









WANL-TME-1001 October, 1964

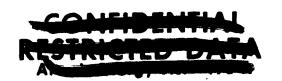
DIRE - A DIGITAL REACTOR MODEL FOR EXCURSION STUDIES

J. W. Riese

G. Collier

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Astronuclear Laboratory
Westinghouse Electric Corporation

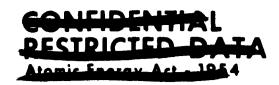


DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.





Astronuclear Laboratory

Box 10864, Pittsburgh, Pa 15236 Telephone 892-5600

MASTER

November 10, 1964

* Mr. J. L. Dooling
NERVA Subcontract Manager
Rocket Engine Operations - Nuclear
Department 7140
Aerojet-General Corporation
Post Office Box 1947
Sacramento, California

Subject:

WANL-TME-1001, "DIRE - A Digital Reactor Model for Excursion Studies,"

dated October 27, 1964

Dear Mr. Dooling:

Transmitted herewith are five (5) copies of the subject report. This

report is transmitted for your information.

Respectfully,

H. F. Faught

Program Manager NERVA Nuclear Subsystem

Enclosures - 5

SNPO-C Pittsburgh Resident Office, w/o enclosure
AGC Pittsburgh Resident Office, w/o enclosure

ruic

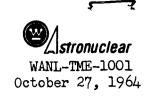
DISTRIBUTION OF THIS DOCUMENT IS UNLINET.

RESTRICTED DATA

WHEN SEPARATED FROM ENCLOSURES, HANDLE THIS DOCUMENT AS UNCLASSIFIED









SUBMITTED BY:

Westinghouse Electric Corporation
Astronuclear Laboratory
Pittsburgh 36, Pennsylvania

This report was prepared as an account of work sponsored by the United States Government. Nether the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

DIRE - A DIGITAL REACTOR MODEL
FOR EXCURSION STUDIES

PR	\mathbf{E}	РΔ	Ri	T)	B	Y	•

J. W. Riese

G. Collier

APPROVED BY:

D. W. Drawbaugh

INFORMATICA CATEGOLY

CONFIDENTIAL RESPRICAD DATA

Authorized Clas Ific - Date

GROUP

Excluded from Automatica Downgrading and Declassification

SPECIAL PLREVIEW Reviewer Class. Date
PROFIL
DETERMINATION
Class: U. Reviewer Class. Date

DISTRIBUTION DE THIS DOCUMENT IS UNLIMITED



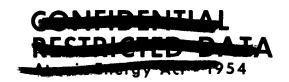




Table of Contents

⊥.	INT	RODUCTION	1
II.	DES	CRIPTION OF CODE	3
	Α.	Physical Model, Equations	3
		A.l. Neutronics	3
		A.2. Heating of the Solid	4
		A.3. Heat Transfer to the Coolant	7
		A.4. Heat Removal by the Coolant	8
		A.5. Flow vs. Pressure: Nozzle Equation	10
		A.6. Hydrogen Equations of State	15
		A.7. Reactivity Feedback	17
	В.	Method of Solution	18
		B.l. Choice of Time-Step Size	18
		B.2. Iterative Procedure	18
III.	INP	UT PREPARATION	21
IV.	REF	ERENCES NOTICE	26
٧.	FOR	TRAN LISTING This report was prepared as an account of work sponsored by the United States Government. Neither sponsored by the United States Government. Neither the United States nor the United States Energy the United States nor any of their contractors, their employees, nor any of their contractors, subcontractors, or their employees, makes any subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal warranty, express or implied, or assumes any legal warranty or responsibility for the accuracy, completeness is ability or responsibility for the accuracy, completeness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.	27

DISTRIBUTION OF THIS DOCUMENT UNLIMITED









I. INTRODUCTION

The code described in this report was written about a month prior to the September NRX-A2 cold flow tests (EP-1B). These tests were scheduled to be run at a fixed low reactor power level of about 1 kilowatt or 1 millionth of full power. It was desired to analyze the nuclear excursions resulting from some postulated flow rate and/or control drum accidents. The usefulness of the TNT code for this type of problem was limited because: the very rapid neutronics changes occurring here required the use of very small time-steps, otherwise the code became unstable; and each time step required a fair amount of computing time because of the very detailed heat transfer and fluid flow model. Hence computer running times (to do a problem satisfactorily) became excessive. We proposed to remedy this situation by: (1) using a much simplified heat transfer-fluid flow model, hence reducing the computation per time-step; and (2) using many small neutronic time-steps for each single fluid flow and heat transfer step, thus reducing the number of time-steps for the nonneutronic part of the calculation.

It was at first believed that the code VARI-QUIR II (ref. 8) could handle the problem directly, with no further programming. A closer look, however, revealed that the extreme low temperatures involved in this problem were quite outside the range of assumptions



د بو

RESTRICTED DATA



made in the fluid-flow portions of VARI-QUIR II.* In addition, we desired a heat transfer-fluid flow analysis in somewhat greater detail than was present in the VARI-QUIR II model.

For the above reasons, we wrote a completely new code tailored to the needs of the cold-flow accident problem. The neutronics was kept very simple - 1 energy group, 6 precursors, point kinetics - since this is adequate for the accuracy needed in an accident calculation. The fluid flow treatment was of necessity fairly detailed, however, since the low temperatures involved in cold flow are far indeed from the "ideal gas" conditions present in a power test.

As an added bonus, it was believed that this more accurate heat transfer-fluid flow model could later be coupled with the quite detailed neutronics of VARI-QUIR to provide an improved code for a variety of transient problems.

The ability of this model to accurately represent a NERVA reactor may perhaps be improved by changing some of the input data we have used. It may even be necessary to modify the code somewhat, by including details which we have neglected, if they are found important. But the basic structure of the code should remain unchanged, and it should be a useful tool for transient analysis. It provides a one-pass

*As a matter of fact, in TNT as well, the nozzle equations assume an ideal gas. This simplification is generally inaccurate at the low temperatures involved in a cold flow test and becomes completely meaningless if two-phase flow enters the nozzle.







program to replace the iterative procedure (previously used in accident calculations) which was carried out between such codes as TNT and RTS, and which involved intermediate hand calculations besides.

An example of the use of this code is described in reference #15.

II. DESCRIPTION OF CODE

A. Physical Model, Equations

A single channel is used to represent flow through the reflector, and another channel, in series with the first, to represent flow through the core. Each channel is divided into an arbitrary number of equal axial increments (we have used 10 core and 10 reflectors in the problems run thus far). Each axial increment is thus a horizontal slice, including both an element of fluid and the surrounding solid which is (generally) heating it.

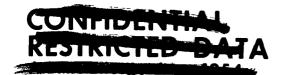
A.l. Neutronics

The very simplest model is used here: 1 energy group, 6 precursors, point kinetics (i.e., the spatial shapes of the flux and precursors are assumed constant for all time, only their magnitudes varying). Thus the equations solved are:

$$\frac{d\phi}{dt} = (k - 1 - \beta k) + \sum_{j=1}^{6} \lambda_{j} C_{j}$$
 (1)









and j = 1 to 6:

$$\frac{\mathrm{d}^{\mathbf{c}}_{\mathbf{j}}}{\mathrm{d}^{\mathbf{t}}} = -\lambda_{\mathbf{j}}^{\mathbf{c}}_{\mathbf{j}} + \boldsymbol{\beta}_{\mathbf{j}} \times \underline{\boldsymbol{\phi}}$$
(1)

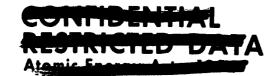
In the problems run thus far, we used $\ell=2.4\times 10^{-5}$ sec., $\beta=.0078$, and delayed neutron constants $\beta_{\rm j}$ and $\lambda_{\rm j}$ as listed below. The multiplication factor k varies with time, depending on control drum position and feedback.

j —	$\underline{\boldsymbol{\beta}_{\mathfrak{j}}}$	λ_{j} (sec. ⁻¹)
1 2 3 4 5	.000203 .001000 .003175 .001466 .001660 .000296	3.87 1.40 0.311 0.115 0.0317 0.0127

A.2. Heating of the Solid

Power is assumed to be always proportional to the flux, the constant of proportionality being chosen one, at the beginning of the problem, in such a way as to bring the starting power to the desired level. This power is a heat source, distributed between core and reflector in the ratio (for the problems run thus far) 0.978 to 0.022 (see ref. 1). The axial distribution of power in both core and reflector, was taken as in reference 2. Thus our input data for power fractions was as follows.







	Reflector Element No.	Power Fraction	Core Element No.	Power <u>Fraction</u>
	1	.000509	1	.07674
	2	.001261	2	.10330
	3	.001936	3	.12450
	4	.002512	4	.13630
	5	.002755	5	.13820
	6	.003112	6	.12250
	7	.003065	7	.11170
	8	.002800	8	.08608
	9	.002320	9	.05608
	10	.001730	10	.02260
Totals:		.022000		.97800

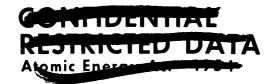
Note that our numbering system proceeds always in the direction of fluid flow; i.e., the nozzle is adjacent to reflector element #1 and core element #10. Thus, if the foregoing power fractions were plotted versus element number, the reflector curve would be the mirror image of the core curve.

These power fractions are input quantities, and can be changed at will. However, care should be taken to make them add up to 1.0, as there is no internal check for this in the code.

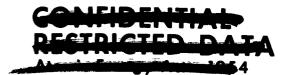
Given these power fractions F_i and the total power P, the temperature rise of any axial element of solid follows the equation

$$\frac{dT_{\text{Wi}}}{dt} = \frac{1}{C_{i}} \cdot \left[P \cdot F_{i} - Q_{i}\right]$$
 (2)

where $\mathbf{Q}_{\hat{\mathbf{I}}}$ is the rate of heat loss to the fluid, to be discussed in the next section.



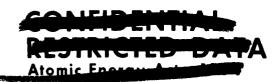






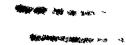
The heat capacity C_1 is determined by the code as a product of two factors: an <u>input</u> value, giving the heat capacity of a core element at $4000\,^{\circ}\text{R}$ and a reflector element at $600\,^{\circ}\text{R}$; and a temperature-dependent reduction factor, built into the code and <u>not</u> available to input data, which reduces the heat capacity of each particular element as its temperature varies below these particular values. The temperature-dependent reduction factors have been obtained from the specific heat curves of reference 4, using the component weights of reference 3. Thus the core factor F_C is a linear interpolation between the points $F_C = 0$ at T = 0, 0.73 at T = 1120, 0.95 at 2600, and 1.0 at 4000, with linear extrapolation above 4000°R using the 2600-4000 slope. Similarly, for the reflector, $F_R = 0$ at T = 0, 0.317 at 160, 0.449 at 300, 1.0 at 600, and 1.31 at $1000^{\circ}R$.

The variable <u>input</u> data to the code consists of two numbers: for the reflector, the reciprocal heat capacity, $1/C_i$, in °R/Btu, for a single axial reflector solid element (assumed the same for each element) at 600°R and a similar number for a core element at 4000°R. For the problems run thus far, with <u>10 axial elements</u>, we have used 6.6707 x 10^{-3} °R/Btu for a core element, and 6.2247 x 10^{-3} °R/Btu for a reflector element; these numbers are again based on lumping the components listed by weights in reference 3, using the specific heats of reference 4. Note that, for identical material assumptions, these numbers must be <u>increased</u> in direct proportion to the number of axial elements into which a channel is divided, since they give the reciprocal heat capacity of each element.











A.3. Heat Transfer to the Coolant

In the flow of heat from solid to the coolant, it is assumed that the most important "resistance" is the heat transfer coefficient at the interface; temperature gradients within the solid are regarded as either neglected or lumped with the heat transfer coefficients at the interface. Thus the rate of heat loss to the coolant (for a core element, for instance) is given by:

$$Q_{Ci} = h_{Ci} \cdot \left[T_{WCi} - T_{HCi} \right]$$
 (3)

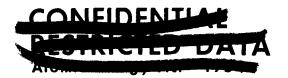
subscripts W, H, and C referring to solid (wall), hydrogen, and core respectively; a similar equation holds in the reflector. The heat transfer coefficients h are computed from (see ref. 5)

$$h_{Ci} = X_C \cdot k_i^{0.6} \cdot W^{0.8} \cdot (\frac{C_{pi}}{\kappa_i})^{0.4} / (1 + M_i)$$
 (4)

where W is the flow rate; k_i , C_{pi} , and μ_i are the gas conductivity, specific heat, and viscosity; X_C is a constant factor depending only on geometry; and M_i is zero except for two-phase flow. When two-phase flow is present, the Martinelli correction for hydrogen is, from reference 6,

$$M_{i} = less of \left\{ \frac{9.0}{3.16 \left(\frac{1-x}{x}\right)^{0.9}} \left(\frac{\mu_{\ell}}{\mu_{g}}\right)^{0.1} \left(\frac{\rho_{g}}{\rho_{\ell}}\right)^{0.5} \right\}$$
 (5)

where x is the fluid quality, and μ and ρ the viscosity and density of gas (subscript g) or liquid (subscript ℓ) phases. We have inserted the cut-off value of 9.0 because experimental data does not extend beyond



♣、奶糖糖





this point, and simple extrapolation of the power functions in (5) to the pure liquid (x = 0) case would be in obvious error.

The only undetermined constants in (4) are two numbers, X_C for the core and X_R for the reflector. We have adjusted these numbers to give roughly correct (see ref. 7) values for the temperature drops between solid and coolant in the 100% power situation. (Since, at steady-state, Q_i is determined from equation 2 as equal to PF_i , changing the heat transfer coefficient will merely vary the temperature drop in (3), and hence the wall temperature.) For the code as it stands presently, the input numbers we have used (which include dimensional conversion factors) are $X_C = 18.2$, $X_R = 12.6$.

Again note that, with no change in physical assumptions (i.e., to obtain the same temperature distribution), if we <u>increase</u> the number of axial regions in the core or reflector above the value 10 used here, we must decrease $X_{\mathbb{C}}$ or $X_{\mathbb{R}}$ by the same ratio (because they include a factor for channel element length).

A.4. Heat Removal by the Coolant, Energy Conservation

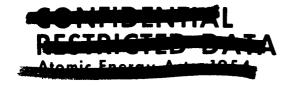
Having followed the energy from its birth in

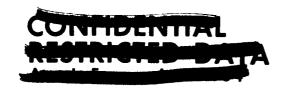
fission, its heating of the solid, and then the transfer of that heat

to the coolant, we are finally ready to consider its being carried

out through the channels by the coolant. As shown in reference 8,

the times required for fluid flow irregularities to disappear are very







short (~ .001 sec. or less). We shall therefore neglect these time lags completely and regard the fluid at any time as being in a quasi-steady-state, i.e., in thermal and mechanical equilibrium with the momentary wall temperatures and nozzle pressures. Again based on the discussion in reference 8, we may neglect both frictional heating of the fluid and its kinetic energy of in-channel flow, compared with the large amounts of energy coming into the fluid from the heated solid walls.

With the above simplifications, the energy equation reduces to the simple statement that heat loss from solid to fluid, in an axial increment, equals the excess of outgoing over incoming fluid enthalpy for that increment. I.e.,

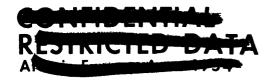
$$Q_{i} = W \cdot \left[H_{i}' - H_{i-1}'\right] \tag{6}$$

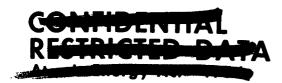
where Q_i is the heat energy per unit time passing from solid to surrounding fluid, W is (as in 4) the fluid flow rate, and $H_i^{'}$ is the fluid enthalpy per unit mass at the downstream edge of the ith element.

Since all of our other equations have referred to average quantities over an element, whereas (6) involves the edge quantity H_{i} , we must make a connection between the two. To do this we use the simple arithmetic mean:

$$H_{i} = (H_{i}' + H_{i-1}')/2 \tag{7}$$

where Hi is the average enthalpy per unit mass in the ith fluid element.





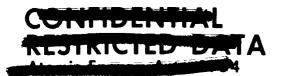


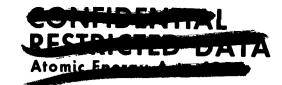
A.5. Flow vs. Pressure Drop: Nozzle Equation

The coolant pressure drops along both core and reflector channels are fairly small compared with the pressure drop through the nozzle (see ref. 9); it is this nozzle pressure drop which essentially determines the coolant flow rate through the reactor. We shall therefore assume, for simplicity, that there is no in-channel pressure drop - i.e., that the pressure of the gas is constant from reflector entrance to core exit and drops only on passage through the nozzle.

We need a relation for this nozzle pressure drop as a function of flow rate so that, given the ambient pressure, we can compute the in-channel fluid pressure, and thence other fluid properties. Such a relation is derived below, following the approach of reference 10 but replacing their ideal gas assumption with arbitrary fluid properties. Thus our results should be good for a cold gas outside the ideal range and even down to two-phase flow through the nozzle.

Since we are not interested in quantities like total thrust, but only flow rate and pressure, it was deemed sufficient to follow the fluid only as far as the nozzle throat and to neglect the diverging portion of the nozzle. (The error involved in this approximation will be discussed at the end of this section.) Thus we consider a simple converging nozzle. In what follows let subscript P refer to conditions in the core exit plenum, upstream from the nozzle;









subscript T, the nozzle throat; and subscript A, the ambient conditions (beyond the nozzle throat, i.e., outside the reactor).

Assuming adiabatic flow through the nozzle (i.e., no heat transfer from the walls), the total fluid energy (enthalpy plus kinetic energy) will be constant from core exit plenum through nozzle. If the plenum has a large area, so the velocity there is negligibly small, we thus have

$$H_{p} = H_{T} + v_{T}^{2} / 2g \tag{8}$$

where v is velocity and H is enthalpy per unit weight of fluid. The weight flow rate W is given by the product of velocity, weight density, and flow area, and is constant from point to point, by conservation. Thus, evaluating it at the nozzle throat,

$$W = A_T v_T \boldsymbol{\rho}_T$$
 (9)

Combining (8) and (9)

$$W = K \mathbf{e}_{T} \sqrt{(H_{P} - H_{T})}$$
 (10)

where the nozzle constant K involves only geometry and dimensional conversion factors.

Given the plenum and ambient conditions, we must therefore determine the nozzle throat conditions \boldsymbol{e}_T and \boldsymbol{e}_T in order to use equation 10. It will be sufficient to determine any two fluid state variables at the nozzle throat, since all the rest can then be









found from the equations of state. One variable, the entropy, can immediately be determined by assuming that the nozzle flow is not merely adiabatic, but reversible (i.e., isentropic):

$$S_{\mathbf{p}} = S_{\mathbf{p}} \tag{11}$$

The other fluid state variable we wish to determine at the throat is the pressure. For low fluid velocities, i.e., for small enough pressure drops from plenum to exterior, the throat pressure is simply in equilibrium with the immediately adjacent* ambient pressure:

$$P_{T} = P_{A}$$
, for small $P_{P} - P_{A}$. (12)

With P_T and S_T determined from (11) and (12), the other thermodynamic variables P_T and H_T are given by the fluid equations of state, and the flow rate may then be found from (10).

However, as we increase the pressure drop $P_P - P_T$ by decreasing P_A , it is found that the flow rate as calculated from (10) rises to a maximum and then begins to decrease. That this decrease does not occur physically may be shown as follows. At the maximum W, we have:

$$\left(\frac{\delta}{\delta}\right)_{\text{PT}}^{\text{T}} = 0$$

*Recall that our model is simply a converging nozzle, which ends at the throat.







where the derivative is taken at constant entropy because (11) always holds. Using our form (10) for W, the preceding equation becomes:

$$\frac{1}{\mathbf{e}_{\mathrm{T}}} \left(\frac{\mathbf{\delta} \mathbf{e}_{\mathrm{T}}}{\mathbf{\delta}_{\mathrm{P}_{\mathrm{T}}}} \right)_{\mathrm{S}_{\mathrm{T}}} - \frac{1}{2(\mathrm{H}_{\mathrm{P}} - \mathrm{H}_{\mathrm{T}})} \left(\frac{\mathbf{\delta}_{\mathrm{H}_{\mathrm{T}}}}{\mathbf{\delta}_{\mathrm{P}_{\mathrm{T}}}} \right)_{\mathrm{S}_{\mathrm{T}}} = 0$$

or:

$$2(H_{P} - H_{T}) = P_{T} \left(\frac{\partial H_{T}}{\partial P_{T}} \right) S_{T}$$
(13)

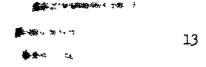
But, from elementary thermodynamics, $e(\frac{\delta H}{\delta e})_S = (\frac{\delta P}{\delta e})_S = C^2$; using this replacement for the right-hand side of equation (13) and equation (8) for the left-hand side, (13) becomes simply

$$v_{\rm m}^2 = c_{\rm m}^2$$
 (13-a)

which occurs when P_T is reduced (following P_A , equation 12) to the point where the W calculated in (10) is a maximum. In other words, the fluid velocity at the nozzle throat has reached sonic velocity; the flow is "choked". Further reductions in ambient pressure P_A , although mathematically (equation 10 and 12) leading to a drop-off in flow rate, cannot physically be telegraphed back into the nozzle throat because the speed of propagation of a pressure pulse is just matched by the speed of fluid flow.

To summarize, as the ambient pressure P_A is reduced below the critical pressure P* corresponding to maximum W, the throat pressure P_T will not follow P_A as in (12), but will remain









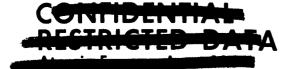
constant (for constant P_p) at P*. Therefore, to cover the range of all flow rates and pressure drops, equation (12) must be replaced by:

$$P_{T} = lesser of \begin{cases} P_{A} \\ P^{*} \end{cases}$$
 (14)

where P* is the critical pressure for given plenum conditions; i.e., with fixed H_p , S_p , and $S_T = S_p$, P* is that value of P_T which maximizes W in equation (10).

Equations (10), (11) and (14) along with the foregoing definition of P* and the equations of state suffice to determine W from the plenum conditions and the ambient pressure. The nozzle constant K (equation 11) may be fixed by requiring the known full-power plenum temperature and pressure to lead to the known full-power flow rate (ref. 7). For this "discharge coefficient" we have used as input data, in the problems run thus far, DISCH = 82.5.

As mentioned earlier in this section, the actual converging-diverging nozzle is replaced in our model by a simple converging nozzle. Since our constant K is adjusted at full power, where the flow is choked, and in choked flow the conditions beyond the nozzle throat cannot be felt in the throat, our model will be completely accurate in the choked flow range. For very low flow rates (crossing into subsonic nozzle throat velocities) the pressure just beyond the nozzle throat does have an effect on flow rate. Our model will take this pressure as ambient, whereas actually it will be somewhat below



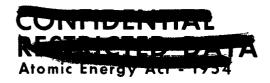


ambient, due to the velocity drop (and hence pressure rise) in the diverging section of the nozzle. In this range, however, the flow rate W is a maximum and therefore varies only slightly with throat pressure anyway. At still lower flow rates, the nozzle pressure drop is so small that it matters not how accurately we calculate it; the pressure inside the reactor will be roughly atmospheric, in any case.

The advantage of our treatment lies in the arbitrariness of the fluid state equations it will allow. For an ideal gas at choked flow, equations (10), (11) and (14) simplify to the well-known W = $\propto P_p/\sqrt{T_p}$ (see, e.g., ref. 11). But they will handle with equal accuracy any other equation of state, even down to two-phase flow, and thus should be suitable for treating cold flow problems.

A.6. Hydrogen Equations of State

In all of the foregoing sections, it was assumed that we would have available the hydrogen equations of state, relating the various thermodynamic properties of the fluid. Except for singular cases, any two thermodynamic state variables will be sufficient to determine the state of the sytem and hence the remaining properties. We should prefer the two <u>independent</u> variables to be enthalpy and pressure since the fluid enthalpy will be directly obtainable from equations like (6) and (7) and the pressure from the







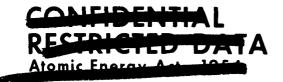


calculations of section A.5. In addition, enthalpy has the advantage that it varies continuously with quality in a two-phase system at constant temperature and pressure, and hence avoids the multi-valued, singular case problem mentioned above.

A code which meets these requirements has been written at LASL (references 12 and 13). Its existence was pointed out to us by B. L. Pierce, who also furnished us with a binary deck. This routine was checked out over a wide range of enthalpies and pressures and found to give reasonable agreement with published measurements (e.g., ref. 14). It was then incorporated into our program as a subroutine.

It would have been particularly useful to us if the LASL subroutine had had an option whereby pressure and entropy. rather than pressure and enthalpy, could be used as the independent variables. This would have simplified the solution of the nozzle equations since entropy, rather than enthalpy, is constant through the nozzle (see equation 11). Like enthalpy, entropy also varies continuously with quality across the two-phase region, thus giving unambiguous results.

However, this option was not available to us. Therefore equation (11) had to be satisfied by guessing an enthalpy $H_{T\!\!\!/}$ and iterating until the entropy $S_{T\!\!\!/}$ was correct.









A.7. Reactivity Feedback

The code assumes a single temperature coefficient of reactivity for the reflector, a single temperature coefficient for the core, and a hydrogen density coefficient for each axial element of the core. This procedure was recommended by A. L. Mowery of Reactor Analysis, who als supplied the actual numbers we used in this study. The core and reflector temperature coefficients were taken as:

$$\frac{\delta k}{\delta T_C} = -8.104 \times 10^{-6} / R$$

$$\frac{\mathbf{\lambda} k}{\mathbf{\delta} T_{R}} = +7.3554 \times 10^{-6} / ^{\circ}R$$

The core hydrogen density coefficients which were used are as follows:

Core Element No.	$\frac{\Delta k}{\Delta \text{In-Channel H}_2 \text{ Density}}$, $\frac{\text{ft.}^3}{\text{lb.}}$
1	.00430
2	.01257
3	.01947
4 5	.02474
5	.02778
6	.02819
7	.02393
8	.01622
9	.01196
10	.00426

These hydrogen density coefficients were calculated for our 10-element axial mesh, starting from Mowery's values for 6 unequal axial increments.









B. Method of Solution

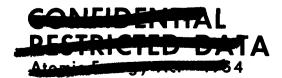
B.l. Choice of Time-Step Size

This is easily the most important problem in a code of this type and can determine its success or failure. Too large a time-step can make the code unstable because of the very short neutron lifetime involved; while too short a time-step can make the computer running time prohibitively long - this is mainly because the fluid flow and heat transfer calculations for each step are so involved.

In the code, therefore, not only is the time-step of variable size, but <u>several</u> neutronics time-steps are used for a <u>single</u> feedback step. The neutronics time-integration is handled by the 7094 System Subroutine ICE, which chooses its own time-step, of whatever size is necessary to meet a specified error criterion (see Section III, card group 11). A feedback calculation is done whenever any one of three criteria are met: (1.) whenever a time PP (card group 11) has elapsed since the last feedback calculation group 11) has elapsed since the last feedback calculation reaches a certain maximum DTFD (card group 11); and (3.) if the power has changed by a factor of 2 since the last feedback calculation.

B.2. Fluid Flow-Heat Transfer Iterative Procedure

There are two options in the fluid flow portion of the code. The first is that, in addition to the fluid enthalpy coming into the reflector, one can specify either fluid pressure or



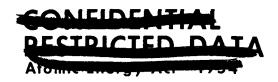




fluid flow rate (see Section III, card groups 4 and 9). The second option allows the transient problem to start from <u>either</u> a steady-state condition (in which case the wall temperatures are determined by the fact that the heat flow is in equilibrium) <u>or</u> a transient condition, in which case wall temperatures must be input (this second choice would be useful for continuing a previously-started problem).

If we choose to start from equilibrium, there is a steady-state problem to solve first. This is done as follows. Since the time derivative in equation (2) is 0, the heat loss Q_i of any solid section is exactly equal to the heat generation rate PF_i . Assuming that the fluid flow rate W has been given, equations (6) and (7) allow immediate determination of the fluid enthalpy at each axial position, starting from the given input enthalpy to the reflector and proceeding all the way to the core outlet plenum. The nozzle equations (Section II., A.5.) now allow us to determine fluid pressure from plenum enthalpy and flow rate. Knowing this pressure and the previously calculated enthalpies, the fluid state at every axial position is completely prescribed. Therefore we may calculate both the heat transfer coefficient and the fluid temperature at each axial position and finally the wall temperature from equation (3).

If pressure rather than flow rate has been specified, we guess a flow rate, carry through the above calculation to the point of pressure determination, compare with the specified



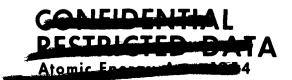




pressure, guess a new flow rate, and iterate until satisfactory accuracy has been achieved. The remainder of the calculation is precisely as before.

If the "transient-start" option is used, the initial wall temperatures are also input to the code (see Section III, card group 9B and 9C). We now know all the wall temperatures, the enthalpy into the reflector and either the fluid pressure or the fluid flow rate depending on the option chosen. The first step is therefore to guess the other variable (i.e., if flow rate is read in, guess the pressure, and vice-versa). It is now straightforward to go through the reactor element by element, knowing the pressure, flow rate, wall temperatures and the enthalpy of the preceding element; to do an "inner iteration" at each element, using equation (3), (4), (6) and (7), determining the fluid enthalpy, heat transfer coefficient, and heat loss rate of that element; and thus to come to the core output plenum with known enthalpy and either pressure or flow rate known exactly, the other having been guessed. The "guessed variable" is now solved for, using the nozzle equation, and this new guess used to repeat the entire process; iteration continuing until satisfactory accuracy has been achieved.

In the time-dependent problem, the feedback calculation for each time-step begins by integrating equation (2) to obtain the change in wall temperature, and hence a new wall temperature, for each axial element. The old heat loss rate Q is used in this calculation, but the power P is actually integrated over the entire







time-step from the neutronics equations. At this point, with the new wall temperatures given, and our quasi-steady state assumption on the fluid flow (see Section II, A.4.), the calculational procedure of the preceding paragraph is applied in its entirety to conclude the fluid-heat transfer determination for this time-step.

III. INPUT PREPARATION

In the following "Card Input Structure"

- 1. Start each card group with a new card
- 2. Use as many cards as needed in each card group

Card Group Number	Format	FORTRAN Variable	Description
(1)	2014	NJ	Number of precursors, NJ ≤ 6
(2)	8E10.4	TLIFE	Neutron prompt lifetime
		FLUX	Flux at time = 0, in arbitrary units (code will print out later values relative to this one)
			Fraction $oldsymbol{\mathcal{B}}$ and decay constant $oldsymbol{\lambda}$ for each precursor group
(3)	Binary Cards	Hydrogen data	These data cards are a table of hydrogen properties the TAB2 subroutine uses (see refs. 12 and 13)







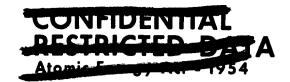


Card Group Number	Format	FORTRAN Variable	Description
(4)	2014	NPOFL	<pre>= 1 If pressure is given = 2 If flow is given</pre>
		NRE	Number of axial reflector regions, NRE < 20
		NC	Number of axial core regions, NC ≤ 20
		NST	<pre>= l If problem starts from a steady-state (equilib- rium) condition = 2 If problem starts from a transient (non- equilibrium) condition</pre>
(5)	8E10.4	RCAPR	Reciprocal heat capacity for each reflector region
		RCAPC	Reciprocal heat capacity for each core region
		XHTR	Reflector heat transfer constant, proportional to "h"
		XHTC	Core heat transfer constant, proportional to "h"
		XMART	Two-phase flow constant - Martinelli
		DISCH	Constant which determines flow rate vs. pressure drop
(6)	8E10.4	POWFR(I), I=1, NRE	Power fraction for each reflector region follow-ing the direction of the flow
(7)	8E10.4	POWFC(I), I=1, NC	Power fraction for each core region following the direction of the flow



22. ,

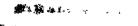
\$ 1. 5m2 12 25





Card Group Number	Format	FORTRAN Variable	Description
			Market and the control of the Contro
(8)	8E10.4	EIGRT	Reflector temperature reactivity coefficient of reactivity (°R ⁻¹)
		EIGCT	Core temperature reactivity coefficient of reactivity (°R ⁻¹)
		EIGCH(I), I=1, N	Hydrogen density reactivity coefficient for each core region (ft./lb.)
(9)	8E10.4	POWST	Starting power (BTU/sec.)
		HIN	Input enthalpy into reflector (e.g100 BTU/lb. for liquid hydrogen)
		POUT	Ambient pressure (lbs./in ²)
		POFL	Pressure or flow rate at starting time. Which it is depends upon the value of NPOFL. (lbs./in2 or lbs/sec.)
		POWINT	<pre>Integrated power (Btu); = 0 if starting a problem from scratch otherwise = final integrated power from earlier problem</pre>
Card groups	(9)A to (9)D	are only included in t	his sequence if NST = 2
(i.e., only	if continuing	a problem)	
(9)A	8E10.4	Т	Starting time (sec.)
		HP	Core outlet enthalpy (btu/lbs.) a rough guess





to start problem





Card Group Number	Format	FORTRAN Variable	Description
	TOTINGO	1 Ottitati Val Lable	<u>Beset iputen</u>
(9) B	8E10.4	TWR(I), I=1, NRE	Wall temperatures for each of the reflector regions
(9)C	8E10.4	TWC(I), I=1, NC.	Wall temperatures for each of the core regions
(9) D	8E10.4	PREC(I), I=1, NJ	The non-equilibrium starting values of the precursors
(10)	8E10.4	Т	Starting time (sec.)
(11)	8E10.4	DT	∆T for the specified pressure or flow ramp
		PP	Maximum allowable integration step. Also the feedback is checked at this time
		El	Lower bound error for controlling ICE step size (e.g. 1.0E - 06)
		E2	Upper bound error for controlling ICE step size (e.g. 1.0E - 04)
		DTFD	Maximum change in \$\int FLUX.\dt (i.e., max.)\$ \$\Delta t \times flux \text{ which}\$ can occur in the neutronics before a fluid flow-heat transfer calculation is done. Here FLUX is relative to the input starting flux, card group 2









Card Group Number	Format	FORTRAN Varible	Description
(12)	2014	MORT	<pre>= 1 If drum reactivity is specified = 2 If total is specified</pre>
		NSPP	<pre>= Number of time-steps (of size PP) per print- out by the code</pre>
		NSTOP	= 2 If this is the final ramp of a problem ≠ 2 otherwise
Card group ()	3) depends	upon the value of MORT.	If MORT = 1 then
(13)	8E10.4	XMLTM2	= control drum reactivity change, △k, occurring over this △T ramp (△T card group 11)
If MORT = 2 t	hen		
(13)	8E10.4	XMLTT2	- total reactivity (drum + feedback) 1.0 + \(\Delta \) k at the end of this \(\Delta \) T
(14)	8E10.4	POFL2	Pressure or flow rate at the end of this $oldsymbol{\Delta}$ T

Repeat card groups (11) through (14) as required to define problem.









IV. REFERENCES

- 1. WANL-TME-483, "NRX-A Reactor Model for Systems Analysis", Knecht, W. and Howarth, W. L., July 31, 1963, p. 15.
- 2. Ibid., p. 23.
- 3. Ibid., p. 20.
- 4. Ibid., pp. 26-29.
- 5. Heat Transmission, McAdams, W. H., McGraw-Hill, New York, 1954, p. 219.
- 6. NASA-TN-D-765, "Experimental Heat Transfer and Pressure Drop of Liquid Hydrogen Flowing Through a Heated Tube" Hendricks, R.C., Graham, R. W., Hsu, Y. Y., and Friedman, R., May 1961, pp. 36, 52.
- 7. WANL-TNR-128, "Reactor Analysis of NRX-A: Vol. III, Thermal and Fluid Flow Analysis", September 1963, pp. 3.14, 6.126, 8.101.
- 8. WANL-TME-684, "VARI-QUIR II: Space-Time Neutron Diffusion with Feedback", Riese, J. W., Collier, G. and Dickson, P. W., February 1964, pp. 17-24.
- 9. Ibid., p. 25.
- 10. Fluid Mechanics, Binder, R. C., Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1955, pp. 284, 298-300.
- 11. WANL-TME-103, "NERVA Power Range Analysis", Maguire, A. F., August 15, 1962, p. 10.
- 12. LASL-N-4-2285, "Tabular 7094 Subroutine for Para-Hydrogen Properties", Farmer, O. A., March 23, 1964.
- 13. LASL-N-4-2291, "Operating Instructions for TAB Code H6312P", Farmer, O. A., April 7, 1964.
- 14. NBS Technical Note 130, "Provisional Thermodynamic Functions for Para-Hydrogen", Roder, H. M. and Goodwin, R. D., December 1961.
- 15. WANL-TME-761, Supplement 6, "Supplement to NRX-A2 Safety Evaluation Report", October 8, 1964.
- 16. ICE-3, Integration with Controlled Error, N. Moraff, Westinghouse Air Arm, Baltimore, October 20, 1959; SHARE Distribution Nos. 411 and 447.





V. FORTRAN LISTING

```
SEPARABLE KINETICS J.W. RIESE AND G. COLLIER
C
                                                                                      1
¥
      LIST
C
       DIMENSION Y(20), YD(20), PREC(10), PRECD(10), X5N(100), BETA(10), XLAM(
     110)
C
       EQUIVALENCE (Y,FLUX), (Y(2), PREC), (YD, FLUXD), (YD(2), PRECD)
C
      COMMON Y, YD, TLIFE, BETA, XLAM, MGRT, XMLTFD, XMLTMA, XMLTT, TF, TI, DT, T,
     INJ, BETA, XLAM, NGPJ, PP, E1, E2, NSPP, X5N, NSTOP
                                                                                     10
C
                                                                                     11
    1 READ INPUT TAPE 5,100,NJ
                                                                                     12
  100 FURMAT(2014)
                                                                                     13
      NGPJ = NJ + 1
                                                                                     14
      READ INPUT TAPE 5,101, TLIFE, FLUX, (BETA(J), XLAM(J,, J=1, NJ)
                                                                                     15
  101 FORMAT( 8E10.4)
                                                                                     16
      00 \ 11 \ I = 1.NJ
                                                                                     17
   11 PREC(I) = FLUX*9ETA(I)/(TLIFE*XLAM(I))
                                                                                     18
      CALL FDSTRT
                                                                                     19
                                                                                     20
      XMLTT = 1.0
                                                                                     21
      XMLTMA = 1.0 - XMLTFD
      READ INPUT TAPE 5.101.T
                                                                                     22
                                                                                     23
      TIM = 0.0
   12 READ INPUT TAPE 5,101,DT,PP,E1,E2,DTFD
                                                                                     24
      TF = T + DT
                                                                                     25
      READ INPUT TAPE 5,100, MORT, NSPP, NSTOP
                                                                                     26
       GO TO ( 14, 13 ), MORT
                                                                                     27
                                                                                     28
   13 \times MLTT1 = \times MLTT
       READ INPUT TAPE 5,101,XMLTT2
                                                                                     29
      GO TO 15
                                                                                     30
   14 XMLTM1 = XMLTMA
                                                                                     31
      READ INPUT TAPE 5,101,XMLTM2
                                                                                     32
      XMLTM2 = XMLTM1 + XMLTM2
                                                                                     33
   15 \text{ ISPP} = \text{NSPP} - 1
                                                                                     34
      TI = -10.0
                                                                                     35
      CALL FEEDBK
                                                                                     36
      TI = T
                                                                                     37
      TP = T
                                                                                     38
      IF(DT) 17, 17,16
                                                                                     39
   17 CALL FORRNT
                                                                                     40
      FULD= FLUX
                                                                                     41
       IF(NSTOP-2) 12, 1, 1
                                                                                     42
   16 CALL ICE ( PP, T, TP, E1, E2, NGOJ, Y, YD, X5N, KAT )
                                                                                     43
   38 GO TO (1000,2000,3000,4000) ,KAT
                                                                                     44
   39 \text{ KAT} = XICEF(0)
                                                                                     45
      GC TO 38
                                                                                     46
C
      BOX A
                                                                                     47
```

3

7

8

	TT = T	95
	T = T + TIM	96
	CALL FDPRNT	97
	FOLD= FLUX	98
	T = TT	99
	GO TO 3002	100
3003	TT= T+ TIM	101
	WRITE DUTPUT TAPE 6,3004,FLUX,TT	102
3004	FORMAT(7HOFLUX = 1PE14.7,5X2HAT 1PE14.7)	103
3002	TP = TP + PP	104
	GO TO 39	105
C	BOX D	106
4000	JTRY = JTRY +1	107
	WRITE OUTPUT TAPE 6,4001, JTRY	108
4001	FORMAT(31HONONCONVERGENCE IN ICE, TRY NO. 14)	109
	IF(JTRY-2)4004,4002,4002	110
4002	IF(NSTOP-2) 4003, 1, 1	111
4003	READ INPUT TAPE 5,101,DT	112
	READ INPUT TAPE 5,100, MORT, NSPP, NSTOP	113
	READ INPUT TAPE 5,101,DT	114
	TI = -10.0	115
	CALL FEEDBK	116
	GO TO 4002	117
4004	TIM = TIM + T	118
	TF = TF -T	119
	TP = TP - T	120
	TI = TI - T	121
	T = 0.0	122
	GO TO 16	123
	END	124

```
SUBROUTINE FOSTRT
                                                                                       2
      LIST
C
                                                                                       3
      DIMENSION Y(20), YD(20), PREC(10), PRECD(10), X5N(100), BETA(10), XLAM(
     110)
C
                                                                                       7
      EQUIVALENCE (Y, FLUX), (Y(2), PREC), (YD, FLUXD), (YD(2), PRECD)
C
                                                                                       9
      COMMON Y, YD, TLIFE, BETA, XLAM, MORT, XMLTFD, XMLTMA, XMLTT, TF, TI, DT, T,
     1NJ, BETA, XLAM, NGPJ, PP, E1, E2, NSPP, X5N, NSTOP
                                                                                      10
                                                                                      11
      DIMENSION EIGCH(20), POWFR(20), POWFC(20), QFR(20), QLR(20), QFC(20),
                                                                                      12
     1 QLC(20), HR(20), RHOR(20), THR(20), TWR(20), HTR(20), HC(20), RHOC(20),
                                                                                     13
     2 THC(20), TWC(20), HTC(20)
                                                                                      14
C
                                                                                      15
      COMMON NPOFL, NRE, NC, XNRE, XNC, RCAPR, RCAPC, XHTR, XHTC, XMART, DISCH,
                                                                                     16
     1 EIGRT, EIGCT, POWER, POWST, HIN, POUT, POFL, POFL, POWFA, GLTUT, PRESS,
                                                                                     17
     2 HP, FLOW, MACH, EIGCH, POWER, POWEC, QER, QLR, QEC, OLC, HR, RHOR, THR, TWR,
                                                                                      18
     3 HTR. HC. RHOC. THC. TWC. HTC. P2 . POWINT
                                                                                      19
                                                                                      20
      NGPJ = NJ + 1
      IF( NSTOP-2 )310,311,310
                                                                                      21
  310 CALL TABIH
                                                                                     22
  311 READ INPUT TAPE 5,100, NPOFL, NRE, NC, NST
                                                                                     23
                                                                                     24
  100 FORMAT(2014)
                                                                                     25
      READ INPUT TAPE 5,101,RCAPR,RCAPC,XHTR,XHTC,XMART,DISCH
  101 FORMAT(8E10.4)
                                                                                     26
      READ INPUT TAPE 5,101, (POWFR(I), I=1, NRE)
                                                                                     27
      READ INPUT TAPE 5,101, (POWFC(I), I=1, NC)
                                                                                     28
      READ INPUT TAPE 5.101.EIGRT.EIGCT.(EIGCH(I).I=1.NC)
                                                                                     29
                                                                                     30
      XNRE = NRE
      XNC = NC
                                                                                     31
                                                                                      32
      READ INPUT TAPE 5,101, POWST, HIN, POUT, POFL, PCWINT
      POWER = POWST
                                                                                     33
                                                                                      34
      QLTQT = 0.0
      POWFA = POWER/FLUX
                                                                                      35
      POFL2 = POFL
                                                                                      36
      DO 1 I = 1.NRE
                                                                                     37
      QFR(I) = POWER*POWFR(I)
                                                                                     38
                                                                                      39
      QLR(I) = QFR(I)
    1 \text{ QLTOT} = \text{QLTOT} + \text{QLR}(I)
                                                                                     40
      DO 2 I = 1.NC
                                                                                     41
      QFC(I) = POWER*POWFC(I)
                                                                                      42
      QLC(I) = QFC(I)
                                                                                     43
                                                                                     44
    2 QLTOT = QLTOT + QLC(I)
                                                                                     45
      GO TO ( 300,301 ),NST
  301 READ INPUT TAPE 5,101,T,HP
                                                                                     46
      TF = T + 1.0
                                                                                      47
```

C

```
TI = 0.0
                                                                                     48
    DT = 1.0
                                                                                     49
    READ INPUT TAPE 5.101, (TWR(I), I = 1.00RE)
                                                                                     50
    READ INPUT TAPE 5.101° (TWC(I).I = 1°NC)
                                                                                    51
                                                                                    52
    READ INPUT TAPE 5,101,(Y(I),I=2,NGPJ)
    CALL FEEDBK
                                                                                    53
    GO TO 67
                                                                                    54
300 GO TO ( 3, 6), NPOFL
                                                                                    55
  3 PRESS = POFL
                                                                                    56
    HP = HIN + 200.0
                                                                                    57
  4 P2 = POUT
                                                                                    58
    CALL NOZPG(HP, PRESS, P2, DISCH, FLOW, MACH)
                                                                                    59
    HNEW = HIN + QLTOT/FLOW
                                                                                    60
    IF(ABSF((HNEW-HP)/HP) -0.005)7.7.5
                                                                                    61
  5 HP = HNEW
                                                                                    62
    GO TO 4
                                                                                    63
  6 \text{ FLOW} = \text{POFL}
                                                                                    64
    HP = HIN + QLTOT/FLOW
                                                                                    65
    P2 = POUT
                                                                                    66
    PRESS = P2 + 1.06E + 03 * FLOW/SQRTF(HP)
                                                                                    67
    CALL NOZWG(HP, FLOW, P2, DISCH, PRESS, MACH)
                                                                                    68
  7 \text{ HTEMP} = \text{HIN}
                                                                                    69
    DO 8 I = 1, NRE
                                                                                    70
    HR(I) = HTEMP + QLR(I)/FLOW
                                                                                    71
    HNOW = (HTEMP + HR(I))/2.0
                                                                                    72
                                                                                    73
    HTEMP = HR(I)
    CALL TAB2(HNOW, PRESS, QUAL, TG, ZG, GAMG, CPG, CG, SG, CONG, VISG, TL, ZL,
                                                                                    74
                                                                                    75
   1GAML, CPL, CL, SL, CONL, VISL)
    YMART = 0.0
                                                                                    76
    IF(QUAL*(1.0-QUAL))10,10,9
                                                                                    77
  9 YMART = XMART*VISL*((1.0-QUAL)*VISG/(QUAL*VISL))**0.9*SQRTF(ZL/ZG)
                                                                                    78
   1/VISG
                                                                                    79
    IF(YMART - 9.0)64,64,63
                                                                                    80
63 \text{ YMART} = 9.0
                                                                                    81
 64 ZG = QUAL*ZG + ZL*(1.0-QUAL)
                                                                                    82
 10 RHOR(I) = 0.1878261*PRESS/(ZG*TG)
                                                                                    83
    THR(I) = TG
                                                                                    84
    HTR(I) = XHTR*CONG*FLOW**O.8*(CPG/(VISG*CONG))**O.4/(1.0+YMART)
                                                                                    85
  8 \text{ TWR}(I) = \text{THR}(I) + \text{QLR}(I)/\text{HTR}(I)
                                                                                    86
    DO 11 I = 1.NC
                                                                                    87
    HC(I) = HTEMP + QLC(I)/FLOW
                                                                                    88
    HNOW = (HTEMP + HC(I))*0.5
                                                                                    89
    HTEMP = HC(I)
                                                                                    90
    CALL TAB2(HNOW, PRESS, QUAL, TG, ZG, GAMG, CPG, CG, SG, CONG, VISG, TL, ZL,
                                                                                    91
   1GAML, CPL, CL, SL, CONL, VISL)
                                                                                    92
    YMART = 0.0
                                                                                    93
    IF(QUAL*(1.0-QUAL))13,13,12
```

```
12 YMART = XMART*VISL*((1.0-QUAL)*VISG/(QUAL*VISL))**0.9*SQRTF(ZL/ZG)
                                                                                  95
                                                                                  96
   1/VISG
    IF( YMART - 9.0 )66,66,65
                                                                                  97
 65 \text{ YMART} = 9.0
                                                                                  98
 66 ZG = QUAL*ZG + ZL*(1.0-QUAL)
                                                                                  99
 13 \text{ RHOC}(I) = 0.1878261*PRESS/(ZG*TG)
                                                                                 100
                                                                                 101
    THC(I) = TG
                                                                                 102
    HTC(I) = XHTC*CONG*FLOW**0.8*(CPG/(VISG*CONG))**0.4/(1.0+YMART)
 11 TWC(I) = THC(I) + QLC(I)/HTC(I)
                                                                                 103
    SUM1 = 0.0
                                                                                 104
                                                                                 105
    DO 61 I = 1.NRE
                                                                                 106
 61 \text{ SUM1} = \text{SUM1} + \text{TWR}(I)
    SUM2 = 0.0
                                                                                 107
    XMLTFD = 0.0
                                                                                 108
    DO 62 I = 1.NC
                                                                                 109
                                                                                 110
    SUM2 = SUM2 + TWC(I)
 62 XMLTFD = XMLTFD + EIGCH(I)*RHOC(I)
                                                                                 111
    XMLTFD = XMLTFD + EIGRT*(SUM1/XNRE -528.) + EIGCT*(SUM2/XNC-528.)
                                                                                112
67 WRITE OUTPUT TAPE 6,200
                                                                                113
                                                                                114
200 FORMAT(1H1)
    CALL FDPRNT
                                                                                115
                                                                                116
    RETURN
    END
                                                                                117
```

```
SUBROUTINE FEEDBK
      LIST
                                                                                      3
C
      DIMENSION Y(20).YD(20).PREC(10).PRECD(10).X5N(100).BETA(10).XLAM(
     110)
C
                                                                                      7
      EQUIVALENCE (Y,FLUX), (Y(2), PREC), (YD,FLUXD), (YD(2), PRECD)
C
                                                                                      9
      COMMON Y.YD.TLIFE.BETA.XLAM.MORT.XMLTFD.XMLTMA.XMLTT.TF.TI.DT.T.
     1NJ, BETA, XLAM, NGPJ, PP, E1, E2, NSPP, X5N, NSTOP
                                                                                     10
                                                                                     11
C
                                                                                     12
      DIMENSION EIGCH(20), POWFR(20), POWFC(20), QFR(20), QLR(20), QFC(20),
     1 QLC(20), HR(20), RHOR(20), THR(20), TWR(20), HTR(20), HC(20), RHOC(20),
                                                                                     13
     2 THC(20), TWC(20), HTC(20)
                                                                                     14
                                                                                     15
C
      COMMON NPOFL, NRE, NC, XNRE, XNC, RCAPR, RCAPC, XHTR, XHTC, XMART, DISCH,
                                                                                     16
                                                                                     17
     1 EIGRT, EIGCT, POWER, POWST, HIN, POUT, POFL, POFL, POWFA, QLTOT, PRESS,
     2 HP.FLOW.MACH.EIGCH.POWER.POWEC.QFR.QLR.QFC.QLC.HR.RHOR.THR.TWR.
                                                                                     18
     3 HTR. HC. RHOC. THC. TWC. HTC. P2. POWINT
                                                                                     19
                                                                                     20
      IF(TI+9.0)1,1,2
                                                                                     21
    1 \text{ POFL} = \text{POFL2}
                                                                                     22
      READ INPUT TAPE 5,101, POFL2
  101 FORMAT(8E10.4)
                                                                                     23
                                                                                     24
      RETURN
                                                                                     25
    2 \text{ TC1} = (\text{TF-T})/\text{DT}
                                                                                     26
      TC2 = 1.0 - TC1
                                                                                     27
      POWER = (POWST + POWFA*FLUX)*0.5
      POWST = POWFA*FLUX
                                                                                     28
                                                                                     29
      POWINT = POWINT+TI*POWER
                                                                                     30
      DO 3 I = 1.NRE
                                                                                     31
      P3 = TWR(I)
                                                                                     32
      IF(P3-160.0)82,82,83
                                                                                     33
   82 P3 = 0.13714*P3/160.0
                                                                                     34
      GO TO 81
                                                                                     35
   83 IF(P3-300.0)84,84,85
                                                                                     36
   84 P3 = 0.13714 + (P3-160.0)*0.31151/140.0
                                                                                     37
      GO TO 81
                                                                                     38
   85 IF(P3-600.0)86,86,87
   86 P3 = 1.0 - (600.0-P3)*0.55135/300.0
                                                                                     39
                                                                                     40
      GO TO 81
                                                                                     41
   87 P3 = 1.0 + (P3-600.0)*0.1549/200.0
   81 QFR(I) = POWER*POWFR(I)
                                                                                     42
                                                                                     43
      TWR(I) = TWR(I) + RCAPR*TI*(QFR(I)-QLR(I))/P3
    3 QFR(I) = POWST*POWFR(I)
                                                                                     44
                                                                                     45
      DO 4 I = 1.NC
                                                                                     46
      P3 = TWC(I)
      IF(P3-1120.0)92,92,93
                                                                                     47
```

```
92 P3 = 0.73*P3/1120.0
                                                                                    48
   GO TO 91
                                                                                    49
93 IF(P3-2600.0)94,94,95
                                                                                    50
                                                                                    51
94 P3 = 0.73 + 0.22*(P3-1120.0)/1480.0
                                                                                    52
   GO TO 91
95 P3 = 1.0 + (P3-4000.0)/28000.0
                                                                                    53
                                                                                    54
91 QFC(I) = POWER*POWFC(I)
   TWC(I) = TWC(I) + RCAPC*TI*(QFC(I)-QLC(I))/P3
                                                                                    55
                                                                                    56
 4 QFC(I) = POWST*POWFC(I)
   P2 = POUT
                                                                                    57
   MA = 0
                                                                                    58
   GO TO (5,6), NPOFL
                                                                                    59
                                                                                    60
 5 PRESS = TC1*POFL + TC2*POFL2
   CALL NOZPG(HP, PRESS, P2, DISCH, FLOW, MACH)
                                                                                    61
   GO TO 7
                                                                                    62
 6 FLOW = TC1*POFL + TC2*POFL2
                                                                                    63
                                                                                    64
   CALL NOZWG(HP, FLOW, P2, DISCH, PRESS, MACH)
 7 \text{ HTEMP} = HIN
                                                                                    65
   MA = MA + 1
                                                                                    66
   DO 8 I = 1.NRE
                                                                                    67
                                                                                    68
   HNOW = HTEMP
                                                                                    69
   IIT = 0
13 CALL TAB2(HNOW, PRESS, QUAL, TG, ZG, GAMG, CPG, CG, SG, CONG, VISG, TL, ZL,
                                                                                    70
  1GAML, CPL, CL, SL, CONL, VISL)
                                                                                    71
   YMART = 0.0
                                                                                    72
                                                                                    73
   IF(QUAL*(1.0-QUAL))10,10,9
 9 YMART = XMART*VISL*((1.0-QUAL)*VISG/(QUAL*VISL))**0.9*SQRTF(ZL/ZG)
                                                                                   74
                                                                                   75
  1/VISG
   IF( YMART - 9.0 )64,64,63
                                                                                   76
                                                                                   77
63 \text{ YMART} = 9.0
                                                                                   78
64 ZG = QUAL*ZG + ZL*(1.0-QUAL)
                                                                                   79
10 HTR(I) = XHTR*CONG*FLOW**0.8*(CPG/(VISG*CONG))**0.4/(1.0+YMART)
   HNOWI = HTEMP + HTR(I)*(TWR(I)-TG)*0.5/FLOW
                                                                                   80
   IF( ABSF( (HNOW1+155.0)/(HNOW+155.0)-1.)-.01 )15,11,11
                                                                                   81
                                                                                    82
11 \text{ IIT} = \text{IIT} + 1
   IF(IIT-25)69,70,15
                                                                                    83
                                                                                   84
70 \text{ HNOW} = 0.5*(\text{HNOW+HNOW1})
   GO TO 13
                                                                                   85
69 \text{ HNOW} = \text{HNOW1}
                                                                                   86
   GO TO 13
                                                                                   87
15 \text{ HR}(I) = 2.0*\text{HNOWl-} \text{ HTEMP}
                                                                                   88
                                                                                   89
   THR(I) = TG
   RHOR(I) = 0.1878261*PRESS/(ZG*TG)
                                                                                   90
                                                                                   91
   QLR(I) = HTR(I)*(TWR(I)-TG)
                                                                                   92
8 \text{ HTEMP} = \text{HR}(I)
   DO 16 I = 1.NC
                                                                                   93
                                                                                    94
   HNOW = HTEMP
```

	IIT = 0	95
20	CALL TAB2(HNOW, PRESS, QUAL, TG, ZG, GAMG, CPG, CG, SG, CONG, VISG, TL, ZL,	96
	IGAML, CPL, CL, SL, CUNL, VISL)	97
	YMART = 0.0	98
	IF(QUAL*(1.0-QUAL))22,22,21	99
21	YMART = XMART*VISL*((1.0-QUAL)*VISG/(QUAL*VISL))**0.9*SQRTF(ZL/ZG)	100
	1/VISG	101
	IF(YMART - 9.0)66,66,65	102
65	YMART = 9.0	103
66	ZG = QUAL*ZG + ZL*(1.0-QUAL)	104
22	HTC(I) = XHTC*CCNG*FLOW**0.8*(CPG/(VISG*CONG))**0.4/(1.0+YMART)	105
	HNOW1 = HTEMP + HTC(I)*(TWC(I)-TG)*0.5/FLOW	106
	IF(ABSF((HNOW1+155.0)/(HNOW+155.0)-1.)01)27,23,23	107
23	IIT = IIT + 1	108
	IF(IIT-25)79,80,27	109
80	HNOW = 0.5*(HNOW+HNOW1)	110
	GO TO 20	111
79	HNOW = HNOW1	112
	GO TO 20	113
27	HC(I) = 2.0*HNOW1- HTEMP	114
	THC(I) = TG	115
	RHOC(I) = 0.1878261*PRESS/(ZG*TG)	116
	QLC(I) = HTC(I)*(TWC(I)-TG)	117
16	HTEMP = HC(I)	118
	HP = HTEMP	119
	P2 = POUT	120
	GO TO (17,18), NPOFL	121
17	FLOW1 = FLOW	122
	CALL NOZPG(HTEMP, PRESS, P2, DISCH, FLOW, MACH)	123
	IF(MA-2)52,53,53	124
53	FLOW = 0.5*(FLOW+FLOW1)	125
	IF(MA-5)52,19,19	126
	IF(ABSF(FLOW/FLOW1 - 1.0) - 0.05) 19,19, 7	127
18	PRESS1 = PRESS	128
	CALL NOZWG(HP,FLOW,P2,DISCH,PRESS,MACH) IF(MA-2)50,51,51	129 130
E 1	PRESS = 0.5*(PRESS+PRESS1)	131
21	IF(MA-5)50,19,19	132
50	IF(ABSF(PRESS/PRESS1 -1.0) - 0.05) 19,19,7	133
	CONTINUE	134
17	WRITE OUTPUT TAPE 6,102,MA	135
102	FORMAT(21H CHANNEL ITERATIONS = 18)	136
102	SUM1 = 0.0	137
	DO 61 I = 1, NRE	138
61	SUM1 = SUM1 + TWR(I)	139
~ .	SUM2 = 0.0	140
	XMLTFD = 0.0	141

Pg. 10

```
SUBROUTINE FORRNT
      LIST
C
      DIMENSION Y(20), YD(20), PREC(10), PRECD(10), X5N(100), BETA(10), XLAM(
     110)
C
                                                                                  7
      EQUIVALENCE (Y,FLUX), (Y(2), PREC), (YD,FLUXD), (YD(2), PRECD)
C
                                                                                  9
      COMMON Y, YD, TLIFE, BETA, XLAM, MORT, XMLTFD, XMLTMA, XMLTT, TF, TI, DT, T,
     1NJ, BETA, XLAM, NGPJ, PP, E1, E2, NSPP, X5N, NSTOP
                                                                                 10
C
                                                                                 11
      DIMENSION EIGCH(20), POWFR(20), POWFC(20), QFR(20), QLR(20), QFC(20),
                                                                                 12
     1 QLC(20),HR(20),RHOR(20),THR(20),TWR(20),HTR(20),HC(20),RHOC(20),
                                                                                 13
     2 THC(20), TWC(20), HTC(20)
                                                                                 14
C
                                                                                 15
      COMMON NPOFL, NRE, NC, XNRE, XNC, RCAPR, RCAPC, XHTR, XHTC, XMART, DISCH,
                                                                                 16
     1 EIGRT, EIGCT, POWER, POWST, HIN, POUT, POFL, POFL2, POWFA, QLTOT, PRESS,
                                                                                 17
     2 HP, FLOW, MACH, EIGCH, POWER, POWEC, OFR, OLR, OFC, OLC, HR, RHOR, THR, TWR,
                                                                                 18
     3 HTR, HC, RHOC, THC, TWC, HTC, P2 , POWINT
                                                                                 19
      WRITE OUTPUT TAPE 6,200,T
                                                                                 20
  200 FORMAT( //// THOTIME = 1PE14.7 /// )
                                                                                 21
      WRITE OUTPUT TAPE 6,201,FLUX,XMLTT,XMLTFD,XMLTMA
                                                                                 22
 201 FORMAT(6X6HFLUX = 1PE14.7, 10X8HKTOTAL = 1PE14.7, 8X11HKFEEDBACK =
                                                                                 23
     1 1PE14.7. 5X6HKMAN = 1PE14.7 // )
                                                                                 24
      WRITE OUTPUT TAPE 6,202, (PREC(J), J=1,NJ)
                                                                                 25
  202 FORMAT(6X21HPRECURSOR DENSITIES = 1PE22.7, 1P2E30.7 /
                                                                                 26
     1 ( 19X1P3E30.7 ) )
                                                                                 27
      WRITE DUTPUT TAPE 6,203, HIN, POWST, P2
                                                                                 28
  203 FORMAT( 19H0 ENTHALPY IN = 1PE14.7, 3X7HPOWER = 1PE14.7
                                                                                 29
     1 • 9X16HNOZ THROAT PSI = 1PE14.7 // )
                                                                                 30
      WRITE OUTPUT TAPE 6,204, FLOW, PRESS, POUT, POWINT
                                                                                 31
 204 FORMAT( 6X11HFLOW RATE = 1PE14.7, 5X10HPRESSURE = 1PE14.7,
                                                                                 32
     1 6X11HAMB PRESS = 1PE14.7 , 5X8HPOWINT = 1PE14.7 // )
                                                                                 33
      WRITE OUTPUT TAPE 6,205
                                                                                 34
 205 FORMAT( 6X20HREFLECTOR STATIONS -
                                               11 )
                                                                                 35
      WRITE OUTPUT TAPE 6,206
                                                                                 36
 206 FORMAT( 11X5HPOWER , 19X9HWALL TEMP , 15X12HHEAT REMOVAL ,
                                                                                 37
     1 12X10HFLUID TEMP , 14X13HFLUID DENSITY // )
                                                                                 38
      DO 1 I = 1.NRE
                                                                                 39
    1 WRITE OUTPUT TAPE 6,207,QFR(I),TWR(I),QLR(I),THR(I),RHOR(I)
                                                                                 40
 207 FORMAT( 1P5E24.7 )
                                                                                 41
      WRITE OUTPUT TAPE 6,208
                                                                                 42
 208 FORMAT( 21HO CORE STATIONS - // )
                                                                                 43
      WRITE DUTPUT TAPE 6,206
                                                                                 44
      DO 2 I = 1.NC
                                                                                 45
    2 WRITE DUTPUT TAPE 6,207,QFC(I),TWC(I),QLC(I),THC(I),RHOC(I)
                                                                                 46
      PRINT209,T
                                                                                 47
```

209 FORMAT(4H T= 1PE14.7)
RETURN
END

48 49 50 7g.12

```
SUBROUTINE NOZPG( H1,P1,P2,DISCH,W,MACH )
   LIST
                                                                               3
   MACH = 0
   IF(P2-P1)12,13,13
                                                                               5
13 W = 0.0
   P2 = P1
   RETURN
12 CALL TAB2(H1,P1,X,TG,ZG,GAMG,CPG,CG,SG,CONG,VISG,TL,ZL,GAML,CPL,CL
                                                                               9
  1,SL,CONL,VISL)
                                                                              10
   IF( X*(1.0-X))1,1,2
                                                                              11
 1 S = SG
                                                                              12
   G = GAMG
   Z = ZG
                                                                              13
                                                                              14
   GD TO 3
                                                                              15
 2 S = X*SG + (1.0-X)*SL
   G = X*GAMG + (1.0-X)*GAML
                                                                              16
   Z = X*ZG + (1.0-X)*ZL
                                                                              17
 3 H2 = H1 - 0.9854707*Z*TG*G*(1.0-(P2/P1)**(1.0-1.0/G))/(G-1.0)
                                                                              18
                                                                              19
   INDX = 1
                                                                              20
   IIT = 0
 6 CALL TAB2(H2,P2,X,TG,ZG,GAMG,CPG,CG,SG,CONG,VISG,TL,ZL,GAML,CPL,CL
                                                                              21
                                                                              22
  1,SL,CONL,VISL)
   IF(X*(1.0-X))5,5,4
                                                                              23
 4 SG = X*SG + (1.0-X)*SL
                                                                              24
                                                                              25
 5 DH = TG*(S-SG)
   H2 = H2 + DH
                                                                              26
   IF( ABSF( DH/H2 ) - 0.001 ) 7.30.30
                                                                              27
                                                                              28
30 IIT = IIT +1
                                                                              29
   IF(IIT-20)6,31,32
                                                                              30
31 H2 = H2-0.5*DH
                                                                              31
   GO TO 6
                                                                              32
32 H2 = H2-DH
 7 IF( X*(1.0-X)) 9,9,8
                                                                              33
                                                                              34
 8 ZG = X*ZG + (1.0 - X)*ZL
                                                                              35
 9 \text{ RHO} = 0.1878261*P2/(ZG*TG)
                                                                              36
   W = DISCH*RHO*SQRTF(H1-H2)
                               ) , INDX
                                                                              37
   GD TO (10,11,17,18
10 P3 = P2
                                                                              38
                                                                              39
   W3 = W
                                                                              40
   H3 = H2
                                                                              41
   P2 = P2 + 0.05*(P1-P2)
                                                                              42
   INDX = 2
   IIT = 0
                                                                              43
                                                                              44
   GO TO 6
11 IF(W -W3)14,14,15
                                                                              45
14 P2 = P3
                                                                              46
   W = W3
                                                                              47
```

	SUBROUTINE NOZWG(H1, W, P2, DISCH, P1, MACH)	1
	LIST	2
	PS = P2	3
	MBO = MB	4
	MB = 0	5
9	CALL NOZPG(H1,P1,P2,DISCH,WS,MACH)	
	MB = MB + 1	6 7 8
11	P2 = PS	8
	IF(WS)15,15,14	9
15	P1 = 1.01*(P2+1.0)	10
	GO TO 9	11
14	P3 = P1*W/WS	12
	P4 = P2 + (P1-P2)*(W/WS)**2	13
	IF (MB- 10) 18,22,22	14
22	P3= 0.5*(P1+ P3)	15
	P4= 0.5*(P1+ P4)	16
18	IF((W-WS)*(P3-P4))17,17,16	17
16	P1 = P4	18
	GO TO 19	19
17	P1 = P3	20
19	IF(ABSF(WS/W -1.0) -0.015) $20,9,9$	21
20	CALL NOZPG(H1,P1,P2,DISCH,WS,MACH)	22
	IF(ABSF(WS/W -1.0) - 0.015) 10,10,21	23
10	RETURN	24
21	MB = MB + 1	25
	P2 = PS	26
	GO TO 14	27
	END	28