ZERO-G THERMODYNAMIC VENTING SYSTEM (TVS)
PERFORMANCE PREDICTION PROGRAM

PREPARED BY

HAN NGUYEN
HAN NGUYEN
MPS ANALYSIS UNIT
CRYOGENIC SYSTEMS GROUP
VEHICLE AND SYSTEMS DESIGN

APPROVED BY

TIBOR LAK
TECHNICAL MANAGER

CHARLES C. WOOD
PROGRAM MANAGER

No. of Pages: 138

Date: May 24, 1994
ZERO-G THERMODYNAMIC VENTING SYSTEM (TVS)
PERFORMANCE PREDICTION PROGRAM

PREPARED BY

HAN NGUYEN
HAN NGUYEN
MPS ANALYSIS UNIT
CRYOGENIC SYSTEMS GROUP
VEHICLE AND SYSTEMS DESIGN

APPROVED BY

TIBOR LAK
TECHNICAL MANAGER

CHARLES C. WOOD
PROGRAM MANAGER

No. of Pages: 138
Date: May 24, 1994
This report documents the Zero-g Thermodynamic Venting System (TVS) performance prediction computer program. The zero-g TVS is a device that destratifies and rejects environmentally induced zero-g thermal gradients in the LH2 storage transfer system. A recirculation pump and spray injection manifold recirculates liquid throughout the length of the tank, thereby destratifying both the ullage gas and liquid bulk. Heat rejection is accomplished by the opening of the TVS control valve which allows a small flow rate to expand to a low pressure thereby producing a low temperature heat sink which is used to absorb heat from the recirculating liquid flow. The program was written in FORTRAN 77 language on the HP-9000 and IBM PC computers. It can be run on various platforms with a FORTRAN compiler.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title Page</td>
<td>i</td>
</tr>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>iii</td>
</tr>
<tr>
<td>List of Illustrations</td>
<td>vi</td>
</tr>
</tbody>
</table>

## 1 INTRODUCTION

1.1 Purpose  
1.2 Problem Description  
1.3 Areas of Applications  
1.4 Description of the Physical Model

## 2 ANALYTICAL MODEL DESCRIPTION

2.1 Heat Exchanger Model

2.1.1 Fluid Quality at Heat Exchanger Outlet  
2.1.2 Two-Phase Pressure Loss in the Heat Exchanger  
2.1.3 Forced Convection Heat-Transfer Model  
2.1.4 Spray Temperature

2.2 Spray Manifold and Injection Tube Model

2.2.1 Spray Manifold  
2.2.2 Spray Injection Tube  
2.2.3 Spray Manifold and Injection Tube Model Algorithm

2.3 Recirculation Pump Model

2.4 Tank Thermal Model

2.4.1 Ullage  
2.4.2 Tank Wall  
2.4.3 Wall Liquid  
2.4.4 Bulk Liquid  
2.4.5 Heat Transfer

2.4.5.1 Free Convection  
2.4.5.2 Forced Convection
2.4.6 Mass Transfer

2.4.6.1 Bulk Liquid Boiling
2.4.6.2 Wall Liquid Boiling
2.4.6.3 Liquid Droplet Evaporation in the Ullage
2.4.6.4 Liquid Spray Falling into Bulk Liquid or Accumulating on Tank Wall
2.4.6.5 Ullage Condensation
2.4.6.6 Liquid Surface Condensation

2.5 References

2.6 Sample Cases

3.0 COMPUTER MODEL DESCRIPTION

3.1 Programming Description

3.2 Flow Charts

3.2.1 Heat Exchanger Model
3.2.2 Spray Manifold/Injection Tube Model
3.2.3 Integrated Zero-g TVS Model

3.3 Description of Variables

3.3.1 Input Variables
3.3.1.1 Heat Exchanger Model
3.3.1.2 Spray Manifold/Injection Tube Model
3.3.1.3 Recirculation Pump Model
3.3.1.4 Integrated Zero-g TVS Model
3.3.1.5 LH₂ Saturated Properties
3.3.1.6 GH₂ Properties

3.3.2 Output Variables
3.3.2.1 Heat Exchanger Model
3.3.2.2 Spray Manifold/Injection Tube Model
3.3.2.3 Integrated Zero-g TVS Model

3.4 Program Listing

3.4.1 Heat Exchanger Model
3.5 Input Data

3.5.1 Heat Exchanger Model
3.5.2 Integrated Zero-g TVS Model
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Zero-g Thermodynamic Venting System (TVS) Concept</td>
<td>I-2</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Spray Manifold Model</td>
<td>II-8</td>
</tr>
<tr>
<td>2.2.2</td>
<td>Spray Injection Tube Model</td>
<td>II-11</td>
</tr>
<tr>
<td>2.3.1</td>
<td>LH$_2$ Recirculation Pump Head-Flow Curve</td>
<td>II-14</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Tank Thermal Model</td>
<td>II-16</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Ullage Thermal Model</td>
<td>II-17</td>
</tr>
<tr>
<td>2.4.3</td>
<td>Tank Wall Thermal Model</td>
<td>II-20</td>
</tr>
<tr>
<td>2.4.4</td>
<td>Wall Liquid Thermal Model</td>
<td>II-20</td>
</tr>
<tr>
<td>2.4.5</td>
<td>Bulk Liquid Thermal Model</td>
<td>II-22</td>
</tr>
<tr>
<td>2.4.6</td>
<td>Free Convection Heat-Transfer Correlation for Interior Surfaces of</td>
<td>II-26</td>
</tr>
<tr>
<td></td>
<td>Vertical Ducts, Vertical Plates and Cylinders, and Horizontal Cylinders</td>
<td></td>
</tr>
<tr>
<td>2.4.7</td>
<td>Forced Convection Heat-Transfer Correlation for Flow over a Sphere</td>
<td>II-27</td>
</tr>
<tr>
<td>2.4.8</td>
<td>Droplet Evaporation Model</td>
<td>II-31</td>
</tr>
<tr>
<td>2.4.9</td>
<td>Ullage Condensation Model</td>
<td>II-31</td>
</tr>
<tr>
<td>2.6.1</td>
<td>TVS Performance Simulation at 90% Liquid Quantity</td>
<td>II-34</td>
</tr>
<tr>
<td>2.6.2</td>
<td>TVS Recirculation Pump and Vent Valve Flow Rate Transient During</td>
<td>II-34</td>
</tr>
<tr>
<td></td>
<td>Ullage Destratification (90% Liquid Quantity)</td>
<td></td>
</tr>
<tr>
<td>2.6.3</td>
<td>TVS Performance Simulation at 25% Liquid Quantity</td>
<td>II-35</td>
</tr>
<tr>
<td>2.6.4</td>
<td>TVS Performance Simulation at 25% Liquid Quantity</td>
<td>II-35</td>
</tr>
<tr>
<td>2.6.5</td>
<td>TVS Recirculation Pump and Vent Valve Flow Rate Transient During</td>
<td>II-36</td>
</tr>
<tr>
<td></td>
<td>Ullage Destratification (25% Liquid Quantity)</td>
<td></td>
</tr>
<tr>
<td>2.6.6</td>
<td>TVS Operation Frequency (Percent) as a Function of Liquid Quantity</td>
<td>II-36</td>
</tr>
</tbody>
</table>
SECTION 1
INTRODUCTION

1.1 Purpose
The purpose of the zero-g thermodynamic venting system (TVS) model is to define the pressure level requirements, propellant loss due to venting, total pump power consumption, venting system operation duration and frequency as a function of liquid level and acceleration environment.

1.2 Problem Description
Long-term storage of subcritical cryogens in space is subjected to thermal stratification which is more severe than experienced in a 1-g environment due to the absence of gravity-induced body forces. If left uncontrolled, the thermal gradients result in excessive tank pressure rise and formation of liquid/vapor mixtures within the liquid bulk, liquid acquisition device, and propellant transfer lines. A subsystem is therefore needed to reduce the thermal gradient to acceptable levels, and reject the environmental heat leakage in an efficient manner.

1.3 Areas of Application
This program is designed to predict the tank ullage pressure and temperature, pump flow rate and power consumption, propellant loss due to venting, venting duration and frequency for different liquid levels, accelerations and heat leak rates. It can also be used to model tank chill down and liquid fill in zero-g and one-g environments. The program therefore has general applicability in areas of long-term cryogenic fluid storage and transfer in space and on the ground.

1.4 Description of the Physical Model
The zero-g TVS concept, shown in Figure 1.1, has been defined to operate in a self-induced, forced convection environment. It includes a recirculation pump, located external to the tank, to flow liquid from the tank, a spray manifold and injection tube to mix and destratify the ullage and liquid bulk, a heat exchanger to absorb heat from the tank, and an overboard vent line to reject the heat. The concept operates in the following manner. When the vent pressure level is reached, the recirculation pump is activated, resulting in liquid flow through the spray injection manifold which destratifies the liquid bulk and ullage gas through forced convection. When the fluid bulk temperature reaches a predetermined level, the vent valve is opened, resulting in a temperature drop of the vent flow by isenthalpic expansion. This liquid is used to absorb heat from the recirculation flow via the heat exchanger, and subsequently vented overboard as a gas.
Figure 1.1 Zero-g Thermodynamic Venting System (TVS) Concept
SECTION 2
ANALYTICAL MODEL DESCRIPTION

The zero-g TVS model consists of the thermal-fluid models of the heat exchanger, spray manifold and injection tubes, recirculation pump, and tank. These models were developed and verified independently before they were integrated into the transient TVS model. Following is a description of each model.

2.1 Heat Exchanger Model

The heat exchanger model is based on the generalized two-phase cryogenic propellant dump model developed to evaluate the Space Shuttle Main Propulsion System (MPS) cryogenic propellant dump/vacuum inerting operations performance (Ref. 29). It is a multi-node finite difference model that simulates two-phase flow in a quasi steady-state mode.

The model uses the fluid properties at the inlet of the spray manifold as the input to the first node. With one of the inlet fluid property, namely the total enthalpy of the fluid at the inlet, the total enthalpy at the exit can be calculated based on the First Law of Thermodynamics. The total enthalpy at the exit, and an assumed mass flow rate are used to determine the exit static pressure. The exit static pressure is determined by an iterative process: with the flow assumed choked at the exit, the exit static pressure is increased incrementally until the maximum entropy is achieved (sonic flow), or when it becomes greater than the back pressure (subsonic flow). From the calculated exit pressure, the other exit fluid properties of the last node, the total pressure loss between the inlet and outlet can then be calculated and the inlet fluid properties can be determined.

The following sections will provide the equations used in the heat exchanger model.

2.1.1 Fluid Quality At Heat Exchanger Outlet

The outlet static pressure is calculated, assuming choked or sonic flow, where the entropy point is maximum. The following equations are solved simultaneously for the liquid quality of the fluid at the outlet

\[ h_o = h_i + \frac{Q}{m} + \Delta h_a \]  
\[ V_o = \frac{m}{\rho_o A} \]  
\[ \rho_o = \frac{1}{(\rho_L)_0 + Y_o \left[ \frac{1}{(\rho_v)_0} - \frac{1}{(\rho_L)_0} \right]} \]  
\[ h_o = (1 - y_o)(h_L)_o + Y_o(h_v)_o + \frac{V_o^2}{2g_e} \]

where the following fluid properties are based on the outlet static pressure
\[(h_L)_o = \text{the outlet liquid enthalpy}\]
\[(h_v)_o = \text{the outlet vapor enthalpy}\]
\[(\rho_L)_o = \text{the outlet liquid density}\]
\[(\rho_v)_o = \text{the outlet vapor density}\]
\[Q = \text{total heat-transfer rate to a specific node}\]
\[\Delta H = \text{change in height of the line between inlet and outlet}\]
\[a = \frac{g_o}{g_o} = \text{acceleration}\]
\[h_o = \text{total enthalpy at the outlet}\]
\[\rho_o = \text{total density at the outlet}\]
\[V_o = \text{fluid velocity at the outlet}\]
\[Y_o = \text{fluid quality at the outlet}\]

With the outlet quality, the total entropy then can be calculated using the following equation

\[
\{S_o \} = (1 - y_o) (S_L)_o + Y_o (S_v)_o \]

where \( (S_L)_o = \text{the outlet liquid entropy} \)
\( (S_v)_o = \text{the outlet vapor entropy} \)

Iteration of the above outlet equations can be performed to obtain the maximum entropy point and the outlet static pressure.

2.1.2 Two-Phase Pressure Loss in the Heat Exchanger

To calculate the two-phase pressure loss (momentum and friction) between the inlet and outlet of the heat exchanger, the Lockhart-Martinelli correlation is used. The outlet pressure is

\[P_o = (P_s)_o + (P_d)_o \]

The total pressure loss term is further defined as

\[\Delta P_T = \Delta P_m + \Delta P_f \]

where \( \Delta P_m = \text{pressure loss due to momentum change} \)
\( \Delta P_f = \text{pressure loss due to frictional forces} \)

The momentum pressure loss is defined as

\[\Delta P_m = \frac{m}{g_o A} (V_o - V_1) \]

The frictional pressure loss is defined as
\[ \Delta P_t = \frac{144K}{2\rho_L g_e} \left[ \frac{\dot{m}(1 - \tilde{Y})}{A} \right]^2 \]  

(2.1.9)

where \( K = \left( \frac{f L}{D} \right) \) is the line loss coefficient

- \( \rho_L \) = the average liquid density between the inlet and outlet
- \( \tilde{Y} \) = the average total liquid quality between the inlet and outlet
- \( \Phi_L = f(X) \) is the Lockhart-Martinelli correlation factor

The Lockhart-Martinelli correlation is approximately defined as

\[ \Phi_L^2 = 1 + \frac{1}{X} + \frac{1}{X^2} \]  

(2.1.10)

\( X \) is defined as

\[ X = \left( \frac{\mu_L}{\mu_v} \right)^{0.1} \left( \frac{1 - \tilde{Y}}{\tilde{Y}} \right)^{0.9} \left( \frac{\rho_v}{\rho_L} \right)^{0.5} \]  

(2.1.11)

where \( \rho_v \) = the average vapor density between the inlet and outlet

- \( \mu_L \) = the average liquid viscosity between the inlet and outlet
- \( \mu_v \) = the average vapor viscosity between the inlet and outlet

2.1.3 Forced Convection Heat-Transfer Model

The heat-transfer equations used in the steady-state model are as follows

Two-phase heat transfer using the correlation proposed by John C. Chen (1963)

\[ \frac{Q}{A} = \left[ h_{FC} F + h_{PZ} S \right] \Delta T \]  

(2.1.12)

where

\[ h_{FC} = 0.023 \left( \frac{DG}{\mu_L} \right)^{0.8} \left( \frac{\mu_L C_L}{k_L} \right)^{0.4} \left( \frac{k_L}{D} \right) \]  

(2.1.13)

\[ h_{PZ} = 0.00122 \frac{k_f^{0.1} C_f^{0.45} \rho_f^{0.49} g_s^{0.25} \Delta T^{0.24} \Delta P^{0.75}}{\sigma^{0.5} \mu_L^{0.29} \lambda^{0.24} \rho_v^{0.24}} \]  

(2.1.14)
\[ F = f(X_u) \]  
\[ \text{Re}_L = \frac{DG(1 - Y)}{\mu_L} \]  
\[ \Delta T = T_w - T_s \]  
\[ \Delta P = \frac{\Delta T \rho \lambda}{T_s} \]  

The single-phase heat-transfer correlation used in the model (liquid and superheated gas)

\[ \frac{Q}{A} = h \Delta T \]  
\[ h = 0.023 \left( \frac{DG}{\mu_L} \right)^{0.8} \left( \frac{\mu_L C_L}{k_L} \right)^{0.4} \left( \frac{k_L}{D} \right) \]  

2.1.4 Spray Temperature

To shorten the run time of the zero-g TVS model, an alternative was provided to calculate the spray temperature as a function of the liquid subcooling and tank pressure. This is obtained from an energy balance between the hot and cold fluids of the heat exchanger

\[ T_s = T_p - \frac{m_v \Delta h}{m_p c_{pl}} \]  

where \( T_s \) is the spray temperature  
\( T_p \) is the pump temperature  
\( m_v \) is the vent flow rate  
\( m_p \) is the pump flow rate  
\( \Delta h \) is the heat absorption capability of the vent flow  
\( c_{pl} \) is the heat capacity of the liquid

The TVS vent flow rate and heat absorption capability were calculated as a function of the liquid subcooling and tank flow rate and are shown in Figs. 2.1 and 2.2, respectively. This data is provided as a table look-up to the zero-g TVS model.

2.2 Spray Manifold and Injection Tube Model

Fluid is recirculated from the tank to the spray manifold and injection tubes where it is sprayed into the ullage and liquid. A one-dimensional, incompressible fluid dynamic model was developed to determine the pressures in the spray manifold and injection tubes, and to calculate the spray flow rates and velocities leaving the injection orifices. Following is a description of the model.
Figure 2.1.1 TVS Vent Flow Rate as a Function of Liquid Subcooling
Figure 2.1.2 Heat Absorption Capability of Vent Flow as a Function of Liquid Subcooling
2.2.1 Spray Manifold

The spray manifold calculates the pressure drop through the manifold and determines the pressure at the inlet of the spray injection tubes (Fig. 2.2.1). The model accounts for frictional pressure drop in the manifold, and pressure losses resulting from flow turning and contraction at the exit of the manifold. From Bernoulli equation

\[
\frac{(p_{SM})_i}{\rho} + \frac{V_{SM}^2}{2g_e} + az_i = \frac{(p_{SM})_o}{\rho} + \frac{V_{SM}^2}{2g_e} + az_o + (h_{L})_{SM}
\]  \hspace{1cm} (2.2.1)

where \( (p_{SM})_i \) is the spray manifold inlet pressure
\( (p_{SM})_o \) is the spray manifold outlet pressure
\( V_{SM} \) is the velocity in the spray manifold
\( z_i, z_o \) are the inlet and outlet elevations
\( a = \frac{g}{g_e} \) is the acceleration

The total head loss is defined as

\[
(h_{L})_{SM} = K_{SM} \frac{V_{SM}^2}{2g_e}
\]  \hspace{1cm} (2.2.2)

The total loss coefficient \( K_{SM} \) is given by

\[
K_{SM} = (K_f)_{SM} + (K_b)_{SM} + (K_c)_{SM}
\]  \hspace{1cm} (2.2.3)

and includes

\[
(K_f)_{SM} = f_{SM} \left( \frac{L}{D_{SM}} \right)
\]  \hspace{1cm} (spray manifold frictional loss coefficient ) \hspace{1cm} (2.2.4a)

\[
(K_b)_{SM} = f_{SM} \left( \frac{L_s}{D_{SM}} \right)
\]  \hspace{1cm} (90-degree bend resistance at the manifold exit ) \hspace{1cm} (2.2.4b)

\[
(K_c)_{SM} = 0.5 \left[ 1 - \left( \frac{D_{SM}}{D_{SM}} \right)^2 \right]
\]  \hspace{1cm} (sudden contraction at the manifold exit ) \hspace{1cm} (2.2.4c)

In Eqs. 2.2.4,

\( L_{SM} = z_o - z_i \) is the spray manifold length
\( L_e \) is the bend equivalent length
\( D_{SI} \) is the spray injection tube ID
\( D_{SM} \) is the spray manifold ID
\( f_{SM} \) is the friction coefficient in the spray manifold obtained from
Figure 2.2.1  Spray Manifold Model
Eq. 2.2.1 can be solved for the spray manifold outlet pressure

\[
(p_{SM})_o = (p_{SM})_i - K_{SM} q_{SM} - \rho a L_{SM}
\]  

(2.2.6)

where the dynamic pressure \( q_{SM} \) in the spray manifold is given by

\[
q_{SM} = \rho \frac{v_{SM}^2}{2g_c} = \frac{1}{2\rho g_c} \left( \frac{m_s}{A_{SM}} \right)^2
\]  

(2.2.7)

### 2.2.2 Spray Injection Tube

The spray injection tube model is a multinode model which assigns a node to each orifice (Fig 2.2.2). Bernoulli's equation is first applied to find the pressure downstream of the inlet 90 degree bend of the injection tube (pressure at the inlet of the straight section)

\[
p_i = (p_{SM})_o - q_i (K_b)_{SI}
\]  

(2.2.8)

In Eq. 2.2.8, \((K_b)_{SI}\) is the 90 degree bend resistance and \(q_i\) is the inlet dynamic pressure given by

\[
q_i = \frac{1}{2\rho g_c} \left( \frac{m_i}{A_{SI}} \right)^2
\]  

(2.2.9)

where \(A_{SI}\) is the flow area of an injection tube and \(m_i\) is the mass flow rate in each tube (equal to the flow rate in the manifold divided by the number of tubes).

The straight section of the spray injection tube is divided into 45 equal nodes corresponding to the 45 spray orifices. Each node has a pressure and a mass flow rate at the inlet (i), center, and outlet (o) of the node. The outlet pressure and mass flow rate of one node is therefore the inlet pressure and mass flow rate of the preceding node

\[
(p_i)_n = (p_o)_{n-1}
\]  

(2.2.9a)

\[
(m_i)_n = (m_o)_{n-1}
\]  

(2.2.9b)

Bernoulli's equation is applied successively from inlet to center, and from center to outlet to determine the pressure at the center and outlet of a node n.
From inlet to center,

\[ p_n = (p_i)_n + \rho a \frac{\Delta z}{2} - K_f(q_i)_n \]  

(2.2.10)

where \( \Delta z \) is the nodal length and \( K_f \) is the frictional loss coefficient.

From center to outlet,

\[ (p_o)_n = p_n + \rho a \frac{\Delta z}{2} - K_f(q_o)_n \]  

(2.2.11)

where the outlet dynamic pressure \( (q_o)_n \) of node \( n \) is given by

\[ (q_o)_n = \frac{1}{2 \rho g_e} \left[ \frac{(m_o)_n}{A_{st}} \right]^2 \]  

(2.2.12)
Figure 2.2.2  Spray Injection Tube Model
The mass flow rate at node n outlet $\dot{m}_o$ is obtained from

$$\dot{m}_o = \dot{m}_i - \dot{m}_s$$  \hspace{1cm} (2.2.13)

$\dot{m}_s$ in Eq. 2.2.13 is the spray flow rate calculated from an incompressible flow relation

$$\dot{m}_s = (A_s)_n \sqrt{\frac{2 \rho g \left[ p_n - (p_T)_n \right]}{K_s}}$$  \hspace{1cm} (2.2.14)

In Eq. 2.2.14, $K_s$ is the loss coefficient of an orifice in a duct given by

$$K_s = \left[ \frac{1}{C_d} - \frac{A_s}{A_T} \right]^2$$  \hspace{1cm} (2.2.15)

where $C_d$ is the discharge coefficient ($C_d = 0.8$), and $A_S/A_T$ is the ratio of the orifice to the tank area ($A_S/A_T = 0$). Thus, $K_s$ is determined to be 1.56.

The tank pressure $(p_T)_n$ at node n is calculated as

$$(p_T)_n = p_u \quad \text{ (ullage nodes)}$$

$$(p_T)_n = p_u + \rho_L g z_n, \quad \text{ (liquid nodes)}$$  \hspace{1cm} (2.2.16)

where $z_n$ is the distance from the liquid surface to node n.

**2.2.3 Spray Manifold and Injection Tube Model Algorithm**

The flow chart of the spray manifold and injection tube model is given in Section 3.1.2. The model starts out with a guess of the pump flow rate and calculates the pressures and mass flow rates at each node. Knowing the pressure and spray flow rate of the last node $N$, it then calculates the tank pressure corresponding to that last node by solving the incompressible flow relation of Eq. 2.2.14

$$\left( p_T \right)_{N, \text{calc}} = p_N - \frac{K_s}{2 \rho g} \sqrt{\left( \frac{\dot{m}_s}{(A_s)_N} \right)^2}$$  \hspace{1cm} (2.2.17)

Next, $(p_T)_{N, \text{calc}}$ is compared with $(p_T)_N$ obtained from the ullage pressure and hydrostatic head (Eq. 2.2.16). If they are not equal within a specified tolerance (0.001 psi), a new guess of the pump flow rate will be made and the process repeated until convergence on $(p_T)_N$ is achieved.

**2.3 Recirculation Pump Model**
The zero-g TVS LH2 recirculation pump is a centrifugal pump which is a constant output pressure device since it imparts kinetic pressure to the fluid due to rotation. Consequently, the pump pressure rise ($\Delta p_p$) is only a function of rotation speed ($N$) and tip velocity ($U$)

$$U = \frac{\pi D_m N}{720}$$  \hspace{1cm} (2.3.1)

where $D_m$ is the impeller diameter.

The fluid horsepower required by the pump flow ($m$), raised to $\Delta p_p$ pressure, is equal to

$$HP_o = \frac{m \Delta p_p}{\eta_p \rho}$$  \hspace{1cm} (2.3.2)

where $\eta_p$ is the pump mechanical efficiency.

The pump operating speed then changes as a result of the energy absorbed by the fluid and the power supplied to the pump through a power source. The instantaneous rate of change in pump operating speed is

$$\frac{dN}{dt} = \left(\frac{HP_I - HP_o}{I_p N}\right) \cdot 6.0185 \times 10^5$$  \hspace{1cm} (2.3.3)

where $I_p$ is the polar moment of inertia of the pump and $HP_I$ is the input power to the pump.

Integration of the pump acceleration results in the pump speed at any given time

$$N = (N)_{0c} + \int \left(\frac{dN}{dt}\right) dt$$  \hspace{1cm} (2.3.4)

By specifying the initial pump speed at zero, a pump start transient may be simulated.

A pump head-flow curve was provided by the pump manufacturer, Barber-Nichols Engineering Co. (Fig. 2.3.1). The curve was fitted with a polynomial function to give the head coefficient ($\psi$) as a function of the flow coefficient ($\phi$)

$$\psi = 0.52889 - 1.4956\phi + 47.819\phi^2 - 485.93\phi^3 + 1633.9\phi^4 - 1833.5\phi^5$$  \hspace{1cm} (2.3.5)

The flow coefficient $\phi$ is obtained from test data in terms of the flow rate (in gpm) and the pump speed as

$$\phi = \frac{\text{gpm}}{0.0531N}$$  \hspace{1cm} (2.3.6)

The pump head is calculated from the pump speed and head coefficient

$$H = 4.507 \times 10^{-6} N^2 \psi$$  \hspace{1cm} (2.3.7)
Figure 2.3.1 LH2 Recirculation Pump Head-Flow Curve

The equation for the curve is:

\[ y = 0.52689 - 1.48556x + 47.819x^2 - 485.93x^3 + 1633.9x^4 - 1633.5x^5 \]

\[ R^2 = 0.999 \]
The pump pressure rise is then obtained as

\[ \Delta p_p = \frac{\rho H}{144} \quad (2.3.8) \]

The lumped pump model requires the pump design flow rate (Q_D) and speed (N_D) in order to define the other operating characteristics (HP, I_p) required by the model.

2.4 Tank Thermal Model

The tank model is a lumped model consisting of four control volumes (Fig. 2.4.1): (1) ullage, (2) tank wall, (3) liquid on the tank wall, and (4) bulk liquid. The thermal model of each control volume is described in the following.

2.4.1 Ullage

The ullage thermal model applies conservation of mass and energy to determine the ullage pressure, temperature and mass (Fig. 2.4.2). From conservation of mass, the change in the ullage mass (M_U) is due to all masses entering and leaving the ullage control volume

1. droplet evaporation rate in the ullage (m_{DU})
2. boiling rate of the liquid on the tank wall (m_{BW})
3. bulk liquid boiling rate (m_{LU}), or ullage condensation (m_{UL})
4. liquid surface condensation (m_{COND})

\[ \frac{dM_u}{dt} = m_{DU} + m_{BW} + m_{LU} - m_{UL} - m_{COND} \quad (2.4.1) \]

These mass flow rates are defined in Section 2.4.6. The ullage mass is obtained by integrating its time rate of change with respect to time

\[ M_u = (M_u)_i + \int \frac{dM_u}{dt} \, dt \quad (2.4.2) \]

From conservation of energy, the change in the ullage temperature (T_U) is the result of

1. heat transfer to the ullage (q_U)
2. work done on the ullage (w_U)
3. energy added to the ullage by incoming and leaving masses (ENTH_U)

\[ \frac{dT_u}{dt} = \frac{q_u - w_u - ENTH_U - c_{vu} T_u \frac{dM_u}{dt}}{M_u c_{vu}} \quad (2.4.3) \]
Figure 2.4.1 Tank Thermal Model
Figure 2.4.2 Ullage Thermal Model
The terms in Eq. 2.4.3 are defined as follows

\( q_U = q_{WU} - q_{UWL} - q_{UL} - q_{UD} - q_{US} \)  

(heat transfer to ullage)

where

- \( q_{WU} \) is the heat-transfer rate between the tank wall and ullage,
  - \(|q_{WU}| > 0\) for a dry wall
  - \(q_{WU} = 0\) for a wet wall

- \( q_{UWL} \) is the heat-transfer rate between the ullage and wall liquid,
  - \(|q_{UWL}| = 0\) for a dry wall
  - \(q_{UWL} > 0\) for a wet wall

- \( q_{UL} \) is the heat-transfer rate between the ullage and bulk liquid

- \( q_{UD} \) is the heat-transfer rate between the ullage and liquid droplet

- \( q_{US} \) is the heat-transfer rate between the ullage and (unsubmerged) spray bars

The above heat-transfer rates are defined in Section 2.4.5.

\( w_u = p_u \frac{dV_u}{dt} \)  

(work done on ullage)

where the change in the ullage volume \( \left( \frac{dV_u}{dt} \right) \) is equal and opposite to the change in the liquid and wall liquid volumes

\[
\frac{dV_u}{dt} = - \frac{dV_L}{dt} - \frac{dV_{WL}}{dt}
\]

(2.4.4)

\( \text{ENTH}_u = \left( \frac{dM_u}{dt} \right) h_{\text{sat}} \)

where \( h_{\text{sat}} = h_{\text{sat}}(p_u) \) is the saturated vapor enthalpy of the ullage.

The ullage volume is obtained as the difference between the tank volume and the bulk liquid and wall liquid volumes

\( V_U = V_T - V_L - V_{WL} \)  

(2.4.5)

Eq. 2.4.3 is integrated with respect to time to obtain the ullage temperature

\( T_U = (T_U)_0 + \int \left( \frac{dT_u}{dt} \right) dt \)

(2.4.6)

With the ullage mass, temperature and volume determined, the ullage pressure is calculated from the equation of state


\[
\frac{p_U}{V_U} = \frac{M_U R_U T_U}{V_U} \quad (2.4.7)
\]

### 2.4.2 Tank Wall

The tank wall is divided into two sections, one facing the liquid and the other facing the ullage. The tank wall facing the bulk liquid is assumed to be at the same temperature as the liquid. Thus, the tank wall thermal model described in this section applies to the section facing the ullage (Fig. 2.4.3). Since liquid can form on the tank wall as a result of spraying, the model must account for both dry and wet wall cases.

From conservation of energy, the change in the tank wall temperature is due to

1. heat input to the wall from the environment \( q_{EW} \)
2. heat-transfer rate between the wall and ullage \( q_{wu} \)
   - \( |q_{wu}| > 0 \) for a dry wall
   - \( = 0 \) for a wet wall
3. heat-transfer rate between the wall and liquid on the wall \( q_{WL} \)
   - \( |q_{WL}| = 0 \) for a dry wall
   - \( > 0 \) for a wet wall

\[
\frac{dT_w}{dt} = \frac{q_{EW} - q_{wu} - q_{WL}}{M_w c_{pw}} \quad (2.4.8)
\]

Section 2.4.5 defines these heat-transfer rates. Eq. 2.4.8 can be integrated with respect to time to obtain the tank wall temperature

\[
T_w = (T_w)_i + \int \left( \frac{dT_w}{dt} \right) \quad (2.4.9)
\]

### 2.4.3 Wall Liquid

The wall liquid thermal model is also governed by the laws of conservation of mass and energy (Fig. 2.4.4). From conservation of mass, the change in the wall liquid mass \( M_{WL} \) is equal to the difference between the liquid mass reaching the wall and the liquid mass boiled off from the wall

\[
\frac{dM_{WL}}{dt} = m_{sw} - m_{bw} \quad (2.4.10)
\]

where \( m_{sw} \) is the spray flow rate reaching the wall and \( m_{bw} \) is the liquid boil-off rate from the wall.

These mass flow rates will be defined in Section 2.4.6. Eq. 2.4.10 can be integrated to obtain the wall liquid mass.
Figure 2.4.3 Tank Wall Thermal Model

Figure 2.4.4 Wall Liquid Thermal Model
From conservation of energy, the change in the wall liquid temperature ($T_w$) is the result of heat transfer to the wall liquid and sensible energy added to the spray to raise its temperature ($T_{sw}$) to the wall liquid temperature. Heat transfer to the wall liquid includes heat-transfer rate between the wall and wall liquid ($q_{WL}$), and heat-transfer rate between the ullage and wall liquid ($q_{UWL}$).

$$\frac{dT_{WL}}{dt} = \frac{q_{WL} + q_{UWL} - m_{sw} c_{pl} (T_{WL} - T_{sw})}{M_{WL} c_{pWL}}$$

(2.4.12)

These heat-transfer rates are defined in Section 2.4.5. Eq. 2.4.12 can be integrated to obtain the wall liquid temperature

$$T_{WL} = (T_{WL})_C + \int \left( \frac{dT_{WL}}{dt} \right) dt$$

(2.4.13)

The wall liquid vapor pressure is then obtained from the thermodynamic data base as

$$P_{WL} = p_{sat}(T_{WL})$$

(2.4.14)

The volume rate of change of the wall liquid is determined from Eq. 1.9 as

$$\frac{dV_{WL}}{dt} = \frac{1}{\rho_{WL}} \frac{dM_{WL}}{dt}$$

(2.4.15)

where $\rho_{WL} = \rho_{sat}(T_{WL})$ is the wall liquid density.

Eq. 2.4.15 is integrated to obtain the wall liquid volume

$$V_{WL} = (V_{WL})_C + \int \left( \frac{dV_{WL}}{dt} \right) dt$$

(2.4.16)

2.4.4 Bulk Liquid

Originally conceived as multi-node, the bulk liquid thermal model is made single node since (1) mixing will destratify the liquid and create a uniform bulk, and (2) uncertainty in heat-transfer modeling does not justify the added complexities of a multinode model.

The liquid thermal model is also based on the laws of conservation of mass and energy. From conservation of mass, the change in the liquid mass must be balanced by a change in the ullage mass and any mass vented overboard (Fig. 2.4.5).

$$\frac{dM_{L}}{dt} = \dot{m}_{SL} + \dot{m}_{SUL} + \dot{m}_{COND} + \dot{m}_{UL} - \dot{m}_{UL} - \dot{m}_{S} - \dot{m}_{V}$$

(2.4.17)
Figure 2.4.5 Bulk Liquid Thermal Model
where \( m_{SL} \) is the liquid spray flow rate into the bulk liquid
\( m_{UL} \) is the unevaporated droplet flow rate
\( m_{COND} \) is the liquid surface condensation flow rate
\( m_{UL} \) is the ullage condensation flow rate
\( m_{LU} \) is the liquid boil-off rate
\( m_s \) is the pump flow rate
\( m_v \) is the overboard venting flow rate.

The liquid mass is obtained by integrating its time rate of change

\[
M_L = (M_{L})_{IC} + \int \left( \frac{dM_L}{dt} \right) \quad (2.4.18)
\]

From conservation of energy, the change in the liquid temperature is caused by

1. heat transfer to the liquid
2. heat added by the unevaporated droplets
3. sensible energy added to the liquid spray to raise its temperature \( T_s \) to the liquid temperature
4. latent heat of vaporization of the liquid

\[
\frac{dT_L}{dt} = \frac{q_L + m_{UL} c_{PL} (T_d - T_L) - m_{LU} (h_g)_{L} - m_s c_{PL} (T_L - T_s)}{M_L c_{PL}} \quad (2.4.19)
\]

The heat-transfer rate to the liquid \( (q_L) \) is given by

\[
q_L = q_{EL} + q_{UL} - q_{LS}
\]

where \( q_{EL} \) is the heat added to the liquid by the environment
\( q_{UL} \) is the heat-transfer rate between the ullage and liquid
\( q_{LS} \) is the heat-transfer rate between the liquid and (submerged) spray bars

These heat-transfer rates are given in Section 2.4.5. Eq. 2.4.19 is integrated with respect to time to give the liquid temperature

\[
T_L = (T_L)_{IC} + \int \left( \frac{dT_L}{dt} \right) dt \quad (2.4.20)
\]

The liquid vapor pressure is obtained from the thermodynamic data base as

\[
p_L = P_{sat}(T_L) \quad (2.4.21)
\]

The liquid volume rate of change is determined from the rate of change of the liquid mass
\[
\frac{dV_L}{dt} = \frac{1}{\rho_L} \frac{dM_L}{dt}
\]

(2.4.22)

where \(\rho_L = \rho_{\text{sat}}(T_L)\) is the liquid density.

Eq. 2.4.22 is integrated to give the liquid volume

\[
V_L = (V_L)_0 + \int \left( \frac{dV_L}{dt} \right) dt
\]

(2.4.23)

2.4.5 Heat Transfer

This section defines the heat-transfer rates which are found in the energy balances of Section 2.4.1 to 2.4.4. These heat-transfer rates can be divided into two groups: free convection and forced convection. Free convection is the dominant heat-transfer mode in the ullage and liquid, while forced convection characterizes liquid droplet heat transfer in the ullage.

The convection heat-transfer rate is generally defined as

\[
q = hA\Delta T
\]

where \(h\) is the convection heat-transfer coefficient

- \(A\) is the surface area of heat transfer

- \(\Delta T\) is the temperature difference between the heat source and sink

The heat-transfer coefficient is obtained from the Nusselt Number (Nu) as

\[
h = \left( \frac{k_f}{L_c} \right) \text{Nu}
\]

where \(k_f\) is the fluid thermal conductivity and \(L_c\) is the surface characteristic length.

The Nusselt number is a function of the Rayleigh number (Ra) defined as

\[
Ra = \frac{a \beta \Delta T L_c^3 \rho^2 c_p}{\mu k}
\]

(2.4.24)

where

- \(a\) is the acceleration

- \(\beta\) is the thermal expansion coefficient,

\[
\beta = \frac{1}{T_f} \text{ for gas, } \frac{1}{\rho} \left( \frac{\partial \rho}{\partial T} \right)_p \text{ for liquid}
\]

- \(L_c\) is the characteristic length

- \(\rho\) is the density

- \(c_p\) is the specific heat at constant pressure

- \(\mu\) is the dynamic viscosity

- \(k\) is the thermal conductivity
All properties must be evaluated at the film temperature ($T_f$) which is defined as the average of the fluid and surface temperatures.

2.4.5.1 Free Convection

Two free convection heat-transfer correlations are used in the model. The first one is a free convection correlation for interior surfaces of vertical ducts, vertical plates and cylinders, and horizontal cylinders (Ref. 28) (Fig 2.4.6)

$$Nu = 0.555Ra^{0.25} + 0.447 \quad (2.4.25)$$

This correlation is used to calculate the heat-transfer coefficients

1. between the ullage and wall ($h_{uw}$)
2. between the ullage and bulk liquid ($h_{ul}$)
3. between the ullage and wall liquid ($h_{uwl}$)
4. between the ullage and (unsubmerged) spray bars ($h_{us}$)
5. between the bulk liquid and (submerged) spray bars ($h_{ls}$)

The characteristic length for $h_{uw}$, $h_{ul}$ and $h_{uwl}$ is the internal tank diameter while that of $h_{us}$ and $h_{ls}$ is the spray bar diameter.

The second correlation is the McAdams correlation for free convection of vertical surfaces in the turbulent range (Ref. 17)

$$Nu = 0.13Ra^{1/3} \quad (2.4.26)$$

This correlation is used to calculate the heat-transfer coefficient between the wall and wall liquid ($h_{wl}$). Because of the $1/3$ power in $Ra$, $h_{wl}$ can be obtained without knowing the characteristic length, thereby removing the uncertainty in determining the wall liquid layer.

2.4.5.2 Forced Convection

The forced convection heat-transfer coefficient between the ullage and liquid droplets ($h_{ud}$) is based on a McAdams recommended correlation for flow over a sphere (Ref. 8) (Fig. 2.4.7)

$$Nu = 0.3125Re^{0.602} \quad (2.4.27)$$

The Reynolds number of the spray flow ($Re$) is defined as

$$Re = \frac{\rho Vel_D D_D}{\mu} \quad (2.4.28)$$

where $Vel_D$ is the droplet velocity in the ullage

$D_D$ is the droplet diameter assumed to be equal to the orifice diameter

$\rho, \mu$ are the density and viscosity of the ullage gas

Since the droplet diameter and velocity vary with the orifice size, the droplet heat-transfer
Figure 2.4.6 Free Convection Heat-Transfer Correlation for Interior Surfaces of Vertical Ducts, Vertical Plates and Cylinders, and Horizontal Cylinders
Figure 2.4.7  Forced Convection Heat-Transfer Correlation for Flow Over a Sphere
coefficient must be determined for each orifice. The total droplet heat-transfer rate is obtained by summing the droplet heat-transfer rates from each orifice

\[ q_{\text{drop}} = \sum_{i=1}^{n} (n_{\text{drop}})_{i} (q_{\text{drop}})_{i} \]  

(2.4.29)

where \((n_{\text{drop}})_{i}\) is the number of droplets sprayed from orifice \(i\) into the ullage. This is given by

\[ (n_{\text{drop}})_{i} = \left( \frac{\dot{m}_{\text{SU}}}{2\rho_{D}(V_{D})_{i}(V_{\text{elD}})_{i}} \right) D_{\text{CHAR}} \]  

(2.4.30)

where \((\dot{m}_{\text{SU}})_{i}\) is the spray flow rate into the ullage from orifice \(i\), \((V_{D})_{i}\) and \((V_{\text{elD}})_{i}\) are the droplet volume and velocity from orifice \(i\), \(\rho_{D}\) is the droplet density, and \(D_{\text{CHAR}}\) is a characteristic length determined empirically.

By correlating the zero-g TVS model with LeRC ullage pressure collapse data, this characteristic length was determined to be \(1/4\) of the tank diameter.

2.4.6 Mass Transfer

This section defines the mass-transfer rates found in the mass balance equations of Section 2.4.1 to 2.4.4 which include

(1) Bulk liquid boiling (\(\dot{m}_{\text{LU}}\))

(2) Liquid boiling from the tank wall (\(\dot{m}_{\text{bw}}\))

(3) Liquid droplet evaporation in the ullage (\(\dot{m}_{\text{DU}}\))

(4) Liquid spray falling into the bulk liquid (\(\dot{m}_{\text{SU}}\)) or accumulating on the tank wall (\(\dot{m}_{\text{sw}}\))

(5) Ullage condensation (\(\dot{m}_{\text{UL}}\))

(6) Liquid surface condensation (\(\dot{m}_{\text{COND}}\))

2.4.6.1 Bulk Liquid Boiling

Bulk liquid boiling occurs when the liquid vapor pressure is equal to the tank ullage pressure. It can be the result of heat transfer to the liquid and/or pressure decay in the ullage. It must also include sensible energy added to the liquid spray to increase its temperature to the liquid temperature.

If \(P_{L} = P_{U}\),

\[ \dot{m}_{\text{LU}} = \frac{1}{h_{L}} \left[ q_{L} - \dot{m}_{\text{SL}} c_{\nuL} (T_{L} - T_{S}) \right], \quad \frac{dP_{U}}{dt} < 0 \]
\[ \frac{1}{(h_f)_L} \left[ q_L - \dot{m}_{SL} c_{pl} (T_L - T_s) - M_L c_{pl} \left( \frac{\partial T}{\partial p} \right)_{sat} \left( \frac{dp_u}{dt} \right) \right], \quad \frac{dp_u}{dt} > 0 \]  

(2.4.31)

A polynomial fit of the LH2 saturation temperature vs. pressure curve was obtained and its derivative taken to give an expression for \( \left( \frac{\partial T}{\partial p} \right)_{sat} \)

\[ \left( \frac{\partial T}{\partial p} \right)_{sat} = 0.37781 - 4.9170 \times 10^{-3} p_L + 21.7623 \times 10^{-6} p_L^2 \]  

(2.4.32)

If the ullage pressure increases above the liquid vapor pressure, boiling stops

\[ \dot{m}_{LU} = 0, \text{ if } p_L < p_U' \]

2.4.6.2 Wall Liquid Boiling

Wall liquid boiling from the tank wall follows the same mechanism as bulk liquid boiling

If \( p_{WL} = p_U' \)

\[ \dot{m}_{WL} = \frac{1}{(h_f)_L} \left[ q_{WL} + q_{UWL} - \dot{m}_{sw} c_{pl} (T_{WL} - T_{sw}) \right], \quad \frac{dp_u}{dt} < 0 \]  

(2.4.33)

\[ = \frac{1}{(h_f)_L} \left[ q_{WL} + q_{UWL} - \dot{m}_{sw} c_{pl} (T_{WL} - T_{sw}) - M_{WL} c_{pl} \left( \frac{\partial T}{\partial p} \right)_{sat} \left( \frac{dp_u}{dt} \right) \right], \quad \frac{dp_u}{dt} > 0 \]  

where \( \left( \frac{\partial T}{\partial p} \right)_{sat} = 0.37781 - 4.9170 \times 10^{-3} p_{WL} + 21.7623 \times 10^{-6} p_{WL}^2 \)  

(2.4.34)

If \( p_{WL} < p_U' \), \( \dot{m}_{sw} = 0 \).

As with bulk boiling, wall liquid boiling includes heat transfer to the wall liquid and sensible energy added to the spray liquid to bring its temperature to the wall liquid temperature.

2.4.6.3 Liquid Droplet Evaporation in the Ullage

Liquid droplets in the ullage will start boiling once the subcooled liquid spray is brought to saturation. From an energy balance on the liquid droplets, an expression for the liquid droplet boiling is obtained

\[ \dot{m}_{DU} = \frac{1}{(h_f)_U} \left[ q_{UD} - \dot{m}_{su} c_{pl} (T_{U_{sat}} - T_s) \right] \]  

(2.4.35)

where \( T_{U_{sat}} = T_{sat}(p_U) \) is the ullage saturation temperature.
2.4.6.4 **Liquid Spray Falling into the Bulk Liquid or Accumulating on the Tank Wall**

The unevaporated sprayed mass in the ullage is assumed to fall into the bulk liquid under 1 g, or to accumulate on the tank wall under 0 g (Fig. 2.4.8), i.e.,

\[ m_{su} = m_{su} - m_{du} \quad \text{(for 1 g)} \]  \hspace{1cm} (2.4.36)

\[ m_{sw} = m_{su} - m_{du} \quad \text{(for 0 g)} \]

2.4.6.5 **Ullage Condensation**

Ullage condensation occurs whenever the ullage temperature is equal to the saturation temperature corresponding to the ullage pressure. It is the result of heat removal from the liquid droplet (when there is spraying) and the wall liquid (Figure 2.4.9)

\[ n_{ul} = \frac{q_{ul} + q_{ul} + q_{uwl}}{(h_{fs})_u} \quad T_u = T_{sat}(p_u) \]  \hspace{1cm} (2.4.37)

2.4.6.6 **Liquid Surface Condensation**

When helium is not present to act as a barrier to mass transfer, bulk liquid mixing during pump operation induces condensation on the liquid surface. This condensation rate is controlled by the heat transfer rate from the ullage to the liquid

\[ m_{Cond} = \frac{q_{ul}}{(h_{fs})_u} \]  \hspace{1cm} (2.4.38)

2.5 **References**


Figure 2.4.8 Droplet Evaporation Model

Figure 2.4.9 Ullage Condensation Model


II-32


2.6 Sample Cases

The sample cases shown are for a tank with 90% and 25% liquid quantities and a 0.25 Btu/hr-ft² heat flux. The tank is the 639-ft³ Multi-purpose Hydrogen Test Bed (MTHB) tank which is a cylindrical tank with elliptical bulkheads at both ends. The cylinder measures 5 ft in length and 10 ft in diameter while the bulkhead has a height of 2.5 ft. The tank has a wall thickness of 0.5 in and is made of aluminum. One-g acceleration is assumed and no helium is present in the tank. The results show the ullage and liquid vapor pressures, recirculation and vent flow rates, time between destratification and venting, destratification time, and TVS operation frequency.
Figure 2.6.1 TVS Performance Simulation at 90% Liquid Quantity

Figure 2.6.2 TVS Recirculation Pump and Vent Valve Flow Rate Transient During Ullage Destratification (90% Liquid Quantity)
Figure 2.6.3 TVS Performance Simulation at 25% Liquid Quantity

Figure 2.6.4 TVS Performance Simulation at 25% Liquid Quantity
Figure 2.6.5  TVS Recirculation Pump and Vent Valve Flow Rate Transient During Ullage Destratification (25% Liquid Quantity)

Figure 2.6.6  TVS Operation Frequency (Percent) as a Function of Liquid Quantity
SECTION 3
COMPUTER MODEL DESCRIPTION

3.1 Programming Description

The Zero-g TVS performance prediction program was developed on the following system:

<table>
<thead>
<tr>
<th>Computer</th>
<th>HP-9000 Series 500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>HP-UX rel. 5.2.1</td>
</tr>
<tr>
<td>Language</td>
<td>FORTRAN 77 rel. 5.12</td>
</tr>
<tr>
<td>Plotter</td>
<td>HP-7550</td>
</tr>
<tr>
<td>Plotting Software</td>
<td>CRTPLT</td>
</tr>
</tbody>
</table>

3.2 Flow Charts

3.2.1 Heat Exchanger Model

The flow chart of the heat exchanger model is shown in Section 1.4 of Reference 29.
3.2.2 Spray Manifold/Injection Tube Model

START

GUESS TOTAL SPRAY INJECTION FLOW RATE

CALCULATE SPRAY MANIFOLD OUTLET PRESSURE

CALCULATE PRESSURE AT INLET OF SPRAY INJECTION TUBE

DIVIDE SPRAY INJECTION TUBE INTO N NODES CORRESPONDING TO N ORIFICES

CALCULATE PRESSURE AND FLOW RATE AT EACH NODE IN THE SPRAY INJECTION TUBE, AND SPRAY FLOW RATE FROM EACH ORIFICE

CALCULATE TANK PRESSURE AT EACH NODE, \( (p_T)_N \)

CALCULATE TANK PRESSURE AT LAST NODE N FROM EQ. 2.2.16, \( (p_T)_{N,\text{calc}} \)

\( (p_T)_{N,\text{calc}} = (p_T)_N \)?

NO

YES

END
3.2.3 Integrated Zero-g TVS Model

START

TIME = 0

INTEGRATE RATES OF CHANGE OF ULLAGE, BULK LIQUID, AND WALL LIQUID MASSES

INTEGRATE RATES OF CHANGE OF ULLAGE, BULK LIQUID, AND WALL LIQUID VOLUMES

INTEGRATE RATES OF CHANGE OF BULK LIQUID, WALL LIQUID, AND WALL TEMPERATURES

CALCULATE ULLAGE, BULK LIQUID, AND WALL LIQUID PRESSURES

COMPARE ULLAGE PRESSURE WITH MIN & MAX CONTROL BAND PRESSURES, AND DETERMINE IF RECIRCULATION PUMP SHOULD BE OPERATING

GUESS PUMP FLOW RATE

CALCULATE PUMP PRESSURE RISE

COMPARE PUMP OUTLET SATURATION PRESSURE WITH MIN CONTROL BAND PRESSURE TO DETERMINE IF VENT VALVE SHOULD BE OPENED

CALCULATE PRESSURE DROP IN RECIRCULATION LINE

CALCULATE PRESSURE AT INLET OF SPRAY MANIFOLD TUBE
CALCULATE SPRAY INJECTION FLOW RATE

PUMP FLOW RATE = SPRAY FLOW RATE?

NO

YES

CALCULATE HEAT-TRANSFER RATES (WALL/ULLAGE, ULLAGE/WALL LIQUID, WALL/WALL LIQUID, ULLAGE/BULK LIQUID, ULLAGE/DROPLET, TANK WALL)

CALCULATE BOIL-OFF RATES (DROPLET, WALL LIQUID, BULK LIQUID), AND CONDENSATION RATES (ULLAGE AND LIQUID SURFACE)

CALCULATE RATES OF CHANGE OF ULLAGE, WALL LIQUID, AND BULK LIQUID MASSES

CALCULATE RATES OF CHANGE OF ULLAGE, BULK LIQUID, AND WALL LIQUID VOLUMES

CALCULATE RATES OF CHANGE OF ULLAGE, BULK LIQUID, WALL LIQUID, AND TANK WALL TEMPERATURES

TIME = TIME + DTIME

TIME = END TIME?

NO

YES

STOP

TIME = END TIME?
### 3.3 Definition of Variables

#### 3.3.1 Input Variables

#### 3.3.1.1 Heat Exchanger Model

The input variables of the heat exchanger model are described in Section 3.2.1 of Reference 29.

#### 3.3.1.2 Spray Manifold/Injection Tube Model

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>dsm</td>
<td>in</td>
<td>spray manifold diameter</td>
</tr>
<tr>
<td>zsm</td>
<td>in</td>
<td>spray manifold length</td>
</tr>
<tr>
<td>dsi</td>
<td>in</td>
<td>spray injection tube diameter</td>
</tr>
<tr>
<td>zsi</td>
<td>in</td>
<td>spray injection tube length</td>
</tr>
<tr>
<td>norf</td>
<td></td>
<td>number of orifices</td>
</tr>
<tr>
<td>nbar</td>
<td></td>
<td>number of spray bars</td>
</tr>
<tr>
<td>ks</td>
<td></td>
<td>spray orifice loss coefficient</td>
</tr>
<tr>
<td>cds</td>
<td></td>
<td>spray orifice discharge coefficient</td>
</tr>
<tr>
<td>roverd</td>
<td></td>
<td>bend r/d of spray manifold</td>
</tr>
<tr>
<td>dmdot</td>
<td>lbm/sec</td>
<td>flow rate increment</td>
</tr>
<tr>
<td>tol</td>
<td>psia</td>
<td>convergence tolerance on tank pressure</td>
</tr>
<tr>
<td>nlim</td>
<td></td>
<td>max number of iterations</td>
</tr>
<tr>
<td>nsec</td>
<td></td>
<td>number of sections with the same orifice sizes</td>
</tr>
<tr>
<td>node</td>
<td></td>
<td>node number</td>
</tr>
<tr>
<td>dorf</td>
<td>in</td>
<td>orifice diameter</td>
</tr>
</tbody>
</table>

#### 3.3.1.3 Recirculation Pump Model

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYMBOL</td>
<td>UNIT</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>mdotd</td>
<td>lbm/sec</td>
<td>design pump flow rate</td>
</tr>
<tr>
<td>dpd</td>
<td>psid</td>
<td>design pump pressure rise</td>
</tr>
<tr>
<td>npumpi</td>
<td>rpm</td>
<td>initial pump speed</td>
</tr>
<tr>
<td>npumpd</td>
<td>rpm</td>
<td>design pump speed</td>
</tr>
<tr>
<td>xhp</td>
<td></td>
<td>multiplier to determine design input horsepower</td>
</tr>
<tr>
<td>xn</td>
<td></td>
<td>fraction of design pump speed used to determine the pump speed operating band</td>
</tr>
<tr>
<td>deltat</td>
<td>sec</td>
<td>time needed to reach design speed</td>
</tr>
<tr>
<td>effp</td>
<td></td>
<td>pump efficiency</td>
</tr>
</tbody>
</table>

3.3.1.4 Integrated Zero-g TVS Model

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>xd</td>
<td></td>
<td>multiplier for spray injection orifice size (to determine droplet size)</td>
</tr>
<tr>
<td>xchar</td>
<td>in</td>
<td>characteristic length used in the equation to calculate the number of droplets</td>
</tr>
<tr>
<td>he</td>
<td></td>
<td>helium injection indicator (1 = yes, 0 = no)</td>
</tr>
<tr>
<td>xcond</td>
<td></td>
<td>multiplier for liquid surface condensation rate</td>
</tr>
<tr>
<td>prtsp</td>
<td></td>
<td>time indicator to print output of subroutine spray (output is printed prtsp&lt;time&lt;prtsp+0.1)</td>
</tr>
<tr>
<td>pui</td>
<td>psia</td>
<td>initial ullage pressure</td>
</tr>
<tr>
<td>tui</td>
<td>R</td>
<td>initial ullage temperature</td>
</tr>
<tr>
<td>pli</td>
<td>psia</td>
<td>initial ullage temperature</td>
</tr>
<tr>
<td>twi</td>
<td>R</td>
<td>initial wall temperature</td>
</tr>
<tr>
<td>twli</td>
<td>R</td>
<td>initial wall liquid temperature</td>
</tr>
<tr>
<td>full</td>
<td>%</td>
<td>percent full level</td>
</tr>
<tr>
<td>Symbol</td>
<td>Unit</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>xl1</td>
<td>in</td>
<td>length of straight section downstream of recirculation line 90-degree bend</td>
</tr>
<tr>
<td>dtank</td>
<td>in</td>
<td>tank diameter</td>
</tr>
<tr>
<td>hcyl</td>
<td>in</td>
<td>cylinder height</td>
</tr>
<tr>
<td>hbulk</td>
<td>in</td>
<td>tank bulkhead height</td>
</tr>
<tr>
<td>tkw</td>
<td>in</td>
<td>tank wall thickness</td>
</tr>
<tr>
<td>dsb</td>
<td>in</td>
<td>spray bar diameter</td>
</tr>
<tr>
<td>d1</td>
<td>in</td>
<td>diameter of recirculation line upstream of reducer</td>
</tr>
<tr>
<td>d2</td>
<td>in</td>
<td>diameter of recirculation line downstream of reducer</td>
</tr>
<tr>
<td>mdvent</td>
<td>lbm/sec</td>
<td>overboard venting flow rate</td>
</tr>
<tr>
<td>mdsi</td>
<td>lbm/sec</td>
<td>initial spray (pump) flow rate</td>
</tr>
<tr>
<td>dthex</td>
<td>R</td>
<td>heat exchanger temperature drop</td>
</tr>
<tr>
<td>qflux</td>
<td>Btu/hr-ft²</td>
<td>heat flux</td>
</tr>
<tr>
<td>g</td>
<td>ft/sec²</td>
<td>acceleration level</td>
</tr>
<tr>
<td>pmin</td>
<td>psia</td>
<td>control band min pressure</td>
</tr>
<tr>
<td>pmax</td>
<td>psia</td>
<td>control band max pressure</td>
</tr>
<tr>
<td>delt2</td>
<td>sec</td>
<td>integration time step when pump is off</td>
</tr>
<tr>
<td>iprint2</td>
<td>number of time steps between output printing when pump is off</td>
<td></td>
</tr>
<tr>
<td>iplot2</td>
<td>number of time steps between output plotting when pump is off</td>
<td></td>
</tr>
<tr>
<td>fintim</td>
<td>sec</td>
<td>end time</td>
</tr>
<tr>
<td>delt1</td>
<td>sec</td>
<td>integration time step when pump is on</td>
</tr>
<tr>
<td>iprint1</td>
<td>number of time steps between output printing when pump is on</td>
<td></td>
</tr>
</tbody>
</table>

III-7
iplot1  number of time steps between output plotting when pump is on

nline  number of output lines printed per page

ovariable  option to plot variable (1=yes)

subhd  plot subheading

xtitl  plot x-title

yvariable  y-title of variable plot

3.3.1.5 LH$_2$ saturation properties

nsat  number of data points

tsat  R  saturation temperature

psat  psia  saturation pressure

enthalpy  Btu/lbm  saturated liquid enthalpy

shpf  Btu/lbm·R  saturated liquid specific heat

densf  lbm/ft$^3$  saturated liquid density

texpf  R$^{-1}$  saturated liquid thermal expansion coefficient

condf  Btu/hr·ft·R  saturated liquid thermal conductivity

viscf  lbm/ft·sec  saturated liquid dynamic viscosity

enthalpy  Btu/lbm  saturated vapor enthalpy

3.3.1.6 GH$_2$ Properties

np (nt)  number of pressures (temperatures)

tnrm  normalized temperature

tconst  R  reference temperature

pvap  psia  pressure

tvap  R  temperature
enth  Btu/lbm  enthalpy
shv  Btu/lbm-R  specific heat at constant vapor
shp  Btu/lbm-R  specific heat at constant pressure
dens  lbm/ft$^3$  density
cond  Btu/hr-ft-R  thermal conductivity
visc  lbm/ft-sec  dynamic viscosity

3.3.2 Output Variables

3.3.2.1 Heat Exchanger Model

The output variables of the heat exchanger model are described in Section 3.2.2 of Reference 29.

3.3.2.2 Spray Manifold/Injection Tube Model

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>mdot</td>
<td>lbm/sec</td>
<td>spray (pump) flow rate</td>
</tr>
<tr>
<td>dppump</td>
<td>psid</td>
<td>pump pressure rise</td>
</tr>
<tr>
<td>dpsm</td>
<td>psi</td>
<td>spray manifold tube pressure drop</td>
</tr>
<tr>
<td>dpsi</td>
<td>psi</td>
<td>spray injection tube pressure drop</td>
</tr>
<tr>
<td>pin</td>
<td>psia</td>
<td>nodal inlet pressure</td>
</tr>
<tr>
<td>pout</td>
<td>psia</td>
<td>nodal outlet pressure</td>
</tr>
<tr>
<td>pnode</td>
<td>psia</td>
<td>nodal pressure</td>
</tr>
<tr>
<td>ptank</td>
<td>psia</td>
<td>tank pressure</td>
</tr>
<tr>
<td>mdin</td>
<td>lbm/sec</td>
<td>nodal inlet mass flow rate</td>
</tr>
<tr>
<td>mdout</td>
<td>lbm/sec</td>
<td>nodal outlet mass flow rate</td>
</tr>
<tr>
<td>mds</td>
<td>lbm/sec</td>
<td>spray flow rate through each orifice</td>
</tr>
</tbody>
</table>
### Symbol Table 3.3.2.3 Tank Model

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>pu</td>
<td>psia</td>
<td>ullage pressure</td>
</tr>
<tr>
<td>tu</td>
<td>R</td>
<td>ullage temperature</td>
</tr>
<tr>
<td>vu</td>
<td>ft³</td>
<td>ullage volume</td>
</tr>
<tr>
<td>mu</td>
<td>lbm</td>
<td>ullage mass</td>
</tr>
<tr>
<td>tw</td>
<td>R</td>
<td>wall temperature</td>
</tr>
<tr>
<td>pl</td>
<td>psia</td>
<td>bulk liquid pressure</td>
</tr>
<tr>
<td>tl</td>
<td>R</td>
<td>bulk liquid temperature</td>
</tr>
<tr>
<td>vl</td>
<td>ft³</td>
<td>bulk liquid volume</td>
</tr>
<tr>
<td>ml</td>
<td>lbm</td>
<td>bulk liquid mass</td>
</tr>
<tr>
<td>pwl</td>
<td>psia</td>
<td>wall liquid pressure</td>
</tr>
<tr>
<td>twl</td>
<td>R</td>
<td>wall liquid temperature</td>
</tr>
<tr>
<td>vwl</td>
<td>ft³</td>
<td>wall liquid volume</td>
</tr>
<tr>
<td>mwl</td>
<td>lbm</td>
<td>wall liquid mass</td>
</tr>
<tr>
<td>mv</td>
<td>lbm</td>
<td>mass vented overboard</td>
</tr>
<tr>
<td>ts</td>
<td>R</td>
<td>spray temperature</td>
</tr>
<tr>
<td>mdlu</td>
<td>lbm/sec</td>
<td>bulk liquid boil-off rate</td>
</tr>
<tr>
<td>mds</td>
<td>lbm/sec</td>
<td>spray (pump) flow rate</td>
</tr>
<tr>
<td>mdsl</td>
<td>lbm/sec</td>
<td>spray flow rate into bulk liquid</td>
</tr>
<tr>
<td>mdsu</td>
<td>lbm/sec</td>
<td>spray flow rate into ullage</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>UNIT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>veld</td>
<td>ft/sec</td>
<td>droplet velocity</td>
</tr>
<tr>
<td>as</td>
<td>in²</td>
<td>spray orifice area</td>
</tr>
<tr>
<td>Symbol</td>
<td>Unit</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>mddu</td>
<td>lbm/sec</td>
<td>droplet evaporation rate</td>
</tr>
<tr>
<td>mdbw</td>
<td>lbm/sec</td>
<td>wall liquid boil-off rate</td>
</tr>
<tr>
<td>mdul</td>
<td>lbm/sec</td>
<td>ullage condensation rate</td>
</tr>
<tr>
<td>mdcond</td>
<td>lbm/sec</td>
<td>liquid surface condensation rate</td>
</tr>
<tr>
<td>qwu</td>
<td>Btu/sec</td>
<td>heat-transfer rate between wall and ullage</td>
</tr>
<tr>
<td>quwl</td>
<td>Btu/sec</td>
<td>heat-transfer rate between ullage and wall liquid</td>
</tr>
<tr>
<td>qwl</td>
<td>Btu/sec</td>
<td>heat-transfer rate between wall and wall liquid</td>
</tr>
<tr>
<td>qul</td>
<td>Btu/sec</td>
<td>heat-transfer rate between ullage and bulk liquid</td>
</tr>
<tr>
<td>qud</td>
<td>Btu/sec</td>
<td>heat-transfer rate between ullage and droplet</td>
</tr>
<tr>
<td>qus</td>
<td>Btu/sec</td>
<td>heat-transfer rate between ullage and (unsubmerged) spray bar</td>
</tr>
<tr>
<td>qls</td>
<td>Btu/sec</td>
<td>heat-transfer rate between bulk liquid and (submerged) spray bar</td>
</tr>
<tr>
<td>npump</td>
<td>rpm</td>
<td>pump speed</td>
</tr>
<tr>
<td>dppump</td>
<td>psid</td>
<td>pump pressure rise</td>
</tr>
</tbody>
</table>
3.4 Program Listing

3.4.1 Heat Exchanger Model

* Zero G Venting System Integrated Steady State Heat Exchanger Performance Program
* By Tibor Lak, David Soo Hoo, & Dr. Han Nguyen
* A HP-9000 Program adapted from the Shuttle Venting Program, Rev. 3
* Includes single & two phase flow heat transfer & pressure losses.
* May 8, 1992

* INITIAL CONDITIONS:
* MASSIC - MASS IN THE TANK OR MANIFOLD TO BE VENTED (LBM)
* PIC - SATURATED PRESSURE OF THE TANK OR MANIFOLD (PSIA)
* PSTI - INITIAL GUESS AT THE INLET PRESSURE DURING BOILING (PSIA)
* MDOT - INITIAL GUESS AT THE VENT OR LEAK FLOWRATE (LB/SEC)

* FLUID PROPERTIES:
* Prop - Type of Propellant (1.0 = Hydrogen, 2.0 = Oxygen)
* PTP - TRIPLE POINT PRESSURE (PSIA)
* AVLIG - SONIC VELOCITY IN LIQUID (FT/SEC)

* EXTERNAL CONDITIONS:
* PAMB - AMBIENT PRESSURE (PSIA)
* G - GRAVITY

* INTERNAL CONDITIONS & CONFIGURATIONS:
* M - NUMBER OF NODES IN THE VENT PATH (Maxium is 20 nodes)
* VT - TANK OR MANIFOLD VOLUME (CUBIC IN)
* AX - EXIT AREA (SQ IN)
* A(1-M) - FLOW AREA OF THE VENT PATH PER NODE (SQ IN)
* K(1-M) - FLOW LOSS COEFFICIENT PER NODE
* DH(1-M) - CHANGE IN HEIGHT BETWEEN NODES (IN)
* QDOT - HEAT FLUX INTO THE FLUID PER NODE (BTU/SEC)
* (1-M)
* SAREA - SURFACE AREA OF THE NODE: USED TO CALCULATE THE
* (1-M) HEAT TRANSFER COEFFICIENT
* LENGTH - LENGTH OF NODES (IN)
* (1-M)
*
* VARIABLE EXIT AREA VARIABLES:
* VA  - Change in area (1 = yes, 0 = no)
* G1  - Acceleration in number of gravities.
* DIAM - Line diameter (in)
* SMAX - Maximum exposed surface area (sq inches)
*
* PROGRAM CONTROL VARIABLES:
* FINTIM - MAXIMUM RUN TIME (SEC)
* PRDEL - PRINT INTERVAL (SEC)
* QDTERR - QDOT ITERATION ERROR
* Delptp - Delta Exit Pressure From the Triple Point Pressure
* DELT - TIME INTERVAL BETWEEN ITERATIONS (SEC)
* OT  - option to plot time vs various parameters (0 = no plot)
* OM  - option to plot mass vs various parameters (0 = no plot)
* OP  - option to plot pressure vs various parameters (0 = no plot)
* DEBUG - option to print different information between iterations
* [0.0 = no debug] [1.0 = results from mdot loop]
* [2.0 = results from PST loop] [3.0 = results from PES loop]
* [4.0 = results from PSFO loop] [5.0 = results from PSFI loop]
* [6.0 = results from Liq PSFO loop] [7.0 = results from PXS loop]
* [8.0 = results from gas calc. loops]
* [9.0 = results from all 1 thru 7 loops]
*
  double precision pic,psti,ptp,avliq,pamb,vt,emnx,pemnx
  double precision delt,delptp,ys,sic,time,rhot,tsps,tsps
  double precision pst,pstl,slt,sgt,rhotl,hg,sgt,rhotv,ptsc,ts1
  double precision berr,ps,pliq,tliq,hls,hgs,mdot,pl,p,ph
  double precision rhole,rhoge,hle,hge,sle,sge,he,qdterr
  double precision rholo,rhogo,hlfo,hgfo,slfo,sgfo,hfo,hfop,dpfo
  double precision htot,pfo,psfo,visgfo,vislfo,asvo,drhdtl,to,qerr
  double precision rholi,rhogi,hlfi,hgfi,slfi,sgfi,hfi,dpfi
  double precision xf,psti,visgfi,vislfi,slf,psxf,fr,fp,rpogx
  double precision dptf,dpmf,perr,del,pxs,pxs,pxs,pxs,pxs,pxs,thrust
  double precision pes(2),tes(2),htote(2),pde(2),rho(2),hs(2)
* double precision RHOLI(45,2),RHOLO(45,2),RHOGI(45,2),RHOGO(45,2)
Define Heat Transfer Variables

real length(45),sarea(45,2)
double precision p13(50),surf(50),q(45,2),cpl(50),thkl(50),
1 stemp(50),cp,ts,st,cond

Define Plot Variables

real ptitle(18,4)
real subhd(18),name1(10),name2(10),name3(10),name4(10)
real name6(10),name7(10),name8(10),name9(10),name10(10)
real name11(10),name12(10),name5(10)
common /plotcom/ misc(3),nc,miss(13),dclim,ltick,nfig,nptmin,
1 nlines,nchlin,ptitle

data name/'mdot','isp','thrust','qerr'
1 , 'rhot', 'pst', 'pes', 'htot'
2 , 'ext area','twall '

Read Input Data

read (5,100) lable
read (5,101) massic,pic,psti,mdot(1),pback
read (5,100) lable
read (5,101) prop,ptp,avliq,pamb,g
read (5,100) label
read (5,102) m,vt,ax,va,tw
read (5,100) label
read (5,*) mdot(2),disp,dth,vtdi,ed
read (5,100) label
read (5,101) dpinc,errmx,perrmx,debug,exdi
read (5,100) label
read (5,101) fintim,prdel,qdterr,delt,delptp
read (5,100) label
read (5,104) ot,om,op,dq
read (5,100) label
do i=l,m
   read (5,105) a(i,1),dh(i,1),qdot(i,1),length(i)
enddo
read (5,106) subhd
read (5,107) subt
100 format (f1,a1)
101 format (5f10.0)
102 format (i10,2f10.0,i10,f10.0)
103 format (4f10.0,e10.1)
104 format (3i10,f10.0)
105 format (4f10.0)
106 format (/,18a4)
107 format (10a8)
*
   Write Input Data
*
if (debug .eq. 1.0 .or. debug .ge. 10.0) then
   write (6,101) massic,pic,psti,mdot(1),pback
   write (6,101) prop,ptp,avliq,pamb,g
   write (6,102) m,vt,ax,va,tw
   write (6,*) mdot(2),disp,dth,vtdi,ed
   write (6,101) dpinc,errmx,perrmx,debug,exdi
   write (6,101) fintim,prdel,qdterr,delt,delptp
   write (6,104) ot,om,op,dq
   do i=l,m
      write (6,105) a(i,1),dh(i,1),qdot(i,1),length(i)
   enddo
endif
*
   read tables of data file misc.data
*
if (prop.eq.1.0) then
   open (unit=2, file='h2misc.data',status='unknown')
else
open (unit=2, file='o2misc.data',status='unknown')
endif
read (2,150) n1
do i=1,n1
    read (2,155) xl(i),lmxpar(i)
enddo
* read (2,150) n2
do i=1,n2
    read (2,155) p2(i),lvisp(i)
enddo
* read (2,150) n3
do i=1,n3
    read (2,155) p3(i),gvisp(i)
enddo
* if (prop .ne. 1.0) then
    read (2,150) n12
    do i=1,n12
        read (2,155) p12(i),drhdto(i)
    enddo
endif
* read tables of data file rho.data
* if (prop .eq. 1.0) then
    open (unit=3, file='h2rho.data',status='unknown')
else
    open (unit=3, file='o2rho.data',status='unknown')
endif
read (3,150) n4
do i=1,n4
    read (3,155) p4(i),lrhop(i)
enddo
* read (3,150) n5
do i=1,n5
    read (3,155) p5(i),grhop(i)
enddo
* read (3,150) n6
do i=1,n6
    read (3,155) rhovp(i),p6(i)
enddo
* read (3,151) n7,maxt
  do i=1,n7
    read (3,*) p7(i),num(i)
    do j=1,num(i)
      read (3,155) t7(i,j),sclent(i,j)
    enddo
  enddo
* read tables of data file enthalpy & entropy data
  if (prop.eq.1.0) then
    open (unit=4, file="h2ent.data",status='unknown')
  else
    open (unit=4, file="o2ent.data",status='unknown')
  endif
  read (4,150) n8
  do i=1,n8
    read (4,155) p8(i),lentp(i)
  enddo
* read (4,150) n9
  do i=1,n9
    read (4,155) p9(i),gentp(i)
  enddo
* read (4,150) n10
  do i=1,n10
    read (4,155) p10(i),lent(i)
  enddo
* read (4,150) n11
  do i=1,n11
    read (4,155) p11(i),gent(i)
  enddo
* read tables of data file thermal conductivity & surface tension data
  if (prop.eq.1.0) then
    open (unit=10, file="h2thermo.data",status='unknown')
  * read (10,150) n13
  do i=1,n13
    read (10,*) p13(i),cpl(i),thkl(i),surft(i),stemp(i)
  enddo
* read (10,154) n14,num1,tcnst
  do i=1,n14
    read (10,152) p14(i)
    do j=1,num1
      read (10,153) cpg(i,j),thkg(i,j),viscg(i,j)
    enddo
  enddo
  do j=1,num1
    read (10,152) t14(j)
  enddo
endif
150 format (/,i3)
151 format (/,i3,4x,i3)
152 format (f10.2)
153 format (f14.0,2e14.3)
154 format (/,i3,4x,i3,4x,f10.0)
155 format (2f14.0)
*
* Define Program Constants & working variables
*
y = 0.0
AI = A(M,1)
AE = A(1,1)
tsp = PTP
K(M,1) = 0.5
K(1,1) = 0.5
PES(1) = PTP
PES(2) = PTP
m1 = m - 1
mass = massic
 call value(n10,p10,lent,pic,sic)
beta = 1.0
gamma = 1.0
time = 0.0
m8 = m * 0.8
* pttime = -pttdel
prtime = -prdel
ncount1 = 1
ncount2 = 1
exit = 0
pyi = 3.14159
*
do ij = 1, m
  twall(ij) = tw
\[ a(ij,2) = p_yi * \text{disp} ** 2.0 / 4.0 \]
\[ \text{sarea}(ij,1) = p_yi * (\text{disp} + 2.0 * \text{dth}) * \text{length}(ij) \]
\[ \text{sarea}(ij,2) = p_yi * \text{disp} * \text{length}(ij) \]
\[ \text{qdot}(ij,2) = -\text{qdot}(ij,1) \]

**Enddo**

* Define Plot Constants

\[ \text{nc} = 0 \]
\[ \text{nfig} = 0 \]
\[ \text{nlines} = 0 \]

* Check to Determine the Correct Time increment is Used

* \[ \text{IF} (\text{delt} > \text{pltdel}) \text{ delt} = \text{pltdel} \]
* \[ \text{IF} (\text{delt} > \text{prdel}) \text{ delt} = \text{prdel} \]

* CALCULATE INITIAL TANK OR MANIFOLD CONDITIONS

1 \[ \text{RHOT} = \text{MASS}/\text{VT} \]
\[ \text{tagl} = 0.0 \]
\[ \text{tagh} = 0.0 \]
\[ \text{mlow} = 0.0 \]
\[ \text{mhigh} = 0.0 \]
\[ \text{pst} = P_{STI} \]
\[ \text{pstyl} = 0.0 \]
\[ j1 = 0 \]

* Loop to Calculate Initial Pst

5 \[ \text{call value}(n10,p10,\text{lent},PST,\text{slt}) \]
\[ \text{call value}(n11,p11,\text{gent},PST,\text{sgt}) \]
\[ \text{call value}(n4,p4,\text{lrhop},PST,\text{rholt}) \]
\[ \text{call value}(n5,p5,\text{grhop},PST,\text{rhoft}) \]
\[ \text{YT} = (\text{SIC-SLT})/(\text{SGT-SLT}) \]
\[ \text{if (yt < 0.0) yt = 0.0} \]
\[ \text{if (yt > 1.0) yt = 1.0} \]
\[ \text{RHOtv} = \frac{\text{YT}(1.0/\text{RHOT}-1.0)}{\text{RHOtv}(1.0-\text{YT})} \]
\[ \text{call value}(n6,\text{rhovp},p6,\text{RHOtv},\text{ptsc}) \]
\[ \text{if (j1 .eq. 0) pstaw = ptsc} \]
\[ \text{beta = dabs}(pstaw - \text{pstaw}) \]
\[ \text{pstaw} = \text{pstaw} \]
\[ \text{berr} = \text{ptsc} - \text{pstaw} \]
\[ \text{IF (dabs(berr) .gt. 0.001) then} \]
\[ \text{IF (berr .gt. 0.0) then} \]
```fortran
C      pst = pst + beta/2.0
C      else
C      pst = pst - beta/2.0
C      endif
C
jl=jl+l
if (jl.gt.100) go to 6
go to 5
endif
PS=PST
IF (debug .eq. 2.0 .or. debug .ge. 10.0) THEN
write (6,110) j1,pst,slt,sgt,rholt,rhogt,ptsc,mass,yt
110 format ('counter = ',i4,4x,'PST = ',f8.3,/7(3x,f8.3))
ENDIF

PSTI=PST
PLIQ=PST
if (prop.eq.1.0) then
  call H2SAT(PST,tliq)
else
  call o2sat(PST,tliq)
endif
YS=YT

SATURATED LIQUID AT SOURCE

IF(PLIQ.le.PST) then
  call value(n8,p8,lentp,PST,hls)
call value(n9,p9,gentp,PST,hgs)
  HS(1) = (1.0-YS)*HLS+YS*HGS
  HS(2) = HS(1)
else

SUBCOOLED LIQUID AT SOURCE

YS=0.0
call value3(np7,maxt,num,p7,t7,sclent,pliq,tliq,hls)
  HS(1) = HLS
  HS(2) = HS(1)
endif

Set Constants for Mdot Calculation

tsp = PTP
```

III-20
START ITERATION ON THE FLOW RATE CONVERGENCE BASED ON CHoke FLOW AT THE ORIFICE BY THE OUTLET OF THE TANK FOR THE TVS SYSTEM

THETA = 1.05
DO L1 = 1, 1000
  if (tagl .ne. 0.0 .and. tagh .ne. 0.0) then
    mdot(1) = mlow + dm * (pl / (pl - ph))
  else
    mdot(1) = (theta + 1.0) * mdot(1)/2.0
  endif

DEFINITION OF TANK EXIT CONDITION (SONIC/SUBSONIC FLOW) (CHoke FLOW AT THE TANK EXIT NODE M)

HTOTE(1) = HS(1) + QDOT(M,1)/MDOT(1) - G*DH(M,1)/(12.0*1728.0)
AT = A(M,1)
seold = 0.0
xpes = 0.9
tsp = xpes * tsp
do i = 1,400
tsp = tsp + dpinc
IF(tsp.GT.PST) GO TO 15
IF(tsp.LT.PTP) tsp = ptp
call value(n4,p4,lrhop,tsp,rhole)
call value(n5,p5,grhop,tsp,rhoge)
call value(n8,p8,lentp,tsp,hle)
call value(n9,p9,gentp,tsp,hge)
call value(n10,p10,lent,tsp,sle)
call value(n11,p11,gent,tsp,sge)
call TPHS(HTOTE(1),RHOLE,RHOGE,MDOT(1),AT,HLE,HGE,SLE,1,seold,ye(1),he,se,rhoe(1),pde(1))
if (se .le. seold) goto 15
seold = se
enddo
15 continue
if (prop.eq.1.0) then
call H2SAT(tsp,tts)
else
call o2sat(tsp,tts)
endif

PO(M,1) = tsp + PDE(1)
YO(M,1) = YE(1)
PRESSURE LOSS CALCULATION THROUGH NODE M

KF = K(M,1)
AREA = A(M,1)

CHANGE IN ELEVATION

DHI = DH(M,1)
DHO = DH(M+1,1)
HO(M,1) = HS(1) + QDOT(M,1)/MDOT(1) - G*DH(M,1)/(12.0*1728.0)
HTOT = HO(M,1)

TWO-PHASE FLOW REGION

PFO = PO(M,1)
PSFO = PFO

DETERMINE THE OUTLET CONDITIONS AT NODE N, GIVEN THE TOTAL PRESSURE AT THE OUTLET

DO I1 = 1,5
   IF(PSFO.LT.PTP) PSFO = PTP
   IF(PSFO.GT.PST) PSFO = PST
   call value(n4,p4,lrhop,PSFO,rholfo)
   call value(n5,p5,grhop,PSFO,rhogfo)
   call value(n8,p8,lentp,PSFO,hlfo)
   call value(n9,p9,gentp,PSFO,hgfo)
   call value(n10,p10,lent,PSFO,slfo)
   call value(n11,p11,gent,PSFO,sgfo)
   call value(n3,p3,gvisp,PSFO,visgfo)
   call value(n2,p2,lvisp,PSFO,vislfo)
   call TPHS(HTOT,RHOLFO,RHOGFO,MDOT(1),AREA,HLFO,HGFO,SLFO,
   sgfo,yfo,hfo,sfo,rhofo,pdfo)
   PSFO = PFO-PDFO
enddo

End of I1 Loop (To Determine Node M Outlet Conditions)

if (prop.eq.1.0) then
call H2SAT(PSfo,ts1)
else
call o2sat(PSfo,ts1)
endif
PSO(M,1)=PSFO
PDO(M,1)=PDFO
RHOLO(M,1)=RHOLFO
RHOGO(M,1)=RHOGFO
RHOO(M,1)=RHOFO
YO(M,1)=YFO
HO(M,1)=HFO
SO(M,1)=SFO
VO(M,1)=144.0*MDOT(1)/(A(M,1)*RHOO(M,1))
TSO(M,1)=ts1
if (prop.le.1.0) then
call H2SVEL(YFO,PSFO,RHOLFO,RHOGFO,asvo)
else
call value(n12,p12,drhdto,psfo,drhdt1)
call o2sat(psfo,to)
call o2svel(yfo,to,rholfo,rhogfo,drhdt1,asvo)
endif
MACHO(M,1)=VO(M,1)/ASVO
IF(MACHO(M,1).LT.0.0) MACHO(M,1)=0.0
IF(MACHO(M,1).GT.1.0) MACHO(M,1)=1.0

* INITIAL GUESS AT NODE INLET BASED ON COMPRESSIBLE LOSS *
call IN1TL(PSFO,KF,MDOT(1),AREA,RHOLFO,psfi)

* TOTAL ENTHALPY AT NODE INLET *
HI(M,1)=HS(1)+QDOT(M+1,1)/MDOT(1)-G*DH(M,1)/(12.0*1728.0)
HTOT=HI(M,1)

* DEFINITION OF STATIC PRESSURE AT NODE INLET *
DO I2 = 1,50
IF(PSFLLT.PTP) PSFI=PTP
IF(G.LT.0.0.AND.PSFLLT.PSFO) PSFI=PSFO
IF(PSFLGT.PST) PSFI=PST
call value(n4,p4,lhop,PSFI,rholfi)
call value(n5,p5,grhop,PSFI,rhogfi)
call value(n8,p8,1entp,PSFI,hlfi)
call value(n9,p9,gentp,PSFI,hgfi)
call value(n10,p10,1ent,PSFI,slfi)
call value(n11,p11,gent,PSFI,sgfi)
call value(n3,p3,gvsp,PSFI,visgfi)
call value(n2,p2,1visp,PSFI,vislfi)
call TPHS(HTOT,RHOLFI,RHOGFI,MDOT(1),AREA,HLFI, HGFI,SLFI,
* LOCKHART-MARTINELLI TWO-PHASE FLOW PARAMETER
* 
call XPARAM(YFO, VISLFO, VISGFO, RHOLFO, RHOGFO, YFI, VISLFI, VISGFI, RHOFLI, RHOGFLI, XFI)
  IF(XF.LT.0.01) XF=0.01
  LGXF=LOG10(XF)
  call value(n1, x1, lmxpar, LGXF, phisqf)
  PHISQF=(10.0**PHISQF)**2
* 
* TWO-PHASE FLOW PRESSURE LOSS AND NODE INLET PRESSURE (PSFN)
* 
call TPS(AREA, G, DHI, DHO, KF, PSFO, YFO, RHOFL, RHOGFL, YFI, 
  rholfi, RHOGFLI, MDOT(1), PHISQF, dpff, psfn, dptf, 
  dpmf)
  ERR(M,1)=ABS(PSFN-PSFI)
  PSFI=(PSFI+PSFN)/2.0
  
  * CONDITION WHERE THE INLET STATIC PRESSURE OF THE NODE IS DETERMINED
  * 
  IF(ERR(M,1).LE.ERRMX) GO TO 20
  * 
  End of I2 Loop (To Determine Node N Inlet Conditions)
  *
  enddo
20 PFI=PSFI+PDFI
* 
* CALCULATE THE INLET CONDITIONS OF THE NODE M
*
  RHOLI(M,1)=RHOFLI
  RHOGI(M,1)=RHOGFLI
  RHOI(M,1)=RHOFLI
  DPM(M,1)=DPMF
  DPF(M,1)=DPFF
  DPT(M,1)=DPTF
  PI(M,1)=PFI
  PDI(M,1)=PDFI
  PSI(M,1)=PSFI
  YI(M,1)=YFI
  HI(M,1)=HFI
  SI(M,1)=SFI
  PHISQ(M,1)=PHISQF
VI(M,1)=144.0*MDOT(1)/(A(M,1)*RHOI(M,1))
  if (prop.le.1.0) then
    call H2SVEL(YFI,PSFI,RHOLFI,RHOGFI,asvi)
  else
    call value(n12,p12,drhdt0,psfi,drhdt2)
    call o2sat(psfi,ti)
    call o2svel(yfi,ti,rholfi,rhogfi,drhdt2,asvi)
  endif
MACHI(M,1)=VI(M,1)/ASVI
IF(MACHI(M,1).LT.0.0) MACHI(M,1)=0.0
IF(MACHI(M,1).GT.1.0) MACHI(M,1)=1.0
*
* CALCULATE THE TWO PHASE HEAT TRANSFER COEFFICIENT FOR NODE M
* BASED ON INLET PRESSURE
*
  call value(n13,p13,cpl,psfi,cp)
  call value(n13,p13,thkl,psfi,cond)
  call value(n13,p13,surft,psfi,st)
  call value(n13,p13,stemp,psfi,ts)
  if (tave(m,1).le.0.0) then
    tave(m,1) = ts
  endif
  hgl = hgfi - hlfi
  *
  dtemp = twall(m) - tave(m,1)
  *
  di = exdi
  *
  dpress = 778.2 * dtemp * rhogfi * hgl / ts
  if (yfi .gt. 0.7) then
    re = 48.0 * mdot(1) * (1.0-0.7)/(pyi * vislfi * di)
  else
    re = 48.0 * mdot(1) * (1.0-yfi)/(pyi * vislfi * di)
  endif
  *
  call hteoeff(phase,dtemp,dpress,cp,cond,vislfi,rholfi,1
              rhogfi,st,hgl,re,xf,di,qdott)
  if (yfi .ge. 0.7) then
    tdim = 0.0
    call value2(n14,num1,p14,t14,cpg,psfi,tdim,cp)
    call value2(n14,num1,p14,t14,thkg,psfi,tdim,cond)
    call value2(n14,num1,p14,t14,viscg,psfi,tdim,visgfi)
    re = 48.0 * mdot(1) /(pyi * visgfi * di)
    call hteoeff(3.0,dtemp,dpress,cp,cond,visgfi,rholfi,1
                rhogfi,st,hgl,re,xf,di,qtt)
\[
q_{dott} = q_{dott}^* (1- y_{fi}) + q_{tt} y_{fi}
\]

\[
q(m,1) = q_{dott} \times (t_{wall}(m) - t_{ave}(m,1)) \times sarea(m,1)/144.0
\]

\[
TSI(M,1) = TS
\]

\[
XTT(M,1) = XF
\]

\[
RN(M,1) = RE
\]

\[
QDTI'(M,1) = QDOTI
\]

\[
TAVE(M,1) = (t_{si}(m,1) + t_{so}(m,1))/2.0
\]

\[
\theta = ps/\pi(m,1)
\]

\[
perr = ps-pi(m,1)
\]

\[
\text{if (perr .lt. 0.0) then}
\]

\[
del = m_{low} - m_{dot}(1),
\]

\[
tagl = 1.0
\]

\[
mlow = m_{dot}(1)
\]

\[
pl = perr
\]

\[
\text{else}
\]

\[
del = m_{high} - m_{dot}(1)
\]

\[
tagh = 1.0
\]

\[
mhigh = m_{dot}(1)
\]

\[
ph = perr
\]

\[
\text{endif}
\]

\[
dm = m_{high} - m_{low}
\]

\[
\text{if (dabs(perr) .le. perfmx .or. dabs(del) .le. 0.0001) goto 30}
\]

\[
\text{endif}
\]

\[
\text{----- END OF MDOT CALCULATION LOOP ---}
\]

\[
\text{--- END OF MDOT CALCULATION LOOP ---}
\]

\[
DO L=1,150
\]

\[
\text{if (L .gt. 1) then}
\]

\[
do ii = 1, m1
\]

\[
\text{--- END OF MDOT CALCULATION LOOP ---}
\]
nn = m - ii
qdot(nn,1) = qdot(nn+1,1) + q(nn,1)
qdot(nn,2) = qdot(nn+1,2) - q(nn,2)
if (sarea(ii,1) .le. 0.0 .or. qdtt(ii,1) .lt. 0.0) then
  c1 = 1.0
else
  c1 = sarea(ii,2) * qdtt(ii,2)/(sarea(ii,1)*qdtt(ii,1))
endif
if (dabs(delta(ii)) .gt. dq) then
  tw = twall(ii)
  twall(ii) = (c1 * tave(ii,2) + tave(ii,1))/(1+c1)
  twa11(ii) = (tw + twall(ii))/2.0
  if(twall(ii) .lt. 24.845) twall(ii) = 24.845
endif
endo
dif
endif

qdt = 0.0
qdtl = 0.0

c------------------------------------------------------------------------
c START THE VENT & SPRAY NODAL NETWORK FLOW PROPETIES
CALCULATION

1

* Definitions of exit condition (sonic/subsonic flow)
* (choke flow at the exit node)
* HTOTE(il)=HS(il)+QDOT(1,il)/MDOT(il)+G*(DH(M+1,il)-DH(1,il))/
  (12.0*1728.0)
 AE = A(1,il)
 seold = 0.0
 xpes = 0.9
 PES(il) = xpes * PES(il)
 do i = 1,500
   if (il .eq. 1) then
     PES(il) = PES(il) + dpinc
   else
     PES(il) = pback
   endif
IF(PES(il).GT.PST) GO TO 40
IF(PES(il).LT.PTP) PES(il)=PTP
call value(n4,p4,lrhop,PES(il),rhole)
call value(n5,p5,grhop,PES(il),rhoge)
call value(n8,p8,lentp,PES(il),hle)
call value(n9,p9,gentp,PES(il),hge)

III-27
call value(n10,p10,lent,PES(il),sle)
call value(n11,p11,gent,PES(il),sge)
call TPHS(HTOTE(il),RHOE,RHOGE,MDOT(il),AE,HLE,HGE,SLE,
   sge,ye(il),he,se,rhoe(il),pde(il))
   if (se .le. seold) goto 40
   seold = se
   enddo

40 continue

* IF (debug .eq. 3.0 .or. debug .ge. 10.0) THEN
  write (6,111) i,pes(il),mdot(il),ye(il),pde(il),sle,sge,
     rhoIe,rhoge,se,seold
  111 format ('counter = ',i4,4x,'PES = ',f8.3,4x,'MDOT = '
     f10.5,2(f8.3,3x),/',6(3x,f5.3))
  ENDIF
* if (prop.eq.1.0) then
  call H2SAT(PES(il),tes(il))
else
  call o2sat(PES(il),tes(il))
endif

* PO(I,il)=PES(il)+PDE(il)
  YO(1,il)=YE(il)

+++++
c PRESSURE LOSS CALCULATION THROUGH EACH NODE +
c (NODE 1 IS AT THE EXIT & NODE M IS AT THE INLET) +
+++++
DO N=I,M
   if(il .eq. 1 .and. n .eq. m) goto 60
   IF(N.GT.1) PO(N,il)=PI(N-1,il)
   IF(N.GT.1) YO(N,il)=YI(N-1,il)
   IF(N.GT.1) TSO(N,il)=TSI(N-1,il)
   AREA=A(N,il)
* if (N .eq. 1 .and. PSFO .le. 0.0) psfo = po(n,il)
call value(n2,p2,lvisp,PSFO,vislfo)
   visc = vislfo
if (il .eq. 1) then
   if (n .ne. 1) then
      dd = vtdi / 12.0
      call frict(dd,ed,mdot(1),visc,ff)
      K(n,1) = ff * length(n)/vtdi
   enddo
else
    K(1,1) = 0.5
endif
else
    dd = disp / 12.0
    call frict(dd, ed, mdot(2), visc, ff)
    K(n,2) = ff * length(n)/disp
endif
KF=K(N,il)

* CHANGE IN ELEVATION *

DHI=DH(N,il)
DHO=DH(N+1,il)
HO(N,il)=HS(il)+QDOT(N,il)/MDOT(il)+G*(DH(M+1,il)-DH(N,il))/(12.0*1728.0)
HTOT=HO(N,il)
IF(YO(N,il),LE.0.0) PHASE=1.0
IF(YO(N,il),GT.0.0) PHASE=2.0

* DETERMINE THE PHASE OF THE FLUID *

IF(PHASE.ge.2.0) then
  * TWO-PHASE FLOW REGION
  PFO=PO(N,il)
  PSFO=PFO
  *
  DETERMINE THE OUTLET CONDITIONS AT NODE N, GIVEN THE TOTAL PRESSURE AT THE OUTLET *
  *
  DO I1 = 1,5
  IF(PSFO.LT.PTP) PSFO=PTP
  IF(PSFO.GT.PST) PSFO=PST
  call value(n4,p4,lrhop,PSFO,rholfo)
  call value(n5,p5,grhop,PSFO,rhogfo)
  call value(n8,p8,1entp,PSFO,hlfo)
  call value(n9,p9,gentp,PSFO,hgfo)
  call value(n10,p10,1ent,PSFO,slfo)
  call value(n11,p11,gent,PSFO,sgfo)
  call value(n3,p3,gvisp,PSFO,visgfo)
  call value(n2,p2,lvisp,PSFO,vislfo)
  call TPHS(HTOT,RHOLFO,RHOGFO,MDOT(il),AREA,HLFO,HGFO,
                     slfo,sgfo,yfo,hfo,sfo,rohfo,pdfo)
PSFO=PFO-PDFO

* End of I1 Loop (To Determine Node N Outlet Conditions)

enddo

* IF (debug .eq. 4.0 .or. debug .ge. 10.0) THEN
  write (6,112) n,il,psfo,slfo,sgfo,rhofo,yfo,sfo,hfo,pdfo
  format ('node #',i2,4x,'counter = ',i4,4x,'PSFO = ', f8.3,7(3x,f8.3))
ENDIF

PSO(N,il)=PSFO
PDO(N,il)=PDFO
RHOLO(N,il)=RHOLFO
RHOGO(N,il)=RHOGFO
RHOO(N,il)=RHOFO
YO(N,il)=YFO
HO(N,il)=HFO
SO(N,il)=SFO
VO(N,il)=144.0*MDOT(il)/(A(N,il)*RHOO(N,il))

* calculate the outlet temperature at node 1

* if (n .eq. 1) then
  call H2SAT(psfo,tsl)
  call value(n13,p13,cpl,psfo,cp)
  hgl = hgfo - hlfo
  qq = qdot(1,il)/mdot(il)
  if (yo(1,il) .gt. 0.99) then
    if (il .eq. 1) then
      tso(1,1) = tsi(m,1) + (qq - hgl)/cp
    else
      tso(1,2) = tliq + (qq + hgl)/cp
    endif
  else
    tso(1,2) = tliq + (qq + hgl)/cp
  endif
else
  tso(1,il) = ts1
endif
endif
endif

* if (prop.le.1.0) then
  call H2SVEL(YFO,PSFO,RHOLFO,RHOGFO,asvo)
else
  call value(n12,p12,drhdt0,psfo,drhdt1)
  call o2sat(psfo,tsl)
endif
call o2svel(yfo, to, rholfo, rhogfo, drhdt1, asvo)
endif

MACHO(N, il) = VO(N, il)/ASVO
IF(MACHO(N, il).LT.0.0) MACHO(N, il)=0.0
IF(MACHO(N, il).GT.1.0) MACHO(N, il)=1.0

* INITIAL GUESS AT NODE INLET BASED ON COMPRESSIBLE LOSS
* call INITL(PSFO,KF,MDOT(il),AREA,RHOLFO,psfi)
*
* TOTAL ENTHALPY AT NODE INLET
* HI(N,il)=HS (il)+QDOT(N+1,il)/MDOT(il)+
1 G*(DH(M,il)-DH(N+1,il))/(12.0*1728.0)
HTOT=HI(N,il)
*
* DEFINITION OF STATIC PRESSURE AT NODE INLET
* DO I2 = 1,50
    IF(PSFI.LT.PTP) PSFI=PTP
    IF(G.LE.0.0.AND.PSFI.LT.PSFO) PSFI=PSFO
    IF(PSFI.GT.PST) PSFI=PST
    call value(n4,p4,lrhop,PSFI,rholfoi)
    call value(n5,p5,grhop,PSFI,rhogfoi)
    call value(n8,p8,lentp,PSFI,hlfi)
    call value(n9,p9,gentp,PSFI,hgfi)
    call value(n10,p10,lent,PSFI,slfi)
    call value(n11,p11,gent,PSFI,sgfi)
    call value(n3,p3,gvisp,PSFI,visgfoi)
    call value(n2,p2,lvisp,PSFI,vislfoi)
    call TPHS(HTOT,RHOLFI,RHOGFI,MDOT(il),AREA,HLFI,HGFI,
    slfi,sgfi,yfi,hlfi,slfi,rhofi,pdfi)
    PFI=PSFI+PDFI
*
* LOCKHART-MARTINELLI TWO-PHASE FLOW PARAMETER
* call XPARAM(YFO,VISLFO, VISGFO,RHOLFO, RHOGFO, YFI, VISLFI,
1 visgfoi,RHOLFI,RHOGFO,xf)
    IF(XF.LT.0.01) XF=0.01
    LGXF=LOG10(XF)
    call value(n1,x1,lmxpar, LGXF,phisqf)
    PHISQF=(10.0**PHISQF)**2
*
* TWO-PHASE FLOW PRESSURE LOSS AND NODE INLET PRESSURE (PSFN)
call TPS(AREA,G,DHI,DHO,KF,PSFO,YFO,RHOLFO,RHOGFO,YFI,
    rholfi,RHOGFI,MDOT(il),PHISQF,dpff,psfn,dptf,
    dpmf)
ERR(N,il)=ABS(PSFN-PSFI)
PSFI=(PSFI+PSFN)/2.0

* CONDITION WHERE THE INLET STATIC PRESSURE OF THE NODE IS DETERMINED
*
IF(ERR(N,il).LE.ERRMX) GO TO 50
*
End of I2 Loop (To Determine Node N Inlet Conditions)
*
enddo
50 PFI=PSFI+PDFI
*
IF (debug .eq. 5.0 .or. debug .ge. 10.0) THEN
    write (6,113) n,i2,psfi,slfi,sgfi,rhofi,yfi,sfi,hfi,
    psufqf
113 format ('node #',i2,4x,'counter = ',i4,4x,'PSFI = ',
    f8.3,J,6(3x,f8.3),/3x,f8.3) ENDIF
*
CALCULATE THE INLET CONDITIONS OF THE NODE N
*
RHOLI(N,il)=RHOLFI
RHOGI(N,il)=RHOGFI
RHOI(N,il)=RHOFI
DPM(N,il)=DPMF
DPF(N,il)=DPFF
DPT(N,il)=DPTF
PI(N,il)=PFI
PDI(N,il)=PDFI
PSI(N,il)=PSFI
YI(N,il)=YFI
HI(N,il)=HFI
SI(N,il)=SFI
PHISQ(N,il)=PHISQF
VI(N,il)=144.0*MDOT(il)/(A(N,il)*RHOI(N,il))
    if (prop.le.1.0) then
      call H2SVEL(YFI,PSFI,RHOLFI,RHOGFI,asvi)
    else
      call value(n12,p12,drhdto,psfi,drhd2)
      call o2sat(psfi,ti)
      call o2svel(yfi,ti,rholfi,rhogfi,drhd2,asvi)
    endif
MACI(N,il)=VI(N,il)/ASVI
IF(MACHI(N,il).LT.0.0) MACHI(N,il)=0.0
IF(MACHI(N,il).GT.1.0) MACI-II(N,il)=1.0

* CALCULATE THE TWO PHASE HEAT TRANSFER COEFFICIENT FOR NODE N
* BASED ON INLET PRESSURE
*
call value(n13,p13,cpl,psfi,cp)
call value(n13,p13,thkl,psfi,cond)
call value(n13,p13,surft,psfi,st)
call value(n13,p13,stemp,psfi,ts)
*
* do nn1 = 1, 2
* *
if (tave(n,il) .le. 0.0) then
  tave(n,il) = ts
endif
hgl = hgfi - hlfi
*
if (il .eq. 1) then
dtemp = twall(n) - tave(n,il)
  di = exdi
else
dtemp = tave(n,il) - twall(n)
  di = disp
endif
*
dpress = 778.2 * dtemp * rhogfi * hgl / ts
if (yfi .gt. 0.7) then
  re = 48.0 * mdot(il) * (1.0-0.7)/(pyi * vislfi * di)
else
  re = 48.0 * mdot(il) * (1.0-yfi)/(pyi * vislfi * di)
endif
write (6,*), dtemp,dpress,cp,cond,vislfi,rholfi,rhogfi,st,hgl,re,xf
c 1
*call htccoeff(phase,dtemp,dpress,cp,cond,vislfi,rholfi,rhogfi, st,hgl,re,xf,di,qdott)
 1
if (yfi .ge. 0.7) then
tdim = 0.0
call value2(n14,num1,p14,t14,cpg,psfi,tdim,cp)
call value2(n14,num1,p14,t14,thkg,psfi,tdim,cond)
call value2(n14,num1,p14,t14,viscg,psfi,tdim,visgfi)
  re = 48.0 * mdot(il) /(pyi * visgfi * di)
call htcoeff(3.0,dtemp,dpress,cp,cond,visgfi,rholfi,
     rhogfi,st,hgl,re,xf,di,qtt)
qdot = qdot*(1-yfi) + qtt*yfi
endif
if (il.eq.1) then
  q(n,il) = qdot * (twall(n) - tave(n,il)) * sarea(n,il)
  /144.0
else
  q(n,il) = qdot * (tave(n,il) - twall(n)) * sarea(n,il)
  /144.0
endif
* calculate the inlet temperature at node il
* call value(u13,p13,cpl,psfi,cp)
if (yi(n,il) .gt. 0.99) then
  if (il.eq.1) then
    tsi(n,il) = tso(n,il) - q(n,il)/(mdot(il) * cp)
  else
    tsi(n,il) = tso(n,il) + q(n,il)/(mdot(il) * cp)
  endif
else
  tsi(n,il) = ts
endif
XTI(N,il) = XF
RN(N,il) = RE
QDTI(N,il) = QDOT
TAVE(N,il) = (tsi(n,il) + tso(n,il))/2.0
* enddo
* if (debug .ge. 10.0) then
  write (6,120) tsi(n,il),twall(n),tave(n,il),re,qdott,
     q(n,il),sarea(n,il)
120 format(4(2x,f14.4),3(2x,f14.4))
endif
ELSE
* ...............................................................
* SINGLE PHASE FLOW (PHASE=1.0, LIQUID FLOW)
* ...............................................................
* CALCULATION OF OUTLET CONDITIONS GIVEN TOTAL PRESSURE AT NODE
* OUTLET (PO(N))
* PFO=PO(N,il)
PSFO=PFO
DO 13 = 1,5
  IF(PSFO.LT.PTP) PSFO=PTP
  IF(PSFO.GT.PST) PSFO=PST
  call value(n4,p4,lhop,PSFO,rholfo)
  PDFO=144.0*MDOT(il)/A(N,il)*MDOT(il)/A(N,il)/
                  (2.0*RHOLFO*32.2)
  PSFO=PFO-PDFO

End of 13 Loop (To Determine Node N Outlet Conditions)
enddo

IF(PSFO.LT.PTP) PSFO=PTP
IF(PSFO.GT.PST) PSFO=PST
call value(n8,p8,lentp,PSFO,hfo)
call value(n10,p10,lent,PSFO,sfo1)
call value(n13,p13,cpl,psfo,cp)
call value(n2,p2,lvisp,PSFO,vislfo)
if (n .eq. 1) then
  if (il .eq. 1) then
    tso(I,il) = tsi(m,il) + qdot(1,il)/(mdot(il) * cp)
  else
    tso(I,il) = tliq + qdot(1,il)/(mdot(il) * cp)
  endif
endif
PSO(N,il)=PSFO
PDO(N,il)=PDFO
RHOLO(N,il)=RHOLFO
RHOGO(N,il)=0.0
RHOO(N,il)=RHOLFO
YO(N,il)=0.0
HO(N,il)=HS(il)+QDOT(N,il)/MDOT(il)+G*
  (DH(M+1,il)-DH(N,il))/(12.0*1728.0)
SO(N,il)=SFO1
VO(N,il)=144.0*MDOT(il)/(A(N,il)*RHOO(N,il))
ASVO=AVLIQ
MACHO(N,il)=VO(N,il)/ASVO
IF(MACHO(N,il).LT.0.0) MACHO(N,il)=0.0
IF(MACHO(N,il).GT.1.0) MACHO(N,il)=1.0
IF (debug .eq. 6.0 .or. debug .ge. 10.0) THEN
  write (6,114) n,i3,psfo,pdfo,tso(n,il),rholfo,hfo,
                    so(n,il),vo(n,il),htote(il),tso(n,il)
  END
INLET CONDITION CALCULATIONS

\[
\text{RHOLI}(N,il) = \text{RHOLO}(N,il) \\
\text{RHOGI}(N,il) = 0.0 \\
\text{RHOI}(N,il) = \text{RHOLI}(N,il) \\
\text{DPM}(N,il) = 0.0 \\
\text{DPF}(N,il) = \frac{144.0 \times K(N,il)}{2.0 \times \text{RHOLI}(N,il)} \times \left(\frac{\text{MDOT}(il)}{A(N,il)}\right)^{2} \\
\text{DPI}(N,il) = \frac{144.0 \times (\text{MDOT}(il) / A(N,il))^{2}}{2.0 \times \text{RHOLI}(N,il)} \\
\text{PSI}(N,il) = \frac{\text{PI}(N,il) - \text{DPI}(N,il)}{\text{RHOLI}(N,il)} \\
\text{YI}(N,il) = 0.0 \\
\text{VI}(N,il) = \frac{144.0 \times \text{MDOT}(il)}{\text{RHOLI}(N,il) \times (A(N,il))} \\
\text{ASVI} = \text{AVLIQ} \\
\text{MACHI}(N,il) = \frac{\text{VI}(N,il)}{\text{ASVI}} \\
\text{PSFI} = \text{PSI}(N,il) \\
\text{IF(P} \text{SFI}<\text{PT}) \text{PSFI}=\text{PT} \text{P} \\
\text{IF(P} \text{SFI}>\text{PST}) \text{PSFI}=\text{PST} \\
\text{call value}(n_{2},p_{2},lvisp,\text{PSFI},vislfi) \\
\text{call value}(n_{8},p_{8},lentp,\text{PSFI},lhf) \\
\text{call value}(n_{10},p_{10},lentp,\text{PSFI},slf) \\
\text{HI}(N,il) = \text{HLF} \\
\text{HI}(N,il) = \text{HS}(il) + \text{QDOT}(N+1,il) / \text{MDOT}(il) + \frac{G \times (\text{DH}(M,il) - \text{DH}(N+1,il))}{12.0 \times 1728.0} \\
\text{SI}(N,il) = \text{SLF} \\
\text{PHISQ}(N,il) = 1.0 \\
\]

IF (debug .eq. 6.0 .or. debug .ge. 10.0) THEN
write (6, 214) n, psfi, pdi(n,il), dpm(n,il),
1 dphp(n,il), dpt(n,il), vi(n,il)
214 format ('Node # = ',i4,4x,'Liquid PSFI = ',f8.3/,
1 5(3x,f8.3))
ENDIF

CALCULATE THE HEAT TRANSFER COEFFICIENT FOR NODE N
BASED ON INLET PRESSURE

\[
\text{call value}(n_{13},p_{13},cpl,\text{PSFI},cp) 
\]
call value(n13,p13,thkl,psfi,cond)
call value(n13,p13,surft,psfi,st)
call value(n13,p13,stemp,psfi,ts)
if (tave(n,il) .le. 0.0) then
  tave(n,il) = tso(n,il)
endif

hgl = hgfi - hlfi

* 
if (il .eq. 1) then
  dtemp = twall(n) - tave(n,il)
  di = exdi
else
  dtemp = tave(n,il) - twall(n)
  di = disp
endif

* 
dpress = 778.2 * dtemp * rhogfi * hgl / ts
re = 48.0 * mdot(il)/(pyi * vislfi * di)
call htcoeff(phase,dtemp,dpress,cp,cond,vislfi,rholfi,rhogfi,ts,hgl,re,xf,di,qdott)
if (il.eq. 1) then
  q(n,il) = qdott * (twall(n) - tave(n,il)) * sarea(n,il)
  /144.0
else
  q(n,il) = qdott * (tave(n,il) - twall(n)) * sarea(n,il)
  /144.0
endif

* 
if (il .eq. 1) then
  tsi(n,il) = tso(n,il) - q(n,il)/(mdot(il) * cp)
else
  tsi(n,il) = tso(n,il) + q(n,il)/(mdot(il) * cp)
endif
XTr,i = XF
RN(N,il) = RE
QDTT(N,il) = QDOTT
TAVE(N,il) = (tsi(n,il) + tso(n,il))/2.0

* 
IF (debug .eq. 6.0 .or. debug .ge. 10.0) THEN
write (6,314) tsi(n,il),q(n,il),dtemp,xf,re,qdott,
1 hgl,cp,di,tave(n,il)
314 format ('Liquid TSI = ',f8.3,3x,f10.4,2(3x,f8.3),3x,
1 f12.2/,5(3x,f8.3))
endif

*
ENDIF
*    PO(N+1,il)=PI(N,il)
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++
c   End of N do loop (Conditions for Both the Inlet & Outlet of the +
c Nodal Network are Defined) +
c++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++
   60 continue
ENDDO
C....................................................................... .
C RE-CALCULATE THE NODAL PROPERTIES OF THE NODES THAT ARE 100% GAS
C....................................................................... .
if (il .eq. 1) then
   do ig = ml, 1, -1
      if (yo(ig,1) .gt. 0.99) goto 130
   enddo
130 phase = 3.0
   dummy = 1.4 * 766.55232 * 32.2
   do ia = ig, 1, -1
      if (yo(ia,1) .ge. 0.99) then
         iia = ia + 1
         psi(ia,1) = pso(iia,1)
         tsi(ia,1) = tso(iia,1)
         hi(ia,1) = ho(iia,1)
         rhogi(ia,1) = psi(ia,1) * 144.0 / (766.55232 * tsi(ia,1))
         rholi(ia,1) = 0.0
         rhoi(ia,1) = rhogi(ia,1)
         vi(ia,1) = 144.0 * mdot(il) / (a(ia,1) * rhoi(ia,1))
         asvi = (dummy * dabs(tsi(ia,1)))**0.5
         machi(ia,1) = vi(ia,1)/asvi
      endif
if (debug .eq. 8.0 .or. debug .ge. 10.0) then
   write(6,9990) ia,psi(ia,1),tsi(ia,1),hi(ia,1)
   write(6,9991) vi(ia,1),asvi,machi(ia,1)
   write(6,9992) rhoi(ia,1),rholi(ia,1),rhogi(ia,1)
9990 format('node # =',3x,i3,3x,'ps, ts, h (in) =',3(2x,f10.4))
9991 format('vel, a, m (in) =',3(2x,f10.4))
9992 format('rho(in) =',3(2x,f10.4))
endif
*
* CALCULATE THE HEAT TRANSFER COEFFICIENT FOR 100% GAS NODES
* BASED ON INLET PRESSURE
*
call value(n13,p13,stemp,psi(ia,1),ts)
tdim = (tsi(ia,1)-ts)/(tcnst-ts)
if (tdim .lt. 0.0) tdim = 0.0

call value2(n14,numl,p14,t14,cpg,psi(ia,1),tdim,cp)
call value2(n14,numl,p14,t14,thkg,psi(ia,1),tdim,cond)
call value2(n14,numl,p14,t14,viscg,psi(ia,1),tdim,visgfi)
ti = tsi(ia,1) * (1.0 + 0.2 * machi(ia,1) ** 2.0)
pi(ia,1) = psi(ia,1) * (1.0 + 0.2 * machi(ia,1) ** 2.0)
pdi(ia,1) = pi(ia,1) - psi(iLl)
hgl = hgfi - hlfi

dtemp = 0.0
dpress = 0.0
re = 48.0 * mdot(il)/(pyi * visgfi * exdi)
call htcoeff(phase,dtemp,dpress,cp,cond,visgfi,rholfi,1)
   rhogfi,st,hgl,re,xf,exdi,qdott)
if (debug .eq. 8.0 .or. debug .ge. 10.0) then
   write(6,9993) cp,cond,visgfi
   write(6,9994) ti,pi(ia,1),pdi(ia,1),re,qdott,
      twall(ia),tave(ia,1)
9993 format(c, k, mu t3(2x,f10.4))
9994 format('To, Po, Pd(in) t3(2x,f12.4)/,
   ' Re, Hc, Twall, Tave t4(2x,f12.4))
endif

CALCULATE THE OUTLET PROPERTIES OF THE GAS NODE

do nnl = 1, 2
   if(tave(ia,1) .le. 0.0) tave(ia,1) = ti
   if(nn1 .eq. 1) tave(ia,1) = ti
dtemp = twall(ia) - tave(ia,1)
to = ti + (pyi * length(ia) * qdott * (twall(ia)-ti) *
   exdi / (mdot(ia) * cp * 144.0))
   if (to .lt. 0.0) to = ti/2.0
   call htandf(machi(ia,1),ti,to,twall(ia),macho(ia,1))
   ratio = (1+0.2*machi(ia,1)**2.0)/(1+0.2*macho(ia,1)**2.0)
   tso(ia,1) = tsi(ia,1) * (to/ti) * ratio
   pso(ia,1) = psi(ia,1) * (machi(ia,1)/macho(ia,1)) *
      (tsi(ia,1)/tsi(ia,1))**0.5
   vo(ia,1) = vi(ia,1) * (machi(ia,1)/macho(ia,1)) *
      (tsi(ia,1)/tsi(ia,1))**0.5
   po(ia,1) = pi(ia,1) * (tso(ia,1)/tsi(ia,1)) * ratio**3.5
   pdo(ia,1) = po(ia,1) - pso(ia,1)
   ho(ia,1) = hi(ia,1) + cp * (to-ti)
   rhoo(ia,1) = rhoi(ia,1) * pso(ia,1) * tsi(ia,1) /
(psi(ia,1) * tso(ia,1))
rholo(ia,1) = 0.0
rhogo(ia,1) = rhoo(ia,1)
q(ia,1) = qdott * dtemp * sarea(ia,1)/144.0
tave(ia,1) = (tsi(ia,1) + tso(ia,1))/2.0
qdtt(ia,1) = qdott
if (debug .eq. 8.0 .or. debug .ge. 10.0) then
  write(6,9995) pso(ia,1),to,tso(ia,1),ho(ia,1),rhoo(ia,1),
  vo(ia,1),asvi,macho(ia,1),q(ia,1),dtemp,
  twall(ia),tave(ia,1)
9995 format('Ps, To, Ts(out) =',3(2x,f12.4),/,'H, rho, vel(out) ='
  ,3(2x,f12.4),/,'a, M, Qdot (out) =',3(2x,f12.4),/
  'Dt, Twall, Tave =',3(2x,f12.4))
endif
enddo
endif
m(ia,1) = rc
enddo
endif
C END OF RE-CALCULATING GAS NODES

C.......................................................................
enddo
C------------------------------------------------------------------------
C END OF THE VENT & SPRAY NODAL NETWORK FLOW PROPERTIES CALCULATION
C------------------------------------------------------------------------

* CALCULATE THE TOTAL QDOT & NEW TWALL FOR ALL THE ELEMENTS IN
THE NETWORK *

flag1 = 0.0
do jj = 1,m
  qdt = qdt + q(jj,1)
  qdtl = qdtl + q(jj,2)
  delta(jj) = q(jj,1)-q(jj,2)
  if (dabs(delta(jj)) .le. dq) flag1 = flag1 + 1.0
enddo
qerr = qdtl - qdt
* qerr = qdot(1,1) - qdt
*
IF (debug .eq. 1.0 .or. debug .ge. 10.0) THEN
  write (6,116) l,time,mass,mdot(1),ps,qerr,pes,po(1,1),
  po(2,1),po(3,1),po(4,1),po(11,1),po(12,1),
  pi(1,1),pi(2,1),pi(3,1),pi(4,1),pi(11,1),
116 format(5x,'Ps, To, Ts(out) =',3(2x,f12.4),/,'H, rho, vel(out) ='
  ,3(2x,f12.4),/,'a, M, Qdot (out) =',3(2x,f12.4),/
  'Dt, Twall, Tave =',3(2x,f12.4))
endif
pi(12,1)

format ('iteration # ',i3,2x,'time =',f8.3,3x,'mass =',
f8.3,4x,'mdot =',f10.5/, 'pst =',f8.3,4x,'qerr =',
f8.3,4x,'pes =',f8.3,6(3x,f8.3),/6(3x,f8.3))

ENDIF
if (debug .eq. 9.0) then
write(6,117) i,qdt,qdtl,qerr
do jj = 1,m
write(6,118) jj,twall(jj),q(jj,1),q(jj,2),delta(jj),

1 tsi(jj,1),tso(jj,1),tsi(jj,2),tso(jj,2),
2 qdtt(jj,1),qdtt(jj,2)
enddo
117 format ('iteration # ',i3,2x,'Gas Qdot =',f8.3,3x,
1 'Liq Qdot =',f8.4,3x,'Q error =',f8.4,//,t3,'Node #',
2 t13,'T wall',t23,'Gas Q',t33,'Liq Q',t43,'Delta Q',t53,
3 'G Tsi',t63,'G Tso',t73,'L Tsi',t83,'L Tso',t93,'G Hc',
4 t103,'L Hc',/)
118 format (6x,i4,10(2x,f8.4))
endif

*************************************************************************
* CONDITION TO END THE QDOT CONVERGENCE LOOP
* *
* IF (dabs(qerr) .le. qdterr .or. flag1 .ge. m8) goto 200
* *
* End of L do loop (The Calculated Qdot to the Nodal Model is Within the Error Margin of PERRMX)
* *
* ENDDO
*************************************************************************

200 do ii = 1,m1
nn = m - ii
qdot(nn,1) = qdot(nn+1,1) + q(nn,1)
qdot(nn,2) = qdot(nn+1,2) - q(nn,2)
enddo

c

c DETERMINE THE EXIT PLANE CONDITIONS
c

j2 = 0
if (va .eq. 1) ax = sa
PXS=PSO(1,1)
205 IF(PXS.LT.PTP) PXS=PTP
IF(PXS.GT.PST) PXS=PST
call value(n8,p8,lentp,PXS,hlx)
call value(n9,p9,gentp,PXS,hgx)

III-41
call value(n10,p10,lent,PXS,slx)
call value(n11,p11,gent,PXS,sgx)
call value(n4,p4,lrhop,PXS,rholx)
call value(n5,p5,grhop,PXS,rhogx)
YX=(SO(1,1)-SLX)/(SGX-SLX)
HX=YX*HGX+(1.0-YX)*HLX
VX=SQRT(2.0*32.2*777.649*ABS(HTOTE(1)-HX))
RHOX=144.0*MDOT(1)/(VX*AX)
RHOXV= YX/(1.0/RHOX-1.0/RHOLX*(1.0-YX))
call value(n6,rhovp,p6,RHOXV,pxsc)
if (j2 .eq. 0) psxl = pxsc
    gamma = dabs(pxs - psxl)
else
    psxl = pxs + gamma/2.0
endif
    j2 = j2 + 1
    if (j2.gt.150) go to 210
    go to 205
endif

* 210  IF (debug .eq. 7.0 .or. debug .ge. 10.0) THEN
    write (6,115) j2,pxs,time,gerr,mass,slx,sgx,rhox,yx,sx,hx
115    format ('counter = 'i4,4x,'PXS = ',fB.3,3(3x,fB.3)4,
1 6(3x,fB.3))
ENDIF
*
if (prop.eq.1.0) then
    call H2SAT(pxs,tx)
else
    call o2sat(pxs,tx)
endif

PXD=144.0/(2.0*RHOX*32.2)*(MDOT(1)/AX)*(MDOT(1)/AX)
PX=PXS+PXD
FM=MDOT(1)*VX/32.2
FP=PXS*AX
THRUST=FM+FP
ISP=THRUST/MDOT(1)
dpx = pxs - ptp
flag = 0.0
flag1 = 0.0
flag3 = 0.0
IF (dpx .le. delptp) then
  exit = 1
ENDIF

* Print Output
* 
do 11 = 1, 2
  if (11 .eq. 1) then
    write (6,1000) subhd,subt
    write (6,1430)
    write (6,1100) name(1),mdot(ll),name(2),isp,name(3),thrust,
1       name(4),qerr,name(5),rhot,name(6),pst,name(7),
2       pes(ll),name(8),htot,name(9),a(1,11)
    write (6,1200)
    write (6,1300) pxs,tx,rhox,yx,hx,vx
    write (6,1400) pes(ll),tes(ll),rhoe(ll),ye(ll),htote(ll)
  else
    write (6,1000) subhd,subt
    write (6,1460)
    write (6,1100) name(1),mdot(ll),name(2),isp,name(3),thrust,
1       name(4),qerr,name(5),rhot,name(6),pst,name(7),
2       pes(ll),name(8),htot,name(9),a(1,11)
    write (6,1200)
    write (6,1400) pes(ll),tes(ll),rhoe(ll),ye(ll),htote(ll)
  endif
do nc1 = 1,m
  write (6,1500) nc1,pso(nc1,11),tsi(ncl,11),rhoo(nc1,11),
1     yo(nc1,11),ho(nc1,11),vo(nc1,11),qdot(nc1,11),
2     a(nc1,11),phisq(nc1,11),length(nc1),dpf(nc1,11),
3     dpm(nc1,11),dpt(nc1,11),xtt(nc1,11),rn(nc1,11),
4     qdtt(nc1,11)
endo
write (6,1600) pst,tliq,rholt,yt,hs(ll)
endo

* write (6,1000) subhd,subt
write (6,1700) ps,tsp,tts,mdot(1)
do 11 = 1, 2
  if (11 .eq. 1) then
    write (6,1430)
    write (6,1800)
  else
    write (6,1460)
    write (6,1800)
  endif
endif

do nc2 = 1,m
    write (6,1900) nc2, pi(nc2,11), psi(nc2,11), pdi(nc2,11),
        1 po(nc2,11), pso(nc2,11), pdo(nc2,11), tsi(nc2,11),
        2 tso(nc2,11), twall(nc2), tave(nc2,11), q(nc2,11),
        3 delta(nc2), k(nc2,11)
endo
endo

* 
1000 format('10,18a4,10a8,/')
1100 format(9(4x,a8,f9.4,))
1200 format(t2,'NODE #',t11,'STATIC P',t23,'TEMP',t32,'DENSITY',
    1 t42,'QUALITY',t51,'ENTHALPY',t61,'VELOCITY',t73,'QDOT',
    2 t83,'AREA',t93,'PHISQ',t102,'LENGTH',t114,'DPF',t124,'DPM',
    3 t134,'DPT',t144,'Xtt',t153,'Re #',t160,'Ht Trans C')
1300 format(/,t2,'OUTLET',t11,f9.4,4(x,f9.4),x,f9.2)
1400 format(/,t2,'EXIT',t11,f9.4,3(x,f9.4),x,f9.2)
1430 format(/,t2,'FLOW PROPERTIES OF THE VENT PART OF THE SYSTEM')
1460 format(/,t2,'FLOW PROPERTIES OF THE HEAT EXCHANGER PART OF',
    1 ' THE SYSTEM')
1500 format(t4,i3,t11,f9.4,7(x,f9.4),x,f9.2,5(x,f9.4),x,f9.2,x,f9.4)
1600 format(/,t2,'INLET',t11,f9.4,5(x,f9.4))
1700 format(/,t2,'Ptank static',t21,f9.4,/,t2,'Pchoke',t21,f9.4,/,t2,
    1 'Tchoke',t21,f9.4,/,t2,'Mdot',t21,f9.4)
1800 format(/,t2,'NODE #',t13,'Pt in',t25,'Ps in',t34,'Pd in',
    1 t44,'Pt out',t53,'Ps out',t63,'Pd out',t73,'Ts in',t83,
    2 'Ts out',t93,'T wall',t103,'T ave',t113,'Qdot',t123,
    3 'Del Q',t133,'K loss',/) 
1900 format(t4,i3,t11,f9.4,12(x,f9.4))
STOP
END

subroutine H2SAT(PST,tsat)

c
double precision pst,tsat,fp,t

c
FP=LOG(PST/187.506)
T=1.00003+FP*(2.12094E-1+FP*(2.83129E-2+FP*1.75686E-3))
TSAT=59.3568*T
return
return

subroutine h2SVEL(Y,PS,RSL,RSV,as)

c
double precision ps,rsl,rsv,as,p,fpf,tstl,ts,tsm,tc,tfun,t,tk

III-44
double precision pa, dpvdt, d2pvt1, d2pvt2, csatt1, csatt2, csat
double precision drodt1, tt, drodt2, drodt, rho, dsdrho, dsvp1, dsvp

c
P = PS / 187.506264
FPF = LOG(P)
TST1 = FPF * (2.83128E-2 + FPF * (1.75686E-3))
TS = 1.00002 + FPF * (2.12094E-1 + TST1)
TSM = TS * 32.976
TC = 32.976
TFUN = (TC - TSM) ** 0.33333
T = TSM * 1.8
IF (Y .LE. 0.0) Y = 0.0
TK = T / 1.8
PA = 10.0 ** (2.00062 - 50.0970 / (TK + 1.0044) + 1.74849E-2 * TK) * 14.696
DPVDT = PA * 2.30258 * ((50.0970 / (TK + 1.0044)) ** 2 + 1.74849E-2) / 1.8
D2PVT1 = DPVDT ** 2 / PA
D2PVDT = D2PVT1 + PA * 2.30258 / 1.8 ** 2 * (-2.0 * 50.0970 / (TK + 1.0044) ** 3)
CSATT1 = 1.68157 * TK / (32.976 - TK) ** 0.1 - 32.8027
CSATT2 = TK * (3.35743E-2 + TK * (-7.68297E-4 + 6.90292E-6 * TK))
CSAT = (CSATT1 + TK * (6.81698 + TK * (-0.731943 + CSATT2))) / 2.01572
DRODT1 = -0.38 * 7.32346E-3 / (32.976 - TK) ** 0.62 + 4.40742E-4
TT = (32.976 - TK) ** 0.33333
DRODT2 = TT * TT * (1.66667 * 2.92263E-04 - 2.0 * 4.00849E-05 * T)
DROLDT = (DRODT1 - 1.33333 * 6.62079E-4 * T + DRODT2) * 69.9099
RHO = 1.0 / (1.0 / RSL + Y * (1.0 / RSV - 1.0 / RSL))
DSDRHO = -1.0 / RHO / RHO * DPVDT * 144.0
DSDP1 = 777.649 / 144.0 * CSAT / (T * DPVDT) + 1.0 / RSL / RSL * DROLDT
DSDP = DSDP1 + (1.0 / RHO - 1.0 / RSL) * D2PVDT / DPVDT
AS = SQRT(ABS(-DSDRHO / DSDP * 32.2))
return
END

subroutine htcoeff(Phase, dt, dp, cp, k, mu, rho1, rho2, st, lambda, re, x, 1
diam, qdot)

c
This subroutine calculates the two phase heat transfer coefficient

c
double precision cp, k, mu, rho1, rho2, st, x
double precision lgx(12), lgf(12), lgsf(12), sf(12), f, s, c2, d3
real lambda

c
data lgx/-1.0, -0.824, -0.745, -0.699, -0.585, 0.0, 0.176, 0.398, 0.663,
1 0.778, 1.778, 2.0/
data lgf/0.0, 0.0212, 0.0414, 0.0569, 0.1139, 0.4472, 0.5563, 0.699,
1 0.8751, 0.9542, 1.699, 1.8692/
data lgsf/0.176,0.415,0.672,1.0,1.19,1.29,1.398,1.544,1.602,
1       1.663,1.778,2.0/
data sf/0.84,0.73,0.58,0.37,0.25,0.2,0.16,0.12,0.11,0.1,0.09,
1       0.09/
c
dtemp = abs(dt)
dpress = abs(dp)
gc = 32.2
a1 = .023
a2 = re**0.8
a3 = (cp * mu * 3600. / k)**0.4
a4 = (k * 12.) / (3600. * diam)
hfc = a1 * a2 * a3 * a4
if (phase .lt. 1.5 .or. phase .ge. 3.0) then  
  qdot = hfc  
endif  
c write(6,*) cp,mu,k,diam,a1,a2,a3,a4,qdot
  return
endif  
c write(6,*) a1,a2,a3,a4
c  
b1 = .00122
b2 = ((k/3600.)**0.79) * (cp**0.45) * (rhol**0.49) * (ge**0.25) *  
1       (dtemp**0.24) * (dpress**0.75)
b3 = ((st*12.)**0.5) * (mu**0.29) * (lambda**0.24) * (rhov**0.24)
hfz = bl * b2/b3
c write(6,*) bl,b2,b3
c  
c1 = 1 / x  
c2 = log10(c1)
if (c2 .gt. 2.0) c2 = 2.0
  call value(12,lgx,lgf,c2,f)
  f = 10.0**f
  write( 6, *) c 1 ,c2,f
c  
d1 = re * f**1.25
d2 = d1 / 10000.0
d3 = log10(d2)
if (d3 .gt. 2.0) d3 = 2.0
  call value(12,1gsf,sf,d3,s)
c write(6,*) d1,d2,d3
  qdot = hfc*f + hfz*s
  write (6,100) b2,b3,c1,c2,d1,d2
  write (6,100) re,hfc,f,hfz,s,qdot
cl00  format (3(3x,f14.4),3(3x,f14.4))
  return

III-46
subroutine INITL(PO,K,MDOT,A,RHO,pin)
  double precision po,rho,pin,dpf
  real k,mdot
  DPF=144.0*K*((MDOT/A)**2.0)/(2.0*RHO*32.2)
PEN=PO+DPF
  return
END

subroutine spray(mdot,psmi,dsm,ed,pback)
c
  subroutine spray.f to model flow in the spray injection tube
  c
  character*1 label
double precision rod(12),lod(12),roverd,loverd,pback
real mdin(200),mdout(200),mds(200),as(200),vel(200)
real pin(200),pout(200),pnode(200),ptank(200),x(200),delpt(200)
real node(10),asec(10)
real subhd(18),xtitl(18),yp(18),ymdot(18),ymds(18),
  yvel(18),yas(18)
real mdot,mdoti,mdoto,mdots,mdotp
real kfsm,kbsm,kcsm,kbsi,ks
integer*2 ibox,iloc
common /contrl/ibox,iloc
data nb/12/rod/1.,1.5,2.,3.,4.,6.,8.,10.,12.,14.,16.,20./
data lod/20.,14.,12.,12.,14.,17.,24.,30.,34.,38.,42.,50./
  ibox = 1
  iloc = 0
*
  open (unit=8, file='spray.data',status='unknown')
*
  read (8,100) label
  read (8,*) pu,zliq,ztank,dmdot
  read (8,100)
  read (8,*) zsm,dsi,zsi,n,nbar
  read (8,100)
  read (8,*) ks,cds,acc,rho,visc,roverd
  read (8,100)
  read (8,*) tol,nlim
  read (8,100)
  read (8,*) opn,opt,omdin,omds,ovel,oas
  read (8,101) subhd,xtitl,yp,ymdot,ymds,yvel,yas
read (8,*) nsec
do i = 1,nsec
    read (8,*) node(i),asec(i)
enddo

100 format (/a)
101 format (18a4/18a4/18a4/18a4/18a4/18a4/18a4)
write (6,1)

1 format (5x,'SPRAY MANIFOLD AND INJECTION TUBE FLOW MODEL')
write (6,2) zliq,pu,psmi,acc,rho,zsm,dsm

2 format (5x,'Liquid Level = ',f6.1, ' in'/
   1 5x,'Ullage Pressure = ',f6.2, ' psia'/
   2 5x,'Spray Manifold Inlet Pressure = ',f6.3, ' psia'/
   3 5x,'Acceleration Level = ',f6.1, ' g'/
   4 5x,'Liquid Density = ',f6.3, ' lbm/ft3'/
   5 5x,'Spray Manifold Tube Length = ',f6.1, ' in'/
   6 5x,'Spray Manifold Tube ID = ',f6.2, ' in')

\[ dzsi = \frac{zsi}{n} \]
write (6,3) nbar,zsi,dsi,n,dzsi,ks

3 format (5x,'Number of Spray Injection Tubes = ',i6/
   1 5x,'Spray Injection Tube Length = ',f6.1, ' in'/
   2 5x,'Spray Injection Tube ID = ',f6.3, ' in'/
   3 5x,'Number of Orifices = ',i6/
   4 5x,'Orifice Spacing = ',f6.2, ' in'/
   5 5x,'Orifice Loss Coefficient = ',f6.2/) 
pu = 144.*pu
psmi = 144.*psmi
tol = 144*tol
zliq = zliq/12.
ztank = ztank/12.
dsm = dsm/12.
ds = dsi/12.
zsm = zsm/12.
zsi = zsi/12.
dzsi = dzsi/12.
as = 3.14159*dsm*dsm/4.
asi = 3.14159*dsi*dsi/4.
do j = 1,nsec
    n1 = n2 + 1
    n2 = node(j)
    do i = n1,n2
        as(i) = asec(j)/144.
    enddo
enddo

kcsm = .5*(1. - (dsi/dsm)**2)
call value(nb,rod,lod,roverd,loverd)
zu = ztank - zliq
ge = 32.2
do 10 j = 1,nlim
   qsm = (mdot/asm)**2/(2.*rho*ge)
call frict(dsm,ed,mdot,visc,fsm)
kfsm = fsm*zsm/dsm
kbsm = fsm*loverd
dpfsm = qsm*(kfsm + kbsm + kcsm)
dpf1 = qsm*(kfsm)
   phsm = rho*acc*zsm
psmo = psmi - dpfsm - phsm
pback = psmi - dpf1
   dpf = qsm*(kfsm)
   phsm = rho*acc*zsm
psmo = psmi - dpfsm - phsm
pback = pback/144.0
   dpfsm = dpfsm + phsm
mdoti = mdot/nbar
   qi = (mdoti/asi)**2/(2.*rho*ge)
call frict(dsi,ed,mdoti,visc,fsi)
kbsi = fsi*loverd
   dpsi = qi*kbsi
   pi = psmo - dpsi
do i = 1,n
   x(i) = i*dzsi - dzsi/2.
aorf = as(i)
   z = ztank - zsi + x(i)
   if (z .lt. zu) pt = pu
   if (z .ge. zu) pt = pu + rho*acc*(z - zu)
call pres(i,n,pi,po,pn,pt,dpf,ph,mdoti,mdoto,mdots,
   dzsi,dsi,ed,asi,aorf,ks,rho,visc,acc)
   if (aorf .le. 0.0) vel(i) = 0.0
   if (aorf .gt. 0.0) vel(i) = mdots/(rho*cds*aorf)
   pin(i) = pi
   pout(i) = po
   pnode(i) = pn
   ptank(i) = pt
   mdin(i) = mdoti
   mdout(i) = mdoto
   mds(i) = mdots
   pi = pout(i)
   mdoti = mdout(i)
   if (i .lt. n) dpsi = dpsi + dpf - ph
   if (i .eq. n) dpsi = dpsi + dpf - ph/2.
enddo
ptcal = pn - ks/(2.*rho*ge)*(mdots/aorf)**2
delpt(j) = ptcal - pt
if (j .eq. 1) then
if (delpt(j) .lt. 0.) mdot = mdot - dmdot
if (delpt(j) .gt. 0.) mdot = mdot + dmdot
else
    prod = delpt(j)*delpt(j-1)
    if (delpt(j) .lt. 0. and. prod .gt. 0.) mdot = mdot - dmdot
    if (delpt(j) .gt. 0. and. prod .gt. 0.) mdot = mdot + dmdot
    if (delpt(j) .lt. 0. and. prod .lt. 0.) then
        dmdot = dmdot/2.
        mdot = mdot - dmdot
        endif
    if (delpt(j) .gt. 0. and. prod .lt. 0.) then
        dmdot = dmdot/2.
        mdot = mdot + dmdot
        endif
endif
10 continue
if (abs(delpt(j)) .gt. tol) write (6,1002) delpt(j)
1002 format ('*** tank pressure does not converge, delpt = ',
     1 f8.4,' psi ***')
15 dppump = (psmi - pt)/144.
dpsm = dpsm/144.
dpsi = dpsm/144.
write (6,200) mdot,dppump,dpsm,dpsi
200 format ('Pump Flow Rate = ',f6.3,' lbm/sec/
     1 5x,'Pump Pressure Rise = ',f6.3,' psi/
     2 5x,'Spray Manifold Tube Delta p = ',f6.3,' psi/
     3 5x,'Spray Injection Tube Delta p = ',f6.3,' psi//)
write (6,4)
4 format (5x,'Node',2x,'Distance',3x,'Inlet',2x,'Outlet',
     1 3x,'Nodal',3x,'Tank',5x,'Inlet',4x,'Outlet',
     2 3x,'Injection',3x,'Injection',3x,'Orifice')
write (6,5)
5 format (24x,'p',7x,'p',7x,'p',7x,'p',7x,'p',
     1 7x,'mdot',5x,'mdot',7x,'mdot',6x,'Velocity',
     2 5x,'CdA')
write (6,6)
6 format (13x,'(in)',4x,'(psia)',2x,'(psia)',2x,'(psia)',2x,'(psia)',
     1 2x,'(lbm/sec)',1x,'(lbm/sec)',1x,'(lbm/sec)',5x,'(fps)',
     2 6x,'(in2)')
do i = 1,n
    pin(i) = pin(i)/144.
pout(i) = pout(i)/144.
pnode(i) = pnode(i)/144.
ptank(i) = ptank(i)/144.
as(i) = 144.*as(i)
x(i) = 12.*x(i)
write (6,210) i,x(i),pin(i),pout(i),pnode(i),ptank(i),
1 mdin(i),mdout(i),mds(i),vel(i),as (i)
210 format (5x,i3,4x,f6.2,3x,4(f6.3,2x),
1 3(e9.3,1x),4x,f5.2,4x,f7.5)
enddo
if (opn .eq. 0.) go to 20
call crtplt(-11,11,2,00,0,1,subhd,xtitl,yp,0.,0.,
1 0.,0.,n,x,pnode,0,0)
20 if (opt .eq. 0.) go to 21
call crtplt(0,0,0,0,1,0,0,0,0,0,,
1 0.,0.,n,x,ptank,0,0)
21 if (omdin .eq. 0.) go to 22
call crtplt(-11,11,2,00,0,1,subhd,xtitl,ymdot,0.,0.,
1 0.,0.,n,x,mdin,0,0)
22 if (omds .eq. 0.) go to 23
call crtplt(-11,11,2,00,0,1,subhd,xtitl,ymds,0.,0.,
1 0.,0.,n,x,mds,0,0)
23 if (ovel .eq. 0.) go to 24
call crtplt(-11,11,2,00,0,1,subhd,xtitl,yvel,0.,0.,
1 0.,0.,n,x,vel,0,0)
24 if (oas .eq. 0.) go to 25
call crtplt(-11,11,2,00,0,1,subhd,xtitl,yas,0.,0.,
1 0.,0.,n,x,as,0,0)
25 return
end

subroutine TPHS(HI,RHOL8,RHOG8,MDOT,AN8,HL8,HG8,SL8,SG8,
1 y8,h8,s8,rho8,p8d)
double precision hi,rhol8,rhog8,h8,s8,rho8,p8d
real k,mdot

c
c
K=0.41405*(MDOT*MDOT)/(AN8*AN8)
A1=K*(1.0/RHOG8-1.0/RHOL8)*(1.0/RHOG8-1.0/RHOL8)
B1=HG8-HL8+2.0*K/RHOL8*(1.0/RHOG8-1.0/RHOL8)
C1=HL8-HI+K/(RHOL8*RHOL8)
Y8=(-B1+SQRT(ABS(B1*B1-4.0*A1*C1)))/(2.0*A1)
IF(Y8.LE.0.00) then
  y8=0.00
endif
IF(Y8.GE.1.0) then
  y8=1.00
endif
H8=Y8*HG8+(1.0-Y8)*HL8
S8=Y8*SG8+(1.0-Y8)*SL8  
RHO8=1.0/(1.0/RHOL8+Y8*(1.0/RHOG8-1.0/RHOL8))  
P8D=144.0/(2.0*32.2*RHO8)*(MDOT/AN8)*(MDOT/AN8)  
return  
END

subroutine TPS(A,G,HI,HO,K,P,YO,RLO,RGO,YI,RLI,RGI,MDT,PHISQ,b,
1 d,e,f)  
double precision p,rlo,rgo,rli,rgi,phisq,b,d,e,f  
double precision dpfl,dpf,vo,vi,dpm,dpt,ph,pinlet  
real k,mdt  

RHOO=1.0/(1.0/RLO+YO*(1.0/RGO-1.0/RLO))  
RHOI=1.0/(1.0/RLI+YI*(1.0/RGI-1.0/RLI))  
RHO=(RHOO+RHOI)/2.0  
YA=(YI+YO)/2.0  
IF(YA.GT.1.0) then  
  ya=0.999  
endif  
RHOLA=(RLI+RLO)/2.0  
aa = ((1.0-ya)**2.0)*PHISQ/(RHOLA)  
bb = 1.0 / rho  
cc = aa  
if (bb .ge. aa) cc = bb  
DPF1=(MDT/A)**2  
DPF=144.0*K*(DPF1)*cc/(2.0*32.2)  
B=DPF  
VO=MDT/(RHOO*A)*144.0  
VI=MDT/(RHOI*A)*144.0  
DPM=ABS(MDT/A*(VO-VI)/32.2)  
DPT=DPM+DPF  
PH=G*RHO*(HI-HO)/1728.0  
PINLET=P+DPT-PH  
D=PINLET  
E=DPT  
F=DPM  
return  
END

subroutine value(np,x,y,xin,yout)  
c  
This subroutine performs lagrangian interpolation within a set  
of (x,y) pairs to give yout corresponding to xin  
c  
double precision x(np),y(np),xin,yout

III-52
if (xin .le. x(1)) yout = y(1)
if (xin .le. x(1)) return
if (xin .ge. x(np)) yout = y(np)
if (xin .ge. x(np)) return
do 10 i = 1,np
   k = i
   if (xin .lt. x(i)) go to 30
   10 continue
30 ffr = (xin - x(k - 1))/(x(k) - x(k - 1))
yout = y(k - 1) + ffr*(y(k) - y(k -1))
c write (6,*) np,x,y,xin,yout
   return
end

SUBROUTINE VALUE2(NPX,NPY,X,Y,Z,XIN,YIN,ZOUT)
c DIMENSION X(NPX),Y(NPY),Z(NPX,NPY)
double precision X(NPX),Y(NPY),Z(NPX,NPY),XIN,YIN,ZOUT
C IF(XIN .LE. X(1) .AND. YIN .LE. Y(1))ZOUT=Z(1,1)
  IF(XIN .LE. X(1) .AND. YIN .LE. Y(1))RETURN
C IF(XIN .GE. X(NPX) .AND. YIN .LE. Y(1))ZOUT=Z(NPX,1)
  IF(XIN .GE. X(NPX) .AND. YIN .LE. Y(1))RETURN
C IF(XIN .LE. X(1) .AND. YIN .GE. Y(NPY))ZOUT=Z(1,NPY)
  IF(XIN .LE. X(1) .AND. YIN .GE. Y(NPY))RETURN
C IF(XIN .GE. X(NPX) .AND. YIN .GE. Y(NPY))ZOUT=Z(NPX,NPY)
  IF(XIN .GE. X(NPX) .AND. YIN .GE. Y(NPY))RETURN
C IF(XIN .GT. X(1))GO TO 30
  DO 20 I=1,NPY
     M=I
     IF(YIN .LT. Y(I))GO TO 25
  20 CONTINUE
25 FFRY=(YIN-Y(M-1))/(Y(M)-Y(M-1))
   ZOUT=Z(1,M-1)+FFRY*(Z(1,M)-Z(1,M-1))
   RETURN
C
30 IF(XIN .LT. X(NPX))GO TO 60
  DO 50 I=1,NPY
     M=I
     IF(YIN .LT. Y(I))GO TO 55
  50 CONTINUE
FFRY = (YIN - Y(M-1)) / (Y(M) - Y(M-1))
ZOUT = Z(NPX, M-1) + FFRY * (Z(NPX, M) - Z(NPX, M-1))
RETURN

C
IF(YIN .GT. Y(1)) GO TO 90
DO 80 I = 1, NPX
   L = I
   IF(XIN .LT. X(I)) GO TO 85
80 CONTINUE
FRX = (XIN - X(L-1)) / (X(L) - X(L-1))
ZOUT = Z(L-1, 1) + FRX * (Z(L, 1) - Z(L-1, 1))
RETURN

C
IF(YIN .LT. Y(NPY)) GO TO 120
DO 110 I = 1, NPX
   L = I
   IF(XIN .LT. X(I)) GO TO 115
110 CONTINUE
FRX = (XIN - X(L-1)) / (X(L) - X(L-1))
ZOUT = Z(L-1, NPY) + FRX * (Z(L, NPY) - Z(L-1, NPY))
RETURN

C
DO 130 I = 1, NPX
   L = I
   IF(XIN .LT. X(I)) GO TO 135
130 CONTINUE
FRX = (XIN - X(L-1)) / (X(L) - X(L-1))
DO 140 I = 1, NPY
   M = I
   IF(YIN .LT. Y(I)) GO TO 145
140 CONTINUE
FYR = (YIN - Y(M-1)) / (Y(M) - Y(M-1))
ZXLO = Z(L-1, M-1) + FYR * (Z(L-1, M-1) - Z(L-1, M-1))
ZXHI = Z(L, M-1) + FYR * (Z(L, M) - Z(L, M-1))
C
ZOUT = ZXLO + FRX * (ZXHI - ZXLO)
RETURN
END

SUBROUTINE VALUE3(NPX, MAXY, NPY, X, Y, Z, XIN, YIN, ZOUT)
integer npy(maxy)
double precision X(NPX), Y(NPX, MAXY), Z(NPX, maxy), xin, yin, zout
double precision X(15), Y(15, 15), Z(15, 15), xin, yin, zout
dimension a(50), b(50), c(50), d(50)
C
   IF(XIN .LE. X(1) .AND. YIN .LE. Y(1,1)) then
      ZOUT=Z(1,1)
      RETURN
   endif
   C
   IF(XIN .GE. X(NPX) .AND. YIN .LE. Y(npx,1)) then
      ZOUT=Z(NPX,1)
      RETURN
   endif
   C
   IF(XIN .LE. X(1) .AND. YIN .GE. Y(1,NPY(1))) then
      ZOUT=Z(I,NPY(1))
      RETURN
   endif
   C
   IF(XIN .GE. X(NPX) .AND. YIN .GE. Y(npx,NPY(npx))) then
      ZOUT=Z(NPX,NPY(npx))
      RETURN
   endif
   C
   IF(XIN .GT. X(1))GO TO 30
   DO I=1,NPY(1)
      M=I
      IF(YIN .LT. Y(I,1))GO TO 25
   enddo
   25 FFRY=(YIN-Y(I,M-1))/(Y(I,M)-Y(I,M-1))
      ZOUT=Z(I,M-1)+FFRY*(Z(I,M)-Z(I,M-1))
      RETURN
   C
   30 IF(XIN .LT. X(NPX))GO TO 60
   DO I=1,NPY(npx)
      M=I
      IF(YIN .LT. Y(npx,I))GO TO 55
   enddo
   55 FFRY=(YIN-Y(npx,M-1))/(Y(npx,M)-Y(npx,M-1))
      ZOUT=Z(NPX,M-1)+FFRY*(Z(NPX,M)-Z(NPX,M-1))
      RETURN
   C
   60 DO I=1,NPX
      L=I
      IF(XIN .LT. X(I))GO TO 85
   enddo
   85 do j = 1, npy(l)
      a(j) = y(l,j)
   }
\[ b(j) = z(l,j) \]
enddo

call value(npy(l),a,b,yin,z1)
do k = 1, npy(l-1)
    c(k) = y(l-1,k)
    d(k) = z(l-1,k)
endo
call value(npy(l-1),c,d,yin,z2)
FFRX=(XIN-X(L-1))/(X(L)-X(L-1))
ZOUT=Z2+FFRX*(Z1-Z2)
RETURN
END

subroutine XPARAM(YO, VLO, VGO, RLO, RGO, YI, VLI, VGI, RLI, RGI, x)
double precision vlo, vgo, rlo, rgo, vli, vgi, rli, rgi, x
double precision visla, visga, rhola, rhoga, x
YA=(YI+YO)/2.0
VISLA=(VLI+VLO)/2.0
VISGA=(VGI+VGO)/2.0
RHOGA=(RGI+RGO)/2.0
RHOLA=(RLI+RLO)/2.0
IF(YA.LE.1.0E-06) then
    ya=1.0e-06
endif
IF(YA.GE.1.0) then
    ya=1.0
endif
X1=((ABS(VISLA/VISGA)**0.1)*((1.0-YA)/YA)**0.9)
X=X1*SIN(ABS(RHOGA/RHOLA))
c write (6,10) yi, yo, ya, vli, vlo, visla, vgi, vgo, visga, rgi, rgo, rhoga,
1   rli, rlo, rhola, x1, x
c 10 format (6(3x,f8.4),/6(3x,f8.4),/3(3x,f8.4),2(3x,f10.4))
return
END

3.4.2 Integrated Zero-g TVS Model

c program tvs.f to model zero-g Thermodynamic Venting System (TVS)
c transient performance
c
character*1 label
real ptitl(18,4), subhd(18), xtitl(18), ketitl(4,2),
1   ypu(18), ytu(18), ymu(18), ypl(18), ytl(18), yml(18),

III-56
real tsat(25), psat(25), enthf(25), enthg(25), 
 1 shpf(25), densf(25), condf(25), viscf(25), texpf(25) 
real pvap(8), tvap(8), tnrm(11), enth(8,11), shv(8,11),
 1 shp(8,11), dens(8,11), cond(8,11), visc(8,11)
real tal(12), shpal(12), node(10), dorf(10)
real as(100), ddrop(100), ad(100), vd(100), veld(100), ndrop(100)
real ton(100), toff(100), tcyc(100), pton(100),
 1 mvent(100), mvnt(100), sfc(100)
real mup, mlp, mwlp, mvp, mdvp
real mdsp, mdslp, mdsup, mddup, mdbwp, mdlup, npumpp
real mu, ml, mwl, mw, mv, mui, mli, mwli, mvi, mwlnax
real kwu, muwu, kuwl, muuw1, kwl, muwl, muul, kul
real kus, muus, kls, muls, kud, muud, nud
real ks, mds, mdsl, mdsu, mddu, mdbw, mdbwmx, mdlu, mdlul
real mdv, mdvent, mdp, mdpump, mdsne, mdsul, mdcnd, mdcndp, mdsi
real mdotd, npumpd, npump, npumpi, nmax, nmin, ip

integer jsymb(2)
integer*2 ibox, iloc
common /pltcom/misc(3), nc, miss(13), dclim, ltick, nfig, nptmin,
 1 nlines, nchlin, ptitl
common /contrl/ibox, iloc
common /sprayin/dsm, zsm, dsi, zsi, norf, nbar,
 1 ks, cds, as, vd, roverd, dmdot, tol, nlim,
 2 rhos, visc
common /pumpin/npumpd, nmin, nmax, hpid, hpmotr, effp, ip, dm
common /tankdim/dtank, hcyl, hbulk, tkw, c1
common /tankvol/v1, v2, vbi, vbo, vci, vco
common /tankarea/ab, ac, at
common /tankout/j, nline, iprint, timep(3000),
 1 pup(3000), tup(3000), vup(3000), mup(3000), twp(3000),
 2 plp(3000), tlp(3000), vlp(3000), mlp(3000),
 3 pwlp(3000), twlp(3000), vwlp(3000), mwlp(3000),
 4 mvp(3000), tsp(3000), mdlup(3000), mdsp(3000), mdslp(3000),
 5 mdsup(3000), mddup(3000), mdbwp(3000), mdlup(3000), mdcndp(3000),
 6 qwp(3000), qwvlp(3000), qwlp(3000), qulp(3000), qudp(3000),
 7 qusp(3000), qlsp(3000), npumpp(3000), dppmp(3000)
nc = 2
nfig = 0
nlines = 4
ibox = 257
jsymb(1) = 40
jsymb(2) = 43
data ketitl/4hEVAP, 4hORAT, 4hTION, 4h ,
read input data for tank model

read (5,100) label
read (5,*) xd,xchar,he,xcond,prtsp,outp
read (5,100) label
read (5,*) pui,tui,pli,twi,twli,full,x11
read (5,100) label
read (5,*) dtank,hcy1,hbulk,tkw,dsb,d1,d2
read (5,100) label
read (5,*) mdvent,mdsi,dthex,qflux,g,hwliq
read (5,100) label
read (5,*) pmin,pmax,delt2,ipmt2,iplot2
read (5,100) label
read (5,*) fintim,delt1,xdelt1,ipmt1,iplot1,nline
read (5,100) label
read (5,*) opu,otu,omu,opl,otl,oml
read (5,100) label
read (5,*) otw,omwl,omdv,omdsu,omddu,omdbw
read (5,100) label
read (5,*) omdlu,omdul,omds,onpump,odppmp
read (5,200) subhd,xtitl,ypu,ytu,ymu,ypl
read (5,200) ytl,ylml,ytw,ymwl,ymdv,ymdsu
read (5,200) ymddu,ymdbw,ymdif,ymds,ynpump,ydppmp
read (5,201) ptitl

read input data for pump model

read (5,100) label
read (5,*) mdotd,dpd,npumpi,npumpd,xhp,xn
read (5,100) label
read (5,*) deltat,effp

read input data for spray manifold/injection tubes model

read (5,100) label
read (5,*) dsm,zsm,dsi,zsi,norf,nbar
read (5,100) label
read (5,*) ks,cds,roverd,dmdot,tl,nlim
read (5,*) nsec
do i = 1,nsec
    read (5,*) node(i),dorf(i)
enddo
100 format (/a1)
200 format (18a4/18a4/18a4/18a4/18a4/18a4)
201 format (18a4)
c read LH2 saturation properties
c open(unit=2,file='h2prop',status='old')
read (2,*) nsat
do i = 1,nsat
    read (2,*) tsat(i),psat(i),enthf(i),dumvar,shpf(i),densf(i),
        texpf(i),condf(i),visef(i)
read (2,*) enthg(i)
endo
c read GH2 properties as a function of pressure and temperature
c read (2,*) np,nt
read (2,*) (tnrm(i),i=1,nt),tcnst
do i = 1,np
    read (2,*) pvap(i),tvap(i)
do j = 1,nt
    read (2,*) enth(i,j),shv(i,j),shp(i,j),dens(i,j),
        cond(i,j),visc(i,j)
endo
dndo
if (outp .eq. 1.0) then
    write (6,1)
1 format (5x,'TANK DIMENSIONS'/)
write (6,2) dtank,tkw,hcyl,hbulk
2 format (5x,'Tank Diameter' ,',f6.1,' in'/
1 5x,'Tank Wall Thickness ' ,',f6.2,' in'/
2 5x,'Cylinder Height ' ,',f6.1,' in'/
3 5x,'Bulkhead Height ' ,',f6.1,' in'//)
write (6,3)
3 format (5x,'SPRAY MANIFOLD/INJECTION TUBE DIMENSIONS'/)
write (6,4) zsm,dsm,zsi,dsi,nbar,norf,ks,cds
4 format (5x,'Spray Manifold Tube Length ' ,',f6.1,' in'/
1 5x,'Spray Manifold Tube ID ' ,',f6.3,' in'/
2 5x,'Spray Injection Tube Length ' ,',f6.1,' in'/
3 5x,'Spray Injection Tube ID ' ,',f6.3,' in'/
4 5x,'Number of Spray Injection Tubes ' ,',i6/
5 5x,'Number of Orifices ',i6/
6 5x,'Orifice Loss Coefficient ',f6.2/
7 5x,'Orifice Discharge Coefficient ',f6.2/
write (6,5)
   5 format (t2,'Time',t10,'pU',t18,'TU',t26,'VU',t34,'MU',
1   t42,'pL',t50,'TL',t58,'VL',t66,'ML',t74,'TW',
2   t82,'TWL',t90,'Npump',t98,'dppump',t106,'dTpump',
3   t114,'HPO',t122,'mdS',t130,'mdSU',t138,'mdDU',
4   t146,'mdBW',t154,'mdLU',t162,'mdcond')
write (6,6)
   6 format (t2,'sec',t9,'psia',t19,'R',t25,'ft3',t33,'lbm',
1   t41,'psia',t51,'R',t57,'ft3',t65,'lbm',t75,'R',
2   t83,'R',t91,'rpm',t99,'psid',t109,'R',
3   t114,'HP',t120,'lbm/sec',t128,'lbm/sec',t136,'lbm/sec',
4   t144,'lbm/sec',t152,'lbm/sec',t160,'lbm/sec')
endif
do j = 1,nsec
   node1 = node2 + 1
   node2 = node(j)
do i = node1,node2
      ds = dorf(j)/12.0
      as(i) = 3.14159*ds**2/4.0
      ddrop(i) = xd*ds
      ad(i) = 3.14159*ddrop(i)**2
      vd(i) = 3.14159*ddrop(i)**3/6.0
endo
dendo
dtank = dtank/12.0
hcy1 = hcy1/12.0
hbulk = hbulk/12.0
htank = hcy1 + 2.0*hbulk
tkw = tkw/12.0
dsb = dsb/12.0
dsm = dsm/12.0
zsm = zsm/12.0
dsi = dsi/12.0
zsi = zsi/12.0
xll = xll/12.0
d1 = d1/12.0
d2 = d2/12.0
dzsi = zsi/norf
dchar = xchar*dtank
c
  c initial ullage and liquid masses
c

III-60
call volarea(vt) 
dtanki = dtank - 2.0*tkw  
call value(nsat,psat,tsat,pli,tli)  
call value(nsat,tsat,densf,tli,rhol)  
call value(nsat,tsat,densf,twli,rhowl)  
vli = full/100.0*vt  
mli = rhol*vli  
vui = vt - vli  
mui = 144.0*pui/vui/(ru*tui)  
tsl = tli  
tsw = ts

c in 1 g, set maximum thickness of wall liquid layer to 0.01 in

c due to liquid run-off and calculate maximum wall liquid mass

call area(vli,y,awu,aul,awl,hliq,hu)
if (g .ge. 1.0) mwlmax = rhol*awu*0.01/12.0

c pump design conditions

dm = 720.0/(3.14159*npumpd)*(2.0*144.0*dpd/rhol)**0.5  
hpid = xhp*mdotd*dpd*144.0/(550.0*effp*rhol)  

tmp = hpid  
nmax = npumpd*(1.0 + xn)  
nmin = npumpd*(1.0 - xn)  

ip = 6.018e+05*hpid*deltat/(npumpd*(nmax - nmin))/2.0  
if (pui .ge. pmax) mdp = mdsi  
if (pui .le. pmin) mdp = 0.0  

mdpump = mdsi

c time integration of variables

c

i = 0  
j = 0  
85 if (npump .gt. 0.0) then  
    delt = delt1  
    if (pu .le. pl) delt = xdel1*delt1  
    iprint = iprnt1/xdel1  
    iplot = iplot1  
else  
    delt = delt2  
    iprint = iprnt2  
    iplot = iplot2
endif  

mu = mui + dmudt*delt
ml = mli + dmldt*delt
if (ml .le. 0.0) ml = 0.0
mwl = mwli + dmwldt*delt
if (mwl .le. 0.0) mwl = 0.0
if (g .ge. 1.0 .and. mwl .ge. mwlmax) mwl = mwlmax
mv = mvi + mdv*delt
vl = vii + dvldt*delt
if (vl .le. 0.0) vl = 0.0
vwl = mwl/rhowl
vu = vt - vl - vwl
tl = tli + dtldt*delt
tu = tui + dtudt*delt
call value(nsat,psat,tsat,pu,tusat)
if (tu .le. tusat) tu = tusat
twl = twli + dtwldt*delt
if (twl .ge. tusat) twl = tusat
if (twl .le. ts) twl = ts
tw = twi + dtwtdt*delt
qpump = qpumpi + hpo*0.707*delt
mui = mu
mli = ml
mwli = mwl
mvi = mv
vli = vl
tui = tu
tli = tl
twli = twl
twi = tw
qpumpi = qpump
c c ullage, bulk liquid, and wall liquid pressures
c pu = mu*ru*tu/(144.0*vu)
call value(nsat,tsat,psat,tl,pl)
call value(nsat,tsat,psat,twl,pwl)
c c pump control logic
c if (pu .ge. pmax) flag1 = 1.0
if (pu .le. pmin) flag1 = 0.0
c c performance calculations
c if (flag1 .eq. 0.0) then
if (cyc .eq. 0.0) ncyc = ncyc + 1

III-62
toff(ncyc) = time - timeon

timeof = time

cyc = 1.0

endif

if (flag1 .eq. 1.0) then

  ton(ncyc) = time - timeof

  tcyc(ncyc) = ton(ncyc) + toff(ncyc)

  pton(ncyc) = 100.0*ton(ncyc)/tcyc(ncyc)

  sfc(ncyc) = 3600.0*mv/time

  mvent(ncyc) = mv

  pup(ncyc) = pu

  tup(ncyc) = tu

  mup(ncyc) = mu

  plp(ncyc) = pl

  tlp(ncyc) = tl

  mlp(ncyc) = ml

  twp(ncyc) = tw

  mdsp(ncyc) = mds

  timeon = time

  cyc = 0.0

endif

c pump model

c
call value(nsat,tsat,shpf,tl,cpl)

nmdot = 0

87 call pump(flag1,npump,npumpi,delt,mdpump,rhol,cpl,dppump,dtpump,
  1 dnt,hpo)

  if (dppump .le. 0.01) mdp = 0.0

  if (dppump .gt. 0.01) mdp = mdpump

  tpump = tl + dtpump

c vent control logic

c
psatp = -36.37 + 4.6054*tpump - 0.20369*tpump*tpump
  1 + 0.0031745*tpump*tpump*tpump

  if (psatp .gt. pmin) flag2 = 1.0

  if (psatp .le. pmin) flag2 = 0.0

  mdv = 0.0

  if (flag1 .gt. 0.0 .and. flag2 .gt. 0.0) mdv = mdvent

c pressure drop in the recirculation line (between the pump outlet
  and spray manifold inlet)

c
  if (mdp .le. 0.0) dprec = 0.0

III-63
if (mdp .gt. 0.0) then
    call value(nsat,tsat,densf,tpump,rhop)
    call value(nsat,tsat,viscf,tpump,viscp)
    call pdrop(mdp,rhop,viscp,xll,d1,d2,dprec)
endif

c
spray manifold/injection tube model
c
call area(vl,y,awu,aul,awl,hliq,hu)
dhn = htank - hu - dzsi/2.0
ptn = pu + rhol*g*dhn/144.0
psmi = ptn + dppump - dprec
if (mdp .gt. 0.0) ts = tpump - dthex
if (mdp .le. 0.0) ts = tpump
    call value(nsat,tsat,densf.ts,rhos)
    call value(nsat,tsat,viscf,ts,viscs)
    if (mdp .gt. 0.0) then
        call spray(psmi,mdp,pu,hu,hliq,htank,dchar,rhol,g,ed,
        time,prtsp,mds,mdsu,mdsl,veld,ndrop,norfu)
    1
        if (mds .le. 0.0) mds = mdsi
        mdpump = (mdp + mds)/2.0
        delmd = mdp - mds
        if (abs(delmd) .lt. 0.01) go to 86
        nmdot = nmdot + 1
        if (nmdot .lt. 10) go to 87
        write (6,8) time,delmd
8      format (** pump flow rate does not converge at time = ',
1         f10.2,' sec, delmd = ',f8.4,' lbm/sec ***)
else
    mdsu = 0.0
    mdsl = 0.0
endif
c
heat-transfer rates
c
c wall-to-ullage
c
86 if (mwl .gt. 0.0) qwu = 0.0
if (mwl .le. 0.0) then
twu = (tw + tu)/2.0
    betawu = 1.0/twu
twn = (twu - tusat)/(tcnst - tusat)
call value2(np,nt,pvap,tnrm,sht,pu,twn,cpwu)
call value2(np,nt,pvap,tnrm,dens,pu,twn,rhowu)
call value2(np,nt,pvap,tnrm,cond,pu,twn,kwu)
call value2(np,nt,pvap,tnrm,visc,pu,twun,muwu)
call htc(tw,tu,dtanki,cpuw,rvhowu,betawu,kwu,muwu,g,hwu)
qwu = hwu*awu*(tw - tu)
endif

c ullaegmento-wall liquid
c
if (mwl .le. 0.0) quwl = 0.0
if (mwl .gt. 0.0) then
tuw = (tu + twl)/2.0
  betuw = 1.0/tuw
  call value(nsat,psat,tsat,pu,tusat)
  twln = (tuw - tusat)/(tcnst - tusat)
  call value2(np,nt,pvap,tnrm,shp,pu,tuwln,cpuwl)
  call value2(np,nt,pvap,tnrm,dens,pu,tuwln,rhouwl)
  call value2(np,nt,pvap,tnrm,cond,pu,tuwln,kuw1)
  call value2(np,nt,pvap,tnrm,visc,pu,tuwln,muw1)
  call htc(tuw,twl,dtanld,cpuwl,rhouwl,betuw,kuw1,muw1,g,hwu)
  quwl = huwl*awu*(tu - twl)
endif

c wall-to-wall liquid
c
if (mwl .le. 0.0) qwl = 0.0
if (mwl .gt. 0.0) then
  call value(nsat,tsat,shpf,twl,cpwl)
  call value(nsat,tsat,texpf,twl,betawl)
  call value(nsat,tsat,condf,twl,kwl)
  call value(nsat,tsat,visd,twl,muw1)
  call value(nsat,tsat,densf,twl,rhouwl)
  hwl = 0.13*(3600.0*32.2*g*betawl*abs(tw - twl)*rhouwl**2
   *kwl**2*cpwl/muw1)**(1.0/3.0)
  if (hwl .gt. hwliq) hwl = hwliq
  qwl = hwl*awu*(tw - twl)
endif

c ullaige-to-liquid
c
tul = (tu + tl)/2.0
  betaul = 1.0/tul
tuln = (tul - tusat)/(tcnst - tusat)
  call value2(np,nt,pvap,tnrm,shp,pu,tuln,cpu1)
  call value2(np,nt,pvap,tnrm,dens,pu,tuln,rhoul)
  call value2(np,nt,pvap,tnrm,cond,pu,tuln,kul)
  call value2(np,nt,pvap,tnrm,visc,pu,tuln,muul)
call htc(tu,tl,dtanki,cpul,rhoul,betaul,kul,muul,g,hul)
quI = hul*aul*(tu - tl)
c
ullage-to-droplet
c
td = ts
tud = (tu + td)/2.0
tudn = (tud - tusat)/(tcnst - tusat)
call value2(np,nt,pvap,tnrm,dens,pu,tudn,rhoud)
call value2(np,nt,pvap,tnrm,visc,pu,tudn,muud)
call value2(np,nt,pvap,tnrm,cond,pu,tudn,kud)
qud = 0.0
if (mdp .le. 0.0) go to 17
do io = 1,norfu
    red = rhoud*vcld(io)*ddrop(io)/muud
textbook heat-transfer correlation for liquid droplets (Kreith)
nud = 0.3125*red**0.602
    if (ddrop(io) .gt. 0.0) hud = nud*kud/ddrop(io)
qud = qud + ndrop(io)*hud*ad(io)*(tu - td)
enddo
17 call value(nsat,tsat,densf,td,rhod)
c
ullage-to-spray bar
c
tus = (tu + ts)/2.0
betaus = 1.0/tus
tusn = (tus - tusat)/(tcnst - tusat)
call value2(np,nt,pvap,tnrm,shp,pu,tusn,cpus)
call value2(np,nt,pvap,tnrm,dens,pu,tusn,rhous)
call value2(np,nt,pvap,tnrm,cond,pu,tusn,kus)
call value2(np,nt,pvap,tnrm,visc,pu,tusn,muus)
call htc(tu,ts,dsb,cpus,rhous,betaus,kus,muus,g,hus)
aus = 3.14159*dsb*hu
qus = nbar*hus*aus*(tu - ts)
qu = 0.0
cliquid-to-spray bar
c
tls = (tl + ts)/2.0
call value(nsat,tsat,shpf,tl,beta)
call value(nsat,tsat,densf,tl,rhols)
call value(nsat,tsat,tepf,tl,beta)
call value(nsat,tsat,condf,tl,kls)
call value(nsat,tsat,viscf,tl,muls)
call htc(tl,ts,dsb,cpls,rohs,betals,kls,muls,g,hls)
als = 3.14159*dsb*hliq
qls = nbar*hls*als*(tl - ts)
qls = 0.0

c environment-to-wall
c
qew = qflux*awu
qel = qflux*awl
c
mass-transfer rates
c
droplet-to-ullage boil-off
c
call value(nsat,psat,enthf,pu,hf)
call value(nsat,psat,enthg,pu,hgsat)
hfgu = hgsat - hf
mddu = (qud/3600.0 - mdsu*cpl*(tsat - ts))/hfgu
if (mddu .lt. 0.0) mddu = 0.0
if (mddu .lt. 0.0) qud = 3600.0*mdsu*cpl*(tsat - ts)
if (mddu .lt. 0.0) ts = tsat - qud/(3600.0*mdsu*cpl)
if (mddu .gt. mdsu) mddu = mdsu
if (mddu .gt. mdsu) qud = mdsu*(hfgu + cpl*(tsat - ts))*3600.0

c non-evaporated spray droplet
c
mdsne = mdsu - mddu
if (g .ge. 1.0) then
  mdsw = 0.0
  mdsul = mdsne
else
  mdsw = mdsne
  mdsul = 0.0
endif
c
droplet boil-off from wall
c
call value(nsat,tsat,enthf,tl,hf)
call value(nsat,tsat,enthg,tl,hg)
hfgl = hg - hf
dpudt = pu*(dmudt/mu + dtudt/tu - dvudt/vu)
if (mwl .le. 0.0) mdbw = 0.0
if (mwl .gt. 0.0) then
  if (pu .gt. pwl) mdbw = 0.0
if (pu .lt. pw .and. dpudt .gt. 0.0)
1 mdbw = ((qwl + quwl)/3600.0 - mdsw*cpl*(twl - tsw))/hfgl
if (pu .lt. pw .and. dpudt .le. 0.) then
dtdp = 0.37781 - 4.9170e-3*pwl + 21.7623e-6*pwl*pwl
mdbw = ((qwl + quwl)/3600.0 - mdsw*cpl*(twl - tsw)
1 - mwl*cpl*dtdp*dpudt)/hfgl
endif
endif
mdbwmx = mwl/delt
if (mdbw .ge. mdbwmx)mdbw = mdbwmx


liquid-to-ullage boil-off

if (pu .gt. pl) mdlu = 0.0
if (pu .le. pl .and. dpudt .gt. 0.0)
1 mdlu = (q1/3600.0 - mdsl*cpl*(tl - ts))/hfgl
if (pu .le. pl .and. dpudt .le. 0.0) then
dtdp = 0.37781 - 4.9170e-3*pl + 21.7623e-6*pl*pl
mdlu = (q1/3600.0 - mdsl*cpl*(tl - ts)
1 - mwl*cpl*dtdp*dpudt)/hfgl
endif
endif

condensation

if (tu .gt. tusat) mdu1 = 0.0
if (tu .le. tusat) mdu1 = (qud + qu1 + quwl)/(3600.0*hfgu)
if (he .eq. 0.0 .and. flagl .eq. 1.0)
1 mdcond = xcond*quu/(3600.0*hfgu)
if (flag1 .eq. 0.0) mdcond = 0.0
if (he .eq. 1.0) mdcond = 0.0

rates of change of ullage, wall liquid, and bulk liquid masses

dmudt = mddu + mdbw + mdlu - mdul - mdcond
dmwldt = mdsw - mdbw
dmldt = mdsl + mdl + mcond + mdsul - mdlu - mds - mdv
if (g .ge. 1.0 .and. mwl .ge. mwlmax) dmldt = dmldt + dmwldt

rates of change of ullage, liquid, and wall liquid volumes

call value(nsat,tsat,densf,tl,rhol)
dvldt = dmldt/rhol
if (mwl .le. 0.0) dvwldt = 0.0
if (mwl .gt. 0.0) dvwldt = dmwldt/rhowl
dvudt = -dvldt - dvwldt
rates of change of temperature

ullage

\[ qu = qwu - quwl - quI - qud - quS \]

if (mdcond .eq. 0.0) \[ qu = qwu - quwl - qud - quS \]

enthu = dmu\( dt \)*hgsat

tun = (tu - tusat)/(tcnst - tusat)

call value2(np,nt,pvap,mrm,shv,pu,tun,cvu)

dtudt = \( qu/3600.0 - 144.0/778.0*pu*dvudt + enthu \)

t1 - cvu\( tu*dmudt\)/(mu*cvu)

bulk liquid

\[ qi = qei + qul - qls \]

if (mddu .lt. 0.0) \[ td = ts \]

if (mddu .ge. 0.0) \[ td = tusat \]

if (ml .gt. 0.0) dtldt = \( qV3600.0 - mdlu*hfgl \)

1 + mdsul\( cpl*(td - tl) - mdsI\( cpl*(tl - ts)\)/(ml\( cpl) \)

if (ml .le. 0.0) dtldt = 0.0

wall liquid

\[ qudchk = 3600.0*mdsu\( cpl*(tusat - ts) \]

if (qud .gt. qudchk) \[ tsw = tusat \]

if (qud .le. qudchk) \[ tsw = ts + qud/3600.0/((mdsu + 0.0001)*cpl) \]

if (mwl .gt. 0.0) dtwldt = \( ((qwI + quwl)/3600.0 \)

1 - mdsW\( cpl*(twl - tsw)\)/(mwl\( cpl) \)

if (mwl .le. 0.0) dtwldt = 0.0

tank wall

call vwall(hliq,vw)

mw = rhow*vw
call value(nal,tal,shpal,tw,cpw)
qw = qew - qwu - qwI

dtwdt = qw/3600.0/(mw\( cpw) \)

output listing

if (outp .eq. 0.0) go to 19

if (mod(i,iprint) .eq. 0.0)
1 write (6,7) time,pu,ru,uu,mu,pl,tl,ml,tw,ts,mpump,
2 dppump.dtpump,ho,nds,mdsu,mddu,mdbw,mlu,mdcond

III-69
if (mod(i,iplot) .ne. 0.0) go to 19
j = j + 1

timep(j) = time
pup(j) = pu
tup(j) = tu
vup(j) = vu
mup(j) = mu
twp(j) = tw
plp(j) = pl
tlp(j) = tl
vlp(j) = vl
mlp(j) = ml
pwlp(j) = pwl
twlp(j) = twl
vwlp(j) = vwl
mwlp(j) = mwl
mvp(j) = mv
mdvp(j) = mdv
tsp(j) = ts
mdlup(j) = mdlu
mdsp(j) = mds
mdslp(j) = mdsl
mdsup(j) = mdsu
mddup(j) = mddu
mdbwp(j) = mdbw
mdulp(j) = mdul
mdcndp(j) = mdcond
qwup(j) = qwu
qwlp(j) = qwul
qwlp(j) = qw1
qulp(j) = qul
qudp(j) = qud
qusp(j) = qu5
qlsp(j) = qls
npumpp(j) = npump
dppmp(j) = dppump

19  i = i + 1
    time = time + delt
    if (time .le. fintim) go to 85
    if (ton(ncyc) .eq. 0.0) ncyc = ncyc - 1
    write (6,999)
999  format ('1')
    do i = 1,ncyc
ston = ston + ton(i)
stoff = stoff + toff(i)
stcyc = stcyc + tcyc(i)
if (i .eq. 1) mvnt(i) = mvent(i)
if (i .gt. 1) mvnt(i) = mvent(i) - mvent(i-1)
write (6,1000) i,ton(i),toff(i),tcyc(i),pton(i),mvnt(i),sfc(i)
1000 format (5x,'Cycle No. = ',i12/
   1 5x,'On Time = ',f12.3,' sec'/
   2 5x,'Off Time = ',f12.3,' sec'/
   3 5x,'Cycle Time = ',f12.3,' sec'/
   4 5x,'% On Time = ',f12.3,' %'/
   5 5x,'Vented Mass = ',f12.3,' Ibm'/
   6 5x,'Specific Fuel Consumption = ',f12.3,' Ibm/hr'/)
write (6,1001) pup(i),tup(i),mup(i),plp(i),tlp(i),mlp(i),
   1 twp(i),mdsp(i)
1001 format (5x,'Ullage pressure = ',f12.3,' psia'/
   1 5x,'Ullage temperature = ',f12.3,' R'/
   2 5x,'Ullage mass = ',f12.3,' Ibm'/
   3 5x,'Liquid pressure = ',f12.3,' psia'/
   4 5x,'Liquid temperature = ',f12.3,' R'/
   5 5x,'Liquid mass = ',f12.3,' Ibm'/
   6 5x,'Wall temperature = ',f12.3,' R'/
   7 5x,'Pump flow rate = ',f12.3,' Ibm/sec'/)
enddo
ptonav = 100.0*ston/stcyc
sfcav = 3600.0*mvent(ncyc)/stcyc
write (6,1002) ston,stoff,stcyc,ptonav,mvent(ncyc),sfcav
1002 format (5x,'On Time = ',f12.3,' sec'/
   1 5x,'Off Time = ',f12.3,' sec'/
   2 5x,'Cycle Time = ',f12.3,' sec'/
   3 5x,'% On Time = ',f12.3,' %'/
   4 5x,'Vented Mass = ',f12.3,' Ibm'/
   5 5x,'Specific Fuel Consumption = ',f12.3,' Ibm/hr')
c
c output listing
c   call prtout
c
c output plotting
c   if (opu .eq. 1.0) call crtplt(-11,12,2,00,0,0,subhd,xtitl,
   1 ypu,0.,0.,0.,0.,0.,timep,pup,0,0)
   if (opl .eq. 1.0) call crtplt(12,12,2,00,0,0,subhd,xtitl,
   1 ypl,0.,0.,0.,0.,0.,timep,plp,0,0)
   if (otu .eq. 1.0) call crtplt(21,00,2,00,0,0,subhd,xtitl,
if (otl .eq. 1.0) call crtplt(22,00,2,00,0,0,subhd,xtitl,
ytl,0.,0.,0.,0.,j,timep,tlp,0,0)
if (omu .eq. 1.0) call crtplt(-11,22,2,00,0,0,subhd,xtitl,
ynmu,0.,0.,0.,0.,j,timep,mup,0,0)
if (oml .eq. 1.0) call crtplt(12,00,2,00,0,0,subhd,xtitl,
yml,0.,0.,0.,0.,j,timep,mlp,0,0)
if (omwl .eq. 1.0) call crtplt(21,00,2,00,0,0,subhd,xtitl,
ymwl,0.,0.,0.,0.,j,timep,mwlp,0,0)
if (omdsu .eq. 1.0) call cplt(-11,22,2,00,0,0,subhd,xtitl,
ynmsu,0.,0.,0.,0.,j,timep,mdsup,0,0)
if (omddu .eq. 1.0) call cplt(12,00,2,00,0,0,subhd,xtitl,
ynmddu,0.,0.,0.,0.,j,timep,mddup,0,0)
if (omdbw .eq. 1.0) call cplt(21,00,2,00,0,0,subhd,xtitl,
ynmdbw,0.,0.,0.,0.,j,timep,mdbwp,0,0)
if (omdlu .eq. 1.0) call cplt(22,00,2,0000,jsymb(1),0,subhd,
xnmlup,0.,0.,0.,0.,j,timep,mdlup,0,0)
if (omdul.eq. 1.0) call cplt(00,00,0,0000,jsymb(2),0,0,0,
ynmldulp,0.,0.,0.,0.,j,timep,mdulp,0,0)
if (omdlu .eq. 1.0 .and. omdul.eq. 1.0)
call ckey(2,jsymb,ketitl,-1,-1)
if (omds .eq. 1.0) call cplt(-11,12,2,00,0,0,subhd,xtitl,
ynmms,0.,0.,0.,0.,j,timep,mdsp,0,0)
if (omdv .eq. 1.0) call cplt(12,12,2,00,0,0,subhd,xtitl,
ynmdv,0.,0.,0.,0.,j,timep,mdvp,0,0)
if (onpump.eq. 1.0) call cplt(-11,12,2,00,0,0,subhd,xtitl,
ynpump,0.,0.,0.,0.,j,timep,npumpp,0,0)
if (otw .eq. 1.0) call cplt(21,00,2,00,0,0,subhd,xtitl,
yntw,0.,0.,0.,0.,j,timep,twp,0,0)
if (odppmp .eq. 1.0) call cplt(12,12,2,00,0,0,subhd,xtitl,
ynppmp,0.,0.,0.,0.,j,timep,dppmp,0,0)
stop
end

subroutine area(vl,y,awu,aul,awl,z,hu)
  c subroutine area to calculate the heat transfer areas, and liquid and gas heights of an elliptical bulkhead tank
  c
  real l
  common /tankdim/do,l,ho,t,c1
  common /tankvol/v1,v2,vbi,vbo,vci,vco
  common /tankarea/ab,ac,at
  data tol,nlim/.01,40/
  r = do/2. - t

III-72
\( h = h_o - t \)
\( v_b = v_b i \)
\( v_c = v_c i \)

C liquid level is in upper bulkhead

\[
\begin{align*}
\text{if } (v_l . gt. v_2) \text{ then} & \\
& v = v_l - v_b - v_c \\
& \text{do } i = 1, nlim \\
& \quad f_y = y*y*y - 3.*h*h*y + 3.*v*h*h/(3.14159*r*r) \\
& \quad d f_y = 3.*y*y - 3.*h*h \\
& \quad dely = f_y/df_y \\
& \quad y = y - dely \\
& \quad \text{if (abs(dely) .le. tol) go to 5} \\
& \text{enddo}
\end{align*}
\]

5 \( z = y + h + 1 \)
\( a w_l = a b + 3.14159*(y*(c1*y*y + r*r)**.5 + r*r/c1**.5) \\
2 \quad *(log(y*c1**.5 + (c1*y*y + r*r)**.5) - log(r)) \)
\( r_u l = r*(1. - y/h) \)
\( a u_l = 3.14159*r_u l*r_u l \)

C liquid level is in cylindrical segment

else

\[
\begin{align*}
\text{if (v}_l . gt. v_l) & \\
& z = (v_l - v_b)/(3.14159*r*r) + h \\
& a w_l = a b + 2.*3.14159*r*(z - h) \\
& r_u l = r \quad a u_l = 3.14159*r_u l*r_u l \end{align*}
\]

C liquid level is in lower bulkhead

else

\[
\begin{align*}
\text{if (v}_l . gt. 0.) & \\
& v = v_b - v_l \\
& \text{do } i = 1, nlim \\
& \quad f_y = y*y*y - 3.*h*h*y + 3.*v*h*h/(3.14159*r*r) \\
& \quad d f_y = 3.*y*y - 3.*h*h \\
& \quad dely = f_y/df_y \\
& \quad y = y - dely \\
& \quad \text{if (abs(dely) .le. tol) go to 10} \\
& \text{enddo}
\end{align*}
\]

10 \( z = h - y \)

else

III-73
z = 0.
endif
awl = 3.14159*(h*(c1*h*h + r*r)**.5
+ (h - z)*(c1*(h - z)*(h - z) + r*r)**.5
+ r*r/c1**.5*(log(h*c1**.5 + (c1*h*h + r*r)**.5)
- log((h - z)*c1**.5 + (c1*(h - z)*(h - z) + r*r)**.5)))

erl = z*r/h
aul = 3.14159*rul*rul
endif
endif
awu = at - awl
dul = 2.*rul
hu = 2.*h + 1 - z
return
end

subroutine frict(d,ed,mdot,visc,f)
c
subroutine frict.f to calculate the friction coefficient
c for flow in a pipe
c
real mdot
re = 4.*mdot/(3.14159*visc*d)
if (re .lt. 2300.) f = 64./(re + 1.)
if (re .gt. 2300.)
1 f = 0.25/(log10(ed/3.7 + 2.51/(re*sqrt(.0056 +
2 .5/(re**.32))))**2
return
end

subroutine htc(t1,t2,d,cp,rho,beta,k,mu,a,h)
c
This subroutine computes the free convection heat-transfer
c coefficient for horizontal and vertical surfaces
c
real k,mu,nu
if (d .eq. 0.) h = 0.
if (d .eq. 0.) return
ra = 3600*32.2*a*beta*abs(t1 - t2)*d**3*rho*rho*cp
1 /(mu*k)
u = 0.555*ra**0.25 + 0.447
h = nu*k/d
return
end
subroutine pdrop(mdot,rho,mu,l1,d1,d2,dptot)
c
! subroutine pdrop.f to calculate the pressure drops between the pump outlet and spray manifold inlet
!
real reno(9),kloss1(9),kloss2(9)
real mdot,mu,ll,kb1,kb2,kb3,kb4,kc,kflm
data reno/1.5e+5,2.0e+5,3.0e+5,4.0e+5,6.0e+5,8.0e+5,1.0e+6,
1 2.0e+6,3.0e+6/
data kloss1/0.32,0.26,0.21,0.19,0.173,0.168,0.163,0.160,0.158/
data kloss2/0.20,0.15,0.14,0.128,0.12,0.118,0.117,0.115,0.114/
ed = 1.0e-6
gc = 32.2
a1 = 3.14159*d1*d1/4.0
a2 = 3.14159*d2*d2/4.0

90-degree bend at pump outlet

re1 = 4.0*mdot/(3.14159*mu*d1)
call value(9,reno,kloss1,re1,kb1)

straight section downstream of 90-degree bend

call frict(d1,ed,mdot,mu,f1)
k1 = f1*l1/d1

reducer

kc = 0.5*(1.0 - (d2/d1)**2)

132.5-degree bend

re2 = 4.0*mdot/(3.14159*mu*d2)
call value(9,reno,kloss1,re2,kb2)
kb2 = 1.22*kb2

flowmeter

kflm = 1.308

95.5-degree bend downstream of flowmeter

call value(9,reno,kloss2,re2,kb3)
kb3 = 1.03*kb3

III-75
c 48-degree bend
c
   call value(9,reno,kloss2,re2,kb4)
   kb4 = 0.66*kb4

c pressure drops
c
   const = mdot*mdot/(2.0*rho*gc*144.0)
   dpbl = kbl*const/a1**2
   dpl = k1*const/a1**2
   dpc = kc*const/a2**2
   dpb2 = kb2*const/a2**2
   dpflm = kflm*const/a2**2
   dpb3 = kb3*const/a2**2
   dpb4 = kb4*const/a2**2
   dptot = dpbl + dpl + dpc + dpb2 + dpflm + dpb3 + dpb4
c
   write (6,12) dpbl,dpl,dpc,dpb2,dpflm,dpb3,dpb4,dptot
c12 format (3x,'dpbl = ',f8.5,3x,'dpl = ',f8.5,3x,'dpc = ',f8.5/
      13x,'dpb2 = ',f8.5,3x,'dpflm = ',f8.5,3x,'dpb3 = ',f8.5/
      14x,'dpb4 = ',f8.5,3x,'dptot = ',f8.5)
   return
end

subroutine prtout
c
   c subroutine prtout.f to print the complete output of
   c program tvs.f in a file named outdat
c
   real mu,ml,mwl,mv
   real mdlu,mds,mdsl,mdsu,mddu,mdbw,mdul,mdcond,npump
   common /tankout/nstep,nline,iprint,time(3000),
      1 pu(3000),tu(3000),vu(3000),mu(3000),tw(3000),
      2 pl(3000),tl(3000),vl(3000),ml(3000),
      3 pwl(3000),twl(3000),vwl(3000),mwl(3000),
      4 mv(3000),ts(3000),mdlu(3000),mds(3000),mdsl(3000),
      5 mdsu(3000),mddu(3000),mdbw(3000),mdul(3000),mdcond(3000),
      6 quw(3000),quwl(3000),qwl(3000),qul(3000),qud(3000),
      7 qus(3000),qls(3000),npump(3000),dppump(3000)
   open (unit=15,file='outdat')
   icount = nstep/(nline*iprint) + 1
   do j = 1,icount
      imin = imax + 1
      imax = j*nline*iprint
      if (imax .gt. nstep) imax = nstep
      write (15,1)
      continue
subroutine spray(pman, mdoti, pull, zu, zliq, ztank, dchar, rhol, acc, ed, 
1 time, prtsp, mdot, mdsu, mdsl, veld, ndrop, norfu)}
c subroutine spray.f to model flow in the spray injection tube

c
real rod(12),lod(12),roverd,loverd
real mdin(100),mdout(100),mds(100),as(100),asp(l00)
real vd(100),veld(lOO),ndrop(lOO)
real pin(l00),pout(100),pnode(lOO),ptank(lOO),x(lOO),delpt(200)
real mdot,mdoti,mdotsi,mdoto,mdots,mdsu,mdsl
real kfsm,kbsm,kcsm,kbsi,ks
common /sprayin/dsm,zsm,dsi,zsi,n,nbar,
1 ks,cds,as,vd,roverd,dmdt,tol,nlim,
2 rho,visc
data nb/12/rod/1.,1.5,2.,3.,4.,6.,8.,10.,12.,14.,16.,20./
data lod/20.,14.,12.,12.,14.,17.,24.,30.,34.,38.,42.,50./
psmi = 144.*pman
pu = 144.*pull
dmdot = dmdt
dzsi = zsi/n
asm = 3.14159*dsm*dsm/4.
asi = 3.14159*dsi*dsi/4.
kcsm = .5*(1. - (dsi/dsm)**2)
call value(nb,rod,lod,roverd,loverd)
gc = 32.2
mdot = mdoti
do 10 j = 1,nlim
qsm = (mdot/asm)**2/(2.*rho*gc)
call frict(dsm,ed,mdot,visc,fsm)
kfsm = fsm*zsm/dsm
kbsm = fsm*loverd
dpfsm = qsm*(kfsm + kbsm + kcsm)
phsm = rho*acc*zsm
dpsm = dpfsm + phsm
psmo = psmi - dpsm
mdotsi = mdot/nbar
qi = (mdotsi/asi)**2/(2.*rho*gc)
call frict(dsi,ed,mdotsi,visc,fsi)
kbsi = fsi*loverd
pi = psmo - qi*kbsi
mdsu = 0.
mdsl = 0.
norfu = 0
do i = 1,n
x(i) = i*dzsi - dzsi/2.
aorf = as(i)
z = ztank - zsi + x(i)

III-78
if (z .lt. zu) pt = pu
if (z .ge. zu) pt = pu + rho*acc*(z - zu)
call pres(i,n,pi,po,pn,pt,dpf,ph,mdotsi,mdoto,mdots,
1
dzsi,dsi,ed,asi,aorf,ks,rho,visc,acc)

pin(i) = pi
pout(i) = po
pnode(i) = pn
ptank(i) = pt
mdin(i) = mdotsi
mdout(i) = mdoto
pi = pout(i)
mdotsi = mdout(i)
mds(i) = mdots
if (aorf .gt. 0.0) veld(i) = mdots/(rho*cds*aorf)
if (aorf .le. 0.0) veld(i) = 0.0
if (veld(i) .gt. 0.0)
1 ndrop(i) = nbar*mdots*dchar/(rho*vd(i)*veld(i))
if (veld(i) .le. 0.0) ndrop(i) = 0.0
if (z .lt. zu) norfu = norfu + 1
if (z .lt. zu) mdsu = mdsu + mdots
if (z .ge. zu) mdsl = mdsl + mdots

enddo
mdsu = nbar*mdsu
mdsl = nbar*mdsl
ptcal = pn - ks/(2.*rho*gc)*(mdots/aorf)**2
delpt(j) = ptcal - pt
dpt = delpt(j)/144.0
if (abs(dpt) .lt. tol) go to 15
if (j .eq. 1) then
   if (delpt(j) .lt. 0.) mdot = mdot - dmdot
   if (delpt(j) .gt. 0.) mdot = mdot + dmdot
else
   prod = delpt(j)*delpt(j-1)
   if (delpt(j) .lt. 0. and. prod .gt. 0.) mdot = mdot - dmdot
   if (delpt(j) .gt. 0. and. prod .gt. 0.) mdot = mdot + dmdot
   if (delpt(j) .lt. 0. and. prod .lt. 0.) then
      dmdot = dmdot/2.
      mdot = mdot - dmdot
   endif
   if (delpt(j) .gt. 0. and. prod .lt. 0.) then
      dmdot = dmdot/2.
      mdot = mdot + dmdot
   endif
endif
10 continue
if (abs(dpt) .gt. tol) write (6,1002) time,dpt

1002 format ('*** tank pressure does not converge at time = ',f10.2,
1 ' sec;',' delpt = ','e9.3,' psi ***')

if (time .ge. prtsp .and. time .le. (prtsp + 0.1)) then
  dppump = (psmi - pt)/144.
  dpsm = dpsm/144.
  dpsi = (pi - pt)/144.
  hliq = 12.*zliq
  write (6,1)
1 format ('SPRAY MANIFOLD/INJECTION TUBE FLOW MODEL/')
  write (6,2) acc,pull,pman,hliq,rho,visc
2 format ('Acceleration Level ',f6.1,' g'/
1 5x,'Ullage Pressure ','f6.3,' psia'/
2 5x,'Spray Manifold Inlet Pressure ','f6.3,' psia'/
3 5x,'Liquid Level ','f6.1,' in'/
4 5x,'Liquid Density ','f6.3,' lbm/ft3'/
5 5x,'Liquid Viscosity ','e9.3,' lbm/ft-sec'/)
  write (6,3) mdot,dppump,dpsm,dpsi
3 format ('Pump Flow Rate ','f6.3,' lbm/sec'/
1 5x,'Pump Pressure Rise ','f6.3,' psi'/
2 5x,'Spray Manifold Tube Delta p ','f6.3,' psi'/
3 5x,'Spray Injection Tube Delta p ','f6.3,' psi'/)
  write (6,4)
4 format ('Node',2x,'Distance',3x,'Inlet',2x,'Outlet',
1 3x,'Nodal',3x,'Tank',5x,'Inlet',4x,'Outlet',
2 3x,'Injection',3x,'Injection',3x,'Orifice')
  write (6,5)
5 format (24x,'p',7x,'p',7x,'p',7x,'p',7x,'p',
1 7x,'mdot',5x,'mdot',7x,'mdot',6x,'Velocity',
2 5x,'CdA')
  write (6,6)
6 format (13x,'(in)',4x,'(psia)',2x,'(psia)',2x,'(psia)',2x,'(psia)',
1 2x,'(lbm/sec)',1x,'(lbm/sec)',1x,'(lbm/sec)',5x,'(fps)',
2 6x,'(in2)/')
  do i = 1,n
    pin(i) = pin(i)/144.
    pout(i) = pout(i)/144.
    pnode(i) = pnode(i)/144.
    ptank(i) = ptank(i)/144.
    asp(i) = 144.*as(i)
    x(i) = 12.*x(i)
    write (6,7) i,x(i),pin(i),pout(i),pnode(i),ptank(i),
1 mdin(i),mdout(i),mds(i),veld(i),asp(i)
7 format (5x,i3,4x,f6.2,3x,4(f6.3,2x),
1 3(e9.3,1x),4x,f5.2,4x,f8.6)
This subroutine performs Lagrangian interpolation within a set of (x,y) pairs to give yout corresponding to xin.

dimension x(np), y(np)
if (xin .le. x(1)) yout = y(1)
if (xin .le. x(1)) return
if (xin .ge. x(np)) yout = y(np)
if (xin .ge. x(np)) return
do 10 i = 1, np
   k = i
   if (xin .lt. x(i)) go to 30
10 continue
30 ffr = (xin - x(k - 1))/(x(k) - x(k - 1))
yout = y(k - 1) + ffr*(y(k) - y(k - 1))
return

SUBROUTINE VALUE2(NPX,NPY,X,Y,Z,XIN,YIN,ZOUT)
DIMENSION X(NPX), Y(NPY), Z(NPX,NPY)
IF(XIN .LE. X(1) .AND. YIN .LE. Y(1)) ZOUT=Z(1,1)
IF(XIN .LE. X(1) .AND. YIN .LE. Y(1)) RETURN
IF(XIN .GE. X(NPX) .AND. YIN .LE. Y(1)) ZOUT=Z(NPX,1)
IF(XIN .GE. X(NPX) .AND. YIN .LE. Y(1)) RETURN
IF(XIN .LE. X(1) .AND. YIN .GE. Y(NPY)) ZOUT=Z(1,NPY)
IF(XIN .LE. X(1) .AND. YIN .GE. Y(NPY)) RETURN
IF(XIN .GE. X(NPX) .AND. YIN .GE. Y(NPY)) ZOUT=Z(NPX,NPY)
IF(XIN .GE. X(NPX) .AND. YIN .GE. Y(NPY)) RETURN
IF(XIN .GT. X(1)) GO TO 30
DO 20 I=1, NPY
   M=I
   IF(YIN .LT. Y(I)) GO TO 25
20 CONTINUE
25 FFRY=(YIN-Y(M-1))/(Y(M)-Y(M-1))
ZOUT = Z(1,M-1) + FFRY*(Z(1,M)-Z(1,M-1))
RETURN
C
30 IF(XIN .LT. X(NPX)) GO TO 60
   DO 50 I = 1, NPY
      M = I
      IF(YIN .LT. Y(I)) GO TO 55
   50 CONTINUE
55 FFRY = (YIN-Y(M-1))/(Y(M)-Y(M-1))
   ZOUT = Z(NPX,M-1) + FFRY*(Z(NPX,M)-Z(NPX,M-1))
   RETURN
C
60 IF(YIN .GT. Y(1)) GO TO 90
   DO 80 I = 1, NPX
      L = I
      IF(XIN .LT. X(I)) GO TO 85
   80 CONTINUE
85 FFRX = (XIN-X(L-1))/(X(L)-X(L-1))
   ZOUT = Z(L-1,1) + FFRX*(Z(L,1)-Z(L-1,1))
   RETURN
C
90 IF(YIN .LT. Y(NPY)) GO TO 120
   DO 110 I = 1, NPX
      L = I
      IF(XIN .LT. X(I)) GO TO 115
   110 CONTINUE
115 FFRX = (XIN-X(L-1))/(X(L)-X(L-1))
   ZOUT = Z(L-1,NPY) + FFRX*(Z(L,NPY)-Z(L-1,NPY))
   RETURN
C
120 DO 130 I = 1, NPX
      L = I
      IF(XIN .LT. X(I)) GO TO 135
130 CONTINUE
135 FXR = (XIN-X(L-1))/(X(L)-X(L-1))
   DO 140 I = 1, NPY
      M = I
      IF(YIN .LT. Y(I)) GO TO 145
140 CONTINUE
145 FYR = (YIN-Y(M-1))/(Y(M)-Y(M-1))
C
ZXLO = Z(L-1,M-1) + FYR*(Z(L-1,M)-Z(L-1,M-1))
ZXHI = Z(L,M-1) + FYR*(Z(L,M)-Z(L,M-1))
C
ZOUT = ZXLO + FXR*(ZXHI-ZXLO)
subroutine volarea(vt)

c subroutine volarea.f to calculate the volumes and areas of
an elliptical bulkhead tank

c real l
common /tankdim/do,l,ho,t,c1
common /tankvol/v1,v2,vbi,vbo,vci,vco
common /tankarea/abi,aci,ati
ro = do/2.
ri = ro - t
hi = ho - t
xkl = ri/hi
c1 = xkl**4 - xkl*xkl

c internal bulkhead

c vbi = 2./3.*3.14159*ri*ri*hi
abi = 3.14159*(hi*(c1*hi*hi + ri*ri)**.5
1 + ri*ri/c1**.5*(log(hi*c1**.5
2 + (c1*hi*hi + ri*ri)**.5) - log(ri)))

c external bulkhead

c vbo = 2./3.*3.14159*ro*ro*ho
abo = 3.14159*(ho*(c1*ho*ho + ro*ro)**.5
1 + ro*ro/c1**.5*(log(ho*c1**.5
2 + (c1*ho*ho + ro*ro)**.5) - log(ro)))

c internal cylinder

c vci = 3.14159*ri*ri*l
aci = 2.*3.14159*ri*l

c external cylinder

c vco = 3.14159*ro*ro*l
aco = 2.*3.14159*ro*l

c tank

c v1 = vbi
v2 = vbi + vci
vt = 2.*vbi + vci
ati = 2.*abi + aci
returnend

subroutine vwall(hliq,vtw)
c subroutine vwall.f to calculate the wall volume exposed to
c the ullage gas
c
real l
common /tankdim/do,l,ho,t,c1
common /tankvol/v1,v2,vbi,vbo,vci,vco
ro = do/2.
ri = ro - t
hi = ho - t
vbw = vbo - vbi
vcw = vco - vci
if (hliq .gt. (hi + l)) then
  h = hliq - (hi + l)
  vi = 3.14159*h*(ri/hi)**2*(hi*hi - h*h/3.)
  vo = 3.14159*h*(ro/ho)**2*(ho*ho - h*h/3.)
  v = vo - vi
  vtw = vbw - v
else
  if (hliq .gt. hi) then
    h = hi + 1 - hliq
    vi = 3.14159*ri*ri*h
    vo = 3.14159*ro*ro*h
    v = vo - vi
    vtw = vbw + v
  else
    h = hi - hliq
    vi = 3.14159*h*(ri/hi)**2*(hi*hi - h*h/3.)
    vo = 3.14159*h*(ro/ho)**2*(ho*ho - h*h/3.)
    v = vo - vi
    vtw = vbw + vcw + v
  endif
endif
return
end
### 3.5 Input Data

#### 3.5.1 Heat Exchanger Model

```plaintext
*********|*********|*********|*********|*********
MASSIC 1 PIC 1 PSTI I MDOT(VNT)1 PBACK 1 Program is vent2
*********|*********|*********|*********|*********
270.0   20.590  20.645  0.00475  19.5
*********|*********|*********|*********|*********
PROP 1   PTP 1  AVLIQ 1 PAMB 1 G 1
*********|*********|*********|*********|*********
1.0      1.021   3500.0  0.0    0.0
*********|*********|*********|*********|*********
M 1 VT 1 AX 1 VA 1 Twall 1
*********|*********|*********|*********|*********
25 68.9342  0.04500  0  32.00
*********|*********|*********|*********|*********
MDOT(sp)1 DI (sp) 1 THKNESS 1 VFLW DI 1 E/D 1
*********|*********|*********|*********|*********
0.32     1.18    0.035   0.25  1.0e-6
*********|*********|*********|*********|*********
DPINC 1  ERRMX 1 PERRMX 1 DEBUG 1 EQ DIAM 1
*********|*********|*********|*********|*********
0.005    0.005   0.0010  0.0  0.134
*********|*********|*********|*********|*********
FINTIM 1 PRDEL 1 QDTERR 1 DELT 1 DELPTP 1
*********|*********|*********|*********|*********
2.2      0.02    0.03    0.01  0.08
*********|*********|*********|*********|*********
OT 1 OM 1 OP 1 DEL Qdotl 1
*********|*********|*********|*********|*********
0 0 0 0 0.0010
*********|*********|*********|*********|*********
A(1-M) 1 DH(1-M) 1 QDOT(1-M) 1 LENGTH 1
*********|*********|*********|*********|*********
0.04500  0.0    1.00   0.0  
0.27721  0.0    1.00   6.0  
0.27721  0.0    0.975  6.0  
0.27721  0.0    0.95   6.0  
0.27721  0.0    0.925  6.0  
0.27721  0.0    0.90   6.0  
0.27721  0.0    0.875  6.0  
0.27721  0.0    0.85   6.0  
0.27721  0.0    0.825  6.0  
0.27721  0.0    0.80   6.0  
```

III-85
Simulation of LH2 Vent Thru Zero g Vent System Heat Exchanger (4/6/93)
Baseline Vent Area of 0.00372 in² (Mdot & Qdot Trade) \( \delta q = 0.0010 \)

<table>
<thead>
<tr>
<th>( \text{LMXPAR} )</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>( -2.0 )</td>
<td>2.107</td>
</tr>
<tr>
<td>( -1.699 )</td>
<td>1.835</td>
</tr>
<tr>
<td>( -1.3979 )</td>
<td>1.585</td>
</tr>
<tr>
<td>( -1.1549 )</td>
<td>1.3874</td>
</tr>
<tr>
<td>( -1.0 )</td>
<td>1.2672</td>
</tr>
<tr>
<td>( -0.6990 )</td>
<td>1.0492</td>
</tr>
<tr>
<td>( -0.3979 )</td>
<td>0.8482</td>
</tr>
<tr>
<td>( -0.1549 )</td>
<td>0.7024</td>
</tr>
<tr>
<td>0.0</td>
<td>0.6232</td>
</tr>
<tr>
<td>0.3010</td>
<td>0.4914</td>
</tr>
<tr>
<td>0.6020</td>
<td>0.3766</td>
</tr>
<tr>
<td>0.8451</td>
<td>0.2923</td>
</tr>
<tr>
<td>1.0</td>
<td>0.2430</td>
</tr>
<tr>
<td>1.301</td>
<td>0.1703</td>
</tr>
<tr>
<td>1.602</td>
<td>0.1106</td>
</tr>
<tr>
<td>1.8451</td>
<td>0.0682</td>
</tr>
<tr>
<td>2.0</td>
<td>0.0453</td>
</tr>
<tr>
<td>10.0</td>
<td>0.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( \text{LVISP} )</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>1.985e-5</td>
</tr>
<tr>
<td>1.021</td>
<td>1.751e-5</td>
</tr>
<tr>
<td>Value</td>
<td>GVISP</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>2.553</td>
<td>1.4e-5</td>
</tr>
<tr>
<td>4.17</td>
<td>1.24e-5</td>
</tr>
<tr>
<td>6.446</td>
<td>1.11e-5</td>
</tr>
<tr>
<td>9.527</td>
<td>1.006e-5</td>
</tr>
<tr>
<td>13.561</td>
<td>0.917e-5</td>
</tr>
<tr>
<td>15.984</td>
<td>0.877e-5</td>
</tr>
<tr>
<td>18.694</td>
<td>0.84e-5</td>
</tr>
<tr>
<td>21.723</td>
<td>0.805e-5</td>
</tr>
<tr>
<td>25.089</td>
<td>0.773e-5</td>
</tr>
<tr>
<td>28.813</td>
<td>0.742e-5</td>
</tr>
<tr>
<td>32.915</td>
<td>0.713e-5</td>
</tr>
<tr>
<td>37.415</td>
<td>0.685e-5</td>
</tr>
<tr>
<td>42.334</td>
<td>0.659e-5</td>
</tr>
<tr>
<td>47.693</td>
<td>0.634e-5</td>
</tr>
<tr>
<td>53.514</td>
<td>0.609e-5</td>
</tr>
<tr>
<td>59.817</td>
<td>0.586e-5</td>
</tr>
<tr>
<td>66.625</td>
<td>0.563e-5</td>
</tr>
<tr>
<td>73.957</td>
<td>0.54e-5</td>
</tr>
<tr>
<td>81.838</td>
<td>0.519e-5</td>
</tr>
<tr>
<td>90.287</td>
<td>0.497e-5</td>
</tr>
<tr>
<td>99.329</td>
<td>0.476e-5</td>
</tr>
<tr>
<td>108.987</td>
<td>0.454e-5</td>
</tr>
<tr>
<td>119.297</td>
<td>0.433e-5</td>
</tr>
<tr>
<td>130.299</td>
<td>0.41e-5</td>
</tr>
<tr>
<td>142.027</td>
<td>0.387e-5</td>
</tr>
<tr>
<td>154.522</td>
<td>0.362e-5</td>
</tr>
<tr>
<td>167.848</td>
<td>0.333e-5</td>
</tr>
<tr>
<td>182.136</td>
<td>0.292e-5</td>
</tr>
<tr>
<td>187.510</td>
<td>0.24e-5</td>
</tr>
</tbody>
</table>

GVISP
24
0.0   0.045e-5
1.021 0.05e-5
2.553 0.057e-5
9.527 0.07e-5
18.694 0.079e-5
32.915 0.089e-5
37.415 0.092e-5
42.334 0.095e-5
47.693 0.097e-5
53.514 0.1e-5
59.817 0.103e-5
66.625 0.106e-5
73.957 0.109e-5
<table>
<thead>
<tr>
<th>( f^*(13/-88) )</th>
<th>( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>187.51</td>
<td>0.24e-5</td>
</tr>
<tr>
<td>182.136</td>
<td>0.196e-5</td>
</tr>
<tr>
<td>172.327</td>
<td>0.183e-5</td>
</tr>
<tr>
<td>167.848</td>
<td>0.168e-5</td>
</tr>
<tr>
<td>163.127</td>
<td>0.159e-5</td>
</tr>
<tr>
<td>159.256</td>
<td>0.149e-5</td>
</tr>
<tr>
<td>142.027</td>
<td>0.139e-5</td>
</tr>
<tr>
<td>130.299</td>
<td>0.135e-5</td>
</tr>
<tr>
<td>122.329</td>
<td>0.127e-5</td>
</tr>
<tr>
<td>111.979</td>
<td>0.118e-5</td>
</tr>
<tr>
<td>108.987</td>
<td>0.115e-5</td>
</tr>
<tr>
<td>99.329</td>
<td>0.12e-5</td>
</tr>
<tr>
<td>90.287</td>
<td>0.116e-5</td>
</tr>
<tr>
<td>81.838</td>
<td>0.112e-5</td>
</tr>
</tbody>
</table>

**H2 RHO DATA**

<table>
<thead>
<tr>
<th>( LRHOP )</th>
<th>( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>5.385</td>
</tr>
<tr>
<td>31</td>
<td>5.385</td>
</tr>
<tr>
<td>32</td>
<td>4.80827</td>
</tr>
<tr>
<td>33</td>
<td>4.80377</td>
</tr>
<tr>
<td>34</td>
<td>4.77434</td>
</tr>
<tr>
<td>35</td>
<td>4.74423</td>
</tr>
<tr>
<td>36</td>
<td>4.71359</td>
</tr>
<tr>
<td>37</td>
<td>4.68219</td>
</tr>
<tr>
<td>38</td>
<td>4.65003</td>
</tr>
<tr>
<td>39</td>
<td>4.61705</td>
</tr>
<tr>
<td>40</td>
<td>4.58321</td>
</tr>
<tr>
<td>41</td>
<td>4.54844</td>
</tr>
<tr>
<td>42</td>
<td>4.51266</td>
</tr>
<tr>
<td>43</td>
<td>4.47582</td>
</tr>
<tr>
<td>44</td>
<td>4.43782</td>
</tr>
<tr>
<td>45</td>
<td>4.39858</td>
</tr>
<tr>
<td>46</td>
<td>4.35801</td>
</tr>
<tr>
<td>47</td>
<td>4.31600</td>
</tr>
<tr>
<td>48</td>
<td>4.27243</td>
</tr>
<tr>
<td>49</td>
<td>4.22718</td>
</tr>
<tr>
<td>50</td>
<td>4.18010</td>
</tr>
<tr>
<td>51</td>
<td>4.13102</td>
</tr>
<tr>
<td>52</td>
<td>4.07975</td>
</tr>
<tr>
<td>53</td>
<td>4.02608</td>
</tr>
<tr>
<td>54</td>
<td>3.96975</td>
</tr>
<tr>
<td>55</td>
<td>3.9105</td>
</tr>
<tr>
<td>56</td>
<td>3.8479</td>
</tr>
<tr>
<td>57</td>
<td>3.7815</td>
</tr>
<tr>
<td>58</td>
<td>3.7108</td>
</tr>
<tr>
<td>59</td>
<td>3.635</td>
</tr>
</tbody>
</table>

**III-88**
<table>
<thead>
<tr>
<th>Value</th>
<th>GRHOP</th>
</tr>
</thead>
<tbody>
<tr>
<td>99.33</td>
<td>3.5534</td>
</tr>
<tr>
<td>108.987</td>
<td>3.4646</td>
</tr>
<tr>
<td>119.297</td>
<td>3.3668</td>
</tr>
<tr>
<td>130.299</td>
<td>3.2572</td>
</tr>
<tr>
<td>142.027</td>
<td>3.1314</td>
</tr>
<tr>
<td>154.522</td>
<td>2.9808</td>
</tr>
<tr>
<td>167.848</td>
<td>2.7851</td>
</tr>
<tr>
<td>182.136</td>
<td>2.4567</td>
</tr>
<tr>
<td>187.51</td>
<td>1.9620</td>
</tr>
</tbody>
</table>

GRHOP
40
<table>
<thead>
<tr>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0325</td>
<td>0.000345</td>
</tr>
<tr>
<td>0.1137</td>
<td>0.00108</td>
</tr>
<tr>
<td>0.3163</td>
<td>0.002742</td>
</tr>
<tr>
<td>1.021</td>
<td>0.00784</td>
</tr>
<tr>
<td>1.073</td>
<td>0.00819</td>
</tr>
<tr>
<td>1.462</td>
<td>0.01077</td>
</tr>
<tr>
<td>1.950</td>
<td>0.01391</td>
</tr>
<tr>
<td>2.553</td>
<td>0.01765</td>
</tr>
<tr>
<td>3.288</td>
<td>0.02207</td>
</tr>
<tr>
<td>4.170</td>
<td>0.02724</td>
</tr>
<tr>
<td>5.217</td>
<td>0.03322</td>
</tr>
<tr>
<td>6.446</td>
<td>0.04008</td>
</tr>
<tr>
<td>7.877</td>
<td>0.04790</td>
</tr>
<tr>
<td>9.527</td>
<td>0.05675</td>
</tr>
<tr>
<td>11.416</td>
<td>0.06671</td>
</tr>
<tr>
<td>13.561</td>
<td>0.07787</td>
</tr>
<tr>
<td>15.984</td>
<td>0.09011</td>
</tr>
<tr>
<td>18.694</td>
<td>0.10395</td>
</tr>
<tr>
<td>21.723</td>
<td>0.11930</td>
</tr>
<tr>
<td>25.089</td>
<td>0.13629</td>
</tr>
<tr>
<td>28.813</td>
<td>0.15504</td>
</tr>
<tr>
<td>32.915</td>
<td>0.17568</td>
</tr>
<tr>
<td>37.415</td>
<td>0.19839</td>
</tr>
<tr>
<td>42.334</td>
<td>0.22335</td>
</tr>
<tr>
<td>47.693</td>
<td>0.25078</td>
</tr>
<tr>
<td>53.514</td>
<td>0.28094</td>
</tr>
<tr>
<td>59.817</td>
<td>0.3141</td>
</tr>
<tr>
<td>66.625</td>
<td>0.35076</td>
</tr>
<tr>
<td>73.957</td>
<td>0.39126</td>
</tr>
<tr>
<td>81.838</td>
<td>0.43621</td>
</tr>
<tr>
<td>90.287</td>
<td>0.48635</td>
</tr>
<tr>
<td>99.329</td>
<td>0.54266</td>
</tr>
<tr>
<td>108.987</td>
<td>0.60647</td>
</tr>
<tr>
<td>RHOP</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
</tr>
<tr>
<td>119.297</td>
<td>0.67967</td>
</tr>
<tr>
<td>130.299</td>
<td>0.76511</td>
</tr>
<tr>
<td>142.027</td>
<td>0.86751</td>
</tr>
<tr>
<td>154.522</td>
<td>0.99566</td>
</tr>
<tr>
<td>167.848</td>
<td>1.17023</td>
</tr>
<tr>
<td>182.136</td>
<td>1.4779</td>
</tr>
<tr>
<td>187.510</td>
<td>1.96202</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SCLENT</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>34.0</td>
<td>-115.878</td>
<td></td>
</tr>
<tr>
<td>34.263</td>
<td>-115.318</td>
<td></td>
</tr>
<tr>
<td>36.483</td>
<td>-110.59</td>
<td></td>
</tr>
<tr>
<td>14.7</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>34.0</td>
<td>-115.740</td>
<td></td>
</tr>
<tr>
<td>36.0</td>
<td>-111.351</td>
<td></td>
</tr>
<tr>
<td>36.483</td>
<td>-110.240</td>
<td></td>
</tr>
<tr>
<td>36.608</td>
<td>-109.95</td>
<td></td>
</tr>
<tr>
<td>15.0</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>34.0</td>
<td>-115.731</td>
<td></td>
</tr>
<tr>
<td>36.0</td>
<td>-111.343</td>
<td></td>
</tr>
</tbody>
</table>

III-90
<table>
<thead>
<tr>
<th>Temperature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>36.608</td>
<td>-109.942</td>
</tr>
<tr>
<td>38.444</td>
<td>-105.71</td>
</tr>
<tr>
<td>20.0</td>
<td>5</td>
</tr>
<tr>
<td>34.0</td>
<td>-115.583</td>
</tr>
<tr>
<td>36.0</td>
<td>-111.201</td>
</tr>
<tr>
<td>38.0</td>
<td>-106.489</td>
</tr>
<tr>
<td>38.444</td>
<td>-105.395</td>
</tr>
<tr>
<td>39.975</td>
<td>-101.62</td>
</tr>
<tr>
<td>25.0</td>
<td>5</td>
</tr>
<tr>
<td>34.0</td>
<td>-115.435</td>
</tr>
<tr>
<td>36.0</td>
<td>-111.059</td>
</tr>
<tr>
<td>38.0</td>
<td>-106.356</td>
</tr>
<tr>
<td>39.975</td>
<td>-101.358</td>
</tr>
<tr>
<td>41.299</td>
<td>-98.01</td>
</tr>
<tr>
<td>30.0</td>
<td>6</td>
</tr>
<tr>
<td>34.0</td>
<td>-115.287</td>
</tr>
<tr>
<td>36.0</td>
<td>-110.917</td>
</tr>
<tr>
<td>38.0</td>
<td>-106.223</td>
</tr>
<tr>
<td>40.0</td>
<td>-101.170</td>
</tr>
<tr>
<td>41.299</td>
<td>-97.667</td>
</tr>
<tr>
<td>42.475</td>
<td>-94.5</td>
</tr>
<tr>
<td>35.0</td>
<td>7</td>
</tr>
<tr>
<td>34.0</td>
<td>-115.139</td>
</tr>
<tr>
<td>36.0</td>
<td>-110.775</td>
</tr>
<tr>
<td>38.0</td>
<td>-106.090</td>
</tr>
<tr>
<td>40.0</td>
<td>-101.052</td>
</tr>
<tr>
<td>42.0</td>
<td>-95.584</td>
</tr>
<tr>
<td>42.475</td>
<td>-94.227</td>
</tr>
<tr>
<td>43.536</td>
<td>-91.2</td>
</tr>
<tr>
<td>40.0</td>
<td>7</td>
</tr>
<tr>
<td>34.0</td>
<td>-114.991</td>
</tr>
<tr>
<td>36.0</td>
<td>-110.633</td>
</tr>
<tr>
<td>38.0</td>
<td>-105.957</td>
</tr>
<tr>
<td>40.0</td>
<td>-100.929</td>
</tr>
<tr>
<td>42.0</td>
<td>-95.477</td>
</tr>
<tr>
<td>43.536</td>
<td>-90.977</td>
</tr>
<tr>
<td>45.406</td>
<td>-85.50</td>
</tr>
<tr>
<td>45.0</td>
<td>7</td>
</tr>
<tr>
<td>34.0</td>
<td>-114.842</td>
</tr>
<tr>
<td>36.0</td>
<td>-110.491</td>
</tr>
<tr>
<td>38.0</td>
<td>-105.823</td>
</tr>
<tr>
<td>40.0</td>
<td>-100.806</td>
</tr>
<tr>
<td>42.0</td>
<td>-95.370</td>
</tr>
<tr>
<td>44.0</td>
<td>-89.454</td>
</tr>
<tr>
<td>45.406</td>
<td>-85.08</td>
</tr>
<tr>
<td>Pressure</td>
<td>Cp</td>
</tr>
<tr>
<td>----------</td>
<td>----</td>
</tr>
<tr>
<td>16</td>
<td>1.021</td>
</tr>
<tr>
<td></td>
<td>1.462</td>
</tr>
<tr>
<td></td>
<td>2.553</td>
</tr>
<tr>
<td></td>
<td>4.17</td>
</tr>
<tr>
<td></td>
<td>6.446</td>
</tr>
<tr>
<td></td>
<td>9.527</td>
</tr>
<tr>
<td></td>
<td>13.561</td>
</tr>
<tr>
<td></td>
<td>18.694</td>
</tr>
<tr>
<td></td>
<td>25.089</td>
</tr>
<tr>
<td></td>
<td>28.813</td>
</tr>
<tr>
<td></td>
<td>32.915</td>
</tr>
<tr>
<td></td>
<td>42.334</td>
</tr>
<tr>
<td></td>
<td>53.514</td>
</tr>
<tr>
<td></td>
<td>66.625</td>
</tr>
<tr>
<td></td>
<td>81.838</td>
</tr>
<tr>
<td></td>
<td>99.329</td>
</tr>
</tbody>
</table>

**GH2 PROPERTIES**

<table>
<thead>
<tr>
<th>Cp</th>
<th><em>Thermal Cond</em></th>
<th>Viscosity</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>14</td>
<td>50.0</td>
</tr>
<tr>
<td>1.0</td>
<td>2.511</td>
<td>7.21e-3</td>
</tr>
<tr>
<td></td>
<td>2.506</td>
<td>7.35e-3</td>
</tr>
<tr>
<td></td>
<td>2.499</td>
<td>7.63e-3</td>
</tr>
<tr>
<td></td>
<td>2.493</td>
<td>7.92e-3</td>
</tr>
<tr>
<td></td>
<td>2.489</td>
<td>8.24e-3</td>
</tr>
<tr>
<td></td>
<td>2.486</td>
<td>8.56e-3</td>
</tr>
<tr>
<td></td>
<td>2.483</td>
<td>8.96e-3</td>
</tr>
<tr>
<td></td>
<td>2.481</td>
<td>9.47e-3</td>
</tr>
<tr>
<td></td>
<td>2.479</td>
<td>9.98e-3</td>
</tr>
<tr>
<td></td>
<td>2.478</td>
<td>1.049e-2</td>
</tr>
<tr>
<td></td>
<td>2.477</td>
<td>1.10e-2</td>
</tr>
<tr>
<td></td>
<td>2.475</td>
<td>1.151e-2</td>
</tr>
<tr>
<td></td>
<td>2.474</td>
<td>1.202e-2</td>
</tr>
</tbody>
</table>

III-92
<table>
<thead>
<tr>
<th>V1</th>
<th>V2</th>
<th>V3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.474</td>
<td>1.252e-2</td>
<td>1.0e-6</td>
</tr>
<tr>
<td>5.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.652</td>
<td>6.33e-3</td>
<td>6.3e-7</td>
</tr>
<tr>
<td>2.636</td>
<td>7.815e-3</td>
<td>6.43e-7</td>
</tr>
<tr>
<td>2.610</td>
<td>8.707e-3</td>
<td>6.72e-7</td>
</tr>
<tr>
<td>2.589</td>
<td>8.879e-3</td>
<td>7.03e-7</td>
</tr>
<tr>
<td>2.574</td>
<td>9.174e-3</td>
<td>7.35e-7</td>
</tr>
<tr>
<td>2.561</td>
<td>9.550e-3</td>
<td>7.73e-7</td>
</tr>
<tr>
<td>2.551</td>
<td>9.933e-3</td>
<td>8.05e-7</td>
</tr>
<tr>
<td>2.543</td>
<td>1.032e-2</td>
<td>8.32e-7</td>
</tr>
<tr>
<td>2.536</td>
<td>1.070e-2</td>
<td>8.56e-7</td>
</tr>
<tr>
<td>2.524</td>
<td>1.109e-2</td>
<td>8.87e-7</td>
</tr>
<tr>
<td>2.524</td>
<td>1.148e-2</td>
<td>9.18e-7</td>
</tr>
<tr>
<td>2.52</td>
<td>1.186e-2</td>
<td>9.49e-7</td>
</tr>
<tr>
<td>2.516</td>
<td>1.224e-2</td>
<td>9.79e-7</td>
</tr>
<tr>
<td>2.512</td>
<td>1.262e-2</td>
<td>1.01e-6</td>
</tr>
<tr>
<td>10.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.790</td>
<td>7.04e-3</td>
<td>7.1e-7</td>
</tr>
<tr>
<td>2.768</td>
<td>7.892e-3</td>
<td>7.21e-7</td>
</tr>
<tr>
<td>2.726</td>
<td>9.425e-3</td>
<td>7.43e-7</td>
</tr>
<tr>
<td>2.696</td>
<td>9.714e-3</td>
<td>7.68e-7</td>
</tr>
<tr>
<td>2.671</td>
<td>1.001e-2</td>
<td>7.93e-7</td>
</tr>
<tr>
<td>2.649</td>
<td>1.030e-2</td>
<td>8.19e-7</td>
</tr>
<tr>
<td>2.633</td>
<td>1.061e-2</td>
<td>8.38e-7</td>
</tr>
<tr>
<td>2.619</td>
<td>1.092e-2</td>
<td>8.61e-7</td>
</tr>
<tr>
<td>2.607</td>
<td>1.122e-2</td>
<td>8.93e-7</td>
</tr>
<tr>
<td>2.596</td>
<td>1.153e-2</td>
<td>9.19e-7</td>
</tr>
<tr>
<td>2.587</td>
<td>1.183e-2</td>
<td>9.43e-7</td>
</tr>
<tr>
<td>2.578</td>
<td>1.214e-2</td>
<td>9.62e-7</td>
</tr>
<tr>
<td>2.571</td>
<td>1.245e-2</td>
<td>9.85e-7</td>
</tr>
<tr>
<td>2.564</td>
<td>1.276e-2</td>
<td>1.01e-6</td>
</tr>
<tr>
<td>15.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.912</td>
<td>9.84e-3</td>
<td>7.6e-7</td>
</tr>
<tr>
<td>2.886</td>
<td>9.95e-3</td>
<td>7.72e-7</td>
</tr>
<tr>
<td>2.837</td>
<td>1.017e-2</td>
<td>7.94e-7</td>
</tr>
<tr>
<td>2.800</td>
<td>1.040e-2</td>
<td>8.16e-7</td>
</tr>
<tr>
<td>2.766</td>
<td>1.064e-2</td>
<td>8.37e-7</td>
</tr>
<tr>
<td>2.740</td>
<td>1.088e-2</td>
<td>8.59e-7</td>
</tr>
<tr>
<td>2.717</td>
<td>1.113e-2</td>
<td>8.78e-7</td>
</tr>
<tr>
<td>2.698</td>
<td>1.138e-2</td>
<td>8.94e-7</td>
</tr>
<tr>
<td>2.681</td>
<td>1.163e-2</td>
<td>9.13e-7</td>
</tr>
<tr>
<td>2.666</td>
<td>1.188e-2</td>
<td>9.34e-7</td>
</tr>
<tr>
<td>2.653</td>
<td>1.213e-2</td>
<td>9.56e-7</td>
</tr>
<tr>
<td>2.641</td>
<td>1.239e-2</td>
<td>9.77e-7</td>
</tr>
<tr>
<td>2.631</td>
<td>1.264e-2</td>
<td>9.99e-7</td>
</tr>
</tbody>
</table>

III-93
<table>
<thead>
<tr>
<th>Number</th>
<th>Value 1</th>
<th>Value 2</th>
<th>Value 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.621</td>
<td>1.290e-2</td>
<td>1.02e-6</td>
<td></td>
</tr>
<tr>
<td>20.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.029</td>
<td>1.054e-2</td>
<td>8.0e-7</td>
<td></td>
</tr>
<tr>
<td>3.001</td>
<td>1.063e-2</td>
<td>8.09e-7</td>
<td></td>
</tr>
<tr>
<td>2.946</td>
<td>1.080e-2</td>
<td>8.27e-7</td>
<td></td>
</tr>
<tr>
<td>2.904</td>
<td>1.098e-2</td>
<td>8.45e-7</td>
<td></td>
</tr>
<tr>
<td>2.865</td>
<td>1.117e-2</td>
<td>8.64e-7</td>
<td></td>
</tr>
<tr>
<td>2.833</td>
<td>1.137e-2</td>
<td>8.82e-7</td>
<td></td>
</tr>
<tr>
<td>2.805</td>
<td>1.158e-2</td>
<td>9.01e-7</td>
<td></td>
</tr>
<tr>
<td>2.781</td>
<td>1.178e-2</td>
<td>9.19e-7</td>
<td></td>
</tr>
<tr>
<td>2.760</td>
<td>1.199e-2</td>
<td>9.38e-7</td>
<td></td>
</tr>
<tr>
<td>2.741</td>
<td>1.220e-2</td>
<td>9.56e-7</td>
<td></td>
</tr>
<tr>
<td>2.725</td>
<td>1.241e-2</td>
<td>9.75e-7</td>
<td></td>
</tr>
<tr>
<td>2.710</td>
<td>1.262e-2</td>
<td>9.92e-7</td>
<td></td>
</tr>
<tr>
<td>2.697</td>
<td>1.284e-2</td>
<td>1.01e-6</td>
<td></td>
</tr>
<tr>
<td>2.684</td>
<td>1.305e-2</td>
<td>1.02e-6</td>
<td></td>
</tr>
<tr>
<td>30.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.261</td>
<td>1.173e-2</td>
<td>8.7e-7</td>
<td></td>
</tr>
<tr>
<td>3.228</td>
<td>1.178e-2</td>
<td>8.8e-7</td>
<td></td>
</tr>
<tr>
<td>3.171</td>
<td>1.189e-2</td>
<td>8.95e-7</td>
<td></td>
</tr>
<tr>
<td>3.121</td>
<td>1.201e-2</td>
<td>9.06e-7</td>
<td></td>
</tr>
<tr>
<td>3.072</td>
<td>1.212e-2</td>
<td>9.16e-7</td>
<td></td>
</tr>
<tr>
<td>3.032</td>
<td>1.225e-2</td>
<td>9.29e-7</td>
<td></td>
</tr>
<tr>
<td>2.998</td>
<td>1.239e-2</td>
<td>9.43e-7</td>
<td></td>
</tr>
<tr>
<td>2.964</td>
<td>1.253e-2</td>
<td>9.57e-7</td>
<td></td>
</tr>
<tr>
<td>2.937</td>
<td>1.267e-2</td>
<td>9.70e-7</td>
<td></td>
</tr>
<tr>
<td>2.912</td>
<td>1.281e-2</td>
<td>9.84e-7</td>
<td></td>
</tr>
<tr>
<td>2.887</td>
<td>1.295e-2</td>
<td>9.98e-7</td>
<td></td>
</tr>
<tr>
<td>2.868</td>
<td>1.310e-2</td>
<td>1.01e-6</td>
<td></td>
</tr>
<tr>
<td>2.849</td>
<td>1.324e-2</td>
<td>1.02e-6</td>
<td></td>
</tr>
<tr>
<td>2.83</td>
<td>1.339e-2</td>
<td>1.02e-6</td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.28</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.36</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.44</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.76</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.84</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.92</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distance from Spray Tube Inlet (in)</td>
<td>Pressure (PSIA)</td>
<td>Flow Rate Through Spray Tube (lbm/sec)</td>
<td>Injection Flow Rate (lbm/sec)</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------</td>
<td>--------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>8</td>
<td>20</td>
<td>.001901</td>
<td>.5</td>
</tr>
<tr>
<td>4</td>
<td>.002224</td>
<td>.002534</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>.002821</td>
<td>.003079</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>.003629</td>
<td>.004122</td>
<td>3</td>
</tr>
<tr>
<td>30</td>
<td>.004647</td>
<td></td>
<td>3.5</td>
</tr>
</tbody>
</table>

3.5.2 Integrated Zero-g TVS Model
| TIME (SEC)          | ULLAGE PRESSURE (PSIA) | ULLAGE TEMPERATURE (R) | ULLAGE MASS (LBM) | BULK LIQUID PRESSURE (PSIA) | BULK LIQUID TEMPERATURE (R) | BULK LIQUID MASS (LBM) | WALL TEMPERATURE (ULLAGE SIDE) (R) | WALL LIQUID MASS (LBM) | OVERBOARD VENT FLOW RATE (LBM/SEC) | ULLAGE SPRAY FLOW RATE (LBM/SEC) | LIQUID DROPLET BOILING RATE (LBM/SEC) | WALL LIQUID BOILING RATE (LBM/SEC) |
**INTERFACIAL MASS-TRANSFER RATE (LBM/SEC)**
**PUMP FLOW RATE (LBM/SEC)**
**PUMP SPEED (RPM)**
**PUMP PRESSURE RISE (PSI)**
**ZERO-g TVS TRANSIENT PERFORMANCE**
(0g, 10% FULL, 0.25 BTU/HR-FT², NO He)

<table>
<thead>
<tr>
<th>Pump Performance Data</th>
<th>Saturated Hydrogen Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3 0.500 0.0 3134.0 5.0 0.01</td>
<td></td>
</tr>
<tr>
<td>27. 1.950 -129.360 1.669 4.74429 .0061184 .04800 1.496e-5</td>
<td></td>
</tr>
<tr>
<td>29. 3.288 -125.846 1.786 4.68219 .0063005 .05004 1.096e-5</td>
<td></td>
</tr>
<tr>
<td>30. 4.170 -123.995 1.849 4.65003 .01077 .0404077 .00739 .0570e-5</td>
<td></td>
</tr>
<tr>
<td>31. 5.217 -122.077 1.915 4.61705 .0073258 .05402 1.172e-5</td>
<td></td>
</tr>
</tbody>
</table>

III-97
<table>
<thead>
<tr>
<th></th>
<th>6.446</th>
<th>-120.090</th>
<th>1.289</th>
<th>1.985</th>
<th>4.58321</th>
<th>0.0076421</th>
<th>0.05489</th>
<th>1.112e-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>74.584</td>
<td>1.520</td>
<td>2.695</td>
<td>.04008</td>
<td>.0362124</td>
<td>.00856</td>
<td>.066e-5</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>7.877</td>
<td>-118.029</td>
<td>1.310</td>
<td>2.051</td>
<td>4.54844</td>
<td>0.0079185</td>
<td>0.05555</td>
<td>1.056e-5</td>
</tr>
<tr>
<td>76.261</td>
<td>1.527</td>
<td>2.734</td>
<td>.04790</td>
<td>.0359198</td>
<td>.00880</td>
<td>.068e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>9.527</td>
<td>-115.893</td>
<td>1.329</td>
<td>2.125</td>
<td>4.51266</td>
<td>0.0082741</td>
<td>0.05601</td>
<td>1.006e-5</td>
</tr>
<tr>
<td>77.848</td>
<td>1.535</td>
<td>2.778</td>
<td>.05675</td>
<td>.0357318</td>
<td>.00904</td>
<td>.070e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>11.416</td>
<td>-113.678</td>
<td>1.348</td>
<td>2.195</td>
<td>4.47582</td>
<td>0.0085840</td>
<td>0.05631</td>
<td>0.959e-5</td>
</tr>
<tr>
<td>79.339</td>
<td>1.543</td>
<td>2.826</td>
<td>.06671</td>
<td>.0356477</td>
<td>.00929</td>
<td>.072e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>13.561</td>
<td>-111.380</td>
<td>1.365</td>
<td>2.270</td>
<td>4.43782</td>
<td>0.0089505</td>
<td>0.05690</td>
<td>0.917e-5</td>
</tr>
<tr>
<td>80.729</td>
<td>1.551</td>
<td>2.879</td>
<td>.07787</td>
<td>.0356683</td>
<td>.00962</td>
<td>.075e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>15.984</td>
<td>-108.997</td>
<td>1.380</td>
<td>2.360</td>
<td>4.39858</td>
<td>0.0094522</td>
<td>0.05748</td>
<td>0.877e-5</td>
</tr>
<tr>
<td>82.105</td>
<td>1.559</td>
<td>2.935</td>
<td>.09011</td>
<td>.0357690</td>
<td>.00998</td>
<td>.077e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>18.694</td>
<td>-106.524</td>
<td>1.395</td>
<td>2.443</td>
<td>4.35801</td>
<td>0.0099007</td>
<td>0.05793</td>
<td>0.840e-5</td>
</tr>
<tr>
<td>83.256</td>
<td>1.567</td>
<td>2.998</td>
<td>.10395</td>
<td>.0360806</td>
<td>.01036</td>
<td>.079e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>21.723</td>
<td>-103.956</td>
<td>1.408</td>
<td>2.532</td>
<td>4.31600</td>
<td>0.0103939</td>
<td>0.05824</td>
<td>0.805e-5</td>
</tr>
<tr>
<td>84.290</td>
<td>1.576</td>
<td>3.068</td>
<td>.11930</td>
<td>.0363705</td>
<td>.01076</td>
<td>.082e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>25.089</td>
<td>-101.289</td>
<td>1.420</td>
<td>2.637</td>
<td>4.27243</td>
<td>0.0110382</td>
<td>0.05843</td>
<td>0.773e-5</td>
</tr>
<tr>
<td>85.199</td>
<td>1.584</td>
<td>3.146</td>
<td>.13629</td>
<td>.0368634</td>
<td>.01117</td>
<td>.084e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>28.813</td>
<td>-98.517</td>
<td>1.431</td>
<td>2.743</td>
<td>4.22718</td>
<td>0.0117045</td>
<td>0.05851</td>
<td>0.742e-5</td>
</tr>
<tr>
<td>85.976</td>
<td>1.592</td>
<td>3.233</td>
<td>.15504</td>
<td>.0374994</td>
<td>.01159</td>
<td>.087e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>32.915</td>
<td>-95.893</td>
<td>1.441</td>
<td>2.848</td>
<td>4.18010</td>
<td>0.0123677</td>
<td>0.05848</td>
<td>0.713e-5</td>
</tr>
<tr>
<td>86.614</td>
<td>1.601</td>
<td>3.331</td>
<td>.17568</td>
<td>.0382948</td>
<td>.01204</td>
<td>.089e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>37.415</td>
<td>-92.637</td>
<td>1.451</td>
<td>2.969</td>
<td>4.13102</td>
<td>0.0131689</td>
<td>0.05835</td>
<td>0.685e-5</td>
</tr>
<tr>
<td>87.103</td>
<td>1.610</td>
<td>3.441</td>
<td>.19839</td>
<td>.0392710</td>
<td>.01251</td>
<td>.092e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>42.334</td>
<td>-89.513</td>
<td>1.459</td>
<td>3.097</td>
<td>4.07975</td>
<td>0.0140457</td>
<td>0.05811</td>
<td>0.659e-5</td>
</tr>
<tr>
<td>87.431</td>
<td>1.619</td>
<td>3.566</td>
<td>.22335</td>
<td>.0404560</td>
<td>.01301</td>
<td>.095e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>47.693</td>
<td>-86.261</td>
<td>1.467</td>
<td>3.242</td>
<td>4.02608</td>
<td>0.0150803</td>
<td>0.05778</td>
<td>0.634e-5</td>
</tr>
<tr>
<td>87.584</td>
<td>1.629</td>
<td>3.709</td>
<td>.25078</td>
<td>.0418864</td>
<td>.01354</td>
<td>.097e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>51.838</td>
<td>-74.939</td>
<td>1.507</td>
<td>4.307</td>
<td>3.71079</td>
<td>0.0235331</td>
<td>0.05459</td>
<td>0.519e-5</td>
</tr>
<tr>
<td>85.043</td>
<td>1.683</td>
<td>4.199</td>
<td>.43621</td>
<td>.0551860</td>
<td>.01694</td>
<td>.112e-5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>130.299</td>
<td>-42.122</td>
<td>1.564</td>
<td>7.528</td>
<td>3.25720</td>
<td>0.0531428</td>
<td>0.04960</td>
<td>0.410e-5</td>
</tr>
<tr>
<td>73.413</td>
<td>1.831</td>
<td>9.187</td>
<td>.76511</td>
<td>.1048799</td>
<td>.02422</td>
<td>.135e-5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8.11 h cv cp rho beta k mu T Superheated hydrogen properties
0. .01 .02 .03 .04 .05 .06 .07 .08 .09 .1 .600.
1. 25.000
60.765 | 1.483 | 2.511 | .00763 | .00721 | .050e-5 | 25.000
75.138 | 1.481 | 2.492 | .00617 | .00804 | .063e-5 | 30.750
89.438 | 1.480 | 2.483 | .00518 | .00915 | .074e-5 | 36.500
103.700 | 1.480 | 2.478 | .00447 | .01055 | .086e-5 | 42.250
117.938 | 1.480 | 2.474 | .00393 | .01202 | .097e-5 | 48.000
132.160 | 1.480 | 2.473 | .00351 | .01347 | .108e-5 | 53.750
146.372 | 1.480 | 2.471 | .00317 | .01477 | .117e-5 | 59.500
160.577 | 1.480 | 2.470 | .00297 | .01607 | .128e-5 | 65.250
174.782 | 1.481 | 2.470 | .00265 | .01737 | .137e-5 | 71.000
187.139 | 1.484 | 2.472 | .00248 | .01850 | .146e-5 | 76.750

III-98
<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>203.223</td>
<td>1.489</td>
<td>2.477</td>
<td>0.00228</td>
<td>0.01996</td>
<td>0.156e-5</td>
<td>82.500</td>
</tr>
<tr>
<td>10.0</td>
<td>30.806</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>78.249</td>
<td>1.537</td>
<td>2.790</td>
<td>0.05926</td>
<td>0.00704</td>
<td>0.071e-5</td>
<td>34.263</td>
</tr>
<tr>
<td>93.566</td>
<td>1.502</td>
<td>2.649</td>
<td>0.04952</td>
<td>0.01030</td>
<td>0.082e-5</td>
<td>39.920</td>
</tr>
<tr>
<td>108.372</td>
<td>1.493</td>
<td>2.591</td>
<td>0.04274</td>
<td>0.01168</td>
<td>0.093e-5</td>
<td>45.578</td>
</tr>
<tr>
<td>122.930</td>
<td>1.489</td>
<td>2.558</td>
<td>0.03766</td>
<td>0.01306</td>
<td>0.103e-5</td>
<td>51.235</td>
</tr>
<tr>
<td>137.339</td>
<td>1.487</td>
<td>2.537</td>
<td>0.03370</td>
<td>0.01439</td>
<td>0.114e-5</td>
<td>56.892</td>
</tr>
<tr>
<td>151.651</td>
<td>1.486</td>
<td>2.523</td>
<td>0.03051</td>
<td>0.01564</td>
<td>0.124e-5</td>
<td>62.550</td>
</tr>
<tr>
<td>165.892</td>
<td>1.485</td>
<td>2.513</td>
<td>0.02788</td>
<td>0.01692</td>
<td>0.133e-5</td>
<td>68.207</td>
</tr>
<tr>
<td>180.088</td>
<td>1.486</td>
<td>2.506</td>
<td>0.02570</td>
<td>0.01819</td>
<td>0.142e-5</td>
<td>73.865</td>
</tr>
<tr>
<td>194.258</td>
<td>1.489</td>
<td>2.503</td>
<td>0.02382</td>
<td>0.01944</td>
<td>0.151e-5</td>
<td>79.522</td>
</tr>
<tr>
<td>208.423</td>
<td>1.493</td>
<td>2.504</td>
<td>0.02220</td>
<td>0.02071</td>
<td>0.160e-5</td>
<td>85.179</td>
</tr>
<tr>
<td>222.605</td>
<td>1.502</td>
<td>2.509</td>
<td>0.02080</td>
<td>0.02198</td>
<td>0.169e-5</td>
<td>90.837</td>
</tr>
<tr>
<td>15.0</td>
<td>36.608</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>81.613</td>
<td>1.556</td>
<td>2.912</td>
<td>0.08508</td>
<td>0.00984</td>
<td>0.076e-5</td>
<td>36.608</td>
</tr>
<tr>
<td>97.298</td>
<td>1.511</td>
<td>2.722</td>
<td>0.07143</td>
<td>0.01107</td>
<td>0.087e-5</td>
<td>42.242</td>
</tr>
<tr>
<td>112.381</td>
<td>1.499</td>
<td>2.640</td>
<td>0.06178</td>
<td>0.01239</td>
<td>0.098e-5</td>
<td>47.876</td>
</tr>
<tr>
<td>127.120</td>
<td>1.493</td>
<td>2.595</td>
<td>0.05459</td>
<td>0.01374</td>
<td>0.108e-5</td>
<td>53.510</td>
</tr>
<tr>
<td>141.657</td>
<td>1.490</td>
<td>2.566</td>
<td>0.04898</td>
<td>0.01500</td>
<td>0.118e-5</td>
<td>59.144</td>
</tr>
<tr>
<td>156.056</td>
<td>1.489</td>
<td>2.546</td>
<td>0.04445</td>
<td>0.01625</td>
<td>0.128e-5</td>
<td>64.778</td>
</tr>
<tr>
<td>170.360</td>
<td>1.488</td>
<td>2.532</td>
<td>0.04072</td>
<td>0.01751</td>
<td>0.137e-5</td>
<td>70.412</td>
</tr>
<tr>
<td>184.599</td>
<td>1.489</td>
<td>2.524</td>
<td>0.03759</td>
<td>0.01876</td>
<td>0.147e-5</td>
<td>76.045</td>
</tr>
<tr>
<td>198.803</td>
<td>1.492</td>
<td>2.519</td>
<td>0.03491</td>
<td>0.02001</td>
<td>0.155e-5</td>
<td>81.679</td>
</tr>
<tr>
<td>212.996</td>
<td>1.498</td>
<td>2.519</td>
<td>0.03260</td>
<td>0.02126</td>
<td>0.164e-5</td>
<td>87.313</td>
</tr>
<tr>
<td>227.201</td>
<td>1.507</td>
<td>2.524</td>
<td>0.03057</td>
<td>0.02252</td>
<td>0.172e-5</td>
<td>92.947</td>
</tr>
<tr>
<td>20.0</td>
<td>38.444</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>83.731</td>
<td>1.571</td>
<td>3.029</td>
<td>0.11058</td>
<td>0.01054</td>
<td>0.080e-5</td>
<td>38.444</td>
</tr>
<tr>
<td>99.896</td>
<td>1.519</td>
<td>2.790</td>
<td>0.09269</td>
<td>0.01169</td>
<td>0.091e-5</td>
<td>44.060</td>
</tr>
<tr>
<td>115.240</td>
<td>1.504</td>
<td>2.689</td>
<td>0.08029</td>
<td>0.01297</td>
<td>0.102e-5</td>
<td>49.675</td>
</tr>
<tr>
<td>130.163</td>
<td>1.498</td>
<td>2.631</td>
<td>0.07106</td>
<td>0.01427</td>
<td>0.112e-5</td>
<td>55.291</td>
</tr>
<tr>
<td>144.826</td>
<td>1.494</td>
<td>2.594</td>
<td>0.06384</td>
<td>0.01550</td>
<td>0.121e-5</td>
<td>60.906</td>
</tr>
<tr>
<td>159.318</td>
<td>1.491</td>
<td>2.568</td>
<td>0.05801</td>
<td>0.01674</td>
<td>0.131e-5</td>
<td>66.522</td>
</tr>
<tr>
<td>173.683</td>
<td>1.490</td>
<td>2.552</td>
<td>0.05327</td>
<td>0.01799</td>
<td>0.140e-5</td>
<td>72.137</td>
</tr>
<tr>
<td>187.979</td>
<td>1.492</td>
<td>2.541</td>
<td>0.04921</td>
<td>0.01923</td>
<td>0.149e-5</td>
<td>77.753</td>
</tr>
<tr>
<td>202.224</td>
<td>1.495</td>
<td>2.534</td>
<td>0.04574</td>
<td>0.02047</td>
<td>0.158e-5</td>
<td>83.368</td>
</tr>
<tr>
<td>216.452</td>
<td>1.502</td>
<td>2.533</td>
<td>0.04273</td>
<td>0.02171</td>
<td>0.167e-5</td>
<td>88.984</td>
</tr>
<tr>
<td>230.687</td>
<td>1.512</td>
<td>2.538</td>
<td>0.04010</td>
<td>0.02296</td>
<td>0.175e-5</td>
<td>94.600</td>
</tr>
<tr>
<td>25.0</td>
<td>39.975</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>85.177</td>
<td>1.584</td>
<td>3.144</td>
<td>0.13584</td>
<td>0.01116</td>
<td>0.084e-5</td>
<td>39.975</td>
</tr>
<tr>
<td>101.806</td>
<td>1.526</td>
<td>2.857</td>
<td>0.11368</td>
<td>0.01224</td>
<td>0.094e-5</td>
<td>45.575</td>
</tr>
<tr>
<td>117.424</td>
<td>1.509</td>
<td>2.734</td>
<td>0.09846</td>
<td>0.01348</td>
<td>0.105e-5</td>
<td>51.176</td>
</tr>
<tr>
<td>132.526</td>
<td>1.501</td>
<td>2.665</td>
<td>0.08719</td>
<td>0.01473</td>
<td>0.115e-5</td>
<td>56.776</td>
</tr>
<tr>
<td>147.319</td>
<td>1.497</td>
<td>2.620</td>
<td>0.07839</td>
<td>0.01594</td>
<td>0.125e-5</td>
<td>62.376</td>
</tr>
<tr>
<td>161.903</td>
<td>1.493</td>
<td>2.590</td>
<td>0.07131</td>
<td>0.01717</td>
<td>0.134e-5</td>
<td>67.976</td>
</tr>
<tr>
<td>176.339</td>
<td>1.493</td>
<td>2.570</td>
<td>0.06554</td>
<td>0.01841</td>
<td>0.143e-5</td>
<td>73.577</td>
</tr>
</tbody>
</table>

III-99
<table>
<thead>
<tr>
<th>Value</th>
<th>1.494</th>
<th>2.556</th>
<th>.06058</th>
<th>.01964</th>
<th>.152e-5</th>
<th>79.177</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>1.498</td>
<td>2.546</td>
<td>.05634</td>
<td>.02087</td>
<td>.161e-5</td>
<td>84.177</td>
</tr>
<tr>
<td>Value</td>
<td>1.505</td>
<td>2.546</td>
<td>.02569</td>
<td>.02210</td>
<td>.169e-5</td>
<td>90.377</td>
</tr>
<tr>
<td>Value</td>
<td>1.517</td>
<td>2.550</td>
<td>.04952</td>
<td>.02335</td>
<td>.177e-5</td>
<td>95.978</td>
</tr>
<tr>
<td>30.</td>
<td>41.299</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>1.595</td>
<td>3.261</td>
<td>.16101</td>
<td>.01173</td>
<td>.087e-5</td>
<td>41.299</td>
</tr>
<tr>
<td>Value</td>
<td>1.533</td>
<td>2.924</td>
<td>.13441</td>
<td>.01274</td>
<td>.098e-5</td>
<td>46.886</td>
</tr>
<tr>
<td>Value</td>
<td>1.514</td>
<td>2.779</td>
<td>.11635</td>
<td>.01393</td>
<td>.108e-5</td>
<td>52.473</td>
</tr>
<tr>
<td>Value</td>
<td>1.505</td>
<td>2.697</td>
<td>.10303</td>
<td>.01523</td>
<td>.117e-5</td>
<td>58.060</td>
</tr>
<tr>
<td>Value</td>
<td>1.499</td>
<td>2.646</td>
<td>.09273</td>
<td>.01634</td>
<td>.127e-5</td>
<td>63.647</td>
</tr>
<tr>
<td>Value</td>
<td>1.495</td>
<td>2.610</td>
<td>.08442</td>
<td>.01878</td>
<td>.146e-5</td>
<td>69.234</td>
</tr>
<tr>
<td>Value</td>
<td>1.505</td>
<td>2.587</td>
<td>.07756</td>
<td>.02210</td>
<td>.161e-5</td>
<td>74.821</td>
</tr>
<tr>
<td>Value</td>
<td>1.496</td>
<td>2.571</td>
<td>.07178</td>
<td>.02210</td>
<td>.169e-5</td>
<td>79.177</td>
</tr>
<tr>
<td>Value</td>
<td>1.500</td>
<td>2.562</td>
<td>.06685</td>
<td>.02122</td>
<td>.163e-5</td>
<td>84.177</td>
</tr>
<tr>
<td>Value</td>
<td>1.509</td>
<td>2.559</td>
<td>.06257</td>
<td>.02245</td>
<td>.171e-5</td>
<td>90.377</td>
</tr>
<tr>
<td>Value</td>
<td>1.521</td>
<td>2.563</td>
<td>.05881</td>
<td>.02369</td>
<td>.179e-5</td>
<td>95.978</td>
</tr>
<tr>
<td>35.</td>
<td>42.475</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>1.605</td>
<td>3.381</td>
<td>.18169</td>
<td>.01226</td>
<td>.090e-5</td>
<td>42.475</td>
</tr>
<tr>
<td>Value</td>
<td>1.539</td>
<td>2.987</td>
<td>.15482</td>
<td>.01319</td>
<td>.100e-5</td>
<td>48.050</td>
</tr>
<tr>
<td>Value</td>
<td>1.518</td>
<td>2.822</td>
<td>.13404</td>
<td>.01434</td>
<td>.110e-5</td>
<td>53.626</td>
</tr>
<tr>
<td>Value</td>
<td>1.508</td>
<td>2.730</td>
<td>.11876</td>
<td>.01551</td>
<td>.120e-5</td>
<td>59.201</td>
</tr>
<tr>
<td>Value</td>
<td>1.501</td>
<td>2.670</td>
<td>.10690</td>
<td>.01671</td>
<td>.129e-5</td>
<td>64.776</td>
</tr>
<tr>
<td>Value</td>
<td>1.498</td>
<td>2.630</td>
<td>.09737</td>
<td>.01791</td>
<td>.139e-5</td>
<td>70.351</td>
</tr>
<tr>
<td>Value</td>
<td>1.497</td>
<td>2.604</td>
<td>.08953</td>
<td>.01912</td>
<td>.147e-5</td>
<td>75.927</td>
</tr>
<tr>
<td>Value</td>
<td>1.499</td>
<td>2.586</td>
<td>.08291</td>
<td>.02033</td>
<td>.156e-5</td>
<td>81.502</td>
</tr>
<tr>
<td>Value</td>
<td>1.503</td>
<td>2.576</td>
<td>.07724</td>
<td>.02155</td>
<td>.165e-5</td>
<td>87.077</td>
</tr>
<tr>
<td>Value</td>
<td>1.512</td>
<td>2.572</td>
<td>.07231</td>
<td>.02277</td>
<td>.173e-5</td>
<td>92.652</td>
</tr>
<tr>
<td>Value</td>
<td>1.524</td>
<td>2.575</td>
<td>.06799</td>
<td>.02399</td>
<td>.181e-5</td>
<td>98.228</td>
</tr>
<tr>
<td>40.</td>
<td>43.536</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>1.615</td>
<td>3.506</td>
<td>.21148</td>
<td>.01278</td>
<td>.093e-5</td>
<td>43.536</td>
</tr>
<tr>
<td>Value</td>
<td>1.545</td>
<td>3.056</td>
<td>.17544</td>
<td>.01362</td>
<td>.103e-5</td>
<td>49.101</td>
</tr>
<tr>
<td>Value</td>
<td>1.523</td>
<td>2.866</td>
<td>.15163</td>
<td>.01472</td>
<td>.113e-5</td>
<td>54.665</td>
</tr>
<tr>
<td>Value</td>
<td>1.512</td>
<td>2.761</td>
<td>.13427</td>
<td>.01587</td>
<td>.122e-5</td>
<td>60.230</td>
</tr>
<tr>
<td>Value</td>
<td>1.503</td>
<td>2.694</td>
<td>.12088</td>
<td>.01705</td>
<td>.132e-5</td>
<td>65.795</td>
</tr>
<tr>
<td>Value</td>
<td>1.500</td>
<td>2.651</td>
<td>.11027</td>
<td>.01823</td>
<td>.140e-5</td>
<td>71.359</td>
</tr>
<tr>
<td>Value</td>
<td>1.499</td>
<td>2.621</td>
<td>.10139</td>
<td>.01943</td>
<td>.149e-5</td>
<td>76.924</td>
</tr>
<tr>
<td>Value</td>
<td>1.500</td>
<td>2.601</td>
<td>.09390</td>
<td>.02063</td>
<td>.158e-5</td>
<td>82.488</td>
</tr>
<tr>
<td>Value</td>
<td>1.506</td>
<td>2.589</td>
<td>.08750</td>
<td>.02185</td>
<td>.166e-5</td>
<td>88.053</td>
</tr>
<tr>
<td>Value</td>
<td>1.515</td>
<td>2.584</td>
<td>.08193</td>
<td>.02306</td>
<td>.175e-5</td>
<td>93.618</td>
</tr>
<tr>
<td>Value</td>
<td>1.528</td>
<td>2.586</td>
<td>.07706</td>
<td>.02428</td>
<td>.183e-5</td>
<td>99.182</td>
</tr>
</tbody>
</table>

III-100