UNCLASSIFIED

AD NUMBER
AD907815
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; 29 DEC 1972. Other requests shall be referred to Air Force Flight Dynmaics Laboratory, Attn: PTB, Wright-Patterson AFB, OH 45433.
AUTHORITY
AFWAL ltr, 11 Feb 1980

THIS PAGE IS UNCLASSIFIED

AD 907815

AFFDL-TR-72-147 – VOL. II

PROPULSION SYSTEM INSTALLATION CORRECTIONS

VOLUME II:

PROGRAMMERS MANUAL

W. H. BALL
THE BOEING COMPANY

D D C
RECEIVED
MAR 6 1973
B

TECHNICAL REPORT AFFDL-TR-72-147 – VOL. II
DECEMBER 1972

Distribution limited to U.S. Government agencies only; test and evaluation; statement applied 29 December 1972. Other requests for this document must be referred to Air Force Flight Dynamics Laboratory (PTB), Wright-Patterson Air Force Base, Ohio 45433

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

Best
Available
Copy

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related hereto.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

PROPULSION SYSTEM INSTALLATION CORRECTIONS

VOLUME II:

PROGRAMMERS MANUAL

W. H. BALL

Distribution limited to U.S. Government agencies only, test and evaluation, statement applied 29 December 1972 Other requests for this document must be referred to Air Force Flight Dynamics Laboratory (PTB), Wright-Patterson Air Force Base, Ohio 45433

FOREWORD

This report was prepared by the Research and Engineering Division, Aerospace Group, of The Boeing Company under Air Force Contract F33615-72-C-1580, "Propulsion System Installation Corrections", Project 1366. The program was conducted under the direction of the Prototype Division, Air Force Flight Dynamics Laboratory, Air Force Systems Command. Mr. Gordon Tamplin was the Air Force Program Monitor.

The program was initiated on 31 December 1971 and draft copies of the final reports were submitted for approval on 31 October 1972.

Dr. P. A. Ross was Program Manager and Mr. W. H. Ball was principal investigator during the program. Significant contributions to the program were made by the following individuals: Mr. Joe Zeeben, engine performance; Dr. Franklin Marshall, inlet and exhaust system technology; Mr. John Petit, nozzle internal performance and nozzle/afterbody drag; and Mr. Gary Shurtleff, programming.

This report contains no classified information extracted from other classified reports.

Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

Ernest J. Cross Jr.

Lt. Col. Ernest J. Cross, Jr.
Chief, Prototype Division
Air Force Flight Dynamics Laboratory

ABSTRACT

This report presents the results of a research program to develop a procedure for calculating propulsion system installation losses. These losses include inlet and nozzle internal losses and external drag losses for a wide variety of subsonic and supersonic aircraft configurations up to Mach 4.5. The calculation procedure, which was largely developed from existing engineering procedures and experimental data, is suitable for preliminary studies of advanced aircraft configurations. Engineering descriptions, equations, and flow charts are provided to help in adapting the calculation procedures to digital computer routines. Many of the calculation procedures have already been programmed on the CDC 6600 computer. Program listings and flow charts are provided for the calculation procedures that have been programmed. The work accomplished during the program is contained in four separate volumes. Volume I contains an engineering description of the calculation procedures. Volume II is a programmer's manual containing flow charts, listings, and subroutine descriptions. Volume III contains sample calculations and sample input data. Volume IV contains bookkeeping definitions and data correlations.

PRECEDING PAGE BLANK-NOT FILMED

TABLE OF CONTENTS

SECTION		Page
I	GENERAL DESCRIPTION OF COMPUTER PROGRAM	1
II	DESCRIPTIONS OF SUBROUTINES AND LISTINGS	5
	MAIN PROGRAM (TEM 333)	5
	COMPUTE	13
	TABIN	21
	INLDRAG	26
	SIZINL	31
	AIRBYP	35
	AIRSPL	43
	DRG	47
	CDCONV	54
	PLUGMX	60
	PLUGNM	64
	ABINPT	68
	SUBBT	72
	SUPBT	78
	DRAGAB	82
	DBOATR	88
	CABTAB	92
	DINTFR	97
	DBASER	101
	DBTTB	105

TABLE OF CONTENTS (Continued)

SECTION		Page
	FIXDIM	107
	ATMOS	110
	PERF	114
	OUTGO	118
	WARNING	122
	STORE	125
	CHKRP	128
	GTHRST	131
	DATA	137
	PLACIN	140
	TABL 1	143
	TABL 2	146
	TABL 3	150
	TABL 22	154
	DSCRUB	157
III	DESCRIPTION OF FUNCTIONS AND LISTINGS	167
	AREA	167
	HOFT	171
	TOFH	173
	PROFH	175
	HOFPR	177
	COFH	180
	COFHS	183

TABLE OF CONTENTS (Continued)

SECTION		Page
	KOUNT	186
	DEMAND	189
	AREAF	192
	AEXHST	194
IV	CROSS LISTING OF PROGRAM TEM 333	196
V	INLET MAP PROGRAM	205

LIST OF ILLUSTRATIONS

Figure No.	Title	Page
1	Flow Chart for Program TEM 333	6
2	Flow Chart for Subroutine COMPUTE	14
3	Flow Chart for Subroutine INLDRAG	27
4	Flow Chart for Subroutine SIZINL	32
5	Flow Chart for Subroutine AIRBYP	36
6	Flow Chart for Subroutine AIRSPL	44
7	Flow Chart for Subroutine DRG	48
8	Flow Chart for Subroutine CDCONV	55
9	Flow Chart for Subroutine PLUGMX	61
10	Flow Chart for Subroutine PLUGNM	65
11	Flow Chart for Subroutine SUBBT	73
12	Flow Chart for Subroutine SUPBT	79
13	Flow Chart for Subroutine DRAGAB	84
14	Flow Chart for Subroutine DBOATR	89
15	Flow Chart for Subroutine CABTAB	93
16	Flow Chart for Subroutine DINTFR	98
17	Flow Chart for Subroutine DBASER	102
18	Flow Chart for Subroutine ATMOS	111
19	Flow Chart for Subroutine PERF	115
20	Flow Chart for Subroutine GTHRST	133

SYMBOLS AND NOMENCLATURE

A	Area, in^2
A_C	Inlet capture area, in^2
A_O	Local stream tube area ahead of the inlet, in^2
A_{O_I}	Free-stream tube area of air entering the inlet, in^2
AR	Aspect ratio, dimensionless
B	Velocity decay exponent, dimensionless
C	Sonic velocity, ft/sec.
C_D	Drag coefficient, $\frac{D}{qA_{REF}}$, dimensionless
C-D	Convergent-divergent
$C_{D_{ADD}}$	Additive drag coefficient, $C_{D_{ADD}} = \frac{D_{ADD}}{qA_C}$, dimensionless
$C_{D_{A_{MAX}}}$	Afterbody drag coefficient, $\frac{DRAG}{qA_{MAX}}$, dimensionless
$C_{D_{Base}}$	Base drag coefficient $\frac{(P_b - P_\infty)A_{BASE}}{qA_{MAX}}$, dimensionless
C_{D_S}	Scrubbing drag coefficient, $\frac{DRAG}{qA_{MAX}}$, dimensionless
C_{f_g}	Thrust coefficient, $\frac{F_g}{\frac{\dot{w}}{g}(V_{cp})}$, dimensionless
C_S	Stream thrust coefficient, dimensionless, (defined by Figure 48 of Volume IV)
C_V	Nozzle velocity coefficient, dimensionless
Conv.	Convergent
D	Drag, lb.; Hydraulic Diameter, $\frac{4A}{P}$, in., diameter, in.

SYMBOLS AND NOMENCLATURE (Continued)

F	Thrust, lb.
F_{g_i}	Ideal gross thrust (fully expanded), lb.
f/a	Fuel/air ratio, dimensionless
g	Gravitational constant, ft/sec^2
h	Enthalpy per unit mass, BTU/lb.; height, in.
h_{fan}	Enthalpy of fan discharge flow, BTU/lb
h_{pri}	Enthalpy of primary exhaust flow after heat addition BTU/lb
h_t	Throat height, in^2
K	Velocity decay coefficient, dimensionless
L	Length, in.
M	Mach number, dimensionless
P	Pressure, lb/in^2
P_r	Relative pressure; the ratio of the pressures p_a and p_b corresponding to the temperatures T_a and T_b, respectively, along a given isentrope, dimensionless
P_T	Total pressure, lb/in^2
Q	Effective heating value of fuel, BTU/lb.
q	Dynamic pressure, lb/in^2
R, r	Radius, in.
S	Nozzle centerline spacing, in.
T	Temperature, °R
V	Velocity, ft/sec

W	Mass flow, lb/sec
W_{BX}	Bleed air removed from engine, lb/sec.
W_C, $\frac{W\sqrt{\theta}}{\delta}$	Corrected airflow, lb/sec.
W_f	Weight flow rate of fuel, lb/sec.
W_2	Weight flow rate of air, primary plus secondary, lb/sec.
W_8	Primary nozzle airflow rate, lb/sec.
x	Length, in.
α	Angle of attack; convergence angle of nozzle, degrees
γ	Ratio of specific heats, dimensionless
ε	Diffuser loss coefficient, $\frac{\Delta P_T}{q}$, dimensionless
η_B	Burner efficiency, dimensionless
ν	Kinematic viscosity, ft^2/sec.
ρ	Density, lb/ft^3

SUBSCRIPTS

B	Burner
b, base	Base flow region
BP	Bypass
BLC	Boundary layer bleed
btail	Boattail
c	Core (nozzle); capture (inlet)
DES	Design conditions

SUBCRIPTS

e	Boattail trailing edge
EFF	Based on effective area
ENG	Refers to engine demand
f	Fuel
g	gross
GEOM	Based on geometric area
int	Interference; internal
ip	Ideal, primary exhaust flow
jet	Exhaust flow jet
max	Maximum
min	Minimum
s	Scrubbing flow region
SPILL	Spillage
T	Total condition; throat station
t_f	Total condition, fan flow
T/O	Takeoff
t_p	Total condition, primary flow
o	Local conditions ahead of inlet
2	Compressor face station
8	Nozzle throat station
9	Nozzle exit station
18	Fan discharge throat station
∞	Free-stream condition
x	Local

SECTION I

GENERAL DESCRIPTION OF COMPUTER PROGRAM

The engine installation calculation procedure naturally breaks into three divisions: inlet, engine, and nozzle/afterbody. Each of these divisions is represented in TEM-333 by a set of subprograms logically subordinate to a monitoring subprogram which functions as a quasi-main program for that set. The term module will be used to designate a quasi-main and its subordinate programs. Further, within a module procedures are subprogrammed to isolate the procedures from one another and to isolate logical direction programming from computational programming. The benefits of this organization are threefold. First, checkout. Each module can be checked out independently of the others. Secondly, logical simplicity. The logical structure of the program as a whole is more apparent. Individual subprograms are simpler; being small with a limited purpose. Finally, ease of modification. Engine installation programs are constantly undergoing modification so the ability to attach and detach new modules or procedures quickly is a necessity. The small subprogram-module arrangement allows for quick change since the connection to the next level is more apparent.

The modules are tied together by subroutine CØMPUTE which calculates the installed performance for a single uninstalled performance point. The main program, TEM-333, handles calls to the input routines, then reads the uninstalled performance points one-by-one; calling CØMPUTE for each, outputting data when required until the last point has been computed.

Programs included in TEM-333 and their purposes are:

TEM-333 - Main Program, monitors program logic

COMPUTE - Controls calculation of a performance point

<u>Inlet Module</u>

TABIN - Reads Inlet Input Deck

INLDRAG - Monitors Inlet Drag Computation

AIRBYP - Computes External-Compression Mode

AIRSPL - Computes Mixed Compression Mode

SIZINL - Computes Inlet Capture Area

DEMAND - Computes Engine Demand

AREAF - Computes F_1 where $A_O = F_1 W_C^{\eta}$

<u>Afterbody Module</u>

ABINPT - Reads Nozzle Input Deck

DRAGAB - Monitors Afterbody Drag Calculations

DINTFR - Computes Nozzle Interference Drag

DBØATR - Monitors Boattail Drag Computation

DBASER - Computes Base Drag

DBTTB - Computes $C_{D_{BT}} = f(M, PS)$

CDCØNV - Monitors C-D Nozzle Boattail Computation

PLUGMX - Computes Boattail Drag for Plug Nozzle, Mixed Flow

PLUGNM - Computes Boattail Drag for Plug Nozzle, Non-mixed Flow

DRG - Computes Boattail Drag for C-D Nozzle

SUBBT - Computes Subsonic $C_{D_{BT}}$ for C-D Nozzle

SUPBT - Computes Supersonic $C_{D_{BT}}$ for C-D Nozzle

CABTAB - Computes $C_{D_{BT}}$ as Function of A_{10}/A_9

AEXHST - Computes Nozzle Exit Area

<u>Engine Module</u>

PERF - Computes Gross Thrust

CHKRP - Computes Thermodynamic Properties at Throat

GTHRST - Computes Performance of Propulsive Nozzles

HØFT - Computes Enthalpy Given Temperature

TØFH - Computes Temperature Given Enthalpy

PRØFH - Computes Relative Pressure Given Enthalpy

HØFPR - Computes Enthalpy Given Relative Pressure

CØFH - Computes Critical Velocity Given Total Enthalpy

CØFHT - Computes Critical Velocity Given Static Enthalpy

Miscellaneous Programs

STØRE - Stores results of a performance point

ØUTGØ - Outputs an engine performance matrix

WARNING - Prints warning message heading

FIXDIM - Determine dimensions of A8 and A9 input

ATMØS - Computes atmospheric data

DATA - Transfers and output input card images

Tabular Input Programs

TABL1 - Inputs table format 1 tables

TABL2 - Inputs table format 2Q tables

TABL22 - Inputs table format 2K tables

TABL3 - Inputs table format 3 tables

PLACIN - Reads a tabular array

KØUNT - Counts non-blank input fields

Linear Interpolation Programs

TABU1 - Interpolates F = f(x)

TABU2 - Interpolates F = f(x,y)

TABU3 - Interpolates F = f(x,y,z)

IFIND - Finds Interpolation Interval

FACINT - Computes Interpolation Factor

CVINT - Perform Linear Interpolation Calculation

SECTION II

DESCRIPTIONS OF SUBROUTINES AND LISTINGS

SUBJECT:	PRØGRAM TEM-333
PURPØSE:	Main Program TEM-333 monitors the overall program logic.
METHØD:	All supporting data is input, then the program reads performance points one by one, computes and stores the results, and outputs an array of results whenever Mach or altitude changes.
USAGE:	PRØGRAM TEM333 (INPUT, ØUTPUT, TAPE5, TAPE6 = ØUTPUT, TAPE1 = INPUT)

INPUT or TAPE1	System Input
ØUTPUT or TAPE6	System output
TAPE5	Card Images of Input
TAPE7	MARK II output

SUBPRØGRAMS:	ABINPT	DATA	FIXDIM
	ØUTGØ	STØRE	TABIN
	TABL1	TABL2	

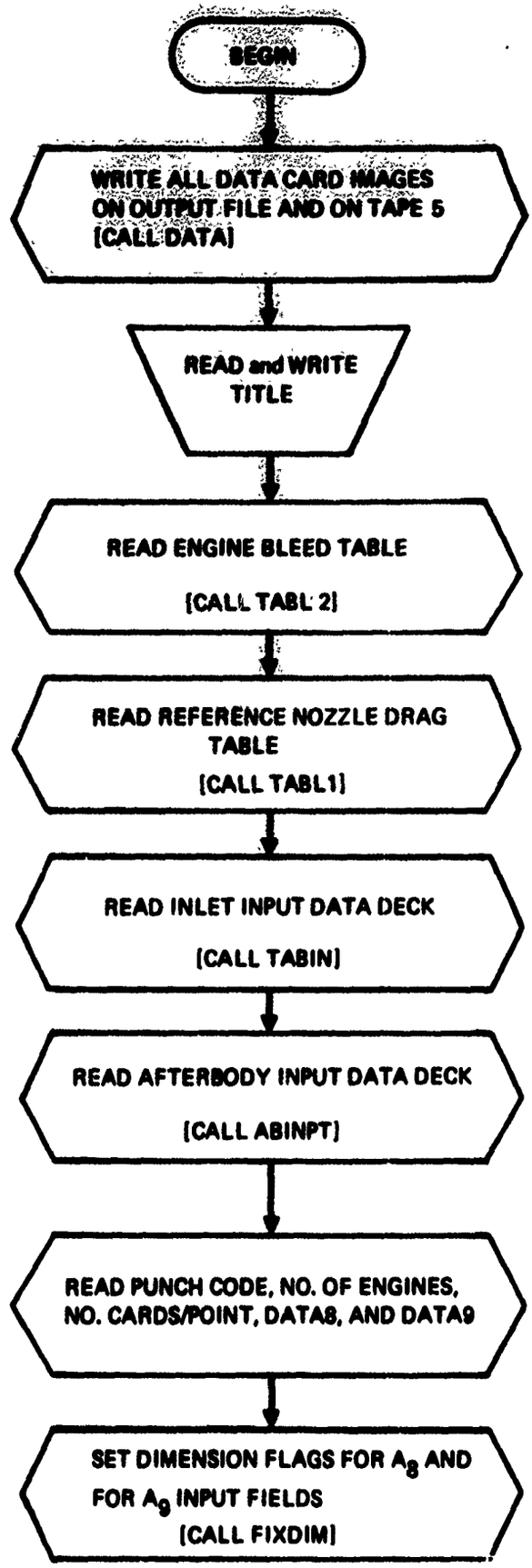


Figure 1: FLOW CHART FOR PROGRAM TEM 333

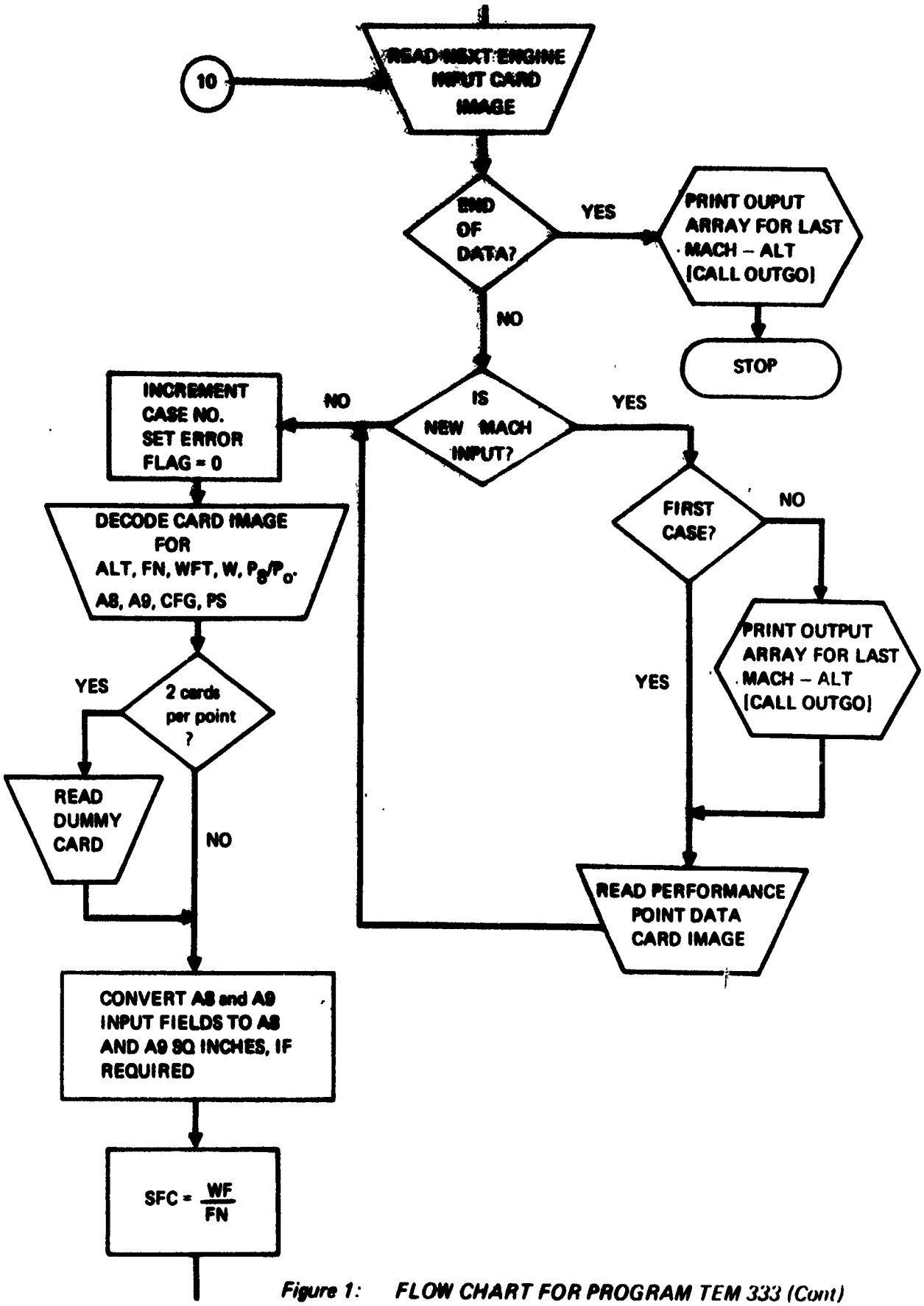


Figure 1: FLOW CHART FOR PROGRAM TEM 333 (Cont)

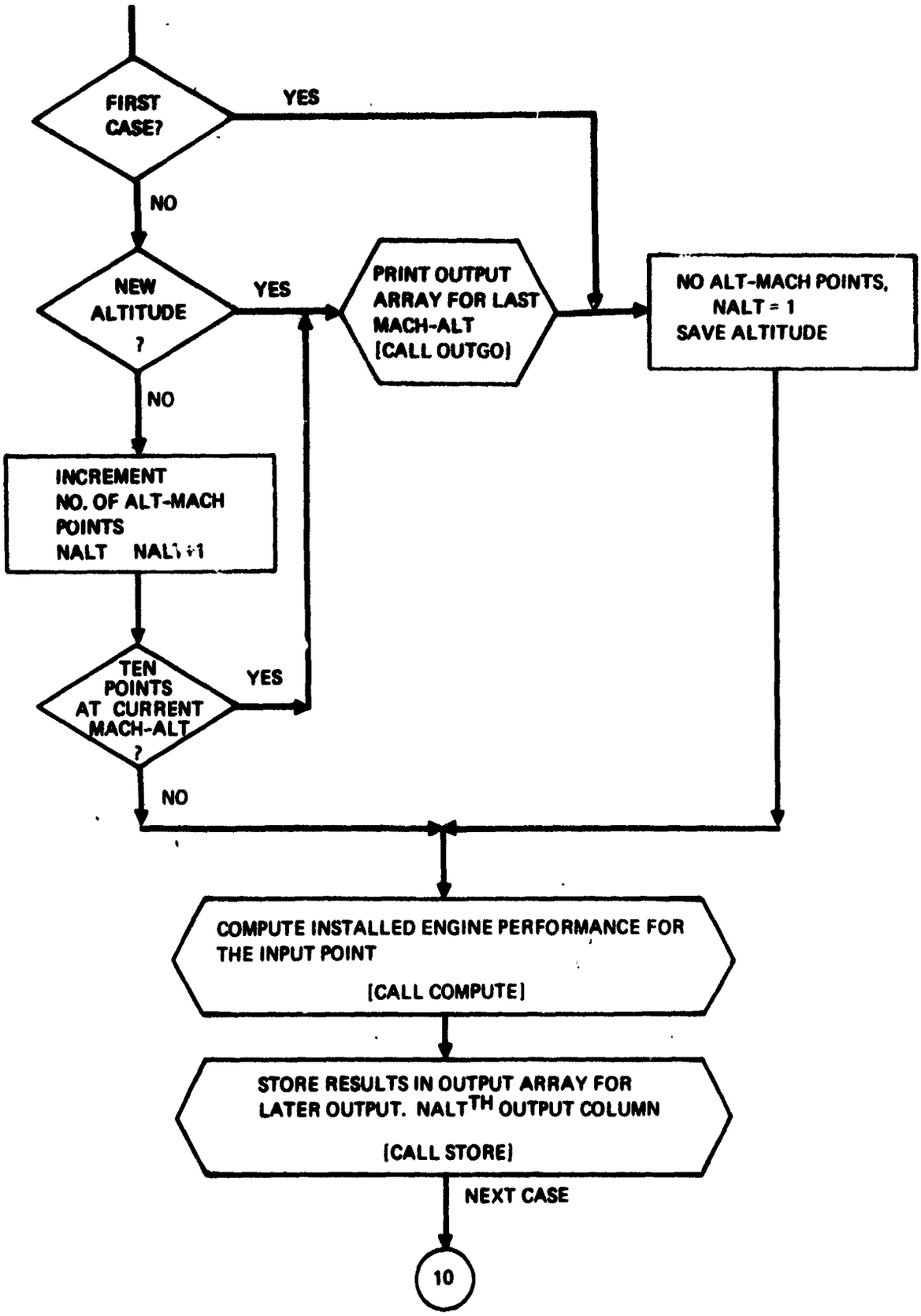


Figure 1: FLOW CHART FOR PROGRAM TEM 333 (Concluded)

```
S*BJAML.  0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
S*BJAML.  0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
```



```
                PROGRAM TEM333(INPUT,OUTPUT,TAPE5,TAPE6=OUTPUT,TAPE1=INPUT,
               X TAPE7)                                                    INDRA020
          C                                                                INDRA030
000003          COMMON/ENGNO/ENGNO,XONI,XOBASE
          C     INITIALIZATION OF PHYSICAL CONSTANTS                       INDRA130
          C        G , GRAVITY ACCELERATION = 32.174 FT/SEC**2             INDRA140
          C        GAMMA , SPECIFIC HEAT RATIO CP/CV = 1.40 FOR AIR        INDRA150
          C        R - GAS CONSTANT = 53.34 FT LB/LB DEG F(ABS)            INDRA160
          C     PROGRAMMING INITIALIZATION                                 INDRA180
000003          COMMON /MACHW8/MACHW8(120),NKW8,NYW8
000003          COMMON/UNREF/UNREFM(20),NUNREF
000003          COMMON/INS/ MACH,OLDALT,CON
000003          COMMON/GAMMA/GAMMA
000003          COMMON/G/G
000003          COMMON/R/R
000003          COMMON/TAPENO/IN,IOUT                                      INDRA190
000003          COMMON/IBAD/IBAD                                           INDRA200
000003          COMMON/IDATA/IDATA                                         INDRA210
000003          COMMON /OUTS/ CASE, ALT, PS, FNA, WFTRF, SFCOUT, FNRF, FRAM, RF,
               1              MILRF, DINLET, CDSPL, CDBLD, CDBYP, CDINL, DNOZ,
               2              CDUT, CDBASE, CDINT, DNOZRF, P8P0, A9, A8, CFG,
               3              BETA, FN, WFT, SFC, W, WABS, FNCOR, WFCOR, SFCCOR,
               4FAM8, P8M8, TTC, Q, ADEAC,ADIAC,AJAC,STATUS,EXTRA(4)
000003          COMMON/TITLE/TITLE(8)
000003          COMMON/PCODE/PCODE
000003          COMMON/DATE/DA,TE
000003          COMMON/CARD/CARD(5)                                        INDRA270
000003          COMMON/KTRAN/KTRAN                                         INDRA280
000003          COMMON/NEWM/NEWM                                           INDRA290
000003          REAL MILRF
000003          REAL MACH
000003          DATA G/32.174/, GAMMA/1.40/, R/53.34/                      INDRA170
000003          DATA BLANK/10H         /                                   INDRA300
000003          DATA IN,IOUT/5,6/                                          INDRA310
000003          DATA OLDALT/-967654./                                      INDRA320
000003          DATA CASE/0.0/                                             INDRA330
000003          DATA NMACH/1/                                              INDRA340
000003          DATA PI8/4./.78540/
000003          DATA MASK20 /10HONE CARD   /
000003          DATA NALT/1/                                               INDRA350
          C                                                                INDRA410
          C     BEGIN EXECUTION OF PROGRAM                                 INDRA420
000003          CALL DATA                                                  INDRA480
          C                                                                INDRA490
          C     CALL DATE(DA,TE)
000034          READ(5,500) TITLE
000012          WRITE(6,504) TITLE
000020          WRITE(6,507)
000024      607 FORMAT(*0W8 TABLE*)
000024          CALL TABL2(MACHW8(1),10,10,NKW8,NYW8)
000030          WRITE(6,508)
000034          CALL TABL1(UNREFM,10,NUNREF)
```

```
S+8JAHL.  0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
S+8JAHL.  0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000

RUN  VERSION JUL 71           22.46.30. 72/08/23.                                                         PAGE 001

                   PROGRAM TEM333(INPUT,OUTPUT,TAPE5,TAPE6=OUTPUT,TAPE1=INPUT,
                  X TAPE7)
            C                                                                       INDRAG20
 000003            COMMON/ENGNO/ENGNO,XONI,XOBASE                                   INDRAG30
            C      INITIALIZATION OF PHYSICAL CONSTANTS                             INDRA130
            C         G , GRAVITY ACCELERATION = 32.174 FT/SEC**2                   INDRA140
            C        GAMMA , SPECIFIC HEAT RATIO CP/CV = 1.40 FOR AIR               INDRA150
            C         R , GAS CONSTANT = 53.34 FT LB/LB DEG F(ABS)                  INDRA160
            C      PROGRAMMING INITIALIZATION                                       INDRA180
 000003            COMMON /MACHW8/MACHW8(120),NXW8,NYW8
 000003            COMMON/NREF/ONREFM(20),NONREF
 000003            COMMON/INS/ MACH,OLDALT,CON
 000003            COMMON/GAMMA/GAMMA
 000003            COMMON/G/G
 000003            COMMON/R/R
 000003            COMMON/TAPENO/IN,IOUT
 000003            COMMON/IBAD/IBAD                                                 INDRA190
 000003            COMMON/IDATA/IDATA                                               INDRA200
 000003            COMMON /OUTS/ CASE, ALT, PS, FNA, WFTRF, SFCOUT, FNRF, FRAM, RF, INDRA210
                  1          MILRF, DINLET, COSPL, COBLD, COBYP, COINL, ONOZ,
                  2          COUT, COBASE, COINT, ONOZRF, P8P0, A9, A8, CFG,
                  3          BETA, FN, WFT, SFC, W, WABS, FNCOR, WFCOR, SFCCOR,
                  4TAMB, PAMB, TTC, Q, AOEAC,AOIAC,AJAC,STATUS,EXTRA(4)
 000003            COMMON/TITLE/TITLE(8)
 000003            COMMON/PCODE/PCODE
 000003            COMMON/DATE/DA,TE
 000003            COMMON/CARD/CARD(8)
 000003            COMMON/KTRAN/KTRAN                                               INDRA270
 000003            COMMON/NEWM/NEWM                                                 INDRA280
 000003            REAL MILRF                                                       INDRA290
 000003            REAL MACH
 000003            DATA G/32.174/, GAMMA/1.40/, R/53.34/                            INDRA
 000003            DATA BLANK/10H          /                                        INDRA300
 000003            DATA IN,IOUT/5,6/                                                INDRA310
 000003            DATA OLDALT/-957654./                                            INDRA320
 000003            DATA CASE/0.0/                                                   INDRA330
 000003            DATA NMACH/1/                                                    INDRA340
 000003            DATA PI3/4/.78540/
 000003            DATA MASK20 /10HONE CARD   /
 000003            DATA NALT/1/                                                     INDRA350
            C                                                                       INDRA410
            C      BEGIN EXECUTION OF PROGRAM                                       INDRA420
 000003            CALL DATA                                                        INDRA450
            C                                                                       INDRA490
            C      CALL DATE(DA,TE)
 000014            READ(5,500) TITLE
 000012            WRITE(6,504) TITLE
 000020            WRITE(6,507)
 000024       607  FORMAT(*1W8 TABLE*)
 000024            CALL TABL2(MACHW8(1),10,10,NXW8,NYW8)
 000030            WRITE(6,508)
 000034            CALL TABL1(ONREFM,10,NONREF)
```

```
000304          CALL STORE(NALT)
000306          GO TO 10                                          INDR1080
000307     7777 CALL OUTSO(NALT)
000311          CALL EXIT                                         INDR1100
000312      600 FORMAT(6A10)                                      INDR1110
000312      601 FORMAT(F7.0,3X)                                   INDR1120
000312      602 FORMAT(7X,9F7.0)                                  INDR1130
000312      603 FORMAT(10F7.0)                                    INDR1140
000312      604 FORMAT(1H1,45X,*TEM-333 ENGINE PERFORMANCE PROGRAM*/1H0,20X,6A10)
000312      606 FORMAT(*OPCODE* F4.1,5X,*NO. ENGINES*,F4.1,5X,A10,* INPUT/POINT*
               A 2F20.1)
000312          END                                               INDR1150
```

PROGRAM LENGTH INCLUDING I/O BUFFERS
010527

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

10	-	000125	15	-	000136	20	-	000153	40	-	000176
50	-	000255	55	-	000257	60	-	000266	70	-	000272
80	-	000277	90	-	000303	600	-	000345	601	-	000347
602	-	000352	603	-	000355	604	-	000357	606	-	000367
607	-	000330	608	-	000334	7777	-	000307			

EXTERNALS AND TAGS

DATA	-	500200	INPUTC.-	500300	OUTPTC.-	500400	TABL2	-	500500		
TABL1	-	500600	TABIN	-	500700	ABINPT	-	501000	FIXDIM	-	501100
IFENDF.-		501200	INPUTS.-	501300	OUTSO	-	501400	COMPUTE-	501500		
STORE	-	501600	EXIT	-	501700	END.	-	502000	QENTRY.-	500100	

BLOCK NAMES AND LENGTHS

ENGNO	-	000003	MACHW8	-	000172	DNREF	-	000025	INS	-	000003
GAMMA	-	000001	G	-	000001	R	-	000031	TAPEND	-	000002
IBAD	-	000001	IDATA	-	000001	OUTS	-	000055	TITLE	-	000010
PCODE	-	000001	DATE	-	000002	CARD	-	000010	KTRAN	-	000001
NEWM	-	000001									

VARIABLE ASSIGNMENTS

ALT	-	000001C13	A8	-	000026C13	A9	-	000025C13	BLANK	-	000376
CARD	-	000000C17	CASE	-	000000C13	CF6	-	000027C13	CON	-	000002C04
DATA3	-	000004	DATA9	-	000005	DNREFM	-	000000C03	ENGNO	-	000000C01
EXTRA	-	000001C13	FN	-	000031C13	G	-	000000C06	GAMMA	-	000000C05
IDATA	-	000000C12	IN	-	000000C10	IOUT	-	000001C10	J8	-	000406
J9	-	000411	KASE	-	000415	K8	-	000407	K9	-	000412
L8	-	000410	L9	-	000413	MAC4	-	000000C04	MACHW8	-	000000C02
MASK2C	-	000001	MILRF	-	000011C13	NALT	-	000402	NONREF	-	000024C03
NEWM	-	000000C21	NMACH	-	000377	NXW3	-	000170C12	NYW8	-	000171C02
OLDALT	-	000001C04	PCODE	-	000000C15	PIBY4	-	000400	PS	-	000002C13
PBPI	-	000024C13	R	-	000000C07	SFC	-	000033C13	STATUS	-	000050C13
TITLE	-	000000C14	TWOPRU	-	000403	W	-	000034C13	WFT	-	000032C13
X	-	000014	XUBASE	-	000002C01	XONI	-	000001C01	ZEEBEN	-	000416

START OF CONSTANTS
000313

START OF TEMPORARIES
000376

START OF INDIRECTS
000376

SPACE REQUIRED TO COMPILE
037300

SUBJECT: SUBRØUTINE CØMPUTE

PURPØSE: SUBRØUTINE CØMPUTE controls the calculation of the installed performance for a single input case.

METHØD: Using the input performance data CØMPUTE first calculates atmospheric data and the reference recovery factor (MIL STD 5008 B). Gross thrust is computed for the input W_F, W_O, and P_8/P_O.

The inlet module is called to find inlet drag and recovery factor. Then a second gross thrust calculation is made using values corrected by the recovery ratio. The input array to the afterbody module is loaded and the module called to compute afterbody drags. Reference afterbody drag is obtained by interpolation. Now installed thrust is calculated by combining the appropriate terms and the total calculation is complete.

USAGE: CALL CØMPUTE

All inputs are in labeled CØMMØN blocks.

Input variables are:

Mach, ALT, PS, P_8/P_O, A_9, A_8, F_N, W_{F_T},

SFC, W_{O_C}, Table 2.1, W_{BLD} = f(Mach, ALT),

and Table 2.2, $C_{D_{AB_{REF}}}$ = f(Mach).

All of these input variables are direct input to the program or immediate consequences of the input, e.g.

SFC = W_F/F_N.

SUBPRØGRAMS:

ATMØS	DRAGAB	INLDRAG
PERF	SQRT	TABU1
TABU2		

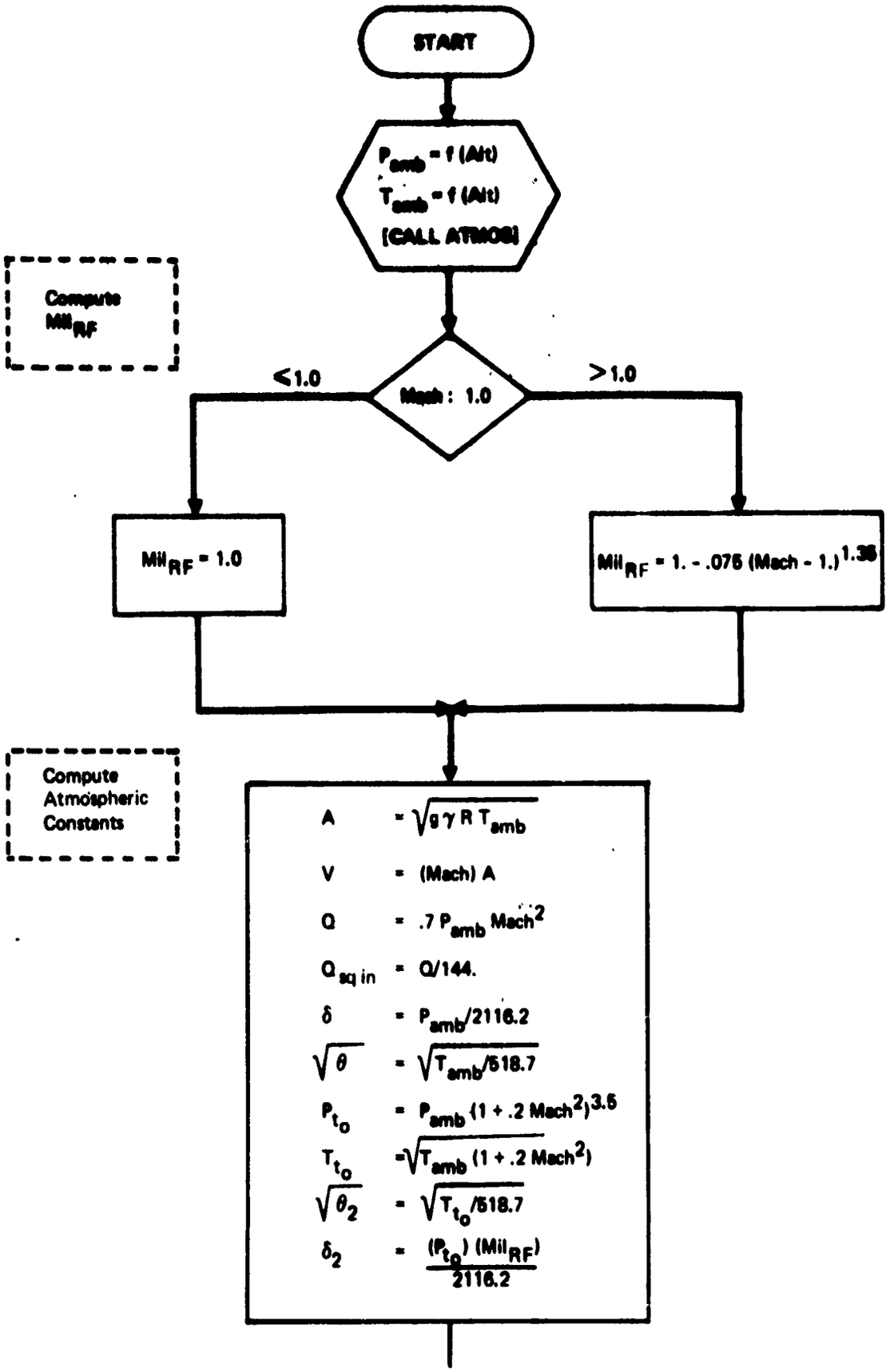


Figure 2: FLOW CHART FOR SUBROUTINE COMPUTE

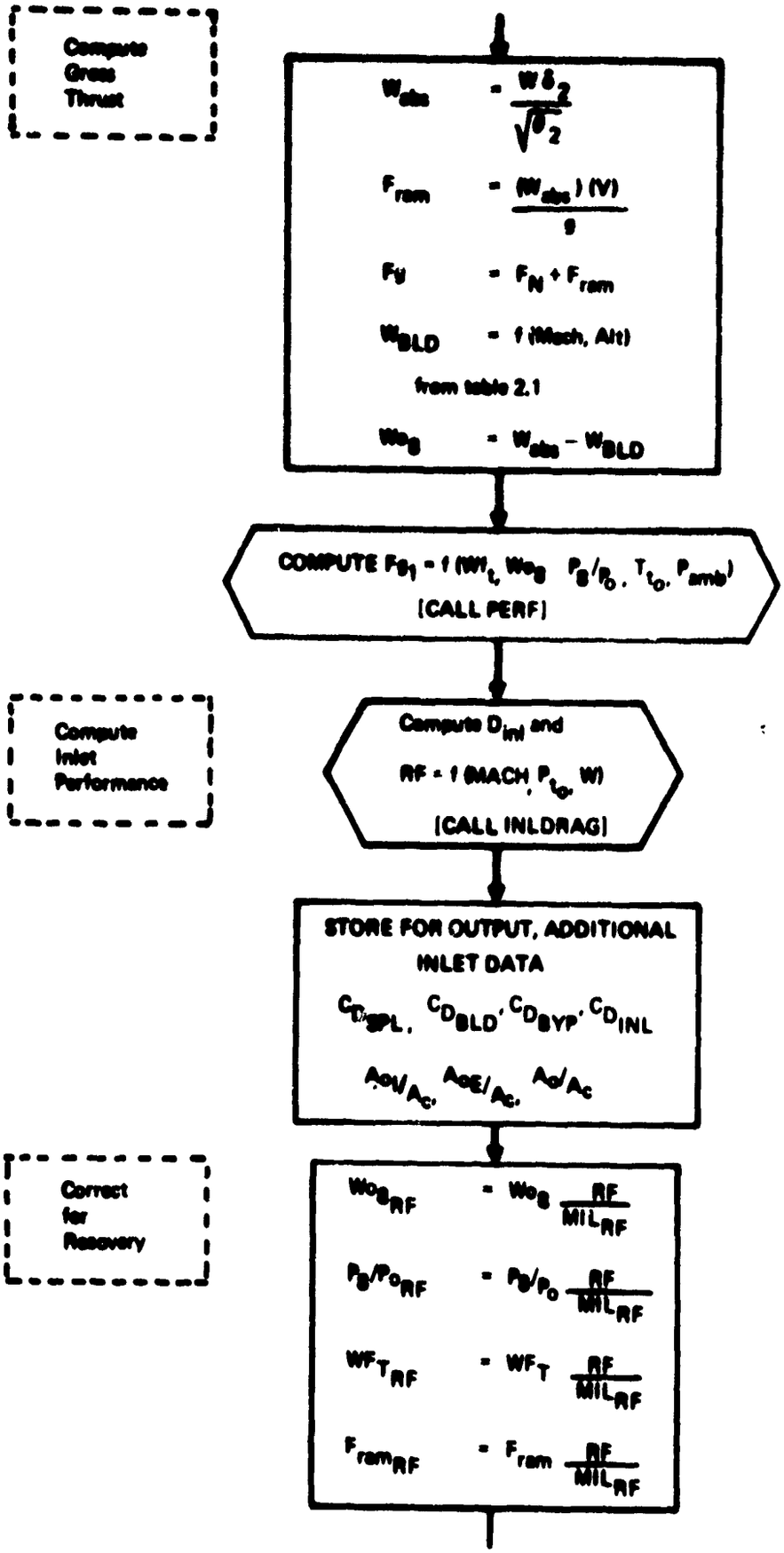


Figure 2: FLOW CHART FOR SUBROUTINE COMPUTE (Cont)

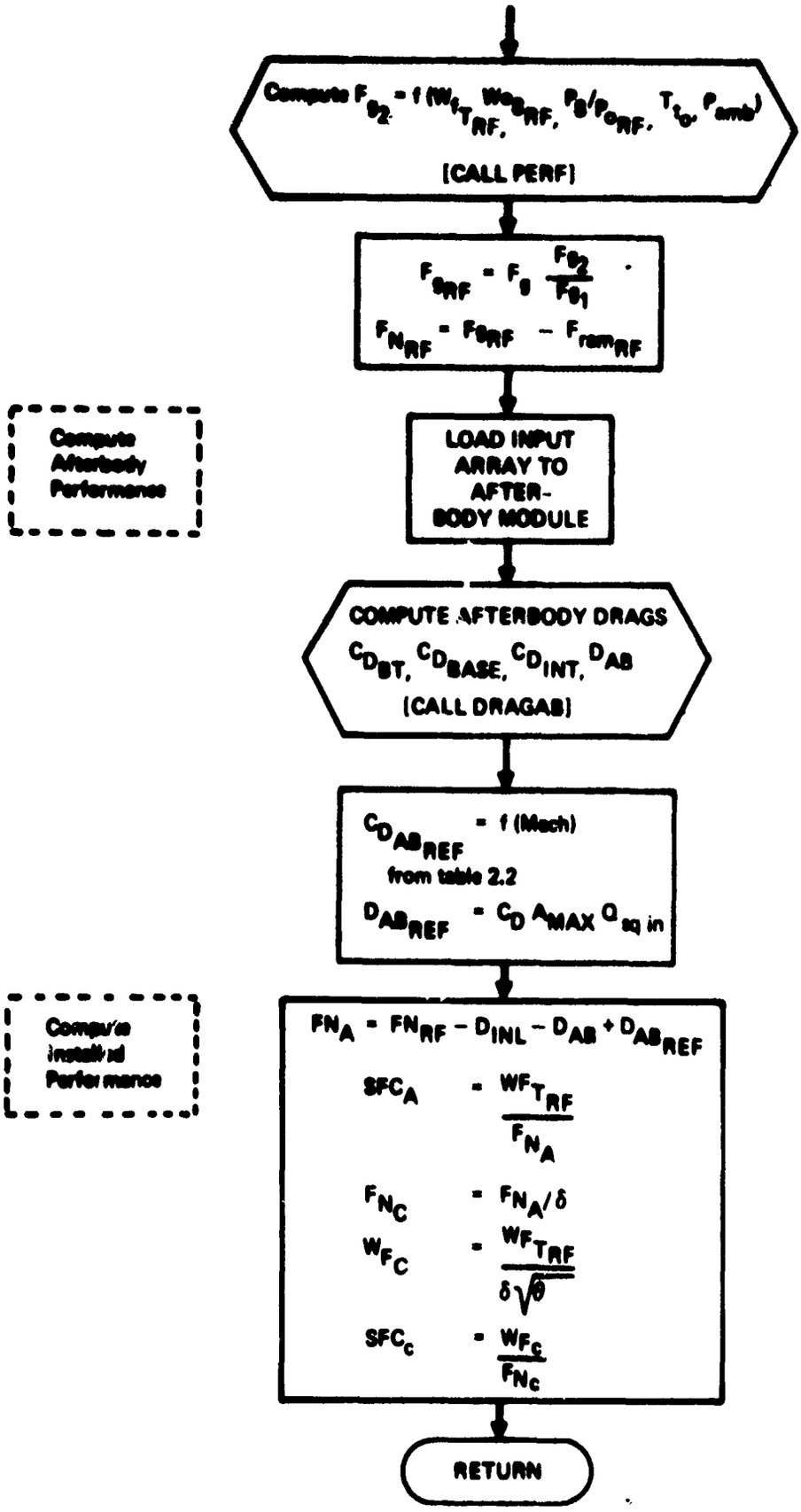


Figure 2: FLOW CHART FOR SUBROUTINE COMPUTE (Concluded)

```
                SUBROUTINE COMPUTE                                      COMPUT10
          C                                                             COMPUT20
000002          COMMON/INS/ MACH,OLDALT,CON
000002          COMMON/G/G
000002          COMMON /OUTS/ CASE, ALT, PS, FNA, WFTRF, SFCOUT, FNRF, FRAM, RF,
                1             MILRF, DINLET, CDSPL, CDBLD, CDBYP, CDINL, DNOZ,
                2             CDBT, CDBASE, CDINT, DNOZRF, P8P0, A9, A8, CFG,
                3             BETA, FN, WFT, SFC, W, WABS, FNCOR, WFCOR, SFCCOR,
               4TAMB, PAMB, TTO, Q, AGEAC,AGIAC,AJAC,STATUS,EXTRA(4)
000002          COMMON/MACHWB/MACHWB(10),ALTWB(13),WB(10,10),NXWB,NYWB
000002          COMMON/DNREF/DNREFM(10),DNREFT(10),NONREF
000002          COMMON /NOZARR/ A(40)
000002          COMMON/OUTINL/OUTINL(10)
000002          COMMON/ENGNO/ENGNO,XONI,XOBASE
000002          DIMENSION O(10)
000002          REAL MILRF
000002          REAL MACH
000002          DATA PIBY4/.78540/
000002          DATA BY144/.00694444/
          C                                                             COMPU140
000002          CALL ATMOS(ALT,TAMB,PAMB)
000005          PSOIN=PAMB*BY144
000007     40   C=SQRT(CON*TAMB)                                        COMPU320
000014          V=MACH*C                                                COMPU330
000015          IF (MACH-1.0) 200,200,205                               COMPU650
000017    200   MILRF=1.0                                               COMPU660
000021          GO TO 210                                               COMPU670
000021    205   MILRF=1.0-.075*(MACH-1.)**1.35                          COMPU680
000030    210   W=.7*PAMB*MACH**2
000033          ASOIN=C*BY144
000035          DELTA=PAMB/2116.22                                      COMPU340
000036          RTHETA=SQRT(TAMB/518.69)                                COMPU350
000043          ZZMACH=1.0+.2*MACH**2
000046          PTO=PAMB*ZZMACH**3.5
000053          TTO=TAMB*ZZMACH
000055          RTHET2=SQRT(TTO/518.69)
000061          DELTA2=PTO*MILRF/2116.22
          C                                                             COMPU360
000063          WABS=W*DELTA2/RTHET2
000066          FRAM=WABS*V/G
000070          FG=FN+FRAM
000072          WOB=WABS-TABU2(MACH,ALT,MACHWB,ALTWB,WB,     NXWB,NYWB,10,10)
000105          CALL PERF(WFT,WOB,P8P0,TTO,PAMB,FG1,AE8)
000114          CALL INLDRAG(MACH,PTO,W,0.0,DINLET,RF)
000120          CDSPL=OUTINL(1)
000122          CDBLD=OUTINL(2)
000123          CDBYP=OUTINL(3)
000125          AGIAC=OUTINL(4)
000126          AGEAC=OUTINL(5)
000130          CDINL=OUTINL(6)
000131          AJAC=OUTINL(7)
000133          RATIO=RF/MILRF
```

```
000135          WO8RF=WO8*RATIO
000136          P8P0RF=P8P0*RATIO
000140          WFTRF=WFT*RATIO
000141          CALL PERF(WFTRF,WO8RF,P8P0RF,TT0,PAMB,FG2,AE8)
000150          FGRF=FG*FG2/FG1
000153          FNRF=FGRF-FRAM*RATIO
          C     SET UP AFTERBODY DRAG MODULE INPUT ARRAY
000156          A(1)=AE8
000157          A(2)=A8
          C     A(3)=0.0
          C     A(4)=0.0
000161          A(5)=P8P0
          C     A(6)=0.0
000162          A(7)=FG2
          C     A(8)=0.0
000164          A(9)=PS
          C     A(10)=GAMMA P
          C     A(11)=0.0
000165          A(12)=MACH
000167          A(13)=PSDIN
000170          A(14)=QSQIN
000172          A(15)=A9
          C     A(16)=0.0
000173          A(17)=CFG
000175          A(18)=DHE
          C     A(19)=0.0
          C     A(20)=PLUG          NOZZLE TYPE DESIGNATION
          C     A(21)=FLAMIX        MIXED OR NON.MIXED FLOW
          C     A(22)=DMAX
          C     A(23)=DMBASE
          C     A(24)=ZLPLUG
          C     A(25)=ZLBTAL
          C     A(26)=SPACE
          C     A(27)=BASE
          C     A(28)=BTTAB    BOATAIL TABLE INDICATOR
          C     A(29)=THETA
000176          A(30)=XDNI
000200          A(31)=XDBASE
000201          CALL DRAGAB(A,D)
000203          DNOZ=D(1)
000205          DBET=D(2)
000206          DBASE=D(3)
000210          DINT=D(4)
000211          DELTA=D(5)
000213          DMAX=A(22)
000214          DNOZRF=TABU1(MACH,DNREFM,DNREFT,  NDNREF)*PI8Y4*QSQIN*DMAX**2
000223          FNA=FNRF-DINLET-DNOZ+DNOZRF
000227          SFCOUT=WFTRF/FNA                                    COMPU810
000230          FNCOR=FNA/DELTA                                     COMPU820
000232          WFCOR=WFTRF/(DELTA*RTHETA)                          COMPU830
000234          SFCOUR=WFCOR/FNCOR                                  COMPU835
          C                                                         COMPU840
000236    7777  RETURN                                              COMPU850
```

```
000237        END                                                    COMPU868
        *DECK OUTGO
```

```
SUBPROGRAM LENGTH
000320

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
46      -  000007    200     -  000017    205     -  000021    210     -  000030
7777    -  000236

EXTERNALS AND TAGS
ATMOS   -  S00100    SQRT    -  S00200    RBAREX.-  S00300    TABU2   -  S00400
PERF    -  S00500    INLDRAG-  S00600    DRAGAB  -  S00700    TABU1   -  S01000
END.    -  S01100

BLOCK NAMES AND LENGTHS
INS     -  000003    G       -  000001    OUTS    -  000055    MACHW8  -  000172
DNREF   -  000025    NOZARR  -  000050    OUTINL  -  000012    ENGNO   -  000003

VARIABLE ASSIGNMENTS
A       -  000000006 AE9     -  000310    ALT     -  000001C03 ALTW8   -  000012C04
ADAC    -  000047C03 ADEAC   -  000045C03 ADIAC   -  000046C03 AB      -  000026C03
A9      -  000025C03 BETA    -  000030C03 BV144   -  000272    C       -  000274
CDBASE  -  000021C03 CDBLD   -  000014C03 CDBT    -  000020C03 CDBVP   -  000015C03
CDINL   -  000016C03 CDINT   -  000022C03 CDSPL   -  000013C03 CFG     -  000027C03
CON     -  000002C01 DELTA   -  000277    DELTA2  -  000304    DME     -  000316
DINLET  -  000012C03 DMAX    -  000317    DNOZ    -  000017C03 DNOZRF  -  000023C03
DNREFM  -  000000C05 DNREFT  -  000012C05 EXTRA   -  000051C03 FG      -  000305
FGRF    -  000315    FG1     -  000307    FG2     -  000314    FN      -  000031C03
FNA     -  000003C03 FNCOR   -  000036C03 FNRF    -  000006C03 FRAM    -  000007C03
G       -  000000002 MACH    -  000300C01 MACHW8  -  000030C04 MILRF   -  000011C03
NONREF  -  000024C05 NXW8    -  000170C04 NYW8    -  000171C04 O       -  000257
OUTINL  -  000000007 PA9B    -  000042C03 PIBV4   -  000271    PS      -  000002C03
PSDIR   -  000273    PTO     -  000302    P8PD    -  000024C03 P8PORF  -  000313
Q       -  000044C03 QSDIN   -  000276    RATIO   -  000311    RF      -  000010C03
RTHETA  -  000300    RTHET2  -  000303    SFCCOR  -  000040C03 SFCOUT  -  000005C03
TAW8    -  000041C03 TTO     -  000043C03 V       -  000275    W       -  000034C03
WABS    -  000035C03 WFCOR   -  000037C03 WFT     -  000032C03 WFTRF   -  000004C03
WO8     -  000306    WOSRF   -  000312    W8      -  000024C04 XDBASE  -  000002C10
XDNI    -  000001C10 ZZMACH  -  000301

START OF CONSTANTS
000240

START OF TEMPORARIES
000254

START OF INDIRECTS
000257

SPACE REQUIRED TO COMPILE
035730
```

SUBJECT: SUBRØUTINE TABIN

PURPØSE: SUBRØUTIN. TABIN reads the Inlet performance data deck, Cards 3.1 to 3.2.12.

METHØD: The input deck is read. Table 2A is enriched as required to evaluate that table. If inlet capture area is not input it is computed.

USAGE: CALL TABIN

All COMMON blocks in this routine are filled by calling this subprogram.

SUBROUTINES CALLED: SIZINL TABL1 TABL2

TABL22 TABU1

```
                SUBROUTINE TABIN
          C     TABIN INPUT TABLES FOR INLET CALCULATIONS
000002          COMMON/TAB1/TAB1(20),NT1
000002          COMMON/TAB2A1/ TAB2AM(10),TAB2AA(10,10),TAB2AP(10,10),NTX2A1,
               A                NTY2A1(10)
000002          COMMON/ENRICH/ ATAB(41,10),PTAB(41,10)
000002          COMMON/XMMIN/XMMIN
000002          COMMON/TAB2B/TAB2B(20),NT2B
000002          COMMON/TAB2C/TAB2C(20),NT2C
000002          COMMON/TAB2D/TAB2D(20),NT2D
000002          COMMON/TAB2E/TAB2E(20),NT2E
000002          COMMON/TAB3/TAB3(120),NTX3,NTY3
000002          COMMON/TAB4/TAB4(120),NTX4,NTY4
000002          COMMON/TAB5/TAB5(120),NTX5,NTY5
000002          COMMON/TAB6A/TAB6A(120),NTX6A,NTY6B
000002          COMMON/TAB7/TAB7(120),NTX7,NTY7
000002          COMMON/TAB6B/TAB6B(20),NT6B
000002          COMMON/AC/AC,ACEAC,ACBYAC,ACBAC,ACIAC
000002          COMMON/IDESGN/XMDES,WCENGD,WCSECD
000002          COMMON/XMACHS/XMACHS
000002          DIMENSION AT(41),PT(41)
000002        1 FORMAT(*0TABLE NUMBER *A3)
000002        2 FORMAT(10F7.0)
000002        3 FORMAT(*1INLET INPUT DECK*5X 8HINPUT AC,5X,5HMACHS,2X,8HMACH DES,
               8*X,6HWC ENG,4X,6HWC SEC /*20X,3F10.3,2F10.1)
000002          READ (5,2) AC,XMACHS,XMDES,WCENGD,WCSECD
000020          WRITE(6,3) AC,XMACHS,XMDES,WCENGD,WCSECD
000036          NO=1H1
000040          WRITE(6,1)NO
000045          CALL TABL1(TAB1,10,NT1)
000050          NO=3H2A
000052          WRITE(6,1)NO
000057          CALL TABL22(TAB2AM,10,10,NTX2A1,NTY2A1)
          C     BUILD ENRICHED TABLE
000063          DO 25 I=1,NTX2A1
000065          N=NTY2A1(I)
000067          IF (N.EQ.1) GO TO 75
000071          AMI=TAB2AA(N,I)
000074          ALO=TAB2AA(1,I)
000076          DA=.05*(AMI-ALO)
000101          DAA=.0155667*(AMI-ALO)
000103          ATAB(1,I)=ALO
000107          PTAB(1,I)=TAB2AP(1,I)
000113          ATAB(41,I)=AMI
000116          PTAB(41,I)=TAB2AP(N,I)
000124          A=ALO
000125          DO 25 K=2,40
000126          IF (K-11) 22,22,23
000130       22 A=A+DA
000132          GO TO 24
000135       23 A=A+DAA
000135       24 ATAB(K,I)=A
```

```
000142          PTAB(K,I)=TABJ1(A,TAB2AA(1,I),TAB2AP(1,I),  N)
000154       25 CONTINUE
000160          WRITE(6,5)
000163        5 FORMAT(*0TABLE 2A AS EXPANDED TO 41 POINTS WITH INTERMEDIATE MACH
                ACURVES INTERPOLATED AS DIAGNOSTIC AID*)
000163          DO 50 I=1,NTX2A1
000165          WRITE(6,4) TAB2AM(I)
000172          WRITE(6,7) (ATAB(K,I),K=1,41)
000206          WRITE(6,7) (PTAB(K,I),K=1,41)
000222          IF (I.EQ.NTX2A1) GO TO 50
000224          TMACH=0.5*(TAB2AM(I+1)+TAB2AM(I))
000227          DO 40 L=1,41
000231          AT(L)=0.5*(ATAB(L,I+1)+ATAB(L,I))
000241       40 PT(L)=0.5*(PTAB(L,I+1)+PTAB(L,I))
000247          WRITE(6,4) TMACH
000254          WRITE(6,7) AT
000262          WRITE(6,7) PT
000270       50 CONTINUE
000273        4 FORMAT(14X,F5.2)
000273        7 FORMAT(9X,11F9.4/(18X,10F9.4))
000273       75 CONTINUE
000273          XMMIN=TAB2AM(1)
000275          WRITE(6,5) XMMIN
000302        6 FORMAT(*0MINIMUM MACH NUMBER FOR INLET DRAG CALCULATIONS*F7.3)
000302          NO=2H2B
000304          WRITE(6,1)NO
000311          CALL TABL1(TAB2B,10,NT2B)
000314          NO=2H2C
000316          WRITE(6,1)NO
000323          CALL TABL1(TAB2C,10,NT2C)
000326          NO=2H2D
000330          WRITE(6,1)NO
000335          CALL TABL1(TAB2D,10,NT2D)
000340          NO=2H2E
000342          WRITE(6,1)NO
000347          CALL TABL1(TAB2E,10,NT2E)
000352          NO=1H3
000354          WRITE(6,1)NO
000361          CALL TABL2(TAB3,10,10,NTX3,NTY3)
000365          NO=1H4
000367          WRITE(6,1)NO
000374          CALL TABL2(TAB4,10,10,NTX4,NTY4)
000400          NO=1H5
000402          WRITE(6,1)NO
000407          CALL TABL2(TAB5,10,10,NTX5,NTY5)
000413          NO=2H6A
000415          WRITE(6,1)NO
000422          CALL TABL2(TAB6A,10,10,NTX6A,NTY6A)
000426          NO=2H6B
000430          WRITE(6,1)NO
000435          CALL TABL1(TAB6B,10,NT6B)
000440          NO=1H7
000442          WRITE(6,1)NO
```

```
CC0447        CALL TABL2(TAB7,10,10,NTX7,NTY7)
CC0453        IF (AC.LE.0.0) CALL SIZINL
CC0456        RETURN
CC0457        END
```

```
SUBPROGRAM LENGTH
000717

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       -  000461    2       -  000465    3       -  000467    4       -  000530
5       -  000512    6       -  000537    7       -  000533    22      -  000130
23      -  000133    24      -  000135    50      -  000270    75      -  000273

EXTERNALS AND TAGS
INPUTC.-  S00100    OUTPTC.-  S00200    TABL1   -  S00300    TABL22  -  S00400
TABUI   -  S00500    TABL2   -  S00600    SIZINL  -  S00700    END.    -  S01000

BLOCK NAMES AND LENGTHS
TAB1    -  000325    TAB2A1  -  000335    ENRICH  -  001464    XMMIN   -  000001
TAB2B   -  000025    TAB2C   -  000025    TAB2D   -  000025    TAB2E   -  000025
TAB3    -  000172    TAB4    -  000172    TAB5    -  000172    TAB6A   -  000172
TAB7    -  000172    TAB6B   -  000025    AC      -  000005    IDESGN  -  000003
XMACHS  -  000001

VARIABLE ASSIGNMENTS
A       -  000713    AC      -  000000C17 AMI     -  000707    ALO     -  000710
AT      -  000562    ATAB    -  000000C03 DA      -  000711    DAA     -  000712
I       -  000705    K       -  000714    L       -  000716    N       -  000706
NO      -  000704    NTX2A1  -  000322C02 NTX3    -  000170C11 NTX4    -  000170C12
NTX5    -  000170C13 NTX6A   -  000170C14 NTX7    -  000170C15 NTY2A1  -  000323C02
NTY3    -  000171C11 NTY4    -  000171C12 NTY5    -  000171C13 NTY6B   -  000171C14
NTY7    -  000171C15 NT1     -  000024C01 NT2B    -  000024C05 NT2C    -  000024C06
NT2D    -  000024C07 NT2E    -  000024C10 NT3     -  000024C16 PT      -  000633
PTAB    -  000632C03 TAB1    -  000000C01 TAB2AA  -  000012C02 TAB2AM  -  000000C02
TAB2AP  -  000156C02 TAB2B   -  000000C05 TAB2C   -  000000C06 TAB2D   -  000000C07
TAB2E   -  000000C10 TAB3    -  000000C11 TAB4    -  000000C12 TAB5    -  000000C13
TAB6A   -  000000C14 TAB6B   -  000000C16 TAB7    -  000000C15 TMACH   -  000715
WCENGD  -  000001C20 WCSECD  -  000002C26 XMACHS  -  000000C21 XMDES   -  000000C20
XMMIN   -  000000C04

START OF CONSTANTS
000660

START OF TEMPORARIES
000667

START OF INDIRECTS
000702

SPACE REQUIRED TO COMPILE
037700
```

SUBJECT: SUBRØUTINE INLDRAG Controls the Calculation of Inlet Performance.

METHØD: After initialization M_0 is interpolated and compared with the minimum Mach number for which inlet drag will be computed and with the start Mach number to determine which inlet mode to use. Low speed mode is computed in INLDRAG: Subroutine AIRBYP computes external-compression mode performance, and subroutine AIRSPL computes mixed-compression mode. In the two latter cases inlet drag coefficients are interpolated and inlet drag determined. The outlet array is loaded and subroutine is complete.

USAGE: CALL INLDRAG (XM, P, WE, WS, DINLET, RF)

XM - Free-stream Mach number

P - Total free-stream pressure LB/FT^2

WE - Engine corrected airflow, primary LB/Sec

WS - Engine corrected airflow, secondary LB/Sec

Output DINLET - Inlet Drag LB

Output RF - Inlet recovery factor

Input from CØMMØN

XMMIN - Minimum Mach for inlet drag calculation

AC - Capture area - ft^2

Inlet Tables 1, 2B, 3, 4 and 5

Output in CØMMØN

ØUTINL (10)

SUBRØUTINES CALLED: AIRBYP AIRSPL AREAF

TABU1 TABU2

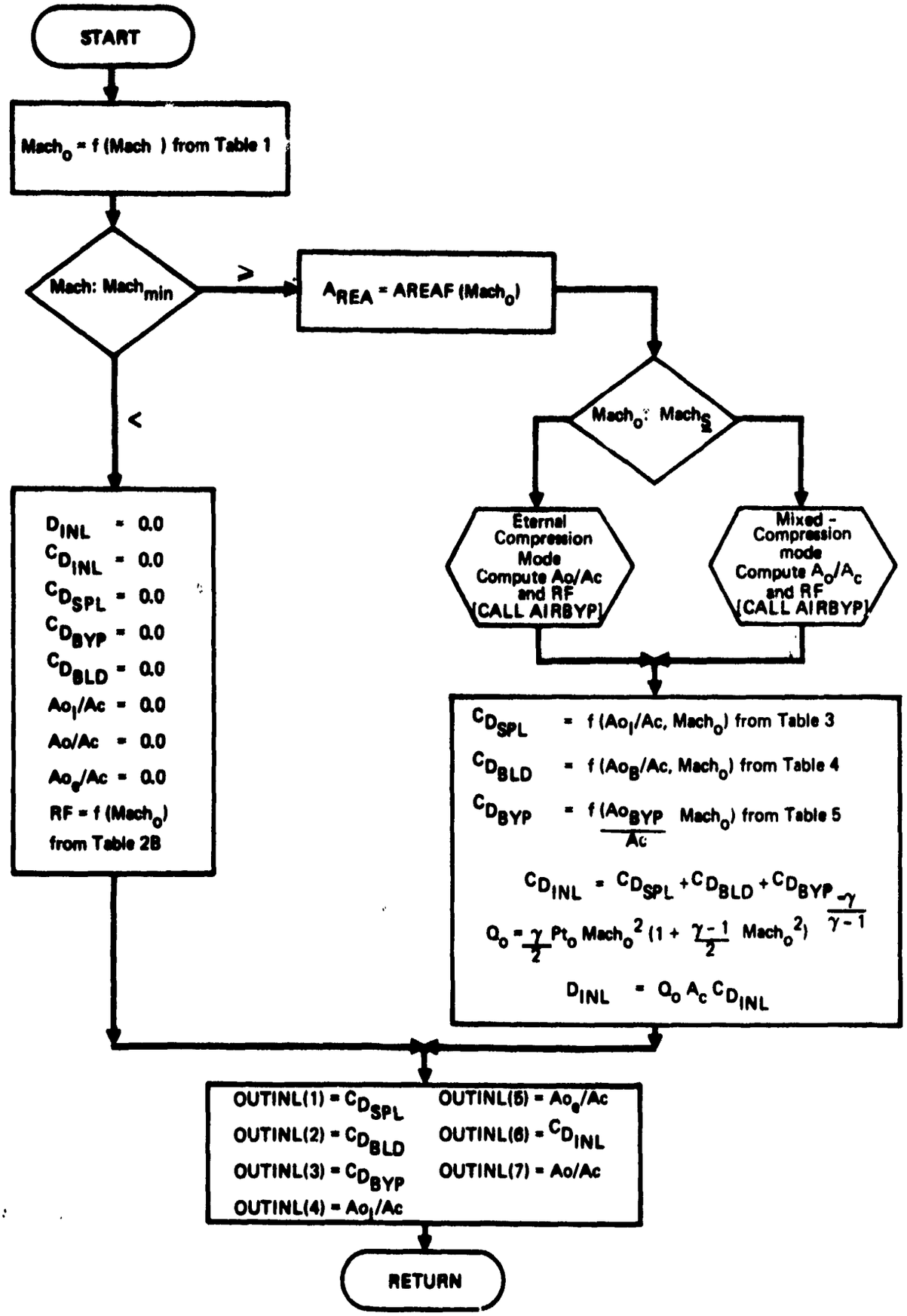


Figure 3: FLOW CHART FOR SUBROUTINE INLDRAG

```
                SUBROUTINE INLDRAG(XM,P,WE,WS,DINLET,RF)
000011          COMMON/OUTINL/OUTINL(10)
000011          COMMON/XMACHS/XMACHS
000011          COMMON/XMMIN/XMMIN
000011          COMMON/AREAVL/AREA
000011          COMMON/PT2PTO/PT2PTO
000011          COMMON/TAB1/TAB1X(10),TAB1Y(10),NT1
000011          COMMON/TAB2B/TAB2BX(10),TAB2BY(10),NT2B
000011          COMMON/TAB3/AIACT(10),XM3(10),CDT3(10,10),NTX3,NTY3
000011          COMMON/TAB4/ABACT(10),XM4(10),CDT4(10,10),NTX4,NTY4
000011          COMMON/TAB5/AVACT(10),XM5(10),CDT5(10,10),NTX5,NTY5
000011          COMMON/AC/AC,ACEAC,AVAC,ABAC,AIAC
000011          COMMON/WC/WCENG,WCSEC
000011          COMMON/XMACH/XMACH
000011          COMMON/GAMMA/GAMMA
000011          COMMON/PT/PT
000011          COMMON/XMZERO/XMZERO
000011          COMMON/AJAC/AJAC
000011          XMACH=XM
000012          PT=P
000013          WCENG=WE
000014          WCSEC=WS
000015          CDSPL=0.0
000016          CDBLD=0.0
000017          CDBVP=0.0
000020          AIAC=0.0
000021          AJEAC=0.0
000022          AJAC=0.0
000023          XMZERO=TABU1(XMACH,TAB1X,TAB1Y,  NT1)
000027          IF (XMZERO.LT.XMMIN) GO TO 300
000035          AREA=AREAF(XMZERO)
000037          IF (XMZERO.LT.XMACHS) GO TO 100
000045          CALL AIRSPL
000046          GO TO 200
000052    100   CALL AIRBVP
000053    200   CDSPL=TABU2(AIAC,XMZERO,AIACT,XM3,CDT3,    NTX3,NTY3,10,10)
000065          CDBLD=TABU2(ABAC,XMZERO,ABACT,XM4,CDT4,    NTX4,NTY4,10,10)
000076          CDBVP=TABU2(AVAC,XMZERO,AVACT,XM5,CDT5,    NTX5,NTY5,10,10)
000107          UZERO=(GAMMA*.5)*PT*(XMZERO**2)*
               1(1.+((GAMMA-1.)*.5)*(XMZERO**2))**(-GAMMA/(GAMMA-1.))
000126          DINLET=UZERO*AC*(CDSPL+CDBLD+CDBVP)
000136          GO TO 400
000136    300   DINLET=0.0
000137          PT2PTO=TABU1(XMZERO,TAB2BX,TAB2BY,  NT2B)
000143    400   RF=PT2PTO
000150          OUTINL(1)=CDSPL
000151          OUTINL(2)=CDBLD
000153          OUTINL(3)=CDBVP
000154          OUTINL(4)=AIAC
000156          OUTINL(5)=AJEAC
000157          OUTINL(6)=CDSPL+CDBLD+CDBVP
000161          OUTINL(7)=AJAC
```

```
00.162        RETURN
00.163        END
```

```
SUBPROGRAM LENGTH
000201

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100     -  000052    200     -  000053    300     -  000136    400     -  000143

EXTERNALS AND TABS
TAB01   -  000100    AREAF   -  000200    AIRSPL  -  000300    AIRBYP  -  000400
TAB02   -  000500    RBAREX. -  000600    END.    -  000700

BLOCK NAMES AND LENGTHS
OUTINL  -  000012    XMACHS  -  000001    XMMIN   -  000001    AREAVL  -  000001
PT2PTO  -  000001    TAB1    -  000025    TAB2B   -  000025    TAB3    -  000172
TAB4    -  000172    TAB5    -  000172    AC      -  000005    WC      -  000002
XMACH   -  000001    GAMMA   -  000001    PT      -  000001    XMZERO  -  000001
ACAC    -  000001

VARIABLE ASSIGNMENTS
ABAC    -  000003C13 ABACT   -  000000C11 AC      -  000000C13 AIAC    -  000004C13
AIACT   -  000005C13 AREA    -  000000C04 AYAC    -  000002C13 AYACT   -  000003C12
ACAC    -  000000C21 AREAC   -  000001C13 COBLD   -  000176    COBYP   -  000177
COSPL   -  000175    COT3    -  000024C10 COT4    -  000024C11 COT5    -  000024C12
GAMMA   -  000000C16 NTX3    -  000170C10 NTX4    -  000170C11 NTX5    -  000170C12
NTY3    -  000171C10 NTY4    -  000171C11 NTY5    -  000171C12 NT1     -  000024C06
NT2B    -  000024C07 OUTIN.  -  000000C01 PT      -  000000C17 PT2PTO  -  000000C05
QZERO   -  000200    TAB1X   -  000000C06 TAB1Y   -  000012C06 TAB2BX  -  000000C07
TAB2BY  -  000012C07 WCENG   -  000000C14 WCSEC   -  000001C14 XMACH   -  000000C15
XMACHS  -  000000C02 XMMIN   -  000000C03 XMZERO  -  000000C20 XM3     -  000012C10
XM4     -  000012C11 XM5     -  000012C12

START OF CONSTANTS
000164

START OF TEMPORARIES
000167

START OF INDIRECTS
000175

SPACE REQUIRED TO COMPILE
035310
```

SUBJECT: SUBRØUTINE SIZINL

PURPØSE: SUBRØUTINE SIZINL computes inlet capture area.

USAGE: CALL SIZINL

Inputs in CØMMØN/IDESIGN/ design Mach and airflow.

Tables 1, 2B and 2C.

Output: AC ft^2

SUBJECT: FUNCTIØN AREAF

PURPØSE:

$$\text{AREAF} = \frac{.013858 \left(1 + \frac{\gamma - 1}{2} \text{Mach}^2 \right)^{\frac{\gamma + 1}{2(\gamma - 1)}}}{\text{Mach}\sqrt{\gamma}}$$

USAGE: A = AREAF (XMACH)

XMACH local Mach number

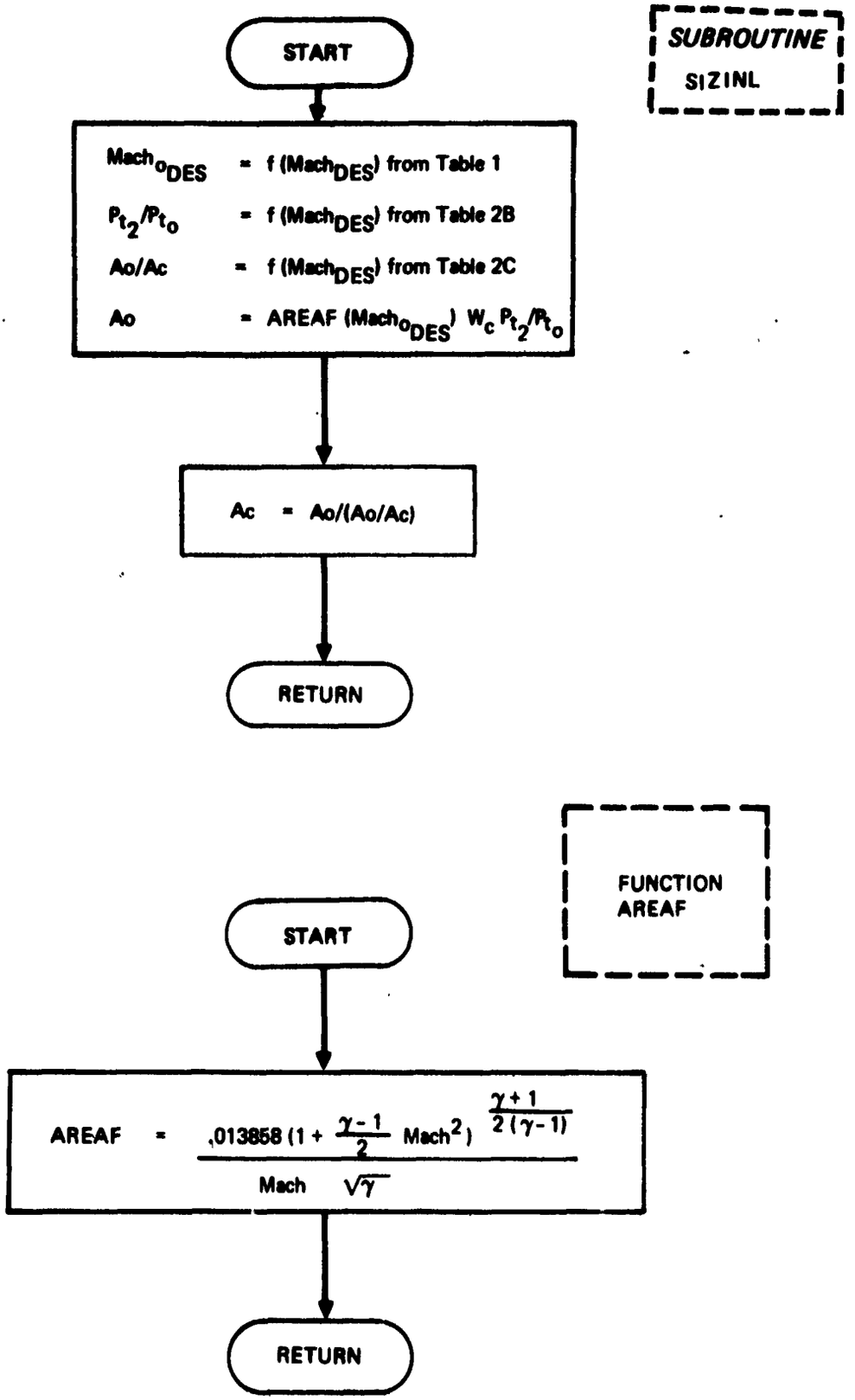


Figure 4: FLOW CHART FOR SUBROUTINE SIZINL AND FUNCTION AREAF

```
              SUBROUTINE SIZINL
000002        COMMON/IDESGN/XMDES,WCENGD,WCSECD
000002        COMMON/TAB1/TAB1X(10),TAB1Y(10),NT1
000002        COMMON/TAB2B/TAB2BX(10),TAB2BY(10),NT2B
000002        COMMON/TAB2C/TAB2CX(10),TAB2CY(10),NT2C
000002        COMMON/AC/AC,ACEAC,ADBYAC,ADBAC,ADIAC
000002        XMDES=TABU1(XMDES,TAB1X,TAB1Y,  NT1)
000015        PTP2RC=TABU1(XMDES,TAB2BX,TAB2BY,  NT2B)
000011        ADACRC=TABU1(XMDES,TAB2CX,TAB2CY,  NT2C)
000015        AD=AREAF(XMDES)*(WCENGD+WCSECD)*PTP2RC
000023        AC=AD/ADACRC
000026        WRITE(6,1) AC,XMDES,WCENGD,WCSECD,XMDES,PTP2RC,ADACRC,AD
000031      1 FORMAT(*0AC =* F6.3,10X,7E12.4)
000031        RETURN
000032        END
```

```
SUBPROGRAM LENGTH
000065

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       -  000055

EXTERNALS AND TABS
TABU1   -  S00100     AREAF   -  S00200     OUTPTC.-  S00300     END.    -  S00400

BLOCK NAMES AND LENGTHS
IDESCN  -  000003     TAB1    -  000025     TAB2B   -  000025     TAB2C   -  000025
AC      -  000005

VARIABLE ASSIGNMENTS
AC      -  000000005  AO      -  000064     AOACRC  -  000063     NT1     -  000024C02
NT2B    -  000024C03  NT2C    -  000024C04  PTP2RC  -  000062     TAB1X   -  000000C02
TAB1Y   -  000012C02  TAB2BX  -  000000C03  TAB2BY  -  000012C03  TAB2CX  -  000000C04
TAB2CY  -  000012C04  WCSWGD  -  000001C01  WCSECD  -  000002C01  XMDES   -  000000C01
XMCDES  -  000061

START OF CONSTANTS
000053

START OF TEMPORARIES
000056

START OF INDIRECTS
000001

SPACE REQUIRED TO COMPILE
035400
```

SUBJECT: SUBRØUTINE AIRBYP

PURPØSE: SUBRØUTINE AIRBYP Computes recovery and mass flow ratios for external compression mode inlets.

METHØD: AIRBYP matches the recovery and mass flow from Table 2A to that obtained from function DEMAND. Table 2A was enriched to 41 points per array when input by SUBRØUTINE TABIN. AIRBYP first interpolates in this table to develop a curve of RF = f (A_O/A_C) at the input Mach number.

The 41st point of this table represents the choked flow point, in Maximum (A_O/A_C). DEMAND is called to compute (A_O/A_C) for the RF at the choked point. This A_O/A_C is compared to the A_O/A_C of the choked point. If the A_O/A_C from DEMAND is greater than the maximum A_O/A_C then the inlet is choked and RF must be found such that DEMAND(RF) = $A_O/A_{C_{MAX}}$. A Newton-Raphson iteration is used to solve this case. Otherwise the inlet is not choked and a solution for A_O/A_C exists less than the maximum. A Regula Falsi iteration is used to match the recovery and mass flow.

In either case buzz and distortion limits are checked and the other mass flow ratios computed.

USAGE: CALL AIRBYP

Inputs in COMMON include Mach number, air flow, capture area and inlet tables. Other COMMON blocks are output.

SUBPRØGRAMS:

DEMAND	FACINT	IFIND
TABU1	TABU2	WARNING

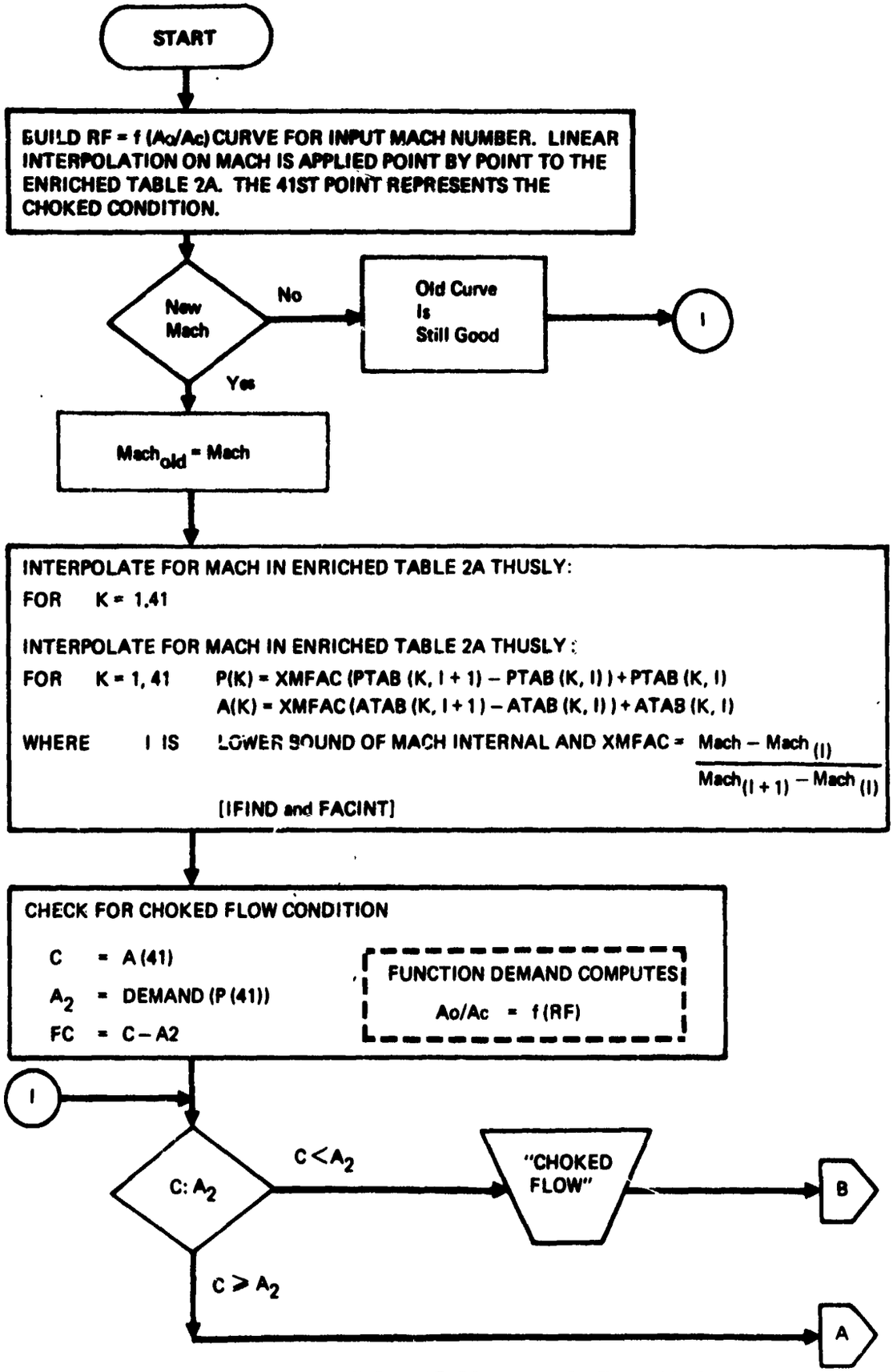
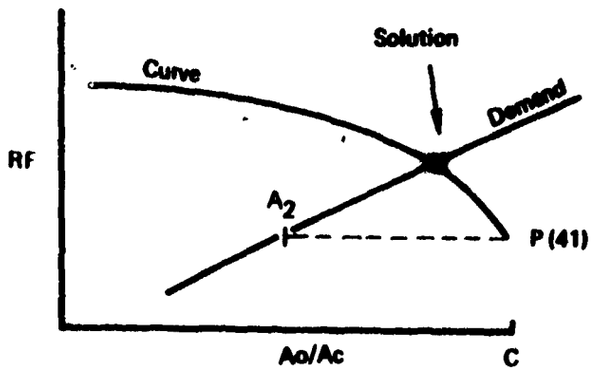


Figure 5: FLOW CHART FOR SUBROUTINE AIRBYP



REGULA FALSI ITERATION TO FIND INTERSECTION

NON-CHOKED FLOW

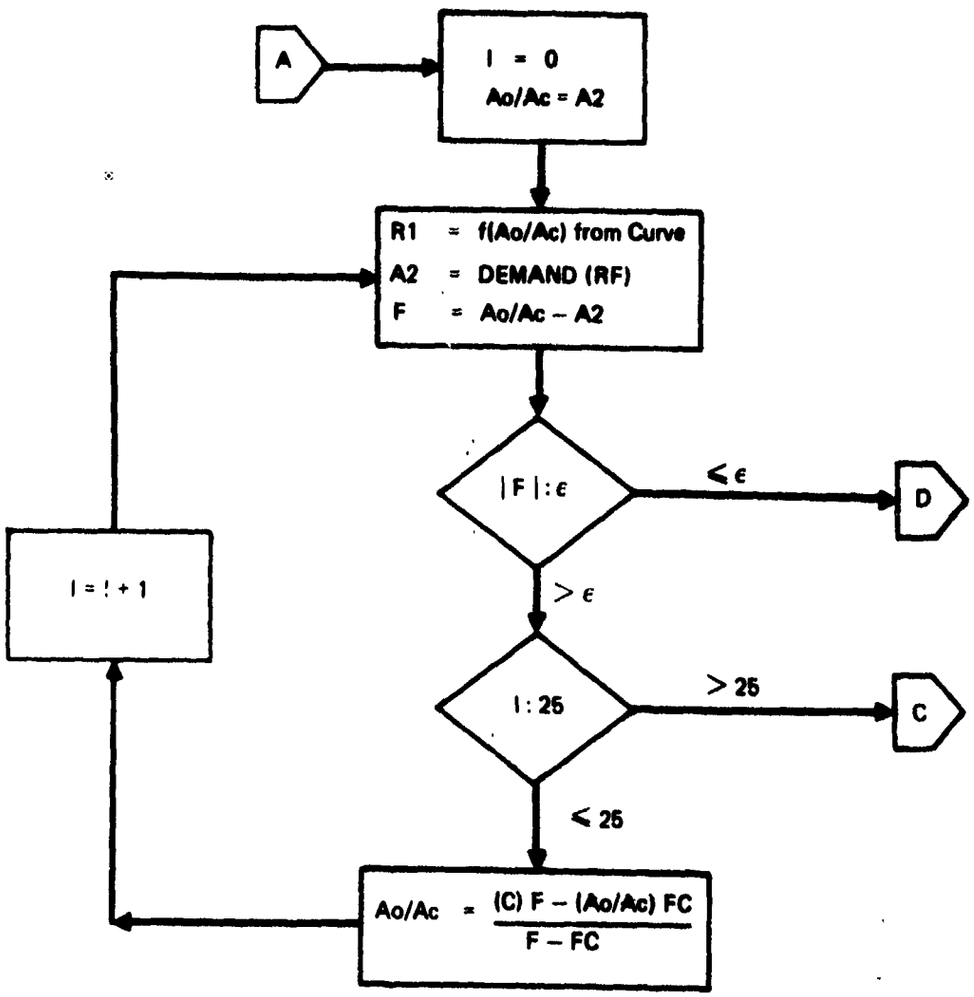
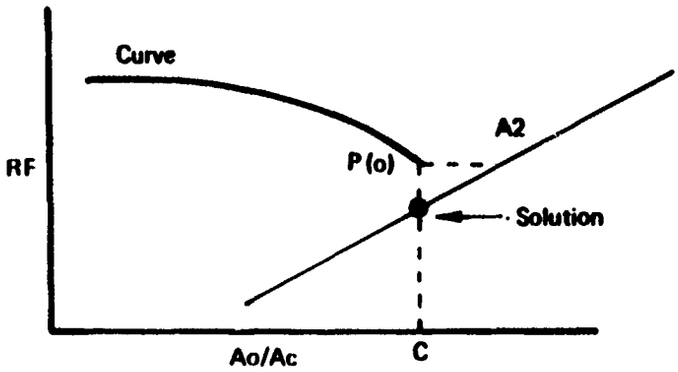


Figure 5: FLOW CHARTS FOR SUBROUTINE AIRBYP (Cont)



NEWTON–RAPHSON ITERATION TO FIND INTERSECTION

CHOKED FLOW

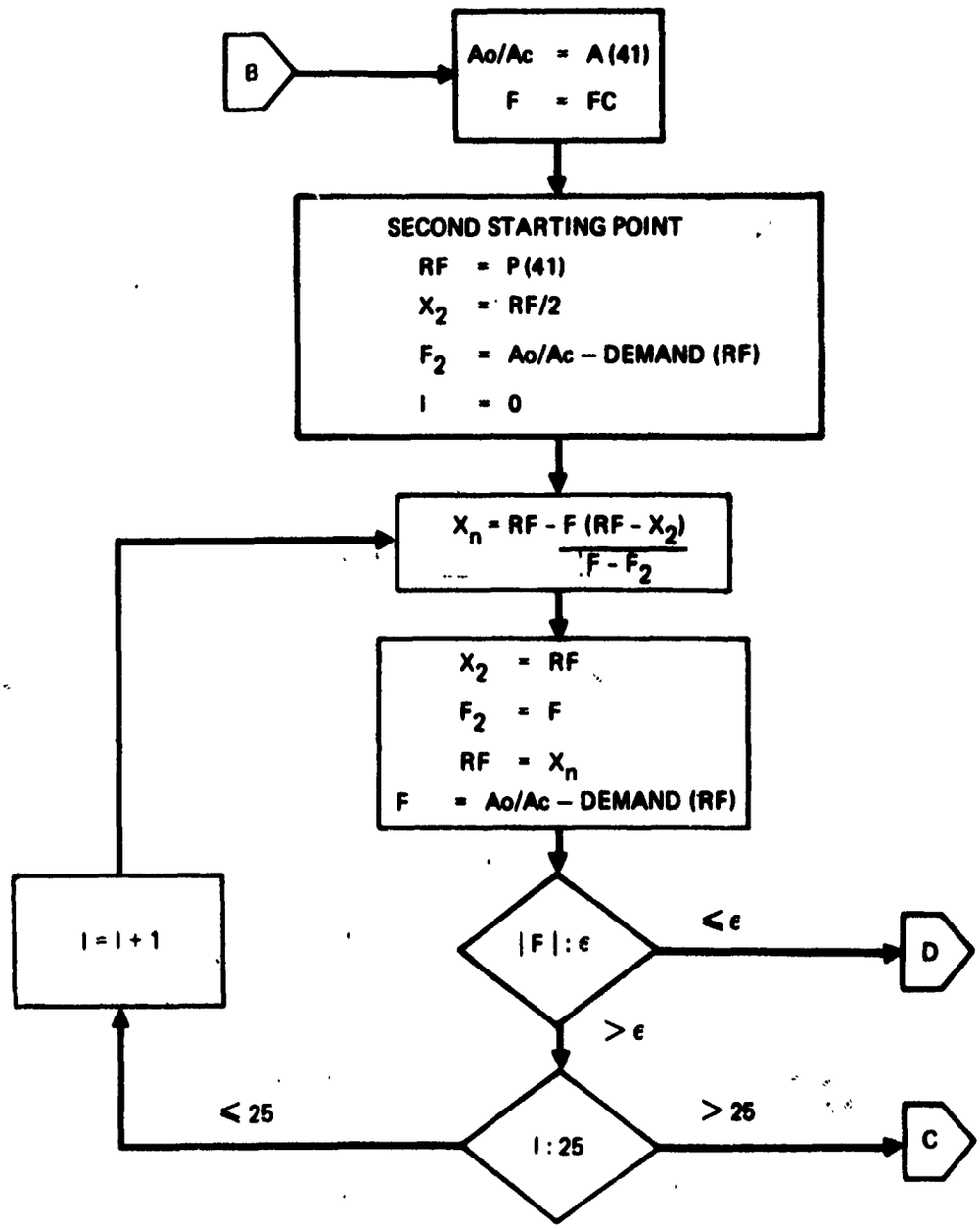


Figure 5: FLOW CHART FOR SUBROUTINE AIRBYP (Cont)

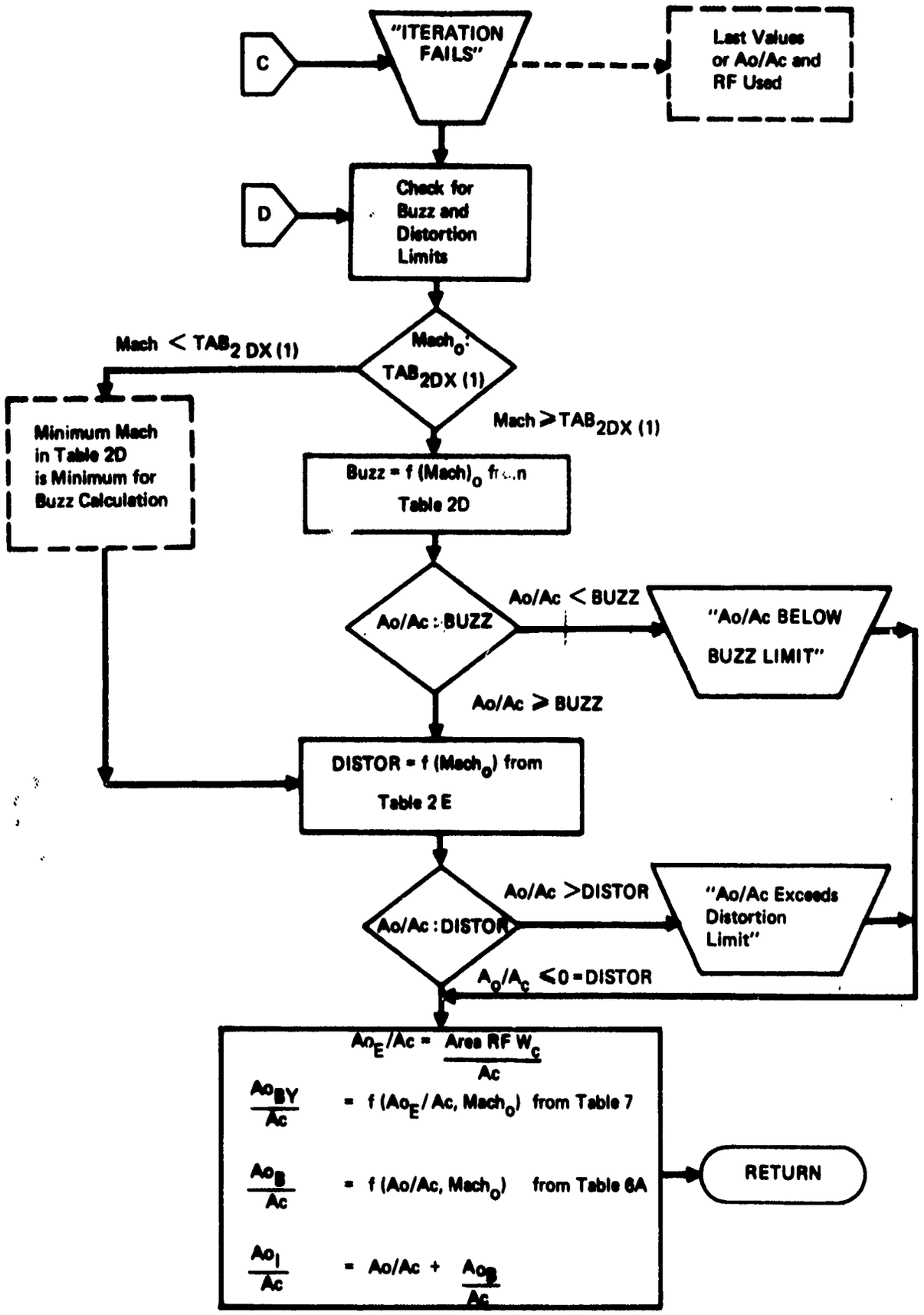


Figure 5: FLOW CHART FOR SUBROUTINE AIRBYP (Concluded)

```
                SUBROUTINE AIRBYP
000002          COMMON/PT2PT0/PT2PT0
000002          COMMON/AREAWL/AREA
000002          COMMON/WC/WCENG,WCSEC
000002          COMMON/GAMMA/GAMMA
000002          COMMON/XMZERO/XMZERO
000002          COMMON/TAB2A1/ XM2A1(10),ARRAY1(200),NTX2A1,NTY2A1(10)
000002          COMMON/ENRICH/ ATAB(41,10),PTAB(41,10)
000002          COMMON/AC/AC,ACEAC,ACBYAC,ACBAC,ACIAC
000002          COMMON/TAB5A/TAB6X(10),TAB6Y(10),TAB6Z(10,10),NTX6,NTY6
000002          COMMON/TAB7 /TAB7X(10),TAB7Y(10),TAB7Z(10,10),NTX7,NTY7
000002          COMMON/TAB2D/TAB2DX(10),TAB2DY(10),NT2D
000002          COMMON/TAB2E/TAB2EX(10),TAB2EY(10),NT2E
000002          COMMON/XMACH/XMACH
000002          COMMON/PT/PT
000002          COMMON/ADAC/ADAC
000002          DIMENSION A(41),P(41)
000002          INTEGER SEARCH
000002          DATA TOL/.0005/
000002          DATA OLDMZ/-999./
          C     BUILD PT2/PT0 = F(AC/AC) AT XMZERO
000002          NA=41
000003          IFLAG=IFIND(XMZERO,XM2A1,NTX2A1,I)
000007          XMFAC=FACINT(XMZERO,XM2A1(I),IFLAG)
000013          N1=NTY2A1(I)
000015          IF (N1.EQ.1) GO TO 2500
000017          IF (XMZERO.EQ.OLDMZ) GO TO 530
000021          OLDMZ=XMZERO
000021          IF (XMFAC)100,300,100
000022      100 N2=NTY2A1(I+1)
000024          DO 200 K=1,41
000026          P(K)=XMFAC*(PTAB(K,I+1)-PTAB(K,I))+PTAB(K,I)
000036          A(K)=XMFAC*(ATAB(K,I+1)-ATAB(K,I))+ATAB(K,I)
000044      200 CONTINUE
000045          GO TO 500
000046      300 DO 400 K=1,41
000050          P(K)=PTAB(K,I)
000054          A(K)=ATAB(K,I)
000060      400 CONTINUE
          C     SEARCH FOR INTERSECTION
          C     FIND SOLUTION INTERVAL
000061      500 C=A(NA)
000063          A2=DEMAND(P(NA))
000066          FC=C-A2
000070          IF (C.LT.A2) GO TO 2000
000071          ICOUNT=C
000072          ADAC=A2
          C     REGULA FALSI METHOD
000073      675 PT2PTC=TABU1(ADAC,A,P  ,NA)
000077          A2=DEMAND(PT2PT0)
000101          F=ADAC-A2
000103          IF (ABS(F).LE.TOL) GO TO 3000
```

```
000106          IF (ICOUNT.GT.25) GO TO 2900
000111          ICOUNT=ICOUNT+1
000112          A0AC=(C*F-A0AC*FC)/(F-FC)
000117          GO TO 675
          C     SOLUTION FOR PT2/PTC SUCH THAT DEMAND=MAX A0/AC
000120     2000 A0AC=A(NA)
000122          CALL WARNING
000123          WRITE(6,4)
000127        4 FORMAT(* CHOKED FLOW CONDITIONS*)
000127          F=FC
000131          PT2PTC=P(NA)
000133          X2=.5*PT2PTC
000134          F2=A0AC-DEMAND(X2)
000140          ICOUNT=0
000141     2100 XNEXT=PT2PTC-F*(PT2PTC-X2)/(F-F2)
000147          X2=PT2PTC
000150          F2=F
000150          PT2PTC=XNEXT
000152          F=A0AC-DEMAND(PT2PTC)
000155          IF (ABS(F).LT.TOL) GO TO 3000
000160          IF (ICOUNT.GT.25) GO TO 2900
000164          ICOUNT=ICOUNT+1
000165          GO TO 2100
000165     2500 PT2PTC=XNEAC*(PTAB(1,I+1)-PTAB(1,I))+PTAB(1,I)
000175          GO TO 3000
000175     2900 CALL WARNING
000176          XDEM=DEMAND(PT2PTC)
000201          WRITE(6,3) PT2PTC,A0AC,XDEM
000212        3 FORMAT(* ITERATION TO MATCH PT2/PTC AND A0/AC DID NOT CONVERGE
               APT2/PTC*F8.4, 5X*A0/AC*F8.4,5X,*DEMAND*F8.4)
000212     3000 IF (XMZERO.LT.TAB2DX(1)) GO TO 3010
000215          BUZZLM=TAB01(XMZERO,TAB2DX,TAB2DY,  NT2D )
000221          IF(A0AC.GE.BUZZLM) GO TO 3010
000223          CALL WARNING
000224          WRITE(6,1) A0AC,BUZZLM
000234        1 FORMAT(* A0/AC BELOW BUZZ LIMIT.  A0/AC*F8.4,5X*BUZZ LIMIT*F8.4)
000234          GO TO 3020
000235     3010 DISTOR=TAB01(XMZERO,TAB2EX,TAB2EY,  NT2E)
000241          IF (A0AC.LE.DISTOR) GO TO 3020
000243          CALL WARNING
000244          WRITE(6,2) A0AC,DISTOR
000254        2 FORMAT(* A0/AC EXCEEDS DISTORTION LIMIT.  A0/AC*F8.4,5X,*DISTORTIO
               1N LIMIT* F8.4)
000254     3020 A0EAC=AREA          *PT2PTC*WCENG/AC
000260          A0BYAC=TAB02(ACEAC,XMZERO,TAB7X,TAB7Y,TAB7Z,    NTX7,NTY7,10,10)
000271          A0BAC=TAB02(A0AC,XMZERO,TAB6X,TAB6Y,TAB6Z,    NTX6,NTY6,10,10)
000302          A0IAC=A0AC+A0BAC
000304          RETURN
000304          END
```

SUBPROGRAM LENGTH
000521

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

1	-	000338	2	-	000337	3	-	000314	4	-	000307
100	-	000022	300	-	000046	500	-	000051	675	-	000073
2000	-	000120	2100	-	000141	2500	-	000155	2900	-	000175
3000	-	000212	3010	-	000235	3020	-	000254			

EXTERNALS AND TAGS

IFIND	-	S00100	FACINT	-	S00200	DEMAND	-	S00300	TABU1	-	S00400
MARKING	-	S00500	OUTPTC.	-	S00600	TABU2	-	S00700	END.	-	S01000

BLOCK NAMES AND LENGTHS

PT2PTG	-	000001	AREAVL	-	000001	WC	-	000002	GAMMA	-	000001
XMZERO	-	000001	TAB2A1	-	000335	ENRICH	-	001464	AC	-	000005
TAB6A	-	000172	TAB7	-	000172	TAB2C	-	000025	TAB2E	-	000025
XMACH	-	000001	PT	-	000001	AGAC	-	000001			

VARIABLE ASSIGNMENTS

A	-	000352	AC	-	000000C10	AREA	-	000000C12	ARRAY2	-	000012C06
ATAB	-	000000C07	AGAC	-	000000C17	ADBAC	-	000003C10	ADBVAC	-	000002C10
ACEAC	-	000001C10	ACIAC	-	000004C10	A2	-	000537	BUZZLM	-	000517
C	-	000500	DISTOR	-	000520	F	-	000512	FC	-	000510
F2	-	000514	I	-	000501	ICOUNT	-	000511	IFLAG	-	000500
K	-	000535	NA	-	000477	NTX2A1	-	000322C06	NTX6	-	000170C11
NTX7	-	000170C12	NTY2A1	-	000323C06	NTY6	-	000171C11	NTY7	-	000171C12
NTZ0	-	000324C13	NTZE	-	000024C14	N1	-	000503	N2	-	000504
OLDMJ	-	000470	P	-	000423	PTAB	-	000632C07	PT2PTG	-	000000C01
SEARCH	-	000474	TAB2DX	-	000000C13	TAB2DY	-	000012C13	TAB2EX	-	000001C14
TAB2EY	-	000012C14	TAB5X	-	000000C11	TAB6Y	-	000012C11	TAB6Z	-	000024C11
TAB7X	-	000000C12	TAB7Y	-	000012C12	TAB7Z	-	000024C12	TOL	-	000475
WCENG	-	000000C03	XDEM	-	000516	XMFAC	-	000502	XMZERO	-	000000C05
XM2A1	-	000000C06	XNEXT	-	000515	X2	-	000513			

START OF CONSTANTS
000335

START OF TEMPORARIES
000350

START OF INDIRECTS
000352

SPACE REQUIRED TO COMPILE
037130

SUBJECT: SUBRØUTINE AIRSPL

PURPØSE: SUBRØUTINE AIRSPL computes recovery and mass flow ratios for mixed-compression mode inlets.

METHØD: The various mass flow ratios are computed from the input airflow and by interpolation in Tables 6B and 2C. Recovery comes from Table 2B. If bypass mass flow is negative; recovery is recomputed so that the bypass mass flow is zero. A message stating that the inlet is undersized results.

USAGE: CALL AIRSPL

Inputs in CØMMØN include Mach number, air flow, capture area, and inlet tables. Other CØMMØN blocks are output.

SUBPRØGRAMS: TABU1 WARNING

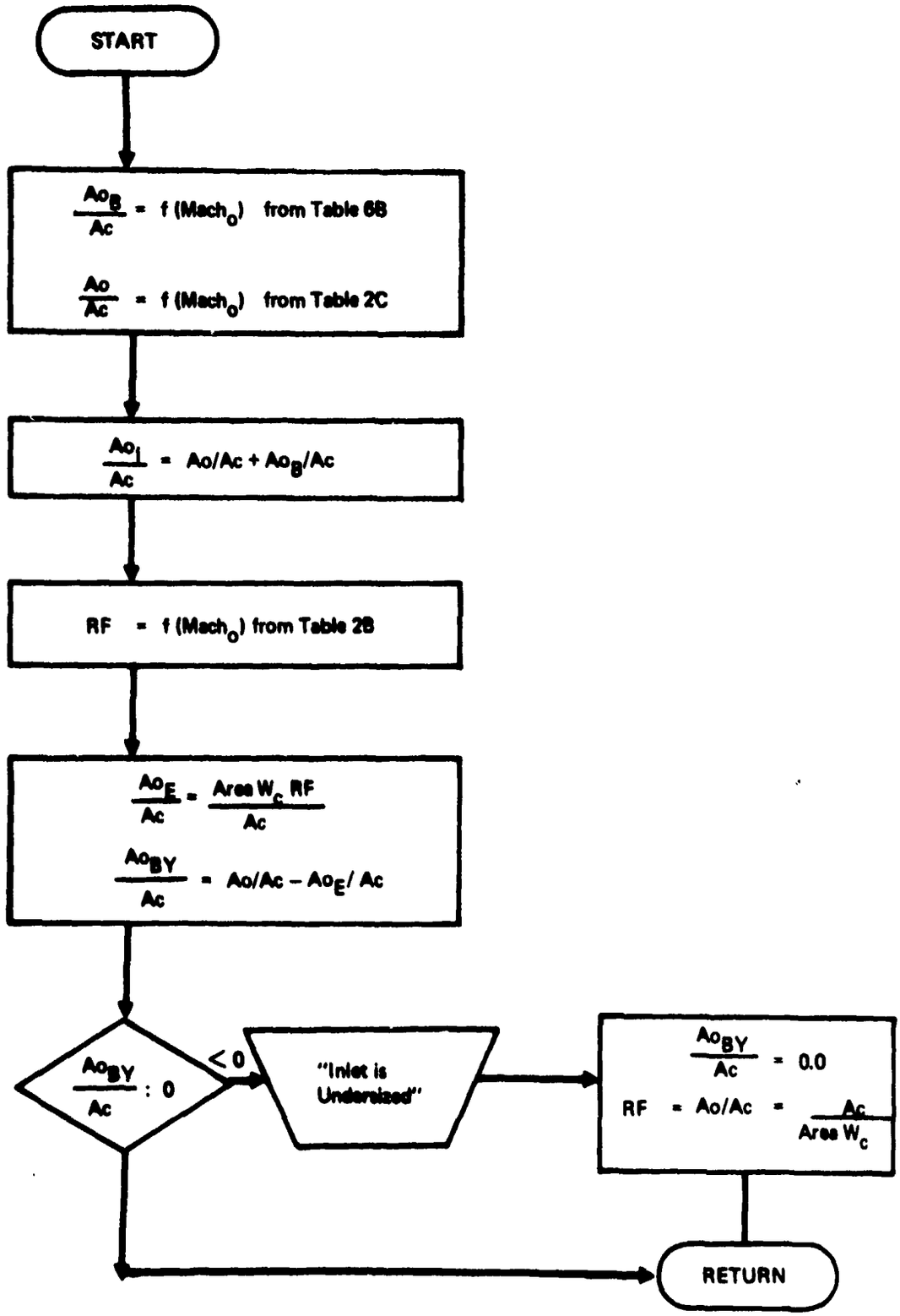


Figure 6: FLOW CHART FOR SUBROUTINE AIRSPL

```
                SUBROUTINE AIRSPL
000002          COMMON/PT2PTO/PT2PTO
000002          COMMON/AREAVL/AREA
000002          COMMON/XMACH/XMACH
000002          COMMON/PT/PT
000002          COMMON/XMZERO/XMZERO
000002          COMMON/AC/AC,AOEAC,AOBYAC,AOBAC,AOIAC
000002          COMMON/WC/WCENG,WCSEC
000002          COMMON/GAMMA/GAMMA
000002          COMMON/TAB6B/TAB6BX(10),TAB6BY(10),NT6B
000002          COMMON/TAB2B/TAB2BX(10),TAB2BY(10),NT2B
000002          COMMON/TAB2C/TAB2CX(10),TAB2CY(10),NT2C
000002          COMMON/TAB2E/TAB2EX(10),TAB2EY(10),NT2E
000002          COMMON/TAB2D/TAB2DX(10),TAB2DY(10),NT2D
000002          COMMON/AOAC/AOAC
000002          AOBAC=TABU1(XMZERO,TAB6BX,TAB6BY,  NT6B)
000005          AOAC=TABU1(XMZERO,TAB2CX,TAB2CY,  NT2C)
000011          AOIAC=AOAC+AOBAC
000014       20 PT2PTO=TABU1(XMZERO,TAB2BX,TAB2BY,  NT2B)
000017          AOEAC=AREA          *(WCENG+WCSEC)*PT2PTO/AC
000024          AOBYAC=AOAC-AOEAC
000025          IF (AOBYAC.GE.0.) GO TO 30
000027          CALL WARNING
000030          PSAVE=PT2PTO
000031          AOBYAC=0.
000032          PT2PTO=AOAC*(AC/AREA          /(WCENG+WCSEC))
000037          WRITE(6,1) AOAC,AOEAC,PSAVE,PT2PTO
000052        1 FORMAT(* INLET IS UNDERSIZED FOR OPERATION WITH THE SCHEDULED SPIL
                ALAGE AND BLEED*/5X,*AO/AC*F7.4,5X,*AOE/AC*F7.4,5X,*RECOVERY REDUCE
                BD FROM*F7.4,3H TO,F7.4)
000052       30 RETURN
000053          END
```

```
SUBPROGRAM LENGTH
000077

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       -  000056    20      -  000014    30      -  000052

EXTERNALS AND TAGS
TABU1   -  S00100    WARNING-   S00200    OUTPTC.-   S00300    END.    -  S00400

BLOCK NAMES AND LENGTHS
PT2PTC  -  000001    AREAVL  -  000001    XMACH   -  000001    PT      -  000001
XMZERO  -  000001    AC      -  000005    WC      -  000002    GAMMA   -  000001
TAB2B   -  000025    TAB2B   -  000025    TAB2C   -  000025    TAB2E   -  000025
TAB2D   -  000025    ACAC    -  000001

VARIABLE ASSIGNMENTS
AC      -  000000C06 AREA    -  000000C02 AOAC    -  000000C16 AOBAC   -  000003C06
ACBYAC  -  000002C06 ACEAC   -  000001C06 AOIAC   -  000004C06 NT28    -  000024C12
NT2C    -  000024C13 NT6B    -  000024C11 PSAVE   -  000076    PT2PTB  -  000000C01
TAB2BX  -  000000C12 TAB2BY  -  000012C12 TAB2CX  -  000000C13 TAB2CY  -  000012C13
TAB2DX  -  000000C15 TAB2DY  -  000012C15 TAB2EX  -  000000C14 TAB2EY  -  000012C14
TAB6BX  -  000000C11 TAB6BY  -  000012C11 WCENG   -  000000C07 WCSEC   -  000001C07
XMZERO  -  000000C05

START OF CONSTANTS
000054

START OF TEMPORARIES
000076

START OF INDIRECTS
000076

SPACE REQUIRED TO COMPILE
035000
```

SUBJECT: SUBRØUTINE DRG

PURPØSE: SUBRØUTINE DRG computes bottail drag for convergent-divergent nozzle.

METHØD: The built in nozzle data tables are defined in the DATA statements of this routine. Boattail angle is computed; then Mach is interrogated. If Mach is less than .9 the subsonic procedure, SUBBT, is used. Similarly if Mach is greater than or equal to 1.0 the supersonic procedure is used. If Mach is in the range from .9 to 1.0, then the subsonic procedure is used with Mach equal to .9 and the supersonic procedure with Mach equal to 1.0. Final results are obtained by interpolating between the two sets of results using Mach as the independent variable. This provides a smooth transition between the two Mach regimes.

USAGE: CALL SUBRØUTINE (DMDG, DBDJ, LDJ, EMF, EMJD, PTPF, THETA, GAM, ICØN)

DMDJ	Ratio of maximum to jet diameter
DBDJ	Ratio of base to jet diameter
LDJ	Ratio of length to jet diameter
EMF	Freestream Mach number
EMJD	Jet design Mach number
PTPF	Ratio of Jet total to free-stream static pressure
THETA	Boattail trailing edge angle - degrees
GAM	Ratio of specific heats
ICØN	= 0 circular are boattail
	= 1 conical boattail

Output in CØMMØN/A/

SUBPRØGRAMS: ATAN SUBBT SUPBT

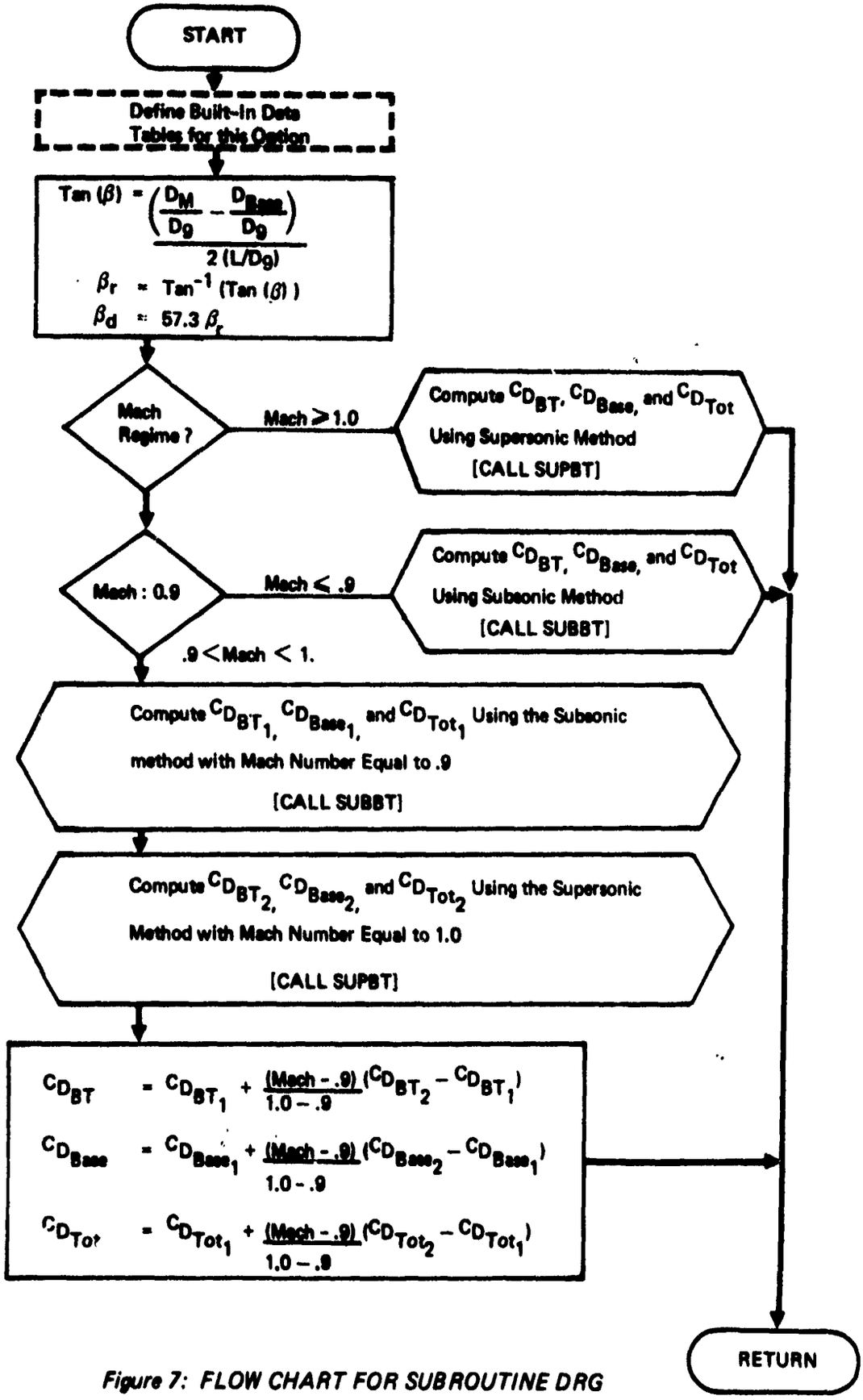


Figure 7: FLOW CHART FOR SUBROUTINE DRG

```
                SUBROUTINE DRG(DMDJ,DBDJ,LDJ,EMF,EMJD,PTPF,THETA,GAM,ICON)
000014          COMMON/A/CD,CDBS,CDBT,BETAD
000014          COMMON/DRGTB/NT1,TB1(3),TBF1(3),NT2,TB2(20),TBF2(20),NT3,TB3(10),
               1             TBF3(10),NTX4,NTY4,TBX4(5),TBY4(11),TBF4(5,11),NT5,
               2             TB5(5),TBF5(6),NTX6,NTY6,TBX6(5),TBY6(7),TBF6(5,7),
               3             NT7,TB7(7),TBF7(7),NT8,TB8(5),TBF8(5),NTX9,NTY9,
               4             TBX8(5),TBY9(9),TBF9(5,9),NTX10,NTY10,TBX10(5),
               5             TBY10(7),TBF10(5,7),NTX11,NTY11,TBX11(5),TBY11(6),
               6             TBF11(5,6),NT12A,TB12A(10),TBF12A(10),NT12B,TB12B(10)
               7            ,TBF12B(10),NT13,TB13(10),TBF13(10)
000014          DIMENSION TBP4(55),TBP6(35),TBP9(45),TBP10(35),TBP11(30)
000014          EQUIVALENCE(TBP4 (1),TBF4(1,1))
000014          EQUIVALENCE(TBP6 (1),TBF6(1,1))
000014          EQUIVALENCE(TBP9 (1),TBF9(1,1))
000014          EQUIVALENCE(TBP10(1),TBF10(1,1))
000014          EQUIVALENCE(TBP11(1),TBF11(1,1))
000014          REAL LDJ
          C        TABLE 1 - G=F(M)
000014          DATA NT1/ 3 /
000014          DATA TB1  / 1.00,1.57,2.60 /
000014          DATA TBF1 / 1.40, .67, .33 /
          C        TABLE 2 - B=F(C)
000014          DATA NT2 / 20 /
000014          DATA TB2  / .06, .08, .10, .15, .20, .25, .30, .35, .40, .45,
               1            .50, .60, .70, .80, .90,1.00,1.50,2.00,2.50,3.00 /
000014          DATA TBF2 /2.15,2.05,1.90,1.63,1.45,1.35,1.20,1.11,1.05,1.00,
               1           1.00,1.05,1.08,1.15,1.21,1.27,1.65,2.05,2.40,2.75 /
          C        TABLE 3 - K=F(PB/PF)
000014          DATA NT3 / 10 /
000014          DATA TB3  / 1.0 ,2.0 ,3.0 ,4.0 ,5.0 ,6.0 ,7.0 ,8.0 ,9.0 ,10.0 /
000014          DATA TBF3 / .820,.575,.380,.245,.190,.165,.145,.135,.125,.120 /
          C        TABLE 4 - CDBZ=F(THETA,DB/DM)
000014          DATA NTX4,NTY4 / 5,11 /
000014          DATA TBX4                    /   3.0  ,5.6  ,8.0  ,16.0 ,24.0 /
000014          DATA TBY4 /.00,.10,.20,.30,.40,.50,.60,.70,.75,.80,.90/
000014          DATA TBP4 /  .0000,  .0000,  .0000,  .0000,  .0000,                .00
               A             .0000,  .0005,  .0025,  .0055,  .0095,                .10
               B             .0005,  .0015,  .0050,  .0120,  .0190,                .20
               C             .0015,  .0030,  .0085,  .0215,  .0295,                .30
               D             .0030,  .0052,  .0140,  .0325,  .0400,                .40
               E             .0050,  .0110,  .0245,  .0430,  .0510,                .50
               F             .0100,  .0215,  .0350,  .0535,  .0615,                .60
               G             .0200,  .0320,  .0450,  .0645,  .0720,                .70
               H             .0250,  .0340,  .0510,  .0700,  .0760,                .75
               I             .0235,  .0300,  .0510,  .0700,  .0770,                .80
```

```
                 J              .0125,  .0160,  .0300,  .0410,  .0460/                    .90

      C          TABLE 5 - DCOB/DCPB=F(THETA)
000014         DATA NT5 / 6 /
000014         DATA TB5  / 0.,5.0,10.0,15.0,20.0,25.0 /
000014         DATA TBF5 / 0.,-.09,-.19,-.38,-.67,-1.1 /

      C          TABLE 6 - DCPB=F(THETA,PTP/P)
000014         DATA NTX6,NTY6 / 5,7 /
000014         DATA TBX6                      /    3.0  ,5.6  ,8.0  ,16.0 ,24.0  /
000014         DATA TBY6 / 1.0,2.0,3.0,4.0,5.0,6.0,7.0 /
000014         DATA TBP6 /  5*-.025,  5*-.030,  5*-.046,  5*-.080,                  1.0-4.0
               A              -.147,  -.142,  -.138,  -.120,  -.100,                 5.0
               B              -.194,  -.184,  -.175,  -.142,  -.115,                 6.0
               C              -.225,  -.210,  -.200,  -.160,  -.120/                 7.0

      C          TABLE 7 - DCPB/D(DJ2/DBDM)=F(PTP/P)
000014         DATA NT7 / 7 /
000014         DATA TB7  / 1.00,2.00,3.00,4.00,5.00,6.00,7.00 /
000014         DATA TBF7 /  .18, .18, .24, .48, .76,1.06,1.34 /

      C          TABLE 8 - DCPB/D(DJ/DB)=F(THETA)
000014         DATA NT8 / 5 /
000014         DATA TB8  / 0.0,10.,15.,20.,25. /
000014         DATA TBF8 / .05,.09,.10,.09,.04 /

      C          TABLE 9 - CPBZC=F(DBDM,THETA)
000014         DATA NTX9,NTY9 / 5,9 /
000014           DATA TBX9                    /    3.0  ,5.6  ,8.0  ,16.0 ,24.0 /
000014         DATA TBY9 / .1,.2,.3,.4,.5,.6,.7,.8,.9 /
000014         DATA TBP9 /  -.130,  -.040,  .010,  2*.190,                          .10
               A              -.130,  -.040,  .010,  2*.185,                          .20
               B              -.130,  -.040,  .010,  2*.175,                          .30
               C              -.130,  -.040,  .010,  2*.150,                          .40
               D              -.130,  -.040,  .010,  2*.140,                          .50
               E              -.130,  -.040,  .010,  2*.105,                          .60
               F              -.130,  -.075,  -.025,  2*.075,                         .70
               G              -.130,  -.085,  -.042,  2*.032,                         .80
               H              -.140,  -.105,  -.075,  2*-.040 /                       .90

      C          TABLE 10 - CPBZK=F(DB/DM,THETA)
000014         DATA NTX10,NTY10 / 3,7 /
000014         DATA (TBX10(I),I=1,3)          /    3.0  ,5.6  ,8.0  /
000014         DATA TBY10 / .3,.4,.5,.6,.7,.8,.9 /
000014         DATA TBP10 /  -.050, .000, .075, -.060, .010, .075, -.060, .000,
               A               .075, -.060, .000, .075, -.050, .000, . 75, .000,
               B              -.040, -.015, -.065, -.055, -.050 /

      C          TABLE 11 - CD=F(EMF,THETA)
000014         DATA NTX11,NTY11 / 5,6 /
000014         DATA TBX11                     /     7.0 ,8.0 ,10.0,14.0,20.0 /
000014         DATA TBY11 /  .400,.600,.850,.900,.925,.950 /
000014         DATA TBP11 / .016,  .023,  .030,  .034,  .046,                        .400
```

```
                A          .016,  .023,  .030,  .034,  .046,                .800
                B          .018,  .024,  .032,  .038,  .050,                .850
                C          .023,  .030,  .040,  .055,  .070,                .900
                D          .033,  .040,  .054,  .078,  .097,                .925
                E          .050,  .067,  .090,  .125,  .170 /               .950

          C        TABLE 12A - PTPF=F(CD)  6.LT.BETA.LE.12 DEGREES
 000014         DATA NT12A / 10 /
 000014         DATA TB12A /2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 6.0, 7.0, 8.0 /
 000014         DATA TBF12A/-0.002,0.0,0.0005,-.0021,-.0075,-.0125,-.0175,-.0280,
                1           -.0385,-.0495 /
          C        TABLE 12B - PTPF=F(CD)  BETA.GT.12 DEGREES
 000014         DATA NT12B / 10 /
 000014         DATA TB12B /2.0,2.5,3.0,3.5,4.0,4.5,5.0,6.0,7.0,8.0/
 000014         DATA TBF12B/-.003,0.,.0005,-.0035,-.0070,-.0070,-.0110,-.020,
                1           -.0295,.0400 /

          C        TABLE 13 - PB/PF=F(PTP/PF)
 000014         DATA NT13 / 7 /
 000014         DATA TB13  / 1.5,2.0,2.5,3.0,3.5,4.0,4.5 /
 000014         DATA TBF13 / .95,.86,.79,.73,.69,.67,.65 /

 000014         DATA RAD/57.29578/


 000014         TANB=.5*(DMDJ-DBDJ)/LDJ
 000017         BETA=ATAN(TANB)
 000021         BETAD=RAD*BETA
 000022         IF(EMF .GE. 1.)GO TO 100

          C        SUBSONIC BOATTAIL DRAG
 000030         IF(EMF .LE. .9)GO TO 50
 000032         EMS=EMF
 000033         EMF=.9
 000034         CALL SUBBT(DMDJ,DBDJ,EMF,PTPF,THETA,ICON,GAM)
 000042         CDBT1=CDBT
 000044         CDBS1=CDBS
 000045         CD1=CD
 000047         EMF=1.0
 000053         CALL SUPBT(DMDJ,DBDJ,EMF,EMDJ,PTPF,GAM,TANB)
 000061         CDBT2=CDBT
 000063         CDBS2=CDBS
 000064         CD2=CD
 000066         EMF=EMS
 000072         CDBT=CDBT1+10.*(EMF-.9)*(CDBT2-CDBT1)
 000101         CDBS=CDBS1+10.*(EMF-.9)*(CDBS2-CDBS1)
 000110         CD=CD1+10.*(EMF-.9)*(CD2-CD1)
 000117         GO TO 500
 000120      50 CONTINUE
 000120         CALL SUBBT(DMDJ,DBDJ,EMF,PTPF,THETA,ICON,GAM)
 000126         GO TO 500

          C        SUPERSONIC BOATTAIL AND BASE DRAG
```

```
000132      100 CONTINUE
000132          CALL SUPBT(DMJJ,DBDJ,EMF,EMJJ,PTPF,GAM,TANB)

000140      500 CONTINUE
000140          RETURN
000141          END
****D- ******************************
```

```
SUBPROGRAM LENGTH
000174

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
50      -  000120    100     -  000132    500     -  000140

EXTERNALS AND TAGS
ATAN    -  S00100    SUBBT   -  S00200    SUPBT   -  S00300    END.    -  S00400

BLOCK NAMES AND LENGTHS
A       -  000004    DRST3   -  000676

VARIABLE ASSIGNMENTS
EETA    -  000164    BETAU   -  000003C01 CD      -  000000C01 CDBS    -  000001C01
CDBS1   -  000167    CDBS2   -  000172    CDBT    -  000002C01 CDBT1   -  000166
CDBT2   -  000171    CD1     -  000170    CD2     -  000173    EMS     -  000165
GAM     -  000001    ICON    -  000002    NTX10   -  000443C02 NTX11   -  000524C02
NTX4    -  000105C02 NTX6    -  000233C02 NTX9    -  000346C02 NTY10   -  000444C02
NTY11   -  000525C02 NTY4    -  000106C02 NTY5    -  000234C02 NTY9    -  000347C02
NT1     -  000303C02 NT12A   -  000577C02 NT12B   -  000624C02 NT13    -  000651C02
NT2     -  000307C02 NT3     -  000060C02 NT5     -  000216C02 NT7     -  000314C02
NT8     -  000333C02 RAD     -  000162    TAN3    -  000163    TBF1    -  000064C02
TBF10   -  000461C02 TBF11   -  000541C02 TBF12A  -  000612C02 TBF12B  -  000637C02
TBF13   -  000664C02 TBF2    -  000034C02 TBF3    -  000073C02 TBF4    -  000127C02
TBF5    -  000225C02 TBF6    -  000251C02 TBF7    -  000324C02 TBF8    -  000341C02
TBF9    -  000365C02 TBP10   -  000461C02 TBP11   -  000541C02 TBP4    -  000127C02
TBP6    -  000251C02 TBP9    -  000366C02 TBX10   -  000445C02 TBX11   -  000526C02
TBX4    -  000107C02 TBX6    -  000235C02 TBX9    -  000350C02 TBY10   -  000452C02
TBY11   -  000533C02 TBY4    -  000114C02 TBY6    -  000242C02 TBY9    -  000355C02
TB1     -  000301C02 TB12A   -  000600C02 TB12B   -  000625C02 TB13    -  000652C02
TB2     -  000310C02 TB3     -  000061C02 TB5     -  000217C02 TB7     -  000315C02
TB8     -  000334C02 THETA   -  000300

START OF CONSTANTS
000142

START OF TEMPORARIES
000152

START OF INDIRECTS
000162

SPACE REQUIRED TO COMPILE
04330C

D-*******VARIABLES EXCEED VALUES IN DATA STATEMENT
     000142
```

SUBJECT: SUBRØUTINE CDCØNV

PURPØSE: SUBRØUTINE CDCØNV moniters the convergent-divergent nozzle boattail drag computation.

METHØD: CDCØNV computes the required input values for SUBRØUTINE DRG from the TEM-333 inputs. If the supersonic procedure is to be used i.e., Mach $\geq$ 1.0, these inputs include the design Mach number of the nozzle which is solved for by iteration. DRG is called; then boattail drag computed from the CD obtained.

Note: The set DRG, SUBBT, and SUPBT were in existence prior to TEM-333, hence the incorporation of this routine into the logic.

USAGE: CALL CDCØNV

Input through CØMMØN/ABIN/ and CØMMØN/AMAX/
Output includes D_S for interference routine

Flags PLUG = 0.0; BTTAB = 0.0 causes this subroutine to be called.

SUBPRØGRAMS: AEXHST DRG SQRT WARNING

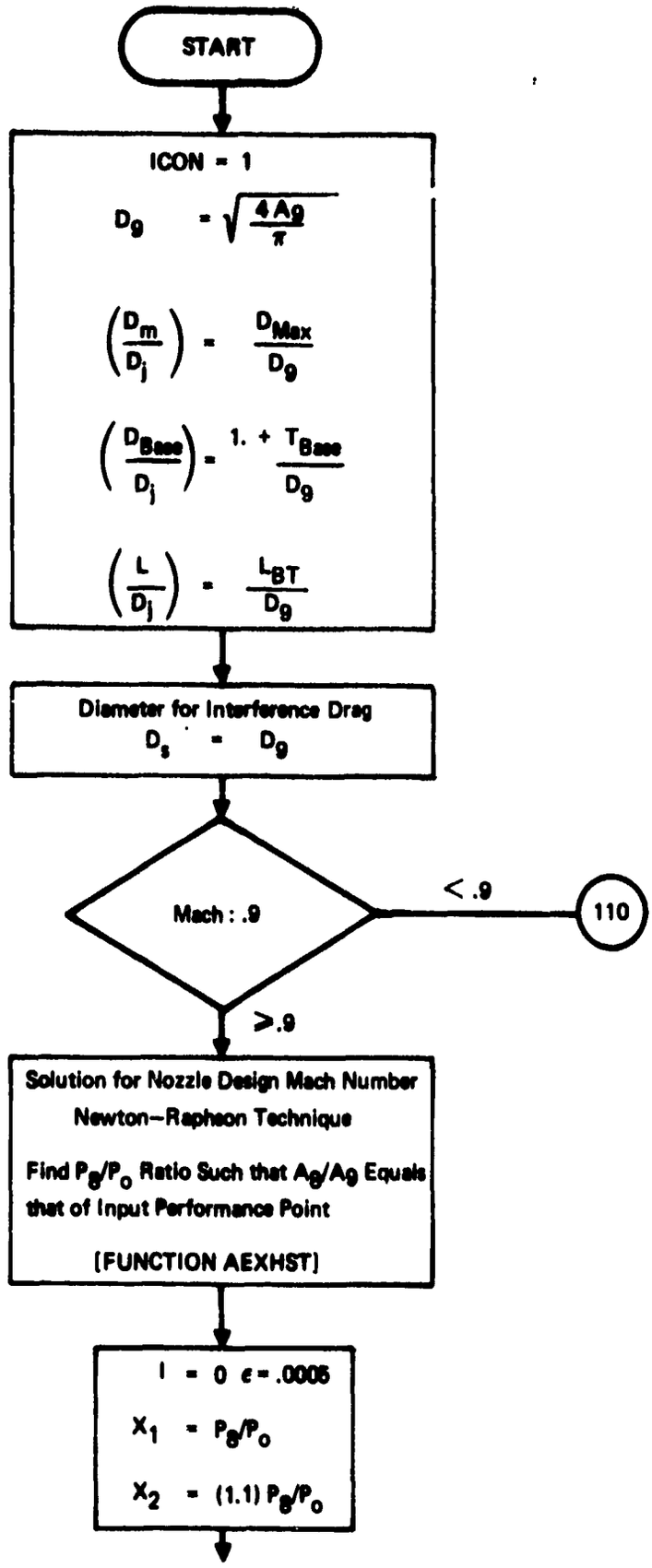


Figure 8: FLOW CHART FOR SUBROUTINE CDCONV

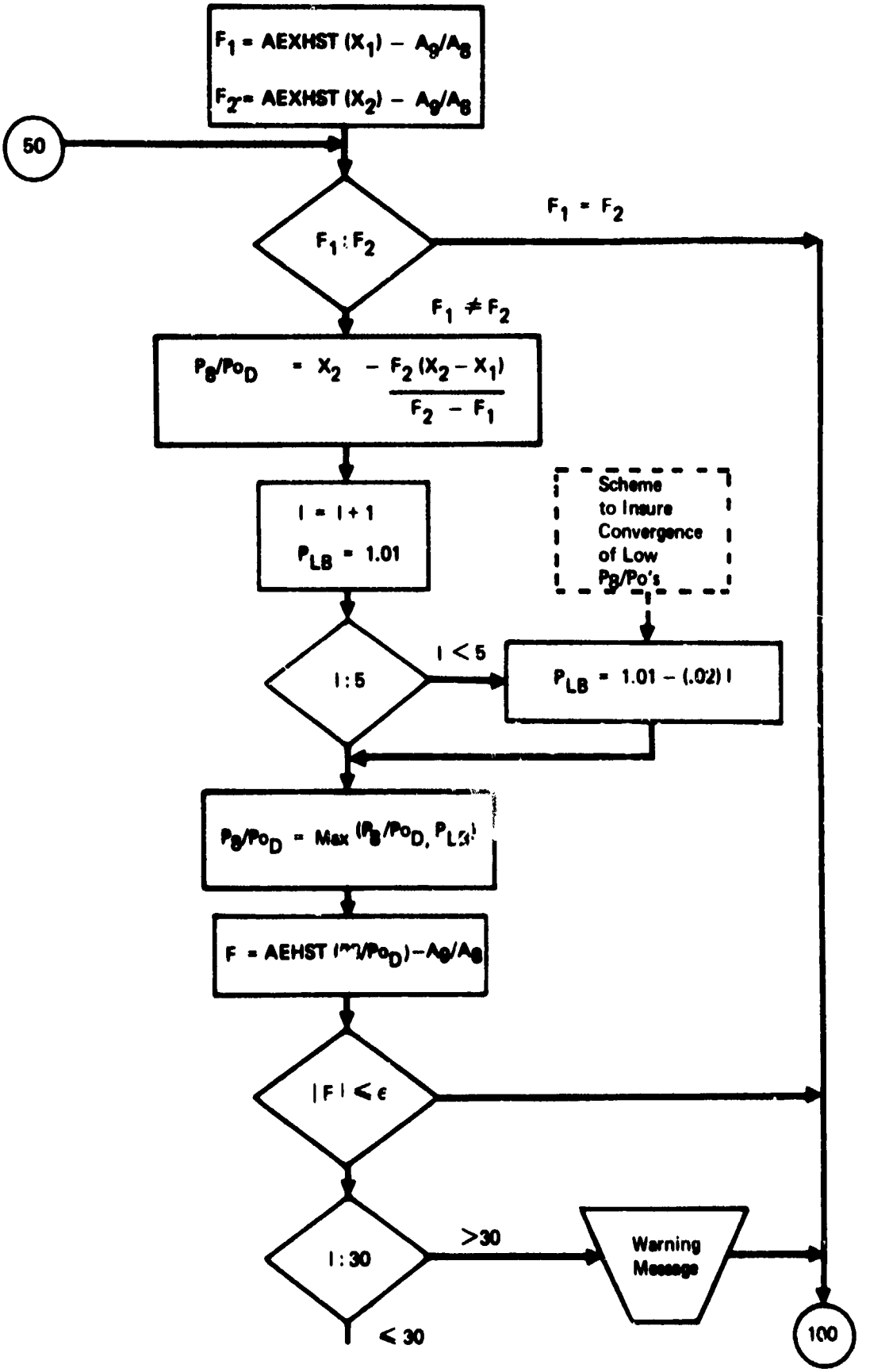


Figure 8: FLOW CHART FOR SUBROUTINE CDCONV (Cont)

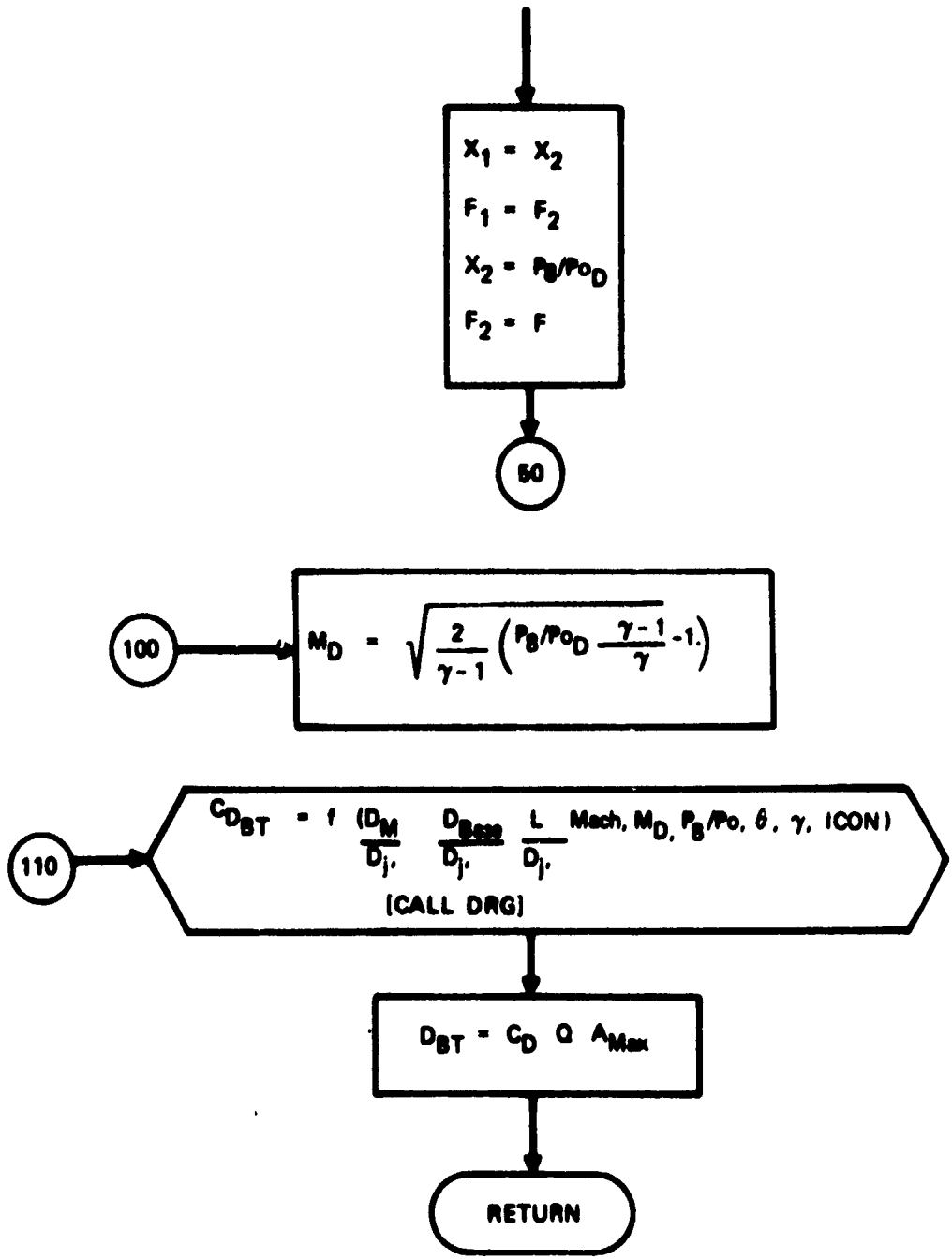


Figure 8: FLOW CHART FOR SUBROUTINE CDCONV (Concluded)

```
                SUBROUTINE COCONV
000002          COMMON/DSPACE/DSPACE
000002          COMMON/A/CD,A(3)
000002          COMMON/AMAX/AMAX
000002          COMMON/ABINP/ AE8, A8, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DMEP,
               B              DMEF, PLUG, FLAMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C              S, ABASE, BTTAB, THETA
               D,             FUTURE(11), NVAR
000002          COMMON/ABDRAG/ DBTAIL, DNI, DBASE
000002          TOL=.0005
000003          ICON=1
000004          D9=SQRT(A9/.785)
000010          DSPACE=D9
000012          DMDJ=DMAX/D9
000013          DBDJ=1.0+THICKB/D9
000016          ZLDJ=ZLBTAL/D9
000017          A9A8=A9/A8
000021          P7QODS=P7QO
000023          IF (XM.LT.9) GO TO 110
000026          I=0
000027          X1=P7QO
000027          F1=AEXMST(1.0,GAMMAP,X1)-A9A8
000034          X2= 1.1*P7QO
000036          F2=AEXMST(1.0,GAMMAP,X2)-A9A8
000043       50 IF (F2.EQ.F1) GO TO 100
000045          P7QODS=X2-F2*(X2-X1)/(F2-F1)
000052          I=I+1
000054          PLIMIT=1.01
000055          IF (I.LT.5) PLIMIT=1.10-FLOAT(I)*.02
000063          P7QODS=AMAX1(P7QODS,PLIMIT)
000067          F=AEXMST(1.0,GAMMAP,P7QODS)-A9A8
000074          IF (ABS(F).LE.TOL) GO TO 100
000100          IF (I.GT.30) GO TO 90
000103          X1=X2
000103          F1=F2
000105          X2=P7QODS
000106          F2=F
000107          GO TO 50
000110       90 CALL WARNING
000111          WRITE(6,1) PTQODS
000117        1 FORMAT(* ITERATION FOR P8/P0 DESIGN FAILED TO CONVERGE.* F8.4,* US
               AED TO COMPUTED NOZZLE DESIGN EXIT MACH NUMBER*)
000117      100 EMJD=SQRT(2./(GAMMAP-1.)*(   P7QODS)**((GAMMAP-1.)/GAMMAP)-1.0))
000133      110 CONTINUE
000133          CALL DRG(DMDJ,DBDJ,ZLDJ,XM,EMJD,P7QO,THETA,GAMMAP,ICON)
000144          DBTAIL=CD*Q*AMAX
000147          RETURN
000147          END
```

```
SUBPROGRAM LENGTH
000220

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1       -  000160    50      -  000043    90      -  000110    100     -  000117
110     -  000133

EXTERNALS AND TABS
SQRT    -  500100    AEXMST  -  500200    WARNING-   500300    OUTPTC.-   500400
RBAREX.-   500500    DRS     -  500600    END.    -  500700

BLOCK NAMES AND LENGTHS
OSPACE  -  000001    A       -  000004    AMAX    -  000001    ABINP   -  000051
ABDRAG  -  000003

VARIABLE ASSIGNMENTS
A       -  000001002 AMAX    -  000000003 A8      -  000031004 A9      -  000016004
A9i0    -  000205    CD      -  000000002 DBDJ    -  000203    DBTAIL  -  000000005
DMAX    -  000025004 DMDJ    -  000202    DSPACE  -  000000001 D9      -  000201
EMJJ    -  000217    F       -  000215    FUTURE  -  000035004 F1      -  000211
F2      -  000213    GAMMAP  -  000011004 I       -  000207    ICON    -  000200
PLIMIT  -  000214    PTJDDS  -  000216    P7QQ    -  000034004 P7QQDS  -  000206
Q       -  000015004 THETA   -  000034004 THICKB  -  000025004 TOL     -  000177
XM      -  000013004 X1      -  000210    X2      -  000212    ZLBTAL  -  000030004
ZLDJ    -  000204

START OF CONSTANTS
000150

START OF TEMPORARIES
000174

START OF INDIRECTS
000177

SPACE REQUIRED TO COMPILE
035000
```

SUBJECT: SUBRØUTINE PLUGMX

PURPØSE: SUBRØUTINE PLUGMX computes boattail drag for a plug nozzle, mixed flow.

METHØD: Boattail angle is computed from the maximum diameter, exhaust diameter, and plug length. Then Mach is checked and either the subsonic or supersonic method applied. Input tables 4.2.2 and 4.2.3 support the supersonic procedure. Boattail drag is returned.

USAGE: CALL SUBRØUTINE PLUGMX

Input from CØMMØN/ABIN/ and CØMMØN/AMAX/
Output includes D_S for interference routine

Flags set such that:

PLUG = 1.0

FLWMIX = 0.0

BTTAB = 0.0

Will cause this routine to be called

SUBPRØGRAMS:

AEXHST	ATAN	SQRT
TABU1	TABU2	

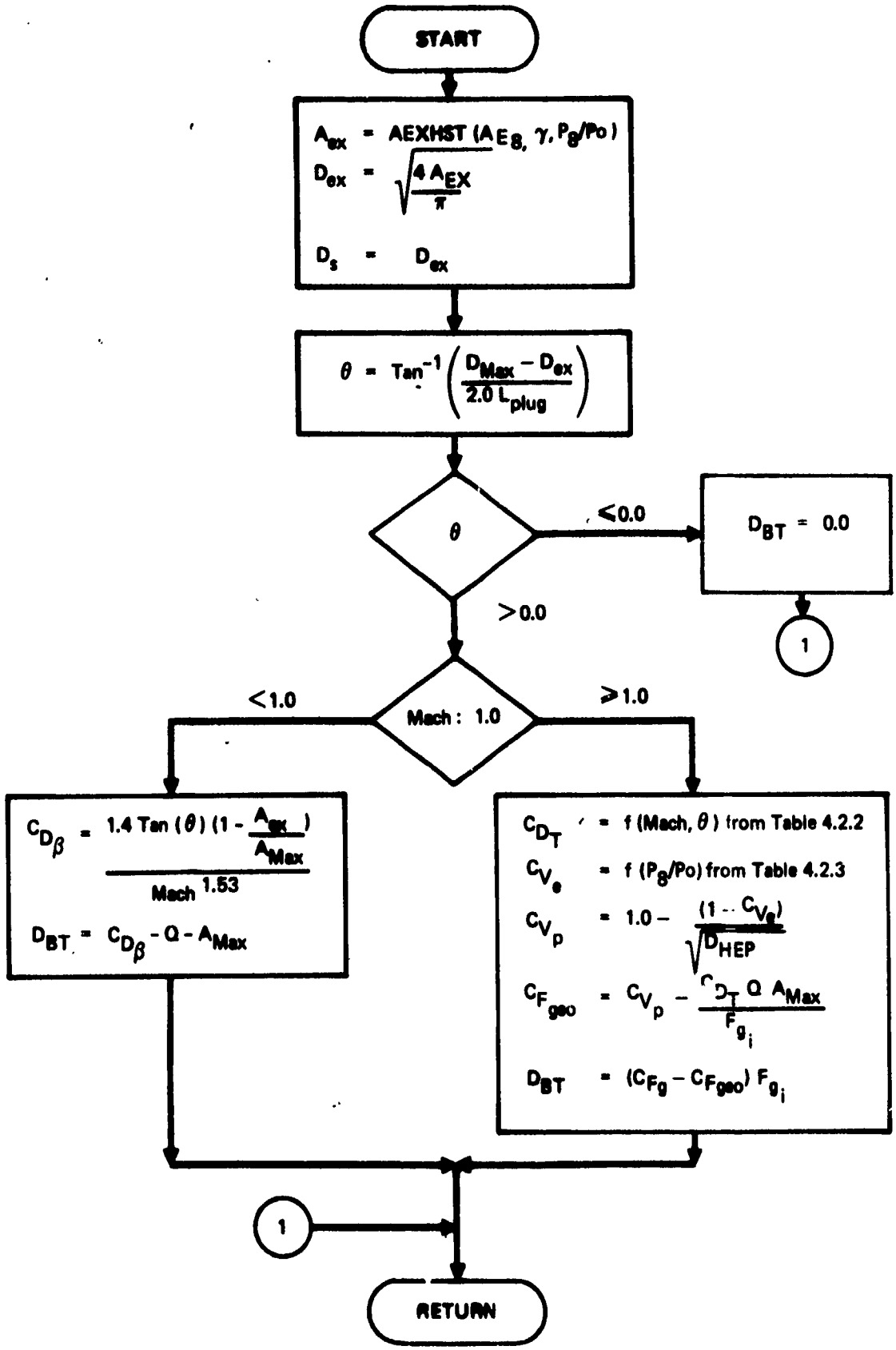


Figure 9: FLOW CHART FOR SUBROUTINE PLUGMX

```
                SUBROUTINE PLUGMX
0000 02         COMMON/AMAX/AMAX
0000 02         COMMON/DSPACE/DSPACE
0000 02         COMMON/ABINP/ AE6, A8, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A                GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DHEP,
               B                DHEF, PLUG, FLWMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C                S, ABASE, BTTAB, THETA
               D,               FUTURE(11), NVAR
0000 02         COMMON/ABDRAG/ DBTAIL, DNI, DBASE
0000 02         COMMON/ABTAB3/ABTX3(10),ABTY3(10),NTX3
0000 02         COMMON/ABTAB2/ABTX2(10),ABTY2(10),ABTZ2(10,10),NTX2,NTY2
0000 02         AEXJET=AEXMST(AE6,GAMMAP,P7QO)
0000 06         DEXJET=SQRT(AEXJET/.785)
0000 12         DSPACE=DEXJET
0000 13         TANTHA=.5*(DMAX-DEXJET)/ZLPLUG
0000 16         THETA=ATAN(TANTHA)*57.296
0000 21         IF (THETA.GT.0.0) GO TO 100
0000 23         DBTAIL=0.0
0000 23         GO TO 777
0000 24     100 IF (XM.GE.1.0) GO TO 200
0000 27         CDBETA=1.4*TANTHA*(1.0-AEXJET/AMAX)/(XM**1.53)
0000 37         DBTAIL=CDBETA*Q*AMAX
0000 41         GO TO 777
0000 41     200 CDT=TABJ2(XM,THETA,ABTX2,ABTY2,ABTZ2,    NTX2,NTY2,10,10)
0000 53         CVE=TABJ1(P7QO,ABTX3,ABTY3,  NTX3)
0000 57         CVP=1.0-((1.0-CVE)/SQRT(DHEP))
0000 65         CFGED=CVP-CDT*Q*AMAX/FGIP
0000 71         DBTAIL=(CFG-CFGED)*FGIP
0000 74     777 RETURN
0000 75         END
```

```
SUBPROGRAM LENGTH
000120

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100     -  000024    200     -  000041    777     -  000074

EXTERNALS AND TAGS
AEXHST  -  S00100    SQRT    -  S00200    ATAN    -  S00300    RBAREX.-  S00400
TABU2   -  S00500    TABU1   -  S00600    END.    -  S00700

BLOCK NAMES AND LENGTHS
AMAX    -  000001    DSPACE  -  000081    ABINP   -  000051    ABDRAG  -  001803
ABTAB3  -  000025    ABTAB2  -  000172

VARIABLE ASSIGNMENTS
ABTX2   -  000000C06 ABTX3   -  000000C05 ABTY2   -  000012C06 ABTY3   -  000012C05
ABTZ2   -  000024C06 AEXJET  -  000110    AE8     -  000000C03 AMAX    -  000001C01
COBETA  -  000113    COT     -  000114    CFG     -  000020C03 CFGEO   -  000117
CVE     -  000115    CVP     -  000116    DBTAIL  -  000000C04 DEXJET  -  000111
DHEP    -  000021C03 DMAX    -  000025C03 DSPACE  -  000000C02 FGIP    -  000006C03
FUTURE  -  000035C03 GAMMAP  -  000011C03 NTX2    -  000170C06 NTX3    -  000024C05
NTY2    -  000171C06 P700    -  000004C03 Q       -  000015C03 TANTHA  -  000112
THETA   -  000034C03 XM      -  000013C03 ZLPLUG  -  000027C03

START OF CONSTANTS
000176

START OF TEMPORARIES
000105

START OF INDIRECTS
000110

SPACE REQUIRED TO COMPILE
03560
```

SUBJECT: SUBRØUTINE PLUGNM

PURPØSE: SUBRØUTINE PLUGNM computes boattail drag for a plug nozzle, non-mixed flow.

METHØD: Boattail angle is computed from the maximum diameter, the exhaust diameter and the plug length. Then Mach is checked and either the subsonic or supersonic method applied. Input tables 4.2.2 and 4.2.3 support the supersonic procedure.

Boattail drag is returned.

USAGE: CALL SUBRØUTINE PLUGNM

Input from CØMMØN/ABIN/ and CØMMØN/AMAX)
Output includes D_S for interference routine

Flags set such that: PLUG = 1.0

FLWMIX = 0.0

BTTAB = 0.0

Will cause this routine to be called

SUBPRØGRAMS: AEXHST ATAN SQRT

TABU1 TABU2

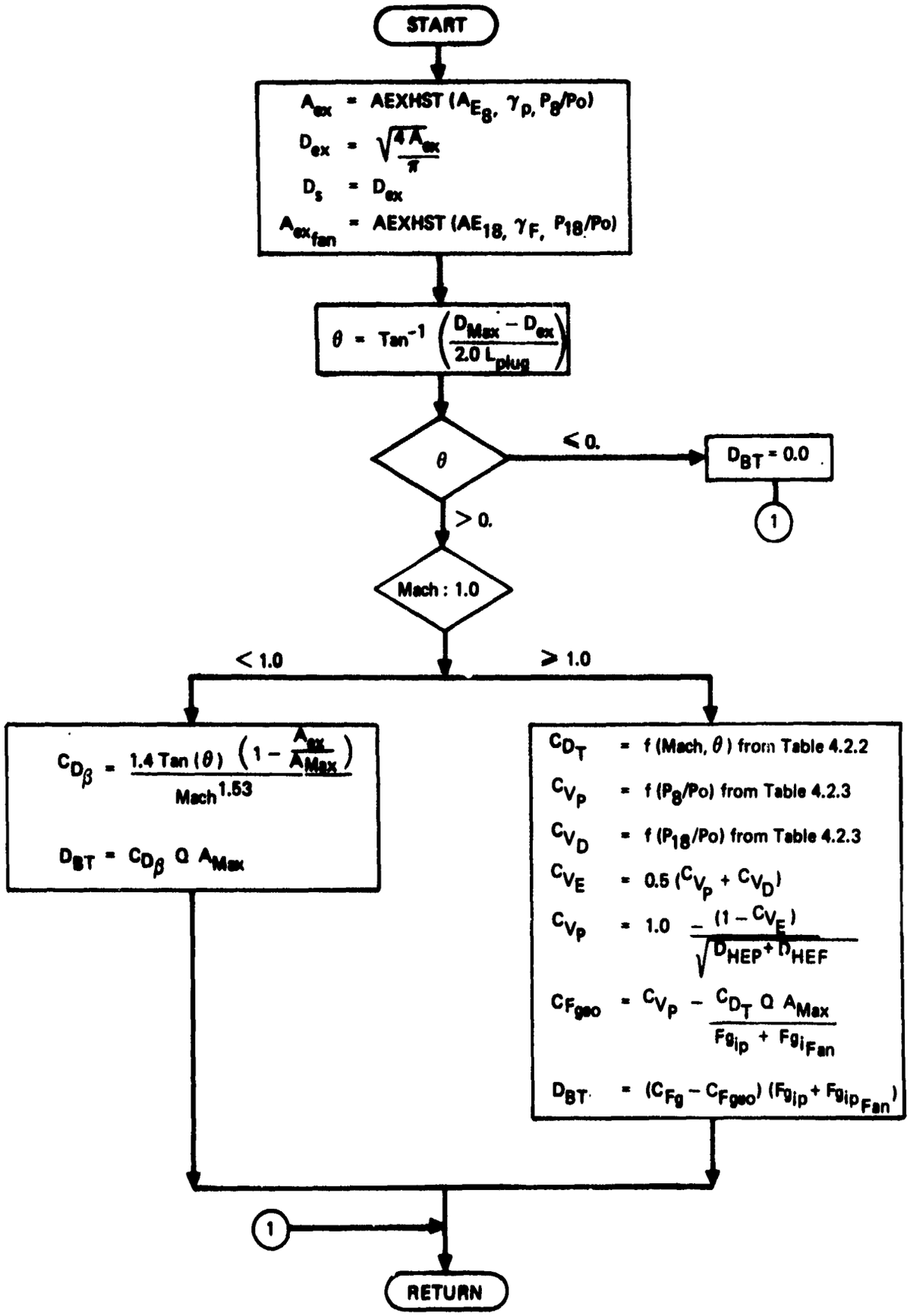


Figure 10: FLOW CHART FOR SUBROUTINE PLUGNM

```
                SUBROUTINE PLUGNM
000002          COMMON/ABINP/ AE8, A8, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DHEP,
               B              DHEF, PLUG, FLMMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C              S, ABASE, BTTAB, THETA
               D,             FUTURE(11), NVAR
000002          COMMON/A4AX/AMAX
000002          COMMON/ABDRAG/ CDTAIL, DNI, DBASE
000002          COMMON/DSPACE/DSPACE
000002          COMMON/ABTAB2/ABTX2(10),ABTY2(10),ABTZ2(10,10),NTX2,NTY2
000002          COMMON/ABTAB3/ABTX3(10),ABTY3(10),NTX3
000002          AEXFAN=AEXHST(AE18,GAMMAF,P17QO)
000006          AEXPRM=AEXHST(AE8,GAMMAP,P7QO)
000011          AEXJET=AEXPRM+AEXFAN
000013          DEXJET=SQRT(AEXJET/.785)
000017          DSPACE=DEXJET
000020          TANTHA=.5*(DMAX-DEXJET)/ZLPLUG
000023          THETA=ATAN(TANTHA)*57.296
000026          IF (THETA.GT.0.0) GO TO 100
000030          DBTAIL=0.0
000030          GO TO 777
000031      100 IF (XM.GE.1.0) GO TO 200
000034          CDTAIL=1.4*TANTHA*(1.0-AEXJET/AMAX)/(XM**1.53)
000044          DBTAIL=CDTAIL*Q*AMAX
000046          GO TO 777
000046      200 CDT=TABJ2(XM,THETA,ABTX2,ABTY2,ABTZ2,    NTX2,NTY2,10,10)
000060          CVP=TABJ1(P7QO,ABTX3,ABTY3,  NTX3)
000064          CVD=TABJ1(P17QO,ABTX3,ABTY3,  NTX3)
000070          CVE=.5*(CVP+CVD)
000073          CVP=1.0-((1.0-CVE)/SQRT(DHEP+DHEF))
000103          CFGE2=CVP-CDT*Q*AMAX/(FGIP+FGIF)
000111          DBTAIL=(CFG-CFGE2)*(FGIP+FGIF)
000115      777 RETURN
000116          END
```

```
SUBPROGRAM LENGTH
000145

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100     -  000031    200     -  000046    777     -  000115

EXTERNALS AND TAGS
AEXHST  -  S00100    SQRT    -  S00200    ATAN    -  S00300    RBAREX.-  S00400
TA3U2   -  S00500    TA3U1   -  S00600    END.    -  S00700

BLOCK NAMES AND LENGTHS
ABINP   -  000351    AMAX    -  000301    ABDRAG  -  000003    DSPACE  -  000001
ABTA32  -  000172    ABTA33  -  000025

VARIABLE ASSIGNMENTS
ABTX2   -  000000C05 ABTX3   -  000000C06 ABTY2   -  000012C05 ABTY3   -  000012C06
ABTZ2   -  000024C05 AEXFAN  -  000132    AEXJET  -  000134    AEXPRM  -  000133
AEIB    -  000022C01 AEB     -  000020C01 AMAX    -  000000C02 CDT     -  000140
COTAIL  -  000137    CFG     -  000020C01 CFGE2   -  000144    CVD     -  000142
CVE     -  000143    CVP     -  000141    DBTAIL  -  000000C03 DEXJET  -  000135
DMEP    -  000022C01 DMEP    -  000021C01 DMAX    -  000025C01 DSPACE  -  000000C04
FGIF    -  000107C01 FGIP    -  000106C01 FUTURE  -  000035C01 GAMMAF  -  000012C01
GAMMAP  -  000011C01 NTX2    -  000017C05 NTX3    -  000024C06 NTY2    -  000171C05
P7JJ    -  000005C01 P7JJ    -  000004C01 Q       -  000015C01 TANTHA  -  000136
THETA   -  000134C01 XM      -  000013C01 ZLPLUS  -  000027C01

START OF CONSTANTS
000117

START OF TEMPORARIES
000126

START OF INDIRECTS
000132

SPACE REQUIRED TO COMPILE
03570C
```

SUBJECT: SUBRØUTINE ABINPT

PURPØSE: SUBRØUTINE ABINPT reads the nozzle performance data deck, cards 4.1 to 4.2.6.

METHØD: The input deck is read. The thickness of the annular base is doubled to account for top and bottom when compared to the total jet diameter. If A_{10} was input the table of $Mach_9$ vs. A_8/A_9 needed by subroutine CABTAB and the P_8/P_{AMB} lower thrust are computed. Tables not indicated by the input flags are not read.

USAGE: CALL ABINPT

All CØMMØN blocks present are defined by calling this subprogram.

SUBPRØGRAMS: TABL1 TABL2 TABL3

```
                SUBROUTINE ABINPT
000002          COMMON /NOZARR/ A(40)
000002          COMMON/ABTAB1/ ABTAB1(120),NTX1,NTY1
000002          COMMON/ABTAB2/ABTAB2(120),NTX2,NTY2
000002          COMMON/ABTAB3/ABTAB3(20),NTX3
000002          COMMON/ABTAB4/ABTAB4(20),NTX4
000002          COMMON/ABTAB5/ABTAB5(120),NTX5,NTY5
000002          COMMON /ABTAB6/ABTAB6(1030),NTX6,NTY6,NTZ6
000002          COMMON/P8PAL/P8PAL
000002          COMMON/AREA10/AREA10,BYA10
000002          COMMON /XMACH9/XMACH9(10),ABA9T(10)
000002          EQUIVALENCE (GAMMAP,A(10))
000002          DATA A /40*0.0/
000002          DATA XMACH9 / 2.8, 2.6, 2.4, 2.2, 2.0, 1.8, 1.6, 1.4, 1.2, 1.0 /
000002          READ (5,2) (A(K),K=20,29)
000010           READ(5,2) A(10),AREA10
000020          A(23)=A(23)+A(23)
000022          WRITE(6,3) (A(K),K=20,29),A(10),AREA10
000033          AREA10=AREA10*144.
000035          BYA10=1.0/AREA10
000036          WRITE(6,1)
000042          IF (A(25).NE.1.0) GO TO 9
000044          WRITE(6,5)
000050             CALL TABL2(ABTAB1,10,10,NTX1,NTY1)
000054        9 IF (A(25).EQ.0.0) GO TO 10
000055          CALL TABL2(ABTAB2,10,10,NTX2,NTY2)
000061          WRITE(6,1)
000065          CALL TABL1(ABTAB3,10,NTX3)
000070       10 IF (AREA10.LE.0.0) GO TO 15
000072          WRITE (6,7)
000075        7 FORMAT(*0CAB TABLE*)
000075          CALL TABL3(ABTAB6,10,10,10,NTX6,NTY6,NTZ6)
000104          GP1=GAMMAP+1.0
000106          GM1=GAMMAP-1.0
000107          GP1GM1=GP1/(2.0*GM1)
000112          DO 23 K=1,10
000113          ABA9T(K)=(XMACH9(K)*(.5*GP1)**GP1GM1)/((1.0+.5*GM1*XMACH9(K)**2)
               A **GP1GM1)
000130       23 CONTINUE
000132          WRITE(6,8) XMACH9,ABA9T
000142        8 FORMAT(*0COMPUTED TABLE - MACH9 VS AB/A9*/(1X,10F12.5))
000142          P8PAL=1.0/((2.0/GP1)**(GAMMAP/GM1))
000152          WRITE(6,11) P8PAL
000157       11 FORMAT(*0LOWER LIMIT OF P8/PAMB FOR P9 CALCULATION* F10.5)
000157       15 WRITE(6,4)
000164          CALL TABL1(ABTAB4,10,NTX4)
000166          WRITE(6,5)
000172          CALL TABL2(ABTAB5,10,10,NTX5,NTY5)
000176          RETURN
000177        1 FORMAT(1H0)
000177        2 FORMAT(10F7.0)
000177        3 FORMAT(*1NOZZLE INPUT DECK*/*0NOZZLE TYPE DESIGNATION* F15.1,20X,
```

```
          1*ENGINE TYPE DESIGNATION*F17.1/* MAXIMUM DIAMETER - IN*F18.1,20X,
          2*NOZZLE BASE OUTER DIAM - IN*F13.1/* OVERALL PLUG LENGTH - IN*F15.
          31,20X,*NOZZLE BOATTAIL LENGTH - IN*F13.1/* NOZZLE SPACING - IN*
          4F20.1,20X,*BASE FLOW AREA - SQ IN*F18.1/* BOATTAIL DRAG TABLE FLAG
          5*F15.1,20X,*BOATTAIL TRAILING EDGE ANGLE - DEG*F6.2/* PRIMARY  SPE
          6CIFIC HEAT RATIO*F11.2,20X,*A10 - SQ FT* F29.2 )
CC3177      4 FORMAT(*0BASE DRAG TABLE*)
CC3177      5 FORMAT(*0INTERFERENCE DRAG TABLE*)
CC3177      6 FORMAT *0BOATTAIL DRAG TABLE*)
CC3177        END
```

```
SUBPROGRAM LENGTH
000333

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      -  000231   2      -  000233   3      -  000235   4      -  000312
5      -  000310   6      -  000322   7      -  000206   8      -  000213
9      -  000054   10     -  000070   11     -  000222   15     -  000157

EXTERNALS AND TAGS
INPUTC.-  S00100   OUTPTC.-  S00200   TABL2  -  S00300   TABL1  -  S00400
TABL3  -  S00500   RBAREX.-  S00600   END.   -  S00700

BLOCK NAMES AND LENGTHS
NOZARR -  000056   ABTAB1 -  000172   ABTAB2 -  000172   ABTAB3 -  000025
ABTAB4 -  000025   ABTAB5 -  000172   ABTAB6 -  002011   PSPAL  -  000001
AREA10 -  000002   XMACM9 -  000024

VARIABLE ASSIGNMENTS
A      -  000000C01 ABTAB1 -  000000C02 ABTAB2 -  000000C03 ABTAB3 -  000000C04
ABTAB4 -  000000C05 ABTAB5 -  000000C06 ABTAB6 -  000000C07 AREA10 -  000000C11
ABAST  -  000012C12 BTA10  -  000001C11 GAMMAP -  000011C01 GM1    -  000331
GP1    -  000330    GP1GM1 -  000332    X      -  000327    NTX1   -  000170C02
NTX2   -  000170C03 NTX3   -  000024C04 NTX4   -  000024C05 NTX5   -  000170C06
NTX6   -  000000C07 NTY1   -  000171C02 NTY2   -  000171C03 NTY5   -  000171C06
NTY6   -  000000C07 NTZ5   -  000010C07 PSPAL  -  000000C10 XMACM9 -  000000C12

START OF CONSTANTS
000200

START OF TEMPORARIES
000325

START OF INDIRECTS
000327

SPACE REQUIRED TO COMPILE
*03600
```

SUBJECT: SUBRØUTINE SUBBT

PURPØSE: SUBRØUTINE SUBBT computes boattail drag for a convergent-divergent nozzle when Mach < .9.

METHØD: The quantity $D_J^2/(D_B \cdot D_M)$ is tested. If it is greater than 0.25 boattail drag is determined directly from built in tables XI and XII. If not, an emperical method to develop base and boattail drag is used.

USAGE: CALL SUBBT (DMDJ, DBDJ, EMF, PTPF, THETA, ICØN, GAM)

DMDJ	Ratio of maximum to jet diameter
DBDJ	Ratio of base to jet diameter
EMF	Free-stream Mach number
PTPF	Ratio of jet total to free-stream static pressure
THETA	Boattail trailing edge angle - degrees
GAM	Ratio of specific heats
ICØN	= 0 circular arc boattail = 1 conical boattail

SUBPRØGRAMS: CABTAB TABU1 TABU2

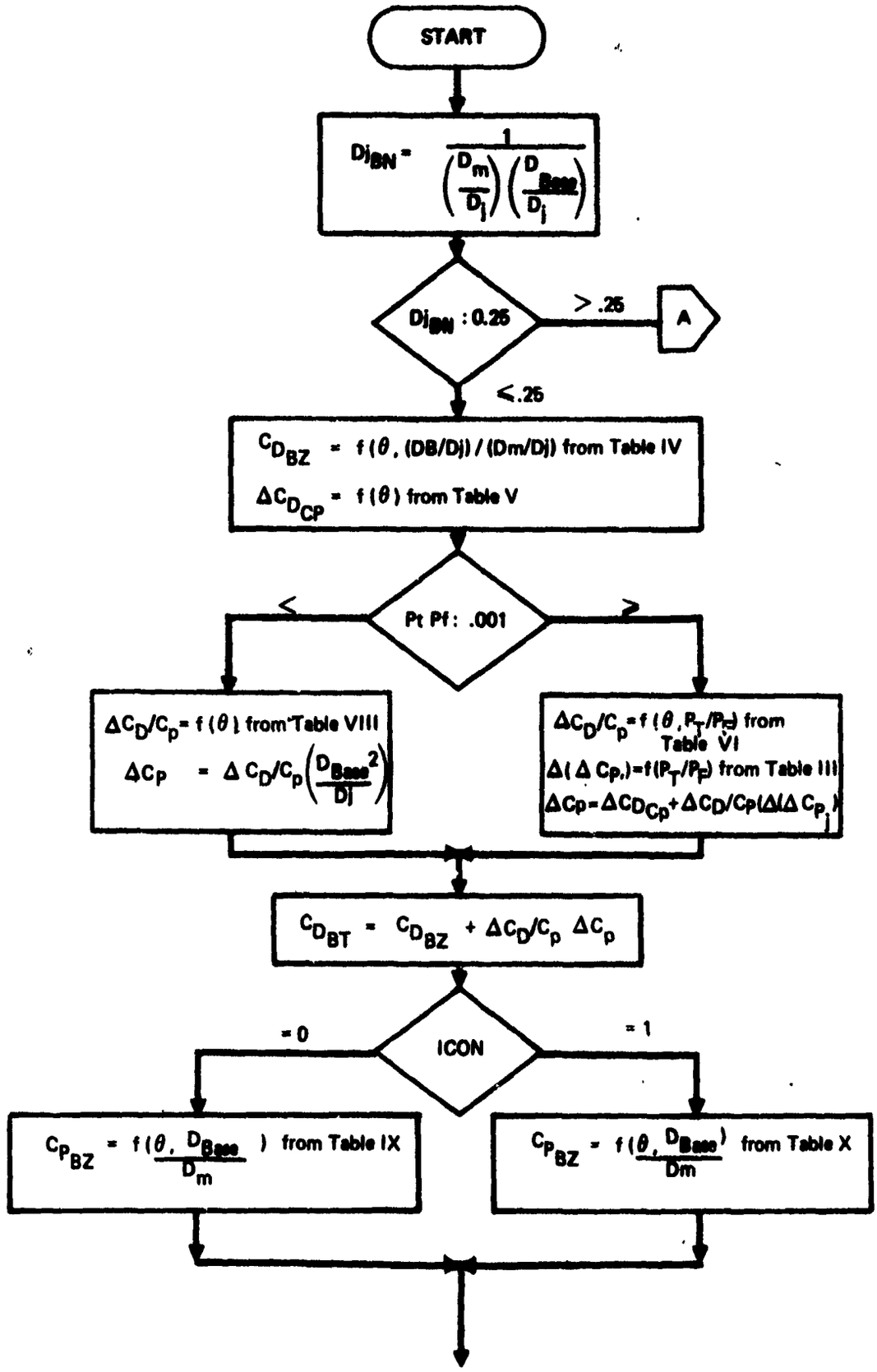


Figure 11: FLOW CHART FOR SUBROUTINE SUBBT

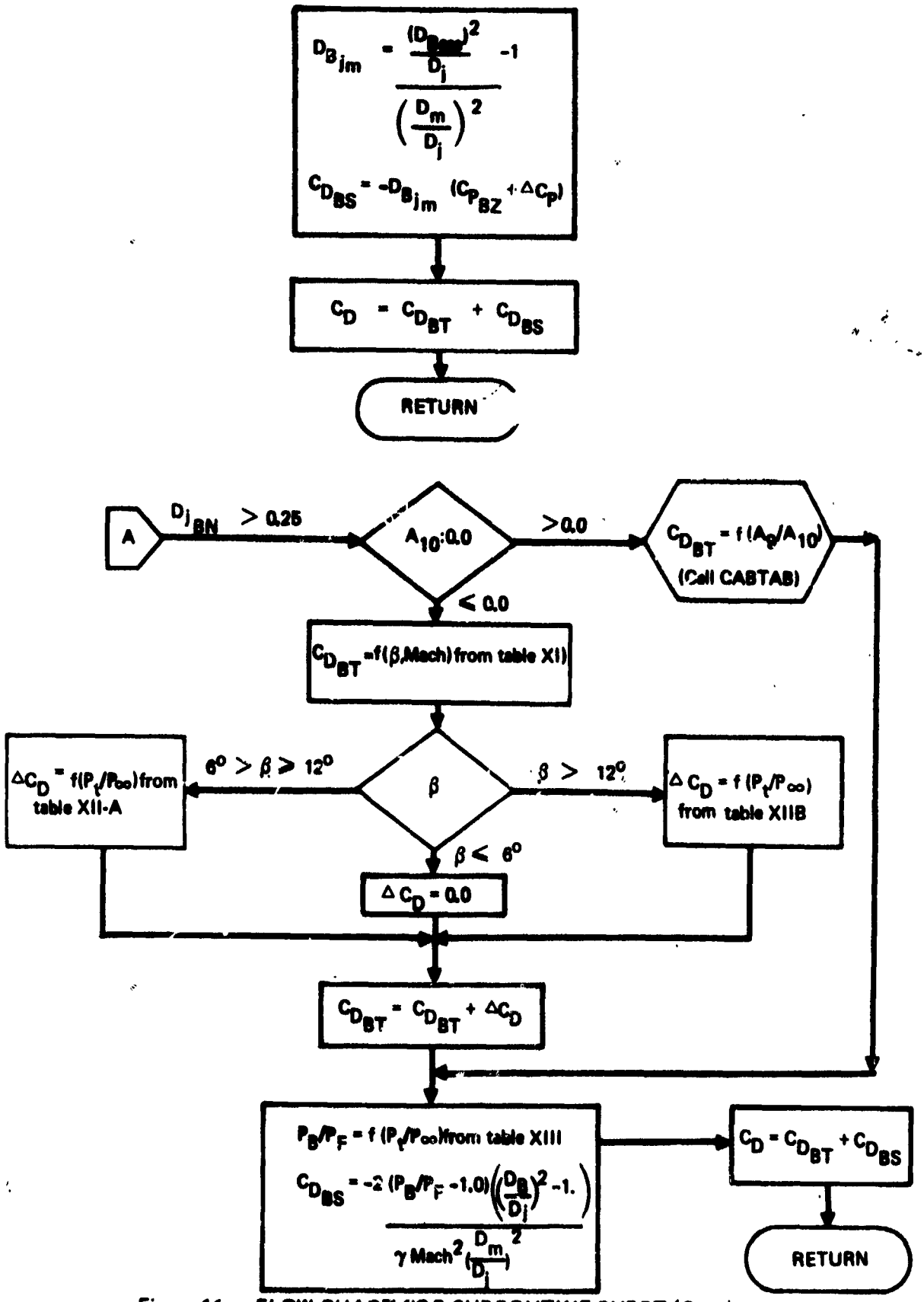


Figure 11: FLOW CHART FOR SUBROUTINE SUBBT (Cont)

```
                SUBROUTINE SUBBT(DMOJ,DBOJ,EMF,PTPF,THETA,ICON,GAM)

000012          COMMON/AREA10/AREA10,BYA10
000012          COMMON/A/CD,CDBS,CDBT,BETAD

000012          COMMON/DRGTB/NT1,TB1(3),TBF1(3),NT2,TB2(20),TBF2(20),NT3,TB3(10),
                1            TBF3(10),NTX4,NTY4,TBX4(5),TBY4(11),TBF4(5,11),NT5,
                2            TB5(6),TBF5(6),NTX6,NTY6,TBX6(5),TBY6(7),TBF6(5,7),
                3            NT7,TB7(7),TBF7(7),NT8,TB8(5),TBF8(5),NTX9,NTY9,
                4            TBX9(5),TBY9(9),TBF9(5,9),NTX10,NTY10,TBX10(5),
                5            TBY10(7),TBF10(5,7),NTX11,NTY11,TBX11(5),TBY11(6),
                6            TBF11(5,6),NT12A,TB12A(10),TBF12A(10),NT12B,TB12B(10)
                7           ,TBF12B(10),NT13,TB13(10),TBF13(10)

000012          REAL LDJ.

000012          DJBM=1./(DMOJ*DBOJ)
000014          IF(DJBM .GT. .25)GO TO 500
000017          DBOM=DBOJ/DMOJ
000020          CDBZ=TABU2(THETA,DBOM,TBX4,TBY4,TBF4,NTX4,NTY4,5,11)
000034          DCDCP=TABU1(THETA,TB5,TBF5,NT5)
000043          IF(PTPF .GE. .001)GO TO 100
000046          DCPDJ=TABU1(THETA,TB8,TBF8,NT8)
000055          DCP=DCPDJ/DBOJ**2
000057          GO TO 200

000060     100  CONTINUE
000060          DCDCP=TABU2(THETA,PTPF,TBX6,TBY6,TBF6,NTX6,NTY6,5,7)
000074          DDCPJ=TABU1(PTPF,TB7,TBF7,NT7)
000103          DCP=DCPD+DDCPJ*DJBM
000106     200  CONTINUE
000106          CDBT=CDBZ+DCDCP*DCP

000111          IF(ICON .EQ. 1)GO TO 300
000113          CPBZ=TABU2(THETA,DBOM,TBX9,TBY9,TBF9,NTX9,NTY9,5,9)
000127          GO TO 400

000130     300  CONTINUE

          C        *NOTE* TABLE 10 - MAXIMUM THETA = 8 DEGREES

000130          CPBZ=TABU2(THETA,DBOM,TBX10,TBY10,TBF10,NTX10,NTY10,5,7)
000145     400  CONTINUE
000145          DBJM=(DBOJ*DBOJ-1.)/(DMOJ*DMOJ)
000150          CDBS=-DBJM*(CPBZ+DCP)
000153          CD=CDBS+CDBT
000155          GO TO 600

000156     500  CONTINUE
000156          IF(AREA10.GT.0.0) GO TO 700
000161          CD=TABU2(BETAD,EMF,TBX11,TBY11,TBF11,NTX11,NTY11,5,6)
000175          DCD=0.0
```

```
000176          IF (BETAO.LE.6.0) GO TO 550
000201          IF (BETAO.GT.12.0) GO TO 525
000204          DCO=TABU1(PTPF,TB12A,TBF12A,NT12A)
000212          GO TO 550
000213      700 CALL CABTAB(COBT,EMF)
000215          GO TO 575
000221      525 DCO=TABU1(PTPF,TB12B,TBF12B,NT12B)
000230      550 COBT=CO+DCO

000232      575 CONTINUE
000232          PBPF=TABU1(PTPF,T313,TBF13,NT13)
000241          COBS=-2.*(PBPF-1.)*(DBOJ*DBOJ-1.)/(GAM*EMF*EMF*DMOJ*DMOJ)

000253          CO=COBT+COBS

000255      600 CONTINUE
000255          RETURN
000256          END
```

```
SUBPROGRAM LENGTH
000313

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100   -  000060   200   -  000106   300   -  000130   400   -  000145
500   -  000156   525   -  000221   550   -  000230   575   -  000232
600   -  000255   700   -  000213

EXTERNALS AND TAGS
TA3U2 -  S00100   TA3U1 -  S00200   CABTAB -  S00300   END.  -  S00400

BLOCK NAMES AND LENGTHS
AREA1C -  000002   A    -  000004   DRGTB  -  000676

VARIABLE ASSIGNMENTS
AREA1C -  000000C01 BETAD  -  000003C02 CD     -  000000C02 CDBS   -  000001C02
CDBT   -  000002C02 CDBZ   -  000301    CPBZ   -  000307    DBDM   -  000300
DBJM   -  000310    DCD    -  000311    DCDCP  -  000302    DCP    -  000304
DCPD   -  000306    DCPDJ  -  000303    DDCPJ  -  000305    DJBM   -  000277
GA4    -  000000    LDJ    -  000276    NTX10  -  000443C03 NTX11  -  000524C03
NTX4   -  000165C03 NTX5   -  000233C03 NTX3   -  000346C03 NTY10  -  000444C03
NTY11  -  000526C03 NTY4   -  000106C03 NTY5   -  000234C03 NTY9   -  000347C03
NT12A  -  000577C03 NT12B  -  000624C03 NT13   -  000651C03 NT5    -  000216C03
NT7    -  000314C03 NT6    -  000333C03 PBPF   -  000312    TBF1   -  000004C03
TBF10  -  000461C03 TBF11  -  000541C03 TBF12A -  000612C03 TBF12B -  000637C03
TBF13  -  000564C03 TBF2   -  000034C03 TBF3   -  000073C03 TBF4   -  000127C03
TBF5   -  000225C03 TBF6   -  000251C03 TBF7   -  000324C03 TBF8   -  000341C03
TBF9   -  000366C03 TBX10  -  000445C03 TBX11  -  000526C03 TBX4   -  000107C03
TBX5   -  000235C03 TBX9   -  000350C03 TBY10  -  000452C03 TBY11  -  000533C03
TBY4   -  000114C03 TBY5   -  000242C03 TBY9   -  000355C03 TB1    -  000001C03
TB12A  -  000600C03 TB12B  -  000625C03 TB13   -  000652C03 TB2    -  000010C03
TB3    -  000061C03 TB5    -  000217C03 TB7    -  000315C03 TB8    -  000334C03

START OF CONSTANTS
000257

START OF TEMPORARIES
000272

START OF INDIRECTS
000276

SPACE REQUIRED TO COMPILE
037000
```

SUBJECT: SUBRØUTINE SUPBT

PURPØSE: SUBRØUTINE SUPBT computes boattail and base drag for a convergent-divergent nozzle when Mach > 1.0.

METHØD: Boattail drag is computed using Table I, then if a base has been defined base drag is computed and added to compute a total drag.

USAGE: CALL SUPBT (DMDJ, DBDJ, EMF, EMJD, PTPF, GAM, TANB)

DMDJ	Ratio of maximum of jet diameter
DBDJ	Ratio of base to jet diameter
EMF	Free-stream Mach number
EMJD	Jet design Mach number
PTPF	Ratio of jet total to free-stream static pressure
GAM	Ratio of specific heats
TANB	Tangent of boattail chord angle

SUBPRØGRAM: ALØG CABTAB EXP

SQRT TABU1

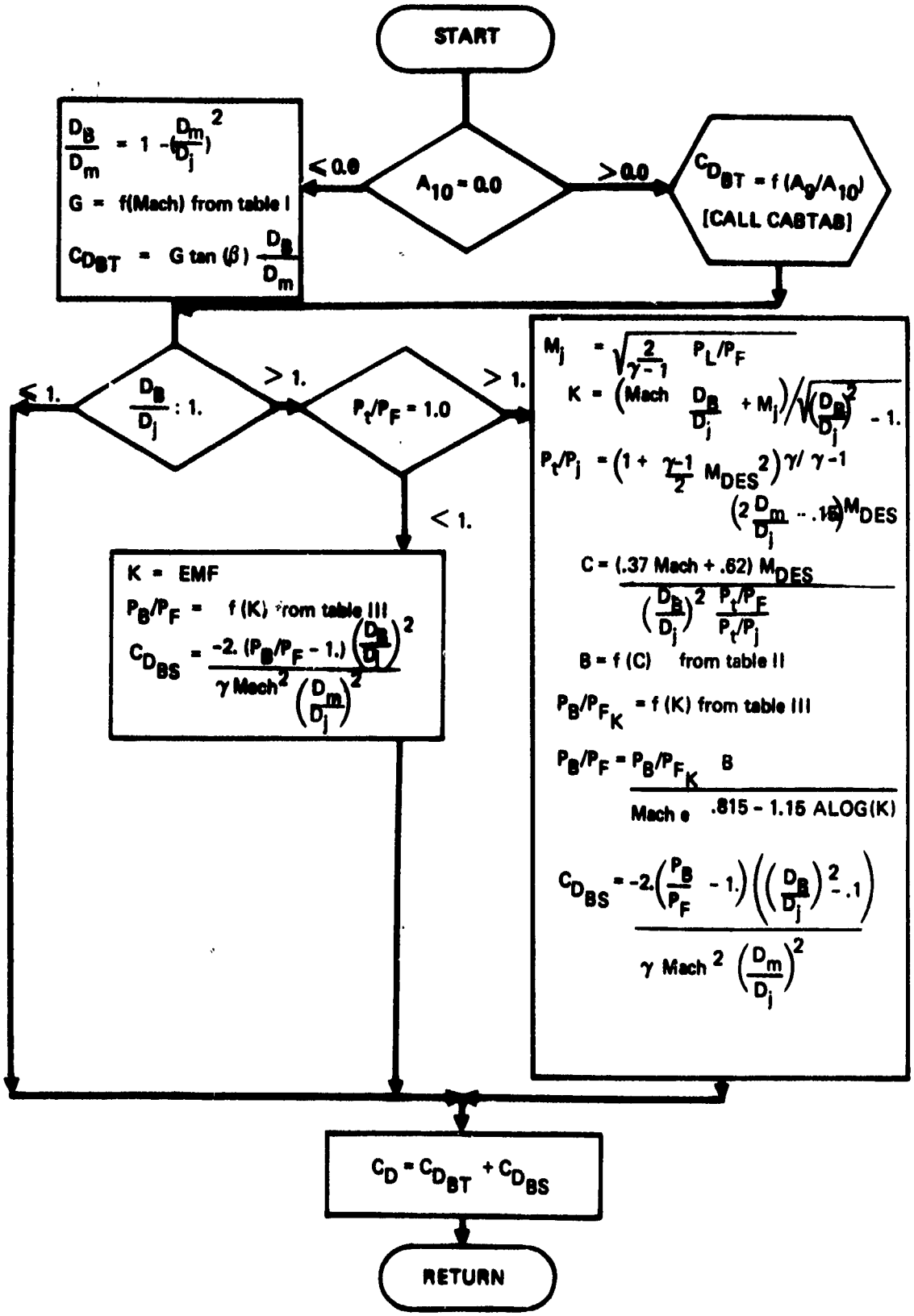


Figure 12: FLOW CHART FOR SUBROUTINE SUPBT

```
      SUBROUTINE SUPBT(DMOJ,DBOJ,EMF,EMJO,PTPF,GAM,TANB)
000012  COMMON/A/CD,CDBS,CDBT,BETAO
000012  COMMON/AREA10/AREA10,BYA10
000012  COMMON/DRGTB/NT1,TB1(3),TBF1(3),NT2,TB2(20),TBF2(20),NT3,TB3(10),
      1          TBF3(10),NTX4,NTY4,TBX4(5),TBY4(11),TBF4(5,11),NT5,
      2          TB5(5),TBF5(6),NTX6,NTY6,TBX6(5),TBY6(7),TBF6(5,7),
      3          NT7,TB7(7),TBF7(7),NT8,TB8(5),TBF8(5),NTX9,NTY9,
      4          TBX9(5),TBY9(9),TBF9(5,9),NTX10,NTY10,TBX10(5),
      5          TBY10(7),TBF10(5,7),NTX11,NTY11,TBX11(5),TBY11(6),
      6          TBF11(5,6),NT12A,TB12A(10),TBF12A(10),NT12B,TB12B(10)
      7         ,TBF12B(10),NT13,TB13(10),TBF13(10)
000012  REAL LDJ,K
000012  IF (AREA10.LE.0.0) GO TO 50
000013  CALL CABTAB(CDBT,EMF)
000015  GO TO 100
000021   50 CONTINUE
000021  GM1=GAM-1.
000023  DBOMF=1.-(DBOJ/DMOJ)**2
000026  G=TABU1(EMF,TB1,TBF1,NT1)
000038  CDBT=G*TANB*DBOMF
000040  100 CONTINUE
000040  IF(DBOJ .LE. 1.)GO TO 300
000043  DMOJ2=DMOJ*DMOJ
000044  DBOJ2=DBOJ*DBOJ
000045  IF(PTPF .GT. 1.)GO TO 200
000050  K=EMF
000051  PBPF=TABU1(K,TB3,TBF3,NT3)
000055  CDBS=-2.*(PBPF-1.)*DBOJ2/(GAM*EMF*EMF*DMOJ2)
000067  GO TO 300
000070  200 CONTINUE
000070  EMJ=SQRT(2./GM1*(PTPF**(GM1/GAM)-1.))
000105  K=(EMF*DBOJ+EMJ)/SQRT(DBOJ2-1.)
000120  PTPJ=(1.+.5*GM1*EMJO*EMJO)**(GAM/GM1)
000130  D=(.37*EMF+.62)*EMJO**((.2*DBOJ-.15)*EMJO)/DBOJ2*PTPF/PTPJ
000145  B=TABU1(D, TB2,TBF2,NT2)
000151  PBPFK=TABU1(K,TB3,TBF3,NT3)
000155  PBPF=PBPFK*B/(EMF* EXP(.815-1.15*ALOG(K)))
000201  CDBS=-2.*(PBPF-1.)*(DBOJ2-1.)/(GAM*EMF*EMF*DMOJ*DMOJ)
000211  300 CONTINUE
000211  CD=CDBT+CDBS
000213  RETURN
000214  END
```

```
SUBPROGRAM LENGTH
000260

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
50      -  000021    100     -  000040    200     -  000070    300     -  000211

EXTERNALS AND TAGS
CASTAB  -  S00100    TABJ1   -  S00200    SQRT    -  S00300    RBAREX.-  S00400
EXP     -  S00500    ALOG    -  S00600    END.    -  S00700

BLOCK NAMES AND LENGTHS
A       -  000004    AREA10  -  000002    ORGTB   -  000676

VARIABLE ASSIGNMENTS
AREA10  -  000000C02 B       -  000256    C       -  000255    CD      -  000000C01
CD35    -  000001C01 CD3T    -  000002C01 DBDJ2   -  000251    DBDMF   -  000246
DMDJ2   -  000250    EMJ     -  000253    G       -  000247    GM1     -  000245
K       -  000244    LDJ     -  000243    NT1     -  000000C03 NT2     -  000007C03
NT3     -  000360C03 PBPF    -  000252    PBPFK   -  000257    PTPJ    -  000254
TAN8    -  000006    TBF1    -  000004C03 TBF10   -  000461C03 TBF11   -  000541C03
TBF12A  -  000612C03 TBF12B  -  000637C03 TBF13   -  000664C03 TBF2    -  000034C03
TBF3    -  000073C03 TBF4    -  000127C03 TBF5    -  000225C03 TBF6    -  000251C03
TBF7    -  000324C03 TBF8    -  000341C03 TBF9    -  000366C03 TBX10   -  000445C03
TBX11   -  000526C03 TBX4    -  000167C03 TBX6    -  000235C03 TBX9    -  000350C03
TBY13   -  000452C03 TBY11   -  000533C03 TBY4    -  000114C03 TBY6    -  000242C03
TBY9    -  000359C03 TB1     -  000001C03 TB12A   -  000600C03 TB12B   -  000625C03
TB13    -  000652C03 TB2     -  000010C03 TB3     -  000051C03 TB5     -  000217C03
TB7     -  000315C03 TB8     -  000334C03

START OF CONSTANTS
000215

START OF TEMPORARIES
000227

START OF INDIRECTS
000243

SPACE REQUIRED TO COMPILE
036500
```

SUBJECT: SUBRØUTINE DRAGAB

PURPØSE: SUBRØUTINE DRABAG monitors the afterbody drag calculation.

METHØD: The input array is transferred to the afterbody CØMMØN block ABINP. A check is made on Mach number. If Mach is equal to zero drag is set equal to zero and no further afterbody calculations performed. If Mach is greater than zero, boattail drag, nozzle interference, and base drag routines are called to compute the various drags. The total airplane drag is returned for nozzle interference and base drag so these are converted to one engine values. The output array is loaded with the various C_D's based upon area AMAX (drag may have been computed using a different area).

USAGE: CALL DRAGAB (ABIN, ABØUT)

ABIN (40) - Input array

ABØUT (10) - Output array

The input array is defined as follows:

ABIN

1	AE8	Primary nozzle effective area - in^2
2	A8	Primary nozzle geometric thrust area - in^2
3	AE18	Fan nozzle effective area - in^2
4	A18	Fan nozzle geometric thrust area - in^2
5	P_8/P_0	Primary nozzle pressure ratio
6	P_{18}/P_0	Fan nozzle pressure ratio
7	FGIP	Primary nozzle ideal thrust (actual mass flow) lb
8	FGIF	Fan nozzle ideal thrust (actual mass flow) lb
9	PC	Power setting
10	GAMMAP	Primary flow specific heat ratio

ABIN (continued)

11	GAMMAF	Fan flow specific heat ratio
12	XM	Free-stream Mach number
13	PAMB	Ambient static pressure psia
14	Q	Free-stream dynamic pressure psia
15	A9	Primary nozzle exit area in^2
16	A19	Fan nozzle exit area in^2
17	CFG	Nozzle internal performance coefficient
18	DHEP	
19	DHEF	
20	PLUG	Nozzle type designation
21	FLWMIX	Mixed flow flag
22	DMAX	Maximum diameter, in.
23	THICKB	Total thickness of annular base, in.
24	ZLPLUG	Overall plug length, in.
25	ZLBTAL	Nozzle boattail length, in.
26	S	Nozzle centerline spacing, in.
27	ABASE	Base flow area, in^2.
28	BTTAB	Boattail table flag*
29	THETA	
30	XDNI	Number of nozzle interference spacings
32-40		Not defined

SUBPRØGRAMS: DBASER DBØATR DINTFR

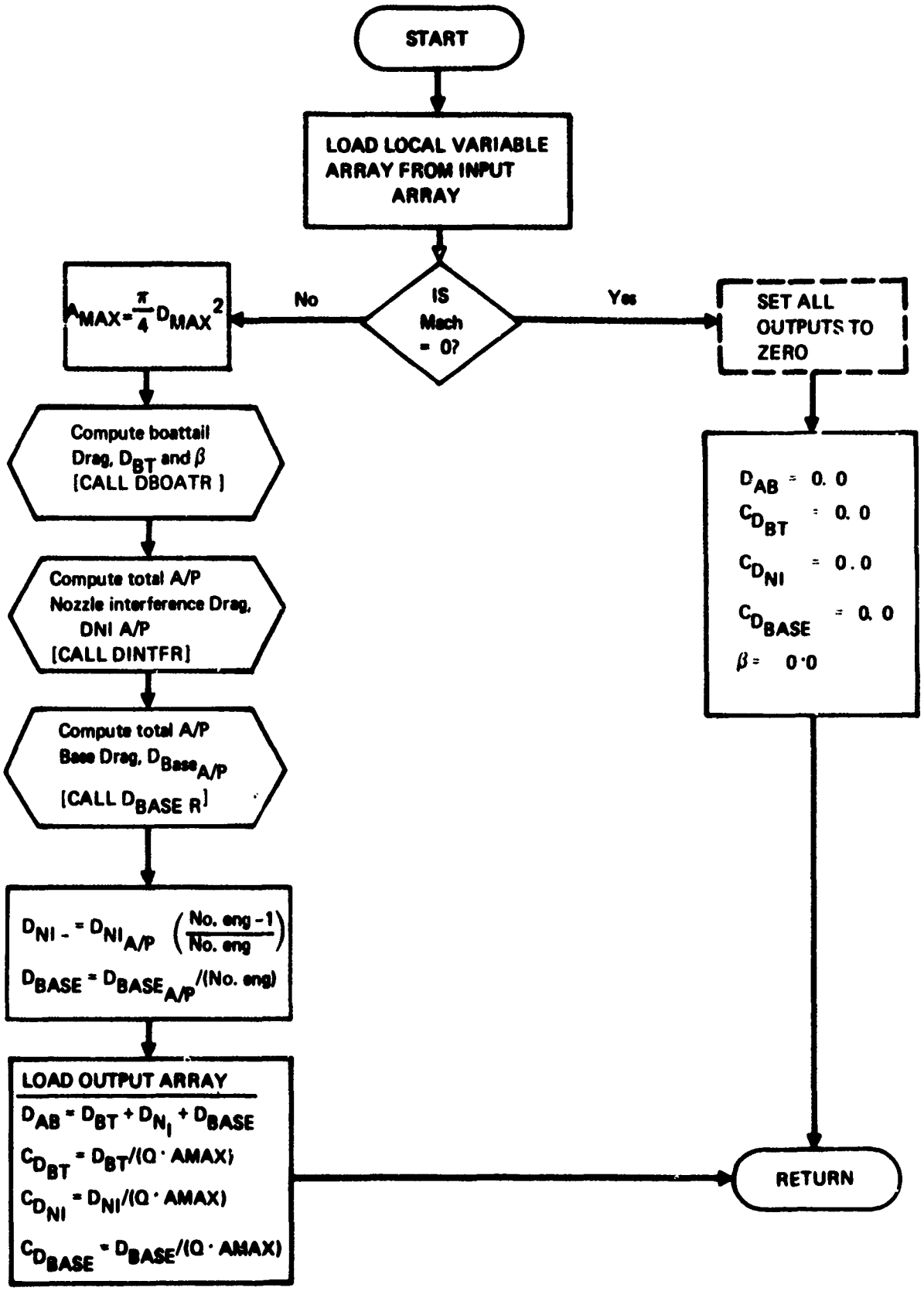


Figure 13: FLOW CHART FOR SUBROUTINE DRAGAB

```
                SUBROUTINE DRAGAB(ABIN,ABOUT)
000035          DIMENSION ABIN(40),ABOUT(10)
000035          COMMON/ABINP/ABINP(48),NVAR
000035          COMMON/ABDRAG/ DBTAIL, DNI, DBASE
000035          COMMON/A/AAA(3),BETAD
000035          COMMON/AMAX/AMAX
000035          EQUIVALENCE (ABINP(12),XM), (ABINP(14),Q), (ABINP(30),XDNI),
               A            (ABINP(31),XDBASE), (ABINP(72),DMAX)
000035          DATA NVAR/31/
000035          DATA PIBY4/.7854/
          C RETRIEVE INPUTS FROM ENGINE PROGRAM
          C          AE8    =  ABIN(1)
          C          A8     =  ABIN(2)
          C          AE18   =  ABIN(3)
          C          A18    =  ABIN(4)
          C          P7D    =  ABIN(5)
          C          P17D   =  ABIN(6)
          C          FGIP   =  ABIN(7)
          C          FGIF   =  ABIN(8)
          C          PC     =  ABIN(9)
          C          GAMMAP=   ABIN(10)
          C          GAMMAF=   ABIN(11)
          C          XM     =  ABIN(12)
          C          PAMB   =  ABIN(13)
          C          Q      =  ABIN(14)
          C          A9     =  ABIN(15)
          C          A19    =  ABIN(16)
          C          CFG    =  ABIN(17)
          C          DMEP   =  ABIN(18)
          C          DMEF   =  ABIN(19)
          C          PLUG   =  ABIN(20)    NOZZLE TYPE DESIGNATION
          C          FLMIX=    ABIN(21)    ENGINE TYPE DESIGNATION
          C          DMAX   =  ABIN(22)
          C          DMBASE=   ABIN(23)
          C          ZLPLUG=   ABIN(24)
          C          ZLBTAL=    ABIN(25)
          C          S      =  ABIN(26)
          C          ABASE  =  ABIN(27.
          C          BTTAB  =  ABIN(28)    BOATTAIL TABLE INDICATOR
000035          DO 10 I=1,NVAR
000036             ABINP(I)=ABIN(I)
000010       10 CONTINUE
000013          IF (XM.LE.0.0) GO TO 15
000014          AMAX=PIBY4*DMAX**2
000017          GAMAX=1.0/(Q*AMAX)
          C CALL SUBPROGRAMS TO COMPUTE VARIOUS DRAGS
000021          CALL DBOATR
000022          CALL DINTER
000023          CALL DBASER
          C     CALL DSCRBR
000024          GO TO 20
000026       15 DO 16 J=1,5
```

```
0000 30        18 ABOUT(J)=0.0
0000 33           RETURN
          C STORE OUTPUT TO ENGINE PROGRAM
0000 34        20 CONTINUE
0000 34           DNI=DNI*XDNI
0000 36           DBASE=DBASE*XDBASE
0000 37           ABOUT(1)=DBTAIL+DNI+DBASE
0000 42           ABOUT(2)=DBTAIL*QAMAX
0000 43           ABOUT(3)=DBASE*QAMAX
0000 44           ABOUT(4)=DNI*QAMAX
0000 45           ABOUT(5)=BETAO
0000 47           RETURN
0000 47           END
```

```
SUBPROGRAM LENGTH
000056

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
15      -  000026    18      -  000030    20      -  000034

EXTERNALS AND TASS
OBDATR  -  S00100    DINTFR  -  S00200    DBASER  -  S00300    END.    -  S00400

BLOCK NAMES AND LENGTHS
ABINP   -  000051    ABDRAG  -  000003    A       -  000004    AMAX    -  000001

VARIABLE ASSIGNMENTS
AXA     -  000000C03 ABINP   -  000000C01 AMAX    -  000000C04 BETAD   -  000003C03
DBASE   -  000002C02 DBTAIL  -  000000C02 DMAX    -  000025C01 DNI     -  000001C02
I       -  000053    J       -  000055    NVAR    -  000050C01 PIBY4   -  000052
C       -  000015C01 QAMAX   -  000054    XDBASE  -  000035C01 XDNI    -  000035C01
XM      -  000015C01

START OF CONSTANTS
000050

START OF TEMPORARIES
000051

START OF INDIRECTS
000052

SPACE REQUIRED TO COMPILE
035400
```

SUBJECT: SUBRØUTINE DBØATR

PURPØSE: SUBRØUTINE DBØATR monitors the calculation of boattail drag.

METHØD: The input flags PLUG, FLWMIX, and BTTAB are interrogated to determine which boattail method is desired and the appropriate routine selected.

USAGE: CAL DBØATR

Input through CØMMØN/ABIN/

SUBPRØGRAMS: CDCØNV DBTTB PLUGMIX PLUGNM

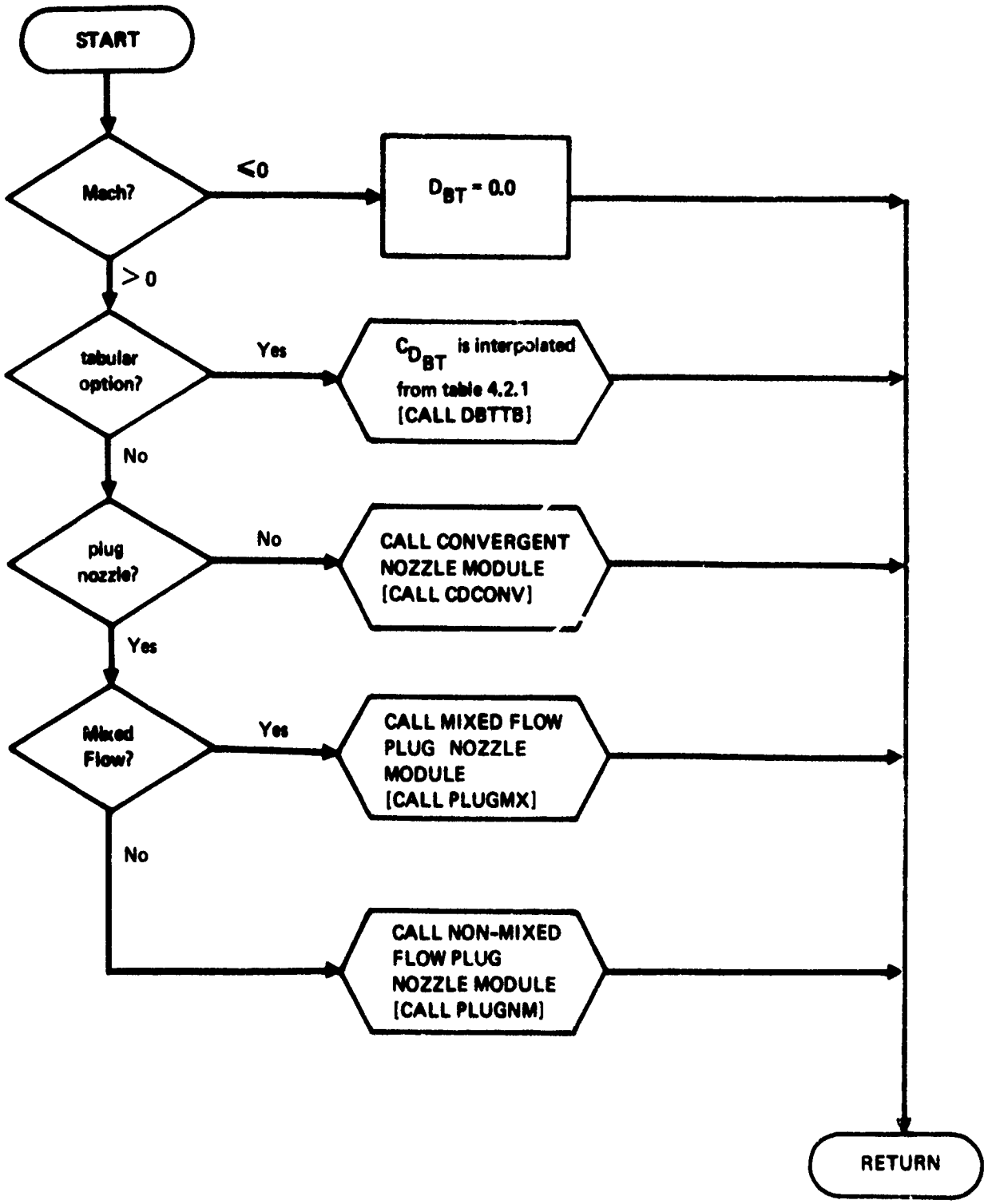


Figure 14: FLOW CHART FOR SUBROUTINE DBOATR

```
                SUBROUTINE DBOATR
0000 02         COMMON/ABINP/ AE8, A8, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DMEP,
               B              DMEF, PLUG, FLWMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C              S, ABASE, BTTA3, THETA
               D,             FUTURE(11), NVAR
0000 02         COMMON/ABDRAG/ DBTAIL, DNI, DBASE
          C     SELECT PROPER OPTION
0000 02         DBTAIL=0.0
0000 03         IF (XM.EQ.0.0) RETURN
0000 05         IF (BTTA3.EQ.1.0) GO TO 300
0000 07         IF (PLUG.EQ.0.0)  GO TO 200
0000 10         IF (FLWMIX.EQ.1.0) GO TO 100
          C NON-MIXED ENGINE FLOW - PLUG NOZZLE
0000 12         CALL PLUGNM
0000 13         GO TO 400
          C MIXED ENGINE FLOW - PLUG NOZZLE
0000 14     100 CALL PLUGMX
0000 15         GO TO 400
          C C-D OR CONVERGENT NOZZLE
0000 16     200 CALL CDCONV
0000 17         GO TO 400
          C TABULAR INPUT
0000 20     300 CALL DBTTB
0000 21     400 RETURN
0000 22         END
```

```
SUBPROGRAM LENGTH
000024

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100    -  000014    200    -  000016    300    -  000020    400    -  000021

EXTERNALS AND TAGS
PLJGNM -  SC0100    PLJGMX -  SJ0200    CDCJNV -  SC0300    OBTTB  -  SC0400
END.   -  SJ0500

BLOCK NAMES AND LENGTHS
ABINP  -  000051    ABDRAG -  000003

VARIABLE ASSIGNMENTS
ETTAB  -  000033C01 OBTAIL -  000000C02 FLW4IX -  000024C01 FUTURE -  000035C01
PLJG   -  000023C01 XM     -  000013C01

START OF CONSTANTS
000023

START OF TEMPORARIES
000024

START OF INDIRECTS
000024

SPACE REQUIRED TO COMPILE
035200
```

SUBJECT: SUBRØUTINE CABTAB

PURPØSE: SUBRØUTINE CABTAB computes afterbody drag as a function of A_9/A_{10}, P_9/P_{AMB}, and Mach.

METHØD: The drag coefficient is input in tabular form in input Table 4.2.4. Mach and A_9/A_{10} are known; the suprogram computes P_9/P_{AMB} after interpolating $Mach_9 = f(A_8/A_9)$ from a table generated by the program. The table generated in the input routine ABINP is developed as follows:

For $Mach_9$ = 2.8, 2.6, 2.4 ..., 1.0 compute

$$A_8/A_9 = \left(\frac{\gamma+1}{2}\right)^{\frac{\gamma+1}{2(\gamma-1)}} Mach_9 \left(1+\frac{\gamma-1}{2} Mach_9^{\,2}\right)^{\frac{-\gamma+1}{2(\gamma-1)}}$$

The resulting array of A_8/A_9 is strictly increasing; hence serves as an independent variable array with $Mach_9$ as the dependent array. The subroutine also checks P_8/P_0 against 1 lower bound value of $P_8/P_{O_{LB}} = 1/($

The $C_{D_{AB}}$ obtained from Table 4.2.4 is based on an area A_{10} whereas the program uses A_{MAX} (most of the time these two inputs will have the same value), consequently the C_D is ratioed by A_{10}/A_{MAX} before returning to the calling program.

USAGE: CALL CABTAB (CAB, AM)

Output CAB - Afterbody drag coefficient

AM Free-stream Mach number

Input from CØMMØN/ABINP/,/AMAX/,/AREA10/,/P8PAL/, /XMACH9/, and /ABTAB6/

SUBPRØGRAMS: TABU1 TABU3

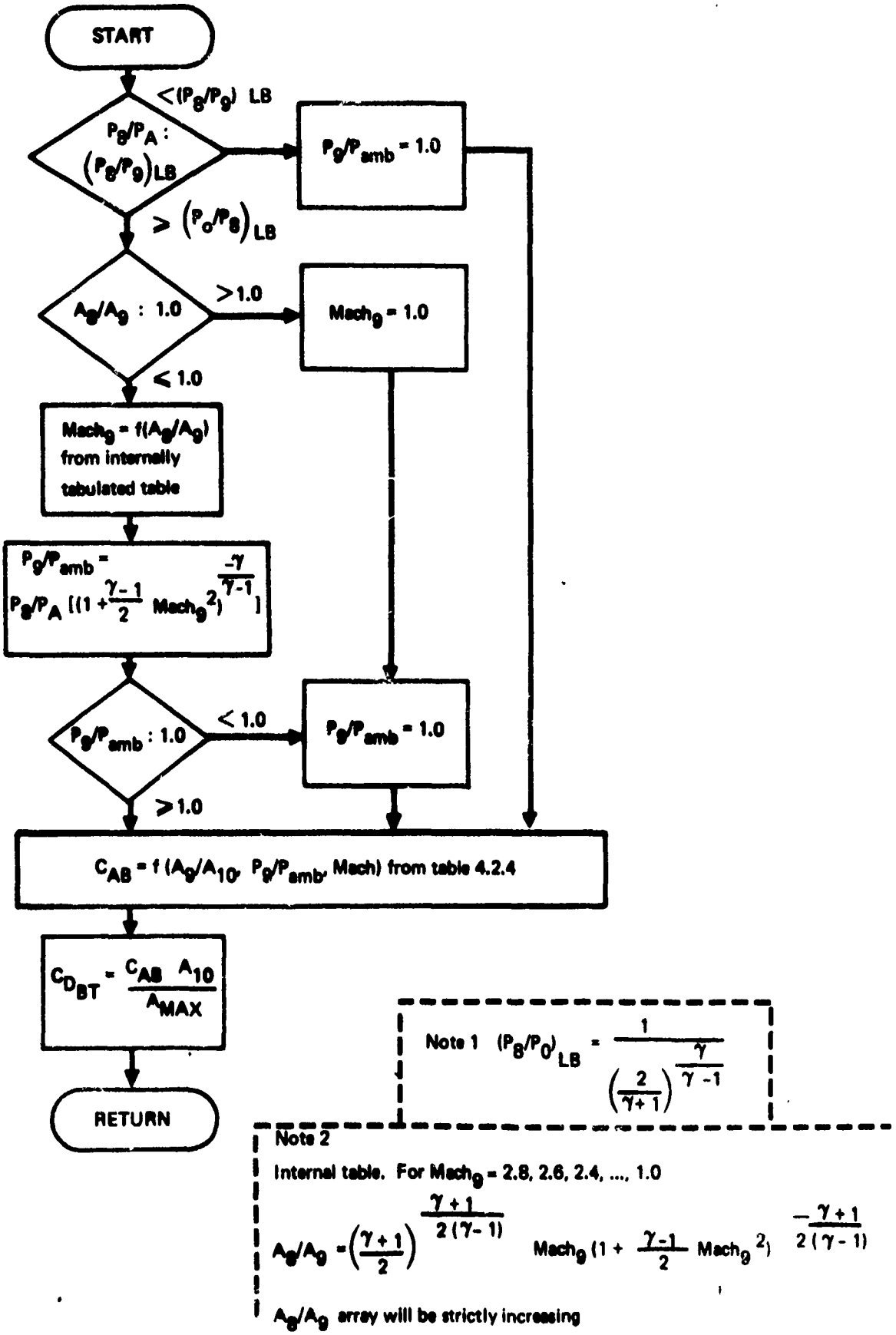


Figure 15: FLOW CHART FOR SUBROUTINE CABTAB

```
              SUBROUTINE CABTAB(CAB,AM)
000005        COMMON /XMACH9/XMACH9(10),A6A9T(10)
000035        COMMON/AREA10/AREA10,BYA10
000035        COMMON/AMAX/AMAX
000035        COMMON/ABINP/ AE6, A6, AE10, A10, P7Q0, P17Q0, FGIP, FGIF, PC,
             A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DHEP,
             B              DHEF, PLUG, FLMMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
             C              S, ABASE, BETAB, THETA
             D,             FUTURE(11), NVAR
000035        COMMON/P9PAL/P6PAL
000035        COMMON /ABTAB6/AB6X(10),AB6Y(10),AB6Z(10),AB6T(1000),NTX6,NTY6,
             A NTZ6
000035        EQUIVALENCE(P7Q0,P6PA)
000035        IF (P6PA-P6PAL) 25,50,50
000037     25 P9PA=1.0
000011        GO TO 100
000011     50 A6A9=A6/A9
000013        IF (A6A9-1.0) 70,60,60
000015     60 AM9=1.0
000017        GO TO 75
000017     70 AM9=TABU1(A6A9,A6A9T,XMACH9,10)
000023     75 P9PA=P6PA*(1.0+0.5*(GAMMAP-1.0)*AM9**2)** (-GAMMAP/(GAMMAP-1.0))
000037        IF (P9PA.LT.1.0) P9PA=1.0
000043    100 CAB=TABU3(A9*BYA10,P9PA,AM,AB6X,AB6Y,AB6Z,AB6T,NTX6,NTY6,NTZ6,
             A 10,10,10)
000064        CAB=CAB*AREA10/AMAX
000066        RETURN
000067        END
```

```
*FOLLOWING VARIABLES ARE EQUIVALENCED BUT NOT REFERENCED
P7Q0
```

```
SUBPROGRAM LENGTH
000102

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
25      -  000007    50      -  000011    60      -  000015    70      -  000017
75      -  000023    100     -  000043

EXTERNALS AND TAGS
TABU1   -  S00100    RBAREX.-   S00200    TABU3   -  S00300    END.    -  S00400

BLOCK NAMES AND LENGTHS
XMACH9  -  000024    AREA10  -  000002    AMAX    -  000001    ABINP   -  000051
P8PAL   -  000001    ABTA86  -  002011

VARIABLE ASSIGNMENTS
ABOT    -  000036C06 AB6X    -  000000C06 AB6Y    -  000012C06 AB6Z    -  000024C06
AMAX    -  000000C03 AM9     -  000101    AREA10  -  000000C02 A8      -  000001C04
ABA9    -  000100    ABA9T   -  000012C01 A9      -  000015C04 BYA10   -  000001C02
FUTURE  -  000035C04 GAMMAP  -  000011C04 NTX6    -  002006C06 NTY6    -  002007C06
NTZ6    -  002016C06 P8PA    -  000004C04 P8PAL   -  000000C05 P9PA    -  000077
XMACH9  -  000000C01

START OF CONSTANTS
000070

START OF TEMPORARIES
000073

START OF INDIRECTS
000077

SPACE REQUIRED TO COMPILE
035500
```

SUBJECT: SUBROUTINE DINTFR

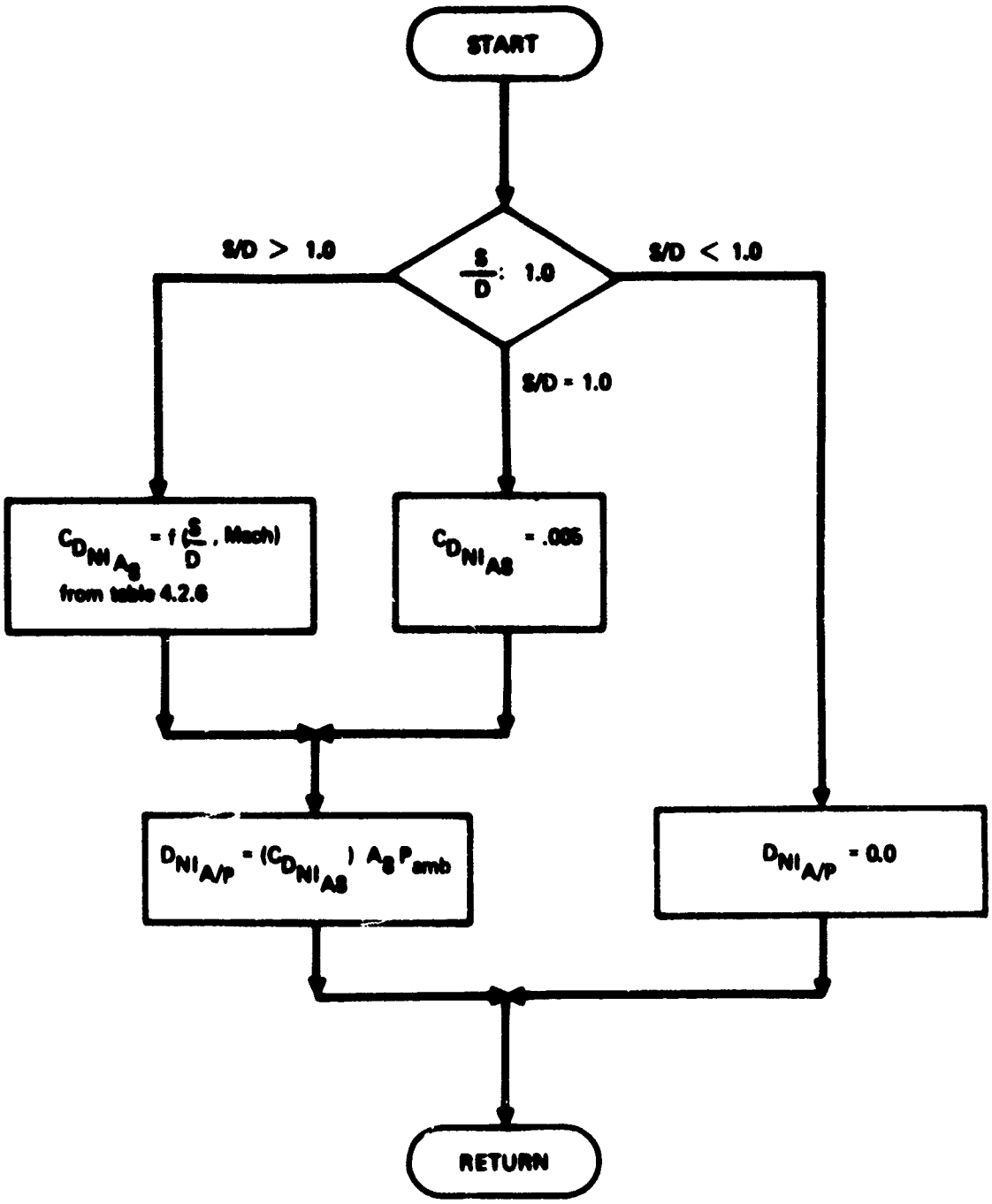
PURPOSE: SUBROUTINE DINTFR computes nozzle interference drag.

METHOD: The ratio S/D is interrogated to determine if it is greater than 1.0. If so, C_D is interpolated from Table 4.2.6 and total interference drag computed.

USAGE: CAL DINTFR

Input through COMMON/ABIN/ except D which is input in COMMON/DSPACG1. D is an output from the boattail routines except when $C_{D_{BT}}$ is interpolated from Table 4.2.1. In this latter case D ≡ 1.0, then nozzle interference drag can be input only as a function of Mach and some constant S/D ratio where the ratio is input instead of the nozzle centerline spacing for variable S.

SUBPROGRAMS: TABU2



Note: D computed by boattail drag routine

Figure 16: FLOW CHART FOR SUBROUTINE DINTFR

```
                SUBROUTINE DINTFR
000002          COMMON/ABDRAG/ DBTAIL, DNI, DBASE
000002          COMMON/ABTABS/ABTX5(10),ABTY5(10),ABTZ5(10,10),NTX5,NTY5
000002          COMMON/ABINP/ AE6, A6, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
                A               GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DMEP,
                B               DMEF, PLUG, FLMMIX, DMAX, DMBASE, ZLPLUG, ZLBTAL,
                C               S, ABASE, BTTA8, THETA
                D,              FUTURE(11), NVAR
000002          COMMON/DSPACE/DSPACE
000002          DATA DSPACE/1.0/
000002          SRATIO=S/DSPACE
000004      100 IF (SRATIO-1.0)  110,120,130
000007      110 DNI=0.0
000010          GO TO 150
000011      120 DDI=.005
000013          GO TO 140
000013      130 DDI=TABD2(SRATIO,XM,ABTX5,ABTY5,ABTZ5,    NTX5,NTY5,10,10)
000025      140 DNI=4.34*DDI*AB*PAMB
000031      150 RETURN
000032          END
```

```
SUBPROGRAM LENGTH
000041

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100     -  000004    110     -  000007    120     -  000011    130     -  000013
140     -  000025    150     -  000031

EXTERNALS AND TABS
TABU2   -  SC0100    END.    -  SC0200

BLOCK NAMES AND LENGTHS
ABDRAG  -  000003    ABTABS  -  000172    ABINP   -  000051    DSPACE  -  000001

VARIABLE ASSIGNMENTS
ABTX5   -  000000C02 ABTY5   -  000012C02 ABTZ5   -  000024C02 A8      -  003001C03
COI     -  000040    UNI     -  000001C01 DSPACE  -  000000C04 FUTURE  -  000035C03
NTX5    -  000170C02 NTY5    -  000171C02 PAM3    -  000014C03 S       -  000031C03
SRATIO  -  000037    XM      -  000013C03

START OF CONSTANTS
000033

START OF TEMPORARIES
000037

START OF INDIRECTS
000037

SPACE REQUIRED TO COMPILE
035300
```

SUBJECT: SUBRØUTINE DBASER

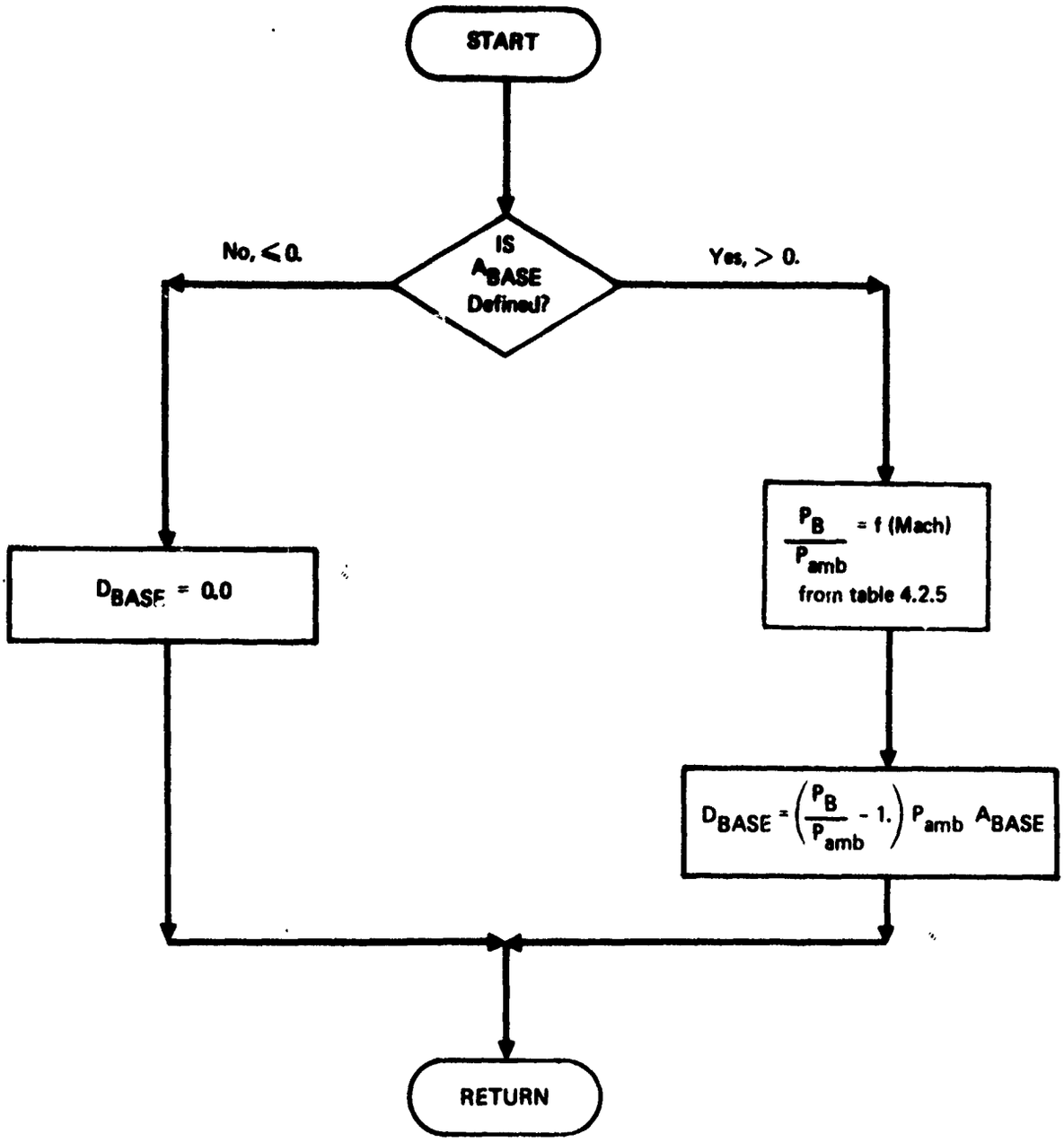
PURPØSE: SUBRØUTINE DBASER computes total airplane base drag.

METHØD: If a base area is input the base pressure ratio is interpolated from Table 4.2.5 and drag computed.

USAGE: CALL DBASER

Input form CØMMØN/ABIN/

SUBPRØGRAMS: TABU1



Note: A_{BASE} is input in field 8 of card 4.1.1

Figure 17: FLOW CHART FOR SUBROUTINE DBASER

```
                SUBROUTINE DBASER
000002          COMMON/ABTAB4/ABT4X(10),ABT4Y(10),NT4
000002          COMMON/ABINP/ AE8, A8, AE13, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DMEP,
               B              DMEF, PLUG, FLWMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C              S, ABASE, BTTA3, THETA
               D,             FUTURE(11), NVAR
000002          COMMON/ABDRAG/ DBTAIL, DNI, DBASE
000002          IF (ABASE) 10,10,20
000004       10 DBASE=0.0
000005          GO TO 30
000006       20 PBOPAM=TABU1(XM,ABT4X,ABT4Y, NT4)
000011          DBASE=(PBOPAM-1.0)*PAMB*ABASE
000015       30 RETURN
000016          END
```

```
SUBPROGRAM LENGTH
000021

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
10      -  000004    20      -  000036    30      -  000015

EXTERNALS AND TAGS
TAGU1   -  S00100    END.    -  S00200

BLOCK NAMES AND LENGTHS
ABTA34  -  000025    ABINP   -  000051    ABDRAG  -  000003

VARIABLE ASSIGNMENTS
ABASE   -  000032C02 ABT+X   -  000000C01 ABT+Y   -  000012C01 DBASE   -  000002C03
FUTURE  -  000035C02 NT+     -  000024C01 PAMB    -  000014C02 PBQPAM  -  000020
XM      -  000013C02

START OF CONSTANTS
000017

START OF TEMPORARIES
000020

START OF INDIRECTS
000020

SPACE REQUIRED TO COMPILE
035300
```

```
                SUBROUTINE DBTTB
000002          COMMON/AMAX/AMAX
000002          COMMON/ABINP/ AE8, A8, AE18, A18, P7QO, P17QO, FGIP, FGIF, PC,
               A              GAMMAP, GAMMAF, XM, PAMB, Q, A9, A19, CFG, DMEP,
               B              DMEF, PLUG, FLWMIX, DMAX, THICKB, ZLPLUG, ZLBTAL,
               C              S, ABASE, BTTAB, THETA
               D,             FUTURE(11), NVAR
000002          COMMON/ABDRAG/ DBTAIL, DNI, DBASE
000002          COMMON/ABTAB1/ABTX1(10),ABTY1(10),ABTZ1(10,10),NTX1,NTY1
000002          CDBETA=TABU2(XM,PC,ABTX1,ABTY1,ABTZ1,     NTX1,NTY1,10,10)
000014          DBTAIL=CDBETA*Q*AMAX
000016          RETURN
000017          END
```

```
SUBPROGRAM LENGTH
000022

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
TABU2  -  S00100    END.   -  S00200

BLOCK NAMES AND LENGTHS
AMAX   -  000001    ABINP  -  000051    ABORAG -  000003    ABTAB1 -  000172

VARIABLE ASSIGNMENTS
ABTX1  -  000000C04 ABTY1  -  000012C04 ABTZ1  -  000024C04 AMAX   -  000000C01
COBETA -  000021    DBTAIL -  000000C03 FUTURE -  000035C02 NTX1   -  000173C04
NTY1   -  000171C04 PC     -  000010C02 Q      -  000015C02 XM     -  000013C02

START OF CONSTANTS
000020

START OF TEMPORARIES
000021

START OF INDIRECTS
000021

SPACE REQUIRED TO COMPILE
035330
```

SUBJECT: SUBRØUTINE FIXDIM

PURPØSE: SUBRØUTINE FIXDIM determines the dimensions of the nozzle throat and nozzle exit inputs.

METHØD: The input flag is interrogated and three flags are returned indicating input type and dimension.

USAGE: CALL FIXDIM (FLAG, J, K, L)

FLAG - Input code value

J = 0 area input, = 1 diameter input

K = 0 inches, = 1 feet

L = 0 ignore, = 1 (A_9/A_8) input.

```
              SUBROUTINE FIXDIM(FLAG,J,K,L)
000037        IFLAG=IFIX(FLAG)
000013        GO TO (1,2,3,4,5),IFLAG
000020      1 J=0
000021        K=0
000021        L=0
000022        RETURN
000022      2 J=0
000023        K=1
000023        L=0
000024        RETURN
000025      3 J=1
000026        K=0
000026        L=0
000027        RETURN
000030      4 J=1
000031        K=1
000032        L=0
000033        RETURN
000033      5 L=1
000034        RETURN
000035        END
```

```
SUBPROGRAM LENGTH
000041

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      -  000020    2      -  000022    3      -  000025    4      -  000030
5      -  000033

EXTERNALS AND TAGS
ACSOER.-  SC0100    END.   -  S00200

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
IFLAG  -  000340

START OF CONSTANTS
000036

START OF TEMPORARIES
000036

START OF INDIRECTS
000040

SPACE REQUIRED TO COMPILE
035200
```

SUBJECT: SUBRØUTINE ATMØS

PURPOSE: SUBRØUTINE ATMØS computes ambient pressure and temperature using empirical equations for the 1962 U. S. Standard Atmosphere.

METHØD: Altitude is tested to determine which region it is in, then the appropriate equations are applied. If altitude exceeds 100,000, an error is assumed and altitude will be divided by ten until it is within the required range.

USAGE: CALL ATMØS (ALT, TAMB, PAMB)

ALT - pressure altitude

TAMB - ambient temperature

PAMB - ambient pressure

SUBPRØGRAMS: EXP WARNING

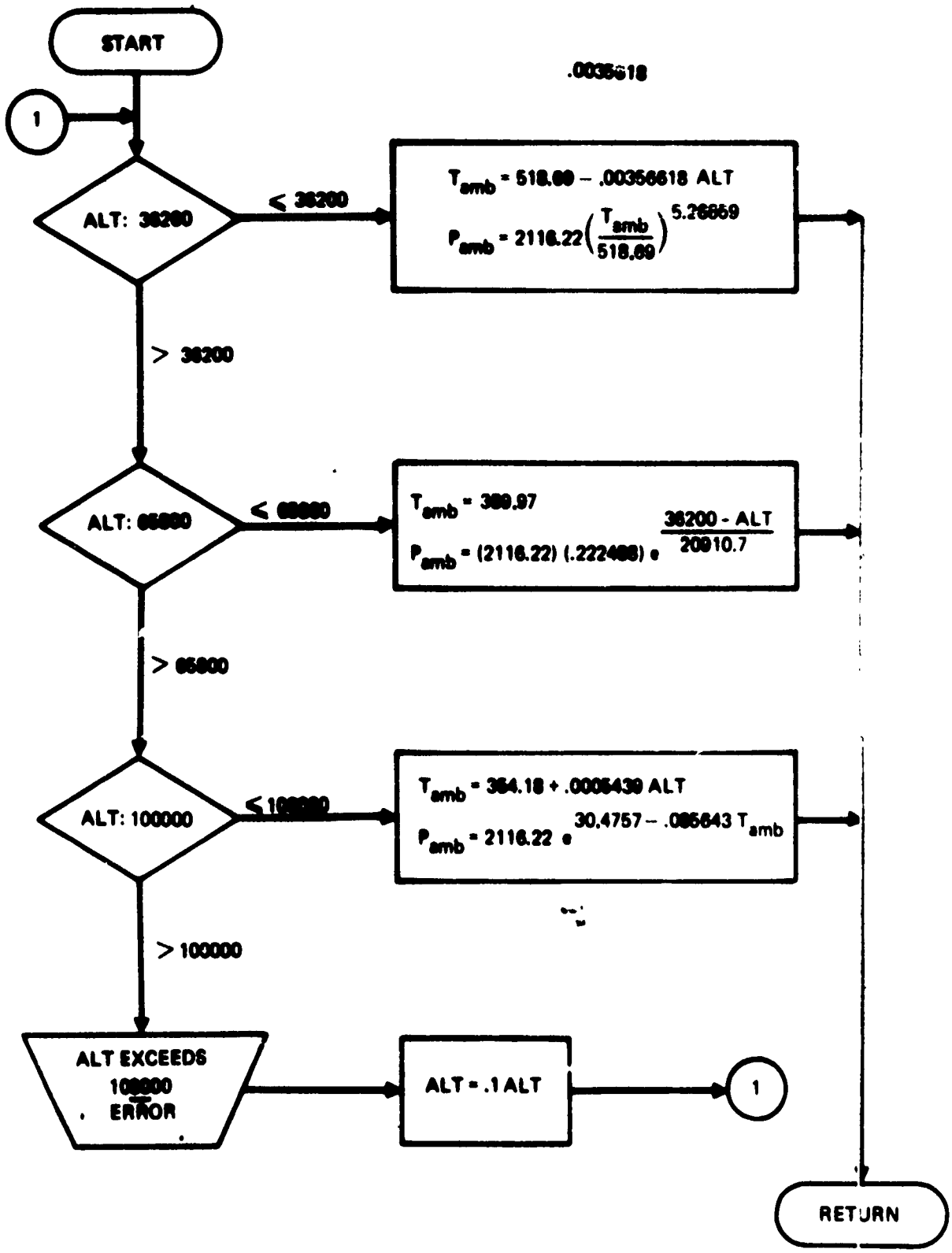


Figure 18: FLOW CHART FOR SUBROUTINE ATMOS

```
                SUBROUTINE ATMOS(ALT,TAMB,PAMB)
000006        5 IF (ALT.GT.36200.) GO TO 10                          COMPU150
000012          TAMB=518.69-.00356618*ALT                            COMPU160
000014          PAMB=2116.22*(TAMB/518.69)**5.26659                  COMPU170
000021          GO TO 40                                             COMPU180
000021       10 IF (ALT.GT.65800.) GO TO 20                          COMPU190
000025          TAMB=389.97                                          COMPU200
000025          PAMB=2116.22*.222468*EXP((36200.-ALT)/20913.7)       COMPU210
000034          GO TO 40                                             COMPU220
000035       20 IF (ALT.GT.100000.) GO TO 30                         COMPU230
000041          TAMB=354.18+.0005439*ALT                             COMPU240
000043          PAMB=2116.22*EXP(30.4757-.065643*TAMB)               COMPU250
000052          GO TO 40                                             COMPU260
000053       30 CALL WARNING
000054          ALT=ALT*.01
000057      700 FORMAT(1X,          *ALTITUDE INPUT EXCEEDS 100000 FT.  ALTITUDCOMPU290
                1E MODIFIED TO* F9.0)                                COMPU300
000057          GO TO5                                               COMPU310
000060       40 RETURN
000061          END
```

```
SUBPROGRAM LENGTH
000122

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
5       -  000000    10      -  000021    20      -  000035    30      -  000053
40      -  000063    700     -  000106

EXTERNALS AND TAGS
RBAREX.-  S00100    EXP     -  S00200    WARNING-  S00300    END.    -  S00400

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS

START OF CONSTANTS
000062

START OF TEMPORARIES
000117

START OF INDIRECTS
000122

SPACE REQUIRED TO COMPILE
035400
```

SUBJECT: SUBRØUTINE PERF

PURPØSE: SUBRØUTINE PERF controls the calculation of gross thrust.

METHØD: Mass flow, enthalpy, pressure, and relative pressure are developed for the nozzle throat. The thermodynamic properties at the throat are computed, then gross thrust computed from propulsive nozzle performance analysis.

USAGE: CALL PERF (WFT, WØ, PT8ØPA, TTØ, PAMB, FG, A8)

Input	WFT	-	Fuel flow lb/hr
	WØ	-	Air Flow lb/sec
	PT8ØPA	-	Nozzle throat pressure ratio
	TTØ	-	Total temperature degrees R
	PAMB	-	Ambient pressure PSF
Output	FG	-	Gross thrust lbs
	A8	-	Nozzle throat (given) ft^2

SUBPRØGRAMS: CHKRP GTHRST PRØFH

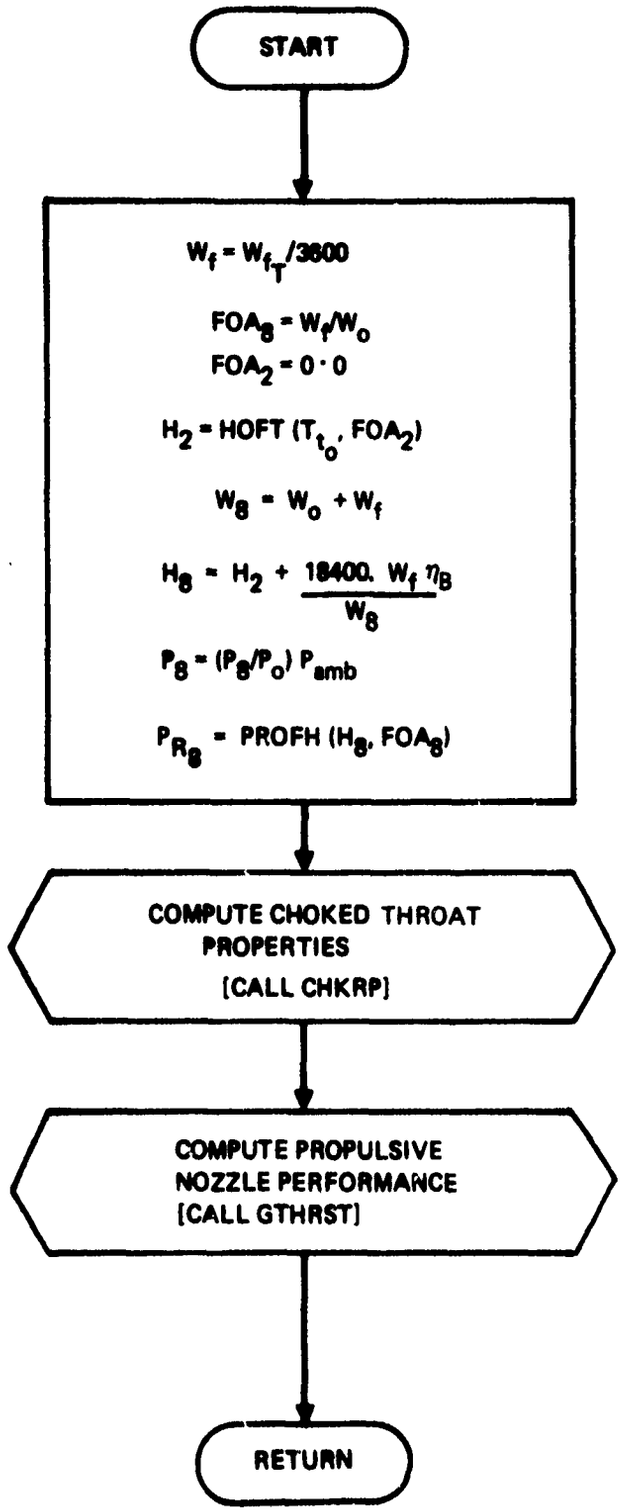


Figure 19: FLOW CHART FOR SUBROUTINE PERF

```
                SUBROUTINE PERF(WFT,WO,PT8OPA,TTO,PAMB,FG,A8)
000012          DATA XNB /.995/
000012          DATA XNOZZ /1.0/
000012          DATA FOA2 /0.0/
000012          DATA EHVF /18400./
000012          DATA CD /1.0/
000012          DATA CV /1.0/
000012          WF=WFT/3600.
000013          FOA8=WF/WO
000014          H2=HOFT(TTO,FOA2).
000023          W8=WO+WF
000025          H8=H2+EHVF*WF*XNB/W8
000031          P8=PT8OPA*PAMB
000032          PR8=PROFH(H8,FOA8)
000036          CALL CHKRP(P8,H8,PR8,FOA8,PAMB,T8S,C8S,P8S,P8OPAM,P8OP8S)
000052          CALL GTHRST(XNOZZ,PR8,P8,H8,W8,FOA8,PAMB,CV,CD,T8S,P8S,C8S,H9,T9S,
               A     W9,A8,FG)
000077          RETURN
000100          END
      *DECK COMPUTE
```

```
SUBPROGRAM LENGTH
000131

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
HOFT   -  S00100    PROFM  -  S00200    CHKRP  -  S00300    GTHRST -  S00400
END.   -  S00500

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
A8     -  000001    CO     -  000110    CV     -  000111    C8S    -  000122
EMVF   -  000107    FOA2   -  000106    FOA8   -  000113    H2     -  000114
H8     -  000116    H9     -  000126    PR8    -  000120    P8     -  000117
P8DPAM -  000124    P8DP8S -  000125    P8S    -  000123    T8S    -  000121
T9S    -  000127    V9     -  000130    WF     -  000112    W8     -  000115
XN8    -  000104    XNOZZ  -  000105

START OF CONSTANTS
000131

START OF TEMPORARIES
000132

START OF INDIRECTS
000134

SPACE REQUIRED TO COMPILE
035500
```

SUBJECT: SUBRØUTINE ØUTGØ

PURPØSE: SUBRØUTINE ØUTGØ prints a matrix of performance output.

METHØD: The output for NALT power settings at a Mach-altitude combination has been stored in the array ØUTP. A FØRMAT is built to print the variable name on both the left and right of the output columns. The matrix is printed and any indicated punching of MARK II data is done.

USAGE: CALL ØUTGØ (NALT)

NALT - Number of columns in output matrix = 10.

Input in CØMMØN

ØUTP (45,10)	output matrix
PCØDE	punched output flag
DA, TE	date, alphanumeric
TITLE (8)	run title
MACH	Mach number

Output

IWARN	Error message indicator introduced

SUBPRØGRAMS: MINØ MØD

```
                SUBROUTINE OUTGO(NALT)
000003          COMMON/RESULT/OUTP(45,10),LABLE(45),NVAR
000003          COMMON/PCODE/PCODE
000003          COMMON/DATE/DA,TE
000003          COMMON/TITLE/TITLE(8)
000003          COMMON/NEWM/NEWM                                        OUTGO 50
000003          COMMON/IWARN/IWARN
000003          COMMON/INS/ MACH,OLDALT,CON
000003          DIMENSION FMT(3)                                        OUTGO140
000003          REAL MACH
000003          DATA NVAR/41/
000003          DATA LABLE / 4HCASE, 3HALT, 2HPS, 3HFNA, 6HWFT RF, 4HSFCA, 4HFNRF,OUTGO 90
                14HFRAM, 2HRF, 5HMILRF, 6HDINLET, 5HCDSPL, 5HCDBLD, 5HCDBYP,
                26HCDINL , 4HDNOZ, 4HCDBT, 6HCDBASE, 5HCDINT, 8HDNOZ REF, 5HP8/P0,
                32HA9, 2HA8, 3HCFG, 4HBETA, 8HFN INPUT, 8HWF INPUT, 9HSFC INPUT,
                47HW INPUT, 5HW ADS, 8HFN/DELTA, 6HWF COR, 7HSFC COR, 4HTAMB,
                54HPA8, 3HTT0, 1HQ,         6HADE/AC, 6HADI/AC, 5HAD/AC, 6HSTATUS,
                6 4*1H   /
000003          DATA BLANK/1H /                                         OUTGO150
000003          KOUT=MIND(NALT,10)
000006          DO 10 I=1,3                                             OUTGO170
000010       10 FMT(I)=BLANK                                            OUTGO180
000014          WRITE(6,3) MACH,TITLE,DA,TE
000027          IWARN=0
000030        3 FORMAT(12H1MACH NUMBER,F7.4,5X,8A10,5X,A10,A2)
000030        6 FORMAT(1H )                                             OUTGO290
000030          WRITE (6,6)                                             OUTGO300
000034          IF (KOUT-10) 20,50,50                                   OUTGO310
C               BUILD FORMAT                                            OUTGO320
000037       20 ENCODE(30,1,FMT) KOUT                                   OUTGO330
000047        1 FORMAT(8H(1X,A10,I1,13HF10.3,5X,A10))                   OUTGO340
000047          DO 30 K=1,NVAR                                          OUTGO350
000052          WRITE (6,FMT)  LABLE(K),(OUTP(K,J),J=1,KOUT),LABLE(K)   OUTGO360
000073          IF (MOD(K,5).EQ.0) WRITE (6,6)                          OUTGO370
000105       30 CONTINUE                                                OUTGO380
000110          GO TO 65                                                OUTGO390
000110       50 DO 60 K=1,NVAR                                          OUTGO400
000112          WRITE (6,2)  LABLE(K),(OUTP(K,J),J=1,KOUT),LABLE(K)     OUTGO410
000132          IF (MOD(K,5).EQ.0) WRITE (6,6)                          OUTGO420
000144       60 CONTINUE                                                OUTGO430
000147        2 FORMAT(1X,A10,10F10.3,5X,A10)                           OUTGO440
000147       65 IF (PCODE) 80,80,70                                     OUTGO450
C               OUTPUT FOR PUNCH OR PLOTTING PROGRAM                    OUTGO460
000151       70 IF (NEWM.GT.0) WRITE(7,150) (OUTP(3,J),J=1,KOUT)        OUTGO470
000167          IF (PCODE-1.0) 74,74,76                                 OUTGO480
000172       74 WRITE (7,151) (OUTP(4,J),J=1,KOUT)                      OUTGO490
000206          WRITE (7,151) (OUTP(5,J),J=1,KOUT)                      OUTGO500
000222          GO TO 80                                                OUTGO510
000224       76 WRITE (7,151) (OUTP( 4,J),J=1,KOUT)
000240          WRITE (7,151) (OUTP( 5,J),J=1,KOUT)
000254          WRITE (7,151) (OUTP(29,J),J=1,KOUT)                     OUTGO530
000270       80 RETURN                                                  OUTGO540
```

```
000271      150 FORMAT(10F7.2)                                OUTG0550
000271      151 FORMAT(10F7.0)                                OUTG0560
000271          END                                           OUTG0570
```

```
SUBPROGRAM LENGTH
000337

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      -  000305   2      -  000312   3      -  000275   6      -  000303
10     -  000011   20     -  000037   30     -  000105   50     -  000110
60     -  000144   65     -  000147   70     -  000151   74     -  000172
76     -  000224   80     -  000270   150    -  000320   151    -  000322

EXTERNALS AND TAGS
OUTPTC.-  S00100   OUTPTS.-  S00200   END.   -  S00300

BLOCK NAMES AND LENGTHS
RESULT -  000750   PCODE  -  000001   DATE   -  000002   TITLE  -  000010
NEWM   -  000001   IWARN  -  000001   INS    -  000003

VARIABLE ASSIGNMENTS
BLANK  -  000332   DA     -  000000C03 FMT    -  000327   I      -  000334
IWARN  -  000000C06 J     -  000336   K      -  000335   KOUT   -  000333
LABLE  -  000762C01 MASH  -  000000C07 NEWM   -  000000C05 NVAR   -  000757C01
OUTP   -  000000C01 PCODE -  000000C02 TE     -  000001C03 TITLE  -  000000C04

START OF CONSTANTS
000272

START OF TEMPORARIES
000323

START OF INDIRECTS
000327

SPACE REQUIRED TO COMPILE
036600
```

SUBJECT: SUBRØUTINE WARNING

PURPØSE: SUBRØUTINE WARNING identifies the point to which the warning which follows applies.

METHØD: If no warning message has been written since the last printing of an output matrix, a heading is printed. In any event, the case number, Mach, altitude, and power setting is printed. The calling subprogram will then print the appropriate message.

USAGE: CALL WARNING

Input in CØMMØN

IWARN	Error message flag
CASE	Case number
ALT	Altitude
MACH	Mach number
PS	Power Setting

Output: STATUS Error flag for case.

```
                SUBROUTINE WARNING
000002          COMMON/IWARN/IWARN
000002          COMMON/OUTS/CASE,ALT,PS,OUTS(37),STATUS,EXTRA(4)
000002          COMMON/INS/ MACH,INS(2)
000002          REAL MACH
000002          DATA IWARN/0/
000002          IF (IWARN.EQ.0) WRITE(6,1)
000007        1 FORMAT(1H1,49X,24H*** WARNING MESSAGES *** )
000007          IWARN=1
000010          WRITE(6,2) CASE,MACH,ALT,PS
000024        2 FORMAT(8HCASE NO,F5.0,10X,4HMACH,F6.3,10X,3HALT,F7.0,10X,2HPS,F7.
               A2)
000024          STATUS=-99.999
000026          RETURN
000026          END
            *DECK STORE
```

```
SUBPROGRAM LENGTH
000046

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      -  000031    2      -  000036

EXTERNALS AND TAGS
OUTPTC.-  S00100    END.   -  S00200

BLOCK NAMES AND LENGTHS
IWARN  -  000001    OUTS   -  000055    INS    -  000003

VARIABLE ASSIGNMENTS
ALT    -  000031C02 CASE   -  000000C02 EXTRA  -  000051C02 INS    -  000001C03
IWARN  -  000000C01 MASH   -  000000C03 OUTS   -  000003C02 PS     -  000002C02
STATUS -  000005C02

START OF CONSTANTS
000027

START OF TEMPORARIES
000046

START OF INDIRECTS
000046

SPACE REQUIRED TO COMPILE
035300
```

SUBJECT: SUBRØUTINE STØRE

PURPØSE: SUBRØUTINE STØRE moves the results for a single case to the output matrix.

METHØD: The CØMMØN block ØUTS contains the output for the case just computed with the individual variables in exactly the order to be printed. This data is transferred to CØMMØN block ØUTP.

USAGE: CALL STØRE (NALT)

NALT	-	number of cases at current Mach altitude = 10
ØUTS (45)	-	output for NALTTH case.
NVAR	-	number of ØUTS locations being used.

Output OUTP (RT,10) Output matrix.

```
                SUBROUTINE STORE(NALT)
000003          COMMON/INS/ MACH,DLDALT,CON
000003          COMMON/OUTS/OUTS(45)
000003          COMMON/RESULT/OUTP(45,10),LABLE(45),NVAR
000003          DO 10 K=1,NVAR                                  STORE 60
000005       10 OUTP(K,NALT)=OUTS(K)                            STORE 70
000015          RETURN                                          STORE 80
000015          END                                             STORE 90
```

```
SUBPROGRAM LENGTH
000020

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
10      -  000005

EXTERNALS AND TAGS
END.    -  500100

BLOCK NAMES AND LENGTHS
INS     -  000003    OUTS    -  000055    RESULT -  000760

VARIABLE ASSIGNMENTS
K       -  000017    LABLE   -  000702C03 NVAR   -  000757C03 OUTP   -  000000C03
OUTS    -  000000C02

START OF CONSTANTS
000016

START OF TEMPORARIES
000016

START OF INDIRECTS
000017

SPACE REQUIRED TO COMPILE
035200
```

SUBJECT: SUBRØUTINE CHKRP

PURPØSE: SUBRØUTINE CHKRP calculates the thermodynamic properties at the throat and the total/static pressure ratio at the sonic condition.

METHØD: FUNCTIØN CØFH is called to compute sonic velocity. Then enthalpy is computed using the entry total enthalpy and the source velocity. Critical static temperature and relative pressure are computed as functions of enthalpy and fuel-air ratio. Pressure is derived from entry pressure and the relative pressure.

USAGE: CALL CHKRP (P1, H1, PR1, FØA1, PAM, TC, C, PC, P1OPAM, PIPCC)

P1	Entry total pressure PSF
H1	Entry total enthalpy, BTU/lb_M
PR1	Entry relative pressure
FØA1	Entry fuel-air ratio
PAM	Ambient static pressure, PSF
TC	Critical static temperature, degrees R
C	Critical sonic velocity ft/sec
PC	Critical static pressure, PSF
P1OPAM	P1/PAM
PIPCC	P1/PC

SUBPRØGRAMS: CØFH PRØFH TØFH

```
                SUBROUTINE CHKRP(P1,H1,PR1,FOA1,PAM,TC,C,PC,P1OPAM,P1PCC)        CHKRP
          C                                                                      CHKRP
          C**   THIS SUBROUTINE CALCULATES THERMODYNAMIC PROPERTIES AT THE THROAT CHKRP
          C**    OF A CHOKED NOZZLE, AND THE TOTAL-TO-STATIC PRESSURE RATIO AT   CHKRP
          C**    THE SONIC CONDITION.                                            CHKRP
          C                                                                      CHKRP
          C**    THE NUMBER 1 IN A FORTRAN NAME REFERS TO THE NOZZLE INLET SIDE  CHKRP
          C                                                                      CHKRP
          C**    P = TOTAL PRESSURE, PSF                                         CHKRP
          C**    H1 = ENTRY TOTAL ENTHALPY, BTU/LBM                              CHKRP
          C**    PR = RELATIVE PRESSURE                                          CHKRP
          C**    FOA = FUEL/AIR RATIO                                            CHKRP
          C**    PAM = AMBIENT STATIC PRESSURE, PSF                              CHKRP
          C**    TC  = CRITICAL STATIC TEMPERATURE, DEGREES R.                   CHKRP
          C**    C = CRITICAL SONIC VELOCITY,FT/SEC                              CHKRP
          C**    PC = CRITICAL STATIC PRESSURE, PSF                              CHKRP
          C**    P1OPAM = P1/PAM                                                 CHKRP
          C**    P1PCC = P1/PC                                                   CHKRP
          C                                                                      CHKRP
          C                                                                      CHKRP
          C**    COFH IS A FUNCTION SUBROUTINE WHICH RETURNS CRITICAL SONIC      CHKRP
          C**     VELOCITY AS A FUNCTION OF TOTAL ENTHALPY AND FUEL/AIR RATIO    CHKRP
          C                                                                      CHKRP
 000015          C= COFH(H1,FOA1)                                                CHKRP
 000022          HC=  H1- C*C*.199708E-4                                         CHKRP
          C                                                                      CHKRP
          C**    .199708E-4 = 1/(2)(G SUB 0)(J)                                  CHKRP
          C                                                                      CHKRP
 000025          TC=TOFH(HC,FOA1)                                                CHKRP
 000033          PRC=  PROFH(HC,FOA1)                                            CHKRP
 000041          P1PCC=  PR1/PRC                                                 CHKRP
 000042          PC=     P1/P1PCC                                                CHKRP
 000044          P1OPAM= P1/PAM                                                  CHKRP
 000046          RETURN                                                          CHKRP
 000047          END                                                             CHKRP
```

RUN VERSION JUL 71 22.46.30. 72/08/23.

SUBPROGRAM LENGTH
000035

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
COFM - S00100 TOF4 - S00200 PROFM - S00300 END. - S00400

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
C - 000000 MC - 000053 PC - 000001 PRC - 000054
P1DPAM - 000002 P1PCC - 000003

START OF CONSTANTS
000050

START OF TEMPORARIES
000051

START OF INDIRECTS
000053

SPACE REQUIRED TO COMPILE
035330

SUBJECT: SUBRØUTINE GTHRST

PURPØSE: SUBRØUTINE GTHRST computes the performance of propulsive nozzles.

METHØD: Outlet conditions are computed, then nozzle ideal exit velocity. Nozzle throat static pressure is compared to ambient pressure to determine which temperature, pressure, and velocity to use in the calculation of nozzle throat area. Gross thrust is computed by the appropriate equation.

USAGE: CALL GTHRST (XNØZZ, PR1, P1, H1, W1, FØA1, PAM, CV, CD, TCS, PCS, VC, H2, T2S, V2, A, FG)

	XNØZZ	- 1.0 for convergent-divergent nozzle; 2.0 for convergent nozzle.
	P121	Nozzle inlet relative pressure
	P1	Nozzle inlet total pressure PSF
	H1	Nozzle inlet total enthalpy BTU/LB_M
	W1	Nozzle inlet mass flow LB/SEC
	FØA1	Nozzle inlet fuel-air ratio
	PAM	Ambient static pressure PSF
	CV	Nozzle velocity coefficient
	CD	Discharge coefficient
	TCS	Nozzle throat static temperature degrees R
	PCS	Nozzle throat static pressure PSF
	VC	Nozzle throat velocity ft/sec
Output	H2	Nozzle outlet static enthalpy BTH/LB_M
	T2S	Nozzle outlet static temperature degrees R

Output (continued)

V2	Nozzle ideal exit velocity ft/sec
A	Nozzle throat area ft^2
FG	Gross thrust lb

SUBPRØGRAMS: AREA HØFPR SQRT

TØFH

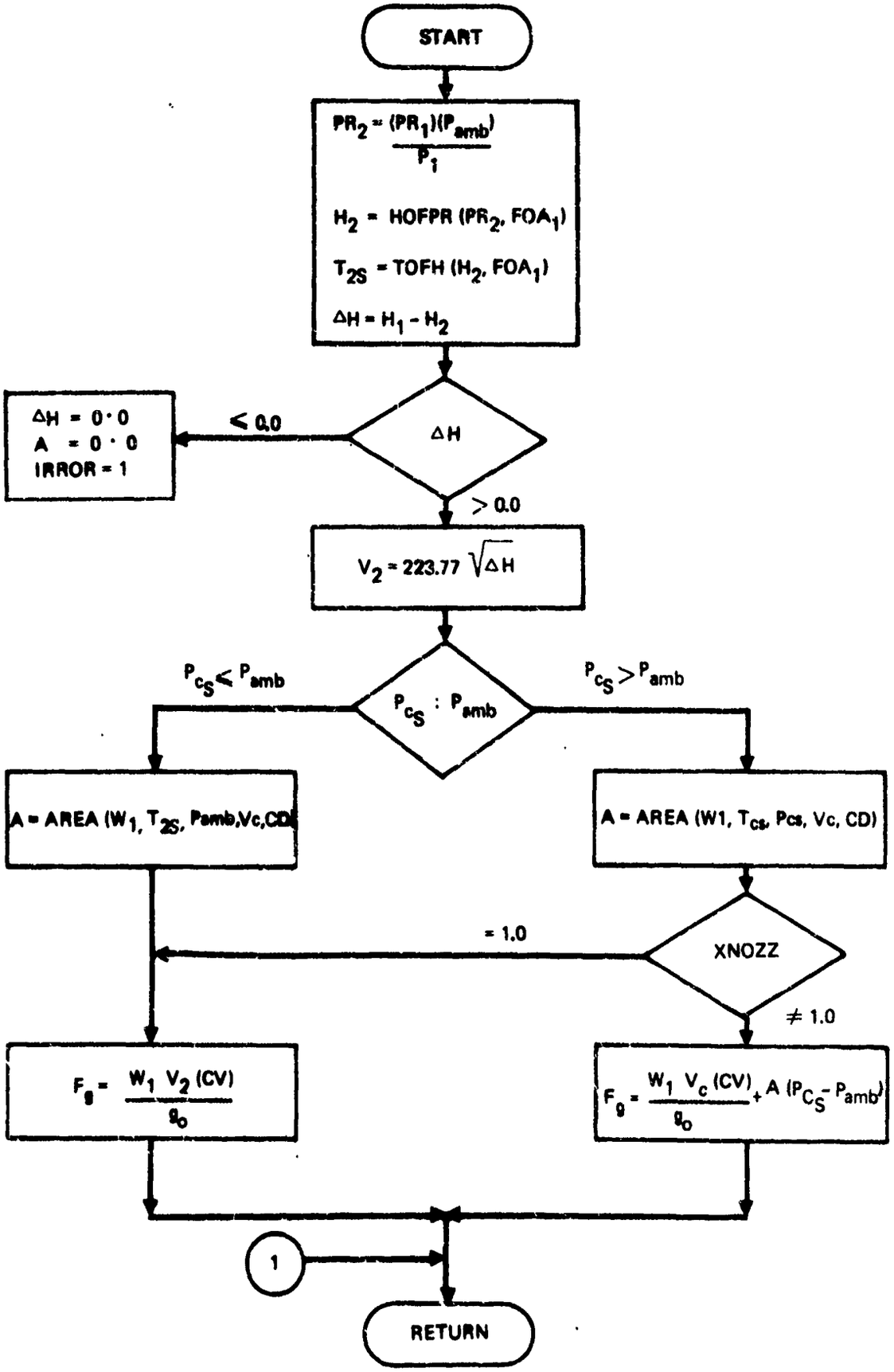


Figure 20: FLOW CHART FOR SUBROUTINE GTHRST

```
      SUBROUTINE GTHRST(XNOZZ,PR1,P1,H1,W1,FOA1,PAM,CV,CD,          KR03
     .TCS,PCS,VC,H2,T2S,V2,A,FG)                                    GTHRST
C                                                                   GTHRST
C**   AUTHOR          CAFFREY, W.              1969                 GTHRST
C                                                                   GTHRST
C**   MODIFIED        RITZAU, K.               FEBRUARY 1970        GTHRST
C                                                                   GTHRST
C**   INPUTS                                                        GTHRST
C      ICHOKE    = 1 FOR NOZZLE UNCHOKED , = 2 FOR NOZZLE CHOKED    GTHRST
C      INOZZ     = 1 CONVERGENT-DIVERGENT NOZZLE, = 2 CONVERGENT NOZZLE GTHRST
C      PR1       = NOZZLE INLET RELATIVE PRESSURE                   GTHRST
C      PAM       = AMBIENT STATIC PRESSURE, PSF                     GTHRST
C      P1        = NOZZLE INLET TOTAL PRESSURE, PSF                 GTHRST
C      FOA1      = NOZZLE INLET FUEL TO AIR RATIO                   GTHRST
C      H1        = NOZZLE INLET TOTAL ENTHALPY, BTU/LBM             GTHRST
C      W1        = NOZZLE INLET MASS FLOW, LBM/SEC                  GTHRST
C      CV        = NOZZLE VELOCITY COEFFICIENT                      GTHRST
C      AC        = NOZZLE THROAT AREA, SQUARE FEET                  GTHRST
C      PCS       = NOZZLE THROAT STATIC PRESSURE, PSF               GTHRST
C      VC        = NOZZLE THROAT VELOCITY, FT/SECOND                GTHRST
C      TCS       = NOZZLE THROAT STATIC TEMPERATURE, DEG. R.        GTHRST
C                                                                   GTHRST
C**   OUTPUTS                                                       GTHRST
C      H2        = NOZZLE OUTLET STATIC ENTHALPY, BTU/LBM           GTHRST
C      T2S       = NOZZLE OUTLET STATIC TEMPERATURE, DEG. R.        GTHRST
C      V2        = NOZZLE IDEAL EXIT VELOCITY, FT/SEC               GTHRST
C      FG        = GROSS THRUST, LBF.                               GTHRST
C      A         = NOZZLE THROAT AREA, SQUARE FEET.                 GTHRST
C                                                                   GTHRST
C**   THIS SUBROUTINE CALCULATES PERFORMANCE FOR THE PRIMARY AND    GTHRST
C     SECONDARY PROPULSIVE NOZZLES.                                 GTHRST
C                                                                   GTHRST
      COMMON / ERMES/  IRROR(5)                                     ERMES
C                                                                   KR05
      PR2  =  PR1 * PAM / P1                                        GTHRST
      H2   =  HOFPR(PR2,FOA1)                                       GTHRST
      T2S  =  TOFH(H2,FOA1)                                         GTHRST
      DELH = H1 - H2                                                GTHRST
      IF(DELH .GT. 0.) GO TO 3                                      GTHRST
      DELH = 0.                                                     GTHRST
      IRROR(5) = 1                                                  GTHRST
      A = 0.                                                        KR05
      RETURN                                                        KR01
    3 V2  =  223.77 * SQRT(DELH)                                    KR04
C                          223.77  = ((2)(G SUB 0)(J))**.5          GTHRST
C                          .031081 = 1. /(G SUB C)                  GTHRST
      IF (PCS.GT.PAM) GO TO 10                                      KR03
      A=AREA(W1,T2S,PAM,V2,CD)                                      KR03
    5 FG  =  .031081 * W1 * V2 * CV                                 KR04
      RETURN                                                        KR03
   10 A=AREA(W1,TCS,PCS,VC,CD)                                      KR03
      IF (XNOZZ.EQ.1.) GO TO 5                                      KR03
```

```
CC0117      FG=.031081*CV*W1*VC+A*(PCS-PAM)                    KR03
CC0130      RETURN                                             GTHRST
CC0131      END                                                GTHRST
```

```
SUBPROGRAM LENGTH
000142

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
3       -  000053    5       -  000077    10      -  000104

EXTERNALS AND TAGS
HOFPR   -  S00100    TOFH    -  S00200    SQRT    -  S00300    AREA    -  S00400
END.    -  S00500

BLOCK NAMES AND LENGTHS
ERRCS   -  000005

VARIABLE ASSIGNMENTS
A       -  000011    CD      -  000002    CV      -  000001    DELH    -  000141
FG      -  000012    H2      -  000006    IRRDR   -  00000001 PAM     -  000000
PCS     -  000004    PR2     -  000140    TCS     -  000003    T2S     -  000007
VC      -  000005    V2      -  000010

START OF CONSTANTS
000132

START OF TEMPORARIES
000135

START OF INDIRECTS
000140

SPACE REQUIRED TO COMPILE
035600
```

SUBJECT: SUBRØUTINE DATA

PURPØSE: SUBRØUTINE DATA provides a card image output of all data cards input to TEM-333.

METHØD: Data cards are read from TAPE1 (INPUT), written on TAPE5 and OUTPUT. A card count is provided. Upon sensing an end-of-file of TAPE1, TAPE5 is end-filed and rewound. All subsequent input is from TAPE5. The INPUT file is equivalent to TAPE1 to provide a logical number for the EØF test.

USAGE: CALL DATA

INPUT file must be in desired position when subroutine is called.

```
                SUBROUTINE DATA                                          DATA  10
          C  THIS DECK HAS BEEN MODIFIED TO TERMINATE UPON AN END OF FILE NOT END DATA  20
000002          DIMENSION IWORD(8)                                       DATA  30
000002          REWIND 5                                                 DATA  40
000004          WRITE (5,1000)                                           DATA  50
000010          DO 100 I=1,100000                                        DATA  60
000012          READ (1,1001) IWORD                                      DATA  70
          C     IF (EOF(1))200,50                                         FTN
000017          IF (EOF,1) 200,50                                         RUN
000022       50 IF (AMOD(FLOAT(I),50.).NE.0.) GO TO 75                   DATA  90
000027          WRITE (5,1000)                                           DATA 100
000033       75 WRITE (5,1002) I,IWORD                                   DATA 110
000043          WRITE (5,1001)IWORD                                      DATA 120
000051      100 CONTINUE                                                 DATA 130
000054      200 END FILE 5                                               DATA 140
000056          REWIND 5                                                 DATA 150
000060          RETURN                                                   DATA 160
000061     1000 FORMAT (1H1*CARD*13X*1*9X*2*9X*3*9X*4*9X*5*9X*6*9X*7*9X*8*/1H  DATA 170
             $        8X*12345678901234567890123456789012345678901234567890*  DATA 180
             $          *12345678901234567890123456789012345678901234567890*/1H )  DATA 190
000061     1001 FORMAT (A3,7A10,A7)                                     DATA 200
000001     1002 FORMAT (1H I4,4X,A3,7A10,A7)                            DATA 210
000051          END                                                      DATA 220
```

```
SUBPROGRAM LENGTH
000135

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
50    -  000022    75    -  000033    200   -  000054    1000  -  000071
1001  -  000114    1002  -  000117

EXTERNALS AND TAGS
REDINM.-  000100    OUTPTC.-  000200    INPUTC.-  000300    IFENDF.-  000400
ENDFIL.-  000500    END.   -  000600

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
I     -  000134    INDRD  -  000123

START OF CONSTANTS
000062

START OF TEMPORARIES
000122

START OF INDIRECTS
000123

SPACE REQUIRED TO COMPILE
035400
```

SUBJECT: SUBRØUTINE PLACIN

PURPØSE: SUBRØUTINE PLACIN reads a tabular array. A count of points is provided for independent variables.

METHØD: If the number of fields input, NF, is equal to zero, then data is read until the maximum allowable number of cards has been read or until a 1 is input in column 72. A field by field count is made on the last card image so that the total number of non-blank fields is returned in NF. If NF is positive, then NF values are read. All arrays are formatted ten field-seven digit.

USAGE: CALL PLACIN (A(1),NF,NCARD)

A - array to be filled

NF - number of fields to read or if zero on input, the number of fields read.

NCARD - the maximum number of cards to be read if array is full.

SUBPRØGRAM: KØUNT

```
                SUBROUTINE PLACIN(A,NF,NCARD)
000006          COMMON/TAPENO/IN,IOUT
000006          COMMON/CARD/CARD(8)
000006          COMMON/IPTAB/IPTAB
000006          DIMENSION A(1)
000006          DATA IPTAB/1/
000006          IF (NF) 10,20,10
          C     READ DEPENDENT ARRAY - COUNT PROVIDED
000007       10 IK=0
000010          NCARDS=(NF-1)/10+1
000014          GO TO 30
          C     READ INDEPENDENT ARRAY - COUNT NO. INPUT FIELDS
000015       20 IK=1
000016          NCARDS=NCARD
000017       30 LO=1
000020          LH=10
000021          DO 50 I=1,NCARDS
000023          READ (IN,2) (A(I),I=LO,LH), II,ID
000042          IF (IPTAB.NE.0) WRITE(IOUT,3)(A(I),I=LO,LH),II,ID
000067          IF (IK) 35,40,35
000076       35 IF (NCARDS.EQ.1) GO TO 60
000072          IF (II) 40,60,40
000073       40 LO=LO+10
000075       50 LH=LH+10
000100          IF (IK.EQ.0) GO TO 70
000101          WRITE(IOUT,4) NCARDS
000107          IPTAB=1
000110          LH=LH-10
000112       60 NF=LH-10+KOUNT(A(LO),10)
000123       70 RETURN
000124        1 FORMAT (8A10)
000124        2 FORMAT(10F7.0,I2,A8)
000124        3 FORMAT(1X,10F12.5,I2,A8)
000124        4 FORMAT(40H0XXXXX THE MAXIMUM NUMBER OF DATA CARDS,,I2,76H, FOR THE
                1ABOVE ARRAY HAS BEEN READ WITHOUT ENCOUNTERING AN ARRAY TERMINATOR
                2.//7X11HCHECK TO SEE IF MISSING 1 IN COL 72 HAS CAUSED INPUT ERROR
                3.  PROGRAM WILL CONTINUE TO READ INPUTS, BUT CARD SEQUENCE/7X20HMA
                4Y NOW BE IN ERROR.//)
000124          END
```

```
SUBPROGRAM LENGTH
000236

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1      -  000127   2      -  000131   3      -  000134   4      -  000137
10     -  000007   20     -  000015   30     -  000017   35     -  000070
40     -  000073   60     -  000112   70     -  000123

EXTERNALS AND TAGS
INPUTC.-  S00100   OUTPTC.-  S00200   KOUNT  -  S00300   END.   -  S00400

BLOCK NAMES AND LENGTHS
TAPENO -  000002   CARD   -  000010   IPTAB  -  000001

VARIABLE ASSIGNMENTS
CARD   -  00000002 I      -  000203   ID     -  000205   II     -  000204
IK     -  000177   IN     -  00000001 IOUT   -  00001001 IPTAB  -  00000003
LH     -  000202   LO     -  000201   NCARDS -  000200

START OF CONSTANTS
000125

START OF TEMPORARIES
000174

START OF INDIRECTS
000177

SPACE REQUIRED TO COMPILE
035730
```

SUBJECT: SUBRØUTINE TABL1

PURPØSE: SUBRØUTINE TABL1 inputs tables of the form F = f(x), table format 1.

METHØD: SUBRØUTINE PLACIN is called to read and count the independent array, then recalled to read the dependent array.

USAGE: CALL TABL1(T(1), MAX, N)

T - Table to be read

MAX - Maximum number of points in X array.

Output - N - Actual number of input points in X array.

STØRAGE: The table T is stored as follows:

Address	$T_{(1)}$	$(T_{(2)}$	$T_{(Max)}$	$T_{(Max+1)}$	$T_{(2\ Max)}$
Content	$X_{(1)}$	$X_{(N)}$		$F_{(1)}$ ...	$F_{(N)}$

N MAX

T can be specified in the calling program by either:

DIMENSION T (2MAX) or CØMMØN/T/ TX(MAX),TY(MAX)

SUBPRØGRAMS: PLACIN

```
                SUBROUTINE TABL1(T,MAX,N)
000036          DIMENSION T(1)
000036          N=0
000036          NCARD=(MAX-1)/10+1
000012          CALL PLACIN(T(1),N,NCARD)
000015          CALL PLACIN(T(MAX+1),N,NCARD)
000022          RETURN
000023          END
```

```
SUBPROGRAM LENGTH
000026

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
PLACIN -  000100    END.   -  000200

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
NCARD  -  000025

START OF CONSTANTS
000024

START OF TEMPORARIES
000024

START OF INDIRECTS
000025

SPACE REQUIRED TO COMPILE
035200
```

SUBJECT: SUBRØUTINE TABL2

PURPØSE: SUBRØUTINE TABL2 inputs tables of the form F = f(X,Y) square matrix, table format 2Q.

METHØD: SUBRØUTINE PLACIN is called to read and count the Y array, then again to read and count the X array. Then for each input Y point PLACIN is called to read NX points of the dependent array.

USAGE: CALL TABLE 2 (T, MAXX, MAXY, NX, NY)

	T	- -	Table to be read
	MAXX	-	Maximum number of points in X array
	MAXY	-	Maximum number of points in Y array
Output	NX	-	Actual number of X array points input
Output	NY	-	Actual number of Y array points input.

STØRAGE: The table T is stored as follows:

Address	$T_{(1)}$ - - - $T_{(MAXX)}$ $T_{(MAXX+1)}$ - - $T_{(MAXX+MAXY)}$
Content	$X_{(1)}$ - - $X_{(NX)}$ $Y_{(1)}$ - - $Y_{(NY)}$
Address	$T_{(MAXX+MAXY+1)}$ - - $T_{(MAXX+MAXY+NX)}$ - - $T_{(12MAXX+MAXY+1)}$
Content	$F_{(1,1)}$ - - - $F_{(NX,1)}$ - - - - $F_{(1,2)}$
Address	$T_{((NY+1)\ MAXX+MAXY+1)}$ $T_{((NY+1)MAXX+MAXY+NX)}$ - - $T_{((MAXY+1)MAXX+MAXY)}$
Content	$F_{(1,NY)}$ - - - - - - - - - - $F_{(NX,NY)}$

T can be specified in the calling program by either:

DIMENSIØN T(MAXX+MAXY+(MAXX)(MAXY)) or

CØMMØN/T/TX(MAXX),TY(MAXY), TF(MAXX, MAXY)

SUBPRØGRAM: PLACIN

```
                SUBROUTINE TABL2(T,MAXX,MAXY,NX,NY)
000010          DIMENSION T(1)
000010          NY=0
000016          NCARD=(MAXY-1)/10+1
000014          CALL PLACIN(T(MAXX+1),NY,NCARD)
000017          NX=0
000023          NCARD=(MAXX-1)/10+1
000027          CALL PLACIN(T(1),NX,NCARD)
000031          ID=MAXX+MAXY+1
000036          DO 10 I=1,NY
000040          CALL PLACIN(T(ID),NX,NCARD)
000043       10 ID=ID+MAXX
000052          RETURN
000052          END
```

```
SUBPROGRAM LENGTH
000060

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
PLACIN -  S00100     END.    -  S00200

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
I       -  000057     ID      -  000056     NCARD  -  000055

START OF CONSTANTS
000053

START OF TEMPORARIES
000053

START OF INDIRECTS
000055

SPACE REQUIRED TO COMPILE
035400
```

SUBJECT: SUBRØUTINE TABL3

PURPØSE: SUBRØUTINE TABLE3 inputs tables of the form F = f(X,Y,Z), table format 3.

METHØD: SUBRØUTINE PLACIN is called to read and count the Z array, the Y array and the X array in turn. Then is recalled to read a dependent array corresponding to the X values for the first Y and Z; then for the second Y, etc. This is repeated for each Z value in turn.

USAGE: CALL TABL3 (T, MAXX, MAXY, MAXZ, NX, NY, NZ)

	T	- -	Table to be read
	MAXX	-	Maximum number of points in X array
	MAXY	-	Maximum number of points in Y array
	MAXZ	-	Maximum number of points in Z array
Output	NX	-	Actual number of X array points input
Output	NY	-	Actual number of Y array points input
Output	NZ	-	Actual number of Z array points input

STØRAGE: The table T is stored as follows:

Address	T(1)	T(MAXX)	T(MAXX+1)	T(MAXX+MAXY)
Content	X(1) X(NX)		Y(1)	Y(NY)
Address	T(MAXX+MAXY+1)	T(MAXX+MAXY+MAXZ)	T(MAXX+MAXY+MAXZ+1)	
Content	Z(1)	Z(NZ)	F(1,1,1)	
Address	T(MAXX+MAXY+MAXZ+NX)- -	T(2MAXX+MAXY+MAXZ+1)		
Content	F(NX,1,1)	F(2,1,1)		

NX≤MAXX NY≤MAXY, NZ≤MAXZ

T can be specified in the calling program by either:

```
DIMENSIØN T(MAXX+MAXY+MAXZ+(MAXX)(MAXY)(MAXZ))
```

or

```
CØMMØN/T/TX(MAXX),TY(MAXY),TZ(MAXZ,
TF(MAXX,MAXY,MAXZ)
```

Note: dependent array stored in natural FØRTRAN order.

SUBPRØGRAM: PLACIN

```
                SUBROUTINE TABL3(T,MAXX,MAXY,MAXZ,NX,NY,NZ)
000012          DIMENSION T(1)
000012          IZ=MAXX*MAXY+1
000013          ITAB=IZ+MAXZ
000015          NZ=0
000015          NCARD=(MAXZ-1)/10+1
000022          CALL PLACIN(T(IZ),NZ,NCARD)
000027          MAXXY=MAXX*MAXY
000034          NY=0
000035          NCARD=(MAXY-1)/10+1
000041          CALL PLACIN(T(MAXX+1),NY,NCARD)
000045          NX=0
000051          NCARD=(MAXX-1)/10+1
000055          CALL PLACIN(T(1),NX,NCARD)
000061          DO 20 I=1,NZ
000066          ID=(I-1)*MAXXY+ITAB
000072          DO 10 K=1,NY
000074          CALL PLACIN(T(ID),NX,NCARD)
000106       10 ID=ID+MAXX
000107       20 CONTINUE
000112          RETURN
000112          END
```

```
SUBPROGRAM LENGTH
000125

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
PLACIN -  000180    END.   -  000200

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
I      -  000122    ID     -  000123    ITAB   -  000117    IZ     -  000116
K      -  000124    MAXKF  -  000121    NCARD  -  000120    NZ     -  000000

START OF CONSTANTS
000113

START OF TEMPORARIES
000113

START OF INDIRECTS
000113

SPACE REQUIRED TO COMPILE
035530
```

SUBJECT: SUBRØUTINE TABL22

PURPØSE: SUBRØUTINE TABL22 inputs tables of the form F = f(X,Y) skewed matrix, table format 2K.

METHØD: SUBRØUTINE PLACIN is called to read and count the Y array. Then for each Y value in turn, PLACIN reads and counts and X array and the corresponding F values. The various X arrays need not have the same number of input points.

USAGE: CALL TABL22 (T,MAXY,MAXX,NY,NX)

T - - Table to be read

MAXY - Maximum of points in Y array

MAXX - Maximum of points in X array

Output NY - Actual number of Y array points input

Output NX - Array with actual numbers of X array points input

STØRAGE:

```
T(1)      T(MAXY)      T(MAXY+1)      T(MAXY+MAXX)

Y(1)   Y(NY)           X(1,1)      X(NX(1),1)

T (MAXY+MAXX11) - - T(MAXY+(MAXY)(MAXX),1) - -

X(1,2)                X(NX(NY),NY)

T(MAXY+(MAXX)(MAXY)+1 - - T(MAXY+MAXY(2MAXX))

F(1,1)                                F(NX(NY),NY)
```

T and NX are specified in the calling program by either:

```
  DIMENSIØN T(MAXY+MAXY(2MAXX)), NX(MAXY)

or CØMMØN/T/TY(MAXY), TX(MAXX,MAXY), TF(MAXX,MAXY)
```

Note: X and Y have been reversed in the actual coding of this routine, but to be consistent with the write-ups for the other TABL routines, the outer loop variable is called Y in this write-up.

SUBPRØGRAM: PLACIN

```
                SUBROUTINE TABL22(T,MAXX,MAXY,NX,NY)
000010          DIMENSION T(1),NY(1)
000010          NX=6
000010          NCARD=(MAXX-1)/10+1
000014          CALL PLACIN(T(1),NX,NCARD)
000017          NCARD=(MAXY-1)/10+1
000026          K=MAXX+1
000027          IK=MAXX+MAXX*MAXY+1
000032          DO 100 I=1,NX
000034          NY(I)=6
000036          CALL PLACIN(T(K),NY(I),NCARD)
000044          CALL PLACIN(T(IK),NY(I),NCARD)
000056          K=K+MAXY
000062          IK=IK+MAXY
000063      100 CONTINUE
000066          RETURN
000066          END
          *DECK KOUNT
```

```
SUBPROGRAM LENGTH
000077

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
PLACIN -  S00100    END.   -  S00200

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
I      -  000076    IK     -  000075    K      -  000074    NCARD  -  000073

START OF CONSTANTS
000067

START OF TEMPORARIES
000067

START OF INDIRECTS
000071

SPACE REQUIRED TO COMPILE
035400
```

```
                PROGRAM DSCRJB(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
          C
          C                    *  PURPOSE - CALCULATES SCRUBBING DRAG                *
          C                    *                                                     *
          C                    *  SUBPROGRAMS REQUIRED - DATABL1,DRAGAFT,SCBOUT      *
          C                    *  RESTRICTIONS/LIMITATIONS - MAXIMUM 1000 CASES      *
          C                    *  AWET NOT CALCULATED.                               *
          C                    *  AUTHOR    T.J. HARRIS            BCS ORG. G-2543   *
          C
          C
000003          COMMON /PRINTNG/  D1(2)   ,P0      ,D2(2)   ,P25P0   ,D3(12) ,TT25     LAST  19
                1 ,D4(16) ,VJS2   ,D5(4)   ,RHOSEC ,D6(4)   ,REYSEC ,D7(43) ,DSCRBS   LAST  90
                2 ,D8(3)  ,DRTAFT ,D9(24)  ,XLCS   ,D10(6)  ,XLAFT  ,D11(21),AWAFT    LAST 154
                3,D12(6),KAWET,D13(4),Q,VBT
000003          COMMON /XCALC  /  D14(1) ,P2oP0   ,D15(3)  ,D16(1)  ,VJS1             LAST   7
000003          COMMON /AMBIENT/  PT0    ,D17(1)  ,V0                                 LAST   3
000003          COMMON /NACELL /  D18(3) ,APJAFT  ,D19(1)  ,HFAN    ,HBT     ,D20(2)   LAST   9
                1                 ,HD    ,D21(1)  ,ROD                                LAST  12
000003          COMMON/SCOUT /TITLE(5),RDATE(2),XLINE(4,1000),BCSN,MAX,ID(2),FIRST

000003          LOGICAL FIRST
          C
          C                    AWET NOT USED THIS TEST
000003          KAWET=0.0
          C         INITIALIZE
000004          MAX=0
000005          KL=0
000006          FIRST=.TRUE.
          C
          C         INPUT HEADER AND BASE GEOMETRY
000007          READ(5,100) TITLE,BCSN
000016          READ(5,200) AWAFT,HFAN,HBT,XLAFT,XLCS,HD,ROD,APJAFT,ID(1)
000044          CALL SCBOUT(1)
          C
          C         INPUT VARIABLE DATA
000046       10 READ(5,300) V1,Q,P0,PT0,P2oP0,TT25,ID(2)
000070          IF(V1.EQ.9999.) GO TO 1000
000072          MAX=MAX+1
          C         WRITE INPUT CARD IMAGES
000074          CALL SCBOUT(2)
          C         CALCULATE ONE CASE
000075       20 CALL DRAGAFT
          C
          C         STORE RESULTS IN OUTPUT ARRAY
000076          KL=KL+1
000100          XLINE(1,KL)= DRTAFT
000102          XLINE(2,KL)= DSCRBS
000105          XLINE(3,KL)= VBT
000107          XLINE(4,KL)=VJS2
000111          GO TO 10
          C                 WRITE OUTPUT ARRAY
000112    1000 CALL SCBOUT(3)
```

```
      C
00011*      100 FORMAT(2A10,10X,F5.0,15X)
00011*      200 FORMAT(9F7.0,16X,A8)
00011*      300 FORMAT(5F7.0,50X,A8)
00011*          END
```

```
              SUBROUTINE SCBOUT(NCALL)
        C
        C                     *  PURPOSE - OUTPUT SCRUBBING DRAG INPUT DECK AND   *
        C                     *            RESULTANT CALCULATIONS--50 LINES/PAGE. *
        C
        C                     *  ARG DEF.                                         *
        C                        . NCALL  CALL FLAG, WHERE:
        C                            1=FIRST CALL-PRINT HEADER INFO.
        C                            2=PRINT INPUT CARD IMAGES
        C                            3=PRINT CALCULATIONS
        C
        C                     *  AUTHOR  T.D.HARRIS                BSC ORG. G-2543 *
        C
000003        COMMON /PRINTNG/  D1(2)   ,P0      ,D2(2)   ,P25P0   ,D3(12) ,TT25     LAST  19
            1 ,D4(16) ,VJS2    ,D5(4)   ,RHOSEC ,D6(4)   ,REYSEC  ,D7(43) ,DSCRBS   LAST  90
            2 ,D8(9)  ,DBTAFT  ,D9(24)  ,XLCS   ,D10(6)  ,XLAFT   ,D11(21),AWAFT     LAST 154
            3,D12(6),XAWET,D13(4),Q,VBT
000003        COMMON /XCALC  /  D14(1) ,P26P0   ,D15(3) ,D16(1)  ,VJS1              LAST   7
000003        COMMON /AMBIENT/  PT0    ,D17(1)  ,V0                                 LAST   3
000003        COMMON /NACELL /  D18(3) ,APJAFT  ,D19(1) ,HFAN    ,HBT     ,D20(2)   LAST   9
            1                 ,HD     ,D21(1)  ,ROD                                LAST  12
000003        COMMON/SCOUT /TITLE(5),RDATE(2),KLINE(4,1000),BCSN,MAX,ID(2),FIRST

000003        INTEGER PGN
000003        LOGICAL FIRST
        C
        C         INITIALIZE LINE AND PAGE COUNT
        C
000003        IF(NCALL.GT.1) GO TO 20
000006        LCN=0
000006        PGN=0
000007        CALL DATE(RDATE(1),RDATE(2))
000012     10 WRITE(6,500)RDATE,TITLE,BCSN
        C         CONSTANTS
000024        WRITE(6,510) AWAFT,HFAN,HBT,XLAFT,XLCS,HD,ROD,APJAFT, ID(1)
000052        GO TO 200
        C
000054     20 IF(NCALL.EQ.3) GO TO 30
000056        LC=LC+1
000057        IF(LC.GE.50)WRITE(6,400)
        C         VARIABLES FOR ONE CASE
000056        WRITE(6,700) V0,Q,P0,PT0,P26P0,TT25,ID(2)
000115        GO TO 200
        C
        C         OUTPUT ARRAY
000112     30 IF(.NOT.FIRST) GO TO 35
000114        BGCN=BCSN
000115        IF(BGCN.EQ.0.0)BGCN=1.0
000117        CSWD=BGCN
        C
000121     35 DO 100  K=1,MAX
000123        IF(FIRST)GO TO 40
```

```
000124          IF(LC.LE.50) GO TO 60
          C
          C         INCREMENT PAGE NO.-RESET LINE COUNTER
000127       40 PGN=PGN+1
000131          LC=0
000132          FIRST=.FALSE.
          C         NEW PAGE  PRINT HEADINGS
000133       50 WRITE(6,800) TITLE,RDATE,PGN
000145          WRITE(6,900)
          C         OUTPUT ONE DATA LINE
000151       60 WRITE(6,1100)CSNO,(XLINE(J,KL),J=1,4)
000106          LC=LC+1
000170          CSNO=CSNO+1.0
000172      100 CONTINUE
000175      400 FORMAT(1H1,*INPUT DECK-CONT.*)
000175      500 FORMAT(1H1,20X,*SCRUBBING DRAG INPUT CARD DECK*,1X,2A10,
               1//1H ,5A10,10X,F5.0)
000175      600 FORMAT(1H ,3F10.4,10X,A8)
000175      700 FORMAT(1H ,5F10.4,20X,A8)
000175      800 FORMAT(1H1,20X,*SCRUBBING DRAG RESULTS*,//10X,5A10,20X,*RUN DATE*,
               11X,2A10,10X,*PAGE NO. *,I4)
          C
000175      900 FORMAT(1H ,*CASE NO.*,15X,*DATAFT*,12X,*SCRBS*,12X,*  VRT  *,14X,
               1* VJS2 *)
000175     1000 FORMAT(1H ,F8.0,4F20.4)
          C
000175     2000 RETURN
000176          END
```

```
                SUBROUTINE DRAGAFT
          C
          C     ****************
          C                    *  PURPOSE - CALCULATES SCRUBBING DRAG (AFTERBODY)  *
          C                    *
          C                    *  REFERENCE TEM116  A.BANAY/N.PREWITT
          C
 000002         EXTERNAL AUX
 000002         DIMENSION Y(4)

 000002         COMMON /PRINTNG/  D1(2)  ,P0     ,D2(2)  ,P25P0  ,D3(12) ,TT25    LAST  19
                1 ,D4(16) ,VJS2   ,D5(4)  ,RHOSEC ,D6(4)  ,REYSEC ,D7(43) ,DSCRBS  LAST  90
                2 ,D8(9)  ,DUTAFT ,D9(24) ,XLCS   ,D10(6) ,XLAFT  ,D11(21),AWAFT   LAST 154
                3,D12(6),XAMET,D13(4),Q,VBT
 000002         COMMON /XCALC  /  D14(1) ,P26P0  ,D15(3) ,D16(1) ,VJS1            LAST   7
 000002         COMMON /AMBIENT/  PTU    ,D17(1) ,V0                              LAST   3
 000002         COMMON /INTGRL /  FX1(16),X1(16) ,B      ,S      ,LANE
 000002         COMMON /NACELL /  D18(3) ,APJAFT ,D19(1) ,HFAN   ,HBT    ,D20(2)  LAST   9
                1                 ,HD    ,D21(1) ,ROD                             LAST  12

 000002         COMMON /DRAGCF /          XMACH(10)     ,VRAT(10)     ,DDDTB(10)
                1                        ,CDIN(4,4,10) ,RDTB(10)     ,APJTB(10)
                2                        ,CDBT(4,2,10) ,HDTB(10)     ,PFP0(10)
                3                        ,RODTB(10)    ,CDAFT(3,3,10)

 000002         COMMON /JETCOR /          PFPATB(16)    ,BCONT1       ,BCONT2(16)
                1                        ,ZKCON1(16)   ,ZKCON2(16)

 000002         DATA CVJ,GM1G/12013.6,0.2857143/
 000002         DATA TO_/0.01/ , RJFAFT/0.06/

 000002         XME2 = SQRT(5.*((P26P0)**GM1G-1.))

 000012         VJS2 = SQRT(CVJ*TT25*(1.-(1./P26P0)**GM1G))

 000023         STFAN = TT25/(1.+.2*XME2**2)
 000027         RHOSEC = P0/STFAN*0.0839
 000042         ZNU = 0.2323E-7*STFAN**1.5/(STFAN+216.)/RHOSEC
 000042         REYSEC = VJS2/ZNU*SQRT(HBT**2+XLAFT**2)

 000051         Y(1) = BCONT1

 000052         Y(2) = TB_U1(P26P0,PFPATB,BCONT2,2,16)
 000057         Y(3) = TB_U1(P26P0,PFPATB,ZKCON1,1,16)
 000063         Y(4) = TB_U1(P26P0,PFPATB,ZKCON2,2,16)

 000067         RATIO = (PTU/P0-1.)/0.49471
 000073         B  = RATIO*(Y(2)-Y(1))+Y(1)
 000076         ZK = RATIO*(Y(4)-Y(3))+Y(3)
 000101         XC = HFAN/ZK**(1./B)
```

```
000110        BRHONU = AWAFT/X_AFT*RHOSEC*ZNU**0.2
000116        COMP = (1.0+0.16*(ME2**2)**(-0.6)
000124        TE4P = 0.035*BRHONU*VJS2**1.6*(COMP+RJFAFT)
000135        DIFF = XLAFT-XC
000137        IF (DIFF) 10,10,12

000140     10 DRG2 = 0.
000141        DRG1 = TE4P*X_AFT**0.8
000146        GO TO 20

000146     12 E = (VJS2-VJ)/XC**8
000155        XCVJ = XC/VJS2**0.25
000161        STEP = 0.
000162        ARG1 = 0.
000163        X1(1)  = XC
000164        FX1(1) = XCVJ**0.2
000167        DIFF = DIFF/15.

000171        DO 14 I=2,15
000172        STEP = STEP+1.
000174        X1(I) = XC+DIFF*STEP
000200        LANE=1

000201        CALL LEGEND(AUX,X1(I-1),X1(I),TOL,ARG2,IERR)

000205        ARG1 = ARG1+ARG2
000207     14 FX1(I) = (ARG1+XCVJ)**0.2
000216        LANE=2

000217        CALL LEGEND(AUX,XC,XLAFT,TOL,ARG1,IERR)

000223        DRG2 = 0.025834*BRHONU*ARG1
000226        DRG1 = TE4P*XC**0.8
000232     15 SAVE=XC-X_OS
000234     16 IF (XC-XLOS) 18,20,20

000237     18 VBT = VJ+E*X_OS**8
000245        GO TO 22
000246     20 VBT = VJS2
000250        IF(Q.NE.0.0) GO TO 25
000251     22 Q = 0.5*RHOSEC*VBT**2

000254     25 CD = TB_UB(MO,P25PO,ROD,HDTB,PFPO,RODTB,CDAFT,1,1,1,3,3,10,3,3,10)
C
000275     26 WRITE(6,33) XC,SAVE,E,B
000321        WRITE(6,100) Q,CD,DBTAFT,XAWET

000325        IF (CD.LT.0.) CD = 0.
000327        DBTAFT = CD*Q*APJAFT
000332        DSDRGS = DRG1+DRG2
000334        IF (XAWET.EQ.0.) RETURN
```

```
000335        DSCRBS = DSCRBS*XAWET/AWAFT
000340        DBTAFT = 0.

000340     93 FORMAT(14 ,* XC=*,F10.4,*XC-XLCS=*,F10.4,* E=*,F10.4,* B=*,F10.4)
000340    100 FORMAT(14 ,* 2.CD.DBTAFT.XAWET =*,4F20.4,10X,*C/D ONLY     *)
000340        RETURN
000341        END DRAGAFT
```

```
              SUBROUTINE AJK(XX,FX)
000005        COMMON /INTGRL /  FX1(16),X1(16) ,B      ,E      ,LANE
000005        COMMON /AMBIENT/  PTO    ,DIF(1) ,VB                      LAST   3
000005        GO TO(10,20),LANE
000013     10 FX= 1./(VB+E*XX**B)**.25
000025        RETURN
000025     20 FX=(VB+E*(XX**3)**1.75/TBLU1(XX,X1,FX1,1,16)
000043        RETURN
000043        END
```

```
                SUBROUTINE DATABL1
          C     ******************

000002          COMMON /DRAGCF /        XMACH(13)     ,WRAT(18)      ,DOOT9(18)     BLKDA 01
                1                      ,COT4(4,4,18) ,ROT3(18)      ,APJT8(18)     BLKDA 02
                2                      ,CDBT(4,5,18) ,HOTB(18)      ,PFP0(18)      BLKDA 03
                3                      ,RODT8(11)    ,CDAFT(3,3,10)                 BLKDA 04

000002          COMMON /JETCOR /        PFPAT8(15)    ,BCONT1        ,BCONT2(16)    BLKDA 05
                1                      ,ZKCON1(16)   ,ZKCON2(16)                   BLKDA 06

000002          COMMON /NOZLCF /        FTP8(20)      ,DELCVP(20)    ,DELCVS(20)    BLKDA 07

000002          DIMENSION CORG(200)                                                 BLKDA 08

000002          EQUIVALENCE (CDBT,CORG)                                             BLKDA 09

000002          DATA XMACH/0.75,0.80,0.85,0.90,6*1./ , WRAT/0.5,0.6,0.7,0.8,6*1./,BLKDA 10
                $  DOOT9/0.78,0.80,0.82,0.84,0.86,0.87,0.88,0.89,0.90,0.91/        BLKDA 11

000002          DATA CDBT/                                                          BLKDA 12
                $ .0868,.0165,.0087,.0092,.0029,.0035,.0081,.0056,.0020,.0025,      BLKDA 13
                $ .0032,.0044,.0015,.0020,.0026,.0029,.0069,.0074,.0135,.0117,      BLKDA 14
                $ .0058,.0042,.0052,.0083,.0027,.0034,.0041,.0055,.0024,.0028,      BLKDA 15
                $ .0036,.0037,.0080,.0084,.0117,.0152,.0046,.0052,.0065,.0106,      BLKDA 16
                $ .0036,.0045,.0054,.0074,.0034,.0040,.0046,.0051,.0032,.0098,      BLKDA 17
                $ .0136,.0193,.0058,.0067,.0068,.0133,.0051,.0056,.0078,.0097,      BLKDA 18
                $ .0145,.0151,.0062,.0072,.0110,.0118,.0154,.0238,.0078,.0085,      BLKDA 19
                $ .0100,.0165,.0067,.0075,.0066,.0122,.0060,.0066,.0082,.0094,      BLKDA 20
                $ .0122,.0130,.0080,.0263,.0059,.0096,.0112,.0178,.0078,.0084,      BLKDA 21
                $ .0039,.0135,.0070,.0076,.0091,.0107,.0132,.0144,.0200,.0290,      BLKDA 22
                $ .0101,.0109,.0126,.0195,.0090,.0094,.0112,.0150,.0080,.0088,      BLKDA 23
                $ .0104,.0128,.0150,.0162,.0230,.0316,.0116,.0124,.0144,.0213,      BLKDA 24
                $ .0132,.0111,.0128,.0100,.0093,.0103,.0120,.0135,.0174,.0190,      BLKDA 25
                $ .0200,.0350,.0137,.0146,.0100,.0233,.0120,.0126,.0146,.0163,      BLKDA 26
                $ .0106,.0115,.0138,.0152,.0196,.0230,.0232,.0387,.0109,.0190,      BLKDA 27
                $ .0220,.0254,.0143,.0152,.0167,.0203,.0125,.0135,.0177,.0170/      BLKDA 28

000002          DATA RDT8/1.2 , 1.8 , 2.0 , 3.0 , 6*7.0 /,                          BLKDA 29
                $  APJT8/7.0,8.0,9.0,10.0,11.0,12.0,13.0,14.0,15.0,16.0/            BLKDA 30

000002          DATA (CORG(I), I=001,100) /                                         BLKDA 31
                $ .0475,.2200,.3850,.5660,.0200,.0590,.2420,.4450,.0000,.0033,      BLKDA 32
                $ .0220,.2030,.0000,.0033,.0150,.0330,.0000,.0000,.0000,.0000,      BLKDA 33
                $ .0450,.2040,.3700,.5520,.0245,.0580,.2350,.4350,.0000,.0030,      BLKDA 34
                $ .0210,.1330,.0000,.0030,.0135,.0300,.0000,.0000,.0000,.0000,      BLKDA 35
                $ .0450,.1950,.3550,.5380,.0300,.0560,.2250,.4260,.0000,.0026,      BLKDA 36
                $ .0230,.1930,.0000,.0026,.0120,.0270,.0000,.0000,.0000,.0000,      BLKDA 37
                $ .0440,.1850,.3400,.5230,.0220,.0550,.2150,.4160,.0000,.0023,      BLKDA 38
                $ .0190,.1860,.0000,.0023,.0110,.0250,.0000,.0000,.0000,.0000,      BLKDA 39
                $ .0430,.1740,.3250,.5100,.0210,.0540,.2030,.4070,.0000,.0021,      BLKDA 40
                $ .0150,.1830,.0000,.0021,.0100,.0230,.0000,.0000,.0000,.0000/      BLKDA 41
```

```
000002        DATA (CDRS(I), I=101,200) /                                    BLKDA 42
             $ .0415,.1550,.3100,.4950,.0200,.0530,.2000,.3980,.0000,.0020,  BLKDA 43
             $ .0145,.1770,.0000,.0020,.0030,.0210,.0000,.0000,.0000,.0000,  BLKDA 44
             $ .0400,.1550,.2350,.4820,.0130,.0520,.1900,.3880,.0010,.0020,  BLKDA 45
             $ .0130,.1720,.0000,.0020,.0030,.0130,.0000,.0000,.0000,.0000,  BLKDA 46
             $ .0330,.1460,.2800,.4660,.0180,.0500,.1800,.3790,.0000,.0020,  BLKDA 47
             $ .0120,.1570,.0000,.0020,.0070,.0160,.0000,.0000,.0000,.0000,  BLKDA 48
             $ .0365,.1360,.2700,.4540,.0150,.0490,.1750,.3700,.0000,.0020,  BLKDA 49
             $ .0110,.1520,.0000,.0020,.0000,.0140,.0000,.0000,.0000,.0000,  BLKDA 50
             $ .0350,.1260,.2550,.4400,.0145,.0480,.1650,.3600,.0010,.0020,  BLKDA 51
             $ .0100,.1560,.0000,.0020,.0050,.0120,.0000,.0000,.0000,.0000/  BLKDA 52

000002        DATA HGT3/0.03, 0.15, 8*0.21/ , PFP8/1.5, 2.2, 8*2.0/ ,        BLKDA 53
             $  ROJTB/2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0,10.0,11.0/             BLKDA 54

000002        DATA CDAFT/                                                    BLKDA 55
             $ .0135,.0195,.0195,.0300,.0625,.0630,.0420,.0990,.1100,        BLKDA 56
             $ .0135,.0145,.0153,.0220,.0375,.0370,.0280,.0570,.0680,        BLKDA 57
             $ .0095,.0110,.0122,.0145,.0215,.0240,.0170,.0310,.0330,        BLKDA 58
             $ .0070,.0082,.0130,.0090,.0120,.0155,.0095,.0155,.0225,        BLKDA 59
             $ .0052,.0050,.0082,.0053,.0065,.0100,.0054,.0072,.0184,        BLKDA 60
             $ .0040,.0050,.0070,.0040,.0050,.0070,.0040,.0050,.0063,        BLKDA 61
             $ .0032,.0045,.0063,.0032,.0045,.0063,.0032,.0045,.0063,        BLKDA 62
             $ .0030,.0045,.0059,.0030,.0045,.0059,.0030,.0045,.0059,        BLKDA 63
             $ .0030,.0045,.0055,.0030,.0045,.0055,.0030,.0045,.0055,        BLKDA 64
             $ .0030,.0045,.0055,.0030,.0045,.0055,.0030,.0045,.0055/        BLKDA 65

000002        DATA PFPATB/                                                   BLKDA 66
             $  1.4    ,1.5    ,1.6    ,2.0    ,2.2    ,2.4    ,2.6    ,2.8    ,  BLKDA 67
             $  3.0    ,3.2    ,3.4    ,3.6    ,3.8    ,4.0    ,4.2    ,4.4    /,  BLKDA 68

             $  dCONT1,dCONT2/-0.413,                                        BLKDA 69
             $  -1.050,-1.430,-1.210,-1.020,-0.878,-0.782,-0.712,-0.660,     BLKDA 70
             $  -0.614,-0.576,-0.545,-0.521,-0.503,-0.488,-0.478,-0.470/,    BLKDA 71

             $  ZKCONT,ZKCONT2/                                              BLKDA 72
             $  2.900 ,2.965 ,3.035 ,3.105 ,3.170 ,3.240 ,3.310 ,3.378 ,     BLKDA 73
             $  3.445 ,3.510 ,3.580 ,3.650 ,3.720 ,3.785 ,3.860 ,3.920 ,     BLKDA 74
             $  79.00 ,59.00 ,42.00 ,28.00 ,18.25 ,13.70 ,10.70 ,8.700 ,     BLKDA 75
             $  7.400 ,5.850 ,5.000 ,5.350 ,5.200 ,5.000 ,4.800 ,4.730 /     BLKDA 76

000002        DATA PTP0,DELCVP,DELCVS/      1.40  ,1.42  ,1.44  ,1.45  ,      BLKDA 77
             $  1.46  ,1.50  ,1.55  ,1.60  ,1.65  ,1.70  ,1.75  ,1.81  ,     BLKDA 78
             $  1.90  ,2.00  ,2.10  ,2.20  ,2.30  ,2.40  ,2.50  ,2.60  ,     BLKDA 79
             $  .00960,.00905,.00858,.00815,.00775,.00743,.00670,.00612,     BLKDA 80
             $  .00565,.00530,.00503,.00481,.00445,.00416,.00366,.00342,     BLKDA 81
             $  .00363,.00357,.00348,.00340,.00340,.00348,.00334,.00352,     BLKDA 82
             $  .00721,.00695,.00645,.00607,.00576,.00553,.00531,.00513,     BLKDA 83
             $  .00463,.00439,.00438,.00419,.00403,.00388,.00374,.00361/     BLKDA 84

000002        RETURN
.00013        END DATA8.1
```

SECTION III

DESCRIPTION OF FUNCTIONS AND LISTINGS

SUBJECT: FUNCTIØN AREA

PURPØSE: FUNCTIØN AREA computes flow area

METHØD: $\text{AREA} = \frac{R\ W\ T}{P\ V\ C_D} = \text{sq. ft.}$

USAGE: A = AREA (W, T, P, V, C_D)

W Inlet mass flow lb/sec

T Nozzle throat static temperature degrees R

P Nozzle throat static pressure lb/ft^2

V Nozzle throat velocity ft/sec

C_D Discharge coefficient = 1.0

```
              FUNCTION AREA(M,T,P,V,CD)
000010        AREA=53.342*M*T/(P*V*CD)
000014        RETURN
000014        END
```

```
SUBPROGRAM LENGTH
000021

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
ENJ.   -  S00100

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
AREA   -  000020

START OF CONSTANTS
000015

START OF TEMPORARIES
000016

START OF INDIRECTS
000020

SPACE REQUIRED TO COMPILE
035200
```

SUBJECT: Thermodynamic function set; HØFT, TØFH, PRØFH, CØFH, and CØFHS.

PURPØSE: The thermodynamic function set produces data from Keenan and Kaye Gas Tables to the program.

METHØD: A polynomial fit of the tabular data is used. The appropriate coefficients are determined by the checking the range of the input variables.

USAGE:

H = HØFT (T,FØA)

T = TØFH (H,FØA)

PR = PRØFH(H,FØA)

H = HØFPR(PR,FØA)

C = CØFH(HT,FØA)

C = CØFHS(HS,FØA)

H	Enthalpy BTU/lb	T	temperature degrees R
FØA	fuel/air ratio	PR	relativepressure
C	sonic velocity ft/sec	HT	total enthalpy BTU/lb
HS	static enthalpy BTU/lb		

The fuel is assumed to be a hydrocarbon of composition, $C_N H_{2_N}$.

```
              FUNCTION HOFT(T, FOA)
        C
        C     AUTHOR                S. C. RAYL          JANUARY, 1971
        C
        C      SUBROUTINE CALCULATES ENTHALPY AS A FUNCTION OF TEMPERATURE AND
        C          FUEL/AIR RATIO. FUEL IS A HYDROCARBON OF COMPOSITION C(N)H(2N)
        C
        C     INPUTS
        C      T       = TEMPERATURE, DEG. R.
        C      FOA     = FUEL/AIR RATIO
        C
        C     OTHER QUANTITIES
        C      ER      = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
        C                     STOICHIOMETRIC FUEL/AIR RATIO)
        C
        C     OUTPUTS
        C      HOFT    = ENTHALPY, BTU/LBM.
        C
000035        DIMENSION A(12), B(12), C(5)
000005        DATA (A(I), I=1,12) /
              A     -.6604879,      242.6325,       -6.752111,      1.337725,
              B     6.733997,       -2.351273,      25.2052667,     188.6858,
              C     43.76564,       -8.994387,      1.053822,       -.0524913 /
000035        DATA (B(I), I=1,12) /
              D     1.635584,       -7.261979,      28.96417,       -12.58440,
              E     2.512776,       0.,             -2.015578,      7.772518,
              F     8.240329,       -.0119861,      -.000119473,    0. /
000035        DATA (C(I), I=1,5) /
              G     .0354312,       -.1061944,      -.7570706,      .1192376,
              H     -.00095369  /
000035        I = 1
000035        IF (T .GT. 1440.) I=7
000011        TX = .001 * T
000012        HOFT = A(I) + TX*(A(I+1) + TX*(A(I+2) + TX*(A(I+3) + TX*(A(I+4)
              I   + TX*A(I+5)))))
000024        IF (FOA .EQ. 0.)  RETURN
000027        XB = B(I) + TX*(B(I+1) + TX*(B(I+2) + TX*(B(I+3)+ TX*B(I+4))))
000037        XC = C(1) + TX*(C(2) + TX*(C(3) + TX*(C(4) + TX*C(5))))
000046        ER = 14.75015 * FOA
000050        HOFT = HOFT + ER *(XB + ER * XC)
000056        RETURN
000056        END
```

```
SUBPROGRAM LENGTH
000126

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END.    - SG0106

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
A       - 000064    B       - 000108    C       - 000114    ER      - 000125
HOFT    - 000063    I       - 000121    TX      - 000122    XB      - 000123
XC      - 000124

START OF CONSTANTS
000037

START OF TEMPORARIES
000062

START OF INDIRECTS
000063

SPACE REQUIRED TO COMPILE
035036
```

```
              FUNCTION TOFH(H, FOA)
        C
        C     AUTHOR                S. C. RAYL          JANUARY, 1971
        C
        C      SUBROUTINE CALCULATES TEMPERATURE AS A FUNCTION OF ENTHALPY AND
        C          FUEL/AIR RATIO. FUEL IS A HYDROCARBON OF COMPOSITION C(N)H(2N)
        C
        C     INPUTS
        C      H       = ENTHALPY, BTU/LBM.
        C      FOA     = FUEL/AIR RATIO
        C
        C     OTHER QUANTITIES
        C      ER      = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
        C                     STOICHIOMETRIC FUEL/AIR RATIO)
        C
        C     OUTPUTS
        C      TOFH    = TEMPERATURE,DEG. R.
        C
000035         DIMENSION  A(12), B(12), C(12)
000035         DATA (A(I), I=1,12) /
               A     5.1364,        4041.695,      1473.279,      -6848.514,
               B     6174.385,      -1283.389,     -51.598,       4883.317,
               C     -2133.152,     1834.790,      -853.3329,     166.2833 /
000035         DATA (B(I), I=1,12) /
               D     -6.4550,       132.0567,      -2278.842,     4793.166,
               E     -3019.591,     0.,            10.1114,       -226.8851,
               F     -269.4027,     72.07638,      -4.734135,     0. /
000035         DATA (C(I), I=1,12) /
               G     .86707,        -25.76924,     342.9403,      -718.8386,
               H     577.1753,      0.,            2.851454,      -.5852354,
               I     45.33835,      -50.47864,     9.416551,      0. /
000035         I = 1
000035         IF (H .GT. 433.468)  I=7
000011         HX = .001 * H
000012         TOFH = A(I) + HX*(A(I+1) + HX*(A(I+2) + HX*(A(I+3) + HX*(A(I+4)
               J  + HX*A(I+5))))))
000024         IF (FOA .EQ. 0.) RETURN
000027         XB = B(I) + HX*(B(I+1) + HX*(B(I+2) + HX*(B(I+3) + HX* B(I+4))))
000037         XC = C(I) + HX*(C(I+1)+ HX*(C(I+2) + HX*(C(I+3) + HX*C(I+4))))
000046         ER = 14.70315 * FOA
000047         TOFH = TOFH + ER*(XB + ER * XC)
000055         RETURN
000055         END
```

```
SUBPROGRAM LENGTH
000134

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END.    -  000100

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
A       -  000063    B       -  000077    C       -  000113    ER      -  000133
MX      -  000130    I       -  000127    TOFM    -  000062    XB      -  000131
XC      -  000132

START OF CONSTANTS
000056

START OF TEMPORARIES
000061

START OF INDIRECTS
000062

SPACE REQUIRED TO COMPILE
035600
```

```
              FUNCTION PROFM(H,FOA)
        C
        C     AUTHOR                S. C. RAYL          JANJARY, 1971
        C
        C      SUBROUTINE CALCULATES RELATIVE PRESSURE AS A FUNCTION OF ENTHALPY
        C      AND FUEL/AIR RATIO.  FUEL IS A HYDROCARBON OF COMPOSITION C(N)H(2N)
        C
        C     INPUTS
        C      H       = ENTHALPY,BTU/LBM
        C      FOA     = FUEL/AIR RATIO
        C
        C     OTHER QUANTITIES
        C      ER      = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
        C                     STOICHIOMETRIC FUEL/AIR RATIO)
        C
        C     OUTPUTS
        C      RELATIVE PRESSURE (DIMENSIONLESS)
        C
000035         DIMENSION A(14),B(14),C(14)
000035         DATA(A(I), I=1,14)/
               A     -.035869328,   1.8093139,     -39.883328,    693.027519,
               B     1270.71594,    139.873914,    318.438817,    11.1827252,
               C     -95.653550,    315.653729,    16.586633,     1881.331732,
               D     182.538424,    36.178329/
000305         DATA(B(I),I=1,14)/
               E     .0275644584,   -1.324704,     26.343109,     -343.703710,
               F     868.624208,    -316.554050,   586.974753,    -13.1470397,
               G     114.649039,    -416.302627,   655.932099,    -639.272328,
               H     1147.663904,   -93.570455/
000305         DATA(C(I),I=1,14)/
               I     .0012308336,   .034872785,    -2.427593,     51.427321,
               J     -212.709122,   196.971116,    -47.809145,    -2.7649713,
               K     21.051354,     -62.621283,    112.113531,    -146.323364,
               L     -24.432159,    113.212657/
000335         I = 1
000035         IF (H .GT. 433.408) I=8
000011         HX = .001 * H
000012         PROFM = A(I) + HX*(A(I+1) + HX*(A(I+2) + HX*(A(I+3) + HX*(A(I+4)
               H + HX*(A(I+5) + HX*A(I+6)))))))
000025         IF (FOA .EQ. 0.) RETURN
000030         XB = B(I) + HX*(B(I+1) + HX*(B(I+2) + HX*(B(I+3) + HX*(B(I+4)
               N + HX*(B(I+5) + HX*B(I+6))))))
000043         XC = C(I) + HX*(C(I+1) + HX*(C(I+2) + HX*(C(I+3) + HX*(C(I+4)
               O + HX*(C(I+5) + HX*C(I+6)))))))
000055         ER = 14.76015 * FOA
000056         PROFM = PROFM + ER *(XB + ER * XC)
000064         RETURN
000064         END
```

SUBPROGRAM LENGTH
000151

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END. - 000130

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS

A	-	000072	B	-	000110	C	-	000126	ER	-	000150
HX	-	000145	I	-	000144	PROFH	-	000071	XB	-	000146
XC	-	000147									

START OF CONSTANTS
000065

START OF TEMPORARIES
000076

START OF INDIRECTS
000071

SPACE REQUIRED TO COMPILE
035730

```
      FUNCTION HOFPR(PR,FOA)
C
C     AUTHOR                S. C. RAYL          JANJARY, 1971
C
C     SUBROUTINE CALCULATES ENTHALPY AS A FUNCTION OF RELATIVE PRESSURE
C     AND FUEL/AIR RATIO. FUEL IS A HYDROCARBON OF COMPOSITION C(N)H(2N)
C
C     INPUTS
C      PR  (DIMENSIONLESS)
C      FOA    = FUEL/AIR RATIO
C
C     OTHER QUANTITIES
C      ER     = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
C                    STOICHIOMETRIC FUEL/AIR RATIO)
C
C     OUTPUTS
C      HOFPR = ENTHALPY,BTU/LBM.
C
000035      DIMENSION A(35),B(35),C(35)
000035      DATA (A(I),I=1,35)/
            A    33.8895665,          8229296.68125,       174424886518.17,
            B    3948257493367 4.E+1, 187844.92625,        1245118736.7728,
            C    125373112.39358E+4,  68.4371986224,       1431962.971228,
            D    2594290477.8428,     5042361312 81.44,    9269.19768,
            E    9155035.4610 ,       816182158.1283,      128.4584235885,
            F    332566.053835,       6334158 1.85517,     1430603949.4649,
            G    1654.19465,          122937.4314 ,        1337688.1442,
            H    226.0305051173,      73274.9365524,       2076179.38711,
            I    0385535.86607,       146.4628128,         2388.88215,
            J    3680.004315,         384.5253717671,      19420.9836934,
            K    37255.36617,         42960.93814,         23.857493,
            L    03.4822981,          16.11137356        /
000035      DATA (B(I),I=1,35)/
            M    3.1513220619,        77522.98151,         -2265401381.3742,
            N    15037124721849.,     -952414266651 9.E+4, 3156140003237 7.E+7,
            O    -4201774641 82.E+12, 5.3216322373,        4442.46644,
            P    -12773345.57330,     1366174249 2.890,    -8005447874970.8,
            Q    2423756672 6112.E+2, -2953236150743.E+5,  6.9061722977,
            R    -2231.88132,         1514306.63426,       -11443244.18617,
            S    018153325.37219,     -1975 6355588.259,   2750013353 11.36,
            T    3.0560291754,        -1225.46637 8,       19606.3119977,
            U    -274428.64924,       2437421.86752,       -11698073.15172,
            V    2424397 3.37542,     -11.9321587185,      -488.605911,
            W    1427.332624,         -3464.406466,        5272.642720,
            X    -4413.38556,         1543.46441         /
000035      DATA (C(I),I=1,35)/
            Y    -.1150147855,        -5043.33861,         62170753.91263,
            Z    -453319426965.92,    171619 8261022.E+2,  -2546618527 4924.E+5,
            A    0.,                  -.3093113042,        -337.366526,
            B    1655804.765232,      -901754007.71809,    365338336676.42,
            C    -5631743849293.,     0.,                  -.4758866541,
            D    300.058573,          -13159.274046,       663816.87816,
```

```
                E     -23371058.45027,     393717503.24254,     0.,
                F     .0847566728,         184.780430,          -2154.622705,
                G     22002.05508,         -127377.01427,       264481.536484,
                H     0.,                  2.6579347583,        81.811924,
                I     -175.05589,          302.29717,           -292.89636,
                J     117.1980897,         0.                   /
000035           I=1
000035           IF(PR .GT. 2.2488)I=8
000011           IF(PR .GT. 22.5614)I=15
000015           IF(PR .GT. 181.1636) I=22
000021           IF(PR .GT. 1208.6219)I=29
000025           PRX= .0001 * PR
000026           HOFPR =(A(I) + PRX*(A(I+1) + PRX*(A(I+2) + PRX* A(I+3))))/
                K (1. + PRX *(A(I+4) + PRX *(A(I+5) + PRX* A(I+6))))
000045           IF (FOA .EQ. 0.) RETURN
000047           XB = B(1) + PRX*(B(I+1) + PRX*(B(I+2) + PRX*(B(I+3) + PRX*(B(I+4)
                L + PRX*(B(I+5) + PRX* B(I+6)))))
000062           XC = C(1) + PRX*(C(I+1) + PRX*(C(I+2) + PRX*(C(I+3) + PRX*(C(I+4)
                M + PRX* C(I+5)))))
000072           ER = 1.476015 * FOA
000074           HOFPR = HOFPR + ER * (XB + ER * XC)
000132           RETURN
000132           END
```

```
SUBPROGRAM LENGTH
000273

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END.    -  S00103

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
A       -  000115    B       -  000160    C       -  000223    ER      -  000272
MOFPR   -  000114    I       -  000206    PRX     -  000267    XB      -  000270
XC      -  000271

START OF CONSTANTS
000133

START OF TEMPORARIES
000112

START OF INDIRECTS
000114

SPACE REQUIRED TO COMPILE
035530
```

```
              FUNCTION COFM(H, FOA)
        C
        C     AUTHOR                S. C. RAYL          JANUARY, 1971
        C
        C     FUNCTION CALCULATES CRITICAL (MACH = 1) SONIC VELOCITY AS A
        C         FUNCTION OF TOTAL ENTHALPY AND FUEL/AIR RATIO. FUEL IS A
        C         HYDROCARBON OF COMPOSITION C(N)H(2N)
        C
        C     INPUTS
        C      H       = TOTAL ENTHALPY, BTU/LBM
        C      FOA     = FUEL/AIR RATIO
        C
        C     OTHER QUANTITIES
        C      ER      = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
        C                     STOICHIOMETRIC FUEL/AIR RATIO)
        C
        C     OUTPUTS
        C      COFM    = CRITICAL SONIC VELOCITY
        C
000035        DIMENSION A(30), B(30), C(30)
000035        DATA (A(I), I=1,30) /
            A     287.73365,      8768.10549,     -34774.99027,   122050.46815,
            B     -294038.34704,  223329.52375,   677.24353,      2627.47247,
            C     3868.23502,     -30713.46231,   46276.23471,    -25109.73645,
            D     331.704145,     6382.33657,     -10614.35870,   11816.68645,
            E     -6552.33102,    1316.05773,     3106.04552,     -10972.50279,
            F     32553.23785,    -41335.83319,   25630.75433,    -6295.60766,
            G     1653.16258,     -922.73017,     5705.83734,     -6352.74300,
            H     3242.37510,     -641.76640/
000035        DATA (B(I), I=1,30) /
            I     -134.44753,     271.37288,      -3950.09389,    13597.19441,
            J     -18575.42276,   6312.31215,     -347.90538,     4612.97154,
            K     -28154.31465,   73169.29856,    -92269.74018,   47771.49423,
            L     4003.39498,     -39756.07127,   129604.13722,   -211856.24957,
            M     172229.93669,   -55795.54381,   -8453.13287,    52978.95471,
            N     -133354.83771,  106887.42429,   -104113.26372,  25939.96315,
            O     1125.77024,     -4777.37150,    7393.33232,     -5636.29346,
            P     2403.40473,     -283.33993/
000035        DATA (C(I), I=1,30) /
            Q     1.92321,        852.25002,      642.83334,      -2059.97789,
            R     2541.93300,     0.,             -67.63421,      722.53325,
            S     -2655.11061,    4459.49806,     -2756.97349,    0.,
            T     7.94337,        -31.91254,      130.83947,      -121.99847,
            U     35.33445,       0.,             -183.39757,     352.91171,
            V     -1393.33069,    1330.86643,     -283.64163,     0.,
            W     -365.73745,     1503.13144,     -2139.30434,    1333.05472,
            X     -316.32364,     0. /
000035        I = 1
000035        IF (H .GT. 291.322) I=7
000011        IF (H .GT. 515.820) I=13
000015        IF (H .GT. 703.331) I=19
000021        IF (H .GT. 914.634) I=25
```

```
000025        HX = .001 * H
000026        COFH = A(I) +HX*(A(I+1) + HX*(A(I+2) + HX*(A(I+3) + HX*(A(I+4)
             Y    +HX*A(I+5))))))
000040        IF (FOA .EQ. 0.)  RETURN
000043        XB = B(I) + HX*(B(I+1) +HX*(B(I+2) + HX*(B(I+3) + HX*(B(I+4)
             Z    + HX*B(I+5))))))
000054        XC = C(I) + HX*(C(I+1) + HX*(C(I+2) + HX*(C(I+3) + HX* C(I+4))))
000063        ER = 14.76015 * FOA
000065        COFH = COFH + ER *(XB + ER * XC)
000073        RETURN
000073        END
```

SUBPROGRAM LENGTH
000243

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END. - SC0100

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS

A	-	000104	B	-	000142	C	-	000200	COFM - 000103
ER	-	000242	HX	-	000237	I	-	000236	XB - 000240
XC	-	000241							

START OF CONSTANTS
000274

START OF TEMPORARIES
000102

START OF INDIRECTS
000103

SPACE REQUIRED TO COMPILE
035436

```
              FUNCTION COFHS(H,FOA)
        C
        C     AUTHOR                S. C. RAYL          JANUARY, 1971
        C
        C      SUBROUTINE CALCULATES SONIC VELOCITY AS A FUNCTION OF STATIC ENTHALPY
        C      AND FUEL/AIR RATIO.  FUEL IS A HYDROCARBON OF COMPOSITION C(N)H(2N)
        C
        C     INPUTS
        C      H       = STATIC ENTHALPY, BTU/LBM.
        C      FOA     = FUEL/AIR RATIO
        C
        C     OTHER QUANTITIES
        C      ER      = EQUIVALENCE RATIO (FUEL/AIR RATIO DIVIDED BY
        C                      STOICHIOMETRIC FUEL/AIR RATIO)
        C
        C     OUTPUTS
        C      COFHS   = SONIC VELOCITY, FT/SEC
        C
000035         DIMENSION A(35),B(35),C(35)
000035         DATA (A(I),I=1,35)/
              A     283.3265,        10797.7055,     -56078.635,     276156.237,
              B     -901657.263,     1633251.076,    -1228974.245,   2725.0400,
              C     -32.52.727,      2000922.877,    -1003542.394,   2109408.2355,
              D     -2325394.359,    1355523.027,    11500.9117,     -122697.6866,
              E     516333.097,      -1613923.3455,  2347553.419,    -1814548.366,
              F     581518.498,      1544.0393,      -2412.4025,     18330.927,
              G     -37471.343,      39654.481,      -21644.153,     4824.6501,
              H     -102408.284,     1034497.552,    -2414549.5854,  3305968.2671,
              I     -2122384.7391,   702759.3932,    -121254.4652    /
000035         DATA (B(I),I=1,35)/
              J     0.372377,        -498.4256,      7412.7167,      -90433.6578,
              K     496456.6757,     -1270950.521,   1247375.8983,   -6554.800795,
              L     147761.2639,     -1050418.5291,  4023685.5963,   -8526354.2777,
              M     3203341.0517,    -4443037.3202,  11026.5075,     -111568.6151,
              N     457335.5561,     -972627.1173,   1117179.3339,   -644185.3436,
              O     139048.4568,     4596.7745,      -33089.0419,    100431.8669,
              P     -159341.3343,    140429.4351,    -65473.2535,    12559.6235,
              Q     512039.3101,     -4500516.3865,  10731807.740,   -13357663.8456,
              R     936433.7488,     -3496414.3017,  543102.1273     /
000035         DATA (C(I),I=1,35)/
              S     2.153333,        -44.231873,     231.5120,       6430.1547,
              T     -96835.9453,     170459.3312,    -194082.4619,   5217.6311,
              U     -90357.3478,     648785.9656,    -2408958.2281,  5253346.6605,
              V     -5923335.5584,   27677839.3485,  -30971.4061,    343020.2405,
              W     -1575083.1954,   3346352.6201,   -5257451.7324,  3815969.6088,
              X     -1143953.8600,   -4673.5059,     35399.2198,     -133777.7593,
              Y     178789.4276,     -104215.3519,   79911.5355,     -16096.9183,
              Z     -712815.17432,   4006761.3184,   -9372293.3747,  11677315.0170,
              1     -6173211.0202,   3046301.8723,   -472627.3643    /
000035         I=1
000035         IF (H .GT. 276.395)  I=8
000011         IF (H .GT. 443.712)  I=15
```

```
000015          IF (M .GT. 617.209)  I=22
000021          IF (M .GT. 968.222)  I=29
000025          HX = .001 * M
000026          COFMS = A(I) + HX*(A(I+1) + HX*(A(I+2) + HX*(A(I+3) + HX*(A(I+4)
               1 + HX*(A(I+5) + HX* A(I+6)))))
000041          IF (FOA .EQ. 0.)  RETURN
000044          XB = B(I) + HX*(B(I+1) + HX*(B(I+2) + HX*(B(I+3) + HX*(B(I+4)
               1 + HX*(B(I+5) + HX*B(I+6)))))
000057          XC = C(I) + HX*(C(I+1) + HX*(C(I+2) + HX*(C(I+3) + HX*(C(I+4)
               1 + HX*(C(I+5) + HX*C(I+6)))))
000071          ER = 14.76015 * FOA
000072          COFMS = COFMS + ER *(XB + ER * XC)
000100          RETURN
000100          END
        *DECK DATA
```

SUBPROGRAM LENGTH
000267

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
END. - 000100

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS

A	-	000111	B	-	000154	C	-	000217	COFHS	-	000110
ER	-	000260	HX	-	000263	I	-	000262	XB	-	000264
XC	-	000265									

START OF CONSTANTS
000101

START OF TEMPORARIES
000107

START OF INDIRECTS
000110

SPACE REQUIRED TO COMPILE
035500

SUBJECT: FUNCTIØN KØUNT

PURPØSE: FUNCTIØN KØUNT counts the number of input points on a ten field seven digit input card.

METHØD: Starting with the right most field a test is made for a blank field. Upon finding a non-blank field that field number is returned as the input count.

USAGE: K = KØUNT (CARD, NFI)

CARD - Array of input points

NFI - Maximum fields defined on card - 10

```
                FUNCTION KOUNT (  CARD,NFI)
          C
          C     KOUNT DETERMINES THE NUMBER OF NON-BLANK INPUT FIELDS ON A DATA
          C     CARD.  THE CARDS ARE PACKED SO THE FIRST BLANK FIELD TERMINATES
          C     THE INPUT.  SEVEN DIGIT FORMAT IS ASSUMED.
          C
000035          DIMENSION   CARD(1)
000035          NF=NFI
000035          K=NF
000037          DO 10 I=1,NF
000011          IF (CARD(K).NE.0.0) GO TO 20
000013          IF (SIGN(1.0,CARD(K)).GT.0.0) GO TO 20
          C     KTH FIELD IS BLANK - CHECK (K-1)TH FIELD
000020       10 K=K-1
          C     KTH FIELD NON-BLANK - K DATA FIELDS INPUT
000023       20 KOUNT=K
000025          RETURN
000025          END
```

```
SUBPROGRAM LENGTH
000034

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
20      -  000023

EXTERNALS AND TAGS
END.    -  500100

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
I       -  000033    K       -  000032    KOUNT  -  000030    NF      -  000031

START OF CONSTANTS
000026

START OF TEMPORARIES
000027

START OF INDIRECTS
000030

SPACE REQUIRED TO COMPILE
035230
```

SUBJECT: FUNCTIØN DEMAND

PURPØSE: FUNCTIØN DEMAND computes engine demand, A_O/A_C, as a function of recovery factor.

METHØD:

$$\text{DEMAND} = \frac{\text{Area } RFW_C}{A_C} + \frac{\text{Area } RFW_S}{A_C} + \frac{A_{O_{BY}}}{A_C}$$

where $\frac{A_{O_{BY}}}{A_C}$ is from Table 7

USAGE: AØAC = DEMAND(RF)

RF is recovery factor

Area = AREAF (Mach)

W_C = Engine airflow

W_S = Secondary airflow

A_C = Capture area

SUBPRØGRAMS: TABU2

```
                FUNCTION DEMAND(PT2PT0)
0000 03         COMMON/AC/AC,ACEAC,ACBVAC,ACGAC,ACIAC
0000 03         COMMON/AREAVL/AREA
0000 03         COMMON/XMZERO/XMZERO
0000 03         COMMON/MC/MCENG,MCSEC
0000 03         COMMON/TAB7/TAB7X(10),TAB7Y(10),TAB7Z(10,10),NTX7,NTY7
0000 03         COMMON/GAMMA/GAMMA
0000 03         CON=AREA          *PT2PT0/AC
0000 06         ACEAC=CON*MCENG
0000 10         ACBVAC=TAB72(ACEAC,XMZERO,TAB7X,TAB7Y,TAB7Z,    NTX7,NTY7,10,10)
0000 21         DEMAND=ACEAC+CON*MCSEC+ACBVAC
0000 25         RETURN
0000 25         END
```

```
SUBPROGRAM LENGTH
000032

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
TA302  -  000100    END.   -  000260

BLOCK NAMES AND LENGTHS
AC     -  000005    AREAVL -  000001    XMZERO -  000001    WC     -  000002
TA37   -  000172    GAMMA  -  000001

VARIABLE ASSIGNMENTS
AC     -  000300C01 AREA   -  000000C02 AQBYAC -  000002C01 AQEAC  -  F10001C01
CON    -  000031    DEMAND -  000030    NTX7   -  000017C05 NTY7   -  000171C05
TA37X  -  000300C05 TA37Y  -  000012C05 TA37Z  -  000024C05 WCENG  -  000000C04
WCSEC  -  000001C04 XMZERO -  000000C03

START OF CONSTANTS
000026

START OF TEMPORARIES
000027

START OF INDIRECTS
000030

SPACE REQUIRED TO COMPILE
035330
```

```
                FUNCTION AREAF(XMACH)
000003          COMMON/GAMMA/GAMMA
000003          DATA C/0.013858/
000003          AREAF=C*(1.+((GAMMA-1.)*.5)*XMACH**2)**((GAMMA+1.)/(2.*(GAMMA-1.)
                ))/(XMACH*SQRT(GAMMA))
000024          RETURN
000024          END
```

```
SUBPROGRAM LENGTH
000036

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
SQRT   - 000100    RBAREX.- 000200    END.   - 000300

BLOCK NAMES AND LENGTHS
GAMMA  - 000001

VARIABLE ASSIGNMENTS
AREAF  - 000034    C      - 000035    GAMMA  - 000000001

START OF CONSTANTS
000025

START OF TEMPORARIES
000036

START OF INDIRECTS
000034

SPACE REQUIRED TO COMPILE
035200
```

```
                FUNCTION AEXMST(AEO,GAMMAP,P7QO)
000036          GM1=GAMMAP-1.0
000037          GP1=GAMMAP+1.0
000040          DVP7=1.0/P7QO
000041          C1=SQRT((2.0/GM1)*(.5*GP1)**(GP1/GM1))
000024          C2=SQRT(1.0-DVP7**(GM1/GAMMAP))
000041          AEXMST=AEO/(C1*C2*DVP7**(1.0/GAMMAP))
000051          RETURN
000052          END
```

```
SUBPROGRAM LENGTH
000074

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

EXTERNALS AND TAGS
SQRT   - S00100   RWAREX.- S00200   END.   - S00300

BLOCK NAMES AND LENGTHS

VARIABLE ASSIGNMENTS
AEXMST - 000060   BP27   - 000071   C1    - 000072   C2    - 000073
GM1    - 000067   GP1    - 000070

START OF CONSTANTS
000053

START OF TEMPORARIES
000056

START OF INDIRECTS
000066

SPACE REQUIRED TO COMPILE
035300
```

SECTION IV

CROSS-LISTING FOR PROGRAM TEM-333

LABELED COMMON CROSS LISTING FOR PROGRAM TEM333

TEM333	17						
ENGNO	3	MACHW8	122	ONREF	21	INS	3
GAMMA	1	G	1	R	1	TAPENO	2
IBAD	1	IDATA	1	OUTS	45	TITLE	8
PCODE	1	DATE	2	CARD	8	KTRAN	1
NEWM	1						
FIXDIM	0						
ATMOS	0						
PERF	0						
COMPUTE	8						
INS	3	G	1	OUTS	45	MACHW8	122
ONREF	21	NOZARR	41	OUTINL	10	ENGNO	3
OUTPUT	7						
RESULT	496	PCODE	1	DATE	2	TITLE	8
NEWM	1	IWARN	1	INS	3		
WARNING	3						
IWARN	1	OUTS	45	INS	3		
STORE	3						
INS	3	OUTS	45	RESULT	496		
AREA	0						
CHKRP	0						
OTHRST	1						
ERMES	5						
HOFT	0						
TOFH	0						
PROFH	0						
HOFPR	0						
OOFH	0						
OOFHS	0						
DATA	0						
PLOTIN	3						
TAPENO	2	CARD	8	IPTAB	1		
TABL1	0						
TABL2	0						

TABLE 3	0						
TABLE 22	0						
KOUNT	0						
TABIN	17						
TAB1	21	TAB2A1	221	ENRICH	820	XMMIN	1
TAB2B	21	TAB2C	21	TAB2D	21	TAB2E	21
TAB3	122	TAB4	122	TAB5	122	TAB6A	122
TAB7	122	TAB8B	21	AC	5	IDESGN	3
XMACHS	1						
INLDRAG	17						
DUTINL	10	XMACHS	1	XMMIN	1	AREAVL	1
PT2PTO	1	TAB1	21	TAB2B	21	TAB3	122
TAB4	122	TAB5	122	AC	5	WC	2
XMACH	1	GAMMA	1	PT	1	XMZERO	1
ADAC	1						
SIZINL	5						
IDESGN	3	TAB1	21	TAB2B	21	TAB2C	21
AC	5						
AREAF	1						
GAMMA	1						
AIRBYP	15						
PT2PTO	1	AREAVL	1	WC	2	GAMMA	1
XMZERO	1	TAB2A1	221	ENRICH	820	AC	5
TAB6A	122	TAB7	122	TAB2D	21	TAB2E	21
XMACH	1	PT	1	ADAC	1		
AIRSPL	14						
PT2PTO	1	AREAVL	1	XMACH	1	PT	1
XMZERO	1	AC	5	WC	2	GAMMA	1
TAB6B	21	TAB2B	21	TAB2C	21	TAB2E	21
TAB2D	21	ADAC	1				
DEMAND	0						
AC	5	AREAVL	1	XMZERO	1	WC	2
TAB7	122	GAMMA	1				
DRG	2						
A	4	DRGTB	445				
DRAGAB	4						
ABINP	41	ABDRAG	3	A	4	AMAX	1
DUTTB	4						
AMAX	1	ABINP	41	ABDRAG	3	ABTAB1	122
BEAMST	0						
DRAGER	3						
ABTAB4	21	ABINP	41	ABDRAG	3		

CSCONV	5						
DSPACE	1	A	4	AMAX	1	ABINP	41
ABDRAG	3						
PLOT4X	6						
AMAX	1	DSPACE	1	ABINP	41	ABDRAG	3
ABTAB3	21	ABTAB2	122				
PLOT4M	6						
ABINP	41	AMAX	1	ABDRAG	3	DSPACE	1
ABTAB2	122	ABTAB3	21				
DINTFR	4						
ABDRAG	3	ABTAB5	122	ABINP	41	DSPACE	1
DBDATR	2						
ABINP	41	ABDRAG	3				
ABINPT	10						
NOZARR	40	ABTAB1	122	ABTAB2	122	ABTAB3	21
ABTAB4	21	ABTAB5	122	ABTAB6	1033	PØPAL	1
AREA1C	2	XMACH9	20				
CABTAB	6						
XMACH9	20	AREA1C	2	AMAX	1	ABINP	41
PØPAL	1	ABTAB6	1033				
SUBBT	3						
AREA1C	2	A	4	DRGTB	446		
SUPBT	3						
A	4	AREA1C	2	DRGTB	446		

PROGRAM TEM333

	COMMON				REFERENCED SUBPROGRAMS				
1	A	LENGTH	4	5 REFERENCES	DRG	DRAGAB	COCONV	SUBBT	SUPBT
2	ABDRAG	LENGTH	3	8 REFERENCES	DRAGAB PLUGNM	DBTTB DINTFR	DBASER DBOATR	COCONV	PLUGMX
3	ABINP	LENGTH	41	9 REFERENCES	DRAGAB PLUGNM	DBTTB DINTFR	DBASER DBOATR	COCONV CABTAB	PLUGMX
4	ABTAB1	LENGTH	122	2 REFERENCES	DBTTB	ABINPT			
5	ABTAB2	LENGTH	122	3 REFERENCES	PLUGMX	PLUGNM	ABINPT		
6	ABTAB3	LENGTH	21	3 REFERENCES	PLUGMX	PLUGNM	ABINPT		
7	ABTAB4	LENGTH	21	2 REFERENCES	DBASER	ABINPT			
8	ABTAB5	LENGTH	122	2 REFERENCES	DINTFR	ABINPT			
9	ABTAB6	LENGTH	1033	2 REFERENCES	ABINPT	CABTAB			
10	AC	LENGTH	5	6 REFERENCES	TABIN DEMAND	INLDRAG	SIZINL	AIRBYP	AIRSPL
11	AMAX	LENGTH	1	6 REFERENCES	DRAGAB CABTAB	DBTTB	COCONV	PLUGMX	PLUGNM
12	AREAWL	LENGTH	1	4 REFERENCES	INLDRAG	AIRBYP	AIRSPL	DEMAND	
13	AREA10	LENGTH	2	4 REFERENCES	ABINPT	CABTAB	SUBBT	SUPBT	
14	ASAC	LENGTH	1	3 REFERENCES	INLDRAG	AIRBYP	AIRSPL		
15	CARD	LENGTH	8	2 REFERENCES	TEM333	PLACIN			
16	DATE	LENGTH	2	2 REFERENCES	TEM333	OUTGO			
17	DAREF	LENGTH	21	2 REFERENCES	TEM333	COMPUTE			

PROGRAM TEM333

COMMON				REFERENCED SUBPROGRAMS				
19	DRGTB	LENGTH 446	3 REFERENCES	DRG	SUBBT	SUPBT		
19	DSPACE	LENGTH 1	4 REFERENCES	DDCONV	PLUGMX	PLUGNM	DINTFR	
20	ENGNO	LENGTH 3	2 REFERENCES	TEM333	COMPUTE			
21	ENRICH	LENGTH 626	2 REFERENCES	TABIN	AIRBYP			
22	ERMES	LENGTH 5	1 REFERENCES	GTHRST				
23	G	LENGTH 1	2 REFERENCES	TEM333	COMPUTE			
24	GAMMA	LENGTH 1	6 REFERENCES	TEM333 DEMAND	INLDRAG	AREAF	AIRBYP	AIRSPL
25	IBAD	LENGTH 1	1 REFERENCES	TEM333				
26	IDATA	LENGTH 1	1 REFERENCES	TEM333				
27	IDESGN	LENGTH 3	2 REFERENCES	TABIN	SIZINL			
28	INS	LENGTH 3	5 REFERENCES	TEM333	COMPUTE	OUTGO	WARNING	STORE
29	IPTAB	LENGTH 1	1 REFERENCES	PLADIN				
30	IWARN	LENGTH 1	2 REFERENCES	OUTGO	WARNING			
31	KTRAN	LENGTH 1	1 REFERENCES	TEM333				
32	MACHNS	LENGTH 122	2 REFERENCES	TEM333	COMPUTE			
33	NEWM	LENGTH 1	2 REFERENCES	TEM333	OUTGO.			
34	NOZARR	LENGTH 60	2 REFERENCES	COMPUTE	ABINPT			

PROGRAM TEM333

	COMMON				REFERENCED SUBPROGRAMS			
35	OUTIN	LENGTH	10	2 REFERENCES	COMPUTE	INLDRAG		
36	OUTS	LENGTH	45	4 REFERENCES	TEM333	COMPUTE	WARNING	STORE
37	PCODE	LENGTH	1	2 REFERENCES	TEM333	OUTGO		
38	PT	LENGTH	1	3 REFERENCES	INLDRAG	AIRBYP	AIRSPL	
39	PT2PT0	LENGTH	1	3 REFERENCES	INLDRAG	AIRBYP	AIRSPL	
40	P8PAL	LENGTH	1	2 REFERENCES	ABINPT	CABTAB		
41	R	LENGTH	1	1 REFERENCES	TEM333			
42	RESULT	LENGTH	436	2 REFERENCES	OUTGO	STORE		
43	TAB1	LENGTH	21	3 REFERENCES	TABIN	INLDRAG	SIZINL	
44	TAB2A1	LENGTH	221	2 REFERENCES	TABIN	AIRBYP		
45	TAB2B	LENGTH	21	4 REFERENCES	TABIN	INLDRAG	SIZINL	AIRSPL
46	TAB2C	LENGTH	21	3 REFERENCES	TABIN	SIZINL	AIRSPL	
47	TAB2D	LENGTH	21	3 REFERENCES	TABIN	AIRBYP	AIRSPL	
48	TAB2E	LENGTH	21	3 REFERENCES	TABIN	AIRBYP	AIRSPL	
49	TAB3	LENGTH	122	2 REFERENCES	TABIN	INLDRAG		
50	TAB4	LENGTH	122	2 REFERENCES	TABIN	INLDRAG		
51	TAB5	LENGTH	122	2 REFERENCES	TABIN	INLDRAG		

PROGRAM TEM333

	COMMON				REFERENCED SUBPROGRAMS			
52	TAB6A	LENGTH	122	2 REFERENCES	TABIN	AIRBVP		
53	TAB6B	LENGTH	21	2 REFERENCES	TABIN	AIRSPL		
54	TAB7	LENGTH	122	3 REFERENCES	TABIN	AIRBVP	DEMAND	
55	TAPEND	LENGTH	2	2 REFERENCES	TEM333	PLACIN		
56	TITLE	LENGTH	8	2 REFERENCES	TEM333	OUTGO		
57	WG	LENGTH	2	4 REFERENCES	INLDRAG	AIRBVP	AIRSPL	DEMAND
58	XMACH	LENGTH	1	3 REFERENCES	INLDRAG	AIRBVP	AIRSPL	
59	XMACHS	LENGTH	1	2 REFERENCES	TABIN	INLDRAG		
60	XPACHB	LENGTH	28	2 REFERENCES	ABINPT	CABTAB		
61	XMMIN	LENGTH	1	2 REFERENCES	TABIN	INLDRAG		
62	XMZERO	LENGTH	1	4 REFERENCES	INLDRAG	AIRBVP	AIRSPL	DEMAND

SECTION V

INLET MAP PROGRAM

```
SASPALV.  000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
SASOALV.  000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000

RUN   VERSION JUL' 71            11.19.18. 72/06/23.

                  PROGRAM INLTMAP(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
000003            COMMON/TAPENO/IN,IOUT
000003            DATA IN,IOUT /5,6/
000003            COMMON/AREAVL/AREA
000003            COMMON/XMACHS/XMACHS
000003            COMMON/XMZERO/XMZERO
000003            COMMON/XMIN/XMMIN
000003            DIMENSION OUT8(10),OUT9(10),OUT10(10,40)
000003            COMMON/OUTINL/OUTINL(5),CD,OUTIN4(4)
000003            COMMON/AC/AC,AZZ(4)
000003            COMMON/TAB1/TAB1X(10),TAB1Y(10),NT1
000003            COMMON/TAB2A /TAB2AX(41),TAB2AY(41),N2A
000003            COMMON/TAB2B/TAB2BX(10),TAB2BY(10),NT2B
000003            COMMON/TAB2C/TAB2CX(10),TAB2CY(10),NT2C
000003            COMMON/TAB2D/TAB2DX(10),TAB2DY(10),NT2D
000003            COMMON/TAB2E/TAB2EX(10),TAB2EY(10),NT2E
000003            COMMON/ADAC/ADAC
000003            DIMENSION TITLE(5),LABL(2)
000003            DIMENSION OUT1(10),OUT2(10),OUT3(10),OUT4(40),OUT5(10,40),OUT6(10)
                 A,          OUT7(10)
000003            DIMENSION TMACH(30)
000003            COMMON/GAMMA/GAMMA
000003            DATA GAMMA/1.4000/
000003            DATA ZERO/10H       0.00 /
000003            DATA PLABLE/10H   PT2/PT0/
000003            DATA OLABLE /10H        CD  /
000003            DATA ASTER/10H      ****/
000003            P=1.0
000005            READ(5,900) TITLE
000012            WRITE(6,800) TITLE
000020            NTMACH=0
000021            WRITE(6,801)
000025            CALL PLACIN(TMACH,NTMACH,3)
000030            READ(5,810) WCL,DELWC,WCU
000042            WRITE(6,802) WCL,DELWC,WCU
000054            CALL TABIN
000055            BYAC=1.0/AC
000057            IF (DELWC) 20,10,20
000060        10  NWC=1
000061            GO TO 30
000062        20  NWC=IFIX((WCU-WCL+.1)/DELWC)+2
000070        30  J=0
000071            NWC=MINO(NWC,40)
000075            DO 300 I=1,NTMACH
000076            J=J+1
000100            XMACH=TMACH(I)
000101            WC=WCL
000103            DO 200 K=1,NWC
000104            CALL INLDRAG(XMACH,P,WC,0.0,0,RF)
000110            IF (I.EQ.1) OUT4(K)=WC*BYAC
000115            OUT5(J,K)=RF
000121            OUT10(J,K)=CD
```

```
000124     200 WC=WC+DELWC
000130         NT2A=-N2A
000132         IF (XMZERO-XMACHS)  205,220,220
000134     205 IF (XMZERO.GE.XMMIN) GO TO 210
000137         WCACOL=ASTER
000140         RFOL=ASTER
000141         CDOL=ASTER
000142         WCACBZ=ASTER
000143         RFBZ=ASTER
000144         CDBZ=ASTER
000145         GO TO 230
000146     210 AJACOL=TBLU1(XMZERO,TAB2EX,TAB2EY,1,NT2E)
000152         RFOL=TBLU1(AJACOL,TAB2AX,TAB2AY,1,NT2A)
000156         WCACOL=AJACOL/(AREA*RFOL)
000160         WC=WCACOL*AC
000162         CALL INLDRAG(XMACH,P,WC     ,0.0,DRAG,RFOL)
000166         CDOL=CD
000170         ENCODE(10,901,RFOL) RFOL
000177         ENCODE(10,902,WCACOL) WCACOL
000207         ENCODE(10,901,CDOL) CDOL
000217         AJACBZ=TBLU1(XMZERO,TAB2DX,TAB2DY,1,NT2D)
000224         IF (AJACBZ.LE.0.0) GO TO 215
000225         RFBZ=TBLU1(AJACBZ,TAB2AX,TAB2AY,1,NT2A)
000232         WCACBZ=AJACBZ/(AREA*RFBZ)
000234         II=0
000235     590 WC=WCACBZ*AC
000237         CALL INLDRAG(XMACH,P,WC     ,0.0,DRAG,RFBZ)
000243         F=AJAC-AJACBZ
000245         IF (ABS(F).LT.0.00005) GO TO 650
000250         IF (II.GT.15) GO TO 640
000254         IF (II.NE.0) GO TO 610
000254         XX=0.5*WCACBZ
000256         GO TO 620
000257     610 IF (F.EQ.FSAVE) GO TO 640
000261         XX=WCACBZ-(WCACBZ-SAVEW)*F/(F-FSAVE)
000267         IF (XX.LT.0.5) XX=0.5
000272     620 II=II+1
000274         SAVEW=WCACBZ
000275         FSAVE=F
000277         WCACBZ=XX
000300         GO TO 590
000306     640 WCACBZ=-0.0
000312         RFBZ=-0.0
000313         CDBZ=-0.0
000315     650 CONTINUE
000315         CDBZ=CD
000317         ENCODE(10,901,RFBZ) RFBZ
000316         ENCODE(10,902,WCACBZ) WCACBZ
000326         ENCODE(10,901,CDBZ) CDBZ
000336         GO TO 230
000337     215 WCACBZ=ZERO
000341         RFBZ=ASTER
000342         CDBZ=ASTER
```

```
000343          GO TO 230
000343      220 ACACDL=TABLU1(AMZERO,TAB2CX,TAB2CY,1,NT2C)
000350          RFDL= TBLU1(XMZERO,TAB2DX,TAB2DY,1,NT2D)
000354          WCACDL=(1.32*ACACDL)/(AREA*RFDL)
000357          WC=WCACDL*AC
000361          CALL INLDRAG(XMACH,P,WC     ,3.0,DRAG,RFDL)
000365          CDDL=CD
000367          ENCODE(10,901,RFDL) RFDL
000376          ENCODE(10,902,WCACDL) WCACDL
000406          ENCODE(10,901,CDDL) CDDL
000416          RFBZ=ASTER
000420          WCACBZ=ASTER
000421          CDBZ=ASTER
000422      230 OUT1(J)=XMACH
000424          OUT2(J)=WCACDL
000426          OUT3(J)=RFDL
000427          OUT6(J)=WCACBZ
000431          OUT7(J)=RFBZ
000432          OUT5(J)=CDDL
000434          OUT9(J)=CDBZ
000435          IF((J.LT.10).AND.(I.LT.NTMACH)) GO TO 330
          C     OUTPUT RECOVERY MAP
000445          LABL(1)=10HINLET RECO
000446          LABL(2)=10HVERY MAP
000450          WRITE(6,903) LABL,TITLE,XMACHS,AC
000463          WRITE(6,904) (OUT1(L),L=1,J)
000476          WRITE(6,905) (OUT2(L),L=1,J)
000511          WRITE(6,906) (OUT3(L),L=1,J)
000524          WRITE(6,907) (OUT6(L),L=1,J)
000537          WRITE(6,908) (OUT7(L),L=1,J)
000552          WRITE(6,909) (PLABLE,L=1,J)
000564          DO 240 K=1,NWC
000566          WRITE(6,910) OUT4(K),(OUT5(L,K),L=1,J)
000604      240 CONTINUE
          C     OUTPUT DRAG MAP
000607          LABL(1)=10HINLET DRAG
000610          LABL(2)=10H MAP
000612          WRITE(6,903) LABL,TITLE,XMACHS,AC
000625          WRITE(6,904) (OUT1(L),L=1,J)
000640          WRITE(6,905) (OUT2(L),L=1,J)
000653          WRITE(6,911) (OUT3(L),L=1,J)
000606          WRITE(6,907) (OUT6(L),L=1,J)
000731          WRITE(6,912) (OUT9(L),L=1,J)
000714          WRITE(6,909) (DLABLE,L=1,J)
000726          DO 250 K=1,NWC
000730          WRITE(6,910) OUT4(K),(OUT10(L,K),L=1,J)
000746      250 CONTINUE
000751          J=0
000752      300 CONTINUE
000755          STOP
000757      600 FORMAT(1H1,20X,9A10)
000757      601 FORMAT(*0MACH NUMBER ARRAY*)
000757      602 FORMAT(*0CORRECTED AIRFLOW POINTS* 10X 7HINITIAL,12X 5HDELTA, 12X
```

```
RUN  VERSION JUL 71          11.19.18. 72/08/23.

              A3HFINAL /25X,3F17.1 )
000757      818 FORMAT(10F7.0)
000757      900 FORMAT(5A10)
000757      901 FORMAT(F10.4)
000757      902 FORMAT(F10.2)
000757      903 FORMAT(1H1,4X,8A10,5X,10HSTART MACH,F5.3,3X,12HCAPTURE AREA,F7.3)
000757      904 FORMAT(3H: 4ACH NO,12X,10F10.3)
000757      905 FORMAT(17H0DISTORTION LIMIT,/3X,10H(WOC/AC)DL,7X,10A10)
000757      906 FORMAT(3X,11H(PT2/PT0)DL,6X,10A10)
000757      907 FORMAT(11H BUZZ LIMIT /3X,10H(WCC/AC)BL,7X,10A10)
000757      908 FORMAT(3X,11H(PT2/PTC)BL,6X,10A10)
000757      909 FORMAT(1H0,5X,6HWOC/AC,8X,10A10)
000757      910 FORMAT(1X,F11.2,8X,10F10.4)
000757      911 FORMAT(5X,6H(CD)DL,9X,10A10)
000757      912 FORMAT(5X,6H(CD)DL,9X,10A10)
000757          END
```

(SAMPLE OUTPUT)

TEST RUN OF INLET MAP PROGRAM

MACH NUMBER ARRAY
.50000 .75000 1.00000 1.25000 1.50000 1.75000 2.00002 .25000 .50000 -0.00000 1

CORRECTED AIRFLOW POINTS	INITIAL	DELTA	FINAL
	60.0	10.0	300.0

INLET 5.0 3.0 -0.0 -0.0 -0.0

TABLE NUMBER 1

0.00000	1.00000	2.00000	3.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-01 TCAP
0.00000	1.00000	2.00000	3.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0

TABLE NUMBER 2A1

.60000	.85000	.90000	1.20000	1.60000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000-02A1 REV
.73000	.75000	.80000	.85000	.87500	.90000	.91000	-0.00000	-0.00000	-0.00000-0.6
.93000	.98000	.97700	.97000	.96500	.95300	.94000	-0.00000	-0.00000	-0.00000-0
.60000	.65000	.70000	.75000	.77500	.78000	.79500	-0.00000	-0.00000	-0.00000-0.8
.98000	.98000	.97500	.97000	.96500	.95300	.92500	-0.00000	-0.00000	-0.00000-0
.60000	.65000	.70000	.73000	.75100	.78000	.77300	-0.00000	-0.00000	-0.00000-0.9
.98200	.97800	.97400	.97000	.96500	.95700	.92500	-0.00000	-0.00000	-0.00000-0
.67500	.76600	.72500	.75000	.77500	.78000	.79000	-0.00000	-0.00000	-0.00000-01.2
.97200	.97000	.96400	.96500	.96500	.95500	.92500	-0.00000	-0.00000	-0.00000-0
.72000	.77500	.82500	.85000	.86100	.87000	.88000	-0.00000	-0.00000	-0.00000-01.6
.92700	.93400	.93100	.92500	.92000	.91400	.87500	-0.00000	-0.00000	-0.00000-0
.65000	.76000	.75000	.80000	.85000	.90000	.95000	-0.00000	-0.00000	-0.00000-02.0
.79200	.79200	.79200	.79200	.79200	.78000	.77000	-0.00000	-0.00000	-0.00000-0

TABLE NUMBER 2B

0.00000	.20000	.40000	.60000	.80000	1.20000	1.40000	1.60000	1.80000	2.00000-02B REV
.80000	.84000	.93400	.96000	.96500	.96200	.95500	.93300	.84000	.79200-0

TABLE NUMBER 2C

.40000	.60000	.80000	1.00000	1.20000	1.40000	1.60000	1.80000	2.00000	-0.00000-02C REV
1.10000	.90000	.77700	.74200	.75200	.78000	.82000	.82000	.81300	-0.00000-0

TABLE NUMBER 2D

0.00000	1.40000	1.60000	1.80000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-02D TCAP
0.00000	0.00000	.86000	.74500	.79000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0

TABLE NUMBER 2E

.40000	.60000	.80000	1.00000	1.20000	1.40000	1.60000	1.80000	2.00000	-0.00000-02E REV
1.25000	.91000	.79500	.76300	.74300	.87200	.87000	.91000	.94000	-0.00000-0

TABLE NUMBER 3

.60000	.80000	.90000	1.20000	1.60000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000-03
.30000	.30000	.60000	.72500	.82000	.84000	.87500	.95000	1.00000	-0.00000-0
.10000	.02900	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	-0.00000-0
.20000	.09000	.04000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	-0.00000-0.8
.33500	.16700	.10800	.03500	0.00000	0.00000	0.00000	0.00000	0.00000	-0.00000-0.9
.49000	.30000	.21700	.10900	.01500	0.00000	0.00000	0.00000	0.00000	-0.00000-01.2
.63500	.41900	.30500	.16500	.06100	.02000	0.00000	0.00000	0.00000	-0.00000-01.6
.91000	.63000	.49000	.31300	.16100	.10500	.10700	0.00000	0.00000	-0.00000-02.0

TABLE NUMBER 4

.90000	1.20000	1.60000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-04 TCAP
0.00000	.01000	.02000	.03000	.04000	.04000	-0.00000	-0.00000	-0.00000	-0.00000-0
0.00000	.01700	.02300	.02650	.01500	.01100	-0.00000	-0.00000	-0.00000	-0.00000-0.8
0.00000	.01600	.02400	.02000	.01050	.01100	-0.00000	-0.00000	-0.00000	-0.00000-01.2
0.00000	.02000	.02200	.01900	.01500	.01050	-0.00000	-0.00000	-0.00000	-0.00000-01.6
0.00000	.01500	.02300	.02350	.02200	.01900	-0.00000	-0.00000	-0.00000	-0.00000-02.0

TABLE NUMBER 5

0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0 5 TCAP

0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0
0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0

TABLE NUMBER 6A

.80000	1.20000	1.60000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-06A TCAP
.60000	.80000	.82500	.85000	.87500	.90000	.92500	.95000	-0.00000	-0.00000-0
.03200	.01900	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	-0.00000	-0.00000-0.8
.03750	.02400	.01900	.01900	.01900	.01900	.01900	.01900	-0.00000	-0.00000-01.2
.04100	.03850	.03720	.03360	.01200	.01200	.01200	.01200	-0.00000	-0.00000-01.6
.05000	.04800	.04790	.04700	.04520	.04500	.04500	.01100	-0.00000	-0.00000-02.0

TABLE NUMBER 6B

0.00000	.68000	.80000	1.20000	2.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-06B TCAP
0.00000	0.00000	.00900	.02600	.04500	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0

TABLE NUMBER 7

0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0 7 TCAP
0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0
0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000-0

INLET RECOVERY MAP TEST RUN OF INLET MAP PROGRAM START MACH 3.000 CAPTURE AREA 5.910

MACH NO	.500	.750	1.000	1.250	1.500	1.750	2.000	.250	.500
DISTORTION LIMIT									
(WCC/AC)DL	****	41.35	39.31	39.32	38.44	37.82	35.58	****	****
(PT2/PT0)DL	****	.9265	.9591	.9491	.9245	.8481	.7736	****	****
BUZZ LIMIT									
(WCC/AC)BL	****	0.00	0.00	0.00	15.30	29.37	29.21	****	****
(PT2/PT0)BL	****	****	****	****	.9014	.8781	.7920	****	****
WCC/AC	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0	PT2/PT0
10.15	.9470	1.0073	1.0120	.9872	.8933	.8469	.7920	.8935	.9470
11.84	.9470	1.0051	1.0093	.9857	.8939	.8495	.7920	.8935	.9470
13.54	.9470	1.0030	1.0066	.9841	.8975	.8522	.7920	.8935	.9470
15.23	.9470	1.0008	1.0039	.9826	.9011	.8549	.7920	.8935	.9470
16.92	.9470	.9987	1.0012	.9810	.9047	.8576	.7920	.8935	.9470
18.61	.9470	.9966	.9986	.9795	.9084	.8603	.7920	.8935	.9470
20.31	.9470	.9944	.9959	.9780	.9121	.8631	.7920	.8935	.9470
22.00	.9470	.9923	.9933	.9764	.9158	.8659	.7920	.8935	.9470
23.69	.9470	.9902	.9907	.9749	.9196	.8686	.7920	.8935	.9470
25.39	.9470	.9881	.9881	.9734	.9233	.8714	.7920	.8935	.9470
27.07	.9470	.9861	.9855	.9719	.9272	.8743	.7920	.8935	.9470
28.77	.9470	.9840	.9829	.9704	.9310	.8771	.7920	.8935	.9470
30.46	.9470	.9819	.9803	.9689	.9349	.8799	.7920	.8935	.9470
32.15	.9470	.9798	.9778	.9674	.9388	.8799	.7905	.8935	.9470
33.84	.9470	.9768	.9752	.9659	.9427	.8786	.7853	.8935	.9470
35.53	.9470	.9734	.9726	.9642	.9403	.8726	.7739	.8935	.9470
37.23	.9470	.9694	.9691	.9609	.9343	.8571	.7472	.8935	.9470
38.92	.9470	.9621	.9623	.9528	.9164	.8298	.7147	.8935	.9470
40.61	.9470	.9391	.9432	.9279	.8873	.7953	.6849	.8935	.9470
42.30	.9470	.9056	.9095	.8941	.8515	.7634	.6575	.8935	.9470
44.00	.9470	.8738	.8745	.8597	.8184	.7341	.6322	.8935	.9470
45.69	.9470	.8385	.8421	.8278	.7885	.7069	.6038	.8935	.9470
47.38	.9470	.8036	.8120	.7983	.7603	.6816	.5871	.8935	.9470
49.07	.9470	.7837	.7843	.7707	.7341	.6581	.5663	.8935	.9470
50.76	.9470	.7547	.7579	.7451	.7095	.6362	.5479	.8935	.9470

INLET DRAG MAP TEST RUN OF INLET MAP PROGRAM START MACH 3.000 CAPTURE AREA 5.910

MACH NO	.500	.750	1.000	1.250	1.500	1.750	2.000	.250	.500
DISTORTION LIMIT									
(WCC/AC)DL	****	41.35	39.31	39.32	38.44	37.82	35.58	****	****
(CD)DL	****	-.0023	.0458	.0449	.0132	.0353	.0231	****	****
BUZZ LIMIT									
(WCC/AC)BL	****	0.00	0.00	0.00	15.38	29.37	29.21	****	****
(CD)BL	****	****	****	****	.5272	.1878	.1773	****	****
WCC/AC	CD	CD	CD	CD	CD	CD	CD	CD	CD
10.15	0.0000	.1969	.3734	.5246	.6432	.7648	.8890	0.0000	0.0000
11.54	0.0000	.1808	.3529	.4958	.6059	.7157	.8257	0.0000	0.0000
13.54	0.0000	.1648	.3325	.4671	.5634	.6663	.7624	0.0000	0.0000
15.23	0.0000	.1489	.3123	.4385	.5306	.6167	.6991	0.0000	0.0000
16.92	0.0000	.1331	.2921	.4039	.4925	.5667	.6358	0.0000	0.0000
18.61	0.0000	.1173	.2721	.3815	.4541	.5164	.5725	0.0000	0.0000
20.31	0.0000	.1016	.2521	.3531	.4153	.4657	.5092	0.0000	0.0000
22.00	0.0000	.0869	.2322	.3248	.3762	.4147	.4470	0.0000	0.0000
23.69	0.0000	.0731	.2125	.2956	.3358	.3634	.3848	0.0000	0.0000
25.38	0.0000	.0597	.1929	.2685	.2971	.3116	.3221	0.0000	0.0000
27.07	0.0000	.0477	.1733	.2405	.2566	.2595	.2578	0.0000	0.0000
28.77	0.0000	.0458	.1539	.2129	.2158	.2069	.1937	0.0000	0.0000
30.46	0.0000	.0337	.1346	.1853	.1716	.1537	.1311	0.0000	0.0000
32.15	0.0000	.0265	.1156	.1579	.1350	.1036	.0735	0.0000	0.0000
33.84	0.0000	.0219	.0969	.1305	.0911	.0584	.0210	0.0000	0.0000
35.53	0.0000	.0224	.0777	.1020	.0549	.0492	.0232	0.0000	0.0000
37.23	0.0000	.0228	.0623	.0738	.0381	.0378	.0167	0.0000	0.0000
38.92	0.0000	.0196	.0483	.0493	.0134	.0323	.0167	0.0000	0.0000
40.61	0.0000	.0036	.0438	.0387	.0219	.0323	.0167	0.0000	0.0000
42.30	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000
44.00	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000
45.69	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000
47.38	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000
49.07	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000
50.76	0.0000	-.0023	.0379	.0358	.0219	.0323	.0167	0.0000	0.0000

Unclassified
Security Classification

PRECEDING PAGE BLANK-NOT FILMED

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)*
The Boeing Company
P. O. Box 3999
Seattle, WA 98124

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b GROUP

3 REPORT TITLE
Propulsion System Installation Corrections
Volume II: Programmers Manual

4 DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Final Report 31 December 1971 to 31 December 1972

5 AUTHOR(S) *(First name, middle initial, last name)*
William H. Ball

6 REPORT DATE
December 1972

7a TOTAL NO OF PAGES
xii + 216 = 228

7b NO OF REFS
0

8a. CONTRACT OR GRANT NO
F33615-72-C-1580

b. PROJECT NO
1366

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
AFFDL-TR-72-147

9b OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10 DISTRIBUTION STATEMENT
Distribution limited to U.S. Government agencies only; test and evaluation; statement applied 29 December 1972. Other requests for this document must be referred to Air Force Flight Dynamics Laboratory, (PTB), Wright-Patterson Air Force Base, Ohio 45433

11 SUPPLEMENTARY NOTES

12 SPONSORING MILITARY ACTIVITY
Flight Dynamics Laboratory
Air Force Systems Command
Wright-Patterson Air Force Base, OH

13 ABSTRACT
This report presents the results of a research program to develop a procedure for use in calculating propulsion system installation losses. These losses include inlet and nozzle internal losses and external drag losses for a wide variety of subsonic and supersonic aircraft configurations up to Mach 4.5. The calculation procedure, which was largely developed from existing engineering procedures and experimental data, is suitable for preliminary studies of advanced aircraft configurations. Engineering descriptions, equations, and flow charts are provided to help in adapting the calculation procedures to digital computer routines. Many of the calculation procedures have already been programmed on the CDC 6600 computer. Program listings and flow charts are provided for the calculation procedures that have been programmed. The work accomplished during the program is contained in four separate volumes. Volume I contains an engineering description of the calculation procedures. Volume II is a programmers manual containing flow charts, listings, and subroutine descriptions. Volume III contains sample calculations and sample input data. Volume IV contains bookkeeping definitions and data correlations.

DD FORM 1473

Unclassified
Security Classification

KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Afterbody Drag						
Boattail Drag						
Bookkeeping Aero-Propulsion Forces						
Boundary Layer Bleed Drag						
Bypass Drag						
Inlet Performance						
Inlet Shock Losses						
Nozzle/Afterbody Installation Losses						
Nozzle Interference Drag						
Nozzle Thrust Coefficient						
Propulsion Installation Losses						
Spillage Drag						
Subsonic Diffuser Losses						
Supersonic Inlets						
Total Pressure Recovery						