1976-06

A study of pressure-volume rates and plenum membrane additions to the captured air bubble surface effect ship XR-3 digital computer loads and motion program

Boggio, John Martin

Monterey, California. Naval Postgraduate School

A STUDY OF PRESSURE-VOLUME RATES AND PLENUM MEMBRANE ADDITIONS TO THE CAPTURED AIR BUBBLE SURFACE EFFECT SHIP XR-3 DIGITAL COMPUTER LOADS AND MOTION PROGRAM

John Martin Boggio

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A STUDY OF PRESSURE-VOLUME RATES AND PLENUM MEMBRANE ADDITIONS TO THE CAPTURED AIR BUBBLE SURFACE EFFECT SHIP XR-3 DIGITAL COMPUTER LOADS AND MOTION PROGRAM

by

John Martin Boggio

June 1976

Thesis Advisor: A. Gerba, Jr.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Study of Pressure-Volume Rates and Plenum Membrane Additions to the Captured Air Bubble Surface Effect Ship XR-3 Digital Computer Loads and Motion Program		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1976
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) John Martin Boggio		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1976
		13. NUMBER OF PAGES 111 pages
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

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

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

A study was conducted of modeling changes to the XR-3 Captured Air Bubble Surface Effect Ship digital computer Loads and Motion program. These changes included the addition of pressure rate and volume rate to the existing 6 degrees of freedom equations. Additional equations were developed to simulate the application of a nonpermeable membrane to the plenum of the XR-3 test craft. The objectives of this study were to determine whether the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

addition of pressure rate and volume rate equations would improve computer execution time and to test some simplified models of the plenum membrane. Computer timing improvements were demonstrated and membrane modeling results are presented.

A Study of Pressure-Volume Rates and Plenum Membrane Additions to the Captured Air Bubble Surface Effect Ship XR-3 Digital Computer Loads and Motion Program

by

John Martin Boggio
Lieutenant, United States Navy
B.S. E.E., Purdue University, 1967

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1976

ABSTRACT

A study was conducted of modeling changes to the XR-3 Captured Air Bubble Surface Effect Ship digital computer Loads and Motion program. These changes included the addition of pressure rate and volume rate to the existing 6 degrees of freedom equations. Additional equations were developed to simulate the application of a nonpermeable membrane to the plenum of the XR-3 test craft. The objectives of this study were to determine whether the addition of pressure rate and volume rate equations would improve computer execution time and to test some simplified models of the plenum membrane. Computer timing improvements were demonstrated and membrane modeling results are presented.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGEMENT

The author wishes to express his appreciation to Associate Professor Alex Gerba, of the U. S. Naval Postgraduate School, for the motivation, guidance, encouragement and above all, the patience which he provided during the course of this work.

The author also wishes to express his appreciation to his wife for her encouragement and support plus the many hours of typing which transformed some indecipherable symbols into a legible manuscript.

I. INTRODUCTION

A. BACKGROUND

The Captured Air Bubble Surface Effect Ship (CAB SES) offers a dynamic new approach to improve surface ship performance. As with any new approach, extensive testing at the design stage of development is required. The U.S. Navy has been conducting sea trials on two 100 ton Captured Air Bubble (CAB) test craft, the 100-A and the 100-B [Ref. 1]. In addition to scale model testing, another approach to design evaluation involves the use of high speed digital computers to simulate the dynamic performance of the craft.

A digital computer Loads and Motion (L&M) program for the CAB SES was developed under government contract by the Oceanics Corp. [Ref. 2]. The L&M program installed at the Naval Postgraduate School, W. R. Church, computer facility on an IBM 360/67 computer was for a 100 ton displacement test craft, the 100-B. Leo and Boncal [Ref. 3] modified this computer program to represent a smaller 3 ton craft, the XR-3. The XR-3 is an operational test craft currently maintained and operated by the Naval Postgraduate School [Ref. 4].

Various modifications and changes to Leo and Boncal's basic work have been implemented. Finley, Forbes, and Menzel, in Refs. 5-7, provides program changes to obtain better representation of the Fan maps, bow and stern seal dynamics and pitch and roll damping.

One adverse characteristic of both the 100-B computer program and the XR-3 computer program was the extensive use of the digital computer time when operating in simulated sea states. Since the 100-B and the XR-3 programs are similiar, the computer time analysis provided by Mitchell [Ref. 8] for the 100-B also applies to the XR-3 program.

Another aid to design is the testing done in towing tanks in which scale models are used. Towing tests were conducted at NSRDC using a rubber membrane installed on the forward portion of a CAB SES model in order to better scale the pressure-volume relationship [Ref. 9]. The addition of this membrane improved the pitch and heave characteristics of the scale model in sea state operation.

B. OBJECTIVES

It is the objective of this work to examine the L&M digital computer program and make any changes that would improve the program with particular attention to changes that would reduce the amount of time required to execute the program.

A second objective is to model the plenum membrane, add the required program changes and study the effects of the membrane on simulated craft performance.

II. PROGRAMMING CHANGES

A. RATE EQUATIONS FOR PRESSURE AND VOLUME

1. Introduction

An examination of the dynamics of the XR-3 L&M program indicate that the key parameters of this type of ship design are those which describe the dynamic behavior of the bubble of the air in the plenum. This pressure of air provides approximately 75% of the lift force when the craft is "on the bubble." Small changes in bubble volume and pressure therefore cause significant changes in total ship response. These variables were examined in an effort to decrease the digital computer simulation execution time.

2. SUBROUTINES RHS and INTGRL

The force and moment equations for the craft are the six degrees of freedom equations plus four auxillary equations contained in SUBROUTINE RHS [Ref. 2].

The integration of these equations is performed in SUBROUTINE INTGRL. INTGRL uses the Runga-Knutta-Merson numerical integration technique with an automatic variable step size. If the error tolerances specified are exceeded the step size is reduced and the step repeated until all error tolerances are met or the step size is reduced to 1×10^{-6} sec., at which point the program stops. Thus the integration values are forced to converge to their correct values. All ten equations are integrated serially at the same time step. When all integration error tolerances are met INTGRL returns the integrated values and increases the simulation time one time-step increment. Since pressure and volume are key variables, integrating their rates should cause the system of equations to converge more rapidly to their true values and avoid any problems associated with the feedback loops in the pressure-volume calculation [Ref. 10].

3. Formulation of Volume Rate and Pressure Rate

The bubble pressure equation is for an adiabatic process [Ref. 2].

$$P_b = P_a \left(\frac{\rho_b}{\rho_a} \right)^{\gamma} \tag{1}$$

where the subscripts a and b represent atmosphere and plenum values respectively

P = pressure absolute

$$\rho = \text{density } \frac{M}{V} \tag{2}$$

γ = adiabatic exponent for air

Substituting (2) into (1)

$$P_b = P_a \left(\frac{M}{V \rho_a} \right)^{\gamma} \tag{3}$$

Differentiating (3) with respect to time where P_a and ρ_a are assumed to be constant yields

$$\dot{P}_b = \frac{P_a}{\rho_a} \gamma \left(\frac{M}{V \rho_a} \right)^{\gamma - 1} \left(\frac{V\dot{M} - M\dot{V}}{V^2} \right) \tag{4}$$

Factoring $\frac{M}{V}$ from the last term of (4)

$$\dot{P}_b = \gamma P_a \left(\frac{M}{\rho_a V}\right)^{\gamma-1} \left(\frac{M}{\rho_a V}\right) \left(\frac{\dot{M}}{M} - \frac{\dot{V}}{V}\right) \qquad (5)$$

Noting $\left(\frac{M}{\rho_a V}\right)^{\gamma-1} \left(\frac{M}{\rho_a V}\right) = \left(\frac{M}{\rho_a V}\right)^{\gamma}$

and

$$P_a \left(\frac{M}{\rho_a V}\right)^{\gamma} = P_b$$

$$\dot{P}_b = \gamma P_b \left(\frac{\dot{M}}{M} - \frac{\dot{V}}{V}\right) \qquad (6)$$

The volume equation for the bubble is computed in RHS in four parts and summed together

$$V = Vnom - D - WA + WT \qquad (7)$$

The four terms that determine volume are

(1) Vnom = the empty plenum volume

(2) D = draft term-change in volume due to draft

(3) WA = Wave term-change in volume due to the presence of waves

(4) WT = WATSLP term-- correction term for the added volume wedge which is a function of speed [Ref. 6].

Differentiating equation (4)

$$\dot{V} = -\dot{D} - \dot{WA} \qquad (8)$$

where

$\dot{WA}$ = rate of change of volume due to waves

$\dot{D}$ = rate of change of volume due to displacement

$\dot{WT}$ = 0.0

$\dot{V}nom$ = 0.0

$\dot{WA}$ is computed in SUBROUTINE WAVES by taking the difference in change of volume due to waves at successive time steps and dividing by the time step. Since WATSLP is a function of speed, the differentiated value of WATSLP, $\dot{WT}$, is a function of acceleration. Forward accelerations are normally small; therefore, the differentiated value of WATSLP was approximated to be zero. Equations (4) and (5) were programmed as follows:

```
VALUE  (11)  = -(((XL*WIDTH)-ABW)*.5*W)-DVWDOT
VALUE  (12)  = GAM*VAL(13)*(VALUE(10)/VAL(11)-VALUE(11)/
                 VAL(12))
```

where

DVWDOT = rate of change of volume due to waves

VALUE(10) = Bubble mass flow rate

VALUE(11) = Volume rate

VALUE(12) = Pressure rate

VAL(11) = Bubble Mass

VAL(12) = Bubble Volume

VAL(13) = Bubble Pressure

GAM = Adiabatic Exponent for air

B. ELIMINATION OF SUBROUTINE DMINV

In an effort to improve program timing, the suggestions given in Ref. 7 were undertaken. One recommendation suggested the removal of SUBROUTINE DMINV in the 100B L&M program by making changes to SUBROUTINES INCON and RHS. An examination of the L&M program indicated that these changes also could be applied to the XR-3 program. Consequently the following was deleted:

```
212   DO 211 I = 1,6
      DO 211 N = 1,6
211   A(I,N)   = 0.0
      DO 213 N = 1.3
213   A(N,N)   = AM
      A(4,4)   = AIXX
      A(5,5)   = AIYY
      A(6,6)   = AIZZ
      A(4,6)   = -AIXZ
      A(6,4)   = -AIXZ
      AIMAX    = AMAXI(AM, AIXX,AIYY,AIZZ,ABS(AIXZ))
      DO 214 I = 1,6
      DO 214 J = 1,6
214   A(I,J)   = A(I,J)/AIMAX
      CALL DMINV(A,G,D,)
      DO 215I  = 1,6
215   A(I,J)   = A(I,J)/AIMAX
      IF (D.NE.0.0.) GO TO 10
      WRITE (6,216)
      STOP
```

In place of the above the following was added:

```
215     AMASSI = 1.0/AM
        D      = 1.0/(AIXX*AIZZ-AIXZ*AIXZ)
        DIXX   = AIXX*D
        DIXZ   = AIZZ*D
        DIZZ   = AIZZ*D
        AIYYI  = 1.0/AIYY
        GO TO 10
```

Linkage between INCON and RHS was provided by the following

```
    COMMON/ATRIX/AMASSI,AIYYI,DIXX,DIXZ,DIZZ
```

In SUBROUTINE RHS the six element matrix GF(J) was deleted and the following identifiers were substituted for the summation of forces: SUMX, SUMY, SUMZ, SUMK, SUMM, SUMN. In addition, the following deletion was made.

```
    DO 1 I = 1,6
    VALUE(I) = 0.0
    DO 1 J = 1,6
    VALUE (I) = VALUE (I) + A(I,J)*GF(J)
    1 CONTINUE
```

Substituted for the above DO LOOPS was the following

```
    VALUE (1)  = SUMX*AMASSI
    VALUE (2)  = SUMY*AMASSI -R*U
    VALUE (3)  = SUMZ*AMASSI
    VALUE (4)  = SUMK*DIZZ + SUMN*DIXZ
```

VALUE (5) = SUMM*AIYYI

VALUE (6) = SUMN*DIXX + SUMK*DIXZ

These changes eliminated SUBROUTINE DMINV and several DO LOOPS.

C. EXPANSION OF WAVE COMPONENTS

Since the program was being tested with sea state, the method used for the wave generation was studied. SUBROUTINE INCON provides a means of introducing sea-state by individual wave components and amplitudes. In addition, the subroutine will accept an average height and lowest and highest wave frequency and/or wave period and generate up to 10 wave components.

These wave components represent a sampling of the spectural energy density of the given sea state condition. Reference 11 suggests for irregular sea, a minimum of 15 - 20 components are required to simulate irregular sea conditions. The program dimension statements were accordingly increased from 10-20 to determine if any significant changes could be observed by increasing the number of wave components.

D. MISCELLANEOUS

Each subroutine was examined with regard to efficiency of coding and changes were implemented such as multiplication in lieu of the use of the exponential. For example, IBM 360/67 fixed point exponentiation requires that the natural logarithm be generated and then a logarithm be computed, as compared to straight forward multiplication. When such exponentiation is nested in DO LOOPS, a significant time savings can be realized by the use of multiplication.

III. TEST PRODECURES

A. COMPUTER SYSTEM

The digital computer used throughout this study is an IBM 360/67 model, VERSION I, located at the NPS W. R. Church Computer Facility. The system's hardware configuration provides 2 M bytes of core. This main core is composed of IBM core storage devices (IBM 2365 Mod. 12) and compatable but slightly different Lockheed core storage devices (MM 365). The Lockheed core devices are about 18% slower in execution time than IBM core devices. Since the system is time-shared and the core is contiguous, there is no practical way to determine in which portion of the main core the program is residing. Therefore, exact timing of a program is not easily achieved. In this report all timing values must be interpreted with this anomaly in mind. Among the options available under this operating system are the FORTRAM G and FORTRAN H compilers [Ref. 12]. Mitchell reported substantial time savings using the H compiler [Ref. 8]. Since one of the main objectives was to reduce execution time, computer model testing was done using both compiler options. Runs made under different compilers will be noted.

A word of caution on the use of the H compiler. The H compiler produces its timing improvements and core reduction by coding optimization. This means the compiler re-arranges the Fortran code. This re-arrangement may move computations outside of DO LOOPS for example. Applicable portions of Ref. 13 should be read and understood before using this feature.

B. TEST CONDITIONS

The L&M program for the XR-3 used in this report is the one given in Ref. 7. This program was then modified to incorporate the changes previously described. The basic program [Ref. 7] computes pressure using eq. (1) and volume using eq. (7) and is referred to as PROGRAM ONE. PROGRAM ONE was used as reference for comparative purposes. The addition of the pressure rate and volume rate equations (6) and (8) and the elimination of SUBROUTINE DMINV constituted major modifications to the basic program. This version of the Loads and Motion program was called PROGRAM TWO. PROGRAM THREE was PROGRAM TWO compiled and executed with the FORTRAN H compiler.

Five test conditions for a given speed were established to provide a full range of disturbances for testing the L&M programs. These test conditions are tabulated in Table I.

In Condition One small disturbances were introduced by changing the initial conditions of draft (DS) and pitch (θ). These changes consisted of decreasing the draft and pitch from previously computed steady state values by the magnitude shown in Table I. In addition, the rudder was held at zero and the sea was calm. In Condition Two, the perturbations were the same as Condition One except that now the rudder

TABLE I

TEST PERTURBATIONS

	Δ DS	$\Delta\theta$	RUDDER	SEA STATE
Condition One	0.1 inch	.03	0.0	Calm water
Condition Two	0.1	.03	20°	Calm water
Condition Three	0.0	0.0	0.0	Regular One
Condition Four	0.0	0.0	0.0	Regular Two
Condition Five	0.0	0.0	0.0	Irregular One

was displaced to the right 20 degrees. The rudder was offset at the start of the simulation and held through the run as this value. In Condition Three, Four and Five the only disturbances used were the sea state conditions.

Sea state is generated in SUBROUTINE WAVES. SUBROUTINE INCON generates wave components and amplitudes based on several different type input parameters. [See users manual Ref. 7]. Regular sea one and two are simple sinusoidal waves. Irregular sea state conditions are simulated by the addition of several regular sinusoidal wave components. Sea state used throughout this study is true sea state and is not scaled in any manner.

The L&M program provides two options for propulsion, either constant speed or constant thrust. The constant thrust option was selected because this is the normal operating mode for the XR-3 test craft.

Integrator tolerance levels were maintained at the values given in Ref. 7. However two new error tolerances had to be determined for the integration of the pressure and volume rates. These new levels were determined by first selecting a very tight value .000001 which was increased in increments until a change in output could be noticed. The value was then decreased by a factor of 10. Pressure and volume tolerance levels were then set at this value which was found to be .0001.

All other variables such as draft and pressure were initialized from the steady state undisturbed conditions for that specific starting speed. The steady state conditions were evaluated by first using the constant speed propulsion option and the initial values in Ref. 7. The program was then run until the key input variables - pitch, draft, thrust and pressure - reached steady state. These values were then read in and propulsion was switched to constant thrust. The program was run again to insure that steady state conditions existed. These steady state values are shown in Table II.

TABLE II

STEADY STATE CONDITIONS

Speed(Knots)	10	20	30
Pressure	23.93	24.84	24.84
Draft	8.17	6.12	5.34
Thrust (each engine)	200.31	218.17	287.22
Pitch	1.62	.48	.26

Pressure in psfg

Draft in inches

Thrust in ft. lb.

Pitch in degrees

IV. MEMBRANE ADDITION

A. INTRODUCTION

During towing tank tests conducted at NSRDC, a rubber membrane was installed on the forward bow of the XR-1 scale model [Ref. 9]. The installation of this membrane improved the pitch and heave characteristics of the scale model during simulated sea conditions. Reference 9 provided the following emperical expression which relates the Pressure and Volume of the membrane.

$$V = KP_b \tag{9}$$

K was experimentally determined and has the value .317 when scaled to the XR-3 test craft dimensions. (See Appendix A).

B. MATHEMATICAL MODELING

The approach used to include the effects of this membrane was to develop mathematical expressions which could be added to the already existing pressure, volume and air mass equations in the L&M program. Five mathematical expressions were developed, tested and referred to as Models One, Two, Three, Four and Five. They are explained below and the results discussed in Section V.

1. Model One

Model One is based on the premise that by correcting the plenum volume with the membrane volume, the equations in SUBROUTINE RHS would show the effect of the membrane. Model One added the following equation to PROGRAM ONE;

$$V_m = K P_m \qquad (10)$$

Then using (10) to correct the volume of the plenum and assuming $P_m = P_b$.

$$V = V_b + V_m \qquad (11)$$

where V_b is given by eq. (7).

2. Model Two

Model Two is an extension of Model One. Based on the premise that by correcting the rate of change of plenum volume with the rate of change of membrane volume, the rate equations given in SUBROUTINE RHS and subsequent integration of these terms would show the effect of the membrane. Model Two added the following rate equation to PROGRAM TWO.

$$\dot{V}_m = K \dot{P}_m \qquad (12)$$

Then using (12) to correct the rate of change of plenum volume and assuming

$$\dot{P}_m = \dot{P}_b$$

$$\dot{V} = \dot{V}_b + \dot{V}_m$$

where $\dot{V}_b$ is given by eq. (8).

3. Model Three

Model Three was based on the premise that the volume rate eq. (8) and the pressure rate eq. (6) as computed without the membrane could be corrected for the membrane by adjusting these terms by appropriate factors within PROGRAM TWO.

Correction terms were sought that would be a function of the membrane. A logical candidate for this correction factor for the pressure was the ratio of rates of change of the membrane volume to plenum volume multiplied by plenum pressure rate.

$$PCT = \frac{\dot{V}_m}{\dot{V}_b} \dot{P}_b \qquad (13)$$

where PCT is the correction factor to be added to the plenum pressure rate term $\dot{P}_b$ before integration. It was assumed that the pressure and pressure rate of the plenum and membrane were equal.

$$P_m = P_b \qquad (14)$$

$$\dot{P}_m = \dot{P}_b \qquad (15)$$

It was also assumed that the membrane volume rate and pressure rate were related by

$$\dot{V}_m = K\dot{P}_m \qquad (16)$$

Substituting (16) into (13) and using (15)

$$PCT = K \frac{\dot{P}_b^{\,2}}{\dot{V}_b} \qquad (17)$$

The volume rate correction term was assumed to be the ratio of the rate of change of membrane pressure to the rate of change of plenum pressure times the rate of change of membrane volume.

$$VCT = \frac{\dot{P}_m}{\dot{P}_b} \quad \dot{V}_m \tag{18}$$

where VCT is the correction factor to be added to the plenum volume rate term $\dot{V}_b$ before integration

Since $\dot{P}_m = \dot{P}_b$ and $\dot{V}_m = K\dot{P}_b$

Then from (18)

$$VCT = K\ \dot{P}_b \tag{19}$$

Then using equations (17) and (19) the rate equations are

$$\dot{V} = \dot{V}_b + VCT \tag{20}$$

$$\dot{P} = \dot{P}_b + PCT \tag{21}$$

where $\dot{V}_b$ is given by eq. (8) and $\dot{P}_b$ by eq. (6).

Consider the following. Since K is a function of the elasticity of the membrane, if K = 0.0. Equations (20) and (21) reduce to the no membrane case.

If the membrane expands at the same rate that the volume of the plenum contracts the system volume rates is zero and the system pressure rate is also zero.

4. Model Four

Again using PROGRAM TWO, Model Four was based on the premise that the mass of the membrane air and mass flow rate of membrane air would be important factors in any correction term to be applied to plenum volume rate and pressure rate.

Again it was assumed that the membrane effect could be described as an adiabatic reversible process. Then using Equation (6) applied to the membrane,

$$\dot{P}_m = P_m \gamma \left(\frac{\dot{M}_m}{M_m} - \frac{\dot{V}_m}{V_m} \right) \tag{22}$$

where m denotes membrane

$\dot{M}_m$ = Mass flow rate of the membrane air

M_m = Mass of the membrane air

Again it was assumed that the pressure and pressure rate of the membrane and plenum were equal.

$$P_m = P_b \tag{23}$$

$$\dot{P}_m = \dot{P}_b \tag{24}$$

It was also assumed that the membrane volume rate and pressure rate were related by

$$\dot{V}_m = K \dot{P}_m \tag{25}$$

Equation (25) was obtained by differentiating

$$V_m = K P_m \tag{26}$$

Substituting (25) and (26) into (22) yields

$$\frac{\dot{M}_m}{M_m} = \left(\frac{1+\gamma}{\gamma} \right) \frac{\dot{P}_m}{P_m} \tag{27}$$

One term is not available, M_m. Two possibilities were considered to determine this value. The first possibility would be to integrate $\dot{M}_m$ and use the result of the integration from the preceding time step. The other possibility would be to hold temperatures constant in the membrane, and use the ideal gas law to determine M_m for varying pressure and volume. This last condition would be in error; however, as a first approximation, it would produce acceptable results.

Because of the past history of difficulty with integrator 10, plenum air mass rate, the second possibility was chosen. Evaluation of the terms in the ideal gas law for a temperature of 68^o degrees excluding mass, pressure and volume produced the constant.

$$4.9 \text{ X } 10^{-7} \text{ slugs/psf - ft}^3$$

therefore;

$$M_m = (4.9 \text{ X } 10^{-7}) \, (P_m)(V_m) \qquad (28)$$

Equations (22), (25) and (27) describe the membrane. These equations are used to correct the mass rate and volume rate of the plenum without the membrane.

$$\dot{M}_s = \dot{M} + \dot{M}_m \qquad (29)$$

$$\dot{V}_s = \dot{V} + \dot{V}_m \qquad (30)$$

where

s = denotes the system of plenum plus membrane

$\dot{M}$ = mass flow rate of plenum

$\dot{V}$ = volume rate of change of plenum

Using (29) and (30) in equation (6) the corrected plenum pressure rate is

$$\dot{P}_s = P_b \gamma \left(\frac{\dot{M}_s}{M} - \frac{\dot{V}_s}{V} \right)$$

where P_b, M and V are values obtained from the previous time-step calculation.

5. Model Five

Model Five is the application of equation (10) to PROGRAM TWO. It is based on the premise that the volume of the membrane could be introduced into the plenum pressure rate equation, and represent the membrane effect.

$$V_m = K \; P_m \tag{31}$$

$$V = V_b + V_m \tag{32}$$

Equation (31) and (32) are computed as in Model One. Then equation (32) is used in the pressure rate equation.

$$\dot{P} = P_b \gamma \left(\frac{\dot{M}}{M} - \frac{\dot{V}}{V} \right) \tag{33}$$

V. DISCUSSION OF RESULTS

A. PRESSURE RATE AND VOLUME RATE CHANGE

The addition of the pressure rate and volume rate equations was implemented without any major difficulty. In order to provide a basis of comparison for computer time analysis a quality factor "Q" was established. "Q" is defined to be the ratio of CPU execution time to problem time. For example, if the problem specified 1 minute of simulation and the CPU execution time (GO STEP in IBM COMPUTER SYSTEMS) was 5 minutes Q = 5.0. PROGRAMS ONE, TWO and THREE were tested with identical input parameters. The results are summarized in Table III and Table IV.

Runs were compared using PROGRAM ONE as a reference. No differences in computed output were noted for Condition One thru Four and only small changes in computed values noted with Condition Five. The integrator error tolerance levels were maintained constant for all runs.

As can be seen from the tabulated values, the addition of pressure rate and volume rate equations reduces the CPU execution time considerably for small disturbances. However, as the sea state is increased the CPU time increases. Apparently other factors in the program begin to dominate the total execution time.

TABLE III

TIMING QUALITY FACTORS

20 Knots

	Program One	Program Two	Program Three
Condition One	11.25	4.8	4.50
Condition Two	13.85	6.0	5.50
Condition Three	23.0	20.5	19.2
Condition Four	42.8	28.01	22.0
Condition Five	71.0	58.1	54.0

TABLE IV

TIMING QUALITY FACTORS

30 Knots

	Program One	Program Two	Program Three
Condition One	10.8	7.5	4.5
Condition Two	18.0	14.25	8.7
Condition Three	24.0	22.0	21.0
Condition Four	45.0	30.0	28.0
Condition Five	75.0	59.2	53.0

This supports the results of Reference (7) that for sea conditions the bulk of computation time is apparently in the SUBROUTINE WAVES.

It was observed that at irregular sea state two and regular sea state three water came in contact with the top of the plenum. The L&M program does not include the effect of this phenomenon in the program simulation; therefore, testing was not conducted at higher sea state conditions. The fact that water hits the top of the plenum is a reasonable condition under certain circumstances. For example, if the pitch angle is 3 degrees up the stern sinks over 6 inches. If the draft of the craft at that time is 11 inches and if the preceding wave passing the stern station is 5 inches, water should hit the top of the plenum at the stern station. The effect of this condition on total craft behavior in terms of forces and moments is not known. The effect of water contact would depend upon such factors as the length of time the wave is in contact with the top of the plenum and the area of the plenum in contact with the wave.

B. INCREASING WAVE COMPONENTS

Tests were made for irregular seas one and two using 8, 10, 15, and 20 wave components. No apparent change in output could be determined. The time at which changes in output occurred changed, but no significant reduction in output values were noted; that is the time when wave components were in-phase or out-of-phase was different, but the craft response was the same. It was felt that if more components were used, the energy (in this case, the magnitude and frequency of the disturbing force) would be distributed over many components and that the irregular combination of these components would spread the energy of the disturbance over a larger time base. This spreading of the energy would then reduce the magnitude of the disturbance and that in turn would provide less rapid rates of change. With less rapid changes in the disturbances, the integration would take less time. The over-all time increase due to more wave components would be offset by the reduction in integration time. This is not the case; using more wave components increased CPU time. Whether 10 or less wave components are sufficient to accurately represent irregular seas was not tested.

C. MEMBRANE MODELS

A large amount of time was expended in attempting to provide adquate modeling of the plenum membrane. Model One using $V_m = K P_b$ in PROGRAM ONE did not work at all. The program would begin to execute and then stop when the integration step size was less than 1×10^{-6}.

As an aid to find the reason for this failure a computational flow diagram was drawn (See Figure 1). The addition of the correction term V_m creates an algebraic inner loop as shown by the dashed line in Fig. 1. Using a small perturbation of D in calm water and using the print switches in RHS as well as using the DEBUG option available with the G compiler, it was observed that with Model One the values of pressure were divergent. It is not known why pressure values diverge; however, it was noted that at certain times the values of pressure and volume were increasing or decreasing together, that is when pressure was increasing while volume was increasing.

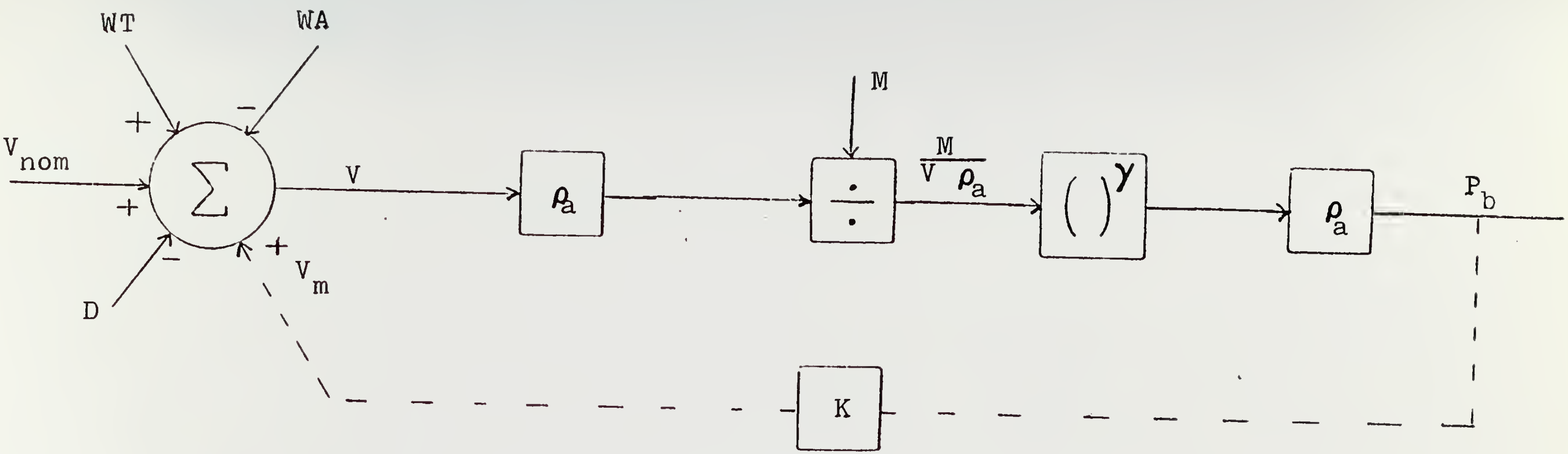
The computational flow diagram for Model Two (Figure 2) shows the inner connections of the pressure, volume and mass variables. Model Two exhibited the same symptoms and the same results as Model One, except in some cases pressure rate and volume rate rose or fell together at the same time step.

The failures of Model One and Two led to Model Three. It was felt that by correcting both pressure rate and volume rate the divergence exhibited in Models One and Two could be corrected. Model Three also failed; however, it provided the motivation to consider mass rate into the membrane. The computations performed in Model Three is shown in Figure 3.

Model Four was marginally successful. The program executes and slightly reduces the center of gravity acceleration. the reduction of the center of gravity acceleration was chosen as the criterion of whether the model was representing the membrane. This model was tested in sea state one and sea state two and provides slight reduction in the maximum values in the center of gravity acceleration. It did not reduce all of the center of gravity values. The computational flow diagram of Model Four is given in Figure 4 and shows no algebraic loops in the RHS computations of pressure and volume rates.

Model Five was the most successful model. The computational flow is shown in Figure 5. Model Five reduced the center of gravity acceleration and for small disturbances, the pitch angle and pitch rate. For sea state one a significant reduction was observed. (See Figures 6 and 7). However, for sea state two the center of gravity accelerations were greater for the membrane when compared to the no membrane case. (See Figures 8 and 9). No reason for this could be found. The only differences in these two runs was the sea state.

One possible reason for the sea state two results, though not yet proven, could be a resonance effect at the sea state encounter frequency. Concurrent work in this area [Ref. 14] investigates the conditions for the existence of any resonance phenomena. Preliminary investigations indicate that at a wave encounter frequency of approximately 4 rad / second resonance in pitch angle occurs. The encounter frequency for regular sea state two is 5 rad / second. Whether resonance phenomena could cause this effect, however, was not determined.



(All values shown for the K^{th} iteration in RHS)

Figure 1. Computational Flow Diagram Model I

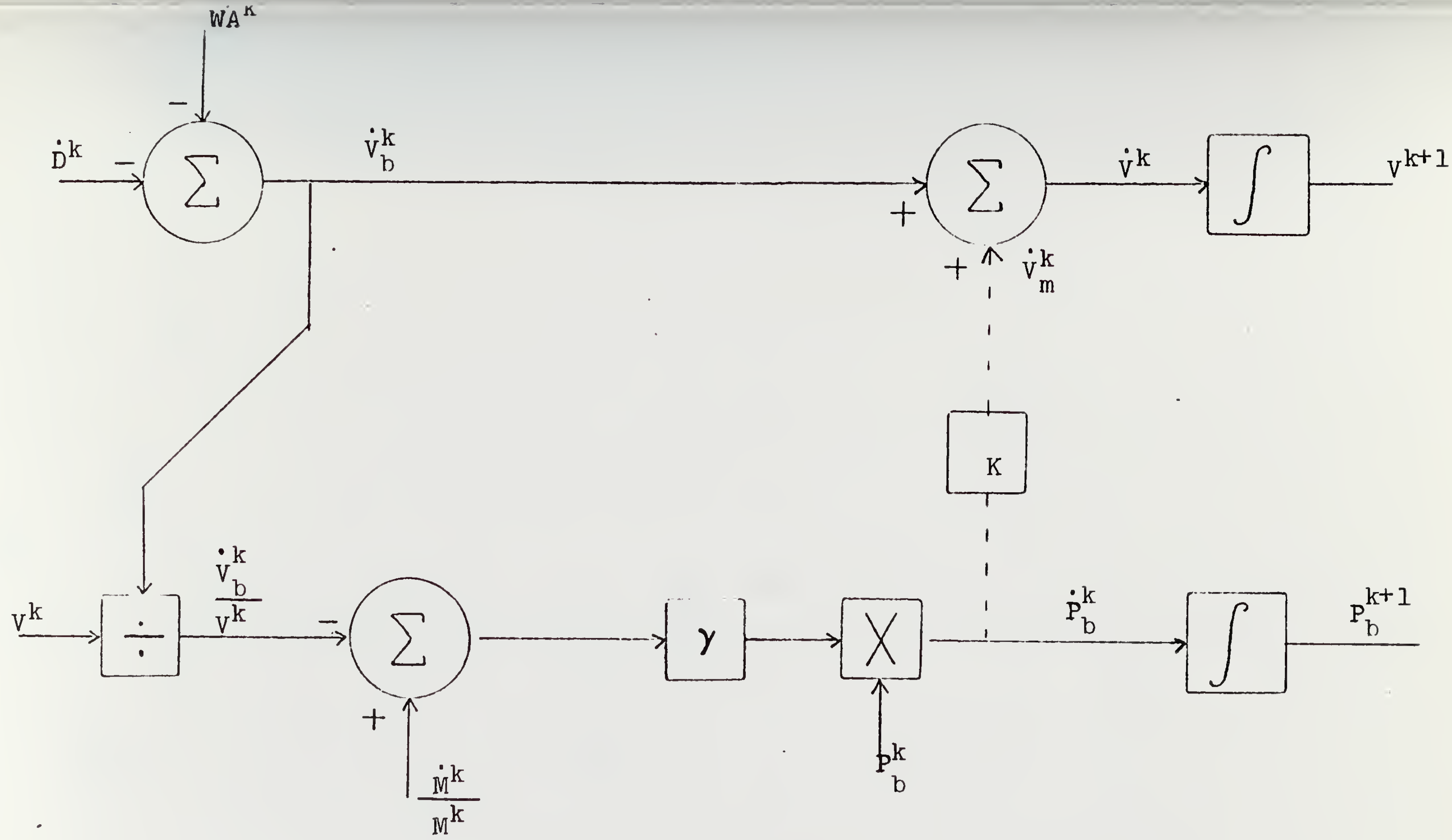


Figure 2. Computational Flow Diagram Model II

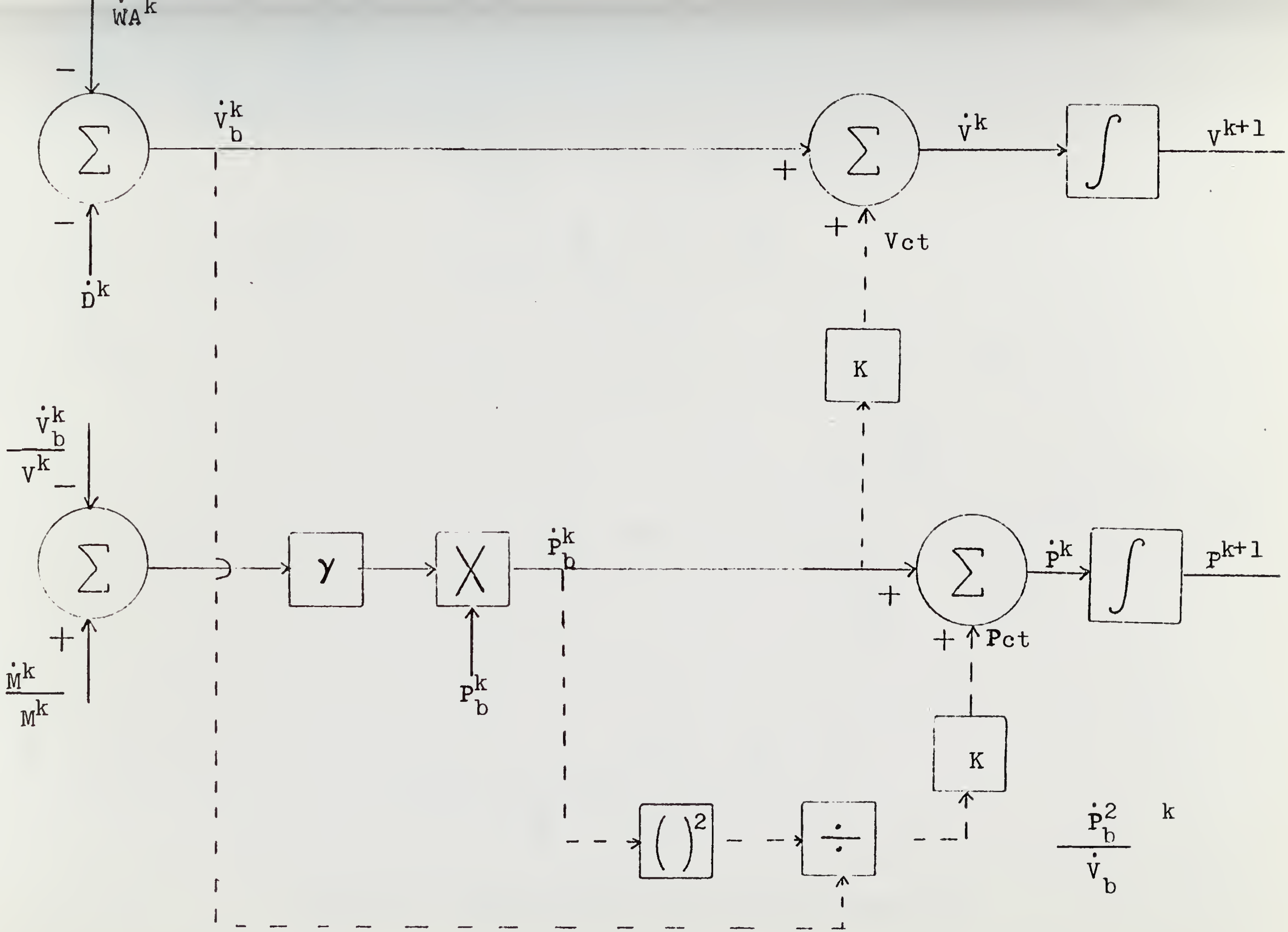


Figure 3. Computational Flow Diagram Model III

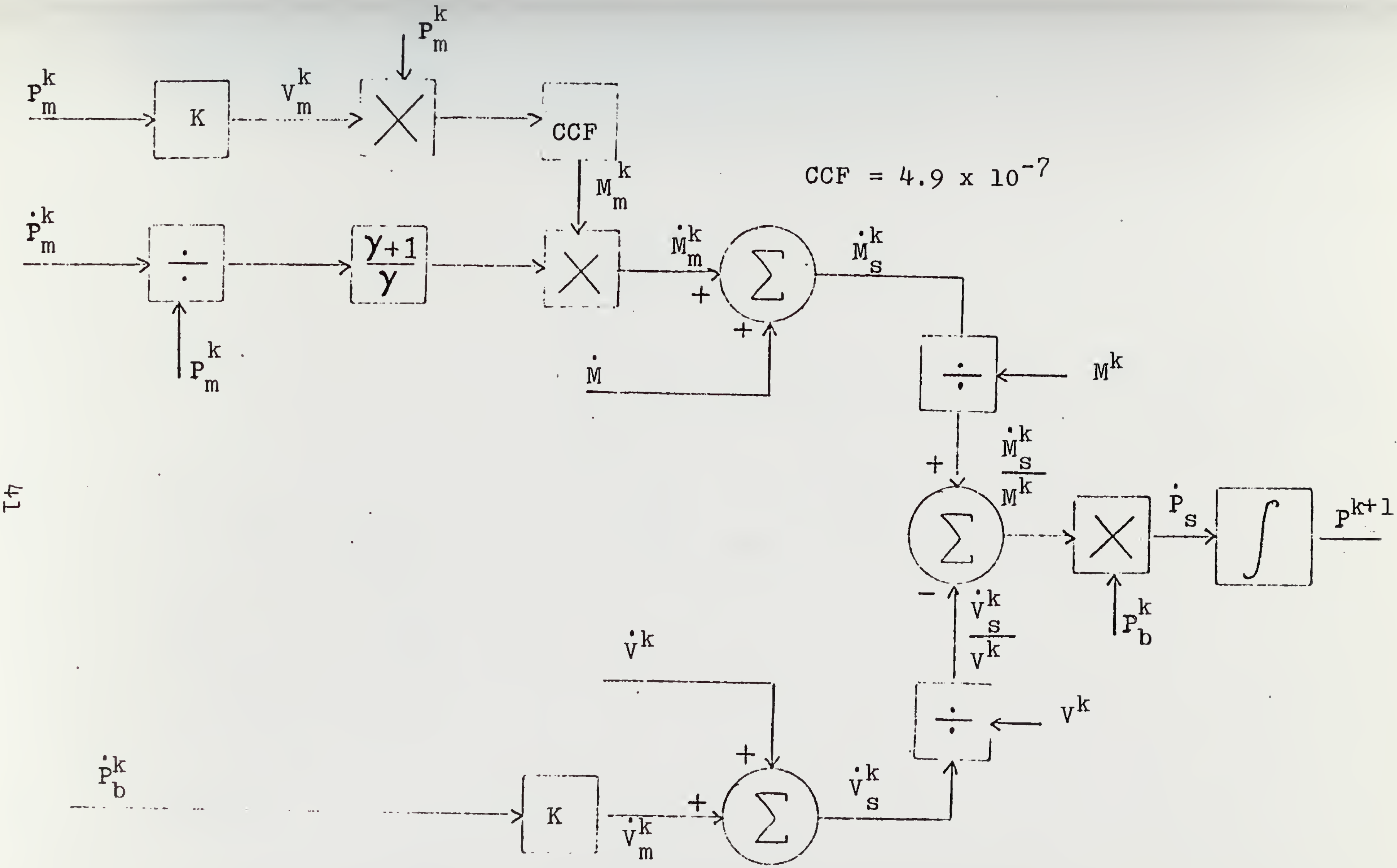


Figure 4. Computational Flow Diagram Model IV

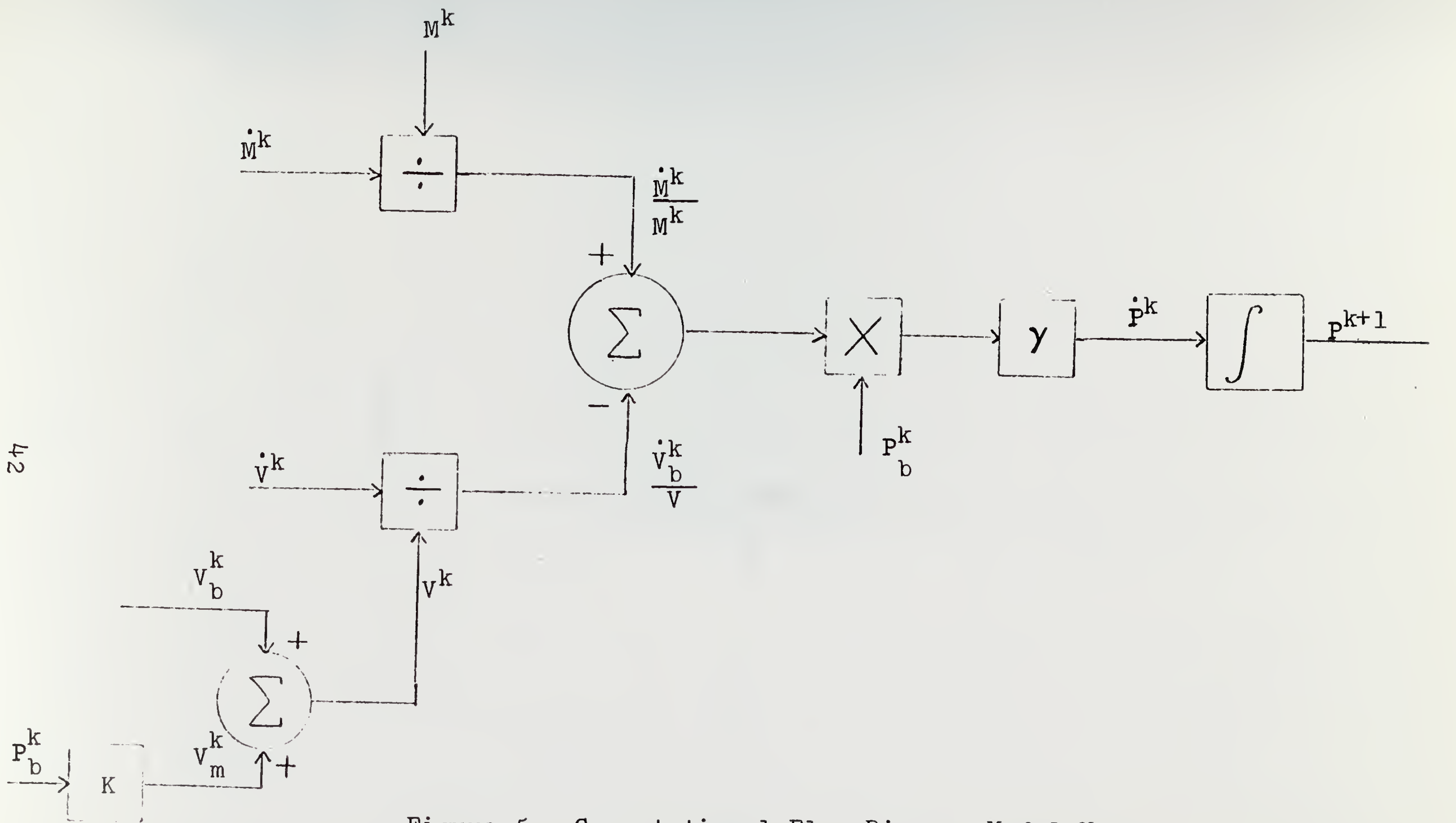


Figure 5. Computational Flow Diagram Model V

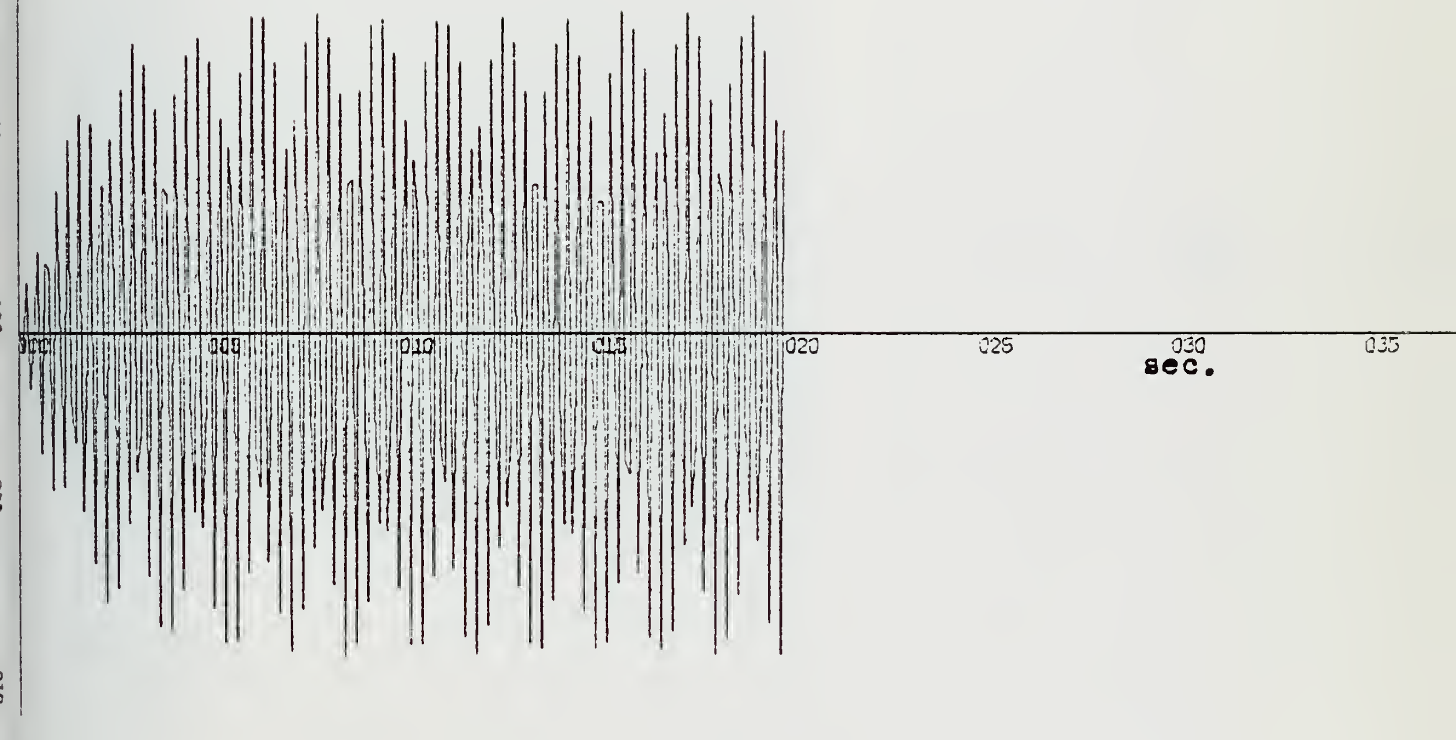


FIGURE 6

X-SCALE=5.00E+00 UNITS INCH.
Y-SCALE=5.00E-02 UNITS INCH.
PROGRAM 2 MODEL 5 20KTS REG. SEA 1
PLOT IS C.G.ACCELERATION VERSUS TIME

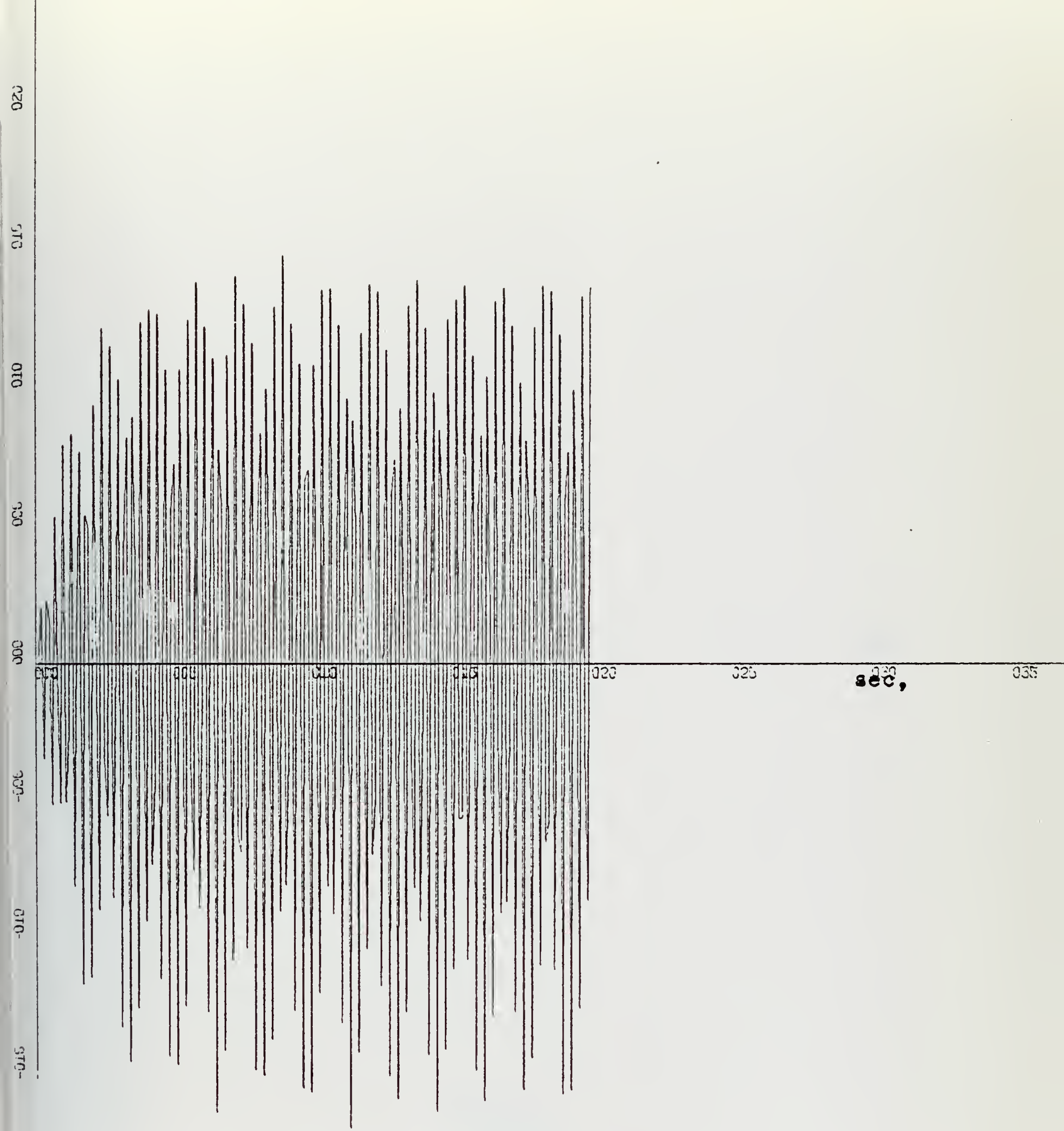


FIGURE 7

X-SCALE=5.00E+00 UNITS INCH.
Y-SCALE=5.00E-02 UNITS INCH.
PROGRAM 2 WITHOUT MEMBRANE REG. SEA 1
PLOT IS C.G.ACCELERATION VERSUS TIME

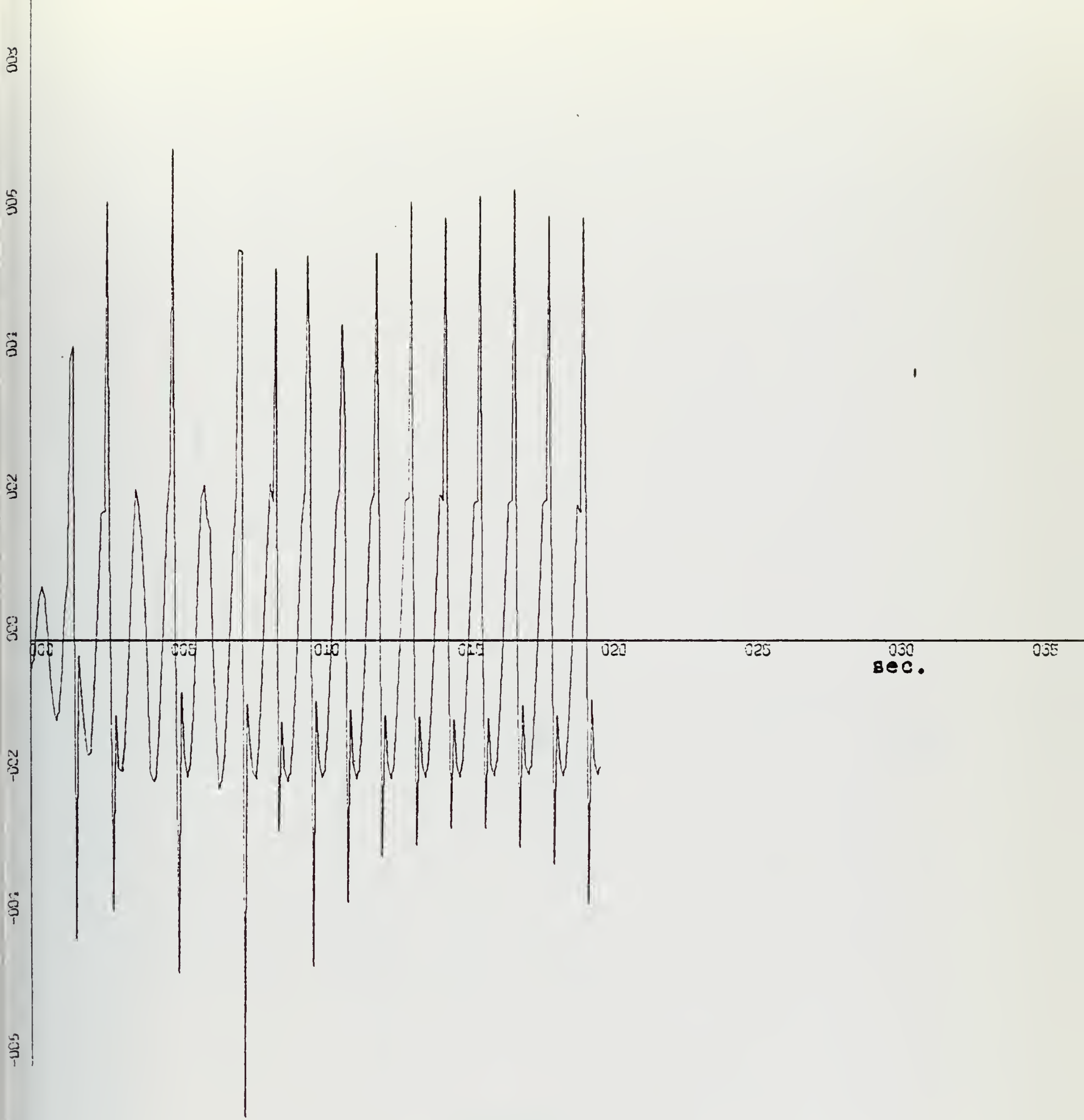


FIGURE 8

X-SCALE=5.00E+00 UNITS INCH.
Y-SCALE=2.00E-01 UNITS INCH.
PROGRAM 2 MODEL 5 20KTS REG. SEA 2
PLOT IS C.G.ACCELERATION VERSUS TIME

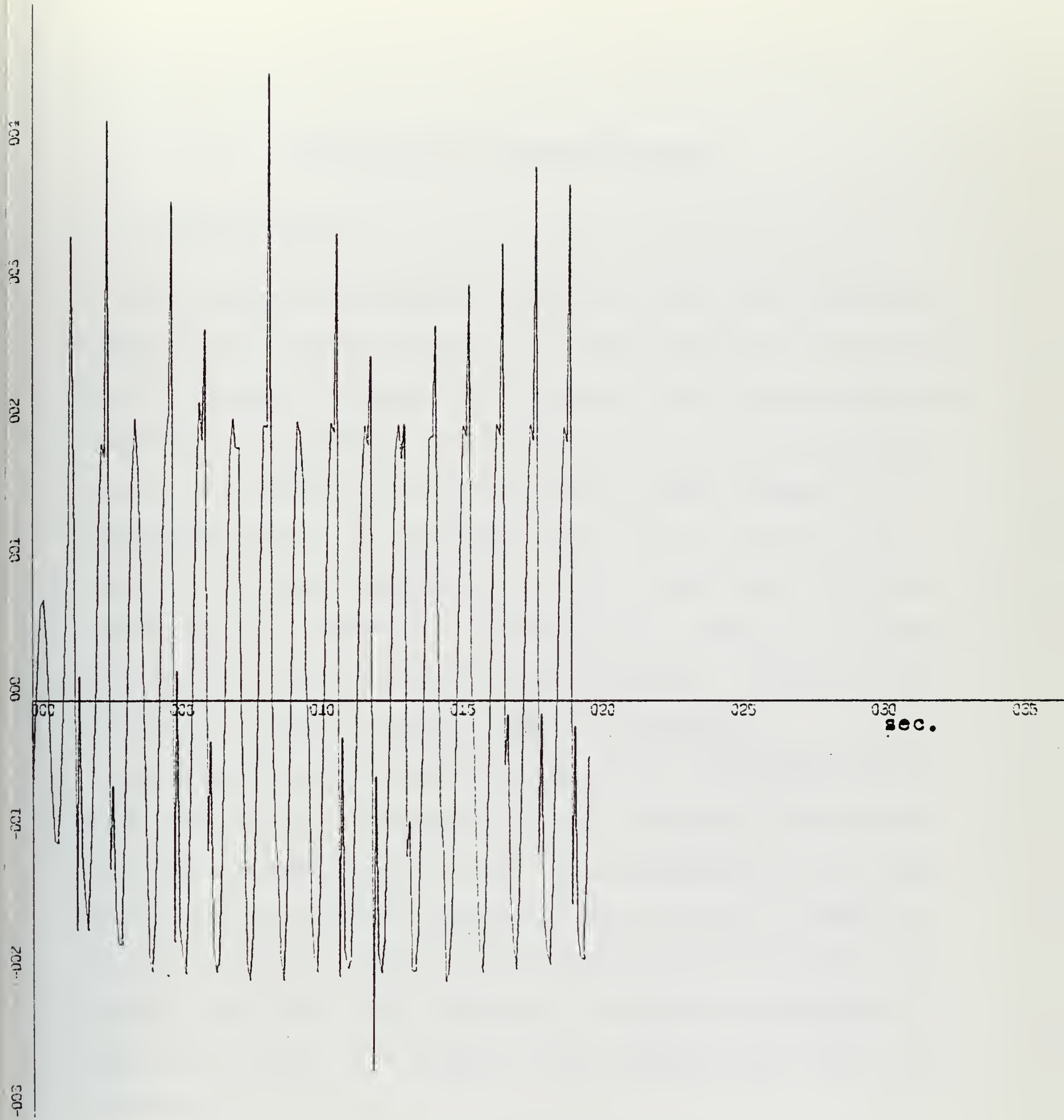


FIGURE 9

X-SCALE=5.00E+00 UNITS INCH.
Y-SCALE=1.00E-01 UNITS INCH.
PROGRAM 2 WITHOUT MEMBRANE REG. SEA 2
PLOT IS C.G.ACCELERATION VERSUS TIME

VI. CONCLUSIONS AND RECOMMENDATIONS

A. PROGRAM TIMING

The inclusion of pressure rate and volume rate equations improved the execution time of the L&M program without decreasing the program accuracy. The changes to the program were not difficult to implement. Further effort to improve execution time of the program should be directed toward changes to SUBROUTINE WAVES. In connection with this, a study of the number of required sidewall and bow and stern seal stations should also be conducted. By reducing the number of sidewall stations the execution time should decrease. A study of the losses in accuracy, however, would be necessary.

Another facet for investigation within SUBROUTINE WAVES would be a study to determine if the trignometric functions could be represented by the in-line programming of the expansion of sine and cosine functions, vice the use of computer library functions. Since some sine and cosine functions are nested in DO LOOPS small savings in time would accumulate. Again, the losses in accuracy to this change would have to be determined.

The elimination of SUBROUTINE DMINV did not produce any noticeable time savings but did save approximately 2 K bytes of core.

The expansion of the number of wave components did not demonstrate any time savings; however the optimum number of wave components required to simulate irregular sea conditions should be determined. A possible course of action would be to generate wave components, calculate the magnitude and frequency and compute the energy spectral density. This energy density spectrum could then be compared to the Pierson and Neumann or Moskowitz model [Ref. 11]. By determining the optimum number of wave components, a savings in CPU execution time could result.

B. ADDITIONAL MODELING

The fact that waves can hit the top of the plenum was discussed in Section V. Whether this effect produces significant forces and moments is unknown. Since the time that the wave is in contact with the plenum and the location (station numbers) is known and since the area of the plenum in contact with the waves could be approximated, (no greater than the difference between wetted stations) additional correction terms could be developed to determine whether significant forces and moments are generated by wave contact. The drag of the wetted plenum top could also be considered.

C. MEMBRANE MODELING

Of the five membrane models developed in this study, Model Five is the most usable. The accuracy of this model must be determined and the model must be validated. Whether mass and mass flow rate terms could improve this model also needs to be determined. In this regard a detailed study of adiabatic-reversible unsteady flow conditions would be useful. This would mean the establishment of a suitable control volume and the solution of energy-work equations.

The membrane modeling in this study was for a continuous non-permeable membrane. Additional modeling for the case of a hole cut into the membrane should be attempted. The equation for a sharp-edge orifice could be used to determine air loss. The area (size) of the hole can be determined by considering as a first approximation the ratio of surface area of a hemisphere to volume of the hemisphere. As an alternative, the experimental results of the towing tank tests conducted at NSRDC, on the XR-1, using suitable scale factors, could be used to determine empirical correction factors to be applied to the Loads and Motion program.

The occurrences of algebraic loops and summation of feed forward terms as shown in the computational flow diagrams and the failure of the models in which they occur indicate that further modeling of the membrane is required. To prevent model failures, elimination of the effects of the computational loops appears to be necessary.

APPENDIX A

MEMBRANE SCALE FACTOR

The empirical relationship between pressure and volume for the membrane given in Ref. 9 is for the XR-1 test craft.

$$V_m = K\ P_m \tag{A1}$$

where

$$K = \frac{40\ in^3}{Psf}$$

It was therefore necessary to scale the constant K for the XR-3 test craft.

In the following development the subscript 1 refers to the XR-1 and subscript 3 refers to XR-3.

Since XR-1 and XR-3 are similar it was assumed that all dimensions scale by the ratio of plenum lengths

$$\frac{L_3}{L_1} = \lambda \tag{A2}$$

where

$L_3 = 20$ ft.

$L_1 = 5.4$ ft.

It was assumed that the ratio of mass to volume was constant

$$\frac{M_1}{V_1} = \frac{M_3}{V_3} = \rho \tag{A3}$$

The ratio of craft volume is

$$\frac{V_3}{V_1} = \lambda^3 \tag{A4}$$

The ratio of craft pressure is

$$\frac{P_3}{P_1} = \frac{M_{3g}/A_3}{M_{1g}/A_1} \tag{A5}$$

where g is the gravitational constant.

Substituting (A2) and (A3) into (A5)

$$\frac{P_3}{P_1} = \frac{V_3\ A_1}{V_1\ A_3} = \frac{\lambda^3}{\lambda^2} = \lambda \tag{A6}$$

Using (A4) and (A6) in (A1)

$$K_1 = \frac{V_1}{P_1} \quad \frac{V_3/\lambda^3}{P_3/\lambda} = \frac{K_3}{\lambda^2} \tag{A7}$$

$$K_3 = \lambda^2 K_1 \tag{A8}$$

Converting in.^{3} to ft.^{3} and using (A2)

$$K_3 = \left(\frac{20}{5.4}\right)^2 \quad \frac{40}{1728} = .3175 \text{ ft.}^3/\text{psf} \tag{A9}$$

APPENDIX B

LISTING OF FORTRAN XR-3

L&M COMPUTER PROGRAM

PROGRAM TWO MODEL FIVE

```
      INTEGER ON
      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /CONST/ PI,RAD,UO
      COMMON /ENGINE/ NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,TIP
     1(25),TIS(25)
      COMMON /EQNCO/ NEQS,TOL(20),JQQ
      COMMON /FPROP/ FXP,FYP,FZP,FKP,FMP,FNP
      COMMON /FROUDE/ FN,FNCRIT
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /ROLL/ PHIMAX,TROLL
      COMMON /RUDDR/ NPR,DELPUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,R
     1CLB,RTC,RUDANG,TIR(25)
      COMMON /VALOLD/ YOLD(20)
      COMMON /VARBLE/ VAL(40)
      COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FY
     1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PB
     2BAR
      EQUIVALENCE (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAL(5),P), (VAL(6
     1),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAL(10),Z), (VAL(
     211),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI), (VAL(24),PB),
     3 (VAL(12),VOL)
      DIMENSION DUMMY(20)
      TC = 1.0
      ON = 1
      PI = 4.*ATAN(1.)
      RAD = 180./PI
      WRITE (6,12)
    1 READ (44,13,END=2) DUMMY
      WRITE (6,14) DUMMY
      WRITE (5,13) DUMMY
      GO TO 1
    2 REWIND 5
    3 CALL INCON (TIME)
      IF (IMM.EQ.3) GO TO 11
      VAL(13) = VAL(24)
C
      DO 4 J=1,20
    4 YOLD(J) = VAL(J+1)
C
      GO TO 8
    5 CONTINUE
      TOLD = TIME
      PBBAR = PBBAR*(1.-DELT/TC)+DELT*(PB-PINF)/TC
```

```
      IF (NWAVE.LE.0) GO TO 6
      ZBAR = (1.-DELT/TC)*ZBAR+DELT*Z/TC
      PHIBAR = (1.-DELT/TC)*PHIBAR+DELT*PHI/TC
      THEBAR = (1.-DELT/TC)*THEBAR+DELT*THETA/TC
      CALL WAVES (TIME)
    6 CALL SIDEWL
      CALL PROP
      CALL RUDDER
      CALL AEROD
      CALL INTGRL (TIME)
      IF (TIME.GT.FTIME) GO TO 10
      IF (FN.GT.FNCRIT) GO TO 7
      PRINT 17
      GO TO 10
    7 DELOLD = TIME-TOLD
      PSI = PSI+DELOLD*R
      X = X+DELOLD*(U*COS(PSI)-V*SIN(PSI))
      Y = Y+DELOLD*(U*SIN(PSI)+V*COS(PSI))
      IF (ABS(TIME-TPRINT).LT.1.E-6) GO TO 8
      GO TO 5
    8 CONTINUE
      IF (ITRAJ.EQ.0) GO TO 9
      DPHI = PHI*RAD
      DPSI = PSI*RAD
      DTHETA = THETA*RAD
      DP = P*RAD
      DQ = Q*RAD
      DR = R*RAD
      VEL = 0.5925*U
      WRITE (6,15) TIME,VEL,V,W,DP,DQ,DR,Z,DPHI,DTHETA,X,Y,DPSI
      BETS = (-V/U)*RAD
      DELRS = RUDANG*RAD
      WRITE (6,16) BETS,DELRS,FXP
    9 CONTINUE
      IMMTAG = (IMM+1)/2
      IF (IMMTAG.EQ.1.AND.TIME.GE.BTIME-1.E-8) IMT=1
      TPRINT = TPRINT+DELPNT
      CN = 1
      GO TO 5
   10 CALL COLFIL
      IF (IMM.LT.1) GO TO 3
      IF (IMM.NE.1) GO TO 11
      END FILE IBMFIL
      GO TO 3
   11 CALL SAM
      GO TO 3
C
   12 FORMAT (1H1//35X,22H LISTING OF INPUT DECK//)
```

```
13 FORMAT (20A4)
14 FORMAT (5X,20A4)
15 FORMAT (//10X,13HTIME (SEC) = F6.2//10X,33HTRANSLATIONAL VELS (KTS
  1)/(FT/SEC)/10X,2HU=F6.2,5X,2HV=F6.2,5X,2HW=F6.3//10X,31HROTATIONAL
  2 VELOCITIES (DEG/SEC)/10X,2HP=F6.2,5X,2HQ=F6.2,5X,2HR=F6.2//10X,30
  3HDISPLACEMENTS (FT AND DEGREES)/10X,2HZ=F7.3,5X,4HPHI=F6.2,3X,6HTH
  4ETA=F6.2//10X,27HTRAJECTORY (FT AND DEGREES)/10X,2HX=F8.2,4X,2HY=F
  58.2,4X,4HPSI=F8.2)
16 FORMAT (1H0,9X,23HSIDESLIP ANGLE (DEG) = F8.2,10X,21HRUDDER ANGLE
  1(DEG) = F8.3,10X,15HTHRUST (LBS) = F12.1)
17 FORMAT (///25X,28HCRAFT SPEED BELOW HUMP SPEED)
   END

      SUBROUTINE AERODC
C
      INTEGER ON
      COMMON /FAERO/ FX,FY,FZ,FK,FM,FN
      COMMON /FAIR/ RHOA,XLAERO
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /PRTINT/ ON,IACCEL,IVFL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /VARBLE/ VAL(40)
      EQUIVALENCE (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAL(5),P), (VAL(6
     1),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAL(10),Z), (VAL(
     211),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI), (VAL(24),PB)
C
      QA = RHOA*U*U
      QAL = QA*XLAERO
      BETA = -V/U
      BETASQ = BETA*BETA
      FX = -(0.90*BETASQ+0.13)*QA
      FY = (0.0*BETASQ+0.53*BETA)*QA
      FZ = -(2.06*BETASQ+0.39)*QA
      FK = -(0.5*BETASQ+0.0*BETA)*QAL
      FM = (0.29*BETASQ+0.12)*QAL
      FN = (0.0*BETASQ+0.076*BETA)*QAL
      IF (IAEROD.NE.ON) RETURN
      WRITE (6,1) FX,FY,FZ,FK,FM,FN
C
C
      RETURN
C
    1 FORMAT (/10X,23HAEROD FX,FY,FZ,FK,FM,FN/6E15.4)
      END

      BLOCK DATA
      COMMON /AIR/ Z1(3)
      COMMON /BMCO/ Z2(25)
```

```
      COMMON /COLUMN/ Z3(2)                                             BLDA  40
      COMMON /CONST/ Z4(3)                                              BLDA  50
      COMMON /CNTRL/ Z5(10)                                             BLDA  60
      COMMON /ENGINE/ Z6(107)                                           BLDA  70
      COMMON /EQNCO/ Z7(22)                                             BLDA  80
      COMMON /FAERO/ Z8(6)                                              BLDA  90
      COMMON /FAIR/ Z9(2)                                               BLDA 100
      COMMON /FANMAP/ Z10(262)                                          BLDA 110
      COMMON /FORBS/ Z11(7)                                             BLDA 120
      COMMON /FORSS/ Z12(8)                                             BLDA 130
      COMMON /FPROP/ Z13(6)                                             BLDA 140
      COMMON /FROUDE/ Z14(2)                                            BLDA 150
      COMMON /FRUD/ Z15(6)                                              BLDA 160
      COMMON /GBOW/ Z16(1)                                              BLDA 170
      COMMON /GEOM/ Z17(138)                                            BLDA 180
      COMMON /GEOMBS/ Z18(62)                                           BLDA 190
      COMMON /GEOMSS/ Z19(62)                                           BLDA 200
      COMMON /GEOMSW/ Z20(11)                                           BLDA 210
      COMMON /KSWTCH/ Z21(1)                                            BLDA 220
      COMMON /LEAKER/ Z22(4)                                            BLDA 230
      COMMON /MASSES/ Z23(817)                                          BLDA 240
      COMMON /MATRIX/ Z24(36)                                           BLDA 250
      COMMON /MSIDW/ Z25(55)                                            BLDA 260
      COMMON /MWAVE/ Z26(12)                                            BLDA 270
      COMMON /OPTION/ Z27(4)                                            BLDA 280
      COMMON /PLENUM/ Z28(7)                                            BLDA 290
      COMMON /PRIME/ Z29(5)                                             BLDA 300
      COMMON /PRTINT/ Z30(12)                                           BLDA 310
      COMMON /PWAVE/ Z31(2)                                             BLDA 320
      COMMON /RISER/ Z32(1)                                             BLDA 330
      COMMON /ROLL/ Z33(2)                                              BLDA 340
      COMMON /RUDDR/ Z34(62)                                            BLDA 350
      COMMON /SIDE/ Z35(22)                                             BLDA 360
      COMMON /SOFTBS/ Z36(20)                                           BLDA 370
      COMMON /SOFTSS/ Z37(19)                                           BLDA 380
      COMMON /STABLE/ Z38(5)                                            BLDA 390
      COMMON /STSLR/ Z39(2)                                             BLDA 400
      COMMON /VALOLD/ Z40(20)                                           BLDA 410
      COMMON /VARBLE/ Z41(40)                                           BLDA 420
      COMMON /WAVE/ Z42(100)                                            BLDA 430
      COMMON /WAVEF/ Z43(80)                                            BLDA 440
      COMMON /SLOPE/ Z44(5)                                             BLDA 450
      COMMON /PROMOD/ Z45(7)                                            BLDA 460
      DATA Z1/3*0.0/                                                    BLDA 470
      DATA Z2/25*0.0/                                                   BLDA 480
      DATA Z3/2*0.0/                                                    BLDA 490
      DATA Z4/3*0.0/                                                    BLDA 500
      DATA Z5/10*0.0/                                                   BLDA 510
```

```
      DATA Z6/107*0.0/                                            BLDA 520
      DATA Z7/21*0.0/                                             BLDA 530
      DATA Z8/6*0.0/                                              BLDA 540
      DATA Z9/2*0.0/                                              BLDA 550
      DATA Z10/262*0.0/                                           BLDA 560
      DATA Z11/7*0.0/                                             BLDA 570
      DATA Z12/8*0.0/                                             BLDA 580
      DATA Z13/6*0.0/                                             BLDA 590
      DATA Z14/2*0.0/                                             BLDA 600
      DATA Z15/6*0.0/                                             BLDA 610
      DATA Z16/0.0/                                               BLDA 620
      DATA Z17/138*0.0/                                           BLDA 630
      DATA Z18/62*0.0/                                            BLDA 640
      DATA Z19/62*0.0/                                            BLDA 650
      DATA Z20/11*0.0/                                            BLDA 660
      DATA Z21/0.0/                                               BLDA 670
      DATA Z22/4*0.0/                                             BLDA 680
      DATA Z23/817*0.0/                                           BLDA 690
      DATA Z24/36*0.0/                                            BLDA 700
      DATA Z25/55*0.0/                                            BLDA 710
      DATA Z26/12*0.0/                                            BLDA 720
      DATA Z27/4*0.0/                                             BLDA 730
      DATA Z28/7*0.0/                                             BLDA 740
      DATA Z29/5*0.0/                                             BLDA 750
      DATA Z30/12*0.0/                                            BLDA 760
      DATA Z31/2*0.0/                                             BLDA 770
      DATA Z32/0.0/                                               BLDA 780
      DATA Z33/2*0.0/                                             BLDA 790
      DATA Z34/62*0.0/                                            BLDA 800
      DATA Z35/22*0.0/                                            BLDA 810
      DATA Z36/20*0.0/                                            BLDA 820
      DATA Z37/19*0.0/                                            BLDA 830
      DATA Z38/5*0.0/                                             BLDA 840
      DATA Z39/2*0.0/                                             BLDA 850
      DATA Z40/20*0.0/                                            BLDA 860
      DATA Z41/40*0.0/                                            BLDA 870
      DATA Z42/100*0.0/                                           BLDA 880
      DATA Z43/80*0.0/                                            BLDA 890
      DATA Z44/5*0.0/                                             BLDA 900
      DATA Z45/7*0.0/                                             BLDA 910
      END                                                         BLDA 920
      SUBROUTINE BOWSL                                            BWSL  10
C                                                                 BWSL  20
      INTEGER CN                                                  BWSL  30
      COMMON /AIR/ PINF,RHOINF,GAM                                BWSL  40
      COMMON /CONST/ PI,RAD,UO                                    BWSL  50
      COMMON /FORBS/ FX,FY,FZ,FK,FM,FN,QL                         BWSL  60
```

```
      COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,
     110),XCP,ZCP
      COMMON /GEOMBS/ DETABX(11),DETABT(11),ARM1B(10),ARM2B(10),DFBS(10)
     1,TSKIB(10)
      COMMON /LEAKER/ ALEAK,BLEAK,CFSS,CFBS
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AM
     1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /SLOPE/ WATSLP,XPWV,XLXPWV,PWVHT,XPWVXS
      COMMON /SOFTBS/ XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAV
     1GB(10),CENCAB
      COMMON /VARBLE/ VAL(40)
      COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FY
     1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PB
     2BAR
      DIMENSION FLSKID(11), WETLEN(11), BWSL(6,24), GAP(11), ELSKI(11),
     1DPFT(11)
      DATA ENU,HINGHT/1.28E-5,1.875/
      DATA BWSL/0.0,3.8,6.9,9.9,12.3,15.0,0.0,4.1,7.3,10.5,13.2,15.7,0.0
     1,4.3,7.7,11.0,13.7,16.7,0.0,4.7,8.2,11.5,14.5,17.4,0.0,5.2,8.9,12.
     24,15.4,18.3,0.0,5.5,9.6,13.3,16.1,19.6,0.0,5.8,9.9,13.8,17.1,20.6,
     30.0,6.1,10.6,14.7,18.4,21.9,0.0,6.4,11.0,15.3,19.3,23.3,0.0,6.6,11
     4.7,16.4,20.5,24.5,0.0,6.9,12.3,17.6,22.1,25.7,0.0,7.2,13.5,19.6,24
     5.1,28.3,0.0,8.2,15.3,21.7,26.3,31.3,0.0,9.6,17.5,24.2,29.7,34.6,0.
     60,11.9,20.0,26.7,32.3,38.0,0.0,14.0,22.5,30.0,35.8,41.0,0.0,17.5,2
     75.5,32.5,39.7,44.5,0.0,21.0,30.0,37.4,42.0,47.0,0.0,24.6,33.7,41.0
     8,45.0,48.7,0.0,29.4,38.5,45.0,48.7,48.7,0.0,33.0,45.0,48.7,48.7,48
     9.7,0.0,40.0,48.7,48.7,48.7,48.7,0.0,48.7,48.7,48.7,48.7,48.7,0.0,4
     $8.7,48.7,48.7,48.7,48.7/
      DATA CORLEN/3.75/
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VA
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VA
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)
     3, (VAL(24),PB)
C
C
      DO 1 J=1,11
      GAP(J) = 0.0
      ELSKI(J) = 0.0
      WETLEN(J) = 0.0
      ELSKID(J) = 0.0
    1 CONTINUF
C
      ALBS = 0.0
      FX = 0.0
      FZ = 0.0
```

```
      FK = 0.0
      FM = 0.0
      FN = 0.0
      DELPBG = PBS-PB
      IF (DELPBG.LT.0.0) DELPBG=0.0
      PBAR = PB-PINF
      DELP = PBAR
      IF (DELP.LT.0.0) DELP=0.0
      ARGO = ELMAXB/CORLEN
      ANGO = ARSIN(ARGO)
      X1 = XBS+ZBS*THETA-CORLEN*COS(ANGO)
      Z1 = -Z-ZBS+XBS*THETA-ELMAXB*COS(THETA)
      DPTHFT = (5.5/(1.+(U/25.))**2)*0.0833
      IF (CENCAB.GT.1.1875) CENCAB=1.1875
      N = NSTA(3)
C
      DO 3 K=1,N
      DPFT(K) = DPTHFT
      ELSKI(K) = (ETA(3,K)-DETABX(K)*(XX(3,K)-X1)-Z1)+YY(3,K)*PHI+XLXPWV
     1*WATSLP
      IF (ELSKI(K).GT.HINGHT) ELSKI(K)=HINGHT
      IF ((HINGHT-ELSKI(K)+DPFT(K)).GE.ELMAXB) DPFT(K)=ELMAXB-HINGHT+ELS
     1KI(K)
      IF (DPFT(K).LT.0.0) DPFT(K)=0.0
      ELSKID(K) = (ELSKI(K)-DPFT(K))*12.0
      IF ((HINGHT-ELSKID(K)/12.0).GE.ELMAXB) ELSKID(K)=(HINGHT-ELMAXB)*1
     12.0
      DPIN = DPFT(K)*12.0
      MM = DPIN
      MM1 = MM+1
      MM2 = MM1+1
      DINC = DPIN-MM
      GAP(K) = -ELSKI(K)+(HINGHT-ELMAXB)
      IF (GAP(K).LT.0.0) GAP(K)=0.0
      IF (ELSKID(K).GE.0.) GO TO 2
      WETLEN(K) = ELSKI(K)
      GO TO 3
    2 MM3 = ELSKID(K)
      MM4 = MM3+1
      MM5 = MM4+1
      DLINC = ELSKID(K)-MM3
      BWSL1 = BWSL(MM1,MM4)
      BWSL2 = BWSL(MM1,MM5)
      BWSL3 = BWSL(MM2,MM4)
      BWSL4 = BWSL(MM2,MM5)
      BWSLA1 = (BWSL2-BWSL1)*DLINC+BWSL1
      BWSLA2 = (BWSL4-BWSL3)*DLINC+BWSL3
      WETLEN(K) = ((BWSLA2-BWSLA1)*DINC+BWSLA1)*0.08333
```

```
    3 CONTINUE
C
      N = NSTA(3)-1
C
      DO 10 J=1,N
      WETLAV = (WETLEN(J+1)+WETLEN(J))*0.5
      IF (WETLAV.LE.0.001) GO TO 8
      DPFTAV = (DPFT(J+1)+DPFT(J))*0.5
      ELSKIA = (ELSKI(J+1)+ELSKI(J))*0.5
      ELSKDA = (ELSKID(J+1)+ELSKID(J))*0.04166
      SEALHT = HINGHT-ELSKDA
      DIFF = 2.0*CENCAB-(SEALHT+0.5)
      IF (DIFF.GT.0.5) DIFF=0.5
      ARM1B(J) = X1+WETLAV*0.5
      ARM2B(J) = ZS-ELSKIA+DPFTAV*0.5
      IF (DIFF.GE.0.25) GO TO 4
      DFBS(J) = -DELP*DELYBS*WETLAV
      GO TO 7
    4 FORLEN = XBF-WETLAV
      IF (FORLEN.EQ.0.0) GO TO 5
      ARGW = (HINGHT-ELSKIA)/FORLEN
      IF (ARGW.GT.1.0) ARGW=1.0
      ANGW = ARSIN(ARGW)
      FORCOS = COS(ANGW)
      GO TO 6
    5 FORCOS = 0.0
    6 DFBS(J) = -DELP*DELYBS*WETLAV-DELP*FORLEN*DELYBS*FORCOS*((FORLEN*0
     1.5*FORCOS)/(FORLEN*FORCOS+WETLAV/2.0))*((DIFF-0.25)*4.0)
    7 ARG = 0.5*RHO*U*U*WETLAV*DELYBS
      RESKI = U*WETLAV/ENU
      CDTSKI = 0.427/(ALOG10(RESKI)-0.407)**2.64
      TSKIB(J) = -ARG*CDTSKI
      GO TO 9
    8 DFBS(J) = 0.0
      TSKIB(J) = 0.0
    9 CONTINUE
      FX = FX+TSKIB(J)
      FZ = FZ+DFBS(J)
      FK = FK+DFBS(J)*YAVGB(J)
      FM = FM-DFBS(J)*ARM1B(J)+TSKIB(J)*ARM2B(J)
      FN = FN-TSKIB(J)*YAVGB(J)
      ALBS = ALBS+(GAP(J)+GAP(J+1))*DELYBS*0.5
   10 CONTINUE
C
      ALBS = ALBS+BLEAK
      SQFAC = SQRT(2.*ABS(PBAR)/RHOINF)
      QL = CFBS*ALBS*SQFAC*SIGN(1.,PBAR)
      IF (IBCWSL.NE.ON) RETURN
```

```
      WRITE (6,11) GAP,WETLEN,FX,FY,FZ,FK,FM,FN
C
C
      RETURN
C
   11 FORMAT (//10X,8HBOW SEAL/26H GAP (FT.) (PORT TO STBD.)/11F10.5/28H
     1 WETLEN(FT.) (PORT TO STBD.)/11F10.5/10X,23HBOWSL FX,FY,FZ,FK,FM,F
     2N/6E15.4)
      END

      SUBROUTINE COLFIL
C
      COMMON /AXIS/ NXYS(26)
      COMMON /COLUMN/ IVERT,ILATRL
      COMMON /CURVE/ NCURV(10)
      COMMON /EQNCO/ NEQS,TOL(20),JQQ
      COMMON /GRAF/ NGRAF,NDRW
      COMMON /HEADG/ TICRD(6)
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /STEP/ STEP2
      COMMON /SUM/ ISUM1(8),ISUM2(8)
      REAL *8TICRD
      REAL LABEL
      REAL LAB(4)/'','1','2','3'/
      REAL *8NAMES(52)/'TIME','','WAVEHEI','GHT','ZDISPLA','CEMENT','PIT
     1CHAN','GLE','PLENUMP','RESSURE','BOWACCE','LERATION','C.G.ACCE','L
     2ERATION','FANPOWE','R(IDEAL)','ROLLANG','LE','YAWANGL','E','LATACC
     3E','LERATION','SPEEDTH','RUWATER','TURNRAD','IUS','PLENUMV','OLUME
     4','XDISPLA','CEMENT','YDISPLA','CEMENT','AIRFLOW','IN','AIRFLOW','
     5CUT','NETFORC','EXDIR','WAVEFOR','CEXDIR','THRUSTS','TARBOARD','TH
     6RUSTP','ORTSIDE','PITCH','RATE','ROLL','RATE','YAW','RATE','RUDDER
     7A','NGLE'/
      REAL *8TITLE(12)
      REAL *8LINE2(2)/'PLOTIS','VERSUS'/
      REAL *8NAMEX(2),NAMEY(2),INAME(16)
      EQUIVALENCE (TITLE(1),TICRD(1)), (TITLE(2),TICRD(2)), (TITLE(3),TI
     1CRD(3)), (TITLE(4),TICRD(4)), (TITLE(5),TICRD(5)), (TITLE(6),TICRD
     2(6))
      DIMENSION PVQQ(26), XOUT(900), YOUT(900), AFILE(8)
      DIMENSION THSTS(1), THSTP(1)
      EQUIVALENCE (PVQQ(1),TIME), (PVQQ(2),ETA), (PVQQ(3),Z), (PVQQ(4),T
     1HETA), (PVQQ(5),PB), (PVQQ(6),BOWACC), (PVQQ(7),ACC), (PVQQ(8),FAN
     2PWR), (PVQQ(9),PHI), (PVQQ(10),BETAS), (PVQQ(11),ACCLAT), (PVQQ(12
     3),U), (PVQQ(13),TRADUS), (PVQQ(14),VOLP), (PVQQ(15),X), (PVQQ(16),
     4Y), (PVQQ(17),QIN), (PVQQ(18),QOUT), (PVQQ(19),GFXXX), (PVQQ(20),F
     5XPWAV), (PVQQ(21),THSTS(1)), (PVQQ(22),THSTP(1)), (PVQQ(23),QDEG),
     6 (PVQQ(24),PDEG), (PVQQ(25),RDEG), (PVQQ(26),DELRS)
C
```

```
      IF (JQQ.NE.2) GO TO 1
      WRITE (6,23) STEP2
    1 END FILE 1
      REWIND 1
      TITLE(7) = LINE2(1)
      TITLE(10) = LINE2(2)
      IF (NGRAF.EQ.0) GO TO 7
      J = 1
      NGF = NGRAF
      INDEX = NGRAF*2
C
      DO 6 I=1,INDEX,2
      INDX = NXYS(I)
      INDY = NXYS(I+1)
      IQ = 0
    2 READ (1,END = 8) TIME,ETA,Z,THETA,PB,BOWACC,ACC,FANPWR,PHI,BETAS,A
     1CCLAT,U,TRADUS,VOLP,X,Y,QIN,QOUT,GFXXX,FXPWAV,THSTS(1),THSTP(1),QD
     2FG,PDEG,RDEG,DELRS
      IF (IQ.GE.900) GO TO 3
      IQ = IQ+1
      XOUT(IQ) = PVQQ(INDX)
      YOUT(IQ) = PVQQ(INDY)
      GO TO 2
    3 REWIND 1
      INX = INDX*2
      INY = INDY*2
      NAMEX(1) = NAMES(INX-1)
      NAMEX(2) = NAMES(INX)
      NAMEY(1) = NAMES(INY-1)
      NAMEY(2) = NAMES(INY)
      IF (NDRW.EQ.1) GO TO 4
C
      CALL PLOTP (XOUT,YOUT,-IQ,0)
C
      WRITE (6,24) NAMEX,NAMEY
      GO TO 6
    4 TITLE(8) = NAMEY(1)
      TITLE(9) = NAMEY(2)
      TITLE(11) = NAMEX(1)
      TITLE(12) = NAMEX(2)
      NCUR = NCURV(J)
      IF (NCUR.NE.0) GO TO 5
      LABEL = LAB(1)
C
      CALL DRAW (IQ,XOUT,YOUT,NCUR,0,LABEL,TITLE,0,0,0,0,0,0,8,8,0,LAST)
C
      J = J+1
      GO TO 6
```

```
    5 WRITE (2) IQ,NCUR,NGF,LABEL,(TITLE(K),K=1,12),(XOUT(L),YOUT(L),L=1
     1,IC)
      NGF = NGF-1
    6 CONTINUE
C
      IF ((NGF.EQ.0).AND.(NCUR.EQ.3)) GO TO 10
    7 IF (IVERT.NE.1) GO TO 16
      WRITE (6,25)
      K = 0
C
      DO 8 I=1,8
      IF (ISUM1(I).NE.0) K=K+1
    8 CONTINUE
C
      NUM1 = K
      IF (K.EQ.0) GO TO 16
      N = K*2
      J = 1
C
      DO 9 I=1,NUM1
      IDEX = ISUM1(I)*2
      INAME(J) = NAMES(IDEX-1)
      INAME(J+1) = NAMES(IDEX)
      J = J+2
    9 CONTINUE
C
      WRITE (6,26) (INAME(I),I=1,N)
      GO TO 14
   10 END FILE 2
      REWIND 2
      NGF = NGRAF
      J = 1
   11 READ (2,END = 38) IQ,NCUR,NGRAF,LABEL,(TITLE(K),K=1,12),(XOUT(L),Y
     1OUT(L),L=1,IQ)
      IF ((NGF.EQ.NGRAF).AND.(NCUR.EQ.J)) GO TO 12
      GO TO 11
   12 LABEL = LAB(J+1)
C
      CALL DRAW (IQ,XOUT,YOUT,NCUR,0,LABEL,TITLE,0,0,0,0,0,0,8,8,0,LAST)
C
      REWIND 2
      J = J+1
      IF (J.EQ.4) GO TO 13
      GO TO 11
   13 NGF = NGF-1
      IF (NGF.EQ.0) GO TO 7
      J = 1
      GO TO 11
```

```
   14 READ (1,END = 13) (PVQQ(I),I=1,26)
C
      DO 15 I=1,NUM1
      J = ISUM1(I)
   15 AFILE(I) = PVQQ(J)
C
      WRITE (6,27) (AFILE(I),I=1,NUM1)
      GO TO 14
      REWIND 1
   16 IF (ILATRL.NE.1) GO TO 22
      WRITE (6,28)
      K = 0
C
      DO 17 I=1,8
      IF (ISUM2(I).NE.0) K=K+1
   17 CONTINUE
C
      NUM2 = K
      IF (K.EQ.0) GO TO 21
      N = K*2
      J = 1
C
      DO 18 I=1,NUM2
      IDEX = ISUM2(I)*2
      INAME(J) = NAMES(IDEX-1)
      INAME(J+1) = NAMES(IDEX)
      J = J+2
   18 CONTINUE
C
      WRITE (6,26) (INAME(I),I=1,N)
   19 READ (1,END = 16) (PVQQ(I),I=1,26)
C
      DO 20 I=1,NUM2
      J = ISUM2(I)
   20 AFILE(I) = PVQQ(J)
C
      WRITE (6,27) (AFILE(I),I=1,NUM2)
      GO TO 19
   21 REWIND 1
   22 RETURN
C
   23 FORMAT ('0',4X,'THIS RUN USED VARIABLE STEP SIZE',/,'0',4X,'THE MI
     1NIMUM STEPSIZE RECORDED DURING THE RUN WAS',2X,E30.5)
   24 FORMAT (' ',20X,2A8,' IS THE INDEPENDENT VARIABLE AND ',2A8,' THE
     1DEPENDENT VARIABLE')
   25 FORMAT ('0',50X,'***SUMMARY ONE***',/,'0')
   26 FORMAT ('0',16A8)
   27 FORMAT ('0',8(3X,F10.2,3X))
```

```
   28 FORMAT ('0',50X,'***SUMMARY TWO***',/,'0')
      END
      SUBROUTINE FAN
C
      INTEGER ON
      COMMON /AIR/  PINF,RHOINF,GAM
      COMMON /FANMAP/ QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,EN
     1SFAN,BRPM,EMRPM,SRPM,NPTSB,NPTSM,NPTSS,PBFAN(25),PMFAN(25),PSFAN(2
     25),TMEB(25),DELB(25),NB,TMES(25),DELS(25),NS
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /SOFTBS/ XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAV
     1GB(10)
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS,ELMAXS,YAV
     1GS(10)
      COMMON /VARBLE/ VAL(40)
      DIMENSION QB(1), QM(1), QS(1), PBOW(1), PM(1), PS(1), HP(8)
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VA
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VA
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)
     3, (VAL(24),PB)
      EQUIVALENCE (VAL(18),FANPWR)
      EQUIVALENCE (QBFAN(1),QB(1)), (QMFAN(1),QM(1)), (QSFAN(1),QS(1)),
     1(PBFAN(1),PBOW(1)), (PMFAN(1),PM(1)), (PSFAN(1),PS(1))
      DATA HP/2.9,2.3,1.95,1.77,1.85,2.05,1.93,1.62/
C
      BRAT = 8000/BRPM
      EMRAT = 8000/EMRPM
      SRAT = 8000/SRPM
      TL = VAL(1)
      IF (NB.EQ.0.0) GO TO 1
      DPBS = FG1(TL,NB,TMEB,DELB,ILB)
      PBS = PB+DPBS
    1 IF (NS.EQ.0.0) GO TO 2
      DPSS = FG1(TL,NS,TMES,DELS,ILS)
      PSS = PB+DPSS
    2 CONTINUE
      PB1 = PBS-PINF
      PB2 = PB-PINF
      PB3 = PSS-PINF
      PBARB = PB1*BRAT*BRAT
      PBARM = PB2*EMRAT*EMRAT
      PBARS = PB3*SRAT*SRAT
      QBOW = ENBFAN*FG1(PBARB,NPTSB,PBOW,QB,IB)/BRAT
      QMAIN = ENMFAN*FG1(PBARM,NPTSM,PM,QM,IM)/EMRAT
      QSTN = ENSFAN*FG1(PBARS,NPTSS,PS,QS,IS)/SRAT
```

```
      QIN = QBOW+QMAIN+QSTN
      MB1 = (QBOW/ENBFAN+5.0)/5.0
      MB2 = MB1+1
      MB3 = MB2+1
      BINC = ((QBOW/ENBFAN+5.0)-MB1*5.0)/5.0
      BFANHP = ((HP(MB3)-HP(MB2))*BINC+HP(MB2))*FNBFAN*(1./BRAT)**3
      MS1 = (QSTN/ENSFAN+5.0)/5.0
      MS2 = MS1+1
      MS3 = MS2+1
      STINC = ((QSTN/ENSFAN+5.0)-MS1*5.0)/5.0
      SFANHP = ((HP(MS3)-HP(MS2))*STINC+HP(MS2))*ENSFAN*(1./SRAT)**3
      MM1 = (QMAIN/ENMFAN+5.0)/5.0
      MM2 = MM1+1
      MM3 = MM2+1
      PLINC = ((QMAIN/ENMFAN+5.0)-MM1*5.0)/5.0
      PFANHP = ((HP(MM3)-HP(MM2))*PLINC+HP(MM2))*ENMFAN*(1./EMRAT)**3
      RELPWR = PFANHP+BFANHP+SFANHP
      FANPWR = (QBOW*PB1+QMAIN*PB2+QSTN*PB3)/550.
      FANEFF = FANPWR/RELPWR
      IF (IRHS.NE.ON) RETURN
      WRITE (6,3) QBOW,QMAIN,QSTN,PBARB,PBARM,PBARS,RELPWR,FANPWR,FANEFF
C
C
      RETURN
C
    3 FORMAT (//4H FAN/32H Q - BOW,MAIN,STERN (CU FT /SEC)3F12.1/28H DEL
     1P - BOW,MAIN,STERN (PSF)3F11.2/60H ACTUAL FAN POWER REQUIRED(HP),
     2IDEAL FAN POWER, EFFICIENCY 3F12.4)
      END

      FUNCTION FG1 (X,N,XT,YT,IX)
C
      DIMENSION XT(1), YT(1)
C
      IF (IX.LT.1) IX = 1
      IF (IX.GT.N-1) IX = N-1
      I = SIGN(1.0,X-XT(IX))
    1 IF (IX.LT.1.OR.IX.GE.N) GO TO 3
      IF (XT(IX).GT.X.OR.X.GT.XT(IX+1)) GO TO 2
      C = (X-XT(IX))/(XT(IX+1)-XT(IX))
      GO TO 4
    2 IX = IX+I
      GO TO 1
    3 C = IX/N
      IX = IX-I
    4 FG1 = YT(IX)+C*(YT(IX+1)-YT(IX))
      RETURN
      END
```

```
      SUBROUTINE INCON (TIME)
C
      REAL *8TICRD
      INTEGER ON
      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON /AXIS/ NXYS(26)
      COMMON /ATRIX/ AMASSI,AIYYI,DIXX,DIXZ,DIZZ
      COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /COLUMN/ IVERT,ILATRL
      COMMON /CONST/ PI,RAD,UO
      COMMON /CNTRL/ CONTW,CONTQ,CONTTH,QMULT,LOUVER,ACONTZ,ACONTW,ZEQUI
     1L,THEQL,ACBASE
      COMMON /CURVE/ NCURV(10)
      COMMON /ENGINE/ NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,TIP
     1(25),TIS(25)
      COMMON /EQNCO/ NEQS,TOL(20),JQQ
      COMMON /FAIR/ RHOA,XLAERO
      COMMON /FANMAP/ QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,EN
     1SFAN,BRPM,EMRPM,SRPM,NPTSB,NPTSM,NPTSS,PBFAN(25),PMFAN(25),PSFAN(2
     25),TMEB(25),DELB(25),NB,TMES(25),DETS(25),NS
      COMMON /FROUDE/ FN,FNCRIT
      COMMON /GBOW/ XBOW
      COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,
     110),XCP,ZCP
      COMMON /GEOMSW/ XAVG(10),DS
      COMMON /GRAF/ NGRAF,NDRW
      COMMON /HEADG/ TICRD(6)
      COMMON /PWAVE/ FNCON,PWVCON
      COMMON /LEAKER/ ALEAK,BLEAK,CFSS,CFBS
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AM
     1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO
      COMMON /MATRIX/ A(6,6)
      COMMON /OPTION/ I3DOF,ISRGE,ITRIM,IDIA
      COMMON /PLENUM/ XLBW,XBBW,ABW,BUBHGT,DVWDOT,VMDOT,CMM
      COMMON /PLVCQQ/ NVI,NVD,NLI,NLD
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /PROMOD/ PROMO1,PROMO2, PROMO3, PROMO4, PROMO5, PROMO6, PROMO7
      COMMON /ROLL/ PHIMAX,TROLL
      COMMON /RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,R
     1CLB,RTC,RUDANG,TIR(25)
      COMMON /RISER/ AMPTC
      COMMON /SOFTBS/ XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAV
     1GB(10),CENCAB
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS,ELMAXS,YAV
     1GS(10)
```

```
      COMMON /SIDE/ FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSINCN
     1W,VAREA,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ,AVBMSW,DELX,VTC          INCN
      COMMON /SLOPE/ WATSLP,XPWV,XLXPWV,PWVHT,XPWVXS                     INCN
      COMMON /STABLE/ S(4),ISTAB                                         INCN
      COMMON /STSLR/ CPHI,CPHID                                          INCN
      COMMON /SUM/ ISUM1(8),ISUM2(8)                                     INCN
      COMMON /VALOLD/ YOLD(20)                                           INCN
      COMMON /VARBLE/ VAL(40)                                            INCN
      COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FYINCN
     1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBINCN
     2BAR                                                                INCN
      COMMON /WAVEF/ WAVLEN(20),OMEGAE(20),WAVSLP(20),ENCPER(20)         INCN
      COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB,AC INCN
     11(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),AC5(20,5,7),AC6(20,5INCN
     2,7),AC7(20,5,7),AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),AS1(20,5,7),AINCN
     3S2(20,5,7),AS3(20,5,7),AS4(20,5,7),AS5(20,5,7),AS6(20,5,7),AS7(20,INCN
     45,7),AS0(20,5,7),AS00(20,5,7),AS8(20,5,7),BB(36),XREF,RX           INCN
      DIMENSION ZZZ(14050)                                               INCN
      EQUIVALENCE (ZZZ,NAL)                                              INCN
      EQUIVALENCE (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAL(5),P), (VAL(6INCN
     1),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAL(10),Z), (VAL(INCN
     211),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI), (VAL(24),PB),INCN
     3 (VAL(12),VOL)                                                     INCN
      DIMENSION TEMP(7), XMO(10)                                         INCN
      DIMENSION TITLC(20)                                                INCN
      DATA BEAM,BETAD,DELO,DELPI,DLRDO,DSO,ISYS,RMAXO,RONO,RRATO,RREVO,TINCN
     1HETO,THSSI,TPRINO,UO,VXO,VZO,XBSI,XCPO,XLTOT,XPO,XRO,XSSI,YPO,ZPO,INCN
     2ZRO,ZSSI/6*0.0,0,20*0.0/                                           INCN
C                                                                        INCN
C     INITIAL CONDITIONS WITH WATSLP                                     INCN
C                                                                        INCN
C                                                                        INCN
      DO 1 I=1,8                                                         INCN
      ISUM1(I) = 0                                                       INCN
    1 ISUM2(I) = 0                                                       INCN
C                                                                        INCN
      PINF = 2116.                                                       INCN
      RHOINF = .002378                                                   INCN
      GAM = 1.4                                                          INCN
      GO TO 3                                                            INCN
    2 READ (5,98) NGRAF,NDRW                                             INCN
      READ (5,99) NXYS                                                   INCN
      READ (5,100) TICRD                                                 INCN
    3 READ (5,107) ISYSL,IOPT,(TEMP(I),I=1,7)                            INCN
      IF (ISYSL.EQ.ISYS.AND.ISYSL.EQ.13) GO TO 97                        INCN
      ISYS = ISYSL                                                       INCN
      IF ((ISYS.LE.0).OR.(ISYS.GT.22)) GO TO 97                          INCN
      GO TO (4,14,23,24,27,28,31,34,37,41,42,52,53,74,75,82,89,92,93,6,7INCN
```

```
     1,2), ISYS
C
C     PROGRAM CONTROL PARAMETERS
C
    4 CONTINUE
      GO TO (5,8,9,10,11,12), IOPT
    5 CONTINUE
      STIME = TEMP(1)
      FTIME = TEMP(2)
      DELO = TEMP(3)
      DELPNT = TEMP(4)
      TPRINO = TEMP(5)
      IF (TPRINO.LT.STIME+DELPNT) TPRINO=STIME+DELPNT
      IF (DELO.GT.DELPNT) DELO=DELPNT
      IF (DELO.EQ.0.0) GO TO 13
      GO TO 3
    6 READ (5,101) NCURV
      GO TO 3
    7 READ (5,102) ISUM1
      READ (5,102) ISUM2
      GO TO 3
    8 READ (5,108) IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IRUD,IP
     1ROP,IAEROD,IRHS
      GO TO 3
    9 READ (5,111) NEQS,JQQ,(TOL(J),J=1,NEQS)
      GO TO 3
   10 READ (5,108) IVERT,ILATRL,NVD,NVI,NLD,NLI
      GO TO 3
   11 CONTINUE
      I3DOF = TEMP(1)
      ISRGE = TEMP(2)
      ITRIM = TEMP(3)
      IDIA = TEMP(4)
      GO TO 3
   12 CONTINUE
      PROMO1 = TEMP(1)
      PROMO2 = TEMP(2)
      PROMO3 = TEMP(3)
      PROMO4 = TEMP(4)
      PROMO5 = TEMP(5)
      PROMO6 = TEMP(6)
      PROMO7 = TEMP(7)
      GO TO 3
   13 WRITE (6,110)
      STOP
C
C     MASS DISTRIBUTION
C
```

```
   14 G = 32.17                                                              INCN1440
      RHO = 1.99                                                             INCN1450
      HRHO = RHO*.5                                                          INCN1460
      GO TO (15,17,22), IOPT                                                 INCN1470
   15 IMM = 0                                                                INCN1480
      WEIGHT = TEMP(1)                                                       INCN1490
      AM = WEIGHT/G                                                          INCN1500
      XS = TEMP(2)                                                           INCN1510
      ZS = TEMP(3)                                                           INCN1520
      AIXX = TEMP(4)                                                         INCN1530
      AIYY = TEMP(5)                                                         INCN1540
      AIZZ = TEMP(6)                                                         INCN1550
      AIXZ = TEMP(7)                                                         INCN1560
C                                                                            INCN1570
C     INERTIA MATRIX OPERATIONS                                              INCN1580
C                                                                            INCN1590
   16 AMASSI = 1.0/AM                                                        INCN1600
      D = 1.0/(AIXX*AIZZ-AIXZ*AIXZ)                                          INCN1610
      DIXX = AIXX*D                                                          INCN1620
      DIXZ = AIXZ*D                                                          INCN1630
      DIZZ = AIZZ*D                                                          INCN1640
      AIYYI = 1.0/AIYY                                                       INCN1650
      GO TO 3                                                                INCN1660
C                                                                            INCN1670
C     READ WEIGHT DISTRIBUTION - ASSUME TRANSVERSE (PORT/STBD) SYMMETRY      INCN1680
C        X INPUT DIST. FWD. OF (SIDEWALL) TRANSOM                            INCN1690
C        Y INPUT DIST. TO STARBOARD                                          INCN1700
C        Z INPUT DIST. UP FROM KEEL-LINE                                     INCN1710
C                                                                            INCN1720
   17 I = 1                                                                  INCN1730
   18 READ (5,109) AMI(I),XI(I),YI(I),ZI(I)                                  INCN1740
      IF (AMI(I).LT.0.0) GO TO 19                                            INCN1750
      I = I+1                                                                INCN1760
      IF (I.GT.201) GO TO 97                                                 INCN1770
      GO TO 18                                                               INCN1780
   19 NMASS = I-1                                                            INCN1790
      SUM = 0.0                                                              INCN1800
      SUX = 0.0                                                              INCN1810
      SUZ = 0.0                                                              INCN1820
C                                                                            INCN1830
      DO 20 I=1,NMASS                                                        INCN1840
      AMI(I) = AMI(I)/G                                                      INCN1850
      SUM = SUM+AMI(I)                                                       INCN1860
      SUX = SUX+AMI(I)*XI(I)                                                 INCN1870
   20 SUZ = SUZ+AMI(I)*ZI(I)                                                 INCN1880
C                                                                            INCN1890
      AM = SUM*2.0                                                           INCN1900
      WEIGHT = AM*G                                                          INCN1910
```

```
      ZS = SUZ/SUM
      XS = SUX/SUM
      SUM = 0.0
      SUX = 0.0
      SUY = 0.0
      SUZ = 0.0
C
      DO 21 I=1,NMASS
      XI(I) = XI(I)-XS
      ZI(I) = -ZI(I)+ZS
      AMK = AMI(I)*2.0
      SUX = SUX+AMK*XI(I)*XI(I)
      SUY = SUY+AMK*YI(I)*YI(I)
      SUZ = SUZ+AMK*ZI(I)*ZI(I)
   21 SUM = SUM+AMK*XI(I)*ZI(I)
C
      AIXX = SUY+SUZ
      AIYY = SUX+SUZ
      AIZZ = SUX+SUY
      AIXZ = SUM
      GO TO 16
   22 GO TO 3
C
C     XX AND YY TABLES
C
   23 CONTINUE
      NSTA(1) = TEMP(1)
      NSTA(2) = TEMP(2)
      NSTA(3) = TEMP(3)
      NSTA(4) = TEMP(4)
      XLTOT = TEMP(5)
      GO TO 3
C
C     SIDEWALL (INCLUDING APPENDAGES)
C
   24 CONTINUE
      GO TO (25,26), IOPT
   25 YSW = TEMP(1)
      XLSW = TEMP(2)
      CFSW = TEMP(3)
      CDSW = TEMP(4)
      AVBMSW = TEMP(5)
      READ (10) ZZZ
      REWIND 10
      GO TO 3
C
C    BLOCK 4 OPTION 2 REMOVED.       NO APENDAGES
C
```

```
   26 CONTINUE
      GO TO 3
C
C     STERNSEAL
C
   27 CONTINUE
      XSSI = TEMP(1)
      ZSSI = TEMP(2)
      ALEAK = TEMP(3)
      CFSS = TEMP(4)
      ELMAXS = TEMP(5)
      DPSS = TEMP(6)
      XLF = TEMP(7)
      ARGA = ELMAXS/XLF
      COSTH = ARGA
      THSSI = ARCOS(ARGA)
      SINTH = SIN(THSSI)
      GO TO 3
C
C     BOWSEAL
C
   28 GO TO (29,30), IOPT
   29 CONTINUE
      XBSI = TEMP(1)
      CFBS = TEMP(2)
      DPBS = TEMP(3)
      ZBSI = TEMP(4)
      ELMAXB = TEMP(5)
      XBF = TEMP(6)
      BLEAK = TEMP(7)
      GO TO 3
   30 CENCAB = TEMP(1)
      GO TO 3
C
C     PLENUM
C
   31 CONTINUE
      GO TO (32,33), IOPT
   32 CONTINUE
      XLBW = TEMP(1)
      XBBW = TEMP(2)
      XPWV = TEMP(3)
      WIDTH = TEMP(4)
      XL = TEMP(5)
      XCPO = TEMP(6)
      BUBHGT = TEMP(7)
      XLXPWV = XLBW-XPWV
      PWVHT = (XPWV*XPWV-XLXPWV*XLXPWV)*.5/XL
```

```
      XPWVXS = XPWV-XS
      ABW = XBBW*XLBW
      AB = WIDTH*XL
      VCLNOM = (ABW+AB)*BUBHGT*.5
      GO TO 3
   33 CONTINUE
      FNCRIT = TEMP(1)
      GO TO 3
C
C     PROPULSICN
C
   34 CONTINUE
      GO TO (35,36), IOPT
   35 CONTINUE
      XPO = TEMP(1)
      YPO = TEMP(2)
      ZPO = TEMP(3)
      GO TO 3
C
C BLOCK 8 OPTION 2 REMOVED. ENGINE OUT INPUT IN BLOCK 16
C
   36 CONTINUE
      GO TO 3
C
C     RUDDER
C
   37 CCNTINUE
      GO TO (38,39,40), IOPT
   38 XRO = TEMP(1)
      YR = TEMP(2)
      ZRO = TEMP(3)
      RSPAN = TEMP(4)
      RASPR = TEMP(5)
      RAREA = TEMP(6)
      RCLB = 2.*PI*RASPR/(RASPR+3.)
      RTC = TEMP(7)
      GO TO 3
C
C     910 NOT USED
C
   39 CONTINUE
      GO TO 3
   40 CONTINUF
      GO TO 3
C
C     AERODYNAMICS
C
   41 CONTINUE
```

```
      XLAERO = TEMP(1)
      BEAM = TEMP(2)
      RHOA = .5*RHOINF*XLAERO*BEAM
      GO TO 3
C
C     WAVES
C
   42 CONTINUE
      IWAVSW = ICPT
      IF (IWAVSW.GT.4) GO TO 97
      NWAVE = TEMP(1)
      IF (NWAVE.EQ.0) GO TO 3
      IF (NWAVE.GT.20) GO TO 97
      BETAD = TEMP(2)
      BETA = BETAD/RAD
      COSBET = COS(BETA)
      SINBET = SIN(BETA)
      TC = 1.0
      GO TO (43,45,47,47), IWAVSW
C
   43 DO 44 I=1,NWAVE
   44 READ (5,116) OMEGA(I),AW(I)
C
      GO TO 3
C
   45 DO 46 I=1,NWAVE
   46 READ (5,116) WAVLEN(I),AW(I)
C
      GO TO 3
   47 SHTWV = TEMP(3)
      G1G = 32.17
      G2 = G1G*G1G
      G4 = G2*G2
      GO TO (3,3,48,49), IWAVSW
   48 CONTINUE
      PERL = TEMP(4)
      PERH = TEMP(5)
      WWN = (2.0*3.141592)*(1.0/PERH)
      WWX = (2.0*3.141592)*(1.0/PERL)
      GO TO 50
   49 CONTINUE
      WWN = TEMP(4)
      WWX = TEMP(5)
   50 CONTINUE
      UUU=SQRT(SHTWV*54.0)*1.6878
      UU4 = UUU*UUU*UUU*UUU
      CCC = (WWX/WWN)**(1./NWAVE)
      WWPO = WWN
```

```
C
      DO 51  I=1,NWAVE
      WWPN = WWPO*CCC
      WW = (WWPN+WWPO)*.5
      DDW = WWPN-WWPO
      WWPO = WWPN
      WW4 = WW*WW*WW*WW
      WW5 = WW*WW4
      SS = 0.0081*G2/(EXP(0.74*G4/(WW4*UU4))*WW5)
      OMEGA(I) = WW
      AW(I) = SQRT(2.*SS*DDW)
   51 CONTINUE
C
      GO TO 3
C
C     INITIAL CONDITIONS
C
   52 CONTINUE
      UO = TEMP(1)
      THETO = TEMP(2)
      DSO = TEMP(3)
      DELPI = TEMP(4)
      DPHI = TEMP(5)
      GO TO 3
   53 CONTINUE
C
C     INPUT COMPLETED. 1) PRINT ALL INPUT
C
      WRITE (6,123) TITLC
      WRITE (6,124) STIME,FTIME,DELO,TPRINO,DELPNT
      WRITE (6,125) IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IRUD,I
     1PROP,IAEROD,IRHS
      WRITE (6,135) I3DOF,ISRGE,ITRIM,IDIA
      WRITE (6,142) PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      WRITE (6,126) NEQS,(TOL(J),J=1,NEQS)
      WRITE (6,113) WEIGHT,XS,ZS,AIXX,AIYY,AIZZ,AIXZ
      WRITE (6,112) A,AIMAX
      WRITE (6,133) NSTA
      WRITE (6,114) YSW,XLSW,CFSW,CDSW,VANGLE,VSPAN,VCHORD,VXO,VY,VZO,AV
     1BMSW,VTC
      WRITE (6,115) NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB
      IF (IMM.GT.0) WRITE (6,121) (XMO(J),J=1,IMNX)
      WRITE (6,119) IMM,IMNX,IMNY,IBMFIL,BTIME,IMT
      IF (IMM.GT.0) WRITE (6,122) (YMI(J),J=1,IMNY)
      WRITE (6,128) XLBW,XBBW
      WRITE (6,129) XL,WIDTH,XCPO,VOLNOM,BUBHGT
      WRITE (6,134) DELPI
      WRITE (6,127) FNCRIT,XLTOT
```

```
      WRITE (6,141) ENBFAN,BRPM,ENMFAN,EMRPM,ENSFAN,SRPM
      WRITE (6,131) XRO,YR,ZRO,RONO,RMAXO,RRATO,RREVO,DLRDO,RSPAN,RASPR,
     1RAREA,RCLB,RTC
      WRITE (6,130) XPO,YPO,ZPO
      WRITE (6,140) XLAERO,BEAM
      WRITE (6,139) XBSI,CFBS,DPBS,ELMAXB
      WRITE (6,138) XSSI,ZSSI,ALEAK,CFSS,ELMAXS,DPSS,XLF
      WRITE (6,132) UO,THETO,DSO
C
C                   AND 2) INITIALIZE VARIABLES FOR CALCS.
C
C
      DO 54 I=1,40
   54 VAL(I) = 0.0
C
      U = UO*1.6889
      XSS = -(XS-XSSI)
      ZSS = ZS-ZSSI
      THETA = THETO/RAD
      PHI = DPHI/RAD
      THEQL = THETA
      DS = DSO*.0833
      Z = -ZS+DS
      ZEQUIL = Z
      PHIMAX = 0.
      TROLL = 0.
      IRDS = 0
      TL = 0.0
C
C     WAVE PARAMETERS TABLE
C
      IF (NWAVE.EQ.0) GO TO 61
      AMPTC = 1.30287
      GO TO (55,57), IWAVSW
C
   55 DO 56 I=1,NWAVE
   56 WAVLEN(I) = 2.*PI*G/(OMEGA(I)*OMEGA(I))
C
      GO TO 59
C
   57 DO 58 I=1,NWAVE
   58 OMEGA(I) = SQRT(2.*PI*G/WAVLEN(I))
C
   59 CONTINUE
C
C     CALCULATE INITIAL FREQUENCIES OF ENCOUNTER
C
C
```

```
      DO 60  I=1,NWAVE
      WAVSLP(I) = 360.0*AW(I)/WAVLEN(I)
      CMEGAE(I) = 2.*PI*(SQRT(G*WAVLEN(I)/(2.*PI))-U*COSBET)/WAVLEN(I)
      ENCPER(I) = 2.0*PI/OMEGAE(I)
   60 CONTINUE
C
      WRITE (6,117) NWAVE,BETAD,(OMEGA(I),OMEGAE(I),WAVLEN(I),AW(I),WAVS
     1LP(I),ENCPER(I),I=1,NWAVE)
      GO TO 62
   61 WRITE (6,118)
   62 CONTINUE
C
      DO 63  I=1,4
C
      DO 63  N=1,11
   63 ETA(I,N) = 0.0
C
      DVOLW = 0.0
      FXWAV = 0.0
      FYWAV = 0.0
      FZWAV = 0.0
      FKWAV = 0.0
      FMWAV = 0.0
      FNWAV = 0.0
      ZBAR = Z
      PHIBAR = PHI
      THEBAR = THETA
      TIME = STIME
      DELT = DELC
      TPRINT = TPRINO-DELPNT
      PWVCON = 4.*WEIGHT/(RHO*G*XLBW)
      FNCON = SQRT(XLBW*G)
      VX = VXO-XS
      VZ = ZS-VZO
      XP = XPO-XS
      XR = XRO-XS
      YP = YPO
      ZP = ZS-ZPO
      ZR = ZS-ZRO
      IF (INM.EQ.0) GO TO 65
C
      DO 64  J=1,IMNX
   64 XMI(J) = XMO(J)-XS
C
   65 CONTINUE
      XCP = XCPO-XS
      ZCP = ZS-BUBHGT
      XBS = XBSI-XS
```

```
      N = NSTA(3)
      ZBS =  ZS-ZBSI
C
      DO 66  J=1,N
      DELYBS  =  XBBW/(N-1)
      XX(3,J)  =  XBS-XSSI
      YY(3,J)  =  -0.5*XBBW+(J-1)*DELYBS
   66 CONTINUE
C
      N  =  N-1
C
      DO 67  J=1,N
      YAVGB(J)  =  (YY(3,J+1)+YY(3,J))*.5
   67 CONTINUE
C
      N  =  NSTA(4)
      DELYSS  =  XBBW/(N-1)
C
      DO 68  J=1,N
      XX(4,J)  =  -XS
      YY(4,J)  =  -.5*XBBW+(J-1)*DELYSS
   68 CONTINUE
C
      N  =  N-1
C
      DO 69  J=1,N
      YAVGS(J)  =  (YY(4,J+1)+YY(4,J))*.5
   69 CCNTINUE
C
      XBOW  =  XLTOT-XS
      N  =  NSTA(1)
      DELX  =  XBSI/(N-1)
C
      DO 70  J=1,2
C
      DO 70  I=1,N
      XX(J,I)  =  (I-1)*DELX-XS
   70 YY(J,I)  =  YSW*(2*J-3)
C
      WRITE (6,103) ((XX(J,N),N=1,11),(YY(J,N),N=1,11),J=1,4)
      N  =  NSTA(1)-1
C
      DO 71  I=1,N
   71 XAVG(I)  =  DELX*(2*I-1)*.5-XS
C
C
      CALL  WAVES (TIME)
C
```

```
C      INITIALIZE BUBBLE PRESSURE, ABSOLUTE (PSF)
C
       PB = PINF+DELPI
       PBBAR = DELPI
       PBAR = DELPI
       PSS = PB+DPSS
       PBS = PB+DPBS
       AB = ABW-(ABW-AB)*(ZS+Z/BUBHGT)
       CF = .37/((U/FNCON)**1.5655981)
       WATSLP = PBBAR*CF*PWVCON/WEIGHT
       IF (IDIA.EQ.1) GO TO 72
       VOL = VOLNOM-.5*(AB+ABW)*(Z+ZS)-DVOLW+.5*WATSLP*XL*AB
       GO TO 73
    72 VOL = VOLNOM-.5*(AB+ABW)*(Z+ZS)-DVOLW+PBAR*.3175+.5*WATSLP*XL*AB
    73 CONTINUE
       BMASS = (PB/PINF)**(1./GAM)*VOL*RHOINF
       WRITE (6,137)
       DVWDOT = 0.0
       RETURN
C
C      RUN TERMINATOR
C
    74 WRITE (6,106)
       STOP
C
C      BENDING MOMENT
C
    75 GO TO (76,77,79,81), IOPT
    76 IMM = TEMP(1)
       IF (IMM.GT.3) GO TO 97
       IMNX = TEMP(2)
       IF (IMNX.GT.10) GO TO 97
       IMNY = TEMP(3)
       IF (IMNY.GT.7) GO TO 97
       IBMFIL = TEMP(4)
       BTIME = TEMP(5)
       IF (IMM.EQ.3) IMT = TEMP(6)
       GO TO 3
C
    77 DO 78 J=1,7
    78 XMO(J) = TEMP(J)
C
       IF (IMNX.LE.7) GO TO 3
       READ 120, (XMO(J),J=8,IMNX)
       GO TO 3
C
    79 DO 80 J=1,IMNY
    80 YMI(J) = TEMP(J)
```

```
C
      GO TO 3
   81 CONTINUE
      GO TO 3
   82 CONTINUE
      GO TO (83,85,87), IOPT
   83 CONTINUE
C
C  VALUES INPUT FOR STBD SCREW
C
      THST1 = TEMP(1)
      NPS = TEMP(2)
      STHS = TEMP(3)
      IF (NPS.EQ.0.0) GO TO 84
      READ (5,104) (TIS(J),J=1,NPS)
      READ (5,104) (THSTS(J),J=1,NPS)
      GO TO 3
   84 THSTS(1) = THST1
   85 CONTINUE
C
C  VALUES INPUT FOR PORT SCREW
C
      THST2 = TEMP(1)
      NPP = TEMP(2)
      STHP = TEMP(3)
      IF (NPP.EQ.0.0) GO TO 86
      READ (5,104) (TIP(J),J=1,NPP)
      READ (5,104) (THSTP(J),J=1,NPP)
      GO TO 3
   86 THSTP(1) = THST2
   87 CONTINUE
C
C  VALUES INPUT FOR RUDDER
C
      DELR = TEMP(1)
      NPR = TEMP(2)
      IF (NPR.EQ.0.0) GO TO 88
      READ (5,104) (TIR(J),J=1,NPR)
      READ (5,104) (DELRUD(J),J=1,NPR)
      GO TO 3
   88 DELRUD(1) = DELR
      GO TO 3
   89 GO TO (90,91), IOPT
   90 NB = TEMP(1)
      READ (5,104) (TMEB(I),I=1,NB)
      READ (5,104) (DELB(I),I=1,NB)
      GO TO 3
   91 NS = TEMP(1)
```

```
      READ (5,104) (TMES(I),I=1,NS)
      READ (5,104) (DETS(I),I=1,NS)
      GO TO 3
C
C     TITLE CARD (ALL 80 COLUMNS )
C
   92 READ (5,136) TITLC
      GO TO 3
C
C     FAN MAPS
C
   93 CONTINUE
      GO TO (94,95,96), IOPT
   94 CONTINUE
      ENBFAN = TEMP(1)
      BRPM = TEMP(2)
      NPTSB = TEMP(3)
      READIN = TEMP(4)
      IF (READIN.EQ.0.0) GO TO 3
      READ (5,104) (PBFAN(J),J=1,NPTSB)
      READ (5,104) (QBFAN(J),J=1,NPTSB)
      GO TO 3
   95 CONTINUE
      ENMFAN = TEMP(1)
      EMRPM = TEMP(2)
      NPTSM = TEMP(3)
      READIN = TEMP(4)
      IF (READIN.EQ.0.0) GO TO 3
      READ (5,104) (PMFAN(J),J=1,NPTSM)
      READ (5,104) (QMFAN(J),J=1,NPTSM)
      GO TO 3
   96 CONTINUE
      ENSFAN = TEMP(1)
      SRPM = TEMP(2)
      NPTSS = TEMP(3)
      READIN = TEMP(4)
      IF (READIN.EQ.0.0) GO TO 3
      READ (5,104) (PSFAN(J),J=1,NPTSS)
      READ (5,104) (QSFAN(J),J=1,NPTSS)
      GO TO 3
C
C     ERROR IN INPUT
C
   97 CONTINUE
      WRITE (6,105) ISYS
      STOP
C
C
```

```
 98 FORMAT (2I2)                                                         INCN7200
 99 FORMAT (26I2)                                                        INCN7210
100 FORMAT (6A8)                                                         INCN7220
101 FORMAT (10I1)                                                        INCN7230
102 FORMAT (8I2)                                                         INCN7240
103 FORMAT (//17H XX AND YY ARRAYS/14H PORT SIDEWALL/2(11F10.2/),15H SINCN7250
   1TBD. SIDEWALL/2(11F10.2/),9H BOW SEAL/2(11F10.2/),11H STERN SEAL/2INCN7260
   2(11F10.2/))                                                          INCN7270
104 FORMAT (8F10.0)                                                      INCN7280
105 FORMAT (34H INPUT ERROR - - - STOP - -  ISYS=,I3)                    INCN7290
106 FORMAT (1H1,20(/),50X,19H COMPLETED ALL RUNS)                        INCN7300
107 FORMAT (I3,I2,7F10.0)                                                INCN7310
108 FORMAT (16I5)                                                        INCN7320
109 FORMAT (5F10.0)                                                      INCN7330
110 FORMAT (///10X,65HERROR IN INPUT --- DELT AND/OR DELPNT EQUALS ZERINCN7340
   1O --- JOB ABORTED)                                                   INCN7350
111 FORMAT (2I2/(8F10.0))                                                INCN7360
112 FORMAT (22H INERTIA MATRIX, AIMAX6E15.4/(22X,6E15.4))                INCN7370
113 FORMAT (30H WEIGHT, C.G., INERTIA MOMENTS7F12.3)                     INCN7380
114 FORMAT (15H SIDEWALL INPUT12(F8.3,1X))                               INCN7390
115 FORMAT (26H SIDEWALL TABLE PARAMETERS4(I4,F7.3,F7.3))                INCN7400
116 FORMAT (2F10.0)                                                      INCN7410
117 FORMAT (/12HONO OF WAVESI2,10H BETA(DEG)F5.0/15H OMEGA(RAD/SEC),5XINCN7420
   1,16H OMEGAE(RAD/SEC)5X,16H WAVE LENGTH(FT)5X,14H AMPLITUDE(FT)5X,1INCN7430
   26HMAX  SLOPE (DEG)5X,13HPERIOD,E(SEC)/(F8.4,12X,F8.4,4F20.3))       INCN7440
118 FORMAT (/11HOCALM WATER)                                             INCN7450
119 FORMAT (32HOMOMENT CALC. CONTROL PARAMETERS4I5,F8.3,I5)              INCN7460
120 FORMAT (5X,7F10.0)                                                   INCN7470
121 FORMAT (22H MOMENT CALCS. AT X OF11F10.3)                            INCN7480
122 FORMAT (22H MOMENT CALCS. AT Y OF11F10.3)                            INCN7490
123 FORMAT (33H1SES MOTIONS AND LOADS PROGRAM - 20A4,/)                  INCN7500
124 FORMAT (23H START AND FINISH TIMES2F10.2/22H INITIAL TIME INTERVALINCN7510
   1F12.4/18H START PRINTING ATF8.2,17H IN INCREMENTS OFF8.2)           INCN7520
125 FORMAT (24H INTERMEDIATE PRINT TAGS16I5)                             INCN7530
126 FORMAT (39H NO. OF STATE EQUATIONS, AND TOLERANCESI5/(10X,10E12.2)INCN7540
   1)                                                                    INCN7550
127 FORMAT (23HOCRITICAL FROUDE NUMBERF15.4,5X,19H TOTAL CRAFT LENGTHFINCN7560
   115.4)                                                                INCN7570
128 FORMAT (34HOPLENUM, LENGTH AND WIDTH AT WATER2F12.4)                 INCN7580
129 FORMAT (34H PLENUM, LENGTH AND WIDTH AT HULL 2F12.4/35H PLENUM, CEINCN7590
   1NTER OF PRESSURE AT HULLF12.4/23H PLENUM, NOMINAL VOLUMEF12.1,10X,INCN7600
   26HHEIGHTF12.4)                                                       INCN7610
130 FORMAT (/33H PROPULSION, X, Y, Z COORDINATES 3F12.4/)                INCN7620
131 FORMAT (/28HORUDDER, X, Y, Z COORDINATES3F12.4/41H RUDDER, ON, MAXINCN7630
   1, RATE, REVERSE, INITIAL 5F12.4/33H RUDDER,SPAN,ASPFCT,AREA,CLB,T/INCN7640
   2C 5F12.4)                                                            INCN7650
132 FORMAT (/39HOINITIAL CONDITIONS, VELOCITY (KNOTS) =F7.2,5X,13HPITCINCN7660
   1H (DEG) =F8.3,5X,12HDRAFT (IN) =F8.2)                               INCN7670
```

```
133 FORMAT (49H NUMBER OF STATIONS, SIDEWALLS (P+S), SEALS (B+S),4I5)
134 FORMAT (38H PLENUM, INITIAL PRESSURE, GAGE (PSF) F8.2)
135 FORMAT (79H PROGRAM OPTION SWITCH SETTINGS (LATERAL PLANE,CONSTANT
   1 SPEED,TRIM,MEMBRANE)    7I5)
136 FORMAT (20A4)
137 FORMAT (1H1)
138 FORMAT (16H STERNSEAL INPUT7F12.4)
139 FORMAT (16H BOWSEAL INPUT  7F12.4)
140 FORMAT (19HOAERODYNAMICS INPUT7F12.4)
141 FORMAT (33HOFANS, NO. + RPM, BOW,MAIN,STERN 3(F10.0,F10.1))
142 FORMAT (32H PROGRAM MODIFICATION SETTINGS  7(F12.4,1X))
    END

    SUBROUTINE INTGRL (TIME)
    INTEGER ON
    COMMON /AIR/ PINF,RHOINF,GAM
    COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
    COMMON /EQNCO/ NEQS,TOL(20),JQQ
    COMMON /KSWTCH/ ITHRST
    COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AM
   1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO
    COMMON /OPTION/ I3DOF,ISRGE,ITRIM,IDIA
    COMMON /PLENUM/ XLBW,XBBW,ABW,BUBHGT,DVWDOT,VMDOT,PMDOT
    COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT
    COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
    COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
   1RUD,IPROP,IAEROD,IRHS
    COMMON /STABLE/ S(4),ISTAB
    COMMON /STEP/ STEP2
    COMMON /VALOLD/ YOLD(20)
    COMMON /VARBLE/ VAL(40)
    EQUIVALENCE (VAL(1),X), (VAL(2),Y(1))
    DIMENSION Y(20), ERROR(20)
    REAL K1(20),K2(20),K3(20),K4(20),K5(20)
    DATA IPASS/0/
    STEP2 = 1.0
    PB = VAL(24)
    BMASS = Y(10)
    IF ((TIME+DELT).LE.TPRINT) GO TO 1
    DEL = DELT
    DELT = TPRINT-TIME
    IPASS = 1
  1 X = TIME
C
    DO 2 J=1,NEQS
    Y(J) = YOLD(J)
  2 CONTINUE
C
```

```
      ITHRST = 1
      CALL RHS (K1)
      ITHRST = 2
      IMT = 0
      IF (IACCEL.NE.ON) GO TO 3
      ACCLAT = (K1(2)+Y(1)*Y(6))/G
      WRITE (6,20) ACCLAT,DELT
    3 ON = 2
    4 H = DELT/3.
      X = TIME+H
C
      DO 5 J=1,NEQS
    5 Y(J) = YOLD(J)+H*K1(J)
C
      CALL RHS (K2)
C
      DO 6 J=1,NEQS
    6 Y(J) = YOLD(J)+.5*H*(K1(J)+K2(J))
C
      CALL RHS (K3)
      X = TIME+.5*DELT
C
      DO 7 J=1,NEQS
    7 Y(J) = YOLD(J)+.375*H*(K1(J)+3.*K3(J))
C
      CALL RHS (K4)
      X = TIME+DELT
C
      DO 8 J=1,NEQS
    8 Y(J) = YOLD(J)+.5*H*(3.*K1(J)-9.*K3(J)+12.*K4(J))
C
      CALL RHS (K5)
      IF (JCC.EQ.1) GO TO 9
C
      DO 9 J=1,NEQS
      ERROR(J) = (K1(J)-4.5*K3(J)+4.*K4(J)-.5*K5(J))*H/5.0
      IF (ABS(ERROR(J)).GT.TOL(J)) GO TO 15
    9 CONTINUE
C
C
      DO 10 J=1,NEQS
      Y(J) = YOLD(J)+.5*H*(K1(J)+4.*K4(J)+K5(J))
   10 YOLD(J) = Y(J)
C
      TIME = TIME+DELT
      IF (IPASS.EQ.1) GO TO 14
      IF (JCC.EQ.1) GO TO 12
C
```

```
      DO 11  J=1,NEQS                                                    INT
      IF (ABS(ERROR(J)).GT.TOL(J)/16.) GO TO 13                          INT
   11 CONTINUE                                                           INT
C                                                                        INT
      DELT = 2.*DELT                                                     INT
      IF (DELT.GT.DELPNT) DELT=DELPNT                                    INT
   12 RETURN                                                             INT
   13 STEP2 = DELT                                                       INT
      GO TO 12                                                           INT
   14 DELT = DEL                                                         INT
      IPASS = 0                                                          INT
      GO TO 12                                                           INT
   15 DELT = DELT/2.                                                     INT
      IF (DELT.LT.1.E-6) GO TO 18                                        INT
      IF (JQQ.EQ.2) GO TO 17                                             INT
      WRITE (6,22) TIME,DELT,J,ERROR(J),TOL(J)                           INT
   16 IPASS = 0                                                          INT
      GO TO 4                                                            INT
   17 STEP1 = DELT*2.0                                                   INT
      IF (STEP1.LT.STEP2) STEP2=STEP1                                    INT
      GO TO 16                                                           INT
   18 WRITE (6,21)                                                       INT
      WRITE (6,19) TIME,DELT,(K1(J),J=1,NEQS),VAL                        INT
      STOP                                                               INT
C                                                                        INT
   19 FORMAT (/10X,23HINTGRL TIME,DELT,K1,VAL/2E15.4/2(6E15.4/),5(8E15.4INT
     1/))                                                                INT
   20 FORMAT (1H0,9X,33HTOTAL LATERAL ACCELERATION (G) = F12.4,12X,5HDT INT
     1= E15.4)                                                           INT
   21 FORMAT (1H1,10X,44HDELTA TIME LESS THAN 1.0E-6  -  -  JOB STOPS)   INT
   22 FORMAT (/10X,5HINT-J2E30.5,I5,2E20.5)                              INT
      END                                                                INT

      SUBROUTINE PROP                                                    PROP
C                                                                        PROP
      INTEGER ON                                                         PROP
      COMMON /CONST/ PI,RAD,UO                                           PROP
      COMMON /FPROP/ FX,FY,FZ,FK,FM,FN                                   PROP
      COMMON /ENGINE/ NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,TIPPROP
     1(25),TIS(25)                                                       PROP
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IPROP
     1RUD,IPROP,IAEROD,IRHS                                              PROP
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7   PROP
      COMMON /RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,RPROP
     1CLB,RTC,RUDANG,TIR(25)                                             PROP
      COMMON /VARBLE/ VAL(40)                                            PROP
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAPROP
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAPROP
```

```
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)
     3, (VAL(24),PB)
      DIMENSION THS(1), THP(1), TS(1), TP(1), RUD(1), TR(1)
      EQUIVALENCE (THSTS(1),THS(1)), (THSTP(1),THP(1)), (TIS(1),TS(1)),
     1(TIP(1),TP(1)), (TIR(1),TR(1)), (DELRUD(1),RUD(1))
C
      FX = 0.0
      FY = 0.0
      FZ = 0.0
      FK = 0.0
      FM = 0.0
      FN = 0.0
      TL = TIME
      IF (NPR.EQ.0.0) GO TO 1
      RUDANG = FG1(TL,NPR,TR,RUD,IR)
      RUDANG = RUDANG/RAD
C
C     CALCULATE THRUSTS AND MOMENTS INDIVIDUALLY
C
      GO TO 2
    1 RUDANG = DELRUD(1)
      RUDANG = RUDANG/RAD
    2 CD = COS(RUDANG)
      SD = SIN(RUDANG)
      IF (NPS.EQ.0.0) GO TO 3
      THSS = FG1(TL,NPS,TS,THS,IS)
      GO TO 4
    3 THSS = THSTS(1)
    4 IF (NPP.EQ.0.0) GO TO 5
      THSP = FG1(TL,NPP,TP,THP,IP)
      GO TO 6
    5 THSP = THSTP(1)
    6 STHSTS = STHS*THSS
      STHSTP = STHP*THSP
      FXS = THSS*CD-STHSTS*SD
      FXP = THSP*CD+STHSTP*SD
      FYS = -STHSTS*CD-THSS*SD
      FYP = STHSTP*CD+SD*THSP
      FZS = -THSS*THETA*CD+STHSTS*SD*PHI
      FZP = -THSP*THETA*CD-STHSTP*SD*PHI
      FX = FXP+FXS
      FY = FYP+FYS
      FZ = FZP+FZS
      FKP = -FZP*YP-FYP*ZP
      FKS = FZS*YP-FYS*ZP
      FK = FKS+FKP
      FMS = FZS*(-XP)+FXS*ZP
      FMP = FZP*(-XP)+FXP*ZP
```

```
      FM = FMS+FMP
      FNS = -FXS*YP-FYS*(-XP)
      FNP = FXP*YP-FYP*(-XP)
      FN = FNS+FNP
      IF (IPROP.NE.ON) RETURN
      WRITE (6,7) FX,FY,FZ,FK,FM,FN
      RETURN
C
    7 FORMAT (/10X,22HPPROP FX,FY,FZ,FK,FM,FN/6F15.4)
      END

      SUBROUTINE RUDDER
C
      INTEGER ON
      COMMON /CONST/ PI,RAD,U0
      COMMON /FRUD/ FX,FY,FZ,FK,FM,FN
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AM
     1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7
      COMMON /PRTINT/ ON,IACCFL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,I
     1RUD,IPROP,IAEROD,IRHS
      COMMON /RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,R
     1CLB,RTC,RUDANG,TIR(25)
      COMMON /VARBLE/ VAL(40)
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VA
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VA
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)
     3, (VAL(24),PB)
      EQUIVALENCE (DELRUD(1),RUD(1)), (TIR(1),TR(1))
      DIMENSION RUD(1), TR(1)
      EQUIVALENCE (VAL(18),FANPWR)
      DATA ENU/1.28E-5/
C
C     CALCULATE PROGRAMMED RUDDER DEFLECTION
C
      TL = TIME
      IF (NPR.EQ.0.0) GO TO 1
      GO TO 2
    1 RUDANG = DELRUD(1)
      RUDANG = RUDANG/RAD
      GO TO 3
    2 RUDANG = FG1(TL,NPR,TR,RUD,IR)
      RUDANG = RUDANG/RAD
C
C     SIDE FORCE ON RUDDER
C
    3 DSR = Z+ZS-XR*THETA
      ENDFAC = (1.+DSR/(DSR+RSPAN))
```

```
      VH = V+XR*R-ZR*P
      QQ = HRHO*U*U*RAREA
      EFFANG = RUDANG-ENDFAC*VH/U
      FY = 2.*QQ*ENDFAC*RCLB*EFFANG
C
C     DRAG FORCE ON RUDDER
C
      REY = U*(RAREA/RSPAN)/ENU
      CFR = .427/(ALOG10(REY)-.407)**2.64
      PI8 = PI/8.
      CD = 2.*CFR+PI8*RTC*RTC*(1.+G*RSPAN/(U*U))+RCLB*EFFANG*EFFANG
      FX = -2.*CD*RAREA*HRHO*U*U
      FZ = 0.
      FK = -ZR*FY
      FM = FX*ZR
      FN = XR*FY
      IF (IRUD.NE.ON) RETURN
      WRITE (6,4) FX,FY,FZ,FK,FM,FN
C
C
      RETURN
C
    4 FORMAT (/10X,24HRUDDER FX,FY,FZ,FK,FM,FN/6E15.4)
      END

      SUBROUTINE RHS (VALUF)
      INTEGER ON
      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON /ATRIX/ AMASSI,AIYYI,DIXX,DIXZ,DIZZ
      COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /COLUMN/ IVERT,ILATRL
      COMMON /CONST/ PI,RAD,UO
      COMMON /CNTRL/ CONTW,CONTQ,CONTTH,QMULT,LOUVER,ACONTZ,ACONTW,ZEQUI
     1L,THEQL,ACBASE
      COMMON /ENGINE/ NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,TIP
     1(25),TIS(25)
      COMMON /FANMAP/ QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,EN
     1SFAN,BRPM,MRPM,SRPM,NPTSB,NPTSM,NPTSS,PBFAN(25),PMFAN(25),PSFAN(25
     2),TMEB(25),DELB(25),NB,TMES(25),DETS(25),NS
      COMMON /FAERO/ FXAED,FYAED,FZAED,FKAED,FMAED,FNAED
      COMMON /FORBS/ FXBS,FYBS,FZBS,FKBS,FMBS,FNBS,QLBS
      COMMON /FORSS/ FXSS,FYSS,FZSS,FKSS,FMSS,FNSS,QLSS,FMS
      COMMON /FPROP/ FXP,FYP,FZP,FKP,FMP,FNP
      COMMON /FROUDE/ FN,FNCRIT
      COMMON /FRUD/ FXRUD,FYRUD,FZRUD,FKRUD,FMRUD,FNRUD
      COMMON /GBOW/ XBOW
      COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,
     110),XCP,ZCP
```

```
      COMMON /GEOMBS/ DETABX(11),DETABT(11),ARM1B(10),ARM2B(10),DFBS(10)RHS
     1,TSKIE(10)                                                        RHS
      COMMON /GEOMSS/ DETADX(11),DETADT(11),ARM1S(10),DFSS(10),TSKIS(10)RHS
     1,ARM2S(10)                                                        RHS
      COMMON /KSWTCH/ ITHRST                                            RHS
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AMRHS
     1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO                         RHS
      COMMON /MSIDW/ DF(2,10),DSWAV(2,10),FXH(2),FYH(2),FZH(2),FMH(2),FNRHS
     1H(2),VFY(2),VFZ(2),FXV                                            RHS
      COMMON /MWAVE/ FXW(2),FYW(2),FZW(2),FKW(2),FMW(2),FNW(2)          RHS
      COMMON /OPTION/ I3DOF,ISRGE,ITRIM,IDIA                            RHS
      COMMON /PLENUM/ XLBW,XBBW,ABW,BUBHGT,DVWDOT,VMDOT,CMM             RHS
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT                     RHS
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IRHS
     1RUD,IPROP,IAEROD,IRHS                                             RHS
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7  RHS
      COMMON /PWAVE/ FNCON,PWVCON                                       RHS
      COMMON /RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,RRHS
     1CLB,RTC,RUDANG,TIR(25)                                            RHS
      COMMON /SIDE/ FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSRHS
     1W,VAREA,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ,AVBMSW,DELX,VTC         RHS
      COMMON /SLOPE/ WATSLP,XPWV,XLXPWV,PWVHT,XPWVXS                    RHS
      COMMON /SOFTBS/ XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAVRHS
     1GB(10),CENCAB                                                     RHS
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS,ELMAXS,YAVRHS
     1GS(10)                                                            RHS
      COMMON /VALOLD/ YOLD(20)                                          RHS
      COMMON /VARBLE/ VAL(40)                                           RHS
      COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FYRHS
     1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBRHS
     2BAR                                                               RHS
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VARHS
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VARHS
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)RHS
     3, (VAL(24),PB)                                                    RHS
      EQUIVALENCE (VAL(18),FANPWR)                                      RHS
      DIMENSION ACCEL(3), ANGACL(3)                                     RHS
      DIMENSION GF(6), VALUE(20)                                        RHS
      DATA CCF/4.9E-07/                                                 RHS
C                                                                       RHS
C                                                                       RHS
      DO 1 J=1,20                                                       RHS
    1 VALUE(J) = 0.0                                                    RHS
C                                                                       RHS
C                                                                       RHS
C     CALCULATION OF BUBBLE WAVE MAKING DRAG                            RHS
C                                                                       RHS
      AB = ABW-(ABW-(XL*WIDTH))*(ZS+Z)/BUBHGT                           RHS
```

```
      FN = U/FNCON
      CF = .37/(FN**1.5655981)
      FXPWAV = -PWVCON*PBBAR*CF
      WATSLP = -FXPWAV/WEIGHT
      VOL = VAL(12)
      PB = VAL(13)
      PBAR = VAL(13)-PINF
      PBS = PB+DPBS
      PSS = PB+DPSS
      ABPB = PBAR*AB
C     CALCULATION OF BUBBLE WAVE MAKING DRAG
C
      FLOW = SQRT(2.*ABS(PBAR)/RHOINF)*SIGN(1.,PBAR)
      CLSW = CFSW*ALSW*FLOW
C
      CALL BOWSL
      CALL STNSL
C
      SUMX = FXBS+FXSS+FXSW+FXRUD+FXP+FXWAV+FXAED+FXPWAV
      IF (ITHRST.NE.ITRIM) GO TO 2
      THSTS(1) = THSTS(1)-SUMX*.5
      THSTP(1) = THSTP(1)-SUMX*.5
      THST = THSTS(1)+THSTP(1)
      SUMX = 0.0
    2 CONTINUE
      SUMY = FYBS+FYSS+FYSW+FYRUD+FYP+FYWAV+FYAED
      SUMZ = WEIGHT-ABPB+FZBS+FZSS+FZSW+FZRUD+FZP+FZWAV+FZAED
      SUMK = FKBS+FKSS+FKSW+FKRUD+FKP+FKWAV+FKAED+ABPB*PHI*(-Z)
C
C     CALCULATION OF EFFECTIVE CENTER OF PRESSURE
C
      UU = U*0.5921-30.0
      XCPU = XCP+0.001975*UU*UU-0.974
      XCPC = SHXYAX(XCPU,ZCP,THETA,PI)
      FMBUB = ABPB*XCPC
      SUMM = FMBS+FMSS+FMSW+FMRUD+FMP+FMWAV+FMAED+FMBUB+FXPWAV*ZS
      FWAVZ = FXPWAV*ZS
      SUMN = FNBS+FNSS+FNSW+FNRUD+FNP+FNWAV+FNAED
      IF (I3DOF.NE.1) GO TO 3
      SUMZ = 0.0
      SUMM = 0.0
    3 CONTINUE
      VALUE(1) = SUMX*AMASSI
      VALUE(2) = SUMY*AMASSI-R*U
      VALUE(3) = SUMZ*AMASSI
      VALUE(4) = SUMK*DIZZ+SUMN*DIXZ
      VALUE(5) = SUMM*AIYYI
      VALUE(6) = SUMN*DIXX+SUMK*DIXZ
```

```
      VALUE(7) = P
      VALUE(8) = Q
      VALUE(9) = W
      IF (I3DOF.EQ.1) GO TO 6
C
C     BUBBLE PRESSURE EQUATION
C
      QOUT = QLBS+QLSS+QLSW
C
      CALL FAN
      VALUE(10) = RHOINF*(QIN-QOUT)
      VALUE(11) = -(((XL*WIDTH)+ABW)*.5)*W-DVWDOT
      VALUE(12) = VAL(13)*GAM*((VALUE(10)/VAL(11))-(VALUE(11)/VAL(12)))
      IF (IDIA.EQ.1) GO TO 4
      GO TO 5
    4 CONTINUE
C
C   MEMBRANE STUDY
C
      VAL(12)=VAL(12)+0.3175*VAL(13)
      VALUE(12)=VAL(13)*GAM*((VALUE(10)/VAL(11))-(VALUE(11)/VAL(12)))
    5 CONTINUE
      GO TO 7
    6 CONTINUE
      VALUE(10) = 0.0
    7 CONTINUE
C
C     WRITE DATA FILE FOR MOMENT AND SHEAR CALCS., IF REQUIRED
C
      IF (IMT.NE.1) GO TO 8
      NBS = NSTA(3)-1
      NSS = NSTA(4)-1
      NSSL = NSS/2+1
      WRITE (IBMFIL) (VAL(I),I=1,24),ZBAR,PHIBAR,THEBAR,FXW,FYW,FZW,FKW,
     1FMW,FNW,(VALUE(I),I=1,10),DF,DSWAV,FXH,FYH,FZH,FMH,FNH,VFY,VFZ,FXV
     2,FXRUD,FYRUD,FXP,FYP,FZP,FZSS,FKSS,FMSS,FXBS,FZBS,FXAED,FYAED,FZAE
     3D,FMAED,FNAED,FXPWAV,FXSS,FKBS,FMBS,FNBS,FNSS,(TSKIS(I),DFSS(I),I=
     4NSSL,NSS),(TSKIB(I),DFBS(I),ARM1B(I),ARM2B(I),I=1,NBS)
    8 CONTINUE
C
C     CONSTANT LONGITUDINAL VELOCITY ( U )
C
      IF (ISRGE.EQ.1) VALUE(1)=0.0
      IF (ON.NE.1) RETURN
C
      DO 9 I=1,3
      ACCEL(I) = VALUE(I)/G
      ANGACL(I) = VALUE(I+3)*RAD
```

```
    9 CONTINUE
C
      BOWACC = ACCEL(3)-XBOW*VALUE(5)/G
      STNACC = ACCEL(3)+XS*VALUE(5)/G
      IF (IVERT.NE.ON) GO TO 10
      ZD = Z+ZS
      VOLP = VOL
      THETAR = THETA*RAD
      QDEG = Q*RAD
   10 IF (ILATRL.NE.ON) GO TO 13
      DEPSI = PSI*RAD
      PDEG = P*RAD
      RDEG = R*RAD
      BETAS = -V/U*RAD
      ACCLAT = (VALUE(2)+U*R)/G
      DPHI = PHI*RAD
      DRFT = 12.0*ZD
      VEL = 0.5925*U
      DELRS = RUDANG*RAD
      IF (R.EQ.0.0) GO TO 11
      TRADUS = U/R
      GO TO 12
   11 TRADUS = 1.E8
   12 WRITE (1) TIME,VAL(16),DRFT,THETAR,PBAR,BOWACC,ACCEL(3),FANPWR,DPH
     1I,DEPSI,ACCLAT,VEL,TRADUS,VOLP,X,Y,QIN,QOUT,GF(1),FXPWAV,THSTS(1),
     2THSTP(1),QDEG,PDEG,RDEG,DELRS
   13 IF (IRHS.NE.ON) RETURN
      WRITE (6,14) FMBS,FMSS,FMSW,FMRUD,FMP,FMWAV,FMAED,FMBUB,FWAVZ
      WRITE (6,15) FXPWAV
      WRITE (6,16) PBAR,FANPWR,QIN,QLBS,QLSW,QLSS
      WRITE (6,17) AB,VOL
      WRITE (6,18) VALUE,VAL
      WRITE (6,19) GF,ACCEL,ANGACL
      WRITE (6,20) BOWACC,STNACC
C
C
      RETURN
C
   14 FORMAT ('0',6X,5HFMBS=,E16.6,2X,5HFMSS=,F16.6,2X,5HFMSW=,
     1 E16.6,/,'0',6X,6HFMRUD=,E16.6,2X,4HFMP=,E16.6,2X,6HFMWAV=,E16.6,
     2 /,'0',6X,6HFMAED=,E16.6,2X,6HFMBUB=,E16.6,2X,6HFWAVZ=,E16.6)
   15 FORMAT ('0',6X,7HFXPWAV=,E16.6)
   16 FORMAT (//10X,3HPHS20HGAGEPRESS.(PSF)=F7.2,5X,21HFAN POWER REQD (H
     1P) =F8.2,5X,27HFAN FLOW RATE (CU FT/SEC) =F9.2,//31H LEAKAGE FLOW
     2RATES (CU FT/SEC)//11H BOW SEAL =F9.2,11H SIDEWALL =F9.2,13H STERN
     3 SEAL =F9.2)
   17 FORMAT (/13H PLENUM AREA=F9.2,10X,14HPLENUM VOLUME=F10.2)
   18 FORMAT (//12H VALUE ARRAY2(/10E13.4)/10H VAL ARRAY4(/10E13.4))
```

```
19 FORMAT (/10X,24HTOTAL FORCES AND MOMENTS6E12.4/10X,24HACCELERATIONRHS
  1S G,DEG/SEC26E12.4)                                                RHS
20 FORMAT (/10X,16HBOW ACCEL. (G) =F12.4,21H   STERN ACCEL. (G) =E12.RHS
  14)                                                                 RHS
   END                                                                RHS

   SUBROUTINE SAM                                                     SAM
C                                                                     SAM
   WRITE (6,1)                                                        SAM
   RETURN                                                             SAM
C                                                                     SAM
 1 FORMAT (1H1,'YOU HAVE CALLED A DUMMY SAM SUBROUTINE.'/             SAM
  110X,'CHANGE TO BHISES TO USE THE SAM SUBROUTINE.')                 SAM
   END                                                                SAM

   SUBROUTINE SIDEWL                                                  SDWL
C                                                                     SDWL
   INTEGER ON                                                         SDWL
   COMMON /AIR/ PINF,RHOINF,GAM                                       SDWL
   COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY SDWL
   CCMMON /CONST/ PI,RAD,UO                                           SDWL
   COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,SDWL
  110),XCP,ZCP                                                        SDWL
   COMMON /GEOMSW/ XAVG(10),DS                                        SDWL
   COMMON /KSWTCH/ ITHRST                                             SDWL
   COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AMSDWL
  1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO                          SDWL
   COMMON /MSIDW/ DF(2,10),DSWAV(2,10),FXH(2),FYH(2),FZH(2),FMH(2),FNSDWL
  1H(2),VFY(2),VFZ(2),FXV                                             SDWL
   COMMON /PLENUM/ XLBW,XBBW,ABW,BUBHGT,DVWDOT                        SDWL
   COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT                      SDWL
   COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,ISDWL
  1RUD,IPROP,IAEROD,IRHS                                              SDWL
   COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7   SDWL
   COMMON /SIDE/ FX,FY,FZ,FK,FM,FN,ALSW,YSW,XLSW,CFSW,CDSW,VAREA,VCHOSDWL
  1RD,VSPAN,VANGLE,VCOS,VX,VY,VZ,AVBMSW,DELX,VTC                      SDWL
   COMMON /SLOPE/ WATSLP,XPWV,XLXPWV,PWVHT,XPWVXS                     SDWL
   CCMMON /VARBLE/ VAL(40)                                            SDWL
   COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FYSDWL
  1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBSDWL
  2BAR                                                                SDWL
   COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB,ACSDWL
  11(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),AC5(20,5,7),AC6(20,5SDWL
  2,7),AC7(20,5,7),AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),AS1(20,5,7),ASDWL
  3S2(20,5,7),AS3(20,5,7),AS4(20,5,7),AS5(20,5,7),AS6(20,5,7),AS7(20,SDWL
  45,7),AS0(20,5,7),AS00(20,5,7),AS8(20,5,7),BB(36),XREF,RX           SDWL
   EQUIVALENCE (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAL(5),P), (VAL(6SDWL
  1),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAL(10),Z), (VAL(SDWL
```

```
 211),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI), (VAL(24),PB)
      DIMENSION GAP(2,11), DSW(2,11)
      DIMENSION FZHOLD(2), FZHDRP(2)
      DATA ENU/1.28E-5/
C
      PBAR = PB-PINF
      PBHEAD = PBAR/(RHO*G)
C
C     GAF OR WETTED DRAFT CALCULATION
C
C
      DO 5 J=1,2
      N = NSTA(J)
C
      DO 5 K=1,N
      DD = ZS+Z+YY(J,K)*PHI-XX(1,K)*THETA+ETA(J,K)
      DDIN = DD-WATSLP*(XPWVXS-XX(J,K))
      IF (DDIN.LT.BUBHGT) GO TO 1
      IF (VAL(1)-TOLD.LT.DELPNT) GO TO 1
      TOLD = VAL(1)
      WRITE (6,14) XX(J,K),VAL(1),DD
    1 CONTINUE
      DSW(J,K) = (SIGN(1.,DD)+1.)*DD/2.
      IF (DDIN) 2,4,4
    2 IF (DSW(J,K)-PBHEAD) 3,4,4
    3 GAP(J,K) = -DDIN*(1.-(DSW(J,K))/PBHEAD)
      GO TO 5
    4 GAP(J,K) = 0.0
    5 CONTINUE
C
C
C     LEAKAGE AREA
C
      ALSW = 0.0
C
      DO 6 J=1,2
      N = NSTA(J)-1
C
      DO 6 I=1,N
      ALSW = ALSW+(GAP(J,I)+GAP(J,I+1))*DELX/2.
    6 CONTINUE
C
C
C     CROSS-FLOW DRAG ON SIDEWALLS
C
      FYD = 0.0
      FKD = 0.0
      FND = 0.0
```

```
C
      DO 7 I=1,2
      N = NSTA(I)-1
C
      DO 7 J=1,N
      DSWAV(I,J) = (DSW(I,J)+DSW(I,J+1))/2.
      VREL = V+XAVG(J)*R-(ZS-DSWAV(I,J)/2.)*P
      DF(I,J) = -HRHO*CDSW*VREL*ABS(VREL)*DELX*DSWAV(I,J)
      FYD = FYD+DF(I,J)
      FND = FND+DF(I,J)*XAVG(J)
    7 FKD = FKD-(ZS-DSWAV(I,J)/2.)*DF(I,J)
C
C
C     SET UP STERN LIMIT OF FORCE DETERMINATION
C
      XSS = -XS
      GO TO 8
      ENTRY SIDWLM
      XSS = XMI(IX)
    8 IP = 1.+(THETA*RAD-STH)/DTH
      IP = MAXO(MINO(IP,NTH),1)
      IP1 = MINO(IP+1,NTH)
      DTHETA = (IP-1)*DTH+STH
      DIP = (THETA*RAD-DTHETA)/DTH
C
C     CALC REYNOLDS NO. AND DRAG COEFF.
C
      REY = U*XLSW/ENU
      CDT = .427/(ALOG10(REY)-.407)**2.64
C
C     SIDEWALL FORCES, P/S
C
C
      DO 11 J=1,2
      WAREA = 0.0
      N = NSTA(J)-1
      NI = (XSS+XS)*N/XLSW+1.5
C
      DO 9 I=NI,N
      ZOR1 = 1.
      IF (DSWAV(J,I).EQ.0.0) ZOR1=0.0
    9 WAREA = WAREA+DELX*(2.*DSWAV(J,I)+ZOR1*AVBMSW)
C
      FXH(J) = -HRHO*CDT*WAREA*U*U
      PM1 = 2*J-3
      YLSW = PM1*YSW
      DS = Z+ZS+YLSW*PHI
      DSS = DS-XSS*THETA
```

```
      ZORI  =  (SIGN(1.,DSS)+1.)*0.5
      DSS  =  DSS*ZORI
      IDSS  =  1.5+(DSS-SBB)/DBB
      IDSS  =  MINO(NBB,IDSS)
      BS  =  BB(IDSS)
      ZORI  =  (SIGN(1.,DSS)+1.)*0.5
      DSS  =  DSS*ZORI
      DRBOW  =  DSS-(XX(J,N+1)-XSS)*THETA
      IF (DRBOW.LT.0.0) DRBOW=0.0
      A33S  =  (RHO*PI*BS**2)*0.125
      A22S  =  (RHO*.4*PI*DSS**2)*0.5
      IF (THETA.LT.0.0) A22S=.4*RHO*PI*DRBOW*DRBOW*0.5
      DSR  =  DS-(XREF-XS)*THETA
      ID  =  1.+(DSR*12.-SDS)/DDS
      ID  =  MAXO(MINO(ID,NDS),1)
      DDSR  =  (ID-1)*DDS+SDS
      ID1  =  MINO(ID+1,NDS)
      DID  =  (DSR*12.-DDSR)/DDS
      BCO  =  ACO(1,ID,IP)
      BCOO  =  ACOO(1,ID,IP)
      BC2  =  AC2(1,ID,IP)
      BC5  =  AC5(1,ID,IP)
      BC6  =  AC6(1,ID,IP)
      BCO  =  BCO+DID*(ACO(1,ID1,IP)-BCO)+DIP*(ACO(1,ID,IP1)-BCO+DID*(ACO(
     11,ID1,IP1)-ACO(1,ID,IP1)-ACO(1,ID1,IP)+BCO))
      BCOO  =  BCOO+DID*(ACOO(1,ID1,IP)-BCOO)+DIP*(ACOO(1,ID,IP1)-BCOO+DID
     1*(ACOO(1,ID1,IP1)-ACOO(1,ID,IP1)-ACOO(1,ID1,IP)+BCOO))
      BC2  =  BC2+DID*(AC2(1,ID1,IP)-BC2)+DIP*(AC2(1,ID,IP1)-BC2+DID*(AC2(
     11,ID1,IP1)-AC2(1,ID,IP1)-AC2(1,ID1,IP)+BC2))
      BC5  =  BC5+DID*(AC5(1,ID1,IP)-BC5)+DIP*(AC5(1,ID,IP1)-BC5+DID*(AC5(
     11,ID1,IP1)-AC5(1,ID,IP1)-AC5(1,ID1,IP)+BC5))
      BC6  =  BC6+DID*(AC6(1,ID1,IP)-BC6)+DIP*(AC6(1,ID,IP1)-BC6+DID*(AC6(
     11,ID1,IP1)-AC6(1,ID,IP1)-AC6(1,ID1,IP)+BC6))
C
C     SHIFT MOMENT CENTER FRCM XREF TO C.G.
C
      BCOO  =  BCOO-(XS-XREF)*BCO
      BC6  =  BC6-(XS-XREF)*BC5
C
C     HYDROSTATIC AND HYDRODYNAMIC FORCES
C
      FZH(J)  =  -G*BCO-U*U*A33S*THETA-U*A33S*W+Q*U*(-BC2+A33S*XSS)-U*A33S
     1*P*YLSW
      FMH(J)  =  -U*XSS*XSS*A33S*Q+G*BCOO+U*(A33S*XSS+BC2)*(W+U*THETA+YLSW
     1*P)
      FYH(J)  =  -A22S*U*(V+XSS*R-ZS*P)
      FNH(J)  =  FYH(J)*XSS-U*((V-ZS*P)*BC5+R*BC6)
C
```

```
C     ADD VERTICAL FORCE DUE TO DEADRISE PROJECTION OF LATERAL FORCE      SDWL1780
C                                                                         SDWL1790
      CTNDR = 0.0                                                         SDWL1800
      IF (DSS.LE.0.0) GO TO 10                                            SDWL1810
      CTNDR = (BS-BB(1))/DSS                                              SDWL1820
      IF (THETA.LT.0.0) CTNDR=.39391                                      SDWL1830
   10 CONTINUE                                                            SDWL1840
      FZHOLD(J) = FZH(J)                                                  SDWL1850
      FZHDRP(J) = PM1*FYH(J)*CTNDR*PROMO1                                 SDWL1860
      FZH(J) = FZH(J)+FZHDPP(J)                                           SDWL1870
      IF (IMT.EQ.2) GO TO 11                                              SDWL1880
C                                                                         SDWL1890
C     CALC OF FORCE ON VENTRAL FINS REMOVED                               SDWL1900
C                                                                         SDWL1910
   11 CONTINUE                                                            SDWL1920
C                                                                         SDWL1930
      IF (IMT.EQ.2) GO TO 12                                              SDWL1940
C                                                                         SDWL1950
C     TOTAL SIDEWALL FORCES AMD MOMENTS                                   SDWL1960
C                                                                         SDWL1970
      FX = FXH(1)+FXH(2)                                                  SDWL1980
      FY = FYH(1)+FYH(2)                                                  SDWL1990
      FZ = FZH(1)+FZH(2)                                                  SDWL2000
      FK = (FZH(2)-FZH(1))*YSW+FKD-FY*ZS                                  SDWL2010
      FY = FY+FYD                                                         SDWL2020
      FM = FMH(1)+FMH(2)+ZS*FX                                            SDWL2030
      FN = FND+FNH(1)+FNH(2)+(FXH(1)-FXH(2))*YSW                          SDWL2040
C                                                                         SDWL2050
C     DRAG FORCE ON FINS REMOVED                                          SDWL2060
C                                                                         SDWL2070
   12 CONTINUE                                                            SDWL2080
C                                                                         SDWL2090
C     ADD ROLL DAMPING DUE TO VERTICAL WAVE GENERATION                     SDWL2100
C                                                                         SDWL2110
      DSS = Z+ZS-XSS*THETA                                                SDWL2120
      ZOR1 = (SIGN(1.,DSS)+1.)/2.                                         SDWL2130
      DSS = DSS*ZOR1                                                      SDWL2140
      DS = Z+ZS                                                           SDWL2150
      DSR = DS-(XREF-XS)*THETA                                            SDWL2160
      ID = 1.+(DSR*12.-SDS)/DDS                                           SDWL2170
      ID = MAXO(MINO(ID,NDS),1)                                           SDWL2180
      DDSR = (ID-1)*DDS+SDS                                               SDWL2190
      ID1 = MINO(ID+1,NDS)                                                SDWL2200
      DID = (DSR*12.-DDSR)/DDS                                            SDWL2210
      BC2 = AC2(1,ID,IP)                                                  SDWL2220
      BC2 = BC2+DID*(AC2(1,ID1,IP)-BC2)+DIP*(AC2(1,ID,IP1)-BC2+DID*(AC2(SDWL2230
     11,ID1,IP1)-AC2(1,ID,IP1)-AC2(1,ID1,IP)+BC2))                        SDWL2240
      FKCLD = FK                                                          SDWL2250
```

```
      FK = FK-PROMO2*YSW*YSW*BC2*P/PI
      FZH(1) = FZH(1)+PROMO2/2.*YSW*BC2*P/PI
      FZH(2) = FZH(2)-PROMO2/2.*YSW*BC2*P/PI
      IF (PROMO3.EQ.1.0) WRITE (6,15) VAL(1),FZHOLD(1),FZHOLD(2),FZHDRP(
     11),FZHDRP(2),FZH(1),FZH(2),FKOLD,FK
      IF (ISIDWL.NE.ON) RETURN
C
      DO 13 I=1,2
C
      DO 13 J=1,11
      GAP(I,J) = 12.0*GAP(I,J)
   13 DSW(I,J) = 12.0*DSW(I,J)
C
      WRITE (6,16) ((GAP(I,J),J=1,11),I=1,2),((DSW(I,J),J=1,11),I=1,2),F
     1X,FY,FZ,FK,FM,FN
C
C
      RETURN
C
   14 FORMAT (/10X,43HWATER CONTACT WITH TOP OF BUBBLE CHAMBER ATF7.2,14
     1H FT.     TIME =F7.2,19H SEC.      IMMERSION=F7.2,4H FT.)
   15 FORMAT (/2X,'TIME=',1X,E15.4,2X,'OLD VERTICAL FORCES',2(5X,E15.4)/
     1 25X,'VERTICAL DEADRISE FORCES',2(5X,E15.4)/25X,'NEW VERTICAL FORC
     2ES',2(5X,E15.4)/25X,'OLD AND NEW ROLL MOMENTS',2(5X,E15.4)/)
   16 FORMAT (/10X,8HSIDEWALL/25H GAP (FT.) (STERN TO BOW)/14H PORT SIDE
     1WALL/11F10.5/14H STBD SIDEWALL/11F10.5/37H IMMERSION DEPTH (FT.) (
     2STERN TO BOW)/14H PORT SIDEWALL/11F10.5/14H STBD SIDEWALL/11F10.5/
     310X,26HSIDEWALL FX,FY,FZ,FK,FM,FN/6E15.4)
      END

      FUNCTION SHXYAX (X,Z,ANGYAX,PI)
C
      H = SQRT(X*X+Z*Z)
      IF (X.EQ.0.0) GO TO 1
      ARG = Z/X
      ANGOLD = ATAN(ARG)
      IF (ANGOLD.GE.0.0) GO TO 2
      ANGNEW = ANGOLD+PI-ANGYAX
      GO TO 3
    1 ANGNEW = PI*.50-ANGYAX
      GO TO 3
    2 ANGNEW = ANGOLD-ANGYAX
    3 SHXYAX = H*COS(ANGNEW)
C
      RETURN
      END

      SUBROUTINE STNSL
```

```
C                                                                       SSL
      INTEGER ON                                                        SSL
      COMMON /AIR/ PINF,RHOINF,GAM                                      SSL
      COMMON /CONST/ PI,RAD,UO                                          SSL
      COMMON /FORSS/ FX,FY,FZ,FK,FM,FN,QL,FMS                           SSL
      COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,SSL
     110),XCF,ZCP                                                       SSL
      COMMON /GEOMSS/ DETADX(11),DETADT(11),ARMIS(10),DFSS(10),TSKIS(10)SSL
     1,ARM2S(10)                                                        SSL
      COMMON /KSWTCH/ ITHRST                                            SSL
      COMMON /LEAKER/ ALEAK,CFSS                                        SSL
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AMSSL
     1I(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO                         SSL
      COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,ISSL
     1RUD,IPROP,IAEROD,IRHS                                             SSL
      COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7  SSL
      COMMON /SLCPE/ WATSLP,XPWV,XLXPWV,PWVHT,XPWVXS                    SSL
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS,ELMAXS,YAVSSL
     1GS(10)                                                            SSL
      COMMON /STSLR/ CPHI,CPHID                                         SSL
      COMMON /VALOLD/ YOLD(20)                                          SSL
      COMMON /VARBLE/ VAL(40)                                           SSL
      COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FYSSL
     1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBSSL
     2BAR                                                               SSL
      EQUIVALENCE (VAL(1),TIME), (VAL(2),U), (VAL(3),V), (VAL(4),W), (VASSL
     1L(5),P), (VAL(6),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VASSL
     2L(10),Z), (VAL(11),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI)SSL
     3, (VAL(24),PB)                                                    SSL
      DIMENSION GAP(11), ELSKI(11), ELSKIL(11), AIRLEN(11), CTNSL(6,24) SSL
      DATA ENU,GPS,HINGHT/1.28E-5,.4644,1.875/                          SSL
      DATA CTNSL/0.0,3.8,7.0,9.9,12.3,15.0,0.0,4.1,7.3,10.5,13.2,15.7,0.SSL
     10,4.3,7.7,11.0,13.7,16.7,0.0,4.7,8.2,11.5,14.5,17.4,0.0,5.2,8.9,12SSL
     2.4,15.4,18.3,0.0,5.5,9.4,13.1,16.1,19.2,0.0,5.8,9.9,13.8,17.1,20.6SSL
     3,0.0,6.1,10.6,14.7,18.4,21.9,0.0,6.4,11.0,15.3,19.3,23.3,0.0,6.6,1SSL
     41.7,16.4,20.5,24.5,0.0,6.9,12.3,17.6,22.1,25.7,0.0,7.2,13.5,19.6,2SSL
     54.1,28.3,0.0,8.2,15.3,22.1,26.3,30.8,0.0,9.6,17.5,24.2,29.7,34.6,0SSL
     6.0,11.9,20.0,25.4,32.3,38.0,0.0,14.0,23.3,30.0,35.8,41.0,0.0,17.5,SSL
     725.5,32.5,39.7,44.5,0.0,21.6,30.0,38.7,42.0,47.0,0.0,24.6,33.7,41.SSL
     80,45.0,48.7,0.0,28.3,38.5,45.0,48.7,48.7,0.0,35.0,45.0,48.7,48.7,4SSL
     98.7,0.0,40.0,48.7,48.7,48.7,48.7,0.0,48.7,48.7,48.7,48.7,48.7,0.0,SSL
     $48.7,48.7,48.7,48.7,48.7/                                         SSL
C                                                                       SSL
      IF (ELMAXS.GT.HINGHT) ELMAXS=HINGHT                               SSL
C                                                                       SSL
      DO 1 J=1,11                                                       SSL
      GAP(J) = 0.0                                                      SSL
      ELSKI(J) = 0.0                                                    SSL
```

```
      ELSKIL(J) = 0.0
      AIRLEN(J) = 0.0
    1 CONTINUE
C
C
C     'EFDEP' IS THE EFFECTIVE LIFTING DEPTH OF THE STERN SEAL
C
      EFDEP = 6.0
      MM = EFDEP
      ALSS = 0.0
      FX = 0.0
      FZ = 0.0
      FK = 0.0
      FM = 0.0
      FN = 0.0
      AGAP1 = 0.0
      AGAP2 = 0.0
      AGAPA1 = 0.0
      DELP = PSS-PB
      IF (DELP.LT.0.0) DELP=0.0
      PBAR = PB-PINF
C
C     CALCULATE ELSKI HERE.
C
      SINDIF = SINTH-COSTH*THETA
      COSDIF = COSTH+SINTH*THETA
      X1 = XSS+ZSS*THETA-XLF*SINDIF
      Z1 = (-Z-ZSS+XSS*THETA-ELMAXS*COS(THETA))
C
C     CALCULATE GAP HERE.
C
      N = NSTA(4)
C
      DO 2 K=1,N
      ELSKI(K) = (ETA(4,K)-DETADX(K)*(XX(4,K)-X1)-Z1)+YY(4,K)*PHI-XPWV*W
     1ATSLP
      IF (ELSKI(K).GT.HINGHT) ELSKI(K)=HINGHT
      ELSKIL(K) = ELSKI(K)+GPS
      IF (ELSKIL(K).GT.HINGHT) ELSKIL(K)=HINGHT
      IF (ELSKIL(K).LT.(HINGHT-ELMAXS)) ELSKIL(K)=HINGHT-ELMAXS
      GAP(K) = -ELSKI(K)+(HINGHT-ELMAXS)
      IF (GAP(K).LT.0.0) GAP(K)=0.0
      MM1 = ELSKIL(K)*12.0
      MM2 = MM1+1
      MM3 = MM2+1
      DLINC = ELSKIL(K)*12.0-MM1
      STNSL1 = CTNSL(MM,MM2)
      STNSL2 = CTNSL(MM,MM3)
```

```
      AIRLEN(K) = ((STNSL2-STNSL1)*DLINC+STNSL1)/12.0
    2 CCNTINUE
C
      N = NSTA(4)-1
C
      DO 5 J=1,N
      ELSKIA = (ELSKI(J+1)+ELSKI(J))*0.5
      ELSKLA = (ELSKIL(J+1)+ELSKIL(J))*0.5
      AIRLAV = (AIRLEN(J+1)+AIRLEN(J))*0.5
      AGAP = ELSKLA-ELSKIA
      AGAP1 = AGAP
      IF (AGAP.LT.GPS) AGAP=GPS
      IF (AGAP1.GT.GPS) AGAP1=GPS
      ARM1S(J) = XX(4,J)+ELSKIA*0.5
      ARM2S(J) = ZS-ELSKIA
      DFSS(J) = -DELP*DELYSS*AIRLAV/(GPS/AGAP)**2
      IF (AIRLAV.LE.0.0) GO TO 3
      ARG = .5*RHO*U*U*AIRLAV*DELYSS
      RESKI = U*AIRLAV/ENU
      CDTSKI = .427/(ALOG10(RESKI)-.407)**2.64
      TSKIS(J) = -ARG*CDTSKI
C
C    THE FOLLOWING CARD REMOVES WATER DRAG EFFECTS OF STERN SEAL
C
      TSKIS(J) = 0.0
      GO TO 4
    3 TSKIS(J) = 0.0
    4 CONTINUE
      FX = FX+TSKIS(J)
      FZ = FZ+DFSS(J)
      FK = FK+DFSS(J)*YAVGS(J)
      FM = FM-DFSS(J)*ARM1S(J)+TSKIS(J)*ARM2S(J)
      FN = FN-TSKIS(J)*YAVGS(J)
      ALSS = ALSS+(GAP(J)+GAP(J+1))*DELYSS*0.5
      AGAP2 = AGAP2+AGAP1
      AGAPA1 = AGAP2/J
    5 CONTINUE
C
      ALSS = ALSS+ALEAK*(AGAPA1/GPS)
      SQFAC = SQRT(2.*ABS(PBAR)/RHOINF)
      QL = CFSS*ALSS*SQFAC*SIGN(1.,PBAR)
      IF (ISTNSL.NE.ON) RETURN
      WRITE (6,6) GAP,AIRLEN,FX,FY,FZ,FK,FM,FN
C
C
      RETURN
C
    6 FORMAT (//12H  STERN SEAL/26H  GAP (FT.) PORT TO STBD. /11E11.3/28
```

```
1H  AIRLEN(FT.) PORT TO STBD. /11E11.3/10X,23HSTNSL FX,FY,FZ,FK,FM,SSL
2FN/6E15.4)                                                        SSL
 END                                                               SSL

 FUNCTION T1 (X)                                                   T1
C                                                                  T1
 IF (ABS(X)-.1) 1,1,2                                              T1
1 T1 = X*(1.-X*X/10.0)/3.                                          T1
 RETURN                                                            T1
2 T1 = (SIN(X)-X*COS(X))/(X*X)                                     T1
 RETURN                                                            T1
 END                                                               T1

 FUNCTION T2 (X)                                                   T2
C                                                                  T2
 IF (ABS(X)-.1) 1,1,2                                              T2
1 T2 = 1.-X*X/6.                                                   T2
 RETURN                                                            T2
2 T2 = SIN(X)/X                                                    T2
 RETURN                                                            T2
 END                                                               T2

 INTEGER ON                                                        WAVS
 COMMON /BMCO/ IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY WAVS
 COMMON /CONST/ PI,RAD,UO                                          WAVS
 COMMON /GEOM/ WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM,DELS(4,WAVS
110),XCP,ZCP                                                       WAVS
 COMMON /GEOMBS/ DETABX(11),DETABT(11),ARM1B(10),ARM2B(10),DFBS(10)WAVS
1,TSKIB(10)                                                        WAVS
 COMMON /GEOMSS/ DETADX(11),DETADT(11),ARM1S(10),DFSS(10),TSKIS(10)WAVS
1,ARM2S(10)                                                        WAVS
 COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,AMWAVS
1I(201),XI(201),YI(201),ZI(201),XS,ZS,HPHO                         WAVS
 COMMON /MWAVE/ FXW(2),FYW(2),FZW(2),FKW(2),FMW(2),FNW(2)          WAVS
 COMMON /PLENUM/ XLBW,XBBW,ABW,BUPHGT,DVWDOT                       WAVS
 COMMON /PRTINT/ ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IWAVS
1RUD,IPROP,IAEROD,IRHS                                             WAVS
 COMMON /PROMOD/ PROMO1,PROMO2,PROMO3,PROMO4,PROMO5,PROMO6,PROMO7  WAVS
 COMMON /RISER/ AMPTC                                              WAVS
 COMMON /SIDE/ FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSWAVS
1W,VAREA,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ,AVRMSW,DELX,VTC         WAVS
 COMMON /VARBLE/ VAL(40)                                           WAVS
 COMMON /WAVE/ ETA(4,11),AW(20),OMEGA(20),DVOLW,NWAVE,BETA,FXWAV,FYWAVS
1WAV,FZWAV,FKWAV,FMWAV,FNWAV,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBWAVS
2BAR                                                               WAVS
 COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB,ACWAVS
11(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),AC5(20,5,7),AC6(20,5WAVS
2,7),AC7(20,5,7),AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),AS1(20,5,7),AWAVS
```

```
3S2(20,5,7),AS3(20,5,7),AS4(20,5,7),AS5(20,5,7),AS6(20,5,7),AS7(20,
45,7),AS0(20,5,7),AS00(20,5,7),AS8(20,5,7),BB(36),XREF,RX
 DIMENSION WC0(2), WC00(2), WC1(2), WC2(2), WC3(2), WC4(2), WC5(2),
1 WC6(2), WC7(2), WC8(2)
 DIMENSICN WS0(2), WS00(2), WS1(2), WS2(2), WS3(2), WS4(2), WS5(2),
1 WS6(2), WS7(2), WS8(2)
 EQUIVALENCE (VAL(2),U), (VAL(3),V), (VAL(4),W), (VAL(5),P), (VAL(6
1),Q), (VAL(7),R), (VAL(8),PHI), (VAL(9),THETA), (VAL(10),Z), (VAL(
211),BMASS), (VAL(21),X), (VAL(22),Y), (VAL(23),PSI), (VAL(24),PB)
 EQUIVALENCE (VAL(16),ETACG)
C
 IF (NWAVE.EQ.0) RETURN
C
C CALCULATION OF SHIFT OF XCPO
C
 XCPU = XCP+0.001975*(U*0.5921-30.0)**2-0.974
 XCPC = SHXYAX(XCPU,ZCP,THETA,PI)
 GAMMA = BETA-PSI
 SIGAM = SIN(GAMMA)
 COGAM = COS(GAMMA)
 FO = -X*COSBET-Y*SINBET
 DVOLW = 0.0
 ETACG = 0.0
 N = NSTA(3)
C
 DO 1 J=1,N
 DETABX(J) = 0.0
1 CONTINUE
C
 N = NSTA(4)
C
 DO 2 J=1,N
 DETADX(J) = 0.0
2 CONTINUE
C
C
 DO 3 J=1,4
 N = NSTA(J)
C
 DO 3 K=1,N
3 ETA(J,K) = 0.0
C
C
 DO 4 J=1,2
 FXW(J) = 0.0
 FYW(J) = 0.0
 FZW(J) = 0.0
 FKW(J) = 0.0
```

```
      FMW(J) = 0.0
      FNW(J) = 0.0
    4 CCNTINUE
C
      XSS = -XS
      IF (IMT.EQ.2) XSS = XMI(IX)
      IP = 1+(THEBAR*RAD-STH)/DTH
      IP = MAXO(MINO(IP,NTH),1)
      IP1 = MINO(IP+1,NTH)
      DTHETA = (IP-1)*DTH+STH
      DIP = (THETA*RAD-DTHETA)/DTH
C     TIME RISE FACTOR FOR WAVE AMPLITUDE
      AMPFAC = 1.-EXP(-TIME/AMPTC)
C
      DO 11 I=1,NWAVE
      OM1 = CMEGA(I)
      OM2 = CM1*CM1
      XWK = CM2/G
      AA = AW(I)*AMPFAC
      FT = OM1*TIME+XWK*FO
      AL = XWK*COGAM
      IAA = 1+(ABS(AL)-SAL)/DAL
      IAA = MAXO(MINO(IAA,NAL),1)
      IAA1 = MINO(IAA+1,NAL)
      DAA = (IAA-1)*DAL+SAL
      DIA = (ABS(AL)-DAA)/DAL
      SALP = SIGN(1.,AL)
C
C     WAVE FORCES AND MOMENTS ON THE SIDEWALLS
C
C
      DO 7 J=1,2
      YLSW = (2*J-3)*YSW
      WE = FT+XWK*SIGAM*YLSW
      ST = SIN(WE)
      CT = COS(WE)
      DS = ZBAR+ZS+YLSW*PHIBAR
      DSR = DS-(XREF-XS)*THEBAR
      ID = 1.+(DSR*12.-SDS)/DDS
      ID = MAXO(MINO(ID,NDS),1)
      DDSR = (ID-1)*DDS+SDS
      CID = (DSR*12.-DDSR)/DDS
      ID1 = MINO(ID+1,NDS)
      DSS = DS-XSS*THEBAR
      ZOR1 = (SIGN(1.,DSS)+1.)*0.5
      DSS = DSS*ZOR1
      IDSS = 1.5+(DSS-SBB)/DBB
      IDSS = MINO(NBB,IDSS)
```

```
      BS = BB(IDSS)                                                     WAVS1230
      CK = CCS(XWK*COGAM*XSS)                                           WAVS1240
      A33S = (RHO*PI*BS*BS)*0.125                                       WAVS1250
      SK = SIN(XWK*COGAM*XSS)                                           WAVS1260
      A22S = (RHO*.4*PI*DSS*DSS)*0.5                                    WAVS1270
      A42S = 0.0                                                        WAVS1280
C                                                                       WAVS1290
C     INTERPCLATION OF WAVE TABLES                                      WAVS1300
C                                                                       WAVS1310
      K = 1                                                             WAVS1320
      L = IAA                                                           WAVS1330
    5 CONTINUE                                                          WAVS1340
      BCO = ACO(L,ID,IP)                                                WAVS1350
      BC00 = AC00(L,ID,IP)                                              WAVS1360
      BC1 = AC1(L,ID,IP)                                                WAVS1370
      BC2 = AC2(L,ID,IP)                                                WAVS1380
      BC3 = AC3(L,ID,IP)                                                WAVS1390
      BC4 = AC4(L,ID,IP)                                                WAVS1400
      BC5 = AC5(L,ID,IP)                                                WAVS1410
      BC6 = AC6(L,ID,IP)                                                WAVS1420
      BC7 = AC7(L,ID,IP)                                                WAVS1430
      BC8 = AC8(L,ID,IP)                                                WAVS1440
      BSO = ASO(L,ID,IP)                                                WAVS1450
      BS00 = AS00(L,ID,IP)                                              WAVS1460
      BS1 = AS1(L,ID,IP)                                                WAVS1470
      BS2 = AS2(L,ID,IP)                                                WAVS1480
      BS3 = AS3(L,ID,IP)                                                WAVS1490
      BS4 = AS4(L,ID,IP)                                                WAVS1500
      BS5 = AS5(L,ID,IP)                                                WAVS1510
      BS6 = AS6(L,ID,IP)                                                WAVS1520
      BS7 = AS7(L,ID,IP)                                                WAVS1530
      BS8 = AS8(L,ID,IP)                                                WAVS1540
      WCO(K) = BCO+DID*(ACO(L,ID1,IP)-BCO)+DIP*(ACO(L,ID,IP1)-BCO+DID*(AWAVS1550
     1CO(L,ID1,IP1)-ACO(L,ID,IP1)-ACO(L,ID1,IP)+BCO))                   WAVS1560
      WC00(K) = BC00+DID*(AC00(L,ID1,IP)-BC00)+DIP*(AC00(L,ID,IP1)-BC00+WAVS1570
     1DID*(AC00(L,ID1,IP1)-AC00(L,ID,IP1)-AC00(L,ID1,IP)+BC00))         WAVS1580
      WC1(K) = BC1+DID*(AC1(L,ID1,IP)-BC1)+DIP*(AC1(L,ID,IP1)-BC1+DID*(AWAVS1590
     1C1(L,ID1,IP1)-AC1(L,ID,IP1)-AC1(L,ID1,IP)+BC1))                   WAVS1600
      WC2(K) = BC2+DID*(AC2(L,ID1,IP)-BC2)+DIP*(AC2(L,ID,IP1)-BC2+DID*(AWAVS1610
     1C2(L,ID1,IP1)-AC2(L,ID,IP1)-AC2(L,ID1,IP)+BC2))                   WAVS1620
      WC3(K) = BC3+DID*(AC3(L,ID1,IP)-BC3)+DIP*(AC3(L,ID,IP1)-BC3+DID*(AWAVS1630
     1C3(L,ID1,IP1)-AC3(L,ID,IP1)-AC3(L,ID1,IP)+BC3))                   WAVS1640
      WC4(K) = BC4+DID*(AC4(L,ID1,IP)-BC4)+DIP*(AC4(L,ID,IP1)-BC4+DID*(AWAVS1650
     1C4(L,ID1,IP1)-AC4(L,ID,IP1)-AC4(L,ID1,IP)+BC4))                   WAVS1660
      WC5(K) = BC5+DID*(AC5(L,ID1,IP)-BC5)+DIP*(AC5(L,ID,IP1)-BC5+DID*(AWAVS1670
     1C5(L,ID1,IP1)-AC5(L,ID,IP1)-AC5(L,ID1,IP)+BC5))                   WAVS1680
      WC6(K) = BC6+DID*(AC6(L,ID1,IP)-BC6)+DIP*(AC6(L,ID,IP1)-BC6+DID*(AWAVS1690
     1C6(L,ID1,IP1)-AC6(L,ID,IP1)-AC6(L,ID1,IP)+BC6))                   WAVS1700
```

```
      WC7(K) = BC7+DID*(AC7(L,ID1,IP)-BC7)+DIP*(AC7(L,ID,IP1)-BC7+DID*(AWAVS1710
     1C7(L,ID1,IP1)-AC7(L,ID,IP1)-AC7(L,ID1,IP)+BC7))                   WAVS1720
      WC8(K) = BC8+DID*(AC8(L,ID1,IP)-BC8)+DIP*(AC8(L,ID,IP1)-BC8+DID*(AWAVS1730
     1C8(L,ID1,IP1)-AC8(L,ID,IP1)-AC8(L,ID1,IP)+BC8))                   WAVS1740
      WS0(K) = BS0+DID*(AS0(L,ID1,IP)-BS0)+DIP*(AS0(L,ID,IP1)-BS0+DID*(AWAVS1750
     1S0(L,ID1,IP1)-AS0(L,ID,IP1)-AS0(L,ID1,IP)+BS0))                   WAVS1760
      WS00(K) = BS00+DID*(AS00(L,ID1,IP)-BS00)+DIP*(AS00(L,ID,IP1)-BS00+WAVS1770
     1DID*(AS00(L,ID1,IP1)-AS00(L,ID,IP1)-AS00(L,ID1,IP)+BS00))         WAVS1780
      WS1(K) = BS1+DID*(AS1(L,ID1,IP)-BS1)+DIP*(AS1(L,ID,IP1)-BS1+DID*(AWAVS1790
     1S1(L,ID1,IP1)-AS1(L,ID,IP1)-AS1(L,ID1,IP)+BS1))                   WAVS1800
      WS2(K) = BS2+DID*(AS2(L,ID1,IP)-BS2)+DIP*(AS2(L,ID,IP1)-BS2+DID*(AWAVS1810
     1S2(L,ID1,IP1)-AS2(L,ID,IP1)-AS2(L,ID1,IP)+BS2))                   WAVS1820
      WS3(K) = BS3+DID*(AS3(L,ID1,IP)-BS3)+DIP*(AS3(L,ID,IP1)-BS3+DID*(AWAVS1830
     1S3(L,ID1,IP1)-AS3(L,ID,IP1)-AS3(L,ID1,IP)+BS3))                   WAVS1840
      WS4(K) = BS4+DID*(AS4(L,ID1,IP)-BS4)+DIP*(AS4(L,ID,IP1)-BS4+DID*(AWAVS1850
     1S4(L,ID1,IP1)-AS4(L,ID,IP1)-AS4(L,ID1,IP)+BS4))                   WAVS1860
      WS5(K) = BS5+DID*(AS5(L,ID1,IP)-BS5)+DIP*(AS5(L,ID,IP1)-BS5+DID*(AWAVS1870
     1S5(L,ID1,IP1)-AS5(L,ID,IP1)-AS5(L,ID1,IP)+BS5))                   WAVS1880
      WS6(K) = BS6+DID*(AS6(L,ID1,IP)-BS6)+DIP*(AS6(L,ID,IP1)-BS6+DID*(AWAVS1890
     1S6(L,ID1,IP1)-AS6(L,ID,IP1)-AS6(L,ID1,IP)+BS6))                   WAVS1900
      WS7(K) = BS7+DID*(AS7(L,ID1,IP)-BS7)+DIP*(AS7(L,ID,IP1)-BS7+DID*(AWAVS1910
     1S7(L,ID1,IP1)-AS7(L,ID,IP1)-AS7(L,ID1,IP)+BS7))                   WAVS1920
      WS8(K) = BS8+DID*(AS8(L,ID1,IP)-BS8)+DIP*(AS8(L,ID,IP1)-BS8+DID*(AWAVS1930
     1S8(L,ID1,IP1)-AS8(L,ID,IP1)-AS8(L,ID1,IP)+BS8))                   WAVS1940
      IF (K.EQ.2) GO TO 6                                               WAVS1950
      K = 2                                                             WAVS1960
      L = IAA1                                                          WAVS1970
      GO TO 5                                                           WAVS1980
    6 BC0 = WC0(1)+DIA*(WC0(2)-WC0(1))                                  WAVS1990
      BC00 = WC00(1)+DIA*(WC00(2)-WC00(1))                              WAVS2000
      BC1 = WC1(1)+DIA*(WC1(2)-WC1(1))                                  WAVS2010
      BC2 = WC2(1)+DIA*(WC2(2)-WC2(1))                                  WAVS2020
      BC3 = WC3(1)+DIA*(WC3(2)-WC3(1))                                  WAVS2030
      BC4 = WC4(1)+DIA*(WC4(2)-WC4(1))                                  WAVS2040
      BC5 = WC5(1)+DIA*(WC5(2)-WC5(1))                                  WAVS2050
      BC6 = WC6(1)+DIA*(WC6(2)-WC6(1))                                  WAVS2060
      BC7 = WC7(1)+DIA*(WC7(2)-WC7(1))                                  WAVS2070
      BC8 = WC8(1)+DIA*(WC8(2)-WC8(1))                                  WAVS2080
      BS0 = (WS0(1)+DIA*(WS0(2)-WS0(1)))*SALP                           WAVS2090
      BS00 = (WS00(1)+DIA*(WS00(2)-WS00(1)))*SALP                       WAVS2100
      BS1 = (WS1(1)+DIA*(WS1(2)-WS1(1)))*SALP                           WAVS2110
      BS2 = (WS2(1)+DIA*(WS2(2)-WS2(1)))*SALP                           WAVS2120
      BS3 = (WS3(1)+DIA*(WS3(2)-WS3(1)))*SALP                           WAVS2130
      BS4 = (WS4(1)+DIA*(WS4(2)-WS4(1)))*SALP                           WAVS2140
      BS5 = (WS5(1)+DIA*(WS5(2)-WS5(1)))*SALP                           WAVS2150
      BS6 = (WS6(1)+DIA*(WS6(2)-WS6(1)))*SALP                           WAVS2160
      BS7 = (WS7(1)+DIA*(WS7(2)-WS7(1)))*SALP                           WAVS2170
      BS8 = (WS8(1)+DIA*(WS8(2)-WS8(1)))*SALP                           WAVS2180
```

```
C
C     SHIFT MOMENT CENTER FROM XREF TO C.G.
C
      BCOO = BCOO-(XS-XREF)*BCO
      BC3 = BC3-(XS-XREF)*BC1
      BC4 = BC4-(XS-XREF)*BC2
      BC6 = BC6-(XS-XREF)*BC5
      BSOO = BSOO-(XS-XREF)*BSO
      BS3 = BS3-(XS-XREF)*BS1
      BS4 = BS4-(XS-XREF)*BS2
      BS6 = BS6-(XS-XREF)*BS5
C
C     CALCULATE WAVE FORCES AND MOMENTS
C
      FZC = BS1-XWK*G*(BS2+BSO)-U*OM1*(-A33S*CK-AL*BS2)
      FZS = BC1-XWK*G*(BC2+BCO)+U*OM1*(-A33S*SK+AL*BC2)
      FMC = BS3-XWK*G*(BS4+BSOO)-U*OM1*(-A33S*XSS*CK-BC2-AL*BS4)
      FMS = BC3-XWK*G*(BC4+BCOO)+U*OM1*(-A33S*XSS*SK-BS2+AL*BC4)
      FYC = XWK*G*(BC5+BCO)-U*OM1*(-A22S*SK+AL*BC5)
      FYS = -XWK*G*(BS5+BSO)-U*OM1*(-A22S*CK-AL*BS5)
      FNC = XWK*G*(BC6+BCOO)-U*OM1*(-A22S*XSS*SK-BS5+AL*BC6)
      FNS = -XWK*G*(BS6+BSOO)-U*OM1*(-A22S*XSS*CK-BC5-AL*BS6)
      FKC = XWK*G*(BC7-BC8)+U*OM1*(-A42S*SK+AL*BC8)
      FKS = -XWK*G*(BS7-BS8)+U*OM1*(-A42S*CK-AL*BS8)
      FZW(J) = FZW(J)-AA*(FZC*CT+FZS*ST)
      FMW(J) = FMW(J)+AA*(FMC*CT+FMS*ST)
      FYW(J) = FYW(J)-AA*(FYC*CT+FYS*ST)*SIGAM
      FNW(J) = FNW(J)-AA*(FNC*CT+FNS*ST)*SIGAM
      FKW(J) = FKW(J)-AA*(FKC*CT+FKS*ST)*SIGAM
      FXW(J) = FXW(J)-2.*AA*RHO*G*BS*DSS*SK*CT
    7 CONTINUE
C
      IF (IMT.EQ.2) GO TO 11
C
C     WAVE ELEVATION AROUND THE SIDEWALLS AND SEALS
C
C
      DO 8 J=1,4
      N = NSTA(J)
C
      DO 8 K=1,N
      ETA(J,K) = ETA(J,K)+SIN(XWK*(-XX(J,K)*COGAM-YY(J,K)*SIGAM)+FT)*AA
    8 CONTINUE
C
      ETACG = ETACG+AA*SIN(FT)
      N = NSTA(3)
C
      DO 9 J=1,N
```

```
      ARG = AA*COS(XWK*(-XX(3,J)*COGAM)+FT)
      DETABX(J) = DETABX(J)-XWK*COGAM*ARG
    9 CONTINUE
C
      N = NSTA(4)
C
      DO 10  J=1,N
      ARG = AA*COS(XWK*(-XX(4,J)*COGAM)+FT)
      DETADX(J) = DETADX(J)-XWK*COGAM*ARG
   10 CONTINUE
C
C
C     WAVE PUMPING
C
      X1 = XWK*XLBW*COGAM*0.5
      X2 = XWK*XBBW*SIGAM*0.5
      FTT = FT-XWK*XCPC*COGAM
      DVOLW = DVOLW+AA*ABW*T2(X1)*T2(X2)*SIN(FTT)
   11 CONTINUE
C
      IF (TIME.EQ.0.0) GO TO 12
      DVWDOT = (DVOLW-DVOLWO)/(TIME-TIMEO)
   12 DVOLWO = DVOLW
      TIMEO = TIME
      IF (IMT.EQ.2) RETURN
C
C     TOTAL WAVE FORCES AND MOMENTS
C
      FXWAV = FXW(1)+FXW(2)
      FYWAV = FYW(1)+FYW(2)
      FZWAV = FZW(1)+FZW(2)
      FKWAV = FKW(1)+FKW(2)+(FZW(2)-FZW(1))*YSW+FYWAV*ZBAR
      FMWAV = FMW(1)+FMW(2)-FXWAV*ZBAR
      FNWAV = FNW(1)+FNW(2)+(FXW(1)-FXW(2))*YSW
      IF (IWAVES.NE.ON) RETURN
      WRITE (6,13) ((ETA(I,J),J=1,11),I=1,4),ETACG,DVOLW,FXWAV,FYWAV,FZW
     1AV,FKWAV,FMWAV,FNWAV
C
C
      RETURN
C
   13 FORMAT (/10X,5HWAVES/63H WAVE ELEVATIONS AT CRAFT STATIONS RELATIV
     1E TO CALM WATER (FT.)/14H PORT SIDEWALL/11F10.5/14H STBD SIDEWALL/
     211F10.5/9H BOW SEAL/11F10.5/11H STERN SEAL/11F10.5/25H WAVE ELEVAT
     3ION AT C.G. =F10.5,10X,43HPLENUM VOLUME LOST DUE TO WAVES (CU. FT.
     4) =F15.5/10X,23HWAVES FX,FY,FZ,FK,FM,FN/6E15.4)
      END
```

LIST OF REFERENCES

1. Janes' Surface Skimmers Hovercraft and Hydrofoils, 9th ed., Franklin Watts, Inc., New York, 1975-1976.

2. Oceanics Incorporated, Report No. 71-84A, August 1971, Technical Industrial Park, Plainview, N. Y., 11803.

3. Leo, D. G., and Boncal, R., XR-3 Surface Effects Ships Test Craft: A Mathematical Model and Simulation Program with Verification, Master's Thesis, Naval Postgraduate School, December, 1973.

4. Layton, D. M., Evaluation of the XR-3 Test Craft, Annual Report, Naval Postgraduate School, 1971.

5. Finley, R. A., Refinements of the Seal Subroutines and Fan Air Flow Maps for the XR-3 Loads and Motion Program, Master's Thesis, Naval Postgraduate School, December 1974.

6. Forbes, G. T., Validation of the Nonlinear Six Degrees of Freedom Mathematical Model of the XR-3 Captured Air Bubble Surface Effect Ship in Calm Water, Master's Thesis, Naval Postgraduate School, December, 1974.

7. Menzel, R. F., Study of the Roll and Pitch Transients in Calm Water Using the Simulated Performance of XR-3 Surface Effect Ship Loads and Motion Computer Program, Master's Thesis, Naval Postgraduate School, December 1975.

8. Mitchell, W. R., Investigation of Methods to Optimize Captured Air Bubble Surface Effect Ship Digital Simulation for Irregular Sea Conditions, Master's Thesis, Naval Postgraduate School, September 1974.

9. Private communication from F. Wilson to Professor A. Gerba, dated, 10 May 1974

10. Private communication from J. E. Blaloch to Professor A. Gerba, dated, 9 September 1975

11. Principles of Naval Architecture, Revised ed., Society of Naval Architects and Marine Engineers, 1967.

12. W. R., Church Computer Center, Naval Postgraduate School, Users Manual, 1st. ed., March 1970.

13. IBM Systems Library, IBM System/360 Operating System FORTRAN IV (G and H) Programmers Guide, Form No. (25) GC-28-6817, 5th ed., September 1973.

14. Booth, B. F., Naval Postgraduate School Thesis. To be published in June 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Associate Professor Alex Gerba, Jr., Code 52Gz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	3
4.	Professor George J. Thaler, Code 52Tr Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
5.	Associate Professor Donald M. Layton, Code 57Ln Department of Aeronautical Engineering Naval Postgraduate School Monterey, California 93940	1
6.	Mr. Arnold W. Anderson, Code PMS-304-31A.1 Department of the Navy Surface Effect Ships Project Office, PM-17 Bethesda, Maryland 20034 (P. O. Box 34401)	6
7.	Lieutenant John M. Boggio Naval Guided Missile School Dam Neck Virginia Beach, Virginia 23461	1