

11010  
17053  
201P

NASA Contractor Report 194941



# Reference Manual for the Thermal Analyst's Help Desk Expert System

Rachel A. Ormsby  
*Lockheed Engineering and Sciences Company  
Langley Program Office, Hampton, Virginia*

(NASA-CR-194941) REFERENCE MANUAL  
FOR THE THERMAL ANALYST'S HELP DESK  
EXPERT SYSTEM (Lockheed  
Engineering and Sciences Corp.)  
201 p

N94-37504

Unclas

G3/18 0017053

Contract NAS1-19000

July 1994

National Aeronautics and  
Space Administration  
Langley Research Center  
Hampton, Virginia 23681-0001

**Reference Manual  
for the  
Thermal Analyst's Help Desk  
Expert System**

**Rachel A. Ormsby**

July 11, 1994

This reference manual contains both technical and programming guides for the expert system, Thermal Analyst's Help Desk. The Thermal Analyst's Help Desk provides the user with a capability for performing the preliminary analyses required to determine the thermal characteristics of a spacecraft. It gives the analyst a first look at the parameters required and provides an assessment of the feasibility of a preliminary thermal system design. Analyses can be performed on a single flat plate or a rectangular box enclosure. The enclosure analysis can be performed at a single orbit point or at a user defined number of orbit points. The orbit itself may be any elliptical conic section around any planet.

This reference manual is intended to give the knowledge engineer insight into the theory and development of the Thermal Analyst's Help Desk. It is not the purpose of this document to describe capabilities and functions that are contained in the EXSYS<sup>®</sup> manual. Therefore, there are locations in this document that are lacking in information. In these locations there are references to the EXSYS<sup>®</sup> manual and other readings about expert systems. It is not intended to replace those references in this manual. In certain instances, EXSYS<sup>®</sup> options are described more fully to give the knowledge engineer further insight into specific aspects of EXSYS<sup>®</sup> that are used in Help Desk.

This work was performed at the Langley Program Office of the Lockheed Engineering and Sciences Company for the Advanced Space Concepts Division at NASA Langley Research Center under contract number NAS1-19000.

## Acknowledgements

The author acknowledges Jeffery T. Farmer, Advanced Space Concepts Division, NASA Langley Research Center for his technical assistance and support during the creation the Thermal Analyst's Help Desk. Patrick A. Cosgrove and Philip Mutton, Langley Program Office, Lockheed Engineering and Sciences Company provided editorial support in the preparation of this document.

### Trademarks:

EXSYS<sup>®</sup> is a trademark of EXSYS Corporation, Albuquerque, NM

IDEAS<sub>TM</sub> is a trademark of Structural Dynamics Research Corporation, Milford, OH

TMG<sub>TM</sub> is a trademark of MAYA Heat Transfer Technologies Limited, Montreal, Quebec, Canada

## Table of Contents

<b>INTRODUCTION</b> .....	<b>1</b>
<b>PART I: TECHNICAL GUIDE</b>	
<b>EXPERT SYSTEMS</b> .....	<b>3</b>
<b>1.1 Chaining</b> .....	<b>6</b>
<b>2 MATHEMATICAL APPROACH</b> .....	<b>11</b>
<b>2.1 Environmental Heat Balance on a Surface:</b> .....	<b>11</b>
<b>2.2 Surface Heat Loads</b> .....	<b>11</b>
<b>2.3 Help Desk Surface Analyses</b> .....	<b>13</b>
<b>2.3.1 Surface Area</b> .....	<b>13</b>
<b>2.3.2 Surface Temperature</b> .....	<b>13</b>
<b>2.3.3 Environmental Temperature</b> .....	<b>14</b>
<b>2.4 Enclosure Heat Loads</b> .....	<b>14</b>
<b>2.4.1 Heat Balance for Each Surface of an Enclosure</b> .....	<b>14</b>
<b>2.4.2 Enclosure Heat Balance</b> .....	<b>15</b>
<b>3 SURFACE PROPERTIES</b> .....	<b>19</b>
<b>3.1 Updating Help Desk surface properties option</b> .....	<b>21</b>
<b>4 ORBITAL INFORMATION</b> .....	<b>23</b>
<b>5 VIEW FACTORS</b> .....	<b>27</b>
<b>5.1 Enclosure View Factors</b> .....	<b>27</b>
<b>5.2 Solar View Factors</b> .....	<b>28</b>
<b>5.3 Planet View Factors</b> .....	<b>30</b>
<b>6 SPACECRAFT ORIENTATIONS</b> .....	<b>33</b>
<b>6.1 Planet orientation</b> .....	<b>33</b>
<b>6.2 Inertial orientation</b> .....	<b>33</b>
<b>7 COORDINATE TRANSFORMATIONS</b> .....	<b>35</b>
<b>7.1 Global to Local:</b> .....	<b>35</b>
<b>7.2 Local to Global:</b> .....	<b>35</b>
<b>7.3 Coordinate Transformations Modifications</b> .....	<b>35</b>
<b>8 SHADOWING</b> .....	<b>39</b>
<b>8.1 Position Vector</b> .....	<b>39</b>
<b>8.2 Shadow Calculations</b> .....	<b>39</b>
<b>PART II: PROGRAMMER'S GUIDE</b>	
<b>1 RADIATE</b> .....	<b>41</b>

2	SCREENS .....	43
	2.1 Default Data Files .....	45
3	RUNNING EXTERNAL PROGRAMS .....	47
4	ADDING A NEW FEATURE .....	49
5	FUTURE DEVELOPMENT .....	55

## APPENDICES

	Rules .....	A1
	Qualifiers .....	B1
	Screens .....	C1
	Default Data Files .....	D1
	Report Generator Listing .....	E1
	Fortran Program List .....	F1
	Fortran Source Code .....	G1
	References .....	H1
	Bibliography .....	H2
	INDEX .....	H3

## List of Figures

Figure 1	Inference Engine Decision Diagram	6
Figure 2	Decision Tree	8
Figure 3	Heat Balance on a Surface	11
Figure 4	Enclosure Geometry and Surface Numbers	14
Figure 5	Surface Material and Application screens	21
Figure 6	Planet Choice screen	25
Figure 7	Two finite rectangles of same length, having one common edge, and at an angle of 90° to each other	27
Figure 8	Solar View Factor Geometry	28
Figure 9	Dot Product Geometry	28
Figure 10	Enclosure Unit Normals Geometry	29
Figure 11	Planet View Factor Geometry	30
Figure 12	Plane element $dA_1$ to sphere of radius $r$ ; normal to center of element passes through center of sphere	30
Figure 13	Plane element $dA_1$ to sphere of radius $r$ ; tangent to element passes through center of sphere	31
Figure 14	Orientation Geometry	33
Figure 15	Local Coordinate Geometry	36
Figure 16	Global Coordinate Geometry	36
Figure 17	Position Vector Geometry	39
Figure 18	Shadowing Geometry	40
Figure 19	Sample Data screen	43
Figure 20	Original Constants screen	49
Figure 21	Planet Selection Screen	51
Figure 22	New Analysis Constants screens	52
Figure 23	~Q1	C3
Figure 24	~Q2	C4
Figure 25	~Q3	C5
Figure 26	~Q4	C6
Figure 27	~Q5	C7
Figure 28	~Q9	C8
Figure 29	~Q10	C9
Figure 30	~Q12	C10
Figure 31	~Q14	C11
Figure 32	~Q16	C12
Figure 33	~Q17	C13
Figure 34	~Q18	C14
Figure 35	~Q21	C15
Figure 36	~[AINST]	C16
Figure 37	~[APPLICATION]	C17
Figure 38	~[AREA]	C18
Figure 39	~[CEPS]	C19
Figure 40	Spacecraft Position screen	C20
Figure 41	~[FE1U]	C21
Figure 42	~[GS1]	C22
Figure 43	~[INC]	C23
Figure 44	~[INS1]	C24
Figure 45	~[MATERIAL]	C25
Figure 46	~[NUMPTS]	C26

Figure 47	~[PLANET]	.....	C27
Figure 48	~[QPOWER]	.....	C28
Figure 49	Surface Temperature screen	.....	C29
Figure 50	~[X]	.....	C30
Figure 51	~[XS]	.....	C32
Figure 52	~[REPORT1]	.....	C33
Figure 53	~[REPORT2]	.....	C35
Figure 54	~[REPORT3]	.....	C37
Figure 55	~[REPORT4]	.....	C39
Figure 56	~[REPORT5]	.....	C42
Figure 57	~[REPORT6]	.....	C45

## List of Tables

TABLE 1: Surface Properties Frame .....	20
TABLE 2: Different Planet Frame .....	24



## Notation

$Q_i$	heat load through the surface
$E_{bi}$	blackbody radiation emitted from the surface = $\sigma_i T_i^4$
$A_i$	area of the surface
$\alpha_{Si}$	solar absorptivity
$\alpha_{IRi}$	infrared absorptivity = emissivity
$\epsilon_i$	emissivity of the surface
$E$	earth flux
$S$	solar flux
$\mu_i$	solar view factor for the surface
$F_{iE}$	earth view factor for the surface
ALB	albedo coefficient
$\mu_i$	$\cos\theta$ where $\theta$ is the angle between the sun vector and the plane normal

## Orbital Elements

$a$	semi-major axis
$e$	eccentricity
$i$	inclination
$\Omega$	longitude of ascending node
$\omega$	argument of periapsis
$v$	true anomaly at epoch

## **INTRODUCTION**

The following document describes the background and implementation of the theory used to create the Thermal Analyst's Help Desk.

The Thermal Analyst's Help Desk (also referred to as Help Desk) was created as a prototype to determine if expert systems can be used in the design and analysis of spacecraft. Its function is to provide first order approximations to thermal design parameters subjected to a radiation environment.

Its features include:

- sizing a radiator with a known surface temperature
- finding the temperature of a surface with a known area
- finding the sink temperature of a surface with only environmental effects
- determining the temperature and heat loads of an enclosure that is affected by environmental radiation
- determining the temperature and heat loads on an enclosure as it proceeds around an orbit
- assistance in determining the required emissivity and absorptivity for a surface.

The Thermal Analyst's Help Desk runs within an expert system shell called EXSYS® from EXSYS, Inc. in Albuquerque, New Mexico.

This document is divided into two parts:

- 1) Technical guide: describes the equations and theory that pertain to Help Desk
- 2) Programmer's guide: describes features about Help Desk that are not explained in the EXSYS® user's manual

# PART I: TECHNICAL GUIDE

## 1 EXPERT SYSTEMS

An expert system uses artificial intelligence (AI) techniques that allows the computer to perform as an expert in a specific and well defined field.

An Expert system has three basic parts,

- 1) a collection of knowledge on a given topic (knowledge base)
- 2) a set of rules pertaining to the knowledge (rule base)
- 3) a method to apply the rules (inference engine).

The knowledge and the rules can pertain to any topic that requires an expert to determine a result. The Thermal Analyst's Help Desk is an expert system that is used to help determine the thermal characteristics of a spacecraft.

In this text the *expert* is the source of the expertise which is included in the knowledge and rule bases. The *knowledge engineer* is the developer of the expert system, in this case the Thermal Analyst's Help Desk. The *user* refers to the individual running an analysis.

A **knowledge base** contains a series of symbolic parameters, called qualifiers, and possible valid responses, called values. The knowledge base also contains the definitions of variables which include variable name, type, numeric bounds and initialization values. The Help Desk knowledge base contains information about spacecraft and thermal analysis. Qualifiers and their values are created by the knowledge engineer using information known about the system. For example, a sample knowledge base is shown in Box 1. It shows the qualifiers and their values for the Surface Properties Analysis in Help Desk. Qualifier 1, "Value to calculate is" has six values listed. Qualifiers 9, 10 and 13 each have two values. These qualifiers do not assume an application or an order, they are simply a collection of knowledge.

/* Qualifier 1	/* Qualifier 10
Q> Value to calculate is	Q> the known surface information is
V> radiator area (given temperature)	V> surface use
V> surface temperature	V> surface material
V> enclosure temperatures and heat loads	
V> orbital average temperatures and heat loads	/* Qualifier 13
V> surface properties	Q> Surface properties are
V> none of the above	V> calculated
	V> unknown
/* Qualifier 9	
Q> the performance time is	
V> BOL (beginning of life)	
V> EOL (end of life)	

Box 1 Sample Knowledge Base

A **rule base** is a set of IF-THEN rules assigning values to qualifiers and numbers to variables. The IF-THEN rules in any expert system are comparable to IF-THEN statements in most higher

level computer languages. An example comparison is shown in Box 2 using the qualifiers and values from Box 1. A sample rule base, also from the Surface Properties analysis in Help Desk is shown in Box 3. The premise clauses (IF statements) for the four rules are similar. They represent all possible combinations of the two qualifiers. If one premise clause is false, the rule is marked False and is never executed. When all the premise clauses in a rule are true, the rule is marked True and is a candidate for execution by the expert system. Since any rule can be true at anytime, the rule base defines a state--it does not imply an order. The rules are ordered to increase efficiency, but not to define a sequence for evaluation.

EXPERT SYSTEM	FORTRAN
RULE NUMBER: 39 (FRAME EPS BOL)	
IF: Value to calculate is surface properties and the performance time is BOL (beginning of life) and the known surface information is surface use	IF ((Q1 .EQ. V1) .AND. (Q9 .EQ. V1) .AND. (Q10.EQ.V1))
THEN: [EPSILON] IS GIVEN THE VALUE... and [ALPHA] IS GIVEN THE VALUE .... and [MATERIAL] IS GIVEN THE VALUE ..... and Surface properties are calculated	THEN [EPSILON] = ... [ALPHA] = ... [MATERIAL]= ... Q13 = V1 ENDIF

**Box 2 FORTRAN Equivalent Statements**

The **inference engine** determines which rules are used, and the order in which they are applied. The inference engine defines the order by "firing" the rules. A rule is "fired" when the THEN clause is performed. The selection of applicable rules can be accomplished by multiple means. This expert system uses the method of chaining, which is explained in detail in the next section. In the example, this process reduced the examination of all possible rules to the examination of just four rules shown . When the inference engine examines a premise clause it asks the following:

**Is the value defined?**

- yes: Is the premise true or false**
- no: can the value be derived from an additional rule?**
  - yes: derive the value**
  - no: prompt the user for the answer.**

Figure 1 shows a flow diagram of the procedure described.

This method is applied to the example shown in Box 3. The inference engine would start with Rule 39 because, when there is no implied order in the chaining process, the inference engine evaluates the rules sequentially. For qualifier numbers and values, see Box 1.

RULE NUMBER: 39 (FRAME EPS BOL)

IF:  
Value to calculate is surface properties  
and the performance time is BOL (beginning of life)  
and the known surface information is surface use

THEN:  
[EPSILON] IS GIVEN THE VALUE....  
and [ALPHA] IS GIVEN THE VALUE ....  
and [MATERIAL] IS GIVEN THE VALUE .....  
and Surface properties are calculated

RULE NUMBER: 40 (FRAME EPS EOL)

IF:  
Value to calculate is surface properties  
and the performance time is EOL (end of life)  
and the known surface information is surface use

THEN:  
[EPSILON] IS GIVEN THE VALUE ....  
and [ALPHA] IS GIVEN THE VALUE ....  
and [MATERIAL] IS GIVEN THE VALUE ....  
and Surface properties are calculated

RULE NUMBER: 41

IF:  
Value to calculate is surface properties  
and the performance time is EOL (end of life)  
and the known surface information is surface material

THEN:  
[EPSILON] IS GIVEN THE VALUE ....  
and [ALPHA] IS GIVEN THE VALUE ....  
and [APPLICATION] IS GIVEN THE VALUE.....  
and Surface properties are calculated

RULE NUMBER: 42

IF:  
Value to calculate is surface properties  
and the performance time is BOL (beginning of life)  
and the known surface information is surface material

THEN:  
[EPSILON] IS GIVEN THE VALUE....  
and [ALPHA] IS GIVEN THE VALUE....  
and [APPLICATION] IS GIVEN THE VALUE....  
and Surface properties are calculated

### Box 3 Sample Rule Base

#### EXAMPLE:

Qualifier 1: "Value to calculate is"

Is the value defined?

- no: can its value be derived?
- no: prompt the user with a screen for the information.

When running Help Desk, the user is prompted at this point with the Main Menu, the menu with the analysis choices. If the user selects "surface properties", the inference engine assigns "surface properties" to qualifier 1 (or V5 to Q1) and proceeds to the next premise clause.

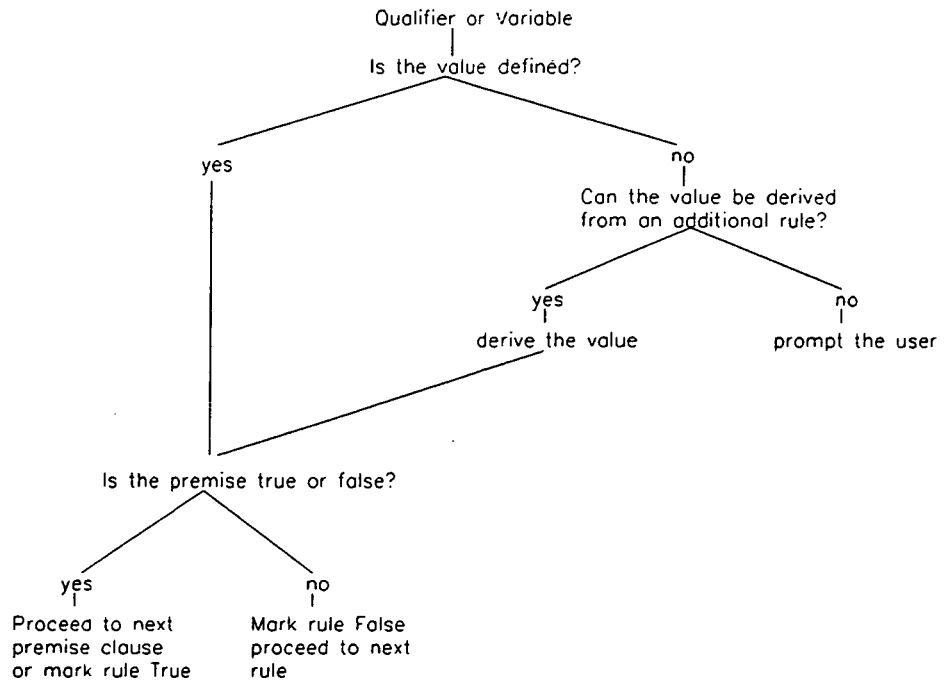
NOTE: In Help Desk, once a qualifier has been assigned a value, it cannot be changed.

Qualifier 9: "the performance time is"

Is the value defined?

- no: can its value be derived?
- no: prompt the user with a screen for the information.

The user is prompted for a choice of performance times: Beginning of Life or End of Life. If the user selects Beginning of Life, the inference engine assigns V1 to Q9 and proceeds to the next premise clause. (If the user selects End of Life, the inference engine assigns V2 to Q9, marks the rule as False and proceeds to the next rule.)



**Figure 1 Inference Engine Decision Diagram**

Qualifier 10: "the known surface information is"

Is the value defined?

no: can its value be derived?

no: prompt the user with a screen for the information.

At this point, the user is prompted for the known surface information. If the user selects surface use, the inference engine assigns V1 to Q10 and marks the rule as True. (If the user selects surface material, the inference engine assigns V2 to Q10, marks the rule as false and proceeds to the next rule.)

When a premise is completely satisfied the inference engine "fires" the conclusion. In this case it defines the value for EPSILON (emissivity), ALPHA (absorptivity), determines the unknown surface information, and defines Q13 to have the value "calculated".

**This sample analysis can be seen by running the "surface properties" option from the main menu of Help Desk.**

### 1.1 Chaining

The Help Desk inference engine uses a deductive procedure called chaining. Chaining is generally one of two forms or a combination of the two forms:

- 1) forward
- 2) backward

Forward chaining starts with a premise to prove a conclusion. The user is asked questions until all the applicable information has been derived from the expert system. The inference engine flows through all the rules sequentially firing all true rules. In this manner it determines a set of all possible conclusions.

Backward chaining starts with an assumed conclusion and shows that the premise is true or false, proving or disproving the assumption. In this form the user specifies a conclusion. The inference engine then tries to prove the premise of the rule containing the conclusion. It does this by finding other rules in the rule base with conclusions that will verify the premise under inspection. The inference engine then tries to prove the premise of the new rule. The rules are thus chained together until the initial premise is proven or disproved. This method is similar to the AI method of *recursion*. Box 4 contains a set of rules taken from Help Desk that are chained together in this manner.

The rules displayed are used in the environmental temperature calculations in Help Desk. The rules shown are only those that will be fired by the expert system when running the analysis. There are more rules available that will calculate the qualifier value for "Thermal analysis is" and a value for the variable [EPSILON]. To simplify the example, only the rules that will be fired are shown.

```

/* RULE NUMBER: 1
IF:
    Thermal analysis is {completed}
THEN:
    X> REPORT(RADIATE.OUT)
and: > Thermal analysis is complete -
    Confidence=1

/* RULE NUMBER: 5
IF:
    Value to calculate is {surface temperature}
and: Surface temperature to be calculated is
    {environmental temperature}
and: [TEMP] > 0
THEN:
    [REPORT] IS GIVEN THE VALUE [REPORT3]
and: Thermal analysis is {completed}

/* RULE NUMBER: 10
IF:
    Value to calculate is {surface temperature}
and: [EB] > 0
and: [SIGMA] > 0.0
THEN:
    [TEMP] IS GIVEN THE VALUE ...

/* RULE NUMBER: 12
IF:
    Surface temperature to be calculated is
    {environmental temperature}
and: [EPSILON]>0
THEN:
    [EB] IS GIVEN THE VALUE ...

/* RULE NUMBER: 22
IF:
    Value to calculate is {radiator area (given
    temperature)}
or: Value to calculate is {surface temperature}
THEN:
    [SIGMA] IS GIVEN THE VALUE...

```

**Box 4 Example of Backward Chaining**

In this example the user wants to prove that the "Thermal analysis is complete" (conclusion of rule 1). The inference engine tries to prove this conclusion using backward chaining as defined above. It starts with Rule 1, which contains the conclusion in question.

EXAMPLE: A decision tree for this example is shown in Figure 2.

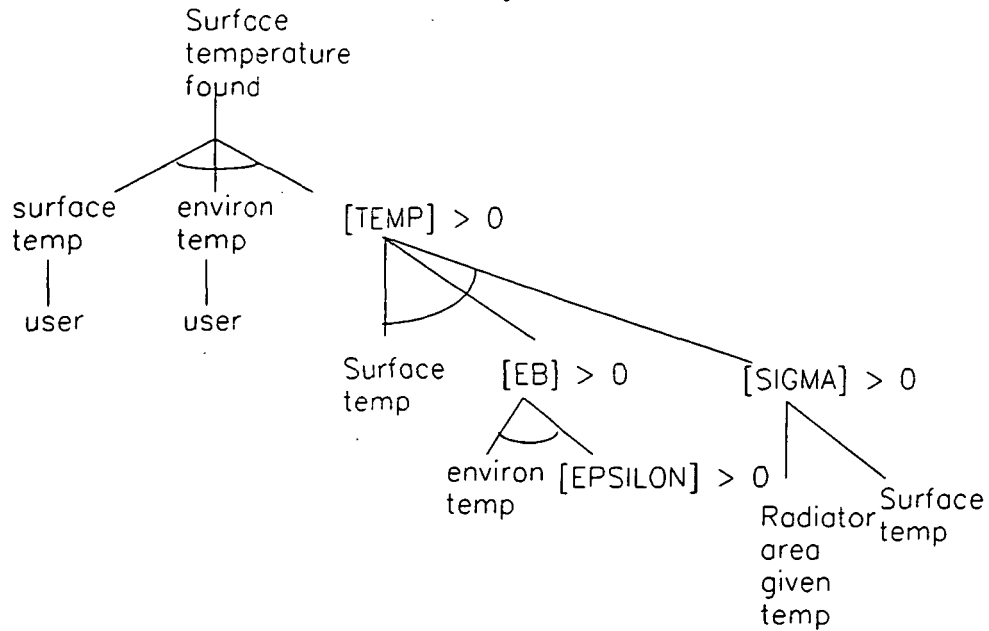


Figure 2 Decision Tree

Rule 1:

Qualifier: "Thermal analysis is"

Is the value defined?

- no: can the value be derived from an additional rule?
- yes: derive using Rule 5.

Rule 5:

Qualifier: "Value to calculate is"

Is the value defined?

- no: can the value be derived from an additional rule?
- no: prompt the user.

Help Desk prompts the user with the Main Menu, which contains a list of the analyses. If the user selects "surface temperature", the inference engine assigns the value "surface temperature" to Q1 and proceeds to the next premise clause.



Qualifier: "Surface temperature to be calculated is"

Is the value defined?

no: can the value be derived from an additional rule?

no: prompt the user.

Help Desk prompts the user with a screen containing two choices:

- 1) Surface temperature (given area)
- 2) Environmental temperature

If the user selects "Environmental temperature" the value is assigned to the qualifier and the inference engine proceeds with the next premise clause. (If the user selects "Surface temperature (given area)" the value is assigned to the qualifier, the rule is marked False, and the inference engine proceeds to the next rule that can satisfy the previous premise.)

Variable: [TEMP] (the temperature of the surface)

Is the value defined?

no: can the value be derived from an additional rule?

yes: derive using Rule 10.

Rule 10:

Qualifier: "value to calculate"

Is the value defined?

yes: Is the premise clause true or false?

true: continue with next premise clause

Variable: [EB] (blackbody radiation)

Is the value defined?

no: can it be derived from an additional rule?

yes: derive using Rule 12

Rule 12:

Qualifier "Surface temperature to be calculated is"

Is the value defined?

yes: Is the premise clause true or false?

true: continue with next premise clause

Variable: [EPSILON] (surface emissivity)

Is the value defined?

no: can it be derived from an additional rule?

no: prompt the user

This user is prompted with a screen that contains a location to input a value for [EPSILON]. The screen also contains positions for the value of other analysis constants.

If the value for [EPSILON] is greater than zero, the rule is marked as True. The THEN clause is fired, defining the value for [EB].

Note: [EPSILON] < 0 has no physical meaning. Help Desk will continue to prompt for this variable until it is properly satisfied. This is due to the variable definition, not the inference engine.

*( Rule 10 continued)*

Once the inference engine is finished with a rule, it returns to the calling rule. Since Rule 10 called Rule 12, control now returns to Rule 10.

Variable: [SIGMA] (Stefan Boltzmann constant)

Is the value defined?

no: can it be derived from an additional rule?  
yes: derive using Rule 22

Rule 22:

Qualifier: "value to calculate is"

Is the value defined?

yes: Is the premise clause true or false?  
true: mark the rule as True and fire the THEN clause.

Rule 22 contains an "or" clause. The first clause is false, but the second "or" clause is true, making the entire premise true. The THEN clause assigns the value to [SIGMA]. Currently EXSYS\* does not handle scientific notation. Therefore, a variable cannot be initialized to such a small value, and consequently the value must be assigned through the use of a rule.

*( Rule 10 continued)*

All the premise clauses for Rule 10 are defined and True. The inference engine marks the rule as True and fires the THEN clause, which defines the value for [TEMP].

*(Rule 5 continued)*

All the premise clauses for Rule 5 are defined. If the variable [TEMP] > 0, then the entire premise is True. The inference engine marks the rule as True and fires the THEN clause. The THEN clause defines the variable [REPORT] (which displays a screen containing the report information) and assigns the value "completed" to the Qualifier "Thermal analysis is".

*(Rule 1 continued)*

The premise for Rule 1 is derived and True. The inference engine marks Rule 1 as true and fires the THEN clause. The THEN clause executes the report generator found in the file RADIATE.OUT, and defines "Thermal Analysis" as "complete", proving the conclusion.

**This brief description of Help Desk and the inference engine should be sufficient for a cursory examination for the knowledge and rule bases of Help Desk. For more information on expert systems and chaining, refer to the text in References 3,4 and 5.**

## 2 MATHEMATICAL APPROACH

The following section covers the derivations of the equations used in Help Desk.

### 2.1 Environmental Heat Balance on a Surface:

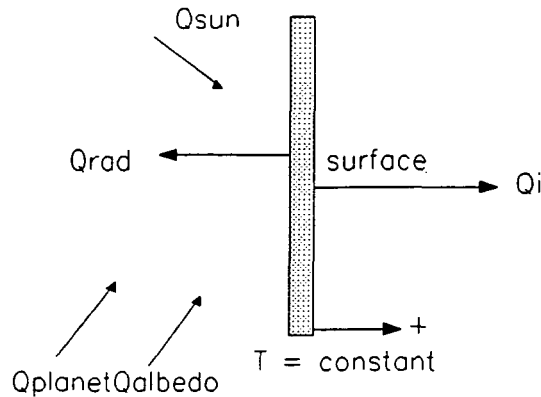


Figure 3 Heat Balance on a Surface

The net heat flux through the surface,  $Q_i$ , equals the sum of the environmental effects on the surface minus the heat radiated from the surface,  $Q_{RAD}$ . The environmental effects include the radiation on the surface due to the sun,  $Q_{SUN}$ ; the planet,  $Q_{PLANET}$ ; and the albedo of the planet,  $Q_{ALBEDO}$ .

$$Q_i = Q_{SUN} + Q_{PLANET} + Q_{ALBEDO} - Q_{RAD} \quad \text{Equation 1}$$

The surface of a radiator must be able to remove all of the environmental heat loads plus any heat load generated by internal equipment,  $Q_{INTERNAL}$ . An example of an internal heat load is the energy that must be removed to cool an instrument to a defined temperature. The addition of this term makes the above equation:

$$Q_i = Q_{SUN} + Q_{PLANET} + Q_{ALBEDO} + Q_{INTERNAL} - Q_{RAD} \quad \text{Equation 2}$$

For a single surface,  $Q_i$  is zero when the surface is in equilibrium.

### 2.2 Surface Heat Loads

$Q_{sun}$  is the heat load on the surface due to the radiation of the sun as defined in Equation 3. It is assumed that the solar rays are essentially parallel in the vicinity of a planet, and consequently solar flux is constant throughout an orbit (except for planetary shadowing effects). However, variations in flux that are due to yearly variations in the distance from the sun are accounted for with the user defined solar flux value. Help Desk also calculates changes associated with the surface properties.

$$Q_{SUN} = A_f S \mu_i \alpha_s \quad \text{Equation 3}$$

$\alpha_s$ , solar absorptivity      Fraction of the total solar energy (in the 0.2 to 2.8  $\mu\text{m}$  range) that is incident on a surface and is absorbed by that surface.

S      Solar flux,  $\text{W}/\text{m}^2$       The radiative flux due to the sun. Varies as the inverse square of the distance from the sun, which changes throughout the a year.

$\mu_i$  solar view factor Fraction of the surface field of view that sees the sun due to variations in the angle of incidence of the surface. See Section 5.2, Figure 7 for more information.

$A_i$  surface area,  $m^2$  Area of the surface under investigation.

$Q_{planet}$  is the heat load on the surface due to the thermal radiation from the planet as defined in Equation 4. The surface of a planet absorbs energy from the sun and emits it as thermal radiation in accordance with the Stefan-Boltzmann law. It is assumed that thermal radiation is emitted from a planet at a constant rate. However, it does vary with the surface infrared absorptivity, the area of the surface, and the view factor from the surface to the planet.

$$Q_{PLANET} = A_i F_{Ei} E \alpha_{IR} \quad \text{Equation 4}$$

$E$  Earth flux,  $W/m^2$  Radiative flux of the planet. A portion of the solar radiation incident on the planet is reemitted as thermal radiation.

$F_{Ei}$  planet view factor: Fraction of the surface that sees the radiation from the planet. For example, in Low Earth Orbit, fully facing the planet,  $F_E \sim .88$ , due to the surface seeing space off to the sides of the field of view. See Section 5.3 Figure 11 for more information.

$\alpha_{IR}$  Infrared absorptivity Fraction of the total energy in the infrared wavelength band (5 to 50  $\mu m$ ) incident on a surface, that is absorbed by that surface. *Assuming* a gray body, this is equal to the emissivity of the surface.

$A_i$  surface area,  $m^2$  Area of the surface under investigation.

$Q_{albedo}$  is the heat load incident on the surface due to the solar energy that is reflected off the surface of a planet as defined in Equation 5. The reflection is due to the planet's surface and the atmospheric conditions. It is assumed that the albedo is constant over the planet's surface. It is a function of the surface infrared absorptivity, the area of the surface, the view factor from the surface to the planet, and the view factor to the sun.

$$Q_{ALBEDO} = A_i F_{Ei} (1 - \mu_i) S_{ALB} \alpha_s \quad \text{Equation 5}$$

$ALB$  albedo coefficient Fraction of the total solar radiation that is reflected off the planet's surface. Depends on the atmospheric conditions of the planet. For Earth it ranges from .1 to .8 with an average of .3.

$E$  Earth flux,  $W/m^2$  Radiative flux of the planet. A portion of the solar radiation incident on the planet is re-emitted as thermal radiation.

$S$  Solar flux,  $W/m^2$  The radiative flux due to the sun. Varies as the inverse square of the distance from the sun, which changes throughout the year.

$(1 - \mu_i)$  view factor Approximation for the reflected view factor.

$F_{Ei}$  planet view factor Fraction of the surface that sees the radiation from the planet. For example, in Low Earth Orbit and when fully facing the planet,  $F_E \sim .88$ . This is due to the fact that the surface sees space near the edges of its field of view. See Section 5.3, Figure 11 for more information.

$\alpha_{IR}$  Infrared absorptivity Fraction of the total solar energy in the infrared wavelength band (5 to 50  $\mu\text{m}$ ) incident on a surface that is absorbed by that surface. Assuming a gray body, this is equal to the emissivity of the surface.

$A_i$  surface area,  $\text{m}^2$  Area of the surface under investigation.

$Q_{\text{internal}}$  is the internal heat load that must be removed from the system. Examples of the internal heat load are:

- The electrical power of the internal systems that must be removed from the system
- The heat load on an instrument due to an enclosure. This can be calculated using the Enclosure analysis discussed later in this document.

$Q_{\text{rad}}$  is the heat radiated from the surface as defined in Equation 6. It depends on the blackbody radiation of the surface, the surface properties, and the surface area.

$$Q_{RAD} = A_i \epsilon_i \sigma_i T_i^4 = A_i \epsilon_i E_b \quad \text{Equation 6}$$

$E_{bi}$  blackbody radiation Emitted power in  $\text{W}/\text{m}^2$  for a body that absorbs all incident radiation.

$T_i$  temperature Temperature of the surface in K.

$\sigma$  Stefan-Boltzmann constant =  $6.78\text{E-}8$

$\epsilon$  emissivity The ability of a surface to radiate energy compared with a blackbody. For example, the emissivity of a blackbody is 1. Assuming a gray body this is equal to the infrared absorptivity of the surface.

$A_i$  surface area,  $\text{m}^2$  Area of the surface under investigation.

### 2.3 Help Desk Surface Analyses

In this subsection a single surface is considered, and the  $i$  subscript is omitted

#### 2.3.1 Surface Area

The surface is in equilibrium. Therefore, the net heat load through the surface is equal to zero,  $Q = 0$ .

Solve Equation 2 for the surface area,  $A$ :

$$0 = AS\mu\alpha_s + AF_E E\alpha_{IR} + AF_E(1-\mu)S ALB \alpha_s + Q_{INTERNAL} - AeE_b$$

$$A = \frac{(-Q_{INTERNAL})}{(S\mu\alpha_s + F_E E\alpha_{IR} + F_E(1-\mu)S ALB \alpha_s \epsilon E_b)} \quad \text{Equation 7}$$

Equation 7 is the basis for the "Surface Area (Given Temperature)" option of the Thermal Analyst's Help Desk.

#### 2.3.2 Surface Temperature

The surface is in equilibrium. Therefore, the net heat load through the surface is equal to zero,  $Q = 0$ .

Solve Equation 2 for the surface temperature, T (or the blackbody radiation,  $E_b$  since  $E_b = \sigma T^4$ ):

$$0 = AS\mu\alpha_s + AF_E E\alpha_{IR} + AF_E(1-\mu)S ALB \alpha_s + Q_{INTERNAL} - AeE_b$$

$$E_b = \sigma T^4$$

$$E_b = \frac{1}{\epsilon} \left( S\mu\alpha_s + F_E E\alpha_{IR} + F_E(1-\mu)S ALB \alpha_s + \frac{Q_{INTERNAL}}{A} \right) \quad \text{Equation 8}$$

Equation 8 is the basis for the "Surface Temperature (given Area)" option of the Thermal Analyst's Help Desk.

### 2.3.3 Environmental Temperature

When there are only environmental effects on the surface due to the sun, planet and albedo, the surface is in equilibrium. The total heat load through the surface is equal to zero,  $Q = 0$ . Solve Equation 2 for the surface temperature, T (or the blackbody radiation,  $E_b$  since  $E_b = \sigma T^4$ ):

$$0 = AS\mu\alpha_s + AF_E E\alpha_{IR} + AF_E(1-\mu)S ALB \alpha_s - AeE_b$$

$$E_b = \sigma T^4$$

$$E_b = \frac{1}{\epsilon} (S\mu\alpha_s + F_E E\alpha_{IR} + F_E(1-\mu)S ALB \alpha_s) \quad \text{Equation 9}$$

Equation 9 is the basis for the "Surface Temperature (given Area)" option of the Thermal Analyst's Help Desk.

## 2.4 Enclosure Heat Loads

### 2.4.1 Heat Balance for Each Surface of an Enclosure

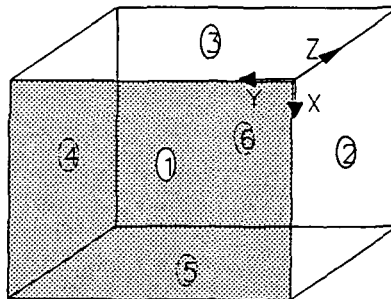


Figure 4 Enclosure Geometry and Surface Numbers

Expand Equation 2 for each surface of a six sided rectangular box shown in Figure 4.

1: South facing	$Q_1 = A_1 S \mu_1 \alpha_{s1} + A_1 F_{1E} E \alpha_{IR1} + A_1 F_{1E} (1-\mu_1) S ALB \alpha_{s1} - A_1 \epsilon_1 E_{b1}$
2: Right	$Q_2 = A_2 S \mu_2 \alpha_{s2} + A_2 F_{2E} E \alpha_{IR2} + A_2 F_{2E} (1-\mu_2) S ALB \alpha_{s2} - A_2 \epsilon_2 E_{b2}$
3: Top	$Q_3 = A_3 S \mu_3 \alpha_{s3} + A_3 F_{3E} E \alpha_{IR3} + A_3 F_{3E} (1-\mu_3) S ALB \alpha_{s3} - A_3 \epsilon_3 E_{b3}$
4: Left	$Q_4 = A_4 S \mu_4 \alpha_{s4} + A_4 F_{4E} E \alpha_{IR4} + A_4 F_{4E} (1-\mu_4) S ALB \alpha_{s4} - A_4 \epsilon_4 E_{b4}$
5: Bottom	$Q_5 = A_5 S \mu_5 \alpha_{s5} + A_5 F_{5E} E \alpha_{IR5} + A_5 F_{5E} (1-\mu_5) S ALB \alpha_{s5} - A_5 \epsilon_5 E_{b5}$
6: North facing	$Q_6 = A_6 S \mu_6 \alpha_{s6} + A_6 F_{6E} E \alpha_{IR6} + A_6 F_{6E} (1-\mu_6) S ALB \alpha_{s6} - A_6 \epsilon_6 E_{b6}$

In matrix form:

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \\ Q_6 \end{bmatrix} = \begin{bmatrix} Q_{SUN1} + Q_{EARTH1} + Q_{ALB1} \\ Q_{SUN2} + Q_{EARTH2} + Q_{ALB2} \\ Q_{SUN3} + Q_{EARTH3} + Q_{ALB3} \\ Q_{SUN4} + Q_{EARTH4} + Q_{ALB4} \\ Q_{SUN5} + Q_{EARTH5} + Q_{ALB5} \\ Q_{SUN6} + Q_{EARTH6} + Q_{ALB6} \end{bmatrix} + \begin{bmatrix} -X_{RAD1} & 0 & 0 & 0 & 0 & 0 \\ 0 & -X_{RAD2} & 0 & 0 & 0 & 0 \\ 0 & 0 & -X_{RAD3} & 0 & 0 & 0 \\ 0 & 0 & 0 & -X_{RAD4} & 0 & 0 \\ 0 & 0 & 0 & 0 & -X_{RAD5} & 0 \\ 0 & 0 & 0 & 0 & 0 & -X_{RAD6} \end{bmatrix} \begin{bmatrix} E_{b1} \\ E_{b2} \\ E_{b3} \\ E_{b4} \\ E_{b5} \\ E_{b6} \end{bmatrix} \quad \text{Equation 10}$$

or:

$$\bar{Q} = \bar{X} + \bar{X}_{RAD} \bar{E}_b \quad \text{Equation 11}$$

Where :

**X** contains the environmental effects.

**XRAD** contains the area times the emissivity

#### 2.4.2 Enclosure Heat Balance

In Section 2.4.1 the heat balance is performed on the external surfaces of the enclosure. In this section the heat balance is performed on the internal surfaces of an enclosure. It is assumed that the temperature is constant across an enclosure wall.

An enclosure heat balance is the net heat load on an internal surface of an enclosure due to the other surfaces of the same enclosure, and to itself if it is concave. A heat balance on single surface *i* yields Equation 13. For the enclosure there are six such equations having values for *i* equal to one through six. The heat balance includes energy arriving on surface *A<sub>i</sub>* from all surfaces of the enclosure, including *A<sub>i</sub>* if it is a concave surface.

**NOTE:** The following equations include the capability for a concave surface. However, Help Desk currently contains only a rectangular box enclosure. Therefore, there are no concave surfaces.

$$\sum_{j=1}^N \left( \frac{\delta_{ij}}{e_j} - F_{i-j} \frac{1-e_j}{e_j} \right) \frac{Q_j}{A_j} = \sum_{j=1}^N (\delta_{ij} - F_{i-j}) \sigma T_j^4 \quad \text{Equation 12}$$

in a different form:

$$Q_i = \sigma T_i^4 A_i - \sum_{j=1}^N \sigma T_j^4 A_j F_{j-i} \quad \text{Equation 13}$$

which simplifies to:

$$Q_i = \sigma A_i \sum_{j=1}^N (T_i^4 - T_j^4) F_{i-j} = Q_i = A_i \sum_{j=1}^N (E_{b_i} - E_{b_j}) F_{i-j} \quad \text{Equation 14}$$

Expand for each of the six surfaces:

$$\begin{aligned} Q_1 &= A_1 [(E_{b_1} - E_{b_1})F_{11} + (E_{b_1} - E_{b_2})F_{12} + (E_{b_1} - E_{b_3})F_{13} + (E_{b_1} - E_{b_4})F_{14} + (E_{b_1} - E_{b_5})F_{15} + (E_{b_1} - E_{b_6})F_{16}] \\ Q_2 &= A_2 [(E_{b_2} - E_{b_1})F_{21} + (E_{b_2} - E_{b_2})F_{22} + (E_{b_2} - E_{b_3})F_{23} + (E_{b_2} - E_{b_4})F_{24} + (E_{b_2} - E_{b_5})F_{25} + (E_{b_2} - E_{b_6})F_{26}] \\ Q_3 &= A_3 [(E_{b_3} - E_{b_1})F_{31} + (E_{b_3} - E_{b_2})F_{32} + (E_{b_3} - E_{b_3})F_{33} + (E_{b_3} - E_{b_4})F_{34} + (E_{b_3} - E_{b_5})F_{35} + (E_{b_3} - E_{b_6})F_{36}] \\ Q_4 &= A_4 [(E_{b_4} - E_{b_1})F_{41} + (E_{b_4} - E_{b_2})F_{42} + (E_{b_4} - E_{b_3})F_{43} + (E_{b_4} - E_{b_4})F_{44} + (E_{b_4} - E_{b_5})F_{45} + (E_{b_4} - E_{b_6})F_{46}] \\ Q_5 &= A_5 [(E_{b_5} - E_{b_1})F_{51} + (E_{b_5} - E_{b_2})F_{52} + (E_{b_5} - E_{b_3})F_{53} + (E_{b_5} - E_{b_4})F_{54} + (E_{b_5} - E_{b_5})F_{55} + (E_{b_5} - E_{b_6})F_{56}] \\ Q_6 &= A_6 [(E_{b_6} - E_{b_1})F_{61} + (E_{b_6} - E_{b_2})F_{62} + (E_{b_6} - E_{b_3})F_{63} + (E_{b_6} - E_{b_4})F_{64} + (E_{b_6} - E_{b_5})F_{65} + (E_{b_6} - E_{b_6})F_{66}] \end{aligned}$$

where

$E_{b_i}$  is the blackbody radiation intensity for the surface

$F_{ji}$  is the view factor from surface  $j$  to surface  $i$

Remove the zero terms ( $F_{ii}$  coefficients) and collect coefficients for the  $E_{b_i}$  terms:

$$\begin{aligned} Q_1 &= A_1 [E_{b_1}(F_{12} + F_{13} + F_{14} + F_{15} + F_{16}) - E_{b_2}F_{12} - E_{b_3}F_{13} - E_{b_4}F_{14} - E_{b_5}F_{15} - E_{b_6}F_{16}] \\ Q_2 &= A_2 [-E_{b_1}F_{21} + E_{b_2}(F_{21} + F_{23} + F_{24} + F_{25} + F_{26}) - E_{b_3}F_{23} - E_{b_4}F_{24} - E_{b_5}F_{25} - E_{b_6}F_{26}] \\ Q_3 &= A_3 [-E_{b_1}F_{31} - E_{b_2}F_{32} + E_{b_3}(F_{31} + F_{32} + F_{34} + F_{35} + F_{36}) - E_{b_4}F_{34} - E_{b_5}F_{35} - E_{b_6}F_{36}] \\ Q_4 &= A_4 [-E_{b_1}F_{41} - E_{b_2}F_{42} - E_{b_3}F_{43} + E_{b_4}(F_{41} + F_{42} + F_{43} + F_{45} + F_{46}) - E_{b_5}F_{45} - E_{b_6}F_{46}] \\ Q_5 &= A_5 [-E_{b_1}F_{51} - E_{b_2}F_{52} - E_{b_3}F_{53} - E_{b_4}F_{54} + E_{b_5}(F_{51} + F_{52} + F_{53} + F_{54} + F_{56}) - E_{b_6}F_{56}] \\ Q_6 &= A_6 [-E_{b_1}F_{61} - E_{b_2}F_{62} - E_{b_3}F_{63} - E_{b_4}F_{64} - E_{b_5}F_{65} + E_{b_6}(F_{61} + F_{62} + F_{63} + F_{64} + F_{65})] \end{aligned}$$

In matrix form:

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \\ Q_6 \end{bmatrix} = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & A_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & A_6 \end{bmatrix} \begin{bmatrix} \sum_{j=1}^n F_{1-j} & -F_{12} & -F_{13} & -F_{14} & -F_{15} & -F_{16} \\ -F_{21} & \sum_{j=1}^n F_{2-j} & -F_{23} & -F_{24} & -F_{25} & -F_{26} \\ -F_{31} & -F_{32} & \sum_{j=1}^n F_{3-j} & -F_{34} & -F_{35} & -F_{36} \\ -F_{41} & -F_{42} & -F_{43} & \sum_{j=1}^n F_{4-j} & -F_{45} & -F_{46} \\ -F_{51} & -F_{52} & -F_{53} & F_{54} & \sum_{j=1}^n F_{5-j} & -F_{56} \\ -F_{61} & -F_{62} & -F_{63} & F_{64} & -F_{65} & \sum_{j=1}^n F_{6-j} \end{bmatrix} \begin{bmatrix} E_{b_1} \\ E_{b_2} \\ E_{b_3} \\ E_{b_4} \\ E_{b_5} \\ E_{b_6} \end{bmatrix} \quad \text{Equation 15}$$

The sum of all view factors on one side of the surface equals 1.

$$\sum_{j=1}^n F_{1-j} = 1 \quad \text{Equation 16}$$



Substitute into Equation 15

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \\ Q_6 \end{bmatrix} = \begin{bmatrix} A_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & A_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & A_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & A_6 \end{bmatrix} \begin{bmatrix} 1 & -F_{12} & -F_{13} & -F_{14} & -F_{15} & -F_{16} \\ -F_{21} & 1 & -F_{23} & -F_{24} & -F_{25} & -F_{26} \\ -F_{31} & -F_{32} & 1 & -F_{34} & -F_{35} & -F_{36} \\ -F_{41} & -F_{42} & -F_{43} & 1 & -F_{45} & -F_{46} \\ -F_{51} & -F_{52} & -F_{53} & F_{54} & 1 & -F_{56} \\ -F_{61} & -F_{62} & -F_{63} & F_{64} & -F_{65} & 1 \end{bmatrix} \begin{bmatrix} E_{b1} \\ E_{b2} \\ E_{b3} \\ E_{b4} \\ E_{b5} \\ E_{b6} \end{bmatrix} \quad \text{Equation 17}$$

Simplify

$$\bar{Q} = \bar{A} \bar{F} \bar{E}_b \quad \text{Equation 18}$$

Solve Equation 11 and Equation 18 for heat load,  $\bar{Q}$ , and blackbody radiation,  $\bar{E}_b$ . The most direct solution is to set Equation 11 and Equation 18 equal and solve for  $\bar{E}_b$  (Equation 19).

$$\bar{X} = [-\bar{X}RAD + \bar{A} \bar{F}] \bar{E}_b \quad \text{Equation 19}$$

This is in the form  $Ax = b$ . Solve for  $\bar{E}_b$  using  $Ax = b$ , and then solve for  $\bar{Q}$  using Equation 18.

This derivation is valid, but limiting in the analysis. Since  $\bar{Q}$  is not explicitly stated in the equation, a single side cannot be constrained to have zero net heat flux ( $Q_1 = 0$ ) across it. This feature is necessary to decouple one side from the rest of the enclosure.

Help Desk allows one or more sides of an enclosure to be decoupled from the environmental effects. A decoupled side is one with no net heat flux across it. An example is the side of an enclosure (or spacecraft bus) that has the thermal radiator attached to it. There is no net heat flux from the enclosure through the side with the radiator. The side is effectively decoupled from the enclosure and is subject to its own environmental fluxes and effects. Physically this is accomplished with insulation between the radiator and enclosure surface.

A different formulation results when Equation 11 is solved for  $\bar{E}_b$ , placed into Equation 18 and solved for  $\bar{Q}$ .

$$\bar{E}_b = \overline{XRAD}^{-1} [\bar{Q} - \bar{X}] \quad \text{Equation 20}$$

$$\left[ \bar{A} \bar{F} \overline{XRAD}^{-1} - [1] \right] \bar{Q} = \bar{A} \bar{F} \overline{XRAD}^{-1} \bar{X} \quad \text{Equation 21}$$

This is also in the form  $Ax = b$ . This form has  $\bar{Q}$  explicitly stated allowing one term to be constrained to zero.

- A** area matrix (known)
- F** enclosure view factor matrix (known)
- XRAD** matrix containing the heat radiated from each surface
- X** total environmental heat load on each surface
- Q** net heat load through a surface

To constrain one or more surfaces to have 0 net heat flux across them, set the XRAD term equal to 9.9E+9. When the inverse is taken the coefficient of Q will be zero. This also makes the right hand side of the equation for Q equal to zero.

Equation 21 is the basis for both the Enclosure Temperature and Heat Loads analysis and for the Orbital Temperatures and Heat Loads analysis.

This concludes the fundamental mathematical approach to the analyses in the Thermal Analyst's help Desk. The following sections discuss the procedures and equations used to derive the terms found in the main equations described above.

### 3 SURFACE PROPERTIES

Solar absorptivity and infrared emissivity are two physical parameters that describe the surface properties. Solar absorptivity describes the amount of ultraviolet radiation that is absorbed into the surface. Infrared emissivity is the amount of infrared radiation that is emitted from a surface. Assuming a gray body, the infrared absorptivity is equal to the infrared emissivity.

These parameters depend on the surface material. For example, white paint has an absorptivity of 0.2 and an emissivity of 0.9, while black paint has an absorptivity of 0.9 and an emissivity of 0.9. The surface properties also depend on the time in the life of the material that is used in the analysis. In the space environment surface degradation occurs and is a major factor in the determination of the surface capabilities. Performance parameters are generally specified for both beginning of life (BOL) and end of life (EOL). When choosing a surface a value is selected from manufacturers specifications, and entered in the appropriate box in Help Desk.

Help Desk allows the user another option. A table containing the surface properties of several surface materials is contained in an AI tool called a "Frame". By definition a frame contains an "object" plus "slots" for related information. In this manner it is very similar to a small database, and it allows reference to a "parent" or "child" frame. A parent or child frame contains information about the object: parent information is more general and child information is more specific. This referencing is called inheritance.

When the frame is referenced it compares a user defined text value describing the surface application, or the surface material, to either the Surface field or to the Typical\_application field. Once a match has been found the value for solar absorptance and emittance are returned to Help Desk.

**NOTE:** The frame call returns the first occurrence of a correct match. If a different value is preferred, the user may enter it directly during an analysis.

The frame for the surface properties is displayed below.

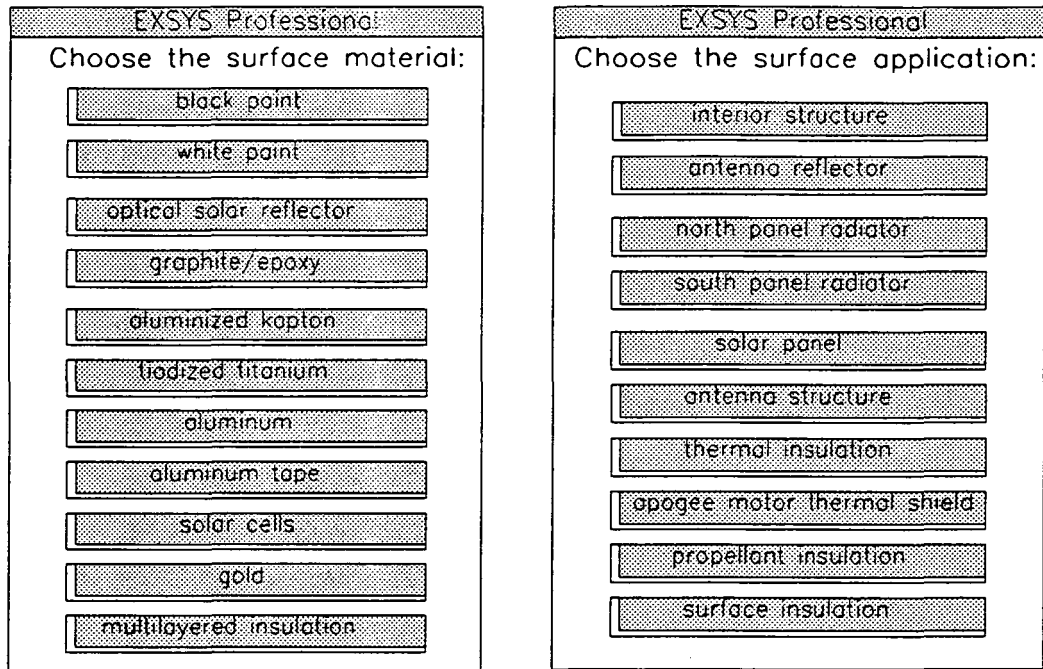
Surface	Typical Application	Solar Absorptance		Emittance	
		BOL	EOL	BOL	EOL
black paint	interior structure	0.90	0.90	0.90	0.90
white paint	antenna reflector	0.20	0.60	0.90	0.90
optical solar reflector	north panel radiator	0.08	0.21	0.80	0.80
optical solar reflector	south panel radiator	0.08	0.21	0.80	0.80
graphite/epoxy	solar panel	0.84	0.84	0.85	0.85
graphite/epoxy	antenna structure	0.84	0.84	0.85	0.85
aluminized kapton	thermal insulation	0.35	0.50	0.60	0.60
tiodized titanium	boogee motor thermal shield	0.60	0.60	0.60	0.60
aluminum	propellant insulation	0.12	0.18	0.06	0.06
aluminum tape	propellant insulation	0.12	0.18	0.06	0.06
deposited aluminum	propellant insulation	0.12	0.18	0.06	0.06
anodized aluminum	interior structure	0.20	0.60	0.80	0.80
solar cells	solar panels	0.70	0.70	0.82	0.82
gold		0.25	0.25	0.05	0.05
multilayered insulation	surface insulation	0.05	0.05	0.05	0.05
BOL, Beginning of Life; EOL, End of Life, 7 years					
surface data taken from: reference 1 Page 275					

**TABLE 1: Surface Properties Frame**

### 3.1 Updating Help Desk surface properties option

The following procedure may be used to add more information to the frame:

- use an ASCII editor to open the file "SURFACE.TXT" in the directory containing Help Desk files.
- Enter another line of data in any location, making sure to exactly repeat the format of the other lines.
- save and exit



**Figure 5** Surface Material and Application screens

To update the user interface for the Surface Properties option, two screens must also be modified. The first screen contains the list of surface materials, and the second screen contains the list of surface applications (Figure 5). One more box must be added to each screen for each new material and surface application pair in the frame. For more information on updating screen files, see Section 2.



## 4 ORBITAL INFORMATION

The user is not constrained to perform a thermal analysis about the Earth. Currently the equations are valid for any elliptical or parabolic conic section for a "two body problem". A two body problem has two assumption:

- 1) Each body is spherically symmetric. The mass can be considered as a single point mass at the geometric center of the body.
- 2) There are no external or internal forces acting on the system other than those defined by the two masses.

The environment of the system is dependent on orbit specific parameters.

- orbital position
- spacecraft orientation
- planet heat flux
- solar flux

The user specifies the values for the spacecraft orbit in terms of the six classical orbital elements:

<b>inclination:</b>	angle between the momentum vector and the global Z axis.
<b>nodal crossing</b>	(longitude of ascending node) the angle between the vernal equinox direction and the line of nodes, in the equatorial plane.
<b>argument of periapsis:</b>	the angle between the line of nodes and the direction of periapsis, in the orbit plane.
<b>true anomaly at epoch:</b>	the angle between the direction of periapsis and the position of the spacecraft at a specific time.
<b>semi-major axis:</b>	defines the size of the conic section.
<b>eccentricity:</b>	defines the shape of the conic section (0 for a circle; >0 and <1 for an ellipse; 1 for a parabola; >1 for a hyperbola).

Four parameters associated with the orbit are also used in this analysis:

<b>Planet Flux</b>	Heat load per unit area incident on the surface due to the planet's thermal radiation.
<b>Solar Flux:</b>	Heat load per unit area incident on the surface due to the sun's radiation.
<b>Albedo coefficient</b>	Percentage of the total solar radiation incident on the planet, that is reflected back to space.
<b>Planet Radius:</b>	Radius of the planet under investigation.

For a given a planet the user may select values for the above variables and enter them into the proper boxes. Alternatively, Help Desk allows the user access to its own table called a "frame".

The frame for the planet information is displayed below. The values in the frame are mean values for planet flux, solar flux and albedo.

PLANET	PLANET_FLUX	SOLAR_FLUX	ALBEDO	RADIUS
MERCURY	4201	8920	.058	2487
VENUS	154.2	2570	.76	6187
EARTH	236.6	1352	.3	6378
MOON	603.25	1352	.047	1738
MARS	123	577.3	.140	3380
JUPITER	6.1	49.6	.51	71370
SATURN	1.8	14.7	.5	60400
URANUS	.31	3.65	.66	23530
NEPTUNE	.14	1.48	.62	22320
PLUTO	.0	0	.0	2284

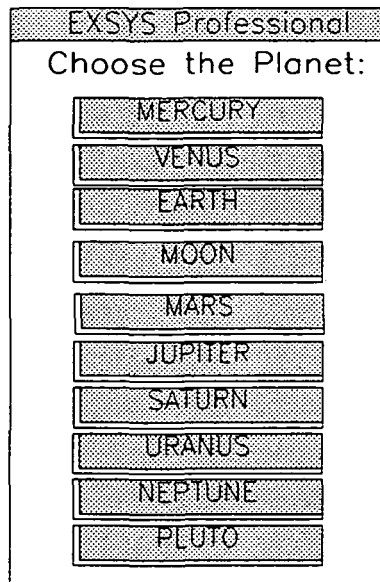
**TABLE 2: Different Planet Frame**

The following procedure may be followed to add more information to the frame:

- use an ASCII editor to open the file "PLANET.TXT", which is in the directory containing the Help Desk files.
- Enter another line containing the information for an additional astronomical body (e.g. the sun) in any location and duplicate the displayed format.
- save and exit

To update the "different planet" option, the screen containing the list of planets must also be modified (Figure 5). One more box must be added to each screen to account for the new astronomical body. For more information on updating screen files see Section 2.





**Figure 6** Planet Choice screen

## 5 VIEW FACTORS

Three sets of view factors are calculated:

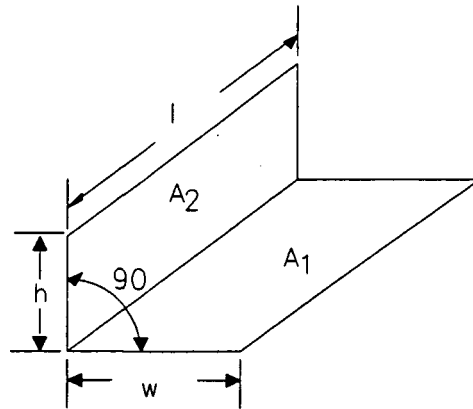
- 1) enclosure view factors dependant on enclosure geometry
- 2) solar view factors dependant on the spacecraft orbit and orientation
- 3) planet view factors dependant on the spacecraft orbit and orientation

### 5.1 Enclosure View Factors

Enclosure view factors are the fraction of the total field of view of one side of an enclosure that sees each of the other sides of the same enclosure. If the surface under examination is concave, it includes the fraction in which it sees itself.

NOTE: Help Desk currently contains only a rectangular box enclosure, therefore there are no concave surfaces.

The enclosure view factors for Help Desk,  $F_{ji}$ , are calculated using an equation from reference 6 Appendix C page 825. Equation 22 is a corrected version of the published equation. The last two terms in Equation 23 are not raised to any power.



**Figure 7** Two finite rectangles of same length, having one common edge, and at an angle of 90° to each other

$$F_{1-2} = \frac{1}{\pi W} \left( W \tan^{-1} \frac{1}{W} + H \tan^{-1} \frac{1}{H} - \sqrt{H^2 + W^2} \tan^{-1} \frac{1}{\sqrt{H^2 + W^2}} + \frac{1}{4} \ln(HW) \right) \quad \text{Equation 22}$$

where:  $H = h/l$        $W = w/l$

$$HW = \frac{(1+W^2)(1+H^2)}{1+W^2+H^2} \left[ \frac{W^2(1+W^2+H^2)}{(1+W^2)(W^2+H^2)} \right] \left[ \frac{H^2(1+W^2+H^2)}{(1+H^2)(W^2+H^2)} \right] \quad \text{Equation 23}$$

Due to the geometry of an enclosure only a few view factors for each surface of an enclosure must be solved this way, the remaining can be solved using the following two equations:

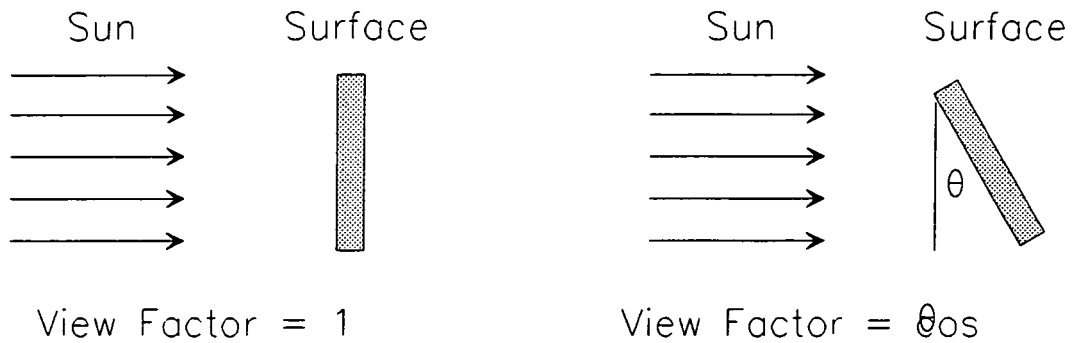
$$A_i F_{ij} = A_j F_{ji}$$

$$\sum F_{ij} = 1$$

$A_i$  is the area of surface  $i$   
 $F_{ij}$  is the view factor from  $i$  to  $j$

### 5.2 Solar View Factors

The solar view factor is the fraction of the total radiation emitted by the sun that is directly incident on the surface. (e.g. 1: fully facing sun, 0: parallel to the sun's rays, 0: facing away from the sun).

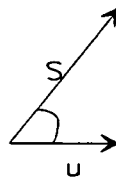


**Figure 8** Solar View Factor Geometry

The solar view factors ( $\mu_i$ ) are found by taking the dot product of the sun vector in local coordinates and the unit normal vector of each side. The dot product is:

$$\vec{S} \cdot \vec{u} = |\vec{S}| |\vec{u}| \cos \theta$$

**Equation 24**



**Figure 9** Dot Product Geometry

The solar view factor,  $\mu$ , is defined as  $\cos \theta$ , where  $\theta$  is the angle between the sun vector and the unit normal.

Therefore:

$$\mu = \frac{\vec{S} \cdot \vec{u}}{|\vec{S}|}$$

Equation 25

In local coordinates the unit normal vectors are constant.

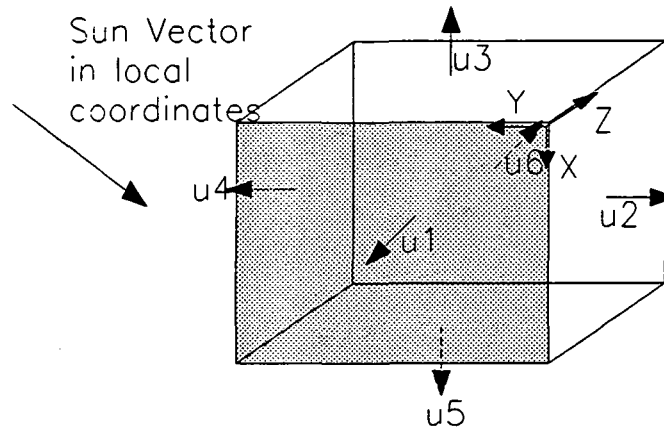


Figure 10 Enclosure Unit Normals Geometry

$$\begin{aligned} u_1 &= 0i + 0j - 1k \\ u_2 &= 0i - 1j + 0k \\ u_3 &= -1i + 0j + 0k \\ u_4 &= 0i + 1j + 0k \\ u_5 &= 1i + 0j + 0k \\ u_6 &= 0i + 0j + 1k \end{aligned}$$

where i, j and k are the direction vectors along the X, Y and Z directions respectively.

The sun vector in local coordinates is defined as:

$$S = X_s i + Y_s j + Z_s k$$

where i,j,k are the unit normals in the coordinate directions, X,Y,Z respectively.

The magnitude of S is:

$$MAGS = \text{SQRT}(X_s^2 + Y_s^2 + Z_s^2)$$

The solar view factors in local coordinates are:

$$\mu_1 = -Z_j/\text{MAGS}$$

$$\mu_2 = -Y_j/\text{MAGS}$$

$$\mu_3 = -X_j/\text{MAGS}$$

$$\mu_4 = Y_j/\text{MAGS}$$

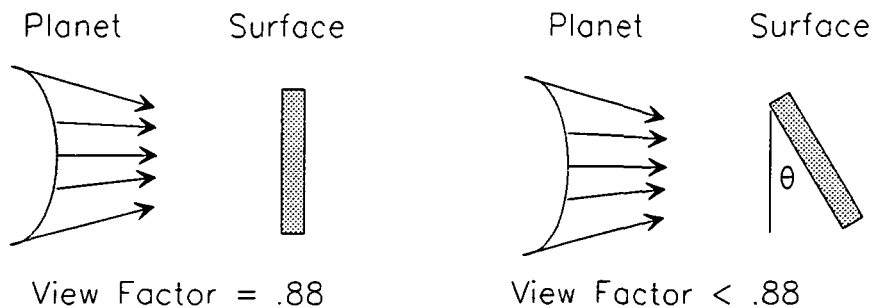
$$\mu_5 = X_j/\text{MAGS}$$

$$\mu_6 = Z_j/\text{MAGS}$$

They are constant in local coordinates since the coordinate axis stays fixed with respect to the enclosure.

### 5.3 Planet View Factors

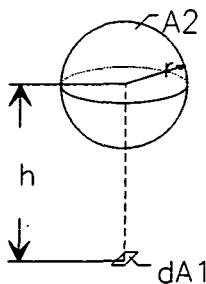
Planet view factors are the fraction of the total thermal radiation emitted by the planet that is directly incident on the surface. (e.g. for Low Earth Orbit:  $\sim .88$  for fully facing planet,  $.287$  for perpendicular to the planet's surface,  $0$  for facing away from the planet). It is not one, because the field of view out each side, sees space. See Figure 11.



**Figure 11** Planet View Factor Geometry

Planet view factors have normal and tangential components. These terms are calculated using an equation from reference 12 Appendix C page 829.

The geometry for the normal component is shown in Figure 12. Equation 26 shows the normal component for the planet view factors.

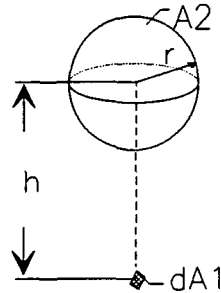


**Figure 12** Plane element  $dA_1$  to sphere of radius  $r$ ; normal to center of element passes through center of sphere

$$F_{d1-2} = \left(\frac{r}{h}\right)^2$$

Equation 26

The geometry for the tangential component is shown in Figure 13. Equation 28 shows the tangential component for the planet view factors.



**Figure 13** Plane element dA1 to sphere of radius r; tangent to element passes through center of sphere

$$H = \frac{h}{r}$$

Equation 27

$$F_{d1-2} = \frac{1}{\pi} \left( \tan^{-1} \frac{1}{\sqrt{H^2-1}} - \frac{\sqrt{H^2-1}}{H^2} \right)$$

Equation 28



## 6 SPACECRAFT ORIENTATIONS

The orientation of the spacecraft is defined relative to the sun and the planet about which it is orbiting. Two orientations are accommodated in Help Desk (Figure 14):

- solar inertial orientation
- planet orientation

In solar inertial orientation the same side is facing the sun at all times. In planet orientation the same side is facing the planet at all times. The orientation is determined by the mission of the spacecraft. The effects of the orientation are seen in the solar and planet view factors.

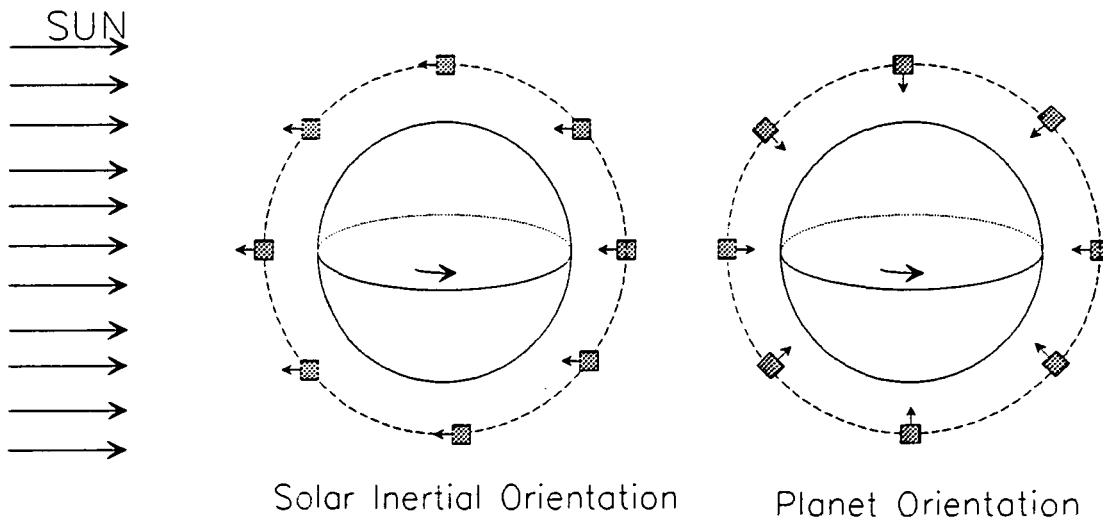


Figure 14 Orientation Geometry

### 6.1 Planet orientation:

Planet orientation has solar view factors that change as the spacecraft travels around the planet. The global sun vector rotates with respect to the local spacecraft coordinates and to the dot product of the sun vector and the unit normals performed at the orbit position.

The planet view factors are always normal or tangential to the planet's surface. To account for the orbits eccentricity, the planet view factors are calculated at every user defined position in the orbit. They are constant only if the orbit is circular.

### 6.2 Inertial orientation:

Solar inertial orientation has constant solar view factors. The global sun vector is rotated to local spacecraft coordinates at the line of nodes ( $\omega = 0$ , and  $v = 0$  in the transformation from local to global see Section 7). In this orbit position the dot product of the sun vector and the unit normals is performed. This value is kept constant at every point in the orbit. The planet view factors change as the spacecraft travels around its orbit. At each location, the normal and tangential components are calculated. The planet view factors are then calculated by rotating the surfaces through the spacecraft position as follows (see Figure 10 for side number referencing):

$$\begin{aligned} FE(1) &= FET \\ FE(2) &= -FEN \cdot \sin \omega + FET \cdot \text{ABS}(\cos \omega) \end{aligned}$$



$FE(3) = FET * ABS(SIN\omega) - FEN * COS\omega$   
 $FE(4) = FEN * SIN\omega + FET * ABS(COS\omega)$   
 $FE(5) = FET * ABS(SIN\omega) + FEN * COS\omega$   
 $FE(6) = FET$

**FET** tangential view factor component  
**FEN** normal view factor component  
 **$\omega$**  spacecraft position in the orbit. The sum of the argument at periapsis and the true anomaly.  
**FE(i)** view factor for the i<sup>th</sup> side

## 7 COORDINATE TRANSFORMATIONS

### 7.1 Global to Local:

This coordinate transformation is never needed. When the sun vector is transformed into local coordinates, the solution is found solving the equation  $Ax = b$ . See Section 5.2 on Solar view factors given Global sun vector and Section 6.2 on inertial orientation, for more information.

### 7.2 Local to Global:

When calculating shadowing effects and spacecraft position, the appropriate vectors must be rotated from local to global coordinates. The transformation used is from Perifocal coordinates to geocentric coordinates. The transformation is taken from reference 2 page 80-83.

$$\begin{bmatrix} r_i \\ r_j \\ r_k \end{bmatrix} = \bar{\bar{R}} \begin{bmatrix} r_p \\ r_q \\ r_w \end{bmatrix} \quad \text{where: } \bar{\bar{R}} = \begin{bmatrix} R11 & R12 & R13 \\ R21 & R22 & R23 \\ R31 & R32 & R33 \end{bmatrix} \quad \text{Equation 29}$$

Where:

$i, j$  and  $k$       the unit vectors in global coordinates  
 $p, q$  and  $w$       the unit vectors in local coordinates.  
 $i$                     inclination  
 $\Omega$                 line of nodes  
 $\omega$                 line of periapsis

$$\begin{aligned}
 R11 &= \cos\Omega \cos\omega - \sin\Omega \sin\omega \cos i \\
 R12 &= -\cos\Omega \sin\omega - \sin\Omega \cos\omega \cos i \\
 R13 &= \sin\Omega \sin i \\
 R21 &= \sin\Omega \cos\omega + \cos\Omega \sin\omega \cos i \\
 R22 &= -\sin\Omega \sin\omega + \cos\Omega \cos\omega \cos i \\
 R23 &= -\cos\Omega \sin i \\
 R31 &= \sin\omega \sin i \\
 R32 &= \cos\omega \sin i \\
 R33 &= \cos i
 \end{aligned}$$

### 7.3 Coordinate Transformations Modifications

To be consistent with the coordinate conventions in IDEAS<sub>TM</sub> Thermal Model Generator (TMG<sub>TM</sub>), there is a slight difference between the local to perifocal and global to geocentric coordinates. The local and perifocal coordinate systems differ by a 180° rotation about W (see Figure 15 and Figure 16).

$$\begin{bmatrix} P_p \\ P_q \\ P_w \end{bmatrix} = \begin{bmatrix} -L_p \\ -L_q \\ L_w \end{bmatrix} \quad \text{also} \quad \begin{bmatrix} G_i \\ G_j \\ G_k \end{bmatrix} = \begin{bmatrix} -g_i \\ -g_j \\ g_k \end{bmatrix} \quad \text{Equation 30}$$

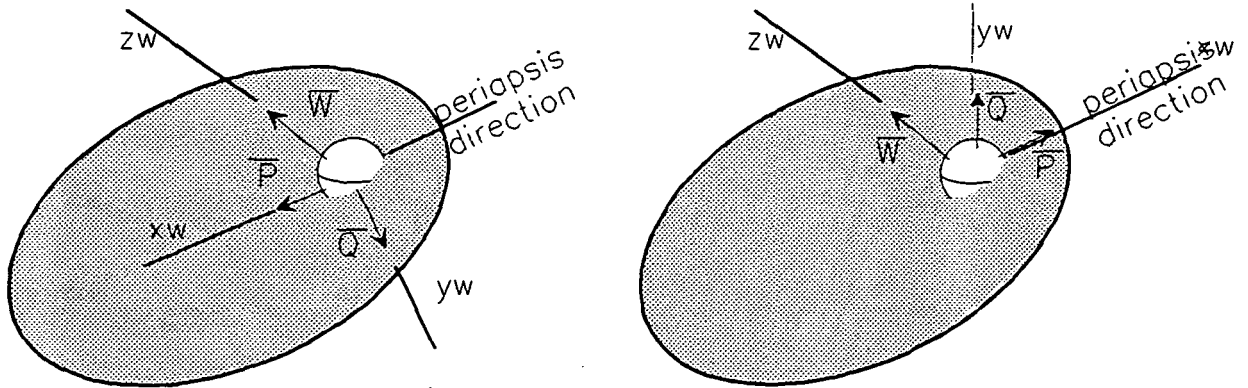
Where:

$P_p$       perifocal coordinates P direction  
 $L_p$       local coordinates P direction.

The global coordinates relate to the geocentric coordinate system in the same way.

$G_I$  the geocentric coordinates in the I direction

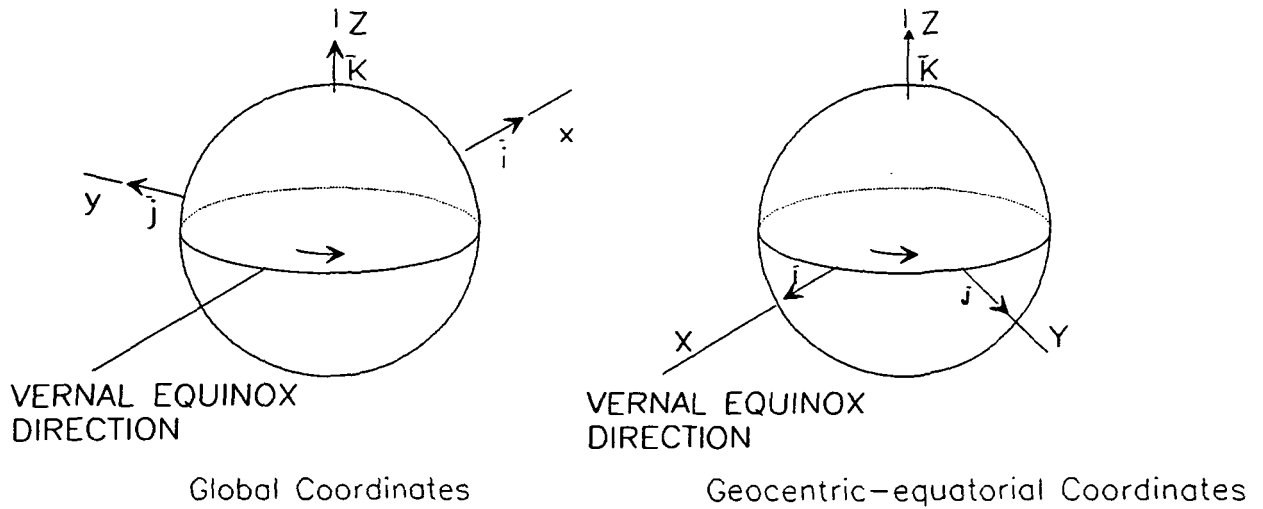
$g_I$  the global coordinates in the I direction



Local Coordinates

Perifocal Coordinates

**Figure 15** Local Coordinate Geometry



VERNAL EQUINOX DIRECTION

VERNAL EQUINOX DIRECTION

Global Coordinates

Geocentric-equatorial Coordinates

**Figure 16** Global Coordinate Geometry

Substituting into the transformation relationship.

$$\begin{bmatrix} -r_i \\ -r_j \\ r_k \end{bmatrix} = \bar{\bar{R}} \begin{bmatrix} -r_p \\ -r_q \\ r_w \end{bmatrix}$$

**Equation 31**

substituting the negatives through the transformation matrix R giving:

$$\begin{bmatrix} r_i \\ r_j \\ r_k \end{bmatrix} = \overline{\overline{R_{\text{mod}}}} \begin{bmatrix} r_p \\ r_q \\ r_w \end{bmatrix} \quad \text{where} \quad \overline{\overline{R_{\text{mod}}}} = \begin{bmatrix} R11 & R12 & -R13 \\ R21 & R22 & -R23 \\ -R31 & -R32 & R33 \end{bmatrix} \quad \text{Equation 32}$$

$R_{\text{mod}}$ :

$$\begin{aligned} R11 &= \cos\Omega \cos\omega - \sin\Omega \sin\omega \cos i \\ R12 &= -\cos\Omega \sin\omega - \sin\Omega \cos\omega \cos i \\ -R13 &= -\sin\Omega \sin i \\ R21 &= \sin\Omega \cos\omega + \cos\Omega \sin\omega \cos i \\ R22 &= -\sin\Omega \sin\omega + \cos\Omega \cos\omega \cos i \\ -R23 &= \cos\Omega \sin i \\ -R31 &= -\sin\omega \sin i \\ -R32 &= -\cos\omega \sin i \\ R33 &= \cos i \end{aligned}$$

Equation 32 states the transformation from the perifocal system coordinate system to the local coordinate system. One more rotation is needed to transform the system into local spacecraft coordinates. The rotation is about the true anomaly at epoch.

LS is the transformation from local spacecraft to perifocal coordinates, ls is the local spacecraft coordinates.

$$\begin{bmatrix} r_p \\ r_q \\ r_w \end{bmatrix} = \overline{\overline{LS}} \begin{bmatrix} ls_p \\ ls_q \\ ls_w \end{bmatrix} \quad \text{where} \quad \overline{\overline{LS}} = \begin{bmatrix} \cos v & -\sin v & 0 \\ \sin v & \cos v & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{Equation 33}$$

## 8 SHADOWING

Shadow calculations are performed using the Orbital Temperature and Heat Loads option in Help Desk. The procedure calculates the spacecraft position, and then determines if the spacecraft falls in the shadow of the planet.

### 8.1 Position Vector

The position vector,  $r$ , points to the location of the spacecraft in the orbit. It is used to determine the spacecraft position relative to the shadow.

Calculate the position vector in perifocal coordinates,  $r$ , and transform into Global coordinates,  $R$ .  $r$  is defined in the perifocal coordinate system as:

$$\bar{R} = r \cos v \bar{P} + r \sin v \bar{Q} \quad \text{Equation 34}$$

$P$  and  $Q$  are the directional unit normals in the perifocal coordinate system.

where the scalar  $r$  (the magnitude of the position vector in perifocal coordinates) can be calculated with the polar equation of a conic section (Equation 34):

- $p$  is the semi-latus rectum
- $e$  is the eccentricity
- $v$  is the true anomaly at epoch
- $a$  is the semi major axis

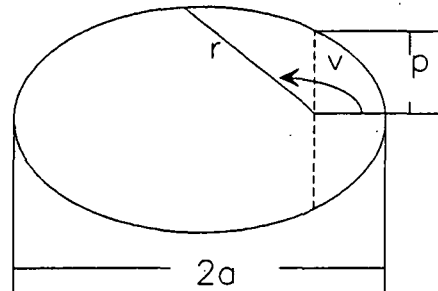


Figure 17 Position Vector Geometry

$$r = \frac{p}{1 + e \cos v} \quad \text{Equation 35}$$

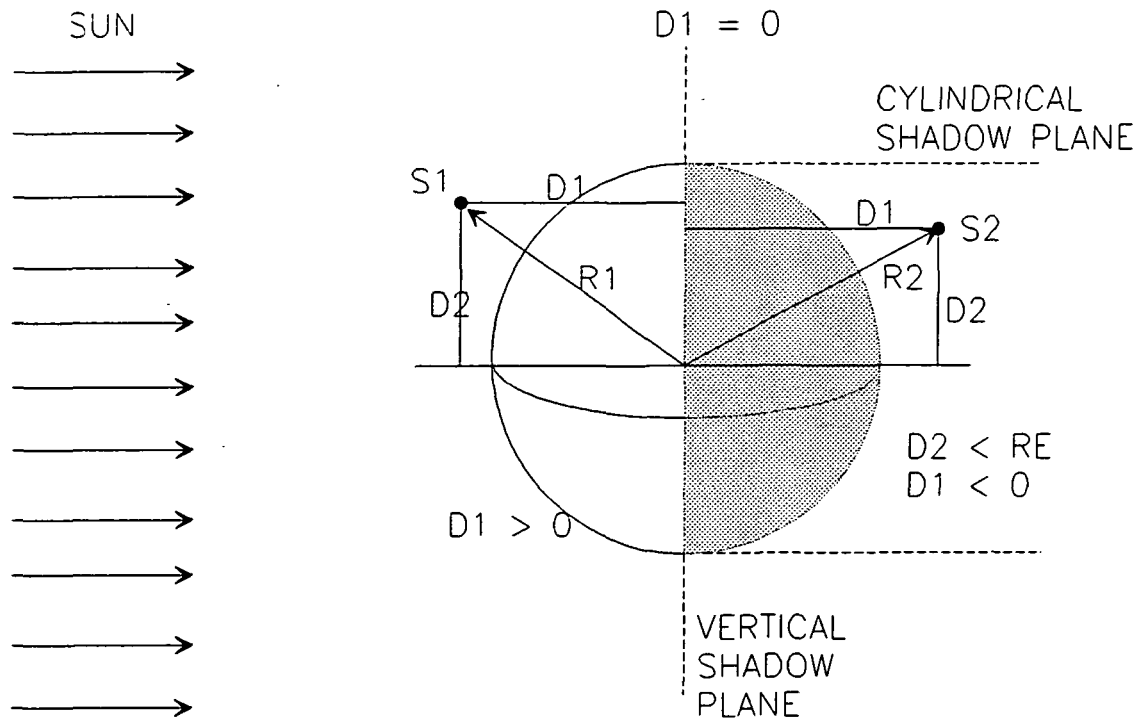
The semi-latus rectum is found using the geometry of the conic section.

$$p = a (1 - e^2)$$

### 8.2 Shadow Calculations

The geometry for shadowing is shown in Figure 18. For a spacecraft to be in the shadow the spacecraft position must satisfy two conditions.

- 1) It must be behind the vertical shadow plane defined by a plane that passes through the center of the planet and is perpendicular to the sun's rays.
- 2) It must be inside a cylindrical shadow defined by a cylinder having the same diameter as the planet and parallel to the sun's rays. The cylindrical shadow extends away from the sun starting at the vertical sun plane where  $D = 0$ .



**Figure 18 Shadowing Geometry**

In Figure 18,  $R1$  is the position of a spacecraft  $S1$ , defined by components  $D1$  and  $D2$ .  $S1$  is on the same side of the vertical plane as the sun placing it outside the shadow. The cylindrical shadow plane effect does not need to be calculated since the cylindrical shadow does not exist on the sun-side of the vertical shadow plane.  $R2$  is the position of a spacecraft  $S2$ , as defined with similar components  $D1$  and  $D2$ .  $D1$  is the distance from the position to the vertical shadow plane and  $D2$  is the distance from the position to the cylindrical plane.

Define in global coordinates:

**R** position vector

**S** solar vector

**O** origin vector (0,0,0 in our case)

**RE** radius of the planet

**V = R-O**

$$D1 = \bar{V} \cdot \bar{S}$$

$$D2 = \sqrt{|\bar{V}|^2 - |D1|^2}$$

$$D2 < RE \wedge D1 < 0 \quad \text{SHADOWED}$$

## **PART II: PROGRAMMER'S GUIDE**

### **1 RADIATE**

An expert system running withing the EXSYS<sup>®</sup> shell has two main files "expert\_system.RUL" and "expert\_system.TXT". In the case of the Thermal Analyst Help Desk they are "RADIATE.RUL" and "RADIATE.TXT". Both files together define the knowledge base and rule base (see the EXSYS<sup>®</sup> Manual for more details). Help Desk also contains the following supplementary files:

- RADIATE.SCR contains all the screen descriptions
- RADIATE.OUT contains the report generator information
- RADIATE.CFG contains the configuration data for the system such as the location of the PASS and RETURN files.
- RADIATE.COM sets up the command environment

The rule base contains 47 Rules. All the rules are listed and explained in Appendix A.

The knowledge base contains 21 qualifiers and 143 variables. All qualifiers are listed in Appendix B.

The screen file, RADIATE.SCR, contains 36 customized screens. All screens and file descriptions are displayed in Appendix C.

The screens use default data files to set the default input values. There are 13 data files. Each file is listed and described in Appendix D.

The report generator file, RADIATE.OUT, has 7 different reports listed. The file uses GOTO statements to point to the file description for each report. The report generator file and a sample file output for each analysis are listed in Appendix E.

There are 47 different FORTRAN routines used by Help Desk. They are contained in 20 ASCII files, and produce 6 executable files. A list of the files, their routines with the calling arguments, and the link commands necessary to produce the executable files are contained in Appendix F.

The FORTRAN source code is listed in Appendix G.





## 2 SCREENS

Most of the screens that appear during an analysis run in Help Desk are custom screens, created with the EXDESIGN software and an ASCII editor. Only two screens are default EXSYS® screens, the description screen, and the HOW? screens. The rest were specifically designed for Help Desk, and may be modified for future applications.

The formats for the screens are in an ASCII file called "RADIATE.SCR". This file is located in the same directory as the expert system and data files. The file contains all the screens, and is organized by qualifier number or variable name. All screens are displayed in Appendix C.

The default values displayed on the screens are stored in ASCII data files that are defined for each screen in the expert system. Any screen value may be changed by the user. Once a value is changed on the screen, it remains at the modified value for the extent of the run. At the start of the next run the values will again be the alues specified in the default data files. To modify the default values for a set of analysis runs, the information in the data files must be changed.

There are two types of data files. The first, "filename.DAT" is used during an analysis run and contains the data used by the expert system. The second, "filename.DT!" contains the original default values. At the end of an analysis sesion, the values may be reset to the original default values by copying the default file "filename.DT!" to the working file "filename.DAT".

The format for the input data files is one variable per line. Each line corresponds to a variable defined in the screen file, in order of appearance. When EXSYS® displays a screen, it places the information from the data file into the available input boxes. All of the applicable data files are shown in Appendix D.

A screen is attached to a qualifier or a variable. When the expert system prompts the user for a qualifier or variable, it searches the screen file for a screen attached to the qualifier or variable. EXSYS® displays the screen defined immediately following the variable name. There are approximately 30 different screens defined in the screen file for this expert system.

The following is a sample screen with the screen description listed and each command explained. While the user creates the screen shown, EXDESIGN creates the file described.

EXSYS Professional	
Planet Constants	
Albedo coefficient:	<input type="text" value=".3"/>
Solar Flux: W/m^2	<input type="text" value="1356"/>
Planet Flux: W/m^2	<input type="text" value="236"/>
Planet radius: km	<input type="text" value="6378"/>
<input type="button" value="different planet"/>	<input type="button" value="OK"/>

Figure 19 Sample Data screen

~Q17	the screen is attached to Qualifier 17 (~Q17)
~SET_WIN 88, 0, 359, 225	the window is located at 88,0 and has a size of 359 wide and 225 tall.
~LINE_COLOR DKGRAY	The line color is dark gray
~FOREGROUND DKGRAY	foreground color is dark gray
~BACKGROUND CYAN	the background is cyan.
~CLS	In order to make the entire screen background one color instead of just the background of the text boxes, the ~CLS command must follow the BACKGROUND command.
~CURSET 102, 28	The first cursor position is at 102, 28
Planet Constants	the text "Planet Constants"
~CURSET 10, 60	the cursor is placed at 10,60
Albedo coefficient:	the text "Albedo coefficient:"
~CURSET 10, 100	the cursor is placed at 10,100
Solar Flux: W/m <sup>2</sup>	the text "Solar Flux: W/m <sup>2</sup> "
~CURSET 10, 140	the cursor is placed at 10,140
Planet Flux: W/m <sup>2</sup>	the text "Planet Flux: W/m <sup>2</sup> "
~EDIT "[ALB]", 225, 45, 103, 35 -M:1	the edit box is attached to the variable [ALB] (albedo). Its is located at 225, 45. Its width is 103 and height 35. The -M:1 option states that the box is only to allow 1 line of entry.
~EDIT "[S]", 225, 85, 103, 35 -M:1	the edit box is attached to the variable [S] (solar flux). Its is located at 225, 85. Its width is 103 and height 35. The -M:1 option states that the box is only to allow 1 line of entry.
~EDIT "[E]", 225, 125, 103, 35 -M:1	the edit box is attached to the variable [E] (planet flux). Its is located at 225, 125. Its width is 103 and height 35. The -M:1 option states that the box is only to allow 1 line of entry.
~EDIT "[RE]", 225, 165, 103, 35 -M:1	the edit box is attached to the variable [RE] (planet radius). Its is located at 225, 165. Its width is 103 and height 35. The -M:1 option states that the box is only to allow 1 line of entry.

~BUTTON "different planet", "Q17 1", 23, 205, 158, 35 -I

the button box is labelled "different planet". If the button is pressed Qualifier 17 (Q17) is assigned the value 1 (a different planet is requested). The box is located at the position 23,205, it is 158 wide and 35 high. The -I option says to return the value immediately after the button is pressed.

~BUTTON "OK", "Q17 2", 229, 205, 75, 35 -I

the button box has the word OK on it. It returns the value 2 to Q17. It is located at 229, 205, has a width of 75 and a height of 35 and returns immediately if pressed.

~EDIT\_FILL ALBSCR.DAT

the ~EDIT\_FILL command specifies the input data file to be accessed to fill in the edit boxes.

~END

defines the end of the screen

## 2.1 Default Data Files

The Planet Constants screen shown in Figure 19 is an example of a data screen using default data values. This data screen is used to enter the variables for albedo coefficient, solar flux, planet flux and planet radius. The following is the associated default data file: ALBSCR.DAT

```
.3
1356
236
6378
```

The value .3 is placed in the edit box for [ALB] (albedo), 1356 is for [S] (solar flux), 236 is for [E] (planet flux), and 6378 for [RE], the planet radius.

### 3 RUNNING EXTERNAL PROGRAMS

Throughout Help Desk, external programs are used to perform complex calculations. They are used to minimize the complexity of the expert system and to increase the capability of Help Desk. When executing Help Desk a user can determine when the expert system is running an external FORTRAN code, because the screen goes blank momentarily. This blanking is due to the Windows-DOS interface.

The executable Fortran files are called using external DOS batch files. The batch files used are:

FETANOR.BAT	Calculates the normal and tangential components for the planet view factors
PLANET.BAT	Calculates the planet view factors for each side of an enclosure
ORBIT.BAT	Performs the orbital analysis
QENCL.BAT	Performs the enclosure analysis
SOLAR.BAT	Calculates the solar view factors
SUNVECT.BAT	Rotates the global sun vector into local coordinates and calculates the solar view factors

A typical DOS batch file, PLANET.BAT, is shown below.

```
TYPE NUL > EXSYS_EP.END
EARTHV.EXE
RENAME EXSYS_EP.END EXSYS_EP.FIN
```

The first line creates a null file called EXSYS\_EP.END.

The second line runs the executable EARTHV.EXE.

The final line renames the file to EXSYS\_EP.FIN.

This batch file procedure is required to run non-windows applications. EXSYS<sup>®</sup> realizes that the analysis is finished when it encounters the file EXSYS\_EP.FIN. As soon as this file exists in the working directory, Help Desk continues its analysis. If the file exists prior to completion of the executable file, Help Desk will take control of the analysis. This may place Help Desk into an infinite loop. The file, EXSYS\_EP.FIN, is created in the method shown above to ensure that the file exists only when the external analysis is complete. If the file is created during the executable, it may be created too early.

The variables needed to execute the code are passed to a file called RADIATE.DAT. The file is defined in the configuration file for the expert system. The path to the file must be defined as the path to the directory containing the batch files and executable files.

An example of a rule call statement to the batch file PLANET.BAT is shown below.

```
IF:
  The spacecraft orientation is {inertial}
and: The spacecraft orbit is {LEO (Low Earth Orbit)}

THEN:
  X> RUN(PLANET.BAT [INC],[ECC],[WOMEGA],[SMA],[VO],[RE] /M /Z)
and: Planet view factors are {calculated}
```

This rule states that if the spacecraft has an inertial orientation, and it is in a low earth orbit, then the system will run the file PLANET.BAT. Six variables are passed to the file

"RADIATE.DAT". Two parameters are also specified, /M and /Z. /M states that multiple variables will be returned in the file "RETURN.DAT" and /Z states that a Ctrl-Z will be added to the end of the file "RADIATE.DAT". Notice that /B option, which the EXSYS<sup>®</sup> manual suggests using, is not used in this case.

NOTE: The /B option is not used for this type of batch file. The /B option indicates that a batch file, containing only DOS commands, is to be run. In this example the batch file includes a FORTRAN subroutine. if the /B option were used in this case, data would not be passed to the RADIATE.DAT file.

#### 4 ADDING A NEW FEATURE

The following procedure was used during the development of Help Desk and is presented here to illustrate the way in which new features can be added.

The only planet data included in the early versions were the default values for Earth. For other planets the user was required to enter the necessary values for the planetary parameters. The expert system has been modified to provide the values for the other planets and the Moon while retaining the option of entering user selected values.

User provided values were entered into the two screens shown in Figure 20. One was used in surface analyses and the other in enclosure analyses. No logic was present to direct the expert system to ask for either a planet selection or the solar and planet flux values. The values were entered only as additional information on screens, which were activated by logic that was searching for emissivity or albedo, respectively.

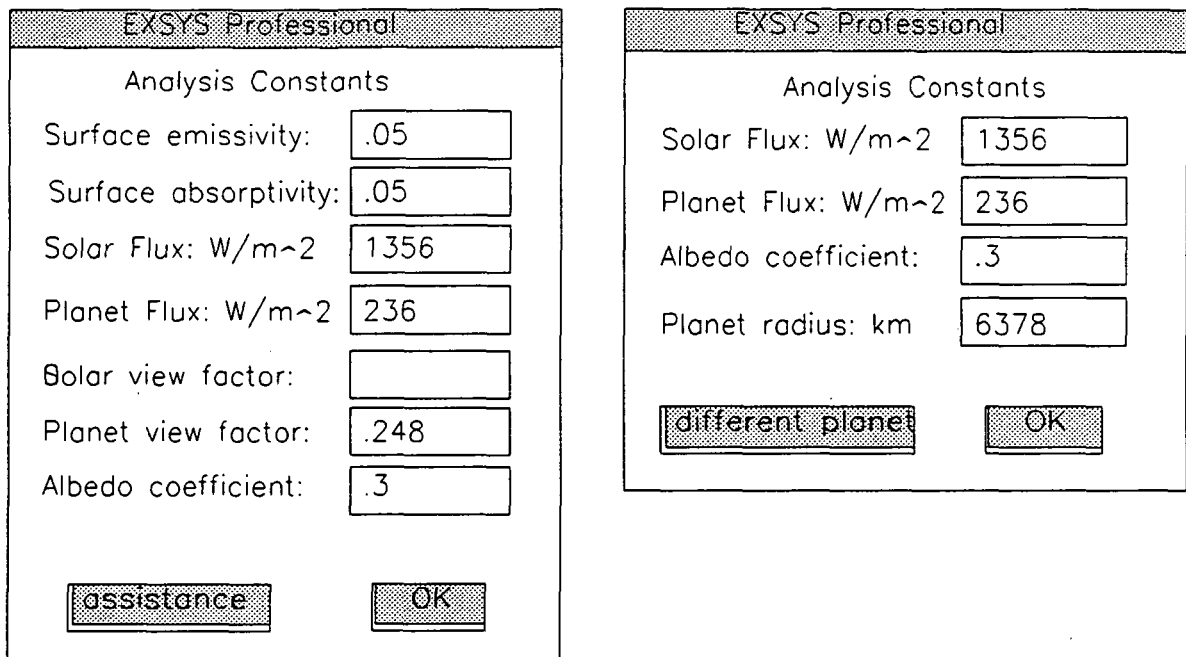


Figure 20 Original Constants screen

To add the other planets it was necessary to split up the information on the screens and add a button called "different planet". This button is similar to the "assistance" button in appearance and function.

The new analysis constants screens called "Surface Constants" and "Planet Constants" are shown in Figure 22. The two screens are now used in both the surface analyses and the enclosure analyses. A new frame containing the values for all the planets was also created. The following sequence describes the process used to modify the expert system.

The first step in adding a new feature is to determine the new qualifiers and variables required for the modification. In this case it requires only one new qualifier, one new rule and a new frame.

The new qualifier is:

Qualifier A different planet is

V requested

V not requested

and the new rule

Rule: If a different planet is requested

Then: [E] IS GIVEN THE VALUE ....

and [S] IS GIVEN THE VALUE...

and [ALB] IS GIVEN THE VALUE...

In this example it is also necessary to define a new frame called "PLANET.TXT", shown below. The frame contains each of the planets, the average values of solar flux and planet flux, and the radius of the planet. Different values may be entered directly at the Planet Constants screen. The frame may also be edited to modify the values for all future analysis runs.

PLANET	PLANET_FLUX	SOLAR_FLUX	ALBEDO	RADIUS
MERCURY	4201	8920	.058	2487
VENUS	154.2	2570	.76	6187
EARTH	236.6	1352	.3	6378
MOON	603.25	1352	.047	1738
MARS	123	577.3	.140	3380
JUPITER	6.1	49.6	.51	71370
SATURN	1.8	14.7	.5	60400
URANUS	.31	3.65	.66	23530
NEPTUNE	.14	1.48	.62	22320
PLUTO	.0	0	.0	2284

The variables used to retrieve the information from the frame must be defined. The format of a frame statement is:

```
FRAME("filename",test_expression,slot_reference)
```

In this example the frame statements are:

```
FRAME("PLANET.TXT",$PLANET$==[PLANET],#SOLAR_FLUX#)  
FRAME("PLANET.TXT",$PLANET$==[PLANET],#PLANET_FLUX#)  
FRAME("PLANET.TXT",$PLANET$==[PLANET],#ALBEDO#)  
FRAME("PLANET.TXT",$PLANET$==[PLANET],#RADIUS#)
```

Using the frame statements, the entire new rule is:

Rule: If a different planet is requested

Then: [S] IS GIVEN THE VALUE

FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#SOLAR\_FLUX#)

and [E] IS GIVEN THE VALUE

FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#PLANET\_FLUX#)

and [ALB] IS GIVEN THE VALUE...

FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#ALBEDO#)

The variable [PLANET] contains the planet name in a string format. To allow the user to choose a planet name, a screen is attached to [PLANET]. When the inference engine processes the frame statements above, it tries to derive the variable [PLANET]. If it cannot derive it, it prompts the user with the screen that is attached to the variable name. The planet screen is shown in Figure 21.

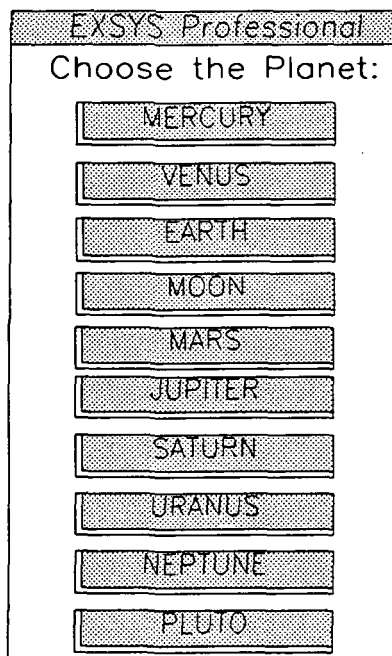


Figure 21 Planet Selection Screen

The new screen definition is shown below. The format is created by the following procedure:

- Define the window size, line color, background and foreground colors.
- Place the cursor at 41, 36 and enter the text "Choose the planet:"
- Place a series of nine buttons each with  
NAME  
VARIABLE and its returned value if the button is pressed  
POSITION OF THE BOX:  
X and Y (component of the upper left hand corner of the box), WIDTH, HEIGHT  
-I return immediately after the button is pressed.
- END



```

~[PLANET]
~SET_WIN 88, 0, 255, 460
~LINE_COLOR DKGRAY
~FOREGROUND DKGRAY
~BACKGROUND CYAN
~CLS
~CURSET 41, 36
Choose the planet:
~BUTTON "MERCURY", "[PLANET] 'MERCURY'", 25, 50, 192, 33 -1
~BUTTON "VENUS", "[PLANET] 'VENUS'", 25, 90, 192, 33 -1
~BUTTON "EARTH", "[PLANET] 'EARTH'", 25, 130, 192, 33 -1
~BUTTON "MOON ", "[PLANET] 'MOON'", 25, 170, 192, 33 -1
~BUTTON "MARS ", "[PLANET] 'MARS'", 25, 210, 192, 33 -1
~BUTTON "JUPITER", "[PLANET] 'JUPITER'", 25, 250, 192, 33 -1
~BUTTON "SATURN", "[PLANET] 'SATURN'", 25, 290, 192, 33 -1
~BUTTON "URANUS", "[PLANET] 'URANUS'", 25, 330, 192, 33 -1
~BUTTON "NEPTUNE", "[PLANET] 'NEPTUNE'", 25, 370, 192, 33 -1
~BUTTON "PLUTO", "[PLANET] 'PLUTO'", 25, 410, 192, 33 -1
~END

```

The original screens were then modified to allow the use of the planet button. The original analysis screens were split into two separate screens with no repetition of user inputs. The first contains all of the surface parameters: emissivity, absorptivity, solar and planet view factors. The second contains all of the parameters related to the planet: albedo coefficient, solar and planet fluxes, and planet radius. Each of the two screen contain a button that allows the user to access information from a frame, as well as a button accepting the user input values.

EXSYS Professional	
<b>Surface Constants</b>	
Solar view factor:	<input type="text" value="0"/>
Planet view factor:	<input type="text" value="248"/>
Surface emissivity:	<input type="text" value=".05"/>
Surface absorptivity:	<input type="text" value=".05"/>
<input type="button" value="assistance"/>	<input type="button" value="OK"/>

EXSYS Professional	
<b>Planet Constants</b>	
Albedo coefficient:	<input type="text" value=".3"/>
Solar Flux: W/m <sup>2</sup>	<input type="text" value="1356"/>
Planet Flux: W/m <sup>2</sup>	<input type="text" value="236"/>
Planet radius: km	<input type="text" value="6378"/>
<input type="button" value="different planet"/>	<input type="button" value="OK"/>

Figure 22 New Analysis Constants screens

Both of the screens are displayed by the inference engine prompting the user for values to qualifiers. The "assistance" and "different planet" buttons are attached to qualifiers. These button values drive the logic flow to the screens.

## SURFACE CONSTANTS SCREEN

Qualifier 16 assistance is:

V needed

V not needed

~Q16

~SET\_WIN 88, 0, 380, 280

~LINE\_COLOR BLACK

~FOREGROUND BLACK

~BACKGROUND CYAN

~CLS

~CURSET 110, 35

Surface Constants

~CURSET 30, 75

Solar view factor:

~CURSET 30, 115

Planet view factor:

~CURSET 30, 155

Emissivity:

~CURSET 30, 195

Absorptivity:

~EDIT "[MU]", 225, 50, 103, 35 -M:1

~EDIT "[FE]", 225, 90, 103, 35 -M:1

~EDIT "[EPSILON]", 225, 130, 103, 35 -M:1

~EDIT "[ALPHA]", 225, 170, 103, 35 -M:1

~BUTTON "OK", "Q16 2", 261, 228, 75, 35 -I

~BUTTON "assistance", "q16 1", 38, 228, 158, 35 -I

~EDIT\_FILL ALPSCR.DAT

~END

## PLANET CONSTANTS SCREEN

Qualifier 17 A different planet is

V requested

V not requested

~Q17

~SET\_WIN 88, 0, 360, 271

~LINE\_COLOR BLACK

~FILL\_COLOR BLACK

~FOREGROUND BLACK

~BACKGROUND CYAN

~CLS

~CURSET 102, 30

Planet Constants

~CURSET 10, 70

Albedo coefficient:

~CURSET 10, 110

Solar Flux: W/m<sup>2</sup>

~CURSET 10, 150

Planet Flux: W/m<sup>2</sup>

~CURSET 10, 190

```
Planet radius: km
~EDIT "[ALB]", 225, 45, 103, 35 -M:1
~EDIT "[S]", 225, 85, 103, 35 -M:1
~EDIT "[E]", 225, 125, 103, 35 -M:1
~EDIT "[RE]", 225, 165, 103, 35 -M:1
~BUTTON "OK", "Q17 2", 241, 217, 75, 35 -I
~BUTTON "different planet", "q17 1", 23, 216, 158, 35 -I
~EDIT_FILL ALBSCR.DAT
~END
```

**Once the above modifications are complete, change the input data files to delete the variable moved.**

**Finally, run the system through every option to make sure that there are no new conflicts.**

## **5 FUTURE DEVELOPMENT**

The Thermal Analyst's Help desk can be expanded to incorporate a number of additional features. Some of the features under consideration are listed here.

- include double sided radiator
- add the planet vector calculations
- add the beta angle calculations
- simplify the parametric trade-off capabilities
- add the planet shield calculations to the Orbital Thermal Analysis
- add the ability to modify the solar view factors for the Enclosure Analysis and allow solar shields to the Orbital Thermal Analysis
- add shadow calculations to the Enclosure analysis
- incorporate the Thermal Electric Cooler expert system into Help Desk
- expand the Surface Properties Analysis

**Appendix A**  
**Rules**

**THIS PAGE INTENTIONALLY LEFT BLANK**

**Purpose:**

Finish the thermal analysis and write the report file. The rule keeps thermal analysis separate from other analyses that may be added, (e.g. Thermal Electric Coolers).

**Premise:**

The qualifier "Thermal analysis is" (Q20) has the value "completed". The value for Q20 is derived using Rules 2-8.

**Conclusion:**

Execute the report generator file "RADIATE.OUT", see the report generator section of this document for further explanation.

Define the Choice "Thermal analysis is complete" to have a confidence value of 1.

**/\* RULE NUMBER: 1**

**RULE: THERMAL COMPLETE**

**IF:**

Thermal analysis is (completed)

**THEN:**

X> REPORT(RADIATE.OUT)

and: > Thermal analysis is complete - Confidence=1

PROCEEDING PAGE BLANK NOT FILMED

**Purpose:**

Define the area of a surface. When the inference engine is proving a premise, it finds the first rule, in sequential order, that can define the variable or qualifier under investigation. In the example, the inference engine is trying to define the qualifier "Thermal analysis is", as show in the premise of Rule 1.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "radiator area (given temperature)". The user is prompted with the screen containing the main menu. When the user selects an analysis, the value is returned and assigned to Q1.

The variable [AREA], the area of the surface, must be within .1% of the equation shown. The value for the variable [AREA] is derived using Rule 9. The terms on the right hand side of the equation are explicitly stated to require each term in the equation to be evaluated regardless of the initialized values. See Rule 9 for more information on the derivations. The characters ~= stand for "approximately equals", which is defined to be  $\pm 0.1\%$  of the value.

The variables [QRAD] the heat load radiated from the surface, [QSUN] the heat load on the surface due to the sun, [QALBEDO] the heat load on the surface due to the planet's albedo, [QPLANET] the heat load on the surface due to the planet, must sum to a value greater than zero. Each of the variables are initialized to zero, thus the sum equals zero. If a value cannot be derived, the expert system will use the initialized value prior to prompting the user for the information. Since the values for each variable is displayed on the final report page, the expert system needs a value for them. If they are not initialized to zero, the values would be derived, regardless of the user selection on the radiation effects screen. Thus, the expert system is required to derive all the terms from existing rules before accepting the initialized value.

**Conclusion:**

Display the report page [REPORT1]. See Appendix C for the figure of [REPORT1].

The qualifier "Thermal analysis is" is assigned the value "completed".

/\* RULE NUMBER: 2

RULE: AREA FOUND

IF:

Value to calculate is (radiator area (given temperature))

and: [AREA] ~=

-([QPOWER]+[QENCL])/([XSUN]+[XPLANET]+[XALBEDO]-[XRAD])

and: [QRAD] + [QPLANET] + [QSUN] + [QALBEDO] > 0

THEN:

[REPORT] IS GIVEN THE VALUE [REPORT1]

and: Thermal analysis is (completed)



**Purpose:**

Defines an area has been calculated less than zero. The rule allows the completion of the analysis if Rule 2 defines a negative area.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "radiator area (given temperature)". The user is prompted with the screen containing the main menu. When the user selects an analysis, the value is returned and assigned to Q1.

The variable [AREA], the area of the surface, has been calculated to have a value less than zero.

**Conclusion:**

The user is informed of a bad area using [REPORT2]. See Appendix C for the figure of [REPORT2].

The qualifier "Thermal analysis is" is assigned the value "completed".

**/\* RULE NUMBER: 3**

**RULE: BAD AREA**

**IF:**

Value to calculate is {radiator area (given temperature)}

and: [AREA] < 0

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT2]

and: Thermal analysis is {completed}

**Purpose:**

Define the temperature of a surface.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "surface temperature", which is derived from the main menu with the analysis chosen by the user.

The qualifier "Surface temperature to be calculated is" (Q12) has the value "radiator temperature given area". The user is prompted with a screen containing two choices, radiator temperature and environmental temperature. Once the user selects a value, it is assigned to Q12.

The variable [TEMP], the temperature of the surface, must be calculated and greater than zero. The value for [TEMP] is derived using Rule 10.

The variables [QRAD] the heat load radiated from the surface, [QSUN] the heat load on the surface due to the sun, [QALBEDO] the heat load on the surface due to the planet's albedo, [QPLANET] the heat load on the surface due to the planet, must sum to a value greater than zero. Each of the variables are initialized to zero, thus the sum equals zero. If a value cannot be derived, the expert system will use the initialized value prior to prompting the user for the information. Since the values for each variable is displayed on the final report page, the expert system needs a value for them. If they are not initialized to zero, the values would be derived, regardless of the user selection on the radiation effects screen. Thus, the expert system is required to derive all the terms from existing rules before accepting the initialized value.

**Conclusion:**

Display the report page containing the final information [REPORT1]. See Appendix C for the figure of [REPORT1].

The qualifier "Thermal analysis is" is assigned the value "completed".

/\* RULE NUMBER: 4

RULE: SUR TEMP FOU

**IF:**

Value to calculate is {surface temperature}

and: Surface temperature to be calculated is {radiator temperature given area}

and: [TEMP] > 0

and: [QRAD] + [QSUN] + [QPLANET] + [QALBEDO] + [QENCL] > 0

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT1]

and: Thermal analysis is {completed}

**Purpose:**

Define the environmental temperature of a surface.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "surface temperature", which is derived from the main menu with the analysis chosen by the user.

The qualifier "Surface temperature to be calculated is" (Q12) has the value "environmental temperature". The user is prompted with a screen containing two choices, radiator temperature and environmental temperature. Once the user selects a value, it is assigned to Q12.

The variable [TEMP], the temperature of the surface, must be calculated and greater than zero. The value for [TEMP] is derived using Rule 10.

**Conclusion:**

Display the report page containing the final information [REPORT3]. See Appendix C for the figure of [REPORT3].

The qualifier "Thermal analysis is" is assigned the value "completed".

**Purpose:**

Define the enclosure surface temperatures and heat loads.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "enclosure temperatures and heat loads", which is derived from the main menu with the analysis chosen by the user.

The qualifier "The temperature values are" (Q7) has the value "calculated". Help Desk derives Q7 using Rule 24.

**Conclusion:**

Display the report page containing the final information [REPORT4]. See Appendix C for the figure of [REPORT4].

The qualifier "Thermal analysis is" is assigned the value "completed".

**/\* RULE NUMBER: 5**  
**RULE: ENV TEMP FOU**

**IF:**

Value to calculate is {surface temperature}  
and: Surface temperature to be calculated is {environmental temperature}  
and: [TEMP] > 0

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT3]  
and: Thermal analysis is {completed}

**/\* RULE NUMBER: 6**  
**RULE: ENCL FOUND**

**IF:**

Value to calculate is {enclosure temperatures and heat loads}  
and: The temperature values are {calculated}

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT4]  
and: Thermal analysis is {completed}

**Purpose:**

Define the enclosure surface temperature and heat loads over an entire orbit.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "orbital average temperatures and heat loads", which is derived from the main menu with the analysis chosen by the user.

The qualifier "Orbital analysis is" (Q11) has the value "finished". Help Desk derives the value for Q11 using Rules 25-26.

**Conclusion:**

Display the report page containing the final information [REPORT5]. See Appendix C for the figure of [REPORT5].

The qualifier "Thermal analysis is" is assigned the value "completed".

**Purpose:**

Calculate the surface emissivity and absorptivity.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "surface properties", which is derived from the main menu with the analysis chosen by the user.

The qualifier "Surface properties are" (Q13) has the value "calculated". Help Desk derives the value for Q13 using Rules 38-41 or Rules 43-46.

**Conclusion:**

Display the report page containing the final information [REPORT6]. See Appendix C for the figure of [REPORT6].

The qualifier "Thermal analysis is" is assigned the value "completed".

**/\* RULE NUMBER: 7**  
**RULE: ORBIT FOUND**

**IF:**

Value to calculate is (orbital average temperatures and heat loads)  
and: Orbital analysis is (finished)

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT5]  
and: Thermal analysis is (completed)

**/\* RULE NUMBER: 8**  
**RULE: PROP FOUND**

**IF:**

Value to calculate is (surface properties)  
and: Surface properties are (finished)

**THEN:**

[REPORT] IS GIVEN THE VALUE [REPORT6]  
and: Thermal analysis is (completed)

**Purpose:**

Calculate the area of the surface (see Rule 2).

$$A = -\frac{(Q_{POWER} + Q_{ENCLOSURE})}{X_{SUN} + X_{PLANET} + X_{ALBEDO} - X_{RAD}}$$

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "radiator area (given temperature)", which is derived from the main menu with the analysis chosen by the user.

The sum of the variables [XSUN] the heat flux on the surface due to the sun's radiation, [XPLANET] the heat flux on the surface due to the planet's radiation, [XALBEDO] the heat flux on the surface due to the albedo of the planet and -[XRAD] the heat flux radiated from the surface is greater than zero. This requires all of the heat flux terms to be evaluated and that the sum is not zero, preventing division by zero in the THEN clause. [XSUN] is derived using Rule 19. [XPLANET] is derived by Rule 20. [XALBEDO] is derived by Rule 21. [XRAD] is derived using Rule 23.

**Conclusion:**

The variable [AREA] is defined. [QPOWER], the internal power to be dissipated, is input by the user. [QENCL] the heat load from the enclosure is derived using Rule 17. The remaining terms are derived in the premise.

**/\* RULE NUMBER: 9**

**RULE: AREA**

**IF:**

Value to calculate is {radiator area (given temperature)}

and: [XSUN] + [XPLANET] + [XALBEDO] - [XRAD] <> 0

**THEN:**

[AREA] IS GIVEN THE VALUE

-([QPOWER]+[QENCL])/([XSUN]+[XPLANET]+[XALBEDO]-[XRAD])

**Purpose:**

Calculate the temperature of the surface (see Rule 4 and Rule 5)

$$E_b = \sigma T^4 \quad \Rightarrow \quad T = \left( \frac{E_b}{\sigma} \right)^{\frac{1}{4}}$$

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "surface temperature", which is derived from the main menu with the analysis chosen by the user.

The variable [EB], the blackbody radiation on the surface, must be greater than zero. It is derived using Rule 11.

The variable [SIGMA], the Stephen-Boltzmann constant must be greater than zero. It is derived using Rule 22. EXSYS® does not use scientific notation and cannot initialize a variable as small as [SIGMA]. However, it can be assigned a small value in a rule.

**Conclusion:**

The variable [TEMP] is assigned the value as shown.

/\* RULE NUMBER: 10

RULE: TEMP

IF:

Value to calculate is (surface temperature)

and: [EB] > 0

and: [SIGMA] > 0.0

THEN:

[TEMP] IS GIVEN THE VALUE SQRT(SQRT((EB)/[SIGMA]))

**Purpose:**

Calculate the blackbody radiation on the surface (see Rule 10).

$$E_b = \frac{1}{\epsilon} \left( X_{SUN} + X_{PLANET} + X_{ALBEDO} - \frac{(Q_{POWER} + Q_{ENCLOSURE})}{A} \right)$$

**Premise:**

The qualifier "Surface temperature to be calculate is" (Q12) has the value "radiator temperature given area". The user is prompted with a screen containing two choices: radiator temperature and environmental temperature. The value selected is assigned to Q12 (see Rule 4 and Rule 5).

The variable [AREA] is greater than zero. The inference engine cannot derive a value for the area, so it prompts the user for a value with a screen that attached to the variable [AREA].

The variable [EPSILON], the emissivity of the surface, is greater than zero, which is derived using Rules 43-46. Even though Rules 38-41 can also derive values for [EPSILON], the rules were marked False as soon as Q1 was assigned a value from the main menu, so the inference engine will never try to "fire" them. The user will be allowed to enter the known value or ask Help Desk for "assistance".

**Conclusion:**

The variable [EB] is defined. [XSUN] the heat flux on the surface due to the sun is calculated using Rule 19. [XPLANET] the heat flux on the surface from the radiation of the planet is derived by Rule 20. [XALBEDO] the heat flux on the surface from the albedo of the planet is derived by Rule 21. [QPOWER] the internal power to be dissipated is input by the user. [QENCL] the heat load from the enclosure is derived using Rule 17. The remaining terms are derived in the premise.

/\* RULE NUMBER: 11

RULE: BLACKBODY

IF:

Surface temperature to be calculated is [radiator temperature given area]

and: [AREA] > 0

and: [EPSILON] > 0

THEN:

[EB] IS GIVEN THE VALUE

(([XSUN]+[XPLANET]+[XALBEDO])+((([QPOWER]+[QENCL])  
/[AREA]))/[EPSILON])

**Purpose:**

Calculate the blackbody radiation on the surface (see Rule 10).

$$E_b = \frac{1}{\epsilon} (X_{SUN} + X_{PLANET} + X_{ALBEDO})$$

**Premise:**

The qualifier "Surface temperature to be calculate is" (Q12) has the value "environmental temperature". The user is prompted with a screen containing two choices, radiator temperature and environmental temperature. The value selected is assigned to Q12 (see Rule 4 and Rule 5).

The variable [EPSILON], the emissivity of the surface, is greater than zero, which is derived using Rules 43-46. Even though Rules 38-41 can also derive values for [EPSILON], the rules were marked False as soon as Q1 was assigned a value from the main menu, so the inference engine will never try to "fire" them. The user will be allowed to enter the known value or ask Help Desk for "assistance".

**Conclusion:**

The variable [EB] is defined. [XSUN] the heat flux on the surface due to the sun is calculated using Rule 19. [XPLANET] the heat flux on the surface from the radiation of the planet is derived by Rule 20. [XALBEDO] the heat flux on the surface from the albedo of the planet is derived by Rule 21. [EPSILON] is derived in the premise.

**Purpose:**

Calculate the heat load on the surface due to the sun's radiation.

$$Q_{SUN} = X_{SUN} A$$

**Premise:**

The qualifier "radiation effects to include" (Q2) has the value "sun". The user is prompted with a screen containing all the possible radiation effects on the surface. The user selects one or multiple values, which are assigned to Q12.

**Conclusion:**

The variable [QSUN] is assigned a value as shown. [AREA] is derived using Rule 9. [XSUN] is derived using Rule 19.

/\* RULE NUMBER: 12  
RULE: ENVIRON TEMP

**IF:**

Surface temperature to be calculated is (environmental temperature) and: [EPSILON]>0

**THEN:**

[EB] IS GIVEN THE VALUE  
([XSUN]+[XPLANET]+[XALBEDO])/[EPSILON]

/\* RULE NUMBER: 13  
RULE: SUN HEAT LOAD

**IF:**

Radiation effects to include (sun)

**THEN:**

[QSUN] IS GIVEN THE VALUE [XSUN]\*[AREA]



**Purpose:**

Calculate the heat load on the surface due to the radiation of the planet.

$$Q_{PLANET} = X_{PLANET}A$$

**Premise:**

The qualifier "radiation effects to include" (Q2) has the value "planet". The user is prompted with a screen containing all the possible radiation effects on the surface. The user selects one or multiple values, which are assigned to Q12.

**Conclusion:**

The variable [QPLANET] is assigned a value as shown. [AREA] is derived using Rule 9. [XPLANET] is derived using Rule 20.

**Purpose:**

Calculate the heat load on the surface due to the albedo of the planet.

$$Q_{ALBEDO} = X_{ALBEDO}A$$

**Premise:**

The qualifier "radiation effects to include" (Q2) has the value "albedo". The user is prompted with a screen containing all the possible radiation effects on the surface. The user selects one or multiple values, which are assigned to Q12.

**Conclusion:**

The variable [QALBEDO] is assigned a value as shown. [AREA] is derived using Rule 9. [XALBEDO] is derived using Rule 21.

/\* RULE NUMBER: 14

RULE: PLANET HEAT LOAD

IF:

Radiation effects to include {planet}

THEN:

[QPLANET] IS GIVEN THE VALUE [XPLANET]\*[AREA]

/\* RULE NUMBER: 15

RULE: ALBEDO HEAT LOAD

IF:

Radiation effects to include {albedo}

THEN:

[QALBEDO] IS GIVEN THE VALUE [XALBEDO]\*[AREA]

**Purpose:**

Calculate the heat load radiated from the surface.

$$Q_{RAD} = X_{RAD}A$$

**Premise:**

The value of [XRAD] must be greater than zero. This forces the evaluation of the variable [XRAD]. This state the radiative capability of the surface. The value is calculated regardless of which radiation effects have been included.

**Conclusion:**

The variable [QRAD] is assigned a value as shown. [AREA], is derived using Rule 9. [XRAD] is derived using Rule 23.

**Purpose:**

Calculate the heat load to be radiated from the surface, due to an enclosure.

$$Q_{ENCL} = A_{INST}e_{INST}E_{B_{enc}}$$

**Premise:**

The qualifier "radiation effects to include" (Q2) has the value "enclosure radiation". The user is prompted with a screen containing all the possible radiation effects on the surface. The user selects one or multiple values, which are assigned to the qualifier.

The qualifier "Enclosure radiation effects are to be" (Q21) has the value "calculated" The user is prompted with a screen with a place to enter a numeric value and two buttons. The screen satisfies the rule regardless of the input. Either the user will input a value for QENCL or will return a value of "calculated" which is assigned to Q21.

**Conclusion:**

The variable [QENCL] is assigned a value as shown. [AINST], area of instrument surface, and [EPSINST], the emmissivity of the surface are input by the user with the screen [AINST]. [EBENC] is derived with assistance from Rule 18.

/\* RULE NUMBER: 16  
RULE: RAD HEAT LOAD

IF:  
[XRAD] >0

THEN:  
[QRAD] IS GIVEN THE VALUE [XRAD]\*[AREA]

/\* RULE NUMBER: 17  
RULE: ENCL HEAT LOAD

IF:  
Radiation effects to include (enclosure radiation)  
and: Enclosure radiation effects are to be (calculated)

THEN:  
[QENCL] IS GIVEN THE VALUE [AINST]\*[EPSINST]\*[EBENC]

**Purpose:**

Calculate the temperature and blackbody radiation of an enclosure.

$$T_{ENCL}^A = \frac{(T_1^A + T_2^A + T_3^A + T_4^A + T_5^A + T_6^A)}{6}$$

**AND:**

$$E_{B_{ENCL}} = \sigma T_{ENCL}^A$$

**Premise:**

The qualifier "The temperature values are" (Q7) has the value "calculated". The qualifier is derived using Rule 24.

**Conclusion:**

[TENCL], the average calculated temperature of the sides of an enclosure, is assigned a value as shown. The variables [T1], [T2], [T3], [T4], [T5], and [T6] are the temperatures of the respective surfaces. The values are calculated using Rule 24.

[EBENC], the average blackbody radiation for the enclosure, is assigned a value as shown. [SIGMA] is defined in Rule 22.

**/\* RULE NUMBER: 18**

**RULE: ENCLOSURE TEMP**

**IF:**

The temperature values are {calculated}

**THEN:**

[TENCL] IS GIVEN THE VALUE

(([T1]^4+[T2]^4+[T3]^4+[T4]^4+[T5]^4+[T6]^4)/6

and: [EBENC] IS GIVEN THE VALUE [SIGMA]\*[TENCL]

**Purpose:**

Calculate the heat flux on the surface due to the effects of the sun.

$$X_{SUN} = \alpha S \mu$$

**Premise:**

The qualifier "Radiation effects to include" (Q2) must include the value "sun". The user is prompted with the screen attached to Q2. The user is allowed to select one or all of the possible radiation effects listed. Q2 is the only multiple value qualifier in the expert system.

**Conclusion:**

The variable [XSUN] is assigned the value as shown. The inference engine derives the value for [ALPHA], the surface absorptivity, using Rules 43-46. Even though Rules 38-41 can also derive values for [ALPHA], the rules were marked False as soon as Q1 was assigned a value from the main menu, so the inference engine will never try to "fire" them. Help Desk displays the screen attached to Q16 "assistance is". It displays variables including [ALPHA] and [MU]. The user can input a value or ask Help Desk for "assistance". Thus [MU] the solar view factor is defined as additional information from the screen.

The inference engine derives the variable [S] the planet solar flux with Rule 47.

**/\* RULE NUMBER: 19**

**RULE: SUN FLUX**

**IF:**

Radiation effects to include {sun}

**THEN:**

[XSUN] IS GIVEN THE VALUE [ALPHA]\*[S]\*[MU]

**Purpose:**

Calculate the heat flux on the surface due to the effects of the planet's radiation.

$$X_{PLANET} = \epsilon F_E E$$

**Premise:**

The qualifier "Radiation effects to include" (Q2) must include the value "planet". The user is prompted with the screen attached to Q2. The user is allowed to select one or all of the possible radiation effects listed. Q2 is the only multiple value qualifier in the expert system.

**Conclusion:**

The variable [XPLANET] is assigned the value as shown. The inference engine derives the value for [EPSILON] the surface emissivity using Rule 43-46. Even though Rules 38-41 can also derive values for [ALPHA], the rules were marked False as soon as Q1 was assigned a value from the main menu, so the inference engine will never try to "fire" them. Help Desk displays the screen attached to Q16 "assistance is". It displays variables including [EPSILON] and [FE]. The user can input a value or ask Help Desk for "assistance". Thus [FE] is defined as additional information from the screen.

The inference engine derives the variable [E] the planet flux with Rule 47.

**/\* RULE NUMBER: 20**

**RULE: PLANET FLUX**

**IF:**

Radiation effects to include {planet}

**THEN:**

[XPLANET] IS GIVEN THE VALUE [EPSILON]\*[FE]\*[E]

**Purpose:**

Calculate the heat flux on the surface due to the planet's albedo.

$$X_{ALBEDO} = \alpha a F_E S (1 - \mu)$$

**Premise:**

The qualifier "Radiation effects to include" (Q2) must include the value "albedo". The user is prompted with the screen attached to Q2. The user is allowed to select one or all of the possible radiation effects listed. Q2 is the only multiple value qualifier in the expert system.

**Conclusion:**

The variable [XALBEDO] is assigned a value as shown. The inference engine derives the value for [ALPHA], the surface absorptivity, using Rules 43-46. Even though Rules 38-41 can also derive values for [ALPHA], the rules were marked False as soon as Q1 was assigned a value from the main menu, so the inference engine will never try to "fire" them. Help Desk displays the screen attached to Q16 "assistance is". It displays variables including [ALPHA], [MU] and [FE]. The user can input values or ask Help Desk for "assistance". Thus [MU] and [FE] are as additional information from the screen.

The inference engine derives the variable [S], the planet solar flux, and [ALB], the planet albedo, with Rule 47.

**Purpose:**

Define the value for the Stephen Boltzmann constant. EXSYS® does not have the ability to use scientific notation. Therefore, it has a problem with values such as 5.67E-8. The problem includes the inability to initialize the variable [SIGMA]. The value must therefore be derived by a rule, as it is done here.

/\* RULE NUMBER: 21

RULE: ALBEDO FLUX

IF:

Radiation effects to include {albedo}

THEN:

[XALBEDO] IS GIVEN THE VALUE  
[ALPHA]\*[ALB]\*[FE]\*[S]\*(1-[MU])

/\* RULE NUMBER: 22

RULE: SIGMA

IF:

Value to calculate is (radiator area (given temperature))  
or: Value to calculate is (surface temperature)

THEN:

[SIGMA] IS GIVEN THE VALUE .0000000567

**Purpose:**

Calculate the radiative flux leaving the surface.

$$X_{RAD} = \epsilon \sigma T^4$$

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "radiator area (given temperature)" or "surface temperature". The user is prompted with the screen containing the main menu. When the user selects an analysis, the value is returned and assigned to Q1.

**Conclusion:**

The variable [XRAD] is assigned a value as shown. The inference engine derives the value for [EPSILON] the surface emissivity using Rule 43-46. [SIGMA], the Stephen-Boltzmann constant, is derived using Rule 22 and [TEMP], the temperature of the surface, is input by the user using the screen attached to the variable [TEMP].

**/\* RULE NUMBER: 23**

**RULE: RAD FLUX**

**IF:**

Value to calculate is (radiator area (given temperature))

or: Value to calculate is (surface temperature)

**THEN:**

[XRAD] IS GIVEN THE VALUE [EPSILON]\*[SIGMA]\*[TEMP]^4

**Purpose:**

Calculate the blackbody radiation generated by the enclosure.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "enclosure temperatures and heat loads". The value is assigned with the main menu.

or:

The qualifier "Radiation effects to include" (Q2) has the value "enclosure radiation". The value is assigned with a screen containing all the possible radiation effects.

The qualifier "Solar view factors are" (Q6) has the value "calculated", which is derived using Rules 28-31. The variable definitions of [MU1], [MU2], [MU3], [MU4], [MU5] and [MU6] are defined when deriving the qualifier value.

The qualifier "Planet view factors calculations are" (Q15) has the value "finished". This is derived using Rules 32-33. The variable definitions of [FE1], [FE2], [FE3], [FE4], [FE5] and [FE6] are defined when deriving the qualifier value.

**Conclusion:**

QENCL.BAT is an external DOS batch file, which passes the information shown, and returns the six temperatures ([T1] - [T6]) and heat loads ([Q1] - [Q6]).

The variables are taken from the expert system's known information. If any variable is not previously defined, the expert system will try to derive the value using rules in the rule base. If the values cannot be defined, the expert system will prompt the user. The expert system first looks for a screen attached to each of the variables. If a screen exists it is shown to the user, if not, the expert system displays its own screen using the defined prompt for the variable.

In this way all the variables are defined. Instead of defining a screen for each variable, variables are grouped together. For example, [INS1] to [INS5] are all defined on the same screen. The screen will always be called by the first variable [INS1]. The knowledge engineer must make sure that every time any of the other variables are called that [INS1] is always listed first.

/\* RULE NUMBER: 24

RULE: ENCLOSURE

IF:

Value to calculate is {enclosure temperatures and heat loads}

or: Enclosure radiation effects are to be {calculated}

and: Solar view factors are {calculated}

and: Planet view factors calculations are {finished}

THEN:

X> RUN(QENCL.BAT [ALB],[S],[E],[X],[Y],[Z],

[FE1],[FE2],[FE3],[FE4],[FE5],[FE6],

[ALPIR1],[ALPIR2],[ALPIR3],[ALPIR4],[ALPIR5],[ALPIR6],

[ALPS1],[ALPS2],[ALPS3],[ALPS4],[ALPS5],[ALPS6],

[MU1],[MU2],[MU3],[MU4],[MU5],[MU6],

[NUMINS],[INS1],[INS2],[INS3],[INS4],[INS5] /M /Z)

and: The temperature values are {calculated}



**Purpose:**

Calculate the orbital temperature and heat loads on an enclosure. Rule 25 is for a planet oriented spacecraft, and Rule 26 is for a solar inertial oriented spacecraft. Otherwise they are identical.

The analysis is similar to the enclosure analysis shown in Rule 24. Exceptions include the two qualifiers that must be defined in Rule 24 and not here. The solar and planet view factors are calculated at every point in the orbit in the Fortran Code, so Help Desk is not required to calculate them first.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "orbital average temperatures and heat loads". The value is assigned with the main menu.

The qualifier "spacecraft orientation is" (Q4) has the value "planet" for Rule 25 and "inertial" for Rule 26. The user is shown a screen containing two buttons. Once the user selects one, it is assigned to Q4.

**Conclusion:**

ORBIT.BAT is an external DOS batch file, which passes the information shown, and returns the orbital average information shown on the screen [REPORT5].

The variables are taken from the expert system's known information. If any variable is not previously defined, the expert system will try to derive the value using rules in the rule base. If the values cannot be defined, the expert system will prompt the user. The expert system first looks for a screen attached to each of the variables. If a screen exists it is shown to the user. If not, the expert system displays its own screen using the defined prompt for the variable.

In this way all the variables are defined. Instead of defining a screen for each variable, variables are grouped together. For example [INS1] to [INS5] are all defined on the same screen. The screen will always be called by the first variable [INS1]. The designer must make sure that every time any of the other variables are called that [INS1] is always listed first.

**/\* RULE NUMBER: 25**

**RULE: ORBITAL PLANET**

**IF:**

Value to calculate is (orbital average temperatures and heat loads)  
and: The spacecraft orientation is (planet)

**THEN:**

```
X> RUN(ORBIT.bat [ALB],[S],[E], [GS1],[GS2],[GS3],
[INC],[OMEGA],[WOMEGA],[ECC],[SMA],0,
[X],[Y],[Z],[RE],
[ALPIR1],[ALPIR2],[ALPIR3],[ALPIR4],[ALPIR5],[ALPIR6],
[ALPS1],[ALPS2],[ALPS3],[ALPS4],[ALPS5],[ALPS6],
[NUMINS], [INS1], [INS2], [INS3], [INS4],[INS5], [NUMPTS] /M
```

**/Z)**

and: Orbital analysis is (finished)

**/\* RULE NUMBER: 26**

**RULE: ORBITAL INERTIAL**

**IF:**

Value to calculate is (orbital average temperatures and heat loads)  
and: The spacecraft orientation is (inertial)

**THEN:**

```
X> RUN(ORBIT.bat [ALB], [S],[E], [GS1],[GS2],[GS3],
[INC],[OMEGA],[WOMEGA],[ECC],[SMA],1,
[X],[Y],[Z], [RE],
[ALPIR1],[ALPIR2],[ALPIR3],[ALPIR4],[ALPIR5],[ALPIR6],
[ALPS1], [ALPS2],[ALPS3],[ALPS4],[ALPS5],[ALPS6],
[NUMINS], [INS1],[INS2], [INS3], [INS4],[INS5], [NUMPTS] /M /Z)
```

and: Orbital analysis is (finished)

**Purpose:**

Restrict the mode of entering the solar vector to global coordinates when performing orbital analysis.

**Premise:**

The qualifier "Value to calculate is" (Q1) has the value "orbital average temperatures and heat loads".

**Conclusion:**

The qualifier "Orbit information will be" (Q3) is assigned the value "user specified global sun and Planet vector with orbit information (orbital elements)". See Rules 28-31.

**Purpose:**

Calculate the solar view factors for an enclosure.

**Premise:**

The qualifier "Orbit information will be" (Q3) has the value "user specified local sun and Planet vector". The sun vector is entered in local coordinates.

**Conclusion:**

Run the external DOS file "SOLAR.BAT", which passes the local sun vector [XS1], [XS2], and [XS3]. The six solar view factors are returned, [MU1], [MU2], [MU3], [MU4], [MU5] and [MU6].

The value "calculated" is assigned to the qualifier "Solar view factors are".

**/\* RULE NUMBER: 27**

**RULE: ORBIT-GLOBAL**

**IF:**

Value to calculate is (orbital average temperatures and heat loads)

**THEN:**

Orbit information will be (user specified global sun and Planet vector with orbit information (orbital elements))

**/\* RULE NUMBER: 28**

**RULE: SOLAR VECTOR**

**IF:**

Orbit information will be (user specified local sun and planet vector)

**THEN:**

X> RUN(SOLAR.BAT [XS],[YS],[ZS] /M /Z)

and: Solar view factors are (calculated)

**Purpose:**

Rules 29 and 30 have similar purposes: to calculate the local sun vector and the solar view factors. Rule 29 is for a spacecraft in planet orientation, and Rule 30 is for a spacecraft in solar inertial orientation.

**Premise:**

The qualifier "Orbit information will be" (Q3) has the value "user specified global sun and Planet vector with orbit information (orbital elements)". The sun vector is entered in global coordinates.

The qualifier "The spacecraft orientation is" (Q4) has the value "planet" or "inertial". The user is prompted with a screen that contains two buttons, one for planet orientation, and one for solar inertial orientation.

**Conclusion:**

Run the external DOS file "SUNVECT.BAT", which passes the global sun vector [GS1], [GS2], and [GS3]. The local sun vector,[XS1], [XS2], and [XS3], and the six solar view factors are returned [MU1], [MU2], [MU3], [MU4], [MU5] and [MU6].

The qualifier "Solar view factors are" is assigned the value "calculated".

**/\* RULE NUMBER: 29**

**RULE: ORBIT ANGELS**

**IF:**

Orbit information will be {user specified global sun and Planet vector with orbit information (orbital elements)}

and: The spacecraft orientation is {planet}

**THEN:**

X> RUN(SUNVECT.BAT [GS1],[GS2],[GS3],  
[INC], [OMEGA], [WOMEGA] , [v0] /M /Z)

and: Solar view factors are {calculated}

**/\* RULE NUMBER: 30**

**IF:**

Orbit information will be {user specified global sun and Planet vector with orbit information (orbital elements)}

and: The spacecraft orientation is {inertial}

**THEN:**

X> RUN(SUNVECT.BAT [GS1],[GS2],[GS3],  
[INC], [OMEGA],0,0 /M /Z)

and: Solar view factors are {calculated}

**Purpose:**

Define the defaults for the enclosure analysis.

**Premise:**

The qualifier "Orbit information will be" (Q3) has the value "defaults".

**Conclusion:**

Use the system defaults which are:

semi-major axis of 6785 km  
eccentricity of 0 deg  
true anomaly of 0 deg  
inclination of 0 deg  
line of nodes of 0 deg  
line of periapsis of 0 deg  
local sun vector of (-1,0,0)  
the solar view factors are (0,0,1,0,0,0)

The qualifier "Solar view factors are" is assigned the value "calculated".

/\* RULE NUMBER: 31

RULE: ORBIT DEFAULTS

**IF:**

Orbit information will be {defaults}

**THEN:**

[SMA] IS GIVEN THE VALUE 6785.0  
and: [ECC] IS GIVEN THE VALUE 0  
and: [V0] IS GIVEN THE VALUE 0.0  
and: [INC] IS GIVEN THE VALUE 0  
and: [OMEGA] IS GIVEN THE VALUE 0  
and: [WOMEGA] IS GIVEN THE VALUE 0  
and: [XS] IS GIVEN THE VALUE -1.0  
and: [YS] IS GIVEN THE VALUE 0.0  
and: [ZS] IS GIVEN THE VALUE 0.0  
and: [MU1] IS GIVEN THE VALUE 0  
and: [MU2] IS GIVEN THE VALUE 0  
and: [MU3] IS GIVEN THE VALUE 1  
and: [MU4] IS GIVEN THE VALUE 0  
and: [MU5] IS GIVEN THE VALUE 0  
and: [MU6] IS GIVEN THE VALUE 0  
and: Solar view factors are {calculated}

**Purpose:**

Finish the planet view factors calculations after the user decides to not modify the calculated planet view factors.

**Premise:**

The qualifier "planet view factors are" (Q8) has the value "calculated", which is derived from Rules 34-36.

The qualifier "planet view factors are to be" (Q14) has the value "left alone". The user is allowed to modify the planet view factors calculated by Help Desk. A screen attached to Q14 is displayed asking the question if the user wants to modify the planet view factors. If the user selects "no", the value "left alone" is returned.

**Conclusion:**

The qualifier "Planet view factors calculations are" is assigned the value "finished".

**Purpose:**

Allows the user to modify the calculated planet view factors.

**Premise:**

The qualifier "planet view factors are" (Q8) has the value "calculated", which is derived from Rules 34-36.

The qualifier "planet view factors are to be" (Q14) has the value "changed". The user is allowed to modify the planet view factors calculated by Help Desk. A screen attached to Q14 is displayed asking the question if the user wants to modify the planet view factors. If the user selects "yes", the value "changed" is returned.

**Conclusion:**

The planet view factors are displayed, and the modified values are assigned to the variables [FE1], [FE2], [FE3], [FE4], [FE5], and [FE6]

The qualifier "Planet view factors calculations are" is assigned the value "finished".

**/\* RULE NUMBER: 32**

**RULE: LEAVE EVF**

**IF:**

Planet view factors are {calculated}

and: Planet view factors are to be {left alone}

**THEN:**

Planet view factors calculations are {finished}

**/\* RULE NUMBER: 33**

**RULE: EVF CHANGED**

**IF:**

Planet view factors are {calculated}

and: Planet view factors are to be {changed}

**THEN:**

[FE1] IS GIVEN THE VALUE [FE1U]

and: [FE2] IS GIVEN THE VALUE [FE2U]

and: [FE3] IS GIVEN THE VALUE [FE3U]

and: [FE4] IS GIVEN THE VALUE [FE4U]

and: [FE5] IS GIVEN THE VALUE [FE5U]

and: [FE6] IS GIVEN THE VALUE [FE6U]

and: Planet view factors calculations are {finished}

**Purpose:**

Calculate the planet view factors for an inertially oriented spacecraft.

**Premise:**

The qualifier "The spacecraft orientation is" (Q4) has the value "inertial".

The qualifier "The spacecraft orbit is" (Q5) has the value "LEO (Low Earth Orbit)".

**Conclusion:**

Run the external program "PLANET.BAT". It returns the six planet view factor, [FE1], [FE2], [FE3], [FE4], [FE5], and [FE6].

The qualifier "Planet view factors are" is assigned the value "calculated".

**Purpose:**

Calculate the planet view factors for an planet oriented spacecraft.

**Premise:**

The qualifier "The spacecraft orientation is" (Q4) has the value "planet".

The qualifier "The spacecraft orbit is" (Q5) has the value "LEO (Low Earth Orbit)".

**Conclusion:**

Assign the values to the six planet view factor, [FE1], [FE2], [FE3], [FE4], [FE5], and [FE6]. In the variable definition of [FET] and [FEN] is a call statement to an external DOS executable, "FETANOR.BAT". The file calculates the normal and tangential components of the solar view factor.

The qualifier "Planet view factors are" is assigned the value "calculated".

/\* RULE NUMBER: 34

RULE: INERTIAL LEO

**IF:**

The spacecraft orientation is {inertial}

and: The spacecraft orbit is {LEO (Low Earth Orbit)}

**THEN:**

X> RUN(PLANET.BAT {INC}, {ECC},{WOMEGA} ,{SMA},{V0},{RE}  
/M /Z)

and: Planet view factors are {calculated}

/\* RULE NUMBER: 35

RULE: PLANET LEO

**IF:**

The spacecraft orientation is {planet}

and: The spacecraft orbit is {LEO (Low Earth Orbit)}

**THEN:**

[FE1] IS GIVEN THE VALUE [FET]

and: [FE2] IS GIVEN THE VALUE [FET]

and: [FE3] IS GIVEN THE VALUE 0.0

and: [FE4] IS GIVEN THE VALUE [FET]

and: [FE5] IS GIVEN THE VALUE [FEN]

and: [FE6] IS GIVEN THE VALUE [FET]

and: Planet view factors are {calculated}

**Purpose:**  
Assign the planet view factors for a Geosynchronous spacecraft.

**Premise:**  
The qualifier "The spacecraft orbit is" (Q5) has the value "GEO (Geosynchronous Earth Orbit)".

**Conclusion:**  
Assign the values to the six planet view factor, [FE1], [FE2], [FE3], [FE4], [FE5], and [FE6]. In GEO, the planet view factors are effectively zero.

Assign the value of 35744 km to the variable [SMA], the semi-major axis.

The qualifier "Planet view factors are" is assigned the value "calculated".

**Purpose:**  
Prevent the user from being prompted for insulated side information, when the user has already defined there are no insulated sides.

**Premise:**  
The variable [NUMINS] has the value 0. There are no insulated sides during the analysis.

**Conclusion:**  
Assign the value 0.0 to the variable [INS1]. The screen containing the additional information on the insulated sides is attached to the variable [INS1]. If the variable has a value, it does not prompt the user for the information.

**/\* RULE NUMBER: 36**  
**RULE: GEO**

**IF:**  
The spacecraft orbit is [GEO (Geosynchronous Earth Orbit)]

**THEN:**  
[FE1] IS GIVEN THE VALUE 0.0  
and: [FE2] IS GIVEN THE VALUE 0.0  
and: [FE3] IS GIVEN THE VALUE 0.0  
and: [FE4] IS GIVEN THE VALUE 0.0  
and: [FE5] IS GIVEN THE VALUE 0.0  
and: [FE6] IS GIVEN THE VALUE 0.0  
and: [SMA] IS GIVEN THE VALUE 35744  
and: Planet view factors are (calculated)

**/\* RULE NUMBER: 37**  
**RULE: NUMINS**

**IF:**  
[NUMINS] = 0

**THEN:**  
[INS1] IS GIVEN THE VALUE 0.0

**Purpose:**

Rules 38-41 have the same purpose: to define the surface properties; solar absorptivity and emissivity.

**Premise:**

The premise clauses for the four rules are similar, will all possible combinations on two qualifiers.

The qualifier "Value to calculate" (Q1) has the value "surface properties". The value is selected from the Main menu.

The qualifier "The performance time is" (Q9) may have the value "BOL" or "EOL". The user is shown a screen with two buttons. The button chosen returns the value to the qualifier.

The qualifier "The known surface information is" (Q10) may have the value "surface use" or "surface application". Another two button screen is displayed for the user to choose from.

**Conclusion:**

The variable [EPSILON] is assigned the value of the variable as shown. These variables have frame calls to the frame "SURFACE.TXT" stated in the variable definitions. Frames are database calls to the small ASCII database called a frame.

The variable [ALPHA], [MATERIAL] or [APPLICATION] are defined as shown with frame calls to the same file.

```
/* RULE NUMBER: 38
RULE: FRAME EPS BOL
```

**IF:**

```
Value to calculate is {surface properties}
and: the performance time is {BOL (beginning of life)}
and: the known surface information is {surface use}
```

**THEN:**

```
[EPSILON] IS GIVEN THE VALUE [A_EPSFB]
and: [ALPHA] IS GIVEN THE VALUE [A_ABSFB]
and: [MATERIAL] IS GIVEN THE VALUE FRAME("surface.txt",
$TYPICAL_APPLICATION$==[APPLICATION], $SURFACE$)
and: Surface properties are {finished}
```

```
/* RULE NUMBER: 39
RULE: FRAME EPS EOL
```

**IF:**

```
Value to calculate is {surface properties}
and: the performance time is {EOL (end of life)}
and: the known surface information is {surface use}
```

**THEN:**

```
[EPSILON] IS GIVEN THE VALUE [A_EPSFE]
and: [ALPHA] IS GIVEN THE VALUE [A_ABSFE]
and: [MATERIAL] IS GIVEN THE VALUE FRAME("surface.txt",
$TYPICAL_APPLICATION$==[APPLICATION], $SURFACE$)
and: Surface properties are {finished}
```



/\* RULE NUMBER: 40

IF:

Value to calculate is {surface properties}

and: the performance time is {EOL (end of life)}

and: the known surface information is {surface material}

THEN:

[EPSILON] IS GIVEN THE VALUE [M\_EPSFE]

and: [ALPHA] IS GIVEN THE VALUE [M\_ABSFE]

and: [APPLICATION] IS GIVEN THE VALUE FRAME("surface.txt",  
\$SURFACE\$==[MATERIAL], \$TYPICAL\_APPLICATION\$)

and: Surface properties are {finished}

/\* RULE NUMBER: 41

IF:

Value to calculate is {surface properties}

and: the performance time is {BOL (beginning of life)}

and: the known surface information is {surface material}

THEN:

[EPSILON] IS GIVEN THE VALUE [M\_EPSFB]

and: [ALPHA] IS GIVEN THE VALUE [M\_ABSFB]

and: [APPLICATION] IS GIVEN THE VALUE FRAME("surface.txt",  
\$SURFACE\$==[MATERIAL], \$TYPICAL\_APPLICATION\$)

and: Surface properties are {finished}

**Purpose:**

To allow the user to modify the surface properties calculated by Help Desk.

**Premise:**

The qualifier "assistance is" (Q) has the value "needed".

The qualifier "surface properties are" (Q) has the value "calculated". This is derived using Rules 43-46.

The qualifier "Change of surface properties is" (Q) has the value "requested". The value is input by the user. The user is shown a screen containing the solution to the surface properties analysis (Rules 43-46). The screen also contains two buttons one to continue, and one to modify the values. The modify button returns the value "requested" to the qualifier.

**Conclusion:**

A screen is displayed allowing the user to modify the emissivity and absorptivity.

/\* RULE NUMBER: 42

RULE: CHANGE SP

**IF:**

assistance is {needed}

and: Surface properties are {calculated}

and: Change of surface properties is {requested}

**THEN:**

[EPSILON] IS GIVEN THE VALUE [CEPS]

and: [ALPHA] IS GIVEN THE VALUE [CALP]

**Purpose:**

Rules 43-46 have the same purpose: to define the surface properties, solar absorptivity and emissivity. During a surface analysis, the user may ask the expert system for assistance in determining the surface properties, emissivity and absorptivity.

**Premise:**

The premise clauses for the four rules are similar, will all possible combinations on two qualifiers.

The qualifier "Assistance is" (Q16) has the value "needed", which is derived using the screen Q16. At the bottom of the screen is a button called "assistance". When this is pressed, the value "needed" is returned.

The qualifier "The performance time is" (Q9) may have the value "BOL" or "EOL". The user is shown a screen with two buttons. The button chosen returns the value to the qualifier.

The qualifier "The known surface information is" (Q10) may have the value "surface use" or "surface application". Another two button screen is displayed for the user to choose from.

**Conclusion:**

The variable [EPSILON] is assigned the value of the variable as shown. These variables have frame calls to the frame "SURFACE.TXT" in the variable definitions. They are database calls to the small ASCII database called a frame.

The variable [ALPHA], [MATERIAL] or [APPLICATION] are defined as shown with frame calls to the same file.

/\* RULE NUMBER: 43  
RULE: ASSIST APP BOL

**IF:**

assistance is {needed}

and: the performance time is {BOL (beginning of life)}

and: the known surface information is {surface use}

**THEN:**

[EPSILON] IS GIVEN THE VALUE [A\_EPSFB]

and: [ALPHA] IS GIVEN THE VALUE [A\_ABSFB]

and: [MATERIAL] IS GIVEN THE VALUE FRAME("surface.txt",  
\$TYPICAL\_APPLICATION\$==[APPLICATION], \$SURFACE\$)

and: Surface properties are {calculated}

/\* RULE NUMBER: 44  
RULE: ASSIST APP EOL

**IF:**

assistance is {needed}

and: the performance time is {EOL (end of life)}

and: the known surface information is {surface use}

**THEN:**

[EPSILON] IS GIVEN THE VALUE [A\_EPSFE]

and: [ALPHA] IS GIVEN THE VALUE [A\_ABSFE]

and: [MATERIAL] IS GIVEN THE VALUE FRAME("surface.txt",  
\$TYPICAL\_APPLICATION\$==[APPLICATION], \$SURFACE\$)

and: Surface properties are {calculated}

/\* RULE NUMBER: 45  
RULE: ASSIST MAT BOL

IF:

assistance is (needed)  
and: the performance time is (EOL (end of life))  
and: the known surface information is (surface material)

THEN:

[EPSILON] IS GIVEN THE VALUE [M\_EPSFE]  
and: [ALPHA] IS GIVEN THE VALUE [M\_ABSFE]  
and: [APPLICATION] IS GIVEN THE VALUE FRAME("surface.txt",  
\$SURFACE\$=[MATERIAL], \$TYPICAL\_APPLICATION\$)  
and: Surface properties are (calculated)

/\* RULE NUMBER: 46  
RULE: ASSIST MAT EOL

IF:

assistance is (needed)  
and: the performance time is (BOL (beginning of life))  
and: the known surface information is (surface material)

THEN:

[EPSILON] IS GIVEN THE VALUE [M\_EPSFB]  
and: [ALPHA] IS GIVEN THE VALUE [M\_ABSFB]  
and: [APPLICATION] IS GIVEN THE VALUE FRAME("surface.txt",  
\$SURFACE\$=[MATERIAL], \$TYPICAL\_APPLICATION\$)  
and: Surface properties are (calculated)

**Purpose:**

Define the values for solar flux, planet flux, planet albedo and planet radius. During an analysis, the user may select to run the analysis around a different planet than the Earth. The user can input the known values for the fluxes, the albedo and the radius of the planet, or the user can select the planet from a list. The system will use the defaults stored in the frame "PLANET.TXT".

**Premise:**

The qualifier "A different planet is" (Q17) has the value "requested", which is derived using the screen Q17. At the bottom of the screen is a button called "different planet". The user has the option to input the values of any planet in the edit boxes supplied. If the values are unknown, the user may select the "different planet" button. The value "requested" is returned. The expert system will then prompt the user with a list of available planets. The list is attached to the variable [PLANET].

**Conclusion:**

The values for the variables are taken from the frame "PLANET.TXT". It is located in the same directory as the expert system files. The values returned from the frame are assigned to the variables and are used in the rest of the analysis.

**/\* RULE NUMBER: 47  
RULE: PLANET FRAME**

**IF:**

a different planet is {requested}

**THEN:**

[S] IS GIVEN THE VALUE  
FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#SOLAR\_FLUX#)  
and: [E] IS GIVEN THE VALUE  
FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#PLANET\_FLUX#)  
and: [ALB] IS GIVEN THE VALUE  
FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#ALBEDO#)  
and: [RE] IS GIVEN THE VALUE  
FRAME("PLANET.TXT",\$PLANET\$==[PLANET],#RADIUS#)

THIS PAGE INTENTIONALLY LEFT BLANK

**Appendix B  
Qualifiers**

**Qualifiers**

**B1**

**PRECEDING PAGE BLANK NOT FILMED**

**THIS PAGE INTENTIONALLY LEFT BLANK**



/\* Qualifier 1

Q> Value to calculate is

V> radiator area (given temperature)

V> surface temperature

V> enclosure temperatures and heat loads

V> orbital average temperatures and heat loads

V> surface properties

V> none of the above

Name: CALCULATION OPT

Maximum acceptable = 1

/\* Qualifier 2

Q> Radiation effects to include

V> sun

V> planet

V> albedo

V> enclosure radiation

V> none (just radiate internal power)

Name: RADIATION EFPEC

/\* Qualifier 3

Q> Orbit information will be

V> user specified local sun and Planet vector

V> user specified global sun and Planet vector with orbit information (orbital elements)

V> defaults

V> User specified Beta angle

Name: ORBIT INFO

Maximum acceptable = 1

/\* Qualifier 4

Q> The spacecraft orientation is

V> planet

V> inertial

Name: PLANET/INERTIAL

Maximum acceptable = 1

/\* Qualifier 5

Q> The spacecraft orbit is

V> LEO (Low Earth Orbit)

V> GEO (Geosynchronous Earth Orbit)

Name: LEO/GEO

Maximum acceptable = 1

/\* Qualifier 6

Q> Solar view factors are

V> calculated

V> unknown

Name: SOLAR VIEW FACT

Default value = 2

Maximum acceptable = 1

/\* Qualifier 7

Q> The temperature values are

V> calculated

V> unknown

Name: TEMPERATURE

Default value = 2

Maximum acceptable = 1

/\* Qualifier 8

Q> Planet view factors are

V> calculated

V> unknown

Name: PLANET VIEW FAC

Maximum acceptable = 1

/\* Qualifier 9

Q> the performance time is

V> BOL (beginning of life)

V> EOL (end of life)

Name: BOL/EOL

Maximum acceptable = 1

/\* Qualifier 10  
Q> the known surface information is

V> surface use  
V> surface material

Name: SURFACE INFO  
Maximum acceptable = 1

/\* Qualifier 11  
Q> Orbital analysis is

V> finished  
V> undone

Name: ORBITAL ANAL  
Maximum acceptable = 1

/\* Qualifier 12  
Q> Surface temperature to be calculated is

V> radiator temperature given area  
V> environmental temperature

Name: SURFACE TEMP  
Maximum acceptable = 1

/\* Qualifier 13  
Q> Surface properties are

V> calculated  
V> unknown

Name: SURF PROPS CAL

/\* Qualifier 14  
Q> Planet view factors are to be

V> changed  
V> left alone

Name: CHANGE EVF

/\* Qualifier 15  
Q> Planet view factors calculations are

V> finished  
V> undecided

Name: PLANET VF FIN

/\* Qualifier 16  
Q> assistance is

V> needed  
V> not needed

Name: ASSISTANCE

/\* Qualifier 17  
Q> a different planet is

V> requested  
V> not requested

Name: DIFFERENT PLANE

/\* Qualifier 18  
Q> Change of surface properties is

V> requested  
V> not needed

Name: CHANGE SP

/\* Qualifier 19  
Q> The report information is to be

V> appended to the current file  
V> written to a new file  
V> no file output

Name: APPEND TO FILE  
Default value = 3  
Maximum acceptable = 1

/\* Qualifier 20  
Q> Thermal analysis is

V> completed  
V> undone

Name: THERMAL DONE

/\* Qualifier 21  
Q> Enclosure radiation effects are to be

V> calculated  
V> user input

Name: CAL RAD EFF

**Appendix C**  
**Screens**

**THIS PAGE INTENTIONALLY LEFT BLANK**

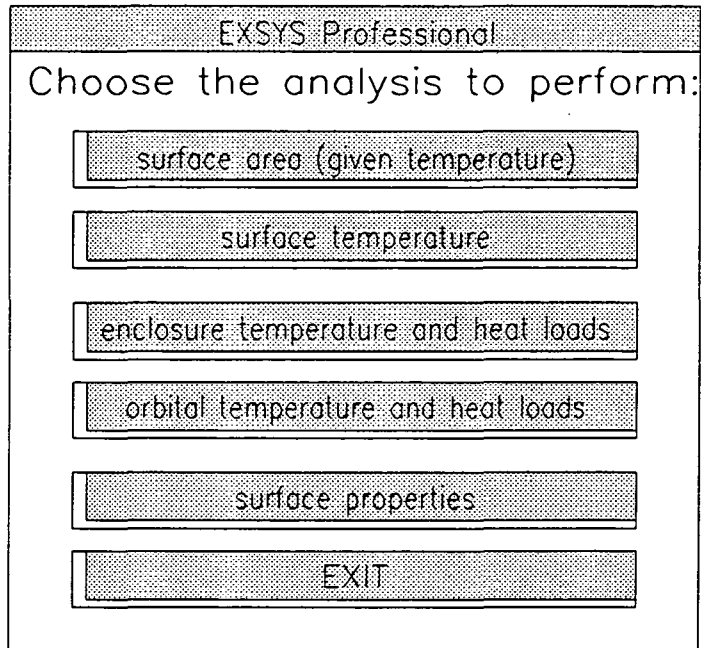
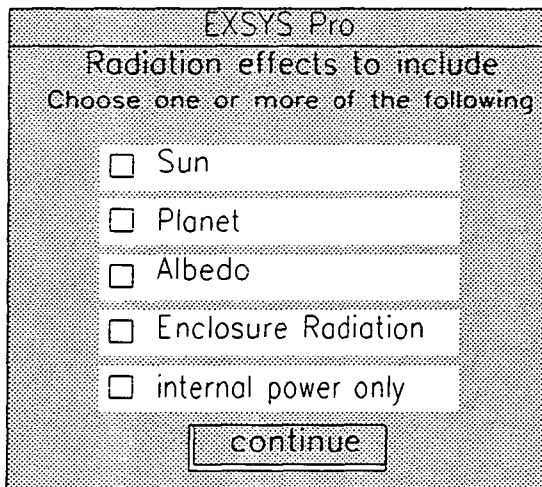


Figure 23 ~Q1

```

-Q1
-SET_WIN 88, 0, 517, 366
-LINE_COLOR BLACK
-FOREGROUND BLACK
-BACKGROUND CYAN
-CLS
-TEXTSIZE LARGE
-CURSET 34, 39
Choose the analysis to perform:
-BUTTON "surface area (given temperature)", "q1 1", 60, 60, 395, 40 -I
-BUTTON "surface temperature", "q1 2", 60, 110, 395, 40 -I
-BUTTON "enclosure temperatures and heat loads", "q1 3", 60, 160, 395, 40 -I
-BUTTON "orbital temperatures and heat loads", "q1 4", 60, 210, 395, 40 -I
-BUTTON "surface properties", "q1 5", 60, 260, 395, 40 -I
-BUTTON "EXIT", "Q1 6", 60, 310, 395, 40 -I
-END

```

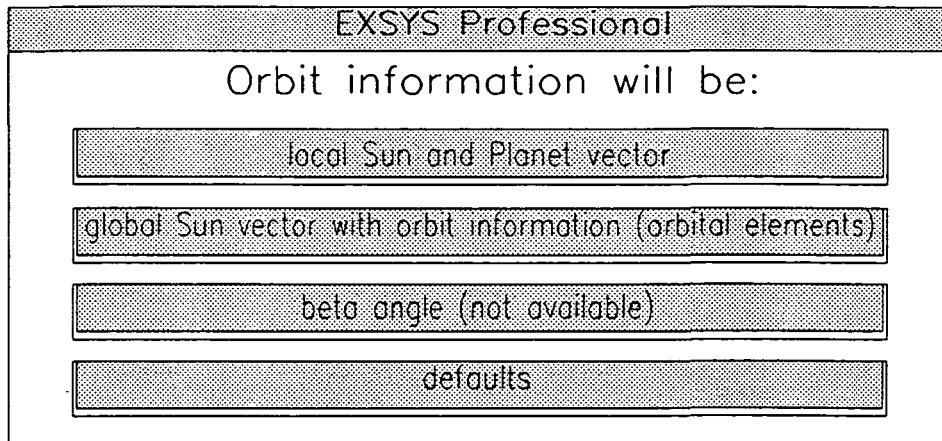


**Figure 24 -Q2**

```

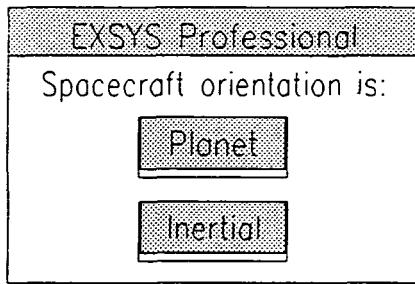
~Q2
~SET_WIN 88, 0, 375, 335
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 70, 25
Radiation effects to include:
~CURSET 32, 59
Choose one or more of the following
~BUTTON "Sun", "q2 1", 65, 77, 238, 35 -C
~BUTTON "Planet", "q2 2", 65, 120, 238, 35 -C
~BUTTON "Albedo", "q2 3", 65, 160, 238, 35 -C
~BUTTON "Enclosure Radiation", "q2 4", 65, 199, 238, 35 -C
~BUTTON "internal power only", "q2 5", 65, 240, 238, 35 -C
~BUTTON "CONTINUE", "", 127, 289, 100, 33 -I
~END

```



**Figure 25 -Q3**

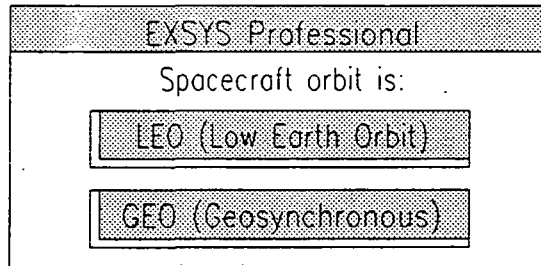
```
~Q3
~SET_WIN 88, 0, 518, 252
~BACKGROUND CYAN
~CLS
~TEXTSIZE LARGE
~CURSET 86, 33
Orbit information will be:
~BUTTON "local sun and Planet vector", "Q3 1", 20, 47, 476, 35 -I
~BUTTON "global sun vector with orbit information (orbital elements)", "Q3 2", 20, 90, 476, 35 -I
~BUTTON "beta angle (not available)", "", 20, 130, 476, 35
~BUTTON "defaults", "Q3 3", 20, 170, 476, 35 -I
~END
```



**Figure 26 -Q4**

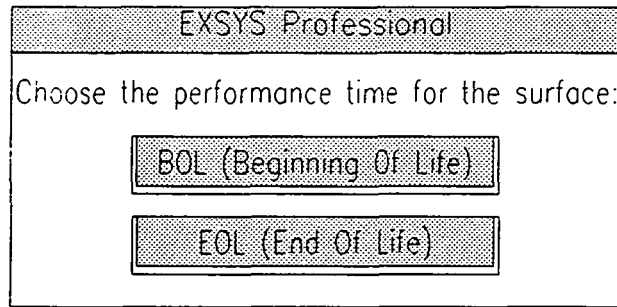
```
-Q4
~SET_WIN 88, 0, 249, 140
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 12, 30
Spacecraft orientation is:
~BUTTON "Planet", "Q4 1", 69, 49, 100, 35 -I
~BUTTON "Inertial", "Q4 2", 68, 93, 100, 35 -I
~END
```





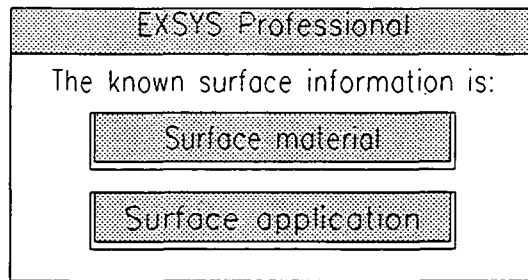
**Figure 27 ~Q5**

```
~Q5
~SET_WIN 88, 0, 306, 143
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 67, 28
Spacecraft orbit is:
~BUTTON "LEO (Low Earth Orbit)", "Q5 1", 27, 45, 243, 35 -I
~BUTTON "GEO (Geosynchronous)", "Q5 2", 26, 92, 243, 35 -I
~END
```



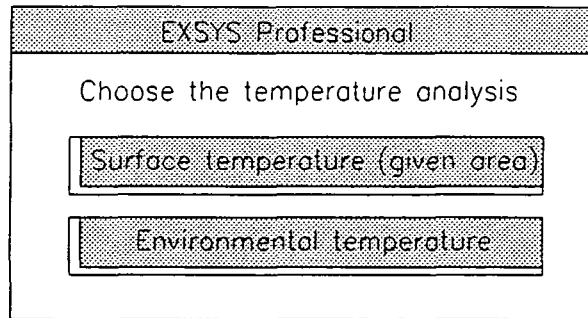
**Figure 28 -Q9**

```
~Q9
~SET_WIN 88, 0, 356, 143
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 19, 31
The performance time for the surface:
~BUTTON "BOL (Beginning Of Life)", "q9 1", 62, 44, 229, 35 -I
~BUTTON "EOL (End Of Life)", "q9 2", 62, 86, 229, 35 -I
~END
```



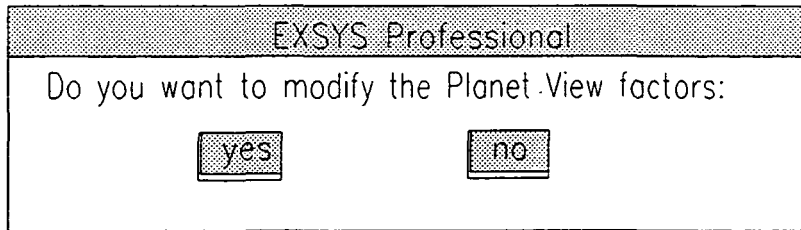
**Figure 29 ~Q10**

```
~Q10
~SET_WIN 88, 0, 313, 158
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 11, 32
The known surface information is:
~BUTTON "Use of the surface", "q10 1", 34, 95, 240, 35 -I
~BUTTON "Surface material", "q10 2", 34, 52, 240, 35 -I
~END
```



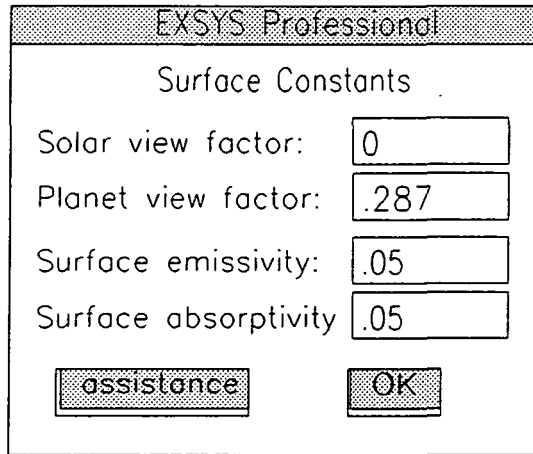
**Figure 30 -Q12**

```
-Q12
-SET_WIN 88, 0, 353, 145
-LINE_COLOR BLACK
-FOREGROUND BLACK
-BACKGROUND CYAN
-CLS
-CURSET 27, 29
Choose the temperature analysis
-BUTTON "Surface temperature (given area)", "q12 1", 35, 52, 274, 35 -I
-BUTTON "Environmental temperature", "q12 2", 35, 97, 275, 35 -I
-END
```



**Figure 31 ~Q14**

```
~Q14
~SET_WIN 88, 0, 312, 138
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 22, 29
modify the planet view factors?
~BUTTON "yes", "Q14 1", 116, 47, 75, 35 -I
~BUTTON "no", "Q14 2", 116, 96, 75, 35 -I
~END
```



**Figure 32 ~Q16**

```

~Q16
~SET_WIN 88, 0, 380, 280
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 110, 35
Surface Constants
~CURSET 30, 75
Solar view factor:
~CURSET 30, 115
Planet view factor:
~CURSET 30, 155
Emissivity:
~CURSET 30, 195
Absorptivity:
~EDIT "[MU]", 225, 50, 103, 35 -M:1
~EDIT "[FE]", 225, 90, 103, 35 -M:1
~EDIT "[EPSILON]", 225, 130, 103, 35 -M:1
~EDIT "[ALPHA]", 225, 170, 103, 35 -M:1
~BUTTON "OK", "Q16 2", 261, 228, 75, 35 -I
~BUTTON "assistance", "q16 1", 38, 228, 158, 35 -I
~EDIT_FILL ALPSCR.DAT
~END

```

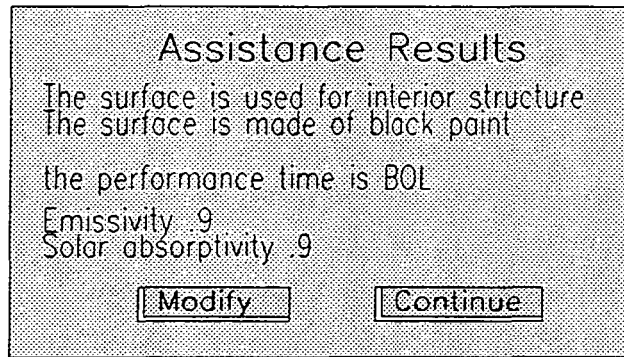
EXSYS Professional	
Planet Constants	
Albedo coefficient:	<input type="text" value=".3"/>
Solar Flux: W/m <sup>2</sup>	<input type="text" value="1356"/>
Planet Flux: W/m <sup>2</sup>	<input type="text" value="236"/>
Planet radius: km	<input type="text" value="6378"/>
<input type="button" value="different planet"/>	<input type="button" value="OK"/>

Figure 33 ~Q17

```

~Q17
~SET_WIN 88, 0, 360, 271
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 102, 30
Planet Constants
~CURSET 10, 70
Albedo coefficient:
~CURSET 10, 110
Solar Flux: W/m^2
~CURSET 10, 150
Planet Flux: W/m^2
~CURSET 10, 190
Planet radius: km
~EDIT "[ALB]", 225, 45, 103, 35 -M:1
~EDIT "[S]", 225, 85, 103, 35 -M:1
~EDIT "[E]", 225, 125, 103, 35 -M:1
~EDIT "[RE]", 225, 165, 103, 35 -M:1
~BUTTON "OK", "Q17 2", 241, 217, 75, 35 -I
~BUTTON "different planet", "q17 1", 23, 216, 158, 35 -I
~EDIT_FILL ALBSCR.DAT
~END

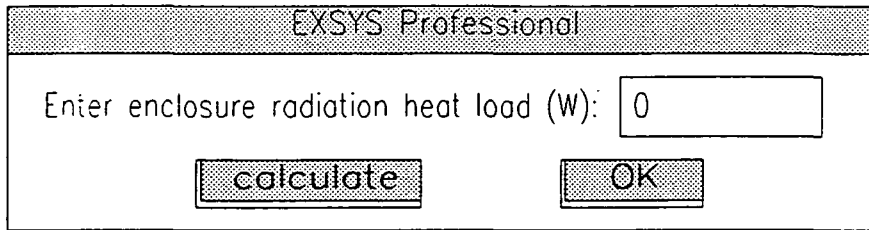
```



**Figure 34 ~Q18**

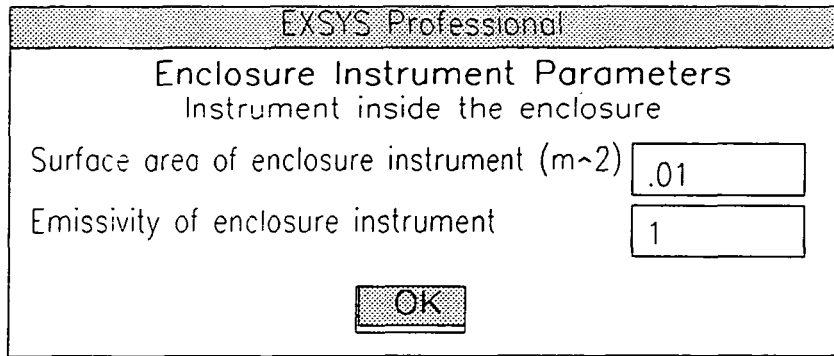
```
-Q18
-SET_WIN 0, 0, 584, 266
-LINE_COLOR BLACK
-FOREGROUND BLACK
-BACKGROUND CYAN
-CLS
-CURSET 182, 29
Assitance Results
-CURSET 54, 62
The surface is used for [[APPLICATION]]
-CURSET 54, 96
The surface is made of [[MATERIAL]]
-CURSET 54, 132
[[*q9]]
-CURSET 54, 168
Emissivity [[EPSILON]]
-CURSET 54, 202
Solar absorptivity [[ALPHA]]
-BUTTON "Modify", "Q18 1", 113, 213, 106, 35 -I
-BUTTON "Continue", "Q18 2", 304, 213, 106, 35 -I
-END
```





**Figure 35 -Q21**

```
~Q21
~SET_WIN 88, 0, 539, 120
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 16, 35
Enter enclosure radiation heat load (W):
~EDIT "[QENCL]", 396, 11, 103, 35 -M:1
~BUTTON "OK", "Q21 2", 330, 66, 75, 35 -I
~BUTTON "calculate", "Q21 1", 118, 66, 158, 35 -I
~EDIT_FILL QENCLSCR.DAT
~END
```

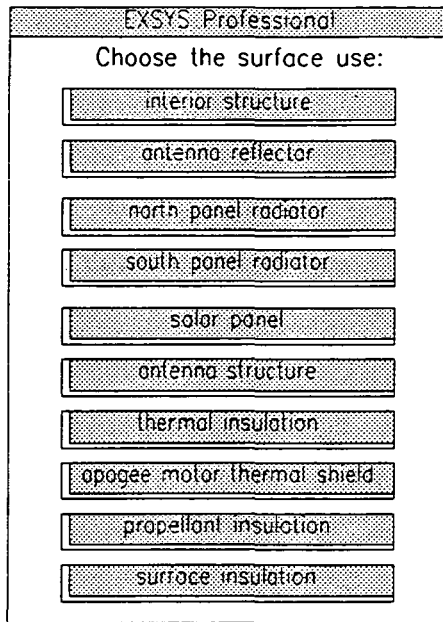


**Figure 36** ~[AINST]

```

-[AINST]
~SET_WIN 88, 0, 546, 176
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 137, 30
Enclosure Instrument Parameters
~TEXTSIZE SMALL
~CURSET 169, 51
Instrument inside the enclosure
~TEXTSIZE NORMAL
~CURSET 29, 77
Surface area of enclosure instrument (m^2)
~CURSET 29, 114
Emissivity of enclosure instrument
~EDIT "[AINST]", 426, 50, 100, 35 -M:1
~EDIT "[EPSINST]", 425, 92, 100, 35 -M:1
~BUTTON "OK", "", 232, 128, 75, 35 -I
~EDIT_FILL AINSTSCR.DAT
~END

```

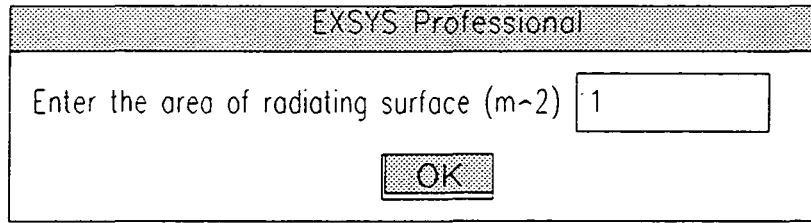


**Figure 37** ~[APPLICATION]

```

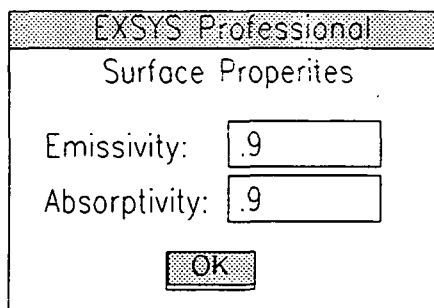
~[APPLICATION]
~SET_WIN 88, 0, 319, 359
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 30, 22
Choose the surface application:
~BUTTON "interior structure", "[APPLICATION] 'interior structure'", 25, 30, 272, 25 -I
~BUTTON "antenna reflector", "[APPLICATION] 'antenna reflector'", 25, 60, 272, 25 -I
~BUTTON "north panel radiator", "[APPLICATION] 'north panel radiator'", 25, 90, 272, 25 -I
~BUTTON "south panel radiator", "[APPLICATION] 'south panel radiator'", 25, 120, 272, 25 -I
~BUTTON "solar panel", "[APPLICATION] 'solar panel'", 25, 150, 272, 25 -I
~BUTTON "antenna structure", "[APPLICATION] 'antenna structure'", 25, 180, 272, 25 -I
~BUTTON "thermal insulation", "[APPLICATION] 'thermal insulation'", 25, 210, 272, 25 -I
~BUTTON "apogee motor thermal shield", "[APPLICATION] 'apogee motor thermal shield'", 25, 240,
272, 25 -I
~BUTTON "propellant insulation", "[APPLICATION] 'propellant insulation'", 25, 270, 272, 25 -I
~BUTTON "surface insulation", "[APPLICATION] 'surface insulation'", 25, 300, 272, 25 -I
~END

```



**Figure 38** ~[AREA]

```
~[AREA]
~SET_WIN 88, 0, 515, 100
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 17, 37
Enter the area of radiating surface (m^2)
~EDIT "[AREA]", 400, 12, 103, 35 -M:1
~BUTTON "OK", "", 203, 54, 75, 35 -I
~EDIT_FILL AREASCR.DAT
~END
```

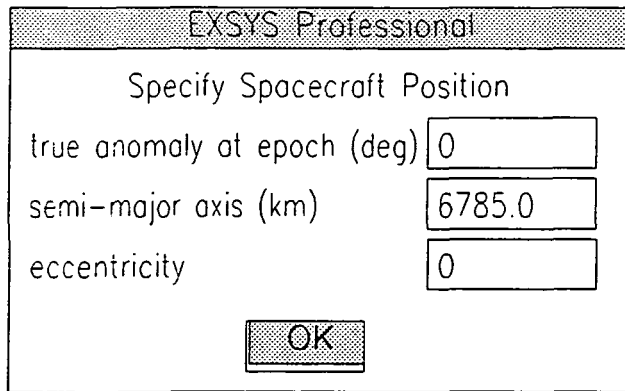


**Figure 39** ~[CEPS]

```

~[CEPS]
~SET_WIN 88, 0, 306, 197
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 79, 31
Surface Properties
~CURSET 30, 75
Emissivity:
~CURSET 30, 115
Absorptivity:
~EDIT "[CEPS]", 161, 50, 103, 35 -L
[[EPSILON]]
~LISTEND
~EDIT "[CALP]", 161, 90, 103, 35 -L
[[ALPHA]]
~LISTEND
~BUTTON "OK", "", 121, 147, 75, 35 -I
~END

```



**Figure 40** Spacecraft Position screen

```
~[ECC]
~SET_WIN 88, 0, 390, 224
~BACKGROUND CYAN
~CLS
~CURSET 91, 32
Specify Spacecraft Position
~CURSET 14, 75
true anomaly at epoch (deg)
~CURSET 14, 115
semi-major axis (km)
~CURSET 14, 155
eccentricity
~EDIT "[v0]", 280, 50, 100, 35 -M:1
~EDIT "[SMA]", 280, 90, 100, 35 -M:1
~EDIT "[ECC]", 280, 130, 100, 35 -M:1
~BUTTON "OK", "", 160, 177, 75, 36 -I
~EDIT_FILL ECCSCR.DAT
~END
```

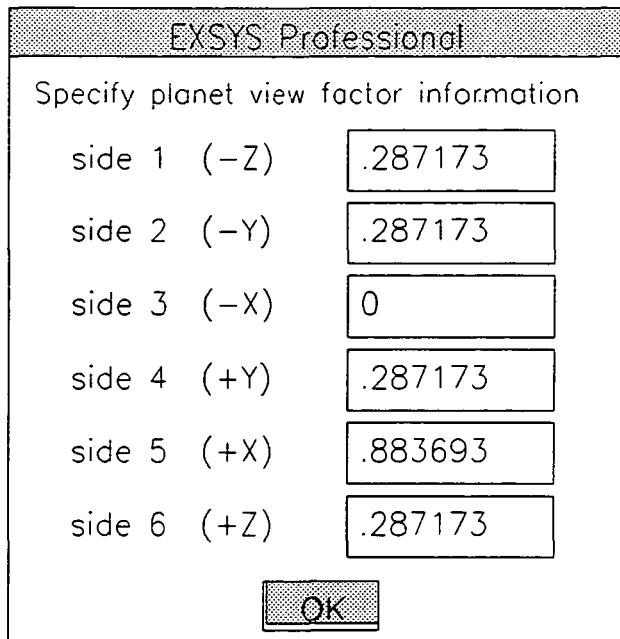


Figure 41 ~[FE1U]

```

~[FE1U]
~SET_WIN 88, 0, 334, 352
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 10, 30
Specify Planet view factor information
~CURSET 35, 65
side 1 (-Z)
~CURSET 35, 105
side 2 (-Y)
~CURSET 35, 145
side 3 (-X)
~CURSET 35, 185
side 4 (+Y)
~CURSET 35, 225
side 5 (+X)
~CURSET 35, 265
side 6 (+Z)
~EDIT "[FE1U]", 180, 40, 100, 35 -L
[[FE1]]
~LISTEND
~EDIT "[FE2U]", 180, 80, 100, 35 -L
[[FE2]]
~LISTEND
~EDIT "[FE3U]", 180, 120, 100, 35 -L
[[FE3]]
~LISTEND
~EDIT "[FE4U]", 180, 160, 100, 35 -L
[[FE4]]
~LISTEND
~EDIT "[FE5U]", 180, 200, 100, 35 -L
[[FE5]]
~LISTEND
~EDIT "[FE6U]", 180, 240, 100, 35 -L
[[FE6]]
~LISTEND
~BUTTON "OK", "", 140, 280, 75, 35 -I
~END

```

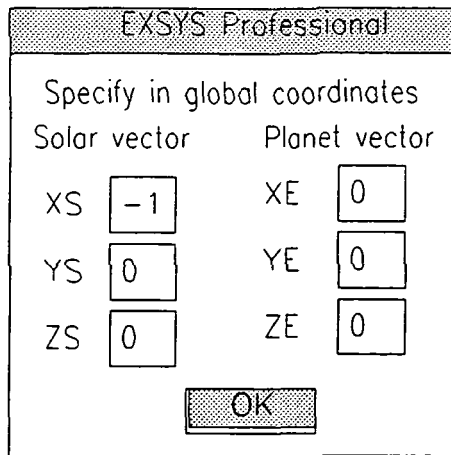


Figure 42 ~[GS1]

```

~[GS1]
~SET_WIN 88, 0, 324, 248
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 44, 23
Specify in global coordinates
~CURSET 18, 53
Solar Vector
~CURSET 20, 90
XS
~CURSET 20, 130
YS
~CURSET 20, 170
ZS
~EDIT "[GS1]", 75, 65, 49, 35 -M:1
~EDIT "[GS2]", 75, 105, 49, 35 -M:1
~EDIT "[GS3]", 75, 145, 49, 35 -M:1
~CURSET 178, 50
Planet Vector
~CURSET 175, 90
XE
~CURSET 175, 130
YE
~CURSET 175, 170
ZE
~EDIT "[GE1]", 230, 65, 49, 35 -M:1
~EDIT "[GE2]", 230, 105, 49, 35 -M:1
~EDIT "[GE3]", 230, 145, 49, 35 -M:1
~BUTTON "OK", "", 116, 201, 75, 35 -I
~EDIT_FILL GSSCR.DAT
~END

```



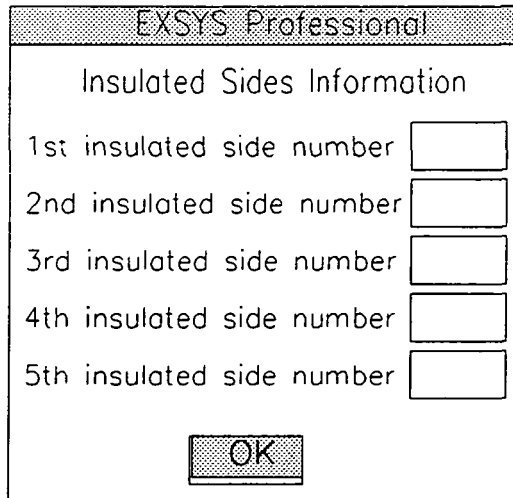
EXSYS Professional	
Specify orbit information	
inclination (deg)	<input type="text" value="28.5"/>
nodal crossing (deg)	<input type="text" value="0"/>
argument of periapsis (deg)	<input type="text" value="0"/>
true anomaly at epoch (deg)	<input type="text" value="0"/>
semi-major axis (km)	<input type="text" value="6785.0"/>
eccentricity	<input type="text" value="0"/>
<input type="button" value="OK"/>	

**Figure 43** ~[INC]

```

-[INC]
-SET_WIN 88, 0, 390, 351
-BACKGROUND CYAN
-CLS
-CURSET 91, 32
Specify orbit information
-CURSET 14, 75
inclination (deg)
-CURSET 14, 115
nodal crossing (deg)
-CURSET 14, 155
argument of periapsis (deg)
-CURSET 14, 195
true anomaly at epoch (deg)
-CURSET 14, 235
semi-major axis (km)
-CURSET 14, 275
eccentricity
-EDIT "[INC]", 280, 50, 100, 35 -M:1
-EDIT "[OMEGA]", 280, 90, 100, 35 -M:1
-EDIT "[WOMEGA]", 280, 130, 100, 35 -M:1
-EDIT "[v0]", 280, 170, 100, 35 -M:1
-EDIT "[SMA]", 280, 210, 100, 35 -M:1
-EDIT "[ECC]", 280, 250, 100, 35 -M:1
-BUTTON "OK", "", 140, 302, 75, 35 -I
-EDIT_FILL INCSCR.DAT
-END

```

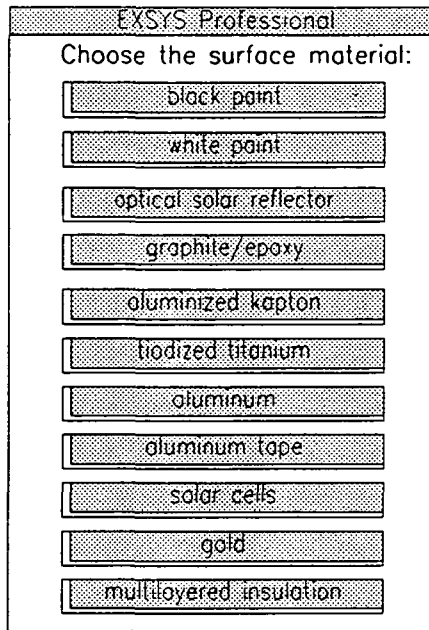


**Figure 44** ~[INS1]

```

~[INS1]
~SET_WIN 88, 0, 352, 310
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 64, 25
Insulated Sides Information
~CURSET 10, 65
1st insulated side number
~CURSET 10, 105
2nd insulated side number
~CURSET 10, 145
3rd insulated side number
~CURSET 10, 185
4th insulated side number
~CURSET 10, 225
5th insulated side number
~EDIT "[INS1]", 261, 39, 60, 33 -M:1
~EDIT "[INS2]", 261, 80, 60, 33 -M:1
~EDIT "[INS3]", 261, 120, 60, 33 -M:1
~EDIT "[INS4]", 261, 160, 60, 33 -M:1
~EDIT "[INS5]", 261, 200, 60, 33 -M:1
~BUTTON "OK", "", 125, 255, 75, 33 -I
~END

```

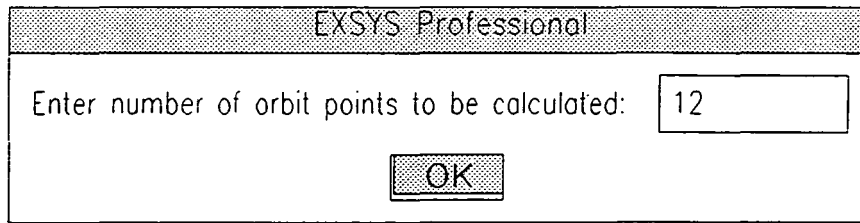


**Figure 45** ~[MATERIAL]

```

~[MATERIAL]
~SET_WIN 88, 0, 343, 440
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 42, 23
Choose the surface material:
~BUTTON "black paint", "[MATERIAL] 'black paint'", 25, 30, 285, 25 -I
~BUTTON "white paint", "[MATERIAL] 'white paint'", 25, 60, 285, 25 -I
~BUTTON "optical solar reflector", "[MATERIAL] 'optical solar reflector'", 25, 90, 285, 25 -I
~BUTTON "graphite/epoxy", "[MATERIAL] 'graphite/epoxy'", 25, 120, 285, 25 -I
~BUTTON "aluminized kapton", "[MATERIAL] 'aluminized kapton'", 25, 150, 285, 25 -I
~BUTTON "iodized titanium", "[MATERIAL] 'iodized titanium'", 25, 180, 285, 25 -I
~BUTTON "aluminum", "[MATERIAL] 'aluminum'", 25, 210, 285, 25 -I
~BUTTON "aluminum tape", "[MATERIAL] 'aluminum tape'", 25, 240, 285, 25 -I
~BUTTON "deposited aluminum", "[MATERIAL] 'deposited aluminum'", 25, 270, 285, 25 -I
~BUTTON "anodized aluminum", "[MATERIAL] 'anodized aluminum'", 25, 300, 285, 25 -I
~BUTTON "solar cells", "[MATERIAL] 'solar cells'", 25, 330, 285, 25 -I
~BUTTON "gold", "[MATERIAL] 'gold'", 25, 360, 285, 25 -I
~BUTTON "multilayered insulation", "[MATERIAL] 'multilayered insulation'", 25, 390, 285, 25 -I
~END

```



**Figure 46** ~[NUMPTS]

```
~[NUMPTS]
~SET_WIN 88, 0, 500, 120
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 13, 42
Enter number of orbit points to be calculated:
~EDIT "[NUMPTS]", 415, 20, 75, 35 -M:1
~BUTTON "OK", "", 201, 64, 75, 35 -I
~EDIT_FILL PTSSCR.DAT
~END
```

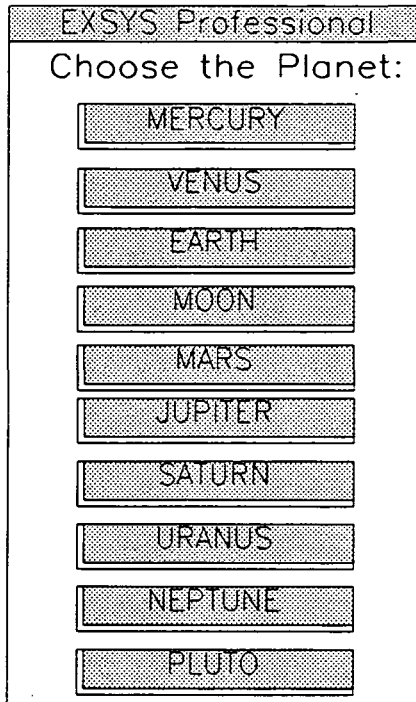
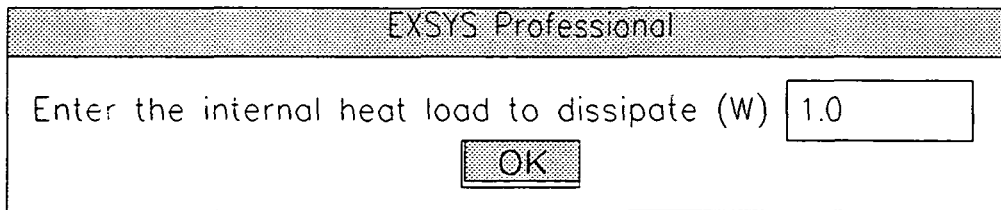


Figure 47 ~[PLANET]

```

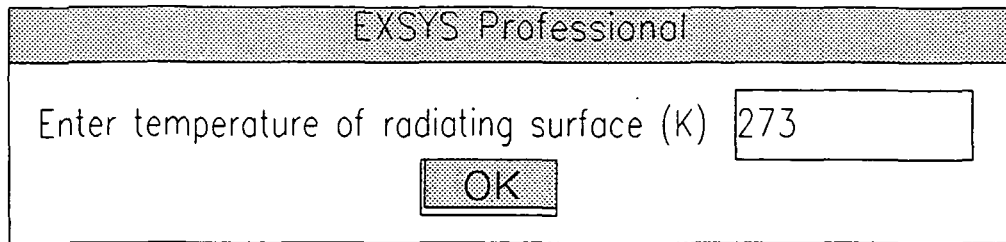
~[PLANET]
~SET_WIN 88, 0, 254, 345
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 38, 23
Choose the planet:
~BUTTON "MERCURY", "[PLANET] 'MERCURY'", 25, 30, 192, 25 -I
~BUTTON "VENUS", "[PLANET] 'VENUS'", 25, 60, 192, 25 -I
~BUTTON "EARTH", "[PLANET] 'EARTH'", 25, 90, 192, 25 -I
~BUTTON "MOON ", "[PLANET] 'MOON'", 25, 120, 192, 25 -I
~BUTTON "MARS ", "[PLANET] 'MARS'", 25, 150, 192, 25 -I
~BUTTON "JUPITER", "[PLANET] 'JUPITER'", 25, 180, 192, 25 -I
~BUTTON "SATURN", "[PLANET] 'SATURN'", 25, 210, 192, 25 -I
~BUTTON "URANUS", "[PLANET] 'URANUS'", 25, 240, 192, 25 -I
~BUTTON "NEPTUNE", "[PLANET] 'NEPTUNE'", 25, 270, 192, 25 -I
~BUTTON "PLUTO", "[PLANET] 'PLUTO'", 25, 300, 192, 25 -I
~END

```



**Figure 48** ~[QPOWER]

```
~[QPOWER]
~SET_WIN 88, 0, 575, 115
~LINE_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 29, 38
Enter the internal heat load to dissipate (W)
~EDIT "[QPOWER]", 460, 15, 103, 35 -M:1
~BUTTON "OK", "", 247, 60, 75, 35 -I
~EDIT_FILL QPSCR.DAT
~END
```



**Figure 49** Surface Temperature screen

```
~[TEMP]
~SET_WIN 88, 0, 575, 105
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 29, 39
Enter the temperature of radiating surface (K)
~EDIT "[TEMP]", 450, 16, 103, 35 -M:1
~BUTTON "OK", "", 242, 60, 75, 35 -I
~EDIT_FILL TEMPSCR.DAT
~END
```

## Enclosure Properties:

Enclosure dimensions:	Solar absorptivity	Emissivity
X: <input type="text" value="1.0"/> (m)	side 1 (-Z) <input type="text" value=".05"/>	<input type="text" value=".05"/>
Y: <input type="text" value="1.0"/> (m)	side 2 (-Y) <input type="text" value=".05"/>	<input type="text" value=".05"/>
Z: <input type="text" value="1.0"/> (m)	side 3 (-X) <input type="text" value=".05"/>	<input type="text" value=".05"/>
	side 4 (+Y) <input type="text" value=".05"/>	<input type="text" value=".05"/>
	side 5 (+X) <input type="text" value=".05"/>	<input type="text" value=".05"/>
	side 6 (+Z) <input type="text" value=".05"/>	<input type="text" value=".05"/>

**Internal Surfaces are Black**

Number of decoupled sides:   
 a decoupled side is one with no net heat across it

Spacecraft Coordinates

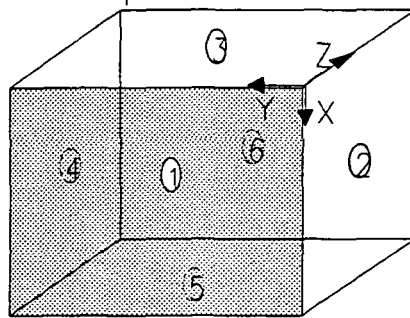


Figure 50 -[X]

```

-[X]
~SET_WIN 0, 0, 608, 685
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~TEXTSIZE LARGE
~CURSET 160, 35
Enclosure Properties
~TEXTSIZE NORMAL
~CURSET 6, 60
Enclosure dimensions:
~CURSET 21, 100
X:
~CURSET 21, 140

```

```

Y:
~CURSET 21, 180
Z:
~CURSET 142, 100
(m)
~CURSET 142, 140
(m)
~CURSET 142, 180
(m)
~EDIT "[X]", 57, 75, 75, 35 -S:S -M:1
~EDIT "[Y]", 57, 115, 75, 35 -S:S -M:1
~EDIT "[Z]", 57, 155, 75, 35 -S:S -M:1
~CURSET 288, 60
Solar absorptivity
~CURSET 206, 100
side 1 (-Z)

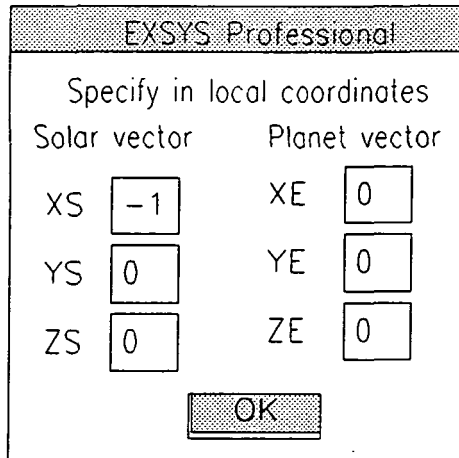
```



```

~CURSET 206, 140
side 2 (-Y)
~CURSET 206, 180
side 3 (-X)
~CURSET 206, 220
side 4 (+Y)
~CURSET 206, 260
side 5 (+X)
~CURSET 206, 300
side 6 (+Z)
~EDIT "[ALPS1]", 325, 75, 100, 35 -M:1
~EDIT "[ALPS2]", 325, 115, 100, 35 -M:1
~EDIT "[ALPS3]", 325, 155, 100, 35 -M:1
~EDIT "[ALPS4]", 325, 195, 100, 35 -M:1
~EDIT "[ALPS5]", 325, 235, 100, 35 -M:1
~EDIT "[ALPS6]", 325, 275, 100, 35 -M:1
~CURSET 477, 60
Emissivity
~EDIT "[ALPIR1]", 473, 70, 100, 36 -M:1
~EDIT "[ALPIR2]", 473, 115, 100, 35 -M:1
~EDIT "[ALPIR3]", 473, 155, 100, 35 -M:1
~EDIT "[ALPIR4]", 473, 195, 100, 35 -M:1
~EDIT "[ALPIR5]", 473, 235, 100, 35 -M:1
~EDIT "[ALPIR6]", 473, 275, 100, 35 -M:1
~CURSET 2, 395
Number of decoupled sides:
~TEXTSIZE SMALL
~CURSET 5, 427
a decoupled side is one with no net heat across
it
~EDIT "[NUMINS]", 273, 359, 47, 35 -M:1
~TEXTSIZE NORMAL
~IMAGE "SSC.PCX", 154, 478, 318, 196
~CURSET 210, 470
Spacecraft Coordinates
~LINE_COLOR WHITE
~FILL_COLOR BLACK
~TEXTSIZE LARGE
~FOREGROUND WHITE
~BACKGROUND BLACK
~CURSET 211, 347
Internal Surfaces are Black
~BUTTON "OK", "", 402, 386, 75, 35 -I
~TEXTSIZE NORMAL
~EDIT_FILL XSCR.DAT
~END

```

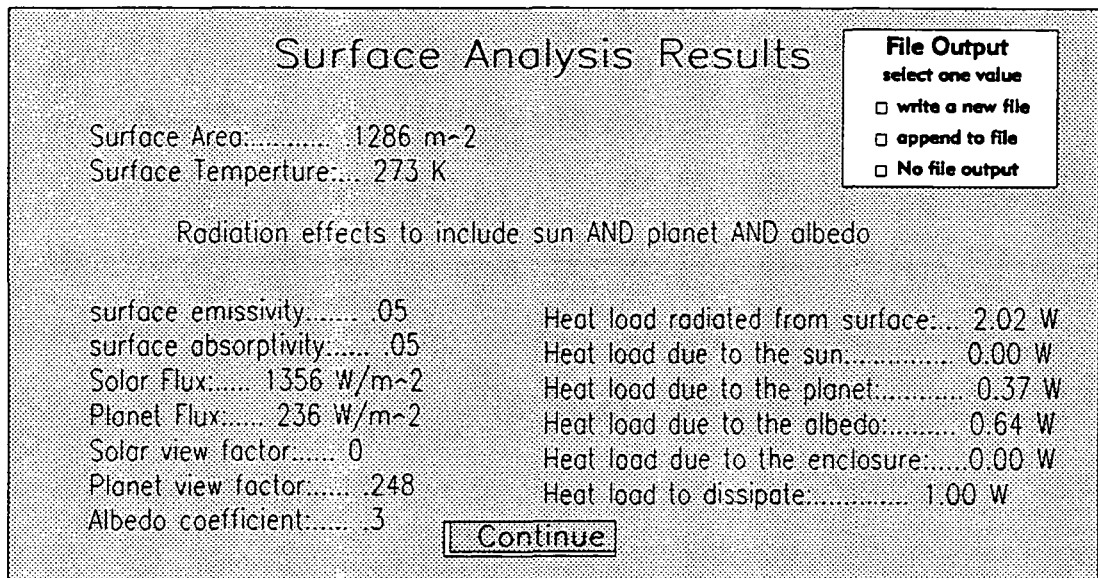


**Figure 51** ~[XS]

```

~[XS]
~SET_WIN 88, 0, 325, 244
~LINE_COLOR BLACK
~FILL_COLOR BLACK
~FOREGROUND BLACK
~BACKGROUND CYAN
~CLS
~CURSET 44, 23
Specify in local coordinates
~CURSET 18, 53
Solar Vector
~CURSET 20, 90
XS
~CURSET 20, 130
YS
~CURSET 20, 170
ZS
~EDIT "[XS]", 75, 65, 49, 35 -M:1
~EDIT "[YS]", 75, 105, 49, 35 -M:1
~EDIT "[ZS]", 75, 145, 49, 35 -M:1
~CURSET 178, 50
Planet Vector
~CURSET 175, 90
XE
~CURSET 175, 130
YE
~CURSET 175, 170
ZE
~EDIT "[XE]", 230, 65, 49, 35 -M:1
~EDIT "[YE]", 230, 105, 49, 35 -M:1
~EDIT "[ZE]", 230, 145, 49, 35 -M:1
~BUTTON "OK", "", 122, 196, 75, 35 -I
~EDIT_FILL XSSCR.DAT
~END

```

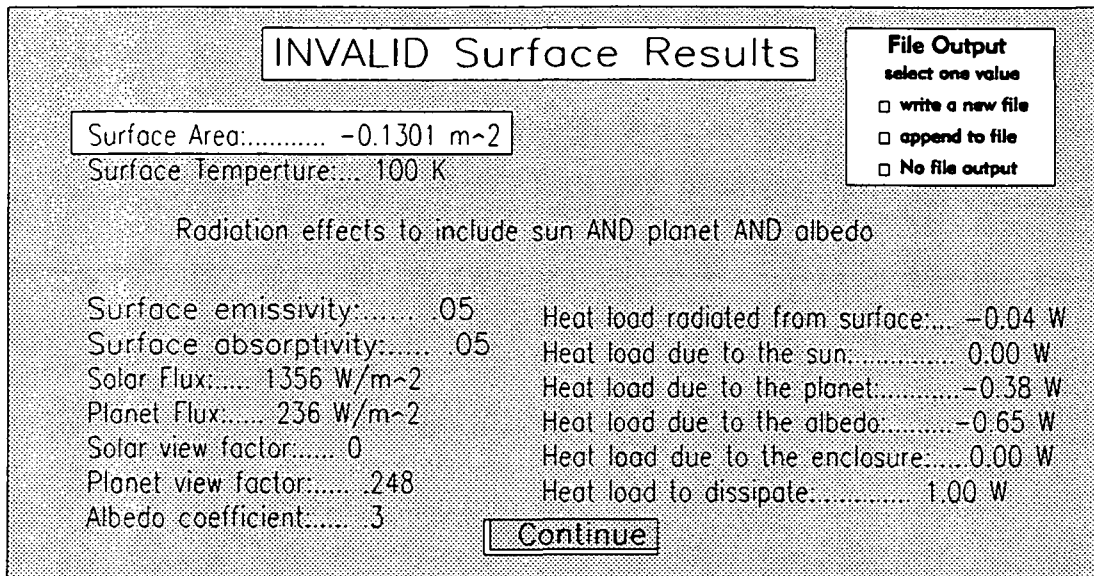


**Figure 52** ~[REPORT1]

```

~[REPORT1]
~SET_WIN 0, 0, 849, 451
~FILL_COLOR BLACK
~BACKGROUND LTGRAY
~CLS
~TEXTSIZE LARGE
~CURSET 204, 42
Surface Analysis Results
~TEXTSIZE NORMAL
~CURSET 9, 89
Surface Area:..... [[area]8.4] m^2
Surface Temperature:... [[temp]7.2] K
~CURSET 59, 180
[[*Q 2]]
~CURSET 350, 220
Heat load radiated from surface:..... [[qrad]5.2] W
Heat load due to the Sun:..... [[qsun]5.2] W
Heat load due to the Planet:..... [[qplanet]5.2] W
Heat load due to the Planet's albedo:.... [[qalbedo]5.2] W
Heat load due to the enclosure:..... [[qencl]5.2] W
Heat load to the dissipate:..... [[qpower]5.2] W
~CURSET 7, 220
surface emissivity:..... [[epsilon]]
surface absorptivity:... [[alpha]]
Solar Flux:..... [[S]] W/m^2
Planet Flux:..... [[E]] W/m^2
Solar view factor:..... [[mu]]
Planet view factor:..... [[fe]]
Albedo coefficient:..... [[alb]]
~FILL_COLOR WHITE
  
```

```
~BACKGROUND WHITE
~RECT 610, 5, 200, 155
~CURSET 650, 26
  File Output
~TEXTSIZE SMALL
~CURSET 645, 48
Select one value
~BUTTON "write a new file", "q19 2", 635, 60, 148, 30 -C
~BUTTON "append to file", "q19 1", 635, 90, 150, 30 -C
~BUTTON "no file output", "q19 3", 635, 120, 150, 30 -C
~BUTTON "Continue", "[report1] = 1", 365, 393, 100, 35 -I
~END
```



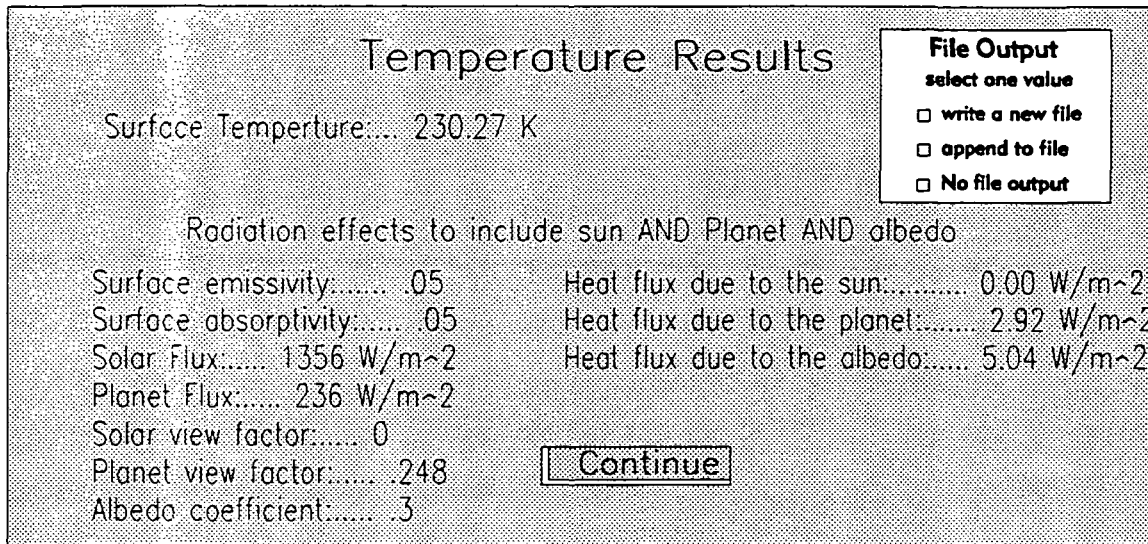
**Figure 53** ~[REPORT2]

```

~[REPORT2]
~SET_WIN 0, 0, 850, 448
~FILL_COLOR BLACK
~BACKGROUND LTGRAY
~CLS
~FILL_COLOR RED
~TEXTSIZE LARGE
~BACKGROUND RED
~CURSET 189, 44
INVALID Surface Results
~FILL_COLOR BLACK
~TEXTSIZE NORMAL
~CURSET 9, 89
Surface Area:..... [[area]] m^2
~BACKGROUND LTGRAY
Surface Temperature:... [[temp]] K
~CURSET 59, 187
[[*Q 2]]
~CURSET 374, 220
Heat load radiated from surface:..... [[qrad]5.2] W
Heat load due to the Sun:..... [[qsun]5.2] W
Heat load due to the Planet:..... [[qplanet]5.2] W
Heat load due to the Planet's albedo:.... [[qalbedo]5.2] W
Heat load due to the enclosure:..... [[qencl]5.2] W
Heat load to the dissipate:..... [[qpower]5.2] W
~CURSET 7, 220
surface emissivity:..... [[epsilon]]
surface absorptivity:... [[alpha]]
Solar Flux:..... [[S]] W/m^2
Planet Flux:..... [[E]] W/m^2

```

Solar view factor:..... [[mu]]  
Planet view factor:..... [[fe]]  
Albedo coefficient:..... [[alb]]  
~FILL\_COLOR WHITE  
~BACKGROUND WHITE  
~RECT 609, 14, 200, 155  
~CURSET 658, 35  
File Output  
~TEXTSIZE SMALL  
~CURSET 652, 57  
Select one value  
~BUTTON "write a new file", "q19 2", 635, 60, 148, 30 -C  
~BUTTON "append to file", "q19 1", 635, 90, 150, 30 -C  
~BUTTON "no file output", "q19 3", 635, 120, 150, 30 -C  
~BUTTON "Continue", "[report2] = 1", 311, 391, 100, 35 -I  
~END



**Figure 54** ~[REPORT3]

```

~[REPORT3]
~SET_WIN 0, 0, 868, 454
~FILL_COLOR BLACK
~BACKGROUND LTGRAY
~CLS
~TEXTSIZE LARGE
~CURSET 203, 51
Temperature Results
~TEXTSIZE NORMAL
~CURSET 8, 90
Surface Temperature: [[temp]] K
~CURSET 58, 129
[[*q 2]]
~CURSET 3, 210
Surface emissivity:.... [[epsilon]]
Surface absorptivity... [[alpha]]
Solar Flux:..... [[S]] W/m^2
Planet Flux:..... [[E]] W/m^2
Solar view factor:..... [[mu]]
Planet view factor:..... [[fe]]
Albedo Coefficient:.... [[alb]]
~CURSET 368, 210
Heat flux due to the Sun:..... [[xsun]5.2] W/m^2
Heat flux due to the Planet:..... [[xplanet]5.2] W/m^2
Heat flux due to the Planet's albedo:.... [[xalbedo]5.2] W/m^2
~FILL_COLOR WHITE
~BACKGROUND WHITE
~RECT 609, 14, 200, 155
~CURSET 658, 35
File Output
~TEXTSIZE SMALL

```

**~CURSET 654, 55**

**Select one value**

**~BUTTON "write a new file", "q19 2", 635, 60, 150, 30 -C**

**~BUTTON "append to file", "q19 1", 635, 90, 150, 30 -C**

**~BUTTON "No file output", "q19 3", 635, 120, 150, 30 -C**

**~BUTTON "Continue", "[report3] = 1", 330, 376, 100, 35 -I**

**~END**



## Enclosure Analysis Results

Surface Temperatures		Net Surface Heat Loads		Orbit Information	
side 1:	295.67 K	side 1:	-12.43 W	Inclination:	28.5 deg
side 2:	295.67 K	side 2:	-12.43 W	Longitude of Ascending node:	0 deg
side 3:	303.36 K	side 3:	43.78 W	Argument of Periapsis:	0 deg
side 4:	295.67 K	side 4:	-12.43 W	True Anomaly at Epoch:	0 deg
side 5:	298.25 K	side 5:	5.96 W	Semi-major axis:	6578 km
side 6:	295.67 K	side 6:	-12.43 W	Eccentricity:	0
				The spacecraft orbit is LEO (Low Earth Orbit)	
				The spacecraft orientation is planet	
External Surface Properties		View Factors		<input type="button" value="Continue"/>	
emissivity	absorptivity	Solar	Planet	<b>File Output</b> select one value <input type="checkbox"/> writes a new file <input type="checkbox"/> append to file <input type="checkbox"/> No file output	
side 1: .05	side 1: .05	side 1: 0	side 1: .28		
side 2: .05	side 2: .05	side 2: 0	side 2: .28		
side 3: .05	side 3: .05	side 3: 1	side 3: 0		
side 4: .05	side 4: .05	side 4: 0	side 4: .28		
side 5: .05	side 5: .05	side 5: 0	side 5: .88		
side 6: .05	side 6: .05	side 6: 0	side 6: .28		

**Figure 55** ~[REPORT4]

```

~[REPORT4]
~SET_WIN 0, 0, 1006, 550
~FILL_COLOR BLACK
~FOREGROUND WHITE
~BACKGROUND LTGRAY
~CLS
~TEXTSIZE LARGE
~FOREGROUND BLACK
~CURSET 270, 39
Enclosure Analysis Results
~TEXTSIZE NORMAL
~CURSET 19, 80
Surface Temperatures
~CURSET 39, 100
side 1: [[T1]6.2] K
side 2: [[T2]6.2] K
side 3: [[T3]6.2] K
side 4: [[T4]6.2] K
side 5: [[T5]6.2] K
side 6: [[T6]6.2] K
~CURSET 279, 80
Net Surface Heat loads
~CURSET 299, 100
side 1: [[Q1]6.2] W
side 2: [[Q2]6.2] W
side 3: [[Q3]6.2] W
side 4: [[Q4]6.2] W
    
```

side 5: [[Q5]6.2] W  
 side 6: [[Q6]6.2] W  
 ~CURSET 85, 325  
 External Surface Properties  
 ~CURSET 55, 350  
 emissivity  
 ~CURSET 25, 385  
 side 1: [[ALPIR1]]  
 side 2: [[ALPIR2]]  
 side 3: [[ALPIR3]]  
 side 4: [[ALPIR4]]  
 side 5: [[ALPIR5]]  
 side 6: [[ALPIR6]]  
 ~CURSET 244, 350  
 absorptivity  
 ~CURSET 216, 385  
 side 1: [[ALPS1]]  
 side 2: [[ALPS2]]  
 side 3: [[ALPS3]]  
 side 4: [[ALPS4]]  
 side 5: [[ALPS5]]  
 side 6: [[ALPS6]]  
 ~CURSET 610, 80  
 Orbit Information  
 ~CURSET 559, 100  
 Inclination: [[INC]] deg  
 Longitude of Ascending node: [[OMEGA]] deg  
 Argument of Periapsis: [[WOMEGA]] deg  
 True Anomaly at Epoch: [[v0]] deg  
 Semi-major axis: [[SMA]] km  
 Eccentricity: [[ECC]]  
 [[\*q 5]]  
 [[\*q 4]]  
 ~CURSET 483, 326  
 View Factors  
 ~CURSET 432, 352  
 Solar  
 ~CURSET 387, 385  
 side 1: [[MU1]]  
 side 2: [[MU2]]  
 side 3: [[MU3]]  
 side 4: [[MU4]]  
 side 5: [[MU5]]  
 side 6: [[MU6]]  
 ~CURSET 593, 353  
 Planet  
 ~CURSET 551, 384  
 side 1: [[FE1]6.2]  
 side 2: [[FE2]6.2]  
 side 3: [[FE3]6.2]  
 side 4: [[FE4]6.2]  
 side 5: [[FE5]6.2]

side 6: [[FE6]6.2]

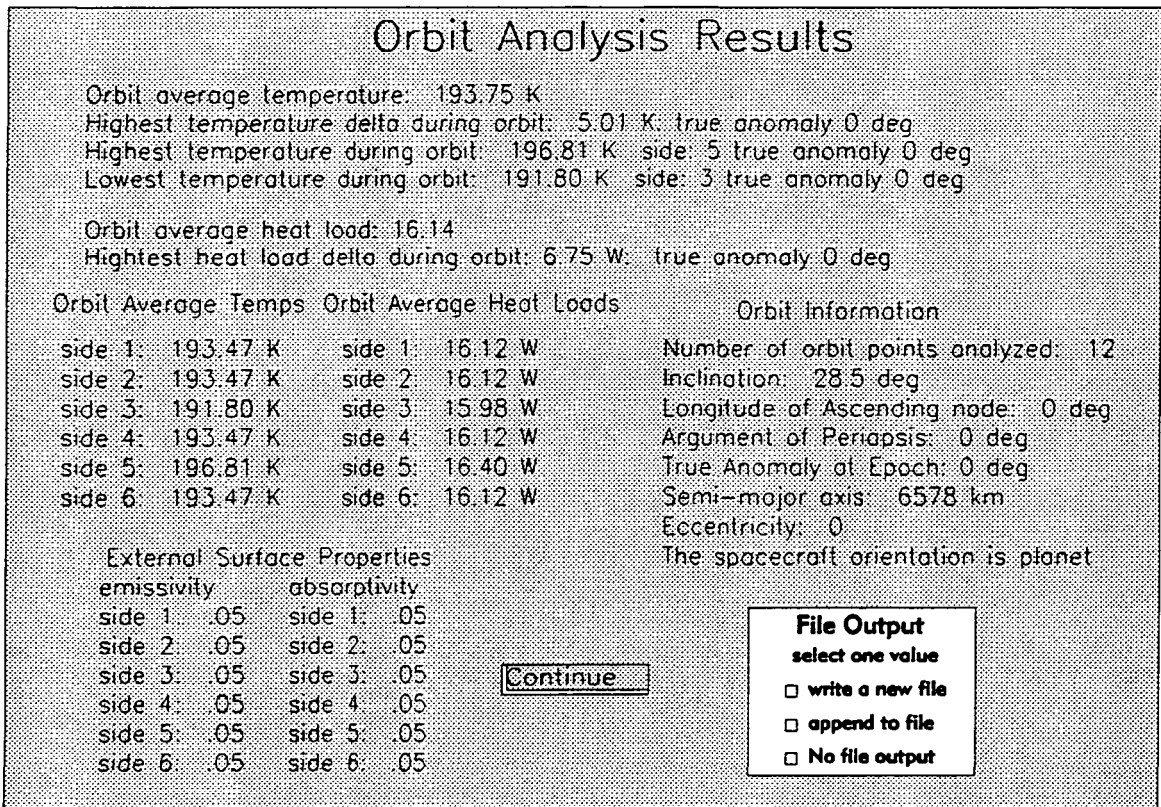
~FILL\_COLOR WHITE  
~BACKGROUND WHITE  
~RECT 791, 379, 200, 155  
~CURSET 842, 402

File Output

~TEXTSIZE SMALL  
~CURSET 838, 421

Select one value

~BUTTON "write a new file", "q19 2", 816, 433, 150, 30 -C  
~BUTTON "append to file", "q19 1", 816, 463, 150, 30 -C  
~BUTTON "No file output", "q19 3", 816, 493, 150, 30 -C  
~BUTTON "Continue", "[REPORT4] = 1", 828, 308, 106, 35 -I  
~END



**Figure 56** ~[REPORT5]

```

~[REPORT5]
~SET_WIN 0, 0, 1006, 705
~FOREGROUND WHITE
~BACKGROUND LTGRAY
~CLS
~TEXTSIZE LARGE
~FOREGROUND BLACK
~CURSET 283, 44
Orbit Analysis Results
~TEXTSIZE NORMAL
~CURSET 42, 84
Orbit average temperature: [[tavet]6.2] K
Highest temperature delta during orbit: [[tdelta]6.2] K : true anomaly [[v0delt]] deg
Highest temperature during orbit: [[thigh]6.2] K side: [[shigh]] true anomaly: [[v0high]] deg
Lowest temperature during orbit: [[tlow]6.2] K side: [[slow]] true anomaly: [[v0low]] deg
~CURSET 42, 200
Orbit average heat load: [[qavet]6.2] W
Highest heat load delta during orbit: [[qdelta]6.2] W : true anomaly [[v0qdel]] deg
~CURSET 10, 270
Orbit Average Temps
~CURSET 30, 300
side 1: [[T1]6.2] K
side 2: [[T2]6.2] K

```

side 3: [[T3]6.2] K  
 side 4: [[T4]6.2] K  
 side 5: [[T5]6.2] K  
 side 6: [[T6]6.2] K  
 ~CURSET 254, 270  
 Orbit Average Heat loads  
 ~CURSET 262, 301  
 side 1: [[Q1]6.2] W  
 side 2: [[Q2]6.2] W  
 side 3: [[Q3]6.2] W  
 side 4: [[Q4]6.2] W  
 side 5: [[Q5]6.2] W  
 side 6: [[Q6]6.2] W  
 ~CURSET 102, 472  
 External Surface Properties  
 ~CURSET 51, 497  
 emissivity  
 ~CURSET 33, 530  
 side 1: [[ALPIR1]]  
 side 2: [[ALPIR2]]  
 side 3: [[ALPIR3]]  
 side 4: [[ALPIR4]]  
 side 5: [[ALPIR5]]  
 side 6: [[ALPIR6]]  
 ~CURSET 296, 497  
 absorptivity  
 ~CURSET 265, 530  
 side 1: [[ALPS1]]  
 side 2: [[ALPS2]]  
 side 3: [[ALPS3]]  
 side 4: [[ALPS4]]  
 side 5: [[ALPS5]]  
 side 6: [[ALPS6]]  
 ~CURSET 558, 270  
 Orbit Information  
 ~CURSET 497, 300  
 Number of orbit points analyzed: [[numpts]]  
 Inclination: [[INC]] deg  
 Longitude of Ascending node: [[OMEGA]] deg  
 Argument of Periapsis: [[WOMEGA]] deg  
 True Anomaly at Epoch: [[v0]] deg  
 Semi-major axis: [[SMA]] km  
 Eccentricity: [[ECC]]  
 [[\*Q4]]  
 ~BACKGROUND WHITE  
 ~RECT 721, 521, 200, 155  
 ~CURSET 773, 547  
 File Output  
 ~TEXTSIZE SMALL  
 ~CURSET 770, 566  
 Select one value  
 ~BUTTON "write a new file", "q19 2", 750, 570, 150, 30 -C

Screens

C43

```
~BUTTON "append to file", "q19 1", 750, 600, 150, 30 -C  
~BUTTON "No file output", "q19 3", 750, 630, 150, 30 -C  
~BUTTON "Continue", "[report5] = 1", 522, 557, 106, 35 -I  
~END
```

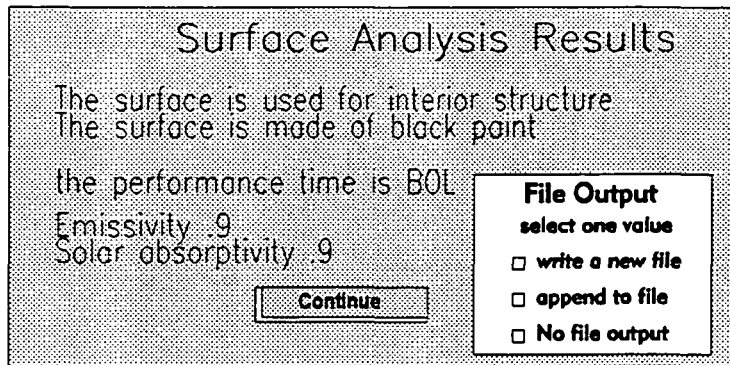


Figure 57 ~[REPORT6]

```

-[REPORT6]
~SET_WIN 0, 0, 651, 292
~BACKGROUND LTGRAY
~CLS
~CURSET 160, 29
Surface Analysis Results
~CURSET 50, 65
The surface is used for [[APPLICATION]]
The surface is made of [[MATERIAL]]
~CURSET 53, 119
[[*Q9]]
~CURSET 50, 190
Emissivity [[EPSILON]]
Solar absorptivity [[ALPHA]]
~BACKGROUND WHITE
~RECT 413, 138, 200, 144
~CURSET 466, 159
File Output
~TEXTSIZE SMALL
~CURSET 458, 175
Select one value
~BUTTON "write a new file", "q21 2", 445, 180, 150, 30 -C
~BUTTON "append to file", "q21 1", 445, 211, 150, 30 -C
~BUTTON "No file output", "q213", 444, 243, 150, 30 -C
~BUTTON "Continue", "[REPORT6] = 1", 252, 244, 106, 35 -I
~END

```

**THIS PAGE INTENTIONALLY LEFT BLANK**



**Appendix D  
Default Data Files  
and  
Value Definitions**

D1

**Default Data Files**

**PRECEDING PAGE BLANK NOT FILMED**

**THIS PAGE INTENTIONALLY LEFT BLANK**

### Screen Default Data Files

Q16	ALPSCR.DAT
Q17	ALBSCR.DAT
[AINST]	AINSTSCR.DAT
[AREA]	AREASCR.DAT
[ECC]	ECCSCR.DAT
[GS1]	GSSCR.DAT
[INC]	INCSCR.DAT
[NUMPTS]	PTSSCR.DAT
[QENCL]	QENCLSCR.DAT
[QPOWER]	QPSCR.DAT
[TEMP]	TEMPSCR.DAT
[X]	XSCR.DAT
[XS]	XSSCR.DAT

AINSTSCR.DAT

1 /\* Surface area of the enclosure  
instrument  
1 /\*emissivity of instrument surface  
inside the enclosure

ALBSCR.DAT

.3 /\* albedo coefficient  
1356 /\* solar flux W/m<sup>2</sup>  
236 /\* planet flux W/m<sup>2</sup>  
6378 /\* radius of the planet km

ALPSCR.DAT

0 /\* solar view factor  
.287 /\* planet view factor  
.05 /\* emissivity of surface  
.05 /\* solar absorptivity of surface

AREASCR.DAT

1 /\* area of surface m<sup>2</sup>

ECCSCR.DAT

0 /\* true anomaly at epoch deg  
6785.0 /\* semi-major axis km  
0 /\* eccentricity

GSSCR.DAT

-1 /\* x component of the global sun  
vector  
0 /\* y component of the global sun  
vector  
0 /\* z component of the global sun  
vector

INCSER.DAT

28.5 /\* inclination deg  
0 /\* line of nodes deg  
0 /\* line of periapsis deg  
0 /\* true anomaly at epoch deg  
6785.0 /\* semi-major axis km  
0 /\* eccentricity

PTSSCR.DAT

12 /\* number of points in the orbit

QENCLSCR.DAT

0 /\* enclosure radiation to dissipate

QPSCR.DAT

1.0 /\* internal power to dissipate W

TEMPSCR.DAT

273 /\* temperature of the surface

XSCR.DAT

1.0 /\* dimension along the x axis  
1.0 /\* dimension along the y axis  
1.0 /\* dimension along the z axis  
.05 /\* solar absorptivity for each surface

.05

.05

.05

.05

.05

.05 /\* emissivity for each surface

.05

.05

.05

.05

.05

0 /\* number of decoupled sides

XSSCR.DAT

-1 /\* x component of the local sun vector  
0 /\* y component of the local sun vector  
0 /\* z component of the local sun vector  
0 /\* x component of the local planet  
vector  
0 /\* y component of the local planet  
vector  
0 /\* z component of the local planet  
vector

**Appendix E**  
**Report Generator Listing**

**THIS PAGE INTENTIONALLY LEFT BLANK**

```

Q19 3 /G:END

Q1 1 /G:REPORT1
Q1 2 /G:REPORT2
Q1 3 /G:REPORT3
Q1 4 /G:REPORT4
Q1 5 /G:REPORT5

:REPORT1
Q19 1 /G:APAREA
Q19 2 /G:NWAREA

:APAREA
FILE AREA.RPT /A
GOTO RPT1

: NWAREA
FILE AREA.RPT
GOTO RPT1

:RPT1
"Surface Analysis Results"
[AREA] /F6.2
[TEMP] /F6.2
Q2
[QRAD] /F6.2
[QSUN] /F6.2
[QPLANET] /F6.2
[QALBEDO] /F6.2
[QENCL] /F6.2
[QPOWER] /F6.2
[EPSILON]
[ALPHA]
[S]
[E]
[MU]
[FE]
[ALB]

```

GOTO END

```

:REPORT2
Q19 1 /G:APTEMP
Q19 2 /G:NWTEMP

```

```

:APTEMP
FILE TEMP.RPT /A
GOTO RPT2

```

```

: NWTEMP
FILE TEMP.RPT
GOTO RPT2

```

Report Generator

```

:RPT2
Q12 1 /G:SURTEMP
Q12 2 /G:ENVTEMP
:SURTEMP
"Surface Analysis Results"
[AREA] /F6.2
[TEMP] /F6.2
Q2
[QRAD] /F6.2
[QSUN] /F6.2
[QPLANET] /F6.2
[QALBEDO] /F6.2
[QENCL] /F6.2
[QPOWER] /F6.2
[EPSILON]
[ALPHA]
[S]
[E]
[MU]
[FE]
[ALB]

```

GOTO END

```

:ENVTEMP
"Temperature Analysis Results"
[TEMP] /F6.2
Q 2
[EPSILON]
[ALPHA]
[S] /F6.2
[E] /F6.2
[MU]
[FE]
[ALB]
[XSUN] /F6.2
[XPLANET] /F6.2
[XALBEDO] /F6.2

```

GOTO END

```

:REPORT3
Q19 1 /G:APENCL
Q19 2 /G:NWENCL

```

```

:APENCL
FILE ENCL.RPT /A
GOTO RPT3

```

```

: NWENCL
FILE ENCL.RPT
GOTO RPT3

```

:RPT3  
"Enclosure Analysis Results"

"Surface Temperatures"

[T1] /f6.2  
[T2] /f6.2  
[T3] /f6.2  
[T4] /f6.2  
[T5] /f6.2  
[T6] /f6.2

"Net Surface Heat loads"

[Q1] /f6.2  
[Q2] /f6.2  
[Q3] /f6.2  
[Q4] /f6.2  
[Q5] /f6.2  
[Q6] /f6.2

"External Surface Properties"

"emissivity"

[ALPIR1]  
[ALPIR2]  
[ALPIR3]  
[ALPIR4]  
[ALPIR5]  
[ALPIR6]

"absorptivity"

[ALPS1]  
[ALPS2]  
[ALPS3]  
[ALPS4]  
[ALPS5]  
[ALPS6]

"Orbit Information"

[INC]  
[OMEGA]  
[WOMEGA]  
[v0]  
[SMA]  
[ECC]  
q5  
q4

"View Factors"

"Solar"

[MU1]  
[MU2]  
[MU3]  
[MU4]  
[MU5]  
[MU6]

"Planet"

[FE1]  
[FE2]  
[FE3]  
[FE4]  
[FE5]  
[FE6]

GOTO END

:REPORT4  
Q19 1 /G:APORBIT  
Q19 2 /G:NWORBIT

:APORBIT  
FILE ORBIT.RPT /A  
GOTO RPT4

: NWORBIT  
FILE ORBIT.RPT  
GOTO RPT4

:RPT4  
"Orbit Analysis Results"

[TAVET] /F6.2  
[TDELTA] /F6.2  
[V0DELTA]  
[THIGH] /F6.2  
[SHIGH]  
[V0HIGH]  
[TLOW] /F6.2  
[SLOW]  
[V0LOW]

[QAVET] /F6.2  
[QDELTA] /F6.2  
[V0QDEL]

"Surface Temperatures"



[T1] /f6.2  
[T2] /f6.2  
[T3] /f6.2  
[T4] /f6.2  
[T5] /f6.2  
[T6] /f6.2

"Net Surface Heat loads"

[Q1] /f6.2  
[Q2] /f6.2  
[Q3] /f6.2  
[Q4] /f6.2  
[Q5] /f6.2  
[Q6] /f6.2

"External Surface Properties"

"emissivity"

[ALPIR1]  
[ALPIR2]  
[ALPIR3]  
[ALPIR4]  
[ALPIR5]  
[ALPIR6]

"absorptivity"

[ALPS1]  
[ALPS2]  
[ALPS3]  
[ALPS4]  
[ALPS5]  
[ALPS6]

"Orbit Information"

[INC]  
[OMEGA]  
[WOMEGA]  
[v0]  
[SMA]  
[ECC]  
Q5  
Q4

"View Factors"

"Solar"

[MU1]

[MU2]  
[MU3]  
[MU4]  
[MU5]  
[MU6]

"Planet"

[FE1]  
[FE2]  
[FE3]  
[FE4]  
[FE5]  
[FE6]

"Orbit Average Temps"  
"Orbit Information"

[NUMPTS]  
[INC]  
[OMEGA]  
[WOMEGA]  
[v0]  
[SMA]  
[ECC]  
Q4

GOTO END

:REPORT5  
Q19 1 /G:APSURF  
Q19 2 /G:NWSURF

:APSURF  
FILE SURF.RPT /A  
GOTO RPT5

: NWSURF  
FILE SURF.RPT  
GOTO RPT5

:RPT5  
"Surface Analysis Results"

[APPLICATION]  
[MATERIAL]

Q9

[EPSILON]  
[ALPHA]  
:END

## AREA.RPT

### Surface Analysis Results

area of radiator (m<sup>2</sup>) = 0.06

temperature of the radiator (K) = 273.00

Radiation effects to include internal power only

heat radiated from the surface (W) = 1.00

heat load on surface due to sun (W) = 0.00

heat load on surface due to planet (W) = 0.00

heat load on the surface due to albedo (W) = 0.00

Enclosure heat to be radiated (W) = 0.00

heat load to be dissipated (W) = 1.00

emissivity of the radiating surface = .05

absorptivity of the radiating surface = .05

solar flux W/m<sup>2</sup> = 1356

planet flux W/m<sup>2</sup> = 236

solar angle (= cos(theta) where theta is the angle (in rad) between the sun vector and the plane normal) of the radiating surface. ~Between 0 and 1 = 0

view factor to the planet of the radiating surface = .248

albedo coefficient = .3

## TEMP.RPT

### Temperature Analysis Results

temperature of the radiator (K) = 238.83

Radiation effects to include sun AND planet AND albedo

emissivity of the radiating surface = .05

absorptivity of the radiating surface = .05

solar flux  $W/m^2 = 1356.00$

planet flux  $W/m^2 = 236.00$

solar angle (=  $\cos(\theta)$  where  $\theta$  is the angle (in rad) between the sun vector and the plane normal) of the radiating surface. ~Between 0 and 1 = 0

view factor to the planet of the radiating surface = .287

albedo coefficient = .3

heat flux on surface due to sun ( $W/m^2$ ) = 0.00

heat flux on surface due to radiation of planet ( $W/m^2$ ) = 3.39

heat flux on surface due to albedo ( $W/m^2$ ) = 5.84

## ENCL.RPT

### Enclosure Analysis Results

#### Surface Temperatures

Temperature of side 1 (K) = 295.67

Temperature of side 2 (K) = 295.67

Temperature of side 3 (K) = 303.36

Temperature of side 4 (K) = 295.67

Temperature of side 5 (K) = 298.25

Temperature of side 6 (K) = 295.67

#### Net Surface Heat loads

Net heat on side 1 = -12.44

Net heat on side 2 = -12.44

Net heat on side 3 = 43.79

Net heat on side 4 = -12.44

Net heat on side 5 = 5.97

Net heat on side 6 = -12.44

#### External Surface Properties

##### emissivity

Infrared absorptivity of side 1 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 2 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 3 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 4 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 5 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 6 (assuming gray bodies = emissivity) = .05

##### absorptivity

Solar absorptivity of side 1 = .05

Solar absorptivity of side 2 = .05

Solar absorptivity of side 3 = .05

Solar absorptivity of side 4 = .05

Solar absorptivity of side 5 = .05

Solar absorptivity of side 6 = .05

#### Orbit Information

orbit inclination (deg) = 0

longitude of ascending node (deg) = 0

satellite position (deg) = 0

true anomaly at epoch = 0

Semi-major axis (km) = 6785

eccentricity of orbit = 0

The spacecraft orbit is LEO (Low Earth Orbit)

The spacecraft orientation is inertial

#### View Factors

##### Solar

Solar view factor for side 1 = 0

solar view factor for side 2 = 0

solar view factor for side 3 = 1

solar view factor for side 4 = 0

Solar view factor for side 5 = 0

solar view factor for side 6 = 0

##### Planet

Planet view factor for side 1 = .287116

planet view factor for side 2 = .287116  
Planet view factor for side 3 = 0  
Planet view factor for side 4 = .287116  
Planet view factor for side 5 = .883628  
Planet view factor for side 6 = .287116

## ORBIT.RPT

### Orbit Analysis Results

the average temperature in the enclosure over the entire orbit (K) = 195.94

Highest temperature delta (K) = 5.01

position with the highest temperature delta = -4.712389

the highest temperature value (K) = 199.96

the side with the highest temperature value = 4

The position with the highest temperature value = -5.235988

Lowest temperature in orbit analysis (K) = 191.80

side with lowest temperature = 3

Position of the lowest temperature = 0

the average heat load in the enclosure over the entire orbit (W) = 16.33

Highest heat load delta (W) = 7.04

position with the highest heat load delta = -5.759587

### Surface Temperatures

Temperature of side 1 (K) = 195.59

Temperature of side 2 (K) = 196.11

Temperature of side 3 (K) = 196.11

Temperature of side 4 (K) = 196.11

Temperature of side 5 (K) = 196.11

Temperature of side 6 (K) = 195.59

### Net Surface Heat loads

Net heat on side 1 = 16.30

Net heat on side 2 = 16.34

Net heat on side 3 = 16.34

Net heat on side 4 = 16.34

Net heat on side 5 = 16.34

Net heat on side 6 = 16.30

### External Surface Properties

#### emissivity

Infrared absorptivity of side 1 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 2 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 3 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 4 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 5 (assuming gray bodies = emissivity) = .05

Infrared absorptivity of side 6 (assuming gray bodies = emissivity) = .05

#### absorptivity

Solar absorptivity of side 1 = .05

Solar absorptivity of side 2 = .05

Solar absorptivity of side 3 = .05

Solar absorptivity of side 4 = .05

Solar absorptivity of side 5 = .05

Solar absorptivity of side 6 = .05

### Orbit Information

orbit inclination (deg) = 28.5

longitude of ascending node (deg) = 0

satellite position (deg) = 0

true anomaly at epoch = 0

Semi-major axis (km) = 6785

eccentricity of orbit = 0

The spacecraft orientation is inertial

View Factors

Solar

Planet

Planet view factor for side 1 = .25

planet view factor for side 2 = .25

Planet view factor for side 3 = 0

Planet view factor for side 4 = .25

Planet view factor for side 5 = .84

Planet view factor for side 6 = .25

Orbit Average Temps

Orbit Information

Number of orbital points to be analyzed = 12

orbit inclination (deg) = 28.5

longitude of ascending node (deg) = 0

satellite position (deg) = 0

true anomaly at epoch = 0

Semi-major axis (km) = 6785

eccentricity of orbit = 0

The spacecraft orientation is inertial

## **SURF.RPT**

### **Surface Analysis Results**

**Surface APPLICATION = antenna reflector**

**Surface material = white paint**

**the performance time is BOL (beginning of life)**

**emissivity of the radiating surface = .05**

**absorptivity of the radiating surface = .05**



**Appendix F**  
**Fortran Program List**

THIS PAGE INTENTIONALLY LEFT BLANK

The following information is organize to create quick references for the designer.

The first section contains a list of the source code routines and a biref description on each. The second section contains a list of fortran files and the routines in each. The third section contians a list of executables and the files that must be linked to create them.

#### SUBROUTINE AREA

Calculates the areas of each side. The side axes are defined.

#### SUBROUTINE CALCEB

Calculates the blackbody radiation for each surface

#### SUBROUTINE CALCQ

Calculates the heat load on each surface

#### SUBROUTINE CALCT

Calculates the temperatures given EB

#### SUBROUTINE CALCVWF

Calculates view factors for the internal enclosure

#### SUBROUTINE CALFEB

Calculates the FEB matrix  $FEB = \text{view factors} * \text{areas}$ . It is the coefficient matrix to the blackbody radiation terms, EB (see documentation)

#### SUBROUTINE CALFEI

Calculates the earth view factor for inertial orientation

#### SUBROUTINE CALFEP

Calculates the earth view factor for planet orientation

#### SUBROUTINE CALFLUX

Calculates the environmental fluxes. Fills the X matrix and the XRAD matrix

#### SUBROUTINE CALSOLR

Calculates the dot product of the solar vector and the unit normals of each surface

#### SUBROUTINE CALVFNOR

Calculates the view factor from a plane element to a sphere. Used as the earth view factor normal from "thermal radiation heat transfer" by Siegel and Howell: APPENDIX C

#### SUBROUTINE CALVFTAN

Calculates the view factor from a plane element to a sphere. Used as the earth view factor tangential from "Thermal Radiation Heat Transfer" by Siegel and Howell: APPENDIX C

#### FUNCTION DOTPROD

Calculates the dot product of two vectors x and y

#### SUBROUTINE EAREAD

Reads the input for the enclosure analysis

#### SUBROUTINE EARWRIT

Writes planet view factors for each side, and the normal and tangential components to "RETURN.DAT"

**SUBROUTINE ENCINIT**

Initialize the constants and sets the matrices to 0.0

**SUBROUTINE ENCLREAD**

Reads the input for the enclosure analysis

**SUBROUTINE ENCWRIT**

Writes temperature and heat load to "RETURN.DAT"

**SUBROUTINE FERREAD**

Reads the input for the planet view factor analysis

**PROGRAM FETANOR**

Calculates the earth view factors normal and tangential plates.

**SUBROUTINE FEWRIT**

Writes normal and tangential view factors to "RETURN.DAT"

**SUBROUTINE IJK2PQR**

\*\*\*\* THIS SUBROUTINE IS CURRENTLY NOT USED, BUT LEFT FOR FUTURE REFERENCE \*\*\*\*  
Fills the coefficient matrix this conversion was taken from "Fundamentals of Astrodynamics" by Bate, Mueller and White

**SUBROUTINE MATADD**

Adds two matrices all three matrices are dimension nxm  
if IFAC =-1 then subtraction if IFAC = 1 then addition

**SUBROUTINE MATMUL**

Multiplies two matrices a is dimension N,M B is dimension M,L C is dimension N,L

**SUBROUTINE ORBCOMP**

Calculates the orbital average values for temperature and heat loads.

**SUBROUTINE ORBFIN**

Finishes the orbital calculations at the end of the total orbit points.

**SUBROUTINE ORBINIT**

Initialize the constants and sets the matrices to 0.0 For the orbital analysis

**PROGRAM ORBITAL**

This program solves for the temperature and heat loads on an enclosure at distinct orbit points as specified by the user. For inertially or planet oriented orbits.

**SUBROUTINE ORBOUT**

Writes temperatures and heat load to "RAD-ORB.DAT"

**SUBROUTINE ORBREAD**

Reads the input for the orbital analysis

**SUBROUTINE ORBWRIT**

Writes all the final statistical analysis to "RETURN.DAT"

#### SUBROUTINE ORBPROPS

Calculates the orbital radius taken from "Fundamentals of Astrodynamics" by Bate, Mueller and White

#### SUBROUTINE POSGLOB

Calculates the spacecraft position in global coordinates

#### SUBROUTINE PQR2IJK

Fills the coefficient matrix for the rotation from perifocal system (P,Q,R) to the global system (I,J,K) see documentation for derivation

#### PROGRAM QENCL

Solves for the temperature and heat loads on an enclosure.

#### SUBROUTINE SHADOW

Determines if the spacecraft is in the shadow of the earth if in the shadows: returns the solar flux as 0.0 And the solar view factors as 0.0 If not returns the original value for solar flux

#### PROGRAM SOLARV

Solves for the dot product of the solar vector and the local unit normals on each side of the enclosure.

#### SUBROUTINE SOLREAD

Reads the input for the solar view factor analysis

#### SUBROUTINE SOLWRIT

Writes blackbody radiation, heat load and view factors to "RETURN.DAT"

#### SUBROUTINE SUNREAD

Reads the input for the sun vector analysis

#### PROGRAM SUNVECT

Calculates the sun vector in local coordinates, given inclination, longitude of ascending node and spacecraft position.

#### SUBROUTINE SUNWRIT

Writes the three components of the sun vector in local coordinates and the 6 solar view factors

#### SUBROUTINE CALSUN

Rotates the global sun vector into local coordinates solving the equation  $ax = b$ . A is the coefficient matrix from perifocal to global, b is the coefficient matrix from perifocal to spacecraft x is the local sun vector g is the global sun vector

#### SUBROUTINE SURFNOR

Calculates the dot product of x into the local surface normals as defined in documentation x is a vector in local coordinates y is the dot product with the surface normals

#### FUNCTION VIEWFAC

Calculates the view factor for two finite plates at 90 deg taken from "Thermal Radiation Heat Transfer" by Robert Siegel and John R. Howell. Appendix C #14 "two finite rectangles of same length, having on common edge, and at an angle of 90 deg to each other."

**PROGRAM EARTHV**

Calculates the earth view factors at any point in an elliptical solar inertial orbit. Given the position in orbit assuming at time 0 the x axis points from nadir, the y axis points in the orbit direction and the z axis completes the right hand rule pointing north.

**FUNCTION XMAG**

Calculates the magnitude of the vector x

FORTRAN FILE	ROUTINE NAME
ORBITAL.FOR	PROGRAM ORBITAL
QENCL.FOR	PROGRAM QENCL SUBROUTINE ENCLREAD(X,Y,Z,FE,ALPIR,ALPS,EPS,XMU, + ALBEDO,NUMINS,INSUL,S,E) SUBROUTINE ENCWRIT(Q,T)
QENCLSUB.FOR	SUBROUTINE AREA(X,Y,Z,A) SUBROUTINE CALFEB(F,A,FEB) SUBROUTINE CALFLUX(S,XMU,A,ALPS,FE,E,AIR,ALB,EPS,X,XRAD,XRADI) SUBROUTINE CALCEB(X,Q,XRADI,EB) SUBROUTINE CALCQ(XFLUX,INSUL,NUMINS,FEB,XRADI,Q) SUBROUTINE CALCT(EB,SIGMA,T) SUBROUTINE CALCWVF(X,Y,Z,A,F) SUBROUTINE ENCINIT(SIGMA,XRAD,XRADI,FEB) FUNCTION VIEWFAC(HI,WI,XL)
EARTHV.FOR	PROGRAM EARTHV SUBROUTINE EAREAD(W,ECC,SMA,XNU,RE) SUBROUTINE EARWRIT(FE,FET,FEN)
CALVFNT.FOR	SUBROUTINE CALVFNR(ECC,SMA,XNU,RE,FEN) SUBROUTINE CALVFTR(ECC,SMA,XNU,RE,FET)
CALEART.FOR	SUBROUTINE CALFEP(FEN,FET,FE) SUBROUTINE CALFEI(W,FEN,FET,FE)
SOLARV.FOR	PROGRAM SOLARV SUBROUTINE SOLREAD(X) SUBROUTINE SOLWRIT(XMU)
CALSOLR.FOR	SUBROUTINE CALSOLR(X,XMU) SUBROUTINE SURFNOF(X,Y)

SUNVECT.FOR	PROGRAM SUNVECT SUBROUTINE SUNREAD(GS,XINC,OMEGA,WOMEGA,XNU) SUBROUTINE SUNWRIT(X,XMU)
CALSUN.FOR	SUBROUTINE CALSUN(G,XINC,OMEGA,WOMEGA,XNU,XS) SUBROUTINE PQR2IJK(XII,OI,WI,XNU,A,B)
MATMUL.FOR	SUBROUTINE MATMUL(N,M,L,A,B,C)
MATADD.FOR	SUBROUTINE MATADD(N,M,A,B,C,IFAC)
POSGLOB.FOR	SUBROUTINE POSGLOB(ECC,SMA,XI,O,W,XNU,RVECTG)
ORBSUBS.FOR	SUBROUTINE ORBCOMP(T,Q,VO,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW, + VOHIGH,VCLOW,QHIGH,QLOW,SQHIG,SQLOW,VOQHIG,VOQLOW,TDELTA, + QDELTA,VCDELTA,VOQDEL)  SUBROUTINE ORBFIN(NUMPTS,TAVET,QAVET,TAVE,QAVE) SUBROUTINE ORBINIT(SIGMA,XRAD,XRADI,FEB,TAVE,QAVE,TDELTA, + QDELTA,THIGH,QHIGH,TLOW,QLOW) SUBROUTINE ORBREAD(ALBEDO,S,E,GS,XI,O,WO,ECC,SMA,INERT,X,Y,Z, + RE,ALPIR,ALPS,EPS,NUMINS,INSUL,NUMPTS) SUBROUTINE ORBOUT(Q,T,XI,O,W,XN) SUBROUTINE ORBWRT(TAVET,QAVET,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW, + VOHIGH,VCLOW,QHIGH,QLOW,SQHIG,SQLOW,VOQHIG,VOQLOW,TDELTA, + QDELTA,VCDELTA,VOQDEL)
ORBPROPS.FOR	SUBROUTINE ORBPROPS(ECC,SMA,XNU,R)
DOTPROD.FOR	FUNCTION DOTPROD(X,Y)
FETANOR.FOR	PROGRAM FETANOR SUBROUTINE FERREAD(ECC,SMA,XNU) SUBROUTINE FEWRIT(FET,FEN)



IJK2PQR.FOR

SUBROUTINE IJK2PQR(XII,OI,WI,XNU,A,B)

MAG.FOR

FUNCTION XMAG(X)

SHADOW.FOR

SUBROUTINESHADOW(ECC,XNU,XINC,OMEGA,WOMEGA,GS,SORIG,S,XMU,RE)



**Appendix G**  
**Fortran Source Code**

**THIS PAGE INTENTIONALLY LEFT BLANK**

Fortran File Locations

QENCL.FOR .....	G6
QENCLSUB.FOR .....	G9
EARTHV.FOR .....	G15
CALVFNT.FOR .....	G17
CALEART.FOR .....	G18
SOLARV.FOR .....	G19
CALSOLR.FOR .....	G21
SUNVECT.FOR .....	G22
CALSUN.FOR .....	G24
MATMUL.FOR .....	G25
MATADD.FOR .....	G25
POSGLOB.FOR .....	G26
ORBSUBS.FOR .....	G27
ORBPROPS.FOR .....	G33
DOTPROD.FOR .....	G33
FETANOR.FOR .....	G34
IJK2PQR.FOR .....	G36
MAG.FOR .....	G37
SHADOW.FOR .....	G38

PREVIOUS PAGE BLANK NOT FILMED

## FORTRAN Source Code

G4

### ORBITAL.FOR

```
PROGRAM ORBITAL
  IMPLICIT REAL*8 (A-H, O-Z)
```

```
C THIS PROGRAM SOLVES FOR THE TEMPERATURE AND
C HEAT LOADS ON AN ENCLOSURE AT DISTINCT ORBIT
C POINTS AS SPECIFIED BY THE USER. FOR INERTIALLY
C OR PLANET ORIENTED ORBITS. IT IS USED IN
C CONJUNCTION WITH THE EXPERT SYSTEM RADIATE.RUL
C RACHEL ORMSBY July 20, 1993
C
C O(6) THE HEAT LOAD ON EACH SIDE OF A RECTANGULAR BOX
C EB(6) THE BLACKBODY RADIATION ON EACH SIDE OF THE BOX
C F(6,6) THE VIEW FACTORS INSIDE THE BOX
C FE(6) THE VIEW FACTORS TO EARTH FOR EACH SIDE
C ALPIR THE INFRARED ABSORPTIVITY FOR THE EXTERNAL SURFACES
C   = EMISSIVITY
C ALPS THE SOLAR ABSORPTIVITY FOR THE EXTERNAL SURFACES
C EPS THE EMISSIVITY FOR THE EXTERNAL SURFACES
C   THE INTERNAL SURFACE IS ASSUMED BLACK
C XMU THE SOLAR VIEW FACTOR TO THE EXTERNAL SURFACES
C A THE AREA OF EACH SIDE
C FEB COEFFICIENTS TO EB; [FEB] [EB] = [Q]
C X MATRIX CONTAINING THE ENVIRONMENTAL FLUX
C   = XSUN + XEARTH + XALBEDO
C NUMINS NUMBER OF INSULATED SIDES
C INSUL CONTAINS THE SIDES THAT ARE INSULATED (NO HEAT
C   FLOW THROUGH (E.g. RADIATOR SIDE))
C NUMPTS NUMBER OF ORBIT POINTS
C
C XINC ORBIT INCLINATION (DEG)
C OMEGA ORBIT LONGITUDE OR ASCENDING NODE (DEG)
C WOMEGA LINE OF PERIAPSIS (DEG)
C XNU TRUE ANAMOLY AT EPOCH (DEG)
C ECC ECCENTRICITY
C SMA SEMI-MAJOR AXIS
C
C INERT INERTIAL FLAG
C   1 - INERTIAL
C   0 - PLANET
C
C XS IS THE LOCAL SUNVECTOR
C GS IS THE GLOBAL SUNVECTOR
```

```
C
  DIMENSION O(6),EB(6),I(6)
  DIMENSION F(6,6), FE(6)
  DIMENSION ALPIR(6), ALPS(6), EPS(6)
  DIMENSION XMU(6), A(6)
  DIMENSION FEB(6,6), XRAD(6,6), XRAID(6,6), X(6)
  DIMENSION INSUL(5)
  DIMENSION XS(3), GS(3)
  DIMENSION IAVE(6),OAVE(6)

C OPEN THE OUTPUT FILE "RAD_ORB.DAT" CONTAINS THE
C INFORMATION FOR EACH OF THE NUMPTS ORBIT POINTS
  OPEN(10,FILE='RAD_ORB.DAT',STATUS='NEW')

C READ INPUT FILE
  CALL ORBREAD(ALBEDO,S,E,GS,XINC,OMEGA,WOMEGA,ECC,SMA,INERT,X,Y,Z,
  + RE,ALPIR,ALPS,EPS,NUMINS,INSUL,NUMPTS)
C INITIALIZE ANALYSIS VALUES
  CALL ORBIT(SIGMA,XRAD,XRAID,FEB,IAVE,OAVE,DELTA,ODELTA,
  + THIGH,QHIGH,LOW,QLOW)
C SET UP THE ORBITAL INTERVAL
  TWOPI = 2*3.141592654
  XNINTERV = TWOPI/NUMPTS

C CALCULATE SURFACE AREAS
  CALL AREA(X,Y,Z,A)
C CALCULATE ENCLOSURE VIEW FACTORS
  CALL CALCVWF(X,Y,Z,A,F)
C MULTIPLIES A MODIFIED F MATRIX WITH THE AREA MATRIX TO
C PRODUCE THE COEFFICIENTS TO THE EB MATRIX
  CALL CALFEB(F,A,FEB)

C INITIALLY DEFINE THE TRUE ANOMALY = - LINE OF PERIAPSIS
C THIS MAKES XNU GO FROM 0 TO 360 DEG. THE SPACECRAFT MOVES
C TO THE LEFT.
C START AT LINE OF NODES
  XNU = -WOMEGA
C
C
  DO 10 I = 1, NUMPTS
C CALCULATE THE NORMAL AND TANGENTIAL EARTH VIEW FACTORS
  CALL CALVFNOR(ECC,SMA,XNU,RE,FEN)
  CALL CALVFTAN(ECC,SMA,XNU,RE,FET)
```

```

C CALCULATE THE EFFECTIVE SPACECRAFT POSITION
  W = (WOMEGA+XNU)
C CALCULATE THE EARTH VIEW FACTORS FOR INERTIAL OR
C PLANET ORIENTED SPACECRAFT
  IF (INERT.EQ.1) THEN
    CALL CALFEI(W,FEN,FET,FE)
  ELSE
    CALL CALFEP(FEN,FET,FE)
  ENDIF

C DETERMINE IF THE SPACECRAFT IS IN THE SHADOW. THE SOLAR FLUX
C RETURNED IS THE EFFECTIVE FLUX AND VIEW FACTORS
  CALL SHADOW(ECC,SMA,XNU,XINC,OMEGA,WOMEGA,GS,S,SEFF,XMU,RE)
C IF NOT IN THE SHADOW
C  CALCULATE THE SUN VECTOR
C  CALCULATE THE SOLAR VIEW FACTORS
  IF (SEFF.GT.0) THEN
C IF THE SPACECRAFT IS INERTIALLY ORIENTED, THE SOLAR VIEW FACTORS
C ARE AT THE LINE OF NODES. THEY MUST BE RECALCULATED IN CASE
C THE SPACECRAFT WAS PREVIOUSLY IN THE SHADOW.
  IF (INERT.EQ.1) THEN
    CALL CALSUN(GS,XINC,OMEGA,0,0,XS)
  ELSE
    CALL CALSUN(GS,XINC,OMEGA,WOMEGA,XNU,XS)
  ENDIF
  CALL CALSOLR(XS,XMU)
ENDIF

C CALCULATE THE SOLAR FLUX
  CALL CALFLUX(SEFF,XMU,A,ALPS,FE,E,ALPIR,ALBEDO,EPS,X,XRAD,
+      XRADI)

C CALCULATE THE Q ON EACH SIDE TAKING INTO ACCOUNT DECOUPLED SIDES
  CALL CALCO(X,INSUL,NUMINS,FEB,XRADI,Q)
C CALCULATE THE BLACKBODY RADIATION
  CALL CALCEB(X,Q,XRADI,EB)
C CALCULATE THE TEMPERATURE
  CALL CALCT(EB,SIGMA,T)

C PRINT THE DATA TO THE FILE "RAD_ORB.DAT"
  CALL ORBOUT(Q,T,XINC,OMEGA,WOMEGA,XNU)

C COMPUTE THE STATISTICAL INFORMATION
  CALL ORBCOMP(T,Q,XNU,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW,VOHIGH,
+      VOLOW,QHIGH,QLOW,SOHIGH,SQLOW,VOOHIGH,VOOLOW,
+      TDELTA,QDELTA,VODEL,T,VODEL)
C INCREMENT THE ANGLE
  XNU = XNU - XINTERV
  10 CONTINUE

C CALCULATE THE FINAL SPACECRAFT STATISTICS
  CALL ORBFIN(NUMPTS,TAVET,QAVET,TAVE,QAVE)

C WRITE DATA TO "RETURN.DAT"
  CALL ORBWRT(TAVET,QAVET,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW,
+      VOHIGH,VOLOW,QHIGH,QLOW,SOHIGH,SQLOW,VOOHIGH,VOOLOW,
+      TDELTA,QDELTA,VODEL,T,VODEL)

  CLOSE(10)
  END

```

**FORTTRAN Source Code**

## FORTRAN Source Code

G6

### QENCL.FOR

```
PROGRAM QENCL
  IMPLICIT REAL*8 (A-H, O-Z)

C THIS PROGRAM SOLVES FOR THE TEMPERATURE AND
C HEAT LOADS ON AN ENCLOSURE. IT IS USED IN
C CONJUNCTION WITH THE EXPERT SYSTEM RADIATE.RUL
C RACHEL ORMSBY June 29, 1993
C
C SOLVES FOR Q INSTEAD OF EB
C
C Q(6) THE HEAT LOAD ON EACH SIDE OF A RECTANGULAR BOX
C EB(6) THE BLACKBODY RADIATION ON EACH SIDE OF THE BOX
C F(6,6) THE VIEW FACTORS INSIDE THE BOX
C FE(6) THE VIEW FACTORS TO EARTH FOR EACH SIDE
C ALPIR THE INFRARED ABSORPTIVITY FOR THE EXTERNAL SURFACES
C   = EMISSIVITY
C ALPS THE SOLAR ABSORPTIVITY FOR THE EXTERNAL SURFACES
C EPS THE EMISSIVITY FOR THE EXTERNAL SURFACES
C   THE INTERNAL SURFACE IS ASSUMED BLACK
C XMU THE SOLAR VIEW FACTOR TO THE EXTERNAL SURFACES
C A THE AREA OF EACH SIDE
C FEB COEFFICIENTS TO EB: [FEB] [EB] = [Q]
C X MATRIX CONTAINING THE ENVIRONMENTAL FLUX
C   = XSUN + XEARTH + XALBEDO
C NUMINS NUMBER OF INSULATED SIDES
C INSUL CONTAINS THE SIDES THAT ARE INSULATED (NO HEAT
C   FLOW THROUGH (E.g. RADIATOR SIDE)

  DIMENSION Q(6),EB(6)
  DIMENSION F(6,6), FE(6)
  DIMENSION ALPIR(6), ALPS(6), EPS(6)
  DIMENSION XMU(6), A(6)
  DIMENSION FEB(6,6), XRAD(6,6), XRADI(6,6), XE(6)
  DIMENSION INSUL(6)

C READ INPUT FROM EXPERT SYSTEM
  CALL ENCLREAD(XX,YY,ZZ,FE,ALPIR,ALPS,EPS,XMU,ALBEDO,NUMINS,
+             INSUL,S,E)
C INITIALIZES ANALYSIS VALUES
  CALL ENCINIT(SIGMA,XRAD,XRADI,FEB)

C CALCULATES SURFACE AREAS
```

```
  CALL AREA(XX,YY,ZZ,A)
C CALCULATES ENCLOSURE VIEW FACTORS
  CALL CALCVWF(XX,YY,ZZ,A,F)
C MULTIPLIES A MODIFIED F MATRIX WITH THE AREA MATRIX TO
C PRODUCE THE COEFFICIENTS TO THE EB MATRIX
  CALL CALFEB(F,A,FEB)
C CALCULATES THE ENVIRONMENTAL FLUXES
  CALL CALFLUX(S,XMU,A,ALPS,FE,E,ALPIR,ALBEDO,EPS,XE,XRAD,XRADI)

C CALCULATES THE HEAT FLUX
  CALL CALCO(XE,INSUL,NUMINS,FEB,XRADI,Q)
C CALCULATES THE BLACKBODY RADIATION
  CALL CALCEB(XE,Q,XRADI,EB)
C CALCULATES THE TEMPERATURES
  CALL CALCT(EB,SIGMA,T)
C WRITES THE OUTPUT TO "RETURN.DAT"
  CALL ENCWRT(Q,T)

  END
```



```

SUBROUTINE ENCLREAD(X,Y,Z,FE,ALPIR,ALPS,EPS,XMU,ALBEDO,
+ NUMINS,INSUL,S,E)
IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C
C X THE X DIMENSION OF THE RECTANGULAR BOX (m)
C Y THE Y DIMENSION OF THE RECTANGULAR BOX (m)
C Z THE Z DIMENSION OF THE RECTANGULAR BOX (m)
C ALBEDO THE ALBEDO COEFFICIENT (USUALLY .3)
C FE THE EARTH VIEW FACTORS
C ALPIR THE INFRARED ABSORPTIVITY = EMISSIVITY
C ALPS THE SOLAR ABSORPTIVITY
C XMU THE SOLAR VIEW FACTOR
C NUMINS THE NUMBER OF INSULATED SIDES
C INSUL THE SIDES THAT ARE INSULATED
C S SOLAR FLUX
C E PLANET FLUX
C
DIMENSION FE(6),ALPIR(6),ALPS(6),EPS(6),XMU(6),INSUL(6)

OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

READ(3,*) ALBEDO
READ(3,*) S
READ(3,*) E

READ(3,*) X
READ(3,*) Y
READ(3,*) Z

DO 10 I=1,6
  READ(3,*) FE(I)
10 CONTINUE

DO 20 I=1,6
  READ(3,*) ALPIR(I)
  EPS(I) = ALPIR(I)
20 CONTINUE

DO 30 I=1,6
  READ(3,*) ALPS(I)
30 CONTINUE

```

```

DO 50 I=1,6
  READ(3,*) XMU(I)
50 CONTINUE

READ(3,*) NUMINS
IF (NUMINS.GT.0) THEN
  DO 60 I=1,NUMINS
    READ(3,*) INSUL(I)
60 CONTINUE
  ENDIF

CLOSE(3)
RETURN
END

```

## FORTRAN Source Code

G8

```
      SUBROUTINE ENCWRT(Q,T)
      IMPLICIT REAL*8 (A-H, O-Z)
C
C WRITES TEMPERATURE AND HEAT LOAD TO "RETURN.DAT"
C
      DIMENSION Q(6), T(6)

      OPEN(10,FILE='RETURN.DAT',STATUS='UNKNOWN')

      WRITE(10,*) 'T1',T(1)
      WRITE(10,*) 'T2',T(2)
      WRITE(10,*) 'T3',T(3)
      WRITE(10,*) 'T4',T(4)
      WRITE(10,*) 'T5',T(5)
      WRITE(10,*) 'T6',T(6)
      WRITE(10,*)

      DO 10 I=1,6
      IF (ABS(Q(I)) .LT. 1.0E-6) THEN
      Q(I) = 0.0
      ENDF
10 CONTINUE

      WRITE(10,*) 'Q1',Q(1)
      WRITE(10,*) 'Q2',Q(2)
      WRITE(10,*) 'Q3',Q(3)
      WRITE(10,*) 'Q4',Q(4)
      WRITE(10,*) 'Q5',Q(5)
      WRITE(10,*) 'Q6',Q(6)
      WRITE(10,*)

      RETURN
      END
```

## QENCLSUB.FOR

```
SUBROUTINE AREA(X,Y,Z,A)
  IMPLICIT REAL*8 (A-H, O-Z)
```

```
C THIS SUBROUTINE CALCULATES THE AREAS OF EACH SIDE
C THE SIDE AXES ARE DEFINED.
```

```
DIMENSION A(6)
```

```
A(1) = X*Y
A(2) = Z*X
A(3) = Z*Y
A(4) = Z*X
A(5) = Z*Y
A(6) = X*Y
```

```
RETURN
END
```

```
SUBROUTINE CALFEB(F,A,FEB)
  IMPLICIT REAL*8(A-H,O-Z)
```

```
C CALCULATES THE FEB MATRIX
C FEB = VIEW FACTORS * AREAS
C IT IS THE COEFFICIENT MATRIX TO THE BLACKBODY
C RADIATION TERMS, EB (SEE DOCUMENTATION)
  DIMENSION F(6,6), FEB(6,6), A(6)
```

```
C F THE ENCLOSURE VIEW FACTOR MATRIX.
C A THE AREA MATRIX
```

```
C THE DIAGONALS OF THE FEB MATRIX ARE THE SUM OF ALL THE
C VIEW FACTORS FOR ONE SIDE (=1) TIMES THE AREA FOR THAT SIDE.
C THE REMAINING POSITIONS ARE -F TIMES THE AREA
```

```
DO 10 I=1,6
  DO 20 J=1,6
    IF (I.EQ.J) THEN
      FEB(I,I) = A(I)
    ELSE
      FEB(I,J) = -F(I,J)*A(I)
    ENDIF
  20 CONTINUE
10 CONTINUE
```

```
RETURN
END
```

## FORTRAN Source Code

G10

```

SUBROUTINE CALFLUX(S,XMU,A,ALPS,FE,E,AIR,ALB,EPS,X,
+ XRAD,XRADI)
IMPLICIT REAL*8(A-H,O-Z)
C
C CALCULATES THE ENVIRONMENTAL FLUXES
C FILLS THE X MATRIX AND THE XRAD MATRIX
C
C S SOLAR FLUX (W/m-2)
C XMU(6) SOLAR VIEW FACTORS
C A AREA MATRIX
C ALPS(6) SOLAR ABSORPTIVITY
C FE(6) EARTH VIEW FACTORS
C AIR(6) INFRARED ABSORPTIVITY
C ALB ALBEDO COEFFICIENT (USUALLY .3)
C EPS(6) EMMISIVITY
C X(6) ENVIRONMENTAL RADIATION ON EACH SIDE
C XRAD(6,6) INTERNAL HEAT TO BE RADIATED
C XRADI(6,6) 1/XRAD

DIMENSION A(6),FE(6),AIR(6),XMU(6),EPS(6), ALPS(6)
DIMENSION X(6), XRAD(6,6), XRADI(6,6)

DO 10 I=1,6
  OSUN = S*XMU(I)*A(I)*ALPS(I)
  OEARTH = A(I)*FE(I)*E*AIR(I)
  OALBEDO = A(I)*FE(I)*(1-XMU(I))*S*ALB*ALPS(I)
  ORAD = (A(I)*EPS(I))

  X(I) = OSUN + OEARTH + OALBEDO
  XRAD(I,I) = -ORAD
C XRAD AND XRADI WERE INITIALIZED TO
C A ZERO MATRIX IN THE ROUTINE "ENCINIT"
C SINCE XRAD IS A DIAGONAL MATRIX, THE INVERSE IS THE
C RECIPRICOL OF EACH OF THE DIAGONAL TERMS.
  XRADI(I,I) = 1/XRAD(I,I)
10 CONTINUE

RETURN
END

```

```

SUBROUTINE CALCEB(X,Q,XRADI,EB)
IMPLICIT REAL*8(A-H,O-Z)
C
C CALCULATES THE BALCKBODY RADIATION FOR EACH SURFACE
C X IS THE ENVIRONMENTAL FLUXES
C Q IS THE CALCULATED HEAT LOAD
C XRADI IS THE INVERSE OF THE RADIATED HEAT
C [Q_X] = [Q] - [X]
C [EB] = [XRADI] * [Q_X]
C
DIMENSION X(6),Q(6), EB(6,6), XRADI(6,6)
DIMENSION Q_X(6)

C IFAC = -1 SUBTRACT MATRICES
IFAC = -1
CALL MATADD(6,1,Q,X,Q_X,IFAC)
C EB MATRIX = 1/XRAD * (Q-X) OR Q = X + XRAD*EB
CALL MATMUL(6,6,1,XRADI,Q_X,EB)
RETURN
END

```

```

SUBROUTINE CALCO(X,INSUL,NUMINS,FEB,XRADI,Q)
  IMPLICIT REAL*8(A-H,O-Z)
  C
  C CALCULATES THE HEAT LOAD ON EACH SURFACE
  C
  DIMENSION FEBXRADI(6,6),B(6),IWK(6)
  DIMENSION X(6),Q(6), FEB(6,6), XRADI(6,6)
  DIMENSION XI(6,6), A(6,6)
  DIMENSION INSUL(6)
  C SET UP THE IDENTITY MATRIX [XI]
  DO 10 I=1,6
    DO 20 J=1,6
      IF (I.EQ.J) THEN
        XI(I,J) = 1.0
      ELSE
        XI(I,J) = 0.0
      ENDIF
    20 CONTINUE
  10 CONTINUE
  C [FEB] = MATRIX OF VIEW FACTORS ([Q] = [FEB] [EB])
  C [XRADI] = INVERSE OF [XRAD] ([Q] = [X] + [XRAD][EB])
  C [X] IS THE ENVIRONMENTAL LOADS
  C
  C DEFINE FOR A*Y = B
  C [A] = [FEB] [XRADI] - [XI]
  C [B] = [FEB] [XRADI] [X]
  C [Y] = [Q]
  C
  C IF NUMBER OF INSULATED SIDES > 0 THEN ZERO
  C THEN ZERO Q BY SETTING X(SIDE) = 0 AND
  C XRAD(SIDE) = 0.0 OR XRADI(SIDE) = 9.9E+9
  IF (NUMINS.GT.0) THEN
    DO 30 K=1,NUMINS
      I = INSUL(K)
      XRADI(I,I) = 9.9E+9
      X(I) = 0.0
    30 CONTINUE
  ENDIF
  C [FEBXRADI] = [FEB] [XRADI]
  CALL MATMUL(6,6,6,FEB,XRADI,FEBXRADI)

```

```

  C CALCULATE A AND B
  IFAC = -1
  CALL MATADD(6,6,FEBXRADI,XI,A,IFAC)
  CALL MATMUL(6,6,1,FEBXRADI,X,B)
  C SOLVE A*Y = B
  INFO = 0
  JOB = 0
  CALL DGEFA(A,6,6,IWK,INFO)
  CALL DGESL(A,6,6,IWK,B,JOB)
  C FILL THE Q VECTOR WITH THE SOLUTION
  DO 80 I=1,6
    Q(I) = B(I)
  80 CONTINUE
  RETURN
  END

```

## FORTRAN Source Code

G12

```
      SUBROUTINE CALCT(EB,SIGMA,T)
      IMPLICIT REAL*8 (A-H, O-Z)
      C
      C CALCULATES THE TEMPERATURES GIVEN EB
      C
      DIMENSION EB(6),T(6)

      C EB = SIGMA*T-4
      DO 10 I=1,6
         T(I) = SORT(SORT(EB(I)/SIGMA))
      10 CONTINUE

      RETURN
      END
```

```
      SUBROUTINE CALCVMWF(X,Y,Z,A,F)
      IMPLICIT REAL*8 (A-H, O-Z)
      C
      C CALCULATES VIEW FACTORS FOR THE INTERNAL ENCLOSURE
      C
      DIMENSION F(6,6), A(6)

      F(1,1) = 0.0
      F(2,2) = 0.0
      F(3,3) = 0.0
      F(4,4) = 0.0
      F(5,5) = 0.0
      F(6,6) = 0.0

      C CALCULATE VIEW FACTOR ON THREE SIDES, USE RECIPROCITY
      C AND SUM OF VIEW FACTORS = 1 FOR THE REST
      F(1,3) = VIEWFAC(X,Z,Y)
      F(2,1) = VIEWFAC(Z,Y,X)
      F(3,2) = VIEWFAC(Y,X,Z)

      F(1,5) = F(1,3)
      F(2,5) = F(2,1)
      F(2,6) = F(2,1)
      F(3,4) = F(3,2)
      F(4,1) = F(2,1)
      F(4,3) = F(3,2)
      F(4,6) = F(2,1)
      F(5,4) = F(3,2)
      F(6,3) = F(1,3)
      F(6,5) = F(1,3)

      F(1,2) = F(2,1)* (A(2)/A(1))
      F(2,3) = F(3,2)* (A(3)/A(2))
      F(3,1) = F(1,3)* (A(1)/A(3))
      F(4,5) = F(5,4)* (A(5)/A(4))

      F(1,4) = F(1,2)
      F(3,6) = F(3,1)
      F(5,1) = F(3,1)
      F(5,2) = F(1,2)
      F(5,6) = F(3,1)
      F(6,2) = F(1,2)
      F(6,4) = F(1,2)
```

```

F(1,6) = 1-(F(1,2)+F(1,3)+F(1,4)+F(1,5))
F(2,4) = 1-(F(2,1)+F(2,3)+F(2,6)+F(2,5))
F(3,5) = 1-(F(3,1)+F(3,2)+F(3,4)+F(3,6))
F(4,2) = 1-(F(4,1)+F(4,3)+F(4,6)+F(4,5))
F(5,3) = 1-(F(5,1)+F(5,2)+F(5,4)+F(5,6))
F(6,1) = 1-(F(6,2)+F(6,3)+F(6,4)+F(6,5))
C
C THE FOLLOWING COMMENTED CODE ALLOWS THE DESIGNER
C TO WRITE OUT THE ENCLOSURE VIEW FACTORS, FOR INSPECTION
C OPEN (10,FILE='OUT.DAT',STATUS='UNKNOWN')
C DO 10 I=1,6
C DO 20 J=1,6
C WRITE(10,*) I,J,F(I,J)
C 20 CONTINUE
C 10 CONTINUE

RETURN
END

```

```

SUBROUTINE ENCINIT(SIGMA,XRAD,XRADI,FEB)
IMPLICIT REAL*8(A-H,O-Z)
C
C INITIALIZES THE CONSTANTS AND SETS THE MATRICES TO 0.0
C
DIMENSION FEB(6,6), XRAD(6,6), XRADI(6,6)

SIGMA = 5.67E-8

DO 10 I=1,6
DO 20 J=1,6
XRAD(I,J) = 0.0
XRADI(I,J) = 0.0
FEB(I,J) = 0.0
20 CONTINUE
10 CONTINUE

RETURN
END

```

## FORTRAN Source Code

G14

```
FUNCTION VIEWFAC(HI,WI,XL)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C CALCULATES THE VIEWFACTOR FOR TWO FINITE PLATES AT 90 DEG
C TAKEN FROM "THERMAL RADIATION HEAT TRANSFER" BY ROBERT SIEGEL AND
C JOHN R. HOWELL. APPENDIX C #14 "TWO FINITE RECTANGLES OF SAME LENGTH,
C HAVING ON COMMON EDGE, AND AT AN ANGLE OF 90 DEG TO EACH OTHER."
C
  XPI = 3.141592654

  H = HI/XL
  W = WI/XL

  A = 1/(XPI*W)
  B = W*ATAN(1/W)
  C = H*ATAN(1/H)
  HH = H*H
  WW = W*W
  D = HH + WW
  E = SORT(D)
  F = E*ATAN(1/E)
  G = 1+WW
  O = 1+HH
  P = 1+HH+WW
  PP = HH+WW
  Q = .25*DLOG(G*O/P)
  R = .25*DLOG(WW*P/(G*PP))
  S = .25*DLOG(HH*P/(O*PP))

  T = B+C-F+Q+R+S
  U = A*T

  VIEWFAC = U

  RETURN
  END
```



## EARTHV.FOR

```
PROGRAM EARTHV
  IMPLICIT REAL*8 (A-H, O-Z)

C THIS PROGRAM CALCULATES THE EARTH VIEW FACTORS
C AT ANY POINT IN AN ELLIPTICAL SOLAR INERTIAL ORBIT.
C GIVEN THE POSITION IN ORBIT
C ASSUMING AT TIME 0 THE X AXIS POINTS FROM NADIR,
C THE Y AXIS POINTS IN THE ORBIT DIRECTION AND
C THE Z AXIS COMPLETES THE RIGHT HAND RULE POINTING NORTH.
C
C THIS PROGRAM IS USED WITH THE EXPERT SYSTEM RADIATE.RUL
C RACHEL ORMSBY July 15, 1993
C
  DIMENSION FE(6)

C READS INPUT
  CALL EAREAD(W,ECC,SMA,XNU,RE)
C CALCULATES THE NORMAL AND TANGENTIAL COMPONENTS OF THE
C EARTH VIEW FACTORS
  CALL CALVFNOR(ECC,SMA,XNU,RE,FEN)
  CALL CALVFTAN(ECC,SMA,XNU,RE,FET)

C DEFINES THE SPACECRAFT POSITION AS THE LINE OF PERIAPSIS, W,
C MINUS THE TRUE ANOMOLY. NOTE THIS ORBITS PROCEEDS TO THE LEFT.
  WI = (W-XNU)
C ROTATES THE VIEW FACTORS TO THE CORRECT SIDES
  CALL CALFEI(WI,FEN,FET,FE)

  CALL EARWRIT(FE,FET,FEN)

END
```

```
SUBROUTINE EAREAD(W,ECC,SMA,XNU,RE)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C
C W LINE OF PERIAPSIS
C ECC ECCENTRICITY
C SMA SEMI-MAJOR AXIS
C XNU TRUE ANOMOLY AT EPOCH
C RE RADIUS OF THE PLANET

  DTR = 1.7453293E-2

  OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

  READ(3,*) INC
  READ(3,*) ECC
  READ(3,*) W
  READ(3,*) SMA
  READ(3,*) XNU
  XNU = -XNU
  READ(3,*) RE

  INC = INC*DTR
  W = W*DTR
  XNU = XNU*DTR

  RETURN
END
```

## FORTRAN Source Code

G16

```
SUBROUTINE EARWRIT(FE,FET,FEN)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C WRITES PLANET VIEW FACTORS FOR EACH SIDE, AND THE
C NORMAL AND TANGENTIAL COMPONENTS TO "RETURN.DAT"
C
  DIMENSION FE(6)
  OPEN(10,FILE='RETURN.DAT',STATUS='UNKNOWN')

  WRITE(10,*) 'FE1',FE(1)
  WRITE(10,*) 'FE2',FE(2)
  WRITE(10,*) 'FE3',FE(3)
  WRITE(10,*) 'FE4',FE(4)
  WRITE(10,*) 'FE5',FE(5)
  WRITE(10,*) 'FE6',FE(6)
  WRITE(10,*) 'FET',FET
  WRITE(10,*) 'FEN',FEN

  CLOSE(10)

  RETURN
  END
```

## CALVFNT.FOR

```
      SUBROUTINE CALVFNOR(ECC,SMA,XNU,RE,FEN)
      IMPLICIT REAL*8 (A-H, O-Z)
      C CALCULATES THE VIEW FACTOR FROM A PLANE ELEMENT TO
      C A SPHERE. USED AS THE EARTH VIEW FACTOR NORMAL
      C FROM "THERMAL RADIATION HEAT TRANSFER" BY SIEGEL AND
      C HOWELL: APPENDIX C

      C CALCULATE THE SPACECRAFT POSITION COEFFICIENT, R.
      CALL ORBPROPS(ECC,SMA,XNU,R)

      FEN = (RE/R)*(RE/R)

      RETURN
      END
```

```
      SUBROUTINE CALVFTAN(ECC,SMA,XNU,RE,FET)
      IMPLICIT REAL*8 (A-H, O-Z)
      C CALCULATES THE VIEW FACTOR FROM A PLANE ELEMENT TO
      C A SPHERE. USED AS THE EARTH VIEW FACTOR TANGENTIAL
      C FROM "THERMAL RADIATION HEAT TRANSFER" BY SIEGEL AND
      C HOWELL: APPENDIX C

      XPI = 3.141592654

      C CALCULATE THE SPACECRAFT POSITION COEFFICIENT, R.
      CALL ORBPROPS(ECC,SMA,XNU,R)

      H = R/RE

      H2 = SQRT(H*H-1)
      HT = ATAN(1/H2)

      FET = (1/XPI)*(HT - (H2/(H*H)))

      RETURN
      END
```

## FORTRAN Source Code

.G18

### CALEART.FOR

```
      SUBROUTINE CALFEP(FEN,FET,FE)
      IMPLICIT REAL*8 (A-H,O-Z)
      C CALCULATES THE EARTH VIEW FACTOR FOR PLANET
      C ORIENTATION
      DIMENSION FE(6)

      FE(1) = FET
      FE(2) = FET
      FE(3) = 0.0
      FE(4) = FET
      FE(5) = FEN
      FE(6) = FET

      RETURN
      END
```

```
      SUBROUTINE CALFEI(W,FEN,FET,FE)
      IMPLICIT REAL*8 (A-H,O-Z)
      C CALCULATES THE EARTH VIEW FACTOR FOR INERTIAL
      C ORIENTATION
      DIMENSION FE(6)

      C W IS THE SPACECRAFT POSITION = LINE OF PERIAPSIS -
      C TRUE ANOMOLY
      SW = SIN(W)
      CW = COS(W)

      FE(1) = FET
      FE(2) = -FEN*SW + FET*ABS(CW)
      FE(3) = FET*ABS(SW) - FEN*CW
      FE(4) = FEN*SW + FET*ABS(CW)
      FE(5) = FE(1)*ABS(SW) + FE(3)*CW
      FE(6) = FET

      C IF THE VIEW FACTOR IS LESS THAN ZERO, THE SPACECRAFT
      C SURFACE CANNOT SEE THE PLANET.
      DO 10 I=1,6
      IF (FE(I).LT.0.0) THEN
      FE(I) = 0.0
      ENDIF
      10 CONTINUE

      RETURN
      END
```

## SOLARV.FOR

```
PROGRAM SOLARV
  IMPLICIT REAL*8 (A-H, O-Z)

C THIS PROGRAM SOLVES FOR THE DOT PRODUCT OF
C THE SOLAR VECTOR AND THE LOCAL UNIT NORMALS
C ON EACH SIDE OF THE ENCLOSURE.
C THIS PROGRAM IS USED WITH THE EXPERT SYSTEM RADIATE.RUL
C RACHEL ORMSBY July 7, 1993
C
  DIMENSION XMU(6),X(3)

  CALL SOLREAD(X)

  CALL CALSOLR(X,XMU)
  CALL SOLWRIT(XMU)

  END
```

```
SUBROUTINE SOLREAD(X)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C
C XS THE X COORDINATE OF THE SOLAR VECTOR
C YS THE Y COORDINATE OF THE SOLAR VECTOR
C ZS THE Z COORDINATE OF THE SOLAR VECTOR
C MAGS THE MAGNITUDE OF THE SOLAR VECTOR

  DIMENSION X(3)

  OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

  READ(3,*) X(1)
  READ(3,*) X(2)
  READ(3,*) X(3)

  RETURN
  END
```

## FORTRAN Source Code

G20

```
SUBROUTINE SOLWRIT(XMU)
  IMPLICIT REAL*8 (A-H, O-Z)
  C
  C WRITES BLACKBODY RADIATION, HEAT LOAD AND
  C VIEW FACTORS TO "RETURN.DAT"
  C
  DIMENSION XMU(6)
  OPEN(10,FILE='RETURN.DAT',STATUS='UNKNOWN')

  WRITE(10,*) 'MU1',XMU(1)
  WRITE(10,*) 'MU2',XMU(2)
  WRITE(10,*) 'MU3',XMU(3)
  WRITE(10,*) 'MU4',XMU(4)
  WRITE(10,*) 'MU5',XMU(5)
  WRITE(10,*) 'MU6',XMU(6)

  CLOSE(10)
  RETURN
  END
```

## CALSOLR.FOR

```
      SUBROUTINE CALSOLR(X,XMU)
      IMPLICIT REAL*8 (A-H, O-Z)
      C CALCULATES THE DOT PRODUCT OF THE SOLAR VECTOR AND THE
      C UNIT NORMALS OF EACH SURFACE
      DIMENSION XMU(6),X(3)

      C DEFINES THE SURFACE NORMAL ON EACH SIDE OF THE ENCLOSURE
      CALL SURFNOR(X,XMU)

      C IF THE VIEW FACTOR IS LESS THAN 0, THE SURFACE CANNOT SEE
      C THE SUN.
      DO 10 I=1,6
      IF (XMU(I).LT.0.0) THEN
      XMU(I) = 0.0
      ELSE
      XMU(I) = ABS(XMU(I))
      ENDIF
      C IF THE VIEW FACTOR IS A NUMBER SMALLER THAT 1.E-3, IT WILL
      C BE WRITTEN TO THE FILE IN SCIENTIFIC NOTATION, THEREFORE
      C ZERO IT OUT TO ELIMINATE THE ERROR. EXSYS* CANNOT READ
      C SCIENTIFIC NOTATION
      IF (XMU(I) .LT. 1.0E-3) THEN
      XMU(I) = 0.0
      ENDIF
      10 CONTINUE

      RETURN
      END
```

```
      SUBROUTINE SURFNOR(X,Y)
      IMPLICIT REAL*8 (A-H, O-Z)
      C CALCULATES THE DOT PRODUCT OF X INTO THE LOCAL
      C SURFACE NORMALS AS DEFINED IN DOCUMENTATION
      C X IS A VECTOR IN LOCAL COORDINATES
      C Y IS THE DOT PRODUCT WITH THE SURFACE NORMALS
```

```
      DIMENSION Y(6),X(3)
```

```
      XS = X(1)
      YS = X(2)
      ZS = X(3)
```

```
      C CALCULATES THE MAGNITUDE OF THE SUN VECTOR, IN CASE
      C IT IS NOT 1. XMAG IS A FUNCTION CALL...
      XMAGS = XMAG(X)
```

```
      C DEFINES THE SOLAR VIEW FACTORS. SINCE EACH SURFACE NORMAL
      C IS IN THE DIRECTION OF A COORDINATE AXIS, THE DOT PRODUCT IS
      C "HARD" CODED IN HERE.
```

```
      Y(1) = -ZS/XMAGS
      Y(2) = -YS/XMAGS
      Y(3) = -XS/XMAGS
      Y(4) = YS/XMAGS
      Y(5) = XS/XMAGS
      Y(6) = ZS/XMAGS
```

```
      RETURN
      END
```

## FORTRAN Source Code

G22

### SUNVECT.FOR

```
PROGRAM SUNVECT
  IMPLICIT REAL*8 (A-H, O-Z)
C
C THIS PROGRAM CALCULATES THE SUN VECTOR IN LOCAL
C COORDINATES, GIVEN INCLINATE, LONGITUDE OF
C ASCENDING NODE AND SATELLITE POSITION.
C THIS CODE IS USED WITH THE EXPERT SYSTEM
C RADIATE.RULE. RAO 7/16/93
C
  DIMENSION X(3),GS(3), XMU(6)

  CALL SUNREAD(GS,XINC,OMEGA,WOMEGA,XNU)
C CALCULATES THE SUN VECTOR IN LOCAL COORDINATES
  CALL CALSUN(GS,XINC,OMEGA,WOMEGA,XNU,X)
C CALCULATES THE SOLAR VIEW FACTORS
  CALL CALSOLR(X,XMU)
  CALL SUNWRIT(X,XMU)

END
```

```

SUBROUTINE SUNREAD(GS,XINC,OMEGA,WOMEGA,XNU)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C
C GS(I) GLOBAL SUN VECTOR
C XINC INCLINATION OF THE ORBIT (DEG)
C OMEGA LONGITUDE OF ASCENDING NODE (DEG)
C WOMEGA LINE OF PERIAPSIS (DEG)
C XNU TRUE ANOMALY AT EPOCH (DEG)
C
  DIMENSION GS(3)

  DTR = 1.7453293E-2

  OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

  READ(3,*) GS(1)
  READ(3,*) GS(2)
  READ(3,*) GS(3)
  READ(3,*) XINC
  READ(3,*) OMEGA
  READ(3,*) WOMEGA
  READ(3,*) XNU

C CONVERT TO RADIANs
  XINC = XINC*DTR
  OMEGA = OMEGA*DTR
  WOMEGA = WOMEGA*DTR
  XNU = XNU*DTR

  CLOSE(3)
  RETURN
END
```



```

SUBROUTINE SUNWRIT(X,XMU)
  IMPLICIT REAL*8 (A-H, O-Z)
  C
  C WRITES THE THREE COMPONENTS OF THE SUN VECTOR
  C IN LOCAL COORDINATES AND THE 6 SOLAR VIEW FACTORS
  C
  DIMENSION X(3)
  DIMENSION XMU(6)

  OPEN(10,FILE='RETURN.DAT',STATUS='UNKNOWN')

  WRITE(10,*) 'XS':X(1)
  WRITE(10,*) 'YS':X(2)
  WRITE(10,*) 'ZS':X(3)

  WRITE(10,*) 'MU1':XMU(1)
  WRITE(10,*) 'MU2':XMU(2)
  WRITE(10,*) 'MU3':XMU(3)
  WRITE(10,*) 'MU4':XMU(4)
  WRITE(10,*) 'MU5':XMU(5)
  WRITE(10,*) 'MU6':XMU(6)

  CLOSE(10)

  RETURN
  END

```

**CALSUN.FOR**

```

SUBROUTINE CALSUN(G,XINC,OMEGA,WOMEGA,XNU,XS)
  IMPLICIT REAL*8 (A-H, O-Z)
  DIMENSION XS(3), A(3,3), B(3,3), G(3), GS(3), C(3,3)
  DIMENSION IPVT(3)

C ROTATES THE GLOBAL SUNVECTOR INTO LOCAL COORDINATES
C SOLVING THE EQUATION Ax = b
C A IS THE COEFFICIENT MATRIX FROM PERIFOCAL TO GLOBAL
C B IS THE COEFFICIENT MATRIX FROM PERIFOCAL TO SPACECRAFT
C X IS THE LOCAL SUN VECTOR
C G IS THE GLOBAL SUN VECTOR

  LDA = 3
  N = 3
  INFO = 0
  JOB = 0

  DO 10 I=1,3
    GS(I) = G(I)
  10 CONTINUE

C DEFINE THE ROTATION MATRIX
  CALL POR2IJK(XINC,OMEGA,WOMEGA,XNU,A,B)
  CALL MATMUL(3,3,3,B,A,C)

C SOLVE Ax = b
  CALL DGEFA(C,LDA,N,IPVT,INFO)
  CALL DGESL(C,LDA,N,IPVT,GS,JOB)

C IF THE VALUE IS LESS THAN 1.0E-3 IT WILL BE WRITTEN
C IN SCIENTIFIC NOTATION, WHICH CANNOT BE READ BY EXSYS*
  DO 20 I=1,3
    IF (DABS(GS(I)) .LT. 1.0E-3) THEN
      XS(I) = 0.0
    ELSE
      XS(I) = GS(I)
    ENDIF
  20 CONTINUE

  RETURN
END

```

```

SUBROUTINE POR2IJK(XI,O,W,XN,A,B)
  IMPLICIT REAL*8 (A-H, O-Z)
C THIS SUBROUTINE FILLS THE COEFFICIENT MATRIX
C FOR THE ROTATION FROM PERIFOCAL SYSTEM (P,O,R) TO THE
C GLOBAL SYSTEM (I,J,K)
C SEE DOCUMENTATION FOR DERIVATION
  DIMENSION A(3,3), B(3,3)

  SXI = SIN(XI)
  SW = SIN(W)
  SO = SIN(O)
  SN = SIN(XN)
  CXI = COS(XI)
  CW = COS(W)
  CO = COS(O)
  CN = COS(XN)

  A(1,1) = CO*CW - SO*SW*SXI
  A(1,2) = -CO*SW - SO*CW*CXI
  A(1,3) = -(SO*SXI)
  A(2,1) = SO*CW + CO*SW*CXI
  A(2,2) = -SO*SW + CO*CW*CXI
  A(2,3) = -(-CO*SXI)
  A(3,1) = -(SW*SXI)
  A(3,2) = -(CW*SXI)
  A(3,3) = CXI

  B(1,1) = CN
  B(1,2) = SN
  B(1,3) = 0
  B(2,1) = -SN
  B(2,2) = CN
  B(2,3) = 0
  B(3,1) = 0
  B(3,2) = 0
  B(3,3) = 1

  RETURN
END

```

## MATMUL.FOR

```
SUBROUTINE MATMUL(N,M,L,A,B,C)
  IMPLICIT REAL*8 (A-H,O-Z)

C MULTIPLIES TWO MATRICES
C A IS DIMENSIONED N,M
C B IS DIMENSIONED M,L
C C IS DIMENSIONED N,L

  DIMENSION A(N,M), B(M,L), C(N,L)

  DO 10 I = 1,L
    DO 20 J = 1,N
      D = 0.0
      DO 30 K = 1,M
        D = A(J,K)*B(K,I) + D
30      CONTINUE
      C(J,I) = D
20    CONTINUE
10  CONTINUE

  RETURN
  END
```

## MATADD.FOR

```
SUBROUTINE MATADD(N,M,A,B,C,IFAC)
  IMPLICIT REAL*8 (A-H,O-Z)
C ADDS TWO MATRICES
C ALL THREE MATRICES ARE DIMENSIONED NXM

C IF IFAC = -1 THEN SUBTRACTION
C IF IFAC = 1 THEN ADDITION

  DIMENSION A(N,M), B(N,M), C(N,M)

  DO 10 I = 1,N
    DO 20 J = 1,M
      C(I,J) = A(I,J) + IFAC*B(I,J)
20    CONTINUE
10  CONTINUE

  RETURN
  END
```

**POSGLOB.FOR**

```
      SUBROUTINE POSGLOB(ECC,SMA,XI,O,W,XNU,RVECTG)
      IMPLICIT REAL*8 (A-H, O-Z)
      C CALCULATES THE ORBITAL POSITION IN GLOBAL COORDINATES
      DIMENSION RVECT(3),RVECTG(3)
      DIMENSION A(3,3), B(3,3)
      C ECC ECCENTRICITY
      C SMA SEMI-MAJOR AXIS (KM)
      C XI ORBIT INCLINATION (RAD)
      C O LONGITUDE OF ASCENDING NODES (RAD)
      C W LINE OF PERIAPSIS (RAD)
      C XNU TRUE ANOMALY AT EPOCH (RAD)
      C
      C RVECTG GLOBAL POSITION VECTOR

      C ORBPROPS CALCULATES THE POSITION VECTOR MAGNITUDE R
      CALL ORBPROPS(ECC,SMA,XN,R)
      C CALCULATES THE POSITION VECTOR RVECT BY PROJECTING INTO
      C THE PERIFOCAL COORDINATE SYSTEM
      RVECT(1) = R*COS(XN)
      RVECT(2) = R*SIN(XN)
      RVECT(3) = 0.0
      C CALCULATES THE TRANSFORMATION MATRIX FOR LOCAL TO
      C GLOBAL AND CALCULATES THE GLOBAL POSITION VECTOR
      CALL POR2IJK(XI,O,W,XNU,A,B)
      CALL MATMUL(3,3,1,A,RVECT,RVECTG)

      RETURN
      END
```

## ORBSUBS.FOR

```
SUBROUTINE ORBCOMP(T,Q,VO,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW,  
+ VOHIGH,VOLOW,QHIGH,QLOW,SOHIGH,SOLOW,VOQHIG,VOQLOW,TDELTA,  
+ QDELTA,VODEL,VOODEL)  
IMPLICIT REAL*8 (A-H,O-Z)
```

```
C CALCULATES THE ORBITAL AVERAGE VALUES FOR TEMPERATURE  
C AND HEAT LOADS.
```

```
C  
C T(6) CURRENT TEMPERATURE  
C Q(6) CURRENT HEAT LOAD  
C VO TRUE ANOMOLY  
C TAVE(6) AVERAGE TEMPERATURE  
C QAVE(6) AVERAGE HEAT LOAD  
C  
C THIGH HIGHEST TEMPERATURE  
C TLOW LOWEST TEMPERATURE  
C SHIGH SIDE WITH THIGH  
C SLOW SIDE WITH TLOW  
C VOHIGH VO FOR THIGH  
C VOLOW VO FOR TLOW  
C  
C OHIGH HIGHEST HEAT LOAD  
C OLOW LOWEST HEAT LOAD  
C SOHIGH SIDE WITH QHIGH  
C SOLOW SIDE WITH QLOW  
C VOQHIG VO FOR QHIGH  
C VOQLOW VO FOR QLOW  
C  
C TDELTA HIGHEST TEMPERATURE DELTA BETWEEN ANY TWO SIDES  
C QDELTA HIGHEST HEAT LOAD DELTA BETWEEN ANY TWO SIDES  
C VODEL VO FOR TDELTA  
C VOODEL VO FOR QDELTA
```

```
C TAVE AND QAVE ARE THE SUM OF ALL TEMPERATURES AND HEAT  
C LOADS DIVIDED BY THE NUMBER OF POINTS FOR EACH SIDE.
```

```
DIMENSION T(6), Q(6)  
DIMENSION TAVE(6),QAVE(6)
```

```
DO 10 I=1,6  
TAVE(I) = TAVE(I) + T(I)  
QAVE(I) = QAVE(I) + Q(I)
```

## FORTRAN Source Code

```
10 CONTINUE
```

```
C INITIALIZE THE HIGHEST T AND Q AS 0  
C LOWEST T AND Q AS 1000.0
```

```
TH = 0.0  
TL = 1000.0  
QH = 0.0  
QL = 1000.0
```

```
C IF THE TEMPERATURE IS HIGHER THAN THE CURRENT HIGHEST  
C TEMP, SAVE THE VALUE AS THE CURRENT HIGH.
```

```
DO 20 I=1,6  
IF (T(I).GT.TH) THEN  
TH = T(I)  
ENDIF  
IF (T(I).LT.TL) THEN  
TL = T(I)  
ENDIF
```

```
C IF THE ABSOLUTE HEAT LOAD IS HIGHER THAN THE CURRENT HIGHEST  
C ABSOLUTE HEAT LOAD, SAVE THE VALUE AS THE CURRENT HIGH.
```

```
IF (ABS(Q(I)).GT.ABS(QH)) THEN  
QH = Q(I)  
ENDIF  
IF (ABS(Q(I)).LT.ABS(QL)) THEN  
QL = Q(I)  
ENDIF
```

```
20 CONTINUE
```

```
C IF DELTA IS LESS THAN THE DIFFERENCE BETWEEN THE HIGH AND  
C LOW, SAVE THE NEW DELTA AND SAVE THE POSITION
```

```
IF (TDELTA.LT.(TH-TL)) THEN  
TDELTA = TH-TL  
VODEL = VO  
ENDIF  
IF (QDELTA.LT.ABS(QH-QL)) THEN  
QDELTA = QH-QL  
VOODEL = VO  
ENDIF
```

```
C CHECK EACH SIDE FOR A HIGHER OR LOWER TEMP THAN THE CURRENT  
C HIGHEST AND LOWEST.
```

```
DO 30 I=1,6  
IF (T(I).GT.THIGH) THEN  
THIGH = T(I)
```

## FORTRAN Source Code

G28

```

      SHIGH = 1
      VOHIGH = VO
    ENDIF
    IF (I(I).LT.TLOW) THEN
      TLOW = I(I)
      SLOW = 1
      VOLOW = VO
    ENDIF
    IF (ABS(O(I)).GT.ABS(OHIGH)) THEN
      OHIGH = O(I)
      SOHIGH = 1
      VOOHIGH = VO
    ENDIF
    IF (ABS(O(I)).LT.ABS(OLOW)) THEN
      OLOW = O(I)
      SOLOW = 1
      VOLOW = VO
    ENDIF
30 CONTINUE

    RETURN
  END

```

```

      SUBROUTINE ORBFIN(NUMPTS,TAVET,QAVET,TAVE,QAVE)
C FINISHES THE ORBITAL CALCULATIONS
      IMPLICIT REAL*8 (A-H, O-Z)
      DIMENSION TAVE(6),QAVE(6)
C NUMPTS NUMBER OF ORBIT POINTS CALCULATED
C TAVET TOTAL AVERAGE TEMPERATURE
C QAVET TOTAL AVERAGE HEAT LOAD
C TAVE(6) AVERAGE TEMPERATURE FOR EACH OF THE SIX SIDES
C QAVE(6) AVERAGE HEAT LOAD FOR EACH OF THE SIX SIDES

      DO 10 I=1,6
        TAVE(I) = TAVE(I)/NUMPTS
        QAVE(I) = TAVE(I)/NUMPTS
10 CONTINUE

      TAVET = 0.0
      QAVET = 0.0
      DO 20 I=1,6
        TAVET = TAVET + TAVE(I)
        QAVET = QAVET + QAVE(I)
20 CONTINUE
      TAVET = TAVET/6
      QAVET = QAVET/6

      RETURN
    END

```

```

SUBROUTINE ORBIT(SIGMA,XRAD,XRADI,FEB,TAVE,QAVE,TDELTA,
+             QDELTA,THIGH,QHIGH,TLOW,QLOW)
IMPLICIT REAL*8(A-H,O-Z)
C
C INITIALIZES THE CONSTANTS AND SETS THE MATRICES TO 0.0
C
  DIMENSION FEB(6,6), XRAD(6,6), XRADI(6,6)
  DIMENSION TAVE(6), QAVE(6)

  SIGMA = 5.67E-8

  THIGH = 0.0
  QHIGH = 0.0
  TLOW = 1000.0
  QLOW = 1000.0
  TDELTA = 0.0
  QDELTA = 0.0

  DO 10 I=1,6
    DO 20 J=1,6
      XRAD(I,J) = 0.0
      XRADI(I,J) = 0.0
      FEB(I,J) = 0.0
20  CONTINUE

  TAVE(I) = 0.0
  QAVE(I) = 0.0
10 CONTINUE

  RETURN
  END

```

```

SUBROUTINE ORBREAD(ALBEDO,S,E,GS,XI,O,WO,ECC,SMA,INERT,X,Y,Z,
+             RE,ALPIR,ALPS,EPS,NUMINS,INSUL,NUMPTS)
IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C
C ALBEDO THE ALBEDO COEFFICIENT (USUALLY .3)
C GS THE GLOBAL SUN VECTOR
C INC ORBIT INCLINATION (DEG)
C OMEGA LONGITUDE OF ASCENDING NODES (DEG)
C WOMEGA LINE OF PERIAPSIS (DEG)
C ECC ECCENTRICITY
C SMA SEMI-MAJOR AXIS
C X THE X DIMENSION OF THE RECTANGULAR BOX
C Y THE Y DIMENSION OF THE RECTANGULAR BOX
C Z THE Z DIMENSION OF THE RECTANGULAR BOX
C ALPIR THE INFRARED ABSORPTIVITY = EMISSIVITY
C INERT INERTIAL FLAG 1-INERTIAL 0-PLANET
C ALPS THE SOLAR ABSORPTIVITY
C NUMINS THE NUMBER OF INSULATED SIDES
C INS THE SIDES THAT ARE INSULATED
C NUMPTS NUMBER OF ORBIT POINTS
C
  DIMENSION ALPIR(6),ALPS(6),EPS(6),INSUL(5),GS(3)

  DTR = 1.7453292E-2

  OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

  READ(3,*) ALBEDO
  READ(3,*) S
  READ(3,*) E

  DO 40 I=1,3
    READ(3,*) GS(I)
40 CONTINUE

  READ(3,*) XI
  READ(3,*) O
  READ(3,*) WO
  READ(3,*) ECC
  READ(3,*) SMA

  XI = XI*DTR

```

## FORTRAN Source Code

G30

```

O = O*DTR
WO = WO*DTR

READ(3,*) INERT

READ(3,*) X
READ(3,*) Y
READ(3,*) Z
READ(3,*) RE

DO 20 I=1,6
  READ(3,*) ALPIR(I)
  EPS(I) = ALPIR(I)
20 CONTINUE

DO 30 I=1,6
  READ(3,*) ALPS(I)
30 CONTINUE

READ(3,*) NUMINS
DO 60 I=1,5
  READ(3,*) INSUL(I)
60 CONTINUE

READ(3,*) NUMPTS

CLOSE(3)
RETURN
END

```

```

SUBROUTINE ORBOUT(O,T,XI,O,W,XN)
IMPLICIT REAL*8 (A-H, O-Z)

C
C WRITES TEMPERATURES AND HEAT LOAD
C TO "RAD-ORB.DAT"
C
  DIMENSION O(6), T(6)

  RTD = 57.29577951
  X = XI *RTD
  XO = O *RTD
  WO = W *RTD
  XNU = XN *RTD

  WRITE(10,*) '[INC]', X
  WRITE(10,*) '[OMEGA]', XO
  WRITE(10,*) '[WOMEGA]', WO
  WRITE(10,*) '[NUO]', XNU
  WRITE(10,*)

  WRITE(10,*) '[T1]',T(1)
  WRITE(10,*) '[T2]',T(2)
  WRITE(10,*) '[T3]',T(3)
  WRITE(10,*) '[T4]',T(4)
  WRITE(10,*) '[T5]',T(5)
  WRITE(10,*) '[T6]',T(6)
  WRITE(10,*)

  DO 10 I=1,6
    IF (ABS(O(I)) .LT. 1.0E-6) THEN
      O(I) = 0.0
    ENDIF
  10 CONTINUE

  WRITE(10,*) '[O1]',O(1)
  WRITE(10,*) '[O2]',O(2)
  WRITE(10,*) '[O3]',O(3)
  WRITE(10,*) '[O4]',O(4)
  WRITE(10,*) '[O5]',O(5)
  WRITE(10,*) '[O6]',O(6)
  WRITE(10,*)

C
  RETURN
END

```



```

SUBROUTINE ORBWRT(TAVET,QAVET,TAVE,QAVE,THIGH,TLOW,SHIGH,SLOW,
+ VOHIGH,VOLOW,QHIGH,QLLOW,SOHIGH,SLOW,VOOHIGH,VOLOW,TDELTA,
+ QDELTA,VODELTA,VOODEL)
IMPLICIT REAL*8 (A-H, O-Z)
DIMENSION TAVE(6),QAVE(6)

```

C WRITES ALL THE FINAL STATISTICAL ANALYSIS TO RETURN.DAT

```
OPEN(11,FILE='RETURN.DAT',STATUS='UNKNOWN')
```

```
RTD = 57.29577951
```

```
WRITE(11,*) '[TAVET]',TAVET
WRITE(11,*) '[QAVET]',QAVET
WRITE(11,*)
```

```
WRITE(11,*) '[THIGH]',THIGH
WRITE(11,*) '[SHIGH]',SHIGH
WRITE(11,*) '[VOHIGH]',VOHIGH*RTD
WRITE(11,*)
```

```
WRITE(11,*) '[TLOW]',TLOW
WRITE(11,*) '[SLOW]',SLOW
WRITE(11,*) '[VOLOW]',VOLOW*RTD
WRITE(11,*)
```

```
WRITE(11,*) '[QHIGH]',QHIGH
WRITE(11,*) '[SOHIGH]',SOHIGH
WRITE(11,*) '[VOOHIGH]',VOOHIGH*RTD
WRITE(11,*)
```

```
WRITE(11,*) '[QLOW]',QLOW
WRITE(11,*) '[SLOW]',SLOW
WRITE(11,*) '[VOLOW]',VOLOW*RTD
WRITE(11,*)
```

```
WRITE(11,*) '[TDELTA]',TDELTA
WRITE(11,*) '[VODELTA]',VODELTA*RTD
WRITE(11,*) '[QDELTA]',QDELTA
WRITE(11,*) '[VOODEL]',VOODEL*RTD
WRITE(11,*)
```

```
WRITE(11,*) '[T1]',TAVE(1)
WRITE(11,*) '[T2]',TAVE(2)
WRITE(11,*) '[T3]',TAVE(3)
```

```
WRITE(11,*) '[T4]',TAVE(4)
WRITE(11,*) '[T5]',TAVE(5)
WRITE(11,*) '[T6]',TAVE(6)
WRITE(11,*)
```

```
DO 10 I=1,6
  IF (ABS(QAVE(I)) .LT. 1.0E-6) THEN
    QAVE(I) = 0.0
  ENDF
10 CONTINUE
```

```
WRITE(11,*) '[Q1]',QAVE(1)
WRITE(11,*) '[Q2]',QAVE(2)
WRITE(11,*) '[Q3]',QAVE(3)
WRITE(11,*) '[Q4]',QAVE(4)
WRITE(11,*) '[Q5]',QAVE(5)
WRITE(11,*) '[Q6]',QAVE(6)
WRITE(11,*)
```

C  
C WRITE TO DATA FILE

```
WRITE(10,*) '[TAVET]',TAVET
WRITE(10,*) '[QAVET]',QAVET
WRITE(10,*)
```

```
WRITE(10,*) '[TAVE 1]',TAVE(1)
WRITE(10,*) '[TAVE 2]',TAVE(2)
WRITE(10,*) '[TAVE 3]',TAVE(3)
WRITE(10,*) '[TAVE 4]',TAVE(4)
WRITE(10,*) '[TAVE 5]',TAVE(5)
WRITE(10,*) '[TAVE 6]',TAVE(6)
WRITE(10,*)
```

```
WRITE(10,*) '[QAVE 1]',QAVE(1)
WRITE(10,*) '[QAVE 2]',QAVE(2)
WRITE(10,*) '[QAVE 3]',QAVE(3)
WRITE(10,*) '[QAVE 4]',QAVE(4)
WRITE(10,*) '[QAVE 5]',QAVE(5)
WRITE(10,*) '[QAVE 6]',QAVE(6)
WRITE(10,*)
```

```
WRITE(10,*) '[THIGH]',THIGH
WRITE(10,*) '[SHIGH]',SHIGH
```

## FORTRAN Source Code

G32

```
WRITE(10,*) '(VOHIGH)',VOHIGH*RTD  
WRITE(10,*) '(TLOW)',TLOW  
WRITE(10,*) '(SLOW)',SLOW  
WRITE(10,*) '(VOLOW)',VOLOW*RTD  
WRITE(10,*)
```

```
WRITE(10,*) '(QHIG)',QHIG  
WRITE(10,*) '(SQHIG)',SQHIG  
WRITE(10,*) '(VOOHIG)',VOOHIG*RTD  
WRITE(10,*) '(QLOW)',QLOW  
WRITE(10,*) '(SLOW)',SLOW  
WRITE(10,*) '(VOLOW)',VOLOW*RTD  
WRITE(10,*)
```

```
WRITE(10,*) '(TDELTA)',TDELTA  
WRITE(10,*) '(VDELTA)',VDELTA  
WRITE(10,*) '(DELTA)',DELTA  
WRITE(10,*) '(VDELTA)',VDELTA*RTD
```

```
CLOSE(11)  
RETURN  
END
```

## ORBPROPS.FOR

```
SUBROUTINE ORBPROPS(ECC,SMA,XNU,R)
  IMPLICIT REAL*8 (A-H, O-Z)
  C CALCULATES THE ORBITAL RADIUS
  C TAKEN FROM "FUNDAMENTALS OF ASTRODYNAMICS" BY
  C BATE, MUELLER AND WHITE
```

```
  XN = XNU * 0.017453293
```

```
  SLR = SMA*(1-ECC+ECC)
  R = SLR/(1+ECC*COS(XN))
```

```
  RETURN
  END
```

## DOTPROD.FOR

```
FUNCTION DOTPROD(X,Y)
  IMPLICIT REAL*8 (A-H, O-Z)
  C CALCULATES THE DOT PRODUCT OF TWO VECTORS X AND Y
```

```
  DIMENSION Y(3),X(3)
```

```
  Z = 0
```

```
  DO 10 I=1,3
```

```
    Z = X(I)*Y(I) + Z
```

```
  10 CONTINUE
```

```
  DOTPROD = Z
```

```
  RETURN
  END
```

## FORTRAN Source Code

G34

### FETANOR.FOR

```
PROGRAM FETANOR
  IMPLICIT REAL*8 (A-H, O-Z)

C THIS PROGRAM CALCULATES THE EARTH VIEW FACTORS
C NORMAL AND TANGENTIAL PLATES. IT IS USED IN
C CONJUNCTION WITH THE EXPERT SYSTEM RADIATE.RUL
C RACHEL ORMSBY July 29, 1993
C
C READS THE INPUT DATA
C ECC ECCENTRICITY
C SMA SEMI-MAJOR AXIS (KM)
C XNU TRUE ANOMALY AT EPOCH (RAD)
  CALL FEREAD(ECC,SMA,XNU,RE)

C CALCULATES THE EARTH VIEW FACTOR FOR
C A PLANE NORMAL TO THE EARTH
  CALL CALVFNOR(ECC,SMA,XNU,RE,FEN)
C CALCULATES THE EARTH VIEW FACTOR FOR
C A PLANE TANGENTIAL TO THE EARTH
  CALL CALVFTAN(ECC,SMA,XNU,RE,FET)

  CALL FEWRIT(FET,FEN)

END
```

```
SUBROUTINE FEREAD(ECC,SMA,XNU,RE)
  IMPLICIT REAL*8 (A-H, O-Z)
C
C READS THE INPUT
C ECC ECCENTRICITY
C SMA SEMI-MAJOR AXIS
C
  OPEN(3,FILE='RADIATE.DAT',STATUS='OLD')

  DTR = 1.7453293E-2

  READ(3,*) ECC
  READ(3,*) SMA
  READ(3,*) XNU
  XNU = -XNU*DTR
  READ(3,*) RE

  CLOSE(3)
  RETURN
END
```

```
SUBROUTINE FEWRIT(FET,FEN)
  IMPLICIT REAL*8 (A-H, O-Z)
  C
  C WRITES NORMAL AND TANGENTIAL
  C VIEW FACTORS TO "RETURN.DAT"
  C
  OPEN(10,FILE='RETURN.DAT',STATUS='UNKNOWN')

  WRITE(10,*) 'FET',FET
  WRITE(10,*) 'FEN',FEN

  CLOSE(10)
  RETURN
  END
```

# FORTRAN Source Code

G36

## IJK2PQR.FOR

SUBROUTINE IJK2PQR(XI,O,W,XN,A,B)  
 IMPLICIT REAL\*8 (A-H, O-Z)

C  
 C \*\*\*\* THIS SUBROUTINE IS CURRENTLY NOT USED, BUT  
 C LEFT HERE FOR FUTURE REFERENCE \*\*\*\*  
 C  
 C THIS SUBROUTINE FILLS THE COEFFICIENT MATRIX  
 C THIS CONVERSION WAS TAKEN FROM "FUNDAMENTALS  
 C OF ASTRODYNAMICS" BY BATE, MUELLER AND WHITE  
 C  
 C XII INCLINATION (RAD)  
 C OI LONGITUDE OF ASCENDING NODE (RAD)  
 C WI LINE OF PERIAPSIS (RAD)  
 C XNU TRUE ANOMOLY AT EPOCH (RAD)  
 C A TRANSFORMATION MATRIX FROM GLOBAL TO PERIFOCAL  
 C B TRANSFORMATION MATRIX FROM PERIFOCAL TO  
 C SPACECRAFT

DIMENSION A(3,3), B(3,3)

SXI = SIN(XI)  
 SW = SIN(W)  
 SO = SIN(O)  
 SN = SIN(XN)  
 CXI = COS(XI)  
 CW = COS(W)  
 CO = COS(O)  
 CN = COS(XN)

A(1,1) = SXI+SW  
 A(1,2) = CXI\*CXI\*CW\*SO + CW\*SXI\*SXI\*SO +  
 CXI\*CO\*SW  
 A(1,3) = -(CXI\*CXI\*CW\*CO + CW\*SXI\*SXI\*CO -  
 CXI\*SO\*SW)  
 A(2,1) = CW\*SXI  
 A(2,2) = CXI\*CO\*CW - CXI\*CXI\*SO\*SW -  
 SXI\*SXI\*SO\*SW  
 A(2,3) = -(-CXI\*CW\*SO - CXI\*CXI\*CO\*SW -  
 CO\*SXI\*SXI\*SW)  
 A(3,1) = -CXI  
 A(3,2) = -(-CO\*SXI)  
 A(3,3) = SXI\*SO

B(1,1) = CN  
 B(1,2) = SN  
 B(1,3) = 0  
 B(2,1) = -SN  
 B(2,2) = CN  
 B(2,3) = 0  
 B(3,1) = 0  
 B(3,2) = 0  
 B(3,3) = 1

RETURN  
 END

32

## **MAG.FOR**

```
FUNCTION XMAG(X)  
  IMPLICIT REAL*8 (A-H, O-Z)  
  C CALCULATES THE MAGNITUDE OF THE VECTOR X
```

```
  DIMENSION X(3)
```

```
  A = X(1)*X(1)
```

```
  B = X(2)*X(2)
```

```
  C = X(3)*X(3)
```

```
  D = A+B+C
```

```
  XMAG = SQRT(D)
```

```
  RETURN
```

```
  END
```

## FORTRAN Source Code

G38

### SHADOW.FOR

```
      SUBROUTINE SHADOW(ECC,XNU,XINC,OMEGA,WOMEGA,GS,SORIG,S,XMU,RE)
C DETERMINES IF THE SPACECRAFT IS IN THE SHADOW OF THE EARTH
C IF IN THE SHADOWS:
C   RETURNS THE SOLAR FLUX AS 0.0
C   AND THE SOLAR VIEW FACTORS AS 0.0
C IF NOT
C   RETURNS THE ORIGINAL VALUE FOR SOLAR FLUX
C
C ECC   ECCENTRICITY
C SMA   SEMI-MAJOR AXIS (KM)
C XNU   TRUE ANOMALY AT EPOCH (DEG)
C XINC  INCLINATION (DEG)
C OMEGA ORBIT LONGITUDE OF ASCENDING NODE (DEG)
C WOMEGA LINE OF PERIAPSIS (DEG)
C GS(3) GLOBAL SUN VECTOR
C SORIG ORIGINAL NON ZERO SOLAR FLUX (W/m-2)
C S     CURRENT SOLAR FLUX (W/m-2)
C       = SORIG NOT IN SHADOW
C       0   IN SHADOW
C XMU(6) SOLAR VIEW FACTORS
C RE    RADIUS OF THE PLANET
C
      IMPLICIT REAL*8 (A-H, O-Z)

      DIMENSION GS(3), ORIGIN(3)
      DIMENSION V(3), XMU(6)
      DIMENSION POS(3)

C CALCULATES THE CURRENT SPACECRAFT POSITION IN LOCAL COORDINATES
      CALL POSGLOB(ECC,SMA,XINC,OMEGA,WOMEGA,XNU,POS)

C THE ORIGIN IS CURRENTLY DEFINED AT (0,0,0)
      ORIGIN(1) = 0.0
      ORIGIN(2) = 0.0
      ORIGIN(3) = 0.0

C MATADD SUBTRACTS THE ORIGIN FROM THE POS VECTOR GIVING
C A CORRECTED POSITION (V)
      CALL MATADD(3,1,POS,ORIGIN,V,-1)
C DISTANCE FROM THE POSITION TO THE VERTICAL SHADOW PLANE
      D1 = DOTPROD(V,GS)
```

```
      C XMAG CALCULATES THE MAGNITUDE OF A VECTOR
      XMAGV = XMAG(V)

C DETERMINES IF THE SPACECRAFT FALLS WITHIN THE SHADOW
C DETERMINED BY THE CYLINDRICAL SHADOW PLANE
      D2 = SQRT(XMAGV*XMAGV - D1*D1)

      IF (D2.LT.RE .AND. D1.LT.0) THEN
          S = 0.0
          DO 10 I=1,6
              XMU(I) = 0.0
10      CONTINUE
      ELSE
          S = SORIG
      ENDIF

      RETURN
      END
```



## References

1. Agrawal, Brij N. (1986). *Design of Geosynchronous Spacecraft* . International Telecommunications Satellite Organization (INTELSAT), Washington, DC; Prentice-Hall, Inc
2. Bate, Roger R., Mueller, Donald D., White, Jerry E. (1971). *Fundamentals of Astrodynamics* United States Air Force Academy, Dover Publications
3. Ignizio, James P. (1991). *Introduction to Expert Systems; The Development and Implementation of Rule-based Expert Systems*, The University of Houston; McGraw-Hill, Inc
4. Jackson, Peter (1990), *Introduction to Expert Systems*, second edition. McDonnell Douglas Research Laboratories, Saint Louis, Missouri; Addison-Wesley Publishing Company, Inc.
5. Rich, Elaine and Knight, Kevin (1991). *Artificial Intelligence*, second edition. Microelectronics and Computer Technology Corporation and Carnegie Mellon University; McGraw-Hill, Inc.
6. Siegel, Robert and Howell, John R. (1982) *Thermal Radiation Heat Transfer*, second edition, Fluid Mechanics and Acoustics Division, NASA Lewis Research Center and University of Texas at Austin; Hemisphere Publishing Corporation, New York

## Bibliography

1. Antreasian, P. G. and Rosborough, G.W. (1992), "Prediction of Radiant Energy Forces on the TOPEX/POSEIDON Spacecraft". University of Colorado, Boulder, Vol 29, No1, January-February, Journal of Spacecraft and Rockets: American Institute of Aeronautics and Astronautics
2. Bird, R. Byron, Stewart. Warren E. and Lightfoot, Edwin N. (1960). *Transport Phenomena*. Department of Chemical Engineering, University of Wisconsin, Madison, Wisconsin; John Wiley & Sons, Inc
3. Forsyth, Richard (1989), *Expert Systems Principles and Case Studies*, second edition, Chapman and Hall Computing
4. Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. The University of Alabama; Addison-Wesley Publishing Company, Inc
5. Kraus, Allan D. and Bar-Cohen, Avram (1983), *Thermal Analysis and Control of Electronic Equipment*. University of South Florida, Tampa, and Ben-Gurion University of the Negev; Hemisphere Publishing Corporation and McGraw-Hill Book Company.
6. Sawyer, Brian and Foster, Dennis L., (1986) *Programming Expert Systems in Pascal* , John Wiley & Sons, Inc.

## Index

albedo flux	A18
backward chaining	7, 8
beginning of life	3-5, 19, 20, A28, A29, A31, A32, B3, C8, E12
child	19
coordinate	29, 30, 35-37, 39, G19, G21, G26
coordinate transformations	35
decoupled sides	C31, D4, G5
default data files	41, 43, 45, D1, D3
description screen	43
earth flux	viii, 12
enclosure analysis	i, 13, 47, 55, A21, A24, C39, E4, E8, F3, F4
enclosure radiation	A14, A20, B3, B4, C4, C15, D4
enclosure temperature and heat loads	18
enclosure view factor	17, G9
end of life	3, 5, 19, 20, A28, A29, A31, A32, B3, C8
EXDESIGN	43
expert system shell	1
EXSYS	i, ii, 1, 10, 41, 43, 47, 48, A10, A18, G21, G24
FORTRAN code	47, A21
FORTRAN program	F1
forward chaining	7
frame	4, 5, 19-21, 23, 24, 49-52, A28, A29, A31, A32, A33
geocentric coordinates	35, 36
global coordinates	35, 36, 39, 40, A22, A23, C22, F5, G26
global sun and planet vector	A22, A23, B3
gray body	12, 13, 19
heat balance	11, 14, 15
heat load	viii, 11-15, 17, 23, A4, A6, A9, A11, A12, A13, A14, C15, C28, C33, C35, C42, E6, E10, F3, F4, F5, G4, G6, G8, G10, G11, G20, G27, G28, G30
how?	43
inference engine	3-10, 51, 52, A4, A11, A12, A16, A17, A18, A19
infrared emissivity	19
internal heat load	11, 13, C28
knowledge base	3, 41
local sun and planet vector	A22, B3, C5
main menu	5, 6, 8, A4, A5, A6, A7, A8, A9, A10, A11, A12, A16, A17, A18, A19, A20, A21, A28
new feature	49
number of orbit points	i, C26, C43, G4, G28, G29
orbital elements	viii, 23, A22, A23, B3, C5
parent	19
performance time	3-5, A28, A29, A31, A32, B3, E12
perifocal coordinates	35, 37, 39
planet flux	23, 24, 44, 45, 49, 50, 53, A17, A33, C13, C33, C35, C37, D4, E6, E7, G7
planet view factor	12, 30, 53, A26, A27, C12, C21, C33, C36, C37, D4, E8, E9, E11, F4
position vector	39, 40, G26
radiation effects	A4, A6, A12, A13, A14, A16, A17, A18, A20, B3, B4, C4, E6, E7
radiative flux	11, 12, A19
rule base	3-5, 7, 41, A20, A21
shadow	39, 40, 55, F5, F9, F10, G5, G38

solar absorptivity	viii, 11, 19, A28, A31, C14, C45, D4, E8, E10, G4, G6, G7, G10, G29
solar and planet view factors	33, 52, A21
solar flux	viii, 11, 12, 23, 24, 44, 45, 50, 53, A16, A18, A33, C13, C33, C35, C37, D4, E6, E7, F5, G5, G7, G10, G38
solar inertial	33, A21, A23, F6, G15
solar view factors	27, 28, 30, 33, 35, 47, 55, A20, A22, A23, A24, B3, F5, G5, G10, G21, G22, G23, G38
spacecraft orbit	23, 27, 47, A26, A27, B3, C7, E8
spacecraft orientation	23, 47, A21, A23, A26, B3, C6, E8, E11
spacecraft position	33-35, 39, C20, F5, G5, G15, G17, G18, G38
Stefan-Boltzmann law	12
surface constants	49, 53, C12
surface properties	3-6, 11, 13, 19-21, 55, A8, A28, A29, A30, A31, A32, B3, B4, C3, C19, C40, C43, E4, E5, E8, E10
symbolic	3
user interface	21
view factors	16, 27, 28, 30, 31, 33, 35, 47, 52, 55, A20, A21, A22, A23, A24, A25, A26, A27, B3, B4, C11, C40, E4, E5, E8, E11, F3, F4, F5, F6, G4, G5, G6, G7, G9, G10, G11, G12, G13, G15, G16, G20, G21, G22, G23, G34, G35, G38
enclosure view factor	17, G9
planet view factor	12, 30, 53, A26, A27, C10, C19, C31, C34, C35, D4, E8, E9, E11, F4
solar and planet view factors	33, 52, A21
solar view factors	27, 28, 30, 33, 35, 47, 55, A20, A22, A23, A24, B3, F5, G5, G10, G21, G22, G23, G38

# REPORT DOCUMENTATION PAGE

Form Approved  
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

1. AGENCY USE ONLY (leave blank)	2. REPORT DATE July 1994	3. REPORT TYPE AND DATES COVERED Contractor Report	
4. TITLE AND SUBTITLE  Reference Manual for the Thermal Analyst's Help Desk Expert System		5. FUNDING NUMBERS  C NAS1-19000	
6. AUTHOR(S)  Rachel A. Ormsby		WU 477-50-00-01	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Lockheed Engineering and Sciences Company Langley Program Office 144 Research Drive, Hampton, VA 23666		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Nataional Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-0001		10. SPONSORING / MONITORING AGENCY REPORT NUMBER  NASA CR-194941	
11. SUPPLEMENTARY NOTES  Langley Technical Monitor: Roger A. Breckenridge			
12a. DISTRIBUTION / AVAILABILITY STATEMENT  Unclassified-Unlimited Subject Category 18		12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words)  This document provides technical information and programming guidance for the maintenance and future development of the Thermal Analyst's Help Desk. Help Desk is an expert system that operates within the EXSYSTEM expert system shell, and is used to determine first approximations of thermal capacity for spacecraft and instruments. The five analyses supported in Help Desk are: (1) surface area required for a radiating surface, (2) equilibrium temperature of a surface, (3) enclosure temperature and heat loads for a defined position in orbit, (4) enclosure temperature and heat loads over a complete orbit, and (5) selection of appropriate surface properties. The two geometries supported by Help Desk are a single flat plate and a rectangular box enclosure. The technical information includes the mathematical approach and analytical derivations used in the analyses such as: radiation heat balance, view factor calculation, and orbit determination with coordinate transformation. The programming guide for developers describes techniques for enhancement of Help Desk. Examples are provided showing the addition of new features, user interface development and enhancement, and external program interfaces.			
14. SUBJECT TERMS thermal analysis, first approximation, thermal design, spacecraft thermal capacity, flat plate, rectangular box enclosure, two-body, orbital, heat load, surface, radiator, radiation.		15. NUMBER OF PAGES 217	16. PRICE CODE A10
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT  UL