0
    Y(5)=0.0
    Y(6) 0.0
    Y(7)=B7
    Y(8)=B8
C
    U2=1.0
    D11=ATAN((B8+W)/B7)
C
    YP(1)=.5*C2
    YP(2)=0.0
    YP(3)=.5*(G+1.0)*(G+1.0)/(C2*C3)
    YP(4)=-W*(G+1.0)/(C2*C3)
    YP(5)=(G+1.0)*(G+1.0)*S2*C2**(1.0/(G-1.0))/(4.0*C3)
    YP(6)=-.5*W*(G+1.0)*S2*C2**(1.0/(G-1.0))/C3
    YP(7)=-B9
    YP(8)=-B10
380 CALL ADMINT
C ******************************************
C INTERPOLATING ON VELOCITY POTENTIAL TABLE*
C ******************************************
390 CALL INTAD(XTABLE(1),YTABLE(1),X,U2,DU2)
    C2=.5*(G+1.0-U2*(G-1.0))
    C=SQRT(C2)
    U=SQRT(U2)
    DC2=-.5*(G-1.0)*DU2
C *******************************************************
C SINGULARITY AT THROAT-FAKE EQUATIONS 7 AND 8 UNTIL PAST*
C *******************************************************
    IF(X-XI)400,400,410
400 B1=U2*(C2-U2)
    B2=A1*U2*DU2/C2
```

```
      B3=W*U2                                                           NOZM2420
      B4=.25*(W2-U*C2**(G/(G-1.0))*S2)                                  NOZM2430
      B5=.25*((G-1.0)*U2*DU2*W)/C2                                      NOZM2440
C                                                                       NOZM2450
      ABD=Y(7)*Y(7)-Y( 8)*Y( 8)                                         NOZM2460
      YP(7)=(B2*Y(7)-B3*Y( 8)-B4)/B1-ABD                                NOZM2470
      YP( 8)=(B3*Y(7)+B2*Y( 8)+B5)/B1-2.0*Y(7)*Y( 8)                    NOZM2480
C                                                                       NOZM2490
  410 YP(1)=.5*(DU2-W*Y(2)/U2)                                          NOZM2500
      YP(2)=.5*W*Y(1)/U2                                                NOZM2510
      D1=-Y(3)*Y(7)+Y(4)*(Y( 8)-W*(.5/U2+2.0/((G+1.0)*(1.0-U2))))       NOZM2520
      D2=S2*C2**(1.0/(G-1.0))*Y(2)/(2.0*W*U)                            NOZM2530
      D3=(-W*Y(2)/U2+DU2*(1.0+(G-1.0)*Y(1)/C2))/(2.0*C2)                NOZM2540
      YP(3)=D1+D2+D3                                                    NOZM2550
C                                                                       NOZM2560
      D4=-Y(4)*Y(7)-Y(5)*(Y( 8)-W*(.5/U2+2.0/((G+1.0)*(1.0-U2))))       NOZM2570
      D5=-S2*C2**(1.0/(G-1.0))*U*((1.0-U2)/(2.0*U2)+Y(1)/U2)/(2.0*W)    NOZM2580
      D6=(W*Y(1)/U2+DU2*(G-1.0)*Y(2)/C2)/(2.0*C2)                       NOZM2590
      YP(4)=D4+D5+D6                                                    NOZM2600
C                                                                       NOZM2610
      D7=S2*C2**(1.0/(G-1.0))/(4.0*U)-Y(5)*Y(7)                         NOZM2620
      D8=Y(6)*(Y( 8)-W*(.5/U2+2.0/((G+1.0)*(1.0-U2))))                  NOZM2630
      YP(5)=D7+D8                                                       NOZM2640
C                                                                       NOZM2650
      ABC=-Y(6)*Y(7)                                                    NOZM2660
      YP(6)=ABC-Y(5)*(Y( 8)-W*(.5/U2+2.0/((G+1.0)*(1.0-U2))))           NOZM2670
C                                                                       NOZM2680
      CALL ADMCOR                                                       NOZM2690
C                                                                       NOZM2700
  420 XMNEW=U/C                                                         NOZM2710
      XNEW=X                                                            NOZM2720
C                                                                       NOZM2730
      IF(XMNEW-DESIRE)430,500,440                                       NOZM2740
C ***********************************************************************NOZM2750
C IF OUR CALCULATED MACH NUMBER IS CLOSE ENOUGH TO DESIRE WE WILL STOP* NOZM2760
C ***********************************************************************NOZM2770
  430 IF(ABS(XMNEW-DESIRE)-7.E-5)500,500,470                            NOZM2780
  440 IF(XMN-XX(KN))450,460,460                                         NOZM2790
  450 XNEW = XX(KN)                                                     NOZM2800
      GO TO 500                                                         NOZM2810
  460 XMOLD = XMNEW                                                     NOZM2820
      XOLD=XNEW                                                         NOZM2830
      GO TO 380                                                         NOZM2840
C ***********************************************************           NOZM2850
C PASSED DESIRED MACH NUMBER                                  *         NOZM2860
C LINEAR INTERPOLATION TO GET X CORRESPONDING TO DESIRED MACH*          NOZM2870
C ***********************************************************           NOZM2880
  470 XPT=XOLD+(XNEW-XOLD)/(XMNEW-XMOLD)*(DESIRE-XMOLD)                 NOZM2890
      XOLD = XNEW                                                       NOZM2900
      XMOLD = XMNEW                                                     NOZM2910
      XNEW = XPT                                                        NOZM2920
      CALL INT4(XTABLE(1),YTABLE(1),XNEW,U2)                            NOZM2930
      C2 = .5*(G+1.0-U2*(G-1.0))                                        NOZM2940
      C = SQRT(C2)                                                      NOZM2950
      U = SQRT(U2)                                                      NOZM2960
      XMNEW = U/C                                                       NOZM2970
      I = I+1                                                           NOZM2980
      IF(I-15)430,480,480                                               NOZM2990
  480 WRITE (6,490)                                                     NOZM3000
  490 FORMAT(T2H0 ITERATION SCHEME DOES NOT CONVERGE AFTER 10 ITERATIONSNOZM3010
     1, SEE PROGRAMMER)                                                 NOZM3020
      GO TO 10                                                          NOZM3030
  500 CALL ADMPAR(XNEW)                                                 NOZM3040
C                                                                       NOZM3050
C                                                                       NOZM3060
C ***********************************************************           NOZM3070
C COMPUTE ADMITTANCE COEFFICIENTS AND PRINT FINAL RESULTS*              NOZM3080
C ***********************************************************           NOZM3090
  510 CALL INT4D (XTABLE(1),YTABLE(1),XNEW,U2,DU2)                      NOZM3100
  520 C2=.5*(G+1.0-U2*(G-1.0))                                          NOZM3110
      C=SQRT(C2)                                                        NOZM3120
      U=SQRT(U2)                                                        NOZM3130
C                                                                       NOZM3140
      DC2=-.5*(G-1.0)*DU2                                               NOZM3150
      D=SQRT(2.0/(G+1.0))*U*C2/((C2/(.5*(G+1.0)))**(1.0*G/(G-1.0)))     NOZM3160
      D9=(1.0-U2)*(G+1.0)                                               NOZM3170
      F3R=YOUT(1)/C2                                                    NOZM3180
      F3I=YOUT(2)/C2                                                    NOZM3190
      XI2R=2.0*YOUT(3)/D9                                               NOZM3200
      XI2I=2.0*YOUT(4)/D9                                               NOZM3210
      XI0R=2.0*YOUT(5)/D9                                               NOZM3220
      XI0I=2.0*YOUT(6)/D9                                               NOZM3230
      ZR=YOUT(7)                                                        NOZM3240
      ZI=YOUT( 8)                                                       NOZM3250
      ER=C2*XI0R-ZR                                                     NOZM3260
      EI=C2*XI0I-ZI                                                     NOZM3270
      FR=U2*C2*XI0R-ZR*U2                                               NOZM3280
      FI=U2*C2*XI0I-ZI*U2-.5*W                                          NOZM3290
      D10=FR*FR+FI*FI                                                   NOZM3300
      AR1= D*(ER*FR+EI*FI)/D10                                          NOZM3310
      AI1= D*(EI*FR-ER*FI)/D10                                          NOZM3320
      IF(S)540,530,540                                                  NOZM3330
  530 G1=W*C2*SQRT(U/(C2**(1.0/(G-1.0))))                               NOZM3340
      GO TO 550                                                         NOZM3350
  540 G1=W*C2*SQRT(U/(C2**(1.0/(G-1.0))))/S                             NOZM3360
  550 BR1= G1*(-FR*XI0I+FI*XI0R)/D10                                    NOZM3370
      BI1= G1*(FR*XI0R+FI*XI0I)/D10                                     NOZM3380
      H1=U*C2*SQRT(2.0/(G+1.0))                                         NOZM3390
      XJR=F3R*ZR-F3I*ZI+.5*(1.0-U2)*XI0R+.5*W*XI2I                      NOZM3400
      XJI=F3I*ZR+F3R*ZI+.5*(1.0-U2)*XI0I-.5*W*XI2R                      NOZM3410
      CR1 = H1*(XJR*FR+XJI*FI)/D10                                      NOZM3420
      CI1 = H1*(XJI*FR-XJR*FI)/D10                                      NOZM3430
C                                                                       NOZM3440
  560 ALPHAR = -AR1/DESIRE                                              NOZM3450
      ALPHAI = -AI1/DESIRE                                              NOZM3460
      CHIR = -CR1/DESIRE                                                NOZM3470
      CHII = -CI1/DESIRE                                                NOZM3480
      TT = (W/(S*SQRT(.5*(G+1.0))))/SQRT(U*(.5*(G+1.0)-U2*(G-1.0)/2.0)**NOZM3490
     1( 1.0/(G-1.0)))                                                   NOZM3500
      T1 = AR1*TT - BI1                                                 NOZM3510
      T2 = AI1*TT + BR1                                                 NOZM3520
C                                                                       NOZM3530
C ******** ******************                                           NOZM3540
C  CASE IS COMPLETED                                                    NOZM3550
C ***************************                                           NOZM3560
C                                                                       NOZM3570
C  FOR S=0  ADM COEF ARE ALPHA FOR 8005  LONGITUDINAL                   NOZM3580
C                                                                       NOZM3590
C  FOR S=N  ADM COEF ARE  T    FOR  993  TRANSVERSE                     NOZM3600
C                                                                       NOZM3610
      IF(S)580,570,580                                                  NOZM3620
```

```
  570 DOUT(IWO+1)=ALPHAR                                                NOZM3630
      DOUT(IWO+101)=ALPHAI                                              NOZM3640
      DOUT(IWO)=W/GEO                                                   NOZM3650
      DOUT(IWO+100)=W/GEO                                               NOZM3660
      GO TO 590                                                         NOZM3670
  580 CONTINUE                                                          NOZM3680
      DOUT(IW+10)=W/GEO                                                 NOZM3690
      DOUT(IW+49)=T1                                                    NOZM3700
      DOUT(IW+79)=T2                                                    NOZM3710
      DOUT(IW+109) = CR1                                                NOZM3720
      DOUT(IW+139) = CI1                                                NOZM3730
  590 IWO =IWO+2                                                        NOZM3740
      IF ( CODE - 9.0)    660,660,600                                   NOZM3750
  600 CONTINUE                                                          NOZM3760
      KKKN=(2*IW)+1                                                     NOZM3770
      IF(NOMEG-KKKN)610,620,620                                         NOZM3780
  610 KKKN=KKKN-1                                                       NOZM3790
  620 KCOUNT=KCOUNT+4                                                   NOZM3800
      IF(KCOUNT-KPAGE)640,640,630                                       NOZM3810
  630 CALL PAGE(70)                                                     NOZM3820
      WRITE (6,330)                                                     NOZM3830
      KCOUNT=0                                                          NOZM3840
      KPAGE=48                                                          NOZM3850
C ********************                                                  NOZM3860
C PRINT FINAL RESULTS                                                   NOZM3870
C ********************                                                  NOZM3880
  640 WRITE (6,650)WC(1),WC(KKKN),DESIRE,AR1,AI1,BR1,BI1,CR1,CI1,S,W,G,ANOZM3890
     1LPHAR,ALPHAI,T1,T2,CHIR,CHII                                      NOZM3900
  650 FORMAT(2(3X,F7.4,2X,F7.4,3X,F7.4,4,6E16.5/1)                      NOZM3910
      FCCPS=(WCONST*WC(KKKN))*12.0                                      NOZM3920
      WRITE (6,670)FCCPS                                                NOZM3930
  660 CONTINUE                                                          NOZM3940
  670 FORMAT(4X,8HFC(CPS)=F10.4//)                                      NOZM3950
      DOUT(205)=DESIRE                                                  NOZM3960
      IF(S) 690,680,690                                                 NOZM3970
  680 DOUT(1)=NP                                                        NOZM3980
      DOUT(101)=NP                                                      NOZM3990
      RETURN                                                            NOZM4000
  690 DOUT(18)=NP                                                       NOZM4010
      DOUT(NP+21)=0.0                                                   NOZM4020
      DOUT(NP+51)=0.0                                                   NOZM4030
      DOUT(NP+81)=0.0                                                   NOZM4040
      DOUT(NP+111)=0.0                                                  NOZM4050
      DOUT(NP+141)=0.0                                                  NOZM4060
      RETURN                                                            NOZM4070
      END                                                               NOZM4080
```

```
$IBFTC VELPOT  LIST,M94
      SUBROUTINE TBLCAL
C TBLCAL SUBROUTINE CALCULATES XX VS U2TBL FROM NOZZLE GEOMETRY
C SIMPSONS RULE IS USED WITH KN INPUT ODD AND CHANGED TO KN/2+1 IN MAIN
C
      LOGICAL  LOGIK,  SL1,  SL2,  EORJ
      COMMON  /PRCLOG/  LOGIK(50), SL1, SL2, EORJ
C
      COMMON  /ABCDF/  EXTRA(100),  ABLOK(600)
     1      ,  XX      , U2TBL    , DESIRE   , RAT      , RAC      , RCC
     2      ,  RCT     , ALFA     , G        , KN       , Y        , YP
     3      ,  YOUT    , TEMP     , E        , XTABLE   , YTABLE   , A
     4      ,  R       , AM       , AP       , AMP      , ZZ       , AMM
     5      ,  AMP2    , CALFA    , CTALFA   , DELAM    , DELTZ    , FKN
     6      ,  G1      , G2       , G3       , G4       , JFLAG1   , KNM1
     7      ,  K       , NN       , PI       , PROD     , RSTA1    , RSTA2
     8      ,  SALFA   , T1       , T2       , T3       , XINT     , XK
     9      ,  ZZ1     , ZZ2      , ZZ3      , A1       , ABC      , ABD
      COMMON  /ABCDF/
     1         A11     , ALPHAI   , ALPHAR   , AR1      , B101     , B102
     2      ,  B10     , B1       , B2       , B3       , B4       , B5
     3      ,  B6      , B7       , B8       , B91      , B92      , B9
     4      ,  BR1     , BR2      , C2       , C3       , CHII     , CHIR
     5      ,  CI1     , CR1      , C        , D10      , D11      , D1
     6      ,  D2      , D3       , D4       , D5       , D6       , D7
     7      ,  D8      , D9       , DC2      , D        , DU2      , EI
     8      ,  ER      , F3I      , F3R      , FI       , FR       , HI
     9      ,  H       , I        , IWO      , IW       , IWW      , J
      COMMON  /ABCDF/
     1         MDESIR  , NK       , NP       , S2       , S        , TT
     2      ,  U2      , U        , W2       , W        , XI0I     , XI0R
     3      ,  XI2I    , XI2R     , XI       , XJI      , XJR      , XMNEW
     4      ,  XMOLD   , XNEW     , XOLD     , XPT      , X        , ZI
     5      ,  ZR
C
      DIMENSION XX(200),U2TBL(200),XTABLE(200),YTABLE(200),ZZ(200)
      DIMENSION  Y(8),YP(8),YOUT(8),TEMP(72),E(8)
      DIMENSION A(200),R(200),AM(200),AP(200),AMP(200)
C
C
      FKN = KN
      KNM1 = KN - 1
      DELAM = 1.0/(FKN+1.0)
      PI = 3.1415927
      ZZ(1)= 0.0
C
      DO 10 J = 1,200
      ZZ(J) = 0.0
      A(J) = 0.0
      R(J) = 0.0
      AMP(J) = 0.0
      AM(J) = 0.0
      XX(J) = 0.0
      U2TBL(J) = 0.0
   10 AP(J) = 0.0
C
      R(1)= RAT
      A(1)=PI*R(1)**2
      U2TBL(1)=1.0
      R(KN) = RAC
      ALFA = ALFA*.01745329
      CALFA=COS(ALFA)
      SALFA=SIN(ALFA)
      CTALFA=CALFA/SALFA
      RSTA1=RAT+RCT*(1.0-CALFA)
      RSTA2=RAC-RCC*(1.0-CALFA)
      ZZ1=RCT*SALFA
      ZZ2=ZZ1+CTALFA*(RSTA2-RSTA1)
      ZZ3=ZZ2+RCC*SALFA
      DELTZ = ZZ3/(FKN-1.0)
C
      JFLAG1=1
C
      DO 80 I = 2,KNM1
      ZZ(I) = ZZ(I-1) + DELTZ
      GO TO (20,40,60),JFLAG1
   20 R(I)=RAT+RCT-SQRT(RCT**2-ZZ(I)**2)
      IF(R(I)-RSTA1)70,70,30
   30 JFLAG1=2
   40 R(I)=RSTA1+(RSTA2-RSTA1)*(ZZ(I)-ZZ1)/(ZZ2-ZZ1)
      IF(R(I) - RSTA2)70,70,50
   50 JFLAG1=3
   60 R(I)=RAC-RCC+SQRT(RCC**2-(ZZ3-ZZ(I))**2)
   70 A(I)=PI*R(I)**2
   80 CONTINUE
C
      ZZ(KN) = ZZ(KNM1) + DELTZ
      A(KN) = PI*RAC**2
      AMM = 1.0+ DELAM
      G1=2.0/(G +1.0)
      G2 =(G - 1.0)/2.0
      G3 = (G + 1.0)/(2.0*G - 2.0)
      G4=1.0/G1
C
      DO 90 J = 1,KN
      AMM = AMM - DELAM
      AM(J) = AMM
      AP(J)=(A(1)/AMM)*(G1*(1.0+G2*AMM**2))**G3
   90 CONTINUE
C
      DO 100 K = 2,KN
      CALL INT4(AP(1),AM(1),A(K),AMP(K))
      AMP2=AMP(K)**2
      U2TBL(K)=(G4*AMP2)/(1.0+G2*AMP2)
  100 CONTINUE
C
      DESIRE = AMP(KN)
      XINT = 0.0
      NN = KN - 2
      K = 1
      XK = SQRT(G1*RAT/RCT)
      PROD = 2.0*XK*DELTZ/3.0
      DO 110 J = 1,NN,2
      T1 = SQRT(U2TBL(J))
      T2 = SQRT(U2TBL(J+1))
      T3 = SQRT(U2TBL(J
      XINT = XINT + PROD   T1+4.0*T2+T3)
      K = K + 1
      XX(K) = -XINT/RAT
      U2TBL(K-1) = (U2TBL(J)
  110 CONTINUE
```

```
BIMPT LONGIT  L*97.*Ma                                                  LONG
      SUBROUTINE  LONG                                                  LONG   10
C                                                                       LONG   20
      COMMON  /ABCDF /  DIMP(6,50)                                      LONG   30
      DIMENSION                                                         LONG   40
     1   EXTRA(100),DISTL(20),DISTM(20),AMIT(100),                      LONG   50
     2   TAB(150,3),  OMG(,50),  ALFR (50),  ALFI(150),  W(2*)          LONG   60
      EQUIVALENCE                                                       LONG   70
     1   (EXTRA(1),GAMMA),    (DIMP(1,1),  U1BAR ) , (EXTRA(2)),  W ( )  LONG   80
     2   (EXTRA(4),RAC),  (EXTRA(5),  ELCM),(EXTRA(6),SOUND,CO),        LONG   90
     3   ( DIMP(1,7),  XLM ),( DIMP(1,8),  ZK ),                        LONG  100
     4   (EXTRA(20),ELROZ),(EXTRA(51),DISTL),(EXTRA(71),DISTM),         LONG  110
     5   (DIMP(1),  EXTRA),                                             LONG  120
     6   (DIMP (51),TAB,OMG),(DIMP(301),ALFR),(DIMP(651),ALFI),         LONG  130
     7    (DIMP(6),AMIT),                                               LONG  140
C                                                                       LONG  150
                                                                        LONG  160
   11 FORMAT (2I8 F7.1, I18 F10.5, 3D9 F10.5, 1D3 F10.5)                LONG  170
   22 FORMAT ( / /  I18 35H RESULTS FOR LONGITUDINAL MODE //21X,THE(CPS) LONG  180
     1 1X,5HOMEGA,8X,5HTAU,MS,1X,6HFREQ )                               LONG  190
   99 FORMAT      ( 1PE  1PE  1PE  1PE ./,5,0,10,7HGAMMA =,E15,0,35,4HE =,E15,0,3LONG  200
     10,7HE =,E,5,0, 7HW =E,5,0 )                                       LONG  210
C                                                                       LONG  220
      CALL  PAGE  (60)                                                  LONG  230
      NINF = 0                                                          LONG  240
      EPS = 1.0E-4                                                      LONG  250
      CALL  INTA  ( DIST, 1), DISTM(1), ELCM, EDIST )                   LONG  260
      CALL  INTGS  ( 0.0, ELCM, X, NINC )                               LONG  270
        CALL  INTA  ( DISTL(1), DISTM(1), X    , UOFX )                 LONG  280
      CALL  INTGS  ( UOFX, SUMM, EPS, NINC )                            LONG  290
      XX = 1.0 - SUMM/(EDIST*ELCM)                                      LONG  300
       UO = XLM / 10.90                                                 LONG  310
      WRITE (6,99),GAM, TAMMA, ZK, XX, XLM                              LONG  320
      WRITE (6,2)                                                       LONG  330
      U1BAR2=U1BAR* U1BAR                                               LONG  340
      T1BAR=1.0-.5*.GAMMA-1.0)*U1BAR2                                   LONG  350
      C1BAR=SQRT(T1BAR)                                                 LONG  360
      ZETA1B=1.5/(T1BAR-GAMMA*U1BAR2* (X-1.0))                          LONG  370
      ASYAR=2.0*C1BAR/(C1BAR**2-U1BAR2)                                 LONG  380
C                                                                       LONG  390
   50 NOMEG = EXTRA(22) + .001                                          LONG  400
      IJ = NOMEG-1                                                      LONG  410
      KK = 1                                                            LONG  420
      IK = 50*IJ + 1                                                    LONG  430
      DO 60 J = 1, IJ                                                   LONG  440
         OMG(KK) = W(J)                                                 LONG  450
         DELT = 0.2*(W(J+1)-W(J))                                       LONG  460
         KK = KK+1                                                      LONG  470
         DO 60  I = 1, 4                                                LONG  480
            OMG(KK) = OMG(KK-1) + DELT                                  LONG  490
            KK = KK + 1                                                 LONG  500
   60 CONTINUE                                                          LONG  510
      OMG(KK) = W(NOMEG)                                                LONG  520
      DO 70 KK = 1, IK                                                  LONG  530
         CALL  INTA  ( AMIT(1), AMIT(51), OMG(KK), ALFR(KK) )           LONG  540
         CALL  INTA  ( AMIT(1), AMIT(651), OMG(KK), ALFI(KK) )          LONG  550
   70    CONTINUE                                                       LONG  560
      DO 120 I = 1, IK                                                  LONG  570
         IF ( I .EQ. 51 .OR. I .EQ. 101 )       CALL  PAGE ( 60 )       LONG  580
      OMEGA=TAB(I,1)                                                    LONG  590
      ALPRE=TAB(I,2)                                                    LONG  600
      ALPIM=TAB(I,3)                                                    LONG  610
      DENB=(C1BAR-U1BAR*ALPRE)**2+U1BAR2*ALPIM*ALPIM                    LONG  620
      BRE=(C1BAR*C1BAR-U1BAR2*(ALPRE*ALPRE+ALPIM*ALPIM))/DENB           LONG  630
      BIM=2.0*C1BAR*U1BAR*ALPIM/DENB                                    LONG  640
      XL= (C-ZK*XX                                                      LONG  650
      ZETAL=ZETA1B*C1BAR/XL                                             LONG  660
      A=2.0*ZETA1B*U1BAR*GAMMA*ZK/OMEGA                                 LONG  670
      B=T1BAR/GAMMA+U1BAR2/GAMMA                                        LONG  680
      C=2.0*U1BAR*C1BAR/GAMMA                                           LONG  690
      D=1.0+ZETAL*ZK/OMEGA                                              LONG  700
      PHI=OMEGA*ASTAR*(1.0-XX)                                          LONG  710
      THETA=2.0*OMEGA*X                                                 LONG  720
      SINPHI=SIN(PHI)                                                   LONG  730
      COSPHI=COS(PHI)                                                   LONG  740
      SINTH=SIN(THETA)                                                  LONG  750
      COSTH=COS(THETA)                                                  LONG  760
      CR=1.0+BRE*COSPHI-BIM*SINPHI                                      LONG  770
      CI=BIM*COSPHI+BRE*SINPHI                                          LONG  780
      DR=1.0-BRE*COSPHI+BIM*SINPHI                                      LONG  790
      DI= -BIM*COSPHI-BRE*SINPHI                                        LONG  800
      ER=1.0-COSTH                                                      LONG  810
      EI=  SINTH                                                        LONG  820
      FR=1.0+COSTH                                                      LONG  830
      FI=-SINTH                                                         LONG  840
      CAPI=(FR/GAMMA1)*(U1BAR*(R-C1BAR*DR)-(FI/GAMMA1)*(U1BAR*CI-C1BAR*DI)LONG  850
     1+(D*B-U1BAR*A)*(ER*CR-EI*CI)+(C1BAR*A-C*D)*(ER*DR-EI*DI)          LONG  860
      CAPJ=(FI/GAMMA1)*(U1BAR*(R-C1BAR*DR)+(FR/GAMMA1)*(U1BAR*CI-C1BAR*DI)LONG  870
     1+(D*B-U1BAR*A)*(ER*CI+EI*CR)+(C1BAR*A-C*D)*(ER*DI+EI*DR)          LONG  880
      CAPK=FR*(B*CR-C*DR)-FI*(B*CI-C*DI)                                LONG  890
      CAPL=FI*(B*CR-C*DR) +FR*(B*CI-C*DI)                               LONG  900
      DENMN=CAPK*CAPK+CAPL*CAPL                                         LONG  910
      CAPM=(CAPK*CAPI+CAPL*CAPJ)/DENMN                                  LONG  920
      CAPN=(CAPK*CAPJ-CAPL*CAPI)/DENMN                                  LONG  930
      SE= (GAMMA*(1.0+ZETAL-ZETA1B)                                     LONG  940
      T=.5*(CAPM+SQRT(CAPM**2+CAPN**2))/CAPM                            LONG  950
      ZN=T/SE                                                           LONG  960
      COSOD=1.0-CAPM/T                                                  LONG  970
      SINOD = CAPN /T                                                   LONG  980
      OMDEL  = ATAN(SINOD/COSOD)                                        LONG  990
      IF (SINOD) 80,110,110                                             LONG1000
   80 IF (COSOD) 100,100,90                                             LONG1010
   90 OMDEL=6.2831853+OMDEL                                             LONG1020
      GO TO 120                                                         LONG1030
  100 OMDEL=3.1415927+OMDEL                                             LONG1040
      GO TO 120                                                         LONG1050
  110 IF (COSOD) 100,120,120                                            LONG1060
  120 DELTA=OMDEL/OMEGA                                                 LONG1070
C     NON-DIMENSIONALIZED RESULTS                                       LONG1080
      TAUMS = DELTA*ELCM/SOUND*89,999999                                LONG1090
      FREQ = OMEGA*SOUND/ELCM*12.0/6.2831853                            LONG1100
      WRITE (6,10)  FREQ, OMEGA, TAUMS, ZN                              LONG1110
  130 CONTINUE                                                          LONG1120
      RETURN                                                            LONG1130
            END                                                         LONG1140
```

```
$IBFTC NTAU    LIST,M94                                                 NTAU
      SUBROUTINE  FFF ( FIN, FOUT, CF, NER, W )                        NTAU  10
C*********************************************************************** NTAU  20
      DIMENSION FIN(1),FOUT(1),A(133),B(133),OMGA(1),XN(1),TAU(1),OMX(1)NTAU  30
     1,HTRINT(1),HTIINT(1)                                             NTAU  40
      DIMENSION XNNEW(41),FCCPS(41),W (1)                              NTAU  50
      COMMON  /ABCDF /  FXTRA(100),  ABLOK(600), A    , B,  XNNEW, FCCPS NTAU  60
      EQUIVALENCE (A(9),OMW),(A(10),OMGA),(A(51),HTRINT),(A(92),HTIINT),NTAU  70
     1(B(9),XNW),(B(10),OMX),(B(51),TAU),(B(92),XN)                    NTAU  80
C                                                                      NTAU  90
C                                                                      NTAU 100
      CONST=FOUT(101)                                                  NTAU 110
      XNMIN=1000.0                                                     NTAU 120
      CALL PAGE(70)                                                    NTAU 130
      CALL DVCHK (KOOOFX)                                              NTAU 140
   10 DO 20 I=1,133                                                    NTAU 150
      A(I) =FIN(I)                                                     NTAU 160
   20 B(I)=0.0                                                         NTAU 170
      NER=IFIX(OMW)                                                    NTAU 180
      XNW=OMW                                                          NTAU 190
      IF(CF-99.0)70,70,30                                              NTAU 200
   30 CALL PAGE (44)                                                   NTAU 210
      WRITE (6,40)                                                     NTAU 220
   40 FORMAT (1H0,50H PROGRAM  F  INPUT  SOLVE FOR N(W) AND TAU(W)     NTAU 230
     1//19X,8H(OMEGA)D,9X,6HHTRINT,10X,6HHTIINT)                       NTAU 240
      DO 50 I =1,40                                                    NTAU 250
   50 WRITE (6,60)OMGA(I),HTRINT(I), HTIINT(I)                         NTAU 260
   60 FORMAT(1H ,10X,3F16.6)                                           NTAU 270
   70 CONTINUE                                                         NTAU 280
      DO 120 I =1,40                                                   NTAU 290
      XN(I)=(HTRINT(I)*HTRINT(I) + HTIINT(I)*HTIINT(I))/(2.0*HTRINT(I)) NTAU 300
      DNOM = XN(I) - HTRINT(I)                                         NTAU 310
      CALL QUAD (DNOM,HTIINT(I),TAU(I))                                NTAU 320
      TAU(I)=(TAU(I)*A(1)*83.333333)/(A(2)*OMGA(I))                    NTAU 330
C A(1)=L,A(2)=CO                                                       NTAU 340
      OMX(I)=OMGA(I)                                                   NTAU 350
      CALL DVCHK  KOOOFX)                                              NTAU 360
       GO TO (80,90),KOOOFX                                            NTAU 370
   80 NER=0                                                            NTAU 380
      GO TO 220                                                        NTAU 390
   90 IF(XN(I))120,120,100                                             NTAU 400
  100 IF(XN(I)-XNMIN) 110,120,120                                      NTAU 410
  110 XNMIN=XN(I)                                                      NTAU 420
      IMIN=I                                                           NTAU 430
  120 CONTINUE                                                         NTAU 440
      DO 130 I=1,100                                                   NTAU 450
  130 FOUT(I) = B(I)                                                   NTAU 460
      IF ( CF-9.0 )          180,180,140                               NTAU 470
  140 CALL PAGE (45)                                                   NTAU 480
      WRITE (6,150)                                                    NTAU 490
  150 FORMAT(1H0,20H PROGRAM  F  OUTPUT  //21X,7HFC(CPS),13X,8H(OMGA)D, NTAU 500
     113X,7HTAU(MS),16X,1HN  )                                         NTAU 510
      DO 160 I=1,40                                                    NTAU 520
      FCCPS(I)=CONST*OMX(I)                                            NTAU 530
  160 WRITE (6,170)FCCPS(I),OMX(I),TAU(I),XN(I)                        NTAU 540
  170 FORMAT(21X,F7.1 ,11X,F10.5,10X,F10.5,10X,F10.5)                  NTAU 550
  180 HTIINT(41) = 0.0                                                 NTAU 560
      HTRINT(41)=0.0                                                   NTAU 570
      XN(41)=0.0                                                       NTAU 580
      OMX(41)=0.0                                                      NTAU 590
      FCCPS(41)=0.0                                                    NTAU 600
  190 FORMAT(36X,5HNMIN=,F10.5,/36X,8HTAU(MS)=,F10.5,/36X,9H(OMEGA)D=,F1NTAU 610
     10.5,/36X,8HFC(CPS)=,F10.1,)                                      NTAU 620
      DO 200 I =1,40                                                   NTAU 630
  200  CALL INT4D(OMX(1),XN(1),OMX(I),SAVNOT,XNNEW(I))                 NTAU 640
      XNNEW(41)=0.0                                                    NTAU 650
       CALL INT4(XNNEW(1),OMX(1),0.0,DOMMIN)                           NTAU 660
      FCMIN=CONST*DOMMIN                                               NTAU 670
      CALL INT4(FCCPS(1),HTRINT(1),FCMIN,HTR1)                         NTAU 680
      CALL INT4(FCCPS(1),HTIINT(1),FCMIN,HTI1)                         NTAU 690
      HTRMIN=(HTR1*HTR1+HTI1*HTI1)/(2.0*HTR1)                          NTAU 700
      DNOM=HTRMIN-HTR1                                                 NTAU 710
      CALL QUAD(DNOM,HTI1,TAUMIN)                                      NTAU 720
      TAUMIN=(TAUMIN*A(1)*83.333333)/(A(2)*DOMMIN)                     NTAU 730
      CALL PAGE (8)                                                    NTAU 740
      WRITE (6,210)                                                    NTAU 750
  210 FORMAT(//21X,58H THE FOLLOWING ARE VALUES INTERPOLATED AT SLOPE OFNTAU 760
     1 N=0.0                          ,/)                             NTAU 770
      WRITE (6,190)HTRMIN,TAUMIN,DOMMIN,FCMIN                          NTAU 780
  220 RETURN                                                           NTAU 790
      END                                                              NTAU 800


$IBFTC QUADR   LIST,M94                                                QUAD
      SUBROUTINE QUAD (A,B,ANGLE)                                      QUAD  10
      IF(B) 10,50,80                                                   QUAD  20
   10 IF(A) 20,30,40                                                   QUAD  30
   20 ROTATE = 3.1415927                                               QUAD  40
      GO TO 110                                                        QUAD  50
   30 ANGLE = 4.7123890                                                QUAD  60
      GO TO 120                                                        QUAD  70
   40 ROTATE = 6.2831853                                               QUAD  80
      GO TO 110                                                        QUAD  90
   50 IF (A) 60,70,70                                                  QUAD 100
   60 ANGLE = 3.1415927                                                QUAD 110
      GO TO 120                                                        QUAD 120
   70 ANGLE = 0.0                                                      QUAD 130
      GO TO 120                                                        QUAD 140
   80 IF(A) 20,90,100                                                  QUAD 150
   90 ANGLE = 1.5707963                                                QUAD 160
      GO TO 120                                                        QUAD 170
  100 ROTATE = 0.0                                                     QUAD 180
  110 ANGLE = ATAN(B/A) + ROTATE                                       QUAD 190
  120 RETURN                                                           QUAD 200
      END                                                              QUAD 210


$IBFTC KORE    LIST,M94                                                KORE
      SUBROUTINE CORE(X,N,CODE)                                        KORE  10
      DIMENSION X(1)                                                   KORE  20
      IF(CODE-500.0)40,10,10                                           KORE  30
   10 CODE=100.0                                                       KORE  40
      CALL PAGE(70)                                                    KORE  50
      WRITE (6,20)                                                     KORE  60
   20 FORMAT(10X,37HINPUT DATA DUMP FOR PROGRAM FAILIER       //)      KORE  70
      WRITE (6,30)(X(I),I=1,N)                                         KORE  80
   30 FORMAT(5X,10(F10.4,2X))                                          KORE  90
   40 RETURN                                                           KORE 100
      END                                                              KORE 110
```

```
$ORIGIN        ALODE.REW

$INCLUDE       JECTOR

$IBFTC INJ     LIST.W94                                                  INJ
      SUBROUTINE  INJCTR                                                 INJ   10
C                                                                        INJ   20
      COMMON  /PROLOG/  LOGIK(38), HEAD(12), SL1, SL2, EORJ              INJ   30
      COMMON  /JECTOR/  EJDATA( 9600)                                    INJ   40
C                                                                        INJ   50
      EQUIVALENCE  (LOGIK(5),ERUN) , (LOGIK(10),JRUN ) , (LOGIK(9),IRUN)INJ   60
      LOGICAL  LOGIK, ERUN, JRUN, SL1, SL2, EORJ, IRUN                   INJ   70
      DIMENSION  HEAD(12)                                                INJ   80
C                                                                        INJ   90
      IF ( SL1 )              GO TO 20                                   INJ  100
      DO 10 I = 1, 9600                                                  INJ  110
   10    EJDATA(I) = 0.0                                                 INJ  120
      SL1 = .TRUE.                                                       INJ  130
      GO TO 30                                                           INJ  140
C                                                                        INJ  150
   20 READ (13)  EJDATA                                                  INJ  160
      BACKSPACE  13                                                      INJ  170
   30 CALL  AS13B  ( EJDATA, HEAD, NE )                                  INJ  180
      IF ( NE .NE. 1 )         CALL  EXIT                                INJ  190
      WRITE (13)  EJDATA                                                 INJ  195
      BACKSPACE  13                                                      INJ  196
      IF ( .NOT. JRUN )       GO TO 40                                   INJ  200
      CALL  JJJ                                                          INJ  210
   40 IF ( ERUN .OR. IRUN )   CALL  INJCTS                               INJ  220
C                                                                        INJ  230
   50 RETURN                                                             INJ  240
          END                                                            INJ  250


$IBFTC JECT    LIST.W94                                                  JECT
      SUBROUTINE JJJ                                                     JECT  10
C                                                                        JECT  20
C  ************  DECK MODIFIED 20 AUG 67     ********************        JECT  30
C                                                                        JECT  40
      DIMENSION  THET1(1450), R1(1450), TMU(1000)                        JECT  50
      DIMENSION            NELE(1000),X(1000),Y(1000),NTYPEE(1000),R(22)JECT  60
     1,THETA(182),AS(20),WTE(1000),AAX(1000),AAF(1000),NBAND(1000),XMRE(JECT  70
     21000),XMUTOT(20),AXTY(200),AFTY(200),XX(1000),YY(1000)             JECT  80
      COMMON  /JECTOR/  DATA ( 9600)                                     JECT  90
C                                                                        JECT 130
      EQUIVALENCE (DATA(5),XM), (DATA(6),XN)                             JECT 140
      EQUIVALENCE (DATA(1021),THET1),(DATA(2471),R1),(DATA(3921),TMU)    JECT 150
      EQUIVALENCE (NTYPEE,DATA(2001)),(AXTY,DATA(1001)),(AFTY,DATA(1201)JECT 160
     1), (DATA(1921),XX),(DATA(2921),YY),(DATA(4921),XMUTOT)             JECT 170
C                                                                        JECT 180
      NERROR=0                                                           JECT 200
      SECT=DATA(5)                                                       JECT 210
      WT=DATA(9575)                                                      JECT 220
      RINJ=DATA(9574)                                                    JECT 230
      XMR=DATA(9575)                                                     JECT 240
      FFC=DATA(9576)                                                     JECT 250
      DFFC=DATA(9577)                                                    JECT 260
      NT=DATA(9570)+.0001                                                JECT 270
      POLAR=DATA(9572)                                                   JECT 280
      NE=DATA(9571)+.0001                                                JECT 290
      ROX=DATA(9579)                                                     JECT 300
      ROF=DATA(9580)                                                     JECT 310
      ERUMAX=5.0                                                         JECT 320
      PXADJ=1.0                                                          JECT 330
      PFADJ=1.0                                                          JECT 340
      CDX=DATA(9585)                                                     JECT 350
      CDF=DATA(9586)                                                     JECT 360
      XFC=DATA(9591)                                                     JECT 370
      DXFC=DATA(9590)                                                    JECT 380
      PFFC = DATA(9589)/100.0                                            JECT 390
      PXFC = DATA(9590)/100.0                                            JECT 400
      IF (XM .EQ. 0.0 ) XM = 70.0                                        JECT 410
      IF ( XN .EQ. 0.0 ) XN = 180.0                                      JECT 420
      K=910                                                              JECT 430
        [illegible] 0.1831853/[illegible]                                JECT 440
      [illegible] (A(322))20,30,30                                       JECT 450
      [illegible] 3.1415926/SECT                                         JECT 460
      GO TO 40                                                           JECT 470
   30 ROTATE=0.0                                                         JECT 480
   40 IF(POLAR [illegible] .0 .                                          JECT 490
      IF(SECT-1.0)180,40,60                                              JECT 500
   60 DO 70 I=1,NE                                                       JECT 510
      KK=K+1                                                             JECT 520
      NELE(I)=DATA(KK)+.0001                                             JECT 530
      X(I)=DATA(KK+1)                                                    JECT 540
      Y(I)=DATA(KK+2)                                                    JECT 550
      NTYPEE(I)=DATA(KK+3)+.0001                                         JECT 560
      SAVE=SQRT(X(I)*X(I)+Y(I)*Y(I))                                     JECT 570
      Y(I)=(ATAN(Y(I)/X(I)))+ROTATE                                      JECT 580
      X(I)=SAVE                                                          JECT 590
      K=K+4                                                              JECT 600
   70 CONTINUE                                                           JECT 610
      GO TO 270                                                          JECT 620
   80 DO 170 I=1,NE                                                      JECT 630
      KK=K+1                                                             JECT 640
      NELE(I)=DATA(KK)+.0001                                             JECT 650
      X(I)=DATA(KK+1)                                                    JECT 660
      Y(I)=DATA(KK+2)                                                    JECT 670
      NTYPEE(I)=DATA(KK+3)+.0001                                         JECT 680
C     THE FOLLOWING SEGMENT OF CODING REPLACES SUBROUTINE MOVXY...       JECT 690
         IF ( Y(I) )          90,130,140                                JECT 700
   90    IF ( X(I) )         100,110,120                                JECT 710
  100    ROTATE = 3.1415927                                             JECT 720
         GO TO 160                                                      JECT 730
  110    X(I) = 1.0E-15                                                 JECT 740
  120    ROTATE = 6.2831853                                             JECT 750
         GO TO 160                                                      JECT 760
  130    ROTATE = 0.0                                                   JECT 770
         GO TO 160                                                      JECT 780
  140    IF ( X(I) )         100,150,130                                JECT 790
  150    X(I) = 1.0E-15                                                 JECT 800
         ROTATE = 0.0                                                   JECT 810
  160 SAVE = SQRT ( X(I)*X(I)+Y(I)*Y(I) )                               JECT 820
      SAVE=SQRT(X(I)*X(I)+Y(I)*Y(I))                                     JECT 830
      Y(I)=(ATAN(Y(I)/X(I)))+ROTATE                                      JECT 840
      X(I)=SAVE                                                          JECT 850
      K=K+4                                                              JECT 860
  170 CONTINUE                                                           JECT 870
      GO TO 270                                                          JECT 880
```

```
  180 IF(SECT-1.0)190,190,200
  190 ROTATE=0.0
  200 DO 210 I=1,NE
      KK=K+1
      NELE(I)=DATA(KK)+.0001
      X(I)=DATA(KK+1)
      Y(I)=(DATA(KK+2)*.0174532)+ROTATE
      NTYPEE(I)=DATA(KK+3)+.0001
      K=K+4
  210 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     CALCULATING AREAS OF ELEMENTS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  220 NN=4944
      DO 280 I=1,NT
      KK=NN+1
      NST=DATA(KK)+.0001
      KK1=KK+1
      JJ=KK1
      NX=DATA(KK1)+.0001
      AXTY(NST)=0.0
      IF  X)230,250,230
  230 DO 240 JX=1,NX
      JJ=KK1+JX
      AXTY(NST)=AXTY(NST)+(.7853981*(DATA(JJ)*DATA(JJ)))
  240 CONTINUE
  250 NN1=JJ+1
      NN=NN1
      NF=DATA(NN1)+.0001
      AFTY(NST)=0.0
      IF(NF)260,280,260
  260 DO 270 JF=1,NF
      NN=NN1+JF
      AFTY(NST)=AFTY(NST)+(.7853981*(DATA(NN)*DATA(NN)))
  270 CONTINUE
  280 CONTINUE
      AXTOT=0.0
      AFTOT=0.0
      DO 290 I=1,NE
      NN=NTYPEE(I)
      AAX(I)=AXTY(NN)
      AAF(I)=AFTY(NN)
      AXTOT=AXTOT+AAX(I)
      AFTOT=AFTOT+AAF(I)
  290 CONTINUE
      AXTOT=AXTOT*SECT
      AFTOT=AFTOT*SECT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     CALCULATING RADII AND ANGLE BOUNDRIES
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  300 NPTS=180.0/SECT
      TPS=NPTS
      DELTH=THFNL/TPS
      AREA = 3.141592 * RINJ**2
      ASECT = AREA/(XN*XN)
      R(1) = 0.0
      ANP = ASECT * 180.0 / 3.141592
      DO 310 I = 2,21
      R(I) = SQRT ( ANP + R(I-1)**2)
  310 CONTINUE
      R(22) = 0.0
      XNUM=0.0
      NPTS1=NPTS+1
      DO 320 J=2,NPTS1
      XNUM=XNUM+1.0
      THETA(J)=DELTH*XNUM
  320 CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     CALCULATING WEIGHT FLOW
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WFT1=WT/(XMR+1.0)
      WXT1=WT-WFT1
      WFT2=WFT1
      WXT2=WXT1
      IF(PFFC)340,340,330
  330 WFT2=WFT1*(1.0-PFFC)
  340 IF(PXFC)360,360,350
  350 WXT2=WXT1*(1.0-PXFC)
  360 SIGMEN=WT/(3.141592*RINJ*RINJ)
  370 AXTMAX=AXTOT
      AXTMIN=AXTOT
      AFTMAX=AFTOT
      AFTMIN=AFTOT
  380 AFFC=0.0
      AXFC=0.0
      IF(DFFC)400,400,390
  390 AFFC=FFC*.785398*DFFC*DFFC
      AFTMIN=AFTMIN-AFFC
      AFTMAX=AFTMAX-AFFC
  400 IF(DXFC)420,420,410
  410 AXFC=XFC*.785398*DXFC*DXFC
      AXTMIN=AXTMIN-AXFC
      AXTMAX=AXTMAX-AXFC
  420 AXNOM=(AXTMAX+AXTMIN)/2.0
      AFNOM=(AFTMAX+AFTMIN)/2.0
      IF(DFFC+DXFC)440,440,430
  430 WXT2=(WXT2*AXNOM)/(AXNOM+AXFC)
      WFT2=(WFT2*AFNOM)/(AFNOM+AFFC)
  440 XMR1=WXT2/WFT2
  450 ETOF=0.0
      ETFF=0.0
      QX=WXT2/AXTOT
      QF=WFT2/AFTOT
      CALL DVCHK (K000FX)
  460 DO 480 I=1,NE
      NBAND(I)=0
      WFE1=AAF(I)*QF
      WXE1=AAX(I)*QX
      WTE(I)=WFE1+WXE1
      XMRE(I)=WXE1/WFE1
      ETOF=ETOF+WXE1
      ETFF=ETFF+WFE1
      CALL DVCHK (K000FX)
       GO TO (470,480),K000FX
  470 NBAND(I)=-1
  480 CONTINUE
      ETOF=SECT*ETOF
      ETFF=SECT*ETFF
      WMRELM=ETOF/ETFF
      AXXTT=AXTOT+AXFC
      AFFTT=AFTOT+AFFC
```

```
      WFT2 = ETFF                                                       JECT2110
      WXT2 = ETOF                                                       JECT2120
     WFT1=WFT2/(1.0-PFFC)                                               JECT2130
     WXT1=WXT2/(1.0-PXFC)                                               JECT2140
      WFT1 = WFT1*AFFTT/AFTOT                                           JECT2150
      WXT1 = WXT1*AXXTT/AXTOT                                           JECT2160
     XMRINJ=WXT1/WFT1                                                   JECT2170
     D= ((WXT1/(CDX*AXXTT))**2)*2.2360248                               JECT2180
     WTOT=WXT1+WFT1                                                     JECT2190
     DPX1=D/ROX                                                         JECT2200
     D= ((WFT1/(CDF*AFFTT))**2)*2.2360248                               JECT2210
     DPF1=D/ROF                                                         JECT2220
      TOFCF = WXT1*(1.0 - PXFC)                                         JECT2230
      TOFCF = (TOFCF*(AXXTT - AXFC))/AXXTT                              JECT2240
      TOFCF = WXT1 - TOFCF                                              JECT2250
      TFFC = WFT1*(1.0 - PFFC)                                          JECT2260
      TFFC = (TFFC*(AFFTT - AFFC))/AFFTT                                JECT2270
      TFFC = WFT1 - TFFC                                                JECT2280
     VINJX=SQRT((9273.6*DPX1 /ROX))                                     JECT2290
     VINJF=SQRT((9273.6*DPF1 /ROF))                                     JECT2300
     XXY=TPS*SIGMEN*SECT                                                JECT2310
490  WRITE (6,820)                                                      JECT2320
     GO TO 500                                                          JECT2330
500  WRITE (6,760)AXTOT,AFTOT,AXFC,AFFC,AXXTT,AFFTT,DPX1,DPF1           JECT2340
     WRITE (6,770)WTOT,XMRELM,XMRINJ,FTOF,ETFF,TOFCF,TFFC, WXT1, WFT1 ,JECT2350
    1VINJX,VINJF                                                        JECT2360
     PFFC=PFFC*100.0                                                    JECT2370
     PXFC=PXFC*100.0                                                    JECT2380
510  WRITE (6,830)NT,NT,NE,CDX                                          JECT2390
     WRITE (6,780)CDF,PFFC,PXFC,DFFC,DXFC,FFC,XFC,ROX,ROF               JECT2400
     CALL PAGE(70)                                                      JECT2410
     WRITE (6,810)                                                      JECT2420
     NPGE=49                                                            JECT2430
     DO 540 I=1,NE                                                      JECT2440
     IF(NPGE-I)520,520,530                                              JECT2450
520  CALL PAGE(70)                                                      JECT2460
     NPGE=NPGE+48                                                       JECT2470
     WRITE (6,810)                                                      JECT2480
530  DEGREE=Y(I)/.0174532                                               JECT2490
     B=X(I)*SIN(Y(I))                                                   JECT2500
     A=X(I)*COS(Y(I))                                                   JECT2510
     WRITE (6,800)NELE(I),NTYPEE(I),X(I),DEGREE,A,B                     JECT2520
540  CONTINUE                                                           JECT2530
     CALL PAGE(70)                                                      JECT2540
     NPGE=46                                                            JECT2550
     JCONT=0                                                            JECT2560
     WRITE (6,840)                                                      JECT2570
     NN=4944                                                            JECT2580
     DO 670 I=1,NT                                                      JECT2590
     KK=NN+1                                                            JECT2600
     NST=DATA(KK)+.0001                                                 JECT2610
     KK1=KK+1                                                           JECT2620
     JJ=KK1                                                             JECT2630
     NX=DATA(KK1)+.0001                                                 JECT2640
     JCONT=JCONT+NX+1                                                   JECT2650
     IF(NPGE-JCONT)550,550,560                                          JECT2660
550  CALL PAGE(70)                                                      JECT2670
     WRITE (6,840)                                                      JECT2680
     JCONT=0                                                            JECT2690
560  AXMAX=0.0                                                          JECT2700
     AFMAX=0.0                                                          JECT2710
     WFE1=0.0                                                           JECT2720
     WXE1=0.0                                                           JECT2730
     WRITE (6,850)NST,NX                                                JECT2740
     IF(NX)570,590,570                                                  JECT2750
570  DO 580 JX=1,NX                                                     JECT2760
     JJ=KK1+JX                                                          JECT2770
     A=.7853981*DATA(JJ)*DATA(JJ)                                       JECT2780
     B=A*QX                                                             JECT2790
     WRITE (6,860)DATA(JJ),A,B                                          JECT2800
     AXMAX=AXMAX+A                                                      JECT2810
     WXE1=WXE1+B                                                        JECT2820
580  CONTINUE                                                           JECT2830
590  NN1=JJ+1                                                           JECT2840
     NN=NN1                                                             JECT2850
     NF=DATA(NN1)+.0001                                                 JECT2860
     JCONT=JCONT+NF+2                                                   JECT2870
     IF(NPGE-JCONT)600,600,610                                          JECT2880
600  CALL PAGE (70)                                                     JECT2890
     WRITE (6,840)                                                      JECT2900
     JCONT=0                                                            JECT2910
610  WRITE (6,870)NF                                                    JECT2920
     IF(NF)620,640,620                                                  JECT2930
620  DO 630 JF=1,NF                                                     JECT2940
     NN=NN1+JF                                                          JECT2950
     A=.7853981*DATA(NN)*DATA(NN)                                       JECT2960
     B=A*QF                                                             JECT2970
     WRITE (6,880)DATA(NN),A,B                                          JECT2980
     AFMAX=AFMAX+A                                                      JECT2990
     WFE1=WFE1+B                                                        JECT3000
630  CONTINUE                                                           JECT3010
640  WTOT1=WXE1+WFE1                                                    JECT3020
     IF(WFE)660,650,660                                                 JECT3030
650  WRITE (6,900)AXMAX,AFMAX,WTOT1                                     JECT3040
     GO TO 670                                                          JECT3050
660  XMR1=WXE1/WFE1                                                     JECT3060
     WRITE (6,890)AXMAX,AFMAX,WTOT1,XMR1                                JECT3070
670  CONTINUE                                                           JECT3080
680  READ  (13)  DATA                                                   JECT3090
     BACKSPACE  13                                                      JECT3100
     LINKNT = 0                                                         JECT3110
     AJS = XM*XN/WT                                                     JECT3120
     DO 700 J = 1,NE                                                    JECT3130
     TMU(J) = WTE(J)*AJS                                                JECT3140
     IF (LINKNT .GT. 0) GO TO 690                                       JECT3150
     CALL PAGE(70)                                                      JECT3160
     WRITE (6,910)                                                      JECT3170
     WRITE (6,930)                                                      JECT3180
     LINKNT = 50                                                        JECT3190
690  WRITE(6,920) J, X(J), Y(J), TMU(J)                                 JECT3200
     LINKNT = LINKNT - 1                                                JECT3210
700  CONTINUE                                                           JECT3220
     DO  705  I = 1, NE                                                 JECT3222
        XX(I) = X(I)                                                    JECT3224
705     YY(I) = Y(I)                                                    JECT3225
     WRITE  (13)       DATA                                             JECT3230
     BACKSPACE  13                                                      JECT3240
     XMUMAX=0.0                                                         JECT3250
     AJS = AJS*SECT/XN                                                  JECT3260
     DO 730 J=1,20                                                      JECT3270
     TOTW=0.0                                                           JECT3280
```

```
      XXX = XXV * ASECT                                                 JECT3290
      J=J                                                               JECT3300
      DO 710 I=1,NE                                                     JECT3310
      IF (X(I) .LE. R(J+1)) .AND. X(I) .GT. R(J)) TOTW = TOTW + WTE(I)  JECT3320
  710 CONTINUE                                                          JECT3330
      XMUTOT(J) = TOTW*AJS                                              JECT3340
      IF(XMUTOT(J)-XMUMAX)730,730,720                                   JECT3350
  720 XMUMAX=XMUTOT(J)                                                  JECT3360
  730 CONTINUE                                                          JECT3370
      CALL PAGE(70)                                                     JECT3380
      WRITE (6,790)                                                     JECT3390
      DO 740 J=1,20                                                     JECT3400
      XXMAX=XMUTOT(J)/XMUMAX                                            JECT3410
      WRITE (6,7-0)R(J+1),XMUTOT(J),XXMAX                               JECT3420
  740 CONTINUE                                                          JECT3430
      RETURN                                                            JECT3440
  750 FORMAT(32X,F6.3,14X,F7.3,13X,F6.4)                                JECT3450
  760 FORMAT(//, 4,28HA....PROPELLENT ORFICE AREAS,//,9X,34HELEMENT TOTAJECT3460
     1L OXIDIZER AREA      =,F11.8,8H SQ. IN.,15X,30HELEMENT TOTAL FUEL JECT3470
     2AREA      =,F11.8,8H SQ. IN.,//,9X,34HTOTAL OXIDIZER FILM COOLING AJECT3480
     3REA =,F11.8,8H SQ. IN.,15X,30HTOTAL FUEL FILM COOLING AREA =,F11.8JECT3490
     4,8H SQ. IN.,//,9X,34HINJECTOR TOTAL OXIDIZER AREA      =,F11.8,8H SQJECT3500
     5. IN.,15X,30HINJECTOR TOTAL FUEL AREA      =,F11.8,8H SQ. IN.,//,5XJECT3510
     6,54HB....INJECTOR PRESSURE DROPS FOR ABOVE INJECTOR DESIGN,//,9X, JECT3520
     724HOXIDIZER PRESSURE DROP =,F6.1,3HPSI,35X,20HFUEL PRESSURE DROP =JECT3530
     8,F6.1,4H PSI,///,5X,71HC....PROPELLANT FLOWS AND INJECTOR VELOCITIJECT3540
     9ES FOR ABOVE INJECTOR DESIGN,/)                                   JECT3550
  770 FORMAT(/,48X,19HTOTAL WEIGHT FLOW =,F6.1,7H LB/SEC,/,42X,39HAVERAGJECT3560
     1E MIXTURE RATIO OF THE ELEMENTS =,F7.3,/,41X,40HOVERALL MIXTURE RAJECT3570
     2TIO FOR THE INJECTOR =,F7.3,///,9X,34HELEMENT TOTAL OXIDIZER FLOW  JECT3580
     3    =,F6.1,7H LB/SEC,2,X,26HELEMENT TOTAL FUEL FLOW  =,F6.1,7H LB/JECT3590
     4SEC,/,9X,34HTOTAL OXIDIZER FILM COOLING FLOW =,F6.1,7H LB/SEC,21X,JECT3600
     526HTOTAL FUEL FILM COOLING  =,F6.1,7H LB/SEC,/,9X,34HINJECTOR TOTAJECT3610
     6L OXIDIZER FLOW      =,F6.1,7H LB/SEC,21X,25HINJECTOR TOTAL FUEL FLJECT3620
     7OW=,F7.1,7H LB/SEC,/,9X,37HOXIDIZER OVERALL INJECTION VELOCITY =,FJECT3630
     86.1,7H FT/SEC,18X,33HFUEL OVERALL INJECTION VELOCITY =,F6.1,7H FT/JECT3640
     9SEC,//)                                                           JECT3650
  780 FORMAT(9X,23HFUEL LOSS COEFFICIENT =,F5.3,/,9X,27HPERCENT FUEL FILJECT3660
     1M COOLING =,F5.1,/,9X,31HPERCENT OXIDIZER FILM COOLING =,F5.1,/,9XJECT3670
     2,38HDIAMETER OF FUEL FILM COOLING ORFICE =,F7.5,7SH IN. (NOTE,,THIJECT3680
     3S MIGHT BE AN EQUIVALENT  DIAMETER FOR MULTIPLE-ROW COOLING),/,9X,JECT3690
     442HDIAMETER OF OXIDIZER FILM COOLING ORFICE =,F7.5,21H IN. (SEE ABJECT3700
     5OVE NOTE),/,9X,51HNUMBER OF FUEL FILM COOLING ORIFICES PER INJECTOJECT3710
     6R =,F5.0,/,9X,54HNUMBER OF OXIDIZER FILM COOLING ORFICES PER INJECJECT3720
     7TOR =,F5.0,/,9X,18HOXIDIZER DENSITY =,F7.2,4H PCF,/,9X,14HFUEL DENJECT3730
     8SITY =,F6.1,4H PCF.///)                                           JECT3740
  790 FORMAT(///,3X,114HSECTION 4..RESULTANT FLOW DISTRIBUTION (MU) AS JECT3750
     1A FUNCTION OF ONLY THE RADIUS. I.E.. AVERAGE OF MU IN A RADIAL BAJECT3760
     2D.     ///,32X,6HRADIUS,15X,5HMU(R),11X,13HMU(R)/MU(MAX),/,34X,5HINJECT3770
     3.,//)                                                             JECT3780
  800 FORMAT(13X,I3,18X,I2,4X,4(10X,F10.5))                             JECT3790
  810 FORMAT(//,2X, 49HSECTION 2.... ELEMENT LOCATION AND INJECTION TYPEJECT3800
     1,///,11X, 7HELEMENT,15X, 4HTYPE,17X,1HR,17X, 5HTHETA,17X,1HX,19X,1JECT3810
     2HY,/,13X, 3HNO.,18X, 3HNO.,14X, 8H(INCHES),12X, 8H(DEGREE),12X, 8HJECT3820
     3(INCHES),12X,8H(INCHES),//)                                       JECT3830
  820 FORMAT(///,3X, 74HSECTION 1....MISCELLANEOUS INFORMATION FOR INJECJECT3840
     1TOR DESIGNED BY PROJECTS        ,/)                               JECT3850
  830 FORMAT(//,5X, 46HD.....INPUT INFORMATION USED IN COMPUTATIONS  ,/JECT3860
     1/,3X, 23HTOTAL PROPELLENT FLOW =,F6.1,7H LB/SEC,/,9X, 59HTOTAL NUMJECT3870
     2BER OF ELEMENT TYPES (SYMMETRICAL SECTION ONLY) = ,I3,/,9X, 56HTOTJECT3880
     3AL NUMBER OF ELEMENTS (SYMMETRICAL SECTION ONLY) = ,I4,/,9X, 27HOXJECT3890
     4IDIZER LOSS COEFFICIENT =,F5.3,)                                  JECT3900
  840 FORMAT(//,2X, 78HSECTION 3.....TYPE DESCRIPTION,ORIFICE AREA, PROPEJECT3910
     1LLANT FLOW, AND MIXTURE RATIO,//,3X, 8HTYPE   0-------- OXIDIZER JECT3920
     2ORIFICE DATA ------00------- -- FUEL ORIFICE DATA ------- 0,//,93X,JECT3930
     3 5HTOTAL,5X, 5HTOTAL,5X, 5HTOTAL,4X, 7HMIXTURE,/,,2X, 6HNUMBER,34XJECT3940
     4, 6HNUMBER,33X, 8HOXIDIZER,4X, 4HFUEL,3X,15HPROPELLENT,3X, 5HRATIOJECT3950
     5,/,14X,2HOF,5X, 8HDIAMETER,4X, 4HAREA,6X, 4HFLOW,7X,2HOF,5X, 8HDIAJECT3960
     6METER,4X, 4HAREA,6X, 4HFLOW,6X, 4HAREA,6X, 4HAREA,4X, 9HFLOW RATE,JECT3970
     7/,11X, 7HORIFICE,6X,3HIN.,5X, 7HSQ. IN.,4X, 6HLB/SEC,3X, 8HORIFICEJECT3980
     85,5X, 3HIN.,4X, 7HSQ. IN.,4X, 6HLB/SEC,3X, 7HSQ. IN.,3X, 7HSQ. IN.JECT3990
     9,4X, 6HLB/SEC,//,)                                                JECT4000
  850 FORMAT(3X,I3,7X,I2)                                               JECT4010
  860 FORMAT(22X,F6.4,3X,F8.6,2X,F8.6)                                  JECT4020
  870 FORMAT(56X,I2)                                                    JECT4030
  880 FORMAT(62X,F6.4,3X,F8.6,2X,F8.6)                                  JECT4040
  890 FORMAT(91X,F8.5,2X,F8.5,2X,F8.4,3X,F7.4)                          JECT4050
  900 FORMAT(91X,F8.5,2X,F8.5,2X,F8.3,2X,8HINFINITY)                    JECT4060
  910 FORMAT (//'10X15HELEMENT RESULTS )                                JECT4070
  920 FORMAT (32X15,11XF10.3,10XF10.4,11XF10.4 )                        JECT4080
  930 FORMAT (//31X7HELEMENT14X6HRADIUS15X5HANGLE11X12HDISTRIBUTION/90X JECT4090
     111HCOEFFICIENT/34X3HNO.35X7HRADIANS15X2HMU// )                    JECT4100
                          END                                           JECT4110
```

```
$IBFTC INJDS   LIST,NOA                                                 INJD
      SUBROUTINE  INJDIS                                                INJD  10
C                                                                       INJD  20
C  ************  DECK MODIFIED 20 AUG 67     ********************       INJD  30
C                                                                       INJD  40
      REAL IPP,IPR,IPT,INTEG,IPX                                        INJD  50
      INTEGER DSCRB, TIME                                               INJD  60
      LOGICAL  LOGIK,  SL1,  SL2,  EORJ, IRUN                           INJD  70
      COMMON  /PROLOG/  LOGIK(90), SL1, SL2, EORJ                       INJD  80
      COMMON  /JECTOR/  DATA ( 9600)                                    INJD  90
      COMMON   GAMMA, RW,  WC,  AVN,  BVN,  CVNR,  CVNI,  CE,  CI       INJD 110
      DIMENSION                                                         INJD 120
     1    U     (1000),  AVN    (  30),  BVN    (  30),  CVNR   (  30), INJD 130
     2    CVNI  (  30),  RR     (1000),  THETA (1000),  FIRST (  30),  INJD 140
     3    SECOND(  30),  Z      (   2),                  WC    (  30),  INJD 150
     4     IPP(50), OPP(50), IPR(50), OPR(50), IPT(50), OPT(50),        INJD 160
     5    X     (1000),  Y     (1000)                                   INJD 170
      EQUIVALENCE  ( IRUN, LOGIK(9) ),                                  INJD 180
     1  (E1     ,DATA(1)),                    (XN    ,DATA(3)),          INJD 190
     2  (XR     ,DATA(4)),(V      ,DATA(6)),(SVN   ,DATA(8)),           INJD 200
     3  (POC    ,DATA(13)),                   (Z1    ,DATA(15)),          INJD 210
     4  (RR, A, DATA(1921) ),(THETA, Y, DATA(2921)),                    INJD 220
     5  (U   ,DATA(3921)),                                              INJD 230
     6                                       (IPP   ,DATA(20)),         INJD 240
     7  (OPP   ,DATA(70)),(IPR ,DATA(120)), (OPR ,DATA(170)),           INJD 250
     8  (IPT ,DATA(220)),(OPT ,DATA(270)),(TFLP,DATA(9596)),            INJD 260
     9  (TFLR,DATA(9597)),(TFLT,DATA(9598))                             INJD 270
C                                                                       INJD 280
   10 FORMAT (//12X30HRESULTS OF DESCRIBING FUNCTION // 29X10HELEMENT   INJD 290
     1 6HRADIUS4X5HANGLE3X10HFRACTIONAL /13X5HOMEGA7X5HNO.7X3H   7X3HRADINJD 300
     2 4X9HFLOW-RATE10X2HFP12X2HFR12X2HFT // )                          INJD 310
   20 FORMAT (10XF9.4,I9,F11.3,F10.4,F11.5,2X3F14.6 )                   INJD 320
   30 FORMAT (//  12X41HRESULTS OF INJECTION DISTRIBUTION EFFECTS //    INJD 330
     14X5HOMEGA6X5HAVN8X5HBVN7X5HCVN7X5HCVN / 48X3(6X4HREAL),6X4HIMAG/  INJD 340
     2/ )                                                               INJD 350
   40 FORMAT (39X5F10.4 )                                               INJD 360
   50 FORMAT ( 44X5HALL  4F10.4 )                                       INJD 370
   60 FORMAT ( // 5X39HINPUT TO INJECTION DISTRIBUTION PROGRAM//10X9HCONINJD 380
     1STANTS//14X10HNUMBER OF OMEGAS = I3//14X20HNUMBER OF ELEMENTS =I5,INJD 39
     1 13H FOR EACH OF I4, 20H SYMMETRIC SECTIONS.                      INJD 391
     2//14X22HRADIAL DIVISIONS(XN) = F5.0//14X24HANGULAR DIVISIONS (XN) INJD 400
     3=F5.0//14X27HACOUSTIC MODE NUMBER(SVN) =F7.4//14X30HORDER OF BESSEINJD 410
     4L FUNCTIONS(V) = F3.0//14X17HINJECTOR RADIUS =F8.3,5H, IN.//14X32HINJD 420
     5RATIO OF SPECIFIC HEATS(GAMMA) =F7.4//14X39HMAXIMUM PRESSURE AMPLIINJD 430
     6TUDE RATIO(POO) =F7.3///14X39HTRANSFER FUNCTIONS FOR LINEAR OPERATINJD 440
     7ION//20X16HPRESSURE(TFLP) =F7.3//20X23HRADIAL VELOCITY(TFLR) =F7.3INJD 450
     8//20X27HTANGENTIAL VELOCITY(TFLT) =F7.3 )                         INJD 460
   70 FORMAT (// 10X17HINPUT FREQUENCIES )                              INJD 470
   80 FORMAT (/ 7X 5F20.4 )                                             INJD 480
   90 FORMAT (///10X19HELEMENT INFORMATION )                            INJD 490
  100 FORMAT (//31X7HELEMENT14X6HRADIUS19X5HANGLE11X12HDISTRIBUTION/90X INJD 500
     111HCOEFFICIENT/34X3HNO.17X3HIN.15X7HRADIANS15X2HRU// )            INJD 510
  110 FORMAT (32XI5,11XF10.3,10XF10.4,11XF10.4 )                        INJD 520
  120 FORMAT (1H1//26H TABULAR NONLINEAR EFFECTS//11X8HPRESSURE11X10HCONINJD 530
     1BUSTION12X6HRADIAL12X10HCOMBUSTION10X10HTANGENTIAL10X10HCOMBUSTIONINJD 540
     2/31X8HVELOCITY32X8HVELOCITY/ 9X 3(12HPERTURBATION13X4HGAIN12X) )  INJD 550
  130 FORMAT ( 6(F18.3,2X) )                                            INJD 560
  140 FORMAT (//33X63H************  THESE VALUES PERTAIN TO A STANDING MOINJD 570
     1EZ  ********** )                                                  INJD 580
  150 FORMAT (//33X63H************  THESE VALUES PERTAIN TO A SPINNING MOINJD 590
     1DE  ********** )                                                  INJD 600
  160 FORMAT (23X42HTHE VALUE OF IZZIT IS ZERO AND NOT ALLOWED)         INJD 610
  170 FORMAT(23X45HVALUE OF BESSEL ARGUMENT TOO HIGH OR LOW Z = F10.4)  INJD 620
C                                                                       INJD 630
  180 DSCRB = 1                                                         INJD 640
      IF ( .NOT. IRUN )     GO TO 190                                   INJD 650
      CE = 100.0                                                        INJD 660
      DSCRB = 2                                                         INJD 670
      GO TO 200                                                         INJD 680
  190 NW = 1                                                            INJD 690
      WC(1) = 1.0                                                       INJD 700
  200 TIME = 1                                                          INJD 710
      RINJ = DATA(9574)                                                 INJD 720
      NE = DATA(9571)  +0.0001                                          INJD 730
      NS = DATA(5) + 0.0001                                             INJD 731
      NUMBR = DATA(9599)  + 0.0001                                      INJD 740
      IZZIT = DATA(2)                                                   INJD 750
      K = V +0.0001                                                     INJD 760
      KK = K + 1                                                        INJD 770
      L = 2*K - 1                                                       INJD 780
      M = L + 2                                                         INJD 790
      N = M + 2                                                         INJD 800
      IF ( E1 .LE. 0.0 )    E1 = 0.0001                                 INJD 810
      IF ( NUMBR .LE. 0 )     NUMBR = 10                                INJD 820
      IF ( CE .LE. 99.0 )     GO TO 260                                 INJD 830
      CALL  PAGE  ( 70 )                                                INJD 840
      WRITE (6,60) NW,NE,NS,XM,XN,SVN,V,RINJ,GAMMA,POO,TFLP,TFLR,TFLT   INJD 850
      GO TO (220,210), DSCRB                                            INJD 860
  210 WRITE (6,70)                                                      INJD 870
      WRITE (6,80) (WC(IW), IW = 1,NW)                                  INJD 880
  220 LINKNT = 0                                                        INJD 890
      DO 240 J = 1,NE                                                   INJD 900
      IF (LINKNT .GT. 0) GO TO 230                                      INJD 910
      CALL  PAGE  ( 70 )                                                INJD 920
      WRITE (6,90)                                                      INJD 930
      WRITE (6,100)                                                     INJD 940
      LINKNT = 50                                                       INJD 950
  230 WRITE (6,110) J, RR(J), THETA(J), U(J)                            INJD 960
      LINKNT = LINKNT - 1                                               INJD 970
  240 CONTINUE                                                          INJD 980
      GO TO (260,250), DSCRB                                            INJD 990
  250 WRITE (6,120)                                                     INJD1000
      WRITE (6,130) (IPP(I),OPP(I),IPR(I),OPR(I),IPT(I),OPT(I),I=1,50)  INJD1010
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXINJD1020
C         RUNNING DESCRIBING FUNCTION IMPLIES THAT THE EXPANSION        INJD1030
C   COEFFICIENTS AVN, BVN, AND CVN ARE FUNCTIONS OF THE FREQUENCY,OMEGA,INJD1040
C   THEREFORE, AN OMEGA LOOP MUST BE ESTABLISHED, OTHERWISE, THIS LOOP  INJD1050
C  IS GONE THROUGH ONLY ONCE.                                           INJD1060
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXINJD1070
  260 DO 470 IW = 1, NW                                                 INJD1080
      WCR = WC(IW)                                                      INJD1090
      VOO = POO/(GAMMA*WCR)                                             INJD1100
      SUMP = 0.0                                                        INJD1110
      SUMR = 0.0                                                        INJD1120
      SUMT = 0.0                                                        INJD1130
      LINKNT = 0                                                        INJD1140
      DO 450 J = 1, NE                                                  INJD1150
      R = RR(J)/RINJ                                                    INJD1160
      ETA = U(J)/(XM*XN)                                                INJD1170
C ******************************************************************** INJD1180
C     THE CALL TO BESSEL REQUIRES THE DEFINITION OF THE REAL PORTION    INJD1190
```

```
C  OF THE ARGUMENT, Z(1), AND THE IMAGINARY PORTION, Z(2),      FOR A    INJD1200
C  GIVEN ORDER V, THE OUTPUT OF BESSEL GIVES JOR(Z1), JOI(Z2), JIR(Z1), INJD1210
C  JII(Z2),  ...., JVR(Z1), JVI(Z2).      THEREFORE, THE ANSWERS WILL BE  INJD1220
C  STORED  IN THE FOLLOWING FASHION,  J(V-1) = FIRST(L),  J(V) =         INJD1230
C  FIRST(N),  AND J(V+1) = FIRST(N).                                     INJD1240
C*********************************************************************INJD1250
      KERR = 0                                                          INJD1260
      Z(1) = SVN*R                                                      INJD1270
      Z(2) = 0.0                                                        INJD1280
      CALL  BESSEL  ( FIRST(1), SECOND(1), KK, Z(1), KERR )             INJD1290
      IF ( KERR )      270,270,950                                      INJD1300
  270 SIVN = FIRST(N)                                                   INJD1310
      DSIVN = ( V*FIRST(N) - Z(1)*FIRST(N) )/R                          INJD1320
      Z(1) = SVN                                                        INJD1330
      Z(2) = 0.0                                                        INJD1340
      CALL  BESSEL  ( FIRST(1), SECOND(1), KK, Z(1), KERR )             INJD1350
      IF  ( KERR )      280,280,950                                     INJD1360
  280 GO TO (290,320),  TIME                                            INJD1370
  290 IF ( K )     300,300,310                                          INJD1380
  300 D = ( FIRST(N)*FIRST(N) + FIRST(L)*FIRST(L) ) / 2.0               INJD1390
      GO TO 320                                                         INJD1400
  310 D = ( FIRST(N)*FIRST(N) - FIRST(L)*FIRST(N))*3.14159/(2.0*DATA(5))INJD1410
  320 IF ( IZZIT )     330,360,340                                      INJD1420
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXINJD1430
C  IZZIT IS NEGATIVE FOR STANDING MODES AND POSITIVE FOR SPINNING MODES INJD1440
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXINJD1450
  330 VT = V*THATA(J)                                                   INJD1460
      CVT = COS(VT)                                                     INJD1470
      SVT = SIN(VT)                                                     INJD1480
      GO TO 350                                                         INJD1490
  340 CVT = 1.0                                                         INJD1500
      SVT = 1.0                                                         INJD1510
  350 FP = 1.0                                                          INJD1520
      FR = 1.0                                                          INJD1530
      FT = 1.0                                                          INJD1540
      GO TO (420,360), DSCRB                                            INJD1550
  360 PO = ABS(POO*SIVN*CVT)                                            INJD1560
      VO = ABS(VOO*DSIVN*CVT)                                           INJD1570
      WO = ABS(VOO*SIVN*V*SVT/R)                                        INJD1580
       A = -3.14159                                                     INJD1590
      B = -A                                                            INJD1600
      IF ( TFLP )      370,380,370                                      INJD1610
  370 SAVEP = 1.0/(3.14159*PO*TFLP)                                     INJD1620
      CALL INTGR(A,B,PSI,NUMBR)                                         INJD1630
      CPSI = COS(PSI)                                                   INJD1640
      IPX = PO*CPSI                                                     INJD1650
      CALL INTA(IPP(1), OPP(1), IPX, OPX)                               INJD1660
      F = OPX*CPSI                                                      INJD1670
       CALL INTGS(F,INTEG,E1,NN)                                        INJD1680
      FP = SAVEP*INTEG                                                  INJD1690
  380 IF ( TFLR )      390,400,390                                      INJD1700
  390 SAVER = 1.0/(3.14159*VO*TFLR)                                     INJD1710
      CALL INTGR(A,B,PSI,NUMBR)                                         INJD1720
      CPSI = COS(PSI)                                                   INJD1730
      IPX = VO*CPSI                                                     INJD1740
      CALL INTA (IPR(1), OPR(1), IPX, OPX)                              INJD1750
      F = OPX*CPSI                                                      INJD1760
      CALL INTGS (F, INTEG, E1, NN)                                     INJD1770
      FR = SAVER*INTEG                                                  INJD1780
  400 IF ( TFLT )      410,420,410                                      INJD1790
  410 SAVET = 1.0/(3.14159*VT*TFLT)                                     INJD1800
       CALL INTGR(A,B,PSI,NUMBR)                                        INJD1810
      CPSI = COS(PSI)                                                   INJD1820
      IPX = WO*CPSI                                                     INJD1830
      CALL INTA (IPT(1), OPT(1), IPX, OPX)                              INJD1840
      F = OPX*CPSI                                                      INJD1850
      CALL INTGS(F,INTEG,E1, NN)                                        INJD1860
      FT = SAVET*INTEG                                                  INJD1870
  420 TERMP = ETA*FP*SIVN**2*CVT**2                                     INJD1880
      SUMP = SUMP + TERMP                                               INJD1890
      TERMR = ETA*FR*SIVN*DSIVN*CVT**2                                  INJD1900
      SUMR = SUMR + TERMR                                               INJD1910
      TERMT = -ETA*FT*SIVN**2*CVT*V*SVT/R                               INJD1920
      SUMT = SUMT + TERMT                                               INJD1930
      TIME = 2                                                          INJD1940
      GO TO (450,430),  DSCRB                                           INJD1950
  430 IF ( LINKNT .GT. 0 )      GO TO 440                               INJD1960
      CALL   PAGE  ( 70 )                                               INJD1970
      WRITE (6,10)                                                      INJD1980
      LINKNT = 50                                                       INJD1990
  440 WRITE (6,20)  WCR, J, R, THATA(J), ETA, FP, FR, FT                INJD2000
      LINKNT = LINKNT - 1                                               INJD2010
  450 CONTINUE                                                          INJD2020
      DD = D                                                            INJD2030
      IF (IZZIT .GT. 0) DD = D/2.0                                      INJD2040
      AVN(IW) = SUMP/DD                                                 INJD2050
      BVN(IW) = SUMR/DD                                                 INJD2060
      IF (IZZIT .GT. 0)  GO TO 460                                      INJD2070
      CVNR(IW) = SUMT/DD                                                INJD2080
      CVNI(IW) = 0.0                                                    INJD2090
      GO TO 470                                                         INJD2100
  460 CVNR(IW) = 0.0                                                    INJD2110
      CVNI(IW) = SUMT/DD                                                INJD2120
  470 CONTINUE                                                          INJD2130
      IF (CE .LT. 10.0 ) GO TO 530                                      INJD2140
      CALL   PAGE  ( 70 )                                               INJD2150
      WRITE (6,30)                                                      INJD2160
      GO TO (480,490),  DSCRB                                           INJD2170
  480 WRITE (6,50)  AVN(1 ), BVN(1 ), CVNR(1 ), CVNI(1 )                INJD2180
      GO TO 500                                                         INJD2190
  490 WRITE (6,40)  ( WC(I), AVN(I), BVN(I), CVNR(I), CVNI(I),  I=1,NW )INJD2200
  500 IF ( IZZIT ) 510,560,520                                          INJD2210
  510 WRITE(6,140)                                                      INJD2220
      GO TO 530                                                         INJD2230
  520 WRITE (6,150)                                                     INJD2240
  530 CONTINUE                                                          INJD2250
  540 RETURN                                                            INJD2260
  550 WRITE (6,170) Z(1)                                                INJD2270
      GO TO 540                                                         INJD2280
  560 WRITE (6,160)                                                     INJD2290
      GO TO 540                                                         INJD2300
      END                                                               INJD2310
```

```
$IBMAP BESS    LIST.REF.DECK                                          BESS0010
*      MAP                                                            BESS0020
       PMC     ON                                                     BESS0030
       ENTRY   BESSEL                                                 BESS0040
       LBL     BESSEL.X                                               BESS0050
*      CALL BESSEL (FIRST.SECOND.N.ARG.KODE )                         BESS0060
*                                                                     BESS0070
*      FIRST  = FIRST LOCATION OF ASCENDING STORING SEQUENCE FOR JN(Z). BESS0080
*               OR IN(Z).                                             BESS0090
*      SECOND = FIRST LOCATION OF ASCENDING STORING SEQUENCE FOR YN(Z). BESS0100
*               OR KN(Z).                                             BESS0110
*      N      = HIGHEST INTEGRAL ORDER DISIRED FOR BOTH JN(Z) AND YN(Z). BESS0120
*               OR IN(Z) AND KN(Z).                                   BESS0130
*      ARG    = LOCATION OF Z = X+I*Y WITH X=ARG(1) AND Y=ARG(2).    BESS0140
*      KODE   = +0.COMPUTE JN(Z) AND YN(Z).                          BESS0150
*               LESS ZERO.COMPUTE IN(Z) AND KN(Z).                   BESS0160
*      DIMENSIONS REQUIREMENTS FIRST(300).SECOND(2*N+2).ARG(2)        BESS0170
*                                                                     BESS0180
COMMON BSS     20                                                     BESS0190
BESSEL SXA     BS4.4                                                  BESS0200
       SXA     BS4+1.1                                                BESS0210
       SXA     BS4+2.2                                                BESS0220
       TXI     BESS1.4.-2                                             BESS0230
 BESS1 CAL     1.4                                                    BESS0240
       STA     BESSP+1                                                BESS0250
       CAL     2.4                                                    BESS0260
       ALS     18                                                     BESS0270
       STD     BESSP+1                                                BESS0280
       LDQ*    3.4                                                    BESS0290
       STQ     BESSP+2                                                BESS0300
       CAL*    5.4                                                    BESS0310
       STP     BESSP+1                                                BESS0320
       CLA     4.4                                                    BESS0330
       STA     BESS2                                                  BESS0340
       ADD     =1                                                     BESS0350
       STA     BESS3                                                  BESS0360
 BESS2 CLA     **                                                     BESS0370
       LDQ     =.5E-5                                                 BESS0380
       TLQ     BESS3                                                  BESS0390
       TRA     ERRR1                                                  BESS0400
 BESS3 LDQ     **                                                     BESS0410
 BESSP TSX     BFBF.4                                                 BESS0420
       PZE     **.,**          L(J).L(Y)                              BESS0430
       PZE     **              N                                      BESS0440
       TRA     ERRR2           ERROR.,BFBF                            BESS0450
       PXD     0.0                                                    BESS0460
 BS4   AXT     **.4                                                   BESS0470
       AXT     **.1                                                   BESS0480
       AXT     **.2                                                   BESS0490
       STO*    7.4                                                    BESS0500
       TRA     1.4                                                    BESS0510
 ERRR1 CLA     =1                                                     BESS0520
       TRA     BS4                                                    BESS0530
 ERRR2 CLA     =2                                                     BESS0540
       TRA     BS4                                                    BESS0550
*      REM ALL ORDERS OF THE BESSEL FUNCTIONS Y SUB K TIMES (Z)       BESS0560
*          AND J SUB K TIMES (Z) FOR COMPLEX Z                        BESS0570
BFBF   STI     BFB1                                                   BESS0580
       LDI     1.4                                                    BESS0590
       LFT     400000                                                 BESS0600
       XCA                                                            BESS0610
       SXD     G20.1                                                  BESS0620
       SXD     G21.2                                                  BESS0630
       SXD     G22.4                                                  BESS0640
       STO     COMMON+15                                              BESS0650
       STO     COMMON+16                                              BESS0660
       CLA     0                                                      BESS0670
       STO     CL                                                     BESS0680
       CLA     8                                                      BESS0690
       STO     CL+1                                                   BESS0700
       CLA     TRA1                                                   BESS0710
       STO     8                                                      BESS0720
       EFTM                                                           BESS0730
R29    CAL     1.4                                                    BESS0740
       STA     BFR1                                                   BESS0750
       COM                                                            BESS0760
       ACL     L11                                                    BESS0770
       STO     COMMON+17                                              BESS0780
       CLA     2.4                                                    BESS0790
       ALS     18                                                     BESS0800
       STD     BFR1                                                   BESS0810
       CLA     COMMON+15                                              BESS0820
       LDQ     COMMON+16                                              BESS0830
       TSX     BF7F.4                                                 BESS0840
BFR1   PZE     **0.0.**0                                              BESS0850
G20    TXL     BFBX.0.*                                               BESS0860
       LDQ     COMMON+15                                              BESS0870
B61    FMP     COMMON+15                                              BESS0880
       STO     COMMON+5                                               BESS0890
       LDQ     COMMON+16                                              BESS0900
       FMP     COMMON+16                                              BESS0910
       FAD     COMMON+5                                               BESS0920
       FDH     LOC4                                                   BESS0930
       STQ     COMMON+5                                               BESS0940
       CLA     COMMON+5                                               BESS0950
       CALL ALOG(COMMON+5)                                            BESS0960
       TRA     G21+1                                                  BESS0970
G21    TXL     BFBX.0.*                                               BESS0980
       FDH     LOC2                                                   BESS0990
       XCA                                                            BESS1000
       FAD     OILFR                                                  BESS1010
       STO     COMMON+8        RL+GAM                                 BESS1020
       LDQ     COMMON+15                                              BESS1030
       CLA     SWCON                                                  BESS1040
       LRS     0                                                      BESS1050
       TLQ     R2                                                     BESS1060
       TRA     R3                                                     BESS1070
R2     LDQ     COMMON+16                                              BESS1080
       CLA     PIV2                                                   BESS1090
       LLS     0                                                      BESS1100
G22    TXL     R4.0.*                                                 BESS1110
R3     CLA     COMMON+16                                              BESS1120
       FDP     COMMON+15                                              BESS1130
       STQ     COMMON                                                 BESS1140
       CALL ATAN(COMMON)                                              BESS1150
       LDQ     COMMON+15                                              BESS1160
       TQP     R4                                                     BESS1170
       STO     COMMON                                                 BESS1180
       CLA     LPI                                                    BESS1190
       LDQ     COMMON+16                                              BESS1200
       LLS     0                                                      BESS1210
```

```
        FAD   COMMON                                    BESS1220
R4      STO   COMMON+9                                  BESS1230
        LXA   COMMON+17,1                               BESS1240
        CLA   0,1                                       BESS1250
        LDQ   1,1                                       BESS1260
        TSX   MULT,4                                    BESS1270
        STO   COMMON+4                                  BESS1280
        STO   COMMON+5                                  BESS1290
        STZ   COMMON+8                                  BESS1300
        STZ   COMMON+9                                  BESS1310
        LXD   A58,2                                     BESS1320
        SXD   R68,2                                     BESS1330
B63     CLA   LOC1                                      BESS1340
R10     STO   COMMON+6                                  BESS1350
        FAD   LOC1                                      BESS1360
        STO   COMMON+7                                  BESS1370
        CLA   8,1                                       BESS1380
B64     FDP   COMMON+7                                  BESS1390
        STO   COMMON+10                                 BESS1400
        CLA   4,1                                       BESS1410
        FDP   COMMON+6                                  BESS1420
        STO   COMMON+11                                 BESS1430
        CLA   COMMON+11                                 BESS1440
R65     FSB   COMMON+10                                 BESS1450
        FAD   COMMON+8                                  BESS1460
        STO   COMMON+8                                  BESS1470
        CLA   9,1                                       BESS1480
B66     FDP   COMMON+7                                  BESS1490
        [illegible]   COMMON+11                         BESS1500
        CLA   5,1                                       BESS1510
        FDP   COMMON+6                                  BESS1520
        STO   COMMON+10                                 BESS1530
        CLA   COMMON+10                                 BESS1540
R67     FSB   COMMON+11                                 BESS1550
        FAD   COMMON+9                                  BESS1560
        STO   COMMON+9                                  BESS1570
        CLA   COMMON+6                                  BESS1580
        FAD   LOC2                                      BESS1590
        TXI   R60,1,-8                                  BESS1600
B6E     TXH   R10,1,0                                   BESS1610
Z60     LDQ   COMMON+8                                  BESS1620
        FMP   LOC2                                      BESS1630
        FAD   COMMON+4                                  BESS1640
        XCA                                             BESS1650
        FMP   TVPI                                      BESS1660
        LXD   COMMON+17,2                               BESS1670
        LXA   COMMON+17,1                               BESS1680
        STO   0,2              RYZ                      BESS1690
        LDQ   COMMON+9                                  BESS1700
        FMP   LOC2                                      BESS1710
        FAD   COMMON+5                                  BESS1720
        XCA                                             BESS1730
        FMP   TVPI                                      BESS1740
        STO   1,2              IYZ                      BESS1750
        LXD   G22,4                                     BESS1760
        CLA   2,4                                       BESS1770
        TZE   EXIT                                      BESS1780
        TSX   YM2,4                                     BESS1790
        LXD   G22,4                                     BESS1800
        CLA   2,4                                       BESS1810
        LRS   1                                         BESS1820
        TZE   EXIT                                      BESS1830
        LLS   1                                         BESS1840
        COM                                             BESS1850
        ACL   L1                                        BESS1860
        ALS   19                                        BESS1870
        ACL   COMMON+17        -L(YSUBN)                BESS1880
        TXI   A880,1,-2                                 BESS1890
AR80    TXI   R71,2,-2                                  BESS1900
R71     STD   R15                                       BESS1910
        STD   R75                                       BESS1920
        CLA   COMMON+16                                 BESS1930
        LDQ   LSIX                                      BESS1940
        LLS   0                                         BESS1950
        TLQ   METH2                                     BESS1960
        CLA   L2                                        BESS1970
R14     FAD   LOC1                                      BESS1980
        STO   COMMON+5                                  BESS1990
        ACL   C2                                        BESS2000
        STO   COMMON+8                                  BESS2010
        STZ   COMMON+9                                  BESS2020
        CLA   COMMON+15                                 BESS2030
        LDQ   COMMON+16                                 BESS2040
        TSX   DIV,4                                     BESS2050
        STO   COMMON+8                                  BESS2060
        STO   COMMON+9                                  BESS2070
        CLA   0,2                                       BESS2080
        LDQ   1,2                                       BESS2090
        TSX   MULT,4                                    BESS2100
        STO   COMMON+8                                  BESS2110
F21     FSB   0-2,2                                     BESS2120
        STO   2,2                                       BESS2130
        CLA   COMMON+8                                  BESS2140
        FSB   0-1,2                                     BESS2150
        STO   3,2                                       BESS2160
        CLA   COMMON+5                                  BESS2170
        TXI   R15,2,-2                                  BESS2180
R15     TXH   R14,2,0                                   BESS2190
YMX     TXL   EXIT,0,0                                  BESS2200
METH2   TSX   YM22,4                                    BESS2210
        TXI   B74,1,-2                                  BESS2220
R74     TXI   B75,2,-2                                  BESS2230
R75     TXH   METH2,2,0                                 BESS2240
EXIT    LXD   G22,4                                     BESS2250
        TRA   RFR                                       BESS2260
BR0     LXD   G21,2                                     BESS2270
        LXD   G20,1                                     BESS2280
        CLA   CL                                        BESS2290
        STO   0                                         BESS2300
        CLA   CL+1                                      BESS2310
        STO   8                                         BESS2320
        TRA   4,4                                       BESS2330
RFBX    CLA   COMMON+15                                 BESS2340
        LDQ   COMMON+16                                 BESS2350
        LXD   G22,4                                     BESS2360
        TXI   R80,4,1                                   BESS2370
YM2     SXD   YMX,4                                     BESS2380
        CLA   TVPI                                      BESS2390
        STO   COMMON+8                                  BESS2400
        STZ   COMMON+9                                  BESS2410
        CLA   COMMON+15                                 BESS2420
```

```
        LDQ     COMMON+16                                       BESS2430
        TSX     DIV+4                                           BESS2440
        STO     COMMON+6                                        BESS2450
        STQ     COMMON+7                                        BESS2460
        TRA     YM5                                             BESS2470
YM2     SXD     YMX+4                                           BESS2480
YM5     CLA     COMMON+6                                        BESS2490
        LDQ     COMMON+7                                        BESS2500
        STO     COMMON+8                                        BESS2510
        STQ     COMMON+9                                        BESS2520
        CLA     0+1                                             BESS2530
        LDQ     1+1                                             BESS2540
        TSX     DIV+4                                           BESS2550
        STO     COMMON                                          BESS2560
        STQ     COMMON+1                                        BESS2570
        CLA     2+1                                             BESS2580
        STO     COMMON+8                                        BESS2590
        CLA     3+1                                             BESS2600
        STO     COMMON+9                                        BESS2610
        CLA     0+1                                             BESS2620
        LDQ     1+1                                             BESS2630
        TSX     DIV+4                                           BESS2640
        STO     COMMON+8                                        BESS2650
        STQ     COMMON+9                                        BESS2660
        CLA     0+2                                             BESS2670
        LDQ     1+2                                             BESS2680
        TSX     MULT+4                                          BESS2690
        STQ     COMMON+3                                        BESS2700
        FSB     COMMON                                          BESS2710
        STO     2+2                                             BESS2720
        CLA     COMMON+3                                        BESS2730
        FSB     COMMON+1                                        BESS2740
        STO     3+2                                             BESS2750
        LXD     YMX+4                                           BESS2760
        TRA     1+4                                             BESS2770
LOG4    DEC 4.                                                  BESS2780
LI1     OCT 1000001                                             BESS2790
OILER   DEC .5772156649                                         BESS2800
LSIX    DEC 6.                                                  BESS2810
PIV2    DEC 1.570796325                                         BESS2820
LPI     DEC 3.14159265                                          BESS2830
TVPI    DEC .63661977236                                        BESS2840
CI      BSS     2                                               BESS2850
BFR     LDI     BF81                                            BESS2860
        CLA     1+4                                             BESS2870
        TPL     BB0             EXIT                            BESS2880
        STA     BF82            L(J)                            BESS2890
        ACL     BF84                                            BESS2900
        STA     BF82+1          L(J)+1                          BESS2910
        ARS     18                                              BESS2920
        STA     BF83            L(Y)                            BESS2930
        ACL     BF84                                            BESS2940
        STA     BF83+1          L(Y)+1                          BESS2950
        CAL     2+4                                             BESS2960
        ADD     BF84                                            BESS2970
        PAX     +1                                              BESS2980
        AXT     0+2                                             BESS2990
BFR6    CLS*    BF82+1          -IJ                             BESS3000
        FSB*    BF83            -RY                             BESS3010
        XCA                                                     BESS3020
        FMP     BF85                                            BESS3030
        STO*    BF83            RK0                             BESS3040
        CLA*    BF82            RJ                              BESS3050
        FSB*    BF83+1          -IY                             BESS3060
        XCA                                                     BESS3070
        FMP     BF85                                            BESS3080
        STO*    BF83+1          IK0                             BESS3090
        TNX     BB0+1+1                                         BESS3100
        TXI      BESS9+2+-2                                     BESS3110
BESS9   CLS*    BF82            -RJ                             BESS3120
        FAD*    BF83+1          IY                              BESS3130
        XCA                                                     BESS3140
        FMP     BF85                                            BESS3150
        XCA                                                     BESS3160
        CLS*    BF83            -RY                             BESS3170
        STQ*    BF83            RK1                             BESS3180
        FSB*    BF82+1          -IJ                             BESS3190
        XCA                                                     BESS3200
        FMP     BF85                                            BESS3210
        STO*    BF83+1          IK1                             BESS3220
        CLS*    BF82                                            BESS3230
        LDQ*    BF82+1                                          BESS3240
        STO*    BF82+1                                          BESS3250
        STQ*    BF82                                            BESS3260
        TNX     BB0+1+1                                         BESS3270
        TXI      BESS6+2+-2                                     BESS3280
BESS6   CLA*    BF82+1          IJ                              BESS3290
        FAD*    BF83            RY                              BESS3300
        XCA                                                     BESS3310
        FMP     BF85                                            BESS3320
        STO*    BF83            RK2                             BESS3330
        CLS*    BF82            -RJ                             BESS3340
        STO*    BF82                                            BESS3350
        FAD*    BF83+1          IY                              BESS3360
        XCA                                                     BESS3370
        FMP     BF85                                            BESS3380
        STO*    BF83+1                                          BESS3390
        CLS*    BF82+1                                          BESS3400
        STO*    BF82+1                                          BESS3410
        TNX     BB0+1+1                                         BESS3420
        TXI      BESS7+2+-2                                     BESS3430
BESS7   CLA*    BF82            RJ                              BESS3440
        FSB*    BF83+1          -IY                             BESS3450
        XCA                                                     BESS3460
        FMP     BF85                                            BESS3470
        XCA                                                     BESS3480
        CLA*    BF83            RY                              BESS3490
        STQ*    BF83            RK3                             BESS3500
        FAD*    BF82+1          IJ                              BESS3510
        XCA                                                     BESS3520
        FMP     BF85                                            BESS3530
        STO*    BF83+1          IK3                             BESS3540
        CLS*    BF82+1                                          BESS3550
        LDQ*    BF82                                            BESS3560
        STO*    BF82                                            BESS3570
        STQ*    BF82+1                                          BESS3580
        TNX     BB0+1+1                                         BESS3590
        TXI     BFR6+2+-2                                       BESS3600
BF81    PZE     **0             INDICATORS                      BESS3610
BF82    PZE     **0+2           L(J)                            BESS3620
        PZE     **0+2                                           BESS3630
```

```
[illegible]  PZW   [illegible]          L99
             PZW   [illegible]
[illegible]  DEC   [illegible]
[illegible]  DEC   [illegible]          PI/2
[illegible]  STO   [illegible]
             STO   [illegible]
             STO   [illegible]
             STO   [illegible]
             STO   [illegible]
             TLO   [illegible]
             STO   [illegible]
             CLA   0
             STO   [illegible]
             CLA   [illegible]
             STO   0
             [illegible]
             CLA   1.0
             TRL   A21
             CLA   COMMON+0
             LDO   COMMON+0
             STO   COMMON+0
             STO   COMMON+0
             STO   COMMON+0
A21          CLA   COMMON+0
             LDO   [illegible]
             LLS   9
             TLO   A99
             LDO   TEN
A01          TRL   A119-[illegible]
A99          CLA   COMMON+0
F29          PDP   COMMON+0
A119         CLA   L99
             LRS   0
             STO   COMMON+7          TANANGS
             TLO   F1
A27          CLA   COMMON+0
             TSX   [illegible]
A00          TRL   [illegible]
             STO   COMMON+0
             STO   COMMON+0
             CLA   COMMON+0
             CALL  COS(COMMON+5)
             RCA
F96          PWP   COMMON+0
             STO   COMMON+0          0(COS(Z))
             CLA   COMMON+0
             CALL  SIN(COMMON+5)
             RCA
F29          PWP   COMMON+0
             CHS
             STO   COMMON+0          0(COS(Z))
F1           CLA   COMMON+2
             LDO   COMMON+6
             SSP
             LRS   0
             TLO   A27
             RCA
F79          STO   COMMON+0
             RCA
             PWP   L997
             STO   COMMON+0          S/2P
             LDO   [illegible]
             CAL   1.4
             CBM
             ACL   L1
             STA   COMMON            (-J)
             ALS   10
             STD   A10
             CLA   1.4
             ADC   10
             SSP
             ADD   C1
             FAD   C1                FLOATING N
             LDO   COMMON+8
             TLO   A24
             CLA   COMMON+8
A24          FAD   TEN
             UFA   C1                N PRIME=MAX(S/2P,N)+10
             STO   COMMON+10
             FAD   C1
             STO   COMMON+1
             CAL   COMMON+10
             ALS   1
             CBM
             ADM   L1
             ADM   COMMON
             STA   COMMON+2
             ALS   10
             STD   A41               -(J+2*PR+2)
             STD   A32
             STD   A45
             STD   A38
             CLA   COMMON+9
             TZE   [illegible]
             CLA   COMMON+6
             SSP
             LDO   LOC29
             TLO   A68
             CLA   AZ
             TRA   A69
A68          CLA   AZ1
A69          LRA   COMMON+2,1
             STO   2,1
             STO   3,1
             STZ   4,1
             STZ   5,1
             CLA   COMMON+1
             FAD   LOC1
[illegible]  STO   COMMON+1
             ACL   C2
             STO   COMMON+8
             STZ   COMMON+9
             CLA   2,1
             LDO   3,1
             TSX   MULT,4
             STO   COMMON+8
             STO   COMMON+9
             CLA   COMMON+5
             LDO   COMMON+6
             TSX   DIV,4
             STO   COMMON+10
[illegible]  FAD   4,1
```

```
                 1,1                              RES54850
        LDQ      COMMON+1                         RES54860
        [illegible]                               RES54870
        [illegible]                               RES54880
        [illegible] COMMON+2                      RES54890
        [illegible]                               RES54900
        [illegible] [illegible],2                 RES54910
[illegible] [illegible] [illegible]   [illegible] RES54920
        [illegible] [illegible]                   RES54930
[illegible] [illegible] COMMON+7                  RES54940
        [illegible] COMMON+1                      RES54950
        [illegible] COMMON+15                     RES54960
        [illegible] COMMON+11                     RES54970
        [illegible] A67                           RES54980
        [illegible] [illegible]                   RES54990
        [illegible] A61                           RES55000
A63     [illegible] [illegible]                   RES55010
[illegible] [illegible] A67                       RES55020
        [illegible] [illegible]                   RES55030
        [illegible] A6[illegible]                 RES55040
[illegible] [illegible] 9,1                       RES55050
[illegible] [illegible]                           RES55060
        [illegible] COMMON+7                      RES55070
        [illegible] COMMON+5                      RES55080
        [illegible] 1,1                           RES55090
A9      [illegible]                               RES55100
                 COMMON+11                        RES55110
        [illegible] COMMON+11                     RES55120
        [illegible] A32,1,-0                      RES55130
[illegible] TXH  A65,1,0                          RES55140
        LXA      COMMON,1                         RES55150
        XCA                                       RES55160
        [illegible] [illegible]                   RES55170
        [illegible] [illegible]                   RES55180
        [illegible] COMMON+1[illegible]           RES55190
        LDQ      COMMON+10                        RES55200
        [illegible] LOC2                          RES55210
        [illegible] 9,1                           RES55220
        STO      COMMON+1[illegible]              RES55230
        CLA      COMMON+7                         RES55240
        LDQ      L[illegible]                     RES55250
        TLQ      A99                              RES55260
        CLA      LOC1                             RES55270
        [illegible] COMMON+9                      RES55280
        [illegible] COMMON+6                      RES55290
A[illegible] CLA COMMON+9                         RES55300
        LDQ      COMMON+6                         RES55310
        STO      COMMON+0                         RES55320
        STQ      COMMON+0                         RES55330
        CLA      COMMON+15                        RES55340
        LDQ      COMMON+11                        RES55350
        TSX      91V,2,4                          RES55360
        STO      COMMON+0                         RES55370
        STQ      COMMON+0                         RES55380
[illegible] CLA  [illegible],1                    RES55390
        LDQ      1,1                              RES55400
        TSX      MULT,4                           RES55410
        STO      [illegible],1                    RES55420
        [illegible] [illegible]                   RES55430
        TXI      A65,1,-2                         RES55440
A65     TXH      A36,1,0                          RES55450
        LXA      COMMON+2,1                       RES55460
        LXD      L2,7                             RES55470
A37     STZ      9,1                              RES55480
        TXI      A35,2,-1                         RES55490
A35     TXI      A36,1,-2                         RES55500
A36     TXH      A37,2,-0                         RES55510
        [illegible] XR[illegible],4               RES55520
        CLA      1,4                              RES55530
        TXL      A7                               RES55540
        LXA      COMMON,2                         RES55550
A[illegible] CLS 2,2                              RES55560
        LDQ      3,2                              RES55570
        STQ      2,2                              RES55580
        STO      3,2                              RES55590
        CLS      4,2                              RES55600
        STO      4,2                              RES55610
        CLS      5,2                              RES55620
        STO      5,2                              RES55630
        CLS      7,2                              RES55640
        LDQ      6,2                              RES55650
        STQ      7,2                              RES55660
        STO      6,2                              RES55670
        TXI      A38,2,-P                         RES55680
A38     TXH      A39,2,0                          RES55690
A7      LXD      XR4,4                            RES55700
A40     LXD      COMMON+14,2                      RES55710
        LXD      XR1,1                            RES55720
        CLA      LFST                             RES55730
        STO      0                                RES55740
        CLA      LFST+1                           RES55750
        STO      0                                RES55760
        TRA      1,4                              RES55770
FRR2    CLA      COMMON+5                         RES55780
        LDQ      COMMON+6                         RES55790
        LXD      XR4,4                            RES55800
        TXI      A40,4,1                          RES55810
RETR    LXA      COMMON,1                         RES55820
        CLA      LOC1                             RES55830
        STO      2,1                              RES55840
A42     STZ      1,1                              RES55850
        TXI      A41,1,-1                         RES55860
A41     TXH      A42,1,0                          RES55870
MULT    TXL      D7,0,0                           RES55880
DIV     STO      COMMON+10                        RES55890
        STQ      COMMON+11                        RES55900
        SSP                                       RES55910
        LRS      0                                RES55920
        TLQ      DP                               RES55930
        XCA                                       RES55940
        TZE      FRR2                             RES55950
        CLA      COMMON+10                        RES55960
D10     FDP      COMMON+11                        RES55970
D11     STO      COMMON+12        C/D             RES55980
D13     FMP      COMMON+12                        RES55990
D14     FAD      LOC1                             RES56000
        XCA                                       RES56010
        FMP      COMMON+11                        RES56020
        STO      COMMON+13        DEN             RES56030
        LDQ      COMMON+12                        RES56040
        FMP      COMMON+8                         RES56050
```

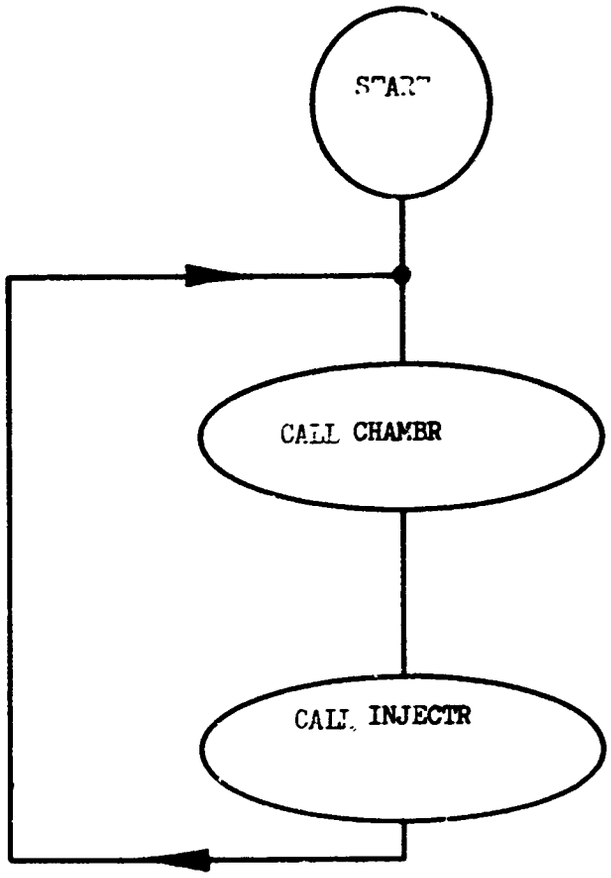
```
235   FAD   COMMON+9                                          BESS6060
F31   FDP   COMMON+13                                         BESS6070
      STO   COMMON+11                                         BESS6080
      LDQ   COMMON+17                                         BESS6090
236   FMP   COMMON+9                                          BESS6100
      FSB   COMMON+8                                          BESS6110
MEX   TXL   RB3.0.0                                           BESS6120
B61   CLA   COMMON+11                                         BESS6130
237   FMP   COMMON+10                                         BESS6140
239   STO   COMMON+12                                         BESS6150
B88   FMP   COMMON+12                                         BESS6160
242   FAD   LOC1                                              BESS6170
      XCA                                                     BESS6180
      FMP   COMMON+10                                         BESS6190
      STO   COMMON+13        DEM                              BESS6200
      LDQ   COMMON+17                                         BESS6210
      FMP   COMMON+9                                          BESS6220
041   FAD   COMMON+8                                          BESS6230
F30   FDP   COMMON+13                                         BESS6240
      STO   COMMON+11                                         BESS6250
      LDQ   COMMON+17                                         BESS6260
242   FMP   COMMON+8                                          BESS6270
244   CHS                                                     BESS6280
      FAD   COMMON+9                                          BESS6290
RB3   FDP   COMMON+13                                         BESS6300
      CLA   COMMON+11                                         BESS6310
      TRA   1.4                                               BESS6320
MULT  STO   COMMON+10                                         BESS6330
      STO   COMMON+11                                         BESS6340
      FMP   COMMON+9                                          BESS6350
MT1   STO   COMMON+12        MD                               BESS6360
      LDQ   COMMON+11                                         BESS6370
      FMP   COMMON+8                                          BESS6380
MT2   STO   COMMON+11        AD                               BESS6390
      LDQ   COMMON+10                                         BESS6400
      FMP   COMMON+8                                          BESS6410
      FSB   COMMON+12                                         BESS6420
MT3   STO   COMMON+12                                         BESS6430
      LDQ   COMMON+10                                         BESS6440
      FMP   COMMON+9                                          BESS6450
      FAD   COMMON+11                                         BESS6460
MT4   XCA                                                     BESS6470
      CLA   COMMON+12                                         BESS6480
      TRA   1.4                                               BESS6490
SPILL STO   SVE                                               BESS6500
      STO   SVE+1                                             BESS6510
      CLA   0                                                 BESS6520
      LRS   21                                                BESS6530
      LBT                                                     BESS6540
      TRA   SP1              FAD.FSB.FMP O.F. OR U.F.         BESS6550
      RND                    FDP U.F. OR O.F.                 BESS6560
      LBT                                                     BESS6570
      TRA   ERR2             OVERFLOW                         BESS6580
      LLS   9                                                 BESS6590
      LBT                                                     BESS6600
      TRA   OOK                                               BESS6610
      LDQ   LZ                                                BESS6620
      TRA*  0                                                 BESS6630
OOK   LDQ   SVE+1                                             BESS6640
      TRA*  0                                                 BESS6650
SP1   LLS   1                                                 BESS6660
      LBT                                                     BESS6670
      TRA   SP2                                               BESS6680
      TRA   ERR2             O.F.                             BESS6690
SP2   LLS   1                                                 BESS6700
      TZF   OK                                                BESS6710
      CLA   LZ                                                BESS6720
      TRA*  0                                                 BESS6730
OK    CLA   SVE                                               BESS6740
      TRA*  0                                                 BESS6750
SVE   BSS   2                                                 BESS6760
TEST  BSS   2                                                 BESS6770
TRA1  TRA   SPILL                                             BESS6780
CON2  OCT 288777777777                                        BESS6790
AZ    OCT 1400000000                                          BESS6800
AZ1   OCT 100400000000                                        BESS6810
TEN   DEC 10.                                                 BESS6820
LOG25 DEC 25.                                                 BESS6830
C1    OCT 233000000000                                        BESS6840
C2    OCT 100000000                                           BESS6850
LOC1  DEC 1.                                                  BESS6860
LZ    PZE   0                                                 BESS6870
L1    DEC                                                     BESS6880
L2    DEC .                                                   BESS6890
LT45  DEC .03492                                              BESS6900
LOC2  DEC 2.                                                  BESS6910
L3V2  DEC 1.5                                                 BESS6920
LFSB4 FSB   4.1                                               BESS6930
LFAD4 FAD   4.1                                               BESS6940
SMCON OCT 1040000000                                          BESS6950
HY3F  LAS   HY3F1              10 EXP-4                       BESS6960
      TRA   HY3F1+3            UNDERFLOW IMPOSSIBLE           BESS6970
HY3F1 TXI   HY3F1+1.0.26531          OCT 1636430..... * 1E-4 BESS6980
      LDQ   HY3F2              SMALL ARGUMENT APPROXIMATION   BESS6990
      TRA   2.4                                               BESS7000
      STO   COMMON                                            BESS7010
      SSP                                                     BESS7020
      LDQ   HY3F3              88.028                         BESS7030
      TLQ   1.4                ERROR                          BESS7040
      LDQ   HY3F4              LN 2/2                         BESS7050
      TLQ   HY3F4                                             BESS7060
      LDQ   COMMON                                            BESS7070
      FMP   COMMON                                            BESS7080
      STO   COMMON+1           X SQUARED                      BESS7090
      FAD   HY3F5              30                             BESS7100
      XCA                                                     BESS7110
      FMP   COMMON+1           X SQ                           BESS7120
      FAD   HY3F6              360                            BESS7130
      FDP   HY3F7              720                            BESS7140
      FMP   COMMON+1           X SQ                           BESS7150
      FAD   HY3F8              1                              BESS7160
      STO   COMMON+2           COSH X                         BESS7170
      CLA   COMMON+1           X SQ                           BESS7180
      FAD   HY3F8              42                             BESS7190
      XCA                                                     BESS7200
      FMP   COMMON+1           X SQ                           BESS7210
      FAD   HY3F9              840                            BESS7220
      FDP   HY3F10             5040                           BESS7230
      FMP   COMMON+1           X SQ                           BESS7240
      XCA                                                     BESS7250
      FMP   COMMON                                            BESS7260
```

```
        FAD     COMMON                                                  AE557270
        TRA     HY3F5                                                   AE557280
        LD      HY                                                      AE557290
        A       HY3F11                                                  AE557300
        STA     AE558                                                   AE557310
        MPY     HY3F12                                                  AE557320
AE558   LRS     0                                                       AE557330
        ALS     HY                                                      AE557340
        ST      COMMON+6                                                AE557350
        LD      0                                                       AE557360
        STQ     COMMON+1          4+31                                  AE557370
        MPY     COMMON+1          4+31                                  AE557380
        STQ     COMMON+2          4+4+8+27                              AE557390
        ADD     HY3F13            8+27                                  AE557400
        STQ     COMMON+3          8+27                                  AE557410
        CLA     HY3F14            17+29                                 AE557420
        DIV     COMMON+3          8+27                                  AE557430
        STQ     COMMON+3          4+31                                  AE557440
        LDQ     COMMON+2          8+27                                  AE557450
        MPY     HY3F15            0+35                                  AE557460
        LLS     6                 4+31                                  AE557470
        SUB     COMMON+3          4+31                                  AE557480
        ADD     HY3F16            4+31                                  AE557490
        STQ     COMMON+1          4+31                                  AE557500
        STQ     COMMON+3          4+31                                  AE557510
        CLA     COMMON+1          5+31 (2P)                             AE557520
        LDQ     HY3F2                                                   AE557530
        SUB     COMMON+3+0+27                                           AE557540
        RCA                                                             AE557550
        DRA     HY3F17                                                  AE557560
        FAD     HY3F2                                                   AE557570
        ADD     COMMON+4                                                AE557580
        SUB     HY3F18            EXP X IN AC. DIVIDE BY 2              AE557590
        STQ     COMMON+4                                                AE557600
        CLA     HY3F19            1/4                                   AE557610
        FMP     COMMON+4          EXP X/2                               AE557620
        STQ     COMMON+1          EXP -X 17                             AE557630
        RCA                                                             AE557640
        FAD     COMMON+4                                                AE557650
        STQ     COMMON+7                                                AE557660
        CLA     COMMON+4                                                AE557670
        FSB     COMMON+1          SIN X IN AC                           AE557680
        LDQ     COMMON            SINH X IN AC                          AE557690
        LLS     9                                                       AE557700
HY3FC   LDQ     COMMON+2                                                AE557710
        TRA     2+4               EXIT                                  AE557720
HY3F2   DEC 1.                                                          AE557730
HY3F3   DEC 88.028                                                      AE557740
HY3F4   OCT 177562710350                LN 2/2. ERROR = 0.000000000095  AE557750
HY3F5   DEC 30.                                                         AE557760
HY3F6   DEC 360.                                                        AE557770
HY3F7   DEC 720.                                                        AE557780
HY3F8   DEC 42.                                                         AE557790
HY3F9   DEC 840.                                                        AE557800
HY3F10  DEC 5040.                                                       AE557810
HY3F11  OCT 242                                                         AE557820
HY3F12  DEC 1.442695040881                                              AE557830
HY3F13  DEC 87.417497202188     A                                       AE557840
HY3F14  DEC 617.972695817       B                                       AE557850
HY3F15  DEC 0.0346573590380     C                                       AE557860
HY3F16  DEC 9.9649957831104     D                                       AE557870
HY3F17  OCT 201000000000                                                AE557880
HY3F18  OCT 100000000                                                   AE557890
HY3F19  DEC 0.25                                                        AE557900
        END                                                             AE557910

SENTRY V8050

$DATA
```

"MAIN" PROGRAM DECK = 73050

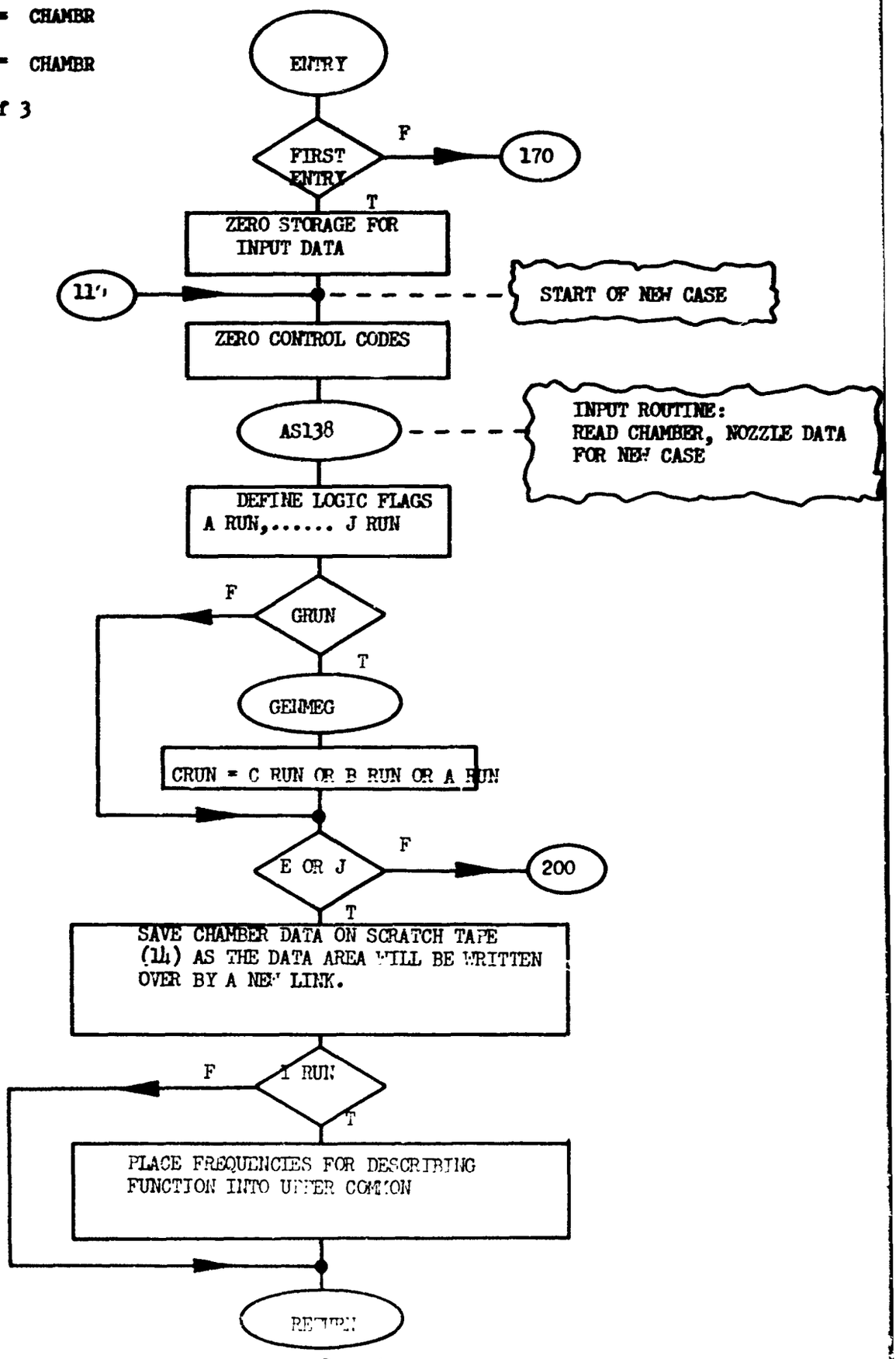


NOTE: SUBROUTINE CHAMBER PERFORMS MOST OF THE LOGICAL AND CONTROL FUNCTIONS OF AN AUTONOMOUS MAIN PROGRAM. CONTROL IS RETURNED TO THIS PROGRAM IF AND ONLY IF AN INJECTOR ROUTINE IS REQUIRED.

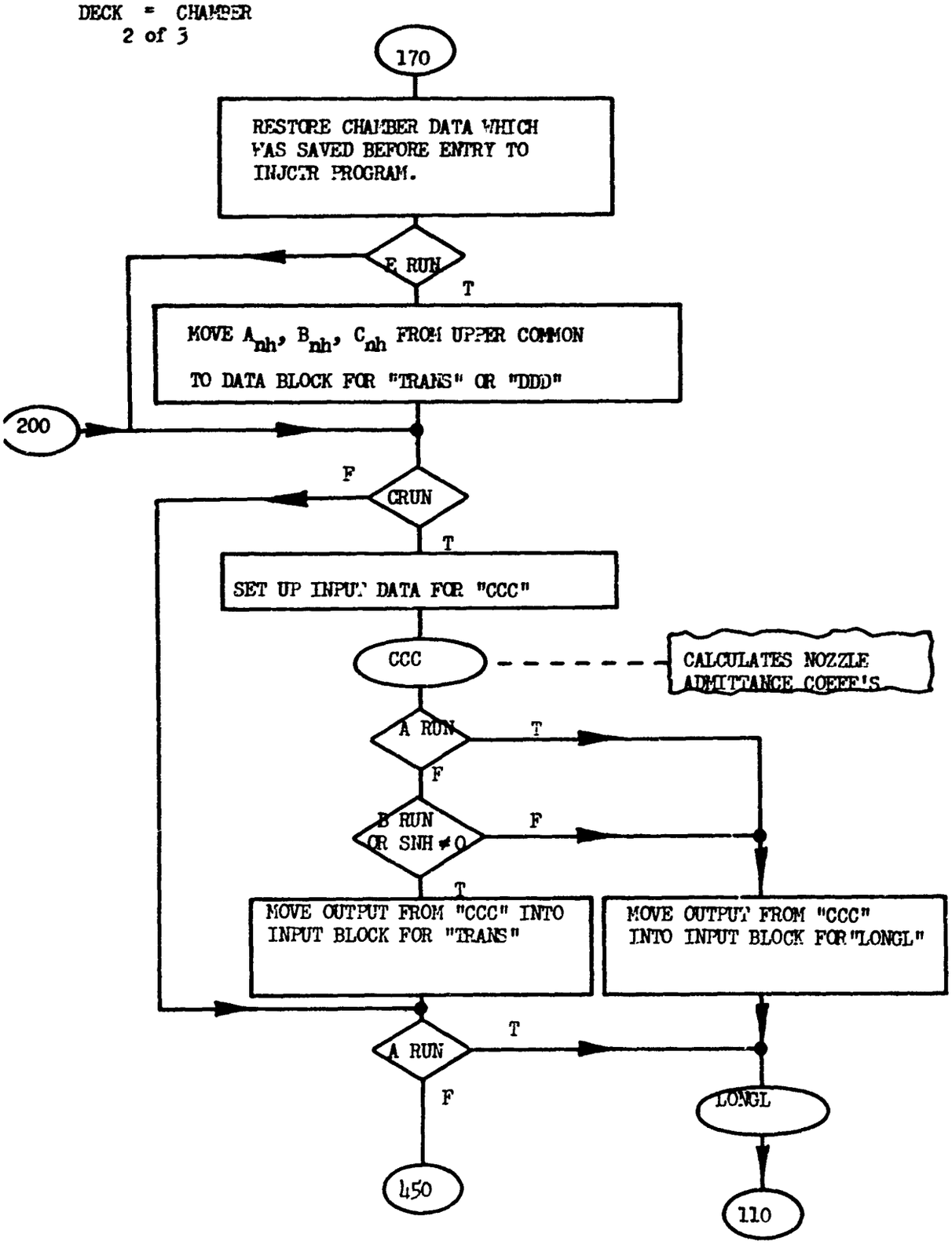
ENTRY = CHAMBR

DECK = CHAMBR

1 of 3

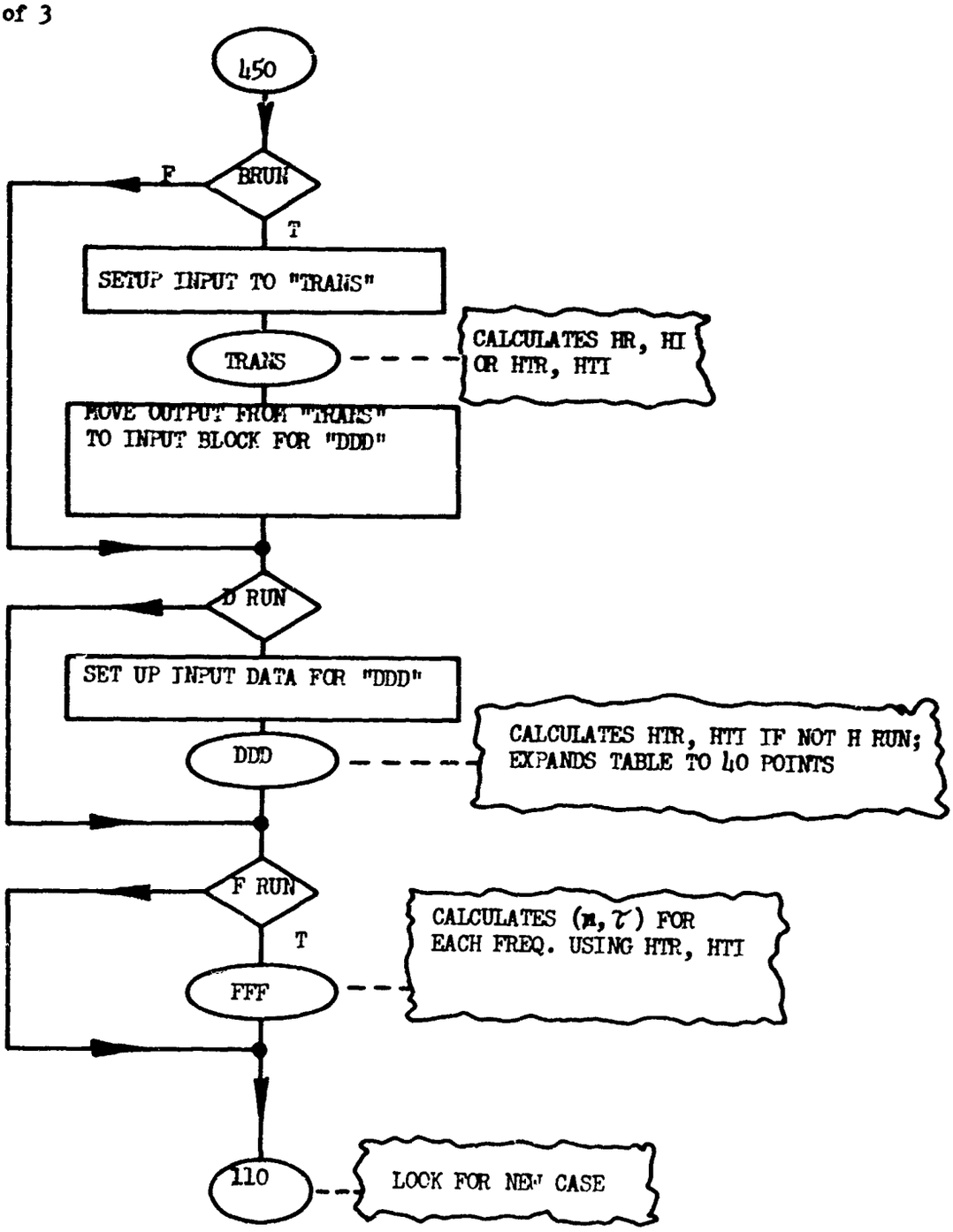


DECK = CHAMBER
2 of 3



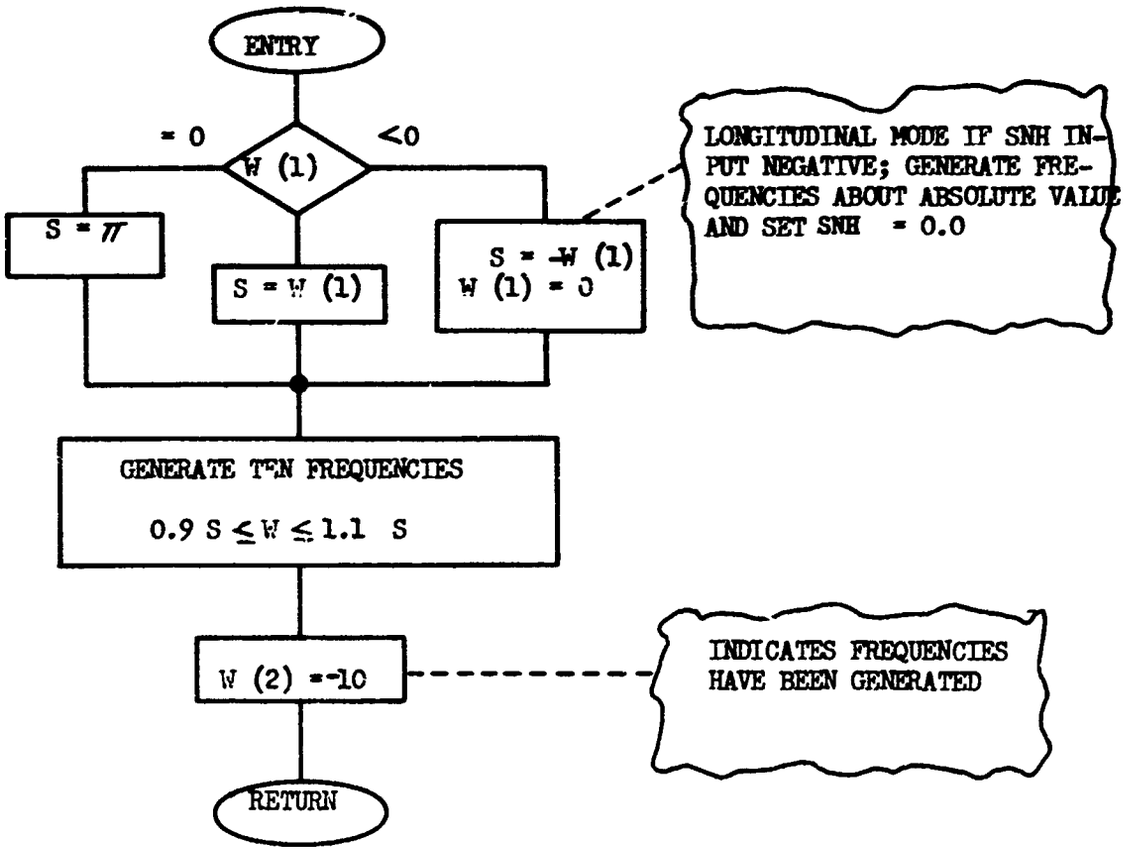
DECK = CHAMBER

3 of 3



ENTRY = GEMMEG

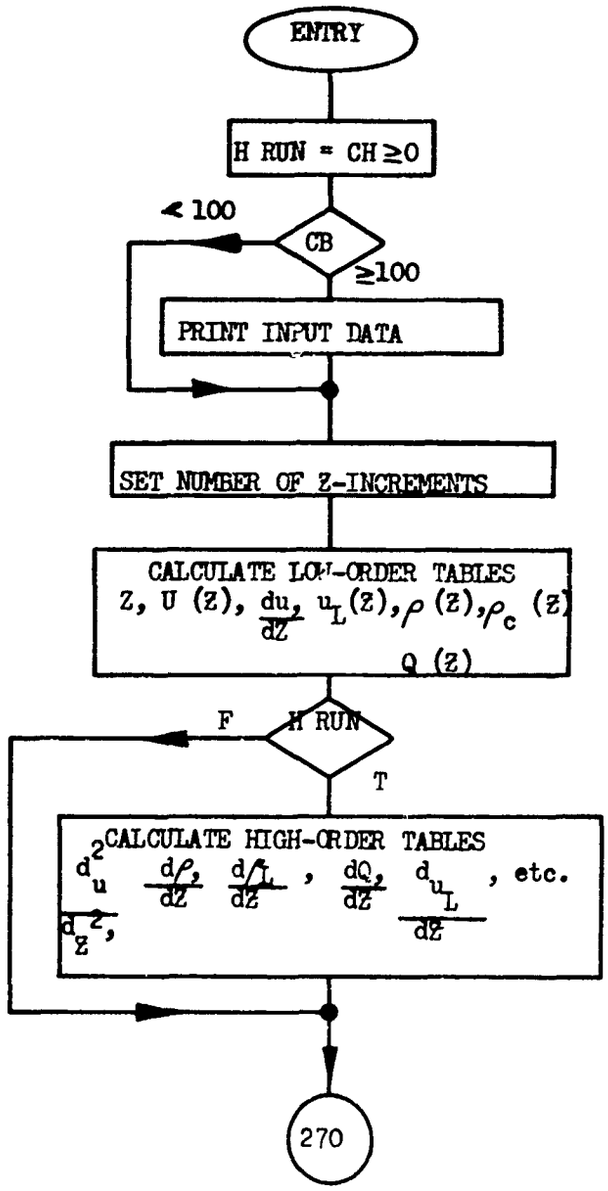
LOCK = WGEN



ENTRY = TRANS

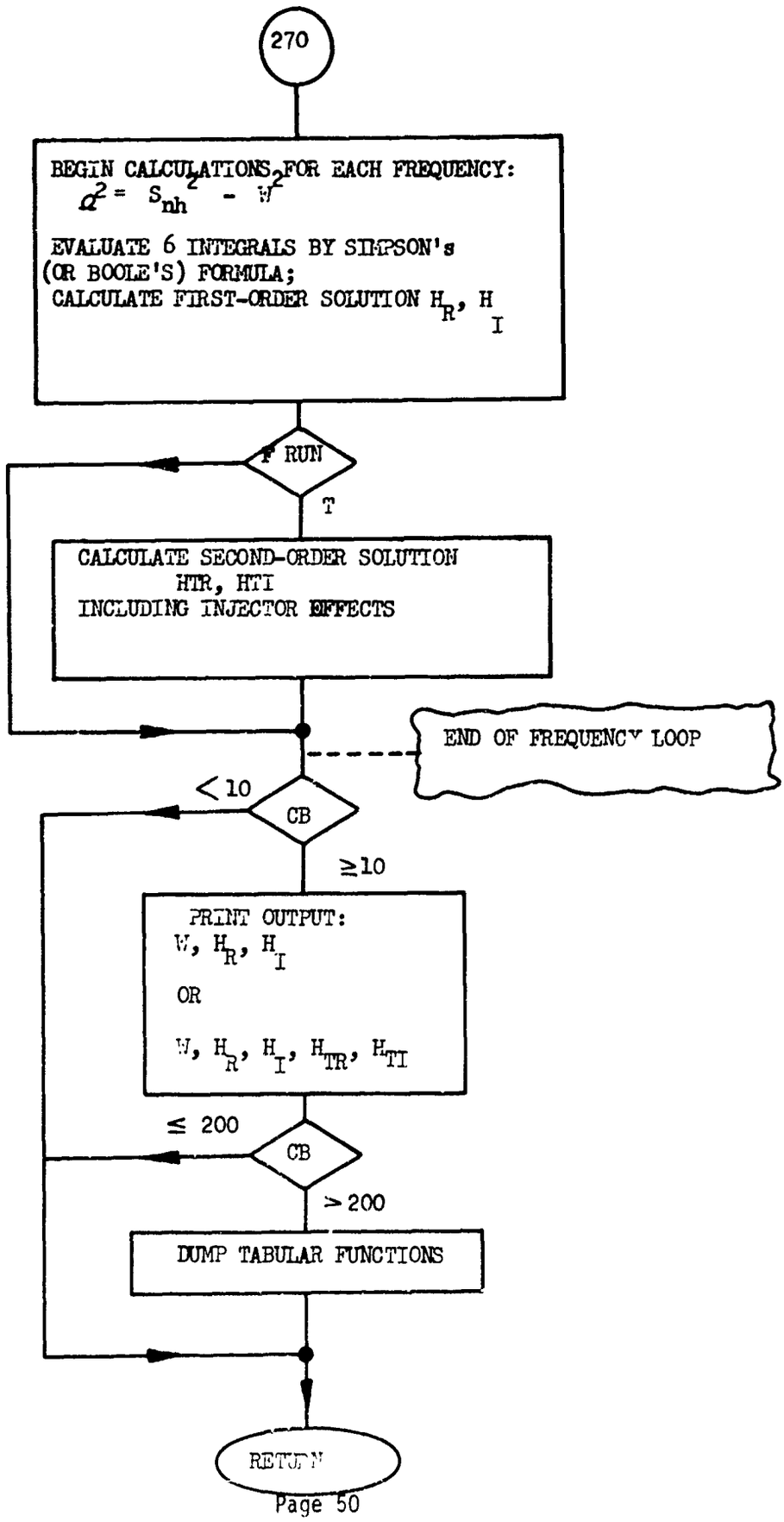
DECK = HYMN

1 of 2

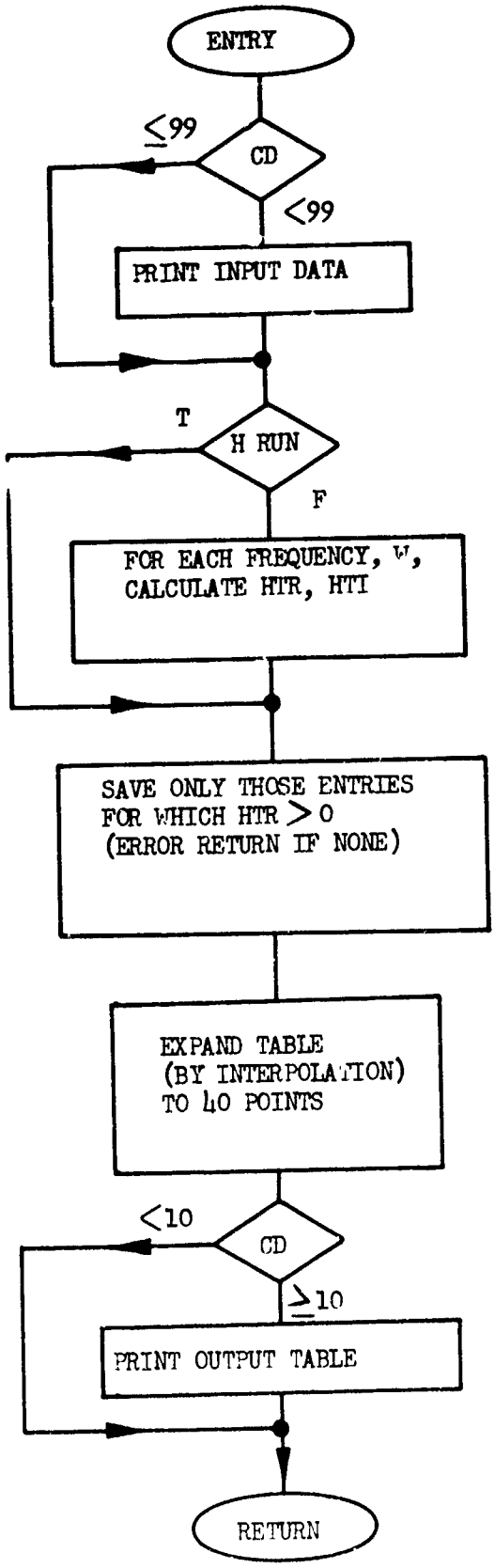


DECK = HYMN

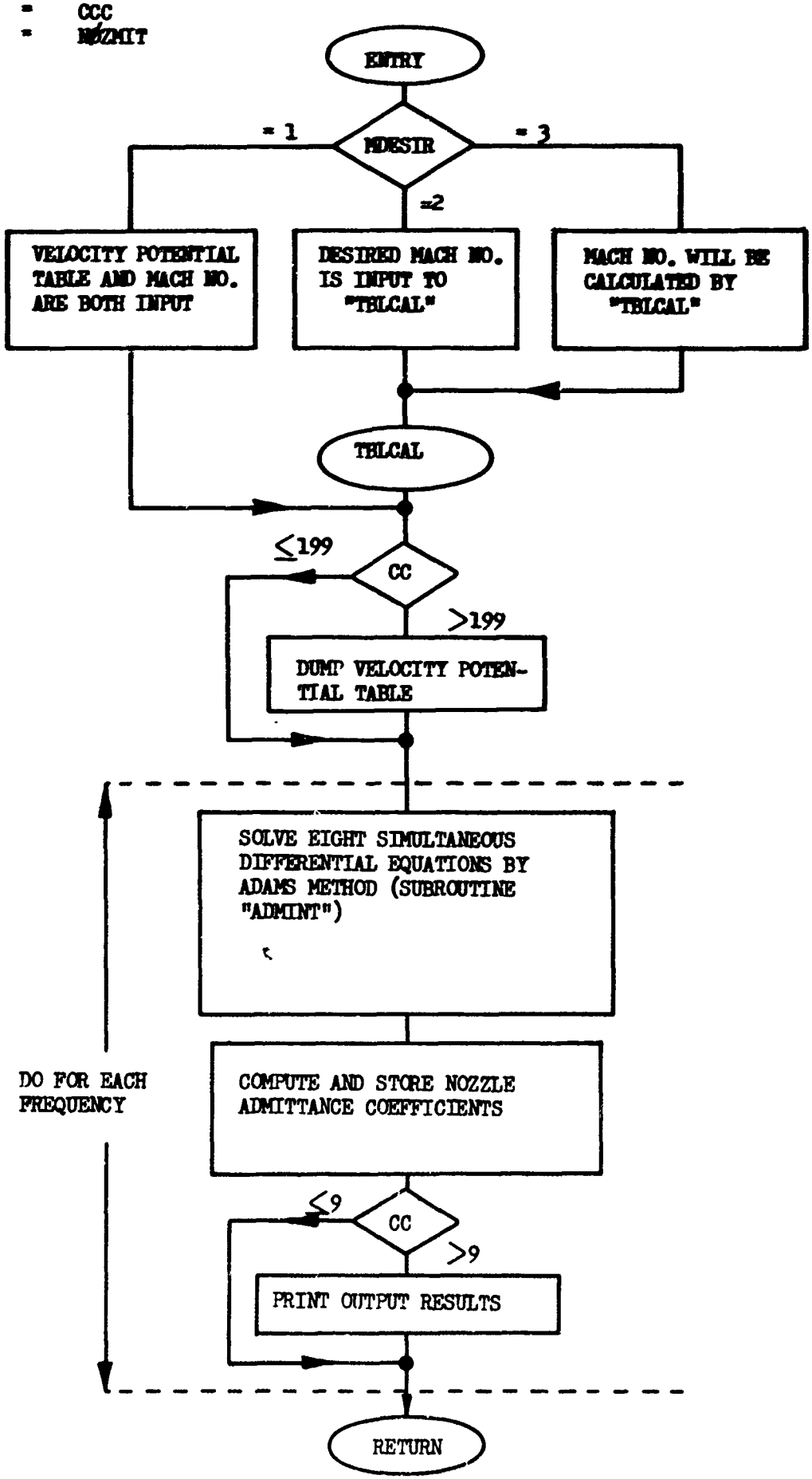
2 of 2



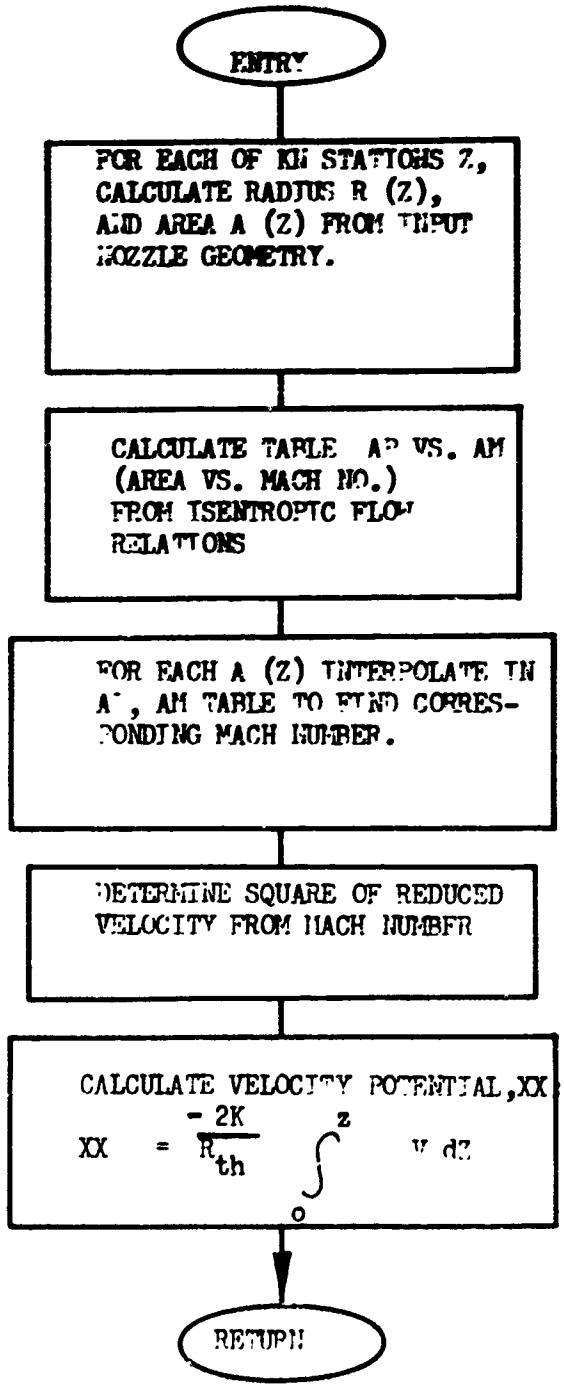
ENTRY = DDD
DECK = HTINT



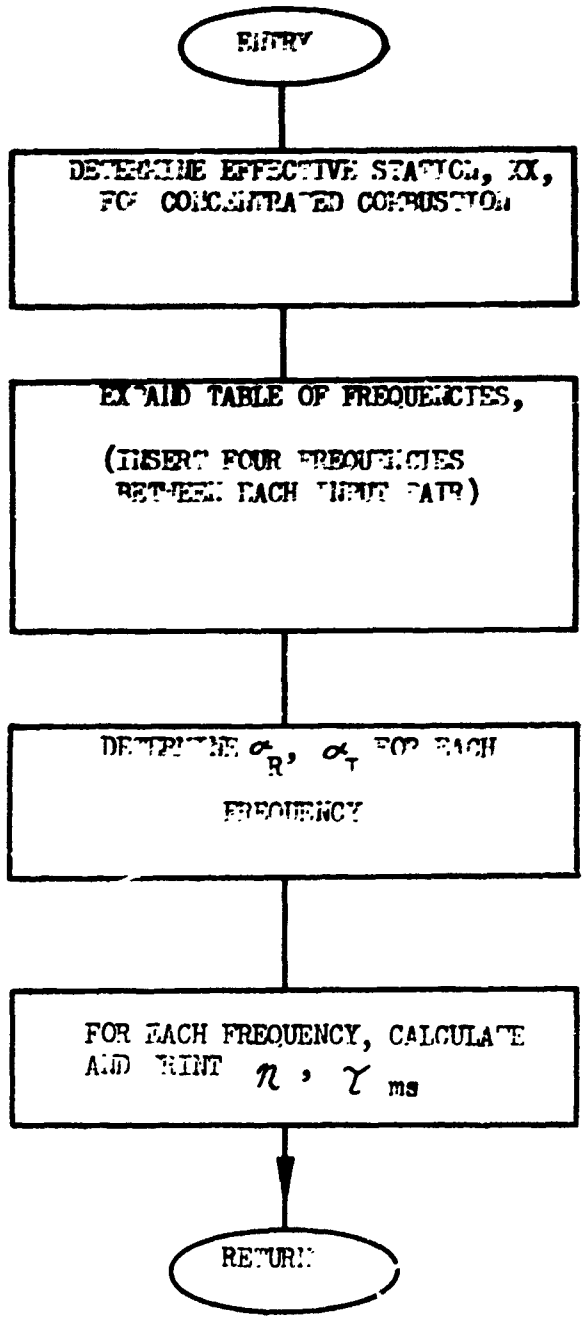
ENTRY = CCC
DECK = NØZMIT



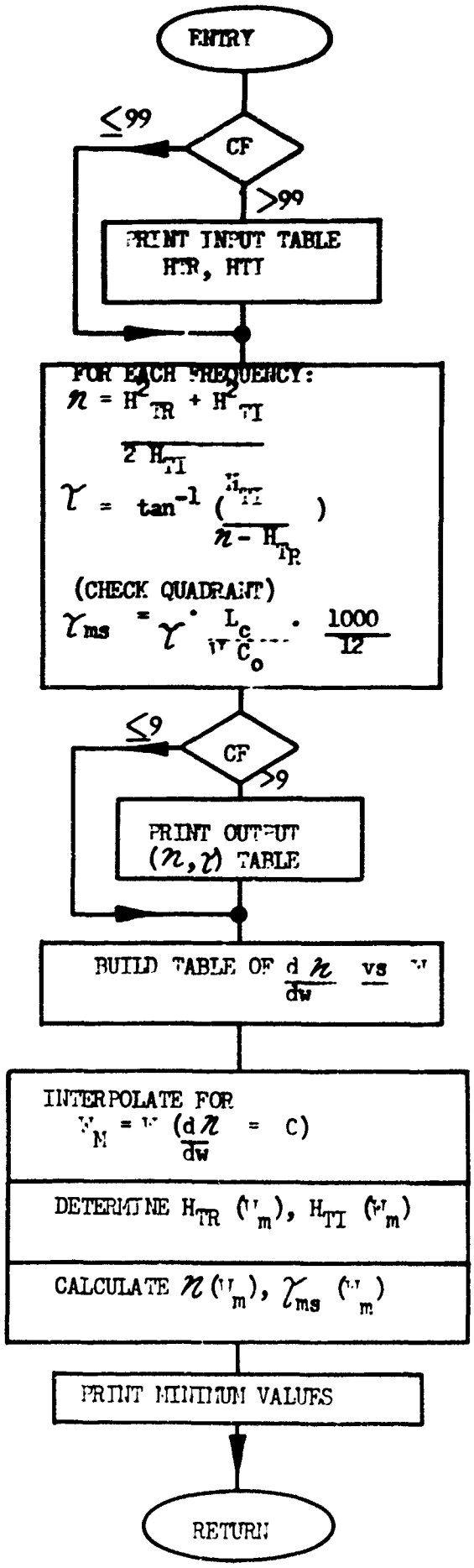
ENTRY = TBLCAL
DECK = VELPOT



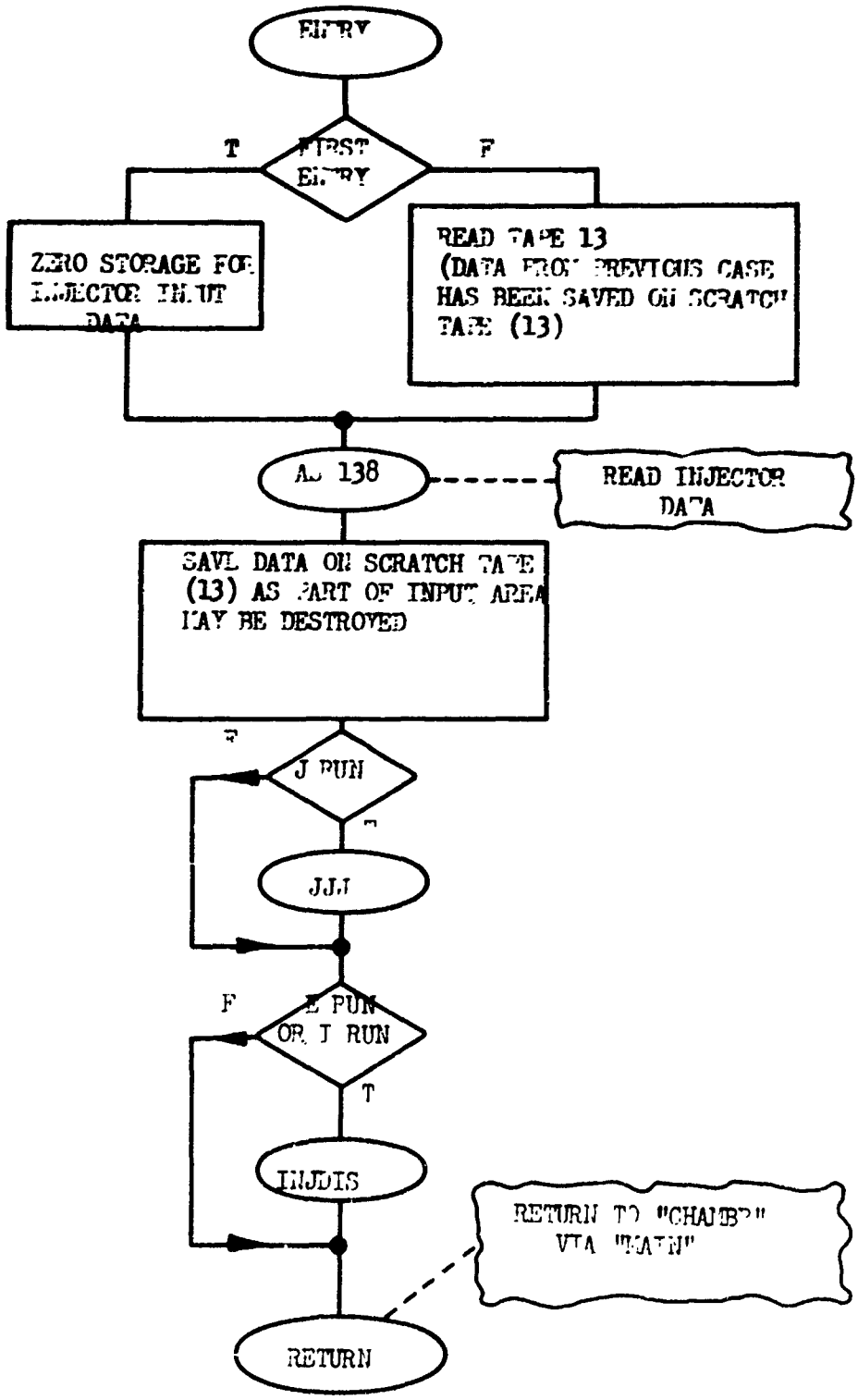
ENTRY = LØNGL
DECK = LØNGL



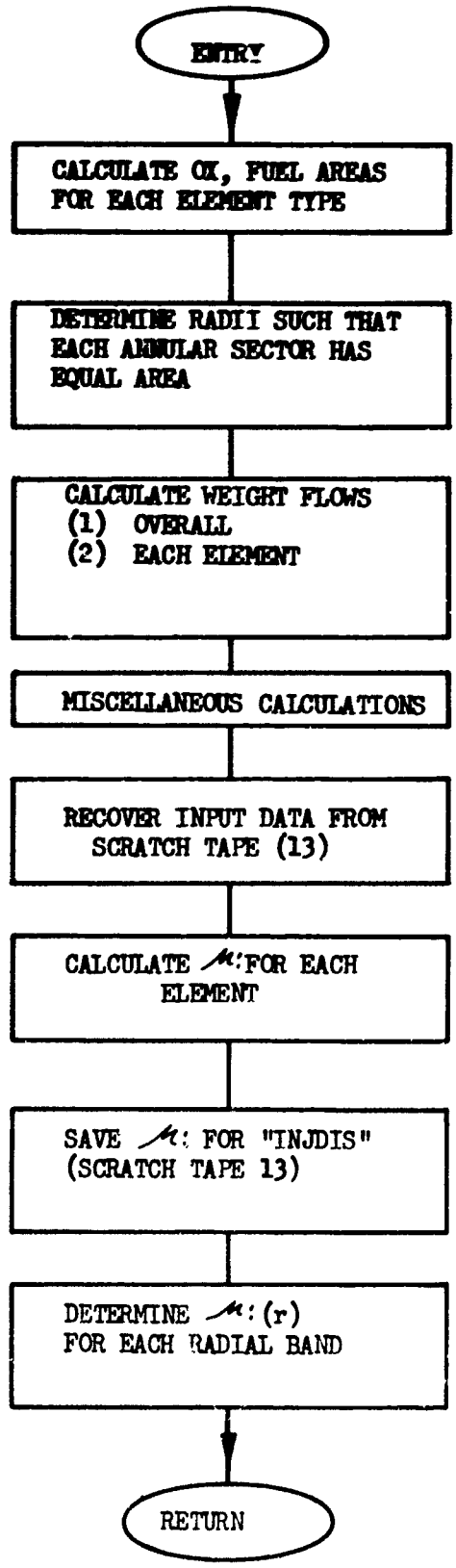
ENTRY = FFF
DECK = NTAU



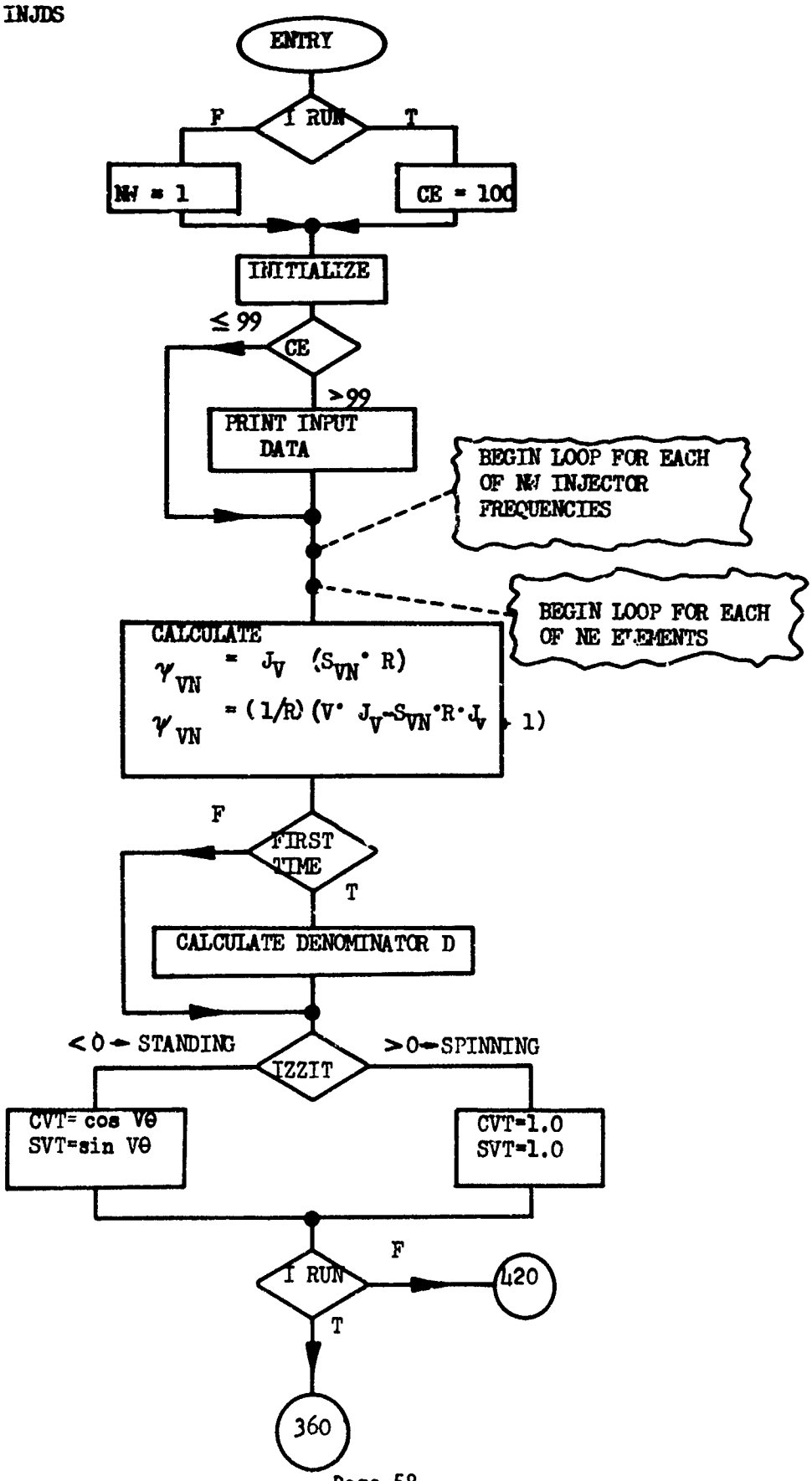
ENTRY = INJCTR
DECK = INJ



ENTRY = JJJ
DECK = JECT

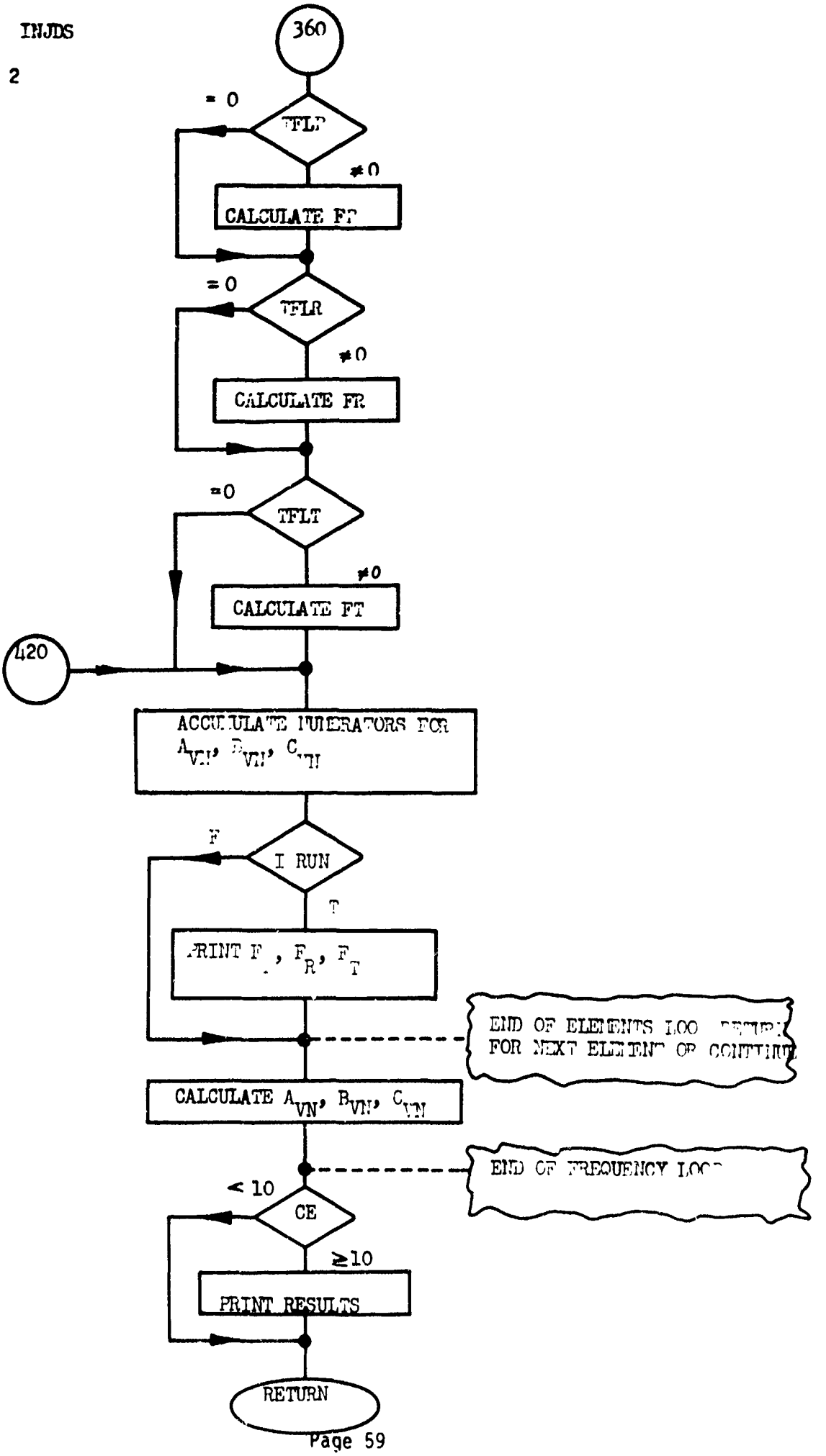


ENTRY = INJDIS
DECK = INJDS



DECK = INJDS

2 of 2



DOCUMENT CONTROL DATA - R&D
(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Aerojet-General Corporation	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE
The Sensitive Time Lag Theory and Its Application to Liquid Rocket Combustion Instability Problems

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Final Report, 2 Sept 1966 to 31 November 1967

5. AUTHOR(S) (Last name, first name, initial)
Smith, A. J., Jr; Reardon, F. H.; et al

6. REPORT DATE March 1968	7a. TOTAL NO. OF PAGES 388 (Vol I and II)	7b. NO. OF REFS 59
8a. CONTRACT OR GRANT NO. b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) AFRPL-TR-67-314 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None	

10. AVAILABILITY/LIMITATION NOTICES
This document ..... (see cover)

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Air Force Rocket Propulsion Laboratory Edwards Air Force Base, California

13. ABSTRACT

The main objective of this report is to include, under one cover, all of the work concerned with the development of the Sensitive Time Lag Theory of liquid rocket engine combustion instability. This work includes all the aspects of the theory from the mathematical formulation of the analytical model to the application of the model to actual engine problems. The initial section of the report reviews the logical considerations of the instability phenomenon and relates how the time lag concept conforms analytically as well as experimentally to the problem. Thereafter, the mathematics of the model are developed with the major emphasis placed on the linearized model; however, various aspects of the nonlinear model are also discussed. The mathematical analysis gives rise to a computer program, which is presented in Volume I in the form of an Engineer's instruction manual and in Volume II in the form of a detailed description for the Programmer. The report then focuses its attention on the designer and instructs him in not only how to use the model in practical situations but also how to interpret and correlate test data. The main body of the text concludes with a critique of the time lag concept and outlines the kind of research that is needed in order to improve the time lag model.