

AD-A245 435



2

IDA PAPER P-2549

A J-8 TACWAR GROUND COMBAT
ATTRITION PRIMERDTIC
ELECTE
DEC 09 1991
S D

Lowell Bruce Anderson

July 1991

Prepared for
Joint Chiefs of Staff

Approved for public release; distribution unlimited.

91 12 5 017

INSTITUTE FOR DEFENSE ANALYSES
1801 N. Beauregard Street, Alexandria, Virginia 22311-177291-17135

DEFINITIONS

IDA publishes the following documents to report the results of its work.

Reports

Reports are the most authoritative and most carefully considered products IDA publishes. They normally embody results of major projects which (a) have a direct bearing on decisions affecting major programs, (b) address issues of significant concern to the Executive Branch, the Congress and/or the public, or (c) address issues that have significant economic implications. IDA Reports are reviewed by outside panels of experts to ensure their high quality and relevance to the problems studied, and they are released by the President of IDA.

Group Reports

Group Reports record the findings and results of IDA established working groups and panels composed of senior individuals addressing major issues which otherwise would be the subject of an IDA Report. IDA Group Reports are reviewed by the senior individuals responsible for the project and others as selected by IDA to ensure their high quality and relevance to the problems studied, and are released by the President of IDA.

Papers

Papers, also authoritative and carefully considered products of IDA, address studies that are narrower in scope than those covered in Reports. IDA Papers are reviewed to ensure that they meet the high standards expected of refereed papers in professional journals or formal Agency reports.

Documents

IDA Documents are used for the convenience of the sponsors or the analysts (a) to record substantive work done in quick reaction studies, (b) to record the proceedings of conferences and meetings, (c) to make available preliminary and tentative results of analyses, (d) to record data developed in the course of an investigation, or (e) to forward information that is essentially unanalyzed and unevaluated. The review of IDA Documents is suited to their content and intended use.

The work reported in this document was conducted under contract MDA 903 89 C 0003 for the Department of Defense. The publication of this IDA document does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official position of that Agency.

This Paper has been reviewed by IDA to assure that it meets high standards of thoroughness, objectivity, and appropriate analytical methodology and that the results, conclusions and recommendations are properly supported by the material presented.

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 other 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 1991	3. REPORT TYPE AND DATES COVERED
4. TITLE AND SUBTITLE A J-8 TACWAR Ground Combat Attrition Primer			5. FUNDING NUMBERS C-MDA 903 89 C 0003 TA-T-16-682
6. AUTHOR(S) Lowell Bruce Anderson			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Institute for Defense Analyses 1801 N. Beauregard Street Alexandria, VA 22311			8. PERFORMING ORGANIZATION REPORT NUMBER IDA Paper P-2549
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Under Secretary of Defense for Acquisition The Pentagon Washington, DC 20301			10. SPONSORING/MONITORING AGENCY REPORT NUMBER
11. SUPPLEMENTARY NOTES			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12b. DISTRIBUTION CODE
13. ABSTRACT (Maximum 200 words) This paper provides an explanation and simple numerical example for the way that (conventional) attrition to ground weapon systems is determined in the current J-8 version of the TACWAR computer model of ground and air warfare. The purpose for presenting this explanation and example is to assist analyst in improving their understanding of the current ground combat attrition algorithms in J-8's TACWAR. Such increased understanding could lead to more appropriate use of the model and could facilitate decisions concerning model improvement.			
14. SUBJECT TERMS Numerical Example, Theater-Level Decision Making, Force Ratios, Weapons Systems Scores, Antipotential Potential, Model Logic Errors, Computer Programming Errors			15. NUMBER OF PAGES 314
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

IDA PAPER P-2549

A J-8 TACWAR GROUND COMBAT ATTRITION PRIMER

Lowell Bruce Anderson

July 1991

Approved for public release; distribution unlimited.



INSTITUTE FOR DEFENSE ANALYSES

Contract MDA 903 89 C 0003

Task T-I6-682

PREFACE

This paper was prepared under IDA contract MDA 903 89 C 0003 in partial fulfillment of Task Order T-I6-682, Net Assessment Methodologies and Critical Data Elements for Strategic and Theater Force Comparisons. This work was sponsored by the Capabilities Assessment Division of the Force Structure, Resource, and Assessment Directorate (J-8) of the Joint Chiefs of Staff.

This paper provides an explanation and simple numerical example for the simulation of conventional ground combat attrition in J-8's TACWAR.

The author is grateful to Dr. Robert J. Atwell, Project Leader; Dr. Peter S. Brooks; Mr. Edward P. Kerlin; Dr. Frederic A. Miercort; and Dr. Leo A. Schmidt--all of IDA--for their careful reviews of this paper.



Accession For	
DTIC CRASI	J
DTIC TAB	71
Unannounced	12
Justification	
By	
D. L. H. H. H.	
Availability Codes	
Dist	Availability Codes
A-1	

ABSTRACT

This paper provides an explanation and simple numerical example for the way that (conventional) attrition to ground weapon systems is determined in the current J-8 version of the TACWAR computer model of ground and air warfare. The purpose for presenting this explanation and example is to assist analysts in improving their understanding of the current ground combat attrition algorithms in J-8's TACWAR. Such increased understanding could lead to more appropriate use of the model and could facilitate decisions related to model improvement.

CONTENTS

Preface.....	iii
Abstract.....	v
Executive Summary.....	ES-1
I. Introduction.....	I-1
II. Discussion of Subroutine GC.....	II-1
A. Part A (Lines 1-1762).....	II-1
1. Section 1 (Lines 1-1613).....	II-1
2. Section 2 (Lines 1614-1665).....	II-3
3. Section 3 (Lines 1666-1740).....	II-3
4. Section 4 (Lines 1741-1762).....	II-6
B. Part B (Lines 1763-1941).....	II-7
1. Section 5 (Lines 1763-1806).....	II-7
2. Section 6 (Lines 1807-1896).....	II-8
3. Section 7 (Lines 1897-1941).....	II-9
C. Part C (Lines 1942-3040).....	II-9
1. Section 8 (Lines 1942-1945).....	II-9
2. Section 9 (Lines 1946-1992).....	II-10
3. Section 10 (Lines 1993-2010).....	II-10
4. Section 11 (Lines 2011-2132).....	II-10
5. Section 12 (Lines 2133-2166).....	II-10
6. Section 13 (Lines 2167-2210).....	II-11
7. Section 14 (Lines 2211-2293).....	II-11
8. Section 15 (Lines 2294-2542).....	II-11
9. Section 16 (Lines 2543-2577).....	II-11
10. Section 17 (Lines 2578-2638).....	II-12
11. Section 18 (Lines 2639-2697).....	II-13
12. Section 19 (Lines 2698-2911).....	II-13
13. Section 20 (Lines 2912-3017).....	II-13
14. Section 21 (Lines 3018-3040).....	II-17

D.	Part D (Lines 3041-3249).....	II-17
1.	Section 22 (Lines 3041-3115).....	II-17
2.	Section 23 (Lines 3116-3249).....	II-17
E.	Part E (Lines 3250-3942).....	II-17
1.	Section 24 (Lines 3250-3397).....	II-18
2.	Section 25 (Lines 3398-3605).....	II-18
3.	Section 26 (Lines 3606-3640).....	II-19
4.	Section 27 (Lines 3641-3771).....	II-19
5.	Section 28 (Lines 3772-3803).....	II-19
6.	Section 29 (Lines 3804-3891).....	II-19
7.	Section 30 (Lines 3892-3942).....	II-19
F.	Part F (Lines 3943-4168).....	II-19
1.	Section 31 (Lines 3943-3953).....	II-20
2.	Section 32 (Lines 3954-4075).....	II-20
3.	Section 33 (Lines 4076-4168).....	II-20
G.	Part G (Lines 4169-4239).....	II-21
1.	Section 34 (Lines 4169-4210).....	II-21
2.	Section 35 (Lines 4211-4239).....	II-21
H.	Part H (Lines 4240-4328).....	II-21
1.	Section 36 (Lines 4240-4309).....	II-21
2.	Section 37 (Lines 4310-4328).....	II-22
I.	Part I (Lines 4329-4598).....	II-22
1.	Section 38 (Lines 4329-4358).....	II-22
2.	Section 39 (Lines 4359-4370).....	II-22
3.	Section 40 (Lines 4371-4426).....	II-22
4.	Section 41 (Lines 4427-4501).....	II-22
5.	Section 42 (Lines 4502-4565).....	II-23
6.	Section 43 (Lines 4566-4598).....	II-23
J.	Part J (Lines 4599-4783).....	II-23
1.	Section 44 (Lines 4599-4607).....	II-23
2.	Section 45 (Lines 4608-4673).....	II-23
3.	Section 46 (Lines 4674-4702).....	II-24
4.	Section 47 (Lines 4703-4723).....	II-24
5.	Section 48 (Lines 4724-4783).....	II-24

III. A Numeric Example for Subroutine GC.....	III-1
A. Part A (Lines 1-1762).....	III-4
1. Section 1 (Lines 1-1613).....	III-4
2. Section 2 (Lines 1614-1665).....	III-4
3. Section 3 (Lines 1666-1740).....	III-4
a. Inputs and Outputs	III-4
b. Numeric Example.....	III-6
4. Section 4 (Lines 1741-1762).....	III-7
B. Part B (Lines 1763-1941).....	III-7
1. Section 5 (Lines 1763-1806).....	III-7
2. Section 6 (Lines 1807-1896).....	III-7
a. Inputs and Outputs	III-7
b. Numeric Example.....	III-8
3. Section 7 (Lines 1897-1941).....	III-8
C. Part C (lines 1942-3040).....	III-8
1. Section 8 (Lines 1942-1945).....	III-8
2. Section 9 (Lines 1946-1992).....	III-8
3. Section 10 (Lines 1993-2010).....	III-8
4. Section 11 (Lines 2011-2132).....	III-9
a. Inputs and Outputs	III-9
b. Numeric Example.....	III-10
5. Section 12 (Lines 2133-2166).....	III-10
a. Inputs and Outputs	III-10
b. Numeric Example.....	III-12
6. Section 13 (Lines 2167-2210).....	III-13
a. Inputs and Outputs	III-13
b. Numeric Example.....	III-14
7. Section 14 (Lines 2211-2293).....	III-14
8. Section 15 (Lines 2294-2542).....	III-14
9. Section 16 (Lines 2543-2577).....	III-14
a. Inputs and Outputs	III-14
b. Numeric Example.....	III-15
10. Section 17 (Lines 2578-2638).....	III-15
a. Inputs and Outputs	III-15
b. Numeric Example.....	III-17

11.	Section 18 (Lines 2639-2697).....	III-17
	a. Inputs and Outputs	III-17
	b. Numeric Example.....	III-18
12.	Section 19 (Lines 2698-2911).....	III-19
13.	Section 20 (Lines 2912-3017).....	III-19
	a. Inputs and Outputs	III-19
	b. Numeric Example.....	III-20
14.	Section 21 (Lines 3018-3040).....	III-20
D.	Part D (Lines 3041-3249).....	III-21
	1. Section 22 (Lines 3041-3115).....	III-21
	a. Inputs and Outputs	III-21
	b. Numeric Example.....	III-21
	2. Section 23 (Lines 3116-3249).....	III-22
	a. Inputs and Outputs	III-22
	b. Numeric Example.....	III-22
E.	Part E (Lines 3250-3942).....	III-24
	1. Section 24 (Lines 3250-3397).....	III-24
	a. Inputs and Outputs	III-24
	b. Numeric Example.....	III-25
	2. Section 25 (Lines 3398-3605).....	III-28
	a. Inputs and Outputs	III-28
	b. Numeric Example.....	III-30
	3. Section 26 (Lines 3606-3640).....	III-30
	a. Inputs and Outputs	III-30
	b. Numeric Example.....	III-31
	4. Section 27 (Lines 3641-3771).....	III-31
	a. Inputs and Outputs	III-31
	b. Numeric Example.....	III-32
	5. Section 28 (Lines 3772-3803).....	III-32
	6. Section 29 (Lines 3804-3891).....	III-33
	a. Inputs and Outputs	III-33
	b. Numeric Example.....	III-34
	7. Section 30 (Lines 3892-3942).....	III-36

F.	Part F (Lines 3943-4168).....	III-36
1.	Section 31 (Lines 3943-3953).....	III-36
2.	Section 32 (Lines 3954-4075).....	III-38
a.	Inputs and Outputs	III-38
b.	Numeric Example.....	III-39
3.	Section 33 (Lines 4076-4168).....	III-45
a.	Inputs and Outputs	III-45
b.	Numeric Example.....	III-46
G.	Part G (Lines 4169-4239).....	III-47
1.	Section 34 (Lines 4169-4210).....	III-47
a.	Inputs and Outputs	III-47
b.	Numeric Example.....	III-48
2.	Section 35 (Lines 4211-4239).....	III-49
a.	Inputs and Outputs	III-49
b.	Numeric Example.....	III-49
H.	Part H (Lines 4240-4328).....	III-50
1.	Section 36 (Lines 4240-4309).....	III-50
a.	Inputs and Outputs	III-50
b.	Numeric Example.....	III-51
2.	Section 37 (Lines 4310-4328).....	III-52
a.	Inputs and Outputs	III-52
b.	Numeric Example.....	III-53
I.	Part I (Lines 4329-4598).....	III-53
1.	Section 38 (Lines 4329-4358).....	III-53
a.	Inputs and Outputs	III-53
b.	Numeric Example.....	III-54
2.	Section 39 (Lines 4359-4370).....	III-54
a.	Inputs and Outputs	III-54
b.	Numeric Example.....	III-55
3.	Section 40 (Lines 4371-4426).....	III-55
a.	Inputs and Outputs	III-55
b.	Numeric Example.....	III-56
4.	Section 41 (Lines 4427-4501).....	III-57
5.	Section 42 (Lines 4502-4565).....	III-57
6.	Section 43 (Lines 4566-4598).....	III-57

J.	Part J (Lines 4599-4783).....	III-58
1.	Section 44 (Lines 4599-4607).....	III-58
2.	Section 45 (Lines 4608-4673).....	III-58
3.	Section 46 (Lines 4674-4702).....	III-58
4.	Section 47 (Lines 4703-4723).....	III-58
5.	Section 48 (Lines 4724-4783).....	III-58
References.....		R-1
Appendix	A FORTRAN Listing of Subroutine GC	

TABLES

Table II-1	A Subdivision of Subroutine GC into Parts and Sections.....	II-2
Table III-1	Values of the Non-Zero Bookkeeping and Miscellaneous Inputs Used in this Example.....	III-2
Table III-2	Values of the Relevant Non-Zero Inputs Used in this Example.....	III-3
Table III-3	Unscaled Attrition Calculations.....	III-41

EXECUTIVE SUMMARY

A. BACKGROUND

In 1988 and 1989, IDA developed a comprehensive structure for assessing point fire attrition in large-scale (e.g., theater-level) models. This attrition assessment structure, which is documented in IDA Paper P-2248, allowed multiple types of weapons on both sides. It did not explicitly consider multiple types of munitions, however, and it considered point fire but not area fire. The paper suggested further research to address multiple types of munitions and area fire.

The Capabilities Assessment Division (CAD) of the Force Structure, Research, and Assessment Directorate (J-8) of the Joint Chiefs of Staff sponsored additional research, which led to the publication of IDA Papers P-2249 and P-2250. These papers present a comprehensive structure for assessing both point fire and area fire attrition in theater-level models, where the attrition assessment can consider both multiple types of weapons and multiple types of munitions for the two sides in combat.

Subsequently, J-8/CAD funded the development of computer code to incorporate this attrition assessment structure into their version of TACWAR. This incorporation did not remove any existing capabilities from the model. Instead, it added as options capabilities to assess ground attrition as described in P-2249 and P-2250. The documentation of the new code is contained in IDA Document D-767, and an annotated briefing describing this code is given in IDA Document D-794.

J-8 decided not to incorporate this code into the official J-8 version of the TACWAR model. J-8 felt that its analysts did not fully understand the existing ground attrition assessment procedure in the model. Further, J-8 felt that it would be inappropriate to add new attrition assessment procedures to the model until its analysts had a better understanding of their current procedure.

J-8 asked IDA for a numeric example of the current ground attrition assessment procedure which would help its analysts understand that procedure and concomitantly help them evaluate the merits of the new attrition structures that IDA had prepared. The

goal of this paper is to present such an example and to explain the current ground attrition assessment procedure in terms of this example.

Given this goal, two issues are raised: (1) What should the form of the example be? and (2) What should be done if any errors in the code are found in the course of developing this example? These two issues are discussed in the next two sections below.

B. THE FORM OF THE EXAMPLE

In coordination with representatives from J-8, it was decided to structure the example as follows. The example would be complete and would emphasize the attrition procedure, but it would not be full-featured. That is, it would start at the beginning of Subroutine GC (ground combat) and would continue through the end of that subroutine; however, certain input switches would be set so that certain parts of the code of GC would not be exercised. For instance, the example assumes that no degradations due to supply shortages are occurring, and the relevant switches and inventories are set accordingly. The example also assumes that no degradation due to possible chemical or tactical nuclear combat is occurring, and the relevant switches and data are set accordingly.

It also was decided that the numeric example should be unclassified and relatively simple, and that no attempt should be made to make it realistic. Consequently, the goal of this paper is to produce an example that will help a reader understand the calculations and attrition mechanisms in Subroutine GC of J-8's TACWAR, not to test whether generally realistic inputs would give generally realistic outputs. Similarly, this paper does not attempt to duplicate by hand the repetitive nature of what computers can do in terms of taking many types of weapons on each side and producing page after page of numbers as outputs. Instead, the example is purposely small enough so that every calculation could reasonably be made using only pen, paper, and a hand calculator. Accordingly, a reader can trace each and every calculation as it is described here. This form of the numeric example is quite consistent with the underlying goal—to enhance understanding of the attrition procedure in Subroutine GC. This goal of achieving enhanced understanding means that the relationship between the numbers in the example and the corresponding part of the code is more important than the individual values of the numbers themselves. Thus, in addition to reporting the numeric values as calculated by GC, it is important here to explain the procedures and structures used in the attrition portion of GC.

The first chapter of the text is a brief introduction. The second chapter contains a non-numeric discussion of the logic and structure of GC with emphasis on the attrition calculations. The third chapter presents the numeric example. The individual sections of the third chapter frequently contain two subsections; the first discusses the particular logic and rationale for any nuances in the calculations, and the second gives and discusses numeric values for the calculations being made.

Thus, this paper gives more than just a numeric example of the attrition calculations in GC—it also gives a careful explanation of those calculations. However, it is not intended to be a documentation of Subroutine GC. This paper is intended to supplement such documentation by enhancing understanding of the attrition calculations of Subroutine GC.

C. PROBLEMS IN THE CODE

Unlike the form of the example, which was carefully decided upon in close coordination with representatives of J-8, the handling of problems found in the code was not considered in advance. Not surprisingly, problems were found. They ranged from what could be called stylistic nuances through errors that could significantly affect the results of the model if the section of the code containing those errors were entered in a manner that caused the errors to occur. Owing to several factors, including the goal of enhancing understanding of the current calculations, the time constraints involved, and the fact that these problems were uncovered on a piecemeal basis, the project deemed it inappropriate to divert significant effort from preparing and explaining the numeric example in order to prepare a critical review of GC. Instead, problems are discussed as part of the discussion of the code, not in a comprehensive but separate discussion.

Discussing problems in-line with explanations of the code inhibits extensive discussions of alternative corrections. Fortunately, for each problem that has the potential to be computationally significant, there seems to be at least one solution or fix which is very easy to implement. It is important to note that not all of these quick and easy-to-implement solutions are necessarily the best solutions, but each possible problem has at least one easy-to-implement solution that could be used.

I. INTRODUCTION

Each version of TACWAR is a FORTRAN computer program intended to model ground and air combat throughout a theater of warfare, where this combat can include conventional, tactical nuclear, and chemical combat. Conventional ground combat in the model is (currently) modeled by its Subroutine GC. The primary purpose of this paper is to explain how Subroutine GC determines attrition to ground weapons systems in the current J-8 version of TACWAR. A listing of the current J-8 version of Subroutine GC is given in the Appendix and is used to discuss the subroutine in detail below.

Some introductory notes may facilitate the reading of this paper. First, the explanations in the body of this paper apply to the current J-8 version of TACWAR, not necessarily to previous versions or to any IDA version of TACWAR.¹ Second, the explanations here concern only Subroutine GC; no other part of the model is discussed. Third, future expansion of the model might also allow modeling conventional ground combat using Subroutine GC90 instead of Subroutine GC; the explanations here apply to GC, not to GC90. Fourth, the fact that Subroutine GC is being described here does not imply that the use of Subroutine GC is being recommended. See References [1], [2], and [3] for further discussion of the relationship between Subroutine GC and Subroutine GC90. Fifth, the model simulates combat as occurring over multiple time periods (or, equivalently, cycles) where the number of time periods to be simulated, NCYCLE, is an input. Currently Subroutine GC is called exactly once for each time period simulated by Program TACWAR, which is the (main) program of J-8's TACWAR model. This structure should be kept in mind when reading the explanations below. Sixth, somewhat

¹ NOTICE: Throughout this paper, the words "current" and "currently," when applied to J-8's TACWAR or to any part of the TACWAR code, mean "current as of September 18, 1990." Clearly, the code can be changed over time. In particular, it is likely and quite reasonable that, as this paper is being reviewed and prepared for publication, the model will be changed to address at least some of the problems described herein. Indeed, this paper urges that these errors be addressed immediately. Modification should not be deferred until this paper is fully reviewed and published. Recognizing that the process of correcting errors may be incomplete by the time this paper is fully reviewed and published, and that code changes in the meantime may introduce new errors, this paper identifies errors in relation to the code in the Appendix so that a potential user of the model can compare the code being considered for use with the code in the Appendix to see which errors have been addressed, how they were addressed, which errors remain, and which parts of the code differ and so should be checked for other (though perhaps related) errors.

short English-language definitions of all input variables and arrays and of selected working variables and arrays are given in References [4] and [5], respectively, and it may be useful to have a copy of these references available when reading the explanations below.

The approach taken below may be summarized as follows. First, while all of the code of GC is described in general, only selected portions of it are discussed in detail. For example, it is assumed here that combat is to be simulated as if both sides had adequate supplies (i.e., the input value for ISEFFD is assumed to be 3). Thus, the portions of the code of GC that deal with supply shortages are generally summarized by simply pointing out this fact. The reader interested in obtaining a full understanding of GC, including supply shortages, could use the description here to obtain a basic understanding of GC, and then could examine the code (or ask experienced analysts) for more information.

Second, the operations of GC are illustrated using a simple numeric example. As just noted, this example assumes that no degradations due to supply shortages are occurring (ISEFFD = 3). Other simplifying features of the example are as follows:

- Only two different types of ground weapons on each side are represented ($NW(L) = 2$).
- Only nonpreferred munitions are represented ($ISMUS(L) = 0$ and $NSM(L) = 0$).
- Only one type of aircraft carrying only one type of air-to-ground munition is represented ($NAC(L) = NAM(L) = 1$).
- No surface-to-surface missile systems are represented ($NSSM(L) = NMM(L) = 0$).
- Only one sector of combat is represented ($NS = 1$).
- Posture and terrain considerations are greatly simplified.
- Multidivisional considerations and considerations concerning distances from the front and weapons being allowed on the front are greatly simplified.
- No degradation due to possible chemical or tactical nuclear combat is considered ($IOMU = 5$).

A full set of numeric calculations is given, however, for those aspects of GC that are considered. That is, sample (but entirely hypothetical) numeric values for all of the required inputs are postulated, and the resulting numeric values of all major working

variables and arrays are given. Interested readers could consider this numeric example as a base upon which to build a better understanding of J-8's TACWAR.

Finally, while the results that would be obtained under several different options are described, only some (not all) possible options are considered. For example, the discussion describes neither the calculations and results for options invoked by setting the value of the input variable MVCASO to 2 or 3 nor the impact on the output of various print options.

The discussion of Subroutine GC is organized as follows. Chapter II subdivides the code of Subroutine GC (given in the Appendix) into parts and sections and discusses the logic of those parts and sections, but it does not discuss the numeric example. Chapter III presents the numeric example in terms of relevant variables and arrays according to those parts and sections. Accordingly, a reader may occasionally (or regularly) want to skip back and forth among the corresponding parts and sections of Chapters II and III, the code in the Appendix, and the definitions of variables and arrays given in [4] and [5].

II. DISCUSSION OF SUBROUTINE GC

Subroutine GC is relatively long. The code of GC listed in the Appendix consists of 2,393 statement lines (of which 166 are continuation lines) and 925 comment lines (of which 252 are essentially blank, and well over one third of the remaining 673 comment lines are "commented-out dead code" which should be removed). Thus, in total, that code is 3,318 lines long (counting each included file as only one line, and counting all of the comment lines, meaningful or not). As the line numbers in the Appendix indicate, if the lines in the included files are also counted, GC is 4,783 lines long (i.e., there are 1,465 lines in those included files).

One way to discuss long code is to divide it into sections and to organize the discussion according to those sections—this is done here. There are many very different ways to subdivide the GC code, and no one way is inherently superior to all others for all purposes. For the purposes here, the code of GC in the Appendix is subdivided into 10 parts (labeled from Part A through Part J), which in turn are subdivided into two or more sections for a total (over all parts) of 48 sections. These subdivisions are defined using the line numbers listed in the Appendix, and the particular line numbers that bound these subdivisions are given on Table II-1.

A. PART A (LINES 1-1762)

Subroutine GC is called by Program TACWAR once for each time period (i.e., cycle) being simulated. Most of the code of GC is contained inside a very long do-loop over sectors (Parts B through I comprise this do-loop). Part A consists of the code from the first line of GC up to the start of this do-loop.

1. Section 1 (Lines 1-1613)

Section 1 consists of the subroutine statement, the specification statements (many of which are in the included files), and some (initial) comment lines. This "front matter" is generally necessary but not particularly informative for an introductory look at GC.

Table II-1. A Subdivision of Subroutine GC Into Parts and Sections

<u>Part</u>	<u>Lines</u>	<u>Code Section</u>	<u>Lines</u>	<u>Text Section</u>
A	0001-1762	1	0001-1613	A1
		2	1614-1665	A2
		3	1666-1740	A3
		4	1741-1762	A4
B	1763-1941	5	1763-1806	B1
		6	1807-1896	B2
		7	1897-1941	B3
C	1942-3040	8	1942-1945	C1
		9	1946-1992	C2
		10	1993-2010	C3
		11	2011-2132	C4
		12	2133-2166	C5
		13	2167-2210	C6
		14	2211-2293	C7
		15	2294-2542	C8
		16	2543-2577	C9
		17	2278-2638	C10
		18	2639-2697	C11
		19	2698-2911	C12
		20	2912-3017	C13
		21	3018-3040	C14
D	3041-3249	22	3041-3115	D1
		23	3116-3249	D2
E	3250-3942	24	3250-3397	E1
		25	3398-3605	E2
		26	3606-3640	E3
		27	3641-3771	E4
		28	3772-3803	E5
		29	3804-3891	E6
		30	3892-3942	E7
		31	3943-3953	F1
F	3943-4168	32	3954-4075	F2
		33	4076-4168	F3
		34	4169-4210	G1
G	4169-4239	35	4211-4239	G2
		36	4240-4309	H1
H	4240-4328	37	4310-4328	H2
		38	4329-4358	I1
I	4329-4598	39	4359-4370	I2
		40	4371-4426	I3
		41	4427-4501	I4
		42	4502-4565	I5
		43	4566-4598	I6
		44	4599-4607	J1
J	4599-4783	45	4608-4673	J2
		46	4674-4702	J3
		47	4703-4723	J4
		48	4724-4783	J5

Sections 2, 3, and 4 consist of the executable statements and related comment lines between the specification statements and the start of the huge do-loop over sectors mentioned above. Some of the calculations made here are important for a first look at GC; others are not.

2. Section 2 (Lines 1614-1665)

Lines 1614 through 1630 concern which side is labeled as being the probable sector attacker. Some problems here are as follows. The statement on line 1620 is wrong (instead, L should be set equal to ISAT(ISA(IS),IS)), and the use of ICSA on line 1621 is inconsistent with its use elsewhere in the code—and these uses are generally inconsistent with reasonable interpretations of the definition of ICSA in Reference [4]. For the purposes here, assume that Side 1 is the probable sector attacker and also will (in Section 27) be declared to be the "actual" sector attacker, and assume that the posture in Sector 1 for the initial value of FEBA(1) is posture type 1.

Lines 1631 through 1640 concern printing outputs.

Lines 1641 through 1665 concern surface-to-surface missile systems (SSMs). For the purposes here, assume that no SSMs are involved on either side.

3. Section 3 (Lines 1666-1740)

From a sufficiently abstract point-of-view, the ground combat attrition structure is quite simple: it allocates fire to targets by type of target, it calculates attrition due to this fire, and then it (optionally) scales this calculated attrition. Lines 1666 through 1740 concern (a portion of) the calculation of the allocation of fire.

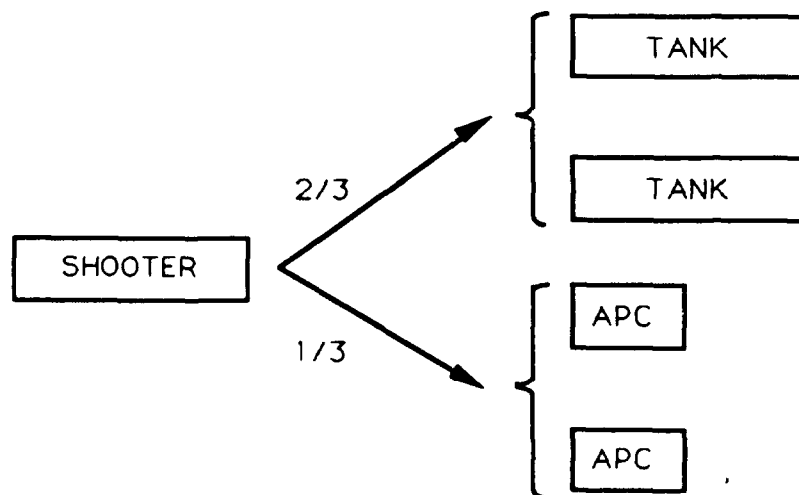
The basic logic is as follows. Suppose, for example, that anti-armor weapons generally prefer to shoot at enemy tanks but sometimes may shoot at armored personnel carriers (APCs) because —

- Occasionally one or more APCs may be between any engageable tank and the shooting weapon in question.
- An APC may be imminently threatening that shooting weapon (or another weapon on its side).
- The shooting weapon may have a "safe" shot against an APC, but (moving and) shooting at a tank would expose it to enemy fire.

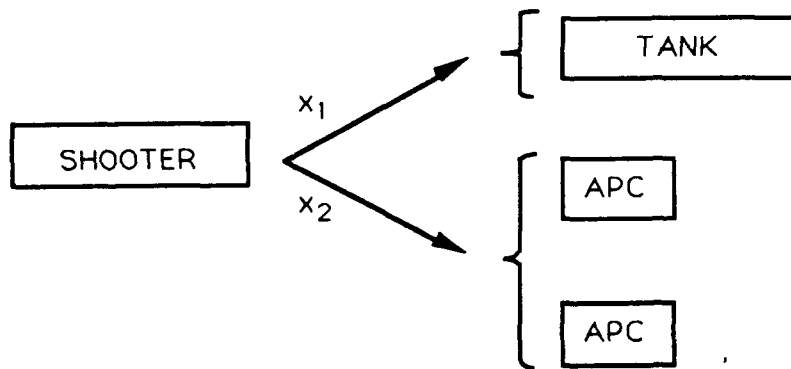
- The relative locations and orientations are such that the shooting weapon may have a much better shot at an APC than it has at any tank.
- The particular tactical situation may (temporarily) make an APC a much more valuable target than the tanks that are present—e.g., the APC is occupying an important piece of terrain, or the APC is full of fresh troops and supplies.

In addition, while it is the case that if the enemy force were to consist of almost all tanks and almost no APCs, then almost all fire would be directed against tanks; it also must be the case that if the enemy force consisted of almost all APCs and almost no tanks, then almost all fire would be directed against APCs.

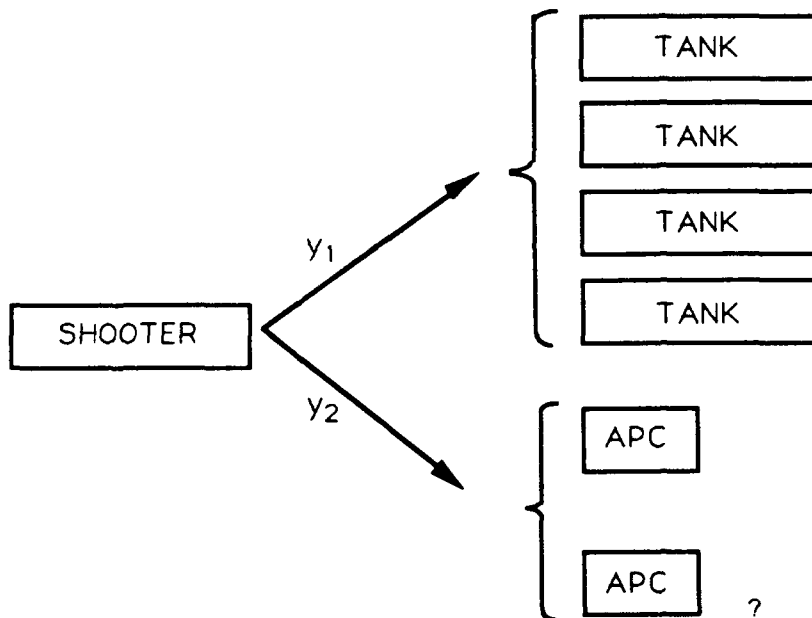
To construct a simple example, suppose that a certain type of shooting weapon generally prefers to fire at tanks, but occasionally will fire at APCs. Suppose also that the allocation of fire of the shooters against these two types of targets varies as the relative numbers of these two types vary. In particular, suppose it is known that if 2 enemy tanks and 2 enemy APCs were present, then a shooting weapon of that type would, on average, make two-thirds of its engagements against tanks and one-third of its engagements against APCs. Given this knowledge, what should the model estimate the allocation of fire to be if 1 enemy tank and 2 enemy APCs are present, and what should this estimate be if 4 enemy tanks and 2 enemy APCs are present? A graphic representation of this example appears below.



then what should x_1 and x_2 be for



and what should y_1 and y_2 be for



Obviously, there is no single correct answer to this question that covers all special cases; but reasonable (on average) values would appear to be as follows:

$$x_1 = 1/2$$

$$x_2 = 1/2,$$

and

$$y_1 = 4/5$$

$$y_2 = 1/5.$$

That is, if a shooter is twice as likely to shoot at tanks than at APCs when they are present in equal numbers, then that shooter is equally likely to shoot at tanks as at APCs when there are twice as many APCs as there are tanks present, and it is four times as likely to

shoot at tanks than at APCs when there are twice as many tanks as there are APCs present.

Detailed discussions of various aspects of this method for determining allocation of fire can be found in several references: [6, Chapter II], [7, pages 98 through 100], [8, pages 31 and 32], [9, pages 42 and 43, and 53 and 54], [10, pages 4 through 8], and [11, pages III-1 through III-9].

Basically, however, this allocation-of-fire method only does the following. It takes as input some typical allocation against some typical force, and then computes a particular allocation against the particular force present in a manner that is consistent with those inputs and with the following property: the more heavily represented any type of weapon is in a force, the more fire that type of weapon will draw (in total, not per weapon of that type present), and vice versa.

This calculation of allocation-of fire can be divided into two steps. In the first step, the input allocation against the input typical force is converted to a per-target-weapon allocation (i.e., the allocation that would "theoretically" apply if there were one weapon of each type present in the target force). The second step converts the per-target-weapon allocation into the allocation to be used against the particular force in question. This two-step approach is unnecessary, but it is embedded in J-8's TACWAR code in that lines 1666 through 1740 address this first step (for nonpreferred munitions), while the second step is considered much later in the code (e.g., for ground weapons, on lines 2141 through 2165 in Step 10).

4. Section 4 (Lines 1741-1762)

Lines 1741 through 1752 simply initialize some arrays that involve sums of data over all sectors, and so must be initialized before the sector loop.

Lines 1753 through 1755 concern lines 1764 through 1779 in code Section 5 (Part B)—they have nothing to do with the executable code in Part A.

There are several arguments against giving the input MVCASO a value of zero, as discussed in References [2] and [12]. Lines 1756 through 1762 provide another reason why MVCASO should not be zero. In particular, $MVCASO = 0$ scales simulated attrition using a force ratio. For interdiction, there is no force ratio. Instead, the model uses unweighted averages of essentially arbitrary combat data to generate scale factors for

interdiction. These lines are used to generate a scaling factor even if no combat is occurring.

B. Part B (Lines 1763-1941)

As noted above, much of GC is contained in a huge do-loop over sectors. From a broad perspective, for each sector this loop (a) calculates kill-rate matrices, (b) calculates weapons systems scores, (c) simulates attrition, (d) optionally scales that simulated attrition, and (e) assesses the (optionally scaled) attrition. From a slightly narrower perspective, this loop can be further subdivided, and Parts B through I form a slightly finer subdivision of this loop. In particular, Part B performs some initial calculations for the sector in question, then Part C, which consists entirely of a long do-loop over side, computes kill-rate matrices based on various inputs and on the results of the initial calculations. Accordingly, Part B runs from the start of the huge do-loop over sectors to the start of the long do-through-1050 loop over side.

1. Section 5 (Lines 1763-1806)

Line 1763 starts this (very) long do-loop over side.

Lines 1764 through 1779 set IHOLD(L) and IDRAW(IS,L) to 0, and then can reset them if the index to nuclear combat mode ICMST(IS,L) is not zero. The example in Chapter III assumes that ICMST(IS,L) is zero. However, note that lines 1771 and 1773 appear to be wrong, causing this code to produce erroneous results if ICMST(IS,L) is not zero. Lines 1780 through 1786 write IHOLD and IDRAW to an output file.

With one exception, lines 1787 through 1802 are comment lines—line 1797 sets the index of terrain type KTI. See Point (7) of (text) Section 2, below, for a discussion of an apparently significant error related to line 1797.

Line 1803 selects a side to be the proposed sector attacker. Roughly speaking, the proposed sector attacker is given first choice as to whether it wants to attack or not, and it is required to attack (even if it doesn't want to) if the other side (also) doesn't want to attack.

Lines 1804 through 1806 use the array ICSA(IS) to make a potentially important assumption concerning posture under a particular set of conditions. (Some aspects of the way that the model considers posture will be discussed further below.) For the purposes

here, assume that these conditions are not occurring—in particular, assume that $ICSA(IS) = 1$.

These calculations, including the determination of which side is to be the sector attacker, can have a major impact on the results of the model (in general) and of GC (in particular). The impact on GC is largely in terms of determining which data are to be used in the attrition calculations, not in determining how the calculations are made given the data. Thus, while it is necessary to understand these calculations in order to understand TACWAR, it is not necessary to understand them in order to understand the details of the simulation of attrition in GC.

2. Section 6 (Lines 1807-1896)

The primary goal of Section 6 is to subdivide the ground weapons in the sector into two groups: (1) those with the capability both to engage some enemy ground weapons and to be engaged by some enemy ground weapons, and (2) those that can neither engage enemy ground weapons nor be engaged by enemy ground weapons. All of these ground weapons can be engaged by enemy aircraft and SSMs and, in GC, none of these ground weapons can engage enemy aircraft and SSMs. As shown in the following tabulation, the number of ground weapons that can engage only "vulnerable" enemy ground weapons but are vulnerable to both enemy air and ground weapons is computed as $WSI(IW, IS, L)$, and the number of ground weapons that cannot engage any enemy ground weapons but are vulnerable to enemy air weapons is calculated as $WS(IW, L)$ minus $WSI(IW, IS, L)$. The total number of ground weapons of type IW on side L considered, $WS(IW, L)$, is an implicit function of IS in the code.

Capability of Ground Weapons as Shooters	Vulnerability of Ground Weapons as Targets			
	Vulnerable to enemy air and ground weapons	Vulnerable only to enemy air weapons	Not Vulnerable	Total
Can engage all enemy ground weapons	nc	nc	nc	nc
Can engage only "vulnerable" enemy ground weapons	WSI	nc	nc	WSI
Cannot engage any enemy ground weapons	nc	WS minus WSI	nc	WS minus WSI
TOTAL	WSI	WS minus WSI	nc	WS

nc = not computed

3. Section 7 (Lines 1897-1941)

Section 7 sets some initial values for working arrays prior to the upcoming do-loop over side which constitutes Part C. After initializing these arrays, it skips to the end of the do-loop over sector if there are no ground weapons present on one or the other (or both) of the sides (line 1930). It also skips to the start of Section 15 if MVCASO = 2 and ICYCLE > 1; however, note that MVCASO = 2 causes allocations of fire to be determined by forces that are not there (see Reference [12] for details).

C. PART C (LINES 1942-3040)

Part C consists of the long do-loop over side that starts at line 1942 and ends at line 3040. Because this loop is so large, it is divided into 14 sections for the discussion here. The main purpose of this loop is to calculate the kill rate matrices PWAKW(IW,JW,L) and PWDKW(IW,JW,L), which is done in Section 18 (lines 2639-2997). These rates can be adjusted (for artillery and helicopters) in Section 21 (lines 3018-3040).

1. Section 8 (Lines 1942-1945)

Line 1942 starts the do-loop over side, where L denotes the side in question. Line 1943 sets II to denote the other side. Lines 1944 and 1945 could be deleted, and the inputs NW(L) and NW(II) could be used directly where needed.

2. Section 9 (Lines 1946-1992)

This section concerns only the possible use of preferred munitions, a detailed discussion of which is beyond the intended purposes of this paper.

3. Section 10 (Lines 1993-2010)

This section's sole purpose is to set posture indices.

4. Section 11 (Lines 2011-2132)

This section consists entirely of the implementation of a really bad idea. Basically, this idea is to input allocations of fire by groups of types of weapons instead of just by types of weapons. This approach is inefficient in that it requires extra code (e.g., this section), it requires extra inputs (the grouping structure must be input as well as the assignment of types to groups), it requires extra running time (e.g., to run this section), it requires extra storage space (both the input allocations by group and the resulting allocations by type must be stored), it limits flexibility (if two different types of weapons are in the same group, one cannot be preferentially engaged over the other), and it adds no additional capabilities (anything that can be done by weapon groups can also and more easily be done by weapon types). All it does is to add an unnecessary layer of complexity which impedes understanding the allocation process. This section of code should be eliminated. But in the meantime, the input LNKWPT should be set so that $LNKWPT(IW,L) = IW$ for all IW and both L. If this is done, then PWIC is always equal to 1.0, and this section of the code does nothing but set each element of the arrays SAWA, SAWD, BBFRASM, BBFRDSM, RRFRASM, RRFRDSM, SSAWA, SSAWD, SSAWWA, and SSAMMD equal to the corresponding element in BSAWA, BSAWD, RSAWA, RSAWD, BFRASM, BFRDSM, RFRASM, RFRDSM, SAMA, SAMD, and SAMMA, and SAMMD.

5. Section 12 (Lines 2133-2166)

Lines 2133 to 2140 (and 2166) concern the option to set $MVCASO = 3$. This option, like $MVCASO = 2$, causes allocations of fire to be specified by forces which are not there.

Lines 2141 through 2165 complete the allocation of fire computations for ground weapons started on lines 1666 through 1716 of Section 3. At this point in the code (i.e., at line 2165), $AAWA(IW,IW,II)$ gives the allocation of fire of ground weapons of type

JW on Side II over all types of munitions if Side II is to be the attacker (and AAWD is the same for Side II on defense). Later (in Section 16), the fraction of the engagements against enemy weapons of type IW that are made using preferred munitions are subtracted from AAWA and AAWD; so after that section, AAWA(JW,IW,II) and AAWD(JW,IW,II) give the fraction of the engagements made by ground weapons of type JW on side II that are made against enemy weapons of type IW using nonpreferred munitions.

6. Section 13 (Lines 2167-2210)

This section does nothing but set the elements of the arrays EERTA and EERTD equal to the corresponding elements of the input arrays ERTA and ERTD. Since EERTA and EERTD are then used without modification, this code should be deleted and (instead) ERTA and ERTD should be used throughout. Note that line 2170 is a comment line (not a statement line) that serves no purpose, as are lines 2177 through 2210. These comment lines (i.e., all of this "dead code") could be deleted.

7. Section 14 (Lines 2211-2293)

Like Section 9 (Lines 1946-1992), this section concerns only the possible use of preferred munitions, a detailed discussion of which is beyond the intended purposes of this paper.

8. Section 15 (Lines 2294-2542)

This section consists of nothing but "dead code," which could be deleted.

9. Section 16 (Lines 2543-2577)

While the details of the simulation of preferred munitions are beyond the intended purposes here, one general point may be worth noting. Looking at Subroutine GC, one might ask what the important differences are between the way it simulates preferred munitions and the way it simulates nonpreferred munitions. It appears that there are none. That is, while the data entered could be significantly different, and while there are some differences in the code concerning the type of data that can be entered and the way that the data are used, these code differences between the treatment of preferred munitions and the treatment of nonpreferred munitions seem to be quite minor.

Given the basic equivalence of this code, a significant simplification can be achieved by essentially deleting all coding distinctions between preferred munitions and nonpreferred munitions, and (instead) just simulating multiple types of munitions for each type of ground weapon. This can be done relatively easily, for example, by incrementing the value of NSM(L) by one (i.e., $NSM(L) \leftarrow NSM(L) + 1$) for both L, then playing the first (new) type of munition as nonpreferred munitions and playing the i^{th} (new) type of munition using the same stockage levels and effectiveness parameters as the i -minus-1st (old) type of preferred munition for $i = 2$ through one plus the old value of NSM(L) for both L. For example:

New Data		Old Data
NSM(L)	$\leftarrow$	NSM(L)+1
BPKASM(TW,JW,ISM+1)	$\leftarrow$	BPKASM(TW,JW,ISM)
BPKASM(TW,JW,1)	$\leftarrow$	BPKAW(TW,JW,1)
BFRASM(TW,JW,ISM+1)	$\leftarrow$	BFRASM(TW,JW,ISM)
BFRASM(TW,JW,1)	$\leftarrow$	$1.0 - \sum_{ISM=1}^{NSM(L)} BFRASM(TW,JW,ISM) .$

One of the simplifications that would result from this conversion is that all of Section 16 would be deleted.

10. Section 17 (Lines 2578-2638)

As discussed in References [2], [11], [13], and [14], three fundamental parameters concerning the calculation of attrition for (one-sided) heterogeneous point fire are: (1) the engagement rates of the various types of shooters, (2) the allocation of fire of various types of shooters against the various types of targets, and (3) the probability of kill of the various types of targets by the various types of shooters. Of the many attrition formulas presented in those references, exactly one (Lanchester) has the property that engagement rates and probabilities of kill always appear as factors of each other. This one attrition formula is the only one used in GC to compute (simulated) losses, and these lines calculate the products of these factors.

11. Section 18 (Lines 2639-2697)

Not only are the Lanchester formulas the only ones in which engagement rates and probabilities of kill always appear as factors of each other, these formulas are the only ones that have the property that all three parameters (engagement rates, probabilities of kill, and allocations of fire) always appear as factors of each other. Accordingly, this section multiplies the third factor (allocation of fire) by the product of the other two (as computed in Section 10) to give the overall product (i.e., the kill rate) for each "class/type" of munition. Summing the kill rate for the one generic type of nonpreferred munition with the kill rates for each of the preferred types of munitions gives the per-shooter kill rate (over all of the types of munitions being used) for each type of shooter against each type of target. With one class of exceptions (discussed in Section 21), these overall kill rates are calculated at the end of Section 18.

It should be noted that computing kill rates as the product of these three factors is not the only (or even the best) way to do so. See References [3] and [15] for discussions of other methods and for general recommendations.

12. Section 19 (Lines 2698-2911)

This section only writes selected results to output files, if desired. It performs no substantive calculations.

13. Section 20 (Lines 2912-3017)

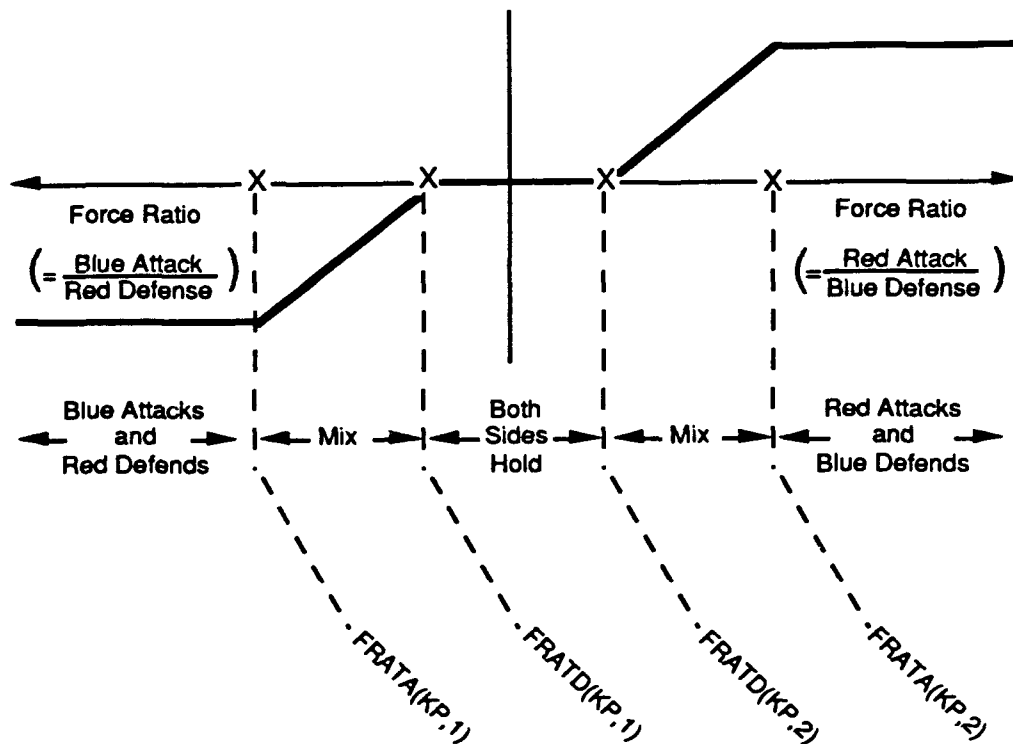
This section performs some relatively simple calculations and raises some potentially important issues.

The computations are almost straightforward. The reason for the "almost" is that calculations made in Section 20 assume that $ISAT(ISA(IS), IS)$ gives the side on attack in the sector. In general, however, $ISAT$ gives the side that last attacked in that sector and that therefore will likely, but not necessarily, be the side that will attack during the current time period. Since $ISAT$ does not necessarily give the side that will attack, this code could be wrong. In addition to occurring here, this potential problem occurs on lines 1804, 1814, 2004, 2924, 3094, and 3282, and the corresponding problem concerning $ISA(IS)$ definitely occurs on lines 1620, 3924, and 4280, and appears to also occur on lines 1768, 1771, and 3743.

Because the input data determine whether this problem will occur and how seriously it will affect results, its impact cannot be stated absolutely. First, in every run made so far, Side 2 could have always been attacking in all of the sectors. Given that assumption, this coding problem would not yet have had any affect on the results, but every future run could be subject to this problem until it was corrected or the data were restricted to prevent this problem from affecting the results. Second, the data used could have been such that, while this problem could have occurred, it would not have had any effect on the results. While theoretically possible, this scenario probably is not what has happened. Third, perhaps this problem has occurred here, and the error has affected some (but not all) of the results, but none of the results it affected have been that important. Fourth, perhaps this problem has occurred and the resultant error has affected important results. Note that there are, in a sense, two ways in which this problem could occur. Either (1) the sector attacker could change from Side 2 to Side 1, or (2) the sector attacker could change from wanting to attack and being allowed to do so by the code to wanting to hold but being forced to use attack parameters by the code. In the latter case, as long as the other side never became the sector attacker, the degree of the error from this problem would depend on how well a holding posture is modeled—and it is argued here that J-8's TACWAR models such a posture quite poorly (see, for example, Section B.1, above, and Reference [3]).

Several fixes are possible. One such fix can be obtained by changing the input data while keeping the current coding. For each input that applies when the side in question is on attack, there is a corresponding input that applies when the side in question is on defense. Thus, this fix would be to set the values of each attack-defense input pair equal to each other, thereby precluding the perception that the model always plays separate effectiveness parameters for attack and defense.

Several other fixes are possible, but a good code fix would also fix the "one more rifle turns the sector attacker into the sector defender" problem. One such fix, which would also accommodate holding postures, could be based on the picture below:



In this case, the inputs must satisfy

$$FRATA(KP,L) > FRATD(KP,L)$$

for all KP and L, and there would need to be a "hard" stop in the code if both

$$FRAD(\text{Blue-Attack/Red-Defense}) > FRATD(KP,1)$$

and

$$FRAD(\text{Red-Attack/Blue-Defense}) > FRATD(KP,2)$$

at the same time in the same sector. With this approach, the arrays ISAP(IS) and ICSA(IS) should be deleted, and the array ISA(IS) should be defined as follows for sector IS:

$$ISA(IS) = \begin{cases} -1 & \text{if the attacking side has not yet been determined} \\ 0 & \text{if a holding posture exists} \\ 1 & \text{if a mixture of holding and Side 1 attacking exists} \\ 2 & \text{if a mixture of holding and Side 2 attacking exists} \\ 3 & \text{if Side 1 is the attacker throughout the cycle} \\ 4 & \text{if Side 2 is the attacker throughout the cycle.} \end{cases}$$

Data for holding could be new inputs or, more simply and reasonably, could be the set equal to the corresponding data for defense, except for ERTD—holding postures would use ERTD(IW,6,L) in place of ERTD(IW,KP,L) no matter what KPS(IS) was.

This is one suggestion—there are several other relatively simple and straightforward methods to consider effectiveness parameters that are functions of attack and defense, that do not involve step functions, and that have relatively modest data requirements. J-8's TACWAR currently does not use such a method. Some thoughts concerning this observation are given in Reference [3]. Note that, while Reference [3] considers both the J-8 version and the IDA version of TACWAR, the comments above necessarily apply only to the J-8 version of TACWAR. These versions differ considerably concerning their treatment of sector attacker and posture. As a rough measure of their differences, the following table compares the number of times that they define and use variables and arrays to denote sector attacker and posture:

<u>J-8 Variable or Array Element</u>	<u>Also IDA ?</u>	<u>J-8 Type</u>	<u>J-8 Common Block</u>	<u>Number of Lines Where:</u>			
				<u>Value Is Set</u>		<u>Other Appearances</u>	
				<u>J-8</u>	<u>IDA</u>	<u>J-8</u>	<u>IDA</u>
Index for Sector Attacker:							
ISA(IS)	Yes	Input	BLANKRS	24	12	13	19
ISATT	No	Working	(local)	4	—	31	—
ISAP(IS)	No	Working	GCFM	<u>1</u>	<u>—</u>	<u>2</u>	<u>—</u>
			Totals	29	12	46	19
Kind of Posture:							
KP	Yes	Working	(local)	7	4	19	20
KPA	No	Working	(local)	2	—	9	—
KPCAS	No	Working	(local)	2	—	10	—
KPD	No	Working	(local)	2	—	11	—
KPI	No	Working	(local)	3	—	2	—
KPS(IS)	Yes	Working	BLANKRS	<u>5</u>	<u>4</u>	<u>8</u>	<u>4</u>
			Totals	21	4	59	24
Other Related Array:							
ICSA(IS)	No	Input	BLANK	3	—	4	—

Note that if ISA(IS) is defined as in the discussion here, it will no longer contain (embedded) information pertaining to issues unrelated to GC, which should be accounted for in separate arrays. For example, it would no longer contain information concerning whether Blue or Red was constrained by front-to-flank ratios, or whether Blue was attacking (after first defending) and had reached but not been allowed to cross the (time

zero) initial FEBA position, or the nuclear combat mode. Simplifying the definition of ISA might help eliminate the types of errors discussed here in that these errors may have resulted in part because the current definition of ISA(IS) is relatively complex.

14. Section 21 (Lines 3018-3040)

Lines 3018 through 3039 apply FRRINA and FRRINH to PWAKW and PWDKW.

Line 3040 ends the loop over side started on line 1942 at the beginning of Part C and, therefore, ends Part C.

D. PART D (LINES 3041-3249)

Part E, which follows, uses the ground-to-ground kill rate matrices, plus some results computed here, to calculate weapons system scores, force strengths, force ratios, and some related quantities. Before these can be calculated, air-to-ground and SSM kill rate matrices must be computed—and an additional calculation concerning the selection of a reference weapon may be needed. Part D calculates air-to-ground and SSM kill rate matrices, and it attempts to make an appropriate calculation concerning the selection of a reference weapon when annihilation occurs.

1. Section 22 (Lines 3041-3115)

The calculations made here concern the selection of the type of weapon designated to be the reference weapon if annihilation occurs. If no weapon types are ever annihilated, then this section has no effect on the results.

2. Section 23 (Lines 3116-3249)

This section just does for air-to-ground systems and SSMs what (code) Section 18 does for ground-to-ground systems—it multiplies engagement rates, allocations of fire, and probabilities of kill to give kill-rate matrices.

E. PART E (LINES 3250-3942)

Part E uses the kill rate matrices determined above in the antipotential potential method to compute weapons scores. A unit effectiveness factor for each unit is computed using these weapon scores and using the numbers of weapons, people, and supplies in the unit. These strengths are then used to compute which side is to be the "real" (not just

"likely") sector attacker, and to compute values for the arrays PCS(L) and VLS(L). These arrays can be used in Part F to compute personnel losses but not weapon losses, or weapon losses but not personnel losses, or both weapon losses and personnel losses, or they may not be used at all (anywhere in the model); their use depends on the values of the inputs MVCASO and IPLSC.

1. Section 24 (Lines 3250-3397)

This section uses Subroutines EIGENV and MPROD to calculate the antipotential potential scores for each type of ground weapon, aircraft sortie, and surface-to-surface missile system. If the three changes described next were made, the code here would be a clear and efficient implementation of the antipotential potential methodology described in References [11], [16], and [17]. The three changes that could be made here are as follows:

First, N3, N4, N5, N6, IADDR, BSUM(IADDR), and RSUM(IAADR) could be deleted entirely from the code, and NW(L), NAC(L), and NSSM(C) and the arrays P[W,A,M][A,D]KW(*,*,*) could be used directly instead. This will require a slight recoding of Subroutines EIGENV and MPROD.

Second, the test in Section 24 for the presence of targets for the reference weapon could be improved and moved to Section 22, and the rest of the current Section 24 could be deleted.

Third, if there are targets for the reference weapon type (either because there were targets for the original reference type or because the reference type has been changed to a type for which targets are present), then IFLAG will have a value of one in Section 24 only if serious errors (e.g., erroneous input data) exist. Accordingly, if Subroutine EIGENV returns with IFLAG = 1, a "hard" stop could be put here (after first writing appropriate reference information such as sector, cycle, and the cause of the stop).

2. Section 25 (Lines 3398-3605)

This section calculates scores for ground weapons and effectiveness measures for the divisions in combat.

3. Section 26 (Lines 3606-3640)

Given that the value (antipotential potential score) of a successfully penetrating CAS sortie by an aircraft of type IA in sector IS for side L on attack is VIAACF(IAC,IS,L) as computed in (code) Section 24, and that ACSABA(IAC,IS,L) of such sorties are flown, then the total value (score) of all of the CAS sorties for side L on attack in sector IS, VAABA(IS,L), is given by

$$VAABA(IS,L) = \sum_{IAC=1}^{NAC(L)} VIAACF(IAC,IS,L) * ACSABA(IAC,IS,L).$$

The model makes this computation (for aircraft and SSMs) in Section 26.

4. Section 27 (Lines 3641-3771)

This section uses the force strengths of the two sides, along with certain input parameters, to determine which side is to be the "true" attacker in the sector and cycle in question. A method for making this determination is discussed above in (text) Section C.13 (which concerns code Section 20).

5. Section 28 (Lines 3772-3803)

This section concerns reduced effectiveness due to supply shortages, and so has no direct effect on the example in Chapter III (which assumes that the possibility of a shortage of supplies is not to be considered).

6. Section 29 (Lines 3804-3891)

The goals of this section are to compute PCS(L), which is used later in the code to compute (personnel) casualties if IPLSC = 0, and to compute VLS(L), which is used later in the code to scale simulated losses if MVCASO = 0.

7. Section 30 (Lines 3892-3942)

This section only concerns the consumption of supplies and related inventories, and so is not directly relevant to the example presented in Chapter III.

F. PART F (LINES 3943-4168)

Parts F, G, and H are contained inside a do-loop over side—the first executable statement of Part F starts this loop and the last of Part H ends it. Part F attempts to

compute the number of weapons of each type lost in combat on each side, Part G makes some calculations concerning the consumption of munitions, and Part H makes computations concerning the number of personnel lost in combat on each side.

1. Section 31 (Lines 3943-3953)

Lines 3943 to 3945 are comment lines. Line 3946 starts the do-loop over side (side being denoted by L) and line 3947 sets $\Pi = 3 - L$. Lines 3948 to 3952 define some variables which could be deleted. Line 3953 could be moved to just after line 3957 in Section 32.

2. Section 32 (Lines 3954-4075)

This section is entered only if the input $MVCASO = 0, 2$, or 3 —if $MVCASO = 1$ the code skips to Section 33, below. If entered, this section can be pictured as making four somewhat intertwined calculations. First, it calculates simulated attrition using a heterogeneous Lanchester square formula in difference equation form. See References [2], [11], and [13] for discussions of why this is not a good formula to use to compute simulated attrition. Second, this section scales the simulated attrition using the term $VLS(L)$ computed in (code) Section 29, above. Determining weapons losses by dynamically scaling simulated attrition has several defects—these defects (and suggested alternatives to this approach) are described in References [2], [12], and [17]. Third, it artificially inflates the effects of close air support—see Reference [12] for details. Finally, it adjusts these scaled CAS-inflated Lanchesterian weapon losses if these losses, denoted at this point in the code by $WLS(IW,IS,L)$, are greater than the number of weapons in the sector, denoted by $WS(IW,L)$. This adjustment is erroneous, and lines 4056 through 4073 should be either fixed or replaced by a "hard" stop.

3. Section 33 (Lines 4076-4168)

This section is entered only if the input $MVCASO = 1$; if $MVCASO = 0, 2$, or 3 , the code (which would have just processed Section 32, above) skips to Section 34. If entered, this section can be pictured as making three of the four calculations (that would have been) made by Section 33. In particular, this section calculates simulated attrition using a heterogeneous Lanchester square formula in difference equation form, it inflates the effects of close air support, and it adjusts these CAS-inflated Lanchesterian weapon losses if these losses, again denoted by $WLS(IW,IS,L)$, are greater than the number of

weapons in the sector, again denoted by $WS(IW,L)$. The calculation made by Section 32 but not Section 33 is the scaling of the simulated attrition using $VLS(L)$. Accordingly, lines 4143 through 4166 should be either fixed or replaced by a "hard" stop.

G. PART G (LINES 4169-4239)

This part makes some calculations concerning munition consumption and it (optionally) writes some results to an output file.

1. Section 34 (Lines 4169-4210)

This section, which is entered for all values of $MVCASO$, scales ground munition expenditures by $VFACTOR$, which is the same scaling factor used to scale weapons losses if $MVCASO = 0$, and which is undefined if $MVCASO = 1$. Note that it does not scale the expenditures of air munitions or SSMs, and that no adjustments are made here if $VLS(IW,IS,L)$ is greater than $WS(IW,L)$.

2. Section 35 (Lines 4211-4239)

This section only writes some results to an output file—it performs no substantive calculations.

H. PART H (LINES 4240-4328)

1. Section 36 (Lines 4240-4309)

This section is entered only if the input $IPLSC = 0$; if $IPLSC = 1$, the code skips to Section 37, below. If entered, this section computes the number of casualties to unit ID (on side L in sector IS) as

$$PCS(L)*PDIV(ID)*FRACW,$$

where $PCS(L)$ is computed in (code) Section 29, $PDIV(ID)$ is the number of personnel in unit ID, and $FRACW$ is a term that depends on the input $DFBACT(L)$, on the FEBA position $FEBA(IS)$, and on $CDVLOC(ID)$, which is an input that, in general, is adjusted over time by the model. If $FRACW = 1.0$, then this method for computing casualties suffers from the weaknesses described in Reference [12]. If $0.0 \leq FRACW < 1.0$, then this method both suffers from these weakness and produces questionable results if any artillery, helicopters, and/or AAA are within $ERGMAX(IW,L)$ of the FEBA for corresponding values of IW . If $FRACW > 1.0$ or $FRACW < 0.0$, the method is incorrect.

2. Section 37 (Lines 4310-4328)

This section is entered only if the input $IPLSC = 1$; if $IPLSCO = 0$, the code (which would have just processed Section 36, above) skips to Section 38. If entered, this section, combined with Section 39 (lines 4359-4370), computes personnel losses to side L by summing (over all weapon types, IW) the product of the number of weapons of type IW times the input factor $FTCWPL(IW,L)$. The last line of this section ends the do-loop over side started at the beginning of Part F.

I. PART I (LINES 4329-4598)

Parts F, G, and H (combined with Section 34, if $IPLSC = 1$) will have calculated the number of weapons lost by type and number of people lost on each side in the sector and cycle in question. Part I assesses these losses to the units on each side and attempts to make several other miscellaneous calculations involving attrition before ending the huge do-loop over sector started at the beginning of Part B.

1. Section 38 (Lines 4329-4358)

As noted above, this section should be merged with Section 40 by moving Section 39 (which currently separates them) to Part H. Taken together, Sections 38 and 40 assess the personnel and weapon losses computed in Parts F and H to the units involved.

2. Section 39 (Lines 4359-4370)

This code is logically part of (code) Section 37.

3. Section 40 (Lines 4371-4426)

This section is logically part of Section 38 above and could be merged with that section if lines 4359 through 4370 are moved to Section 37.

4. Section 41 (Lines 4427-4501)

This section performs some calculations to determine the amounts of supplies destroyed by enemy fire in the combat being simulated. Given that supplies are being considered, the computations made here are quite reasonable. However, as currently coded, these computations are made whether or not supplies are being considered. Also, while the formulas used here are quite reasonable, the definition of the input

SLWCB(IW,ISPT,L) does not match its use in these formulas; the use is good, the definition is incorrect and should be changed—see Reference [12] for details.

5. Section 42 (Lines 4502-4565)

This section attempts to compute the numbers of weapons by type that are repairably damaged in the combat being simulated. It currently contains two errors (one coding error and one modeling error) that render its use questionable. It can simulate that weapons are either fully functional or fully destroyed by setting the input PWATS(IW,I,L) to 1.0 for all values of IW, I, and L. These errors are as follows.

The coding error is that WDIV(IW,ID) is treated on lines 4532 and 4534 as if it were the number of weapons of type IW in unit ID at the start of the combat being simulated. Due to line 4371, it isn't.

The modeling error is that line 4545 places no upper bound on the number of weapons that can be repaired at division level per cycle. It is wrong to require the assumption that division-level repair capabilities are essentially infinite, which the model currently does here.

6. Section 43 (Lines 4566-4598)

This section attempts to compute the numbers of casualties and weapon losses inflicted by air sorties in the sector without making use of previously computed numbers of casualties and weapon losses inflicted by aircraft. In particular, CDACS(IS,L) is correctly calculated here only if both $FICAS(2,3-L) = 1.0$ and $IPLSC = 0$, and $WLDAS(IW,IS,L)$ is essentially never correct (no matter what the values of $FICAS(2,3-L)$ and $MVCASO$, or any other inputs, are). The last line of this section ends the huge do-loop over sector started at the beginning of Part B, above.

J. PART J (LINES 4599-4783)

1. Section 44 (Lines 4599-4607)

This section computes the unweighted average of essentially arbitrary (for interdiction) scale factors discussed in (text) Section A.4, above.

3. Section 46 (Lines 4674-4702)

This section attempts to compute the average numbers of casualties and weapon losses inflicted by air sorties per sector in the region. Since CDACS is generally wrong and WLDAS is always wrong (see Section 43), so (respectively) are FPDCAS and FWDCAS.

4. Section 47 (Lines 4703-4723)

If these calculations are to be made here, then—

- (1) the factors AMLFD(IAC,IS,L) and AMNLAF(IAC,1,L) should be applied to AMNL(IAC,IAM,IMT) here as they are on line 3191.
- (2) munitions on aircraft that are killed in the air before those aircraft can attack ground targets (and any munitions that are jettisoned) could be considered somehow.

Alternatively, these calculations could be made in some routine called by Subroutine AC.

5. Section 48 (Lines 4724-4783)

Lines 4724, 4726, and 4728 are blank spacers; line 4725 resets IPRB; line 4727 is not needed; lines 4729 through 4782 are formats; and line 4783 ends GC.

III. A NUMERIC EXAMPLE FOR SUBROUTINE GC

Subroutine GC has no arguments—it receives all of its input data through common blocks. For the purposes here, "a subroutine input" is defined to be a variable or array in a common block that can be used without first being computed when code is processed according to some valid set of (model) input data.

For the example given below, all of the Subroutine GC input variables and arrays are grouped into the following three mutually exclusive and collectively exhaustive categories:

- (a) Inputs whose values are identically zero in this example.
- (b) "Bookkeeping," or extremely simplistic, inputs whose values are not identically zero here.
- (c) Non-zero inputs that are directly "relevant" for this example.

With one set of exceptions (described below), no mention is made in the following discussions of any input in category (a). The distinction made here between inputs in categories (b) and (c) is as follows. Inputs in category (b) are listed with their example values on Table III-1, but are not discussed further in this chapter. Conversely, inputs in category (c) are both listed with their example values on Table III-2 and listed, again with their data values, in each text section below that corresponds to a code section in which that input appears. Deciding which inputs to put in which of these two categories is somewhat judgmental; the basic idea used here is as follows. Some inputs, like the number of types of ground weapons on each side, are used throughout, and giving their values once is good enough. Indeed, repeating their values everywhere that they are used would unnecessarily clutter the example. On the other hand, inputs like probabilities of kill are used in many fewer places, and those places are (generally) organically relevant to those inputs. Accordingly, it is useful for understanding the code to identify each such input and its value in conjunction with each place where these inputs are used.

Table III-1. Values of the Non-Zero Bookkeeping and Miscellaneous Inputs Used In this Example

BPD (ID) = PDIV(ID)
 BWD(IW,ID) = WDIV(IW,ID)
 DFBACT(L) = 1.0
 FRRING(IT,KP,KT) = 1.0
 ICBIT(ID) = ID
 ICYCLE = 1
 IDLABA(IDS,1) = IDS
 IOMU = 5
 ISA(1) = 1
 ISAP(1) = 1
 ISEFFD = 3
 ITD(ID) = ID
 IWTIGT(IWTGT,1,L) = IWTGT
 IWTIWC(INWPC,1,L) = INWPC
 LNKWPT(IW,L) = IW
 KPS(1) = 1
 KTER(1) = 1
 NAC(L) = 1
 NAM(L) = 1
 NDS(1,L) = 1
 NEAEF(IC) = 2
 NEDEF(IC) = 2
 NS = 1
 NW(L) = 2
 NWAEFF(IC) = 2
 NWDEFF(IC) = 2
 NWTIGT(1,L) = 2
 NWTIWC(1,L) = 2
 TNGMOD(ID) = 1.0
 TWD(IW,1) = WDIV(IW,1)
 TWD(IW,2) = WDIV(IW,2)
 XAEF(2,IC) = 1.0
 XDEF(2,IC) = 1.0
 XWAEF(2,IC) = 1.0
 XWDEF(2,IC) = 1.0
 YAEF(2,IC) = 1.0
 YDEF(2,IC) = 1.0
 YWAEF(2,IC) = 1.0
 YWDEF(2,IC) = 1.0

Table III-2. Values of the Relevant Non-Zero Inputs Used In this Example

ACSABA(1,1,•)	=	10.		20.					
AMLFD(1,1,•)	=	1.		1.					
AMNL(1,1,2)	=	4.							
AMNL(1,1,8)	=	11.							
AMNLAF(1,1,•)	=	1.		1.					
BPKAW(1,•,1)	=	0.500		0.500					
BPKAW(2,•,1)	=	0.005		0.005					
BSAWA(1,•,1)	=	0.5		0.5					
BSAWA(2,•,1)	=	0.2		0.2					
EFCE	=	0.00001							
ERTA(•,1,1)	=	0.4		0.4					
ERTD(•,1,2)	=	0.4		11.0					
FCVLS(•)	=	3.059		2.228					
FICAS(2,•)	=	1.		1.					
FPCR(•)	=	0.4		0.4					
FTCWPL(•,1)	=	3.		1.					
FTCWPL(•,2)	=	2.		2.					
IPLSC	=	{either 0 or 1}							
IWUCE(1)	=	1							
MNIE	=	1000							
MVCASO	=	{either 0 or 1}							
NPCAF(1)	=	8							
NPCDF(2)	=	8							
PDIV(•)	=	3000.		800.					
PWATS(•,1,1)	=	1.		1.					
PWATS(•,3,2)	=	1.		1.					
PWSF(•,1)	=	10.0		1.0					
PWSF(•,2)	=	0.5		0.5					
RPKDW(1,•,1)	=	0.500		0.500					
RPKDW(2,•,1)	=	0.001		1.000					
RSAWD(1,•,1)	=	0.90909		0.09091					
RSAWD(2,•,1)	=	0.5		0.5					
SAMA(1,•,1)	=	0.5		0.5					
SAMD(1,•,2)	=	0.0		1.0					
SFCAS(1,1,1)	=	1.							
SFCAS(1,2,2)	=	1.							
SFGND(1,1,1)	=	1.							
SFGND(1,2,2)	=	1.							
WDIV(•,1)	=	100.		200.					
WDIV(•,2)	=	40.		40.					
VAMAW(1,•,1)	=	0.5		0.5					
VAMDW(1,•,2)	=	0.0		1.0					
VIWASF(•,1)	=	1.		1.					
XPCAF(•,1)	=	0.0	0.5	1.0	2.0	3.0	4.0	6.0	20.0
XPCDF(•,2)	=	0.0	0.5	1.0	2.0	3.0	4.0	6.0	20.0
YPCAF(1,•,1)	=	.100	.020	.015	.012	.010	.009	.008	.005
YPCDF(1,•,2)	=	.000	.010	.015.	.020	.022	.023	.024	.025

The one set of exceptions concerning the (zero-valued) inputs belonging to category (a) is as follows. The discussions below list the names of all "primary" inputs used by relevant sections of the code, and then list the values of the "relevant" inputs. All "relevant" inputs are considered to be "primary" inputs here; in addition, several potentially important (for full-up runs, if not for this example) zero-valued inputs are also considered to be "primary" and, thus, are also listed (by name only) in sections in which they appear. The overall goal here is to list the potentially important inputs concerning each section of the code, to list with their data values those inputs that are important for the particular example at hand, and to ignore (except in Table III-1) pure bookkeeping inputs. Note that the values of some of the inputs listed on Table III-1 are given in terms of arrays whose values are listed on Table III-2.

A. PART A (LINES 1-1762)

1. Section 1 (Lines 1-1613)

No calculations relevant to the numeric example given here are made in this section of the code.

2. Section 2 (Lines 1614-1665)

No calculation relevant to the numeric example given here are made in this section of the code.

3. Section 3 (Lines 1666-1740)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
PWSF(IW,L)	Model Input
BSAWA(IW,IWGT,KP)	Model Input
BSAWD(IW,IWGT,KP)	Model Input
RSAWA(IW,IWGT,KP)	Model Input
RSAWD(IW,IWGT,KP)	Model Input
SAMA(IAM,IGWT,L)	Model Input
SAMD(IAM,IGWT,L)	Model Input
SAMMA(IMM,IGWT,L)	Model Input
SAMMD(IMM,IGWT,L)	Model Input

<u>Section Outputs</u>	<u>Next Relevant Use</u>
PWSF(TW,L)	Section 11
BSAWA(TW,IWGT,KP)	Section 11
BSAWD(TW,IWGT,KP)	Section 11
RSAWA(TW,IWGT,KP)	Section 11
RSAWD(TW,IWGT,KP)	Section 11
SAMA(IAM,IGWT,L)	Section 11
SAMD(IAM,IGWT,L)	Section 11
SAMMA(IMM,IGWT,L)	N/A
SAMMD(IMM,IGWT,L)	N/A

Discussion

This section is entered only once during any given run of the model. In particular, it is entered only when GC is called for the first conventional combat cycle being simulated. When it is entered, it computes new (different) values for the elements of the arrays BSAWA, BSAWD, RSAWA, RSAWD, SAMA, SAMD, SAMMA, and SAMMD. Along with changing their values, this section changes the definition of these arrays from those given in Reference [4]. For example, the definition of BSAWA(TW,IWGT,KP) after Section 3 has been processed should be (assuming that each generic weapon type contains exactly one simulated weapon type) as follows.

BSAWA(TW,IWGT,KP): BSAWA(TW,IWGT,KP) is an input whose value is changed in Section 3 of GC. Its definition as an input is given in Reference [4]; its definition as a working array (i.e., its definition after its values have been changed by Section 3) is as follows. If there are $W(KW)$ Red weapons of type KW present for $KW = 1, 2, \dots, NW(2)$, then the probability that a Blue weapon of type TW will allocate its fire against a particular Red weapon of type IWGT when Blue is on attack in posture KP is given by

$$\frac{BSAWA(TW,IWGT,KP)}{WEIGHTED-SUM(TW,KP)}$$

where:

$$\begin{aligned} WEIGHTED-SUM(TW,KP) = \\ BSAWA(TW,1,KP)*W(1) + BSAWA(TW,2,KP)*W(2) \\ + \dots + BSAWA(TW,NW(2),KP)*W(NW(2)). \end{aligned}$$

Accordingly, the probability that a Blue weapon of type TW will allocate its fire against some Red weapon of type IWGT when Blue is on attack in posture KP is given by

$$\frac{\text{BSAWA}(\text{IW}, \text{IWGT}, \text{KP}) * \text{W}(\text{IWGT})}{\text{WEIGHTED-SUM}(\text{IW}, \text{KP})}$$

Thus, the model assumes that fire is allocated against target weapons according to the weighted number of target weapons of each type present, where the weighting factor for Red target weapons of type IWGT is given by BSAWA(IW,IWGT,KP) when the shooters are Blue weapons of type IW on attack in posture KP. In algebraic notation, before Section 3 is processed BSAWA corresponds to $\hat{A}_{ij}^s$ as defined on page III-8 of Reference [11]; afterwards it corresponds to C_{ij}^s as defined on page III-7 of that reference.

The other seven arrays whose values are changed here should be redefined analogously.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
PWSF(•,1)	10.0	1.0
PWSF(•,2)	0.5	0.5
BSAWA(1,•,1)	0.5	0.5
BSAWA(2,•,1)	0.2	0.2
RSAWD(1,•,1)	0.90909	0.09091
RSAWD(2,•,1)	0.5	0.5
SAWA(1,•,1)	0.5	0.5
SAWD(1,•,2)	0.0	1.0
<u>Relevant Section Outputs</u>	<u>Data</u>	
BSAWA(1,•,1)	0.5	0.5
BSAWA(2,•,1)	0.5	0.5
RSAWD(1,•,1)	0.5	0.5
RSAWD(2,•,1)	0.5	0.5
RSAWD(2,•,1)	0.0909	0.9091
SAWA(1,•,1)	0.5	0.5
SAWD(1,•,2)	0.0	1.0

Discussion

The value of BSAWA(IW,IWGT,KP) after Section 3 has been processed can also be interpreted as the probability that a Blue weapon of type IW would fire at the one Red

weapon of type IWGT present if there were exactly one Red weapon of each type present (and if Blue were on attack in posture KP), and the analogous comment applies to the other allocation arrays. Note that, while each entry of PWSF should be strictly positive, these entries need not satisfy any other requirement. Note also that each entry of the input allocation arrays should be nonnegative, but these entries need not satisfy any other requirement. Each entry of the output allocation arrays, however, will always be between 0.0 and 1.0 (inclusive), and will always sum to 1.0 when summed over all target types, IWGT, for any particular shooting system, posture (if applicable), and side.

4. Section 4 (Lines 1741-1762)

No calculations relevant to the numeric example given here are made in this section of the code.

B. PART B (LINES 1763-1941)

1. Section 5 (Lines 1763-1806)

No calculations relevant to the numeric example given here are made in this section of the code.

2. Section 6 (Lines 1807-1896)

a. Inputs and Outputs

Primary Section Input

WDIV(IW,ID)

Source

Model Input

Section Outputs

WS(IW,L)

WSI(IW,IS,L)

Next Relevant Use

Section 23

Section 12

Discussion

For each relevant value of IW, this section of the code sets WSI(IW,IS,L) equal to the sum of WDIV(IW,ID) over those ID that are currently indexing a unit in combat in sector IS for side L.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
WDIV(*,1)	100.	200.
WDIV(*,2)	40.	40.
<u>Relevant Section Outputs</u>	<u>Data</u>	
WS(*,1)	100.	200.
WS(*,2)	40.	40.
WSI(*,1,1)	100.	200.
WSI(*,1,2)	40.	40.

Discussion

Under the simplifying assumptions made here,

$$WS(IW,L) = WSI(IW,IS,L)$$

for all relevant IW, IS, and L. In general, this equality need not hold.

3. Section 7 (Lines 1897-1941)

This section initializes some arrays, but it makes no nontrivial calculations relevant to the numeric example given here.

C. PART C (LINES 1942-3040)

1. Section 8 (Lines 1942-1945)

This section initializes some arrays; but no nontrivial calculations relevant to the numeric example given here are made in this section of the code.

2. Section 9 (Lines 1946-1992)

No calculations relevant to the numeric example given here are made in this section of the code.

3. Section 10 (Lines 1993-2010)

No calculations relevant to the numeric example given here are made in this section of the code.

4. Section 11 (Lines 2011-2132)

a. Inputs and Outputs

Primary Section Inputs

Source

WSI(TW,IS,L)	Section 6
BSAWA(TW,IWGT,KP)	Section 3 for Cycle 1
BSAWD(TW,IWGT,KP)	Section 3 for Cycle 1
RSAWA(TW,IWGT,KP)	Section 3 for Cycle 1
RSAWD(TW,IWGT,KP)	Section 3 for Cycle 1
BFRASM(TW,IWGT,ISM)	Model Input
BFRDSM(TW,IWGT,ISM)	Model Input
RFRASM(TW,IWGT,ISM)	Model Input
RFRASM(TW,IWGT,ISM)	Model Input
SAMA(IAM,IGWT,L)	Section 3 for Cycle 1
SAMD(IAM,IGWT,L)	Section 3 for Cycle 1
SAMMA(IMM,IGWT,L)	Section 3 for Cycle 1
SAMMD(IMM,IGWT,L)	Section 3 for Cycle 1

Section Outputs

Next Relevant Use

SAWA(TW,KW,L)	Section 12
SAWD(TW,KW,L)	Section 12
BBFRASM(TW,KW,ISM)	N/A
BBFRDSM(TW,KW,ISM)	N/A
RRFRASM(TW,KW,ISM)	N/A
RRFRDSM(TW,KW,ISM)	N/A
SSAMA(IAM,KW,L)	Section 23
SSAMD(IAM,KW,L)	Section 23
SSAMMA(IMM,KW,L)	N/A
SSAMMD(IMM,KW,L)	N/A

Discussion

This section only concerns converting data entered by "generic" weapon type into data by "simulated" (or "actual") weapon type. A better approach might be to enter all data by "simulated" weapon type and to delete this section entirely.

b. Numeric Example

Relevant Section Inputs

	<u>Data</u>	
BSAWA(1,*,1)	0.5	0.5
BSAWA(2,*,1)	0.5	0.5
RSAWD(1,*,1)	0.5	0.5
RSAWD(2,*,1)	0.0909	0.09091
SAMA(1,*,1)	0.5	0.5
SAMA(1,*,2)	0.0	1.0
WSI(*,1,1)	100.	200.
WSI(*,1,2)	40.	40.

Relevant Section Outputs

	<u>Data</u>	
SAWA(1,*,1)	0.5	0.5
SAWA(2,*,1)	0.5	0.5
SAWD(1,*,2)	0.5	0.5
SAWD(2,*,2)	0.0909	0.9091
SSAMA(1,*,1)	0.5	0.5
SSAMD(1,*,2)	0.0	1.0

Discussion

Since in this example each generic weapon type corresponds to a simulated weapon type on a one-to-one basis, all this section does is to set the corresponding elements of various pairs of arrays equal to each other.

5. Section 12 (Lines 2133-2166)

a. Inputs and Outputs

Primary Section Inputs

MVCASO

SAWA(IW,KW,L)

SAWD(IW,KW,L)

WSI(IW,IS,L)

Source

Model Input

Section 11

Section 11

Section 6

Section OutputsNext Relevant Use

AAWA(IW,KW,L)

Section 16

AAWD(IW,KW,L)

Section 16

Discussion

The discussion in Section 3.a, above, states that TACWAR allocates fire according to the weighted numbers of targets present, where the weighting factors can depend on the type of shooter and type of target involved. The number of targets of type KW "present" for (i.e., vulnerable to) ground fire by side L is given by WSI(IW,K), where $K = 3 - L$. The weighting factors for, say, Blue weapons of type IW on attack in posture KP were calculated in Section 3, above, and the values of those factors were stored in BSAWA(IW,IWGT,KP) for all relevant values of IWGT. Given that each generic type of weapon corresponds to exactly one simulated type of weapon for each side, Section 11 sets

$$SAWA(IW,KW,L) = \begin{cases} BSAWA(IW,IWGT,KP) & L = 1 \\ RSAWA(IW,IWGT,KP) & L = 2 \end{cases}$$

and

$$SAWD(IW,KW,L) = \begin{cases} BSAWD(IW,IWGT,KP) & L = 1 \\ RSAWD(IW,IWGT,KP) & L = 2 \end{cases}$$

where IW is the simulated weapon type that (by the assumption above, uniquely) corresponds to the generic weapon type IWGT, and KP denotes the supposed posture in the sector in question. Thus, substituting SAWA for BSAWA and substituting WSI for W in the expression for the allocation of fire given in Section A.3.a, above, and doing the same for RSAWA and for defensive allocations of fire, yields

AAWA(IW,KW,L) =

$$\frac{SAWA(IW,KW,L) * WSI(KW,K)}{SAWA(IW,1,L) * WSI(1,K) + SAWA(IW,2,L) * WSI(2,K) + \dots + SAWA(IW,NW(K),L) * WSI(NW(K),K)}$$

and

AAWD(TW,KW,L) =

$$\frac{\text{SAWD(TW,KW,L)} * \text{WSI(KW,K)}}{\text{SAWD(TW,1,L)} * \text{WSI(1,K)} + \text{SAWD(TW,2,L)} * \text{WSI(2,K)} + \dots + \text{SAWD(TW,NW(K),L)} * \text{WSI(NW(K),K)}}$$

where AAWA(TW,KW,L) and AAWD(TW,KW,L) give the fraction (allocation) of the fire by weapons of type IW on side L that will be directed against enemy weapons of type KW for side L on attack and on defense, respectively, and where $K = 3 - L$. Section 12 implements these equations for AAWA and AAWD.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
SAWA(1,*,1)	0.5	0.5
SAWA(2,*,1)	0.5	0.5
SAWD(1,*,2)	0.5	0.5
SAWD(2,*,2)	0.0909	0.9091
WSI(*,1,1)	100.	200.
WSI(*,1,2)	40.	40.
<u>Relevant Section Outputs</u>	<u>Data</u>	
AAWA(1,*,1)	0.5	0.5
AAWA(2,*,1)	0.5	0.5
AAWD(1,*,2)	0.3333	0.6667
AAWD(2,*,2)	0.0476	0.9524

Discussion

The numbers of Red weapons present—40 and 40—are in the same proportion as the input PWSF(*,2)—0.5 and 0.5; thus, the simulated allocations of Blue fire are the same as the input allocations, when normalized to sum to 1.0. The weighting factors calculated in Section 11 say that Red weapons of type 1 are equally likely to shoot at Blue type 2 weapons as at Blue type 1 weapons when there are equal numbers of each type of Blue weapon present; that is, the methodology used here assumes that Red type 1 weapons are twice as likely to shoot at Blue type 2 weapons than at Blue type 1 weapons when there are twice as many Blue type 2 weapons present. Similarly, the weighting factors calculated in Section 11 say that Red weapons of type 2 are 10 times more likely

to shoot at Blue type 2 weapons than at Blue type 1 weapons when there are equal numbers of each type present. Hence, Red type 2 weapons are 20 times more likely to shoot at Blue type 2 weapons than at Blue type 1 weapons when there are twice as many Blue type 2 weapons present. Accordingly, AAWD here satisfies the properties that

$$AAWD(1,2,2) = 2 * AAWD(1,1,2)$$

and

$$AAWD(2,2,2) = 20 * AAWD(2,1,2).$$

Of course, these allocations must sum to 1.0 across target types; that is, AAWD must satisfy both

$$AAWD(1,1,2) + AAWD(1,2,2) = 1.0$$

and

$$AAWD(2,1,2) + AAWD(2,2,2) = 1.0 .$$

Note that the values for AAWD above are the unique solutions to these equations.

6. Section 13 (Lines 2167-2210)

a. Inputs and Outputs

Primary Section Inputs

ERTA(IW,KP,L)

ERTD(IW,KP,L)

Section Outputs

EERTA(IW,KP,L)

EERTD(IW,KP,L)

Source

Model Input

Model Input

Next Relevant Use

Section 17

Section 17

Discussion

As noted in Chapter II, above, either some meaningful calculations could be made here (such as applying the factors FRRINA and FRRINH here instead of in Section 21, and then deleting Section 21), or this section could be deleted, and EERTA and EERTD could be replaced by ERTA and ERTD, respectively, throughout the code.

b. Numeric Example

<u>Relevant Section Inputs</u>		<u>Data</u>
ERTA(*,1,1)	0.4	0.4
ERTD(*,1,2)	0.4	11.0
<u>Relevant Section Outputs</u>		<u>Data</u>
EERTA(*,1,1)	0.4	0.4
EERTD(*,1,2)	0.4	11.0

7. Section 14 (Lines 2211-2293)

This section concerns only preferred munitions and therefore contains no calculations relevant to the numeric example.

8. Section 15 (Lines 2294-2542)

No calculations whatsoever are made here.

9. Section 16 (Lines 2543-2577)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
BAWASM(IW,KW,ISM)	Section 14
BAWDSM(IW,KW,ISM)	Section 14
RAWASM(IW,KW,ISM)	Section 14
RAWDSM(IW,KW,ISM)	Section 14
AAWA(IW,KW,L)	Section 12
AAWD(IW,KW,L)	Section 12
<u>Section Outputs</u>	<u>Next Relevant Use</u>
AAWA(IW,KW,L)	Section 18
AAWD(IW,KW,L)	Section 18

Discussion

This section changes the definitions of AAWA and AAWD. Just before this section, AAWA(IW,KW,L) gives the fraction of the engagements by weapons of type IW

on side L that will be made against enemy weapon of type KW if side L is on attack. Accordingly, just before this section, AAWA(IW,KW,L) sums to 1.0 over KW for each relevant IW and L. Just after this section, AAWA(IW,KW,L) gives the fraction of the engagements by weapons of type IW on side L that will be made against enemy weapons of type KW using nonpreferred munitions if side L is on attack. Accordingly, just after this section, AAWA(IW,KW,L) sums over KW to the fraction of engagements by weapons of type IW on side L that will be made (against all enemy weapons) using non-preferred munitions if side L is on attack. Of course, the analogous comments apply to AAWD and side L on defense.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
AAWA(1,*,1)	0.5000	0.5000
AAWA(2,*,1)	0.5000	0.5000
AAWD(1,*,2)	0.3333	0.6667
AAWD(2,*,2)	0.0476	0.9524
<u>Relevant Section Outputs</u>	<u>Data</u>	
AAWA(1,*,1)	0.5000	0.5000
AAWA(2,*,1)	0.5000	0.5000
AAWD(1,*,2)	0.3333	0.6667
AAWD(2,*,2)	0.0476	0.9524

Discussion

Since, in this example, it is assumed that all weapons are using nonpreferred munitions in all of their engagements, this section does not change the values of AAWA and AAWD.

10. Section 17 (Lines 2578-2638)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
EERTA(IW,KP,L)	Section 13
EERTD(IW,KP,L)	Section 13

BPKASM(TW,KW,ISM)	Model Input
BPKDSM(TW,KW,ISM)	Model Input
RPKASM(TW,KW,ISM)	Model Input
RPKDSM(TW,KW,ISM)	Model Input
FPKASM(KP)	Model Input
FPKDSM(KP)	Model Input
BPKAW(TW,KW,KP)	Model Input
BPKDW(TW,KW,KP)	Model Input
RPKAW(TW,KW,KP)	Model Input
RPKDW(TW,KW,KP)	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
BVWASM(TW,KW,L)	N/A
BVWDSM(TW,KW,L)	N/A
RVWASM(TW,KW,L)	N/A
RVWDSM(TW,KW,L)	N/A
VIWAW(TW,KW,L)	Section 18
VIWDW(TW,KW,L)	Section 18

Discussion

J-8's TACWAR makes extensive use of the product of the number of engagements each shooter can make, on average, per time period (e.g., EERTA) times the allocation of those engagements to the various types of targets (e.g., AAWA) times the probability of kill by that type of shooter against that type of target in one of those engagements (i.e., BPKAW). This section computes only part of this product; e.g., it computes

$$\text{EERTA}(\text{TW},\text{KP},1) * \text{BPKAW}(\text{TW},\text{KW},\text{KP}).$$

Section 18, which follows, finishes the multiplication by computing, for example,

$$\text{AAWA}(\text{TW},\text{KW},1) * [\text{EERTA}(\text{TW},\text{KP},1) * \text{BPKAW}(\text{TW},\text{KW},\text{KP})].$$

The only role played by VIWAW and VIWDW is to pass this (partial) product from Section 17 to Section 18. In short, the relatively small arrays VIWAW and VIWDW, as well as the corresponding (for preferred munitions) quite large arrays BVWASM, BVWDSM, RVWASM, and RVWDSM, serve a very limited purpose.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
EERTA(*,1,1)	0.4	0.4
EERTD(*,1,2)	0.4	11.0
BPKAW(1,*,1)	0.500	0.500
BPKAW(2,*,1)	0.005	0.005
RPKDW(1,*,1)	0.500	0.500
RPKDW(2,*,1)	0.001	1.000
<u>Relevant Section Outputs</u>	<u>Data</u>	
VIWAW(1,*,1)	0.200	0.200
VIWAW(2,*,1)	0.002	0.002
VIWDW(1,*,2)	0.200	0.200
VIWDW(2,*,2)	0.011	11.000

11. Section 18 (Lines 2639-2697)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
BAWASM(IW,KW,L)	Section 14
BAWDISM(IW,KW,L)	Section 14
BVWASM(IW,KW,L)	Section 17
BVWDISM(IW,KW,L)	Section 17
RAWASM(IW,KW,L)	Section 14
RAWDISM(IW,KW,L)	Section 14
RVWASM(IW,KW,L)	Section 17
RVWDISM(IW,KW,L)	Section 17
AAWA(IW,KW,L)	Section 16
AAWD(IW,KW,L)	Section 16
VIWAW(IW,KW,L)	Section 17
VIWDW(IW,KW,L)	Section 17

Section Outputs**Next Relevant Use**

PWAKSM(TW,KW,ISM,L)

N/A

PWDKSM(TW,KW,ISM,L)

N/A

PWAKW(TW,KW,L)

Section 24

PWDKW(TW,KW,L)

Section 24

Discussion

After the loop from line 2670 through line 2675, PWAKW and PWDKW give weapon-on-weapon kill rates counting only those kills caused by nonpreferred munitions; however, after this section has been processed (i.e., after the loop ending on line 2697), these arrays give weapon-on-weapon kill rates counting all kills (i.e., caused by any type of munition, preferred or not). Of course, in the numeric example given here, the values in these arrays just after line 2697 are the same as the values just after line 2675 since no preferred weapons are being used in this example.

b. Numeric Example**Relevant Section Inputs****Data**

AAWA(1,*,1)	0.5000	0.5000
AAWA(2,*,1)	0.5000	0.5000
AAWD(1,*,2)	0.3333	0.6667
AAWD(2,*,2)	0.0476	0.9524
VIWAW(1,*,1)	0.2000	0.2000
VIWAW(2,*,1)	0.0020	0.0020
VIWDW(1,*,2)	0.2000	0.2000
VIWDW(2,*,2)	0.0110	11.0000

Relevant Section Outputs**Data**

PWAKW(1,*,1)	0.1000	0.1000
PWAKW(2,*,1)	0.0010	0.0010
PWDKW(1,*,2)	0.0667	0.1333
PWDKW(2,*,2)	0.0005	10.4762

Discussion

This section completes the multiplications started in Section 17.

12. Section 19 (Lines 2698-2911)

This section concerns writing outputs only; it makes no relevant calculations for the numeric example given here.

13. Section 20 (Lines 2912-3017)

a. Inputs and Outputs

Primary Section Inputs

EERTA(TW,KP,L)	Section 13
EERTD(TW,KP,L)	Section 13
AAWA(TW,KW,L)	Section 16
AAWD(TW,KW,L)	Section 16
BAWASM(TW,KW,ISM)	Section 14
BAWDISM(TW,KW,ISM)	Section 14
RAWASM(TW,KW,ISM)	Section 14
RAWDISM(TW,KW,ISM)	Section 14

Source

Section Outputs

EXNPMN(TW,KW,L)
TEXNPM(TW,IS,L)
EXPMN(TW,KW,ISM,IS,L)
SMENG(TW,ISM,IS,L)

Next Relevant Use

Section 34
Section 34
N/A
N/A

Discussion

The outputs of this section are scaled in Section 34, but these outputs are not used to calculate the values of any other variable or array in Subroutine GC (nor are they cumulative values).

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
EERTA(*,1,1)	0.4000	0.4000
EERTD(*,1,2)	0.4000	11.0000
AAWA(1,*,1)	0.5000	0.5000
AAWA(2,*,1)	0.5000	0.5000
AAWD(1,*,2)	0.3333	0.6667
AAWD(2,*,2)	0.0476	0.9524

<u>Relevant Section Outputs</u>	<u>Data</u>	
EXNPMN(1,*,1)	0.2000	0.2000
EXNPMN(2,*,1)	0.2000	0.2000
EXNPMN(1,*,2)	0.1333	0.2667
EXNPMN(2,*,2)	0.5238	10.4762
TEXNPM(*,1,1)	0.4000	0.4000
TEXNPM(*,1,2)	0.4000	11.0000

Discussion

It is no coincidence that TEXNPM equals EERTA for the attacker and EERTD for the defender. These equalities must hold whenever the weapons systems involved use only nonpreferred munitions.

14. Section 21 (Lines 3018-3040)

No calculations relevant to the numeric example given here are made in this section of the code.

D. PART D (LINES 3041-3249)

1. Section 22 (Lines 3041-3115)

a. Inputs and Outputs

Primary Section Inputs

MVCASO

IWUCE(IW)

WSI(IW,IS,1)

Source

Model Input

Model Input

Section 6

Section Outputs

IWUCEE

ISA(IS)

FRAD(IS)

KPS(IS)

IWUCES(IS)

Next Relevant Use

Section 24

N/A

N/A

N/A

Subroutine Output

b. Numeric Example

Relevant Section Inputs

MVCASO

IWUCE(1)

WSI(*,1,1)

Data

either 0 or 1

1

100. 200.

Relevant Section Outputs

IWUCEE

IWUCES(1)

Data

1

1

Discussion

According to the data in this example, this section of the code sets IWUCEE = IWUCE(1), and it does not change the values of ISA(1), FRAD(1), or KPS(1); accordingly, the code here has no effect on this example.

2. Section 23 (Lines 3116-3249)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
MVCASO	Model Input
SSAMA(IAM,KW,L)	Section 11
SAMMD(IAM,KW,L)	Section 11
SSAMMA(IMM,KW,L)	Section 11
SSAMMD(IMM,KW,L)	Section 11
WS(TW,L)	Section 6
AMLFD(IAC,IS,L)	Subroutine Input
AMNLAF(IAC,1,L)	Model Input
AMNL(IAC,IAM,2)	Model Input
AMNL(IAC,IAM,8)	Model Input
VAMAW(IAM,KW,L)	Model Input
VAMDW(IAM,KW,L)	Model Input
SSMNLI(ISSM,IMM,L)	Model Input
VMMAW(IMM,KW,L)	Model Input
VMMDW(IMM,KW,L)	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
PAAKW(IAC,KW,L)	Section 24
PADKW(IAC,KW,L)	Section 24
PMAKW(ISSM,KW,L)	N/A
PMDKW(ISSM,KW,L)	N/A

Discussion

This section essentially does for both aircraft and SSMs what Sections 12, 13, 14, 16, 17, 18, and 19 (combined) do for ground weapons.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
MVCASO	either 0 or 1	
SSAMA(1,•,1)	0.5	0.5
SSAMD(1,•,2)	0.0	1.0

WS(*,1)	100.	200.
WS(*,2)	40.	40.
AMLFD(1,1,*)	1.	1.
AMNLAF(1,1,*)	1.	1.
AMNL(1,1,2)		4.
AMNL(1,1,8)		11.
VAMAW(1,*,1)	0.5	0.5
VAMDW(1,*,2)	0.0	1.0
<u>Relevant Section Outputs</u>		<u>Data</u>
PAAKW(1,*,1)	1.0	1.0
PADKW(1,*,2)	0.0	11.0

Discussion

Given that the data for SSAMA(1,*,1) say that Blue aircraft would allocate their fire equally to the two types of Red ground weapons if equal numbers of those weapons were present, that the data for WS(*,2) say that equal numbers are present, and that the data for AMLFD(1,1,1), AMNLAF(1,1,1), and AMNL(1,1,2) say that each penetrating Blue aircraft can attempt 4 engagements, it follows that each Blue aircraft attempts 2 engagements against Red ground weapons of each type. Since the data for VAMAW(1,*,1) say that these aircraft have a probability of kill per engagement of 0.5 against either type of Red ground weapon, each Blue aircraft can (potentially) kill 1 ground weapon of each type per (successfully penetrating) sortie. Thus, these data yield that $PAAKW(1,KW,1) = 1.0$ for both $KW = 1$ and $KW = 2$.

Given that the data for SSAMD(1,*,2) say that Red aircraft allocate all of their fire against Blue ground weapons of type 2; that, based on these data, each penetrating Red aircraft can attempt 11 engagements; and that the data for VAMDW(1,2,2) say that these aircraft have a probability of kill per engagement of 1.0 against Blue weapons of type 2, $PADKW(1,1,2) = 0.0$ and $PADKW(1,2,2) = 11.0$.

E. PART E (LINES 3250-3942)

1. Section 24 (Lines 3250-3397)

a. Inputs and Outputs

Primary Section Inputs

PWAKW(TW,KW,L)

Source

Section 23

PWDKW(TW,KW,L)

Section 23

VIWASF(TW,1)

Subroutine Input

IWUCEE

Section 22

MNIE

Model Input

EFCE

Model Input

PAAKW(IAC,KW,L)

Section 23

PADKW(IAC,KW,L)

Section 23

PMAKW(ISSM,KW,L)

Section 23

PMDKW(ISSM,KW,L)

Section 23

Section Outputs

Next Relevant Use

VIWACF(TW,IS,L)

Section 25

VIWDCF(TW,IS,L)

Section 25

VIAACF(IAC,IS,L)

Section 26

VIADCF(IAC,IS,L)

Section 26

VIMACF(ISSM,IS,L)

N/A

VIMDCF(ISSM,IS,L)

N/A

Discussion

This section computes the antipotential potential scores for ground weapons (VIWACF and VIWDCF), for aircraft (VIAACF and VIADCF), and for SSMs (VIMACF and VIMDCF) based on the kill rate matrices PWAKW and PWDKW (for ground weapons versus ground weapons), PAAKW and PADKW (for aircraft firing at ground weapons), and PMAKW and PMDKW (for SSMs firing at ground weapons.) This section calls Subroutines EIGENV and MPROD to help make these calculations, and it is the only section in GC that calls EIGENV or MPROD.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
PWAKW(1,*,1)	0.1000	0.1000
PWAKW(2,*,1)	0.0010	0.0010
PWDKW(1,*,2)	0.0667	0.1333
PWDKW(2,*,2)	0.0005	10.4762
VIWASF(*,1)	1.	1.
IWUCEE	1	
MNIE	1000	
EFCE	0.00001	
PAAKW(1,*,1)	1.0	1.0
PADKW(1,*,2)	0.0	11.0
<u>Relevant Section Outputs</u>	<u>Data</u>	
VIWACF(*,1,1)	1.0000	0.0100
VIWDCF(*,1,2)	0.5166	0.7998
VIAACF(1,1,1)	10.000	
VIADCF(1,1,2)	0.836	

Discussion

Antipotential potential scores for ground weapons are eigenvectors of the product matrix of ground versus ground kill rate matrices (see Appendix A of Reference [11] for details). These eigenvectors are computed in Subroutine EIGENV using an iterative technique. VIWASF(TW,1), MNIE, and EFCE are used only to give a starting point for the iterative technique, to put an upper bound on the number of iterations attempted, and to stop the iteration when convergence is close enough, respectively. If the kill rate matrices are "well behaved," then this technique will converge very quickly (and so, in general, the value of MNIE is not important just as long as it is not too small, and a very small value can be used for EFCE). Also, if the kill rate matrices are "well behaved," this technique is guaranteed to converge to a unique (up to a scaling constant) eigenvector, so (except for running time) the values for VIWASF are irrelevant—any strictly positive vector would suffice (of course, the closer VIWASF is to the eigenvector to be found, the shorter the running time will be).

The point here is not to specify how Subroutine EIGENV works; instead, it is to observe what EIGENV produces. For VIWACF(*,1,1) to be an antipotential potential weapons score vector, it must be an eigenvector of the matrix $R_1(IW,JW)$, where

$$R_1(IW,JW) = \sum_{KW=1}^{NW(2)} PWAKW(IW,KW,1) * PWDKW(KW,JW,2).$$

In addition, it must satisfy two scaling properties, one of which is simply that $VIWACF(IWUCEE,1,1) = 1$; the other scaling property is discussed below.

To see that VIWACF is an eigenvector of R_1 , note that

$$VIWACF(*,1,1) = \begin{bmatrix} 1.0000 \\ 0.0100 \end{bmatrix},$$

$$[R_1(IW,JW)] = \begin{bmatrix} 0.006722 & 1.06095 \\ 0.000067 & 0.01061 \end{bmatrix},$$

and

$$\lambda_1 \begin{bmatrix} 1.0000 \\ 0.0100 \end{bmatrix} = \begin{bmatrix} 0.006722 & 1.06095 \\ 0.000067 & 0.01061 \end{bmatrix} \begin{bmatrix} 1.0000 \\ 0.0100 \end{bmatrix} = \begin{bmatrix} 0.01733 \\ 0.00017 \end{bmatrix}$$

for $\lambda_1 = 0.01733$. Analogously for Side 2, note that

$$R_2(IW,JW) = \sum_{KW=1}^{NW(1)} PWDKW(IW,KW,2) * PWAKW(KW,JW,1),$$

so

$$[R_2(IW,JW)] = \begin{bmatrix} .006800 & .006800 \\ .010531 & .010531 \end{bmatrix}$$

and

$$\lambda_2 \begin{bmatrix} 0.5166 \\ 0.7998 \end{bmatrix} = \begin{bmatrix} .006800 & .006800 \\ .010531 & .010531 \end{bmatrix} \begin{bmatrix} 0.5166 \\ 0.7998 \end{bmatrix} = \begin{bmatrix} 0.00895152 \\ 0.013863 \end{bmatrix},$$

where, as required by the theory of eigenvalues, $\lambda_2 = \lambda_1 = 0.01733$.

Let

$$\beta = \sqrt{\lambda_1} = \sqrt{\lambda_2} = 0.1316.$$

Then antipotential potential also requires that VIWDCF(*,1,2) be scaled so that

$$\beta * \text{VIWDCF}(\text{IW}, 1, 2) = \sum_{\text{KW}=1}^{\text{NW}(1)} \text{PWDKW}(\text{IW}, \text{KW}, 2) * \text{VIWACF}(\text{KW}, 1, 1)$$

for all IW from 1 through NW(2), and so (necessarily)

$$\beta * \text{VIWACF}(\text{IW}, 1, 1) = \sum_{\text{KW}=1}^{\text{NW}(2)} \text{PWAKW}(\text{IW}, \text{KW}, 1) * \text{VIWDCF}(\text{KW}, 1, 2)$$

for all IW from 1 through NW(1). To see that these equations hold here, note that

$$0.1316 \begin{bmatrix} 0.5166 \\ 0.7998 \end{bmatrix} = \begin{bmatrix} 0.0667 & 0.1333 \\ 0.0005 & 10.4762 \end{bmatrix} \begin{bmatrix} 1.0000 \\ 0.0010 \end{bmatrix}$$

and that

$$0.1316 \begin{bmatrix} 1.0000 \\ 0.0100 \end{bmatrix} = \begin{bmatrix} 0.1000 & 0.1000 \\ 0.0100 & 0.0100 \end{bmatrix} \begin{bmatrix} 0.5166 \\ 0.7998 \end{bmatrix}$$

to within a reasonable tolerance for rounding in the final decimal place. Note also that the last two equations give a numeric example for the basic antipotential potential assumptions: (1) that the scores for Red weapons,

Red type 1: 0.5166

Red type 2: 0.7998 ,

are proportional to the rate at which these weapons are killing Blue weapons,

Shooter	Target	
	Blue type 1	Blue type 2
Red type 1:	0.0667	0.1333
Red type 2:	0.0005	10.4762 ,

times the scores for the Blue weapons,

Blue type 1: 1.0000

Blue type 2: 0.0100 ;

and (2) that the scores for Blue weapons (given just above) are proportional to the rate at which these weapons are killing Red weapons,

Shooter	Target	
	Red type 1	Red type 2
Blue type 1:	0.1000	0.1000
Blue type 2:	0.0100	0.0100 ,

times the scores for the Red weapons (given above), where the constant of proportionality, 0.1316 here, must be the same for both sides.

Antipotential potential calculates the scores for aircraft using an analogous assumption with the same constant of proportionality. Accordingly, in the example here,

$$\text{VIAACF}(1,1,1) = \left(\frac{1}{\beta}\right) * [1.0 \ 1.0] \begin{bmatrix} 0.5166 \\ 0.7998 \end{bmatrix}$$

$$= \frac{1}{0.1316} * (1.3164) = 10.000 ,$$

and

$$\text{VIADCF}(1,1,2) = \left(\frac{1}{\beta}\right) * [0.0 \ 11.0] \begin{bmatrix} 1.000 \\ 0.010 \end{bmatrix}$$

$$= \left(\frac{1}{0.1316}\right) * (0.11) = 0.836 ,$$

which gives the scores for the Blue and Red aircraft, respectively.

2. Section 25 (Lines 3398-3605)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
BWD(IW,ID)	Subroutine Input
WDIV(IW,ID)	Model Input
TWD(IW,IT)	Model Input
VIWACF(IW,IS,L)	Section 24
VIWDCF(IW,IS,L)	Section 24
BPD(ID)	Subroutine Input
PDIV(ID)	Model Input
NEAEF(IC)	Model Input

NEDEF(IC)	Model Input
XAEF(IEAEF,IC)	Model Input
XDEF(IEDEF,IC)	Model Input
YAEF(IEAEF,IC)	Model Input
YDEF(IEDEF,IC)	Model Input
NWAEFF(IC)	Model Input
NWDEFF(IC)	Model Input
XWAEF(TWAEF,IC)	Model Input
XWDEF(TWDEF,IC)	Model Input
YWAEF(TWAEF,IC)	Model Input
YWDEF(TWDEF,IC)	Model Input
TNGMOD(ID)	Subroutine Input
FSEFED	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
EFFDA(ID)	Subroutine Output
EFFDD(ID)	Subroutine Output
VDAC(ID)	Subroutine Output
VDDC(ID)	Subroutine Output
VGABA(IS,L)	Section 29
VGDBA(IS,L)	Section 29
VGABAI(IS,L)	Section 27
VGDBAI(IS,L)	Section 27

Discussion

For each ground unit in combat, this section (re)calculates the numbers of weapons by type that are allowed on the front and then (using the weapon scores from Section 24) calculates both the total scores of all of the weapons on each side that are somehow involved in ground combat (VGABA and VGDBA) and the total scores of the portions of these weapons that are allowed on the front (VGABAI and VGDBAI). These scores can then be degraded as follows: nonlinearly to account for shortfalls (relative to fully authorized strengths) in the number of people and the number of equivalent weapons, by a weighted average of nonlinear degradations due to shortfalls of supplies (by type of supplies), and by an input factor to account for training.

In the relatively simple example here, all of the weapons involved in the ground combat calculations are allowed on the front (so VGABA = VGABAI and VGDBA =

VGDBAI), all units are at full strength in terms of numbers of weapons and people, supply shortages are not simulated, and the training factor is assumed to be 1.0. Thus, in this example, this section only computes VGABA, VGABAI, VGDBA, and VGDBAI as the sum over weapon types of the number of weapons present by type times the score per weapon for weapons of that type (on attack or defense, as appropriate). The code used here becomes more complex, and, arguably, contains some potential problems if these other factors (shortages in people, weapons, and supplies, and limited fronts) are considered (see Reference [12]).

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
PDIV(•)	3000.	800.
WDIV(•,1)	100.	200.
WDIV(•,2)	40.	40.
VIWACF(•,1,1)	1.0000	0.0100
VIWDCF(•,1,2)	0.5166	0.7998
<u>Relevant Section Outputs</u>	<u>Data</u>	
VGABA(1,1)	102.00	
VGABAI(1,1)	102.00	
VGDBA(1,2)	105.31	
VGDBAI(1,2)	105.31	

3. Section 26 (Lines 3606-3640)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
ACSABA(IAC,IS,L)	Subroutine Input
VIAACF(IAC,IS,L)	Section 24
VIADCF(IAC,IS,L)	Section 24
SSMALO(ISSM,IS,L)	Section 2
VIMACF(ISSM,IS,L)	Section 24
VIMDCF(ISSM,IS,L)	Section 24

Section OutputsNext Relevant Use

VAABA(IS,L)

Section 27

VADBA(IS,L)

Section 27

VMABA(IS,L)

N/A

VMDBA(IS,L)

N/A

Discussion

This section only states, for example, that if ACSABA(1,1,1) sorties of aircraft of type 1 on Side 1 penetrate Side 2's defenses far enough to deliver their ordnance on close air support, and if each such sortie has a score of VIAACF(1,1,1), then the total score for all of these sorties is the product ACSABA(1,1,1)*VIAACF(1,1,1).

b. Numeric ExampleRelevant Section InputsData

ACSABA(1,1,•)

10. 20.

VIAACF(1,1,1)

10.000

VIADCF(1,1,2)

0.836

Relevant Section OutputsData

VAABA(1,1)

100.00

VADBA(1,1)

16.72

4. Section 27 (Lines 3641-3771)**a. Inputs and Outputs**Primary Section InputsSource

VGABAI(IS,L)

Section 25

VGDBAI(IS,L)

Section 25

VAABA(IS,L)

Section 26

VADBA(IS,L)

Section 26

VMABA(IS,L)

Section 26

VMDBA(IS,L)

Section 26

FRATA(KP,ISATT)

Model Input

ISCEF(IS)

Subroutine Input

Section Outputs

Next Relevant Use

ICSA

N/A

ISA(1)

N/A

ISATT

N/A

FRAD(1)

Subroutine Output

Discussion

Due to the simplifying assumptions made here, Side 1 is necessarily both the "probable" and the "actual" sector attacker, so most of the calculations in this section, while potentially quite important, are not applicable in this example. Note, however, that FRAD is calculated here for use by Subroutine FEBA, which Program TACWAR calls after it calls GC.

b. Numeric Example

Relevant Section Inputs

Data

VGABAI(1,1)

102.00

VGDBAI(1,2)

105.31

VAABA(1,1)

100.00

VADBA(1,2)

16.72

Relevant Section Output

Data

FRAD(1)

1.655

Discussion

The total Side 1 (ground plus air) force strength is $102 + 100 = 202$, and the (ground plus air) Side 2 strength is $105.31 + 16.72 = 122.03$, which gives a force ratio of $202.00/122.03 = 1.655$

5. Section 28 (Lines 3772-3803)

This section concerns reductions in effectiveness due to simultaneous supply shortages, and so is not relevant to the numerical example here.

6. Section 29 (Lines 3804-3891)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
VGABAI(IS,L)	Section 25
VGDBAI(IS,L)	Section 25
VGABA(IS,L)	Section 25
VGDBA(IS,L)	Section 25
VAABA(IS,L)	Section 26
VADBA(IS,L)	Section 26
VMABA(IS,L)	Section 26
VMDBA(IS,L)	Section 26
FASFRC	Model Input
FDSFRC	Model Input
PPERE	Section 28
NPCAF(L)	Model Input
XPCAF(PCAF,L)	Model Input
YPCAF(KP,IPCAF,L)	Model Input
NPCDF(L)	Model Input
XPCDF(IPCDF,L)	Model Input
YPCDF(KP,IPCDF,L)	Model Input
FCVLS(L)	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
PCS(L)	Section 38
VLS(L)	Section 32

Discussion

This section is (in a logical sense) backwards. Logically, the code should first compute weapons lost by type based on an appropriate attrition structure, then compute total weapon value lost and associated personnel losses based on these weapons losses. This section does the opposite. It first attempts to compute total weapon value lost and (optionally) percent casualties, and then later computes weapons lost by type and (optionally) personnel losses based on this total value lost and this percent casualties. There are two errors in the code here. First, as mentioned above, the use of PPERE on lines 3867 and 3876 "double counts" the impact of supply shortages when both sides are

suffering such shortages. Since the example being discussed does not consider supply shortages, this error has no effect here. (If supplies are not being addressed, Sections 25 and 28 set PPESE = 1.) Second, as discussed in Reference [11], FASFRC should apply to VMABA as well as to VAABA on lines 3827 and 3828, and FDSFRC should apply to VMDBA as well as to VADBA on line 3829. Since the example being discussed does not consider SSMs, this error also has no effect here.

If MVCASO = 1, then VLS(L) is never used (code Section 32 is not processed), the values input for FCVLS(L) are ignored, and the values computed here for VLS(L) mean nothing. If MVCASO $\neq$ 1, then the values computed here for VLS(L) are used by GC as described in the discussion of (code) Section 32, below.

If IPLSC = 0 and MVCASO = 0, then PCS(L) has two simultaneous meanings. First, it gives the fraction (percent) of the number of personnel (as determined by PDIV(ID) for the relevant values of ID) that are lost (i.e., become casualties) on side L due to the combat (in the sector and cycle) being simulated. Second, it also gives a term which, when multiplied by the input FCVLS(L), is assumed to give the absolute amount of value lost on side L (where value is measured by antipotential potential scores) due to all losses of ground weapons in this combat. If IPLSC = 0 and MVCASO = 1, then PCS(L) has only the first meaning, not the second. If IPLSC = 1 and MVCASO = 0, then PCS(L) has only the second meaning, not the first—accordingly in this case, PCS(L) does not (except as a fluke) give the percent casualties in a sector. Finally, if IPLSC = 1 and MVCASO = 1, then, like VLS(L), PCS(L) is never used and the values computed here for PCS(L) mean nothing. Accordingly, if IPLSC = 1 and MVCASO = 1, all of the computations made in this section are useless, and the values of the inputs FASFRC, FDSFRC, NPCAF, NDCDF, XPCAF, XPCDF, YPCAF, and YPCAF, as well as FCVLS, have no effect on any of the results of GC.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>
VGABAI(1,1)	102.00
VGDBAI(1,2)	105.31
VGABA(1,1)	102.00
VGDBA(1,2)	105.31
VAABA(1,1)	100.00
VADBA(1,2)	16.72

PPESE					1.0			
NPCAF(1)					8			
XPCAF(*,1)	0.	.5	1.	2.	3.	4.	6.	20.
XPCAF(1,*,1)	.100	.020	.015	.012	.010	.009	.008	.005
NPCDF(2)					8			
XPCDF(*,2)	0.	.5	1.	2.	3.	4.	6.	20.
YPCDF(1,*,2)	.0	.010	.015	.020	.022	.023	.024	.025
FCVLS(*)				3.059		2.228		

Relevant Section Outputs

Data

PCS(*)	0.01664	0.01959
VLS(*)	5.192	4.596

Discussion

As discussed in Reference [12], 0.0 is the only logically correct value that can be assigned to FASFRC and FDSFRC concerning aircraft—and the code is simply wrong concerning SSMs. Rather than extend FASFRC and FRSFRC to apply to SSMs as well as aircraft, a better fix to the code here might be to eliminate those inputs entirely, and to replace lines 3827 and 3828 with

$$FRCA = VGABA(IS,L) / TEMP,$$

to delete line 3829, and to replace TEMP1 with VGABAI(IS,L) on lines 3830, 3831, 3833, and 3834.

Setting

$$FASFRC = FDSFRC = 0.0$$

yields

$$FRCA = 102.00/122.03 = 0.836$$

and

$$FRCD = 202.00/105.31 = 1.918.$$

Thus,

$$PCS(1) = (.020) \left(\frac{.164}{.500} \right) + (.015) \left(\frac{.336}{.500} \right) = 0.01664$$

and

$$PCS(2) = (.015) (.082) + (.020) (.918) = 0.01959.$$

Thus,

$$VLS(1) = (102.00) (0.01664) (3.059) = 5.192$$

and

$$VLS(2) = (105.31) (0.01959) (2.228) = 4.596 .$$

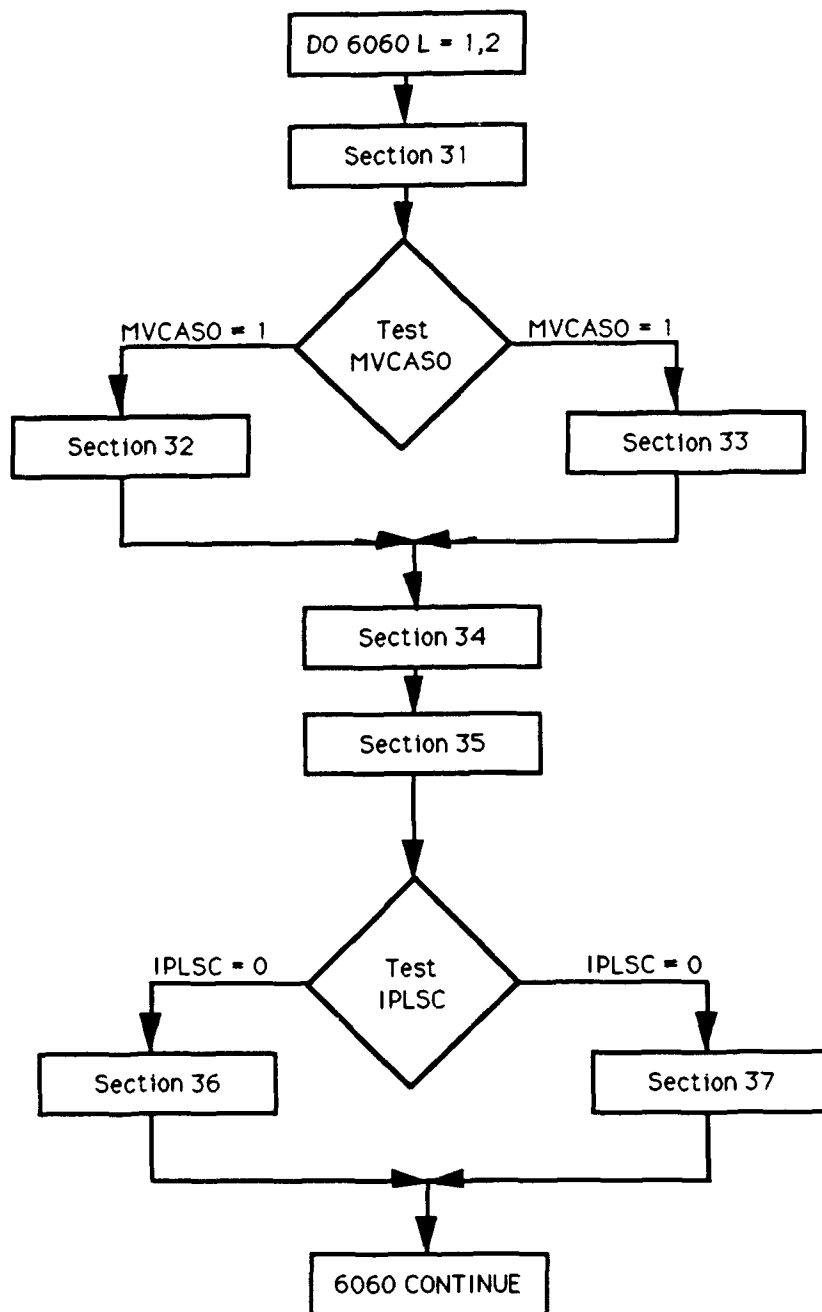
7. Section 30 (Lines 3892-3942)

This section concerns the consumption of supplies and resulting supply inventories; accordingly, it makes no calculations relevant to the numeric example presented here.

F. PART F (LINES 3943-4168)

1. Section 31 (Lines 3943-3953)

This section starts a relatively large do-loop (over side) which consists of Sections 31 through 37, inclusive (i.e., all of Parts F, G, and H). In any given cycle, only some of these sections will be processed—the choice depending on the values of the input variables MVCASO and IPLSC. The structure here is as shown in the following chart.



In the example here, all four cases will be considered, labeled as follows:

	IPLSC = 0	IPLSC = 1
MVCASO = 0	Case M0/0	Case M0/1
MVCASO = 1	Case M1/0	Case M1/1

The remainder of Section 31 just sets: (a) one meaningful working variable—II, (b) five meaningless working variables—N2, N3, N4, N5, and N6, and (c) one poorly named working variable—TEMP—that belongs in Section 32, not 31.

2. Section 32 (Lines 3954-4075)

As noted above, this section of the code is processed by GC only if the input MVCASO is not 1 (i.e., only if MVCASO is 0, 2, or 3).

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
MVCASO	Model Input
WS(IW, L)	Section 6
WSI(IW,IS,L)	Section 6
PWAKW(IW,KW,L)	Section 23
PWDKW(IW,KW,L)	Section 23
ACSABA(IAC,IS,L)	Subroutine Input
PAAKW(IAC,KW,L)	Section 23
PADKW(IAC,KW,L)	Section 23
SSMALO(ISSM,IS,L)	Subroutine Input
PMACKW(ISSM,KW,L)	Section 23
PMDKW(ISSM,KW,L)	Section 23
VIWACF(IW,IS,L)	Section 24
VIWDCF(IW,IS,L)	Section 24
VLS(L)	Section 29
SFAPP(L)	Section 4
NUMINC(L)	Section 4
FICAS(2,L)	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
SFAPP(L)	Section 42
NUMINC(L)	Section 42
VFACTOR	Section 34
GKGS(IW,KW,L)	Section 35
DGRD(IW)	Section 37
AKGS(IAC,KW,L)	Section 35

DAIR(TW)	Section 37
XMKGS(ISSM,KW,L)	N/A
DSSM(TW)	N/A
WLS(TW,IS,L)	Section 38

Discussion

Since (as noted above) the basic approach here is essentially backwards, the logic implementing this approach is necessarily somewhat convoluted. With one exception, this logic is discussed in conjunction with the numeric data below. The one exception concerns the input FICAS. This input is discussed in Reference [12] and, except for setting its value to 1.0, it will not be discussed further here.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
MVCASO	0 (for Cases M0/0 & M0/1)	
WS(•,1)	100.	200.
WS(•,2)	40.	40.
WSI(•,1,1)	100.	200.
WSI(•,1,2)	40.	40.
PWAKW(1,•,1)	0.1000	0.1000
PWAKW(2,•,1)	0.0010	0.0010
PWDKW(1,•,2)	0.0667	0.1333
PWDKW(2,•,2)	0.0005	10.4762
ACSABA(1,1,•)	10.	20.
PAAKW(1,•,1)	1.0000	1.0000
PADKW(1,•,2)	0.0000	11.0000
VTWACF(•,1,1)	1.0000	0.0100
VTWDCF(•,1,2)	0.5166	0.7998
VLS(•)	5.192	4.596
SFAPP(•)	0.	0.
NUMINC(•)	0	0
FICAS(2,•)	1.	1.

Relevant

Loss-Related Section Outputs	Data			
	Losses on Side L = 1		Losses on Side L = 2	
NUMINC(L)	1		1	
SFAPP(L)	0.5684		0.17284	
VFACTOR	0.5684		0.17284	
DGRD(•)	1.53	131.72	1.76	1.76
DAIR(•)	0	68.28	1.73	1.73
WLS(•,1,L)	1.53	200.00	3.49	3.49

Relevant KVS-Related Section Outputs	Data			
	Side L = 2 Killing Side 1		Side L = 1 Killing Side 2	
GKGS(1,•,L)	1.52	1.654	1.73	1.73
GKGS(2,•,L)	0.01	130.063	0.03	0.03
AKGS(1,•,L)	0	68.283	1.76	1.76

Discussion

First, Section 32 computes how many target weapons by type would be killed by each type of shooting weapon, assuming the following:

- (1) Each engagement made by each shooter is made against a target that both
 - (a) is not simultaneously being engaged by another shooter, and
 - (b) has not already been "killed" (before or) during the combat (in the cycle) being simulated.
- (2) All "bullets pass in mid-air" in that all of the weapons involved participate in all of their engagements (as determined by the inputs ERTA and ERTD) before any shooter or either side is killed.
- (3) The values input for these engagement rates and for the probabilities of kill per engagement are generally appropriate (i.e., are good enough for the analyses being made).

Given these assumptions and the data above, unscaled attrition to ground weapons in combat is calculated as indicated on Table III-3:

Table III-3. Unscaled Attrition Calculations

Shooting System Type	Number of Shooting Systems	Kill Rate Per Shooting System	Potential Kills	Target System Type	Target Side
Side 1 Shooting at Side 2					
Ground-1	100.	0.1000	10.0	Ground-1	2
		0.1000	10.0	Ground-2	2
Ground-2	200.	0.0010	0.2	Ground-1	2
		0.0010	0.2	Ground-2	2
Air-1	10.	1.0000	10.0	Ground-1	2
		1.0000	10.0	Ground-2	2
Side 2 Shooting at Side 1					
Ground-1	40.	0.0667	2.67	Ground-1	1
		0.1333	5.33	Ground-2	1
Ground-2	40.	0.0005	0.02	Ground-1	1
		10.4762	419.05	Ground-2	1
Air-1	20.	0.0000	0.00	Ground-1	1
		11.0000	220.00	Ground-2	1

These calculations yield the following "potential" killer-victim scoreboards and total attrition:

Shooting System Type on Side 1	Unscaled Kills On Side 2	
	Target System Type: Ground-1	Ground-2
Ground-1	10.0	10.0
Ground-2	0.2	0.2
Total Ground	10.2	10.2
Air-1	10.0	10.0
Grand Total	20.2	20.2

Shooting System Type on Side 2	Unscaled Kills On Side 1	
	Target System Type: Ground-1	Ground-2
Ground-1	2.67	5.33
Ground-2	0.02	419.05
Total Ground	2.69	424.38
Air-1	0.00	220.0
Grand Total	2.69	644.38

These results are, at best, "potential" because they are based on the three assumptions given above and, no matter how reasonable the data are (i.e., no matter how good

Assumption 3 is), Assumptions 1 and 2 above are questionable. Since Assumptions 1 and 2 each overstate attrition, the attrition results above are (if all else is acceptable) too high and should be reduced. In fact, the model does reduce this attrition, but it essentially does so by retaining Assumptions 1 and 2 while dropping Assumption (3).

To reduce this attrition, the model proceeds as follows. First, it computes the unscaled total value lost on each side as follows:

Target System Type	Antipotential Potential Value Per System	Unscaled Losses	Unscaled Value Lost
Targets on Side 1			
Ground-1	1.000	2.69	2.690
Ground-2	0.010	644.38	<u>6.444</u>
			9.134
Targets on Side 2			
Ground-1	0.5166	20.2	10.435
Ground-2	0.7998	20.2	<u>16.591</u>
			26.591

Second, the model assumes that the real total value lost on side L is given by VLS(L) as computed in Section 29. It then computes, for Side 1,

$$\begin{aligned} \text{VFACTOR} &= \text{VLS}(1) / \text{Total Unscaled Value Lost on Side 1} \\ &= 5.192/9.134 = 0.5684, \end{aligned}$$

and it computes, for Side 2,

$$\begin{aligned} \text{VFACTOR} &= \text{VLS}(2) / \text{Total Unscaled Value Lost on Side 2} \\ &= 4.596/26.591 = 0.17284. \end{aligned}$$

Next, the model assumes that the input intensity of the kill rates for Side 1 killing Side 2 is really too big by a factor of

$$1/0.17284 = 5.79,$$

and that the input intensity of the kill rates for Side 2 killing Side 1 is really too big by a factor of

$$1/0.5684 = 1.76.$$

Operating on this assumption, the model divides each of the unscaled kills computed above by these factors (i.e., scales these kills by VFACTOR) to obtain, with one

exception, what it assumes to be the actual losses for the combat being simulated. The one exception concerns checking to be sure not to "kill" more systems than there are present to be killed. Before considering this exception, however, the model scales the unscaled losses by the intensity factors just computed. In the example here, the model would now make the following calculations:

Shooting System Type on Side 1	Target System Type on Side 2					
	Ground-1			Ground-2		
	Unscaled Kills	Scale Factor	Tentative (Actual) Kills	Unscaled Kills	Scale Factor	Tentative (Actual) Kills
Ground-1	10.0	0.17284	1.73	10.0	0.17284	1.73
Ground-2	<u>0.2</u>	0.17284	<u>.03</u>	<u>0.2</u>	0.17284	<u>.03</u>
Total Ground	10.2		1.76	10.2		1.76
Air-1	<u>10.0</u>	0.17284	<u>1.73</u>	<u>10.0</u>	0.17284	<u>1.73</u>
Grand Total	20.2		3.49	20.2		3.49

Shooting System Type on Side 2	Target System Type on Side 1					
	Ground-1			Ground-2		
	Unscaled Kills	Scale Factor	Tentative (Actual) Kills	Unscaled Kills	Scale Factor	Tentative (Actual) Kills
Ground-1	2.67	0.5684	1.52	5.33	0.5684	3.03
Ground-2	<u>0.02</u>	0.5684	<u>.01</u>	<u>419.05</u>	0.5684	<u>238.19</u>
Total Ground	2.69		1.53	424.38		241.22
Air-1	<u>0</u>	0.5684	<u>0</u>	<u>220.00</u>	0.5684	<u>125.05</u>
Grand Total	2.69		1.53	644.38		366.27

The last set of calculations in Section 32 checks, weapon type by weapon type, whether the tentative number of kills just calculated is less than or equal to the number of weapons of that type present as given by WS(IW,L), not the number vulnerable to enemy air and ground weapons as given by WSI(IW,IS,L). This check is satisfied for three of the four type and side combinations in this example, so actual kills equal tentative kills for these types of ground weapons. This check fails for ground weapon type 2 on Side 1. Since, in this example, WS(2,1) = WSI(2,1,1), the code here scales the tentative losses of type 2 weapons on Side 1 by

$$\frac{\text{number present}}{\text{tentative number lost}} = \frac{200.00}{366.27} = 0.54605 ,$$

which gives:

Shooting System Type on Side 2	Target System Type 2 on Side 1		
	Tentative Kills	(Second) Scale Factor	Resulting Actual Kills
Ground-1	3.03	0.54605	1.654
Ground-2	<u>238.19</u>	0.54605	<u>130.063</u>
Total Ground	241.22		131.717
Air-1	<u>125.05</u>	0.54605	<u>68.283</u>
Grand Total	366.27		200.000

The table below summarizes the first (unscaled losses to tentative losses) and second (tentative losses to actual losses) scalings for both types of weapons on both sides:

		Unscaled Losses		Tentative Losses		Actual Losses	
Type of Weapon	Anti- potential Potential Value	Number Lost	Value Lost	Number Lost	Value Lost	Number Lost	Value Lost
Side 1							
1	1.00	2.69	2.690	1.529	1.529	1.529	1.529
2	0.01	644.38	<u>6.444</u>	366.27	<u>3.663</u>	200.000	<u>2.000</u>
Total	--	--	9.134	--	5.192	--	3.529
Side 2							
1	0.5166	20.2	10.435	3.491	1.804	3.491	1.804
2	0.7998	20.2	<u>16.156</u>	3.491	<u>2.792</u>	3.491	<u>2.792</u>
Total	--	--	26.591	--	4.596	--	4.596

This portion of the example points out another potential problem with the code. In general, it seems that one would not want to scale simulated losses unless one were relatively sure that one wanted to make the resulting value lost equal to that computed in Section 29. But this example shows that the model does not necessarily scale losses to produce this equality. In the example here, the results for Side 2 do have the property that the numbers of weapons lost (by type) times the values of these weapons sum (over type) to VLS(2) as computed by Section 29—but the results for Side 1 do not have this property. In particular, for Side 1, Section 29 gives that

$$VLS(1) = 5.192 ,$$

yet

$$\begin{aligned} WLS(1)*VIWACF(1,1,1) + WLS(2)*VIWACF(2,1,1) = \\ 1.529*1.00 + 200.000*0.01 = 3.529, \end{aligned}$$

a difference (based on weapons lost) of almost 50 percent. This difference is so big that it changes Side 1 from losing almost 13 percent more value than Side 2 to losing about 23 percent less value than Side 2.

Several fixes are possible here. One would be to put the "sanity" check that $WLS(IW,IS,L) \leq WS(IW,L)$ (including the corresponding adjustments if not) before the scaling to match $VLS(L)$ —i.e., to essentially copy lines 4051 through 4076 to just after line 3957, and to put a STOP statement in place of the current lines 4056 through 4074, inclusive. One of the problems with this fix concerns what to do if this STOP is encountered. In such a case a smaller time period could be used, but this could be quite tedious. Another fix would be to never set $MVCASO = 0$, and to use $MVCASO = 1$ instead—which is a step towards using the attrition logic in GC90.

3. Section 33 (Lines 4076-4168)

As noted above, this section of the code is processed only if the input $MVCASO$ is 1.

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
$MVCASO$ (implicitly)	Model Input
$WS(IW,L)$	Section 6
$WSI(IW,IS,L)$	Section 6
$PWAKW(IW,KW,L)$	Section 23
$PWDKW(IW,KW,L)$	Section 23
$SFGND(KP,I,L)$	Model Input
$ACSABA(IAC,IS,L)$	Subroutine Input
$PAAKW(IAC,KW,L)$	Section 23
$PADKW(IAC,KW,L)$	Section 23
$SFCAS(KP,I,L)$	Model Input
$SSMALO(ISSM,IS,L)$	Subroutine Input
$PMAKW(ISSM,KW,L)$	Section 23
$PMDKW(ISSM,KW,L)$	Section 23
$SFSSM(KP,I,L)$	Model Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
$GKGS(IW,KW,L)$	Section 35

DGRD(IW)	Section 37
AKGS(IAC,KW,L)	Section 35
DAIR(IW)	Section 37
XMKGS(ISSM,KW,L)	N/A
DSSM(IW)	N/A
WLS(IW,IS,L)	Section 38

Discussion

Since GC processes either Section 32 or Section 33, but not both, to calculate WLS(IW,IS,L) for all relevant IW, IS, and L, it is appropriate to compare them. Note that many, but not all, of the inputs and outputs are the same for these two sections. Concerning inputs, the differences are as follows. Unlike Section 32, Section 33 makes no use of the (section) inputs VIWACF(IW,IS,L), VIWDCF(IW,IS,L), VLS(L), SFAPP(L), NUMINC(L) and FICAS(2,L). Conversely, unlike Section 33, Section 32 makes no use of the inputs SFGND(KP,I,L), SFCAS(KP,I,L), and SFSSM(KP,I,L). Concerning outputs, Sections 32 and 33 each produce data on losses (GKGS, DGRD, AKGS, DAIR, XMKGS, DSSM, and WLS); however, Section 32 (but not 33) also calculates new values for SFAPP(L), NUMINC(L), and VFACTOR.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
MVCASO	1 (for Cases M1/0 and M1/1)	
WS(*,1)	100.	200.
WS(*,2)	40.	40.
WSI(*,1,1)	100.	200.
WSI(*,1,2)	40.	40.
PWAKW(1,*,1)	0.1000	0.1000
PWAKW(2,*,1)	0.0010	0.0010
PWDKW(1,*,2)	0.0667	0.1333
PWDKW(2,*,2)	0.0005	10.4762
ACSABA(1,1,*)	10.	20.
PAAKW(1,*,1)	1.0000	1.0000
PADKW(1,*,2)	0.0000	11.0000
SFGND(1,1,1)	1.	
SFGND(1,2,2)	1.	

SFCAS(1,1,1)

1.

SFCAS(1,2,2)

1.

Relevant
Loss-Related
Section Outputs

Data				
Losses on Side L = 1			Losses on Side L = 2	
DGRD(•)	2.69	131.717	10.2	10.2
DAIR(•)	0.	68.283	10.0	10.0
WLS(•,1,L)	2.69	200.000	20.2	20.2

Relevant
KVS-Related
Section Outputs

Data				
Side L = 2 Killing Side 1			Side L = 1 Killing Side 2	
GKGS(1,•,L)	2.67	1.654	10.0	10.0
GKGS(2,•,L)	0.02	130.063	0.2	0.2
AKGS(1,•,L)	0	68.283	10.0	10.0

Discussion

In this example, $FICAS(2,L) = 1$ for both L , $SFGND(1,I,L) = SFAIR(1,I,L) = 1$ for the relevant I and L , and $DGRD(IW) \leq WSI(IW,IS,L)$ whenever $WLS(IW,IS,L) \leq WS(IW,L)$. Therefore, the outputs of Section 33 (if MVCASO is 1) must necessarily equal the "unscaled" kills that would be computed by Section 32 if MVCASO were 0. Comparing the results here with the corresponding intermediate results of Section 32 shows that that is indeed the case in this example. Note that WLS is not scaled to match VLS here, and VLS is not recomputed to be consistent with WLS. Thus, if MVCASO = 1, the data reported for VLS by the model is (except by pure coincidence) inconsistent, and possibly quite inconsistent, with that reported by the model for WLS.

G. PART G (LINES 4169-4239)

1. Section 34 (Lines 4169-4210)

a. Inputs and Outputs

Primary Section Inputs

Source

EXNPMN(IW,KW,L)

Section 20

TEXNPM(IW,IS,L)

Section 20

EXPMN(IW,KW,ISM,IS,L)

Section 20

SMENG(TW,ISM,IS,L)

Section 20

SMSTK(ISM,IS,L)

Model Input

WSI(TW,IS,L)

Section 6

VFACTOR

Section 32

Section Outputs

Next Relevant Use

EXNPMN(TW,KW,L)

Subroutine Output

TEXNPM(TW,IS,L)

Subroutine Output

EXPMN(TW,KW,ISM,IS,L)

N/A

SMENG(TW,ISM,IS,L)

N/A

SMSTK(ISM,IS,L)

N/A

Discussion

The calculations here are trivial—Subroutine ADJUST can be described in five lines as follows:

```
SUBROUTINE ADJUST (Z,N,V)
  DIMENSION Z (N)
  DO 10 J = 1,N
    10  Z (J) = Z (J) * V
  END
```

b. Numeric Example

Relevant Section Inputs

Data

EXNPMN(1,•,1)	0.2000	0.2000
EXNPMN(2,•,1)	0.2000	0.2000
EXNPMN(1,•,2)	0.1333	0.2667
EXNPMN(2,•,2)	0.5238	10.4762
TEXNPM(•,1,1)	0.4000	0.4000
TEXNPM(•,1,2)	0.4000	11.0000
WSI(•,1,1)	100.	200.
WSI(•,1,2)	40.	40.

VFACTOR for {
Side 1, MVCASO = 0: 0.17284
Side 2, MVCASO = 0: 0.5684
Side 1, MVCASO = 1: Undefined
Side 2, MVCASO = 1: Undefined

Relevant Section Outputs	Data			
	MVCASO = 0		MVCASO = 1	
EXNPMN(1,•,1)	0.03457	0.03457	Undefined	Undefined
EXNPMN(2,•,1)	0.03457	0.03457	Undefined	Undefined
EXNPMN(1,•,2)	0.07577	0.15159	Undefined	Undefined
EXNPMN(2,•,2)	0.29773	5.95467	Undefined	Undefined
TEXNPM(•,1,1)	0.06914	0.06914	Undefined	Undefined
TEXNPM(•,1,2)	0.22736	6.2524	Undefined	Undefined

2. Section 35 (Lines 4211-4239)

a. Inputs and Outputs

Primary Section Inputs

GKGS(IW,KW,L)
SGKGS(IW,KW,L)
AKGS(IAC,KW,L)
SAKGS(IAC,KW,L)
XMKGS(ISSM,KW,L)
SMKGS(ISSM,KW,L)

Source

Section 32 or 33
Subroutine Input
Section 32 or 33
Subroutine Input
Section 32 or 33
Subroutine Input

Section Outputs

SGKGS(IW,KW,L)
SAKGS(IAC,KW,L)
SMKGS(ISSM,KW,L)

Next Relevant Use

Subroutine Output
Subroutine Output
Subroutine Output

Discussion

This section simply accumulates the per-cycle values of GKGS, AKGS, and XMKGS into the cumulative (over all cycles) arrays SGKGS, SAKGS, and SMKGS, respectively.

b. Numeric Example

Relevant Section Inputs	Data			
	MVCASO = 0		MVCASO = 1	
GKGS(1,•,1)	1.73	1.73	10.0	10.0
GKGS(2,•,1)	0.03	0.03	0.2	0.2
GKGS(1,•,2)	1.52	1.654	2.67	1.654

GKGS(2,•,2)	0.01	130.063	0.02	130.063
AKGS(1,•,1)	1.73	1.73	10.0	10.0
AKGS(1,•,2)	0.	68.283	0.	68.283

SGKGS(IW,KW,L) = SAKGS(IAC,KW,L) = 0.0 for all relevant arguments.

<u>Relevant Section Outputs</u>	<u>Data</u>
SGKGS(IW,KW,L)	GKGS(IW,KW,L) as above for all relevant IW, KW, and L
SAKGS(1,KW,L)	AKGS(1,KW,L) as above for all relevant KW and L

Discussion

Cumulative killer-victim scoreboards are identical to the per-cycle killer-victim scoreboards for the first cycle, which is the case in the example here.

H. PART H (LINES 4240-4328)

1. Section 36 (Lines 4240-4309)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
IPLSC	Model Input
PDIV(ID)	Model Input
PCS(L)	Section 29
VGABAI(IS,L)	Section 25
VGDBAI(IS,L)	Section 25
VAABA(IS,L)	Section 26
VADBA(IS,L)	Section 26
VMABA(IS,L)	Section 26
VMDBA(IS,L)	Section 26
SCASUAL(I,IS,L)	Subroutine Input
PLST(L)	Section 4
TPDIV(L)	Section 4
CPLS(IS,IC)	Subroutine Input
FPCR(IC)	Model Input
PIARP(IC)	Subroutine Input

<u>Section Outputs</u>	<u>Next Relevant Use</u>
TPDIV(L)	Model Output
SCASUAL(I,IS,L)	Model Output
PLS(IS,L)	Section 43
PLST(1)	Model Output
CPLS(IS,IC)	Model Output
PDIV(ID)	Model Output
PIARP(IC)	Model Output

Discussion

As noted above, this section is processed only if the input IPLSC is 0.

b. Numeric Example

<u>Relevant Section Inputs</u>	<u>Data</u>	
IPLSC	0 (for Cases M0/0 and M1/0)	
PDIV(•)	3000.	800.
PCS(•)	0.01665	0.01959
VGABAI(1,1)	102.00	
VGDBAI(1,2)		105.31
VAABA(1,1)	100.00	
VADBA(1,2)		16.72
SCASUAL(1,•,1)	0.	0.
SCASUAL(2,•,2)	0.	0.
CPLS(1,•)	0.	0.
FPCR(•)	0.4	0.4
PIARP(•)	0.	0.
<u>Relevant Section Outputs</u>	<u>Data</u>	
TPDIV(•)	3000.	800.
SCASUAL(1,•,1)	43.08	7.91
SCASUAL(2,•,1)	6.84	7.76
PLS(1,•)	49.92	15.67
PLST(•)	49.92	15.67
CPLS(1,•)	49.92	15.57
PDIV(•)	2950.08	784.33
PIARP(•)	1180.03	313.73

Discussion

To simplify making comparisons, the data used for PDIV here are based on the assumption that each type of weapon on each side requires 10 people in the unit to fire and to support the firing of that weapon. Note that, for IPLSC = 0, the resulting personnel losses are the same for MVCASO = 0 as for MVCASO = 1. Indeed, given PDIV and PLS, the number of personnel lost on each side is independent of the numbers of weapons lost when IPLSC = 0.

2. Section 37 (Lines 4310-4328)

a. Inputs and Outputs

Primary Section Inputs

IPLSC (implicitly)

DGRD(IW)

DAIR(IW)

DSSM(IW)

FTCWPL(IW,L)

SCASUAL(I,IS,L)

Section Output

SCASUAL(I,IS,L)

Source

Model Input

Section 32 or 33

Section 32 or 33

Section 32 or 33

Model Input

Subroutine Input

Next Relevant Use

Model Output

Discussion

As noted above, this section is processed only if the input IPLCS is not 0 (i.e., IPLSC = 1). In particular, if this section is processed, then the array SCASUAL still has its (subroutine) input values, not the values that would have been computed in Section 36 (since that section would not have been processed). Additional outputs concerning personnel losses when IPLSC = 1 are calculated in Section 39.

b. Numeric Example

Relevant Section Inputs and Outputs	Data							
	L = 1				L = 2			
	MVCASO = 0		MVCASO = 1		MVCASO = 0		MVCASO = 1	
IPLSC	1 (for Case M0/1)		1 (for Case M1/1)		1 (for Case M0/1)		1 (for Case M1/1)	
DGRD(•)	1.53	131.72	2.69	131.72	1.76	1.76	10.2	10.2
DAIR(•)	0.	68.28	0.	68.28	1.73	1.73	10.0	10.0
FICWPL(•)	3.	1.	3.	1.	2.	2.	2.	2.
SCASUAL(1,1,•)	136.31		139.79		3.52		40.8	
SCASUAL(2,1,•)	68.28		68.28		3.46		40.0	

Discussion

The underlying assumption for the data here is that three people are lost, on average, for each weapon of type 1 lost on Side 1, one person is lost for each weapon of type 2 lost on Side 1, and two people are lost for each weapon of either type lost on Side 2. Clearly, for IPLSC = 1, the resulting personnel losses depend directly on the numbers of weapons lost (and so depend on MVCASO as well as on other weapon-loss related inputs).

I. Part I (Lines 4329-4598)

1. Section 38 (Lines 4329-4358)

a. Inputs and Outputs

Primary Section Inputs

WS(IW,L)

WDIV(IW,ID)

WLS(IW,IS,L)

Section Output

TEMP 1

Source

Section 6

Model Input

Section 32 or 33

Next Relevant Use

Sections 39 and 40

Discussion

Lines 4329 through 4358 (Section 38) plus lines 4371 through 4426 (Section 40) form one logical section—the lines in between (Section 39, lines 4359-4370) are logically part of Section 37 as described above.

b. Numeric Example

<u>Relevant Section Inputs</u>		<u>Data</u>	
WS(•,1)	100.	200.	
WS(•,2)	40.	40.	
WDIV(•,1)	100.	200.	
WDIV(•,2)	40.	40.	
	<u>MVCASO = 0</u>		<u>MVCASO = 1</u>
WLS(•,1,1)	1.53	200.00	2.69 200.00
WLS(•,1,2)	3.49	3.49	20.20 20.20

<u>Relevant Section Outputs</u>		<u>Data</u>	
		<u>MVCASO = 0</u>	<u>MVCASO = 1</u>
TEMP1 (L = 1)	1.53	200.00	2.69 200.00
TEMP1 (L = 2)	3.49	3.49	20.20 20.20

Discussion

Since there is only one unit on each side in combat in this example, TEMP1 = WLS(IW,1,L) for all relevant IW and L.

2. Section 39 (Lines 4359-4370)

a. Inputs and Outputs

<u>Primary Section Inputs</u>	<u>Source</u>
IPLSC	Model Input
FTCWPL(IW,L)	Model Input
PDIV(ID)	Model Input
PLS(IS,IC)	Undefined
CPLS(IS,IC)	Subroutine Input
FPCR(IC)	Model Input
PIARP(IC)	Subroutine Input
<u>Section Outputs</u>	<u>Next Relevant Use</u>
PLS(IS,L)	Section 43
CPLS(IS,L)	Model Output
PDIV(ID)	Model Output
PIARP(IC)	Model Output

Discussion

This section is processed only if IPLSC = 1. As noted above, it is logically part of Section 37—it belongs there, not here.

b. Numeric Example

Relevant Section Inputs

Data

IPLSC	1 (for Cases M0/1 and M1/1)	
FTCWPL(•,1)	3.	1.
FTCWPL(•,2)	2.	2.
PDIV(•)	3000.	800.
PLS(1,•)	Undefined	
CPLS(1,•)	0.	0.
FPCR(•)	0.4	0.4
PIARP(•)	0.	0.

Relevant Section Outputs

	<u>Data</u>			
	<u>MVCASO = 0</u>		<u>MVCASO = 1</u>	
PLS(1,•)	Undefined		Undefined	
CPLS(1,•)	204.59	6.98	208.07	80.80
PDIV(•)	2795.41	793.02	2791.93	719.20
PIARP(•)	1118.16	317.21	1116.77	287.68

Discussion

If the code properly initialized PLS, in this example it would have the same values that CPLS has at the end of the section.

3. Section 40 (Lines 4371-4426)

a. Inputs and Outputs

Primary Section Inputs

Source

TEMP1	Section 38
WDIV(IW,ID)	Model Input
CWLS(IW,IS,IC)	Subroutine Input
WLST(IW,L)	Section 4
WLS(IW,IS,L)	Section 32 or 33

Section Outputs

WDIV(IW,ID)

CWLS(IW,IS,IC)

WLST(IW,L)

Next Relevant Use

Model Output

Model Output

Model Output

Discussion

The calculations here are very simple but quite important. The number of weapons of type IW in unit ID that were lost in the combat being simulated, given by TEMP1 here, is subtracted from the number of weapons of that type in that unit at the start of the combat, WDIV(IW,ID), so that at the end of this section WDIV(IW,ID) gives the number of weapons of type IW still operational in unit ID at the end of this combat. WLST(IW,L) totals this number over all units in all sectors and CWLS(IW,IS,IC) cumulates this number over all units belonging to country IC in the sector in question.

b. Numeric Example

Relevant Section Inputs	Data			
	MVCASO = 0		MVCASO = 1	
TEMP1 (L=1)	1.53	200.00	2.69	200.00
TEMP1 (L=2)	3.49	3.49	20.20	20.20
WDIV(*,1)	100.	200.	100.	200.
WDIV(*,2)	40.	40.	40.	40.
CWLS(*,1,1)	0	0	0	0
CWLS(*,1,2)	0	0	0	0
WLS(*,1)	1.53	200.00	2.69	200.000
WLS(*,2)	3.49	3.49	20.20	20.20
WLST(*,1)	0	0	0	0
WLST(*,2)	0	0	0	0
WDIV(*,1)	98.47	0.00	97.31	0.00
WDIV(*,2)	36.51	36.51	19.80	19.80
CWLS(*,1,1)	1.53	200.00	2.69	200.00
CWLS(*,1,2)	3.49	3.49	20.20	20.20
WLST(*,1)	1.53	200.00	2.69	200.00
WLST(*,2)	3.49	3.49	20.20	20.20

Discussion

Since there is only one unit on each side and only one sector, WLS, WLST, and CWLS all have the same values for the first cycle.

4. Section 41 (Lines 4427-4501)

This section (apparently) concerns the destruction of supplies, and so (apparently) no calculations relevant to this numeric example are made here.

5. Section 42 (Lines 4502-4565)

As discussed in Chapter II, the code incorrectly represents the repair of damaged weapons. Until these problems are fixed, the model could be used in conjunction with the assumption that each weapon in combat is either damaged so lightly (if at all) that it is (or can be made to be) fully operational by the start of the next time period or that the weapon is damaged so severely in that combat that it cannot be made operational no matter how long the war being simulated lasts. This assumption is easy to incorporate. All relevant probabilities of kill should include probabilities of severe damage but not light damage, and the input PWATS should be set so that

$$PWATS(IW,I,L) = 1.0$$

for all IW, I, and L.

6. Section 43 (Lines 4566-4598)

As discussed in Chapter II, except for degenerate cases, WLDAS(IW,IS,L) is never calculated correctly by this code, and CDACS(IS,L) is calculated correctly here only if both FICAS(1,3-L) is 1.0 and IPLSC is 0 (in which case CDACS(IS,L) equals TEMPA divided by SUM, where TEMPA is as used on line 4297—thus, CDACS should be set there rather than (re)calculated here). Until this section of the code is fixed, all of its calculations and outputs (i.e., the values of WLDAS and CDACS) should be ignored, as should any results based on these calculations and outputs (i.e., the values computed for FWDCAS(IW,IR) and FPDCAS(IR)).

J. PART J (LINES 4599-4783)

1. Section 44 (Lines 4599-4607)

If MVCASO = 0, this section computes SFAPP(L) as an unweighted (except trivially) average of the scale factors for each sector as computed in Section 32. Since there is only one sector in this example, SFAPP(1) = 0.5684 and SFAPP(2) = 0.17284 if MVCASO = 0. IF MVCASO = 1, this section sets SFAPP(•) to essentially meaningless values, and it gives users of the model no warning of this.

2. Section 45 (Lines 4608-4673)

This section concerns writing outputs only; it makes no relevant calculations for the numeric example given here.

3. Section 46 (Lines 4674-4702)

As noted in the discussion of code Section 43 (in text Section 6 of Part I, above), the values of CDACS and WLDAS (computed there) and the values of FPDACS and FWDCAS (computed here) are quite likely to be completely wrong and so should be ignored entirely.

4. Section 47 (Lines 4703-4723)

As discussed in Chapter II, the code in this section contains several potential problems (e.g., it does not consider AMLFD and AMNLAF) and, perhaps more importantly, it should not even be in GC. Since this section of the code is both erroneous and irrelevant to the calculation of attrition in GC, no numeric values are given for the example being presented.

5. Section 48 (Lines 4724-4783)

No calculations relevant to the numeric example given here are made in this (final) section of the code.

REFERENCES

- [1] Anderson, L. B. , *Data Dictionary for the GC90 Additions to TACWAR*, IDA Document D-767, Institute for Defense Analyses, Alexandria VA, September 1990.
- [2] Anderson, L. B. , *Annotated Briefing Slides for the GC90 Additions to TACWAR*, IDA Document D-794, Institute for Defense Analyses, Alexandria VA, September 1990.
- [3] Anderson, L. B. , *Three Issues Concerning the Use of Effectiveness Parameters That Depend on Who Is Attacking and Who Is Defending*, Working Paper LBA-1 of IDA Project D021-682, Institute for Defense Analyses, Alexandria VA, August 1991.
- [4] *TACTical WARfare Model (TACWAR) Input Data Dictionary for version 2.0*, Potomac Systems Engineering, September 1989, Draft.
- [5] *TACTical WARfare Model (TACWAR) Working Data Dictionary for version 2.0*, Potomac Systems Engineering, September 1989, Draft.
- [6] Anderson, L. B. , *Attrition Papers Referenced in IDAGAM I*, IDA Note N-846, Institute for Defense Analyses, Arlington VA, April 1979.
- [7] Karr, A. F., *Stochastic Attrition Models of Lanchester Type*, IDA Paper P-1030, Institute for Defense Analyses, Arlington VA, June 1974.
- [8] Karr, A. F., *A Class of Lanchester Attrition Processes*, IDA Paper P-1230, Institute for Defense Analyses, Arlington VA, December 1976.
- [9] Karr, A. F., *Lanchester Attrition Processes and Theater-Level Combat Models*, IDA Paper P-1528, Institute for Defense Analyses, Arlington VA, September 1981. This paper also appears as a chapter of: Shubik, M., et al., *Mathematics of Conflict*, North-Holland, Amsterdam, The Netherlands, 1983.
- [10] Schwartz, E. L., *A Short Proof of the Basic Binominal Heterogeneous Linear Attrition Equation, with indications for Extensions*, Working Paper WP-21 of IDA Project 2371, Institute for Defense Analyses, Alexandria VA, November 1983, revised June 1984.
- [11] Anderson, L. B. and F.A. Miercort, *COMBAT: A Computer Program to Investigate Aimed Fire Attrition Equations, Allocations of Fire, and the Calculation of Weapons Scores*, IDA Paper P-2248, Institute for Defense Analyses, Alexandria VA, September 1989.
- [12] Anderson, L. B. , *A Discussion of Several Potential Errors in the Current TACWAR Ground Combat Attrition Mechanisms*, Working Paper LBA-2 of IDA Project D021-682, Institute for Defense Analyses, Alexandria VA, August 1991.
- [13] Anderson, L. B. , *Heterogeneous Point Fire and Area Fire Attrition Processes that Explicitly Consider Various Types of Munitions and Levels of Coordination*, IDA Paper P-2249, Institute for Defense Analyses, Alexandria VA, October 1989.

- [14] Anderson, L. B. , *Heterogeneous Shoot-Look-Shoot Attrition Processes*, IDA Paper P-2250, Institute for Defense Analyses, Alexandria VA, October 1989.
- [15] Anderson, L. B. , et al., *The JCS Forces Planning Model, Part II: Effectiveness Module Documentation, Volume 3: Conventional Portion of the Theater Land-Air Module*, Volume 3 of Part II of IDA Report R-309, Institute for Defense Analyses, Alexandria VA, December 1989.
- [16] Anderson, L. B. , *Antipotential Potential*, IDA Note N-845, Institute for Defense Analyses, Arlington VA, April 1979.
- [17] Anderson, L. B. , *On Dynamically Recalculating Weapons Systems Scores and Force Ratios*, Working Paper LBA-3 of IDA Project D021-682, Institute for Defense Analyses, Alexandria VA, August 1991.

APPENDIX

A FORTRAN LISTING OF SUBROUTINE GC

SUBROUTINE GC

Change Number: 1

Date of Change: 29 Jan 1990

Subj of Change: Common Merge

Desc of Change: COMACHRS has been merged into COMACH.

Change Number: 2

Date of Change: 30 Jan 1990

Changed By: Annette Wilson (PSE) per recommendation of
Daniel Purcell (SYSCON)

Subj of Change: Casualty Calculations

Desc of Change: In order to enable the accurate calculation of
casualties, the casualty calculations were added
to the 6060 loop.

Change Number: 3

Date of Change: 1 Mar 1990

Changed By: Annette Wilson

Subj of Change: Precision Difference

Desc of Change: Due to precision differences between the VAX and
Sequent computers it was necessary to set weapon
losses to 0.0 when they fall below 0.0001.

— GC is the Ground Combat section of TACWAR. Included in GC:

SSM assignments
 Standard Allocation adjustment
 Ground Kill Rate Matrices
 Preferred and Non-Preferred Munitions
 Allocations of weapon type vs. weapon type
 Adjusted Allocation for Ground Weapons
 Weapon Engagement Rate Adjustment
 Kill Probability
 Kill Potential
 Munition Expenditures
 Air and Missile Kill Rate Matrices
 Ground, Air, and SSM values on attack and defense
 Sector Attacker computation
 Value Lost
 Supplies Consumed and new supply inventories
 Weapons lost by type
 Personnel Losses
 Weapons repairable after ground combat

C Casualties and weapon losses per sortie
C CAS munition expenditure

C/MODIFIED: Scott Kesling 06-14-88
C Potomac Systems Engineering
C 642-1000

C Modify the Munition Expenditure matrix to contain Adjusted
C values rather than maximum expenditures.

INCLUDE 'INCL\$DIR:PARAM.CMN'
INCLUDE 'INCL\$DIR:AGCOM.CMN'
INCLUDE 'INCL\$DIR:AGWK.CMN'
INCLUDE 'INCL\$DIR:BLANK.CMN'
INCLUDE 'INCL\$DIR:BLANKRS.CMN'
INCLUDE 'INCL\$DIR:BLANKSV.CMN'
INCLUDE 'INCL\$DIR:BLANKW.CMN'
INCLUDE 'INCL\$DIR:COMACH.CMN'

REAL VFACTOR

INCLUDE 'INCL\$DIR:GCFM.CMN'
INCLUDE 'INCL\$DIR:GCHIM.CMN'
INCLUDE 'INCL\$DIR:GCRS.CMN'
INCLUDE 'INCL\$DIR:GRDCOM.CMN'
INCLUDE 'INCL\$DIR:GRDRS.CMN'
INCLUDE 'INCL\$DIR:GRDSV.CMN'
INCLUDE 'INCL\$DIR:SUPCOM.CMN'
INCLUDE 'INCL\$DIR:TCCOM.CMN'

+++++
DIMENSION AAMA(LNAM,LNW,2)
DIMENSION AAWA(LNAM,LNW,2)
DIMENSION AAWD(LNW,LNW,2)
DIMENSION AKGS(LNAC,LNW,2)
DIMENSION CDACS(LNS,2)
DIMENSION DGRD(LNW)
DIMENSION DSSM(LNW)
DIMENSION EDDXR(LND)
DIMENSION GKGS(LNW,LNW,2)
DIMENSION PADKW(LNAC,LNW,2)
DIMENSION PCSAR(LNS,2)
DIMENSION PEDAR(LND)
DIMENSION PMAKW(LNSSM,LNW,2)
DIMENSION PPESAR(LNS,2)
DIMENSION PMAKW(LNW,LNW,2)
DIMENSION RSUM(LNADDR)
DIMENSION SEFAR(LND)
DIMENSION SMAALO(LNSSM,LNS,2)
DIMENSION TSC(LNSPT,2)
DIMENSION TWS(2)
DIMENSION TYPCDF(LNLNFC)
DIMENSION VDDC(LND)
DIMENSION VGBAI(LNS,2)
DIMENSION VIWAW(LNW,LNW,2)
AAMD(LNAM,LNW,2)
AAAWD(LNAM,LNW,2)
AAWD(LNW,LNW,2)
BSUM(LNADDR)
DAIR(LNW)
DSHAR(LND)
EDAXSR(LND)
FRCAR(LNS,2)
IHOLD(2)
PAAKW(LNAC,LNW,2)
PCS(2)
PEAAR(LND)
PLS(LNS,2)
PMDKW(LNSSM,LNW,2)
PPSAR(LND)
PMDKW(LNW,LNW,2)
SFAPRT(2)
SUMAW(LNW)
TSL(LNS,LNSPT,2)
TYPCAF(LNLNFC)
VDAC(LND)
VGABAI(LNS,2)
VLS(2)
VIWOW(LNW,LNW,2)

```

1577 DIMENSION WATSAR(LND)
1578 DIMENSION WDSAR(LND)
1579 DIMENSION WEDAR(LND)
1580 DIMENSION WS(LNW,2)
1581 DIMENSION WDDC(LND)
1582
1583 DIMENSION BAWASM(LNW,LNW,LNSMN), BAWDSM(LNW,LNW,LNSMN)
1584 DIMENSION BSUMA(LNW,LNW,2), BSUMD(LNW,LNW,2)
1585 DIMENSION BYWASM(LNW,LNW,LNSMN), BYWDSM(LNW,LNW,LNSMN)
1586 DIMENSION ISMFR(LNW,LNSMN,2), NSMFR(LNSMN,2)
1587 DIMENSION PWAKSM(LNW,LNW,LNSMN,2)
1588 DIMENSION PWOKSM(LNW,LNW,LNSMN,2)
1589 DIMENSION PWSUMA(LNW,LNW,2), PWSUMD(LNW,LNW,LNW,2)
1590 DIMENSION RAWASM(LNW,LNW,LNSMN), RAWDSM(LNW,LNW,LNSMN)
1591 DIMENSION RSUMA(LNW,LNW,2), RSUMD(LNW,LNW,2)
1592 DIMENSION RYWASM(LNW,LNW,LNSMN), RYWDSM(LNW,LNW,LNSMN)
1593 DIMENSION SAWA(LNW,LNW,2), SAWD(LNW,LNW,2)
1594 DIMENSION SSAMA(LNAM,LNW,2), SSAMD(LNAM,LNW,2)
1595 DIMENSION TWTGT(LNCWT,2), SSAMD(LNAM,LNW,2)
1596 DIMENSION BSHTRO(LNSMN)
1597 DIMENSION BSHTRA(LNSMN)
1598 DIMENSION BTRGT(LNSMN)
1599 DIMENSION BSTRAS(LNSMN)
1600 DIMENSION BSTRS(LNSMN)
1601 DIMENSION BSTRAS(LNSMN)
1602 DIMENSION BFRASM(LNW,LNW,LNSMN), BFRDSM(LNW,LNW,LNSMN)
1603 DIMENSION RFRASM(LNW,LNW,LNSMN), RFRDSM(LNW,LNW,LNSMN)
1604 DIMENSION EERTA(LNW,LNTP,2), EERTD(LNW,LNTP,2)
1605 DIMENSION AERTA(LNW,LNS,2), AERTD(LNW,LNS,2)
1606 DIMENSION RSTVA(LNW,LNS,2), RSTVD(LNW,LNS,2)
1607 DIMENSION WSGWA(LNW,2), WSGWD(LNW,2)
1608 DIMENSION WTCWA(LNW,2), WTCWD(LNW,2)
1609
1610 INCLUDE 'INCLDIR:TEST.CMN'
1611
1614
1615
1616
1617
1618
1619
1620 L = ISA(1S)
1621 IF (ICSA(1S).EQ.1.AND.L.NE.0) THEN
1622   IF (ICPDHD(L).EQ.1) THEN
1623     IF (L.EQ.1) THEN
1624       IF ((FEBA(1S).GT.FEBATZ(1S)).AND.(KPS(1S).EQ.3)) KPS(1S)=2
1625     ELSE
1626       IF ((FEBA(1S).LT.FEBATZ(1S)).AND.(KPS(1S).EQ.3)) KPS(1S)=2
1627     END IF
1628   END IF
1629 END IF
1630
1631
1632 IPRBST = IPRB
1633 IPRB = 0
1634 IF (KPSMGA(3).NE.1) GO TO 401
1635 WRITE(MOT,402) ICYCLE
1636 FORMAT(1H1,///,1X,'TABLE 5.3 CYCLE',14,/,10X,80(1H-),/).
402

```

C315: [CILLO, TACWAR30, IDA]GC, FOR: 9

4-Apr-1990 13:59:47

* 20X, 'DETAILED GROUND CALCULATION OF VALUES BY ANTI-'.
 * 'POTENTIAL POTENTIAL TO FOLLOW' .////)

IPRB = 1

401 CONTINUE

C

C 2) CALCULATE SSM'S ASSIGNED TO ATTACK DIVISIONS IN ACTIVE BATTLE AREA

C

DO 240 L = 1,2

ITAR = 10 + (LNSTMT * (L - 1))

DO 230 ISSM = 1, NSSM(L)

FRAC = (1. - FSKIF(ISSM, 1, L))

DO 220 IS = 1, NS

FRM = SSMPIM(ISSM, 9, L) * SSMCON(ISSM, IS, L)

IF (FRM.GT. SSMRDS(ISSM, L)) FRM = SSMRDS(ISSM, L)

SSMALO(ISSM, IS, L) = FRM * FRAC

SSMRDS(ISSM, L) = SSMRDS(ISSM, L) - FRM

CSSMST(ISSM, IS, ITAR) = CSSMST(ISSM, IS, ITAR) + FRM

CSSMAR(ISSM, IS, ITAR) = CSSMAR(ISSM, IS, ITAR)

+ SSMALO(ISSM, IS, L)

*

220 CONTINUE

230 CONTINUE

240 CONTINUE

IF (IPRB.EQ.1) THEN

CALL DEBUG(MOT, 'SSMCON', SSMCON, LNSSM, LNS, 2)

CALL DEBUG(MOT, 'SSMALO', SSMALO, LNSSM, LNS, 2)

CALL DEBUG(MOT, 'SSMRDS', SSMRDS, LNSSM, LNS, 2)

ENDIF

C

C 4) ADJUST THE STANDARD ALLOCATION.

C

IF (ICYCLE.EQ.1) THEN

DO 450 L = 1,2

N3 = NW(L)

K = 3 - L

N4 = NW(K)

N6 = NAM(L)

N5 = NAM(L)

IF (L.EQ.1) THEN

DO KWGT = 1, LNCWT

IF (PWSF(KWGT, K) .NE. 0.) THEN

DO KP = 1, LNPTP

DO IW = 1, N3

BSAWA(IW, KWGT, KP) = BSAWA(IW, KWGT, KP) /

PWSF(KWGT, K)

BSAWD(IW, KWGT, KP) = BSAWD(IW, KWGT, KP) /

PWSF(KWGT, K)

ENDDO

ENDDO

ELSE

DO KP = 1, LNPTP

DO IW = 1, N3

BSAWA(IW, KWGT, KP) = 0.0

BSAWD(IW, KWGT, KP) = 0.0

ENDDO

ENDDO

1637

1638

1639

1640

1641

1642

1643

1644

1645

1646

1647

1648

1649

1650

1651

1652

1653

1654

1655

1656

1657

1658

1659

1660

1661

1662

1663

1664

1665

1666

1667

1668

1669

1670

1671

1672

1673

1674

1675

1676

1677

1678

1679

1680

1681

1682

1683

1684

1685

1686

1687

1688

1689

1690

1691

1692

1693

```

1694      ENDIF
1695      ENDDO
1696      ELSE
1697          DO KWGT = 1, LNGWT
1698              IF (PWSF(KWGT, K) .NE. 0.) THEN
1699                  DO KP = 1, LNPTP
1700                      DO IW = 1, N3
1701                          RSAWA(IW, KWGT, KP) = RSAWA(IW, KWGT, KP) /
1702                              PWSF(KWGT, K)
1703                          RSAWD(IW, KWGT, KP) = RSAWD(IW, KWGT, KP) /
1704                              PWSF(KWGT, K)
1705                      ENDDO
1706                  ENDDO
1707              ELSE
1708                  DO KP = 1, LNPTP
1709                      DO IW = 1, N3
1710                          RSAWA(IW, KWGT, KP) = 0.0
1711                          RSAWD(IW, KWGT, KP) = 0.0
1712                      ENDDO
1713                  ENDDO
1714              ENDIF
1715          ENDDO
1716      ENDIF
1717  C
1718      DO KWGT = 1, LNGWT
1719          IF (PWSF(KWGT, K) .GT. 0.) THEN
1720              DO IAM = 1, N5
1721                  SAMA(IAM, KWGT, L) = SAMA(IAM, KWGT, L) / PWSF(KWGT, K)
1722                  SAMD(IAM, KWGT, L) = SAMD(IAM, KWGT, L) / PWSF(KWGT, K)
1723              ENDDO
1724              DO IAM = 1, N6
1725                  SAMMA(IAM, KWGT, L) = SAMMA(IAM, KWGT, L) / PWSF(KWGT, K)
1726                  SAMMD(IAM, KWGT, L) = SAMMD(IAM, KWGT, L) / PWSF(KWGT, K)
1727              ENDDO
1728          ELSE
1729              DO IAM = 1, N5
1730                  SAMA(IAM, KWGT, L) = 0.0
1731                  SAMD(IAM, KWGT, L) = 0.0
1732              ENDDO
1733              DO IAM = 1, N6
1734                  SAMMA(IAM, KWGT, L) = 0.0
1735                  SAMMD(IAM, KWGT, L) = 0.0
1736              ENDDO
1737          ENDIF
1738      ENDDO
1739      450 CONTINUE
1740      ENDIF
1741  C
1742      C INITIALIZATION OF 2.10 ARRAY VARIABLES
1743  C
1744      DO L = 1, 2
1745          TPDIV(L) = 0
1746          PLST(L) = 0
1747          N3 = NW(L)
1748          DO IW = 1, N3
1749              WLST(IW, L) = 0
1750              WST(IW, L) = 0

```

```

1751      ENDDO
1752      ENDDO
1753
1754      C 5) CHECK FOR COMBAT DEPLOYMENT CHANGES THIS CYCLE
1755      C
1756
1757      DO 505 L = 1,2
1758          SFAPPT(L) = SFAPP(L)
1759          IF (SFAPPT(L) .EQ. 0.0) SFAPPT(L) = 1.0
1760          NUMINC(L) = 0
1761          SFAPP(L) = 0.0
1762
1763      505 CONTINUE
1764      DO 9999 IS = 1,NS
1765          DO 500 L = 1,2
1766              I HOLD(L) = 0
1767              IWDRAW(IS,L) = 0
1768              IF (ICMST(IS,L) .NE. 0) THEN
1769                  IF (ISA(IS) .EQ. 0) THEN
1770                      ICHECK = ITA
1771                      ELSE
1772                          ICHECK = ISA(IS)
1773                      ENDIF
1774                      IF (ICHECK .EQ. 1) THEN
1775                          I HOLD(L) = 1
1776                      ELSE
1777                          IWDRAW(IS,L) = 1
1778                      ENDIF
1779                  ENDIF
1780          CONTINUE
1781          IF (IPRB .EQ. 1) THEN
1782              WRITE (MOT,1) ICYCLE,IS
1783              FORMAT('1',//10X,'GC CALCULATIONS FOR CYCLE',I4,X,
1784                  & 'SECTOR',I3,///)
1785              WRITE(MOT,131)
1786              WRITE(MOT,10) (I HOLD(L),L = 1,2),(IWDRAW(IS,L),L = 1,2)
1787          ENDIF
1788
1789      C 10) COMPUTE PERCENT RED GROUND KILLS BLUE, PERCENT BLUE GROUND KILLS
1790      C RED.
1791
1792      C GET THE TERRAIN TYPE IN THE SECTOR. THIS IS USED AS AN INDEX INTO THE
1793      C INTENSITY ARRAYS FRRING,FRRINA,FRRINH. THESE ARRAYS ARE USED TO ADJUST
1794      C THE NUMBER AND MIX OF GROUND WEAPONS DUE TO "FAIR", KTI=2, OR "POOR",
1795      C KTI=3 TERRAIN.....
1796
1797      KTI = KTER(IS)
1798
1799      C FIND THE SECTOR ATTACKER: FOR THE INITIAL OPENING THE SECTOR ATTACKER
1800      C IS SET BY DATA BASE INPUT TO 'ISA'. THIS IS USED TO GET THE POSTURE
1801      C INDEX INTO THE INTENSITY ARRAYS.
1802
1803      ISATT = ISAT(ISA(IS),IS)
1804      IF (ISATT .NE. ISAP(IS)) THEN 1 SECTOR ATTACKERS HAVE CHANGED
1805      IF (ICSA(IS) .NE. 1) KPS(IS) = 1
1806      ENDIF
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907

```

```

1808 C NOW GET THE POSTURE INDEX
1809 C
1810 KP = KPS(1S)
1811 N2 = 0
1812 DO 1020 L = 1,2
1813 KP1 = KP
1814 IF (L.EQ. ISATT) KP1 = 1      IL IS SECTOR ATTACKER
1815 N1 = 1 + N2
1816 N2 = N2 + NDS(1S,L)
1817 N3 = NW(L)
1818 DO 1W = 1,N3
1819   WS(1W,L) = 0.
1820   WSI(1W,1S,L) = 0.
1821 ENDDO
1822 IF (NDS(1S,L).GT. 0) THEN
1823   DO 1015 IDS = N1,N2
1824
1825 C ONLY COMPUTE FOR THE UNITS IN CONTACT
1826 C
1827   JDS = IDLABA(IDS,1S)
1828   IT = ITD(JDS)
1829
1830 C MAKE SURE NO HIGHER ECHELON UNITS ARE ON THE FEBA
1831 C
1832   IF (IT.EQ. -1) GOTO 1015
1833
1834 C COMPUTE HOW FAR THE UNIT IS FROM THE FEBA. FOR WEAPONS PHASE IN.
1835 C
1836   DIST = ABS(FEBA(1S) - CDVLOC(JDS))
1837   FRACW = (DFBACT(L) - DIST)/DFBACT(L)
1838
1839 C FRACW IS EQUAL TO THE FRACTION OF THE UNITS WEAPONS WHICH ARE IN
1840 C CONTACT WHEN THE UNIT IS A DISTANCE OF DIST FROM THE FEBA. THIS
1841 C FACTOR IS USED FOR DIRECT FIRE WEAPONS. WEAPONS SYSTEMS
1842 C SUCH AS HELOS AND ARTILLERY ARE TREATED AS AN ALL OR NOTHING
1843 C CALCULATION. WHEN THESE WEAPONS ARE WITHIN THEIR ENGAGEMENT RANGE
1844 C THEN ALL OF THEM ARE INCLUDED IN THE CALCULATION OTHERWISE NONE
1845 C OF THEM ARE INCLUDED. THIS ENGAGEMENT RANGE IS GIVEN BY THE VARIABLE
1846 C ERGMAX(1W,L).
1847 C
1848 C NOTE: WHEN ERGMAX(1W,L)=0 INDIRECT FIRE WEAPONS ARE TREATED LIKE
1849 C DIRECT FIRE WEAPONS WITH AN INCREASING FRACTION OF THE UNITS WEAPONS
1850 C COMING IN CONTACT AS THE UNIT MOVES CLOSER TO THE FEBA.
1851 C
1852 C COMPUTE THE NUMBER OF SIDE L WEAPONS IN THE SECTOR (WS). ALSO COMPUTE
1853 C THE NUMBER OF SIDE L WEAPONS IN THE SECTOR WHICH ARE TARGETS FOR GROUND
1854 C WEAPONS (WS1). FRRING IS THE REDUCTION FACTOR OF WEAPONS BASED ON TER-
1855 C RAIN, POSTURE AND UNIT TYPE. FRRING IS A PERCENTAGE OF THE TOE NUMBER
1856 C OF WEAPONS WHICH CAN FIT IN THE RESTRICTED TERRAIN. SO, TO GET THE NUM-
1857 C BER OF WEAPONS THAT ARE TARGETS YOU MUST TAKE THE MINIMUM OF THE NUMBER
1858 C OF WEAPONS ON-HAND AND THE NUMBER THAT CAN FIT.
1859 C
1860 C THE IWTWC ARRAY HOLDS THE WEAPON TYPES OF EACH CLASS OF WEAPON
1861 C THE ONLY CLASS OF WEAPONS AFFECTED BY THE RESTRICTIVE TERRAIN
1862 C IN TERMS OF THE NUMBER OF SHOOTERS AND TARGETS AVAILABLE ARE THE
1863 C GROUND-TO-GROUND WEAPON SYSTEMS WHICH IS WEAPON CLASS 1.
1864 C

```

```

1865 DO IWPCC = 1,NWTIWC(1,L)
1866   IW = IWTIWC(IWPCC,1,L)
1867   WS(IW,L) = WS(IW,L) + WDIV(IW,JDS) * FRACW
1868   WFIT = FRRING(IT,KPI,KTI) * TWD(IW,IT)  I#WPNs THAT FIT
1869   WSI(IW,IS,L) = WSI(IW,IS,L) +
1870     (AMIN1(WDIV(IW,JDS),WTFIT) * FRACW)
1871   ENDDO
1872
1873 C ADD IN REST OF WEAPONS TYPES (NOT AFFECTED BY TERRAIN)
1874 C THESE ARE WEAPON CLASSES:
1875 C
1876 C 2 => ARTILLERY
1877 C 3 => HELOS
1878 C 4 => ANTI-AIR
1879 C
1880 DO IWPCC = 2,IWPCC
1881   IF (NWTIWC(IWPCC,L) .GT. 0) THEN
1882     DO IWPCC = 1,NWTIWC(IWPCC,L)
1883       IW = IWTIWC(IWPCC,IWPCC,L)
1884       IF (ERGMX(IW,L) .EQ. 0.) THEN
1885         WS(IW,L) = WS(IW,L) + WDIV(IW,JDS) * FRACW
1886         WSI(IW,IS,L) = WSI(IW,IS,L) + WDIV(IW,JDS) * FRACW
1887       ELSE IF (DIST .LE. ERGMX(IW,L)) THEN
1888         WS(IW,L) = WS(IW,L) + WDIV(IW,JDS)
1889         WSI(IW,IS,L) = WSI(IW,IS,L) + WDIV(IW,JDS)
1890       ENDIF
1891     ENDDO
1892     IF (WEAPONS IN THIS CLASS
1893      .AND. IF WEAPONS IN THIS CLASS
1894      .AND. I CLASSES NOT DIRECTLY AFFECTED BY TERRAIN)
1895     CONTINUE
1896   ENDIF
1897 DO 1522 L = 1,2
1898   N3 = NW(L)
1899   TWS(L) = 0.
1900   DO 1521 IW = 1,N3
1901     TWS(L) = TWS(L) + WS(IW,L)
1902     WLS(IW,IS,L) = 0.
1903     WLDAS(IW,IS,L) = 0
1904     WST(IW,L) = WST(IW,L) + WS(IW,L)
1905   CONTINUE
1906   PLS(IS,L) = 0.
1907   CDACS(IS,L) = 0
1908 CONTINUE
1909 FRAD(IS) = 999.
1910 DO 1523 L = 1,2
1911   VGABA(IS,L) = 0.
1912   VGDBA(IS,L) = 0.
1913   VGABAI(IS,L) = 0.
1914   VGDBAI(IS,L) = 0.
1915   VAABA(IS,L) = 0.
1916   VADBA(IS,L) = 0.
1917 CONTINUE
1918 IF (TWS(1) .LE. 0.0.AND. TWS(2) .LE. 0.0) GO TO 9999
1919 DO 1524 L = 1,2
1920   IF (TWS(L) .NE. 0) GO TO 1524
1921   K = 3 - L

```

```

1822      ISA(IS) = K
1823      IF (KSTOPB.EQ.1.AND.ISA(IS).EQ.1.AND.FEBA(IS).LE.FEBATZ(IS))
1824      1      ISA(IS) = -3
1825      VGABA(IS,K) = 1.0
1826      VAABA(IS,K) = 1.0
1827      VGABAI(IS,K) = 1.0
1828      VGOBAI(IS,K) = 1.0
1829      1524 CONTINUE
1830      IF (TWS(1).LE.0.0.OR.TWS(2).LE.0.0) GO TO 9999
1831
1832      C+++++
1833      C      MVCASO = 0 USE APP RECALCULATE ALLOCATIONS EVERY
1834      C      CYCLE
1835      C      1 USE SCALING FACTORS (DONT USE APP)
1836      C      2 USE APP, CALCULATE ALLOCATIONS FIRST
1837      C      CYCLE ONLY
1838      C      3 USE APP WITH STANDARD ALLOCATIONS
1839      C+++++
1840
1841      IF (MVCASO.EQ.2.AND.ICYCLE.GT.1) GOTO 2500
1842      DO 1050 L = 1,2
1843      11 = 3 - L
1844      N3 = NW(L)
1845      N4 = NW(11)
1846
1847      C      COMPUTE SMART/PREFERRED MUNITIONS FIRING REQUIREMENTS
1848      C
1849      DO IW = 1,N3
1850      DO ISM = 1,NSM(11)
1851      ISMFR(ISM,IW,11) = 0
1852      ENDDO
1853      ENDDO
1854      IF (ISMUS(11).NE.0) THEN
1855      DO ISM = 1,NSM(11)
1856      NSMFR(ISM,11) = 0
1857      IF (SMSTK(ISM,IS,11).GT.0.0) THEN
1858      IF (NSMT(ISM,11).GT.0) THEN
1859      DO INSMT=1,NSMT(ISM,11)
1860      IGWT = ISMT(INSMT,ISM,11)
1861      DO INWTIGT = 1,NWTIGT(IGWT,L)
1862      IW = IWTIGT(INWTIGT,IGWT,L)
1863      IF (WSI(IW,IS,L).GT.0.0) THEN
1864      NSMFR(ISM,11) = NSMFR(ISM,11) + 1
1865      ISMFR(NSMFR(ISM,11),ISM,11) = IW
1866      ENDIF 1 IF THERE ARE TARGETS PRESENT
1867      ENDDO
1868      ENDDO 1 POSSIBLE TARGETS FOR THIS PREFERRED MUNITION
1869      ENDDIF 1 IF TARGETS FOR THIS PREFERRED MUNITION
1870      ENDDIF 1 IF THERE ARE PREFERRED MUNITION AVAILABLE
1871      ENDDO 1 PREFERRED MUNITION
1872      ENDDIF
1873      IF (KPSMGA(15).EQ.1.AND.ISMUS(11).EQ.1) THEN
1874      WRITE (NOT,1009)
1875      FORMAT(1X,'WEAPON TYPES WHICH WILL FIRE PREFERRED ',
1876      'MUNITIONS')
1877      DO ISM = 1,NSM(11)
1878      IF (NSMFR(ISM,11).GT.0) THEN

```

```

1979      DO INSMW=1,NSMW(ISM,II)
1980      IW = ISMW(INSMW,ISM,II)
1981      WRITE (MOT,1008)ISM,
1982      (AWT(IW,IWORD,II),IWORD=1,2),WSI(IW,IS,II)
1983      ENDDO
1984      ENDIF
1985      ENDDO
1986      FORMAT (1X,I3,1X,2A4,1X,F10.3)
1987      WRITE (MOT,1007) ICYCLE,II
1988      FORMAT (/, ' MUNITIONS AVAILABLE CYCLE',I4,' SIDE',I4)
1989      DO 1005 ISM = 1,NSM(II)
1990      WRITE (MOT,1006) SMSTK(ISM,IS,II)
1991      FORMAT (1X,F10.3)
1992      ENDIF
1993      C+-----+
1994      C+
1995      C+ IF SIDE II IS THE SECTOR ATTACKER, THEN HE IS ATTACKING A +
1996      C+ DEFENDER IN POSTURE KPS. WHEN HE GOES ON THE DEFENSE HE +
1997      C+ WILL BE DEFENDING IN POSTURE 1 (DELAY). +
1998      C+
1999      C+ IF SIDE II IS THE DEFENDER, THE HE IS DEFENDING IN +
2000      C+ POSTURE KPS. WHEN HE GOES ON THE ATTACK, HE WILL BE +
2001      C+ ATTACKING A DEFENDER IN POSTURE 1 (DELAY). +
2002      C+
2003      C+-----+
2004      IF (II.EQ. ISATT) THEN 1 SIDE II IS THE ATTACKER
2005      KPD = 2
2006      KPA = KPS(IS)
2007      ELSE
2008      KPA = 2
2009      KPD = KPS(IS)
2010      ENDIF
2011      C+-----+
2012      C+
2013      C+ BREAK OUT ALLOCATIONS BY GENERIC WEAPON TYPE TO BE BY WEAPON +
2014      C+ TYPE. +
2015      C+
2016      C+-----+
2017      C+
2018      C+ COMPUTE THE TOTAL NUMBER OF WEAPONS IN EACH GENERIC WEAPON +
2019      C+ CATEGORY. +
2020      C+
2021      C+-----+
2022      C
2023      DO IGWT = 1,LNGWT
2024      C
2025      C CHECK TO SEE IF THERE ARE WEAPON TYPES IN THIS GENERIC CATEGORY
2026      C
2027      IF (NWTGT(IGWT,L) .GT. 0) THEN
2028      TNWGT(IGWT,L) = 0.
2029      DO IWTGT = 1,NWTGT(IGWT,L)
2030      IW = IWTGT(IWTGT,IGWT,L)
2031      TNWGT(IGWT,L) = TNWGT(IGWT,L) + WSI(IW,IS,L)
2032      ENDDO
2033      ENDIF
2034      ENDDO
2035      C

```

```

2036 C+
2037 C+
2038 C+ COMPUTE THE ALLOCATION OF WEAPON TYPE VS WEAPON TYPE.
2039 C+
2040 C+ THE ALLOCATION OF WEAPON TYPE VS WEAPON TYPE IS COMPUTED FROM THE
2041 C+ ALLOCATION OF WEAPON TYPE VS GENERIC WEAPON CATEGORY:
2042 C+
2043 C+ THE ALLOCATION VS A WEAPON TYPE IS THE ALLOCATION VS THE GENERIC
2044 C+ WEAPON CATEGORY TIME THE PERCENT THAT THE WEAPON TYPE CONTRIBUTES
2045 C+ TO THE TOTAL NUMBER OF WEAPONS IN THE GENERIC WEAPON CATEGORY.
2046 C+
2047 C+ 1E:
2048 C+
2049 C+ FOR GENERIC WEAPON CATEGORY(IGWT) 4 (TANKS):
2050 C+   SPPE: WEAPON TYPES 4 & 15 ARE OF GWT 4
2051 C+   AND THAT TYPE 4 WEAPONS CONTRIBUTE 80% OF GWT 4
2052 C+   AND THAT TYPE 15 WEAPONS CONTRIBUTE 20% OF GWT 4
2053 C+   AND THAT THE ALLOCATION OF WT 1 VS GWT 4 = .2
2054 C+
2055 C+ THEN:
2056 C+
2057 C+ THE RESULTING ALLOCATION OF WT 1 VS WT 4 IS:
2058 C+   .2 * .8 = .16
2059 C+ THE RESULTING ALLOCATION OF WT 1 VS WT 15 IS:
2060 C+   .2 * .2 = .04
2061 C+
2062 C+ THERE IS A 20% ALLOCATION VERSUS TANKS OF WHICH
2063 C+ 16% IS VERSUS TANKS OF TYPE 4
2064 C+ AND 4% IS VERSUS TANKS OF TYPE 15
2065 C+
2066 C+
2067 C+
2068 C+
2069 C+
2070 C+ COMPUTE THE PERCENTAGE THAT EACH WEAPON TYPE CONTRIBUTES TO THE
2071 C+ NUMBER OF WEAPONS IN THE GENERIC WEAPON CATEGORY OF WHICH IT IS A
2072 C+ MEMBER.
2073 C+
2074 C+
2075 C+
2076 C+ IF (IT.EQ.1) THEN
2077 C+   DO IW = 1,N3
2078 C+     IGWT = LNKWPT(IW,L)
2079 C+     PWIC = 1.
2080 C+     IF (IGWT.GT.0) THEN
2081 C+       IF (TNWIGT(IGWT,L).GT.0.) THEN
2082 C+         PWIC = WSI(IW,IS,L) / TNWIGT(IGWT,L)
2083 C+       ENDIF
2084 C+     ELSE
2085 C+       IGWT = IW
2086 C+     ENDIF
2087 C+   DO JW = 1,N4
2088 C+     SAMA(JW,IW,11) = PWIC * BSAMA(JW,IGWT,KPA)
2089 C+     SAMD(JW,IW,11) = PWIC * BSAMD(JW,IGWT,KPD)
2090 C+     DO ISM = 1,NSM(1)
2091 C+       BFRASM(JW,IW,ISM) = PWIC * BFRASM(JW,IGWT,ISM)
2092 C+       BFRDSM(JW,IW,ISM) = PWIC * BFRDSM(JW,IGWT,ISM)

```

```

2093      ENDDO
2094      ENDDO
2095      DO IAM = 1,NAM(II)
2096          SSAMA(IAM,IW,II) = PWIC * SAMA(IAM,IGWT,II)
2097          SSAMD(IAM,IW,II) = PWIC * SAMD(IAM,IGWT,II)
2098      ENDDO
2099      DO IMM = 1,NAM(II)
2100          SSAMA(IMM,IW,II) = PWIC * SAMA(IMM,IGWT,II)
2101          SSAMD(IMM,IW,II) = PWIC * SAMD(IMM,IGWT,II)
2102      ENDDO
2103      ENDDO
2104      ELSE
2105          DO IW = 1,N3
2106              IGWT = LNKWPT(IW,L)
2107              PWIC = 1.
2108              IF (IGWT.GT. 0) THEN
2109                  IF (TNWGT(IGWT,L).GT. 0.) THEN
2110                      PWIC = WSI(IW,IS,L) / TNWGT(IGWT,L)
2111                  ENDIF
2112              ELSE
2113                  IGWT = IW
2114              ENDIF
2115              DO JW = 1,N4
2116                  SAMA(JW,IW,II) = PWIC * SAMA(JW,IGWT,KPA)
2117                  SAMD(JW,IW,II) = PWIC * SAMD(JW,IGWT,KPD)
2118                  DO ISM = 1, NSM(2)
2119                      RFRASM(JW,IW,ISM) = PWIC * RFRASM(JW,IGWT,ISM)
2120                      RFRDSM(JW,IW,ISM) = PWIC * RFRDSM(JW,IGWT,ISM)
2121                  ENDDO
2122              ENDDO
2123              DO IAM = 1,NAM(II)
2124                  SSAMA(IAM,IW,II) = PWIC * SAMA(IAM,IGWT,II)
2125                  SSAMD(IAM,IW,II) = PWIC * SAMD(IAM,IGWT,II)
2126              ENDDO
2127              DO IMM = 1,NAM(II)
2128                  SSAMA(IMM,IW,II) = PWIC * SAMA(IMM,IGWT,II)
2129                  SSAMD(IMM,IW,II) = PWIC * SAMD(IMM,IGWT,II)
2130              ENDDO
2131              ENDDO
2132              ENDIF
2133              IF (MYCASSO.EQ. 3) THEN
2134                  DO IW = 1,N3
2135                      DO JW = 1,N4
2136                          AAWA(JW,IW,II) = SAMA(JW,IW,II)
2137                          AAWD(JW,IW,II) = SAMD(JW,IW,II)
2138                      ENDDO
2139                  ENDDO
2140                  ELSE
2141
2142      C      COMPUTE THE ADJUSTED ALLOCATIONS OF SIDE II'S WEAPONS ON THE ATTACK
2143      C      AND ON THE DEFENSE AGAINST SIDE I'S WEAPONS, BASED ON THE MIX OF
2144      C      WEAPONS IN THIS SECTOR.
2145      C
2146          DO 1035 JW = 1,N4      ISIDE II'S WEAPONS
2147              SUMA = 0.
2148              SUMD = 0.
2149              DO 1025 IW = 1,N3      ISIDE I'S WEAPONS

```

```

2150 AAWA(JW,IW,II) = 0.
2151 AAWD(JW,IW,II) = 0.
2152 SUMA = SUMA + SAAWA(JW,IW,II) * WSI(IW,IS,L)
2153 SUMD = SUMD + SAAMD(JW,IW,II) * WSI(IW,IS,L)
2154 CONTINUE
2155 IF (SUMA.GT.0.) THEN
2156   DO 1026 IW = 1,N3
2157     AAWA(JW,IW,II) = (SAAWA(JW,IW,II) * WSI(IW,IS,L))/SUMA
2158     CONTINUE
2159   ENDIF
2160   IF (SUMD.GT.0.) THEN
2161     DO 1027 IW = 1,N3
2162       AAWD(JW,IW,II) = (SAAMD(JW,IW,II) * WSI(IW,IS,L))/SUMD
2163       CONTINUE
2164     ENDIF
2165   CONTINUE
2166   ENDF
2167 C+
2168 C+ CYCLE-BY-CYCLE DYNAMIC ADJUSTMENT OF WEAPON ENGAGEMENT RATES
2169 C+
2170 C   IF (ICYLE.EQ.1) THEN
2171     DO KP = 1,LNPTP
2172       DO JW = 1,N4
2173         EERTA(JW,KP,II) = ERTA(JW,KP,II)
2174         EERTD(JW,KP,II) = ERTD(JW,KP,II)
2175       ENDDO
2176     ENDDO
2177   ELSE
2178     DO JW = 1,N4
2179       WTGWVA(JW,II) = 0.0
2180       WTGWVD(JW,II) = 0.0
2181       DO IW = 1,N3
2182         WTGWVA(JW,II) = WTGWVA(JW,II) + AAWA(JW,IW,II) *
2183           WSI(IW,IS,L) + VIWDCF(IW,IS,L)
2184         WTGWVD(JW,II) = WTGWVD(JW,II) + AAWD(JW,IW,II) *
2185           WSI(IW,IS,L) + VIWDCF(IW,IS,L)
2186       ENDDO
2187       WSGWVA(JW,II) = VIWACF(JW,IS,II) * WSI(JW,IS,II)
2188       WSGWVD(JW,II) = VIWDCF(JW,IS,II) * WSI(JW,IS,II)
2189     ENDDO
2190     DO JW = 1,N4
2191       IF (WTGWVA(JW,II).GT..001) THEN
2192         RSTVA(JW,IS,II) = WSGWVA(JW,II) / WTGWVA(JW,II)
2193       ENDIF
2194       IF (WTGWVD(JW,II).GT..001) THEN
2195         RSTDV(JW,IS,II) = WSGWVD(JW,II) / WTGWVD(JW,II)
2196       ENDIF
2197     ENDDO
2198     DO JW = 1,N4
2199       CALL CVFW (NERTA(JW,II),XAERTA(1,JW,II),YAERTA(1,JW,II),
2200         RSTVA(JW,IS,II),AERTA(JW,IS,II))
2201       CALL CVFW (NERTD(JW,II),XAERTD(1,JW,II),YAERTD(1,JW,II),
2202         RSTDV(JW,IS,II),AERTD(JW,IS,II))
2203     ENDDO
2204     DO KP = 1,LNPTP
2205       DO JW = 1,N4
2206         EERTA(JW,KP,II) = ERTA(JW,KP,II) + AERTA(JW,IS,II)

```

```

2207 C      EERTD(JW,KP,II) = ERTD(JW,KP,II) * AERTD(JW,IS,II)
2208 C      ENDDO
2209 C      ENDDO
2210 C      ENDDO
2211 C      ENDDO
2212 C      SET ALLOCATION FOR WEAPONS USING SMART (PREFERRED) AND NON-PREFERRED
2213 C      MUNITIONS
2214 C
2215 C      DO ISM = 1,NSM(II)
2216 C      DO IW = 1,N3
2217 C      DO JW = 1,N4
2218 C      BAWASM(JW,IW,ISM) = 0.
2219 C      BAWDSM(JW,IW,ISM) = 0.
2220 C      RAWASM(JW,IW,ISM) = 0.
2221 C      RAWDSM(JW,IW,ISM) = 0.
2222 C      BVWASM(JW,IW,ISM) = 0.
2223 C      BVWDSM(JW,IW,ISM) = 0.
2224 C      RVWASM(JW,IW,ISM) = 0.
2225 C      RVWDSM(JW,IW,ISM) = 0.
2226 C      ENDDO
2227 C      ENDDO
2228 C      ENDDO
2229 C      IF (II.EQ. 1) THEN
2230 C      DO ISM = 1,NSM(II)
2231 C
2232 C      CHECK TO SEE IF THERE ARE POSSIBLE FIRERS AND TARGETS FOR
2233 C      C PREFERRED MUNITION ISM
2234 C
2235 C      IF (NSMW(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2236 C
2237 C      THE ARRAY ISMW HOLDS THE WEAPONS TYPES WHICH MAY FIRE PREFERRED
2238 C      MUNITION ISM.
2239 C
2240 C      DO INSMW = 1, NSMW(ISM,II)
2241 C      JW = ISMW(INSMW,ISM,II)
2242 C
2243 C      CHECK TO SEE IF THERE ARE ANY OF THE FIRERS IN CONTACT
2244 C
2245 C      IF (WSI(JW,IS,II) .GT. 0.) THEN
2246 C
2247 C      THE ARRAY ISMFR CONTAINS THE WEAPON TYPES AT WHICH TO FIRE PREFERRED
2248 C      MUNITION ISM.
2249 C
2250 C      DO INSMFR = 1,NSMFR(ISM,II)
2251 C      IW = ISMFR(INSMFR,ISM,II)
2252 C      BAWASM(JW,IW,ISM) = AAWA(JW,IW,II) *
2253 C      BBFRASM(JW,IW,ISM)
2254 C      BAWDSM(JW,IW,ISM) = AAWD(JW,IW,II) *
2255 C      BBFRDSM(JW,IW,ISM)
2256 C      ENDDO ! TARGETS AT WHICH TO FIRE
2257 C      ENDDO ! IF THERE ARE FIRERS AVAILABLE
2258 C      ENDDO ! POSSIBLE FIRERS
2259 C      ENDDO ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2260 C      ENDDO ! PREFERRED MUNITIONS
2261 C      ELSE
2262 C      DO ISM = 1,NSM(II)
2263 C

```

```

2264 C CHECK TO SEE IF THERE ARE POSSIBLE FIRERS AND TARGETS FOR
2265 C PREFERRED MUNITION ISM
2266 C
2267 C IF (NSMW(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2268 C
2269 C THE ARRAY ISMW HOLDS THE WEAPONS TYPES WHICH MAY FIRE PREFERRED
2270 C MUNITION ISM.
2271 C
2272 C DO ISMW = 1, NSMW(ISM,II)
2273 C JW = ISMW(INSMW,ISM,II)
2274 C
2275 C CHECK TO SEE IF THERE ARE ANY OF THE FIRERS IN CONTACT
2276 C
2277 C IF (WSI(JW,IS,II) .GT. 0.) THEN
2278 C
2279 C THE ARRAY ISMFR CONTAINS THE WEAPON TYPES AT WHICH TO FIRE PREFERRED
2280 C MUNITION ISM.
2281 C
2282 C DO ISMFR = 1, NSMFR(ISM,II)
2283 C IW = ISMFR(INSMFR,ISM,II)
2284 C RAWASM(JW,IW,ISM) = AAWA(JW,IW,II) *
2285 C RFRFRASM(JW,IW,ISM)
2286 C RAWDSM(JW,IW,ISM) = AAWD(JW,IW,II) *
2287 C RFRFRDSM(JW,IW,ISM)
2288 C
2289 C ENDDO ! TARGETS AT WHICH TO FIRE
2290 C ENDIF ! IF THERE ARE FIRERS AVAILABLE
2291 C ENDDO ! POSSIBLE FIRERS
2292 C ENDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2293 C ENDDO ! PREFERRED MUNITIONS
2294 C ENDIF
2295 C+
2296 C+
2297 C+ COMPUTE THE NUMBER OF ROUNDS OF PREFERRED MUNITION TYPE ISM WHICH SIDE II +
2298 C+ WISHES TO FIRE BASED ON: +
2299 C+
2300 C+ THE NUMBER OF WEAPON SYSTEMS FIRING MUNITION TYPE ISM +
2301 C+ THE ENGAGEMENT RATE OF THOSE SYSTEMS +
2302 C+ AND THE ALLOCATION OF THOSE SYSTEMS WHEN FIRING MUNITION TYPE ISM +
2303 C+
2304 C+
2305 C DO ISM = 1, NSM(II)
2306 C BSHTRA(ISM) = 0.
2307 C BSHTRD(ISM) = 0.
2308 C RSHTRA(ISM) = 0.
2309 C RSHTRD(ISM) = 0.
2310 C BTRGT(ISM) = 0.
2311 C RTRGT(ISM) = 0.
2312 C ENDDO
2313 C IF (II .EQ. 1) THEN
2314 C DO ISM = 1, NSM(II)
2315 C IF (NSMW(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2316 C DO INSMW = 1, NSMW(ISM,II)
2317 C JW = ISMW(INSMW,ISM,II)
2318 C IF (WSI(JW,IS,II) .GT. 0.) THEN
2319 C DO ISMFR = 1, NSMFR(ISM,II)
2320 C IW = ISMFR(INSMFR,ISM,II)

```

```

2321 C      BSHTRA(ISM) = BSHTRA(ISM) + EERTA(JW,KPA,II) *
2322 C      BAWASM(JW,IW,ISM) * WSI(JW,IS,II)
2323 C      BSHTRD(ISM) = BSHTRD(ISM) + EERTD(JW,KPD,II) *
2324 C      BAWDSM(JW,IW,ISM) * WSI(JW,IS,II)
2325 C      BSTRAS(ISM) = BSHTRA(ISM) ! STORE BLUE SHOOTERS ON ATTACK
2326 C      BSTRDS(ISM) = BSHTRD(ISM) ! STORE BLUE SHOOTERS ON DEFENSE
2327 C      ENDDO ! TARGETS AT WHICH TO FIRE
2328 C      ENDIF ! IF THERE ARE FIRERS AVAILABLE
2329 C      ENDDO ! POSSIBLE FIRERS
2330 C      ENDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2331 C      ENDDO ! PREFERRED MUNITIONS
2332 C      ELSE ! II IS SIDE 2
2333 C      DO ISM = 1,NSM(II)
2334 C      IF (NSMAM(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2335 C      DO INSMAM = 1, NSMAM(ISM,II)
2336 C      JW = ISMAM(INSMAM,ISM,II)
2337 C      IF (WSI(JW,IS,II) .GT. 0.) THEN
2338 C      DO INSMFR = 1,NSMFR(ISM,II)
2339 C      IW = ISMFR(INSMFR,ISM,II)
2340 C      RSHTRA(ISM) = RSHTRA(ISM) + EERTA(JW,KPA,II) *
2341 C      RAWASM(JW,IW,ISM) * WSI(JW,IS,II)
2342 C      RSHTRD(ISM) = RSHTRD(ISM) + EERTD(JW,KPD,II) *
2343 C      RAWDSM(JW,IW,ISM) * WSI(JW,IS,II)
2344 C      RSTRAS(ISM) = RSHTRA(ISM) ! STORE RED SHOOTERS ON ATTACK
2345 C      RSTRDS(ISM) = RSHTRD(ISM) ! STORE RED SHOOTERS ON DEFENSE
2346 C      ENDDO ! TARGETS AT WHICH TO FIRE
2347 C      ENDIF ! IF THERE ARE FIRERS AVAILABLE
2348 C      ENDDO ! POSSIBLE FIRERS
2349 C      ENDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2350 C      ENDDO ! PREFERRED MUNITIONS
2351 C      ENDIF
2352 C+-----+
2353 C+
2354 C+ COMPUTE THE NUMBER OF TARGETS AVAILABLE AT WHICH PREFERRED MUNITION ISM
2355 C+ WILL BE FIRED.
2356 C+-----+
2357 C+
2358 C
2359 C      IF (II .EQ. 1) THEN
2360 C      DO ISM = 1,NSM(II)
2361 C      IF (NSMAM(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2362 C      DO INSMAM = 1, NSMAM(ISM,II)
2363 C      JW = ISMAM(INSMAM,ISM,II)
2364 C      IF (WSI(JW,IS,II) .GT. 0.) THEN
2365 C      DO INSMFR = 1,NSMFR(ISM,II)
2366 C      IW = ISMFR(INSMFR,ISM,II)
2367 C      BTRGT(ISM) = BTRGT(ISM) + WSI(IW,IS,L)
2368 C      ENDDO ! TARGETS AT WHICH TO FIRE
2369 C      ENDIF ! IF THERE ARE FIRERS AVAILABLE
2370 C      ENDDO ! POSSIBLE FIRERS
2371 C      ENDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2372 C      ENDDO ! PREFERRED MUNITIONS
2373 C      ELSE
2374 C      DO ISM = 1,NSM(II)
2375 C      IF (NSMAM(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2376 C      DO INSMAM = 1, NSMAM(ISM,II)
2377 C      JW = ISMAM(INSMAM,ISM,II)

```

```

2378 C      IF (WSI(JW,IS,II) .GT. 0.) THEN
2379 C      DO INSMFR = 1,NSMFR(ISM,II)
2380 C      IW = ISMFR(INSMFR,ISM,II)
2381 C      RTRGT(ISM) = RTRGT(ISM) + WSI(IW,IS,L)
2382 C      ENDDO ! TARGETS AT WHICH TO FIRE
2383 C      ENDIF ! IF THERE ARE FIRERS AVAILABLE
2384 C      ENDDO ! POSSIBLE FIRERS
2385 C      ENDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2386 C      ENDDO ! PREFERRED MUNITIONS
2387 C      ENDIF
2388 C+-----+
2389 C+
2390 C+ CHECK TO SEE IF THERE IS ENOUGH STOCK OF PREFERRED MUNITION ISM BASED
2391 C+ THE NUMBER TO FIRE.
2392 C+
2393 C+-----+
2394 C      IF (II .EQ. 1) THEN
2395 C      DO ISM = 1,NSM(II)
2396 C      IF (BSHTRA(ISM) .GT. SMSTK(ISM,IS,II)) THEN
2397 C      BSHTRA(ISM) = SMSTK(ISM,IS,II)
2398 C      ENDIF
2399 C      IF (BSHTRD(ISM) .GT. SMSTK(ISM,IS,II)) THEN
2400 C      BSHTRD(ISM) = SMSTK(ISM,IS,II)
2401 C      ENDIF
2402 C      ENDDO
2403 C      ELSE
2404 C      DO ISM = 1,NSM(II)
2405 C      IF (RSHTRA(ISM) .GT. SMSTK(ISM,IS,II)) THEN
2406 C      RSHTRA(ISM) = SMSTK(ISM,IS,II)
2407 C      ENDIF
2408 C      IF (RSHTRD(ISM) .GT. SMSTK(ISM,IS,II)) THEN
2409 C      RSHTRD(ISM) = SMSTK(ISM,IS,II)
2410 C      ENDIF
2411 C      ENDDO
2412 C      ENDIF
2413 C+-----+
2414 C+
2415 C+ MODIFY THE ALLOCATION OF PREFERRED MUNITION TO REFLECT LESS TARGETS THEN
2416 C+ THE AMOUNT OF PREFERRED MUNITION ISM TO BE FIRED.
2417 C+
2418 C+-----+
2419 C+
2420 C      IF (II .EQ. 1) THEN
2421 C      DO ISM = 1,NSM(II)
2422 C      IF (BSHTRA(ISM) .GT. BTRGT(ISM)) THEN
2423 C      DO IW = 1,N3
2424 C      DO JW = 1,N4
2425 C      BBFRASM(JW,IW,ISM) = BBFRASM(JW,IW,ISM) *
2426 C      BTRGT(ISM)/BSHTRA(ISM)
2427 C      ENDDO
2428 C      ENDDO
2429 C      ELSE IF (BSHTRA(ISM) .LT. BSTRAS(ISM)) THEN
2430 C      DO IW = 1,N3
2431 C      DO JW = 1,N4
2432 C      BBFRASM(JW,IW,ISM) = BBFRASM(JW,IW,ISM) *
2433 C      BSHTRA(ISM)/BSTRAS(ISM)
2434 C      ENDDO

```

```

2435 C      ENDDO
2436 C      ENDIF
2437 C      IF (BSHTRD(ISM) .GT. BTRGT(ISM)) THEN
2438 C      DO IW = 1,N3
2439 C      DO JW = 1,N4
2440 C      BFRDSM(JW,IW,ISM) = BFRDSM(JW,IW,ISM) *
2441 C      BTRGT(ISM)/BSHTRD(ISM)
2442 C      *
2443 C      ENDDO
2444 C      ENDDO
2445 C      ELSE IF (BSHTRD(ISM) .LT. BSTRDS(ISM)) THEN
2446 C      DO IW = 1,N3
2447 C      DO JW = 1,N4
2448 C      BFRDSM(JW,IW,ISM) = BFRDSM(JW,IW,ISM) *
2449 C      BSHTRD(ISM)/BSTRDS(ISM)
2450 C      *
2451 C      ENDDO
2452 C      ENDDO
2453 C      ENDIF
2454 C      ENDDO
2455 C      ELSE
2456 C      DO ISM = 1,NSM(11)
2457 C      IF (RSHTRA(ISM) .GT. RTRGT(ISM)) THEN
2458 C      DO IW = 1,N3
2459 C      DO JW = 1,N4
2460 C      RFRASM(JW,IW,ISM) = RFRASM(JW,IW,ISM) *
2461 C      RTRGT(ISM)/RSHTRA(ISM)
2462 C      *
2463 C      ENDDO
2464 C      ENDDO
2465 C      ELSE IF (RSHTRA(ISM) .LT. RSTRAS(ISM)) THEN
2466 C      DO IW = 1,N3
2467 C      DO JW = 1,N4
2468 C      RFRASM(JW,IW,ISM) = RFRASM(JW,IW,ISM) *
2469 C      RSHTRA(ISM)/RSTRAS(ISM)
2470 C      *
2471 C      ENDDO
2472 C      ENDDO
2473 C      ENDIF
2474 C      IF (RSHTRD(ISM) .GT. RTRGT(ISM)) THEN
2475 C      DO IW = 1,N3
2476 C      DO JW = 1,N4
2477 C      RFRDSM(JW,IW,ISM) = RFRDSM(JW,IW,ISM) *
2478 C      RTRGT(ISM)/RSHTRD(ISM)
2479 C      *
2480 C      ENDDO
2481 C      ENDDO
2482 C      ELSE IF (RSHTRD(ISM) .LT. RSTRDS(ISM)) THEN
2483 C      DO IW = 1,N3
2484 C      DO JW = 1,N4
2485 C      RFRDSM(JW,IW,ISM) = RFRDSM(JW,IW,ISM) *
2486 C      RSHTRD(ISM)/RSTRDS(ISM)
2487 C      *
2488 C      ENDDO
2489 C      ENDDO
2490 C      ENDIF
2491 C      ENDDO

```

```

C+ RECOMPUTE THE ALLOCATION IF THE PREVIOUSLY COMPUTED ALLOCATION HAS BEEN
C+ MODIFIED BASED ON THE ABOVE CALCULATIONS.
C+
C+

```

```

2492 C      IF (II.EQ.1) THEN
2493 C      DO ISM = 1,NSM(II)
2494 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2495 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2496 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2497 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2498 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2499 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2500 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2501 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2502 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2503 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2504 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2505 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2506 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2507 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2508 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2509 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2510 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2511 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2512 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2513 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2514 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2515 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2516 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2517 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2518 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2519 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2520 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2521 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2522 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2523 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2524 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2525 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2526 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2527 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2528 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2529 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2530 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2531 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2532 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2533 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2534 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2535 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2536 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2537 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2538 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2539 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2540 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2541 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2542 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2543 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2544 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2545 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2546 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2547 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN
2548 C      IF (BSHTRA(ISM).GT.0..OR..BSHTRD(ISM).GT.0.) THEN

```

```

2549      ENDDO
2550      DO IW = 1,N3
2551      DO JW = 1,N4
2552      DO ISM = 1,NSM(IJ)
2553      BSUMA(JW,IW,IJ) = BSUMA(JW,IW,IJ) + BAWASM(JW,IW,ISM)
2554      BSUMD(JW,IW,IJ) = BSUMD(JW,IW,IJ) + BAWDSM(JW,IW,ISM)
2555      ENDDO
2556      AAWA(JW,IW,IJ) = AAWA(JW,IW,IJ) + BSUMA(JW,IW,IJ)
2557      AAWD(JW,IW,IJ) = AAWD(JW,IW,IJ) + BSUMD(JW,IW,IJ)
2558      ENDDO
2559      ENDDO
2560      ELSE
2561      DO IW = 1,N3
2562      DO JW = 1,N4
2563      RSUMA(JW,IW,IJ) = 0.0
2564      RSUMD(JW,IW,IJ) = 0.0
2565      ENDDO
2566      ENDDO
2567      DO IW = 1,N3
2568      DO JW = 1,N4
2569      DO ISM = 1,NSM(IJ)
2570      RSUMA(JW,IW,IJ) = RSUMA(JW,IW,IJ) + RAWASM(JW,IW,ISM)
2571      RSUMD(JW,IW,IJ) = RSUMD(JW,IW,IJ) + RAWDSM(JW,IW,ISM)
2572      ENDDO
2573      AAWA(JW,IW,IJ) = AAWA(JW,IW,IJ) + RSUMA(JW,IW,IJ)
2574      AAWD(JW,IW,IJ) = AAWD(JW,IW,IJ) + RSUMD(JW,IW,IJ)
2575      ENDDO
2576      ENDDO
2577      ENDIF
2578      C -----
2579      C POSTURE DEPENDENT ENGAGEMENT RATE AND KILL PROBABILITIES
2580      C -----
2581      IF (IJ.EQ.1) THEN
2582      DO ISM = 1,NSM(IJ)
2583      IF (NSMW(ISM,IJ).GT.C.AND. NSMFR(ISM,IJ).GT.1) THEN
2584      DO INSMW = 1,NSMW(ISM,IJ)
2585      JW = ISMW(INSMW,ISM,IJ)
2586      IF (WSI(JW,IS,IJ).GT.0.) THEN
2587      DO INSMFR = 1,NSMFR(ISM,IJ)
2588      IW = ISMFR(INSMFR,ISM,IJ)
2589      RVWASM(JW,IW,ISM) = CERTA(JW,KPA,IJ) *
2590      BPKASM(JW,IW,ISM) *
2591      BPKASM(KPA)
2592      EERTD(JW,KPD,IJ) *
2593      BPKDSM(JW,IW,ISM) *
2594      BPKDSM(KPD)
2595      ENDDO ! TARGETS AT WHICH TO FIRE
2596      ENDDIF ! IF THERE ARE FIRERS AVAILABLE
2597      ENDDO ! POSSIBLE FIRERS
2598      ENDDIF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
2599      ENDDO ! PREFERRED MUNITIONS
2600      ELSE
2601      DO ISM = 1,NSM(IJ)
2602      IF (NSMW(ISM,IJ).GT.C.AND. NSMFR(ISM,IJ).GT.1) THEN
2603      DO INSMW = 1,NSMW(ISM,IJ)
2604      JW = ISMW(INSMW,ISM,IJ)
2605      IF (WSI(JW,IS,IJ).GT.0.) THEN

```

```

DO ISMFR = 1, NSMFR(ISM, II)
  IW = ISMFR(ISMFR, ISM, II)
  RVWASM(JW, IW, ISM) = EERTA(JW, KPA, II) *
    RPKASM(JW, IW, ISM) *
    FPKASM(KPA)
  RVWDSM(JW, IW, ISM) = EERTD(JW, KPD, II) *
    RPKDSM(JW, IW, ISM) *
    FPKDSM(KPD)
  ENDDO
  I TARGETS AT WHICH TO FIRE
  ENDOF
  I IF THERE ARE FIRERS AVAILABLE
  ENDDO
  I POSSIBLE FIRERS
  ENDOF
  I IF THERE ARE POSSIBLE FIRERS AND TARGETS
  ENDDO
  I PREFERRED MUNITIONS
  ENDOF

```

C KILL PROBABILITIES FOR NON-PREFERRED MUNITIONS FOR POSTURE
C AND VALUE OF WEAPONS ON ATTACK AND DEFENSE
C

```

IF (II.EQ. 1) THEN
  DO IW = 1, N3
    DO JW = 1, N4
      VIWAW(JW, IW, II) = EERTA(JW, KPA, II) * BPKAW(JW, IW, KPA)
      VIWDW(JW, IW, II) = EERTD(JW, KPD, II) * BPKDW(JW, IW, KPD)
    ENDDO
  ENDDO
ELSE
  DO IW = 1, N3
    DO JW = 1, N4
      VIWAW(JW, IW, II) = EERTA(JW, KPA, II) * RPKAW(JW, IW, KPA)
      VIWDW(JW, IW, II) = EERTD(JW, KPD, II) * RPKDW(JW, IW, KPD)
    ENDDO
  ENDDO
ENDIF

```

C KILL RATE MATRICES FOR USE BY PREFERRED AND NON-PREFERRED MUNITIONS
C

```

IF (II.EQ. 1) THEN
  DO ISM = 1, NSM(II)
    DO IW = 1, N3
      DO JW = 1, N4
        PWAWSM(JW, IW, ISM, II) = BVWASM(JW, IW, ISM) *
          BAWASM(JW, IW, ISM)
        PWOKSM(JW, IW, ISM, II) = BVWDSM(JW, IW, ISM) *
          BAWDSM(JW, IW, ISM)
      ENDDO
    ENDDO
  ELSE
    DO ISM = 1, NSM(II)
      DO IW = 1, N3
        DO JW = 1, N4
          PWAWSM(JW, IW, ISM, II) = RVWASM(JW, IW, ISM) *
            RAWASM(JW, IW, ISM)
          PWOKSM(JW, IW, ISM, II) = RVWDSM(JW, IW, ISM) *
            RAWDSM(JW, IW, ISM)
        ENDDO
      ENDDO
    ENDDO
  ENDDO

```

```

2663      ENDDO
2664      ENDIF
2665
2666      C COMPUTE THE KILL POTENTIALS FOR JW TYPE WEAPONS OF SIDE II AGAINST
2667      C IW TYPE WEAPONS FOR SIDE L. WHEN SIDE II IS ON THE ATTACK AND ON THE
2668      C DEFENSE
2669      C
2670      DO IW = 1,N3
2671      DO JW = 1,N4
2672      PWAKW(JW,IW,II) = VIWAW(JW,IW,II) * AAWA(JW,IW,II)
2673      PWOKW(JW,IW,II) = VIWDW(JW,IW,II) * AAWD(JW,IW,II)
2674      ENDDO
2675      ENDDO
2676      DO IW = 1,N3
2677      DO JW = 1,N4
2678      PWSUMA(JW,IW,II) = 0.
2679      PWSUMD(JW,IW,II) = 0.
2680      DO ISM = 1,NSM(II)
2681      PWSUMA(JW,IW,II) = PWSUMA(JW,IW,II) +
2682      & PWAKSM(JW,IW,ISM,II)
2683      PWSUMD(JW,IW,II) = PWSUMD(JW,IW,II) +
2684      & PWOKSM(JW,IW,ISM,II)
2685      ENDDO
2686      ENDDO
2687      ENDDO
2688
2689      C COMPOSITE KILL RATE MATRICES FOR USE OF NON-PREFERRED
2690      C AND PREFERRED MUNITIONS
2691      C
2692      DO IW = 1,N3
2693      DO JW = 1,N4
2694      PWAKW(JW,IW,II) = PWAKW(JW,IW,II) + PWSUMA(JW,IW,II)
2695      PWOKW(JW,IW,II) = PWOKW(JW,IW,II) + PWSUMD(JW,IW,II)
2696      ENDDO
2697      ENDDO
2698
2699      C PRINT IN FILE TABLOUT.DAT TEMPORARILY FOR DEBUGGING
2700      C
2701      IF (KPSMGA(15).EQ.1) THEN
2702      WRITE (MOT,644) IS,(APT(KPS(1S),IWRD),IWRD=1,2)
2703      FORMAT ('1',' SECTOR',14,' POSTURE',2A4)
2704      NN = NW(L)
2705      IENDW = 0
2706      LINCNT = 0
2707      DO WHILE (NN.GT.0)
2708      ISTW = IENDW + 1
2709      IF (NN.GT.12) THEN
2710      IENDW = IENDW + 12
2711      ELSE
2712      IENDW = IENDW + NN
2713      ENDIF
2714      NUMPS = (IENDW - ISTW) + 1
2715      WRITE (MOT,600) II,L,ICYCLE,
2716      & ((ANT(IW,IWRD,L),IWRD=1,2),IW=ISTW,IENDW)
2717      FORMAT ('/',' ADJUSTED ALLOCATION (ATTACK) --- SIDE',13,
2718      & ', AGAINST SIDE',13,' CYCLE',14/9X,12(1X,2A4)/)
2719      DO JW = 1,N4

```

```

2720      &
2721      602      WRITE (MOT,602)(AWT(JW,IWRD,11),IWRD=1,2),
2722              (AAWA(JW,IW,11),IW=1STW,IENDW)
2723      FORMAT (1X,2A4,12(1X,F8.3))
2724      LINCNT = LINCNT + 1
2725      ENDDO
2726      NN = NN - NUMPS
2727      ENDDO
2728      NN = NW(L)
2729      IENDW = 0
2730      XNLIN = (N3 / 12.) + .5
2731      NUMLIN = LINCNT + XNLIN
2732      IF (NUMLIN.GT. 45) THEN
2733      WRITE (MOT,644) IS,(APT(KPS(1S),IWRD),IWRD=1,2)
2734      LINCNT = 0
2735      ENDIF
2736      DO WHILE (NN.GT. 0)
2737      ISTW = IENDW + 1
2738      IF (NN.GT. 12) THEN
2739      IENDW = IENDW + 12
2740      ELSE
2741      IENDW = IENDW + NN
2742      ENDIF
2743      NUMPS = (IENDW - ISTW) + 1
2744      WRITE (MOT,603) 11,L,ICYCLE,
2745              ((AWT(IW,IWRD,L),IWRD=1,2),IW=1STW,IENDW)
2746      FORMAT (/, ' ADJUSTED ALLOCATION (DEFENSE) -- SIDE',13,
2747              ' AGAINST SIDE',13, ' CYCLE',14/9X,12(1X,2A4)/)
2748      DO JW = 1,N4
2749      WRITE (MOT,602)(AWT(JW,IWRD,11),IWRD=1,2),
2750              (AAWD(JW,IW,11),IW=1STW,IENDW)
2751      LINCNT = LINCNT + 1
2752      ENDDO
2753      NN = NN - NUMPS
2754      ENDDO
2755      IF (11.EQ. 1) THEN
2756      DO ISM = 1,NSM(11)
2757      WRITE (MOT,606) ISM,ICYCLE
2758      FORMAT (/, ' ADJUSTED ALLOCATION OF BLUE (ATTACK)',
2759              ' FOR PREFERRED MUNITION TYPE',13, ' CYCLE',14)
2760      DO INSMW = 1,NSMW(ISM,11)
2761      JW = ISMW(INSMW,ISM,11)
2762      WRITE (MOT,6002)(AWT(JW,IWORD,11),IWORD=1,2),
2763              (BAWASM(JW,IW,ISM),IW=1,N3)
2764      ENDDO
2765      FORMAT(1X,2A4,1X,12F8.3)
2766      WRITE (MOT,609) ISM,ICYCLE
2767      FORMAT (/, ' ADJUSTED ALLOCATION OF BLUE (DEFENSE)',
2768              ' FOR PREFERRED MUNITION TYPE',13, ' CYCLE',14)
2769      DO INSMW = 1,NSMW(ISM,11)
2770      JW = ISMW(INSMW,ISM,11)
2771      WRITE (MOT,6002)(AWT(JW,IWORD,11),IWORD=1,2),
2772              (BAWDSM(JW,IW,ISM),IW=1,N3)
2773      ENDDO
2774      WRITE (MOT,618) ISM,ICYCLE
2775      FORMAT (/, ' BLUE SINGLE WEAPON VALUE (ATTACK) FOR ',
2776              'PREFERRED MUNITION TYPE',13, ' CYCLE',14)
2777      DO INSMW = 1,NSMW(ISM,11)

```

```

2777 JW = ISMW(ISMW,ISM,II)
2778 WRITE (MOT,6002)(AWT(JW,IWORD,II),IWORD=1,2),
2779 (BVWASM(JW,IW,ISM),IW = 1,N3)
2780 ENDDO
2781 WRITE (MOT,621) ISM,ICYCLE
2782 FORMAT (/,' BLUE SINGLE WEAPON VALUE (DEFENSE)FOR ',
2783 'PREFERRED MUNITION TYPE',I3,' CYCLE',I4)
2784 DO ISMW = 1,NSMW(ISM,II)
2785 JW = ISMW(ISMW,ISM,II)
2786 WRITE (MOT,6002)(AWT(JW,IWORD,II),IWORD=1,2),
2787 (BVWDSM(JW,IW,ISM),IW = 1,N3)
2788 ENDDO
2789 ENDDO
2790 NN = NW(L)
2791 IENDW = 0
2792 LINCNT = 0
2793 WRITE(MOT,623)
2794 FORMAT('1')
2795 DO WHILE (NN.GT. 0)
2796 ISTW = IENDW + 1
2797 IF (NN.GT. 12) THEN
2798 IENDW = IENDW + 12
2799 ELSE
2800 IENDW = IENDW + NN
2801 ENDF
2802 NUMPS = (IENDW - ISTW) + 1
2803 WRITE (MOT,624) ICYCLE,
2804 ((AWT(IW,IWRD,L),IWRD=1,2),IW=ISTW,IENDW)
2805 FORMAT (/,' KILL RATE MATRIX FOR BLUE ON ATTACK CYCLE',
2806 14/9X,12(1X,2A4))
2807 DO JW = 1,N4
2808 WRITE (MOT,602)(AWT(JW,IWRD,II),IWRD=1,2),
2809 (PWAKW(JW,IW,II),IW=ISTW,IENDW)
2810 LINCNT = LINCNT + 1
2811 ENDDO
2812 NN = NN - NUMPS
2813 ENDDO
2814 NN = NW(L)
2815 IENDW = 0
2816 XNLI = (NN / 12.) + .5
2817 NUMLIN = LINCNT + XNLI
2818 IF (NUMLIN.GT. 45) THEN
2819 WRITE (MOT,623)
2820 LINCNT = 0
2821 ENDF
2822 DO WHILE (NN.GT. 0)
2823 ISTW = IENDW + 1
2824 IF (NN.GT. 12) THEN
2825 IENDW = IENDW + 12
2826 ELSE
2827 IENDW = IENDW + NN
2828 ENDF
2829 NUMPS = (IENDW - ISTW) + 1
2830 WRITE (MOT,627) ICYCLE,
2831 ((AWT(IW,IWRD,L),IWRD=1,2),IW=ISTW,IENDW)
2832 FORMAT (/,' KILL RATE MATRIX FOR BLUE ON DEFENSE CYCLE',
2833 14/9X,12(1X,2A4))

```

```

2834      DO JW = 1,N4
2835         WRITE (MOT,602)((AWT(JW,IWRD,II),IWRD=1,2),
2836            (PWAKW(JW,IW,II),IW=1STW,IENDW)
2837         LINCNT = LINCNT + 1
2838      ENDDO
2839      NN = NN - NUMMPS
2840      ENDDO
2841      ELSE
2842         DO ISM = 1,NSM(II)
2843            WRITE (MOT,612) ISM,ICYCLE
2844            FORMAT (/,' ADJUSTED ALLOCATION OF RED (ATTACK)',
2845               ' FOR PREFERRED MUNITION TYPE',I3,' CYCLE',I4)
2846            DO INSMW = 1,NSMW(ISM,II)
2847               JW = ISMW(INSMW,ISM,II)
2848               WRITE (MOT,6002)((AWT(JW,IWORD,II),IWORD=1,2),
2849                  (RAWASM(JW,IW,ISM),IW = 1,N3)
2850            ENDDO
2851            WRITE (MOT,615) ISM,ICYCLE
2852            FORMAT (/,' ADJUSTED ALLOCATION OF RED (DEFENSE)',
2853               ' FOR PREFERRED MUNITION TYPE',I3,' CYCLE',I4)
2854            DO INSMW = 1,NSMW(ISM,II)
2855               JW = ISMW(INSMW,ISM,II)
2856               WRITE (MOT,6002)((AWT(JW,IWORD,II),IWORD=1,2),
2857                  (RAWDSM(JW,IW,ISM),IW = 1,N3)
2858            ENDDO
2859            NN = NN(L)
2860            IENDW = 0
2861            LINCNT = 0
2862            WRITE(MOT,623)
2863            DO WHILE (NN.GT. 0)
2864               ISTW = IENDW + 1
2865               IF (NN.GT. 12) THEN
2866                  IENDW = IENDW + 12
2867               ELSE
2868                  IENDW = IENDW + NN
2869            ENDIF
2870            NUMMPS = (IENDW - ISTW) + 1
2871            WRITE (MOT,638) ICYCLE,
2872               ((AWT(IW,IWRD,I),IWRD=1,2),IW=1STW,IENDW)
2873            FORMAT (/,' KILL RATE MATRIX FOR RED ON ATTACK CYCLE',
2874               14/9X,12(1X,2A4))
2875            DO JW = 1,N4
2876               WRITE (MOT,602)((AWT(JW,IWRD,II),IWRD=1,2),
2877                  (PWAKW(JW,IW,II),IW=1STW,IENDW)
2878               LINCNT = LINCNT + 1
2879            ENDDO
2880            NN = NN - NUMMPS
2881            ENDDO
2882            NN = NN(L)
2883            IENDW = 0
2884            XNLIN = (NN / 12.) + .5
2885            NUMLIN = LINCNT + XNLIN
2886            IF (NUMLIN.GT. 45) THEN
2887               WRITE (MOT,623)
2888               LINCNT = 0
2889            ENDIF
2890

```

```

2891 DO WHILE (NN.GT. 0)
2892   ISTW = IENDW + 1
2893   IF (NN.GT. 12) THEN
2894     IENDW = IENDW + 12
2895   ELSE
2896     IENDW = IENDW + NN
2897   ENDIF
2898   NUMMPS = (IENDW - ISTW) + 1
2899   WRITE (MOT,641) ICYCLE.
2900   &
2901   FORMAT (/ ' KILL RATE MATRIX FOR RED ON DEFENSE CYCLE' ,
2902     & 14/9X,12(1X,2A4))
2903   DO JW = 1,N4
2904     WRITE (MOT,602)(AWT(JW,IWRD,II),IWRD=1,2),
2905     & (PWDKW(JW,IW,II),IW=1,ISTW,IENDW)
2906     LINCNT = LINCNT + 1
2907   ENDDO
2908   NN = NN - NUMMPS
2909   ENDDO
2910   ENDIF
2911   ENDIF
2912
2913 C INVENTORY OF PREFERRED AND NON-PREFERRED MUNITIONS
2914 C
2915 DO ISM = 1,NSM(II)
2916   DO JW = 1,N4
2917     SMENG(JW,ISM,IS,II) = 0.
2918   ENDDO
2919   ENDDO
2920   DO JW = 1,N4
2921     TEXNPM(JW,IS,II) = 0.
2922   ENDDO
2923   ISATT = ISAT(ISA(IS),IS) I FIND THE SECTOR ATTACKER W/FUNC ISAT
2924   IF (ISATT.EQ. II) THEN
2925
2926 C COMPUTE THE EXPENDITURE OF NON-PREFERRED MUNITIONS
2927 C
2928   DO IW = 1,N3
2929     DO JW = 1,N4
2930       EXNPMN(JW,IW,II) = EERTA(JW,KPA,II) * AAWA(JW,IW,II)
2931       TEXNPM(JW,IS,II) = TEXNPM(JW,IS,II) + EXNPMN(JW,IW,II)
2932     ENDDO
2933   ENDDO
2934
2935 C COMPUTE THE EXPENDITURE OF PREFERRED MUNITIONS.
2936 C
2937   IF (II.EQ. 1) THEN
2938     DO ISM = 1,NSM(II)
2939       IF (NSM(ISM,II).GT. 0 .AND. NSMFR(ISM,II).GT. 0) THEN
2940         DO INSMW = 1, NSMW(ISM,II)
2941           JW = ISMW(INSMW,ISM,II)
2942           IF (WSI(JW,IS,II).GT. 0.) THEN
2943             DO INSMFR = 1,NSMFR(ISM,II)
2944               IW = ISMFR(INSMFR,ISM,II)
2945               EXPAN(JW,IW,ISM,IS,II) = EERTA(JW,KPA,II) *
2946               & BAWASM(JW,IW,ISM)
2947             SMENG(JW,ISM,IS,II) = SMENG(JW,ISM,IS,II) +

```

```

2948      *
2949      ENDDO      I TARGETS AT WHICH TO FIRE
2950      ENDIF      I IF THERE ARE FIRERS AVAILABLE
2951      ENDDO      I POSSIBLE FIRERS
2952      ENDIF      I IF THERE ARE POSSIBLE FIRERS AND TARGETS
2953      ENDDO      I PREFERRED MUNITIONS
2954      ELSE
2955      DO ISM = 1, NSM(II)
2956      IF (NSAW(ISM,II) .GT. 0 .AND. NSMFR(ISM,II) .GT. 0) THEN
2957      DO INSAW = 1, NSAW(ISM,II)
2958      JW = ISAW(INSAW,ISM,II)
2959      IF (WSI(JW,IS,II) .GT. 0.) THEN
2960      DO INSMFR = 1, NSMFR(ISM,II)
2961      IW = ISMFR(INSMFR,ISM,II)
2962      EXPAN(JW,IW,ISM,IS,II) = EERTA(JW,KPA,II) *
2963      RAWASM(JW,IW,ISM)
2964      SMENG(JW,ISM,IS,II) = SMENG(JW,ISM,IS,II) +
2965      EXPAN(JW,IW,ISM,IS,II)
2966      *
2967      ENDDO      I TARGETS AT WHICH TO FIRE
2968      ENDIF      I IF THERE ARE FIRERS AVAILABLE
2969      ENDDO      I POSSIBLE FIRERS
2970      ENDIF      I IF THERE ARE POSSIBLE FIRERS AND TARGETS
2971      ENDDO      I PREFERRED MUNITIONS
2972      ENDIF
2973      ELSE
2974      *
2975      *
2976      *
2977      *
2978      *
2979      *
2980      *
2981      *
2982      *
2983      *
2984      *
2985      *
2986      *
2987      *
2988      *
2989      *
2990      *
2991      *
2992      *
2993      *
2994      *
2995      *
2996      *
2997      *
2998      *
2999      *
3000      *
3001      *
3002      *
3003      *
3004      *

```

```

3005      DO INSMFR = 1,NSMFR(ISM,II)
3006      IW = ISMFR(INSMFR,ISM,II)
3007      EXPAN(JW,IW,ISM,IS,II) = EERTD(JW,KPD,II) *
3008      SMENG(JW,ISM,IS,II) = SMENG(JW,ISM,IS,II) +
3009      RAWDSM(JW,IW,ISM)
3010      EXPAN(JW,IW,ISM,IS,II)
3011      ENDDO ! TARGETS AT WHICH TO FIRE
3012      ENDF ! IF THERE ARE FIRERS AVAILABLE
3013      ENDDO ! POSSIBLE FIRERS
3014      ENDF ! IF THERE ARE POSSIBLE FIRERS AND TARGETS
3015      ENDDO ! PREFERRED MUNITIONS
3016      ENDF
3017
3018
3019      C ADJUST ARTILLERY'S KILL POTENTIAL DUE TO TERRAIN ON ATTACK AND DEFENSE
3020      C ARTILLERY IS WEAPON CLASS 2
3021      C
3022      DO IW = 1,N3
3023      DO JWIC = 1,NWTIWC(2,II)
3024      JW = IWTIWC(JWIC,2,II)
3025      PWAKW(JW,IW,II) = PWAKW(JW,IW,II) * FRRINA(1,KTI)
3026      PWOKW(JW,IW,II) = PWOKW(JW,IW,II) * FRRINA(KPD,KTI)
3027      ENDDO
3028      ENDDO
3029
3030      C ADJUST HELO'S KILL POTENTIAL DUE TO TERRAIN ON ATTACK AND ON DEFENSE
3031      C HELO'S IS WEAPON CLASS 3
3032      C
3033      DO IW = 1,N3
3034      DO JWIC = 1,NWTIWC(3,II)
3035      JW = IWTIWC(JWIC,3,II)
3036      PWAKW(JW,IW,II) = PWAKW(JW,IW,II) * FRRINH(1,KTI)
3037      PWOKW(JW,IW,II) = PWOKW(JW,IW,II) * FRRINH(KPD,KTI)
3038      ENDDO
3039      ENDDO
3040
3041      1050 CONTINUE
3042      C
3043      C CHECK TO SEE IF BLUE STILL HAS SOME OF THE BLUE REFERENCE WEAPONS,
3044      C AND THAT THE BLUE REFERENCE WEAPON HAS TARGETS.
3045      C IF BLUE HAS NONE OF ITS REFERENCE WEAPON THEN
3046      C IF (BLUE IS THE DEFENDER) THEN
3047      C RED CONTINUES ATTACK
3048      C WITH FRAD = 999.
3049      C ELSE
3050      C BLUE GOES TO A HOLDING POSTURE
3051      C ENDF
3052      C ENDF
3053      C
3054      C FIND THE SECTOR ATTACKER
3055      C
3056      IF (MVCASO .NE. 3) THEN
3057      C
3058      C CHECK TO SEE IF BLUE HAS ANY OF THE REFERENCE WEAPON
3059      C
3060      IFLG = 0
3061      ICNT = 1

```

```

3062 DO WHILE (IFLG.EQ.0.AND.IWUCE(ICNT).GT.0)
3063   IWUCEE = IWUCE(ICNT)
3064   IF (WSI(IWUCEE,IS,1).EQ.0.) THEN 1 BLUE HAS NONE OF THE REF WEAPON
3065
3066 C THERE ARE NONE OF THE PRIMARY REFERENCE WEAPON PRESENT, CHECK TO SEE
3067 C IF ANOTHER WEAPON OF THE SAME GENERIC TYPE IS PRESENT.
3068 C
3069   ICNT = LNKWPT(IWUCEE,1)
3070   IF (NWTIGT(ICNT,1).GT.0) THEN
3071     NUMWEP = 1
3072     DO WHILE (IWUCEE.EQ.IWUCE(ICNT).AND.
3073              NUMWEP.LE.NWTIGT(ICNT,1))
3074       IWEP = IWTIGT(NUMWEP,ICNT,1)
3075       IF (WSI(IWEP,IS,1).GT.0.) THEN
3076         IWUCEE = IWEP
3077         IFLG = 1
3078       ENDIF
3079       NUMWEP = NUMWEP + 1
3080     ENDDO
3081   ELSE
3082     IFLG = 1
3083   ENDIF
3084   ICNT = ICNT + 1
3085   IF (IWUCE(ICNT).GT.NW(1))IFLG = 1
3086   ENDDO
3087   IWUCES(IS) = IWUCEE
3088   IF (WSI(IWUCEE,IS,1).EQ.0.) THEN
3089     IWUCES(IS) = 0
3090     WRITE(MOT,1012) IS, ICYCLE
3091     FORMAT('1',10('/),' SECTOR:',13,' CYCLE:',13,5('/).
3092     20X,'BLUE HAS NONE OF ITS REFERENCE WEAPONS')
3093     IF (ISATT.EQ.1) THEN
3094       ISA(IS) = 0
3095       FRAD(IS)=0.
3096       KPS(IS)=6
3097       WRITE(MOT,1013)
3098       FORMAT(/20X,'BLUE WAS ATTACKING AND WILL GO INTO ',
3099             'HOLDING.')
3100     ELSE
3101       IF (ISCEF(IS).EQ.0) THEN
3102         ISA(IS)=2
3103       ELSE
3104         ISA(IS) = -2
3105       ENDIF
3106       FRAD(IS)=999.
3107       WRITE(MOT,1014)
3108       FORMAT(/20X,'RED WAS ATTACKING AND WILL CONTINUE THE ',
3109             'ATTACK WITH A FRAD = 999.')

```

```

3119      DO 1550 L = 1,2
3120      I1 = 3 - L
3121      N3 = NW(L)
3122      N4 = NAM(L)
3123      N6 = NAM(I1)
3124      IF (MVCASO.EQ. 3) THEN
3125      DO IW = 1,N3
3126      DO IAM = 1,N4
3127      AAMA(IAM,IW,I1) = SSAMA(IAM,IW,I1)
3128      AAMD(IAM,IW,I1) = SSAMD(IAM,IW,I1)
3129      ENDDO
3130      DO IW = 1,N3
3131      DO IAM = 1,N6
3132      AAMA(IAM,IW,I1) = SSAMA(IAM,IW,I1)
3133      AAMD(IAM,IW,I1) = SSAMD(IAM,IW,I1)
3134      ENDDO
3135      ELSE
3136      DO 1530 IAM = 1,N4
3137      SUMA = 0.
3138      SUMD = 0.
3139      DO 1510 IW = 1,N3
3140      AAMA(IAM,IW,I1) = 0.
3141      AAMD(IAM,IW,I1) = 0.
3142      SUMA = SUMA + (SSAMA(IAM,IW,I1) * WS(IW,L))
3143      SUMD = SUMD + (SSAMD(IAM,IW,I1) * WS(IW,L))
3144      CONTINUE
3145      IF (SUMA.GT. 0.) THEN
3146      DO 1511 IW = 1,N3
3147      AAMA(IAM,IW,I1) = (SSAMA(IAM,IW,I1) * WS(IW,L)) / SUMA
3148      CONTINUE
3149      ENDIF
3150      IF (SUMD.GT. 0.) THEN
3151      DO 1512 IW = 1,N3
3152      AAMD(IAM,IW,I1) = (SSAMD(IAM,IW,I1) * WS(IW,L)) / SUMD
3153      CONTINUE
3154      ENDIF
3155      CONTINUE
3156      DO 1560 IAM = 1,N6
3157      SUMA = 0.
3158      SUMD = 0.
3159      DO 1562 IW = 1,N3
3160      AAMA(IAM,IW,I1) = 0.
3161      AAMD(IAM,IW,I1) = 0.
3162      SUMA = SUMA + SSAMA(IAM,IW,I1) * WS(IW,L)
3163      SUMD = SUMD + SSAMD(IAM,IW,I1) * WS(IW,L)
3164      CONTINUE
3165      IF (SUMA.GT. 0) THEN
3166      DO 1564 IW = 1,N3
3167      AAMA(IAM,IW,I1) = (SSAMA(IAM,IW,I1) * WS(IW,L)) / SUMA
3168      CONTINUE
3169      ENDIF
3170      CONTINUE
3171      DO 1582 IAM = 1,N3
3172      SUMA = 0.
3173      SUMD = 0.
3174      DO 1584 IW = 1,N3
3175      AAMA(IAM,IW,I1) = (SSAMA(IAM,IW,I1) * WS(IW,L)) / SUMA

```

```

3176      CONTINUE
3177      ENDIF
3178      IF (SUMD.GT. 0) THEN
3179      DO 1568 IW = 1,N3
3180      AAMD(IMM,IW,II) = (SSAAMD(IMM,IW,II) * WS(IW,L)) / SUMD
3181      CONTINUE
3182      ENDIF
3183      CONTINUE
3184      ENDIF
3185      NS = NAC(II)
3186      IMT = 2 + 6 * (II - 1)
3187      DO 1540 IW = 1,N3
3188      DO 1535 IAC = 1,N5
3189      PAAKW(IAC,IW,II) = 0.
3190      PADKW(IAC,IW,II) = 0.
3191      ALOAD = AMLFD(IAC,IS,II) * AMNLAF(IAC,1,II)
3192      DO 1534 IAM = 1,N4
3193      PAAKW(IAC,IW,II) = PAAKW(IAC,IW,II) + (ALOAD *
3194      AMNL(IAC,IAM,IMT) * AAMA(IAM,IW,II) *
3195      VAMAW(IAM,IW,II))
3196      PADKW(IAC,IW,II) = PADKW(IAC,IW,II) + (ALOAD *
3197      AMNL(IAC,IAM,IMT) * AAMD(IAM,IW,II) *
3198      VAMDW(IAM,IW,II))
3199      &
3200      CONTINUE
3201      DO 1537 ISSM = 1,NSSM(II)
3202      PMAKW(ISSM,IW,II) = 0.
3203      PMDKW(ISSM,IW,II) = 0.
3204      DO 1536 IMM = 1,N6
3205      PMAKW(ISSM,IW,II) = PMAKW(ISSM,IW,II) +
3206      (SSMNL1(ISSM,IMM,II) *
3207      AAMAA(IMM,IW,II) * VAMAW(IMM,IW,II))
3208      PMDKW(ISSM,IW,II) = PMDKW(ISSM,IW,II) +
3209      (SSMNL1(ISSM,IMM,II) *
3210      AAMD(IMM,IW,II) * VAMDW(IMM,IW,II))
3211      &
3212      CONTINUE
3213      CONTINUE
3214      CONTINUE
3215      IF (IPRB.EQ. 1) THEN
3216      WRITE(MOT,105)
3217      DO 1590 L = 1,2
3218      II = 3 - L
3219      N3 = NW(L)
3220      N4 = NAM(II)
3221      N5 = NAC(II)
3222      N6 = NAM(II)
3223      N7 = NSSM(II)
3224      WRITE(MOT,100) II,L
3225      DO 1580 IAM=1,N4
3226      WRITE(MOT,22) (AAMA(IAM,IW,II),IW=1,N3)
3227      WRITE(MOT,101) II,L
3228      DO 1581 IAM=1,N4
3229      WRITE(MOT,22) (AAMD(IAM,IW,II),IW=1,N3)
3230      WRITE(MOT,102) II,L
3231      DO 1582 IAC=1,N5
3232      WRITE(MOT,22) (PAAKW(IAC,IW,II),IW=1,N3)

```

```

3233 WRITE(MOT,103) II,L
3234 DO 1583 IAC=1,N5
3235   1583 WRITE(MOT,22) (PADKW(IAC,IW,II),IW=1,N3)
3236   WRITE(MOT,140) II,L
3237   DO 1584 IMA=1,NAM(II)
3238     1584 WRITE(MOT,22) (AAMMA(IMA,IW,II),IW=1,N3)
3239     WRITE(MOT,141) II,L
3240     DO 1585 IMA=5,NAM(II)
3241       1585 WRITE(MOT,22) (AAMMD(IMA,IW,II),IW=1,N3)
3242       WRITE(MOT,142) II,L
3243       DO 1586 ISSM=1,NSSM(II)
3244         1586 WRITE(MOT,22) (PMAKW(ISSM,IW,II),IW=1,N3)
3245         WRITE(MOT,143) II,L
3246         DO 1587 ISSM=1,NSSM(II)
3247           1587 WRITE(MOT,22) (PMDKW(ISSM,IW,II),IW=1,N3)
3248           1590 CONTINUE
3249         ENDIF
3250       C
3251     C
3252   C 20) COMPUTE THE VALUE OF AN INDIVIDUAL WEAPON AND AN INDIVIDUAL
3253   C AIRCRAFT SORTIE BY ANTI-POTENTIAL POTENTIAL.
3254   C
3255   C
3256   N3 = NW(1)
3257   N4 = NW(2)
3258   DO 2010 IW = 1,N3
3259     DO 2010 JW = 1,N4
3260       IADDR = (JW - 1) * N3 + IW
3261       BSUM(IADDR) = PMAKW(IW,JW,1)
3262       IADDR = (IW - 1) * N4 + JW
3263       RSUM(IADDR) = PMDKW(JW,IW,2)
3264     2010 CONTINUE
3265     CALL EIGENV(BSUM,RSUM,VIWASF(1,1),VIWACF(1,IS,1),VIWDCF(1,IS,2),
3266     & N3,N4,IWUCEE,MNIE,EFCE,ALAM1,IFLAG)
3267     IF (IFLAG.EQ.0) THEN
3268       DO 2015 IW = 1,N3
3269         DO 2015 JW = 1,N4
3270           IADDR = (JW - 1) * N3 + IW
3271           BSUM(IADDR) = PMDKW(IW,JW,1)
3272           IADDR = (IW - 1) * N4 + JW
3273           RSUM(IADDR) = PMAKW(JW,IW,2)
3274         2015 CONTINUE
3275         CALL EIGENV(BSUM,RSUM,VIWACF(1,IS,1),VIWDCF(1,IS,1),
3276         & VIWACF(1,IS,2),N3,N4,IWUCEE,MNIE,EFCE,ALAM2,IFLAG)
3277       ENDIF
3278       IF (IFLAG.EQ.1) THEN 1 BLUE'S REF WEAPON HAS NO TARGETS
3279       WRITE(MOT,1016) IS,ICYCLE
3280       1016 FORMAT(10('/),20X,' SECTOR: ',I3,' CYCLE: ',I3.5('/),
3281       & 20X,'BLUE'S REFERENCE WEAPON HAS NO TARGETS.')
3282       IF (ISATT.EQ.1) THEN
3283         FRAD(IS)=999.
3284         IF (ISCEF(IS).EQ.0) THEN
3285           ISA(IS)=1
3286         ELSE
3287           ISA(IS) = - 1
3288         ENDIF
3289       WRITE(MOT,1017)

```

```

3290      &      FORMAT(/20X,'BLUE WAS ATTACKING AND WILL CONTINUE THE ',
3291      ELSE      'ATTACK WITH A FRAD = 999.')
```

1 RED IS THE SECTOR ATTACKER
1 GO INTO HOLDING

```

3293      ISA(IS) = 0
3294      FRAD(IS)=0.
3295      KPS(IS)=6
3296      WRITE(MOT,1018)
3297      FORMAT(/20X,'RED WAS ATTACKING AND WILL GO INTO HOLDING.')
```

END IF
GOTO 9999

END IF

N* = NAC(1)
DO 2020 IAC = 1,N5
DO 2020 JW = 1,N4
IADDR = (JW - 1) * N5 + IAC
BSUM(IADDR) = PAAKW(IAC,JW,1)
RSUM(IADDR) = PADKW(IAC,JW,1)

2020 CONTINUE
CALL MPROD(BSUM,VIMDCF(1,IS,2),VIAACF(1,IS,1),N5,N4,1)
CALL MPROD(RSUM,VIMDCF(1,IS,2),VIADCF(1,IS,1),N5,N4,1)
DO 2025 IAC = 1,N5
VIAACF(IAC,IS,1) = ALAM1 * VIAACF(IAC,IS,1)
VIADCF(IAC,IS,1) = ALAM2 * VIADCF(IAC,IS,1)

2025 CONTINUE
N5 = NAC(2)
DO 2030 IAC = 1,N5
DO 2030 IW = 1,N3
IADDR = (IW - 1) * N5 + IAC
BSUM(IADDR) = PAAKW(IAC,IW,2)
RSUM(IADDR) = PADKW(IAC,IW,2)

2030 CONTINUE
CALL MPROD(BSUM,VIMDCF(1,IS,1),VIAACF(1,IS,2),N5,N3,1)
CALL MPROD(RSUM,VIMDCF(1,IS,1),VIADCF(1,IS,2),N5,N3,1)
DO 2035 IAC = 1,N5
VIAACF(IAC,IS,2) = ALAM2 * VIAACF(IAC,IS,2)
VIADCF(IAC,IS,2) = ALAM1 * VIADCF(IAC,IS,2)

2035 CONTINUE
N6 = NSSM(1)
DO 2040 ISSM = 1,N6
DO 2040 JW = 1,N4
IADDR = (JW - 1) * N6 + ISSM
BSUM(IADDR) = PMAKW(ISSM,JW,1)
RSUM(IADDR) = PMDKW(ISSM,JW,1)

2040 CONTINUE
CALL MPROD(BSUM,VIMDCF(1,IS,2),VIMACF(1,IS,1),N6,N4,1)
CALL MPROD(RSUM,VIMDCF(1,IS,2),VIMDCF(1,IS,1),N6,N4,1)
DO 2050 ISSM = 1,N6
VIMACF(ISSM,IS,1) = ALAM1 * VIMACF(ISSM,IS,1)
VIMDCF(ISSM,IS,1) = ALAM2 * VIMDCF(ISSM,IS,1)

2050 CONTINUE
N6 = NSSM(2)
DO 2060 ISSM = 1,N6
DO 2060 IW = 1,N3
IADDR = (IW - 1) * N6 + ISSM
BSUM(IADDR) = PMAKW(ISSM,IW,2)
RSUM(IADDR) = PMDKW(ISSM,IW,2)

2060 CONTINUE

```

3347 CALL MPROD(BSUM,VIMDCF(1,IS,1),VIMACF(1,IS,2),N6,N3,1)
3348 CALL MPROD(RSUM,VIMACF(1,IS,1),VIMDCF(1,IS,2),N6,N3,1)
3349 DO 2070 ISSM = 1,N6
3350   VIMACF(ISSM,IS,2) = ALAM2 * VIMACF(ISSM,IS,2)
3351   VIMDCF(ISSM,IS,2) = ALAM1 * VIMDCF(ISSM,IS,2)
3352 2070 CONTINUE
3353 IF (KPSMGA(15).EQ. 1) THEN
3354   DO 2090 L = 1,2
3355     NN = NW(L)
3356     IENDW = 0
3357     DO WHILE (NN.GT. 0)
3358       ISTW = IENDW + 1
3359       IF (NN.GT. 12) THEN
3360         IENDW = IENDW + 12
3361       ELSE
3362         IENDW = IENDW + NN
3363       ENDIF
3364       NUMPS = (IENDW - ISTW) + 1
3365       WRITE(MOT,106) L
3366       WRITE(MOT,146) ((AWT(IW,IWRD,L),IWRD=1,2),IW=ISTW,IENDW)
3367       FORMAT(1X,12(1X,2A4))
3368       WRITE(MOT,23) (VIMACF(IW,IS,L),IW = ISTW,IENDW)
3369       FORMAT(1X,12(1X,F8.3))
3370       NN = NN - NUMPS
3371     ENDDO
3372     NN = NW(L)
3373     IENDW = 0
3374     DO WHILE (NN.GT. 0)
3375       ISTW = IENDW + 1
3376       IF (NN.GT. 12) THEN
3377         IENDW = IENDW + 12
3378       ELSE
3379         IENDW = IENDW + NN
3380       ENDIF
3381       NUMPS = (IENDW - ISTW) + 1
3382       WRITE(MOT,107) L
3383       WRITE(MOT,146) ((AWT(IW,IWRD,L),IWRD=1,2),IW=ISTW,IENDW)
3384       WRITE(MOT,23) (VIMDCF(IW,IS,L),IW = ISTW,IENDW)
3385       NN = NN - NUMPS
3386     ENDDO
3387     N3 = NAC(L)
3388     WRITE(MOT,108) L
3389     WRITE(MOT,22) (VIMACF(IAC,IS,L),IAC = 1,N3)
3390     WRITE(MOT,109) L
3391     WRITE(MOT,22) (VIMDCF(IAC,IS,L),IAC = 1,N3)
3392     WRITE(MOT,144) L
3393     WRITE(MOT,22) (VIMACF(ISSM,IS,L),ISSM = 1,NSSM(L))
3394     WRITE(MOT,145) L
3395     WRITE(MOT,22) (VIMDCF(ISSM,IS,L),ISSM = 1,NSSM(L))
3396 2090 CONTINUE
3397   ENDIF
3398 C
3399 C 25) CALCULATE TOE AND CURRENT WEAPON VALUES, EFFECTIVENESS OF DIVISION
3400 C IN ACTIVE BATTLE AREAS, AND GROUND VALUES ON ATTACK AND DEFENSE
3401 C
3402 C
3403 C

```

```

3404      2500 N2 = 0
3405      KPI = KPS(IS)
3406      DO 2530 L = 1,2
3407      IF (NDS(IS,L) .GT. 0) THEN
3408      N1 = N2 + 1
3409      N2 = N2 + NDS(IS,L)
3410      N3 = NW(L)
3411      DO 2525 IDS = N1,N2
3412      ID = IDLABA(IDS,IS)
3413      IT = ITD(ID)
3414
3415      C
3416      C MAKE SURE NO HIGHER ECHELON UNITS ARE ON THE FEBA
3417      C
3418      IF (IT .EQ. -1) GOTO 2525
3419      IC = ICBIT(ID)
3420      WVDAC(ID) = 0.
3421      WDDC(ID) = 0.
3422      WVDACI = 0.
3423      WDDCI = 0.
3424      WVDAT = 0.
3425      WDDT = 0.
3426
3427      C
3428      C COMPUTE DISTANCE UNIT IS FROM THE FEBA
3429      C
3430      DIST = ABS(FEBA(IS) - CDVLOC(ID))
3431      FRACW = (DFBACT(L) - DIST)/DFBACT(L)
3432
3433      C
3434      C COMPUTE WEAPONS VALUES FOR DIVISION ID. WVDAC, WDDC ARE THE WEAPONS VALUES
3435      C FOR THE DIVISION'S WEAPONS ON-HAND WITHOUT ADJUSTING FOR TERRAIN RESTRICTIONS.
3436      C WVDACI, WDDCI ARE THE VALUES FOR ON-HAND WEAPONS ADJUSTING FOR TERRAIN
3437      C RESTRICTIONS.
3438      C
3439      DO 2505 IWIC = 1,NWTWC(1,L)  ! NUMBER OF WEAPON TYPES IN CLASS 1
3440      IW = IWTWC(IWIC,1,L)
3441      WVDAC(ID) = WVDAC(ID) + VIWACF(IW,IS,L) *
3442      (WDIV(IW,ID) * FRACW)
3443      WDDC(ID) = WDDC(ID) + VIWDCF(IW,IS,L) *
3444      (WDIV(IW,ID) * FRACW)
3445      WVDAT = WVDAT + VIWACF(IW,IS,L) * BWD(IW,ID)
3446      WDDT = WDDT + VIWDCF(IW,IS,L) * BWD(IW,ID)
3447
3448      C
3449      C TERRAIN ADJUSTMENT:
3450      C
3451      C VIWACF AND VIWDCF CONTAIN THE VALUES OF INDIVIDUAL WEAPONS USING THE NEW
3452      C MIX OF WEAPONS DUE TO TERRAIN RESTRICTIONS. THESE VALUES ESSENTIALLY
3453      C ADJUST THE NUMBER OF TARGETS IN THE RESTRICTED TERRAIN. NOW THE NUMBER OF
3454      C FIRES MUST BE ADJUSTED. THE MAXIMUM NUMBER OF FIRERS THAT CAN FIT IS
3455      C OBTAINED BY MULTIPLYING THE TERRAIN ADJUSTMENT FACTOR BY THE TOE NUMBER OF
3456      C WEAPONS. THE NUMBER OF FIRERS IS FOUND BY TAKING THE MINIMUM OF THIS NUMBER
3457      C AND THE NUMBER OF WEAPONS ON-HAND.
3458      C
3459      WTFITA = FRRING(IT,1,KTI) * TWD(IW,IT)
3460      WTFITA = (AMIN1(WTFITA,WDIV(IW,ID)) * FRACW)
3461      WTFITD = FRRING(IT,KPI,KTI) * TWD(IW,IT)
3462      WTFITD = (AMIN1(WDIV(IW,ID),WTFITD) * FRACW)
3463      WVDACI = WVDACI + VIWACF(IW,IS,L) * WTFITA
3464      WDDCI = WDDCI + VIWDCF(IW,IS,L) * WTFITD

```

```

3461      2505      CONTINUE
3462      C
3463      C ADD IN REST OF WEAPON TYPES (NOT AFFECTED BY TERRAIN)
3464      C
3465      DO 2506 IWPC = 2, LNWP
3466      IF (NWTIWC(IWPC,L) .NE. 0) THEN
3467      DO IWIC = 1, NWTIWC(IWPC,L)
3468      IW = IWTIWC(IWIC,IWPC,L)
3469      WVDAT = WVDAT + VIWACF(IW,IS,L) * BWD(IW,ID)
3470      WVDOT = WVDOT + VIWDCF(IW,IS,L) * BWD(IW,ID)
3471      IF (ERGMAX(IW,L) .EQ. 0.) THEN
3472      WVDAC(ID) = WVDAC(ID) + VIWACF(IW,IS,L) *
3473      WVDAC(ID)
3474      WVDAC(ID) = WVDAC(ID) + VIWDCF(IW,IS,L) *
3475      WVDAC(ID)
3476      WVDACI = WVDACI + VIWACF(IW,IS,L) *
3477      WVDACI
3478      WVDDCI = WVDDCI + VIWDCF(IW,IS,L) *
3479      WVDDCI
3480      ELSE IF (DIST .LE. ERGMAX(IW,L)) THEN
3481      WVDAC(ID) = WVDAC(ID) + VIWACF(IW,IS,L) *
3482      WVDAC(ID)
3483      WVDAC(ID) = WVDAC(ID) + VIWDCF(IW,IS,L) *
3484      WVDAC(ID)
3485      WVDACI = WVDACI + VIWACF(IW,IS,L) *
3486      WVDACI
3487      WVDDCI = WVDDCI + VIWDCF(IW,IS,L) *
3488      WVDDCI
3489      ENDIF
3490      ENDDO
3491      ENDIF
3492      CONTINUE
3493      C
3494      C CALCULATE THE EFFECTIVENESS OF PERSONNEL AND WEAPONS.
3495      C
3496      PPS = PDIV(ID) / BPD(ID)
3497      CURINP = PPS
3498      CALL CVFW(NEAEF(IC),XAEF(1,IC),YAEF(1,IC))
3499      PEA = CURVES
3500      PEA = PEA * TNGMOD(ID)
3501      CURINP = PPS
3502      CALL CVFW(NEDEF(IC),XDEF(1,IC),YDEF(1,IC))
3503      PED = CURVES
3504      PED = PED * TNGMOD(ID)
3505      WATS = 0.
3506      WDS = 0.
3507      IF (WVDAT .GT. 0.) WATS = WVDAC(ID) / WVDAT
3508      IF (WVDOT .GT. 0.) WDS = WVDDC(ID) / WVDOT
3509      CURINP = WATS
3510      CALL CVFW(NWAEFF(IC),XWAEF(1,IC),YWAEF(1,IC))
3511      WEA = CURVES
3512      WEA = WEA * TNGMOD(ID)
3513      CURINP = WDS
3514      CALL CVFW(NWDEFF(IC),XWDEF(1,IC),YWDEF(1,IC))
3515      WED = CURVES
3516      WED = WED * TNGMOD(ID)
3517      EDAXSR(ID) = AMIN1(WEA,PEA)

```

```

3518      EDDXSR(ID) = AMIN1(WED,PED)
3519
3520      C
3521      C CALCULATE EFFECTIVENESS REDUCTION DUE TO SUPPLY SHORTAGE.
3522      C
3523      TEMP1 = 0.
3524      TEMP2 = 0.
3525      IF (ISEFFD .LE. 2) THEN
3526          SEF = 0.
3527
3528      C SUPPLY CONSUMPTION RATES ARE BASED ON TOE SIZE UNITS. TO CONVERT THESE RATES
3529      C TO A PARTICULAR SIZE UNIT, THE CONSUMPTION RATE IS MULTIPLIED BY THE
3530      C BASE TO TOE PERSONNEL RATIO. THIS METHOD TREATS SUPPLY CONSUMPTION AS
3531      C TONS PER MAN PER DAY. A BETTER REPRESENTATION WOULD TO USE A WEAPON
3532      C RATIO FOR SUPPLY TYPES BASED ON WEAPONS AND A PERSONNEL RATIO FOR
3533      C PERSONNEL SUPPLIES. TO DO THIS A WEIGHTING SCHEME WOULD HAVE TO BE
3534      C ESTABLISHED TO RELATE SUPPLY CONSUMPTION BY SUPPLY TYPE TO WEAPON TYPE.
3535      C
3536      BITPRAT = BPD(ID) / TPD(IT)
3537      DO 2510 ISPT = 1, NSPT
3538          TEMP1 = TEMP1 + SDIV(ID, ISPT)
3539          APCSD = PCSD(1, IT, ISPT) * BITPRAT ! ADJUSTED CONSUMPTION RATE
3540          TEMP2 = TEMP2 + APCSD
3541          DSH = SDIV(ID, ISPT) / APCSD
3542          CURINP = DSH
3543          CALL CVFW(NSEFF(L), XSEFF(1, ISPT, L), YSEFF(1, ISPT, L))
3544          TEMP = CURVES
3545          SEF = SEF + FDWST(IT, ISPT) * TEMP
3546      CONTINUE
3547      IF (ISEFFD .EQ. 1) THEN
3548          EFFDA(ID) = SEF * EDAXSR(ID)
3549          EFFDD(ID) = SEF * EDDXSR(ID)
3550      ELSE 1 ISEFFD = 2
3551          EFFDA(ID) = AMIN1(SEF, WEA, PEA)
3552          EFFDD(ID) = AMIN1(SEF, WED, PED)
3553      ENDIF
3554      ELSE 1 ISEFFD = 3
3555          SEF = 1.
3556          EFFDA(ID) = AMIN1(WEA, PEA)
3557          EFFDD(ID) = AMIN1(WED, PED)
3558      ENDIF
3559      IF (IOMJ .NE. 3 .AND. IOMJ .NE. 5) THEN IDEGRADE FOR CHEM USE
3560          EFFDA(ID) = EFFDA(ID) * FDEFCP(ID)
3561          EFFDD(ID) = EFFDD(ID) * FDEFCP(ID)
3562      ENDIF
3563      PPSAR(ID) = PPS
3564      PEAAR(ID) = PEA
3565      PEDAR(ID) = PED
3566      WATSAR(ID) = WATS
3567      WEAAR(ID) = WEA
3568      WEDAR(ID) = WED
3569      DSHAR(ID) = 0.
3570      IF (TEMP2 .NE. 0) DSHAR(ID) = TEMP1 / TEMP2
3571      SEFAR(ID) = SEF
3572      VDAC(ID) = EFFDA(ID) * WDAT
3573      VDDC(ID) = EFFDD(ID) * WDDT
3574      VGABA(1S, L) = VGABA(1S, L) + VDAC(ID)

```

2510

```

3575      VGDBA(IS,L) = VGDBA(IS,L) + VDDC(ID)
3576
3577 C COMPUTE THE VALUE OF THE GROUND FORCES IN THE RESTRICTED TERRAIN.
3578 C FFF IS THE FRACTIONAL REDUCTION OF SIDE L'S GROUND FORCES DUE TO TERRAIN
3579 C RESTRICTIONS.
3580 C
3581      FFF = 1.0
3582      IF (WVDAC(ID) .GE. .0001) FFF = WVDAC1 / WVDAC(ID)
3583      VGABAI(IS,L) = VGABAI(IS,L) + (FFF * VDAC(ID))
3584      FFF = 1.0
3585      IF (WVDDC(ID) .GE. .0001) FFF = WVDDC1 / WVDDC(ID)
3586      VGDBAI(IS,L) = VGDBAI(IS,L) + (FFF * VDDC(ID))
3587
3588 C
3589      IF (KPSMGA(15) .EQ. 1) THEN
3590        WRITE(MOT,110)
3591        WRITE(MOT,10) ID,IT
3592        WRITE(MOT,111)
3593        WRITE(MOT,22) WVDAC(ID),WVDDC(ID),WVDAC1, WVDDC1,
3594          & WVDAT, WVDDT, EDAXSR(ID),
3595          & EDXXSR(ID), EFFDA(ID), EFFDD(ID), VDAC(ID),
3596          & VDDC(ID)
3597        WRITE(MOT,112)
3598        WRITE(MOT,22) PEA,PED,DSH,SEF
3599      ENDIF
3600
3601      2525 CONTINUE
3602      ENDIF
3603      IF (KPSMGA(15) .EQ. 1) THEN
3604        WRITE(MOT,113)
3605        WRITE(MOT,22) (VGABA(IS,L),VGDBA(IS,L),L = 1,2)
3606      ENDIF
3607
3608 C
3609 C 30) CALCULATE AIR VALUES ON ATTACK AND DEFENSE
3610 C
3611      DO 3020 L = 1,2
3612        VAABA(IS,L) = 0.
3613        VADBA(IS,L) = 0.
3614        N3 = NAC(L)
3615        DO 3010 IAC = 1,N3
3616          VAABA(IS,L) = VAABA(IS,L)+ACSABA(IAC,IS,L)*VIAACF(IAC,IS,L)
3617          VADBA(IS,L) = VADBA(IS,L)+ACSABA(IAC,IS,L)*VIADCF(IAC,IS,L)
3618        CONTINUE
3619      IF (IPRB .EQ. 1) THEN
3620        WRITE(MOT,9000) IS,L,(ACSABA(IAC,IS,L),IAC = 1,N3)
3621        FORMAT(' ACSABA--SECTOR ',15,' SIDE ',12,7(2X,F10.2))
3622      ENDIF
3623 CONTINUE
3624
3625 C 31) CALCULATE SSM VALUES ON ATTACK AND DEFENSE
3626 C
3627      DO 3120 L = 1,2
3628        VMABA(IS,L) = 0.
3629        VMDBA(IS,L) = 0.
3630        N6 = NSSM(L)
3631        DO 3110 ISSM = 1,N6
3632          VMABA(IS,L) = VMABA(IS,L)+SSMALO(ISSM,IS,L)*VIMACF(ISSM,IS,L)
3633          VMDBA(IS,L) = VMDBA(IS,L)+SSMALO(ISSM,IS,L)*VIMDCF(ISSM,IS,L)

```

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47

VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR:9

Page 39

```

3632 3110 CONTINUE
3633 3120 CONTINUE
3634 C
3635 IF (IPRB.EQ.1) THEN
3636 WRITE(MOT,114)
3637 WRITE(MOT,22) (VAABA(IS,L),VADBA(IS,L),L=1,2)
3638 WRITE(MOT,135)
3639 WRITE(MOT,22)(VMABA(IS,L),VMDBA(IS,L),L=1,2)
3640 ENDIF
3641 C
3642 C 35) COMPUTE THE SECTOR ATTACKER IF THERE IS AN ATTACK IN SECTOR IS.
3643 C
3644 KP = KPS(IS)
3645 ISATT = ISAT(ISA(IS),IS)
3646 II = 3 - ISATT
3647 IF (ICSA(IS).EQ.0) THEN
3648 IF (VGDBAI(IS,II) + VADBA(IS,II) + VMDBA(IS,II).LE.0.0) THEN
3649 IF (ISCEF(IS).EQ.0) THEN
3650 ISA(IS) = ISATT I SIDE II IS DEFENSELESS
3651 ELSE
3652 ISA(IS) = -ISATT I SECTOR ATTACKER IS FLANK CONSTRAINED.
3653 ENDIF
3654 ELSE
3655 IF (VGDBAI(IS,ISATT) + VADBA(IS,ISATT) +
3656 VMDBA(IS,ISATT).LE.0.0) THEN
3657 ISA(IS) = II
3658 I SECTOR ATTACKER IS DEFENSELESS SO SECTOR ATKR II
3659 ELSE
3660 FRAD(IS) = (VGABAI(IS,ISATT) + VAABA(IS,ISATT) +
3661 VMABA(IS,ISATT)) /
3662 (VGDBAI(IS,II) + VADBA(IS,II) + VMDBA(IS,II))
3663 IF (FRAD(IS).GT.999.9) FRAD(IS) = 999.9
3664 IFLAG=1
3665 IF (FRAD(IS).GE.FRATA(KP,ISATT).AND.ISCEF(IS).EQ.0) THEN
3666 C
3667 C ISATT ATTACKER HAS ENOUGH FORCE RATIO TO ATTACK AND IS NOT FLANK CONS
3668 C
3669 ISA(IS) = ISATT
3670 ELSE
3671 C
3672 C SECTOR ATTACKER DOES NOT HAVE ENOUGH COMBAT POWER OR IS FLANK CONSTRAI
3673 C DOES SECTOR DEFENDER HAVE ENOUGH POWER TO TAKE THE ATTACK
3674 C
3675 IF (FRAD(IS).GE.FRATA(KP,ISATT).AND.
3676 ISCEF(IS).NE.0) IFLAG = 2
3677 TMPFRD = (VGABAI(IS,II) + VAABA(IS,II) + VMABA(IS,II)) /
3678 (VGDBAI(IS,ISATT) + VADBA(IS,ISATT) +
3679 VMDBA(IS,ISATT))
3680 IF (TMPFRD.GT.999.9) TMPFRD = 999.9
3681 IF (TMPFRD.GE.FRATD(KP,II)) THEN
3682 C
3683 C THEATER DEFENDER HAS ENOUGH COMBAT POWER TO TAKE THE ATTACK AND BECOME
3684 C ATTACKER.
3685 C
3686 ISA(IS) = II
3687 FRAD(IS) = TMPFRD
3688 ELSE

```

```

3689      IF (IFLAG.EQ. 1) THEN
3690      ISA(IS) = 0
3691      C      I NEITHER SIDE HAS ENOUGH COMBAT POWER TO AT
3692      ELSE
3693      ISA(IS) = - ISATT
3694      C      I ISATT CAN ATTACK BUT IS FLANK RESTR
3695      ENDIF
3696      ENDIF
3697      ENDIF
3698      ENDIF
3699      END IF
3700      ELSE
3701      IF (VGDBAI(IS,ISATT) + VADBA(IS,ISATT) +
3702      VADBA(IS,ISATT).LE.0.0) THEN
3703      &
3704      ISA(IS) = 11
3705      ICSA(IS) = 0
3706      C      I SECTOR ATTACKER IS DEFENSELESS SO SECTOR ATKR II
3707      ELSE
3708      FRAD(IS) = (VCABAI(IS,ISATT) + VAABA(IS,ISATT) +
3709      VADBA(IS,ISATT)) /
3710      (VGDBAI(IS,11) + VADBA(IS,11) + VADBA(IS,11))
3711      IF (FRAD(IS).GT. 999.9) FRAD(IS) = 999.9
3712      IF (FRAD(IS).GE. FRATA(KP,ISATT)) THEN
3713      C
3714      C ISATT ATTACKER HAS ENOUGH FORCE RATIO TO ATTACK
3715      C
3716      ISA(IS) = ISATT
3717      ELSE
3718      C
3719      C SECTOR ATTACKER DOES NOT HAVE ENOUGH COMBAT POWER
3720      C DOES SECTOR DEFENDER HAVE ENOUGH POWER TO TAKE THE ATTACK
3721      C
3722      TMPFRD = (VCABAI(IS,11) + VAABA(IS,11) + VAABA(IS,11)) /
3723      (VGDBAI(IS,ISATT) + VADBA(IS,ISATT) +
3724      VADBA(IS,ISATT))
3725      IF (TMPFRD.GT. 999.9) TMPFRD = 999.9
3726      IF (TMPFRD.GE. FRATD(KP,11)) THEN
3727      C
3728      C THEATER DEFENDER HAS ENOUGH COMBAT POWER TO TAKE THE ATTACK AND BECOME
3729      C ATTACKER.
3730      C
3731      ISA(IS) = 11
3732      FRAD(IS) = TMPFRD
3733      ELSE
3734      ISA(IS) = 0
3735      END IF
3736      ICSA(IS) = 0
3737      END IF
3738      ENDIF
3739      C
3740      C SET ISA = -3 IF THE SECTOR ATTACKER IS BLUE BUT THEY HAVE REACHED THE INITIAL FEBA
3741      C AND THE FLAG "KSTOPB" IS SET SO THEY CAN'T GO FURTHER.
3742      C
3743      IF (KSTOPB.EQ. 1 .AND. ISA(IS).EQ. 1 .AND.
3744      FEBATZ(IS).LE. FEBATZ(IS)) THEN
3745      &
3746      ISA(IS) = -3

```

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47

VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR:9

```

3746       ICSA(IS) = 0
3747       END IF
3748
3749       C SET ISA = -4 IF SIDE 1 IS THE SECTOR ATTACKER BUT THE I HOLD FLAG IS SET
3750       C SET ISA = -5 IF SIDE 2 IS THE SECTOR ATTACKER BUT THE I HOLD FLAG IS SET
3751       C
3752       IF (ISA(IS) .EQ. 1 .AND. I HOLD(1) .EQ. 1) ISA(IS) = -4
3753       IF (ISA(IS) .EQ. 2 .AND. I HOLD(2) .EQ. 1) ISA(IS) = -5
3754
3755       C SAVE THE LAST SECTOR ATTACKER
3756       C
3757       ISATT = ISAT(ISA(IS), IS)
3758       IF (ISATT .NE. ISAP(IS)) THEN
3759         ISAP(IS) = ISATT
3760       ENDIF
3761
3762       C SET ISA = -6 IF THERE IS A CHANGE IN COMBAT MODE FOR EITHER SIDE
3763       C
3764       IF (ICMST(IS,1) .GT. 0 .OR. ICMST(IS,2) .GT. 0) ISA(IS) = -6
3765       C
3766       IF (IPRB .EQ. 1) THEN
3767         WRITE(MOT,115)
3768         WRITE(MOT,22) FRAD(IS)
3769         WRITE(MOT,132)
3770         WRITE(MOT,10) ISA(IS)
3771       ENDIF
3772
3773       C
3774       C 40) CALCULATE EFFECTIVENESS REDUCTION DUE TO SUPPLY SHORTAGE IN SECTOR
3775       C
3776       PPSE = -999.
3777       N2 = 0
3778       DO 4020 L = 1,2
3779         IF (NDS(IS,L) .GT. 0) THEN
3780           N1 = N2+1
3781           N2 = NDS(IS,L) + N2
3782           SEF = 0.
3783           DO 4010 IDS = N1,N2
3784             ID = IDLABA(IDS, IS)
3785           C MAKE SURE NO HIGHER ECHELON UNITS ARE ON THE FEBA/ABA
3786           C
3787           IF (ITD(ID) .EQ. -1) GOTO 4010
3788           SEF = SEF + SEFAR(ID)
3789           CONTINUE
3790           SEF = SEF / NDS(IS,L)
3791         C
3792         C PPSE IS USED TO SCALE CASUALTIES BASED ON THE RATIO OF THE SIDE WITH THE
3793         C GREATEST SEF AND THAT SIDE'S 100% SEF.
3794         C
3795         PPSE = AMAX1(PPSE, SEF)
3796         PPESAR(IS,L) = PPSE
3797       ENDIF
3798       4020 CONTINUE
3799       C
3800       IF (IPRB .EQ. 1) THEN
3801         WRITE(MOT,116)
3802         WRITE(MOT,22) PPSE, DSH

```

```

3803      ENDIF
3804
3805      C 45) COMPUTE VALUE LOST FOR BOTH SIDES.
3806
3807      C
3808      C ADJA, ADJD ARE ADJUSTMENT FACTORS USED TO ADJUST THE CASUALTY PERCENTAGES
3809      C BASED ON THE GROUND FORCES VALUE REDUCTIONS FOR FAIR AND POOR TERRAIN.
3810
3811      C VDI, VAI ARE REDUCTION FACTORS FOR GROUND FORCES ON FAIR AND POOR
3812      C TERRAIN IN RELATION TO IF THEY WERE ON GOOD TERRAIN.
3813
3814      FRCA = 999.9
3815      FRCD = 999.9
3816      ADJA = 1.0
3817      ADJD = 1.0
3818      VDI = 0.
3819      VAI = 0.
3820      L = ISATT
3821      II = 3 - L
3822      TEMP = VGDBAI(1S,11) + VADBA(1S,11) + VMDBA(1S,11)
3823      IF (TEMP .GT. .0001) THEN
3824          VAFPD = VADBA(1S,11) / TEMP 1 FRACTION DUE TO AIR FOR DEFENDER
3825          VMFPD = VMDBA(1S,11) / TEMP 1 FRACTION DUE TO SSM'S FOR DEFENDER
3826          IF (VGDBA(1S,11) .GT. .0001) VDI = VGDBAI(1S,11) /
3827          &
3828          FRCA = (VGABAI(1S,L) + FASFRCA * VAABA(1S,L) +
3829          & VMABA(1S,L)) / TEMP
3830          TEMP1 = VGDBAI(1S,11) + (FDSFRCA * VADBA(1S,11)) + VMDBA(1S,11)
3831          IF (TEMP1 .GT. .0001) THEN
3832              FRCD = (VGABAI(1S,L) + VAABA(1S,L) + VMABA(1S,L)) / TEMP1
3833              IF (FRCD .GT. .0001) THEN
3834                  VAFPA = VAABA(1S,L) / (FRCD * TEMP1)
3835                  VMFPA = VMABA(1S,L) / (FRCD * TEMP1)
3836                  IF (VGABA(1S,L) .GT. .0001)
3837                      & VAI = VGABAI(1S,L) / VGABA(1S,L)
3838                      ADJD = VAFPA + VMFPA + VDI * (1-(VAFPA+VMFPA))
3839
3840          C
3841          C VALUE DUE TO GROUND ATTACKER
3842          C VDI * VALUE DUE TO ATTACKER'S GROUND TAKES INTO ACCOUNT THE REDUCED NUMBER
3843          C OF TARGETS THAT SIDE L'S GROUND WEAPONS HAVE TO SHOOT AT IN FAIR OR POOR GND
3844
3845      ENDIF
3846      ADJA = VAFPD + VMFPD + VAI * (1-(VAFPD+VMFPD))
3847      ENDIF
3848      IF (ISA(1S) .EQ. 0) THEN
3849          KPCAS = 6
3850      ELSE
3851          KPCAS = KPS(1S)
3852      ENDIF
3853      N4 = NPCAF(L)
3854      N5 = NPCDF(11)
3855      DO 4511 IPCAF = 1,N4
3856          TYPCAF(IPCAF) = YPCAF(KPCAS,IPCAF,L)
3857      CONTINUE
3858      DO 4512 IPCDF = 1,N5
3859          TYPCDF(IPCDF) = YPCDF(KPCAS,IPCDF,11)
3860      CONTINUE
4511
4512

```

```

3860 C
3861 C GET THE CASUALTY PERCENTAGE FROM THE FORCE RATIO AND COMPUTE VALUE LOST FOR
3862 C THE ATTACKER.
3863 C
3864 CURINP = FRCA
3865 CALL CVFW(NPCAF(L),XPCAF(1,L),TYPCAF)
3866 PCFR = CURVES
3867 PCS(L) = PCFR * PPESE * ADJA
3868 VLS(L) = PCS(L) * VGABA(1S,L) IVALUE LOST DUE TO CASUALTIES
3869 VLS(L) = FCVLS(L) * VLS(L)
3870 C
3871 C COMPUTE CASUALTIES AND VALUE LOST FOR THE DEFENDER
3872 C
3873 CURINP = FRCD
3874 CALL CVFW(NPCDF(11),XPCDF(1,11),TYPCDF)
3875 PCFR = CURVES
3876 PCS(11) = PCFR * PPESE * ADJD
3877 VLS(11) = PCS(11) * VGDBA(1S,11)
3878 VLS(11) = FCVLS(11) * VLS(11)
3879 FRCAR(1S,L) = FRCA
3880 FRCAR(1S,11) = FRCD
3881 PCSAR(1S,L) = PCS(L)
3882 PCSAR(1S,11) = PCS(11)
3883 VLSAR(1S,L) = VLS(L)
3884 VLSAR(1S,11) = VLS(11)
3885 C
3886 IF(IPRB.EQ.1) THEN
3887 WRITE(MOT,117)
3888 WRITE(MOT,22) FRCA,FRCD,PCFR
3889 WRITE(MOT,118)
3890 WRITE(MOT,22) (VLS(L),L = 1,2),(PCS(L),L = 1,2)
3891 ENDIF
3892 C
3893 C 54) COMPUTE TONS OF SUPPLIES CONSUMED BY BOTH SIDES AND NEW SUPPLY
3894 C INVENTORIES.
3895 C
3896 N2 = 0
3897 DO 5450 L = 1,2
3898 N1 = N2 + 1
3899 N2 = N2 + NDS(1S,L)
3900 DO 5405 ISPT = 1,NSPT
3901 TSC(1SPT,L) = 0.
3902 CONTINUE
3903 IF(NDS(1S,L).GT.0) THEN
3904 DO 5430 IDS = N1,N2
3905 ID = IDLABA(IDS,1S)
3906 IC = ICBIT(ID)
3907 IT = ITD(ID)
3908 C
3909 C MAKE SURE NO HIGHER ECHELON UNITS ARE PROCESSED
3910 C
3911 IF (IT.EQ.-1) GOTO 5430
3912 C
3913 C AS BEFORE CONSUMPTION RATES MUST BE ADJUSTED, USE BASE TO TOE PERSONNEL
3914 C RATIO AGAIN.
3915 C
3916 BTPRAT = BPD(ID) / TPD(IT)

```

```

3917 TEMP = PDIV(ID) / BPD(ID)
3918 IF (ISATT.NE.L) THEN
3919   TEMP2 = EDOXSR(ID)
3920 ELSE
3921   TEMP2 = EDAXSR(ID)
3922 ENDIF
3923 DO 5410 ISPT = 1, NSPT
3924   IF (ISA(IS) .NE. L) THEN
3925     TEMP3 = CSDDP(IT, KPCAS, ISPT)
3926   ELSE
3927     TEMP3 = CSDAP(IT, KPCAS, ISPT)
3928   ENDIF
3929   TEMP1 = AMAX1(TEMP2 * (TEMP3 * BTPRAT),
3930     &      TEMP * (CSDR(1, IT, ISPT) * BTPRAT))
3931   TEMP1 = AMIN1(TEMP1, SDIV(ID, ISPT))
3932   SDIV(ID, ISPT) = SDIV(ID, ISPT) - TEMP1
3933   TSC(ISPT, L) = TSC(ISPT, L) + TEMP1
3934   CTSCD(1, ISPT, IC) = CTSCD(1, ISPT, IC) + TEMP1
3935 5410 CONTINUE
3936 5430 CONTINUE
3937 ENDIF
3938 CONTINUE
3939 IF (IPRB.EQ.1) THEN
3940   WRITE(MOT, 128)
3941   WRITE(MOT, 22) ((TSC(ISPT, L), ISPT = 1, NSPT), L = 1, 2)
3942 ENDIF
3943
C
C 60) CALCULATE WEAPONS LOST BY TYPE.
C
DO 6060 L = 1, 2
  11 = 3 - L
  N2 = NW(L)
  N3 = NW(L)
  N4 = NW(11)
  N5 = NAC(11)
  N6 = NSSM(11)
  TEMP = 0.
  C ++++++
  C MVCASO = 0 USE APP METHOD
  C ++++++
  IF (MVCASO.NE.1) THEN
    IF (ISATT.NE.L) THEN
      DO 6010 IW = 1, N2
        SUMMW(IW) = 0.
        DO 6005 KW = 1, N4
          SUMMW(IW) = SUMMW(IW) + WSI(KW, IS, 11) * PWAKW(KW, IW, 11)
        CONTINUE
        DO 6006 IAC = 1, N5
          SUMMW(IW) = SUMMW(IW) + ACSABA(IAC, IS, 11) *
            &      PAKW(IAC, IW, 11)
        CONTINUE
        DO 6007 ISSM = 1, N6
          SUMMW(IW) = SUMMW(IW) + (SSMALO(ISSM, IS, 11) *
            &      PMAKW(ISSM, IW, 11))
        CONTINUE
        TEMP = TEMP + SUMMW(IW) * VIWDCF(IW, IS, L)
      CONTINUE
    6010
  6060
  &
  6006
  &
  6007
  6010

```

```

3974 IF (ISATT .NE. 0) THEN
3975   SFAPP(L) = SFAPP(L) + (VLS(L)/TEMP)
3976   NUMINC(L) = NUMINC(L) + 1
3977 ENDIF
3978 VFACTOR = VLS(L)/TEMP
3979 DO 6020 IW = 1,N3
3980   WLS(IW,IS,L) = 0.0
3981   DGRD(IW) = 0.0
3982   DO 6015 KW = 1,N4
3983     GKGS(KW,IW,II) = (VLS(L) * WSI(KW,IS,II) *
3984       PPAKW(KW,IW,II))/TEMP
3985   DGRD(IW) = DGRD(IW) + GKGS(KW,IW,II)
3986   CONTINUE
3987   DAIR(IW) = 0.0
3988   DO 6016 IAC = 1,N5
3989     AKGS(IAC,IW,II) = (VLS(L) * ACSABA(IAC,IS,II) *
3990       PPAKW(IAC,IW,II)) / TEMP
3991   CONTINUE
3992   &
3993   C
3994   C
3995   C
3996   AKGS(IAC,IW,II) = AKGS(IAC,IW,II) * FICAS(2,II)
3997   DAIR(IW) = DAIR(IW) + AKGS(IAC,IW,II)
3998   CONTINUE
3999   DSSM(IW) = 0.0
4000   DO 6017 ISSM = 1,N6
4001     XMKGS(ISSM,IW,II) = (VLS(L) * SSMALO(ISSM,IS,II) *
4002       PPAKW(ISSM,IW,II)) / TEMP
4003   CONTINUE
4004   DSSM(IW) = DSSM(IW) + XMKGS(ISSM,IW,II)
4005   CONTINUE
4006   WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4007   CONTINUE
4008   ELSE
4009     DO 6035 IW = 1,N2
4010       SUMAW(IW) = 0.
4011       DO 6030 KW = 1,N4
4012         SUMAW(IW) = SUMAW(IW) + WSI(KW,IS,II) * PWDKW(KW,IW,II)
4013       CONTINUE
4014       DO 6031 IAC = 1,N5
4015         SUMAW(IW) = SUMAW(IW) + ACSABA(IAC,IS,II) *
4016           PADKW(IAC,IW,II)
4017       CONTINUE
4018       DO 6032 ISSM = 1,N6
4019         SUMAW(IW) = SUMAW(IW) + (SSMALO(ISSM,IS,II) *
4020           PWDKW(ISSM,IW,II))
4021       CONTINUE
4022       TEMP = TEMP + SUMAW(IW) * VIWACF(IW,IS,L)
4023       CONTINUE
4024       SFAPP(L) = SFAPP(L) + (VLS(L)/TEMP)
4025       NUMINC(L) = NUMINC(L) + 1
4026       VFACTOR = VLS(L) / TEMP
4027       DO 6044 IW = 1,N3
4028         WLS(IW,IS,L) = 0.0
4029         DGRD(IW) = 0.0
4030         DO 6040 KW = 1,N4
4031           GKGS(KW,IW,II) = (VLS(L) * WSI(KW,IS,II) *

```

```

4031      &
4032      DGRD(IW) = DGRD(IW) + GKGS(KW,IW,II) / TEMP
4033      CONTINUE
4034      DAIR(IW) = 0.0
4035      DO 6041 IAC = 1,N5
4036      AKGS(IAC,IW,II) = (VLS(L) * ACSABA(IAC,IS,II) *
4037      PADKW(IAC,IW,II)) / TEMP
4038      AKGS(IAC,IW,II) = AKGS(IAC,IW,II) * FICAS(2,II)
4039      DAIR(IW) = DAIR(IW) + AKGS(IAC,IW,II)
4040      CONTINUE
4041      DSSM(IW) = 0.0
4042      DO 6042 ISSM = 1,N6
4043      XMKGS(ISSM,IW,II) = (VLS(L) * SSMALO(ISSM,IS,II) *
4044      PMOKW(ISSM,IW,II)) / TEMP
4045      DSSM(IW) = DSSM(IW) + XMKGS(ISSM,IW,II)
4046      CONTINUE
4047      WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4048      CONTINUE
4049      ENDIF
4050      C
4051      C
4052      C
4053      C
4054      DO 6045 IW=1,N3
4055      IF (WLS(IW,IS,L) .GT. WS(IW,L)) THEN
4056      DGRD(IW) = 0.0
4057      FRAC = WSI(IW,IS,L)/WLS(IW,IS,L)
4058      DO 6037 KW=1,N4
4059      GKGS(KW,IW,II) = GKGS(KW,IW,II)*FRAC
4060      DGRD(IW) = DGRD(IW) + GKGS(KW,IW,II)
4061      CONTINUE
4062      DAIR(IW) = 0.0
4063      FRAC = WS(IW,L)/WLS(IW,IS,L)
4064      DO 6038 IAC=1,N5
4065      AKGS(IAC,IW,II) = AKGS(IAC,IW,II)*FRAC
4066      DAIR(IW) = DAIR(IW) + AKGS(IAC,IW,II)
4067      CONTINUE
4068      DSSM(IW) = 0.0
4069      DO 6039 ISSM=1,N6
4070      XMKGS(ISSM,IW,II) = XMKGS(ISSM,IW,II)*FRAC
4071      DSSM(IW) = DSSM(IW) + XMKGS(ISSM,IW,II)
4072      CONTINUE
4073      WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4074      ENDIF
4075      CONTINUE
4076      C
4077      ELSE
4078      C
4079      C
4080      C
4081      C
4082      C
4083      C
4084      C
4085      C
4086      C
4087      C

```

***** CHECK OUT THE PROPER SUBSCRIPT FOR 6024 & 6021 *****

ALTERNATE METHOD OF COMPUTING GROUND WEAPON LOSS.

```

4088 C
4089 C
4090
4091 IF (L.NE.ISATT) THEN
4092   DO 6024 IW = 1,N3
4093     WLS(IW,IS,L) = 0.0
4094     DGRD(IW) = 0.0
4095     DO 6021 KW = 1,N4
4096       GKGS(KW,IW,II) = WSI(KW,IS,II) * PWAkW(KW,IW,II) *
4097         SFGND(KPCAS,2,L)
4098     CONTINUE
4099     DAIR(IW) = 0.0
4100     DO 6022 IAC = 1,N5
4101       AKGS(IAC,IW,II) = ACSABA(IAC,IS,II) * PAAkW(IAC,IW,II)
4102       * SFCAS(KPCAS,2,L)
4103     DAIR(IW) = DAIR(IW) + AKGS(IAC,IW,II)
4104     CONTINUE
4105     DSSM(IW) = 0.0
4106     DO 6023 ISSM = 1,N6
4107       XMKGS(ISSM,IW,II) = SSMALO(ISSM,IS,II) *
4108         PMAkW(ISSM,IW,II) * SFSSM(KPCAS,2,L)
4109     DSSM(IW) = DSSM(IW) + XMKGS(ISSM,IW,II)
4110     CONTINUE
4111     WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4112     CONTINUE
4113   ELSE 1 L IS SECTOR ATTACKER
4114   DO 6049 IW = 1,N3
4115     WLS(IW,IS,L) = 0.0
4116     DGRD(IW) = 0.0
4117     DO 6046 KW = 1,N4
4118       GKGS(KW,IW,II) = WSI(KW,IS,II) * PWDkW(KW,IW,II) *
4119         SFGND(KPCAS,1,L)
4120     DGRD(IW) = DGRD(IW) + GKGS(KW,IW,II)
4121     CONTINUE
4122     DAIR(IW) = 0.0
4123     DO 6047 IAC = 1,N5
4124       AKGS(IAC,IW,II) = ACSABA(IAC,IS,II) * PADkW(IAC,IW,II) *
4125         SFCAS(KPCAS,1,L)
4126     DAIR(IW) = DAIR(IW) + AKGS(IAC,IW,II)
4127     CONTINUE
4128     DSSM(IW) = 0.0
4129     DO 6048 ISSM = 1,N6
4130       XMKGS(ISSM,IW,II) = SSMALO(ISSM,IS,II) *
4131         PMDKW(ISSM,IW,II) * SFSSM(KPCAS,1,L)
4132     DSSM(IW) = DSSM(IW) + XMKGS(ISSM,IW,II)
4133     CONTINUE
4134     WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4135     CONTINUE
4136   ENDIF
4137
4138 ADJUST GKGS, AKGS, AND XMKGS
4139
4140 DO 6085 IW=1,N3
4141   IF (WLS(IW,IS,L) .LE. WS(IW,L)) THEN
4142     IF (DGRD(IW) .GT. WSI(IW,IS,L)) THEN
4143       FRAC = WSI(IW,IS,L)/DGRD(IW)
4144

```

```

4145      DO 6076 KW=1,N4
4146         GKGS(KW,IW,II) = GKGS(KW,IW,II)*FRAC
4147      CONTINUE
4148      DGRD(IW) = DGRD(IW)*FRAC
4149      ENDIF
4150      ELSE      I WLS > WS
4151         FRAC = WSI(IW,IS,L)/WLS(IW,IS,L)
4152      DO 6077 KW=1,N4
4153         GKGS(KW,IW,II) = GKGS(KW,IW,II)*FRAC
4154      CONTINUE
4155      DGRD(IW) = DGRD(IW)*FRAC
4156      FRAC = WS(IW,L)/WLS(IW,IS,L)
4157      DO 6078 IAC=1,N5
4158         AKGS(IAC,IW,II) = AKGS(IAC,IW,II)*FRAC
4159      CONTINUE
4160      DAIR(IW) = DAIR(IW)*FRAC
4161      DO 6079 ISSM=1,N6
4162         XMKGS(ISSM,IW,II) = XMKGS(ISSM,IW,II)*FRAC
4163      CONTINUE
4164      DSSM(IW) = DSSM(IW)*FRAC
4165      ENDIF
4166      WLS(IW,IS,L) = DGRD(IW) + DAIR(IW) + DSSM(IW)
4167      CONTINUE
4168      ENDIF
4169      C
4170      C
4171      C
4172      CALL ADJUST(EXNPM(1,1,II),(LNW*LNW),VFACOR)
4173      CALL ADJUST(TEXNPM(1,IS,II),(LNW*LNW),VFACOR)
4174      CALL ADJUST(EXPM(1,1,IS,II),(LNW*LNW*LNW*LNW),VFACOR)
4175      CALL ADJUST(SMENG(1,1,IS,II),(LNW*LNW*LNW),VFACOR)
4176      DO JW = 1,N4
4177         DO ISM = 1,NSM(II)
4178            TSMSTK = SMSTK(ISM,IS,II) - SMENG(JW,ISM,IS,II) *
4179                & WSI(JW,IS,II)
4180            SMSTK(ISM,IS,II) = MAX(0.0, TSMSTK)
4181            ENDDO
4182            ENDDO
4183            C
4184            IF (KPSMGA(15).EQ.1.AND.ISMUS(II).EQ.1) THEN
4185               DO 636 ISM=1,NSM(II)
4186                  WRITE (MOT,630) ISM, II, ICYCLE
4187                  FORMAT (/, ' MUNITIONS TYPE', I4, ' FIRED ON TARGETS BY SIDE', I4,
4188                      & ' CYCLE', I4)
4189                  WRITE(MOT,633) ((AWT(IW,IWORD,L),IWORD=1,2),IW=1,N3)
4190                  FORMAT(1X,<N3>(2A4))
4191                  DO 631 JW = 1,N4
4192                     WRITE (MOT,632)(EXPAN(JW,IW,ISM,IS,II),IW = 1,N3)
4193                     FORMAT (1X,16F8.3)
4194                     WRITE (MOT,633) ISM, II, ICYCLE
4195                     FORMAT (/, ' MUNITIONS TYPE', I4, ' ON ALL TARGETS BY SIDE', I4,
4196                         & ' CYCLE', I4)
4197                     DO 634 JW = 1,N4
4198                        WRITE (MOT,635)(AWT(JW,IWORD,II),IWORD=1,2),SMENG(JW,ISM,IS,II)
4199                        FORMAT (1X,2A4,1X,10F8.3)
4200                        CONTINUE
4201                     ENDIF

```

C CALCULATE TOTAL MUNITION SPENT BY SINGLE WEAPON TYPE FOR ALL SECTORS

```

C
DO ISM=1,NSM(11)
DO JW=1,N4
  CSMENG(ISM,IS,II)=CSMENG(ISM,IS,II)+(SMENG(JW,ISM,IS,II)*WSI(JW,IS,II))
1  END DO
END DO

```

C FOR DETAILED

```

C
IF (IPRB.EQ.1) THEN
  IF (L.EQ.ISATT) THEN
    WRITE(MOT,6070) L
6070 FORMAT(/,' WEAPON LOSSES FOR ATTACKER --- SIDE ',I2)
  ELSE
    WRITE(MOT,6073) L
6073 FORMAT(/,' WEAPON LOSSES FOR DEFENDER --- SIDE ',I2)
  ENDIF
  CALL DEBUG(MOT,'DAIR',DAIR,LNW,1,1)
  CALL DEBUG(MOT,'DGRD',DGRD,LNW,1,1)
  CALL DEBUG(MOT,'DSSM',DSSM,LNW,1,1)
ENDIF

```

```

DO 6056 IW = 1,N3
DO 6053 KW = 1,N4
  SGKGS(KW,IW,IS,II) = SGKGS(KW,IW,IS,II) + GKGS(KW,IW,II)
CONTINUE
DO 6054 IAC = 1,N5
  SAKGS(IAC,IW,IS,II) = SAKGS(IAC,IW,IS,II) + AKGS(IAC,IW,II)
CONTINUE
DO 6055 ISSM = 1,N6
  SAKGS(ISSM,IW,IS,II) = SAKGS(ISSM,IW,IS,II) +
    XAKGS(ISSM,IW,II)

```

```

6055 CONTINUE
6056 CONTINUE

```

C COMPUTE PERSONNEL LOSSES USING THE WEAPON LOSSES AND THE FACTOR

C FTWPL

C CIRCUMVENT THE FOLLOWING CODE TO ALLOW FOR THE PERSONNEL CASUALTIES TO BE
C CALCULATED USING WEAPONS AND FTWPL (FACTOR TO COMPUTE PERSONNEL CASUALTIES
C FROM WEAPON LOSSES SUFFERED IF THE CURVE SWITCH IS OFF)

C 55) CALCULATE CASUALTIES TO BOTH SIDES AND SUBTRACT OUT LOSSES

```

IF( IPLSC.EQ.0 )THEN
  IF (L.EQ.1) THEN
    N1 = 1
    N2 = NDS(1S,1)
  ELSE I L = 2

```

```

4259      N1 = NDS(IS,1) + 1
4260      N2 = NDS(IS,1) + NDS(IS,2)
4261      ENDIF
4262      PLS(IS,L) = 0.
4263      IF (NDS(IS,L).GT.0) THEN
4264          DO IDS = N1,N2
4265              ID = IDLABA(IDS,IS)
4266
4267      C MAKE SURE NO HIGHER ECHELON UNITS ARE PROCESSED
4268      C
4269      IF (ITD(ID).NE.-1) THEN
4270          DIST = ABS(FEBA(IS) - CDVLOC(ID))
4271          FRACW = (DFBACT(L) - DIST)/DFBACT(L)
4272          TEMP = PCS(L) * (PDIV(ID) * FRACW)
4273          TPDIV(L) = TPDIV(L) + PDIV(ID)
4274
4275      C SPECIAL GAS EFFECT ADJUSTMENT
4276      C
4277      TEMPG = 0.
4278      TEMPA = 0.
4279      TEMPM = 0.
4280      IF (II.NE.ISA(IS)) THEN      ! DEFENDER INFLECTING LOSSES
4281          TEMP1 = VGDBAI(IS,II) + VADBA(IS,II) + VMDBA(IS,II)      ! TOTAL KILLS
4282          IF (TEMP1.GT..0001) THEN
4283              TEMPG = TEMP * VGDBAI(IS,II) / TEMP1      ! KILLS BY GROUND
4284              TEMPA = TEMP * VADBA(IS,II) / TEMP1      ! KILLS BY AIR
4285              TEMPM = TEMP * VMDBA(IS,II) / TEMP1      ! KILLS BY SSM'S
4286          ENDIF
4287      ELSE      ! ATTACKER INFLECTING LOSSES
4288          TEMP1 = VGABAI(IS,II) + VAABA(IS,II) + VMABA(IS,II)
4289          IF (TEMP1.GT..0001) THEN
4290              TEMPG = TEMP * (VGABAI(IS,II) / TEMP1)
4291              TEMPA = TEMP * (VAABA(IS,II) / TEMP1)
4292              TEMPM = TEMP * (VMABA(IS,II) / TEMP1)
4293          ENDIF
4294      ENDIF
4295      TEMPA = TEMPA * FICAS(1,II)
4296      SCASUAL(1,IS,L) = SCASUAL(1,IS,L) + TEMPG
4297      SCASUAL(2,IS,L) = SCASUAL(2,IS,L) + TEMPA
4298      SCASUAL(3,IS,L) = SCASUAL(3,IS,L) + TEMPM
4299      TEMP = TEMPG + TEMPA + TEMPM      ! CAS ADJUSTED KILLS
4300      TEMP = AMINI(TEMP,PDIV(ID)*FRACW)
4301      PDIV(ID) = PDIV(ID) - TEMP
4302      PLS(IS,L) = PLS(IS,L) + TEMP
4303      PLST(L) = PLST(L) + PLS(IS,L)
4304      IC = ICBIT(ID)
4305      CPLS(IS,IC) = CPLS(IS,IC) + TEMP
4306      PIARP(IC) = PIARP(IC) + FPCR(IC) * TEMP
4307      ENDIF
4308      ENDDO
4309      ENDIF
4310
4311      C COMPUTE TOTAL PERSONNEL LOSSES FROM WEAPON LOSSES
4312      C
4313      ELSE
4314          PLSG = 0.
4315          PLSA = 0.

```

```

4316      PLSM = 0.
4317      DO IW = 1,N3
4318          PLSG = DGRD(IW) * FTCWPL(IW,L) + PLSG
4319          PLSA = DAIR(IW) * FTCWPL(IW,L) + PLSA
4320          PLSM = DSSM(IW) * FTCWPL(IW,L) + PLSM
4321      ENDDO
4322      C
4323      SCASUAL(1,IS,L) = SCASUAL(1,IS,L) + PLSG
4324      SCASUAL(2,IS,L) = SCASUAL(2,IS,L) + PLSA
4325      SCASUAL(3,IS,L) = SCASUAL(3,IS,L) + PLSM
4326      C
4327      ENDIF
4328      6060 CONTINUE
4329      C
4330      C 65) COMPUTE WEAPONS LOST BY TYPE FOR EACH DIVISION.
4331      C
4332      N2 = 0
4333      DO 6520 L = 1,2
4334          IF (NDS(IS,L) .GT. 0) THEN
4335              N1 = 1 + N2
4336              N2 = N2 + NDS(IS,L)
4337              N3 = NW(L)
4338              DO 6510 IDS = N1,N2
4339                  ID = IDLABA(IDS,IS)
4340              C
4341              C MAKE SURE NO HIGHER ECHELON UNITS ARE PROCESSED
4342              C
4343              IF (ITD(ID) .EQ. -1) GOTO 6510
4344              IC = ICBIT(ID)
4345              DIST = ABS(FEBA(IS) - CDVLOC(IU))
4346              FRACW = (DFBACT(L) - DIST)/DFBACT(L)
4347              DO 6505 IW = 1,N3
4348                  IF (WS(IW,L) .GT. 0.0) THEN
4349                      IF (ERGMX(IW,L) .EQ. 0.) THEN
4350                          WPSOH = WDIV(IW,ID) * FRACW
4351                      ELSE IF (DIST .LE. ERGMX(IW,L)) THEN
4352                          WPSOH = WDIV(IW,ID)
4353                      ELSE
4354                          WPSOH = 0.
4355                      ENDIF
4356                      TEMP = WLS(IW,IS,L) * (WPSOH / WS(IW,L))
4357                      TEMP1 = AMIN1(TEMP,WPSOH)
4358                      IF (TEMP1 .LE. .00001) TEMP1 = 0.
4359      C
4360      C COMPUTE PERSONNEL LOST TO THE UNIT BASED ON THE WEAPONS LOST
4361      C IF IPLSC FLAG IS SET.
4362      C
4363          IF (IPLSC .NE. 0) THEN
4364              TEMPP = TEMP1 * FTCWPL(IW,L)
4365              PDIV(ID) = PDIV(ID) - TEMPP
4366              CPLS(IS,IC) = CPLS(IS,IC) + TEMPP
4367              PLS(IS,L) = PLS(IS,L) + TEMPP
4368              PIARP(IC) = PIARP(IC) + FPCR(IC) * TEMPP
4369              ENDIF
4370      C
4371      WDIV(IW,ID) = WDIV(IW,ID) - TEMP1
4372      IF (WDIV(IW,ID) .LT. 0.00001) WDIV(IW,ID) = 0.0

```

```

4373      CWLS(IW,IS,IC) = CWLS(IW,IS,IC) + TEMP1
4374      ENDIF
4375      CONTINUE
4376      CONTINUE
4377      ENDIF
4378      CONTINUE
4379      IF (IPRB.EQ.1) THEN
4380        WRITE(MOT,123)
4381        WRITE(MOT,22) (PLS(IS,L),L = 1,2)
4382        WRITE(MOT,127)
4383        WRITE(MOT,22) (CPLS(IS,IC),IC = 1,NIC)
4384      ENDIF
4385
4386      C TOTAL EQUIP OF THIS TYPE LOST THIS CYCLE
4387      C
4388      DO L = 1,2
4389        N3 = NW(L)
4390        DO IW = 1,N3
4391          WLST(IW,L) = WLST(IW,L) + WLS(IW,IS,L)
4392        ENDDO
4393      ENDDO
4394
4395      IF (IPRB.EQ.1) THEN
4396        DO 6580 L=1,2
4397          N3 = NW(L)
4398          WRITE(MOT,121) L
4399          WRITE(MOT,22) (WLS(IW,IS,L),IW = 1,N3)
4400          N1 = 1
4401          N2 = NICL - 1
4402          IF (L.EQ.1) GOTO 6565
4403          N1 = NICL
4404          N2 = NIC
4405          CONTINUE
4406          DO 6575 IC = N1,N2
4407            WRITE(MOT,122) IC
4408            WRITE(MOT,22) (CWLS(IW,IS,IC),IW = 1,N3)
4409          CONTINUE
4410          DO 6580 L = 1,2
4411            I1 = 3-L
4412            N3 = NW(L)
4413            N4 = NW(I1)
4414            N5 = NAC(I1)
4415            WRITE(MOT,124) I1,L
4416            DO 6581 KW = 1,N4
4417              WRITE(MOT,22) (GKGS(KW,IW,I1),IW = 1,N3)
4418            CONTINUE
4419            WRITE(MOT,125) I1,L
4420            DO 6582 IAC = 1,N5
4421              WRITE(MOT,22) (AKGS(IAC,IW,I1),IW = 1,N3)
4422            CONTINUE
4423            CONTINUE
4424            CALL DEBUG(MOT,'XMKGS',XMKGS,LNSSL,LNW,2)
4425          ENDIF
4426
4427      C
4428      C
4429      C 70) COMPUTE TONS OF SUPPLY LOST.

```

```

4430 C
4431 C
4432 IF (ISEFFD.NE.3) THEN
4433 N2 = 0
4434 DO 7025 L = 1,2
4435 N1 = N2 + 1
4436 N2 = N2 + NDS(IS,L)
4437 IF (NDS(IS,L).GT.0) THEN
4438 K = 3 - L
4439 N3 = NW(K)
4440 N4 = NAC(K)
4441 N5 = NSSM(K)
4442 DO 7024 ISPT = 1, NSPT
4443 TT1 = 0.0
4444 TSL(IS,ISPT,L) = 0.
4445 DO 7005 IW = 1, N3
4446 TSL(IS,ISPT,L) = TSL(IS,ISPT,L) + SLWCB(IW,ISPT,K) *
4447 WSI(IW,IS,K)
4448 CONTINUE
4449 TT1 = TSL(IS,ISPT,L)
4450 TEMP1 = 0.0
4451 TEMP2 = 0.0
4452 DO 7010 IAC = 1, N4
4453 ISUPPLIES DESTROYED BY CAS AIRCRAFT
4454 TEMP1 = TEMP1 + SLSCAS(IAC,ISPT,K) * ACSABA(IAC,IS,K)
4455 CONTINUE
4456 DO 7012 ISSM = 1, N5
4457 TEMP2 = TEMP2 + SLSSSM(ISSM,ISPT,K) * SSMALO(ISSM,IS,K)
4458 CONTINUE
4459 TSL(IS,ISPT,L) = TSL(IS,ISPT,L) + TEMP1 + TEMP2
4460 TSS = 0.
4461 DO 7015 IDS = 1, N2
4462 ID = IDLABA(IDS,IS)
4463 TSS = TSS + SDIV(ID,ISPT)
4464 CONTINUE
4465 IF (TSS.GT.0.) THEN
4466 DO 7020 IDS = 1, N2
4467 ID = IDLABA(IDS,IS)
4468 RRR = SDIV(ID,ISPT) / TSS
4469 TEMP = TSL(IS,ISPT,L) * RRR
4470 TEMP = AMIN1(TEMP,SDIV(ID,ISPT))
4471 SDIV(ID,ISPT) = SDIV(ID,ISPT) - TEMP
4472 IC = ICBIT(ID)
4473 FRAC = 1.0
4474 IF (TEMP.GT.0.) FRAC = TEMP / TEMP
4475 CSDGW(ISPT,IC) = CSDGW(ISPT,IC) + (RRR * FRAC * TT1)
4476 CSDAW(1,ISPT,L) = CSDAW(1,ISPT,L) + (RRR*FRAC*TEMP1)
4477 CSDSW(1,ISPT,L) = CSDSW(1,ISPT,L) + (RRR*FRAC*TEMP2)
4478 CONTINUE
4479 ENDIF
4480 CONTINUE
4481 IF (IPRB.EQ.1) THEN
4482 WRITE(MOT,129)
4483 N2 = 0
4484 DO 7080 L = 1,2
4485 IF (NDS(IS,L).GT.0) THEN
4486

```

```

4487 N3 = NW(L)
4488 WRITE(MOT,126) L
4489 DO 7079 IDS = N1,N2
4490 ID = IDLABA(IDS,IS)
4491
4492 C MAKE SURE HIGHER ECHELON UNITS ARE NOT PROCESSED
4493 C
4494 IF (IT.EQ.-1) GOTO 7079
4495 WRITE(MOT,21),ID,PDIV(ID),(WDIV(IW,ID),IW=1,N3),
4496 (SDIV(ID,ISPT),ISPT=1,NSPT)
4497
4498 &
4499 CONTINUE
4500 ENDIF
4501 CONTINUE
4502 ENDIF
4503
4504 C 75) COMPUTE NUMBER OF WEAPONS REPAIRABLE AFTER GROUND COMBAT
4505 C
4506 C
4507 N2 = 0
4508 DO 7520 L = 1,2
4509 IF (NDS(IS,L).GT.0) THEN
4510 N1 = 1 + N2
4511 N2 = N2 + NDS(IS,L)
4512 N3 = NW(L)
4513 I = 0
4514 IF (ISATT.EQ.L) I = 1
4515 IF (I.EQ.0) THEN
4516 IF (KP.EQ.2.OR.KP.EQ.3.OR.KP.EQ.4) I = 2
4517 IF (KP.EQ.1.OR.KP.EQ.5) I = 3
4518 IF (KP.EQ.6) I = 4
4519 ENDIF
4520 DO 7515 IDS = N1,N2
4521 ID = IDLABA(IDS,IS)
4522
4523 C MAKE SURE NO HIGHER ECHELON UNITS ARE PROCESSED
4524 C
4525 IF (ITD(ID).EQ.-1) GOTO 7515
4526 IC = ICBIT(ID)
4527 DIST = ABS(FEBA(IS) - CDVLOC(ID))
4528 FRACW = (DFBACT(L) - DIST)/DFBACT(L)
4529 DO 7510 IW = 1,N3
4530 IF (WS(IW,L).GT.0.) THEN
4531 IF (ERGMX(IW,L).EQ.0.) THEN
4532 WPSOH = WDIV(IW,ID) * FRACW
4533 ELSE IF (DIST.LE.ERGMX(IW,L)) THEN
4534 WPSOH = WDIV(IW,ID)
4535 ELSE
4536 WPSOH = 0.
4537 ENDIF
4538 TEMP = WLS(IW,IS,L) * (WPSOH / WS(IW,L))
4539 TEMP1 = (1. - PWATS(IW,1,L)) * TEMP
4540 C ALLOCATE TO APPROPRIATE REPAIR POOLS
4541 WDRRPC(IW,IS,IC) = WDRRPC(IW,IS,IC) +
4542 TEMP1 * PWDRP(2,IW,L)
4543 &
4544 WDRRP(IW,IC) = WDRRP(IW,IC) + TEMP * PWDRP(3,IW,L)

```

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47

Page 55

VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR;9

```

4544 C WEAPONS REPAIRED AT DIVISION LEVEL ARE RETURNED TO THE DIVISION IMMEDIATELY
4545     WDIV(IW,ID) = WDIV(IW,ID) + TEMP1 * PWDRP(1,IW,L)
4546 C
4547 C ADD WEAPONS REPAIRED AT THE DIVISION TO CUMULATIVE REPAIRED WEAPONS
4548 C RETURNED TO DIVISION
4549     REPWPN(IW,IS,IC) = REPWPN(IW,IS,IC) +
4550     &     TEMP1 * PWDRP(1,IW,L)
4551     ENDIF
4552 7510 CONTINUE
4553 7515 CONTINUE
4554     ENDIF
4555 7520 CONTINUE
4556 C
4557     IF (IPRB.EQ.1) THEN
4558     WRITE(MOT,130)
4559     DO 7580 IC = 1,NIC
4560     L = 1
4561     IF (IC.GE. NICL) L = 2
4562     N3 = NW(L)
4563     WRITE(MOT,22)(WDRRP(IW,IC),IW = 1,N3)
4564 7580 CONTINUE
4565     ENDIF
4566 C
4567 C 80) CALCULATE CASUALTIES AND WEAPON LOSSES INFLICTED BY AIR, PER
4568 C SORTIE.
4569 C
4570 DO 8030 L=1,2
4571 K = 3 - L
4572 TEMP = 0.
4573 IF (ISATT.EQ.K) THEN
4574     TEMP1 = VAABA(IS,K) + VGABAI(IS,K) + VMABA(IS,K)
4575     IF (TEMP1.GT..0001) TEMP = VAABA(IS,K) / TEMP1
4576 ELSE
4577     TEMP1 = VADBA(IS,K) + VGDBAI(IS,K) + VMDBA(IS,K)
4578     IF (TEMP1.GT..0001) TEMP = VADBA(IS,K) / TEMP1
4579     ENDIF
4580 CDACS(IS,L) = PLS(IS,L) * TEMP
4581 NX = NW(L)
4582 DO 8015 IW = 1,NX
4583     WLDAS(IW,IS,L) = WLS(IW,IS,L) * TEMP
4584 8015 CONTINUE
4585     N9 = NAC(K)
4586     SUM = 0.
4587     DO 8020 IAC = 1,N9
4588     SUM = SUM + ACSABA(IAC,IS,K)
4589 8020 CONTINUE
4590     IF (SUM.GT..001) THEN
4591     CDACS(IS,L) = CDACS(IS,L) / SUM
4592     DO 8025 IW = 1,NX
4593     WLDAS(IW,IS,L) = WLDAS(IW,IS,L) / SUM
4594 8025 CONTINUE
4595     ENDIF
4596 8030 CONTINUE
4597 C
4598 9999 CONTINUE
4599     DO 8099 L = 1,2
4600     IF (SFAPP(L).EQ.0.0) THEN

```

```

4601 SFAPP(L) = SFAPPT(L)
4602 GOTO 8099
4603 ENDIF
4604 SFAPP(L) = SFAPP(L)/NUMINC(L)
4605
4606 8099 CONTINUE
4607 C
4608 IF (IPRB .EQ. 1) CALL DEBUG(MOT,'SFAPP',SFAPP,2,1,1)
4609 C
4610 OUTPUT CONTRIBUTIONS TO DIVISION EFFECTIVENESS
4611 C
4612 IF (KPSMGA(2) .EQ. 1) THEN
4613 WRITE(MOT,8110)ICYCLE
4614 FORMAT(1H1,/,1X,'TABLE 5.2 CYCLE',14,/,
4615 & ' DIVISION EFFECTIVENESS -- FRACTIONS BASED ON BASE -- ',
4616 & 'EFFECTIVENESS FROM TABLES',
4617 & ,1X,120(1H-),/)
4618 WRITE(MOT,8115)
4619
4620 8115 FORMAT(1H0,4H DIV,4H DIV, 7H* FRAC,7H EFF AT, 7H EFF DF,
4621 & 7H*FRC AT, 7H EFF AT, 7H FRC DF, 7H EFF DF, 7H* FRAC,
4622 & 10H VALUE,/, 1X,4H ID,4H TYPE,7H*PEOPLE,7H PEOPLE,
4623 & 7H PEOPLE, 7H* WPNS,7H WPNS,7H WPNS,7H WPNS,
4624 & 7H*SUPPLY,7H SUPPLY, 7H* CHEM,7H*ATTACK,7HDEFENSE,
4625 & 10H* ATTACK, 10H DEFENSE,/)
4626 DO 8120 IDI = 1,LND
4627 ID = IDENTR(IDI)
4628 IF (1TD(ID) .EQ. -1) GOTO 8120
4629 IF (ID .NE. 0) THEN
4630 WRITE(MOT,8130) ID,1TD(ID),PPSAR(ID),PEAR(ID),PEDAR(ID),
4631 & WATSAR(ID),WEAR(ID),WOTSAR(ID),WEDAR(ID),
4632 & DSHAR(ID),SEFAR(ID),FDEFAR(ID),EFFDA(ID),
4633 & EFFDD(ID),VDAC(ID),VDDC(ID)
4634 FORMAT(1H,214,12F7.3,2F10.2)
4635 ENDF
4636 8120 CONTINUE
4637 ENDIF
4638 C
4639 NOW WRITE SECTOR VALUES
4640 C
4641 IF (KPSMGA(1) .EQ. 1) THEN
4642 WRITE(MOT,8201)ICYCLE,(IS,IS = 1,NS)
4643 FORMAT(1H1,/,TABLE 5.1 CYCLE',14,/,21X,80(1H-),/,
4644 & 21X,'GROUND CAPABILITIES AND FORCE RATION BY SECTOR',/,
4645 & 1X,20X,6HSECTOR,14X,8110,/)
4646 WRITE(MOT,8205)(ISA(IS),IS = 1,NS)
4647 FORMAT(1H0,21HSECTOR ATTACKER -ISA-, 19X,8110)
4648 DO 8210 L = 1,2
4649 WRITE(MOT,8211)ASIDE(L),(IS,IS = 1,NS)
4650 FORMAT(1H0,21X,17HCAPABILITIES FOR ,A4,/,
4651 & 1X,20X,6HSECTOR,14X,8110,/)
4652 WRITE(MOT,8221)(VGABA(IS,L),IS = 1,NS)
4653 FORMAT(1H0,32HGROUND VALUES FOR ATTACK -VGABA-, 8X,8F10.2)
4654 WRITE(MOT,8222)(VGDBA(IS,L),IS = 1,NS)
4655 FORMAT(1H,33HGROUND VALUES FOR DEFENSE -VGDBA-, 7X,8F10.2)
4656 WRITE(MOT,8223)(VAABA(IS,L),IS = 1,NS)
4657 FORMAT(1H0,29HVAIR VALUES FOR ATTACK -VAABA-,11X,8F10.2)

```

```

4658 WRITE(MOT,8224)(VADBA(IS,L),IS = 1,NS)
4659 FORMAT(1H , 30HAI R VALUES FOR DEFENSE -VADBA-,10X,8F10.2)
4660 WRITE(MOT,8212)(VMABA(IS,L),IS=1,NS)
4661 FORMAT(/ , 30HAI R VALUES FOR ATTACK -VMABA-,11X,8F10.2)
4662 WRITE(MOT,8213)(VMOBA(IS,L),IS=1,NS)
4663 FORMAT(1H , 30HAI R VALUES FOR DEFENSE -VMOBA-,10X,8F10.2)
4664 WRITE(MOT,8225)(PPESAR(IS,L),IS = 1,NS)
4665 FORMAT(1H0 , 30HSECTOR SUPPLY EFFECTIVENESS -PPESAR-,5X,8F10.4)
4666 WRITE(MOT,8226)(FRCA(D-,20X,8F10.4)
4667 FORMAT(1H0 , 20HVALUE RATIO -FRCA/D-,20X,8F10.4)
4668 WRITE(MOT,8227)(PCASAR(IS,L),IS = 1,NS)
4669 FORMAT(1H0 , 30HVALUE LOST FOR PERSONELL -PCS-,10X,8F10.4)
4670 WRITE(MOT,8228)(VLSAR(IS,L),IS = 1,NS)
4671 FORMAT(1H , 20HVALUE LOST FOR WEAPONS -VLS-,12X,8F10.4)
4672 CONTINUE
4673 ENDIF
4674
4675 C
4676 C 85) AVERAGE CASUALTIES AND WEAPON LOSSES PER SORTIE OVER REGIONS.
4677 C
4678 M2 = 0
4679 DO 8550 L = 1,2
4680 NX = NW(L)
4681 M1 = M2 + 1
4682 M2 = M2 + NR(L)
4683 DO 8530 IR = M1,M2
4684 FPCAS(IR) = 0.
4685 DO 8510 IW = 1,NX
4686 FWCAS(IW,IR) = 0.
4687 CONTINUE
4688 IS1 = NLSR(IR)
4689 IS2 = NLSR(IR)
4690 DO 8520 IS = IS1,IS2
4691 FPCAS(IR) = FPCAS(IR) + CDACS(IS,L)
4692 DO 8515 IW = 1,NX
4693 FWCAS(IW,IR) = FWCAS(IW,IR) + WLDAS(IW,IS,L)
4694 CONTINUE
4695 TOT = IS2 - IS1 + 1
4696 FPCAS(IR) = FPCAS(IR) / TOT
4697 DO 8525 IW = 1,NX
4698 FWCAS(IW,IR) = FWCAS(IW,IR) / TOT
4699 CONTINUE
4700 F550 CONTINUE
4701
4702 C+
4703 C+
4704 C+
4705 C+ COMPUTE CUMULATIVE AIRMUNITIONS EXPENDED ON CAS. +
4706 C+ FOR TABLE 1.18 +
4707 C+
4708 DO L = 1,2
4709 IMT = 2 + 6 * (L - 1)
4710 DO IS = 1,NS
4711 DO IAM = 1,NAM(L)
4712 DO IAC = 1,NAC(L)
4713 CAMCAS(IAC,IAM,IS,L) = CAMCAS(IAC,IAM,IS,L) +
4714 (ACSABA(IAC,IS,L) * AMNL(IAC,IAM,IMT))

```



```
4772 FORMAT(1H0,38HMAABA(1) VMOBA(1) VMAABA(2) VMOBA(2))
4773 FORMAT(/, ' ADJUSTED ALLOCATION FOR MISSILE (ATTACK) — SIDE',
4774 & 13, ' AGAINST SIDE',13)
4775 FORMAT(/, ' ADJUSTED ALLOCATION FOR MISSILE (DEFENSE) — SIDE',
4776 & 13, ' AGAINST SIDE',13)
4777 FORMAT(/, ' ADJUSTED POTENTIAL FOR MISSILE (ATTACK) — SIDE',
4778 & 13, ' AGAINST SIDE',13)
4779 FORMAT(/, ' ADJUSTED POTENTIAL FOR MISSILE (DEFENSE) — SIDE',
4780 & 13, ' AGAINST SIDE',13)
4781 FORMAT(1H0, ' MISSILE ATTACK VALUES — SIDE',13)
4782 FORMAT(1H0, ' MISSILE DEFENSE VALUES — SIDE',13)
4783 END
```

PROGRAM SECTIONS

Name	Bytes	Attributes
0 \$CODE	59428	PIC CON REL LCL SHR EXE RD NOWRT LONG
1 \$PDATA	4842	PIC CON REL LCL SHR NOEXE RD NOWRT LONG
2 \$LOCAL	845592	PIC CON REL LCL NOSHR NOEXE RD WRT LONG
3 AGCOM	76852	PIC OVR REL GBL SHR NOEXE RD WRT LONG
4 AGWK	266320	PIC OVR REL GBL SHR NOEXE RD WRT LONG
5 BLANK	324752	PIC OVR REL GBL SHR NOEXE RD WRT LONG
6 BLANKRS	262416	PIC OVR REL GBL SHR NOEXE RD WRT LONG
7 BLANKSV	6668	PIC OVR REL GBL SHR NOEXE RD WRT LONG
8 BLANKW	80208	PIC OVR REL GBL SHR NOEXE RD WRT LONG
9 COMACH	39979	PIC OVR REL GBL SHR NOEXE RD WRT LONG
10 GCFM	157980	PIC OVR REL GBL SHR NOEXE RD WRT LONG
11 GCHTM	3688	PIC OVR REL GBL SHR NOEXE RD WRT LONG
12 GCRS	219848	PIC OVR REL GBL SHR NOEXE RD WRT LONG
13 GRDCOM	470076	PIC OVR REL GBL SHR NOEXE RD WRT LONG
14 GRDRS	13236	PIC OVR REL GBL SHR NOEXE RD WRT LONG
15 GRDSV	851872	PIC OVR REL GBL SHR NOEXE RD WRT LONG
16 SUPCOM	444452	PIC OVR REL GBL SHR NOEXE RD WRT LONG
17 TCCOM	58636	PIC OVR REL GBL SHR NOEXE RD WRT LONG
Total Space Allocated	4177845	

ENTRY POINTS

Address	Type	Name	References
---------	------	------	------------

0-00000000	GC		1#
------------	----	--	----

VARIABLES

Address	Type	Name	Attributes	References
2-000CE0D0	R*4	ADJA		3815= 3844= 3867
2-000CE0D4	R*4	ADJD		3816= 3837= 3876
2-000CDFD8	R*4	ALAM1		3265A 3311 3325 3337 3351
2-000CDFE8	R*4	ALAM2		3275A 3312 3324 3338 3350

2-000CDF78	R*4	ALOAD	3191=	3193	3196				
2-000CE0A0	R*4	APCSD	3538=	3539	3540				
2-000CE094	R*4	BTPRAT	3535=	3538	3916=				
8-00001134	R*4	CURINP	936	3497=	3501=	3513=	3541=		
8-00001138	R*4	CURVES	3864=	3873=					
			937	3499	3503	3511	3515	3543	
2-000CDCB8	R*4	DIST	3866	3875	1887	3428=	3429	3480	
			1836=	1837	4345=	4346	4351	4527=	
			4270=	4271					
			4528	4533					
2-000CE0A4	R*4	DSH	3540=	3541	3597	3802			
13-00034ABC	R*4	EFCE	1176	3265A	3275A				
4-0004104C	R*4	END_AGWK	525						
9-00009C27	CHAR	END_COMCH	1035	1074					
10-00026918	R*4	END_GC	1111						
11-00000E64	R*4	END_GCHIM	1121						
12-00035AC4	R*4	END_GCRS	1156						
15-000CFF9C	R*4	END_GSV	1284						
13-00035418	R*4	FASFR	1180						
13-0003541C	R*4	FCABFD	1181	3827					
13-00035420	R*4	FCDBFD	1182						
13-0003543C	R*4	FDSFR	1186						
2-000CE0AC	R*4	FFF	3581=	3829	3583	3584=	3585=	3586	
2-000CDC48	R*4	FRAC	1649=	3582=	4057=	4059	4063=	4065	
			4070	1653	4146	4148	4151=	4153	
			4155	4144=	4158	4160	4162	4164	
			4471=	4156=	4473	4474	4475		
			1837=	4472=	1869	1885	1886	3429=	
			3438	1867	3440	3458	3472	3474	
			3476	3478	4271=	4272	4300	4346=	
			4350	4528=	4532				
2-000CE0C8	R*4	FRCA	3813=	3827=	3864	3879	3888		
2-000CE0CC	R*4	FRCD	3814=	3831=	3832	3833	3834	3873	
			3880	3888					
2-000CDC50	R*4	FRM	1651=	1652(2)=	1653	1654	1655		
2-000CE264	I*4	I	4513=	4514=	4515	4516=	4517=	4518=	
			4539						
2-000CDF70	I*4	IAC	3189	3189	3190	3191(2)	3193(3)	3196(3)	
			3232	3232	3234=	3235	3302=	3304	
			3305	3306	3310=	3311(2)	3312(2)	3315=	
			3317	3318	3319	3323=	3324(2)	3325(2)	
			3389(2)=	3391(2)=	3613=	3614(2)	3615(2)	3618(2)=	
			3964=	3965(2)	3988=	3989(3)	3996(2)	3997	
			4013=	4014(2)	4035=	4036(3)	4038(2)	4039	
			4064=	4065(2)	4066	4100=	4101(3)	4103	
			4123=	4124(3)	4126	4157=	4158(2)	4232=	
			4233(3)	4421=	4422	4451=	4452(2)	4587=	
			4588	4712=	4713(4)	4718(2)			
2-000CDFD4	I*4	IADDR	3260=	3261	3262=	3263	3270=	3271	
			3272=	3273	3304=	3305	3306	3317=	
			3318	3319	3330=	3331	3332	3343=	

3344	3345	1721(2)	1722(2)	1729=	1730	1731
1720=	1721(2)	2096(2)	2097(2)	2123=	2124(2)	2125(2)
2095=	2096(2)	3128(2)	3129(2)	3144=	3148	3149
3127=	3128(2)	3151	3155(2)	3160(2)	3192=	3193(3)
3150	3151	3225	3226	3228=	3229	4711=
3196(3)	3225=	4718(3)	3502(3)	3510(3)	3514(3)	3906=
4713(3)	4718(3)	3498(3)	4305(2)	4306(3)	4344=	4366(2)
3418=	3498(3)	4304=	4383(2)=	4406=	4407	4408
3934(2)	4304=	4373(2)	4526=	4541(2)	4543(2)	4549(2)
4368(3)	4373(2)	4473(2)	4563			
4470=	4473(2)	4561				
4559=	4561					
938						
1769=	1771=	1773				

3344	3345	1721(2)	1722(2)	1729=	1730	1731
1720=	1721(2)	2096(2)	2097(2)	2123=	2124(2)	2125(2)
2095=	2096(2)	3128(2)	3129(2)	3144=	3148	3149
3127=	3128(2)	3151	3155(2)	3160(2)	3192=	3193(3)
3150	3151	3225	3226	3228=	3229	4711=
3196(3)	3225=	4718(3)	3502(3)	3510(3)	3514(3)	3906=
4713(3)	4718(3)	3498(3)	4305(2)	4306(3)	4344=	4366(2)
3418=	3498(3)	4304=	4383(2)=	4406=	4407	4408
3934(2)	4304=	4373(2)	4526=	4541(2)	4543(2)	4549(2)
4368(3)	4373(2)	4473(2)	4563			
4476=	4473(2)	4561				
4559=	4561					
938						
1769=	1771=	1773				

4339=	4361	4363
938		
1769=	1771=	1773

642
939
643
3061=
644

3062 3063 3072 3085(2) = 3086

941	1635	1659	1781	1941	1987
646					
878					

1635	1669	1781	1941	1987
878	2715	2765	2773	2781
2743	2853	2851	2872	2899
2803	3279	4186	4613	4642
3091	3413	3418	3420	3428
3412=	3440(3)	3442	3456	3458
3438(3)	3470	3472(3)	3476	3478
3469	3483(3)	3485	3496(2)	3500
3481(3)	3507	3512	3516	3517
3504	3535	3537	3547(2)	3548(2)
3518	3551	3556	3559(3)	3560(3)
3550	3562	3564	3566	3567
3562	3569	3571	3572(2)	3573(2)
3568	3575	3582(2)	3585(2)	3586
3574	3592(8)	3583	3788	3905=
3590	3597	3787	3919	3921
3906	3916	3917(2)	4270	4272
3931	3932(2)	4269	4339=	4343
4273	4300	4301(2)	4365(2)	4371(2)
4344	4345	4350	4352	4468
4372(2)	4460=	4461	4466	4525
4469(2)	4470	4490=	4521=	4525
4526	4527	4495(4)	4545(2)	4627=
4628	4629	4534	4545(2)	4627=
648	4630(16)			

4029 4030(10)
648

[illegible]

2705=	2708	2710(2)=	2712(2)=	2714	2715
2720	2728=	2736	2738(2)=	2740(2)=	2742
2743	2748	2791=	2796	2798(2)=	2800(2)

2802 2803 2808 2815= 2823 2825(2)=
2827(2)= 2829 2830 2835 2861=
2867(2)= 2869(2)= 2871 2872 2877 2884=
2892 2894(2)= 2896(2)= 2898 2899 2904
3356= 3358 3360(2)= 3362(2)= 3364 3366
3368 3373= 3375 3377(2)= 3379(2)= 3381
3383 3384
3265A 3267 3275A 3278 3664= 3675=
3689

2-000C0DFDC I.4 IFLAG

2-000C0DF24 I.4 IFLG
8-00001488 I.4 IGO
2-000C0CF4 I.4 IGWT

COMM

3060= 3062 3077= 3083= 3086=
942
1960= 1961 1962 2023= 2027 2028
2029 2030 2031(2) 2078= 2080 2081
2082 2085= 2088 2089 2091 2092
2096 2097 2100 2106= 2108
2109 2110 2113= 2116 2117 2119
2120 2124 2125 2128 2129 3069=
3070 3072 3074 1951 1954 1955
1943= 1945 1950 1951 1954 1955
1956 1957 1958 1959 1960 1964(2)
1965(2) 1973 1977 1978 1979 1980
1981(2) 1987 1989 1990 2004 2076
2088 2089 2095 2096(2) 2097(2) 2099
2100(2) 2101(2) 2116 2117 2123 2124(2)
2125(2) 2127 2128(2) 2129(2) 2136(2) 2137(2)
2150 2151 2152 2153 2157(2) 2162(2)
2173(2) 2174(2) 2215 2229 2230 2235(2)
2240 2241 2245 2250 2251 2252
2254 2262 2267(2) 2272 2273 2277
2282 2283 2286 2543 2546
2547 2552 2553(2) 2554(2) 2556(3) 2557(3)
2563 2564 2569 2570(2) 2571(2) 2573(3)
2574(3) 2581 2582 2583(2) 2584 2585
2586 2587 2588 2589 2592 2601 2607
2602(2) 2603 2604 2605 2606 2607
2608 2611 2624 2627(2) 2628(2) 2634(2)
2635(2) 2642 2643 2646 2648 2654
2657 2659 2672(3) 2673(3) 2678 2679
2680 2681(3) 2683(3) 2694(3) 2695(3) 2715
2720(2) 2743 2748(2) 2754 2755 2759
2760 2761 2768 2769 2770 2776
2777 2778 2784 2785 2786 2808(2)
2835(2) 2842 2846 2847 2848 2854
2855 2877(2) 2904(2) 2915 2917
2921 2924 2930(3) 2931(3) 2937 2938
2939(2) 2940 2942 2943 2944
2945(2) 2947(3) 2955 2956(2) 2957 2958
2959 2960 2961 2962(2) 2964(3) 2978(3)
2979(3) 2982 2983 2984(2) 2985 2986
2987 2988 2989 2990(2) 2992(3) 3000
3001(2) 3002 3003 3004 3005 3006
3007(2) 3009(3) 3023 3024 3025(2) 3026(2)
3034 3035 3036(2) 3037(2) 3121=
3124 3128(2) 3129(2) 3134(2) 3148
3149 3150 3151 3155(2) 3160(2) 3168
3169 3170 3171 3175(2) 3180(2) 3185

2-000C0C0D8 I.4 II

2-000CDC7C	I*4	IMM	3186	3189	3190	3191(2)	3193(4)	3196(4)
			3201	3202	3203	3205(5)	3208(5)	3218=
			3220	3221	3222	3223	3224	3226
			3227	3229	3230	3232	3233	3235
			3236	3237	3230	3239	3240	3241
			3242	3243	3244	3245	3246	3247
			3646=	3648(3)	3657	3660(3)	3677(3)	3681
			3686	3703	3707(3)	3721(3)	3725	3730
			3820=	3821(3)	3823	3824	3825(3)	3829(3)
			3853	3858	3874(2)	3876	3877(3)	3878(3)
			3880	3882(2)	3884(2)	3947=	3950	3951
			3952	3962(2)	3965(2)	3969(2)	3983(3)	3985
			3989(3)	3996(3)	3997	4001(3)	4003	4011(2)
			4014(2)	4018(2)	4030(3)	4032	4036(3)	4038(3)
			4039	4043(3)	4045	4059(2)	4060	4065(2)
			4066	4070(2)	4071	4095(3)	4097	4101(3)
			4103	4107(3)	4109	4118(3)	4120	4124(3)
			4126	4130(3)	4132	4146(2)	4153(2)	4158(2)
			4162(2)	4172	4173	4174	4175	4177
			4178(3)	4180	4184	4185	4186	4192
			4194	4198(2)	4205	4207(4)	4230(3)	4233(3)
			4236(3)	4280	4281(3)	4283	4284	4285
			4288(3)	4290	4291	4292	4295	4412=
			4414	4415	4416	4418	4420	4422
			1724=	1725(2)	1726(2)	1733=	1734	1735
			2099=	2100(2)	2101(2)	2127=	2128(2)	2129(2)
			3133=	3134(2)	3135(2)	3164=	3168	3169
			3170	3171	3175(2)	3180(2)	3204=	3205(3)
			3208(3)	3237=	3238	3240=	3241	
			3186=	3193	3196	4709=	4713	4718
2-000CDF68	I*4	IMT	943					
8-000014BC	I*4	INCYL	1032					
9-00009C18	CHAR	IND	944					
8-000014C0	I*4	INO	945					
6-000014C4	I*4	INOSTT						
2-000CDD8C	I*4	INSMFR	2250=	2251	2282=	2283	2587=	2588
			2606=	2607	2943=	2944	2960=	2961
			2988=	2989	3005=	3006		
			1959=	1960				
2-000CDCEC	I*4	INSMT	1979=	1980	2240=	2241	2272=	2273
2-000CDD04	I*4	INSMW	2584=	2585	2603=	2604	2759=	2760
			2768=	2769	2776=	2777	2784=	2785
			2846=	2847	2854=	2855	2940=	2941
			2957=	2958	2985=	2986	3002=	3003
			1865=	1866	1882=	1883		
			1961=	1962				
2-000CDCC0	I*4	INWPC	655					
2-000CDCF8	I*4	INWTIGT	3558(2)					
			3854=					
			3857=					
5-0002C6CC	I*4	IOMU	1202	4363				
2-000CE0F4	I*4	IPCAF	657					
2-000CE0FC	I*4	IPCDF						
13-00037A48	I*4	IPLSC						
5-0002C748	I*4	IPOPIN						
5-0002C74C	I*4	IPRB	658	1632	1633=	1639=	1661	1780
			3215	3617	3635	3766	3800	3886
			3939	4216	4379	4395	4481	4557

2-0000DC38 I.4 IPRST
8-000014C8 I.4 IPRD
5-0002C7F0 I.4 IPRS
8-000014CC I.4 IPT

8-000014D0 I.4 IPTSTT
2-000CE2D0 I.4 IR

13-00037A4C I.4 IRDOFF
4-0003B8B4 I.4 IS

COMM
COMM
COMM

COMM

COMM
COMM

4607	4725=				
1632=	4725				
946					
660					
947					
948					
4682=	4683	4685	4687	4688	4690(2)
4692(2)	4696(2)	4698(2)			
1203					
462	1619=	1620	1621	1624(4)	1626(4)
1650=	1651	1653	1655(2)	1656(3)	1763=
1766	1767	1768	1771	1776	1781
1785	1797	1803(2)A	1804	1805(2)	1810
1816	1820	1822	1827	1836	1869(2)
1886(2)	1889(2)	1902	1903	1906	1907
1909	1911	1912	1913	1914	1915
1916	1922	1923(4)	1925	1926	1927
1928	1957	1963	1981	1990	2006
2009	2031	2082	2110	2152	2153
2157	2162	2245	2277	2586	2605
2702(2)	2732(2)	2917	2921	2923(2)A	2931(2)
2942	2945	2947(3)	2959	2962	2964(3)
2979(2)	2987	2990	2992(3)	3004	3007
3009(3)	3064	3075	3088	3089	3090
3091	3095	3096	3097	3102	3103
3105	3107	3115(2)	3191	3265(2)	3275(3)
3279	3283	3284	3285	3287	3293
3294	3295	3308(2)	3309(2)	3311(2)	3312(2)
3321(2)	3322(2)	3324(2)	3325(2)	3334(2)	3335(2)
3337(2)	3338(2)	3347(2)	3348(2)	3350(2)	3351(2)
3368	3384	3389	3391	3393	3395
3405	3407	3409	3410	3428	3438
3440	3442	3443	3444	3460	3469
3470	3472	3474	3476	3478	3481
3483	3485	3487	3574(2)	3575(2)	3583(2)
3586(2)	3604(2)	3610	3611	3614(4)	3615(4)
3618(2)	3626	3627	3630(4)	3631(4)	3637(2)
3639(2)	3644	3647	3648(3)	3649	3649
3650	3652	3645(2)A	3657	3660(7)	3663(2)
3665(2)	3669	3655(3)	3677(6)	3686	3687
3690	3693	3675(2)	3703	3704	3707(7)
3710(2)	3711	3701(3)	3721(6)	3730	3731
3733	3735	3715	3721(6)	3745	3752(2)
3753(2)	3757(2)A	3743(3)	3759	3746	3752(2)
3770	3778	3780	3783	3764(3)	3768
3821(3)	3823	3824	3825(3)	3790	3796
3831(3)	3833	3834	3835(3)	3827(3)	3829(3)
3868	3877	3879	3880	3847	3850
3883	3884	3899	3903	3881	3882
3962	3965	3969	3972	3924	3924
3989	4001	4005	4011	3980	3983
4021	4027	4030	4036	4014	4018
4055	4057(2)	4063	4073	4043	4047
4101	4107	4111	4115	4092	4095
4130	4134	4142	4143	4118	4124
4156	4166	4173	4174	4144	4151(2)
				4175	4178(3)

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR:9

Page 65

2-000CE2DC	I*4	IS1	4180	4192	4198	4207(4)	4230(2)	4233(2)
2-000CE2E0	I*4	IS2	4236(2)	4257	4259	4260(2)	4262	4263
2-000CDC9C	I*4	ISATT	4265	4270	4280	4281(3)	4283	4284
			4285	4288(3)	4290	4291	4292	4296(2)
			4297(2)	4298(2)	4302(2)	4303	4305(2)	4323(2)
			4324(2)	4325(2)	4334	4336	4339	4345
			4356	4366(2)	4367(2)	4373(2)	4381	4383
			4391	4399	4408	4435	4436	4443
			4445(3)	4448	4452	4455	4457(2)	4460
			4465	4467	4483	4486	4490	4509
			4511	4521	4527	4538	4541(2)	4549(2)
			4574(3)	4575	4577(3)	4578	4580(2)	4583(2)
			4588	4591(2)	4593(2)	4642(2)=	4646(2)=	4649(2)=
			4652(2)=	4654(2)=	4656(2)=	4658(2)=	4660(2)=	4662(2)=
			4664(2)=	4666(2)=	4668(2)=	4670(2)=	4689=	4690
			4692	4710=	4713(3)	4718		
			4687=	4689	4695			
			4688=	4689	4695			
			1803=	1804	1814	2004	2923=	2924
			3094	3282	3645=	3646	3650	3652
			3655(3)	3660(3)	3665	3669	3675	3677(3)
			3693	3701(3)	3707(3)	3711	3715	3721(3)
			3757=	3758	3759	3819	3918	3958
			3974	4090	4217	4514		
			463					
4-0003BAB8	I*4	ISDAC	3524	3524	3546	1956	1957	1958
5-0002CC14	I*4	ISEFFD	1950=	1951	1955=	1965(2)	1977=	1978
2-000CDCE0	I*4	ISM	1959	1960	1964(2)	1989=	1990	2090=
			1979	1980	1981	2119(2)	2120(2)	2215=
			2091(2)	2092(2)	2118=	2221	2222	2223
			2218	2219	2220	2235(2)	2240	2241
			2224	2225	2230=	2254(2)	2262=	2267(2)
			2250	2251	2252(2)	2283	2284(2)	2286(2)
			2272	2273	2282	2569=	2570	2571
			2552=	2553	2554	2585	2587	2588
			2582=	2583(2)	2584	2602(2)	2603	2604
			2589(2)	2592(2)	2601=	2611(2)	2643=	2646(3)
			2606	2607	2608(2)	2659(3)	2680=	2681
			2648(3)	2654=	2657(3)	2759	2760	2761
			2683	2755=	2756	2770	2773	2776
			2765	2768	2769	2784	2785	2786
			2777	2778	2781	2847	2848	2851
			2842=	2843	2846	2915=	2917	2938=
			2854	2855	2856	2943	2944	2945(2)
			2939(2)	2940	2941	2957	2958	2960
			2947(3)	2955=	2956(2)	2983=	2984(2)	2985
			2961	2962(2)	2964(3)	2990(2)	2992(3)	3000=
			2986	2988	2989	3005	3006	3007(2)
			3001(2)	3002	3003	4180	4185=	4186
			3009(3)	4177=	4178(2)	4205=	4207(3)	
			4192	4194	4198			
			3536=	3537	3538	3540	3542(2)	3544
			3900=	3901	3923=	3925	3927	3929
			3931	3932(2)	3933(2)	3934(2)	3941(2)=	4441=
			4443	4445(3)	4448	4452	4455	4457(2)

2-000CE098 I*4 ISPT

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47

VAX FORTRAN V5.3-50

Page 66

C315:[CILLO.TACWAR30.IDA]GC.FOR:9

2-000CDC40 I.4 ISSM

4461	4466	4467	4468	4469(2)	4473(2)
4474(2)	4475(2)	4483(2)=	4495(2)=		
1648=	1649	1651(2)	1652(2)	1653	1654(2)
1655(2)	1656(3)	3201=	3202	3203	3205(3)
3208(3)	3243=	3244	3246=	3247	3328=
3330	3331	3332	3336=	3337(2)	3338(2)
3341=	3343	3344	3345	3349=	3350(2)
3351(2)	3393(2)=	3395(2)=	3629=	3630(2)	3631(2)
3968=	3969(2)	4000=	4001(3)	4003	4017=
4018(2)	4042=	4043(3)	4045	4069=	4070(2)
4071	4106=	4107(3)	4109	4129=	4130(3)
4132	4161=	4162(2)	4235=	4236(3)	4454=
4455(2)					
2708=	2714	2715	2720	2736=	2742
2743	2748	2796=	2802	2803	2808
2823=	2829	2830	2835	2865=	2871
2872	2877	2892=	2898	2899	2904
3358=	3364	3366	3368	3375=	3381
3383	3384				
1092					
673					

10-0001EF74 I.4 ISUM_COLLAPSED
5-0002CCF8 I.4 ISVHMCOMM
COMM

2-000CDCB4 I.4 IT

1828=	1832	1868(2)	3413=	3417	3455(2)
3457(2)	3535	3538	3544	3590	3907=
3911	3916	3925	3927	3929	4494
674	1769				

COMM

5-0002CCFC I.4 ITA

2-000CDC3C I.4 ITAR

8-000014D4 I.4 ITM

8-000014D8 I.4 ITWSTT

1647=

1655(2)

1656(2)

1656(2)

13-00037A58 I.4 IUSFRC

5-0002D660 I.4 IUTAM

2-000CDC60 I.4 IW

COMM
COMM

1205					
676					
1680=	1681(2)	1683(2)	1689=	1690	1691
1700=	1701(2)	1703(2)	1709=	1710	1711
1748=	1749	1750	1818=	1819	1820
1865=	1867(3)	1868	1869(3)	1883=	1884
1885(3)	1886(3)	1887	1888(3)	1889(3)	1900=
1901	1902	1903	1904(3)	1949=	1951
1962=	1963	1965	1980=	1981(2)	2030=
2031	2077=	2078	2082	2085	2088
2089	2091	2092	2096	2097	2100
2101	2105=	2106	2110	2113	2116
2117	2119	2120	2124	2125	2128
2129	2134=	2136(2)	2137(2)	2149=	2150
2151	2152(2)	2153(2)	2156=	2157(3)	2161=
2162(3)	2216=	2218	2219	2220	2221
2222	2223	2224	2225	2251=	2252(3)
2254(3)	2283=	2284(3)	2286(3)	2544=	2546
2547	2550=	2553(3)	2554(3)	2556(3)	2557(3)
2561=	2563	2564	2567=	2570(3)	2571(3)
2573(3)	2574(3)	2588=	2589(2)	2592(2)	2607=
2608(2)	2611(2)	2625=	2627(2)	2628(2)	2632=
2634(2)	2635(2)	2644=	2646(3)	2648(3)	2655=
2657(3)	2659(3)	2670=	2672(3)	2673(3)	2676=
2678	2679	2681(3)	2683(3)	2692=	2694(3)
2695(3)	2715(2)=	2720(2)=	2743(2)=	2748(2)=	2761(2)=
2770(2)=	2778(2)=	2786(2)=	2803(2)=	2808(2)=	2830(2)=

2835(2)=	2848(2)=	2856(2)=	2872(2)=	2877(2)=	2899(2)=
2904(2)=	2928=	2930(2)	2931	2944=	2945(2)
2947	2961=	2962(2)	2964	2976=	2978(2)
2979	2989=	2990(2)	2992	3006=	3007(2)
3009	3022=	3025(2)	3026(2)	3033=	3036(2)
3037(2)	3126=	3128(2)	3129(2)	3132=	3134(2)
3135(2)	3147=	3148	3149	3150(2)	3151(2)
3154=	3155(3)	3159=	3160(3)	3167=	3168
3169	3170(2)	3171(2)	3174=	3175(3)	3179=
3180(3)	3187=	3189	3190	3193(4)	3196(4)
3202	3203	3205(4)	3208(4)	3226(2)=	3229(2)=
3232(2)=	3235(2)=	3238(2)=	3241(2)=	3244(2)=	3247(2)=
3258=	3260	3261	3262	3263	3268=
3271	3271	3270	3273	3316=	3317
3318	3319	3342=	3343	3344	3345
3366(2)=	3368(2)=	3383(2)=	3384(2)=	3437=	3438(2)
3440(2)	3442(2)	3443(2)	3455	3456	3457
3458	3459	3460	3468=	3469(2)	3470(2)
3471	3472(2)	3474(2)	3476(2)	3478(2)	3480
3481(2)	3483(2)	3485(2)	3487(2)	3959=	3980
3962(3)	3965(3)	3969(3)	3972(2)	3979=	3988
3981	3983(2)	3985(3)	3987	3989(2)	3996(2)
3997(3)	3999	4001(2)	4003(3)	4005(4)	4008=
4009	4011(3)	4014(3)	4018(3)	4021(2)	4026=
4027	4028	4030(2)	4032(3)	4034	4036(2)
4038(2)	4039(3)	4041	4043(2)	4045(3)	4047(4)
4054=	4055(2)	4056	4057(2)	4059(2)	4060(3)
4062	4063(2)	4065(2)	4066(3)	4068	4070(2)
4071(3)	4073(4)	4091=	4092	4093	4095(2)
4097(3)	4099	4101(2)	4103(3)	4105	4107(2)
4109(3)	4111(4)	4114=	4115	4116	4118(2)
4120(3)	4122	4124(2)	4126(3)	4128	4130(2)
4132(3)	4134(4)	4141=	4142(2)	4143(2)	4144(2)
4146(2)	4148(2)	4151(2)	4153(2)	4156(2)	4158(2)
4158(2)	4160(2)	4162(2)	4164(2)	4166(4)	4189(2)=
4192(2)=	4228=	4230(3)	4233(3)	4236(3)	4317=
4318(2)	4319(2)	4320(2)	4347=	4348	4349
4350	4351	4352	4356(2)	4364	4371(2)
4372(2)	4373(2)	4390=	4391(3)	4399(2)=	4408(2)=
4418(2)=	4422(2)=	4444=	4445(2)	4495(2)=	4529=
4530	4531	4532	4533	4534	4538(2)
4539	4541(3)	4543(3)	4545(3)	4549(3)	4563(2)=
4582=	4583(2)	4592=	4593(2)	4584=	4685
4691=	4692(3)	4697=	4698(2)		
3074=	3075	3076			
3436=	3437	3467=	3468		
1981(2)=	2761(2)=	2770(2)=	2778(2)=	2786(2)=	2848(2)=
2856(2)=	4189(2)=	4198(2)=			
1890=	1881	1882	1883	3465=	3466
3467	3468				
2702(2)	2715(2)=	2720(2)=	2732(2)=	2743(2)=	2748(2)=
2803(2)=	2808(2)=	2830(2)=	2835(2)=	2772(2)=	2877(2)=
2899(2)=	2904(2)=	3366(2)=	3383(2)=		
2029=	2030				
3063=	3064	3069	3072	3076=	3088
3089	3265A	3275A			

Model	WEP	WIC	WORD	WPC	WRD	WTGT	WUCEE
2-000CDF34	1.4						
2-000CE058	1.4						
2-000CD00C			1.4				
2-000C0CC				1.4			
2-000CDE24					1.4		
2-000CD01C						1.4	
2-000CDF2C							1.4

2-0000DCB0	I 04	JDS	1827=	1828	1836	1867	1869	1885
5-0002DFC4	I 04	JESCFG	1886	1888	1889			
8-000014DC	I 04	JIN	678					
9-00009C1B	CHAR	JNAMEF	951					
5-0002DFC8	I 04	JNSW	1033	1072				
			679					
8-000014E0	I 04	JOUT	952					
2-0000CDD2C	I 04	JW	2087=	2088(2)	2089(2)	2091(2)	2092(2)	2115=
			2116(2)	2117(2)	2119(2)	2120(2)	2135=	2136(2)
			2137(2)	2146=	2150	2151	2152	2153
			2157(2)	2162(2)	2172=	2173(2)	2174(2)	2217=
			2218	2219	2220	2221	2222	2223
			2224	2225	2241=	2245	2252(3)	2254(3)
			2273=	2277	2284(3)	2286(3)	2545=	2546
			2547	2551=	2553(3)	2554(3)	2556(3)	2557(3)
			2562=	2563	2564	2568=	2570(3)	2571(3)
			2573(3)	2574(3)	2585=	2586	2589(3)	2592(3)
			2604=	2605	2608(3)	2611(3)	2626=	2627(3)
			2628(3)	2633=	2634(3)	2635(3)	2645=	2646(3)
			2648(3)	2656=	2657(3)	2659(3)	2671=	2672(3)
			2673(3)	2677=	2678	2679	2681(3)	2683(3)
			2693=	2694(3)	2695(3)	2719=	2720(2)	2747=
			2748(2)	2760=	2761(2)	2769=	2770(2)	2777=
			2778(2)	2785=	2786(2)	2807=	2808(2)	2834=
			2835(2)	2847=	2848(2)	2855=	2856(2)	2876=
			2877(2)	2903=	2904(2)	2916=	2917	2920=
			2921	2929=	2930(3)	2931(3)	2941=	2942
			2945(3)	2947(3)	2958=	2959	2962(3)	2964(3)
			2977=	2978(3)	2979(3)	2986=	2987	2990(3)
			2992(3)	3003=	3004	3007(3)	3009(3)	3024=
			3025(2)	3026(2)	3035=	3036(2)	3037(2)	3259=
			3260	3261	3262	3263	3269=	3270
			3271	3272	3273	3303=	3304	3305
			3306	3329=	3330	3331	3332	4176=
			4178(2)	4191=	4192	4197=	4198(2)	4206=
			4207(2)					
2-0000CDF14	I 04	JWIC	3023=	3024	3034=	3035		
4-0003BABC	I 04	K	464	1672=	1673	1678	1681	1683
			1698	1701	1703	1719	1721	1722
			1725	1726	1921=	1922	1925	1926
			1927	1928	4437=	4438	4439	4440
			4445(2)	4452(2)	4455(2)	4571=	4573	4574(3)
			4575	4577(3)	4578	4585		
			1508					
17-0000B144	I 04	KDVSUB	COMM					
8-000014E4	I 04	KFLAG	953					
8-000014E8	I 04	KISS	954					
5-0002E1DC	I 04	KOCNUC	684					
2-0000CDD5C	I 04	KP	1679=	1681(2)	1683(2)	1688=	1690	1691
			1699=	1701(2)	1703(2)	1708=	1710	1711
			1810=	1813	2171=	2173(2)	2174(2)	3644=
			3665	3675	3681	3711	3725	4516(3)
			4517(2)	4518				
			2006=	2008=	2088	2116	2589(2)	2608(2)
2-0000CDD18	I 04	KPA	2627(2)	2634(2)	2930	2945	2962	2962

18-Sep-1990 15:02:12 VAX FORTRAN V5.3-50
4-Apr-1990 13:59:47 C315:[CILLO.TACWAR30.IDA]GC.FOR:9

2-000CE0F0	I-4	KPCAS	3848-	3850-	3855	3858	3925	3927
			4095	4101	4107	4118	4124	4130
2-000CD014	I-4	KPD	2005-	2009-	2089	2117	2592(2)	2611(2)
			2628(2)	2635(2)	2978	2990	3007	3026
5-0002E460	I-4	KPECY	3037					
5-0002E404	I-4	KPECYD	689					
2-000CDCA0	I-4	KPI	690					
		COMM	1813-	1814-	1868	3405-	3457	
8-000014EC	I-4	KPMXA	955					
8-000014F0	I-4	KPMXAA	956					
8-000014F4	I-4	KPMXC	957					
8-000014F8	I-4	KPMXCA	958					
8-000014FC	I-4	KPMXG	959					
8-00001500	I-4	KPMXGA	960					
8-00001504	I-4	KPMXI	961					
8-00001508	I-4	KPMXN	962					
8-0000150C	I-4	KPMXNA	963					
5-0002FD7C	I-4	KSTOPB	714					
		COMM	1923	3743				
2-000CDC98	I-4	KTI	1797-	1868	3025	3026	3036	3037
			3455	3457				
2-000CE120	I-4	KW	3981-	3982(2)	3982-	3983(3)	3985	4010-
			4011(2)	4029-	4030(3)	4032	4058-	4059(2)
			4060	4094-	4095(3)	4097	4117-	4118(3)
			4120	4145-	4146(2)	4152-	4153(2)	4229-
			4230(3)	4417-	4418			
			1677-	1678	1681(3)	1683(3)	1690	1691
			1697-	1698	1701(3)	1703(3)	1710	1711
			1718-	1719	1721(3)	1722(3)	1725(3)	1726(3)
			1730	1731	1734	1735		
			465	1620-	1621	1622	1623	1646-
			1647	1648	1649	1651(2)	1652(2)	1653
			1654(2)	1656	1670-	1671	1672	1674
			1675	1676	1721(2)	1722(2)	1725(2)	1726(2)
			1730	1731	1734	1735	1744-	1745
			1746	1747	1749	1750	1757-	1758(2)
			1759(2)	1760	1761	1764-	1765	1766
			1767	1774	1776	1785(4)-	1812-	1814
			1816	1817	1819	1820	1822	1837(2)
			1865	1866	1867(2)	1869(2)	1881	1882
			1883	1884	1885(2)	1886(2)	1887	1888(2)
			1889(2)	1897-	1898	1899	1901(3)	1902
			1903	1904(3)	1906	1907	1910-	1911
			1912	1913	1914	1915	1916	1919-
			1920	1921	1942-	1943	1944	1961
			1962	1963	2027	2028	2029	2030
			2031(3)	2078	2081	2082(2)	2106	2109
			2110(2)	2152	2153	2157	2162	2163
			2155(2)	2727	2743(2)	2790	2803	2814
			2830	2860	2872	2883	2899	3120-
			3121	3122	3150	3151	3155	3160
			3170	3171	3175	3180	3217-	3218
			3219	3224	3227	3230	3233	3236
			3239	3242	3245	3354-	3355	3365
2-000CDC50	I-4	KWGT						
4-0003BAC0	I-4	L						

1--Sep-1990 15:02:12

VAX FORTRAN V5.3-50

Page 70

4-Apr-1990 13:59:47 C315:[CILLO.TACWAR30.IDA]GC.FOR:9

3366	3368	3372	3382	3383	3384
3387	3388	3389	3390	3391	3392
3393(2)	3394	3395(2)	3406-	3407	3409
3410	3429(2)	3436	3437	3438	3440
3442	3443	3459	3460	3466	3467
3468	3469	3470	3471	3472	3474
3476	3478	3480	3481	3483	3485
3487	3542(3)	3574(2)	3575(2)	3583(2)	3586(2)
3604(3)-	3609-	3610	3611	3612	3614(4)
3615(4)	3618(2)	3625-	3626	3627	3628
3630(4)	3631(4)	3637(3)-	3639(3)-	3777-	3778
3780	3790	3796	3819-	3820	3827(3)
3831(3)	3833	3834	3835(3)	3852	3855
3865(2)	3867	3868(3)	3869(3)	3879	3881(2)
3883(2)	3890(4)-	3897-	3899	3901	3903
3918	3924	3933(2)	3941(2)-	3946-	3947
3948	3949	3958	3972	3975(3)	3976(2)
3978	3980	3983	3989	4001	4005
4021	4023(3)	4024(2)	4025	4027	4030
4036	4043	4047	4055(2)	4057(2)	4063(2)
4073	4090	4092	4095	4101	4107
4111	4115	4118	4124	4130	4134
4142(2)	4143	4144	4151(2)	4156(2)	4166
4189	4217	4218	4221	4255	4262
4263	4271(2)	4272	4273(2)	4296(2)	4297(2)
4298(2)	4302(2)	4303(3)	4318	4319	4320
4323(2)	4324(2)	4325(2)	4333-	4334	4336
4337	4346(2)	4348	4349	4351	4356(2)
4364	4367(2)	4381(2)-	4388-	4389	4391(3)
4396-	4397	4398	4399	4402	4411-
4412	4413	4416	4420	4433-	4435
4436	4437	4443	4445(2)	4448	4457(2)
4467	4474(2)	4475(2)	4483(2)-	4485-	4486
4487	4488	4508-	4509	4511	4512
4514	4528(2)	4530	4531	4533	4538(2)
4539	4541	4543	4545	4549	4560-
4561-	4562	4570-	4571	4580(2)	4581
4583(2)	4591(2)	4593(2)	4599-	4600	4601(2)
4604(3)	4648-	4649	4652	4654	4656
4658	4660	4662	4664	4666	4668
4670	4678-	4679	4681	4690	4692
4708-	4709	4711	4712	4713(3)	4718(3)
2706-	2723(2)-	2730	2733-	2750(2)-	2792-
2810(2)-	2817	2820-	2837(2)-	2862-	2879(2)-
2886	2889-	2906(2)-			
4680-	4682				
4677-	4680	4681(2)-	4682		
1510					
1207	3265A	3275A	1663A	1664A	1781
964	1635	1662A	1981	1987	1990
1784	1785	1974	2732	2743	2748
2702	2715	2720	2730	2773	2778
2756	2761	2765	2770	2773	2778
2781	2786	2793	2803	2808	2819
2830	2835	2843	2848	2851	2856
2863	2872	2877	2888	2899	2904

2-000CDE30 I.4 LINCNT

2-000CE2CC I.4 M1
2-000CE2C8 I.4 M2
17-0000BC38 I.4 MAPOUT
13-00037AB8 I.4 MNIE
8-00001510 I.4 MOT

COMM
COMM
COMM

[illegible]

[illegible]

2-000CE278	I*4	NX	4581=	4582	4592	4679=	4684	4691
2-000CE104	R*4	PCFR	4697	3867	3875=	3876	3888	
2-000CE070	R*4	PEA	3499=	3500(2)=	3517	3550	3555	3563
			3597					
2-000CE074	R*4	PED	3503=	3504(2)=	3518	3551	3556	3564
			3597					
2-000CE1F0	R*4	PLSA	4315=	4319(2)=	4324			
2-000CE1EC	R*4	PLSG	4314=	4318(2)=	4323			
2-000CE1F4	R*4	PLSM	4316=	4320(2)=	4325			
2-000CE0C0	R*4	PPESE	3775=	3795(2)=	3796	3802	3867	3876
2-000CE06C	R*4	PPS	3496=	3497	3501	3562		
2-000CDD28	R*4	PWIC	2079=	2082=	2088	2091	2091	2092
			2096	2097	2100	2101	2107=	2110=
			2116	2117	2119	2120	2124	2125
			2128	2129				
COMM								
2-000CE250	R*4	RRR	4466=	4467	4473	4474	4475	
9-00009C21	CHAR	RUNNAM	1034	1073				
2-000CE090	R*4	SEF	3525=	3544(2)=	3547	3548	3550	3551
			3554=	3571	3597	3781=	3788(2)=	3790(2)=
			3795					
2-000CE284	R*4	SUM	4586=	4588(2)=	4590	4591	4593	
2-000CDD060	R*4	SUMA	2147=	2152(2)=	2155	2157	3145=	3150(2)=
			3153	3155	3165=	3170(2)=	3173	3175
2-000CDD064	R*4	SUMD	2148=	2153(2)=	2160	2162	3146=	3151(2)=
			3158	3160	3166=	3171(2)=	3178	3180
2-000CE0A8	R*4	TEMP	3543=	3544	3821=	3822	3823	3824
			3827	3917=	3929	3953=	3972(2)=	3975
			3978	3983	3989	4001	4021(2)=	4023
			4025	4030	4036	4043	4272=	4283
			4284	4285	4290	4291	4292	4299=
			4300(2)=	4301	4302	4305	4306	4356=
			4357	4467=	4468	4472(2)	4538=	4539
			4543	4572=	4575=	4578=	4580	4583
			3522=	3537(2)=	3570	3829=	3830	3831
			3833	3834	3929=	3931(2)=	3932	3933
			3934	4281=	4282	4283	4284	4285
			4288=	4289	4290	4291	4292	4357=
			4358(2)=	4364	4371	4373	4449=	4452(2)=
			4457	4474	4539=	4541	4545	4549
			4574=	4575(2)	4577=	4578(2)		
			3523=	3539(2)=	3570(2)	3919=	3921=	3929
			4450=	4455(2)=	4457	4475		
			3925=	3927=	3929			
2-000CE08C	R*4	TEMP2	4278=	4284=	4291=	4295(2)=	4297	4299
2-000CE114	R*4	TEMP3	4468=	4469	4472			
2-000CE1E4	R*4	TEMPA	4277=	4283=	4290=	4296	4299	
2-000CE1E0	R*4	TEMPG	4279=	4285=	4292=	4298	4299	
2-000CE1E8	R*4	TEMPM	4364=	4365	4366	4367	4368	
2-000CE208	R*4	TEMPPP	3677=	3680(2)=	3681	3687	3721=	3724(2)=
2-000CE08C	R*4	TMPPRD	3725	3731				

2-000CE2EC	R*4	TOT
2-000CE1AC	R*4	TSMSTK
2-000CE244	R*4	TSS
2-000CE234	R*4	TT1
4-00041018	R*4	UBABAF
COMM		
4-0004101C	R*4	UBABAR
4-00041020	R*4	UBABAZ
2-000CE0E8	R*4	VAFPA
2-000CE0E0	R*4	VAFPD
2-000CE0DC	R*4	VAI
2-000CE0D8	R*4	VDI
2-000CDC30	R*4	VFACTOR
COMM		
2-000CE0EC	R*4	VMFPA
2-000CE0E4	R*4	VMFPD
2-000CE078	R*4	WATS
5-0004CD64	R*4	WDIVR
2-000CE07C	R*4	WDTS
2-000CE080	R*4	WEA
2-000CE084	R*4	WED
2-000CE204	R*4	WPSOH
COMM		
2-000CDC8B	R*4	WTFIT
2-000CE060	R*4	WTFITA
2-000CE064	R*4	WTFITD
2-000CE048	R*4	WVDACI
2-000CE050	R*4	WVDAT
2-000CE04C	R*4	WVDDCI
2-000CE054	R*4	WVDDT
2-000CDE48	R*4	XNLIN

ARRAYS

Address	Type	Name	Attributes	Bytes	Dimensions	References
4-00000004	R*4	AA	COMM	100	(25)	341
3-00000000	R*4	AAAAG	COMM	4	(1)	254
5-00000000	R*4	AAABLK	COMM	4	(1)	540
8-00000000	R*4	AAABLW	COMM	4	(1)	933
6-00000000	R*4	AAABRS	COMM	4	(1)	854
7-00000000	R*4	AAABSV	COMM	4	(1)	910
9-00000000	CHAR	AAACH	COMM	4	(1)	987
9-00000004	CHAR	AAACT	COMM	48	(3, 2, 2)	1038
13-00000000	R*4	AAAGRD	COMM	4	(1)	1164
14-00000000	R*4	AAAGRS	COMM	4	(1)	1255
4-00000000	R*4	AAAGWK	COMM	4	(1)	340
16-00000000	R*4	AAASUP	COMM	4	(1)	1292
17-00000000	R*4	AAATC	COMM	4	(1)	1452
5-00000004	R*4	AAGDS	COMM	240	(3, 10, 2)	541

5-000000F4	R*4	AACS	COMM	240 (3, 10, 2)	542	
9-00000034	CHAR	AACT	COMM	400 (25, 2, 2)	989	1039
9-000001C4	CHAR	AACT	COMM	32 (2, 2, 2)	990	1040
9-000001E4	CHAR	AAFS	COMM	64 (4, 2, 2)	991	1041
2-00000000	R*4	AAMA		5520 (30, 23, 2)	1552	3148=
				3155=	3193	3226
2-00001590	R*4	AAMD		5520 (30, 23, 2)	1552	3129=
				3160=	3229	3149=
2-00002B20	R*4	AAMA		736 (4, 23, 2)	1553	3134=
				3175=	3238	3168=
2-00002E00	R*4	AAMD		736 (4, 23, 2)	1553	3135=
				3180=	3241	3169=
9-00000224	CHAR	AAMT	COMM	480 (30, 2, 2)	992	1042
9-00000404	CHAR	AAST	COMM	64 (4, 2, 2)	993	1043
2-000030E0	R*4	AAWA		4232 (23, 23, 2)	1554	2136=
				2157=	2284	2150=
				2672	2930	2556(2)=
						2573(2)=
2-00004168	R*4	AAMD		4232 (23, 23, 2)	1554	2137=
				2162=	2286	2151=
				2673	2978	2557(2)=
						2574(2)=
4-00000068	R*4	ABAAFA	COMM	2000 (25, 10, 2)	342	
4-00000838	R*4	ABAAFD	COMM	2000 (25, 10, 2)	343	
4-00001008	R*4	ABAARA	COMM	2000 (25, 10, 2)	344	
4-000017D8	R*4	ABAARD	COMM	2000 (25, 10, 2)	345	
4-00001FA8	R*4	ABAAZA	COMM	2000 (25, 10, 2)	346	
4-00002778	R*4	ABAAZD	COMM	2000 (25, 10, 2)	347	
4-00002F48	R*4	ABADFA	COMM	2000 (25, 10, 2)	348	
4-00003718	R*4	ABADFANT	COMM	2000 (25, 10, 2)	349	
4-00003EE8	R*4	ABADED	COMM	2000 (25, 10, 2)	350	
4-000046B8	R*4	ABADFDMT	COMM	2000 (25, 10, 2)	351	
4-00004E88	R*4	ABADFH	COMM	2000 (25, 10, 2)	352	
4-00005658	R*4	ABADFKMT	COMM	2000 (25, 10, 2)	353	
4-00005E28	R*4	ABADKA	COMM	2000 (25, 10, 2)	354	
4-000065F8	R*4	ABADRAWT	COMM	2000 (25, 10, 2)	355	
4-00006DC8	R*4	ABADRD	COMM	2000 (25, 10, 2)	356	
4-00007598	R*4	ABADRDWT	COMM	2000 (25, 10, 2)	357	
4-00007D68	R*4	ABADRH	COMM	2000 (25, 10, 2)	358	
4-00008538	R*4	ABADRKWT	COMM	2000 (25, 10, 2)	359	
4-00008D08	R*4	AGADZA	COMM	2000 (25, 10, 2)	360	
4-000094D8	R*4	ABADZAWT	COMM	2000 (25, 10, 2)	361	
4-00009CAB	R*4	ABADZD	COMM	2000 (25, 10, 2)	362	
4-0000A478	R*4	ABADZDWT	COMM	2000 (25, 10, 2)	363	
4-0000AC48	R*4	ABADZH	COMM	2000 (25, 10, 2)	364	
4-0000B418	R*4	ABADZKWT	COMM	2000 (25, 10, 2)	365	
4-0000BBE8	R*4	ABAEFA	COMM	2000 (25, 10, 2)	366	

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR:9

4-0000C3B8	R*4	ABAEAWT	COMM	2000	(25, 10, 2)	367
4-0000C8B8	R*4	ABAEFD	COMM	2000	(25, 10, 2)	368
4-0000D358	R*4	ABAEFDWT	COMM	2000	(25, 10, 2)	369
4-0000D828	R*4	ABAEFH	COMM	2000	(25, 10, 2)	370
4-0000E2F8	R*4	ABAEFKWT	COMM	2000	(25, 10, 2)	371
4-0000EAC8	R*4	ABAERA	COMM	2000	(25, 10, 2)	372
4-0000F298	R*4	ABAERAWT	COMM	2000	(25, 10, 2)	373
4-0000FA68	R*4	ABAERD	COMM	2000	(25, 10, 2)	374
4-00010238	R*4	ABAERDWT	COMM	2000	(25, 10, 2)	375
4-00010A08	R*4	ABAERH	COMM	2000	(25, 10, 2)	376
4-000111D8	R*4	ABAERKWT	COMM	2000	(25, 10, 2)	377
4-000119A8	R*4	ABAEZA	COMM	2000	(25, 10, 2)	378
4-00012178	R*4	ABAEZAWT	COMM	2000	(25, 10, 2)	379
4-00012948	R*4	ABAEZD	COMM	2000	(25, 10, 2)	380
4-00013118	R*4	ABAEZDWT	COMM	2000	(25, 10, 2)	381
4-000138E8	R*4	ABAEZH	COMM	2000	(25, 10, 2)	382
4-00014088	R*4	ABAEZKWT	COMM	2000	(25, 10, 2)	383
4-00014888	R*4	ABASFA	COMM	2000	(25, 10, 2)	384
4-00015058	R*4	ABASFD	COMM	2000	(25, 10, 2)	385
4-00015828	R*4	ABASRA	COMM	2000	(25, 10, 2)	386
4-00015FF8	R*4	ABASRD	COMM	2000	(25, 10, 2)	387
4-000167C8	R*4	ABASZA	COMM	2000	(25, 10, 2)	388
4-00016F98	R*4	ABASZD	COMM	2000	(25, 10, 2)	389
4-00017768	R*4	ABATEF	COMM	80	(10, 2)	390
4-000177B8	R*4	ABATER	COMM	80	(10, 2)	391
4-00017808	R*4	ABATEZ	COMM	80	(10, 2)	392
3-00000004	R*4	ABATPS	COMM	200	(25, 2)	255
3-000000CC	R*4	ABMTPS	COMM	48	(6, 2)	256
9-00000044	CHAR	ABNAME	COMM	6160	(7, 220)	1044
9-00001C54	CHAR	ACAP0D	COMM	480	(30, 2, 2)	1045
4-00017858	R*4	ACCZDC	COMM	2000	(25, 10, 2)	393
4-00018028	R*4	ACCZKC	COMM	2000	(25, 10, 2)	394
4-000187F8	R*4	ACCZT	COMM	2000	(25, 10, 2)	395
9-00001E34	CHAR	ACDPDS	COMM	320	(20, 2, 2)	1046
9-00001F74	CHAR	ACDPDS	COMM	320	(20, 2, 2)	1047
9-000020B4	CHAR?	ACFCT	COMM	240	(15, 2, 2)	1048
4-00018FC8	R*4	ACFSDC	COMM	2000	(25, 10, 2)	396
4-00019798	R*4	ACFSKC	COMM	2000	(25, 10, 2)	397
4-00019F68	R*4	ACFST	COMM	2000	(25, 10, 2)	398
9-000021A4	CHAR	ACHAT	COMM	36	(9)	1049
9-000021C8	CHAR	ACHDM	COMM	12	(3)	1050
9-000021D4	CHAR	ACISN	COMM	480	(30, 2, 2)	1052
9-000023B4	CHAR	ACPORT	COMM	160	(10, 2, 2)	1051
4-0001A738	R*4	ACRSDC	COMM	2000	(25, 10, 2)	399
4-0001AF08	R*4	ACRSKC	COMM	2000	(25, 10, 2)	400
4-0001B6D8	R*4	ACRST	COMM	2000	(25, 10, 2)	401
8-00000004	R*4	ACSABA	COMM	2000	(25, 10, 2)	934
				3618	3965	3614
						4014
						3615
						4036

4-0001BEA8	R*4	AD		4101	4124	4452	4588	4713
9-00002454	CHAR	ADT	100 (25)					
2-0000BC90	R*4	AERTA	256 (32, 2)			402		
2-0000CC3C0	R*4	AERTD	1840 (23, 10, 2)			1003	1053	
9-00002554	CHAR	AGST	1840 (23, 10, 2)			1605		
			80 (5, 2, 2)			1004	1054	
9-000025A4	CHAR	AGWT						
4-0001BF0C	R*4	AH	96 (12, 2)			1005	1055	
16-00000004	R*4	AIFCT	100 (25)			403		
16-00005DC4	R*4	AIRCAP	24000 (10, 600)			1293		
16-00006094	R*4	AIRPLE	720 (30, 2, 3)			1294		
			480 (2, 30, 2)			1295		
16-00006274	R*4	AIRPRQ						
9-00002604	CHAR	AISM	240 (30, 2)			1296		
9-00002744	CHAR	AIWS	320 (20, 2, 2)			1006	1056	
4-0001BF70	R*4	AK	96 (3, 4, 2)			1007	1057	
2-000051F0	R*4	AKGS	100 (25)			404		
			4600 (25, 23, 2)			1555	3989=	3996(2)=
			3997	4036=		4038(2)=	4039	4065(2)=
			4066	4101=		4103	4124=	4126
			4158(2)=	4233		4422		
9-000027A4	CHAR	AKT						
9-000027C4	CHAR	ALRS	32 (4, 2)			1008	1058	
5-000001E4	R*4	ALRSF	32 (2, 2, 2)			1009	1059	
4-0001BFD4	R*4	ALRSFA	160 (2, 10, 2)			543		
4-0001C024	R*4	ALRSFD	80 (2, 10)			405		
			80 (2, 10)			406		
4-0001C074	R*4	ALRSFS						
5-00000284	R*4	ALRSR	80 (2, 10)			407		
4-0001C0C4	R*4	ALRSRA	160 (2, 10, 2)			544		
4-0001C114	R*4	ALRSRD	80 (2, 10)			408		
4-0001C164	R*4	ALRSRS	80 (2, 10)			409		
			80 (2, 10)			410		
5-00000324	R*4	ALRSZ						
4-0001C184	R*4	ALRSZA	160 (2, 10, 2)			545		
4-0001C204	R*4	ALRSZD	80 (2, 10)			411		
4-0001C254	R*4	ALRSZS	80 (2, 10)			412		
16-00006364	R*4	ALTSPA	80 (2, 10)			413		
			2400 (10, 2, 30)			1297		
16-00006CC4	R*4	ALTSPD						
4-0001C2A4	R*4	AMLFD	2400 (10, 2, 30)			1298		
6-00000004	R*4	AMNL	2000 (25, 10, 2)			414	3191	
			36000 (25, 30, 12)			855	3193	3196
			4713	4718				
5-000003C4	R*4	AMNLAF						
17-00000004	R*4	AMOVR	2400 (25, 12, 2)			546	3191	
17-00000014	R*4	AMTIAM	16 (4)			1453		
17-00000104	R*4	AMTINT	240 (30, 2)			1454	4718(2)=	
5-00000024	R*4	AMTLRS	2000 (50, 10)			1455		
			16 (2, 2)			547		
5-00000034	R*4	AMTSRS						
3-000000FC	R*4	AMXRPL	32 (4, 2)			548		
9-000027E4	CHAR	ANIWS	8 (2)			257		
16-00007624	R*4	APDARA	216 (3, 9, 2)			1010	1060	
			2400 (30, 10, 2)			1299		

16-00007F84	R*4	APDARV	COMM	2400 (30, 10, 2)	1300		
3-00000104	R*4	APDCAA	COMM	6000 (25, 30, 2)	258		
3-00001874	R*4	APDCAM	COMM	1440 (6, 30, 2)	259		
5-00000054	R*4	APDDAM	COMM	240 (30, 2)	549		
16-000008E4	R*4	APRLD	COMM	2400 (30, 10, 2)	1301		
16-000009244	R*4	APRLDA	COMM	2400 (30, 10, 2)	1302		
16-000009BA4	R*4	APSPD	COMM	24000 (100, 30, 2)	1303		
9-0000028BC	CHAR	APT	COMM	48 (6, 2)	1061	2702	
				2732			
3-00001E14	R*4	APTINT	COMM	80 (10, 2)	260		
16-0000F964	R*4	APTTRK	COMM	480 (30, 2, 2)	1304		
16-0000FB44	R*4	ARPCAP	COMM	720 (30, 2, 3)	1305		
9-0000028EC	CHAR	ASIDE	COMM	8 (2)	1062	4649	
16-0000FE14	R*4	ASL	COMM	480 (6, 10, 2)	1306		
9-0000028F4	CHAR	ASRS	COMM	64 (4, 2, 2)	1013		
9-000002934	CHAR	ASSM	COMM	96 (6, 2, 2)	1014		
9-000002994	CHAR	ASUT	COMM	112 (7, 2, 2)	1015		
9-000002A04	CHAR	ATR	COMM	16 (2, 2)	1016		
5-00000E44	R*4	AVMP5M	COMM	48 (6, 2)	550		
9-000002A14	CHAR	AWT	COMM	368 (23, 2, 2)	1017	1981	
				2715	2748	2761	
				2770	2803	2808	
				2830	2856	2872	
				2877	3366	3383	
				4189			
5-00000E74	R*4	BALOC	COMM	6560 (4, 410)	551		
17-0000008D4	R*4	BAOFST	COMM	40 (10)	1456		
2-00019750	R*4	BAW5M		42320 (23, 23, 20)	1583	2218=	2252=
				2553	2761	2945	
2-000023CA0	R*4	BAW5M		42320 (23, 23, 20)	1583	2219=	2254=
				2554	2770	2990	
2-000A1EB0	R*4	BBFR5M		42320 (23, 23, 20)	1602	2091=	2252
2-000AC400	R*4	BBFR5M		42320 (23, 23, 20)	1602	2092=	2254
5-000002814	R*4	BCODE	COMM	2400 (600)	552		
5-000003174	R*4	BED	COMM	2400 (600)	553		
13-000000004	R*4	BFR5M	COMM	42320 (23, 23, 20)	1165	2091	
13-0000A554	R*4	BFR5M	COMM	42320 (23, 23, 20)	1166	2092	
4-0001CA74	R*4	BMRSA	COMM	80 (2, 10)	415		
4-0001CAC4	R*4	BMRSB	COMM	80 (2, 10)	416		
4-0001CB14	R*4	BMRSS	COMM	80 (2, 10)	417		
17-0000008FC	R*4	BNDIS	COMM	2800 (70, 10)	1457		
5-000003AD4	R*4	BPD	COMM	2400 (600)	554	3496	3535
				3916			
13-00014AA4	R*4	BPK5M	COMM	42320 (23, 23, 20)	1167	2589	
13-0001EFF4	R*4	BPKAW	COMM	12696 (23, 23, 6)	1168	2627	
13-00002218C	R*4	BPKDSM	COMM	42320 (23, 23, 20)	1169	2592	

4-000211B4	R*4	CASDKWT	COMM	2000 (25, 10, 2)	427	
4-000219B4	R*4	CASEA	COMM	2000 (25, 10, 2)	428	
4-00022154	R*4	CASEAWT	COMM	2000 (25, 10, 2)	429	
4-00022924	R*4	CASED	COMM	2000 (25, 10, 2)	430	
4-000230F4	R*4	CASEDWT	COMM	2000 (25, 10, 2)	431	
4-000238C4	R*4	CASEH	COMM	2000 (25, 10, 2)	432	
4-00024094	R*4	CASEKWT	COMM	2000 (25, 10, 2)	433	
4-00024864	R*4	CASSA	COMM	2000 (25, 10, 2)	434	
4-00025034	R*4	CASSD	COMM	2000 (25, 10, 2)	435	
3-00001F2C	R*4	CASSNM	COMM	8 (2)	262	
3-00001F34	R*4	CASSNM	COMM	240 (10, 3, 2)	263	
5-000125FC	R*4	CCAAPD	COMM	480 (2, 30, 2)	558	
5-000127DC	R*4	CCAPRT	COMM	160 (2, 10, 2)	559	
2-00006CE4	R*4	CDACS	COMM	80 (10, 2)	1556	1907= 4580=
				4591(2)= 4690		
6-0000917C	R*4	CDVLOC	COMM	2400 (600)	863	1836
				4270	4527	3428
				4345		
6-00009ADC	R*4	CFEBA	COMM	40 (10)	864	
5-0001287C	R*4	CKABSA	COMM	600 (25, 3, 2)	560	
5-00012AD4	R*4	CKABSM	COMM	144 (6, 3, 2)	561	
17-000013EC	R*4	CKPINT	COMM	4000 (50, 10, 2)	1458	
17-0000238C	R*4	CKPPHL	COMM	160 (2, 10, 2)	1459	
5-00012864	R*4	CKSACA	COMM	600 (25, 3, 2)	562	
5-00012DBC	R*4	CKSACM	COMM	144 (6, 3, 2)	563	
5-00012E4C	R*4	CKUACA	COMM	600 (25, 3, 2)	564	
5-000130A4	R*4	CKUACM	COMM	144 (6, 3, 2)	565	
5-00013134	R*4	CLAHEA	COMM	3600 (15, 10, 6)	566	
5-00013F44	R*4	CLAHEC	COMM	3600 (15, 10, 6)	567	
5-00014D54	R*4	CLMHEA	COMM	3600 (15, 10, 6)	568	
5-00015B64	R*4	CLMHEC	COMM	3600 (15, 10, 6)	569	
5-00016974	R*4	CLRSOM	COMM	16 (2, 2)	570	
5-00016984	R*4	CLRSRP	COMM	16 (2, 2)	571	
5-00016994	R*4	CNCWLS	COMM	368 (23, 4)	572	
16-000100E4	R*4	COPDHE	COMM	120 (15, 2)	1308	
5-00016B04	R*4	COPDIT	COMM	128 (32)	573	
5-00016B84	R*4	CPLAI	COMM	160 (10, 4)	574	
5-00016C24	R*4	CPLAIT	COMM	160 (10, 4)	575	
5-00016CC4	R*4	CPLARA	COMM	160 (10, 4)	576	
5-00016D64	R*4	CPLMI	COMM	160 (10, 4)	577	
5-00016E04	R*4	CPLMIT	COMM	160 (10, 4)	578	
5-00016EA4	R*4	CPLMRA	COMM	160 (10, 4)	579	
5-00016F44	R*4	CPLS	COMM	160 (10, 4)	580	
				4383		4305(2)= 4366(2)=
5-00016FE4	R*4	COKNSF	COMM	40 (5, 2)	581	
5-0001700C	R*4	COKNSR	COMM	40 (5, 2)	582	
5-00017034	R*4	COKNSZ	COMM	40 (5, 2)	583	
5-0001705C	R*4	COKSHF	COMM	40 (5, 2)	584	

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR;9

5-00017004	R*4	CKSHR	COM	40 (5, 2)	585	
5-000170AC	R*4	CKSHZ	COM	40 (5, 2)	586	
6-00009804	R*4	CSABAF	COM	200 (25, 2)	865	
6-0000980C	R*4	CSABAR	COM	200 (25, 2)	866	
6-000098C9	R*4	CSABAZ	COM	200 (25, 2)	867	
16-0001015C	I*4	CSACUS	COM	48000 (3, 40, 100)	1309	1439
16-0001BCDC	R*4	CSAJSW	COM	8 (2)	1310	
16-0001BCE4	R*4	CSATRK	COM	320 (20, 2, 2)	1311	
6-00009D5C	R*4	CSCASA	COM	200 (25, 2)	868	
13-00032C34	R*4	CSDAP	COM	7680 (32, 6, 10)	1173	3927
5-00017004	R*4	CSDAW	COM	160 (2, 10, 2)	587	4474(2)=
5-00017174	R*4	CSDDP	COM	7680 (32, 6, 10)	588	3925
5-00018F74	R*4	CSDFS	COM	320 (2, 10, 4)	589	
5-00019084	R*4	CSDGW	COM	160 (10, 4)	590	4473(2)=
5-00019154	R*4	CSDR	COM	2560 (2, 32, 10)	591	3929
5-00019854	R*4	CSDSW	COM	160 (2, 10, 2)	592	4475(2)=
16-0001BE24	R*4	CSHE	COM	4800 (30, 10, 4)	1312	
6-00009E24	R*4	CSINDA	COM	200 (25, 2)	869	
12-00000004	R*4	CSMENG	COM	1600 (20, 10, 2)	1129	4207(2)=
5-000198F4	R*4	CSFSDM	COM	32 (4, 2)	593	
5-00019C14	R*4	CSRSRP	COM	32 (4, 2)	594	
5-00019C34	R*4	CSSMAR	COM	6720 (5, 10, 28)	595	1656(2)=
5-0001B674	R*4	CSSMST	COM	6720 (5, 10, 28)	596	1655(2)=
5-0001D084	R*4	CTSCA	COM	8800 (10, 220)	597	
5-0001F314	R*4	CTSCD	COM	320 (2, 10, 4)	598	3934(2)=
6-00009EEC	R*4	CWLAI	COM	3680 (23, 10, 4)	600	
6-0000AD4C	R*4	CWLAIT	COM	3680 (23, 10, 4)	601	
6-000088AC	R*4	CWLARA	COM	3680 (23, 10, 4)	871	
5-0001F454	R*4	CWLMI	COM	3680 (23, 10, 4)	872	
5-00020284	R*4	CWLMIT	COM	3680 (23, 10, 4)	599	
5-00021114	R*4	CWLMRA	COM	3680 (23, 10, 4)	600	
5-00021F74	R*4	CWLS	COM	3680 (23, 10, 4)	602	4373(2)= 4408
4-00025804	R*4	DA	COM	100 (25)	436	
5-000220D4	R*4	DABOCC	COM	8 (2)	603	
2-000060J4	R*4	DAIR	COM	92 (23)	1556	3987= 3997(2)=
				4005	4034=	
				4066(2)=	4039(2)=	
				4122=	4073	4047 4062=
				4224A	4126(2)=	4103(2)= 4111
					4319	4160(2)= 4166
17-0000242C	R*4	DBDMF	COM	80 (10, 2)	1460	
4-00025868	R*4	DD	COM	100 (25)	437	
13-00034A34	R*4	DOWDCD	COM	8 (2)	1174	
5-000220DC	R*4	DEPDR	COM	8 (2)	604	
13-00034A3C	R*4	DEQ	COM	128 (32)	1175	
9-00002884	CHAR	DESC	COM	20 (5)	1018	1068
9-00002898	CHAR	DESIG	COM	16800 (7, 600)	1030	1069
16-0001D0E4	R*4	DFACSA	COM	160 (20, 2)	1313	
16-0001D184	R*4	DFADSA	COM	160 (20, 2)	1314	

17-0000247C	R*4	DFASN	240 (30, 2)	1461			
17-0000250C	R*4	DFBACT	8 (2)	1462	1837(2)	3429(2)	
			4271(2)	4346(2)	4528(2)		
2-00006090	R*4	DGRD	92 (23)	1557	3981=	3985(2)=	
			4005	4028=	4032(2)=	4047	4056=
			4060(2)=	4073	4093=	4097(2)=	4111
			4116=	4120(2)=	4134	4143	4144
			4148(2)=	4155(2)=	4166	4225A	4318
4-000258CC	R*4	DH	100 (25)	438			
17-00002574	R*4	D1S2EC	8 (2)	1463			
17-0000257C	R*4	D1S3EC	8 (2)	1464			
4-00025930	R*4	DK	100 (25)	439			
17-00002584	R*4	DLYMAX	4000 (50, 10, 2)	1465			
16-00010224	R*4	DPCDS	160 (20, 2)	1315			
16-000102C4	R*4	DPCSBA	160 (20, 2)	1316			
16-00010364	R*4	DPCSIS	160 (20, 2)	1317			
16-00010404	R*4	DPCSLN	320 (2, 20, 2)	1318			
16-00010544	R*4	DPCSTF	160 (20, 2)	1319			
16-000105E4	R*4	DPDDS	160 (20, 2)	1320			
16-00010604	R*4	DPDSBA	160 (20, 2)	1321			
16-00010724	R*4	DPDSIS	160 (20, 2)	1322			
16-000107C4	R*4	DPDSLN	320 (2, 20, 2)	1323			
16-00010904	R*4	DPDSTF	160 (20, 2)	1324			
16-000109A4	R*4	DSA	80 (10, 2)	1325			
16-000109F4	I*4	DSACUS	32000 (2, 40, 100)	1326	1440		
16-000256F4	R*4	DSAJSW	8 (2)	1327			
16-000256FC	R*4	DSATRK	320 (20, 2, 2)	1328			
16-0002583C	R*4	DSO	320 (2, 10, 4)	1329			
2-00006DEC	R*4	DSHAR	2400 (600)	1557	3569=	3570=	
			4630				
16-0002597C	R*4	DSHE	1200 (10, 15, 2)	1330			
16-00025E2C	R*4	DSM	480 (6, 10, 2)	1331			
2-0000774C	R*4	DSSM	92 (23)	1558	3999=	4003(2)=	
			4005	4041=	4045(2)=	4047	4058=
			4071(2)=	4073	4105=	4109(2)=	4111
			4128=	4132(2)=	4134	4164(2)=	4166
			4226A	4320			
16-0002600C	R*4	DSSF	320 (4, 10, 2)	1332			
17-00003524	R*4	DSTCKP	2000 (50, 10)	1466			
17-00003CF4	R*4	DSTPHL	16 (2, 2)	1467			
17-00003D04	R*4	DSULT	896 (7, 32)	1468			
5-00022DE4	R*4	DVASAM	4800 (2, 600)	605			
5-000240A4	R*4	DVDPTH	1200 (300)	606			
5-00024554	R*4	DVMDTH	1200 (300)	607			
5-00024A04	R*4	EAACP	1200 (15, 10, 2)	608			
5-00024EB4	R*4	EAEAC	1200 (15, 10, 2)	609			
17-00004004	R*4	EAETE	120 (15, 2)	1469			

18-Sep-1998 15:02:12
4-Apr-1998 13:59:47

VAX FORTRAN V5.3-50

Page 83

C315:[CILLO.TACMAR30.IDA]GC.FOR:9

5-00025364	R*4	EAPAC	1200 (15, 10, 2)	610	
5-00025814	R*4	EAPTE	120 (15, 2)	611	
5-0002588C	R*4	EATAC	1200 (15, 10, 2)	612	
17-000040FC	R*4	EATTE	120 (15, 2)	1470	
5-00025D3C	R*4	ECACP	1200 (15, 10, 2)	613	
5-000261EC	R*4	ECEAC	1200 (15, 10, 2)	614	
17-00004174	R*4	ECETE	120 (15, 2)	1471	
5-0002669C	R*4	ECAPC	1200 (15, 10, 2)	615	
5-00026B4C	R*4	ECPTC	120 (15, 2)	616	
5-00026B4C	R*4	ECTAC	1200 (15, 10, 2)	617	
17-000041EC	R*4	ECTTE	120 (15, 2)	1472	
3-00002024	R*4	EDADPB	240 (30, 2)	264	
3-00002114	R*4	EDADPT	240 (30, 2)	265	
3-00002204	R*4	EDADVB	5520 (30, 23, 2)	266	
3-00003794	R*4	EDADVH	10800 (30, 15, 6)	267	
3-000061C4	R*4	EDADVT	5520 (30, 23, 2)	268	
2-000077A8	R*4	EDAXSR	2400 (600)	1558	3517= 3547
2-00008108	R*4	EDDXSR	2400 (600)	1559	3518= 3548
3-00007754	R*4	EDMDVH	1440 (4, 15, 6)	269	
3-00007CF4	R*4	EDSDPB	32 (4, 2)	270	
3-00007D14	R*4	EDSDPT	32 (4, 2)	271	
3-00007D34	R*4	EDSDVB	736 (4, 23, 2)	272	
3-00008014	R*4	EDSDVT	736 (4, 23, 2)	273	
2-0000CB3F0	R*4	EERTA	1104 (23, 6, 2)	1604	2173= 2589
			2608	2634	2930 2945
			2962		
2-0000CB840	R*4	EERTD	1104 (23, 6, 2)	1604	2174= 2592
			2611	2635	2978 2990
			3007		
6-0000CA0C	R*4	EFFDA	2400 (600)	873	3547= 3550=
			3555=	3572	4630
6-0000D36C	R*4	EFFDD	2400 (600)	874	3548= 3551=
			3556=	3573	4630
5-00027074	R*4	EFFHEA	1200 (15, 10, 2)	618	
5-00027524	R*4	EFFHEC	1200 (15, 10, 2)	619	
10-0002614C	R*4	E1FCT	2400 (600)	1333	
17-00004264	R*4	ERDWLE	8 (2)	1473	
13-00034AC0	R*4	ERGMAX	184 (23, 2)	1177	1887
			3471	4349	4531
			4533		
13-00034878	R*4	ERTA	1104 (23, 6, 2)	1178	2173
13-00034FC8	R*4	ERTD	1104 (23, 6, 2)	1179	2174
3-000082F4	R*4	ESDAMI	480 (6, 10, 2)	274	

18-Sep-1990 15:02:12 VAX FORTRAN V5.3-50
4-Apr-1990 13:59:47 C315:[CILLO.TACWAR30.IDA]GC.FOR:9

3-00000404	R*4	ESDASI	2000	(25, 10, 2)	275				
10-000104C4	R*4	EXNPMN	4232	(23, 23, 2)	1087	2930=	2931		
			2978=	2979	4172A				
15-00000004	R*4	EXPMN	846400	(23, 23, 20, 10, 2)	2947	2962=	2964		
			1273	2945=	3007=	3009	4174A		
			2990=	2992					
			4192						
5-000279D4	R*4	FAACTD	128	(32)	620				
4-00025994	R*4	FAADFF	2000	(25, 10, 2)	440				
4-00026164	R*4	FAADRF	2000	(25, 10, 2)	441				
4-00026934	R*4	FAADRR	2000	(25, 10, 2)	442				
4-00027104	R*4	FAADZF	2000	(25, 10, 2)	443				
4-000278D4	R*4	FAADZR	2000	(25, 10, 2)	444				
4-000280A4	R*4	FAADZZ	2000	(25, 10, 2)	445				
5-00027A54	R*4	FABDST	80	(10, 2)	621				
4-00028874	R*4	FACDFC	20000	(25, 10, 20)	446				
4-0002D694	R*4	FACDRC	20000	(25, 10, 20)	447				
4-000324B4	R*4	FACDZC	20000	(25, 10, 20)	448				
3-00008CA4	R*4	FACTRG	600	(25, 3, 2)	276				
3-00008EFC	R*4	FAGSCN	200	(25, 2)	277				
16-00026AAC	R*4	FAPCSA	8	(2)	1334				
16-00026AB4	R*4	FAPDSA	8	(2)	1335				
3-00008FC4	R*4	FAUSHL	8	(2)	278				
3-00008FCC	R*4	FCAIA	200	(25, 2)	279				
13-00035424	R*4	FCMP	8	(2)	1183				
13-0003542C	R*4	FCVLS	8	(2)	1184	3869	3878		
5-00027AA4	R*4	FD0PTH	8	(2)	622				
4-000372D4	R*4	FDEAC	100	(25)	449				
4-00037338	R*4	FDEAF	100	(25)	450				
4-0003739C	R*4	FDEAR	100	(25)	451				
4-00037400	R*4	FDEAZ	100	(25)	452				
5-00027AAC	R*4	FDEFCP	2400	(600)	623	3559	3560		
			4630						
13-00035434	R*4	FD0DC	8	(2)	1185				
17-0000426C	R*4	FDVMT	1280	(32, 10)	1474				
16-00026ABC	R*4	FDVWSH	1200	(15, 10, 2)	1336				
17-0000476C	R*4	FDVST	1280	(32, 10)	1475	3544			
17-00004C6C	R*4	FEAFBA	8	(2)	1476				
6-0000DCCC	R*4	FEBA	40	(10)	875	1624	1626		
			1836	1923	3428	3743	4270		
			4345	4527					
5-0002840C	R*4	FEBATZ	40	(10)	624	1624	1626		
			1923	3743					
10-0001E54C	R*4	FEBDLY	80	(10, 2)	1088				
13-00035440	R*4	FEDW	8	(2)	1187				
5-00028434	R*4	FEFFDG	2400	(600)	625				
17-00004C74	R*4	FEISF	8	(2)	1477				

5-00028094	R*4	FHEWTG	COMAM	360 (3, 15, 2)	626		
13-00035448	R*4	FICAS	COMAM	16 (2, 2)	1188	3996	4038
				4295			
13-00035458	R*4	FMBP	COMAM	8 (2)	1189		
13-00035460	R*4	FMDPT	COMAM	3072 (32, 6, 4)	1190		
17-00004C7C	R*4	FMNC	COMAM	80 (10, 2)	1478		
17-00004CCC	R*4	FMOVR	COMAM	16 (4)	1479		
5-00028EFC	R*4	FMXE	COMAM	8 (2)	627		
5-00028F04	R*4	FMXEHE	COMAM	8 (2)	628		
13-00036060	R*4	FPCR	COMAM	16 (4)	1191	4306	4368
16-00026F6C	R*4	FPCSCP	COMAM	48 (6, 2)	1337		
16-00026F9C	R*4	FPCSPC	COMAM	40 (5, 2)	1338		
5-00028F0C	R*4	FPCDAS	COMAM	80 (20)	629	4683=	4690(2)=
				4696(2)=			
16-00026FC4	R*4	FPDSCP	COMAM	48 (6, 2)	1339		
16-00026FF4	R*4	FPDSPC	COMAM	40 (5, 2)	1340		
5-00026F5C	R*4	FPHEPC	COMAM	600 (15, 5, 2)	630		
13-00036070	R*4	FPKASM	COMAM	24 (6)	1192	2589	2608
8-000007D4	R*4	FPKC	COMAM	2400 (600)	935		
13-00036088	R*4	FPKDSM	COMAM	24 (6)	1193	2592	2611
3-00009094	R*4	FPKPOD	COMAM	8 (2)	280		
16-0002701C	R*4	FPPTCP	COMAM	48 (6, 2)	1341		
16-0002704C	R*4	FPPTPC	COMAM	40 (5, 2)	1342		
5-000291B4	R*4	FPSWDU	COMAM	56 (7, 2)	631		
3-0000909C	R*4	FORA	COMAM	120 (5, 3, 2)	281		
17-00004CDC	R*4	FRACIO	COMAM	8 (2)	1480		
7-00000004	R*4	FRAD	COMAM	40 (10)	911	1909=	3096=
				3107=	3283=	3660=	3663(2)=
				3665	3675	3707=	3710(2)=
				3711	3731=		
13-000360A0	R*4	FRATA	COMAM	48 (6, 2)	1194	3665	3675
				3711			
13-000360D0	R*4	FRATD	COMAM	48 (6, 2)	1195	3681	3725
2-00008A68	R*4	FRCAR	COMAM	80 (10, 2)	1559	3879=	3880=
				4666			
5-000291EC	R*4	FRDSUZ	COMAM	64 (4, 2, 2)	632		
5-0002922C	R*4	FRHECT	COMAM	2400 (4, 15, 10)	633		
13-00036100	R*4	FRRINA	COMAM	96 (6, 4)	1196	3025	3026
13-00036160	R*4	FRRING	COMAM	3072 (32, 6, 4)	1197	1868	3455
				3457			
13-000360D0	R*4	FRRINH	COMAM	96 (6, 4)	1198	3036	3037
17-00004CE4	R*4	FRRPHE	COMAM	368 (2, 23, 2)	1481		
16-00027074	R*4	FRSPGA	COMAM	160 (2, 10, 2)	1343		
16-00027114	R*4	FRSPTL	COMAM	640 (10, 4, 4)	1344		
16-00027394	R*4	FRSPTR	COMAM	640 (10, 4, 4)	1345		
3-00009114	R*4	FSKPOD	COMAM	80 (10, 2)	282		

18-Sep-1990 15:02:12
4-Apr-1990 13:59:47

Page 86

VAX FORTRAN V5.3-50
C315:[CILLO.TACWAR30.IDA]GC.FOR:9

3-00000164	R*4	FSKSAK	200 (25, 2)	283	
3-0000022C	R*4	FSKSAK	48 (6, 2)	284	
17-00004E54	R*4	FSLOSS	80 (10, 2)	1482	
3-0000025C	R*4	FSMTRG	144 (6, 3, 2)	285	
5-0002988C	R*4	FSSCAL	3360 (6, 7, 20)	634	
5-0002A8AC	R*4	FSSKIF	144 (6, 3, 2)	635	1649
5-0002A93C	R*4	FSSMLK	480 (6, 10, 2)	636	
3-000002EC	R*4	FTCWPL	184 (23, 2)	286	4318 4319
			4320 4364		
17-00004EA4	R*4	FTOSTR	24 (3, 2)	1483	
17-00004EBC	R*4	FTOWTR	552 (3, 23, 2)	1484	
5-0002AB1C	R*4	FWDGAS	1840 (23, 20)	637	4685= 4692(2)=
			4698(2)=		
11-00000000	R*4	GCHIM_STRT	4 (1)	1118	
12-00000000	R*4	GCRS_STRT	4 (1)	1128	
10-00000000	R*4	GC_STRT	4 (1)	1084	
5-0002B24C	R*4	GD8A	1600 (400)	638	
2-000008AB8	R*4	GKGS	4232 (23, 23, 2)	1560	3983= 3985
			4030= 4032	4059(2)= 4060	4095=
			4097 4118=	4120	4146(2)= 4153(2)=
			4230 4418		
15-00000000	R*4	GSV_STRT	4 (1)	1272	
17-000050E4	R*4	HECWPR	736 (2, 23, 4)	1485	
16-00027614	R*4	HEMOVR	120 (15, 2)	1348	
9-00006D38	CHAR	HEUDES	12000 (600)	1031	1070
17-000053C4	R*4	HEURPC	16 (2, 2)	1486	
16-0002768C	R*4	HEUSPL	2400 (600)	1347	
16-00027FEC	R*4	HEUSPS	2400 (600)	1348	
16-0002894C	R*4	HEUTYP	2400 (600)	1349	
3-000093A4	I*4	IABAAG	8 (2)	287	
6-00000CF4	I*4	IABAS	40 (10)	876	
17-000053D4	I*4	IABASP	40 (10)	1487	
13-00036DC0	I*4	IADJSH	1600 (400)	1199	
13-00037400	I*4	IADJSL	1600 (400)	1200	
6-00000D1C	I*4	IBALD	2400 (600)	877	
5-0002B88C	I*4	ICADTA	8 (2)	639	
5-0002B894	I*4	ICADTD	8 (2)	640	
5-0002B89C	I*4	ICBIT	2400 (600)	641	3418 3906
			4304 4344	4470 4526	
8-00001144	I*4	ICHTGAB	880 (220)	940	
17-000053FC	I*4	ICKPNT	6000 (50, 3, 10)	1488	
17-0000686C	I*4	ICMES	32 (4, 2)	1489	
17-0000688C	I*4	ICMS	80 (10, 2)	1490	
17-000068DC	I*4	ICMST	80 (10, 2)	1491	1767 3764(2)
13-00037A40	I*4	ICPDHD	8 (2)	1201	1622
17-00006C2C	I*4	ICPDLY	2400 (600)	1492	
5-0002C208	I*4	ICSA	40 (10)	645	1805
			3115 3647	3704= 3735=	3746=

17-0000750C	I*4	ICSMA	COMM	8 (2)	1493	
5-0002C234	I*4	ICSUPT	COMM	40 (5, 2)	647	
17-00007594	I*4	ICTIM	COMM	2400 (600)	1494	
17-00007EF4	I*4	ICTIMF	COMM	2400 (600)	1495	
6-0000E680	I*4	IDENTR	COMM	2400 (600)	879	4627
6-0000EFE0	I*4	IDLABA	COMM	12000 (300, 10)	880	1827 3412
				3783 3905	4265	4339 4460
				4465 4490	4521	
6-00011EC0	I*4	IDLIBA	COMM	24000 (300, 10, 2)	881	
6-00017C80	I*4	IDTRAF	COMM	24000 (300, 10, 2)	882	
6-0001DA40	I*4	IDTRAR	COMM	24000 (300, 10, 2)	883	
6-00023800	I*4	IDTRNF	COMM	24000 (300, 10, 2)	884	
6-000295C0	I*4	IDTRNR	COMM	24000 (300, 10, 2)	885	
5-0002C264	I*4	IDWTSU	COMM	560 (5, 7, 4)	650	
5-0002C494	I*4	IEML	COMM	240 (10, 3, 2)	651	
5-0002C584	I*4	IESC	COMM	240 (10, 3, 2)	652	
5-0002C674	I*4	IFMSK	COMM	8 (2)	653	
5-0002C67C	I*4	IFULL	COMM	80 (10, 2)	654	
2-00009B40	I*4	IHOLD		8 (2)	1560	1765= 1774=
				1785 3752	3753	
17-00008854	I*4	IMUTMF	COMM	8 (2)	1496	
7-0000002C	R*4	INTDA	COMM	2000 (25, 10, 2)	912	926
4-00037464	R*4	INTDAA	COMM	2000 (25, 10, 2)	453	527
4-00037C34	R*4	INTDAD	COMM	2000 (25, 10, 2)	454	528
7-000007FC	R*4	INTDE	COMM	2000 (25, 10, 2)	913	927
4-00038404	R*4	INTDEA	COMM	2000 (25, 10, 2)	455	529
4-00038BD4	I*4	INTDEANT	COMM	2000 (25, 10, 2)	456	
4-000393A4	R*4	INTDED	COMM	2000 (25, 10, 2)	457	530
4-00039B74	I*4	INTDEDWT	COMM	2000 (25, 10, 2)	458	
4-0003A344	R*4	INTDEH	COMM	2000 (25, 10, 2)	459	531
4-0003AB14	I*4	INTDEKWT	COMM	2000 (25, 10, 2)	460	
7-00000FCC	R*4	INTDS	COMM	2000 (25, 10, 2)	914	928
4-000382E4	R*4	INTDSD	COMM	2000 (25, 10, 2)	461	532
3-000093AC	I*4	INTDSW	COMM	8 (2)	288	
6-0002F380	I*4	INTRVL	COMM	40 (10)	886	
3-000093B4	I*4	IOPCLS	COMM	880 (220)	289	
5-0002C6D0	I*4	IOSN	COMM	120 (30)	656	
16-000292AC	I*4	IPASHE	COMM	480 (4, 15, 2)	1350	
17-0000885C	I*4	IPASS	COMM	512 (4, 32)	1497	
5-0002C750	I*4	IPROO	COMM	160 (40)	659	
5-0002C7F4	I*4	IPRSO	COMM	160 (40)	661	
17-00008A5C	I*4	IPRTY	COMM	2400 (600)	1498	
5-0002C894	I*4	IPSHLA	COMM	200 (25, 2)	662	
3-00009724	I*4	IPSQRA	COMM	40 (5, 2)	290	
3-0000974C	I*4	IPTINT	COMM	80 (10, 2)	291	
5-0002C95C	I*4	IORAP	COMM	40 (5, 2)	663	

5-0002C984	I.4	IRAC	600	(25, 3, 2)	664	1768
5-0002C8DC	I.4	IRADTA	8	(2)	665	2923A
5-0002C8E4	I.4	IRADTD	8	(2)	666	3285=
17-000093BC	I.4	IREPLC	2400	(600)	1499	3652=
3-0000979C	I.4	IRNCLS	880	(220)	292	3690=
6-0002F3A8	I.4	ISA	40	(10)	887	3730=
			1771	1803A	1620	3743
			3095=	3103=	1923(2)=	3752(2)=
			3287=	3293=	3115=	3757A
			3657=	3669=	3650=	4280
			3703=	3715=	3686=	
			3745=	3752(2)=	3730=	
			3770	3847	3924	
10-0001EF4C	I.4	ISAP	40	(10)	1091	3758
				3759=		
5-0002C8EC	I.4	ISCEF	40	(10)	667	3284
				3649	3675	
				3665		
10-0001EF78	I.4	ISECTOR_COLLAPSED	40	(10)	1093	
3-0000980C	I.4	ISHCLS	880	(220)	293	
6-0002F3D0	I.4	ISMA	40	(10)	888	
17-00009D1C	I.4	ISMAI	80	(10, 2)	1500	
2-00044DA0	I.4	ISMFR	3680	(23, 20, 2)	1586	1951=
			2251	2283	2607	1965=
			2961	2989	2944	
14-00000004	I.4	ISMT	3680	(23, 20, 2)	1256	
13-00037A50	I.4	ISMUS	8	(2)	1204	1973
				4184		
14-00000E64	I.4	ISMW	3680	(23, 20, 2)	1257	2241
			2273	2585	2760	2769
			2777	2785	2855	2941
			2958	2986	3003	
5-0002CC18	I.4	ISNPT	80	(10, 2)	669	
5-0002CC68	I.4	ISSMLC	48	(6, 2)	670	
5-0002CC98	I.4	ISSMNS	48	(6, 2)	671	
5-0002CCC8	I.4	ISSMRG	48	(6, 2)	672	
5-0002CD00	I.4	ITD	2400	(600)	675	3413
			3787	3907	4269	4525
			4628	4630	4343	
17-00009D6C	I.4	ITRKAM	240	(30, 2)	1501	
10-0002948C	I.4	IULAT	32	(4, 2)	1351	
10-0001EEFC	I.4	IWDRAW	80	(10, 2)	1090	1776=
				1785		
10-0001EFA0	I.4	IWTIGT	2208	(23, 12, 2)	1094	2030
				3074		
10-0001F840	I.4	IWTIWC	736	(23, 4, 2)	1095	1883
			3024	3035	3437	3468

13-00037A5C	I.4	IWUCE	COMM	92 (23) 3072	3086	1206	3062	3063
15-000CEA44	I.4	IWUCES	COMM	40 (10)		1274	3088=	3090=
10-0001E59C	I.4	I_COMMIT	COMM	2400 (600)		1089		
5-0002D664	I.4	JCHIPP	COMM	2400 (600)		677		
17-00009E5C	I.4	JRPWSW	COMM	16 (2, 2)		1502		
17-00009E6C	I.4	JSPSW	COMM	8 (2)		1503		
17-00009E74	I.4	JSHWSW	COMM	8 (2)		1504		
5-0002DFCC	I.4	JSHWEU	COMM	160 (10, 2, 2)		680		
17-00009E7C	I.4	KCEXTM	COMM	8 (2)		1505		
5-0002E06C	I.4	KCHAB	COMM	92 (23)		681		
5-0002E0C8	I.4	KCHSN	COMM	16 (4)		682		
17-00009E84	I.4	KDVAT	COMM	2400 (600)		1506		
17-0000A7E4	I.4	KDVMT	COMM	2400 (600)		1507		
5-0002E0D8	I.4	KNTARG	COMM	260 (65)		683		
5-0002E1E0	I.4	KPASA	COMM	160 (40)		685		
5-0002E280	I.4	KPASC	COMM	160 (40)		686		
5-0002E320	I.4	KPASG	COMM	160 (40)		687		
5-0002E3C0	I.4	KPASN	COMM	160 (40)		688		
5-0002E468	I.4	KPIS	COMM	2800 (70, 10)		691		
6-0002F3F8	I.4	KPS	COMM	40	1810 3097=	889	1624(2)=	1626(2)=
				1805=	2006	2009	3405	2702
				2732	3295=	3405		3644
				3850				
5-0002EF58	I.4	KPSMA	COMM	160 (40)		692		
5-0002EFF3	I.4	KPSMAA	COMM	160 (40)		693		
5-0002F098	I.4	KPSMC	COMM	160 (40)		694		
5-0002F138	I.4	KPSMCA	COMM	160 (40)		695		
5-0002F1D8	I.4	KPSMG	COMM	160 (40)		696		
5-0002F278	I.4	KPSMGA	COMM	160 (40)		697	1634	1973
				2701	3353	3588	3502	4184
				4612	4641			
5-0002F318	I.4	KPSMT	COMM	320 (80)		698		
5-0002F458	I.4	KPSMN	COMM	160 (40)		699		
5-0002F4F8	I.4	KPSMNA	COMM	160 (40)		700		
5-0002F598	I.4	KPSMR	COMM	160 (40)		701		
5-0002F638	I.4	KPSUPT	COMM	60 (15)		702		
5-0002F674	I.4	KPSY	COMM	40 (10)		703		
5-0002F69C	I.4	KSSMA	COMM	160 (40)		704		
5-0002F73C	I.4	KSSMAA	COMM	160 (40)		705		
5-0002F7DC	I.4	KSSMC	COMM	160 (40)		706		
5-0002F87C	I.4	KSSMCA	COMM	160 (40)		707		
5-0002F91C	I.4	KSSMG	COMM	160 (40)		708		
5-0002F9BC	I.4	KSSMGA	COMM	160 (40)		709		
5-0002FA5C	I.4	KSSMI	COMM	320 (80)		710		
5-0002FB9C	I.4	KSSMN	COMM	160 (40)		711		
5-0002FC3C	I.4	KSSMNA	COMM	160 (40)		712		

5-0002FCDC	I*4	KSSMR	160 (40)	713		
6-0002F420	I*4	KTER	40 (10)	890	1797	
17-0000B148	I*4	KTERIS	2800 (70, 10)	1509		
5-0002FD00	I*4	LNKHEU	80 (10, 2)	715		
16-000294AC	I*4	LNKSPT	80 (10, 2)	1352		
5-0002FD00	I*4	LNKWPC	184 (23, 2)	716		
6-0002F448	I*4	LNKWPT	184 (23, 2)	891	2078	2106
			3069			
6-0002F500	I*4	MAD	8 (2)	892		
17-0000BC3C	I*4	MBTLA	80 (2, 10)	1511		
5-0002FE88	I*4	MCDIV	4800 (600, 2)	717		
5-00031148	I*4	MFOPT	8 (2)	718		
5-00031150	I*4	MNFCON	48 (6, 2)	719		
16-000294FC	R*4	MOVTRK	8 (2)	1353	1444	
17-0000BC8C	I*4	MSW	80 (2, 10)	1512		
5-00031184	I*4	NAAC	8 (2)	721		
6-0002F508	I*4	NAC	8 (2)	893	3185	3221
			3301	3387	3612	3951
			4415	4585	4712	
16-00029504	I*4	NAEEFF	16 (4)	1354		
16-00029514	I*4	NAEEFF	16 (4)	1355		
16-00029524	I*4	NAEEFS	16 (4)	1356		
16-00029534	I*4	NAEEFT	16 (4)	1357		
14-00001CC4	I*4	NAM	8 (2)	1258	1675	2095
			2123	3220	4711	
16-00029544	I*4	NAPOD	8 (2)	1358		
6-0002F510	I*4	ND	8 (2)	894		
6-0002F518	I*4	ND1BA	80 (10, 2)	895		
16-0002954C	I*4	NDPCS	8 (2)	1359		
16-00029554	I*4	NDPDS	8 (2)	1360		
6-0002F568	I*4	NDS	80 (10, 2)	896	1816	1822
			3407	3778	3780	3790
			3899	4257	4259	4260(2)
			4263	4334	4435	4436
			4486	4511		
6-0002F588	I*4	NDTRAF	80 (10, 2)	897		
6-0002F608	I*4	NDTRAR	80 (10, 2)	898		
6-0002F658	I*4	NDTRNF	80 (10, 2)	899		
6-0002F6A8	I*4	NDTRNR	80 (10, 2)	900		
5-0003119C	I*4	NEAEF	16 (4)	726	3498A	
5-000311AC	I*4	NEDEF	16 (4)	727	3502A	
5-000311BC	I*4	NFCT	8 (2)	728		
17-0000BCDC	I*4	NFRNTG	8 (2)	1513		
16-0002955C	I*4	NHEEFF	16 (4)	1361		
16-0002956C	I*4	NHEEFP	16 (4)	1362		
16-0002957C	I*4	NHEEFS	16 (4)	1363		
16-0002958C	I*4	NHEEFT	16 (4)	1364		
14-00001CD0	I*4	NHSGR	12 (3)	1260		

5-000311C4	I+4	NMSR	COAM	80	(20)	729	4688
16-0002959C	I+4	NIFCT	COAM	8	(2)	1365	
16-0001FB24	I+4	NLSGR	COAM	12	(3)	1097	
5-00031220	I+4	NLSR	COAM	80	(20)	733	4687
5-00031270	I+4	NAM	COAM	8	(2)	734	1674
				2127	3124	3222	3237
3-00009E7C	I+4	NOCCDU	COAM	8	(2)	294	
5-0003127C	I+4	NOCDPC	COAM	8	(2)	736	
16-000295A4	I+4	NOCKPH	COAM	120	(15, 2)	1366	
17-0000BCE4	I+4	NOCKPT	COAM	128	(32)	1514	
17-0000BD64	I+4	NOCTCD	COAM	8	(2)	1515	
3-00009EB4	I+4	NOPCLS	COAM	8	(2)	295	
13-00037AC0	I+4	NPCAF	COAM	8	(2)	1209	3852
13-00037AC8	I+4	NPCDF	COAM	8	(2)	1210	3853
5-00031284	I+4	NPORTS	COAM	8	(2)	737	
5-0003128C	I+4	NORAT	COAM	8	(2)	738	
5-00031294	I+4	NR	COAM	8	(2)	739	4681
3-00009EBC	I+4	NRNCLS	COAM	8	(2)	296	
5-000312A0	I+4	NSEFF	COAM	8	(2)	741	3542A
3-00009E94	I+4	NSHCLS	COAM	8	(2)	297	
14-00001CDC	I+4	NSM	COAM	8	(2)	1261	1950
				1977	1989	2090	1955
				2230	2262	2552	2215
				2601	2643	2654	2582
				2842	2915	2938	2755
				3000	4177	4185	2983
							4205
2-00045C00	I+4	NSMFR		160	(20, 2)	1586	1956=
				1965	2235	2250	2282
				2583	2587	2602	2939
				2943	2956	2960	2984
				3001	3005		2988
10-0001FB30	I+4	NSMT	COAM	160	(20, 2)	1098	1959
10-0001FBD0	I+4	NSMW	COAM	160	(20, 2)	1099	1979
				2235	2240	2267	2583
				2584	2602	2603	2768
				2776	2784	2846	2939
				2940	2956	2957	2985
				3001	3002		
5-000312AB	I+4	NSN	COAM	8	(2)	742	
5-000312B4	I+4	NSSM	COAM	8	(2)	744	3201
				3223	3243	3246	3340
				3393	3395	3628	4440
5-000312BC	I+4	NSU	COAM	8	(2)	745	
5-000312C4	R+4	NSUTD	COAM	16800	(7, 600)	746	844
5-00035464	I+4	NT	COAM	8	(2)	747	
5-0003546C	R+4	NTSUOT	COAM	896	(7, 32)	748	845
5-000357EC	I+4	NTWSU	COAM	8	(2)	749	
7-0000BD6C	I+4	NUMCKP	COAM	40	(10)	1516	

2-00009848	I.4	NUMINC	8 (2)	4024(2)=	4604	1561	1760=	3976(2)=
13-00037AD4	I.4	NVERTA	184 (23, 2)			1212		
13-00037B8C	I.4	NVERTD	184 (23, 2)			1213		
5-000357F4	I.4	NW	8 (2)			750	1671	1673
			1747	1817		1898	1944	1945
			2704	2727		2790	2814	2860
			2883	3086		3122	3219	3256
			3257	3355		3372	3410	3948
			3949	3950		4337	4389	4397
			4413	4414		4438	4487	4512
			4562	4581		4679		
5-000357FC	I.4	NWAEFF	16 (4)			751	3510A	
5-0003580C	I.4	NWDEFF	16 (4)			752	3514A	
10-0001FC70	I.4	NWTIGT	96 (12, 2)			1100	1961	2027
			2029	3070		3072		
10-0001FCD0	I.4	NWTIWC	32 (4, 2)			1101	1865	1881
			1882	3023		3034	3436	3466
			3467					
5-0003581C	I.4	NZ	8 (2)			753		
8-0002F6F8	R.4	OCNUC	880 (220)			901		
17-0000BD94	R.4	OCWRP	736 (2, 23, 4)			1517		
3-00009E9C	R.4	OPCEQ	240 (30, 2)			298		
3-00009F8C	R.4	OPCEQM	32 (4, 2)			299		
5-00035824	R.4	PAACAM	72 (3, 3, 2)			754		
2-00009850	R.4	PAAKW	4600 (25, 23, 2)			1561	3189=	3193(2)=
			3232	3305		3318	3965	3989
			4101					
2-0000AD48	R.4	PADKW	4600 (25, 23, 2)			1562	3190=	3196(2)=
			3235	3306		3319	4014	4036
			4124					
17-0000C074	R.4	PAFRAM	24 (3, 2)			1518		
3-00009FAC	R.4	PARKAB	880 (220)			300		
2-0000BF40	R.4	PCS	8 (2)			1562	3867=	3868
			3876=	3877		3881	3882	3890
			4272					
2-0000BF48	R.4	PCSAR	80 (10, 2)			1563	3881=	3882=
			4668					
5-0003586C	R.4	PCSD	3840 (3, 32, 10)			755	3538	
3-0000A31C	R.4	PDANS	200 (25, 2)			301		
5-0003676C	R.4	PDAUT	120 (15, 2)			756		
5-000367E4	R.4	PDCUT	120 (15, 2)			757		
5-0003685C	R.4	PDDIA	16 (2, 2)			758		
5-0003686C	R.4	PDDRA	8 (2)			759		
5-00036874	R.4	PDDTA	24 (3, 2)			760		
5-0003688C	R.4	PDENS	1600 (400)			761		
8-0002FA68	R.4	PD1V	2400 (600)			902	3496	3917

3-0000A3E4	R*4	PDSHL	200 (25, 2)	4272	4273	4300	4301(2)=	4365(2)=
5-00036ECC	R*4	PDSSMS	48 (6, 2)	4495				
16-0002961C	R*4	PDIF	80 (10, 2)					
2-000008F9B	R*4	PEAAR	2400 (600)				3563=	4630
2-00000C8F8	R*4	PEDAR	2400 (600)				3564=	4630
5-00036EFC	R*4	PEOTRK	8 (2)					
3-0000A4AC	R*4	PEQDRP	8 (2)					
8-00001520	R*4	PERDEG	480 (6, 10, 2)					
5-00036F04	R*4	PERDGL	480 (6, 10, 2)					
5-000370E4	R*4	PFAFN	80 (10, 2)					
5-00037134	R*4	PHEC	120 (15, 2)					
5-000371AC	R*4	PIAIM	3400 (25, 17, 2)					
3-0000A4B4	R*4	PIAMM	288 (6, 6, 2)					
3-0000A5D4	R*4	PIAMMA	1200 (25, 6, 2)					
5-00037EF4	R*4	PIARP	16 (4)				4301(2)=	4368(2)=
16-0002968C	R*4	PIBA	80 (10, 2)					
17-0000C08C	R*4	PICVDR	8 (2)					
16-0002968C	R*4	PIFCT	2400 (600)					
3-0000AAB4	R*4	PKANSO	240 (30, 2)					
3-0000AB74	R*4	PKASD	960 (30, 4, 2)					
3-0000AF34	R*4	PKASSM	1200 (25, 6, 2)					
3-0000B3E4	R*4	PKMNSD	32 (4, 2)					
3-0000B404	R*4	PKMSD	128 (4, 4, 2)					
3-0000B484	R*4	PKMSSM	288 (6, 6, 2)					
2-0000D256	R*4	PLS	80 (10, 2)			1564	1906=	4262=
			4302(2)=	4303		4367(2)=	4381	4580
15-0000CEA6C	R*4	PLST	8 (2)					
2-0000D2A8	R*4	PMAKW	1104 (6, 23, 2)			1275	1746=	4303(2)=
			3244	3331		1565	3202=	3205(2)=
			4107			3344	3969	4001
2-0000D6F8	R*4	PMOKW	1104 (6, 23, 2)			1565	3203=	3208(2)=
			3247	3332		3345	4018	4043
			4130					
5-00037F04	R*4	PMILSN	120 (30)			769		
5-00037F7C	R*4	PMLSSM	480 (6, 10, 2)			770		
2-0000DB48	R*4	PPESAR	80 (10, 2)			1566	3796=	4664
5-0003815C	R*4	PPRCRP	16 (4)			771		
2-0000D898	R*4	PPSAR	2400 (600)			1566	3562=	4630
16-0002A01C	R*4	PRIFCT	120 (15, 2)			1370		
17-0000C094	R*4	PRIPSH	160 (10, 4)			1520		
16-0002A094	R*4	PRISDD	480 (4, 15, 2)			1371		
17-0000C134	R*4	PRIWRP	736 (2, 23, 4)			1521		
17-0000C414	R*4	PRIWSH	3680 (23, 10, 4)			1522		
16-0002A274	R*4	PRPCAP	240 (10, 2, 3)			1372		
17-0000D274	R*4	PRPHE	16 (2, 2)			1523		

17-00000284	R*4	PRPPSH	8 (2)	1524
16-0002A364	R*4	PRPSDC	8 (2)	1373
16-0002A36C	R*4	PRPSPO	8 (2)	1374
17-0000028C	R*4	PRPWRP	16 (2, 2)	1525
17-0000029C	R*4	PRPWSH	8 (2)	1526
16-0002A374	R*4	PRTARA	800 (10, 10, 2)	1375
16-0002A694	R*4	PRTARV	800 (10, 10, 2)	1376
3-000005A4	R*4	PRTCAA	2000 (25, 10, 2)	312
3-00000074	R*4	PRTCAM	480 (6, 10, 2)	313
5-0003816C	R*4	PRTCAP	240 (10, 2, 3)	772
5-0003825C	R*4	PRTCTN	80 (10, 2)	773
16-0002A984	I*4	PRTCUS	40000 (5, 20, 100)	1377
3-00000BF54	R*4	PRTDA1	200 (25, 2)	314
5-000382AC	R*4	PRTDAM	80 (10, 2)	774
3-0000001C	R*4	PRTDMI	48 (6, 2)	315
16-000345F4	R*4	PRTJSW	8 (2)	1378
5-000382FC	R*4	PRTPLE	160 (2, 10, 2)	775
16-000345FC	R*4	PRTPRO	160 (10, 4)	1379
16-0003469C	R*4	PRTPSU	8 (2)	1380
17-000002A4	R*4	PRTRED	80 (10, 2)	1527
16-000346A4	R*4	PRTTRK	160 (10, 2, 2)	1381
16-00034744	R*4	PRUCSA	480 (20, 3, 2)	1382
16-00034924	R*4	PRUDSA	320 (20, 2, 2)	1383
16-00034A64	R*4	PRUPRT	400 (10, 5, 2)	1384
16-00034BF4	R*4	PRUSDN	960 (30, 4, 2)	1385
5-0003839C	R*4	PSRADS	56 (7, 2)	776
4-00038AD8	R*4	PSRSCA	160 (4, 10)	471
4-00038B78	R*4	PSRSCD	16 (4)	472
4-00038B88	R*4	PSRSCS	16 (4)	473
4-00038B98	R*4	PSRSFA	160 (4, 10)	474
4-00038C38	R*4	PSRSFD	160 (4, 10)	475
4-00038CD8	R*4	PSRSFS	160 (4, 10)	476
4-00038D78	R*4	PSRSIA	160 (4, 10)	477
4-00038E18	R*4	PSRSID	16 (4)	478
4-00038E28	R*4	PSRSIS	16 (4)	479
4-00038E38	R*4	PSRSRA	160 (4, 10)	480
4-00038ED8	R*4	PSRSRD	160 (4, 10)	481
4-00038F78	R*4	PSRSRS	160 (4, 10)	482
4-0003C018	R*4	PSRSZA	160 (4, 10)	483
4-0003C088	R*4	PSRSZD	160 (4, 10)	484
4-0003C158	R*4	PSRSZS	160 (4, 10)	485
5-000383D4	R*4	PSZD	224 (7, 4, 2)	777
8-00001700	R*4	PSZDDS	33600 (7, 4, 300)	969
3-0000004C	R*4	PTACWT	200 (25, 2)	316
3-00000C114	R*4	PTFRCA	6000 (25, 3, 20)	317
3-000000884	R*4	PTFROM	1440 (6, 3, 20)	318
16-00034FB4	R*4	PTIS	80 (10, 2)	1386
16-00035004	R*4	PTLTN	160 (2, 10, 2)	1387

3-00000E24 R*4 PTSMIT	48 (6, 2)	319		
2-00045CA0 R*4 PWAKSM	84640 (23, 23, 20, 2)	1587	2646=	2657=
	2681			
2-0000E4F8 R*4 PWAKW	4232 (23, 23, 2)	1567	2672=	2694(2)=
	2808 (23, 23, 2)	3025(2)=	3036(2)=	3261
	3273 3962	3983	4095	
14-00001CE4 R*4 PWATS	736 (23, 4, 2)	1262	4539	
2-0005A740 R*4 PWKSM	84640 (23, 23, 20, 2)	1588	2648=	2659=
	2683			
2-0000F580 R*4 PWKWW	4232 (23, 23, 2)	1567	2673=	2695(2)=
	2835 (23, 23, 2)	3026(2)=	3037(2)=	3263
	3271 4011	4030	4118	
14-00001FC4 R*4 PWDRP	1104 (6, 23, 2)	1263	4541	4543
	4545 4549			
17-0000D2F4 R*4 PWACTS	368 (23, 2, 2)	1528		
17-0000D464 R*4 PWRCRP	736 (2, 23, 4)	1529		
13-00037C44 R*4 PWSF	184 (23, 2)	1214	1678	1681
	1683 1698	1701	1703	1719
	1721 1722	1725	1726	
2-0006F1E0 R*4 PWSUMA	4232 (23, 23, 2)	1589	2678=	2681(2)=
	2694			
2-00070268 R*4 PWSUND	4232 (23, 23, 2)	1589	2679=	2683(2)=
	2695			
5-00038484 R*4 PZDPHT	32 (4, 2)	778		
5-000384D4 R*4 GRAAFT	4400 (5, 220)	779		
17-0000D744 R*4 RAAAM	72 (3, 3, 2)	1530		
5-00039604 R*4 RAAFRM	24 (3, 2)	780		
5-0003961C R*4 RACAM	240 (10, 3, 2)	781		
17-0000D78C R*4 RACCZ	80 (10, 2)	1531		
17-0000D7DC R*4 RACFS	80 (10, 2)	1532		
17-0000D82C R*4 RACRS	80 (10, 2)	1533		
2-000712F0 R*4 RAWASM	42320 (23, 23, 20)	1590	2220=	2284=
	2570 2657	2848	2962	
2-0007BB40 R*4 RAWDSM	42320 (23, 23, 20)	1590	2221=	2286=
	2571 2659	2856	3007	
3-0000DE54 R*4 RDOPPT	8 (2)	320		
5-0003970C R*4 RDSUR	56 (7, 2)	782		
10-000350A4 R*4 REPAVL	3680 (23, 10, 4)	1388		
10-00035F04 R*4 REPCZ	368 (23, 4)	1389		
10-00036074 R*4 REPSEC	3680 (23, 10, 4)	1390		
10-00036ED4 R*4 REPWPN	3680 (23, 10, 4)	1391	4549(2)=	
5-00039744 R*4 RETRAN	2400 (600)	783		
13-00037CFC R*4 RFRASM	42320 (23, 23, 20)	1215	2119	
13-0004224C R*4 RFRDSM	42320 (23, 23, 20)	1216	2120	

5-0003A0A4	R*4	RPAC	COMM	8 (2)	784	
16-00037D34	R*4	RPAVL	COMM	160 (10, 4)	1392	
5-0003A0AC	R*4	RPCZ	COMM	16 (4)	785	
5-0003A00C	R*4	RPEHE	COMM	120 (15, 2)	786	
13-0004C79C	R*4	RPKASM	COMM	42320 (23, 23, 20)	2608	
13-00056CEC	R*4	RPKAW	COMM	12696 (23, 23, 6)	2634	
13-00059EB4	R*4	RPKDSM	COMM	42320 (23, 23, 20)	2611	
13-000643D4	R*4	RPKDW	COMM	12696 (23, 23, 6)	2635	
17-0000087C	R*4	RPPAB	COMM	8 (2)	1534	
17-00000884	R*4	RPPHE	COMM	8 (2)	1535	
17-0000088C	R*4	RPPSN	COMM	8 (2)	1536	
17-00000894	R*4	RPPSSM	COMM	8 (2)	1537	
5-0003A134	R*4	RPREHE	COMM	120 (15, 2)	787	
17-0000089C	R*4	RPRSSM	COMM	8 (2)	1538	
5-0003A1AC	R*4	RPRTK	COMM	8 (2)	788	
5-0003A1B4	R*4	RPSEC	COMM	160 (10, 4)	789	
5-0003A254	R*4	RPTRK	COMM	8 (2)	790	
17-000008A4	R*4	RPWMC	COMM	16 (2, 2)	1539	
3-00000DE5C	R*4	RRAPL	COMM	200 (25, 2)	321	
2-00086950	R*4	RRFRASM		42320 (23, 23, 20)	2119=	2284
2-000C0EA0	R*4	RRFRDSM		42320 (23, 23, 20)	2120=	2286
3-00000DF24	R*4	RRLSPL	COMM	16 (2, 2)	322	
3-00000DF34	R*4	RRSSPL	COMM	32 (4, 2)	323	
13-0006756C	R*4	RSWA	COMM	6624 (23, 12, 6)	1221	1701(2)= 1710=
13-00068F4C	R*4	RSAMD	COMM	6624 (23, 12, 6)	1222	1703(2)= 1711=
2-000A1CD0	R*4	RSHTRA		80 (20)	1599	
2-000A1D20	R*4	RSHTRD		80 (20)	1599	
3-00000DF54	R*4	RSTCD	COMM	8 (2)	324	
2-000A1E10	R*4	RSTRAS		80 (20)	1601	
2-000A1E60	R*4	RSTRDS		80 (20)	1601	
2-000CCAF0	R*4	RSTVA		1840 (23, 10, 2)	1606	
2-000CD220	R*4	RSTVD		1840 (23, 10, 2)	1606	
2-00010608	R*4	RSUM		2300 (575)	1568	3263= 3265A
				3273= 3275A	3306=	3309A 3319=
				3322A 3332=	3335A	3345= 3348A
2-00085D90	R*4	RSUMA		4232 (23, 23, 2)	1591	2563= 2570(2)=
				2573		
2-00086E18	R*4	RSUMD		4232 (23, 23, 2)	1591	2564= 2571(2)=
				2574		
2-000A1C80	R*4	RTRGT	COMM	80 (20)	1598	
17-000008B4	R*4	RUEPSP	COMM	184 (23, 2)	1540	
17-0000096C	R*4	RUEPWP		184 (23, 2)	1541	
2-00087EA0	R*4	RWASM		42320 (23, 23, 20)	1592	2224= 2608=
				2657		

2-000923F0	R*4	RWDSM	42320	(23, 23, 20)	1592	2225=	2611=
4-0003C1F8	R*4	SA	100	(25)	486		
12-00000644	R*4	SAAFSK	4000	(25, 2, 10, 2)	1130		
12-000015E4	R*4	SAARSK	4000	(25, 2, 10, 2)	1131		
12-00002504	R*4	SAATSK	4000	(25, 2, 10, 2)	1132		
7-0000179C	R*4	SACZBK	80	(10, 2)	915		
12-00003524	R*4	SADFSK	4000	(25, 2, 10, 2)	1133		
12-000044C4	R*4	SADRSK	4000	(25, 2, 10, 2)	1134		
12-00005404	R*4	SADZSK	4000	(25, 2, 10, 2)	1135		
12-00006404	R*4	SAEFSK	4000	(25, 2, 10, 2)	1136		
12-000073A4	R*4	SAERSK	4000	(25, 2, 10, 2)	1137		
12-00008344	R*4	SAEZSK	4000	(25, 2, 10, 2)	1138		
12-000092E4	R*4	SAKGS	40000	(25, 23, 10, 2)	1139	4233(2)=	
13-00006A92C	R*4	SAMA	2880	(30, 12, 2)	1223	1721(2)=	1730=
			2096	2124			
13-00008B46C	R*4	SAMD	2880	(30, 12, 2)	1224	1722(2)=	1731=
			2097	2125			
5-0003A25C	R*4	SAMHEA	10800	(30, 15, 6)	791		
5-0003C08C	R*4	SAMHEM	1440	(4, 15, 6)	792		
3-0000DF5C	R*4	SAMIA	5760	(30, 12, 4)	325		
3-0000F5DC	R*4	SAMIM	768	(4, 12, 4)	326		
13-00006BFAC	R*4	SAMMA	384	(4, 12, 2)	1225	1725(2)=	1734=
			2100	2128			
13-00006C12C	R*4	SAMMO	384	(4, 12, 2)	1226	1726(2)=	1735=
			2101	2129			
12-00014694	R*4	SASFSK	4000	(25, 2, 10, 2)	1140		
12-00015634	R*4	SASRSK	4000	(25, 2, 10, 2)	1141		
12-000165D4	R*4	SASZSK	4000	(25, 2, 10, 2)	1142		
2-0009C940	R*4	SAWA	4232	(23, 23, 2)	1593	2088=	2116=
			2136	2152	2157		
2-0009D9C8	R*4	SAMD	4232	(23, 23, 2)	1593	2089=	2117=
			2137	2153	2162		
12-00017574	R*4	SBSFSK	4000	(25, 2, 10, 2)	1143		
12-00018514	R*4	SCACSK	4000	(25, 2, 10, 2)	1144		
12-00019484	R*4	SCASUAL	240	(3, 10, 2)	1145	4296(2)=	4297(2)=
			4298(2)=	4323(2)=	4324(2)=	4325(2)=	
12-000195A4	R*4	SCDCSK	4000	(25, 2, 10, 2)	1146		
12-0001A544	R*4	SCDCSK	4000	(25, 2, 10, 2)	1147		
12-0001B4E4	R*4	SCDCSK	4000	(25, 2, 10, 2)	1148		
4-0003C25C	R*4	SD	100	(25)	487		
16-00037DD4	R*4	SDDTF	240	(30, 2)	1393		
3-0000F8DC	R*4	SDFROP	32	(4, 2)	327		
16-00037EC4	R*4	SDIS	240	(30, 2)	1394		
5-0003D22C	R*4	SDIV	24000	(600, 10)	793	3537	3540

16-00037FB4	I-4	SDNCUS	COAM	96000	(4, 60, 100)	1395	1442
16-0004F6B4	R-4	SDNJSW	COAM	8	(2)	1396	
16-0004F6BC	R-4	SDN`RK	COAM	480	(30, 2, 2)	1397	
2-00010F04	R-4	SEFAR		2400	(600)	1569	3788
					4630		
10-00025AB0	R-4	SFAPP	COAM	8	(2)	1108	1758
					3975(2)=	4023(2)=	1761=
					4607A		4604(2)=
2-00011864	R-4	SFAPPT		8	(2)	1569	1758=
					4601		1759(2)=
5-00042FEC	R-4	SFCAS	COAM	96	(6, 2, 2)	794	4101
5-0004304C	R-4	SFGND	COAM	96	(6, 2, 2)	795	4124
5-000430AC	R-4	SFIAHE	COAM	24	(3, 2)	796	4118
5-000430C4	R-4	SFINT	COAM	16	(2, 2)	797	
13-0006C2AC	R-4	SFRFE	COAM	16	(2, 2)	1227	
5-000430D4	R-4	SFSSM	COAM	96	(6, 2, 2)	798	4130
12-0001C4B4	R-4	SGKGS	COAM	42320	(23, 23, 10, 2)	1149	4230(2)=
4-0003C2C0	R-4	SH	COAM	100	(25)	488	
12-000269D4	R-4	SIAFSK	COAM	4000	(25, 2, 10, 2)	1150	
10-0001FCF0	R-4	SIARSK	COAM	4000	(25, 2, 10, 2)	1102	
10-00022BD0	R-4	SIATZSK	COAM	4000	(25, 2, 10, 2)	1105	
16-0004F89C	R-4	SIBA	COAM	240	(30, 2)	1398	
12-00027974	R-4	SIEFSK	COAM	4000	(25, 2, 10, 2)	1151	
10-00020C90	R-4	SIERSK	COAM	4000	(25, 2, 10, 2)	1103	
10-00023B70	R-4	SIEZSK	COAM	4000	(25, 2, 10, 2)	1106	
16-0004F9BC	R-4	SIFCT	COAM	24000	(10, 600)	1399	
8-00009A40	R-4	SINTDA	COAM	6000	(25, 10, 6)	970	
12-00028914	R-4	SISFSK	COAM	4000	(25, 2, 10, 2)	1152	
10-00021C30	R-4	SISRSK	COAM	4000	(25, 2, 10, 2)	1104	
10-00024B10	R-4	SISZSK	COAM	4000	(25, 2, 10, 2)	1107	
4-0003C324	R-4	SK	COAM	100	(25)	489	
13-0006C2BC	R-4	SLSCAS	COAM	2000	(25, 10, 2)	1228	4452
13-0006CABC	R-4	SLSSSM	COAM	480	(6, 10, 2)	1229	4455
16-0005574C	R-4	SLTSPA	COAM	800	(10, 2, 10)	1400	
16-00055A6C	R-4	SLTSPD	COAM	800	(10, 2, 10)	1401	
13-0006CC6C	R-4	SLWCB	COAM	1840	(23, 10, 2)	1230	4445
12-000298B4	R-4	SMENG	COAM	36800	(23, 20, 10, 2)	1153	2917=
					2964(2)=	2992(2)=	2947(2)=
					4198	4207	4178
12-00032874	R-4	SMKGS	COAM	11640	(6, 23, 10, 2)	1154	4236(2)=
14-00002414	R-4	SMSTK	COAM	1600	(20, 10, 2)	1264	1957
					4178	4180=	1990
3-0000F8FC	R-4	SMXRPL	COAM	8	(2)	328	
5-00043134	R-4	SNLTLN	COAM	480	(2, 30, 2)	799	

16-0005508C	R*4	SPRLD	800 (10, 10, 2)	1402	
16-000560AC	R*4	SPRLDA	800 (10, 10, 2)	1403	
16-000563CC	R*4	SQCSA	160 (20, 2)	1404	
16-0005646C	R*4	SQDSA	160 (20, 2)	1405	
5-00043314	R*4	SRRAC	8 (2)	800	
4-0003C388	R*4	SS	100 (25)	490	
2-0009EA50	R*4	SSAMA	5520 (30, 23, 2)	1594	2096= 2124=
			3128 3150	3155	
2-0009FFE0	R*4	SSAMD	5520 (30, 23, 2)	1594	2097= 2125=
			3129 3151	3160	
2-000A1570	R*4	SSAMMA	736 (4, 23, 2)	1595	2100= 2128=
			3134 3170	3175	
2-000A1850	R*4	SSAMMD	736 (4, 23, 2)	1595	2101= 2129=
			3135 3171	3180	
16-0005650C	R*4	SSDIS	240 (30, 2)	1406	
2-0001186C	R*4	SSMALO	480 (6, 10, 2)	1570	1653= 1656
			1663A 3630	3631	3969 4001
			4018 4043	4107	4130 4455
5-0004331C	R*4	SSMAMT	480 (6, 10, 2)	801	
8-00008180	R*4	SSMCON	480 (6, 10, 2)	971	1651 1662A
3-0000F904	R*4	SSMILA	192 (4, 6, 2)	329	
5-000434FC	R*4	SSMNL1	192 (6, 4, 2)	802	3205 3208
5-0004358C	R*4	SSMPTM	576 (6, 12, 2)	803	1651
5-000437FC	R*4	SSMRDS	48 (6, 2)	804	1652(2) 1654(2)=
			1664A		
16-000565FC	R*4	SUAQUA	2400 (10, 30, 2)	1407	
16-00056F5C	R*4	SUAQUE	2400 (10, 30, 2)	1408	
4-0003C3EC	R*4	SUAM	2500 (625)	491	
4-0003C080	R*4	SUNM1	2500 (625)	492	
2-00011A4C	R*4	SUNMY	92 (23)	1570	3960= 3962(2)=
			3965(2)= 3969(2)=	3972	4009= 4011(2)=
			4014(2)= 4018(2)=	4021	
5-0004382C	R*4	SUPASN	4800 (30, 10, 4)	805	
16-000578BC	R*4	SUPQUA	800 (10, 10, 2)	1409	
16-000578DC	R*4	SUPQUE	800 (10, 10, 2)	1410	
16-00057EFC	R*4	SUPSD	2400 (10, 30, 2)	1411	
16-0005885C	R*4	SUPSDA	2400 (10, 30, 2)	1412	
4-0003D774	R*4	TA	100 (25)	493	
5-00044AEC	R*4	TACND	24 (3, 2)	806	
4-0003D7D8	R*4	TABAAF	400 (25, 4)	494	
4-0003D968	R*4	TABAAR	400 (25, 4)	495	
4-0003DAF8	R*4	TABAAZ	400 (25, 4)	496	
4-0003DC88	R*4	TABAEF	400 (25, 4)	497	
4-0003DE18	R*4	TABAEI	400 (25, 4)	498	
4-0003DFA8	R*4	TABAEZ	400 (25, 4)	499	
4-0003E138	R*4	TABASF	400 (25, 4)	500	

4-0003E2C8	R*4	TABASR	COMM	400 (25, 4)	501		
4-0003E458	R*4	TABASZ	COMM	400 (25, 4)	502		
5-00044804	R*4	TADSR4	COMM	56 (7, 2)	807		
5-0004483C	R*4	TADTHE	COMM	8 (2)	808		
5-00044844	R*4	TADTSZ	COMM	32 (4, 2)	809		
5-00044864	R*4	TAESRA	COMM	56 (7, 2)	810		
8-0008390	R*4	TAESZD	COMM	33600 (7, 4, 300)	972		
5-0004489C	R*4	TASEHD	COMM	8 (2)	811		
5-000448A4	R*4	TASESZ	COMM	32 (4, 2)	812		
4-0003E5E8	R*4	TBSUP	COMM	400 (25, 4)	503		
4-0003E778	R*4	TCASA	COMM	400 (25, 4)	504		
4-0003E908	R*4	TCASD	COMM	400 (25, 4)	505		
4-0003EA98	R*4	TCASE	COMM	400 (25, 4)	506		
4-0003EC28	R*4	TCASS	COMM	400 (25, 4)	507		
5-000448C4	R*4	TCDCUM	COMM	9600 (4, 600)	813		
8-000303C8	R*4	TCDCUR	COMM	9600 (4, 600)	903		
3-0000F9C4	R*4	TCDEQ	COMM	240 (30, 2)	330		
3-0000FAB4	R*4	TCDEQM	COMM	32 (4, 2)	331		
5-00047144	R*4	TDIV	COMM	2400 (600)	814		
12-00035394	R*4	TEXNPM	COMM	1840 (23, 10, 2)	1155	2921=	2931(2)=
				2979(2)=	4173A		
16-000591BC	R*4	TIFCT	COMM	2400 (600)	1413		
4-0003ED88	R*4	TINTDAF	COMM	400 (25, 4)	508		
4-0003F268	R*4	TINTDAR	COMM	400 (25, 4)	511		
4-0003F718	R*4	TINTDAZ	COMM	400 (25, 4)	514		
4-0003EF48	R*4	TINTDEF	COMM	400 (25, 4)	509		
4-0003F3F8	R*4	TINTDER	COMM	400 (25, 4)	512		
4-0003F8A8	R*4	TINTDEZ	COMM	400 (25, 4)	515		
4-0003F008	R*4	TINTDSF	COMM	400 (25, 4)	510		
4-0003F588	R*4	TINTDSR	COMM	400 (25, 4)	513		
4-0003FA38	R*4	TINTDSZ	COMM	400 (25, 4)	516		
4-0003FBC8	R*4	TJOR	COMM	100 (25)	517		
7-000017EC	R*4	TLRSSC	COMM	16 (2, 2)	916		
7-000017FC	R*4	TMRSSC	COMM	16 (2, 2)	917		
13-0006D39C	R*4	TNGDIV	COMM	2400 (600)	1231		
14-00002A54	R*4	TNGMOD	COMM	2400 (600)	1265	3500	3504
				3512	3516		
2-000A1B30	R*4	TNWIGT		96 (12, 2)	1596	2028=	2031(2)=
				2081	2109	2110	
5-00047AA4	R*4	TPD	COMM	128 (32)	815	3535	3916
15-000CEA74	R*4	TPDIV	COMM	8 (2)	1276	1745=	4273(2)=
5-00047B24	R*4	TPSUTD	COMM	896 (7, 32)	816		
16-00059B1C	R*4	TRKCAP	COMM	24 (3, 2)	1414		
16-00059B34	R*4	TRKMVR	COMM	8 (2)	1415		
17-0000DA24	R*4	TRKRDC	COMM	8 (2)	1542		
4-0003FC2C	R*4	TS	COMM	100 (25)	518		
2-00011AA8	R*4	TSC		80 (10, 2)	1571	3901=	3933(2)=

3941

2-00011AF8	R*4	TSL	800	(10, 10, 2)	1571	4443=	4445(2)=
			4448	4457(2)=	4467	4483	
7-0000180C	R*4	TSRSSC	32	(4, 2)	918		
8-000136D0	R*4	TSSMSF	480	(6, 10, 2)	973		
5-00047EA4	R*4	TTD	128	(32)	817		
4-0003FC90	R*4	TUMM	2500	(625)	519		
4-00040654	R*4	TUMM1	2500	(625)	520		
5-00047F24	R*4	TWD	2944	(23, 32)	818	1868	3455
			3457				
2-00011E18	R*4	TWS	8	(2)	1572	1899=	1901(2)=
			1918(2)	1920	1930(2)		
5-00048AA4	R*4	TWSJTD	4480	(5, 7, 32)	819		
2-00011E20	R*4	TYPCAF	32	(8)	1572	3855=	3865A
2-00011E40	R*4	TYPCDF	32	(8)	1573	3858=	3874A
17-00000A2C	R*4	UNOVR	128	(32)	1543		
7-0000182C	R*4	VAABA	80	(10, 2)	919	1915=	1926=
			3610=	3614(2)=	3637	3660	3677
			3707	3721	3827	3831	3833
			4288	4291	4574	4575	4656
7-0000187C	R*4	VADBA	80	(10, 2)	920	1916=	3611=
			3615(2)=	3637	3648	3655	3660
			3677	3701	3707	3721	3821
			3823	3829	4281	4284	4577
			4578	4658			
13-0006DCFC	R*4	VAMAW	5520	(30, 23, 2)	1232	3193	
13-0006F28C	R*4	VAMDW	5520	(30, 23, 2)	1233	3196	
5-00049C24	R*4	VAMHEA	10800	(30, 15, 6)	820		
5-0004C654	R*4	VAMHEM	1440	(4, 15, 6)	821		
3-0006FAD4	R*4	VAMIA	11040	(30, 23, 4)	332		
3-000125F4	R*4	VAMIM	1472	(4, 23, 4)	333		
2-00011E60	R*4	VDAC	2400	(600)	1573	3572=	3574
			3583	3592	4630		
2-000127C0	R*4	VDDC	2400	(600)	1574	3573=	3575
			3586	3592	4630		
7-000018CC	R*4	VCABA	80	(10, 2)	921	1911=	1925=
			3574(2)=	3604	3835(2)	3868	4652
2-00013120	R*4	VCABAI	80	(10, 2)	1574	1913=	1927=
			3583(2)=	3660	3677	3707	3721
			3827	3831	3835	4288	4290
			4574				
7-0000191C	R*4	VGDBA	80	(10, 2)	922	1912=	3575(2)=
			3604	3825(2)	3877	4654	
2-00013170	R*4	VGDBAI	80	(10, 2)	1575	1914=	1928=

15-000CEA7C R*4 VIACF	COMM	2000 (25, 10, 2) 3321A 3324(2)=	1277 3308A 3614	3311(2)=
		3586(2)= 3648 3701 3707 3655 3829 4281 3721 3821 4283 4577		3677 3825
15-000CF24C R*4 VIADCF	COMM	2000 (25, 10, 2) 3322A 3325(2)=	1278 3391	3312(2)=
15-000CFA1C R*4 VIMACF	COMM	480 (6, 10, 2) 3347A 3350(2)=	1279 3393	3337(2)=
15-000CFBFC R*4 VIMDCF	COMM	480 (6, 10, 2) 3348A 3351(2)=	1280 3395	3338(2)=
10-00025AB8 R*4 VIWACF	COMM	1840 (23, 10, 2) 3309A 3322A 3438 3442 3476 3481	1109 335A 3459 3485	3275(2)A 3368 3472
5-0004CBF4 R*4 VIWASF	COMM	184 (23, 2) 4232 (23, 23, 2)	822 1576	2634=
2-000131C8 R*4 VIWAW		2672		
10-000261E8 R*4 VIWDCF	COMM	1840 (23, 10, 2) 3308A 3321A 3440 3443 3478 3483	1110 334A 3460 3487	3275A 3384 3474
5-0004CCAC R*4 VIWDSF	COMM	184 (23, 2) 4232 (23, 23, 2)	823 1576	2635=
2-000131C0 R*4 VLS		8 (2) 3877= 3878(2)= 3975 3978 3983 4023 4025 4030	1575 3883 3989 4036	3869(2)= 3890 4001 4043
15-000CFDDC R*4 VLSAR	COMM	80 (10, 2) 4670	1281	3884=
7-0000196C R*4 VMABA	COMM	80 (10, 2) 3639 3660 3827 3831 4574 4660	923 3677 3834	3630(2)= 3721 4292
7-0000198C R*4 VMDBA	COMM	80 (10, 2) 3639 3648 3701 3707 3829 4281	924 3655 3721 4285	3631(2)= 3677 3824 4662
13-0007081C R*4 VMMAW	COMM	736 (4, 23, 2)	1234	
13-00070AFC R*4 VMMDW	COMM	736 (4, 23, 2)	1235	
2-00015208 R*4 WATSAR		2400 (600)	1577	4630
6-00032948 R*4 WDIV	COMM	55200 (23, 600) 1885 1886	904 1888	1869 3438

17-0000DAAC	R*4	WVDATS	COMM	2400 (600)	1544		
2-000189A0	R*4	WVDDC		2400 (600)	1581	3420=	3440(2)=
				3474(2)=	3508	3585(2)	3592
5-0004E1B0	R*4	WVDDTS	COMM	2400 (600)	830		
16-00069124	R*4	XAEFF	COMM	960 (8, 15, 2)	1422		
16-000694E4	R*4	XAEFF	COMM	960 (8, 15, 2)	1423		
16-000698A4	R*4	XAEFFS	COMM	640 (8, 10, 2)	1424		
16-00069824	R*4	XAEFT	COMM	960 (8, 15, 2)	1425		
5-0004EB10	R*4	XAEF	COMM	128 (8, 4)	831	3498A	
13-000700DC	R*4	XAERTA	COMM	1472 (8, 23, 2)	1236		
13-0007139C	R*4	XAERTD	COMM	1472 (8, 23, 2)	1237		
5-0004EB90	R*4	XDEF	COMM	128 (8, 4)	832	3502A	
13-0007195C	R*4	XFMF	COMM	32 (8)	1238		
17-0000E40C	R*4	XFRNTG	COMM	64 (8, 2)	1545		
16-00069EE4	R*4	XHEFF	COMM	960 (8, 15, 2)	1426		
16-0006A2A4	R*4	XHEFF	COMM	960 (8, 15, 2)	1427		
16-0006A664	R*4	XHEFFS	COMM	640 (8, 10, 2)	1428		
16-0006A8E4	R*4	XHEFT	COMM	960 (8, 15, 2)	1429		
2-00019300	R*4	XMKGS		1104 (6, 23, 2)	1581	4001=	4003
				4043=	4070(2)=	4071	4107=
				4109	4132	4162(2)=	4236
				4425A			
3-000128B4	R*4	XOCCDU	COMM	64 (8, 2)	334		
5-0004EC10	R*4	XOCDP	COMM	64 (8, 2)	833		
17-0000E44C	R*4	XOCTCD	COMM	64 (8, 2)	1546		
13-0007197C	R*4	XPCAF	COMM	64 (8, 2)	1239	3865A	
13-0007198C	R*4	XPCDF	COMM	64 (8, 2)	1240	3874A	
5-0004EC50	R*4	XSEFF	COMM	640 (8, 10, 2)	834	3542A	
13-000719FC	R*4	XTGEF	COMM	32 (8)	1241		
5-0004EED0	R*4	XWAEF	COMM	128 (8, 4)	835	3510A	
5-0004EF50	R*4	XWDEF	COMM	128 (8, 4)	836	3514A	
16-0006ACA4	R*4	YAEFF	COMM	960 (8, 15, 2)	1430		
16-0006B064	R*4	YAEFFP	COMM	960 (8, 15, 2)	1431		
16-0006B424	R*4	YAEFFS	COMM	640 (8, 10, 2)	1432		
16-0006B6A4	R*4	YAEFT	COMM	960 (8, 15, 2)	1433		
5-0004EFD0	R*4	YAEF	COMM	128 (8, 4)	837	3498A	
13-00071A1C	R*4	YAERTA	COMM	1472 (8, 23, 2)	1242		
13-00071FDC	R*4	YAERTD	COMM	1472 (8, 23, 2)	1243		
5-0004F050	R*4	YDEF	COMM	128 (8, 4)	838	3502A	
13-0007259C	R*4	YFMF	COMM	768 (8, 6, 4)	1244		
17-0000E48C	R*4	YFRNTG	COMM	64 (8, 2)	1547		
16-0006BA64	R*4	YHEFF	COMM	960 (8, 15, 2)	1434		
16-0006BE24	R*4	YHEFFP	COMM	960 (8, 15, 2)	1435		
16-0006C1E4	R*4	YHEFFS	COMM	640 (8, 10, 2)	1436		
16-0006C464	R*4	YHEFT	COMM	960 (8, 15, 2)	1437		
3-00012BF4	R*4	YOCCDU	COMM	64 (8, 2)	335		
5-0004F0D0	R*4	YOCDF	COMM	64 (8, 2)	839		

17-0000E4CC R*4 YOCTCD
13-0007289C R*4 YPCAF
13-00072A1C R*4 YPCDF
5-0004F110 R*4 YSEFF
13-0007289C R*4 YTGEF

5-0004F390 R*4 YWAEF
5-0004F410 R*4 YWDEF

COMM
COMM
COMM
COMM
COMM

COMM
COMM

64 (8, 2)
384 (6, 8, 2)
384 (6, 8, 2)
640 (8, 10, 2)
160 (8, 5)

128 (8, 4)
128 (8, 4)

1548
1245
1246
840
1247

841
842

3855
3858
3542A

PARAMETER CONSTANTS

Type	Name	References
R*4	ACTIVE	81#
R*4	ASSEMBLY	73#
R*4	BACK	79#
R*4	BARRIER	75#
R*4	COMBAT	72#
R*4	DELUNT	76#
R*4	FORWARD	78#
I*4	INACT1	82#
I*4	INACT2	83#
I*4	LNAAC	89# 988
I*4	LNAACS	208#
I*4	LNAB	90# 597
I*4	LNABR	91#
I*4	LNABT	209#
I*4	LNABTRG	92#
I*4	LNAC	93#
I*4	LNAC	210 261 301 316 345 351 357 363 369 375 381 387 396 414 423 429 435 441 447 453 459 489
I*4	LNAC	210 275 302 317 346 352 358 364 370 376 382 388 397 418 424 430 436 442 448 454 460 490
I*4	LNAC	212 276 305 321 347 353 359 365 371 377 383 389 398 420 425 431 437 443 449 455 461 493
I*4	LNAC	214 277 308 342 348 354 360 366 372 378 384 393 399 420 426 432 438 444 450 456 486 494
I*4	LNAC	255 279 312 343 349 355 361 367 373 379 380 385 394 400 421 427 433 439 445 451 457 487 495
I*4	LNAC	258 283 314 344 350 356 362 368 374 380 386 395 401 422 428 434 440 446 452 458 488 496

I-4	LNACSQ	497	498	499	500	501	502
I-4	LNACT2	503	504	505	506	507	508
I-4	LNACTT	509	510	511	512	513	514
I-4	LNADDR	515	516	517	518	546	557
		560	562	564	662	664	767
		855	856	857	858	859	860
		861	865	866	867	868	869
		912	913	914	934	970	989
		1085	1086	1102	1103	1104	1105
		1106	1107	1130	1131	1132	1133
		1134	1135	1136	1137	1138	1139
		1140	1141	1142	1143	1144	1146
		1147	1148	1150	1151	1152	1228
		1277	1278	1555	1561	1562	
		210#	491	492	519	520	
		211#					
		94#	211	767			
		212#	1555	1568			
		95#	990				
		96#	991				
		213#					
		97#					
		98#					
		99#	213	999			
		100#					
		101#	264	265	266	267	268
		298	306	307	325	330	332
		791	820	855	992	1085	1086
		1223	1224	1232	1233	1454	1501
		1552(2)	1594(2)				
		102#	258	259	549	558	995
		1294	1295	1296	1297	1298	1299
		1300	1301	1302	1303	1304	1305
		1307	1407	1408			
		103#					
		104#					
		105#	993				
		214#	341	402	403	404	
		106#	215	638	761	1199	1200
		215#	551				
		107#					
		108#	243				
		109#					
		110#	216	1007			
		216#					
		111#	1455	1458	1465	1466	1488
		217#					
		112#					
		113#					
		114#	1337	1339	1341		
		218#					
		10#					
		11#					
		12#					
		13#					
		14#					
		15#					
		16#					
		17#					
		18#					
		19#					
		20#					
		21#					
		22#					
		23#					
		24#					
		25#					
		26#					
		27#					
		28#					
		29#					
		30#					
		31#					
		32#					
		33#					
		34#					
		35#					
		36#					
		37#					
		38#					
		39#					
		40#					
		41#					
		42#					
		43#					
		44#					
		45#					
		46#					
		47#					
		48#					
		49#					
		50#					
		51#					
		52#					
		53#					
		54#					
		55#					
		56#					
		57#					
		58#					
		59#					
		60#					
		61#					
		62#					
		63#					
		64#					
		65#					
		66#					
		67#					
		68#					
		69#					
		70#					
		71#					
		72#					
		73#					
		74#					
		75#					
		76#					
		77#					
		78#					
		79#					
		80#					
		81#					
		82#					
		83#					
		84#					
		85#					
		86#					
		87#					
		88#					
		89#					
		90#					
		91#					
		92#					
		93#					
		94#					
		95#					
		96#					
		97#					
		98#					
		99#					
		100#					
		101#					
		102#					
		103#					
		104#					
		105#					
		106#					
		107#					
		108#					
		109#					
		110#					
		111#					
		112#					
		113#					
		114#					
		115#					
		116#					
		117#					
		118#					
		119#					
		120#					
		121#					
		122#					
		123#					
		124#					
		125#					
		126#					
		127#					
		128#					
		129#					
		130#					
		131#					
		132#					
		133#					
		134#					
		135#					
		136#					
		137#					
		138#					
		139#					
		140#					
		141#					
		142#					
		143#					
		144#					
		145#					
		146#					
		147#					
		148#					
		149#					
		150#					
		151#					
		152#					
		153#					
		154#					
		155#					
		156#					
		157#					
		158#					
		159#					
		160#					
		161#					
		162#					
		163#					
		164#					
		165#					
		166#					
		167#					
		168#					
		169#					
		170#					
		171#					
		172#					
		173#					
		174#					
		175#					
		176#					
		177#					
		178#					
		179#					
		180#					
		181#					
		182#					
		183#					
		184#					
		185#					
		186#					
		187#					
		188#					
		189#					
		190#					
		191#					
		192#					
		193#					
		194#					
		195#					
		196#					
		197#					
		198#					
		199#					
		200#					
		201#					
		202#					
		203#					
		204#					
		205#					
		206#					
		207#					
		208#					
		209#					
		210#					
		211#					
		212#					
		213#					
		214#					
		215#					
		216#					
		217#					
		218#					
		219#					
		220#					
		221#					
		222#					
		223#					
		224#					
		225#					
		226#					
		227#					
		228#					
		229#					
		230#					
		231#					
		232#					
		233#					
		234#					
		235#					
		236#					
		237#					
		238#					
		239#					
		240#					
		241#					
		242#					
		243#		</			

I.4 LNIC2	590	598	599	600	601	602
I.4 LNIDT	633	650	726	727	751	752
	768	771	785	789	805	825
	826	828	831	832	835	836
	837	838	841	842	870	871
	872	1191	1247	1329	1344(2)	1345(2)
	1354	1355	1356	1357	1361	1362
	1363	1364	1379	1388	1389	1390
	1391	1392	1417	1418	1485	1517
	1520	1521	1522	1529		
	223#					
	138#	573	588	591	620	748
	755	815	816	817	818	819
	829	1003	1173	1175	1190	1197
	1468	1474	1475	1497	1514	1543
I.4 LNIFCT	139#	1031	1293	1333	1347	1348
	1349	1369	1399	1413	1416	
I.4 LNINTS	140#	691	1457	1509		
I.4 LNITBL	141#	698	710			
I.4 LNITM	142#					
I.4 LNITW	143#					
I.4 LNKAS	144#					
I.4 LNKWPS	145#					
I.4 LNKWS	146#					
I.4 LNLDP	224#	1309				
I.4 LNLDPD	225#	1326				
I.4 LNLMT	147#	855				
I.4 LNLNFC	148#	334	335	831	832	833
	834	835	836	837	838	839
	840	841	842	1236	1237	1238
	1239	1240	1241	1242	1243	1244
	1245	1246	1247	1422	1423	1424
	1425	1426	1427	1428	1429	1430
	1431	1432	1433	1434	1435	1436
	1437	1545	1546	1547	1548	1572
	1573					
	149#	226				
	226#	970				
	227#	1377				
I.4 LNLOC	150#					
I.4 LNLOCS	228#	1395				
I.4 LNLPR	229#					
	230#					
	151#					
I.4 LNMSD	152#					
I.4 LNMM	153#	269	270	271	272	273
	299	309	310	326	329	331
	333	792	802	821	1225	1226
	1234	1235	1553(2)	1595(2)		
	154#					
	155#	231	246	1010		
	231#					
I.4 LNNJFM						
I.4 LNNJW						
I.4 LNNJWD						

I.4 LNOPCLS
I.4 LNPCH
I.4 LNPNU
I.4 LNPORT

156#
157#
158#
159#
559
775
1376
1400
1527
160#

227
669
1002
1379
1401

260
765
1367
1381
1402

291
772
1368
1384
1403

312
773
1372
1386
1409

313
774
1375
1387
1410

I.4 LNPP

I.4 LNPPM
I.4 LNPSCT
I.4 LNPTP

161#
162#
163#
1168
1179
1196
1222
1688
164#
584
165#
345
351
357
363
369
375
381
387
397
408
415
421
427
433
443
453
459
480
543
1102

630
588
1170
1190
1197
1244
1699
281
585
232
346
352
358
364
370
376
382
388
399
409
416
422
428
434
444
454
460
481
544
1103

1338
794
1171
1192
1198
1245
1708
290
586
234
347
353
359
365
371
377
383
389
400
410
417
423
429
435
445
455
461
482
545
1104

1340
795
1172
1193
1218
1246
2171
581
663
342
348
354
360
366
372
378
384
393
405
411
418
424
430
440
446
456
474
483
912
1105

1342
798
1173
1194
1220
1604(2)
582
779
343
349
355
361
367
373
379
385
394
406
412
419
425
431
441
447
457
475
484
913
1106

583
1011
1178
1195
1221
1679
583
1011
1178
1195
1221
1679

I.4 LNORAT

I.4 LNR

166#
232#
167#
168#
169#
170#
233#
234#
171#
172#
391
471

446
281
233
629
215
392
477

637
729
235
395
524

448
733
236
398
541

390
414
566

I.4 LNRBST
I.4 LNRD
I.4 LNRIC
I.4 LNRINCLS
I.4 LNRIP

I.4 LNRS
I.4 LNRSB
I.4 LNRT
I.4 LNRYC
I.4 LNS

166#
232#
167#
168#
169#
170#
233#
234#
171#
172#
391
471

446
281
233
629
215
392
477

637
729
235
395
524

448
733
236
398
541

390
414
566

567	568	569	574	575	576
577	578	579	580	595	596
599	600	601	602	608	609
610	612	613	614	615	617
618	619	624	633	636	645
651	652	654	667	691	703
764	770	781	789	801	826
862	864	870	871	872	875
876	880	881	882	883	884
885	886	887	888	889	890
895	896	897	898	899	900
905	911	915	919	920	921
922	923	924	934	968	970
971	973	974	1085	1086	1088
1090	1091	1093	1109	1110	1119
1120	1129	1130	1131	1132	1133
1134	1135	1136	1137	1138	1139
1140	1141	1142	1143	1144	1145
1146	1147	1148	1149	1150	1151
1152	1153	1154	1155	1264	1273
1274	1277	1278	1279	1280	1281
1388	1390	1391	1392	1417	1418
1455	1456	1457	1458	1459	1460
1465	1466	1478	1487	1488	1490
1491	1500	1509	1511	1512	1516
1520	1522	1531	1532	1533	1556
1559	1563	1564	1566	1570	1571
1574	1575	1579	1605(2)	1606(2)	1662
1663					

I.4 LNSBT
I.4 LNSCYC
I.4 LNSD
I.4 LNSHCLS
I.4 LNSHT

I.4 LNSIC
I.4 LNSIDE

173#	217	218	
174#	317	318	634
235#	307	310	
175#			
176#			

236#	226	258	259	260	276
177#	281	282	285	291	295
280	296	297	298	299	306
307	307	309	310	312	316
319	319	320	329	549	559
636	636	766	787	796	811
1088	1088	1109	1110	1236	1242
1243	1243	1277	1278	1279	1280
1420	1420	1513	1531	1545	1547
178#	178#	415	416	417	917
237#	179#	1098	1099	1129	1153
1165	1165	1167	1169	1215	1216
1217	1217	1256	1257	1264	1273
1583(2)	1583(2)	1585(2)	1586(2)	1587	1590(2)
1592(2)	1592(2)	1597(2)	1598(2)	1599(2)	1601(2)
1602(2)	1602(2)	1603(2)	1604(2)	1605(2)	1606(2)

I.4 LNSM
I.4 LNSMAX
I.4 LNSMN

I.4 LNSMTT
I.4 LNSN

180#	595	596	1647	769	799
181#	228	238	656		

I.4 LNSNP
I.4 LNSPT

805	1001	1385	1393	1394	1397
1398	1406	1411	1412	1461	
238#					
182#	218	263	274	275	282
587	588	589	590	591	592
597	598	621	755	793	805
834	840	1173	1228	1229	1230
1293	1297	1298	1299	1300	1301
1302	1306	1312	1325	1329	1330
1331	1332	1336	1343	1344	1345
1352	1375	1376	1399	1400	1401
1402	1403	1407	1408	1409	1410
1411	1412	1424	1428	1432	1436
1474	1475	1482	1571(2)		
183#	229	772	1294	1305	1372
1414					

I.4 LNSPTG

I.4 LNSS

184#	237	323	471	472	473
474	475	476	477	478	479
480	481	482	483	484	485
548	593	594	918	1013	
185#	322	405	406	407	408
409	410	411	412	413	543
544	545	547	570	571	605
916	1009				
186#	208	214	237	240	256
259	274	284	285	304(2)	305
308	311(2)	313	315	318	319
329	550	561	563	565	595
596	634	635	636	670	671
672	719	762	764	770	801
802	803	804	968	971	973
1014	1154	1229	1279	1280	1565(2)
1570	1581	1662	1663	1664	4425
239#					
187#	240	631	650	746	748
776	777	782	807	810	816
819	969	972	1015	1468	

I.4 LNSSPT
I.4 LNSUI.4 LNSUB
I.4 LNSVST
I.4 LNTART
I.4 LNTBLS

188#					
189#					
240#					
190#	659	661	685	686	687
688	692	693	694	695	696
697	699	700	701	704	705
706	707	708	709	711	712
713					
191#	651	652			

I.4 LNTC

I.4 LNTERD
I.4 LNTERTI.4 LNTGT
I.4 LNTGTAB
I.4 LNTGR

I.4 LNTRK

241#					
192#	242	1008	1190	1196	1197
1198	1244	1350	1479	1497	
193#					
194#					
195#	241	632			
196#	1304	1311	1328	1381	1397

I-4	LNTRID
I-4	LNTP
I-4	LNVARQ
I-4	LNW

242f	212(2)	230	266	268	272
197f	286	332	333	556	572
198f	600	601	602	637	716
199f	822	823	825	826	828
273	871	872	891	904	1017
599	1094	1095	1109	1110	1119
818	1139	1149(2)	1153	1154	1155
870	1166(2)	1167(2)	1168(2)	1169(2)	1170(2)
1087(2)	1172	1177	1178	1179	1206
1120	1213	1214	1215(2)	1216(2)	1217(2)
1165(2)	1219(2)	1220(2)	1221	1222	1230
1171	1233	1234	1235	1236	1237
1212	1243	1256	1257	1262	1263
1218(2)	1282	1283	1388	1389	1390
1232	1416	1417	1418	1481	1484
1242	1517	1521	1522	1528	1529
1273(2)	1541	1552(2)	1553(2)	1554(4)	1555
1391	1557	1558	1560(2)	1561	1562
1485	1567(4)	1570	1576(4)	1579	1580
1540	1583(4)	1584(4)	1585(4)	1586	1587(2)
1556(2)	1589(4)	1590(4)	1591(4)	1592(4)	1593(4)
1581	1595(2)	1602(4)	1603(4)	1604(2)	1605(2)
1588(2)	1607(2)	1608(2)	4172(2)	4173	4174(2)
1594(2)	4224	4225	4226	4425	
1606(2)					
4175					

1.4 LNWAD
1.4 LNWCC
1.4 LNWCCD
1.4 LNWCN
1.4 LNWCOND

I-4	LNIAD	243#				
I-4	LNIACC	200#	244	1007		
I-4	LNIWCCD	244#				
I-4	LNIWCN	201#	245	246	1010	
I-4	LNIWCND	245#				
I-4	LNIWORD	202#	1005			
I-4	LNIWPC	203#	1095	1101	1880	3465
I-4	LNIWSU	204#	650	819		
I-4	LNIYLD	205#	246			
I-4	LNIYDS	246#				
I-4	LNIZ	206#	240	632	777	778
		812	969	972		809
I-4	LOCATION	86#				
I-4	MARCH	77#				
R-4	REAR	84#				
R-4	STATUS	87#				
R-4	WAIT	74#				

Labels

Address	Label	References
1-000000CA	1'	1781
1-00000C3E	10'	1785
1-00000C45	21'	4495

1-00000C4F 22'

1-000006FB 23.

1-000000C57 100.
1-000000C94 101.
1-000000CD1 102.
1-00000000E 103.
.. 104.

1-000000D4B 105.
1-000000D62 106.
1-000000D86 107.
1-000000DAB 108.
1-000000DCC 109.

1-000000EE 110.
1-000000E0C 111.
1-000000E87 112.
1-000000EA7 113.
1-000000ED3 114.

1-00000EFF 115.
1-00000F0F 116.
1-00000F23 117.
1-00000F3F 118.
.. 119.

1-00000F6B 120:
1-00000F94 121:
1-00000FCB 122:
1-00000FF6 123:
1-00000FF6 124:

1-00001025 125.
1-00001051 126.
1-000010A5 127.
1-000010DD 128.
1-000010F5 129.

1-00001113 130.
1-00001124 131.
1-00001130 132.
1-0000114C 135.
1-0000117B 140.

1-000011C1 141.
1-0000120A 142.
1-00001253 143.
1-0000129C 144.
1-000012C3 145.

1-000006ED 146'

3226	3229	3232	3235	3238	3241
3244	3247	3389	3391	3393	3395
3592	3597	3604	3637	3639	3768
3802	3888	3890	3941	4381	4383
4399	4408	4418	4422	4483	4563
4731#					
3368	3369#	3384			

3224 4732#
3227 4734#
3230 4736#
3233 4738#
4740#

3216 4741#
3365 4742#
3382 4743#
3388 4744#
3390 4745#

3589	4746#
3591	4747#
3596	4750#
3603	4751#
3636	4752#

3767
3801
3887
3889
4753#
4754#
4755#
4756#
4757#

4758# 4759#
4398 4760#
4407 4761#
4380 4762#
4416

4420 4763#
4488 4764#
4382 4766#
3940 4767#
4482 4768#

4558 4769#
1784 4770#
3769 4771#
3638 4772#
3236 4773#

3239 4775#
3242 4777#
3245 4779#
3392 4781#
3394 4782#

3366 3367 3383

18-Sep-1990 15:02:12 VAX FORTRAN V5.3-50
4-Apr-1990 13:59:47 C315:[CILLO.TACWAR30.IDA]GC.FOR:9

0-00000284 220
0-0000028F 230
0-00000297 240
0-0000013B 401

1-0000004C 402
0-00000694 450
0-00000833 500
0-00000776 505
1-00000194 600

1-000001E9 602

1-000001FA 603
1-00000250 606
0-00000325 607
1-000002B1 609

1-00000426 612
1-00000478 615
1-00000305 618
1-00000355 621
1-000003A5 623

1-000003A9 624
1-000003E7 627
1-00000731 630
0-00009F49 631
1-0000077C 632

1-00000784 633
0-0000A00B 634
1-000007BE 635
0-0000A095 636
1-000004CB 638

1-00000508 641
1-00000176 644
0-0000135A 1005
1-00000170 1006
1-00000145 1007

1-00000135 1008
1-000000FE 1009
1-00000546 1012
1-0000059E 1013
1-000005D2 1014

0-00000CED 1015
1-0000061B 1016
1-00000672 1017
1-000006BC 1018
0-00000CF5 1020

0-00001B79 1025
0-00001BEE 1026
0-00001C81 1027

1650 1658#
1648 1659#
1646 1660#
1634 1640#

1635 1636#
1670 1739#
1764 1779#
1757 1762#
2715 2717#

2720 2722# 2748 2808 2835 2877
2904
2743 2745#
2756 2757#
2761#
2765 2766#

2843 2844#
2851 2852#
2773 2774#
2781 2782#
2793 2794# 2819 2863 2888

2803 2805#
2830 2832#
4186 4187#
4191 4192#
4192 4193#

4194 4195#
4197 4198#
4198 4199#
4185 4200#
2872 2874#

2899 2901#
2702 2703# 2732
1989 1990#
1990 1991#
1987 1988#

1981 1986#
1974 1975#
3091 3092#
3098 3099#
3108 3109#

1823 1832 1894#
3279 3280#
3289 3290#
3296 3297#
1812 1896#

2149 2154#
2156 2158#
2161 2163#

0-00001C68 1035
 0-00004A75 1050

 0-00004EF1 1510
 0-00004F5A 1511
 0-00004FC1 1512
 0-0000008C 1521
 0-000000F1 1522

 0-00000E99 1523
 0-00000F94 1524
 0-00004FC8 1530
 0-00005300 1534
 0-00005316 1535

 0-00005443 1536
 0-0000544C 1537
 0-00005455 1540
 0-0000545E 1550
 0-00005092 1562

 0-000050FB 1564
 0-00005162 1568
 0-00005169 1569
 0-00005536 1580
 0-00005505 1581

 0-00005675 1582
 0-00005717 1583
 0-0000578E 1584
 0-00005864 1585
 0-00005909 1586

 0-000059AE 1587
 0-00005A08 1590
 0-00005AB2 2010
 0-00005083 2015
 0-00005040 2020

 0-00005DEC 2025
 0-00005E72 2030
 0-00005F1D 2035
 0-00005F9A 2040
 0-0000603F 2050

 0-000060B7 2060
 0-00006158 2070
 0-00006626 2090
 0-00006630 2500
 0-00006984 2505

 0-00006C61 2506
 0-00006F76 2510
 0-000073FE 2525
 0-00007407 2530
 0-0000759D 3010

2146 2165#
 1942 3040#

 3147 3152#
 3154 3156#
 3159 3161#
 1900 1905#
 1897 1908#

 1910 1917#
 1919 1920
 3144 3163#
 3192 3199#
 3188 3200#

 3204 3211#
 3201 3212#
 3187 3213#
 3120 3214#
 3167 3172#

 3174 3176#
 3179 3181#
 3164 3183#
 3225 3226#
 3228 3229#

 3231 3232#
 3234 3235#
 3237 3238#
 3240 3241#
 3243 3244#

 3246 3247#
 3217 3248#
 3258 3259
 3268 3269
 3302 3303

 3310 3313#
 3315 3316
 3323 3326#
 3328 3329
 3336 3339#

 3341 3342
 3349 3352#
 3354 3396#
 1941 3404#
 3436 3437#

 3465 3492#
 3536 3545#
 3411 3417
 3406 3601#
 3613 3616#

1929#

 3264#
 3274#
 3307#

3320#

3333#

3346#

3599#

0-00007625	3020	3609	3621#			
0-00007727	3110	3629	3632#			
0-00007730	3120	3625	3633#			
0-000077F5	4010	3782	3787	3789#		
0-00008047	4020	3777	3798#			
0-000083F7	4511	3854	3856#			
0-00008438	4512	3857	3859#			
0-00008763	5405	3900	3902#			
0-0000895C	5410	3923	3935#			
0-00008966	5430	3904	3911	3936#		
0-0000896F	5450	3897	3938#			
1-000002A3	6002	2761	2764#	2770	2778	2786
		2856				2848
0-00008AF6	6005	3961	3963#			
0-00008B53	6006	3964	3977#			
0-00008BAA	6007	3968	3971#			
0-00008BDE	6010	3959	3973#			
0-00008D06	6015	3982	3986#			
0-00008D08	6016	3988	3998#			
0-00008E83	6017	4000	4004#			
0-00008ECA	6020	3979	4006#			
0-00009633	6021	4094	4098#			
0-000096E4	6022	4100	4104#			
0-0000978D	6023	4106	4110#			
0-000097D4	6024	4091	4112#			
0-00008F4C	6030	4010	4012#			
0-00008FA9	6031	4013	4016#			
0-00009000	6032	4017	4020#			
0-00009034	6035	4008	4022#			
0-00009406	6037	4058	4061#			
0-0000949C	6038	4064	4067#			
0-000094FB	6039	4069	4072#			
0-00009157	6040	4029	4033#			
0-0000922C	6041	4035	4040#			
0-000092D4	6042	4042	4046#			
0-0000931B	6044	4026	4048#			
0-00009540	6045	4054	4075#			
0-000098BA	6046	4117	4121#			
0-0000996B	6047	4123	4127#			
0-00009A14	6048	4129	4133#			
0-00009A5B	6049	4114	4135#			
0-0000A222	6053	4229	4231#			
0-0000A279	6054	4232	4234#			
0-0000A2CA	6055	4235	4238#			
0-0000A2D1	6056	4228	4239#			
0-0000A7CC	6060	3946	4328#			
1-000007CC	6070	4218	4219#			
1-000007F7	6073	4221	4222#			

0-00000B2F	6076	
0-00000B01	6077	
0-00000C26	6078	
0-00000C65	6079	
0-00000C87	6085	
1-0000076D	6331	
0-0000AA9E	6505	
0-0000AAA7	6510	4376#
0-0000AAB0	6520	
0-0000ACD3	6565	
0-0000AD6C	6575	
0-0000AD76	6580	
0-0000AE60	6581	
0-0000AF02	6582	
0-0000AF0A	6585	
0-0000B054	7005	
0-0000B0D7	7010	
0-0000B12F	7012	
0-0000B1CA	7015	
0-0000B31D	7020	
0-0000B326	7024	
0-0000B330	7025	
0-0000B513	7079	4497#
0-0000B51C	7080	
0-0000B826	7510	
0-0000B82F	7515	4553#
0-0000B838	7520	
0-0000B8F7	7580	
0-0000BA88	8015	
0-0000BAD7	8020	
0-0000B834	8025	
0-0000B83B	8030	
0-0000B8A8	8099	4605#
1-00000822	8110	
1-0000089F	8115	
0-0000BD84	8120	4636#
1-000009C8	8130	
1-000009DA	8201	
1-00000A44	8205	
0-0000C215	8210	
1-00000A65	8211	
1-00000845	8212	
1-0000086E	8213	
1-00000A95	8221	
1-00000AC2	8222	
1-00000AF0	8223	
1-00000B1A	8224	
1-00000899	8225	

1-000000C9 8226.
1-000000EA 8227.
1-000000C15 8228.
0-000000C29C 8510
0-000000C33B 8515

0-000000C342 8520
0-000000C38F 8525
0-000000C396 8530
0-000000C3A0 8550
1-000000708 9000.

0-000000B45 9999

FUNCTIONS AND SUBROUTINES REFERENCED

Type	Name	References
ADJUST		
CVFW		
DEBUG		
EIGENV		
I*4	ISAT	
MPROD		

4666	4667#				
4668	4669#				
4670	4671#				
4684	4686#				
4691	4693#				
4689	4694#				
4697	4699#				
4682	4700#				
4678	4701#				
3618	3619#				
1763	1918	1930	3112	3299	4598#

4172	4173	4174	4175		
3498	3502	3510	3514	3542	3865
3874					
1662	1663	1664	4224	4225	4226
4425	4607				
3265	3275				
1803	2923	3645	3757		
3308	3309	3321	3322	3334	3335
3347	3348				

KEY TO REFERENCE FLAGS

- Value Modified
- # Defining Reference
- A Actual Argument, possibly modified
- D Data Initialization
- (n) Number of occurrences on line

COMMAND QUALIFIERS

```

FOR/NOLIST/NOOPT/OBJECT=C315:[CILLO.TACWAR30.IDA]GC.OBJ/CROSS/LIS=C315:[CILLO.TACWAR30.IDA]GC.LIS C315:[CILLO.TACWAR30.I
/CHECK=(NOBOUNDS,OVERFLOW,NOUNDERFLOW)
/DEBUG=(NOSYMBOLS,TRACEBACK)
/SHOW=(MODIFICATION,NOINCLUDE,MAP,NOPREPROCESSOR,SINGLE)
/STANDARD=(NOSEMANTIC,NOSOURCE_FORM,NOSYNTAX)
/WARNINGS=(NODECLARATIONS,GENERAL,NOULTRIX,NOVAXELN)
/CONTINUATIONS=19 /CROSS_REFERENCE /MOD_LINES /NOEXTEND_SOURCE
/F77 /NOG_FLOATING /I4 /NOMACHINE_CODE /NOOPTIMIZE /NOPARALLEL
/NOANALYSIS_DATA
/NO DIAGNOSTICS
/LIST=C315:[CILLO.TACWAR30.IDA]GC.LIS;1
/OBJECT=C315:[CILLO.TACWAR30.IDA]GC.OBJ;1

```

COMPILATION STATISTICS

```

Run Time:          97.61 seconds
Elapsed Time:      209.49 seconds
Page Faults:       11286
Dynamic Memory:    5617 pages

```

Distribution List for IDA Paper P-2549
A J-8 TACWAR Ground Combat Attrition Primer

27 Copies

Copies

Department of Defense

Director, Conventional Forces Analysis Division
Force Structure, Resource, and Assessment Directorate (J-8)
Joint Chiefs of Staff
The Pentagon
Washington, DC 20301

4

Mr. Arthur Paarmann
Force Structure, Resource, and Assessment Directorate (J-8)
Joint Chiefs of Staff
The Pentagon
Washington, DC 20301

2

Mr. Vincent P. Roske, Jr.
Deputy Director for Technical Operations
Force Structure, Resource, and Assessment Directorate (J-8)
Joint Chiefs of Staff
The Pentagon
Washington, DC 20301

1

Chief, Force Structure and Analysis Branch (C315)
Joint Data Systems Support Center
Defense Communications Agency
The Pentagon
Washington, DC 20301

3

Defense Technical Information Center
Cameron Station
Alexandria, VA 22314

2

Institute for Defense Analyses

15

Attention:	General Larry D. Welch	1
	Mr. Philip L. Major	1
	Dr. Robert E. Roberts	1
	Ms. Ruth L. Greenstein	1
	Dr. William J. Schultis	1
	Dr. Lowell Bruce Anderson	1
	Dr. Robert J. Atwell	1
	Dr. Peter S. Brooks	1
	Dr. Jeffrey H. Grotte	1
	Mr. Edward P. Kerlin	1
	Mr. D. Graham McBryde	1
	Dr. Frederic A. Miercort	1
	Dr. Leo A. Schmidt	1
	Control and Distribution	2