ARL 65-58

A DIGITAL COMPUTER PROGRAM FOR CONDENSATION IN EXPANDING ONE-COMPONENT FLOWS

LEONARD J. HARDING

UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN

MARCH 1965

Contract AF 33(657)-8867
Project 7116
Task 7116-01

AEROSPACE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

FOREWORD

This technical report was prepared by the College of Engineering, The University of Michigan, Ann Arbor, Michigan, on Contract AF 33(657)-8867 for the Aerospace Research Laboratories, Office of Aerospace Research, United States Air Force. The research reported herein was accomplished on Task 7116-01, "Internal Flow Research," of Project 7116, "Energy Conversion Research," under the technical cognizance of Everett D. Stephens of the Thermo-Mechanics Research Laboratory of the Aerospace Research Laboratories.

This report represents a continuation of the work, "Digital Computer Analysis of Condensation in Highly Expanded Flows," by James L. Griffin reported in ARL 63-206, November 1963.

ABSTRACT

This report describes a digital computer program for calculating vapor condensation processes that occur in rapidly expanding flows. The treatment emphasizes the program logic required to satisfy the requirements of the mathematical model. Among the most important of these requirements are the search for the onset of nucleation, in an isentropically expanding flow, and the iteration procedure necessary for the joint solution of the nucleation and growth equations and the diabatic flow equations. The program features flexibility in allowable input conditions, short execution time, and a convenient output format.

TABLE OF CONTENTS

1. INTRODUCTION

In Ref. 1, Griffin reported on a digital computer program for analyzing condensation processes occurring in rapidly expanding flows of pure vapors. The present report describes, from the computer programmer's point-of-view, the current version of that original program. Additional details on the basis for changes made in the mathematical model and on results obtained with the current program are presented in Ref. 2.

As was the original program, this version is coded in the Michigan Algorithm Decoder (MAD) language, Ref. 3, for running on an IBM 7090 with The University of Michigan Executive System. (Information on the Michigan Executive System is available through the SHARE organization.) In actuality the program herein described represents, except for a few subroutines, a recording of the original program that allows considerably more flexibility and at the same time a substantial decrease in computer usage time per problem. Although the mathematical model has not been altered significantly, the logical structure has been changed considerably. The program has been tested and extensively use; however, no guarantee as to its accuracy or functioning can be made.

Manuscript released by the author September 1964 for publication as an ARL Technical Documentary Report.

2. MATHEMATICAL MODEL

(by K. R. Sivier)

The following listing of equations, along with brief comments about their use and the interrelationships between them, forms the basis of the mathematical model for the computer program and is included for completeness. The treatment has been kept brief since the mathematical model is discussed in detail elsewhere. Reference 1 presents a detailed discussion of the mathematical model of the original program. Most of this material continues to apply to the present modified program. Reference 2 presents discussions of Tolman surface tension correction, the change in the method of dealing with the vapor saturation data, and the equations involved in the ISEN subroutine. The foreign nuclei and variable step modifications involve program logic and operation only and, hence, require no new equations. They are, however, also discussed in Reference 2.

Much of the following material has been taken directly from Section V-B of Ref. 1 with little or no change. The treatment there is sufficiently concise to recommend its inclusion here.

2.1 NOMENCLATURE

The following list contains only those symbols used in the equations in the present description of the mathematical model. To be consistent with the computer program itself, all units are in the cgs system.

A	Nozzle cross-section area (cm^2)
A*	Nozzle throat area (cm^2)
C_p	Vapor specific heat at constant pressure (dyne-cm/gm-^oK)
D	Tolman constant used to correct surface tension for finite drop size (cm)
g	Mass fraction of the mixture that is in the condensed phase

J	Specific drop nucleation rate (drops/sec-cm^3)
k	Boltzmann's gas constant (1.379×10^{-16} dyne-cm/^{o}K)
L	Latent heat of vaporization (dyne-cm/gm)
$(L) \ldots (L)_1$	Empirical latent heat constants, see Eq. (12)
M	Mach number
$\mathring{m}$	Rate of mass flow (gm/sec)
$\mathring{N}$	Drop nucleation rate (drops/sec)
N_A	Avogadro's number (6.027×10^{23} molecules/gmol)
p	Free-stream static pressure (dyne/cm^2)
p_o	Supply (stagnation) pressure (dyne/cm^2)
p_o'	Stagnation pressure behind a normal shock wave (dyne/cm^2)
p_∞	Saturation vapor pressure for a plane surface of liquid (dyne/cm^2)
$(PSAT) \ldots (PSAT)_4$	Empirical saturation curve constants, see Eqs. (10) and (11)
R	Universal gas constant (8.314×10^7 dyne-cm/gmol-^{o}K)
$(RHOL) \ldots (RHOL)_1$	Empirical liquid density constants, see Eq. (13)
r	Drop radius (cm)
r^*	Critical drop radius (cm)
$(SIGMA) \ldots (SIGMA)_1$	Empirical surface tension constants, see Eq. (14)
T	Free-stream static temperature (^{o}K)
T_o	Supply (stagnation) temperature (^{o}K)
T_{drop}	Temperature of a drop of condensate (^{o}K)
T_∞	Saturation vapor temperature for a plane surface of liquid (^{o}K)

U	Velocity (cm/sec)
x	Nozzle station measured from the throat (cm)
α	Accomodation coefficient
γ	Ratio of specific heats
ϵ_p	Fractional deviation, in a distance Δx, of the static pressure change from the corresponding change for an isentropic expansion
ϵ_T	Fractional deviation, in a distance Δx, of the static temperature change from the corresponding change for an isentropic expansion
$\epsilon_{critical}$	fractional deviation, in a distance Δx, of the change in static pressure or temperature, from the corresponding isentropic change, that occurs at the condensation onset point
μ	Molecular weight (gm/gmol)
ρ	Free-stream static density of the mixture (gm/cm^3)
ρ_o	Supply (stagnation) density (gm/cm^3)
ρ_L	Liquid phase density (gm/cm^3)
σ	Surface tension (dyne/cm)
σ_∞	Surface tension of a plane liquid surface (dyne/cm)
θ_{in}	Half-angle of the convergent section of the nozzle (deg)
θ_{out}	Half-angle of the divergent section of the nozzle (deg)

Subscripts

$(\ )_{con}$	Refers to conditions existing at the condensation onset point as established by the program
$(\ )_i$	Refers to the incremental step in which the droplets were originally formed

$(\ )_j$	Refers to the incremental step presently being evaluated
$(\ )_s$	Refers to conditions existing in an isentropically expanding flow
$(\ )_{sat}$	Refers to conditions existing at the vapor saturation point
$(\ )_T$	Refers to an incremental step Δx, at temperature T, that is being tested in the search for the onset of condensation

2.2 ISENTROPIC EXPANSION

The expansion of the vapor from its supply condition to the saturation point and, if the condensation onset point is found, the expansion of the supersaturated vapor from the saturation point to the condensation onset point are governed by the following isentropic flow equations:

$$p = p_o \left(\frac{T}{T_o}\right)^{\frac{\gamma}{\gamma - 1}} \tag{1}$$

$$M = \left[\frac{2}{\gamma - 1}\left(\frac{T_o}{T} - 1\right)\right]^{\frac{1}{2}} \tag{2}$$

$$\rho = \rho_o \left(1 + \frac{\gamma - 1}{2} M^2\right)^{-\frac{1}{\gamma - 1}} \tag{3}$$

$$A = \frac{A^*}{M}\left[\frac{2}{\gamma + 1}\left(1 + \frac{\gamma - 1}{2} M^2\right)\right]^{\frac{\gamma + 1}{2(\gamma - 1)}} \tag{4}$$

$$U = [2\, C_p\, (T_o - T)]^{\frac{1}{2}} \tag{5}$$

For a given value of T, the above equations are readily evaluated for given values of p_o, T_o, ρ_o, γ, C_p, and A^*.

2.3 NOZZLE GEOMETRY

In the case of a wedge nozzle (two-dimensional flow), the nozzle is assumed to have unit width and its area is given by

$$A = A^* - 2x \tan \theta_{in} \tag{6}$$

upstream ($x < 0$) of the throat ($x = 0$), and

$$A = A^* + 2x \tan \theta_{out} \tag{7}$$

downstream ($x > 0$) of the throat.

For the conical nozzle (three-dimensional flow), the area is given by

$$A = \pi \left(\sqrt{\frac{A^*}{\pi}} - x \tan \theta_{in} \right)^2 \tag{8}$$

upstream of the throat, and

$$A = \pi \left(\sqrt{\frac{A^*}{\pi}} + x \tan \theta_{out} \right)^2 \tag{9}$$

downstream of the throat.

2.4 VAPOR SATURATION DATA

Vapor saturation data are supplied to the program in the form of curve fits to empirical data, expressed in terms of $\ln p_\infty$ and $1/T_\infty$. Below the triple point of the vapor, the linear (Clausius-Clapeyron) approximation

$$\frac{1}{T_\infty} = (PSAT) + (PSAT)_1 \ln p_\infty \quad , \tag{10}$$

is used and, above the triple point, the slight non-linearity in the data is accounted for by the quadratic approximation

$$\frac{1}{T_\infty} = (PSAT)_2 + (PSAT)_3 \ln p_\infty + (PSAT)_4 (\ln p_\infty)^2 \quad , \tag{11}$$

where (PSAT) . . . $(PSAT)_4$ are empirically determined constants.

2.5 LATENT HEAT AND LIQUID DENSITY DATA

Latent heat of vaporization and liquid density data are supplied to the program as linear functions of temperature;

$$L = (L) + (L)_1 \ T \tag{12}$$

and

$$\rho_L = (RHOL) + (RHOL)_1 \ T_{drop} \quad , \tag{13}$$

where (L), $(L)_1$, (RHOL), and $(RHOL)_1$ are empirically determined constants and the temperature T_{drop} is taken as the saturation temperature corresponding to the local ambient pressure of the vapor.

2.6 SURFACE TENSION DATA AND CRITICAL DROP SIZE

Surface tension data, corresponding to an infinite plane liquid surface, is supplied to the program as a linear function of temperature (an Eötvös-type variation), i. e.,

$$\sigma_\infty = (SIGMA) + (SIGMA)_1 \ T_{drop} \quad , \tag{14}$$

where T_{drop} is taken as the saturation temperature corresponding to the local vapor pressure. An estimate of the surface tension for very small drops is obtained from this value by use of the Tolman relation

$$\sigma = \frac{\sigma_\infty}{1 + \frac{D}{r}} \quad , \tag{15}$$

where D is taken as an empirically evaluated constant.

As shown by Eq. (29) below, the critical drop size is a linear function of surface tension, i. e.,

$$r^* = \left(\frac{2\mu}{\rho_L \, RT \ln \frac{p}{p_\infty}} \right) \sigma \quad .$$

The joint solution of this relation with Eq. (15) for surface tension results in the following simple expression for critical drop size with the corrected surface tension value:

$$r^* = \left(\frac{2\mu}{\rho_L \, RT \ln \frac{p}{p_\infty}} \right) \sigma_\infty - D \quad . \tag{16}$$

2.7 SATURATION POINT DETERMINATION

The state of the vapor at its saturation point is established by the joint solution of the appropriate saturation data approximation, Eq. (10) or (11), and the isentropic relation

$$p_{sat} = p_o \left(\frac{T_{sat}}{T_o} \right)^{\frac{\gamma}{\gamma - 1}} \quad . \tag{17}$$

The values of p_{sat} and T_{sat}, thus found, are used with the isentropic relations, Eqs. (2) through (5), to calculate M_{sat}, A_{sat}, ρ_{sat}, and U_{sat}. The location of the saturation point, x_{sat}, is determined from the value of A_{sat} and the appropriate nozzle equation, one of eqs. (6) through (9). The total mass flow through the nozzle is calculated from the equation

$$\mathring{m} = \rho_{sat} \, U_{sat} \, A_{sat} \quad . \tag{18}$$

2.8 ONSET OF CONDENSATION DETERMINATION

The point of onset of condensation is determined by taking incremental temperature steps, ΔT, of supersaturation and testing for significant condensation effects at the end of each successive step. To initiate this process, the isentropic values for the state of the vapor at the end of each ΔT step are introduced into the nucleation equations for critical drop size and formation rate; i. e.,

$$r_T^* = \frac{2\mu\sigma_\infty}{\rho_L RT \ln\left(\frac{p}{p_\infty}\right)} - D \tag{19}$$

$$J_T = \left(\frac{p}{kT}\right)^2 \frac{1}{\rho_L} \sqrt{\frac{2\sigma\mu}{\pi N_A}}\, e^{-\frac{4\pi\sigma r^{*2}}{3kT}} \quad . \tag{20}$$

Note that the choice of a very small ΔT will cause the first step to remain very near the saturation point and that, for p/p_∞ near unity, r^* approaches infinity. However, r^* falls off very rapidly as supersaturation increases and for most cases a choice of $\Delta T \geq 5^\circ K$ is sufficient to avoid this problem.

Since it is assumed that the critical drops experience no growth in the Δx increment in which they are formed and since the nucleation rate for this increment is

$$N_T = J_T A_T \Delta x_T \quad , \tag{21}$$

the mass fraction of condensate formed in this increment is

$$\Delta g_T = \frac{4\pi\rho_L}{3\mathring{m}} \mathring{N}_T r_T^{*3} \tag{22}$$

Next, the epsilon equations

$$\epsilon_p = \frac{A}{\Delta A}\left(\frac{L}{C_p T} - 1\right) \Delta g_T \tag{23}$$

$$\epsilon_T = \frac{A}{\Delta A}\left[\left(\frac{\gamma - \frac{1}{M^2}}{\gamma - 1}\right)\frac{L}{C_p T} - 1\right]\Delta g_T \quad , \tag{24}$$

which test the effect of condensation on static pressure and temperature, respectively, are used to determine if this amount of condensate is sufficient to cause a significant deviation from the isentropic expansion. Because of their relative sensitivities, ϵ_p is used for subsonic Mach numbers and ϵ_T is used for supersonic Mach numbers.

The value of ϵ_T, or ϵ_p, is compared with the value of $\epsilon_{critical}$, which is assumed to represent a significant condensation effect and which is supplied as part of the initial data for the program. If the ϵ_T is less than $\epsilon_{critical}$ the calculations are repeated at step $T + \Delta T$. This process continues until $\epsilon_{critical}$ is equalled or exceeded. When exceeded, a bracketing procedure is employed to improve the estimate. All quantities computed for Δg at values of $\epsilon < \epsilon_{critical}$ are discarded and it is assumed that there has been no condensate formed prior to $\epsilon = \epsilon_{critical}$. Once the condensation onset temperature is found, the isentropic equations, Eqs. (1) through (5), and the appropriate nozzle equation, Eqs. (6) through (9), are used to determine the following values at the condensation point:

$$p_{con},\ M_{con},\ \rho_{con},\ A_{con},\ U_{con},\ \text{and}\ x_{con} \quad .$$

2.9 CONDENSING FLOW

The condensing portion of the flow requires a joint solution of the nucleation and growth equations and the diabatic flow equation. The calculations are performed for increments of Δx, starting from the onset of condensation. The diabatic flow equations, in incremental form, are

$$\frac{\Delta\rho}{\rho} + \frac{\Delta U}{U} + \frac{\Delta A}{A} = 0 \qquad \text{(Continuity)} \qquad (25)$$

$$\frac{\Delta p}{p} = - \frac{U\Delta U}{(1 - g)\frac{R}{\mu}T} \qquad \text{(Momentum)} \qquad (26)$$

$$\frac{\Delta p}{p} = \frac{\Delta\rho}{\rho} + \frac{\Delta T}{T} - \frac{\Delta g}{(1 - g)} \qquad \text{(State)} \qquad (27)$$

$$U\Delta U + C_p\Delta T - L\Delta g = 0 \qquad \text{(Energy)} \qquad (28)$$

The value of Δg, appearing in the above equations, is determined from the following nucleation and growth equations:

$$r^*_i = \frac{2\sigma\mu}{\rho_L RT \ln\frac{p}{p_\infty}} \qquad (29)$$

$$J_i = \left(\frac{p}{kT}\right)^2 \frac{1}{\rho_L}\left(\frac{2\sigma\mu}{\pi N_A}\right)^{1/2} e^{-\frac{4\pi\sigma r^{*2}}{3kT}} \qquad (30)$$

$$\mathring{N}_i = J_i A_i \Delta x_i \qquad (31)$$

$$\Delta r_j = \frac{\alpha}{L}\frac{p}{\rho_L}\left(\frac{2}{\pi}\right)^{1/2}\left(\frac{kN_A}{\mu T}\right)^{1/2}(T_{drop} - T)\frac{\Delta x}{U} \qquad (32)$$

$$r_{ij} = r_{i(j-1)} + \Delta r_j \qquad (33)$$

$$\Delta g_j = \frac{4\pi\rho_L}{\mathring{m}}\left[\sum_{i=1}^{j-1} \mathring{N}_i r_{ij}^2 \Delta r_j + \frac{1}{3}\mathring{N}_j r^*_j{}^3\right] \qquad (34)$$

$$g_j = g_{j-1} + \Delta g_j \qquad (35)$$

In the above, the subscripts i and j are numbered increments of x starting from the condensation point. i acts as a label for each group of droplets of a particular size and denotes the particular increment in which these drops originated as critical sized drops. j denotes the increments presently under consideration. Thus a designation r_{13} denotes droplets which were formed as critical drops in increment 1, have undergone the growth Δr_2, and are now undergoing the growth Δr_3.

2.10 ISEN SUBROUTINE CALCULATIONS

The ISEN subroutine is used to evaluate the changes in the local flow conditions that have been produced by the condensation processes. To do this, the conditions computed for the condensing flow are compared with the corresponding isentropic flow conditions that would exist if condensation was not occurring.

The isentropic flow conditions are based entirely on the geometric area ratio. An inversion of Eq. (4) is performed to determine the isentropic Mach number, M_s, corresponding to the local nozzle area. M_s is determined by a half-interval iteration technique, starting with the following first approximation for M_s;

$$M_s = \left[1 + \left(\frac{A}{A^*} - 1\right)\left(\frac{\gamma + 1}{\gamma - 1}\right)^{\frac{\gamma + 1}{2(\gamma - 1)}}\right]^{\frac{\gamma - 1}{2}} \quad . \tag{36}$$

Once M_s is determined, the isentropic flow equations yield the following isentropic ratios for static temperature, pressure, and density:

$$\bar{T}_s = \frac{T_s}{T_o} = \left(1 + \frac{\gamma - 1}{2} M_s^2\right)^{-\frac{1}{\gamma - 1}} \tag{37}$$

$$\bar{p}_s = \frac{p_s}{p_o} = \left(1 + \frac{\gamma - 1}{2} M_s^2\right)^{-\frac{\gamma}{\gamma - 1}} \tag{38}$$

$$\bar{\rho}_s = \frac{\rho_s}{\rho_o} = \left(1 + \frac{\gamma - 1}{2} M_s^2\right)^{-\frac{\gamma}{\gamma - 1}} \tag{39}$$

In addition, M_s is used to calculate the stagnation pressure ratio across a normal shock wave for the isentropically expanded flow; i. e.,

$$\left(\frac{p_o'}{p_o}\right)_s = \left[\frac{(\gamma + 1) M_s^2}{(\gamma - 1) M_s^2 + 2}\right]^{\frac{\gamma}{\gamma - 1}} \left[\frac{\gamma + 1}{2\gamma M_s^2 - (\gamma - 1)}\right]^{\frac{1}{\gamma - 1}} . \tag{40}$$

The static ratios for the condensing flow are found simply from the relations

$$\bar{T} = \frac{T}{T_o} \tag{41}$$

$$\bar{p} = \frac{p}{p_o} \tag{42}$$

$$\bar{\rho} = \frac{\rho}{\rho_o} \tag{43}$$

In addition, the Mach number of the condensing flow is used to obtain the ratio of stagnation pressure downstream of a normal shock wave to the upstream static pressure (completely neglecting the effect of the condensate on the recovery process); i. e.,

$$\frac{p_o'}{p} = \left[\frac{(\gamma + 1)\, M^2}{2}\right]^{\frac{\gamma}{\gamma - 1}} \left[\frac{\gamma + 1}{2\gamma M^2 - (\gamma - 1)}\right]^{\frac{1}{\gamma - 1}} \quad . \tag{44}$$

Finally, the ratios indicating the extent of the non-isentropic condensation effects are found directly as:

$$\hat{T} = \frac{T}{T_s} = \frac{\overline{T}}{\overline{T}_s} \tag{45}$$

$$\hat{p} = \frac{p}{p_s} = \frac{\overline{p}}{\overline{p}_s} \tag{46}$$

$$\hat{\rho} = \frac{\rho}{\rho_s} = \frac{\overline{\rho}}{\overline{\rho}_s} \tag{47}$$

and

$$\hat{p}_o' = \frac{\dfrac{p_o'}{p_o}}{\left(\dfrac{p_o'}{p_o}\right)_s} = \frac{\overline{p}\left(\dfrac{p_o'}{p}\right)}{\left(\dfrac{p_o'}{p_o}\right)_s} \tag{48}$$

3. PROGRAM NOMENCLATURE

This partial dictionary of symbols used in the main program and subroutines has been divided into seven groups. Two of these groupings are based on the universality of definition (special storage allocation) of the variables; four more are input lists for different parts of the program and the remaining group is a partial list of output designators.

3.1 PROGRAM COMMON VARIABLES, GROUP 1

These symbols have the meanings given below in the main program and the subroutines IFLOW, CFLOW, NUCLE1, NUCLE2, CONDEN, ISEN and NOZZLE.

ASTAR	Throat area (cm^2)
PZERO	Supply pressure ($dyne/cm^2$)
TZERO	Supply temperature (^oK)
RHZERO	Supply density (gm/cm^3)
GAMMA	Ratio of specific heats
MU	Molecular weight (gm/gmol)
CP	Specific heat at constant pressure (dyne*cm/gm*^oK)
L	Current value of the latent heat (dyne*cm/gm)
SIGMA	Current value of the surface tension (dyne/cm)
RHOL	Current value of the liquid density (gm/cm^3)
D	Intermolecular distance in the liquid (cm)
ALPHA	Accomodation coefficient
N	Current value of the index. The main body of the program is concerned with computing the conditions at station N, based on the conditions at stations 0, . . ., N-1.

3.2 PROGRAM COMMON VARIABLES, GROUP 2

All of these variables are vectors of length 500. Except for RADIUS and DELRAD, the N-th entries of these vectors define the flow conditions at the N-th station. These symbols have the meanings given below in the main program and the subroutines CFLOW, NUCLE1, NUCLE2, and CONDEN.

X(I)	Nozzle coordinate at station I (cm)
DELX(I)	By definition DELX(I) = X(I) - X(I-1). The initial entry DELX(0) = DELX is an input parameter.
A(I)	Area at station I (cm^2)
P(I)	Pressure at station I ($dyne/cm^2$)
T(I)	Temperature at station I (^{o}K)
RHO(I)	Density at station I (gm/cm^3)
U(I)	Velocity at station I (cm/sec)
M(I)	Mach number corresponding to U(I)
G(I)	Mass fraction of condensate at station I
DELG(I)	Increase in mass fraction of condensate between stations I-1 and I
NDOT(I)	Number of drops per second formed between stations I-1 and I
RADIUS	When computing the conditions at station N, RADIUS(I) is the radius, at station N-1, of the drops initially formed at station I. Clearly, only entries 0, . . ., N-1 are thus defined when computing the conditions at station N; the N-th entry is computed at station N (cm)
DELRAD	When computing the conditions at station N, DELRAD(I) is the increase in radius of the drops, initially formed at station I, since passing station N-1. When the conditions at station N have been computed, RADIUS(I) must thus be incremented by DELRAD(I) for I = 0, . . ., N-1.

3.3 MAIN PROGRAM INPUT PARAMETERS

These symbols have the meanings given below only in the main program. They do not represent the entire list of input parameters to the main program, but all of them are input variables.

PE	Input parameter used as an initial approximation to the saturation pressure (dyne/cm^2)
XRANGE	Length of the interval, starting at the initial point, in which the flow conditions are to be computed (cm)
MAXG	If the mass fraction of condensate exceeds this value, the calculations are halted
MING	When the mass fraction of condensate exceeds this value, then DELX(N) will be varied so that DELG(N) is greater than MNDELG but less than MXDELG
MNDELG	See MING
MXDELG	See MING
MNDELX	Minimum value that DELX(N) will be given, regardless of the computed DELG(N) (cm)
MXDELX	Maximum value that DELX(N) will be given, regardless of the computed DELG(N). This bound only applies when NDOT(N) is zero (cm)
MXDELX(1)	Maximum value that DELX(N) will be given, regardless of the computed DELG(N). This bound only applies when NDOT(N) is non-zero (cm)
XPOINT	Only those stations with subscripts which are integral multiples of XPOINT will be printed in the three output tables.

3.4 SUBROUTINE VAPOR INPUT PARAMETERS

These symbols have the meanings given below only in the subroutine VAPOR. Three of these names are in the PROGRAM COMMON GROUP 1; however, VAPOR has no communication with PROGRAM COMMON and the meanings are different. These variables must be supplied to VAPOR, as they provide the basis for the entries FSIGMA, FRHOL, FL, FTSAT and LNPSAT.

PTP	The natural log of the pressure in $dyne/cm^2$ at the triple point.
TTP	The inverse of the temperature in ^oK at the triple point.
SIGMA . . . SIGMA (1)	Coefficients of the linear approximation to the surface tension in terms of the drop temperature, $SIGMA + SIGMA(1)*T_{DROP}$.
RHOL . . . RHOL(1)	Coefficients of the linear approximation to the liquid density in terms of the drop temperature, $RHOL + RHOL(1)*T_{DROP}$.
L . . . L(1)	Coefficients of the linear approximation to the latent heat in terms of the vapor temperature, $L + L(1)*T_{VAPOR}$.
PSAT . . . PSAT(4)	Coefficients of the linear approximation to the saturation curve below the triple point, $$1/T = PSAT + PSAT(1)*\ln(P)$$ and of the quadratic approximation to it above the triple point $$1/T = PSAT(2) + PSAT(3)*\ln(P) + PSAT(4)*[\ln(P)]^2.$$

3.5 SUBROUTINE NOZZLE INPUT PARAMETERS

These five symbols have the meanings given below only in NOZZLE. These variables, together with ASTAR, are read by this subroutine and serve to define the nozzle geometry.

XMIN	X co-ordinate of the nozzle inlet (cm)
XMAX	X co-ordinate of the nozzle exit (cm)
INANG	Half-angle of the convergent section of the nozzle (Deg)
OUTANG	Half-angle of the divergent section of the nozzle (Deg)

WEDGE	Switch denoting whether the geometry is for a wedge nozzle (1B) or for a conical nozzle (0B). This variable is of Boolean mode and may assume only the values 1B = true or 0B = false.

3.6 SUBROUTINE CONDEN INPUT PARAMETERS

The symbols given below are defined only in CONDEN. The input parameter EPSLON must always be provided. If EPSLON is zero, then any PROGRAM COMMON symbol is a legal input parameter. If EPSLON is non-zero, then a condensation point search employing the $\epsilon(T)$ criterion is initiated; in this case, these input parameters have the meanings given below.

DELT(1)	Initial step in T from saturation point that will be used in searching for a maximum DELG point (^{o}K)
TRANGE	The search for the maximum DELG point will be confined to the interval T_{SAT} to T_{SAT} - TRANGE. If TRANGE exceeds T_{SAT} the lower end of the interval is taken as zero (^{o}K)
EPS	Condensation is assumed to be started if $\epsilon(T)$ exceeds EPS at some point
EPS(1)	Upper limit of the absolute error in the temperature at the maximum DELG point.
EPS(2)	Upper limit for the relative error of the temperature at the condensation point.

3.7 SUBROUTINE ISEN OUTPUT VARIABLES

Except for the symbol MI, these names are not defined in any part of the program. They are, rather, headings for some of the columns of Table 1. All are computed by the subroutine ISEN, and are returned to the main program via PROGRAM COMMON.

PHAT	Ratio of the computed static pressure to the corresponding isentropic pressure.
THAT	Ratio of the computed static temperature to the corresponding isentropic temperature.
RHOHAT	Ratio of the computed static density to the corresponding isentropic density.
PZERO'HAT	Ratio of the computed total head pressure behind a normal shock wave to the corresponding value for an isentropic flow.
MI	Isentropic Mach number corresponding to the local geometric area ratio.

4. PROGRAM STRUCTURE

Before beginning the general description of the logical structure of the program, a few preliminary remarks are in order. To provide easy communication between the main program and some of the subroutines, many of the variables have been placed in PROGRAM COMMON; this storage assignment function is conceptually the same as FORTRAN COMMON. The function of the 'READ DATA' statement referred to below is explained in the section on the data deck.

Because the development of the program has been essentially experimental in nature, no attempt has been made to provide flow charts. The program has been so organized, however, that the particulars of the various iterations used in the calculations are easily obtainable from the program listing in Appendix A. To a large extent, this is due to the fact that the main program plays the role of an upgraded I/O monitor for a system of computational subroutines. Thus, this section, though intended to be a description only of the main program, makes repeated references to the subroutines and their functions.

4.1 PROGRAM INPUT AND INITIALIZATION

The first action of the main program, hereafter referred to as MAIN, is the reading of the program initialization deck. This deck begins with a set of 72-column comment cards which are continuously read and printed until a card with ENDbbb[1] in columns 1-6 is encountered. Immediately following this, MAIN calls VAPOR which gives a 'READ DATA' statement to obtain the SUBROUTINE VAPOR INPUT PARAMETERS. It should be

---

[1] A small 'b' represents a blank.

noted that these parameters uniquely specify the vapor and, as a consequence, all problems in a given run are based on the expansion of a single pure vapor. This input represents the end of the program initialization deck, and MAIN proceeds to the processing of the first problem of the run.

4.2 PROBLEM INPUT AND INITIALIZATION

The data deck for each problem also begins with a set of 72-column comment cards, in this case terminated by a card with DATAbb in columns 1-6. Following this comment card processing, MAIN gives a 'READ DATA' statement to obtain the PROGRAM COMMON variables, TZERO, PZERO, GAMMA, MU, CP, D and ALPHA, and the entire list of MAIN PROGRAM INPUT PARAMETERS.

After printing these input data, MAIN calls NOZZLE with a first argument of $THROAT$[(2)]. Because of this value for its first argument, this subroutine gives a 'READ DATA' to obtain the PROGRAM COMMON variable ASTAR and the SUBROUTINE NOZZLE INPUT PARAMETERS. NOZZLE then prints these data and does some initialization for its computational entries which are specified by first arguments of $AREA$ or $INVERS$.

4.3 SATURATION POINT COMPUTATION

With these data, MAIN proceeds to the calculation and printing of the conditions at the saturation point. The saturation temperature is computed by performing an iteration on temperature and pressure that leads to the point of intersection of the saturation curve specified by FTSAT and the isentropic expansion curve given by

$$P = \text{PZERO}\left(\frac{T}{\text{TZERO}}\right)^{\text{GAMMA}/(\text{GAMMA}-1)}$$

[(2)] Hollerith arguments to MAD subroutines may be given directly in this form, e.g., NOZZLE. ($THROAT$, X, A.)

The conditions at the saturation point are then supplied by IFLOW and NOZZLE ($INVERS$); IFLOW provides the pressure, density, area, velocity and Mach number corresponding to an isentropic expansion to the saturation temperature, and NOZZLE provides the coordinate corresponding to this area and Mach number. The saturation temperature and rate of mass flow, MDOT = A_{SAT} * RHO_{SAT} * U_{SAT}, then serve as the arguments to CONDEN.

4.4 CALCULATION OF THE INITIAL POINT

The initial point X = X(0) and the associated flow conditions A, P, T, RHO, U, M, RADIUS, NDOT, G and DELG are supplied to MAIN by the subroutine CONDEN. An explanation of how these initial conditions are computed is given in the description of CONDEN and will not be repeated here. It should be noted, however, that CONDEN gives a 'READ DATA' in order to decide how to compute these initial conditions, prints these conditions, and may give an error return to MAIN if it finds 'insufficient' nucleation. If this error return is given, MAIN immediately proceeds to the next problem. Having executed CONDEN with a successful return, MAIN proceeds with the calculation of the flow conditions at stations 1, 2, 3 . . . The flow conditions to be computed at station N consist of:

X(N), DELX(N), A(N),
P(N), T(N), RHO(N), U(N), M(N),
NDOT(N), G(N), DELG(N) and
RADIUS . . . RADIUS(N).

In passing from the conditions at station N-1 to those at station N, the vector elements RADIUS . . . RADIUS(N-1) are incremented by the growth these drops experience between these stations and RADIUS(N) becomes defined. Thus the entire set of conditions at station N-1 are no longer available.

4.5 ITERATIVE CALCULATION OF THE FLOW CONDITIONS

Given the flow conditions at station N-1 and taking DELX(N) = DELX(N-1), the computation of the conditions at X(N) = X(N-1) + DELX(N) is accomplished via an iteration on the value of G(N). Starting with DELG(N-1) as an initial approximation to DELG(N), CFLOW is used to solve the diabatic flow equations for the pressure, temperature, density and velocity corresponding to X(N) and G(N). The initial approximation to U(N) required by CFLOW is 2 * U(N-1) - U(N-2) on the first iteration; thereafter it is supplied automatically by the previous iteration. These values of temperature, pressure and velocity are then used by NUCLE2 to calculate the corresponding DELG(N) and, hence, a new G(N). This alternating use of CFLOW and NUCLE2 is continued until two successive values of G(N) differ by no more than .00001 or until ten iterations have been performed. If this iteration fails to converge, then DELX(N) is halved and the computations are restarted. Consistent failure to converge results in termination of the problem; however, only rarely has this iteration failed and then convergence was obtained after halving DELX(N) only once.

Assuming the iteration has converged, the values of X(N), DELX(N), . . . are accepted as specifying the conditions at station N if any one of the following is true.

1) It was necessary to halve DELX(N) in order to obtain convergence.
2) G(N) does not exceed MING.
3) DELG(N) is greater than MNDELG but less than MXDELG.
4) DELG(N) is less than MNDELG but 2 * DELX(N) exceeds MXDELX or MXDELX(1), whichever applies.
5) DELG(N) exceeds MXDELG but .5 * DELX(N) is less than MNDELX.

The decision to accept or reject DELX(N) as specifying the nozzle coordinate at station N is reached by testing these conditions in the order given. If DELX(N) is rejected, a new value is obtained by halving it when DELG(N) exceeds MXDELG and doubling it when DELG(N) is less than MNDELG. Note that conditions (4) and (5) insure that this new DELX(N) lies in the range specified for it. Having thus obtained a new value for DELX(N), MAIN returns to compute the corresponding flow conditions. Consideration of these five conditions shows that a slightly more sophisticated technique is required, since currently the program can enter an infinite loop by alternately halving and doubling DELX(N). The essential point is that, at present, the program does not remember whether or why it rejected the last value of DELX(N). If a procedure for remembering this information was incorporated, the algorithm would be improved and effective. To date there are no positive indications that this loop has been encountered in our use of the program; however, its occurrence could be forced with proper values for the parameters MNDELG, MXDELG, MNDELX, MXDELX and MXDELX(1).

When MAIN accepts X(N), DELX(N), . . . as specifying the conditions at station N, the internal subroutine IOCTRL is executed. If N is not an integral multiple of XPOINT, this subroutine returns without taking any action. Otherwise, IOCTRL calls on ISEN to compute the SUBROUTINE ISEN OUTPUT VARIABLES, prints the Table 1 output line for station N, computes MDRAD(N) and RHODRP(N) and saves RADIUS(N) in SVRAD(N). These last three quantities are printed in Table 3.

4.6 PROBLEM TERMINATION

Termination of these calculations occurs when any one of the following conditions is found to hold.

1) N exceeds 500.
2) The next point at which the flow conditions are to be calculated exceeds X(0) + XRANGE.

3) The subroutine NOZZLE ($AREA$) indicates that the next point at which the flow conditions are to be calculated is outside the nozzle.

4) The iteration on the diabatic flow equations in CFLOW has failed to converge.

5) The iteration to balance the diabatic flow and nucleation equations has failed to converge.

6) The mass fraction of condensate exceeds MAXG.

The cause of termination and the output Table 1 line for this last station are printed. Finally, Tables 2 and 3 are printed and MAIN continues to the next problem. The output lines, in Tables 2 and 3, for the last station are not printed unless the index of the last point is a multiple of XPOINT. The contents of all three output tables are fully described in the section on I/O structure.

5. STRUCTURE OF THE SUBROUTINES

The following sections provide an explanation of both implicit (PROGRAM COMMON) and explicit arguments, length, transfer vectors and functions of the subroutines coded for this program. Subroutines dealing with I/O are omitted from the transfer vectors. Of those system subroutines listed, all but ZERO are fully explained by their names; this subroutine simply sets its arguments to zero. The formulas underlying the computations performed by CFLOW, CONDEN, NUCLE1 and NUCLE2 have been incorporated, while those for the remainder may be easily obtained from the program listing.

CFLOW

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

PROGRAM COMMON VARIABLES, Group 2.

DELA   Area increment corresponding to N, A(N) - A(N - 1).

LOC   Location to be given control if the iteration fails to converge.

Length: 277 octal

Transfer Vector: FL, SQRT

Purpose:

Compute values of P(N), T(N), RHO(N), U(N) and M(N) that satisfy the diabatic flow equations

$$\frac{DELRHO}{RHO(N)} + \frac{DELU}{U(N)} + \frac{DELA}{A(N)} = 0 \quad , \qquad \text{(Continuity)}$$

$$\frac{DELP}{P(N)} = - \frac{MU * U(N) * DELU}{(1 - G(N)) * R * T(N)} \quad , \qquad \text{(Momentum)}$$

$$\frac{DELP}{P(N)} = \frac{DELRHO}{RHO(N)} + \frac{DELT}{T(N)} - \frac{DELG(N)}{1 - G(N)} \quad , \qquad \text{(State)}$$

$$U(N) * DELU + CP * DELT - L(T(N)) * DELG(N) = 0 \quad , \qquad \text{(Energy)}$$

for the given values of A(N), DELA, G(N) and DELG(N). The symbol R is the universal gas constant. The symbols DELRHO, DELU, . . . denote the differences RHO(N) - RHO(N-1), U(N) - U(N-1), . . . The constants MU and CP are obtained from PROGRAM COMMON. CFLOW uses FL to compute the latent heat L, and leaves the final value corresponding to T(N) in PROGRAM COMMON. An initial approximation to U(N) must be supplied in U(N). Iteration on the value of U(N) is accomplished by solving the energy equation for DELT and, hence, T(N) and then using the equations of continuity, state and momentum to compute the corresponding DELU. The iteration is continued until the relative error between two successive values of U(N) is less than 10^{-6} or until fifty iterates have been computed. This iteration appears to converge quite rapidly, the iteration limit of fifty being much too high. Having obtained values for U(N) and T(N) from this iteration, P(N) is obtained from the momentum equation using P(N-1), RHO(N) from the equation of continuity using RHO(N-1) and

$$M(N) = U(N) * SQRT\left(\frac{MU * T(N)}{R * GAMMA}\right) .$$

Finally, for those interested in recoding this progra , it should be mentioned that convergence problems occurred whenever attempts were made to 'improve' this iteration. Experience indicates that there is a fine line here between a convergent and a non-convergent iteration. For example, some of the attempted 'improvements' did not seem to alter the logic, but were aimed only at reducing the amount of computation per iteration.

CONDEN

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

PROGRAM COMMON VARIABLES, Group 2.

TSAT Computed saturation temperature.

MDOT Rate of mass flow at the saturation point,

$A_{SAT} * RHO_{SAT} * U_{SAT}$.

Length: 1124 octal

Transfer Vector: FL, FRHOL, FSIGMA, IFLOW, NOZZLE, NUCLE1, ZERO

Purpose:

Supply to MAIN an initial point from which the calculations are to be started. The conditions at station zero are specified by X, A, P, T, RHO, U, M, NDOT, RADIUS, G and DELG. In order to permit the introduction of an externally computed initial point CONDEN first sets this point to the saturation conditions with no condensate or foreign particles. It then gives a 'READ DATA' statement which must be supplied with a value for EPSLON.

If EPSLON = 0, the saturation conditions with no condensate or foreign particles and any overrides to the conditions at the initial point due to this 'READ DATA' statement are returned to MAIN.

If a non-zero value of EPSLON is obtained, CONDEN computes either the point of onset of condensation according to the $\epsilon(T)$ criterion or, if this criterion cannot be satisfied, it finds the maximum point of DELG as computed by NUCLE1. This task is accomplished via a technique similar to the half-interval method and is most easily explained as follows.

1) Initialize DELG to zero, T to TSAT and DELT to DELT(1).
2) T = T - DELT.
3) Use IFLOW and NOZZLE to obtain the flow conditions corresponding to an isentropic expansion to temperature T.
4) Save DELG in DELG(1).

5) Use NUCLE1 to compute the critical drop size, the nucleation rate for the volume element specified by A and DELX and the resulting mass fraction of condensate DELG.

6) If DELG is zero, return to step (2), otherwise compute

$$EPSLON = \frac{A}{DELA} * \left(f(M) * \frac{L(T)}{CP * T} - 1 \right) * DELG$$

where

$$f(M) = MAX \left(1, \frac{GAMMA - 1/M^2}{GAMMA - 1} \right) .$$

7) If EPSLON exceeds EPS, the onset of condensation occurs between temperatures T + DELT and T and is obtained via a half-interval technique which begins at step (9). If EPSLON does not exceed EPS, continue the maximum DELG point search with step (8).

8) If DELG exceeds DELG(1), return to step (2), otherwise the maximum DELG point has been passed. Set T = T + 2 * DELT and recompute the flow conditions (including DELG) for this temperature. If DELT is less than or equal to EPS(1) print these conditions as the maximum DELG point and give an error return to MAIN, otherwise divide DELT by 10 and return to step (2).

9) Initialize LOW to T and HIGH to T + DELT.

10) Set T = .5 * (HIGH + LOW) and duplicate the calculations of steps (3), (5) and (6) to obtain EPSLON.

11) If EPSLON exceeds EPS set LOW to T, otherwise set HIGH to T.

12) If the absolute value of (1 - LOW/HIGH) exceeds EPS(2) return to step (10). When this convergence criterion becomes satisfied, accept the temperature T and the associated flow conditions as those at the zeroth station, print them and give a successful return to MAIN.

IFLOW

Arguments:

PROGRAM COMMON VARIABLES, Group 1

A Area corresponding to an isentropic expansion to temperature T.

M Mach number corresponding to an isentropic expansion to temperature T.

P Pressure corresponding to an isentropic expansion to temperature T.

RHO Density corresponding to an isentropic expansion to temperature T.

T Temperature for which the isentropic conditions are desired.

U Velocity corresponding to an isentropic expansion to temperature T.

Length: 167 octal

Transfer Vector: SQRT

Purpose:

Compute the area, pressure, density, velocity and Mach number corresponding to an isentropic expansion to the given temperature T.

ISEN

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

A Area

P Pressure

T Temperature

RHO Density

M Mach number

Length: 503 octal

Purpose:

ISEN is called from the internal subroutine IOCTRL of MAIN that controls the printing of the first output table. This subroutine computes PHAT, THAT, RHOHAT, PZERO'HAT and MI on the basis of the five arguments and returns these values via the PROGRAM COMMON vector ZQ, ZQ(1) . . . ZQ(5) respectively. The first four of these quantities are easily computed from the value of MI, which is obtained via a modified half-interval technique so that the area ratio corresponding to MI lies in the interval (A/STAR - .001, A/ASTAR + .001).

NOZZLE

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

A1 — Hollerith valued switch with the values 'THROAT', 'INVERS' or 'AREA'. This argument actually provides three entry points for NOZZLE. Throughout this report this subroutine is referred to both by name and by the value of the first argument enclosed in dollar signs.

A2 — Unused when A1 = $THROAT$. Nozzle coordinate for which the area is desired when A1 = $AREA$. When A1 = $INVERS$, A2 must be the Mach number on entry and will be returned as the nozzle coordinate with area A3.

A3 — Location for returning the throat area when = $THROAT$, the area at A2 when A1 = $AREA$, and when A1 = $INVERS$ it is the area for which the nozzle coordinate is desired.

Length: 541 octal

Transfer Vector: COS, SIN, SQRT

Purpose:

The origin of the nozzle coordinate is the throat and the coordinate increases positively in the direction of flow. In this system, the X-coordinate of the intake is always negative and that of the exit always positive. Since the meaning of the throat area is basically different when using a wedge rather than conical nozzle, the type of nozzle is given as a Boolean input variable.

A1 = $THROAT$. Compute the area at the X-coordinate in A2 and return it in A3. If A2 is not in the interval (XMIN, XMAX), a zero area is returned.

A1 = $INVERS$. Compute and return in A2 the nozzle coordinate corresponding to an area A3 and Mach number A2. If A3 is less than ASTAR, A2 is set to zero and A2(1) to 1. The nozzle coordinate will be computed in the diverging portion of the nozzle if the flow is supersonic and in the converging portion if it is subsonic. If the computed nozzle coordinate does not lie inside the nozzle, A2(1) is returned as zero; otherwise it is given the value 1.

NUCLE1

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

PROGRAM COMMON VARIABLES, Group 2.

MDOT Rate of mass flow at the saturation point.

Length: 252 octal

Transfer Vector: ELOG, EXP, FSPILL, LNPSAT, RSPILL, SQRT, ZERO

Purpose:

Using the current values of D, SIGMA, RHOL, T(N), P(N), A(N) and DELX(N) compute: the critical drop size, a value for surface tension corrected for the intermolecular distance, the nucleation rate for the volume element specified by A(N) and DELX(N), and the resulting mass fraction of condensate due to the formation of these drops. The nucleation quantities are returned in RADIUS(N), NDOT(N) and DELG(N) and the corrected value of SIGMA replaces the old value. The values of SIGMA and RHOL should correspond to P(N) via FSIGMA and FRHOL. The subroutine FSPILL is used to control floating-point traps. This subroutine sets underflows to zero and transfers to the argument location if an overflow occurs. All quantities are returned as zero if an overflow occurs, if RADIUS(N) exceeds 10^{-6}, or if NDOT(N) is less than 1. The computations are based on the following equations.

$$KBAR = \frac{2 * MU}{RHOL * T(N) * R * (\ln P(N) - \ln P_{SAT})}$$

$$RADIUS(N) = KBAR * SIGMA - D$$

$$SIGMA = SIGMA - D/KBAR$$

$$TEXP = -\frac{4\pi}{3} * \frac{SIGMA * RADIUS(N)^2}{K * T(N)}$$

$$NDOT(N) = \left(\frac{P(N)}{K * T(N)}\right)^2 * \frac{1}{RHOL} * \sqrt{\frac{2}{\pi} * \frac{SIGMA * MU}{N_A}}$$

$$* \exp(TEXP) * A(N) * DELX(N)$$

$$DELG(N) = \frac{4\pi}{3} * RHOL * NDOT(N) * RADIUS(N)^3/MDOT$$

where R is the universal gas constant, K is Boltzmann's constant, N_A is Avogadro's number, and $\ln P_{SAT}$ is computed by LNPSAT and is the natural log of the saturation pressure corresponding to T(N).

NUCLE2

Arguments:

PROGRAM COMMON VARIABLES, Group 1.

PROGRAM COMMON VARIABLES, Group 2.

MDOT Rate of mass flow at the saturation point.

Length: 157 octal

Transfer Vector: FTSAT, NUCLE1, SQRT

Purpose:

Using the current values of ALPHA, D, L, RHOL, SIGMA, P(N), T(N) and U(N) compute the mass fraction of condensate formed in the volume element specified by A(N) and DELX(N) due both to formation of 'new' drops and growth of 'old' drops. The values of RHOL and SIGMA should correspond to P(N), via FRHOL and FSIGMA, and L should correspond to T(N) via FL. This subroutine first calls NUCLE1 to obtain RADIUS(N), NDOT(N) and the mass fraction of condensate due to the formation of these 'new' drops. The radial increment of the drops initially formed at station I (DELRAD(I)) is not a function of the radius of these drops at station N - 1 (RADIUS (I)); the radial increments for all 'old' drops is the same. The equation for this uniform increment is

$$\left(\frac{2 * K * N_A}{\pi * MU * T(N)}\right)^{1/2} * \frac{ALPHA * P(N) * DELX(N)}{L * RHOL * U(N)} * (T_{DROP} - T)$$

where T_{DROP} is the saturation temperature corresponding to P(N) as computed by FTSAT. The mass fraction of condensate due to this growth of the 'old' drops is

$$\frac{4\pi * RHOL}{MDOT} * \sum_{I=0}^{N-1} NDOT(I) * DELRAD(I) * RADIUS(I)^2 \quad .$$

The value of DELG(N) returned is the sum of the two mass fractions of condensate.

VAPOR
FL
FRHOL
PSIGMA
FTSAT
LNPSAT

Argument:

ARG The entry VAPOR uses no argument. For the entries FL and LNPSAT, the argument is a temperature; for FRHOL, FSIGMA and FTSAT it is a pressure.

Length: 637 octal

Transfer Vector: ELOG, SQRT

Purpose:

The entry VAPOR is used to supply, via a 'READ DATA' statement, the linear and quadratic approximations that form the computational basis for the other entries, i. e., the SUBROUTINE VAPOR INPUT PARAMETERS.

The entry FL computes the latent heat corresponding to the temperature ARG from the equation L + L(1) * ARG.

The entry FRHOL computes the liquid density from the linear approximation RHOL + RHOL(1) * T_{DROP}, where T_{DROP} is the saturation temperature corresponding to the pressure ARG.

The entry FSIGMA computes the surface tension from the linear approximation SIGMA + SIGMA * T_{DROP}, where T_{DROP} is the saturation temperature corresponding to the pressure ARG.

The entry FTSAT computes the saturation temperature corresponding to the pressure ARG from the linear approximation.

$$1/T = PSAT + PSAT(1) * \ln P, \qquad (L)$$

if ln ARG is less than or equal to PTP, and from the quadratic approximation

$$1/T = PSAT(2) + \ln P * PSAT(3) + \ln P^2 * PSAT(4) \qquad (Q)$$

otherwise.

The entry LNPSAT computes the natural log of the saturation pressure corresponding to the temperature ARG. When the inverse of ARG exceeds TTP the result is obtained by solving (L) for ln P. Otherwise, the roots of the quadratic (Q) must be computed. The proper root is chosen by taking the smallest root that exceeds PTP.

6. I/O STRUCTURE

The following section describes how the input and output were implemented and organized in the program. The inflexibility of this aspect of the original program, Ref. 1, was one of the chief motivations for the rather drastic changes that have been made. The two most important changes concerning the I/O abilities of the program are reflected in greater flexibility in choosing the initial point and better organization in the output.

6.1 INPUT STRUCTURE

Since the M. A. D. 'READ DATA' statement is used for obtaining the numerical input, an understanding of the data deck requires a knowledge of the function of this statement. The description of this statement given in the MAD manual is:

"This statement causes information to be read from cards; no list of variable names or format specification is necessary. The values to be read and the variable names are punched in the data cards in a sequence of fields of the form:

$$V_1 = n_1, V_2 = n_2, \ldots, V_K = n_K*$$

The $V_1, \ldots, V_K$ are the variable names and $n_1, \ldots, n_K$ the corresponding values. Reading is continued from card to card until the terminating mark * is encountered. Only the first 72 columns of a card may be used for data; . . . However, as a convenience, the end of the card is treated as an implied comma . . .

For convenience in entering values of array elements, it is possible to designate only one variable name and have successive numbers, written without names, interpreted as the consecutive values of the array, i.e.,

$$V(J) = n_1, n_2, \ldots, n_K$$

would be the same as

$$V(J) = n_1, \; V(J+1) = n_2, \; \ldots, \; V(J+K-1) = n_K \quad ."$$

The use of this statement for input allows considerable flexibility, since no list or format is necessary. Thus for example, if a number of problems with the same nozzle geometry are to be run the nozzle parameters need only be read the first time. The remaining data sets for this subroutine could then consist only of a card with an * in column 1. This situation is in fact true for all inputs, except EPSLON which is read by the subroutine CONDEN; that is, once they are read they maintain that value for the remainder of the run unless specifically changed by a subsequent 'READ DATA'.

Finally, it should be remarked that many of the input parameters are assigned to storage in PROGRAM COMMON. Those variables assigned to PROGRAM COMMON have been divided into two groups. Some of the subroutines have no access to these variables, some have access only to the first group and some have access to both groups. In those subroutines that have access to some of PROGRAM COMMON and that also give 'READ DATA' statements, care should be taken as to what is punched on the data cards.

An example of an input deck and the resulting output have been included in Appendix B. The input deck was listed on an IBM 407. The problem required 9.5 seconds of execution time on the IBM 7090.

The input deck for a run consists of the program initialization deck followed by any number of problem decks. The program initialization deck structure is as follows:

(1A) Any number of 72-column comment cards. For those familiar with FORTRAN, each card is read according to a format of 12A6 and immediately printed with a double space.

(1B) A single card with 'ENDbbb', where 'b' denotes a blank, punched in columns 1-6. This card is used to terminate the comment card deck.

(2) A deck of cards containing the SUBROUTINE VAPOR INPUT PARAMETERS. These parameters are obtained by a 'READ DATA' statement. The variables that must be supplied at this point are PTP, TTP, PSAT . . . PSAT(4), SIGMA . . . SIGMA(1), RHOL . . . RHOL(1) and L . . . L(1). Only these variables may be given values in this deck.

A program deck consists of essentially four parts.

(1A) Any number of 72-column comment cards. These are processed exactly as the comment card deck in the program initialization deck. They are double-spaced and are immediately preceded by the heading 'BEGINNING OF PROBLEM NUMBER ___', which always begins a new page.

(1B) A single card with 'DATAbb' punched in columns 1-6. This card is used to terminate the comment card deck.

(2) A deck of cards containing the PROGRAM COMMON parameters TZERO, PZERO, GAMMA, MU, CP, D and ALPHA and the entire set of MAIN PROGRAM INPUT PARAMETERS. These variables are obtained by a 'READ DATA' statement. If any of these expected parameters do not appear, then they maintain their current value in the computer. The following is a list of the parameters that are loaded initially with the program and, hence, need never appear in the data deck unless these values are unsatisfactory:

XPOINT = 1	MNDELX = .001
MNDELG = .0005	MXDELX = 1.
MXDELG = .0015	MXDELX(1) = .1

Note that XPOINT is an integer and hence its value must be given without a decimal point. The remaining parameters are not preset by the program and must appear at least in problem 1. Since all problems of a run must deal with the same vapor, normally GAMMA, MU, CP, D and ALPHA will appear only in the first problem.

(3) A deck of cards containing the PROGRAM COMMON parameter ASTAR and the SUBROUTINE NOZZLE INPUT PARAMETERS, again obtained by a 'READ DATA' statement. Only these parameters should be included. Note that the two half-angles are specified in degrees, that XMIN is negative and that XMAX is positive. Since the parameter WEDGE is of Boolean mode, it has only two possible values; i.e., OB = true and 1B = false.

(4) The final deck of cards is obtained via a 'READ DATA' statement in the subroutine CONDEN and must supply a value for EPSLON. If EPSLON is given a non-zero value, the only other parameters that should occur are DELT(1), TRANGE and EPS . . . EPS(2). These five parameters are preset, however, and thus need be supplied only if the values

DELT(1) = 5.0	EPS = .001
TRANGE = 100.0	EPS(1) = .01
	EPS(2) = .00001 ,

are unsatisfactory. If EPSLON is given a value of zero, however, then any of the parameters specifying the flow conditions at the initial point are legitimate inputs; i.e., X, DELX, A, P, T, RHO, U, M, RADIUS, NDOT, G, or DELG. If not supplied, these values will correspond to those at the saturation point.

6.2 OUTPUT STRUCTURE

The output structure is most easily understood by studying the example output given in Appendix C. The first page for each run consists of the comment cards in the program initialization deck and a list of the input supplied to VAPOR.

Each problem is headed by the comment 'BEGINNING OF PROBLEM NUMBER ___' at the top of the first page, the program supplying the problem number. This heading is followed by any comment cards for the problem. Following these comments is a complete list of the program parameters read by MAIN and NOZZLE ($THROAT$), the saturation conditions printed by MAIN, and finally a comment about and the conditions at the initial point, both printed by CONDEN. The remainder of the output consists of the three output tables: the first one printed while the calculations are being done and the last two after the problem has been terminated. Each station for which the index is an integral multiple of XPOINT produces one line of output in each of the three tables. Each page of a table consists of a heading for the columns followed by at most 27 double-spaced lines of output.

The headings for the columns of TABLE 1 are:

X	Nozzle coordinate. (cm)
PHAT	Ratio of the computed static pressure to the corresponding isentropic pressure.
THAT	Ratio of the computed static temperature to the corresponding isentropic temperature.
RHOHAT	Ratio of the computed static density to the corresponding isentropic density.
PZERO'HAT	Ratio of the computed total head pressure behind a normal shock wave to the corresponding value for an isentropic flow.

M	Computed Mach number.
MI	Isentropic Mach number for the corresponding area ratio.
U	Computed velocity. (cm/sec)
G	Computed mass fraction of condensate multiplied by 100. This scale factor means that G actually represents the percent condensate.

The headings for the columns of TABLE 2 are:

X	Nozzle coordinate. (cm)
P	Computed static pressure. (dyne/cm^2)
PBAR	Ratio of the computed static pressure to the supply pressure.
T	Computed static temperature. (^oK)
TBAR	Ratio of the computed static temperature to the supply temperature.
RHO	Computed static density. (gm/cm^3)
RHOBAR	Ratio of the computed static density to the supply density.
A	Computed area. (cm^2)
ABAR	Ratio of the computed area to the throat area, i. e., the geometric area ratio.

The headings for the columns of TABLE 3 are:

X	Nozzle coordinate. (cm)
DELX	Nozzle coordinate increment since the last station. (cm)
G	Same as the last column of TABLE 1.
DELG	Increase in mass fraction of condensate since the last station times 100.

RADIUS	Critical drop size at this station. (cm)
NDOT	Number of drops per second formed between the last station and this one.
RHODRP	The number density of the drops. (drops/cm^3)
MEAN RADIUS	The arithmetic average radius of all drops in the flow. (cm)

7. REFERENCES

1. Griffin, J. L., "Digital Computer Analysis of Condensation in Highly Expanded Flows", (Univ. of Mich.) USAF Aerospace Research Labs, OAR, Report ARL 63-206, November 1963.

2. Sivier, K. R., "Digital Computer Studies of Condensation in Expanding One-Component Flows", (Univ. of Mich.) in preparation, to be published as a USAF Aerospace Research Labs., OAR, report.

3. Arden, B., Galler, B., Graham, R., "The Michigan Algorithm Decoder," Univ. of Mich. Press, November, 1963.

APPENDIX A

PROGRAM LISTING

```
MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...


                     AERONAUTICAL ENGINEERING
                   NOZZLE CONDENSATION RESEARCH

                           MAIN PROGRAM

             PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                     *001
            1               GAMMA,MU,CP,L,SIGMA,RHOL,                     *001
            2               D,B,C,ALPHA,                                  *001
            3               N,PPOINT,LINENO,PERPG,ZQ                      *001
             DIMENSION ZQ(10)                                             *002
             INTEGER N,PPOINT,LINENO,PERPG                                *003
             PROGRAM COMMON X,DELX,A,P,T,RHO,U,M,                         *004
            1                G,DELG,RADIUS,NDOT,DELRAD                    *004
             DIMENSION X(500),DELX(500),A(500),                           *005
            1          P(500),T(500),RHO(500),U(500),M(500),              *005
            2          G(500),DELG(500),RADIUS(500),NDOT(500),DELRAD(500) *005
             DIMENSION SVRAD(500),RHODRP(500),MDRAD(500)                  *006

            PROGRAM INPUT AND INITIALIZATION

             PRINT COMMENT $2NOZZLE CONDENSATION RESEARCH PROGRAM$        *007
PCMNT        READ FORMAT VVA,CMNT(1)...CMNT(12)                           *008
             WHENEVER CMNT(1).E.$END   $, TRANSFER TO PCMNT1              *009
             PRINT FORMAT VVB,CMNT(1)...CMNT(12)                          *010
             TRANSFER TO PCMNT                                            *011
PCMNT1       PROBNO=0                                                     *012
             VAPOR.(Z)                                                    *013

            PROBLEM INPUT AND INITIALIZATION

START        PROBNO=PROBNO+1                                              *014
             READ FORMAT VVA,CMNT(1)...CMNT(12)                           *015
             PRINT FORMAT VVC,PROBNO                                      *016
PCMNT2       WHENEVER CMNT(1).E.$DATA  $, TRANSFER TO START1              *017
             PRINT FORMAT VVB, CMNT(1)...CMNT(12)                         *018
             READ FORMAT VVA,CMNT(1)...CMNT(12)                           *019
             TRANSFER TO PCMNT2                                           *020
START1       READ DATA                                                    *021
             RHZERO=MU/TZERO*PZERO/8.314E+07                              *022

            THIS DATA SHOULD INCLUDE

             PZERO  = SUPPLY PRESSURE (DYNE/CM**2)
             TZERO  = SUPPLY TEMPERATURE (DEG. K)
             PE     = ISENTROPIC EXIT PRESSURE. THIS IS USED AS AN
                      INITIAL APPROXIMATION TO THE SATURATION PRESSURE.

             GAMMA  = RATIO OF THE SPECIFIC HEATS
             MU     = MOLECULAR WEIGHT OF THE VAPOR
             CP     = SPECIFIC HEAT AT CONSTANT PRESSURE

             D      = INTERMOLECULAR DISTANCE IN THE LIQUID
             ALPHA  = ACCOMODATION COEFFICIENT
```

```
 DELX    = IF X(0) IS THE FIRST POINT THEN X(1)=X(0)+DELX.
           THIS SAME DELX WILL BE USED TO OBTAIN X(K+1) FROM
           X(K) AS LONG AS THE OTHER PROGRAM PARAMETERS DO NOT
           INDUCE A CHANGE FOR DELX(K).
 XRANGE  = LENGTH OF INTERVAL, STARTING AT THE CONDENSATION
           POINT, IN WHICH THE FLOW CONDITIONS WILL BE
           COMPUTED.
  MAXG   = STOP COMPUTATION AT POINT WHERE G EXCEEDS MAXG.
  MING   = WHEN G EXCEEDS MING THEN START VARIATION OF DELX.


THIS DATA MAY ALSO INCLUDE

 XPOINT  = THE INITIAL POINT IS ASSIGNED A SUBSCRIPT OF
           ZERO, ONLY THOSE POINTS WITH SUBSCRIPTS XPOINT,
           2*XPOINT,... WILL BE PRINTED.
 MNDELG  = IF FOR STEP DELX FROM SOME POINT DELG IS LESS THAN
           MNDELG THEN DELX IS DOUBLED AND THE POINT IS RE-
           CALCULATED WITH THIS NEW DELX.
 MXDELG  = IF FOR STEP DELX FROM SOME POINT DELG IS GREATER
           THAN MXDELG THEN DELX IS HALVED AND THE POINT IS
           RECALCULATED WITH THIS NEW DELX.
 MNDELX  = MINIMUM VALUE THAT WILL BE GIVEN TO DELX REGARD-
           LESS OF THE DELG PRODUCED
 MXDELX  = MAXIMUM VALUE THAT WILL BE GIVEN TO DELX REGARD-
           LESS OF THE DELG PRODUCED IF NDOT=0 -- OTHERWISE
           MXDELX(1) IS THE MAXIMUM ALLOWABLE VALUE.

NOTE THAT IF THESE PARAMETERS ARE ADJUSTED BY THE READ DATA
STATEMENT THAT THEY WILL MAINTAIN THE ADJUSTED VALUE UNTIL
CHANGED BY ANOTHER READ DATA STATEMENT EXECUTION.

           PRINT COMMENT $OCOMPLETE LIST OF PROGRAM PARAMETERS FOR THIS     *023
          1 PROBLEM.$                                                        *023
           PRINT FORMAT VIA,PZERO,TZERO,RHZERO,PE                            *024
           PRINT FORMAT VIB,GAMMA,MU,CP                                      *025
           PRINT FORMAT VIC,D,ALPHA                                          *026
           PRINT FORMAT VID,DELX,MNDELX,MXDELX,MXDELX(1)                     *027
           PRINT FORMAT VIE,MNDELG,MXDELG,MING                               *028
           PRINT FORMAT VIF,XRANGE,MAXG                                      *029
           EXECUTE NOZZLE.($THROAT$,0.,ASTAR)                                *030

CALCULATE A SATURATION POINT BY MATCHING THE ISENTROPIC AND
SATURATION CURVES.

           P=PE                                                              *031
           Z=GAMMA/(GAMMA-1.)                                                *032
S1         T=FTSAT.(P)                                                       *033
           PSAT=PZERO*( (T/TZERO) .P. Z )                                    *034
           TEST=.ABS.(1.-PSAT/P)-.001                                        *035
           WHENEVER TEST.L.0.,TRANSFER TO S2                                 *036
           P=P+.5*(PSAT-P)                                                   *037
           TRANSFER TO S1                                                    *038
S2         TSAT=T                                                            *039
           EXECUTE IFLOW.(ASAT,MSAT,PSAT,RHOSAT,TSAT,USAT)                   *040
           Z=MSAT                                                            *041
           EXECUTE NOZZLE.($INVERS$,Z,ASAT)                                  *042
S3         XSAT=Z                                                            *043
           MDOT=ASAT*RHOSAT*USAT                                             *044
```

```
           PRINT COMMENT $OSATURATION POINT SPECIFICATIONS --$            *045
           PRINT FORMAT VIH,PSAT,TSAT,RHOSAT                              *046
           PRINT FORMAT VII,MSAT,USAT                                     *047
           PRINT FORMAT VIJ,XSAT,ASAT                                     *048
           PRINT FORMAT VIK,MDOT                                          *049
           WHENEVER Z(1).NE.1.                                            *050
           PRINT COMMENT $4     THE SATURATION POINT FOUND IS NOT INSIDE  *051
         1  THE NOZZLE, THE PROGRAM WILL CONTINUE.$                       *051
           TRANSFER TO START                                              *052
           END OF CONDITIONAL                                             *053

          COMPUTE AN INITIAL POINT VIA SUBROUTINE CONDEN

           N=0                                                            *054
           Z=CONDEN. (TSAT,MDOT)                                          *055
           WHENEVER Z.E.0., TRANSFER TO START                             *056
           XLIMIT=X+XRANGE                                                *057
           EXECUTE IOCTRL.                                                *058

                     COMPUTE POINT N GIVEN POINT N-1

S29        ITER(1)=1                                                      *059
           N=N+1                                                          *060
           WHENEVER N.G.5CO                                               *061
           PRINT COMMENT $OTHE NUMBER OF MESH POINTS HAS REACHED THE PRO  *062
         1 GRAMED UPPER LIMIT.$                                           *062
           N=N-1                                                          *063
           TRANSFER TO END                                                *064
           END OF CONDITIONAL                                             *065
           DELX(N)=DELX(N-1)                                              *066

          RESTART POINT FOR COMPUTING POINT N IF DELX(N) IS ALTERED
          FROM ITS INITIAL VALUE OF DELX(N-1)

S30        X(N)=X(N-1)+DELX(N)                                            *067
           WHENEVER X(N).G.XLIMIT                                         *068
           PRINT COMMENT $OTHE NEXT MESH POINT WOULD EXCEED THE RANGE DE  *069
         1 SIRED.$                                                        *069
           N=N-1                                                          *070
           TRANSFER TO END                                                *071
           END OF CONDITIONAL                                             *072
           NOZZLE.($AREA$,X(N),A(N))                                      *073
           WHENEVER A(N).E.0.                                             *074
           PRINT COMMENT $ONOZZLE SUBROUTINE ERROR INDICATION - MESH POI  *075
         1 NT DOES NOT LIE WITHIN THE NOZZLE.$                            *075
           N=N-1                                                          *076
           TRANSFER TO END                                                *077
           END OF CONDITIONAL                                             *078
           DELA=A(N)-A(N-1)                                               *079
           DELG(N)=DELG(N-1)                                              *080
           G(N)=G(N-1)+DELG(N)                                            *081
           WHENEVER N.G.1                                                 *082
           DELU=U(N-1)-U(N-2)                                             *083
           OTHERWISE                                                      *084
           DELU=.CC01*U                                                   *085
           END OF CONDITIONAL                                             *086
           U(N)=U(N-1)+DELU                                               *087
           ITER=1                                                         *088

          BASIC ITERATION TO BALANCE NUCLEATION AND DIABATIC FLOW EQNS.
```

```
S31       CFLOW.(DELA,END)                                              *089
          RHOL=FRHOL.(P(N))                                             *090
          SIGMA=FSIGMA.(P(N))                                           *091
          TEST=G(N)                                                     *092
          NUCLE2.(MDOT)                                                 *093
          WHENEVER ITER.E.1                                             *094
          WHENEVER (NDOT(N).G.1.) .AND. (DELX(N).G.MXDELX(1))           *095
          DELX(N)=MXDELX(1)                                             *096
          TRANSFER TO S30                                               *097
          END OF CONDITIONAL                                            *098
          END OF CONDITIONAL                                            *099
          G(N)=G(N-1)+DELG(N)                                           *100
          TEST=.ABS.(G(N)-TEST)                                         *101
          WHENEVER TEST.L. .00001, TRANSFER TO S32                      *102
          ITER=ITER+1                                                   *103
          WHENEVER ITER.LE.GITER, TRANSFER TO S31                       *104
          ITER(1)=ITER(1)+1                                             *105
          DELX(N)=.5*DELX(N)                                            *106
          WHENEVER ITER(1).LE.5, TRANSFER TO S30                        *107
          PRINT FORMAT VVD,X(N),PROBNO                                  *108
          TRANSFER TO END                                               *109

         VARIATION OF DELX(N) IF THE CALCULATED POINT N SHOWS TOO MUCH
         OR TOO LITTLE CONDENSATION SINCE X(N-1)

S32       WHENEVER (G(N).L.MING) .OR. (ITER(1).G.1), TRANSFER TO S33    *110
          I=0                                                           *111
          WHENEVER NDOT(N).G.1., I=1                                    *112
          WHENEVER DELG(N).L.MNDELG                                     *113
          TEST=2.*DELX(N)                                               *114
          WHENEVER TEST.G.MXDELX(I), TRANSFER TO S33                    *115
          OR WHENEVER DELG(N).G.MXDELG                                  *116
          TEST=.5*DELX(N)                                               *117
          WHENEVER TEST.L.MNDELX, TRANSFER TO S33                       *118
          OTHERWISE                                                     *119
          TRANSFER TO S33                                               *120
          END OF CONDITIONAL                                            *121
          DELX(N)=TEST                                                  *122
          TRANSFER TO S30                                               *123

         THE CURRENT VALUES AT POINT N HAVE BEEN ACCEPTED, ADJUST
         RADIUS(0)...RADIUS(N-1) FOR GROWTH SINCE X(N-1)

S33       I=0                                                           *124
S34       RADIUS(I)=RADIUS(I)+DELRAD(I)                                 *125
          I=I+1                                                         *126
          WHENEVER I.L.N, TRANSFER TO S34                               *127
          WHENEVER G(N).G.MAXG                                          *128
          PRINT FORMAT VVE,X(N)                                         *129
          TRANSFER TO END                                               *130
          END OF CONDITIONAL                                            *131
          EXECUTE IOCTRL.                                               *132
          TRANSFER TO S29                                               *133

         THE COMPUTATIONS HAVE IN SOME WAY BEEN TERMINATED. PRINT THE
         LAST TWO TABLES AND CONTINUE ON TO THE NEXT PROBLEM.

END       PPOINT=N                                                      *134
          PRINT FORMAT VVF,PROBNO                                       *135
```

```
            EXECUTE IOCTRL.                                                         *136

           TABLE 2 -- X,P,T,RHO,A AND THE ASSOCIATED BAR QUANTITIES

            I=0                                                                     *137
            PRINT FORMAT VVR                                                        *138
            LINENO=1                                                                *139
IOF1        PBAR=P(I)/PZERO                                                         *140
            TBAR=T(I)/TZERO                                                         *141
            RHOBAR=RHO(I)/RHZERO                                                    *142
            ABAR=A(I)/ASTAR                                                         *143
            PRINT FORMAT VVS,X(I),P(I),PBAR,T(I),TBAR,RHO(I),RHOBAR,A(I),ABAR       *144
            LINENO=LINENO+1                                                         *145
            WHENEVER LINENO.G.PERPG                                                 *146
            PRINT FORMAT VVR                                                        *147
            LINENO=1                                                                *148
            END OF CONDITIONAL                                                      *149
            I=I+XPOINT                                                              *150
            WHENEVER I.LE.N, TRANSFER TO IOF1                                       *151

           TABLE 3 -- X,DELX AND NUCLEATION INFORMATION

            I=0                                                                     *152
            PRINT FORMAT VVT                                                        *153
            LINENO=1                                                                *154
IOF2        PRINT FORMAT VVU,X(I),DELX(I),G(I),DELG(I),SVRAD(I),NDOT(I),RHODRP(I),  *155
          1 MDRAD(I)                                                                *155
            LINENO=LINENO+1                                                         *156
            WHENEVER LINENO.G.PERPG                                                 *157
            PRINT FORMAT VVT                                                        *158
            LINENO=1                                                                *159
            END OF CONDITIONAL                                                      *160
            I=I+XPOINT                                                              *161
            WHENEVER I.LE.N, TRANSFER TO IOF2                                       *162
            TRANSFER TO START                                                       *163

           THIS INTERNAL FUNCTION MONITORS THE PRINTING FOR TABLE 1 AND
           -- SAVES IN SVRAD(N) THE INITIAL RADIUS OF DROPS FORMED AT
           X(N) -- COMPUTES MDRAD(N), THE MEAN DROP RADIUS AT X(N) --
           COMPUTES RHODRP(N), THE DROP DENSITY AT X(N).

            INTERNAL FUNCTION                                                       *164
            ENTRY TO IOCTRL.                                                        *165
            WHENEVER N.G.0, TRANSFER TO IOC1                                        *166
            PPOINT=0                                                                *167
            TRANSFER TO IOC2                                                        *168
IOC1        WHENEVER LINENO.G.PERPG                                                 *169
IOC2        LINENO=1                                                                *170
            PRINT FORMAT VVP                                                        *171
            END OF CONDITIONAL                                                      *172
            WHENEVER N.E.PPOINT                                                     *173
            ISEN.(A(N),P(N),T(N),RHO(N),M(N))                                       *174
            PRINT FORMAT VVQ,X(N),ZQ(1)...ZQ(4),M(N),ZQ(5),U(N),G(N)                *175
            ZERO.(ZQ...ZQ(3))                                                       *176
            THROUGH IOC3, FOR I=0,1,I.G.N                                           *177
            ZQ=ZQ+NDOT(I)                                                           *178
            ZQ(4)=NDOT(I)                                                           *179
            ZQ(1)=ZQ(1)+ZQ(4)                                                       *180
IOC3        ZQ(2)=ZQ(2)+ZQ(4)*RADIUS(I)                                             *181
            RHODRP(N)=ZQ/A(N)/U(N)                                                  *182
```

```
      MDRAD(N)=ZQ(2)/ZQ(1)                                                  *183
      SVRAD(N)=RADIUS(N)                                                    *184
      PPOINT=N+XPOINT                                                       *185
      LINENO=LINENO+1                                                       *186
      END OF CONDITIONAL                                                    *187
      FUNCTION RETURN                                                       *188
      END OF FUNCTION                                                       *189

      DIMENSION CMNT(13)                                                    *190
      DIMENSION Z(10),ITER(2)                                               *191
      INTEGER ITER                                                          *192
      INTEGER CMNT,PROBNO,I                                                 *193
      VECTOR VALUES XPOINT = 1                                              *194
      VECTOR VALUES MNDELX = .001                                           *195
      VECTOR VALUES MXDELX = 1.,.1                                          *196
      VECTOR VALUES MNDELG = .0005                                          *197
      VECTOR VALUES MXDELG = .0015                                          *198
      VECTOR VALUES LITER = 10.                                             *199
      VECTOR VALUES GITER = 10.                                             *200
      VECTOR VALUES PERPG = 28                                              *201
      VECTOR VALUES VVA =$12C6*$                                            *202
      VECTOR VALUES VVB = $1H0,S6,12C6*$                                    *203
      VECTOR VALUES VVC = $1H1,S40,H*BEGINNING OF PROBLEM NUMBER*,I3*$      *204
      VECTOR VALUES VVD=$1H0,H*ITERATION FOR MESH POINT*,F9.5,H* HAS FAILED T *205
    1 O CONVERGE. THIS TERMINATES PROBLEM NUMBER*,I3*$                      *205
      VECTOR VALUES VVE=$1H0,H*PERCENT CONDENSATE EXCEEDS THE UPPER LIMIT 'MA *206
    1 XG' AT MESH POINT*,F9.5*$                                             *206
      VECTOR VALUES VVF=$1H0,H*FINAL OUTPUT IN TABLE 1 FOR PROBLEM*,I3*$    *207
      VECTOR VALUES VVP=$1H1,S4,1HX,S12,4HPHAT,S11,4HTHAT,S11,6HRHOHAT,S8,  *208
    1 9HPZERO'HAT,S4,1HM,S9,2HMI,S14,1HU,S11,1HG*$                          *208
      VECTOR VALUES VVQ=$1H0,F10.5,4F15.8,2F10,5,F15.3,2PF12.8*$            *209
      VECTOR VALUES VVR=$1H1,S4,1HX,S12,1HP,S12,4HPBAR,S13,1HT,S12,4HTBAR,S11 *210
    1 ,3HRHO,S11,6HRHOBAR,S11,1HA,S13,4HABAR*$                              *210
      VECTOR VALUES VVS=$1H0,F10.5,8E15.7*$                                 *211
      VECTOR VALUES VVT=$1H1,S4,1HX,S10,4HDELX,S12,1HG,S13,4HDELG,S10,6HRADIU *212
    1 S,S10,4HNDOT,S10,6HRHODRP,S7,11HMEAN RADIUS*$                         *212
      VECTOR VALUES VVU = $1H0,F10.5,F12.8,S3,2(2PF12.8,S3),5E15.7*$        *213
      VECTOR VALUES VIA=$HQOPZERO(DYNE/CM**2)Q,1PE10.3,S5,H*TZERO(DEG. K)*, *214
    1 F10.4,S5,HQRHOZERO(GM/CM**3)Q,1PE10.3,S5,HQPE(DYNE/CM**2)Q,           *214
    2 1PE10.3*$                                                             *214
      VECTOR VALUES VIB=$6HOGAMMA,F6.3,S5,H*MU(GM/GMOL)*,F7.2,S5,H*CP(DYNE-CM *215
    1 /GM-DEG. K)*,1PE10.3*$                                                *215
      VECTOR VALUES VIC=$H*OINTERMOLECULAR DISTANCE*,1PE10.3,S5,H*ACCOMODATIO *216
    1 N COEFFICIENT*,F7.3*$                                                 *216
      VECTOR VALUES VID=$H*ODELX*,F6.4,S5,H*MIN(DELX)*,F6.4,S5,H*MAX(DELX) IF *217
    1  NDOT=0*,F7.4,S5,H*MAX(DELX)*,F6.4*$                                  *217
      VECTOR VALUES VIE=$H*OWILL BEGIN CHANGING 'DELX' TO KEEP DELG GREATER T *218
    1 HAN*,2PF7.4,H* PERCENT AND LESS THAN*,2PF7.4,H* PERCENT WHEN          *218
    2 G EXCEEDS*,2PF7.4,H* PERCENT**$                                       *218
      VECTOR VALUES VIF=$H*OXRANGE IS*,F7.2,H* BUT WILL TERMINATE IF G EXCEED *219
    1 S*,2PF7.2,H* PERCENT.**$                                              *219
      VECTOR VALUES VIH=$H*OPRESSURE*,1PE10.3,S5,H*TEMP.*,1PE10.3,S5,H*DENSIT *220
    1 Y*,1PE10.3*$                                                          *220
      VECTOR VALUES VII=$H*OMACH NUMBER*,F7.3,S5,H*VELOCITY(CM/SEC)*,F10.2*$ *221
      VECTOR VALUES VIJ=$H*ONOZZLE POINT*,F10.4,S5,H*AREA*,F9.5*$           *222
      VECTOR VALUES VIK=$H*OTOTAL MASS FLOW(GM-CM/SEC)*,F10.5*$             *223

      END OF PROGRAM                                                        *224

THE FOLLOWING NAMES HAVE OCCURRED ONLY ONCE IN THIS PROGRAM.
```

```
$COMPILE MAD,PRINT OBJECT,PUNCH OBJECT                    CFLOW101                999066   06/24/64    11 11  0.7 AM

MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...

                   AERONAUTICAL ENGINEERING
                 NOZZLE CONDENSATION RESEARCH

                      CONDENSATION FLOW

           EXTERNAL FUNCTION (DELA,LOC)                                          *001

           STATEMENT LABEL LOC                                                   *002
           PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                              *003
         1          GAMMA,MU,CP,L,SIGMA,RHOL,                                    *003
         2          D,B,C,ALPHA,                                                 *003
         3          N,PPOINT,LINENO,PERPG,ZQ                                     *003
           DIMENSION ZQ(10)                                                      *004
           INTEGER N,PPOINT,LINENO,PERPG                                         *005
           PROGRAM COMMON X,DELX,A,P,T,RHO,U,M,                                  *006
         1          G,DELG,RADIUS,NDOT,DELRAD                                    *006
           DIMENSION X(500),DELX(500),A(500),                                    *007
         1        P(500),T(500),RHO(500),U(500),M(500),                          *007
         2        G(500),DELG(500),RADIUS(500),NDOT(500),DELRAD(500)             *007

          USING THE CURRENT VALUE OF N AND THE SUPPLIED VALUES FOR A(N)
          DELA(N),G(N) AND DELG(N) CALCULATE T(N),P(N),RHO(N) AND U(N)
          SATISFYING THE DIABATIC FLOW EQUATIONS. THE CONDITIONS AT
          POINT N-1 AND AN INITIAL APPROXIMATION TO U(N) ARE REQUIRED.

           ENTRY TO CFLOW.                                                       *008
           NM1=N-1                                                               *009
           DELU=U(N)-U(NM1)                                                      *010
           UNMG=1.-G(N)                                                          *011
           MDQG=DELG(N)/UNMG                                                     *012
           DQA=DELA/A(N)                                                         *013
           L=FL.(T(NM1))                                                         *014
           THROUGH S1, FOR I=1,1,I.G.50                                          *015
           DELT=(L*DELG(N)-U(N)*DELU)/CP                                         *016
           T(N)=T(NM1)+DELT                                                      *017
           L=FL.(T(N))                                                           *018
           DELU=DQA+MDQG+DELU/U(N)-DELT/T(N)                                     *019
           DELU=DELU*R/MU*T(N)/U(N)*UNMG                                         *020
           UT=U(NM1)+DELU                                                        *021
           TEST=.ABS.(1.-UT/U(N))                                                *022
           U(N)=UT                                                               *023
S1         WHENEVER TEST.L.EPSLON,TRANSFER TO S2                                 *024
            ITERATION LIMIT HAS BEEN REACHED WITH NO CONVERGENCE
           PRINT COMMENT $  'CFLOW' - ITERATION FAILED TO CONVERGE$              *025
           TRANSFER TO LOC                                                       *026
            ITERATION HAS CONVERGED
S2         P(N)=P(NM1)/(1.+(U(N)*DELU*MU)/(T(N)*R*(1.-G(N))))                    *027
           RHO(N)=RHO(NM1)/(1.+DELU/U(N)+DELA/A(N))                              *028
           M(N)=U(N)/SQRT.(R*GAMMA/MU*T(N))                                      *029
           FUNCTION RETURN                                                       *030

           VECTOR VALUES R=8.314E+07                                             *031
           VECTOR VALUES EPSLON=1.E-06                                           *032
```

```
INTEGER I,NM1                                                  *C33
END OF FUNCTION                                                *034
```

```
MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...

                     AERONAUTICAL ENGINEERING
                    NOZZLE CONDENSATION RESEARCH

                    INITIAL/CONDENSATION POINT

            EXTERNAL FUNCTION (TSAT,MDOT)                                    *001
            PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                         *002
          1          GAMMA,MU,CP,L,SIGMA,RHOL,                               *002
          2          D,B,C,ALPHA,                                            *002
          3          N,PPOINT,LINENO,PERPG,ZQ                                *002
            DIMENSION ZQ(10)                                                 *003
            INTEGER N,PPOINT,LINENO,PERPG                                    *004
            PROGRAM COMMON X,DELX,A,P,T,RHO,U,M,                             *005
          1           G,DELG,RADIUS,NDOT,DELRAD                              *005
            DIMENSION X(500),DELX(500),A(500),                               *006
          1        P(500),T(500),RHO(500),U(500),M(500),                     *006
          2        G(500),DELG(500),RADIUS(500),NDOT(500),DELRAD(500)        *006

            ENTRY TO CONDEN.                                                 *007
           SET INITIAL POINT TO SATURATION POINT WITH NO DIRT OR DROPS.

            N=0                                                              *008
            T=TSAT                                                           *009
               IFLOW. (A,M,P,RHO,T,U)                                        *010
               X=M                                                           *011
               NOZZLE. ($INVERS$,X,A)                                        *012
               ZERO. (NDOT,RADIUS,G,DELG)                                    *013

           THIS READ MAY ALTER ANY OF THE VALUES AT THE INITIAL POINT
           OR MAY CAUSE THE PROGRAM TO COMPUTE A 'CONDENSATION' POINT
           BASED ON THE EPSILON(P,T) CRITERIA.
            READ DATA                                                        *014
            WHENEVER EPSLON.NE.0., TRANSFER TO SZERO1                        *015
            PRINT COMMENT $0      THE CALCULATIONS ARE TO BE STARTED AT TH   *016
          1 E SATURATION POINT WITH ALL NUCLEATION QUANTITIES ZERO.$         *016
            PRINT COMMENT $ A 'READ DATA' STATEMENT FOLLOWS ALL THESE COM    *017
          1 PUTATIONS CONSEQUENTLY ANY OR ALL OF THE FOLLOWING MAY BE $      *017
            PRINT COMMENT $ CHANGED - X,A,P,RHO,T,U,M,RADIUS,NDOT,G,DELG$    *018
            PRINT FORMAT VVA,X,A,P,RHO,T,U,M                                 *019
            PRINT FORMAT VVB,RADIUS,NDOT,G                                   *020
            FUNCTION RETURN X(1)                                             *021
SZERO1      PRINT COMMENT $0   DETERMINE A CONDENSATION ON THE BASIS OF T    *022
          1 HE VALUE OF EPSILON(P,T), WHICH EVER APPLIES.$                   *022
            TMIN=T-TRANGE                                                    *023
            WHENEVER TMIN.L.0., TMIN=0.                                      *024
            CELT=CELT(1)                                                     *025
            DELG=0.                                                          *026

           MAXIMUM DELG POINT SEARCH -- IF EPSILON(P,T) EXCEEDSEPS, JUMP
           IMMEDIATELY TO HALF-INTERVAL FOR 'CONDENSATION' POINT
S1          T=T-CELT                                                         *027
            WHENEVER T.L.TMIN, TRANSFER TO S3                                *028
            IFLOW. (A,M,P,RHO,T,U)                                           *029
```

```
          X=M
          NOZZLE. ($INVERS$,X,A)
          WHENEVER X(1).E.0., TRANSFER TO S3
          RHOL=FRHOL. (P)
          SIGMA=FSIGMA. (P)
          DELG(1)=DELG
          NUCLE1. (MDOT)
          WHENEVER DELG.E.0., TRANSFER TO S1
          L=FL.(T)
          NOZZLE.($AREA$,X-DELX,A(1))
          DELA=A-A(1)
          EPSLON=(GAMMA-1./M/M)/(GAMMA-1.)
          WHENEVER EPSLON.L.1.,EPSLON=1.
          EPSLON=A*(1.-L/CP*EPSLON/T)/DELA*DELG
          WHENEVER .ABS.EPSLON.G.EPS, TRANSFER TO S5
S2        WHENEVER DELG.G.DELG(1), TRANSFER TO S1
S3        T=T+DELT+DELT

         TEMPERATURE T IS MAXIMUM POINT OF DELG. TESTS ON EPSLON(P,T)
         INDICATE THAT CONDENSATION IS NOT SUFFICIENT TO CALCULATE
         IN THE MAIN PROGRAM.
S4        IFLOW. (A,M.P,RHO,T,U)
          X=M
          NOZZLE. ($INVERS$,X,A)
          RHOL=FRHOL. (P)
          SIGMA=FSIGMA. (P)
          NUCLE1. (MDOT)
          WHENEVER DELT.G.EPS(1)
          DELT=DELT*.1
          TRANSFER TO S1
          END OF CONDITIONAL
          PRINT COMMENT $0   INSUFFICIENT INITIAL CONDENSATION TO INITI
          ATE CALCULATION.$
          PRINT FORMAT VVA,X,A,P,RHO,T,U,M
          PRINT FORMAT VVB,RADIUS,NDOT,DELG
          FUNCTION RETURN 0.

         HIGH IS A TEMPERATURE AT WHICH DELG.NE.0. AND WHERE EPSLON
         DOES NOT EXCEED EPS. TEMPERATURE T IS SUCH THAT EPSLON EXCEED
         EPS. BEGIN HALF-INTERVAL SEARCH FOR 'CONDENSATICN' POINT
S5        LOW=T
          HIGH=LOW+DELT
S6        T=.5*(HIGH+LOW)
          IFLOW. (A,M,P,RHO,T,U)
          X=M
          NOZZLE. ($INVERS$,X,A)
          L=FL.(T)
          RHOL=FRHOL. (P)
          SIGMA=FSIGMA. (P)
          NUCLE1. (MDOT)
          NOZZLE. ($AREA$,X-DELX,A(1))
          DELA=A-A(1)
          EPSLON=(GAMMA-1./M/M)/(GAMMA-1.)
          WHENEVER EPSLON.L.1., EPSLON=1.
          EPSLON=A*(1.-L/CP*EPSLON/T)/DELA*DELG
          WHENEVER .ABS.EPSLON.G.EPS
          LOW=T
          OTHERWISE
          HIGH=T
          END OF CONDITIONAL
```

```
      WHENEVER .ABS.(1.-LOW/HIGH).G.EPS(2), TRANSFER TO S6                  *081
        CONDENSATION POINT HAS BEEN FOUND
      G=DELG                                                                *082
      PRINT COMMENT $0   SUFFICIENT INITIAL CONDENSATION TO INITIAT         *083
    1 E THE CALCULATIONS. CONDENSATION POINT FOLLOWS.$                      *083
      PRINT FORMAT VVA,X,A,P,RHO,T,U,M                                      *084
      PRINT FORMAT VVB,RADIUS,NDOT,DELG                                     *085
      FUNCTION RETURN 1.                                                    *086

      VECTOR VALUES DELT = 0.,5.                                            *087
      VECTOR VALUES EPS=.001,.01,.00001                                     *088
      VECTOR VALUES TRANGE = 100.                                           *089
      VECTOR VALUES VVA = $1H0,2HX=,F8.4,3H A=,F9.4,3H P=,E11.5,5H RHO=,E11.5  *090
    1 ,3H T=,F9.4,3H U=,F9.2,3H M=,F8.4*$                                   *090
      VECTOR VALUES VVB = $1H0,7HRADIUS=,1PE12.5,6H NDOT=,1PE12.5,3H G=,    *091
    1 1PE12.5*$                                                             *091

      END OF FUNCTION                                                       *092

THE FOLLOWING NAMES HAVE OCCURRED ONLY ONCE IN THIS PROGRAM.
COMPILATION WILL CONTINUE.

     S2
     S4
```

MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ...

```
                    AERONAUTICAL ENGINEERING
                  NOZZLE CONDENSATION RESEARCH

                        ISENTROPIC FLOW

       EXTERNAL FUNCTION (A1,A2,A3,A4,A5,A6)                               *001

       PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                            *002
     1               GAMMA,MU,CP,L,SIGMA,RHOL,                             *002
     2              D,B,C,ALPHA,                                           *002
     3              N,PPOINT,LINENO,PERPG,ZQ                               *002
       DIMENSION ZQ(10)                                                    *003
       INTEGER N,PPOINT,LINENO,PERPG                                       *004

       ENTRY TO IFLOW.                                                     *005
       T=A5                                                                *006
             PRESSURE
       EXP=GAMMA/(GAMMA-1.)                                                *007
       Z=(T/TZERO) .P. EXP                                                 *008
       A3=PZERO*Z                                                          *009
             VELOCITY AND MACH NUMBER
       Z=2.*(TZERO-T)                                                      *010
       A6=SQRT.(CP*Z)                                                      *011
       Z=Z/(GAMMA-1.)                                                      *012
       A2=SQRT.(Z/T)                                                       *013
             DENSITY AND NOZZLE AREA
       EXP=1./(GAMMA-1.)                                                   *014
       Z=1.+.5*A2*A2/EXP                                                   *015
       A4=RHZERO/(Z .P. EXP)                                               *016
       EXP=.5*(GAMMA+1.)*EXP                                               *017
       Z=2.*Z/(GAMMA+1.)                                                   *018
       Z=(Z .P. EXP)/A2                                                    *019
       A1=ASTAR*Z                                                          *020

       FUNCTION RETURN                                                     *021

       END OF FUNCTION                                                     *022
```

```
MAC (12 MAR 1964 VERSICN) PROGRAM LISTING ... ... ...

                      AERCNAUTICAL ENGINEERING
                    NOZZLE CONDENSATION RESEARCH

           EXTERNAL FUNCTION (A,P,TEMP,RHO,M)                                   *001

          GIVEN THE ABOVE QUANTITIES, THIS SUBROUTINE CCMPUTES AND
          PLACES IN PROGRAM COMMON
              MI = ISENTROPIC MACH NUMBER CORRESPCNDING TO GIVEN AREA
                   RATIC - ABAR.
            PHAT = RATIO OF COMPUTED STATIC PRESSURE TC CORRESPONDING
                   ISENTROPIC PRESSURE.
            THAT = STATIC TEMPERATURE RATIO.
          RHCHAT = STATIC DENSITY RATIO.
          PZPHAT = PITOT PRESSURE RATIO.

           PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERC,                             *002
         1               GAMMA,MU,CP,L,SIGMA,RHOL,                              *002
         2               D,B,C,ALPHA,                                           *002
         3               N,PPOINT,LINENC,PERPG,ZQ                               *002
           DIMENSION ZQ(10)                                                     *003
           INTEGER N,PPCINT,LINENO,PERPG                                        *004

          INTERNAL FUNCTION FOR MI IN TERMS OF ABAR.
           INTERNAL FUNCTION F.(ARG)=(T(1)/ARG)*((1.+T(2)*ARG*ARG) .P.          *005
         1  T) - ABAR                                                           *005

           ENTRY TC ISEN.                                                       *006
           ABAR=A/ASTAR                                                         *007

          COMPUTE ISENTROPIC MACH NUMBER FOR GIVEN AREA RATIO
           T=(GAMMA+1.)/(2.*(GAMMA-1.))                                         *008
           T(1)=(2./(GAMMA+1.)) .P. T                                           *009
           T(2)=.5*(GAMMA-1.)                                                   *010
           MI=((GAMMA+1.)/(GAMMA-1.)) .P. T                                     *011
           MI= (1.+(ABAR-1.)*MI) .P. T(2)                                       *012
           WHENEVER MI.L.1.4                                                    *013
           CONTINUE                                                             *014
           OR WHENEVER MI.L.4.                                                  *015
           MI=.8*MI                                                             *016
           OR WHENEVER MI.L.6.                                                  *017
           MI=.9*MI                                                             *018
           OR WHENEVER MI.L.10.                                                 *019
           MI=.95*MI                                                            *020
           END OF CONDITIONAL                                                   *021
           T(3)=F.(MI)                                                          *022
           WHENEVER .ABS.T(3) .L. EPS, TRANSFER TO S3                           *023
           DELM=.1*(T(3)/.ABS.T(3))                                             *024
S1         T(4)=F.(MI-DELM)                                                     *025
           WHENEVER .ABS.T(4) .L. EPS, TRANSFER TO S2                           *026
           WHENEVER T(3)*T(4) .G. 0.                                            *027
           MI=MI-DELM                                                           *028
           OR WHENEVER T(3)*T(4) .L. 0.                                         *029
           DELM=.5*DELM                                                         *030
```

```
         END OF CONDITIONAL                                            *031
         TRANSFER TO S1                                                *032
S2       MI=MI-DELM                                                    *033

        HALF-INTERVAL TECHNIQUE FOR THE ISENTROPIC MACH NUMBER HAS
        CONVERGED. COMPUTE ISENTROPIC DENSITY, PRESSURE AND TEMP.
S3       TI=1./(1.+T(2)*MI*MI)                                         *034
         PI= TI .P. (GAMMA/(GAMMA-1.))                                 *035
         RHOI= TI .P. (1./(GAMMA-1.))                                  *036

        COMPUTE PRESSURE RATIOS NECESSARY FOR COMPUTING THE PITOT
        PRESSURE RATIO.
         T=GAMMA+1.                                                    *037
         T(1)=GAMMA-1.                                                 *038
         T(2)=2.*GAMMA                                                 *039
         RM=(T/(T(2)*M*M-T(1))) .P. (1./T(1))                          *040
         RMI=(T/(T(2)*MI*MI-T(1))) .P. (1./T(1))                       *041
         T(2)=GAMMA/T(1)                                               *042
         RMI=RMI*(T/(T(1)+2./(MI*MI))) .P. T(2)                        *043
         RM=RM*(.5*T*M*M) .P. T(2)                                     *044

        COMPUTE FINAL QUANTITIES DESIRED AND PLACE IN PROGRAM COMMON
         ZC(1)=P/PZERO/PI                                              *045
         ZC(2)=TEMP/TZERO/TI                                           *046
         ZC(3)=RHO/RHZERO/RHOI                                         *047
         ZC(4)=P/PZERO*RM/RMI                                          *048
         ZC(5)=MI                                                      *049
         FUNCTION RETURN                                               *050

         VECTOR VALUES EPS = 1.E-03                                    *051
         DIMENSION T(7)                                                *052

         END OF FUNCTION                                               *053
```

```
MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...


                  AERONAUTICAL ENGINEERING
                 NOZZLE CONDENSATION RESEARCH

                          NOZZLE

         ORIGIN OF THE COORDINATE SYSTEM FOR THE NOZZLE IS THE CENTER
         OF THE THROAT, I.E., A(0) = ASTAR. THE VARIABLE X INCREASES
         POSITIVELY IN THE DIRECTION OF FLOW AND IS NEGATIVE UPSTREAM
         FROM THE THROAT. THE NOZZLE PARAMETERS ARE
              ASTAR  = THROAT AREA
               XMIN  = X-COORDINATE OF INTAKE
               XMAX  = X-COORDINATE OF EXIT
              INANG  = INTAKE HALF ANGLE IN DEGREES
              OUTANG = EXIT SIDE HALF ANGLE IN DEGREES
              WEDGE  = BOOLEAN VARIABLE, 0B OR 1B, INDICATING WHETHER
                       THE NOZZLE IS CONICAL OR WEDGE RESPECTIVELY.
         THE ENTRY THROAT READS AND PRINTS THESE PARAMETERS.

          EXTERNAL FUNCTION (A1,A2,A3)                                  *001

          PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                      *002
        1              GAMMA,MU,CP,L,SIGMA,RHOL,                        *002
        2              D,B,C,ALPHA,                                     *002
        3              N,PPOINT,LINENO,PERPG,ZQ                         *002
          DIMENSION ZQ(10)                                              *003
          INTEGER N,PPOINT,LINENO,PERPG                                 *004

          ENTRY TO NOZZLE.                                              *005

                  WHENEVER A1 .E. $THROAT$                              *006
              RETURN THROAT AREA AND INITIALIZE IF NECCESSARY
          READ DATA                                                     *007
          WHENEVER WEDGE                                                *008
          PRINT COMMENT $0A WEDGE NOZZLE WITH THE FOLLOWING SPECS. IS B *009
        1 EING USED --$                                                 *009
          DSTAR=ASTAR                                                   *010
          OTHERWISE                                                     *011
          PRINT COMMENT $0A CONICAL NOZZLE WITH THE FOLLOWING SPECS. IS *012
        1  BEING USED --$                                               *012
          DSTAR= 1.1283792*SQRT.(ASTAR)                                 *013
          END OF CONDITIONAL                                            *014
          PRINT FORMAT VVA,ASTAR,INANG,XMIN                             *015
          PRINT FORMAT VVB,OUTANG,XMAX                                  *016
          A=INANG/RADIAN                                                *017
          INTAN=SIN.(A)/COS.(A)                                         *018
          A=OUTANG/RADIAN                                               *019
          OUTTAN=SIN.(A)/COS.(A)                                        *020
          A3=ASTAR                                                      *021
          FUNCTION RETURN                                               *022

                  OR WHENEVER A1 .E. $AREA$                             *023
              COMPUTE THE AREA AT A2 AND RETURN IN A3
          WHENEVER (XMIN.L.A2) .AND. (A2.L.0.)                          *024
```

```
      A=INTAN*(.ABS. A2)                                              *025
      OR WHENEVER (A2.G.0.) .AND. (A2.L.XMAX)                         *026
      A=OUTTAN*A2                                                     *027
      OTHERWISE                                                       *028
      A3=0.                                                           *029
      WHENEVER A2.E.0.,A3=ASTAR                                       *030
      FUNCTION RETURN                                                 *031
      END OF CONDITIONAL                                              *032
      WHENEVER WEDGE                                                  *033
      A3=DSTAR+2.*A                                                   *034
      OTHERWISE                                                       *035
      A=DSTAR+2.*A                                                    *036
      A3=A*A*.7853982                                                 *037
      END OF CONDITIONAL                                              *038
      FUNCTION RETURN                                                 *039

            OR WHENEVER A1 .E. $INVERS$                               *040
         COMPUTE THAT POINT ON THE X-AXIS OF THE NOZZLE THAT
         HAS AREA 'A3'. THIS POINT DEPENDS ON THE MACH NUMBER
         WHICH MUST BE IN A2 ON ENTRY, SEE THE PROGRAM FOR
         THE PARTICULARS. IF A2(1) IS 1. ON RETURN THEN THE
         SOLUTION IS IN A2, IF IT IS ZERO THEN THE SOLUTION IS
         STORED IN A2 BUT DOES NOT LIE INSIDE THE NOZZLE.
      WHENEVER WEDGE                                                  *041
      A=.5*(A3-ASTAR)                                                 *042
      OTHERWISE                                                       *043
      A=.5641896*SQRT.(A3)-.5*DSTAR                                   *044
      END OF CONDITIONAL                                              *045
      WHENEVER A.L.0.,A=0.                                            *046
      X(1)=1.                                                         *047
      MACH=A2                                                         *048
      WHENEVER MACH.L.1.                                              *049
         X=-A/INTAN                                                   *050
         WHENEVER XMIN.G.X,X(1)=0.                                    *051
      OTHERWISE                                                       *052
         X=A/OUTTAN                                                   *053
         WHENEVER X.G.XMAX,X(1)=0.                                    *054
      END OF CONDITIONAL                                              *055
      A2=X                                                            *056
      A2(1)=X(1)                                                      *057
      FUNCTION RETURN                                                 *058

            END OF CONDITIONAL                                        *059

      INTEGER A1                                                      *060
      BOOLEAN WEDGE                                                   *061
      DIMENSION X(1)                                                  *062
      VECTOR VALUES RADIAN=57.2957795                                 *063
      VECTOR VALUES VVA=$H*OTHROAT AREA*,F9.5,S5,H*INPUT HALF-ANGLE(DEG.)*,   *064
    1 F9.5,S5,H*INPUT LENGTH*,F9.4*$                                  *064
      VECTOR VALUES VVB=$1H0,S24,H*OUTPUT HALF-ANGLE(DEG.)*,F9.5,S4,H*OUTPUT  *065
    1 LENGTH*,F9.4*$                                                  *065

      END OF FUNCTION                                                 *066
```

```
$COMPILE MAD,PRINT OBJECT,PUNCH OBJECT                 NUCL1101            999066   06/24/64     11 11   8.1 AM

MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...

                           AERONAUTICAL ENGINEERING
                          NOZZLE CONDENSATION RESEARCH

                             NUCLEATION THEORY - 1

             EXTERNAL FUNCTION (MDOT)                                          *001

             PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                          *002
           1               GAMMA,MU,CP,L,SIGMA,RHOL,                           *002
           2               D,B,C,ALPHA,                                        *002
           3               N,PPOINT,LINENO,PERPG,ZQ                            *002
             DIMENSION ZQ(10)                                                  *003
             INTEGER N,PPOINT,LINENO,PERPG                                     *004
             PROGRAM COMMON X,DELX,A,P,T,RHO,U,M,                              *005
           1                G,DELG,RADIUS,NDOT,DELRAD                          *005
             DIMENSION X(500),DELX(500),A(500),                                *006
           1          P(500),T(500),RHO(500),U(500),M(500),                    *006
           2          G(500),DELG(500),RADIUS(500),NDOT(500),DELRAD(500)       *006

             ENTRY TO NUCLE1.                                                  *007
             AT=T(N)                                                           *008

                 COMPUTE RADIUS ( CM. ) OF THE DROPS THAT ARE CONDENSING

             FSPILL.(S1)                                                       *009
             KBAR=R*(ELOG. (P(N)) - LNPSAT. (AT))                              *010
             KBAR=2./RHOL*MU/AT/KBAR                                           *011
             WHENEVER KBAR.E.0., TRANSFER TO S1                                *012
             ARAD=KBAR*SIGMA-D                                                 *013
             SIGMA=SIGMA-D/KBAR                                                *014
             WHENEVER ARAD.G.1.E-06, TRANSFER TO S1                            *015
             WHENEVER (SIGMA.L.0.) .OR. (ARAD.L.0.)                            *016
  S1         ZERO. (SIGMA,RADIUS(N),NDOT(N),DELG(N))                           *017
             RSPILL.(0)                                                        *018
             FUNCTION RETURN                                                   *019
             END OF CONDITIONAL                                                *020

                 COMPUTE THE NUMBER OF DROPS OF THIS SIZE THAT ARE
                 CONDENSING PER CENTIMETER CUBED PER SECOND.

             TEXP=-(4.18879*ARAD/K*ARAD/AT*SIGMA)                              *021
             ANDOT=SQRT.(SIGMA*MU/NA(1))/K(1)*P(N)/AT                          *022
             ANDOT=EXP.(TEXP)/AT*ANDOT/RHOL*P(N)/K                             *023
             NDOT(N)=.7978846*ANDOT*A(N)*DELX(N)                               *024
             WHENEVER NDOT(N).L.1., TRANSFER TO S1                             *025

                 COMPUTATION OF PERCENT OF LIQUID MASS, DELTAG

             DELG(N)=4.18879*RHOL*NDOT(N)*ARAD*ARAD/MDOT*ARAD                  *026
             RADIUS(N)=ARAD                                                    *027
             RSPILL.(0)                                                        *028
             FUNCTION RETURN                                                   *029
```

```
VECTOR VALUES NA=6.027E+23,6.027E+03                          *030
VECTOR VALUES K=1.379E-16,1.379E-06                           *031
VECTOR VALUES R=8.314E+07                                     *032

END OF FUNCTION                                               *033
```

```
MAD (12 MAR 1964 VERSION) PROGRAM LISTING ... ... ...

                  AERONAUTICAL ENGINEERING
                NOZZLE CONDENSATION RESEARCH

                    NUCLEATION THEORY - 2

          EXTERNAL FUNCTION (MDOT)                                          *001

          PROGRAM COMMON ASTAR,PZERO,TZERO,RHZERO,                          *002
        1                 GAMMA,MU,CP,L,SIGMA,RHOL,                         *002
        2                 D,P,C,ALPHA,                                      *002
        3                 N,PPOINT,LINENO,PERPG,ZQ                          *002
          DIMENSION ZQ(10)                                                  *003
          INTEGER N,PPOINT,LINENO,PERPG                                     *004
          PROGRAM COMMON X,DELX,A,P,T,RHO,U,M,                              *005
        1                G,DELG,RADIUS,NDOT,DELRAD                          *005
          DIMENSION X(500),DELX(500),A(500),                                *006
        1           P(500),T(500),RHO(500),U(500),M(500),                   *006
        2           G(500),DELG(500),RADIUS(500),NDOT(500),DELRAD(500)      *006

          ENTRY TO NUCLE2.                                                  *007

          NUCLE1.(MDOT)                                                     *008

               WE HAVE NOW ONLY TO COMPUTE THE RADIAL INCREMENTS AND
               THEIR CONTRIBUTION TO THE CONDENSATE MASS INCREASE.

          TDROP=FTSAT. (P(N))                                               *009
          DELRAD=SQRT.(1.379/T(N)*.6027/MU)*7978.846*(TDROP-T(N))           *010
          DELRAD=DELX(N)*ALPHA/L*P(N)/RHOL*DELRAD/U(N)                      *011
          I=1                                                               *012
S1        DELRAD(I)=DELRAD                                                  *013
          I=I+1                                                             *014
          WHENEVER I.L.N,TRANSFER TO S1                                     *015

               COMPUTATION OF RESULTANT TOTAL CONDENSATE MASS INCREASE

          GROW=RADIUS*RADIUS*NDOT*DELRAD                                    *016
          I=1                                                               *017
S2        GROW=GROW+RADIUS(I)*RADIUS(I)*NDOT(I)*DELRAD(I)                   *018
          I=I+1                                                             *019
          WHENEVER I.L.N, TRANSFER TO S2                                    *020
          DELG(N)=DELG(N)+12.56637*RHOL/MDOT*GROW                           *021
          FUNCTION RETURN                                                   *022

          INTEGER I                                                         *023

          END OF FUNCTION                                                   *024
```

```
$CCMPILE MAC,PRINT OBJECT,PUNCH OBJECT                    VAPOR001          999066   06/24/64     11 10 33.5 AM

MAD (12 MAR 1964 VERSICN) PROGRAM LISTING ... ... ...

                         AERCNAUTICAL ENGINEERING
                        NCZZLE CONDENSATION RESEARCH

                              VAPOR PARAMETERS

             EXTERNAL FUNCTION (ARG)                                           *001

             ENTRY TO VAPCR.                                                   *002
            THE FCLLOWING READ DATA STATEMENT MUST SUPPLY THE QUANTITIES
            PRINTEC BELOW.
             REAC CATA  (VAPOR PARAMETER APPRCXIMATICNS AND TRIPLE POINT)      *003
             PRINT FORMAT VCC,PTP,TTP                                          *004
             PRINT FORMAT VOD,PSAT,PSAT(1)                                     *005
             PRINT FCRMAT VOE,PSAT(2)...PSAT(4)                                *006
             PRINT FCRMAT VOF,SIGMA,SIGMA(1)                                   *007
             PRINT FORMAT VCG,RHOL,RHOL(1)                                     *008
             PRINT FORMAT VOH,L,L(1)                                           *009
             FUNCTION RETURN                                                   *010

            CCMPUTE THE LATENT HEAT -- ARG IS VAPOR TEMPERATURE
             ENTRY TC FL.                                                      *011
             FUNCTION RETURN L+L(1)*ARG                                        *012

            CCMPUTE THE TEMPERATURE CORRESPONDING TO PRESSURE ARG
             ENTRY TO FTSAT.                                                   *013
             TP=C                                                              *014
SS1          P=ELCG.(ARG)                                                      *015
             WHENEVER P.LE.PTP                                                 *016
             TSAT=1./(PSAT+PSAT(1)*P)                                          *017
             CTHERWISE                                                         *018
             TSAT=1./(PSAT(2)+P*(PSAT(3)+P*PSAT(4)))                           *019
             END CF CONDITIONAL                                                *020
             TRANSFER TC SS2(TP)                                               *021
SS2(C)       FUNCTION RETURN TSAT                                              *022
SS2(1)       FUNCTION RETURN SIGMA+SIGMA(1)*TSAT                               *023
SS2(2)       FUNCTION RETURN RHOL+RHOL(1)*TSAT                                 *024

            CCMPUTE SURFACE TENSION -- ARG IS VAPOR PRESSURE
             ENTRY TC FSIGMA.                                                  *025
             TP=1                                                              *026
             TRANSFER TC SS1                                                   *027

            CCMPUTE LIQUID DENSITY -- ARG IS VAPOR PRESSURE
             ENTRY TO FRHCL.                                                   *028
             TP=2                                                              *029
             TRANSFER TO SS1                                                   *030

            CCMPUTE LOG OF PRESSURE CORRESPONDING TC TEMPERATURE ARG
             ENTRY TO LNPSAT.                                                  *031
             T=1./ARG                                                          *032
             WHENEVER T.GE.TTP                                                 *033
             LNP=(T-PSAT)/PSAT(1)                                              *034
SS3          FUNCTION RETURN LNP                                               *035
```

```
  OR WHENEVER PSAT(4).NE.0.                                            *036
  REAL=-PSAT(3)/2.                                                     *037
  DISCRM=REAL*REAL-PSAT(4)*(PSAT(2)-T)                                 *038
  WHENEVER DISCRM.L.0.                                                 *039
  PRINT COMMENT $0SATURATION CURVE APPROXIMATION ABOVE THE TRIP        *040
1 LE POINT GIVES IMAGINARY LOG.$                                       *040
  ERROR.                                                               *041
  END OF CONDITIONAL                                                   *042
  DISCRM=SQRT. (DISCRM)                                                *043
  LNP=(REAL-DISCRM)/PSAT(4)                                            *044
  REAL=(REAL+DISCRM)/PSAT(4)                                           *045
  WHENEVER LNP.L.PTP, LNP=REAL                                         *046
  TRANSFER TO SS3                                                      *047
  OTHERWISE                                                            *048
  LNP= (T-PSAT(2))/PSAT(3)                                             *049
  TRANSFER TO SS3                                                      *050
  END OF CONDITIONAL                                                   *051

  INTEGER I,TABLE,TP                                                   *052
  DIMENSION SIGMA(1),RHOL(1),L(1),PSAT(4),LMTX(10),QMTX(19),TP(        *053
1 1)                                                                   *053
  VECTOR VALUES VIA=$2F20,C1*$                                         *054
  VECTOR VALUESVOA=$H0OSATURATION CURVE DATA - NATURAL LOG OF P(DYNE/CM**  *055
1 2) VERSUS 1/T(DEG. K)0*$                                             *055
  VECTOR VALUES VOB=$1H ,S10,C6,2(1PE20.7)*$                           *056
  VECTOR VALUES VOC=$H*0TRIPLE POINT*,S4,2(1PE20.7)*$                  *057
  VECTOR VALUES VOD=$H*0LINEAR APPROXIMATION TO 1/T BELOW TRIPLE POINT --  *058
1 *,S5,1PE12.5,HQ +LN(P)*Q,1PE12.5*$                                   *058
  VECTOR VALUES VOE=$H*0QUADRATIC APPROXIMATION TO 1/T ABOVE TRIPLE POINT  *059
1  --*,S5,1PE12.5,HQ +LN(P)*Q,1PE12.5,HQ +LN(P).P.2*Q,1PE12.5*$        *059
  VECTOR VALUES VOF=$H*0LINEAR APPROXIMATION TO SURFACE TENSION --*,S5,  *060
1 1PE12.5,HQ +(DROP TEMP.)*Q,1PE12.5*$                                 *060
  VECTOR VALUES VOG=$H*0LINEAR APPROXIMATION TO LIQUID DENSITY --*,S5,   *061
1 1PE12.5,HQ +(DROP TEMP.)*Q,1PE12.5*$                                 *061
  VECTOR VALUES VOH=$H*0LINEAR APPROXIMATION TO LATENT HEAT --*,S5,     *062
1 1PE12.5,HQ +(VAPOR TEMP.)*Q,1PE12.5*$                                *062

  END OF FUNCTION                                                      *063
```

APPENDIX B

EXAMPLE INPUT

```
COMMENT CARDS FOR THIS RUN
         EXAMPLE OF AN UNDEREXPANDED SONIC FLOW IN A FREE JET. THE
    VAPOR IS NITROGEN.
END
SIGMA=23.94,  -.1933,
RHOL=1.181,  -.0048,
L=2.9E+09,  -.0117E+09,
PTP=1.173883E+01,   TTP=1.5832805E-02,
PSAT=3.00396E-02, -1.21024E-03, 3.20E-02, -1.3548E-03, -1.913E-06
   *
COMMENT CARDS FOR THE FIRST PROBLEM
         THE OUTPUT FOR THIS PROBLEM IS TO BE REPRODUCED IN THE PROGRAM
    WRITE-UP AS AN ILLUSTRATION OF THE OUTPUT STRUCTURE. ALSO, A LISTING
    OF THE DATA DECK IS TO BE INCLUDED.
DATA
PZERO=6.895E+06,  TZERO=298.,  PE=3000.
GAMMA=1.4,  MU=28.02,  CP=1.0364E+07,  D=.65E-08,  ALPHA=1.
DELX=.05,  XRANGE=5.,  MAXG=.1,  XPOINT=5
MING=.0001,  MNDELG=.0005,  MXDELG=.0015
MNDELX=.00001,  MXDELX=.2, .2     *
WEDGE=0B,  ASTAR=.0507
INANG=20.,  OUTANG=39.57,  XMIN=-10.,  XMAX=5.     *
EPSLON=1.,  DELT(1)=5.,  TRANGE=100.
EPS=.001, .0001, .00001     *
```

APPENDIX C

EXAMPLE OUTPUT

```
$EXECUTE,FULL DUMP,I/O DUMP                                                     999234   07/06/64     1 27 21.9 PM

$DATA                                                                           999234   07/06/64     1 27 29.4 PM

MAP
    SCARDS 00000*    SPEEK  00000*    SYSTEM 00000*    ERROR  00000*    SKIP6  00000*    SPRINT 00000*
    (MAIN) 10000     FL     32353     FTSAT  32353     FSIGMA 32353     FRHOL  32353     LNPSAT 32353
    VAPOR  32353     IFLOW  33212     CONDEN 33401     CFLOW  34524     NUCLE1 35023     NUCLE2 35275
    ISEN   35454     NOZZLE 36157     .IOH   36720*    DFDP   43065*    DFMP   43065*    .ERK   43234*
    .03311 43322*    .PRINT 43337*    .READ  43423*    .RDATA 43570*    .PCOMT 44670*    SQRT   44707*
    .EXIT  44766*    ELOG   45060*    .01300 45162*    EXP    45250*    ZERO   45363*    COS    45417*
    SIN    45417*    FSPILL 45616*    RSPILL 45616*    (PROG) 45676     (SUBT) 73676     (ERAS) 77741

32043 LOCS. CAN BE SAFELY USED IN EXPANDING PROG. (OCTAL)
26000 LOCS. CAN BE SAFELY USED IN EXPANDING PROG. (OCTAL)   BEFORE FULL CORE LOADING PROCEDURE IS USED


NOZZLE CONDENSATION RESEARCH PROGRAM

      COMMENT CARDS FOR THIS RUN

               EXAMPLE OF AN UNDEREXPANDED SONIC FLOW IN A FREE JET. THE

          VAPOR IS NITROGEN.

TRIPLE POINT            1.1738830E 01        1.5832805E-02

LINEAR APPROXIMATION TO 1/T BELOW TRIPLE POINT --       3.00396E-02 +LN(P)*-1.21024E-03

QUADRATIC APPROXIMATION TO 1/T ABOVE TRIPLE POINT --       3.20000E-02 +LN(P)*-1.35480E-03 +LN(P).P.2*-1.91300E-06

LINEAR APPROXIMATION TO SURFACE TENSION --       2.39400E 01 +(DROP TEMP.)*-1.93300E-01

LINEAR APPROXIMATION TO LIQUID DENSITY --       1.18100E 00 +(DROP TEMP.)*-4.80000E-03

LINEAR APPROXIMATION TO LATENT HEAT --       2.90000E 09 +(VAPOR TEMP.)*-1.17000E 07
```

```
                              BEGINNING OF PROBLEM NUMBER  1
   COMMENT CARDS FOR THE FIRST PROBLEM
            THE OUTPUT FOR THIS PROBLEM IS TO BE REPRODUCED IN THE PROGRAM
      WRITE-UP AS AN ILLUSTRATION OF THE OUTPUT STRUCTURE. ALSO, A LISTING
      OF THE DATA DECK IS TO BE INCLUDED.
COMPLETE LIST OF PROGRAM PARAMETERS FOR THIS PROBLEM.
PZERO(DYNE/CM**2) 6.895E 06     TZERO(DEG. K)  298.0000     RHOZERO(GM/CM**3) 7.798E-03     PE(DYNE/CM**2) 3.000E 03
GAMMA 1.400     MU(GM/GMOL)  28.02     CP(DYNE-CM/GM-DEG. K) 1.036E 07
INTERMOLECULAR DISTANCE 6.500E-09     ACCOMODATION COEFFICIENT  1.000
DELX .0500     MIN(DELX) .0000     MAX(DELX) IF NDOT=0  .2000     MAX(DELX) .2000
WILL BEGIN CHANGING 'DELX' TO KEEP DELG GREATER THAN  .0500 PERCENT AND LESS THAN  .1500 PERCENT WHEN G EXCEEDS  .0100 PERCENT
XRANGE IS   5.00 BUT WILL TERMINATE IF G EXCEEDS  10.00 PERCENT.
A CONICAL NOZZLE WITH THE FOLLOWING SPECS. IS BEING USED --
THROAT AREA   .05070     INPUT HALF-ANGLE(DEG.) 20.00000     INPUT LENGTH -10.0000
                         OUTPUT HALF-ANGLE(DEG.) 39.57000    OUTPUT LENGTH   5.0000
SATURATION POINT SPECIFICATIONS --
PRESSURE 1.881E 04     TEMP. 5.516E 01     DENSITY 1.149E-04
MACH NUMBER  4.692     VELOCITY(CM/SEC)  70947.86
NOZZLE POINT     .5242     AREA   .98609
TOTAL MASS FLOW(GM-CM/SEC)    8.04159
   DETERMINE A CONDENSATION ON THE BASIS OF THE VALUE OF EPSILON(P,T), WHICH EVER APPLIES.
   SUFFICIENT INITIAL CONDENSATION TO INITIATE THE CALCULATIONS. CONDENSATION POINT FOLLOWS.
X=    .9257 A=    2.4999 P= .48695E 04 RHO= .43775E-04 T=  37.4901 U= 73483.67 M=   5.8944
RADIUS= 4.10282E-08 NDOT= 1.29392E 17 G= 4.30682E-06
```

X	PHAT	THAT	RHOHAT	PZERO'HAT	M	MI	U	G
.92572	1.00000165	1.00000046	1.00000125	1.00000124	5.89439	5.89439	73483.665	.00043668
1.22572	1.11637212	1.04835926	1.05232979	1.06049494	6.43497	6.60377	74347.828	.33630150
1.35072	1.17677504	1.10211183	1.05917951	1.06259473	6.52466	6.86909	74598.180	.79741888
1.47572	1.24937637	1.16972242	1.06505322	1.06231128	6.56156	7.12033	74808.747	1.34568967
1.60072	1.32770474	1.24422354	1.07009847	1.06092468	6.57306	7.35937	74994.598	1.91609053
1.72572	1.40912810	1.32286535	1.07440567	1.05889918	6.57061	7.58760	75162.813	2.48007035
1.85072	1.49146686	1.40331079	1.07809214	1.05659957	6.56237	7.80624	75318.182	3.02480909
1.97572	1.57300285	1.48366651	1.08126585	1.05428338	6.55373	8.01632	75463.894	3.54369906
2.10072	1.65311074	1.56325606	1.08399963	1.05200428	6.54629	8.21866	75601.555	4.03296441
2.22572	1.73123863	1.64142771	1.08635716	1.04982834	6.54128	8.41397	75732.398	4.49093986
2.35072	1.80707572	1.71777581	1.08839522	1.04779327	6.53920	8.60286	75857.274	4.91764873
2.47572	1.88039872	1.79199050	1.09016861	1.04592089	6.54032	8.78587	75976.842	5.31425929
2.60072	1.95110263	1.86392486	1.09170452	1.04419923	6.54455	8.96346	76091.563	5.68261015
2.72572	2.01933250	1.93369046	1.09303683	1.04260689	6.55139	9.13601	76201.710	6.02482647
2.85072	2.08551580	2.00169969	1.09420782	1.04110962	6.55993	9.30393	76307.410	6.34296513
2.97572	2.14929032	2.06749055	1.09523097	1.03974678	6.57081	9.46751	76409.187	6.63914955
3.10072	2.21079704	2.13117468	1.09612484	1.03850290	6.58366	9.62703	76507.231	6.91541952
3.22572	2.27018130	2.19287246	1.09690614	1.03736521	6.59817	9.78276	76601.724	7.17364043
3.47572	2.38762268	2.30990592	1.10038704	1.03767382	6.63233	10.08373	76779.726	7.63846189
3.72572	2.49839348	2.42076355	1.10330294	1.03799909	6.66970	10.37202	76945.393	8.05122972
3.97572	2.60334232	2.52622792	1.10576496	1.03832151	6.70912	10.64895	77099.935	8.42032981
4.22572	2.70310935	2.62686610	1.10785832	1.03863707	6.74988	10.91565	77244.466	8.75251126
4.47572	2.79834184	2.72325906	1.10964979	1.03894433	6.79133	11.17305	77379.929	9.05321681
4.72572	2.88977814	2.81615660	1.11119476	1.03921627	6.83271	11.42198	77507.104	9.32682598

NOZZLE SUBROUTINE ERROR INDICATION - MESH POINT DOES NOT LIE WITHIN THE NOZZLE.

FINAL OUTPUT IN TABLE 1 FOR PROBLEM 1

4.92572	2.96298575	2.88817078	1.11312878	1.04029332	6.86559	11.61550	77602.939	9.52749002

X	P	PBAR	T	TBAR	RHO	RHOBAR	A	ABAR
.92572	.4869515E 04	.7062385E-03	.3749010E 02	.1258057E 00	.4377510E-04	.5613723E-02	.2499909E 01	.4930787E 02
1.22572	.2686620E 04	.3896476E-03	.3213461E 02	.1078342E 00	.2784482E-04	.3570823E-02	.4082548E 01	.8052363E 02
1.35072	.2209171E 04	.3204018E-03	.3146813E 02	.1055977E 00	.2347028E-04	.3009831E-02	.4855958E 01	.9577827E 02
1.47572	.1867038E 04	.2707815E-03	.3129109E 02	.1050037E 00	.2005171E-04	.2571434E-02	.5696414E 01	.1123553E 03
1.60072	.1606664E 04	.2330188E-03	.3133679E 02	.1051570E 00	.1732804E-04	.2222150E-02	.6603916E 01	.1302547E 03
1.72572	.1401386E 04	.2032468E-03	.3150099E 02	.1057080E 00	.1512256E-04	.1939318E-02	.7578463E 01	.1494766E 03
1.85072	.1234755E 04	.1790798E-03	.3171087E 02	.1064123E 00	.1331150E-04	.1707068E-02	.8620055E 01	.1700208E 03
1.97572	.1096323E 04	.1590026E-03	.3191766E 02	.1071062E 00	.1180606E-04	.1514010E-02	.9728694E 01	.1918874E 03
2.10072	.9796600E 03	.1420827E-03	.3210705E 02	.1077418E 00	.1054121E-04	.1351806E-02	.1090438E 02	.2150765E 03
2.22572	.8801297E 03	.1276475E-03	.3226771E 02	.1082809E 00	.9468344E-05	.1214221E-02	.1214711E 02	.2395879E 03
2.35072	.7943914E 03	.1152127E-03	.3239480E 02	.1087074E 00	.8550551E-05	.1096524E-02	.1345688E 02	.2654217E 03
2.47572	.7199197E 03	.1044119E-03	.3248589E 02	.1090130E 00	.7759376E-05	.9950632E-03	.1483370E 02	.2925779E 03
2.60072	.6548050E 03	.9496809E-04	.3254198E 02	.1092013E 00	.7072606E-05	.9069918E-03	.1627756E 02	.3210565E 03
2.72572	.5975946E 03	.8667071E-04	.3256815E 02	.1092891E 00	.6472701E-05	.8300599E-03	.1778847E 02	.3508575E 03
2.85072	.5471688E 03	.7935733E-04	.3257353E 02	.1093071E 00	.5945657E-05	.7624718E-03	.1936643E 02	.3819809E 03
2.97572	.5024169E 03	.7286685E-04	.3255248E 02	.1092365E 00	.5480150E-05	.7027750E-03	.2101143E 02	.4144267E 03
3.10072	.4625539E 03	.6708541E-04	.3250879E 02	.1090899E 00	.5066987E-05	.6497910E-03	.2272348E 02	.4481948E 03
3.22572	.4269242E 03	.6191794E-04	.3244590E 02	.1088789E 00	.4698630E-05	.6025528E-03	.2450257E 02	.4832854E 03
3.47572	.3669289E 03	.5321666E-04	.3226199E 02	.1082617E 00	.4080590E-05	.5232953E-03	.2826189E 02	.5574338E 03
3.72572	.3180479E 03	.4612733E-04	.3203924E 02	.1075142E 00	.3576472E-05	.4586471E-03	.3228940E 02	.6368718E 03
3.97572	.2777969E 03	.4028961E-04	.3179117E 02	.1066818E 00	.3159971E-05	.4052350E-03	.3658509E 02	.7215994E 03
4.22572	.2443223E 03	.3543471E-04	.3152629E 02	.1057929E 00	.2811946E-05	.3606042E-03	.4114896E 02	.8116165E 03
4.47572	.2162396E 03	.3136179E-04	.3125192E 02	.1048722E 00	.2518203E-05	.3229346E-03	.4598101E 02	.9069232E 03
4.72572	.1925029E 03	.2791920E-04	.3097611E 02	.1039467E 00	.2268041E-05	.2908538E-03	.5108124E 02	.1007519E 04

X	DELX	G	DELG	RADIUS	NDOT	RHODRP	MEAN RADIUS
.92572	.05000000	.00043668	.00043668	.4102821E-07	.1293921E 18	.7043566E 12	.4102821E-07
1.22572	.05000000	.33630150	.12917235	.3006707E-07	.5587431E 20	.5074788E 15	.3496398E-07
1.35072	.02500000	.79741888	.10226842	.2958053E-07	.2518761E 20	.8098124E 15	.3734238E-07
1.47572	.02500000	1.34568967	.11275868	.3000236E-07	.1375553E 20	.9029885E 15	.4044262E-07
1.60072	.02500000	1.91609053	.11416107	.3084336E-07	.5871292E 19	.8643484E 15	.4387141E-07
1.72572	.02500000	2.48007035	.11154598	.3193885E-07	.2108742E 19	.7809273E 15	.4725228E-07
1.85072	.02500000	3.02480909	.10703834	.3316260E-07	.6782238E 18	.6938583E 15	.5035530E-07
1.97572	.02500000	3.54369906	.10149754	.3441202E-07	.2074347E 18	.6160090E 15	.5309507E-07
2.10072	.02500000	4.03296441	.09537515	.3564983E-07	.6181483E 17	.5492370E 15	.5547891E-07
2.22572	.02500000	4.49093986	.08907345	.3683696E-07	.1850859E 17	.4923669E 15	.5754956E-07
2.35072	.02500000	4.91764873	.08288397	.3794640E-07	.5714726E 16	.4437598E 15	.5935557E-07
2.47572	.02500000	5.31425929	.07699641	.3895567E-07	.1865910E 16	.4019513E 15	.6094066E-07
2.60072	.02500000	5.68261015	.07150978	.3985300E-07	.6559556E 15	.3657490E 15	.6234143E-07
2.72572	.02500000	6.02482647	.06645243	.4064288E-07	.2492984E 15	.3342008E 15	.6358763E-07
2.85072	.02500000	6.34296513	.06180332	.4134879E-07	.1009637E 15	.3065457E 15	.6470292E-07
2.97572	.02500000	6.63914955	.05758089	.4194211E-07	.4497957E 14	.2821699E 15	.6570691E-07
3.10072	.02500000	6.91541952	.05375301	.4243022E-07	.2202255E 14	.2605762E 15	.6661574E-07
3.22572	.02500000	7.17364043	.05028376	.4282174E-07	.1181059E 14	.2413581E 15	.6744266E-07
3.47572	.05000000	7.63846189	.08849882	.4328169E-07	.9453040E 13	.2087682E 15	.6888326E-07
3.72572	.05000000	8.05122972	.07881771	.4348930E-07	.4948282E 13	.1823348E 15	.7011168E-07
3.97572	.05000000	8.42032981	.07067130	.4350470E-07	.3217063E 13	.1606031E 15	.7117311E-07
4.22572	.05000000	8.75251126	.06376064	.4337197E-07	.2492444E 13	.1425233E 15	.7210066E-07
4.47572	.05000000	9.05321681	.05784621	.4313149E-07	.2209091E 13	.1273226E 15	.7291910E-07
4.72572	.05000000	9.32682598	.05273105	.4282645E-07	.2140046E 13	.1144219E 15	.7364724E-07

**** ALL INPUT DATA HAVE BEEN PROCESSED.
AT LOC 43552

```
JOB NO. 999234      MONDAY, JULY 06, 1964                1 27 17.5 PM          1 27 54.8 PM

    SIVIER  (AERO. ENG.)          S418F              002   200   000

    999234SIVIERS418FC070664484375    55    15    0    0    0    25    14        $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $
CARDS PROCESSED     264,   LINES PRINTED    144,   PAGES PRINTED     7,   CARDS PUNCHED     0


       ELAPSED TIME --

       TOTAL              PROCESSING          EXECUTION          LOADING
        0' 33.0''          0'  8.4''           0'  9.5''          0' 15.0''



SEE NOTICE ON BULLETIN BOARD DATED 6/8/64 CONCERNING REPLACEMENT OF MANY LINEAR SYSTEM SUBROUTINES (GJRDT ET AL).
```

Unclassified

Security Classification

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) Aerospace Research Laboratories Thermomechanics Research Laboratory	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

A Digital Computer Program for Condensation in Expanding One-Component Flows

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

TDR Interim - continuous from Nov 63

5. AUTHOR(S) (Last name, first name, initial)

Leonard J. Harding

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1965	73	3

8a. CONTRACT OR GRANT NO. AF 33(657) 8867	9a. ORIGINATOR'S REPORT NUMBER(S) ARL 65-58
b. PROJECT NO. 7116	
c. Task No 7116-01	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. AVAILABILITY/LIMITATION NOTICES

Copies may be obtained from DDC

Released to OTS.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	USAF

13. ABSTRACT

This report describes a digital computer program for calculating vapor condensation processes that occur in rapidly expanding flows. The treatment emphasizes the program logic required to satisfy the requirements of the mathematical model. Among the most important of these requirements are the search for the onset of nucleation, in an isentropically expanding flow, and the iteration procedure necessary for the joint solution of the nucleation and growth equations and the diabatic flow equations. The program features flexibility in allowing input conditions, short execution time, and a convenient output format.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
______________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
______________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
______________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.