

# NASA Contractor Report 178251

(NASA-CR-178251) CARE 3 USER-FRIENDLY  
INTERFACE USER'S GUIDE (PRC Kentron, Inc.)  
88 p CSCI 09B

N87-19930

Unclass

63/61 43645

## CARE III USER FRIENDLY INTERFACE USER'S GUIDE

A.L. Martensen

PRC Kentron, Inc.  
Hampton, VA

Contract NAS1-18000  
January 1987

*Corrected Copy*

**NASA**

National Aeronautics and  
Space Administration

Langley Research Center  
Hampton, Virginia 23665-5225

## Preface

This document reflects recent changes to the CARE3MENU program. The program is written to be compatible with the Computer-Aided Reliability Estimation program (CARE III) Version VI. All NAMELIST format input files used with CARE III Version V are compatible with Version VI and can be edited and altered with this version of the CARE3MENU program.

The original CARE III User Friendly Interface User's Guide, Contractor Report 172608, was co-authored by C.L. Conrad, W.F. Ingogly, S.E. McBride, and J.L. Pierce of Research Triangle Institute. Changes to the CARE3MENU program necessitated the development of a new User's Guide. Because of the outstanding organization of the original document, this updated version maintains the same style and structure. Additionally, parts of this document appear exactly, word for word, as they did in the original. At this point, I would like to acknowledge the effort of the original authors and give credit to their work. Specifically, the following pages or sections contain 90% or more original User's Guide material:

- Introduction, page 1
- The Model Input Process, pages 7 and 8
- Fault Handling Models, pages 21 and 22
- Fault Handling Model Accuracy, page 29
- Example One explanation, page 52
- Example Two explanation, page 61
- Chapter 4 - CARE3MENU Error Messages
- Chapter 5 - CARE III Parameters and Value Ranges

## Table of Contents

Preface .....	i
Table of Contents .....	iii
Introduction .....	1
Additional Documents of Interest .....	1
Chapter 1 -- A CARE3MENU Overview .....	3
What CARE3MENU Does .....	3
Using the CARE3MENU Interface .....	5
Chapter 2 -- Entering Data .....	7
The Model Input Process .....	7
CARE3MENU Screens .....	8
Screen 1: Main Level Menu .....	10
Screen 2: Stage Description Input .....	12
Screens 3A, 5A, 12A, and 14A .....	16
Screens 3B, 5B, 12B, and 14B - Partial Review Screen Selection ..	18
Screen 4: Fault Handling Models (FHM) .....	20
Screen 6A: Fault Occurrence Models (FOM) .....	24
Screen 6B: Fault Occurrence Models (FOM) .....	26
Screen 7: Fault Handling Model Accuracy .....	28
Screen 8: Information Summary .....	30
Screen 9: Ready to Begin Failure Configuration Input .....	32
Screen 10: System Fault Tree Input .....	34
Screen 11: Critical Pairs Fault Tree Input .....	36
Screen 13: Output Control Options Input .....	40
Screen 15: Runtime Control Options Input .....	42
Screen 16: Model Input Complete .....	44
Screen 17: File Name Input .....	46
Screen 18: Exit Without Store .....	48
Screen 19: File Recovery .....	50
Chapter 3 -- Example CARE3MENU Sessions .....	52
Example One .....	52
Example Two .....	61
Chapter 4 -- CARE3MENU Error Messages .....	75
Chapter 5 -- CARE III Parameters and Range Values .....	81

PRECEDING PAGE BLANK NOT FILMED

## Introduction

CARE III predicts the unreliability of highly reliable reconfigurable fault-tolerant systems that include redundant computers or computer systems. CARE3MENU is a user-friendly interface used to create an input file for the CARE III program. The CARE3MENU interface has been designed to minimize user input errors. Although a CARE3MENU session may be successfully completed and all parameters may be within specified limits or ranges, the CARE III program is not guaranteed to produce meaningful results if the user incorrectly interprets the CARE III stochastic model. The reader is strongly encouraged to first read Reference 1, then study Reference 4 for pertinent CARE III Version VI changes.

The CARE3MENU User Guide provides complete information on how to create a CARE III model with the interface. The CARE3MENU interface runs under the VAX/VMS operating system. For an introduction to the VAX/VMS operating system, see the VAX/VMS Introduction to VAX/VMS.

The CARE3MENU User Interface was developed for the Langley Research Center. It was designed by Salvatore J. Bavuso (Langley Research Center), Paul Petersen (Kentron International), John Pierce (RTI), and Alan Roberts (Tesserract Systems). The design was primarily implemented by Alan Roberts with additional work done by Sam McBride and John Pierce. Version II changes and improvements were designed and implemented by Anna Martensen (PRC Kentron, Inc.).

### Additional Documents of Interest

#### CARE III:

1. S.J. Bavuso and P.L. Petersen, **CARE III Model Overview and User's Guide** (first revision), NASA Technical Memorandum 86404, April 1985.
2. S.J. Bavuso, J.E. Brunelle, and P.L. Petersen, **CARE III Hands-On Demonstration and Tutorial**, NASA Technical Memorandum 86404, April 1985.
3. S.J. Bavuso, **A User's View of CARE III**, Proceedings of the 1984 Annual Reliability and Maintainability Symposium, January 1984.
4. L.A. Bryant and J.J. Stiffler, **CARE III, Version 4 Enhancements**, NASA Contractor Report 177963, November 1985.

VAX/VMS:

5. VAX/VMS Introduction to VAX/VMS, Digital Equipment Corporation, Maynard, MA, September 1984.
7. VAX-11 Fortran Language Reference Manual, Digital Equipment Corporation, Maynard, MA, April 1982.
8. VAX-11 Fortran User's Guide, Digital Equipment Corporation, Maynard, MA, April 1982.

## Chapter 1 - A CARE3MENU Overview

### What CARE3MENU Does

The CARE III input file is not easily created with a text editor. Additionally, many of the variable names are not clearly related to the model values they represent. Therefore, mistakes are easily made when generating a CARE III input file with the text editor. The CARE3MENU program provides a user-friendly front end for the CARE III program which prompts the user for data. Detailed on-line help is available for most screen displays.

The CARE3MENU program has been extensively revised and improved. The original program required the user to completely describe the entire model in one session. Once the file was created, it could not be recalled and modified with CARE3MENU. The following improvements have been implemented in this version:

- 1) During a CARE3MENU session, an input file can be reviewed at any time. This capability allows the user to correct past mistakes, verify previous inputs, or change the model.
- 2) Partial files may be created during a session and finished in later sessions. For very large models, the ability to create the model through a series of CARE3MENU sessions has several distinct advantages:
  - a) The model can be periodically saved to assure that computer system failures will not destroy the entire input file.
  - b) Large models can be input incrementally as the user finds time to work the model.
  - c) Multiple users can supply information to the same file. For example, one user can define the system architecture and another can describe the fault handling characteristics of the system.
- 3) An existing model can be recalled and altered.

All CARE3MENU users should have a working knowledge of the CARE III program and input data. The references listed in this guide's Introduction provide information about CARE III.

## Using the CARE3MENU Interface

1. Verifying input: At the bottom of most screen menus, the user is asked to verify the input. Y accepts the input and moves to the next input or screen. N, carriage return (denoted in this paper by <cr>), or any other character other than Y will move the cursor to the top of the section so that input(s) can be corrected.
2. Continuing to the next section: After verifying the input (see 1.) the cursor returns to the first line of the menu. For cyclic menus, type **END** to continue to the next input section.
3. Default values: The default values presented on the menus may be selected by pressing <cr>. Another value is entered by typing the appropriate value.
4. Menu wrap around: The user moves through a menu by typing the return key. Direction arrows will NOT advance the cursor. To return to a value on the menu, cycle through the menu (through the "Verify Input (Y or N)" line) until the desired location is reached.
5. Correcting bad data: Erroneous data entered on the menu presently displayed may be corrected by advancing through the menu until the desired location is reached. Erroneous data on past screens may be altered by selecting either T,

**T - Total Review of the Model**



or P,

**P - Partial Review of the Model**  
between input sections. Selecting T will allow all data entered to that point to be displayed screen by screen. If the user selects P, the specific review sections may be selected. To advance through the screens, select C,

**C - Continue.**

When the erroneous data is reached, type A,

**A - Alter screen,**

and make the appropriate changes. To continue entering data, continue reviewing the screens until the message "ALL DATA HAS BEEN REVIEWED" is displayed.

**6. Command case:**

All data and commands can be entered in upper or lower case.

**7. Help:**

In the menu sections of the interface, help is available by typing ?. To exit the help screens, type exit.

**8. Error messages:**

If the input entered is not in range, or is syntactically incorrect, error messages will appear at the bottom of the screen. If needed, the value ranges are provided in the help screens.

## Chapter 2 - Entering Data

### The Model Input Process

CARE3MENU constructs a CARE III model input file by prompting the user for model parameters with a series of menus. System Fault Trees and Critical Pair Fault Trees (two input sections) are entered by scrolling screens (line-by-line) since the number of input lines is variable.

The model input process is divided into several sections. Each section is either a menu, a cyclical menu, a scrolling section, or a cyclical scrolling section. Cyclical menus and cyclical scrolling sections are designed to repeat so that more than one set of inputs can be entered. Sections of the model input process are:

1. Stage Description Input (cyclical menu)
2. Fault Handling Models (cyclical menu)
3. Fault Occurrence Models (cyclical menu)
4. Fault Handling Model Accuracy (a single menu)
5. Information Summary for use in checking System and Critical Pair Trees
6. System Fault Tree (scrolling)
  - Enter the System Tree Title\*
  - Enter the Output Gate ID Range
  - Enter the System Tree Logic Block\*

---

\* The input in these subsections can be continued to another line by typing a blank space followed by a c at the end of the line.

\*\* The entry of the Critical Pair Tree can be terminated at this point by typing **END**.

7. Critical Pair Fault Tree Input (cyclical scrolling)
  - Enter the Fault Tree Title\* \*\* (see page 7)
  - Enter Module and Logic Range ID
  - Enter Module Unit and Logic Range ID
  - Enter Critical Pair Tree Logic Block\*

8. Output Control Options Input (a single menu)

9. Runtime Control Options Input (a single menu)

Between each section, the following menu appears:

C - Continue to next screen  
T - Total Review of the model  
P - Partial Review of the model  
S - Store current model  
E - Exit CARE3MENU

and provides the user with the flexibility to correct past mistakes or review the data entered. Simply typing a C will advance to the next screen.

### CARE3MENU Screens

This section of the document describes each screen the user encounters when using the CARE3MENU interface. Note that all screen information is bolded, including default values. Each screen documentation section has two parts, the screen as it appears, and an accompanying explanation. The explanation section defines the function of the screen and lists the input parameters and their valid inputs or range of values. A CARE III input file can be written in list-directed or NAMELIST form. CARE3MENU generates the input file in NAMELIST paragraphs (see Ref. 6 for NAMELIST details). In parenthesis, the CARE III input file namelist paragraph (denoted by "\$") and variable are indicated.

For more information on the input data, see the CARE III Model Overview and User's Guide (second revision) and CARE III, Version 4 Enhancements.

(This page intentionally left blank.)

**Screen 1: Main Level Menu**

Contents:

**CARE3MENU — Menu-Based Data Input for CARE III**

- I - Input CARE III model
- A - Alter an existing model
- S - Store the current model
- E - Exit CARE3MENU

**Enter Desired Function:**

**PRECEDING PAGE BLANK NOT FILMED**

## Screen 1: Main Level Menu

### Explanation:

Commands are selected by typing the first letter of the appropriate choice.

- |       |  |
|-------|--|
| Input | The <b>Input</b> command prompts for data until the user chooses to store the model and/or exit the program.   |
| Alter | The <b>Alter</b> command queries the user for the VMS file containing the existing model. Note that the existing file may be complete or partial.  |
| Store | The <b>Store</b> command requests a file name from the user and stores the data. If no file has been entered, the program returns to the main level menu.  |
| Exit  | The <b>Exit</b> command exits the program. If a file has been entered and has not yet been stored, the user is asked to verify the exit. In this manner, data is not inadvertently lost. The user may choose to continue input from the last screen, or input a new model. |

## Screen 2: Stage Description Input

Contents:

### STAGE DESCRIPTION INPUT

Stage Name:

Number of Beginning Modules in Stage: 1

Minimum Number of Modules for Stage Operation: 1

Number of Beginning Submodules per Module: 0

Minimum Number of Submodules for Module Operation: 0

Spare Submodules On-Line (T/F): T

Set(s) of Modules Subject to Critical

Pair Failures:

Critical Fault Threshold: 0

Number of Fault Handling Models

Assigned to this Stage (opt.):

Verify Input (Y or N):

## Screen 2: Stage Description Input

### Explanation:

Stages are the fundamental units of input into CARE III. A stage is a set of identical modules.

The values printed on the screen are default values. Different values may be entered by typing in the parameter value. The menu will prompt for new stages until **END** is entered for the Stage Name.

The following parameters are entered:

Stage Name	A character string of up to 20 characters. (Optional \$STGNAMES paragraph -- STGNAME)
Number of Beginning Modules in Stage	Positive integer with maximum value of 70. (\$STAGES -- N)
Minimum Number of Modules for Stage Operation	A non-negative integer which specifies the minimum number of modules needed for stage functioning. This value must be less than or equal to the number of beginning modules in the stage. (\$STAGES -- M)
Number of Beginning Submodules per Module	A non-negative integer specifying the number of redundant units per module. (\$STAGES -- NSUB)
Minimum Number of Submodules for Module Operation	A non-negative integer which specifies the minimum number of submodules needed for module operation. This value must be less than or equal to the number of beginning submodules per module. (\$STAGES -- MSUB)



Spare Submodules On-Line (T/F)	A flag indicating whether the (NSUB-MSUB) spare modules are powered up (response = T) or powered down (response = F). (\$STAGES -- ACSP)
Set(s) of Modules Subject to Critical Pair Failures	These values correspond to the CARE III "NOP" variable. A maximum of five values is allowed. The default value is automatically calculated by CARE III. (\$STAGES -- NOP)
Critical Fault Threshold	A non-negative integer with a maximum value of (N-M). This parameter indicates how many units of a particular stage have to fail before the stage is susceptible to critical pair faults. (\$STAGES -- LC)
Number of Fault Handling Models Assigned to This Stage (opt.)	An optional non-negative integer with values (0 to 5). With this parameter the user may pre-specify the number of FHMs attached to the current stage.

(This page intentionally left blank.)

**Screens 3A, 5A, 12A, and 14A**

Contents:

- C - Continue to next screen
- T - Total Review of the model
- P - Partial Review of the model
- S - Store current model
- E - Exit CARE3MENU

**Enter Desired Function:**

**PRECEDING PAGE BLANK NOT FILMED**

## Screens 3, 5, 12, and 14

### Explanation:

The inputs are as follows:

Continue	To continue reviewing, type C (the default) to go to the next screen.
Total Review	The model, with the complete or partial information it contains up to this point, is reviewed from the first stage description screen.
Partial Review	The user can select the screen section(s) he or she wishes to review.
Store	The model, whether it be complete or partial, is stored.
Exit	The CARE3MENU program is exited. If the user attempts to exit the program without saving the model, the program asks the user to 1) verify the exit, 2) continue input from where the model was stopped, or 3) begin the input of a new model.

## **Screens 3B, 5B, 12B, and 14B - Partial Review Screen Selection**

### **Contents:**

#### **Review Selections**

- 1) **Stage Description Screens**
- 2) **Fault Handling Model Screens**
- 3) **Fault Occurrence Model Screens**
- 4) **Information Summary**
- 5) **System Fault Tree**
- 6) **Critical Pairs Fault Tree(s)**
- 7) **Output Control Options Screen**
- 8) **Runtime Control Options Screen**
- 9) **All**

**Make Selection(s):**

## Screens 3B, 5B, 12B, and 14B - Review Selections

### Explanation:

The user may select one or more screens to review. The following rules apply:

- 1) If a review section is chosen that has not yet been input, the message  
**\*\*\* Bad Response, Please Reenter \*\*\***  
will appear. WAIT for the review screen to reappear before changing the selection(s).
- 2) The inputs may be entered in any order.
- 3) If a review screen is chosen more than once, the redundant selections are ignored. In a review pass, a screen section will be reviewed only once.
- 4) To review the Fault Handling Model Accuracy data, review the Fault Occurrence Model screens.

This screen also appears when the model has been completely entered and the user chooses to review or alter the model (see Screen 16). NOTE: In a final review (after the model has been completely specified), the program completes the review and immediately requests the input file name without providing the opportunity to review the model in whole or in part again.

## Screen 4: Fault Handling Models (FHM)

### Contents:

#### Fault Handling Models

##### Fault Type:

Alpha = 0.0	(Exponential)
Beta = 0.0	(Exponential)
Delta = 3600	Delta FHM: EXPONENTIAL
Rho = 0.0	Rho FHM: EXPONENTIAL
Epsilon = 0.0	Epsilon FHM: EXPONENTIAL
Pa = 1.000000	
Pb = 0.0	
C = 1.000000	

Verify Input (Y or N):

## Screen 4: Fault Handling Models (FHM)

### Explanation:

Up to five fault types can be entered with this screen. The names chosen to describe the different fault handling models will be referenced in the next screens, **Fault Occurrence Models (FOM)**. Also, the names will be preserved in the CARE III input file in the namelist paragraph titled **FHMNAMES**.

The menu will prompt for fault types until either five models have been input or the user types **END** for the fault type name. At least one fault handling model must be entered, even if the user wants only a fault tree analysis of the system.

The following parameters are entered with this screen:

Alpha	The Active-to-Benign transition rate -- a real number in the range $\geq 0.0$ . (\$FLTTPY -- ALP)
Beta	The Benign-to-Active transition rate -- a real number in the range $\geq 0.0$ . (\$FLTTPY -- BET)
Delta	The rate at which a fault is detected -- a real number in the range $\geq 0.0$ . (\$FLTTPY -- DEL)
Delta FHM	The fault detection density function -- a character string set equal to UNIFORM or EXPONENTIAL. (\$FLTTPY -- IDELF)
Rho	The rate at which errors are generated -- a real number in the range $\geq 0.0$ . (\$FLTTPY -- RHO)
Rho FHM	The error generation density function -- a character string set equal to UNIFORM or EXPONENTIAL. (\$FLTTPY -- IRHOF)
Epsilon	The rate at which errors propagate -- a real number in the range $\geq 0.0$ . (\$FLTTPY -- EPS)



- Epsilon FHM The error propagation density function — a character string set equal to UNIFORM or EXPONENTIAL. (\$FLTTYP — IEPSF)
- Pa The probability that a fault detected in the active state is identified as permanent — a real number in the range 0.0 to 1.0. (\$FLTTYP — PA)
- Pb The probability that a fault detected in the benign state is identified as permanent — a real number in the range 0.0 to 1.0. (\$FLTTYP — PB)
- C The probability that an error will cause total system failure (single point failure probability) — a real number in the range 0.0 to 1.0. (\$FLTTYP — C)

(This page intentionally left blank.)

# Screen 6A: Fault Occurrence Models (FOM)

## Contents:

### Fault Occurrence Models

Stage: <Stage Name>

FHM Number: <#>

Fault Type:

FOM (Weibull/Exponential): EXPONENTIAL

Lambda = 0.100000E-03

Omega = 1.000000

### Fault Handling Model Names

.  
. .  
. .  
. .  
. .

Verify Input (Y or N):

PRECEDING PAGE BLANK NOT FILMED

## Screen 6A: Fault Occurrence Models (FOM)

### Explanation:

This screen appears for stages with no internal redundancy (number of submodules = 0).

The Stage Name that was assigned in the Stage Description Input section appears on the second line of this screen. The Fault Handling Model Names that were entered in the Fault Handling Models section also appear to remind the user of the names given to the FHMs.

At least one FHM must be assigned to each stage. Note that the screen indicates the FHM number in the upper right hand corner. A maximum of five fault handling models may be associated with a stage. A fault handling model may be assigned to the same stage more than once, and can also be assigned to more than one stage. The menu will keep prompting for FHMs until the user types **END** for the Fault Type, or until five FHMs have been assigned to the stage, or until the number of FHMs specified in the stage description screen has been entered for the stage.

The following parameters are entered with this screen:

Fault Type	A character string of up to 20 characters which must match one of the fault handling model names listed on the screen. (FLTCAT -- JTYP)
FOM	This parameter describes the density function dictating the rate at which faults arrive. (No Input File value)
Lambda	The rate at which faults arrive -- a real number > 0.0 (\$FLTCAT -- RLM)
Omega	The parameter $\omega$ of the Weibull fault occurrence rate -- a real number > 0.0. If the model is EXPONENTIAL, Omega should equal 1.0. (\$FLTCAT -- OMG)

# Screen 6B: Fault Occurrence Models (FOM)

## Contents:

### Fault Occurrence Models

Stage: <Stage Name>

FHM Number: <#>

**\*\* MODULES \*\***

\*

**\*\* SUBMODULES \*\***

\*

Fault Type:

\* Fault Type:

FOM (Weibull/Exponential): EXP

\* FOM (Weibull/Exponential): EXP

Lambda = 0.1000000E-03

\* Lambda = 0.1000000E-03

Omega = 1.000000

\* Omega = 1.0

Fault Handling Models: .

.

.

.

.

Verify Input (Y or N):

## Screen 6B: Fault Occurrence Model (FOM)

### Explanation:

This screen appears for stages with no internal redundancy (number of submodules > 0).

The Stage Name that was assigned in the Stage Description Input section appears on the second line of this screen. The number of FHMs assigned to the stage is indicated on the same line. The Fault Handling Model Names representing PERMANENT faults (alpha = 0.0) that were entered in the Fault Handling Models section also appear to remind the user of the names given to those FHMs. For stages with internal redundancy, only PERMANENT faults may be assigned.

At least one FHM must be assigned to each stage. A maximum of five fault handling models may be associated with a stage. A fault handling model may be assigned to the same stage more than once, and can also be assigned to more than one stage. The menu will keep prompting for FHMs until the user types **END** for the Fault Type; until five FHMs have been assigned to the stage; or until the number of FHMs specified in the stage description screen has been entered.

The following parameters are entered with this screen:

Fault Type	A character string of up to 20 characters which must match one of the fault handling model names listed on the screen. For stages with internal redundancy, only fault handling models of type PERMANENT (alpha = 0.0) are allowed. (\$FLTCAT -- JTYP and JSBTYP)
FOM	This parameter describes the density function dictating the rate at which faults arrive. (No Input File value)
Lambda	The rate at which faults arrive -- a real number > 0.0. (\$FLTCAT -- RLM and RLMSUB)
Omega	The parameter $\omega$ of the Weibull fault occurrence rate -- a real number > 0.0. If the model is EXPONENTIAL, Omega should equal 1.0. (\$FLTCAT -- OMG and OMGSUB)

## Screen 7: Fault Handling Model Accuracy

### Contents:

#### Fault Handling Model Accuracy

DBLDF (Doubling Step Difference) = 0.500000E-01

TRUNC (Coverage Function Truncation) = 0.100000E-03

Verify Input (Y or N):

## Screen 7: Fault Handling Model Accuracy

### Explanation:

This menu prompts for two parameters which control the accuracy of CARE III when processing fault handling functions where at least one uniform distribution was assigned; otherwise this screen will not appear.

The two parameters are:

DBLDF      A real value in the range 0.01 to 0.1 (\$FLTTYP -- DBLDF)

TRUNC      A real value in the range 1.0E-03 to 1.0E-06. (\$FLTTYP -- TRUNC)

For more information, see the **CARE III Model Overview and User's Guide**, second revision.



## Screen 8: Information Summary

### Contents:

#### Information Summary for Use in Checking System and Critical Pair Trees

Stage Name	Stage Number	No. of Modules In Stage	No. of Submodules per Module
---------------	-----------------	----------------------------	---------------------------------

[The appropriate information is displayed]

- C - Continue to next screen \*
- T - Total Review of the model
- P - Partial Review of the model
- S - Store current model
- E - Exit CARE3MENU

Enter Desired Function:

---

\* This is the only option allowed when viewing this screen during a model review; the other options will not be displayed.

## Screen 8: Information Summary

### Explanation:

This information is used for labeling the system tree and critical pair tree(s). The stage numbers are used to label the tree's inputs. At this point the user would transfer these numbers to the system tree or verify that the tree is numbered correctly.

Note that at this point, past data can be reviewed or model input can be terminated through a STORE or EXIT.

When reviewing a model, CONTINUE is the only option allowed and displayed.

## Screen 9: Ready to Begin Failure Configuration Input

### Contents:

#### Ready to Begin Failure Configuration Input

The input of failure trees is currently done with scrolling screens, as opposed to menus. This allows you to see the previously typed data for some time. However, with the scrolling input no online HELP will be available. By entering the help prompt below, the tree input HELP will be displayed now.

Type any key to continue...

## **Screen 9: Ready to Begin Failure Configuration Input**

### Explanation:

This screens warns the user that scrolling screens will be used and that corrections must be made before <cr> is typed. The user must wait until the Continue/Total/Partial/Store/Exit menu is reached to change data entered on these screens. A ? at this point will display the HELP screens, but once inside the scrolling screens, the user will not be allowed to access the HELP utility.

## **Screen 10: System Fault Tree Input**

### **Contents:**

#### **System Fault Tree Input**

**[Subsection Title]\***

- 
- \* (a) Enter System Tree Label**
  - (b) Enter Output Gate ID Range**
  - (c) Enter System Fault Tree Logic Block**

## Screen 10: System Fault Tree Input

### Explanation:

This is a scrolling screen with three subsections. Each of the subsections is described below:

#### Enter System Tree Title

This name is preserved in the CARE III input file and serves to indicate to the user the system being modeled. The title can be continued to more than one line by leaving one or more blank spaces and C or c to indicate continuation. The user can also select the CARE III default system tree (an OR gate with all stages as input) by typing **DEFAULT**. If the **DEFAULT** option is chosen, the program passes to the Critical Pairs Tree inputs.

#### Enter Output Gate ID Range

This screen displays the Input Event ID Range (from 1 to number-of-stages). The user must enter two integers representing the lowest and greatest output gate numbers. A comma or space(s) may separate the two numbers.

#### Enter System Fault Tree Logic Block

Enter the logic block. The program prompts with the gate numbers. The next character on the line represents the gate type, followed by the input numbers.

## Screen 11: Critical Pairs Fault Tree Input

Contents:

### Critical Pairs Fault Tree Input

[Subsection Title]\*

- 
- \* (a) Enter Fault Tree Label  
Enter "END" for no more Trees
  - (b) Enter Module and Logic Range ID
  - (c) Enter Module Unit to Stage Association  
Enter "END" to end Stage Input
  - (d) Enter Logic Gate ID

## Screen 11: Critical Pairs Fault Tree Input

### Explanation:

This is the second scrolling input screen. The top of the screen indicates that Critical Pairs Fault Trees are being input, with four subsection titles indicating which part of the tree is currently being entered:

#### Enter Fault Tree Label

Enter "END" for no more Trees

This line serves to indicate the critical pairs being modeled. If the system of interest has no critical coupling, typing "END" will terminate the critical pairs tree input. The title may be continued to two or more lines by leaving one or more blank spaces before typing C or c for continuation.

#### Enter Module and Logic Range ID

This line indicates the range of inputs and the range of outputs in the critical pair tree with four integers separated by commas or spaces. The inputs to the critical pair are modules (not stages); therefore, the range indicates the number of modules being modeled with the tree. The range of outputs, as with the System Tree, is simply the range of logic gates in the critical pair tree.

#### Restrictions:

- (a) The range of inputs must start at 1.
- (b) No more than 70 modules may be modeled in a critical pair tree.
- (c) The output gates must be numbered consecutively. As a result, the last integer minus the third integer must equal one less than the number of gates in the tree.

#### Enter Module Unit to Stage Association

Each line consists of three integers, where the first integer represents a stage number, and the last two integers represent the range of modules comprising that stage. Enter **END** to terminate the input on this screen.



**Restrictions:**

- (a) No more than twenty stages may be represented in one critical pairs tree.
- (b) The stages must be listed in increasing order. They do not, however, have to be consecutive.
- (c) The range of modules for each stage must equal the number of modules comprising that stage. Though all modules in the stage must be accounted for in this section, the modules do not have to be used in the tree.

**Enter Logic Gate ID**

The program prompts for the logic block line by line with the gate numbers. The first character input by the user indicates the type of gate and the integers that follow indicate the inputs to the gate.

(This page intentionally left blank.)

**Screen 13: Output Control Options Input**

Contents:

**Output Control Options Input**

**Output Option (1-4): 1**

**Coverage Functions Plot (T or F): F**

**Coverage Y-Axis Selection (Y-Linear, Y-LOG, Both, Log-log): Y-LOG**

**Reliability Functions Plot (T or F): F**

**Reliability Y-Axis Selection (Y-Linear, Y-LOG, Both, Log-log): Y-LOG**

**Enter "?" for Menu Help**

**Verify Input (Y or N):**

**PRECEDING PAGE BLANK NOT FILMED**

## Screen 13: Output Control Options Input

### Explanation:

This screen provides several options for output plots. The following parameters are entered:

Output Option	Selecting <u>1</u> will provide summary results only and is the default value. Selecting <u>2</u> will provide the $P(t 1)$ (probability of successful operation) values plus summary results. Selecting <u>3</u> will print $Q(t 1)$ (probability of a fault-handling error) plus summary results. To get $P(t 1)$ , $Q(t 1)$ , and summary results, select <u>4</u> . (\$STAGES -- IRLPCD)
Coverage Functions Plot (T or F)	A flag for the plot of single and double fault-handling functions. (\$FLTTYP -- CVPLOT)
Coverage Y-Axis Selection (Y-Linear, Y-Log, Both, Log-log)	Y-axis scale for plotting fault-handling function values.
Reliability Functions Plot (T or F)	A flag indicating whether the summary information -- QSUM, P*SUM, and QSUM + P*SUM -- is to be plotted against time.
Reliability Y-Axis Selection (Y-Linear, Y-Log, Both, Log-log)	Y-axis scale for plotting summary information.

## Screen 15: Runtime Control Options Input

### Contents:

#### Runtime Control Options Input

Mission Time: 1.000000

Integration Steps: LOGARITHMIC

Timebase (Hours, Minutes, Seconds, Days, Years): HOURS

Cut Truncation Value = 0.1000000E-09

QPTRNC Value = 0.1000000E-01

Number of Stages per Subrun: 20

Verify Correctness of Data (T/F): T

Enter "?" For Menu Help

Verify Input (Y or N):

## Screen 15: Runtime Control Options Input

### Explanation:

Mission Time	A real number > 0.0. (\$RNTIME -- FT)
Integration Steps:	In most cases, users will select LOG, which indicates that varying (referred to as "logarithmic") time steps are to be used. If the user chooses even time steps, an integer from 17 to 64 is selected. This indicates the number of integration steps used by the CARE III program. (If LOG is chosen, \$FLTTYP -- LGTMST = T, and the value for parameter \$RNTIME -- NSTEPS will automatically be set by CARE III. If a linear time step is chosen, \$FLTTYP -- LGTMST = F, and \$RNTIME -- NSTEPS = value specified.)
Timebase (Hours, Minutes, Seconds, Days, Years)	The time scale for the mission time. (\$RNTIME -- ITBASE)
Cut Truncation Value	A parameter used to limit the number of fault vectors used in computing the QSUM values. (\$RNTIME -- PSTRNC)
QPTRNC Value	A parameter used to limit the number of fault vectors used in computing the QSUM values. (\$RNTIME -- QPTRNC)
Number of Stages per Subrun	The number of noncritically-paired stages that are to be grouped together in a subrun. (\$RNTIME -- NPSBRN)
Verify Correctness of Data (T/F)	CARE III verifies that all data is correct and in range. (\$RNTIME -- CKDATA)

**Screen 16: Model Input Complete**

Contents:

**\*\*\*MODEL INPUT COMPLETE\*\*\***

**Do you wish to review or alter this model (Y/N)?**

## **Screen 16: Model Input Complete**

### **Explanation:**

At this point, the model may be totally or partially reviewed and alterations made where necessary. If the user elects to review the model, the Review Selection Menu appears.



## Screen 17: File Name Input

### Contents:

File Name Input

Enter File Name for Model Storage:

Verify Input (Y or N):

## Screen 17: File Name Input

### Explanation:

The program automatically requests the name of the input file. Typically, a file name with a three character extent is chosen; for example, EXAMPLE.DAT. The program then returns to Screen 1, CARE3MENU — Menu-Based Data Input for CARE III.

**Screen 18: Exit Without Store**

Contents:

**WARNING: All current data will be lost**

**Verify QUIT (Y or N):**

## **Screen 18: Exit Without Store**

### **Explanation:**

If the user attempts to exit without first storing the model, CARE3MENU warns the user that all data will be lost and verifies the user's intent. If the user chooses N, the options of continuing input from the last input section or inputting a new model are provided.

**Screen 19: File Recovery**

Contents:

**File Recovery**

**Enter Name of File to be Altered:**

## Screen 19: File Recovery

### Explanation:

The program prompts for the name of the input file to be altered. If no file is found with that name, the message appears:

```
*** Error opening file <filename> ***.
```

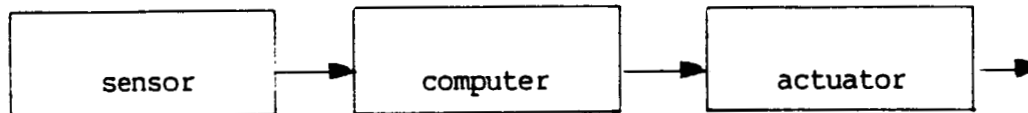
The user may exit the screen by typing "END", or may continue entering input file names.

## Chapter 3 - Example CARE3MENU Sessions

### Example One\*

This example illustrates a simple model with no critical-pairs or single-point failures. The system is non-redundant and has three components: a sensor, a computer, and an actuator. The probabilities of failure for times from 0 to 10 hours are to be predicted for constant failure rates. This example ignores the fault-handling capabilities of the CARE III program and therefore reduces to a simple fault tree analysis.

Functional Block Diagram of the System



Non-redundant Flight Control System

---

\* S.J. Bavuso, J.E. Brunelle, and P.L. Petersen, **CARE III Hands-On Demonstration and Tutorial (Second Revision)**, NASA Technical Memorandum 86404, April 1985.

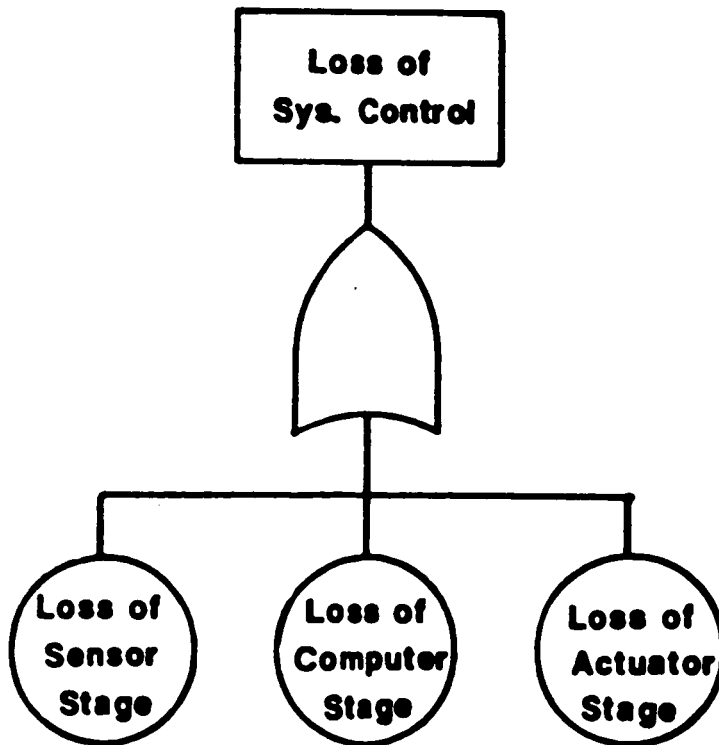
**System Failure Criteria:**

The system fails if any stage fails. A stage fails when a failure occurs within the stage.

Input Data	
Parametric Data for Failure Occurrence Model (Exponential)	
sensor stage	$\lambda_s = 1.5 \times 10^{-5}$
computer stage	$\lambda_c = 4.8 \times 10^{-4}$
actuator stage	$\lambda_a = 3.7 \times 10^{-5}$

**System Tree**

The system tree describes system failure as a result of stage failure interactions.





## Using the User-Friendly Interface -- Example One

This section shows the step-by-step entry of the model in Example One into the CARE3MENU program. Separate screens are separated, and user entries are in bold face type. Special instructions to the user are shown on the right.

CARE3MENU -- Menu Based Data Input for CARE III

Enter Desired Function: **I**

---

### Stage Description Input

Stage Name: **SENSOR**

Number of Beginning Modules in Stage: **1** <cr>

.

.

(Select default values for all of the remaining items.)

Verify Input (Y or N):

(Type **Y** to accept input, **N** to make corrections.)

---

### Stage Description Input

Stage Name: **COMPUTER**

Number of Beginning Modules in Stage: **1** <cr>

.

.

(Select default values for all of the remaining items.)

Verify Input (Y or N): **Y**

---

### Stage Description Input

Stage Name: **ACTUATOR**

Number of Beginning Modules in Stage: **1** <cr>

.

.

(Select default values for all of the remaining items.)

Verify Input (Y or N): **Y**

---

### Stage Description Input

Stage Name: **END**

(All stages have been input.)

---

Enter Desired Function: C

(Continue to the next screen.)

---

Fault Handling Models

Fault Type: NULL

Alpha = 0.0 <cr>

·  
·  
·

(Select default values.)

Verify Input (Y or N): Y

---

Fault Handling Models

Fault Type: END

(All fault handling models have been input.)

---

Enter Desired Function: C

(Continue to the next screen.)

---

Fault Occurrence Models

Stage: SENSOR

Fault Type: NULL

FOM (Weibull/Exponential): EXPONENTIAL <cr>

Lambda = 1.5E-5

Omega = 1.0 <cr>

(A fault type name must be assigned to each stage, even though this is simply a fault tree analysis.)

Verify Input (Y or N): Y

---

Fault Occurrence Models

Stage: SENSOR

Fault Type: END

(All fault occurrence models for this stage have been input.)

---

Fault Occurrence Models

Stage: COMPUTER

Fault Type: NULL  
FOM (Weibull/Exponential): EXPONENTIAL <cr>  
Lambda = 4.0E-4  
Omega = 1.0 <cr>

Verify Input (Y or N): Y

---

Fault Occurrence Models

Stage: COMPUTER

Fault Type: END

---

Fault Occurrence Models

Stage: ACTUATOR

Fault Type: NULL  
FOM (Weibull/Exponential): EXPONENTIAL <cr>  
Lambda = 3.7E-5  
Omega = 1.0 <cr>

Verify Input (Y or N): Y

---

Fault Occurrence Models

Stage: ACTUATOR

Fault Type: END

---

Information Summary for Use in Checking  
System and Critical Pair Trees

.  
.  
.

Enter Desired Function: C

---

Ready to Begin Failure Configuration Input

.  
.  
.

Type any key to continue... <cr>

---

System Fault Tree Input

Enter System Fault Tree Label

SYSTEM TREE, CUF1 USER'S GUIDE EX 1

---

System Fault Tree Input

Enter Output Gate ID Range

INPUT EVENT ID RANGE: 1 3  
OUTPUT GATE ID RANGE: 4 4

(The output gate is numbered 4. Because the input requires a range, the range becomes 4 to 4.)

---

System Fault Tree Input

Enter System Fault Tree Logic Block

4 0 1 2 3

(The 0 indicates an OR gate, and the numbers 1, 2, and 3 indicate that the three inputs are stages 1 to 3.)

---

Critical Pairs Fault Tree Input

Enter Fault Tree Label

END

(This example has no critical pair tree.)

---

Enter Desired Function: C

(Continue to the next screen.)

---

Output Control Options Input

Output Option (1-4): 1 <cr>

.  
.  
.

(Select default values.)

Verify Input (Y or N): Y

---

Enter Desired Function: C

(Continue to the next screen.)

---

Runtime Control Options Input

Mission Time: 10

Integration Steps: LOGARITHMIC <cr>

.  
.  
.

(Select default values.)

Verify Input (Y or N): Y

---

\*\*\*MODEL INPUT COMPLETE\*\*\*

Do you wish to review or alter this model (Y/N)?

(Type N to continue, Y  
to do otherwise.)

---

File Name Input

Enter File Name for Model Storage: EX1.DAT

---

CARE3MENU -- Menu Based Data Input for CARE III

Enter Desired Function: E

(Exit the program.)

=====

# EX1.DAT -- CARE III Input File

\$FHMNAMES

FHMNAME(1)= 'NULL'

\$END

\$FLTTYF

NFTYPS=1,

ALP= 0.0 ,

BET= 0.0 ,

DEL= 3600.0 ,

RHO= 0.0 ,

EPS= 0.0 ,

IDELF= 1 ,

IRHOF= 1 ,

IEPSF= 1 ,

MARKOV= 1 ,

PA= 1.0 ,

PB= 0.0 ,

C= 1.0 ,

LGTMST=T

\$END

\$STGNAMES

STGNAME(1)= 'SENSOR',

STGNAME(2)= 'COMPUTER',

STGNAME(3)= 'ACTUATOR'

\$END

\$STAGES

NSTGES=3,

N = 1, 1, 1,

M = 1, 1, 1,

NSUB= 0, 0, 0,

MSUB= 0, 0, 0,

LC= 0, 0, 0,

IRLPCD=1,

RLPLOT=F, IAXSRL=2

\$END

\$FLTCAT

NFCATS=1,1,1,

JTYP(1,1)= 1,

JTYP(1,2)= 1,

JTYP(1,3)= 1,

OMG(1,1)= 1.0 ,

OMG(1,2)= 1.0 ,

OMG(1,3)= 1.0 ,

RLM(1,1)= 1.500000E-05,

RLM(1,2)= 4.000000E-04,

RLM(1,3)= 3.700000E-05

\$END

\$RNTIME

FT= 10.0000 ,ITBASE=1,  
PSTRNC= 0.100000E-09,  
QPTRNC= 0.100000E-01,  
NPSBRN=20,  
CKDATA=T,  
SYSFLG=T,CPLFLG=F

\$END

SYSTEM TREE, CUFI USER'S GUIDE EX 1

1 3 4 4

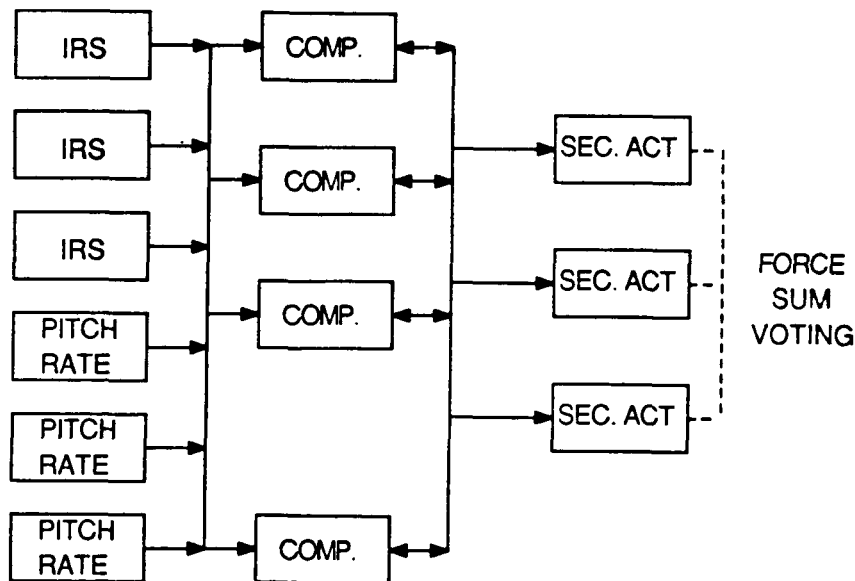
4 0 1 2 3

## Example Two\*

Example 2 (Example 6 in the reference) introduces critical coupling within stages and the use of hot spares. In this example, the critical pair failures will be defined for the computer stage only. The computers will also be subject to single point failures.

To illustrate the critical coupling between computers, the computer stage is configured as a triplex with a hot spare. The triplex computers randomly execute a self-test program to detect faults, and the hot spare is constantly being flexed by randomly replacing a triad module, which in turn becomes the hot spare. A critically coupled failure can only occur between two of the three computers in the triad. This condition will result in immediate system failure. The conservative assumption is made here that the presence of two latent (undetected) faults in the triad will preclude a correct majority vote.

Functional Block Diagram of System



\* S.J. Bavuso, J.E. Brunelle, and P.L. Petersen, **CARE III Hands-On Demonstration and Tutorial (Second Revision)**, NASA Technical Memorandum 86404, April 1985.



### System Failure Criteria:

The system fails if any stage fails.

A single point failure in the computer stage causes system failure.

The system fails if a critically coupled failure occurs in the computer stage.

The inertial reference sensor stage fails if 2 out of 3 modules fail.

The pitch rate stage fails if 2 out of 3 modules fail.

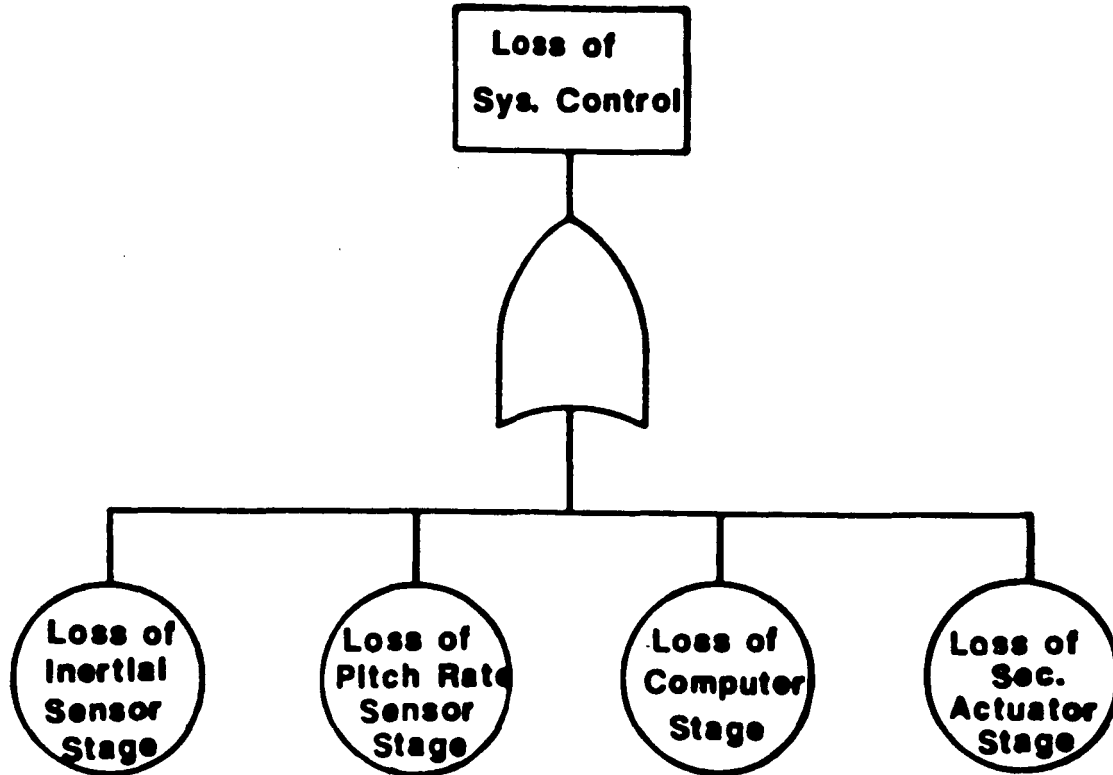
The computer stage fails if 3 out of 4 modules fail.

The secondary actuator stage fails if 2 out of 3 modules fail.

Input Data for Fault Occurrence Model	
Parametric Data	Exponential
inertial sensor	$\lambda_i = 1.5 \times 10^{-5}$
pitch rate sensor	$\lambda_p = 1.9 \times 10^{-5}$
computer	$\lambda_c = 4.8 \times 10^{-4}$
secondary actuator	$\lambda_s = 3.7 \times 10^{-5}$

## System Tree

The system tree describes system failure as a result of stage failure interactions.



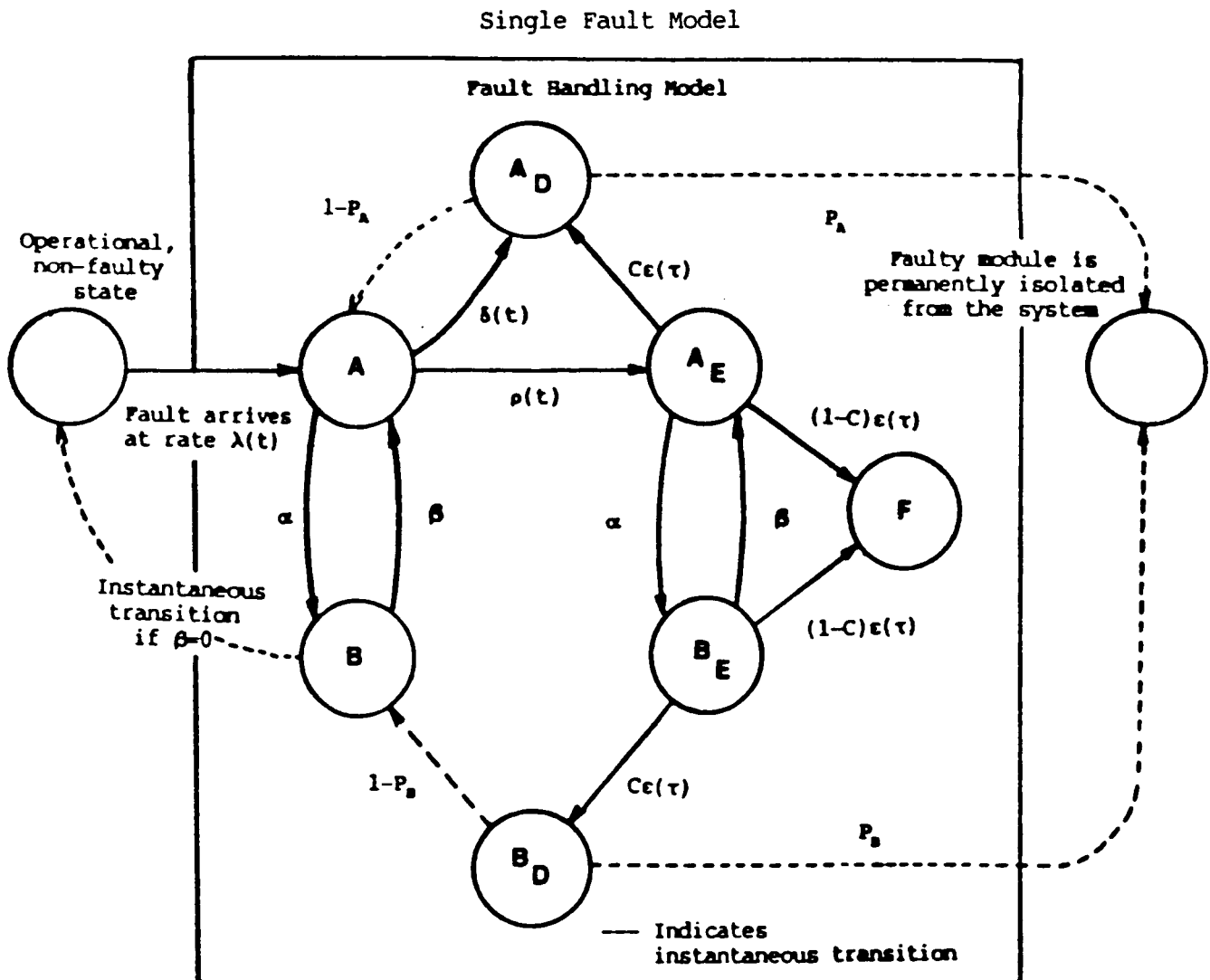
### Input Data for Single Fault Model Parametric Data

Self test rate,  $\delta(t) = 360$  detections/hour (exponential detection times)

Error detection rate,  $\epsilon(t) = 3600$  detections/hour (exponential detection times)

Error recovery probability,  $C = 0.999$

Error generation rate,  $\rho(t) = 180$  errors/hour (exponential generation times)

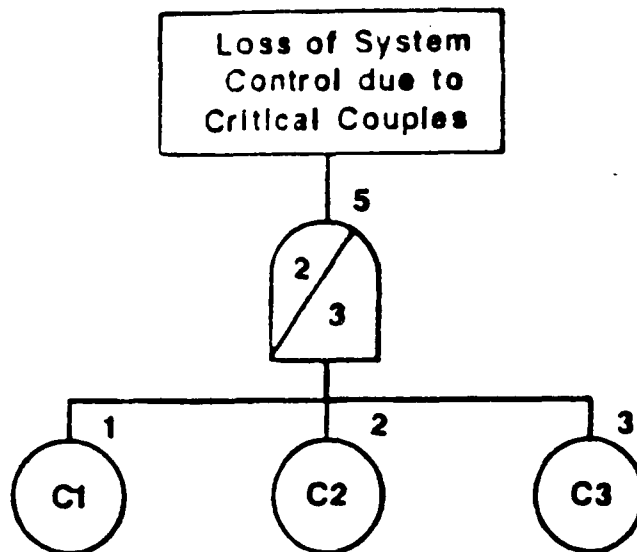


## Critical Pair Tree

The parameters for the double fault model are defined by the single fault model; however, the user must define the conditions that constitute a critically coupled failure. This definition is done by specifying a critical pair tree that identifies the modules that are critically coupled.

Computers 1-3 are initially operational and are subject to the critical pair failure criterion. Computer 4 is treated as a hot spare which, if it fails as a spare, will simply vanish. For that reason, its failure is not part of the critical pair tree. During the mission duration, the hot spare will replace one of the operationally trio on a random basis (called flexing). When it becomes a member of the voting trio, a failure in it plus another (near-coincident) failure in the voting trio will cause system failure.

If a single member of the voting trio fails and fault-handling is successful, the spare, if it is still good, will replace the failed voting member. Another failure, leaving two good in-use computers, will still be subject to the critical pair failure criterion. The next failure, leaving one in-use computer, will cause the computer stage to fail since the 2 out of 4 success criterion will not be met for the stage survival. In this example, loss of the computer stage will cause system failure because there is no system tree redundancy.



## Using the User-Friendly Interface -- Example Two

Enter the program as in Example One.

---

### Stage Description Input

Stage Name: **INERTIAL REF**  
Number of Beginning Modules in Stage: 3  
Minimum Number of Modules for Stage Operation: 2  
:  
:  
Number of Fault Handling Models  
Assigned to this Stage (opt.): 1

(Select default values.)

Verify Input (Y or N): (Type Y to accept input, N to  
make corrections.)

---

### Stage Description Input

Stage Name: **PITCH RATE**  
Number of Beginning Modules in Stage: 3  
Minimum Number of Modules for Stage Operation: 2  
:  
:  
Number of Fault Handling Models  
Assigned to this Stage (opt.): 1

(Select default values.)

Verify Input (Y or N): Y

---

### Stage Description Input

Stage Name: **COMPUTER**  
Number of Beginning Modules in Stage: 4  
Minimum Number of Modules for Stage Operation: 2  
:  
:  
Set(s) of Modules Subject to Critical  
Pair Failures: 3  
Critical Fault Threshold: 0 <cr>  
Number of Fault Handling Models  
Assigned to this Stage (opt.): 1

(Select default values.)

Verify Input (Y or N): Y

---

Stage Description Input

Stage Name: **SECONDARY ACTUATOR**  
Number of Beginning Modules in Stage: 3  
Minimum Number of Modules for Stage Operation: 2

.  
.  
(Select default values.)

Number of Fault Handling Models  
Assigned to this Stage (opt.): 1

Verify Input (Y or N): Y

---

Stage Description Input

Stage Name: **END** (All stages have been input.)

---

Enter Desired Function: C (Continue to the next screen.)

---

Fault Handling Models

Fault Type: **NULL**

Alpha = 0.0 <cr>

.  
.  
(Select default values.)

Verify Input (Y or N): Y

---

Fault Handling Models

Fault Type: **PERMANENT**

Alpha = 0.0 <cr>

Beta = 0.0 <cr>

Delta = 360

Delta FHM: **EXPONENTIAL** <cr>

Rho = 180

Rho FHM: **EXPONENTIAL** <cr>

Epsilon = 3600

Epsilon FHM: **EXPONENTIAL** <cr>

Pa = 1.000000 <cr>  
Pb = 0.0 <cr>  
C = 0.999

Verify Input (Y or N): Y

---

Fault Handling Models

Fault Type: END

(All fault handling models have  
been input.)

---

Enter Desired Function: C

(Continue to the next screen.)

---

Fault Occurrence Models

Stage: INERTIAL REF

Fault Type: NULL

FOM (Weibull/Exponential): EXPONENTIAL <cr>

Lambda = 1.5E-5

Omega = 1.0 <cr>

(A fault type name must be  
assigned to each stage.)

Verify Input (Y or N): Y

\*\*\* Specified No. of FHMs Reached \*\*\*

---

Fault Occurrence Models

Stage: PITCH RATE

Fault Type: NULL

FOM (Weibull/Exponential): EXPONENTIAL <cr>

Lambda = 1.9E-5

Omega = 1.0 <cr>

Verify Input (Y or N): Y

\*\*\* Specified No. of FHMs Reached \*\*\*

---

Fault Occurrence Models

Stage: COMPUTER

Fault Type: PERMANENT

FOM (Weibull/Exponential): EXPONENTIAL <cr>  
Lambda = 4.8E-4  
Omega = 1.0 <cr>

Verify Input (Y or N): Y

\*\*\* Specified No. of FHMs Reached \*\*\*

---

Fault Occurrence Models

Stage: SECONDARY ACTUATOR

Fault Type: NULL  
FOM (Weibull/Exponential): EXPONENTIAL <cr>  
Lambda = 3.7E-5  
Omega = 1.0 <cr>

Verify Input (Y or N): Y

\*\*\* Specified No. of FHMs Reached \*\*\*

---

Information Summary for Use in Checking  
System and Critical Pair Trees

.  
.  
.

Enter Desired Function: C

---

Ready to Begin Failure Configuration Input

.  
.  
.

Type any key to continue... <cr>

---

System Fault Tree Input

Enter System Fault Tree Label

SYSTEM TREE, CUF1 USER'S GUIDE EX 2

---



System Fault Tree Input

Enter Output Gate ID Range

INPUT EVENT ID RANGE: 1 4  
OUTPUT GATE ID RANGE: 5 5

(The output gate is numbered 5. Because the input requires a range, the range becomes 5 to 5.)

---

System Fault Tree Input

Enter System Fault Tree Logic Block

5 0 1 2 3 4

(The 0 indicates an OR gate, and the numbers 1, 2, 3, and 4 indicate that the four inputs are stages 1 to 4.)

---

Critical Pairs Fault Tree Input

Enter Fault Tree Label

CRITICAL PAIR TREE, COMPUTER STAGE C  
CUFI USER'S GUIDE EX 2

(Note the continuation mark.)

---

Critical Pairs Fault Tree Input

Enter Module and Logic Range ID

1 4 5 5

(The inputs to the tree are the modules numbered 1 to 4, as indicated. The range of logic gates indicates that one gate, numbered 5, defines this tree.)

---

Critical Pairs Fault Tree Input

Enter Module Unit to Stage Association

3 1 4  
END

(The four modules from stage 3, the computer stage, are numbered 1 to 4 in the critical pair tree.)

---

Critical Pairs Fault Tree Input

Enter Logic Gate ID

5 2 1 2 3 4

(The logic gate, numbered 5, has four inputs, numbered 1 to 4. The 2-of-4 gate indicates that a fault in 2 of the four modules will constitute a critical pair.)

---

Critical Pairs Fault Tree Input

Enter Fault Tree Label

END

(More than one critical pair tree may be entered. This example, however, has only one tree.)

---

Enter Desired Function: C

(Continue to the next screen.)

---

Output Control Options Input

Output Option (1-4): 1 <cr>

·  
·  
·

(Select default values.)

Verify Input (Y or N): Y

---

Enter Desired Function: C

(Continue to the next screen.)

---

Runtime Control Options Input

Mission Time: 10

Integration Steps: LOGARITHMIC <cr>

·  
·  
·

(Select default values.)

Verify Input (Y or N): Y

---

\*\*\*MODEL INPUT COMPLETE\*\*\*

Do you wish to review or alter this model (Y/N)?  
(Type N to continue, Y  
to do otherwise.)

---

File Name Input

Enter File Name for Model Storage: **EX2.DAT**

---

CARE3MENU -- Menu Based Data Input for CARE III

Enter Desired Function: **E**

(Exit the program.)

=====

## EX2.DAT -- CARE III Input File

```

$FHMNAMES
  FHMNAME(1)= 'NULL',
  FHMNAME(2)= 'PERMANENT'
$END
$FLITYP
  NFTYPS=2,
  ALP=      0.0      ,      0.0      ,
  BET=      0.0      ,      0.0      ,
  DEL= 3600.0      ,      360.0      ,
  RHO=      0.0      ,      180.0      ,
  EPS=      0.0      ,      3600.0      ,
  IDELF=     1      ,      1      ,
  IRHOF=     1      ,      1      ,
  IEPF=     1      ,      1      ,
  MARKOV=    1      ,
  PA=       1.0      ,      1.0      ,
  PB=       0.0      ,      0.0      ,
  C=        1.0      , 9.990000E-01,
  LGTMST=T
$END
$STGNAMES
  STGNAME(1)= 'INERTIAL REF',
  STGNAME(2)= 'PITCH RATE',
  STGNAME(3)= 'COMPUTER',
  STGNAME(4)= 'SECONDARY ACTUATOR'
$END
$STAGES
  NSTGES=4,
  N = 3, 3, 4, 3,
  M = 2, 2, 2, 2,
  NSUB= 0, 0, 0, 0,
  MSUB= 0, 0, 0, 0,
  LC= 0, 0, 0, 0,
  IRLPCD=1,
  RLPLOT=F, IAXSRL=2
$END
$FLTCAT
  NFCATS=1,1,1,1,
  JTYP(1,1)= 1,
  JTYP(1,2)= 1,
  JTYP(1,3)= 2,
  JTYP(1,4)= 1,
  OMG(1,1)=      1.0      ,
  OMG(1,2)=      1.0      ,
  OMG(1,3)=      1.0      ,
  OMG(1,4)=      1.0      ,
  RLM(1,1)= 1.500000E-05,
  RLM(1,2)= 1.900000E-05,
  RLM(1,3)= 4.800000E-04,
  RLM(1,4)= 3.700000E-05
$END

```

\$RNTIME

FT= 10.0000 ,ITBASE=1,  
PSTRNC= 0.100000E-09,  
QPTRNC= 0.100000E-01,  
NPSBRN=20,  
CKDATA=T,  
SYSFLG=T,CPLFLG=T

\$END

SYSTEM TREE, CUFI USER'S GUIDE EX 2

1 4 5 5

5 0 1 2 3 4

CRITICAL PAIR TREE, COMPUTER STAGE C

CUFI USER'S GUIDE EX 2

1 4 5 5

3 1 4

5 2 1 2 3

## Chapter 4 -- CARE3MENU Error Messages

These error messages pertain **ONLY** to the system tree and/or critical pairs tree.

The following error messages, listed in alphabetical order, may be encountered when reviewing an input file, or when entering the system or critical pair trees. Each error message lists the meaning of the error and the action that must be taken to correct it.

**\* Bad Stage Number. Must appear in form: 1 <= Stage Number <=NSTGES**

Meaning: Stage Number is not in valid range.

Action: Reenter all stage numbers and their input gate ranges.

**\* For stage <N>, (3rd #)-(2nd #)+1 must equal <N(X)>**

Meaning: X is a valid stage number, but the number of modules indicated on the line does not equal the number of modules comprising the stage. The last two values on the line must satisfy the equation:  $N(X) = (2nd \#) - (3rd \#) + 1$ , where  $N(X)$  is the number of modules in stage X.

Action: Reenter all stage numbers and their input gate range.

**\* Gate Input May Appear Only ONCE in the Gate Logic**

Meaning: Self explanatory.

Action: Reenter the critical pairs fault tree logic block.

**\* Gate Output Numbers must be SEQUENTIAL**

Meaning: Output numbers must be sequentially increasing integers.

Action: Reenter the critical pairs fault tree logic block.

**\* Gate Type  $\geq$  2 must have at least as many Gate Inputs**

Meaning: If the gate type is greater than or equal to 2, at least as many inputs as the output number must be entered.

Action: Reenter the system fault tree logic block.

**\* Gate Type "A" MUST have TWO Gate Inputs**

Meaning: If the gate type equals "A" for a critical pair tree gate, exactly two inputs must be entered.

Action: Reenter the critical pairs fault tree logic block.

**\* Gate Type 2 MUST Have at Least TWO Gate Inputs**

Meaning: If the gate type equals 2, at least two inputs must be entered.

Action: Reenter the critical pairs fault tree logic block.

**\* Input Line NOT Complete: Gate\_Output\_No. Gate\_Type Gate\_Input(s) [c]**

**\* Input MUST Appear in form: Gate\_Output\_No. Gate\_Type Gate\_Input(s) [c]**

Meaning: Three fields must be present on the gate line -- a gate output number, a gate type, and at least one gate input. The gate inputs have the possibility of being continued onto multiple lines. The continuation symbol "c" must be present at the end of the line to continue the gate inputs.

Action: Reenter the system fault tree logic block.

**\* Invalid Gate Type: Must be "A", "O", "0", "1", or "2"**

Meaning: A gate type other than "A" (AND), "O" or "0" (OR), "1", or "2" was entered for the critical pair tree. The gate type must be one of the gates listed above.

Action: Reenter the critical pairs fault tree logic block.

**\* Invalid Gate Type: Must be "A", "O", "V", or any positive integer  $\geq$  zero.**

Meaning: A gate type other than an "A" (AND), "O" (OR), "V" (INVERT), or a positive integer  $\geq 0$  was entered. The gate type must be one of the gates listed above.

**\* Invert Gates may have only ONE Gate Input**

Meaning: Gate type "V" may have only one input.

Action: Reenter system fault tree logic block.

**\* Maximum Tree Lines Reached - Alter MAX\_SYST\_LINES**

Meaning: All available memory has been used; no more inputs will be accepted.

Action: Ask the programmer to increase magnitude of parameter MAX\_SYST\_LINES in file C3MDEF.FOR.

**\* No Gate Input may be Greater Than its Gate Output Number**

Meaning: Self explanatory.

Action: Reenter the critical pairs fault tree logic block.

**\* Output Number of First Logic Gate Must Equal I: < > of Gate Outputs Range**

Meaning: Output number of first logic gate must equal the minimum gate output of critical pairs fault tree input and output gate ID range.

Action: Reenter the critical pairs fault tree logic block.



**\* Output Number of Last Logic Gate Must Equal J: < > of Gate Outputs Range**

**Meaning:** Output number of last logic gate must equal maximum gate output of critical pairs fault tree input and output gate id range.

**Action:** Reenter the critical pairs fault tree logic block.

**\* Output Number of First Logic Gate Must Equal I of Gate Outputs Range**

**Meaning:** Output number of first logic gate must equal minimum gate output of system fault tree input and output gate ID range.

**Action:** Reenter the system fault tree logic block.

**\* Output Number of Last Logic Gate Must Equal J of Gate Outputs Range**

**Meaning:** Output number of last logic gate must equal maximum gate output of system fault tree input and output gate ID range.

**Action:** Reenter the system fault tree logic block.

**\* Ranges Must Appear in Form:  $1 \leq K \leq (L \leq 1000) < I \leq J \leq 2000$**

**Meaning:** Four numbers must appear in the critical pairs fault tree input line. The first number, K, is the minimum gate input, and it must be greater than or equal to 1. The second number, L, is the maximum gate input and it must be greater than or equal to K but less than or equal to 1000. The third number, I, is the minimum gate output, and it must be greater than L. The fourth number, J, is the maximum gate output, and it must be greater than or equal to I but less than or equal to 2000.

**Action:** Reenter input and output gate id range for critical pairs fault tree.

**\* Stage Reference Numbers MUST be in Increasing Order**

**Meaning:** Stage Numbers must be entered as increasing positive integers.

**Action:** Reenter all stage numbers and their input gate ranges.

**\* Sys. Tree Ranges must appear as:  $1 = K \leq (L = NSTGES) < I \leq J \leq 2000$**

**Meaning:** Four numbers must appear on the system fault tree input line. The first number, K, is the minimum gate input and it must be equal to 1. The second number, L, is the maximum gate input and it must equal the number of stages but be not less than K. The third number, I, is the minimum gate output, and it must be greater than L. The fourth number, J, is the maximum gate output, and it must be greater than or equal to I but less than or equal to 2000.

**Action:** Reenter input and output gate id range for critical pairs fault tree.

**\* The 2nd # in the 1st line must equal < > (the lower bound of the gate input range)**

**Meaning:** For the first stage referenced in the Module to Stage Association, the lower bound of the gate input range must equal the minimum gate input value entered in the Module and Logic Range ID.

**Action:** All stage numbers and input ranges must be reentered.

**\* The 3rd # in the last line must equal < > (upper bound of gate input range)**

**Meaning:** The upper bound of the gate input range for last stage must equal the maximum gate input value which was entered with the critical pairs fault tree's input and output gate ID range.

**Action:** All stage numbers and input ranges must be reentered.

**\* Too Many Input Gates, Use Subrun**

Meaning: Specifically,  $(MAX\_GATE\_INPUT - MIN\_GATE\_INPUT + 1)$  is greater than 70.

Action: Reenter critical pairs fault tree input and output gate ID range.

**\* Too Many Inputs for the Gate**

Meaning: Specifically, the first index value (100) for the variable `gonum(100,MAX_SYSTLINES)` is too small.

Action: Change the value of "`gonum(100,MAX_SYST_LINES)`" in `CARE3MENU.FOR`, `GETLBLK.FOR`, `VERCFTLBLK.FOR`, and `VERSYSLBLK.FOR`.

## Chapter 5 -- CARE III Parameters and Value Ranges

Alpha	Real number in the range $\geq 0.0$
Beta	Real number in the range $\geq 0.0$
C	Real number in the range 0.0 to 1.0
Coverage Functions Plot	Logical variable T or F
Coverage Y-Axis Selection	Character string set to Y-Linear, Y-Log, Both, or Log-log
Critical Fault Threshold	Non-negative integer with a maximum value of (N-M)
Cut Truncation Value	Parameter used to limit the number of fault vectors, <u>l</u> , used in computing the fault handling unreliability QSUM
DBLDF	Real number in the range 0.1 to 0.01
Delta	Real number in the range $\geq 0.0$
Delta FHM	Character string set to either UNIFORM or EXPONENTIAL
Epsilon	Real number in the range $\geq 0.0$
Epsilon FHM	Character string set to ether UNIFORM or EXPONENTIAL
FOM	Character string set to either WEIBULL or EXPONENTIAL

Fault Type	Character string of up to 20 characters
Integration Steps	Integer in the range 17 - 64
Lambda	Real number in the range > 0.0
Minimum Number of Modules for Stage Operation	Non-negative integer which specifies the minimum number of modules the stage needs for operation
Mission Time	A real number > 0.0
Number of Beginning Modules in Stage	Positive integer with maximum value of 70
Number of Stages per Subrun	Number of stages 1 to 20 that a subrun contains provided it has no critically coupled modules
Omega	Real number in the range > 0.0
Output Option	Integer in the range 1 to 4
Pa	Real Number in the range 0.0 to 1.0
Pb	Real number in the range 0.0 to 1.0
QPTRNC	Parameter used to limit the number of fault vectors <u>l</u> used in computing the fault handling unreliability QSUM. A real number > 0.0 with default value of 1.0E-2.
Reliability Functions Plot	Logical variable of T or F
Reliability Y-Axis Selection	Character string set to either Y-Linear, Y- Log, Both or Log-log

Rho	Real number in the range $\geq 0.0$
Rho RHM	Character string set to either UNIFORM or EXponential
Set(s) of Modules Subject to Critical Pair Failures	NOP variable values
Stage Name	Character string up to 20 characters in length
Verify Correctness of Data	Logical variable of T or F

Standard Bibliographic Page

1. Report No. NASA CR-178251, Corrected Copy	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle CARE III User-Friendly Interface User's Guide		5. Report Date January 1987	
		6. Performing Organization Code	
7. Author(s) A. L. Martensen		8. Performing Organization Report No.	
		10. Work Unit No.	
9. Performing Organization Name and Address PRC Kentron, Inc. 303 Butler Farm Road, Suite 100 Hampton, VA 23666		11. Contract or Grant No. NAS1-18000	
		13. Type of Report and Period Covered Contractor Report	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		14. Sponsoring Agency Code 505-66-21-02	
		15. Supplementary Notes NASA Langley Project Engineer: Salvatore J. Bavuso Supersedes NASA CR-172608, Authors: C.L. Controd, W.F. Ingogly, S.E. McBride, and J.L. Pierce of the Research Triangle Institute.	
16. Abstract  This document provides an updated user's guide for use with the CARE3MENU computer program. This interactive, user-friendly program creates the input file used with the CARE III, Version VI computer program. The CARE3MENU program executes on the VAX 700 series computers under the VMS operating system, and makes use of the VAX/VMS Runtime Library.			
17. Key Words (Suggested by Author(s))  CARE III                      CARE3MENU Reliability Fault tolerance Coverage User Friendly Interface		18. Distribution Statement  Unclassified - Unlimited  Subject Category 61	
19. Security Classif.(of this report) unclassified	20. Security Classif.(of this page) unclassified	21. No. of Pages 87	22. Price A05