1983-12

Marine steam condenser design optimization

Buckingham, Thomas M.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

MARINE STEAM CONDENSER
DESIGN OPTIMIZATION

by

Thomas M. Buckingham

December 1983

Thesis Advisor: R. H. Nunn

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Marine Steam Condenser Design Optimization	5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1983
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Thomas M. Buckingham	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943	12. REPORT DATE December 1983
	13. NUMBER OF PAGES 224
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Condenser Optimization
CONDIP
ORCON1
Condenser Computer Program

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A surface-condenser analysis code was coupled with a constrained function minimization code to produce an automated marine condenser design and optimization package. The program, CONDIP, was based on the principles developed in ORCON1, a sophisticated computer code produced by the Oak Ridge National Laboratory. CONMIN, the optimization program, was developed at the Ames Research Center.

CONDIP is an extremely versatile design tool, incorporating a detailed analysis of the complex steam-side thermodynamic processes occurring at each row in the condenser. The additional capability of tube enhancement is also included. However, in coupling CONDIP with CONMIN numerous problems had to be overcome in order to make CONDIP capable of completing an analysis even when thermodynamic conditions in the condenser became infeasible. This had to be accomplished while ensuring continuity in all constraint and objective function evaluations. A series of test cases were conducted to evaluate and compare the importance of various objective functions and design criteria.

Approved for public release; distribution unlimited.

Marine Steam Condenser Design Optimization

by

Thomas M. Buckingham
Lieutenant, United States Navy
B.A., College of the Holy Cross, 1977

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1983

ABSTRACT

A surface-condenser analysis code was coupled with a constrained function minimization code to produce an automated marine condenser design and optimization package. The program, CONDIP, was based on the principles developed in ORCON1, a sophisticated computer code produced by the Oak Ridge National Laboratory. CONMIN, the optimization program, was developed at the Ames Research Center.

CONDIP is an extremely versatile design tool, incorporating a detailed analysis of the complex steam-side thermodynamic processes occurring at each row in the condenser. The additional capability of tube enhancement is also included. However, in coupling CONDIP with CONMIN numerous problems had to be overcome in order to make CONDIP capable of completing an analysis even when thermodynamic conditions in the condenser became infeasible. This had to be accomplished while ensuring continuity in all constraint and objective function evaluations. A series of test cases were conducted to evaluate and compare the importance of various objective functions and design criteria.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to Professor Robert Nunn and Professor Gary Vanderplaats for their advice and guidance during the development of this project. Without their encouragement and valuable assistance this thesis could never have been completed.

Many thanks and appreciation are due my wife, Susan, for her understanding, encouragement and participation in this endeavor.

Finally, this thesis is dedicated to Louis Perra, a close friend, whose pride and faith in my goals and ability inspired me to do the best job possible.

I. INTRODUCTION

A. BACKGROUND

For many years the steam plant was unchallenged in its role as the primary type of marine propulsion systems. But recently gas turbines have become a desirable alternative despite the fact that they are less efficient than comparable steam plants. The primary advantage of gas turbines is their light weight and compact size. Thus, in order for the marine steam plant to survive, it is imperative that lighter, more compact and efficient steam plants be developed.

While there are numerous advanced concepts in all areas of steam propulsion which can be explored, one simple way to streamline the steam plant is the elimination of overdesign. Most overdesign is due to unnecessary safety factors used to offset lack of detailed knowledge about the thermal processes in the plant. Identification of the minimum safe design could significantly reduce unnecessary overdesign and result in the development of a smaller, more compact power plant.

B. METHODOLOGY

In the United States the most prevalent criterion for the design and specification of surface condensers is based on the "square-root of V" relationship, as developed by the Heat Exhange Institute (HEI) [Ref. 1]. The HEI method was adopted by the Department of the Navy Bureau of Ships (now Naval Sea System Command) for specification of U.S. Naval condensers.

The HEI method is very simple in its approach, calculating the overall heat transfer coefficient as a function of cooling water velocity through the tubes, inlet coolant temprature, tube wall thickness and material, and fouling. The limitations of this method are apparent. Designs based on HEI are insensitive to shell-side conditions. Saturation steam pressure, temperature and enthalpy are assumed to be constant as steam passes through the bundle, whereas in reality there is a continual pressure drop as steam flow passes over the rows of tubes, with a corresponding decrease in saturation temperature. There is also no provision for any effects of condensate film, external tube enhancement, etc. on the shell-side of the bundle. In addition, the HEI method does not account for the presence and effect of non-condensible gases that inevitably contaminate a condenser.

With the capabilities of high speed computers now available, more comprehensive methods have been developed to account for the deficiencies of the HEI method. In particular, a radial flow computer code was developed to calculate the local heat transfer and thermodynamic properties on a row by row basis. Known as ORCON1, this code was developed by Oak Ridge National Laboratory (ORNL) under contract to the Office of Saline Water during the period from 1968-1970 [Ref. 2]. The program was based , in part, on the work performed by Eissenberg [Ref. 3]. Eissenberg's experimental results led to correction factors on the basic Nusselt equation to account for condensate inundation effects on tubes within a condenser bundle. Basically, ORCON1 divides the condenser into sectors and performs a row by row analysis within each sector, determing local heat transfer coefficients, heat flux, steam characteristics, the effect of condensate inundation and numerous other parameters at each row. ORCON1 is also capable of incorporating the effects of both tube-side and steam-side enhancement factors. Since

ORCON1 represented a much more comprehensive and detailed analysis of the condenser then the less exact HEI method, its results could be expected to be more precise.

Some work has been done at the Naval Postgraduate School to improve the capabilities of ORCON1. In his developement of OPCODE2, Johnson [Ref. 5] added subroutines to ORCON1 which calculated tube-side pressure drops, corresponding pumping power and condenser volume. Nunn and Marto [Ref. 14] have incorporated the effects of vapor shear in an amended version of ORCON1 called MORCON. MORCON includes the correlations developed by Fujii [Ref. 4] to determine the effect of vapor velocity on the thermal resistance of the condensate film on the condenser tubes. In general, vapor shear effects tend to enhance the condenser heat transfer on the steam-side of the tube while condensate inundation tends to inhibit it.

The ability to represent numerically the actual thermodynamic processes occurring within the condenser has improved dramatically. However, the capability to couple these increasingly comprehensive and complex condenser design programs with an optimizing procedure has not made comparable progress. Optimization is a powerful tool which can help in reducing overdesign and achieving the goal of a safe compact condenser design.

There are currently numerous computer optimization programs available which can be coupled with general design programs of all types to numerically improve and ultimately determine the best design. The key is to properly write the design program so that it is compatible with the optimizer. Johnson [Ref. 5] developed a computer program called OPCODE1, based on the HEI method of condenser design, and was able to couple it with one such numerical optimizer. The results of OPCODE1 demonstrated how condenser designs can indeed be safely improved upon. It also revealed the

versatility of condenser design optimization as a powerful design tool. However, Johnson was unsuccessful in coupling OPCODE2 (his derivative of ORCON1) with an optimizer. This failure does not alter the fact that that in order to fully appreciate more sophisticated condenser design analyses, such as that used in ORCON1, it is imperative that computer programs be developed which will be compatible with current numerical optimizers.

C. OBJECTIVE

There were two primary objectives of this thesis. The first objective was to develop a computer code which incorporates the basic condenser analysis of ORCON1 and the subsequent improvements made in MORCON and OPCODE2, but which will be capable of being coupled with a numerical optimizer to yield a complete, detailed design package. This design package can then be used as a tool in obtaining a much more reasonable conceptual design and for use in comparison studies. It would provide the naval architect the ability to optimize weight, volume, cost or any other potential design objective of the marine plant.

The second objective was to make this design package capable of determining the single best design rather than simply an improvement over the initial design. The key was to construct the program in such a way so the optimizer does not stop at some relative optimum, but continues the analysis until no further improvement can be realized. It is most desirable to be able to reach this single true optimium design regardless of initial design variable values.

II. NUMERICAL OPTIMIZATION

A. BACKGROUND

Nearly all design problems require either the minimization or maximization of a parameter. This parameter will be called the problem's objective function or design objective [Ref. 6]. For a given design to be feasible or acceptable, it must satisfy a set of design constraints which are either maximum or minimum limiting values for a pre-determined set of parameters or functions of parameters. For example, in any condenser design the outer diameter of a condenser tube can never be less than zero and there is normally some practical upper limit which also cannot be exceeded. These limits are design constraints on the tube outer diameter. In the design problem there is also a set of design variables which are parameters whose values can be changed within specified limits in order to minimize or maximize the design objective. For example, in minimizing the condenser volume an engineer may want to vary tube inner diameter, tube wall thickness and tube length. These three parameters would thus be examples of typical design variables.

For such complex design problems as the treatment of the condenser design in ORCON1, it is necessary to choose an optimization scheme which can handle the problem and provide a rational, rapid approach to design automation and optimization. An optimization program based on direct methods for solution of constrained problems [Ref. 11] was chosen for this research work.

B. CONSTRAINED FUNCTION MINIMIZATION (CONMIN)

Vanderplaats [Ref. 7] developed an optimization program, CONMIN, capable of optimizing a very wide class of engineering problems. CONMIN is a fortran program, in subprogram form, that optimizes a multi-variable function subject to a set of inequality constraints.

It is practical at this point to introduce three basic definitions and their respective conditions [Ref. 8].

DESIGN VARIABLES: Those parameters which the optimzation program is permitted to change in order to improve the design. Design variables appear only on the right side of an equation, are continuous, and have continuous first derivatives.

DESIGN CONSTRAINTS: Any parameter which must not exceed specified bounds for the design to be acceptable. Design constraints may be linear or nonlinear, implicit or explicit, but they must be functions of the design variables. Design constraints appear only on the left side of equations.

OBJECTIVE FUNCTION: The parameter which is going to be minimized or maximized during the optimization process. The objective function may also be either linear or nonlinear, implicit or explicit, and must be a function of the design variables. The objective function usually appears on the left side of an equation. The only exception is if the objective function is also a design variable.

Assuming that the optimization process requires the minimization of a particular objective function the general optimzation problem can be stated as:

Find the vector of design variables, $\underline{X}$,
To minimize the objective function, $F(\underline{X})$,
Subject to the constraints:

$$G_j(\underline{X}) \leq 0.0 \qquad j = 1,NCON \qquad \text{(eqn 2.1)}$$

$$VLB_i \leq \underline{X} \leq VUB_i \qquad i = 1,NDV \qquad \text{(eqn 2.2)}$$

In the general problem, $G_j(\underline{X})$ are the constraint functions; there are NCON constraints and NDV design

variables; VLB_i and VUB_i are the lower bounds and upper bounds of the i-th design variable. If the equality condition is met, ($G_j(\underline{X})=0.$), the constraint is active. If the inequality is met, ($G_j(\underline{X})<0.$), the constraint is inactive. Finally, if the inequality of equation 2.1 is violated, ($G_j(\underline{X})>0.$), that constraint is said to be violated. Because of numerical inaccuracies representing exact zero on the computer, the equality condition is represented by a band around the value $G_j(\underline{X})=(0.\pm CT)$ where CT is the constraint thickness.

Any design which satisfies the inequalities of equations 2.1 and 2.2, thus having no violated constraints, is said to be feasible. If the design violates any of these constraints it is said to be an infeasible design. The design which best mimimizes the objective function while still remaining feasible is said to be optimal.

CONMIN requires an initial set of values for the design variables $\underline{X}$ to obtain an initial design which is either feasible or infeasible. If the initial design is feasible, CONMIN moves in a direction which will minimize the objective function. If the initial design is infeasible, CONMIN moves toward a feasible solution with minimal increase in the object function.

The optimization process proceeds in an iterative fashion. Johnson [Ref. 5] presents in greater detail the procedures utilized in CONMIN to search for the minimum objective value. In general, the methods used by CONMIN to determine search direction include the method of steepest descent, the method of conjugate direction, and the method of feasible directions. For further background concerning CONMIN and the numerical techniques utilized in optimization, consult Vanderplaats [Ref. 7], Fletcher and Reeves [Ref. 9], Zoutendijk [Ref. 10], and Vanderplaats and Moses [Ref. 12]. However, it is necessary to stress a few pertinent points which will aid in understanding how the program was developed in this thesis.

The optimization process begins by calculating the gradient of the objective function using a finite difference technique. A perturbation is applied to each of the design variables in a single forward step and the gradient vector is determined.

$$\Delta F(\underline{X}) = \begin{bmatrix} \frac{\partial F(\underline{X})}{\partial X} \\ \\ \frac{\partial F(\underline{X})}{\partial X} \end{bmatrix}$$

The search direction is then calculated and is a function of this gradient and any active or violated constraints resulting from the applied perturbation. Subsequent search directions are a function of previous search directions, as well as current gradient information and any appropriate constraint factors. Obviously, the size of the perturbation and the size of the bandwidth about an active constraint will have a great deal of effect on the search direction and ultimate optimization process. This detail will be recalled later-on during the code development.

There are some limitations to CONMIN. The number of design variables (NDV) directly affects the computational time required to reach an optimum. Since the calculation of the gradient information required for each design variable at the beginning of each design iteration is found by using a single forward finite difference step, requiring a complete pass through the analysis portion of the program, there is an increase in CPU time as NDV inceases. Also, as NDV increases, there is the corresponding rise in machine related numerical innacuracy. Vanderplaats [Ref. 6] recommends no more than twenty as a practical limit for the number of design variables.

It is quite possible that while design improvement may be obtained, the single best design optimum or true optimum

may not be reached. This is not an uncommon occurrence and there are several possible explanations. For example, the design problem may not be formulated properly or the analysis may be extremely complex and non-linear. However, a more common reason is that there are "relative optimums" between the initial design and the single true optimum. This concept of relative vs true optimum design can be better explained through an analogy. The search for the best optimum design can be likened to a blind man climbing to the top of a mountain. The blind man knows he is proceeding up the mountain by sensing the direction of ascent. However, the paths he takes may be limited by barriers or fences which will restrict the directions he can go. These fences represent constraints in the optimization problem. During the journey he may also encounter small crests and valleys. If the available paths lead the blind man up to one of these crests prior to reaching the mountain top, he will be confronted with a situation where he will sense no further rate of ascent and he will stop his journey. So although he has made progress from his initial starting point, the man did not achieve his ultimate goal of climbing the mountain.

During optimization, the search for a true optimum may proceed along a path on which the objective function assumes such relative optimum values. If the optimizer can not be made to "look beyond" these relative peaks, then the optimization will cease - at a design which may be an improvement over the initial one but short of the true optimum. This problem may be overcome by starting the design with several different initial design vectors, $\underline{X}$, until the same optimal design is repeated. Another alternative may be to increase the size of the finite difference so that the optimizer uses larger perturbations of $\underline{X}$ thus looking beyond any small increases in the objective function which could stand in the way of further design progress. This second

alternative will be specifically addressed during the discussion of the code development.

C. CONTROL PROGRAM FOR ENGINEERING SYNTHESIS (COPES)

The optimizer, CONMIN, was written in subroutine form. Vanderplaats [Ref. 13] has developed a main program to simplify the use of CONMIN and aid in the design optimization process.

The user must supply an analysis subroutine called ANALIZ, which consists of three segments: input, analysis and output. COPES acts as an interface between ANALIZ and the optimizer CONMIN. Based on a flag from COPES (ICALC=1,2,3) ANALIZ performs the proper function. Figure 2.1 offers a simplified illustration of the interrelationship between COPES, ANALIZ and CONMIN.

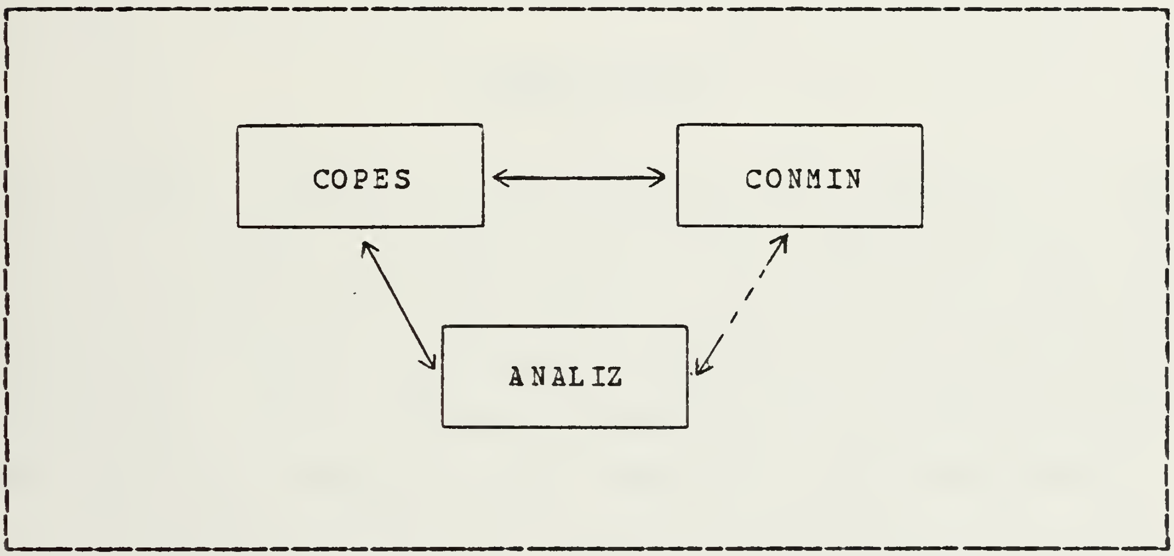


Figure 2.1 Flow Diagram For ANALIZ and COPES/CONMIN.

COPES currently provides four specific capabilities, two of which will be applied in this work:

1. Single analysis - just as if COPES/CONMIN were not used.
2. Optimization - minimization or maximization of a multi-variable function with corresponding constraints.

COPES requires certain initial data from the user in order to coordinate the optimization process. Initial values for the design variables as well optimizer control parameters are utilized by CONMIN to conduct its numerical analysis. There are a few optimizer parameters which are particularly important to the treatment of condenser designs. One is the finite difference step used in gradient calculations. Another is the normalization factor used in COPES evaluation of a constraint function. COPES utilizes the following expressions in determining constraint function violations:

$$\frac{BU - CFV}{SCAL1} \leq 0.$$

$$\frac{CFV - BL}{SCAL2} \leq 0.$$

where SCAL1 and SCAL2 are the normalization factors, BU and BL are the upper and lower limits of the constraint, and CFV is the constraint function value. It is intuitive that the normalization factor can play an important role in determining the size of the active region about a given design constraint. Both finite difference and constraint normalization will be recalled later during the code development.

The power of COPES is that it has simplified the procedures involved in using a sophisticated program such as CONMIN. The user is therefore freed from the unwanted role of systems analyst and can concentrate on the design analysis.

III. CONDENSER DESIGN IMPROVEMENT PROGRAM (CONDIP)

A. BACKGROUND

In the late 1960's, engineers at the Oak Ridge National Laboratory developed a sophisticated computer code under contract to the Office of Saline Water. This code, called ORCON1, [Ref. 2] was generated to aid in the analysis and parametric study of large, generally circular condensers. Much of ORCON1 was dependent on Eissenberg's research work [Ref. 3] on the effects of condensate rain on the shell-side convective heat transfer coefficient. Johnson [Ref. 5] took ORCON1 and made a few minor modifications to determine tube side pressure losses and volumetric calculations. Nunn and Marto [Ref. 14] further incorporated the correlations proposed by Fujii [Ref. 4] to determine the effects of shearing forces exerted by high vapor velocities on the condensate film and resulting shell-side heat transfer coefficient.

It was at this point that the development of CONDIP was begun. CONDIP was dependent primarily on the principles detailed in ORCON1 but also incorporated subsequent developments to the basic program. CONDIP was written , however, in such a way as to be compatible with the optimizer, CONMIN.

CONDIP analyzes a single or double pass, circular or semicircular condenser, with steam flowing radially inward on the shell-side of the tubes and variable salinity water flowing on the tube-side. An optional, rectangular air cooler bundle is provided-for as well as shell-side baffles. The circular bundle is normally divided into 30-degree sectors with symmetry about the central axis to reduce computational effort. Unless otherwise specified, tubes are

placed on a 60-degree equilateral, triangular pattern of concentric rows with the rows added from an inner void out to the outermost row. The void serves as a collection header for non-condensible gases prior to passage through an air cooler, if specified. As in ORCON1, CONDIP proceeds sector by sector, row by row through the condenser utilizing an average tube to represent the row segment, and calculates the following quantities in each sector:

a) Steam pressure losses at the entrance of a sector.
b) Total pressure of the steam/non-condensible gas mixture entering a row segment.
c) Saturation pressure of steam entering a row segment.
d) Saturation temperature of the steam entering a row segment.
e) Steam flow entering the row segment.
f) Velocity of the steam/non-condensible gas mixture at the minimum cross-section in the row segment.
g) The fraction of non-condensible gas in the mixture by weight.
h) The overall heat transfer coefficient for the average tube in the row segment.
i) The steam-side condensing coefficient.
j) The tube-side heat transfer coefficient.
k) The shell-side film heat transfer coefficient composed of the non-condenible gas film and the condensate film.
l) The shell-side friction factor.
m) The shell-side pressure loss as steam passes over the row segment.
n) The shell-side Reynolds number based on the mass flow at the minimum cross-sectional area in the row segment.
o) The heat transfer rate per square foot of condenser tube.
p) The mass flow rate of steam/non-condensable gas mixture at the minmum cross-section in the row segment.

q) The mass flow of condensate produced as steam passes over the row segment.

r) The cooling water temperature at the outlet end of the condenser.

s) The coolant pressure loss on the tube-side.

t) The average Reynolds number of the coolant through the tube.

u) The heat transfer coefficient for the non-condensible gas film.

v) The internal heat film heat transfer coefficient.

w) The number of tubes per row segment.

x) The cross-section area available for steam flow per row segment.

y) The cumulative shell-side pressure drop.

z) The LMTD based on inlet and outlet coolant temperatures and saturation temperature at each row segment.

In addition to the above parameters, the area-weighted overall heat transfer coefficients for the condenser, cooler and combined condenser are used to calculate the "back calculated" log mean temperature difference (LMTD). Steam exit-fraction, condenser volume, coolant pumping power and numerous other factors are calculated from the cumulative results of the row and sector analysis.

There are two significant contributions to the external film heat transfer coefficient which have a profound impact on the overall analysis. As mentioned earlier, Eissenberg [Ref. 3] corrected for condensate inundation effects on the external heat transfer coefficient with a series of emperical relations. He created a flooding factor F using the following relation:

$$F_n = .6F_d + (1 - .5647\ F_d)\,n^s \qquad \text{(eqn 3.1)}$$

where F_d is a constant indicating the effect of tube spacing and orientation on condensate side drainage. With closely packed tubes, significant side drainage can occur in low velocity steam flow. Condensate generated on tubes above may, due to surface tension effects, proceed laterally to adjacent tubes rather than down. Thus F_d tends to approach 1.0 for closely packed, staggered tube bundles and zero for disperse bundle layouts. S is a constant ranging in value from:

$$(.07 < S < .25)$$

If the condensate rain is acting under the influence of gravity alone S approaches 0.25. But the influence of any steam velocity present begins to alter the rate and direction of condensate flow and correspondingly decrease S. Thus S is a function of vapor velocity and directon, as well as bundle geometry.

The condensate film coefficient for the average tube in the n-th vertical row is then calculated from the uncorrected heat transfer coefficient, h_o, as follows:

$$h_n = [\, nF_n - (n-1)F_{n-1}]\, h_o \qquad \text{(eqn 3.2)}$$

It is obvious that the determination of a corrected heat transfer coefficient is highly dependent on the choice of S an F_d in equation 3.1. S and F_d are extremely subjective constants and there does not exist a current analytical expression to determine them. Yet the choice of these constants can have profound impact on the condenser design. In CONDIP, as in ORCON1 [Ref. 2], the following, relatively conservative values for S and F_d were used:

$$S = 0.2$$

$$F_d = 0.5$$

An additional important contribution to the external film coefficient is the effect of velocity shear forces on the condensate film. Fujii [Ref. 4] developed the following experimental correlations to correct the Nusselt number for the effect of velocity shear:

$$[Nu_m/Nu_o] = c_1 Nu_o^{(4a-1)} Re_L^{(.5-2a)} \qquad \text{(eqn 3.3)}$$

where Nu_m is the mean Nusselt number, Re_L is the two-phase Reynolds number (based on vapor velocity, tube outside diameter and kinematic viscosity of the condensate) and Nu_o is the standard Nusselt number for the zero shear case. The empirical constants c_1 and a lie within the following ranges:

$$1.13 < c_1 < 1.24$$
$$0.196 < a < 0.2$$

depending on how tube thermal conditions are described. In CONDIP the values for c_1 and a were:

$$c_1 = 1.24$$
$$a = 0.2$$

It should be noted that equation 3.3 is only valid in the range:

$$3.3 < (Re_L/Nu_o) < .28.$$

For smaller values of this parameter Fujii recommends the use of a slightly reduced value of the standard Nusselt number:

$$Nu_m = 0.96\ Nu_o \qquad \text{(eqn 3.4)}$$

It is apparent that the vapor velocities commonly encountered in naval condensers can have an impact on the heat transfer coeffcient.

B. CODE DEVELOPMENT

Johnson [Ref. 5] attempted to couple OPCODE2 (his version of ORCON1) with the optimizer CONMIN, but with little success. There were several reasons for this.

ORCON1 uses iterative techniques to solve for such quantities as condensate rate, steam mass flow rate and steam pressure through the sector. If unrealistic values are encountered, such as negative pressures or steam flow, or if the final steam exit-fraction exceeds a predetermined value, ORCON1 stops the analysis, returns to the beginning of the program and changes certain initial input parameters. The analysis begins again and the process is repeated until a satisfactory design is achieved. Thus ORCON1 has a limited capability to make design decisions to obtain a feasible design.

CONMIN, as do most optimizers, requires complete control in determining all iterative design variable values. As explained earlier, it uses perturbation techniques to calculate gradient information for each design variable and active design constraint, which it then uses to determine search directions. A perturbation of the design variable by CONMIN requires a complete, once-through analysis. If ORCON1 is coupled with CONMIN then any adjustment by ORCON1 will yield false gradient information to the optimizer and hinder, if not completely prevent, CONMIN from arriving at the optimum design. During program development it became apparent that the two programs were working independently against each other and that in its present state ORCON1 was incompatible with CONMIN.

In the formulation of CONDIP, it was necessary to locate and neutralize all the places where such design decisions are made. By removing the ability for CONDIP to make any design decisions it became totally passive and dependent on CONMIN for design variable changes.

However, once this was accomplished, another problem area was discovered. In the ORCON1 code and subsequently in CONDIP there are numerous thermal process and properties calculations that use logarithmic functions and other mathematical relationships which could produce singularities if the variables in the arguments approach zero or are negative. For example, saturation steam temperature is calculated from steam pressure using a logarithmic relationship. If, during a design analysis, saturation pressure approaches a negative value, this represents a clear violation of physical realities and of the limits of that property. Yet a computer cannot make that distinction so it tries to calculate the corresponding saturation temperature which, because of the logarithmic relationship, would be undefined. As just explained, ORCON1 with its built-in decision capability simply starts over when this situation is encountered. But CONDIP, being completely dependent on CONMIN for design decisions, does not have that capability. Remembering that CONMIN requires a complete once-through analysis in order to collect enough information to make a design decision, it was necessary to somehow bypass such mathematical instabilities in order to keep the program operating. Yet the analysis still had to yield reasonable results from the given design in order to obtain meaningful gradients. This prompted the formulation of mathematical relationships to create "penalty" constraints which, if properly written, would indicate to the optimizer that a function or thermal property has violated its physical limits. However, not only would penalty constraints have to be defined, but a "fix up" or "correction" of the violated property or variable would be required in order to allow the analysis to continue. A good physical understanding of the inter-relationship between condenser parameters and the thermal processes resulting from the condenser design is

necessary so that the "fix up" of the violated property would still yield fairly accurate condenser information on which CONMIN could base its search for the optimized design.

For example, a condenser is usually designed around a given steam load. If the condenser has too many tubes, is too long, or coolant flow is too great, then the condenser will be overdesigned. There will be dry tubes within the condenser as all the steam is condensed before steam flow reaches the inner void. In CONDIP this means that zero or negative steam flow will be encountered in the analysis.

If, on the other hand, tube surface area is too small, coolant flow is inadequate or the condenser tube spacing is too tight and is choking the steam flow, then one of two things will happen. Either a quantity of uncondensed steam will make it completely through the condenser, or steam pressure loss in the condenser will cause steam pressure to drop below zero. In addition, there are two reasons why all the steam might not condense. It could be simply due to insufficient heat transfer surface area or it could be because the saturation temperature of the steam has dropped below the coolant inlet temperature. If the latter situation occurs, then there is no driving force for heat to transfer from the steam to the coolant. There is only one way that this situation can occur: if steam saturation pressure drops below some value indirectly determined by the coolant inlet temperature. In any event this condenser is certainly underdesigned and not capable of supporting the required steam load.

As stated earlier, the purpose of optimization is to obtain the best, feasible design. Thus, an understanding of the relationship between the physical characteristics of a given design and the subsequent thermal performances will certainly help in defining the appropriate penalty constraints and their corresponding limits. It will also

aid in the determination of appropriate "corrections" when those limits are violated. It is important to note that with the introduction of these penalty constraints, the definition of a feasible design is revised. A feasible design is now defined as one in which thermal properties and functions are not allowed to violate their physical limits, as well as other design constraints, anywhere in the condenser.

In CONDIP there are three basic thermal properties which could create the above mentioned problems if they fall below a certain value. They are steam saturation pressure, steam flow and steam temperature. Because of the direct relationship between steam saturation pressure and temperature, it was possible to deal with them simultaneously. The solutions that were developed to overcome the effects of these thermal violations determined the extent which CONDIP would optimize.

1. <u>Steam Flow Effects</u>

One source of mathematical instabilities within the program is if steam flow over the tubes falls to zero or below. It is intuitively obvious that steam flow can not physically fall below zero and that in order to keep the program running the steam flow rate must be kept greater than zero. However, correcting for this alone would certainly alter the results of that particular condenser design, perhaps even imply a feasible design.

To indicate to the optimizer an infeasible design was actually encountered - one in which steam flow had dropped below zero - the penalty function WTST was created. Since the condenser analysis is performed sector by sector and assuming there are J sectors in the condenser, then there had to be arrangements for J penalty constraints. This prompted the creation of the array, WTST(J), representing constraint penalty functions for each sector. The absolute

magnitude of these functions were directly related to both the severity of the steam flow violation and the number of dry tubes remaining in the sector. WTST(J) ranged in value from negative infinity to zero, where a value of zero represented a condenser design in which no flow violations occurred. Thus, WTST(J) were constrained functions whose lower limit was zero. For example, during an analysis, if steam flow was determined to fall below zero, then WTST for that sector would be given some negative value. In subsequent designs, as the number of dry tubes approached zero and better designs were obtained, then the magnitude of the penalty function approached zero indicating no constraint violation.

Since penalty constraints are entirely contrived relationships with no real physical basis, it is desirable to minimize their number to avoid the possibility of sending innacurate signals to the optimizer.

To eliminate the need to use the WTST penalty functions as constraints, the values of WTST(J) were consolidated at the end of the sector analysis into the condenser steam exit-fraction constraint. Normally, steam exit-fraction ranges from zero to one and is simply equal to:

$$\frac{\text{steam leaving the condenser}}{\text{steam entering the condenser}}$$

By incorporating the WTST violations into steam exit-fraction, the exit-fraction was made a continuous function ranging in value from negative infinity to one. The negative steam exit-fraction represented a partially dry condenser and its magnitude was in direct proportion to the number of dry tubes. Thus instead of having to evaluate and calculate gradients for J number of WTST(J) contraints, the optimizer simply had to evaluate a previously defined and now expanded exit-fraction constraint.

It should be stressed that making steam exit-fraction continuous through zero was equally as important as eliminating the need for additional constraints. It can be reasonably assumed that for all practical condenser applications, exit steam fraction will always be limited to some positive number near zero. Here is where one applies the physical knowledge of the condenser and its relation to the thermal property of steam flow. As explained earlier, dry tubes represent an over-designed condenser. Thus the natural tendency is for the optimizer to alter those design variables so as to create a more compact condenser. As this occurs, steam exit-fraction will naturally increase. The upper active limit of that constraint will determine the optimum feasible design. While it is not necessary to have a lower limit for steam exit-fraction, it is very important for it to be a continuous smooth function especially in the region near the upper limit. It is therefore critical to properly define the penalty functions WTST(J) in a way so as to provide a smooth transition from the negative, artificial values of negative steam exit-fraction to the real, positive values.

Since the steam flow penalty functions will not be used as constraints, the analytical results will provide gradient information to the optimizer. However, once steam flow has been determined to fall below zero, steam flow for that first dry row of tubes and all subsequent rows must be fixed up with dummy values to allow the program analysis to continue. How that "fix-up" is accomplished will ultimately determine the search direction for the optimizer.

Physically, once steam flow has gone to zero, there should be no further latent heat transfer, no subsequent condensate production, and further pressure losses should be only due to the flow of non-condensible gases. It is necessary to make the computer generated analysis reflect as

closely as possible these physical realities. Since the optimizer no longer has the penalty functions to use in calculating a search direction, other constraint values obtained from the analysis will dictate the next search direction. Gradients will also be calculated using these results and the determination of the next search direction will incorporate these gradients as well. In the case of negative steam flow, steam flow and condensate production over dry rows were given nominal values which were as small as the computer analysis would tolerate. These extremely small values closely approximate zero steam flow and generate results which resemble physical reality as closely as possible.

The following example is provided to better illustrate the logic used in CONDIP to handle negative steam flow. CONDIP determines flow rate through each row in each sector. During a sector analysis, CONDIP calculates the condensate generated at a given row and subtracts that value from the steam flow entering that row to calculate the steam leaving. The exiting steam flow rate is then checked to determine whether steam flow has gone to zero. If it has not, then the analysis continues. If it has, then the following two events occur.

The penalty function, WTST(J), is calculated for that sector and dummy values are inserted for steam flow and condensate rate at the row where the violation occurred. For the remainder of the analysis condensate generation and subsequent steam flow calculations are bypassed and the remaining rows in the sector are fixed up with dummy values for steam flow and condensate. The analysis continues utilizing these dummy values in all appropriate heat transfer and pressure calculations. At the conclusion of the sector analysis, the values of the penalty functions, WTST(J), of each sector are incorporated into the steam exit-fraction.

If the analysis revealed zero dry tubes then WTST(J) for all sectors would be zero and the steam exit-fraction would simply be calculated as:

$$\frac{\text{steam leaving the condenser}}{\text{steam entering the condenser}}$$

If, however, dry tubes were encountered in the condenser analysis then WTST(J) of some or all the sectors would be negative and dependent in magnitude on the number of dry tubes in each of the J sectors, as well as the severity of the steam flow violation. Steam flow leaving any sector which has gone dry would be zero and steam exit-fraction would be evaluated as:

$$\frac{\text{steam leaving any wet sectors}}{\text{steam entering the condenser}}$$

plus a weighted value of all the WTST(J) penalty function values. Using the relationships just described, it is apparent that steam exit-fraction: is negative if condenser tubes are dry; approaches zero as the design becomes feasible; and is greater than zero if there is steam leaving the condenser.

2. <u>Steam Pressure and Temperature Effects</u>

The other possible source of mathematical instability occurs when steam pressure falls below some preset limit. If pressure falls to zero, numerous mathematical singularities will be generated. Yet, before this situation can occur steam temperature will have already fallen below inlet coolant temperature causing singularities in the log mean temperature difference (LMTD) heat transfer calculation. Thus, the lower pressure limit which cannot be physically exceeded is not zero but the minimum saturation pressure established by the inlet coolant temperature. In CONDIP, this lower limit is given the variable name, PTLIM.

As steam flows through the condenser, pressure continually decreases due to friction losses and therefore, it is evaluated at each row in each sector. When the steam saturation pressure drops below PTLIM, indicating a physical violation of realistic limits, then the creation of a penalty function and a corresponding "fix up" of saturation pressure is required to allow the program to continue. The treatment of the problem was therefore analogous to the previous situation dealing with negative steam flow.

PTST(J) was the penalty function devised to indicate to the optimizer that the pressure limit, PTLIM, was violated in any of the J sectors. Values of these constraints ranged from negative infinity to zero, depending on the degree and location in the condenser of the violation. Since pressure is calculated on a row by row basis in each sector, the magnitudes of the pressure penalty functions were directly dependent not only on how much the calculated pressure dropped below PTLIM, but also on the number of rows remaining in the sector. Thus, as the condenser approached a feasible design the PTST(J) consraint values approached zero, indicating lessening violation of the minimum pressure.

As emphasized earlier, it is important to minimize the number of constraints, not only to avoid the possibility of sending confusing signals to the optimizer but also to reduce cost and improve program efficiency. This was accomplished here by inserting dummy values not only into the violated pressure variables but also associated thermal properties such as condensate generation, heat transfer coefficients and heat transfer rates for the row where the violation occurred and all subsequent rows in the sector. The dummy values were chosen such that realistic gradient information would be sent to the optimizer. The proper choice of "fix-up" values for these varaiables resulted in

the elimination of penalty functions as design constraints, and provided sufficient information to determine subsequent search directions.

It is necessary to understand the influence that steam pressure and temperature exert on the overall condenser analysis. With this knowledge it will be easier to predict the physical designs which could cause violations of the pressure limit. PTLIM is violated due to excessive steam flow pressure losses. As explained earlier, these large pressure losses would result from large steam velocities that are found in condensers which are too tightly designed. Thus, the particular condenser design is incapable of handling the required steam load, implying an infeasible design. Understanding this relationship will aid in choosing the appropriate "fix-up" values which will indicate to the optimizer that when the pressure limit is violated an infeasible condenser has been designed.

Physically, When steam temperature falls below coolant inlet temperature (PTLIM is violated) there is no heat transfer from the steam to the coolant and no additional steam is condensed. These physical realities must be reflected in the condenser analysis. Therefore, in subsequent rows, condensation and heat transfer rates were set equal to zero. Since there is no further condensation, the steam exit-fraction is equal to the steam flow at the point of violation divided by the total flow into the condenser. Thus PTLIM indirectly determines the exit steam fraction of the infeasible design. This relationship between exit-fraction and the PTLIM violation is what makes the penalty constraints obsolete. If PTLIM is violated early in the steam's passage through the condenser, steam exit-fractions will be large, violating its upper constraint limit and thus reflecting an underdesigned condenser. As the condenser design improves, then exit-fractions will decrease.

Physically, this can only be accomplished if the condenser design "opens up", reducing pressure losses in the condenser. Consequently, as condenser designs become larger, steam exit-fractions decrease and the condenser is driven towards a feasible design.

The following example illustrates the logic employed by CONDIP to handle steam pressure and temperature violations within the analysis. Condenser inlet flow is divided by the number of sectors in the condenser. Condenser inlet saturation pressure is determined by the steam inlet temperature. Entrance pressure losses are calculated and subtracted from the inlet pressure. The resulting pressure is checked against PTLIM and a violation at this point indicates a totally infeasible condenser in which no steam is condensed. Steam exit-fraction will thus be equal to one. If the saturation pressure is greater than PTLIM the analysis continues row by row through the sector. Pressure losses over each row are calculated and subtracted from the row inlet pressure to determine pressure into the next row of tubes. If this next-row steam pressure is determined to fall below PTLIM, then a thermal violation has occurred requiring "fix up". Subsequent rows are made to indicate zero condensate generation and zero heat transfer. Steam flow over the remaining rows is maintained at a constant value, which will subsequently be used to determine steam exit-fraction. Pressure variables over the remaining rows are given small positive values just large enough to allow the analysis to continue. Although all heat transfer and condensate calculations will be bypassed, the analysis must be allowed to continue so that pressure losses will continue to be calculated based on the steam flow at the point of violation. This is important since steam flow adjustments to the sectors are based on certain pressure comparisons between the sectors. The cumulative sum of all row pressure losses

in each of the sectors must be equal to within some tolerance. If they are not then steam flow into each of the sectors is altered so that the exit pressures from each sector converge to some common value. Thus, an accurate reflection of true pressure losses is important to this calculation.

The value of the steam exit-fraction is again determined to be the single constraint necessary to drive subsequent condenser designs to a feasible optimum configuration. The pressure penalty constraints proved to be superflous information, but the corresponding variable "fix-up" was critical in the determination of search direction.

C. LIMITATIONS

During the development of CONDIP, it became apparent that steam exit-fraction would become the key constraint during optimization of any objective function. A feasible design implies that steam exit-fraction is a small positive number perhaps somewhere between zero and 0.1 percent. As explained earlier, violations of either steam flow or pressure physical limits resulted in penalty functions and variable "fix-up" which were later directly or indirectly incorporated in the calculation of steam exit fraction. Thus any feasible design, let alone the optimum one, centers on the limits placed upon this design constraint. Any number of design variable combinations will yield a feasible design, and each design variable affects steam exit-fraction differently. The intertwined, complex calculations used to ultimately determining exit-fraction are done by sector and row with each design variable repeatedly playing a factor. For example, the profound effect of both vapor shear and condensate inundation on the shell-side heat transfer coefficient and consequently steam exit-fraction, is

indirectly determined by numerous design variables. However, their effects are impossible to predict. The cascading effect of the thousands of calculations performed during the course of a design analysis is to ultimately create a single, highly non-linear variable in the form of the steam exit-fraction, upon which design decisions will be made.

As more design variables were involved in the analysis, the optimizer had difficulty determining their often conflicting effects on both the objective function and the steam exit-fraction. A small perturbation of each of the design variables independently would yield gradients indicating design improvement. But when these gradients were evaluated simultaneously to actually determine the direction of the subsequent design, their combined effect would actually indicate either no improvement of the objective function or a violation of the steam exit fraction design constraint. The end result would be that the optimization process would stall as no feasible search direction could be obtained. Larger perturbations to the design variables were required to properly evaluate their relative effects on the objective function and any active or violated constraints. This would enable the optimizer to overcome either small inconsistencies or discontinuities in the objective function and the constraint functions which would otherwise prevent the optimizer from reaching the optimum design. This was accomplished during data input by changing the normalized finite difference step from 0.01 to about 0.1. Increasing the finite difference is not without its drawbacks. As the optimum objective value is approached, the optimizer overlooks the subtle effects of small changes in the design variables because of the relatively large perturbations. Thus, depending on the initial design variables, the optimizer will improve the design to some point near, but necessarily at, the optimum.

When a design becomes feasible, steam exit-fraction will always become an active constraint. But the stated goal is not in achieving a feasible design but in driving the design to a feasible optimum. However, this iterative process can not be accomplished at the expense of violating a constraint and it was here that further complication was introduced. The initial impetus in any optimization process is to first obtain a feasible design. However, once the very small steam exit-fractions are obtained that are necessary for a feasible design, the exit-fraction becomes extremely sensitive to any further design variable changes. Thus any effort to further improve the current design could easily cause exit fraction to increase. Even slight increases would be perceived as violations of the constraint limit and thus prevent further optimization from the first feasible design. There are two possible solutions to this problem. Either increase the upper limit on the exit-fraction constraint or redefine the constraint. COPES formulates the general constraint function in such a way as to allow the user to increase the active region about the constraint limit. This is accomplished here by increasing the normalization factor in the following expression for the exit-fraction constraint function:

$$\frac{BU - EXITFR}{SCAL1} \leq 0.$$

where BU is the upper constraint bound, EXITFR is the exit-fraction constraint value and SCAL1 is the normalization factor for this constraint. Increasing the normalization factor reduces the optimizer's sensitivity to constraint violations by enlarging the range of constraint values in which the constraint is active. This enlarges the region of feasibility and allows the optimizer more flexibilty in altering design variables by reducing the risk of violating

the constraint. The overall effect is that the design optimization can continue but at the expense of accurate constraint limits. The normalization factor used effectively for the exit-fraction in CONDIP analysis was approximately 0.1.

One of the stated objectives was to create a robust program which would consistently yield the single best optimum design independent of the initial design and not get hung up on a relative optimum. As it was explained earlier, although relative optimums represented design improvement, they also indicated the inability of the optimizer to locate the single best or true optimum design. However, the objective was achieved for only three design variables. When more than three design variables were used, the optimum designs became loosely dependent on the initial input, although not in any predictable way. This is not to say that the condenser design did not optimize. By incorporating the finite difference and scaling normalization on exit-fraction as described above, final designs did yield objective function values which were continually within about ten percent of the true optimum regardless of the initial design. However, there was just no guarantee that the single, best optimum design could be consistently obtained. In summary, the reasons why CONDIP did not consistently optimize to the single, true optimum were: the extreme non-linearity of the steam exit-fraction, the need for a large finite difference gradient, and the need for a normalization factor for the exit-fraction upper limit constraint.

While the optimum design solutions obtained from CONDIP may be sufficient, there are several ways to improve the results and increase the chances of obtaining the best possible design. The easiest way is to try several initial input values until the user is satisfied that the best solution has been obtained. The problem with this approach is

that it is both costly and time consuming. A second recommendation is to couple an extremely simplified version of the condenser analysis with the optimizer to obtain a educated guess as to what the optimum design should be. The results of this analysis could then be used as input for CONDIP. OPCODE1, which utilizes the HEI methods in its analysis, is a likely candidate. The advantage of this approach is that a quicker, cheaper analysis can be used to obtain a rough idea of the anticipated optimum design. CONDIP can then use these design results to obtain even better and more accurate solutions, faster. There is still no guarantee, however, that the true optimum will be solved. Perhaps with the development of more robust and versatile optimizers, ones which uses numerical techniques and methods that are better suited to this type of problem than CONMIN, more precise solutions can be obtained. However, there is little more that can be done to simplify the analysis of the steam exit-fraction and subsequently linearize the problem.

IV. DESCRIPTION OF THE MAIN AND SUPPORTING SUBROUTINES

A. MAJOR SUBROUTINES

The following section contains a brief description of the major subroutines in CONDIP. The appropriate flow diagrams are also provided to better illustrate and complement the explanations. For further information concerning the various subroutines and functions see the CONDIP listing in Appendix C and ORCON1 [Ref. 2].

1. ANALIZ

This subroutine basically arranges CONDIP in a standardized form which is compatible with COPES/CONMIN. COPES uses a variable flag, ICALC, to coordinate the optimization process with ANALIZ. Utilizing this flag ANALIZ then calls the input, analysis and output portions of CONDIP as required. When COPES sets ICALC equal to one ANALIZ reads in all initial input. This is the only time any input can be entered. When ICALC equals two, COPES works with CONMIN to optimize the design. ANALIZ makes available the analysis portion of the program to be used repeatedly by CONMIN. When COPES sets ICALC equal to three, the optimization is complete and ANALIZ calls all applicable output subroutines. Figure 4.1 illustrates the flow process for ANALIZ.

2. INPUT

This subroutine enters all initial input of data by which the initial design is determined. The resultant design may be either feasible or infeasible, subject to the limitations previously discussed, so it is not critical what values are initially assigned to the design variables.

However, the initial input is screened to prevent the introduction of totally unrealistic values of variables into the program. For example, initial tube thickness, tube inner diameter, tube number and tube length are all checked to ensure that their values are greater than zero. If any of the screened initial inputs do not satisfy the minimum requirements, then the program exits prior to entry into the optimizer. The limits of the design variales and constraints will prevent similar situations from occurring during the analysis. Figure 4.2 presents the flow diagram for the INPUT subroutine.

3. OUT3

This subroutine simply prints all the initial values entered in the INPUT subroutine.

4. ORCON

This subroutine calculates the bundle geometry, flooding factors and such coolant flow parameters as pressure loss, flow rate, and pumping power. There are two options available to determine bundle geometry, each with certain advantages and disadvantages.

Option1: The number of rows is entered as a constant and the tube number is determined based on pitch, tube outer diameter, and row spacing. The advantage of this method to determine bundle geometry is that it allows the user to linearly vary pitch and/or tube inner diameter by row. The disadvantage is that tube number is a dependent variable. The optimzer is therefore limited in determining the optimum design by the specified number of rows.

Option2: The number of tubes can be used as a design variable while the number of rows is determined by tube number, row spacing, pitch and tube outer diameter. There

is more flexibility in this method of condenser design but it is not possible to linearly vary tube pitch and inner diameter. The condenser bundle is generated from a specified inner void out, and all the appropriate condenser geometry is determined for one of the identical 30 degree sectors. Overall bundle volume is then calculated as is the ratio of tube hole area to tube sheet area.

Once the basic condenser geometry has been determined, the code then proceeds through an algorithm to calculate baffle location based on an input value specifying the number of baffles desired in the condenser bundle. After this has been completed, flooding factors are determined. That is, the number of tubes in a vertical row above the central tube in each row is calculated. This is done for each of the six sectors on one side of a circular bundle. Symmetry is assumed for the other side. These flooding factors are later used in calculations to determine the effect of condensate inundation on shell-side heat transfer coefficients.

Finally, ORCON calculates coolant mass flow, coolant velocity, header pressure difference and pumping power based on the type of coolant flow input received. The flow chart in Figure 4.3 is a simple illustration of the logic used in ORCON. From ORCON, the subroutine SECALC is called.

5. SECALC

This subroutine determines all the parameters of each of the sectors in the condenser by row. The first calculation made in SECALC is the determination of the cooler geometry, if there is one. Entrance pressure losses into the condenser bundle are calculated for each sector and saturation pressure is checked to ensure that it is greater that PTLIM. From this point, much of the remaining

subroutine is comprised of two do-loops with one nested inside the other. The outer loop cycles through however many sectors are in the condenser model. The inner loop cycles through the rows in the sectors. Pressure, temperature, mixture velocity, steam flow and condensate flow are calculated at each sector row. The subroutine HETTRN is called repeatedly to provide the necessary heat transfer information. Pressure and steam flow is checked continually at each row to ensure that neither falls below its predetermined lower limits. In the event that either situation occurs, the appropriate penalty function and fix-up procedure is implemented to enable the analysis to continue. As previously discussed, these values are chosen to reflect as accurately as possible real conditions which would occur when steam flow or pressure violate their physical limits.

Once all the sectors have been analyzed the cumulative steam-side pressure losses from each sector are compared. Steam pressure at the inner void must be uniform, therefore the sector pressure losses are required to be equal within some allowable tolerance. If they are not, then the distribution of inlet steam flow into each sector is altered to force the pressure losses to converge to a single value. Once steam flow to the sectors has been adjusted, the sector and row analysis in SECALC is repeated until the pressure losses within each sector approach a common value. After the pressure comparison has been satisfied, certain overall condenser parameters are calculated such as steam exit-fraction, bundle heat load, and steam-side pressure drops.

Finally, if a cooler is required, the subroutine COOLEX is called. Otherwise the condenser analysis is complete. The flow diagram for SECALC is presented in Figure 4.4.

6. HETTRN

This subroutine is called repeatedly in SECALC to solve for all shell and tube-side heat transfer properties for each row in each sector of the condenser. In particular, values for the overall heat transfer coefficient and log mean temperature difference are utilized by SECALC in computing condensate production and heat transfer rate at each row of tubes.

On entering this subroutine, a series of estimates for certain row variables are calulated. Based on an assumed initial value for the overall heat transfer coefficient, the exit coolant temperature and corresponding film temperature are calculated. Utilizing these temperatures, the LMTD, thermal resistances, individual heat transfer coefficients and numerous other heat transfer parameters are then calculated. Finally, another value for the overall heat transfer coefficient is determined based on the above-mentioned analysis, and this final value is subsequently compared to the initial value. If they are not in agreement, within a specified degree of tolerance, then the initial value for the overall heat transfer coefficient is updated and the entire process is repeated until the initial and final values converge. This iterative process is necessary as temperature dependent heat transfer coefficients, film temperature drops, and exit coolant temperatures are all being calculated simultaneously.

Note that it is in HETTRN that the concepts of vapor shear and condensate inundation are incorporated. Heat transfer coefficients are corrected for both effects based on the calculations presented earlier. Also note that since steam temperature is never allowed to drop below inlet coolant temperature in the calling subroutine SECALC, resultant LMTD calculations in HETTRN will not yield singularities.

Once all the heat transfer variables have been determined, control is returned to SECALC where the appropriate results are utilized and stored. The appropriate flow diagram for HETTRN presented in Figure 4.5.

7. COOLEX

This subroutine solves for all the necessary parameters required in the cooler analysis. The cooler is assumed to be of rectangular cross-secton with the height of the cooler not to exceed the difference between the condenser inner and outer radii. The values used for tube pitch and tube diameters in the cooler are the same as the innermost row of the condenser bundle.

Steam exits the condenser bundle, collects in the inner void and enters the bottom row of the cooler. The steam then proceeds vertically up through the cooler. The physical location of the cooler is not a prerequisite to the subsequent design, although it is expected that the cooler will be placed within the condenser bundle, thus the limit on cooler height.

The first calculation in COOLEX determines the steam velocity at minimum cross-secton in the first row of tubes, VLCMAX. VLCMAX is directly proportional to the amount of steam and non-condensible gas entering the cooler as well as the cooler geometry. Therefore the constrained limits for VLCMAX will play a major factor in the overall condenser design.

Subsequent row analysis is treated identically as in SECALC. However, all pertinent heat transfer data are calculated directly within COOLEX, making it independent of HETTRN. Steam pressure and steam flow are checked at each row to ensure that the appropriate limits are not violated and all thermodynamic parameters are calculated. At the conclusion of COOLEX, cooler performance variables such as

heat load, exit-fraction steam, steam pressure losses and overall heat transfer coefficients are calculated and control is returned to SECALC. The flow diagram for the COOLEX subroutine is illustrated in Figure 4.6.

8. OUT2

This subroutine prints the overall condenser bundle results including heat load, steam exit-fraction, overall heat transfer coefficient, overall condenser LMTD and bundle volume. Normally, OUT2 is called once after the initial design is analyzed and again after the optimum design has been determined. Final design variable values such as tube number, coolant flow, tube pitch, tube wall thickness and tube inner diameter are also printed.

9. OUT2C

This subroutine prints the cooler results as well as the combined cooler/condenser results. This subroutine is called from the subroutine OUT2 and is called only if a cooler is required and subsequently designed. Therefore, these results will always be printed in conjunction with OUT2 output.

10. OUT3

This subroutine prints a very detailed output of the condenser and cooler results by row and sector. Nearly all the thermodynamic and heat transfer properties are presented, thus providing a rather complete picture of conditions everywhere in the condenser. This is extremely helpful in determining, for example, where additional heat transfer enhancement would be most beneficial, or where baffles should be best located to reduce the effects of condensate inundation.

B. SUPPORTING SUBROUTINES

The following is a brief description of supporting functions and subroutines called frequently by the main subroutines.

1. DFSVTY: This subroutine returns the value of the mutual diffusivity of the steam and non-condensible gas present.

2. XTR: This funcion subroutine transforms the calculated data, received in the argument list, to the log values and performs a linear regression on two or more points using the model.

3. AMUFN: This function subroutine calculates the viscosity of the non-condensible gas in lbm/(ft-sec).

4. BMUFN: This function subroutine calculates the viscosity of a saline solution in the range of 0-24 percent concentration and temperatures of 40-210 ^{o}F in lbm/(hr-ft).

5. CPAFN: This function subroutine returns a value for the heat capacity of the inert, non-condensible gas mixed in with the steam in units of Btu/(lbm-mol-^{o}F).

6. CPFN: This function subroutine calculates the specific heat of a saline solution units of Btu/(lbm-^{o}F).

7. CPSFN: This function subroutine calculates the heat capacity of steam in Btu/(lbm-mol-^{o}F).

8. HFGFN: This function subroutine returns a value for the latent heat of vaporization of water in Btu/lbm.

9. PRSDRP: This subroutine returns the shell-side pressure drop across a row of tubes in psia.

10. PSATFN: This function subroutine calculates saturation

pressure of steam as function of temperature. Pressure is returned in units of psia.

11. ROEFN: This function subroutine calculates the density of a saline solution ofconcentration range 0-24 percent and temperature range of 40-300 ^oF. Density is returned in units of lbm/(cu.ft.).

12. SKBFN: This function subroutine calculates the thermal conductivity of a saline solution of concentration range 0-24 percent and a temperature range of 40-300 ^oF. Thermal conductivity is in (Btu)/(hr-f-^oF).

13. TSATFN: This function suroutine returns the value for steam temperature in ^oR given a pressure in psia.

14. VGFN: This function subroutine calculates the specific volume of steam as a function of temperature and pressure. It has units of (cu.ft.)/lbm.

15. SWITCH: This function subroutine reverses the order of a stored array.

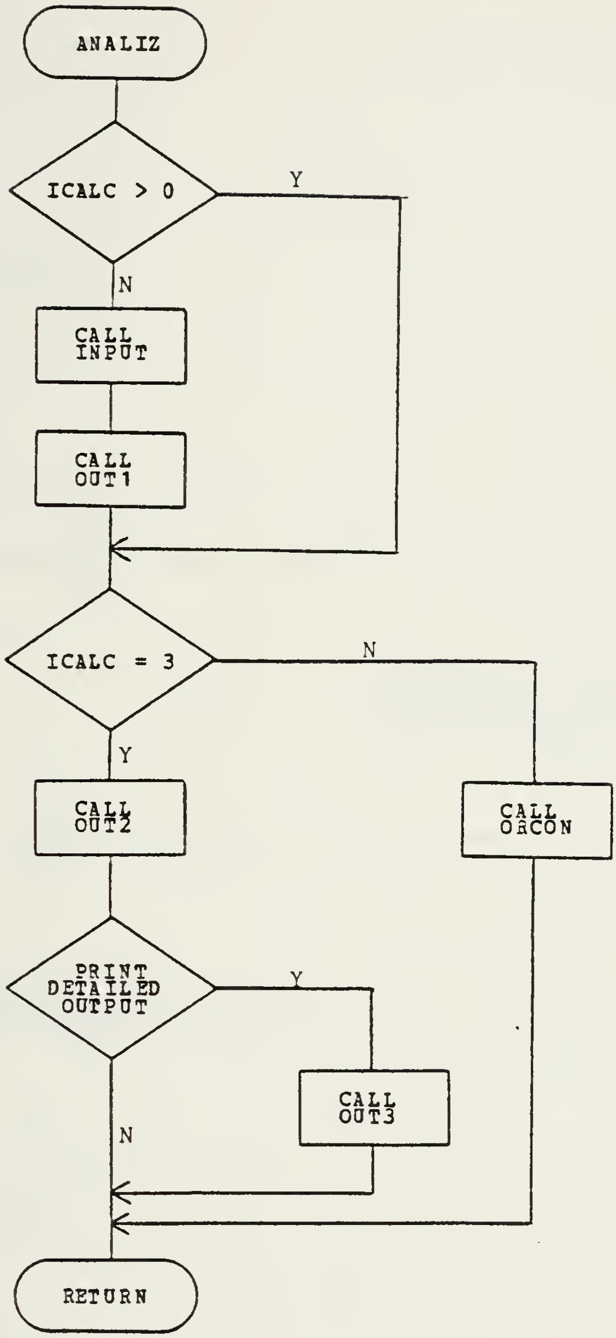


Figure 4.1 Flow Diagram for the ANALIZ Subroutine.

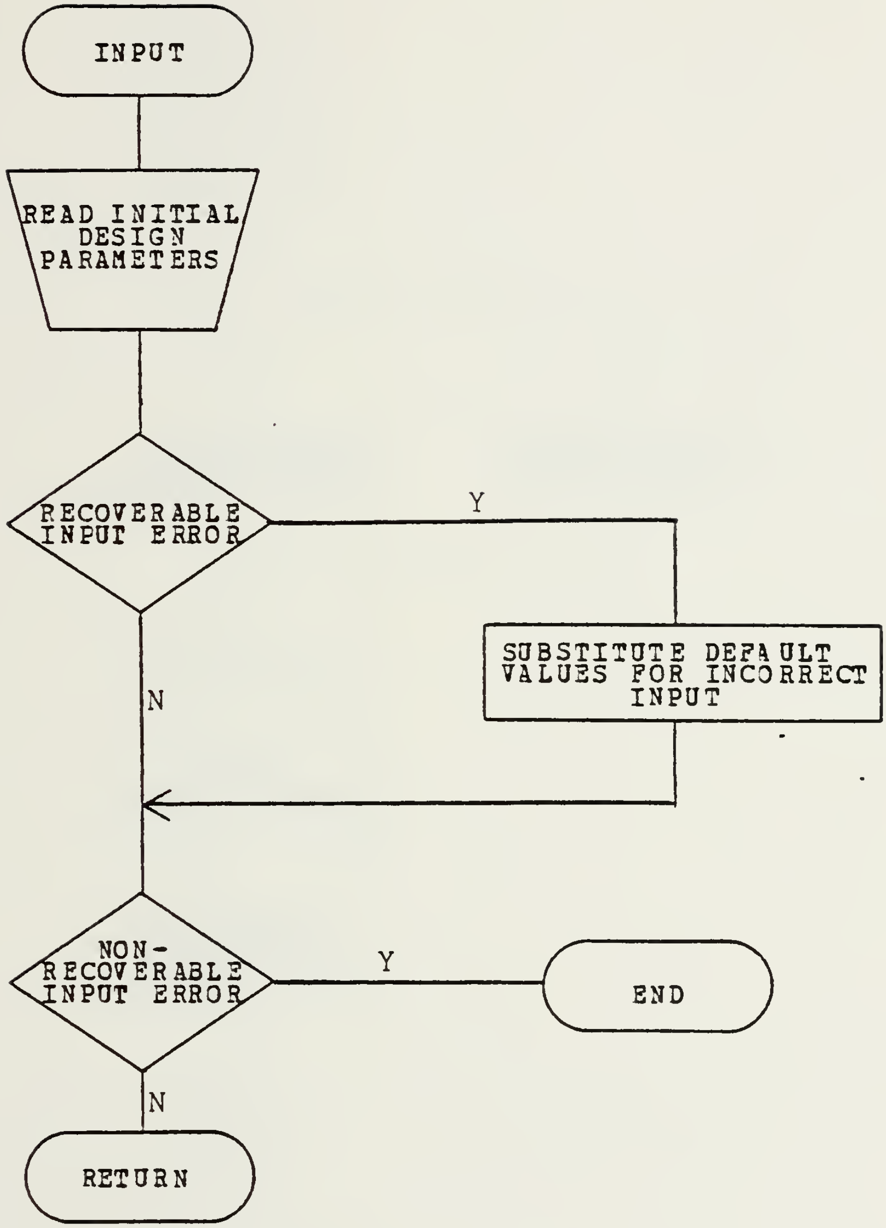


Figure 4.2 Flow Diagram for the INPUT Subroutine.

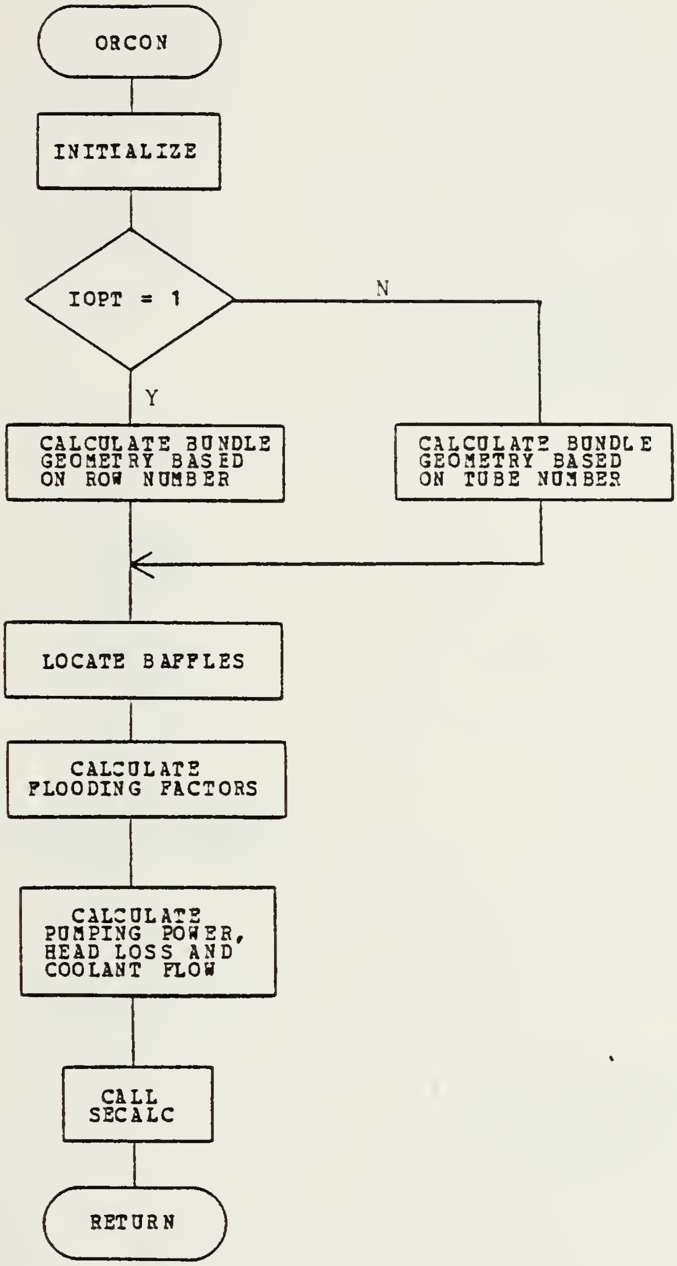


Figure 4.3 Flow Diagram for the ORCON Subroutine.

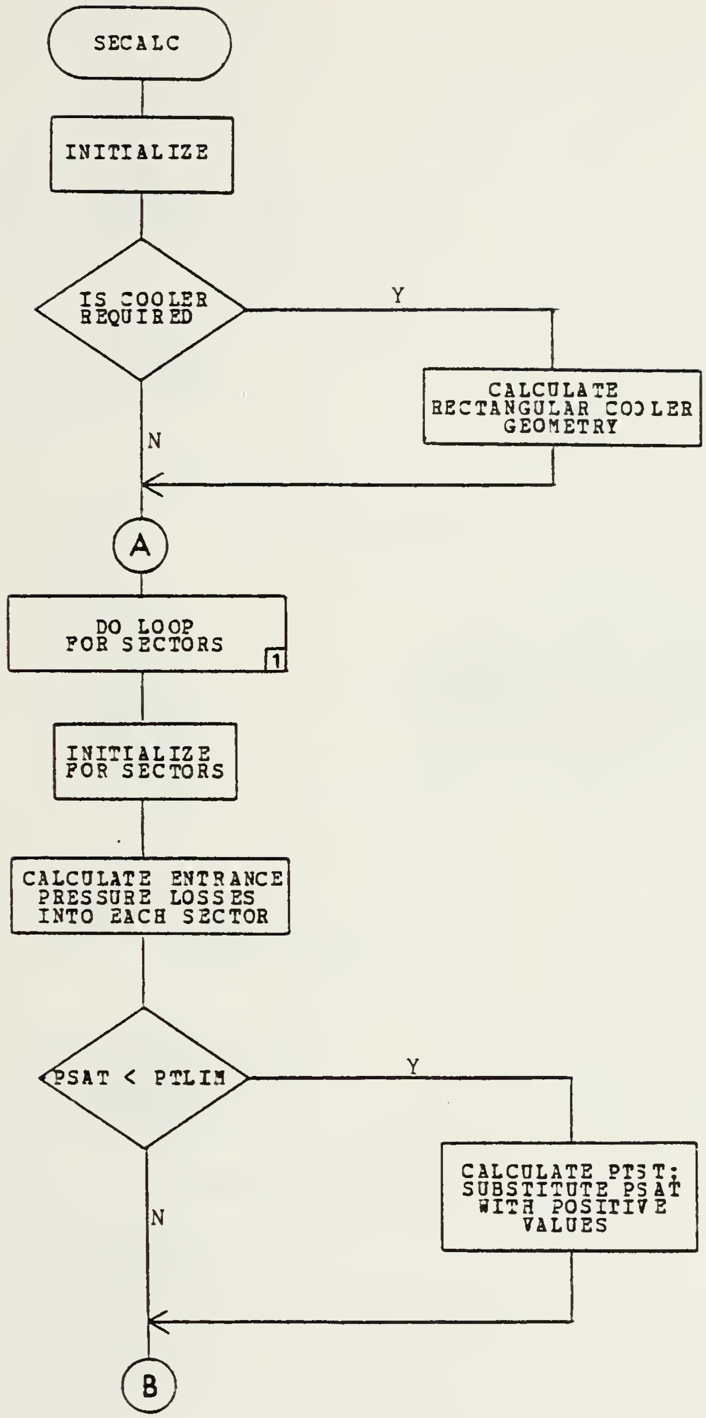
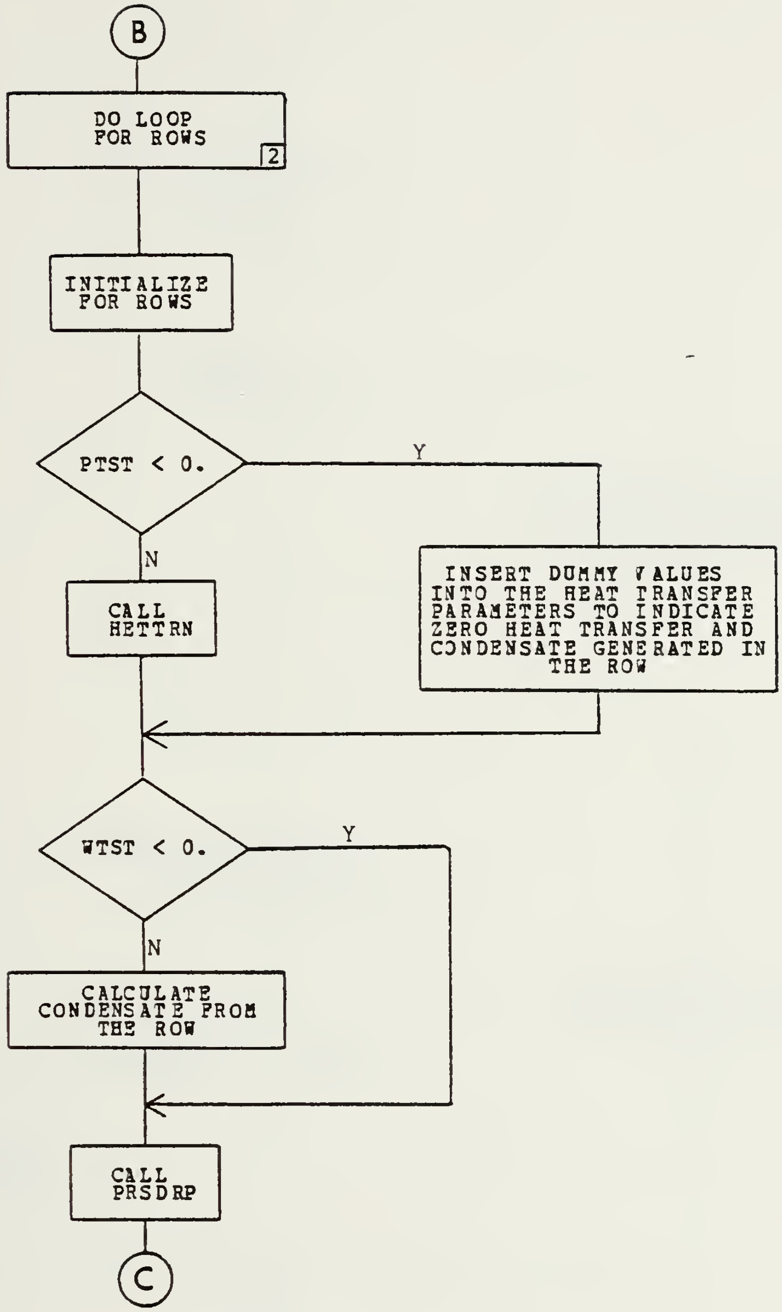
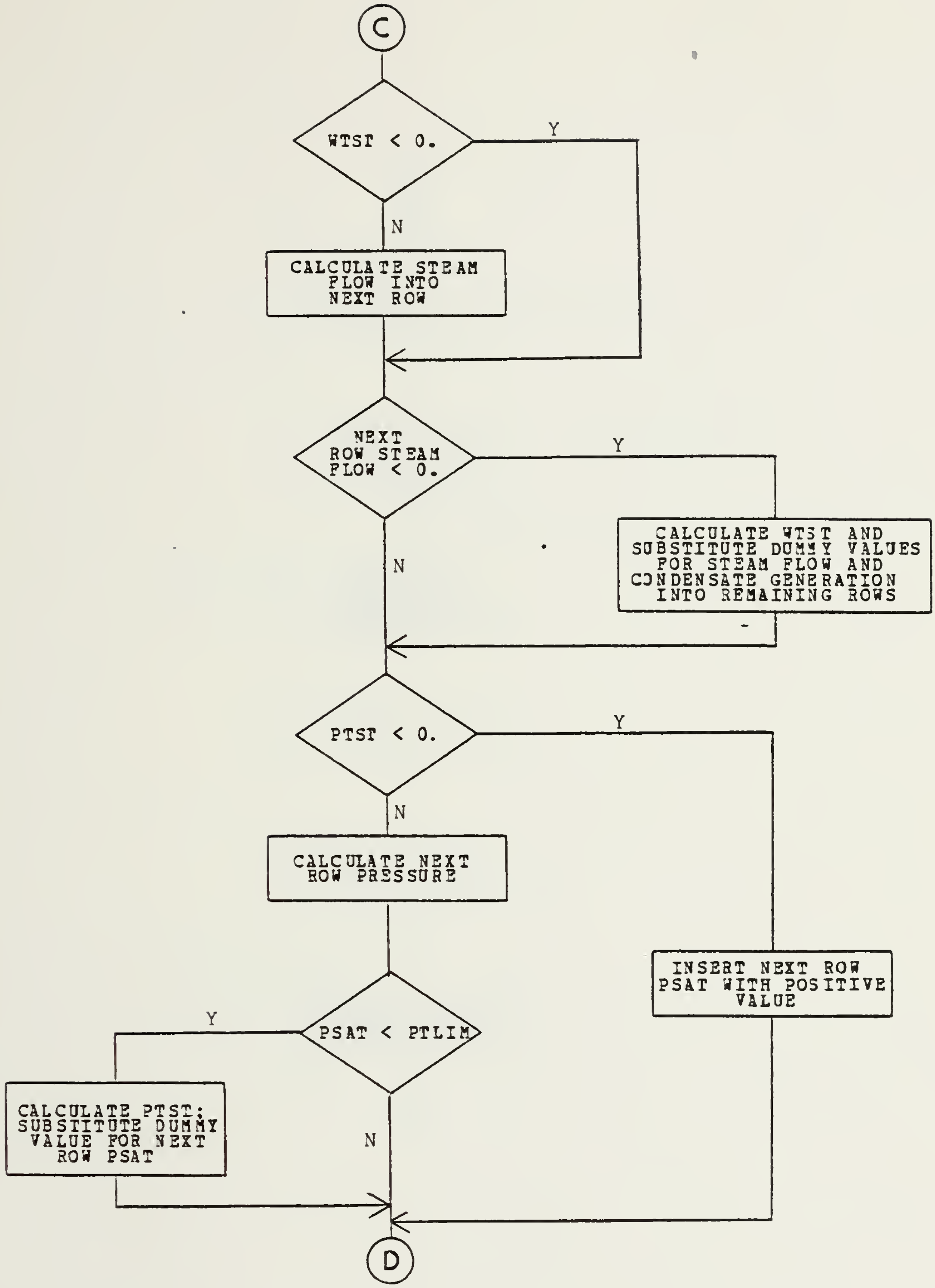


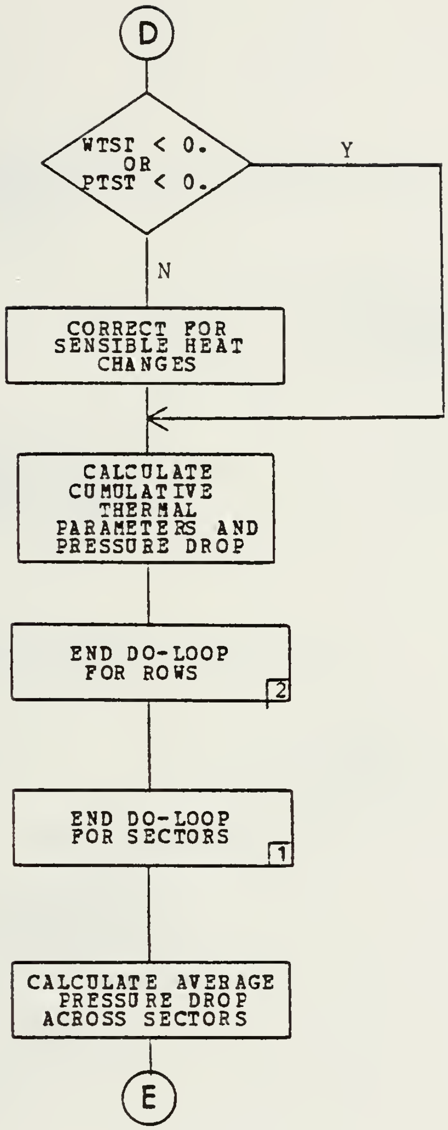
Figure 4.4 Flow Diagram for the SECALC Subroutine.



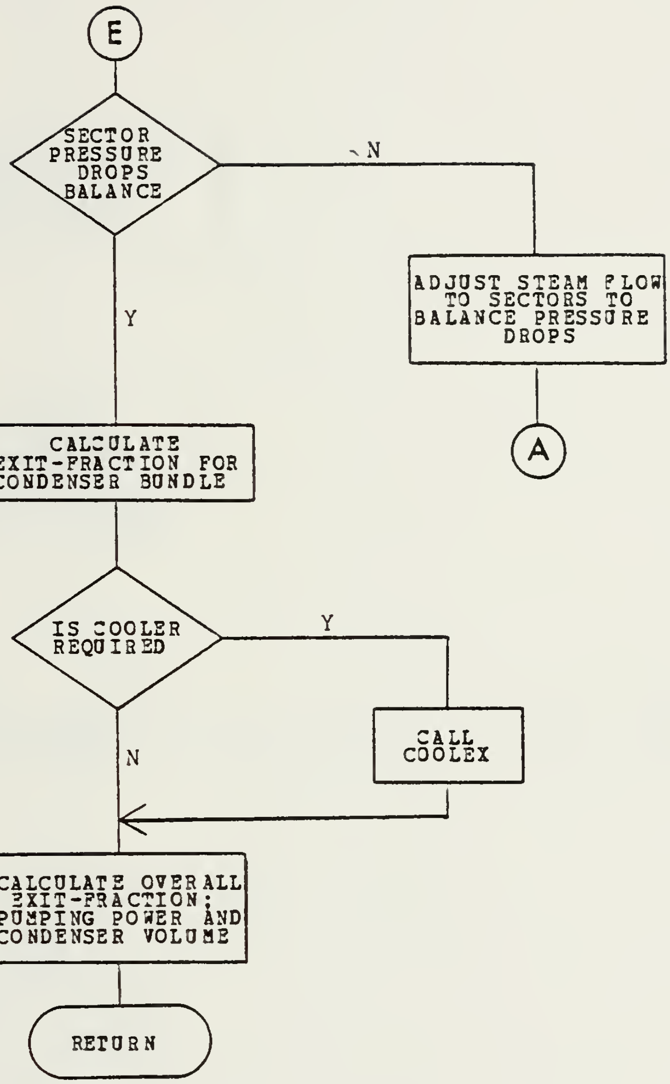
SECALC Flow Diagram (continued)



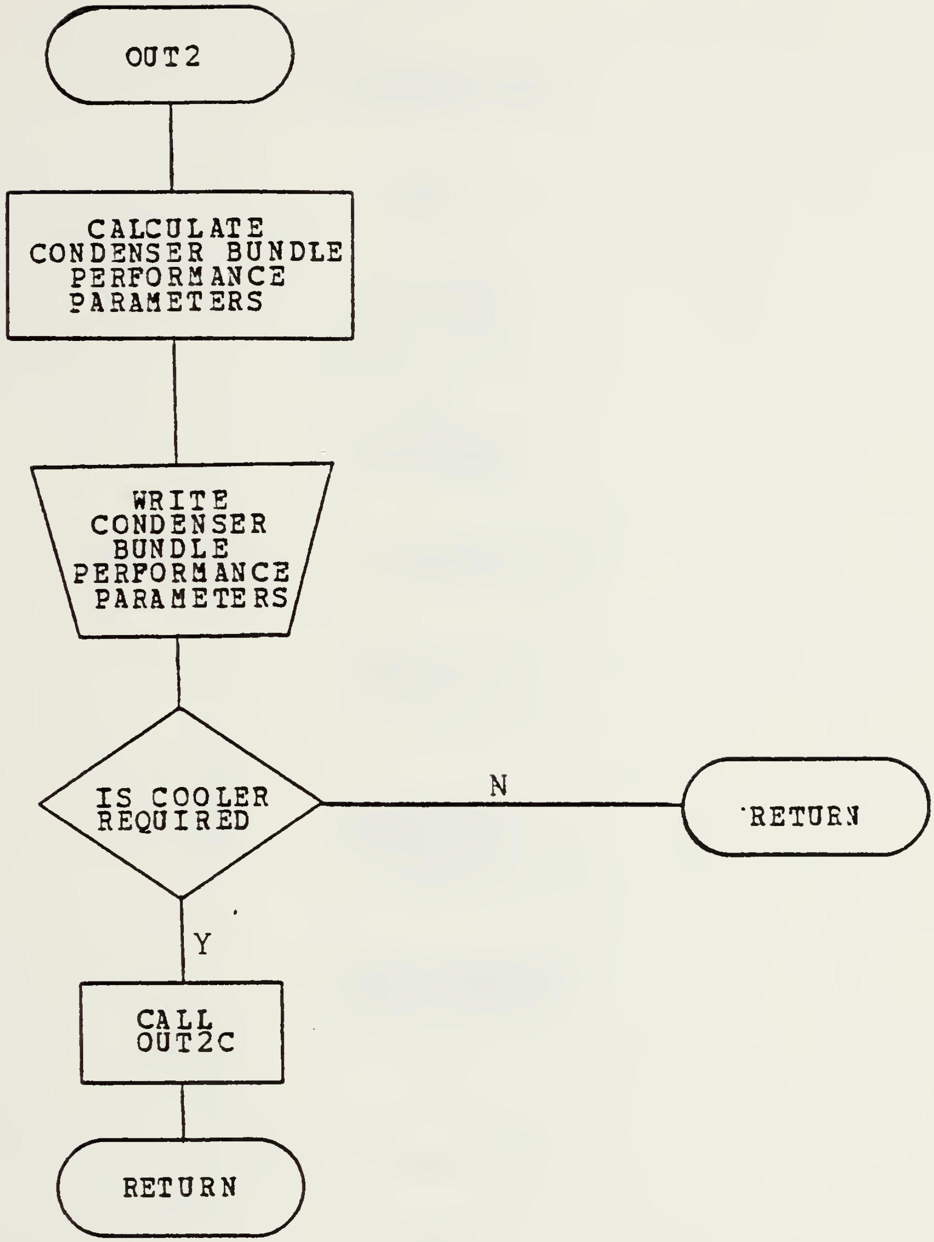
SECALC Flow Diagram (continued)



SECALC Flow Diagram (continued)



SECALC Flow Diagram (continued)



Flow Diagram for the OUT2 Subroutine

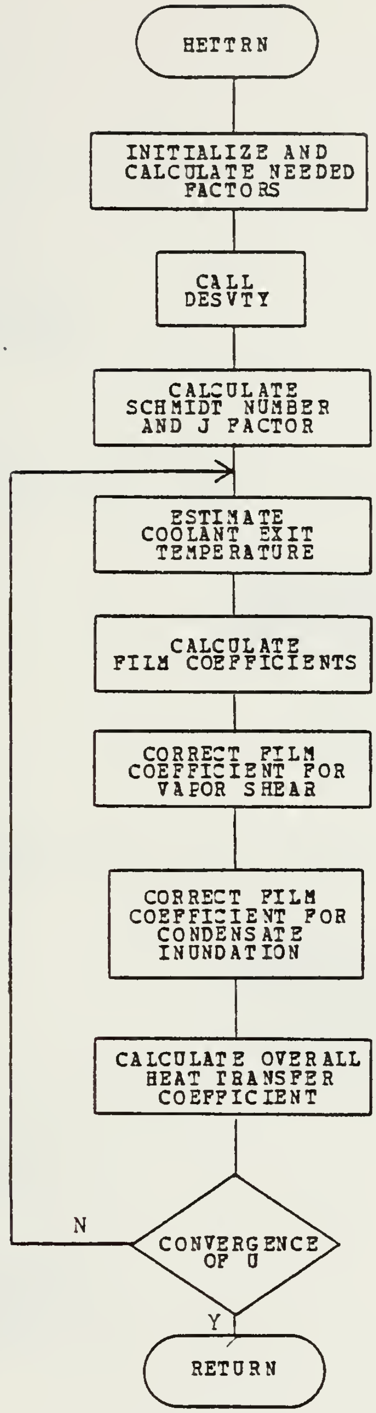


Figure 4.5 Flow Diagram for the HETTRN Subroutine.

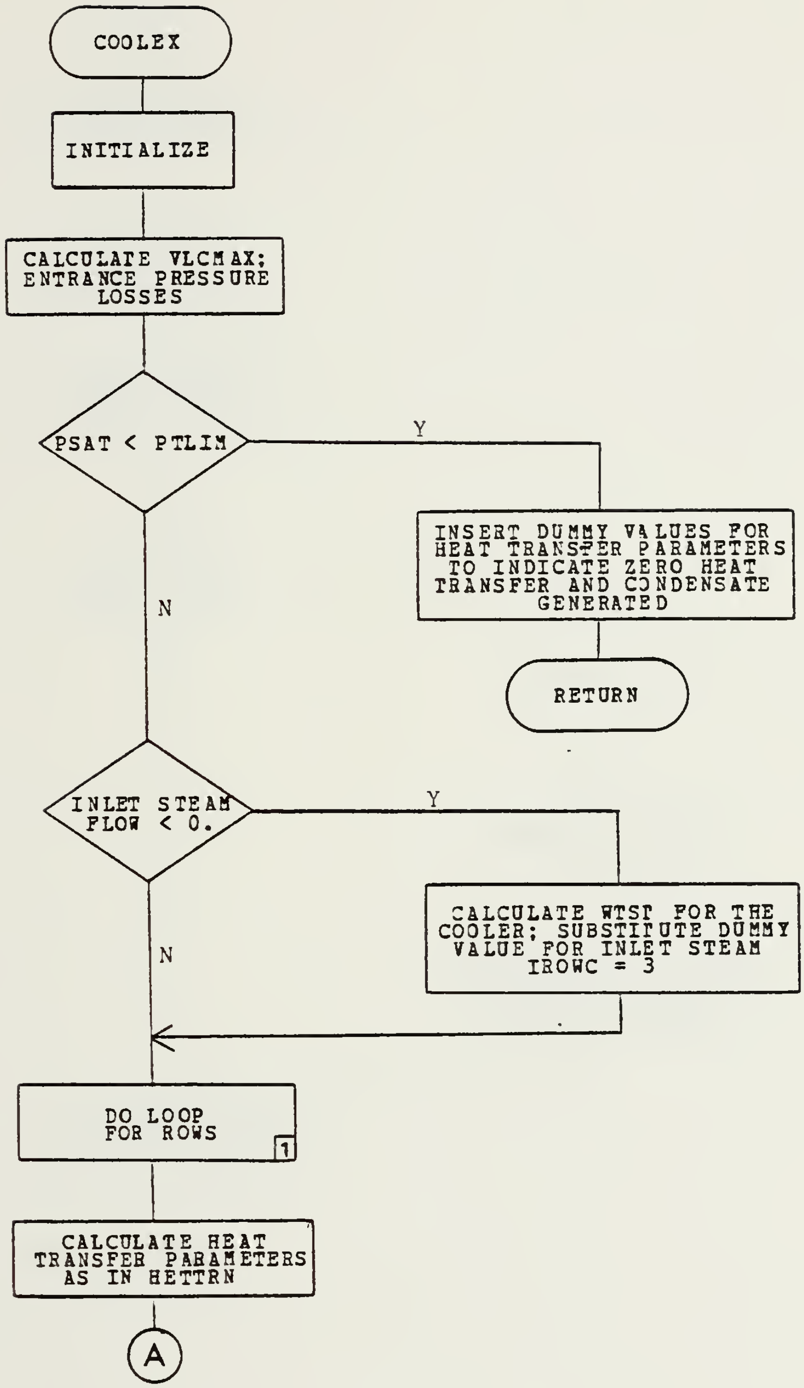
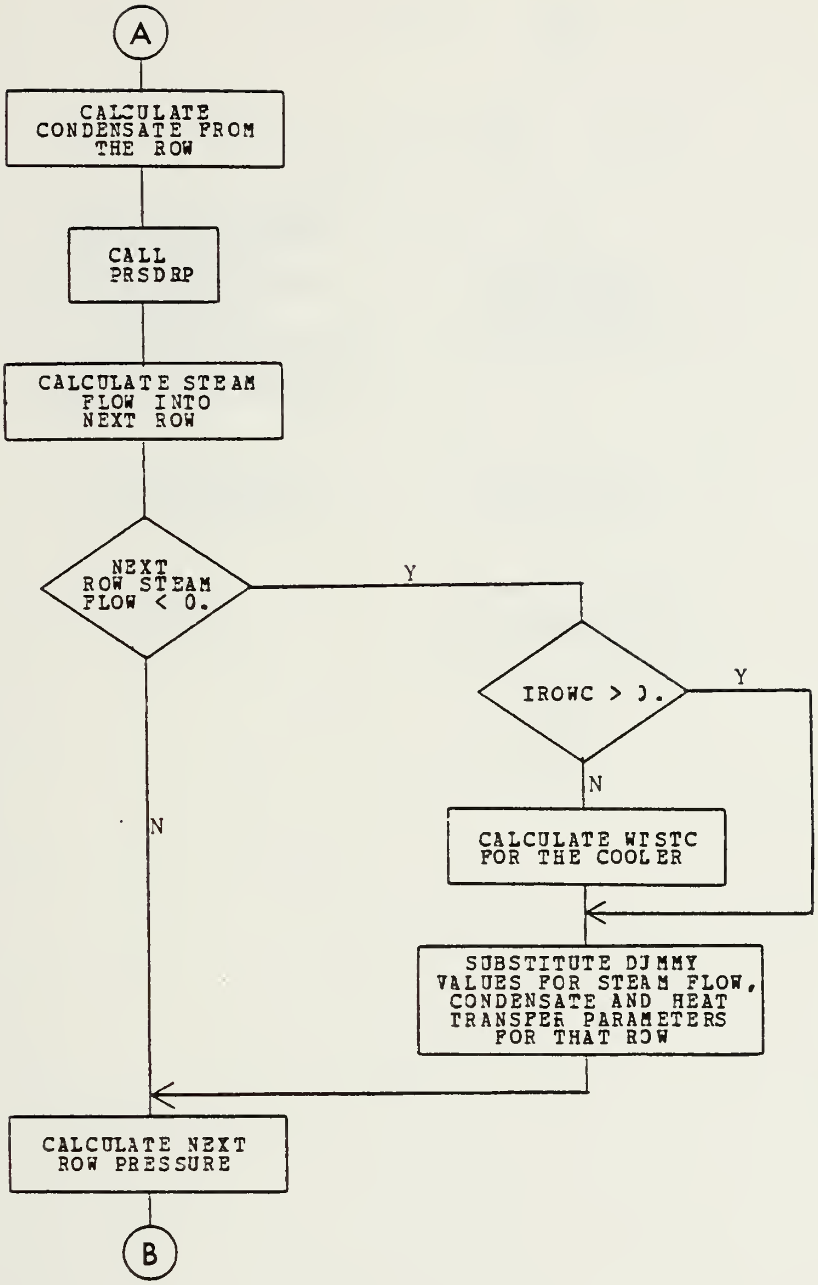
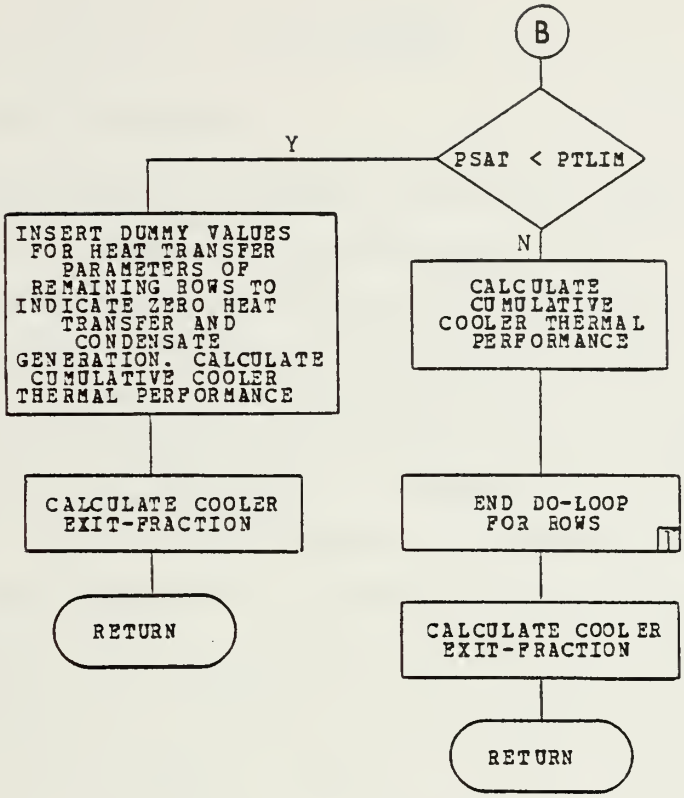


Figure 4.6   Flow Diagram for the COOLEX Subroutine.



COOLEX Flow Diagram (continued)



COOLEX Flow Diagram (continued)

V. RESULTS

A. CONDIP VERIFICATION

It was desirable to verify the single pass capability of CONDIP (i.e. without optimization) as a predictor of condenser performance by comparison with actual experimental data. However, complete and accurate data on condenser design and corresponding performance is not always readily available. Lynch [Ref. 18] encountered this same problem in attempting to verify ORCON1. However, he did manage to locate some actual experimental condenser data, obtained during a test conducted to determine the general performance of the DDG-37 class propulsion machinery [Ref. 16]. The test took place at the Naval Boiler and Turbine Laboratory and was conducted primarily to determine the performance of the turbine and reduction gears. Some limited condenser data was taken as a by-product. The various measurements were obtained as described below:

1. Steam flow measurements were made by weighing the condensate.

2. Cooling water inlet and outlet temperatures were measured by thermometers installed in the inlet and discharge lines.

3. The heat load is calculated based on total steam flow into the condenser multiplied by the difference between inlet steam and condensate enthalpies.

4. Circulating water flow was determined from a heat balance around the condenser. The total heat load was divided by the circulating water heat capacity and temperature rise to obtain flow rate.

5. Condenser inlet pressure was determined by pressure instruments located above the condenser inlet flanges.

6. Non-condensible gas flow was measured by a flowrator.

7. Pressure at the air ejector was measured directly. This pressure and the inlet pressure determined the pressure drop across the condenser.

It should be pointed out that this data were not recorded with the care that normally accompanies scientific data collection. Neither the instruments nor the techniques employed were particularly accurate. The possibility that this observed data are in error casts a cloud over the credibility of the corresponding condenser performance, which was calculated based on those values. However, for a lack of better alternatives, this data and the resulting condenser analysis will be used to determine the reliability of CONDIP.

The DDG-37 condenser geometric design variables obtained from the technical manual [Ref. 15] and input parameters corresponding to a full speed run are presented in Table I. An attempt was made to repeat the design using CONDIP. The results of CONDIP's proposed design as well as the experimental performance are presented for comparison in Table II. Percentages were calculated to quantify the differences between the actual and hypothetical performance.

Before elaborating on the results of this verification, some notable differences between the two designs must be clarified. First, a specific fouling factor was not determined at the time of the experiment and was therefore not provided. A somewhat realistic cleanliness factor of 87.5 percent (fouling factor of .0002) was utilized.

Second, in the DDG-37 condenser the rectangular cooler appears to be inserted directly into the condenser bundle.

However, in order to accomodate the cooler, the bundle must expand or distort. In addition, a void of some dimensions must be provided-for in the center of the bundle to collect any uncondensed steam and non-condensible gases. Diagrams in [Ref. 15] indicate that the DDG-37 condenser is indeed nearly elliptical in shape with bundle axes of 5.67 and 7.17 feet. Although it is apparent that a void does exist, exact dimensions can not be readily determined from the available information.

CONDIP approximates this design by creating separately a circular condenser bundle and a rectangular cooler, the height of which cannot exceed the difference between the outer and inner bundle radii. A circular void of predetermined size is provided-for when determining the condenser bundle geometry. Subsequent volume calculations are performed on the condenser bundle and cooler separately and the overall condenser volume is computed as simply their sum.

Although CONDIP does not exactly duplicate the geometric configuration of the DDG-37 condenser, it was possible to manipulate certain initial design variables in order to cause CONDIP to develop an approximately equivalent configuration. These variables were chosen because, for small changes in their values, there is a rather significant change in the bundle geometry with relatively small effects on the overall condenser performance. Since there are no specific dimensions provided for the inner void in the DDG-37 technical manual, it was picked to be one of the design variables to be adjusted. Row spacing was also adjusted because it satisfied the conditions described above. Through trial and error a combination of row spacing and inner void radius were determined, from which CONDIP yielded a geometric design similar to the DDG-37 condenser and that satisfied condenser requirements specified in

[Ref. 17]. In this particular case, the void diameter and row spacing were determined to be 1.1 feet and 1.35 inches respectively. This arrangement enabled the condenser model to closely approximate the tube sheet area ratio of the actual DDG-37 condenser. This manipulation, however, must be interpreted as another source of error and innacuracy when comparing CONDIP's condenser performance with the experimental results.

Lastly, condenser designs in [Ref. 15] reveal that three different tube patterns were employed in the DDG-37 condenser. In addition, two different values for tube pitch were used - a pitch of 1.4 in the condenser bundle and a pitch of 1.3 in the cooler. This situation cannot be duplicated in CONDIP. Therefore a constant pitch of 1.40 and a uniform tube pattern were utilized throughout the condenser.

The design approximations utilized in CONDIP to try to geometrically simulate the actual DDG-37 condenser introduce significant uncertainty into subsequent design comparisons. This, coupled with the fact that the data collected is also suspect, would imply that it is rather difficult to verify CONDIP's analysis with the information available. It should also be noted that CONDIP is sensitive to even small variance in either the data collected (i.e. steam inlet temperature) or the approximated design variables. However, despite the above-mentioned problems associated with equating CONDIP's condenser to the DDG-37 condenser, the experimental data obtained from the DDG-37 condenser still provide the best available base upon which to make a reasonable determination of CONDIP's capabilities and limitations.

In comparing the results in Table II, it is immediately clear that there is significant difference between certain condenser performance parameters predicted by CONDIP and the corresponding experimentally derived condenser perfcrmance. Already, much has been said about the numerous geometric

approximations used to model the DDG-37 condenser. But questionable data and geometric manipulations may not completely explain the 13 percent exit-fraction and general poor performance generated in CONDIP's analysis. Its values for the average overall heat transfer coefficient and heat rejected were significantly lower than the experimental results. One source of the problem lies in the actual heat transfer analysis performed in the code. Lynch [Ref. 18] graphically illustrated how sensitive this analysis is to the effects of condensate inundation. In particular, by making small changes - within the allowable ranges - in the constants used in Eissenberg's correlations for condensate rain, significant improvement could be realized in the overall heat transfer characteristics of the condenser. CONDIP's results, when compared to the experimental data support the argument that the values currently used in the inundation correlations are rather conservative in nature, and cause the overall analysis to yield a poor preformance for the given steam load and condenser design.

Therefore, in order to present CONDIP with a fair test to determine its credibility as a design predictor, some additional work must be first accomplished. A condenser geometrically identical to the general model created in CONDIP should be constructed with complete and accurate data acquisition systems to establish a thorough data base from which to compare. Also, more research should be performed on the effects of condensate inundation and velocity shear to obtain more precise correlations in determining their overall effects on the film heat transfer coefficients.

One last additional point should be mentioned. In comparing the steam-side pressure drops through the condenser, it was shown that CONDIP's pressure losses were nearly 72 percent larger the the actual physical measurements. However this radical difference is mainly due to the

high pressure losses experienced at the entrance of the cooler as a large volume of steam tried to force its way through the small available area. Therefore, the significance of this large disagreement in results is relatively minor and can be treated simply as a consequence of the more important heat transfer limitations in the comparison run.

Although the goal of verifying CONDIP as a design predictor has proven elusive, it was still possible to demonstrate its capabilities through comparison studies. Therefore, the remaining emphasis in this thesis is to show the ability of CONDIP (in combination with the optimizer COPES/CONMIN) to take an initial design with a given framework of constraints and design variables, and obtain better designs based on a desired objective function.

B. EXPLANATION OF THE CASE STUDIES

The following case studies were devised to best exercise the capabilities of CONDIP. They were made as realistic as possible so as to simulate the problem of condenser design and specification confronting the engineer during the early stages of power plant design. The condenser performance returned by CONDIP during the verification run and contained in Table II will serve as a baseline for comparing the results of each case study. The baseline condenser performance is based on the design parameters from the DDG-37 condenser listed in Table I. It was stated earlier that CONDIP's optimization results are slightly sensitive to the initial design if more than three design variables are used. Since all the cases involve eight or more design variables it would be best, for the purposes of comparison, to start from the same initial design in all cases. Therefore, the initial design variables used for the verification run and contained in Table I will be utilized as the baseline

design. Although many of these initial design variables will be allowed to change during optimization, certain basic condenser requirements will not. They include: steam flow into the condenser, inlet steam saturation pressure and temperature, cooling water injection temperature, the fraction of non-condensible gases in the steam, the tube fouling factor, and the tube material. It should be noted that although there was an initial value for row spacing given in Table I, row spacing was not used as a design variable during any of the optimizations. Instead, the program used the default method of row spacing calculation available in the code where the rows are spaced such that a 60-degree equilateral triangle pattern of concentric rows is obtained. Row spacing is therefore dependent on tube pitch and tube outer diameter by the following relation:

$$RSPA = (SDDO * ODOI) * .866 \qquad \text{(eqn 5.1)}$$

where RSPA is row spacing, SDDO is tube pitch, and ODOI is tube outer diameter in inches.

There are a few key points to be kept in mind when comparing the results of the case studies with the baseline. First, the baseline design is an infeasible and inadequate design. Its performance indicates that it is not capable of supporting the required steam load by returning a steam exit-fraction in excess of 13 percent. So any gains in the objective function that were realized in the case studies is even more remarkable since it is a necessary condition that the optimum design be a feasible design, defined as having an exit-fraction not greater than 1 percent. Second, the percent change referred to when analyzing the results is calculated based on the baseline design. Thus the baseline serves as a uniform frame of reference. Next, it should be noted that because of the large number of design variables

and constraints, intuition on how an optimized result will turn out is not always applicable. Finally, it will be easier to understand the effects of the various design parameters by keeping in mind the following, very basic, heat transfer correlation:

$$Q = U * A * LMTD \qquad \text{(eqn 5.2)}$$

where Q is the rate of heat released as the steam condenses; U is the overall heat transfer coefficient; A is the heat transfer surface area; and LMTD can be interpreted as the thermal driving force between the steam and the coolant. Q is directly dependent on steam flow and pressure into the condenser, the percentage of that steam that is condensed, and any subcooling of the condensate. For a given steam load and a very small exit-fraction Q is nearly constant as the optimized results in all the case studies will indicate.

1. Constraint Framework for CONDIP

In order to simulate an actual trade-off study, the constraints and their respective limits were kept constant for all the case studies. The condenser was to be designed with a maximum bundle diameter of ten feet, a maximum and minimum tube outer diameter of 0.625 and 1.25 inches respectively, a steam exit-fraction of not more than 1 percent, a maximum cooler inlet velocity (VLCMAX) of 200 feet/second, and a ratio of tube sheet hole area to total tube sheet area of less than 0.30.

The constraint on bundle diameter was chosen somewhat arbitrarily. It seems unlikely that this limit would be realistically exceeded, although certainly space requirements would dictate the exact configuration. Tube outer diameter is dependent on the values for tube wall thickness and tube inner diameter. Thus the limits imposed on tube

outer diameter represent realistic restrictions on the possible cobinations of inner diameter and wall thickness. These restrictions are based loosely on anticipated tube structural and strength requirements and correspond to values of normally available tubes [Ref. 19]. The maximum limit of 200 feet/second for VLCMAX was also a somewhat arbitrary but realistic limit. It is assumed that steam velocities often exceed that value in the condenser bundle.

It is recalled that steam exit-fraction will play a significant role in the determination of the final optimum design. The baseline exit-fraction of 13 percent predicted by CONDIP for the DDG-37 condenser is unsatisfactory. Therefore a more reasonable upper limit of 1 percent was placed on this constraint. Although CONDIP will return a much more conservative design if 1 percent vice 13 percent is used as the upper limit, the subsequent design will be much more credible.

Finally, the amount of tube sheet material that can be removed by drilling for the installation of condenser tubes is specified at 24 percent of the total tube sheet area in [Ref. 17]. This area ratio limit represents a structural limit imposed to ensure that the tube sheets do not fail due to heat and pressure stresses in the condenser. However, CONDIP does not take into account the space between the condenser and tube shell normally used in area ratio calculations as blank tube sheet area. For this reason and to allow more flexibility in the design analysis, the constraint limit was set at 30 percent.

In summary, the general design constraints and the associated upper and lower bounds were:

$$0.625 \le \text{tube outer diameter (inch)} \le 1.25$$

$$1.0 \le \text{bundle diameter (feet)} \le 10.0$$

$$\text{steam exit-fraction (\%)} \le 1.0$$

$$\text{VLCMAX (ft/sec)} \le 200.0$$

$$\text{area ratio} \le .30$$

These design constraints and associated bounds were used in all the case studies except where specifically modified.

2. Design Variable Framework for CONDIP

At least eight design variables were used in all the case studies. They include tube inner diameter, tube wall thickness, tube pitch, the number of tubes in the condenser, tube length, the inner void radius, the percent of the tubes in the cooler, and cooling water velocity. Side constraints were placed on all of these variables to correspond to either realistic physical limits or available standardized materials.

Tube wall thickness was not allowed to fall below 0.022 inches (BWG 24) or exceed .109 inches (12 BWG), sizes normally available commercially. Tube inner diameter was restricted to values between .407 and 1.206 inches so as to yield tube outer diameters within the limits specified earlier.

Tube pitch is defined as the ratio of the center to center spacing between adjacent tubes in a row to the tube outer diameter. Tube pitch is an accurate measure of how closely packed the tube bundle is. Generally accepted values for pitch lie in the range of 1.3 to 1.7. However, to provide more latitude in the design process this design variable was allowed to vary in the range between 1.1 to 2.0.

There is no guidance available as to the allowable range for tube length in the condenser. Since the lower limit was not expected to be crucial, it was set randomly at 1.0 feet. The upper limit of 25.0 feet was a realistic limit considering the size of the tube diameter being worked with. Inner void radius and the percent of tubes in the cooler were chosen to be design variables simply to enhance the

flexibility of the code in designing the condenser model. The bounds for both variables were entirely arbitrary with only common sense as the determining factor. The upper and lower limits on the percent of tubes in the cooler was established as 10.0 and 2.0 percent respectively. The upper and lower bounds on the inner void radius was set at 1.0 and 0.1 feet.

Cooling water velocity generally ranges from three to nine feet per second in value for all common tube materials, except titanium which has an upper limit of 15 feet per second. Exceeding these upper limits risks excessive tube erosion and material damage. Finally the number of tubes was permitted to vary between 1000 and 8000 tubes for the purpose of improving design flexibilty. It is extremely unlikely that, for most propulsion applications, tube number would fall below 1000. The upper limit was simply chosen as a realistic cutoff point in terms of complexity, cost and maintainability.

In summary, the general design variables and the associated side-constraints were:

$$0.407 \le \text{tube inner diameter (inches)} \le 1.206$$
$$0.022 \le \text{tube thickness (inches)} \le 0.109$$
$$2.0 \le \text{percent of tubes in cooler} \le 10.0$$
$$0.10 \le \text{inner void radius (feet)} \le 1.0$$
$$3.0 \le \text{coolant velocity (ft/sec)} \le 9.0$$
$$1.0 \le \text{tube length (feet)} \le 25.0$$
$$1000 \le \text{tube number} \le 8000$$
$$1.1 \le \text{tube pitch} \le 2.0$$

As in the case of design constraints, these design variables and their respective limits were used consistently in all the case studies unless otherwise specified.

C. CASE STUDIES USING CONDIP

1. Case One

The objective of this case was to minimize condenser volume. The final results of the optimization along with the initial parameters is listed in Table III.

These results show a 16 percent decrease in condenser volume with a corresponding 24 percent increase in pumping power. The source of the improvement can be understood by noting the following:

1) Tube wall thickness was reduced from 0.049 to 0.022, the minimum side-constraint, thus allowing tube inner diameter to increase while maintaining a minimum tube outer diameter.

2) The number of tubes shrank slightly as did tube length, resulting in a smaller heat transfer surface area.

3) Tube pitch increased markedly, causing a reduction in steam pressure losses which then ensured that high values for steam saturation pressure and temperature would be maintained throughout the condenser. The large pitch also reduced steam velocities, allowing the cooler inlet velocity limit to be satisfied. Row spacing decreased from the initial value of 1.35 inches, thus decreasing condenser volume.

4) Cooling water velocity increased to the maximum allowable value of 9 ft/sec which correspondingly resulted in larger head losses and coolant flow, causing overall pumping power to increase.

As cooling water velocity increased and tube wall thickness decreased, then their respective thermal resistances were diminished. The cumulative effect was to improve the overall heat transfer coefficient. LMTD rose primarily as a result of the higher steam temperatures throughout the condenser. It is apparent by looking at equation 5.2 that increasing the driving forces for heat

transfer, such as the overall heat transfer coefficient and LMTD, allows the heat transfer surface area to decrease. This resulted in a similar reduction in condenser volume.

The constraint limits that prevented further design improvement were the upper bound on the cooling water velocity, the upper limit on the tube sheet area and the upper limit on VLCMAX.

2. Case Two

The objective of this case was to minimize the pumping power required to overcome the tube-side head losses and drive the cooling water through the condenser tubes. The final results of the optimization are presented along with the initial design in Table IV.

The results indicate a dramatic 90 percent reduction in required pumping power with an equally large 120 percent increase in condenser volume. The major factors involved in the design improvement along with their relative effects are briefly explained below:

1) Tube inner diameter increased 27 percent while tube thickness remained relatively unchanged. Thus tube outer diameter was caused to increase.

2) The number of tubes in the condenser rose significantly, along with tube length. This, coupled with the enlarged tube outer diameter resulted in nearly doubling the heat transfer area.

3) Tube pitch icreased 29 percent, which allowed steam saturation pressure and temperature to be maintained at consistently large values in the condenser. This had a benefiting effect on the associated LMTD calculation. The large pitch also helped satisfy the steam velocity limit into the cooler. The tube spacing decreased from the initial value of 1.35, but by a smaller amount than the previous case because of the large values for tube pitch and outer diameter.

4) Cooling water velocity dropped to the minimum allowable limit of 3 ft/sec. This had the effect of reducing tube-side head losses and coolant flow through the condenser. Consequently, pumping power was drastically reduced.

The combined effect of all these changes can again be put in perspective by looking at equation 5.2. For the given steam and corresponding heat load, the heat transfer area increased drastically, allowing both LMTD and the overall heat transfer coefficient to decrease. A smaller overall heat transfer implies a smaller convective tube-side contribution which in turn permits coolant velocity to reduce to its lowest allowable value. The LMTD decrease is explained by the fact that cooling water was spending more time in the tubes, thus causing the average cooling water temperature to rise. However, the subsequent reduction in LMTD was minimized by the fact that a high steam temperature was maintained in the condenser.

There were no active constraints in this design outside of cooling water velocity which prevented further design improvement. However, the penalty paid in terms of a huge condenser volume, appears prohibitive.

3. Case Three

The objective of this case was to minimize pumping power while holding condenser volume constant at the initial value of 432 cubic feet. This was a particularly interesting test case as the results in Table V bear out. The required pumping power was reduced by nearly 38 percent with no change in volume. The effects of the design changes which resulted in the design improvement are presented below:

1) Tube inner diameter increased noticeably. However, the effects of this increase on tube outer diameter was minimized by a large drop in tube wall thickness. Thus, tube outer diameter remained relatively unchanged.

2) The number of tubes experienced a minor reduction, while tube length increased. The overall effect was to increase heat transfer surface area.

3) Tube pitch again rose by nearly 25 percent, causing steam saturation temperature and pressure to maintain a nearly constant value throughout the condenser. This had a beneficial effect on the LMTD between the steam and the cooling water. The larger pitch also had the additional effect of reducing steam velocity thus allowing the subsequent design to satisfy the upper limit on steam velocity into the cooler (VLCMAX). The combination of tube pitch and tube outer diameter resulted in a reduction in row spacing from the initial value of 1.35

4) Cooling water velocity decreased by about 21 percent. This effect was manifested in subsequent pressure head, coolant flow and pumping power calculations.

Looking at equation 5.2 we see the same general pattern emerging as in Case 3, but with more subtlety in the changes. Heat transfer increased, but not at the expense of volume. Cooling water velocity was allowed to decrease while the overall heat transfer coefficient actually rose. One explanation is that as the tube wall got thinner its thermal resistance got smaller which more than offset the loss of convective heat transfer contribution from the coolant. The LMTD dropped slightly due to the higher average coolant temperature of the coolant in the tubes.

The constraints which became active and prevented further improvement in the design include tube sheet area ratio as well as tube wall thickness. However, tube wall thickness was particular crucial because of its related effect on heat transfer.

4. Case Four

The objective of this case was to minimize condenser volume while holding pumping power constant at the initial value of 55.7 horsepower. The results of this case can be found in Table VI. Chosen to contrast the results in Case 4, the relative improvement in this design objective was not nearly so impressive. Condenser volume shrank by only 12 percent. An explanation of the causes and effects is provided below:

1) Tube inner diameter increased, with a corresponding decrease in tube wall thickness to yield the minimum allowable tube outer diameter.

2) The number of tubes decreased noticeably, tending minimize bundle volume. Note, there was only slight increase in tube length. The overall effect was to similarly reduce heat transfer surface area as condenser volume decreased.

3) Tube pitch again increased significantly, having the same effects on steam pressure, temperature and steam velocity into the cooler as discussed earlier. A large tube pitch benefits the LMTD between the steam and the cooling water. Row spacing was again a factor in reducing condenser volume as before.

4) Cooling velocity decreased slightly as did head loss. But overall coolant flow increased due to an increase in tube inner diameter. The net effect was to maintain pumping power.

Again, referring to equation 5.2 , it is clear that the slight decrease in heat transfer area was offset by the slight rise in LMTD resulting from higher condenser steam temperatures. The significant improvement in overall heat transfer coefficient, therefore, is what makes the heat balance work. The large decrease in tube wall thickness and corresponding reduction in thermal resistance contributed heavily to this improvement.

There were several contraint limits which prevented any additional objective optimization. They include the minimum tube wall thickness, tube sheet area ratio and steam velocity entering the cooler.

5. Case Five

The objective of this case was to minimize condenser volume while exercising CONDIP's capabilities to linearly vary tube pitch and tube inner diameter by row. Thirty-five rows were used, which was the identical number as the initial design. Tube pitch and tube inner diameter of both the outermost and innermost rows served as design variables in this case. However - because tube number is now a dependent variable based on the number of rows, tube pitch, and tube diameter - it could not be used as a design variable. The optimized results of this analysis along with the initial design are presented in Table VII.

The results of this test case indicate a condenser volume which is 20 percent smaller than the initial design as compared to a 16 percent decrease in Case 1. The basic reasons and explanations as to why volume was able to be reduced remain fundamentally the same as in Case 1. Attention will therefore be focussed on the effects of linearly varying pitch and tube diameter. Final tube pitch ranged in value from 1.75 in the inner row to 1.44 in the outer row. Similarly, tube inner diameter ranged from .729 in the inner row to .583 in the outer row.

It is believed that smaller pitch and inner diameter were used in the outer row because of the higher available steam saturation pressure and temperature. The resulting higher steam velocities enhanced the beneficial effects of vapor shear on the external heat transfer coefficient thereby improving heat transfer on the outer rows. As steam pressure decreased, then tube pitch and tube inner diameter

increased to compensate and extract all the available heat from the steam. Consequently, steam velocity decreased and was able to satisfy to the limit imposed on the cooler entrance velocity. The end result is a condenser geometry that makes complete use of the available resources and conforms to the geometry to take advantage of the thermal conditions in the condenser. The big limitation with this approach is that the number of rows is held constant. Thus the subsequent condenser is designed around that value and the subsequent optimum design is a function of the number of rows specified.

TABLE I

Input Design Data

PARAMETER	VALUE
Total number of tubes	5230
Tube length (feet)	10.3
Tube inner diameter (inches)	0.527
Tube wall thickness (inches)	0.049
Tube outer diameter (inches)	0.625
Tube mat'l thermal conductivity btu/(ft-hr-°F)	26.0
Tube pitch	1.38
Percent of tubes in the cooler	7.0
Steam inlet flow (lbm/hr)	161,961
Fraction of non-condensible gas (ppm)	37.1
Steam inlet pressure (psia)	1.294
Steam inlet temperature (°F)	110.52
Coolant inlet velocity (ft/sec)	8.473
Coolant inlet temperature (°F)	75.66
Fouling factor	.0002
Inner void diameter (feet)	1.1
Row spacing (inches)	1.35

TABLE II

CONDIP Verification Results

PARAMETER	EXPERIMENT RESULTS	CONDIP RESULTS	CHANGE (%)
Heat transfer area (sq.ft.)	8805	8814	+0.10
Overall heat transfer coefficient btu/(hr-sq.ft.-°F)	635.2	547.9	-9.5
Log mean temperature difference (°F)	28.24	28.62	+1.3
Coolant temperature rise (°F)	10.61	9.90	-6.7
Coolant mass flow rate (10^7 lbm/hr)	1.503	1.540	+2.5
Condenser volume (cu.ft.)	-----	432.3	-----
Bundle diameter (ft)	5.7 7.2	7.17	-----
Shell-side pressure drop (psia)	0.751	1.29	+71.8
Steam exit-fraction (% of input)	-----	13.3	-----
Heat rejected (10^8 btu/hr)	1.595	1.451	-9.03
Area ratio	0.291	0.266	-8.59

TABLE III

Volume Minimization

PARAMETER	BASELINE RESULTS	OPTIMIZED RESULTS	CHANGE (%)
Total number of tubes	5230	5117	-2.2
% of tubes in cooler	7.0	7.01	+0.1
Tube length (ft)	10.3	9.92	-3.7
Tube inner diam. (in)	.527	.582	+10.4
Tube wall thick. (in)	.049	.022	-55.1
Tube outer diam. (in)	.625	.626	+0.2
Tube pitch	1.4	1.73	+23.6
Void diameter (ft)	1.10	1.34	+21.8
Bundle diameter (ft)	7.17	6.59	-8.1
Condenser volume (cu.ft.)	432.3	362.2	-16.2
Area ratio	0.266	0.300	+12.7
Coolant inlet vel. (ft./sec)	8.473	9.00	+6.2
Coolant mass flow rate (10^7 lbm/hr)	1.540	1.955	+26.9
Head loss (ft H2O)	7.35	7.18	-2.3
Pumping power (hp)	55.69	68.99	+23.9
Coolant temperature rise (°F)	9.90	9.00	-9.1
Log mean temperature difference (°F)	28.62	29.13	+1.8
Heat transfer area (sq.ft.)	8814.	8320.	-5.6
Average overall heat transfer coefficient btu/(hr-sq.ft.-°F)	574.9	689.9	+20.0
Steam exit-fraction (% of input)	13.3	0.0	-100.
Heat rejected (10^8 btu/hr)	1.451	1.672	+15.2

TABLE IV
Power Minimization

PARAMETER	BASELINE RESULTS	OPTIMIZED RESULTS	CHANGE (%)
Total number of tubes	5230	6393	+22.2
% of tubes in cooler	7.0	7.4	+5.7
Tube length (ft)	10.3	13.34	+29.5
Tube inner diam. (in)	.527	.667	+26.6
Tube wall thick. (in)	.049	.0455	-7.1
Tube outer diam. (in)	.625	.758	+21.3
Tube pitch	1.4	1.812	+29.4
Void diameter (ft)	1.10	1.13	+2.7
Bundle diameter (ft)	7.17	9.25	+29.0
Condenser volume (cu.ft.)	432.3	964.4	+123.1
Area ratio	0.266	0.277	+4.1
Coolant inlet vel. (ft./sec)	8.473	3.00	-64.6
Coolant mass flow rate (10^7 lbm/hr)	1.540	1.067	-30.7
Head loss (ft H2O)	7.35	1.11	-84.9
Pumping power (hp)	55.69	5.84	-89.5
Coolant temperature rise (°F)	9.90	16.3	+64.6
Log mean temperature difference (°F)	28.62	24.82	-13.3
Heat transfer area (sq.ft.)	8814.	16,921	+92.0
Average overall heat transfer coefficient btu/(hr-sq.ft.-°F)	574.9	393.8	-31.5
Steam exit-fraction (% of input)	13.3	1.0	-92.5
Heat rejected (10^8 btu/hr)	1.451	1.654	+14.0

TABLE V

Power Minimization With Volume Constant

PARAMETER	BASELINE RESULTS	OPTIMIZED RESULTS	CHANGE (%)
Total number of tubes	5230	5062	+3.2
% of tubes in cooler	7.0	6.7	-4.3
Tube length (ft)	10.3	11.39	+10.6
Tube inner diam. (in)	.527	.594	+12.7
Tube wall thick. (in)	.049	.022	-55.1
Tube outer diam. (in)	.625	.638	+2.1
Tube pitch	1.4	1.753	+25.2
Void diameter (ft)	1.10	1.14	+3.6
Bundle diameter (ft)	7.17	6.73	-6.1
Condenser volume (cu.ft.)	432.3	431.6	-0.2
Area ratio	0.266	0.297	+11.7
Coolant inlet vel. (ft./sec)	8.473	6.74	-20.5
Coolant mass flow rate (10^7 lbm/hr)	1.540	1.508	-2.1
Head loss (ft H2O)	7.35	4.69	-36.2
Pumping power (hp)	55.69	34.77	-37.6
Coolant temperature rise (°F)	9.90	11.6	+17.2
Log mean temperature difference (°F)	28.62	27.63	-3.5
Heat transfer area (sq.ft.)	8814.	9631.4	+9.3
Average overall heat transfer coefficient btu/(hr-sq.ft.-°F)	574.9	627.3	+9.1
Steam exit-fraction (% of input)	13.3	0.0	-100.
Heat rejected (10^8 btu/hr)	1.451	1.669	+15.0

TABLE VI

Volume Minimization With Power Constant

PARAMETER	BASELINE RESULTS	OPTIMIZED RESULTS	CHANGE (%)
Total number of tubes	5230	4867	-6.9
% of tubes in cooler	7.0	7.2	+2.9
Tube length (ft)	10.3	10.92	+6.0
Tube inner diam. (in)	.527	.581	+10.2
Tube wall thick. (in)	.049	.022	-55.1
Tube outer diam. (in)	.625	.625	0.0
Tube pitch	1.4	1.739	+24.2
Void diameter (ft)	1.10	1.20	+9.1
Bundle diameter (ft)	7.17	6.41	-10.6
Condenser volume (cu.ft.)	432.3	378.4	-12.4
Area ratio	0.266	0.299	+12.4
Coolant inlet vel. (ft./sec)	8.473	8.27	-2.4
Coolant mass flow rate (10^7 lbm/hr)	1.540	1.701	+10.5
Head loss (ft H2O)	7.35	6.67	-9.3
Pumping power (hp)	55.69	55.80	+0.2
Coolant temperature rise (°F)	9.90	10.34	+4.4
Log mean temperature difference (°F)	28.62	28.38	-0.8
Heat transfer area (sq.ft.)	8814.	8697.	-1.3
Average overall heat transfer coefficient btu/(hr-sq.ft.-°F)	574.9	677.4	+17.5
Steam exit-fraction (% of input)	13.3	0.0	-100.
Heat rejected (10^8 btu/hr)	1.451	1.672	+15.2

TABLE VII

Volume Minimization With Linear Variations

PARAMETER	BASELINE RESULTS	OPTIMIZED RESULTS	CHANGE (%)
Total number of tubes	5230	5348	+2.3
% of tubes in cooler	7.0	7.3	+4.3
Tube length (ft)	10.3	8.7	-15.5
Tube inner diam. (in)	.527	*.729 .583	----
Tube wall thick. (in)	.049	.022	-55.1
Tube outer diam. (in)	.625	*.773 .627	----
Tube pitch	1.4	*1.75 1.44	----
Void diameter (ft)	1.10	1.28	+16.4
Bundle diameter (ft)	7.17	6.8	-5.2
Condenser volume (cu.ft.)	432.3	345.8	-20.0
Area ratio	0.266	0.298	+12.0
Coolant inlet vel. (ft./sec)	8.473	9.0	+6.2
Coolant mass flow rate (10^7 lbm/hr)	1.540	2.48	+61.0
Head loss (ft H2O)	7.35	5.84	-20.5
Pumping power (hp)	55.69	71.1	+27.7
Coolant temperature rise (°F)	9.90	7.1	-28.8
Log mean temperature difference (°F)	28.62	30.14	+5.3
Heat transfer area (sq.ft.)	8814.	7746.	-12.1
Average overall heat transfer coefficient btu/(hr-sq.ft.-°F)	574.9	713.1	+24.0
Steam exit-fraction (% of input)	13.3	0.5	-96.2
Heat rejected (10^8 btu/hr)	1.451	1.665	+14.7

* Inner row values followed by outer row values.

VI. CONCLUSIONS

The intent of this research was to create a detailed condenser analysis code capable of being coupled with a numerical optimizer and to test the program to prove its versatility. An additional objective was to validate the analysis with existing data. The results of the test cases were presented in Chapter Five; the resulting conlusions are summarized here.

A. There were significant difficulties encountered in formulating the complex condenser design anaysis, CONDIP, in a way that was compatible with the optimizer COPES/CONMIN. However, the majority of those problems were overcome resulting in the creation of a program which, when combined with the optimizer, is capable of taking any initial design, no matter how impractical or infeasible, and solving for an optimum solution based on a set of pre-determined constraints and design variables. There are still some minor limitations as to the degree of optimization, but the final design is usually within 10 pecent of the single best optimum. In addition, the test cases indicate that as many as ten design variables and six constraints can be used simultaneously in the design optimization with CONDIP.

B. The test cases demonstrated the effectiveness of CONDIP as a design tool for not only the conceptual design of a condenser, but also in evaluating comparison studies based on any number of design variable combinations. The number of possible combinations of design objectives, design variables and design constraints implies limitless possibilities to be explored and evaluated.

C. An attempt was made to verify CONDIP with existing data with inconclusive results. Part of the blame can be placed on the rather inadequate quality and quantity of the data, but the general performance of CONDIP's condenser indicates a weakness in the analysis. As stated earlier, the source of the this weakness may be found in the correlations used for condensate inundation. The constants used in the expression for correcting shell-side heat transfer coefficients are based somewhat on conjecture. Yet they play a significant role in the overall condenser performance. Despite this limitation, the ability to optimize CONDIP's detailed analysis is a significant step forward over using the traditional and limited HEI method.

D. CONDIP incorporates features that further increase its appeal as a design tool. By possessing the ability to linearly vary pitch and tube diameter, a better understanding of how to improve condenser performance based on its configuration is realized. The capability to incorporate shell-side tube enhancement is another added plus. The possibilities that can now be investigated are limitless.

VII. RECOMMENDATIONS

In addition to the insight that this investigation has given into the generation of automated condenser design programs, it has specifically addressed the shortcomings and pitfalls which may be encountered along the way and offered possible solutions to overcome them. Presented below are recommendations for furthering the developmentof CONDIP as a completely versatile and accepted design program.

A. Since the weak link and the most signifcant unknown in condenser analysis is the effect of condensate rain in typical condenser environments, subsequent research should be devoted to investigating this phenomenon and developing more precise analytic correlations. In particular, the effects of velocity and flow direction on the condensate film should be attended to.

B. Perhaps in conjunction with the above, a test condenser should be constructed which is geometrically similar to the model proposed in CONDIP in order to physically observe and record the condenser performance. This data could then be used to either verify CONDIP or strengthen some of its anaysis. In addition, this condenser should be built such that the tube bank can be arranged in any combination of pitch, tube diameter and row spacing to fully appreciate the effects of these variables.

C. A series of sensitivity studies should be conducted on CONDIP to fully exercise its capabilities and determine the relative effects of various designvariables on condenser performance. Tradeoff studies similar to those performed in this research would be most beneficial to fully understand condenser behavior.

D. Additional subroutines should be created which would allow tube enhancement to be a design variable. This involves developing correlations between heat transfer enhancement and associated frictional losses. This type of relationship can be developed for both tube-side and shell-side enhancement.

E. Finally, it is recommended that additional refinement be performed on the code to increase its capability and flexibility. One such way is to somehow allow pitch, tube diameter and tube wall thickness to vary linearly by row while still allowing the number of tubes to be a design variable. The options available are limitless.

APPENDIX A
GLOSSARY

While it would most beneficial to present a complete glossary of all the variables used in CONDIP, the sheer number makes it difficult to present a comprehensive list. However, CONDIP makes liberal use of comment cards to define as many variables as possible to make the code easier to follow. Therefore, the computer listing in Appendix C is available for reference. A list of the possible design variables and constraints is provided here along with its corresponding position in the **GLOBCM common block** for easy reference in writing the appropriate COPES data cards. In addition, it will be specified whether these variables can be used as design constraints or design variables.

1. ALST: The length of the condenser and cooler tubes in feet. ALST is to be used only as a design variable.

2. DELWP: The pressure difference between the inlet and outlet coolant headers of the condenser bundle in psi. DELWP is to be used only as a design variable.

3. DELWPC: The pressure difference between the inlet and outlet coolant headers of the condenser bundle in psi. DELWPC is to be used only as a design constraint.

4. GFLOW: The mass flow rate of the coolant in lbm/hour. GFLOW cannot be used as a design variable simultanouesly with VELBI. Otherwise it can be used as a design variable or a design constraint.

5. SIDI: The tube inner diameter of the innermost row of

the condenser bundle in inches. SIDI is to be used only as a design variable.

7. __SIDO__: The tube inner diameter of the outermost row of the condenser bundle in inches. If there is to linear variation of tube inner diameter then this variable represents the tube inner diameter of the entire condenser bundle. SIDO is to be used only as a design variable.

8. __PHP__: The coolant pumping power in horsepower. PHP is to be used only as a design constraint.

9. __RSPA__: The spacing between concentric rows in the condenser bundle in inches. RSPA is to be used only as a design variable.

10. __RADINS__: The inner void radius of the condenser bundle in feet. RADINS is to be used only as a design variable.

11. __REWI__: The tube-side Reynolds number of the coolant in the innermost row of the condenser bundle. REWI is to be used only as a design constraint.

12. __REWO__: The tube-side Reynolds number of the coolant in the outermost row of the condenser bundle. If there is no linear variation of tube inner diameter then this variable represents the tube-side Reynolds number of the entire condenser bundle. REWO is to be used only as a constraint.

13. __SDDI__: Tube pitch (tube spacing/tube outer diameter) of the innermost row of the condenser bundle. SDDI is to be used only as a a design variable.

14. __SDDO__: The tube pitch of the outermost row of the condenser bundle. If there is no linear variation of tube pitch then this variable represents the tube pitch for the entire condenser bundle. SDDO is to be used only as a design variable.

15. SLDI: Ratio of tube length to tube outer diameter of the outermost row of the condenser bundle. SLDI is to be used only as a design constraint.

16. SLDO: Ratio of tube length to tube outer diameter of the outermost row of the condenser bundle. If there is no linear variation of tube pitch then this variable represents the tube pitch for the entire condenser bundle. SLDO is to be used only as a design constraint.

17. VELBI: The velocity of the coolant in feet/sec. VELBI cannot be used as a design variable simultaneously with GFLOW. Otherwise it can be used as either a design constraint or as a design variable.

18. XW1: The ratio of tube thickness to tube inner diameter. XW1 can be used only as a design variable.

19. XW2: Tube thickness in inches. XW2 is to be used only as a design variable. XW2 and XW1 cannot be used simultaneously.

20. VOL1: The overall condenser and cooler volume in cubic feet. VOL1 is to be used only as a design constraint.

21. VOL2: The volume occupied by the tube bank, excluding the volume of the inner void, in cubic feet. VOL2 is to be used only as a design constraint.

22. TNOTOT: The total number of tubes in the condenser and cooler combined. If Option 1 is being used then TNOTOT is to be used only as a design constraint. If Option 2 is being used then TNOTOT is to be used only as a design variable.

23. BNDRAD: The condenser bundle in feet. BNDRAD is to be used only as a design constraint.

24. ARATIO: The ratio of the total cross-sectional area of

the tubes (based on the tube outer diameter) to the tube sheet area. ARATIO is to be used only as a design constraint.

25. ODII: The tube outer diameter of the innermost row in inches. ODII is to be used only as a design constraint.

26. ODOI: The tube outer diameter of the outermost row in inches. If there is no linear variation of tube inner diameter then this variable represents the tube outer diameter of the entire condenser bundle. ODOI is to be used only as a design constraint.

27. VLCMAX: The maximum allowable steam velocity into the cooler. VLCMAX can be used only as a design constraint and only when a cooler is being designed in the system.

28. PRCCLR: The percent of the total number of tubes in the cooler. PRCCLR can be used only as a design variable.

APPENDIX B

USERS MANUAL FOR CONDIP

This appendix describes the data cards that are necessary in order to couple any design program with COPES/CONMIN. Also described are cards illustrating data input required by CONDIP to initiate analysis. Thus, the data is divided into the COPES/CONMIN program section and the CONDIP-based condenser design program section.

The COPES data is segmented into "blocks" for convenience. All formats are alphanumeric for title, end and stop cards; F10 for real data; and I10 for integer data. The formatted input may be overridden by inserting commas between data entries. Comment cards may be inserted anywhere in the data stack prior to the end card and are identified by a dollar sign ($) in column 1. The COPES data stack must terminate with an end card containing the word "END" in column 1-3. It should be noted that information pertaining only to single analysis and optimization is presented here. Information concerning the other options available in COPES along with further explanation of COPES capabilities can be found in [Ref. 13].

The analysis data is also segmented into blocks for convenience and they begin immediately following the "END" card in the COPES data. No comment cards are permitted here, and the analysis data stack must terminate with the word "STOP" in columns 1-4. This is where the initial design values are placed for entry into CONDIP.

Default values are recommended for use in the following COPES data cards unless otherwise noted. It is recommended that these values in the COPES data blocks be used until the user becomes familiar with the program. In addition a

sample data input is illustrated in figure B.1 at the end of this appendix.

DATA BLOCK A

DESCRIPTION: COPES Title Card

FORMAT: 20A4

1	2	3	4	5	6	7	8
TITLE							

REMARKS:

1) This line is available for a brief description.

DATA BLOCK B

DESCRIPTION: COPES Program Control Parameters

FORMAT: 7I10

1	2	3	4	5	6	7	8
NCALC	NDV						

FIELD		CONTENTS
1	NCALC:	Calculation control
	0	Read input and stop. Data of blocks A-B is required. Remaining data is optional.
	1	One cycle through the program. Data of blocks A-B is required. Remaining data is optional.
	2	Optimization. Data of blocks A-I is required. Remaining data is optional.
2	NDV:	Number of independent design variables in optimization or optimum sensitivity study.

REMARKS:

1) Field 1 determines program execution

2) Fields 3-8 are to be left blank for the CONDIP application of COPES/CONMIN.

DATA BLOCK C

DESCRIPTION: COPES Integer Optimization Control Parameters

FORMAT: 8I10

1	2	3	4	5	6	7	8
IPRINT	ITMAX	ICNDIR	NSCAL	ITRM	LINOBJ	NACMX1	NFDG

FIELD		CONTENTS
1	IPRINT:	Print control used in optimization program, CONMIN.
	0	No print during optimization.
	1	Print initial and final optimization information.
	2	Print above plus function value and design variable values at each iteration.
	3	Print above plus constraint values, direction vector and move parameter at each iteration.
	4	Print above plus gradient information.
	5	Print above plus each proposed design vector, objective function and constraints during the one-dimensional search. required. Remaining data is optional.
2	ITMAX:	Maximum number of optimization iterations allowed. DEFAULT = 20.
3	ICNDIR:	Conjugate direction restart parameter. DEFAULT = NDV+1.
4	NSCAL:	Scaling parameter. GT.0 - Scale design variables to order of magnitude one every NSCAL iterations. LT.0 - Scale design variables according to scaling values input. DEFAULT = No scaling.
5	ITRM:	Number of subsequent iterations which must satisfy relative or absolute convergence criterion before optimization process is terminated. DEFAULT = 3.
6	LINOBJ:	Linear objective function identifier. If the optimization objective is known to be a linear function of the design variables, set LINOBJ = 1. DEFAULT = Non-Linear.
7	NACMX1:	one plus the maximum number of active constraints anticipated. DEFAULT = NDV+2.

DATA BLOCK C (Continued)

FIELD		CONTENTS
8	NFDG:	Finite difference gradient identifier.
	0	All gradient information is computed by finite difference.
	1	Gradient of objective is computed anaytically. Gradients of constraints are computed by finite difference.
	2	All gradient information is computed analytically

REMARKS:

1) The value of NSCAL = 0 is suggested and ITRM = NACMX1 = 0 should be used.

2) The value of IPRINT may be reduced when the user becomes familiar with the optimization output.

3) The default values will be used if the card is either left blank or a value of zero is entered.

4) Because of the complexity of the problem it is necessary to have a large value for ITMAX so the problem will not be terminated prematurely. Recommended value is ITMAX = 40

5) The complexity of the condenser analysis ensure that no function can be considered linearly dependent on any combination of variables. This justifies using the DEFAULT value for LINOBJ.

DATA BLOCK D

DESCRIPTION: COPES Floating Point Optimization Program Parameters

FORMAT: 8F10

1	2	3	4	5	6	7	8
FDCH	FDCHM	CT	CTMIN	CTL	CTLMIN	THETA	PHI

FIELD		CONTENTS
1	FDCH:	Relative change in design variables in calculating finite difference gradients. DEFAULT = 0.01
2	FDCHM:	Minimum absolute step in finite difference gradient calculations. DEFAULT = 0.001.
3	CT:	Constraint thickness parameter. DEFAULT = -0.1.
4	CTMI:	Minimum absolute value of CT considered in the optimization process. DEFAULT = 0.004
5	CTL:	Constraint thickness parameter for linear and side constraints.
6	CTLMIN:	Minimum absolute value of CTL considered in the optimization process. DEFAULT = 0.001
7	THETA:	Mean value of push-off factor in the method of feasible directions. DEFAULT = 1.0
8	PHI:	Participation coefficient, used if one or more constraints are violated. DEFAULT = 5.0.

DATA BLOCK D (continued)

FORMAT: 2F10

1	2	3	4	5	6	7	8
DELFUN	DABFUN						

FIELD		CONTENTS
1	DELFUN:	Minimum relative change in objective function to indicate convergence of optimization process. DEFAULT = 0.001.
2	DABFUN:	Minimum absolute change in objective function to indicate convergence of the optimization process. DEFAULT = 0.001 times the initial objective value.

REMARKS:

1) Note that data for Data Block D is entered on two separate cards. A blank card indicates the default value is to be used.

2) If the NDV is greater than 3, the recommended value for FDCH is between 0.05 and 0.10.

DATA BLOCK E

DESCRIPTION: Total Number of Design Variables, Design Objective Identification and Sign on Design Objective.

FORMAT: 2I10, F10

1	2	3	4	5	6	7	8
NDVTOT	IOBJ	SGNOPT					

FIELD		CONTENTS
1	NDVTOT:	Total number of variables linked to the design variables. NDVTOT must be greater or equal to NDV. This option allows two or more parameters to be assigned to a single design variable. The value of each parameter is the value of the design variable times a multiplier which may be different for each parameter. DEFAULT = NDV.
2	IOBJ:	Global variable number associated with objective function in optimization or optimum sensitivity analysis.
3	SGNOPT:	Sign used on objective of optimization to identify whether function is to be maximized or minimized. +1.0 indicates maximization; -1.0 indicates minimization. DEFAULT = -1.0

REMARKS:

1) Currently there are not any variables in CONDIP which are linked to any of the design variables. Therefore the DEFAULT value is used for NDVTOT.

DATA BLOCK F

DESCRIPTION: Design Variable Bounds, Initial Values, and Scaling Factors.

FORMAT: 4F10

1	2	3	4	5	6	7	8
VLB	VUB	X	SCAL				

FIELD		CONTENTS
1	VLB:	Lower bound on the design variable.
2	VUB:	Upper bound on the design variable.
3	X:	Initial value of the design variable. If X is non-zero, this will supercede the value initialized by subroutine ANALIZ.
4	SCAL:	Design variable scale factor. Not used if NSCAL ≥ 0 in Block C

REMARKS:

1) There must be one separate data card for each design variable. Therefore there will be NDV data cards.

2) For all applications with CONDIP, initial values for the design variables will be entered through the INPUT subroutine called in ANALIZ.

DATA BLOCK G

DESCRIPTION: Design Variable Identification

FORMAT: 2I10,F10

1	2	3	4	5	6	7	8
NDSGN	IDSGN	AMULT					

FIELD		CONTENTS
1	NDSGN:	Design variable number associated with the variable.
2	IDSGN:	Global variable number associated with the variable.
3	AMULT:	Constant multiplier on the variable. The value of the variable will be the value of the design variable, NDSGN, times AMULT. DEFAULT = 1.0.

REMARKS:

1) There must be one separate card for each of the NDVTOT design variables. These data cards must follow the same order as the corresponding design variable parameter cards in Block F.

DATA BLOCK H

DESCRIPTION: Number of Constrained Parameters.

FORMAT: I10

1	2	3	4	5	6	7	8
NCONS							

FIELD		CONTENTS
1	NCONS:	Number of constraint SETS in the optimization problem.

REMARKS:

1) If two or more adjacent parameters in the Global common block have the same limits imposed, these are part of the same constraint set.

DATA BLOCK I

DESCRIPTION: Constraint Identification and Bounds.

FORMAT: 3I10

1	2	3	4	5	6	7	8
ICON	JCON	LCON					

FIELD		CONTENTS
1	ICON:	First Global number corresponding to the constraint set.
2	JCON:	Last Global number corresponding to the constraint set. DEFAULT = ICON.
3	LCON:	Linear constraint identifier for this set of constrained variables. LCON = 1 indicates linear constraints. DEFAULT = 0 = Nonlinear constraint.

REMARKS:

1) In CONDIP there is only Global number and thus one constraint that comprise a constraint set. Therefore the DEFAULT value is used for JCON.

2) All the constraints in this analysis are nonlinear. The DEFAULT value was therefore used for LCON as well.

3) This is the first card of a two card set which must be read together.

DATA BLOCK I (Continued)

FORMAT: 4F10

1	2	3	4	5	6	7	8
BL	SCAL1	BU	SCAL2				

FIELD		CONTENTS
1	BL:	Lower bound on the constrained variables. Value less than -2.0E+15 is assumed unbounded
2	SCAL1:	Normalization factor on lower bound. DEFAULT = Max of ABS(BL) or 0.1.
3	BU:	Upper bound on the constrained variables. Value greater than +2.0E+15 is assumed unbounded.
4	SCAL2	Normalization factor on upper bound. DEFAULT = Max of ABS(BU) or 0.1.

REMARKS:

1) The normalization factor can usually be defaulted, with the notable exception of exit-fraction. the normalization factor used for this constraint is usually ten times the upper bound.

DATA BLOCK P

DESCRIPTION: COPES Data 'END' Card.

FORMAT: 3A1

1	2	3	4	5	6	7	8
END							

FIELD	CONTENTS
1	The word 'END' in column 1-3.

REMARKS:

1) This card must appear at the end of the COPES data.

2) This ends the COPES input deck.

DATA BLOCK AA

DESCRIPTION: Geometry Option

FORMAT: I5

1	2	3	4	5	6	7	8
IOPT							

FIELD		CONTENTS
1	IOPT	Two condenser geometric options
	1	IOPT = 1. Number of condenser rows is a input as a constant; the number of tubes is a dependent variable. Use data blocks EE, FF, and GG.
	2	IOPT = 2. Number of tubes is allowed to be an independent variable and the number of rows is a dependent variable. Use data blocks HH and II. DEFAULT value is IOPT = 2.

REMARKS:

1) Data is right-justified and blanks will be interpreted as zeros.

2) If IOPT = 1, a smaller finite difference (FDCH) can be utilized in data Block D. This is because with this option the design analysis is less sensitive to the problems discussed earlier. Recommended using the DEFAULT value of 0.01 for FDCH.

DATA BLOCK BB

DESCRIPTION: Condenser Orientation

FORMAT: 1I5,3F10

1	2	3	4	5	6	7	8
ISEC	SECWID	PHI	PRCCLR				

FIELD		CONTENTS
1	ISEC	The number of sectors in the condenser.
2	SECWID	Sector width in degrees of arc.
3	PHI	Symmetry angle measure from the vertical.
4	PRCCLR	The percent of the tubes in the cooler.

REMARKS:

1) Data BB is required, no matter what geometry option is chosen.

2) The only limitation on ISEC and SECWID is that their product is less than 360 degrees. If the product is exactly 360 degrees, certain trigonometric functions will return a singularity.

3) PHI is that angle from the vertical that cuts the condenser in half.

4) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK CC

DESCRIPTION: Void Size, Tube Length, Row Spacing

FORMAT: 3F10

1	2	3	4	5	6	7	8
RADINS	ALST	RSPA					

FIELD		CONTENTS
1	RADINS	The inner void radius; feet.
2	ALST	The tube length; feet.
3	RSPA	Concentric row spacing about the void; inches.

REMARKS:

1) Data CC is required, no matter what geometry option is chosen.

DATA BLOCK DD

DESCRIPTION: Tube Material Parameters

FORMAT: I5,4F10

1	2	3	4	5	6	7	8
IWALL	XW	TUBESW	SKW	FOUL			

FIELD		CONTENTS
1	IWALL	A Flag indicating the tube thickness specification.
	1	IWALL = 1. Tube thickness is input as ratio of tube thickness to tube inner diameter.
	2	IWALL = 2. Tube thickness is input in inches.
2	XW	The input for wall thickness, dependent on the value for IWALL.
3	TUBESW	Specifc weight of the tube material; lbm/(cu.ft.)
4	SKW	Tube material thermal conductivity; (btu-ft)/(sq.ft.-hr-^{o}F)
5	FOUL	Tube fouling factor.

REMARKS:

1) Data DD is required, no matter what geometry option is chosen.

2) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK EE

DESCRIPTION: Number of Rows

FORMAT: I5

1	2	3	4	5	6	7	8
NOROWS							

FIELD		CONTENTS
1	NOROWS	The number of concentric rows in the condenser bundle built around the center void

REMARKS:

1) Data EE is used only when IOPT = 1

2) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK FF

DESCRIPTION: Tube Inner Diameter

FORMAT: 1I5,2F10

1	2	3	4	5	6	7	8
MDIAM	SIDO	SIDI					

FIELD		CONTENTS
1	MDIAM	A flag to indicate whether tube inner diameter will linearly vary by row through the condenser bundle.
	1	MDIAM = 1. Tube inner diameter is uniform through the condenser bundle.
	2	MDIAM = 2. Tube inner diameter varies linearly through the bundle by row.
2	SIDO	Tube inner diameter of the outer row; inches.
3	SIDI	Tube inner diameter of the inner row; inches.

REMARKS:

1) Data FF is used only when IOPT = 1

2) Data is right-justified and blanks will be interpreted as zeros.

3) Cooler tubes use the inner diameter of the innermost bundle row.

4) The DEFAULT value is MDIAM = 1

DATA BLOCK GG

DESCRIPTION: Tube Pitch

FORMAT: 1I5,2F10

1	2	3	4	5	6	7	8
MPITCH	SDDO	SDDI					

FIELD		CONTENTS
1	MPITCH	A flag to indicate whether tube pitch will linearly vary by row through the condenser bundle.
	1	MPITCH = 1. Tube pitch is uniform through the condenser bundle.
	2	MPITCH = 2. Tube pitch varies linearly through the bundle by row.
2	SDDO	Tube pitch of the outer row;
3	SDDI	Tube pitch of the inner row;

REMARKS:

1) Data GG is used only when IOPT = 1

2) Data is right-justified and blanks will be interpreted as zeros.

3) Cooler tubes use the pitch of the innermost bundle row.

4) The DEFAULT value is MPITCH = 1

DATA BLOCK HH

DESCRIPTION: Tube Inner Diameter and Tube Pitch

FORMAT: 2F10

1	2	3	4	5	6	7	8
SIDO	SDDO						

FIELD		CONTENTS
1	SIDO	Tube inner diameter for the entire condenser; inches
2	SDDO	Tube pitch for the entire condenser

REMARKS:

1) Data HH is used only when IOPT = 2.

2) Data is right-justified and blanks will be interpreted as zeros.

3) In the calculations, SIDI is set equal to SIDO and SDDI is set equal to SDDO. This avoids the need for two systems of nomenclature for each geometry option.

DATA BLOCK II

DESCRIPTION: Total Number of Tubes in the Condenser

FORMAT: F12

1	2	3	4	5	6	7	8
TNOTOT							

FIELD		CONTENTS
1	TNOTOT	The total number of tubes in the condenser (cooler and the bundle).

REMARKS:

1) Data II is used only when IOPT = 2.

2) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK JJ

DESCRIPTION: Inlet Steam Mixture.

FORMAT: 15,2F10

1	2	3	4	5	6	7	8
JGAS	WSI	WNCIR					

FIELD		CONTENTS
1	JGAS	A Flag indicating the type of non-condensable gas entering the system.
	1	JGAS = 1. This indicates that the gas is air.
	2	JGAS = 2. This indicates that the gas is carbon dioxide.
	3	JGAS = 3. This indicates that the gas is a mixture of the two.
2	WSI	Steam flow rate entering the condenser; lbm/hr.
3	WNCIR	Ratio of the non-condensable gas flow to inlet steam flow; lbm/hr.

REMARKS:

1) Data JJ is required, no matter what geometry option is chosen.

2) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK KK

DESCRIPTION: Inlet Temperatures.

FORMAT: 2F10

1	2	3	4	5	6	7	8
STBI	STSAT1						

FIELD CONTENTS

1	STBI	Coolant inlet temperature; °F.
2	STSAT1	Inlet steam saturation temperature; °F.

REMARKS:

1) Data KK is required no matter, what geometry option is chosen.

2) Data is right-justified and blanks will be interpreted as zeros.

DATA BLOCK LL

DESCRIPTION: Cooling Water Parameters

FORMAT: 2F10

1	2	3	4	5	6	7	8
IFLOW	X5						

FIELD		CONTENTS
1	IFLOW	A control flag for cooling water specifications.
	1	IFLOW = 1. Input pressure drop across cooling water headers in psia.
	2	IFLOW = 2. Input cooling water velocity in ft/sec.
	3	IFLOW = 3. Input coolant flow in lbm/hr.
2	X5	Actually input the value for flow into this variable. The specification for flow to be determined by IFLOW

REMARKS:

1) Data LL is required, no matter what geometry option is chosen.

2) Data is right-justified and blanks will be interpreted as zeros.

3) X5 acts as a temporary all-purpose storage variable for whatever expression for coolant flow is used.

DATA BLOCK MM

DESCRIPTION: Internal Enhancement Regions

FORMAT: I5

1	2	3	4	5	6	7	8
NEI							

FIELD		CONTENTS
1	NEI	Number of internal enhancement regions. A value between 1 and 6.

REMARKS:

1) Data MM is optional. There must be NEI subsequent data cards providing the necessary parameters for each region.

2) enhancement can only be used if IOPT = 1.

3) Data is right-justified and blanks will be interpreted as zeros.

4) This value was zero for all runs.

DATA BLOCK NN

DESCRIPTION: Internal Enhancement Parameters

FORMAT: 2I5,3F10

1	2	3	4	5	6	7	8
NRNI	NETI	ENI	BC	BE			

FIELD		CONTENTS
1	NRNI	Row number of first row in internal enhancement region.
2	NETI	Number of tubes in each internal enhancement region.
3	ENI	Internal heat transfer enhancement factor.
4	BC	Coefficient in internallyenhanced tube coolant pressure drop calculation.
5	BE	Exponent in coolant pressure drop calculation.

REMARKS:

1) Data NN is optional. However, if NEI is greater than zero then there must NEI "NN" data cards to provide the necessary data for each enhancement region.

2) enhancement can only be used if IOPT = 1.

3) Data is right-justified and blanks will be interpreted as zeros.

4) These values are constant for entire run and cannot be changed by the optimizer.

5) This value was zero for all runs.

DATA BLOCK OO

DESCRIPTION: External Enhancement Regions

FORMAT: I5

1	2	3	4	5	6	7	8
NEE							

FIELD		CONTENTS
1	NEE	Number of external enhancement regions. A value between 1 and 6.

REMARKS:

1) Data OO is optional. There must be NEE subsequent data cards providing the necessary parameters for each region.

3) Data is right-justified and blanks will be interpreted as zeros.

4) This value was zerc for all runs.

DATA BLOCK PP

DESCRIPTION: External Enhancement Parameters

FORMAT: 2I5,2F10

1	2	3	4	5	6	7	8
NRNE	NETE	ENO	ENH				

FIELD		CONTENTS
1	NRNE	Row number of first row in external enhancement region.
2	NETE	Number of tubes in each external enhancement region.
3	ENI	External heat transfer enhancement factor.
4	ENH	Steam-side pressure drop factors.

REMARKS:

1) Data PP is optional. However, if NEE is greater than zero then there must NEE "PP" data cards to provide the necessary data for each enhancement region.

2) Data is right-justified and blanks will be interpreted as zeros.

3) Enhancement can only be used if IOPT = 1.

4) These values are constant for an entire run and cannot be changed by the optimizer.

5) This value was zero for all runs.

DATA BLOCK QQ

DESCRIPTION: Baffle Options

FORMAT: I5

1	2	3	4	5	6	7	8
IBAF							

FIELD		CONTENTS
1	IBAF	A flag to be used to determine baffle number and location.

REMARKS:

1) Data QQ is optional.

2) Data is right-justified and blanks will be interpreted as zeros.

3) These values are constant for an entire run and cannot be changed by the optimizer.

4) Additional specified baffles were not used in any of the runs. This value was zero for all runs.

DATA BLOCK RR

DESCRIPTION: Baffle Location

FORMAT: I5

1	2	3	4	5	6	7	8
JBAF							

FIELD		CONTENTS
1	JBAF	An array containing baffle locations

REMARKS:

1) Data RR is optional.

2) Data is right-justified and blanks will be interpreted as zeros.

3) These values are constant for an entire run and cannot be changed by the optimizer.

4) Additional specified baffles were not used in any of the runs. This value was zero for all runs.

DATA BLOCK SS

DESCRIPTION: Detailed Printout

FORMAT: I5

1	2	3	4	5	6	7	8
IPRT							

FIELD		CONTENTS
1	IPRT	A flag to generate a detailed output of the condenser analysis (OUT3)

REMARKS:

1) Data SS is optional.

2) Data is right-justified and blanks will be interpreted as zeros.

3) These values are constant for an entire run and cannot be changed by the optimizer.

4) This value was zero for all runs.

```
VOLUME MINIMIZATION - CIRCULAR CONDENSOR
2,8
1,40,,9,,,20
.10

1,20,-1.
1.0,2.0E+15
1000.,8000.
1.1,2.
3.,9.0
.022,.109
.407,1.206
.1,1.
2.,10.
1,1
2,22
3,14
4,17
5,19
6,7
7,10
8,29
5
4
-2.0E+15,,.01,.01
23
.5,,5.
24
.1,,.30
26
.625,,1.25
28
-2.E+15,,200.
END
     2
    12   29.99    180.          7.0
   0.55           10.3
     2    .049       282.          26.0     .0002
    .527           1.40
    5230.0
     1   161970.   .0000371
   76.66    110.52
     2    8.473
     0
     0
     0
     0
```

Figure B.1   Sample Data Input.

APPENDIX C
CONDIP LISTING

The following Appendix contains a complete listing for CONDIP. An effort has been made to make the program as readable as possible through liberal use of comment cards.

```
      SUBROUTINE ANALIZ (ICALC)
C     IF (ICALC.GT.2) RETURN
      IF (ICALC.GT.1) GO TO 10
      ICOUNT=0
      CALL INPUT (IPRT)
      CALL OUT1
10    IF (ICALC.EQ.3) GO TO 20
      CALL ORCON (ICALC)
      GO TO 30
20    CALL OUT2
C     ICOUNT = 1
      IF (IPRT.NE.0.AND.ICOUNT.NE.0) CALL OUT3
      ICOUNT=ICOUNT+1
30    RETURN
      END
C
C*****************************************************************
C*****************************************************************
C
      SUBROUTINE INPUT (IPRT)
C--------------
C INP 1
C                   SUBROUTINE INPUT
C
C               INPUT CALLED BY ANALIZ WITH ICALC = 1
C
C ZZZ INP-1
C--------------
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTO
     2T,BNDRAD,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,NNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /INPT1/ IFLOW,IWALL
C$-------------
C INP 2
C                   READ ENTERING DATA SET
C
C         ALST = TUBE LENGTH IN FEET
C         BC = COEFICIENT IN INTERNALLY ENHANCED TUBE COOLANT PRESSURE
C              DROP CALCULATION.
C         BE = EXPONENT IN COOLANT PRESSURE DRCP CALCULATION.
C         DELWP = PRESSURE DROP ACROSS COOLANT WATER HEADERS,  PSI
C         ENI = INTERNAL HEAT TRANSFER ENHANCEMENT FACTORS.
C         ENH = STM SIDE PRESSURE DROP FACTORS.
C         ENO = EXTERNAL HEAT TRANSFER ENHANCEMENT FACTORS.
```

```
C          FOUL = TUBE FOULING FACTOR.                                  CON00490
C          GFLOW = COOLANT MASS FLOW RATE  LBM/HR                       CON00500
C          IBAF = CONTROL FLAG FOR BAFFLE OPTIONS                       CON00510
C          IFLOW = CONTROL FLAG FOR COOLING WATER SPECIFICATION.        CON00520
C          IPRT = CONTROL FLAG FOR OUTPUT.  (EXTENDED OUTPUT IF IPRT>0.) CON00530
C          ISEC = NO. OF SECTORS IN CONDENSER                           CON00540
C          IWALL = CONTROL FLAG FOR TUBE THICKNESS SPECIFICATION.       CON00550
C          JBAF = ARRAY CONTAINING BAFFLE LOCCATIONS.                   CON00560
C          JGAS = N/C GAS IDENTIFIER 1,2, OR 3                          CON00570
C          MDIAM = CONTROL FLAG FOR LINEAR VARIATION IN TUBE DIAMETER.  CON00580
C          MPITCH = CONTROL FLAG FOR LINEAR VARIATION IN TUBE PITCH.    CON00590
C          NEE = NUMBER OF EXTERNAL ENHANCEMENT REGIONS.                CON00600
C          NEI = NUMBER OF INTERNAL ENHANCEMENT REGIONS.                CON00610
C          NETE = NUMBER OF TUBES IN EACH EXTERNAL ENHANCEMENT REGION.  CON00620
C          NETI = NUMBER OF TUBES IN EACH INTERNAL ENHANCEMENT REGION.  CON00630
C          NOROWS = NUMBER OF ROWS IN THE CONDENSER                     CON00640
C          NRNE = ROW NUMBER OF FIRST ROW IN EXTERNAL ENHANCEMENT REGION. CON00650
C          NRNI = ROW NUMBER OF FIRST ROW IN INTERNAL ENHANCEMENT REGION. CON00660
C          SIDI = TUBE ID AT INSIDE ROW - IN                            CON00670
C          SIDO = TUBE ID AT OUTSIDE ROW - IN                           CON00680
C          IOPT = FLAG TO INDICATE WHICH GEOMETRY MODULE IS BEING USED  CON00690
C                IF OPT=1 THEN NOROWS IS A CONSTANT IN THE MODEL AND    CON00700
C                VARIATION IN PITCH AND TUBE DIAMETER IS PERMITTED.     CON00710
C                IF OPT=2 THEN TNOTOT IS A DESIGN VARIABLE AND LINEAR   CON00720
C                VARIATIONS ARE NOT PERMITTED.                          CON00730
C          PHI   = SYMETRY ANGLE MEASURED FROM THE VERTICAL             CON00740
C          RADINS = BUNDLE INSIDE RADIUS - FT                           CON00750
C          RSPA = TUBE ROW RADIAL SEPARATION - IN                       CON00760
C          SDDI = PITCH AT INSIDE ROW                                   CON00770
C          SDDO = PITCH AT OUTSIDE ROW                                  CON00780
C          SECWID = SECTOR WIDTH IN DEGREES OF ARC                      CON00790
C          SKW = TUBE MATERIAL THERMAL CONDUCTIVITY  BTU-FT/HR-FT**2-F  CON00800
C          STBI = COOLANT INLET TEMP    F                               CON00810
C          STSAT1 = INLET STM SAT TEMP    F                             CON00820
C          TNOTOT = TOTAL NUMBER OF TUBES IN THE CONDENSOR/COOLER       CON00830
C          TUBESW = SPECIFIC WEIGHT OF THE TUBE MATERIAL  LBM/CU FT     CON00840
C          VELBI = COOLANT FLOW VELOCITY  FT/SEC                        CON00850
C          WNCIR = RATIO OF N/C MASS FLOW TO STEAM MASS FLOW            CON00860
C          WSI = STM FLOW RATE  LBM/HR                                  CON00870
C          XW = TEMP INPUT VALUE OF XW1 AND XW2                         CON00880
C          XW1 = RATIO OF TUBE WALL THICKNESS TO TUBE ID                CON00890
C          XW2 = TUBE WALL THICKNESS  INCHES                            CON00900
C                                                                       CON00910
C ZZZ INP-2                                                             CON00920
C--------------                                                         CON00930
      READ (5,330) IOPT                                                 CON00940
      READ (5,340) ISEC,SECWID,PHI,PRCCLR                               CON00950
      READ (5,360) RADINS,ALST,RSPA                                     CON00960
```

```
      READ (5,400) IWALL,XW,TUBESW,SKW,FOUL                             CON00970
C                                                                       CON00980
      IF ((IOPT.GE.2).OR.(IOPT.LT.1)) GO TO 10                          CON00990
C                                                                       CON01000
      READ (5,380) NOROWS                                               CON01010
      READ (5,370) MDIAM,SIDO,SIDI                                      CON01020
      READ (5,370) MPITCH,SDDO,SDDI                                     CON01030
      GO TO 20                                                          CON01040
10    READ (5,360) SIDO,SDDO                                            CON01050
      READ (5,390) TNOTOT                                               CON01060
      MPITCH=1                                                          CON01070
      MDIAM=1                                                           CON01080
C                                                                       CON01090
20    IF (MDIAM.EQ.1) SIDI=SIDO                                         CON01100
      IF (MPITCH.EQ.1) SDDI=SDDO                                        CON01110
      IW=IWALL                                                          CON01120
      IF (IWALL.NE.2) IW=1                                              CON01130
      XW1=0.                                                            CON01140
      XW2=0.                                                            CON01150
      IF (IW.EQ.1) XW1=XW                                               CON01160
      IF (IW.EQ.2) XW2=XW                                               CON01170
      READ (5,350) JGAS,WSI,WNCIR                                       CON01180
      READ (5,410) STBI,STSAT1                                          CON01190
      READ (5,420) IFLOW,X5                                             CON01200
      IFL=IFLOW                                                         CON01210
      GFLOW=0.                                                          CON01220
      VELBI=0.                                                          CON01230
      DELWP=0.                                                          CON01240
      IF (IFLOW.EQ.1) DELWP=X5                                          CON01250
      IF (IFLOW.EQ.2) VELBI=X5                                          CON01260
      IF (IFLOW.EQ.3) GFLOW=X5                                          CON01270
      READ (5,430) NEI                                                  CON01280
      IF (NEI.EQ.0) GO TO 40                                            CON01290
      IF ((NEI.LT.0).AND.(NEI.GT.6)) GO TO 120                          CON01300
      DO 30 I=1,NEI                                                     CON01310
30    READ (5,440) NRNI(I),NETI(I),ENI(I),BC(I),BE(I)                   CON01320
40    READ (5,430) NEE                                                  CON01330
      IF (NEE.EQ.0) GO TO 60                                            CON01340
      IF ((NEE.LE.0).OR.(NEE.GT.6)) GO TO 120                           CON01350
      DO 50 I=1,NEE                                                     CON01360
50    READ (5,450) NRNE(I),NETE(I),ENO(I),ENH(I)                        CON01370
60    READ (5,430) IBAF                                                 CON01380
      IF ((IBAF.LE.-2).OR.(IBAF.GT.ISEC)) GO TO 120                     CON01390
      IF ((IBAF.EQ.0).OR.(IBAF.EQ.-1)) GO TO 80                         CON01400
      DO 70 I=1,IBAF                                                    CON01410
70    READ (5,430) JBAF(I)                                              CON01420
80    CONTINUE                                                          CON01430
      READ (5,430) IPRT                                                 CON01440
```

```
C$-------------
C INP 3
C                    TEST FOR RECOVERABLE INPUT ERRORS
C
C ZZZ INP-3
C-------------
      IF ((IOPT.EQ.1).OR.(IOPT.EQ.2)) GO TO 90
      WRITE (6,460)
      WRITE (6,470) IOPT
      WRITE (6,510)
      IOPT=2
90    IF (IOPT.EQ.2) GO TO 110
      IF ((MDIAM.EQ.2).OR.(MDIAM.EQ.1)) GO TO 100
      WRITE (6,460)
      WRITE (6,480) MDIAM
      WRITE (6,510)
      MDIAM=1
100   IF ((MPITCH.EQ.1).OR.(MPITCH.EQ.2)) GO TO 110
      WRITE (6,460)
      WRITE (6,490) MPITCH
      WRITE (6,510)
      MPITCH=1
110   IF ((IWALL.EQ.1).OR.(IWALL.EQ.2)) GO TO 120
      WRITE (6,460)
      WRITE (6,500) IWALL
      WRITE (6,510)
      IWALL=1
C$-------------
C INP 4
C                    LOCATE NON-RECOVERABLE INPUT ERRORS
C
C ZZZ INP-4
C-------------
120   IERR=0
      IF ((ISEC.GT.0).AND.(ISEC.LE.15)) GO TO 130
      IERR=1
      WRITE (6,520)
      WRITE (6,530) ISEC
130   X2=SECWID*FLOAT(ISEC)
      IF ((X2.LE.360.).AND.(X2.GT.0.)) GO TO 140
      WRITE (6,520)
      WRITE (6,540) X2
      WRITE (6,550) ISEC,SECWID
      IERR=1
140   IF (IOPT.EQ.2) GO TO 150
      IF ((NOROWS.GT.0).AND.(NOROWS.LE.99)) GO TO 160
      IERR=1
      WRITE (6,520)
```

```
      WRITE (6,560) NCROWS                                              CONO1930
      GO TO 160                                                         CONO1940
C                                                                       CONO1950
150   IF (TNOTOT.GT.0) GO TO 160                                        CONO1960
      IERR=1                                                            CONO1970
      WRITE (6,520)                                                     CONO1980
      WRITE (6,570) TNOTOT                                              CONO1990
C                                                                       CONO2000
160   IF (RADINS.GT.0.) GO TO 170                                       CONO2010
      IERR=1                                                            CONO2020
      WRITE (6,520)                                                     CONO2030
      WRITE (6,580) RADINS                                              CONO2040
170   IF (ALST.GT.0.) GO TO 180                                         CONO2050
      IERR=1                                                            CONO2060
      WRITE (6,520)                                                     CONO2070
      WRITE (6,590) ALST                                                CONO2080
180   IF (SIDO.GT.0.) GO TO 190                                         CONO2090
      IERR=1                                                            CONO2100
      WRITE (6,520)                                                     CONO2110
      WRITE (6,600) SIDO                                                CONO2120
190   IF ((MDIAM.EQ.1).OR.(SIDI.GT.0.)) GO TO 200                       CONO2130
      IERR=1                                                            CONO2140
      WRITE (6,520)                                                     CONO2150
      WRITE (6,610) SIDI                                                CONO2160
200   IF (SDDO.GT.1.) GO TO 210                                         CONO2170
      IERR=1                                                            CONO2180
      WRITE (6,520)                                                     CONO2190
      WRITE (6,620) SDDO                                                CONO2200
210   IF ((MPITCH.EQ.1).OR.(SDDI.GT.1.)) GO TO 220                      CONO2210
      IERR=1                                                            CONO2220
      WRITE (6,520)                                                     CONO2230
      WRITE (6,630) SDDI                                                CONO2240
220   IF (IW.EQ.2) GO TO 230                                            CONO2250
      IF (XW1.GT.0.) GO TO 240                                          CONO2260
      IERR=1                                                            CONO2270
      WRITE (6,520)                                                     CONO2280
      WRITE (6,640) XW1                                                 CONO2290
      GO TO 240                                                         CONO2300
230   X=XW2*2.                                                          CONO2310
      IF (X.GT.0.) GO TO 240                                            CONO2320
      IERR=1                                                            CONO2330
      WRITE (6,520)                                                     CONO2340
      WRITE (6,650) XW2                                                 CONO2350
240   IF ((WNCIR.GE.0.).AND.(WNCIR.LT.1.)) GO TO 250                    CONO2360
      IERR=1                                                            CONO2370
      WRITE (6,520)                                                     CONO2380
      WRITE (6,660) WNCIR                                               CONO2390
250   IF ((STBI.GT.-32.).AND.(STBI.LT.212)) GO TO 260                   CONO2400
```

```
      IERR=1                                                            CONO2410
      WRITE (6,520)                                                     CONO2420
      WRITE (6,670) STBI                                                CONO2430
  260 IF (STBI.LT.STSAT1) GO TO 270                                     CONO2440
      IERR=1                                                            CONO2450
      WRITE (6,520)                                                     CONO2460
      WRITE (6,680) STBI,STSAT1                                         CONO2470
  270 IF ((IFLOW.GT.0).OR.(IFLOW.LE.3)) GO TO 280                       CONO2480
      IERR=1                                                            CONO2490
      WRITE (6,520)                                                     CONO2500
      WRITE (6,690) IFLOW                                               CONO2510
  280 IF ((NEI.GE.0).AND.(NEI.LE.6)) GO TO 290                          CONO2520
      IERR=1                                                            CONO2530
      WRITE (6,520)                                                     CONO2540
      WRITE (6,700) NEI                                                 CONO2550
  290 IF ((NEE.GE.0).AND.(NEE.LE.6)) GO TO 300                          CONO2560
      IERR=1                                                            CONO2570
      WRITE (6,520)                                                     CONO2580
      WRITE (6,710) NEE                                                 CONO2590
  300 IF ((IBAF.GE.-1).AND.(IBAF.LE.ISEC)) GO TO 310                    CONO2600
      IERR=1                                                            CONO2610
      WRITE (6,520)                                                     CONO2620
      WRITE (6,720) IBAF,ISEC                                           CONO2630
      WRITE (6,730)                                                     CONO2640
  310 IF (IERR.EQ.0) GO TO 320                                          CONO2650
      STOP                                                              CONO2660
  320 RETURN                                                            CONO2670
  330 FORMAT (I5)                                                       CONO2680
  340 FORMAT (I5,3F10.5)                                                CONO2690
  350 FORMAT (I5,F10.5,F10.7)                                           CONO2700
  360 FORMAT (4F10.5)                                                   CONO2710
  370 FORMAT (I5,F10.5,F10.5)                                           CONO2720
  380 FORMAT (I5)                                                       CONO2730
  390 FORMAT (F12.5)                                                    CONO2740
  400 FORMAT (I5,4F10.5)                                                CONO2750
  410 FORMAT (2F10.5)                                                   CONO2760
  420 FORMAT (I5,F10.5)                                                 CONO2770
  430 FORMAT (I5)                                                       CONO2780
  440 FORMAT (2I5,3F10.5)                                               CONO2790
  450 FORMAT (2I5,2F10.5)                                               CONO2800
  460 FORMAT (1H0,31H***  WARNING - INPUT ERROR  ***)                   CONO2810
  470 FORMAT (1H ,22H INPUT VALUE OF OPT = ,I5,27H OUT OF RANGE, OPT SETCONO2820
     1 TO 2)                                                            CONO2830
  480 FORMAT (1H ,21HINPUT VALUE OF MDIAM=,I5,29H OUT OF RANGE, MDIAM SECONO2840
     1T TO 1)                                                           CONO2850
  490 FORMAT (1H ,22HINPUT VALUE OF MPITCH=,I5,30H OUT OF RANGE, MPITCH CONO2860
     1SET TO 1)                                                         CONO2870
  500 FORMAT (1H ,21HINPUT VALUE OF IWALL=,I5,29H OUT OF RANGE, IWALL SECONO2880
```

```
     1T TO 1)                                                                  CONO2890
510   FORMAT (1H ,26H   EXECUTION CONTINUES ...)                               CONO2900
520   FORMAT (1HO,38H****  RUN TERMINATED DURING INPUT ****)                   CONO2910
530   FORMAT (1H ,43HRANGE OF ALLOWABLE ISEC IS 1 THRU 15, ISEC=,I5)           CONO2920
540   FORMAT (1H ,60HCONDENSER ANGULAR WIDTH IS OUT OF RANGE 0 TO 360 DECONO2930
     1G, WIDTH=,E10.5,5H DEG.)                                                 CONO2940
550   FORMAT (1H ,5HISEC=,I5,10H   SECWID=,E10.3)                              CONO2950
560   FORMAT (1H ,49HNUMBER OF ROWS MUST BE IN RANGE 0 TO 100, NOROWS=,ICONO2960
     15)                                                                       CONO2970
570   FORMAT (1H ,51HTHE NUMBER OF TUBES MUST BE GREATER THAN 0, TNOTOT=CONO2980
     1,F12.5)                                                                  CONO2990
580   FORMAT (1H ,48HBUNDLE RADIUS MUST BE GREATER THAN ZERO, RADINS=,E1CONO3000
     10.3)                                                                     CONO3010
590   FORMAT (1H ,44HTUBE LENGTH MUST BE GREATER THAN ZERO, ALST=,E10.3)CONO3020
600   FORMAT (1H ,50HTUBE INNER DIAMETER MUST BE GREATER THAN 0, SIDO =,CONO3030
     1E10.3)                                                                   CONO3040
610   FORMAT (1H ,50HTUBE INNER DIAMETER MUST BE GREATER THAN 0, SIDO =,CONO3050
     1E10.3)                                                                   CONO3060
620   FORMAT (1H ,40HTUBE PITCH MUST BE GREATER THAN 1, SDDO=,E10.3)      CONO3070
630   FORMAT (1H ,40HTUBE PITCH MUST BE GREATER THAN 1, SDDI=,E10.3)      CONO3080
640   FORMAT (1H ,75HRATIO OF TUBE THICKNESS TO TUBE INNER DIAMETER MUSTCONO3090
     1 BE GREATER THAN 0, XW1=,E10.3)                                          CONO3100
650   FORMAT (1H ,49HTUBE THICKNESS MUST BE GREATER THAN 0, XW2=,E10.3)  CONO3110
660   FORMAT (1H ,93HRATIO OF N/C GASSES TO STM FLOW MUST BE LESS THAN OCONO3120
     1NE  AND GREATER THAN EQUAL TO ZERO, WNCIR=,E10.3)                        CONO3130
670   FORMAT (1H ,63HINJECTION TEMP MUST BE GREATER THAN 32 AND LESS THACONO3140
     1N 212, STBI=,E10.3)                                                      CONO3150
680   FORMAT (1H ,58HINJECTION TEMP MUST BE LESS THAN EXHAUST TRUNK TEMPCONO3160
     1, STBI=,E10.3,8H  STSAT=,E10.3)                                          CONO3170
690   FORMAT (1H ,31HIFLOW MUST BE 1,2 OR 3.  IFLOW=,I5)                       CONO3180
700   FORMAT (1H ,73HNUMBER OF INTERNAL ENHANCEMENT REGIONS MUST BE BETWCONO3190
     1EEN ZERO AND SIX, NEI=,I5)                                               CONO3200
710   FORMAT (1H ,73HNUMBER OF EXTERNAL ENHANCEMENT REGIONS MUST BE BETWCONO3210
     1EEN ZERO AND SIX, NEE=,I5)                                               CONO3220
720   FORMAT (1H ,50HIBAF MUST BE BETWEEN -1 AND ISEC INCLUSIVE.  IBAF=,CONO3230
     1I5,7H  ISEC=,I5)                                                         CONO3240
730   FORMAT (1H ,40H(IF USER WANTS IBAF=ISEC+1, SET IBAF=-1))                CONO3250
      END                                                                      CONO3260
C                                                                              CONO3270
C***********************************************************************      CONO3280
C***********************************************************************      CONO3290
C                                                                              CONO3300
      SUBROUTINE OUT1                                                          CONO3310
C-------------                                                                 CONO3320
C OUT1 1                                                                       CONO3330
C                    OUTPUT OF ENTERING DATA SET                               CONO3340
C ZZZ OUT1-1                                                                   CONO3350
C               SEE INP-2 FOR VARIABLE DEFINITIONS                             CONO3360
```

```
C---------------                                                         CONO3370
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,CONO3380
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTOCONO3390
     2T,BNDRAD,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR                        CONO3400
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,SCONO3410
     1TSAT1,TUBESW,NNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MDCONO3420
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)CONO3430
     3,WSI,PFILL                                                          CONO3440
      COMMON /INPT1/ IFLOW,IWALL                                          CONO3450
      WRITE (6,60)                                                        CONO3460
      WRITE (6,70)                                                        CONO3470
      WRITE (6,80)                                                        CONO3480
      WRITE (6,90)                                                        CONO3490
      WRITE (6,100)                                                       CONO3500
      WRITE (6,90)                                                        CONO3510
      WRITE (6,80)                                                        CONO3520
      WRITE (6,110)                                                       CONO3530
      WRITE (6,120) ISEC                                                  CONO3540
      WRITE (6,130) SECWID                                                CONO3550
      WRITE (6,140) PHI                                                   CONO3560
      WRITE (6,150) PRCCLR                                                CONO3570
      IF (IOPT.EQ.2) WRITE (6,170) TNOTOT                                 CONO3580
      IF (IOPT.EQ.2) GO TO 10                                             CONO3590
      WRITE (6,160) NOROWS                                                CONO3600
   10 WRITE (6,180) RADINS                                                CONO3610
      WRITE (6,190) ALST                                                  CONO3620
      WRITE (6,200) MDIAM                                                 CONO3630
      IF (MDIAM.EQ.0) GO TO 20                                            CONO3640
      WRITE (6,210) SIDO                                                  CONO3650
      WRITE (6,240)                                                       CONO3660
      GO TO 30                                                            CONO3670
   20 WRITE (6,220) SIDO,SIDI                                             CONO3680
   30 WRITE (6,230) MPITCH                                                CONO3690
      IF (MPITCH.EQ.0) GO TO 40                                           CONO3700
      WRITE (6,260) SDDO                                                  CONO3710
      WRITE (6,280)                                                       CONO3720
      GO TO 50                                                            CONO3730
   40 WRITE (6,270) SDDO,SDDI                                             CONO3740
   50 WRITE (6,250) IWALL                                                 CONO3750
      IF (IWALL.EQ.1) WRITE (6,300) XW1                                   CONO3760
      IF (IWALL.EQ.2) WRITE (6,290) XW2                                   CONO3770
      WRITE (6,310) TUBESW                                                CONO3780
      WRITE (6,320) SKW                                                   CONO3790
      WRITE (6,330) FOUL                                                  CONO3800
      IF (JGAS.EQ.1) WRITE (6,360)                                        CONO3810
      IF (JGAS.EQ.2) WRITE (6,370)                                        CONO3820
      IF (JGAS.EQ.3) WRITE (6,380)                                        CONO3830
      WRITE (6,340) WSI                                                   CONO3840
```

```
      WRITE (6,350) WNCIR                                                CONO3850
      WRITE (6,390) STBI                                                 CONO3860
      WRITE (6,400) STSAT1                                               CONO3870
      WRITE (6,410) IFLOW                                                CONO3880
      IF (IFLOW.EQ.3) WRITE (6,420) GFLOW                                CONO3890
      IF (IFLOW.EQ.2) WRITE (6,430) VELBI                                CONO3900
      IF (IFLOW.EQ.1) WRITE (6,440) DELWP                                CONO3910
      WRITE (6,450) NEI                                                  CONO3920
      WRITE (6,460) NEE                                                  CONO3930
      WRITE (6,470) IBAF                                                 CONO3940
      RETURN                                                             CONO3950
60    FORMAT (1H1)                                                       CONO3960
70    FORMAT (1H ,51H                   GLOBAL COMMON BLOCK              CONO3970
     1/62H    1       2        3        4         5       6       7        8      9   CONO3980
     2/62H ALST    DELWP DELWPC EXITOA  GFLOW   SIDI    SIDO    PHP   RSPA  CONO3990
     3//62H     10       11       12       13       14       15       16       17      18 CONO4000
     4 /61H   RADINS   REWI   REWO    SDDI    SDDO    SLDI    SLDO   VELBI   XW1 CONO4010
     5//62H   19       20       21       22       23       24       25       26     CONO4020
     6 /56H  XW2     VOL1   VOL2  TNOTOT  BNDRAD   ARATIO   ODII    ODOI//) CONO4030
80    FORMAT (1H ,15X,16X,28H****************************)               CONO4040
90    FORMAT (1H ,15X,16X,1H*,26X,1H*)                                   CONO4050
100   FORMAT (1H ,15X,16X,28H* CONDIP ENTERING DATA SET *)               CONO4060
110   FORMAT (1H0///)                                                    CONO4070
120   FORMAT (1H ,15X,17HNUMBER OF SECTORS,12X,I8)                       CONO4080
130   FORMAT (1H ,15X,12HSECTOR WIDTH,18X,F9.1,10X,11HDEG. OF ARC)       CONO4090
140   FORMAT (1H ,15X,24HBUNDLE ORIENTATION ANGLE,6X,F9.1,7X,14HDEG. FROCONO4100
     1M VERT)                                                            CONO4110
150   FORMAT (1H ,15X,53H% OF TOTAL TUBES ASSIGNED TO COOLER IS PRCCLR   CONO4120
     1   = ,F5.2/)                                                       CONO4130
160   FORMAT (1H ,15X,19HNUMBER OF TUBE ROWS,13X,I5)                     CONO4140
170   FORMAT (1H ,15X,32HNUMBER OF TUBES IN THE CONDENSOR,8X,F12.5)      CONO4150
180   FORMAT (1H ,15X,20HBUNDLE INSIDE RADIUS,12X,F10.4,16X,4HFEET)      CONO4160
190   FORMAT (1H ,15X,11HTUBE LENGTH,19X,F10.2,5X,11X,4HFEET/)           CONO4170
200   FORMAT (1H ,15X,37HFLAG FOR TUBE OD VARIATION IS MDIAM =,I2)       CONO4180
210   FORMAT (1H ,15X,20HTUBE INSIDE DIAMETER,9X,F10.4,14X,6HINCHES)     CONO4190
220   FORMAT (1H ,15X,20HTUBE INSIDE DIAMETER/15X,13X,11HOUTSIDE ROW,7X,CONO4200
     1F10.2,14X,6HINCHES/15X,14X,10HINSIDE ROW,7X,F10.2,14X,6HINCHES/)  CONO4210
230   FORMAT (1H ,15X,36HFLAG FOR PITCH VARIATION IS MPITCH =,I2)        CONO4220
240   FORMAT (1H ,15X,39HTUBE DIAMETER IS A CONSTANT THRU BUNDLE/)       CONO4230
250   FORMAT (1H ,15X,40HFLAG FOR TUBE WALL INPUT UNIT IS IWALL =,I2)    CONO4240
260   FORMAT (1H ,15X,26HTUBE CIRCUMFERENTIAL PITCH,4X,F10.2)            CONO4250
270   FORMAT (1H ,15X,26HTUBE CIRCUMFERENTIAL PITCH/15X,13X,11HOUTSIDE RCONO4260
     1OW,7X,F11.3/15X,14X,10HINSIDE ROW,7X,F11.3/)                       CONO4270
280   FORMAT (1H ,15X,29HPITCH IS CONSTANT THRU BUNDLE/)                 CONO4280
290   FORMAT (1H ,15X,19HTUBE WALL THICKNESS,11X,F13.5,11X,6HINCHES)     CONO4290
300   FORMAT (1H ,15X,31HWALL THICKNESS TO TUBE ID RATIO,F10.3)          CONO4300
310   FORMAT (1H ,15X,25HTUBE WALL SPECIFIC WEIGHT,5X,F10.2,9X,8HLB/FT**CONO4310
     13)                                                                 CONO4320
```

```
320   FORMAT (1H ,15X,25HTUBE THERMAL CONDUCTIVITY,5X,F10.2,9X,11HBTU/FTCON04330
     1-HR-F)                                                            CON04340
330   FORMAT (1H ,15X,19HTUBE FOULING FACTOR,11X,F13.5/)                CON04350
340   FORMAT (1H ,15X,20HSTEAM MASS FLOW RATE,9X,F9.0,16X,6HLBM/HR)     CON04360
350   FORMAT (1H ,15X,27HWEIGHT FRACTION OF N/C FLOW,3X,1PE13.5/)       CON04370
360   FORMAT (1H ,15X,14HN/C GAS IS AIR)                                CON04380
370   FORMAT (1H ,15X,25HN/C GAS IS CARBON DIOXIDE)                     CON04390
380   FORMAT (1H ,15X,46HN/C GAS IS A MISTURE OF AIR AND CARBON DIOXIDE)CON04400
390   FORMAT (1H ,15X,29HCOOLANT INJECTION TEMPERATURE,F10.1,16X,5HDEG FCON04410
     1)                                                                 CON04420
400   FORMAT (1H ,15X,26HENTERING STEAM TEMPERATURE,4X,F9.2,16X,5HDEG F/CON04430
     1)                                                                 CON04440
410   FORMAT (1H ,15X,28HCW FLOW SPECIFIER IS IFLOW =,I2)               CON04450
420   FORMAT (1H ,22X,22HCOOLANT MASS FLOW RATE,15X,F9.1,15X,6HLBM/HR/) CON04460
430   FORMAT (1H ,15X,16HCOOLANT VELOCITY,14X,F9.1,15X,6HFT/SEC/)       CON04470
440   FORMAT (1H ,15X,21HCOOLANT PRESSURE DROP,9X,F11.3,16X,3HPSI/)     CON04480
450   FORMAT (1H ,15X,38HNO. OF INTERNAL ENHANCEMENT REGIONS IS,I2)     CON04490
460   FORMAT (1H ,15X,38HNO. OF EXTERNAL ENHANCEMENT RETIONS IS,I2)     CON04500
470   FORMAT (1H ,15X,26HFLAG FOR BAFFLE IS IBAF = ,I5)                 CON04510
      END                                                               CON04520
C                                                                       CON04530
C*******************************************************************     CON04540
C*******************************************************************     CON04550
C                                                                       CON04560
      SUBROUTINE ORCON (ICALC)                                          CON04570
C--------------                                                         CON04580
C ORC 1                                                                 CON04590
C                    SUBROUTINE ORCON                                   CON04600
C                                                                       CON04610
C              DIMENSION SIZE                                           CON04620
C                   X = NO. OF ROWS + 1                                 CON04630
C                   Y = NO. OF SECTORS                                  CON04640
C                   Z = NO. OF ENHANCEMENT REGIONS                      CON04650
C                                                                       CON04660
C         TPAR(X)   AOTFLW(X)   RADIUS(X)   TBNPR(X)  VW(X)   SID(X)    CON04670
C         SECANG(Y)  WGAS(Y)   JBAF(Y+1)                                CON04680
C         ANT(X,Y)   WS(X,Y)    VEL(X,Y)    PMIX(X,Y)                   CON04690
C         BC(Z)  BE(Z)  ENI(Z)  END(Z)  NETI(Z)  NETE(Z)                CON04700
C         NRNE(Z)  NRNI(Z)                                              CON04710
C                                                                       CON04720
C ZZZ ORC-1                                                             CON04730
C--------------                                                         CON04740
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,CON04750
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTOCON04760
     2T,BNDRAD,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR                      CON04770
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,SCON04780
     1TSAT1,TUBESW,WNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MDCON04790
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)CON04800
```

```
      3,WSI,PFILL
       COMMON /ORO1/ AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDOT(10
      10)
       COMMON /ORC2/ ANT(100,15),STB2ES,VW(100)
       COMMON /OUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNO,VEL2,FR
       COMMON /CONST/ AMWNC,CBI,CPB,PI,SG,IFIRST
       COMMON /SEC/ ALMTD(100,15),CUMDP(100,15),DELP(15),GFLW(100,15),HEF
      1F(100,15),PMIX(100,15),PSAT(100,15),QOA(100,15),RC(100,15),ROUT(10
      20,15),SHI(100,15),SHN(100,15),STSAT(100,15),UN(100,15),VEL(100,15)
      3,VNRE(100,15),VPSHH(100,15),WCND(100,15),WGAS(15),WP(15),WS(100,15
      4),WSP(15)
       COMMON /OUT2A/ RCORR,RSPF,PHPCON,VCLIC
C
       DIMENSICN TPAR(100),RADIUS(100),SECANG(15),BAFA(16)
C
C$------------
C ORC 2
C                      DATA INITIALIZATION
C
C          AMWNC = MOLECULAR WEIGHT OF N/C GAS PRESENT
C          IFIRST IS A FLAG TO HETTRN TO INDICATE FIRST EXECUTION
C          CBI  = SALT CONCENTRATION OF COOLANT   WT. PRCNT
C          CPB  = COOLANT SPECIFIC HEAT (UNITS?)
C          PTST AND WTST ARE CONTRAINTS USED IN SEC-6
C
C          VW(1) = IS INITIALIZED AS A SEED FOR AN ITERATIVE
C             CALCULATICN IN ORC-9
C
C ZZZ ORC-2
C------------
       IF (ICALC.EQ.2) GO TO 20
       SG=32.174
       PI=3.14159
       CBI=.035
       CPB=1.
       SECFLG=FLOAT(ISEC)
       WNCI=WSI*WNCIR
       A=1.
       DO 10 I=1,ISEC
       WS(1,I)=WSI/SECFLG*A
10     WGAS(I)=WNCI/SECFLG*A
       IFIRST=1
       IF (JGAS.EQ.1) AMWNC=28.965
       IF (JGAS.EQ.2) AMWNC=44.
       IF (JGAS.EQ.3) AMWNC=36.5
       VW(1)=10.
       GO TO 20
C$------------
```

```
C ORC 3                                                                 CONO5290
C                  ENTERING POINT FOR ICALC = 2                         CONO5300
C                                                                       CONO5310
C ZZZ ORC-3                                                             CONO5320
C--------------                                                         CONO5330
20    CONTINUE                                                          CONO5340
      DO 30 I=1,ISEC                                                    CONO5350
      WTST(I)=0.                                                        CONO5360
30    PTST(I)=0.                                                        CONO5370
C$-------------                                                         CONO5380
C ORC 4                                                                 CONO5390
C                      BUNDLE GEOMETRY                                  CONO5400
C                                                                       CONO5410
C           TUBE PITCH IS DEFINED AS THE RATIO CF TUBE CENTER TO CENTECONO5420
C           SPACING TO TUBE OD, MEASURED ALONG A ROW OF CONSTANT BUNDLCONO5430
C           RADIUS.  ROW RADIAL SEPARATION IS NOT DIRECTLY RELATED TO   CONO5440
C           PITCH.                                                      CONO5450
C                                                                       CONO5460
C        AOTFLW = AREA OPEN TO STEAM FLOW IN EACH ROW I OF              CONO5470
C                  A SECTOR                                             CONO5480
C        AHOLES = AREA OF THE ENDS OF THE TUBES                         CONO5490
C        ARATIO = TOTAL AREA OF THE ENDS OF THE TUBES DIVIDED BY THE    CONO5500
C                 THE TUBE SHEET AREA THRU WHICH THE TUBES PASS         CONO5510
C        ARCPR = CIRCUMFERENTIAL ARC LENGTH OF EACH SECTOR PER UNIT     CONO5520
C                 RADIUS OF THE ROW BEING CONSIDERED                    CONO5530
C        ALST = TUBE LENGTH   FT                                        CONO5540
C        BNDRAD = TUBE BUNDLE RADIUS - FT                               CONO5550
C        CSP = TUBE CNTR - CNTR SPACING AT A GIVEN ROW                  CONO5560
C        CSPI = CSP FOR INNER ROW                                       CONO5570
C        CSPO = CSP FOR OUTER ROW                                       CONO5580
C        DELCSP = RADIAL CHANGE IN CSP PER ROW  FT                      CONO5590
C        DELODF = CHANGE IN TUBE OD / ROW  FT                           CONO5600
C        FCWID = ANGULAR WIDTH OF CONDENSER MINUS ONE SECTOR   DEG      CONO5610
C        ISEC = NUMBER OF SECTORS IN THE CONDENSER                      CONO5620
C        NOROWS = NO. OF ROWS IN EACH SECTOR                            CONO5630
C        OD = TUBE OD ON A GIVEN ROW  FT                                CONO5640
C        ODOI = TUBE OD, OUTER ROW - IN                                 CONO5650
C        ODIF = TUBE OD, INNER ROW - FT                                 CONO5660
C        ODII = TUBE OD, INNER ROW - IN                                 CONO5670
C        ODOF = TUBE OD, OUTER ROW - FT                                 CONO5680
C        PRCCLR = PERCENT OF TOTAL TUBES ASSIGNED TO COOLER             CONO5690
C        RAD = RADIUS OF A GIVEN ROW  FT                                CONO5700
C        RADINS = BUNDLE INSIDE RADIUS - FT                             CONO5710
C        RADIUS = RADIUS OF ROW I  FT                                   CONO5720
C        RCORR = A CORRECTION FACTOR INTRODUCED TC SMOOTH THE STEP DIFF CONO5730
C                BETWEEN ROWS WHEN TUBES ARE ADDED OR DELETED           CONO5740
C        ROWS = NO OF ROWS IN CONDENSER                                 CONO5750
C        RSPA = TUBE ROW RADIAL SEPARATION, IN                          CONO5760
```

```
C          RSPF = TUBE ROW RADIAL SEPARATION, FT
C          SDDI = TUBE PITCH ON INSIDE ROW
C          SDDO =  TUBE PITCH  ON  OUTSIDE ROW
C          SECA = TEMP VALUE FOR SECANG    DEG
C          SECANG = ANGLE FROM THE VERTICAL PASSING THROUGH EACH SECTOR
C          SECFLG =  THE NUMBER OF SECTORS IN THE CONDENSER
C          SECWID = SECTOR WIDTH IN DEGREES OF ARC
C          SID = TUBE ID FOR EACH ROW   -   FT
C          SIDIF = TUBE ID FOR INNER ROW IN FEET
C          SIDI = TUBE  ID FOR INNER ROW IN INCHES
C          SIDOF = TUBE ID FOR OUTER ROW IN FEET
C          SIDO = TUBE  ID FOR OUTER ROW IN INCHES
C          DELSCF = LINEAR VARIATION OF TUBE INNER DIAMETER
C          TBNPR = NO. OF TUBES IN ROW I OF EACH SECTOR
C          TNOTCT = TOTAL NO. OF TUBES IN CONDENSER + COOLER
C          TNO = NO. OF TUBES IN THE CONDENSER MODEL
C          TNOS = TOTAL NUMBER OF TUBES PER SECTOR
C          TNR = NUMBER OF TUBES IN A ROW OF A SECTCR
C          TPAR = TUBE DENSITY /UNIT AREA FOR EACH ROW  TUBES/FT**2
C          TSAREA = TUBE SHEET AREA
C          VOL1 = CONDENSOR VOLUME
C          VOL1C = STORED CONDENSER VOLUME
C          VOL2 = VOLUME ACTUALLY OCCUPIED BY THE CONDENSOR TUBE BUNDLE
C          XW1 = RATIO OF TUBE WALL THICKNESS TO TUBE OD
C          XW2 = TUBE WALL THICKNESS  INCHES
C
C                    NOTE:  ADD CALC FOR SID
C
C               NOTE: RADIAL ROW SEPARATION IS CALCULATED
C                     AS THE AVERAGE OF CSPI AND CSPO UNLESS
C                     RSPA HAS A NON-ZERO VALUE FOR INPUT.
C
C                VALUES CARRIED INTO THIS ROUTINE
C          COMMON/NEW/ - , SECWID , NOROWS , RADINS , SDDO , SDDI ,
C          COMMCN/NEW/ - , ODII , ODOI , RSP , ISEC ;
C
C ZZZ ORC-4
C---------------
      IF (MDIAM.EQ.1) SIDI=SIDO
      IF (MPITCH.EQ.1) SDDI=SDDO
      ARCPR=6.283/360.*SECWID
      TNOS=0.0
      SECFLG=FLOAT(ISEC)
C
      IF (IOPT.EQ.2) GO TO 50
C
C     COMPUTATION OF GEOMETRY WHEN IOPT = 1
C         NOROWS IS A CONSTANT
```

```
C
      AHOLES=0.
      ROWS=FLOAT(NOROWS)
      SIDOF=SIDO/12.
      SIDIF=SIDI/12.
      DELSDF=(SIDOF-SIDIF)/(ROWS-1.)
      IF (XW2.GT.0) ODOF=SIDOF+XW2/6.
      IF (XW2.GT.0) ODIF=SIDIF+XW2/6.
      IF (XW2.LE.0) ODOF=SIDOF+(2*SIDOF*XW1)
      IF (XW2.LE.0) ODIF=SIDIF+(2*SIDIF*XW1)
      ODOI=ODOF*12.
      ODII=ODIF*12.
      DELODF=(ODOF-ODIF)/(ROWS-1.)
      DELOD=DELODF*12.
      CSPO=SDDO*ODOF
      CSPI=SDDI*ODIF
      DELCSP=(CSPO-CSPI)/(ROWS-1.)
      RSPF=.866*(CSPO+CSPI)/2.
      RSP=RSPF*12.
      IF (RSPA.NE.0.) RSP=RSPA
      RSPF=RSP/12.
      BNDRAD=RADINS+RSPF*(ROWS-1.)
      DO 40 I=1,NOROWS
      RAD=BNDRAD-RSPF*FLOAT(I-1)
      CSP=CSPO-DELCSP*FLOAT(I-1)
      TPAR(I)=1./(CSP*RSPF)
      TNR=ARCPR*RAD/CSP
      OD=ODOF-DELODF*FLOAT(I-1)
      SID(I)=SIDOF-DELSDF*FLOAT(I-1)
      AOLFLW(I)=(CSP-OD)*TNR
      AOTFLW(I)=AOLFLW(I)*ALST
      TNOS=TNOS+TNR
      RADIUS(I)=RAD
      AHOLES=AHOLES+PI*(ODOF**2)*TNR/4.
40    TBNPR(I)=TNR
      TNO=TNOS*SECFLG
      AHOLES=AHOLES*SECFLG
      TNOTOT=TNO*100./(100.-PRCCLR)
      GO TO 100
C
C
50    CONTINUE
C
C     COMPUTATION OF BUNDLE GEOMETRY IF IOPT = 2
C        SIDI=SIDO
C        SDDI=SDDO
C
      ERR1=0.
```

```
      TNOC=0.
      TNOS=0.
      TNR=0.
      TDIFF=0.
      SIDOF=SIDO/12.
      SIDIF=SIDI/12.
      IF (XW2.GT.0) ODOF=SIDOF+XW2/6.
      IF (XW2.GT.0) ODIF=SIDIF+XW2/6.
      IF (XW2.LE.0) ODOF=SIDOF+(2*SIDOF*XW1)
      IF (XW2.LE.0) ODIF=SIDIF+(2*SIDIF*XW1)
      ODOI=ODOF*12.
      ODII=ODIF*12.
      DELODF=0.
      DELOD=DELODF
      CSPO=SDDO*ODOF
      CSPI=SDDI*ODIF
      DELCSP=0.
      RSPF=.866*(CSPO+CSPI)/2.
      RSP=RSPF*12.
      IF (RSPA.NE.0.) RSP=RSPA
      RSPF=RSP/12.
C
C        DETERMINE THE NUMBER OF TUBES IN THE CONDENSOR
C        ALL SECTORS ARE IDENTICAL IN GEOMETRY
C        BEGINNING WITH THE INNERMOST ROW, CALCULATE THE NUMBER
C        OF ROWS IN THE CONDENSOR
C
      I=1
      NOROWS=1
      TNO=TNOTOT*(100.-PRCCLR)/100.
      RADIUS(1)=RADINS
C
60    SID(I)=SIDOF
      TNR=(ARCPR*RADIUS(I))/CSPO
      TNOS=TNOS+TNR
C
C        CHECK TO ENSURE THAT THE NUMBER OF TUBES IN THE SECTOR
C        HAS NOT EXCEEDED THE NUMBER OF TUBES AVAILABLE
C
      IF (TNOS.LT.(TNO/SECFLG)) GO TO 70
      TNOS=TNOS-TNR
      TNOC=TNOS*SECFLG
      TDIFF=TNO-TNOC
      TBNPR(I)=TDIFF/SECFLG
      PFILL=TBNPR(I)/TNR
      GO TO 80
C
70    TPAR(I)=1./(CSPO*RSPF)
```

```
      AOLFLW(I)=(CSPO-ODOF)*TNR                                         CONO7210
      AOTFLW(I)=AOLFLW(I)*ALST                                          CONO7220
      TBNPR(I)=TNR                                                      CONO7230
      RADIUS(I+1)=RADIUS(I)+RSPF                                        CONO7240
      I=I+1                                                             CONO7250
      NOROWS=NOROWS+1                                                   CONO7260
C                                                                       CONO7270
C        CHECK TO SEE THAT THE NUMBER OF ROWS DOESNOT EXCEED 199        CONO7280
C        BECAUSE 200 IS THE LIMIT FOR THE DIMENSIONS OF NUMEROUS ARRAYS CONO7290
C                                                                       CONO7300
      IF (NOROWS.LT.100) GO TO 60                                       CONO7310
      ERR1=TNO-(TNOS*SECFLG)                                            CONO7320
      TNO=TNOS*SECFLG                                                   CONO7330
      WRITE (6,640)                                                     CONO7340
      NOROWS=NOROWS-1                                                   CONO7350
      GO TO 90                                                          CONO7360
C                                                                       CONO7370
C        IF LAST ROW IS INCOMPLETE, FIND ITS PERTINENT GEOMETRY         CONO7380
C                                                                       CONO7390
80    SDDL=(RADIUS(I)*ARCPR)/(ODOF*TBNPR(I))                            CONO7400
      CSPL=SDCL*ODOF                                                    CONO7410
      TPAR(I)=1./(CSPL*RSPF)                                            CONO7420
      AOLFLW(I)=(CSPL-ODOF)*TBNPR(I)                                    CONO7430
      AOTFLW(I)=AOLFLW(I)*ALST                                          CONO7440
      TNOS=TNCS+TBNPR(I)                                                CONO7450
90    CONTINUE                                                          CONO7460
      AHOLES=PI*(ODOF**2)*TNO/4.                                        CONO7470
      ROWS=FLCAT(NOROWS)                                                CONO7480
      BNDRAD=RADINS+RSPF*(ROWS-1.)                                      CONO7490
C                                                                       CONO7500
C                                                                       CONO7510
C        THIS NEXT SUBROUTINE SWITCHES THE ARRAYS SO THE ARRAY VALUES   CONO7520
C        START WITH THE OUTSIDE ROW AND WORK TOWARDS THE CENTER.        CONO7530
C        THIS WILL MAKE THE GEOMETRY VALUES COMPATIBLE WITH THE REST    CONO7540
C        OF THE PROGRAM.                                                CONO7550
C                                                                       CONO7560
      CALL SWITCH (RADIUS,NOROWS)                                       CONO7570
      CALL SWITCH (SID,NOROWS)                                          CONO7580
      CALL SWITCH (TBNPR,NOROWS)                                        CONO7590
      CALL SWITCH (TPAR,NOROWS)                                         CONO7600
      CALL SWITCH (AOLFLW,NOROWS)                                       CONO7610
      CALL SWITCH (AOTFLW,NOROWS)                                       CONO7620
C                                                                       CONO7630
100   CONTINUE                                                          CONO7640
C                                                                       CONO7650
      AOTFLW(NOROWS+1)=ALST*(RADINS-SDI*ODII/24.)*PI/6.                 CONO7660
C                                                                       CONO7670
      IF (ISEC.EQ.1) SECANG(1)=PHI                                      CONO7680
```

```
      IF (ISEC.EQ.1) GO TO 110
      FCWID=SECWID*(SECFLG-1.)
      DO 110 I=1,ISEC
      SECA=(PHI-FCWID/2.)+SECWID*(I-1)
      IF (SECA.GT.360.) SECA=SECA-360.
110   SECANG(I)=SECA
C
      DUMMY=ALST*PI*SECWID*SECFLG/360.
      BNDRA1=BNDRAD
      VOL1=DUMMY*BNDRA1**2
      VOL1C=VCL1
C
C     IF TNOTOT IS A DESIGN VARIABLE THEN THE INCREASE OF VOLUME  AND
C     OUTSIDE BUNDLE RADIUS MUST BE A GRADUAL PROCESS AND NOT A STEP
C     INCREASE OR DECREASE
C
      IF (IOPT.EQ.1) GO TO 120
      RCORR=RSPF
      RCORR=RCORR*((1.-PFILL))
      BNDRAD=BNDRAD-RCORR
      VOL1=DUMMY*BNDRAD**2
      VOL1C=VCL1
C
120   VOL2=DUMMY*(BNDRAD**2-RADINS**2)
C
C     COMPUTE THE AREA RATIO WHICH IS THE AREA OF TUBE HOLES DIVIDED
C     BY THE TUBE SHEET AREA
C
      TSAREA=PI*(BNDRAD**2)*SECFLG*SECWID/360.
      ARATIO=AHOLES/TSAREA
C
C$---------------
C ORC A4
C                    TEMP OUTPUT SECTION
C
C                THIS SECTION FOR DEBUGGING AID
C
C ZZZ ORC-A4
C--------------
      ORCA4=0.
      IF (ORCA4.NE.1.) GO TO 150
      WRITE (6,650)
      WRITE (6,660)
      WRITE (6,700) ODOF,ODIF,DELODF
      WRITE (6,670)
      WRITE (6,700) CSPO,CSPI,DELCSP
      WRITE (6,680)
      WRITE (6,700) RSPF,RSP,TNOS
```

```
      WRITE (6,690)
      WRITE (6,700) AHOLES,TSAREA,ARATIO
      WRITE (6,710)
      DO 130 I=1,NOROWS
130   WRITE (6,740) AOTFLW(I),RADIUS(I),TBNPR(I),TPAR(I)
      WRITE (6,720)
      DO 140 I=1,ISEC
140   WRITE (6,730) SECANG(I)
150   CONTINUE
C$-------------
C ORC 5
C                      LOCATE BAFFLES
C
C             THIS SECTION CALCULATES BAFFLE ANGULAR LOCATION
C             BASED ON THE INPUT VALUES OF IBAF AND JBAF.
C
C        BAFA = BAFFLE LOCATIONS IN RADIANS MEASURED C-C WISE FROM VERT.
C        FCWI = ONE-HALF OF THE BUNDLE ANGULAR WIDTH IN RADIANS
C        FCWID = FCWI EXPRESSED IN DEGREES OF ARC
C        IBAF = BAFFLE CONTROL VARIABLE
C             IBAF = -1 IMPLIES BAFFLES BETWEEN EACH SECTOR
C             IBAF =  0 IMPLIS NO BAFFLES (SEE NOTE)
C             IBAF =  N IMPLIES N BAFFLES
C        IBAFT = NUMBER OF BAFFLES IN CONDENSER
C             IBAFT = ISEC + 1 FOR IBAF = -1
C             IBAFT = 2 FOR IBAF = 0
C             IBAFT = N FOR IBAF = N
C        ISEC = NUMBER OF SECTORS IN THE CONDENSER
C        JBAF = SECTOR NUMBERS OF THOSE SECTORS WHOSE COUNTER-CLOCKWISE
C               SIDE HAS A BAFFLE ATTACHED.
C        PHI  = BUNDLE ORIENTATION ANGLE IN DEGREES OF ARC CLOCK-WISE
C              FROM VERTICLE.
C        PHII = PHI EXPRESSED IN RADIANS
C        SECI = SECTOR ARC WIDTH IN RADIANS
C        SECWID = SECTOR ARC WIDTH IN DEGREES OF ARC
C
C ZZZ ORC-5
C-------------
      IBAFT=IBAF
      SECI=SECWID*PI/180.
      FCWID=SECWID*SECFLG/2.
      FCWI=FCWID*PI/180.
      PHII=PHI*PI/180.
      IF (IBAF) 160,200,180
160   IBAFT=ISEC+1
      DO 170 I=1,IBAFT
170   JBAF(I)=I
180   DO 190 I=1,IBAFT
```

```
      IT=JBAF(I)                                                        CONO8650
190   BAFA(I)=PHII-FCWI+SECI*FLOAT(IT-1)                                CONO8660
      GO TO 210                                                         CONO8670
200   BAFA(1)=PHII-FCWI                                                 CONO8680
      BAFA(2)=PHII+FCWI                                                 CONO8690
      IBAFT=2                                                           CONO8700
      JBAF(1)=1                                                         CONO8710
      JBAF(2)=ISEC+1                                                    CONO8720
210   CONTINUE                                                          CONO8730
C$-------------                                                         CONO8740
C ORC A5                                                                CONO8750
C                     TEMP OUTPUT SECTION                               CONO8760
C ZZZ ORC-A5                                                            CONO8770
C-------------                                                          CONO8780
      ORCA5=0.                                                          CONO8790
      IF (ORCA5.NE.1) GO TO 240                                         CONO8800
      WRITE (6,750)                                                     CONO8810
      WRITE (6,760) IBAF,IBAFT                                          CONO8820
      WRITE (6,770)                                                     CONO8830
      DO 220 I=1,IBAFT                                                  CONO8840
220   WRITE (6,780) I,JBAF(I)                                           CONO8850
      WRITE (6,790)                                                     CONO8860
      DO 230 I=1,IBAFT                                                  CONO8870
      X=BAFA(I)*180./PI                                                 CONO8880
230   WRITE (6,800) X,BAFA(I)                                           CONO8890
240   CONTINUE                                                          CONO8900
C$-------------                                                         CONO8910
C ORC 6                                                                 CONO8920
C                     LOCATE ACTIVE BAFFLE FOR EACH SECTOR              CONO8930
C                                                                       CONO8940
C ZZZ ORC-6                                                             CONO8950
C---------                                                              CONO8960
      DO 430 IS=1,ISEC                                                  CONO8970
      SECA=SECANG(IS)*PI/180.                                           CONO8980
      SEC=PI/2.-SECA                                                    CONO8990
      IF ((SECA.GT.0.).AND.(SECA.LT.PI)) GO TC 270                      CONO9000
      DO 250 I=1,IBAFT                                                  CONO9010
      BAFAT=BAFA(I)                                                     CONO9020
      IF (BAFAT.GT.SECA) GO TO 260                                      CONO9030
250   CONTINUE                                                          CONO9040
260   BAF=PI/2.-BAFAT                                                   CONO9050
      IF ((BAFAT.GT.0.).AND.(SECA.LT.0.)) BAF=PI/2.                     CONO9060
      GO TO 300                                                         CONO9070
270   DO 280 I=1,IBAFT                                                  CONO9080
      J=IBAFT+1-I                                                       CONO9090
      BAFAT=BAFA(J)                                                     CONO9100
      IF (BAFAT.LT.SECA) GO TO 290                                      CONO9110
280   CONTINUE                                                          CONO9120
```

```
      BAF=PI/2.
      GO TO 300
290   BAF=PI/2.-BAFAT
300   CONTINUE
      IF (BAFAT.LT.0.) BAF=PI/2.
C$-------------
C ORC A6
C                   TEMP OUTPUT SECTION
C
C ZZZ ORC-A6
C-------------
      ORCA6=C.
      IF (ORCA6.NE.1.) GO TO 310
      WRITE (6,810)
      WRITE (6,820)
      X=SEC*180./PI
      WRITE (6,830) IS,X,SEC
      X=BAF*180./PI
      WRITE (6,840) X,BAF
310   CONTINUE
C$-------------
C ORC 7
C                   CALCULATE FLOODING FACTORS
C
C          AN = SUMMING RAGISTER FOR NO. OF TUBES
C          ANT = NO. OF TUBES OVER THE TARGET TUBE
C          BAF = ANGLE ABOUT HORIZONTAL OF 1ST BAFFLE ABOVE CURRENT SECTOR
C          DELODF = SEE ORC-4
C          D1 = FARTHEST DIST FROM HORIZONTAL TO CURRENT ROW INTERSECTION
C          D2 = SHORTEST  "     "        "       "     "      "
C          IR = ROW NO. OF CURRENT TARGET TUB.
C          IS = SECTOR NO. OF CURRENT TARGET TUBE
C          ITR = ROW NO. FOR 1ST ROW TO BE SUMMED IN UPPER QUADRANT
C          I1 = ROW WHOSE CONTRIBUTION TO ANT IS BEING CALCULATED
C          I2 = LOOP COUNTER IN UPPER QUADRANT CALCULATION
C          NOROWS = NO. OF ROWS IN THE CONDENSER
C          OD = TUBE OD IN ROW BEING CONSIDERED  - FT
C          ODOF = TUBE DIAMETER IN OUTERMOST (IR = 1) ROW OF TUBES
C          RADIUS = RADIUS OF EACH TUBE ROW
C          RSPF = ROW RADIAL SEPARATION  - FT
C          R1 = RADIUS OF UPPER OR LOWER BOUNDARY OF ROW I1
C          SEC = ANGLE FROM HORIZONTAL OF CURRENT SECTOR MIDPOINT
C          SECA= ANGLE FROM VERTICAL OF CURRENT SECTOR MIDPOINT
C          TEST = TEST FOR VERTICAL BAFFLES
C          TPAR = TUBE DENSITY PER UNIT AREA PFOR EACH ROW - TUBES/FT**2
C          XC = HORIZONTAL DIST OR CURRENT TUBE FROM COND CENTER
C          XCS = XC SQUARED
C          YC = VERTICAL (+/-) DIST OF CURRENT TUBE FROM COND CNTR - FT
```

```
C          ZT  = VERT DIST FROM TARGET TUBE TO BAFFLE INTERSECTION  -  FT   CON09610
C                                                                           CON09620
C ZZZ ORC-7                                                                 CON09630
C---------------                                                            CON09640
      ORCA7=0.                                                              CON09650
      IF (ORCA7.EQ.1.) WRITE (6,860)                                        CON09660
      DO 420 IR=1,NOROWS                                                    CON09670
C     IF(IS.EQ.ISEC.AND.IR.GE.20)ORCA7 = 1.                                 CON09680
      ITR=IR                                                                CON09690
      AN=0.                                                                 CON09700
      OD=ODOF-DELODF*FLOAT(IR-1)                                            CON09710
      R1=RADIUS(IR)                                                         CON09720
      YC=R1*SIN(SEC)                                                        CON09730
      XC=R1*COS(SEC)                                                        CON09740
      XCS=XC*XC                                                             CON09750
      TEST=ABS(ABS(BAF)-PI/2.)                                              CON09760
      X=OD*12.                                                              CON09770
      IF (ORCA7.EQ.1.) WRITE (6,870) IR,IS                                  CON09780
      IF (ORCA7.EQ.1.) WRITE (6,880) X,R1,YC,XC,TEST                        CON09790
      IF (TEST.GT.1E-2) GO TO 320                                           CON09800
      ZT=1E10                                                               CON09810
      GO TO 330                                                             CON09820
320   ZT=XC*TAN(BAF)                                                        CON09830
330   XC=ABS(XC)                                                            CON09840
      IF (YC.GT.0.) GO TO 370                                               CON09850
      IF (ORCA7.EQ.1.) WRITE (6,890) ZT                                     CON09860
      D1=YC                                                                 CON09870
      DO 340 I1=IR,NOROWS                                                   CON09880
      R1=RADIUS(I1)-RSPF/2.                                                 CON09890
      IF (R1.LE.XC) GO TO 350                                               CON09900
      D2=-1.*SQRT(R1*R1-XCS)                                                CON09910
      IF (ORCA7.EQ.1.) WRITE (6,900) I1,R1,D2                               CON09920
      IF (D2.GT.ZT) GO TO 360                                               CON09930
      IF (ORCA7.EQ.1.) WRITE (6,910)                                        CON09940
      AN=AN+(D2-D1)/(RSPF*2.)                                               CON09950
      IF (ORCA7.EQ.1.) WRITE (6,920) D1,D2,AN                               CON09960
      D1=D2                                                                 CON09970
340   CONTINUE                                                              CON09980
      D2=-D2                                                                CON09990
      IF (ORCA7.EQ.1.) WRITE (6,940)                                        CON10000
      IF (D2.GT.ZT) GO TO 410                                               CON10010
      ITR=NOROWS                                                            CON10020
      D1=D2                                                                 CON10030
      IF (ORCA7.EQ.1.) WRITE (6,950)                                        CON10040
      GO TO 380                                                             CON10050
350   AN=AN-D1/(RSPF*2.)                                                    CON10060
      D1=0.                                                                 CON10070
      ITR=I1                                                                CON10080
```

```
      IF (ORCA7.EQ.1.) WRITE (6,930) AN,ITR
      GO TO 380
360   AN=AN+(ZT-D1)/(RSPF*2.)
      IF (ORCA7.EQ.1.) WRITE (6,960)
      GO TO 410
370   D1=YC
      IF (ORCA7.EQ.1.) WRITE (6,970) D1
380   DO 390 I2=1,ITR
      I1=ITR-I2+1
      R1=RADIUS(I1)+RSPF/2.
      D2=SQRT(R1*R1-XCS)
      IF (ORCA7.EQ.1.) WRITE (6,980) IR,I1,R1
      IF (ORCA7.EQ.1.) WRITE (6,990) D1,D2,ZT
      IF (D2.GT.ZT) GO TO 400
      AN=AN+(D2-D1)/(RSPF*2.)
      IF (ORCA7.EQ.1.) WRITE (6,1000) AN
390   D1=D2
      GO TO 410
400   AN=AN+(ZT-D1)/(RSPF*2.)
      IF (ORCA7.EQ.1.) WRITE (6,1010)
410   NF=AN+1.0
      RNF=NF
      ANT(IR,IS)=RNF
      IF (ORCA7.EQ.1.) WRITE (6,1020) IS,IR,AN
420   CONTINUE
430   CONTINUE
      ORCA7=0.
      IF (ORCA7.NE.1.) GO TO 450
      DO 440 I=1,NOROWS
440   WRITE (6,850) I,(ANT(I,J),J=1,ISEC)
450   CONTINUE
C$-------------
C ORC 9
C                    PUMPING POWER CALCULATION
C
C        ALST = TUBE LENGTH   FT
C        A1 = COEF IN FF CALCULATION
C        B1 = EXP IN FF CALCULATION
C        CBI = COOLANT SALT CONCENTRATION IN WT PRCNT
C        CMDOT = COOLANT MASS FLOW/TUBE IN EACH ROW  LBM/SEC
C        CMTOT = TOTAL COOLANT MASS FLOW THROUGH CONDENSER LBM/SEC
C        DELWP = HEADER PRESSURE DIFFERENCE  PSI
C        FF = TUBE FRICTION FACTORS
C        HEAD = FT**2/SEC**
C        SID = STORED INNER TUBE DIAMETER
C        TID = TEMPORARY VALUE FOR TUBE INNER DIAMETER
C        SLD = STORED (LENGTH/TUBE INNER DIAMETER) RATIO
C        SLDO = SLD RATIO OF TUBES IN OUTER ROW
```

```
C          SLDI = SLD RATIO OF TUBES IN INNER ROW                         CON10570
C          SLD = THE AVERAGE OF SLDO AND SLDI                             CON10580
C          HLOSS = HEAD LOSS                                              CON10590
C          REW = REYNOLDS # OF COOLANT                                    CON10600
C          REWO = REYNOLDS # OF TUBES IN THE OUTER ROW                    CON10610
C          REWI = REYNOLDS # OF TUBES IN INNER ROW                        CON10620
C          PHP = PUMPING POWER IN HP                                      CON10630
C          PHPCCN = PUMPING POWER OF THE CONDENSER BUNDLE ONLY            CON10640
C          DELWPC = PRESSURE LOSS THRU TUBES IN PSI                       CON10650
C          GFLOW = OVERALL COOLANT FLOW IN LBM/HR                         CON10660
C          RHOW = DENSITY OF COOLANT                                      CON10670
C          XMUW = VISCOSITY OF COOLANT                                    CON10680
C          VELBI = COOLING WATER VELOCITY IN FT/SEC                       CON10690
C                                                                         CON10700
C  ZZZ ORC-9                                                              CON10710
C--------------                                                           CON10720
      ORCA9=0.                                                            CON10730
      RHOW=ROEFN(CBI,STBI)                                                CON10740
      XMUW=BMUFN(CBI,STBI)/3600.                                          CON10750
      XNUW=XMUW/RHOW                                                      CON10760
      CMTOT=0.                                                            CON10770
      IF ((GFLOW.GT.0.).OR.(VELBI.GT.0.)) GO TO 530                       CON10780
      HEAD=2.*DELWP*144.*SG/RHOW                                          CON10790
      DO 520 IR=1,NOROWS                                                  CON10800
      IF (ORCA9.EQ.1.) WRITE (6,1030) IR                                  CON10810
      ITEM=IR-1                                                           CON10820
      IF (ITEM.EQ.0) ITEM=1                                               CON10830
      VG=VW(ITEM)                                                         CON10840
      IF (ORCA9.EQ.1.) WRITE (6,1040) IR,ITEM,VG                          CON10850
      IENHAN=0                                                            CON10860
      IF (NEI.EQ.0) GO TO 480                                             CON10870
      DO 460 IT=1,NEI                                                     CON10880
      NRNI1=NRNI(IT)                                                      CON10890
      NRNI2=NRNI1+NETI(IT)-1                                              CON10900
      IF (IR.GE.NRNI1.AND.IR.LE.NRNI2) GO TO 470                          CON10910
460   CONTINUE                                                            CON10920
      GO TO 480                                                           CON10930
470   IENHAN=1                                                            CON10940
      A1=BC(IT)                                                           CON10950
      B1=BE(IT)                                                           CON10960
480   TID=SID(IR)                                                         CON10970
      SLD=ALST/TID                                                        CON10980
      IF (IR.EQ.1) SLDO=SLD                                               CON10990
      IF (IR.EQ.NOROWS) SLDI=SLD                                          CON11000
      IF (ORCA9.EQ.1.) WRITE (6,1050) TID,SLD,IR                          CON11010
490   REW=VG*TID/XNUW                                                     CON11020
      IF (ORCA9.EQ.1.) WRITE (6,1060) REW,TID,XNUW                        CON11030
      IF (IENHAN.EQ.1) GO TO 510                                          CON11040
```

```
      IF (REW.GE.51904.4) GO TO 500
      A1=.3164
      B1=-.25
      GO TO 510
500   A1=.184
      B1=-.2
510   CONTINUE
      IF (ORCA9.EQ.1.) WRITE (6,1070) A1,B1,REW
      FF=A1*REW**(B1)
      HLOSS=(1.1+FF*SLD)
      IF (ORCA9.EQ.1.) WRITE (6,1080) FF,SLD,HLOSS,HEAD
      VG1=SQRT(HEAD/HLOSS)
      TEST=ABS(VG-VG1)/VG
      IF (ORCA9.EQ.1.) WRITE (6,1090) TEST,VG,VG1
      VG=VG1
      IF (TEST.GT.0.01) GO TO 490
      IF (IR.EQ.1) REWO=REW
      IF (IR.EQ.NORDWS) REWI=REW
      CMDOT(IR)=PI*TID**2*VG*RHOW/4.
      VW(IR)=VG
520   CMTOT=SECFLG*CMDOT(IR)*TBNPR(IR)+CMTOT
      PHP=CMTOT*DELWP*144./(RHOW*550.)
      PHPCON=PHP
      DELWPC=DELWP
      GO TO 600
530   A=0.
      DO 540 IR=1,NOROWS
      TID=SIC(IR)
      IF (IR.EQ.1) SLDO=ALST/TID
      IF (IR.EQ.NORDWS) SLDI=ALST/TID
540   A=A+TID*TID*PI*TBNPR(IR)/4.
      SLD=(SLDO+SLDI)/2.
      TID=ALST/SLD
      IF (GFLOW.LE.0.) GO TO 550
      VX=GFLOW/((RHOW*A*SECFLG)*3600.)
      CMTOT=GFLOW/3600.
      GO TO 560
550   CMTOT=VELBI*RHOW*A*SECFLG
      VX=VELBI
560   REW=VX*TID/XNJW
      IF (REW.GT.51904.4) GO TO 570
      FF=.3164*REW**(-.25)
      GO TO 580
570   FF=.184*REW**(-.2)
580   DELWPC=(1.1+FF*SLD)*VX**2*RHOW/(2.*SG*144.)
      DO 590 IR=1,NOROWS
      CMDOT(IR)=PI*SID(IR)**2*VX*RHOW/4.
590   VW(IR)=VX
```

```
      REWO=VX*SID(1)/XNUW
      REWI=VX*SID(NDROWS)/XNUW
      HEAD=2.*DELWPƆ*144.*SG/RHOW
      PHP=CMTCT*DELWPC*144./(RHOW*550.)
      PHPCON=PHP
      DELWP=DELWPC
600   CONTINUE
      HFG=HFGFN(STSAT1)
      STB2ES=(WSI*HFG)/(CPB*CMTOT*3600.)+STBI
C$-------------
C ORC A9
C                    TEMP OUTPUT SECTION
C ZZZ ORC-A9
C--------------
      ORCA9=0.
      IF (ORCA9.NE.1.) GO TO 630
      WRITE (6,1100)
      WRITE (6,1110) RHOW
      WRITE (6,1120) XMUW
      WRITE (6,1130) XNUW
      WRITE (6,1140) HEAD
      WRITE (6,1150) REW,FF
      WRITE (6,1160) DELWP
      WRITE (6,1170) SLDI,SLDO
      WRITE (6,1180) REWI,REWO
      WRITE (6,1190) CMTOT
      WRITE (6,1200) PHP
      WRITE (6,1210)
      DO 610 I=1,NOROWS
610   WRITE (6,1220) I,VW(I)
      DO 620 I=1,NOROWS
620   WRITE (6,1220) I,CMDOT(I)
630   CONTINLE
C$-------------
C ORC 10
C                    CALL TO SECALC
C ZZZ ORC-10
C--------------
      CALL SECALC (ARCPR,DELODF,ODIF,ODOF)
      RETURN
640   FORMAT (1H0,46HTHE NUMBER OF ROWS EXCEEDS THE MAXIMUM ALLOWED)
650   FORMAT (1H1,24HOUTPUT SECTION FOR ORC-4)
660   FORMAT (1H0,10HODOF      ,10HODIF      ,10HCELODF    )
670   FORMAT (1H0,10HCSPO      ,10HCSPI      ,10HCELCSP    )
680   FORMAT (1H0,10HRSPF      ,10HRSP       ,10HTNOS      )
690   FORMAT (1H0,10HAHOLES    ,10HTSAREA    ,10HARATIO    )
700   FORMAT (1H ,3F10.5)
710   FORMAT (1H0,10HAOTFLW    ,10HRADIUS    ,10HTBNPR     ,10HTPAR
```

```
     1 )                                                                   CON12010
720   FORMAT (1H0,10HSECANG     )                                          CON12020
730   FORMAT (1H ,F10.6)                                                   CON12030
740   FORMAT (1H ,4F10.5)                                                  CON12040
750   FORMAT (1H1,24HOUTPUT FOR SECTION ORC-5)                             CON12050
760   FORMAT (1H0,7HIBAF = ,I4,5X,8HIBAFT = ,I4)                           CON12060
770   FORMAT (1H0,22HFOR I =  ;  JBAF(I) = )                               CON12070
780   FORMAT (1H ,3X,I4,8X,I4)                                             CON12080
790   FORMAT (1H0,36HBAFFLE ANGLES IN DEGREES AND RADIANS)                 CON12090
800   FORMAT (1H ,3X,F6.1,3X,F6.4)                                         CON12100
810   FORMAT (1H1,24HOUTPUT FOR SECTION ORC-6)                             CON12110
820   FORMAT (1H0,50HANGLES ARE MEASURED CNTR-CLCK WISE FROM HORIZONTAL)CON12120
830   FORMAT (1H0,16HCURRENT SECTOR =,I3,16H  ANGLE IN DEG =,F6.1,21H  ACON12130
     1NGLE  IN RADIANS =,F6.4)                                             CON12140
840   FORMAT (1H0,32HANGLE OF ACTIVE BAFFLE IN DEG. =,F6.1,14H  IN RADIACON12150
     1NS =,F6.4)                                                           CON12160
850   FORMAT (1H ,I3,2X,6(F5.2,2X))                                        CON12170
860   FORMAT (1H0,28H***   OUTPUT FOR ORC-7   ***)                         CON12180
870   FORMAT (1H0,5HROW =,I4,11H  SECTOR = ,I4)                            CON12190
880   FORMAT (1H ,4HOD =,F6.4,10H  RADIUS =,F6.2,6H  YC =,F6.2,6H  XC =,CON12200
     1F6.2,8H  TEST =,E10.3)                                               CON12210
890   FORMAT (1H0,36HTARGET TUBE IN LOWER QUADRANT,  ZT =,F20.4)           CON12220
900   FORMAT (1H ,15HLWR QUAD - ROW=,I3,5H  R1=,F7.3,5H  D2=,F7.3)         CON12230
910   FORMAT (1H ,18HBAFFLE WAS NOT HIT)                                   CON12240
920   FORMAT (1H ,15HSTILL LWR - D1=,F7.3,5H  D2=,F7.3,5H  AN=,F6.2)       CON12250
930   FORMAT (1H ,34HPASSING INTO UPPER QUADRANT - AN =,F6.2,15H  CURRENCON12260
     1T ROW =,I4)                                                          CON12270
940   FORMAT (1H ,30HMOVING THRU VOID TO UPPER QUAD)                       CON12280
950   FORMAT (1H ,22HBAFFLE NOT HIT IN VOID)                               CON12290
960   FORMAT (1H ,28HBAFFLE HIT IN LOWER QUADRANT)                         CON12300
970   FORMAT (1H ,35HTARGET TUBE IS IN UPPER QUAD - D1 =,F8.4)             CON12310
980   FORMAT (1H ,37HCALCULATING IN UPPER QUAD - TRGET RW=,I4,12H  CALC CON12320
     1ROW =,I4,11H  CORR RAD=,F8.4)                                        CON12330
990   FORMAT (1H ,4HD1 =,F6.3,6H  D2 =,F6.3,6H  ZI =,E10.3)                CON12340
1000  FORMAT (1H ,34HUPPER QUAD - BAFFLE NOT HIT - AN =,F6.2)              CON12350
1010  FORMAT (1H ,23HUPPER QUAD - BAFFLE HIT)                              CON12360
1020  FORMAT (1H0,26H** FINAL FOR TARGET ** IS=,I4,4H IR=,I4,5H ANT=,F8.CON12370
     12)                                                                   CON12380
1030  FORMAT (1H ,16HAT DO LOOP - IR=,I4)                                  CON12390
1040  FORMAT (1H ,9HTEMP  IR=,I4,7H  ITEM=,I4,5H  VG=,E10.3)               CON12400
1050  FORMAT (1H ,16HABOVE 380 - TID=,E10.3,6H  SLD=,E10.3,5H  IR=,I4)   CON12410
1060  FORMAT (1H ,16HBELOW 380 - REW=,E10.3,6H  TID=,E10.3,7H  XNUW=,E10CON12420
     1.3)                                                                  CON12430
1070  FORMAT (1H ,12HAT 400 - A1=,E10.3,5H  B1=,E10.3,6H  REW=,E10.3)      CON12440
1080  FORMAT (1H ,3HFF=,E10.3,5H SLD=,E10.3,7H HLCSS=,E10.3,6H HEAD=,E10CON12450
     1.3)                                                                  CON12460
1090  FORMAT (1H ,5HTEST=,E10.3,5H  VG=,E10.3,6H  VG1=,E10.3)              CON12470
1100  FORMAT (1H1,16HOUTPUT FOR ORC-9)                                     CON12480
```

```
1110  FORMAT (1H0,8H  RHO = ,E10.3,10H LBM/FT**3)                        CON12490
1120  FORMAT (1H ,8H  MU  = ,E10.3,11H LBM/SEC-FT)                       CON12500
1130  FORMAT (1H ,8H  NU  = ,E10.3,10H FT**2/SEC)                        CON12510
1140  FORMAT (1H ,8H  HEAD= ,E10.3,13H FT**2/SEC**2)                     CON12520
1150  FORMAT (1H ,8H  REW = ,E10.3,3X,6HFF  = ,E10.3)                    CON12530
1160  FORMAT (1H ,8HDELWP = ,E10.3,4H PSI)                               CON12540
1170  FORMAT (1H ,39HLENGTH TO DIAMETER RATIO - INSIDE ROW =,F6.2,16H    CON12550
     1OUTSIDE ROW =,F6.2)                                                CON12560
1180  FORMAT (1H ,27HREYNOLDS NO. - INSIDE ROW =,E10.3,15H  OUTSIDE ROW  CON12570
     1=,E10.3)                                                           CON12580
1190  FORMAT (1H ,24HCOOLANT MASS FLOW RATE =,F15.3,8H LBM/SEC)          CON12590
1200  FORMAT (1H ,24HPUMPING POWER REQUIRED =,F10.2,3H HP)               CON12600
1210  FORMAT (1H ,37HCOOLANT VELOCITY IN EACH ROW - FT/SEC)              CON12610
1220  FORMAT (1H ,I4,3X,F10.3)                                           CON12620
      END                                                                CON12630
C                                                                        CON12640
C************************************************************************CON12650
C************************************************************************CON12660
C                                                                        CON12670
      SUBROUTINE SECALC (ARCPR,DELODF,ODIF,ODCF)                         CON12680
C--------------                                                          CON12690
C SEC 1                                                                  CON12700
C                     SUBROUTINE SECALC                                  CON12710
C                                                                        CON12720
C               SUBROUTINES CALLED BY SECALC                             CON12730
C          HETTRN                                                        CON12740
C                                                                        CON12750
C               FUNCTION ROUTINES CALLED BY SECALC                       CON12760
C        , TSATFN , VGFN , HFGFN , CPAFN ,                               CON12770
C                                                                        CON12780
C ZZZ SEC-1                                                              CON12790
C--------------                                                          CON12800
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,CON12810
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTOCON12820
     2T,BNDRAC,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR                       CON12830
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,SCON12840
     1TSAT1,TUBESW,NNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MDCON12850
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)CON12860
     3,WSI,PFILL                                                         CON12870
      COMMON /ORC1/ AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDOT(10CON12880
     10)                                                                 CON12890
      COMMON /OUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNG,VEL2,FR               CON12900
      COMMON /OUTC/ EXITFC,EXITFR,HTCLR,ACC,SUMQC,SDIC,SDOC,UBARW,SMWBC,CON12910
     1SMTBIC,SMTB2C,VWBAR,STSATX,PMIX1,ACT,BUNWT,AVTBI,AVTB2,WTSTC,VOLC  CON12920
      COMMON /SEC/ ALMTD(100,15),CUMDP(100,15),DELP(15),GFLW(100,15),HEFCON12930
     1F(100,15),PMIX(100,15),PSAT(100,15),QOA(100,15),RC(100,15),ROUT(10CON12940
     20,15),SHI(100,15),SHN(100,15),STSAT(100,15),UN(100,15),VEL(100,15)CON12950
     3,VNRE(100,15),VPSHH(100,15),WCND(100,15),WGAS(15),WP(15),WS(100,15CON12960
```

```
     4), WSP(15)
      COMMON /CONST/  AMWNC,CBI,CPB,PI,SG,IFIRST
      COMMON /COOL1/  VELC(100),PMIXC(100),PSATC(100),STSATC(100),WSC(100
     1), ALMTDC(100),SHIC(100),SHNFC(100),RCC(100),ROUTC(100),VNREC(100),
     2VPSHHC(100),HEFFC(100),UNC(100),WCNDC(100),GFLOWC(100),QOAC(100),C
     3UMDPC(100)
      COMMON /COOL/  IVNOC,HNOC,TNOC,TSATEX,PMXEXT,AMLSEX,WSEXIT,VELEXT,T
     1B2C,ENHIC,ENHOC,ENHFC,SHWINC,WBC
      DIMENSICN C(15)
C$------------
C SEC 2
C                INITIALIZE :OLD ORC 8,9
C                            PART - 1
C
C        ALST = TUBE LENGTH  FT
C        AMOLNC = TOTAL MOLES OF N/C GAS ENTERING CONDENSER
C        AMOLSS =  MOLES OF STM  ENTERING COND
C        AMOLST = SUM OF MOLES OF STM AND N/C  ENTERING COND
C        AMWNC = MOLECULAR WEIGHT OF N/C
C        ANT = TOTAL NO OF TUBE IN 360 DEG COND
C        ARAT = RATIO OF OUTSIDE ROW MIN FLOW AREA TO SECTOR FACE AREA
C        ARCPR = CIRCUMFERENTIAL ARC LENGTH PER UNIT RADIUS
C        AOTFLW = AREA OPEN TO FLOW IN A ROW OF TUBES IN A SECTOR FT**2
C        BNDRAD = TUBE BUNDLE OD  FT
C        DELODF = CHANGE IN TUBE OD PER ROW  FT
C        DELP12 = ENTRANCE STM PRESS LOSS  PSI
C        DELP23 = STM PRESS LOSS AS STM IS ACCEL INTO FIRST ROW  PSI
C        G2 =
C        HFG = STEAM LATENT HEAT  OF VAPORIZATION  BTU/LBM
C        NOROWS = NO. OF ROWS IN THE CONDENSER
C        ODOF = TUBE OD, OUTSIDE ROW  FT
C        ODIF = TUBE OD, INSIDE ROW   FT
C        ODOI = TUBE OD, OUTSIDE ROW  IN
C        ODII = TUBE OD, INSIDE ROW   IN
C        PMIX1 = PRESS OF STM - N/C MIXTURE ENERING COND  PSIA
C        PSAT1 = ENTERING STEAM SAT PRESS PSIA
C        PTLIM = LOWEST SATURATION PRESSURE THE STEAM CAN GO BEFORE THE
C          CORRESPONDING SATURATION TEMPERATURE FALLS BELOW INLET
C          COOLANT TEMPERATURE. THE RESULT IS 0 HEAT TRANSFER
C        RADINS = BUNDLE VOID RADIUS  FT
C        ROWS = NO OF ROWS IN THE CONDENSER
C        SECFAR = FACE AREA FOR ONE SECTOR
C        SECFLG = NO. OF SECTORS IN THE CONDENSER MODEL
C        SMTBI = ROW BY ROW SUM OF (COOLANT INLET TEMP * FLOW RATE)
C        SMWB = ROW BY ROW SUM OF COOLANT FLOW
C        SMTB2 = ROW BY ROW SUM OF (COOLANT OUTLET TEMP * FLOW RATE)
C        SUMQ = ROW BY ROW SUM OF HEAT DUTY ON TUBES
C        SG = GRAVITY FACTOR  32.2
```

```
C         STSAT1 = ENTERING STM SAT TEMP   F
C         SVMIX1 = ENTERING MIXTURE SPECIFIC VOL   FT**3/LBM
C         SVNC1 = ENTERING N/C GAS SPECIFIC VOL  FT**3/LBM
C         TSAT1 = ENTERING STEAM SATURATION TEMP   R
C         WGAS = N/C FLOW TO EACH SECTOR   LBM/HR
C         WNCI = N/C GAS FLOW TO CONDENSER LBM/HR
C         WNCIR = RATIO OF MASS FLOW OF N/C TO MASS FLOW OF STEAM
C         WS = STEAM FLOW TO EACH SECTOR   LBM/HR
C         WSI = TOTAL STM FLOW TO CONDENSER LBM/HR
C         VEL1 = MIXTURE VELOCITY PRIOR TO ENTERING FIRST ROW   FT/SEC
C         VEL2 = MIXTURE VELOCITY IN FIRST ROW OF TUBES   FT/SEC
C         IVNOC = THE NUMBER OF VERTICAL TUBES IN THE COOLER
C         IHNOC = THE NUMBER OF HORIZONTAL TUBES IN THE COOLER
C         HTCLR = THE HEIGHT OF THE COOLER
C         VG1 = ENTERING STM SPECIFIC VOL   FT**3/LBM
C
C ZZZ SEC-2
C---------------
      ROWS=FLOAT(NOROWS)
      TNOC=0.
      IVNOC=1
      IF (PRCCLR.EQ.0.0) GO TO 10
C
C    COMPUTE THE NUMBER OF VERTICAL AND HORIZONTAL ROWS IN THE COOLER
C    IF THERE IS ONE
C
      SDDC=SDDI
      TNOC=(PRCCLR/(100.-PRCCLR))*TNO
      HTCLR=BNDRAD-RADINS
      VNOC=(HTCLR/(SDDC*ODIF*0.886))+1.
      IHNOC=TNOC/VNOC
      HNOC=FLOAT(IHNOC)
      VNOC=TNOC/HNOC
      IVNOC=VNOC+.5
      VNOC=FLOAT(IVNOC)
C
10    KSTOP=NCROWS+1
      SECFLG=FLOAT(ISEC)
      WNCI=WSI*WNCIR
      IPLOOP=0
      WSAVE=(WSI+WNCI)/SECFLG
C
C
C                INITIALIZE TEST VARIABLES TO 0
C
20    CONTINUE
      STMSUM=0.
      WTSAV=0.
```

```
      STBIR=STBI+459.69
      PTLIM=PSATFN(STBIR)
      TSAT1=STSAT1+459.69
      PSAT1=PSATFN(TSAT1)
      AMOLNC=WNCI/AMWNC
      HFG=HFGFN(STSAT1)
      AMOLSS=WSI/18.015
      AMOLST=AMOLSS+AMOLNC
      PMIX1=PSAT1*(AMOLST/AMOLSS)
      VG1=VGFN(TSAT1,PMIX1)
      SVNC1=10.729*TSAT1/(AMWNC*PMIX1)
      SVMIX1=1.0/(AMOLSS/(AMOLST*VG1)+AMOLNC/(SVNC1*AMOLST))
      G2=288.0*SG*SVMIX1
      SMTBI=0.0
      SMWB=0.0
      WB=0.
      SMTB2=0.0
      SUMQ=0.0
      CUMDRY=0.
      SUBFLO=0.
      NBAD=0
      WCNDAV=0.
      SUBAVE=0.
C
C
C     COMPUTE THE PRESSURE LOSSES AT THE ENTRANCE OF THE SECTOR
C
      SECFAR=ARCPR*(BNDRAD+ODOF/2.)*ALST
      ARAT=AOTFLW(1)/SECFAR
      VEL1=(WSI+WNCI)*SVMIX1/(3600.0*SECFAR*SECFLG)
      VEL2=VEL1/ARAT
      IF (ARAT-0.715) 30,30,40
30    DELP12=0.4*(1.25-ARAT)*VEL2**2/G2
      GO TO 50
40    DELP12=0.75*(1.0-ARAT)*VEL2**2/G2
50    DELP23=(VEL2**2-VEL1**2)/G2
C$--------------
C SEC A2
C                  TEMP OUTPUT SECTION
C
C ZZZ SEC-A2
C---------------
      SECA2=0.
      IF (SECA2.EQ.0.) GO TO 70
      DO 60 I=1,ISEC
      WRITE (6,610)
      WRITE (6,620)
      WRITE (6,700) WSI,WNCI
```

```
      WRITE (6,630)
      WRITE (6,700) WS(1,I),WGAS(I)
      WRITE (6,640)
      WRITE (6,700) HFG,AOTFLW(I)
      WRITE (6,650)
      WRITE (6,700) TSAT1,STSAT1
      WRITE (6,660)
      WRITE (6,700) PSAT1,PMIX1
      WRITE (6,670)
      WRITE (6,710) AMOLNC,AMOLSS,AMOLST
      WRITE (6,680)
      WRITE (6,700) SVNC1,SVMIX1
      WRITE (6,690)
      WRITE (6,700) VG1,G2
      WRITE (6,720)
      WRITE (6,750) SECFAR,ARAT
      WRITE (6,730)
      WRITE (6,750) VEL1,VEL2
      WRITE (6,740)
60    WRITE (6,750) DELP12,DELP23
70    CONTINUE
C$-------------
C SEC 3
C                DO LOOP ON SECTORS - HEREAFTER REFERRED TO AS
C                                MAIN SECTOR LOOP
C
C            PROGRAM ENTERS SEC-3 FROM SEC-2 AND ALWAYS EXITS
C            TO SEC-4.
C
C            THIS SECTION STARTS THE J LOOP AND ESTABLISHES
C            SOME OF THE INITIAL CONDITIONS FOR THE FIRST
C            ROW OF THE CURRENT SECTOR CALCULATION.
C
C            SUBSRIPTED QUANTITIES ARE SET UP AS
C              VARIABLE (SUB1,SUB2) WHERE SUB1 IS
C              THE ROW NUMBER, AND SUB2 IS THE SECTOR
C              NUMBER.
C
C        PMIX = PRESSURE OF STM-N/C MIXTURE
C        PMIX1 = PRESSURE OF STM-N/C MIXTURE ENTERING SECTOR
C        J=DUMMY VARIABLE REPRESENTING SECTOR NUMBER
C        SECFLG = NO OF SECTORS IN THE CONDENSER
C        DELP = PRESSURE DROP ACROSS A SECTOR
C        DELP12 = ENTRANCE PRESSURE LOSS IN THE CCNDENSER
C        DELP23 = PRESSURE LOSS AT BUNDLE DUE TO VELOCITY CHANGE
C        AMOLSS = MOLES OF STEAM ENTERING THE CONCENSER
C        ANOLST = TOTAL MOLES OF STEAM AND N/C ENTERING CONDENSER
C        AMOLNC = MOLES OF N/C GASSES ENTERING THE CONDENSER
```

```
C         PSAT = SATURATED STEAM PRESS
C         PAL = SAME AS PSAT
C         PTLIM = LOWEST SATURATION PRESSURE THE STEAM CAN GO BEFORE THE
C           CORRESPONDING SATURATION TEMPERATURE FALLS BELOW INLET
C           COOLANT TEMPERATURE. THE RESULT IS 0 HEAT TRANSFER
C         TAL = SAT STEAM TEMP IN RANKIN
C         STSAT = STAT STEAM TEMP F
C         VG = SPECIFIC VOLUME OF STEAM FT**3/LB
C         AMWNC = MOLECULAR WEIGHT OF N/C GASSES
C         VNC = SPECIFIC VOLUME OF N/C GAS FT**3/LB
C         VMIX = SPECIFIC VOLUME OF STEAM AND N/C GASSES IN MIXTURE
C         WS = STEAM FLOW TO A ROW OF TUBES IN A SECTOR LB/HR
C         WGAS = FLOW OF N/C TO A SECTOR LB/HR
C         WNC = SAME AS WGAS
C         AOTFLW = AREA OPEN TO FLOW IN A BANK OF TUBES
C         VEL = STEAM-N/C MIXTURE VELOCITY IN A ROW OF TUBES
C
C ZZZ SEC-3
C--------------
      DO 390 J=1,ISEC
C     INITIALIZE SECTOR TEST VARIABLES
C
      WTST(J)=0.
      PTST(J)=0.
      CHECK1=0.
      CHECK2=0.
      SUBRAT=0.
      TUBDRY=0.
      IIII=0
      DELP(J)=0.
      DELPP=0.
      DELPTP=0.
      PMIX(1,J)=PMIX1-DELP12-DELP23
      PSAT(1,J)=PMIX(1,J)*(AMOLSS/AMOLST)
C
C     CHECK TO SEE IF SATURATION PRESSURE INTC THE SECTOR IS BELOW THE
C     SATURATION PRESSURE CORRESPONDING TO THE COOLANT INLET TEMPERATURE
C     (PTLIM). STEAM SATURATION PRESSURE CAN ONLY FALL TOO LOW DUE TO
C     EXCESSIVE ENTRANCE PRESSURE LOSSES (DELP12,DELP23)
C
      IF (PSAT(1,J).GT.PTLIM) GO TO 90
C
C     IF PSAT<PTLIM THEN DUMMY VALUES MUST BE INSERTED FOR PSAT,PMIX
C     AND DELPP IN ORDER TO KEEP THE PROGRAM RUNNING. A VALUE FOR PTST
C     IS ALSO DETERMINED.
C
      PTST(J)=PSAT(1,J)-PTLIM-NOROWS
      RTG=FLOAT(NOROWS+1)
```

```
      NRWSP3=NOROWS+1                                                   CON15370
      DELPP=PTLIM/(RTG+1.)                                              CON15380
      CHECK2=-2.                                                        CON15390
      DO 80 M=1,NRWSP3                                                  CON15400
      PSAT(M,J)=PTLIM-(M-1)*DELPP                                       CON15410
80    CONTINUE                                                          CON15420
      PMIX(1,J)=PSAT(1,J)/(AMOLSS/AMOLST)                               CON15430
C                                                                       CON15440
90    PAL=PSAT(1,J)                                                     CON15450
      TAL=TSATFN(PAL)                                                   CON15460
      STSAT(1,J)=TAL-459.69                                             CON15470
      VG=VGFN(TAL,PMIX(1,J))                                            CON15480
      VNC=(10.729*TAL)/(AMWNC*PMIX(1,J))                                CON15490
      VMIX=1.0/((AMOLSS/(VG*AMOLST))+(AMOLNC/(VNC*AMOLST)))             CON15500
      VEL(1,J)=(WS(1,J)+WGAS(J))*VMIX/(3600.0*AOTFLW(1))                CON15510
      WNC=WGAS(J)                                                       CON15520
      AMOLSC=WNC/AMWNC                                                  CON15530
C$-------------                                                         CON15540
C SEC A3                                                                CON15550
C                   TEMP OUTPUT SECTION                                 CON15560
C                                                                       CON15570
C ZZZ SEC-A3                                                            CON15580
C-------------                                                          CON15590
      SECA3=0.                                                          CON15600
      IF (SECA3.NE.1.) GO TO 100                                        CON15610
      WRITE (6,760) J                                                   CON15620
      WRITE (6,770) PMIX(1,J),PSAT(1,J)                                 CON15630
      WRITE (6,780) STSAT(1,J),TAL                                      CON15640
      WRITE (6,790) VG,VNC,VMIX                                         CON15650
      WRITE (6,800) VEL(1,J)                                            CON15660
100   CONTINUE                                                          CON15670
C$-------------                                                         CON15680
C SEC 4                                                                 CON15690
C                   DO LOOP ON ROWS - HEREAFTER REFERRED TO AS          CON15700
C                              MAIN ROW LOOP                            CON15710
C                                                                       CON15720
C            THIS SECTION IS INSIDE THE MAIN SECTOR LOOP.  IT           CON15730
C            INITIALIZES SOME VARIABLES PRICR TO CALLING HETTRN         CON15740
C            IN SEC-5.                                                  CON15750
C                                                                       CON15760
C        I = ROW NUMBER                                                 CON15770
C        IL = ROW NUMBER DUMMY VARIABLE                                 CON15780
C        AXO = AREA OPEN TO FLOW IN A ROW OF TUBES  FT**2               CON15790
C        AOTFLW = AME AS AXO                                            CON15800
C        L = SECTCR NUMBER                                              CON15810
C        J = SECTOR NUMBER DUMMY VARIABLE                               CON15820
C        NOROWS = NUMBER OF ROWS OF TUBES IN A SECTOR                   CON15830
C        DELODF = CHANGE IN TUBE OD / ROW  FT                           CON15840
```

```
C          OD = TUBE OD    FT
C          ODOF = TUBE CD, OUTSIDE ROW   FT
C          XW = TUBE WALL THICKNESS    FT
C          SDW = LOG MEAN TUBE THICKNESS   FT
C          AXI = TUBE INNER CROSS-SECTIONAL AREA  FT**2
C          AI = TUBE INNER SURFACE AREA  FT**2
C          AO = TUBE OUTER SURFACE AREA  FT**2
C          AW = LOG MEAN TUBE SURFACE AREA  FT**2
C          SHWINV = INVERSE OF WALL HEAT X-FER COEF  HR-FT**2-F/BTU
C          PTLIM = LOWEST SATURATION PRESSURE THE STEAM CAN GO BEFORE THE
C            CORRESPONDING SATURATION TEMPERATURE FALLS BELOW INLET
C            COOLANT TEMPERATURE. THE RESULT IS 0 HEAT TRANSFER
C          SKW = TUBE THERMAL CONDUCTIVITY    BTU-FT/HR-FT**2-F
C          TID = TUBE INNER DIAMETER  FT
C          WST = STEAM FLOW TO A ROW IN A SECTOR  LBM/HR
C
C ZZZ SEC-4
C---------------
      DO 380 IL=1,NJROWS
      I=IL
      L=J
      IJX=0
      LQ=0
      WB=CMDOT(I)*3600.
      AXO=AOTFLW(I)
      OD=ODOF-DELODF*FLOAT(IL-1)
      TID=SID(IL)
      XW=(OD-TID)/2.
      AI=ALST*PI*TID
      AO=ALST*PI*OD
      AXI=PI*(TID/2.)**2
      SDW=(OD-TID)/ALOG(OD/TID)
      AW=ALST*PI*SDW
      SHWINV=(AO*XW)/(SKW*AW)
      WST=WS(IL,J)
C*****
      IF (WST.LE.0) WRITE (6,600) J,IL
C$-------------
C SEC A4
C                     TEMP OUTPUT SECTION
C
C ZZZ SEC-A4
C---------------
      SECA4=0.
      IF (SECA4.NE.1) GO TO 110
      WRITE (6,810) IL
      WRITE (6,820) AI,AO
      WRITE (6,830) TID,AXI
```

```
      WRITE (6,840) SDW,AW
      WRITE (6,850) SHWINV
110   CONTINUE
C$-------------
C SEC 5
C                    CALL TO HETTRN AND PRSDRP
C
C               THIS SECTION INSIDE BOTH MAIN ROW AND SECTOR LOOPS
C
C
C         STSAT = STEAM SATURATION TEMP  F
C         ANT = TARGET TUBE LOCATION IN VERTICAL RCW
C         WS = STEAM FLOW TO A ROW OF A SECTOR   LBM/HR
C         WNC = N/C FLOW TO A ROW OF A SECTOR   LBM/HR
C         UN = OVERALL HEAT TRANSFER COEF FOR CNE ROW OF TUBES IN A SECT.
C         ALMTD = LMTD FOR A ROW IN A SECTOR  F
C         HOMCI = STSAT - COOLANT INLET TEMP
C         HOMCO = STSAT - COOLANT OUTLET TEMP
C         STFO = AVE TEMP OF OUTER TUBE FILM
C         SHI = INTERNAL TUBE FILM HEAT TRANSFER COEF.
C         SHN = EXTERNAL TUBE FILM HEAT TRANSFER COEF
C         RC = RATIO OF N/C GAS TO STM AT ONE ROW OF TUBES
C         ROUT = EXTERNAL HEAT TRANSFER COEF
C         AXO = AREA OPEN TO FLOW IN A ROW OF TUBES  FT**2
C         VNRE = REYNOLDS NUMBER ON VAPOR SIDE AT A ROW IN A SECTOR
C         DUMMY = NUSSELT NUMBER
C         VPSHH = OUTLET COOLANT TEMP  F
C         HEFF = N/C GAS FILM HEAT TRANSFER COEF
C         L = SECTOR NUMBER
C         HFG = STEAM LATENT HEAT OF VAPORIZATION  BTU/LBM
C         WCND = CONDENSATE FROM ONE ROW OF TUBES IN A SECTOR  LBM/HR
C         AO = OUTER SURFACE AREA OF A TUBE  FT**2
C         PTLIM = LOWEST SATURATION PRESSURE THE STEAM CAN GO BEFORE THE
C           CORRESPONDING SATURATION TEMPERATURE FALLS BELOW INLET
C           COOLANT TEMPERATURE. THE RESULT IS 0 HEAT TRANSFER
C         TBNPR = NUMBER OF TUBES IN A GIVEN RCW OF ASECTOR
C         TAL = SATURATED STEAM TEMP  R
C         VMIX = SPECIFIC VOLUME OF STM-N/C MIXTURE  FT**3/LBM
C         SDO = TUBE OD  FT
C         VPSH = FRICTION FACTOR USED IN PRESSURE DROP CALC
C         DELPTP = PRESS DROP ACROSS A ROW OF TUBES  PSI
C         ENHF = PRESSURE DROP FRICTION FACTOR ENHANCEMENT
C         ENHF = PRESS DROP FRICTION FACTOR ENHANCEMENT FACTOR, STM SIDE.
C
C ZZZ SEC-5
C-------------
C
C     CHECK TO ENSURE THAT THE SATURATION PRESSURE IN AN EARLIER ROW
```

```
C       CALCULATION HAS NOT DROPPED BELOW THE SATURATION PRESSURE
C       CORRESPCNDING TO THAT OF THE INLET COOLANT TEMPERATURE (PTLIM) FOR
C       A GIVEN SECTOR. IF IT HAS, CHECK2<0 AND WE SKIP HETTRN CALCS SINCE
C       DUMMY VARIABLES HAVE BEEN INSERTED FOR PSAT
C
        IF (CHECK2.LT.0) GO TO 120
C
        CALL HETTRN (ALST,AXO,I,L,OD,WB,SHWINV,STSAT(I,L),TID,WNC,WST,ALMT
       1D(I,L),ENHF,HEFF(I,L),HOMCI,HOMCO,RC(I,L),RCUT(I,L),SHI(I,L),SHN(I
       2,L),STFO,VPSHH(I,L),UN(I,L),VNRE(I,L),ENHI,ENHO,SDDU)
C
C       CHECK TO SEE IF THE FIRST ROW IN THE SECTOR HAS STEAM SATURATION
C       HAS DROPPED BELOW PTLIM. IF IT HAS THEN CHECK2<0 AND I=1.
C       DUMMY VALUES MUST THEN BE PLACED IN THOSE VARIABLES NORMALLY
C       RETURNED BY HETTRN SINCE HETTRN HAS BEEN BYPASSED.
C
120     IF ((CHECK2.GE.0.).OR.(I.NE.1)) GO TO 130
C
C       SINCE THE SATURATION STEAM PRESSURE IS BELOW PTLIM THEN THE STEAM
C       TEMPERATURE IS BELOW COOLANT INLET TEMPERATURE. THIS MEANS NO
C       HEAT IS REMOVED FROM THE STEAM AND THE ALMTD IS 0. THE COOLANT
C       OUTLET TEMP IS THE SAME AS THE COOLANT INLET TEMP AND THE TUBE
C       FILM TEMP IS THE SAME AS COOLANT INLET TEMPS. THE REMAINING
C       VARIABLE VALUES ARE APPROXIMATED.
C
        ALMTD(1,J)=0.
        HOMCI=0.
        HOMCO=0.
        VPSHH(1,J)=STBI
        HEFF(1,J)=1.E+8
        RC(1,J)=.161E-3
        VNRE(1,J)=.9E+5
        SHI(1,J)=.136E+4
        SHN(1,J)=.136E+5
        UN(1,J)=655.
        ROUT(1,J)=.136E+5
        STFO=STBI
        GO TO 150
C
C       CHECK TC SEE IF A ROW OTHER THAN THE FIRST ONE IN THE SECTOR HAS
C       HAD STEAM SATURATION DROP BELOW PTLIM. IF IT HAS THEN CHECK2<0.
C       DUMMY VALUES MUST THEN BE PLACED IN THOSE VARIABLES NORMALLY
C       RETURNED BY HETTRN SINCE HETTRN HAS BEEN BYPASSED.
C
130     IF (CHECK2.LT.0) GO TO 140
        ENHFSV=ENHF
        GO TO 150
C
```

```
C       IF CHECK2 < 0, THAT INDICATES THAT THE SATURATION STEAM PRESSURE
C       IS BELOW PTLIM AND THUS THE STEAM TEMPERATURE IS BELOW COOLANT
C       INLET TEMPERATURE. THIS MEANS THAT NO HEAT IS REMOVED FROM THE
C       STEAM AND THE ALMTD IS 0. THE COOLANT OUTLET TEMP EQUALS THE
C       COOLANT INLET TEMP AND THE TUBE FILM TEMP IS THE SAME AS THE
C       COOLANT INLET TEMP. THE REMAINING VARIABLES ARE GIVEN THE LAST
C       GOOD VALUES RETURNED BY HETTRN.
C
140     ALMTD(I,J)=0.
        HOMCI=0.
        HOMCO=0.
        VPSHH(I,J)=STBI
        HEFF(I,J)=HEFF(I-1,J)
        RC(I,J)=RC(I-1,J)
        VNRE(I,J)=VNRE(I-1,J)
        SHI(I,J)=SHI(I-1,J)
        SHN(I,J)=SHN(I-1,J)
        UN(I,J)=UN(I-1,J)
        ROUT(I,J)=ROUT(I-1,J)
        STFO=STBI
        ENHF=ENHFSV
C
150     TB2=VPSHH(I,L)
        HFG=HFGFN(STSAT(I,J))
C
C       CHECK TO SEE IF STEAM FLOW IN A SECTOR HAS ALREADY GONE TO 0
C       IF IT HAS CHECK1<0 AND DUMMY VALUES HAVE ALREADY BEEN ASSIGNED
C       TO WCNC.
C
        IF (CHECK1.LT.0.) GO TO 160
C
        WCND(I,J)=UN(I,J)*AO*ALMTD(I,J)/HFG*TBNPR(I)
C
C       CHECK TC SEE IF SATURATION PRESSURE HAS ALREADY GONE BELOW PTLIM
C       AND DUMMY VARIABLES HAVE BEEN ASSIGNED TO DELPTP
C
C       IF (CHECK2.LT.0) GO TO 170
C
160     CALL PRSDRP (TAL,VMIX,WS(I,J),WNC,AXO,OD,VPSH,DELPTP,ENHF)
C170    CONTINUE
        JX=0
C$--------------
C SEC A5
C                       TEMP OUTPUT SECTION
C
C ZZZSEC-A5
C--------------
        SECA5=0.
```

```
      IF (SECA5.NE.1.) GO TO 170
      WRITE (6,860) I,J
      WRITE (6,910) TB2,UN(I,L),VNRE(I,L)
      WRITE (6,870) ALMTD(I,L),HEFF(I,L)
      WRITE (6,880) HOMCI,HOMCO
      WRITE (6,890) RC(I,L),ROUT(I,L)
      WRITE (6,900) SHI(I,L),SHN(I,L)
      WRITE (6,920) WNC,WS(I,L)
      WRITE (6,930) DELPTP,VPSH
      WRITE (6,940) WCND(I,J),HFG
      WRITE (6,950) AO,TBNPR(I)
C$-------------
C SEC 6
C                 TEST FOR POSITIVE STM FLOW AND SAT PRESS
C
C          WS = STM FLOW TO ONE ROW IN A SECTOR  LB/HR
C          WNCD = CONDENSATE FROM ONE ROW OF TUBES IN A SECTOR  LB/HR
C          WNC = N/C GAS FLOW TO ONE ROW OF TUBES IN ASECTOR  LB/HR
C          AXO = AREA OPEN TO FLOW IN A ROW OF A SECTOR
C          GFLW = MASS FLW RATE OF MIXTURE IN A TUBE BUNDLE LB/(HR-FT**2)
C          I = ROW NUMBER
C          AMOLS = MOLES OF STEAM IN A ROW OF A SECTOR
C          PMIX = PRESS OF STM - N/C MIXTURE
C          DELPTP = PRESS DROP ACROSS A ROW OF TUBES
C          AMWNC = MOLECULAR WEIGHT OF N/C
C          PSAT = SATURATED STEAM PRESS
C          J = SECTOR NUMBER
C          STSAT = SATURATED STEAM TEMP - F
C          TSAT = SATURATED STEAM TEMP - R
C          PAL = DUMMY FOR PSAT
C          CHECK1 = INDICATOR THAT NEG. STEAM FLOW HAS BEEN ENCOUNTERED
C                   IN SECTOR ANALYSIS
C          CHECK2 = INDICATOR THAT PRESS<PSATFN(STBI) HAS BEEN
C                   ENCOUNTERED IN SECTOR ANALYSIS
C          PTEST = NEXT ROW PRESSURE VALUE TO CHECK FOR A BAD VALUE
C          WTEST = NEXT ROW STEAM FLOW TO CHECK FOR A NEGATIVE VALUE
C          WTST = TEST VARIABLE WHICH IS NEGATIVE WHEN NEGATIVE STEAM FLOW
C                 IS ENCOUNTERED IN A SECTOR
C          PTST = TEST VARIABLE WHICH IS A NEGATIVE WHEN PSAT DROPS BELOW
C                 PTLIM IN A SECTOR
C
C ZZZ SEC-6
C-------------
C     CHECK TO SEE IF STEAM FLOW HAS ALREADY GONE TO 0 IN AN EARLIER
C     ROW CALCULATIO AND FIXUP HAS BEEN MADE TO WS AND WCND
C
170   IF (CHECK1.LT.0) GO TO 180
      WS(I+1,J)=WS(I,J)-WCND(I,J)
```

```
C                                                                       CON18250
C     CHECK TO SEE IF STEAM FLOW OF NEXT ROW IS 0 AND DUMMY VALUES      CON18260
C     MUST BE ASSIGNED TO WS AND WCND TO ALLOW THE PROGRAM TO CONTINUE  CON18270
C                                                                       CON18280
      IF (WS(I+1,J).LE.1.) IIII=1                                       CON18290
      IF (WS(I+1,J).LE.1.) GO TO 300                                    CON18300
C                                                                       CON18310
C     BEGIN COMPUTING NEXT ROW PRESSURES                                CON18320
C                                                                       CON18330
180   AMOLS=WS(I+1,J)/18.015                                            CON18340
      SECA6=0.                                                          CON18350
      IIII=0                                                            CON18360
C                                                                       CON18370
C     CHECK TC SEE IF SATURATION PRESSURES HAVE ALREADY GONE BELOW      CON18380
C     PTLIM AND STANDARD FIXUP TO PSAT HAS BEEN MADE. PRESSURE FALLING  CON18390
C     BELOW PTLIM IS CONSIDERED ONLY FOR THE CASE OF EXCESSIVE PRESSURE CON18400
C     LOSSES CNLY AND NOT AS A CONSEQUENCE OF LOSS OF STEAM FLOW        CON18410
C                                                                       CON18420
      IF (CHECK2.GE.0) GC TO 190                                        CON18430
      PAL=PSAT(I+1,J)                                                   CON18440
      IF (I.EQ.1) DELPTP=DELPP/(AMOLS/(AMOLS+WNC/AMWNC))                CON18450
      PMIX(I+1,J)=PMIX(I,J)-DELPTP                                      CON18460
      IF (PMIX(I+1,J).LE..001) PMIX(I+1,J)=.001                         CON18470
      GO TO 230                                                         CON18480
C                                                                       CON18490
190   PMIX(I+1,J)=PMIX(I,J)-DELPTP                                      CON18500
      PSAT(I+1,J)=PMIX(I+1,J)*AMOLS/(AMOLS+WNC/AMWNC)                   CON18510
      PAL=PSAT(I+1,J)                                                   CON18520
      PTEST=PSAT(I+1,J)                                                 CON18530
      IF (SECA6.NE.1) GO TO 200                                         CON18540
      IF (SECA6.EQ.1) WRITE (6,1020) I,J                                CON18550
      IF (SECA6.EQ.1) WRITE (6,970) WS(I,J),WS(I+1,J)                   CON18560
      IF (SECA6.EQ.1) WRITE (6,1000) PSAT(I,J),PSAT(I+1,J)              CON18570
C                                                                       CON18580
C     CHECK FCR NEXT ROW SATURATION PRESSURE TO SEE IF DROPS BELOW PTLIMCON18590
C                                                                       CON18600
200   IF (PTEST-PTLIM) 210,210,230                                      CON18610
C                                                                       CON18620
C      PSAT(I+1) < PTLIM AND SUBSEQUENT ROW PRESSURES MUST BE FIXED     CON18630
C      UP WITH DUMMY VALUES IN ORDER TO KEEP THE PROGRAM RUNNING.       CON18640
C      PSAT,AND DELPP ARE GIVEN DUMMY VALUES.                           CON18650
C                                                                       CON18660
210   IK2=I+1                                                           CON18670
      SECA6=0.                                                          CON18680
      NRWSP2=NOROWS+1                                                   CON18690
      ROTOG=FLOAT(NRWSP2-I)                                             CON18700
      DELPP=PTLIM/(ROTOG+1)                                             CON18710
C                                                                       CON18720
```

```
C      DELPP IS AN ARBITRARY INCREMENTAL PRESS. DROP OVER REMAINING ROWS
C
       DO 220 IK=IK2,NRWSP2
       PSAT(IK,J)=PSAT(IK-1,J)-DELPP
220    CONTINUE
       X1=PTEST-PTLIM
       X2=PSAT(I,J)
C
C       CALCULATE PTST WHICH IS A TEST VALUE INDICATING THAT SATURATION
C       PRESSURES WERE REACHED IN THE ANALYSIS FROM WHICH THE
C       CORRESPONDING TSAT IS EXCEEDED BY THE INLET COOLANT TEMP(IE PTLIM
C       WAS VIOLATED) A LARGE NEGATIVE VALUE FOR PTST INDICATES A BAD
C       CONDENSOR DESIGN IN WHICH PTLIM WAS VIOLATED EARLY IN THE SECTOR
C       ANALYSIS. AS THE CONDENSOR MODEL IMPROVES AND SATURATION PRESSURE
C       STAYS ABOVE PTLIM THEN PTST CONVERGES TO 0.
C
       PTST(J)=(X1/(X2-X1))-FLOAT(NOROWS-I)
       CHECK2=-2.
       PAL=PSAT(I+1,J)
       PMIX(I+1,J)=PMIX(I,J)-DELPTP
       IF (PMIX(I+1,J).LE..001) PMIX(I+1,J)=.001
C
       IF (SECA6.EQ.1) WRITE (6,990) I,J
       IF (SECA6.EQ.1) WRITE (6,1000) PSAT(I,J),PSAT(I+1,J)
       IF (SECA6.EQ.1) WRITE (6,1010) PTST(J)
C
       GO TO 230
C$-------------
C SEC 7
C                    CORRECTION TO CONDENSATE RATE FOR
C                           SENSIBLE HEAT
C
C          PAL = SAT STM PRESS
C          STFO = AVE TEMP OF OUTER TUBE FILM
C          UN = OVERALL HEAT X-FER COEF FOR A ROW OF TUBES
C          AO = TUBE OUTER SURFACE AREA
C          ALMTD = LMTD
C          TBNPR = NUMBER OF TUBES IN A ROW OF A SECTOR
C          WS = STEAM FLOW TO A ROW OF A SECTOR
C          WNC = N/C FLOW TO A ROW OF A SECTOR
C          JGAS = FLAG FOR TYPE OF N/C
C          AMWNC = MOLECULAR WEIGHT OF N/C
C          TAL = SAT STM TEMP ENTERING SECTOR  R
C          WCNDP = CONDENSATE FROM ONE ROW CORRECTED FOR SENSIBLE
C                   HEAT CHANGE  LB/HR
C          WCND = CONDENSATE FROM ONE ROW
C          JX =
C          IDENT = USER INPUT IDENTIFICALTION STRING FOR RUN
```

```
C          AMOLS = MOLES OF STEAM IN A ROW OF A SECTOR                  CON19210
C          PMIX = PRESSURE OF STEAM - N/C MIXTURE                        CON19220
C          PSAT = SATURATED STEAM PRESS                                  CON19230
C          TSAT = SATURATED STEAM TEMP  R                                CON19240
C          STSAT = SATURATED STEAM TEMP  F                               CON19250
C                                                                        CON19260
C                                                                        CON19270
C ZZZ SEC-7                                                              CON19280
C---------------                                                         CON19290
230   TSAT=TSATFN(PAL)                                                   CON19300
      SECA7=0.                                                           CON19310
      STSAT(I+1,J)=TSAT-459.69                                           CON19320
      IF (STSAT(I+1,J)-STFO) 240,240,250                                 CON19330
240   STSAT(I+1,J)=STFO                                                  CON19340
      TSAT=STSAT(I+1,J)+459.69                                           CON19350
250   CONTINUE                                                           CON19360
      WCNDP=(UN(I,J)*AO*ALMTD(I,J)*TBNPR(I)-(WS(I,J)*CPSFN(TSAT)/18.015+CON19370
     1WNC*CPAFN(TSAT,JGAS)/AMWNC)*(TAL-TSAT))/HFG                        CON19380
      IF (SECA7.EQ.1) WRITE (6,1070) WCND(I,J),WCNDP                     CON19390
      IF (SECA7.EQ.1) WRITE (6,1080) TAL,TSAT                            CON19400
C                                                                        CON19410
C                                                                        CON19420
C     CHECK TO SEE IF WCND IS 0 WHICH CAN OCCUR IF PSAT HAS DROPPED      CON19430
C     BELOW PTLIM (CHECK2<0). IF IT HAS THERE IS NO HEAT TRANSFER FROM   CON19440
C     THE STEAM AND NO NEED TO CORRECT FOR SENSIBLE HEAT                 CON19450
C                                                                        CON19460
      IF (CHECK2.LT.0) WCNDP=WCND(I,J)                                   CON19470
C                                                                        CON19480
C     CHECK TO SEE IF STEAM FLOW HAS ALREADY GONE TO 0 (CHECK<0)         CON19490
C     AND WCND HAS BEEN FIXED UP WITH DUMMY VALUES                       CON19500
C                                                                        CON19510
      IF (CHECK1.LT.0) WCNDP=WCND(I,J)                                   CON19520
C                                                                        CON19530
C     IF CONDENSATE FLOW HAS GONE TO 0 AVOID ITERATIONS                  CON19540
C                                                                        CON19550
      IF (WCND(I,J).EQ.0.) GO TO 290                                     CON19560
C                                                                        CON19570
C     IF EITHER CHECK IS TRUE THEN AVOID CORRECTIVE ITERATION TO WCND    CON19580
C                                                                        CON19590
      IF (ABS(WCNDP/WCND(I,J)-1.0)-.005) 290,260,260                     CON19600
260   JX=JX+1                                                            CON19610
      IF (JX-50) 280,280,270                                             CON19620
270   IF (SECA7.EQ.1) WRITE (6,1120) IDENT,I,J,I,J                       CON19630
      GO TO 290                                                          CON19640
280   WCND(I,J)=WCNDP                                                    CON19650
      GO TO 170                                                          CON19660
290   CONTINUE                                                           CON19670
      SECA6=0.                                                           CON19680
```

```
C                                                                          CON19690
C     CHECK TO SEE IF STEAM FLOW HAS GONE TO 1 IN AN EARLIER ROW           CON19700
C     CALCULATION AND SUBSEQUENT ROWS HAVE BEEN FIXED UP WITH DUMMY        CON19710
C     VALUES. A NEGATIVE CHECK1 INDICATES FIXUP HAS BEEN MADE FOR          CON19720
C     THAT SECTOR                                                          CON19730
C                                                                          CON19740
      IF (CHECK1.GE.0) GO TO 300                                           CON19750
      GO TO 350                                                            CON19760
C                                                                          CON19770
300   WTEST=WS(I+1,J)                                                      CON19780
C                                                                          CON19790
C     CHECK THE NEXT ROW TO SEE IF STEAM FLOW < 1                          CON19800
C                                                                          CON19810
      IF (WTEST-1.) 310,310,340                                            CON19820
C                                                                          CON19830
C     IF STEAM FLOW HAS GONE TO 1 THEN A STANDARD FIXUP IS MADE TO         CON19840
C     ALLOW THE PROGRAM TO CONTINUE AND DUMMY VALUES FOR STEAM FLOW        CON19850
C     AND CONDENSATE ARE ENTERED INTO THE REMAINING ROWS. IF STEAM FLOW    CON19860
C     HAS GONE TO 1 THAT INDICATES ZERO STEAM FLOW. A VERY SMALL NUMBER    CON19870
C     WILL BE USED TO SIMULATE 0 STEAM FLOW IN THE REMAINING ROWS.         CON19880
C                                                                          CON19890
310   IK1=I+1                                                              CON19900
      ROTOGO=FLOAT(NOROWS-I)                                               CON19910
      WSEND=(WTEST-1.)/(WS(I,J)-WTEST+1.)                                  CON19920
      WTST(J)=WSEND-ROTOGO                                                 CON19930
C                                                                          CON19940
      IF (SECA6.EQ.1) WRITE (6,960) I,J                                    CON19950
      IF (SECA6.EQ.1) WRITE (6,970) WS(I,J),WTEST                          CON19960
      IF (SECA6.EQ.1) WRITE (6,980) WTST(J)                                CON19970
C                                                                          CON19980
C     CALCULATE NEGATIVE FLOW RATE FOR USE IN CALCULATING EXTITFR          CON19990
C                                                                          CON20000
      DO 320 KK=IK1,NCROWS                                                 CON20010
      TUBDRY=TUBDRY+TBNPR(KK)                                              CON20020
320   CONTINUE                                                             CON20030
      TUBDRY=TUBDRY-WSEND*TBNPR(I)                                         CON20040
      CUMDRY=CUMDRY+TUBDRY                                                 CON20050
      SUBRAT=WCND(I-1,J)/TBNPR(I)                                          CON20060
      SUBAVE=SUBAVE+SUBRAT                                                 CON20070
      NBAD=NBAD+1                                                          CON20080
      SUBFLO=SUBFLO+SUBRAT*TUBDRY                                          CON20090
C                                                                          CON20100
      NRWSP1=NOROWS+1                                                      CON20110
      RSTOGO=FLOAT(NRWSP1-I)                                               CON20120
      WS(I,J)=(1.0)*(RSTOGO+1)                                             CON20130
      WCND(I,J)=WS(I,J)                                                    CON20140
      DELWS=WS(I,J)/(RSTOGO+1)                                             CON20150
      DELWC=WCND(I,J)/(RSTOGO+1)                                           CON20160
```

```
      DO 330 IK=IK1,NRWSP1
      IF (SECA6.EQ.1 ) WRITE (6,970) WS(IK,J)
      WS(IK,J)=WS(IK-1,J)-DELWS
330   WCND(IK,J)=WCND(IK-1,J)-DELWC
C
C     CALCULATE A VALUE FOR THE TEST VARIABLE (WTST) INDICATING 0 OR
C     NEGATIVE STEAM FLOW. A LARGE NEGATIVE VALUE FOR WTST INDICATES
C     A VERY BAD CONDENSOR MODEL WHERE NEGATIVE STEAM FLOW WAS
C     ENCOUNTERED EARLY IN THE SECTORS. AS THE CONDENSOR MODEL IMPROVES
C     STEAM FLOW STAYS POSITIVE THROUGHOUT THE CONDENSOR AND WTST
C     CONVERGES TO 0.
C
      CHECK1=-2.
      IF (IIII.EQ.1) GO TO 180
      GO TO 350
340   WCND(I,J)=WCNDP
      WS(I+1,J)=WS(I,J)-WCND(I,J)
350   GFLW(I,J)=(WS(I,J)+WNC)/AXO
      AMOLS=WS(I+1,J)/18.015
C
C     CHECK TO SEE IF PRESURE HAS ALREDY GONE BELOW PTLIM
C     AND DUMMY VALUES ARE IN THE PRESSURE ARRAY (CHECK2<0)
C
      IF (CHECK2.LT.0.) GO TO 360
C
      PSAT(I+1,J)=PMIX(I+1,J)*AMOLS/(AMOLS+WNC/AMWNC)
360   TSAT=TSATFN(PSAT(I+1,J))
      STSAT(I+1,J)=TSAT-459.69
C$-------------
C SEC A7
C                   TEMP OUTPUT SECTION
C
C ZZZ SEC-A7
C--------------
      IF (SECA7.NE.1.) GO TO 370
      WRITE (6,1030) I,J
      WRITE (6,1040) WS(I,J),WCND(I,J)
      WRITE (6,1050) TSAT,STSAT(I,J)
      WRITE (6,1060) GFLW(I,J),PSAT(I,J)
C$-------------
C SEC 8
C                   COMPLETION OF MAIN ROW AND SECTOR LOOPS
C
C             NOTE THAT THE MAIN ROW LOOP IS COMPLETED IN IN LINE 380,
C             AND THE MAIN SECTOR LOOP IS COMPLETED IN LINE 390.
C
C        VG = SPECIFIC VOLUME OF STM
C        VMIX = SPECIFIC VOLUME OF STM-N/C MIXTURE
```

```
C          AMOLS = MOLES OF STM TO A ROW
C          TSAT = STM SATURATION TEMP  R
C          PMIX = ST -N/C MIXTURE PRESSURE
C          AMWNC = MOLECULAR WEIGHT OF N/C GASSES
C          VNC = SPECIFIC VOLUME OF N/C GASSES
C          WNC = N/C FLOW TO A ROW OF A SECTOR
C          AMOLSC = MOLES OF A N/C TO A SECTOR
C          VEL = STM-N/C MIXTURE VELOCITY IN A ROW OF TUBES
C          WS = STM FLOW TO A ROW IN A SECTOR  LB/HR
C          AOTFLW = AREA OPEN TO STM FLOW IN A ROW OF TUBES
C          SMTBI = SUM OF (COOLANT INLET TEMP*COOLANT FLOW RATE)
C          WB = COLLANT FLOW TO ONE TUBE IN THE CONDENSER
C          STBI = COOLANT INLET TEMP
C          TBNPR = NUMBER OF TUBES IN A ROW OF A SECTOR
C          SMWB = ROW BY ROW SUM OF COOLANT FLOW
C          SMTB2 = ROW BY ROW SUM OF (COOLANT OUTLET TEMP*FLOW RATE)
C          TB2 = COOLANT OUTLET TEMP FOR A TUBE
C          SUMQ = SUM OF HEAT DUTY ON ROWS OF TUBES
C          UN = OVERALL HEAT X-FER COEF FOR A ROW
C          AO = TUBE OUTER SURFACE AREA
C          ALMTC = LMTD
C          QOA = HEAT FLUX ACROSS TUBE WALL
C          CUMDP = ACCUMULATED PRESS DROP FROM INLET TO CURRENT ROW
C          PMIX = PRESS OF STM-N/C MIXTURE
C          PHP = OVERALL POWER TO DRIVE THE COOLANT THROUGH THE CONDENSER
C                AND THE COOLER TUBES IN HP.
C          DELP = PRESSURE DROP ACROSS A SECTOR
C          DELPTP = PRESSURE DROP ACROSS A ROW
C          TAL = SAT STEAM TEMP  R
C          I = ROW NUMBER
C          IT = DUMMY LOOP VARIABLE(IT = I)
C          SVMIX1 = SPECIFIC VOLUME OF STM-N/C MIXTURE ENTERING CONDENSER
C          WGAS = FLOW OF N/C GAS TO A SECTOR
C          AOTFLW = AREA OPEN TO STEAM FLOW IN A ROW OF TUBES
C          C = USED TO ADJUST STEAM FLOW TO MATCH PRESS DROPS IN SEC-10
C ZZZ SEC-8
C--------------
370    VG=VGFN(TSAT,PMIX(I+1,J))
       AMOLSC=WNC/AMWNC
       VNC=(10.729*TSAT)/(AMWNC*PMIX(I+1,J))
       VMIX=1.0/((AMOLS/(VG*(AMOLS+AMOLSC)))+(AMOLSC/(VNC*(AMOLS+AMOLSC))
      1))
C      IF (I.EQ.NROWS) GOTO 190
       VEL(I+1,J)=(WS(I+1,J)+WNC)/(AOTFLW(I+1))*VMIX/3600.0
       SMTBI=SMTBI+WB*STBI*TBNPR(I)
       SMWB=SMWB+WB*TBNPR(I)
       SMTB2=SMTB2+WB*TBNPR(I)*TB2
       SUMQ=SUMQ+UN(I,J)*AO*ALMTD(I,J)*TBNPR(I)
```

```
      QOA(I,J)=UN(I,J)*ALMTD(I,J)
      CUMDP(I,J)=(PMIX1)-PMIX(I+1,J)
      DELP(J)=DELP(J)+DELPTP
      TAL=TSAT
380   CONTINUE
      C(J)=(WS(1,J)+WGAS(J))/SQRT(DELP(J))
C
C     INCLUDE CNLY THOSE SECTORS WHERE STEAM FLOW HAS NOT GONE TO 0
C     IN STMSUM SO AS NOT TO INCLUDE DUMMY VALUES IN THE CALCULATION
C     OF EXIT STEAM FRACTION
C
      IF (CHECK1.GE.0) STMSUM=STMSUM+WS(NOROWS+1,J)
390   CONTINUE
C$--------------
C SEC A8
C                    TEMP OUTPUT SECTION
C ZZZ SEC-A8
C--------------
      SECA8=0.
      IF (SECA8.NE.1.) GO TO 410
      WRITE (6,1090)
      DO 400 I=1,ISEC
400   WRITE (6,1100) WTST(I),PTST(I)
410   CONTINUE
C$--------------
C SEC 9
C                    EXIT FRACTION AND PRESS ADJUST
C ZZZ SEC-9
C--------------
      PMXEXT=0.
      DELPVE=0.
      PSUM=0.
      VELEXT=0.
      DO 420 I=1,ISEC
      PAL=PSAT(KSTOP,I)
      PSUM=PSUM+PAL/SECFLG
      VELEXT=VELEXT+VEL(NOROWS+1,I)
      DELPVE=DELPVE+DELP(I)
420   PMXEXT=PMXEXT+PMIX(NOROWS+1,I)
C
C      MAKE PTST AND EXIT STEAM FRACTION VARIABLES CONTINUOUS
C      THIS IS DONE TO SATISFY AND ENHANCE OPTIMIZER CALCULATIONS.
C      THE VARIABLE PTST IS GIVEN THE LAST ROW +1 PRESSURE VALUES IF
C      IT IS NON-NEGATIVE. IN ORDER TO MAKE EXIT STEAM FRACTION
C      (EXITFR) CONTINOUS THE VARIABLE FR IS CREATED WHICH INCORPORATES
C      WTST OF ALL THE SECTORS. IF THE WTST VALUES ARE LARGE NEGATIVE
C      NUMBERS THEN FR-->-1. AS WTST VALUES APPROACH 0 INDICATING STEAM
C      THROUGHOUT THE CONDENSOR THEN FR-->0. FR IS THEN ADDED TO EXITFR
```

```
C       TO MAKE IT CONTINOUS FROM 1-->-1 REFLECTING THE CONDENSOR IN
C       NO STEAM IS EXTRACTED AND THE CONDENSOR WHERE ALL THE TUBES ARE
C       DRY.
C
      DO 430 I=1,ISEC
      IF (PTST(I).GE.0.) PTST(I)=PSAT(NOROWS+1,I)-PTLIM
430   WTSAV=WTSAV+WTST(I)
      FR=-(SUBFLO)/WSI
      DELPVE=DELPVE/SECFLG
      DO 450 J=1,ISEC
C
      IF (ABS(DELP(J)/DELPVE-1.0)-.01) 450,450,440
440   IF (ABS(DELPVE).GT.0.001) GO TO 460
450   CONTINUE
      GO TO 580
460   CONTINUE
      DLPVES=DELPVE
      CONSM=0.0
      WNCFR=WNCI/(WNCI+WSI)
      DO 470 J=1,ISEC
      CONSM=CONSM+C(J)
470   CONTINUE
      DELPVE=(WSAVE*SECFLG)**2/CONSM**2
      DELPVE=(DELPVE+DLPVES)/2.0
      DMAX=ABS(DELPVE-DELP(1))
      JMAX=1
      DO 480 J=2,ISEC
      DHOLD=ABS(DELPVE-DELP(J))
      IF (DHOLD.LE.DMAX) GO TO 480
      DMAX=DHOLD
      JMAX=J
480   CONTINUE
      WSPT=0.0
      WPT=0.0
      DO 500 J=1,ISEC
      WP(J)=C(J)*SQRT(DELPVE)
      WSP(J)=WP(J)*WSI/(WNCI+WSI)
      IF (J.EQ.JMAX) GO TO 500
      FAC=WSP(J)-WS(1,J)
      AFAC=ABS(FAC)
      IF (AFAC.LT.1.E-3) GO TO 490
      CHGSC=0.5*WS(1,J)**0.26
      IF (IPLOOP.GT.3) CHGSC=CHGSC/IPLOOP*2.0
      WSP(J)=WS(1,J)+FAC/AFAC*SQRT(AFAC)*CHGSC
490   CONTINUE
      WSPT=WSPT+WSP(J)
      WGAS(J)=WSP(J)*WNCFR
      WP(J)=WSP(J)+WGAS(J)
```

```
      WPT=WPT+WP(J)                                                     CON22090
  500 CONTINUE                                                          CON22100
      WSP(JMAX)=WSI-WSPT                                                CON22110
      WSPT=WSPT+WSP(JMAX)                                               CON22120
      WGAS(JMAX)=WSP(JMAX)*WNCFR                                        CON22130
      WP(JMAX)=WSP(JMAX)+WGAS(JMAX)                                     CON22140
      WPT=WPT+WP(JMAX)                                                  CON22150
      IPLOOP=IPLOCP+1                                                   CON22160
C                                                                       CON22170
C     CHECK FOR STEAMM ADJUSTMENTS WHICH TRY TO PUT TO LOW A STEAM      CON22180
C     VALUE INTO A SECTOR                                               CON22190
C                                                                       CON22200
  510 IFAIL=0                                                           CON22210
      DO 520 K=1,ISEC                                                   CON22220
      COMP=WSI/200.                                                     CON22230
      IF (WSP(K).LT.COMP) IFAIL=K                                       CON22240
      IF (WSP(K).LT.COMP) WRITE (6,999)                                 CON22250
  999 FORMAT (1X,'STEAM FLOW ADJUSTMENT WAS REQUIRED')                  CON22260
      IF (WSP(K).LT.COMP) GO TO 530                                     CON22270
  520 CONTINUE                                                          CON22280
      GO TO 550                                                         CON22290
  530 DIFF=2.*COMP-WSP(IFAIL)                                           CON22300
      WSP(IFAIL)=2.*COMP                                                CON22310
      ADJ=DIFF/FLOAT(ISEC-1)                                            CON22320
      DO 540 J=1,ISEC                                                   CON22330
      IF (J.EQ.IFAIL) GO TO 540                                         CON22340
      WSP(J)=WSP(J)-ADJ                                                 CON22350
  540 CONTINUE                                                          CON22360
      GO TO 510                                                         CON22370
  550 CONTINLE                                                          CON22380
C                                                                       CON22390
      DO 560 J=1,ISEC                                                   CON22400
  560 WS(1,J)=WSP(J)                                                    CON22410
      IF (IPLOOP-50) 20,20,570                                          CON22420
  570 WRITE (6,1110) IDENT                                              CON22430
C                                                                       CON22440
C     INCORPORATE FR INTO EXITFR TO MAKE IT CONTINUOUS FROM 1-->-1      CON22450
C                                                                       CON22460
  580 EXITFR=STMSUM/WSI+FR                                              CON22470
      WSEXIT=STMSUM                                                     CON22480
      VELEXT=VELEXT/SECFLG                                              CON22490
C     DELPVE = DELPVE/SECFLG                                            CON22500
      PMXEXT=PMXEXT/SECFLG                                              CON22510
      TSATEX=TSATFN(PSUM)                                               CON22520
      AMLSEX=STMSUM/18.015                                              CON22530
      IF ((STMSUM.LE.0.).AND.(PRCCLR.GT.0.)) AMLSEX=1.0/18.015          CON22540
      EXITFC=1.                                                         CON22550
      IF (PRCCLR.EQ.0.0) EXITOA=EXITFR                                  CON22560
```

```
      IF (PRCCLR.EQ.0.0) GO TO 590                                      CON22570
C                                                                       CON22580
C     CALL TO COOLER, IF THERE IS ONE                                   CON22590
C                                                                       CON22600
      ENHIC=ENHI                                                        CON22610
      ENHOC=ENHO                                                        CON22620
      ENHFC=ENHF                                                        CON22630
      SHWINC=SHWINV                                                     CON22640
      TB2C=TB2                                                          CON22650
      WBC=WB                                                            CON22660
      IF (NBAD.GT.0) WCNDAV=SUBAVE/FLOAT(NBAD)                          CON22670
      CALL COCLEX (TBDRYC,WCNDAV,SBFLOC,WSOUT)                          CON22680
C                                                                       CON22690
C     CALCULATE OVERALL POWER REQUIRE TO DRIVE THE COOLANT              CON22700
C                                                                       CON22710
      PHP=PHP*(SMWBC/(SMWBC+SMWB)+1.)                                   CON22720
C                                                                       CON22730
C     CALCULATE THE OVERALL COOLER/CONDENSER VOLUME                     CON22740
      VOL1=(VOL1+VOLC)                                                  CON22750
C                                                                       CON22760
C     CALCULATE OVERALL EXIT FRACTION                                   CON22770
C                                                                       CON22780
      IF ((EXITFR.GT.0.).AND.(EXITFC.GT.0.)) EXITOA=EXITFR*EXITFC       CON22790
      IF ((EXITFR.GT.0.).AND.(EXITFC.GT.0.)) RETURN                     CON22800
C                                                                       CON22810
      EXITOA=WSOUT/WSI+(-(SUBFLO+SBFLOC)/WSI)                           CON22820
C                                                                       CON22830
590   RETURN                                                            CON22840
600   FORMAT (1X,7HSECTOR ,I3,2X,5HROW  ,I4)                            CON22850
610   FORMAT (1H1,37H     *****   OUTPUT FOR SEC-2   *****)             CON22860
620   FORMAT (1H0,10HWSI       ,10HWNCI     )                           CON22870
630   FORMAT (1H0,10HWS(1,I)   ,10HWGAS(I)  )                           CON22880
640   FORMAT (1H0,10HHFG       ,10HAOTFLW(I) )                          CON22890
650   FORMAT (1H0,10HTSAT1     ,10HSTSAT1   )                           CON22900
660   FORMAT (1H0,10HPSAT1     ,10HPMIX1    )                           CON22910
670   FORMAT (1H0,10HAMOLNC    ,10HAMOLSS    ,10HAMOLST    )            CON22920
680   FORMAT (1H0,10HSVNC1     ,10HSVMIX1   )                           CON22930
690   FORMAT (1H0,10HVG1       ,10HG2       )                           CON22940
700   FORMAT (1H ,2F10.3)                                               CON22950
710   FORMAT (1H ,3F10.3)                                               CON22960
720   FORMAT (1H0,10HSECFAR    ,10HARAT     )                           CON22970
730   FORMAT (1H0,10HVEL1      ,5HVEL2 )                                CON22980
740   FORMAT (1H0,10HDELP1     ,5HDELP2)                                CON22990
750   FORMAT (1H ,2F10.4)                                               CON23000
760   FORMAT (1H1,29HOUTPUT FOR SEC-3 -SECTOR NO. ,I2)                  CON23010
770   FORMAT (1H0,36HENTERING PRESSURES IN PSI - MIXTURE=,E10.3,13H SATUCON23020
     1RATION= ,E10.3)                                                   CON23030
780   FORMAT (1H ,21HENTERING STM TEMPS - ,F6.2,4H F  ,F6.2,2H R)       CON23040
```

```
790    FORMAT (1H ,32HENT SP. VOL. IN FT**3/LBM - STM=,E10.3,9H N/C GAS=,CON23050
      1E10.3,9H MIXTURE=,E10.3)                                          CON23060
800    FORMAT (1H ,23HENTERING STM VELOCITY =,E10.3,8H  FT/SEC)           CON23070
810    FORMAT (1H0,33H** OUTPUT FOR SEC-4 - ROW NUMBER ,I4)               CON23080
820    FORMAT (1H0,33HTUBE WALL AREA - SQ.FT. - INSIDE=,E10.3,10H  OUTSIDCON23090
      1E=,E10.3)                                                         CON23100
830    FORMAT (1H ,9HTUBE ID= ,F6.3,30H  FT - TUBE X-SECTIONAL AREA =,E10CON23110
      1.3,6H FT**2)                                                      CON23120
840    FORMAT (1H ,24HLOG MEAN TUBE THICKNESS=,E10.3,25H FT - LOG MEAN TUCON23130
      1BE AREA=,E10.3,6H FT**2)                                          CON23140
850    FORMAT (1H ,30HINVERSE WALL HEAT X-FER COEF =,E10.3,17H HR-FT**2-FCON23150
      1/B  TU)                                                           CON23160
860    FORMAT (1H0,22HIN SECALC, SEC-5, ROW,,I4,9H, SECTOR,,I3)           CON23170
870    FORMAT (1H ,7HALMTD =,E10.3,8H HEFF  =,E10.3)                      CON23180
880    FORMAT (1H ,7HHOMCI =,E10.3,8H HOMCO =,E10.3)                      CON23190
890    FORMAT (1H ,7HRC    =,E10.3,8H ROUT  =,E10.3)                      CON23200
900    FORMAT (1H ,7HSHI   =,E10.3,8H SHN   =,E10.3)                      CON23210
910    FORMAT (1H ,7HTB2   =,E10.3,8H UN    =,E10.3,8H VNRE  =,E10.3)     CON23220
920    FORMAT (1H ,7HWNC   =,E10.3,8H WS    =,E10.3)                      CON23230
930    FORMAT (1H ,7HDELPTP=,E10.3,8H VPSH  =,E10.3)                      CON23240
940    FORMAT (1H ,7HWCND  =,E10.3,8H HFG   =,E10.3)                      CON23250
950    FORMAT (1H ,7HAO    =,E10.3,8H TBNPR =,E10.3)                      CON23260
960    FORMAT (1H0,32HNEG STEAM MASS FLOW IN SEC-6 ROW,I4,10H OF SECTOR,ICON23270
      14)                                                                CON23280
970    FORMAT (1H ,18HCURRENT FLOW RATE=,F12.6,17H  NEXT FLOW RATE=,F12.6CON23290
      1)                                                                 CON23300
980    FORMAT (1H ,8HWTST(I)=,F12.6)                                      CON23310
990    FORMAT (1H0,30HNEGATIVE SAT PRSS IN SEC-6 RCW,I4,9H  SECTOR ,I4)   CON23320
1000   FORMAT (1H ,18HCURRENT SAT PRESS=,E10.3,11H  NEXT ROW=,E10.3)      CON23330
1010   FORMAT (1H ,8HPTST(I)=,E10.3)                                      CON23340
1020   FORMAT (1H ,25HNO PROBLEMS IN SE-6, ROW=,I4,8H SECTOR=,I4)         CON23350
1030   FORMAT (1H0,23HOUTPUT FROM SEC-7, ROW ,I4,11H OF SECTOR ,I4)       CON23360
1040   FORMAT (1H ,9HWS(I,J) =,E10.3,13H  WCND(I,J) =,E10.3)              CON23370
1050   FORMAT (1H ,6HTSAT =,E10.3,14H  STSAT(I,J) =,E10.3)                CON23380
1060   FORMAT (1H ,11HGFLW(I,J) =,E10.3,13H  PSAT(I,J) =,E10.3)           CON23390
1070   FORMAT (1H ,28HSEC-7, BELOW 130, WCND(I,J)=,E10.3,8H  WCNDP=,E10.3CON23400
      1)                                                                 CON23410
1080   FORMAT (1H ,22HSEC-7, BELOW 130, TAL=,E10.3,6H TSAT=,E10.3)        CON23420
1090   FORMAT (1H1,3X,4HWTST,8X,4HPTST/)                                  CON23430
1100   FORMAT (1H ,E10.3,2X,E10.3,2X,E10.3)                               CON23440
1110   FORMAT (1H1,19A4,A3,40HNO CONVERGENCE IN COND. DELTA P (IPLOOP))   CON23450
1120   FORMAT (1H1,19A4,A3/1H0,5X,2HI=,I2,5X,2HJ=,I1/1H0,24HNO CONVERGENCCON23460
      1E FOR WCND(,I2,1H,,I1,1H))                                        CON23470
       END                                                               CON23480
C                                                                        CON23490
C***********************************************************************  CON23500
C***********************************************************************  CON23510
C                                                                        CON23520
```

```
      SUBROUTINE HETTRN (ALST,AXO,IR,LJ,CD,WB,SHWINV,STSAT,TID,WNC,WST,A
     1LMTD,ENHF,HEFF,HOMCI,HOMCO,RC,ROUT,SHI,SHNF,STFO,TB2,VNF,XNRE,ENHI
     2,ENHO,SDD)
C--------------
C HET 1
C                 SUBROUTINE HETTRN
C
C         HETTRN CALLED BY: SECALC
C         SUBROUTINES CALLED BY HETTRN:
C              , DSFVTY ,
C         FUNCTION ROUTINES CALLED BY HETTRN:
C              , PSATFN , SMUFN , AMUFN , HFCFN , SKBFN ,
C              , BMUFN , CPFN , ROEFN ,
C
C--------------
C**   DEC 69 REV.   COLBJ,FDAVE
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,NNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /ORC2/ ANT(100,15),STB2ES,VW(100)
      COMMON /CONST/ AMWNC,CBI,CPB,PI,SG,IFIRST
      DIMENSICN R2(3),E2OK(3)
      HETON=0.
      IF (HETCN.NE.1.) GO TO 10
      M5=1
      WRITE (6,310) M5
      WRITE (6,320) IR,LJ
      WRITE (6,330) OD,TID
      WRITE (6,340) SHWINV,STSAT
      WRITE (6,350) WST,WNC
C$-------------
C HET 2
C                 DATA INITIALIZATION:
C
C          BB = COOLANT FLOW IN TUBE  LBM/HR-FT**2
C     DGFLM =CALCULATED TEMP DROP ACROSS FILM???
C     DLFLM = CALCULATED TEMP DROP ACROSS LIQUID FILM???
C     HEFF = NONCONDENSIBLE FILM HEAT TRANSFER COEF, BTU/HR-FT**2-F
C     UEST = ESTIMATED OVERALL HEAT TRANSFER COEF
C
C          VALUES FOR USE IN SUBROUTINE DFSVTY
C     E2OK = FORCE CONSTANT FOR NONCOND GAS
C     R2 = GAS COLLISION DIAMETER IN ANGSTROMS
C ZZZ HET-2
C--------------
10    IF (WNC.EQ.0.) DGFLM=0.
      IF ((LJ.EQ.1).AND.(IR.EQ.1)) GO TO 20
```

```
      IF (IR.EQ.1) GO TO 30                                             CON24010
      UEST=UESTSV                                                       CON24020
      HEFF=HEFFSV                                                       CON24030
      DGFLM=DGFLMS                                                      CON24040
      DLFLM=DLFLMS                                                      CON24050
      TB2=TB2SV                                                         CON24060
      XNRET=XNRES                                                       CON24070
      GO TO 40                                                          CON24080
   20 IF (IFIRST.EQ.0) GO TO 30                                         CON24090
      HEFF=2100.                                                        CON24100
      DATA UEST,DGFLM,DLFLM,XNRET/700.,0.05,2.0,100000./                CON24110
      DATA R2,E20K/3.617,3.996,3.838,97.0,190.0,140.75/                 CON24120
      TB2=STB2ES                                                        CON24130
      IFIRST=0                                                          CON24140
      GO TO 40                                                          CON24150
   30 UEST=UEST1                                                        CON24160
      HEFF=HEFF1                                                        CON24170
      DGFLM=DGFLM1                                                      CON24180
      DLFLM=DLFLM1                                                      CON24190
      TB2=TB21                                                          CON24200
      XNRET=XNRE1                                                       CON24210
   40 CONTINUE                                                          CON24220
C                                                                       CON24230
C                                                                       CON24240
C    FDAVE IS AN INPUT ITEM WITH A VALUE BETWEEN 0 AND 1 RELATED TO TUBE CON24250
C    SPACING AND ORIENTATION.                                           CON24260
      FDAVE=C.5                                                         CON24270
C                                                                       CON24280
C                                                                       CON24290
      BNF=ANT(IR,LJ)                                                    CON24300
      IF (BNF.LT.1.) BNF=1.                                             CON24310
      BB=WB/(PI*TID*TID/4.)                                             CON24320
      AI=PI*TID*ALST                                                    CON24330
      AO=PI*CC*ALST                                                     CON24340
      ENHI=1.                                                           CON24350
      ENHF=1.                                                           CON24360
      ENHO=1.                                                           CON24370
      IF (NEI.EQ.0) GO TO 60                                            CON24380
      DO 50 I=1,NEI                                                     CON24390
      ITEM1=NRNI(I)                                                     CON24400
      ITEM2=ITEM1+NETI(I)-1                                             CON24410
   50 IF ((I.GE.ITEM1).AND.(I.LE.ITEM2)) ENHI=ENI(I)                    CON24420
   60 IF (NEE.EQ.0) GO TO 80                                            CON24430
      DO 70 I=1,NEE                                                     CON24440
      ITEM1=NRNE(I)                                                     CON24450
      ITEM2=ITEM1+NETE(I)-1                                             CON24460
      IF (I.LT.ITEM.OR.I.GT.ITEM2) GO TO 70                             CON24470
      EHNO=ENO(I)                                                       CON24480
```

```
      ENHF=ENF(I)                                                       CON24490
70    CONTINUE                                                          CON24500
80    CONTINUE                                                          CON24510
      IF (HETCN.NE.1.) GO TO 90                                         CON24520
      M5=2                                                              CON24530
      WRITE (6,310) M5                                                  CON24540
      WRITE (6,360) UEST,HEFF                                           CON24550
      WRITE (6,370) DGFLM,DLFLM                                         CON24560
      WRITE (6,380) TB2,XNRET                                           CON24570
      WRITE (6,390) BNF,AO,AI                                           CON24580
      WRITE (6,400) ENHI,ENHO                                           CON24590
C$--------------                                                        CON24600
C HET 3                                                                 CON24610
C                 CALCULATE SUBPROGRAM INITIAL VALUES                   CON24620
C                                                                       CON24630
C        AMOLSS & AMOLSC ARE MOLECULAR WEIGHTS USED TO CALCULAT         CON24640
C           AMWAV, THE MIXTURE MOLECULAR WEIGHT.                        CON24650
C        AXO = AREA OF TUBE BANK OPEN TO FLOW/TUBE                      CON24660
C        BB = COOLENT FLOW IN TUBE    LB/HR-FT**2   .                   CON24670
C        GMAX = STEAM-NC FLOW  LB/FT**2-HR                              CON24680
C        HSTO = LATENT HEAT OF VAPORIZATICN OF THE STEAM                CON24690
C        JGAS = INDICATOR FOR TYPE OF N/C GAS PRESENT                   CON24700
C        K = COUNTER FOR NUMBER OF ITERATIONS                           CON24710
C        PSAT = STEAM SATURATION PRESS IN PSI                           CON24720
C        RC = RATIO OF NON-CONDENSIBLES TO TOTAL FLOW                   CON24730
C        RMWNC & RMWSC ARE WEIGHTING FACTCRS USED IN DETERMINATION      CON24740
C           OF THE MIXTURE VISCOSITY.                                   CON24750
C        OD = TUBE OUTER DIAMETER                                       CON24760
C        STBAVE = AVERAGE COOLANT TEMP                                  CON24770
C        STBI = INLET COOLANT TEMP                                      CON24780
C        STSAT = LOCAL STEAM SAT TEMP  F                                CON24790
C        STWO = TUBE OUTER WALL TEMP                                    CON24800
C        TB2 = OUTLET COOLANT TEMP                                      CON24810
C        WBOTH = SUM OF WS AND WNC                                      CON24820
C        WNC = NON-CONDENSIBLE FLOW LB(MASS)/HR                         CON24830
C        WST = STM FLCW TO ROW BEING CONSIDERED LB(M)/HR                CON24840
C        XNRE = MIXTURE REYNOLDS NUMBER                                 CON24850
C                                                                       CON24860
C            VALUES CARRIED INTO THIS ROUTINE                           CON24870
C        COMMCN - , AMWNC ,                                             CON24880
C            ORC2 - , AMWNC ,                                           CON24890
C        COMMCN/NEW/ - , STBI , JGAS ,                                  CON24900
C        PASS LIST - , STSAT ,                                          CON24910
C            SEC3 - , STSAT ,                                           CON24920
C        HET2 - , DLFLM , DGFLM , OD , AXO , WST , WNC ,                CON24930
C                                                                       CON24940
C            FUNCTION ROUTINES CALLED IN HET-3                          CON24950
C      , PSATFN , SMUFN , AMUFN , HFGFN ,                               CON24960
```

```
C
C ZZZ HET-3
C---------------
90     K=0
       WBOTH=WST+WNC
       RC=WNC/(WBOTH)
       TSAT=STSAT+459.69
       GMAX=(WBOTH)/AXO
       PSAT=PSATFN(TSAT)
       AMOLSS=WST/18.015
       AMOLSC=WNC/AMWNC
       AMOLT=AMOLSS+AMOLSC
       AMWAV=(WBOTH)/AMOLT
       RMWNC=SQRT(AMWNC)*AMOLSC/AMOLT
       RMWSC=4.24444*AMOLSS/AMOLT
       AVIS=(SMUFN(TSAT)*RMWSC+AMUFN(TSAT,JGAS)*RMWNC)/(RMWSC+RMWNC)
       HSTO=HFGFN(STSAT)
       STBAVE=(STBI+TB2)/2.0
       DELFLM=DLFLM
       DLGFLM=DGFLM
       STWO=STSAT-DELFLM-DLGFLM
       STFO=(STSAT+STWO)/2.0
       XNRE=OD*(WBOTH)/(AXO*3600.*AVIS)
       IF (HETCN.NE.1.) GO TO 100
       M5=3
       WRITE (6,310) M5
       WRITE (6,410) GMAX,PSAT
       WRITE (6,420) AMOLSS,AMOLSC,AMOLT
       WRITE (6,430) RMWNC,RMWSC
       WRITE (6,440) AVIS,HSTO,XNRE
       WRITE (6,450) STBAVE,STFO,STWO
       WRITE (6,460) DELFLM,DLGFLM
C$--------------
C HET 5
C                  COLBY J FACTOR CALCULATION AND BRANCH
C
C          WNCI = TOTAL WEIGHT OF NONCOND ENTERING CONDENSER
C                 IF NO NONCOND, THEN SKIP DIFUSIVITY CALCULATION.
C
C ZZZ HET-5
C---------------
100    CONTINUE
       COLBJ=EXP(0.53883-0.544*ALOG(XNRE))
       IF (HETCN.NE.1.) GO TO 110
       M5=5
       WRITE (6,310) M5
       WRITE (6,470) COLBJ,WNC
110    IF (WNC.EQ.0.) GO TO 120
```

```
C$--------------
C HET 6
C                   PREPARATION FOR CALL TO SUBROUTINE DFSVTY;
C                   CALCULATE SCHMIDT NUMBER AND PARAMETER CJ.
C
C              THIS SECTICN IS BRANCHED ARCUND IF NO NONCOND GASSES ARE
C                   PRESENT.
C
C         AMWAV = STEAM-N/C GAS MIXTURE MOLECULAR WEIGHT
C         AVIS = AVG VISCOSITY OF STEAM-N/C GAS MIXTURE
C         CJ = PARAMETER USED IN CALCULATION OF THE HEAT TRANSFER COEF.
C            SEE EQUATION 11 IN APP B
C         DD = DIFFUSIVITY (SEE NOTE)
C         DDG = DUMMY PASSING PARAMETER FOR SUB DSFVTY
C         E2OK = N/C GAS FORCE CONSTANT (SEE SUB DSFVTY)
C         PATM = SAT PRESS IN ATM
C         PGB = NONCOND PARTIAL PRESSURE
C         PSAT = STEAM SATURATION PRESS PSI
C         R = N/C GAS COLLISION DIAMETER (SEE SUB CSFVTY)
C         TSAT = STEAM SATURATION TEMP
C         TSATK = SATURATION TEMP IN DEGREES KELVIN
C         XNSCH = SCHMIDT NUMBER
C
C              NOTE:   PATM, PGB, AND TSATK ARE USED FOR CALL TO DFSVTY
C                      SEE NOTE IN HET-3, BUT REM PGB IS USED AGAIN
C                      IN HET-12.
C
C              NOTE:   18.015 IS THE MOLECULAR WEIGHT OF WATER
C                      2.655 IS THE STEAM COLLISION DIAMETER IN ANGSTROMS
C                      363. IS THE STEAM FORCE CONSTANT USED IN DFSVTY
C
C              NOTE:   ONLY DG (DIFFUSIVITY IN CM**2/SEC) IS MODIFIED IN
C                      SUBROUTINE DFSVTY.
C
C              NOTE:   0.258 CONVERTS UNITS OF DIFUSIVITY FROM
C                      CM**2/SEC TO FT**2/HR, AND 3600 CONVERTS
C                      VISCOSITY  FROM LB(F)-HR TO LB(F)-SEC.
C                      IN XNSCH CALC, .258*3600 = 928.8
C
C
C ZZZ HET-6
C--------------
      PATM=PSAT/14.7
      PGB=PSAT*(AMOLSC/AMOLSS)
      TSATK=TSAT/1.8
      CALL DFSVTY (TSATK,PATM,18.015,AMWNC,2.655,R2(JGAS),363.,E2OK(JGAS
     1),0.,0.,DG,DGG)
      XNSCH=(AVIS*TSAT*10.73/((PSAT+PGB)*AMWAV*DG))*928.8
```

```
      CJ=COLBJ*GMAX/(XNSCH**0.666667)
      IF (HETCN.NE.1.) GO TO 120
      M5=6
      WRITE (6,310) M5
C$-------------
C HET 7
C                  BEGIN ITERATIVE CALCULATIONS FOR:
C                        HEAT TRANSFER COEF
C                    TEMP DROP ACROSS LIQUID FILM
C                  TEMP DROP ACROSS N/C FILM (IF ANY)
C
C              THE PROGRAM CONTINUES TO RETURN HERE UNTIL ONE
C              OR MORE OF THE FOLLOWING CONDITIONS ARE MET:
C                  1)  CHANGE IN THE CALCULATED HEAT TRANSFER COEF IS
C                     LESS THAN  OF THE PREVIOUS VALUE AND AT LEAST
C                     FOUR ITERATIONS HAVE BEEN MADE.
C                  2)  CHANGE IN THE CALCULATED TEMP ACROSS THE LIQUID
C                     FILM IS LESS THAN .01 DEGREES AND THE CHANGE
C                     IN CALCULATED TEMP DIFF ACROSS THE GAS FILM
C                     IS LESS THEN .001 DEGREES.
C                  3)  TEN ITERATIONS HAVE BEEN COMPLETED.
C
C         AI = INTERNAL SURFACE AREA OF ONE TUBE   FT**2
C         AO = EXTERNAL SURFACE AREA OF ONE TUBE   FT**2
C         BB = COOLANT FLOW IN TUBE  LBM/HR-FT**2
C         BMU = COOLANT VISCOSITY  LBM/HR-FT
C         BNF = TUBE FLOODING FACTOR
C         CBI = COOLANT CONCENTRATION
C         ENHI = INTERNAL FILM COEF ENHANCEMENT FACTOR
C         ENHO = EXTERNAL FILM COEF ENHANCEMENT FACTOR
C         RIN = RESISTANCE TO HEAT TRANSFER DUE INNER FILM
C         TID = TUBE ID  FT
C         OD = TUBE OD  FT
C         SHBI = SPECIFIC HEAT OF COOLANTBTU/LBM-F
C         SHI = INTERNAL FILM HEAT TRANSFER COEF   BTU/HR-FT**2-F
C         SHMK = EXTERNAL FILM HEAT TRANSFER COEF  BTU/HR-FT**2-F
C         SKB = THERMAL CONDUCTIVITY OF THE COOLANT  BTU-FT/HR-FT**2-F
C         SKBO = THERMAL CONDUCTIVITY OF OUTER LIQUID FILM
C         STBAVE = AVE COOLANT TEMP.
C         STFO = AVE TEMP OF OUTER LIQUID FILM (SEE NOTE)
C         STSAT = LOCAL SAT STM TEMP
C         TB2 = OUTLET COOLANT TEMP.  (SEE NOTE ON TB2 IN HET-3)
C         WB = COOLANT FLOW RATE PER TUBE  LBM/HR
C         XNREB = REYNOLDS NUMBER OF COOLANT
C         XNPRB = PRANDTL NUMBER OF COOLANT
C
C                  NOTE:SHMK IS CALCULATED FIRST WHTHOUT TAKING FLOODING
C                       INTO ACCOUNT.  SEE NUSSELT EQN PG. 16.
```

```
C
C                     NOTE: STFO  IS INITIALLY GUESSED BASED ON STWO WHICH
C                           IS IN TURN BASED ON DLFLM AND DGFLM.  DGFLM AND
C                           DLFLM ARE INITIALIZED BY CATA STATEMENT UPON
C                           ENTERING THIS SUBROUTINE.  ALL FOUR OF
C                           THESE VARIABLES ARE UPDATED BELOW.
C
C
C ZZZ HET-7
C---------------
120   TB2=STSAT-(STSAT-STBI)/EXP(UEST*AO/(WB*CPFN(CBI,STBAVE)))
      STBAVE=(STBI+TB2)/2.
      SKBO=SKBFN(O., STFO)
      SKB=SKBFN(CBI, STBAVE)
      BMU=BMUFN(CBI, STBAVE)
      SHBI=CPFN(CBI, STBAVE)
      XNREB=TID*BB/BMU
      XNPRB=SHBI*BMU/SKB
      SHI=0.023*(XNREB)**0.8*XNPRB**0.33333*(SKB/TID)*ENHI
      RIN=AO/(SHI*AI)
      SHMK=0.725*(SKBO**3*ROEFN(0.0,STFO)**2*HFGFN(STFO)*416975040.0/(BM
     1UFN(O,STFO)*OD*DELFLM))**.25*ENHO
C
C  THIS IS THE PLACE TO ENTER CORRECTIONS TO THE EXTERNAL FILM COEFF.THE
C  FOLLOWING IS A CORRECTION ACCORDING TO FUJII, ET. AL, TO ACCOUNT FOR
C  THE EFFECTS OF VAPOR SHEAR AT HIGH VAPOR REYNOLDS NUMBERS.
C
      SHMK1=SHMK
      ANUS=SHMK*OD/SKBO
      EMUL=BMUFN(O., STSAT)
      EMUS=SMUFN(TSAT)*3600.
      RHOL=ROEFN(O., STSAT)
      RHOS=1./VGFN(TSAT,PSAT)
      VISRAT=(EMUS/RHOS)/(EMUL/RHOL)
      RETP=XNRE*VISRAT
      TERM=SQRT(RETP)/ANUS
      IF (TERM.LT..278) GO TO 130
      ANUSP=1.24*(ANUS**0.8)*(RETP**0.1)
      GO TO 140
130   ANUSP=ANUS*0.70/0.725
140   SHMKP=ANUSP*SKBO/OD
      SHMK=SHMKP
C
C
      IF (HETCN.NE.1.) GO TO 150
      M5=7
      WRITE (6,310) M5
      WRITE (6,480) TB2, STBAVE
```

```
      WRITE (6,490) SKBO,SKB
      WRITE (6,500) BMU,SHBI
      WRITE (6,510) XNREB,XNPRB
      WRITE (6,520) SHI,RIN,SHMK
C$------------
C HET 9
C                   TEST FOR TUBE FLOODING, MODIFY
C                       SHMK AS NEEDED
C
C              THIS SECTION MODIFIES THE EXTERNAL HEAT TRANSFER
C                COEF GIVEN BY NUSSELT EQN. FOR TUBE FLOODING.
C
C         BNF = CALCULATED TUBE FLOODING FACTOR (SEE NOTE BELOW)
C         FDAVE = INPUT VALUE OF TUBE FLOODING FACTOR (SEE NOTE BELOW)
C         FDAVN & FDAVM ARE INTERMEDIATE VALUES USED LATER.
C         FOUL = TUBE FOULING FACTOR INPUT BY USER
C         RFACT = SUM OF THERMAL RESISTANCES NEGLECTING GAS FILM
C         RIN = RESESTANCE TO HEAT X-FER DUE TO INTERNAL FILM
C         SHNF = EXTERIOR HEAT X-FER COEF, SHMK MODIFIED FOR RAIN
C         SHWINV = INVERSE OF WALL HEAT TRANSFER CCEF
C
C                   NOTE:ADD CALC FOR FDAVE     CURRENTLY UNDEF
C
C                   NOTE:CALC FOR FDAVM AND FDAVN SHOULD BE MOVED
C                        ABOVE LINE 40.
C
C
C ZZZ HET-9
C--------------
150   CONTINUE
C
C      INUNDATION EFFECT MAY BE BYPASSED BY SETTING BNF = 1.
C
C     BNF = 1.
      IF (FDAVE.LT.0..OR.BNF.LT.2.) GO TO 160
      FDAVN=0.6*FDAVE+(1.-0.5647*FDAVE)/BNF**.20
      FDAVM=0.6*FDAVE+(1.-0.5647*FDAVE)/(BNF-1.)**.20
      SHNF=SHMK*(BNF*FDAVN-(BNF-1.)*FDAVM)
      GO TO 170
160   SHNF=0.95*(BNF**0.9-(BNF-1.)**0.9)*SHMK
170   RFACT=RIN+SHWINV+FOUL+1./SHNF
      IF (HETCN.NE.1.) GO TO 180
      M5=9
      WRITE (6,310) M5
      WRITE (6,530) FDAVN,FDAVM
      WRITE (6,540) SHNF,RFACT
C$------------
C HET 10
```

```
C                 CALCULATE LOG-MEAN TEMP-DIFFERENCE
C
C             THIS CALCULATION IS COMPLETED ONCE DURING EACH
C             ITERATION.  UPON COMPLETION, A TEST IS MADE
C             FOR THE PRESENCE OF N/C GASSES.  IF ANY
C             GAS IS PRESENT, PROGRAM BRANCES TO LINE 110 (HET-12).
C             IF NOT, THEN PROGRAM FLOWS TO HET-11, BELOW.
C
C         ALMDT = LMTD
C         HOMCO = TEMP DIFF BETWEEN STEAM AND COOLANT OUTLET
C         HOMCI = TEMP DIFF BETWEEN STEAM AND COOLANT AT INLET
C         JRC = PRINT CONTROL FLAG
C         RODT = RATIO OF HOMCI TO HOMCO
C         STBI = COOLANT INLET TEMP
C         STSAT = STEAM SATURATION TEMP
C         TB2 = COOLANT OUTLET TEMP
C         WNCI = TOTAL N/C GAS FLOW TO CONDENSER
C
C
C ZZZ HET-10
C--------------
180    HOMCI=STSAT-STBI
       HOMCO=STSAT-TB2
C
C      IF (HOMCO.LT.1.E-15) WRITE(6,20600) HOMCI,HCMCO
C      IF (HOMCI.LT.1.E-15) WRITE(6,20600) HOMCI,HOMCO
       IF (HOMCO.LT.1.E-15) HOMCO=1.E-15
C
       RODT=HCMCI/HOMCO
C
C      IF INLET/OUTLET DELTA T RATIO IS NEAR 1.0, ALMTD = ARITH. AVG.
       IF (RODT.GT.1.1) GO TO 190
       ALMTD=0.5*(HOMCI+HOMCO)
       IF (HETCN.EQ.1.) WRITE (6,670) STSAT,STBI,TB2,LJ,IR,II
       GO TO 200
190    ALMTD=(HOMCI-HOMCO)/ALOG(RODT)
200    CONTINUE
       IF (HETCN.NE.1.) GO TO 210
       M5=10
       WRITE (6,310) M5
       WRITE (6,550) HOMCI,HOMCO
       WRITE (6,560) RCDT,ALMTD
210    IF (WNC.NE.0.) GO TO 230
C$--------------
C HET 11
C                  CALCULATE FINAL HEAT TRANSFER PARAMETERS
C                  FOR NO N/C GASSES ENTERING THE CONDENSER.
C
```

```
C              CONTROL PASSES FROM THIS ROUTINE TO ONE OF TWO PLACES,     CON27850
C              LINE 40 (HET-7) TO START A NEW ITERATION, OR TO            CON27860
C              LINE 170 (HET-13) TO TERMINATE THIS HETTRN CALL            CON27870
C              WHEN ONE OF THE FOLLOWING CRITERIA ARE MET:                CON27880
C                   1) 10 ITERATIONS COMPLETED WITHOUT CONDITION 2 BEING CON27890
C                      MET.                                               CON27900
C                   2) CHANGE IN HEAT TRANSFER COEF .LT. 1 PERCENT        CON27910
C                      BETWEEN SUCCESSIVE ITERATIONS.                     CON27920
C                                                                         CON27930
C         DELFLM = SAVED VALUE OF DLFLM,USED ONLY FOR PRINT               CON27940
C         DGFLM = TEMP DROP ACROSS GAS FILM, SET TO 0 HERE                CON27950
C         DLFLM = TEMP DROP ACROSS LIQUID CONDENSATE FILM                 CON27960
C         K = COUNTER FOR NUMBER OF ITERATIONS                            CON27970
C         RFACT = SUM OF THERMAL RESISTANCES                              CON27980
C         ROUT = EXTERNAL FILM RESISTANCE. (SEE NOTE)                     CON27990
C         OD = TUBE OD                                                    CON28000
C         SHNF = EXTERNAL FILM HEAT X-FER COEF                            CON28010
C         SKBO = THERMAL CONDUCTIVITY OF EXTERNAL LIQUID FILM             CON28020
C         STCO = TEMP AT SURFACE OF OUTER LIQUID FILM                     CON28030
C         STFO = AVE TEMP OF OUTER LIQUID FILM                            CON28040
C         STBAVE = AVE COOLANT TEMP                                       CON28050
C         STSAT = STEAM SATURATION TEMP                                   CON28060
C         STWO = TUBE OUTER WALL TEMP                                     CON28070
C         UEST = VALUE OF UEST FROM LAST ITERATION                        CON28080
C         UESTS = SAVED VALUE OF UEST, USED ONLY FOR PRINT                CON28090
C         UTEST = TUBE HEAT X-FER COEF.                                   CON28100
C         XNU = NUSSELT NUMBER                                            CON28110
C                                                                         CON28120
C                   NOTE:CONSIDER REQUIREMENT FOR MINIMUM OF              CON28130
C                        FOUR ITERATIONS PRIOR TO UPDATING UEST.  COULD  CON28140
C                        BE AN ALLOWANCE FOR DELFLM TO CATCH UP IN        CON28150
C                        CONVERGENCE.  IF SO, TEST DELFLM FOR ITS         CON28160
C                        OWN CONVERGENCE, ELIMINATE MIN OF 4 ITERATIONS. CON28170
C                                                                         CON28180
C                   NOTE:ROUT WILL BE REDEFINED AS 1/ROUT                 CON28190
C                        IN HET-13 PRIOR TO EXIT FROM HETTRN              CON28200
C                                                                         CON28210
C                                                                         CON28220
C ZZZ HET-11                                                              CON28230
C---------------                                                          CON28240
      ROUT=1./SHNF                                                        CON28250
      UTEST=1./RFACT                                                      CON28260
      XNU=SHNF*OD/SKBO                                                    CON28270
      IF (ABS(UEST-UTEST)/UTEST.LT..01) GO TO 290                         CON28280
      UESTSV=UEST                                                         CON28290
      UEST=UTEST                                                          CON28300
      IF (K.GT.4) UEST=0.5*(UTEST+UESTSV)                                 CON28310
      STWO=STSAT-(STSAT-STBAVE)*UTEST/SHNF                                CON28320
```

```
      STCO=STSAT
      STFO=(STCO+STWO)/2.
      DGFLM=0.
      DLFLM=STCO-STWO
      DELFLM=DLFLM
      K=K+1
      IF (HETCN.NE.1.) GO TO 220
      M5=11
      WRITE (6,310) M5
      WRITE (6,570) UTEST,UESTSV
      WRITE (6,580) XNU,ROUT
      WRITE (6,590) STWO,STCO,STFO
      WRITE (6,600) DGFLM,DLFLM
220   IF (K.LT.20) GO TO 120
      WRITE (6,640) UESTSV,UTEST
      GO TO 290
C$--------------
C HET 12
C                 CALCULATE FINAL HEAT TRANSFER PARAMETERS
C                 FOR N/C GASSES ENTERING CONDENSER
C
C            PROGRAM CONTROL COMES HERE FROM HET-10. CONTROL PASSES
C            FROM HERE TO ONE OF TWO PLACES, EITHER LINE 40 (HET-7) TO
C            BEGIN THE NEXT ITERATION, OR TO LINE 170 (HET-13) WHEN
C            ONE OF THE FOLLOWING CONDITIONS ARE MET:
C                 1) 10 ITERATIONS HAVE BEEN COMPLETED PRIOR TO MEETING
C                    CONDITION 2.
C                 2) CHANGES IN UEST .LT. .001, IN DLFLM .LT.
C                    .01 AND IN DGFLM .LT. .001
C
C         ALMTD = LMTD
C         BONE = CONSTANT IN SOLUTION OF EQN 11A, APP A.
C         CJ - SEE BONE
C         CONE - SEE BONE
C         DELFLM = DLFLM FROM PREVIOS ITERATION
C         DELTAU = HEAT DUTY ON HEAT X-FER SURFACE
C         DENOM =
C         DGFLM = TEMP DIFF ACROSS N/C GAS FILM
C         DLFLM = TEMP DIFF ACROSS CONDENSATE FILM
C         DLGFLM = DGFLM FROM PREVIOS ITERATION
C         DTDP = INVERSE OF DPDT (SEE NOTE)
C         HEFF = N/C GAS HEAT X-FER COEF
C         HSTO = STEAM LATENT HEAT OF VAPORIZATION
C         IBT = PRINT CONTROL
C         IR = CURRENT ROW NUMBER
C         K = ITERATION COUNTER
C         LJ = CURRENT SECTOR NUMBER
C         PGB = PARTIAL PRESSURE OF N/C GASSES
```

```
C           RFACT = SUM CF THERMAL RESISTANCES NEGLECTING N/C GASSES      CON28810
C           ROUT = EXTERNAL HEAT X-FER COEF                               CON28820
C           OD = TUBE OD                                                  CON28830
C           SHNF = EXTERNAL LIQUID FILM HEAT X-FER COEF                   CON28840
C           SKBO = THERMAL CONDUCTIVITY OF OUTER LIQUID                   CON28850
C           STBAVE = AVG COOLANT TEMP                                     CON28860
C           STCO = TEMP AT SURFACE OF OUTER LIQUID FILM                   CON28870
C           STFC = OUTER LIQUID FILM AVG TEMP                             CON28880
C           STSAT = STM SATURATION TEMP                                   CON28890
C           STWO = TUBE OUTER WALL TEMP                                   CON28900
C           T = AVG LIQUID FILM TEMP IN RANKIN                            CON28910
C           TDIF = TEMP DIFF BETWEEN SAT STM AND AVE COOLANT TEMP         CON28920
C           TSAT = STM SAT TEMP IN RANKIN                                 CON28930
C           UEST & UESTSV - VALUE OF UTEST AND UEST FROM PREVIOS ITERATION CON28940
C           UTEST = OVERALL HEAT X-FER COEF.                              CON28950
C           XNU = NUSSELT NUMBER                                          CON28960
C                                                                         CON28970
C                                                                         CON28980
C  ZZZ HET-12                                                             CON28990
C---------------                                                          CON29000
230     CONTINUE                                                          CON29010
        M5=12                                                             CON29020
        IF (HETCN.EQ.1.) WRITE (6,310) M5                                 CON29030
        STCO=STSAT-DLGFLM                                                 CON29040
        T=(TSAT+STCO+459.69)/2.0                                          CON29050
        DTDP=1./(((6452.562+2.*837533.2/T)/T**2)*EXP(14.15012-(6452.562+83CON29060
       117533.2/T)/T))                                                    CON29070
        BONE=-1.*(CJ*HSTO/ALMTD+(PGB*DTDP/ALMTD+1.)/RFACT)                CON29080
        CONE=CJ*HSTO/(ALMTD*RFACT)                                        CON29090
        UTEST=(-SQRT(BONE**2-4.*CONE)-BONE)/2.                            CON29100
        DENOM=1.-UTEST*RFACT                                              CON29110
        IF (DENOM.GT.1.E-7) GO TO 240                                     CON29120
        HEFF=1.0E8                                                        CON29130
        GO TO 250                                                         CON29140
240     HEFF=UTEST/DENOM                                                  CON29150
250     CONTINUE                                                          CON29160
        ROUT=(HEFF+SHNF)/(HEFF*SHNF)                                      CON29170
        TDIF=STSAT-STBAVE                                                 CON29180
        IF (TDIF.GT.0) GO TO 260                                          CON29190
        IF (IBT.EQ.0) WRITE (6,660) STSAT,STBAVE,LJ,IR                    CON29200
        IBT=1                                                             CON29210
        TDIF=.1                                                           CON29220
260     CONTINUE                                                          CON29230
        DELTAU=TDIF*UTEST                                                 CON29240
        DGFLM=DELTAU/HEFF                                                 CON29250
        DLFLM=DELTAU/SHNF                                                 CON29260
        STCO=STSAT-DGFLM                                                  CON29270
        STWO=STCO-DLFLM                                                   CON29280
```

```
      STFO=(STCO+STWO)/2.
      XNU=OD/(SKBO*ROUT)
      IF (ABS(UTEST-UEST)/UTEST.LT..001) GO TC 270
      IF (HETCN.EQ.1.) WRITE (6,610) K
      DELFLM=CLFLM
      DLGFLM=DGFLM
      UESTSV=UEST
      UEST=UTEST
      K=K+1
      IF (K.GT.4) UEST=0.5*(UTEST+UESTSV)
      IF (K.LT.10) GO TO 120
      IF (HETCN.EQ.1.) WRITE (6,640) UESTSV,UTEST
      GO TO 290
  270 IF (ABS(DELFLM-DLFLM).LT..01) GO TO 280
      IF (HETCN.EQ.1.) WRITE (6,620) K
      DELFLM=DLFLM
      DLGFLM=DGFLM
      K=K+1
      IF (K.LT.10) GO TO 120
      WRITE (6,650) DELFLM,LJ,IR
  280 IF (ABS(DLGFLM-DGFLM).LT..001) GO TO 290
      DLGFLM=DGFLM
      K=K+1
      IF (K.LT.20) GO TO 120
      WRITE (6,630) DLGFLM,UEST,UTEST,LJ,IR
C$--------------
C HET 13
C                    FINAL CALCULATIONS; EXIT HETTRN
C
C
C         HEFF = SEE XHEFF
C         TB2 = TUBE OUTLET TEMP
C         VNF = OVERAL HEAT X-FER COEF
C         VPSHH = OUTLET COOLANT TEMP
C         XHEFF = N/C GAS FILM HEAT X-FER COEF
C
C                    NOTE: HERE VALUES OF HEFF, UEST, AND TB2 ARE PASSED
C                          OUT OF HETTRN AND SAVED, BUT NOT RE-USED
C                          TO START NEXT ITERATICN.  SEE NOTES IN
C                          HET-2 AND HET-3.
C
C ZZZ HET-13
C--------------
      M5=13
      WRITE (6,310) M5
  290 VNF=UTEST
      ROUT=1./ROUT
      IF (IR.GT.1) GO TO 300
      UEST1=UEST
```

```
      HEFF1=HEFF
      DLFLM1=DLFLM
      DGFLM1=DGFLM
      TB21=TB2
      XNRE1=XNRE
300   CONTINUE
      UESTSV=UEST
      HEFSV=HEFF
      DGFLMS=DGFLM
      DLFLMS=DLFLM
      TB2SV=TB2
      XNRESV=XNRE
      RETURN
C$     TOP OF PAGE CHARACTOR FOR USE WITH $PRT
310   FORMAT (1H0,17HIN HETTERN    -HET,I2)
320   FORMAT (1H ,4HROW=,I4,10H  SECTOR =,I4)
330   FORMAT (1H ,3HOD=,F6.4,6H  TID=,F6.4)
340   FORMAT (1H ,7HSHWINV=,E10.3,8H  STSAT=,E10.3)
350   FORMAT (1H ,4HWST=,F8.2,6H  WNC=,F8.2)
360   FORMAT (1H ,5HUEST=,E10.3,7H  HEFF=,E10.3)
370   FORMAT (1H ,6HDGFLM=,E10.3,8H  DLFLM=,E10.3)
380   FORMAT (1H ,4HTB2=,E10.3,8H  XNRET=,E10.3)
390   FORMAT (1H ,4HBNF=,E10.3,5H  AO=,E10.3,5H  AI=,E10.3)
400   FORMAT (1H ,5HEHNI=,E10.3,7H  EHNO=,E10.3)
410   FORMAT (1H ,5HGMAX=,E10.3,7H  PSAT=,E10.3)
420   FORMAT (1H ,7HAMOLSS=,E10.3,9H  AMOLSC=,E10.3,8H  AMOLT=,E10.3)
430   FORMAT (1H ,6HRMWNC=,E10.3,8H  RMWSC=,E10.3)
440   FORMAT (1H ,5HAVIS=,E10.3,7H  HSTO=,E10.3,7H  XNRE=,E10.3)
450   FORMAT (1H ,7HSTBAVE=,E10.3,7H  STFC=,E10.3,7H  STWC=,E10.3)
460   FORMAT (1H ,7HDELFLM=,E10.3,9H  DLGFLM=,E10.3)
470   FORMAT (1H ,6HCOLBJ=,E10.3,6H  WNC=,E10.3)
480   FORMAT (1H ,4HTB2=,E10.3,9H  STBAVE=,E10.3)
490   FORMAT (1H ,5HSKBO=,E10.3,6H  SKB=,E10.3)
500   FORMAT (1H ,4HBMU=,E10.3,7H  SHBI=,E10.3)
510   FORMAT (1H ,6HXNREB=,E10.3,8H  XNPRB=,E10.3)
520   FORMAT (1H ,4HSHI=,E10.3,6H  RIN=,E10.3,5HSHMK=,E10.3)
530   FORMAT (1H ,6HFDAVN=,E10.3,8H  FDAVM=,E10.3)
540   FORMAT (1H ,5HSHNF=,E10.3,8H  RFACT=,E10.3)
550   FORMAT (1H ,6HHOMCI=,E10.3,8H  HOMCO=,E10.3)
560   FORMAT (1H ,5HRODT=,E10.3,8H  ALMTD=,E10.3)
570   FORMAT (1H ,6HUTEST=,E10.3,8H  ESTSV=,E10.3)
580   FORMAT (1H ,4HXNU=,E10.3,7H  ROUT=,E10.3)
590   FORMAT (1H ,5HSTWO=,E10.3,7H  STCO=,E10.3,7H  STFO=,E10.3)
600   FORMAT (1H ,6HDGFLM=,E10.3,8H  DLFLM=,E10.3)
610   FORMAT (1H ,50HCONVERGENCE NOT YET ACHIEVED ON UEST IN HET-12, K=,
     1I4)
620   FORMAT (1H ,52HCONVERGENCE NOT YET ACHIEVED ON DELFLM IN HET-12, K
     1=,I4)
```

```
630   FORMAT (1H0,36H NO CONVERGENCE IN FILM TEMPERATURES,8H DLGFLM=,E12
     1.5,7H UEST =,E12.5,8H UTEST =,E12.5/10H SECTOR = ,I5,5H ROW=,I5)
640   FORMAT (1H0,32H NO CONVERGENCE IN UESY, UEST = ,E12.5,9H UTEST = ,
     1E12.5)
650   FORMAT (1H0,34H NO CONVERGENCE IN DELFLM, DELFLM=,E12.5,8H SECTOR=
     1,I5,5H ROW=,I5)
660   FORMAT (1H0,53H  SAT STEAM TEMP LESS THAN AVG BRINE TEMP,     STSAT
     1 =,E12.5,10H STBAVE =  ,E12.5,9H   SECTOR,I3,6H   ROW,I4)
670   FORMAT (1H0,47H INPET/OUTLET DELTA T RATIO IS NEAR 1.0, STSAT=,E13
     1.6,7H  STBI=,E13.6,6H  TB2=,E13.6,8H  SECTOR,I3,5H  ROW,2I5)
      END
C
C*********************************************************************
C*********************************************************************
C
      SUBROUTINE COOLEX (TBDRYC,WCNDAV,SBFLOC,WSOUT)
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNCTO
     2T,BNDRAC,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,WNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /ORC1/ AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDOT(10
     10)
      COMMON /OUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNC,VEL2,FR
      COMMON /OUTC/ EXITFC,EXITFR,HTCLR,AOC,SUMQC,SDIC,SDOC,UBARW,SMWBC,
     1SMTBIC,SMTB2C,VWBAR,STSATX,PMIX1,AOT,BUNWT,AVTBI,AVTB2,WTSTC,VOLC
      COMMON /CONST/ AMWNC,CBI,CPB,PI,SG,IFIRST
      COMMON /COOL1/ VELC(100),PMIXC(100),PSATC(100),STSATC(100),WSC(100
     1),ALMTDC(100),SHIC(100),SHNFC(100),RCC(100),ROUTC(100),VNREC(100),
     2VPSHHC(100),HEFFC(100),UNC(100),WCNDC(100),GFLOWC(100),QOAC(100),C
     3UMDPC(100)
      COMMON /COOL/ IVNOC,HNOC,TNOC,TSATEX,PMXEXT,AMLSEX,WSEXIT,VELEXT,T
     1B2C,ENHIC,ENHOC,ENHFC,SHWINC,WBC
      DIMENSICN R2(3),E2OK(3),VPSHC(100)
      DATA HEFF,UEST,DGFLM,DLFLM/2100.,700.,0.05,2.0/
      DATA R2,E2OK/3.617,3.996,3.838,97.0,109.0,140.75/
C
C              ***VARIABLES USED IN COOLER***
C
C
C         ALMTCC = LMTD FOR A ROW IN COOLER
C         AOC = OUTER SURFACE AREA OF A TUBE   FT**2
C         AHOLES = THE X-SECTIONAL AREA OCCUPIED BY THE TUBES
C         AXOC = AREA OPEN TO FLOW IN A ROW OF TUBES  FT**2
C         AMWNC = MOLECULAR WEIGHT OF N/C GASSES
C         DELPC = PRESSURE DROP ACROSS COOLER
```

```
C     DELPCT = PRESSURE LOSS AT COOLER DUE TO VELOCITY CHANGE
C     DELTPC = PRESS DROP ACROSS A ROW OF TUBES  PSI
C     HEFFC = N/C GAS FILM HEAT TRANSFER COEF
C     HFG = STEAM LATENT HEAT OF VAPORIZATION  BTU/LBM
C     HTCLR = THE HEIGHT OF THE COOLER
C     IHNOC = THE NUMBER OF HORIZONTAL TUBES IN THE COOLER
C     IVNOC = THE NUMBER OF VERTICAL TUBES IN THE COOLER
C     PAL = SAME AS PSAT
C     PMIX1 = PRESS OF STM - N/C MIXTURE ENERING COND   PSIA
C     PSATC = SATURATED STEAM PRESS
C     PTLIM = LOWEST SATURATION PRESSURE THE STEAM CAN GO BEFORE THE
C       CORRESPONDING SATURATION TEMPERATURE FALLS BELOW INLET
C       COOLANT TEMPERATURE. THE RESULT IS 0 HEAT TRANSFER
C     RCC = RATIO OF N/C GAS TO STM AT ONE ROW OF TUBES
C     ROUTC = EXTERNAL HEAT TRANSFER COEF
C     SDDC = COOLER TUBE PITCH
C     SDO = TUBE OD  FT
C     SHIC = INTERNAL TUBE FILM HEAT TRANSFER COEF.
C     SHNFC = EXTERNAL TUBE FILM HEAT TRANSFER COEF
C     SMTBIC = ROW BY ROW SUM OF (COOLANT INLET TEMP * FLOW RATE)
C     SMWBC = ROW BY ROW SUM OF COOLANT FLOW
C     SMTB2C = ROW BY ROW SUM OF (COOLANT OUTLET TEMP * FLOW RATE)
C     SUMQC = ROW BY ROW SUM OF HEAT DUTY ON TUBES
C     SVMXEX = ENTERING MIXTURE SPECIFIC VOL  FT**3/LBM
C     SVNCEX = ENTERING N/C GAS SPECIFIC VOL  FT**3/LBM
C     STSATC = STAT STEAM TEMP AT A GIVEN ROW F
C     STSATC = STEAM SATURATION TEMP  F
C     STFO = AVE TEMP OF OUTER TUBE FILM
C     TAL = SATURATED STEAM TEMP  R
C     TNOC = THE NUMBER OF TUBES IN THE COOLER
C     TSAREA = TUBE SHEET AREA
C     UNC = OVERALL HEAT TRANSFER COEF FOR ONE ROW OF TUBES IN COOLER
C     VG = SPECIFIC VOLUME OF STEAM FT**3/LB
C     VMIX = SPECIFIC VOLUME OF STEAM AND N/C GASSES IN MIXTURE
C     VNREC = REYNOLDS NUMBER ON VAPOR SIDE AT A ROW IN COOLER
C     VNC = SPECIFIC VOLUME OF N/C GAS FT**3/LB
C     VOLC = COOLER VOLUME
C     VPSHHC = OUTLET COOLANT TEMP  F
C     WCNDC = CONDENSATE FROM ONE ROW OF TUBES IN COOLER  LBM/HR
C     WS = STEAM FLOW TO A ROW OF TUBES IN A SECTOR LB/HR
C     WSC = STEAM FLOW TO A ROW OF COOLER   LBM/HR
C     WNCC = N/C FLOW TO A ROW OF COOLER   LBM/HR
C     WSEXIT = TOTAL STM FLOW TO COOLER LBM/HR
C
      WNCC=WNCIR
      SDOC=ODII/12.
      SDO=SDOC
      SDDMIN=SDDC
```

```
      TB2=TB2C                                                          CON31210
      AOC=PI*ALST*SDO                                                   CON31220
      AO=AOC                                                            CON31230
      WB=WBC                                                            CON31240
      FDAVE=0.5                                                         CON31250
      SDIC=SID(NOROWS)                                                  CON31260
      SDI=SDIC                                                          CON31270
      AI=PI*ALST*SDI                                                    CON31280
      SHWINV=SHWINC                                                     CON31290
      ENHI=ENHIC                                                        CON31300
      ENHO=ENHOC                                                        CON31310
      ENHF=ENHFC                                                        CON31320
      STBIR=STBI+459.69                                                 CON31330
      PTLIM=PSATFN(STBIR)                                               CON31340
      SMTBIC=0.0                                                        CON31350
      SMWBC=0.0                                                         CON31360
      SMTB2C=0.0                                                        CON31370
      SUMQC=0.0                                                         CON31380
      WTSTC=0.                                                          CON31390
      IROWC=0                                                           CON31400
      WSOUT=0.                                                          CON31410
      SBFLOC=0.                                                         CON31420
      SBRATC=0.                                                         CON31430
      TBDRYC=0.                                                         CON31440
      AMLSCC=WNCC/AMWNC                                                 CON31450
      AXOC=SDC*HNOC*ALST*(SDDMIN-1.0)                                   CON31460
      SVGEXT=VGFN(TSATEX,PMXEXT)                                        CON31470
      SVNCEX=10.729*TSATEX/(AMWNC*(PMXEXT))                             CON31480
      AMLT=AMLSEX+AMLSCC                                                CON31490
      SVMXEX=1.0/(AMLSEX/(AMLT*SVGEXT)+AMLSCC/(AMLT*SVNCEX))            CON31500
      VELC(1)=(WSEXIT+WNCC)*SVMXEX/(AXOC*3600.0)                        CON31510
C                                                                       CON31520
      VLCMAX=VELC(1)                                                    CON31530
      G2=288.0*SG*SVMXEX                                                CON31540
      IF (VELC(1)-VELEXT) 10,10,20                                      CON31550
10    A2DA1=VELC(1)/VELEXT                                              CON31560
      DELPCT=((1.0-A2DA1)*VELEXT**2)/G2                                 CON31570
      GO TO 50                                                          CON31580
20    A2DA1=VELEXT/VELC(1)                                              CON31590
      IF (A2DA1-0.715) 40,40,30                                         CON31600
30    DELPCT=(0.75*(1.0-A2DA1)*VELC(1)**2)/G2                           CON31610
      GO TO 50                                                          CON31620
40    DELPCT=(0.4*(1.25-A2DA1)*VELC(1)**2)/G2                           CON31630
50    PMIXC(1)=PMXEXT-DELPCT-(VELC(1)**2-VELEXT**2)/G2                  CON31640
      PSATC(1)=PMIXC(1)*AMLSEX/(AMLSEX+AMLSCC)                          CON31650
C                                                                       CON31660
C     RECALCULATE THE AREA RATIO (THE RATIO OF TUBE X-SECTIONAL         CON31670
C     AREA TO TTUBE SHEET AREA) TO INCLUDE THE COOLER                   CON31680
```

```
C
      SECFLG=FLOAT(ISEC)
      TSAREA=PI*(BNDRAD**2)*SECFLG*SECWID/360.
      AHOLES=ARATIO*TSAREA
      AHOLES=AHOLES+(PI*SDO**2)*TNOC/4.
      TSAREA=TSAREA+HTCLR*HNOC*SDDC*SDOC
      ARATIO=AHOLES/TSAREA
C
C     CALCULATE THE CCOLER VOLUME
C
      VOLC=HTCLR*SDDC*SDOC*HNOC*ALST
C
C    CHECK TO SEE IF INITIAL PRESSURE LOSSES DROP THE PRESSURE
C    BELOW PTLIM
C
      IF (PSATC(1).GT.PTLIM) GO TO 80
      SUMQC=0.
      NRWSP4=IVNOC+1
      CONS1A=PTLIM/FLOAT(NRWSP4)
      CONS1B=PMXEXT/FLOAT(NRWSP4)
      CUMDPC(1)=PMIX1-PMXEXT
      PSATC(1)=PTLIM
      PMIXC(1)=PMXEXT
      QOAC(1)=0.
      DELPC=PMXEXT
      UNC(1)=700.
      WSC(1)=WSEXIT
      WCNDC(1)=0.
C
      DO 60 M=2,NRWSP4
      WSC(M)=WSEXIT
      WCNDC(M)=0.
      PMIXC(M)=PMIXC(M-1)-CONS1B
      PSATC(M)=PSATC(M-1)-CONS1A
      CUMDPC(M)=PMIX1-PMIXC(M)
60    CONTINUE
C
      DO 70 M=2,IVNOC
      UNC(M)=700.
      QOAC(M)=0.
      ALMTDC(M)=0.
70    CONTINUE
      SMWBC=SMWBC+WB*TNOC
      SMTBIC=WB*TNOC*STBI
      SMTB2C=SMTBIC
      EXITFC=1.
      RETURN
C
```

```
80    PAL=PSATC(1)
      TAL=TSATFN(PAL)
      STSATC(1)=TAL-459.69
      WSC(1)=WSEXIT
C
C     CHECK TO SEE IF INLET STEAM TO COOLER IS 0
C
      IF (WSC(1).GT.1.) GO TO 90
C
C     ENTER DUMMY VALUES IN COOLER VARIABLES
C     CREATE NEGATIVE FLOW PARAMETER FOR USE IN COMPUTING EXITFRC
C
      TBDRYC=TNOC
      SBRATC=WCNDAV
      SBFLOC=SBRATC*TBDRYC
C
C
      WSC(1)=1.
      IROWC=3
C
90    VG=VGFN(TAL,PMIXC(1))
      AMLSSC=WSC(1)/18.015
      VNC=10.729*TAL/(AMWNC*PMIXC(1))
      VMIX=1.0/((AMLSSC/(VG*(AMLSSC+AMLSCC)))+(AMLSCC/(VNC*(AMLSSC+AMLSC
     1C))))
      VELC(1)=(WSC(1)+WNCC)*VMIX/(AXOC*3600.0)
      NFC=(VNCC+1.)/2.0
      ANFC=NFC
      IDT=0
      JRC=0
      IBT=0
      DO 450 IK=1,IVNOC
      I=IK
      LJ=7
C
C     ROW BY ROW CALCULATION A LA HETTRN
C
      STSAT=STSATC(I)
      ANF=ANFC
      WS=WSC(I)
      WNC=WNCC
      AXO=AXOC
      IR=I
C     DATA HEFF,UEST,DGFLM,DLFLM/2100.,700.,0.05,2.0/
C     DATA R2,E20K / 3.617,3.996,3.838,97.0,109.0,140.75/
      DATA R2/3.617,3.996,3.838/
      K=0
      RC=WNC/(WS+WNC)
```

```
      TSAT=STSAT+459.69                                                 CON32650
      GMAX=(WS+WNC)/AXO                                                 CON32660
      PSAT=PSATFN(TSAT)                                                 CON32670
      AMOLSS=WS/18.015                                                  CON32680
      AMOLSC=WNC/AMWNC                                                  CON32690
      AMOLT=AMOLSS+AMOLSC                                               CON32700
      AMWAV=(WS+WNC)/AMOLT                                              CON32710
      RMWNC=SQRT(AMWNC)*AMOLSC/AMOLT                                    CON32720
      RMWSC=4.24444*AMOLSS/AMOLT                                        CON32730
      AVIS=(SMUFN(TSAT)*RMWSC+AMUFN(TSAT,JGAS)*RMWNC)/(RMWSC+RMWNC)     CON32740
      HSTO=HFGFN(STSAT)                                                 CON32750
      BB=WB/(PI*SDI*SDI/4.)                                             CON32760
      STBAVE=(STBI+TB2)/2.0                                             CON32770
      DELFLM=DLFLM                                                      CON32780
      DLGFLM=DGFLM                                                      CON32790
      STWO=STSAT-DELFLM-DLGFLM                                          CON32800
      STFO=(STSAT+STWO)/2.0                                             CON32810
      XNRE=SDO*(WS+WNC)/(AXO*3600.*AVIS)                                CON32820
      IF (XNRE.GE.100.) GO TO 100                                       CON32830
100   CONTINUE                                                          CON32840
      COLBJ=EXP(0.53883-0.544*ALOG(XNRE))                               CON32850
      IF (WNCIR.EQ.0.) GO TO 110                                        CON32860
      PATM=PSAT/14.7                                                    CON32870
      PGB=PSAT*(AMOLSC/AMOLSS)                                          CON32880
      TSATK=TSAT/1.8                                                    CON32890
      CALL DFSVTY (TSATK,PATM,18.015,AMWNC,2.655,R2(JGAS),363.,E2OK(JGASCON32900
     1),0.,0.,DG,DGG)                                                   CON32910
C*****                                                                  CON32920
C   0.258 CONVERTS CM2/S TO FT2/HR AND 3600. CONVERTS VISCOSITY TO      CON32930
C   LB/F-HR FROM LB-F-S.          0.258 * 3600. = 928.8                 CON32940
C*****                                                                  CON32950
      XNSCH=(AVIS*TSAT*10.73/((PSAT+PGB)*AMWAV*DG))*928.8               CON32960
      CJ=COLBJ*GMAX/(XNSCH**0.666667)                                   CON32970
110   TB2=STSAT-(STSAT-STBI)/EXP(UEST*AO/(WB*CPFN(CBI,STBAVE)))         CON32980
      STBAVE=(STBI+TB2)/2.                                              CON32990
      SKBO=SKBFN(0.,STFO)                                               CON33000
      SKB=SKBFN(CBI,STBAVE)                                             CON33010
      BMU=BMUFN(CBI,STBAVE)                                             CON33020
      SHBI=CPFN(CBI,STBAVE)                                             CON33030
      XNREB=SDI*BB/BMU                                                  CON33040
      XNPRB=SHBI*BMU/SKB                                                CON33050
      SHI=0.023*(XNREB)**0.8*XNPRB**0.33333*(SKB/SDI)*ENHI              CON33060
      RIN=AO/(SHI*AI)                                                   CON33070
      BNF=ANF                                                           CON33080
      IF (ANF.LE.1.0) BNF=1.0                                           CON33090
      SHMK=0.725*(SKBO**3*ROEFN(0.0,STFO)**2*HFGFN(STFO)*416975040.0/(BMCON33100
     1UFN(0,STFO)*SDO*DELFLM))**.25*ENHO                                CON33110
C                                                                       CON33120
```

```
C THIS IS THE PLACE TO ENTER CORRECTIONS TO EXTERNAL FILM COEFFICIENT
C FOLLOWING IS A CORRECTION ACCORDING TO FUJII, ET. AL, TO ACCOUNT FOR
C EFFECTS OF VAPOR SHEAR AT HIGH VAPOR REYNOLDS NUMBERS
C
      SHMK1=SHMK
      ANUS=SHMK*SDO/SKBO
      EMUL=BMUFN(0.,STSAT)
      EMUS=SMUFN(TSAT)*3600.
      RHOL=RCEFN(0.,STSAT)
      RHOS=1./VGFN(TSAT,PSAT)
      VISRAT=(EMUS/RHOS)/(EMUL/RHOL)
      RETP=XNRE*VISRAT
C
      IF (RETP.LT.0.) RETP=0.
      IF (RETP.LT.0.) GO TO 120
C
120   TERM=SQRT(RETP)/ANUS
      IF (TERM.LT..278) GO TO 130
      ANUSP=1.24*(ANUS**0.8)*(RETP**0.1)
      GO TO 140
130   ANUSP=ANUS*0.70/0.725
140   SHMKP=ANUSP*SKBC/SDO
      SHMK=SHMKP
C
C
C###################################################################
C#                                                                 #
C#          IF HO > 40,000                                         #
C#                                                                 #
C###################################################################
C     IF (SHMK .GT. 40000.)  GO TO 49
C     IF (SHMK.LT.40000..AND.DELFLM.GT..0001) GO TO 49
C  49 CONTINUE
C
C       INUNDATION EFFECT MAY BE BYPASSED BY SETTING BNF = 1.
C
C
      BNF = 1.
      IF (FDAVE.LT.0..OR.BNF.LT.2.) GO TO 150
      FDAVN=0.6*FDAVE+(1.-0.5647*FDAVE)/BNF**.20
      FDAVM=0.6*FDAVE+(1.-0.5647*FDAVE)/(BNF-1)**.20
      SHNF=SHMK*(BNF*FDAVN-(BNF-1.)*FDAVM)
      GO TO 160
150   SHNF=0.95*(BNF**0.9-(BNF-1.)**0.9)*SHMK
C
160   RFACT=RIN+SHWINV+FOUL+1./SHNF
C
C    COMPUTE LOG MEAN DELTA TEMP., ALMTD
C
```

```
C*****                                                                   CON33610
      HOMCI=STSAT-STBI                                                   CON33620
      HIMCO=STSAT-TB2                                                    CON33630
C                                                                        CON33640
C     IF (HIMCO.LT.1.E-15) WRITE (6,530) HOMCI,HIMCO,LJ,IR               CON33650
      IF (HIMCO.LT.1.E-15) HIMCO=1.E-15                                  CON33660
C                                                                        CON33670
      RODT=HOMCI/HIMCO                                                   CON33680
C                                                                        CON33690
C     IF INLET/OUTLET DELTA T RATIO IS NEAR 1.0, ALMTD = ARITH AVE       CON33700
      IF (RODT.GT.1.1) GO TO 170                                         CON33710
      ALMTD=0.5*(HOMCI+HIMCO)                                            CON33720
C     IF (JRC.EQ.0) WRITE (6,540) STSAT,STBI,TB2,LJ,IR                   CON33730
      JRC=1                                                              CON33740
      GO TO 180                                                          CON33750
170   ALMTD=(HOMCI-HIMCO)/ALOG(RODT)                                     CON33760
180   CONTINUE                                                           CON33770
      IF (WNCIR.NE.0.) GO TO 190                                         CON33780
      ROUT=1./SHNF                                                       CON33790
      UTEST=1./RFACT                                                     CON33800
      XNU=SHNF*SDO/SKBO                                                  CON33810
      IF (ABS(UEST-UTEST)/UTEST.LT..001) GO TC 280                       CON33820
      UESTSV=UEST                                                        CON33830
      UEST=UTEST                                                         CON33840
      IF (K.GT.4) UEST=0.5*(UTEST+UESTSV)                                CON33850
      STWO=STSAT-(STSAT-STBAVE)*UTEST/SHNF                               CON33860
      STCO=STSAT                                                         CON33870
      STFO=(STCO+STWO)/2.                                                CON33880
      DGFLM=0.                                                           CON33890
      DLFLM=STCO-STWO                                                    CON33900
      DELFLM=DLFLM                                                       CON33910
      K=K+1                                                              CON33920
      IF (K.LT.20) GO TO 110                                             CON33930
      WRITE (6,510) UESTSV,UTEST                                         CON33940
      GO TO 280                                                          CON33950
190   CONTINUE                                                           CON33960
      STCO=STSAT-DLGFLM                                                  CON33970
      T=(TSAT+STCO+459.69)/2.0                                           CON33980
      DTDP=1./(((6452.562+2.*837533.2/T)/T**2)*EXP(14.15012-(6452.562+83CON33990
     17533.2/T)/T))                                                      CON34000
      BONE=-1.*(CJ*HSTO/ALMTD+(PGB*DTDP/ALMTD+1.)/RFACT)                 CON34010
      CONE=CJ*HSTO/(ALMTD*RFACT)                                         CON34020
      TTERM1=BONE**2                                                     CON34030
      TTERM2=4.*CONE                                                     CON34040
      TTERM=TTERM2/TTERM1                                                CON34050
      IF (TTERM.LT.0.999999) GO TO 200                                   CON34060
      UTEST=-BONE/2.                                                     CON34070
      GO TO 220                                                          CON34080
```

```
200   IF ((4.*CONE).GT.(BONE**2)) UTEST=0.
      IF ((4.*CONE).GT.(BONE**2)) GO TO 210
      UTEST=(-SQRT(BONE**2-4.*CONE)-BONE)/2.
210   DENOM=1.-UTEST*RFACT
220   CONTINUE
      IF (DENOM.GT.1.E-7) GO TO 230
      HEFF=1.0E8
      GO TO 240
230   HEFF=UTEST/DENOM
240   CONTINUE
      ROUT=(HEFF+SHNF)/(HEFF*SHNF)
      TDIF=STSAT-STBAVE
      IF (TDIF.GT.0) GO TO 250
      IF (IBT.EQ.0) WRITE (6,530) STSAT,STBAVE,LJ,IR
      IBT=1
      TDIF=.1
250   CONTINUE
      DELTAU=TDIF*UTEST
      DGFLM=DELTAU/HEFF
      DLFLM=DELTAU/SHNF
      STCO=STSAT-DGFLM
      STWO=STCO-DLFLM
      STFO=(STCO+STWO)/2.
      XNU=SDO/(SKBO*ROUT)
      IF (ABS(UTEST-UEST)/UTEST.LT..001) GO TO 260
      DELFLM=DLFLM
      DLGFLM=DGFLM
      UESTSV=UEST
      UEST=UTEST
      K=K+1
      IF (K.GT.4) UEST=0.5*(UTEST+UESTSV)
      IF (K.LT.20) GO TO 110
      WRITE (6,510) UESTSV,UTEST
      GO TO 280
260   IF (ABS(DELFLM-DLFLM).LT..01) GO TO 270
      DELFLM=DLFLM
      DLGFLM=DGFLM
      K=K+1
      IF (K.LT.20) GO TO 110
      WRITE (6,520) DELFLM,LJ,IR
270   IF (ABS(DLGFLM-DGFLM).LT..001) GO TO 280
      DLGFLM=DGFLM
      K=K+1
      IF (K.LT.20) GO TO 110
      WRITE (6,500) DLGFLM,UEST,UTEST,LJ,IR
280   VNF=UTEST
      VPSHH=TB2
      XHEFF=HEFF
```

```
      ROUT=1./ROUT
      UNC(I)=VNF
      ALMTDC(I)=ALMTD
      SHIC(I)=SHI
      SHNFC(I)=SHNF
      RCC(I)=RC
      ROUTC(I)=ROUT
      VNREC(I)=XNRE
      VPSHHC(I)=VPSHH
      HEFFC(I)=SHEFF
C
C    COMPLETICN OF TUBE H/T CALCULATION
C
      HFG=HFGFN(STSATC(I))
      WCNDC(I)=UNC(I)*AO*HNOC*ALMTDC(I)/HSTO
      CALL PRSDRP (TAL,VMIX,WSC(I),WNCC,AXOC,SDO,VPSHC(I),DELTPC,ENHF)
      JY=0
290   WSC(I+1)=WSC(I)-WCNDC(I)
      GFLOWC(I)=(WSC(I)+WNCC)/AXOC
C
C     CHECK FOR NEGATIVE STEAM FLOW
C
      IF (WSC(I+1)-1) 300,300,360
C
C     IF PREVIOUS COND. RATE PREDICTS STEAM FLOW WILL GO NEG AFTER THE
C     NEXT ROW, SET CONDENSATE EQUAL TO STEAM FLOW.
C
300   WCNDC(I)=WSC(I)
C
C     IF STEAM FLOW HAS ALREADY GONE NEGATIVE THEN AVOID CALCULATING
C     TEST VARIABLE
C
      IF (IROWC.GT.0) GO TO 310
C
C     CREATE A TEST VARIABLE INDICATING POINT IN COOLER WHERE STEAM
C     BAD AND A NEGATIVE FLOW RATE FOR USE IN CALCULATING EXITFRC
C
      WTSTC=(WSC(I+1)-1.)/(WSC(I)-WSC(I+1)+1.)
      WTSTC=WTSTC-FLOAT(IVNOC-I)
      IROWC=1
C
      NRODRY=IVNOC-I
      TBDRYC=TNOC*(FLOAT(NRODRY)/FLOAT(IVNOC))
      TBDRYC=TBDRYC-WTSTC*HNOC
      SBRATC=WCNDC(I)/HNOC
      SBFLOC=SBRATC*TBDRYC
C
C
```

```
310   STSATC(I+1)=STFC
      TSATMN=STSATC(I+1)+459.69
      PMIN=PSATFN(TSATMN)
C
C     ENTER DUMMY VALUE FOR NEXT ROW WSC
C
      WSC(I+1)=1.
C
      QGC=(WSC(I)*CPSFN(TAL)/18.015+WNCC*CPAFN(TAL,JGAS)/AMWNC)*(TAL-TSA
     1TMN)
      QTC=QGC+WCNDC(I)*HFG
      IF (STSATC(I)-TB2) 320,320,330
320   TB2=STSATC(I)-0.1
330   CONTINUE
      IF (STSATC(I+1).GT.STBI) GO TO 340
      ALMTDC(I)=ALMTDC(I-1)
      GO TO 350
340   CONTINUE
      ALMTDC(I)=((STSATC(I+1)-STBI)-(STSATC(I)-TB2))/ALOG((STSATC(I+1)-S
     1TBI)/(STSATC(I)-TB2))
350   CONTINUE
      UNC(I)=QTC/(AO*HNOC*ALMTDC(I))
      AMLSSC=WSC(I+1)/18.015
      PMIXC(I+1)=PMIXC(I)-DELTPC
      PSATC(I+1)=(PMIXC(I+1)*AMLSSC)/(AMLSSC+AMLSCC)
      TSATC=TSATMN
      GO TO 440
C
C
360   AMLSSC=WSC(I+1)/18.015
      PMIXC(I+1)=PMIXC(I)-DELTPC
      PSATC(I+1)=(PMIXC(I+1)*AMLSSC)/(AMLSSC+AMLSCC)
C
C     CHECK TO SEE IF STEAM CORRECTIONS FOR DRY TUBES DROVE PSATC TOO LO
C
      IF ((IROWC.GT.0).AND.(PSATC(I+1).LE.PTLIM)) WRITE (6,560) I
      IF ((IROWC.GT.0).AND.(PSATC(I+1).LE.PTLIM)) PSATC(I+1)=PSATC(I)
C
C     CHECK FOR PRESSURE DROPPING BELOW PTLIM
C
      IF (PSATC(I+1).GT.PTLIM) GO TO 390
C
C     CALCULATE PARAMETERS FOR CURRENT ROW
C
      PSATC(I)=PTLIM
      CUMDPC(I)=PMIX1-PMIXC(I)
      SUMQC=SUMQC+UNC(I)*AO*ALMTDC(I)*HNOC
      QOAC(I)=UNC(I)*ALMTDC(I)
```

```
      SMWBC=SMWBO+WB*HNOC                                               CON35530
      SMTBIC=SMTBIC+WB*HNOC*STBI                                        CON35540
      SMTB2C=SMTB2C+WB*HNOC*TB2                                         CON35550
C                                                                       CON35560
      NRWSP4=IVNOC+1                                                    CON35570
      RTG=FLOAT(NRWSP4-I)                                               CON35580
      NRWSP5=I+1                                                        CON35590
      CONST=PMIXC(I)/RTG                                                CON35600
      CONST2=PTLIM/RTG                                                  CON35610
C                                                                       CON35620
C     CALCULATE PARAMETERS FOR THE REMAINING ROWS                       CON35630
C                                                                       CON35640
      DO 370 M=NRWSP5,NRWSP4                                            CON35650
      QOAC(M)=0.                                                        CON35660
      WSC(M)=WSC(M-1)                                                   CON35670
      WCNDC(M)=0.                                                       CON35680
      STSATC(M)=STSATC(M-1)                                             CON35690
      PSATC(M)=PSATC(M-1)-CONST2                                        CON35700
      PMIXC(M)=PMIXC(M-1)-CONST                                         CON35710
      VELC(M)=VELC(M-1)                                                 CON35720
  370 CONTINUE                                                          CON35730
C                                                                       CON35740
      DO 380 IFX=NRWSP5,IVNOC                                           CON35750
      UNC(IFX)=UNC(IFX-1)                                               CON35760
      CUMDPC(IFX)=PMIX1-PMIXC(IFX-1)                                    CON35770
      SMTBIC=SMTBIC+WB*HNOC*STBI                                        CON35780
      SMTB2C=SMTBIC                                                     CON35790
      QOAC(IFX)=0.                                                      CON35800
      SMWBC=SMWBC+WB*HNOC                                               CON35810
  380 ALMTDC(IFX)=0.                                                    CON35820
C                                                                       CON35830
      DELPC=PMIXC(1)-PMIXC(NRWSP4)                                      CON35840
C                                                                       CON35850
      GO TO 460                                                         CON35860
C                                                                       CON35870
  390 PAL=PSATC(I+1)                                                    CON35880
      IF (PAL.GT.0.) GO TO 400                                          CON35890
      WRITE (6,550) PMIXC(I),I                                          CON35900
  400 CONTINUE                                                          CON35910
      TSATC=TSATFN(PAL)                                                 CON35920
      STSATC(I+1)=TSATC-459.69                                          CON35930
      IF (ABS(TSATC-TAL).LT.0.001) TSATC=TAL-0.001                      CON35940
      WCNDP=(UNC(I)*AO*HNOC*ALMTDC(I)-(WSC(I)*CPSFN(TSATC)/18.015+WNCC*CCON35950
     1PAFN(TSATC,JGAS)/AMWNC)*(TAL-TSATC))/HFG                          CON35960
      IF (ABS(WCNDP/WCNDC(I)-1.0)-.005) 430,410,410                     CON35970
  410 WCNDC(I)=WCNDP                                                    CON35980
      JY=JY+1                                                           CON35990
      IF (JY-50) 290,290,420                                            CON36000
```

```
420   WRITE (6,540) I
430   CONTINUE
      WCNDC(I)=WCNDP
      WSC(I+1)=WSC(I)-WCNDC(I)
      AMLSSC=WSC(I+1)/18.015
      PSATC(I+1)=(PMIXC(I+1)*AMLSSC)/(AMLSSC+AMLSCC)
      PAL=PSATC(I+1)
      TSATC=TSATFN(PAL)
      STSATC(I+1)=TSATC-459.69
440   VG=VGFN(TSATC,PMIXC(I+1))
      VNC=10.729*TSATC/(AMWNC*PMIXC(I+1))
      VMIX=1.0/((AMLSSC/(VG*(AMLSSC+AMLSCC)))+(AMLSCC/(VNC*(AMLSSC+AMLSC
     1C))))
      VELC(I+1)=(WSC(I+1)+WNCC)*VMIX/(AXOC*3600.0)
      SMTBIC=SMTBIC+WB*HNOC*STBI
      SMWBC=SMWBO+WB*HNOC
      SMTB2C=SMTB2C+WB*HNOC*TB2
      SUMQC=SUMQC+UNC(I)*AO*ALMTDC(I)*HNOC
      QOAC(I)=UNC(I)*ALMTDC(I)
      CUMDPC(I)=PMIX1-PMIXC(I)
      TAL=TSATC
      ANFC=ANFC-1.0
450   CONTINUE
      DELPC=PMIXC(1)-PMIXC(IVNOC+1)
460   CONTINUE
C
C     CHECK TO SEE IF INLET STEAM FLOW WAS 0
C
      IF (IROWC.LT.2) GO TO 480
      NRWSP4=IVNOC+1
      SUMQC=0.
      DO 470 M=1,NRWSP4
      QOAC(M)=0.
      ALMTDC(M)=0.
      WSC(M)=0.
470   WCNDC(M)=0.
      EXITFC=-999
      WTSTC=FLOAT(IVNOC)
      RETURN
C
C     IF COOLER TUBES WENT DRY CREATING A NEGATIVE EXITFC
C
480   IF (WTSTC.GE.0.) GO TO 490
      EXITFC=-(SBFLOC)/WSC(1)
      RETURN
C
490   EXITFC=WSC(I+1)/WSC(1)
      WSOUT=WSC(I+1)
```

```
C
      RETURN
C
500   FORMAT (1H0,36H NO CONVERGENCE IN FILM TEMPERATURES,8H DLGFLM=,E12
     1.5,7H UEST =,E12.5,8H UTEST =,E12.5/10H SECTOR = ,I5,5H ROW=,I5)
510   FORMAT (1H0,32H NO CONVERGENCE IN UESY, UEST = ,E12.5,9H UTEST = ,
     1E12.5)
520   FORMAT (1H0,34H NO CONVERGENCE IN DELFLM, DELFLM=,E12.5,8H SECTOR=
     1,I5,5H ROW=,I5)
530   FORMAT (1H0,53H  SAT STEAM TEMP LESS THAN AVG BRINE TEMP,    STSAT
     1 =,E12.5,10H STBAVE = ,E12.5,9H   SECTOR,I3,6H   ROW,I4)
540   FORMAT (1H0,25HNO CONVERGENCE FOR WCNDC(,I2,2H).)
550   FORMAT (1H0,68H**** SATURATED STEAM PRESSURE IN COOLER HAS GONE NE
     1GATIVE,   PMIXC =,E15.5,6H   ROW,I5)
560   FORMAT (1X,52HERROR,COOLER PTLIM VIOLATED DUE TO STEAM CORRECTIONS
     1/1X,41HFOR NEGATIVE FLOW. PTLIM VIOLATED IN ROW ,I5/)
      END
C
C*********************************************************************
C*********************************************************************
C
      SUBROUTINE OUT2
C--------------
C OUT2 1
C                       SUMMARY OUTPUT
C ZZZ OUT2
C--------------
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTO
     2T,BNDRAD,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,WNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /ORC1/ AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDOT(10
     10)
      COMMON /ORC2/ ANT(100,15),STB2ES,VW(100)
      COMMON /OUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNC,VEL2,FR
      COMMON /OUTC/ EXITFC,EXITFR,HTCLR,ACC,SUMQC,SDIC,SDOC,UBARW,SMWBC,
     1SMTBIC,SMTB2C,VWBAR,STSATX,PMIX1,AOT,BUNWT,AVTBI,AVTB2,WTSTC,VOLC
      COMMON /CONST/ AMWNC,CBI,CPB,PI,SG,IFIRST
      COMMON /SEC/ ALMTD(100,15),CUMDP(100,15),DELP(15),GFLW(100,15),HEF
     1F(100,15),PMIX(100,15),PSAT(100,15),QOA(100,15),RC(100,15),ROUT(10
     20,15),SHI(100,15),SHN(100,15),STSAT(100,15),UN(100,15),VEL(100,15)
     3,VNRE(100,15),VPSHH(100,15),WCND(100,15),WGAS(15),WP(15),WS(100,15
     4),WSP(15)
      COMMON /COOL1/ VELC(100),PMIXC(100),PSATC(100),STSATC(100),WSC(100
     1),ALMTDC(100),SHIC(100),SHNFC(100),RCC(100),ROUTC(100),VNREC(100),
```

```
     2VPSHHC(100),HEFFC(100),UNC(100),WCNDC(100),GFLOWC(100),QOAC(100),CCON36970
     3UMDPC(100)                                                          CON36980
      COMMON /COOL/  IVNOC,HNOC,TNOC,TSATEX,PMXEXT,AMLSEX,WSEXIT,VELEXT,TCON36990
     1B2C,ENHIC,ENHOC,ENHFC,SHWINC,WBC                                    CON37000
      COMMON /OUT2A/  RCORR,RSPF,PHPCON,VOL1C                             CON37010
      DIMENSION AWTST(15)                                                 CON37020
      DELPVE=0.                                                           CON37030
      AOT=0.                                                              CON37040
      VEL3=0.                                                             CON37050
      CAWT=0.                                                             CON37060
      UBARW=0.                                                            CON37070
      VWBAR=0.                                                            CON37080
      SECFLG=FLOAT(ISEC)                                                  CON37090
      DO 10 I=1,NOROWS                                                    CON37100
      OD=ODOI-DELOD*FLOAT(I-1)                                            CON37110
      CAWT=CAWT+(OD*OD/144.-SID(I)*SID(I))*TBNPR(I)                       CON37120
      AO=3.14159*OD*ALST*TBNPR(I)/12.                                     CON37130
      VWBAR=VWBAR+VW(I)                                                   CON37140
   10 AOT=AOT+AO                                                          CON37150
      BUNWT=PI*CAWT*ALST*TUBESW*SECFLG/4.                                 CON37160
      AOT=AOT*SECFLG                                                      CON37170
      DO 30 J=1,ISEC                                                      CON37180
      DO 20 I=1,NOROWS                                                    CON37190
   20 UBARW=UBARW+UN(I,J)*TBNPR(I)                                        CON37200
      VEL3=VEL3+VEL(NOROWS,J)                                             CON37210
      DELPVE=DELPVE+DELP(J)                                               CON37220
   30 CONTINUE                                                            CON37230
      UBARW=UBARW/TNO                                                     CON37240
      VWBAR=VWBAR/FLOAT(NOROWS)                                           CON37250
      VEL3=VEL3/SECFLG                                                    CON37260
      DELPVE=DELPVE/SECFLG                                                CON37270
      ADTCND=SUMQ/(UBARW*AOT)                                             CON37280
      STSATX=TSATEX-459.69                                                CON37290
      TDROP1=STSAT(1,1)-STSATX                                            CON37300
C                                                                         CON37310
      RSPI=RSPF*12.                                                       CON37320
C                                                                         CON37330
      BNDIAM=BNDRAD*2.                                                    CON37340
      BL=ALST/2.                                                          CON37350
      VOIDID=RADINS*2.                                                    CON37360
      AVTBI=SMTBI/SMWB                                                    CON37370
      AVTB2=SMTB2/SMWB                                                    CON37380
      HDLOSS=DELWP*144./62.366                                            CON37390
C                                                                         CON37400
C     THERE IS THE POSSIBILITY THAT THE AVERAGE INLET COOLANT TEMP        CON37410
C     (AVTBI) EQUALS THE AVERAGE OUTLET COOLANT TEMP (AVTB2). WHEN        CON37420
C     THE CONDENSER IS COMNFIGURED SUCH THAT NO HEAT IS REMOVED FROM      CON37430
C     THE STEAM                                                           CON37440
```

```
C                                                                       CON37450
      IF (AVTBI.EQ.AVTB2) DTCND2=0.                                     CON37460
      IF (AVTBI.EQ.AVTB2) UPCOND=UBARW                                  CON37470
      IF (AVTBI.EQ.AVTB2) GO TO 40                                      CON37480
C                                                                       CON37490
      DTCND2=(AVTB2-AVTBI)/ALOG((STSAT1-AVTBI)/(STSAT1-AVTB2))          CON37500
      UPCOND=SUMQ/(AOT*DTCND2)                                          CON37510
40    WRITE (6,60)                                                      CON37520
      WRITE (6,80)                                                      CON37530
      WRITE (6,90)                                                      CON37540
      WRITE (6,100)                                                     CON37550
      WRITE (6,90)                                                      CON37560
      WRITE (6,80)                                                      CON37570
      WRITE (6,70)                                                      CON37580
      WRITE (6,110) UBARW,ADTCND,DELPVE,TDROP1,VEL2,VEL3                CON37590
      WRITE (6,120) SUMQ                                                CON37600
      WRITE (6,130) WSI                                                 CON37610
      WRITE (6,160) AOT                                                 CON37620
      WRITE (6,140) EXITFR                                              CON37630
C                                                                       CON37640
      DO 50 I=1,ISEC                                                    CON37650
      IF (WTST(I).LT.0.) AWTST(I)=-WTST(I)                              CON37660
      IF (WTST(I).LT.0.) WRITE (6,150) I,AWTST(I)                       CON37670
50    CONTINUE                                                          CON37680
C                                                                       CON37690
      WRITE (6,170) TNO,NOROWS                                          CON37700
      WRITE (6,180) ALST                                                CON37710
      WRITE (6,190) VOIDID,BNDIAM,BL                                    CON37720
      IF (MDIAM.EQ.1) WRITE (6,200) SIDO,ODOI                           CON37730
      IF (MDIAM.EQ.2) WRITE (6,210) SIDI,ODII                           CON37740
      IF (MDIAM.EQ.2) WRITE (6,220) SIDO,ODOI                           CON37750
      IF (MPITCH.EQ.1) WRITE (6,230) SDDO                               CON37760
      IF (MPITCH.EQ.2) WRITE (6,240) SDDI,SDDO                          CON37770
      IF (XW2.GT.0.) WRITE (6,250) XW2                                  CON37780
      IF (XW2.LE.0.) WRITE (6,260) XW1                                  CON37790
      WRITE (6,270) RSPI                                                CON37800
      WRITE (6,280) VOL1C,VOL2,BUNWT                                    CON37810
      WRITE (6,290) AVTBI                                               CON37820
      WRITE (6,300) AVTB2                                               CON37830
      WRITE (6,320) SMWB                                                CON37840
      WRITE (6,330) VWBAR                                               CON37850
      WRITE (6,310) HDLOSS,PHPCON                                       CON37860
      WRITE (6,340) DTCND2                                              CON37870
      WRITE (6,350) UPCOND                                              CON37880
      WRITE (6,360) PFILL                                               CON37890
      IF (PRCCLR.LE.0) WRITE (6,370) ARATIO                             CON37900
      IF (PRCCLR.GT.0.) CALL OUT2C                                      CON37910
      RETURN                                                            CON37920
```

```
60     FORMAT (1H1)                                                      CON37930
70     FORMAT (1H0,31X,29H********  CONDENSER  ********//)               CON37940
80     FORMAT (1H ,15X,16X,29H*****************************)             CON37950
90     FORMAT (1H ,15X,16X,1H*,27X,1H*)                                  CON37960
100    FORMAT (1H ,15X,16X,29H* CONDIP SUMMARY OF RESULTS *)             CON37970
110    FORMAT (1H ,10X,56HOVERALL U  LOG-MEAN  PRESSURE    TEMP.    STEAMCON37980
      1 VELOCITY/10X,54HB/HR-DEG.F  DELTA T    DROP      DROP          FCON37990
      2T/SEC/11X,57H-SQ.FT.     DEG.F      PSIA      DEG.F     INLET    OUCON38000
      3TLET//12X,F6.2,3X,F6.2,4X,1PE9.2,1P3E10.2/)                       CON38010
120    FORMAT (1H ,10X,12HHEAT DUTY    ,8X,1PE11.4,2X,6HBTU/HR)          CON38020
130    FORMAT (1H ,10X,18HSTEAM TO BUNDLE   ,2X,1PE11.4,2X,5HLB/HR)      CON38030
140    FORMAT (1H ,10X,13HEXIT FRAC.   ,7X,1PE11.4/)                     CON38040
150    FORMAT (1H ,10X,10HIN SECTOR ,I2,2X,29HTHE FRACTION OF DRY ROWS WACON38050
      1S ,F10.6)                                                         CON38060
160    FORMAT (1H ,10X,19HHT. TRF. SURF. AREA,1X,1PE11.4,2X,6HSQ.FT.)    CON38070
170    FORMAT (1H ,1X/10X,16HBUNDLE DIAMETER,,3X,2HFT,7X,F10.2,1X,6HTUBESCON38080
      1,,I5,1X,4HROWS)                                                   CON38090
180    FORMAT (1H ,12X,6HINSIDE,4X,7HOUTSIDE,13X,10HTUBES ARE ,F5.2,1X,7HCON38100
      1FT LONG)                                                          CON38110
190    FORMAT (1H ,12X,F5.2,5X,F5.2,14X,11HBUNDLE IS  ,F5.2,1X,7HFT LONG/CON38120
      1)                                                                 CON38130
200    FORMAT (1H ,10X,20HTUBE INSIDE DIAMETER,15X,F7.4,2X,6HINCHES/1H ,1CON38140
      10X,21HTUBE OUTSIDE DIAMETER,14X,F7.4,2X,6HINCHES)                 CON38150
210    FORMAT (1H ,10X,30HTUBE INSIDE DIAMETER INNER ROW,6X,F7.4,2X,3HIN.CON38160
      1/11X,31HTUBE OUTSIDE DIAMETER INNER ROW,5X,F7.4,2X,3HIN./)        CON38170
220    FORMAT (1H ,11X,30HTUBE INSIDE DIAMETER OUTER ROW,5X,F7.4,2X,3HIN.CON38180
      1/11X,31HTUBE OUTSIDE DIAMETER OUTER ROW,5X,F7.4,2X,3HIN.)         CON38190
230    FORMAT (1H ,10X,26HTUBE CIRCUMFERENTIAL PITCH,9X,F7.4)            CON38200
240    FORMAT (1H ,10X,36HTUBE CIRCUMFERENTIAL PITCH INNER ROW,5X,F7.4/1HCON38210
      1 ,10X,36HTUBE CIRCUMFERENTIAL PITCH OUTER ROW,5X,F7.4)            CON38220
250    FORMAT (1H ,10X,14HTUBE THICKNESS,21X,F7.4,2X,6HINCHES)           CON38230
260    FORMAT (1H ,10X,34HRATIO OF TUBE THICKNESS TO TUBE OD,5X,F7.4)    CON38240
270    FORMAT (1H ,10X,28HSPACE BETWEEN COCENTRIC ROWS,7X,F7.4,2X,3HIN./)CON38250
280    FORMAT (1H ,10X,13HBUNDLE VOLUME,7X,1PE11.4,2X,6HCU.FT./11X,19HTUBCON38260
      1E SPACE VOLUME  ,1X,1PE11.4,2X,6HCU.FT./11X,19HTUBE METAL WEIGHT  CON38270
      2,1X,1PE11.4,2X,2HLB/)                                             CON38280
290    FORMAT (1H ,10X,19HINLET WATER TEMP   ,5X,F9.2,1X,5HDEG.F)        CON38290
300    FORMAT (1H ,10X,20HOUTLET WATER TEMP   ,4X,F9.2,1X,5HDEG.F)       CON38300
310    FORMAT (1H ,10X,9HHEAD LOSS,15X,F9.2,1X,6HFT H2O/11X,13HPUMPING POCON38310
      1WER,11X,F9.2,1X,2HHP/)                                            CON38320
320    FORMAT (1H ,10X,13HWATER FLOW   ,12X,1PE11.4,1X,5HLB/HR)          CON38330
330    FORMAT (1H ,10X,17HWATER VELOCITY   ,8X,1PE11.4,1X,6HFT/SEC/)     CON38340
340    FORMAT (1H ,10X,12HAVG. LMTD   ,14X,F7.2,1X,5HDEG.F)              CON38350
350    FORMAT (1H ,10X,9HAVG. U   ,17X,F7.2,1X,18HBTU/HR-SQ.FT-DEG.F/)   CON38360
360    FORMAT (1H ,10X,46HTHE PERCENTAGE OF THE OUTERMOST ROW FILLED IS ,CON38370
      1F7.4//)                                                           CON38380
370    FORMAT (1H ,10X,18HTHE AREA RATIO IS ,F7.4//)                     CON38390
       END                                                               CON38400
```

```
C
C***********************************************************************
C***********************************************************************
C
      SUBROUTINE OUT2C
C-------------------
C
C              SUMMARY OUTPUT FOR COOLER SECTION
C
C-------------------
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDC,VELBI,XW1,XW2,VOL1,VOL2,TNOTO
     2T,BNDRAC,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR
      COMMON /INPT/  BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,WNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /ORC1/  AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDUT(10
     10)
      COMMON /ORC2/  ANT(100,15),STB2ES,VW(100)
      COMMON /JUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNO,VEL2,FR
      COMMON /OUTC/  EXITFC,EXITFR,HTCLR,ACC,SUMQC,SDIC,SDOC,UBARW,SMWBC,
     1SMTBIC,SMTB2C,VWBAR,STSATX,PMIX1,AOT,BUNWT,AVTBI,AVTB2,WTSTC,VOLC
      COMMON /CONST/  AMWNC,CBI,CPB,PI,SG,IFIRST
      COMMON /SEC/ ALMTD(100,15),CUMDP(100,15),DELP(15),GFLW(100,15),HEF
     1F(100,15),PMIX(100,15),PSAT(100,15),QOA(100,15),RC(100,15),ROUT(10
     20,15),SHI(100,15),SHN(100,15),STSAT(100,15),UN(100,15),VEL(100,15)
     3,VNRE(100,15),VPSHH(100,15),WCND(100,15),WGAS(15),WP(15),WS(100,15
     4),WSP(15)
      COMMON /COOL1/  VELC(100),PMIXC(100),PSATC(100),STSATC(100),WSC(100
     1),ALMTDC(100),SHIC(100),SHNFC(100),RCC(100),ROUTC(100),VNREC(100),
     2VPSHHC(100),HEFFC(100),UNC(100),WCNDC(100),GFLOWC(100),QOAC(100),C
     3UMDPC(100)
      COMMON /COOL/  IVNOC,HNOC,TNOC,TSATEX,PMXEXT,AMLSEX,WSEXIT,VELEXT,T
     1B2C,ENHIC,ENHOC,ENHFC,SHWINC,WBC
      UBARWC=0.
      DO 10  J=1,IVNOC
      UBARWC=UBARWC+UNC(J)*HNOC
10    CONTINUE
      DELPVC=PMIXC(1)-PMIXC(IVNOC+1)
      UBARWC=UBARWC/TNOC
      AOTC=AOC*TNOC
      ADTCLR=SUMQC/(UBARWC*AOTC)
      TDROPC=STSATC(1)-STSATC(IVNOC+1)
      AVTBIC=SMTBIC/SMWBC
      AVTB2C=SMTB2C/SMWBC
      WDCLR=SDOC*SDDC*HNOC
      COOLWT=TUBESW*TNOC*ALST*PI*(SDOC*SDOC-SDIC*SDIC)/4.
```

```
      VWBARC=VWBAR
C
      IF (AVTBIC.EQ.AVTB2C) DTCOL2=0.
      IF (AVTBIC.EQ.AVTB2C) UPCOOL=UBARWC
      IF (AVTBIC.EQ.AVTB2C) GO TO 20
C
      DTCOL2=(AVTB2C-AVTBIC)/ALOG((STSATX-AVTBIC)/(STSATX-AVTB2C))
      UPCOOL=SUMQC/(AOTC*DTCOL2)
C
C     VALUES FOR OVERALL OUTPUT
C
20    AOTOT=AOTC+AOT
      TNOT=TNC+TNOC
      UBAROA=(UBARW*TNO+UBARWC*TNOC)/TNOT
      ADTOA=(SUMQC+SUMQ)/(UBAROA*AOTOT)
      DELPOA=PMIX1-PMIXC(IVNOC+1)
      TDRPOA=STSAT1-STSATC(IVNOC+1)
      SUMQOA=SUMQC+SUMQ
      WTOA=BUNWT+COOLWT
      AVTBIA=(AVTBI*TNO+AVTBIC*TNOC)/TNOT
      AVTB2A=(AVTB2*TNO+AVTB2C*TNOC)/TNOT
      SMWBOA=SMWBC+SMWB
      HDLSOA=DELWP*144./62.366
      DTOA=(AVTB2A-AVTBIA)/ALOG((STSAT1-AVTBIA)/(STSAT1-AVTB2A))
      UPOA=SUMQOA/(AOTOT*DTOA)
      WRITE (6,40)
      WRITE (6,50) UBARWC,ADTCLR,DELPVC,TDROPC,VELC(1),VELC(IVNOC)
      WRITE (6,60) SUMQC
      WRITE (6,70) WSC(1)
      WRITE (6,110) AOTC
C
      IF (EXITFC.LT.-900.) WRITE (6,100)
      IF (EXITFC.LT.-900.) GO TO 30
C
      WRITE (6,80) EXITFC
C
30    IF (WTSTC.LT.0) WRITE (6,90) WTSTC
C
      WRITE (6,120) TNOC,IVNOC
      WRITE (6,130)
      WRITE (6,140) HTCLR,WDCLR
      WRITE (6,150) VCLC,COOLWT
      WRITE (6,160) AVTBIC
      WRITE (6,170) AVTB2C
      WRITE (6,180) SMWBC
      WRITE (6,190) VWBARC
      WRITE (6,210) DTCOL2
      WRITE (6,220) UPCOOL
```

```
      WRITE (6,230) SDDC                                                CON39370
      WRITE (6,250)                                                     CON39380
      WRITE (6,260) UBAROA,ADTOA,DELPOA,TDRPOA                          CON39390
      WRITE (6,60) SUMQOA                                               CON39400
      WRITE (6,110) AOTOT                                               CON39410
      WRITE (6,80) EXITOA                                               CON39420
      WRITE (6,270) TNOT                                                CON39430
      WRITE (6,280) VOL1,WTOA                                           CON39440
      WRITE (6,160) AVTB1A                                              CON39450
      WRITE (6,170) AVTB2A                                              CON39460
      WRITE (6,180) SMWBOA                                              CON39470
      WRITE (6,200) HDLSOA,PHP                                          CON39480
      WRITE (6,210) DTOA                                                CON39490
      WRITE (6,220) UPOA                                                CON39500
      IF (PRCCLR.GT.0) WRITE (6,240) ARATIO                             CON39510
      RETURN                                                            CON39520
40    FORMAT (1H0,31X,29H********   COOLER    ********//)               CON39530
50    FORMAT (1H ,10X,56HOVERALL U  LOG-MEAN  PRESSURE    TEMP.    STEAMCON39540
     1 VELOCITY/10X,54HB/HR-DEG.F  DELTA T    DROP       DROP          FCON39550
     2T/SEC/11X,57H-SQ.FT.    DEG.F      PSIA      DEG.F     INLET    OUCON39560
     3TLET//12X,F6.2,3X,F6.2,4X,1PE9.2,1P3E10.2/)                      CON39570
60    FORMAT (1H ,10X,12HHEAT DUTY   ,8X,1PE11.4,2X,6HBTU/HR)           CON39580
70    FORMAT (1H ,10X,18HSTEAM TO COOLER   ,2X,1PE11.4,2X,5HLB/HR)      CON39590
80    FORMAT (1H ,10X,13HEXIT FRAC.   ,7X,1PE11.4/)                     CON39600
90    FORMAT (1H ,10X,39HIN COOLER, THE FRACTION OF DRY ROWS WAS,F10.6/)CON39610
100   FORMAT (1H ,10X,46HTHE COOLER IS ENTIRELY DRY AND THUS NO EXIT FR/CON39620
     1)                                                                 CON39630
110   FORMAT (1H ,10X,19HHT. TRF. SURF. AREA,1X,1PE11.4,2X,6HSQ.FT.)    CON39640
120   FORMAT (1H ,14X,11HDIMENSIONS,,7X,2HFT,7X,F10.2,1X,6HTUBES,,I5,1X,CON39650
     14HROWS)                                                           CON39660
130   FORMAT (1H ,12X,6HHEIGHT,5X,5HWIDTH)                              CON39670
140   FORMAT (1H ,12X,F5.2,5X,F5.2/)                                    CON39680
150   FORMAT (1H ,10X,19HTOTAL COOLER VOLUME,1X,1PE11.4,2X,6HCU.FT./11X,CON39690
     115HCOOLER WEIGHT  ,1X,1PE11.4,2X,2HLB/)                           CON39700
160   FORMAT (1H ,10X,19HINLET WATER TEMP   ,5X,F9.2,1X,5HDEG.F)        CON39710
170   FORMAT (1H ,10X,20HOUTLET WATER TEMP   ,4X,F9.2,1X,5HDEG.F)       CON39720
180   FORMAT (1H ,10X,13HWATER FLOW   ,9X,1PE11.4,1X,5HLB/HR/)          CON39730
190   FORMAT (1H ,10X,17HWATER VELOCITY   ,5X,1PE11.4,1X,6HFT/SEC/)     CON39740
200   FORMAT (1H ,10X,9HHEAD LOSS,15X,F9.2,1X,6HFT H2O/11X,13HPUMPING POCON39750
     1WER,11X,F9.2,1X,2HHP/)                                            CON39760
210   FORMAT (1H ,10X,12HAVG. LMTD   ,14X,F7.2,1X,5HDEG.F)              CON39770
220   FORMAT (1H ,10X,9HAVG. U   ,17X,F7.2,1X,18HBTU/HR-SQ.FT-DEG.F)    CON39780
230   FORMAT (1H ,10X,35HTHE COOLER CIRCUMFERENTIAL PITCH IS,2X,F6.2//) CON39790
240   FORMAT (1H ,10X,26HTHE OVERALL AREA RATIO IS ,F7.4//)             CON39800
250   FORMAT (1H0,31X,29H********   OVERALL    ********//)              CON39810
260   FORMAT (1H ,10X,38HOVERALL U  LOG-MEAN  PRESSURE    TEMP./10X,36HBCON39820
     1/HR-DEG.F  DELTA T    DROP      DROP/11X,37H-SQ.FT.    DEG.F     PCON39830
     2SIA      DEG.F//12X,F6.2,3X,F6.2,4X,1PE9.2,1PE10.2/)              CON39840
```

```
270   FORMAT (1H /12X,F10.2,2X,5HTUBES/)
280   FORMAT (1H ,10X,12HTOTAL VOLUME,7X,1PE11.4,2X,6HCU.FT./11X,14HTOTA
     1L WEIGHT  ,5X,1PE11.4,2X,2HLB/)
      END
C
C*****************************************************************
C*****************************************************************
C
      SUBROUTINE OUT3
C--------------
C OUT3 1
C                    DETAILED OUTPUT
C ZZZ OUT3-1
C--------------
      COMMON /GLOBCM/ ALST,DELWP,DELWPC,EXITOA,GFLOW,SIDI,SIDO,PHP,RSPA,
     1RADINS,REWI,REWO,SDDI,SDDO,SLDI,SLDO,VELBI,XW1,XW2,VOL1,VOL2,TNOTO
     2T,BNDRAC,ARATIO,ODII,ODOI,SDDC,VLCMAX,PRCCLR
      COMMON /INPT/ BC(6),BE(6),ENH(6),ENI(6),ENO(6),FOUL,PHI,SKW,STBI,S
     1TSAT1,TUBESW,WNCIR,IOPT,PTST(15),SECWID,IBAF,ISEC,JBAF(16),JGAS,MD
     2IAM,MPITCH,NEE,NEI,NETE(6),NETI(6),NRNE(6),NRNI(6),NOROWS,WTST(15)
     3,WSI,PFILL
      COMMON /ORC1/ AOTFLW(100),SID(100),TBNPR(100),AOLFLW(100),CMDOT(10
     10)
      COMMON /ORC2/ ANT(100,15),STB2ES,VW(100)
      COMMON /OUT/ DELOD,SMTBI,SMTB2,SMWB,SUMQ,TNO,VEL2,FR
      COMMON /OUTC/ EXITFC,EXITFR,HTCLR,ACC,SUMQC,SDIC,SDOC,UBARW,SMWBC,
     1SMTBIC,SMTB2C,VWBAR,STSATX,PMIX1,AOT,BUNWT,AVTBI,AVTB2,WTSTC,VOLC
      COMMON /CONST/ AMWNC,CBI,CPB,PI,SG,IFIRST
      COMMON /COOL1/ VELC(100),PMIXC(100),PSATC(100),STSATC(100),WSC(100
     1),ALMTDC(100),SHIC(100),SHNFC(100),RCC(100),ROUTC(100),VNREC(100),
     2VPSHHC(100),HEFFC(100),UNC(100),WCNDC(100),GFLOWC(100),QOAC(100),C
     3UMDPC(100)
      COMMON /COOL/ IVNOC,HNOC,TNOC,TSATEX,PMXEXT,AMLSEX,WSEXIT,VELEXT,T
     1B2C,ENHIC,ENHOC,ENHFC,SHWINC,WBC
      COMMON /SEC/ ALMTD(100,15),CUMDP(100,15),DELP(15),GFLW(100,15),HEF
     1F(100,15),PMIX(100,15),PSAT(100,15),QOA(100,15),RC(100,15),ROUT(10
     20,15),SHI(100,15),SHN(100,15),STSAT(100,15),UN(100,15),VEL(100,15)
     3,VNRE(100,15),VPSHH(100,15),WCND(100,15),WGAS(15),WP(15),WS(100,15
     4),WSP(15)
      ROWS=FLOAT(NOROWS)
      IF (MDIAM.EQ.1) ODII=ODOI
      IF (MPITCH.EQ.1) SDDI=SDDO
      DELSDD=(SDDO-SDDI)/(ROWS-1.)
      WRITE (6,140)
      DO 10 I=1,NOROWS
      A=CMDOT(I)*3600.
      OD=ODOI-DELOD*FLOAT(I-1)
      SDD=SDDO-DELSDD*FLOAT(I-1)
```

```
10     WRITE (6,160) I,VW(I),A,OD,SDD,TBNPR(I)                              CON40330
       DO 30 J=1,ISEC                                                       CON40340
       WRITE (6,120) J                                                      CON40350
       DO 20 I=1,NCROWS                                                     CON40360
       OD=ODOI-DELOD*FLOAT(I-1)                                             CON40370
       AO=3.14159*OD*ALST                                                   CON40380
       Q=UN(I,J)*ALMTD(I,J)*AO                                              CON40390
20     WRITE (6,160) I,Q,CUMDP(I,J),ALMTD(I,J),RC(I,J),VNRE(I,J)            CON40400
30     CONTINUE                                                             CON40410
       DO 50 J=1,ISEC                                                       CON40420
       WRITE (6,130) J                                                      CON40430
       DO 40 I=1,NOROWS                                                     CON40440
40     WRITE (6,160) I,UN(I,J),ROUT(I,J),HEFF(I,J),SHN(I,J),SHI(I,J)        CON40450
50     CONTINUE                                                             CON40460
       DO 70 J=1,ISEC                                                       CON40470
       WRITE (6,150) J                                                      CON40480
       DO 60 I=1,NOROWS                                                     CON40490
60     WRITE (6,160) I,WS(I,J),WCND(I,J),STSAT(I,J),PSAT(I,J),PMIX(I,J)     CON40500
70     CONTINUE                                                             CON40510
       IF (PRCCLR.LE.0.) GO TO 110                                          CON40520
       WRITE (6,170)                                                        CON40530
       DO 80 I=1,IVNOC                                                      CON40540
       QC=UNC(I)*ALMTDC(I)*AOC                                              CON40550
       WRITE (6,200) I,QC,CUMDPC(I),ALMTDC(I),RCC(I),VNREC(I)               CON40560
80     CONTINUE                                                             CON40570
       WRITE (6,180)                                                        CON40580
       DO 90 I=1,IVNOC                                                      CON40590
       WRITE (6,200) I,UNC(I),ROUTC(I),HEFFC(I),SHNFC(I),SHIC(I)            CON40600
90     CONTINUE                                                             CON40610
       WRITE (6,190)                                                        CON40620
       DO 100 I=1,IVNOC                                                     CON40630
       WRITE (6,200) I,WSC(I),WCNDC(I),STSATC(I),PSATC(I),PMIXC(I)          CON40640
100    CONTINUE                                                             CON40650
110    RETURN                                                               CON40660
120    FORMAT (1H1////15X,15X,28HROW BY ROW OUTPUT FOR SECTOR,I3//15X,3HCON40670
      1ROW,4X,4HHEAT,5X,8HPRESSURE,3X,8HLOG MEAN,3X,8HRATIO OF,2X,10HSTEACON40680
      2M SIDE/15X,3HNO.,2X,8HTRANSFER,5X,4HDROP,5X,9HTEMP DIFF,2X,7HN/C GCON40690
      3AS,3X,12HREYNOLDS NO./15X,5X,8HPER TUBE,3X,7HPER ROW,14X,10HTO STMCON40700
      4 FLW/15X,6X,6HBTU/HR,6X,3HPSI,8X,5HDEG F/)                           CON40710
130    FORMAT (1H1////15X,5X,47HROW BY ROW HEAT TRANSFER COEFICIENTS FORCON40720
      1 SECTOR,I3/15X,23X,14HBTU/HR-FT**2-F//15X,3HROW,2X,8HOVER-ALL,3X,8CON40730
      2HEXTERNAL,3X,8HEXTERNAL,3X,8HEXTERNAL,3X,8HINTERNAL/15X,3HNO.,4X,4CON40740
      3HTUBE,7X,4HTUBE,5X,8HGAS FILM,3X,8HLIQ FILM,3X,8HLIQ FILM/)          CON40750
140    FORMAT (1H1////15X,9X,41HROW BY RCW VALUES OF CONDENSER PARAMETERCON40760
      1S//15X,3HROW,3X,7HCOOLANT,2X,12HCOOLANT MASS,3X,4HTUBE,7X,4HTUBE,4CON40770
      2X,9HNUMBER OF/15X,3HNO.,3X,21HVELOCITY  FLOW/TUBE  ,4X,2HOD,8X,5HPCON40780
      3ITCH,3X,9HTUBES PER/15X,17X,6HLBM/HR,6X,6HINCHES,14X,11HROW PER SECON40790
      4C/)                                                                  CON40800
```

```
150   FORMAT (1H1/////15X,4X,49HROW BY RCW STEAM AND CONDENSATE VALUES FCON40810
     1OR SECTOR,I3//15X,3HROW,3X,8HSTM FLOW,2X,10HCONDENSATE,1X,10HSATURCON40820
     2ATION,1X,10HSATURATION,2X,7HMIXTURE/15X,3HNO.,3X,8HENTERING,2X,9HGCON40830
     3ENERATED,4X,4HTEMP,5X,8HPRESSURE,3X,8HPRESSURE/15X,7X,6HLBM/HR,5X,CON40840
     46HLBM/HR,6X,5HDEG F,7X,3HPSI,8X,3HPSI/)                           CON40850
160   FORMAT (1H ,14X,I3,5(2X,1PE9.3))                                  CON40860
170   FORMAT (1H1/////15X,15X,28HROW BY ROW OUTPUT FOR COOLER//15X,3HROWCON40870
     1,4X,4HHEAT,5X,8HPRESSURE,3X,8HLOG MEAN,3X,8HRATIO OF,2X,10HSTEAM SCON40880
     2IDE/15X,3HNO.,2X,8HTRANSFER,5X,4HDROP,5X,9HTEMP DIFF,2X,7HN/C GAS,CON40890
     33X,12HREYNOLDS NO./15X,5X,8HPER TUBE,3X,7HPER ROW,14X,10HTO STM FLCON40900
     4W/15X,6X,6HBTU/HR,6X,3HPSI,8X,5HDEG F/)                           CON40910
180   FORMAT (1H1/////15X,5X,47HROW BY ROW HEAT TRANSFER COEFICIENTS FORCON40920
     1 COOLER/15X,23X,14HBTU/HR-FT**2-F//15X,3HROW,2X,8HOVER-ALL,3X,8HEXCON40930
     2TERNAL,3X,8HEXTERNAL,3X,8HEXTERNAL,3X,8HINTERNAL/15X,3HNO.,4X,4HTUCON40940
     3BE,7X,4HTUBE,5X,8HGAS FILM,3X,8HLIQ FILM,3X,8HLIQ FILM/)           CON40950
190   FORMAT (1H1/////15X,4X,49HROW BY RCW STEAM AND CONDENSATE VALUES FCON40960
     1OR COOLER//15X,3HROW,3X,8HSTM FLOW,2X,10HCONDENSATE,1X,10HSATURATICON40970
     2ON,1X,10HSATURATION,2X,7HMIXTURE/15X,3HNO.,3X,8HENTERING,2X,9HGENECON40980
     3RATED,4X,4HTEMP,5X,8HPRESSURE,3X,8HPRESSURE/15X,7X,6HLBM/HR,5X,6HLCON40990
     4BM/HR,6X,5HDEG F,7X,3HPSI,8X,3HPSI/)                              CON41000
200   FORMAT (1H ,14X,I3,5(2X,1PE9.3))                                  CON41010
      END                                                               CON41020
C                                                                       CON41030
C*********************************************************************  CON41040
C                          SUPPORTING SUBROUTINES                       CON41050
C*********************************************************************  CON41060
C                                                                       CON41070
      SUBROUTINE DFSVTY (TK,PATM,AM1,AM2,R1,R2,E10K,E20K,V1,V2,DG,DGG)  CON41080
C  DG = GAS DIFFUSIVITY, SQ.CM./SEC - CHEM.ENGRS.HANDBOOK, 4TH ED.,     CON41090
C  R.H.PERRY, 14-20, 1963, MC GRAW-HILL  (WILKE-LEE MOD.)               CON41100
C  CID = VALUES OF COLLISION INTEGRAL. FUNCTION FITTED BY R.W.BROWELL   CON41110
C        FROM TABLE 14-45 OF ABOVE REF. 4/11/68                         CON41120
C  TK   = ABS. TEMP., DEG. KELVIN                                       CON41130
C  PATM= ABS. PRESS., ATMOSPHERES                                       CON41140
C  R12 = COLLISION DIA., ANGSTROMS                                      CON41150
C  R1  = COMPONENT 1 COLLISION DIA., ANGSTROMS                          CON41160
C  R2  = COMPONENT 2 COLLISION DIA., ANGSTROMS                          CON41170
C E120K= COMBINED FORCE CONSTANT FOR COMPONENTS 1 & 2., DEG. KELVIN     CON41180
C  DGG = GAS DIFFUSIVITY - GILLILAND'S CRUDE APPROX. METHOD,PP.14-21 REFCON41190
      IF (E10K.EQ.0.0.AND.V1.EQ.0.0) GO TO 30                           CON41200
      IF (E10K.EQ.0.0) GO TO 20                                         CON41210
      IF (R1.GT.0.0) GO TO 10                                           CON41220
      R1=1.18*V1**.333                                                  CON41230
      R2=1.18*V2**.333                                                  CON41240
10    CONTINUE                                                          CON41250
      R12=(R1+R2)/2.0                                                   CON41260
      E120K=(E10K*E20K)**0.5                                            CON41270
      TKOE=TK/E120K                                                     CON41280
```

```
      CID=0.1685/TKOE**1.2063+0.5328/TKOE**0.1579
      RTSM=(1./AM1+1./AM2)**0.5
      B=(10.7-2.46*RTSM)*1.E-4
      DG=B*TK**1.5*RTSM/PATM/R12**2/CID
      IF (V1.EQ.0.0) GO TO 40
20    CONTINUE
      DGG=0.0043*TK**1.5*RTSM/PATM/(V1**.333+V2**.333)**2
      GO TO 40
30    CONTINUE
      WRITE (6,50)
40    CONTINUE
      RETURN
50    FORMAT (103H0  ** NEITHER FORCE CONSTANTS NCR MOLECULAR VOLUMES EX
     1IST AS ARGUMENTS. CALCULATIONS CANNOT PROCEED. **)
      END
C
C
      FUNCTICN XTR (X,Y,N,T,SW,X1N,YN,M)
C******
C         FUNCTION XTR ASSUMES A LOG-LOG MODEL, IE  LOG Y = A + B1 * LOG
C         AND PERFORMS A LINEAR LEAST SQUARE REGRESSION ON THE
C         TRANSFORMED POINTS, ALL POINTS ARE USED FOR EACH REGRESSION
C         EXCEPT FOR THE FIRST , WHICH IS NOT USED AT ALL
C********
      DIMENSICN X1N(11),YN(11)
      IF (Y.LE.0.) Y=.00001
      IF (N.EQ.0) GO TO 30
      X1N(N)=ALOG(X)
      YN(N)=ALOG(Y)
      IF (N.LT.2) GO TO 40
      SX1N=0.0
      SYN=0.0
      IF (M.LT.N-3) M=N-3
      DO 10 I=M,N
      SX1N=SX1N+X1N(I)
      SYN=SYN+YN(I)
10    CONTINUE
      X1BAR=SX1N/(N-M+1)
      YBAR=SYN/(N-M+1)
      SX12=0.0
      SX1Y=0.0
      DO 20 I=M,N
      SX12=SX12+(X1N(I)-X1BAR)**2
      SX1Y=SX1Y+(X1N(I)-X1BAR)*(YN(I)-YBAR)
20    CONTINUE
      B1=SX1Y/SX12
      A=YBAR-B1*X1BAR
      TN=ALOG(T)
```

```
      XTR=EXP((TN-A)/B1)                                                CON41770
      GO TO 50                                                          CON41780
30    M=1                                                               CON41790
40    XTR=X*(1.+(Y-T)*5.*SW)                                            CON41800
C******                                                                 CON41810
C          CHANGE OF VARIABLE IS LIMITED TO 10 PERCENT                  CON41820
C******                                                                 CON41830
50    IF (ABS(XTR-X)/X.LT.0.1) GO TO 60                                 CON41840
      XTR=X*(1.+0.1*ABS(Y-T)/(Y-T)*SW)                                  CON41850
60    RETURN                                                            CON41860
      END                                                               CON41870
C                                                                       CON41880
C                                                                       CON41890
      FUNCTICN AMUFN (T,JGAS)                                           CON41900
      IMIX=0                                                            CON41910
      GO TO (30,20,10), JGAS                                            CON41920
10    IMIX=1                                                            CON41930
20    CONTINUE                                                          CON41940
C  INERT GAS IS CO2                                                     CON41950
C**   VISCOSITY FOR CARBON DIOXIDE  LB/FT-SEC                           CON41960
      GAS2=1.0E-5*(-0.046+(2.282E-3+(-6.131E-7+9.699E-11*T)*T)*T)       CON41970
      AMUFN=GAS2                                                        CON41980
      IF (IMIX.EQ.0) RETURN                                             CON41990
30    CONTINUE                                                          CON42000
C  INERT GAS IS AIR                                                     CON42010
      GAS1=1.0E-5*(0.1490+(0.238E-2)*T-(.7209E-6)*T**2+(.1184E-9)*T**3) CON42020
      AMUFN=GAS1                                                        CON42030
      IF (IMIX.EQ.0) RETURN                                             CON42040
C  INERT GAS IS MIXTURE OF AIR & CO2                                    CON42050
      GAS3=(GAS1+GAS2)/2.0                                              CON42060
      AMUFN=GAS3                                                        CON42070
      RETURN                                                            CON42080
      END                                                               CON42090
C                                                                       CON42100
C                                                                       CON42110
      FUNCTICN BMUFN (C,T)                                              CON42120
C     VISCOSITY OF SALINE SOLUTION.  RANGE OF DATA WAS 0 - 24 PERCENT   CON42130
C     CONCENTRATION AND 40 - 210 DEGREES FARENHEIT.                     CON42140
      R=T+459.69                                                        CON42150
      BMUFN=DEXP(-0.11591155D+2*C+0.12602329D-1*C*R+0.38637378D+4*C/R+0.CON42160
     1460653 2D-2*R+0.47595941D+4/R-0.1059252566D+2)                     CON42170
      RETURN                                                            CON42180
      END                                                               CON42190
C                                                                       CON42200
C                                                                       CON42210
      FUNCTICN CPAFN (T,JGAS)                                           CON42220
      IMIX=0                                                            CON42230
      GO TO (30,20,10), JGAS                                            CON42240
```

```
10     IMIX=1
20     CONTINUE
C  INERT GAS IS CO2
C** HEAT CAPACITY FOR MW = 40.1   BTU/LB-MOL - DEG R
       GAS2=0.209*40.1
       CPAFN=GAS2
       IF (IMIX.EQ.0) RETURN
30     CONTINUE
C  INERT GAS IS AIR
C AIR CP, BTU / LB MOLE-DEGREE RANKIN
       GAS1=7.139-0.9884E-3*T+0.1393E-5*T**2-0.3367E-9*T**3
       CPAFN=GAS1
       IF (IMIX.EQ.0) RETURN
C  INERT GAS IS MIXTURE OF AIR & CO2
       GAS3=(GAS1+GAS2)/2.0
       CPAFN=GAS3
       RETURN
       END
C
C
       FUNCTION CPFN (C,T)
10     IF (C-0.005) 20,20,30
C*****EQUATION SPECIFIC HEAT FOR PURE WATER
20     CP=1.0121559+(-0.24618473E-3+0.10282155E-5*T)*T
       GO TO 40
C*****EQUATION SPECIFIC HEAT FOR BRINE
30     CP=.96946859+(2.*(0.00010404965)*T)-(.91199294*C)+(2.*(-.000648296
      159)*C*T)+(-1.555779*(C**2))+(2.*(.0076721469)*(C**2)*T)+(6.7981008
      2*(C**3))+(2.*(-.012610354)*(C**3)*T)
40     CPFN=CP
       RETURN
       END
C
C
       FUNCTION CPSFN (T)
C***
C      CPSFN CALCULATES THE HEAT CAP OF STEAM IN BTU/LB-MOL-R GIVEN T IN
C     DEG R. EQUATIONS FROM H. NORITAKE, BASED ON TABULATED VALUES IN
C     NASA TR-R-132
C***
       CPSFN=(7.838-(.2531E-3-(.2892E-6-.7693E-10*T)*T)*T)
       RETURN
       END
C
C
       FUNCTION HFGFN (T)
       HFGFN=1093.88-0.5703*T+.00012819*T**2-.0000008824*T**3
       RETURN
```

```
      END
C
C
      SUBROUTINE PRSDRP (TSAT,VMIX,WS,WNC,AXC,SDO,SF,DELPTP,ENHF)
CCC
C    PRSDRP  REBUILT ON 9-16-69 TO USE EQUATICN FCR SF
CCC
      SG=32.174
      GSTAR=(WS+WNC)/(AXO*3600.)
      ANRE=(SCO*GSTAR)/SMUFN(TSAT)
      SF=(0.102+52.2/ANRE)*ENHF
      DELPTP=SF*GSTAR**2*VMIX/(72.0*SG)
      RETURN
      END
C
C
      FUNCTICN PSATFN (T)
      PSATFN=2.718**(14.150119-(6452.5621/T)-(837533.21/T**2))
      RETURN
      END
C
C
      FUNCTICN ROEFN (C,T)
C     DENSITY OF SALINE SOLUTION.  RANGE OF DATA WAS 0 - 26 PERCENT
C     CONCENTRATION AND 40 - 300 DEGREES FARENHEIT.
      ROEFN=0.62707172E2+0.49364088E2*C-(0.43955304E-2+0.32554667E-1*C+(
     10.46076921E-4-0.63240299E-4*C)*T)*T
      RETURN
      END
C
C
      FUNCTICN SKBFN (C,T)
C             THERMAL CONDUCTIVITY OF SALINE SCLUTION.  RANGE OF DATA
C             0 - 24 PERCENT CONCENTRATION AND 40 - 300 DEGREES FARENHEIT
      SKBFN=(.30157913+.697989E-3*T-.12506E-5*T**2-.2072E-10*T**3)*(-.16
     187109*C+1.)
      RETURN
      END
C
C
      FUNCTICN SMUFN (T)
      SMUFN=1.0E-5*(0.122+(1.001E-3)*T+(2.892E-7)*T**2-(7.693E-11)*T**3)
      RETURN
      END
C
C
      FUNCTION TSATFN (P)
      AA=(ALOG(P)-14.150119)
```

```
      DET=6452.5621**2-(4.0*AA*837533.21)
      IF (DET) 10,20,20
10    WRITE (6,60)
      CALL EXIT
20    X=(-6452.5621+SQRT(DET))/(2.0*AA)
      Y=(-6452.5621-SQRT(DET))/(2.0*AA)
      IF (X-Y) 30,40,40
30    TSATFN=Y
      GO TO 50
40    TSATFN=X
50    CONTINUE
      RETURN
60    FORMAT (1H1,37HSUBROUTINE TSATFN FINDS COMPLEX ROOTS)
      END
C
C
      FUNCTION VGFN (T,P)
      X=ALOG(T/P)
      VGFN=EXP(((.103758E-2*X-.0177861)*X+1.10267)*X-.72240)
      RETURN
      END
C
C
      SUBROUTINE SWITCH (A,N)
      DIMENSION A(N)
      NN=N/2
      K=N+1
      DO 10 I=1,NN
      T=A(I)
      A(I)=A(K-I)
      A(K-I)=T
10    CONTINUE
      RETURN
      END
```

LIST OF REFERENCES

1. Heat Exchange Institute, Standards for Steam Surface Condensers, Sixth Edition, 1970.

2. Oak Ridge National Laboratory Report ORNL-TM-4248, ORCON1: A Fortran Code for the Calculation of a Steam Condenser of Circular Cross Section, by J.A. Hafford, July 1973.

3. Eissenberg, D.M., An Investigation of the Variables Affecting Steam Condensation on the Outside of a Horizontal Tube Bundle, Ph.D. Thesis, University of Tennessee, December 1972.

4. Fujii, T., Honda, H., and Oda, K., "Condensation of Steam on a Horizontal Tube - the Influence of Oncoming Velocity and Thermal Condition at the Tube Wall," Proc., 18th Nat'l. Heat Transfer Conference, San Diego, August 1979, ASME, pp. 35-43.

5. Johnson, C.M., Marine Steam Condenser Design Using Numerical Optimization, MSME Thesis, Naval Postgraduate School, December 1977.

6. Vanderplaats, G.N., Numerical Optimization Techniques for Engineering Design: With Applications, course notes for a graduate course of the same title presented at the Naval Postgraduate School, Monterey, California, September-December 1983.

7. NASA Ames Research Center Tehnical Memorandum NASA TM-X-62,282, CONMIN - A Fortran Program for Constrained Function Minimization User's Manual, by G.N. Vanderplaats, August 1973.

8. Naval Postgraduate School Report NPS 69-77-0004, Lastop - A Computer Program for Laser Turret Optimization, by G.N. Vanderplaats and A.E. Fuhs, 1977.

9. Fletcher, R. and Reeves, C.M., "Function Minimization by Conjugate Directions", British Computer Journal, v. 7, n. 2, p. 149-154, 1964.

10. Zoutendijk, G.G., Methods of Feasible Directions, Elsivier, Amsterdam, 1960.

11. Fox, R.L., Optimization Methods for Engineering Design, Addison-Wesley, 1971.

12. Vanderplaats, G.N. and Moses, F., "Structural

Optimization by Methods of Feasible Directions," *Journal of Computers and Structures*, v. 3, p. 739-755, 1973

13. Naval Postgraduate School Report NPS 69-81-003, *Copes - A Fortran Control Program for Engineering Synthesis*, by G.N. Vanderplaats and L.E. Madsen, March 1982.

14. Naval Postgraduate School Report NPS 69-82-005, *Performance Rating of Enhanced Marine Condensers*, by R.H. Nunn and P.J. Marto, August 1982.

15. Naval Ships Systems Command 346-0250, *Techanical Manual, Main Condenser and Air Ejector* (DLG-6).

16. Naval Boiler and Turbine Laboratory Test T-235, *DLG-6 Class Propulsion Machinery*, by J.J. Foti, 15 January 1960.

17. Department of the Navy Military Specifications, MIL-C-15430 J (Ships), *Condensers, Steam, Naval Shipboard*, 19 June 1975.

18. Lynch, V.J., *A Comparative Study of a Steam Surface Condenser Computer Model to Field Test Data*, MSME Thesis, Naval Postgraduate School, December 1979.

19. Heat Exchange Institute, *Standards for Steam Surface Condensers*, Sixth Edition, 1970.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Chairman, Code 69 Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93943	1
4.	Mr. Isidro A. Diaz-Tous EPRI P.O.Box 01412 Palo Alto, California 94303	1
5.	Mr. Ray Kornbau, Code 2721 David Taylor Naval Ship Research and Development Center Bethesda, Maryland 20084	1
6.	Mr. Dan A. Groghan NSEA 05R33 Department of the Navy Naval Sea Systems Command Washington, D.C. 20362	1
7.	Professor R.H. Nunn, Code 69Nn Naval Postgraduate School Monterey, California 93943	3
8.	Lt. Thomas Buckingham 15003 Kamputa Drive Centerville, Virginia 22020	3

205410

Thesis
B8316 Buckingham
c.1 Marine steam condenser design optimization.