Report No. FHWA-RD-76-163

HIGHWAY-VEHICLE-OBJECT SIMULATION MODEL--1976

Vol. 2. Programmers Manual

February 1976

Final Report

Document is available to the public through
the National Technical Information Service,
Springfield, Virginia 22161

Prepared for

FEDERAL HIGHWAY ADMINISTRATION

Offices of Research & Development

Washington, D. C. 20590

NOTICE

FHWA DISTRIBUTION NOTICE

Limited copies of this report are being distributed by memorandum to individual researchers involved in computer simulations of highway vehicles and impacts with roadside obstacles.

A limited number of additional copies of this report is available from the Protective Systems Group of the Structures and Applied Mechanics Division, Office of Research.

Copies may also be obtained from the National Technical Information Service (NTIS), Department of Commerce, 5285 Port Royal Road, Springfield, Virginia 22161. A small charge is imposed for copies provided by NTIS.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA-RD-76-163		

4. Title and Subtitle	5. Report Date
Highway-Vehicle-Object Simulation Model - 1976, Programmers Manual	February 1976
	6. Performing Organization Code
7. Author(s): David J. Segal	8. Performing Organization Report No.: ZR-5461-V-7
9. Performing Organization Name and Address: Calspan Corporation, P. O. Box 235, Buffalo, New York 14221	10. Work Unit No. (TRAIS): FCP 31H2232
	11. Contract or Grant No.: DOT-FH-11-8265
12. Sponsoring Agency Name and Address: U.S. Department of Transportation, Federal Highway Administration, Contracts and Procurement Division, Washington, D.C. 20590	13. Type of Report and Period Covered: Final Report, February 1974 - February 1976
	14. Sponsoring Agency Code: S0608

15. Supplementary Notes

FHWA Contract Manager: Morton S. Oskard

16. Abstract

A series of reports have been written to document revised and updated versions of the simulation of highway-vehicle-object interactions in a single vehicle highway environment. The programs documented were developed under FHWA sponsorship to provide the highway safety community with an analytical means of evaluating the effects of highway/roadside environment on safety.

This manual is addressed to the applications programmer who might wish to modify or extend the HVOSM. The detailed descriptions of the subroutines and linkages among them are designed to be used in conjunction with a source program listing. Ancillary information of interest to the programmer is also included.

This manual is one of four volumes.

Contractors Report No. ZR-5461-V	Short Title
-6	HVOSM - 1976 Users Manual
-7	HVOSM - 1976 Programmers Manual
-8	HVOSM - 1976 Engineering Manual - Analysis
-4	HVOSM - 1976 Engineering Manual - Validation

17. Key Words	18. Distribution Statement
HVOSM, Vehicle Dynamics Computer Simulation	No restrictions. This report is available to the public through the National Technical Information Service, Springfield, Virginia, 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	575	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

FOREWORD

This report is one of four manuals prepared under Contract Number DOT-FH-11-8265 for the Federal Highway Administration, U.S. Department of Transportation for the purpose of summarizing and upgrading documentation of the Highway-Vehicle-Object Simulation Model (HVOSM). The HVOSM had been previously developed for the Federal Highway Administration (FHWA) by the Calspan Corporation (formerly Cornell Aeronautical Laboratory) under Contract Number CPR-11-3988 during the period from 1966 to 1971 and extended under this contract. Contained in this report is a description of the experimental validation procedures employed in determining the degree and range of validity of the HVOSM.

Complete documentation of the HVOSM is contained in the following manuals:

- Highway-Vehicle-Object Simulation Model
  Volume 1 - Users Manual
- Highway-Vehicle-Object Simulation Model
  Volume 2 - Programmers Manual
- Highway-Vehicle-Object Simulation Model
  Volume 3 - Engineering Manual - Analysis
- Highway-Vehicle-Object Simulation Model - Volume 4 - Engineering Manual - Validation

This report has been reviewed and is approved by:

Edwin A. Kidd

Edwin A. Kidd, Head
Transportation Research Department

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

In 1966 Calspan Corporation (formerly Cornell Aeronautical Laboratory, Inc.) began development of a general mathematical model and computer simulation of the dynamic responses of an automobile in accident situations under Contract CPR-11-3988 with the Bureau of Public Roads.

The mathematical model of vehicle dynamics developed in the first year of that effort included the general three-dimensional motion resulting from vehicle control inputs, traversal of irregular terrain, or from collisions with simple roadside barriers. The model was subsequently named the Highway-Vehicle-Object Simulation Model (HVOSM). Later, the model was further developed and a comprehensive validation program was carried out including a series of repeatable full-scale tests with an instrumented vehicle in order to objectively assess the degree of validity of the vehicle model. Extensive measurements of the vehicle parameters required for input to the HVOSM were made under a subcontract with the Ford Motor Company as a part of the validation procedure. This effort was reported in Reference 1 and the model as described therein has been referred to as the V-3 version of the HVOSM.

Modifications were subsequently made to the simulation in order to study the effects of terrain (specifically, railroad grade crossings) on vehicle control ability. The impact routines were removed and extended terrain definition capabilities were added along with a more realistic model of suspension properties. This program version (Reference 2) has been informally referred to as the V-4 version of the HVOSM and has since been used extensively for study of roadway and roadside geometrics.

Further developments of HVOSM aimed at providing a simulation model suitable for the study of the complex dynamics resulting from pre-collision evasive maneuvers were reported in Reference 3. This version, informally called the V-7 version of the HVOSM, includes a detailed model of the braking and driving systems and an empirically based definition of the relationships between longitudinal and lateral tire forces through the inclusion of rotational degrees of freedom of the four vehicle wheels.

During development of the HVOSM, documentation efforts primarily fulfilled the objectives of maintaining communication within the program development structure, ensuring quality control of the development and providing a historical reference. It was, however, recognized early in the development of the HVOSM, that this state-of-the-art advance in the modeling of a vehicle and its environment could be put to best use through its widespread distribution to organizations interested in its application to highway safety. As a result, distribution of the HVOSM was begun before its development was complete and before instructional documentation could be provided.

Recognizing the need to bring documentation of the several HVOSM versions together and to provide the highway safety community with an effective description of the programs and their use, the Federal Highway Administration (FHWA) awarded Calspan Corporation contract number DOT-FH-11-8265 for the purpose of providing such documentation for the then existing versions of the HVOSM.

Three versions of the HVOSM were covered by this documentation. They were, the HVOSM-SMI1 (Sprung Mass Impact) version (formerly known as the V-3 version), the HVOSM-RD1 (Road Design) version (formerly known as the V-4 version) and the HVOSM-VD1 (Vehicle Dynamics) version (formerly known as the V-7 version). Under the first phase of that effort, only those versions as developed by Calspan were covered by the documentation.

The second phase of contract number DOT-FH-11-8265 called for extension of the capabilities of the HVOSM by adding new features, including some additional modifications made by other research organizations, and providing additional ease of use features.

Accordingly, Calspan has:

- Generalized the basic vehicle model to include the capability for simulating an independent front and rear suspension vehicle and a vehicle with solid front and rear axles.

- Generalized the tire model to allow specification of up to four different tires on a vehicle and revised the friction ellipse tire model.

- Combined the sprung mass impact version with the roadside design version resulting in only two program versions at the end of the second phase.

- Incorporated the Preview-Predictor Driver Model described in Reference 4 into the vehicle dynamics model.

- Incorporated impact forces due to localized structural hard points into the sprung mass impact algorithm. This modification was originally developed by the Texas Transportation Institute (TTI) and was added as reported in Reference 5.

- Extended the curb impact algorithm to allow up to six planes to describe a curb. This modification was also developed by TTI and was reported in Reference 6.

- Developed a road roughness algorithm to allow determination of the effects of road roughness on vehicle performance.

- Revised imput and output format to provide an easy to use, more flexible data interface.

- Developed a Pre-Processing Program to calculate a number of program inputs including vehicle and terrain data or to supply input cards from a stored library of vehicle data.

The documentation provided now covers the two program versions: the HVOSM-RD2 Version (Roadside Design) and the HVOSM-VD2 Version (Vehicle Dynamics). It is intended to be a base to which further developments and modifications to the HVOSM can be added, thus providing a uniform reporting format and centralized source of information for the many HVOSM users. It consists of four volumes, each describing a separate aspect of the HVOSM. Two volumes are directed toward the engineer/analyst containing the analysis (derivation of governing equations, assumptions, and development of controling logic) and experimental validation. Another volume is directed toward the general program user and contains analysis/program symbology, descriptions of the models and solution procedures, descriptions of input requirements and program application examples. The fourth volume of documentation is intended for use by those interested in the detailed computer programs. This volume contains descriptions of the computer code including a discussion of subroutine functions, annotated flowcharts and program listings. Also included are a list of program changes, a description of program stops and messages, and computer system requirements necessary to run the programs.

This report contains detailed descriptions of the three HVOSM computer programs. Symbology is defined in Section 2. Section 3 contains detailed descriptions of the programs including subroutine functions, flowcharts, listings, messages and codes, and program changes. Section 4 contains system requirements for execution of the programs, a listing of the HVOSM Preprocessing program is given in Section 5, and references are contained in Section 6.

2. SYMBOLOGY

The HVOSM symbology is presented in this section with a cross-reference between analytical and programming symbols. The first listing of symbols is ordered with respect to analytical symbol and includes a corresponding program symbol, a brief definition and an equation number referencing the calculation of the variable in the "HVOSM Engineering Manual - Analysis". Input variables are indicated by an I in the equation number column.

The second listing of variables is organized by program symbol name and includes a corresponding analytical variable or expression and variable usage in each program version. The codes U and A under the program version name indicate that the variable is used or appears, but is not used respectively in that version.

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
a	A	I	Distance along vehicle fixed x axis from the sprung mass center of gravity to the center line of the front wheels	in.
a_i, b_i, c_i		155	Directional components of a line perpendicular to both the normal to the wheel plane and the radial tire force, F_{R_i}	
APD APDMAX	APD APDMAX	345 I	Accelerator pedal deflection and maximum accelerator pedal deflection	in
$a_{s_i}, b_{s_i}, c_{s_i}$	AS(4) BS(4) CS(4)	258	Directional components of a line perpendicular to both a normal to the tire-terrain contact plane and the line of intersection of the wheel and ground planes	
$a_{x_i}, b_{x_i}, c_{x_i}$	AX(4) BX(4) CX(4)	99	Direction components of a line perpendicular to both a normal to the tire-terrain contact plane and the vehicle fixed y axis	
$a_{y_i}, b_{y_i}, c_{y_i}$	AY(4) BY(4) CY(4)	104	Direction component of a line perpendicular to both a normal to the tire-terrain contact plane and the vehicle fixed x axis	
[A]	AMTX(3,3)	53	Transformation matrix from vehicle fixed to space fixed coordinate systems	
$(A_{INT})_i$	AINTI	287	Intersection area of cutting plane i with the sprung mass	in^2
$[A_j]$	AJMTX(3,3)	134	Transformation matrix from wheel fixed to vehicle fixed coordinate systems	
AMU	AMU	I	Tire-terrain friction coefficient at zero speed and nominal tire loading	
AMUG	AMUG(5)	I	Tire-terrain friction coefficient factor for 5 terrain tables	
$(AP)_F$	APF(21)	I	Anti-pitch coefficients for front suspension positive for anti-pitch for forward braking	lb/lb-ft
$(AP)_R$	APR(21)	I	Anti-pitch coefficients for rear suspension, effective at the wheels; positive for anti-pitch effect for forward braking	lb/lb-ft
$(AR)_j$	ARBRF ARBRR	I	Drive axle ratio (propeller shaft speed/wheel speed). Default of 1.0	
A_0, A_1, A_2	A0,A1,A2	I	Constant coefficients for tire side force due to slip angle	
A_3, A_4	A3,A4	I	Constant coefficients for tire side force due to camber angle	
b	B	I	Distance along the vehicle fixed x axis from the sprung mass center of gravity to the centerline of the rear wheels (entered positive)	in.
[B]	BMTX(3,3)	134	Transformation matrix from wheel fixed to space fixed coordinate systems	
B_{FP1} B_{FP2}	BFP1 BFP2	I	First and second order coefficients for relationship between brake pedal force and brake system pressure	psi/lb psi/lb^2
$[B_n]$	BNMTX(3,3)	60	Transformation matrix from orientation of vehicle axes at indexing to space fixed axes (Euler angles= $\psi_n', \theta_n', \phi_n'$)	
C_{co}		215	Small angle camber stiffness	lb/rad
C_F, C_R	CF CR	I	Front and rear viscous damping coefficient for a single wheel, effective at the wheel for the front and at the spring at the rear	lb-sec/in
C'_F, C'_R	CFP CRP	I	Front and rear coulomb damping for a single wheel, effective at the wheel for the front and at spring for the rear	lb
$[C_i]$	CMTX(3,4)	110	Coefficient matrix for simultaneous solution of the ground contact point	
$\overline{CONS}$	CONS	I	Ratio of conserved energy to total energy absorbed by the sprung mass	
$[C_n]$	CNMTX(3,3)	60	Transformation matrix from vehicle fixed axes to most recently indexed axes (Euler angles= $\psi_r', \theta_r', \phi_r$)	
C_{RRMi}	RRMC(4)	I	Rolling resistance moment coefficient	lb-in/lb
C_{So}		214	Small angle cornering stiffness	lb/rad
(CT)	TCT(12)	I	Closed throttle engine torque	lb-ft

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
C_{Ti}	CT(4)	I	Circumferential tire force stiffness	lb
C'_ψ	CPSP	I	Coulomb resistance torque in the steering system effective at the wheels	lb-in
C_1, C_2, C_3	CONE CTWO CTHREE	I	Coefficients in relationship approximating aerodynamic and rolling resistance	
[D]	DMTX(10,11)	48	Mass matrix of coupled second order differential equations. Column 11 contains the forcing functions	
Dax	DELTAX	342	Desired vehicle acceleration	in/sec^2
DELB DELE DDEL	DELB DELE DDEL	I I I	Beginning, end, and incremental wheel deflection for entered front wheel camber table	in
DIST	DIST	I	Desired speed differential nulling distance	in
DRWHJ	DRWHJ	I	Incremental tire deflection for calculation of the equivalent tire force-deflection characteristic in the radial mode	in
D_{1i}, D_{2i}, D_{3i}	D1(4) D2(4) D3(4)	37	Direction components of a line perpendicular to the normals of both the wheel plane and the tire-terrain contact plane	
e_i	EI	320	Error between predicted and desired path at the ith viewing position	in
EN	EN	I	Number of points at which e_i is determined	
F_{APi}	APITCH	188	Anti-pitch force at wheel i	lb
F_{ARi}		187	Force at wheel i due to auxiliary roll stiffness	lb
F_B	FB		Resistance force normal to the contact surface of a deformable barrier	lb
FBRK	FBRK	346	Brake pedal force	lb
F_{c_i}	FC(4)	225	Circumferential tire force	lb
Fcxui Fcyui Fczui	FCXU(4) FCYU(4) FCZU(4)	254	Components of the circumferential tire force along the x,y, and z axes	lb
F_j	FJP(35)	144	Table of equivalent radial spring forces as a function of deflection	lb
F_{JFi}	FJF(4)	179	Jacking force at wheel i	lb
$(Fn)_t$	FN	298	Vehicle force produced by deformation of the vehicle structure normal to the contacted surface	lb

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
F_{NSTi}	FNSTI(3)	300	Structural hard point force	lb
F_{R_i}	FR(4)	114	Radial tire force in the plane of the wheel	lb
F'_{Ri}	FRCP(4)	212	Tire force perpendicular to the tire-terrain contact plane	lb
(FRICT)	FRICT	304	Friction force acting between the vehicle sprung mass and barrier	lb
F_{Rxui} F_{Ryui} FRzui	FRXU(4) FRYU(4) FRZU(4)	253	Components of F'_{Ri} along the sprung mass axes for wheel i	lb
$\Sigma F_{Rx}{}'_i$ $\Sigma F_{Ry}{}'_i$ $\Sigma F_{Rz}{}'_i$	SFRX(4) SFRY(4) SFRZ(4)	144 145 146	Summation of the components of radial spring mode forces over tire i, with respect to space	lb
F_{Si}	FS(4)	227	Tire side force in the plane of the tire-terrain contact patch perpendicular to the line of intersection of the wheel plane and ground plane	lb
F'_{Si}		220	Resultant side force corresponding to small angle properties for slip and camber angles	lb
$(F_{si})_{max}$		224	Maximum achievable side force as limited by the available friction	lb
ΣF_{xs}	SFXS	307 351	Sprung mass impact force or combination of rolling resistance and aerodynamic drag acting along the vehicle x axis	lb
Fsxui Fsyui Fszui	FSXU(4) FSYU(4) FSZU(4)	256	Components of tire side force, F_{si} along the sprung mass axes	lb
Fxui Fyui Fzui	FXU(4) FYU(4) FZU(4)	259 260 261	Total tire force components along the vehicle axes	lb
Σ Fxu Σ Fyu	SFXU SFYU	353 354	Resultant forces acting on the vehicle through the unsprung masses in the x and y directions	lb
Σ Fys	SFYS	307	Sprung mass impact force acting along the vehicle y axis	lb
Σ Fzs	SFZS	307	Sprung mass impact force acting along the vehicle z axis	lb
ΣF_{Z1}	SFZ1	356	Resultant force transmitted through the suspensions in the z direction	lb
F_{1Fi} F_{1Ri}	F1FI(2) F1RI(2)	174 184	Front and rear suspension coulomb damping forces for a wheel, effective at the wheel for the front and at the spring for the rear	lb

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO	DEFINITION	UNITS
F_{2Fi} F_{2Ri}	F2FI(2) F2RI(2)	175 185	Front and rear suspension spring and bumper forces for a wheel, effective at the wheel for the front and at the spring for the rear	lb
g	G	I	Acceleration due to gravity	in/sec^2
$GEAR_1$ $GEAR_2$ $GEAR_3$ $GEAR_4$	GEAR1 GEAR2 GEAR3 GEAR4	I	Transmission gear ratios	—
G_{1j}	GN(1,J)	I	Lever arm lengths in brake types 1,2 and 3	in
G_{2j}	GN(2,J)	I	Brake actuation constant, assumed to be equal for both shoes of brake types 1 and 2	
G_{3j}	GN(3,J)	I	Effective lining-to-drum or lining to-disk friction coefficient at design temperature for all shoes or disks in types 1,2 and 4 and for the primary shoe of type 3	
G_{4j}	GN(4,J)	I	Cylinder area for actuation of leading shoe of brake type 1, or for each shoe in types 2 and 3. Also used for total cylinder area per side of disk in type 4	in^2
G_{5j}	GN(5,J)	I	Cylinder area for actuation of trailing shoe of brake Type 1	in^2
G_{6j}-G_{11j}	GN(6,J)- GN(11,J)	I	Brake dimensions for type 3.	in
G_{12j}	GN(12,J)	I	Effective lining to drum friction coefficient for secondary shoe of brake type 3	
G_{13j}	GN(13,J)	I	Mean lining radius for brake type 4	in
G_{14j}	GN(14,J)	I	Coefficient of heat transfer for convective losses	
G_{15j}	GN(15,J)	I	Specific heat of brake assembly	BTU/lb/°F
G_{16j}	GN(16,J)	I	Effective weight of brake assembly for heat absorption	lb
h_i	HI(4)		Tire rolling radius	in
I_{Dj}	FIDJ(2)	I	Driveline inertia for front or rear (Note that a value of zero is entered at the non-driving end of the vehicle)	lb-sec^2-in

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO	DEFINITION	UNITS
I_R	XIR	I	Rear unsprung mass moment of inertia about a line through its center of gravity and parallel to the vehicle x axis	lb-sec^2-in
I_{wj}	FIWJ(4)	I	Rotational inertia of an individual wheel at the front or rear	lb-sec^2-in
Ix,Iy,Iz	XIX XIY XIZ	I	Spring mass moments of inertia about the vehicle axes	lb-sec^2-in
Ixz	XIXZ	I	Spring mass roll-yaw product of inertia	lb-sec^2-in
(I'x)t		47	Effective inertial term due to time varying positions of the unspring masses	
(I'z)t		47	Effective inertial term due to time varying positions of the unspring masses	
(I'xz)t		47	Effective inertial term due to time varying positions of the unspring masses	
(I'yz)t		47	Effective inertia term due to time varying positions of the unspring masses	
I_ψ	XIPS	I	Moment of inertia of the steering system effective at the front wheels (includes both wheels)	lb-sec^2-in
K_d	FKD	I	Performance parameter characterizing understeer/oversteer properties of the vehicle	sec^2/in
K_F,K_R	AKF AKR	I	Front and rear suspension load deflection rate in the quasi-linear range about the design position effective at the front wheels and the rear springs	lb/in
K_{FC},K_{RC}	AKFC AKRC	I	Coefficients for the compression bumpers of the front and rear suspension effective at the front wheels and rear springs	
K'_{FC}, K'_{RC}	AKFCP AKRCP	I	Coefficients for the cubic terms of the suspension compression bumpers	
K_{FE},K_{RE}	AKFF AKRE	I	Coefficients for the extension bumpers of the front and rear suspension effective at the front wheels and rear springs	

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
K'_{FE}, K'_{RE}	AKFEP AKREP	I	Coefficients for the cubic terms of the suspension extension bumpers	
K_P	FKP	328	Driver steer control gain	
K_{RS}	AKRS	I	Rear axle roll-steer coefficient, positive for roll understeer	
K_{S1}, K_{S2}	FKS1 FKS2	I	Drivers estimate of vehicle braking and accelerating gains	
K_{STi}	AKST(3)	I	Structural hard point spring rates	lb/in
K_T	AKT	I	Radial tire rate in the quasi-linear range	lb/in
K_V	AKV	I	Load-deflection characteristic of the vehicle structure	lb/in^3
$K_{\delta S}, K_{\delta S1}, K_{\delta S2}, K_{\delta S3}$	AKDS AKDS1 AKDS2 AKDS3	I	Coefficients of the cubic representation of rear wheel steer as a function of deflection for independent rear suspension	
K_ψ	AKPS	I	Load-deflection rate for the linear steering stop, effective at the wheels	lb-in/rad
K_1	AK1	I	Slope of P_R vs P_F for values of P_F between P_1 and P_2	
K_2	AK2	I	Slope of P_R vs P_F for values of P_F greater than P_2	
$(LF)_i$	FLF	I	Fade coefficient for brake at wheel i	
M_S	XMS	I	Sprung mass	$lbsec^2/in$
M_{UF}, M_{UR}	XMUF XMUF	I	Front (both sides) and rear unsprung masses. Note $M_1=M_2=M_{UF}/2$, $M_3=M_{UR}$	$lbsec^2/in$
M_1, M_2	$\frac{XMUF}{2}$		Right and left front unsprung masses	$lbsec^2/in$
M_3	XMUR	I	Rear unsprung mass	$lbsec^2/in$
NBX	NBX(5)	I	Number of x' boundaries supplied for 5 terrain tables	
NBY	NBY(5)	I	Number of y' boundaries supplied for 5 terrain tables	
NDEL	NDEL		Number of entries in the front wheel camber table	
NX	NX(5)		Number of x' grid points in 5 terrain tables	
NY	NY(5)		Number of y' grid points in 5 terrain tables	
NZTAB	NZTAB	I	Number of terrain tables entered	

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
P_1, P_2	PONE PTWO	I	"Break" pressures for brake system proportioning valve	psig
$(RATIO)_i$			Factor used to modify the nominal tire-terrain friction coefficient at wheel i to reflect the effects of vehicle speed and tire loading	
R_{BB}	RBB	280	Constant for barrier bottom plane	in
R_{Bi}	RBI	269	Constant for barrier face plane	in
R_{BT}	RBT	281	Constant for barrier top plane	in
R_{B_1}	RB1	273	Constant for the plane perpendicular to the barrier face plane and containing the axis of rotation	in
NZ5	NZ5	I	Flag to indicate whether the variable increment terrain table is supplied, =0,no,≠0, yes	
$\Sigma N_{\phi F}$	SNPF	367	Roll moment acting on the front axle	lb-in
$\Sigma N_{\phi R}$	SNPR	360	Roll moment acting on the rear axle	lb-in
$\Sigma N_{\phi S}$	SNPS	308	Roll moment on the sprung mass resulting from sprung mass impact forces	lb-in
$\Sigma N_{\theta S}$	SNTS	309	Pitch moment on the sprung mass resulting from sprung mass impact forces	lb-in
$\Sigma N_{\psi S}$	SNPSS	310	Yaw moment on the sprung mass resulting from sprung mass impact forces	lb-in
$\Sigma N_{\phi U}$ $\Sigma N_{\theta U}$ $\Sigma N_{\psi U}$	SNPU SNTU SNPSU	357 358 359	Moments acting on the sprung mass produced by forces acting on the unsprung masses	lb-in
P,Q,R	P,Q,R	+8	Scalar components of the sprung mass angular velocity along the vehicle x,y and z axes	rad/sec
P_C	PC	I	Hydraulic pressure in brake system master cylinder	psig
P_j	PP(2)	197	Hydraulic pressure in brake cylinders at front or rear brakes	psig
(PS)			Prop shaft speed	rpm

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
(PT)	XPS	I	Pneumatic trail of front tires	in
R_F, R_R	RF,RR	I	Auxiliary roll stiffness of the front and rear suspensions	lb/in/rad
(RPME)	RPME	211	Engine speed	rpm
$(RPS)_i$	RPSI(4)	44	Rotational velocity of wheel i, positive for forward motion of the vehicle	rad/sec
R_{RMi}	RRM(4)	352	Rolling resistance moment acting on wheel i	lb-in
R_W	RW	I	Undeflected tire radius	in
RWHJB RWHJE	RWHJB RWHJE	I	Beginning and ending radii for calculation of the radial tire force-deflection characteristic used in the radial tire mode	in
SET	SET	I	Ratio of permanent deflection to maximum deflection of deformable barrier	
S_i	SI(4)	173 183	Total suspension force for a wheel, acting at the front wheels and rear springs	lb
$(SLIP)_i$	SLIP(I)	241	The amount by which the rotational speed of wheel i is less than that of free rotation expressed as a decimal portion of the speed of free rotation	
$(SLIP)_{pi}$	SLIPP	198	The value of (SLIP)i, at a given wheel center speed U_{Gi} for which the value of μ_{x_i} is a maximum	
SP_n	ST(5,2)	I	Coefficients for straight line segments defining the desired path	
$(S_1)_i$, $(S_2)_i$, $(S_3)_i$	S1I S2I S3I	284 285 286	Characteristic lengths of intersection area between the sprung mass and barrier	in
t	T		Time	sec
T_b	TESTB	I	Braking indifference level	in/sec
T_B, T_E TINCR	TB,TE TINCR	I	Beginning, ending and incremental times for entry of control tables $(TQ)_F$, $(TQ)_R$ and ψ_F	sec
T_F, T_R	TF,TR	I	Front and rear track	in
T_i	TI(4)	225	Circumferential tire force resulting from applied torque	lb
T_I, T_L	TIL TL	I	Driver steering model lag and lead times	sec
$(TQ)_{Bi}$	TQB(4)	204	Brake torque at wheel i	lb-ft
$(TQ)_{Dj}$	TQD(4)	211	Drive line torque at prop shaft at vehicle end j	lb-ft

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
$(TQ)_E$	TQE	210	Engine torque	lb-ft
$(\overline{TQ})_F (\overline{TQ})_R$	TQF(50) TQR(50)	I	Front and rear torque tables for a single wheel and effective at the wheel (positive for traction, negative for braking)	lb-ft
(TR)	TTTR	I	Transmission ratio (speed ratio of engine to prop shaft)	
T_{R_1}, T_{R_2}	TESTR1 TESTR2	I	Lower and upper skid thresholds	
T_S	TS	I	Distance between spring mounts for a solid rear axle	in
T_{SF}	TSF	I	Distance between spring mounts for a solid front axle	in
(TS)	TTTS	I	Throttle setting expressed as the decimal portion of wide open throttle	
T_{S1}, T_{S2}	TESTS1 TESTS2	I	Driver threshold/indifference levels for positive and negative speed errors	in/sec
(TYPE)	NBTYPE	I	Brake type indicator	
$T_{1\psi}$	T1PSI	36	Coulomb friction torque in steering system effective at the wheel	lb-in
$T_{2\psi}$	T2PSI	36	Resistance torque produced by the front wheel steer stops, effective at the wheel	lb-in
u,v,w	U,V,W	48	Scalar components of linear velocity of the sprung mass along the sprung mass x,y and z axes	in/sec
u',v',w'	DXCP DYCP DZCP	64	Scalar components of linear velocity of the sprung mass along the space fixed x',y' and z' axes	in/sec
u_i, v_i, w_i	UI(4) VI(4) WI(4)	90-98	Scalar components of the tire contact points linear velocity along the vehicle axes	in/sec
U_{Gi}	UG(4)	103	Wheel center forward velocity in direction parallel to the tire-terrain contact plane	in/sec
U_{Gwi}	UGW(4)	195	Ground contact point velocity along the circumferential direction of the wheel	in/sec
u'_n v'_n w'_n	UNP(17) VNP(17) WNP(17)	282	Components of the velocity of the three or four points that define the intersection area of the barrier and vehicle along the space-fixed axes	in/sec

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
u'_r v'_r w'_r	URP VRP WRP	303	Components of the velocity of the point of application of the sprung mass impact force along the space-fixed axes	in/sec
U'_{STi} V'_{STi} W'_{STi}	UPT(4) VPT(4) WPT(4)	299	Components of the velocity of the deformed structural hard points along the space fixed axes	in/sec
U_T	UT	313	Total vehicle velocity	in/sec
v_{Gi}	VG(4)	106	Contact point lateral velocity in the direction parallel to the tire-terrain contact plane	in/sec
VGR_{12} VGR_{21} VGR_{23} VGR_{32} VGR_{34} VGR_{43}	VGR12 VGR21 VGR23 VGR32 VGR34 VGR43	I	Vehicle speed at which transmission upshifts and downshifts occur	mph
(VTAN)	VTAN	305	Tangential velocity between the vehicle and barrier	in/sec
WE_i	WEIGHT(I)	228	Driver steering error weighting function	
WI_i	XIMPOR(I)	I	Driver steering error importance weighting function	
(WOT)	TWOT		Wide open throttle torque	lb-ft
X_B, X_E XINCR	XB(5) XE(5) XINCR(5)	I	Beginning, ending and incremental x' for terrain tables	in
x_{BB}, y_{BB} z_{BB}	XBB YBB ZBB	279	Coordinates of the intersection of the z' axis with the barrier bottom plane in the vehicle axes	in
X_{BDRY}	XBDRY(4,5)	I	x' intercept for angled boundaries within terrain tables	in
x_{Bi} y_{Bi} z_{Bi}	XBI YBI ZBI	267	Coordinates of the intersection of the y' axis with cutting plane i, in the vehicle axes	in
x_{BT} y_{BT} z_{BT}	XBT YBT ZBT	278	Coordinates of the intersection with the barrier top plane in the vehicle axes	in
X'_c, Y'_c, Z'_c	XCP YCP ZCP	65 66 67	Coordinates of the origin of the vehicle axes (sprung mass center of gravity) with respect to the space fixed axes	in
x_{cpn} y_{cpn} z_{cpn}	XCPN(3) YCPN(3) ZCPN(3)	214	Coordinates of the vehicle corner n in the vehicle axes	in
x'_{cpn} y'_{cpn} z'_{cpn}	XCPNP(3) YCPNP(3) ZCPNP(3)	214	Coordinates of the vehicle corner n in the space-fixed axes	in
$X'_{GPi}, Y'_{GPi}, Z'_{GPi}$	XGPP(4) YGPP(4) ZGPP(4)	150 151	Coordinates of the ground contact points with respect to the space-fixed axes	in
x'_i, y'_i, z'_i	XP(4) YP(4) ZP(4)	68-82	Coordinates of the wheel centers with respect to the space fixed axes	in

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
x_n y_n z_n	XNN(17) YNN(17) ZNN(17)	276	Coordinates of intercept points between the barrier and sprung mass in the vehicle axes	in
X'_{Pi} y'_{Pi}	X Y	318	Coordinates of the location on the desired path at which the ith error is determined	in
x_{Ri} y_{Ri} z_{Ri}	XRI YRI ZRI	294	Coordinates of the centroid of the intersection area on cutting plane i, projected on to the actual vehicle barrier interface of the previous time increment	in
$(\Sigma x_R)_t$ $(\Sigma y_R)_t$ $(\Sigma z_R)_t$	SXR SYR SZR	295 296 297	Coordinates of the point of application of the sprung mass impact force	in
x_{STi} y_{STi} z_{STi}	XSTI(3) YSTI(3) ZSTI(3)	301	Coordinates of the deformed structural hard points in the vehicle axes	in
x_{STi_o} y_{STi_o} z_{STi_o}	XSTIO(3) YSTIO(3) ZSTIO(3)	I	Coordinates of the underformed structural hard points in the vehicle axes	in
x_{VF}	XVF	I	Distance from the sprung mass c.g. to the vehicle front along the x axis	in
x_{VR}	XVR	I	Distance from the sprung mass c.g. to the vehicle rear along the x axis	
X'_{VP_i} y'_{VP_i}	XVP YVP	312 313	Driver prediction of vehicle location at the ith sample increment in the future	in
x_1, y_1, z_1 x_2, y_2, z_2	X1,Y1,Z1 X2,Y2,Z2	I	Coordinates of accelerometer positions with respect to the vehicle axes for which acceleration components are output	in
$\{y\}$	VAR		System dependent variable, integral of $\{\dot{y}\}$	
$\{\dot{y}\}$	DER	47	First derivatives with respect to time of the system dependent variables	
Y_B Y_E YINCR	YB(5) YE(5) YINCR(5)	I	Beginning, ending and incremental y' for terrain tables	in
Y_{BDRY}	YBDRY(4,5)	I	Lateral position of y' terrain boundaries with respect to space	in
y'_B	YBP	I	Lateral position of the barrier face plane with respect to space	in
y'_{C1}, y'_{C2} y'_{C3}, y'_{C4} y'_{C5}, y'_{C6}	YC1P YC2P YC3P YC4P YC5P YC6P	I	Lateral positions of slope changes defining a curb	in
y_v	YV	I	Distance from the sprung mass c.g. to the vehicle side	in
z'_{BB}	ZBBP	I	Elevation of the bottom barrier plane in space	in

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
Z'_{BT}	ZBTP	I	Elevation of the top barrier plane in space	in
Z'_{C2}, Z'_{C3}	ZC2P	I	Elevation of curb at slope$_{C2}$	in
Z'_{C4}, Z'_{C5} Z'_{C6}	ZC3P ZC4P ZC5P ZC6P	I	Change lateral positions	
Z_F	ZF	I	Static distance along z axis between the sprung mass center of gravity and the center of gravity of the front unsprung masses	in
$\bar{Z}'_G$	ZGP(21,21,5)	I	Input elevations of the terrain table grid points	in
Z'_{Gi}	ZPGI(4)	126	Ground elevation with respect to the space axes of the point beneath the wheel centers	in
Z'_{Gi}			A vector through the ground contact point normal to the actual or equivalent ground contact plane	
Z_R	ZR	I	Static distance along the z axis between the sprung mass center of gravity and the rear axle roll center	in
Z_{VB}	ZBV	I	Distance from the sprung mass c.g. to the plane defining the bottom of the vehicle along the z axis	in
Z_{VT}	ZVT	I	Distance from the sprung mass c.g. to the plane defining the top of the vehicle, along the z axis	in
$\alpha_B, \beta_B, \gamma_B$		266	Direction angles of a normal to the barrier face plane in the vehicle axes	
$\alpha_{BT}, \beta_{BT}, \gamma_{BT}$		277	Direction angles of a normal to the barrier top plane in the vehicle axes	
$\alpha_{B1}, \beta_{B1}, \gamma_{B1}$		273	Direction angles of a normal to the plane perpendicular to the barrier face plane and containing the axis of rotation	

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
$\alpha_{ci}, \beta_{ci}, \gamma_{ci}$		255	Direction angles of a line perpendicular to the normals of both the wheel plane and tire-terrain contact plane with respect to space	rad
$\alpha_{GZ_i}, \beta_{GZ_i}, \gamma_{GZ_i}$		84	Direction angles of a normal to the tire-terrain contact plane at wheel i with respect to space	rad
$\alpha_{ki}, \beta_{ki}, \gamma_{ki}$		116	Direction angles of the resultant radial force on wheel i with respect to the vehicle axes	rad
$\alpha_j, \beta_j, \gamma_j$		143	Direction angles of a line from wheel center i to the ground contact point of tire radial spring j with respect to the vehicle axes	
$\alpha_{Ri}, \beta_{Ri}, \gamma_{Ri}$		148	Direction angles of the resultant radial force on wheel i with respect to the space axes	rad
$\alpha_{s_i}, \beta_{s_i}, \gamma_{s_i}$		257	Direction angles of a line perpendicular to both a normal to the tire-terrain contact plane and the wheel axis with respect to space	rad
$\alpha_x, \beta_x, \gamma_x$		102	Direction angles of the x axis with respect to space	
$\alpha_y, \beta_y, \gamma_y$		100	Direction angles of the y axis with respect to space	
$\alpha_{yw_i}, \beta_{yw_i}, \gamma_{yw_i}$		85	Direction angles of a normal to the wheel i with respect to space	
$\alpha_{w_i}, \beta_{w_i}, \gamma_{w_i}$		88	Direction angles of kingpin axis of wheel i	
β_i			Slip angle at wheel i	rad
β'_i	BETP(4)	219	Equivalent slip angle produced by camber of wheel i	rad
$\bar{\beta}_i$	BETBR(4)	223	Non dimensional slip angle variable for wheel i	
γ_1	GAM1	47	Inertial expressions	
$(\gamma_2)_t$	GAM2			
$(\gamma_3)_t$	GAM3			

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
$(\gamma_4)_t$	GAM4		Inertial expressions	
$(\gamma_5)_t$	GAM5	47		
$(\gamma_6)_t$	GAM6			
$(\gamma_7)_t$	GAM7			
$(\gamma_8)_t$	GAM8			
$(\gamma_9)_t$	GAM9			
	DELBB		Barrier deflection	in
δ_1	DEL1		Right front suspension deflection for independent front suspension or front axle roll center deflection relative to the vehicle from position of statis equilibrium	in
δ_2	DEL2		Left front suspension deflection relative to the vehicle from static equilibrium position	in
δ_3	DEL3		Right rear suspension deflection for independent rear suspension or rear axle roll center deflection relative to the vehicle from static equilibrium position	in
δ_4	DEL4		Left rear suspension deflection relative to the vehicle from static equilibrium position	in
Δ_G	DELG	I	Distance between road roughness input points	in
Δi	DELTA(4)	112	Distance between the wheel center and ground contact point	in
Δt	DT	I	Numerical integration step interval	sec
Δt_B	DELTB	I	Time increment for use during barrier impacts	sec
Δt	DELTC	I	Numerical integration step size for curb impact option	sec
Δt_n	DTR		Integration step size for use with wheel spin equations of motion	sec
ΔT_{HFi}	DTHF1 DTHF2	I	Front and rear half-track changes with suspension deflection	in
ΔT_{HRi}	DTHR3 DTHR4			
$\Delta y'_B$	DELYBP	I	Incremental deflection of the barrier position	in
$\Delta\psi_{Fj}$	DPSILF	328	Ideal steer angle change	rad
ϵ_B	EPSB	I	Acceptable error in the force balance between the vehicle structure and barrier	lbs

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
ϵ_F, ϵ_R	EPSF EPSR	I	Friction lag in front and rear suspensions	in/sec
ϵ_n	EPSL		Permanent set of the barrier for secondary impacts	in
ϵ_V	EPSV	I	Friction lag in the vehicle-barrier friction force	in/sec
ϵ_ψ	EPSPS	I	Friction lag in steering system	deg/sec
ζ_B	ZETAB	I	Threshold value of wheel rotational velocity below which logic is applied to limit brake torques	rad/sec
ζ_i		171	Suspension displacement of the relative to the vehicle from the position of static equilibrium	in
$(\zeta_0)_n, (\zeta_1)_n, (\zeta_2)_n$	CDD CD1 CD2		Coefficients for unloading force deflection characteristic of the barrier	
θ_C	THESKD	336	Vehicle slip angle	rad
θ_{G_i}	THGI(4)	124	Pitch angle of terrain under wheel i relative to the space axes	rad
θ'_n	THETN		Value of θ at t=D or at the nth indexing of the axes	rad
ϕ'_t	THETT	57	integrated value of $\dot\theta$ from t=D or the nth indexing of the axes	
θ'_{xG_i}		101	Angle between the x axis and the tire-terrain contact plane at wheel i	rad
λ_B	TLAMB	204	Coefficient for inertial coupling terms in relationships for driving end of vehicle	
$\lambda_F\ \lambda_R$	XLAMF XLAMR	I	Ratio of conserved to absorbed energy in the front and rear suspension bumpers or multiple of K_F, K_R for use in simulating suspension bumpers	
λ_T	XLAMT	I	Multiple of K_T for use in non-linear range of tire deflection	
$\lambda_{1i}\ \lambda_{2i}\ \lambda_{3i}$	XLM1(4) XLM2(4) XLM3(4)	107 108 109	Contants for simultaneous solution of the ground contact point	
μ_B	AMUB	I	Effective coefficient of friction between the vehicle sprung mass and barrier	
μ_C	AMUC	I	Tire-curb friction factor	

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
μ_{G_i}	XMUGI(4)	I	Nominal coefficient of friction between tire i and ground	
μ_i	XMUI(4)	I	Peak value of friction coefficient for side forces for prevailing conditions of speed and load at wheel i	
μ_{m_i}	XMUM(4)	I	Nominal test surface friction coefficient on which tire properties were measured	
μ_{x_i}	XMUX(4)	240	Effective friction coefficient between tire and terrain at wheel i in the direction along the tire circumference	
μ_{xp_i}	XMUXP(4)	I	Peak circumferential friction coefficient for tire i	
μ_{xs_i}	XMUXS(4)	I	Sliding circumferential friction coefficient for tire i	
π	PI		3.14159...	
ρ	RHO	I	Distance between rear axle center of gravity and roll center, positive for roll center above c.g.	in
ρ_F	RHOF	I	Distance between front axle center of gravity and roll center, positive for roll center above c.g.	in
ρ_{s_i}	RHOS(I)	198	Ratio of circumferential to peak side force friction coefficients for prevailing conditions of speed and load	
$(\rho_{s_i})_{max}$	RHOMAX	198	Maximum value of si at the existing forward velocity of wheel i	
σ_R	SIGR	I	Coefficients for the polynominal form of barrier load deflection characteristic	
σ_T	SIGT	I	Maximum radial tire deflection for quasi-linear load-deflection characteristic	in
τ_A	TAUA	I	Ambient temperature	°F
τ_i	TAU(4)		Temperature of brake assembly	°F
τ_{i_o}	TAUO(4)	I	Initial temperature of brake assembly	°F
ϕ,θ,ψ	PHIT THETT PSIT		Euler angles of sprung mass axes relative to inertial axes	rad
ϕ_C ϕ_{C_R}	PHIC(50) PHIRC(50)	I	Table of front and rear wheel camber as a function of deflection	deg
ϕ_{CG_i}	PHICI(4)	86	Camber angles of wheels relative to the normal to tire-terrain contact plane	rad
ϕ_{c_1},ϕ_{c_2} ϕ_{c_3},ϕ_{c_4} ϕ_{c_5},ϕ_{c_6}	PHIC1, PHIC2 PHIC3, PHIC4 PHIC5, PHIC6	I	Curb slope angles	rad

ANALYTICAL SYMBOL	PROGRAM SYMBOL	EQN NO.	DEFINITION	UNITS
ϕ_F	PHIF		Angular displacement of front axle relative to the vehicle about a line parallel to the x-axis through the front roll center	rad
ϕ_{G_i}	PHGI(4)	125	Camber angle of terrain under wheel i	rad
ϕ_i	PHII(4)		Camber angles of four wheels relative to vehicle	rad
ϕ'_n	PHIN		Value of at t=0 or at the nth indexing of the axes	rad
ϕ_R	PHIR		Angular displacement of rear axle relative to the vehicle about a line parallel to the x - axis through the rear roll center	rad
ϕ'_T	PHIT	58	Integrated value of $\dot{\varphi}$ from t=0 or the nth axis indexing	rad
ϕ_{yG_i}		105	Angle between y axis and tire-terrain contact plane	rad
ψ_{BDRY}	PSBDRY(4,5)	I	Angle of interpolation boundaries in terrain tables, measured from the x'axis	rad
ψ_f	PSIF(50)	I	Table of front wheel steer angle vs time	rad
ψ_i	PSII(4)		Steer angles of wheels relative to vehicle (positive-clockwise as viewed from above)	rad
ψ'_i	PSIIP(4)	89	Steer angles of wheels in tire-terrain contact plane	rad
ψ'_n	PSIN		Value of ψ at t=0 or the nth indexing of the axes	rad
ψ'_t	PSIT	59	Integrated value of $\dot{\psi}$ from t=0 or the nth axis indexing	rad
Ω_F Ω_R	OMEGF OMEGR	I	Maximum suspension deflections from the equilibrium position for linear load-deflection characteristic of the springs	in
Ω_{FC} Ω_{RC}	OMEGFC OMEGRC	I	Front and rear suspension deflections at which the compression bumpers are contacted, measured at the front wheels and the rear springs	in
Ω_{FE} Ω_{RE}	OMEGFE OMEGRE	I	Front and rear suspension deflections at which the extension bumpers are contacted, measured at the front wheels and rear springs	in
Ω_T	OMEGT	I	Multiple of A_2 at which the assumed parabolic variations of small angle cornering and camber stiffnesses with tire loading are abandoned	
Ω	OMGPS	I	Front wheel steering angle at which the linear steering stops are engaged	rad

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
A	U	U	INPT	a
AAI	U		BARIER	AA_I
AA2	U		BARIER	AA_2
AAA	U	U	INPT	α
AAR		A	INPT4	NOT USED
ABSUGW		U	COMP5	$/U_{GW_i}/$
AE		U	DRIVE	$\vert e_i \vert$
AINTI	U		BARIER	$(A_{INT})_i$
AINTP	U		BARIER	$(A_{INT})_{t-1}$
AJMTX	U	U	COMPN	$[A_j]$
AKDS	U	U	INSUS	$K_{\delta s}$
AKDS1	U	U	INSUS	$K_{\delta sI}$
AKDS2	U	U	INSUS	$K_{\delta s2}$
AKDS3	U	U	INSUS	$K_{\delta s3}$
AKF	U	U	INPT	K_F
AKFC	U	U	INPT3	K_{FC}
AKFCP	U	U	INPT3	K'_{FC}
AKFE	U	U	INPT3	K_{FE}
AKFEP	U	U	INPT3	K'_{FE}
AKPS	U	U	INPT1	K_ψ
AKR	U	U	INPT	K_R
AKRC	U	U	INPT3	K_{RC}
AKRCP	U	U	INPT3	K'_{RC}
AKRE	U	U	INPT3	K_{RE}
AKREP	U	U	INPT3	K'_{RE}
AKRS	U	U	INPT	K_{RS}
AKST	U		BARSTR	K_{ST}
AKT	U	U	TIRIN	K_T
AKV	U		INPT2	K_V
AK1		U	INPT5	K_1

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
AK2		U	INPT5	K_2
AMTX	U	U	DIMV	$[A]$
AMU	U	U	TIRIN	μ
AMUB	U		INPT2	μ_B
AMUC	U	U	INPT1	μ_C
AMUCMP	A	A	COMPN	NOT USED
AMUF	U	U	COMP	aM_{UF}
AMUG	U	U	INPT	AMUG(n)
ANGI	A	A	COMP	NOT USED
ANG2	A	A	COMP	NOT USED
A02APB	U	U	COMP	$aM_s g/2(a+b)$
APB		U	DRIVE	$a + b$
APD		U	DRIVE	APD
APDMAX		U	DRIVI	APDMAX
APSI		U	DRIVE	
APSIM		U	DRIVE	
APF	U	U		$\overline{AP}_F$
APFR	U	U	APTABL	$\overline{AP}_F, \overline{AP}_R$
APITCH	U	U		F_{APi}
APR	U	U		$\overline{AP}_R$
APTCH1	U	U	ADTNL	F_{AP1}
APTCH2	U	U	ADTNL	F_{AP2}
APTCH3	U	U	ADTNL	F_{AP3}
APTCH4	U	U	AFTNL	F_{AP4}
ARBR		U	INPT4	$\overline{AR}_F, \overline{AR}_R$
ARBRF		U		$\overline{AR}_F$
ARBRI		U	COMP4	$\overline{AR}_F, \overline{AR}_R$
ARBRR		U		$\overline{AR}_R$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
ARCAPE		U	DRIVE	$WE_i WI_i e_i$
AREI		U	DRIVE	e_i
ARFAC1		U	COMP4	$\frac{6\ \overline{AR}_j I_{wj}}{(I_{wj} + 1/4\ I_{Dj}\overline{AR}_j^{\ 2})^2 + (1/4\ I_{Dj}\overline{AR}_j^{\ 2})^2}$
ARFAC2		U	COMP4	$6\ \overline{AR}_j/(I_{wj} + 1/4(I_{Dj}\overline{AR}_j^{\ 2}))$
ARFAC3		U	COMP5	$12\ I_{wj}$
ARTQ6		A	COMP4	NOT USED
AS	U	U	DIMV	a_{s_i}
AX	U	U	DIMV	a_{x_i}
AXP		U	DRIVE	a'_x
AXMF02	U	U	COMP	$aM_{UF}/2$
AY	U	U	DIMV	a_{y_i}
AYP		U	DRIVE	a'_y
AO	U	U	TIRIN	A_o
A1	U	U	TIRIN	A_1
A12	U	U	TIRIN	A_1/A_2
A2	U	U	TIRIN	A_2
A23	U	U	TIRIN	A_2A_3/A_1
A234	U	U	TIRIN	A_2A_3/A_4
A3	U	U	TIRIN	A_3
A4	U	U	TIRIN	A_4
B	U	U	INPT	b
BB1	U		BARIER	BB_1
BB2	U		BARIER	BB_2
BET	U	U	INPI	β
BETBR	U	U	DIMV	$\overline{\beta}_i$
BETP	U	U	DIMV	β'_i
BETR		A	INPT4	NOT USED
BFP1		U	DRIVI	B_{FP1}
BFP2		U	DRIVI	B_{FP2}
BMTX	U	U	COMPN	[B]
BMUR	U	U	COMP	bM_{UR}
BNMTX	U	U	EINDEX	$[B_n]$
BO2APB	U	U	COMP	$bM_s g/2(a+b)$
BROMUR	U	U	COMP	ρbM_{UR}
BRPM		U	INPT5	
BS	U	U	DIMV	b_{s_i}
BTLF		U	INPT5	
BTT		U	INPT5	
BX	U	U	DIMV	b_{x_i}
BXMRØ2	U	U	SUSCMP	$bM_{UR}/2$
BY	U	U	DIMV	b_{y_i}
CAB	U		BARIER	$\cos\alpha_B$
CABT	U		BARIER	$\cos\alpha_{BT}$
CAB1	U		BARIER	$\cos\alpha_{B_1}$
CAC	U	U	DIMV	$\cos\alpha_{C_i}$
CAGZ	U	U	DIMV	$\cos\alpha_{GZ'_i}$
CAH	U	U	DIMV	$\cos\alpha_{h_i}$
CAR	U	U	DIMV	$\cos\alpha_{R_i}$
CAS	U	U	DIMV	$\cos\alpha_{S_i}$
CAX	U	U	COMP	$\cos\alpha_x$
CAXW	A	A	DIMV	NOT USED

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
CAY	U	U	COMP	$\cos\alpha_Y$
CAYW	U	U	DIMV	$\cos\alpha_{YW_i}$
CBB	U		BARIER	$\cos\beta_B$
CBBT	U		BARIER	$\cos\beta_{BT}$
CBB1	U		BARIER	$\cos\beta_{B1}$
CBC	U	U	DIMV	$\cos\beta_{C_i}$
CBGZ	U	U	DIMV	$\cos\beta_{GZ'_i}$
CBH	U	U	DIMV	$\cos\beta_{h_i}$
CBR	U	U	DIMV	$\cos\beta_{R_i}$
CBS	U	U	DIMV	$\cos\beta_{S_i}$
CBX	U	U	COMP	$\cos\beta_X$
CBXW	A	A	DIMV	NOT USED
CBY	U	U	COMP	$\cos\beta_Y$
CBYW	U	U	DIMV	$\cos\beta_{YW_i}$
CC1	U		BARIER	CC_1
CC2	U		BARIER	CC_2
CF	U	U	INPT	C_F
CFP	U	U	INPT	C'_F
CGB	U		BARIER	$\cos\gamma_B$
CGBT	U		BARIER	$\cos\gamma_{BT}$
CGB1	U		BARIER	$\cos\gamma_{BI}$
CGC	U	U	DIMV	$\cos\gamma_{C_i}$
CGGZ	U	U	DIMV	$\cos\gamma_{GZ'_i}$
CGH	U	U	DIMV	$\cos\gamma_{h_j}$
CGR	U	U	DIMV	$\cos\gamma_{R_i}$
CGS	U	U	DIMV	$\cos\gamma_{S_i}$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
CGX	U	U	COMP	$\cos\gamma_X$
CGXW	A	A	DIMV	NOT USED
CGY	U	U	COMP	$\cos\gamma_Y$
CGYW	U	U	DIMV	$\cos\gamma_{YW_1}$
CHED	U	U	HEAD	CONTROL TITLE
CMTX	U	U	DIMV	$[C_j]$
CNMTX	U	U	EINDEX	$[C_n]$
COMEN4		U	INPT4	CØMEN4
COMEN5		A	COMP5	NOT USED
CONE		U	INPT5	C_1
CONMPH		U	DRIVE	3600/(12x5280)
CONS	U	A	INPT2	$\overline{\text{CONS}}$
COSPH	U	U	COMP	$\cos\phi$
COSPHN	U	U	EINDEX	$\cos\phi_n$
COSPS	U	U	COMP	$\cos\psi$
COSPSN	U	U	EINDEX	$\cos\psi_n$
COSTH	U	U	COMP	$\cos\theta$
COSTHN	U	U	EINDEX	$\cos\theta_n$
CPG	U	U	DIMV	$\cos\phi_{G_i}$
CPHI	A	A	COMP	NOT USED
CPHIC	U	A	ADTNL	$\cos\phi_{C_i}$
CPHICI		U	COMP4	$\cos\phi_{C_i}$
CPHTP	U	U	COMP	$\cos\phi_t$
CPSI	A	A	COMP	NOT USED
CPSP	U	U	INPTI	C'_ψ
CPSTP	U	U	EINDEX	$\cos\psi'_t$
CPYG	U	U	DIMV	$\cos\phi_{y_{Gi}}$
CR	U	U	INPT	C_R
CRP	U	U	INPT	C_R'

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
CS	U	U	DIMV	C_{Si}
CT		U	INPT4	C_{Ti}
CTG	U	U	DIMV	$\cos \emptyset_{G_i}$
CTHETP	U	U	EINDEX	$\cos \Theta_t$
CTHREE		U	INPT5	C_3
CTWO		U	INPT5	C_2
CTXG	U	U	DIMV	$\cos\Theta_{XGi}$
CX	U	U	DIMV	C_{x_i}
CY	U	U	DIMV	C_{y_i}
DADE	U	U	INPT	DATE ARRAY
DAPFB	U	U	APTABL	
DADFE	U	U	APTBL	
DAPRB	U	U	APTABL	
DAPRE	U	U	APTABL	
DDAPF	U	U	APTABL	
DDAPR	U	U	APTABL	
DDD	A		INPT2	NOT USED
DDEL	U	U	INPT	DDEL
DDEL1	U	U		$\dot{\delta}_1$
DDEL1D	U	U		$\ddot{\delta}_1$
DDEL2	U	U		$\dot{\delta}_2$
DDEL2D	U	U		$\ddot{\delta}_2$
DDEL3	U	U		$\dot{\delta}_3$
DDEL3D	U	U		$\ddot{\delta}_3$
DDEL4	U	U		$\dot{\delta}_4$
DDEL4D	U	U		$\ddot{\delta}_4$
DDPSFI	U	U		$\ddot{\psi}_F$
DD1M2	U	U	COMP	$\dot{\delta}_1-\dot{\delta}_2$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
DD1P2	U	U	COMP	$\dot{\delta}_1+\dot{\delta}_2$
DD3M4	U	U	SUSCMP	$\dot{\delta}_3-\dot{\delta}_4$
DD3P4	U	U	SUSCMP	$\dot{\delta}_3+\dot{\delta}_4$
DELB	U	U	INPT	DELB
DELBB	U		BARIER	δ_B
DELBBP		U	BARIER	$(\delta_B)_{t-1}$
DELE	U	U	INPT	DELE
DELG	U	U	RUFNES	ΔG
DELPTH	U	U	DRFVTT	EMDT
DELTA	U	U	DIMV	Δ_i
DELTAE		U	COMP5	ΔE_i
DELTAX		U	DRIVE	D_{ax}
DELTB	U		INPT2	Δt_B
DELTC	U	U	INPT1	Δt_c
DELTY		U	DRIVE	
DELX	U		BARIER	
DELYBP	U		INPT2	$\Delta y'_B$
DEL1	U	U		δ_1
DEL1D	U	U		$\dot{\delta}_1$
DEL10	U	U	INPT	δ_{1o}
DEL10D	U	U	INPT	$\dot{\delta}_{1o}$
DEL2	U	U		δ_2
DEL2D	U	U		$\dot{\delta}_2$
DEL20	U	U	INPT	δ_{2o}
DEL20D	U	U	INPT	$\dot{\delta}_{2o}$
DEL3	U	U		δ_3
DEL3D	U	U		$\dot{\delta}_3$
DEL30	U	U	INPT	δ_{3o}
DEL30D	U	U	INPT	$\dot{\delta}_{3o}$
DEL4	U	U		δ_4
DEL4D	U	U		$\dot{\delta}_4$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
DEL40	U	U	INSUS	δ_{40}
DEL40D	U	U	INSUS	$\dot{\delta}_{40}$
DEND		U	DRIVE	
DER	U	U	INTG	$\dot{y}$
DERR		A	INTR	NOT USED
DESS		U	DRIVE	DS
DESSI		U	DRIVI	
DGMAX	U	U	RUFNES	
DI		U	DRIVE	
DISS	U		BARIER	
DIST		U	DRIVE	DIST
DISTC		U	DRIVE	
DISTD	U	U	COMP	$\sqrt{D_{1_i}^2 + D_{2_i}^2 + D_{3_i}^2}$
DISTI		U	DRIVI	
DISTS	U	U	COMP	$\sqrt{a_{s_i}^2 + b_{s_i}^2 + c_{s_i}^2}$
DISTX	U	U	COMP	$\sqrt{a_{x_i}^2 + b_{x_i}^2 + c_{x_i}^2}$
DISTY	U	U	COMP	$\sqrt{a_{y_i}^2 + b_{y_i}^2 + c_{y_i}^2}$
DMATX	U	U	DIMV	[D] and [E]
DP	U	U		$\dot{P}$
DPHIF	U	U		$\dot{\phi}_F$
DPHIFD	U	U		$\ddot{\phi}_F$
DPHIR	U	U		$\dot{\phi}_R$
DPHIRD	U	U		$\ddot{\phi}_R$
DPHITP	U	U		$\dot{\phi}'_t$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
DPSIFI	U	U		$\dot{\psi}_F$
DPSILF		U	DRIVE	$\Delta\psi_{Fj}(t)$
DPSINT	A	A	COMPN	NOT USED
DPSISF		U	DRIVE	$\Delta\psi_{fi}$
DPSITP	U	U		$\ddot{\psi}'_t$
DQ	U	U		$\dot{Q}$
DR	U	U		$\dot{R}$
DRIEND		A	DRIVI	NOT USED
DRPSI		U		$\frac{d}{dt}(RPS)_i$
DRWHJ	U	U	INPT1	DRWHJ
DS		U	DRIVI	ΔS
DSØES		U	DRIVE	
DT	U	U	INTG	Δt
DTCMP1	A	A	INPT	NOT USED
DTCOMP	U	U	INPT	Δt
DTDD1	U	U	SUSCMP	$d(\Delta T_{HF1})/d\delta_1$
DTDD2	U	U	SUSCMP	$d(\Delta T_{HF2})/d\delta_2$
DTDD3	U	U	SUSCMP	$d(\Delta T_{HR3})/d\delta_3$
DTDD4	U	U	SUSCMP	$d(\Delta T_{HR4})/d\delta_4$
DTHF	U	U	INSUS	
DTHF1	U	U	SUSCMP	ΔT_{HF1}
DTHF2	U	U	SUSCMP	ΔT_{HF2}
DTHR	U	U	INSUS	
DTHR3	U	U	SUSCMP	ΔT_{HR3}
DTHR4	U	U	SUSCMP	ΔT_{HR4}
DTHTTP	U	U		$\dot{\Theta}'_t$
DTINT		U	COMP4	$(RPS)_i / \frac{d(RPS_i)}{dt}$
DTLF		U	INPT5	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
DTPRNT	U	U	INPT	
DTR		U	INTR	Δt_n
DTSTEP		U	COMP4	
DTT		U	INPT5	
DTTEST		U	COMP4	$(RPS)/\frac{d(RPS)_i}{dt}$
DU	U	U		$\dot{u}$
DV	U	U		$\dot{v}$
DW	U	U		$\dot{w}$
DXCP	U	U		$\dot{u}'$
DYCP	U	U		$\dot{v}'$
DZCP	U	U		$\dot{w}'$
D1	U	U	DIMV	D_{1i}
D1MD2	U	U	COMP	$\delta_1-\delta_2$
D1PD2	U	U	COMP	$\delta_1+\delta_2$
D2	U	U	DIMV	D_{2i}
D21	U	U	COMP	$\delta_2-\delta_1$
D3	U	U	DIMV	D_{3i}
D3MD4	U	U	SUSCMP	$\delta_3-\delta_4$
D3PD4	U	U	SUSCMP	$\delta_3+\delta_4$
D43	U	U	SUSCMP	$\delta_4-\delta_3$
EBAR	U	U	INPT	$\overline{E}$
EEE	U		BARIER	$(E_1)_t$
EI		U	DRIVE	
EPSL	U		BARIER	ϵ_{n-1}
EM	U	U	INPT	$\overline{M}$
EN		U	DRIVE	EN
ENDEIN	A	A	EINDEX	NOT USED
EMDT		U	DRIVI	EMDT
END3	A	A	INPT3	NOT USED

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
ENRGY	U		BARIER	ΣE
				t
EPSB	U		INPT2	ϵ_B
EPSE	U	U	INPT	ϵ_F
EPSL	A		BARIER	
EPSPS	U	U	INPT1	ϵ_ψ
EPSR	U	U	INPT	ϵ_R
EPSS		U	COMP4	ϵ_{s_i}
EPSSFC		U	COMP4	$\epsilon_{Si} F_{Ci}$
EPST	A		INPT2	NOT USED
EPSV	U		INPT2	ϵ_V
ERPM		U	INPT5	
ES		U	DRIVE	
ET		U	DRIVE	$\Sigma WE_i WI_i e_i$
ETLF		U	INPT5	
ETT		U	INPT5	
EWT		U	DRIVE	$WE_i WI_i e_i$
FP	U		BARIER	F_B
FBRK		U	DRIVE	F_{BRK}
FC	U	U	DIMV	F_{Ci}
FCAV		U	COMP4	$(\Sigma F_{C_j} \Delta t_n)/\Delta t$
FCSLSM		U	COMP4	
FCXFAC		U	COMP4	$A_{11}\cos\alpha_{C_i}+A_{21}\cos\beta_{C_i}+A_{31}\text{cis}\,\gamma_{C_i}$
FCXU	U	U	DIMV	F_{CXU_i}
FCYFAC		U	COMP4	$A_{12}\cos\alpha_{C_i}+A_{22}\cos\beta_{C_i}+A_{32}\cos\gamma_{C_i}$
FCYU	U	U	DIMV	F_{cyu_i}
FCZFAC		U	COMP4	$A_{13}\cos\alpha_{C_i}+A_{23}\cos\beta_{C_i}+A_{33}\cos\gamma_{C_i}$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
FCZU	U	U	DIMV	$FCZU_i$
FIDAR		A	COMP4	NOT USED
FIDIW		A	COMP4	NOT USED
FIDJ		U	INPT4	I_{DF}, I_{DR}
FIDJF		U		I_{DF}
FIDJR		U		I_{DR}
FIDWR2		A	COMP4	NOT USED
FIWJ		U	INPT4	I_{WF}, I_{WR}
FIWJF		U		I_{WF}
FIWJR		U		I_{WR}
FJF	U	U	SUSCMP	F_{JFi}
FJP	U	U	TIRIN	F_j'
FKD		U	DRIVE	Kd
FKDO		U	DRIVI	Kd
FKP		U	DRIVE	K_p
FKPO		U	DRIVI	K_p
FKS1		U	DRIVE	K_{s_1}
FKS10		U	DRIVI	K_{s_1}
FKS2		U	DRIVE	K_{s_2}
FKS20		U	DRIVI	K_{s_2}
FKSKDO		U	DRIVI	K_s
FKSKID		U	DRIVE	K_s
FN	U		BARIER	F_N
FNP	U		BARIER	$(F_N)_{t-1}$
FNSTI	U		BARSTR	F_{NSTi}
FR	U	U	DIMV	F_{Ri}
FRCP	U	U	COMPN	F'_{R_i}
FRCPAV		U	COMP4	$(\Sigma F'_{R_i} \Delta t_n)/\Delta t$
FRCMPU		U	COMP4	$\mu_i F'_{R_i}$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
FRICF	U		HARDPT	$\frac{\mu_B F_{NSTi}}{FRICT}$
FRICT	U		BARIER	
FRSP	A	A	COMPN	NOT USED
FRTEST		U	COMP4	$F_{R_i} - F_{S_i} \sin \emptyset_{C_i}$
FRXFAC		U	COMP4	$A_{11}\cos\alpha_{GZ'_i} + A_{21}\cos\beta_{GZ'_i} + A_{31}\cos\gamma_{GZ'_i}$
FRXU	U	U	DIMV	F_{Rxu_i}
FRYFAC		U	COMP4	$A_{12}\cos\alpha_{GZ'_i} + A_{22}\cos\beta_{GZ'_i} + A_{32}\cos\gamma_{GZ'_i}$
FRYU	U	U	DIMV	F_{Ryu_i}
FRZFAC		U	COMP4	$A_{13}\cos\alpha_{GZ'_i} + A_{23}\cos\beta_{GZ'_i} + A_{33}\cos\gamma_{GZ'_i}$
FRZU	U	U	DIMV	F_{Rzu_i}
FS	U	U	DIMV	F_{S_i}
FSAV		U	COMP4	$(\Sigma F_{S_i} \Delta t_n)/\Delta t$
FSXFAC		U	COMP4	$A_{11}\cos\alpha_{S_i} + A_{21}\cos\beta_{S_i} + A_{31}\cos\gamma_{S_i}$
FSXU	U	U	DIMV	$F_{SX_{ui}}$
FSYFAC		U	COMP4	$A_{12}\cos\alpha_{S_i} + A_{22}\cos\beta_{S_i} + A_{32}\cos\gamma_{S_i}$
FSYU	U	U	DIMV	F_{Syu_i}
FSZFAC		U	COMP4	$A_{13}\cos\alpha_{S_i} + A_{23}\cos\beta_{S_i} + A_{33}\cos\gamma_{S_i}$
FSZU	U	U	DIMV	F_{Szu_i}
FXU	U	U	DIMV	F_{xu_i}
FYU	U	U	DIMV	F_{yu_i}
FZU	U	U	DIMV	F_{zu_i}
F1FI	U	U	DIMV	F_{1Fi}
F1RI	U	U	DIMV	F_{1Ri}
F2FI	U	U	DIMV	F_{2Fi}

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
F2RI	U	U	DIMV	F_{2Ri}
G	U	U	INPT	g
GAM1	U	U	COMP	γ_1
GAM2	U	U	COMP	$(\gamma_2)_t$
GAM3	U	U	COMP	$(\gamma_3)_t$
GAM4	U	U	COMP	$(\gamma_4)_t$
GAM5	U	U	COMP	$(\gamma_5)_t$
GAM6	U	U	COMP	$(\gamma_6)_t$
GAM7	U	U	COMP	$(\gamma_7)_t$
GAM8	U	U	COMP	$(\gamma_8)_t$
GAM9	U	U	COMP	$(\gamma_9)_t$
GCTCP	U	U	COMP	$g\ A_{33}$
GCTH	U	U	COMP	$g \cos \Theta$
GCTSP	U	U	COMP	$g\ A_{32}$
GEAR1		U	DRIVI	$GEAR_1$
GEAR2		U	DRIVI	$GEAR_2$
GEAR3		U	DRIVI	$GEAR_3$
GEAR4		U	DRIVI	$GEAR_4$
GHED	U	U	HEAD	TERRAIN TITLE
GN		U	INPT5	G_{iF}, G_{iR}
GSTH	U	U	COMP	$g \sin \Theta$
HCAH	U	U		$h_i \cos\alpha_{h_i}$
HCAH1	U	U	COMP	$h_1 \cos\alpha_{h_i}$
HCAH2	U	U	COMP	$h_2 \cos\alpha_{h_2}$
HCAH3	U	U	COMP	$h_3 \cos\alpha_{h_3}$
HACH4	U	U	COMP	$h_4 \cos\alpha_{h_4}$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
HCBH	U	U		$h_i \cos\beta_{h_i}$
HCBH1	U	U	COMP	$h_1 \cos\beta_{h_1}$
HCBH2	U	U	COMP	$h_2 \cos\beta_{h_2}$
HCBH3	U	U	COMP	$h_3 \cos\beta_{h_3}$
HCBH4	U	U	COMP	$h_4 \cos\beta_{h_4}$
HCGH	U	U		$h_i \cos\gamma_{h_i}$
HCGH1	U	U	COMP	$h_1 \cos\gamma_{h_1}$
HCGH2	U	U	COMP	$h_2 \cos\gamma_{h_2}$
HCGH3	U	U	COMP	$h_3 \cos\gamma_{h_3}$
HCGH4	U	U	COMP	$h_4 \cos\gamma_{h_4}$
HED	U	U	INPT	RUN TITLE
HI	U	U	DIMV	h_i
HMAX	U	U	INPT	h_{max}
HMIN	U	U	INPT	h_{min}
HMINR		A	INPT4	NOT USED
HRPSFA		A	COMP4	NOT USED
HRPSFB		A	COMP4	NOT USED
HRPSFC		A	COMP4	NOT USED
HTRERM		U	COMP4	$/h_i (RPS)_i/$
IAPFR	U	U	APTABL	
IBHIT	U		BARIER	
IBTYP		U	INPT5	TYPE
IBUG		U	INPT5	
ICBHIT	U	U	COMPN	
IDPT	U		BARIER	
IDRIVE		U	DRIVTT	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
IDRIVER		U	DRIVTT	
IDTCNT		U	COMP4	
IGEAR		U	DRIVE	
IHIT	U	U	COMPN	
ILOAD	U		BARIER	
INDB	U		INPT2	
INDCRB	U	U	INPT1	
INDXPT	U			
ININD	U		BARIER	
IPATHT		U	DRIVTT	
IPLN	U		BARIER	
IPT	U		BARIER	
IRPS		U	COMP4	
IRUF	U	U	RUFNES	
ISKIDP		U	DRIVE	
ISMAIN		U	DRIVE	
ISTEP		U	COMP4	
ISTOP		U	COMP4	
ISTOP	U		NSTOP	
ISUS	U	U	INSUS	
ITCHNG		U	DRIVTT	
ITESTT		U	DRIVTT	
ITIR	U	U	TIRIN	
IUVB		U	COMP4	
IUVS		U	COMP4	
IX	U	U	COMP	
IY	U	U	COMP	
I1	U		BARIER	
I2	U		BARIER	
I3	U		BARIER	
I4	U		BARIER	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
JBHIT	U		BARIER	
JCBHIT	U	U	COMPN	
JDEND		U	COMP5	
KCOUNT		U	DRIVE	
LCB1	U	U	COMPN	
LCB2	U	U	COMPN	
LLL	U	U	COMP	
MODE	U	U	INPT	
NAPF	U	U	APTABL	
NAPR	U	U	APTABL	
NBTYP		U	COMP5	
NBX	U	U	INPT	
NBY	U	U	INPT	
NCAMF	U	U	INSUS	
NCAMR	U	U	INSUS	
NCRBSL	U	U	NEWCRB	
NCYC	U		BARIER	
NDEL	U	U	INPT	NDEL
NDTHF	U	U	INSUS	
NDTHR	U	U	INSUS	
NEN		U	DRIVI	
NEND	U	U	RUFNES	
NEQ	U	U	INTG	
NEQR		A	INTR	NOT USED
NLDCTR	U		BARIER	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
NPAGE	U	U	HEAD	
NPD		U	DRIVE	
NRPM		U	INPT5	
NSEG		U	BARIER	
NTBL1	U	U	INPT	
NTBL2	U	U	INPT	
NTBL3	U	U	INPT	
NTLF		U	INPT5	
NTRAN		U	DR1VI	
NTTS		U	INPT5	
NTTI		U	INPT5	
NTT2		U	INPT5	
NTT3		U	INPT5	
NUNLD	U		BARIER	
NX	U	U	INPT	
NXFRCP		U	INPT4	
NXUGMU		U	INPT4	
NY	U	U	INPT	
NZTAB	U	U	INPT	
NZ5	U	U	INPT	
OMEGAO		U	DRIVI	
OMEGF	A	A	INPT	NOT USED
OMEGFC	U	U	INPT3	Ω_{FC}
OMEGFE	U	U	INPT3	Ω_{FE}
OMEGR	A	A	INPT	NOT USED
OMEGRC	U	U	INPT3	Ω_{RC}
OMEGRE	U	U	INPT3	Ω_{RE}
OMEGT	U	U	TIRIN	Ω_T
OMGPS	U	U	INPT1	Ω_ψ

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
OMT2A2	U	U	TIRIN	$\dfrac{\Omega_T A_2 A_3 (A_4 - A_2 \Omega_T)}{A_4 [\Omega_T A_1 A_2 (\Omega_1 - 1) - A_o]}$
OMT2M1	U	U	TIRIN	$\Omega_T A_1 A_2 (\Omega_T - 1)$
P	U	U		P
PC		U	COMP5	P_c
PCAB	U		BARIER	$(\cos\alpha_B)_{t-1}$
PCBB	U		BARIER	$(\cos\beta_B)_{t-1}$
PCGB	U		BARIER	$(\cos\gamma_B)_{t-1}$
PHFP	U	U	SUSCMP	NOT USED
PHGI	U	U	DIMV	$\emptyset_{G_i}$
PHG1	A	A	COMP	NOT USED
PHG2	A	A	COMP	NOT USED
PHIC	U	U	INPT	$\emptyset_C$
PHICI	U	U	DIMV	$\emptyset_{CGi}$
PHICLR	U	U	NEWCRB	
PHICM	U	U	NEWCRB	
PHIC1	U	U	INPT1	$\emptyset_{C1}$ (deg)
PHIC1R	U	U	COMPN	$\emptyset_{C1}$ (rad)
PHIC2	U	U	INPT1	$\emptyset_{C2}$ (deg)
PHIC2R	U	U	COMPN	$\emptyset_{C2}$ (rad)
PHIC3	U	U	NEWCRB	$\emptyset_{c3}$ (deg)
PHIC3R	U	U	NEWCRB	$\emptyset_{c3}$ (rad)
PHIC4	U	U	NEWCRB	$\emptyset_{c4}$ (deg)
PHIC4R	U	U	NEWCRB	$\emptyset_{c4}$ (rad)
PHIC5	U	U	NEWCRB	$\emptyset_{c5}$ (deg)
PHIC5R	U	U	NEWCRB	$\emptyset_{c5}$ (rad)
PHIC6	U	U	NEWCRB	$\emptyset_{c6}$ (deg)
PHIC6R	U	U	NEWCRB	$\emptyset_{c6}$ (rad)
PHIF	U	U		$\emptyset_F$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
PHIFD	U	U		$\dot{\phi}_F$
PHIFD2	U	U	SUSCMP	$\dot{\phi}_F^{\,2}$
PHIFO	U	U	INSUS	ϕ_{FO}
PHIFDO	U	U	INSUS	$\dot{\phi}_{FO}$
PHIF2	U	U	SUSCMP	$\phi_F^{\,2}$
PHII	U	U		ϕ_i
PHIN	U	U	COMP	ϕ'_n
PHIR	U	U		ϕ_R
PHIRC	U	U	INSUS	ϕ_{CR}
PHIRD	U	U		ϕ_R
PHIRD2	U	U	COMP	ϕ^2_R
PHIRO	U	U	INPT	ϕ_{R_o}
PHIROD	U	U	INPT	$\dot{\phi}_{R_o}$
PHIR2	U	U	COMP	ϕ^2_R
PHIT	U	U	COMP	ϕ_t or ϕ
PHITL	U	U	EINDEX	ϕ_{t-1}
PHITP	U	U		ϕ'_t
PHIO	U	U	INPT	ϕ_o
PHI1	U	U	DIMV	ϕ_1
PHI1D	U	U	COMPN	$\frac{d\phi_1}{d\delta_1} \dot{\delta}_1$
PHI2	U	U	DIMV	ϕ_2
PHI2D	U	U	COMPN	$\frac{d\phi_2}{d\delta_2} \dot{\delta}_2$
PHI3	U	U	DIMV	ϕ_3
PHI3D	U	U	SUSCMP	$\dot{\phi}_3$
PHI4	U	U	DIMV	ϕ_4
PHI4D	U	U	SUSCMP	$\dot{\phi}_4$
PHRP	A	A	COMP	NOT USED

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
PI	U	U	COMP	π
PI015R		U	COMP5	$15/\pi$
PI02	U	U	EINDEX	$\pi/2$
PI04	U	U	EINDEX	$\pi/4$
PONE		U	INPT5	P_1
PP		U	COMP5	P_j
PPD		U	DRIVE	
PPRB	U		BARIER	$(R_B)_{t-1}$
PQ	U	U	COMP	PQ
PQRMIN	U	U	INPT	
PR	U	U	COMP	PR
PSBDRY	U	U	INPT	Ψ_{BDRY}(rad)
PSBDRO	U	U	INPT	Ψ_{BDRY}(deg)
PSIF	U	U	INPT	$\dot{\Psi}_F$
PSIFDO	U	U	INPT1	Ψ_{Fio}
PSIFFH		U	DRIVE	
PSIFHO		U	DRIVI	
PSIFI	U	U		$\dot{\Psi}_{Fi}$
PSIFID	U	U		Ψ_{Fi}
PSIFIO	U	U	INPT1	Ψ_{Fio}
PSII	U	U		Ψ_i
PSIIP	U	U	DIMV	Ψ'_i
PSIJ		U	DRIVE	
PSIM		U	DRIVE	$(\psi_F)_{IDEAL}$
PSIN	U	U	COMP	Ψ'_n
PSISKD		U	DRIVE	$\Delta\psi_{sj}$
PSIT	U	U	COMP	Ψ_t or Ψ
PSITEM		U	COMP4	Ψ'_i sgn U_{Gi}
PSITL	U	U	EINDEX	Ψ_{t-1}
PSITP	U	U		Ψ'_t
PSIO	U	U	INPT	Ψ_o

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
PSI1	U	U	DIMV	Ψ_1
PSI2	U	U	DIMV	Ψ_2
PSI3	U	U	DIMV	Ψ_3
PSI4	U	U	DIMV	Ψ_4
PSZR	U		BARIER	$(Z_R)_{t-1}$
PTWO		U	INPTS	P_2
PVDEF	U		BARIER	$(y'_{cpm})_t - (y'_{cpm})_{t-1}$
PZERO		U	INPTS	P_{Fo}, P_{Ro}
PO	U	U	INPT	P_o
P1	U	U	COMP	$\cos\beta_{yw_i} \cos\gamma_{GZ'_i}$
P2	U	U	COMP	p^2
P3	U	U	COMP	$\cos\gamma_{yw_i} \cos\alpha_{GZ'_i}$
P4	U	U	COMP	$\cos\gamma_{GZ'_i} \cos\alpha_{yw_i}$
P5	U	U	COMP	$\cos\alpha_{yw_i} \cos\beta_{GZ'_i}$
P6	U	U	COMP	$\cos\alpha_{GZ'_i} \cos\beta_{yw_i}$
P7	U	U	COMP	$\cos\beta_{GZ'_i} \cos\gamma_{yw_i}$
QAY		U	DRIVE	a_y
QR	U	U	COMP	QR
QO	U	U	INPT	Q_O
Q2	U	U	COMP	Q^2
R	U	U		R
RAD	U	U	COMP	$180/\pi$
RB	U		BARIER	$(R_B)_t$
RB1	U		BARIER	R_{B1}
RF	U	U	INPT	R_F

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
RFPF	U	U	SUSCMP	$\rho_F\,\text{Ø}_F$
RFTF	U	U	COMP	R_F/T_F^2
RHFMUF	U	U	SUSCMP	$\rho_F M_{UF}$
RHF2MF	U	U	SUSCMP	$\rho_F^2 M_{UF}$
RF2MFI	U	U	SUSCMP	$I_F+\rho_F^2 M_{UF}$
RHMU2	U	U	COMP	$\rho^2 M_{UR}$
RHMR2I	U	U	COMP	$\rho^2 M_{UR} + I_R$
RHO	U	U	INPT	ρ
RHØF	U	U	INSUS	ρ_F
RHOF2	U	U	SUSCMP	ρ_F^2
RHOMAX		U	COMP5	$(\rho_{si})_{max}$
RHOMUR	U	U	COMP	ρM_{UR}
RHOS		U	COMP4	ρ_{S_i}
RHOSAV		U	COMP5	$(\Sigma\rho_{S_i}(\Delta t_n)/\Delta t$
RHOSMX		U	COMP5	
RHO2	U	U	COMP	ρ^2
RPF2M	U	U	SUSCMP	$\rho_F^2 M_{UF}\text{Ø}_F^2$
RPHFD	U	U	SUSCMP	$R\dot{\text{Ø}}_F$
RPHRD	U	U	COMP	$R\text{Ø}_R$
RPME		U	COMP5	
RPR	U	U	COMP	$\rho\text{Ø}_R$
RPSFA		U	COMP4	$\dfrac{I_{Wj}+1/4 I_{Dj} AR_j^2}{(I_{Wj}+1/4 I_{Dj} AR_j^2)^2-(1/4 I_{Dj} AR_j)^2}$
RPSFB		U	COMP4	$\dfrac{I_{Dj} AR_j^2}{(4I_{Wj}+I_{Dj} AR_j^2)-(1/2 I_{Dj} AR_j^2)^2}$
RPSFC		U	COMP4	$1/I_{Wj}+1/4 I_{Dj} AR_j^2$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
RPSFD		U	COMP4	$\frac{I_{Dj} AR_j^2}{4 I_{Wj} + I_{Dj} AR_j^2}$
RPSFE		U	COMP5	$12/(I_{wj} + 1/4\ I_{Dj} \overline{AR}_2^2)$
RPSI		U		$(RPS)_i$
RPSSM		U	COMP4	
RR	U	U	INPT	R_R
RRM		U	INPT4	R_{RMi}
RRMC		U	INPT4	C_{RRMi_2}
RRTR	U	U	SUSCMP	R_R/T_R
RRTS	U	U	COMP	$R_R T_S$
RR1	U		BARIER	RR_1
RR2	U		BARIER	RR_2
RR2P	U		BARIER	$(RR_2)_{t-1}$
RTF	U	U	SUSCMP	R_F/T_{SF}
RTR	U	U	COMP	R_R/T_S
RW	U	U	TIRIN	R_W
RWDRIV		U	COMP5	
RWHJB	U	U	INPT1	RWHJB
RWHJE	U	U	INPT1	RWHJE
RO	U	U	INPT	R_o
R2	U	U	CØMP	R^2
S		U	DRIVI	
SDEN	U		BARIER	$\Sigma\ (A_{INT})_i$
SECTP	U	U	COMP	$\sec \Theta_t$
SET	U		INPT2	$\overline{SET}$
SFCDTR		U	COMP4	$\Sigma F_{C_i} \Delta t_n$
SFRCPR		U	COMP4	$\Sigma F'_{R_i} \Delta t_n$
SFRX	U	U	COMPN	$\Sigma F_{Rx'_i}$
SFRY	U	U	COMPN	$\Sigma F_{Ry'_i}$
SFRZ	U	U	COMPN	$\Sigma F_{Rz'_i}$
SFSDTY		U	COMP4	$\Sigma F_{S_i} \Delta t_n$
SFXS	U	U	COMP	$\Sigma\ F_{XS}$
SFXU	U	U	COMP	$\Sigma\ F_{XU}$
SFYS	U	A	COMP	$\Sigma\ F_{YS}$
SFYU	U	U	COMP	$\Sigma\ F_{YU}$
SFYUF	A	A	COMP	NOT USED
SFYUR	A	A	COMP	NOT USED
SFZS	U	A	COMP	$\Sigma\ F_{ZS}$
SFZU	A	A	COMP	NOT USED
SFZ1	U	U	COMP	$\Sigma\ F_{Z1}$
SHED	U	U	HEAD	INITIAL CONDITION TITLE
SI	U	U	DIMV	S_i
SIGR	U		INPT2	σR_j
SIGT	U	U	TIRIN	σ_T
SINPH	U	U	COMP	$\sin \emptyset$
SINPHN	U	U	EINDEX	$\sin \emptyset'_n$
SINPS	U	U	COMP	$\sin \Psi$
SINPSN	U	U	EINDEX	$\sin \Psi'_n$
SINTH	U	U	COMP	$\sin \Theta$
SINTHN	U	U	EINDEX	$\sin \Theta'_n$
SLIP		U	COMP4	$(SLIP)_i$
SLIPAV		U	COMP4	$[\Sigma (SLIP)_i \Delta t_n]/\Delta t$
SLIPMT		U	INPT4	
SLIPMX		U	COMP5	
SLIPP		U	COMP5	$SLIP_P$
SLIPT		U	COMP4	$/(SLIP)_i/$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
SLOPE		U	DRIVE	
SLOPER		U	DRIVE	
SLOPE1	U	U	ADTNL	$\frac{d\emptyset_1}{d\delta_1}$
SLOPE2	U	U	ADTNL	$\frac{d\emptyset_2}{d\delta_2}$
SLOPE3	U	U	SUSCMP	$d\emptyset/d\delta_3$
SLOPE4	U	U	SUSCMP	$d\emptyset/d\delta_4$
SLPFAC		U	COMP4	h_i/U_{GWi}
SNPF	U	U	SUSCMP	$\Sigma N_{\emptyset F}$
SNPR	U	U	COMP	$\Sigma N_{\emptyset R}$
SNPS	U	A	COMP	$\Sigma N_{\emptyset S}$
SNPSS	U	A	COMP	$\Sigma N_{\Psi S}$
SNPSU	U	U	COMP	$\Sigma N_{\Psi U}$
SNPU	U	U	COMP	$\Sigma N_{\emptyset U}$
SNTS	U	A	COMP	$\Sigma N_{\Theta S}$
SNTU	U	U	COMP	$\Sigma N_{\Theta U}$
SPENGY	U		BARIER	
SPG	U	U	DIMV	$\sin\emptyset_{Gi}$
SPHI	A	A	COMP	NOT USED
SPHIC	U	A	ADTNL	$\sin\emptyset_{C_i}$
SPHICI		U	COMP4	$\sin\emptyset_{C_i}$
SPHTP	U	U	COMP	$\sin\emptyset'_t$
SPSI	A	A	COMP	NOT USED
SPSTP	U	U	EINDEX	$\sin\Psi'_t$
SPYG	U	U	DIMV	$\sin\emptyset_{yGi}$
SRHOS		U	COMP5	$\Sigma\rho s_i \Delta t_n$
SSLIP		U	COMP4	$\Sigma(SLIP)_i \Delta t_n$
ST		U	DRIVE	$(\Delta S.i/U_T)$
STEPD		U	COMP4	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
STG	U	U	DIMV	$\sin\Theta_{Gi}$
STHETP	U	U	EINDEX	$\sin\Theta'_t$
STSØ2		U	DRIVE	$(\Delta S.i/U_T)^2/2$
STXG	U	U	DIMV	$\sin\Theta_{XGi}$
SUMM	U	U	COMP	ΣM
SWORK	U		BARIER	
SXR	U		BARIER	$(\Sigma X_R)_t$
SYR	U		BARIER	$(\Sigma Y_R)_t$
SZR	U		BARIER	$(\Sigma Z_R)_t$
T	U	U	INTG	t
TANPCL	U	U	NEWCRB	
TANPCI	U	U	COMPN	$\tan\emptyset_{c_1}$
TANPC2	U	U	COMPN	$\tan\emptyset_{c_2}$
TANPC3	U	U	NEWCRB	$\tan\emptyset_{c_3}$
TANPC4	U	U	NEWCRB	$\tan\emptyset_{c_4}$
TANPC5	U	U	NEWCRB	$\tan\emptyset_{c_5}$
TANPC6	U	U	NEWCRB	$\tan\emptyset_{c_6}$
TANTP	U	U	COMP	$\tan\Theta'_t$
TAU		U	COMP5	τ_i
TAUA		U	INPT5	τ_A
TAUF		U	DRIVI	τ
TAUO		U	INPT5	$(\tau_i)_o$
TB	U	U	INPT	TB
TCT		U	INPT5	CT
TCTEST		U	DRIVTT	
TE	U	U	INPT	TE
TEMPOR		U	DRIVE	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
TERM		U	COMP4	arctan $\frac{\gamma_{Gi}}{/U_{Gi}/}$
TERMB		U	COMP4	$\emptyset_{C_i} - .6366\emptyset_{C_i}/\emptyset_{C_i}/$
TERMP		U	COMP4	$[\frac{V_{Gi}}{/U_{Gi}/} - \tan(\psi'_i \text{ sgn } U_{Gi}]^2$
TERMX		U	DRIVE	
TERMY		U	DRIVE	
TERM1	U	U	COMP	$Z_f + \delta_1 + h_1\cos\gamma_{h_1}$
TERM2	U	U	COMP	$Z_f + \delta_2 + h_2\cos\gamma_{h_2}$
TESTB		U	DRIVE	T_b
TESTBO		U	DRIVI	T_b
TESTR1		U	DRIVE	T_{R_1}
TESTR2		U	DRIVE	T_{R_2}
TESTS1		U	DRIVE	T_{S_1}
TESTS2		U	DRIVE	T_{S_2}
TESTT		U	DRIVI	
TF	U	U	INPT	T_F
TF02	U	U	COMP	$T_F/2$
TG61	U	U	COMP	$M_{UR}(\dot{\delta}_3 - \emptyset_R\dot{\emptyset}_R)$
THED	U	U	HEAD	TIRE TITLE
THESKD		U	DRIVE	Θ_c
THETAO	U	U	INPT	Θ_o
THETN	U	U	COMP	Θ'_n
THETT	U	U	COMP	Θ_t or Θ
THETTL	U	U	EINDEX	Θ_{t-1}
THETTP	U	U		Θ'_t
THGI	U	U	DIMV	Θ_{Gi}
THG1	U	U	COMP	
THG2	A	A	COMP	NOT USED

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
THMAX	U	U	INPT	Θ_{max}
TI	U	A	DIMV	T_i
TIHI		A	COMP4	NOT USED
TIL		U	DRIVI	T_I
TIMR		A	INTR	NOT USED
TINCR	U	U	INPT	
TITE		U	DRIVE	$t-t_j-\tau$
TIZ	U	U	COMP	$M_{UF}[a^2 + (\frac{T_F}{2})^2] + M_{UR}b^2$
TIZ2	U	U	COMP	$M_{UR}\rho^2\dot{\emptyset}_R$
TJ		U	DRIVE	
TL		U	DRIVI	T_L
TLAMB		U	COMP5	$I_{D_j}\overline{AR}^2/(41_{W_j} + I_{D_j}\overline{AR}^2)$
TLF		U	INPT5	LF
TMT		U	DRIVE	$(T_I-T_L)/T_L$
TM4	U	U	COMP	$T_F M_{UF}/4$
TPATH		U	DRIVTT	
TPC		U	INPT5	P_C
TPD		U	DRIVE	
TPF	U	U	SUSCMP	$T_F\emptyset_F/2$
TPR	U	U	COMP	$T_R\emptyset_R/2$
TPRINT				
TQB		U	COMP5	$(TQ)_{Bi}$
TQD		U	COMP5	$(TQ)_{Di}$
TQE		U	COMP5	$(TQ)_E$
TQF	U	A	INPT	$\overline{TQ}_F$
TQFAC		A	COMP4	NOT USED
TQR	U	A	INPT	$\overline{TQ}_R$
TR	U	U	INPT	T_R
TRH	U	U	COMP	$R_W - h_i$

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
TRO2	U	U	COMP	$T_R/2$
TRKIN		U	DRIVE	
TRPME		U	INPT5	RPME
TS	U	U	INPT	T_S
TSF	U	U	INSUS	T_{SF}
TSFØ2	U	U	SUSCMP	$T_{SF}/2$
TS02	U	U	COMP	$T_S/2$
TSTR10		U	DRIVI	T_{R_1}
TSTR20		U	DRIVI	T_{R_1}
TSTS10		U	DRIVI	T_{S_1}
TSTS20		U	DRIVI	T_{S_1}
TT		U	INPT5	
TTAU		U	INPT5	
TTEM		U	DRIVE	
TTPSIT		U	DRIVE	
TTR		U	INPT5	(TR)
TTS		U	INPT5	(TS)
TWOPI	U	U	EINDEX	2π
TWOPIR		U	COMP4	$1/(2\pi)$
TWOT		U	INPT5	WOT
TX	U	U	COMP	$X'_{GPi} - X'_i$
TY	U	U	COMP	$Y'_{GPi} - Y'_i$
TZ	U	U	COMP	$Z'_{GPi} - Z'_i$
TO	U	U	INPT	
T1	U	U	INPT	
T1PSI	U	U	COMPN	$T_{1\psi}$
T2PSI	U	U	COMPN	$T_{2\psi}$
U	U	U		u
UG	U	U	DIMV	u_{Gi}

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
UGW		U	COMP5	u_{GW_i}
UI	U	U		u_i
UNP	U		BARIER	u'_n
UP		A	DRIVE	NOT USED
UPT	U		HARDPT	U'_{Ri}
UQ	U	U	COMP	uQ
UR	U	U	COMP	uR
URP	U		BARIER	u'_r
UT		U	DRIVE	u_T
UTMPH		U	DRIVE	U_T
UVWMIN	U	U	INPT	
UO	U	U	INPT	u_o
U1	U	U	ADTNL	u_1
U2	U	U	ADTNL	u_2
U3	U	U	ADTNL	u_3
U4	U	U	ADTNL	u_4
V	U	U		v
VAR	U	U	INTG	[y]
VARR		A	INTR	NOT USED
VDEF	U		BARIER	$Y'_{CPm} - (Y'_{BP})_t$
VECS		U	COMP5	$\sqrt{U_{G_1}{}^2 + Y_{G_i}{}^2}$
VG	U	U	DIMV	v_{Gi}
VGR12		U	DRIVI	VGR_{12}
VGR21		U	DRIVI	VGR_{21}
VGR23		U	DRIVI	VGR_{23}
VGR32		U	DRIVI	VGR_{32}
VGR34		U	DRIVI	VGR_{34}

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
VGR43		U	DRIVI	VGR_{43}
VHED		U	HEAD	VEHICLE TITLE
VI	U	U		v_i
VL	U		BARIER	$\delta_B - \epsilon_{n-1}$
VMAX	U		BARIER	δ_B
VNP	U		BARIER	v'_n
VP	U	U	COMP	vP
VPT	U		HARDPT	v'_{Ri}
VR	U	U	COMP	vR
VRP	U		BARIER	v'_r
VTAN	U		BARIER	VTAN
VO	U	U	INPT	v_o
V1	U	U	ADTNL	v_1
V2	U	U	ADTNL	v_2
V3	U	U	ADTNL	v_3
V4	U	U	ADTNL	v_4
W	U	U		w
WI	U	U		w_i
WNP	U		BARIER	w'_n
WP	U	U	COMP	wP
WPT	U		HARDPT	w'_{Ri}
WQ	U	U	COMP	wQ
WRP	U		BARIER	w'_r
WO	U	U	INPT	w_o
W1	U	U	ADTNL	w_1
W2	U	U	ADTNL	w_2
W3	U	U	ADTNL	w_3
W4	U	U	ADTNL	w_4

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
X		U	DRIVE	
XB	U	U	INPT	X_B
XBB	U		BARIER	X_{BB}
XBT	U		BARIER	X_{BI}
XBDRY	U	U	INPT	X_{BDRY}
XBRAK		A	COMP4	NOT USED
XCP	U	U		X'_C
XCPBP	U		BARIER	X'_{CPB}
XCPN	U		BARIER	X_{CPn}
XCPNP	U		BARIER	X'_{CPn}
XCPTP	U		BARIER	X'_{CPT}
XCOP	U	U	INPT	X'_{CO}
XE	U	U	INPT	X_E
XF	U		BARIER	$(F_B)_{t-1}$
XGPP	U	U	DIMV	X'_{GPi}
XIF	U	U	INSUS	I_F
XIMPOR		U	DRIVI	WI_i
XINCR	U	U	INPT	X_{INCR}
XINDL	U	U	EINDEX	
XINDN	U	U	EINDEX	
XINPT	A		INPT2	NOT USED
XINPT5		A	INPT5	NOT USED
XINT		U	DRIVE	
XIPS	U	U	INPT1	I_ψ
XIR	U	U	INPT	I_R
XIX	U	U	INPT	I_X
XIXP	U	U	COMP	$(I'_X)_t$
XIXZ	U	U	INPT	I_{XZ}
XIXZP	U	U	COMP	$(I'_{XZ})_t$
XIY	U	U	INPT	I_Y

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
XIYP	U	U	ADTNL	$(I'_Y)_t$
XIYZP	U	U	COMP	$(I'_{YZ})_t$
XIZ	U	U	INPT	I_Z
XIZP	U	U	COMP	$(I'_Z)_t$
XIZR	U	U	COMP	$I_Z + I_R$
XLAMF	U	U	INPT	λ_F
XLAMR	U	U	INPT	λ_R
XLAMT	U	U	TIRIN	λ_T
XLDP	A		BARIER	NOT USED
XLM1	U	U	DIMV	λ_{1i}
XLM2	U	U	DIMV	λ_{2i}
XLM3	U	U	DIMV	λ_{3i}
XM	A		INPT2	NOT USED
XMS	U	U	INPT	M_S
XMTF04	U	U	COMP	$T_F M_{UF}/4$
XMTR04	U	U	SUSCMP	$M_{UR} T_R/4$
XMTX	U		BARIER	
XMUR	U	U	INPT	M_{UF}
XMUF02	U	U	COMP	$M_{UF}/2$
XMUGI	U	U	COMPN	$AMUG_i$, AMU
XMUI	U	U	COMP4	μ_i
XMUM		U	INPT4	μ_{mi}
XMUMAT		U	INPT4	
XMUR	U	U	INPT	M_{UR}
XMURØ2	U	U	SUSCMP	$M_{UR}/2$
XMUXP		U	INPT4	μ_{XP}
XMUXS		U	INPT4	μ_{XS}
XMXPMT		U	INPT4	
XMXSMT		U	INPT4	
XNN	U		BARIER	x_N

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
XP	U	U		x'_i
XPS	U	U	INPT1	$\overline{PT}$
XRI	U		BARIER	$(x_R)_i$
XSTI	U		BARSTR	X_{ST_i}
XSTIØ	U		BARSTR	X_{STi_0}
XSTIP	U		BARSTR	X'_{ST_i}
XTRA	A	A	ADTNL	NOT USED
XVF	U		INPT2	x_{VF}
XVP		U	DRIVE	X'_{VP_i}
XVR	U		INPT2	x_{VR}
XXFRCP		U	INPT4	
XXUGMU		U	INPT4	
XXX	U	U	COMP	
XXZGP5	U	U	INPT	
XX1	U	U	COMP	
XX2	U	U	COMP	
X1	U	U	INPT	x_1
X1P	U	U	DIMV	x'_1
X2	U	U	INPT	x_2
X2P	U	U	DIMV	x'_2
X3P	U	U	DIMV	x'_3
X4P	U	U	DIMV	x'_4
Y		U	DRIVE	
YB	U	U	INPT	Y_B
YBB	U		BARIER	y_{BB}
YBPT	U		BARIER	y'_{BT}
YBDRY	U	U	INPT	Y_{BDRY}
YḂPTP	U		BARIER	y'_{BPT}
YBPO	U		INPT2	y'_{Bo}
YBT	U		BARIER	y_{BT}

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
YCIP	U	U		y'_{Ci}
YCLP	U	U	NEWCRB	
YCMP	U	U	NEWCRB	
YCP	U	U		y'_{C}
YCPBP	U		BARIER	y'_{CPB}
YCPMP	U		BARIER	y'_{CPm}
YCPN	U		BARIER	y_{CPn}
YCPNP	U		BARIER	y'_{CPn}
YCPTP	U		BARIER	y'_{CPT}
YCOP	U	U	INPT	y'_{Co}
YCIP	U	U	INPTI	y'_{CI}
YC2P	U	U	INPTI	y'_{C2}
YC3P	U	U	NEWCRB	y'_{c3}
YC4P	U	U	NEWCRB	y'_{c4}
YC5P	U	U	NEWCRB	y'_{c5}
YC6P	U	U	NEWCRB	y'_{c6}
YE	U	U	INPT	Y_{E}
YGPPP	U	U	DIVM	y'_{GPi}
YINCR	U	U	INPT	Y_{INCR}
YNN	U		BARIER	y_{n}
YP	U	U		y'_{i}
YRI	U		BARIER	$(y_R)_i$
YSTI	U		BARSTR	y_{STi}
YSTIØ	U		BARSTR	y_{STi0}
YSTIP	U		BARSTR	y'_{STi}
YSTIPØ	U		BARSTR	y'_{STi0}
YTRANS		U	DRIVI	
YV	U		INPT2	y_{V}
YVP		U	DRIVE	y'_{VPi}
YYY	U	U	COMP	

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
YYZGP5	U	U	INPT	
YY1	U	U	COMP	
YY2	U	U	COMP	
Y1	U	U	INPT	y_{1}
YIP	U	U	DIMV	y'_{I}
Y2	U	U	INPT	y_{2}
Y2P	U	U	DIMV	y'_{2}
Y3P	U	U	DIMV	y'_{3}
Y4P	U	U	DIMV	y'_{4}
ZBB	U		BARIER	Z_{BB}
ZBBP	U		INPT2	z'_{BB}
ZBT	U		BARIER	z_{BT}
ZBTP	U		INPT2	z'_{BT}
ZCLP	U	U	NEWCRB	
ZCMP	U	U	NEWCRB	
ZCP	U	U		z'_{C}
ZCPBP	U		BARIER	z_{CPB}
ZCPN	U		BARIER	z_{CPn}
ZCPNP	U		BARIER	z'_{CPn}
ZCPTP	U		BARIER	z'_{CPT}
ZCOP	U	U	INPT	z'_{CO}
ZC2P	U	U	INPTI	z'_{C2}
ZC3P	U	U	NEWCRB	Z'_{C3}
ZC4P	U	U	NEWCRB	Z'_{C4}
ZC5P	U	U	NEWCRB	Z'_{C5}
ZC6P	U	U	NEWCRB	Z'_{C6}
ZETAB		U	INPT5	ζ_{B}
ZETA3	A	A	COMP	NOT USED
ZETA3D	A	A	COMP	NOT USED

PROGRAM VARIABLE	R D	V D	COMMON BLOCK	ANALYTICAL VARIABLE OR EXPRESSION
ZETA4	A	A	COMP	NOT USED
ZETA4D	A	A	COMP	NOT USED
ZF	U	U	INPT	Z_F
ZFD1	U	U	SUSCMP	$Z_F+\delta_1$
ZFD1RF	U	U	SUSCMP	$Z_F+\delta_1+\rho_F$
ZFD12	U	U	COMP	$Z_F + (\delta_1 + \delta_2)/2$
ZFD2	U	U	SUSCMP	$Z_F+\delta_2$
ZFD3R	U	U	COMP	$Z_F + \rho + \delta_3$
ZFØ	U	U	SUSCMP	$Z_F+\rho_F$
ZGP	U	U	INPT	Z'_G
ZGPP	U	U	DIMV	Z'_{GPi}
ZNN	U	U	BARIER	z_n
ZP	U	U		z'_i
ZPGI	U	U	DIMV	z'_{Gi}
ZPR	U	U	COMP	$z_F + \rho$
ZR	U	U	INPT	z_R
ZRD3	U	U	COMP	$z_R + \delta_3$
ZRD3R	U	U	COMP	$z_R + \delta_3 + \rho$
ZRD34	U	U	SUSCMP	$Z_R+(\delta_3+\delta_4)/2$
ZRD4	U	U	SUSCMP	$Z_R+\delta_4$
ZRI	U		BARIER	$(z_R)_i$
ZRO	U	U	COMP	$z_R + \rho$
ZSTI	U		BARSTR	Z_{STi}
ZSTIØ	U		BARSTR	Z_{STi_0}
ZSTIP	U		BARSTR	Z'_{STi}
ZVB	U		INPT2	z_{VB}
ZVT	U		INPT2	z_{VT}
ZZ1	U	U	COMP	
ZZZ	U	U	COMP	
Z1	U	U	INPT	z_1
Z1P	U	U	DIMV	z'_1
Z2	U	U	INPT	z_2
Z2P	U	U	DIMV	z'_2
Z3P	U	U	DIMV	z'_3
Z4P	U	U	DIMV	z'_4

3. PROGRAM DOCUMENTATION

3.1 Roadside Design Version

A description of each computational subroutine of the HVOSM-RD2 is provided in this section. Included is a brief description of the purpose of the subroutine, a description of the linkages to the rest of the program in the forms of subroutine called, calling arguments, common blocks appearing, variables within the common blocks that are the result of a computation, and, in the subroutine size. Also included is a description of the computational procedure employed either in the form of a verbal listing of the computational steps or an annotated flowchart illustrating the logical sequence of computations. Since this part of the subroutine description is intended to illustrate the procedure, it does not always illustrate each individual line of coding. When a detailed investigation of the coding is required, the computational procedure should be used in conjunction with a subroutine listing.

An overall program block diagram is shown in Figure 3.1-1, a matrix of common blocks appearing in each subroutine in Figure 3.1-2, and a matrix of subroutine calls in Figure 3.1-3.

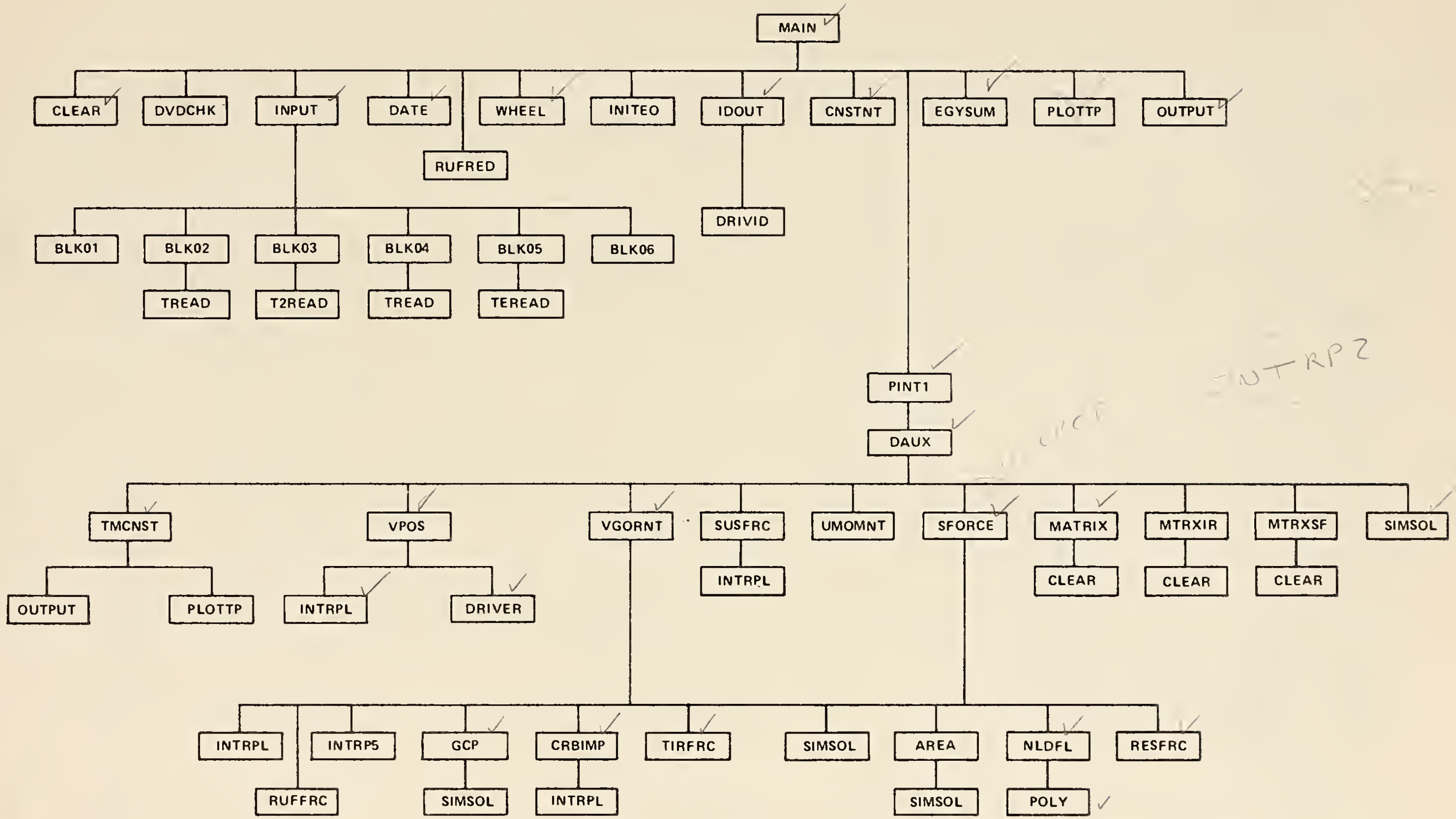


Figure 3.1-1 HVOSM-RD2 OVERALL PROGRAM BLOCK DIAGRAM

COMMON BLOCK

SUBROUTINE	HEAD	INPT	INPT1	INTG	DIMV	COMP	COMPN	EINDEX	ADTNL	INPT2	INPT3	TIRIN	BARIER	APTABL	INSUS	SUSCMP	NEWCRB	BARSTR	HARDPT	RUFNES	DRIVTT	DRIVI	NSTOP	
MAIN	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	
AREA					●								●											
BLKO1	●	●	●							●					●		●							
BLKO2	●	●	●							●	●			●	●			●						
BLKO3	●		●									●												
BLKO4	●	●																						
BLKO5	●	●	●							●							●			●				
BLKO6	●	●	●												●									
CLEAR																								
CNSTNT	●	●	●	●		●	●	●		●		●	●		●	●	●							
CRBIMP			●	●	●	●	●					●					●							
DATE																								
DAUX		●	●	●	●	●	●		●	●					●	●					●		●	
DRIVER																								
DRIVID																								
DVDCHK																								
EGYSUM										●			●											
GCP					●	●						●												
IDOUT	●	●	●			●				●	●	●		●	●		●	●						
INITEQ																								
INPUT																								
INTRPL																								
INTRP5		●			●	●	●					●												
MATRIX		●		●	●	●			●															
MTRXIR		●		●	●	●			●							●								
MTRXSF		●		●	●	●			●						●	●								
NLDFL				●						●			●											
OUTPUT	●	●		●	●	●	●		●						●	●		●						
PINT1																								
PLOTTP		●		●	●	●	●					●												
POLY																								
RESFRC				●	●	●				●			●					●	●					
RUFRED																								
RUFFRC			●		●	●	●					●								●				
SIMSOL																								
SFORCE				●	●	●	●			●			●					●	●					
SUSFRC		●		●	●	●			●		●			●	●	●								
TEREAD		●																						
TIRFRC					●	●	●		●			●												
TMCNST		●		●		●		●							●	●								
TREAD																								
T2READ																								
UMOMNT		●			●	●									●	●								
VGORNT		●	●	●	●	●	●		●			●			●	●	●			●				
VPOS		●	●	●	●	●	●		●						●	●								
WHEEL																								

Figure 3.1-2 HVOSM-RD2 COMMON BLOCK ALLOCATIONS

CALLED ROUTINE

CALLING ROUTINE	MAIN	AREA	BLKO1	BLKO2	BLKO3	BLKO4	BLKO5	BLKO6	CLEAR	CNSTNT	CRBIMP	DATE	DAUX	DRIVER	DRIVID	DVDCHK	EGYSUM	GCP	IOOUT	INITEO	INPUT	INTRPL	INTRP5	MATRIX	MTRXIR	MTRXSF	NLDFL	OUTPUT	PINT1	PLOTTP	POLY	RESFRC	RUFFRC	RUFREO	SIMSOL	SFORCE	SUSFRC	TEREAO	TIRFRC	TMCNST	TREAO	T2READ	UMOMNT	VGORNT	VPOS	WHEEL
MAIN									●	●		●				●	●		●	●	●							●	●	●				●												●
AREA																																			●											
BLKO1																																														
BLKO2																																									●					
BLKO3																																										●				
BLKD4																																									●					
BLKO5																																						●								
BLKO6																																														
CLEAR																																														
CNSTNT																																														
CRBIMP																						●																								
DATE																																														
DAUX																								●	●	●									●	●	●			●			●	●	●	
DRIVER																																														
DRIVID																																														
DVDCHK																																														
EGYSUM																																														
GCP																																			●											
IDDUT															●																															
INITEO																																														
INPUT			●	●	●	●	●	●																																						
INTRPL																																														
INTRP5																																														
MATRIX									●																																					
MTRXIR									●																																					
MTRXSF									●																																					
NLDFL																															●															
OUTPUT																																														
PINT1													●																																	
PLOTTP																																														
POLY																																														
RESFRC																																														
RUFFRC																						●																								
RUFRED																																														
SIMSOL																																														
SFDRCE		●																									●					●			●											
SUSFRC																						●																								
TEREAD																																														
TIRFRC																																														
TMCNST																												●		●																
TREAD																																														
T2READ																																														
UMOMNT																																														
VGORNT											●							●					●										●						●							
VPOS														●							●																									
WHEEL																																														

Figure 3.1-3 HVOSM-RD2 MATRIX OF SUBROUTINE CALLS

3.1.1 HVOSM-RD2 Subroutine Documentation

1. MAIN ROUTINE

a. Purpose:

1. Clear selected COMMON blocks
2. Obtain input and print input
3. Program initialization
4. Control computation of constants
5. Control the integration loop
6. Control abnormal program stops
7. Control indexing of coordinate system
8. Control integration step size for curb and sprung mass impact
9. Control output

b. Common Blocks Required:

HEAD, INPT, INPT1, INTG, DIMV, COMP, COMPN, EINDEX, ADTNL, INPT2, INPT3, TIRIN, BARIER, APTABL, INSUS, SUSCMP, NEWCRB, BARSTR, DRIVTT, DRIVI, NSTOP, HARDPT, RUFNES

c. Subroutines Required:

CLEAR, CNSTNT, DATE, DVDCHK, EGYSUM, IDOUT, INITEQ, INPUT, OUTPUT, PINT1, PLOTTP, WHEEL, RUFRED

d. Arguments:

None

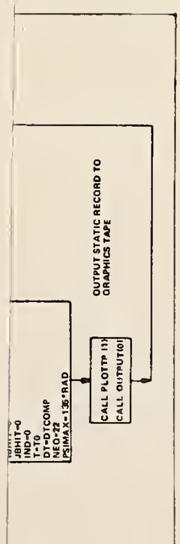
e. Common Variables Claculated:

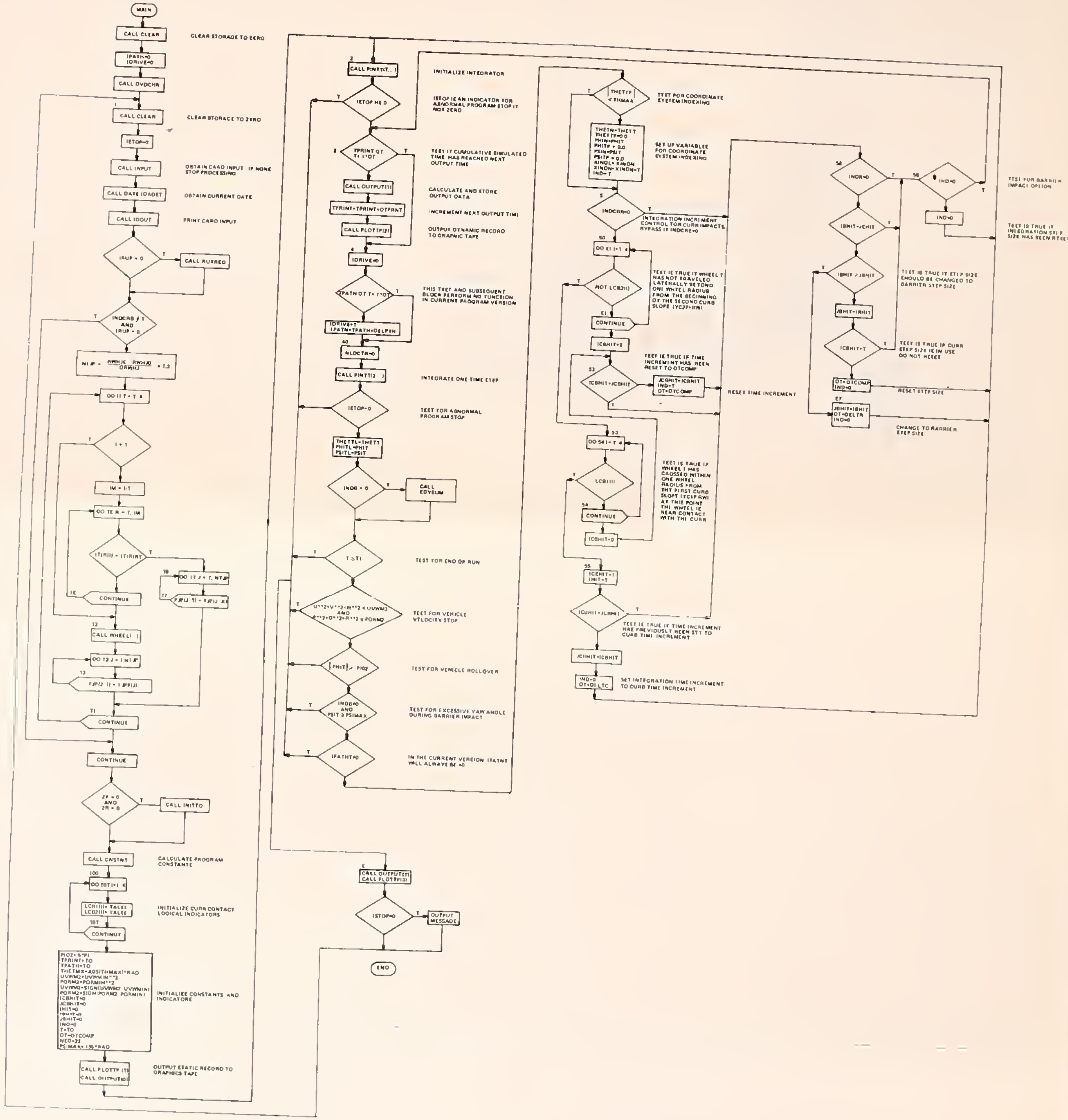
T, DT, FJP, NEQ, ZGP, DADE, DELG, IHIT, IRUF, LCB1, LCB2, NEND, NFJP, PHIN, PIO2, PSIN, DGMAX, ILOAD, JBHIT, PHITL, THETN, TPATH, UVWM2, XINDL, XINDN, ICBHIT, IDRIVE, IPATHT, JCBHIT, NLDCTR, PSIMAX, THETTL

f. Size:

$D42)_{16} = 3394)_{10}$ bytes

g. Computational Procedure:


JBHIT=0
IND=0
T=T0
DT=DTCOMP
NEQ=22
PSIMAX=135*RAD
CALL PLOTTP (1)
CALL OUTPUT(0)
OUTPUT STATIC RECORD TO
GRAPHICS TAPE


MAIN
CALL CLEAR
CLEAR STORAGE TO ZERO
CALL INPUT
CALL DATE
OBTAIN CURRENT DATE
PRINT CARD INPUT
CALL CNSTNT
CALCULATE PROGRAM CONSTANTS
INITIALIZE CURB CONTACT LOGICAL INDICATORS
INITIALIZE CONSTANTS AND INDICATORS
OUTPUT STATIC RECORD TO GRAPHICS TAPE
INITIALIZE INTEGRATOR
CALCULATE AND STORE OUTPUT DATA
INCREMENT NEXT OUTPUT TIME
OUTPUT DYNAMIC RECORD TO GRAPHIC TAPE
INTEGRATE ONE TIME STEP
TEST FOR ABNORMAL PROGRAM STOP
TEST FOR END OF RUN
TEST FOR VEHICLE VELOCITY STOP
TEST FOR VEHICLE ROLLOVER
TEST FOR EXCESSIVE YAW ANGLE DURING BARRIER IMPACT
OUTPUT MESSAGE
END
TEST FOR COORDINATE SYSTEM INDEXING
SET UP VARIABLES FOR COORDINATE SYSTEM INDEXING
RESET TIME INCREMENT
SET INTEGRATION TIME INCREMENT TO CURB TIME INCREMENT
RESET STEP SIZE
CHANGE TO BARRIER STEP SIZE

?. SUBROUTINE AREA

a. Purpose:

1. Compute the intersection area between the vehicle and barrier
2. Compute the projection of the intersection volume centroid on the interface

b. Common Blocks Required:

DIMV, BARIER

c. Subroutines Required:

SIMSOL

d. Arguments:

None

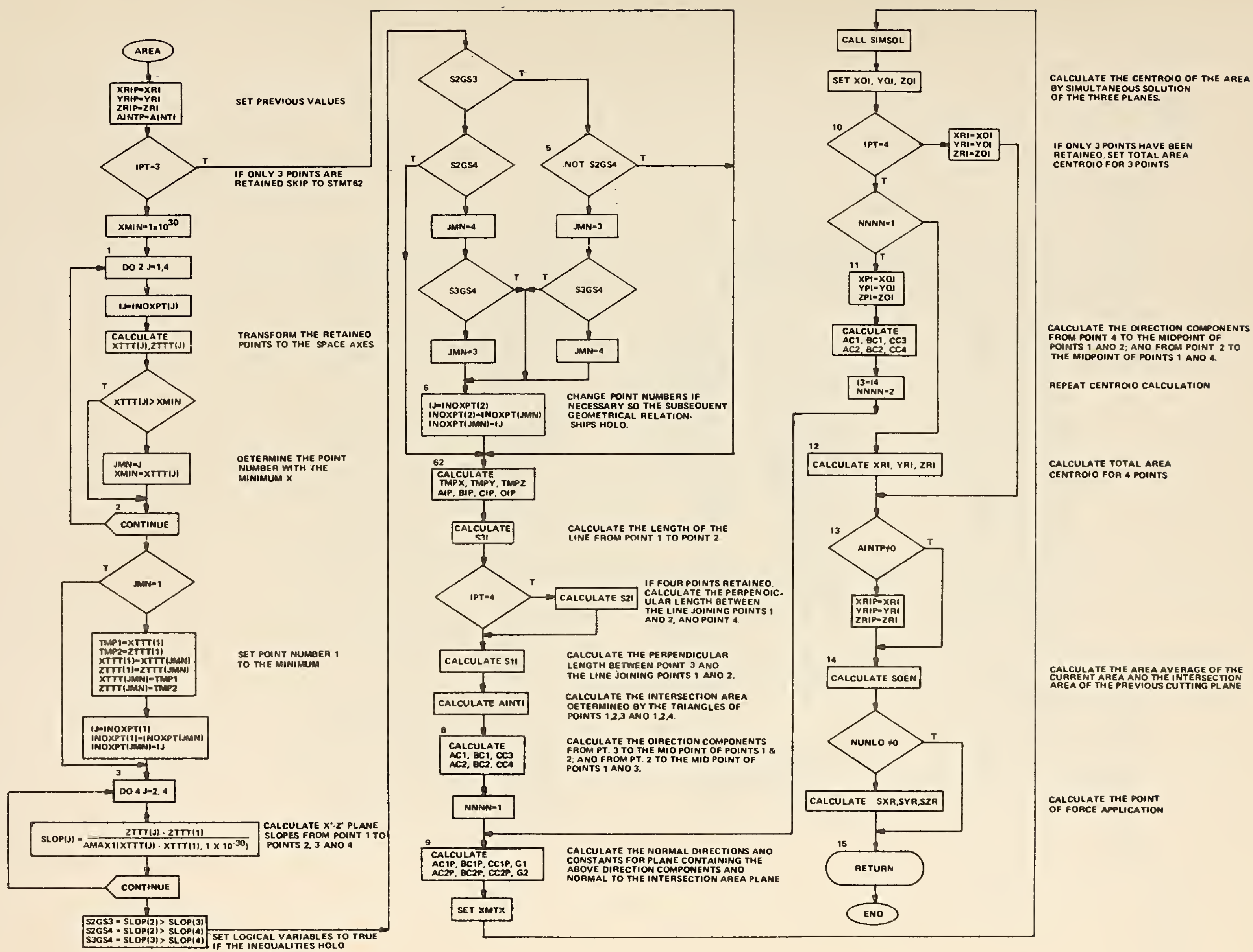
e. Common Variables Calculated:

I3, SRX, SRY, SRZ, XRI, YRI, ZRI, SDEN, XMTX, AINTI, AINTP, INDXPT

f. Size:

$9F6)_{16} = 2550)_{10}$ bytes

g. Computational Procedure:


AREA
XRIP=XRI
YRIP=YRI
ZRIP=ZRI
AINTP=AINTI
SET PREVIOUS VALUES
IPT=3
T
IF ONLY 3 POINTS ARE RETAINED SKIP TO STMT62
XMIN=1x10^30
1
DO 2 J=1,4
IJ=INOXPT(J)
CALCULATE XTTT(J),ZTTT(J)
TRANSFORM THE RETAINED POINTS TO THE SPACE AXES
XTTT(J)> XMIN
JMN=J
XMIN=XTTT(J)
DETERMINE THE POINT NUMBER WITH THE MINIMUM X
2
CONTINUE
JMN=1
TMP1=XTTT(1)
TMP2=ZTTT(1)
XTTT(1)=XTTT(JMN)
ZTTT(1)=ZTTT(JMN)
XTTT(JMN)=TMP1
ZTTT(JMN)=TMP2
SET POINT NUMBER 1 TO THE MINIMUM
IJ=INOXPT(1)
INOXPT(1)=INOXPT(JMN)
INOXPT(JMN)=IJ
3
DO 4 J=2, 4
SLOP(J) = ZTTT(J) - ZTTT(1) / AMAX1(XTTT(J) - XTTT(1), 1 X 10^-30)
CALCULATE X'-Z' PLANE SLOPES FROM POINT 1 TO POINTS 2, 3 AND 4
CONTINUE
S2GS3 = SLOP(2) > SLOP(3)
S2GS4 = SLOP(2) > SLOP(4)
S3GS4 = SLOP(3) > SLOP(4)
SET LOGICAL VARIABLES TO TRUE IF THE INEQUALITIES HOLD
S2GS3
S2GS4
5
NOT S2GS4
JMN=4
JMN=3
S3GS4
S3GS4
JMN=3
JMN=4
6
IJ=INOXPT(2)
INOXPT(2)=INOXPT(JMN)
INOXPT(JMN)=IJ
CHANGE POINT NUMBERS IF NECESSARY SO THE SUBSEQUENT GEOMETRICAL RELATIONSHIPS HOLD.
62
CALCULATE TMPX, TMPY, TMPZ AIP, BIP, CIP, DIP
CALCULATE S3I
CALCULATE THE LENGTH OF THE LINE FROM POINT 1 TO POINT 2
IPT=4
CALCULATE S2I
IF FOUR POINTS RETAINED, CALCULATE THE PERPENDICULAR LENGTH BETWEEN THE LINE JOINING POINTS 1 AND 2, AND POINT 4.
CALCULATE S1I
CALCULATE THE PERPENDICULAR LENGTH BETWEEN POINT 3 AND THE LINE JOINING POINTS 1 AND 2.
CALCULATE AINTI
CALCULATE THE INTERSECTION AREA DETERMINED BY THE TRIANGLES OF POINTS 1,2,3 AND 1,2,4.
8
CALCULATE AC1, BC1, CC3 AC2, BC2, CC4
CALCULATE THE DIRECTION COMPONENTS FROM PT. 3 TO THE MID POINT OF POINTS 1 & 2; AND FROM PT. 2 TO THE MID POINT OF POINTS 1 AND 3.
NNNN=1
9
CALCULATE AC1P, BC1P, CC1P, G1 AC2P, BC2P, CC2P, G2
CALCULATE THE NORMAL DIRECTIONS AND CONSTANTS FOR PLANE CONTAINING THE ABOVE DIRECTION COMPONENTS AND NORMAL TO THE INTERSECTION AREA PLANE
SET XMTX
CALL SIMSOL
SET XOI, YOI, ZOI
CALCULATE THE CENTROID OF THE AREA BY SIMULTANEOUS SOLUTION OF THE THREE PLANES.
10
IPT=4
XRI=XOI
YRI=YOI
ZRI=ZOI
IF ONLY 3 POINTS HAVE BEEN RETAINED, SET TOTAL AREA CENTROID FOR 3 POINTS
NNNN=1
11
XPI=XOI
YPI=YOI
ZPI=ZOI
CALCULATE AC1, BC1, CC3 AC2, BC2, CC4
CALCULATE THE DIRECTION COMPONENTS FROM POINT 4 TO THE MIDPOINT OF POINTS 1 AND 2; AND FROM POINT 2 TO THE MIDPOINT OF POINTS 1 AND 4.
13=14
NNNN=2
REPEAT CENTROID CALCULATION
12
CALCULATE XRI, YRI, ZRI
CALCULATE TOTAL AREA CENTROID FOR 4 POINTS
13
AINTP≠0
XRIP=XRI
YRIP=YRI
ZRIP=ZRI
14
CALCULATE SOEN
CALCULATE THE AREA AVERAGE OF THE CURRENT AREA AND THE INTERSECTION AREA OF THE PREVIOUS CUTTING PLANE
NUNLO ≠0
CALCULATE SXR,SYR,SZR
CALCULATE THE POINT OF FORCE APPLICATION
15
RETURN
END

3. SUBROUTINE BLK01

a. Purpose:

1. Assign input values of simulation control data

b. Common Blocks Required:

HEAD, INPT, INPT1, INPT2, INSUS, NEWCRB

c. Subroutines Required:

None

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

EM, T0, T1, AAA, BET, HED, EBAR, HMAX, HMIN, INDB, ISUS, MODE, DELTB, DELTC, NPAGE, THMAX, DTCOMP, DTPRNT, INDCRB, NCRBSL, PQRMIN, UVWMIN

f. Size:

$480)_{16} = 1152)_{10}$ bytes

4. SUBROUTINE BLK02

a. Purpose:

1. Assign input values of simulation vehicle data

b. Common Blocks Required:

HEAD, INPT, INPT1, INPT2, INPT3, APTABL, INSUS, BARSTR

c. Subroutine Required:

TREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

A, B, G, CF, CR, RF, RR, TF, TR, TS, X1, X2, YV, Y1, Y2, ZF, ZR, Z1, Z2, AKF, AKR, AKV, APF, APR, CFP, CRP, RHO, TSF, XIF, XIR, XIX, XIY, XIZ, XMS, XPS, XVF, XVR, ZVB, ZVT, AKDS, AKFC, AKFE, AKPS, AKRC, AKRE, AKRS, AKST, CPSP, DDEL, DELB, DELE, DTHF, DTHR, EPSF, EPSR, NAPF, NAPR, NDEL, PHIC, RHOF, VHED, XIPS, XIXZ, XMUF, XMUR, AKDS1, AKDS2, AKDS3, AKFCP, AKFEP, AKRCP, AKREP, DAPFB, DAPFE, DAPRB, DAPRE, DDAPF, DDAPR, EPSPS, IAPFR, NDTHF, NDTHR, NPAGE, OMGPS, PHIRC, XLAMF, XLAMR, XSTIO, YSTIO, ZSTIO, OMEGFC, OMEGFE, OMEGRC, OMEGRE

f. Size:

$906)_{16} = 2310)_{10}$ bytes

5. SUBROUTINE BLK03

a. Purpose:

1. Assign input values of simulation tire data

b. Common Blocks Required:

HEAD, INPT1, TIRIN

c. Subroutines Required:

T2READ

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

A0, A1, A2, A3, A4, RW, AKT, AMU, ITIR, SIGT, THED, DRWHJ, OMEGT, RWHJE, XLAMT

f. Size:

$45A)_{16} = 1114)_{10}$ bytes

6. SUBROUTINE BLK04

a. Purpose:

1. Assign input values of simulation vehicle control data

b. Common Blocks Required:

HEAD, INPT

c. Subroutines Required:

TREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

TQF, TQR, CHED, PSIF, NPAGE, NTBL1, NTBL2, NTBL3, TINCR

f. Size:

$3AC)_{16} = 940)_{10}$ bytes

7. SUBROUTINE BLK05

a. Purpose:

1. Assign input values of simulation terrain, curb and barrier data

b. Common Blocks Required:

HEAD, INPT, INPT1, INPT2, NEWCRB, RUFNES

c. Subroutines Required:

TEREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

NX, NY, XB, XE, YB, YE, NBX, NBY, NZ5, SET, AMUB, AMUC, AMUG, CONS, DELG, EPSB, EPSV, GHED, IRUF, NEND, SIGR, YBP0, YC1P, YC2P, YC3P, YC4P, YC5P, YC6P, ZBBP, ZBTP, ZC2P, ZC3P, ZC4P, ZC5P, ZC6P, DGMAX, EPSP, NPAGE, NZTAB, PHIC1, PHIC2, PHIC3, PHIC4, PHIC5, PHIC6, XINCR, YINCR, DELYBP

f. Size:

$746)_{16} = 1862)_{10}$ bytes

8. SUBROUTINE BLK06

a. Purpose:

1. Assign input values of simulation initial conditions

b. Common Blocks Required:

HEAD, INPT, INPT1, INSUS

c. Subroutines Required:

None

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

P0, Q0, R0, U0, V0, W0, PHI0, PSI0, SHED, XCOP, YCOP, ZCOP, DEL10, DEL20, DEL30, DEL40, PHIF0, PHIR0, DEL10D, DEL20D, DEL30D, DEL40D, PHIF0D, PHIR0D, PSIFD0, PSIFI0, THETA0

f. Size:

$318)_{16} = 792)_{10}$ bytes

9. SUBROUTINE CLEAR(A,B)

a. Purpose:

1. To set a block of storage to zero

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

A - beginning address to be cleared
B - end of the full-word address to be cleared

e. Common Variables Calculated:

None

f. Size:

$182)_{16} = 386)_{10}$ bytes

10. SUBROUTINE CNSTNT

a. Purpose:

1. Evaluate program constants
2. Initialize dependent variables and derivatives to input initial conditions

b. Common Blocks Required:

HEAD, INPT, INPT1, INTG, COMP, COMPN, EINDEX, INPT2, TIRIN, BARIER, INSUS, SUSCMP, NEWCRB

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

P, Q, R, U, V, W, AA1, AA2, A12, A23, BB1, BB2, CC1, CC2, RR1, RR2, RTF, RTR, TIZ, TM4, XCP, YCP, ZCP, ZF0, ZPR, ZR0, AMUF, A234, BMUR, DEL1, DEL2, DEL3, DEL4, GAM1, PHIF, PHIN, PHIR, PI02, PI04, PSIN, RFTF, RH02, RRTR, RRTS, SUMM, TF02, TR02, TS02, XCPN, YBPT, YCLP, YCPN, YC3P, YC4P, YC5P, YC6P, ZCPN, ZC3P, ZC4P, ZC5P, ZC6P, DEL1D, DEL2D, DEL3D, DEL4D, PHIFD, PHIRD, PHITL, PHITP, PSIFI, PSITL, PSITP, RHMR2, RH0F2, YBPTP, A02APB, AXMF02, B02APB, BR0MUR, BXMR02, 0MT2A2, 0MT2M1, PHICLR, PHIC1R, PHIC2R, PHIC3R, PHIC4R, PHIC5R, PHIC6R, PSIFID, RF2MFI, RHFMUF, RHF2MF, RHMR2I, RH0MUR, TANPCL, TANPC1, TANPC2, TANPC3, TANPC4, TANPC5, TANPC6, THETTL, THETTP, XMTF04, XMTR04, XMUF02, XMUR02

f. Size:

$A2E)_{16} = 2606)_{10}$ bytes

g. Computational Procedure:

1. Compute program constants
2. Initialize dependent variables converting degrees to radians
3. Initialize XINDT = 10, if THETN or PHIN are not zero for use in MAIN and TMCNST to control coordinate system indexing

11. SUBROUTINE CRBIMP(I)

a. Purpose:

1. Determine the radial tire force and equivalent ground contact point when a tire is in contact with a curb

b. Common Blocks Required:

INPT1, DIMV, COMP, COMPN, INTG, TIRIN, NEWCRB

c. Subroutines Required:

INTRPL

d. Arguments:

The argument I indicates the wheel number for which calculations are made

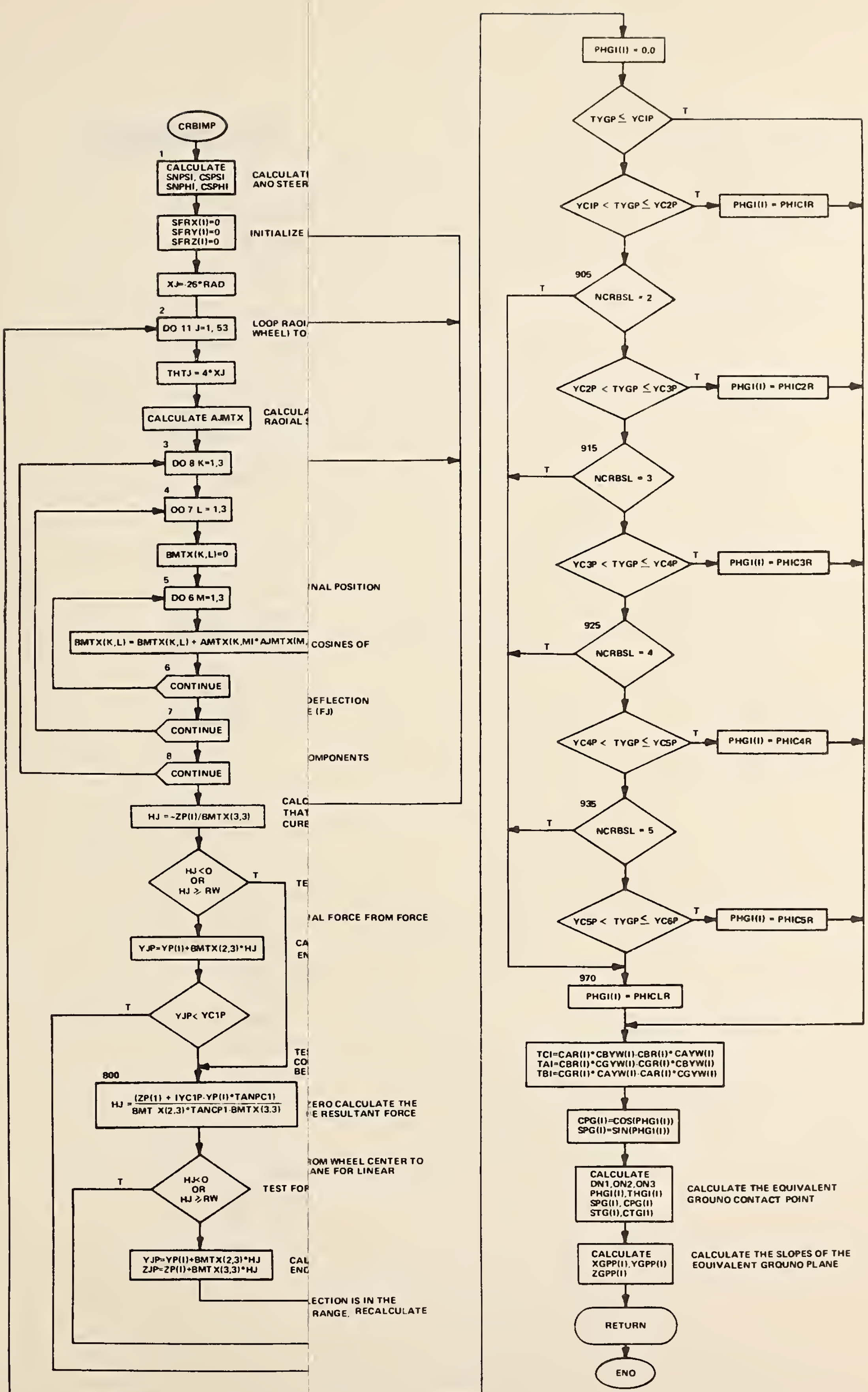
e. Common Variables Calculated:

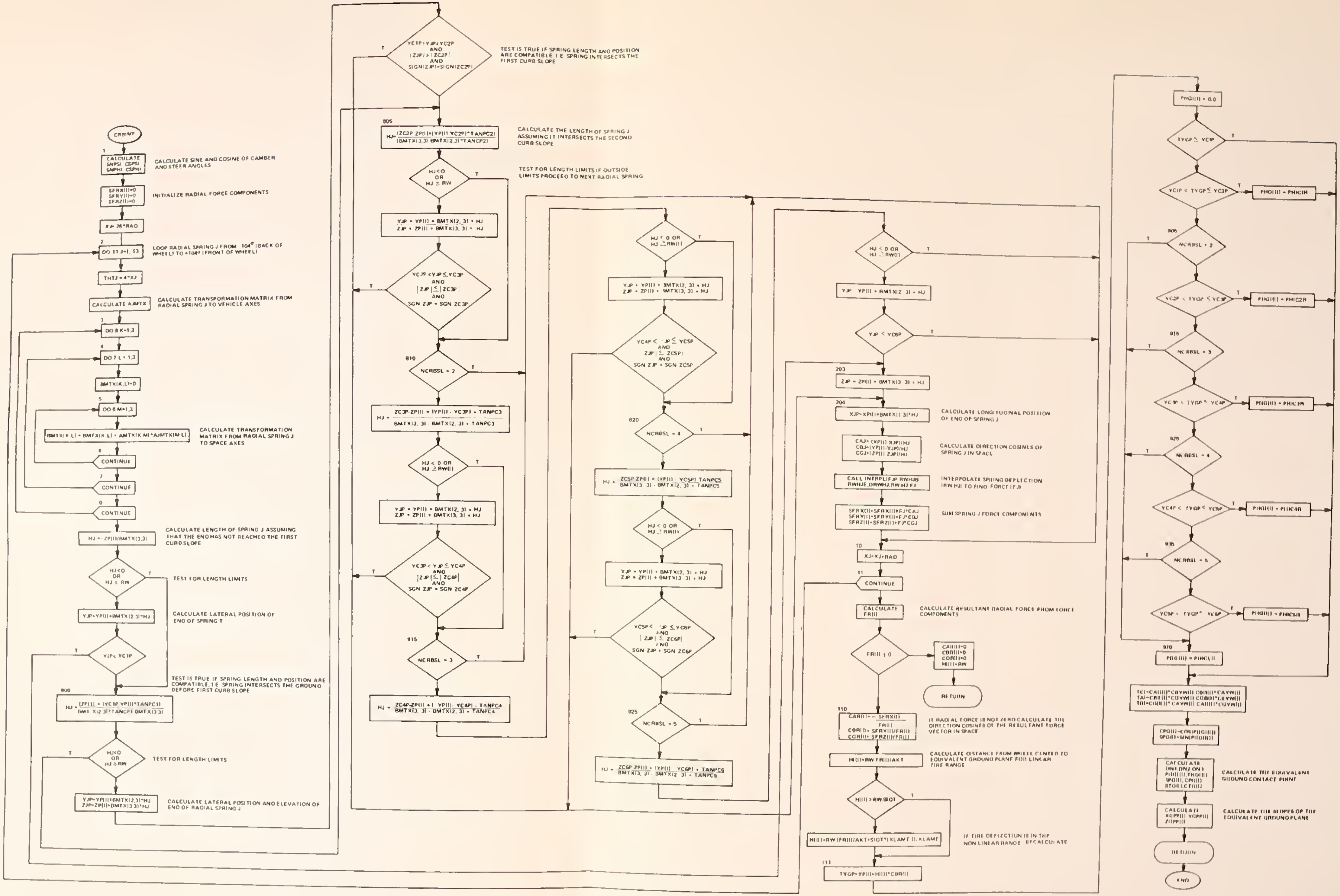
FR, HI, RW, CAR, CBR, CGR, CPG, CTG, SPG, STG, BMTX, PHGI, SFRX, SFRY, SFRZ, THGI, XGPP, YGPP, ZGPP, AJMTX

f. Size:

$F32)_{16} = 3890)_{10}$ bytes

g. Computational Procedure:


CRBIMP
1
CALCULATE SNPSI, CSPSI SNPHI, CSPHI
CALCULATI ANO STEER
SFRX(I)=0 SFRY(I)=0 SFRZ(I)=0
INITIALIZE
XJ=-26*RAD
2
DO 11 J=1, 53
LOOP RAOI WHEEL) TO
THTJ = 4*XJ
CALCULATE AJMTX
CALCULA RAOIAL
3
DO 8 K=1,3
4
OO 7 L = 1,3
BMTX(K,L)=0
5
DO 6 M=1,3
NAL POSITION
BMTX(K,L) = BMTX(K,L) + AMTX(K,MI*AJMTX(M,
COSINES OF
6
CONTINUE
DEFLECTION E (FJ)
7
CONTINUE
8
CONTINUE
OMPONENTS
HJ = -ZP(I)/BMTX(3,3)
CALC THAT CURE
HJ<O OR HJ ≥ RW
T
TE
AL FORCE FROM FORCE
YJP=YP(I)+BMTX(2,3)*HJ
CA EN
T
YJP< YC1P
TE CO BE
800
HJ = (ZP(1) + IYC1P-YP(I)*TANPC1) / BMT X(2,3)*TANCP1-BMTX(3,3)
ZERO CALCULATE THE E RESULTANT FORCE
ROM WHEEL CENTER TO ANE FOR LINEAR
T
HJ<O OR HJ ≥ RW
TEST FOR
YJP=YP(I)+BMTX(2,3)*HJ ZJP=ZP(I)+BMT X(3,3)*HJ
CAL ENC
ECTION IS IN THE RANGE. RECALCULATE
PHGI(I) = 0.0
TYGP ≤ YCIP
T
YCIP < TYGP ≤ YC2P
T
PHGI(I) = PHICIR
905
T
NCRBSL = 2
YC2P < TYGP ≤ YC3P
T
PHGI(I) = PHIC2R
915
T
NCRBSL = 3
YC3P < TYGP ≤ YC4P
T
PHGI(I) = PHIC3R
925
T
NCRBSL = 4
YC4P < TYGP ≤ YC5P
T
PHGI(I) = PHIC4R
935
T
NCRBSL = 5
YC5P < TYGP ≤ YC6P
T
PHGI(I) = PHIC5R
970
PHGI(I) = PHICLR
TCI=CAR(I)*CBYW(I)-CBR(I)*CAYW(I)
TAI=CBR(I)*CGYW(I)-CGR(I)*CBYW(I)
TBI=CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
CPG(I)=COS(PHGI(I))
SPG(I)=SIN(PHGI(I))
CALCULATE DN1,ON2,ON3 PHGI(I),THGI(I) SPG(I), CPG(I) STG(I),CTG(I)
CALCULATE THE EQUIVALENT GROUNO CONTACT POINT
CALCULATE XGPP(I),YGPP(I) ZGPP(I)
CALCULATE THE SLOPES OF THE EOUIVALENT GROUNO PLANE
RETURN
ENO


CRBIMP
CALCULATE SINE AND COSINE OF CAMBER AND STEER ANGLES
INITIALIZE RADIAL FORCE COMPONENTS
LOOP RADIAL SPRING J FROM -104° (BACK OF WHEEL) TO +104° (FRONT OF WHEEL)
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING J TO VEHICLE AXES
CALCULATE AJMTX
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING J TO SPACE AXES
CONTINUE
CALCULATE LENGTH OF SPRING J ASSUMING THAT THE END HAS NOT REACHED THE FIRST CURB SLOPE
TEST FOR LENGTH LIMITS
CALCULATE LATERAL POSITION OF END OF SPRING T
TEST IS TRUE IF SPRING LENGTH AND POSITION ARE COMPATIBLE, I.E. SPRING INTERSECTS THE GROUND BEFORE FIRST CURB SLOPE
CALCULATE LATERAL POSITION AND ELEVATION OF END OF RADIAL SPRING J
TEST IS TRUE IF SPRING LENGTH AND POSITION ARE COMPATIBLE, I.E. SPRING INTERSECTS THE FIRST CURB SLOPE
CALCULATE THE LENGTH OF SPRING J ASSUMING IT INTERSECTS THE SECOND CURB SLOPE
TEST FOR LENGTH LIMITS IF OUTSIDE LIMITS PROCEED TO NEXT RADIAL SPRING
NCRBSL = 2
NCRBSL = 3
NCRBSL = 4
NCRBSL = 5
CALCULATE LONGITUDINAL POSITION OF END OF SPRING J
CALCULATE DIRECTION COSINES OF SPRING J IN SPACE
INTERPOLATE SPRING DEFLECTION (RW-HJ) TO FIND FORCE (FJ)
SUM SPRING J FORCE COMPONENTS
CALCULATE RESULTANT RADIAL FORCE FROM FORCE COMPONENTS
RETURN
IF RADIAL FORCE IS NOT ZERO CALCULATE THE DIRECTION COSINES OF THE RESULTANT FORCE VECTOR IN SPACE
CALCULATE DISTANCE FROM WHEEL CENTER TO EQUIVALENT GROUND PLANE FOR LINEAR TIRE RANGE
IF TIRE DEFLECTION IS IN THE NON LINEAR RANGE RECALCULATE
CALCULATE THE EQUIVALENT GROUND CONTACT POINT
CALCULATE THE SLOPES OF THE EQUIVALENT GROUND PLANE
RETURN
END

SUBROUTINE DATE

a. Purpose:

1. Return the calender date in 8 byte form, e.g., 23MAR'68

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

DADE - Array into which the date is loaded

e. Common Variables Calculated:

None

f. Size:

$D6)_{16} = 214)_{10}$ bytes

g. Procedure:

This subroutine is written in IBM S/360 Assembler Language

13. SUBROUTINE DAUX

a. Purpose:

1. Evaluate the derivatives of the dependent variables for subsequent integration in PINT1

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, INPT2, INSUS, SUSCMP, NSTOP

c. Subroutines Required:

VPOS, VGORNT, TMCNST, MATRIX, SIMSOL, MTRXIR, MTRXSF, SFORCE, SUSFRC, UMOMNT

d. Arguments:

None

e. Common Variables Calculated:

DP, DQ, DR, DU, DV, DW, DXCP, DYCP, DZCP, GAM2, GAM3, GAM4, GAM5, GAM6, GAM7, GAM8, GAM9, XIYP, XIXP, XIZP, DDEL1, DDEL2, DDEL3, DDEL4, DMATX, DPHIF, DPHIR, T1PSI, T2PSI, XIXZP, XIYZP, DDEL1D, DDEL2D, DDEL3D, DDEL4D, DDPSFI, DPHIFD, DPHIRD, DPHITP, DPSIFI, DPSITP, DTHTTP

f. Size:

$AD4)_{16} = 2772)_{10}$ bytes

g. Computational Procedure:

1. Test for abnormal program stop (ISTOP $\neq$ 0) and return if indicated.
2. Calculate time dependent variables by calling subroutine TMCNST.
3. Calculate time dependent inertial terms: XIXP, XIYP, XIZP, XIXZP, XIYZP, GAM2, GAM3, GAM4, GAM5, GAM6, GAM7, GAM8, GAM9. Note that these variables differ with the suspension option in effect, thus branching to the appropriate set of calculation occurs based in ISUS.
4. Call subroutines VPOS and VGORNT to determine the position and orientation of the vehicle.
5. Calculate suspension displacements and velocities depending on suspension option.
6. Call subroutines SUSFRC to calculate suspension forces, and UMOMNT to calculate moments acting on the sprung mass and solid axles (if being used).

7. If the barrier option is being used ($INDB \neq 0$) call subroutine SFORCE to obtain impact forces and moments.
8. Depending on the suspension option in effect, call either subroutine MATRIX, MATRXIR or MTRXSF to evaluate the inertial matrix and forcing function stored in the array DMATX.
9. Call subroutine SIMSOL to solve the 10x10 set of simultaneous equations of motion for the 10 derivatives of the dependent variables.
10. Set the solution vector from SIMSOL, DMATX(I,11), to the appropriate variable names and set the remaining 10 derivatives depending on suspension option.
11. Compute the derivatives of the steering degree-of-freedom if in effect as indicated by either $INDCRB<0$ or $IHIT=1$ and $INDCRB>0$.

14. SUBROUTINE DRIVER(SA,SADOT, ISA)

a. Purpose:

1. DRIVER is a dummy subroutine included to provide linkages for an automatic steering algorithm and provides no other function in this program version.

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

SA - front wheel steer angle
SADOT - front wheel steer angle velocity
ISA - driver option indicator

e. Common Variables Calculated:

None

f. Size:

$E0)_{16} = 224)_{10}$ bytes

g. Computational Procedure:

1. This subroutine sets the driver indicator argument (ISA) to zero insuring this option is not used.

15. SUBROUTINE DRIVID

a. Purpose:

1. DRIVID is a dummy subroutine to provide linkages in order to print automatic steering control inputs. It provides no function in this program version.

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$A4)_{16} = 164)_{10}$ bytes

16. SUBROUTINE DVDCHK

a. Purpose:

1. This subroutine processes interuptions caused by arithmetic instructions

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$452)_{16} = 1108)_{10}$ bytes

g. Procedure:

1. A call to DVDCHK processes the following interuptions:

   1. fixed point divide exception
   2. exponent overflow exception
   3. exponent underflow exception
   4. floating point divide exception

   This subroutine is written in IBM Assembler Language. The services provided are also given by extended FORTRAN error handling.

17. SUBROUTINE EGYSUM

a. Purpose:

1. Calculate energy absorbed by the vehicle and barrier during a barrier impact
2. Set previous values of some barrier impact variables

b. Common Blocks Required:

INPT2, BARIER

c. Subroutines Required:

None

d. Arguments:

None

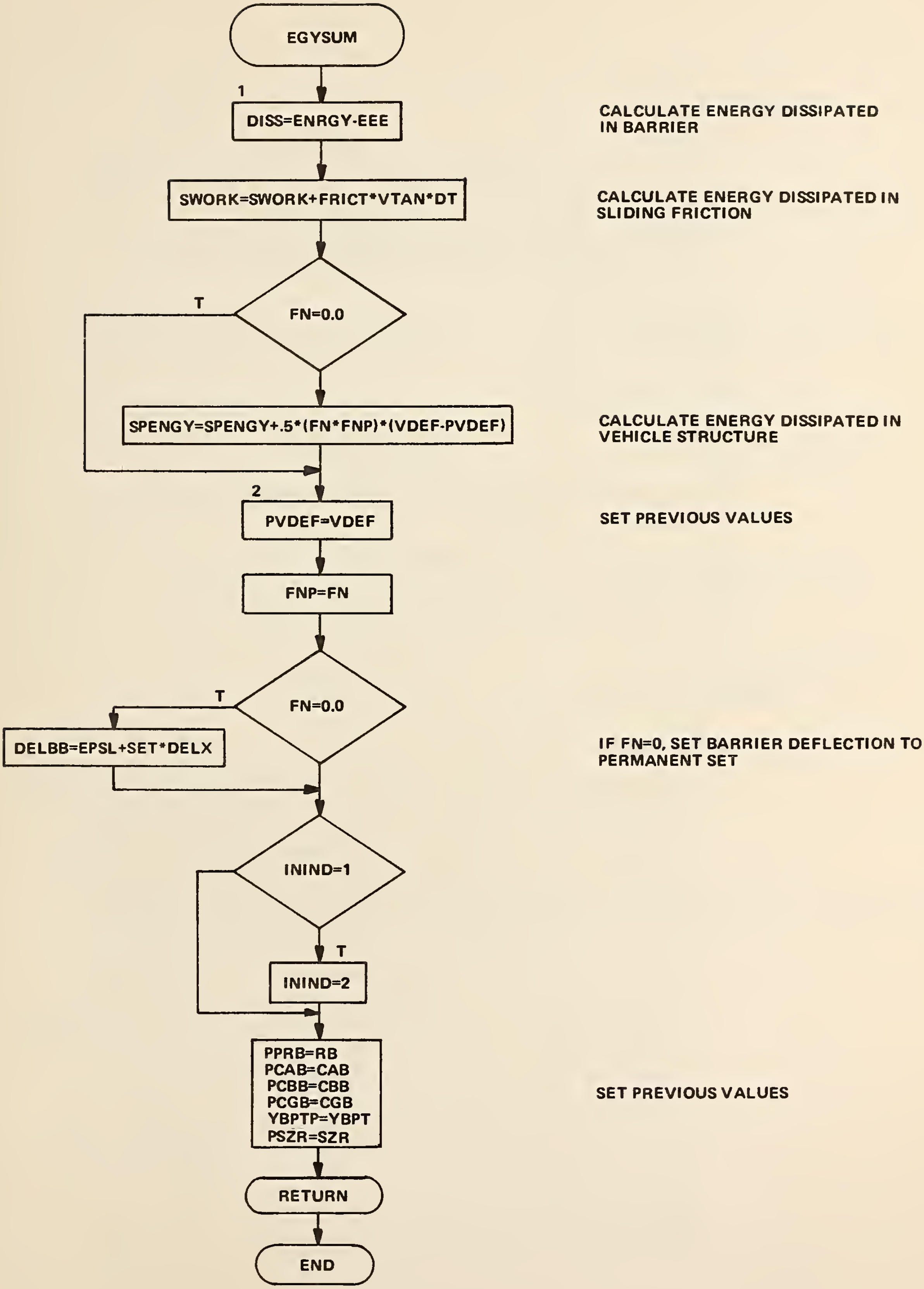
e. Common Variables Calculated:

FNP, DISS, PCAB, PCBB, PCGB, PPRB, PSZR, DELBB, ININD, PVDEF, SWORK, YBPTP, DELBBP, SPENGY

f. Size:

$18E)_{16} = 398)_{10}$ bytes

g. Computational Procedure:


EGYSUM
1
DISS=ENRGY-EEE
CALCULATE ENERGY DISSIPATED IN BARRIER
SWORK=SWORK+FRICT*VTAN*DT
CALCULATE ENERGY DISSIPATED IN SLIDING FRICTION
T
FN=0.0
SPENGY=SPENGY+.5*(FN*FNP)*(VDEF-PVDEF)
CALCULATE ENERGY DISSIPATED IN VEHICLE STRUCTURE
2
PVDEF=VDEF
SET PREVIOUS VALUES
FNP=FN
T
FN=0.0
DELBB=EPSL+SET*DELX
IF FN=0, SET BARRIER DEFLECTION TO PERMANENT SET
ININD=1
T
ININD=2
PPRB=RB
PCAB=CAB
PCBB=CBB
PCGB=CGB
YBPTP=YBPT
PSZR=SZR
SET PREVIOUS VALUES
RETURN
END

18. SUBROUTINE GCP(I)

a. Purpose:

1. Compute the coordinates of the tire ground contact point in space
2. Compute the rolling radius of the tire
3. Compute the direction and magnitude of the tire radial force

b. Common Blocks Required:

DIMV, COMP, TIRIN

c. Subroutines Required:

SIMSOL

d. Arguments:

The argument I indicates the wheel number for which calculations are made

e. Common Variables Calculated:

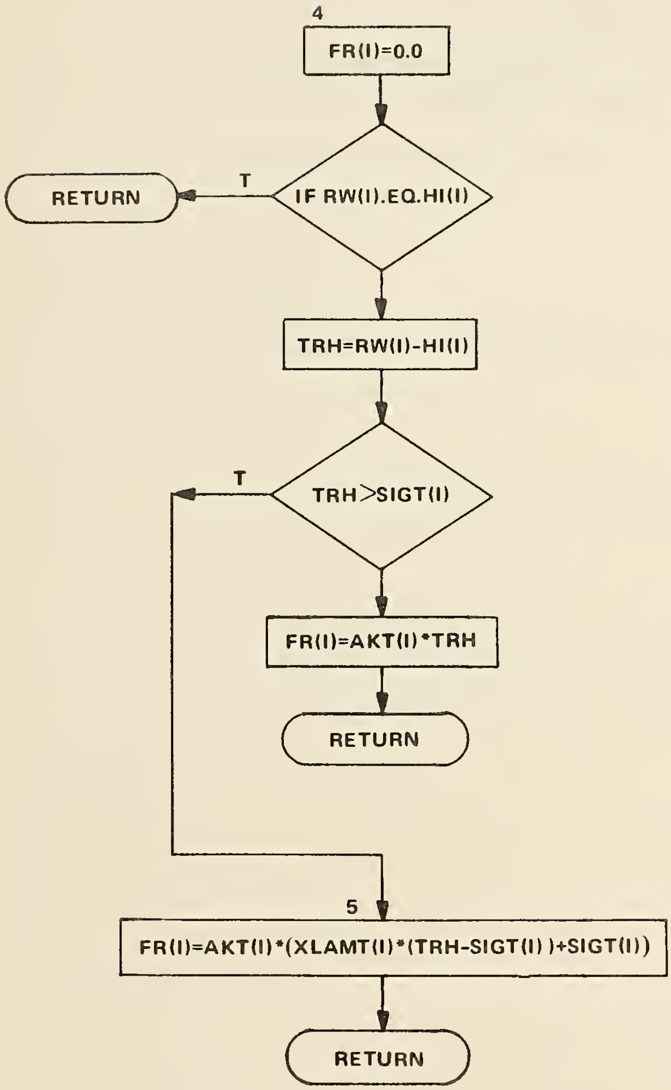
FR, HI, TX, TY, TZ, CAR, CBR, CGR, TRH, CMTX, XGPP, XLM1, XLM2, XLM3, YGPP, ZGPP, DELTA

f. Size:

$36C)_{16} = 876)_{10}$ bytes

g. Computational Procedure:

1. Calculate the coordinates of the ground contact point by simultaneous solution of the intersection of three planes: the wheel plane (normal direction CAYW(I), CBYW(I), CGYW(I)); the ground plane (normal direction CAGZ(I), CBGZ(I), CGGZ(I); and a plane perpendicular to both passing through the wheel center (normal direction D1(I), D2(I), D3(I)). The simultaneous solution is performed by SIMSOL with the CMTX array containing the above direction cosines and the target array (XLM1(I), XLM2(I), XLM3(I)) contained in the fourth column of CMTX. The solution is returned in the fourth column of CMTX and set to the coordinates of the ground contact point (XGPP(I), YGPP(I), ZGPP(I)).
2. Calculate the distance between the wheel center and ground contact point, DELTA(I).
3. Calculate the direction cosines of the line of action of the tire radial force with respect to the space axes (CAR(I), CBR(I), CGR(I)).
4. Determine the rolling radius, HI(I).
5. Calculate the radial tire force, FR(I), as shown:



INITIALIZE RADIAL FORCE TO ZERO

TEST IS TRUE IF TIRE IS NOT DEFLECTED FR(I) IS ZERO

CALCULATE TIRE DEFLECTION

TEST IS TRUE IF DEFLECTION IS NON-LINEAR PART OF TIRE

FOR LINEAR PORTION OF TIRE CALCULATE FORCE

CALCULATE FORCE FOR NON-LINEAR PART OF TIRE

19. SUBROUTINE IDOUT

a. Purpose:

1. Print input values with units and headings

b. Common Blocks Required:

HEAD, INPT, INPT1, COMP, INPT2, INPT3, APTABL, TIRIN, INSUS, NEWCRB, BARSTR

c. Subroutines Required:

DRIVID

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$47D0)_{16} = 18384)_{10}$ bytes

20. SUBROUTINE INITEQ

a. Purpose:

1. To perform calculations to situate the vehicle in initial vertical equilibrium on flat, level terrain

b. Common Blocks Required:

INPT, COMP, DIMV, COMPN, INSUS, TIRIN

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

FR, HI, ZF, ZR

f. Size:

$324)_{16} = 804)_{10}$ bytes

g. Computational Procedure:

If ZF and ZR are input as zero, this subroutine calculates these variables based on the requirement for initial vertical equilibrium of the vehicle. Also calculated are tire radial forces and rolling radii.

21. SUBROUTINE INPUT

a. Purpose:

1. Obtain card input
2. Print card images

b. Common Blocks Required:

None

c. Subroutines Required:

BLK01, BLK02, BLK03, BLK04, BLK05, BLK06

d. Arguments:

None

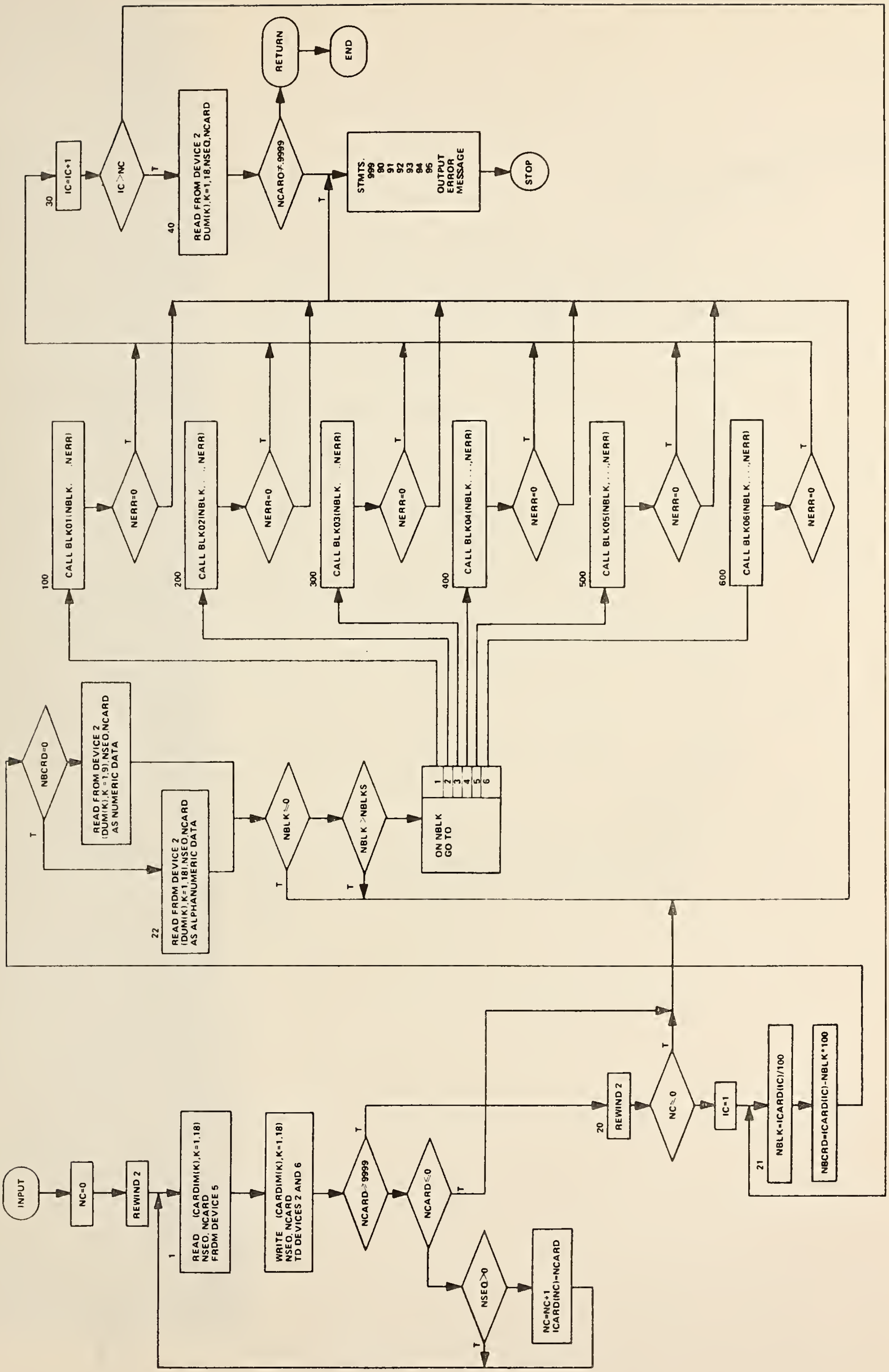
e. Common Variables Calculated:

None

f. Size:

$D5E)_{16} = 3422)_{10}$ bytes

g. Computational Procedure:



INPUT
NC=0
REWIND 2
1
READ (CARD(K),K=1,18)
NSEQ, NCARD
FROM DEVICE 5
WRITE (CARD(K),K=1,18)
NSEQ, NCARD
TO DEVICES 2 AND 6
NCARD≥9999
T
NCARD≤0
T
NSEQ>0
T
NC=NC+1
ICARD(NC)=NCARD
20
REWIND 2
NC≤0
T
IC=1
21
NBLK=ICARD(IC)/100
NBCRD=ICARD(IC)-NBLK*100
NBCRD=0
T
READ FROM DEVICE 2
(DUM(K),K=1,9),NSEQ,NCARD
AS NUMERIC DATA
22
READ FROM DEVICE 2
(DUM(K),K=1,18),NSEQ,NCARD
AS ALPHANUMERIC DATA
NBLK≤0
T
NBLK>NBLKS
T
ON NBLK
GO TO
1
2
3
4
5
6
100
CALL BLK01(NBLK, . . .,NERR)
NERR=0
T
200
CALL BLK02(NBLK, . . .,NERR)
NERR=0
T
300
CALL BLK03(NBLK, . . .,NERR)
NERR=0
T
400
CALL BLK04(NBLK, . . .,NERR)
NERR=0
T
500
CALL BLK05(NBLK, . . .,NERR)
NERR=0
T
600
CALL BLK06(NBLK, . . .,NERR)
NERR=0
T
30
IC=IC+1
IC>NC
T
40
READ FROM DEVICE 2
DUM(K),K=1,18,NSEQ,NCARD
NCARD≠9999
T
RETURN
END
STMTS.
999
90
91
92
93
94
95
OUTPUT
ERROR
MESSAGE
STOP

22. SUBROUTINE INTRPL

a. Purpose

1. To obtain a quadratic interpolation of a one-dimensional table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

TABLE - one-dimensional array of data
XMIN - minimum abscissa value
XMAX - maximum abscissa value
DX - abscissa increment
X - abscissa value at which ordinate is desired
Y - ordinate of X

e. Common Variables Calculated:

None

f. Size:

$4A2)_{16} = 1186)_{10}$ bytes

g. Procedure:

1. Quadratic interpolation of the values of TABLE at X
2. ENTRY INTRPC also includes the additional argument SLOPE which is calculated as $\frac{d(TABLE)}{dx}$ at X

23. SUBROUTINE INTRP5(INDX)

a. Purpose:

1. Calculate the elevation and slopes under the wheel indicated by the argument INDX.
2. Set the nominal friction coefficient according to the table for the wheel location.

b. Common Blocks Required:

INPT, DIMV, COMP, COMPN, TIRIN

c. Subroutines Required:

None

d. Arguments:

INDX - wheel number for which calculations are to be made

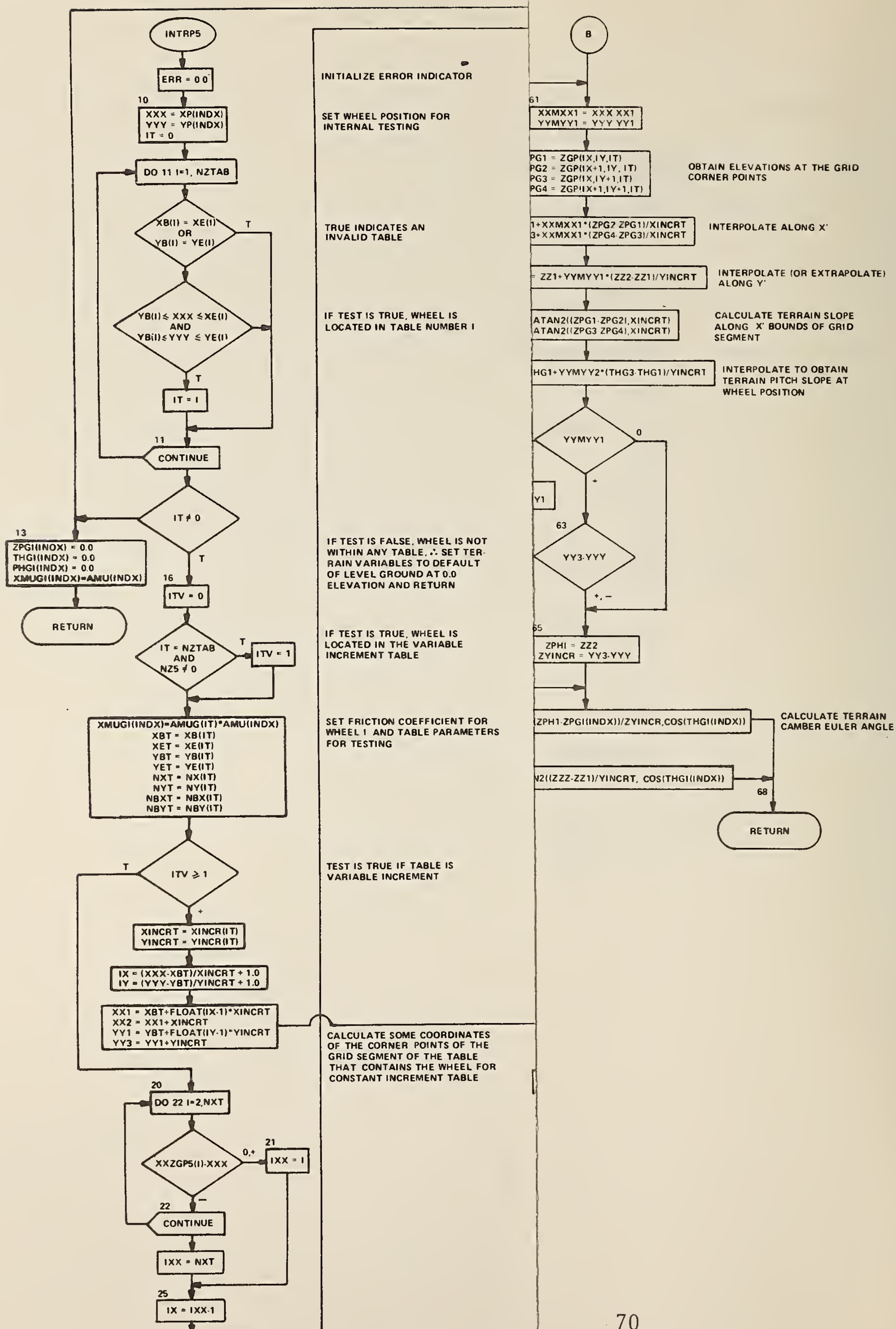
e. Common Variables Calculated:

IX, IY, XXX, XX1, XX2, YYY, YY1, YY2, ZZ1, ZZ2, PHGI, THGI, THG1, ZPGI, XMUGI

f. Size:

$129C)_{16} = 4764)_{10}$ bytes

g. Computational Procedure:


INTRP5
ERR = 0.0
INITIALIZE ERROR INDICATOR
10
XXX = XP(INDX)
YYY = YP(INDX)
IT = 0
SET WHEEL POSITION FOR INTERNAL TESTING
DO 11 I=1, NZTAB
XB(I) = XE(I) OR YB(I) = YE(I)
T
TRUE INDICATES AN INVALID TABLE
YB(I) ≤ XXX ≤ XE(I) AND YB(I) ≤ YYY ≤ YE(I)
IF TEST IS TRUE, WHEEL IS LOCATED IN TABLE NUMBER I
T
IT = I
11
CONTINUE
IT ≠ 0
13
ZPGI(INDX) = 0.0
THGI(INDX) = 0.0
PHGI(INDX) = 0.0
XMUGI(INDX)=AMU(INDX)
RETURN
T
IF TEST IS FALSE, WHEEL IS NOT WITHIN ANY TABLE, ∴ SET TERRAIN VARIABLES TO DEFAULT OF LEVEL GROUND AT 0.0 ELEVATION AND RETURN
16
ITV = 0
IT = NZTAB AND NZ5 ≠ 0
T
ITV = 1
IF TEST IS TRUE, WHEEL IS LOCATED IN THE VARIABLE INCREMENT TABLE
XMUGI(INDX)=AMUG(IT)*AMU(INDX)
XBT = XB(IT)
XET = XE(IT)
YBT = YB(IT)
YET = YE(IT)
NXT = NX(IT)
NYT = NY(IT)
NBXT = NBX(IT)
NBYT = NBY(IT)
SET FRICTION COEFFICIENT FOR WHEEL I AND TABLE PARAMETERS FOR TESTING
T
ITV ≥ 1
+
TEST IS TRUE IF TABLE IS VARIABLE INCREMENT
XINCRT = XINCR(IT)
YINCRT = YINCR(IT)
IX = (XXX-XBT)/XINCRT + 1.0
IY = (YYY-YBT)/YINCRT + 1.0
XX1 = XBT+FLOAT(IX-1)*XINCRT
XX2 = XX1+XINCRT
YY1 = YBT+FLOAT(IY-1)*YINCRT
YY3 = YY1+YINCRT
CALCULATE SOME COORDINATES OF THE CORNER POINTS OF THE GRID SEGMENT OF THE TABLE THAT CONTAINS THE WHEEL FOR CONSTANT INCREMENT TABLE
20
DO 22 I=2,NXT
XXZGP5(I)-XXX
0,+
21
IXX = I
−
22
CONTINUE
IXX = NXT
25
IX = IXX-1
B
61
XXMXX1 = XXX-XX1
YYMYY1 = YYY-YY1
PG1 = ZGP(IX,IY,IT)
PG2 = ZGP(IX+1,IY,IT)
PG3 = ZGP(IX,IY+1,IT)
PG4 = ZGP(IX+1,IY+1,IT)
OBTAIN ELEVATIONS AT THE GRID CORNER POINTS
1+XXMXX1*(ZPG2-ZPG1)/XINCRT
3+XXMXX1*(ZPG4-ZPG3)/XINCRT
INTERPOLATE ALONG X'
= ZZ1+YYMYY1*(ZZ2-ZZ1)/YINCRT
INTERPOLATE (OR EXTRAPOLATE) ALONG Y'
ATAN2((ZPG1-ZPG2),XINCRT)
ATAN2((ZPG3-ZPG4),XINCRT)
CALCULATE TERRAIN SLOPE ALONG X' BOUNDS OF GRID SEGMENT
HG1+YYMYY2*(THG3-THG1)/YINCRT
INTERPOLATE TO OBTAIN TERRAIN PITCH SLOPE AT WHEEL POSITION
YYMYY1
0
+
Y1
63
YY3-YYY
+, −
65
ZPHI = ZZ2
ZYINCR = YY3-YYY
(ZPH1-ZPGI(INDX))/ZYINCR,COS(THGI(INDX))
CALCULATE TERRAIN CAMBER EULER ANGLE
N2((ZZ2-ZZ1)/YINCRT, COS(THGI(INDX))
68
RETURN

24. SUBROUTINE MATRIX

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the independent front/solid axle rear suspension option
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1,10) for the independent front/solid rear axle option

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

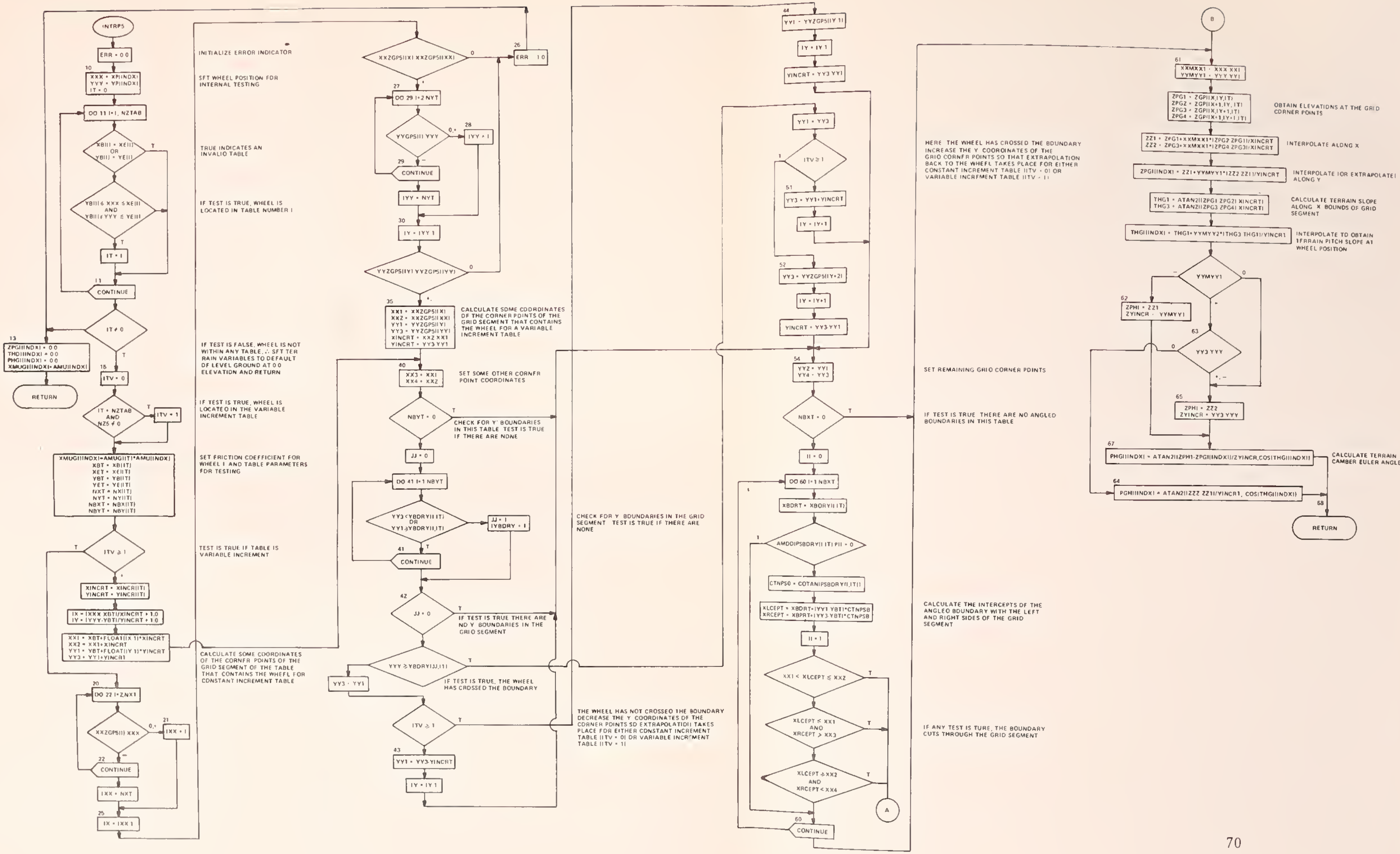
DMATX, GCTCP, GCTSP

f. Size:

$72C)_{16} = 1836)_{10}$

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero.
2. Calculate the elements of DMATX.



INTRP5
ERR = 0.0
INITIALIZE ERROR INDICATOR
XXX = XP(INDX)
YYY = YP(INDX)
IT = 0
SET WHEEL POSITION FOR INTERNAL TESTING
DO 11 I=1, NZTAB
TRUE INDICATES AN INVALID TABLE
IF TEST IS TRUE, WHEEL IS LOCATED IN TABLE NUMBER I
IT = I
CONTINUE
IT ≠ 0
IF TEST IS FALSE, WHEEL IS NOT WITHIN ANY TABLE, ∴ SET TERRAIN VARIABLES TO DEFAULT OF LEVEL GROUND AT 0.0 ELEVATION AND RETURN
RETURN
ITV = 0
IF TEST IS TRUE, WHEEL IS LOCATED IN THE VARIABLE INCREMENT TABLE
ITV = 1
SET FRICTION COEFFICIENT FOR WHEEL I AND TABLE PARAMETERS FOR TESTING
TEST IS TRUE IF TABLE IS VARIABLE INCREMENT
CALCULATE SOME COORDINATES OF THE CORNER POINTS OF THE GRID SEGMENT OF THE TABLE THAT CONTAINS THE WHEEL FOR CONSTANT INCREMENT TABLE
CONTINUE
ERR = 1.0
CONTINUE
CALCULATE SOME COORDINATES OF THE CORNER POINTS OF THE GRID SEGMENT THAT CONTAINS THE WHEEL FOR A VARIABLE INCREMENT TABLE
SET SOME OTHER CORNER POINT COORDINATES
CHECK FOR Y BOUNDARIES IN THIS TABLE. TEST IS TRUE IF THERE ARE NONE
CONTINUE
IF TEST IS TRUE THERE ARE NO Y BOUNDARIES IN THE GRID SEGMENT
IF TEST IS TRUE, THE WHEEL HAS CROSSED THE BOUNDARY
CHECK FOR Y BOUNDARIES IN THE GRID SEGMENT. TEST IS TRUE IF THERE ARE NONE
THE WHEEL HAS NOT CROSSED THE BOUNDARY. DECREASE THE Y COORDINATES OF THE CORNER POINTS SO EXTRAPOLATION TAKES PLACE FOR EITHER CONSTANT INCREMENT TABLE (ITV = 0) OR VARIABLE INCREMENT TABLE (ITV = 1)
HERE THE WHEEL HAS CROSSED THE BOUNDARY. INCREASE THE Y COORDINATES OF THE GRID CORNER POINTS SO THAT EXTRAPOLATION BACK TO THE WHEEL TAKES PLACE FOR EITHER CONSTANT INCREMENT TABLE (ITV = 0) OR VARIABLE INCREMENT TABLE (ITV = 1)
SET REMAINING GRID CORNER POINTS
IF TEST IS TRUE THERE ARE NO ANGLED BOUNDARIES IN THIS TABLE
CALCULATE THE INTERCEPTS OF THE ANGLED BOUNDARY WITH THE LEFT AND RIGHT SIDES OF THE GRID SEGMENT
IF ANY TEST IS TURE, THE BOUNDARY CUTS THROUGH THE GRID SEGMENT
CONTINUE
OBTAIN ELEVATIONS AT THE GRID CORNER POINTS
INTERPOLATE ALONG X
INTERPOLATE (OR EXTRAPOLATE) ALONG Y
CALCULATE TERRAIN SLOPE ALONG X BOUNDS OF GRID SEGMENT
INTERPOLATE TO OBTAIN TERRAIN PITCH SLOPE AT WHEEL POSITION
CALCULATE TERRAIN CAMBER EULER ANGLE
RETURN

25. SUBROUTINE MTRXIR

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the independent rear suspension option.
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1, 10) for the independent rear suspension option.

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL, SUSCMP

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

DMATX, GCTCP, GCTSP

f. Size:

$4C0)_{16} = 1216)_{10}$ bytes

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero
2. Calculate the elements of DMATX

26. SUBROUTINE MTRXSF

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the solid front axle option
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1,10) for the solid front axle option

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL, SUSCMP, INSUS

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

DMATX, GCTCP, GCTSP

f. Size:

$962)_{16} = 2402)_{10}$ bytes

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero
2. Calculate the elements of DMATX

27. SUBROUTINE NLDFL

a. Purpose:

1. Calculate barrier force due to deflection and velocity of deflection
2. Calculate energy absorbed in barrier deformation

b. Common Blocks Required:

INTG, INPT2, BARIER

c. Subroutines Required:

POLY

d. Arguments:

None

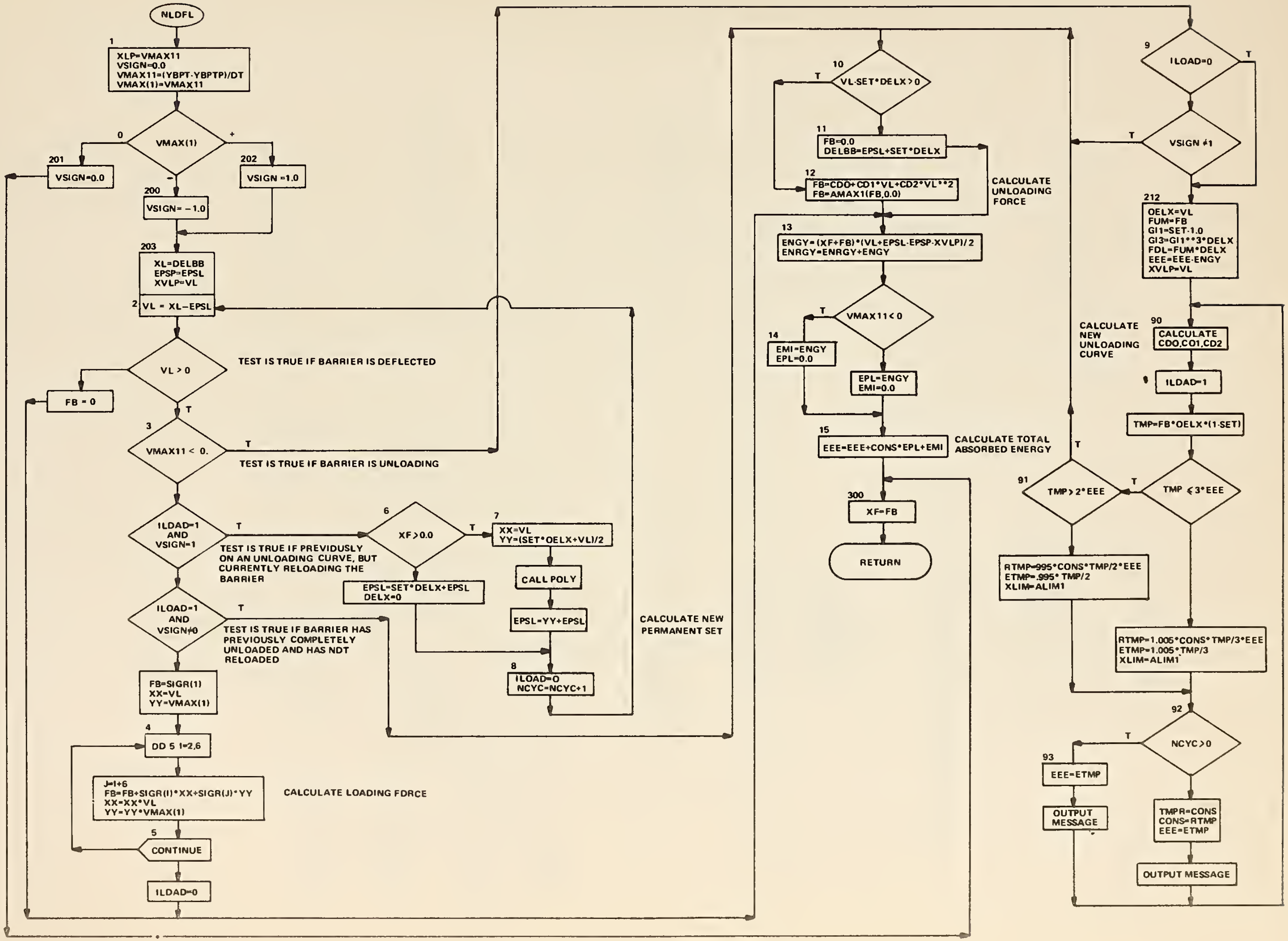
e. Common Variables Calculated:

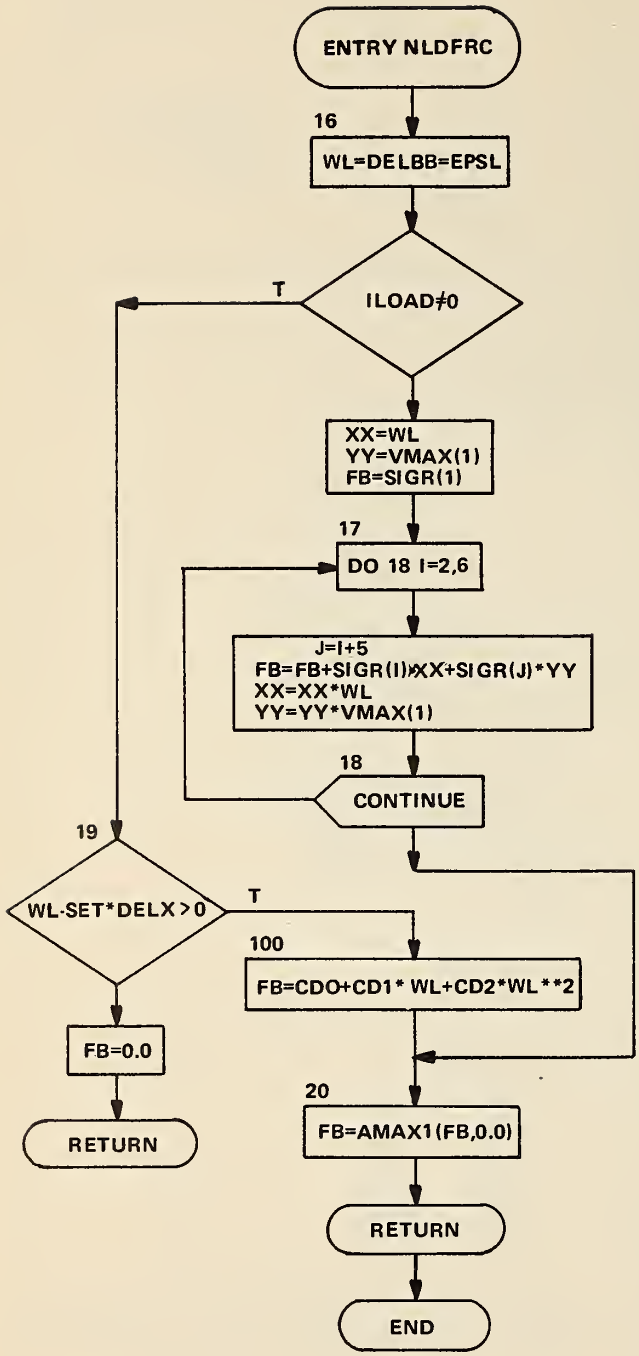
FB, VL, SF, XF, EEE, CONS, DELX, EPSL, NCYC, VMAX, DELBB, ENRGY

f. Size:

$7FC)_{16} = 2044)_{10}$ bytes

g. Computational Procedure:


NLDFL
1
XLP=VMAX11
VSIGN=0.0
VMAX11=(YBPT-YBPTP)/DT
VMAX(1)=VMAX11
VMAX(1)
0
+
-
201
VSIGN=0.0
202
VSIGN =1.0
200
VSIGN= −1.0
203
XL=DELBB
EPSP=EPSL
XVLP=VL
2
VL = XL−EPSL
VL > 0
TEST IS TRUE IF BARRIER IS DEFLECTED
FB = 0
T
3
VMAX11 < 0.
T
TEST IS TRUE IF BARRIER IS UNLOADING
ILDAD=1 AND VSIGN=1
T
TEST IS TRUE IF PREVIDUSLY ON AN UNLOADING CURVE, BUT CURRENTLY RELOADING THE BARRIER
6
XF > 0.0
T
7
XX=VL
YY=(SET*OELX+VL)/2
CALL POLY
EPSL=YY+EPSL
EPSL=SET*DELX+EPSL
DELX=0
8
ILOAD=0
NCYC=NCYC+1
CALCULATE NEW PERMANENT SET
ILOAD=1 AND VSIGN≠0
T
TEST IS TRUE IF BARRIER HAS PREVIOUSLY COMPLETELY UNLOADED AND HAS NDT RELOADED
FB=SIGR(1)
XX=VL
YY=VMAX(1)
4
DD 5 I=2,6
J=I+6
FB=FB+SIGR(I)*XX+SIGR(J)*YY
XX=XX*VL
YY=YY*VMAX(1)
CALCULATE LOADING FDRCE
5
CONTINUE
ILDAD=0
10
VL-SET*DELX > 0
T
11
FB=0.0
DELBB=EPSL+SET*DELX
12
FB=CDO+CD1*VL+CD2*VL**2
FB=AMAX1(FB,0.0)
CALCULATE UNLOADING FORCE
13
ENGY=(XF+FB)*(VL+EPSL-EPSP-XVLP)/2
ENRGY=ENRGY+ENGY
VMAX11< 0
T
14
EMI=ENGY
EPL=0.0
EPL=ENGY
EMI=0.0
15
EEE=EEE+CONS*EPL+EMI
CALCULATE TOTAL ABSORBED ENERGY
300
XF=FB
RETURN
9
ILOAD=0
T
VSIGN ≠1
T
212
OELX=VL
FUM=FB
GI1=SET-1.0
GI3=GI1**3*DELX
FDL=FUM*DELX
EEE=EEE-ENGY
XVLP=VL
CALCULATE NEW UNLOADING CURVE
90
CALCULATE CDO,CO1,CD2
ILDAD=1
TMP=FB*OELX*(1-SET)
TMP ≤ 3*EEE
T
91
TMP > 2*EEE
T
RTMP=.995*CONS*TMP/2*EEE
ETMP=.995* TMP/2
XLIM=ALIM1
RTMP=1.005*CONS*TMP/3*EEE
ETMP=1.005*TMP/3
XLIM=ALIM1
92
NCYC > 0
T
93
EEE=ETMP
OUTPUT MESSAGE
TMPR=CONS
CONS=RTMP
EEE=ETMP
OUTPUT MESSAGE



CALCULATE LOADING FORCE

CALCULATE UNLOADING FORCE

28. SUBROUTINE OUTPUT(IND)

a. Purpose:

1. Print output page titles and output data

b. Common Blocks Required:

HEAD, INPT, INTG, DIMV, COMP, COMPN, ADTNL, INSUS, SUSCMP, BARSTR

c. Subroutines Required:

None

d. Arguments:

If IND = 0, an output line counter is initialized to zero.

e. Common Variables Calculated:

None

f. Size:

$3EBC)_{16} = 16060)_{10}$ bytes

g. Computational Procedure:

Each time a call to this subroutine is executed, an output line of data is written to FORTRAN devices 11 through, at most, 29. The number of devices actually written to is dependent on the indicators contained in the NPAGE array. These indicators are set either by the user on input card 104 or by the program depending on the options in use.

On either the first call to the subroutine with IND $\neq$ 0 or after 50 lines of data have been written, page headings are written for each page of data.

An entry point, THPLOT, is provided to write static and dynamic data to FORTRAN device 3 for the purpose of subsequent plotting of time history data.

29. SUBROUTINE PINT1(IN, MODE, N, X, H, Y, YP, A)

a. Purpose:

1. To integrate a system of N ordinary differential equations of the first order

b. Common Blocks Required:

None

c. Subroutines Required:

DAUX

d. Arguments:

IN is the control word (= 1 or 2) for initialization or to integrate one step-size;

IN = 1 - to set up the routine for integration;
IN = 2 - to integrate one step-size;

MODE is the option word (= 0, 1 or 2) for using one of the three modes of integration. When MODE equals

0 - the Adams-Moulton variable step-size is used;
1 - the Runge-Kutta fixed step-size is used;
2 - the Adams-Moulton fixed step-size is used;

N is the number of first order differential equations;
X is the independent or source variable;
H is the step-size or increment in the source variable;
Y is the array of dependent or target variables updated by PINT1;
YP is the array of first derivatives of the target variables Y(N) computed in the subroutine DAUX;
A is an array of 6 cells containing the parameters $(\bar{E}, \bar{M}, \alpha, h_{max}, h_{min}, \beta)$ needed for the variable mode only;
A(1) ($\equiv \bar{E}$) is an upper bound on the truncation error (the number of significant digits which the user desired to preserve locally) for the variable Adams-Moulton method, normally $10^{-8} \leq A(1) \leq 10^{-3}$;
A(2) ($\equiv \bar{M}$) is a positive number from which the lower bound on the truncation error is computed. In particular, when A(2) is zero the routine used the normal value of 100 and in all other cases the lower bound is computed as the quotient of A(1) by A(2);
A(3) ($\equiv \alpha$) is a positive number used to prevent unnecessary reduction in the variable step-size when the dependent variables are sufficiently small. When A(3) is zero the routine uses the normal value of one;
A(4) ($\equiv h_{max}$) is a positive upper bound for the magnitude of the variable step-size. If A(4) is zero the routine assumes there is no upper bound;

A(5) ($\equiv h_{min}$) is positive lower bound for the magnitude of the variable step-size. The routine assumes there is no lower bound when A(5) is zero;

A(6) ($\equiv \beta$) is a positive number between zero and one used to increase or decrease the variable step-size. When A(6) is zero the routine assumes the value of one-half.

IN, N and MODE are integers while X, H, Y, YP and A are all single precision floating point numbers.

The arguments X, H, Y, YP, of the PINT1 calling sequence must be in a COMMON type statement.

Before executing the first PINT1 call, the user must initialize X, H and each of the Y(N) variables. The first call must use control word (IN = 1) to set up the routine for integration. The control word (IN = 2) may be used any number of times after the first to integrated one step-size, provided X, H and Y have not been redefined between integration steps.

e. Common Variables Calculated:

None

f. Size:

$1B2C)_{16} = 6956)_{10}$ bytes

g. Computational Procedure:

In this routine the user is allowed an option of using either the Runge-Kutta classical fourth-order method as modified by E. K. Blum or the Adams-Moulton predictor-corrector method using the Runge-Kutta method for starting the process.

Let the system of equations to be solved be given in the form

$$y_i = f_i(x, y_1, y_2, \cdots, y_N) \qquad (1.1)$$

$$y_i(x_0) = y_{i0} \qquad i = 1, 2, \ldots, N$$

Let y_{in} be the value of y_i at $x = x_n$ and f_{in} the derivative of y_{in} at $x = x_n$. If h is the increment (step-size) of the independent variable x, the classical Runge-Kutta fourth-order method uses the formulas

$$
\begin{aligned}
K_{i1} &= hf_i(x_n,\ y_{in}) \\
K_{i2} &= hf_i(x_n + 1/2\,h,\ y_{in} + 1/2\,K_{i1}) \\
K_{i3} &= hf_i(x_n + 1/2\,h,\ y_{in} + 1/2\,K_{i2}) \\
K_{i4} &= hf_i(x_n + h,\ y_{in} + K_{i3}) \\
y_{i,n+1} &= y_{in} + 1/6(K_{i1} + 2K_{i2} + 2K_{i3} + K_{i4})
\end{aligned}
\qquad (1.2)
$$

where $i = 1, 2, \ldots N$

The E. K. Blum Modification:

The following recursive form of the E. K. Blum's exact modification of the Runge-Kutta is used in this routine:

$$
\left\{
\begin{aligned}
z_0 &= y_n \\
q_0 &= y_n \\
P_0 &= hf(z_0)
\end{aligned}
\right.
\qquad \text{at } x = x_0 \qquad (2.1)
$$

$$
\left\{
\begin{aligned}
z_1 &= z_0 + P_0/2 \\
q_1 &= P_0 \\
P_1 &= hf(z_1)
\end{aligned}
\right.
\qquad \text{at } x = x_0 + h/2 \qquad (2.2)
$$

$$
\left\{
\begin{aligned}
z_2 &= z_1 + P_1/2 - q_1/2 \\
q_2 &= q_1/6 \\
P_2 &= hf(z_2) - P_1/2
\end{aligned}
\right.
\qquad \text{at } x = x_0 + h/2 \qquad (2.3)
$$

$$
\left\{
\begin{aligned}
z_3 &= z_2 + P_2 \\
q_3 &= q_2 + P_2 \\
P_3 &= hf(z_3) + 2P_2
\end{aligned}
\right.
\qquad \text{at } x = x_0 + h \qquad (2.4)
$$

$$
y_{i,n+1} \equiv z_4 = z_3 + q_3 + P_3/6 \qquad (2.5)
$$

(we omit the subscript from each of the vectors z_j, q_j and P_j for reasons of economy)

The main advantage of the modified Runge-Kutta formulas is that they reduce considerably the rounding error arising from the unavoidable use of digital numbers and pseudo-operations.

Adams-Moulton Predictor-Corrector Method:

The routine uses the following formulas for the system (1.1):

$$y_{i,n+1}^{[P]} = y_{i\,n} + h/24(55 f_{i\,n} - 59 f_{i,n-1} + 37 f_{i,n-2} - 9 f_{i,n-3}) \tag{3.1}$$

$$y_{i,n+1} = y_{i,n+1}^{[C]} = y_{i,n} + h/24(9 f_{i,n+1}^{[P]} + 19 f_{i\,n} - 5 f_{i,n-1} + f_{i,n-2}) \tag{3.2}$$

The starting values needed in the predictor formula (3.1) are obtained using the Runge-Kutta-Blum (RKB) method. In the evaluation of y_i at $x = x_{n+1}$ the predictor and corrector formulas are applied only once so that only two derivative evaluations ($f_{i,n+1}^{[P]}$ and $f_{i\,n}$) are needed for each Adams-Moulton (variable or fixed step-size) integration step.

The Variable Adams-Moulton:

The step-size h to be used in the variable mode is determined mainly by:

$$E_{n+1} = \max_i \frac{\left| y_{i,n+1}^{[P]} - y_{i,n+1}^{[C]} \right|}{14 D_i} \tag{3.3}$$

$$D_i = \max_i \left\{ \left| y_{i,n+1}^{[C]} \right|, \alpha \right\} \quad , \quad i = 1, 2, \ldots N$$

where

E_{n+1} is the local truncation error estimate in the actual evaluation of y_{n+1} ; α (> 0) is a constant used to prevent unnecessary reductions in $|h|$ whenever $|y_{i,n+1}|$ is small (normally the routine will set α = 1, unless otherwise specified by the user).

Let

$\bar{E}$ be the upper bound on the truncation error estimate, specified by the user, that is the number of significant digits which the user desires to preserve locally throughout the integration. Normally $\bar{E}$ should be in the range $10^{-8} \leq \bar{E} \leq 10^{-3}$ and in double precision $\bar{E}$ should be in the range $10^{-16} \leq \bar{E} \leq 15^{12}$;

M (> 0) be a constant, specified by the user, from which a lower bound $\underline{E} = \bar{M}^{-1}\bar{E}$ is obtained (normally M ranges from 50 to 150 and in double precision from 1000 to 1500);

β be a constant between 0 to 1 used to increase or decrease the step-size. The routine will take $\beta = 1/2$ unless β is otherwise specified by the user.

The step-size h will be then increased or decreased according to the following inequalities:

If

(4.1) $E_{n+1} > \bar{E}$ the step-size is reduced to βh, where $0 < \beta < 1$;

(4.2) $M^{-1}\bar{E} < E_{n+1} < \bar{E}$ the step-size remains unchanged;

(4.3) $E_{n+1} < M^{-1}\bar{E}$ for 3 successive integration steps the step-size is increased to h/β.

Increasing and Decreasing the Step-Size:

The starting values, the first three successive points after the initial point p_0, for the Adams-Moulton formulas are always obtained using the RKB method whenever the interval size is changed, just as at the beginning of an integration.

In the variable mode if the starting values, the first three successive points, have been obtained using the RKB method then the next point is computed using the Adams-Moulton predictor-corrector formulas (3.1) and (3.2). Whenever the truncation error at this point calls for a decrease in h the routine returns to the initial point p_0 and computes new starting values with the decreased value of h. However, if the step-size is to be decreased at a point p_i, where the preceeding point p_{i-1} was computed in the variable mode and the inequality (4.2) held at p_{i-1}, then a new start is initiated at p_{i-1} with decreased value of $|h|$.

If for three successive variable integration steps p_{i-1}, p_i and p_{i+1} inequality (4.3) holds, then a new start is initiated at p_{i+1} with the increased value of $|h|$. After an interval is increased, the routine prevents increasing again until 6 more points have been complete. However, the routine may decrease the interval as often as necessary. The truncation error test based on (3.3) will guarantee that the local error does not exceed $\bar{E}$, however the cumulative error will usually exceed $\bar{E}$. Hence $\bar{E}$ should be chosen sufficiently small to allow for an accumulation of truncation error.

The user must always provide a starting value for h and he may, if desired, specify a maximum value of $|h|$, h_{max} beyond which the routine will not increase h and a minimum value of $|h|$, h_{min}, below which it will not decrease h. If no value is specified for h_{max} and h_{min} the routine will set the values at 10^3 and 10^{-17}, respectively.

Negative values of h may be used for backward integration.

Control and DAUX:

There are two entries to this routine. The first (control word = 1) must be used once at the beginning to set up the routine for integration of a given set of N differential equations. The second entry (control word = 2) may be used any number of times after the first to integrate all y_i from x to $x+h$.

Whenever the control word is 1 the routine uses the auxiliary subroutine DAUX to evaluate the derivatives at the initial point $x = x_0$ and returns with all y_i unchanged. The routine also checks and sets up the six parameter words $\bar{E}$, M, α, h_{max}, h_{min} and β needed in the variable mode of operation. Before executing the initialization entry, the user must have already set up the appropriate values for x, h and y_i, $i = 1, 2, \ldots N$. Ordinarily, after an execution of the second entry all y_i assume new values, x will have been advanced to the value $x+h$ and h will be unchanged, unless in the variable mode. On exit the values y_i are always these which correspond to the point $x+h$ and y_i.

Whenever an integration step involves RKB integration, four derivative evaluations are needed, mainly

$$\begin{aligned} &f_i(x_n + 1/2h,\ y_{in} + 1/2\, K_{1i}) \\ &f_i(x_n + 1/2h,\ y_{in} + 1/2\, K_{i2}) \\ &f_i(x_n + h,\ y_{in} + K_{i3}) \\ &y_{i,n+1} = f_i(x_n + h,\ y_{n+1}) \end{aligned} \tag{5.1}$$

where the K_{ij} are given by (1.2) and modified by (2.1) - (2.5). In the fixed h predictor-corrector mode, the first three integration entries involve RKB integration and subsequent ones involve AM integration. Each AM integration step requires two derivative evaluations.

$$\begin{aligned} f_{i,n+1}^{[P]} &= f_i(x_n + h,\ y_{i,n+1}^{[P]}) \\ y'_{i,n+1} &= f_i(x_n + h,\ y_{i,n+1}) \end{aligned} \tag{5.2}$$

A particular integration set up, in the variable mode, may involve either AM or RKB or both.

References:

(1) SHARE Write-Up No. 0602 (D2RWINT)
(2) SHARE Write-Up No. 0450 (D2RDE2F)
(3) Blum, K. E., A Modification of the Runge-Kutta Fourth Order Method, Mathematics of Computation, April 1962, pp. 176-187

30. SUBROUTINE PLOTTP(IPLT)

a. Purpose:

1. Write output to FORTRAN device 1 for post-processing graphic displays

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, COMPN, TIRIN

c. Subroutines Required:

None

d. Arguments:

IPLT controls the type of record written; static, dynamic or end of data, for values of IPLT of 1, 2 and 3, respectively

e. Common Variables Calculated:

None

f. Size:

$324)_{16} = 804)_{10}$ bytes

g. Computational Procedure:

1. If IPLT = 1 a static header record is written to device 1 consisting of the following variables: HED, DADE, A, B, TS, ZR, RHO, ZF, RW, TF, TR
2. If IPLT = 2 a dynamic record is written consisting of: T, XCP, YCP, ZCP, PHIT, THETT, PSIT, DEL1, DEL2, DEL3, PHIR, PSI1, PHI1, PHI2, (XGPP(I), YGPP(I), ZGPP(I), I = 1,4), (ICONTW(I), I = 1,4). Note: ICONTW is an indicator. If 1, wheel I is rolling; if -1, wheel I is skidding; if 0, wheel I is off the ground.
3. If IPLT = 3, an end of data record consisting of 30 works of -9999.0 is written.

31. SUBROUTINE POLY

a. Purpose:

1. To find root of a fifth degree polynomial

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

C0 - zeroth order polynomial coefficient
C - array containing polynomial coefficients
X - initial approximation
Y - polynomial root
C1 - polynomial value

e. Common Variables Calculated:

None

f. Size:

$234)_{16} = 564)_{10}$ bytes

g. Computational Procedure:

The root of the polynomial is found by the Newton-Raphson Method

32. SUBROUTINE RESFRC

a. Purpose:

1. Calculate frictional force between the vehicle and barrier
2. Calculate the force and moment components acting on the vehicle

b. Common Blocks Required:

INTG, DIMV, COMP, INPT2, BARIER, BARSTR, HARDPT

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

SXR, SYR, SZR, UPT, URP, VPT, VRP, WPT, WRP, SFXS, SFYS, SFZS, SNPS, SNTS, VTAN, FRICF, FRICT, SNPSS

f. Size:

$4F4)_{16} = 1268)_{10}$ bytes

g. Computational Procedure:

1. Compute the location of the point of application of the vehicle crush force (SXR, SYR, SZR) and store in the first element of the X, Y and Z arrays
2. Store the locations of the point of application of the vehicle hard point forces in the second, third and fourth elements of those arrays
3. Compute the velocity components of those four points in the space axes (UPT(I), VPT(I), WPT(I), I = 1,4)
4. Compute the friction force components for each point, the total force vector components in the vehicle axis system (SFXS, SFYS, SFZS) and the moments acting on the vehicle sprung mass (SNPS, SNTS, SNPSS)

SUBROUTINE RUFFRC(I,ZGM)

a. Purpose:

1. To determine an equivalent radial tire force and ground contact point from the distributed tire spring model when the road roughness option is being used

b. Common Blocks Required:

INPT1, DIMV, COMP, COMPN, TIRIN, RUFNES

c. Subroutines Required:

INTRPL

d. Arguments:

I = wheel number for which calculations are made
ZGM = single dimensional array containing the road roughness data

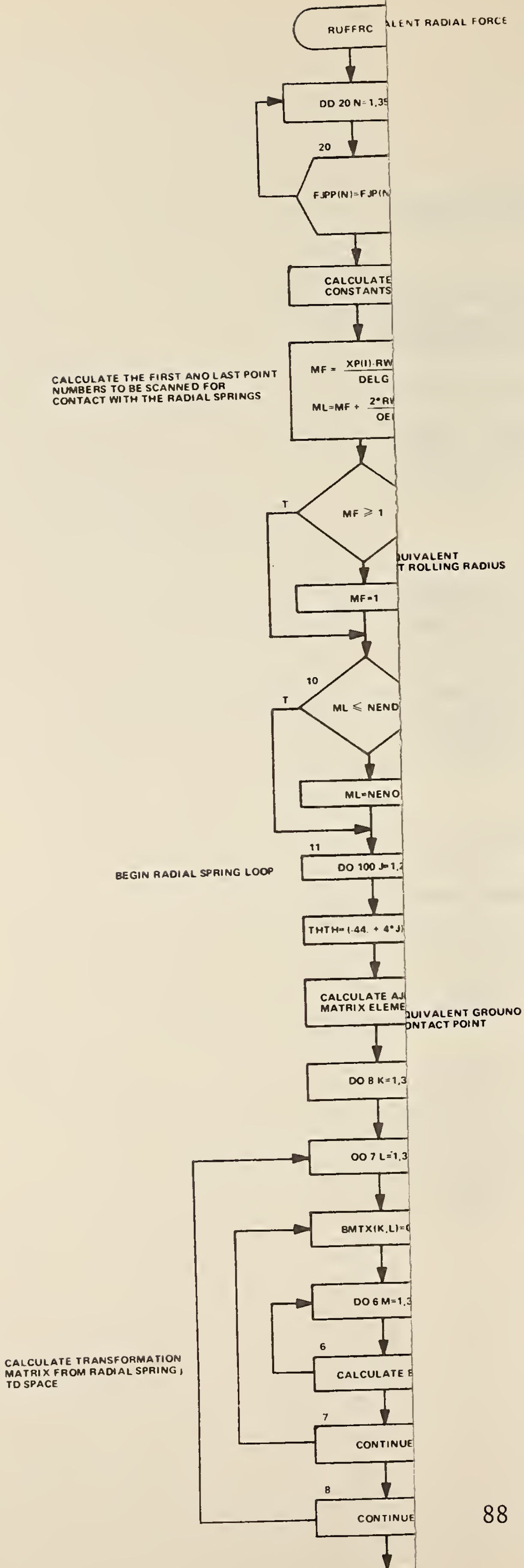
e. Common Variables Calculated:

FR, HI, CAR, CBR, CGR, CPG, CTG, SPG, STG, BMTX, PHGI, SFRX, SFRY, SFRZ, XGPP, YGPP, ZGPP, AJMTX

f. Size:

$DC4)_{16} = 3524)_{10}$ bytes

g. Computational Procedure:


RUFFRC
ALENT RADIAL FORCE
DD 20 N=1,35
20
FJPP(N)=FJP(N
CALCULATE
CONSTANTS
CALCULATE THE FIRST ANO LAST POINT NUMBERS TO BE SCANNED FOR CONTACT WITH THE RADIAL SPRINGS
MF = XP(I)-RW / DELG
ML=MF + 2*RW / OEI
T
MF ≥ 1
UIVALENT
T ROLLING RADIUS
MF=1
10
T
ML ≤ NEND
ML=NENO
11
BEGIN RADIAL SPRING LOOP
DO 100 J=1,2
THTH=(-44. + 4*J)
CALCULATE AJ
MATRIX ELEME
UIVALENT GROUNO
NTACT POINT
DO 8 K=1,3
OO 7 L=1,3
BMTX(K,L)=(
DO 6 M=1,3
6
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING TD SPACE
CALCULATE B
7
CONTINUE
8
CONTINUE

34. SUBROUTINE RUFRED(NEND,DELG,DGMAX,ZRTAB)

a. Purpose:

1. Read road roughness data from FORTRAN device 4

b. Common Blocks Required:

None

c. Subroutine Required:

None

d. Arguments:

NEND = the number of road roughness points to be read from FORTRAN unit 4
DELG = the distance increment between points
DGMAX = (NEND-1) * DELG
ZRTAB = a single dimension array into which the road roughness data is read

e. Common Variables Calculated:

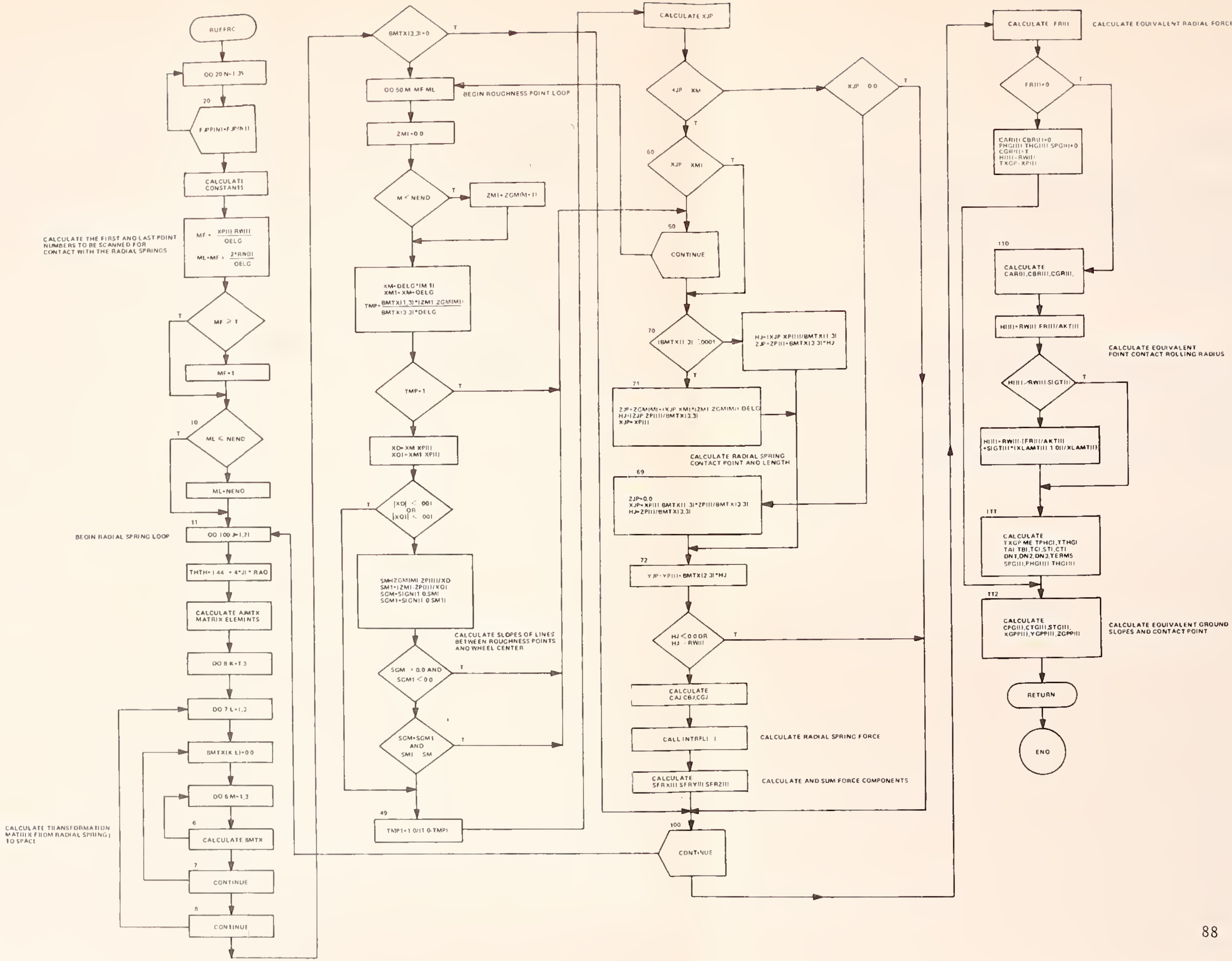
None

f. Size:

$2B8)_{16} = 696)_{10}$ bytes

g. Computational Procedure:

The road roughness data is read via an unformatted READ statement into the ZRTAB array. The maximum number of points allowed is 2200.


RUFFRC
CALCULATE CONSTANTS
CALCULATE THE FIRST AND LAST POINT NUMBERS TO BE SCANNED FOR CONTACT WITH THE RADIAL SPRINGS
BEGIN RADIAL SPRING LOOP
CALCULATE AMTX MATRIX ELEMENTS
CALCULATE BMTX
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING I TO SPACE
CONTINUE
BEGIN ROUGHNESS POINT LOOP
CALCULATE SLOPES OF LINES BETWEEN ROUGHNESS POINTS AND WHEEL CENTER
CALCULATE XJP
CONTINUE
CALCULATE RADIAL SPRING CONTACT POINT AND LENGTH
CALCULATE CAJ,CBJ,CGJ
CALCULATE RADIAL SPRING FORCE
CALCULATE AND SUM FORCE COMPONENTS
CALCULATE FR(I)
CALCULATE EQUIVALENT RADIAL FORCE
CALCULATE EQUIVALENT POINT CONTACT ROLLING RADIUS
CALCULATE EQUIVALENT GROUND SLOPES AND CONTACT POINT
RETURN
END

35. SUBROUTINE SIMSOL

a. Purpose:

1. This subroutine solves a set of real simultaneous linear algebraic equations AX = B, with input, output and internal computation all in single precision.

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

A - is a 2-dimensional (ND1 x ND2) matrix of coefficients
N - is the number of equations and unknowns
ND1 - is the first dimension of A in the calling program (ND1. GE. N and ND2. GE. N+1)

e. Common Variables Calculated:

None

f. Size:

$5E8)_{16} = 1512)_{10}$ bytes

g. Computational Procedure:

The routine will find the solution X of AX = B where A is a N by N matrix and B(I) is stored in A(I, N+1).
The solution X(I) is returned in A(I, N+1).

Note: The Matrix A is destroyed by the subroutine.

Example: REAL A(20,25)
CALL SIMSOL (A, 10, 20)

The solution is obtained by elimination using the largest pivotal divisor of each column. Each stage of elimination consists of interchanging rows when necessary to avoid division by zero or small numbers.

The forward solution to obtain variable N is done in N stages. The back solution for the other variables is calculated by successive substitutions. The final solution values are developed in column N+1 of matrix A, with variable 1 and A (1, N+1), variable 2 in A(2, N+1), ..., and variable N in A (N, N+1).

36. SUBROUTINE SFORCE

a. Purpose:

1. Calculate sprung mass impact force due to vehicle interference with barrier

b. Common Blocks Required:

INTG, DIMV, COMP, COMPN, INPT2, BARIER, BARSTR, HARDPT

c. Subroutines Required:

SIMSOL, AREA, NLDFL, RESFRC

d. Arguments:

None

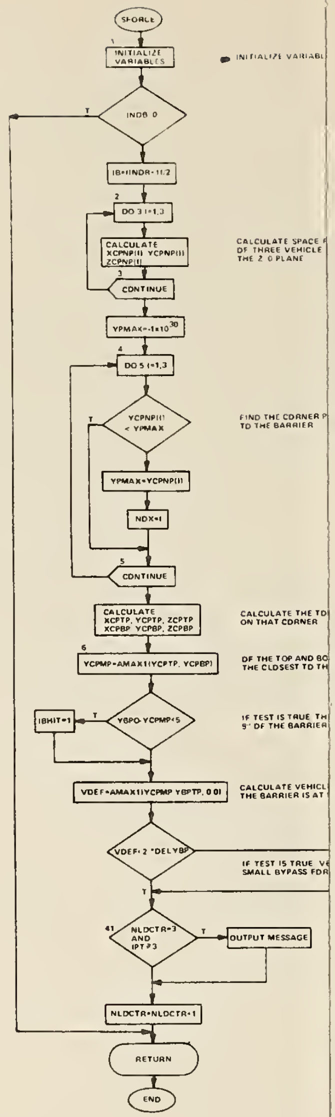
e. Common Variables Calculated:

FB, FN, RB, AA2, BB2, CAB, CBB, CC2, CGB, IPT, RB1, RR2, SXR, SYR, SZR, UNP, VNP, WNP, XBB, XBT, XNN, XRI, YBT, YNN, YRI, ZBB, ZBT, ZNN, ZRI, CABT, CAB1, CBBT, CBB1, CGBT, CGB1, IPLN, IDPT, NSEG, RR2P, SDEN, SFXS, SFYS, SFZS, SNPS, SNTS, VDEF, VMAX, VTAN, XMTX, XSTI, YBPT, YSTI, ZSTI, AINTI, DELBB, FNSTI, FRICT, ININD, IBHIT, NUNLD, SNPSS, XCPBP, XCPNP, XCPTP, XSTIP, YCPBP, YCPMP, YCPNP, YCPTP, YSTIP, ZCPBP, ZCPNP, ZCPTP, ZSTIP, INDXPT, NLDCTR

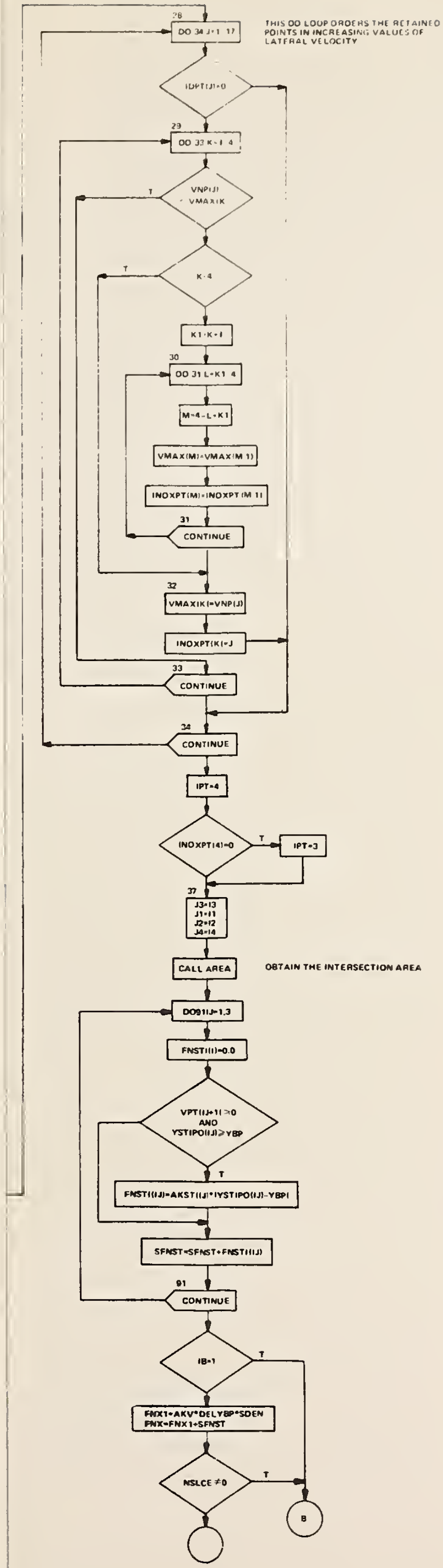
f. Size:

$1E90)_{16} = 7824)_{10}$ bytes

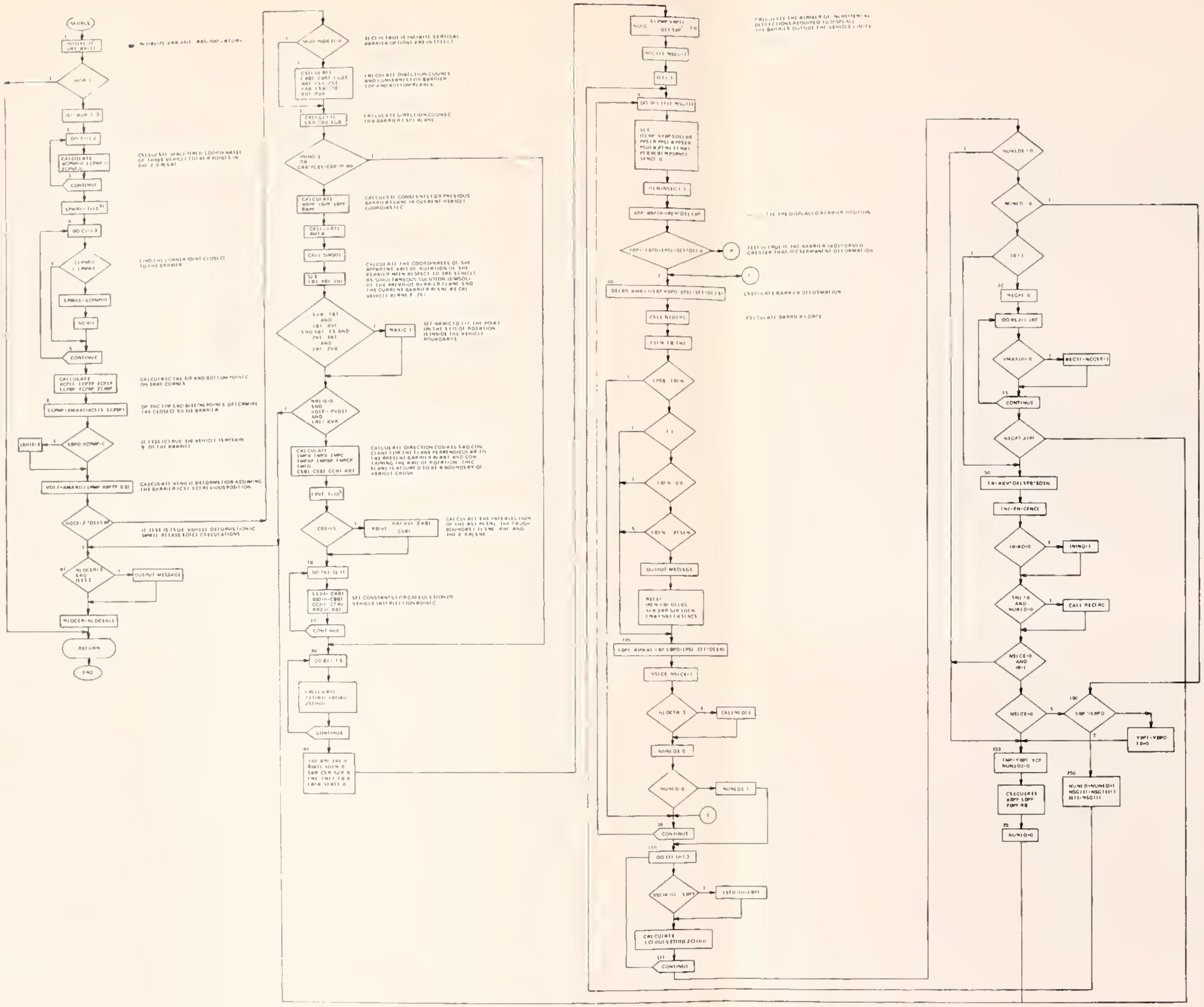
g. Computational Procedure:

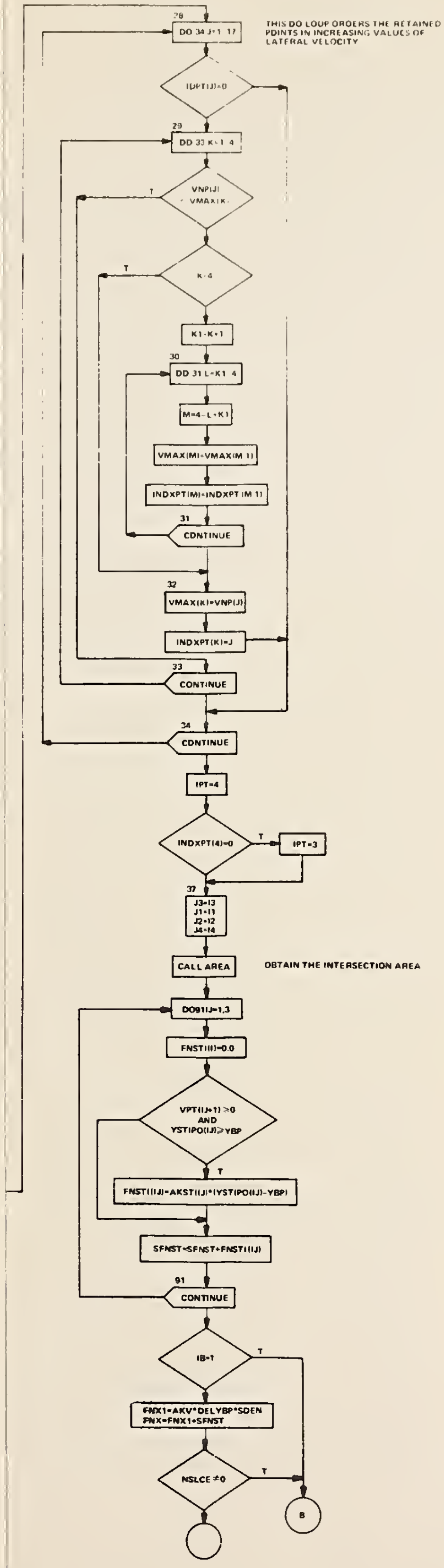


SFORCE
INITIALIZE VARIABLES
INITIALIZE VARIABL
INDB 0
T
IB=(INDB+1)/2
2
DO 3 I=1,3
CALCULATE XCPNP(I) YCPNP(I) ZCPNP(I)
CALCULATE SPACE P
OF THREE VEHICLE
THE Z 0 PLANE
3
CONTINUE
YPMAX=-1x10^30
4
DO 5 I=1,3
YCPNP(I) < YPMAX
T
FIND THE CORNER P
TO THE BARRIER
YPMAX=YCPNP(I)
NDX=I
5
CONTINUE
CALCULATE XCPTP, YCPTP, ZCPTP XCPBP YCPBP, ZCPBP
CALCULATE THE TO
ON THAT CORNER
6
YCPMP=AMAX1(YCPTP, YCPBP)
OF THE TOP AND BO
THE CLOSEST TO TH
IBHIT=1
T
YBPO-YCPMP<5
IF TEST IS TRUE TH
5" OF THE BARRIER
VDEF=AMAX1(YCPMP YBPTP, 0.01)
CALCULATE VEHICL
THE BARRIER IS AT
VDEF>2 *DELYBP
IF TEST IS TRUE VE
SMALL BYPASS FOR
T
41
NLDCTR=3 AND IPT≠3
T
OUTPUT MESSAGE
NLDCTR=NLDCTR+1
RETURN
END

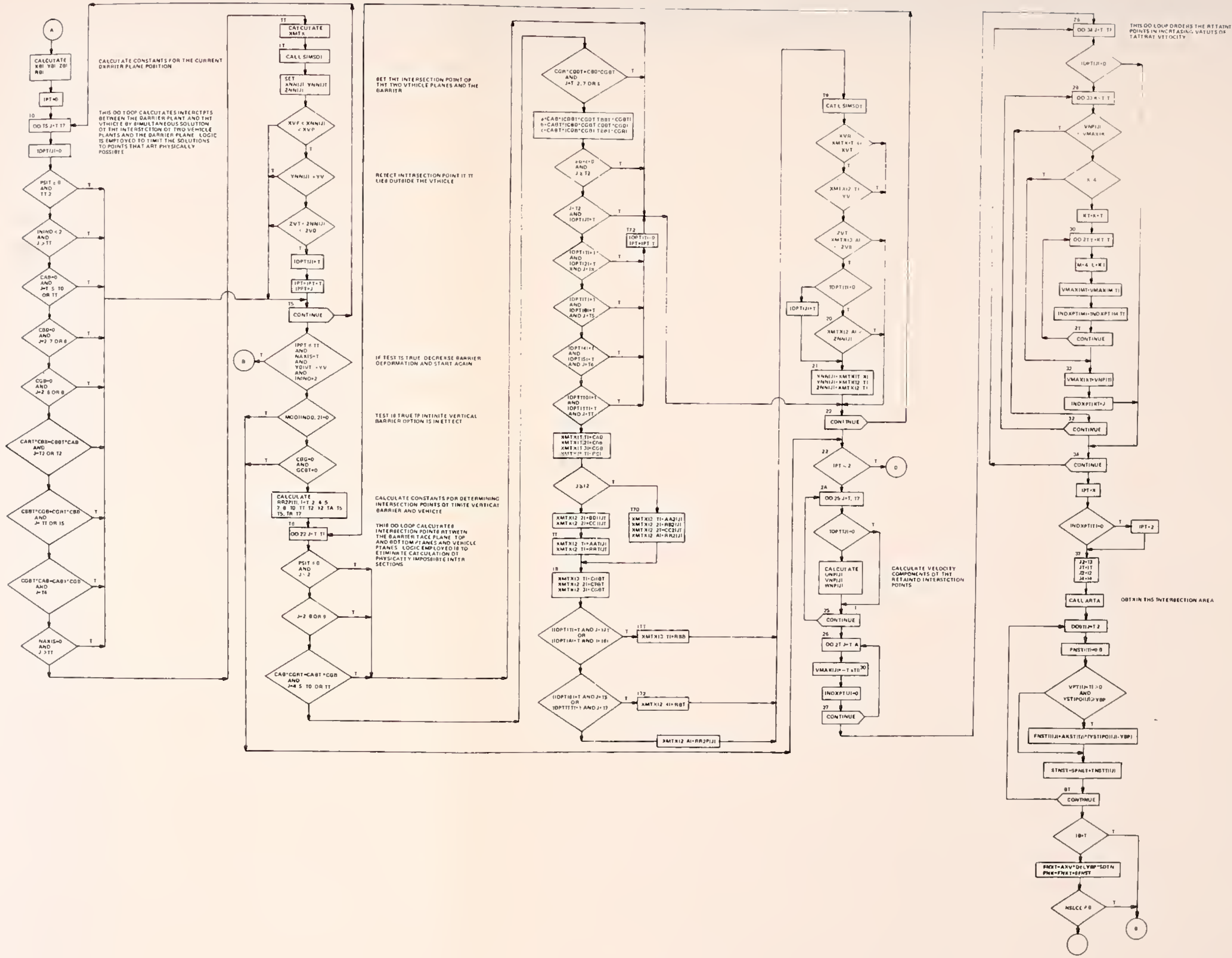


28
DO 34 J=1 17
THIS DO LOOP ORDERS THE RETAINED POINTS IN INCREASING VALUES OF LATERAL VELOCITY
IDPT(J)=0
29
DO 33 K=1 4
T
VNP(J) < VMAX(K)
T
K=4
K1=K+1
30
DO 31 L=K1 4
M=4-L+K1
VMAX(M)=VMAX(M-1)
INOXPT(M)=INOXPT(M-1)
31
CONTINUE
32
VMAX(K)=VNP(J)
INOXPT(K)=J
33
CONTINUE
34
CONTINUE
IPT=4
INOXPT(4)=0
T
IPT=3
37
J3=I3
J1=I1
J2=I2
J4=I4
CALL AREA
OBTAIN THE INTERSECTION AREA
DO91IJ=1,3
FNST(IJ)=0.0
VPT(IJ+1)≥0
AND
YSTIPO(IJ)≥YBP
T
FNST(IJ)=AKST(IJ)*(YSTIPO(IJ)-YBP)
SFNST=SFNST+FNST(IJ)
91
CONTINUE
IB=1
T
FNX1=AKV*DELYBP*SDEN
FNX=FNX1+SFNST
NSLCE≠0
T
B





28
DO 34 J=1 17
THIS DO LOOP ORDERS THE RETAINED POINTS IN INCREASING VALUES OF LATERAL VELOCITY
IDPT(J)=0
29
DO 33 K=1 4
T
VNP(J) < VMAX(K)
T
K=4
K1=K+1
30
DO 31 L=K1 4
M=4-L+K1
VMAX(M)=VMAX(M 1)
INDXPT(M)=INDXPT(M 1)
31
CONTINUE
32
VMAX(K)=VNP(J)
INDXPT(K)=J
33
CONTINUE
34
CONTINUE
IPT=4
INDXPT(4)=0
T
IPT=3
37
J3=I3
J1=I1
J2=I2
J4=I4
CALL AREA
OBTAIN THE INTERSECTION AREA
DO91IJ=1,3
FNST(I)=0.0
VPT(IJ+1) ≥0
AND
YSTIPO(IJ)≥YBP
T
FNST(IJ)=AKST(IJ)*(YSTIPO(IJ)-YBP)
SFNST=SFNST+FNST(IJ)
91
CONTINUE
IB=1
T
FNX1=AKV*DELYBP*SDEN
FNX=FNX1+SFNST
NSLCE ≠0
T
B


CALCULATE CONSTANTS FOR THE CURRENT BARRIER PLANE POSITION
THIS DO LOOP CALCULATES INTERCEPTS BETWEEN THE BARRIER PLANE AND THE VEHICLE BY SIMULTANEOUS SOLUTION OF THE INTERSECTION OF TWO VEHICLE PLANES AND THE BARRIER PLANE. LOGIC IS EMPLOYED TO LIMIT THE SOLUTIONS TO POINTS THAT ARE PHYSICALLY POSSIBLE
GET THE INTERSECTION POINT OF THE TWO VEHICLE PLANES AND THE BARRIER
REJECT INTERSECTION POINT IF IT LIES OUTSIDE THE VEHICLE
IF TEST IS TRUE DECREASE BARRIER DEFORMATION AND START AGAIN
TEST IS TRUE IF INFINITE VERTICAL BARRIER OPTION IS IN EFFECT
CALCULATE CONSTANTS FOR DETERMINING INTERSECTION POINTS OF FINITE VERTICAL BARRIER AND VEHICLE
THIS DO LOOP CALCULATES INTERSECTION POINTS BETWEEN THE BARRIER FACE PLANE, TOP AND BOTTOM PLANES AND VEHICLE PLANES. LOGIC EMPLOYED IS TO ELIMINATE CALCULATION OF PHYSICALLY IMPOSSIBLE INTERSECTIONS
CALCULATE VELOCITY COMPONENTS OF THE RETAINED INTERSECTION POINTS
THIS DO LOOP ORDERS THE RETAINED POINTS IN INCREASING VALUES OF LATERAL VELOCITY
OBTAIN THE INTERSECTION AREA
CALL SIMSOL
CALL AREA
CONTINUE

JBROUTINE SUSFRC(DISP,VEL)

. Purpose:

1. This subroutine calculates the suspension forces acting between the sprung and unsprung masses at the four vehicle corners

. Common Blocks Required:

INPT, INPT3, INTG, DIMV, COMP, ADTNL, APTABL, INSUS, SUSCMP

. Subroutines Required:

INTRPL

. Arguments:

DISP - a four element array containing the suspension displacements
VEL - a four element array containing the suspension velocities

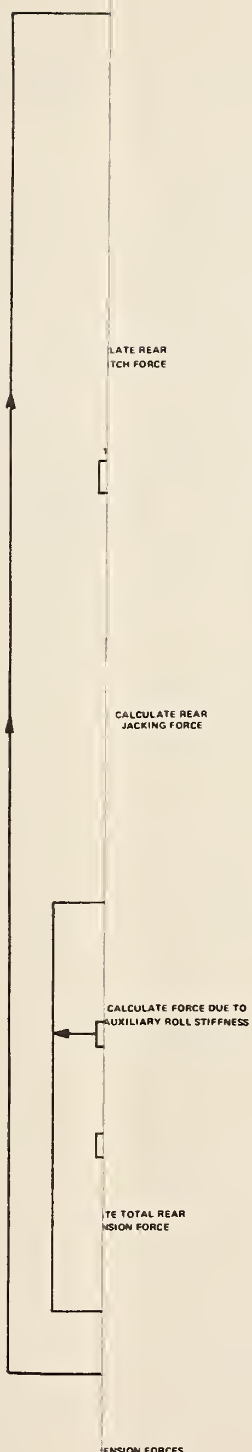
. Common Variables Calculated:

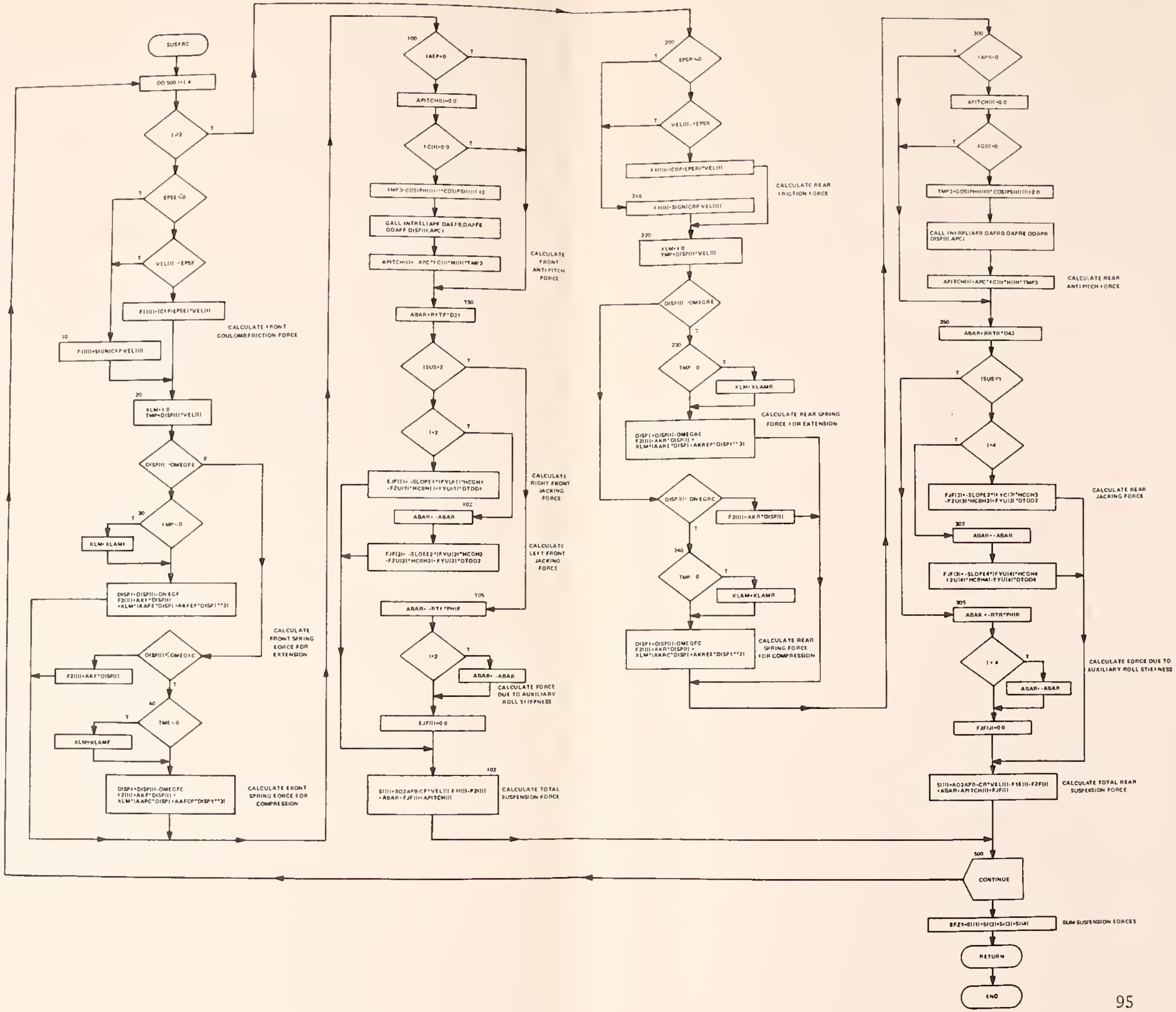
SI, FJF, F1I, F2I, SFZ1, APITCH

. Size:

$7BC)_{16} = 1980)_{10}$ bytes

. Computational Procedure:


LATE REAR
TCH FORCE
CALCULATE REAR
JACKING FORCE
CALCULATE FORCE DUE TO
AUXILIARY ROLL STIFFNESS
TE TOTAL REAR
NSION FORCE
ENSION FORCES


SUSFRC
CALCULATE FRONT COULOMB FRICTION FORCE
CALCULATE FRONT SPRING FORCE FOR EXTENSION
CALCULATE FRONT SPRING FORCE FOR COMPRESSION
CALCULATE FRONT ANTI-PITCH FORCE
CALCULATE RIGHT FRONT JACKING FORCE
CALCULATE LEFT FRONT JACKING FORCE
CALCULATE FORCE DUE TO AUXILIARY ROLL STIFFNESS
CALCULATE TOTAL SUSPENSION FORCE
CALCULATE REAR FRICTION FORCE
CALCULATE REAR SPRING FORCE FOR EXTENSION
CALCULATE REAR SPRING FORCE FOR COMPRESSION
CALCULATE REAR ANTIPITCH FORCE
CALCULATE REAR JACKING FORCE
CALCULATE FORCE DUE TO AUXILIARY ROLL STIFFNESS
CALCULATE TOTAL REAR SUSPENSION FORCE
CONTINUE
SUM SUSPENSION FORCES
RETURN
END

38. SUBROUTINE TEREAD

a. Purpose:

1. This subroutine reads terrain table input cards

b. Common Blocks Required:

INPT

c. Subroutines Required:

None

d. Arguments:

I - Terrain table number
NNBX - Number of X' boundaries
NNBY - Number of Y' boundaries
NNX - Number of X' terrain entries
NNY - Number of Y' terrain entries
NZ5T - Indicator for variable increment table
NERR - Error indicator

e. Common Variables Calculated:

ZGP, XBDRY, YBDRY, PSBDRO, XXZGP5, YYZGP5

f. Size:

$626)_{16} = 1574)_{10}$ bytes

J
38

39. SUBROUTINE TIRFRC(J)

a. Purpose:

1. Calculate tire circumferer

b. Common Blocks Required:

DIMV, COMP, COMPN, ADTNL, TIRI

c. Subroutines Required:

None

d. Arguments:

The argument, J, indicates the
calculations are made

e. Common Variables Calculated:

FC, FS, FXU, FYU, FZU, BETP, F
FRXU, FRYU, FRZU, FSXU, FSYU,
SFZU, BETBR, CPHIC, PHICI, SPH

f. Size:

$922)_{16} = 2338)_{10}$ bytes

g. Computational Procedure:

SIDE FORCE CALCULATIONS FOR TIRE J OR FOR A PAIR OF TIRES IF ONE TIRE OF THE PAIR IS SPINNING.

IF BOTH LATERAL ANO LONGITUOINAL VELOCITY COMPONENTS OF THE TIRE ARE SMALL, DO ND COMPUTE A SIOE FORCE.

CALCULATE MAXIMUM AVAILABLE SIOE FORCE

TEST IS TRUE IF THE TIRE IS HEAVILY LOAOEO

CALCULATE NON-DIMENSIONAL SLIP ANGLE VARIABLE OUE TO CAMBER (BETP) ANO NON-OIMENSIONAL SLIP ANGLE (BETBR)

TEST IS FALSE IF SIOE FORCE CAPABILITIES ARE SATURATED.

CALCULATE SIOE FORCE

SINCE FRCP REOUIRE FS, REPEAT ONCE FROM START 2 USING UPDATED FS

39. SUBROUTINE TIRFRC(J)

a. Purpose:

1. Calculate tire circumferential and side forces

b. Common Blocks Required:

DIMV, COMP, COMPN, ADTNL, TIRIN

c. Subroutines Required:

None

d. Arguments:

The argument, J, indicates the wheel number for which calculations are made

e. Common Variables Calculated:

FC, FS, FXU, FYU, FZU, BETP, FCXU, FCYU, FCZU, FRCP, FRXU, FRYU, FRZU, FSXU, FSYU, FSZU, SFXU, SFYU, SFZU, BETBR, CPHIC, PHICI, SPHIC

f. Size:

$922)_{16} = 2338)_{10}$ bytes

g. Computational Procedure:

SIOE FORCE CALCULATIONS FOR TIRE J OR FOR A PAIR OF TIRES IF ONE TIRE OF THE PAIR IS SPINNING

IF BOTH LATERAL ANO LONGITUOINAL VELOCITY COMPONENTS OF THE TIRE ARE SMALL, DO NO COMPUTE A SIDE FORCE.

CALCULATE MAXIMUM AVAILABLE SIDE FORCE

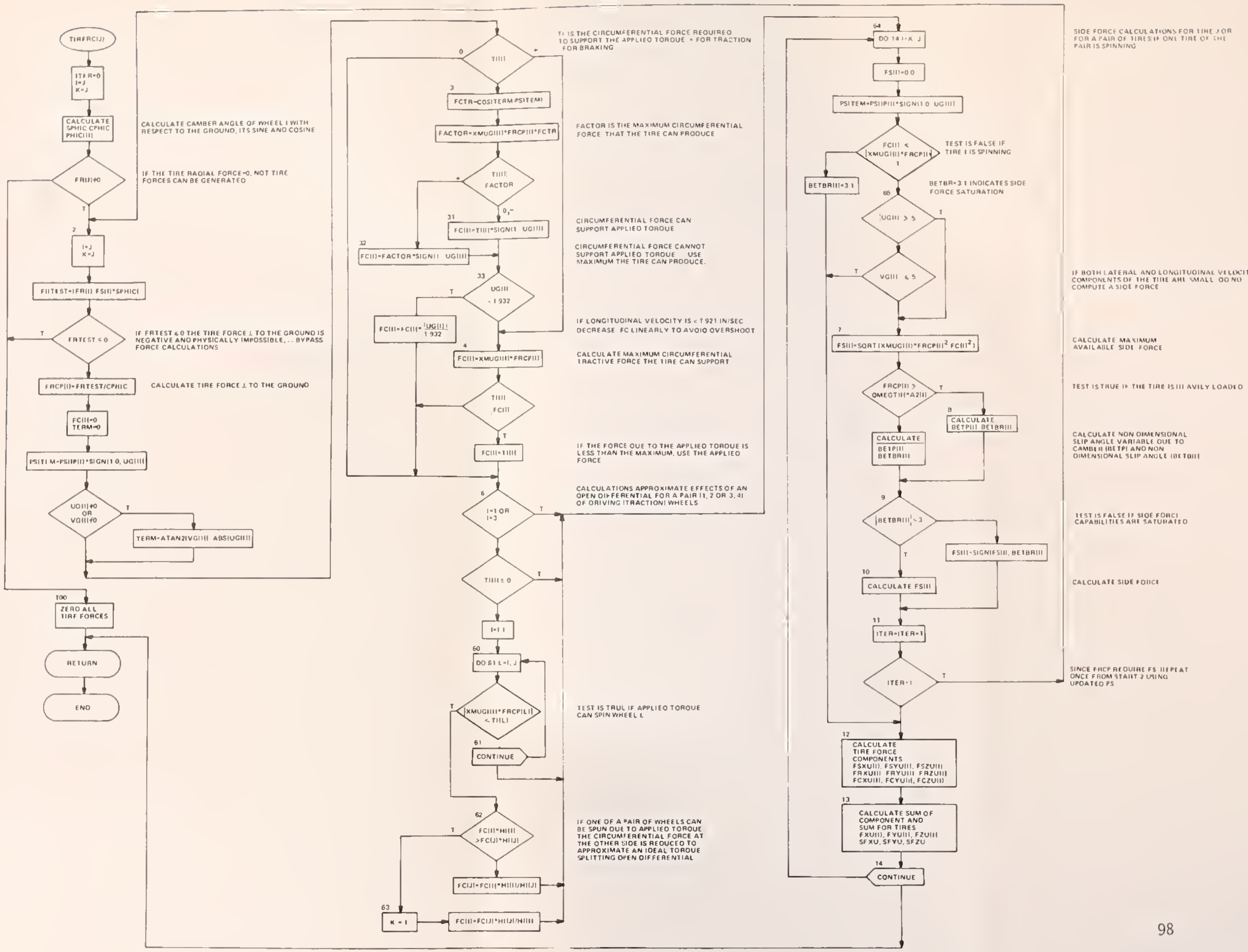
TEST IS TRUE IF THE TIRE IS HEAVILY LOAOEO

CALCULATE NON-OIMENSIONAL SLIP ANGLE VARIABLE DUE TO CAMBER (BETP) ANO NON-DIMENSIONAL SLIP ANGLE (BETBR)

TEST IS FALSE IF SIDE FORCE CAPABILITIES ARE SATURATED.

CALCULATE SIOE FORCE

SINCE FRCP REDUIRE FS, REPEAT ONCE FROM START 2 USING UPOATED FS


TIREFRC(J)
CALCULATE CAMBER ANGLE OF WHEEL I WITH RESPECT TO THE GROUND, ITS SINE AND COSINE
IF THE TIRE RADIAL FORCE=0, NOT TIRE FORCES CAN BE GENERATED
IF FRTEST ≤ 0 THE TIRE FORCE ⊥ TO THE GROUND IS NEGATIVE AND PHYSICALLY IMPOSSIBLE, .. BYPASS FORCE CALCULATIONS
FRCP(I)=FRTEST/CPHIC
CALCULATE TIRE FORCE ⊥ TO THE GROUND
ZERO ALL TIRE FORCES
RETURN
END
FACTOR IS THE MAXIMUM CIRCUMFERENTIAL FORCE THAT THE TIRE CAN PRODUCE
CIRCUMFERENTIAL FORCE CAN SUPPORT APPLIED TORQUE
CIRCUMFERENTIAL FORCE CANNOT SUPPORT APPLIED TORQUE USE MAXIMUM THE TIRE CAN PRODUCE.
CALCULATE MAXIMUM CIRCUMFERENTIAL TRACTIVE FORCE THE TIRE CAN SUPPORT
IF THE FORCE DUE TO THE APPLIED TORQUE IS LESS THAN THE MAXIMUM, USE THE APPLIED FORCE
CALCULATIONS APPROXIMATE EFFECTS OF AN OPEN DIFFERENTIAL FOR A PAIR (1, 2 OR 3, 4) OF DRIVING (TRACTION) WHEELS
CONTINUE
IF ONE OF A PAIR OF WHEELS CAN BE SPUN DUE TO APPLIED TORQUE THE CIRCUMFERENTIAL FORCE AT THE OTHER SIDE IS REDUCED TO APPROXIMATE AN IDEAL TORQUE SPLITTING OPEN DIFFERENTIAL
BETBR=3.1 INDICATES SIDE FORCE SATURATION
CALCULATE MAXIMUM AVAILABLE SIDE FORCE
CALCULATE SIDE FORCE
CALCULATE TIRE FORCE COMPONENTS
CALCULATE SUM OF COMPONENT AND SUM FOR TIRES
CONTINUE

40. SUBROUTINE TMCNST

a. Purpose:

1. Evaluate time dependent va
   other subroutines
2. Test for and index coordin

b. Common Blocks Required:

INPT, INTG, COMP, EINDEX, ADTN

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

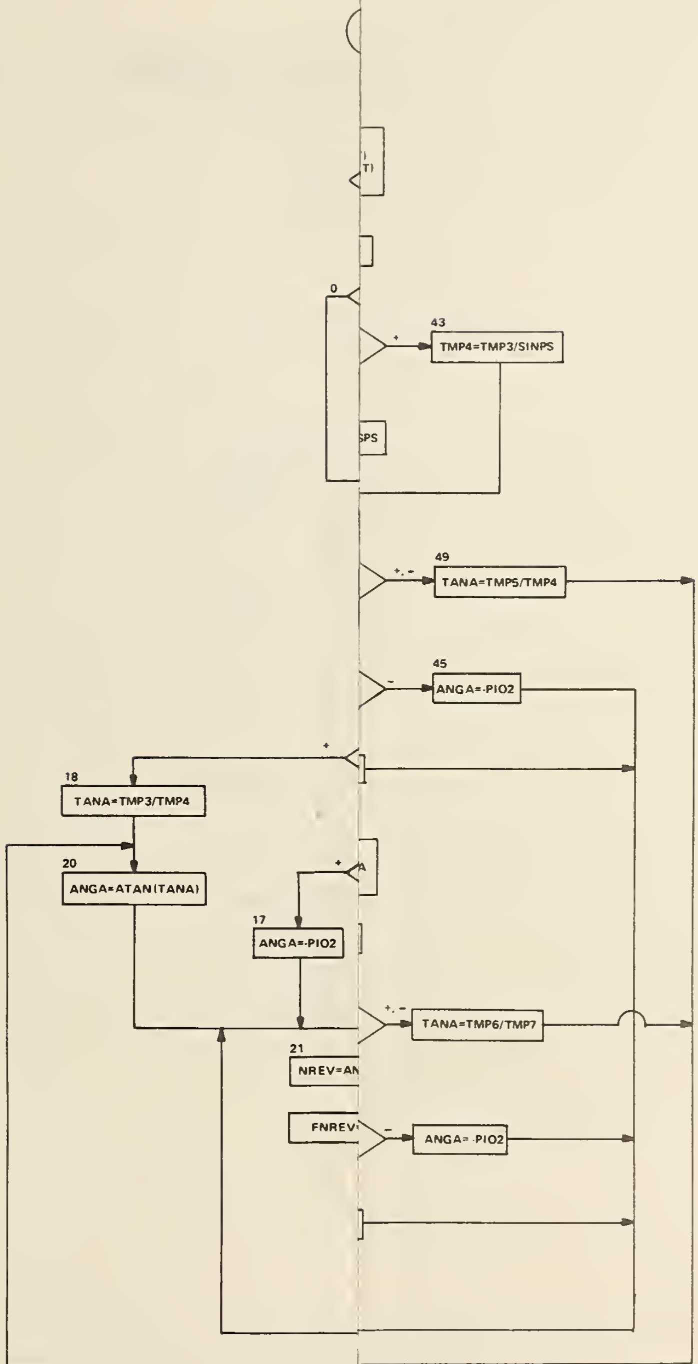
PQ, PR, P2, QR, R2, UQ, UR, VP
D21, D43, RPR, TPF, TPR, GCTH,
PHRP, PSIT, RFPF, TG61, TIZ2,
ZRD3, ZRD4, CNMTX, COSPH, COSP
CPSTP, DD1M2, DD1P2, DD3M4, DD
D3MD4, D3PD4, ISTOP, PHIF2, PH
RPHRD, SECTP, SINPH, SINPS, SI
TANTP, THETT, XINDL, ZFD12, ZF
COSPHN, COSPSN, COSTHN, CTHETP
SINPHN, SINPSN, SINTHN, STHETP

f. Size:

$D36)_{16} = 3382)_{10}$ bytes

g. Computational Procedure:

1. Compute time dependent var
2. Test for coordinate system
   shown below


43
TMP4=TMP3/SINPS
49
TANA=TMP5/TMP4
45
ANGA=-PIO2
18
TANA=TMP3/TMP4
20
ANGA=ATAN(TANA)
17
ANGA=-PIO2
TANA=TMP6/TMP7
21
NREV=AN
FNREV
ANGA=-PIO2

40. SUBROUTINE TMCNST

a. Purpose:

1. Evaluate time dependent variables that are required in other subroutines
2. Test for and index coordinate system if necessary

b. Common Blocks Required:

INPT, INTG, COMP, EINDEX, ADTNL, TIRIN

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

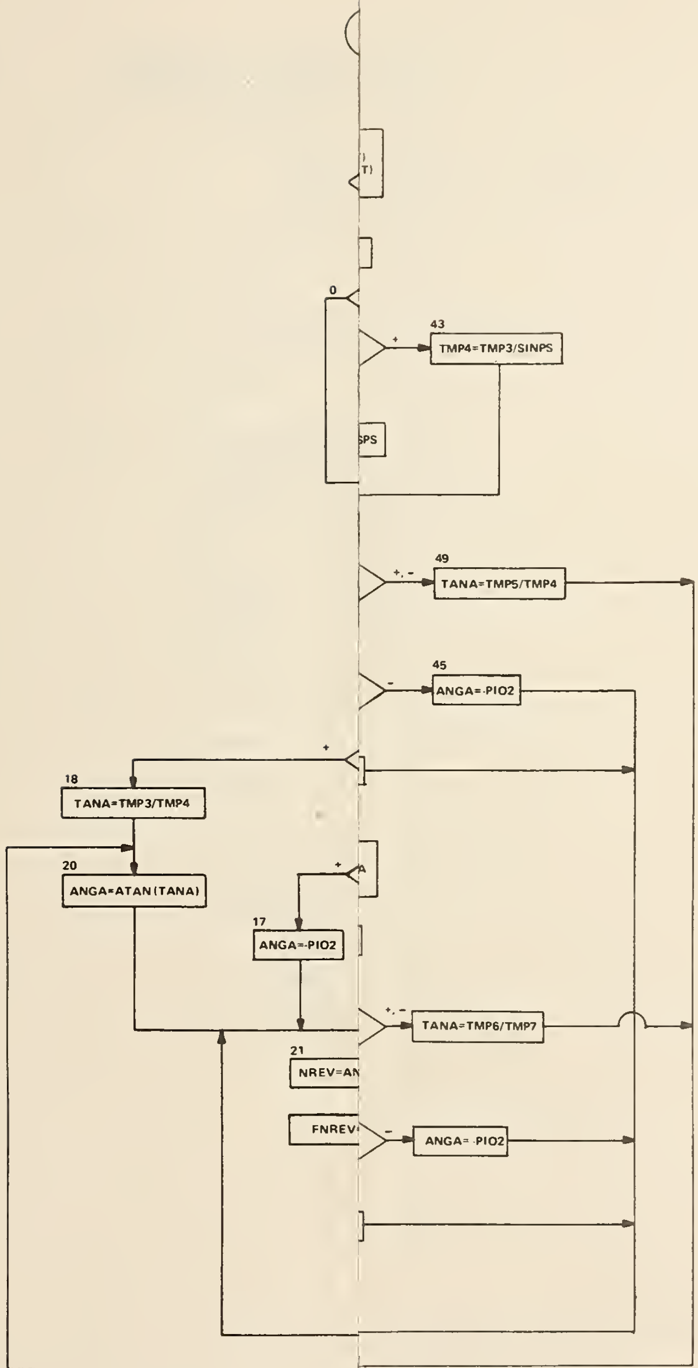
PQ, PR, P2, QR, R2, UQ, UR, VP, VR, WP, WQ, D21, D43, RPR, TPF, TPR, GCTH, GSTH, PHFP, PHIT, PHRP, PSIT, RFPF, TG61, TIZ2, WFMF, ZFDI, ZFD2, ZRD3, ZRD4, CNMTX, COSPH, COSPS, COSTH, CPHTP, CPSTP, DD1M2, DD1P2, DD3M4, DD3P4, D1MD2, D1PD2, D3MD4, D3PD4, ISTOP, PHIF2, PHIR2, RPF2M, RPHFD, RPHRD, SECTP, SINPH, SINPS, SINTH, SPHTP, SPSTP, TANTP, THETT, XINDL, ZFD12, ZFD3R, ZRD3R, ZRD34, COSPHN, COSPSN, COSTHN, CTHETP, PHIFD2, PHIRD2, SINPHN, SINPSN, SINTHN, STHETP, ZFD1RF

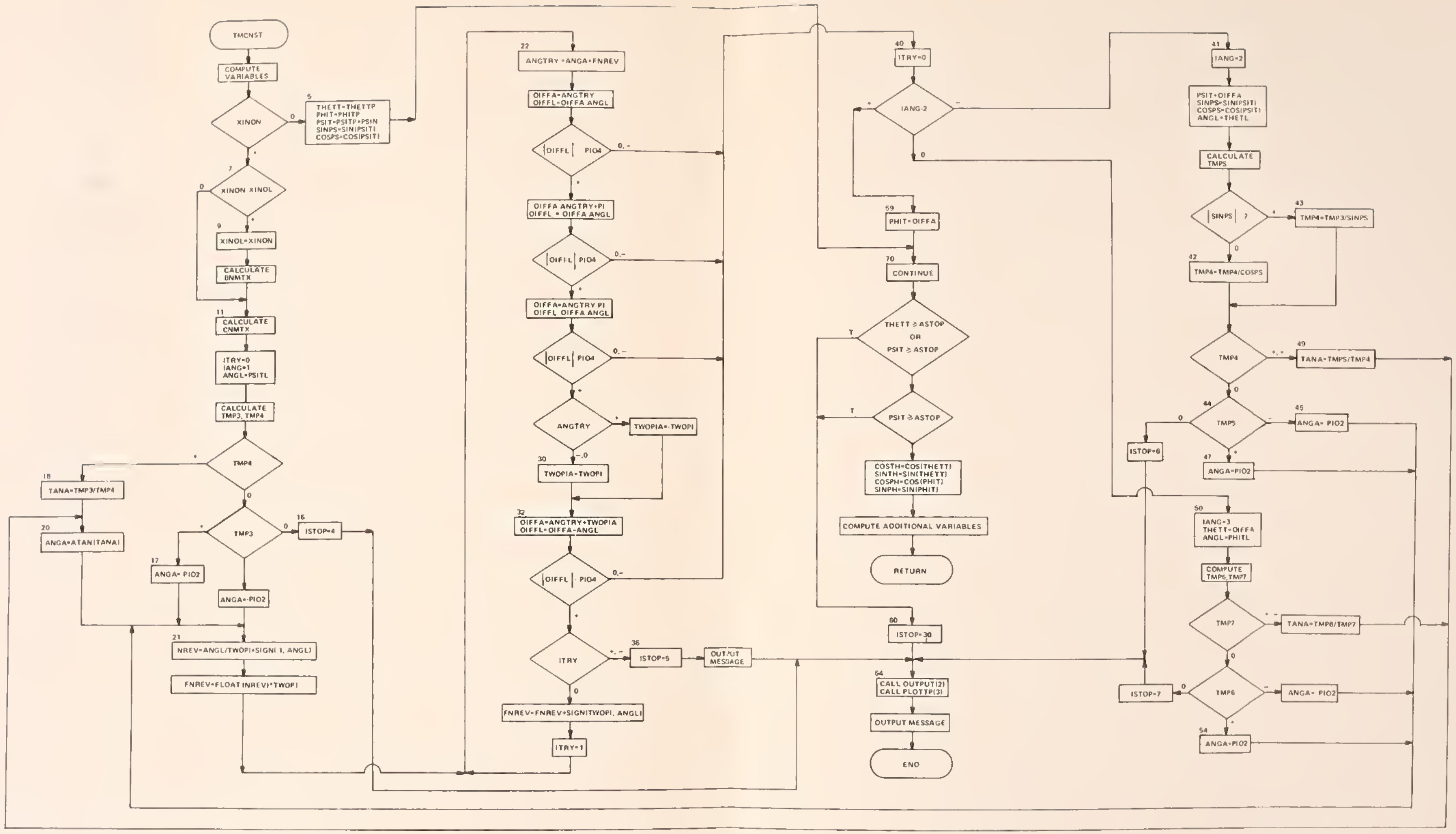
f. Size:

$D36)_{16} = 3382)_{10}$ bytes

g. Computational Procedure:

1. Compute time dependent variables
2. Test for coordinate system indexing; if required as shown below


43
TMP4=TMP3/SINPS
49
TANA=TMP5/TMP4
45
ANGA=-PIO2
18
TANA=TMP3/TMP4
20
ANGA=ATAN(TANA)
17
ANGA=-PIO2
TANA=TMP6/TMP7
21
ANGA= -PIO2


TMCNST
COMPUTE VARIABLES
XINON
THETT=THETTP
PHIT=PHITP
PSIT=PSITP+PSIN
SINPS=SIN(PSIT)
COSPS=COS(PSIT)
XINON XINOL
XINOL=XINON
CALCULATE BNMTX
CALCULATE CNMTX
ITRY=0
IANG=1
ANGL=PSITL
CALCULATE TMP3, TMP4
TMP4
TANA=TMP3/TMP4
ANGA=ATAN(TANA)
TMP3
ISTOP=4
ANGA= PIO2
ANGA=-PIO2
NREV=ANGL/TWOPI+SIGN(1, ANGL)
FNREV=FLOAT(NREV)*TWOPI
ANGTRY=ANGA+FNREV
OIFFA=ANGTRY
OIFFL=OIFFA ANGL
OIFFA ANGTRY+PI
OIFFL = OIFFA ANGL
OIFFA=ANGTRY PI
OIFFL OIFFA ANGL
ANGTRY
TWOPIA=-TWOPI
TWOPIA=TWOPI
OIFFA=ANGTRY+TWOPIA
OIFFL=OIFFA-ANGL
ITRY
ISTOP=5
OUTPUT MESSAGE
FNREV=FNREV+SIGN(TWOPI, ANGL)
ITRY=1
ITRY=0
IANG-2
PHIT=OIFFA
CONTINUE
THETT ≥ ASTOP
OR
PSIT ≥ ASTOP
PSIT ≥ ASTOP
COSTH=COS(THETT)
SINTH=SIN(THETT)
COSPH=COS(PHIT)
SINPH=SIN(PHIT)
COMPUTE ADDITIONAL VARIABLES
RETURN
ISTOP=30
CALL OUTPUT(2)
CALL PLOTTP(3)
OUTPUT MESSAGE
END
IANG=2
PSIT=OIFFA
SINPS=SIN(PSIT)
COSPS=COS(PSIT)
ANGL=THETL
CALCULATE TMP5
SINPS
TMP4=TMP3/SINPS
TMP4=TMP4/COSPS
TMP4
TANA=TMP5/TMP4
TMP5
ANGA= PIO2
ISTOP=6
ANGA=PIO2
IANG=3
THETT=OIFFA
ANGL=PHITL
COMPUTE TMP6,TMP7
TMP7
TANA=TMP6/TMP7
TMP6
ISTOP=7
ANGA= PIO2
ANGA=PIO2

41. SUBROUTINE TREAD

a. Purpose:

1. This subroutine reads a one-dimensional card input table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

NCARD - Input card number
NCRDS - Number of cards to be read
NT - Number of elements to be read into the table
NDIM - Maximum table dimension
ARRAY - Table array
NERR - Error indicator

e. Common Variables Calculated:

None

f. Size:

$258)_{16} = 600)_{10}$ bytes

g. Computational Procedure:

1. Read table input cards checking to insure that the table sequence number increases with each card.
2. Load the variables into the table array.

42. SUBROUTINE T2READ

a. Purpose:

1. This subroutine reads a two-dimensional input table.

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

NCARD - Input card number
ND1 - Row dimension of the input table
NI - Number of rows to be read
NJ - Number of columns to be read
ARRAY - Table array
NERR - Error indicator

e. Common Variables Calculated:

None

f. Size:

$2C4)_{16} = 708)_{10}$ bytes

g. Computational Procedure:

The input table is read rowwise with the second subscript varying most rapidly.

43. SUBROUTINE UMOMNT(IS)

a. Purpose:

1. This subroutine calculates the moments acting on the sprung and unsprung masses

b. Common Blocks Required:

INPT, DIMV, COMP, INSUS, SUSCMP

c. Subroutines Required:

None

d. Arguments:

IS - Suspension option indicator

e. Common Variable Calculated:

SNPF, SNPR, SNPU, SNTU, SNPSU, TERM1, TERM2, TERM3

f. Size:

$79C)_{16} = 1948)_{10}$ bytes

g. Computational Procedure:

1. For IS = 0 (independent front, solid axle rear suspension) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU) and the rear axle roll moment (SNPR).
2. For IS = 1 (independent front and rear suspension) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU).
3. For IS = 2 (solid front and rear axles) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU) and the front and rear axle roll moments (SNPF, SNPR).

44. SUBROUTINE VGORNT

a. Purpose:

1. Determine the orientation of the vehicle wheels with respect to the ground
2. Calculate the circumferential tire forces due to applied wheel torques

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, TIRIN, INSUS, SUSCMP, NEWCRB

c. Subroutines Required:

INTRP5, GCP, CRBIMP, TIRFRC, RUFFRC

d. Arguments:

None

e. Variables Calculated:

AS, AX, AY, BS, BX, BY, CS, CX, CY, D1, D2, D3, P1, P3, P4, P5, P6, P7, TI, UG, VG, V1, V2, V3, V4, W1, W2, W3, W4, CAC, CAH, CAS, CBC, CBH, CBS, CGC, CGH, CGS, CPG, CTG, SPG, STG, CAGZ, CAYW, CAZW, CBGZ, CBYW, CBZW, CGGZ, CGYW, CGZW, CPYG, CTXG, HCAH, HCBH, HCGH, LCB1, LCB2, PHGI, STXG, THGI, TMP3, TMP4, ZPGI, DISTD, DISTS, DISTX, DISTY, PSIIP

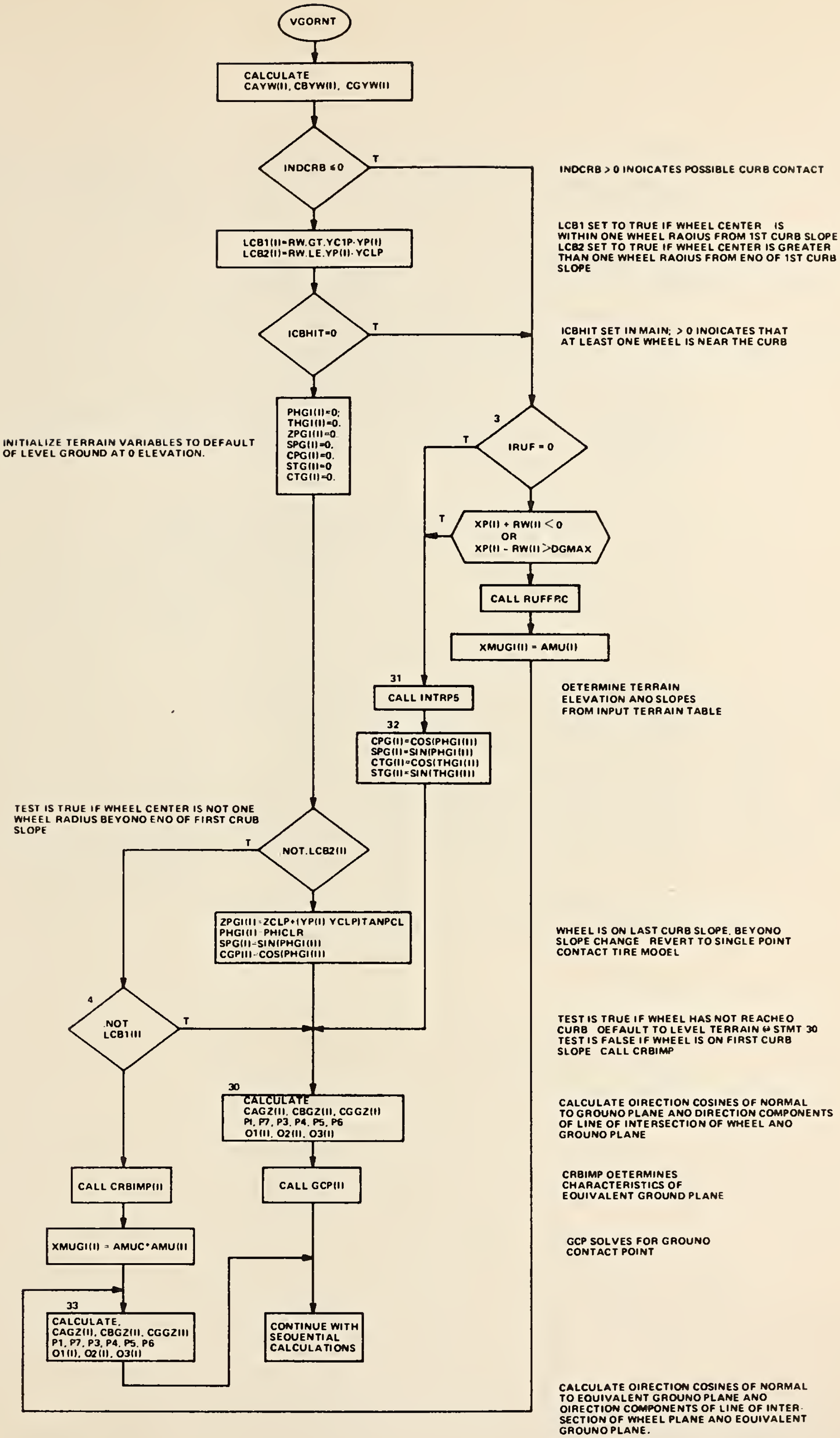
f. Size:

$1028)_{16} = 4136)_{10}$ bytes

g. Computational Procedure:

For wheels I = 1 to 4

1. Calculate the direction cosines of the normal to the wheel plane.
2. Determine the direction cosines of a normal to the ground plane and direction components of the intersection of the wheel plane and ground plane as follows:


VGORNT
CALCULATE CAYW(I), CBYW(I), CGYW(I)
INDCRB ≤ 0
T
INDCRB > 0 INOICATES POSSIBLE CURB CONTACT
LCB1(I)=RW.GT.YC1P-YP(I)
LCB2(I)=RW.LE.YP(I)-YCLP
LCB1 SET TO TRUE IF WHEEL CENTER IS WITHIN ONE WHEEL RAOIUS FROM 1ST CURB SLOPE LCB2 SET TO TRUE IF WHEEL CENTER IS GREATER THAN ONE WHEEL RAOIUS FROM ENO OF 1ST CURB SLOPE
ICBHIT=0
T
ICBHIT SET IN MAIN; > 0 INOICATES THAT AT LEAST ONE WHEEL IS NEAR THE CURB
PHGI(I)=0; THGI(I)=0. ZPGI(I)=0 SPG(I)=0. CPG(I)=0. STG(I)=0 CTG(I)=0.
INITIALIZE TERRAIN VARIABLES TO DEFAULT OF LEVEL GROUND AT 0 ELEVATION.
3
IRUF = 0
T
XP(I) + RW(I) < 0 OR XP(I) - RW(I) > DGMAX
T
CALL RUFFRC
XMUGI(I) = AMU(I)
31
CALL INTRP5
OETERMINE TERRAIN ELEVATION ANO SLOPES FROM INPUT TERRAIN TABLE
32
CPG(I)=COS(PHGI(I))
SPG(I)=SIN(PHGI(I))
CTG(I)=COS(THGI(I))
STG(I)=SIN(THGI(I))
TEST IS TRUE IF WHEEL CENTER IS NOT ONE WHEEL RADIUS BEYONO ENO OF FIRST CRUB SLOPE
NOT.LCB2(I)
T
ZPGI(I)=ZCLP+(YP(I)-YCLP)TANPCL
PHGI(I)=PHICLR
SPG(I)=SIN(PHGI(I))
CGP(I)=COS(PHGI(I))
WHEEL IS ON LAST CURB SLOPE, BEYONO SLOPE CHANGE REVERT TO SINGLE POINT CONTACT TIRE MOOEL
4
.NOT LCB1(I)
T
TEST IS TRUE IF WHEEL HAS NOT REACHEO CURB OEFAULT TO LEVEL TERRAIN @ STMT 30 TEST IS FALSE IF WHEEL IS ON FIRST CURB SLOPE CALL CRBIMP
30
CALCULATE CAGZ(I), CBGZ(I), CGGZ(I) P1, P7, P3, P4, P5, P6 O1(I), O2(I), O3(I)
CALCULATE OIRECTION COSINES OF NORMAL TO GROUNO PLANE ANO DIRECTION COMPONENTS OF LINE OF INTERSECTION OF WHEEL ANO GROUNO PLANE
CALL CRBIMP(I)
CALL GCP(I)
CRBIMP OETERMINES CHARACTERISTICS OF EOUIVALENT GROUND PLANE
XMUGI(I) = AMUC*AMU(I)
GCP SOLVES FOR GROUNO CONTACT POINT
33
CALCULATE, CAGZ(I), CBGZ(I), CGGZ(I) P1, P7, P3, P4, P5, P6 O1(I), O2(I), O3(I)
CONTINUE WITH SEOUENTIAL CALCULATIONS
CALCULATE OIRECTION COSINES OF NORMAL TO EQUIVALENT GROUNO PLANE ANO OIRECTION COMPONENTS OF LINE OF INTERSECTION OF WHEEL PLANE ANO EOUIVALENT GROUNO PLANE.

3. Calculate the direction cosines of the line of action of the radial tire force with respect to the vehicle axes, CAH(I), CBH(I), CGH(I).
4. Calculate the circumferential tire force due to applied torques, TI(I).
5. Calculate the lateral and vertical velocities of the tire at the ground contact point with components resolved in the vehicle axes, (V1, W1); (V2, W2); (V3, W3); (V4, W4).
6. Calculate the direction components of the vehicle x axis projected into the ground plane, AX(I), BX(I), CX(I).
7. Calculate the sine and cosine of the angle between the vehicle x axis and its projection into the ground plane STXG(I), CTXG(I).
8. Calculate the longitudinal velocity of the tire contact point parallel to the ground plane UG(I).
9. Calculate the direction components of the vehicle y axis projected into the ground plane, AY(I), BY(I), CY(I).
10. Calculate the sine and cosine of the angle between the vehicle y axis and its projection into the ground plane SPYG(I), CPYG(I).
11. Calculate the lateral velocity of the tire contact point parallel to the ground plane, VG(I).
12. Calculate the direction cosines of the steering axis of the wheel.
13. Calculate the steer angle in the ground plane, PSIIP(I).
14. Calculate the direction cosines of the line of action of the circumferential tire force (CAC(I), CBC(I), CGC(I)) and of the tire side force (CAS(I), CBS(I), CGS(I)).
15. Call TIRFRC(I) to obtain magnitudes of side and circumferential tire forces.

45. SUBROUTINE VPOS

a. Purpose:

1. Compute positions, orientations and velocities of the vehicle wheels
2. Calculate torques acting on front and rear wheels
3. Calculate directions of the x and y axis in space

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, INSUS, SUSCMP

c. Subroutines Required:

INTRPL, DRIVER

d. Arguments:

None

e. Common Variables Calculated:

TI, U1, U2, U3, U4, CAX, CAY, CBX, CBY, CGX, CGY, TQF, TQR, X1P, X2P, X3P, X4P, Y1P, Y2P, Y3P, Y4P, Z1P, Z2P, Z3P, Z4P, PHI1, PHI2, PHI3, PHI4, PSI1, PSI2, PSI3, PSI4, SFXU, SFYU, SFZU, DTDD1, DTDD2, DTDD3, DTDD4, DTHF1, DTHF2, DTHR3, DTHR4, PHI1D, PHI2D, PHI3D, PHI4D, PSIFI, SFYUF, SFYUR, PHIFID, SLOPE1, SLOPE2, SLOPE3, SLOPE4

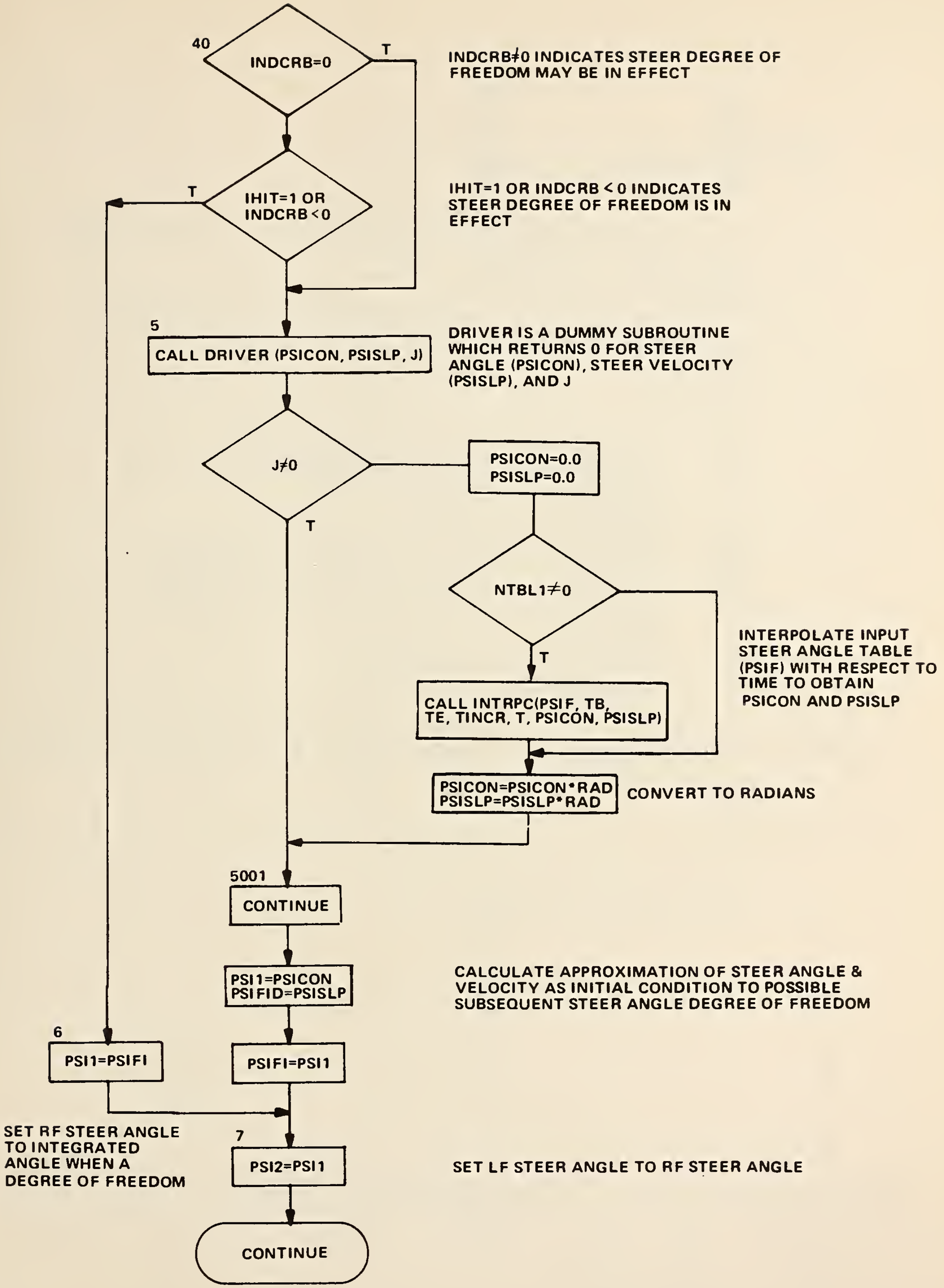
f. Size:

$B50)_{16} = 2896)_{10}$ bytes

g. Computational Procedure:

1. Call INTRPL to interpolate input torque tables TQF, TQR with respect to simulated time, T. TQF yields front wheel torques TI(1), TI(2); TQR yields rear wheel torques TI(3), TI(4).
2. Calculate longitudinal velocities of wheel centers along the vehicle axes, U1, U2, U3, U4. Note that for independent suspension options, INTRPC is called to obtain the track change and rate of track change as a function of suspension position.
3. Zero forces acting on the unsprung masses SFYU = SFXU = SFYUF = SFYUR = SFZU = 0.
4. Calculate AMTX, the transformation matrix from vehicle to space coordinate systems.
5. Calculate direction cosines of the vehicle x and y axis in space (CAX, CBX, CGX and CAY, CBY, CGY).
6. Calculate positions of the wheel centers in space (X1P, Y1P, Z1P); (X2P, Y2P, Z2P); (X3P, Y3P, Z3P); (X4P, Y4P, Z4P).

7. Call INTRPC (entry point in INTRPL) to obtain wheel camber angles and rates of change of camber angles with deflection by interpolation of the input camber tables with respect to suspension deflection for independent suspension options. Note that since the input table of camber is in units of degrees, a conversion to radians is also made.
8. Determine the front wheel steer angle with the following logic.



40
INDCRB=0
T
INDCRB≠0 INDICATES STEER DEGREE OF FREEDOM MAY BE IN EFFECT
T
IHIT=1 OR INDCRB < 0
IHIT=1 OR INDCRB < 0 INDICATES STEER DEGREE OF FREEDOM IS IN EFFECT
5
CALL DRIVER (PSICON, PSISLP, J)
DRIVER IS A DUMMY SUBROUTINE WHICH RETURNS 0 FOR STEER ANGLE (PSICON), STEER VELOCITY (PSISLP), AND J
J≠0
T
PSICON=0.0
PSISLP=0.0
NTBL1≠0
T
INTERPOLATE INPUT STEER ANGLE TABLE (PSIF) WITH RESPECT TO TIME TO OBTAIN PSICON AND PSISLP
CALL INTRPC(PSIF, TB, TE, TINCR, T, PSICON, PSISLP)
PSICON=PSICON*RAD
PSISLP=PSISLP*RAD
CONVERT TO RADIANS
5001
CONTINUE
PSI1=PSICON
PSIFID=PSISLP
CALCULATE APPROXIMATION OF STEER ANGLE & VELOCITY AS INITIAL CONDITION TO POSSIBLE SUBSEQUENT STEER ANGLE DEGREE OF FREEDOM
6
PSI1=PSIFI
PSIFI=PSI1
SET RF STEER ANGLE TO INTEGRATED ANGLE WHEN A DEGREE OF FREEDOM
7
PSI2=PSI1
SET LF STEER ANGLE TO RF STEER ANGLE
CONTINUE

46. SUBROUTINE WHEEL

a. Purpose:

1. To calculate equivalent tire radial mode spring rates

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

AKT - Point contact model tire spring rate
SIGT - Point contact model tire deflection at which spring rate increases
XLAMT - Spring rate increase
RWHJB - Beginning deflection for radial spring table
RWHJE - Ending deflection for radial spring table
DRWHJ - Deflection increment for radial spring table
NFJP - Number of radial spring table entries
RW - Undeflected tire radius
FJP - Radial spring force table
NO - Maximum number of entries in radial spring force table

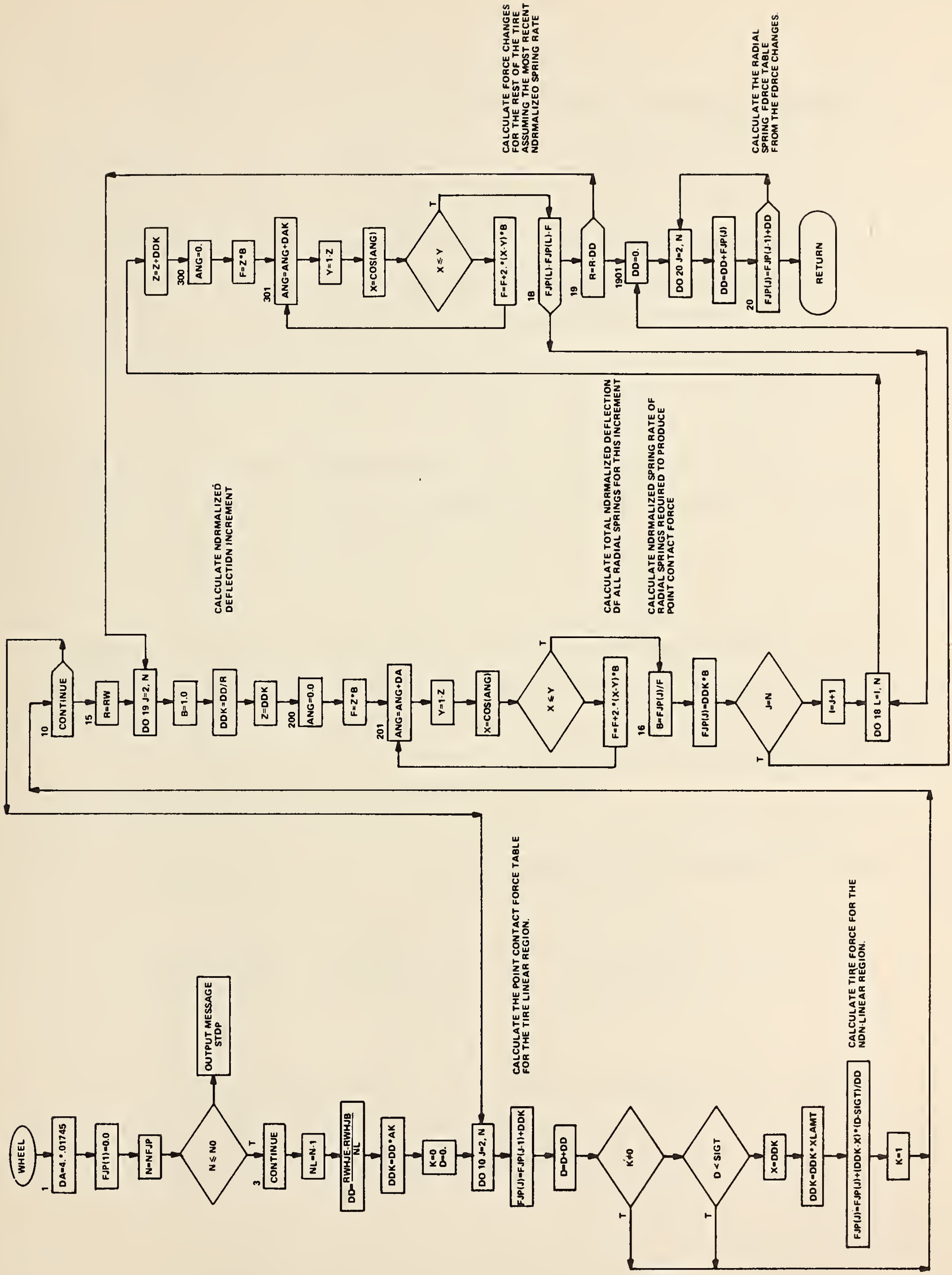
e. Common Variables Calculated:

None

f. Size:

$4FA)_{16} = 1274)_{10}$ bytes

g. Computational Procedure:


WHEEL
1
DA=4.*.01745
FJP(1)=0.0
N=NFJP
N ≤ NO
T
OUTPUT MESSAGE STOP
3
CONTINUE
NL=N-1
DD=RWHJE-RWHJB/NL
DDK=DD*AK
K=0 D=0.
DO 10 J=2, N
FJP(J)=FJP(J-1)+DDK
D=D+DD
K≠0
T
D < SIGT
T
X=DDK
DDK=DDK*XLAMT
FJP(J)=FJP(J)+(DDK-X)*(D-SIGT)/DD
K=1
CALCULATE THE POINT CONTACT FORCE TABLE FOR THE TIRE LINEAR REGION.
CALCULATE TIRE FORCE FOR THE NON-LINEAR REGION.
10
CONTINUE
15
R=RW
DO 19 J=2, N
B=1.0
DDK=DD/R
Z=DDK
200
ANG=0.0
F=Z*B
201
ANG=ANG+DA
Y=1-Z
X=COS(ANG)
X ≤ Y
T
F=F+2.*(X-Y)*B
16
B=FJP(J)/F
FJP(J)=DDK*B
J=N
T
I=J+1
DO 18 L=I, N
CALCULATE NORMALIZED DEFLECTION INCREMENT
CALCULATE TOTAL NORMALIZED DEFLECTION OF ALL RADIAL SPRINGS FOR THIS INCREMENT
CALCULATE NORMALIZED SPRING RATE OF RADIAL SPRINGS REQUIRED TO PRODUCE POINT CONTACT FORCE
Z=Z+DDK
300
ANG=0.
F=Z*B
301
ANG=ANG+DAK
Y=1-Z
X=COS(ANG)
X ≤ Y
T
F=F+2.*(X-Y)*B
18
FJP(L)=FJP(L)-F
19
R=R-DD
1901
DD=0.
DO 20 J=2, N
DD=DD+FJP(J)
20
FJP(J)=FJP(J-1)+DD
RETURN
CALCULATE FORCE CHANGES FOR THE REST OF THE TIRE ASSUMING THE MOST RECENT NORMALIZED SPRING RATE
CALCULATE THE RADIAL SPRING FORCE TABLE FROM THE FORCE CHANGES.

3.1.2 HVOSM-RD2 Program Stops and Messages

Program stops include both normal and abnormal stops. Normal stops occur when the cumulative simulated time (T) exceeds the desired final time (T1) as input in field 2 of card 101, or when the magnitudes of both the linear and angular velocities of the vehicle sprung mass are less than or equal to the input minimums (UVWMIN and PQRMIN, card 101, fields 6 and 7). When these stops occur, no message is output and the program attempts to read another set of data cards.

Abnormal stops occur when a condition is encountered that the program is not designed to handle or an unresolvable error has occurred. The first type of abnormal stop occurs when rollover of the vehicle is immenent. That is, when the vehicle has rolled to an angle of 90° in either direction.

The second program stop occurs when the barrier option is in effect (INDB $\neq$ 0) and the vehicle yaw angle (PSIT) is greater than 135°. This stop is necessary since the left rear corner of the vehicle is not tested for contact with the barrier.

Abnormal stops are also indicated by a non-zero value for the variable ISTOP. The following codes identify the type and location of error.

ISTOP = 4 — Subroutine TMCNST. The denominator of the expression used to calculate the value of PSIT after indexing of coordinate system is zero.

ISTOP = 5 — Subroutine TMCNST. The logic associated with coordinate system indexing has been unable to determine the correct quadrant for PSIT, PHIT or THETT.

ISTOP = 6 Subroutine TMCNST. The numerator in the expression for calculation of THETT after coordinate system indexing is zero.

ISTOP = 7 Subroutine TMCNST. The numerator in the expression for calculation of PHIT after coordinate system indexing is zero.

ISTOP = 30 Subroutine TMCNST. One of the recalculated Euler angles (PSIT, THETT, PHIT) has been computed as being very large (>3000 radians) after coordinate system indexing. A probable error has occurred.

When an ISTOP ≠ 0 condition is encountered, the program prints all output up to the time of the error, prints the value of ISTOP, terminates execution of the current run and attempts to read another set of data cards.

In subroutine INPUT, the following messages are printed if difficulties are encountered in reading the card data deck.

UNEXPECTED END OF FILE ENCOUNTERED IN STMT NO. 1 OF SUBROUTINE INPUT. LAST CARD READ WAS XXXX.

A CARD NUMBERED LESS THAN OR EQUAL TO ZERO WAS ENCOUNTERED IN SUBROUTINE INPUT. CARD IMAGE PRINTED ABOVE.

THE NUMBER OF CARDS READ IS ZERO.

A BLOCK NUMBER OF LESS THAN OR EQUAL TO ZERO HAS BEEN OBTAINED.

A BLOCK NUMBER LARGER THAN THE ALLOWED NUMBER HAS BEEN OBTAINED.

```
AN ERROR HAS OCCURRED IN STORING INPUT VALUES IN ONE
OF THE BLKXX SUBROUTINES.  THE CALLING ARGUMENTS
FROM INPUT ARE:  NBLK = XXXX   NBCRD = XXXX
NSEQ = XXXX   NCARD = XXXX   NERR = XXXX
```

In subroutine NLDFL, messages may be printed if the program determines that both constraints on the unloading curve (the input ratio of conserved to total energy, CONS, and the ratio of maximum to permanent displacement, SET) cannot be simultaneously satisfied. If this occurs, the energy ratio, CONS, is modified and a diagnostic is output.

In subroutine RUFRED, two messages may be printed if difficulties are encountered in reading road roughness data from FORTRAN device 4. They are:

```
END OF FILE ENCOUNTERED IN READ OF ROUGHNESS DATA
BEFORE NEND POINTS WERE READ.

NUMBER OF LAST ROUGHNESS DATA POINT IS GREATER THAN
THE ALLOWED 2200.  PROGRAM TERMINATED.
```

3.1.3 HVOSM-RD2 Program Listing

```
C           HIGHWAY VEHICLE OBJECT SIMULATION MODEL
C             MAIN ROUTINE
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL1O,DEL2O,DEL3O,PHIRO,DEL1OD,DEL2OD,DEL3OD,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1            (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1            (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
```

```
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1            (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
C       COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,
     1               COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,
     2               STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,
     2              SLOPE1,SLOPE2,XTRA(300)
      DIMENSION UI(4),VI(4),WI(4)
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)
      DIMENSION APITCH(4)
      EQUIVALENCE (APITCH(1),APTCH1)
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,
     2               DELTB,XINPT(100)
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,
     1              OMEGRC,AKRE,AKREP,OMEGRE,END3
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),AO(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
```

```
     2              A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)            MAIN1010
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),   MAIN1020
     1             YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),      MAIN1030
     2             AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,     MAIN1040
     3             CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),        MAIN1050
     4             YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT   MAIN1060
     5             ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,   MAIN1070
     6             XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,      MAIN1080
     7             AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,MAIN1090
     8             FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,MAIN1100
     9             NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,        MAIN1110
     A             RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,           MAIN1120
     B             SWORK,SPENGY,DISS,IPLN,ILOAD                          MAIN1130
      DIMENSION INDXPT(4)                                                MAIN1140
      EQUIVALENCE (INDXPT(1),I1)                                         MAIN1150
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,         MAIN1160
     1              DAPRB,DAPRE,DDAPR,NAPR                               MAIN1170
      DIMENSION APF(21),APR(21)                                          MAIN1180
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                  MAIN1190
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        MAIN1200
     1              AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),MAIN1210
     2              NCAMF,NCAMR,NDTHF,NDTHR                              MAIN1220
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       MAIN1230
     1               RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,       MAIN1240
     2               DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,      MAIN1250
     3               PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,      MAIN1260
     4               PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,          MAIN1270
     5               DTDD2,DTDD3,DTDD4,FJF(4),SNPF                       MAIN1280
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                           MAIN1290
     1              ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                            MAIN1300
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                      MAIN1310
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                  MAIN1320
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                  MAIN1330
     5              YCMP(6),ZCMP(6),PHICM(6)                             MAIN1340
      COMMON/BARSTR/XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),          MAIN1350
     1             ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),         MAIN1360
     2             FNSTI(3),AKST(3)                                      MAIN1370
      COMMON/HARDPT/ FRICF(4),UPT(4),VPT(4),WPT(4)                       MAIN1380
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF                               MAIN1390
C                                                                        MAIN1400
      COMMON/DRIVTT/ IPATHT,IDRIVE                                       MAIN1410
      COMMON/DRIVI/ DELPTH,XVP,GAIN,YPPE,NPPIO,YPPIO(4),SPI(30)          MAIN1420
      COMMON/NSTOP/ISTOP                                                 MAIN1430
      DIMENSION FJPP(35)                                                 MAIN1440
C                                                                        MAIN1450
C       SUBROUTINES DVDCHK AND DATE ARE RELATED TO THE OPERATING SYSTEMMAIN1460
C            AT OUR INSTALLATION                                         MAIN1470
C      SUBROUTINE DVDCHK CAN CAUSE HALT ON ATTEMPTED DIVIDE BY ZERO,     MAIN1480
C          EXPONENT OVERFLOW, AND MESSAGE ON EXPONENT UNDERFLOW.         MAIN1490
C        THE SERVICES GIVEN BY SUBROUTINE DVDCHK CAN NOW GIVEN BY        MAIN1500
C            FORTRAN EXTENDED ERROR HANDLING                             MAIN1510
```

```
C        SUBROUTINE DATE RETURNS THE CURRENT DATE IN EIGHT  CHARACTERS.
C
      CALL CLEAR(VHED(1),NPAGE(20))
      CALL CLEAR(DELPTH,SPI(30))
      CALL CLEAR(PHIO,PQRMIN)
      CALL CLEAR(YC1P,PSIFDO)
      CALL CLEAR(YBPO,XINPT(100))
      CALL CLEAR( AKFC,END3)
      CALL CLEAR(APFR(I,1),NAPR)
      CALL CLEAR(AKT(I),FJP(35,4))
      CALL CLEAR(ITIR(1),ITIR(4))
      CALL CLEAR(XIF,NDTHR)
      CALL CLEAR(YC3P,PHICM(6))
      CALL CLEAR(XSTIO(1),AKST(3))
      CALL CLEAR(DELG,IRUF)
      IPATHT = 0
      IDRIVE = 0
C       SET IDRIVE=1 IN DRIVID, AT START OF COMPUTATION, IF USING DRIVER
C     SUBROUTINE DRIVID  IS CALLED BY SUBROUTINE IDOUT
      CALL DVDCHK
    I CALL CLEAR(NEQ,DER(50))
      CALL CLEAR(SUMM,LLL)
      CALL CLEAR(FRSP(I),FRCP(4))
      CALL CLEAR(DPSINT,XMUGI(4))
      CALL CLEAR(XINDN,ENDEIN)
      CALL CLEAR(U1,XTRA(300))
      CALL CLEAR(X1P,CGH(4))
      CALL CLEAR(FN,ILOAD)
      CALL CLEAR(A234(1),A23(4))
      CALL CLEAR(XMURO2,SNPF)
      ISTOP = 0
      CALL INPUT
      CALL DATE(DADE)
      IF(IRUF.NE.0) CALL RUFRED(NEND,DELG,DGMAX,ZGP)
      IF(INDCRB.NE.1.AND.IRUF.EQ.0) GO TO 10
      NFJP = (RWHJE-RWHJB)/DRWHJ + I.2
      DO 11 I=1,4
      IF(I.EQ.1) GO TO I2
      IM = I-1
      DO I5 K=I,IM
      IF(ITIR(I).EQ.ITIR(K)) GO TO 16
   15 CONTINUE
   12 CALL WHEEL(AKT(I),SIGT(I),XLAMT(I),RWHJB,RWHJE,DRWHJ,NFJP,
     I           RW(I),FJPP,35)
      DO 13 J=1,NFJP
   13 FJP(J,I) = FJPP(J)
      GO TO 1I
   16 DO 17 J=1,NFJP
   17 FJP(J,I) = FJP(J,K)
   11 CONTINUE
   10 CONTINUE
```

```
      IF(ZF.EQ.0.0.AND.ZR.EQ.0.0) CALL INITEQ                           MAIN2030
      CALL IDOUT                                                        MAIN2040
      CALL CNSTNT                                                       MAIN2050
  100 DO 101  I=1,4                                                     MAIN2060
      LCB1(I) = .FALSE.                                                 MAIN2070
      LCB2(I) = .FALSE.                                                 MAIN2080
  101 CONTINUE                                                          MAIN2090
      PIO2 = .5*PI                                                      MAIN2100
      TPRINT = TO                                                       MAIN2110
      TPATH = TO                                                        MAIN2120
      THETMX = ABS(THMAX) * RAD                                         MAIN2130
      UVWM2 = UVWMIN**2                                                 MAIN2140
      PQRM2 = PQRMIN**2                                                 MAIN2150
       UVWM2 = SIGN(UVWM2,UVWMIN)                                       MAIN2160
       PQRM2 = SIGN(PQRM2,PQRMIN)                                       MAIN2170
      ICBHIT = 0                                                        MAIN2180
      JCBHIT = 0                                                        MAIN2190
      IHIT = 0                                                          MAIN2200
      IBHIT = 0                                                         MAIN2210
      JBHIT = 0                                                         MAIN2220
      IND = 0                                                           MAIN2230
      T = TO                                                            MAIN2240
      DT = DTCOMP                                                       MAIN2250
      NEQ = 22                                                          MAIN2260
      PSIMAX = 135.0*RAD                                                MAIN2270
      CALL PLOTTP(1)                                                    MAIN2280
      CALL OUTPUT(0)                                                    MAIN2290
    2 CALL PINT1(1,MODE,NEQ,T,DT,U,DU,EBAR)                             MAIN2300
      IF (ISTOP.NE. 0) GO TO 6                                          MAIN2310
    3 IF(TPRINT.GT.T+.1*DT) GO TO 4                                     MAIN2320
      CALL OUTPUT(1)                                                    MAIN2330
      TPRINT = TPRINT+DTPRNT                                            MAIN2340
      CALL PLOTTP(2)                                                    MAIN2350
    4 IDRIVE = 0                                                        MAIN2360
      IF(TPATH.GT. T+0.1*DT) GO TO 40                                   MAIN2370
C          SUBROUTINE DRIVER WILL DETERMINE PSI1 DURING FIRST INCREMENT MAIN2380
C          TO AVOID, INITIALIZE TPATH ABOVE AS TO+DELPTH                MAIN2390
      IDRIVE = 1                                                        MAIN2400
      TPATH = TPATH + DELPTH                                            MAIN2410
   40 NLDCTR = 0                                                        MAIN2420
      CALL PINT1(2,MODE,NEQ,T,DT,U,DU,EBAR)                             MAIN2430
      IF (ISTOP.NE. 0) GO TO 6                                          MAIN2440
C           THETTL,PHITL,PSITL ARE VALUES OF THETT,PHIT,PSIT FROM PREVIOUMAIN2450
C             TIME INTERVAL, USED TO TEST NEW ANGLES     IN SUBROUTINE TMCMAIN2460
      THETTL = THETT                                                    MAIN2470
      PHITL  = PHIT                                                     MAIN2480
      PSITL  = PSIT                                                     MAIN2490
      IF(INDB.NE.0) CALL EGYSUM                                         MAIN2500
      IF(T.GE.T1) GO TO 6                                               MAIN2510
      IF(U**2+V**2+W**2.LE.UVWM2.AND.P2+Q2+R2.LE.PQRM2) GO TO 6         MAIN2520
      IF(ABS(PHIT).GE.PIO2) GO TO 6                                     MAIN2530
```

```
      IF(INDB.NE.0.AND.PSIT.GE.PSIMAX) GO TO 6
      IF(IPATHT .NE. 0) GO TO 6
      IF(ABS(THETTP).LT.THETMX) GO TO 5
C      XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTX
C            XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING
C            THAT IS  THETN OR PHIN NOW .NE. 0.0
C            USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST
      THETN = THETT
      THETTP= 0.0
      PHIN  = PHIT
      PHITP = 0.0
      PSIN  = PSIT
      PSITP = 0.0
      XINDL = XINDN
      XINDN = XINDN + 1.0
C     IND=1  INDICATOR FOR RE-INITIALIZATION IN PINT1
      IND = 1
    5 IF(INDCRS.EQ.0) GO TO 56
   50 DO 51  I=1,4
      IF(.NOT.LCB2(I)) GO TO 53
   51 CONTINUE
      ICBHIT = 2
   52 IF(ICBHIT.EQ.JCBHIT) GO TO 56
      JCBHIT = ICBHIT
      IND = 1
      DT = DTCOMP
      GO TO 56
   53 DO 54  I=1,4
      IF(LCB1(I)) GO TO 55
   54 CONTINUE
      ICBHIT = 0
      GO TO 52
   55 ICBHIT = 1
      IHIT = 1
      IF (ICBHIT.EQ.JCBHIT) GO TO 56
      JCBHIT = ICBHIT
      IND = 0
      DT = DELTC
      GO TO 2
   56 IF(INDB.EQ.0) GO TO 58
      IF(IBHIT.EQ.JBHIT) GO TO 58
      IF(IBHIT.GT.JBHIT) GO TO 57
      JBHIT = IBHIT
      IF(ICBHIT.EQ.1) GO TO 58
      DT = DTCOMP
      IND = 0
      GO TO 2
   57 JBHIT = IBHIT
      DT = DELTB
      IND = 0
      GO TO 2
```

```
C                                                                       MAIN3050
   58 IF(IND.EQ.0) GO TO 3                                              MAIN3060
      IND = 0                                                           MAIN3070
      GO TO 2                                                           MAIN3080
C                                                                       MAIN3090
    6 CALL OUTPUT(1)                                                    MAIN3100
      CALL PLOTTP(3)                                                    MAIN3110
      IF(ISTOP .NE. 0) WRITE(6,59) ISTOP                                MAIN3120
   59 FORMAT(17H   ERROR, ISTOP = ,  I3)                                MAIN3130
C       CALL PLOTTP(3) CAUSES DISTINCTIVE RECORD ON TAPE FOR END OF RUN.MAIN3140
      GO TO 1                                                           MAIN3150
      END                                                               MAIN3160
```

```
      SUBROUTINE AREA
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1          (PSII(1),PSI1)
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),
     1            YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),
     2            AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,
     3            CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),
     4            YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT
     5            ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,
     6            XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,
     7            AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,
     8            FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,
     9            NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,
     A            RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,
     B            SWORK,SPENGY,DISS,IPLN,ILOAD
      DIMENSION INDXPT(4)
      EQUIVALENCE (INDXPT(1),I1)
      DIMENSION XTTT(4),ZTTT(4),SLOP(4)
      LOGICAL S2GS3,S2GS4,S3GS4
      XRIP = XRI
      YRIP = YRI
      ZRIP = ZRI
      AINTP = AINTI
      IF(IPT.EQ.3)GO TO 62
      XMIN = 1.0E30
    1 DO 2 J=1,4
      IJ = INDXPT(J)
      XTTT(J) = AMTX(1,1)*XNN(IJ)+AMTX(1,2)*YNN(IJ)+AMTX(1,3)*ZNN(IJ)
      ZTTT(J) = AMTX(3,1)*XNN(IJ)+AMTX(3,2)*YNN(IJ)+AMTX(3,3)*ZNN(IJ)
      IF(XTTT(J).GT.XMIN)GO TO 2
      JMN = J
      XMIN = XTTT(J)
```

```
    2 CONTINUE                                                          AREA0500
      IF(JMN.EQ.1)GO TO 3                                               AREA0510
      TMP1 = XTTT(1)                                                    AREA0520
      TMP2 = ZTTT(1)                                                    AREA0530
      XTTT(1) = XTTT(JMN)                                               AREA0540
      ZTTT(1) = ZTTT(JMN)                                               AREA0550
      XTTT(JMN) = TMP1                                                  AREA0560
      ZTTT(JMN) = TMP2                                                  AREA0570
      IJ = INDXPT(1)                                                    AREA0580
      INDXPT(1) = INDXPT(JMN)                                           AREA0590
      INDXPT(JMN) = IJ                                                  AREA0600
    3 DO 4 J=2,4                                                        AREA0610
      TMP1 = AMAX1(XTTT(J)-XTTT(1),1.0E-30)                             AREA0620
      TMP2 = ZTTT(J)-ZTTT(1)                                            AREA0630
      SLOP(J) = TMP2/TMP1                                               AREA0640
    4 CONTINUE                                                          AREA0650
      S2GS3 = SLOP(2).GT.SLOP(3)                                        AREA0660
      S2GS4 = SLOP(2).GT.SLOP(4)                                        AREA0670
      S3GS4 = SLOP(3).GT.SLOP(4)                                        AREA0680
      IF(S2GS3)GO TO 5                                                  AREA0690
      IF(S2GS4)GO TO 62                                                 AREA0700
      JMN = 4                                                           AREA0710
      IF(S3GS4)GO TO 6                                                  AREA0720
      JMN = 3                                                           AREA0730
      GO TO 6                                                           AREA0740
    5 IF(.NOT.S2GS4)GO TO 62                                            AREA0750
      JMN = 3                                                           AREA0760
      IF(S3GS4)GO TO 6                                                  AREA0770
      JMN = 4                                                           AREA0780
    6 IJ = INDXPT(2)                                                    AREA0790
      INDXPT(2) = INDXPT(JMN)                                           AREA0800
      INDXPT(JMN) = IJ                                                  AREA0810
   62 TMPX = XNN(I1)-XNN(I2)                                            AREA0820
      TMPY = YNN(I1)-YNN(I2)                                            AREA0830
      TMPZ = ZNN(I1)-ZNN(I2)                                            AREA0840
    7 AIP = TMPY*CGB-TMPZ*CBB                                           AREA0850
      BIP = TMPZ*CAB-TMPX*CGB                                           AREA0860
      CIP = TMPX*CBB-TMPY*CAB                                           AREA0870
      DIP = SQRT(AIP**2+BIP**2+CIP**2)                                  AREA0880
      S3I = SQRT(TMPX**2+TMPY**2+TMPZ**2)                               AREA0890
      S2I = 0.0                                                         AREA0900
      IF(IPT.EQ.4) S2I = ABS(AIP*(XNN(I4)-XNN(I2))+BIP*(YNN(I4)-YNN(I2))AREA0910
     1                    +CIP*(ZNN(I4)-ZNN(I2)))/DIP                   AREA0920
      S1I = ABS(AIP*(XNN(I3)-XNN(I2))+BIP*(YNN(I3)-YNN(I2))+CIP*(ZNN(I3)AREA0930
     1      -ZNN(I2)))/DIP                                              AREA0940
      AINTI = .5*S3I*(S1I+S2I)                                          AREA0950
    8 AC1 = .5*(XNN(I1)+XNN(I2))-XNN(I3)                                AREA0960
      BC1 = .5*(YNN(I1)+YNN(I2))-YNN(I3)                                AREA0970
      CC3 = .5*(ZNN(I1)+ZNN(I2))-ZNN(I3)                                AREA0980
      AC2 = .5*(XNN(I1)+XNN(I3))-XNN(I2)                                AREA0990
      BC2 = .5*(YNN(I1)+YNN(I3))-YNN(I2)                                AREA1000
```

```
      CC4 = .5*(ZNN(I1)+ZNN(I3))-ZNN(I2)                          AREA1010
      NNNN = 1                                                    AREA1020
    9 AC1P = BC1*CGB-CC3*CBB                                      AREA1030
      BC1P = CC3*CAB-AC1*CGB                                      AREA1040
      CC1P = AC1*CBB-BC1*CAB                                      AREA1050
      G1 = AC1P*XNN(I3)+BC1P*YNN(I3)+CC1P*ZNN(I3)                 AREA1060
      AC2P = BC2*CGB-CC4*CBB                                      AREA1070
      BC2P = CC4*CAB-AC2*CGB                                      AREA1080
      CC2P = AC2*CBB-BC2*CAB                                      AREA1090
      G2 = AC2P*XNN(I2)+BC2P*YNN(I2)+CC2P*ZNN(I2)                 AREA1100
      XMTX(1,1) = CAB                                             AREA1110
      XMTX(1,2) = CBB                                             AREA1120
      XMTX(1,3) = CGB                                             AREA1130
      XMTX(1,4) = RB                                              AREA1140
      XMTX(2,1) = AC1P                                            AREA1150
      XMTX(2,2) = BC1P                                            AREA1160
      XMTX(2,3) = CC1P                                            AREA1170
      XMTX(2,4) = G1                                              AREA1180
      XMTX(3,1) = AC2P                                            AREA1190
      XMTX(3,2) = BC2P                                            AREA1200
      XMTX(3,3) = CC2P                                            AREA1210
      XMTX(3,4) = G2                                              AREA1220
      CALL SIMSOL(XMTX,3,3)                                       AREA1230
      XQI = XMTX(1,4)                                             AREA1240
      YQI = XMTX(2,4)                                             AREA1250
      ZQI = XMTX(3,4)                                             AREA1260
   10 IF(IPT.EQ.4) GO TO (11,12),NNNN                             AREA1270
      XRI = XQI                                                   AREA1280
      YRI = YQI                                                   AREA1290
      ZRI = ZQI                                                   AREA1300
      GO TO 13                                                    AREA1310
   11 XPI = XQI                                                   AREA1320
      YPI = YQI                                                   AREA1330
      ZPI = ZQI                                                   AREA1340
      AC1 = .5*(XNN(I1)+XNN(I2))-XNN(I4)                          AREA1350
      BC1 = .5*(YNN(I1)+YNN(I2))-YNN(I4)                          AREA1360
      CC3 = .5*(ZNN(I1)+ZNN(I2))-ZNN(I4)                          AREA1370
      AC2 = .5*(XNN(I1)+XNN(I4))-XNN(I2)                          AREA1380
      BC2 = .5*(YNN(I1)+YNN(I4))-YNN(I2)                          AREA1390
      CC4 = .5*(ZNN(I1)+ZNN(I4))-ZNN(I2)                          AREA1400
      I3 = I4                                                     AREA1410
      NNNN = 2                                                    AREA1420
      GO TO 9                                                     AREA1430
   12 TEMP = S2I/(S1I+S2I)                                        AREA1440
      XRI = XPI+TEMP*(XQI-XPI)                                    AREA1450
      YRI = YPI+TEMP*(YQI-YPI)                                    AREA1460
      ZRI = ZPI+TEMP*(ZQI-ZPI)                                    AREA1470
   13 IF(AINTP.NE.0.0)GO TO 14                                    AREA1480
      XRIP = XRI                                                  AREA1490
      YRIP = YRI                                                  AREA1500
      ZRIP = ZRI                                                  AREA1510
```

```
14 TEMP = .5*(AINTI+AINTP)                                   AREA1520
   SDEN = SDEN+TEMP                                          AREA1530
   IF(NUNLD.NE.0)GO TO 15                                    AREA1540
   SXR = SXR+TEMP*(XRI+XRIP)*.5                              AREA1550
   SYR = SYR+TEMP*(YRI+YRIP)*.5                              AREA1560
   SZR = SZR+TEMP*(ZRI+ZRIP)*.5                              AREA1570
15 RETURN                                                    AREA1580
   END                                                       AREA1590
```

```
      SUBROUTINE BLK01(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK10010
C            HVOSM-RD2 VERSION                                          BLK10020
C            REVISED OCTOBER 1975   CALSPAN CORPORATION                 BLK10030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK10040
     1            NPAGE(20)                                             BLK10050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK10060
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      BLK10070
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      BLK10080
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK10090
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  BLK10100
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     BLK10110
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   BLK10120
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   BLK10130
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    BLK10140
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  BLK10150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK10160
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN BLK10170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK10180
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,   BLK10190
     2            PSIFIO,PSIFDO                                         BLK10200
      DIMENSION YCIP(2)                                                 BLK10210
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK10220
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),    BLK10230
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,   BLK10240
     2               DELTB,XINPT(100)                                   BLK10250
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       BLK10260
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),BLK10270
     2               NCAMF,NCAMR,NDTHF,NDTHR                            BLK10280
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          BLK10290
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                          BLK10300
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                    BLK10310
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                BLK10320
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                BLK10330
     5               YCMP(6),ZCMP(6),PHICM(6)                           BLK10340
      DIMENSION DUM(18)                                                 BLK10350
      DATA NBS/4/                                                       BLK10360
      NBT = NBCRD+1                                                     BLK10370
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK10380
      GO TO(100,101,102,103,104),NBT                                    BLK10390
      GO TO 98                                                          BLK10400
  100 IF(NCARD.NE.100) GO TO 98                                         BLK10410
      DO 10 I=1,18                                                      BLK10420
   10 HED(I) = DUM(I)                                                   BLK10430
      GO TO 99                                                          BLK10440
  101 IF(NCARD.NE.101) GO TO 98                                         BLK10450
      TO      = DUM(1)                                                  BLK10460
      T1      = DUM(2)                                                  BLK10470
      DTCOMP = DUM(3)                                                   BLK10480
      DTPRNT = DUM(4)                                                   BLK10490
```

```
      THMAX  = DUM(5)
      UVWMIN = DUM(6)
      PQRMIN = DUM(7)
      GO TO 99
  102 IF(NCARD.NE.102) GO TO 98
      ISUS = IFIX(DUM(1))
      INDCRB = IFIX(DUM(2))
      NCRBSL = IFIX(DUM(3))
      DELTC = DUM(4)
      INDB = IFIX(DUM(5))
      DELTB = DUM(6)
      IF(INDCRB.NE.0) NPAGE(5) = 1
      IF(INDB.EQ.0) GO TO 99
      NPAGE(17) = 1
      NPAGE(18) = 1
      NPAGE(19) = 1
      GO TO 99
  103 IF(NCARD.NE.103) GO TO 98
      MODE = DUM(1)
      EBAR = DUM(2)
      EM   = DUM(3)
      AAA  = DUM(4)
      HMAX = DUM(5)
      HMIN = DUM(6)
      BET  = DUM(7)     COLL = DUM(8)
      GO TO 99          NUM = DUM(9)
  104 IF(NCARD.NE.104) GO TO 98
      NPAGE(4) = DUM(1)
      NPAGE(6) = DUM(2)
      NPAGE(7) = DUM(3)
      NPAGE(8) = DUM(4)
      NPAGE(9) = DUM(5)
      NPAGE(10)= DUM(6)
      NPAGE(14)= DUM(7)
      GO TO 99
   98 NERR = 1
   99 RETURN
      END
```

```
      SUBROUTINE BLK02(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK20010
C              HVOSM-RD2 VERSION                                        BLK20020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               BLK20030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK20040
     1             NPAGE(20)                                            BLK20050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK20060
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     BLK20070
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     BLK20080
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK20090
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, BLK20100
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    BLK20110
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  BLK20120
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  BLK20130
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   BLK20140
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 BLK20150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK20160
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN BLK20170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK20180
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK20190
     2             PSIFIO,PSIFDO                                        BLK20200
      DIMENSION YC1P(2)                                                 BLK20210
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK20220
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),    BLK20230
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,   BLK20240
     2               DELTB,XINPT(100)                                   BLK20250
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,     BLK20260
     1              OMEGRC,AKRE,AKREP,OMEGRE,END3                       BLK20270
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,        BLK20280
     1               DAPRB,DAPRE,DDAPR,NAPR                             BLK20290
      DIMENSION APF(21),APR(21)                                         BLK20300
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                 BLK20310
      COMMON /INSUS/ X1F,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       BLK20320
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),BLK20330
     2               NCAMF,NCAMR,NDTHF,NDTHR                            BLK20340
      COMMON/BARSTR/XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),         BLK20350
     1              ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),       BLK20360
     2              FNSTI(3),AKST(3)                                    BLK20370
      DIMENSION DUM(18)                                                 BLK20380
      DATA NBS/14/                                                      BLK20390
      NBT=NBCRD+1                                                       BLK20400
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK20410
      GO TO(200,201,202,203,204,205,206,207,208,209,210,211,            BLK20420
     1      212,213,214),NBT                                            BLK20430
      GO TO 98                                                          BLK20440
  200 IF(NCARD.NE.200) GO TO 98                                         BLK20450
      DO 10 I=1,18                                                      BLK20460
   10 VHED(I) = DUM(I)                                                  BLK20470
      GO TO 99                                                          BLK20480
  201 IF(NCARD.NE.201) GO TO 98                                         BLK20490
```

```
      XMS   = DUM(1)                                    BLK20500
      XMUF  = DUM(2)                                    BLK20510
      XMUR  = DUM(3)                                    BLK20520
      XIX   = DUM(4)                                    BLK20530
      XIY   = DUM(5)                                    BLK20540
      XIZ   = DUM(6)                                    BLK20550
      XIXZ  = DUM(7)                                    BLK20560
      XIR   = DUM(8)                                    BLK20570
      XIF = DUM(9)                                      BLK20580
      GO TO 99                                          BLK20590
  202 IF(NCARD.NE.202) GO TO 98                         BLK20600
      A    = DUM(1)                                     BLK20610
      B    = DUM(2)                                     BLK20620
      TF   = DUM(3)                                     BLK20630
      TR   = DUM(4)                                     BLK20640
      RHO = DUM(5)                                      BLK20650
      TS = DUM(6)                                       BLK20660
      RHOF = DUM(7)                                     BLK20670
      TSF = DUM(8)                                      BLK20680
      G = 386.4                                         BLK20690
      IF(DUM(9).NE.0.0) G = DUM(9)                      BLK20700
      GO TO 99                                          BLK20710
  203 IF(NCARD.NE.203) GO TO 98                         BLK20720
      X1 = DUM(1)                                       BLK20730
      Y1 = DUM(2)                                       BLK20740
      Z1 = DUM(3)                                       BLK20750
      X2 = DUM(4)                                       BLK20760
      Y2 = DUM(5)                                       BLK20770
      Z2 = DUM(6)                                       BLK20780
      DO 30 J=1,6                                       BLK20790
      IF(DUM(J).NE.0.0) NPAGE(16) = 1                   BLK20800
   30 CONTINUE                                          BLK20810
      ZF = DUM(7)                                       BLK20820
      ZR = DUM(8)                                       BLK20830
      GO TO 99                                          BLK20840
  204 IF(NCARD.NE.204) GO TO 98                         BLK20850
      AKF     = DUM(1)                                  BLK20860
      AKFC    = DUM(2)                                  BLK20870
      AKFCP   = DUM(3)                                  BLK20880
      AKFE    = DUM(4)                                  BLK20890
      AKFEP   = DUM(5)                                  BLK20900
      XLAMF   = DUM(6)                                  BLK20910
      OMEGFC = DUM(7)                                   BLK20920
      OMEGFE = DUM(8)                                   BLK20930
      GO TO 99                                          BLK20940
  205 IF(NCARD.NE.205)GO TO 98                          BLK20950
      AKR     = DUM(1)                                  BLK20960
      AKRC    = DUM(2)                                  BLK20970
      AKRCP   = DUM(3)                                  BLK20980
      AKRE    = DUM(4)                                  BLK20990
      AKREP   = DUM(5)                                  BLK21000
```

```
      XLAMR  = DUM(6)                                                  BLK21010
      OMEGRC = DUM(7)                                                  BLK21020
      OMEGRE = DUM(8)                                                  BLK21030
      GO TO 99                                                         BLK21040
  206 IF(NCARD.NE.206) GO TO 98                                        BLK21050
      CF   = DUM(1)                                                    BLK21060
      CFP  = DUM(2)                                                    BLK21070
      EPSF = DUM(3)                                                    BLK21080
      CR   = DUM(4)                                                    BLK21090
      CRP  = DUM(5)                                                    BLK21100
      EPSR = DUM(6)                                                    BLK21110
      GO TO 99                                                         BLK21120
  207 IF(NCARD.NE.207) GO TO 98                                        BLK21130
      RF = DUM(1)                                                      BLK21140
      RR = DUM(2)                                                      BLK21150
      AKRS = DUM(3)                                                    BLK21160
      AKDS = DUM(4)                                                    BLK21170
      AKDS1 = DUM(5)                                                   BLK21180
      AKDS2 = DUM(6)                                                   BLK21190
      AKDS3 = DUM(7)                                                   BLK21200
      GO TO 99                                                         BLK21210
  208 IF(NCARD.NE.208) GO TO 98                                        BLK21220
      XIPS  = DUM(1)                                                   BLK21230
      CPSP  = DUM(2)                                                   BLK21240
      OMGPS = DUM(3)                                                   BLK21250
      AKPS  = DUM(4)                                                   BLK21260
      EPSPS = DUM(5)                                                   BLK21270
      XPS   = DUM(6)                                                   BLK21280
      GO TO 99                                                         BLK21290
  209 IF(NCARD.NE.209.OR.NSEQ.NE.0) GO TO 98                           BLK21300
      DELB = DUM(1)                                                    BLK21310
      DELE = DUM(2)                                                    BLK21320
      DDEL = DUM(3)                                                    BLK21330
      NDTHF = DUM(4)                                                   BLK21340
      NDTHR = DUM(5)                                                   BLK21350
      NDEL = (DELE-DELB)/DDEL +1                                       BLK21360
      NCRDS = (NDEL-1)/9 + 1                                           BLK21370
      CALL TREAD(NCARD,NCRDS,NDEL,50,PHIC,NERR)                        BLK21380
      IF(NERR.NE.0) GO TO 98                                           BLK21390
      IF(ISUS.EQ.1) CALL TREAD(NCARD,NCRDS,NDEL,50,PHIRC,NERR)         BLK21400
      IF(NERR.NE.0) GO TO 98                                           BLK21410
      IF(NDTHF.NE.0) CALL TREAD(NCARD,NCRDS,NDEL,50,DTHF,NERR)         BLK21420
      IF(NERR.NE.0) GO TO 98                                           BLK21430
      IF(NDTHR.NE.0) CALL TREAD(NCARD,NCRDS,NDEL,50,DTHR,NERR)         BLK21440
      IF(NERR.NE.0) GO TO 98                                           BLK21450
      GO TO 99                                                         BLK21460
  210 IF(NCARD.NE.210.OR.NSEQ.NE.0) GO TO 98                           BLK21470
      DAPFB = DUM(1)                                                   BLK21480
      DAPFE = DUM(2)                                                   BLK21490
      DDAPF = DUM(3)                                                   BLK21500
      NAPF = (DAPFE-DAPFB)/DDAPF + 1                                   BLK21510
```

```
      NCRDS = (NAPF-1)/9 + 1
      CALL TREAD(NCARD,NCRDS,NAPF,21,APF,NERR)
      IAPFR(1) = 1
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  211 IF(NCARD.NE.211.OR.NSEQ.NE.0) GO TO 98
      DAPRB = DUM(1)
      DAPRE = DUM(2)
      DDAPR = DUM(2)
      NAPR = (DAPRE-DAPRB)/DDAPR + 1
      NCRDS = (NAPF-1)/9 + 1
      CALL TREAD(NCARD,NCRDS,NAPR,21,APR,NERR)
      IAPFR(2) = 1
      IF(NERR.NE.0)GO TO 98
      GO TO 99
  212 IF(NCARD.NE.212) GO TO 98
      XVF = DUM(1)
      XVR = DUM(2)
      YV  = DUM(3)
      ZVT = DUM(4)
      ZVB = DUM(5)
      AKV = DUM(6)
      GO TO 99
  213 IF(NCARD.NE.213) GO TO 98
      XSTIO(1) = DUM(1)
      XSTIO(2) = DUM(2)
      XSTIO(3) = DUM(3)
      YSTIO(1) = DUM(4)
      YSTIO(2) = DUM(5)
      YSTIO(3) = DUM(6)
      GO TO 99
  214 IF(NCARD.NE.214) GO TO 98
      ZSTIO(1) = DUM(1)
      ZSTIO(2) = DUM(2)
      ZSTIO(3) = DUM(3)
      AKST(1) = DUM(4)
      AKST(2) = DUM(5)
      AKST(3) = DUM(6)
      GO TO 99
   98 NERR =  1
   99 RETURN
      END
```

```
      SUBROUTINE BLK03(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK30010
C             HVOSM-RD2 VERSION                                         BLK30020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                BLK30030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK30040
     1             NPAGE(20)                                            BLK30050
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK30060
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK30070
     2             PSIFIO,PSIFDO                                        BLK30080
      DIMENSION YCIP(2)                                                 BLK30090
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK30100
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   BLK30110
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     BLK30120
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          BLK30130
      DIMENSION DUM(18),TDUM(9,4)                                       BLK30140
      DATA NBS/2/                                                       BLK30150
      NBT = NBCRD+1                                                     BLK30160
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK30170
      GO TO(300,301,302),NBT                                            BLK30180
      GO TO 98                                                          BLK30190
  300 IF(NCARD.NE.300) GO TO 98                                         BLK30200
      DO 10 I=1,18                                                      BLK30210
   10 THED(I) = DUM(I)                                                  BLK30220
      GO TO 99                                                          BLK30230
  301 IF(NCARD.NE.301.OR.NSEQ.NE.0) GO TO 98                            BLK30240
      ITIR(1) = DUM(1)                                                  BLK30250
      ITIR(2) = DUM(2)                                                  BLK30260
      ITIR(3) = DUM(3)                                                  BLK30270
      ITIR(4) = DUM(4)                                                  BLK30280
      RWHJE = DUM(5)                                                    BLK30290
      DRWHJ = DUM(6)                                                    BLK30300
      N = MAX0(ITIR(1),ITIR(2),ITIR(3),ITIR(4))                         BLK30310
      CALL T2READ(NCARD,9,9,N,TDUM,NERR)                                BLK30320
      IF(NERR.NE.0) GO TO 98                                            BLK30330
      DO 20 I=1,4                                                       BLK30340
      J = ITIR(I)                                                       BLK30350
      AKT(I) = TDUM(1,J)                                                BLK30360
      SIGT(I) = TDUM(2,J)                                               BLK30370
      XLAMT(I) = TDUM(3,J)                                              BLK30380
      A0(I) = TDUM(4,J)                                                 BLK30390
      A1(I) = TDUM(5,J)                                                 BLK30400
      A2(I) = TDUM(6,J)                                                 BLK30410
      A3(I) = TDUM(7,J)                                                 BLK30420
      A4(I) = TDUM(8,J)                                                 BLK30430
      OMEGT(I) = TDUM(9,J)                                              BLK30440
   20 CONTINUE                                                          BLK30450
      GO TO 99                                                          BLK30460
  302 IF(NCARD.NE.302) GO TO 98                                         BLK30470
      DO 30 I=1,4                                                       BLK30480
      J = ITIR(I)                                                       BLK30490
```

```
                                                                BLK30500
      AMU(I) = DUM(J)                                           BLK30510
      RW(I) = DUM(J+4)                                          BLK30520
   30 CONTINUE                                                  BLK30530
      GO TO 99                                                  BLK30540
   98 NERR = 1.0                                                BLK30550
   99 RETURN                                                    BLK30560
      END
```

```
      SUBROUTINE BLK04(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)
C              HVOSM-RD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1            NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      DIMENSION DUM(18)
      DATA NBS/1/
      NBT = NBCRD+1
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98
      GO TO(400,401),NBT
      GO TO 98
  400 IF(NCARD.NE.400) GO TO 98
      DO 10 I=1,18
   10 CHED(I) = DUM(I)
      GO TO 99
  401 IF(NCARD.NE.401.OR.NSEQ.NE.0) GO TO 98
      TB     = DUM(1)
      TE     = DUM(2)
      TINCR  = DUM(3)
      NTBL1  = IFIX(DUM(4))
      NTBL2  = IFIX(DUM(5))
      NTBL3  = IFIX(DUM(6))
      IF(NTBL2.NE.0.OR.NTBL3.NE.0) NPAGE(13) = 1
      IF(NTBL1+NTBL2+NTBL3.EQ.0) GO TO 99
      NT = IFIX((TE-TB)/TINCR + 1.2)
      NCRDS = (NT-1)/9 + 1
      IF(NTBL1.EQ.0) GO TO 11
      CALL TREAD(NCARD,NCRDS,NT,50,PSIF,NERR)
      IF(NERR.NE.0) GO TO 98
   11 IF(NTBL2.EQ.0) GO TO 12
      CALL TREAD(NCARD,NCRDS,NT,50,TQF,NERR)
      IF(NERR.NE.0) GO TO 98
   12 IF(NTBL3.EQ.0) GO TO 99
      CALL TREAD(NCARD,NCRDS,NT,50,TQR,NERR)
      IF(NERR.EQ.0) GO TO 99
   98 NERR = 1
   99 RETURN
```

END

```
      SUBROUTINE BLK05(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK50010
C           HVOSM-RD2 VERSION                                           BLK50020
C           REVISED OCTOBER 1975   CALSPAN CORPORATION                  BLK50030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK50040
     1           NPAGE(20)                                              BLK50050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK50060
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,       BLK50070
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                       BLK50080
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK50090
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  BLK50100
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,      BLK50110
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   BLK50120
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,    BLK50130
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    BLK50140
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  BLK50150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK50160
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  BLK50170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK50180
     1           CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,    BLK50190
     2           PSIFIO,PSIFDO                                          BLK50200
      DIMENSION YCIP(2)                                                 BLK50210
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK50220
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),    BLK50230
     1              SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,    BLK50240
     2              DELTB,XINPT(100)                                    BLK50250
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          BLK50260
     1              ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                           BLK50270
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                     BLK50280
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                 BLK50290
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                 BLK50300
     5              YCMP(6),ZCMP(6),PHICM(6)                            BLK50310
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF                              BLK50320
      DIMENSION DUM(18)                                                 BLK50330
      DATA NBS/13/                                                      BLK50340
      NBT = NBCRD+1                                                     BLK50350
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK50360
      GO TO (500,501,502,503,504,505,506,507,508,509,510,               BLK50370
     1    511,512,513),NBT                                              BLK50380
      GO TO 98                                                          BLK50390
  500 IF(NCARD.NE.500) GO TO 98                                         BLK50400
      DO 10 I=1,18                                                      BLK50410
   10 GHED(I) = DUM(I)                                                  BLK50420
      GO TO 99                                                          BLK50430
  501 IF(NCARD.NE.501) GO TO 98                                         BLK50440
      IF(NZTAB.LT.1) NZTAB=1                                            BLK50450
      I = 1                                                             BLK50460
      GO TO 20                                                          BLK50470
  502 IF(NCARD.NE.502) GO TO 98                                         BLK50480
      IF(NZTAB.LT.2) NZTAB = 2                                          BLK50490
```

```
      I = 2
      GO TO 20
  503 IF(NCARD.NE.503) GO TO 98
      IF(NZTAB.LT.3) NZTAB = 3
      I = 3
      GO TO 20
  504 IF(NCARD.NE.504) GO TO 98
      IF(NZTAB.LT.4) NZTAB = 4
      I = 4
      GO TO 20
  505 IF(NCARD.NE.505) GO TO 98
      NZTAB = 5
      I = 5
   20 NPAGE(15) = 1
      XB(I)    = DUM(1)
      XE(I)    = DUM(2)
      XINCR(I) = DUM(3)
      YB(I)    = DUM(4)
      YE(I)    = DUM(5)
      YINCR(I) = DUM(6)
      NBX(I) = IFIX(DUM(7))
      NBY(I) = IFIX(DUM(8))
      NZ5T = IFIX(DUM(9))
      NNBX = NBX(I)
      NNBY = NBY(I)
      IF(NZ5T.EQ.1) GO TO 21
      NNX = IFIX((XE(I)-XB(I))/XINCR(I) + 1.2)
      NNY = IFIX((YE(I)-YB(I))/YINCR(I) + 1.2)
      NX(I) = NNX
      NY(I) = NNY
      CALL TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
   21 NNX = IFIX(DUM(3))
      NNY = IFIX(DUM(6))
      NX(I) = NNX
      NY(I) = NNY
      NZ5 = 1
      CALL TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  506 IF(NCARD.NE.506) GO TO 98
      DO 30 J=1,5
   30 AMUG(J) = DUM(J)
      GO TO 99
  507 IF(NCARD.NE.507) GO TO 98
      YC1P = DUM(1)
      YC2P = DUM(2)
      YC3P = DUM(3)
      YC4P = DUM(4)
      YC5P = DUM(5)
```

```
      YC6P = DUM(6)                                          BLK51010
      AMUC = DUM(7)                                          BLK51020
      GO TO 99                                               BLK51030
  508 IF(NCARD.NE.508) GO TO 98                              BLK51040
      ZC2P = DUM(1)                                          BLK51050
      ZC3P = DUM(2)                                          BLK51060
      ZC4P = DUM(3)                                          BLK51070
      ZC5P = DUM(4)                                          BLK51080
      ZC6P = DUM(5)                                          BLK51090
      GO TO 99                                               BLK51100
  509 IF(NCARD.NE.509) GO TO 98                              BLK51110
      PHIC1 = DUM(1)                                         BLK51120
      PHIC2 = DUM(2)                                         BLK51130
      PHIC3 = DUM(3)                                         BLK51140
      PHIC4 = DUM(4)                                         BLK51150
      PHIC5 = DUM(5)                                         BLK51160
      PHIC6 = DUM(6)                                         BLK51170
      GO TO 99                                               BLK51180
  510 IF(NCARD.NE.510) GO TO 98                              BLK51190
      YBPO = DUM(1)                                          BLK51200
      ZBTP = DUM(2)                                          BLK51210
      ZBBP = DUM(3)                                          BLK51220
      DELYBP = DUM(4)                                        BLK51230
      AMUB = DUM(5)                                          BLK51240
      EPSV = DUM(6)                                          BLK51250
      EPSB = DUM(7)                                          BLK51260
      SET = DUM(8)                                           BLK51270
      CONS = DUM(9)                                          BLK51280
      GO TO 99                                               BLK51290
  511 IF(NCARD.NE.511) GO TO 98                              BLK51300
      DO 40 I=1,9                                            BLK51310
   40 SIGR(I) = DUM(I)                                       BLK51320
      GO TO 99                                               BLK51330
  512 IF(NCARD.NE.512) GO TO 98                              BLK51340
      SIGR(10) = DUM(1)                                      BLK51350
      SIGR(11) = DUM(2)                                      BLK51360
      GO TO 99                                               BLK51370
  513 IF(NCARD.NE.513) GO TO 98                              BLK51380
      DELG = DUM(1)                                          BLK51390
      NEND = IFIX(DUM(2))                                    BLK51400
      IRUF = 1                                               BLK51410
      NPAGE(8) = 1                                           BLK51420
      DGMAX = (NEND-1)*DELG                                  BLK51430
      GO TO 99                                               BLK51440
   98 NERR = 1                                               BLK51450
   99 RETURN                                                 BLK51460
      END                                                    BLK51470
```

```
      SUBROUTINE BLK06(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      DIMENSION DUM(18)
      DATA NBS/3/
      NBT = NBCRD+1
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98
      GO TO (600,601,602,603),NBT
      GO TO 98
  600 IF(NCARD.NE.600) GO TO 98
      DO 10 I=1,18
   10 SHED(I) = DUM(I)
      GO TO 99
  601 IF(NCARD.NE.601) GO TO 98
      PHIO    = DUM(1)
      THETAO = DUM(2)
      PSIO    = DUM(3)
      PO      = DUM(4)
      QO      = DUM(5)
      RO = DUM(6)
      PSIFIO = DUM(7)
      PSIFDO = DUM(8)
      GO TO 99
  602 IF(NCARD.NE.602) GO TO 98
      XCOP = DUM(1)
      YCOP = DUM(2)
      ZCOP = DUM(3)
```

```
      UO    = DUM(4)                                                    BLK60500
      VO    = DUM(5)                                                    BLK60510
      WO    = DUM(6)                                                    BLK60520
      GO TO 99                                                          BLK60530
  603 IF(NCARD.NE.603) GO TO 98                                         BLK60540
      DEL10  = DUM(1)                                                   BLK60550
      DEL20 = DUM(2)                                                    BLK60560
      IF(ISUS.EQ.2) PHIFO = DUM(2)                                      BLK60570
      DEL30  = DUM(3)                                                   BLK60580
      PHIRO = DUM(4)                                                    BLK60590
      IF(ISUS.EQ.1) DEL40 = DUM(4)                                      BLK60600
      DEL10D = DUM(5)                                                   BLK60610
      DEL20D = DUM(6)                                                   BLK60620
      IF(ISUS.EQ.2) PHIFOD = DUM(6)                                     BLK60630
      DEL30D = DUM(7)                                                   BLK60640
      PHIROD = DUM(8)                                                   BLK60650
      IF(ISUS.EQ.1) DEL40D = DUM(8)                                     BLK60660
      GO TO 99                                                          BLK60670
   98 NERR = 1                                                          BLK60680
   99 RETURN                                                            BLK60690
      END                                                               BLK60700
```

```
      SUBROUTINE CLEAR(A,B)
C         CLEARS (SETS TO ZERO) A BLOCK OF STORAGE IDENTIFIED BY THE
C          ADDRESSES OF THE TWO ARGUMENTS.
C
C         CALL CLEAR(P,Q)
C         WILL CAUSE ALL BYTES TO BE SET TO ZERO FROM ADDRESS
C         P THROUGH THE FULL-WORD AT ADDRESS Q
C
      DIMENSION A(1),B(1)
      B(1) = 1.0
      I = 0
   10 IF(B(1).EQ.0.0) RETURN
      I=I+1
      A(I) = 0.0
      END
```

```
      SUBROUTINE CNSTNT
C              HVOSM-RD2 VERSION     -
C              REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSF1,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
```

```
8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  CNST0500
9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   CNST0510
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   CNST0520
1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  CNST0530
2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,CNST0540
3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2CNST0550
4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,CNST0560
5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           CNST0570
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  CNST0580
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        CNST0590
 COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                      CNST0600
1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    CNST0610
2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    CNST0620
3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        CNST0630
 LOGICAL LCB1,LCB2                                                  CNST0640
C  COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST   CNST0650
 COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,     CNST0660
1               COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,   CNST0670
2               STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN    CNST0680
 COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),     CNST0690
1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,    CNST0700
2               DELTB,XINPT(100)                                    CNST0710
 COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),    CNST0720
1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),      CNST0730
2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)           CNST0740
 COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),   CNST0750
1              YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),     CNST0760
2              AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,    CNST0770
3              CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),       CNST0780
4              YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT  CNST0790
5              ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,  CNST0800
6              XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,     CNST0810
7              AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,CNST0820
8              FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,CNST0830
9              NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,       CNST0840
A              RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,          CNST0850
B              SWORK,SPENGY,DISS,IPLN,ILOAD                         CNST0860
 DIMENSION INDXPT(4)                                                CNST0870
 EQUIVALENCE (INDXPT(1),I1)                                         CNST0880
 COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        CNST0890
1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),CNST0900
2               NCAMF,NCAMR,NDTHF,NDTHR                             CNST0910
 COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       CNST0920
1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,      CNST0930
2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,     CNST0940
3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,     CNST0950
4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,         CNST0960
5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                      CNST0970
 COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                           CNST0980
1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                           CNST0990
2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                     CNST1000
```

```
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,
     5              YCMP(6),ZCMP(6),PHICM(6)
      NPAGE(1) = 1
      NPAGE(2) = 1
      NPAGE(3) = 1
      NPAGE(11) = 1
      NPAGE(12) = 1
C  OTHER OUTPUT PAGE INDICATORS EITHER READ OR SET IN BLKXX SUBROUTINES
      PI = 3.141592653D0
      TWOPI =  2.0*PI
      PIO2  =  0.5 * PI
      PIO4  =  0.25* PI
      RAD =  .0174532925D0
      DO 7 I=1,4
      A12(I) = A1(I)/A2(I)
      A23(I) = A2(I)*A3(I)/A1(I)
      A234(I) = A2(I)*A3(I)/A4(I)
      OMT2M1(I) = OMEGT(I)*A1(I)*A2(I)*(OMEGT(I)-1.0)
      OMT2A2(I) = (OMEGT(I)*A2(I)*A3(I)*(A4(I)-OMEGT(I)*A2(I)))
     1            /(A4(I)*(OMT2M1(I)-A0(I)))
    7 CONTINUE
      TRO2 = 0.5*TR
      TFO2 = 0.5*TF
      AMUF = A*XMUF
      BMUR = B*XMUR
      XMUFO2 = 0.5*XMUF
      AXMFO2 = A*XMUFO2
      XMTFO4 = XMUFO2*TFO2
      TM4 = 0.25*XMUF*TF
      GMSTMP = 0.5*XMS*G/(A+B)
      AO2APB = A*GMSTMP
      BO2APB = B*GMSTMP
      GAM1 = AMUF-BMUR
      SUMM = XMS+XMUF+XMUR
      DEL1 = DEL10
      DEL1D = DEL10D
      DEL3 = DEL30
      DEL3D = DEL30D
      IF(ISUS.EQ.1) GO TO 10
      ZRO = ZR+RHO
      TSO2 = 0.5*TS
      RHO2 = RHO*RHO
      RHOMUR = RHO*XMUR
      RHMR2 = RHO*RHOMUR
      RTR = RR/TS
      BROMUR = RHOMUR*B
      RHMR2I = RHMR2+XIR
      PHIR = PHIRO
      PHIRD = PHIROD
   10 IF(ISUS.NE.0) GO TO 20
```

```
      ZPR = ZF+RHO                                                      CNST1520
      RRTS = RR*TS                                                      CNST1530
      TIZ = XMUF*(A*A+TFO2*TFO2)+BMUR                                   CNST1540
      XIZR = XIZ+XIR                                                    CNST1550
   20 IF(ISUS.EQ.2) GO TO 30                                            CNST1560
      RFTF = RF/(TF*TF)                                                 CNST1570
      DEL2 = DEL20                                                      CNST1580
      DEL2D = DEL20D                                                    CNST1590
   30 IF(ISUS.NE.2) GO TO 40                                            CNST1600
      ZFO = ZF+RHOF                                                     CNST1610
      TSFO2 = 0.5*TSF                                                   CNST1620
      RHOF2 = RHOF*RHOF                                                 CNST1630
      RHFMUF = RHOF*XMUF                                                CNST1640
      RHF2MF = RHOF*RHFMUF                                              CNST1650
      RF2MFI = RHF2MF+XIF                                               CNST1660
      RTF = RF/TSF                                                      CNST1670
      PHIF = PHIFO                                                      CNST1680
      PHIFD = PHIFOD                                                    CNST1690
   40 IF(ISUS.NE.1) GO TO 50                                            CNST1700
      RRTR = RR/(TR*TR)                                                 CNST1710
      XMURO2 = 0.5*XMUR                                                 CNST1720
      BXMRO2 = B*XMURO2                                                 CNST1730
      XMTRO4 = XMURO2*TRO2                                              CNST1740
      DEL4 = DEL40                                                      CNST1750
      DEL4D = DEL40D                                                    CNST1760
   50 CONTINUE                                                          CNST1770
      U = UO                                                            CNST1780
      V = VO                                                            CNST1790
      W = WO                                                            CNST1800
      P = PO*RAD                                                        CNST1810
      Q = QO*RAD                                                        CNST1820
      R = RO*RAD                                                        CNST1830
      THETTP = 0.0                                                      CNST1840
      PHITP = 0.0                                                       CNST1850
      PSITP = 0.0                                                       CNST1860
      THETN = THETAO*RAD                                                CNST1870
      PHIN = PHIO*RAD                                                   CNST1880
      PSIN = PSIO*RAD                                                   CNST1890
C      XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTXCNST1900
C            XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING CNST1910
C            THAT IS  THETN OR PHIN NOW .NE. 0.0                        CNST1920
C            USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST      CNST1930
      IF( THETN.NE.0.0 .OR.  PHIN.NE. 0.0) XINDN = 10.0                 CNST1940
      THETTL = THETN                                                    CNST1950
      PHITL  = PHIN                                                     CNST1960
      PSITL  = PSIN                                                     CNST1970
      XCP = XCOP                                                        CNST1980
      YCP = YCOP                                                        CNST1990
      ZCP = ZCOP                                                        CNST2000
      PHIC1R = PHIC1*RAD                                                CNST2010
      PHIC2R = PHIC2*RAD                                                CNST2020
```

```
      PHIC3R = PHIC3*RAD                                    CNST2030
      PHIC4R = PHIC4*RAD                                    CNST2040
      PHIC5R = PHIC5*RAD                                    CNST2050
      PHIC6R = PHIC6*RAD                                    CNST2060
      TANPC2 = TAN(PHIC2R)                                  CNST2070
      TANPC1 = TAN(PHIC1R)                                  CNST2080
      TANPC3 = TAN(PHIC3R)                                  CNST2090
      TANPC4 = TAN(PHIC4R)                                  CNST2100
      TANPC5 = TAN(PHIC5R)                                  CNST2110
      TANPC6 = TAN(PHIC6R)                                  CNST2120
      NCB = NCRBSL-1                                        CNST2130
      GO TO (72,73,74,75,76),NCB                            CNST2140
   72 PHICLR = PHIC2R                                       CNST2150
      YCLP = YC2P                                           CNST2160
      ZCLP = ZC2P                                           CNST2170
      TANPCL = TANPC2                                       CNST2180
      YC3P = 1.0E+6                                         CNST2190
      ZC3P = ZC2P+SIGN(1.0,ZC2P)                            CNST2200
      GO TO 71                                              CNST2210
   73 PHICLR = PHIC3R                                       CNST2220
      YCLP = YC3P                                           CNST2230
      ZCLP = ZC3P                                           CNST2240
      TANPCL = TANPC3                                       CNST2250
      YC4P = 1.0E+6                                         CNST2260
      ZC4P = ZC3P+SIGN(1.0,ZC3P)                            CNST2270
      GO TO 71                                              CNST2280
   74 PHICLR = PHIC4R                                       CNST2290
      YCLP = YC4P                                           CNST2300
      ZCLP = ZC4P                                           CNST2310
      TANPCL = TANPC4                                       CNST2320
      YC5P = 1.0E+6                                         CNST2330
      ZC5P = ZC4P+SIGN(1.0,ZC4P)                            CNST2340
      GO TO 71                                              CNST2350
   75 PHICLR = PHIC5R                                       CNST2360
      YCLP = YC5P                                           CNST2370
      ZCLP = ZC5P                                           CNST2380
      TANPCL = TANPC5                                       CNST2390
      YC6P = 1.0E+6                                         CNST2400
      ZC6P = ZC5P+SIGN(1.0,ZC5P)                            CNST2410
      GO TO 71                                              CNST2420
   76 PHICLR = PHIC6R                                       CNST2430
      YCLP = YC6P                                           CNST2440
      ZCLP = ZC6P                                           CNST2450
      TANPCL = TANPC6                                       CNST2460
   71 CONTINUE                                              CNST2470
      PSIFI  = PSIFIO*RAD                                   CNST2480
      PSIFID = PSIFDO                                       CNST2490
      DO 9 I=1,5                                            CNST2500
      DO 9 J=1,4                                            CNST2510
    9 PSBDRY(J,I) = PSBDRO(J,I) * RAD                       CNST2520
      XCPN(1) = XVF                                         CNST2530
```

```
      YCPN(1) = YV                                                      CNST2540
      ZCPN(1) = 0.0                                                     CNST2550
      XCPN(2) = XVR                                                     CNST2560
      YCPN(2) = YV                                                      CNST2570
      ZCPN(2) = 0.0                                                     CNST2580
      XCPN(3) = XVF                                                     CNST2590
      YCPN(3) = -YV                                                     CNST2600
      ZCPN(3) = 0.0                                                     CNST2610
      AA1(1) = 1.0                                                      CNST2620
      AA1(2) = 1.0                                                      CNST2630
      AA1(3) = 1.0                                                      CNST2640
      AA1(7) = 1.0                                                      CNST2650
      AA1(8) = 1.0                                                      CNST2660
      AA1(9) = 1.0                                                      CNST2670
      AA1(14) = 1.0                                                     CNST2680
      AA1(15) = 1.0                                                     CNST2690
      BB1(4) = 1.0                                                      CNST2700
      BB1(5) = 1.0                                                      CNST2710
      BB1(6) = 1.0                                                      CNST2720
      BB1(10) = 1.0                                                     CNST2730
      BB1(11) = 1.0                                                     CNST2740
      BB1(16) = 1.0                                                     CNST2750
      BB1(17) = 1.0                                                     CNST2760
      CC1(12) = 1.0                                                     CNST2770
      CC1(13) = 1.0                                                     CNST2780
      CC2(1) = 1.0                                                      CNST2790
      CC2(2) = 1.0                                                      CNST2800
      RR1(1) = XVF                                                      CNST2810
      RR1(2) = XVF                                                      CNST2820
      RR1(3) = XVF                                                      CNST2830
      RR1(4) = YV                                                       CNST2840
      RR1(5) = YV                                                       CNST2850
      RR1(6) = YV                                                       CNST2860
      RR1(7) = XVR                                                      CNST2870
      RR1(8) = XVR                                                      CNST2880
      RR1(9) = XVF                                                      CNST2890
      RR1(10) = -YV                                                     CNST2900
      RR1(11) = -YV                                                     CNST2910
      RR1(12) = ZVT                                                     CNST2920
      RR1(13) = ZVB                                                     CNST2930
      RR1(14) = XVF                                                     CNST2940
      RR1(15) = XVR                                                     CNST2950
      RR1(16) = YV                                                      CNST2960
      RR1(17) = -YV                                                     CNST2970
      AA2(6) = 1.0                                                      CNST2980
      BB2(3) = 1.0                                                      CNST2990
      BB2(9) = 1.0                                                      CNST3000
      CC2(4) = 1.0                                                      CNST3010
      CC2(5) = 1.0                                                      CNST3020
      CC2(7) = 1.0                                                      CNST3030
      CC2(8) = 1.0                                                      CNST3040
```

```
      CC2(10) = 1.0                                                     CNST3050
      CC2(11) = 1.0                                                     CNST3060
      RR2(1) = ZVT                                                      CNST3070
      RR2(2) = ZVB                                                      CNST3080
      RR2(3) = YV                                                       CNST3090
      RR2(4) = ZVT                                                      CNST3100
      RR2(5) = ZVB                                                      CNST3110
      RR2(6) = XVR                                                      CNST3120
      RR2(7) = ZVT                                                      CNST3130
      RR2(8) = ZVB                                                      CNST3140
      RR2(9) = -YV                                                      CNST3150
      RR2(10) = ZVT                                                     CNST3160
      RR2(11) = ZVB                                                     CNST3170
      YBPT = YBPO                                                       CNST3180
      YBPTP = YBPO                                                      CNST3190
      RETURN                                                            CNST3200
      END                                                               CNST3210
```

```
      SUBROUTINE CRBIMP(I)                                              CRBI0010
C             HVOSM-RD2 VERSION                                         CRBI0020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                CRBI0030
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          CRBI0040
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,   CRBI0050
     2            PSIFIO,PSIFDO                                         CRBI0060
      DIMENSION YCIP(2)                                                 CRBI0070
      EQUIVALENCE (YCIP(1),YC1P)                                        CRBI0080
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             CRBI0090
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))CRBI0100
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),CRBI0110
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       CRBI0120
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      CRBI0130
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        CRBI0140
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          CRBI0150
     6            (PSIFID,VAR(22))                                      CRBI0160
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      CRBI0170
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))CRBI0180
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),      CRBI0190
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    CRBI0200
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   CRBI0210
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         CRBI0220
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     CRBI0230
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         CRBI0240
     1            (DER(10),DPHIFD)                                      CRBI0250
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       CRBI0260
     1            (DER(14),DDEL4D)                                      CRBI0270
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,CRBI0280
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   CRBI0290
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4), CRBI0300
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     CRBI0310
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  CRBI0320
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  CRBI0330
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     CRBI0340
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      CRBI0350
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4), CRBI0360
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4) CRBI0370
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      CRBI0380
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     CRBI0390
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   CRBI0400
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  CRBI0410
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       CRBI0420
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   CRBI0430
     1            (PSII(1),PSI1)                                        CRBI0440
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,CRBI0450
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         CRBI0460
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  CRBI0470
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    CRBI0480
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    CRBI0490
```

```
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPCRBI0500
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     CRBI0510
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    CRBI0520
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  CRBI0530
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   CRBI0540
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3, CRBI0550
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  CRBI0560
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,CRBI0570
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2CRBI0580
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,CRBI0590
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           CRBI0600
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                CRBI0610
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)      CRBI0620
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                    CRBI0630
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,  CRBI0640
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),  CRBI0650
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)      CRBI0660
      LOGICAL LCB1,LCB2                                                CRBI0670
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),  CRBI0680
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),    CRBI0690
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)         CRBI0700
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                         CRBI0710
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                         CRBI0720
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                   CRBI0730
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,               CRBI0740
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,               CRBI0750
     5               YCMP(6),ZCMP(6),PHICM(6)                          CRBI0760
      DIMENSION FJPP(35)                                               CRBI0770
      DO 20 N=1,35                                                     CRBI0780
   20 FJPP(N) = FJP(N,I)                                               CRBI0790
    1 SNPSI = SIN(PSII(I))                                             CRBI0800
      CSPSI = COS(PSII(I))                                             CRBI0810
      SNPHI = SIN(PHII(I))                                             CRBI0820
      CSPHI = COS(PHII(I))                                             CRBI0830
      SFRX(I) = 0.0                                                    CRBI0840
      SFRY(I) = 0.0                                                    CRBI0850
      SFRZ(I) = 0.0                                                    CRBI0860
      TTAJ21 = CSPHI * SNPSI                                           CRBI0870
      TTAJ31 = SNPHI * SNPSI                                           CRBI0880
      AJMTX(1,2) = -SNPSI                                              CRBI0890
      AJMTX(2,2) = CSPHI * CSPSI                                       CRBI0900
      AJMTX(3,2) = SNPHI * CSPSI                                       CRBI0910
      XJ = -26.0*RAD                                                   CRBI0920
    2 DO 11  J=1,53                                                    CRBI0930
      THTJ = 4.0*XJ                                                    CRBI0940
      STJ = SIN(THTJ)                                                  CRBI0950
      CTJ = COS(THTJ)                                                  CRBI0960
      AJMTX(1,1) = CTJ*CSPSI                                           CRBI0970
      AJMTX(2,1) = TTAJ21*CTJ + SNPHI*STJ                              CRBI0980
      AJMTX(3,1) = TTAJ31*CTJ - CSPHI*STJ                              CRBI0990
      AJMTX(1,3) = CSPHI*STJ                                           CRBI1000
```

```
      AJMTX(2,3) = TTAJ21*STJ - SNPHI*CTJ                              CRBI1010
      AJMTX(3,3) = TTAJ31*STJ + CSPHI*CTJ                              CRBI1020
C       AJMTX ANGLE SEQUENCE IS PHI,PSI,THJ                            CRBI1030
    3 DO 8  K=1,3                                                      CRBI1040
    4 DO 7  L=1,3                                                      CRBI1050
      BMTX(K,L) = 0.0                                                  CRBI1060
    5 DO 6  M=1,3                                                      CRBI1070
      BMTX(K,L) = BMTX(K,L)+AMTX(K,M)*AJMTX(M,L)                       CRBI1080
    6 CONTINUE                                                         CRBI1090
    7 CONTINUE                                                         CRBI1100
    8 CONTINUE                                                         CRBI1110
      HJ = -ZP(I)/BMTX(3,3)                                            CRBI1120
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 800                           CRBI1130
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1140
      IF(YJP.LT.YC1P) GO TO 203                                        CRBI1150
  800 HJ = (-ZP(I)+(YP(I)-YC1P)*TANPC1)/(BMTX(3,3)-BMTX(2,3)*TANPC1)   CRBI1160
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 805                           CRBI1170
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1180
      ZJP = ZP(I)+BMTX(3,3)*HJ                                         CRBI1190
      IF(YJP.GE.YC1P.AND.YJP.LE.YC2P.AND.(ABS(ZJP).LE.ABS(ZC2P)).AND.  CRBI1200
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC2P))) GO TO 204                   CRBI1210
  805 HJ = (ZC2P-ZP(I)+(YP(I)-YC2P)*TANPC2)/(BMTX(3,3)-BMTX(2,3)*      CRBI1220
     1      TANPC2)                                                    CRBI1230
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 810                           CRBI1240
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1250
      ZJP = ZP(I)+BMTX(3,3)*HJ                                         CRBI1260
      IF(YJP.GT.YC2P.AND.YJP.LE.YC3P.AND.(ABS(ZJP).LE.ABS(ZC3P)).AND.  CRBI1270
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC3P))) GO TO 204                   CRBI1280
  810 IF(NCRBSL.EQ.2) GO TO 10                                         CRBI1290
      HJ = (ZC3P-ZP(I)+(YP(I)-YC3P)*TANPC3)/(BMTX(3,3)-BMTX(2,3)*TANPC3)CRBI1300
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 815                           CRBI1310
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1320
      ZJP = ZP(I)+BMTX(3,3)*HJ                                         CRBI1330
      IF(YJP.GT.YC3P.AND.YJP.LE.YC4P.AND.(ABS(ZJP).LE.ABS(ZC4P)).AND.  CRBI1340
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC4P))) GO TO 204                   CRBI1350
  815 IF(NCRBSL.EQ.3) GO TO 10                                         CRBI1360
      HJ= (ZC4P-ZP(I)+(YP(I)-YC4P)*TANPC4)/(BMTX(3,3)-BMTX(2,3)*TANPC4) CRBI1370
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 820                           CRBI1380
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1390
      ZJP = ZP(I)+BMTX(3,3)*HJ                                         CRBI1400
      IF(YJP.GT.YC4P.AND.YJP.LE.YC5P.AND.(ABS(ZJP).LE.ABS(ZC5P)).AND.  CRBI1410
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC5P))) GO TO 204                   CRBI1420
  820 IF(NCRBSL.EQ.4) GO TO 10                                         CRBI1430
      HJ= (ZC5P-ZP(I)+(YP(I)-YC5P)*TANPC5)/(BMTX(3,3)-BMTX(2,3)*TANPC5) CRBI1440
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 825                           CRBI1450
      YJP = YP(I)+BMTX(2,3)*HJ                                         CRBI1460
      ZJP = ZP(I)+BMTX(3,3)*HJ                                         CRBI1470
      IF(YJP.GT.YC5P.AND.YJP.LE.YC6P.AND.(ABS(ZJP).LE.ABS(ZC6P)).AND.  CRBI1480
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC6P))) GO TO 204                   CRBI1490
  825 IF(NCRBSL.EQ.5) GO TO 10                                         CRBI1500
      HJ= (ZC6P-ZP(I)+(YP(I)-YC6P)*TANPC6)/(BMTX(3,3)-BMTX(2,3)*TANPC6) CRBI1510
```

```
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 10                            CRBI1520
      YJP = YP(I)+BMTX(2,3)*HJ                                          CRBI1530
      IF(YJP.LT.YC6P) GO TO 10                                          CRBI1540
  203 ZJP = ZP(I)+BMTX(3,3)*HJ                                          CRBI1550
  204 XJP = XP(I)+BMTX(1,3)*HJ                                          CRBI1560
      CAJ = (XP(I)-XJP)/HJ                                              CRBI1570
      CBJ = (YP(I)-YJP)/HJ                                              CRBI1580
      CGJ = (ZP(I)-ZJP)/HJ                                              CRBI1590
      CALL INTRPL(FJPP,RWHJB,RWHJE,DRWHJ,RW(I)-HJ,FJ)                   CRBI1600
      SFRX(I) = SFRX(I)+FJ*CAJ                                          CRBI1610
      SFRY(I) = SFRY(I)+FJ*CBJ                                          CRBI1620
      SFRZ(I) = SFRZ(I)+FJ*CGJ                                          CRBI1630
   10 XJ = XJ+RAD                                                       CRBI1640
   11 CONTINUE                                                          CRBI1650
      FR(I) = SQRT(SFRX(I)**2+SFRY(I)**2+SFRZ(I)**2)                    CRBI1660
      IF(FR(I).NE.0.0)GO TO 110                                         CRBI1670
      CAR(I) = 0.0                                                      CRBI1680
      CBR(I) = 0.0                                                      CRBI1690
      CGR(I) = 0.0                                                      CRBI1700
      HI(I) = RW(I)                                                     CRBI1710
      RETURN                                                            CRBI1720
  110 CAR(I) = -SFRX(I)/FR(I)                                           CRBI1730
      CBR(I) = -SFRY(I)/FR(I)                                           CRBI1740
      CGR(I) = -SFRZ(I)/FR(I)                                           CRBI1750
      HI(I) = RW(I)-FR(I)/AKT(I)                                        CRBI1760
      IF(HI(I).GT.RW(I)-SIGT(I)) GO TO 111                              CRBI1770
      HI(I) = RW(I)-(FR(I)/AKT(I)+SIGT(I)*(XLAMT(I)-1.0))/XLAMT(I)      CRBI1780
  111 TYGP = YP(I)+HI(I)*CBR(I)                                         CRBI1790
      PHGI(I) = 0.0                                                     CRBI1800
      IF(TYGP.LE.YC1P)GO TO 12                                          CRBI1810
      IF(TYGP.GT.YC1P.AND.TYGP.LE.YC2P) GO TO 900                       CRBI1820
      GO TO 905                                                         CRBI1830
  900 PHGI(I) = PHIC1R                                                  CRBI1840
      GO TO 12                                                          CRBI1850
  905 IF(NCRBSL.EQ.2) GO TO 970                                         CRBI1860
      IF(TYGP.GT.YC2P.AND.TYGP.LE.YC3P) GO TO 910                       CRBI1870
      GO TO 915                                                         CRBI1880
  910 PHGI(1) = PHIC2R                                                  CRBI1890
      GO TO 12                                                          CRBI1900
  915 IF(NCRBSL.EQ.3) GO TO 970                                         CRBI1910
      IF(TYGP.GT.YC3P.AND.TYGP.LE.YC4P) GO TO 920                       CRBI1920
      GO TO 925                                                         CRBI1930
  920 PHGI(I) = PHIC3R                                                  CRBI1940
      GO TO 12                                                          CRBI1950
  925 IF(NCRBSL.EQ.4) GO TO 970                                         CRBI1960
      IF(TYGP.GT.YC4P.AND.TYGP.LE.YC5P) GO TO 930                       CRBI1970
      GO TO 935                                                         CRBI1980
  930 PHGI(I) = PHIC4R                                                  CRBI1990
      GO TO 12                                                          CRBI2000
  935 IF(NCRBSL.EQ.5) GO TO 970                                         CRBI2010
      IF(TYGP.GT.YC5P.AND.TYGP.LE.YC6P) GO TO 940                       CRBI2020
```

```
      GO TO 970
  940 PHGI(I) = PHIC5R
      GO TO 12
  970 PHGI(I) = PHICLR
   12 TCI = CAR(I)*CBYW(I)-CBR(I)*CAYW(I)
      TAI = CBR(I)*CGYW(I)-CGR(I)*CBYW(I)
      TBI = CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
      CPG(I) = COS(PHGI(I))
      SPG(I) = SIN(PHGI(I))
      TERM3 = TBI*SPG(I)
      TERM4 = TCI*CPG(I)
      DN1 = TAI * (TERM3 - TERM4)
      DN2 = -TBI*TERM4 - (TAI**2 + TCI**2)*SPG(I)
      DN3 =  (TAI**2 + TBI**2)*CPG(I)  + TCI*TERM3
      TERM5 = SQRT(DN1**2 + DN2**2 + DN3**2)
      SPG(I) = (-DN2/TERM5)
      PHGI(I) = ARSIN(SPG(I))
      THGI(I) = ATAN (DN1/DN3)
      CPG(I) = COS(PHGI(I))
      TERM6 = SQRT(DN1**2 + DN3**2)
      CTG(I) = DN3/TERM6
      STG(I) = DN1/TERM6
C     STORE XGPP(I),YGPP(I) AS WELL AS ZGPP(I) IN CRBIMP FOR PLOTTING
      XGPP(I) = XP(I) + HI(I) * CAR(I)
      YGPP(I) = TYGP
      ZGPP(I) = ZP(I)+HI(I)*CGR(I)
      RETURN
      END
```

```
      SUBROUTINE DAUX                                                   DAUX0010
C              HVOSM-RD2 VERSION                                        DAUX0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               DAUX0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    DAUX0040
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     DAUX0050
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     DAUX0060
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,DAUX0070
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, DAUX0080
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    DAUX0090
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  DAUX0100
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  DAUX0110
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   DAUX0120
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 DAUX0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),DAUX0140
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN   DAUX0150
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          DAUX0160
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  DAUX0170
     2             PSIFIO,PSIFDO                                        DAUX0180
      DIMENSION YCIP(2)                                                 DAUX0190
      EQUIVALENCE (YCIP(1),YC1P)                                        DAUX0200
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             DAUX0210
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))DAUX0220
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),DAUX0230
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       DAUX0240
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      DAUX0250
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        DAUX0260
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          DAUX0270
     6            (PSIFID,VAR(22))                                      DAUX0280
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      DAUX0290
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))DAUX0300
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       DAUX0310
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    DAUX0320
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   DAUX0330
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         DAUX0340
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     DAUX0350
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         DAUX0360
     1            (DER(10),DPHIFD)                                      DAUX0370
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       DAUX0380
     1            (DER(14),DDEL4D)                                      DAUX0390
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,DAUX0400
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  DAUX0410
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),DAUX0420
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    DAUX0430
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), DAUX0440
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), DAUX0450
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    DAUX0460
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     DAUX0470
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),DAUX0480
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)DAUX0490
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),       DAUX0500
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     DAUX0510
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   DAUX0520
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  DAUX0530
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                        DAUX0540
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),    DAUX0550
     1            (PSII(1),PSI1)                                         DAUX0560
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,DAUX0570
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          DAUX0580
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  DAUX0590
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    DAUX0600
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    DAUX0610
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPDAUX0620
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     DAUX0630
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    DAUX0640
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  DAUX0650
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   DAUX0660
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   DAUX0670
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  DAUX0680
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,DAUX0690
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2DAUX0700
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,DAUX0710
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           DAUX0720
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  DAUX0730
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        DAUX0740
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                      DAUX0750
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    DAUX0760
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    DAUX0770
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        DAUX0780
      LOGICAL LCB1,LCB2                                                  DAUX0790
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                 DAUX0800
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,        DAUX0810
     2              SLOPE1,SLOPE2,XTRA(300)                              DAUX0820
      DIMENSION UI(4),VI(4),WI(4)                                        DAUX0830
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       DAUX0840
      DIMENSION APITCH(4)                                                DAUX0850
      EQUIVALENCE (APITCH(1),APTCH1)                                     DAUX0860
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),     DAUX0870
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,    DAUX0880
     2               DELTB,XINPT(100)                                    DAUX0890
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        DAUX0900
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),DAUX0910
     2               NCAMF,NCAMR,NDTHF,NDTHR                             DAUX0920
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       DAUX0930
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,      DAUX0940
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,     DAUX0950
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,     DAUX0960
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,         DAUX0970
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                      DAUX0980
      COMMON/NSTOP/ISTOP                                                 DAUX0990
      DIMENSION DISP(4),VEL(4)                                           DAUX1000
```

```
C                                                                       DAUX1010
      IF(ISTOP.NE.0) RETURN                                             DAUX1020
      CALL TMCNST                                                       DAUX1030
      IS1 = ISUS+1                                                      DAUX1040
      D12D22 = DEL1*DEL1 + DEL2*DEL2                                    DAUX1050
      GO TO (10,11,12),IS1                                              DAUX1060
   10 XIXP = XMUF*(ZF*(ZF+D1PD2)+.5*D12D22) + XMUR*ZRD3*ZRD3R           DAUX1070
      XIYP = X1XP+RHOMUR*ZRD3R                                          DAUX1080
      XIZP = TIZ+TIZ2                                                   DAUX1090
      XIXZP = AMUF*ZFD12 - BMUR*ZRD3                                    DAUX1100
      XIYZP = TM4*D1MD2-RHOMUR*PHIR*ZRD3R                               DAUX1110
      GAM2 = XMUF*ZFD12+XMUR*ZRD3R                                      DAUX1120
      GAM3 = GAM2-RHOMUR                                                DAUX1130
      GAM4 = XIYZP+RHMR2*PHIR                                           DAUX1140
      GAM5 = TIZ-XMUF*TFO2*TF-TIZ2                                      DAUX1150
      GAM6 = XMUF*DD1P2+2.0*TG61                                        DAUX1160
      GAM7 = XMUF*(ZF*DD1P2+DEL1*DEL1D+DEL2*DEL2D)+2.0*ZRD3*TG61         DAUX1170
      GAM8 = 2.0*(TM4*DD1M2-RPR*TG61)                                   DAUX1180
      GAM9 = AMUF*DD1P2 - 2.0*B*TG61                                    DAUX1190
      GO TO 3                                                           DAUX1200
   11 XIXP = XMUFO2*(ZFD1*ZFD1+ZFD2*ZFD2) + XMURO2*(ZRD3*ZRD3+ZRD4*ZRD4)DAUX1210
      XIYP = XIXP                                                       DAUX1220
      XIZP = XMUF*(A*A+TFO2*TFO2) +XMUR*(B*B+TRO2*TRO2)                 DAUX1230
      X1XZP = AXMFO2*(ZFD1+ZFD2) -  BXMRO2*(ZRD3+ZRD4)                  DAUX1240
      XIYZP = XMTFO4*D1MD2 + XMTRO4*D3MD4                               DAUX1250
      GAM2 = XMUF*ZFD12 + XMUR*ZRD34                                    DAUX1260
      GAM5 = XMUF*(A*A-TFO2*TFO2) + XMUR*(B*B-TRO2*TRO2)                DAUX1270
      GAM6 = XMUF*DD1P2 + XMUR*DD3P4                                    DAUX1280
      GAM7 = XMUF*(ZFD1*DEL1D+ZFD2*DEL2D) + XMUR*(ZRD3*DEL3D+ZRD4*DEL4D)DAUX1290
      GAM8 = XMUF*TFO2*DD1M2 + XMUR*TRO2*DD3M4                          DAUX1300
      GAM9 = AMUF*DD1P2 - BMUR*DD3P4                                    DAUX1310
      GO TO 3                                                           DAUX1320
   12 XIXP = XMUF*ZFD1*ZFD1 + RHFMUF*ZFD1 + XMUR*ZRD3*ZRD3 + RHOMUR*ZRD3DAUX1330
      XIYP = XIXP + RHFMUF*ZFD1RF + RHOMUR*ZRD3R                        DAUX1340
      XIZP = XMUF*(A*A+RFPF*RFPF) + XMUR*(B*B+RPR*RPR)                  DAUX1350
      XIXZP = AMUF*ZFD1 - BMUR*ZRD3                                     DAUX1360
      XIYZP = -XMUF*RFPF*ZFD1RF - XMUR*RPR*ZRD3R                        DAUX1370
      GAM2 = XMUF*ZFD1RF + XMUR*ZRD3R                                   DAUX1380
      GAM3 = GAM2 - RHFMUF - RHOMUR                                     DAUX1390
      GAM4 = XIYZP + RHF2MF*PHIF + RHMR2*PHIR                           DAUX1400
      GAM5 = XMUF*(A*A-RFPF*RFPF) + XMUR*(B*B-RPR*RPR)                  DAUX1410
      GAM6 = 2.0*WFMF + 2.0*TG61                                        DAUX1420
      GAM7 = 2.0*ZFD1*WFMF + 2.0*ZRD3*TG61                              DAUX1430
      GAM8 = -2.0*RFPF*WFMF - 2.0*RPR*TG61                              DAUX1440
      GAM9 = 2.0*A*WFMF - 2.0*B*TG61                                    DAUX1450
    3 CALL VPOS                                                         DAUX1460
      CALL VGORNT                                                       DAUX1470
      IF(ISUS.EQ.2) GO TO 20                                            DAUX1480
      DISP(1) = DEL1                                                    DAUX1490
      DISP(2) = DEL2                                                    DAUX1500
      VEL(1) = DEL1D                                                    DAUX1510
```

```
      VEL(2) = DEL2D                                                    DAUX1520
      GO TO 21                                                          DAUX1530
   20 DISP(1) = DEL1+TSFO2*PHIF                                         DAUX1540
      DISP(2) = DEL1-TSFO2*PHIF                                         DAUX1550
      VEL(1) = DEL1D+TSFO2*PHIFD                                        DAUX1560
      VEL(2) = DEL1D-TSFO2*PHIFD                                        DAUX1570
      GO TO 22                                                          DAUX1580
   21 IF(ISUS.NE.1) GO TO 22                                            DAUX1590
      DISP(3) = DEL3                                                    DAUX1600
      DISP(4) = DEL4                                                    DAUX1610
      VEL(3) = DEL3D                                                    DAUX1620
      VEL(4) = DEL4D                                                    DAUX1630
      GO TO 23                                                          DAUX1640
   22 DISP(3) = DEL3+TSO2*PHIR                                          DAUX1650
      DISP(4) = DEL3-TSO2*PHIR                                          DAUX1660
      VEL(3) = DEL3D+TSO2*PHIRD                                         DAUX1670
      VEL(4) = DEL3D-TSO2*PHIRD                                         DAUX1680
   23 CALL SUSFRC(DISP,VEL)                                             DAUX1690
      CALL UMOMNT(ISUS)                                                 DAUX1700
      IF(INDB.NE.0) CALL SFORCE                                         DAUX1710
      GO TO (30,31,32),IS1                                              DAUX1720
   30 CALL MATRIX                                                       DAUX1730
      GO TO 34                                                          DAUX1740
   31 CALL MTRXIR                                                       DAUX1750
      GO TO 34                                                          DAUX1760
   32 CALL MTRXSF                                                       DAUX1770
   34 CALL SIMSOL(DMATX,10,10)                                          DAUX1780
      DU = DMATX(1,11)                                                  DAUX1790
      DV = DMATX(2,11)                                                  DAUX1800
      DW = DMATX(3,11)                                                  DAUX1810
      DP = DMATX(4,11)                                                  DAUX1820
      DQ = DMATX(5,11)                                                  DAUX1830
      DR = DMATX(6,11)                                                  DAUX1840
      DXCP = AMTX(1,1)*U + AMTX(1,2)*V + AMTX(1,3)*W                    DAUX1850
      DYCP = AMTX(2,1)*U + AMTX(2,2)*V + AMTX(2,3)*W                    DAUX1860
      DZCP = AMTX(3,1)*U + AMTX(3,2)*V + AMTX(3,3)*W                    DAUX1870
      DTHTTP = Q*CPHTP - R*SPHTP                                        DAUX1880
      DPHITP = P + (Q*SPHTP + R*CPHTP)*TANTP                            DAUX1890
      DPSITP = (Q*SPHTP + R*CPHTP)*SECTP                                DAUX1900
      IF(ISUS.EQ.2) GO TO 40                                            DAUX1910
      DDEL1D = DMATX(7,11)                                              DAUX1920
      DDEL2D = DMATX(8,11)                                              DAUX1930
      DDEL1 = DEL1D                                                     DAUX1940
      DDEL2 = DEL2D                                                     DAUX1950
      GO TO 41                                                          DAUX1960
   40 DDEL1D = DMATX(7,11)                                              DAUX1970
      DPHIFD = DMATX(8,11)                                              DAUX1980
      DDEL1 = DEL1D                                                     DAUX1990
      DPHIF = PHIFD                                                     DAUX2000
      GO TO 43                                                          DAUX2010
   41 IF(ISUS.NE.1) GO TO 43                                            DAUX2020
```

```
      DDEL3D = DMATX(9,11)                                              DAUX2030
      DDEL4D = DMATX(10,11)                                             DAUX2040
      DDEL3 = DEL3D                                                     DAUX2050
      DDEL4 = DEL4D                                                     DAUX2060
      GO TO 44                                                          DAUX2070
   43 DDEL3D = DMATX(9,11)                                              DAUX2080
      DPHIRD = DMATX(10,11)                                             DAUX2090
      DDEL3 = DEL3D                                                     DAUX2100
      DPHIR = PHIRD                                                     DAUX2110
   44 CONTINUE                                                          DAUX2120
      IF(IHIT.EQ.0.AND.INDCRB.GE.0) RETURN                              DAUX2130
      DPSIFI = PSIFID                                                   DAUX2140
      T1PSI = 0.0                                                       DAUX2150
      T2PSI = 0.0                                                       DAUX2160
      IF(ABS(PSIFID).GT.EPSPS) T1PSI = SIGN(CPSP,PSIFID)                DAUX2170
      IF(SIGN(1.,PSIFID) .NE. SIGN(1.,PSIFI)) GO TO 7                   DAUX2180
      ABSPSF = ABS(PSIFI)                                               DAUX2190
      IF(ABSPSF .GT. OMGPS) T2PSI=SIGN((AKPS*(ABSPSF-OMGPS)),PSIFI)     DAUX2200
    7 DDPSFI = (FYU(1)*(HCAH1-XPS*COS(PSIIP(1))*CTXG(1))+               DAUX2210
     1           FYU(2)*(HCAH2-XPS*COS(PSIIP(2))*CTXG(2))-              DAUX2220
     2         FXU(1)*(HCBH1+PHI1*HCGH1)-FXU(2)*(HCBH2+PHI2*HCGH2)-     DAUX2230
     3          T1PSI-T2PSI+FZU(1)*HCAH1*PHI1+FZU(2)*HCAH2*PHI2)/XIPS   DAUX2240
      RETURN                                                            DAUX2250
      END                                                               DAUX2260
```

```
      SUBROUTINE DRIVER(/SA/,/SADOT/,/ISA/)                             DRIV0010
C               HVOSM-RD2 VERSION                                       DRIV0020
C.              REVISED OCTOBER 1975  CALSPAN CORPORATION               DRIV0030
      ISA = 0                                                           DRIV0040
      RETURN                                                            DRIV0050
      END                                                               DRIV0060
```

```
      SUBROUTINE DRIVID                                                 DRVD0010
C              HVOSM-RD2 VERSION                                        DRVD0020
C              REVISED OCTOBER 1975  CALSPAN CORPORATION                DRVD0030
      RETURN                                                            DRVD0040
      END                                                               DRVD0050
```

```
      SUBROUTINE EGYSUM                                                  EGYS0010
C              HVOSM-RD2 VERSION                                         EGYS0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION                EGYS0030
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),     EGYS0040
     1              SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,     EGYS0050
     2              DELTB,XINPT(100)                                     EGYS0060
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),   EGYS0070
     1             YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),      EGYS0080
     2             AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,     EGYS0090
     3             CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),        EGYS0100
     4             YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT   EGYS0110
     5             ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,   EGYS0120
     6             XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,      EGYS0130
     7             AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,EGYS0140
     8             FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,EGYS0150
     9             NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,        EGYS0160
     A             RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,           EGYS0170
     B             SWORK,SPENGY,DISS,IPLN,ILOAD                          EGYS0180
      DIMENSION INDXPT(4)                                                EGYS0190
      EQUIVALENCE (INDXPT(1),I1)                                         EGYS0200
    1 DISS = ENRGY-EEE                                                   EGYS0210
      SWORK = SWORK+FRICT*VTAN*DT                                        EGYS0220
      IF(FN.EQ.0.0) GO TO 2                                              EGYS0230
      SPENGY = SPENGY+.5*(FN+FNP)*(VDEF-PVDEF)                           EGYS0240
    2 PVDEF = VDEF                                                       EGYS0250
      FNP = FN                                                           EGYS0260
      IF(FN.EQ.0.0)DELBB = EPSL+SET*DELX                                 EGYS0270
      DELBBP = DELBB                                                     EGYS0280
      IF(ININD.EQ.1)ININD = 2                                            EGYS0290
      PPRB = RB                                                          EGYS0300
      PCAB = CAB                                                         EGYS0310
      PCBB = CBB                                                         EGYS0320
      PCGB = CGB                                                         EGYS0330
      YBPTP = YBPT                                                       EGYS0340
      PSZR = SZR                                                         EGYS0350
      RETURN                                                             EGYS0360
      END                                                                EGYS0370
```

```
      SUBROUTINE GCP(I)
C                 HVOSM-RD2 VERSION
C                 REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1           (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
    1  XLM1(I) = XP(I)*CAYW(I)+YP(I)*CBYW(I)+ZP(I)*CGYW(I)
      XLM2(I) = XP(I)*CAGZ(I)+YP(I)*CBGZ(I)+ZPG1(I)*CGGZ(I)
      XLM3(I) = D1(I)*XP(I)+D2(I)*YP(I)+D3(I)*ZP(I)
    2 CMTX(1,1) = CAYW(I)
      CMTX(1,2) = CBYW(I)
      CMTX(1,3) = CGYW(I)
      CMTX(1,4) = XLM1(I)
      CMTX(2,1) = CAGZ(I)
```

```
      CMTX(2,2) = CBGZ(I)
      CMTX(2,3) = CGGZ(I)
      CMTX(2,4) = XLM2(I)
      CMTX(3,1) = D1(I)
      CMTX(3,2) = D2(I)
      CMTX(3,3) = D3(I)
      CMTX(3,4) = XLM3(I)
      CALL SIMSOL(CMTX,3,3)
    3 XGPP(I) = CMTX(1,4)
      YGPP(I) = CMTX(2,4)
      ZGPP(I) = CMTX(3,4)
      TX = XGPP(I)-XP(I)
      TY = YGPP(I)-YP(I)
      TZ = ZGPP(I)-ZP(I)
      DELTA(I) = SQRT(TX**2+TY**2+TZ**2)
      CAR(I) = TX/DELTA(I)
      CBR(I) = TY/DELTA(I)
      CGR(I) = TZ/DELTA(I)
      HI(I) = AMIN1(DELTA(I),RW(I))
    4 FR(I) = 0.0
      IF(RW(I).EQ.HI(I)) RETURN
      TRH = RW(I)-HI(I)
      IF(TRH.GT.SIGT(I)) GO TO 5
      FR(I) = AKT(I)*TRH
      RETURN
    5 FR(I) = AKT(I)*(XLAMT(I)*(TRH-SIGT(I))+SIGT(I))
      RETURN
      END
```

```
      SUBROUTINE IDOUT
C           HVOSM-RD2 VERSION
C           REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),
     1                SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,
     2                DELTB,XINPT(100)
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,
     1              OMEGRC,AKRE,AKREP,OMEGRE,END3
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,
     1                DAPRB,DAPRE,DDAPR,NAPR
      DIMENSION APF(21),APR(21)
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))
```

```
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),       IDOT0500
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),         IDOT0510
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)              IDOT0520
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,           IDOT0530
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),IDOT0540
     2               NCAMF,NCAMR,NDTHF,NDTHR                                IDOT0550
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                              IDOT0560
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                              IDOT0570
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                        IDOT0580
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                    IDOT0590
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                    IDOT0600
     5               YCMP(6),ZCMP(6),PHICM(6)                               IDOT0610
      COMMON/BARSTR/XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),             IDOT0620
     1              ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),           IDOT0630
     2              FNSTI(3),AKST(3)                                        IDOT0640
C                                                                           IDOT0650
      DATA ZERO/0.0/                                                        IDOT0660
      DIMENSION TTARG(50) ,NTARG(10)                                        IDOT0670
      DATA TTARG/50*0.0/ , NTARG/10*0/                                      IDOT0680
      DIMENSION TXARG(21), TYARG(21)                                        IDOT0690
      DATA TXARG/21*0.0/,TYARG/21*0.0/                                      IDOT0700
      DATA CON1/4HCONS/, VARI/4HVARI/                                       IDOT0710
      DIMENSION DINCH(2),DEG(2),DIPS(2),DPS(2),PS2PI(3),PS2I(3),            IDOT0720
     1          DIPS2(3),PIPR(3),RAPRA(2),RADS(2),RPI(2),RPI2(3),           IDOT0730
     2          RPI3(3),PPI(2),PPI3(2),PSPI(3),RAPS(2)                      IDOT0740
      DATA DINCH/4HINCH,4HES  /,  DEG/4HDEGR,4HEES /                        IDOT0750
      DATA   DPS/4HDEG/,4HSEC /, DIPS/4HIN/S,4HEC  /                        IDOT0760
      DATA RAPRA/4HRAD/,4HRAD /, RADS/4HRADI,4HANS /                        IDOT0770
      DATA   RPI/4HRAD/,4HIN  /,  PPI/4HLB/I,4HN   /                        IDOT0780
      DATA  PPI3/4HLB/I,4HN**3/, RAPS/4HRAD/,4HSEC /                        IDOT0790
      DATA PS2PI/4HLB-S,4HEC**,4H2/IN/, PS2I/4HLB-S,4HEC**,4H2-IN/          IDOT0800
      DATA DIPS2/4HIN/S,4HEC**,4H2   /, PIPR/4HLB-I,4HN/RA,4HD   /          IDOT0810
      DATA  RPI2/4HRAD/,4HIN**,4H2   /, RPI3/4HRAD/,4HIN**,4H3   /          IDOT0820
      DATA  PSPI/4HLB-S,4HEC/I,4HN   /                                      IDOT0830
      DATA SEC/4HSEC /                                                      IDOT0840
      DIMENSION PDI(2)                                                      IDOT0850
      DATA PDI/4HLB-I,4HN   /,PD/4HLB  /                                    IDOT0860
      DIMENSION  TD1(2),TD2(2)                                              IDOT0870
      DATA UD2/4HDEL2/,UPF/4HPHIF/,UD4/4HDEL4/,UPR/4HPHIR/                  IDOT0880
      DATA UDE/4HO  =/,UVE/4HOD =/                                          IDOT0890
      DIMENSION TNU2(2),TNU3(3)                                             IDOT0900
      DATA TNU2/4HNOT ,4HUSED/, TNU3/4HNOT ,4HUSED,4H    /                  IDOT0910
      DIMENSION TD3(2),T3D1(3),T3D2(3)                                      IDOT0920
C                                                                           IDOT0930
   11 WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                        IDOT0940
     1      (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),             IDOT0950
     2      (GHED(I),I=1,10),(SHED(I),I=1,10)                               IDOT0960
 1000 FORMAT(1H1,9X,18A4,30X,2A4 / 5X,3(10A4) / 5X,2(10A4) / )              IDOT0970
      WRITE(6,1001) TO,SEC,T1,SEC,DTCOMP,SEC,MODE,DTPRNT,SEC                IDOT0980
 1001 FORMAT(1H0,24X,39HP R O G R A M   C O N T R O L   D A T A /           IDOT0990
     1 10X,38HSTART TIME                        TO      =,F10.4,2X,A4 /     IDOT1000
```

```
    2 10X,38HEND TIME                          T1     =,F10.4,2X,A4 /       IDOT1010
    3 10X,38HINTEGRATION INCREMENT             DTCOMP =,F10.4,2X,A4 /       IDOT1020
    4 62X,30H(0=VARIABLE STEP ADAMS-MOULTON     /                           IDOT1030
    5 10X,38HINTEGRATION MODE                  MODE   =,I5,                 IDOT1040
    6  8X,16H-)1= RUNGA-KUTTA      /                                        IDOT1050
    7 62X,28H(2= FIXED STEP ADAMS-MOULTON    /                              IDOT1060
    8 10X,38HPRINT INTERVAL                    DTPRNT =,F10.4,2X,A4 )       IDOT1070
      WRITE(6,1002) ISUS,INDCRB,DELTC,SEC                                   IDOT1080
 1002 FORMAT(1H ,                                                           IDOT1090
    1 61X,50H(0= INDEPENDENT FRONT SUSPENSION, SOLID REAR AXLE   /          IDOT1100
    2 10X,38HSUSPENSION OPTION                 ISUS   =,I5,                 IDOT1110
    3  8X,42H-)1= INDEPENDENT FRONT AND REAR SUSPENSION   /                 IDOT1120
    4 62X,42H(2= SOLID FRONT AND REAR AXLES              /                  IDOT1130
    5 62X,42H(0= NO CURB, NO STEER DEGREE OF FREEDOM     /                  IDOT1140
    6 10X,38HCURB/STEER OPTION                 INDCRB =,I5,                 IDOT1150
    7  8X,10H-)1= CURB      /                                               IDOT1160
    8 62X,42H(-1=STEER DEGREE OF FREEDOM, NO CURB        /                  IDOT1170
    9 10X,38HCURB INTEGRATION INCR.            DELTC  =,F10.5,2X,A4   )     IDOT1180
      WRITE(6,1003) INDB,DELTB,SEC                                          IDOT1190
 1003 FORMAT(1H ,61X,14H(0= NO BARRIER    /                                 IDOT1200
    1 62X,42H|1= RIGID BARRIER , FINITE VERT. DIM.       /                  IDOT1210
    2 10X,38HBARRIER OPTION                    INDB   =,I5,                 IDOT1220
    3  8X,42H-)2=  ''      ''    ,INFINITE  ''    ''     /                  IDOT1230
    4 62X,42H|3= DEFORM. ''     , FINITE  ''    ''       /                  IDOT1240
    5 62X,42H(4=    ''     ''   ,INFINITE  ''    ''      /                  IDOT1250
    6 10X,38HBARRIER INTEGRATION INCR.         DELTB  =,F10.5,2X,A4   )     IDOT1260
      IF(MODE.EQ.0) WRITE(6,1008) EBAR,EM,AAA,HMAX,HMIN,BET                 IDOT1270
 1008 FORMAT(1H0,9X,34HARGUMENTS FOR MODE 0 INTEGRATION : /                 IDOT1280
    A       8X,6(2X,F12.3)  )                                               IDOT1290
      WRITE(6,1004) XCOP,DINCH,UO,DIPS ,YCOP,DINCH,VO,DIPS,                 IDOT1300
    A               ZCOP,DINCH,WO,DIPS                                      IDOT1310
 1004 FORMAT(1H0,//,52X,38HI N I T I A L   C O N D I T I O N S   //         IDOT1320
    1 40X, 8HXCOP    =,F8.2,3X,2A4,39X,6HUO    =,F8.2,3X,2A4      /         IDOT1330
    2 10X,38HSPRUNG MASS C.G. POSITION     YCOP    =,F8.2,3X,2A4,           IDOT1340
    3  7X,38HSPRUNG MASS LINEAR VELOCITY     VO    =,F8.2,3X,2A4  /         IDOT1350
    4 40X, 8HZCOP    =,F8.2,3X,2A4,39X,6HWO    =,F8.2,3X,2A4      )         IDOT1360
      WRITE(6,1005) PHIO,DEG,PO,DPS,THETAO,DEG,QO,DPS,                      IDOT1370
    1               PSIO,DEG,RO,DPS                                         IDOT1380
 1005 FORMAT(1H ,                                                           IDOT1390
    1 39X, 8HPHIO    =,F8.2,3X,2A4,39X,6HPO    =,F8.2,3X,2A4      /         IDOT1400
    2 10X,38HSPRUNG MASS ORIENTATION       THETAO =,F8.2,3X,2A4 ,           IDOT1410
    3  7X,38HSPRUNG MASS ANGULAR VELOCITY    QO    =,F8.2,3X,2A4 /          IDOT1420
    4 40X, 8HPSIO    = F8.2,3X,2A4,39X,6HRO    =,F8.2,3X,2A4      )         IDOT1430
      IF(ISUS.EQ.2) GO TO 101                                               IDOT1440
      UMP1 = UD2                                                            IDOT1450
      TD1(1) = DINCH(1)                                                     IDOT1460
      TD1(2) = DINCH(2)                                                     IDOT1470
      TD2(1) = DIPS(1)                                                      IDOT1480
      TD2(2) = DIPS(2)                                                      IDOT1490
      UMP = DEL20                                                           IDOT1500
      UMV = DEL20D                                                          IDOT1510
```

```
      GO TO 102
  101 UMP1 = UPF
      TD1(1) = DEG(1)
      TD1(2) = DEG(2)
      TD2(1) = DPS(1)
      TD2(2) = DPS(2)
      UMP = PHIFO
      UMV = PHIFOD
  102 WRITE(6,1006) DEL10,DINCH,DEL10D,DIPS,UMP1,UDE,UMP,TD1,UMP1,UVE,
     1              UMV,TD2
 1006 FORMAT(1H0,39X,8HDEL10   =,F8.2,3X,2A4,37X,8HDEL10D =,F8.2,3X,2A4/
     1 10X,30HUNSPRUNG MASS POSITIONS         ,2A4,F8.2,3X,2A4,
     2  7X,30HUNSPRUNG MASS VELOCITIES        ,2A4,F8.2,3X,2A4     )
      IF(ISUS.EQ.1) GO TO 103
      UMP1 = UPR
      TD1(1) = DEG(1)
      TD1(2) = DEG(2)
      TD2(1) = DPS(1)
      TD2(2) = DPS(2)
      UMP = PHIRO
      UMV = PHIROD
      GO TO 104
  103 UMP1 = UD4
      TD1(1) = DINCH(1)
      TD1(2) = DINCH(2)
      TD2(1) = DIPS(1)
      TD2(2) = DIPS(2)
      UMP = DEL40
      UMV = DEL40D
  104 WRITE(6,1007) DEL30,DINCH,DEL30D,DIPS,UMP1,UDE,UMP,TD1,UMP1,UVE,
     1              UMV,TD2,PSIFIO,DEG,PSIFDO,DPS
 1007 FORMAT(1H ,39X,8HDEL30   =,F8.2,3X,2A4,37X,8HDEL30D =,F8.2,3X,2A4/
     1 40X,2A4,F8.2,3X,2A4,37X,2A4,F8.2,3X,2A4  /
     2 10X,38HSTEER ANGLE                       PSIFIO =,F8.2,3X,2A4,
     3  7X,38HSTEER VELOCITY                    PSIFDO =,F8.2,3X,2A4     )
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)
      WRITE(6,2001) XMS,PS2PI,     A,DINCH,
     1              XMUF,PS2PI,    B,DINCH,
     2              XMUR,PS2PI,    ZF,DINCH
 2001 FORMAT(1H0,
     1  9X,37HSPRUNG MASS                       XMS   =,F10.3,1X,3A4,
     2  5X,32HFRONT WHEEL X LOCATION    A     =,     F10.3,1X,2A4  /
     3 10X,37HFRONT UNSPRUNG MASS               XMUF  =,F10.3,1X,3A4,
     4  5X,32HREAR WHEEL X LOCATION     B     =,     F10.3,1X,2A4  /
     5 10X,37HREAR UNSPRUNG MASS                XMUR  =,F10.3,1X,3A4,
     6  5X,32HFRONT WHEEL Z LOCATION    ZF    =,     F10.3,1X,2A4  )
      TD1(1) = TNU2(1)
      TD1(2) = TNU2(2)
      IF(ISUS.EQ.2) GO TO 201
```

```
      GO TO 202
  201 TD1(1) = DINCH(1)
      TD1(2) = DINCH(2)
  202 CONTINUE
      WRITE(6,2002) XIX, PS2I,      ZR  ,DINCH,
     1              XIY, PS2I,      TF  ,DINCH,
     2              XIZ, PS2I,      TR  ,DINCH,
     3              XIXZ,PS2I,     RHOF,TD1
 2002 FORMAT(1H ,
     1  9X,37HX MOMENT OF INERTIA             XIX   =,F10.3,1X,3A4 ,
     2  5X,32HREAR WHEEL Z LOCATION      ZR   =,     F10.3,1X,2A4 /
     3 10X,37HY MOMENT OF INERTIA             XIY   =,F10.3,1X,3A4 ,
     4  5X,32HFRONT WHEEL TRACK          TF   =,     F10.3,1X,2A4 /
     5 10X,37HZ MOMENT OF INERTIA             XIZ   =,F10.3,1X,3A4 ,
     6  5X,32HREAR WHEEL TRACK           TR   =,     F10.3,1X,2A4 /
     7 10X,37HXZ PRODUCT OF INERTIA           XIXZ  =,F10.3,1X,3A4 ,
     8  5X,32HFRONT ROLL AXIS            RHOF =,     F10.3,1X,2A4 )
      DO 203 K=1,3
      T3D1(K) = TNU3(K)
  203 T3D2(K) = TNU3(K)
      DO 204 K=1,2
      TD1(K) = TNU2(K)
      TD2(K) = TNU2(K)
  204 TD3(K) = TNU2(K)
      IF(ISUS.EQ.1) GO TO 206
      DO 205 K=1,2
      T3D2(K) = PS2I(K)
      TD1(K) = DINCH(K)
  205 TD3(K) = DINCH(K)
      T3D2(3) = PS2I(3)
  206 IF(ISUS.NE.2) GO TO 208
      DO 207 K=1,2
      T3D1(K) = PS2I(K)
  207 TD2(K) = DINCH(K)
      T3D1(3) = PS2I(3)
  208 WRITE(6,2003) XIF, T3D1, RHO, TD1,
     1              XIR, T3D2, TSF, TD2,
     2              G   ,DIPS2,  TS,TD3
 2003 FORMAT(1H ,
     1  9X,37HFRONT AXLE MOMENT OF INERTIA  XIF   =,F10.3,1X,3A4 ,
     2  5X,32HREAR ROLL AXIS             RHO  =,     F10.3,1X,2A4 /
     3 10X,37HREAR AXLE MOMENT OF INERTIA   XIR   =,F10.3,1X,3A4 ,
     4  5X,32HFRONT SPRING TRACK         TSF  =,     F10.3,1X,2A4 /
     5 10X,37HGRAVITY                       G     =,F10.3,1X,3A4 ,
     6  5X,32HREAR SPRING TRACK          TS   =,     F10.3,1X,2A4 )
      DO 209 K=1,3
      T3D1(K) = TNU3(K)
      T3D2(K) = TNU3(K)
      IF(K.EQ.3) GO TO 209
      TD1(K) = TNU2(K)
      TD2(K) = TNU2(K)
```

```
      TD3(K) = TNU2(K)                                                  IDOT2540
  209 CONTINUE                                                          IDOT2550
      IF(ISUS.EQ.1) GO TO 211                                           IDOT2560
      TD1(1) = RAPRA(1)                                                 IDOT2570
      TD1(2) = RAPRA(2)                                                 IDOT2580
      GO TO 213                                                         IDOT2590
  211 DO 212 K=1,3                                                      IDOT2600
      T3D1(K) = RPI2(K)                                                 IDOT2610
      T3D2(K) = RPI3(K)                                                 IDOT2620
      IF(K.EQ.3) GO TO 212                                              IDOT2630
      TD2(K) = RADS(K)                                                  IDOT2640
      TD3(K) = RPI(K)                                                   IDOT2650
  212 CONTINUE                                                          IDOT2660
  213 WRITE(6,2004) X1,       DINCH,    RF,       PIPR,                 IDOT2670
     5              Y1,       DINCH,    RR,       PIPR,                 IDOT2680
     2              Z1,       DINCH,    AKRS,     TD1,                  IDOT2690
     3              X2,       DINCH,    AKDS,     TD2,                  IDOT2700
     4              Y2,       DINCH,    AKDS1,    TD3,                  IDOT2710
     5              Z2,       DINCH,    AKDS2,    T3D1,                 IDOT2720
     6                                  AKDS3,    T3D2                  IDOT2730
 2004 FORMAT(1H0,39X,7HX1    = ,F10.2,1X,2A4  ,                         IDOT2740
     1  9X,32HFRONT AUX ROLL STIFFNESS  RF    =,F10.2,1X,3A4 /          IDOT2750
     2 10X,37HACCELEROMETER 1 POSITION      Y1    =,F10.2,1X,2A4 ,      IDOT2760
     3  9X,32HREAR AUX ROLL STIFFNESS   RR    =,    F10.2,1X,3A4 /      IDOT2770
     4 40X,7HZ1    =,F10.2,1X,2A4 ,                                     IDOT2780
     5  9X,32HREAR ROLL-STEER COEF.     AKRS =,     F10.4,1X,2A4 /      IDOT2790
     6 40X,7HX2    =,F10.2,1X,2A4 ,35X,6HAKDS =,F10.3,1X,2A4      /     IDOT2800
     7 10X,37HACCELEROMETER 2 POSITION      Y2    =,F10.2,1X,2A4 ,      IDOT2810
     8  9X,32HREAR DEFL-STEER COEFS.    AKDS1=,     F10.3,1X,2A4 /      IDOT2820
     9 40X,7HZ2    =,F10.2,1X,2A4,35X,6HAKDS2=,F10.3,1X,3A4       /     IDOT2830
     A101X,6HAKDS3=,F10.3,1X,3A4  )                                     IDOT2840
      WRITE(6,2005) XIPS,PS2I,CPSP,PDI,EPSPS,RAPS,AKPS,PIPR,            IDOT2850
     1              OMGPS,RADS,XPS,DINCH                                IDOT2860
 2005 FORMAT(1H0,15X,29HS T E E R I N G   S Y S T E M      /            IDOT2870
     1 10X,31HMOMENT OF INERTIA       XIPS  =,F10.3,1X,3A4 /            IDOT2880
     2 10X,31HCOULOMB FRICTION TORQUE CPSP  =,F10.3,1X,2A4 /            IDOT2890
     3 10X,31HFRICTION LAG            EPSP  =,F10.3,1X,2A4 /            IDOT2900
     4 10X,31HANGULAR STOP RATE       AKPS  =,F10.3,1X,3A4 /            IDOT2910
     5 10X,31HANGULAR STOP POSITION   OMGPS =,F10.3,1X,2A4 /            IDOT2920
     6 10X,31HPNEUMATIC TRAIL         XPS   =,F10.3,1X,2A4 )            IDOT2930
      WRITE(6,2006) AKF,      PPI,      AKR,      PPI,                  IDOT2940
     1              AKFC,     PPI,      AKRC,     PPI,                  IDOT2950
     2              AKFCP,    PPI3,     AKRCP,    PPI3                  IDOT2960
 2006 FORMAT(1H0,36X,16HFRONT SUSPENSION,20X,15HREAR SUSPENSION  //     IDOT2970
     1 10X,41HSUSPENSION RATE                        AKF    =,F10.3,1X,2A4,  IDOT2980
     2  9X,8HAKR     =,F10.3,1X,2A4  /                                  IDOT2990
     3 10X,41HCOMPRESSION STOP COEFS.                AKFC   =,F10.3,1X,2A4,  IDOT3000
     4  9X,8HAKRC    =,F10.3,1X,2A4  /                                  IDOT3010
     5 43X,8HAKFCP   =,F10.3,1X,2A4,9X,8HAKRCP   =,F10.3,1X,2A4   )     IDOT3020
      WRITE(6,2007) AKFE,     PPI,      AKRE,     PPI,                  IDOT3030
     1              AKFEP,    PPI3,     AKREP,    PPI3,                 IDOT3040
```

```
   2                OMEGFC,   DINCH,    OMEGRC,   DINCH,
   3                OMEGFE,   DINCH,    OMEGRE,   DINCH
2007 FORMAT(1H ,
   1  9X,41HEXTENSION STOP COEFS.              AKFE   =,F10.3,1X,2A4,
   2  9X, 8HAKRE   =,F10.3,1X,2A4  /
   3 43X, 8HAKFEP  =,F10.3,1X,2A4,9X,8HAKREP  =,F10.3,1X,2A4  /
   4 10X,41HCOMPRESSION STOP LOCATION          OMEGFC =,F10.3,1X,2A4,
   5  9X, 8HOMEGRC =,F10.3,1X,2A4  /
   6 10X,41HEXTENSION STOP LOCATION            OMEGFE =,F10.3,1X,2A4,
   7  9X, 8HOMEGRE =,F10.3,1X,2A4  )
     WRITE(6,2008) XLAMF,                XLAMR,
   1                CF,       PSPI,     CR,       PSPI,
   2                CFP,      PD,       CRP,      PD,
   3                EPSF,     DIPS,     EPSR,     DIPS
2008 FORMAT(1H ,
   1  9X,41HSTOP ENERGY DISSIPATION FACTOR     XLAMF  =,F10.3,
   2 18X, 8HXLAMR  =,F10.3  /
   3 10X,41HVISCOUS DAMPING COEF.              CF     =,F10.3,1X,3A4,
   4  5X, 8HCR     =,F10.3,1X,3A4  /
   5 10X,41HCOULOMB FRICTION                   CFP    =,F10.3,1X,1A4,
   6 13X, 8HCRP    =,F10.3,1X,1A4  /
   7 10X,41HFRICTION LAG                       EPSF   =,F10.3,1X,2A4,
   8  9X, 8HEPSR   =,F10.3,1X,2A4  )
     IF(ISUS.EQ.2.AND.TINCR.EQ.0.0) GO TO 304
     WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
   1      (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
   2      (GHED(I),I=1,10),(SHED(I),I=1,10)
     IF(ISUS.EQ.2) GO TO 301
     DO 306 K=1,2
     TD1(K) = DINCH(K)
 306 TD2(K) = DEG(K)
     IF(ISUS.EQ.1) GO TO 308
     DO 307 K=1,2
     TD1(K) = TNU2(K)
 307 TD2(K) = TNU2(K)
 308 WRITE(6,3001) DINCH,DEG,TD1,TD2,DINCH,DINCH,TD1,TD1
3001 FORMAT(1H0,
   A 10X,18HFRONT WHEEL CAMBER, 8X,17HREAR WHEEL CAMBER,
   B  6X,23HFRONT HALF-TRACK CHANGE, 4X,22HREAR HALF-TRACK CHANGE  /
   C 18X,2HVS,24X,2HVS,24X,2HVS,24X,2HVS  /
   D  9X,21HSUSPENSION DEFLECTION, 5X,21HSUSPENSION DEFLECTION,
   E  5X,21HSUSPENSION DEFLECTION, 5X,21HSUSPENSION DEFLECTION    //
   F 12X,15HDELTAF      PHIC,11X,16HDELTAR      PHIRC ,
   G 10X,15HDELTAF      DTHF,11X,15HDELTAR      DTHR  /
   H 12X,2A4,2X,2A4,8X,2A4,2X,2A4,8X,2A4,2X,2A4,8X,2A4,2X,2A4      )
     Y = DELB
     DO 302 I=1,NDEL
     TTARG(I) = Y
     Y = Y+DDEL
 302 CONTINUE
     WRITE(6,3002) (TTARG(I),PHIC(I),TTARG(I),PHIRC(I),
```

```
     1                TTARG(I),DTHF(I),TTARG(I),DTHR(I),I=1,NDEL)        IDOT3560
 3002 FORMAT(1H0,4(8X,F8.2,2X,F8.2)/(1X,4(8X,F8.2,2X,F8.2))      )       IDOT3570
  301 CONTINUE                                                           IDOT3580
      NTPR = 0                                                           IDOT3590
      TTARG(1) = 0.0                                                     IDOT3600
      IF(TINCR.EQ.0.0) GO TO 304                                         IDOT3610
      NTPR = (TE-TB)/TINCR + 1.5                                         IDOT3620
      Y = TB                                                             IDOT3630
      DO 305 I=1,NTPR                                                    IDOT3640
      TTARG(I) = Y                                                       IDOT3650
      Y = Y+TINCR                                                        IDOT3660
  305 CONTINUE                                                           IDOT3670
      WRITE(6,3003)                                                      IDOT3680
 3003 FORMAT(1H0,//56X,21HDRIVER CONTROL TABLES  //                      IDOT3690
     1 4(32H      T      PSIF      TQF      TQR )      /                 IDOT3700
     2 4(32H     SEC      DEG     LB-FT    LB-FT)      /)                IDOT3710
C          NTPR4 IS NUMBER OF LINES FOR TABLES IN FOUR GROUPS PER LINE   IDOT3720
      NNADD = 0                                                          IDOT3730
      IF((MOD(NTPR,4)).NE.0) NNADD=1                                     IDOT3740
      NTPR4 = NTPR/4 + NNADD                                             IDOT3750
      NTPR43 = 3*NTPR4                                                   IDOT3760
      DO 303 J=1,NTPR4                                                   IDOT3770
      I1 = J                                                             IDOT3780
      I4 = MIN0(NTPR ,I1+NTPR43)                                         IDOT3790
      WRITE(6,3004)((TTARG(II),PSIF(II),TQF(II),TQR(II)),II=I1,I4,NTPR4)IDOT3800
 3004 FORMAT(1X,4(F8.3,F8.3,F8.1,F8.1)  )                                IDOT3810
  303 CONTINUE                                                           IDOT3820
  304 CONTINUE                                                           IDOT3830
C                                                                        IDOT3840
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                     IDOT3850
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),           IDOT3860
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)                             IDOT3870
      WRITE(6,4001)                                                      IDOT3880
 4001 FORMAT(1H0,60X,17HT I R E   D A T A   /                            IDOT3890
     A 54X,2HRF,10X,2HLF,10X,2HRR,10X,2HLR   )                           IDOT3900
      WRITE(6,4002) AKT,PPI,SIGT,DINCH,XLAMT,A0,A1,A2,A3,A4,OMEGT,       IDOT3910
     A               RW,DINCH,AMU                                        IDOT3920
 4002 FORMAT(1H0,                                                        IDOT3930
     A  9X,39HTIRE LINEAR SPRING RATE          AKT    =,4(F10.3,2X),2A4 / IDOT3940
     B 10X,39HDEFL. FOR INCREASED RATE         SIGT   =,4(F10.3,2X),2A4 / IDOT3950
     C 10X,39HSPRING RATE INCREASING FACTOR    XLAMT  =,4(F10.3,2X)     / IDOT3960
     D 41X, 8HA0      =,4(F10.3,2X)  /                                   IDOT3970
     E 41X, 8HA1      =,4(F10.3,2X)  /                                   IDOT3980
     F 10X,39HSIDE FORCE COEFFICIENTS          A2     =,4(F10.3,2X)     / IDOT3990
     G 41X, 8HA3      =,4(F10.3,2X)  /                                   IDOT4000
     H 41X, 8HA4      =,4(F10.3,2X)  /                                   IDOT4010
     I 10X,39HTIRE OVERLOAD FACTOR             OMEGT  =,4(F10.3,2X)     / IDOT4020
     J 10X,39HTIRE UNDEFLECTED RADIUS          RW     =,4(F10.3,2X),2A4 / IDOT4030
     K 10X,39HTIRE / GROUND FRICTION COEF.     AMU    =,4(F10.3,2X)     ) IDOT4040
      IF( IAPFR(1) .EQ.0 .AND. IAPFR(2) .EQ.0) GO TO 400                 IDOT4050
      WRITE(6,4004)                                                      IDOT4060
```

```
4004 FORMAT(1H0,8X,48HANTI-PITCH TABLES FOR CIRCUMFERENTIAL TIRE FORCE IDOT4070
   1     // 9X,11HFRONT WHEEL,5X,3HAPF,5X,10HREAR WHEEL,5X,3HAPR /      IDOT4080
   2     9X,11HDEFL. - IN.,3X,8HLB/LB-FT,5X,10HDEFL.- IN.,              IDOT4090
   3     3X,8HLB/LB-FT  / )                                             IDOT4100
     FDEF = DAPFB                                                       IDOT4110
     RDEF = DAPRB                                                       IDOT4120
     MAP = NAPF                                                         IDOT4130
     IF(NAPF.NE.NAPR) MAP = MINO(NAPF,NAPR)                             IDOT4140
     IF(NAPF.EQ.0) GO TO 402                                            IDOT4150
     IF(NAPR.EQ.0) GO TO 406                                            IDOT4160
     DO 401 I=1,MAP                                                     IDOT4170
     WRITE(6,4005) FDEF,APF(I),RDEF,APR(I)                              IDOT4180
4005 FORMAT(5X,4(5X,F8.4))                                              IDOT4190
     FDEF = FDEF+DDAPF                                                  IDOT4200
 401 RDEF = RDEF+DDAPR                                                  IDOT4210
     IF(NAPF.EQ.NAPR) GO TO 404                                         IDOT4220
     IF(NAPR.GT.NAPF) GO TO 402                                         IDOT4230
 406 MAP1 = MAP+1                                                       IDOT4240
     DO 403 I=MAP1,NAPF                                                 IDOT4250
     WRITE(6,4006) FDEF,APF(I)                                          IDOT4260
4006 FORMAT(5X,2(5X,F8.4))                                              IDOT4270
 403 FDEF = FDEF+DDAPF                                                  IDOT4280
     GO TO 404                                                          IDOT4290
 402 MAP1 = MAP+1                                                       IDOT4300
     DO 405 I=MAP1,NAPR                                                 IDOT4310
     WRITE(6,4007) RDEF,APR(I)                                          IDOT4320
4007 FORMAT(31X,2(5X,F8.4))                                             IDOT4330
 405 RDEF = RDEF+DDAPR                                                  IDOT4340
     GO TO 404                                                          IDOT4350
 400 WRITE(6,4008)                                                      IDOT4360
4008 FORMAT(21HONO ANTI-PITCH TABLES)                                   IDOT4370
 404 CONTINUE                                                           IDOT4380
     IF(INDCRB.NE.1) GO TO 702                                          IDOT4390
     WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                     IDOT4400
   1   (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),              IDOT4410
   2    (GHED(I),I=1,10),(SHED(I),I=1,10)                               IDOT4420
     WRITE(6,6010)                                                      IDOT4430
6010 FORMAT(1H0,22X,17HC U R B   D A T A    //                          IDOT4440
   A 10X,54HCURB SLOPE CHANGE     ELEVATION AT      CURB FACE ANGLE  /  IDOT4450
   B 10X,34H LATERAL POSITION     SLOPE CHANGE   /                      IDOT4460
   C 18X,6HINCHES,11X,6HINCHES,11X,7HDEGREES      //  )                 IDOT4470
     WRITE(6,6011) YC1P,                   PHIC1,                       IDOT4480
   A               YC2P,      ZC2P,        PHIC2,                       IDOT4490
   B               YC3P,      ZC3P,        PHIC3,                       IDOT4500
   C               YC4P,      ZC4P,        PHIC4,                       IDOT4510
   D               YC5P,      ZC5P,        PHIC5,                       IDOT4520
   E               YC6P,      ZC6P,        PHIC6,                       IDOT4530
   F               NCRBSL,    AMUC                                      IDOT4540
6011 FORMAT(1H ,                                                        IDOT4550
   A 11X,6HYC1P =,F9.2,23X,7HPHIC1 =,F9.2,  /                           IDOT4560
   B 12X,6HYC2P =,F9.2,3X,6HZC2P =,F9.2,5X,7HPHIC2 =,F9.2,    /         IDOT4570
```

```
     C 12X,6HYC3P =,F9.2,3X,6HZC3P =,F9.2,5X,7HPHIC3 =,F9.2,     /          IDOT4580
     D 12X,6HYC4P =,F9.2,3X,6HZC4P =,F9.2,5X,7HPHIC4 =,F9.2,     /          IDOT4590
     E 12X,6HYC5P =,F9.2,3X,6HZC5P =,F9.2,5X,7HPHIC5 =,F9.2,     /          IDOT4600
     F 12X,6HYC6P =,F9.2,3X,6HZC6P =,F9.2,5X,7HPHIC6 =,F9.2,     /          IDOT4610
     G 12X,8HNCRBSL =,I4   /                                                IDOT4620
     F 10X,43HCURB FRICTION COEFFICIENT FACTOR    AMUC  =,F8.3         )   IDOT4630
      WRITE(6,7001) RWHJB,RWHJE,DRWHJ                                       IDOT4640
 7001 FORMAT(37HOWHEEL RADIUS-RADIAL SPRING FOR TABLE  /17H RWHJB(BEGIN)IDOT4650
     1   =,F8.3,7H INCHES / 17H RWHJE(END)      =,F8.3,5H   '' /,           IDOT4660
     2 17H DRWHJ(INCRE.)  =,F8.3,5H   '' )                                  IDOT4670
      NFJP = 0                                                              IDOT4680
      IF(DRWHJ.EQ.0.0) GO TO 702                                            IDOT4690
      NFJP = (RWHJE-RWHJB)/DRWHJ + 1.2                                      IDOT4700
      IF(NFJP.LE.0) GO TO 702                                               IDOT4710
      Y = RWHJB                                                             IDOT4720
      DO 701 I=1,NFJP                                                       IDOT4730
      TTARG(I) = Y                                                          IDOT4740
      Y = Y + DRWHJ                                                         IDCT4750
  701 CONTINUE                                                              IDOT4760
      WRITE(6,7002)                                                         IDOT4770
 7002 FORMAT(//1H ,3X,5HRW-HJ,6X,4HFJP.,6X,4HFJP.,6X,4HFJP.,6X,4HFJP. / IDOT4780
     A        5X,3HIN.,7X,4HLBS.,6X,4HLBS.,6X,4HLBS.,6X,4HLBS.    /         IDOT4790
     B        16X,2HRF,8X,2HLF,8X,2HRR,8X,2HLR    /  )                      IDOT4800
      DO 703 J=1,NFJP                                                       IDOT4810
      WRITE(6,7003) TTARG(J),(FJP(J,II),II=1,4)                             IDOT4820
 7003 FORMAT(1H ,G9.3,4G10.3)                                               IDOT4830
  703 CONTINUE                                                              IDOT4840
  702 CONTINUE                                                              IDOT4850
      IF(INDE.EQ.0) GO TO 501                                               IDOT4860
 5001 FORMAT(1HO,36X,31HSPRUNG MASS-BARRIER IMPACT DATA  //                 IDOT4870
     A      6X,18HBARRIER DIMENSIONS ,56X,24HBARRIER LOAD DEFL. COEF.  )IDOT4880
      WRITE(6,5002) YBPO,AKV,SIGR(1),DELYBP,SET,SIGR(2),                    IDOT4890
     A              ZBTP,CONS,SIGR(3),ZBBP,AMUB,SIGR(4),                    IDOT4900
     B              EPSV,SIGR(5)                                            IDOT4910
 5002 FORMAT(1HO,3X,9H(YB')0 = ,F10.3,7H INCHES,6X,12HKV         = ,F10.3IDOT4920
     A       ,13H LB/IN**3     ,9X,11HSIGMAR 0 = ,F10.4   /                 IDOT4930
     B        4X,9HDELYB' = ,F10.3,7H  ''    ,6X,12HSET         = ,F10.3 ,  IDOT4940
     C        13H DEFL. RATIO ,9X,11HSIGMAR 1 = ,F10.4   /                  IDOT4950
     D        4X,9HZBT'   = ,F10.3,7H  ''    ,6X,12HCONS        = ,F10.3,   IDOT4960
     E        13H ENERGY RATIO,9X,11HSIGMAR 2 = ,F10.4   /                  IDOT4970
     F        4X,9HZBB'   = ,F10.3,7H  ''    ,6X,12HMUB         = ,F10.3,   IDOT4980
     G        13H              ,9X,11HSIGMAR 3 = ,F10.4   /                 IDOT4990
     H        4X,18HVEHICLE DIMENSIONS,12X,12HEPSILON V = ,F10.3,           IDOT5000
     I        13H IN/SEC       ,9X,11HSIGMAR 4 = ,F10.4   )                 IDOT5010
      WRITE(6,5003) XVF,EPSB,SIGR(6),XVR,       SIGR(7),                    IDOT5020
     1              YV,SIGR(8),ZVT,SIGR(9),ZVB,SIGR(10),SIGR(11)            IDOT5030
 5003 FORMAT(1H ,3X,9HXVF     = ,F10.3,7H INCHES,6X,12HEPSILON B = ,F10.3IDOT5040
     A       ,3H LB,18X,11HSIGMAR 5 = ,F10.4   /                            IDOT5050
     B        4X,9HXVR    = ,F10.3,7H  ''    ,  47X,                        IDOT5060
     C                                3X,11HSIGMAR 6 = ,F10.4   /           IDOT5070
     D        4X,9HYV     = ,F10.3,7H  ''    ,50X,11HSIGMAR 7 = ,F10.4  / IDOT5080
```

```
     E       4X,9HZVT     = ,F10.3,7H  ''    ,50X,11HSIGMAR 8 = ,F10.4  / IDOT5090
     F       4X,9HZVB     = ,F10.3,7H  ''    ,50X,11HSIGMAR 9 = ,F10.4  / IDOT5100
     G       80X,11HSIGMAR10 = ,F10.4   )                                 IDOT5110
      WRITE(6,5004) (I,XSTIO(I),YSTIO(I),ZSTIO(I),AKST(I),I=1,3)          IDOT5120
 5004 FORMAT(1H0,18X,27HSPRUNG MASS HARD POINT DATA    //                 IDOT5130
     A 19X,37HLOCATION IN VEH. COORDS.    STIFFNESS    /                  IDOT5140
     B  9X,45HPOINT     XSTIO     YSTIO     ZSTIO      AKST  /            IDOT5150
     C  9X,44H NO.        IN.        IN.        IN.     LB/IN  //         IDOT5160
     D  (9X,I2,4X,F8.2,2X,F8.2,2X,F8.2,2X,F8.2  )  )                      IDOT5170
  501 CONTINUE                                                            IDOT5180
      IF(NZTAB.EQ.0) GO TO 700                                            IDOT5190
      DO 601 I=1,50                                                       IDOT5200
  601 TTARG(I) = 0.0                                                      IDOT5210
      DO 602 I=1,10                                                       IDOT5220
  602 NTARG(I) = 0                                                        IDOT5230
      DO 603 I=1,NZTAB                                                    IDOT5240
      TTARG(I) =       XB(I)                                              IDOT5250
      TTARG(5 + I)= XE(I)                                                 IDOT5260
      TTARG(10 + I) = XINCR(I)                                            IDOT5270
      TTARG(15 + I) = YB(I)                                               IDOT5280
      TTARG(20 + I) = YE(I)                                               IDOT5290
      TTARG(25 + I) = YINCR(I)                                            IDOT5300
      TTARG(30 + I) = AMUG(I)                                             IDOT5310
      NTARG(I)      = NBX(I)                                              IDOT5320
      NTARG(5 + I) = NBY(I)                                               IDOT5330
  603 CONTINUE                                                            IDOT5340
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                      IDOT5350
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),            IDOT5360
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)                              IDOT5370
      WRITE(6,6001)                                                       IDOT5380
 6001 FORMAT(//1H ,26X,25HTERRAIN TABLE ARGUMENTS        )                IDOT5390
      WRITE(6,6002)  (TTARG(I),I=1,5),                                    IDOT5400
     1               (TTARG(I),I=6,10),                                   IDOT5410
     2               (TTARG(I),I=11,14),ZERO,                             IDOT5420
     3               (TTARG(I),I=16,20),                                  IDOT5430
     4               (TTARG(I),I=21,25),                                  IDOT5440
     5               (TTARG(I),I=26,29),ZERO,                             IDOT5450
     6               (NTARG(I),I=1,5),                                    IDOT5460
     7               (NTARG(I),I=6,10),                                   IDOT5470
     8               (TTARG(I),I=31,35),                                  IDOT5480
     9                NZTAB                                               IDOT5490
 6002 FORMAT(1H0,25X,11H XB(BEGIN)=,5F12.3,7H INCHES   /                  IDOT5500
     A          26X,11H XE(END)  =,5F12.3,5H   ''     /                   IDOT5510
     B          26X,11H X(INCR)  =,5F12.3,5H   ''     /                   IDOT5520
     C          26X,11H YB(BEGIN)=,5F12.3,5H   ''     /                   IDOT5530
     D          26X,11H YE(END)  =,5F12.3,5H   ''     /                   IDOT5540
     E          26X,11H Y(INCR)  =,5F12.3,5H   ''     /                   IDOT5550
     F         25X,12HNO.X BOUNDS=,I8,4I12            /                   IDOT5560
     G         25X,12HNO.Y BOUNDS=,I8,4I12            /                   IDOT5570
     H          26X,11H AMUG     =,5F12.3             /                   IDOT5580
     I          25X,18HNO.TERRAIN TABLES=,I4          )                   IDOT5590
```

```
      IF(NZ5.EQ.0) GO TO 600                                            IDOT5600
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                    IDOT5610
     1      (VHED(I),I=1,10),(THED(1),I=1,10),(CHED(1),I=1,10),         IDOT5620
     2      (GHED(I),I=1,10),(SHED(I),I=1,10)                           IDOT5630
      NX5 = NX(NZTAB)                                                   IDOT5640
      NY5 = NY(NZTAB)                                                   IDOT5650
      WRITE(6,6004) NX5, (XXZGP5(I),I=1,NX5)                            IDOT5660
 6004 FORMAT(66H0 ARGUMENTS FOR TERRAIN TABLE WITH VARYING INCREMENTS (LIDOT5670
     1AST TABLE) /10H NO.OF X =, I3,2X,9H, X(ZGP)=, 12F9.3/24X,9F9.3)   IDOT5680
      WRITE(6,6003) NY5, (YYZGP5(I),I=1,NY5)                            IDOT5690
 6003 FORMAT(10H0NO.OF Y =, I3,2X,9H, Y(ZGP)=,12F9.3/24X, 9F9.3)        IDOT5700
C                                                                       IDOT5710
  600 IF(NZTAB) 604,700,604                                             IDOT5720
  604 WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                    IDOT5730
     1      (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),         IDOT5740
     2      (GHED(I),I=1,10),(SHED(I),I=1,10)                           IDOT5750
      LINES =3                                                          IDOT5760
      DO 614 I=1,NZTAB                                                  IDOT5770
      NNBX = NBX(I)                                                     IDOT5780
      NNBY = NBY(I)                                                     IDOT5790
      NNX = NX(I)                                                       IDOT5800
      NNY = NY(I)                                                       IDOT5810
      LINES = LINES + 9 + (NNY+1)*(NNX/7 + 2)                           IDOT5820
      IF(I.EQ.1) GO TO 606                                              IDOT5830
      IF(LINES .LT.55) GO TO 606                                        IDOT5840
      WRITE(6,1000) (HED(N),N=1,18),DADE(1),DADE(2),                    IDOT5850
     1      (VHED(N),N=1,10),(THED(N),N=1,10),(CHED(N),N=1,10),         IDOT5860
     2      (GHED(N),N=1,10),(SHED(N),N=1,10)                           IDOT5870
      LINES =3                                                          IDOT5880
  606 WRITE(6,6005) I,AMUG(I),(XBDRY(J,I),J=1,NNBX)                     IDOT5890
 6005 FORMAT(19H0 TERRAIN TABLE NO. ,I3,  20X, 6H AMUG=, F13.5//        IDOT5900
     X 1X,16H  X  BOUNDARIES=,4F13.5)                                   IDOT5910
      WRITE(6,6006) (PSBDRO(J,I),J=1,NNBX)                              IDOT5920
 6006 FORMAT(1X,16H PSI BOUNDARIES=,4F13.5)                             IDOT5930
      WRITE(6,6007) (YBDRY(J,I),J=1,NNBY)                               IDOT5940
 6007 FORMAT(1X,16H  Y  BOUNDARIES=,2F13.5)                             IDOT5950
      IF( I.EQ.NZTAB .AND. NZ5.NE.0) GO TO 607                          IDOT5960
      ANAME = CON1                                                      IDOT5970
      Y= XB(I)                                                          IDOT5980
      YYY = XINCR(I)                                                    IDOT5990
      DO 605 J=1,NNX                                                    IDOT6000
      TXARG(J) = Y                                                      IDOT6010
      Y = Y + YYY                                                       IDOT6020
  605 CONTINUE                                                          IDOT6030
      Y = YB(I)                                                         IDOT6040
      YYY = YINCR(I)                                                    IDOT6050
      DO 609 J=1,NNY                                                    IDOT6060
      TYARG(J) = Y                                                      IDOT6070
      Y = Y + YYY                                                       IDOT6080
  609 CONTINUE                                                          IDOT6090
      GO TO 610                                                         IDOT6100
```

```
  607 ANAME = VAR1
      DO 611 J=1,NNX
  611 TXARG(J) = XXZGP5(J)
      DO 612 J=1,NNY
  612 TYARG(J) = YYZGP5(J)
  610 WRITE(6,6008)ANAME,(TXARG(J),J=1,NNX)
 6008 FORMAT(1H0,A4,17H. INCREMENTS    X=,2X,7F13.5/26X,7F13.5/28X,7F13.5
     X )
      DO 613 II=1,NNY
      WRITE(6,6009) TYARG(II),(ZGP(JJ,II,I),JJ=1,NNX)
 6009 FORMAT(/2X,3H Y=,F13.5, 6X,7F13.5/26X,7F13.5/28X,7F13.5)
  613 CONTINUE
  614 CONTINUE
C
  700 CONTINUE
C
      CALL DRIVID
      WRITE(6,8000)
 8000 FORMAT (1H1)
      RETURN
      END
```

```
      SUBROUTINE INITEQ                                                 INIT0010
C             HVOSM-VD2 VERSION                                         INIT0020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                INIT0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    INIT0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      INIT0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      INIT0060
     3           XMS,XMUF,XIX,X1Y,XIZ,X1XZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,INIT0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  INIT0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     INIT0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   INIT0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   INIT0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    INIT0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  INIT0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),INIT0140
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN INIT0150
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,INIT0160
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          INIT0170
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  INIT0180
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    INIT0190
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    INIT0200
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPINIT0210
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     INIT0220
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    INIT0230
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  INIT0240
     9            ANG2,CPHI,SPH1,CPS1,SPS1,P1,P7,P3,P4,P5,P6,TX,TY,TZ   INIT0250
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  INIT0260
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  INIT0270
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,INIT0280
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2INIT0290
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,INIT0300
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           INIT0310
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 INIT0320
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       INIT0330
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,INIT0340
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   INIT0350
     2            CGYW(4),ZPG1(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),INIT0360
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     INIT0370
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  INIT0380
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  INIT0390
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     INIT0400
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      INIT0410
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),INIT0420
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)INIT0430
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      INIT0440
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     INIT0450
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   INIT0460
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  INIT0470
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       INIT0480
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   INIT0490
```

```
     1            (PSII(1),PSI1)                                        INIT0500
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     INIT0510
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   INIT0520
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   INIT0530
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       INIT0540
      LOGICAL LCB1,LCB2                                                 INIT0550
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       INIT0560
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),INIT0570
     2               NCAMF,NCAMR,NDTHF,NDTHR                            INIT0580
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   INIT0590
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     INIT0600
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          INIT0610
      DATA RPD/.01745329/                                               INIT0620
      RHF = 0.0                                                         INIT0630
      RHR = 0.0                                                         INIT0640
      IF(ISUS.NE.1) RHR = RHO                                           INIT0650
      IF(ISUS.EQ.2) RHF = RHOF                                          INIT0660
      CTHO = COS(THETAO*RPD)                                            INIT0670
      STHO = SIN(THETAO*RPD)                                            INIT0680
      SIR = XMS*A*G*CTHO/(A+B)                                          INIT0690
      SIF = XMS*G*CTHO-SIR                                              INIT0700
      DTF = (SIF/CTHO+XMUF*G)*0.5/AKT(1)                                INIT0710
      DTR = (SIR/CTHO+XMUR*G)*0.5/AKT(3)                                INIT0720
      SD1 = 0.5*(B*XMS*G/(A+B)-SIF)/AKF                                 INIT0730
      SD3 = 0.5*(A*XMS*G/(A+B)-SIR)/AKR                                 INIT0740
      HCG = -ZCOP                                                       INIT0750
      ZF = (HCG+A*STHO-RW(1)+DTF)/CTHO-RHF-SD1                          INIT0760
      ZR = (HCG-B*STHO-RW(3)+DTR)/CTHO-RHR-SD3                          INIT0770
      FR(1) = AKT(1)*DTF                                                INIT0780
      FR(2) = FR(1)                                                     INIT0790
      FR(3) = AKT(3)*DTR                                                INIT0800
      FR(4) = FR(3)                                                     INIT0810
      HI(1) = RW(1)-DTF                                                 INIT0820
      HI(2) = HI(1)                                                     INIT0830
      HI(3) = RW(3)-DTR                                                 INIT0840
      HI(4) = HI(3)                                                     INIT0850
      RETURN                                                            INIT0860
      END                                                               INIT0870
```

```
      SUBROUTINE INPUT
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION CARDIM(20),ICARD(300),DUM(18)
      DATA NBLKS/6/
      WRITE(6,1000)
 1000 FORMAT(1H1)
C          SET INPUT CARD COUNTER
      NC = 0
C        REWIND UNIT 2
      REWIND 2
C         READ A CARD
    1 READ(5,5000,END=999) (CARDIM(K),K=1,18),NSEQ,NCARD
 5000 FORMAT(18A4,2I4)
C        OUTPUT CARD IMAGE
      WRITE(2,2001) (CARDIM(K),K=1,18),NSEQ,NCARD
      WRITE(6,6000) (CARDIM(K),K=1,18),NSEQ,NCARD
 6000 FORMAT(1H ,18A4,2I4)
      IF(NCARD.GE.9999) GO TO 20
      IF(NCARD.LE.0) GO TO 90
      IF(NSEQ.GT.0) GO TO 1
      NC = NC+1
      ICARD(NC) = NCARD
      GO TO 1
   20 REWIND 2
C        TEST FOR AT LEAST ONE CARD OTHER THAN 9999
      IF(NC.LE.0) GO TO 91
C        SET COUNTER TO PROCESS ALL BLOCK NUMBERED CARDS
      IC = 1
C       DETERMINE CARD FORMAT AND TRANSFER TO PROPER CARD BLOCK
C        SUBROUTINE TO STORE DATA
   21 NBLK = ICARD(IC)/100
      NBCRD = ICARD(IC)-NBLK*100
C        FORMAT TEST
      IF(NBCRD.EQ.0) GO TO 22
C        NUMERIC INPUT
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD
 2000 FORMAT(9F8.0,2I4)
      GO TO 23
   22 CONTINUE
C        ALPHANUMERIC INPUT
      READ(2,2001) (DUM(K),K=1,18),NSEQ,NCARD
 2001 FORMAT(18A4,2I4)
C        BRANCH TO PROPER SUBROUTINE TO STORE INPUT
   23 IF(NBLK .LE.0) GO TO 92
      IF(NBLK.GT.NBLKS) GO TO 93
      GO TO(100,200,300,400,500,600),NBLK
C  PRINT ERROR MESSAGE HERE ?
  100 NERR = 0
```

```
      CALL BLK01(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0500
C        TEST FOR ERROR                                                 INPT0510
      IF(NERR.EQ.0) GO TO 30                                            INPT0520
      GO TO 94                                                          INPT0530
  200 CALL BLK02(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0540
      IF(NERR.EQ.0) GO TO 30                                            INPT0550
      GO TO 94                                                          INPT0560
  300 NERR = 0                                                          INPT0570
      CALL BLK03(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0580
      IF(NERR.EQ.0) GO TO 30                                            INPT0590
      GO TO 94                                                          INPT0600
  400 NERR = 0                                                          INPT0610
      CALL BLK04(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0620
      IF(NERR.EQ.0) GO TO 30                                            INPT0630
      GO TO 94                                                          INPT0640
  500 NERR = 0                                                          INPT0650
      CALL BLK05(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0660
      IF(NERR.EQ.0) GO TO 30                                            INPT0670
      GO TO 94                                                          INPT0680
  600 NERR = 0                                                          INPT0690
      CALL BLK06(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0700
      IF(NERR.EQ.0) GO TO 30                                            INPT0710
      GO TO 94                                                          INPT0720
   30 CONTINUE                                                          INPT0730
C        TEST IF ALL CARDS ARE READ                                     INPT0740
      IC = IC+1                                                         INPT0750
      IF(IC.GT.NC) GO TO 40                                             INPT0760
C        GET NEXT CARD FROM UNIT 2                                      INPT0770
      GO TO 21                                                          INPT0780
   40 CONTINUE                                                          INPT0790
C        SEARCH FOR END OF DATA                                         INPT0800
      READ(2,2001) (DUM(K),K=1,18),NSEQ,NCARD                           INPT0810
      IF(NCARD.NE.9999) GO TO 95                                        INPT0820
      GO TO 50                                                          INPT0830
  999 WRITE(6,6001) NCARD                                               INPT0840
 6001 FORMAT(56H UNEXPECTED END OF FILE ENCOUNTERED IN STMT NO. 1 OF SUBINPT0850
     1       34HROUTINE INPUT.  LAST CARD READ WAS   ,I4  )             INPT0860
      GO TO 49                                                          INPT0870
   90 WRITE(6,6002)                                                     INPT0880
 6002 FORMAT(56H A CARD NUMBERED LESS THAN OF EQUAL TO ZERO WAS ENCOUNTEINPT0890
     1       50HRED IN SUBROUTINE INPUT.  CARD IMAGE PRINTED ABOVE  )   INPT0900
      GO TO 49                                                          INPT0910
   91 WRITE(6,6003)                                                     INPT0920
 6003 FORMAT(33H THE NUMBER OF CARDS READ IS ZERO   )                   INPT0930
      GO TO 49                                                          INPT0940
   92 WRITE(6,6004)                                                     INPT0950
 6004 FORMAT(56H A BLOCK NUMBER OF LESS THAN OF EQUAL TO ZERO HAS BEEN OINPT0960
     1       7HBTAINED   )                                              INPT0970
      GO TO 49                                                          INPT0980
   93 WRITE(6,6005)                                                     INPT0990
 6005 FORMAT(56H A BLOCK NUMBER LARGER THAN THE ALLOWED NUMBER HAS BEEN INPT1000
```

```
    1         8HOBTAINED    )                                             INPT1010
      GO TO 49                                                            INPT1020
   94 WRITE(6,6006) NBLK,NBCRD,NSEQ,NCARD,NERR                            INPT1030
 6006 FORMAT(56H AN ERROR HAS OCCURRED IN STORING INPUT VALUES IN ONE OFINPT1040
    1         23H THE BLKXX SUBROUTINES.     /                            INPT1050
    2         39H THE CALLING ARGUMENTS FROM INPUT ARE :   /              INPT1060
    3          7H NBLK =,I4,2X,7HNBCRD =,I4,2X,6HNSEQ =,I4,2X,7HNCARD =,  INPT1070
    4          I4,2X,6HNERR =,I4     )                                    INPT1080
      GO TO 49                                                            INPT1090
   95 WRITE(6,6007)                                                       INPT1100
 6007 FORMAT(56H AN EXPECTED 9999 CARD HAS NOT BEEN ENCOUNTERED AFTER STINPT1110
    1         20H MT NO. 40 IN INPUT.      )                              INPT1120
   49 STOP                                                                INPT1130
   50 RETURN                                                              INPT1140
      END                                                                 INPT1150
```

```
      SUBROUTINE INTRPL(TABLE,XMIN,XMAX,DX,X,Y)
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
C     QUADRATIC INTERPOLATION SUBROUTINE INTRPL,ADDITIONAL ENTRY INTRPC
      DIMENSION TABLE(1)
      ENTRY INTRPC(TABLE,XMIN,XMAX,DX,X,Y,SLOPE)
    1 XLK = AMIN1(X,XMAX)
      XLK = AMAX1(XLK,XMIN)
      N1 = (XLK-XMIN)/DX+1.2
      N2 = N1+1
      NT = (XMAX-XMIN)/DX+1.2
      NO = N1-1
    2 IF(NO.GT.0) GO TO 3
      NO = N1
      N1 = N2
      N2 = N1+1
    3 IF(N2.LE.NT) GO TO 4
      N2 = N1
      N1 = NO
      NO = N1-1
    4 XXX = FLOAT(NO)*DX+XMIN
      DX2 = DX**2
      A = (TABLE(N2)-2.0*TABLE(N1)+TABLE(NO))/(2.0*DX2)
      B = (TABLE(N1)-TABLE(NO))/DX-A*(2.0*XXX-DX)
      C = TABLE(N1)-(A*XXX**2+B*XXX)
      Y = (A*XLK+B)*XLK+C
      SLOPE = 2.0 * A * XLK + B
      RETURN
      END
```

```
      SUBROUTINE INTRP5(INDX)
C              HVOSM-RD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1            (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
```

```
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHICIR,PHIC2R,AMUCMP,PHIID,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
C
C     NWHEEL = INDX
C     IXBDRY = 0
C     IYBDRY = 0
C     XLCEPT=0.0
C     XRCEPT=0.0
C     I5 = 0
C
      ERR = 0.0
   10 XXX = XP(INDX)
      YYY = YP(INDX)
      IT = 0
      DO 11 I=1,NZTAB
      IF( XB(I).EQ.XE(I) .OR. YB(I).EQ.YE(I)) GO TO 11
      IF(XXX.GE.XB(I).AND.XXX.LE.XE(I).AND.YYY.GE.YB(I).AND.YYY.LE.YE(I)
     X ) IT =I
   11 CONTINUE
      IF(IT.NE.0) GO TO 15
   13 ZPGI(INDX)= 0.0
      THGI(INDX)= 0.0
      PHGI(INDX)= 0.0
      XMUGI(INDX) = AMU(INDX)
      RETURN
C               ITV = 1 IDENTIFIES THE VARIABLE INCREMENT TABLE HERE.
   15 ITV = 0
      IF( IT.EQ. NZTAB .AND. NZ5.NE.0) ITV = 1
      XMUGI(INDX) = AMU(INDX)*AMUG(IT)
      XBT = XB(IT)
      XET = XE(IT)
      YBT = YB(IT)
      NXT = NX(IT)
      NYT = NY(IT)
      NBXT= NBX(IT)
      NBYT= NBY(IT)
      IF(ITV.GE.1) GO TO 20
C       TABLES WITH CONSTANT INCREMENT
      XINCRT  =  XINCR(IT)
      YINCRT  =  YINCR(IT)
      IX =(XXX-XBT)/XINCRT + 1.0
      IY =(YYY-YBT)/YINCRT + 1.0
      XX1 = XBT + FLOAT(IX-1)*XINCRT
      XX2 = XX1 + XINCRT
```

```
      YY1 = YBT + FLOAT(IY-1)*YINCRT
      YY3 = YY1 + YINCRT
      GO TO 40
C       IX IS LOW INDEX FOR X , IY IS LOW INDEX FOR Y
C       FLOAT(IX-1) IS   COUNT   OF INCREMENTS
C      VARIABLE INCREMENT TABLE (ARGUMENTS GIVEN, XXZPG5(21),YYZPG5(21))
   20 DO 22 I=2,NXT
      IF (XXZGP5(I) - XXX) 22,21,21
   21 IXX = I
      GO TO 25
   22 CONTINUE
      IXX = NXT
   25 IX = IXX-1
      IF (XXZGP5(IX) -XXZGP5(IXX)) 27,26,27
   26 ERR = 1.0
      GO TO 13
   27 DO 29 I=2,NYT
      IF (YYZGP5(I) - YYY) 29,28,28
   28 IYY= I
      GO TO 30
   29 CONTINUE
      IYY= NYT
   30 IY = IYY - 1
      IF(YYZGP5(IY)   - YYZGP5(IYY))35,26,35
   35 XX1 = XXZGP5(IX)
      XX2 = XXZGP5(IXX)
      YY1 = YYZGP5(IY)
      YY3 = YYZGP5(IYY)
      XINCRT = XX2 - XX1
      YINCRT = YY3 - YY1
   40 XX3 = XX1
      XX4 = XX2
C     SEARCH FOR Y BOUNDARIES IN THIS MESH.Y BOUNDARIES HAVE CONSTANT Y.
      IF (NBYT .EQ. 0) GO TO 54
      JJ = 0
      DO 41 I= 1,NBYT
      IF(YY1.GE.YBDRY(I,IT).OR. YBDRY(I,IT).GT.YY3) GO TO 41
      JJ = I
C
C     IYBDRY = I
C
      GO TO 42
   41 CONTINUE
   42 IF(JJ.EQ.0) GO TO 54
      IF(YYY.GE.YBDRY(JJ,IT))GO TO 50
      YY3 = YY1
      IF(ITV.GE.1) GO TO 44
   43 YY1 = YY3 - YINCRT
      IY = IY -1
      GO TO 54
   44 YY1 = YYZGP5(IY-1)
```

```
      IY = IY-1
      YINCRT = YY3 - YY1
      GO TO 54
   50 YY1 = YY3
      IF(ITV.GE.1) GO TO 52
   51 YY3 = YY1 + YINCRT
      IY = IY + 1
      GO TO 54
   52 YY3 = YYZGP5(IY +2)
      IY = IY + 1
      YINCRT =  YY3 - YY1
   54 YY2 = YY1
      YY4 = YY3
C       SEARCH FOR SLANTED BOUNDARIES
      IF (NBXT .EQ. 0) GO TO 61
      II = 0
      DO 60 I=1,NBXT
      XBDRT = XBDRY(I,IT)
C             PI AND 2.*PI ARE SINGULARITIES FOR COTAN
      IF( AMOD(PSBDRY(I,IT) , PI)  .EQ. 0.0) GO TO 60
      CTNPSB = COTAN(PSBDRY(I,IT))
      XLCEPT =  XBDRT + (YY1-YBT)*CTNPSB
      XRCEPT =  XBDRT + (YY3-YBT)*CTNPSB
      II= I
      IF( XX1.LE.XLCEPT .AND. XLCEPT.LE.XX2) GO TO 80
      IF(XLCEPT.LE.XX1 .AND. XRCEPT.GT.XX3) GO TO 80
      IF(XLCEPT.GE.XX2 .AND. XRCEPT.LT.XX4) GO TO 80
   60 CONTINUE
C       NO SLANT BOUNDARY IN THIS MESH
   61 XXMXX1 = XXX-XX1
      YYMYY1 = YYY-YY1
      ZPG1 = ZGP(IX ,IY ,IT)
      ZPG2 = ZGP(IX+1 ,IY ,IT)
      ZPG3 = ZGP(IX ,IY+1 ,IT)
      ZPG4 = ZGP(IX +1,IY+1 ,IT)
      ZZ1 =  ZPG1 + XXMXX1* (ZPG2-ZPG1)/XINCRT
      ZZ2 =  ZPG3 + XXMXX1* (ZPG4-ZPG3)/XINCRT
      ZPGI(INDX) = ZZ1 + YYMYY1*(ZZ2-ZZ1)/YINCRT
      THG1 = ATAN2 ((ZPG1-ZPG2),XINCRT)
      THG3 = ATAN2 ((ZPG3-ZPG4),XINCRT)
      THGI(INDX) =  THG1 + YYMYY1 *(THG3- THG1)/YINCRT
      IF(YYMYY1) 62,65,63
   62 ZPH1 = ZZ1
      ZYINCR =    -YYMYY1
      GO TO 67
   63 IF( YY3- YYY) 65,64,65
   64 PHGI(INDX) = ATAN2( (ZZ2 - ZZ1)/YINCRT, COS(THG1))
C     NOTE THG1, AS ROLL REFERENCE IS TO POINT 1 HERE
      GO TO 68
   65 ZPH1 = ZZ2
      ZYINCR =      YY3 - YYY
```

```
   67 PHGI(INDX) = ATAN2( (ZPH1-ZPGI(INDX))/ZYINCR, COS(THGI(INDX)))
   68 RETURN
 C 68 ZPGI10 = ZPGI(INDX)
 C    THGI10 = THGI(INDX)/RAD
 C    PHGI10 = PHGI(INDX)/RAD
 C3000 RETURN
 C       SLANT BOUNDARY IN THIS MESH
   80 ZXINCR = XINCRT
 C
 C    IXBDRY = II
 C
      IF( XXX .GT.(XBDRT + (YYY - YBT)* CTNPSB) ) GO TO 140
 C
 C    WHEEL HAS NOT CROSSED THE SLANT BOUNDARY, STEP BACK ON X ,PERHAPS.
 C    INDEX FOR HIGH GRID X IS IX+1, (XX2 AT IX+1,IY),(XX4 AT IX+1,IY+1)
 C       COUNT OF CONSTANT INCREMENTS FOR XX2 IS IX
      NXW = IX
      IF(ITV.GE.1) GO TO 93
   83 XX2W = XX2 + XINCRT
      DO 85 I=1,NXW
      XX2W = XX2W - XINCRT
      IF( XX2W .GE. XLCEPT) GO TO 85
      IX2W= IX +2 - I
      GO TO 90
   85 CONTINUE
      IX2W = 2
      XX2W = XBT+ XINCRT
   90 XX1 = XX2W - XINCRT
      XX4W = XX4 + XINCRT
      DO 92  I=1,NXW
      XX4W = XX4W- XINCRT
      IF(XX4W .GE. XRCEPT) GO TO 92
      IX4W = IX  +2 - I
      GO TO  100
   92 CONTINUE
      IX4W = 2
      XX4W = XBT+ XINCRT
      GO TO 100
   93 NXW5 = IX
      NXWW = IX +2
      DO 95  I= 1,NXW5
      IX2W = NXWW - I
      IF( XXZGP5(IX2W) .LT. XLCEPT) GO TO 96
   95 CONTINUE
      IX2W = 2
   96 XX2W = XXZGP5(IX2W)
      XX1   = XXZGP5(IX2W-1)
      XINCRT =  XX2W - XX1
      DO 97  I= 1,NXW5
      IX4W = NXWW - I
      IF(XXZGP5(IX4W) .LT. XRCEPT) GO TO 98
```

```
   97 CONTINUE
      IX4W =2
   98 XX4W = XXZGP5(IX4W)
  100 IX1W = IX2W -1
      IX3W = IX4W -1
      IF(IX1W  -   IX3W) 104,103,104
  103 IX = IX1W
      GO TO 61
  104 ZPG1 = ZGP(IX1W, IY, IT)
      ZPG2 = ZGP(IX2W, IY, IT)
      ZPG3 = ZGP(IX3W, IY+1, IT)
      ZPG4 = ZGP(IX4W, IY+1, IT)
      IF(IX2W - IX3W) 106,107,110
  106 ZPH1  =  ZGP(IX3W-1 ,IY+1, IT)
C         ZPH1 IS POINT FIVE HERE
      GO TO 108
  107 ZPH1   = ZPG3
  108 ZPH2   = ZPG2
      ZTH1   = ZPG3
      ZTH2   = ZPG4
      IF(ITV.GE.1)ZXINCR = XXZGP5(IX4W) - XXZGP5(IX3W)
      GO TO 115
  110 IF(IX1W - IX4W) 115,112,111
  111 I5 =MAX0(IX1W-1 ,1)
C       ZPH2  IS POINT FIVE HERE
      ZPH2     = ZGP(I5, IY,IT)
      GO TO 113
  112 ZPH2     = ZPG1
  113 ZPH1     = ZPG4
      ZTH1     = ZPG1
      ZTH2     = ZPG2
      IF(ITV.GE.1)ZXINCR = XXZGP5(IX2W) - XXZGP5(IX1W)
  115 ZZZ1  =  ZPG2
      XXMXX1 = XXX - XX2W
      YYMYY1 = YYY - YY2
      GO TO 180
C         WHEEL HAS CROSSED SLANT BOUNDARY. STEP AHEAD ON X, PERHAPS.
  140 NXW = NXT -1
      KXW = IX
      IF(ITV.GE.1) GO TO 153
  143 XX1W = XX1- XINCRT
      DO 145 I = KXW,NXW
      XX1W = XX1W + XINCRT
      IF(XX1W .LT. XLCEPT) GO TO 145
      IX1W = I
      GO TO 150
  145 CONTINUE
      XX1W = XET- XINCRT
      IX1W = NXW
  150 XX1 = XX1W
      XX3W = XX3 - XINCRT
```

```
      DO 152 I= KXW ,NXW                                                INT53050
      XX3W = XX3W + XINCRT                                              INT53060
      IF(XX3W .LT. XRCEPT) GO TO 152                                    INT53070
      IX3W = I                                                          INT53080
      GO TO 160                                                         INT53090
  152 CONTINUE                                                          INT53100
      IX3W = NXW                                                        INT53110
      XX3W = XET- XINCRT                                                INT53120
      GO TO 160                                                         INT53130
  153 DO 155  I = KXW, NXW                                              INT53140
      IF(XXZGP5(I) .LT. XLCEPT) GO TO 155                               INT53150
      IX1W = I                                                          INT53160
      GO TO 156                                                         INT53170
  155 CONTINUE                                                          INT53180
      IX1W = NXW                                                        INT53190
  156 XX1W = XXZGP5(IX1W)                                               INT53200
      XX1  = XX1W                                                       INT53210
      XINCRT =  XXZGP5(IX1W + 1) - XX1                                  INT53220
      DO 157 I= KXW ,NXW                                                INT53230
      IF(XXZGP5(I) .LT. XRCEPT) GO TO 157                               INT53240
      IX3W = I                                                          INT53250
      GO TO 158                                                         INT53260
  157 CONTINUE                                                          INT53270
      IX3W = NXW                                                        INT53280
  158 XX3W = XXZGP5(IX3W)                                               INT53290
  160 IX2W = IX1W + 1                                                   INT53300
      IX4W = IX3W + 1                                                   INT53310
      IF(IX1W - IX3W) 164,163,164                                       INT53320
  163 IX = IX1W                                                         INT53330
      GO TO 61                                                          INT53340
  164 ZPG1 = ZGP(IX1W,IY,IT)                                            INT53350
      ZPG2 = ZGP(IX2W,IY,IT)                                            INT53360
      ZPG3 = ZGP(IX3W,IY+1,IT)                                          INT53370
      ZPG4 = ZGP(IX4W,IY+1,IT)                                          INT53380
      IF(IX2W - IX3W) 166,167,170                                       INT53390
  166 ZPH2 = ZGP( IX2W+1,IY,IT)                                         INT53400
C         ZPH2 IS POINT FIVE HERE                                       INT53410
      GO TO 168                                                         INT53420
  167 ZPH2 = ZPG2                                                       INT53430
  168 ZPH1 = ZPG3                                                       INT53440
      ZTH1 = ZPG1                                                       INT53450
      ZTH2 = ZPG2                                                       INT53460
      IF(ITV.GE.1)ZXINCR = XXZGP5(IX2W) -XXZGP5(IX1W)                   INT53470
      GO TO 175                                                         INT53480
  170 IF(IX1W - IX4W) 175,172,171                                       INT53490
  171 I5 = MINO(IX4W+1,NXT)                                             INT53500
C       ZPH1 IS POINT FIVE HERE                                         INT53510
      ZPH1 = ZGP(I5,IY+1,IT)                                            INT53520
      GO TO 173                                                         INT53530
  172 ZPH1 = ZPG4                                                       INT53540
  173 ZPH2 = ZPG1                                                       INT53550
```

```
      ZTH1 = ZPG3
      ZTH2 = ZPG4
      IF(ITV.GE.1)ZXINCR= XXZGP5(IX4W) - XXZGP5(IX3W)
  175 ZZZ1 = ZPG1
      XXMXX1 = XXX - XX1
      YYMYY1 = YYY - YY1
  180 ZTH12 =  ZTH1-ZTH2
      TTANTH =  ZTH12/ZXINCR
      THGI(INDX) = ATAN2( ZTH12    , ZXINCR)
      TCOSTH = COS(THGI(INDX))
      PFAC = (ZPH1 - ZPH2)/YINCRT
      PHGI(INDX) = ATAN2( PFAC, TCOSTH)
      IF(TCOSTH) 186,185,186
  185 TTANPH = 0.0
      GO TO 187
  186 TTANPH =  PFAC/TCOSTH
  187 ZPGI(INDX) = ZZZ1 + YYMYY1*TCOSTH*TTANPH - XXMXX1* TTANTH
      RETURN
      END
```

```
      SUBROUTINE MATRIX                                                 MTRX0010
C              HVOSM-RD2 VERSION                                        MTRX0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               MTRX0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    MTRX0040
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,       MTRX0050
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                       MTRX0060
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,MTRX0070
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,   MTRX0080
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,      MTRX0090
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,    MTRX0100
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,    MTRX0110
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),     MTRX0120
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                   MTRX0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),MTRX0140
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN MTRX0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             MTRX0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))MTRX0170
     1         ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),MTRX0180
     2          (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),         MTRX0190
     3          (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),        MTRX0200
     4          (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),          MTRX0210
     5          (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),            MTRX0220
     6          (PSIFID,VAR(22))                                        MTRX0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      MTRX0240
     1          (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))MTRX0250
     2         ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),        MTRX0260
     3          (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),      MTRX0270
     4          (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),     MTRX0280
     5          (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),           MTRX0290
     6          (DPSIFI,DER(21)),(DDPSFI,DER(22))                       MTRX0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         MTRX0310
     1          (DER(10),DPHIFD)                                        MTRX0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       MTRX0330
     1          (DER(14),DDEL4D)                                        MTRX0340
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,MTRX0350
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),    MTRX0360
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),MTRX0370
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),      MTRX0380
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),   MTRX0390
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),   MTRX0400
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),      MTRX0410
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),       MTRX0420
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),MTRX0430
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)MTRX0440
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      MTRX0450
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),      MTRX0460
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),    MTRX0470
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                   MTRX0480
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       MTRX0490
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   MTRX0500
     1            (PSII(1),PSI1)                                         MTRX0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTRX0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         MTRX0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, MTRX0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   MTRX0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   MTRX0560
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTRX0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    MTRX0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   MTRX0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, MTRX0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  MTRX0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  MTRX0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, MTRX0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTRX0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTRX0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTRX0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          MTRX0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 MTRX0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       MTRX0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                MTRX0700
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       MTRX0710
     2              SLOPE1,SLOPE2,XTRA(300)                             MTRX0720
      DIMENSION UI(4),VI(4),WI(4)                                       MTRX0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                      MTRX0740
      DIMENSION APITCH(4)                                               MTRX0750
      EQUIVALENCE (APITCH(1),APTCH1)                                    MTRX0760
    1 CALL CLEAR (DMATX,DMATX(10,11))                                   MTRX0770
      DMATX(1,1) = SUMM                                                 MTRX0780
      DMATX(1,5) = GAM2                                                 MTRX0790
      DMATX(1,6) = RHOMUR*PHIR                                          MTRX0800
    2 DMATX(2,2) = SUMM                                                 MTRX0810
       DMATX(2,4) = -GAM2                                               MTRX0820
      DMATX(2,6) = GAM1                                                 MTRX0830
      DMATX(2,10) = -RHOMUR                                             MTRX0840
    3 DMATX(3,3) = XMS                                                  MTRX0850
    4 DMATX(4,2) = -GAM3                                                MTRX0860
      DMATX(4,4) = XIX+XIXP                                             MTRX0870
      DMATX(4,6) = -XIXZ-XIXZP                                          MTRX0880
      DMATX(4,10) = RHOMUR*ZRD3                                         MTRX0890
    5 DMATX(5,1) = GAM2                                                 MTRX0900
      DMATX(5,5) = XIY+XIYP                                             MTRX0910
      DMATX(5,6) = -XIYZP                                               MTRX0920
    6 DMATX(6,1) = DMATX(1,6)                                           MTRX0930
      DMATX(6,2) = GAM1                                                 MTRX0940
      DMATX(6,4) = DMATX(4,6)+BROMUR                                    MTRX0950
      DMATX(6,5) = -XIYZP                                               MTRX0960
      DMATX(6,6) = XIZR+XIZP                                            MTRX0970
      DMATX(6,10) = BROMUR                                              MTRX0980
    7 DMATX(7,3) = XMUFO2                                               MTRX0990
      DMATX(7,4) = XMTFO4                                               MTRX1000
```

```
      DMATX(7,5) = -AXMF02
      DMATX(7,7) = XMUF02
    8 DMATX(8,3) = XMUF02
      DMATX(8,4) = -XMTF04
      DMATX(8,5) = -AXMF02
      DMATX(8,8) = XMUF02
    9 DMATX(9,3) = XMUR
      DMATX(9,4) = -DMATX(1,6)
      DMATX(9,5) = BMUR
      DMATX(9,9) = XMUR
      DMATX(9,10) = DMATX(9,4)
   10 DMATX(10,2) = -RHOMUR
      DMATX(10,3) = DMATX(9,4)
      DMATX(10,4) = XIR+ZRD3R*RHOMUR
      DMATX(10,5) = -BROMUR*PHIR
      DMATX(10,6) = BROMUR
      DMATX(10,9) = DMATX(9,4)
      DMATX(10,10) = RHMR2I
   11 GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
   12 DMATX(1,11) = SUMM*(VR-WQ-GSTH)-GAM2*PR+RHOMUR*PHIR*PQ+GAM1*(Q2+
     1            R2)-GAM6*Q-2.0*RHOMUR*RPHRD+SFXS+SFXU
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)+GAM6*P-GAM1*PQ-GAM2*QR-RHOMUR*
     1            PHIR*(P2+R2+PHIRD2)+SFYS+SFYU
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = GAM3*(UR-WP-GCTSP)+(XIXZ+XIXZP)*PQ-GAM7*P+(XIY-XIZ+
     1            XIXP)*QR-GAM4*(P2+R2)+RHOMUR*PHIR*ZRD3*PHIRD2+SNPS+
     2            SNPU
      DMATX(5,11) = XIXZ*(R2-P2)+(XIZ-XIX-XIYP)*PR+GAM2*(VR-WQ-GSTH)-
     1            (GAM7+2.*RHO*TG61)*Q+(XIXZP-BROMUR)*(Q2+R2)-
     2            XIYZP*PQ-2.0*XMUR*ZRD3R*RHO*RPHRD+SNTS+SNTU
   13 DMATX(6,11) = (XIX-XIY-GAM5)*PQ-(XIXZ+XIXZP-BROMUR)*QR+GAM8*Q+
     1            XIYZP*PR+GAM9*P+RHOMUR*PHIR*(VR-WQ-2.0*RHO*RPHRD-B*
     2            (Q2-P2-PHIRD2)-GSTH)+GAM1*(WP-UR+GCTSP)+SNPSS+SNPSU
      DMATX(7,11) = XMUF02*(UQ-VP-A*PR-TF02*QR+(ZF+DEL1)*(P2+Q2)+GCTCP)+
     1            FZU(1)+SI(1)
      DMATX(8,11) = XMUF02*(UQ-VP-A*PR+TF02*QR+(ZF+DEL2)*(P2+Q2)+GCTCP)+
     1            FZU(2)+SI(2)
      DMATX(9,11) = XMUR*(UQ-VP+RHO*PHIRD2+2.0*P*RHO*PHIRD+B*PR+RHO*PHIR
     1            *QR+ZRD3R*(P2+Q2)+GCTCP)+FZU(3)+FZU(4)+SI(3)+SI(4)
   14 DMATX(10,11) = RHOMUR*(UR-WP-2.0*P*(DEL3D-RHO*PHIR*PHIRD)-B*PQ+
     1             RHO*PHIR*(P2+R2)+ZRD3R*QR-GCTH*SIN(PHIT+PHIR))+
     2         PHIR*RHOMUR*(VP-UQ-2.0*P*RHO*PHIRD-B*PR-RHO*PHIR*QR-
     3             ZRD3R*(P2+Q2))-XIR*PHIR*(R2-Q2)-XIR*QR+
     4             SNPR
      RETURN
      END
```

```
      SUBROUTINE MTRXIR                                                 MTXI0010
C               HVOSM-RD2 VERSION                                       MTXI0020
C               REVISED OCTOBER 1975   CALSPAN CORPORATION              MTXI0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    MTXI0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      MTXI0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      MTXI0060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,MTXI0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  MTXI0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     MTXI0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   MTXI0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   MTXI0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    MTXI0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  MTXI0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),MTXI0140
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN MTXI0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             MTXI0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))MTXI0170
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),MTXI0180
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        MTXI0190
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       MTXI0200
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         MTXI0210
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           MTXI0220
     6           (PSIFID,VAR(22))                                       MTXI0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      MTXI0240
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))MTXI0250
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       MTXI0260
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     MTXI0270
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    MTXI0280
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          MTXI0290
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      MTXI0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         MTXI0310
     1           (DER(10),DPHIFD)                                       MTXI0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       MTXI0330
     1           (DER(14),DDEL4D)                                       MTXI0340
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,MTXI0350
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   MTXI0360
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),MTXI0370
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     MTXI0380
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  MTXI0390
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  MTXI0400
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     MTXI0410
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      MTXI0420
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),MTXI0430
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)MTXI0440
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      MTXI0450
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     MTXI0460
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   MTXI0470
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  MTXI0480
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       MTXI0490
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),    MTXI0500
     1            (PSII(1),PSI1)                                         MTXI0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTXI0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          MTXI0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  MTXI0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    MTXI0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    MTXI0560
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTXI0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     MTXI0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    MTXI0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  MTXI0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   MTXI0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   MTXI0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  MTXI0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTXI0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTXI0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTXI0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           MTXI0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  MTXI0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        MTXI0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                 MTXI0700
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,        MTXI0710
     2              SLOPE1,SLOPE2,XTRA(300)                              MTXI0720
      DIMENSION UI(4),VI(4),WI(4)                                        MTXI0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       MTXI0740
      DIMENSION APITCH(4)                                                MTXI0750
      EQUIVALENCE (APITCH(1),APTCH1)                                     MTXI0760
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       MTXI0770
     1                 RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     MTXI0780
     2                 DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    MTXI0790
     3                 PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    MTXI0800
     4                 PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        MTXI0810
     5                 DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     MTXI0820
C                                                                        MTXI0830
      CALL CLEAR(DMATX(1,1),DMATX(10,11))                                MTXI0840
      DMATX(1,1) = SUMM                                                  MTXI0850
      DMATX(1,5) = GAM2                                                  MTXI0860
      DMATX(2,2) = SUMM                                                  MTXI0870
      DMATX(2,4) = -GAM2                                                 MTXI0880
      DMATX(2,6) = GAM1                                                  MTXI0890
      DMATX(3,3) = XMS                                                   MTXI0900
      DMATX(4,2) = -GAM2                                                 MTXI0910
      DMATX(4,4) = XIX+XIXP                                              MTXI0920
      DMATX(4,6) = -XIXZ-XIXZP                                           MTXI0930
      DMATX(5,1) = GAM2                                                  MTXI0940
      DMATX(5,5) = XIY+XIYP                                              MTXI0950
      DMATX(5,6) = -XIYZP                                                MTXI0960
      DMATX(6,2) = GAM1                                                  MTXI0970
      DMATX(6,4) = -XIXZ-XIXZP                                           MTXI0980
      DMATX(6,5) = -XIYZP                                                MTXI0990
      DMATX(6,6) = XIZ+XIZP                                              MTXI1000
```

```
      DMATX(7,3) = XMUFO2
      DMATX(7,4) = XMTFO4
      DMATX(7,5) = -AXMFO2
      DMATX(7,7) = XMUFO2
      DMATX(8,3) = XMUFO2
      DMATX(8,4) = -XMTFO4
      DMATX(8,5) = -AXMFO2
      DMATX(8,8) = XMUFO2
      DMATX(9,3) = XMURO2
      DMATX(9,4) = XMTRO4
      DMATX(9,5) = BXMRO2
      DMATX(9,9) = XMURO2
      DMATX(10,3) = XMURO2
      DMATX(10,4) = -XMTRO4
      DMATX(10,5) = BXMRO2
      DMATX(10,10) = XMURO2
      GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
      DMATX(1,11) = SUMM*(VR-WQ-GSTH)+GAM1*(Q2+R2)-GAM2*PR-GAM6*Q
     1            +SFXU+SFXS
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)-GAM1*PQ-GAM2*QR+GAM6*P
     1            +SFYU+SFYS
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = -GAM2*(WP-UR+GCTSP)+(XIXZ+XIXZP)*PQ-XIYZP*(P2+R2)
     1            +(XIY-XIZ+XIXP)*QR-GAM7*P+SNPU+SNPS
      DMATX(5,11) = GAM2*(VR-WQ-GSTH)-(XIX-XIZ+XIYP)*PR-GAM7*Q
     1            +XIXZP*(Q2+R2)-XIYZP*PQ+XIXZ*(R2-P2)+SNTU+SNTS
      DMATX(6,11) = GAM1*(WP-UR+GCTSP)+(XIX-XIY-GAM5)*PQ-(XIXZ+XIXZP)
     1            *QR+GAM8*Q+XIYZP*PR+GAM9*P+SNPSU+SNPSS
      DMATX(7,11) = XMUFO2*(UQ-VP+GCTCP-A*PR+ZFD1*(P2+R2)
     1            -TFO2*QR)+FZU(1)+SI(1)
      DMATX(8,11) = XMUFO2*(UQ-VP+GCTCP-A*PR+ZFD2*(P2+R2)
     1            +TFO2*QR)+FZU(2)+SI(2)
      DMATX(9,11) = XMURO2*(UQ-VP+GCTCP+B*PR+ZRD3*(P2+R2)
     1            -TRO2*QR)+FZU(3)+SI(3)
      DMATX(10,11) = XMURO2*(UQ-VP+GCTCP+B*PR+ZRD4*(P2+R2)
     1            +TRO2*QR)+FZU(4)+SI(4)
      RETURN
      END
```

```
      SUBROUTINE MTRXSF
C           HVOSM-RD2 VERSION
C           REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1           A,B,DEL1O,DEL2O,DEL3O,PHIRO,DEL1OD,DEL2OD,DEL3OD,
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),     MTXS0500
     1            (PSII(1),PSI1)                                          MTXS0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTXS0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          MTXS0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  MTXS0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    MTXS0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    MTXS0560
     5             ZRD3,ZRD3R, ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTXS0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     MTXS0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    MTXS0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  MTXS0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   MTXS0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   MTXS0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  MTXS0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTXS0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTXS0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTXS0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           MTXS0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                   MTXS0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        MTXS0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                  MTXS0700
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,        MTXS0710
     2              SLOPE1,SLOPE2,XTRA(300)                               MTXS0720
      DIMENSION UI(4),VI(4),WI(4)                                         MTXS0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                        MTXS0740
      DIMENSION APITCH(4)                                                 MTXS0750
      EQUIVALENCE (APITCH(1),APTCH1)                                      MTXS0760
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        MTXS0770
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),MTXS0780
     2               NCAMF,NCAMR,NDTHF,NDTHR                              MTXS0790
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       MTXS0800
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,      MTXS0810
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,     MTXS0820
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,     MTXS0830
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,         MTXS0840
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                      MTXS0850
C                                                                         MTXS0860
      CALL CLEAR(DMATX(1,1),DMATX(10,11))                                MTXS0870
      DMATX(1,1) = SUMM                                                   MTXS0880
      DMATX(1,5) = GAM2                                                   MTXS0890
      DMATX(1,6) = RHOMUR*PHIR+RHFMUF*PHIF                                MTXS0900
      DMATX(2,2) = SUMM                                                   MTXS0910
      DMATX(2,4) = -GAM2                                                  MTXS0920
      DMATX(2,6) = GAM1                                                   MTXS0930
      DMATX(2,8) = -RHFMUF                                                MTXS0940
      DMATX(2,10)= -RHOMUR                                                MTXS0950
      DMATX(3,3) = XMS                                                    MTXS0960
      DMATX(4,2) = -GAM3                                                  MTXS0970
      DMATX(4,4) = XIX+XIXP                                               MTXS0980
      DMATX(4,6) = -XIXZ-XIXZP                                            MTXS0990
      DMATX(4,8) = RHFMUF*ZFD1                                            MTXS1000
```

```
      DMATX(4,10)= RHOMUR*ZRD3
      DMATX(5,1) = GAM2
      DMATX(5,5) = XIY+XIYP
      DMATX(5,6) = -XIYZP
      DMATX(6,1) = RHFMUF*PHIF+RHOMUR*PHIR
      DMATX(6,2) = GAM1
      DMATX(6,4) = -XIXZ-XIXZP-RHFMUF*A+RHOMUR*B
      DMATX(6,5) = -XIYZP
      DMATX(6,6) = XIZ+XIZP+XIR+XIF
      DMATX(6,8) = -RHFMUF*A
      DMATX(6,10)= BROMUR
      DMATX(7,3) = XMUF
      DMATX(7,4) = -RHFMUF*PHIF
      DMATX(7,5) = -AMUF
      DMATX(7,7) = XMUF
      DMATX(7,8) = -RHFMUF*PHIF
      DMATX(8,2) = -RHFMUF
      DMATX(8,3) = -RHFMUF*PHIF
      DMATX(8,4) = XIF+RHFMUF*ZFD1RF
      DMATX(8,5) = AMUF*RFPF
      DMATX(8,6) = -RHFMUF*A
      DMATX(8,7) = -RHFMUF*PHIF
      DMATX(8,8) = RF2MFI
      DMATX(9,3) = XMUR
      DMATX(9,4) = -RHOMUR*PHIR
      DMATX(9,5) = BMUR
      DMATX(9,9) = XMUR
      DMATX(9,10)= -RHOMUR*PHIR
      DMATX(10,2) = -RHOMUR
      DMATX(10,3) = -RHOMUR*PHIR
      DMATX(10,4) = XIR+RHOMUR*ZRD3R
      DMATX(10,5) = -BMUR*RPR
      DMATX(10,6) = BROMUR
      DMATX(10,9) = -RHOMUR*PHIR
      DMATX(10,10)= RHMR2I
      GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
      DMATX(1,11) = SUMM*(VR-WQ-GSTH)-GAM2*PR+(RHOMUR*PHIR+RHFMUF*PHIF)
     1             *PQ+GAM1*(Q2+R2)-GAM6*Q-2.0*(RHOMUR*RPHRD+RHFMUF*
     2              RPHFD)+SFXU+SFXS
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)+GAM6*P-GAM1*PQ-GAM2*QR
     1             -RHOMUR*PHIR*(P2+R2+PHIRD2)-RHFMUF*PHIF*(P2+R2+
     2              PHIFD2)+SFYU+SFYS
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = GAM3*(UR-WP-GCTSP)+(XIXZ+XIXZP)*PQ-GAM7*P
     1             +(XIY-XIZ+XIXP)*QR-GAM4*(P2+R2)+RHOMUR*PHIR*ZRD3*
     2              PHIRD2+RHFMUF*PHIF*ZFDI*PHIFD2+SNPS+SNPU
      DMATX(5,11) = GAM2*(VR-WQ-GSTH)+XIXZ*(R2-P2)+(XIZ-XIX-XIYP)*PR
     1             -GAM7*Q-2.0*Q*(RHOF*WFMF+RHO*TG6I)+(XIXZP-BROMUR
     2             +RHOF*AMUF)*(Q2+R2)-XIYZP*PQ-2.0*RHOMUR*ZRD3R*RPHRD
     3             -2.0*RHFMUF*ZFDIRF*RPHFD+SNTU+SNTS
```

```
      DMATX(6,11) = GAM1*(WP-UR+GCTSP)+(XIX-XIY-GAM5)*PQ                MTXS1520
     1              -(XIXZ+XIXZP-BROMUR+AMUF*RHOF)*QR+GAM8*Q+XIYZP*PR   MTXS1530
     2              +GAM9*P+XMUR*RPR*(VR-WQ-2.0*RHO*RPHRD-B*(Q2-P2      MTXS1540
     3             -PHIRD2)-GSTH)+XMUF*RFPF*(VR-WQ-2.0*RHOF*RPHFD       MTXS1550
     4              +A*(Q2-P2-PHIFD2)-GSTH)+SNPSS+SNPSU                 MTXS1560
      DMATX(7,11) = XMUF*(UQ-VP+RHOF*PHIFD2+2.0*P*RHOF*PHIFD-A*PR       MTXS1570
     1              +RFPF*QR+ZFD1RF*(P2+Q2)+GCTCP)                      MTXS1580
     2              +FZU(1)+FZU(2)+SI(1)+SI(2)                          MTXS1590
      DMATX(8,11) = RHFMUF*(UR-WP-2.0*P*DEL1D+2.0*P*RFPF*PHIFD+A*PQ     MTXS1600
     1              +RFPF*(P2+R2)+ZFD1RF*QR-GCTH*SIN(PHIT+PHIF))        MTXS1610
     2              +RHFMUF*PHIF*(VP-UQ-2.0*P*RHOF*PHIFD+A*PR           MTXS1620
     3              -ZFD1RF*(P2+Q2))-XIF*PHIF*(R2-Q2)-XIF*QR+SNPF       MTXS1630
      DMATX(9,11) = XMUR*(UQ-VP+RHO*PHIRD2+2.0*P*RHO*PHIRD+B*PR         MTXS1640
     1              +RPR*QR+ZRD3R*(P2+Q2)+GCTCP)                        MTXS1650
     2              +FZU(3)+FZU(4)+SI(3)+SI(4)                          MTXS1660
      DMATX(10,11)= RHOMUR*(UR-WP-2.0*P*DEL3D+2.0*P*RPR*PHIRD-B*PQ      MTXS1670
     1              +RPR*(P2+R2)+ZRD3R*QR-GCTH*SIN(PHIT+PHIR))          MTXS1680
     2              +RHOMUR*PHIR*(VP-UQ-2.0*P*RHO*PHIRD-B*PR            MTXS1690
     3              -ZRD3R*(P2+Q2))-XIR*PHIR*(R2-Q2)-XIR*QR+SNPR        MTXS1700
      RETURN                                                            MTXS1710
      END                                                               MTXS1720
```

```
      SUBROUTINE NLDFL
C               HVOSM-RD2 VERSION
C               REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,
     2               DELTB,XINPT(100)
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),
     1              YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),
     2              AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,
     3              CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),
     4              YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT
     5              ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,
     6              XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,
     7              AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,
     8              FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,
     9              NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,
     A              RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,
     B              SWORK,SPENGY,DISS,IPLN,ILOAD
      DIMENSION INDXPT(4)
      EQUIVALENCE (INDXPT(1),I1)
      REAL*8 ALIM1/'UPPER'/,ALIM2/'LOWER'/,XLIM
    1 XLP = VMAX11
      VSIGN = 0.0
      VMAX11 = (YBPT-YBPTP)/DT
      VMAX(1) = VMAX11
      IF(VMAX(1))200,201,202
  200 VSIGN = -1.0
      GO TO 203
  201 VSIGN = 0.0
      GO TO 300
```

```
202 VSIGN = 1.0                                                            NLDF0500
203 XL = DELBB                                                             NLDF0510
    EPSP = EPSL                                                            NLDF0520
    XVLP = VL                                                              NLDF0530
  2 VL = XL-EPSL                                                           NLDF0540
    IF(VL.GT.0.0) GO TO 3                                                  NLDF0550
    FB = 0.0                                                               NLDF0560
    GO TO 13                                                               NLDF0570
  3 IF(VMAX11.LT.0.0) GO TO 9                                              NLDF0580
    IF(ILOAD.EQ.1.AND.VSIGN.EQ.1.0)GO TO 6                                 NLDF0590
    IF(ILOAD.EQ.1.AND.VSIGN.NE.1.0)GO TO 10                                NLDF0600
    FB = SIGR(1)                                                           NLDF0610
    XX = VL                                                                NLDF0620
    YY = VMAX(1)                                                           NLDF0630
  4 DO 5  I=2,6                                                            NLDF0640
    J = I+5                                                                NLDF0650
    FB = FB+SIGR(I)*XX+SIGR(J)*YY                                          NLDF0660
    XX = XX*VL                                                             NLDF0670
    YY = YY*VMAX(1)                                                        NLDF0680
  5 CONTINUE                                                               NLDF0690
    ILOAD = 0                                                              NLDF0700
    GO TO 13                                                               NLDF0710
  6 IF(XF.GT.0.0)GO TO 7                                                   NLDF0720
    EPSL = SET*DELX+EPSL                                                   NLDF0730
    DELX = 0.0                                                             NLDF0740
    GO TO 8                                                                NLDF0750
  7 XX = VL                                                                NLDF0760
    YY = (SET*DELX+VL)/2.0                                                 NLDF0770
    CALL POLY(SIGR(1),SIGR(2),XX,YY,FB)                                    NLDF0780
    EPSL = YY+EPSL                                                         NLDF0790
  8 ILOAD = 0                                                              NLDF0800
    NCYC = NCYC+1                                                          NLDF0810
    GO TO 2                                                                NLDF0820
  9 IF(ILOAD.EQ.0)GO TO 212                                                NLDF0830
    IF(VSIGN.NE.1.0)GO TO 10                                               NLDF0840
212 DELX = VL                                                              NLDF0850
    FUM = FB                                                               NLDF0860
    GI1 = SET-1.0                                                          NLDF0870
    GI3 = GI1**3*DELX                                                      NLDF0880
    FDL = FUM*DELX                                                         NLDF0890
    EEE = EEE - ENGY                                                       NLDF0900
    XVLP = VL                                                              NLDF0910
 90 CD0 = SET*(FDL*(SET**2+SET-2.0)+6.0*EEE)/GI3                           NLDF0920
    CD1 = -2.0*(FDL*(2.0*SET+1.0)*GI1+3.0*EEE*(SET+1.0))/(DELX*GI3)        NLDF0930
    CD2 = 3.0*(FDL*GI1+2.0*EEE)/(DELX**2*GI3)                              NLDF0940
    ILOAD = 1                                                              NLDF0950
    TMP = FB*DELX*(1.0-SET)                                                NLDF0960
    IF(TMP.LE.3.0*EEE) GO TO 91                                            NLDF0970
    RTMP = 1.005*CONS*TMP/(3.0*EEE)                                        NLDF0980
    ETMP = 1.005*TMP/3.0                                                   NLDF0990
    XLIM = ALIM2                                                           NLDF1000
```

```
      GO TO 92
   91 IF(TMP.GE.2.0*EEE) GO TO 10
      RTMP = .995*CONS*TMP/(2.0*EEE)
      ETMP = .995*TMP/2.0
      XLIM = ALIM1
   92 IF(NCYC.GT.0) GO TO 93
      TMPR = CONS
      CONS = RTMP
      EEE = ETMP
      WRITE(6,1000) TMPR,CONS,T
 1000 FORMAT(19HOCONS CHANGED FROM ,E12.5,4H TO ,E12.5,8H AT T = ,F9.6)
      GO TO 90
   93 EEE = ETMP
      WRITE(6,1001) XLIM,T
 1001 FORMAT(27HOCONSERVED ENERGY RESET TO ,A6,14H LIMIT AT T = ,F9.6)
      GO TO 90
   10 IF(VL-SET*DELX.GT.0.0) GO TO 12
   11 FB = 0.0
      DELBB = EPSL+SET*DELX
      GO TO 13
   12 FB = CD0+CD1*VL+CD2*VL**2
      FB = AMAX1(FB,0.0)
   13 ENGY = (XF+FB)*(VL+EPSL-EPSP-XVLP)/2.0
      ENRGY = ENRGY+ENGY
      IF(VMAX11.LT.0.0) GO TO 14
      EPL = ENGY
      EMI = 0.0
      GO TO 15
   14 EMI = ENGY
      EPL = 0.0
   15 EEE = EEE+CONS*EPL+EMI
  300 XF = FB
      RETURN
      ENTRY NLDFRC
   16 WL = DELBB-EPSL
      IF(ILOAD.NE.0)GO TO 19
      XX = WL
      YY = VMAX(1)
      FB = SIGR(1)
   17 DO 18 I=2,6
      J = I+5
      FB = FB+SIGR(I)*XX+SIGR(J)*YY
      XX = XX*WL
      YY = YY*VMAX(1)
   18 CONTINUE
      GO TO 20
   19 IF(WL-SET*DELX.GT.0.0)GO TO 100
      FB = 0.0
      RETURN
  100 FB = CD0+CD1*WL+CD2*WL**2
   20 FB = AMAX1(FB,0.0)
```

```
      RETURN                                                        NLDF1520
      END                                                           NLDF1530
```

```
      SUBROUTINE OUTPUT(IND)                                            OUTP0010
C          HVOSM-RD2 VERSION                                            OUTP0020
C          REVISED OCTOBER 1975   CALSPAN CORPORATION                   OUTP0030
C          HVOSM-RD2 VERSION                                            OUTP0040
C          REVISED OCTOBER 1975  CALSPAN CORPORATION                    OUTP0050
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        OUTP0060
     1             NPAGE(20)                                            OUTP0070
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    OUTP0080
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     OUTP0090
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     OUTP0100
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,OUTP0110
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, OUTP0120
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    OUTP0130
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  OUTP0140
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  OUTP0150
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   OUTP0160
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 OUTP0170
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),OUTP0180
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN OUTP0190
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             OUTP0200
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))OUTP0210
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),OUTP0220
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       OUTP0230
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      OUTP0240
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        OUTP0250
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          OUTP0260
     6            (PSIFID,VAR(22))                                      OUTP0270
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      OUTP0280
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))OUTP0290
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       OUTP0300
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    OUTP0310
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   OUTP0320
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         OUTP0330
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     OUTP0340
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         OUTP0350
     1            (DER(10),DPHIFD)                                      OUTP0360
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       OUTP0370
     1            (DER(14),DDEL4D)                                      OUTP0380
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,OUTP0390
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  OUTP0400
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),OUTP0410
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    OUTP0420
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), OUTP0430
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), OUTP0440
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    OUTP0450
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     OUTP0460
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),OUTP0470
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)OUTP0480
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      OUTP0490
```

```
1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     OUTP0500
2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   OUTP0510
3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  OUTP0520
 DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                        OUTP0530
 EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),    OUTP0540
1           (PSII(1),PSI1)                                          OUTP0550
 COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,OUTP0560
1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,           OUTP0570
2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  OUTP0580
3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    OUTP0590
4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    OUTP0600
5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPOUTP0610
6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     OUTP0620
7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    OUTP0630
8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  OUTP0640
9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   OUTP0650
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   OUTP0660
1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  OUTP0670
2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,OUTP0680
3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2OUTP0690
4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,OUTP0700
5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL            OUTP0710
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  OUTP0720
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        OUTP0730
 COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                      OUTP0740
1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,     OUTP0750
2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),     OUTP0760
3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)         OUTP0770
 LOGICAL LCB1,LCB2                                                  OUTP0780
 COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                 OUTP0790
1             XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,         OUTP0800
2             SLOPE1,SLOPE2,XTRA(300)                               OUTP0810
 DIMENSION UI(4),VI(4),WI(4)                                        OUTP0820
 EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       OUTP0830
 DIMENSION APITCH(4)                                                OUTP0840
 EQUIVALENCE (APITCH(1),APTCH1)                                     OUTP0850
 COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),   OUTP0860
1             YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),      OUTP0870
2             AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,     OUTP0880
3             CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),        OUTP0890
4             YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT  OUTP0900
5             ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4,  OUTP0910
6             XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,      OUTP0920
7             AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,OUTP0930
8             FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,OUTP0940
9             NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,        OUTP0950
A             RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,           OUTP0960
B             SWORK,SPENGY,DISS,IPLN,ILOAD                          OUTP0970
 DIMENSION INDXPT(4)                                                OUTP0980
 EQUIVALENCE (INDXPT(1),I1)                                         OUTP0990
 COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        OUTP1000
```

```
     1                AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2                NCAMF,NCAMR,NDTHF,NDTHR
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,
     1                 RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,
     2                 DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,
     3                 PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,
     4                 PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,
     5                 DTDD2,DTDD3,DTDD4,FJF(4),SNPF
      COMMON/BARSTR/XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),
     1              ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),
     2              FNSTI(3),AKST(3)
      DIMENSION ASTR(4),SLPANG(4)
      DIMENSION HDEF(3)
      DATA STAR,BLNK/1H*,1H /
      DATA LPP/50/
      DATA TTTTTT/-9999.0/
      IF(IND.NE.0) GO TO 400
      LINES = 0
      RETURN
  400 LINES = LINES+1
      IF(MOD(LINES,LPP).NE.1) GO TO 500
      XPAGE = 0.01*(LINES+LPP-1)/LPP
      NT = 10
      IS1 = ISUS+1
      DO 410 J=1,19
      IF(NPAGE(J).EQ.0) GO TO 410
      NT = NT+1
      PAGE = NT+XPAGE
      WRITE(NT,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2      (GHED(I),I=1,10),(SHED(I),I=1,10),PAGE
 1000 FORMAT(1H1,19X,18A4,20X,2A4 / 5X,3(10A4) / 5X,2(10A4),
     1       22X,4HPAGE,1X,F8.2  /  )
     GO TO(111,112,113,114,115,116,117,118,119,120,121,122,123,124,
     *      125,126,127,128,129),J
C  POSSIBLE ERROR MESSAGE
      GO TO 410
  111 WRITE(NT,1100)
 1100 FORMAT(1H0,48X,23HS P R U N G    M A S S   /
     A62H  TIME   |          POSITION (FEET)           |      VELOCITY (
     B62HFT/SEC)            |            ACCELERATION (G-UNITS)
     C 6H     |  /
     D62H   SEC   |     XC'    |     YC'    |     ZC'    | FORWARD  |  LA
     E62HTERAL  | VERTICAL |    LONG.  |      LAT.  |      VERT.  |  RESU
     F 6HLT.  |   /)
      GO TO 410
  112 IF(ISUS.EQ.1) GO TO 1121
      WRITE(NT,1200)
 1200 FORMAT(
     A62H0        |                                S P R U N G   M A S S
     B62H                   |  SIDESLIP |    COURSE |FRONT STEER| REAR
```

```
    C 6HSTEER|  /
    D62H  TIME   |   ANGULAR VELOCITIES (DEG/SEC)     |      ORIENTATIO
    E62HN (DEGREES)        |     ANGLE   |    ANGLE   |    ANGLE   |    AN
    F 6HGLE  |  /
    G62H   SEC   |     P     |     Q     |     R     |    ROLL    |   P
    H62HITCH   |     YAW    |    DEG    |    DEG    |    DEG    |     D
    I 6HEG   |  /)
     GO TO 410
1121 WRITE(NT,1201)
1201 FORMAT(
    A62HO        |                        S P R U N G   M A S S
    B62H                   |  SIDESLIP |    COURSE |FR.STEER|RR STEER|
    C 9HLR STEER|  /
    D62H  TIME   |   ANGULAR VELOCITIES (DEG/SEC)     |      ORIENTATIO
    E62HN (DEGREES)        |     ANGLE   |    ANGLE   | ANGLE |  ANGLE |
    F 9H  ANGLE |  /
    G62H   SEC   |     P     |     Q     |     R     |    ROLL    |   P
    H62HITCH   |     YAW    |    DEG    |    DEG    |  DEG  |   DEG  |
    I 9H   DEG  |  /   )
     GO TO 410
 113 WRITE(NT,1300)
1300 FORMAT(
    A62HO TIME   |       WHEEL RIDE DISPLACEMENTS (INCHES)        |
    B44H     WHEEL RIDE VELOCITIES (IN/SEC)          |   /
    C62H   SEC   |       RF    |     LF     |      RR     |       LR    |
    D44H  RF   |      LF    |      RR    |      LR    |   /)
     GO TO 410
 114 GO TO(1140,1141,1142),IS1
1140 WRITE(NT,1400)
1400 FORMAT(55HO          |               SPRUNG MASS              |   WHEEL
    A62HRIDE ACCEL  |     REAR ROLL CENTER RIDE        |     REAR AXLE A
    B15HNGULAR         |  /
    C62H  TIME    | ANGULAR ACCELERATIONS (DEG/SEC**2)|    (IN/SEC**2)
    D62H      | DEFL    | VELOCITY |ACCELERATION|  DEFL   | VELOCITY |  A
    E 9HCCEL   |   /
    F62H   SEC   |     DP/DT |     DQ/DT |     DR/DT |       RF   |
    G62H LF   | INCHES |   IN/SEC | IN/SEC**2   |   DEG   |  DEG/SEC |DEG
    H 8H/SEC**2|   /)
     GO TO 410
1141 WRITE(NT,1401)
1401 FORMAT(1HO,
    A62H        |       SPRUNG MASS              |        WHEEL RIDE ACCE
    B14HL             | /
    C62H   TIME    | ANGULAR ACCEL. (DEG/SEC**2) |          (IN/SEC**2)
    D15H              | /
    E62H     SEC    |   DP/DT |   DQ/DT |   DR/DT |      RF |      LF |
    F15H  RR |      LR | / )
     GO TO 410
1142 WRITE(NT,1402)
1402 FORMAT(
    A62HO         |        SPRUNG MASS              |    FRONT ROLL CENTER
```

```
     B62H      |      REAR ROLL CENTER      | FR. AXLE ANGULAR|REAR AXLE A OUTP2030
     C 7HNGULAR|   /                                                     OUTP2040
     D62H   TIME    | ANGULAR ACCEL. (DEG/SEC**2) |   DEFL   |VELOCITY|  A OUTP2050
     E62HCCEL |  DEFL  |VELOCITY|  ACCEL |VELOCITY|  ACCEL |VELOCITY|     OUTP2060
     F 7HACCEL |   /                                                     OUTP2070
     G62H    SEC    |   DP/DT |   DQ/DT |   DR/DT | INCHES | IN/SEC | IN OUTP2080
     H62H/SEC2| INCHES | IN/SEC | IN/SEC2| DEG/SEC|DEG/SEC2| DEG/SEC|DE OUTP2090
     I 7HG/SEC2|   /   )                                                 OUTP2100
      GO TO 410                                                          OUTP2110
  115 WRITE(NT,1500)                                                     OUTP2120
 1500 FORMAT(1H0,8X,48H| STEER FRIC| STEER STOP|    STEER   |    STEER   OUTP2130
     A 1H|  /                                                            OUTP2140
     B59H  TIME    |   TORQUE   |   TORQUE   |     VEL    |    ACCEL   | /  OUTP2150
     C59H   SEC    |   LB-IN    |   LB-IN    |  DEG/SEC | DEG/SEC**2|  /) OUTP2160
      GO TO 410                                                          OUTP2170
  116 GO TO(1160,1161,1162),IS1                                          OUTP2180
 1160 WRITE(NT,1600)                                                     OUTP2190
 1600 FORMAT(                                                            OUTP2200
     A62H0 TIME    |    STEER ANGLE IN GROUND PLANE (DEG)          |   C OUTP2210
     B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) |  CAMBER ANGLE (DE OUTP2220
     C 6HG)  |   /                                                       OUTP2230
     D62H    SEC    |      RF    |      LF    |      RR    |      LR    |   OUTP2240
     E62H  RF    |      LF    |      RR    |      LR    |      RF    |      OUTP2250
     F 6HLF    |   /)                                                    OUTP2260
      GO TO 410                                                          OUTP2270
 1161 WRITE(NT,1601)                                                     OUTP2280
 1601 FORMAT(                                                            OUTP2290
     A62H0 TIME    |    STEER ANGLE IN GROUND PLANE (DEG)          |   C OUTP2300
     B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) |    CAMBER ANGLE ( OUTP2310
     C 9HDEG)    |   /                                                   OUTP2320
     D62H    SEC    |      RF    |      LF    |      RR    |      LR    |   OUTP2330
     E62H  RF    |      LF    |      RR    |      LR    |  RF |   LF |   R OUTP2340
     F 9HR |  LR |    /  )                                               OUTP2350
      GO TO 410                                                          OUTP2360
 1162 WRITE(NT,1602)                                                     OUTP2370
 1602 FORMAT(                                                            OUTP2380
     A62H0 TIME    |    STEER ANGLE IN GROUND PLANE (DEG)          |   C OUTP2390
     B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) | AXLE ROLL ANGLE ( OUTP2400
     C 6HDEG) |   /                                                      OUTP2410
     D62H    SEC    |      RF    |      LF    |      RR    |      LR    |   OUTP2420
     E62H  RF    |      LF    |      RR    |      LR    | FRONT-PHIF| REAR- OUTP2430
     F 6HPHIR |   / )                                                    OUTP2440
      GO TO 410                                                          OUTP2450
  117 WRITE(NT,1700)                                                     OUTP2460
 1700 FORMAT(                                                            OUTP2470
     A62H0         |      LONGITUDINAL WHEEL CENTER VELOCITY          |     OUTP2480
     B44H     LATERAL CONTACT POINT VELOCITY          |    /             OUTP2490
     C62H  TIME    |       PARALLEL TO GROUND PLANE (FT/SEC)        |     OUTP2500
     D44H    PARALLEL TO GROUND PLANE (FT/SEC)        |    /             OUTP2510
     E62H    SEC    |      RF    |      LF    |      RR    |      LR    |   OUTP2520
     F44H  RF    |      LF    |      RR    |      LR    |    /)          OUTP2530
```

```
      GO TO 410                                                          OUTP2540
  118 WRITE(NT,1800)                                                     OUTP2550
 1800 FORMAT(                                                            OUTP2560
     A58H0 TIME   |   TIRE CONTACT POINT ELEVATION (INCHES)        | /   OUTP2570
     B58H   SEC   |       RF    |       LF    |       RR    |      LR   | /)  OUTP2580
      GO TO 410                                                          OUTP2590
  119 WRITE(NT,1900)                                                     OUTP2600
 1900 FORMAT(                                                            OUTP2610
     A62H0 TIME   |          TOTAL SUSPENSION FORCE (LBS)            |   OUTP2620
     B44H      SUSPENSION ANTI-PITCH FORCE (LBS)       |    /            OUTP2630
     C62H   SEC   |       RF    |       LF    |       RR    |      LR   |   OUTP2640
     D44H  RF    |      LF    |      RR    |      LR    |    /)           OUTP2650
      GO TO 410                                                          OUTP2660
  120 WRITE(NT,2000)                                                     OUTP2670
 2000 FORMAT(                                                            OUTP2680
     A62H0 TIME   |        SUSPENSION DAMPING FORCE (LBS)            |   OUTP2690
     B44H       SUSPENSION SPRING FORCE (LBS)        |     /             OUTP2700
     C62H   SEC   |       RF    |       LF    |       RR    |      LR   |   OUTP2710
     D44H  RF    |      LF    |      RR    |      LR    |    /)           OUTP2720
      GO TO 410                                                          OUTP2730
  121 WRITE(NT,2100)                                                     OUTP2740
 2100 FORMAT(                                                            OUTP2750
     A62H0 TIME   |          RADIAL TIRE FORCES (LBS)                |   OUTP2760
     B44H       ROLLING RADIUS (INCHES)              |   /               OUTP2770
     C62H   SEC   |       RF    |       LF    |       RR    |      LR   |   OUTP2780
     D44H  RF    |      LF    |      RR    |      LR    |   /)            OUTP2790
      GO TO 410                                                          OUTP2800
  122 WRITE(NT,2200)                                                     OUTP2810
 2200 FORMAT(                                                            OUTP2820
     A62H0 TIME   |          TIRE NORMAL FORCE (LBS)            |        OUTP2830
     B62H TIRE SIDE FORCE (LBS)              |         SLIP ANGLE (DEG)  OUTP2840
     C 6H     |   /                                                      OUTP2850
     D62H  SEC    |       RF   |      LF   |      RR   |      LR   |     RF  OUTP2860
     E62H  |      LF   |       RR  |      LR   |   RF   |   LF   |   RR   |  OUTP2870
     F 6H LR  |   /)                                                     OUTP2880
      GO TO 410                                                          OUTP2890
  123 WRITE(NT,2300)                                                     OUTP2900
 2300 FORMAT(1H0,56X,25H|FRONT WHEEL| REAR WHEEL|  /                     OUTP2910
     A62H  TIME   |          TIRE TRACTIVE FORCE (LBS)               |   T  OUTP2920
     B20HORQUE  |   TORQUE   |  /                                        OUTP2930
     C62H   SEC   |       RF    |       LF    |       RR    |      LR   |   L  OUTP2940
     D20HB-FT   |   LB-FT    |  /)                                       OUTP2950
      GO TO 410                                                          OUTP2960
  124 WRITE(NT,2400)                                                     OUTP2970
 2400 FORMAT(                                                            OUTP2980
     A62H0 TIME   |   Z'-VERTICAL TIRE FORCE (LBS)         |   X'-HORIZO  OUTP2990
     B62HNTAL TIRE FORCE (LBS)       |   Y'-HORIZONTAL TIRE FORCE (LBS)  OUTP3000
     C 6H     |   /                                                      OUTP3010
     D62H   SEC   |    RF    |    LF    |    RR    |    LR    |    RF    |   OUTP3020
     E62H  LF    |    RR    |    LR    |    RF    |    LF    |    RR    |   OUTP3030
     F 6HLR    |    /)                                                   OUTP3040
```

```
      GO TO 410                                                         OUTP3050
  125 WRITE(NT,2500)                                                    OUTP3060
 2500 FORMAT(                                                           OUTP3070
     A62H0 TIME    |       TERRAIN ELEVATION (IN)          |   TERRAIN S OUTP3080
     B62HLOPE-CAMBER (PHIG) (DEG)    |   TERRAIN SLOPE-PITCH (THETAG) (D OUTP3090
     C 6HEG)  |    /                                                    OUTP3100
     D62H   SEC    |     RF     |     LF    |     RR    |     LR    |     RF    |  OUTP3110
     E62H  LF    |     RR    |    LR     |     RF    |     LF    |     RR    |    OUTP3120
     F 6HLR    |    /)                                                  OUTP3130
      GO TO 410                                                         OUTP3140
  126 WRITE(NT,2600)                                                    OUTP3150
 2600 FORMAT(                                                           OUTP3160
     A62H0 TIME    | SPRUNG MASS ACCELERATION LOCATION 1(G-UNITS)   | SPR OUTP3170
     B44HUNG MASS ACCELERATION LOCATION 2 (G-UNITS) |      /            OUTP3180
     C62H   SEC    |   LONG.    |    LAT.    |   VERT.    |   RESULT. |   L OUTP3190
     D44HONG.   |    LAT.    |    VERT.    |   RESULT. |      /)        OUTP3200
      GO TO 410                                                         OUTP3210
  127 WRITE(NT,2700)                                                    OUTP3220
 2700 FORMAT(                                                           OUTP3230
     A62H0          | INTERFACE |   VEHICLE |   NORMAL   |  FRICTION |  BA OUTP3240
     B44HRRIER  |     POSITION OF APPLIED LOAD          | /             OUTP3250
     C62H  TIME    |   AREA    |DEFORMATION|     FORCE   |    FORCE   |DEFL OUTP3260
     D44HECTION |     XR    |      YR    |      ZR    | /               OUTP3270
     E62H   SEC    |   IN**2   |   INCHES  |      LBS   |     LBS    |   1 OUTP3280
     F44HNCHES  |   INCHES  |   INCHES  |   INCHES   | )                OUTP3290
      GO TO 410                                                         OUTP3300
  128 WRITE(NT,2800)                                                    OUTP3310
 2800 FORMAT(1H0,8X,13H|VELOCITY OF|,59X,13H|SPRUNG MASS|  /            OUTP3320
     A62H          |  BARRIER  |      VELOCITY OF CONTACT POINT      |  OUTP3330
     B44H BARRIER ENERGY     | DISSIPATED| FRICTION   | /               OUTP3340
     C62H  TIME    | DEFLECTION|     UR'    |     VR'    |     WR'    | CON OUTP3350
     D44HSERVED | DISSIPATED|   ENERGY   |    WORK    | /               OUTP3360
     E62H   SEC    |    IN/SEC |    IN/SEC |    IN/SEC |    IN/SEC |    OUTP3370
     F44HFT-LB  |    FT-LB   |    FT-LB   |    FT-LB   | )              OUTP3380
      GO TO 410                                                         OUTP3390
  129 WRITE(NT,2900)                                                    OUTP3400
 2900 FORMAT(                                                           OUTP3410
     A62H0 TIME    |  HARD POINT DEFLECTION - INCHES      |     HARD POINT  OUTP3420
     B20HFORCE - LBS         |      /                                   OUTP3430
     C62H   SEC    |    NO. 1   |    NO. 2   |    NO. 3   |    NO. 1   |  OUTP3440
     D20HNO. 2  |    NO. 3   |    )                                     OUTP3450
  410 CONTINUE                                                          OUTP3460
  500 NT = 10                                                           OUTP3470
      DO 600 J=1,19                                                     OUTP3480
      IF(NPAGE(J).EQ.0) GO TO 600                                       OUTP3490
      GO TO(11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29),J OUTP3500
   11 NT = NT+1                                                         OUTP3510
      ACLON = (DU-VR+WQ)/G                                              OUTP3520
      ACLAT = (DV+UR-WP)/G                                              OUTP3530
      ACVER = (DW+VP-UQ)/G                                              OUTP3540
      ULON = U/12.                                                      OUTP3550
```

```
      VLAT = V/12.                                                      OUTP3560
      WVER = W/12.                                                      OUTP3570
      ACRES = SQRT(ACLON**2+ACLAT**2+ACVER**2)                          OUTP3580
      OXCP = XCP/12.                                                    OUTP3590
      OYCP = YCP/12.                                                    OUTP3600
      OZCP = ZCP/12.                                                    OUTP3610
      WRITE(NT,5000) T,OXCP,OYCP,OZCP,ULON,VLAT,WVER,ACLON,ACLAT,ACVER, OUTP3620
     *               ACRES                                              OUTP3630
 5000 FORMAT(' ',F7.4,10(2X,F10.2))                                     OUTP3640
      GO TO 600                                                         OUTP3650
   12 NT = NT+1                                                         OUTP3660
      ONU = 0.0                                                         OUTP3670
      IF(DYCP.EQ.0.0.AND.DXCP.EQ.0.0) GO TO 212                         OUTP3680
      ONU = ATAN2(DYCP,DXCP)/RAD                                        OUTP3690
  212 ROLL = P/RAD                                                      OUTP3700
      PITCH = Q/RAD                                                     OUTP3710
      YAW = R/RAD                                                       OUTP3720
      PHIO = PHIT/RAD                                                   OUTP3730
      THTAO = THETT/RAD                                                 OUTP3740
      PSIO = PSIT/RAD                                                   OUTP3750
      OBETA = ONU-PSIO                                                  OUTP3760
      PSIFO = PSI1/RAD                                                  OUTP3770
      IF(ISUS.EQ.1) GO TO 213                                           OUTP3780
      OPSIR = PSI3/RAD                                                  OUTP3790
      WRITE(NT,5000) T,ROLL,PITCH,YAW,PHIO,THTAO,PSIO,OBETA,ONU,PSIFO,  OUTP3800
     *               OPSIR                                              OUTP3810
      GO TO 600                                                         OUTP3820
  213 OPSI3 = PSI3/RAD                                                  OUTP3830
      OPSI4 = PSI4/RAD                                                  OUTP3840
      WRITE(NT,5004) T,ROLL,PITCH,YAW,PHIO,THTAO,PSIO,OBETA,ONU,PSIFO,  OUTP3850
     *               OPSI3,OPSI4                                        OUTP3860
      GO TO 600                                                         OUTP3870
   13 NT = NT+1                                                         OUTP3880
      GO TO(131,132,133),IS1                                            OUTP3890
  131 OETA3 = DEL3+TRO2*PHIR                                            OUTP3900
      OETA4 = DEL3-TRO2*PHIR                                            OUTP3910
      OETA3D = DEL3D+TRO2*PHIRD                                         OUTP3920
      OETA4D = DEL3D-TRO2*PHIRD                                         OUTP3930
      WRITE(NT,5000) T,DEL1,DEL2,OETA3,OETA4,DEL1D,DEL2D,OETA3D,OETA4D  OUTP3940
      GO TO 600                                                         OUTP3950
  132 WRITE(NT,5000) T,DEL1,DEL2,DEL3,DEL4,DEL1D,DEL2D,DEL3D,DEL4D      OUTP3960
      GO TO 600                                                         OUTP3970
 5004 FORMAT(1H ,F7.4,8(2X,F10.2),3(2X,F7.2) )                          OUTP3980
  133 OETA1 = DEL1+TFO2*PHIF                                            OUTP3990
      OETA2 = DEL1-TFO2*PHIF                                            OUTP4000
      OETA3 = DEL3+TRO2*PHIR                                            OUTP4010
      OETA4 = DEL3-TRO2*PHIR                                            OUTP4020
      OETA1D = DEL1D+TFO2*PHIFD                                         OUTP4030
      OETA2D = DEL1D-TFO2*PHIFD                                         OUTP4040
      OETA3D = DEL3D+TRO2*PHIRD                                         OUTP4050
      OETA4D = DEL3D-TRO2*PHIRD                                         OUTP4060
```

```
      WRITE(NT,5000) T,OETA1,OETA2,OETA3,OETA4,OETA1D,OETA2D,OETA3D,
     *                OETA4D
      GO TO 600
   14 NT = NT+1
      ODP = DP/RAD
      ODQ = DQ/RAD
      ODR = DR/RAD
      IF(ISUS.EQ.1) GO TO 141
      DPHDTO = PHIRD/RAD
      OPHDD = DPHIRD/RAD
      IF(ISUS.EQ.2) GO TO 142
      PHIRO = PHIR/RAD
      WRITE(NT,5001) T,ODP,ODQ,ODR,DDEL1D,DDEL2D,DEL3,DEL3D,DDEL3D,
     *                PHIRO,DPHDTO,OPHDD
 5001 FORMAT(' ',F7.4,3(2X,F10.2),2(2X,F9.1),2X,F7.2,2X,F9.1,2X,
     *       F9.1,2X,F7.2,2X,F9.1,2X,F9.1 )
      GO TO 600
  141 WRITE(NT,5005) T,ODP,ODQ,ODR,DDEL1D,DDEL2D,DDEL3D,DDEL4D
 5005 FORMAT(1H ,F7.3,3(2X,F8.2),10(2X,F7.1) )
      GO TO 600
  142 DPFDTO = PHIFD/RAD
      OPFDD = DPHIFD/RAD
      WRITE(NT,5005) T,ODP,ODQ,ODR,DEL1,DEL1D,DDEL1D,DEL3,DEL3D,DDEL3D,
     *                DPFDTO,OPFDD,DPHDTO,OPHDD
      GO TO 600
   15 NT = NT+1
      ODPSFI = DPSIFI/RAD
      ODDPSF = DDPSFI/RAD
      WRITE(NT,5000) T,T1PSI,T2PSI,ODPSFI,ODDPSF
      GO TO 600
   16 NT = NT+1
      PHRF = PHICI(1)/RAD
      PHLF = PHICI(2)/RAD
      PHRR = PHICI(3)/RAD
      PHLR = PHICI(4)/RAD
      PSRF = PSIIP(1)/RAD
      PSLF = PSIIP(2)/RAD
      PSRR = PSIIP(3)/RAD
      PSLR = PSIIP(4)/RAD
      IF(ISUS.EQ.2) GO TO 162
      PHI10 = PHI1/RAD
      PHI20 = PHI2/RAD
      IF(ISUS.EQ.1) GO TO 161
      WRITE(NT,5000) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHI10,
     *                PHI20
      GO TO 600
  161 PHI30 = PHI3/RAD
      PHI40 = PHI4/RAD
      WRITE(NT,5006) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHI10,
     *                PHI20,PHI30,PHI40
      GO TO 600
```

```
  162 PHIFO = PHIF/RAD                                                   OUTP4580
      PHIRO = PHIR/RAD                                                   OUTP4590
      WRITE(NT,5000) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHIFO,    OUTP4600
     *               PHIRO                                               OUTP4610
 5006 FORMAT(1H ,F7.4,8(2X,F10.2),4(1X,F6.2) )                           OUTP4620
      GO TO 600                                                          OUTP4630
   17 NT = NT+1                                                          OUTP4640
      VLNRF = UG(1)/12.                                                  OUTP4650
      VLNLF = UG(2)/12.                                                  OUTP4660
      VLNRR = UG(3)/12.                                                  OUTP4670
      VLNLR = UG(4)/12.                                                  OUTP4680
      VLTRF = VG(1)/12.                                                  OUTP4690
      VLTLF = VG(2)/12.                                                  OUTP4700
      VLTRR = VG(3)/12.                                                  OUTP4710
      VLTLR = VG(4)/12.                                                  OUTP4720
      WRITE(NT,5000) T,VLNRF,VLNLF,VLNRR,VLNLR,VLTRF,VLTLF,VLTRR,VLTLR   OUTP4730
      GO TO 600                                                          OUTP4740
   18 NT = NT+1                                                          OUTP4750
      WRITE(NT,5000) T,(ZGPP(I),I=1,4)                                   OUTP4760
      GO TO 600                                                          OUTP4770
   19 NT = NT+1                                                          OUTP4780
      WRITE(NT,5000) T,(SI(I),I=1,4),(APITCH(I),I=1,4)                   OUTP4790
      GO TO 600                                                          OUTP4800
   20 NT = NT+1                                                          OUTP4810
      IF(ISUS.EQ.2) GO TO 201                                            OUTP4820
      OD1 = -CF*DEL1D                                                    OUTP4830
      OD2 = -CF*DEL2D                                                    OUTP4840
      GO TO 202                                                          OUTP4850
  201 OD1 = -CF*(DEL1D+TSF02*PHIFD)                                      OUTP4860
      OD2 = -CF*(DEL1D-TSF02*PHIFD)                                      OUTP4870
  202 IF(ISUS.EQ.1) GO TO 203                                            OUTP4880
      OD3 = -CR*(DEL3D+TSO2*PHIRD)                                       OUTP4890
      OD4 = -CR*(DEL3D-TSO2*PHIRD)                                       OUTP4900
      GO TO 204                                                          OUTP4910
  203 OD3 = -CR*DEL3D                                                    OUTP4920
      OD4 = -CR*DEL4D                                                    OUTP4930
  204 CONTINUE                                                           OUTP4940
      OSP1 = -F2FI(1)                                                    OUTP4950
      OSP2 = -F2FI(2)                                                    OUTP4960
      OSP3 = -F2RI(1)                                                    OUTP4970
      OSP4 = -F2RI(2)                                                    OUTP4980
      WRITE(NT,5000) T,OD1,OD2,OD3,OD4,OSP1,OSP2,OSP3,OSP4               OUTP4990
      GO TO 600                                                          OUTP5000
   21 NT = NT+1                                                          OUTP5010
      WRITE(NT,5000) T,(FR(I),I=1,4),(HI(I),I=1,4)                       OUTP5020
      GO TO 600                                                          OUTP5030
   22 NT = NT+1                                                          OUTP5040
      DO 220 I=1,4                                                       OUTP5050
      ASTR(I) = BLNK                                                     OUTP5060
      IF(ABS(BETBR(I)).GT.3.0) ASTR(I)=STAR                              OUTP5070
      PSITEM = PSIIP(I)*SIGN(1.0,UG(I))                                  OUTP5080
```

```
      TERM = 0.0
      IF(UG(I).NE.0.0.OR.VG(I).NE.0.0) TERM = ATAN2(VG(I),ABS(UG(I)))
      SLPANG(I) = (TERM-PSITEM)/RAD
  220 CONTINUE
      WRITE(NT,5003) T,(FRCP(I),I=1,4),(FS(I),ASTR(I),I=1,4),
     *               (SLPANG(I),I=1,4)
 5003 FORMAT(1H ,F7.4,1X,4(1X,F10.2),4(1X,F9.2,A1),4(1X,F7.2)
      GO TO 600
   23 NT = NT+1
      TQFO = TI(1)*HI(1)/12.
      TQRO = TI(3)*HI(3)/12.
      WRITE(NT,5000) T,(FC(I),I=1,4),TQFO,TQRO
      GO TO 600
   24 NT = NT+1
      FR1O = AMTX(3,1)*FXU(1)+AMTX(3,2)*FYU(1)+AMTX(3,3)*FZU(1)
      FR2O = AMTX(3,1)*FXU(2)+AMTX(3,2)*FYU(2)+AMTX(3,3)*FZU(2)
      FR3O = AMTX(3,1)*FXU(3)+AMTX(3,2)*FYU(3)+AMTX(3,3)*FZU(3)
      FR4O = AMTX(3,1)*FXU(4)+AMTX(3,2)*FYU(4)+AMTX(3,3)*FZU(4)
      FXPU1 = AMTX(1,1)*FXU(1)+AMTX(1,2)*FYU(1)+AMTX(1,3)*FZU(1)
      FXPU2 = AMTX(1,1)*FXU(2)+AMTX(1,2)*FYU(2)+AMTX(1,3)*FZU(2)
      FXPU3 = AMTX(1,1)*FXU(3)+AMTX(1,2)*FYU(3)+AMTX(1,3)*FZU(3)
      FXPU4 = AMTX(1,1)*FXU(4)+AMTX(1,2)*FYU(4)+AMTX(1,3)*FZU(4)
      FYPU1 = AMTX(2,1)*FXU(1)+AMTX(2,2)*FYU(1)+AMTX(2,3)*FZU(1)
      FYPU2 = AMTX(2,1)*FXU(2)+AMTX(2,2)*FYU(2)+AMTX(2,3)*FZU(2)
      FYPU3 = AMTX(2,1)*FXU(3)+AMTX(2,2)*FYU(3)+AMTX(2,3)*FZU(3)
      FYPU4 = AMTX(2,1)*FXU(4)+AMTX(2,2)*FYU(4)+AMTX(2,3)*FZU(4)
      WRITE(NT,5002) T,FR1O,FR2O,FR3O,FR4O,FXPU1,FXPU2,FXPU3,FXPU4,
     *               FYPU1,FYPU2,FYPU3,FYPU4
 5002 FORMAT(' ',F7.4,12(2X,F8.1) )
      GO TO 600
   25 NT = NT+1
      PHG1O = PHGI(1)/RAD
      PHG2O = PHGI(2)/RAD
      PHG3O = PHGI(3)/RAD
      PHG4O = PHGI(4)/RAD
      THG1O = THGI(1)/RAD
      THG2O = THGI(2)/RAD
      THG3O = THGI(3)/RAD
      THG4O = THGI(4)/RAD
      WRITE(NT,5002) T,(ZPGI(I),I=1,4),PHG1O,PHG2O,PHG3O,PHG4O,THG1O,
     *               THG2O,THG3O,THG4O
      GO TO 600
   26 NT = NT+1
      AX1 = (DU-VR+WQ-X1*(Q2+R2)+Y1*(PQ-DR)+Z1*(PR+DQ))/G
      AX2 = (DU-VR+WQ-X2*(Q2+R2)+Y2*(PQ-DR)+Z2*(PR+DQ))/G
      AY1 = (DV+UR-WP+X1*(PQ+DR)-Y1*(P2+R2)+Z1*(QR-DP))/G
      AY2 = (DV+UR-WP+X2*(PQ+DR)-Y2*(P2+R2)+Z2*(QR-DP))/G
      AZ1 = (DW+VP-UQ+X1*(PR-DQ)+Y1*(QR+DP)-Z1*(P2+Q2))/G
      AZ2 = (DW+VP-UQ+X2*(PR-DQ)+Y2*(QR+DP)-Z2*(P2+Q2))/G
      A1R = SQRT(AX1**2+AY1**2+AZ1**2)
      A2R = SQRT(AX2**2+AY2**2+AZ2**2)
```

```
      WRITE(NT,5000) T,AX1,AY1,AZ1,A1R,AX2,AY2,AZ2,A2R
      GO TO600
   27 NT = NT+1
      WRITE(NT,5000) T,AINTI,VDEF,FN,FRICT,DELBB,SXR,SYR,SZR
      GO TO 600
   28 NT = NT+1
      SWORKO = SWORK/12.
      EEEO = EEE/12.
      DISSO = DISS/12.
      SPENGO = SPENGY/12.
      WRITE(NT,5000) T,VMAX(1),URP,VRP,WRP,EEEO,DISSO,SPENGO,SWORKO
      GO TO 600
   29 NT = NT+1
      DO 291 I=1,3
      HDEF(I) = 0.0
      IF(FNSTI(I).NE.0.0) HDEF(I) = YSTIPO(I)-YBP
  291 CONTINUE
      WRITE(NT,5000) T,(HDEF(I),I=1,3),(FNSTI(I),I=1,3)
  600 CONTINUE
      RETURN
      ENTRY THPLOT(IPLT)
      GO TO(901,902,903),IPLT
  901 WRITE(3) (HED(I),I=1,18),DADE(1),DADE(2),(VHED(I),I=1,10),
     A     (THED(I),I=1,10),(CHED(I),I=1,10),(GHED(I),I=1,10),
     B      (SHED(I),I=1,10)
      RETURN
  902 WRITE(3) T,ULON,VLAT,ACLON,ACLAT,ACVER,ACRES,ROLL,PITCH,
     1       YAW,PHIO,THTAO,PSIO,AX1,AY1,AZ1,A1R,AX2,AY2,AZ2,A2R
      RETURN
  903 WRITE(3) (TTTTTT,I=1,21)
      RETURN
      END
```

```
      SUBROUTINE PINT1(IN,MODE,N,/X/,/HH/,YY,YYP,A)
C***********************************************************************
C*                                                                     *
C*    SUBROUTINE PINT1                                                 *
C*                                                                     *
C*    PURPOSE                                                          *
C*      TO SOLVE A SYSTEM OF N REAL ORDINARY DIFFERENTIAL EQUATIONS OF *
C*      THE FIRST ORDER                                                *
C*                                                                     *
C*    USAGE                                                            *
C*      CALL PINT1(IN,MODE,N,X,HH,YY,YYP,A)                            *
C*                                                                     *
C*    DESCRIPTION OF PARAMETERS                                        *
C*      N      NUMBER OF EQUATIONS                                     *
C*      IN     INDICATOR FOR INITIALIZATION OF INTEGRATION STEP , IF   *
C*             IN = 1 THE ROUTINE INITIALIZES                          *
C*             IN = 2 THE ROUTINE INTEGRATES ONE STEP                  *
C*      MODE   THE OPTION WORD(=0,1,OR 2) FOR USING ONE OF THE THREE MOD*
C*             ES OF INTEGRATION.   IF MODE EQUALS                     *
C*                0 - THE ADAMS-MOULTON VARIABLE STEP-SIZE IS USED,    *
C*                1 - THE RUNGE-KUTTA FIXED STEP-SIZE IS USED,         *
C*                2 - THE ADAMS FIXED STEP-SIZE IS USED                *
C*      A      IS AN ARRAY OF DIMENSION SIX CONTAINING THE PARAMETERS  *
C*                FOR THE VARIABLE MODE                                *
C*      X      THE SOURCE VARIABLE                                     *
C*      HH     THE INCREMENT IN SOURCE VARIABLE OR THE STEP SIZE       *
C*      YY     THE TARGET VARIABLES UPDATED BY THIS ROUTINE            *
C*      YYP    THE ARRAY OF FIRST DERIVATIVES OF THE TARGET VARIABLES  *
C*              COMPUTED  IN  THE SUBROUTINE DAUX                      *
 1000 CONTINUE
C*   METHOD                                                            *
C*      THE ROUTINE USES THE E.K.BLUM MODIFICATION OF THE RUNGE-KUTTA  *
C*      FOURTH-ORDER METHOD,THE FOURTH ORDER ( FIXED AND VARIABLE )    *
C*      ADAMS-MOULTON PREDICTOR -CORRECTOR METHOD.                     *
C*                                                                     *
C*    REMARKS                                                          *
C*    BEFORE EXECUTING THE FIRST PINT1 CALL, THE USER MUST INITIALIZE  *
C*    X,HH, AND EACH OF THE TARGET VARIABLE.                           *
C*                                                                     *
C*      THE SECOND ENTRY POINT ( IN=2 ) MAY BE USED ANY NUMBER OF TIMES*
C*    AFTER THE FIRST PINT1 CALL (IN=1) TO INTEGRATE ONE STEP-SIZE.    *
C*                                                                     *
C*    SUBROUTINES REQUIRED                                             *
C*        ( ERRMSG ) NOT USED, SEE CARD SERIAL NUMBER05302840          *
C*      THE USER MUST PROVIDE A SUBROUTINE NAMED 'DAUX' WHICH EVALUATES*
C*      THE N DERIVATIVES OF THE SYSTEM OF N FIRST  ORDER DIFFERENIAL  *
C*      EQUATION                                                       *
C*                                                                     *
C*    AUTHOR                                                           *
C*       SQUARE PARTEE                                                 *
C*       AUGUST 1966                                                   *
C*       CORNELL AERONAUTICAL LAB.                                     *
```

```
C*
C**********************************************************************
      DIMENSION YY(1),YYP(1),A(1)
      DIMENSION Y(30),YNO(30),YN1(30),YN2(30),YN3(30),YPNO(30),YN(30),
     *          YPN(30),YPN1(30),YPN2(30),YPN3(30),P(30),Q(30)
      DOUBLE PRECISION H,DY,Y,YNO,YN,YN1,YN2,YN3,YPNO,YPN,YPN1,YPN2,
     *                 YPN3,DABS,P,Q
      EQUIVALENCE (YPNO(1),P(1))
      EQUIVALENCE (YNO(1),Q(1))
C
C     MODE=0     VARIABLE ADAMS MOULTON METHOD
C     MODE= 1     FIXED RUNGE-KUTTA
C     MODE= 2     FIXED ADAMS MOULTON METHOD
C
    1 INN = IN
      GO TO (2,50),INN
    2 NMODE = MODE + 1
      NDO = 1
      NGO = 1
      NSS = 1
    3 GO TO (8,50,5),NMODE
    5 NGO = 2
      GO TO 50
C
C     SET UP VARIABLE MODE PARAMETERS
C
    8 NGO = 3
      EMAX = ABS(A(1))
      IF (EMAX.EQ.0.0) EMAX = .1E-03
      EMIN = EMAX * .01
      IF (A(2).NE.0.0) EMIN = EMAX/ABS(A(2))
      AA = ABS(A(3))
      IF (AA.EQ.0.0) AA = 1.0
      HMAX = ABS(A(4))
      IF(HMAX.EQ.0.0) HMAX = 10.E+03
      HMIN = ABS(A(5))
      IF (HMIN.EQ.0.0)  HMIN = .1E-06
      BETA = ABS(A(6))
      IF (BETA.GE.1.0 .OR. BETA.LE.0.0) BETA = .5
      NMSG = 0
C
   50 GO  TO ( 100, 111, 200, 300    ) , NDO
C
C         FIXED RUNGE - KUTTA       INITIALIZATION
  100 DO 102  I=1,N
      Q(I) = 0.0
      Y(I) = YY(I)
  102 CONTINUE
      NSTEP = 0
  103 CALL  DAUX
  106 XOO = X
      H = HH
      IF (NGO.EQ.1) GO TO 110
```

```
  108 XN3 = X
      DO 109  I=1,N
      YPN3(I) = YYP(I)
      YN3(I) = YY(I)
  109 CONTINUE
  110 NDO = 2
      IF (   NSS .EQ. 1 )   RETURN
C
C                ONE  POINT  INTEGRATE
C
  111 XOO = X
      H = HH
      DO 112  I=1,N
      DY = YYP(I)
      P(I) = H*DY
      Y(I) = Y(I)+.5D0*P(I)
      Q(I) = P(I)
        YY(I)  = Y(I)
  112 CONTINUE
      X = XOO +  .5 * HH
      CALL  DAUX
  113 DO 115  I=1,N
      DY = YYP(I)
      P(I) = H*DY
      Y(I) = Y(I)+.5D0*P(I)-.5D0*Q(I)
      Q(I) = Q(I)/6.D0
        YY(I)  = Y(I)
  115 CONTINUE
  116 CALL  DAUX
  117 DO 120  I=1,N
      DY = YYP(I)
      P(I) = H*DY-.5D0*P(I)
      Y(I) = Y(I)+P(I)
      Q(I) = Q(I)-P(I)
        YY(I)  = Y(I)
  120 CONTINUE
      X = XOO  +  HH
      CALL  DAUX
  121 DO 125  I=1,N
      DY = YYP(I)
      P(I) = H*DY+2.0*P(I)
      Y(I) = Y(I) + Q(I)+P(I)/6.0D0
        YY(I)  = Y(I)
  125 CONTINUE
      CALL  DAUX
C
C              END OF FIXED STEP RUNGE - KUTTA
C
C
C
  130 IF  ( NGO   .EQ. 1   )     RETURN
  135 NSTEP = NSTEP + 1
      GO TO (136,140,145), NSTEP
```

```
C
C     SET UP THREE POINTS FOR ADAMS-MOULTON'S
C
  136 XN2 = X
      DO 138  I=1,N
      YPN2(I)  =  YYP(I)
  138 YN2(I) = Y(I)
      RETURN
  140 XN1 = X
      DO 142  I=1,N
      YPN1(I)  =  YYP(I)
      YN1(I) = Y(I)
  142 CONTINUE
      RETURN
  145 XN = X
      DO 146  I=1,N
      YN(I) = Y(I)
      YPN (I)  =  YYP(I)
  146 CONTINUE
      NSTEP = 0
      NFIRST = 1
      NCRE = 0
      NDO = NGO + 1
      RETURN
C
C
C      FIXED ADAMS MOULTON PREDICTOR METHOD
C
  200 XOO = X
      H = HH
      X = XOO  +  HH
      DO 220  I=1,N
      Y(I) = YN(I)+H*(55.DO*YPN(I)-59.DO*YPN1(I)+37.DO*YPN2(I)-9.0DO*
     4   YPN3(I)) / 24.DO
        YY(I)  = Y(I)
  220 CONTINUE
      CALL  DAUX
      DO 225  I=1,N
      DY = YYP(I)
      Y(I) = YN(I)+H*(9.DO*DY   +19.DO*YPN(I)-5.DO*YPN1(I)+YPN2(I))
     5    / 24.DO
        YY(I)  = Y(I)
  225 CONTINUE
      CALL  DAUX
      DO 250  I=1,N
C     SAVE VALUES
      YPN3(I) = YPN2(I)
      YPN2(I) = YPN1(I)
      YPN1(I) = YPN(I)
      YPN(I)    =   YYP(I)
      YN3(I) = YN2(I)
      YN2(I) = YN1(I)
      YN1(I) = YN(I)
```

```
      YN(I) = Y(I)
  250 CONTINUE
  251 XN3 = XN2
      XN2 = XN1
      XN1 = XN
      XN = X
      RETURN
C
C     VARIABLE ADAMS MOULTON METHOD
C
C
  300 XOO = X
      H = HH
      X = XOO  +  HH
      DO 364  I=1,N
       Y(I)= YN(I)+H*(55.DO*YPN(I)-59.DO*YPN1(I)+37.DO*YPN2(I)-9.DO*
     6        YPN3(I)) / 24.DO
        YY(I)  = Y(I)
      P(I) = Y(I)
  364 CONTINUE
      CALL  DAUX
      DO 365  I=1,N
      DY = YYP(I)
      Y(I) = YN(I)+H*(9.DO*DY   +19.DO*YPN(I)-5.DO*YPN1(I)+YPN2(I))
     7         / 24.DO
        YY(I)  = Y(I)
  365 CONTINUE
      CALL  DAUX
C
C     END VARIABLE ADAM MOULTON
C
      ERROR = 0.0
      DO 370  I=1,N
      PRED = SNGL(P(I))
C
C     SAVE VALUES
  366 YPNO(I) = YPN3(I)
      YPN3(I) = YPN2(I)
      YPN2(I) = YPN1(I)
      YPN1(I) = YPN(I)
      YPN(I)  = YYP(I)
      YNO(I) = YN3(I)
      YN3(I) = YN2(I)
      YN2(I) = YN1(I)
      YN1(I) = YN(I)
      YN(I) = Y(I)
      DD = AMAX1(ABS(SNGL(Y(I))),AA)
      DERR = ABS(PRED-SNGL(Y(I)))/(14.0*DD)
      ERROR = AMAX1(ERROR,DERR)
  370 CONTINUE
  375 XNO = XN3
      XN3 = XN2
      XN2 = XN1
```

```
      XN1 = XN
      XN = X
C     ERROR TESTS     ADAMS MOULTON
C
  305 IF (ERROR.GT.EMAX) GO TO 315
      NFIRST = 2
      IF (ERROR.LT.EMIN) GO TO 330
  306 NFIRST = 2
      NCRE = 0
      RETURN
C
C     REDUCE STEP SIZE
C
  315 NSS=2
      NCRE = 0
  316 HH = HH*BETA
      IF( ABS(HH) .GT.  HMIN  )  GO  TO  319
      HH = SIGN( HMIN, HH  )
      IF (NMSG.NE.0) GO TO 306
C
C     CALL ERRMSG(10  ,39H  MINIMUM STEP SIZE  IN  PINT1
      WRITE(6,317)
  317 FORMAT(28HO MINIMUM STEP SIZE IN PINT1)
C
      NMSG = 1
      A(1) =-A(1)
C
  319 GO TO (     320,325), NFIRST
C     ERROR FIRST VARIABLE POINT
  320 X = XNO
      DO 321  I=1,N
      YY(I) = YNO(I)
  321 CONTINUE
      GO TO   100
C     ERROR DURING VARIABLE MODE
  325 X = XN1
      DO 327  I=1,N
      YY(I) = YN1(I)
  327 CONTINUE
      GO TO   100
C
C     INCREASE STEP SIZE HERE
C
  330 NSS = 1
      NCRE = NCRE + 1
      IF (NCRE.LE.2) RETURN
C     NOW INCREASE
  335 NCRE = 0
      HH = SIGN(AMIN1(ABS(HH/BETA),HMAX),HH)
      GO TO 106
      END
```

```
      SUBROUTINE PLOTTP(IPLT)                                           PLOT0010
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    PLOT0020
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     PLOT0030
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     PLOT0040
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,PLOT0050
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, PLOT0060
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    PLOT0070
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  PLOT0080
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  PLOT0090
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   PLOT0100
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 PLOT0110
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),PLOT0120
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN PLOT0130
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             PLOT0140
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))PLOT0150
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),PLOT0160
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       PLOT0170
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      PLOT0180
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        PLOT0190
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          PLOT0200
     6            (PSIFID,VAR(22))                                      PLOT0210
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      PLOT0220
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))PLOT0230
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),      PLOT0240
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    PLOT0250
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   PLOT0260
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         PLOT0270
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     PLOT0280
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         PLOT0290
     1            (DER(10),DPHIFD)                                      PLOT0300
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       PLOT0310
     1            (DER(14),DDEL4D)                                      PLOT0320
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,PLOT0330
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   PLOT0340
     2            CGYW(4),ZPG1(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),PLOT0350
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     PLOT0360
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  PLOT0370
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  PLOT0380
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     PLOT0390
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      PLOT0400
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),PLOT0410
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)PLOT0420
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      PLOT0430
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     PLOT0440
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   PLOT0450
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  PLOT0460
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       PLOT0470
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   PLOT0480
     1            (PSII(1),PSI1)                                        PLOT0490
```

```
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1             GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2              TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3              BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4              XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5              ZRO3,ZRO3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6             ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7              SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8              SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9              ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1              ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2              TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3              HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4             ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1).(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1                DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2                PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3                SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1                A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2                A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
      DIMENSION ICONTW(4)
      DATA TTTTTT/-9999.0/
    1 GO TO (2,3,4),IPLT
    2 WRITE(1)HED,DADE,A,B,TS,ZR,RHO,ZF,RW(1),TF,TR
      RETURN
    3 DO 6 J=1,4
      IF(FRCP(J).GT.0.01) GO TO 5
      ICONTW(J) = 0
      GO TO 6
5     ICONTW(J) = 1
      IF(ABS(BETBR(J)).GT.3.0) ICONTW(J)= -1
6     CONTINUE
      WRITE(1) T,XCP,YCP,ZCP,PHIT,THETT,PSIT,DEL1,DEL2,DEL3,PHIR,PSI1,
     1 PHI1,PHI2,(XGPP(J),YGPP(J),ZGPP(J),J=1,4),ICONTW
      RETURN
    4 WRITE(1) (TTTTTT,I=1,30)
      RETURN
      END
```

```
      SUBROUTINE POLY(C0,C,X,Y,C1)
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
C       SINGLE VEHICLE ACCIDENT SIMULATION - SUBROUTINE POLY
C     SUBROUTINE TO FIND ROOT OF FIFTH DEGREE POLYNOMIAL USING NEWTON-
C           RAPHSON METHOD
      DIMENSION C(5)
    1 C2 = C0-C1
      Z = X-Y
      KK = 0
    2 KK = KK+1
      IF(KK.GT.100)GO TO 8
      P = C2
      PP = 0.0
      XX = 1.0
      YY = Z
    3 DO 5 I=1,5
      P = P+C(I)*YY
      PP = PP+C(I)*XX*I
      XX = YY
      YY = YY*Z
    5 CONTINUE
    7 H1 = P/PP
      Z = Z-H1
      IF(ABS(H1/Z).GT.1.0E-6)GO TO 2
    8 Y = X-Z
      RETURN
      END
```

```
      SUBROUTINE RESFRC
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DELI,VAR(7)),(DELID,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDELI,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1           (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
```

```
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   RESF0500
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  RESF0510
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,RESF0520
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2RESF0530
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,RESF0540
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          RESF0550
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                RESF0560
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)      RESF0570
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),   RESF0580
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,  RESF0590
     2               DELTB,XINPT(100)                                  RESF0600
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),  RESF0610
     1              YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),   RESF0620
     2              AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,  RESF0630
     3              CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),     RESF0640
     4              YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT RESF0650
     5              ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4, RESF0660
     6              XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,   RESF0670
     7              AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,RESF0680
     8              FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,RESF0690
     9              NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,     RESF0700
     A              RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DLLBBP,        RESF0710
     B              SWORK,SPENGY,DISS,IPLN,ILOAD                       RESF0720
      DIMENSION INDXPT(4)                                              RESF0730
      EQUIVALENCE (INDXPT(1),I1)                                       RESF0740
      COMMON/BARSTR/ XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),       RESF0750
     1               ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),     RESF0760
     2               FNSTI(3),AKST(3)                                  RESF0770
      COMMON /HARDPT/ FRICF(4),UPT(4),VPT(4),WPT(4)                    RESF0780
      DIMENSION X(4),Y(4),Z(4),F(4)                                    RESF0790
      SFXS = 0.0                                                       RESF0800
      SFYS = 0.0                                                       RESF0810
      SFZS = 0.0                                                       RESF0820
      SNPS = 0.0                                                       RESF0830
      SNTS = 0.0                                                       RESF0840
      SNPSS= 0.0                                                       RESF0850
      X(1) = SXR/SDEN                                                  RESF0860
      Y(1) = SYR/SDEN                                                  RESF0870
      Z(1) = SZR/SDEN                                                  RESF0880
      SXR = X(1)                                                       RESF0890
      SYR = Y(1)                                                       RESF0900
      SZR = Z(1)                                                       RESF0910
      F(1) = FN                                                        RESF0920
      DO 4 J=1,3                                                       RESF0930
      K = J+1                                                          RESF0940
      X(K) = XSTI(J)                                                   RESF0950
      Y(K) = YSTI(J)                                                   RESF0960
      Z(K) = ZSTI(J)                                                   RESF0970
    4 F(K) = FNSTI(J)                                                  RESF0980
      DO 5 J=1,4                                                       RESF0990
      FRICF(J) = 0.0                                                   RESF1000
```

```
      TEMP1 = U-R*Y(J)+Q*Z(J)
      TEMP2 = V+R*X(J)-P*Z(J)
      TEMP3 = W+P*Y(J)-Q*X(J)
      UPT(J) = AMTX(1,1)*TEMP1+AMTX(1,2)*TEMP2+AMTX(1,3)*TEMP3
      VPT(J) = AMTX(2,1)*TEMP1+AMTX(2,2)*TEMP2+AMTX(2,3)*TEMP3
      WPT(J) = AMTX(3,1)*TEMP1+AMTX(3,2)*TEMP2+AMTX(3,3)*TEMP3
      TMPV = SQRT(UPT(J)**2+WPT(J)**2)
      IF(J.EQ.1) VTAN = TMPV
      TEMP1 = 0.0
      TEMP2 = 0.0
      IF(TMPV.LT.EPSV) GO TO 6
      AA = AMUB*F(J)
      FRICF(J) = AA
      AA = AA/TMPV
      TEMP1 = -AA*UPT(J)
      TEMP2 = -AA*WPT(J)
    6 CONTINUE
      FX = AMTX(1,1)*TEMP1-AMTX(2,1)*F(J)+AMTX(3,1)*TEMP2
      FY = AMTX(1,2)*TEMP1-AMTX(2,2)*F(J)+AMTX(3,2)*TEMP2
      FZ = AMTX(1,3)*TEMP1-AMTX(2,3)*F(J)+AMTX(3,3)*TEMP2
      TEMP1 = 0.
      TEMP2 = 0.
      SFXS = SFXS+FX
      SFYS = SFYS+FY
      SFZS = SFZS+FZ
      SNPS = SNPS+FZ*Y(J)-FY*Z(J)
      SNTS = SNTS+FX*Z(J)-FZ*X(J)
    5 SNPSS= SNPSS+FY*X(J)-FX*Y(J)
      FRICT = FRICF(1)
      URP = UPT(1)
      VRP = VPT(1)
      WRP = WPT(1)
      RETURN
      END
```

```
      SUBROUTINE RUFFRC(I,ZGM)                                          RUFF0010
C          HVOSM-RD2 VERSION                                            RUFF0020
C          HVOSM-VD2 VERSION                                            RUFF0030
C              REVISED OCTOBER 1975  CALSPAN CORPORATION                RUFF0040
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          RUFF0050
     1           CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,    RUFF0060
     2           PSIFIO,PSIFDO                                          RUFF0070
      DIMENSION YCIP(2)                                                 RUFF0080
      EQUIVALENCE (YCIP(1),YC1P)                                        RUFF0090
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,RUFF0100
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),    RUFF0110
     2           CGYW(4),ZPG1(4),THG1(4),PHGI(4),CPG(4),SPG(4),CTG(4),RUFF0120
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),      RUFF0130
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),   RUFF0140
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),   RUFF0150
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),      RUFF0160
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),       RUFF0170
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),RUFF0180
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)RUFF0190
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      RUFF0200
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),      RUFF0210
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),    RUFF0220
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                   RUFF0230
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       RUFF0240
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   RUFF0250
     1           (PSII(1),PSI1)                                         RUFF0260
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,RUFF0270
     1          GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,           RUFF0280
     2           TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,   RUFF0290
     3           BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,     RUFF0300
     4           XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,     RUFF0310
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPRUFF0320
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,      RUFF0330
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,     RUFF0340
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,   RUFF0350
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ    RUFF0360
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  RUFF0370
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,   RUFF0380
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,RUFF0390
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2RUFF0400
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,RUFF0410
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL            RUFF0420
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 RUFF0430
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       RUFF0440
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     RUFF0450
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    RUFF0460
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    RUFF0470
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        RUFF0480
      LOGICAL LCB1,LCB2                                                 RUFF0490
```

```
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF
      DIMENSION ZGM(2205)
      DIMENSION FJPP(35)
      DO 20 N=1,35
   20 FJPP(N) = FJP(N,I)
      SNPSI = SIN(PSII(I))
      CSPSI = COS(PSII(I))
      SNPHI = SIN(PHII(I))
      CSPHI = COS(PHII(I))
      SFRX(I) = 0.0
      SFRY(I) = 0.0
      SFRZ(I) = 0.0
      TTAJ21 = CSPHI*SNPSI
      TTAJ31 = SNPHI*SNPSI
      AJMTX(1,2) = -SNPSI
      AJMTX(2,2) = CSPHI*CSPSI
      AJMTX(3,2) = SNPHI*CSPSI
      INDF = 0
      INDL = 0
      MF = IFIX((XP(I)-RW(I))/DELG)
      ML = MF+IFIX(2.0*RW(I)/DELG)
      IF(MF.GE.1) GO TO 10
      MF = 1
      INDF = 1
   10 IF(ML.LE.NEND) GO TO 11
      ML = NEND
      INDL = 1
   11 DO 100 J=1,21
      THTJ = (-44.0+4.0*J)*RAD
      STJ = SIN(THTJ)
      CTJ = COS(THTJ)
      AJMTX(1,1) = CTJ*CSPSI
      AJMTX(2,1) = TTAJ21*CTJ+SNPHI*STJ
      AJMTX(3,1) = TTAJ31*CTJ-CSPHI*STJ
      AJMTX(1,3) = CSPHI*STJ
      AJMTX(2,3) = TTAJ21*STJ-SNPHI*CTJ
      AJMTX(3,3) = TTAJ31*STJ+CSPHI*CTJ
      DO 8 K=1,3
      DO 7 L=1,3
      BMTX(K,L) = 0.0
      DO 6 M=1,3
    6 BMTX(K,L) = BMTX(K,L)+AMTX(K,M)*AJMTX(M,L)
    7 CONTINUE
    8 CONTINUE
      IF(BMTX(3,3).EQ.0.0) GO TO 100
      DO 50 M=MF,ML
      ZM1 = 0.0
      IF(M.LT.NEND) ZM1 = ZGM(M+1)
```

```
      XM = DELG*(M-1)                                                   RUFF1010
      XM1 = XM+DELG                                                     RUFF1020
      TMP = (BMTX(1,3)*(ZM1-ZGM(M))/(BMTX(3,3)*DELG))                   RUFF1030
      IF(TMP.EQ.1) GO TO 50                                             RUFF1040
      XD = XM-XP(I)                                                     RUFF1050
      XD1 = XM1-XP(I)                                                   RUFF1060
      IF(ABS(XD).LT.0.001.OR.ABS(XD1).LT.0.001) GO TO 49                RUFF1070
      SM = (ZGM(M)-ZP(I))/XD                                            RUFF1080
      SM1 = (ZM1-ZP(I))/XD1                                             RUFF1090
      SGM = SIGN(1.0,SM)                                                RUFF1100
      SGM1 = SIGN(1.0,SM1)                                              RUFF1110
      IF(SGM.GT.0.0.AND.SGM1.LT.0.0) GO TO 50                           RUFF1120
      IF(SGM.EQ.SGM1.AND.SM1.GT.SM) GO TO 50                            RUFF1130
   49 TMP1 = 1.0/(1.0-TMP)                                              RUFF1140
      XJP = TMP1*(XP(I)+BMTX(1,3)*(ZGM(M)-ZP(I)-XM*                     RUFF1150
     1      (ZM1-ZGM(M))/DELG)/BMTX(3,3))                               RUFF1160
      IF(XJP.GE.XM) GO TO 60                                            RUFF1170
      IF(XJP.GE.0.0) GO TO 100                                          RUFF1180
      GO TO 69                                                          RUFF1190
   60 IF(XJP.LE.XM1) GO TO 70                                           RUFF1200
   50 CONTINUE                                                          RUFF1210
      GO TO 69                                                          RUFF1220
   70 IF(ABS(BMTX(1,3)).LT.0.0001) GO TO 71                             RUFF1230
      HJ = (XJP-XP(I))/BMTX(1,3)                                        RUFF1240
      ZJP = ZP(I)+BMTX(3,3)*HJ                                          RUFF1250
      GO TO 72                                                          RUFF1260
   71 ZJP = ZGM(M)+(XJP-XM)*(ZM1-ZGM(M))/DELG                           RUFF1270
      HJ = (ZJP-ZP(I))/BMTX(3,3)                                        RUFF1280
      XJP = XP(I)                                                       RUFF1290
      GO TO 72                                                          RUFF1300
   69 ZJP = 0.0                                                         RUFF1310
      XJP = XP(I)-BMTX(1,3)*ZP(I)/BMTX(3,3)                             RUFF1320
      HJ = -ZP(I)/BMTX(3,3)                                             RUFF1330
   72 YJP = YP(I)+BMTX(2,3)*HJ                                          RUFF1340
      IF(HJ.LT.0.0.OR.HJ.GT.RW(I)) GO TO 100                            RUFF1350
      CAJ = (XP(I)-XJP)/HJ                                              RUFF1360
      CBJ = (YP(I)-YJP)/HJ                                              RUFF1370
      CGJ = (ZP(I)-ZJP)/HJ                                              RUFF1380
      CALL INTRPL(FJPP,RWHJB,RWHJE,DRWHJ,RW(I)-HJ,FJ)                   RUFF1390
      SFRX(I) = SFRX(I)+FJ*CAJ                                          RUFF1400
      SFRY(I) = SFRY(I)+FJ*CBJ                                          RUFF1410
      SFRZ(I) = SFRZ(I)+FJ*CGJ                                          RUFF1420
  100 CONTINUE                                                          RUFF1430
      FR(I) = SQRT(SFRX(I)**2+SFRY(I)**2+SFRZ(I)**2)                    RUFF1440
      IF(FR(I).NE.0.0) GO TO 110                                        RUFF1450
      CAR(I) = 0.0                                                      RUFF1460
      CBR(I) = 0.0                                                      RUFF1470
      CGR(I) = 1.0                                                          1480
      HI(I) = RW(I)                                                         1490
      PHGI(I) = 0.0                                                         1500
      THGI(I) = 0.0                                                         1510
```

```
      SPG(I) = 0.0
      TXGP = XP(I)
      GO TO 112
  110 CAR(I) = -SFRX(I)/FR(I)
      CBR(I) = -SFRY(I)/FR(I)
      CGR(I) = -SFRZ(I)/FR(I)
      HI(I) = RW(I)-FR(I)/AKT(I)
      IF(HI(I).GT.RW(I)-SIGT(I)) GO TO 111
      HI(I) = RW(I)-(FR(I)/AKT(I)+SIGT(I)*(XLAMT(I)-1.0))/XLAMT(I)
  111 TXGP = XP(I)+HI(I)*CAR(I)
      ME = TXGP/DELG+1
      TPHGI = 0.0
      TTHGI = ATAN2((ZGM(ME)-ZGM(ME+1)),DELG)
      TAI = CBR(I)*CGYW(I)-CGR(I)*CBYW(I)
      TBI = CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
      TCI = CAR(I)*CBYW(I)-CBR(I)*CAYW(I)
      STI = SIN(TTHG1)
      CTI = COS(TTHGI)
      DN1 = (TCI*TCI+TBI*TBI)*STI-TAI*TCI*CTI
      DN2 = -TBI*(TAI*STI+TCI*CTI)
      DN3 = (TAI*TAI+TBI*TBI)*CTI-TAI*TCI*STI
      TERM5 = SQRT(DN1*DN1+DN2*DN2+DN3*DN3)
      SPG(I) = -DN2/TERM5
      PHGI(I) = ARSIN(SPG(I))
      THGI(I) = ATAN(DN1/DN3)
  112 CPG(I) = COS(PHGI(I))
      CTG(I) = COS(THGI(I))
      STG(I) = SIN(THGI(I))
      XGPP(I) = TXGP
      YGPP(I) = YP(I)+HI(I)*CBR(I)
      ZGPP(I) = ZP(I)+HI(I)*CGR(I)
      RETURN
      END
```

```
      SUBROUTINE RUFRED(NEND,DELG,DGMAX,ZRTAB)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
C             HVOSM-RD2 VERSION
C          HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975  CALSPAN CORPORATION
      DIMENSION ZRTAB(2205)
      IF(NEND.GT.2200) GO TO 900
      READ(4,END=901) (ZRTAB(I),I=1,NEND)
      GO TO 12
  901 WRITE(6,9001)
 9001 FORMAT('     END OF FILE ENCOUNTERED IN READ OF ROUGHNESS '/
     1  '  DATA BEFORE NEND POINTS WERE READ.'  )
      NEND = I
   12 DGMAX = (NEND-1)*DELG
      RETURN
  900 WRITE(6,9000)
 9000 FORMAT('     NUMBER OF LAST ROUGHNESS DATA POINT IS GREATER '/
     1  ' THAN THE ALLOWED 2200.  PROGRAM TERMINATED.'  )
      STOP
      END
```

```
      SUBROUTINE SFORCE                                                 SFOR0010
C             HVOSM-RD2 VERSION                                         SFOR0020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                SFOR0030
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             SFOR0040
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))SFOR0050
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),SFOR0060
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       SFOR0070
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      SFOR0080
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        SFOR0090
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          SFOR0100
     6            (PSIFID,VAR(22))                                      SFOR0110
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      SFOR0120
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))SFOR0130
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),      SFOR0140
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    SFOR0150
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   SFOR0160
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         SFOR0170
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     SFOR0180
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         SFOR0190
     1            (DER(10),DPHIFD)                                      SFOR0200
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       SFOR0210
     1            (DER(14),DDEL4D)                                      SFOR0220
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,SFOR0230
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  SFOR0240
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),SFOR0250
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    SFOR0260
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), SFOR0270
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), SFOR0280
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    SFOR0290
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     SFOR0300
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),SFOR0310
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)SFOR0320
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      SFOR0330
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),    SFOR0340
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1F1(2),F1RI(2),  SFOR0350
     3             F2F1(2),F2RI(2),CAH(4),CBH(4),CGH(4)                 SFOR0360
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       SFOR0370
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   SFOR0380
     1            (PSII(1),PSI1)                                        SFOR0390
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,SFOR0400
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         SFOR0410
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, SFOR0420
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   SFOR0430
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   SFOR0440
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPSFOR0450
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    SFOR0460
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   SFOR0470
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, SFOR0480
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  SFOR0490
```

```
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  SFOR0500
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,   SFOR0510
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,SFOR0520
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2SFOR0530
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,SFOR0540
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL                SFOR0550
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 SFOR0560
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       SFOR0570
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     SFOR0580
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   SFOR0590
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   SFOR0600
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       SFOR0610
      LOGICAL LCB1,LCB2                                                 SFOR0620
      COMMON /INPT2/ YBPO,ZBTP,ZBBP,XVF,XVR,YV,ZVT,ZVB,AKV,SIGR(11),    SFOR0630
     1               SET,CONS,AMUB,EPSV,EPSB,XM,EPST,DDD,INDB,DELYBP,   SFOR0640
     2               DELTB,XINPT(100)                                   SFOR0650
      COMMON/BARIER/FN,IBHIT,JBHIT,XCPNP(3),YCPNP(3),ZCPNP(3),XCPN(3),  SFOR0660
     1              YCPN(3),ZCPN(3),AA1(17),BB1(17),CC1(17),RR1(17),    SFOR0670
     2              AA2(17),BB2(17),CC2(17),RR2(17),CAB,CBB,CGB,CABT,   SFOR0680
     3              CBBT,CGBT,RB,XBT,YBT,ZBT,XBB,YBB,ZBB,RR2P(17),      SFOR0690
     4              YBPT,XNN(17),YNN(17),ZNN(17),XMTX(3,4),IDPT(17),IPT SFOR0700
     5              ,ININD,UNP(17),VNP(17),WNP(17),VMAX(4),I1,I2,I3,I4, SFOR0710
     6              XCPTP,YCPTP,ZCPTP,XCPBP,YCPBP,ZCPBP,YCPMP,AINTI,    SFOR0720
     7              AINTP,SXR,SYR,SZR,SDEN,XRI,YRI,ZRI,FRICT,DELBB,VTAN,SFOR0730
     8              FNP,FB,URP,VRP,WRP,EPSL,XLDP,DELX,VL,NCYC,EEE,ENRGY,SFOR0740
     9              NSEG,YBPTP,PCAB,PCBB,PCGB,PPRB,CAB1,CBB1,CGB1,      SFOR0750
     A              RB1,NUNLD,NLDCTR,VDEF,PVDEF,PSZR,XF,DELBBP,         SFOR0760
     B              SWORK,SPENGY,DISS,IPLN,ILOAD                        SFOR0770
      DIMENSION INDXPT(4)                                               SFOR0780
      EQUIVALENCE (INDXPT(1),I1)                                        SFOR0790
      COMMON/BARSTR/ XSTIO(3),YSTIO(3),ZSTIO(3),XSTI(3),YSTI(3),        SFOR0800
     1               ZSTI(3),YSTIPO(3),XSTIP(3),YSTIP(3),ZSTIP(3),      SFOR0810
     2               FNSTI(3),AKST(3)                                   SFOR0820
      COMMON /HARDPT/ FRICF(4),UPT(4),VPT(4),WPT(4)                     SFOR0830
    1 SFXS = 0.0                                                        SFOR0840
      YBP = 0.0                                                         SFOR0850
      SFYS = 0.0                                                        SFOR0860
      SFZS = 0.0                                                        SFOR0870
      SNPS = 0.0                                                        SFOR0880
      SNTS = 0.0                                                        SFOR0890
      SNPSS = 0.0                                                       SFOR0900
      FN = 0.0                                                          SFOR0910
      IBHIT = 0                                                         SFOR0920
      IPLN = 0                                                          SFOR0930
      NAXIS = 0                                                         SFOR0940
      FRICT = 0.0                                                       SFOR0950
      VTAN = 0.0                                                        SFOR0960
      VMAX(1) = 0.0                                                     SFOR0970
      NSLCE = 0                                                         SFOR0980
      NUNLD = 0                                                         SFOR0990
      NUNLD2 = 0                                                        SFOR1000
```

```
      YB1VF = 0.0
      IF(INDB.EQ.0) RETURN
      IB = (INDB+1)/2
    2 DO 3  I=1,3
      XCPNP(I) = XCP+AMTX(1,1)*XCPN(I)+AMTX(1,2)*YCPN(I)+AMTX(1,3)*
     1           ZCPN(I)
      YCPNP(I) = YCP+AMTX(2,1)*XCPN(I)+AMTX(2,2)*YCPN(I)+AMTX(2,3)*
     1           ZCPN(I)
      ZCPNP(I) = ZCP+AMTX(3,1)*XCPN(I)+AMTX(3,2)*YCPN(I)+AMTX(3,3)*
     1           ZCPN(I)
      YSTIPO(I) = YCP+AMTX(2,1)*XSTIO(I)+AMTX(2,2)*YSTIO(I)
     1           +AMTX(2,3)*ZSTIO(I)
    3 CONTINUE
      YPMAX = -1.0E30
    4 DO 5  I=1,3
      IF(YCPNP(I).LT.YPMAX) GO TO 5
      YPMAX = YCPNP(I)
      NDX = I
    5 CONTINUE
      XCPTP = XCP+AMTX(1,1)*XCPN(NDX)+AMTX(1,2)*YCPN(NDX)+AMTX(1,3)*ZVT
      YCPTP = YCP+AMTX(2,1)*XCPN(NDX)+AMTX(2,2)*YCPN(NDX)+AMTX(2,3)*ZVT
      ZCPTP = ZCP+AMTX(3,1)*XCPN(NDX)+AMTX(3,2)*YCPN(NDX)+AMTX(3,3)*ZVT
      XCPBP = XCP+AMTX(1,1)*XCPN(NDX)+AMTX(1,2)*YCPN(NDX)+AMTX(1,3)*ZVB
      YCPBP = YCP+AMTX(2,1)*XCPN(NDX)+AMTX(2,2)*YCPN(NDX)+AMTX(2,3)*ZVB
      ZCPBP = ZCP+AMTX(3,1)*XCPN(NDX)+AMTX(3,2)*YCPN(NDX)+AMTX(3,3)*ZVB
    6 YCPMP = AMAX1(YCPTP,YCPBP)
      IF(YBPO-YCPMP.LT.5.0) IBHIT=1
      VDEF = AMAX1(YCPMP-YBPTP,0.0)
      IF(VDEF.LT.2.0*DELYBP) GO TO 41
      IF(MOD(INDB,2).EQ.0) GO TO 8
    7 CABT = AMTX(3,1)
      CBBT = AMTX(3,2)
      CGBT = AMTX(3,3)
      TMP = ZBTP-ZCP
      XBT = -AMTX(1,1)*XCP-AMTX(2,1)*YCP+AMTX(3,1)*TMP
      YBT = -AMTX(1,2)*XCP-AMTX(2,2)*YCP+AMTX(3,2)*TMP
      ZBT = -AMTX(1,3)*XCP-AMTX(2,3)*YCP+AMTX(3,3)*TMP
      RBT = XBT*CABT+YBT*CBBT+ZBT*CGBT
      TMP = ZBBP-ZCP
      XBB = -AMTX(1,1)*XCP-AMTX(2,1)*YCP+AMTX(3,1)*TMP
      YBB = -AMTX(1,2)*XCP-AMTX(2,2)*YCP+AMTX(3,2)*TMP
      ZBB = -AMTX(1,3)*XCP-AMTX(2,3)*YCP+AMTX(3,3)*TMP
      RBB = XBB*CABT+YBB*CBBT+ZBB*CGBT
    8 CAB = AMTX(2,1)
      CBB = AMTX(2,2)
      CGB = AMTX(2,3)
      TMP = YBPTP-YCP
      IF(ININD.LT.2.OR.CAB*PCBB.EQ.CBB*PCAB) GO TO 80
      XBPP = -AMTX(1,1)*XCP+AMTX(2,1)*TMP-AMTX(3,1)*ZCP
      YBPP = -AMTX(1,2)*XCP+AMTX(2,2)*TMP-AMTX(3,2)*ZCP
      ZBPP = -AMTX(1,3)*XCP+AMTX(2,3)*TMP-AMTX(3,3)*ZCP
```

```
      RBPP = XBPP*CAB+YBPP*CBB+ZBPP*CGB                                 SFOR1520
      XMTX(1,1) = CAB                                                   SFOR1530
      XMTX(1,2) = CBB                                                   SFOR1540
      XMTX(1,3) = CGB                                                   SFOR1550
      XMTX(1,4) = RBPP                                                  SFOR1560
      XMTX(2,1) = PCAB                                                  SFOR1570
      XMTX(2,2) = PCBB                                                  SFOR1580
      XMTX(2,3) = PCGB                                                  SFOR1590
      XMTX(2,4) = PPRB                                                  SFOR1600
      XMTX(3,1) = 0                                                     SFOR1610
      XMTX(3,2) = 0                                                     SFOR1620
      XMTX(3,3) = 1                                                     SFOR1630
      XMTX(3,4) = PSZR                                                  SFOR1640
      CALL SIMSOL(XMTX,3,3)                                             SFOR1650
      XB1 = XMTX(1,4)                                                   SFOR1660
      YB1 = XMTX(2,4)                                                   SFOR1670
      ZB1 = XMTX(3,4)                                                   SFOR1680
      IF (XVR.LE.XB1.AND.XB1.LE.XVF.AND.ABS(YB1).LT.YV.AND.ZVT.LE.ZB1   SFOR1690
     1.AND.ZB1.LE.ZVB) NAXIS = 1                                        SFOR1700
      IF(NAXIS.EQ.0.AND.VDEF.LT.PVDEF.AND.XB1.LT.XVR) GO TO 41          SFOR1710
      TMPA = CBB*PCGB-CGB*PCBB                                          SFOR1720
      TMPB = CGB*PCAB-CAB*PCGB                                          SFOR1730
      TMPC = CAB*PCBB-CBB*PCAB                                          SFOR1740
      TMPAP = TMPB*CGB-TMPC*CBB                                         SFOR1750
      TMPBP =-TMPC*CAB+TMPA*CGB                                         SFOR1760
      TMPCP =-TMPA*CBB+TMPB*CAB                                         SFOR1770
      TMPD = SQRT(TMPAP**2+TMPBP**2+TMPCP**2)                           SFOR1780
      CAB1 = TMPAP/TMPD                                                 SFOR1790
      CBB1 = TMPBP/TMPD                                                 SFOR1800
      CGB1 = TMPCP/TMPD                                                 SFOR1810
      RB1 = XB1*CAB1+YB1*CBB1+ZB1*CGB1                                  SFOR1820
      YB1VF = 1.0E6                                                     SFOR1830
      IF(CBB1.NE.0.) YB1VF=(RB1-XVF*CAB1)/CBB1                          SFOR1840
   78 DO 79 I=12,17                                                     SFOR1850
      AA2(I) = CAB1                                                     SFOR1860
      BB2(I) = CBB1                                                     SFOR1870
      CC2(I) = CGB1                                                     SFOR1880
      RR2(I) = RB1                                                      SFOR1890
   79 CONTINUE                                                          SFOR1900
C  PRESENT LOCATION OF HARDPOINTS IN SPACE FIXED COORDINATES            SFOR1910
   80 DO 81 I=1,3                                                       SFOR1920
      XSTIP(I)=XCP+AMTX(1,1)*XSTI(I)+AMTX(1,2)*YSTI(I)+AMTX(1,3)*ZSTI(I)SFOR1930
      YSTIP(I)=YCP+AMTX(2,1)*XSTI(I)+AMTX(2,2)*YSTI(I)+AMTX(2,3)*ZSTI(I)SFOR1940
      ZSTIP(I)=ZCP+AMTX(3,1)*XSTI(I)+AMTX(3,2)*YSTI(I)+AMTX(3,3)*ZSTI(I)SFOR1950
   81 CONTINUE                                                          SFOR1960
      XRI = 0.0                                                         SFOR1970
      YRI = 0.0                                                         SFOR1980
      ZRI = 0.0                                                         SFOR1990
      AINTI = 0.0                                                       SFOR2000
      SXR = 0.0                                                         SFOR2010
      SYR = 0.0                                                         SFOR2020
```

```
      SZR = 0.0                                                        SFOR2030
      SDEN = 0.0                                                       SFOR2040
      FNX = 0.0                                                        SFOR2050
      FNX1 = 0.0                                                       SFOR2060
      FB = 0.0                                                         SFOR2070
      FBFN = 0.0                                                       SFOR2080
      SFNST = 0.0                                                      SFOR2090
      NSEG = (YCPMP-YBPTP)/DELYBP+1.0                                  SFOR2100
      IPLN = NSEG                                                      SFOR2110
      YBP = YBPTP+IPLN*DELYBP                                          SFOR2120
      NSG111 = NSEG+1                                                  SFOR2130
      I111 = 1                                                         SFOR2140
    9 DO 38 I=I111,NSG111                                              SFOR2150
      IPLNP = IPLN                                                     SFOR2160
      PYBP = YBP                                                       SFOR2170
      PDELBB = DELBB                                                   SFOR2180
      PPSXR = SXR                                                      SFOR2190
      PPSYR = SYR                                                      SFOR2200
      PPSZR = SZR                                                      SFOR2210
      PSDEN = SDEN                                                     SFOR2220
      PFNX = FNX                                                       SFOR2230
      PFNX1 = FNX1                                                     SFOR2240
      PFB = FB                                                         SFOR2250
      PFBFN = FBFN                                                     SFOR2260
      PSFNST = SFNST                                                   SFOR2270
      SFNST = 0.                                                       SFOR2280
      IPLN = NSEG-I+1                                                  SFOR2290
      YBP = YBPTP+IPLN*DELYBP                                          SFOR2300
      IF(YBP.LT.YBPO+EPSL+SET*DELX)GO TO 40                            SFOR2310
      TMP = YBP-YCP                                                    SFOR2320
      XBI = -AMTX(1,1)*XCP+AMTX(2,1)*TMP-AMTX(3,1)*ZCP                 SFOR2330
      YBI = -AMTX(1,2)*XCP+AMTX(2,2)*TMP-AMTX(3,2)*ZCP                 SFOR2340
      ZBI = -AMTX(1,3)*XCP+AMTX(2,3)*TMP-AMTX(3,3)*ZCP                 SFOR2350
      RBI = XBI*CAB+YBI*CBB+ZBI*CGB                                    SFOR2360
      IPT = 0                                                          SFOR2370
   10 DO 15  J=1,17                                                    SFOR2380
      IDPT(J) = 0                                                      SFOR2390
      IF(PSIT.LE.0.0.AND.J.LE.2)GO TO 15                               SFOR2400
      IF(ININD.LT.2.AND.J.GT.11) GO TO 15                              SFOR2410
      IF(CAB.EQ.0..AND.(J.EQ.4.OR.J.EQ.5.OR.J.EQ.10.OR.J.EQ.11))GO TO 15SFOR2420
      IF(CBB.EQ.0..AND.(J.LE.2.OR.J.EQ.7.OR.J.EQ.8)) GO TO 15          SFOR2430
      IF(CGB.EQ.0..AND.(J.EQ.3.OR.J.EQ.6.OR.J.EQ.9)) GO TO 15          SFOR2440
      IF(CAB1*CBB.EQ.CBB1*CAB.AND.(J.EQ.12.OR.J.EQ.13)) GO TO 15       SFOR2450
      IF(CBB1*CGB.EQ.CGB1*CBB.AND.(J.EQ.14.OR.J.EQ.15)) GO TO 15       SFOR2460
      IF(CGB1*CAB.EQ.CAB1*CGB.AND.J.GE.16) GO TO 15                    SFOR2470
      IF(NAXIS.EQ.0.AND.J.GT.11) GO TO 15                              SFOR2480
   11 XMTX(1,1) = CAB                                                  SFOR2490
      XMTX(1,2) = CBB                                                  SFOR2500
      XMTX(1,3) = CGB                                                  SFOR2510
      XMTX(1,4) = RBI                                                  SFOR2520
   12 XMTX(2,1) = AA1(J)                                               SFOR2530
```

```
      XMTX(2,2) = BB1(J)                                                SFOR2540
      XMTX(2,3) = CC1(J)                                                SFOR2550
      XMTX(2,4) = RR1(J)                                                SFOR2560
   13 XMTX(3,1) = AA2(J)                                                SFOR2570
      XMTX(3,2) = BB2(J)                                                SFOR2580
      XMTX(3,3) = CC2(J)                                                SFOR2590
      XMTX(3,4) = RR2(J)                                                SFOR2600
   14 CALL SIMSOL(XMTX,3,3)                                             SFOR2610
      XNN(J) = XMTX(1,4)                                                SFOR2620
      YNN(J) = XMTX(2,4)                                                SFOR2630
      ZNN(J) = XMTX(3,4)                                                SFOR2640
      IF(XNN(J).LT.XVR.OR.XNN(J).GT.XVF) GO TO 15                       SFOR2650
      IF(ABS(YNN(J)).GT.YV) GO TO 15                                    SFOR2660
      IF(ZNN(J).LT.ZVT.OR.ZNN(J).GT.ZVB) GO TO 15                       SFOR2670
      IDPT(J) = 1                                                       SFOR2680
      IPT = IPT+1                                                       SFOR2690
      IPPT = J                                                          SFOR2700
   15 CONTINUE                                                          SFOR2710
      IF(IPPT.LE.11.AND.(NAXIS.EQ.1.AND.YB1VF.GT.YV.AND.ININD.EQ.2))    SFOR2720
     1 GO TO 38                                                         SFOR2730
      IF(MOD(INDB,2).EQ.0) GO TO 23                                     SFOR2740
      IF(CGB.EQ.0.0.AND.CGBT.EQ.0.0)GO TO 23                            SFOR2750
      RR2P(1) = RBT                                                     SFOR2760
      RR2P(2) = RBB                                                     SFOR2770
      RR2P(4) = RBT                                                     SFOR2780
      RR2P(5) = RBB                                                     SFOR2790
      RR2P(7) = RBT                                                     SFOR2800
      RR2P(8) = RBB                                                     SFOR2810
      RR2P(10) = RBT                                                    SFOR2820
      RR2P(11) = RBB                                                    SFOR2830
      RR2P(12) = RBT                                                    SFOR2840
      RR2P(13) = RBB                                                    SFOR2850
      RR2P(14) = RBT                                                    SFOR2860
      RR2P(15) = RBB                                                    SFOR2870
      RR2P(16) = RBT                                                    SFOR2880
      RR2P(17) = RBB                                                    SFOR2890
   16 DO 22  J=1,17                                                     SFOR2900
      IF(PSIT.LE.0.0.AND.J.LE.2)GO TO 22                                SFOR2910
      IF(J.EQ.3.OR.J.EQ.6.OR.J.EQ.9) GO TO 22                           SFOR2920
      IF(CAB*CGBT.EQ.CGB*CABT.AND.(J.EQ.4.OR.J.EQ.5.OR.J.EQ.10.OR.      SFOR2930
     1   J.EQ.11)) GO TO 22                                             SFOR2940
      IF(CGB*CBBT.EQ.CBB*CGBT.AND.(J.LE.2.OR.J.EQ.7.OR.J.EQ.8)) GO TO 22SFOR2950
      IF(CAB*(CBB1*CGBT-CBBT*CGB1)-CAB1*(CBB*CGBT-CBBT*CGB)+CABT*(CBB*  SFOR2960
     1   CGB1-CBB1*CGB).EQ.0.0.AND.J.GE.12) GO TO 22                    SFOR2970
      IF(J.GE.12.AND.IDPT(J).NE.1) GO TO 22                             SFOR2980
      IF(IDPT(1).EQ.1.AND.IDPT(2).EQ.1.AND.J.EQ.14) GO TO 173           SFOR2990
      IF(IDPT(7).EQ.1.AND.IDPT(8).EQ.1.AND.J.EQ.15) GO TO 173           SFOR3000
      IF(IDPT(4).EQ.1.AND.IDPT(5).EQ.1.AND.J.EQ.16) GO TO 173           SFOR3010
      IF(IDPT(10).EQ.1.AND.IDPT(11).EQ.1.AND.J.EQ.17) GO TO 173         SFOR3020
      XMTX(1,1) = CAB                                                   SFOR3030
      XMTX(1,2) = CBB                                                   SFOR3040
```

```
      XMTX(1,3) = CGB
      XMTX(1,4) = RBI
      IF(J.GE.12) GO TO 170
      XMTX(2,2) = BB1(J)
      XMTX(2,3) = CC1(J)
   17 XMTX(2,1) = AA1(J)
      XMTX(2,4) = RR1(J)
      GO TO 18
  170 XMTX(2,1) = AA2(J)
      XMTX(2,2) = BB2(J)
      XMTX(2,3) = CC2(J)
      XMTX(2,4) = RR2(J)
   18 XMTX(3,1) = CABT
      XMTX(3,2) = CBBT
      XMTX(3,3) = CGBT
      IF((IDPT(1).EQ.1.AND.J.EQ.14).OR.(IDPT(4).EQ.1.AND.J.EQ.16))
     1 GO TO 171
      IF((IDPT(8).EQ.1.AND.J.EQ.15).OR.(IDPT(11).EQ.1.AND.J.EQ.17))
     1 GO TO 172
      XMTX(3,4) = RR2P(J)
      GO TO 19
  171 XMTX(3,4) = RBB
      GO TO 19
  172 XMTX(3,4) = RBT
   19 CALL SIMSOL(XMTX,3,3)
      IF(XMTX(1,4).LT.XVR.OR.XMTX(1,4).GT.XVF) GO TO 22
      IF(ABS(XMTX(2,4)).GT.YV) GO TO 22
      IF(XMTX(3,4).LT.ZVT.OR.XMTX(3,4).GT.ZVB) GO TO 22
      IF(IDPT(J).NE.0) GO TO 20
      IDPT(J) = 1
      GO TO 21
   20 IF(ABS(XMTX(3,4)).GE.ABS(ZNN(J)))GO TO 22
   21 XNN(J) = XMTX(1,4)
      YNN(J) = XMTX(2,4)
      ZNN(J) = XMTX(3,4)
      GO TO 22
  173 IDPT(J) = 0
      IPT = IPT-1
   22 CONTINUE
   23 IF(IPT.LT.3) GO TO 38
   24 DO 25  J=1,17
      IF(IDPT(J).EQ.0) GO TO 25
      TMPU = U-YNN(J)*R+ZNN(J)*Q
      TMPV = V+XNN(J)*R-ZNN(J)*P
      TMPW = W+YNN(J)*P-XNN(J)*Q
      UNP(J) = AMTX(1,1)*TMPU+AMTX(1,2)*TMPV+AMTX(1,3)*TMPW
      VNP(J) = AMTX(2,1)*TMPU+AMTX(2,2)*TMPV+AMTX(2,3)*TMPW
      WNP(J) = AMTX(3,1)*TMPU+AMTX(3,2)*TMPV+AMTX(3,3)*TMPW
   25 CONTINUE
   26 DO 27  J=1,4
      VMAX(J) = -1.0E30
```

```
      INDXPT(J) = 0
   27 CONTINUE
   28 DO 34  J=1,17
      IF(IDPT(J).EQ.0) GO TO 34
   29 DO 33  K=1,4
      IF(VNP(J).LT.VMAX(K)) GO TO 33
      IF(K.EQ.4) GO TO 32
      K1 = K+1
   30 DO 31  L=K1,4
      M = 4-L+K1
      VMAX(M) = VMAX(M-1)
      INDXPT(M) = INDXPT(M-1)
   31 CONTINUE
   32 VMAX(K) = VNP(J)
      INDXPT(K) = J
      GO TO 34
   33 CONTINUE
   34 CONTINUE
      IPT = 4
      IF(INDXPT(4).EQ.0) IPT = 3
   37 J3 = I3
      J1 = I1
      J2 = I2
      J4 = I4
      CALL AREA
      DO 91 IJ = 1,3
      FNSTI(IJ) = 0.0
      IF(VPT(IJ+1).GE.0.0.AND.YSTIPO(IJ).GE.YBP) FNSTI(IJ) =
     1          AKST(IJ)*(YSTIPO(IJ)-YBP)
      SFNST = SFNST+FNSTI(IJ)
   91 CONTINUE
      IF(IB.EQ.1) GO TO 38
      FNX1 = AKV*DELYBP*SDEN
      FNX = FNX1+SFNST
      IF(NSLCE.NE.0) GO TO 36
   40 DELBB = AMAX1(YBP-YBPO,EPSL+SET*DELX)
      CALL NLDFRC
      FBFN = FB-FNX
      IF(EPSB.LT.FBFN) GO TO 38
      IF(1.EQ.1) GO TO 105
      IF(FBFN.GE.0.0)  GO TO 105
      IF(ABS(FBFN).LT.ABS(PFBFN)) GO TO 105
      WRITE(6,1001) T,I,YBP,PYBP,FNX,PFNX
 1001 FORMAT(4H  T=,F7.4,3H I=,I3,5H YBP=,F10.4,6H PYBP=,F10.4,
     1          5H FNX=,G13.5,6H PFNX=,G13.5,27HEQUILIB AT PREV SLICE RESET
     2          )
      IPLN = IPLNP
      YBP = PYBP
      DELBB = PDELBB
      SXR = PPSXR
      SYR = PPSYR
```

```
      SZR = PPSZR
      SDEN = PSDEN
      FNX = PFNX
      FNX1 = PFNX1
      FB = PFB
      SFNST = PSFNST
  105 YBPT = AMAX1(YBP,YBPO+EPSL+SET*DELX)
      NSLCE = NSLCE+1
      IF(NLDCTR.EQ.3)CALL NLDFL
      NUNLD2 = 0
      IF(NUNLD.EQ.0) GO TO 38
      NUNLD2 = 1
      GO TO 110
   38 CONTINUE
  110 DO 111 IJ=1,3
      IF(YSTIP(IJ).GT.YBPT) YSTIP(IJ) = YBPT
      AA = XSTIP(IJ)-XCP
      BB = YSTIP(IJ)-YCP
      CC = ZSTIP(IJ)-ZCP
      XSTI(IJ) = AMTX(1,1)*AA+AMTX(2,1)*BB+AMTX(3,1)*CC
      YSTI(IJ) = AMTX(1,2)*AA+AMTX(2,2)*BB+AMTX(3,2)*CC
      ZSTI(IJ) = AMTX(1,3)*AA+AMTX(2,3)*BB+AMTX(3,3)*CC
  111 CONTINUE
      IF(NUNLD2.NE.0) GO TO 103
      IF(NUNLD.NE.0) GO TO 100
      IF( IB .NE. 1)  GO TO 50
   45 NEGPT=0
      DO 46  J=1,IPT
      IF( VMAX(J) .LT.  0.0 )  NEGPT=NEGPT + 1
   46 CONTINUE
      IF( NEGPT .GE. IPT) GO TO 41
   50 FN = AKV*DELYBP*SDEN
      FN1 = FN+SFNST
      IF(ININD.EQ.0) ININD = 1
      IF(FN1.NE.0.0.AND.NUNLD.EQ.0) CALL RESFRC
      IF(NSLCE.EQ.0.AND.IB.EQ.1) GO TO 103
      IF(NSLCE.EQ.0) GO TO 100
  103 TMP = YBPT-YCP
      NUNLD2 = 0
      XBPP = -AMTX(1,1)*XCP+AMTX(2,1)*TMP-AMTX(3,1)*ZCP
      YBPP = -AMTX(1,2)*XCP+AMTX(2,2)*TMP-AMTX(3,2)*ZCP
      ZBPP = -AMTX(1,3)*XCP+AMTX(2,3)*TMP-AMTX(3,3)*ZCP
      RB = XBPP*CAB + YBPP*CBB + ZBPP*CGB
      GO TO 39
  100 IF(YBP.GT.YBPO) GO TO 250
      YBPT = YBPO
      FB = 0.0
      GO TO 103
  250 NUNLD = NUNLD+1
      NSG111 = NSG111+1
      I111 = NSG111
```

```
      GO TO 9                                                           SFOR4580
   39 NUNLD = 0                                                         SFOR4590
   41 IF(NLDCTR.EQ.3.AND.IPT.GE.3)WRITE(6,1000)T,XB1,YB1,IPT,J1,J2,J3,  SFOR4600
     1         J4,XNN(J1),YNN(J1),ZNN(J1),XNN(J2),YNN(J2),ZNN(J2),      SFOR4610
     2            XNN(J3),YNN(J3),ZNN(J3),XNN(J4),YNN(J4),ZNN(J4)       SFOR4620
 1000 FORMAT(F7.4,2F7.1,5I3,12F8.1)                                     SFOR4630
      NLDCTR = NLDCTR+1                                                 SFOR4640
      RETURN                                                            SFOR4650
      END                                                               SFOR4660
```

```
      SUBROUTINE SIMSOL (A,KK,LL)
C***********************************************************************
C*                                                                     *
C*    SUBROUTINE SIMSOL      (SINGLE PRECISION VERSION)                *
C*                                                                     *
C*    AUTHOR                                                           *
C*       DR.JOHN T. FLECK                                              *
C*       (REVISED BY F.E. BUTLER)                                      *
C*                                                                     *
C*    REFERENCE                                                        *
C*       ''SUBROUTINES TO SOLVE AN INDEPENDENT SET OF LINEAR           *
C*       SIMULTANEOUS EQUATIONS''  HS/FEB/PAW-84, 21 JULY 1965.        *
C*                                                                     *
C*    PURPOSE                                                          *
C*       TO SOLVE A SET OF SIMULTANEOUS LINEAR EQUATIONS, AX=B.        *
C*                                                                     *
C*    USAGE                                                            *
C*       CALL SIMSOL (A,N,ND1)                                         *
C*                                                                     *
C*    DESCRIPTION OF PARAMETERS                                        *
C*       A - IS A 2-DIMENSIONAL (ND1*ND2) MATRIX OF COEFFICIENTS.      *
C*       N - IS THE NUMBER OF EQUATIONS AND UNKNOWNS.                  *
C*       ND1 - IS THE FIRST DIMENSION OF A IN CALLING PROGRAM.         *
C*            ( ND1.GE.N  AND  ND2.GE.N+1 )                            *
C*                                                                     *
C*    CALLING PROGRAM SETUP                                            *
C*       A(I,J) FOR I,J=1,N                                            *
C*       A(I,N+1)=B(I) FOR I=1,N                                       *
C*       THE SOLUTION IS RETURNED IN COLUMN N+1 OF MATRIX A.           *
C*       MATRIX A IS DESTROYED BY THE SUBROUTINE.                      *
 1000 CONTINUE
C*    REMARKS                                                          *
C*       IF MATRIX A IS SINGULAR, AN ERROR MESSAGE IS PRINTED          *
C*       AND THE JOB IS TERMINATED.                                    *
C*                                                                     *
C*    METHOD                                                           *
C*       SOLUTION IS OBTAINED BY ELIMINATION USING LARGEST PIVOTAL     *
C*       DIVISOR OF EACH COLUMN. EACH STAGE OF ELIMINATION CONSISTS    *
C*       OF INTERCHANGING ROWS WHEN NECESSARY TO AVOID DIVISION BY     *
C*       ZERO OR SMALL NUMBERS.                                        *
C*       THE FORWARD SOLUTION TO OBTAIN VARIABLE N IS DONE IN N        *
C*       STAGES. THE BACK SOLUTION FOR THE OTHER VARIABLES IS          *
C*       CALCULATED BY SUCCESSIVE SUBSTITUTIONS. FINAL SOLUTION        *
C*       VALUES ARE DEVELOPED IN COLUMN N+1 OF MATRIX A, WITH          *
C*       VARIABLE 1 IN A(1,N+1), VARIABLE 2 IN A(2,N+1),.....,         *
C*       VARIABLE N IN A(N,N+1).                                       *
C*                                                                     *
C***********************************************************************
      REAL A(LL,1),B,BIG
      N  = KK
      N1 = N+1
      DO 50  L=1,N
```

```
      L1 = L+1
      BIG = 0.0
      DO 25  I=L,N
      IF (ABS(A(I,L)).LE.ABS(BIG)) GO TO 25
      K = I
      BIG = A(I,L)
   25 CONTINUE
      IF (BIG.NE.0.0) GO TO 30
      WRITE(6,32000)
32000  FORMAT(24H SIMSOL MATRIX SINGULAR.)
      STOP
   30 DO 40  J=L,N1
      IF (K.EQ.L) GO TO 40
      B      = A(K,J)
      A(K,J) = A(L,J)
      A(L,J) = B
   40 A(L,J) = A(L,J)/BIG
      IF (L.EQ.N) GO TO 50
      DO 48  I=L1,N
      IF (A(I,L).EQ.0.0) GO TO 48
      DO 45  J=L1,N1
   45 A(I,J) = A(I,J)-A(I,L)*A(L,J)
   48 CONTINUE
   50 CONTINUE
      IF (N.EQ.1) RETURN
      N2 = N-1
      DO 60  L=1,N2
      I  = N-L
      L1 = I+1
      DO 60  J=L1,N
   60 A(I,N1) = A(I,N1)-A(I,J)*A(J,N1)
      RETURN
      END
```

```
      SUBROUTINE SUSFRC(DISP,VEL)
C              HVOSM-RD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
C
C   SUBROUTINE TO COMPUTE SUSPENSION FORCES ACTING BETWEEN SPRUNG
C   AND UNSPRUNG MASSES
C
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1             A,B,DEL1O,DEL2O,DEL3O,PHIRO,DEL1OD,DEL2OD,DEL3OD,
     2             PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3             XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4             RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5             T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6             HED(36),DADE(3),X1R,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7             DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8             NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9             NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1           XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,
     1               OMEGRC,AKRE,AKREP,OMEGRE,END3
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
```

```
9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)SUSF0500
 COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),         SUSF0510
1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),       SUSF0520
2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),     SUSF0530
3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                    SUSF0540
 DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                          SUSF0550
 EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),      SUSF0560
1            (PSII(1),PSI1)                                           SUSF0570
 COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,SUSF0580
1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,            SUSF0590
2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,    SUSF0600
3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,      SUSF0610
4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,DIPD2,D1MD2,      SUSF0620
5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPSUSF0630
6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,       SUSF0640
7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,      SUSF0650
8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,    SUSF0660
9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ     SUSF0670
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,     SUSF0680
1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,    SUSF0690
2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,SUSF0700
3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2SUSF0710
4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,SUSF0720
5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             SUSF0730
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                    SUSF0740
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)          SUSF0750
 COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                   SUSF0760
1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,          SUSF0770
2              SLOPE1,SLOPE2,XTRA(300)                                SUSF0780
 DIMENSION UI(4),VI(4),WI(4)                                          SUSF0790
 EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                         SUSF0800
 DIMENSION APITCH(4)                                                  SUSF0810
 EQUIVALENCE (APITCH(1),APTCH1)                                       SUSF0820
 COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,           SUSF0830
1               DAPRB,DAPRE,DDAPR,NAPR                                SUSF0840
 DIMENSION APF(21),APR(21)                                            SUSF0850
 EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                    SUSF0860
 EQUIVALENCE (IAPF,IAPFR(1)),(IAPR,IAPFR(2))                          SUSF0870
 COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,          SUSF0880
1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),SUSF0890
2               NCAMF,NCAMR,NDTHF,NDTHR                               SUSF0900
 COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,         SUSF0910
1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,        SUSF0920
2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,       SUSF0930
3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,       SUSF0940
4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,           SUSF0950
5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                        SUSF0960
 DIMENSION DISP(4),VEL(4),F1I(4),F2I(4)                               SUSF0970
 EQUIVALENCE (F1I(1),F1FI(1)),(F2I(1),F2FI(1))                        SUSF0980
C                                                                     SUSF0990
 DO 500 I=1,4                                                         SUSF1000
```

```
      IF(I.GE.3) GO TO 200
      IF(EPSF.LE.0.0) GO TO 10
      IF(ABS(VEL(I)).GE.EPSF) GO TO 10
      F1I(I) = (CFP/EPSF)*VEL(I)
      GO TO 20
   10 F1I(I) = SIGN(CFP,VEL(I))
   20 XLM = 1.0
      TMP = DISP(I)*VEL(I)
      IF(DISP(I).GT.OMEGFE) GO TO 30
      IF(DISP(I).LT.OMEGFC) GO TO 40
      F2I(I) = AKF*DISP(1)
      GO TO 100
   30 IF(TMP.LT.0.0) XLM = XLAMF
      DISP1 = DISP(I)-OMEGFE
      F2I(1) = AKF*DISP(I)+XLM*(AKFE*DISP1+AKFEP*DISP1**3)
      GO TO 100
   40 IF(TMP.LT.0.0) XLM = XLAMF
      DISP1 = DISP(I)-OMEGFC
      F2I(I) = AKF*DISP(I)+XLM*(AKFC*DISP1+AKFCP*DISP1**3)
      GO TO 100
  100 IF(IAPF.EQ.0) GO TO 150
      APITCH(I) = 0.0
      IF(FC(I).EQ.0.0) GO TO 150
      TMP3 = COS(PHII(I))*COS(PSII(I))/12.0
      CALL INTRPL(APF,DAPFB,DAPFE,DDAPF,DISP(I),APC)
      APITCH(I) = -APC*FC(I)*HI(I)*TMP3
  150 ABAR = RFTF*D21
      IF(ISUS.EQ.2) GO TO 105
      IF(I.EQ.2) GO TO 102
      FJF(1) = -SLOPE1*(FYU(1)*HCGH1-FZU(1)*HCBH1) + FYU(1)*DTDD1
      GO TO 103
  102 ABAR = -ABAR
      FJF(2) = -SLOPE2*(FYU(2)*HCGH2-FZU(2)*HCBH2) - FYU(2)*DTDD2
      GO TO 103
  105 ABAR =-RTF*PHIF
      IF(I.EQ.2) ABAR = -ABAR
      FJF(I) = 0.0
  103 SI(I) = BO2APB-CF*VEL(I)-F1I(I)-F2I(I)+ABAR+FJF(I)+APITCH(I)
      GO TO 500
  200 IF(EPSR.LE.0.0) GO TO 210
      IF(ABS(VEL(I)).GE.EPSR) GO TO 210
      F1I(I) = (CRP/EPSR)*VEL(I)
      GO TO 220
  210 F1I(I) = SIGN(CRP,VEL(I))
  220 XLM = 1.0
      TMP = DISP(I)*VEL(I)
      IF(DISP(I).GT.OMEGRE) GO TO 230
      IF(DISP(I).LT.OMEGRC) GO TO 240
      F2I(I) = AKR*DISP(I)
      GO TO 300
  230 IF(TMP.LT.0.0) XLM = XLAMR
```

```
      DISP1 = DISP(I)-OMEGRE
      F2I(I) = AKR*DISP(I)+XLM*(AKRE*DISP1+AKREP*DISP1**3)
      GO TO 300
  240 IF(TMP.LT.0.0) XLM = XLAMR
      DISP1 = DISP(I)-OMEGRC
      F2I(I) = AKR*DISP(I)+XLM*(AKRC*DISP1+AKRCP*DISP1**3)
  300 IF(IAPR.EQ.0) GO TO 350
      APITCH(I) = 0.0
      IF(FC(I).EQ.0.0) GO TO 350
      TMP3 = COS(PHII(I))*COS(PSII(I))/12.0
      CALL INTRPL(APR,DAPRB,DAPRE,DDAPR,DISP(I),APC)
      APITCH(I) =  APC*FC(I)*HI(I)*TMP3
  350 ABAR = RRTR*D43
      IF(ISUS.NE.1) GO TO 305
      IF(I.EQ.4) GO TO 302
      FJF(3) = -SLOPE3*(FYU(3)*HCGH3-FZU(3)*HCBH3) + FYU(3)*DTDD3
      GO TO 303
  302 ABAR = - ABAR
      FJF(4) = -SLOPE4*(FYU(4)*HCGH4-FZU(4)*HCBH4) - FYU(4)*DTDD4
      GO TO 303
  305 ABAR = -RTR*PHIR
      IF(I.EQ.4) ABAR = -ABAR
      FJF(I) = 0.0
  303 SI(I) = AO2APB-CR*VEL(I)-F1I(I)-F2I(I)+ABAR+APITCH(I)+FJF(I)
  500 CONTINUE
      SFZ1 = SI(1)+SI(2)+SI(3)+SI(4)
      RETURN
      END
```

```
      SUBROUTINE TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)
C              HVOSM-RD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      DIMENSION DUM(18)
      LSEQ = 0
      IF(NNBX.LE.0) GO TO 10
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD
 2000 FORMAT(9F8.0,2I4)
      IF(NSEQ.LT.LSEQ) GO TO 98
      LSEQ = NSEQ
      IF(NNBX.GT.4) GOTO 98
      DO 11 K=1,NNBX
   11 XBDRY(K,I) = DUM(K)
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD
      IF(NSEQ.LT.LSEQ) GO TO 98
      LSEQ = NSEQ
      DO 12 K=1,NNBX
   12 PSBDRO(K,I) = DUM(K)
   10 IF(NNBY.LE.0) GO TO 20
      IF(NNBY.GT.2) GO TO 98
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD
      IF(NSEQ.LT.LSEQ) GO TO 98
      LSEQ = NSEQ
      DO 13 K=1,NNBY
   13 YBDRY(K,I) = DUM(K)
   20 NYCDS = (NNY-1)/9+1
      DO 30 J=1,NNX
      M = 0
      DO 40 K=1,NYCDS
      READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD
      IF(NSEQ.LT.LSEQ) GO TO 98
      LSEQ = NSEQ
      DO 50 N=1,9
      M = M+1
      ZGP(J,M,I) = DUM(N)
      IF(M.GE.NNY) GO TO 30
   50 CONTINUE
```

```
40 CONTINUE                                          TERE0500
30 CONTINUE                                          TERE0510
   IF(NZ5T.EQ.0) GO TO 99                            TERE0520
   M = 0                                             TERE0530
   DO 60 K=1,NYCDS                                   TERE0540
   READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD            TERE0550
   IF(NSEQ.LT.LSEQ) GO TO 98                         TERE0560
   LSEQ = NSEQ                                       TERE0570
   DO 61 N=1,9                                       TERE0580
   M = M+1                                           TERE0590
   YYZGP5(M) = DUM(N)                                TERE0600
   IF(M.GE.NNY) GO TO 70                             TERE0610
61 CONTINUE                                          TERE0620
60 CONTINUE                                          TERE0630
70 NXCDS = (NNX-1)/9 + 1                             TERE0640
   M = 0                                             TERE0650
   DO 71 K=1,NXCDS                                   TERE0660
   READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD            TERE0670
   IF(NSEQ.LT.LSEQ) GO TO 98                         TERE0680
   LSEQ = NSEQ                                       TERE0690
   DO 72 N=1,9                                       TERE0700
   M = M+1                                           TERE0710
   XXZGP5(M) = DUM(N)                                TERE0720
   IF(M.GE.NNX) GO TO 99                             TERE0730
72 CONTINUE                                          TERE0740
71 CONTINUE                                          TERE0750
98 NERR = 1                                          TERE0760
99 RETURN                                            TERE0770
   END                                               TERE0780
```

```
      SUBROUTINE TIRFRC(J)                                              TIRF0010
C               HVOSM-RD2 VERSION                                       TIRF0020
C               REVISED OCTOBER 1975   CALSPAN CORPORATION              TIRF0030
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,TIRF0040
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   TIRF0050
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),TIRF0060
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     TIRF0070
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  TIRF0080
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  TIRF0090
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     TIRF0100
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      TIRF0110
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),TIRF0120
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)TIRF0130
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      TIRF0140
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     TIRF0150
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   TIRF0160
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  TIRF0170
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       TIRF0180
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   TIRF0190
     1            (PSII(1),PSI1)                                        TIRF0200
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,TIRF0210
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          TIRF0220
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  TIRF0230
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    TIRF0240
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    TIRF0250
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPTIRF0260
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     TIRF0270
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    TIRF0280
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  TIRF0290
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   TIRF0300
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  TIRF0310
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  TIRF0320
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,TIRF0330
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2TIRF0340
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,TIRF0350
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           TIRF0360
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 TIRF0370
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       TIRF0380
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     TIRF0390
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    TIRF0400
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    TIRF0410
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        TIRF0420
      LOGICAL LCB1,LCB2                                                 TIRF0430
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                TIRF0440
     1             XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,        TIRF0450
     2             SLOPE1,SLOPE2,XTRA(300)                              TIRF0460
      DIMENSION UI(4),VI(4),WI(4)                                       TIRF0470
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                      TIRF0480
      DIMENSION APITCH(4)                                               TIRF0490
```

```
      EQUIVALENCE (APITCH(1),APTCH1)
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
    1 ITER = 0
      I = J
      K = J
      SPHIC = CAYW(I)*CAGZ(I)+CBYW(I)*CBGZ(I)+CGYW(I)*CGGZ(I)
      PHICI(I) = ARSIN(SPHIC)
      CPHIC = COS(PHICI(I))
      IF(FR(I).NE.0.0) GO TO 2
  100 FC(I) = 0.0
      FS(I) = 0.0
      FRCP(I) = 0.0
      FCXU(I) = 0.0
      FCYU(I) = 0.0
      FCZU(I) = 0.0
      FRXU(I) = 0.0
      FRYU(I) = 0.0
      FRZU(I) = 0.0
      FSXU(I) = 0.0
      FSYU(I) = 0.0
      FSZU(I) = 0.0
      FXU(I) = 0.0
      FYU(I) = 0.0
      FZU(I) = 0.0
C      NOTE THAT SFXU, SFYU, AND SFZU ARE UNCHANGED
      RETURN
    2 I = J
      K = J
      FRTEST = (FR(I) - FS(I)*SPHIC)
      IF(FRTEST.LE.0.0)GO TO 100
      FRCP(I) = FRTEST/CPHIC
      FC(I) = 0.0
      TERM = 0.0
      PSITEM = PSIIP(I) * SIGN(1.0,UG(I))
      IF(UG(I).NE.0.0.OR.VG(I).NE.0.0)TERM = ATAN2(VG(I),ABS(UG(I)))
      IF(TI(I))3,6,4
    3 FCTR = - COS(TERM - PSITEM   )
      FACTOR = XMUGI(I)*FRCP(I) * FCTR
      IF( ABS(TI(I)) - ABS(FACTOR) ) 31,31,32
   31 FC(I) = TI(I) * SIGN(1.0,UG(I))
      GO TO 33
   32 FC(I) = FACTOR* SIGN(1.0,UG(I))
   33 IF( ABS(UG(I)) . LT.  (1.932))  FC(I) = FC(I)*ABS(UG(I))/1.932
      GO TO 6
    4 FC(I) = XMUGI(I) * FRCP(I)
      IF(ABS(TI(I)).LT.ABS(FC(I))) FC(I) = TI(I)
    6 IF(I.EQ.1.OR.I.EQ.3)GO TO 64
      IF(TI(I).LE.0.0)GO TO 64
      I = I-1
```

```
60 DO 61 L=I,J                                                          TIRF1010
   IF( ABS(XMUGI(L)*FRCP(L)) .LT. TI(L)) GO TO 62                       TIRF1020
61 CONTINUE                                                             TIRF1030
   GO TO 64                                                             TIRF1040
62 IF(FC(I)*HI(I).GT.FC(J)*HI(J))GO TO 63                               TIRF1050
   FC(J) = FC(I)*HI(I)/HI(J)                                            TIRF1060
   GO TO 64                                                             TIRF1070
63 K = I                                                                TIRF1080
   FC(I) = FC(J)*HI(J)/HI(I)                                            TIRF1090
64 DO 14 I=K,J                                                          TIRF1100
   FS(I) = 0.0                                                          TIRF1110
   PSITEM = PSIIP(I) * SIGN(1.0,UG(I))                                  TIRF1120
   IF( ABS(FC(I)).LT. ABS(XMUGI(I)*FRCP(I))-1.0) GO TO 65               TIRF1130
   BETBR(I) = 3.1                                                       TIRF1140
   GO TO 12                                                             TIRF1150
C          BETBR(I) SET TO INDICATE SKID  ON OUTPUT.                    TIRF1160
65 IF( ABS(UG(I)) .GT. 0.5) GO TO 7                                     TIRF1170
   IF( ABS(VG(I)) .LE. 0.5) GO TO 12                                    TIRF1180
 7 FS(I) = SQRT((XMUGI(I)*FRCP(I))**2 - FC(I)**2)                       TIRF1190
   IF(FRCP(I).GT.OMEGT(I)*A2(I)) GO TO 8                                TIRF1200
   BETP(I) = (PHICI(I)-.6366*PHICI(I)*ABS(PHICI(I)))*A234(I)*FRCP(I)    TIRF1210
  1          *(A4(I)-FRCP(I)) / (A1(I)*FRCP(I)*(FRCP(I)-A2(I))          TIRF1220
  2          -A0(I)*A2(I))                                              TIRF1230
   BETBR(I) = (TERM+BETP(I)-PSITEM)*(A12(I)*FRCP(I)*                    TIRF1240
  1           (FRCP(I)-A2(I))-A0(I)) / FS(I)                            TIRF1250
   GO TO 9                                                              TIRF1260
 8 BETP(I) = (PHICI(I)-.6366*PHICI(I)*ABS(PHICI(I)))*OMT2A2(I)          TIRF1270
   BETBR(I) = (TERM+BETP(I)-PSITEM)*(OMT2MI(I)-A0(I)) / FS(I)           TIRF1280
 9 IF(ABS(BETBR(I)).LT.3.0) GO TO 10                                    TIRF1290
   FS(I) = SIGN(FS(I),BETBR(I))                                         TIRF1300
   GO TO 11                                                             TIRF1310
10 FS(I) = FS(I)*(BETBR(I)-BETBR(I)*ABS(BETBR(I))/3.+BETBR(I)**3/27.)TIRF1320
11 ITER = ITER+1                                                        TIRF1330
   GO TO (2,12),ITER                                                    TIRF1340
12 FSXU(I) =  FS(I)*(AMTX(1,1)*CAS(I)+AMTX(2,1)*CBS(I)+AMTX(3,1)*       TIRF1350
  1          CGS(I))                                                    TIRF1360
   FSYU(I) =  FS(I)*(AMTX(1,2)*CAS(I)+AMTX(2,2)*CBS(I)+AMTX(3,2)*       TIRF1370
  1          CGS(I))                                                    TIRF1380
   FSZU(I) =  FS(I)*(AMTX(1,3)*CAS(I)+AMTX(2,3)*CBS(I)+AMTX(3,3)*       TIRF1390
  1          CGS(I))                                                    TIRF1400
   FRXU(I) = -FRCP(I)*(AMTX(1,1)*CAGZ(I)+AMTX(2,1)*CBGZ(I)+AMTX(3,1)*TIRF1410
  1          CGGZ(I))                                                   TIRF1420
   FRYU(I) = -FRCP(I)*(AMTX(1,2)*CAGZ(I)+AMTX(2,2)*CBGZ(I)+AMTX(3,2)*TIRF1430
  1          CGGZ(I))                                                   TIRF1440
   FRZU(I) = -FRCP(I)*(AMTX(1,3)*CAGZ(I)+AMTX(2,3)*CBGZ(I)+AMTX(3,3)*TIRF1450
  1          CGGZ(I))                                                   TIRF1460
   FCXU(I) = FC(I)*(AMTX(1,1)*CAC(I)+AMTX(2,1)*CBC(I)+AMTX(3,1)*        TIRF1470
  1          CGC(I))                                                    TIRF1480
   FCYU(I) = FC(I)*(AMTX(1,2)*CAC(I)+AMTX(2,2)*CBC(I)+AMTX(3,2)*        TIRF1490
  1          CGC(I))                                                    TIRF1500
   FCZU(I) = FC(I)*(AMTX(1,3)*CAC(I)+AMTX(2,3)*CBC(I)+AMTX(3,3)*        TIRF1510
```

```
 1          CGC(I))                                    TIRF1520
13 FXU(I) = FRXU(I)+FCXU(I)+FSXU(I)                    TIRF1530
   SFXU = SFXU+FXU(I)                                  TIRF1540
   FYU(I) = FRYU(I)+FCYU(I)+FSYU(I)                    TIRF1550
   SFYU = SFYU+FYU(I)                                  TIRF1560
   FZU(I) = FRZU(I)+FCZU(I)+FSZU(I)                    TIRF1570
   SFZU = SFZU+FZU(I)                                  TIRF1580
14 CONTINUE                                            TIRF1590
   RETURN                                              TIRF1600
   END                                                 TIRF1610
```

```
      SUBROUTINE TMCNST                                                 TMCN0010
C           HVOSM-RD2 VERSION                                           TMCN0020
C           REVISED OCTOBER 1975   CALSPAN CORPORATION                  TMCN0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    TMCN0040
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     TMCN0050
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     TMCN0060
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,TMCN0070
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, TMCN0080
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    TMCN0090
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  TMCN0100
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  TMCN0110
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   TMCN0120
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 TMCN0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),TMCN0140
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN TMCN0150
      COMMON /INTG/NEQ,T,DT, VAR(50),DER(50)                            TMCN0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))TMCN0170
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),TMCN0180
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       TMCN0190
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      TMCN0200
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        TMCN0210
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          TMCN0220
     6            (PSIFID,VAR(22))                                      TMCN0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      TMCN0240
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))TMCN0250
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),      TMCN0260
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    TMCN0270
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   TMCN0280
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         TMCN0290
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     TMCN0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         TMCN0310
     1            (DER(10),DPHIFD)                                      TMCN0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       TMCN0330
     1            (DER(14),DDEL4D)                                      TMCN0340
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,TMCN0350
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         TMCN0360
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, TMCN0370
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   TMCN0380
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   TMCN0390
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPTMCN0400
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    TMCN0410
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   TMCN0420
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, TMCN0430
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  TMCN0440
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  TMCN0450
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, TMCN0460
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,TMCN0470
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2TMCN0480
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,TMCN0490
```

```
     5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL              TMCN0500
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                      TMCN0510
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)            TMCN0520
C      COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST        TMCN0530
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,         TMCN0540
     1               COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,       TMCN0550
     2               STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN        TMCN0560
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,            TMCN0570
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),TMCN0580
     2               NCAMF,NCAMR,NDTHF,NDTHR                                 TMCN0590
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,           TMCN0600
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,          TMCN0610
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,         TMCN0620
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,         TMCN0630
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,             TMCN0640
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                          TMCN0650
      DIMENSION ANAME(3)                                                     TMCN0660
      DATA ANAME(1)/4HPSIT/,ANAME(2)/4HTHET/,ANAME(3)/4HPHIT/                TMCN0670
      COMMON/NSTOP/ISTOP                                                     TMCN0680
C       * * * * FOR TEMPORARY ERROR STOP, USE THE VARIABLE ASTOP AS SHOWTMCN0690
C     ASTOP IS SOME LARGE NUMBER TO BE COMPARED TO THE ANGLES IN RADIANSTMCN0700
      DATA ASTOP/3000./                                                      TMCN0710
      IF(PHITP.GE.ASTOP .OR. THETTP.GE.ASTOP) GO TO 60                       TMCN0720
C          * * * * ** * * *                                                  TMCN0730
C          THETTL,PHITL,PSITL ARE VALUES OF THETT,PHIT,PSIT FROM PREVIOUTMCN0740
C           TIME INTERVAL, USED TO TEST NEW ANGLES     IN SUBROUTINE TMCTMCN0750
C      XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTXTMCN0760
C            XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING TMCN0770
C            THAT IS  THETN OR PHIN NOW .NE. 0.0                         TMCN0780
C            USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST       TMCN0790
      UQ = U*Q                                                               TMCN0800
      WP = W*P                                                               TMCN0810
      UR = U*R                                                               TMCN0820
      QR = Q*R                                                               TMCN0830
      VP = V*P                                                               TMCN0840
      PR = P*R                                                               TMCN0850
      P2 = P*P                                                               TMCN0860
      Q2 = Q*Q                                                               TMCN0870
      R2 = R*R                                                               TMCN0880
      VR = V*R                                                               TMCN0890
      WQ = W*Q                                                               TMCN0900
      PQ = P*Q                                                               TMCN0910
      ZFD1 = ZF+DEL1                                                         TMCN0920
      ZRD3 = ZR+DEL3                                                         TMCN0930
      IF(ISUS.NE.1) GO TO 100                                                TMCN0940
      D3PD4 = DEL3+DEL4                                                      TMCN0950
      D3MD4 = DEL3-DEL4                                                      TMCN0960
      D43 = -D3MD4                                                           TMCN0970
      DD3P4 = DEL3D+DEL4D                                                    TMCN0980
      DD3M4 = DEL3D-DEL4D                                                    TMCN0990
      ZRD34 = ZR+0.5*D3PD4                                                   TMCN1000
```

```
      ZRD4 = ZR+DEL4
      GO TO 200
  100 IF(ISUS.NE.2) GO TO 200
      PHIF2 = PHIF*PHIF
      PHIFD2 = PHIFD*PHIFD
      ZFD1RF = ZFD1+RHOF
      RPF2M = RHF2MF*PHIF2
      RFPF = RHOF*PHIF
      WFMF = XMUF*(DEL1D-RFPF*PHIFD)
      PHFP = PHIF-PHITP
      RPHFD = R*PHIFD
      TPF = 0.5*TF*PHIF
      GO TO 300
  200 IF(ISUS.EQ.2) GO TO 300
      ZFD2 = ZF+DEL2
      D1PD2 = DEL1+DEL2
      D1MD2 = DEL1-DEL2
      DD1P2 = DEL1D+DEL2D
      DD1M2 = DEL1D-DEL2D
      D21 = -D1MD2
      ZFD12 = ZF+0.5*D1PD2
  300  IF(ISUS.EQ.1) GO TO 400
      PHIR2 = PHIR*PHIR
      PHIRD2 = PHIRD*PHIRD
      ZRD3R = ZRD3+RHO
      ZFD3R = ZF+DEL3+RHO
      RPR = RHO*PHIR
      TIZ2 = RHMR2*PHIR2
      TG61 = XMUR*(DEL3D-RPR*PHIRD)
      PHRP = PHIR-PHITP
      TPR = 0.5*TR*PHIR
      RPHRD = R*PHIRD
  400 CONTINUE
    2 SPHTP = SIN(PHITP)
      CPHTP = COS(PHITP)
      TANTP = TAN(THETTP)
      CTHETP = COS(THETTP)
      SECTP = 1.0/CTHETP
      IF(XINDN)    7, 5, 7
    5 THETT  = THETTP
      PHIT   = PHITP
      PSIT   = PSITP + PSIN
      SINPS = SIN(PSIT)
      COSPS = COS(PSIT)
      GO TO 70
    7 IF(XINDN - XINDL)  9,11,9
C          COMPUTE BNMTX  ONCE AFTER EACH INDEXING ON THETMX
    9 XINDL = XINDN
C          IF THETAO OR PHIO .NE.0.0 COMPUTE BNMTX  ONCE AT T=TO
      COSTHN = COS(THETN)
      SINTHN = SIN(THETN)
```

```
      COSPHN = COS(PHIN)                                                TMCN1520
      SINPHN = SIN(PHIN)                                                TMCN1530
      COSPSN = COS(PSIN)                                                TMCN1540
      SINPSN = SIN(PSIN)                                                TMCN1550
      BNMTX (1,1)= COSTHN *COSPSN                                       TMCN1560
      BNMTX (2,1) = COSTHN*SINPSN                                       TMCN1570
      BNMTX (3,1) = -SINTHN                                             TMCN1580
      BNMTX (1,2) = -COSPHN*SINPSN + SINPHN*SINTHN*COSPSN               TMCN1590
      BNMTX (2,2) =  COSPHN*COSPSN + SINPHN*SINTHN*SINPSN               TMCN1600
      BNMTX (3,2) =  COSTHN*SINPHN                                      TMCN1610
      BNMTX (1,3) =  SINPHN*SINPSN + COSPHN*SINTHN*COSPSN               TMCN1620
      BNMTX (2,3) = -COSPSN*SINPHN + COSPHN*SINTHN*SINPSN               TMCN1630
      BNMTX (3,3) =  COSTHN*COSPHN                                      TMCN1640
   11 STHETP = SIN(THETTP)                                              TMCN1650
      SPSTP  = SIN(PSITP)                                               TMCN1660
      CPSTP  = COS(PSITP)                                               TMCN1670
      CNMTX (1,1) = CTHETP*CPSTP                                        TMCN1680
      CNMTX (2,1) = CTHETP*SPSTP                                        TMCN1690
      CNMTX (3,1) = -STHETP                                             TMCN1700
      TMP1   =   SPHTP * STHETP                                         TMCN1710
      TMP2   =   CPHTP * STHETP                                         TMCN1720
      CNMTX (1,2) =  -CPHTP*SPSTP  + TMP1*CPSTP                         TMCN1730
      CNMTX (2,2) =   CPHTP*CPSTP  + TMP1*SPSTP                         TMCN1740
      CNMTX (3,2) =   CTHETP*SPHTP                                      TMCN1750
      CNMTX (1,3) =  SPHTP*SPSTP   + TMP2*CPSTP                         TMCN1760
      CNMTX (2,3) = -CPSTP*SPHTP   + TMP2*SPSTP                         TMCN1770
      CNMTX (3,3) =  CTHETP*CPHTP                                       TMCN1780
C           COMPUTE CNMTX EACH R-K STEP IF XINDN.NE.0.0                 TMCN1790
C     ITRY, INDICATOR TO ALLOW ONE ADDITIONAL REVOLUTION FOR TRIAL ANGLETMCN1800
      ITRY = 0                                                          TMCN1810
C           IANG = 1  FOR PSIT, =2 FOR THETT,  =3 FOR PHIT  DETERMINATION TMCN1820
      IANG = 1                                                          TMCN1830
      ANGL = PSITL                                                      TMCN1840
      TMP3 =  BNMTX(2,1)*CNMTX(1,1) + BNMTX(2,2)*CNMTX(2,1)  +          TMCN1850
     X        BNMTX(2,3)*CNMTX(3,1)                                     TMCN1860
      TMP4 =        BNMTX(1,1)*CNMTX(1,1) + BNMTX(1,2)*CNMTX(2,1) +     TMCN1870
     X              BNMTX(1,3)*CNMTX(3,1)                               TMCN1880
C     NOTE, TANA AND ANGA=ATAN(TANA) NOT USED WHEN DENOMINATOR TANA ZEROTMCN1890
      IF(TMP4)  18,14,18                                                TMCN1900
   14 IF(TMP3) 15,16,17                                                 TMCN1910
   15 ANGA = - PIO2                                                     TMCN1920
      GO TO 21                                                          TMCN1930
   16 ISTOP = 4                                                         TMCN1940
      GO TO 64                                                          TMCN1950
C                                                                       TMCN1960
   17 ANGA =   PIO2                                                     TMCN1970
      GO TO 21                                                          TMCN1980
   18 TANA = TMP3/TMP4                                                  TMCN1990
C                                                                       TMCN2000
   20 ANGA = ATAN(TANA)                                                 TMCN2010
   21 NREV = ANGL/TWOPI  + SIGN(0.1 ,ANGL)                              TMCN2020
```

```
      FNREV = FLOAT(NREV)  * TWOPI                                      TMCN2030
   22 ANGTRY =  ANGA + FNREV                                            TMCN2040
      DIFFA  =  ANGTRY                                                  TMCN2050
      DIFFL =  DIFFA - ANGL                                             TMCN2060
      IF(ABS(DIFFL) - PI04) 40,40,25                                    TMCN2070
   25 DIFFA = ANGTRY + PI                                               TMCN2080
      DIFFL = DIFFA - ANGL                                              TMCN2090
      IF(ABS(DIFFL) - PI04) 40,40,27                                    TMCN2100
   27 DIFFA = ANGTRY - PI                                               TMCN2110
      DIFFL = DIFFA - ANGL                                              TMCN2120
      IF(ABS(DIFFL) - PI04) 40,40,29                                    TMCN2130
   29 IF(ANGTRY) 30,30,31                                               TMCN2140
   30 TWOPIA = TWOPI                                                    TMCN2150
      GO TO 32                                                          TMCN2160
   31 TWOPIA = - TWOPI                                                  TMCN2170
   32 DIFFA  =  ANGTRY + TWOPIA                                          TMCN2180
      DIFFL = DIFFA - ANGL                                              TMCN2190
      IF(ABS(DIFFL) - PI04) 40,40,33                                    TMCN2200
   33 IF (ITRY) 36,34,36                                                TMCN2210
   34 FNREV = FNREV + SIGN(TWOPI,ANGL)                                  TMCN2220
      ITRY = 1                                                          TMCN2230
C        ONCE ONLY, INCREASE FNREV BY ONE REVOLUTION AND TRY AGAIN      TMCN2240
      GO TO 22                                                          TMCN2250
   36 ISTOP = 5                                                         TMCN2260
      WRITE(6,1005) T,ANAME(IANG),ANGL, DIFFA ,ANGA,ANGTRY              TMCN2270
 1005 FORMAT( 7HO TIME=,F8.3,5X,A4,11H  PREVIOUS=,1PE13.5,6H, NEW=,E13.5TMCN2280
     X,12H, AS ARCTAN=, E13.5, 16H,  CORR.FOR REV=,E13.5 ,8H   STOP5)   TMCN2290
      GO TO 64                                                          TMCN2300
C                                                                       TMCN2310
   40 ITRY = 0                                                          TMCN2320
      IF(IANG-2) 41,50,59                                               TMCN2330
   41 IANG = 2                                                          TMCN2340
      PSIT = DIFFA                                                      TMCN2350
      SINPS  = SIN(PSIT)                                                TMCN2360
      COSPS = COS(PSIT)                                                 TMCN2370
      ANGL = THETTL                                                     TMCN2380
      TMP5   =        -(BNMTX(3,1)*CNMTX(1,1) + BNMTX(3,2)*CNMTX(2,1) + TMCN2390
     X                  BNMTX(3,3)*CNMTX(3,1) )                         TMCN2400
      IF( ABS(SINPS) - 0.7) 42,42,43                                    TMCN2410
   42 TMPP4 = TMP4/COSPS                                                TMCN2420
      IF (TMPP4) 49,44,49                                               TMCN2430
   43 TMPP4 = TMP3/SINPS                                                TMCN2440
      IF (TMPP4) 49,44,49                                               TMCN2450
   44 IF( TMP5) 45,46,47                                                TMCN2460
   45 ANGA = - PIO2                                                     TMCN2470
      GO TO 21                                                          TMCN2480
   46 ISTOP = 6                                                         TMCN2490
      GO TO 64                                                          TMCN2500
   47 ANGA = PIO2                                                       TMCN2510
      GO TO 21                                                          TMCN2520
   49 TANA = TMP5/TMPP4                                                 TMCN2530
```

```
      GO TO 20                                                          TMCN2540
   50 IANG = 3                                                          TMCN2550
      THETT  = DIFFA                                                    TMCN2560
      ANGL   = PHITL                                                    TMCN2570
      TMP6   =  BNMTX(3,1)*CNMTX(1,2) +  BNMTX(3,2)*CNMTX(2,2)  +       TMCN2580
     X          BNMTX(3,3)*CNMTX(3,2)                                   TMCN2590
      TMP7   =  BNMTX(3,1)*CNMTX(1,3) +  BNMTX(3,2)*CNMTX(2,3)  +       TMCN2600
     X          BNMTX(3,3)*CNMTX(3,3)                                   TMCN2610
      IF(TMP7)  55,51,55                                                TMCN2620
   51 IF(TMP6)  52,53,54                                                TMCN2630
   52 ANGA = - PIO2                                                     TMCN2640
      GO TO 21                                                          TMCN2650
   53 ISTOP = 7                                                         TMCN2660
      GO TO 64                                                          TMCN2670
   54 ANGA =   PIO2                                                     TMCN2680
      GO TO 21                                                          TMCN2690
   55 TANA =  TMP6/TMP7                                                 TMCN2700
      GO TO 20                                                          TMCN2710
   59 PHIT = DIFFA                                                      TMCN2720
C       AT  ST  70 HAVE NEW PSIT,THETT,PHIT                             TMCN2730
   70 CONTINUE                                                          TMCN2740
C          * * * * * * * * *                                            TMCN2750
      IF(THETT.GE.ASTOP .OR. PSIT  .GE.ASTOP) GO TO 60                  TMCN2760
      IF(PHIT .GE.ASTOP) GO TO 60                                       TMCN2770
C  70 COSTH = COS(THETT)                                                TMCN2780
C          * * * * * * * * *                                            TMCN2790
      COSTH = COS(THETT)                                                TMCN2800
      SINTH = SIN(THETT)                                                TMCN2810
C        COSPS,SINPS COMPUTED ABOVE EITHER AFTER ST 5 OR AFTER ST 41    TMCN2820
      COSPH = COS(PHIT)                                                 TMCN2830
      SINPH = SIN(PHIT)                                                 TMCN2840
    3 CONTINUE                                                          TMCN2850
      GCTH = G*COSTH                                                    TMCN2860
      GSTH = G*SINTH                                                    TMCN2870
      RETURN                                                            TMCN2880
C          * * * * * * * * *                                            TMCN2890
C       ISTOP.NE.0 CAUSES PRINTING OF OUTPUT UP TO CURRENT RUNGE-KUTTA  TMCN2900
C         INTERVAL, MESSAGE, AND TERMINATION OF THIS RUN AT END OF THIS TMCN2910
C         INTERVAL IN THE MAIN PROGRAM.                                 TMCN2920
C                                                                       TMCN2930
C                                                                       TMCN2940
   60 ISTOP = 30                                                        TMCN2950
C              AT 64 TEMPORARY ERROR STOP.                              TMCN2960
   64 CALL OUTPUT(2)                                                    TMCN2970
      CALL PLOTTP(3)                                                    TMCN2980
      WRITE(6,1006) T, ISTOP                                            TMCN2990
 1006 FORMAT(7H0 TIME=,F8.3,5X, 7H ISTOP=,I3,21H IN SUBROUTINE TMCNST)  TMCN3000
C     CALL ABDUMP                                                       TMCN3010
C     SUBR ABDUMP CAUSES 'ABNORMAL END' AND DUMP ON OUR OPERATING SYSTEMTMCN3020
      STOP                                                              TMCN3030
C            IF STOP IS CODED AS HERE, DOES NOT RETURN TO MAIN PROGRAM.  TMCN3040
```

```
C        * * * * * * * * *
C   64 WRITE(6,1006) T, ISTOP                                    TMCN3050
C      RETURN                                                    TMCN3060
       END                                                       TMCN3070
                                                                 TMCN3080
```

```
      SUBROUTINE TREAD(NCARD,NCRDS,NT,NDIM,ARRAY,NERR)
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION ARRAY(2),DUM(9)
      IF(NT.GT.NDIM) GO TO 90
      K = 0
      LSEQ = 0
      DO10 I=1,NCRDS
      READ(2,2000) (DUM(N),N=1,9),NSEQ,LCARD
 2000 FORMAT(9F8.0,2I4)
      IF(NCARD.NE.LCARD) GO TO 90
      IF(NSEQ.LE.LSEQ) GO TO 90
      LSEQ = NSEQ
      DO 20 N=1,9
      K = K+1
      ARRAY(K) = DUM(N)
      IF(K.GE.NT) GO TO 91
   20 CONTINUE
   10 CONTINUE
   91 RETURN
   90 NERR = 1
      RETURN
      END
```

```
      SUBROUTINE T2READ(NCARD,ND1,NI,NJ,ARRAY,NERR)
C             HVOSM-RD2 VERSION
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION ARRAY(ND1,NJ),DUM(9)
      LSEQ = 0
      NICRDS = (NI-1)/9 + 1
      DO 30 J=1,NJ
      K = 0
      DO 20 I=1,NICRDS
      READ(2,2000) (DUM(N),N=1,9),NSEQ,LCARD
 2000 FORMAT(9F8.0,2I4)
      IF(NCARD.NE.LCARD) GO TO 90
      IF(NSEQ.LE.LSEQ) GO TO 90
      LSEQ = NSEQ
      DO 10 N=1,9
      K = K+1
      ARRAY(K,J) = DUM(N)
      IF(K.GE.NI) GO TO 30
   10 CONTINUE
   20 CONTINUE
   30 CONTINUE
      RETURN
   90 NERR = 1
      RETURN
      END
```

```
      SUBROUTINE UMOMNT(IS)                                             UMOM0010
C              HVOSM-RD2 VERSION                                        UMOM0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               UMOM0030
C   SUBROUTINE TO COMPUTE THE MOMENTS ACTING ON THE SPRUNG AND          UMOM0040
C   UNSPRUNG MASSES RESULTING FROM TIRE FORCES AND SUSPENSIN FORCES.    UMOM0050
C                                                                       UMOM0060
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    UMOM0070
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      UMOM0080
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      UMOM0090
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,UMOM0100
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  UMOM0110
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     UMOM0120
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   UMOM0130
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   UMOM0140
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    UMOM0150
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  UMOM0160
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),UMOM0170
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  UMOM0180
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,UMOM0190
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),    UMOM0200
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),UMOM0210
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),      UMOM0220
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),   UMOM0230
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),   UMOM0240
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),      UMOM0250
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),       UMOM0260
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),UMOM0270
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)UMOM0280
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      UMOM0290
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),      UMOM0300
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),    UMOM0310
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                   UMOM0320
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       UMOM0330
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   UMOM0340
     1           (PSII(1),PSI1)                                         UMOM0350
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,UMOM0360
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          UMOM0370
     2           TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,   UMOM0380
     3           BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,     UMOM0390
     4           XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,     UMOM0400
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPUMOM0410
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,      UMOM0420
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,     UMOM0430
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,   UMOM0440
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ    UMOM0450
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  UMOM0460
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,   UMOM0470
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,UMOM0480
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2UMOM0490
```

```
    4              ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,UMOM0500
    5               XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             UMOM0510
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                   UMOM0520
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)         UMOM0530
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,         UMOM0540
    1                AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),UMOM0550
    2                NCAMF,NCAMR,NDTHF,NDTHR                              UMOM0560
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,        UMOM0570
    1                 RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,       UMOM0580
    2                 DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,      UMOM0590
    3                 PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,      UMOM0600
    4                 PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,          UMOM0610
    5                 DTDD2,DTDD3,DTDD4,FJF(4),SNPF                       UMOM0620
C                                                                         UMOM0630
      IS1 = IS+1                                                          UMOM0640
      GO TO (10,20,30),IS1                                                UMOM0650
C                                                                         UMOM0660
C  MOMENTS FOR SUSPENSION OPTION 0 , INDEPENDENT FRONT, SOLID AXLE REAR UMOM0670
C                                                                         UMOM0680
   10 TERM1 = ZFD1+HCGH1                                                  UMOM0690
      TERM2 = ZFD2+HCGH2                                                  UMOM0700
C                                                                         UMOM0710
C  ROLL MOMENT                                                            UMOM0720
C                                                                         UMOM0730
      SNPU = -FYU(1)*TERM1 - FYU(2)*TERM2 - (FYU(3)+FYU(4))*ZRD3          UMOM0740
    1        +SI(2)*(TFO2+DTHF2) - SI(1)*(TFO2+DTHF1)                     UMOM0750
    2        +(SI(4)-SI(3))*TSO2                                          UMOM0760
C                                                                         UMOM0770
C  PITCH MOMENT                                                           UMOM0780
C                                                                         UMOM0790
      SNTU =  (SI(1)+SI(2))*A - (SI(3)+SI(4))*B                           UMOM0800
    1        +FXU(1)*TERM1 + FXU(2)*TERM2                                 UMOM0810
    2        +FXU(3)*(ZRD3R+TPR+HCGH3) + FXU(4)*(ZRD3R-TPR+HCGH4)         UMOM0820
C                                                                         UMOM0830
C  YAW MOMENT                                                             UMOM0840
C                                                                         UMOM0850
      SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)                         UMOM0860
    1        -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)                         UMOM0870
    2        -FXU(1)*(TFO2+DTHF1+HCBH1) + FXU(2)*(TFO2+DTHF2-HCBH2)       UMOM0880
    3        -FXU(3)*(TRO2-RPR+HCBH3) + FXU(4)*(TRO2+RPR-HCBH4)           UMOM0890
C                                                                         UMOM0900
C  REAR AXLE ROLL MOMENT                                                  UMOM0910
C                                                                         UMOM0920
      SNPR =  FZU(3)*(TRO2-RPR+HCBH3) - FZU(4)*(TRO2+RPR-HCBH4)           UMOM0930
    1        -FYU(3)*(RHO+TPR+HCGH3)  - FYU(4)*(RHO-TPR+HCGH4)            UMOM0940
    2        +(SI(3)-SI(4))*TSO2                                          UMOM0950
      RETURN                                                              UMOM0960
C                                                                         UMOM0970
C  MOMENTS FOR SUSPENSION OPTION 1, INDEPENDENT FRONT AND REAR            UMOM0980
C                                                                         UMOM0990
   20 TERM1 = ZFD1+HCGH1                                                  UMOM1000
```

```
      TERM2 = ZFD2+HCGH2
      TERM3 = ZRD3+HCGH3
      TERM4 = ZRD4+HCGH4
C
C  ROLL MOMENT
C
      SNPU =  SI(2)*(TFO2+DTHF2) - SI(1)*(TFO2+DTHF1)
     1        +SI(4)*(TRO2+DTHR4) - SI(3)*(TRO2+DTHR3)
     2        -FYU(1)*TERM1 - FYU(2)*TERM2 - FYU(3)*TERM3 - FYU(4)*TERM4
C
C  PITCH MOMENT
C
      SNTU = (SI(1)+SI(2))*A - (SI(3)+SI(4))*B
     1        +FXU(1)*TERM1 + FXU(2)*TERM2 + FXU(3)*TERM3 + FXU(4)*TERM4
C
C  YAW MOMENT
C
      SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)
     1        -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)
     2        -FXU(1)*(TFO2+DTHF1+HCBH1) + FXU(2)*(TFO2+DTHF2-HCBH2)
     3        -FXU(3)*(TRO2+DTHR3+HCBH3) + FXU(4)*(TRO2+DTHR4-HCBH4)
      RETURN
C
C  MOMENTS FOR SUSPENSION OPTION 2, SOLID FRONT AND REAR AXLES
C
C  ROLL MOMENT
C
   30 SNPU = -(FYU(1)+FYU(2))*ZFD1 - (FYU(3)+FYU(4))*ZRD3
     1        +(SI(2)-SI(1))*TSFO2  + (SI(4)-SI(3))*TSO2
C
C  PITCH MOMENT
C
      SNTU =  (SI(1)+SI(2))*A - (SI(3)+SI(4))*B
     1        +FXU(1)*(ZFD1RF+TPF+HCGH1) + FXU(2)*(ZFD1RF-TPF+HCGH2)
     2        +FXU(3)*(ZRD3R+TPR+HCGH3)  + FXU(4)*(ZRD3R-TPR+HCGH4)
C
C  YAW MOMENT
C
      SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)
     1        -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)
     2        -FXU(1)*(TFO2-RFPF+HCBH1) + FXU(2)*(TFO2+RFPF-HCBH2)
     3        -FXU(3)*(TRO2-RPR+HCBH3)  + FXU(4)*(TRO2+RPR-HCBH4)
C
C  FRONT AXLE ROLL MOMENT
C
      SNPF =  FZU(1)*(TFO2-RFPF+HCBH1) - FZU(2)*(TFO2+RFPF-HCBH2)
     1        -FYU(1)*(RHOF+TPF+HCGH1)  - FYU(2)*(RHOF-TPF+HCGH2)
     2        +(SI(1)-SI(2))*TSFO2
C
C  REAR AXLE ROLL MOMENT
C
```

```
 SNPR =  FZU(3)*(TRO2-RPR+HCBH3) - FZU(4)*(TRO2+RPR-HCBH4)        UMOM1520
1        -FYU(3)*(RHO+TPR+HCGH3) - FYU(4)*(RHO-TPR+HCGH4)         UMOM1530
2        +(SI(3)-SI(4))*TSO2 -                                    UMOM1540
 RETURN                                                           UMOM1550
 END                                                              UMOM1560
```

```
      SUBROUTINE VGORNT                                                 VGOR0010
C             HVOSM-RD2 VERSION                                         VGOR0020
C             REVISED OCTOBER 1975  CALSPAN CORPORATION                 VGOR0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    VGOR0040
     1             A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,    VGOR0050
     2             PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                    VGOR0060
     3             XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,VGOR0070
     4             RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, VGOR0080
     5             T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,   VGOR0090
     6             HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB, VGOR0100
     7             DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR, VGOR0110
     8             NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),  VGOR0120
     9             NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                VGOR0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),VGOR0140
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN VGOR0150
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          VGOR0160
     1              CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB, VGOR0170
     2              PSIFIO,PSIFDO                                       VGOR0180
      DIMENSION YCIP(2)                                                 VGOR0190
      EQUIVALENCE (YCIP(1),YC1P)                                        VGOR0200
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             VGOR0210
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))VGOR0220
     1         ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),VGOR0230
     2             (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),      VGOR0240
     3             (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),     VGOR0250
     4             (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),       VGOR0260
     5             (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),         VGOR0270
     6             (PSIFID,VAR(22))                                     VGOR0280
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      VGOR0290
     1             (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))VGOR0300
     2            ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),     VGOR0310
     3             (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),   VGOR0320
     4             (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),  VGOR0330
     5             (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),        VGOR0340
     6             (DPSIFI,DER(21)),(DDPSFI,DER(22))                    VGOR0350
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         VGOR0360
     1             (DER(10),DPHIFD)                                     VGOR0370
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       VGOR0380
     1             (DER(14),DDEL4D)                                     VGOR0390
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,VGOR0400
     1              PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4), VGOR0410
     2              CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),VGOR0420
     3              STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),   VGOR0430
     4              XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), VGOR0440
     5              YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), VGOR0450
     6              CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),   VGOR0460
     7              CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),    VGOR0470
     8              SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),VGOR0480
     9              FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)VGOR0490
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      VGOR0500
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     VGOR0510
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   VGOR0520
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  VGOR0530
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       VGOR0540
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   VGOR0550
     1            (PSII(1),PSI1)                                        VGOR0560
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,VGOR0570
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         VGOR0580
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, VGOR0590
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   VGOR0600
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   VGOR0610
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPVGOR0620
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    VGOR0630
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   VGOR0640
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, VGOR0650
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  VGOR0660
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  VGOR0670
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, VGOR0680
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,VGOR0690
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PH1R2VGOR0700
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,VGOR0710
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          VGOR0720
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 VGOR0730
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       VGOR0740
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     VGOR0750
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   VGOR0760
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   VGOR0770
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       VGOR0780
      LOGICAL LCB1,LCB2                                                 VGOR0790
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                VGOR0800
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       VGOR0810
     2              SLOPE1,SLOPE2,XTRA(300)                             VGOR0820
      DIMENSION UI(4),VI(4),WI(4)                                       VGOR0830
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                      VGOR0840
      DIMENSION APITCH(4)                                               VGOR0850
      EQUIVALENCE (APITCH(1),APTCH1)                                    VGOR0860
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   VGOR0870
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     VGOR0880
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          VGOR0890
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       VGOR0900
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),VGOR0910
     2               NCAMF,NCAMR,NDTHF,NDTHR                            VGOR0920
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,      VGOR0930
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     VGOR0940
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    VGOR0950
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    VGOR0960
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        VGOR0970
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     VGOR0980
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          VGOR0990
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                          VGOR1000
```

```
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,
     5              YCMP(6),ZCMP(6),PHICM(6)
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF
    1 DO 17  I=1,4
      XCPHI = COS(PHII(I))
      XSPHI = SIN(PHII(I))
      XCPSI = COS(PSII(I))
      XSPSI = SIN(PSII(I))
      TMP4 =   XCPHI * XCPSI
      TMP3 =   XSPHI * XCPSI
    2 CAYW(I) =-AMTX(1,1)*XSPSI+ AMTX(1,2)*TMP4 + AMTX(1,3)*TMP3
      CBYW(I) =-AMTX(2,1)*XSPSI+ AMTX(2,2)*TMP4 + AMTX(2,3)*TMP3
      CGYW(I) =-AMTX(3,1)*XSPSI+ AMTX(3,2)*TMP4 + AMTX(3,3)*TMP3
      IF(INDCRB.LE.0) GO TO 3
      LCB1(I) = RW(I).GT.YC1P-YP(I)
      LCB2(I) = RW(I).LE.YP(I)-YCLP
      IF(ICBHIT.EQ.0) GO TO 3
      PHGI(I) = 0.0
      THGI(I) = 0.0
      ZPGI(I) = 0.0
      SPG(I) = 0.0
      CPG(I) = 1.0
      STG(I) = 0.0
      CTG(I) = 1.0
      IF(.NOT.LCB2(I)) GO TO 4
      ZPGI(I) = ZCLP+(YP(I)-YCLP)*TANPCL
      PHGI(I) = PHICLR
      SPG(I) = SIN(PHGI(I))
      CPG(I) = COS(PHGI(I))
      GO TO 30
C         INTRP5 LOOKS UP THGI, PHGI, ZPGI, AND XMUGI FOR EACH WHEEL.
    3 IF(IRUF.EQ.0) GO TO 31
      IF(XP(I)+RW(I).LT.0.0.OR.XP(I)-RW(I).GT.DGMAX) GO TO 31
      CALL RUFFRC(I,ZGP)
      XMUGI(I) = AMU(I)
      GO TO 33
   31 CALL INTRP5(I)
   32 CPG(I) = COS(PHGI(I))
      SPG(I) = SIN(PHGI(I))
      CTG(I) = COS(THGI(I))
      STG(I) = SIN(THGI(I))
   30 CAGZ(I) = CPG(I)*STG(I)
      CBGZ(I) = -SPG(I)
      CGGZ(I) = CTG(I)*CPG(I)
      P1 = CBYW(I)*CGGZ(I)
      P7 = CBGZ(I)*CGYW(I)
      P3 = CGYW(I)*CAGZ(I)
      P4 = CGGZ(I)*CAYW(I)
      P5 = CAYW(I)*CBGZ(I)
```

```
      P6 = CAGZ(I)*CBYW(I)
      D1(I) = P1-P7
      D2(I) = P3-P4
      D3(I) = P5-P6
      CALL GCP(I)
C             XMUGI(I) IS SET IN INTRP5
C             IF ICBHIT.NE.0 AND LCB1(I) AND LCB2(I) BOTH  FALSE,  XMUGI(I)
C              IS NOT SET IN THIS INTERVAL.  RETAINS LAST VALUE, SHOULD BE
C                   FOR FLAT TERRAIN. (RADIAL SPRING TIRE MODE IN CRBIMP
C              REQUIRES FLAT TERRAIN  PREVIOUS TO CURB HIT.)
      GO TO 5
    4 IF(.NOT.LCB1(I))GO TO 30
      CALL CRBIMP(I)
      XMUGI(I) = AMUC*AMU(I)
   33 CAGZ(I) = CPG(I)*STG(I)
      CBGZ(I) = -SPG(I)
      CGGZ(I) = CTG(I)*CPG(I)
      PI = CBYW(I)*CGGZ(I)
      P7 = CBGZ(I)*CGYW(I)
      P3 = CGYW(I)*CAGZ(I)
      P4 = CGGZ(I)*CAYW(I)
      P5 = CAYW(I)*CBGZ(I)
      P6 = CAGZ(I)*CBYW(I)
      D1(I) = P1-P7
      D2(I) = P3-P4
      D3(I) = P5-P6
    5 CAH(I) = AMTX(1,1)*CAR(I)+AMTX(2,1)*CBR(I)+AMTX(3,1)*CGR(I)
      CBH(I) = AMTX(1,2)*CAR(I)+AMTX(2,2)*CBR(I)+AMTX(3,2)*CGR(I)
      CGH(I) = AMTX(1,3)*CAR(I)+AMTX(2,3)*CBR(I)+AMTX(3,3)*CGR(I)
      TI(I) = 12.0*TI(I)/HI(I)
      HCAH(I) = HI(I)*CAH(I)
      HCBH(I) = HI(I)*CBH(I)
      HCGH(I) = HI(I)*CGH(I)
   17 CONTINUE
C
      IF(ISUS.NE.0) GO TO 90
      V1 = V+A*R-ZFD1*P-HCGH1*(P+PHI1D)+DTDD1*DEL1D
      V2 = V+A*R-ZFD2*P-HCGH2*(P+PHI2D)-DTDD2*DEL2D
      V3 = V-B*R-ZRD3*P-(RHO+TPR+HCGH3)*(P+PHIRD)
      V4 = V-B*R-ZRD3*P-(RHO-TPR+HCGH4)*(P+PHIRD)
      W1 = W-A*Q+(TFO2+DTHF1)*P+DEL1D+HCBH1*(P+PHI1D)
      W2 = W-A*Q-(TFO2+DTHF2)*P+DEL2D+HCBH2*(P+PHI2D)
      W3 = W+B*Q+DEL3D-(RPR-TRO2-HCBH3)*(P+PHIRD)
      W4 = W+B*Q+DEL3D-(RPR+TRO2-HCBH4)*(P+PHIRD)
      GO TO 95
   90 IF(ISUS.EQ.2) GO TO 91
      V1 = V+A*R-ZFD1*P-HCGH1*(P+PHI1D)+DTDD1*DEL1D
      V2 = V+A*R-ZFD2*P-HCGH2*(P+PHI2D)-DTDD2*DEL2D
      V3 = V-B*R-ZRD3*P-HCGH3*(P+PHI3D)+DTDD3*DEL3D
      V4 = V-B*R-ZRD4*P-HCGH4*(P+PHI4D)-DTDD4*DEL4D
      W1 = W-A*Q+(TFO2+DTHF1)*P+DEL1D+HCBH1*(P+PHI1D)
```

```
      W2 = W-A*Q-(TFO2+DTHF2)*P+DEL2D+HCBH2*(P+PHI2D)
      W3 = W+B*Q+(TRO2+DTHR3)*P+DEL3D+HCBH3*(P+PHI3D)
      W4 = W+B*Q-(TRO2+DTHR4)*P+DEL4D+HCBH4*(P+PHI4D)
      GO TO 95
   91 V1 = V+A*R-ZFD1*P-(RHOF+TPF+HCGH1)*(P+PHIFD)
      V2 = V+A*R-ZFD1*P-(RHOF-TPF+HCGH2)*(P+PHIFD)
      V3 = V-B*R-ZRD3*P-(RHO+TPR+HCGH3)*(P+PHIRD)
      V4 = V-B*R-ZRD3*P-(RHO-TPR+HCGH4)*(P+PHIRD)
      W1 = W-A*Q+DEL1D-(RFPF-TFO2-HCBH1)*(P+PHIFD)
      W2 = W-A*Q+DEL1D-(RFPF+TFO2-HCBH2)*(P+PHIFD)
      W3 = W+B*Q+DEL3D-(RPR-TRO2-HCBH3)*(P+PHIRD)
      W4 = W+B*Q+DEL3D-(RPR+TRO2-HCBH4)*(P+PHIRD)
C
   95 DO 170 I=1,4
   10 AX(I) = CBY*CGGZ(I)-CGY*CBGZ(I)
      BX(I) = CGY*CAGZ(I)-CAY*CGGZ(I)
      CX(I) = CAY*CBGZ(I)-CBY*CAGZ(I)
      DISTX = SQRT(AX(I)**2+BX(I)**2+CX(I)**2)
      CTXG(I) = (CAX*AX(I)+CBX*BX(I)+CGX*CX(I))/DISTX
      CTXG(I) = SIGN(AMIN1(ABS(CTXG(I)),1.0),CTXG(I))
      STXG(I) = SIGN(SQRT(1.0-CTXG(I)**2),CGX*DISTX-CX(I))
      UG(I) = UI(I)*CTXG(I)-WI(I)*STXG(I)
   11 AY(I) = CGX*CBGZ(I)-CBX*CGGZ(I)
      BY(I) = CAX*CGGZ(I)-CGX*CAGZ(I)
      CY(I) = CBX*CAGZ(I)-CAX*CBGZ(I)
      DISTY = SQRT(AY(I)**2+BY(I)**2+CY(I)**2)
   12 CPYG(I) = (CAY*AY(I)+CBY*BY(I)+CGY*CY(I))/DISTY
      CPYG(I) = SIGN(AMIN1(ABS(CPYG(I)),1.0),CPYG(I))
      SPYG(I) = SIGN(SQRT(1.0-CPYG(I)**2),CGY*DISTY-CY(I))
      VG(I) = VI(I)*CPYG(I)-WI(I)*SPYG(I)
      DISTD = SQRT(D1(I)**2+D2(I)**2+D3(I)**2)
   13 CAZW = - AMTX(1,2)*XSPHI + AMTX(1,3)*XCPHI
      CBZW = - AMTX(2,2)*XSPHI + AMTX(2,3)*XCPHI
      CGZW = - AMTX(3,2)*XSPHI + AMTX(3,3)*XCPHI
      PSIIP(I) = PSII(I)*(CAGZ(I)*CAZW+CBGZ(I)*CBZW+CGGZ(I)*CGZW)
   14 CAC(I) = D1(I)/DISTD
      CBC(I) = D2(I)/DISTD
      CGC(I) = D3(I)/DISTD
C           CAXW(I),CBXW(I),CGXW(I)  NO LONGER USED
   15 AS(I) = CGC(I)*CBGZ(I)-CBC(I)*CGGZ(I)
      BS(I) = CAC(I)*CGGZ(I)-CGC(I)*CAGZ(I)
      CS(I) = CBC(I)*CAGZ(I)-CAC(I)*CBGZ(I)
      DISTS = SQRT(AS(I)**2+BS(I)**2+CS(I)**2)
      CAS(I) = AS(I)/DISTS
      CBS(I) = BS(I)/DISTS
      CGS(I) = CS(I)/DISTS
   16 CALL TIRFRC(I)
  170 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VPOS                                                   VPOS0010
C           HVOSM-RD2 VERSION                                           VPOS0020
C           REVISED OCTOBER 1975   CALSPAN CORPORATION                  VPOS0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    VPOS0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      VPOS0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      VPOS0060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,VPOS0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  VPOS0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     VPOS0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   VPOS0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   VPOS0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    VPOS0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  VPOS0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),VPOS0140
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN   VPOS0150
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          VPOS0160
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  VPOS0170
     2             PSIFIO,PSIFDO                                        VPOS0180
      DIMENSION YCIP(2)                                                 VPOS0190
      EQUIVALENCE (YCIP(1),YC1P)                                        VPOS0200
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             VPOS0210
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))VPOS0220
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),VPOS0230
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        VPOS0240
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       VPOS0250
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         VPOS0260
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           VPOS0270
     6           (PSIFID,VAR(22))                                       VPOS0280
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      VPOS0290
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))VPOS0300
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       VPOS0310
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     VPOS0320
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    VPOS0330
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          VPOS0340
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      VPOS0350
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         VPOS0360
     1           (DER(10),DPHIFD)                                       VPOS0370
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       VPOS0380
     1           (DER(14),DDEL4D)                                       VPOS0390
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,VPOS0400
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  VPOS0410
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),VPOS0420
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    VPOS0430
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), VPOS0440
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), VPOS0450
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    VPOS0460
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     VPOS0470
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),VPOS0480
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)VPOS0490
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),        VPOS0500
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),       VPOS0510
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1F1(2),F1RI(2),     VPOS0520
     3            F2F1(2),F2RI(2),CAH(4),CBH(4),CGH(4)                    VPOS0530
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                         VPOS0540
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),     VPOS0550
     1            (PSII(1),PSI1)                                          VPOS0560
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,VPOS0570
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,            VPOS0580
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,    VPOS0590
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,      VPOS0600
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,      VPOS0610
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPVPOS0620
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,       VPOS0630
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,      VPOS0640
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,    VPOS0650
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ     VPOS0660
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,    VPOS0670
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,    VPOS0680
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,VPOS0690
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2VPOS0700
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,VPOS0710
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             VPOS0720
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                   VPOS0730
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)         VPOS0740
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                       VPOS0750
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,     VPOS0760
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),     VPOS0770
     3               SFRX(4),SFRY(4),SFRZ(4),T1PS1,T2PSI,XMUGI(4)         VPOS0780
      LOGICAL LCB1,LCB2                                                   VPOS0790
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                  VPOS0800
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,         VPOS0810
     2              SLOPE1,SLOPE2,XTRA(300)                               VPOS0820
      DIMENSION UI(4),VI(4),WI(4)                                         VPOS0830
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                        VPOS0840
      DIMENSION APITCH(4)                                                 VPOS0850
      EQUIVALENCE (APITCH(1),APTCH1)                                      VPOS0860
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,         VPOS0870
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),VPOS0880
     2               NCAMF,NCAMR,NDTHF,NDTHR                              VPOS0890
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,        VPOS0900
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,       VPOS0910
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,      VPOS0920
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,      VPOS0930
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,          VPOS0940
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                       VPOS0950
    1 TI(1) = 0.0                                                         VPOS0960
      IF(NTBL2.NE.0) CALL INTRPL(TQF,TB,TE,TINCR,T,TI(1))                 VPOS0970
      TI(2) = TI(1)                                                       VPOS0980
      TI(3) = 0.0                                                         VPOS0990
      IF(NTBL3.NE.0) CALL INTRPL(TQR,TB,TE,TINCR,T,TI(3))                 VPOS1000
```

```
      TI(4) = TI(3)
C
      IS1 = ISUS+1
C
C  LONGITUDINAL WHEEL CENTER VELOCITIES
C
      GO TO (10,11,12),IS1
C
C  SUSPENSION OPTION 0, INDEPENDENT FRONT AND SOLID AXLE REAR
C
   10 IF(NDTHF.EQ.0) GO TO 101
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL1,DTHF1,DTDD1)
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL2,DTHF2,DTDD2)
  101 U1 = U-(TFO2+DTHF1)*R+ZFD1*Q
      U2 = U+(TFO2+DTHF2)*R+ZFD2*Q
      U3 = U-(TRO2-RPR)*R+(ZRD3R+TPR)*Q
      U4 = U+(TRO2+RPR)*R+(ZRD3R-TPR)*Q
      GO TO 13
C
C  SUSPENSION OPTION 1, INDEPENDENT FRONT AND REAR
C
   11 IF(NDTHF.EQ.0) GO TO 111
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL1,DTHF1,DTDD1)
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL2,DTHF2,DTDD2)
  111 IF(NDTHR.EQ.0) GO TO 112
      CALL INTRPC(DTHR,DELB,DELE,DDEL,DEL3,DTHR3,DTDD3)
      CALL INTRPC(DTHR,DELB,DELE,DDEL,DEL4,DTHR4,DTDD4)
  112 U1 = U-(TFO2+DTHF1)*R+ZFD1*Q
      U2 = U+(TFO2+DTHF2)*R+ZFD2*Q
      U3 = U-(TRO2+DTHR3)*R + ZRD3*Q
      U4 = U+(TRO2+DTHR4)*R + ZRD4*Q
      GO TO 13
C
C  SUSPENSION OPTION 2, SOLID FRONT AND REAR AXLES
C
   12 U1 = U-(TFO2-RFPF)*R+(ZFD1RF+TPF)*Q
      U2 = U+(TFO2+RFPF)*R+(ZFD1RF-TPF)*Q
      U3 = U-(TRO2-RPR)*R +(ZRD3R+TPR)*Q
      U4 = U+(TRO2+RPR)*R +(ZRD3R-TPR)*Q
   13 CONTINUE
      SFYU = 0.0
      SFXU = 0.0
      SFYUF = 0.0
      SFYUR = 0.0
C             SFYUF AND SFYUR NO LONGER USED
      SFZU = 0.0
    2 AMTX(1,1) = COSTH*COSPS
      AMTX(2,1) = COSTH*SINPS
      AMTX(3,1) = -SINTH
      AMTX(1,2) = -COSPH*SINPS+SINPH*SINTH*COSPS
      AMTX(2,2) = COSPH*COSPS+SINPH*SINTH*SINPS
```

```
      AMTX(3,2) = COSTH*SINPH
      AMTX(1,3) = SINPH*SINPS+COSPH*SINTH*COSPS
      AMTX(2,3) = -COSPS*SINPH+COSPH*SINTH*SINPS
      AMTX(3,3) = COSTH*COSPH
      CAY = AMTX(1,2)
      CBY = AMTX(2,2)
      CGY = AMTX(3,2)
      CAX = AMTX(1,1)
      CBX = AMTX(2,1)
      CGX = AMTX(3,1)
C
      IF(ISUS.EQ.2) GO TO 21
      YTMP = TFO2+DTHF1
      ZTMP = ZFD1
      GO TO 31
   21 YTMP = TFO2-RFPF
      ZTMP = ZFO+DEL1+TPF
   31 X1P = XCP+AMTX(1,1)*A+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP
      Y1P = YCP+AMTX(2,1)*A+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP
      Z1P = ZCP+AMTX(3,1)*A+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP
      IF(ISUS.EQ.2) GO TO 22
      YTMP = -TFO2-DTHF2
      ZTMP = ZFD2
      GO TO 32
   22 YTMP = -TFO2-RFPF
      ZTMP = ZFO+DEL2-TPF
   32 X2P = XCP+AMTX(1,1)*A+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP
      Y2P = YCP+AMTX(2,1)*A+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP
      Z2P = ZCP+AMTX(3,1)*A+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP
      IF(ISUS.EQ.1) GO TO 23
      YTMP = TRO2-RPR
      ZTMP = ZRO+DEL3+TPR
      GO TO 33
   23 YTMP = TRO2+DTHR3
      ZTMP = ZRD3
   33 X3P = XCP-AMTX(1,1)*B+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP
      Y3P = YCP-AMTX(2,1)*B+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP
      Z3P = ZCP-AMTX(3,1)*B+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP
      IF(ISUS.EQ.1) GO TO 24
      YTMP = -TRO2-RPR
      ZTMP = ZRO+DEL3-TPR
      GO TO 34
   24 YTMP = -TRO2-DTHR4
      ZTMP = ZRD4
   34 X4P = XCP-AMTX(1,1)*B+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP
      Y4P = YCP-AMTX(2,1)*B+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP
      Z4P = ZCP-AMTX(3,1)*B+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP
C
C     QUADRATIC INTERPOLATION SUBROUTINE INTRPL,ADDITIONAL ENTRY INTRPC
C
C
```

```
      IF(ISUS.EQ.2) GO TO 50
      CALL INTRPC(PHIC,DELB,DELE,DDEL,DEL1,PHI1,SLOPE1)
      PHI1 = PHI1*RAD
      SLOPE1 = SLOPE1*RAD
      PHI1D = SLOPE1*DEL1D
      CALL INTRPC(PHIC,DELB,DELE,DDEL,DEL2,PHI2,SLOPE2)
      PHI2 = -PHI2*RAD
      SLOPE2 = -SLOPE2*RAD
      PHI2D = SLOPE2*DEL2D
      GO TO 51
   50 PHI1 = PHIF
      PHI2 = PHIF
      PHI1D = PHIFD
      PHI2D = PHIFD
   51 IF(ISUS.EQ.1) GO TO 52
      PHI3 = PHIR
      PHI4 = PHIR
      PHI3D = PHIRD
      PHI4D = PHIRD
      GO TO 53
   52 CALL INTRPC(PHIRC,DELB,DELE,DDEL,DEL3,PHI3,SLOPE3)
      PHI3 = PHI3*RAD
      SLOPE3 = SLOPE3*RAD
      PHI3D = SLOPE3*DEL3D
      CALL INTRPC(PHIRC,DELB,DELE,DDEL,DEL4,PHI4,SLOPE4)
      PHI4 = -PHI4*RAD
      SLOPE4 = -SLOPE4*RAD
      PHI4D = SLOPE4*DEL4D
   53 CONTINUE
C
   40 IF(INDCRB.EQ.0) GO TO 5
      IF(IHIT.EQ.1.OR.INDCRB.LT.0) GO TO 6
    5 CALL DRIVER(PSICON,PSISLP,J)
      IF(J.NE.0) GO TO 5001
      PSICON = 0.0
      PSISLP = 0.0
      IF(NTBL1.NE.0) CALL INTRPC(PSIF,TB,TE,TINCR,T,PSICON,PSISLP)
      PSICON = PSICON*RAD
      PSISLP=PSISLP*RAD
 5001 CONTINUE
C     FORMERLY PSIFP=PSI1,NO LONGER USED.FORMERLY PSIFID=(PSI1-PSIFP)/DT
      PSI1 = PSICON
      PSIFID = PSISLP
      PSIFI = PSI1
      GO TO 7
    6 PSI1 = PSIFI
    7 PSI2 = PSI1
C
      IF(ISUS.EQ.1) GO TO 54
      PSI3 = AKRS*PHIR
      PSI4 = PSI3
```

```
      RETURN                                                    VPOS2540
   54 PSI3 = AKDS+AKDS1*DEL3+AKDS2*DEL3**2+AKDS3*DEL3**3         VPOS2550
      PSI4 = -(AKDS+AKDS1*DEL4+AKDS2*DEL4**2+AKDS3*DEL4**3)      VPOS2560
      RETURN                                                    VPOS2570
      END                                                       VPOS2580
```

```
      SUBROUTINE WHEEL(/AKT/,/SIGT/,/XLAMT/,/RWHJB/,/RWHJE/,/DRWHJ/,
     1 /NFJP/,/RW/,FJP,/NO/)
C             HVOSM-RD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION FJP(50)
    1 DA = 4.0*0.01745
      FJP(1) = 0.0
      N = NFJP
      IF (N.LE.NO) GO TO 3
      PRINT 2,N,NO
2       FORMAT ('ODIM. FOR FJP TOO SMALL,',I6,' NEEDED.', I6,' PROVIDED.
     1 ')
      STOP
3     CONTINUE
      NL = N-1
      DD = (RWHJE-RWHJB)/FLOAT(NL)
      DDK = DD*AKT
      K = 0
      D = 0.0
      DO 10  J=2,N
      FJP(J) = FJP(J-1)+DDK
      D = D+DD
      IF (K.NE.0) GO TO 10
      IF (D.LT.SIGT) GO TO 10
      X = DDK
      DDK = DDK*XLAMT
      FJP(J) = FJP(J)+(DDK-X)*(D-SIGT)/DD
      K = 1
   10 CONTINUE
   15 R = RW
      DO 19  J=2,N
      B = 1.0
      DDK = DD/R
      Z=DDK
  200 ANG = 0.0
      F = Z*B
  201 ANG = ANG+DA
      Y=1-Z
      X = COS(ANG)
      IF(X.LE.Y) GO TO 16
      F = F+2.0*(X-Y)*B
      GO TO 201
   16 B = FJP(J)/F
      FJP(J) = DDK*B
      IF (J.EQ.N) GO TO 1901
      I=J+1
      DO 18  L=I,N
      Z=Z+DDK
  300 ANG = 0.0
```

```
      F = Z*B                                                           WHEE0500
  301 ANG = ANG+DA                                                      WHEE0510
      Y=1-Z                                                             WHEE0520
      X = COS(ANG)                                                      WHEE0530
      IF(X.LE.Y) GO TO 18                                               WHEE0540
      F = F+2.0*(X-Y)*B                                                 WHEE0550
      GO TO 301                                                         WHEE0560
   18 FJP(L) = FJP(L)-F                                                 WHEE0570
   19 R = R-DD                                                          WHEE0580
 1901 DD =0.0                                                           WHEE0590
      DO 20 J=2,N                                                       WHEE0600
      DD=DD+FJP(J)                                                      WHEE0610
20    FJP(J)=FJP(J-1)+DD                                                WHEE0620
      RETURN                                                            WHEE0630
      END                                                               WHEE0640
```

3.2 Vehicle Dynamics Version

A description of each computational subroutine of the HVOSM-VD2 is provided in this section. Included is a brief description of the purpose of the subroutine, a description of the linkages to the rest of the program in the forms of subroutines called, calling arguments, common blocks appearing, variables within the common blocks that are the result of a computation, and, the subroutine size. Also included is a description of the computational procedure employed either in the form of a verbal listing of the computational steps or an annotated flowchart illustrating the logical sequence of computations. Since this part of the subroutine description is intended to illustrate the procedure, it does not always illustrate each individual line of coding. When a detailed investigation of the coding is required, the computational procedure should be used in conjunction with a subroutine listing.

An overall program block diagram is shown in Figure 3.2-1, a matrix of common blocks appearing in each subroutine in Figure 3.2-2, and a matrix of subroutine calls in Figure 3.2-3.

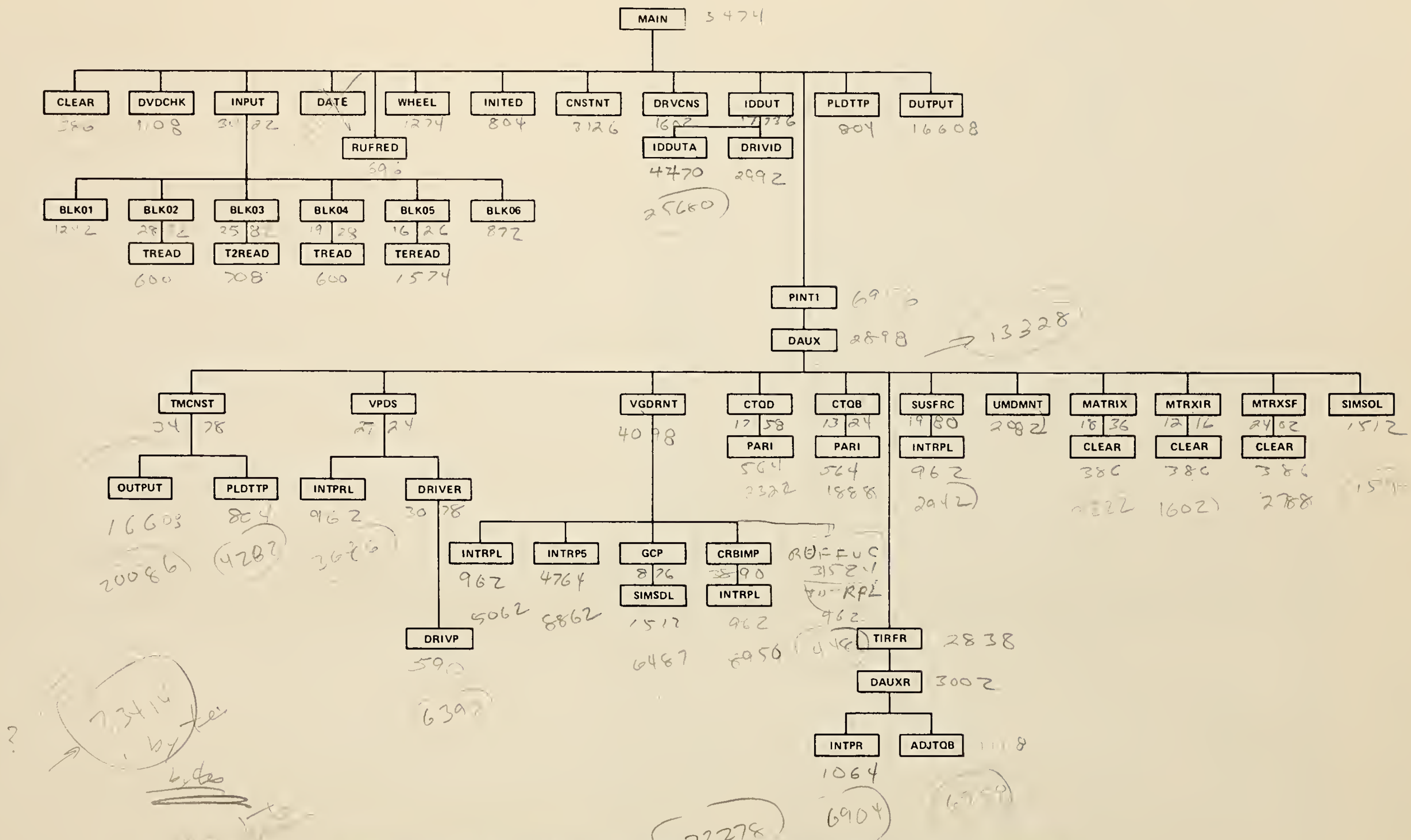


Figure 3.2-1 HVOSM-VD2 OVERALL PROGRAM BLOCK DIAGRAM

COMMON BLOCK

	HEAD	INPT	INPT1	INTG	DIMV	COMP	COMPN	EINDEX	ADTNL	INPT3	APTABL	TIRIN	INPT4	COMP4	INPT5	COMP5	INTR	INSUS	SUSCMP	DRIVTT	DRIVI	DRIVE	NEWCRB	RUFNES
MAIN	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
ADJTOB					●										●	●	●							
BLKO1	●	●	●										●					●		●			●	
BLKO2	●	●	●							●	●		●		●			●						
BLKO3	●		●									●	●											
BLKO4	●	●													●						●	●		
BLKO5	●	●	●																				●	●
BLKO6	●	●	●												●			●						
CLEAR																								
CNSTNT	●	●	●	●		●	●	●				●	●	●	●	●	●	●	●				●	
CRBIMP			●	●	●	●	●					●											●	
CTQB															●	●								
CTOD				●									●		●	●	●			●	●	●		
DATE																								
DAUX		●	●	●	●	●	●		●				●	●			●	●	●					
DAUXR					●		●					●	●	●	●	●	●							
DRIVER			●	●	●	●		●												●	●	●		
DRIVID	●	●																		●	●	●		
DRIVP																				●	●	●		
DRVCNS		●	●		●	●	●						●		●	●	●			●	●	●		
DVDCHK																								
GCP					●	●						●												
IDOUT	●	●	●			●				●	●	●	●					●		●			●	
IDOUTA	●												●		●									
INITEO		●			●	●	●					●						●						
INPUT																								
INTPR																								
INTRPL																								
INTRP5		●			●	●	●					●												
MATRIX		●		●	●	●			●															
MTRXIR		●		●	●	●			●										●					
MTRXSF		●		●	●	●			●									●	●					
OUTPUT	●	●		●	●	●	●		●				●	●		●	●	●	●			●		
PARI																								
PINT1																								
PLOTTP		●		●	●	●	●					●												
RUFFRC			●		●	●	●					●												●
RUFRED																								
SIMSOL																								
SUSFRC		●		●	●	●			●	●	●							●	●					
TEREAD		●																						
TIRFR		●		●	●	●	●						●	●	●	●	●							
TMCNST		●		●		●		●						●	●			●	●					
TREAD																								
T2READ																								
UMOMNT		●			●	●										●		●	●					
VPOS		●	●	●	●	●	●		●									●	●					
VGORNT		●	●	●	●	●	●		●			●						●	●				●	●
WHEEL																								

Figure 3.2-2 HVOSM-VD2 COMMON BLOCK ALLOCATIONS

CALLEO ROUTINE

CALLING ROUTINE	MAIN	ADJTOB	BLKO1	BLKO2	BLKO3	BLKD4	BLKO5	BLKD6	CLEAR	CNSTNT	CRBIMP	CTOB	CTOD	OATE	DAUX	DAUXR	ORIVER	DRIVIO	DRIVP	DRVCNS	OVDCHK	GCP	IDOUT	IDOUTA	INITEO	INPUT	INTPR	INTRPL	INTRP5	MATRIX	MTRXIR	MTRXSF	OUTPUT	PARI	PINT1	PLOTTP	RUFFRC	RUFRED	SIMSOL	SUSFRC	TEREAD	TIRFR	TMCNST	TREAD	T2READ	UMOMNT	VPOS	VGORNT	WHEEL
MAIN									●	●				●						●	●		●		●	●							●		●	●		●											●
AOJTOB																																																	
BLKO1																																																	
BLKO2																																												●					
BLKD3																																													●				
BLKO4																																												●					
BLKO5																																									●								
BLKO6																																																	
CLEAR																																																	
CNSTNT																																																	
CRBIMP																												●																					
CTOB																																		●															
CTOD																																		●															
OATE																																																	
OAUX												●	●																	●	●	●							●	●		●	●			●	●	●	
OAUXR		●																									●																						
ORIVER																			●																														
DRIVID																																																	
ORIVP																																																	
ORVCNS																																																	
DVDCHK																																																	
GCP																																							●										
IDOUT																		●						●																									
IDDUTA																																																	
INITEO																																																	
INPUT			●	●	●	●	●	●																																									
INTPR																																																	
INTRPL																																																	
INTRP5																																																	
MATRIX									●																																								
MTRXIR									●																																								
MTRXSF									●																																								
DUTPUT																																																	
PARI																																																	
PINT1															●																																		
PLOTTP																																																	
RUFFRC																												●																					
RUFREO																																																	
SIMSOL																																																	
SUSFRC																												●																					
TEREAD																																																	
TIRFR																●																																	
TMCNST																																	●			●													
TREAO																																																	
T2REAO																																																	
UMOMNT																																																	
VPOS																	●											●																					
VGORNT											●											●						●	●								●												
WHEEL																																																	

Figure 3.2-3 HVOSM-VD2 MATRIX OF SUBROUTINE CALLS

3.2.1 HVOSM-VD2 Subroutine Documentation

1. MAIN ROUTINE

a. Purpose:

1. Clear selected COMMON blocks
2. Obtain input and print input
3. Program initialization
4. Control computation of constants
5. Control the integration loop
6. Control abnormal program stops
7. Control indexing of coordinate system
8. Control integration step size for curb
9. Control output

b. Common Blocks Required:

HEAD, INPT, INPT1, INTG, DIMV, COMP, COMPN, EINDEX, ADTNL, INPT3, APTBL, DRIVTT, DRIVI, DRIVE, COMP4, COMP5, INPT4, INPT5, INTR, TIRIN, INSUS, SUSCMP, NEWCRB, RUFNES

c. Subroutines Required:

CLEAR, DVDCHK, INPUT, DATE, IDOUT, CNSTNT, PLOTTP, PINT1, OUTPUT, WHEEL, RUFRED, INITEQ, DRVCNS

d. Arguments:

None

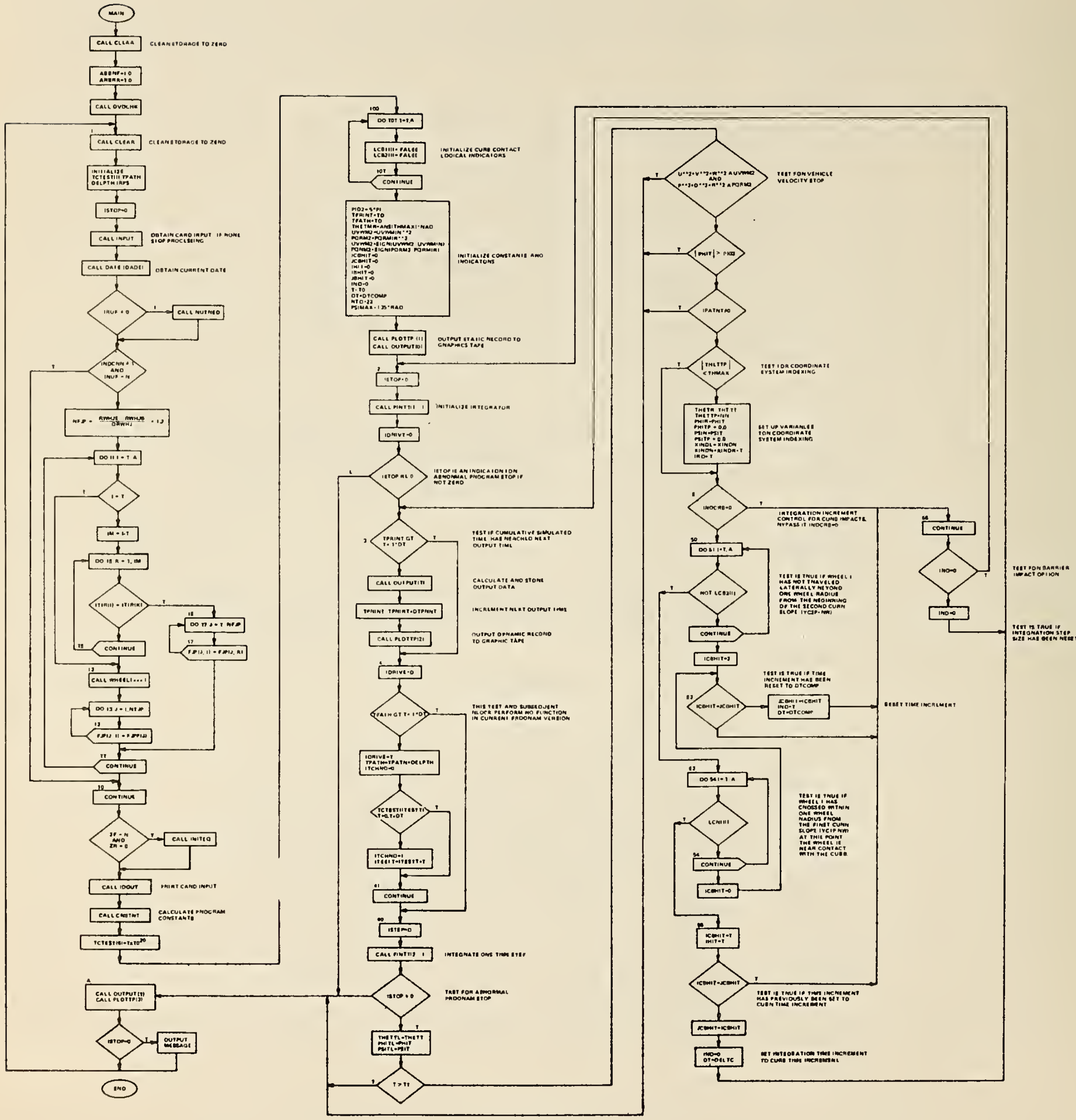
e. Common Variables Calculated:

DT, FJP, NEQ, ZGP, DADE, IHIT, IRPS, LCB1, LCB2, PHIN, PSIN, ISTEP, ISTOP, PHIB1, PHIB2, PHITL, PHITP, PSITL, PSITP, TPATH, XINDL, XINDN, DELPTH, ICBHIT, IDRIVE, ITCHNG, ITESTT, JCBHIT, PQRMIN, TCTEST, THETTL, THETTP

f. Size:

$D92)_{16} = 3474)_{10}$ bytes

g. Computational Procedure:



2. SUBROUTINE ADJTQB

a. Purpose:

1. To adjust braking torques at low values of wheel rotational velocity to prevent sign reversal

b. Common Blocks Required:

DIMV, INTR, INPT5, COMP5

c. Subroutines Required:

None

d. Arguments:

None

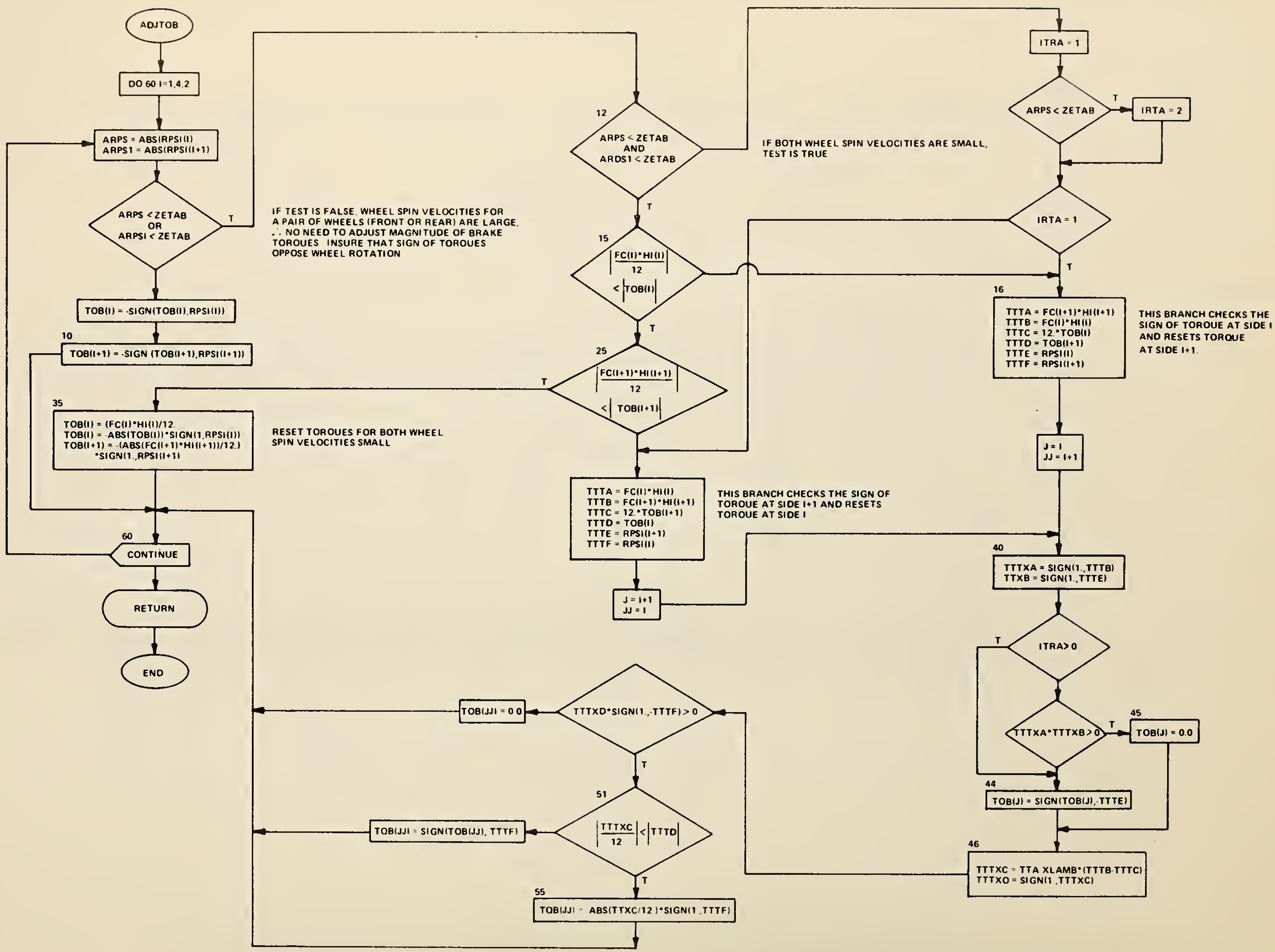
e. Common Variables Calculated:

TQB

f. Size:

$45E)_{16} = 1118)_{10}$ bytes

g. Computational Procedure:


ADJTOB
DO 60 I=1,4,2
ARPS = ABS(RPSI(I)
ARPS1 = ABS(RPSI(I+1)
ARPS < ZETAB OR ARPSI < ZETAB
T
IF TEST IS FALSE, WHEEL SPIN VELOCITIES FOR A PAIR OF WHEELS (FRONT OR REAR) ARE LARGE, ∴ NO NEED TO ADJUST MAGNITUDE OF BRAKE TORQUES INSURE THAT SIGN OF TORQUES OPPOSE WHEEL ROTATION
TOB(I) = -SIGN(TOB(I),RPSI(I))
10
TOB(I+1) = -SIGN (TOB(I+1),RPSI(I+1))
35
TOB(I) = (FC(I)*HI(I)/12.
TOB(I) = -ABS(TOB(I))*SIGN(1,RPSI(I))
TOB(I+1) = -(ABS(FC(I+1)*HI(I+1))/12.)
*SIGN(1.,RPSI(I+1)
RESET TORQUES FOR BOTH WHEEL SPIN VELOCITIES SMALL
60
CONTINUE
RETURN
END
12
ARPS < ZETAB AND ARDS1 < ZETAB
IF BOTH WHEEL SPIN VELOCITIES ARE SMALL, TEST IS TRUE
15
\|FC(I)*HI(I) / 12\| < \|TOB(I)\|
25
\|FC(I+1)*HI(I+1) / 12\| < \|TOB(I+1)\|
TTTA = FC(I)*HI(I)
TTTB = FC(I+1)*HI(I+1)
TTTC = 12.*TOB(I+1)
TTTD = TOB(I)
TTTE = RPSI(I+1)
TTTF = RPSI(I)
THIS BRANCH CHECKS THE SIGN OF TORQUE AT SIDE I+1 AND RESETS TORQUE AT SIDE I
J = I+1
JJ = I
ITRA = 1
ARPS < ZETAB
IRTA = 2
IRTA = 1
16
TTTA = FC(I+1)*HI(I+1)
TTTB = FC(I)*HI(I)
TTTC = 12.*TOB(I)
TTTD = TOB(I+1)
TTTE = RPSI(I)
TTTF = RPSI(I+1)
THIS BRANCH CHECKS THE SIGN OF TORQUE AT SIDE I AND RESETS TORQUE AT SIDE I+1.
J = I
JJ = I+1
40
TTTXA = SIGN(1.,TTTB)
TTXB = SIGN(1.,TTTE)
ITRA> 0
TTTXA*TTTXB > 0
45
TOB(J) = 0.0
44
TOB(J) = SIGN(TOB(J),-TTTE)
46
TTTXC = TTA XLAMB*(TTTB-TTTC)
TTTXO = SIGN(1 ,TTTXC)
TTTXD*SIGN(1.,-TTTF) > 0
TOB(JJ) = 0 0
51
\|TTTXC / 12\| < \|TTTD\|
TOB(JJ) = SIGN(TOB(JJ), TTTF)
55
TOB(JJ) = ABS(TTXC/12.)*SIGN(1.,TTTF)

3. SUBROUTINE BLK01

a. Purpose:

1. Assign input values of simulation control data

b. Common Blocks Required:

HEAD, INPT, INPT1, INPT4, INSUS, NEWCRB, DRIVTT

c. Subroutines Required:

None

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

EM, T0, T1, AAA, BET, HED, EBAR, HMAX, HMIN, IBUG, ISUS, MODE, NERR, DELTC, NPAGE, DTCOMP, DTPRNT, IDRVER, INDCRB, NCRBSL, PQRMIN

f. $4DA)_{16} = 1242)_{10}$ bytes

4. SUBROUTINE BLK02

a. Purpose:

1. Assign input values of simulation vehicle data

b. Common Blocks Required:

HEAD, INPT, INPT1, INPT3, APTABL, INSUS, INPT4, INPT5

c. Subroutines Required:

TREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

A, B, G, CF, CR, GN, RF, RR, TF, TR, TS, X1, X2, Y1, Y2, ZF, ZR, Z1, Z2, AKF, AKR, AK1, AK2, APF, APR, CFP, CRP, RHO, TCT, TLF, TSF, XIF, XIR, XIX, XIY, XIZ, XMS, XPS, AKDS, AKFC, AKFE, AKPS, AKRC, AKRE, AKRS, BRPM, BTLF, CONE, CPSP, CTWO, DDEL, DELB, DELE, DRPM, DTHF, DTHR, DTLF, EPSF, EPSR, ERPM, ETLF, NAPF, NAPR, NDEL, NRPM, NTLF, PHIC, PONE, PTWO, RHOF, TWOT, VHED, XIPS, XIXZ, XMUF, XMUR, AKDS1, AKDS2, AKDS3, AKFCP, AKFEP, AKRCP, AKREP, ARBRF, ARBRR, DAPFB, DAPFE, DAPRB, DAPRE, DDAPF, DDAPR, EPSPS, FIDJF, FIDJR, FIWJF, FIWJR, IAPFR, IBTYP, NDTHF, NDTHR, NPAGE, PZERO, XLAMF, XLAMR, ZETAB, CTHREE, OMEGFC, OMEGFE, OMEGRC, OMEGRE

f. Size:

$B38)_{16} = 2872)_{10}$ bytes

5. SUBROUTINE BLK03

a. Purpose:

1. Assign input values of simulation tire data

b. Common Blocks Required:

HEAD, INPT1, TIRIN, INPT4

c. Subroutines Required:

T2READ

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

A0, A1, A2, A3, A4, CT, RW, AKT, AMU, ITIR, RRMC, SIGT, THED, XMUM, DRWHJ, OMEGT, RWHJE, XLAMT, NXFRCP, NXUGMU, XMUMAT, XMXPMT, XMXSMT, XXFRPD, XXUGMU

f. Size:

$A16)_{16} = 2582)_{10}$ bytes

6. SUBROUTINE BLK04

a. Purpose:

1. Assign input values of vehicle control data

b. Common Blocks Required:

HEAD, INPT, INPT5, DRIVI, DRIVE

c. Subroutines Required:

TREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

S, DS, EN, TB, TE, TL, BTT, DTT, ETT, TIL, TPC, TTR, TTS, BFP1, BFP2, CHED, EMDT, FKD0, NTTS, NTT1, NTT2, NTT3, TAUF, DESSI, DISTI, FKS10, FKS20, GEAR1, GEAR2, GEAR3, GEAR4, NPAGE, NTBL1, NTRAN, TINCR, VGR12, VGR21, VGR23, VGR32, VGR34, VGR43, APDMAX, FKSKD0, TESTB0, TSTR10, TSTR20, TSTS10, TSTS20, XIMPOR, YTRANS

f. Size:

$788)_{16} = 1928)_{10}$ bytes

7. SUBROUTINE BLK05

a. Purpose:

1. Assign input values of terrain and curb data

b. Common Blocks Required:

HEAD, INPT, INPT1, NEWCRB

c. Subroutines Required:

TEREAD

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

NX, NY, XB, XE, YB, YE, NBX, NBY, NZ5, AMUG, DELG, GHED, IRUF, NEND, YC1P, YC2P, YC3P, YC4P, YC5P, YC6P, ZC2P, ZC3P, ZC4P, ZC5P, ZC6P, DGMAX, NPAGE, NZTAB, PHIC1, PHIC2, PHIC3, PHIC4, PHIC5, PHIC6, XINCR, YINCR

f. Size:

$65A)_{16} = 1626)_{10}$ bytes

8. SUBROUTINE BLK06

a. Purpose:

1. Assign input values of simulation initial conditions

b. Common Blocks Required:

HEAD, INPT, INPT1, INSUS, INPT5

c. Subroutines Required:

None

d. Arguments:

NBLK - Input data block number
NBCRD - Card number within the block
NSEQ - Table sequence number
NCARD - Card number
DUM - Array containing input values read in Subroutine INPUT to be assigned to the appropriate variable names within this subroutine
NERR - Error indicator

e. Common Variables Calculated:

PO, QO, RO, UO, VO, WO, PHIO, PSIO, SHED, TAUA, TAUO, XCOP, YCOP, ZCOP, DEL1O, DEL2O, DEL3O, DEL4O, PHIFO, PHIRO, DEL1OD, DEL2OD, DEL3OD, DEL4OD, PHIFOD, PHIROD, PSIFDO, PSIFIO, THETAO

f. Size:

$368)_{16} = 872)_{10}$ bytes

9. SUBROUTINE CLEAR(A,B)

a. Purpose:

1. To set a block of storage to zero

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

A - beginning address to be cleared
B - end of the full-word address to be cleared

e. Common Variables Calculated:

None

f. Size:

$182)_{16} = 386)_{10}$ bytes

10. SUBROUTINE CNSTNT

a. Purpose:

1. Evaluate program constants
2. Initialize dependent variables and derivatives to input initial conditions

b. Common Blocks Required:

HEAD, INPT, INPT1, INTG, COMP, COMPN, EINDEX, TIRIN, COMP4, COMP5, INPT4, INPT5, INTR, INSUS, SUSCMP, NEWCRB

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

P, Q, R, U, V, W, TT, A12, A23, RAD, RTF, RTR, TAU, TIZ, TM4, XCP, YCP, ZCP, ZF0, ZPR, ZR0, AMUF, BMUR, DEL1, DEL2, DEL3, DEL4, GAM1, PHIF, PHIN, PHIR, PI02, PI04, PSIN, RFTF, RH02, RRTR, RRTS, TR02, TS02, XIZR, YCLP, YC3P, YC4P, YC5P, YC6P, ZCLP, ZC3P, ZC4P, ZC5P, ZC6P, ARBRI, DEL1D, DEL2D, DEL3D, DEL4D, JDEND, NPAGE, PHIFD, PHIRD, PHITL, PHITP, PSIFI, PSITL, PSITP, RHMR2, RHOF2, RPSFA, RPSFB, RPSFC, RPSFD, RPSFE, THETN, TLAMB, TRPME, TSF02, TWOPI, XINDN, A02APB, ARFAC1, ARFAC2, ARFAC3, AXMF02, B02APB, BROMUR, BXMR02, OMT2A2, OMT2M1, PHICLR, PHIC1R, PHIC2R, PHIC3R, PHIC4R, PHIC5R, PHIC6R, PI015R, RF2MFI, RHFMUF, RHF2MF, RHMR2I, RHOMUR, TANPC1, TANPC2, TANPC3, TANPC4, TANPC5, TANPC6, TBRAKA, TBRAKB, TBRAKD, THETTL, THETTP, TWOPIR, XMTF04, XMTR04, XMUF02, XMUR02

f. Size:

$C36)_{16} = 3126)_{10}$ bytes

g. Computational Procedure:

1. Compute program constants
2. Initialize dependent variables converting degrees to radians
3. Initialize XINDN = 10, if THETN or PHIN are not zero for use in MAIN and TMCNST to control coordinate system indexing

SUBROUTINE CRBIMP(I)

a. Purpose:

1. Determine the radial tire force and equivalent ground contact point when a tire is in contact with a curb

b. Common Blocks Required:

INPT1, DIMV, COMP, COMPN, INTG, TIRIN, NEWCRB

c. Subroutines Required:

INTRPL

d. Arguments:

The argument I indicates the wheel number for which calculations are made

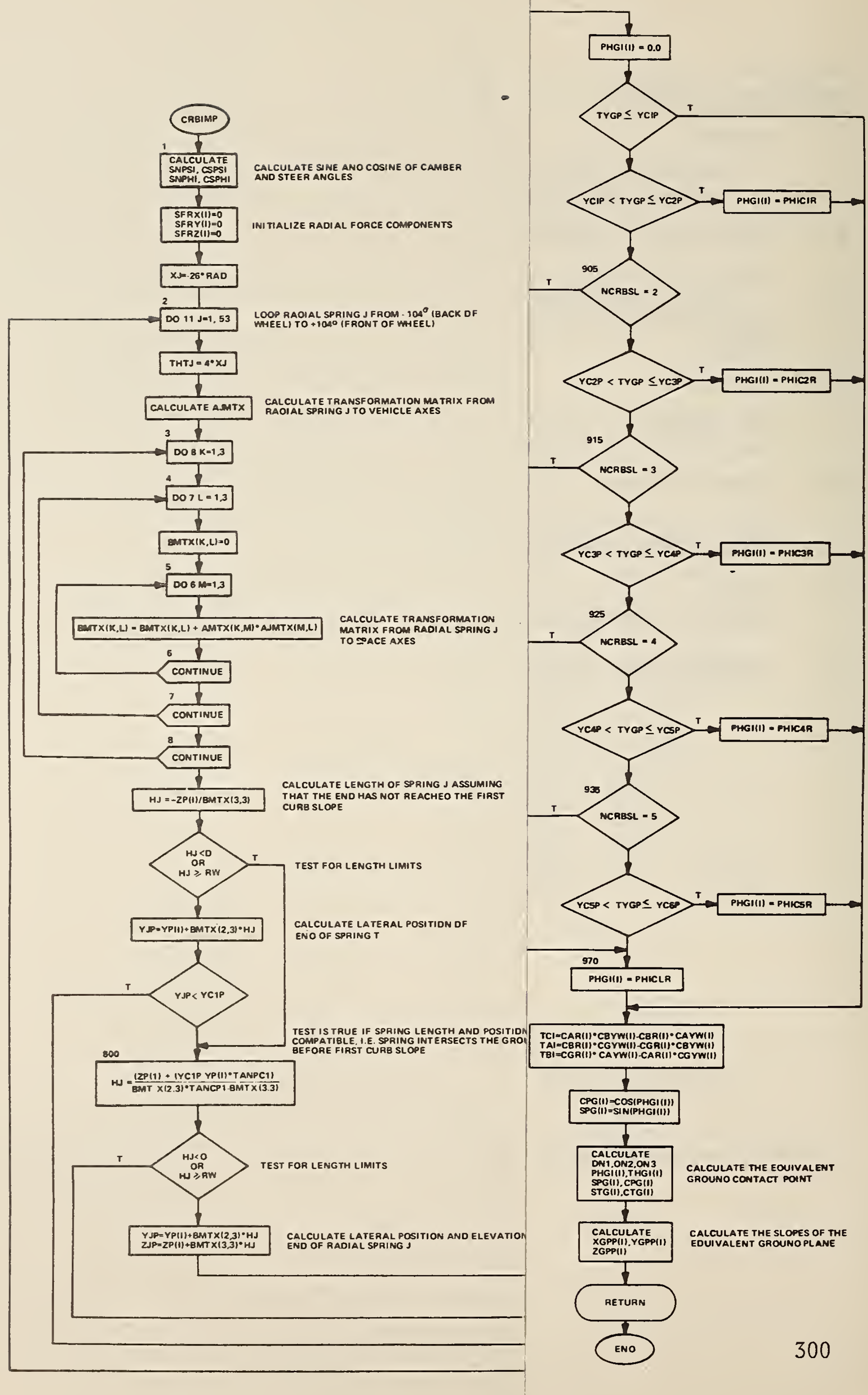
e. Common Variables Calculated:

FR, HI, RW, CAR, CBR, CGR, CPG, CTG, SPG, STG, BMTX, PHGI, SFRX, SFRY, SFRZ, THGI, XGPP, YGPP, ZGPP, AJMTX

f. Size:

$F32)_{16} = 3890)_{10}$ bytes

g. Computational Procedure:



CRBIMP
1
CALCULATE SNPSI, CSPSI SNPHI, CSPHI
CALCULATE SINE AND COSINE OF CAMBER AND STEER ANGLES
SFRX(I)=0 SFRY(I)=0 SFRZ(I)=0
INITIALIZE RADIAL FORCE COMPONENTS
XJ=-26*RAD
2
DO 11 J=1, 53
LOOP RADIAL SPRING J FROM -104° (BACK OF WHEEL) TO +104° (FRONT OF WHEEL)
THTJ = 4*XJ
CALCULATE AJMTX
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING J TO VEHICLE AXES
3
DO 8 K=1,3
4
DO 7 L = 1,3
BMTX(K,L)=0
5
DO 6 M=1,3
BMTX(K,L) = BMTX(K,L) + AMTX(K,M)*AJMTX(M,L)
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING J TO SPACE AXES
6
CONTINUE
7
CONTINUE
8
CONTINUE
HJ = -ZP(I)/BMTX(3,3)
CALCULATE LENGTH OF SPRING J ASSUMING THAT THE END HAS NOT REACHED THE FIRST CURB SLOPE
HJ<0 OR HJ ≥ RW
T
TEST FOR LENGTH LIMITS
YJP=YP(I)+BMTX(2,3)*HJ
CALCULATE LATERAL POSITION OF END OF SPRING T
YJP< YC1P
T
TEST IS TRUE IF SPRING LENGTH AND POSITION COMPATIBLE, I.E. SPRING INTERSECTS THE GROU BEFORE FIRST CURB SLOPE
800
HJ = (ZP(1) + (YC1P YP(I)*TANPC1) / BMT X(2,3)*TANCP1-BMTX(3,3)
HJ<0 OR HJ ≥ RW
T
TEST FOR LENGTH LIMITS
YJP=YP(I)+BMTX(2,3)*HJ ZJP=ZP(I)+BMTX(3,3)*HJ
CALCULATE LATERAL POSITION AND ELEVATION END OF RADIAL SPRING J
PHGI(I) = 0.0
TYGP ≤ YC1P
T
YC1P < TYGP ≤ YC2P
T
PHGI(I) = PHIC1R
905
NCRBSL = 2
T
YC2P < TYGP ≤ YC3P
T
PHGI(I) = PHIC2R
915
NCRBSL = 3
T
YC3P < TYGP ≤ YC4P
T
PHGI(I) = PHIC3R
925
NCRBSL = 4
T
YC4P < TYGP ≤ YC5P
T
PHGI(I) = PHIC4R
935
NCRBSL = 5
T
YC5P < TYGP ≤ YC6P
T
PHGI(I) = PHIC5R
970
PHGI(I) = PHICLR
TCI=CAR(I)*CBYW(I)-CBR(I)*CAYW(I) TAI=CBR(I)*CGYW(I)-CGR(I)*CBYW(I) TBI=CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
CPG(I)=COS(PHGI(I)) SPG(I)=SIN(PHGI(I))
CALCULATE DN1,DN2,DN3 PHGI(I),THGI(I) SPG(I),CPG(I) STG(I),CTG(I)
CALCULATE THE EQUIVALENT GROUND CONTACT POINT
CALCULATE XGPP(I),YGPP(I) ZGPP(I)
CALCULATE THE SLOPES OF THE EQUIVALENT GROUND PLANE
RETURN
END

12. SUBROUTINE CTQB

a. Purpose:

   1. To calculate braking torques from hydraulic pressure, brake type and fade coefficient

b. Common Blocks Required:

   INPT5, COMP5

c. Subroutines Required:

   Function PARI

d. Arguments:

   None

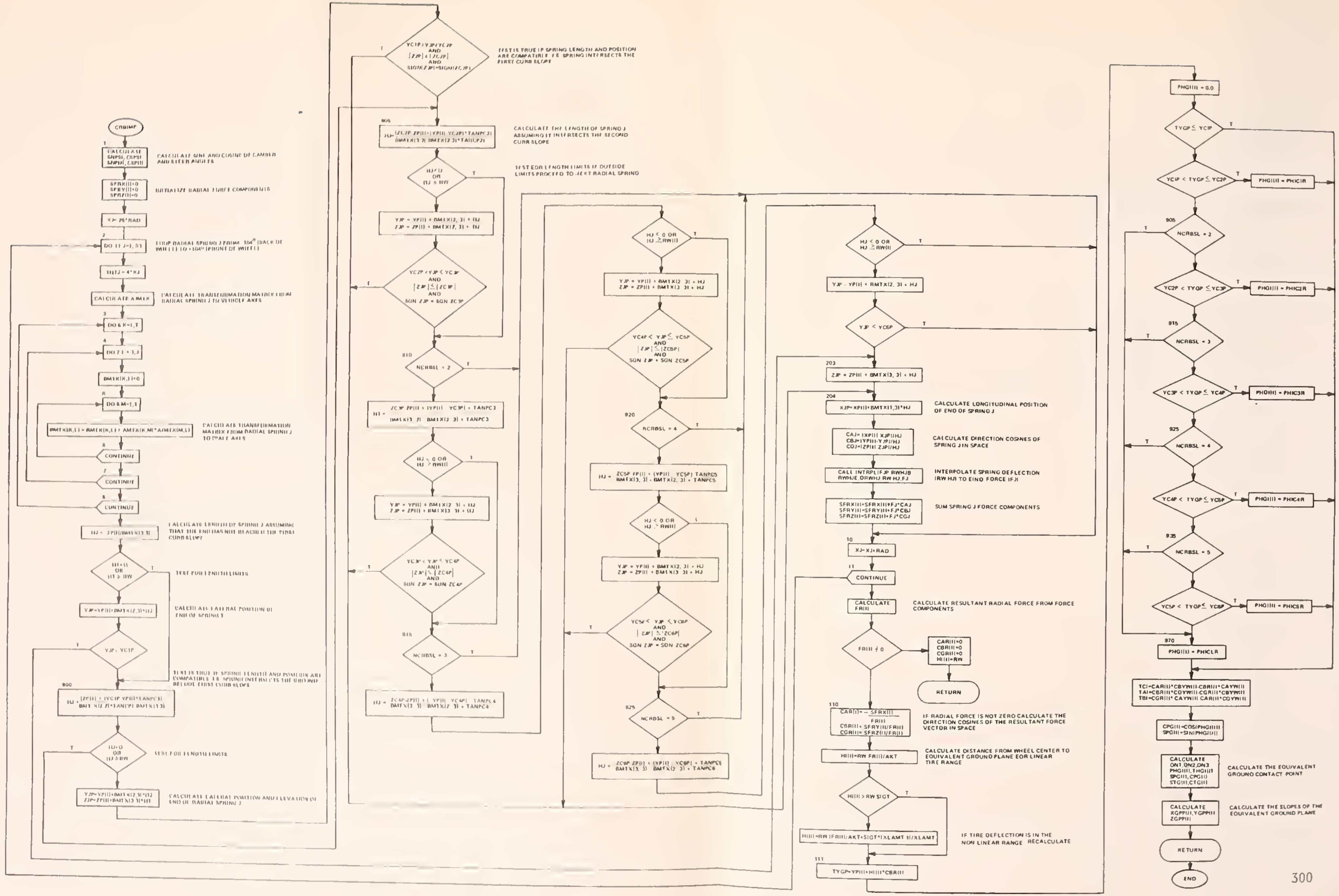
e. Common Variables Calculated:

   NBTYP, PP, TQB

f. Size:

   $52C)_{16} = 1324)_{10}$ bytes

g. Computational Procedure:


CGBIMP
CALCULATE SINE AND COSINE OF CAMBER AND STEER ANGLES
INITIALIZE RADIAL FORCE COMPONENTS
CALCULATE TRANSFORMATION MATRIX FROM RADIAL SPRING J TO VEHICLE AXES
CONTINUE
TEST FOR LENGTH LIMITS
CALCULATE LATERAL POSITION OF END OF SPRING J
CALCULATE LATERAL POSITION AND ELEVATION OF END OF RADIAL SPRING J
TEST IS TRUE IF SPRING LENGTH AND POSITION ARE COMPATIBLE, I.E. SPRING INTERSECTS THE FIRST CURB SLOPE
CALCULATE THE LENGTH OF SPRING J ASSUMING IT INTERSECTS THE SECOND CURB SLOPE
TEST FOR LENGTH LIMITS IF OUTSIDE LIMITS PROCEED TO NEXT RADIAL SPRING
NCRBSL = 2
NCRBSL = 3
NCRBSL = 4
NCRBSL = 5
CALCULATE LONGITUDINAL POSITION OF END OF SPRING J
CALCULATE DIRECTION COSINES OF SPRING J IN SPACE
INTERPOLATE SPRING DEFLECTION (RW HJ) TO FIND FORCE (FJ)
SUM SPRING J FORCE COMPONENTS
XJ=XJ+RAD
CONTINUE
CALCULATE FR(I)
CALCULATE RESULTANT RADIAL FORCE FROM FORCE COMPONENTS
FR(I) ≠ 0
CAR(I)=0
CBR(I)=0
CGR(I)=0
H(I)=RW
RETURN
IF RADIAL FORCE IS NOT ZERO CALCULATE THE DIRECTION COSINES OF THE RESULTANT FORCE VECTOR IN SPACE
H(I)=RW FR(I)/AKT
CALCULATE DISTANCE FROM WHEEL CENTER TO EQUIVALENT GROUND PLANE FOR LINEAR TIRE RANGE
H(I) > RW SIGT
IF TIRE DEFLECTION IS IN THE NON LINEAR RANGE RECALCULATE
TYGP=YP(I)+H(I)*CBR(I)
PHG(I) = 0.0
TYGP ≤ YC1P
PHG(I) = PHC1R
PHG(I) = PHC2R
PHG(I) = PHC3R
PHG(I) = PHC4R
PHG(I) = PHC5R
PHG(I) = PHCLR
CALCULATE THE EQUIVALENT GROUND CONTACT POINT
CALCULATE THE SLOPES OF THE EQUIVALENT GROUND PLANE
RETURN
END

12. SUBROUTINE CTQB

a. Purpose:

1. To calculate braking torques from hydraulic pressure, brake type and fade coefficient

b. Common Blocks Required:

INPT5, COMP5

c. Subroutines Required:

Function PARI

d. Arguments:

None

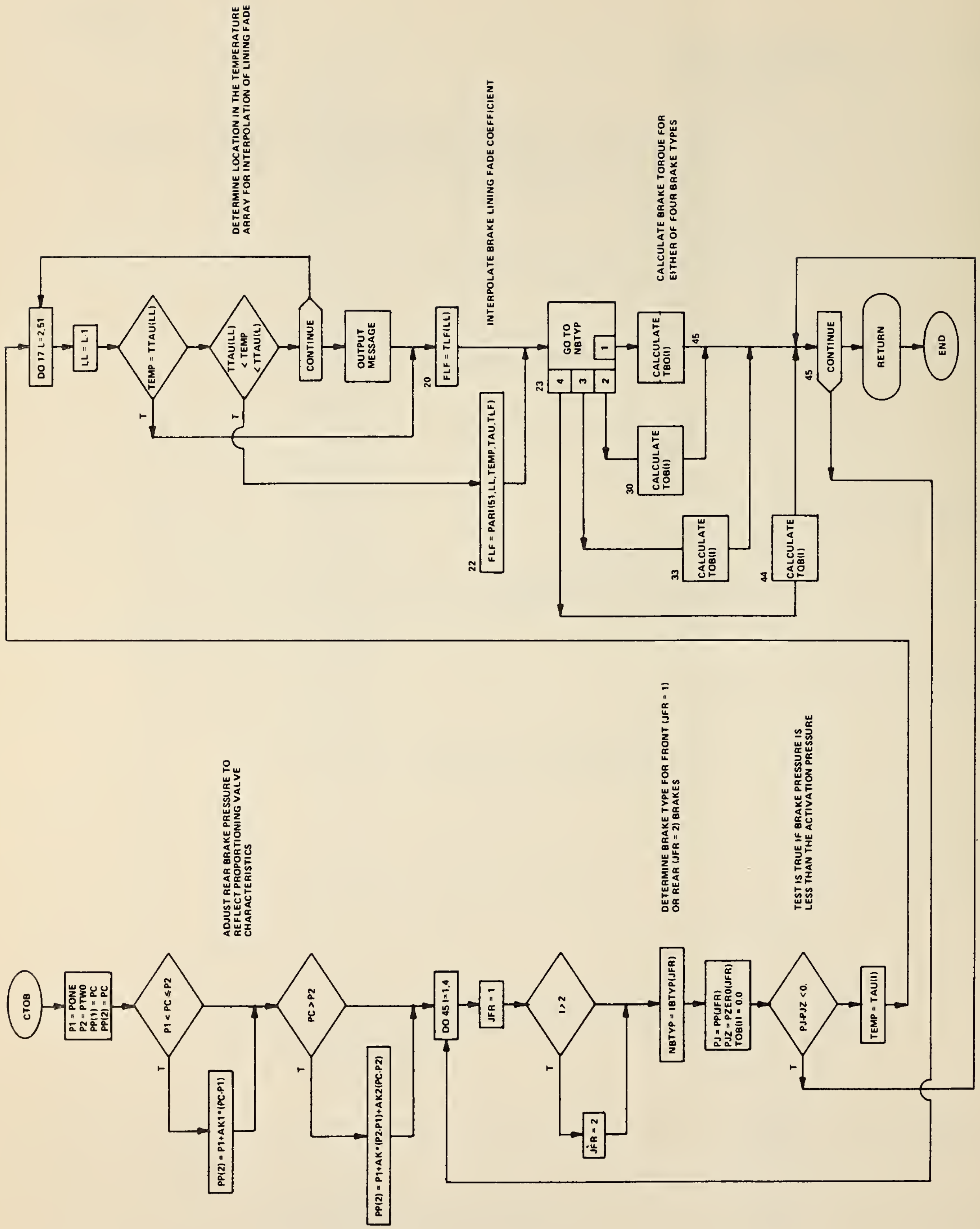
e. Common Variables Calculated:

NBTYP, PP, TQB

f. Size:

$52C)_{16} = 1324)_{10}$ bytes

g. Computational Procedure:


CTOB
P1 = PONE
P2 = PTWO
PP(1) = PC
PP(2) = PC
P1 < PC ≤ P2
T
PP(2) = P1+AK1*(PC-P1)
ADJUST REAR BRAKE PRESSURE TO REFLECT PROPORTIONING VALVE CHARACTERISTICS
PC > P2
T
PP(2) = P1+AK*(P2-P1)+AK2(PC-P2)
DO 45 I=1,4
JFR = 1
I>2
T
JFR = 2
DETERMINE BRAKE TYPE FOR FRONT (JFR = 1) OR REAR (JFR = 2) BRAKES
NBTYP = IBTYP(JFR)
PJ = PP(JFR)
PJZ = PZERO(JFR)
TOB(I) = 0.0
PJ-PJZ <0.
T
TEST IS TRUE IF BRAKE PRESSURE IS LESS THAN THE ACTIVATION PRESSURE
TEMP = TAU(I)
DO 17 L=2,51
LL = L-1
TEMP = TTAU(LL)
T
TTAU(LL) < TEMP < TTAU(L)
T
CONTINUE
DETERMINE LOCATION IN THE TEMPERATURE ARRAY FOR INTERPOLATION OF LINING FADE
OUTPUT MESSAGE
20
FLF = TLF(LL)
22
FLF = PARI(51,LL,TEMP,TAU,TLF)
INTERPOLATE BRAKE LINING FADE COEFFICIENT
23
GO TO NBTYP
1
2
3
4
CALCULATE TBO(I)
30
CALCULATE TOB(I)
33
CALCULATE TOB(I)
44
CALCULATE TOB(I)
CALCULATE BRAKE TORQUE FOR EITHER OF FOUR BRAKE TYPES
45
45
CONTINUE
RETURN
END

13. SUBROUTINE CTQD

a. Purpose:

1. To compute the driveline torque at the driving end of the vehicle from throttle setting, transmission ratio and engine speed

b. Common Blocks Required:

INTG, INTR, INPT4, INPT5, COMP5, DRIVTT, DRIVI, DRIVE

c. Subroutines Required:

Function PARI

d. Arguments:

None

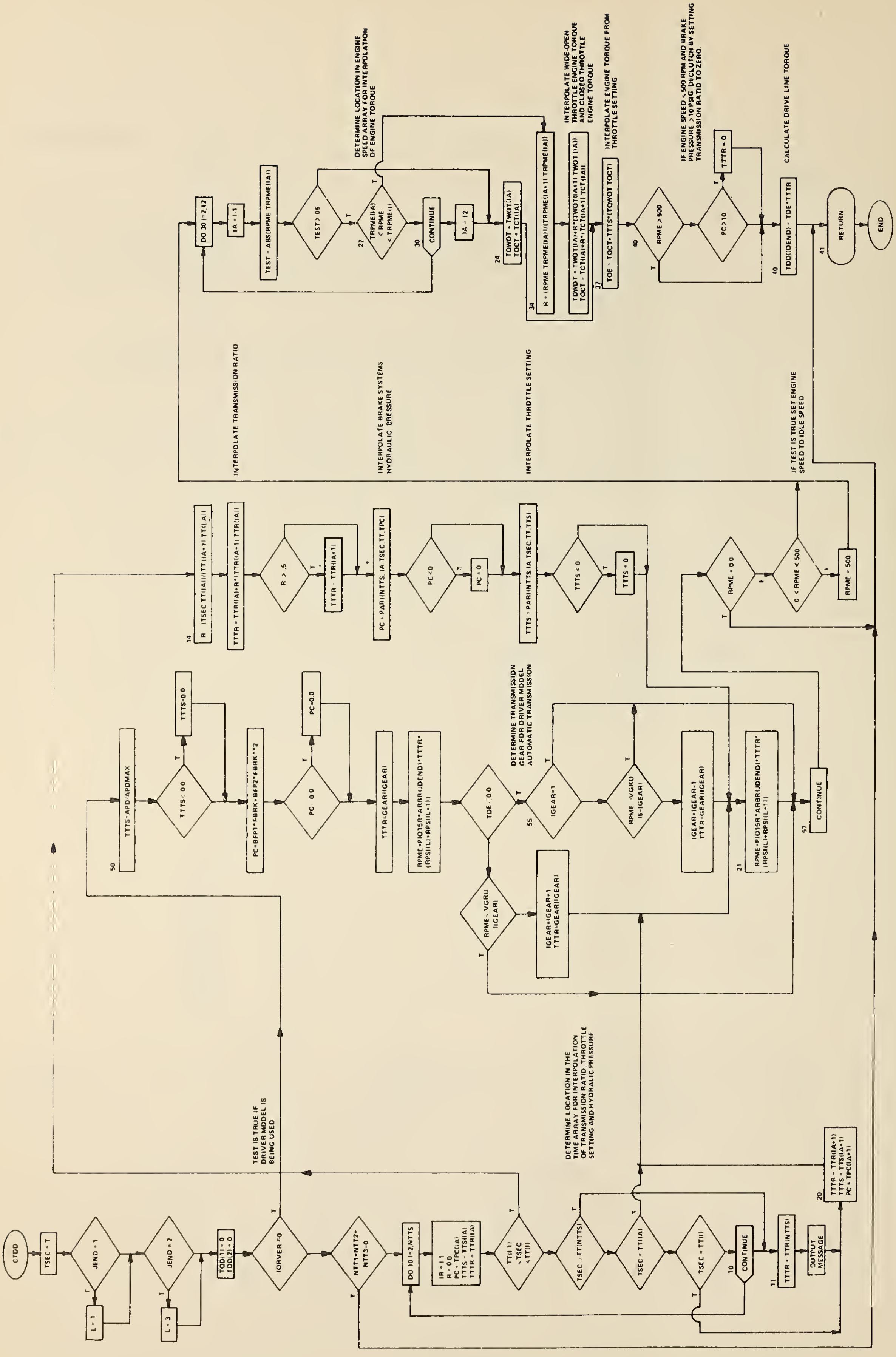
e. Common Variables Calculated:

PC, TQD, TQE, RPME, IGEAR

f. Size:

$6DE)_{16} = 1758)_{10}$ bytes

g. Computational Procedure:


CTDD
TSEC = T
TEST IS TRUE IF DRIVER MODEL IS BEING USED
DETERMINE LOCATION IN THE TIME ARRAY FOR INTERPOLATION OF TRANSMISSION RATIO, THROTTLE SETTING AND HYDRALIC PRESSURF
OUTPUT MESSAGE
CONTINUE
DETERMINE TRANSMISSION GEAR FOR DRIVER MODEL AUTOMATIC TRANSMISSION
INTERPOLATE TRANSMISSION RATIO
INTERPOLATE BRAKE SYSTEMS HYDRAULIC PRESSURE
INTERPOLATE THROTTLE SETTING
IF TEST IS TRUE SET ENGINE SPEED TO IDLE SPEED
DETERMINE LOCATION IN ENGINE SPEED ARRAY FOR INTERPOLATION OF ENGINE TORQUE
INTERPOLATE WIDE-OPEN THROTTLE ENGINE TORQUE AND CLOSED THROTTLE ENGINE TORQUE
INTERPOLATE ENGINE TORQUE FROM THROTTLE SETTING
IF ENGINE SPEED < 500 RPM AND BRAKE PRESSURE > 10 PSIG, DECLUTCH BY SETTING TRANSMISSION RATIO TO ZERO.
CALCULATE DRIVE LINE TORQUE
RETURN
END

14. SUBROUTINE DATE

a. Purpose:

1. Return the calendar date in 8 byte form, e.g., 23MAR'68

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

DADE - Array into which the date is loaded

e. Common Variables Calculated:

None

f. Size:

$D6)_{16} = 214)_{10}$ bytes

g. Procedure:

This subroutine is written in IBM S/360 Assembler Language

15. SUBROUTINE DAUX

a. Purpose:

1. Evaluate the derivatives of the dependent variables for subsequent integration in PINT1

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, COMP4, INPT4, INTR, INSUS, SUSCMP

c. Subroutines Required:

VPOS, VGORNT, MATRIX, SIMSOL, CTQB, SUSFRC, CTQD, TMCNST, TIRFR, UMOMNT, MTRXIR, MTRXSF

d. Arguments:

None

e. Common Variables Calculated:

DP, DQ, DR, DU, DV, DW, DXCP, DYCP, DZCP, GAM2, GAM3, GAM4, GAM5, GAM6, GAM7, GAM8, GAM9, XIXP, XIYP, XIZP, DDEL1, DDEL2, DDEL3, DDEL4, DPHIF, DPHIR, T1PSI, T2PSI, XIXZP, XIYZP, DDEL1D, DDEL2D, DDEL3D, DDEL4D, DDPSFI, DPHIFD, DPHIRD, DPHITP, DPSIFI, DPSITP, DTHTTP

f. Size:

$B52)_{16} = 2898)_{10}$ bytes

g. Computational Procedure:

1. Test for abnormal program stop (ISTOP≠0) and return if indicated.
2. Calculate time dependent variables by calling subroutine TMCNST.
3. Calculate time dependent inertial terms: XIXP, XIYP, XIZP, XIXZP, XIYZP, GAM2, GAM3, GAM4, GAM5, GAM6, GAM7, GAM8, GAM9. Note that these variables differ with the suspension option in effect, thus branching to the appropriate set of calculation occurs based on ISUS.
4. Call subroutines VPOS and VGORNT to determine the position and orientation of the vehicle.
5. Calculate suspension displacements and velocities depending on suspension option.
6. Call subroutines SUSFRC to calculate suspension forces, and UMOMNT to calculate moments acting on the sprung mass and solid axles (if being used).
7. Depending on the suspension option in effect, call either subroutine MATRIX, MTRXIR or MTRXSF to evaluate the inertial matrix and forcing function stored in the array DMATX.

8. Call subroutine SIMSOL to solve the 10x10 set of simultaneous equations of motion for the 10 derivatives of the dependent variables.
9. Set the solution vector from SIMSOL, DMATX(I,11), to the appropriate variable names and set the remaining 10 derivatives depending on suspension option.
10. Compute the derivatives of the steering degree-of-freedom if in effect as indicated by either INDCRB<0 or IHIT=1 and INDCRB>0.

16. SUBROUTINE DAUXR(NTRA)

a. Purpose:

1. Calculate tire side and circumferential forces
2. Calculate time derivatives of wheel spin velocities

b. Common Blocks Required:

DIMV, COMPN, INPT4, COMP4, INPT5, COMP5, TIRIN, INTR

c. Subroutines Required:

INTPR, ADJTQB

d. Arguments:

NTRA = 1 for initialization of wheel spin derivatives
2 for integration of wheel spin derivatives

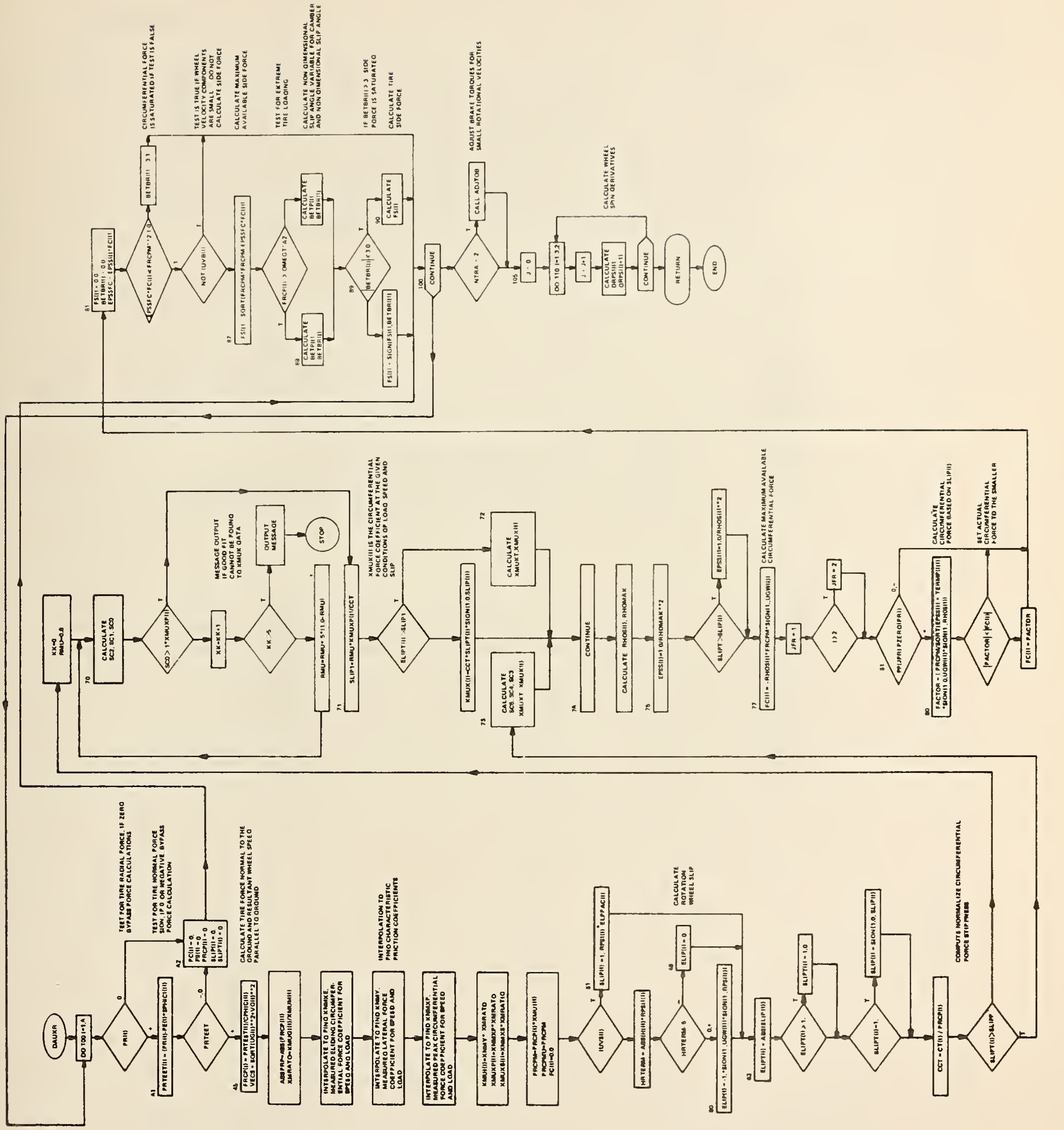
e. Common Variables Calculated:

FC, FS, RRM, UGW, BETP, EPSS, FRCP, RHOS, SLIP, VECS, BETBR, DRPSI, SLIPP, SLIPT, XMUXP, XMUXS, EPSSFC, FRCPMU, FRTEST, HRTERM, NXFRCP, NXUGMU, RHOMAX, SLIPMT, XMUMAT, XMXPMT, XMXSMT, XXFRCP, XXUGMU

f. Size:

$BBA)_{16} = 3002)_{10}$ bytes

g. Computational Procedure:



DAUXR
DO 100 I=1,4
TEST FOR TIRE RADIAL FORCE, IF ZERO BYPASS FORCE CALCULATIONS
TEST FOR TIRE NORMAL FORCE SIGN, IF 0 OR NEGATIVE BYPASS FORCE CALCULATION
CALCULATE TIRE FORCE NORMAL TO THE GROUND AND RESULTANT WHEEL SPEED PARALLEL TO GROUND
INTERPOLATION TO FIND CHARACTERISTIC FRICTION COEFFICIENTS
CALCULATE ROTATION WHEEL SLIP
COMPUTE NORMALIZE CIRCUMFERENTIAL FORCE STIFFNESS
CALCULATE SC2, SC1, SC0
MESSAGE OUTPUT IF GOOD FIT CANNOT BE FOUND TO XMUX DATA
OUTPUT MESSAGE
STOP
CALCULATE XMUXT, XMUX(I)
CONTINUE
CALCULATE RHOS(I), RHOMAX
CALCULATE MAXIMUM AVAILABLE CIRCUMFERENTIAL FORCE
SET ACTUAL CIRCUMFERENTIAL FORCE TO THE SMALLER
FC(I) = FACTOR
CIRCUMFERENTIAL FORCE IS SATURATED IF TEST IS FALSE
TEST IS TRUE IF WHEEL VELOCITY COMPONENTS ARE SMALL DO NOT CALCULATE SIDE FORCE
CALCULATE MAXIMUM AVAILABLE SIDE FORCE
TEST FOR EXTREME TIRE LOADING
CALCULATE TIRE SIDE FORCE
CALCULATE FS(I)
CALL ADJTOB
ADJUST BRAKE TORQUES FOR SMALL ROTATIONAL VELOCITIES
CALCULATE WHEEL SPIN DERIVATIVES
CONTINUE
RETURN
END

17. SUBROUTINE DRIVER

a. Purpose:

1. Compute the front wheel steer angle and angular velocity based on either the path following or skid control modes of driver operation
2. Obtain accelerator pedal deflection or brake pedal force for speed control mode of operation

b. Common Blocks Required:

INPT1, INTG, DIMV, COMP, EINDEX, DRIVTT, DRIVI, DRIVE

c. Subroutines Required:

DRIVP

d. Arguments:

PHIFF - computed front wheel steer angle
PSIFFD - computed front wheel steer angular velocity
JJ - indicator, set$\neq$0 when subroutine DRIVER has calculated PSIFF and PSIFFD

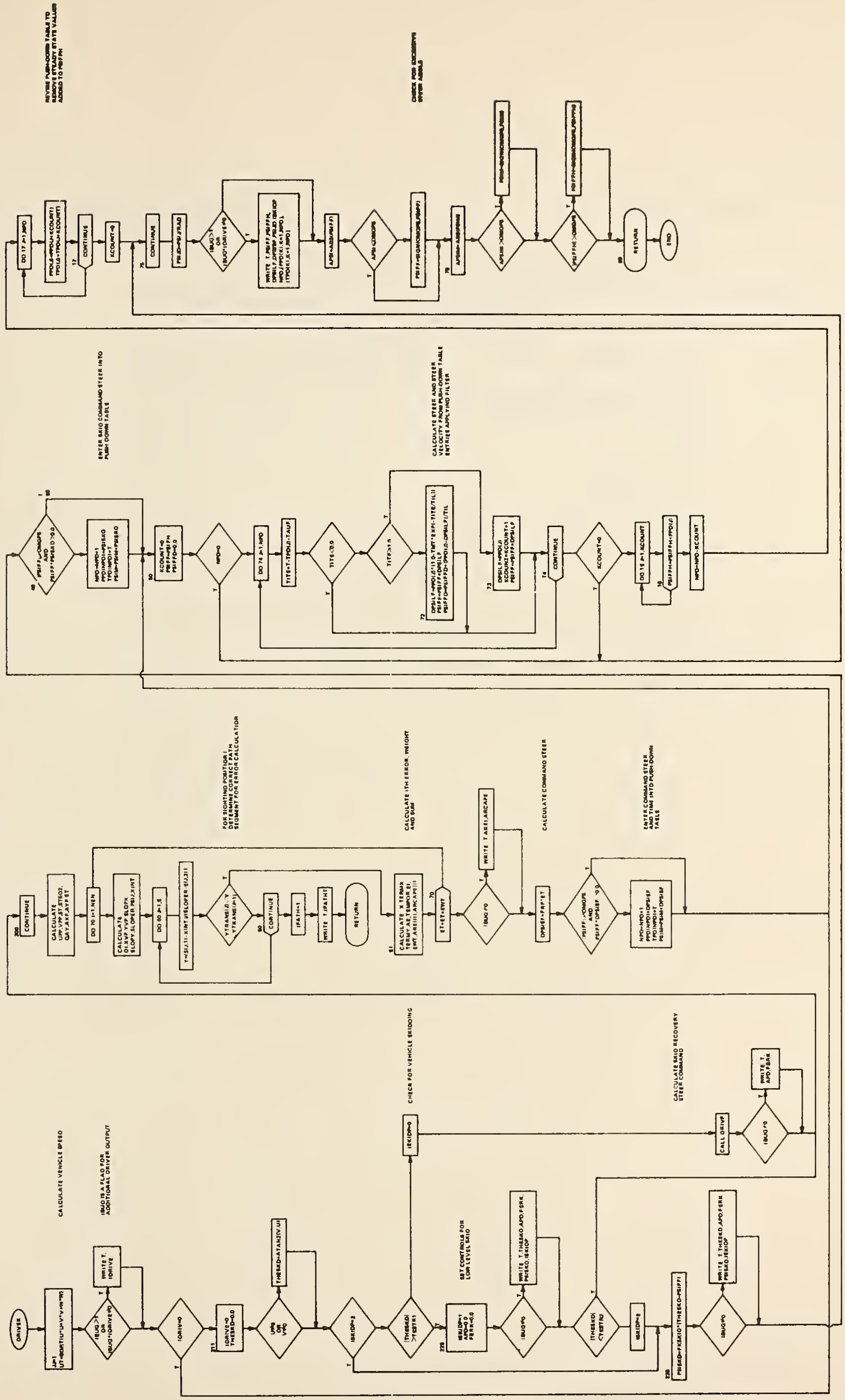
e. Common Variables Calculated:

X, Y, AE, DI, EI, ET, ST, UT, APD, AXP, AYP, EWT, NPD, PPD, QAY, TPD, XVP, YVP, APSI, AREI, FBRK, PSIM, TITE, XINT, APSIM, PSIJD, STS02, TERMX, TERMY, ARCAPE, DPSILF, DPSISF, IDRIVE, IPATHT, ISKIDP, KCOUNT, PHIFFD, PSIFFH, PSISKD, SLOPER, TEMPOR, THESKD

f. Size:

$C06)_{16} = 3078)_{10}$ bytes

g. Computational Procedure:



18. SUBROUTINE DRIVID

a. Purpose:

1. Printout driver control inputs

b. Common Blocks Required:

INPT, HEAD, DRIVI, DRIVE, DRIVTT

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$BB0)_{16} = 2992)_{10}$ bytes

19. SUBROUTINE DRIVP

a. Purpose:

1. Compute accelerator pedal deflection or brake pedal force for driver speed control

b. Common Blocks Required:

DRIVTT, DRIVI, DRIVE

c. Subroutines Required:

None

d. Arguments:

None

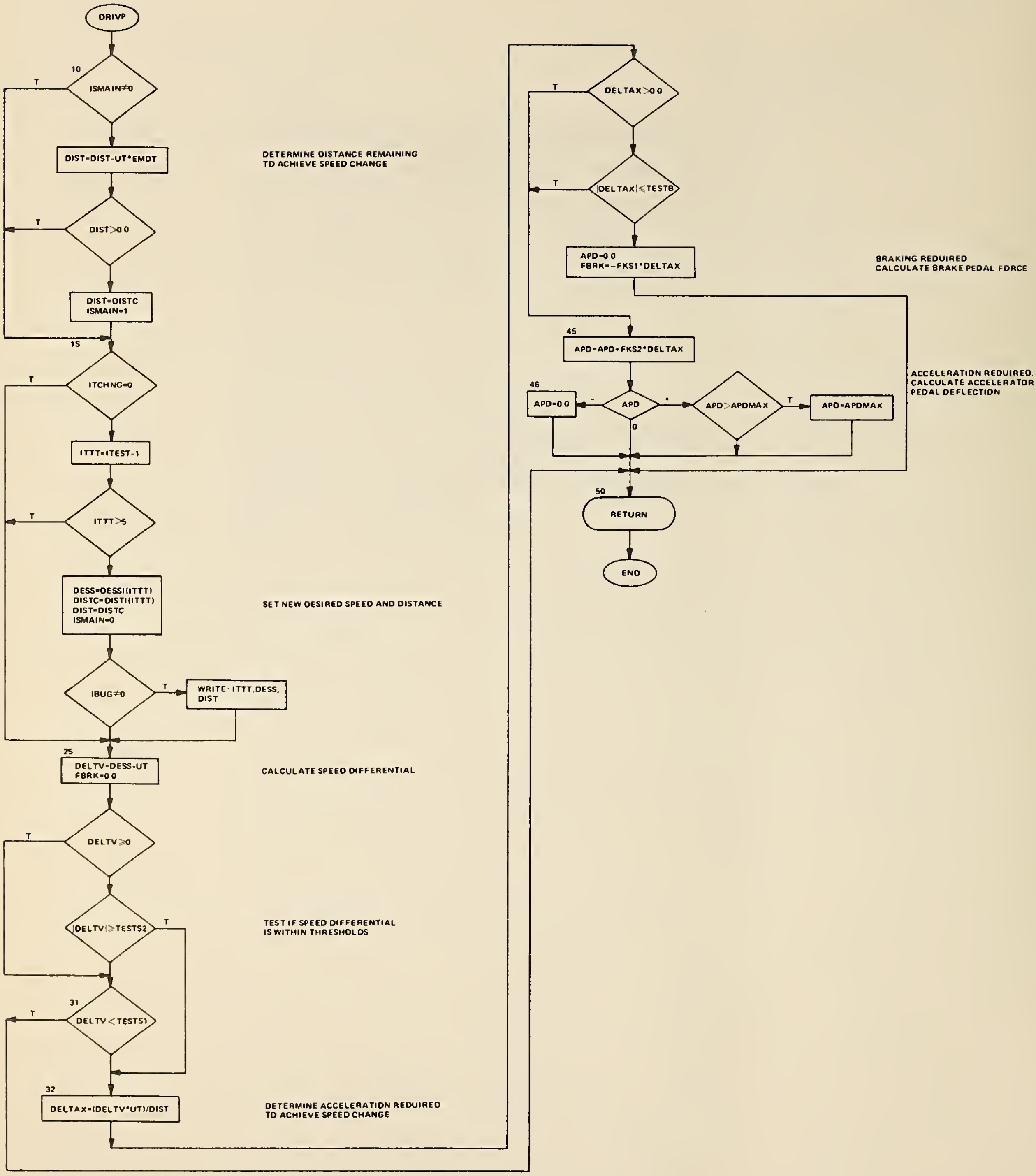
e. Common Variables Calculated:

APD, DESS, DIST, FBRK, DELTV, DISTC, DELTAX, ISMAIN

f. Size:

$24E)_{16} = 590)_{10}$ bytes

g. Computational Procedure:


DRIVP
10
ISMAIN≠0
T
DIST=DIST-UT*EMDT
DETERMINE DISTANCE REMAINING TO ACHIEVE SPEED CHANGE
DIST>0.0
T
DIST=DISTC
ISMAIN=1
15
ITCHNG=0
T
ITTT=ITEST-1
ITTT>5
T
DESS=DESSI(ITTT)
DISTC=DISTI(ITTT)
DIST=DISTC
ISMAIN=0
SET NEW DESIRED SPEED AND DISTANCE
IBUG≠0
T
WRITE ITTT,DESS, DIST
25
DELTV=DESS-UT
FBRK=0.0
CALCULATE SPEED DIFFERENTIAL
DELTV≥0
T
|DELTV|≥TESTS2
T
TEST IF SPEED DIFFERENTIAL IS WITHIN THRESHOLDS
31
DELTV<TESTS1
T
32
DELTAX=(DELTV*UT)/DIST
DETERMINE ACCELERATION REQUIRED TO ACHIEVE SPEED CHANGE
DELTAX>0.0
T
|DELTAX|≤TESTB
T
APD=0.0
FBRK=-FKS1*DELTAX
BRAKING REQUIRED
CALCULATE BRAKE PEDAL FORCE
45
APD=APD+FKS2*DELTAX
ACCELERATION REQUIRED.
CALCULATE ACCELERATOR PEDAL DEFLECTION
46
APD=0.0
-
APD
+
0
APD>APDMAX
T
APD=APDMAX
50
RETURN
END

20. SUBROUTINE DRVCNS

a. Purpose:

1. Calculate variables used in subroutine DRIVER
2. Initialize accelerator pedal deflection for constant speed

b. Common Blocks Required:

INPT, INPT1, COMP, COMPN, DIMV, INPT4, INPT5, COMP5, INTR, DRIVTT, DRIVI, DRIVE

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

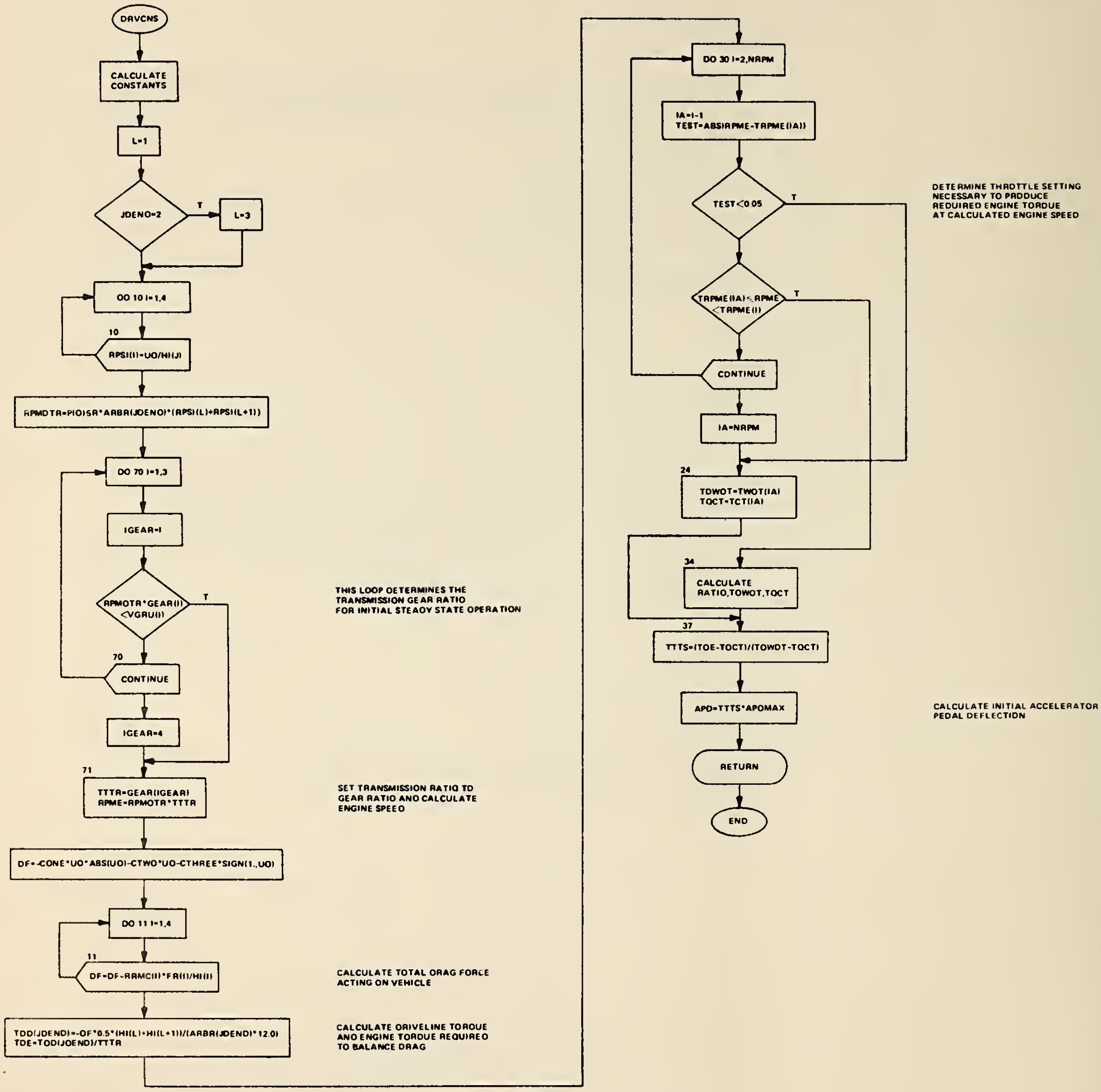
DI, APB, APD, FKD, FKP, TMT, TQD, TQE, DESS, DIST, FBRK, FKS1, FKS2, RPME, RPSI, DISTC, IGEAR, OMGPS, TESTB, TPATH, TRKIN, CONMPH, DELPTH, FKSKID, IDRIVE, ISMAIN, ITCHNG, ITESTT, TCTEST, TESTR1, TESTR2, TESTS1, TESTS2, THESKD, TTPSIT, WEIGHT

f. Size:

$642)_{16} = 1602)_{10}$ bytes

g. Computational Procedure:

1. Compute constants required by DRIVER.
2. Initialize accelerator pedal deflection as shown:


DRVCNS
CALCULATE CONSTANTS
L=1
JDENO=2
T
L=3
OO 10 I=1,4
10
RPSI(I)=UO/HI(J)
RPMOTR=PIO)SR*ARBR(JDENO)*(RPSI(L)+RPSI(L+1))
DO 70 I=1,3
IGEAR=I
RPMOTR*GEAR(I) <VGRU(I)
T
70
CONTINUE
IGEAR=4
71
TTTR=GEAR(IGEAR)
RPME=RPMOTR*TTTR
DF=-CONE*UO*ABS(UO)-CTWO*UO-CTHREE*SIGN(1.,UO)
DO 11 I=1,4
11
DF=DF-RRMC(I)*FR(I)/HI(I)
TOD(JDEND)=-OF*0.5*(HI(L)+HI(L+1))/(ARBR(JDEND)*12.0)
TDE=TOD(JOEND)/TTTR
THIS LOOP OETERMINES THE
TRANSMISSION GEAR RATIO
FOR INITIAL STEAOY STATE OPERATION
SET TRANSMISSION RATIQ TO
GEAR RATIO ANO CALCULATE
ENGINE SPEEO
CALCULATE TOTAL ORAG FORCE
ACTING ON VEHICLE
CALCULATE ORIVELINE TOROUE
ANO ENGINE TORDUE REQUIREO
TO BALANCE DRAG
DO 30 I=2,NRPM
IA=I-1
TEST=ABS(RPME-TRPME(IA))
TEST<0.05
T
TRPME(IA)≤RPME <TRPME(I)
T
CONTINUE
IA=NRPM
24
TOWOT=TWOT(IA)
TOCT=TCT(IA)
34
CALCULATE
RATIO,TOWOT,TOCT
37
TTTS=(TOE-TOCT)/(TOWOT-TOCT)
APO=TTTS*APOMAX
RETURN
END
DETERMINE THROTTLE SETTING
NECESSARY TO PRODUCE
REOUIRED ENGINE TORDUE
AT CALCULATED ENGINE SPEED
CALCULATE INITIAL ACCELERATOR
PEDAL DEFLECTION

21. SUBROUTINE DVDCHK

a. Purpose:

1. This subroutine processes interuptions caused by arithmetic instructions

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$452)_{16} = 1108)_{10}$ bytes

g. Procedure:

A call to DVDCHK processes the following interruptions:

1. fixed point divide exception
2. exponent overflow exception
3. exponent underflow exception
4. floating point divide exception

This subroutine is written in IBM Assembler Language. The services provided are also given by extended FORTRAN error handling.

22. SUBROUTINE GCP(I)

a. Purpose:

1. Compute the coordinates of the tire ground contact point in space
2. Compute the rolling radius of the tire
3. Compute the direction and magnitude of the tire radial force

b. Common Blocks Required:

DIMV, COMP, TIRIN

c. Subroutines Required:

SIMSOL

d. Arguments:

The argument I indicates the wheel number for which calculations are made

e. Common Variables Calculated:

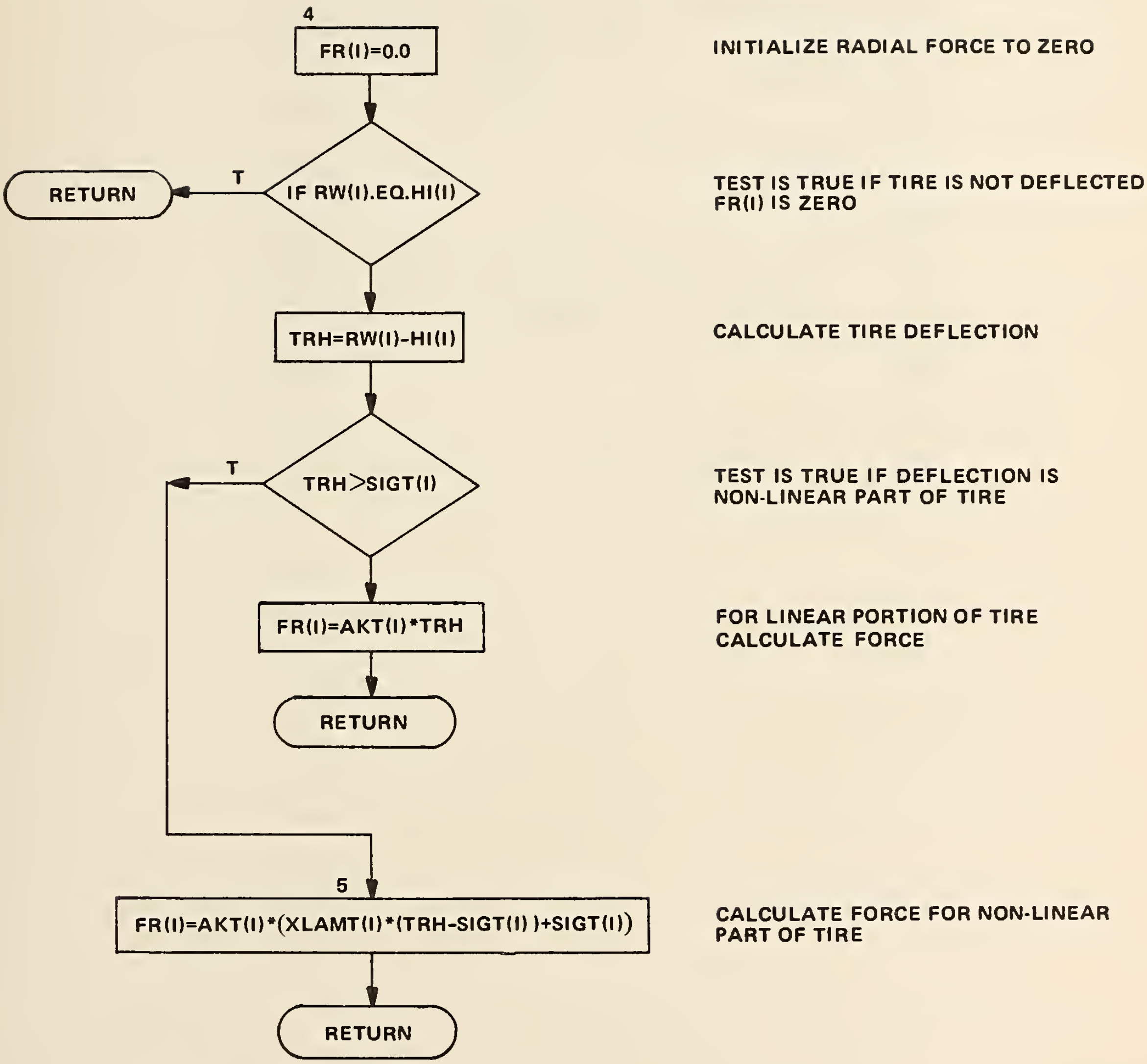
CAR, CBR, CGR, CMTX, DELTA, FR, HI, TRH, TX, TY, TZ, XGPP, XLM1, XLM2, XLM3, YGPP, ZGPP

f. Size:

$36C)_{16} = 876)_{10}$ bytes

g. Computational Procedure:

1. Calculate the coordinates of the ground contact point by simultaneous solution of the intersection of three planes: the wheel plane (normal direction CAYW(I), CBYW(I), CGYW(I)); the ground plane (normal direction CAGZ(I), CBGZ(I), CGGZ(I)); and a plane perpendicular to both passing through the wheel center (normal direction D1(I), D2(I), D3(I)). The simultaneous solution is performed by SIMSOL with the CMTX array containing the above direction cosines and the target array (XLM1(I), XLM2(I), XLM3(I)) contained in the fourth column of CMTX. The solution is returned in the fourth column of CMTX and set to the coordinates of the ground contact point (XGPP(I), YGPP(I), ZGPP(I)).
2. Calculate the distance between the wheel center and ground contact point, DELTA(I).
3. Calculate the direction cosines of the line of action of the tire radial force with respect to the space axes (CAR(I), CBR(I), CGR(I)).
4. Determine the rolling radius, HI(I).
5. Calculate the radial tire force, FR(I), as shown:


4
FR(I)=0.0
RETURN
T
IF RW(I).EQ.HI(I)
TRH=RW(I)-HI(I)
T
TRH>SIGT(I)
FR(I)=AKT(I)*TRH
RETURN
5
FR(I)=AKT(I)*(XLAMT(I)*(TRH-SIGT(I))+SIGT(I))
RETURN
INITIALIZE RADIAL FORCE TO ZERO
TEST IS TRUE IF TIRE IS NOT DEFLECTED
FR(I) IS ZERO
CALCULATE TIRE DEFLECTION
TEST IS TRUE IF DEFLECTION IS
NON-LINEAR PART OF TIRE
FOR LINEAR PORTION OF TIRE
CALCULATE FORCE
CALCULATE FORCE FOR NON-LINEAR
PART OF TIRE

23. SUBROUTINE IDOUT

a. Purpose:

1. Print input values with units and headings

b. Common Blocks Required:

HEAD, INPT, INPT1, COMP, APTABL, INPT3, INPT4, TIRIN, DRIVTT, NEWCRB

c. Subroutines Required:

DRIVID, IDOUTA

d. Arguments:

None

e. Common Variables Calculated:

None

f. Size:

$4548)_{16} = 17736)_{10}$ bytes

24. SUBROUTINE IDOUTA

a. Purpose:

1. Print input table values

b. Common Blocks Required:

HEAD, INPT4, INPT5

c. Subroutines Required:

None

d. Arguments:

HDD - array containing run title
DATE - array containing current date

e. Variables Calculated:

None

f. Size:

$1176)_{16} = 4470)_{10}$ bytes

25. SUBROUTINE INITEQ

a. Purpose:

1. To perform calculations to situate the vehicle in initial vertical equilibrium on flat, level terrain

b. Common Blocks Required:

INPT, COMP, DIMV, COMPN, INSUS, TIRIN

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

FR, HI, ZF, ZR

f. Size:

$324)_{16} = 804)_{10}$ bytes

g. Computational Procedure:

If ZF and ZR are input as zero, this subroutine calculates these variables based on the requirement for initial vertical equilibrium of the vehicle. Also calculated are tire radial forces and rolling radii.

26. SUBROUTINE INPUT

a. Purpose:

1. Obtain card input
2. Print card images

b. Common Blocks Required:

None

c. Subroutines Required:

BLK01, BLK02, BLK03, BLK04, BLK05, BLK06

d. Arguments:

None

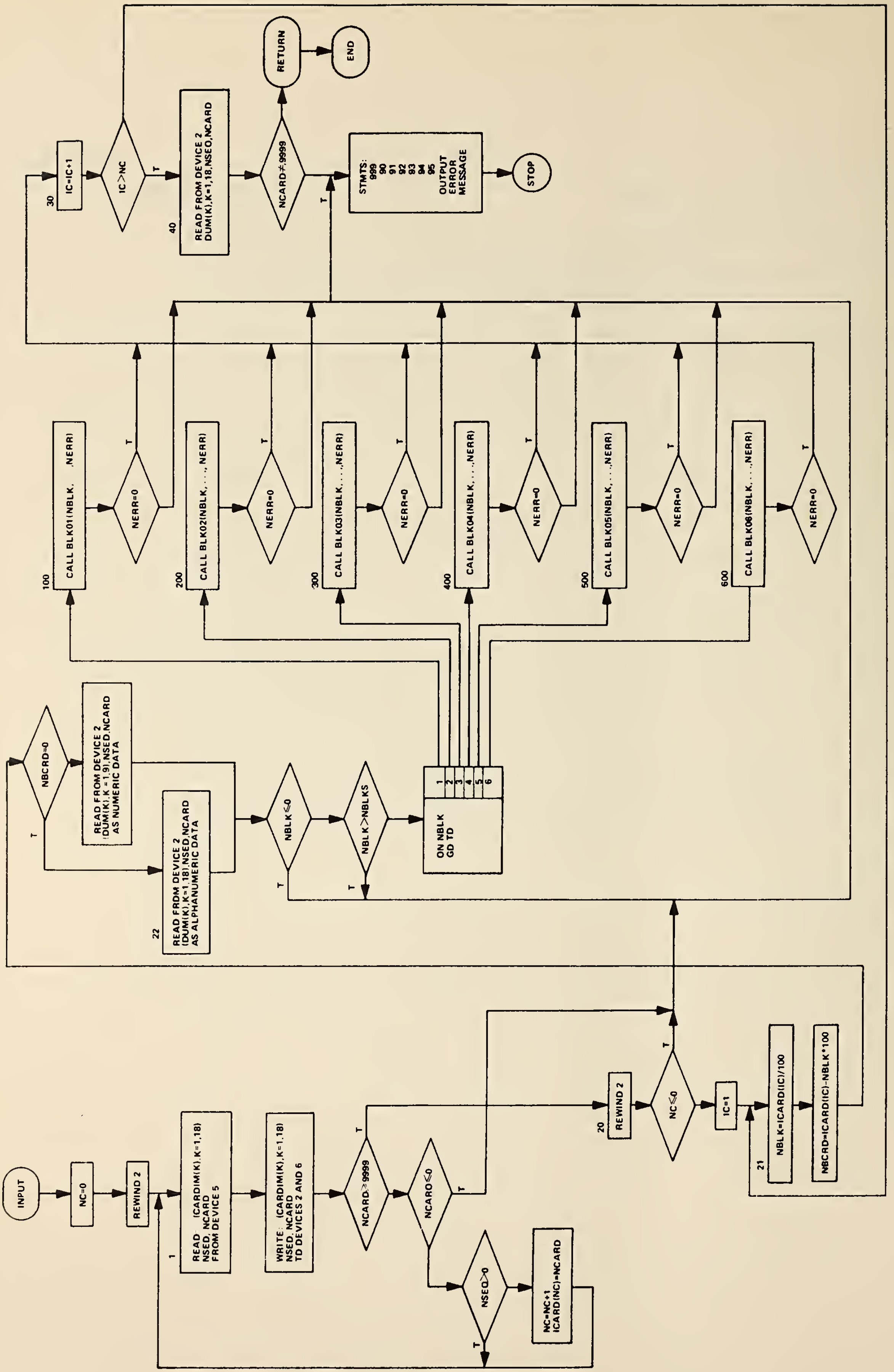
e. Common Variables Calculated:

None

f. Size:

$D5E)_{16} = 3422)_{10}$ bytes

g. Computational Procedure:


INPUT
NC=0
REWIND 2
1
READ (CARDIM(K),K=1,18) NSED, NCARD FROM DEVICE 5
WRITE: (CARDIM(K),K=1,18) NSED, NCARD TO DEVICES 2 AND 6
NCARD≥9999
T
NCARD≤0
T
NSED>0
T
NC=NC+1 ICARD(NC)=NCARD
20
REWIND 2
NC≤0
T
IC=1
21
NBLK=ICARD(IC)/100
NBCRD=ICARD(IC)-NBLK*100
NBCRD=0
T
READ FROM DEVICE 2 (DUM(K),K=1,9),NSED,NCARD AS NUMERIC DATA
22
READ FROM DEVICE 2 (DUM(K),K=1,18),NSED,NCARD AS ALPHANUMERIC DATA
NBLK≤0
T
NBLK>NBLKS
T
ON NBLK GO TO
1
2
3
4
5
6
100
CALL BLK01(NBLK, ,NERR)
NERR=0
T
200
CALL BLK02(NBLK, . . . ,NERR)
NERR=0
T
300
CALL BLK03(NBLK, . . . ,NERR)
NERR=0
T
400
CALL BLK04(NBLK, . . . ,NERR)
NERR=0
T
500
CALL BLK05(NBLK, . . . ,NERR)
NERR=0
T
600
CALL BLK06(NBLK, . . . ,NERR)
NERR=0
T
30
IC=IC+1
IC>NC
T
40
READ FROM DEVICE 2 DUM(K),K=1,18,NSED,NCARD
NCARD≠9999
T
RETURN
END
STMTS: 999 90 91 92 93 94 95 OUTPUT ERROR MESSAGE
STOP

27. SUBROUTINE INTPR(F,XMA, ALP, NM, NA, XM, AX, ANS, ERR, ND1)

a. Purpose:

1. Interpolate a two dimensional table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

F - two dimensional array containing values to be interpolated
XMA - one dimensional array containing values corresponding to the first dimension of F
ALP - one dimensional array containing values corresponding to the second dimension of F
NM - size of the XMA array
NA - size of the ALP array
XM - target value along the XMA dimension
AX - target value along the ALP dimension
ANS - interpolated value of the F array at (XM,AX)
ERR - error flag
ND1 - size of the first dimension of the F array

e. Common Variables Calculated:

None

f. Size:

$428)_{16} = 1064)_{10}$ bytes

g. Computational Procedure:

The answer is obtained by linear interpolation first along the XMA dimension, then along the ALP dimension.

28. SUBROUTINE INTRPL

a. Purpose:

1. To obtain a quadratic interpolation of a one-dimensional table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

TABLE - one-dimensional array of data
XMIN - minimum abscissa value
XMAX - maximum abscissa value
DX - abscissa increment
X - abscissa value at which ordinate is desired
Y - ordinate at X

e. Common Variables Calculated:

None

f. Size:

$3C2)_{16} = 962)_{10}$ bytes

g. Procedure:

1. Quadratic interpolation of the values of TABLE at X
2. ENTRY INTRPC also includes the additional argument SLOPE which is calculated as $\frac{d(TABLE)}{dx}$ at X

29. SUBROUTINE INTRP5(INDX)

a. Purpose:

1. Calculate the elevation and slopes under the wheel indicated by the argument INDX
2. Set the nominal friction coefficients according to the table for the wheel location

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL

c. Subroutines Required:

None

d. Arguments:

INDX - wheel number for which calculations are to be made

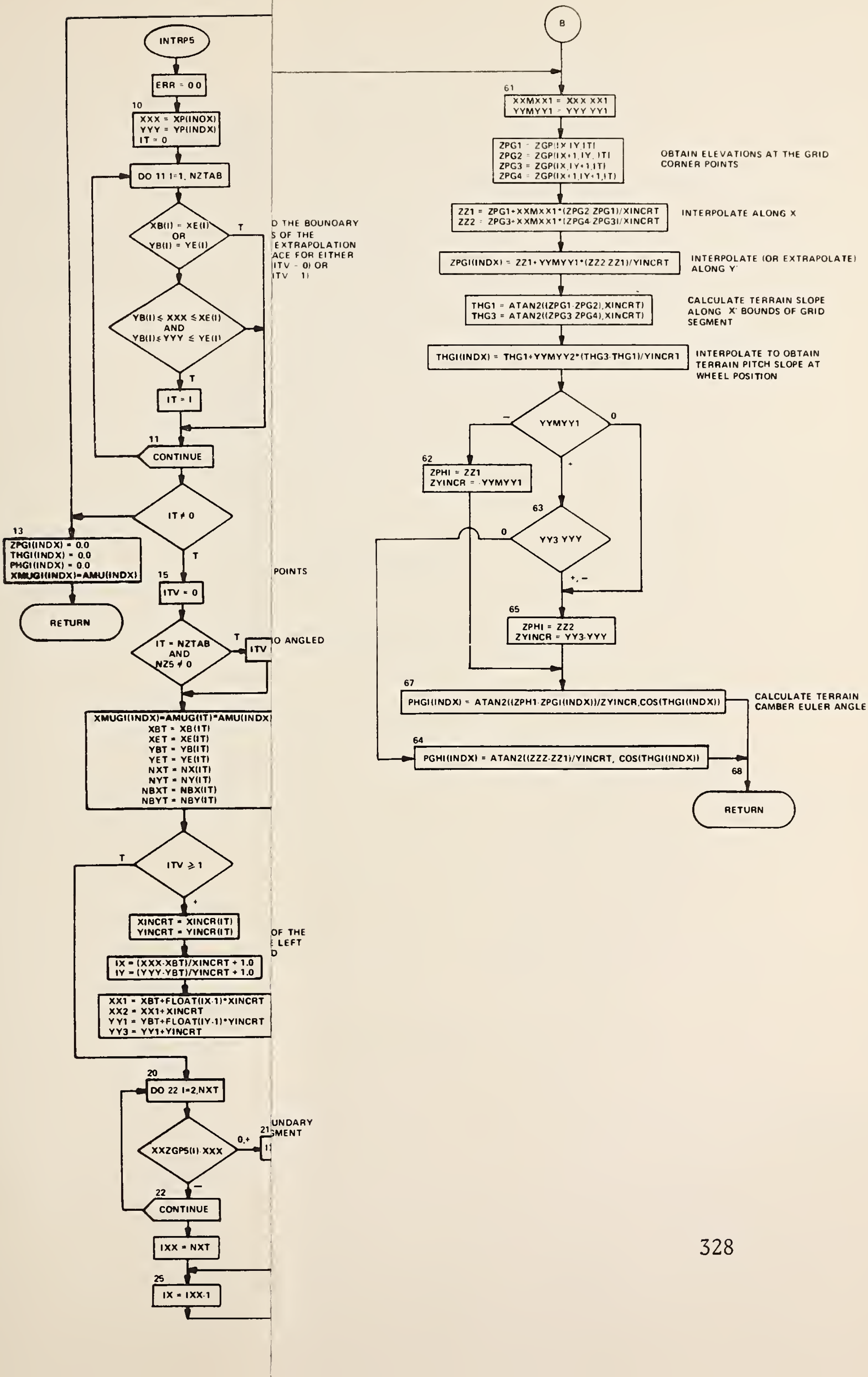
e. Common Variables Calculated:

IX, IY, PHG1, THG1, THG1, XMUG1, XXX, XX1, XX2, YYY, YY1, YY2, ZPG1

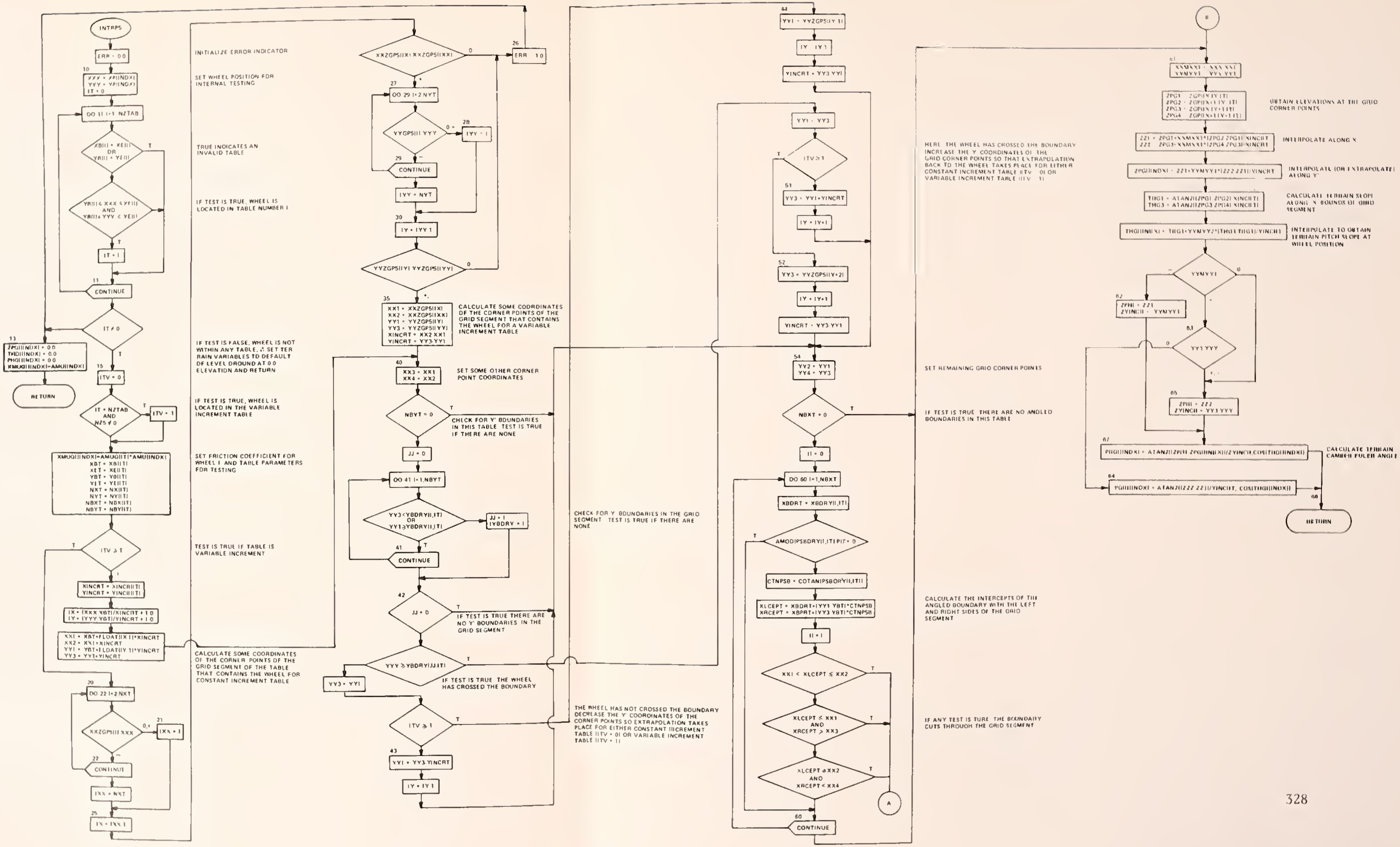
f. Size:

$129C)_{16} = 4764)_{10}$ bytes

g. Computational Procedure:



INTRP5
ERR = 0.0
XXX = XP(INDX)
YYY = YP(INDX)
IT = 0
DO 11 I=1, NZTAB
XB(I) = XE(I) OR YB(I) = YE(I)
YB(I) ≤ XXX ≤ XE(I) AND YB(I) ≤ YYY ≤ YE(I)
IT = I
CONTINUE
IT ≠ 0
ZPGI(INDX) = 0.0
THGI(INDX) = 0.0
PHGI(INDX) = 0.0
XMUGI(INDX)=AMU(INDX)
RETURN
ITV = 0
IT = NZTAB AND NZ5 ≠ 0
XMUGI(INDX)=AMUG(IT)*AMU(INDX)
XBT = XB(IT)
XET = XE(IT)
YBT = YB(IT)
YET = YE(IT)
NXT = NX(IT)
NYT = NY(IT)
NBXT = NBX(IT)
NBYT = NBY(IT)
ITV ≥ 1
XINCRT = XINCR(IT)
YINCRT = YINCR(IT)
IX = (XXX-XBT)/XINCRT + 1.0
IY = (YYY-YBT)/YINCRT + 1.0
XX1 = XBT+FLOAT(IX-1)*XINCRT
XX2 = XX1+XINCRT
YY1 = YBT+FLOAT(IY-1)*YINCRT
YY3 = YY1+YINCRT
DO 22 I=2,NXT
XXZGP5(I)-XXX
CONTINUE
IXX = NXT
IX = IXX-1
XXMXX1 = XXX-XX1
YYMYY1 = YYY-YY1
ZPG1 = ZGP(IX,IY,IT)
ZPG2 = ZGP(IX+1,IY,IT)
ZPG3 = ZGP(IX,IY+1,IT)
ZPG4 = ZGP(IX+1,IY+1,IT)
OBTAIN ELEVATIONS AT THE GRID CORNER POINTS
ZZ1 = ZPG1+XXMXX1*(ZPG2-ZPG1)/XINCRT
ZZ2 = ZPG3+XXMXX1*(ZPG4-ZPG3)/XINCRT
INTERPOLATE ALONG X
ZPGI(INDX) = ZZ1+YYMYY1*(ZZ2-ZZ1)/YINCRT
INTERPOLATE (OR EXTRAPOLATE) ALONG Y
THG1 = ATAN2((ZPG1-ZPG2),XINCRT)
THG3 = ATAN2((ZPG3-ZPG4),XINCRT)
CALCULATE TERRAIN SLOPE ALONG X BOUNDS OF GRID SEGMENT
THGI(INDX) = THG1+YYMYY2*(THG3-THG1)/YINCRT
INTERPOLATE TO OBTAIN TERRAIN PITCH SLOPE AT WHEEL POSITION
YYMYY1
ZPHI = ZZ1
ZYINCR = -YYMYY1
YY3-YYY
ZPHI = ZZ2
ZYINCR = YY3-YYY
PHGI(INDX) = ATAN2((ZPHI-ZPGI(INDX))/ZYINCR,COS(THGI(INDX)))
CALCULATE TERRAIN CAMBER EULER ANGLE
PGHI(INDX) = ATAN2((ZZZ-ZZ1)/YINCRT, COS(THGI(INDX)))
RETURN


INTRPS
INITIALIZE ERROR INDICATOR
SET WHEEL POSITION FOR INTERNAL TESTING
TRUE INDICATES AN INVALID TABLE
IF TEST IS TRUE, WHEEL IS LOCATED IN TABLE NUMBER I
CONTINUE
RETURN
IF TEST IS FALSE, WHEEL IS NOT WITHIN ANY TABLE, ∴ SET TERRAIN VARIABLES TO DEFAULT OF LEVEL GROUND AT 0.0 ELEVATION AND RETURN
IF TEST IS TRUE, WHEEL IS LOCATED IN THE VARIABLE INCREMENT TABLE
SET FRICTION COEFFICIENT FOR WHEEL I AND TABLE PARAMETERS FOR TESTING
TEST IS TRUE IF TABLE IS VARIABLE INCREMENT
CALCULATE SOME COORDINATES OF THE CORNER POINTS OF THE GRID SEGMENT OF THE TABLE THAT CONTAINS THE WHEEL FOR CONSTANT INCREMENT TABLE
CALCULATE SOME COORDINATES OF THE CORNER POINTS OF THE GRID SEGMENT THAT CONTAINS THE WHEEL FOR A VARIABLE INCREMENT TABLE
SET SOME OTHER CORNER POINT COORDINATES
CHECK FOR Y' BOUNDARIES IN THIS TABLE. TEST IS TRUE IF THERE ARE NONE
IF TEST IS TRUE THERE ARE NO Y' BOUNDARIES IN THE GRID SEGMENT
IF TEST IS TRUE, THE WHEEL HAS CROSSED THE BOUNDARY
CHECK FOR Y' BOUNDARIES IN THE GRID SEGMENT. TEST IS TRUE IF THERE ARE NONE
THE WHEEL HAS NOT CROSSED THE BOUNDARY. DECREASE THE Y' COORDINATES OF THE CORNER POINTS SO THAT EXTRAPOLATION TAKES PLACE FOR EITHER CONSTANT INCREMENT TABLE (ITV = 0) OR VARIABLE INCREMENT TABLE (ITV = 1)
HERE THE WHEEL HAS CROSSED THE BOUNDARY. INCREASE THE Y' COORDINATES OF THE GRID CORNER POINTS SO THAT EXTRAPOLATION BACK TO THE WHEEL TAKES PLACE FOR EITHER CONSTANT INCREMENT TABLE (ITV = 0) OR VARIABLE INCREMENT TABLE (ITV = 1)
SET REMAINING GRID CORNER POINTS
IF TEST IS TRUE THERE ARE NO ANGLED BOUNDARIES IN THIS TABLE
CALCULATE THE INTERCEPTS OF THE ANGLED BOUNDARY WITH THE LEFT AND RIGHT SIDES OF THE GRID SEGMENT
IF ANY TEST IS TURE THE BOUNDARY CUTS THROUGH THE GRID SEGMENT
OBTAIN ELEVATIONS AT THE GRID CORNER POINTS
INTERPOLATE ALONG X
INTERPOLATE (OR EXTRAPOLATE) ALONG Y'
CALCULATE TERRAIN SLOPE ALONG X BOUNDS OF GRID SEGMENT
INTERPOLATE TO OBTAIN TERRAIN PITCH SLOPE AT WHEEL POSITION
CALCULATE TERRAIN CAMBER EULER ANGLE
RETURN

30. SUBROUTINE MATRIX

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the independent front/solid axle rear suspension option
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1,10) for the independent front/solid rear axle option

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

DMATX, GCTCP, GCTSP

f. Size:

$72C)_{16} = 1836)_{10}$ bytes

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero.
2. Calculate the elements of DMATX.

31. SUBROUTINE MTRXIR

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the independent rear suspension option
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1,10) for the independent rear suspension option

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL, SUSCMP

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

DMATX, GCTCP, GCTSP

f. Size:

$4C0)_{16} = 1216)_{10}$ bytes

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero.
2. Calculate the elements of DMATX.

32. SUBROUTINE MTRXSF

a. Purpose:

1. Evaluate the elements of the inertial matrix for the ten coupled degrees of freedom (DMATX(I,J), I = 1,10, J = 1,10) for the solid front axle option
2. Evaluate the forcing column matrix for the ten coupled degrees of freedom (DMATX(I,11), I = 1,10) for the solid front axle option

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, ADTNL, SUSCMP, INSUS

c. Subroutines Required:

CLEAR

d. Arguments:

None

e. Common Variables Calculated:

DMATX, GCTCP, GCTSP

f. Size:

$962)_{16} = 2402)_{10}$ bytes

g. Computational Procedure:

1. Call CLEAR to zero the DMATX. This is necessary since the subroutine which decouples the equations of motion also destroys the DMATX in the process and may leave meaningless values in array elements which should be zero
2. Calculate the elements of DMATX

33. SUBROUTINE OUTPUT(IND)

a. Purpose:

1. Print output page titles and output data

b. Common Blocks Required:

HEAD, INPT, INTG, DIMV, COMP, COMPN, ADTNL, INSUS, SUSCMP, BARSTR

c. Subroutines Required:

None

d. Arguments:

If IND = 0, an output line counter is initialized to zero.

e. Common Variables Calculated:

None

f. Size:

$40E0)_{16} = 16608)_{10}$ bytes

g. Computational Procedure:

Each time a call to this subroutine is executed, an output line of data is written to FORTRAN devices 11 through, at most, 29. The number of devices actually written to is dependent on the indicators contained in the NPAGE array. These indicators are set either by the user on input card 104 or by the program depending on the options in use.

On either the first call to the subroutine with IND $\neq$ 0 or after 50 lines of data have been written, page headings are written for each page of data.

An entry point, THPLOT, is provided to write static and dynamic data to FORTRAN device 3 for the purpose of subsequent plotting of time history data.

34. FUNCTION PARI(NN,IA, TSEC,X,Y)

a. Purpose:

1. Lagrangian interpolation

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

NN - size of X and Y arrays
IA - subscript of first tabular point
TSEC - target value of the X dimension
X - array containing values of the abscissa
Y - array containing values of the ordinate

e. Common Variables Calculated:

None

f. Size:

$234)_{16} = 564)_{10}$ bytes

35. SUBROUTINE PINT1(IN, MODE, N, X, H, Y, YP, A)

a. Purpose:

1. To integrate a system of N ordinary differential equations of the first order

b. Common Blocks Required:

None

c. Subroutines Required:

DAUX

d. Arguments:

IN is the control word (= 1 or 2) for initialization or to integrate one step-size;

IN = 1 - to set up the routine for integration;
IN = 2 - to integrate one step-size;

MODE is the option word (= 0, 1 or 2) for using one of the three modes of integration. When MODE equals

0 - the Adams-Moulton variable step-size is used;
1 - the Runge-Kutta fixed step-size is used;
2 - the Adams-Moulton fixed step-size is used;

N	is the number of first order differential equations;
X	is the independent or source variable;
H	is the step-size or increment in the source variable;
Y	is the array of dependent or target variables updated by PINT1;
YP	is the array of first derivatives of the target variables Y(N) computed in the subroutine DAUX;
A	is an array of 6 cells containing the parameters $(\bar{E}, \bar{M}, \alpha, h_{max}, h_{min}, \beta)$ needed for the variable mode only;
A(1)	$(\equiv \bar{E})$ is an upper bound on the truncation error (the number of significant digits which the user desired to preserve locally) for the variable Adams-Moulton method, normally $10^{-8} < A(1) < 10^{-3}$;
A(2)	$(\equiv \bar{M})$ is a positive number from which the lower bound on the truncation error is computed. In particular, when A(2) is zero the routine used the normal value of 100 and in all other cases the lower bound is computed as the quotient of A(1) by A(2);
A(3)	(α) is a positive number used to prevent unnecessary reduction in the variable step-size when the dependent variables are sufficiently small. When A(3) is zero the routine uses the normal value of one;
A(4)	(h_{max}) is a positive upper bound for the magnitude of the variable step-size. If A(4) is zero the routine assumes there is no upper bound;

A(5) ($\equiv h_{min}$) is positive lower bound for the magnitude of the variable step-size. The routine assumes there is no lower bound when A(5) is zero;

A(6) ($\equiv \beta$) is a positive number between zero and one used to increase or decrease the variable step-size. When A(6) is zero the routine assumes the value of one-half.

IN, N and MODE are integers while X, H, Y, YP and A are all single precision floating point numbers.

The arguments X, H, Y, YP, of the PINT1 calling sequence must be in a COMMON type statement.

Before executing the first PINT1 call, the user must initialize X, H and each of the Y(N) variables. The first call must use control word (IN = 1) to set up the routine for integration. The control word (IN = 2) may be used any number of times after the first to integrated one step-size, provided X, H and Y have not been redefined between integration steps.

e. Common Variables Calculated:

None

f. Size:

$1B2C)_{16} = 6956)_{10}$ bytes

g. Computational Procedure:

In this routine the user is allowed an option of using either the Runge-Kutta classical fourth-order method as modified by E. K. Blum or the Adams-Moulton predictor-corrector method using the Runge-Kutta method for starting the process.

Let the system of equations to be solved be given in the form

$$y_i = f_i\,(x, y_1, y_2, \cdots, y_N) \tag{1.1}$$

$$y_i\,(x_0) = y_{i0} \qquad i = 1, 2, \ldots, N$$

Let y_{in} be the value of y_i at $x = x_n$ and f_{in} the derivative of y_i at $x = x_n$. If h is the increment (step-size) of the independent variable x, the classical Runge-Kutta fourth-order method uses the formulas

$$\begin{aligned}
K_{i1} &= h f_i(x_n, y_{in}) \\
K_{i2} &= h f_i(x_n + 1/2 h, \; y_{in} + 1/2 K_{i1}) \\
K_{i3} &= h f_i(x_n + 1/2 h, \; y_{in} + 1/2 K_{i2}) \\
K_{i4} &= h f_i(x_n + h, \; y_{in} + K_{i3}) \\
y_{i,n+1} &= y_{in} + 1/6(K_{i1} + 2K_{i2} + 2K_{i3} + K_{i4})
\end{aligned} \tag{1.2}$$

where $i = 1, 2, \ldots N$

The E. K. Blum Modification:

The following recursive form of the E. K. Blum's exact modification of the Runge-Kutta is used in this routine:

$$\left\{\begin{aligned} z_0 &= y_n \\ q_0 &= y_n \\ p_0 &= h f(z_0) \end{aligned}\right. \qquad \text{at } x = x_0 \tag{2.1}$$

$$\left\{\begin{aligned} z_1 &= z_0 + p_0/2 \\ q_1 &= p_0 \\ p_1 &= h f(z_1) \end{aligned}\right. \qquad \text{at } x = x_0 + h/2 \tag{2.2}$$

$$\left\{\begin{aligned} z_2 &= z_1 + p_1/2 - q_1/2 \\ q_2 &= q_1/6 \\ p_2 &= h f(z_2) - p_1/2 \end{aligned}\right. \qquad \text{at } x = x_0 + h/2 \tag{2.3}$$

$$\left\{\begin{aligned} z_3 &= z_2 + p_2 \\ q_3 &= q_2 + p_2 \\ p_3 &= h f(z_3) + 2p_2 \end{aligned}\right. \tag{2.4}$$

$$y_{i,n+1} \equiv z_4 = z_3 + q_3 + p_3/6 \tag{2.5}$$

(we omit the subscript i from each of the vectors z_j, q_j and p_j for reasons of economy)

The main advantage of the modified Runge-Kutta formulas is that they reduce considerably the rounding error arising from the unavoidable use of digital numbers and pseudo-operations.

Adams-Moulton Predictor-Corrector Method:

The routine uses the following formulas for the system (1.1):

$$y_{i,n+1}^{[p]} = y_{i,n} + h/24(55 f_{i,n} - 59 f_{i,n-1} + 37 f_{i,n-2} - 9 f_{i,n-3}) \quad (3.1)$$

$$y_{i,n+1} = y_{i,n+1}^{[c]} = y_{i,n} + h/24(9 f_{i,n+1}^{[p]} + 19 f_{i,n} - 5 f_{i,n-1} + f_{i,n-2}) \quad (3.2)$$

The starting values needed in the predictor formula (3.1) are obtained using the Runge-Kutta-Blum (RKB) method. In the evaluation of y_i at $x = x_{n+1}$ the predictor and corector formulas are applied only once so that only two derivative evaluations ($f_{i,n+1}^{[p]}$ and $f_{i,n}$) are needed for each Adams-Moulton (variable or fixed step-size) integration step.

The Variable Adams-Moulton:

The step-size h to be used in the variable mode is determined mainly by:

$$E_{n+1} = \underset{i}{MAX} \frac{y_{i,n+1}^{[p]} - y_{i,n+1}^{[c]}}{14 D_i} \quad (3.3)$$

$$D_i = \underset{i}{MAX} \quad y_{i,n+1}^{[c]} , \alpha \qquad , i = 1, 2, \ldots, N$$

where

E_{n+1} is the local truncation error estimate in the actual evaluation of y_{n+1} ; α (> 0) is a constant used to prevent unnecessary reductions in $|h|$ whenever $|y_{i,n+1}|$ is small (normally the routine will set α = 1, unless otherwise specified by the user).

Let

$\bar{E}$ be the upper bound on the truncation error estimate, specified by the user, that is the number of significant digits which the user desires to preserve locally throughout the integration. Normally $\bar{E}$ should be in the range $10^{-8} \leq \bar{E} \leq 10^{-3}$ and in double precision $\bar{E}$ should be in the range $10^{-16} \leq \bar{E} \leq 15^{12}$;

M (> 0) be a constant, specified by the user, from which a lower bound $\underline{E} = M^{-1}\bar{E}$ is obtained (normally M range from 50 to 150 and in double precision from 1000 to 1500);

β be a constant between 0 to 1 used to increase or decrease the step-size. The routine will take $\beta = 1/2$ unless β is otherwise specified by the user.

The step-size h will be then increased or decreased according to the following inequalities:

If

(4.1)	$E_{n+1} > \bar{E}$	the step-size is reduced to βh, where $0 < \beta < 1$;
(4.2)	$M^{-1}\bar{E} < E_{n+1} < \bar{E}$	the step-size remains unchanged;
(4.3)	$E_{n+1} < M^{-1}\bar{E}$	for 3 successive integration steps the step-size is increased to h/β.

Increasing and Decreasing the Step-Size:

The starting values, the first three successive points after the initial point p_0, for the Adams-Moulton formulas are always obtained using the RKB method whenever the interval size is changed, just as at the beginning of an integration.

In the variable mode if the starting values, the first three successive points, have been obtained using the RKB method then the next point is computed using the Adams-Mouton predictor-corrector formulas (3.1) and (3.2). Whenever the truncation error at this point calls for a decrease in h the routine returns to the initial point p_0 and computes new starting values with the decreased value of h. However, if the step-size is to be decreased at a point p_i, where the preceeding point p_{i-1} was computed in the variable mode and the inequality (4.2) held at p_{i-1}, then a new start is initiated at p_{i-1} with decreased value of $|h|$.

If for three successive variable integration steps p_{i-1}, p_i and p_{i+1} inequality (4.3) holds, then a new start is initiated at p_{i+1} with the increased value of $|h|$. After an interval is increased, the routine prevents increasing again until 6 more points have been complete. However, the routine may decrease the interval as often as necessary. The truncation error test based on (3.3) will guarantee that the local error does not exceed $\bar{E}$, however the cumulative error will usually exceed $\bar{E}$. Hence $\bar{E}$ should be chosen sufficiently small to allow for an accumulation of truncation error.

The user must always provide a starting value for h and he may, if desired, specify a maximum value of $|h|$, h_{max} beyond which the routine will not increase $|h|$ and a minimum value of $|h|$, h_{min}, below which it will not decrease . If no value is specified for h_{max} and h_{min} the routine will set the values at 10^3 and 10^{-17}, respectively.

Negative values of h may be used for backward integration.

Control and DAUX:

There are two entries to this routine. The first (control word = 1) must be used once at the beginning to set up the routine for integration of a given set of N differential equations. The second entry (control word = 2) may be used any number of times after the first to integrate all y_i from x to $x+h$.

Whenever the control word is 1 the routine uses the auxiliary subroutine DAUX to evaluate the derivatives at the initial point $x = x_0$ and returns with all y_i unchanged. The routine also checks and sets up the six parameter words $\bar{E}$, M , α , h_{max} , h_{min} and β needed in the variable mode of operation. Before executing the initialization entry, the user must have already set up the appropriate values for x , h and y_i i = 1, 2, ... N . Ordinarily, after an execution of the second entry all y_i assume new values, x will have been advanced to the value $x+h$ and h will be unchanged, unless in the variable mode. On exit the values y_i are always these which correspond to the point $x+h$ and y_i .

Whenever an integration step involves RKB integration, four derivative evaluations are needed, mainly

$$
\begin{aligned}
& f_i(x_n + 1/2\, h \, , \, y_{in} + 1/2\, K_{i1}) \\
& f_i(x_n + 1/2 h, \, y_{in} + 1/2\, K_{i2}) \\
& f_i(x_n + h \, , \, y_{in} + K_{i3} \quad) \\
& y_{i,n+1} = f_i(x_n + h, \, y_{n+1})
\end{aligned}
\qquad (5.1)
$$

where the K_{ij} are given by (1.2) and modified by (2.1)-(2.5). In the fixed predictor-corrector mode, the first three integration entries involve RKB integration and subsequent ones involve AM integration. Each AM integration step requires two derivative evaluations.

$$
\begin{aligned}
f_{i,n+1}^{[P]} &= f_i(x_n + h \, , \, y_{i,n+1}^{[P]}) \\
y'_{i,n+1} &= f_i(x_n + h, \, y_{n+1})
\end{aligned}
\qquad (5.2)
$$

A particular integration set up, in the variable mode, may involve either AM or RKB or both.

References:

(1) SHARE Write-Up No. 0602 (D2RWINT)
(2) SHARE Write-Up No. 0450 (D2RDE2F)
(3) Blum, K. E., A Modification of the Runge-Kutta Fourth Order Method, Mathematics of Computation, April 1962, pp. 176-187

36. SUBROUTINE PLOTTP(IPLT)

a. Purpose:

1. Write output to FORTRAN device 1 for post-processing graphic displays

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, COMPN, TIRIN

c. Subroutines Required:

None

d. Arguments:

IPLT controls the type of record written; static, dynamic or end of data, for values of IPLT of 1, 2 and 3, respectively

e. Common Variables Calculated:

None

f. Size:

$324)_{16} = 804)_{10}$ bytes

g. Computational Procedure:

1. If IPLT = 1 a static header record is written to device 1 consisting of the following variables: HED, DADE, A, B, TS, ZR, RHO, ZF, RW, TF, TR
2. If IPLT = 2 a dynamic record is written consisting of: T, XCP, YCP, ZCP, PHIT, THETT, PSIT, DEL1, DEL2, DEL3, PHIR, PSI1, PHI2, (XGPP(I), YGPP(I), ZGPP(I), I = 1,4), (ICONTW(I), I = 1,4) ·
   Note: ICONTW is an indicator. If 1, wheel I is rolling; if -1, wheel I is skidding; if 0, wheel I is off the ground.
3. If IPLT = 3, an end of data record consisting of 30 words of -9999.0 is written.

37. <u>SUBROUTINE RUFFRC(I,ZGM)</u>

a. Purpose:

1. To determine an equival
contact point from the
when the road roughness

b. Common Blocks Required:

INPT1, DIMV, COMP, COMPN, T

c. Subroutines Required:

INTRPL

d. Arguments:

I = wheel number for whic
ZGM = single dimensioned ar
roughness data

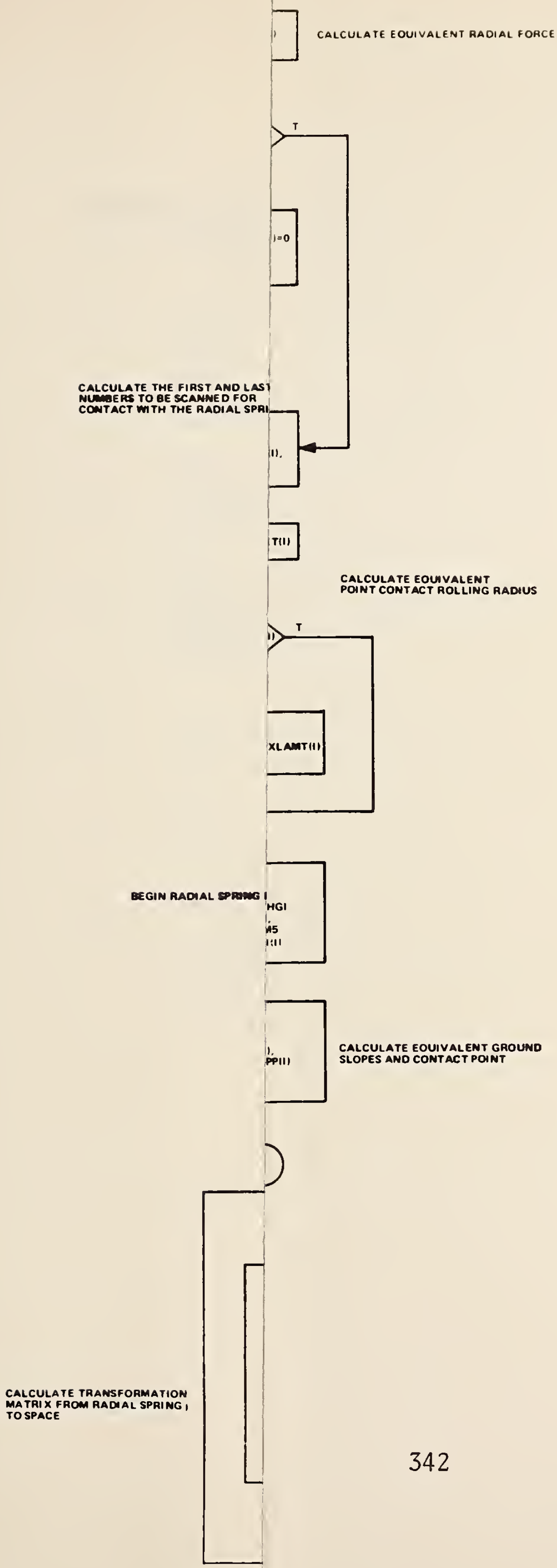
e. Common Variables Calculated

FR, HI, CAR, CBR, CGR, CPG,
PHGI, SFRX, SFRY, SFRZ, XGP

f. Size:

$DC4)_{16} = 3524)_{10}$ bytes

g. Computational Procedure:


CALCULATE EQUIVALENT RADIAL FORCE
T
)=0
CALCULATE THE FIRST AND LAST
NUMBERS TO BE SCANNED FOR
CONTACT WITH THE RADIAL SPRI
I),
T(I)
CALCULATE EQUIVALENT
POINT CONTACT ROLLING RADIUS
T
XLAMT(I)
BEGIN RADIAL SPRING
HGI
M5
PP(I)
CALCULATE EQUIVALENT GROUND
SLOPES AND CONTACT POINT
CALCULATE TRANSFORMATION
MATRIX FROM RADIAL SPRING
TO SPACE

37. SUBROUTINE RUFFRC(I,ZGM)

a. Purpose:

1. To determine an equivalent radial tire force and ground contact point from the distributed tire spring model when the road roughness option is being used

b. Common Blocks Required:

INPT1, DIMV, COMP, COMPN, TIRIN, RUFNES

c. Subroutines Required:

INTRPL

d. Arguments:

I = wheel number for which calculations are made
ZGM = single dimensioned array containing the road roughness data

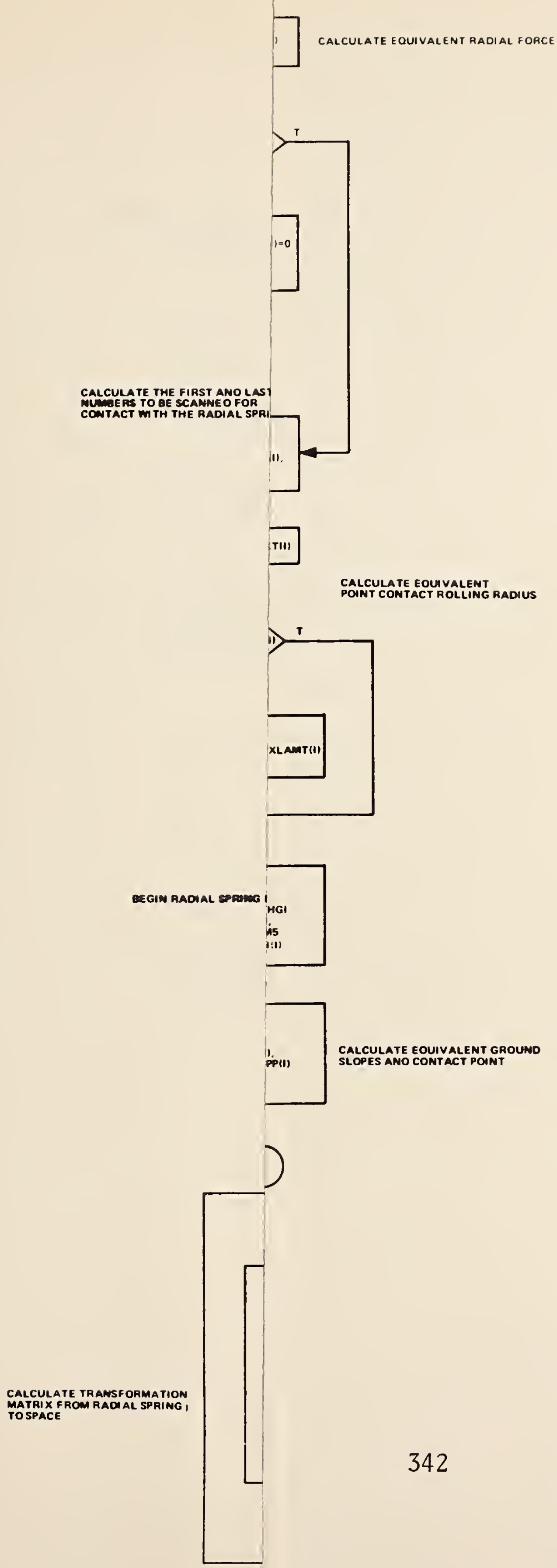
e. Common Variables Calculated:

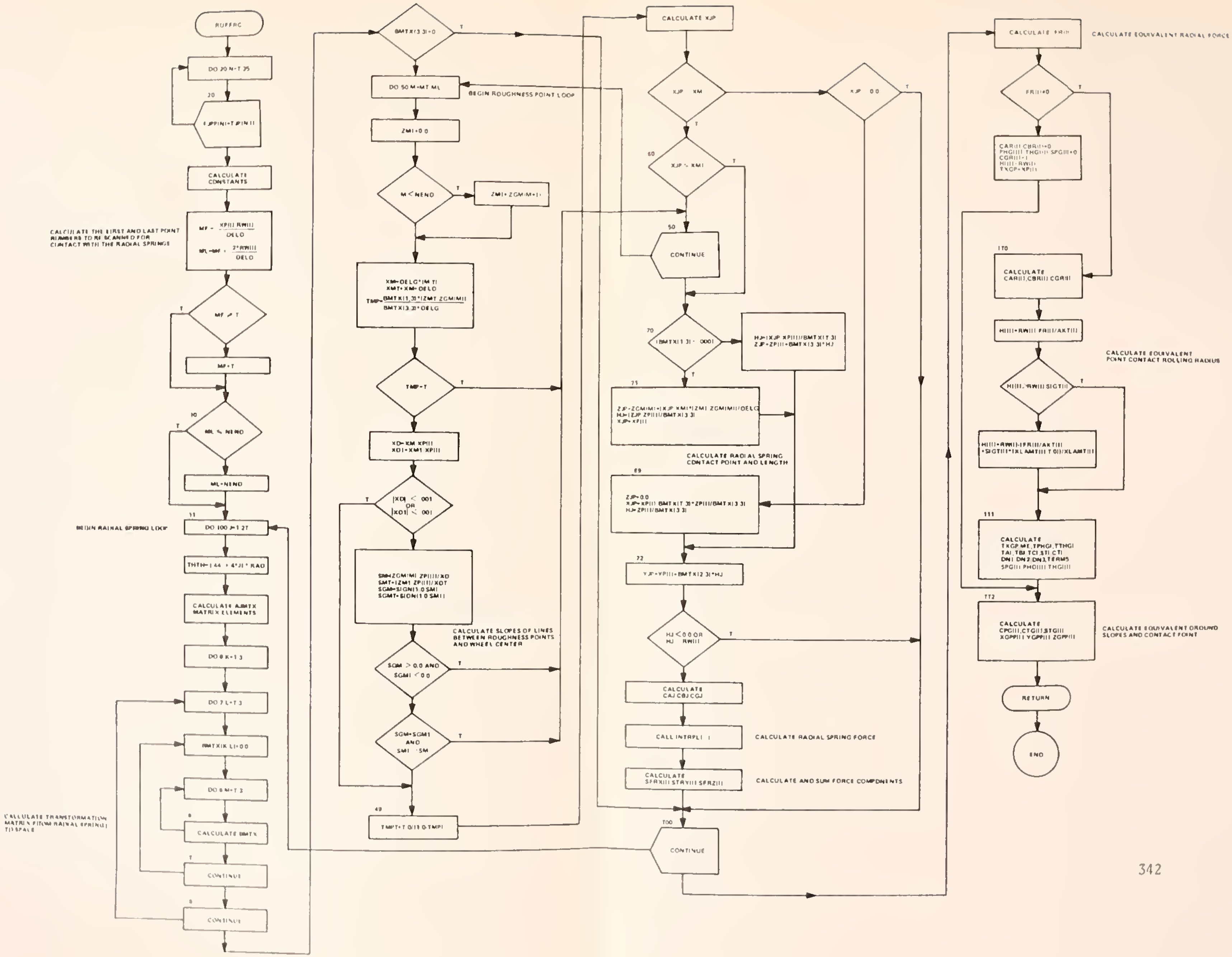
FR, HI, CAR, CBR, CGR, CPG, CTG, SPG, STG, BMTX, PHGI, SFRX, SFRY, SFRZ, XGPP, YGPP, ZGPP, AJMTX

f. Size:

$DC4)_{16} = 3524)_{10}$ bytes

g. Computational Procedure:


CALCULATE EQUIVALENT RADIAL FORCE
T
CALCULATE THE FIRST AND LAST
NUMBERS TO BE SCANNED FOR
CALCULATE EQUIVALENT
POINT CONTACT ROLLING RADIUS
T
XLAMT(I)
CALCULATE EQUIVALENT GROUND
SLOPES AND CONTACT POINT
CALCULATE TRANSFORMATION
TO SPACE



CALCULATE CONSTANTS
MF+T
ML+NEND
BEGIN RADIAL SPRING LOOP
CALCULATE AMTX MATRIX ELEMENTS
CONTINUE
BEGIN ROUGHNESS POINT LOOP
ZMI+0.0
M < NEND
CALCULATE SLOPES OF LINES BETWEEN ROUGHNESS POINTS AND WHEEL CENTER
SGM > 0.0 AND SGMI < 0.0
CALCULATE XJP
CONTINUE
CALCULATE RADIAL SPRING CONTACT POINT AND LENGTH
CALCULATE RADIAL SPRING FORCE
CALCULATE AND SUM FORCE COMPONENTS
CALCULATE EQUIVALENT RADIAL FORCE
CALCULATE EQUIVALENT POINT CONTACT ROLLING RADIUS
CALCULATE EQUIVALENT GROUND SLOPES AND CONTACT POINT
RETURN
END

38. SUBROUTINE RUFRED(NEND,DELG,DGMAX,ZRTAB)

a. Purpose:

1. Read road roughness data from FORTRAN device 4

b. Common Blocks Required:

None

c. Subroutine Required:

None

d. Arguments:

NEND = the number of road roughness points to be read from FORTRAN unit 4
DELG = the distance increment between points
DGMAX = (NEND-1) * DELG
ZRTAB = a single dimension array into which the road roughness data is read

e. Common Variables Calculated:

None

f. Size:

$2B8)_{16} = 696)_{10}$ bytes

g. Computational Procedure:

The road roughness data is rear via an unformatted READ statement into the ZRTAB array. The maximum number of points allowed is 2200.

39. SUBROUTINE SIMSOL

a. Purpose:

This subroutine solves a set of real simultaneous linear algebraic equations AX = B, with input, output and internal computation all in single precision

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

A - is a 2-dimensional (ND1xND2) matrix of coefficients
N - is the number of equations and unknowns
ND1 - is the first dimension of A in the calling program (ND1. GE. N and ND2. GE. N+1)

e. Common Variables Calculated:

None

f. Size:

$5E8)_{16} = 1512)_{10}$ bytes

g. Computational Procedure:

The routine will find the solution X of AX = B where A is a N by N matrix and B(I) is stored in A(I, N+1). The solution X(I) is returned in A(I, N+1).

Note: The Matrix A is destroyed by the subroutine.

Example:
```
REAL A(20,25)
CALL SIMSOL(A,10,20)
```

The solution is obtained by elimination using the largest pivotal divisor of each column. Each stage of elimination consists of interchanging rows when necessary to avoid division by zero or small numbers.

The forward solution to obtain variable N is done in N stages. The back solution for the other variables is calculated by successive substitutions. The final solution values are developed in column N+1 of matrix A, with variable 1 and A (1, N+1), variable 2 in A (2, N+1),..., and variable N in A (N, N+1).

38. SUBROUTINE RUFRED(NEND,DELG,DGMAX,ZRTAB)

a. Purpose:

1. Read road roughness data from FORTRAN device 4

b. Common Blocks Required:

None

c. Subroutine Required:

None

d. Arguments:

NEND = the number of road roughness points to be read from FORTRAN unit 4
DELG = the distance increment between points
DGMAX = (NEND-1) * DELG
ZRTAB = a single dimension array into which the road roughness data is read

e. Common Variables Calculated:

None

f. Size:

$2B8)_{16} = 696)_{10}$ bytes

g. Computational Procedure:

The road roughness data is rear via an unformatted READ statement into the ZRTAB array. The maximum number of points allowed is 2200.

39. SUBROUTINE SIMSOL

a. Purpose:

This subroutine solves a set of real simultaneous linear algebraic equations AX = B, with input, output and internal computation all in single precision

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

A - is a 2-dimensional (ND1xND2) matrix of coefficients
N - is the number of equations and unknowns
ND1 - is the first dimension of A in the calling program (ND1. GE. N and ND2. GE. N+1)

e. Common Variables Calculated:

None

f. Size:

$5E8)_{16} = 1512)_{10}$ bytes

g. Computational Procedure:

The routine will find the solution X of AX = B where A is a N by N matrix and B(I) is stored in A(I, N+1).
The solution X(I) is returned in A(I, N+1).

Note: The Matrix A is destroyed by the subroutine.

```
Example: REAL A(20,25)
         CALL SIMSOL(A,10,20)
```

The solution is obtained by elimination using the largest pivotal divisor of each column. Each stage of elimination consists of interchanging rows when necessary to avoid division by zero or small numbers.

The forward solution to obtain variable N is done in N stages. The back solution for the other variables is calculated by successive substitutions. The final solution values are developed in column N+1 of matrix A, with variable 1 and A (1, N+1), variable 2 in A (2, N+1),..., and variable N in A (N, N+1).

40. SUBROUTINE SUSFRC(DISP,VEL)

a. Purpose:

1. This subroutine calculates between the sprung and uns corners

b. Common Blocks Required:

INPT, INPT3, INTG, DIMV, COMP, INSUS, SUSCMP

c. Subroutines Required:

INTRPL

d. Arguments:

DISP - a four element array cc displacements
VEL - a four element array cc velocities

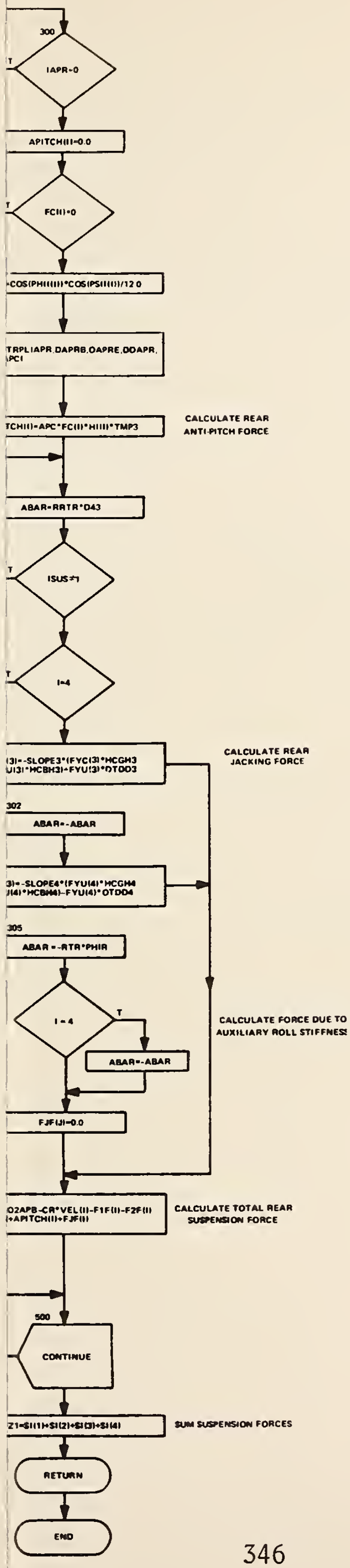
e. Common Variables Calculated:

SI, FJF, F1I, F2I, SFZ1, APITC

f. Size:

$7BC)_{16} = 1980)_{10}$ bytes

g. Computational Procedure:


300
IAPR=0
APITCH(I)=0.0
FC(I)=0
=COS(PHI(I(I))*COS(PSI(I(I))/12.0
TRPL(IAPR,DAPRB,OAPRE,DDAPR,
APC)
TCH(I)=APC*FC(I)*H(I)*TMP3
CALCULATE REAR
ANTI-PITCH FORCE
ABAR=RRTR*D43
ISUS≠1
I=4
CALCULATE REAR
JACKING FORCE
302
ABAR=-ABAR
305
ABAR=-RTR*PHIR
I=4
ABAR=-ABAR
CALCULATE FORCE DUE TO
AUXILIARY ROLL STIFFNESS
FJF(J)=0.0
CALCULATE TOTAL REAR
SUSPENSION FORCE
500
CONTINUE
SUM SUSPENSION FORCES
RETURN
END

40. SUBROUTINE SUSFRC(DISP,VEL)

a. Purpose:

1. This subroutine calculates the suspension forces acting between the sprung and unsprung masses at the four vehicle corners

b. Common Blocks Required:

INPT, INPT3, INTG, DIMV, COMP, ADTNL, APTABL, INSUS, SUSCMP

c. Subroutines Required:

INTRPL

d. Arguments:

DISP - a four element array containing the suspension displacements
VEL - a four element array containing the suspension velocities

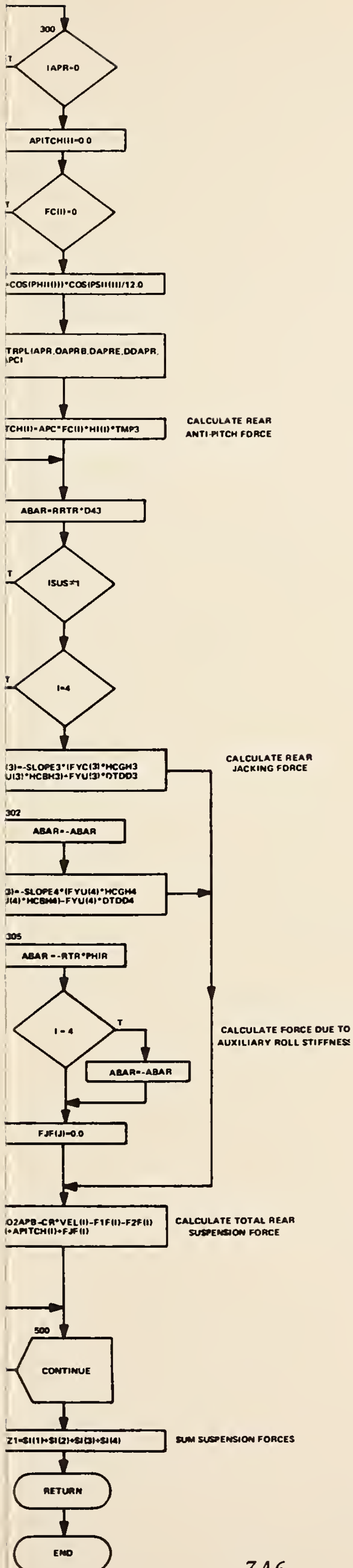
e. Common Variables Calculated:

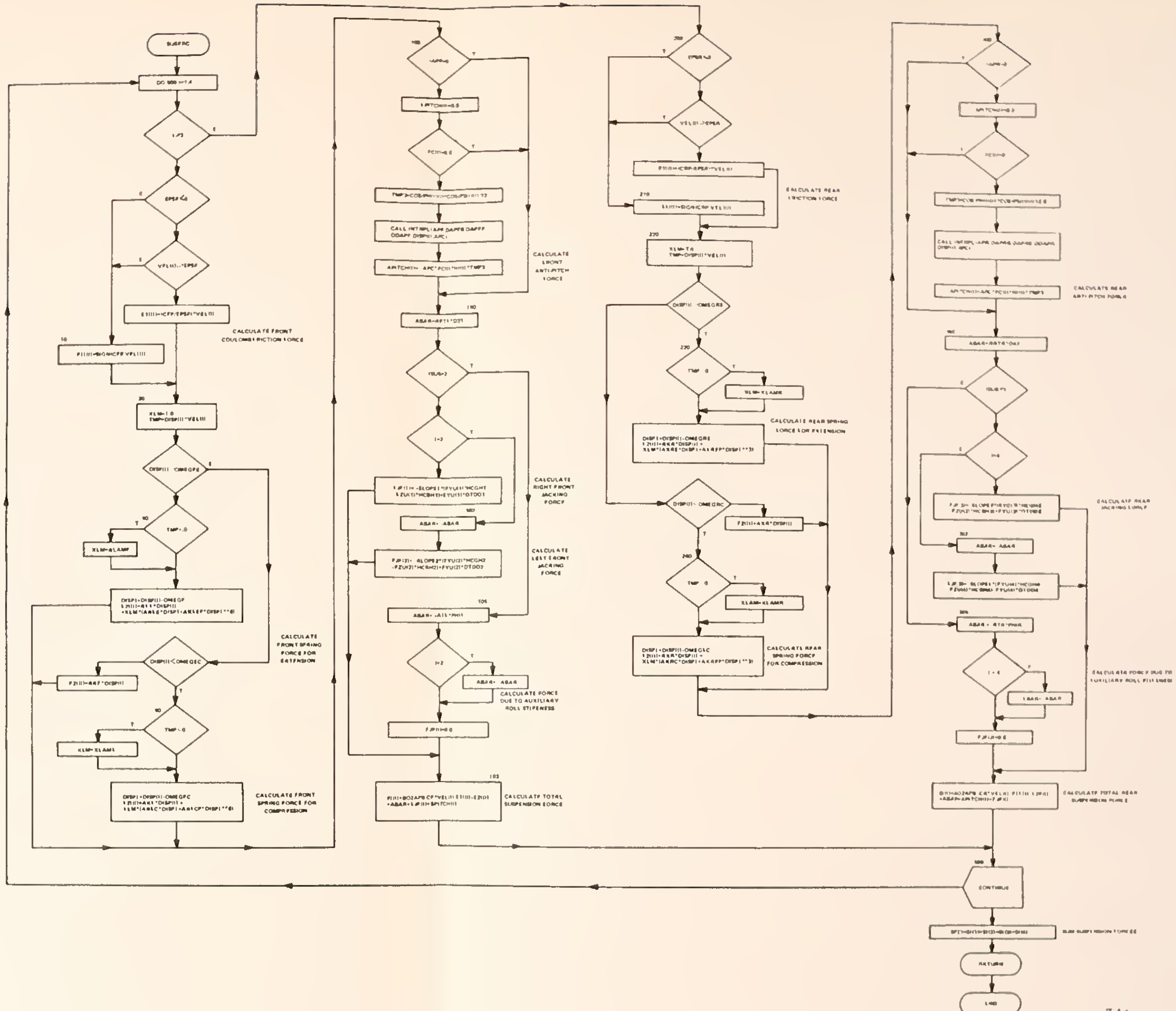
S1, FJF, F1I, F2I, SFZ1, APITCH

f. Size:

$7BC)_{16} = 1980)_{10}$ bytes

g. Computational Procedure:


300
IAPR=0
APITCH(I)=0.0
FC(I)=0
CALCULATE REAR
ANTI-PITCH FORCE
ABAR=RRTR*D43
ISUS≠1
I=4
CALCULATE REAR
JACKING FORCE
302
ABAR=-ABAR
305
ABAR=-RTR*PHIR
I=4
ABAR=-ABAR
CALCULATE FORCE DUE TO
AUXILIARY ROLL STIFFNES
FJF(J)=0.0
CALCULATE TOTAL REAR
SUSPENSION FORCE
500
CONTINUE
SUM SUSPENSION FORCES
RETURN
END


CALCULATE FRONT COULOMB FRICTION FORCE
CALCULATE FRONT SPRING FORCE FOR EXTENSION
CALCULATE FRONT SPRING FORCE FOR COMPRESSION
CALCULATE FRONT ANTI-PITCH FORCE
CALCULATE RIGHT FRONT JACKING FORCE
CALCULATE LEFT FRONT JACKING FORCE
CALCULATE FORCE DUE TO AUXILIARY ROLL STIFFNESS
CALCULATE TOTAL SUSPENSION FORCE
CALCULATE REAR FRICTION FORCE
CALCULATE REAR SPRING FORCE FOR EXTENSION
CALCULATE REAR SPRING FORCE FOR COMPRESSION
CALCULATE REAR ANTI-PITCH FORCE
CALCULATE REAR JACKING FORCE
CALCULATE TOTAL REAR SUSPENSION FORCE
CONTINUE
RETURN
END

41. SUBROUTINE TEREAD

a. Purpose:

1. This subroutine reads terrain table input cards

b. Common Blocks Required:

INPT

c. Subroutines Required:

None

d. Arguments:

I - Terrain table number
NNBX - Number of X' boundaries
NNBY - Number of Y' boundaries
NNX - Number of X' terrain entries
NNY - Number of Y' terrain entries
NZ5T - Indicator for variable increment table
NERR - Error indicator

e. Common Variables Calculated:

ZGP, XBDRY, YBDRY, PSBDR0, XXZGP5, YYZGP5

f. Size:

$626)_{16} = 1574)_{10}$ bytes

42. SUBROUTINE TIRFR

a. Purpose:

1. Control integration step-size for wheel spin degrees-of-freedom
2. Integrate wheel spin degrees-of-freedom
3. Calculate brake temperatures
4. Calculate time averages of tire forces over DT

b. Common Blocks Required:

INPT, INTG, DIMV, COMP, COMPN, INPT4, COMP4, INTR, INPT5, COMP5

c. Subroutines Required:

DAUXR

d. Arguments:

None

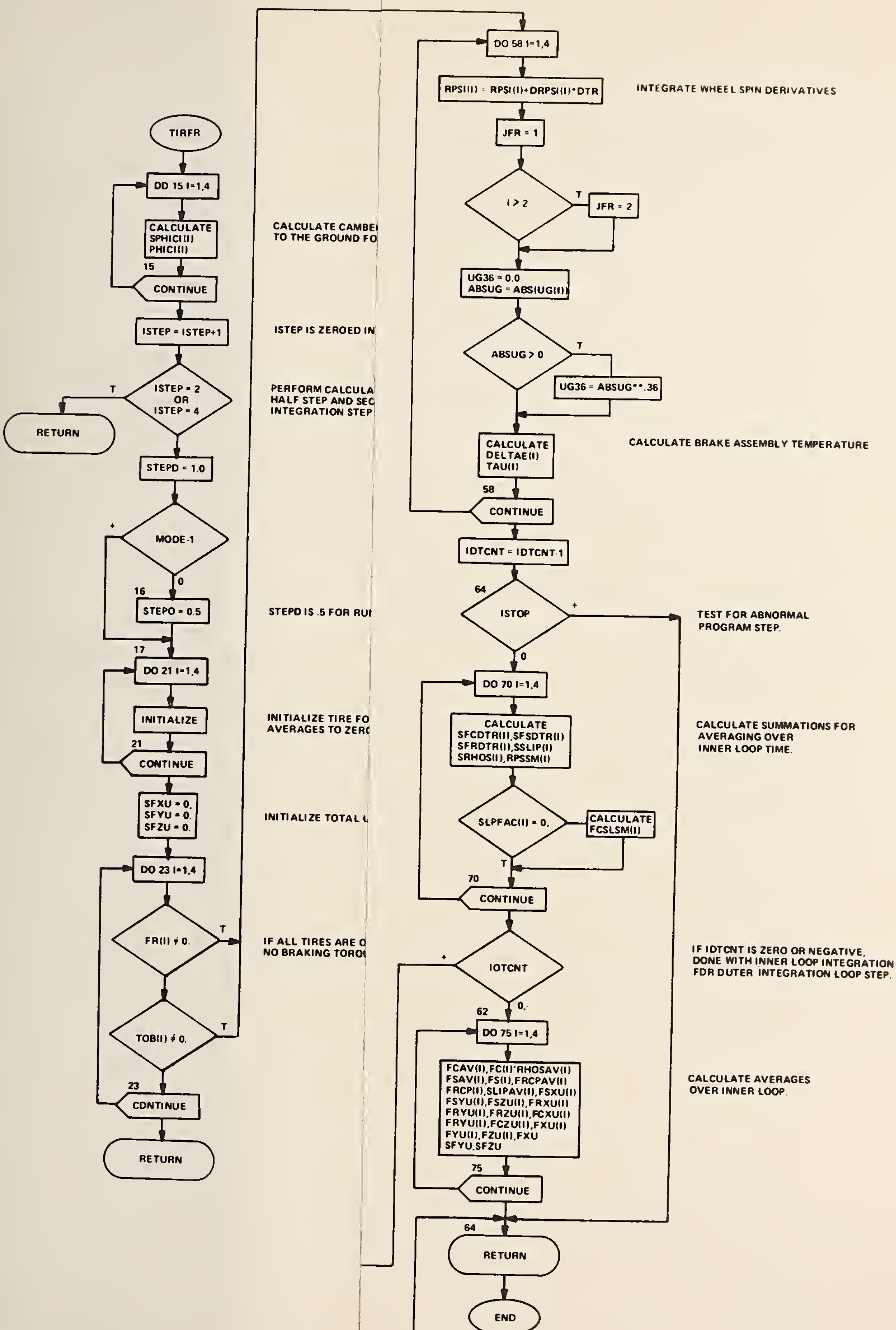
e. Common Variables Calculated:

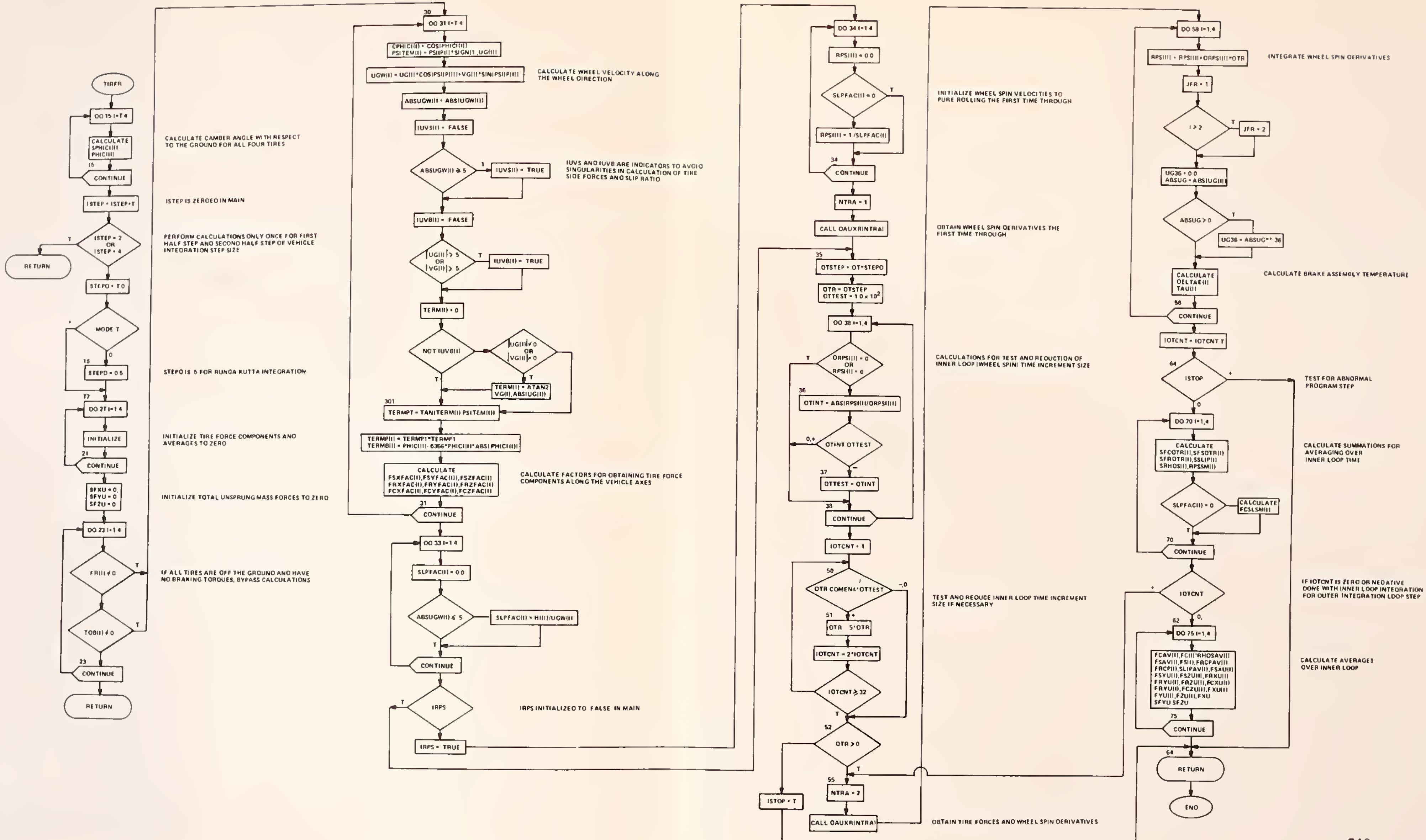
FC, FS, DTR, FXU, FYU, FZU, TAU, UGW, FCAV, FCXU, FCYU, FCZU, FRCP, FRXU, FRYU, FRZU, FSAV, FSXU, FSYU, FSZU, IRPS, IUVB, IUVS, RPSI, SFXU, SFYU, SFZU, TERM, DTINT, ISTEP, ISTOP, RPSSM, SRHOS, SSLIP, STEPD, TERMB, TERMP, ABSUGW, CPHICI, DELTAE, DISTEP, DTTEST, FCSLSM, FCXFAC, FCYFAC, FCZFAC, FRCPAV, FRXFAC, FRYFAC, FRZFAC, FSXFAC, FSYFAC, FSZFAC, IDTCNT, PSITEM, RHOSAV, SFCDTR, SFRCPR, SFSDTR, SLIPAV, SLPFAC, SPHICI

f. Size:

$B16)_{16} = 2838)_{10}$ bytes

g. Computational Procedure:


TIRFR
DO 15 I=1,4
CALCULATE SPHICI(I) PHICI(I)
15
CONTINUE
ISTEP = ISTEP+1
ISTEP = 2 OR ISTEP = 4
T
RETURN
STEPD = 1.0
MODE-1
+
0
16
STEPO = 0.5
17
DO 21 I=1,4
INITIALIZE
21
CONTINUE
SFXU = 0.
SFYU = 0.
SFZU = 0.
DO 23 I=1,4
FR(I) ≠ 0.
T
TOB(I) ≠ 0.
T
23
CDNTINUE
RETURN
CALCULATE CAMBE
TO THE GROUND FO
ISTEP IS ZEROED IN
PERFORM CALCULA
HALF STEP AND SEC
INTEGRATION STEP
STEPD IS .5 FOR RU
INITIALIZE TIRE FO
AVERAGES TO ZER
INITIALIZE TOTAL U
IF ALL TIRES ARE O
NO BRAKING TORQ
DO 58 I=1,4
RPSI(I) = RPSI(I)+DRPSI(I)*DTR
INTEGRATE WHEEL SPIN DERIVATIVES
JFR = 1
I > 2
T
JFR = 2
UG36 = 0.0
ABSUG = ABS(UG(I))
ABSUG > 0
T
UG36 = ABSUG**.36
CALCULATE DELTAE(I) TAU(I)
CALCULATE BRAKE ASSEMBLY TEMPERATURE
58
CONTINUE
IDTCNT = IDTCNT-1
64
ISTOP
+
TEST FOR ABNORMAL PROGRAM STEP.
0
DO 70 I=1,4
CALCULATE SFCDTR(I),SFSDTR(I) SFRDTR(I),SSLIP(I) SRHOS(I),RPSSM(I)
CALCULATE SUMMATIONS FOR AVERAGING OVER INNER LOOP TIME.
SLPFAC(I) = 0.
CALCULATE FCSLSM(I)
T
70
CONTINUE
IOTCNT
+
IF IDTCNT IS ZERO OR NEGATIVE, DONE WITH INNER LOOP INTEGRATION FDR DUTER INTEGRATION LOOP STEP.
0,-
62
DO 75 I=1,4
FCAV(I),FC(I)*RHOSAV(I) FSAV(I),FS(I),FRCPAV(I) FRCP(I),SLIPAV(I),FSXU(I) FSYU(I),FSZU(I),FRXU(I) FRYU(I),FRZU(I),FCXU(I) FRYU(I),FCZU(I),FXU(I) FYU(I),FZU(I),FXU SFYU,SFZU
CALCULATE AVERAGES OVER INNER LOOP.
75
CONTINUE
64
RETURN
END



TIRER
CALCULATE CAMBER ANGLE WITH RESPECT TO THE GROUND FOR ALL FOUR TIRES
ISTEP IS ZEROED IN MAIN
PERFORM CALCULATIONS ONLY ONCE FOR FIRST HALF STEP AND SECOND HALF STEP OF VEHICLE INTEGRATION STEP SIZE
STEPO IS .5 FOR RUNGA KUTTA INTEGRATION
INITIALIZE TIRE FORCE COMPONENTS AND AVERAGES TO ZERO
INITIALIZE TOTAL UNSPRUNG MASS FORCES TO ZERO
IF ALL TIRES ARE OFF THE GROUND AND HAVE NO BRAKING TORQUES, BYPASS CALCULATIONS
CALCULATE WHEEL VELOCITY ALONG THE WHEEL DIRECTION
IUVS AND IUVB ARE INDICATORS TO AVOID SINGULARITIES IN CALCULATION OF TIRE SIDE FORCES AND SLIP RATIO
CALCULATE FACTORS FOR OBTAINING TIRE FORCE COMPONENTS ALONG THE VEHICLE AXES
IRPS INITIALIZED TO FALSE IN MAIN
INITIALIZE WHEEL SPIN VELOCITIES TO PURE ROLLING THE FIRST TIME THROUGH
OBTAIN WHEEL SPIN DERIVATIVES THE FIRST TIME THROUGH
CALCULATIONS FOR TEST AND REDUCTION OF INNER LOOP (WHEEL SPIN) TIME INCREMENT SIZE
TEST AND REDUCE INNER LOOP TIME INCREMENT SIZE IF NECESSARY
OBTAIN TIRE FORCES AND WHEEL SPIN DERIVATIVES
INTEGRATE WHEEL SPIN DERIVATIVES
CALCULATE BRAKE ASSEMBLY TEMPERATURE
TEST FOR ABNORMAL PROGRAM STEP
CALCULATE SUMMATIONS FOR AVERAGING OVER INNER LOOP TIME
IF IOTCNT IS ZERO OR NEGATIVE DONE WITH INNER LOOP INTEGRATION FOR OUTER INTEGRATION LOOP STEP
CALCULATE AVERAGES OVER INNER LOOP
RETURN
END

ndent variables that are required in

coordinate system if necessary

:

EX, COMP4, INPT5, INSUS,

lated:

, UQ, UR, VP, VR, WP, WQ, D21,
TH, GSTH, PHFP, PHIT, PHRP,
, TIZ2, WFMF, ZFD1, ZFD2, ZRD3,
SPH, COSPS, COSTH, CPHTP, CPSTP,
D3P4, D1MD2, D1PD2, D3MD4,
HIR2, RPF2M, RPHFD, RPHRD, SECTP,
PHTP, SPSTP, TANTP, THETT, XINDL,
RD34, COSPHN, COSPSN, COSTHN,
, SINPHN, SINPSN, SINTHN, STHETP,

s

e:

dent variables
te system indexing; if required as

43. SUBROUTINE TMCNST

a. Purpose:

1. Evaluate time dependent variables that are required in other subroutines
2. Test for and index coordinate system if necessary

b. Common Blocks Required:

INPT, INTG, COMP, EINDEX, COMP4, INPT5, INSUS, SUSCMP, NEWCRB

c. Subroutines Required:

None

d. Arguments:

None

e. Common Variables Calculated:

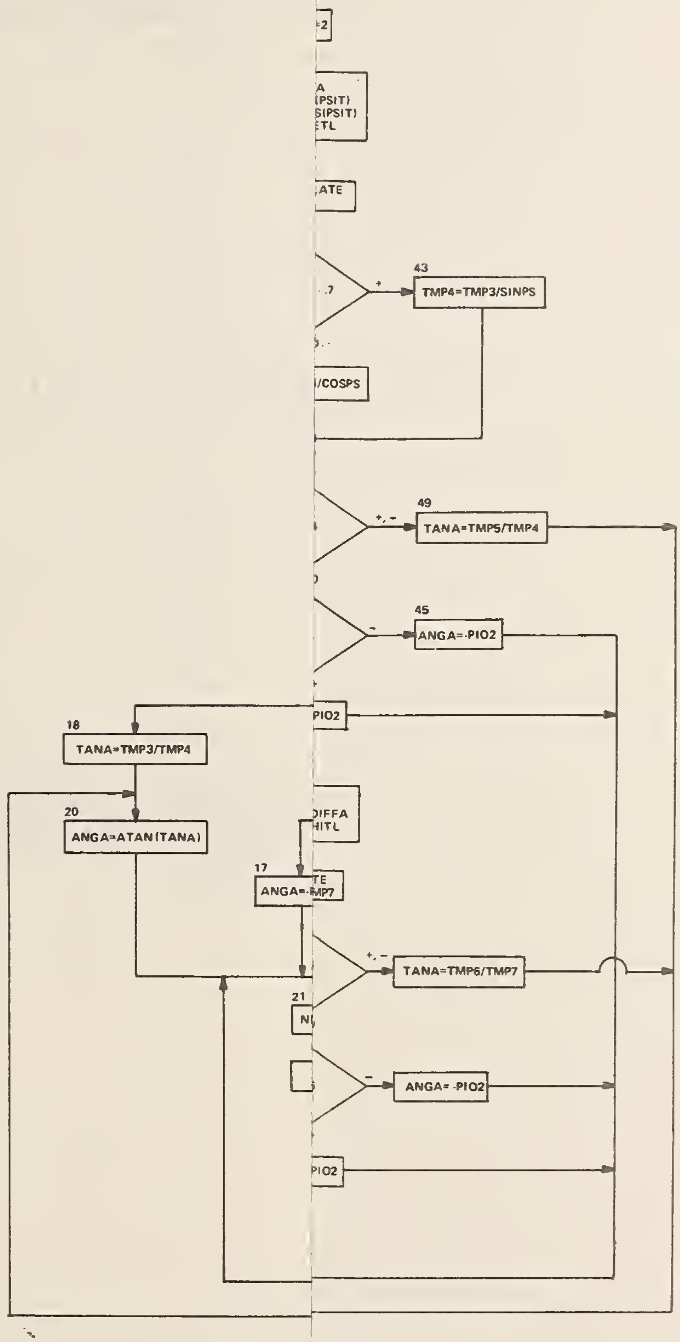
PQ, PR, P2, QR, Q2, R2, UQ, UR, VP, VR, WP, WQ, D21, D43, RPR, TPF, TPR, GCTH, GSTH, PHFP, PHIT, PHRP, PSIT, RFPF, SFXS, TG61, TIZ2, WFMF, ZFD1, ZFD2, ZRD3, ZRD4, BNMTX, CNMTX, COSPH, COSPS, COSTH, CPHTP, CPSTP, DD1M2, DD1P2, DD3M4, DD3P4, D1MD2, D1PD2, D3MD4, D3PD4, ISTOP, PHIF2, PHIR2, RPF2M, RPHFD, RPHRD, SECTP, SINPH, SINPS, SINTH, SPHTP, SPSTP, TANTP, THETT, XINDL, ZFD12, ZFD3R, ZRD3R, ZRD34, COSPHN, COSPSN, COSTHN, CTHETP, PHIFD2, PHIRD2, SINPHN, SINPSN, SINTHN, STHETP, ZFD1RF

f. Size:

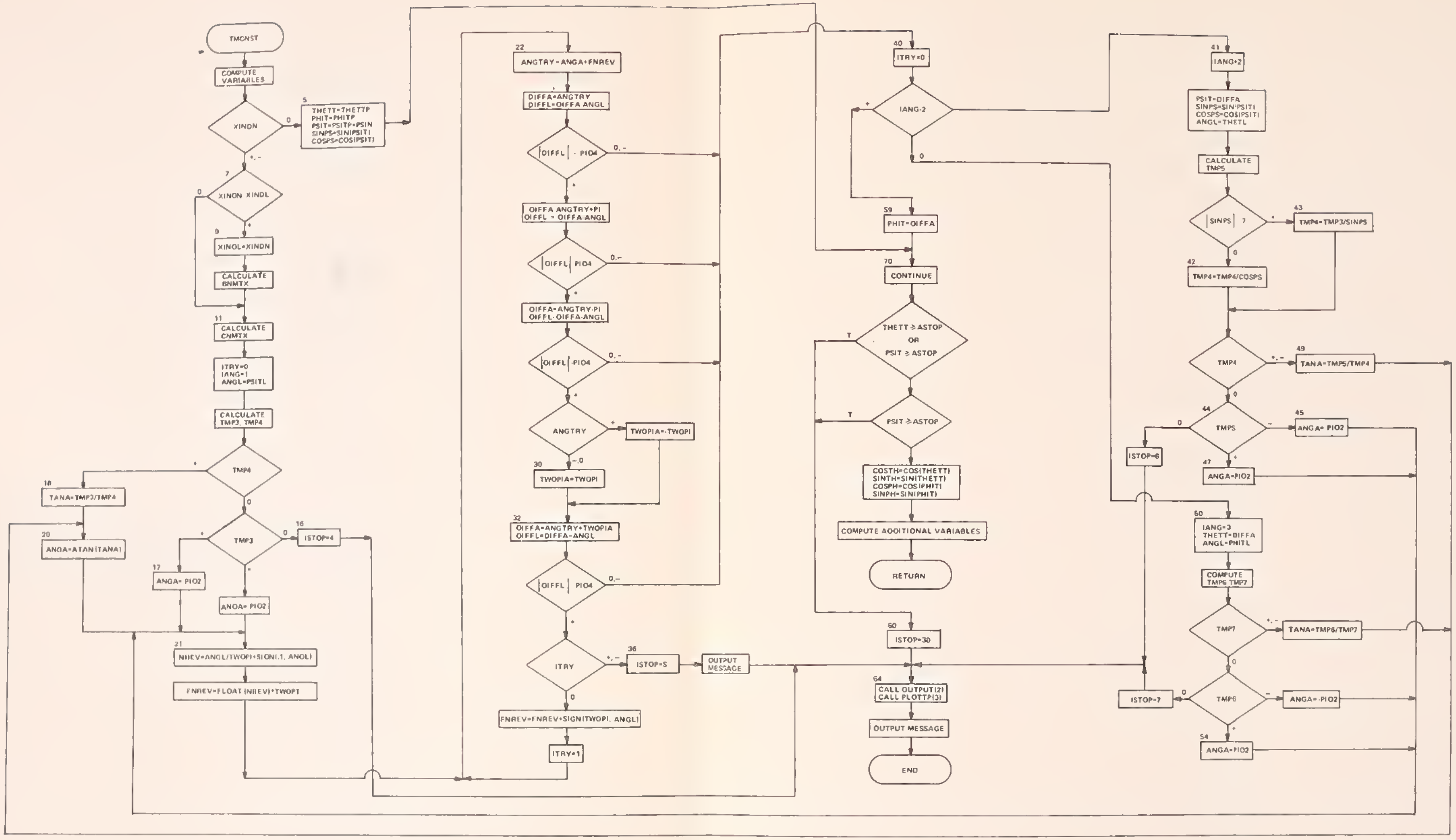
$D96)_{16} = 3478)_{10}$ bytes

g. Computational Procedure:

1. Compute time dependent variables
2. Test for coordinate system indexing; if required as shown below



43
TMP4=TMP3/SINPS
/COSPS
49
TANA=TMP5/TMP4
45
ANGA=-PIO2
PIO2
18
TANA=TMP3/TMP4
20
ANGA=ATAN(TANA)
17
ANGA=-TMP7
TANA=TMP6/TMP7
21
ANGA= -PIO2
PIO2


TMCNST
COMPUTE VARIABLES
XINON
THETT=THETTP
PHIT=PHITP
PSIT=PSITP+PSIN
SINPS=SIN(PSIT)
COSPS=COS(PSIT)
XINON XINOL
XINOL=XINON
CALCULATE BNMTX
CALCULATE CNMTX
ITRY=0
IANG=1
ANGL=PSITL
CALCULATE TMP3, TMP4
TMP4
TANA=TMP3/TMP4
ANGA=ATAN(TANA)
TMP3
ISTOP=4
ANGA= PIO2
NREV=ANGL/TWOPI+SIGN(.1, ANGL)
FNREV=FLOAT(NREV)*TWOPI
ANGTRY=ANGA+FNREV
ANGTRY
TWOPIA=-TWOPI
TWOPIA=TWOPI
ITRY
ISTOP=5
OUTPUT MESSAGE
ITRY=1
ITRY=0
IANG-2
PHIT=DIFFA
CONTINUE
THETT ≥ ASTOP
OR
PSIT ≥ ASTOP
PSIT ≥ ASTOP
COSTH=COS(THETT)
SINTH=SIN(THETT)
COSPH=COS(PHIT)
SINPH=SIN(PHIT)
COMPUTE ADDITIONAL VARIABLES
RETURN
ISTOP=30
CALL OUTPUT(2)
CALL PLOTTP(3)
OUTPUT MESSAGE
END
IANG=2
PSIT=DIFFA
SINPS=SIN(PSIT)
COSPS=COS(PSIT)
ANGL=THETL
CALCULATE TMP5
TMP4=TMP3/SINPS
TMP4=TMP4/COSPS
TMP4
TANA=TMP5/TMP4
TMP5
ISTOP=6
ANGA=-PIO2
ANGA=PIO2
IANG=3
THETT=DIFFA
ANGL=PHITL
COMPUTE TMP6 TMP7
TMP7
TANA=TMP6/TMP7
TMP6
ISTOP=7
ANGA=-PIO2
ANGA=PIO2

44. SUBROUTINE TREAD

a. Purpose:

1. This subroutine reads a one-dimensional card input table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

NCARD - Input card number
NCRDS - Number of cards to be read
NT - Number of elements to be read into the table
NDIM - Maximum table dimension
ARRAY - Table array
NERR - Error indicator

e. Common Variables Calculated:

None

f. Size:

$258)_{16} = 600)_{10}$ bytes

g. Computational Procedure:

1. Read table input cards checking to insure that the table sequence number increases with each card.
2. Load the variables into the table array.

45. SUBROUTINE T2READ

a. Purpose:

1. This subroutine reads a two-dimensional input table

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

NCARD - Input card number
ND1 - Row dimension of the input table
NI - Number of rows to be read
NJ - Number of columns to be read
ARRAY - Table array
NERR - Error indicator

e. Common Variables Calculated:

None

f. Size:

$2C4)_{16} = 708)_{10}$ bytes

g. Computational Procedure:

The input table is read rowwise with the second subscript varying most rapidly.

46. SUBROUTINE UMOMNT(IS)

a. Purpose:

1. This subroutine calculates the moments acting on the sprung and unsprung masses

b. Common Blocks Required:

INPT, DIMV, COMP, INSUS, SUSCMP

c. Subroutines Required:

None

d. Arguments:

IS - suspension option indicator

e. Common Variables Calculated:

SNPF, SNPR, SNPU, SNTU, SNPSU, TERM1, TERM2, TERM3

f. Size:

$822)_{16} = 2082)_{10}$ bytes

g. Computational Procedure:

1. For IS=0 (independent front, solid axle rear suspension) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU) and the rear axle roll moment (SNPR).
2. For IS=1 (independent front and rear suspension) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU).
3. For IS=2 (solid front and rear axles) calculate the sprung mass roll, pitch and yaw moments (SNPU, SNTU, SNPSU) and the front and rear axle roll moments (SNPF, SNPR).

47. SUBROUTINE VGORNT

a. Purpose:

1. Determine the orientation of the vehicle wheels with respect to the ground.

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, TIRIN, INSUS, SUSCMP, NEWCRB, RUFNES

c. Subroutines Required:

INTRP5, GCP, CRBIMP, RUFFRC

d. Arguments:

None

e. Variables Calculated:

AS, AX, AY, BS, BX, BY, CS, CX, CY, D1, D2, D3, P1, P3, P4, P5, P6, P7, UG, VG, V1, V2, V3, V4, W1, W2, W3, W4, CAC, CAH, CAS, CBC, CBH, CBS, CGC, CGH, CGS, CTG, STG, CAGZ, CAYW, CAZW, CBGZ, CBYW, CBZW, CGGZ, CGYW, CGZW, CPYG, CTXG, HCAH, HCBH, HCGH, LCB1, LCB2, PHGI, SPYG, STXG, THGI, TMP3, TMP4, ZPGI, DISTX, DISTY, PSIIP, XMUGI

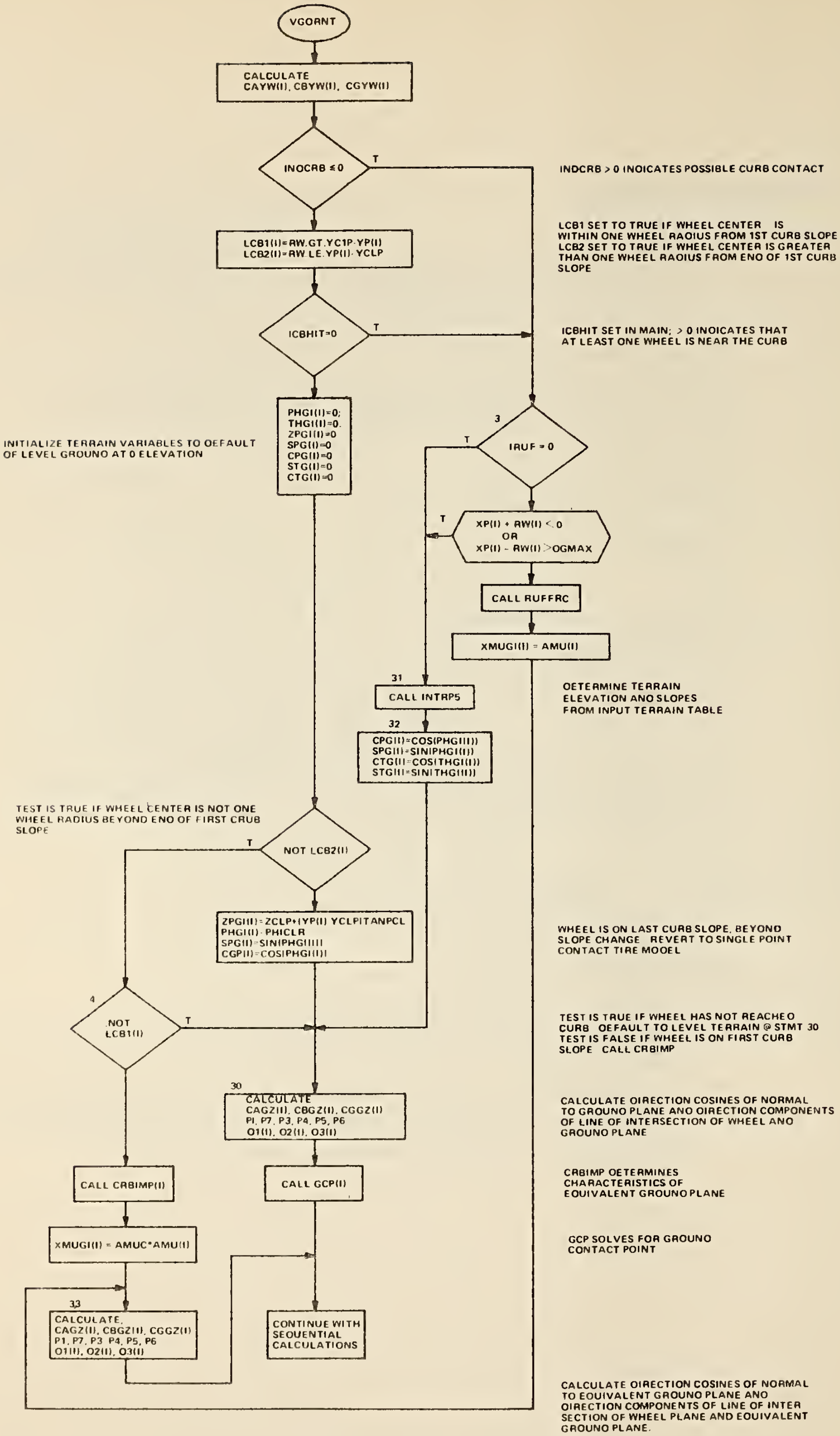
f. Size:

$1002)_{16} = 4098)_{10}$ bytes

g. Computational Procedure:

For wheels I = 1 to 4

1. Calculate the direction cosines of the normal to the wheel plane.
2. Determine the direction cosines of a normal to the ground plane and direction components of the intersection of the wheel plane and ground plane as follows:


VGORNT
CALCULATE CAYW(I), CBYW(I), CGYW(I)
INDCRB ≤ 0
T
INDCRB > 0 INDICATES POSSIBLE CURB CONTACT
LCB1(I)=RW.GT.YC1P-YP(I)
LCB2(I)=RW.LE.YP(I)-YCLP
LCB1 SET TO TRUE IF WHEEL CENTER IS WITHIN ONE WHEEL RADIUS FROM 1ST CURB SLOPE
LCB2 SET TO TRUE IF WHEEL CENTER IS GREATER THAN ONE WHEEL RADIUS FROM END OF 1ST CURB SLOPE
ICBHIT=0
T
ICBHIT SET IN MAIN; > 0 INDICATES THAT AT LEAST ONE WHEEL IS NEAR THE CURB
INITIALIZE TERRAIN VARIABLES TO DEFAULT OF LEVEL GROUND AT 0 ELEVATION
PHGI(I)=0; THGI(I)=0. ZPGI(I)=0 SPG(I)=0 CPG(I)=0 STG(I)=0 CTG(I)=0
3
IRUF = 0
T
XP(I) + RW(I) < 0 OR XP(I) - RW(I) > DGMAX
T
CALL RUFFRC
XMUGI(I) = AMU(I)
31
CALL INTRP5
DETERMINE TERRAIN ELEVATION AND SLOPES FROM INPUT TERRAIN TABLE
32
CPG(I)=COS(PHGI(I)) SPG(I)=SIN(PHGI(I)) CTG(I)=COS(THGI(I)) STG(I)=SIN(THGI(I))
TEST IS TRUE IF WHEEL CENTER IS NOT ONE WHEEL RADIUS BEYOND END OF FIRST CRUB SLOPE
T
NOT LCB2(I)
ZPGI(I)=ZCLP+(YP(I)-YCLP)TANPCL PHGI(I)=PHICLR SPG(I)=SIN(PHGI(I)) CGP(I)=COS(PHGI(I))
WHEEL IS ON LAST CURB SLOPE, BEYOND SLOPE CHANGE REVERT TO SINGLE POINT CONTACT TIRE MODEL
4
.NOT LCB1(I)
T
TEST IS TRUE IF WHEEL HAS NOT REACHED CURB DEFAULT TO LEVEL TERRAIN @ STMT 30 TEST IS FALSE IF WHEEL IS ON FIRST CURB SLOPE CALL CRBIMP
30
CALCULATE CAGZ(I), CBGZ(I), CGGZ(I) P1, P7, P3, P4, P5, P6 O1(I), O2(I), O3(I)
CALCULATE DIRECTION COSINES OF NORMAL TO GROUND PLANE AND DIRECTION COMPONENTS OF LINE OF INTERSECTION OF WHEEL AND GROUND PLANE
CALL CRBIMP(I)
CALL GCP(I)
CRBIMP DETERMINES CHARACTERISTICS OF EQUIVALENT GROUND PLANE
XMUGI(I) = AMUC*AMU(I)
GCP SOLVES FOR GROUND CONTACT POINT
33
CALCULATE, CAGZ(I), CBGZ(I), CGGZ(I) P1, P7, P3 P4, P5, P6 O1(I), O2(I), O3(I)
CONTINUE WITH SEQUENTIAL CALCULATIONS
CALCULATE DIRECTION COSINES OF NORMAL TO EQUIVALENT GROUND PLANE AND DIRECTION COMPONENTS OF LINE OF INTER SECTION OF WHEEL PLANE AND EQUIVALENT GROUND PLANE.

3. Calculate the direction cosines of the line of action of the radial tire force with respect to the vehicle axes, CAH(I), CBH(I), CGH(I).
4. Calculate the lateral and vertical velocities of the tire at the ground contact point with components resolved in the vehicle axes, (V1,W1); (V2,W2); (V3,W3); (V4,W4).
5. Calculate the direction components of the vehicle x axis projected into the ground plane, AX(I), BX(I), CX(I).
6. Calculate the sine and cosine of the angle between the vehicle x axis and its projection into the ground plane STXG(I), CTXG(I).
7. Calculate the longitudinal velocity of the tire contact point parallel to the ground plane UG(I).
8. Calculate the direction components of the vehicle y axis projected into the ground plane, AY(I), BY(I), CY(I).
9. Calculate the sine and cosine of the angle between the vehicle y axis and its projection into the ground plane SPYG(I), CPYG(I).
10. Calculate the lateral velocity of the tire contact point parallel to the ground plane, VG(I).
11. Calculate the direction cosines of the steering axis of the wheel.
12. Calculate the steer angle in the ground plane, PSIIP(I).
13. Calculate the direction cosines of the line of action of the circumferential tire force (CAC(I), CBC(I), CGC(I)) and of the tire side force (CAS(I), CBS(I), CGS(I)).

48. SUBROUTINE VPOS

a. Purpose:

1. Compute positions, orientations and velocities of the vehicle wheels
2. Calculate directions of the x and y axis in space

b. Common Blocks Required:

INPT, INPT1, INTG, DIMV, COMP, COMPN, ADTNL, INSUS, SUSCMP

c. Subroutines Required:

INTRPL, DRIVER

d. Arguments:

None

e. Variables Calculated:

U1, U2, U3, U4, CAX, CAY, CBX, CBY, CGX, CGY, X1P, X2P, X3P, X4P, Y1P, Y2P, Y3P, Y4P, Z1P, Z2P, Z3P, Z4P, AMTX, PHI1, PHI2, PHI3, PHI4, PSI1, PSI2, PSI3, PSI4, DTDD1, DTDD2, DTDD3, DTDD4, DTHF1, DTHF2, DTHF3, DTHF4, PHI1D, PHI2D, PHI3D, PHI4D, PSIFI, SFYUF, SFYUR, PHIFID, SLOPE1, SLOPE2, SLOPE3, SLOPE4

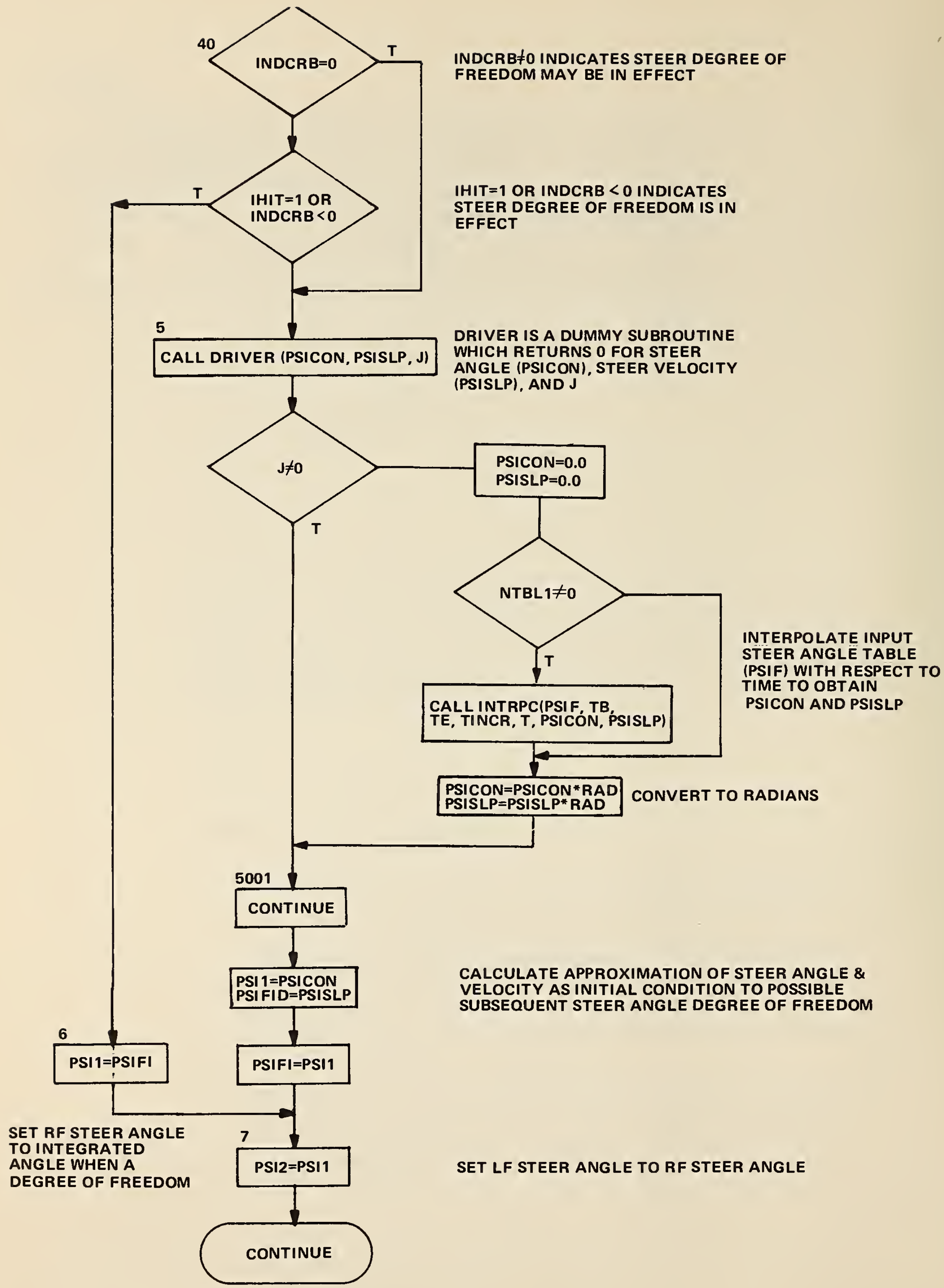
f. Size:

$AA4)_{16} = 2724)_{10}$ bytes

g. Computational Procedure:

1. Calculate longitudinal velocities of wheel centers along the vehicle axes, U1, U2, U3, U4. Note that for independent suspension options, INTRPC is called to obtain the track change and rate of track change as a function of suspension position.
2. Zero forces acting on the unsprung masses SFYU = SFXU = SFYUF = SFYUR = SFZU = 0.
3. Calculate AMTX, the transformation matrix from vehicle to space coordinate systems.
4. Calculate direction cosines of the vehicle x and y axis in space (CAX, CBX, CGX and CAY, CBY, CGY).
5. Calculate positions of the wheel centers in space (X1P, Y1P, Z1P); (X2P, Y2P, Z2P); (X3P, Y3P, Z3P); (X4P, Y4P, Z4P).

6. Call INTRPC (entry point in INTRPL) to obtain wheel camber angles and rates of change of camber angles with deflection by interpolation of the input camber tables with respect to suspension deflection for independent suspension options. Note that since the input table of camber is in units of degrees, a conversion in radians is also made.
7. Determine the front wheel steer angle with the following logic.


40
INDCRB=0
T
INDCRB≠0 INDICATES STEER DEGREE OF FREEDOM MAY BE IN EFFECT
T
IHIT=1 OR INDCRB<0
IHIT=1 OR INDCRB < 0 INDICATES STEER DEGREE OF FREEDOM IS IN EFFECT
5
CALL DRIVER (PSICON, PSISLP, J)
DRIVER IS A DUMMY SUBROUTINE WHICH RETURNS 0 FOR STEER ANGLE (PSICON), STEER VELOCITY (PSISLP), AND J
J≠0
T
PSICON=0.0
PSISLP=0.0
NTBL1≠0
T
CALL INTRPC(PSIF, TB, TE, TINCR, T, PSICON, PSISLP)
INTERPOLATE INPUT STEER ANGLE TABLE (PSIF) WITH RESPECT TO TIME TO OBTAIN PSICON AND PSISLP
PSICON=PSICON*RAD
PSISLP=PSISLP*RAD
CONVERT TO RADIANS
5001
CONTINUE
PSI1=PSICON
PSIFID=PSISLP
CALCULATE APPROXIMATION OF STEER ANGLE & VELOCITY AS INITIAL CONDITION TO POSSIBLE SUBSEQUENT STEER ANGLE DEGREE OF FREEDOM
6
PSI1=PSIFI
PSIFI=PSI1
SET RF STEER ANGLE TO INTEGRATED ANGLE WHEN A DEGREE OF FREEDOM
7
PSI2=PSI1
SET LF STEER ANGLE TO RF STEER ANGLE
CONTINUE

49. SUBROUTINE WHEEL

a. Purpose:

1. To calculate equivalent tire radial mode spring rates

b. Common Blocks Required:

None

c. Subroutines Required:

None

d. Arguments:

AKT - Point contact model tire spring rate
SIGT - Point contact model tire deflection at which spring rate increases
XLAMT - Spring rate increase
RWHJB - Beginning deflection for radial spring table
RWHJE - Ending deflection for radial spring table
DRWHJ - Deflection increment for radial spring table
NFJP - Number of radial spring table entries
RW - Undeflected tire radius
FJP - Radial spring force table
NO - Maximum number of entries in radial spring force table

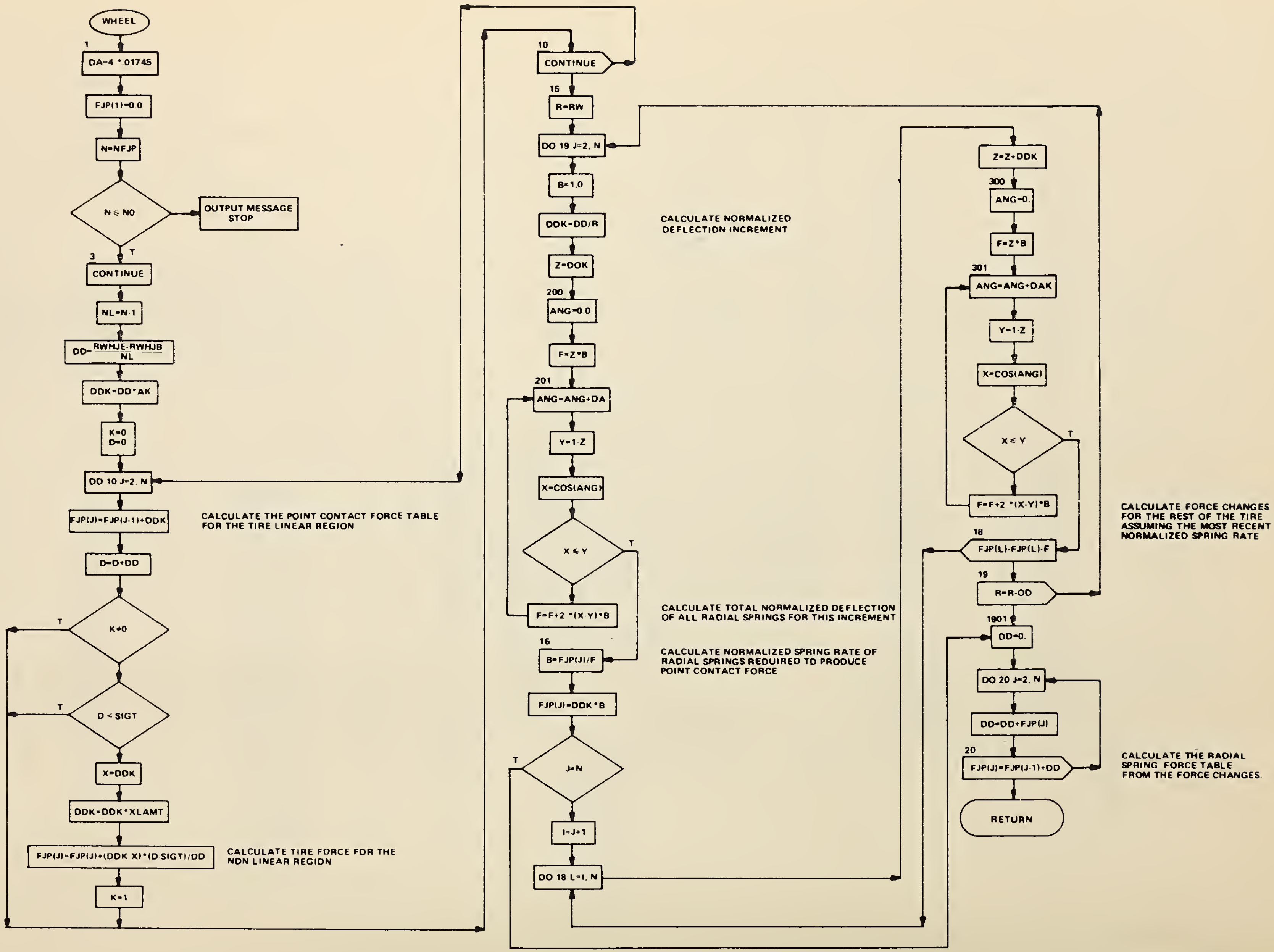
e. Common Variables Calculated:

None

f. Size:

$4FA)_{16} = 1274)_{10}$ bytes

g. Computational Procedure:


WHEEL
1
DA=4 * .01745
FJP(1)=0.0
N=NFJP
N ≤ NO
OUTPUT MESSAGE STOP
T
3
CONTINUE
NL=N-1
DD= RWHJE-RWHJB / NL
DDK=DD*AK
K=0 D=0
DO 10 J=2, N
FJP(J)=FJP(J-1)+DDK
CALCULATE THE POINT CONTACT FORCE TABLE FOR THE TIRE LINEAR REGION
D=D+DD
K≠0
T
D < SIGT
T
X=DDK
DDK=DDK*XLAMT
FJP(J)=FJP(J)+(DDK X)*(D-SIGT)/DD
CALCULATE TIRE FORCE FOR THE NON LINEAR REGION
K=1
10
CONTINUE
15
R=RW
DO 19 J=2, N
B=1.0
DDK=DD/R
CALCULATE NORMALIZED DEFLECTION INCREMENT
Z=DDK
200
ANG=0.0
F=Z*B
201
ANG=ANG+DA
Y=1-Z
X=COS(ANG)
X ≤ Y
T
F=F+2 *(X-Y)*B
CALCULATE TOTAL NORMALIZED DEFLECTION OF ALL RADIAL SPRINGS FOR THIS INCREMENT
16
B=FJP(J)/F
CALCULATE NORMALIZED SPRING RATE OF RADIAL SPRINGS REQUIRED TO PRODUCE POINT CONTACT FORCE
FJP(J)=DDK*B
J=N
T
I=J+1
DO 18 L=I, N
Z=Z+DDK
300
ANG=0.
F=Z*B
301
ANG=ANG+DAK
Y=1-Z
X=COS(ANG)
X ≤ Y
T
F=F+2 *(X-Y)*B
CALCULATE FORCE CHANGES FOR THE REST OF THE TIRE ASSUMING THE MOST RECENT NORMALIZED SPRING RATE
18
FJP(L)-FJP(L)-F
19
R=R-DD
1901
DD=0.
DO 20 J=2, N
DD=DD+FJP(J)
20
FJP(J)=FJP(J-1)+DD
CALCULATE THE RADIAL SPRING FORCE TABLE FROM THE FORCE CHANGES.
RETURN

3.2.2 HVOSM-VD2 Program Stops and Messages

Program stops include both normal and abnormal stops. Normal stops occur when the cumulative simulated time (T) exceeds the desired final time (T1) as input in field 2 of card 101, or when the magnitudes of both the linear and angular velocities of the vehicle sprung mass are less than or equal to the input minimums (UVWMIN and PQRMIN, card 101, fields 6 and 7). When these stops occur, no message is output and the program attempts to read another set of data cards.

Abnormal stops occur when a condition is encountered that the program is not designed to handle or an unresolvable error has occurred. The first type of abnormal stop occurs when rollover of the vehicle is immenent. That is, when the vehicle has rolled to an angle of 90° in either direction.

Abnormal stops are also indicated by a non-zero value for the variable ISTOP. The following codes identify the type and location of the error.

ISTOP = 1	Subroutine TIRFR. An error has occurred in determination of the wheel spin integration interval.
ISTOP = 4	Subroutine TMCNST. The denomenator of the expression used to calculate the value of PSIT after indexing of coordinate system is zero.
ISTOP = 5	Subroutine TMCNST. The logic associated with coordinate system indexing has been usable to determine the correct quadrant for PSIT, PHET or THETT.

ISTOP = 6 Subroutine TMCNST. The numerator in the expression for calculation of THETT after coordinate system indexing is zero.

ISTOP = 7 Subroutine TMCNST. The numerator in the expression for calculation of PHIT after coordinate system indexing is zero.

ISTOP = 30 Subroutine TMCNST. One of the recalculated Euler angles (PSIT, THETT, PHIT) has been computed as being very large (>3000 radians) after coordinate system indexing. A probable error has occurred.

When an ISTOP $\neq$ 0 condition is encountered, the program prints all output up to the time of the error, prints the value of ISTOP, terminates execution of the current run and attempts to read another set of data cards.

In subroutine CTQD, a message will be printed if the tabular time range of the TTS, TTR and TPC tables is exceeded. The program continues execution with the last entries in the tables.

Similarly, in subroutine CTQB, a message is printed if the temperature range of the FLF table is exceeded. The program again continues execution using the last value in the table.

In subroutine INPUT, the following messages are printed if difficulties are encountered in reading the card data deck.

UNEXPECTED END OF FILE ENCOUNTERED IN STMT NO. 1 OF
SUBROUTINE INPUT. LAST CARD READ WAS XXXX.

A CARD NUMBERED LESS THAN OR EQUAL TO ZERO WAS
ENCOUNTERED IN SUBROUTINE INPUT. CARD IMAGE
PRINTED ABOVE.

```
THE NUMBER OF CARDS READ IS ZERO.

A BLOCK NUMBER OF LESS THAN OR EQUAL TO ZERO HAS
BEEN OBTAINED.

A BLOCK NUMBER LARGER THAN THE ALLOWED NUMBER HAS
BEEN OBTAINED.

AN ERROR HAS OCCURRED IN STORING INPUT VALUES IN ONE
OF THE BLKXX SUBROUTINES.  THE CALLING ARGUMENTS
FROM INPUT ARE:  NBLK = XXXX   NBCRD = XXXX
NSEQ = XXXX   NCARD = XXXX.
```

In subroutine RUFRED, two messages may be printed if difficulties are encountered in reading road roughness data from FORTRAN device 4. They are:

```
END OF FILE ENCOUNTERED IN READ OF ROUGHNESS DATA
BEFORE NEND POINTS WERE READ.

NUMBER OF LAST ROUGHNESS DATA POINT IS GREATER THAN
THE ALLOWED 2200.  PROGRAM TERMINATED.
```

3.2.3 HVOSM-VD2 Program Listing

```
C          HIGHWAY VEHICLE OBJECT SIMULATION MODEL
C             MAIN ROUTINE
C         HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1            NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2            PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6           (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1           (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1           (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
```

```
     7         CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),          MAIN0500
     8         SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),MAIN0510
     9         FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)MAIN0520
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),     MAIN0530
     1         BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),       MAIN0540
     2         FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),     MAIN0550
     3         F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                    MAIN0560
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSTI(4)                      MAIN0570
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PH11),  MAIN0580
     1         (PSII(1),PSI1)                                          MAIN0590
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,P1,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MAIN0600
     1        GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,            MAIN0610
     2         TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,    MAIN0620
     3         BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,      MAIN0630
     4         XIZR,RTF,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,DIPD2,DIMD2,      MAIN0640
     5         ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMAIN0650
     6        ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,       MAIN0660
     7         SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,      MAIN0670
     8         SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,    MAIN0680
     9         ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ     MAIN0690
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3, MAIN0700
     1         ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,    MAIN0710
     2         TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MAIN0720
     3         HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PH1R2MAIN0730
     4        ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MAIN0740
     5         XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             MAIN0750
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                MAIN0760
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)      MAIN0770
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                    MAIN0780
     1             DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    MAIN0790
     2             PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    MAIN0800
     3             SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        MAIN0810
      LOGICAL LCB1,LCB2                                                MAIN0820
C      COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST  MAIN0830
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,   MAIN0840
     1             COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,   MAIN0850
     2             STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN    MAIN0860
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,               MAIN0870
     1             XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       MAIN0880
     2             SLOPE1,SLOPE2,XTRA(300)                             MAIN0890
      DIMENSION UI(4),VI(4),WI(4)                                      MAIN0900
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                     MAIN0910
      DIMENSION APITCH(4)                                              MAIN0920
      EQUIVALENCE (APITCH(1),APTCH1)                                   MAIN0930
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,    MAIN0940
     1            OMEGRC,AKRE,AKREP,OMEGRE,END3                        MAIN0950
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,       MAIN0960
     1             DAPRB,DAPRE,DDAPR,NAPR                              MAIN0970
      DIMENSION APF(21),APR(21)                                        MAIN0980
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                MAIN0990
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),  MAIN1000
```

```
     1              A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),      MAIN1010
     2              A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)           MAIN1020
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),   MAIN1030
     A              XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),              MAIN1040
     B              XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),          MAIN1050
     C              XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4             MAIN1060
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)       MAIN1070
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)       MAIN1080
      COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),    MAIN1090
     1             TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,MAIN1100
     2       DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),   MAIN1110
     3             SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),  MAIN1120
     4             EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),      MAIN1130
     5             FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),   MAIN1140
     6             SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),      MAIN1150
     7             FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),      MAIN1160
     8         ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),MAIN1170
     9         RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD    MAIN1180
      COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP MAIN1190
      LOGICAL  IUVS,IUVB,IRPS                                           MAIN1200
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, MAIN1210
     1             CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),      MAIN1220
     2             TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)MAIN1230
     3            ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,         MAIN1240
     4             BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)            MAIN1250
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,MAIN1260
     1             NDTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,  MAIN1270
     2             RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,     MAIN1280
     3             DELTAE,PIO15R,COMEN5,TQE,RPMF                        MAIN1290
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                     MAIN1300
      DIMENSION RPSI(4),DRPSI(4)                                        MAIN1310
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                   MAIN1320
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       MAIN1330
     1              AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),MAIN1340
     2              NCAMF,NCAMR,NDTHF,NDTHR                             MAIN1350
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,      MAIN1360
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     MAIN1370
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    MAIN1380
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    MAIN1390
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        MAIN1400
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     MAIN1410
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE, MAIN1420
     1             IDRVER,IBUG                                          MAIN1430
C        IPATHT - STOP FOR DRIVER MODEL                                 MAIN1440
C        IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL                MAIN1450
C        ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE           MAIN1460
C        ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES     MAIN1470
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,  MAIN1480
     1           TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,    MAIN1490
     2           TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,     MAIN1500
     3           VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,                   MAIN1510
```

```
     4            TESTT(5),DESSI(5),DIST1(5),PSIFHO,XIMPOR(9),                  MAIN1520
     5            BFP1,BFP2,DRIEND                                              MAIN1530
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,             MAIN1540
     1            TESTR1,TESTR2,THESKD,FBRK,AFB,DSDES,                          MAIN1550
     2            TRKIN,TMT,DESS,DIST,DISTC,CUNMPH,UT,UTMPH,                    MAIN1560
     3            APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSO2,QAY,          MAIN1570
     4            AXP,AYP,D1,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,             MAIN1580
     5            TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,            MAIN1590
     6            PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),            MAIN1600
     7            PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),            MAIN1610
     8            DEND                                                          MAIN1620
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                                  MAIN1630
     1              ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                                   MAIN1640
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                             MAIN1650
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                         MAIN1660
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                         MAIN1670
     5              YCMP(6),ZCMP(6),PHICM(6)                                    MAIN1680
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF                                      MAIN1690
      DIMENSION FJPP(35)                                                        MAIN1700
C                                                                               MAIN1710
C       VARIABLES NO LONGER USED INCLUDE TQF(50),TQR(50),TI(4),TIHI(4)          MAIN1720
C                                                                               MAIN1730
C                                                                               MAIN1740
C         SUBROUTINES DVDCHK AND DATE ARE RELATED TO THE OPERATING SYSTEMMAIN1750
C              AT OUR INSTALLATION                                              MAIN1760
C        SUBROUTINE DVDCHK CAN CAUSE HALT ON ATTEMPTED DIVIDE BY ZERO,          MAIN1770
C             EXPONENT OVERFLOW, AND MESSAGE ON EXPONENT UNDERFLOW.             MAIN1780
C          THE SERVICES GIVEN BY SUBROUTINE DVDCHK CAN NOW GIVEN BY             MAIN1790
C              FORTRAN EXTENDED ERROR HANDLING                                  MAIN1800
C        SUBROUTINE DATE RETURNS THE CURRENT DATE IN EIGHT  CHARACTERS.         MAIN1810
C                                                                               MAIN1820
      CALL CLEAR(VHED(1),NPAGE(20))                                             MAIN1830
      CALL CLEAR(TPATH,IBUG)                                                    MAIN1840
      CALL CLEAR(PHIO,PQRMIN)                                                   MAIN1850
      CALL CLEAR(YC1P,PSIFDO)                                                   MAIN1860
      CALL CLEAR( AKFC,END3)                                                    MAIN1870
      CALL CLEAR(APFR(1,1),NAPR)                                                MAIN1880
      CALL CLEAR(AKT(1),FJP(35,4))                                              MAIN1890
      CALL CLEAR(ITIR(1),ITIR(4))                                               MAIN1900
      CALL CLEAR (FIDJ(1),COMEN4)                                               MAIN1910
      CALL CLEAR(IBTYP(1),XINPT5(9))                                            MAIN1920
      CALL CLEAR(XIF,NDTHR)                                                     MAIN1930
      CALL CLEAR(NEN,DRIEND)                                                    MAIN1940
      CALL CLEAR(YC3P,PHICM(6))                                                 MAIN1950
      CALL CLEAR(DELG,IRUF)                                                     MAIN1960
C                                                                               MAIN1970
      ARBRF = 1.0                                                               MAIN1980
      ARBRR = 1.0                                                               MAIN1990
      CALL DVDCHK                                                               MAIN2000
    1 CALL CLEAR(NEQ,DER(50))                                                   MAIN2010
      CALL CLEAR(SUMM,LLL)                                                      MAIN2020
```

```
      CALL CLEAR(FRSP(1),FRCP(4))
      CALL CLEAR(DPSINT,XMUGI(4))
      CALL CLEAR(XINDN,ENDEIN)
      CALL CLEAR(U1,XTRA(300))
      CALL CLEAR(X1P,CGH(4))
      CALL CLEAR(A234(1),A23(4))
      CALL CLEAR (FIDAR(1),ISTOP)
      CALL CLEAR(TAU(1),RPME)
      CALL CLEAR(NEQR,DERR(12))
      CALL CLEAR(XMURO2,SNPF)
      CALL CLEAR(EN,DEND)
C
C       SET IDRIVE = 1 IN DRVCNS AT BEGINNING OF COMPUTATION
C       SET ITESTT = 2 IN DRVCNS AND INITIALIZE DESS AND DIST
C       SET TCTEST(I),I=1,5  TO INPUT VALUES. TCTEST(6)SET LARGE VALUE.
C       SET ITCHNG = 0 FOR FIRST INTERVAL
C       SUBROUTINE DRIVER WILL DETERMINE PSII DURING FIRST INTERVAL
      DO 99 I=1,6
   99 TCTEST(I) = 1.0E20
      TPATH = 1.0E20
      DELPTH = 0.0
C       SET TPATH AND DELPTH TO INPUT VALUES IN DRVCNS
C       TPATH AND DELTPH ARE CONTROLS FOR DRIVER SAMPLING USED TO
C       RESET THE INDICATOR IDRIVE.
C
      IRPS = .FALSE.
      CALL INPUT
      CALL DATE(DADE)
      IF(IRUF.NE.0) CALL RUFRED(NEND,DELG,DGMAX,ZGP)
      IF(INDCRB.NE.1.AND.IRUF.EQ.0) GO TO 10
      NFJP = (RWHJE-RWHJB)/DRWHJ + 1.2
      DO 11 I=1,4
      IF(I.EQ.1) GO TO 12
      IM = I-1
      DO 15 K=1,IM
      IF(ITIR(I).EQ.ITIR(K)) GO TO 16
   15 CONTINUE
   12 CALL WHEEL(AKT(I),SIGT(I),XLAMT(I),RWHJB,RWHJE,DRWHJ,NFJP,
     1           RW(I),FJPP,35)
      DO 13 J=1,NFJP
   13 FJP(J,I) = FJPP(J)
      GO TO 11
   16 DO 17 J=1,NFJP
   17 FJP(J,I) = FJP(J,K)
   11 CONTINUE
   10 CONTINUE
      IF(ZF.EQ.0.0.AND.ZR.EQ.0.0) CALL INITEQ
      CALL CNSTNT
      IF(IDRVER.NE.0) CALL DRVCNS
      CALL IDOUT
      TCTEST(6) = 1.0E20
```

```
  100 DO 101  I=1,4                                                      MAIN2540
      LCB1(I) = .FALSE.                                                  MAIN2550
      LCB2(I) = .FALSE.                                                  MAIN2560
  101 CONTINUE                                                           MAIN2570
      TPRINT  = TO                                                       MAIN2580
      THETMX = ABS(THMAX) * RAD                                          MAIN2590
      UVWM2  = UVWMIN**2                                                 MAIN2600
      PQRM2  =  PQRMIN**2                                                MAIN2610
       UVWM2  = SIGN(UVWM2,UVWMIN)                                       MAIN2620
       PQRM2  = SIGN(PQRM2,PQRMIN)                                       MAIN2630
      ICBHIT  = 0                                                        MAIN2640
      JCBHIT  = 0                                                        MAIN2650
      IHIT = 0                                                           MAIN2660
      IND  = 0                                                           MAIN2670
      T = TO                                                             MAIN2680
      DT = DTCOMP                                                        MAIN2690
      NEQ = 22                                                           MAIN2700
      NEQR   = 4                                                         MAIN2710
      PSIMAX = 135.0*RAD                                                 MAIN2720
      CALL PLOTTP(1)                                                     MAIN2730
      CALL OUTPUT(0)                                                     MAIN2740
C      ISTEP FOR COUNT OF OUTER INTEGRATION STEP FOR USE IN 'NESTED SUM'MAIN2750
    2 ISTEP = 0                                                          MAIN2760
      CALL PINT1(1,MODE,NEQ,T,DT,U,DU,EBAR)                              MAIN2770
      IDRIVE = 0                                                         MAIN2780
      IF (ISTOP.NE. 0) GO TO 6                                           MAIN2790
    3 IF(TPRINT.GT.T+.1*DT) GO TO 4                                      MAIN2800
      CALL OUTPUT(I)                                                     MAIN2810
      TPRINT  = TPRINT+DTPRNT                                            MAIN2820
      CALL PLOTTP(2)                                                     MAIN2830
    4 IDRIVE = 0                                                         MAIN2840
      IF(TPATH.GT. T+0.1*DT) GO TO 40                                    MAIN2850
C          SUBROUTINE DRIVER WILL DETERMINE PSI1 DURING FIRST INCREMENT  MAIN2860
C          TO AVOID, INITIALIZE TPATH ABOVE AS TO+DELPTH                 MAIN2870
      IDRIVE = 1                                                         MAIN2880
      TPATH = TPATH + DELPTH                                             MAIN2890
      ITCHNG = 0                                                         MAIN2900
      IF(TCTEST(ITESTT).GT.(T+0.I*DT)) GO TO 41                          MAIN2910
      ITCHNG  = 1                                                        MAIN2920
      ITESTT  = ITESTT+1                                                 MAIN2930
C        AT DRIVER SAMPLE TIME, TEST THE TIME FOR CHANGING DESIRED       MAIN2940
C        SPEED AND DISTANCE. FIRST VALUE OF TCTEST SHOULD BE EQUAL TO    MAIN2950
C         TO AND FIRST VALUE OF DESIRED SPEED SHOULD BE UO.              MAIN2960
   41 CONTINUE                                                           MAIN2970
   40 ISTEP = 0                                                          MAIN2980
      CALL PINT1(2,MODE,NEQ,T,DT,U,DU,EBAR)                              MAIN2990
      IF (ISTOP.NE. 0) GO TO 6                                           MAIN3000
C         THETTL,PHITL,PSITL ARE VALUES OF THETT,PHIT,PSIT FROM PREVIOUMAIN3010
C           TIME INTERVAL, USED TO TEST NEW ANGLES     IN SUBROUTINE TMCMAIN3020
      THETTL = THETT                                                     MAIN3030
      PHITL  = PHIT                                                      MAIN3040
```

```
      PSITL  = PSIT
C     CALL EGYSUM   FORMERLY USED IN SPRUNG MASS IMPACT VERSION
      IF(T.GE.T1) GO TO 6
      IF(U**2+V**2+W**2.LE.UVWM2.AND.P2+Q2+R2.LE.PQRM2) GO TO 6
      IF(ABS(PHIT).GE.PIO2) GO TO 6
      IF(IPATHT .NE. 0) GO TO 6
      IF(ABS(THETTP).LT.THETMX) GO TO 5
C      XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTX
C            XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING
C            THAT IS  THETN OR PHIN NOW .NE. 0.0
C            USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST
      THETN = THETT
      THETTP= 0.0
      PHIN  = PHIT
      PHITP = 0.0
      PSIN  = PSIT
      PSITP = 0.0
      XINDL = XINDN
      XINDN = XINDN + 1.0
C     IND=1  INDICATOR FOR RE-INITIALIZATION IN PINT1
      IND = I
    5 IF(INDCRB.EQ.0) GO TO 56
   50 DO 51  I=1,4
      IF(.NOT.LCB2(I)) GO TO 53
   51 CONTINUE
      ICBHIT = 2
   52 IF(ICBHIT.EQ.JCBHIT) GO TO 56
      JCBHIT = ICBHIT
      IND = 1
      DT = DTCOMP
      GO TO 56
   53 DO 54  I=1,4
      IF(LCB1(I)) GO TO 55
   54 CONTINUE
      ICBHIT = 0
      GO TO 52
   55 ICBHIT = 1
      IHIT = 1
      IF (ICBHIT.EQ.JCBHIT) GO TO 56
      JCBHIT = ICBHIT
      IND = 0
      DT = DELTC
      GO TO 2
C
   56 CONTINUE
   58 IF(IND.EQ.0) GO TO 3
      IND = 0
      GO TO 2
C
    6 CALL OUTPUT(1)
      CALL PLOTTP(3)
```

```
      IF(ISTOP .NE. 0) WRITE(6,59) ISTOP                                MAIN3560
   59 FORMAT(17H    ERROR, ISTOP = ,  I3)                               MAIN3570
C       CALL PLOTTP(3) CAUSES DISTINCTIVE RECORD ON TAPE FOR END OF RUN.MAIN3580
      GO TO 1                                                           MAIN3590
      END                                                               MAIN3600
```

```
      SUBROUTINE ADJTQB                                                 ADJT0010
C             HVOSM-VD2 VERSION                                         ADJT0020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                ADJT0030
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,ADJT0040
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   ADJT0050
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),ADJT0060
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     ADJT0070
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  ADJT0080
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  ADJT0090
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     ADJT0100
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      ADJT0110
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),ADJT0120
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)ADJT0130
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      ADJT0140
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     ADJT0150
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   ADJT0160
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  ADJT0170
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       ADJT0180
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   ADJT0190
     1           (PSII(1),PSI1)                                         ADJT0200
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, ADJT0210
     1             CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),      ADJT0220
     2             TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)ADJT0230
     3            ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,         ADJT0240
     4             BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)            ADJT0250
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,ADJT0260
     1             NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,  ADJT0270
     2             RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,     ADJT0280
     3             DELTAE,PIO15R,COMEN5,TQE,RPME                        ADJT0290
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                     ADJT0300
      DIMENSION RPSI(4),DRPSI(4)                                        ADJT0310
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                   ADJT0320
C                                                                       ADJT0330
C  SUBROUTINE FOR ADJUSTMENT OF TQB                                     ADJT0340
C       CALLED BY SUBROUTINE DAUXR, WHICH IS CALLED BY SUBROUTINE TIRFADJT0350
C                                                                       ADJT0360
      XLAMB = TLAMB(JDEND)                                              ADJT0370
      DO 60 I=1,4,2                                                     ADJT0380
      ARPS = ABS(RPSI(I))                                               ADJT0390
      ARPS1 = ABS(RPSI(I+1))                                            ADJT0400
      ITRA = 0                                                          ADJT0410
      IF(ARPS.LE.ZETAB.OR.ARPS1.LE.ZETAB) GO TO 12                      ADJT0420
      TQB(I) = -SIGN(TQB(I),RPSI(I))                                    ADJT0430
   10 TQB(I+1) = -SIGN(TQB(I+1),RPSI(I+1))                              ADJT0440
      GO TO 60                                                          ADJT0450
   12 IF(ARPS.LE.ZETAB.AND.ARPS1.LE.ZETAB) GO TO 15                     ADJT0460
      ITRA = 1                                                          ADJT0470
      IF(ARPS.LE.ZETAB) ITRA=2                                          ADJT0480
      GO TO (16,26), ITRA                                               ADJT0490
```

```
15 IF((ABS(FC(I)*HI(I))/12.0).LT.ABS(TQB(I))) GO TO 25
16 TTTA = FC(I+1) * HI(I+1)
   TTTB = FC(I)  * HI(I)
   TTTC = 12.0 * TQB(I)
   TTTD = TQB(I+1)
   TTTE = RPSI(I)
   TTTF = RPSI(I+1)
   J = I
   JJ = I + 1
   GO TO 40
25 IF((ABS(FC(I+1)*HI(I+1))/12.0).LT.ABS(TQB(I+1)))  GO TO 35
26 TTTA = FC(I)   * HI(I)
   TTTB = FC(I+1) * HI(I+1)
   TTTC = 12.0 * TQB(I+1)
   TTTD = TQB(I)
   TTTE = RPSI(I+1)
   TTTF = RPSI(I)
   J = I+1
   JJ = I
   GO TO 40
35 TQB(I) =(FC(I) * HI(I))/12.0
   TQB(I) = -ABS(TQB(I)) * SIGN(1.0,RPSI(I))
   TQB(I+1) = -(ABS(FC(I+1)*HI(I+1))/12.0) * SIGN(1.0,RPSI(I+1))
   GO TO 60
40 TTTXA = SIGN(1.0,TTTB)
   TTTXB = SIGN(1.0,TTTF)
   IF(ITRA.GT.0) GO TO 44
   IF(TTTXA*TTTXB.GT.0.0) GO TO 45
44 TQB(J) =  SIGN(TQB(J), -TTTE)
   GO TO 46
45 TQB(J) = 0.0
46 TTTXC = TTTA -  XLAMB *(TTTB - TTTC)
   TTTXD = SIGN(1.0,TTTXC)
   IF(TTTXD*SIGN(1.0,-TTTF) .GT. 0.0) GO TO 51
   TQB(JJ)=0.0
   GO TO 60
51 IF(ABS(TTTXC/12.0).LT.ABS(TTTD)) GO TO 55
   TQB(JJ) = SIGN(TQB(JJ),-TTTF)
   GO TO 60
55 TQB(JJ) = -ABS(TTTXC/12.0) * SIGN(1.0, TTTF)
60 CONTINUE
   RETURN
   END
```

```
      SUBROUTINE BLK01(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK10010
C             HVOSM-VD2 VERSION                                         BLK10020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                BLK10030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK10040
     1             NPAGE(20)                                            BLK10050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK10060
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      BLK10070
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      BLK10080
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK10090
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  BLK10100
     5           TI,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     BLK10110
     6           HED(36),DADE(3),XIR,XI,YI,Z1,X2,Y2,Z2,PHIC(50),DELB,   BLK10120
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   BLK10130
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    BLK10140
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  BLK10150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK10160
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PORMIN  BLK10170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK10180
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK10190
     2             PSIFIO,PSIFDO                                        BLK10200
      DIMENSION YC1P(2)                                                 BLK10210
      EQUIVALENCE (YC1P(1),YC1P)                                        BLK10220
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),   BLK10230
     A               XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),             BLK10240
     B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),         BLK10250
     C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4            BLK10260
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)       BLK10270
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)       BLK10280
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       BLK10290
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),BLK10300
     2               NCAMF,NCAMR,NDTHF,NDTHR                            BLK10310
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE, BLK10320
     1              IDRVER,IBUG                                         BLK10330
C        IPATHT - STOP FOR DRIVER MODEL                                 BLK10340
C        IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL                BLK10350
C        ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE           BLK10360
C        ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES     BLK10370
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          BLK10380
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                          BLK10390
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                    BLK10400
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                BLK10410
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                BLK10420
     5               YCMP(6),ZCMP(6),PHICM(6)                           BLK10430
      DIMENSION DUM(18)                                                 BLK10440
      DATA NBS/4/                                                       BLK10450
      NBT = NBCRD+1                                                     BLK10460
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK10470
      GO TO(100,101,102,103,104),NBT                                    BLK10480
      GO TO 98                                                          BLK10490
```

```
100 IF(NCARD.NE.100) GO TO 98
    DO 10 I=1,18
 10 HED(I) = DUM(I)
    GO TO 99
101 IF(NCARD.NE.101) GO TO 98
    T0     = DUM(1)
    T1     = DUM(2)
    DTCOMP = DUM(3)
    DTPRNT = DUM(4)
    THMAX  = DUM(5)
    UVWMIN = DUM(6)
    PQRMIN = DUM(7)
    COMEN4 = DUM(8)
    GO TO 99
102 IF(NCARD.NE.102) GO TO 98
    ISUS = IFIX(DUM(1))
    INDCRB = IFIX(DUM(2))
    NCRBSL = IFIX(DUM(3))
    DELTC = DUM(4)
    IDRVER = IFIX(DUM(5))
    IBUG = IFIX(DUM(6))
    IO = 0
    IF(IDRVER.NE.0) IO = 1
    NPAGE(18) = IO
    NPAGE(20) = IO
    IF(INDCRB.NE.0) NPAGE(5) = 1
    GO TO 99
103 IF(NCARD.NE.103) GO TO 98
    MODE = DUM(1)
    EBAR = DUM(2)
    EM   = DUM(3)
    AAA  = DUM(4)
    HMAX = DUM(5)
    HMIN = DUM(6)
    BET  = DUM(7)
    GO TO 99
104 IF(NCARD.NE.104) GO TO 98
    NPAGE(4) = IFIX(DUM(1))
    NPAGE(6) = IFIX(DUM(2))
    NPAGE(7) = IFIX(DUM(3))
    NPAGE(8) = IFIX(DUM(4))
    NPAGE(9) = IFIX(DUM(5))
    NPAGE(10)= IFIX(DUM(6))
    NPAGE(14)= IFIX(DUM(7))
    NPAGE(19)= IFIX(DUM(8))
    GO TO 99
 98 NERR = 1
 99 RETURN
    END
```

```
      SUBROUTINE BLK02(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK20010
C             HVOSM-VD2 VERSION                                         BLK20020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                BLK20030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK20040
     1             NPAGE(20)                                            BLK20050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK20060
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     BLK20070
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     BLK20080
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK20090
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, BLK20100
     5            T1,DTCMP1,DTPRNT,MODE,FBAR,EM,AAA,HMAX,HMIN,BET,G,    BLK20110
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  BLK20120
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  BLK20130
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   BLK20140
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 BLK20150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK20160
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN   BLK20170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK20180
     1             CPSP,OMGPS.AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK20190
     2             PSIFIO,PSIFDO                                        BLK20200
      DIMENSION YC1P(2)                                                 BLK20210
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK20220
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,     BLK20230
     1              OMEGRC,AKRE,AKREP,OMEGRE,END3                       BLK20240
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,        BLK20250
     1               DAPRB,DAPRE,DDAPR,NAPR                             BLK20260
      DIMENSION APF(21),APR(21)                                         BLK20270
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                 BLK20280
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),   BLK20290
     A               XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),             BLK20300
     B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),         BLK20310
     C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4            BLK20320
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)       BLK20330
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)       BLK20340
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, BLK20350
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),     BLK20360
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)BLK20370
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,        BLK20380
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)           BLK20390
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       BLK20400
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),BLK20410
     2               NCAMF,NCAMR,NDTHF,NDTHR                            BLK20420
      DIMENSION DUM(18)                                                 BLK20430
      DATA NBS/17/                                                      BLK20440
      NBT=NBCRD+1                                                       BLK20450
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK20460
      GO TO(200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,BLK20470
     1      215,216,217),NBT                                            BLK20480
      GO TO 98                                                          BLK20490
```

```
200 IF(NCARD.NE.200) GO TO 98
    DO 10 I=1,18
 10 VHED(I) = DUM(I)
    GO TO 99
201 IF(NCARD.NE.201) GO TO 98
    XMS   = DUM(1)
    XMUF  = DUM(2)
    XMUR  = DUM(3)
    XIX   = DUM(4)
    XIY   = DUM(5)
    XIZ   = DUM(6)
    XIXZ  = DUM(7)
    XIR   = DUM(8)
    XIF = DUM(9)
    GO TO 99
202 IF(NCARD.NE.202) GO TO 98
    A    = DUM(1)
    B    = DUM(2)
    TF   = DUM(3)
    TR   = DUM(4)
    RHO  = DUM(5)
    TS = DUM(6)
    RHOF  = DUM(7)
    TSF  = DUM(8)
    G = 386.4
    IF(DUM(9).NE.0.0) G = DUM(9)
    GO TO 99
203 IF(NCARD.NE.203) GO TO 98
    X1 = DUM(1)
    Y1 = DUM(2)
    Z1 = DUM(3)
    X2 = DUM(4)
    Y2 = DUM(5)
    Z2 = DUM(6)
    DO 30 J=1,6
    IF(DUM(J).NE.0.0) NPAGE(16) = 1
 30 CONTINUE
    ZF = DUM(7)
    ZR = DUM(8)
    GO TO 99
204 IF(NCARD.NE.204) GO TO 98
    AKF     = DUM(1)
    AKFC    = DUM(2)
    AKFCP   = DUM(3)
    AKFE    = DUM(4)
    AKFEP   = DUM(5)
    XLAMF   = DUM(6)
    OMEGFC  = DUM(7)
    OMEGFE  = DUM(8)
    GO TO 99
205 IF(NCARD.NE.205)GO TO 98
```

```
      AKR    = DUM(1)                                                   BLK21010
      AKRC   = DUM(2)                                                   BLK21020
      AKRCP  = DUM(3)                                                   BLK21030
      AKRE   = DUM(4)                                                   BLK21040
      AKREP  = DUM(5)                                                   BLK21050
      XLAMR  = DUM(6)                                                   BLK21060
      OMEGRC = DUM(7)                                                   BLK21070
      OMEGRE = DUM(8)                                                   BLK21080
      GO TO 99                                                          BLK21090
  206 IF(NCARD.NE.206) GO TO 98                                         BLK21100
      CF   = DUM(1)                                                     BLK21110
      CFP  = DUM(2)                                                     BLK21120
      EPSF = DUM(3)                                                     BLK21130
      CR   = DUM(4)                                                     BLK21140
      CRP  = DUM(5)                                                     BLK21150
      EPSR = DUM(6)                                                     BLK21160
      GO TO 99                                                          BLK21170
  207 IF(NCARD.NE.207) GO TO 98                                         BLK21180
      RF = DUM(1)                                                       BLK21190
      RR = DUM(2)                                                       BLK21200
      AKRS = DUM(3)                                                     BLK21210
      AKDS = DUM(4)                                                     BLK21220
      AKDS1 = DUM(5)                                                    BLK21230
      AKDS2 = DUM(6)                                                    BLK21240
      AKDS3 = DUM(7)                                                    BLK21250
      GO TO 99                                                          BLK21260
  208 IF(NCARD.NE.208) GO TO 98                                         BLK21270
      XIPS  = DUM(1)                                                    BLK21280
      CPSP  = DUM(2)                                                    BLK21290
      OMGPS = DUM(3)                                                    BLK21300
      AKPS  = DUM(4)                                                    BLK21310
      EPSPS = DUM(5)                                                    BLK21320
      XPS   = DUM(6)                                                    BLK21330
      GO TO 99                                                          BLK21340
  209 IF(NCARD.NE.209.OR.NSEQ.NE.0) GO TO 98                            BLK21350
      DELB = DUM(1)                                                     BLK21360
      DELE = DUM(2)                                                     BLK21370
      DDEL = DUM(3)                                                     BLK21380
      NDTHF = DUM(4)                                                    BLK21390
      NDTHR = DUM(5)                                                    BLK21400
      NDEL = (DELE-DELB)/DDEL +1                                        BLK21410
      NCRDS = (NDEL-1)/9 + 1                                            BLK21420
      CALL TREAD(NCARD,NCRDS,NDEL,50,PHIC,NERR)                         BLK21430
      IF(NERR.NE.0) GO TO 98                                            BLK21440
      IF(ISUS.EQ.1) CALL TREAD(NCARD,NCRDS,NDEL,50,PHIRC,NERR)          BLK21450
      IF(NERR.NE.0) GO TO 98                                            BLK21460
      IF(NDTHF.NE.0) CALL TREAD(NCARD,NCRDS,NDEL,50,DTHF,NERR)          BLK21470
      IF(NERR.NE.0) GO TO 98                                            BLK21480
      IF(NDTHR.NE.0) CALL TREAD(NCARD,NCRDS,NDEL,50,DTHR,NERR)          BLK21490
      IF(NERR.NE.0) GO TO 98                                            BLK21500
      GO TO 99                                                          BLK21510
```

```
210 IF(NCARD.NE.210.OR.NSEQ.NE.0) GO TO 98                            BLK21520
    DAPFB = DUM(1)                                                    BLK21530
    DAPFE = DUM(2)                                                    BLK21540
    DDAPF = DUM(3)                                                    BLK21550
    NAPF = (DAPFE-DAPFB)/DDAPF + 1                                    BLK21560
    NCRDS = (NAPF-1)/9 + 1                                            BLK21570
    CALL TREAD(NCARD,NCRDS,NAPF,21,APF,NERR)                          BLK21580
    IAPFR(1) = 1                                                      BLK21590
    IF(NERR.NE.0) GO TO 98                                            BLK21600
    GO TO 99                                                          BLK21610
211 IF(NCARD.NE.211.OR.NSEQ.NE.0) GO TO 98                            BLK21620
    DAPRB = DUM(1)                                                    BLK21630
    DAPRE = DUM(2)                                                    BLK21640
    DDAPR = DUM(2)                                                    BLK21650
    NAPR = (DAPRE-DAPRB)/DDAPR + 1                                    BLK21660
    NCRDS = (NAPF-1)/9 + 1                                            BLK21670
    CALL TREAD(NCARD,NCRDS,NAPR,21,APR,NERR)                          BLK21680
    IAPFR(2) = 1                                                      BLK21690
    IF(NERR.NE.0)GO TO 98                                             BLK21700
    GO TO 99                                                          BLK21710
212 IF(NCARD.NE.212) GO TO 98                                         BLK21720
    FIDJF = DUM(1)                                                    BLK21730
    FIWJF = DUM(2)                                                    BLK21740
    FIDJR = DUM(3)                                                    BLK21750
    FIWJR = DUM(4)                                                    BLK21760
    ARBRF = DUM(5)                                                    BLK21770
    ARBRR = DUM(6)                                                    BLK21780
    IF(ARBRF.EQ.0.0) ARBRF = 1.0                                      BLK21790
    IF(ARBRR.EQ.0.0) ARBRR = 1.0                                      BLK21800
    GO TO 99                                                          BLK21810
213 IF(NCARD.NE.213) GO TO 98                                         BLK21820
    AK1 = DUM(1)                                                      BLK21830
    AK2 = DUM(2)                                                      BLK21840
    PONE = DUM(3)                                                     BLK21850
    PTWO = DUM(4)                                                     BLK21860
    PZERO(1) = DUM(5)                                                 BLK21870
    PZERO(2) = DUM(6)                                                 BLK21880
    ZETAB = DUM(7)                                                    BLK21890
    GO TO 99                                                          BLK21900
214 IF(NCARD.NE.214.OR.NSEQ.NE.0) GO TO 98                            BLK21910
    IBTYP(1) = DUM(1)                                                 BLK21920
    IBTYP(2) = DUM(2)                                                 BLK21930
    CALL T2READ(NCARD,16,16,2,GN,NERR)                                BLK21940
    IF(NERR.NE.0) GO TO 98                                            BLK21950
    GO TO 99                                                          BLK21960
215 IF(NCARD.NE.215.OR.NSEQ.NE.0) GO TO 98                            BLK21970
    BRPM = DUM(1)                                                     BLK21980
    ERPM = DUM(2)                                                     BLK21990
    DRPM = DUM(3)                                                     BLK22000
    NRPM = (ERPM-BRPM)/DRPM + 1.2                                     BLK22010
    NCRDS = (NRPM-1)/9 + 1                                            BLK22020
```

```
      CALL TREAD(NCARD,NCRDS,NRPM,12,TWOT,NERR)
      IF(NERR.NE.0) GO TO 98
      CALL TREAD(NCARD,NCRDS,NRPM,12,TCT,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  216 IF(NCARD.NE.216.OR.NSEQ.NE.0) GO TO 98
      BTLF = DUM(1)
      ETLF = DUM(2)
      DTLF = DUM(3)
      NTLF = (ETLF-BTLF)/DTLF + 1.2
      NCRDS = (NTLF-1)/9 + 1
      CALL TREAD(NCARD,NCRDS,NTLF,51,TLF,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  217 IF(NCARD.NE.217) GO TO 98
      CONE = DUM(1)
      CTWO = DUM(2)
      CTHREE = DUM(3)
      GO TO 99
   98 NERR= 1
   99 RETURN
      END
```

```
      SUBROUTINE BLK03(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK30010
C             HVOSM-VD2 VERSION                                         BLK30020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                BLK30030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK30040
     1             NPAGE(20)                                            BLK30050
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK30060
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK30070
     2             PSIFIO,PSIFDC                                        BLK30080
      DIMENSION YC1P(2)                                                 BLK30090
      EQUIVALENCE (YC1P(1),YC1P)                                        BLK30100
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   BLK30110
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     BLK30120
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          BLK30130
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),   BLK30140
     A               XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),             BLK30150
     B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),         BLK30160
     C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4            BLK30170
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)       BLK30180
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)       BLK30190
      DIMENSION DUM(18),TDUM(9,4)                                       BLK30200
      DIMENSION TDUM1(6,6),TDUM2(6,6),TDUM3(6,6),TDUM4(6,6)             BLK30210
      DATA NBS/6/                                                       BLK30220
      NBT = NBCRD+1                                                     BLK30230
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK30240
      GO TO (300,301,302,303,304,305),NBT                               BLK30250
      GO TO 98                                                          BLK30260
  300 IF(NCARD.NE.300) GO TO 98                                         BLK30270
      DO 10 I=1,18                                                      BLK30280
   10 THED(I) = DUM(I)                                                  BLK30290
      GO TO 99                                                          BLK30300
  301 IF(NCARD.NE.301) GO TO 98                                         BLK30310
      ITIR(1) = DUM(1)                                                  BLK30320
      ITIR(2) = DUM(2)                                                  BLK30330
      ITIR(3) = DUM(3)                                                  BLK30340
      ITIR(4) = DUM(4)                                                  BLK30350
      DO 319 I=1,4                                                      BLK30360
  319 AMU(I) = DUM(5)                                                   BLK30370
      RWHJE = DUM(6)                                                    BLK30380
      DRWHJ = DUM(7)                                                    BLK30390
      NXFRCP = IFIX(DUM(8))                                             BLK30400
      NXUGMU = IFIX(DUM(9))                                             BLK30410
      CALL TREAD(NCARD,1,NXFRCP,6,XXFRCP,NERR)                          BLK30420
      IF(NERR.NE.0) GO TO 98                                            BLK30430
      CALL TREAD(NCARD,1,NXUGMU,6,XXUGMU,NERR)                          BLK30440
      IF(NERR.NE.0) GO TO 98                                            BLK30450
      GO TO 99                                                          BLK30460
  302 IF(NCARD.NE.302) GO TO 98                                         BLK30470
      NTIR = DUM(1)                                                     BLK30480
      GO TO 320                                                         BLK30490
```

```
303 IF(NCARD.NE.303) GO TO 98                                  BLK30500
    NTIR = DUM(1)                                              BLK30510
    GO TO 320                                                  BLK30520
304 IF(NCARD.NE.304) GO TO 98                                  BLK30530
    NTIR = DUM(1)                                              BLK30540
    GO TO 320                                                  BLK30550
305 IF(NCARD.NE.305) GO TO 98                                  BLK30560
    NTIR = DUM(1)                                              BLK30570
320 CALL TREAD(NCARD,1,8,9,DUM,NERR)                           BLK30580
    IF(NERR.NE.0) GO TO 98                                     BLK30590
    DO 321 I=1,4                                               BLK30600
    IF(ITIR(I).NE.NTIR) GO TO 321                              BLK30610
    AKT(I) = DUM(1)                                            BLK30620
    SIGT(I) = DUM(2)                                           BLK30630
    XLAMT(I) = DUM(3)                                          BLK30640
    A0(I) = DUM(4)                                             BLK30650
    A1(I) = DUM(5)                                             BLK30660
    A2(I) = DUM(6)                                             BLK30670
    A3(I) = DUM(7)                                             BLK30680
    A4(I) = DUM(8)                                             BLK30690
321 CONTINUE                                                   BLK30700
    CALL TREAD(NCARD,1,5,9,DUM,NERR)                           BLK30710
    IF(NERR.NE.0) GO TO 98                                     BLK30720
    DO 322 I=1,4                                               BLK30730
    IF(ITIR(I).NE.NTIR) GOTO 322                               BLK30740
    OMEGT(I) = DUM(1)                                          BLK30750
    RW(I) = DUM(2)                                             BLK30760
    XMUM(I) = DUM(3)                                           BLK30770
    CT(I) = DUM(4)                                             BLK30780
    RRMC(I) = DUM(5)                                           BLK30790
322 CONTINUE                                                   BLK30800
    CALL T2READ(NCARD,6,NXFRCP,NXUGMU,TDUM1,NERR)              BLK30810
    IF(NERR.NE.0) GO TO 98                                     BLK30820
    CALL T2READ(NCARD,6,NXFRCP,NXUGMU,TDUM2,NERR)              BLK30830
    IF(NERR.NE.0) GO TO 98                                     BLK30840
    CALL T2READ(NCARD,6,NXFRCP,NXUGMU,TDUM3,NERR)              BLK30850
    IF(NERR.NE.0) GO TO 98                                     BLK30860
    CALL T2READ(NCARD,6,NXFRCP,NXUGMU,TDUM4,NERR)              BLK30870
    IF(NERR.NE.0) GO TO 98                                     BLK30880
    DO 325 I=1,4                                               BLK30890
    IF(ITIR(I).NE.NTIR) GO TO 325                              BLK30900
    DO 326 J=1,NXFRCP                                          BLK30910
    DO 326 K=1,NXUGMU                                          BLK30920
    XMUMAT(J,K,I) = TDUM1(J,K)                                 BLK30930
    XMXPMT(J,K,I) = TDUM2(J,K)                                 BLK30940
    XMXSMT(J,K,I) = TDUM3(J,K)                                 BLK30950
326 SLIPMT(J,K,I) = TDUM4(J,K)                                 BLK30960
325 CONTINUE                                                   BLK30970
    GO TO 99                                                   BLK30980
 98 NERR = 1.0                                                 BLK30990
 99 RETURN                                                     BLK31000
```

END BLK31010

```
      SUBROUTINE BLK04(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK40010
C               HVOSM-VD2 VERSION                                        BLK40020
C               REVISED OCTOBER 1975   CALSPAN CORPORATION               BLK40030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK40040
     1              NPAGE(20)                                           BLK40050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK40060
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     BLK40070
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     BLK40080
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK40090
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, BLK40100
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    BLK40110
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  BLK40120
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  BLK40130
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   BLK40140
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 BLK40150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK40160
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN BLK40170
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, BLK40180
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),     BLK40190
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)BLK40200
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,        BLK40210
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)           BLK40220
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,  BLK40230
     1             TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,  BLK40240
     2             TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,   BLK40250
     3             VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,                 BLK40260
     4             TESTT(5),DESSI(5),DISTI(5),PSIFHO,XIMPOR(9),         BLK40270
     5             BFP1,BFP2,DRIEND                                     BLK40280
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,     BLK40290
     1             TESTR1,TESTR2,THESKD,FBRK,APB,DSDES,                 BLK40300
     2             TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,           BLK40310
     3             APD,DELTAX,DELTV,UJ,TTEM,TTPSIT,PSISKD,ST,STSO2,QAY, BLK40320
     4             AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,    BLK40330
     5             TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,   BLK40340
     6             PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),   BLK40350
     7             PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),   BLK40360
     8             DEND                                                 BLK40370
      DIMENSION DUM(18)                                                 BLK40380
      DATA NBS/11/                                                      BLK40390
      NBT = NBCRD+1                                                     BLK40400
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK40410
      GO TO (400,401,402,403,404,405,406,407,408,409,410,411),NBT       BLK40420
      GO TO 98                                                          BLK40430
  400 IF(NCARD.NE.400) GO TO 98                                         BLK40440
      DO 10 I=1,18                                                      BLK40450
   10 CHED(I) = DUM(I)                                                  BLK40460
      GO TO 99                                                          BLK40470
  401 IF(NCARD.NE.401.OR.NSEQ.NE.0) GO TO 98                            BLK40480
      TB     = DUM(1)                                                   BLK40490
```

```
      TE     = DUM(2)
      TINCR  = DUM(3)
      NTBL1  = DUM(4)
      IF(NTBL1.EQ.0) GO TO 99
      NT = IFIX((TE-TB)/TINCR + 1.2)
      NCRDS = (NT-1)/9 + 1
      CALL TREAD(NCARD,NCRDS,NT,50,PSIF,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  402 IF(NCARD.NE.402.OR.NSEQ.NE.0) GO TO 98
      BTT = DUM(1)
      ETT = DUM(2)
      DTT = DUM(3)
      NTT1 = DUM(4)
      NTT2 = DUM(5)
      NTT3 = DUM(6)
      IF(NTT1+NTT2+NTT3.EQ.0) GO TO 99
      NTTS = IFIX((ETT-BTT)/DTT+1.2)
      NCRDS = (NTTS-1)/9+1
      IF(NTT1.EQ.0) GO TO 21
      NPAGE(18) = 1
      CALL TREAD(NCARD,NCRDS,NTTS,101,TPC,NERR)
      IF(NERR.NE.0) GO TO 98
   21 IF(NTT2.EQ.0) GO TO 22
      NPAGE(18) = 1
      CALL TREAD(NCARD,NCRDS,NTTS,101,TTS,NERR)
      IF(NERR.NE.0) GO TO 98
   22 IF(NTT3.EQ.0) GO TO 99
      CALL TREAD(NCARD,NCRDS,NTTS,101,TTR,NERR)
      IF(NERR.NE.0) GO TO 98
      GO TO 99
  403 IF(NCARD.NE.403) GO TO 98
      EMDT = DUM(1)
      EN = DUM(2)
      NEN = IFIX(EN)
      DS = DUM(3)
      TAUF = DUM(4)
      TIL = DUM(5)
      TL = DUM(6)
      TSTS10 = DUM(7)
      TSTS20 = DUM(8)
      TESTBO = DUM(9)
      GO TO 99
  404 IF(NCARD.NE.404) GO TO 98
      TSTR10 = DUM(1)
      TSTR20 = DUM(2)
      APDMAX = DUM(3)
      FKDO = DUM(4)
      FKS10 = DUM(5)
      FKS20 = DUM(6)
      FKSKDO = DUM(7)
```

```
      BFP1 = DUM(8)                                              BLK41010
      BFP2 = DUM(9)                                              BLK41020
      GO TO 99                                                   BLK41030
  405 IF(NCARD.NE.405) GO TO 98                                  BLK41040
      GEAR1 = DUM(1)                                             BLK41050
      GEAR2 = DUM(2)                                             BLK41060
      GEAR3 = DUM(3)                                             BLK41070
      GEAR4 = DUM(4)                                             BLK41080
      GO TO 99                                                   BLK41090
  406 IF(NCARD.NE.406) GO TO 98                                  BLK41100
      VGR12 = DUM(1)                                             BLK41110
      VGR23 = DUM(2)                                             BLK41120
      VGR34 = DUM(3)                                             BLK41130
      VGR43 = DUM(4)                                             BLK41140
      VGR32 = DUM(5)                                             BLK41150
      VGR21 = DUM(6)                                             BLK41160
      IF(VGR34.LT.VGR23) VGR34 = 10000.0                         BLK41170
      GO TO 99                                                   BLK41180
  407 IF(NCARD.NE.407) GO TO 98                                  BLK41190
      DO 4071 I=1,7                                              BLK41200
 4071 XIMPOR(I) = DUM(I)                                         BLK41210
      GO TO 99                                                   BLK41220
  408 IF(NCARD.NE.408) GO TO 98                                  BLK41230
      DO 4081 I=1,5                                              BLK41240
 4081 TESTT(I) = DUM(I)                                          BLK41250
      GO TO 99                                                   BLK41260
  409 IF(NCARD.NE.409) GO TO 98                                  BLK41270
      DO 4091 I=1,5                                              BLK41280
 4091 DESSI(I) = DUM(I)                                          BLK41290
      GO TO 99                                                   BLK41300
  410 IF(NCARD.NE.410) GO TO 98                                  BLK41310
      DO 4101 I=1,5                                              BLK41320
 4101 DISTI(I) = DUM(I)                                          BLK41330
      GO TO 99                                                   BLK41340
  411 IF(NCARD.NE.411.OR.NSEQ.NE.0) GO TO 98                     BLK41350
      NTRAN = IFIX(DUM(1))                                       BLK41360
      CALL TREAD(NCARD,1,NTRAN,6,YTRANS,NERR)                    BLK41370
      IF(NERR.NE.0) GO TO 98                                     BLK41380
      CALL T2READ(NCARD,5,NTRAN,2,S,NERR)                        BLK41390
      IF(NERR.NE.0) GO TO 98                                     BLK41400
      GO TO 99                                                   BLK41410
   98 NERR = 1                                                   BLK41420
   99 RETURN                                                     BLK41430
      END                                                        BLK41440
```

```
      SUBROUTINE BLK05(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                  BLK50010
C             HVOSM-VD2 VERSION                                         BLK50020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                BLK50030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        BLK50040
     1             NPAGE(20)                                            BLK50050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    BLK50060
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     BLK50070
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUP,                     BLK50080
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,BLK50090
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, BLK50100
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    BLK50110
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  BLK50120
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  BLK50130
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   BLK50140
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 BLK50150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),BLK50160
     1           XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN BLK50170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          BLK50180
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  BLK50190
     2             PSIFIO,PSIFDO                                        BLK50200
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          BLK50210
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                          BLK50220
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                    BLK50230
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                BLK50240
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                BLK50250
     5               YCMP(6),ZCMP(6),PHICM(6)                           BLK50260
      DIMENSION YCIP(2)                                                 BLK50270
      EQUIVALENCE (YCIP(1),YC1P)                                        BLK50280
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF                              BLK50290
      DIMENSION DUM(18)                                                 BLK50300
      DATA NBS/10/                                                      BLK50310
      NBT = NBCRD+1                                                     BLK50320
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98                             BLK50330
      GO TO (500,501,502,503,504,505,506,507,508,509,510),NBT           BLK50340
      GO TO 98                                                          BLK50350
  500 IF(NCARD.NE.500) GO TO 98                                         BLK50360
      DO 10 I=1,18                                                      BLK50370
   10 GHED(I) = DUM(I)                                                  BLK50380
      GO TO 99                                                          BLK50390
  501 IF(NCARD.NE.501) GO TO 98                                         BLK50400
      IF(NZTAB.LT.1) NZTAB=1                                            BLK50410
      I = 1                                                             BLK50420
      GO TO 20                                                          BLK50430
  502 IF(NCARD.NE.502) GO TO 98                                         BLK50440
      IF(NZTAB.LT.2) NZTAB = 2                                          BLK50450
      I = 2                                                             BLK50460
      GO TO 20                                                          BLK50470
  503 IF(NCARD.NE.503) GO TO 98                                         BLK50480
      IF(NZTAB.LT.3) NZTAB = 3                                          BLK50490
```

```
      I = 3                                                        BLK50500
      GO TO 20                                                     BLK50510
  504 IF(NCARD.NE.504) GO TO 98                                    BLK50520
      IF(NZTAB.LT.4) NZTAB = 4                                     BLK50530
      I = 4                                                        BLK50540
      GO TO 20                                                     BLK50550
  505 IF(NCARD.NE.505) GO TO 98                                    BLK50560
      NZTAB = 5                                                    BLK50570
      I = 5                                                        BLK50580
   20 NPAGE(15) = 1                                                BLK50590
      XB(I)    = DUM(1)                                            BLK50600
      XE(I)    = DUM(2)                                            BLK50610
      XINCR(I) = DUM(3)                                            BLK50620
      YB(I)    = DUM(4)                                            BLK50630
      YE(I)    = DUM(5)                                            BLK50640
      YINCR(I) = DUM(6)                                            BLK50650
      NBX(I)   = DUM(7)                                            BLK50660
      NBY(I)   = DUM(8)                                            BLK50670
      NZ5T     = DUM(9)                                            BLK50680
      NNBX = NBX(I)                                                BLK50690
      NNBY = NBY(I)                                                BLK50700
      IF(NZ5T.EQ.1) GO TO 21                                       BLK50710
      NNX = (XE(I)-XB(I))/XINCR(I)+1                               BLK50720
      NNY = (YE(I)-YB(I))/YINCR(I)+1                               BLK50730
      NX(I) = NNX                                                  BLK50740
      NY(I) = NNY                                                  BLK50750
      CALL TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)                   BLK50760
      IF(NERR.NE.0) GO TO 98                                       BLK50770
      GO TO 99                                                     BLK50780
   21 NNX = DUM(3)                                                 BLK50790
      NNY = DUM(6)                                                 BLK50800
      NX(I) = NNX                                                  BLK50810
      NY(I) = NNY                                                  BLK50820
      NZ5 = 1                                                      BLK50830
      CALL TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)                   BLK50840
      IF(NERR.NE.0) GO TO 98                                       BLK50850
      GO TO 99                                                     BLK50860
  506 IF(NCARD.NE.506) GO TO 98                                    BLK50870
      DO 30 J=1,5                                                  BLK50880
   30 AMUG(J) = DUM(J)                                             BLK50890
      GO TO 99                                                     BLK50900
  507 IF(NCARD.NE.507) GO TO 98                                    BLK50910
      YC1P = DUM(1)                                                BLK50920
      YC2P = DUM(2)                                                BLK50930
      YC3P = DUM(3)                                                BLK50940
      YC4P = DUM(4)                                                BLK50950
      YC5P = DUM(5)                                                BLK50960
      YC6P = DUM(6)                                                BLK50970
      AMUC = DUM(7)                                                BLK50980
      GO TO 99                                                     BLK50990
  508 IF(NCARD.NE.508) GO TO 98                                    BLK51000
```

```
      ZC2P = DUM(1)                                                    BLK51010
      ZC3P = DUM(2)                                                    BLK51020
      ZC4P = DUM(3)                                                    BLK51030
      ZC5P = DUM(4)                                                    BLK51040
      ZC6P = DUM(5)                                                    BLK51050
      GO TO 99                                                         BLK51060
  509 IF(NCARD.NE.509) GO TO 98                                        BLK51070
      PHIC1 = DUM(1)                                                   BLK51080
      PHIC2 = DUM(2)                                                   BLK51090
      PHIC3 = DUM(3)                                                   BLK51100
      PHIC4 = DUM(4)                                                   BLK51110
      PHIC5 = DUM(5)                                                   BLK51120
      PHIC6 = DUM(6)                                                   BLK51130
      GO TO 99                                                         BLK51140
  510 IF(NCARD.NE.510) GO TO 98                                        BLK51150
      DELG = DUM(1)                                                    BLK51160
      NEND = IFIX(DUM(2))                                              BLK51170
      IRUF = 1                                                         BLK51180
      NPAGE(8) = 1                                                     BLK51190
      DGMAX = (NEND-1)*DELG                                            BLK51200
      GO TO 99                                                         BLK51210
   98 NERR = 1                                                         BLK51220
   99 RETURN                                                           BLK51230
      END                                                              BLK51240
```

```
      SUBROUTINE BLK06(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)
C               HVOSM-VD2 VERSION -
C               REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),
     2              TRPME(12),TWCT(12),TCT(12),TT(101),TPC(101),TTR(101)
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      DIMENSION DUM(18)
      DATA NBS/4/
      NBT = NBCRD+1
      IF(NBT.LT.1.OR.NBT.GT.NBS+1) GO TO 98
      GO TO(600,601,602,603,604),NBT
      GO TO 98
  600 IF(NCARD.NE.600) GO TO 98
      DO 10 I=1,18
   10 SHED(I) = DUM(I)
      GO TO 99
  601 IF(NCARD.NE.601) GO TO 98
      PHIO   = DUM(1)
      THETAO = DUM(2)
      PSIO   = DUM(3)
      PO     = DUM(4)
      QO     = DUM(5)
      RO = DUM(6)
      PSIFIO = DUM(7)
      PSIFDO = DUM(8)
```

```
      GO TO 99                                                          BLK60500
  602 IF(NCARD.NE.602) GO TO 98                                         BLK60510
      XCOP = DUM(1)                                                     BLK60520
      YCOP = DUM(2)                                                     BLK60530
      ZCOP = DUM(3)                                                     BLK60540
      UO   = DUM(4)                                                     BLK60550
      VO   = DUM(5)                                                     BLK60560
      WO   = DUM(6)                                                     BLK60570
      GO TO 99                                                          BLK60580
  603 IF(NCARD.NE.603) GO TO 98                                         BLK60590
      DEL10  = DUM(1)                                                   BLK60600
      DEL20 = DUM(2)                                                    BLK60610
      IF(ISUS.EQ.2) PHIFO = DUM(2)                                      BLK60620
      DEL30  = DUM(3)                                                   BLK60630
      PHIRO = DUM(4)                                                    BLK60640
      IF(ISUS.EQ.1) DEL40 = DUM(4)                                      BLK60650
      DEL10D = DUM(5)                                                   BLK60660
      DEL20D = DUM(6)                                                   BLK60670
      IF(ISUS.EQ.2) PHIFOD = DUM(6)                                     BLK60680
      DEL30D = DUM(7)                                                   BLK60690
      PHIROD = DUM(8)                                                   BLK60700
      IF(ISUS.EQ.1) DEL40D = DUM(8)                                     BLK60710
      GO TO 99                                                          BLK60720
  604 IF(NCARD.NE.604) GO TO 98                                         BLK60730
      TAUA = DUM(1)                                                     BLK60740
      TAUO(1) = DUM(2)                                                  BLK60750
      TAUO(2) = DUM(3)                                                  BLK60760
      TAUO(3) = DUM(4)                                                  BLK60770
      TAUO(4) = DUM(5)                                                  BLK60780
      GO TO 99                                                          BLK60790
   98 NERR = 1                                                          BLK60800
   99 RETURN                                                            BLK60810
      END                                                               BLK60820
```

```
      SUBROUTINE CLEAR(A,B)
C          CLEARS (SETS TO ZERO) A BLOCK OF STORAGE IDENTIFIED BY THE
C           ADDRESSES OF THE TWO ARGUMENTS.
C
C          CALL CLEAR(P,Q)
C          WILL CAUSE ALL BYTES TO BE SET TO ZERO FROM ADDRESS
C          P THROUGH THE FULL-WORD AT ADDRESS Q
C
      DIMENSION A(1),B(1)
      B(1) = 1.0
      I=0
   10 IF(B(1).EQ.0.0)  RETURN
      I=I+1
      A(I) = 0.0
      GO TO 10
      END
```

```
      SUBROUTINE CNSTNT                                                 CNST0010
C           HVOSM-VD2 VERSION                                           CNST0020
C           REVISED OCTOBER 1975  CALSPAN CORPORATION                   CNST0030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        CNST0040
     1            NPAGE(20)                                             CNST0050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    CNST0060
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      CNST0070
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      CNST0080
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,CNST0090
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  CNST0100
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     CNST0110
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   CNST0120
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   CNST0130
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    CNST0140
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  CNST0150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),CNST0160
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRG(4,5),UVWMIN,PQRMIN  CNST0170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          CNST0180
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,   CNST0190
     2            PSIF10,PSIFD0                                         CNST0200
      DIMENSION YC1P(2)                                                 CNST0210
      EQUIVALENCE (YCIP(1),YC1P)                                        CNST0220
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             CNST0230
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))CNST0240
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),CNST0250
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        CNST0260
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       CNST0270
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         CNST0280
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           CNST0290
     6           (PSIFID,VAR(22))                                       CNST0300
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      CNST0310
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8)))CNST0320
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       CNST0330
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     CNST0340
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    CNST0350
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          CNST0360
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      CNST0370
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         CNST0380
     1           (DER(10),DPHIFD)                                       CNST0390
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       CNST0400
     1           (DER(14),DDEL4D)                                       CNST0410
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,CNST0420
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          CNST0430
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  CNST0440
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    CNST0450
     4            XIZR,RTR,RHMR21,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    CNST0460
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPCNST0470
     6           ,TAN1P,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     CNST0480
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    CNST0490
```

```
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, CNST0500
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  CNST0510
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3, CNST0520
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, CNST0530
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,CNST0540
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2CNST0550
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,CNST0560
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          CNST0570
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                CNST0580
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)      CNST0590
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                    CNST0600
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   CNST0610
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   CNST0620
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       CNST0630
      LOGICAL LCB1,LCB2                                                CNST0640
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),  CNST0650
     1              A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     CNST0660
     2              A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIS(4)          CNST0670
C       COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST CNST0680
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,   CNST0690
     1              COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,  CNST0700
     2              STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN   CNST0710
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),  CNST0720
     A              XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),             CNST0730
     B              XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),         CNST0740
     C              XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4            CNST0750
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)      CNST0760
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)      CNST0770
      COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),   CNST0780
     1            TIH1(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,CNST0790
     2      DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),   CNST0800
     3            SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),  CNST0810
     4            EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),      CNST0820
     5            FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),   CNST0830
     6            SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),      CNST0840
     7            FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),      CNST0850
     8        ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),CNST0860
     9        RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD    CNST0870
      COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP CNST0880
      LOGICAL  IUVS,IUVB,IRPS                                          CNST0890
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, CNST0900
     1            CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),      CNST0910
     2            TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)CNST0920
     3           ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,         CNST0930
     4            BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)            CNST0940
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,CNST0950
     1            NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,  CNST0960
     2            RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,     CNST0970
     3            DELTAE,PIO15R,COMEN5,TQE,RPME                        CNST0980
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                    CNST0990
      DIMENSION RPSI(4),DRPSI(4)                                       CNST1000
```

```
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                   CNST1010
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,       CNST1020
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),CNST1030
     2               NCAMF,NCAMR,NDTHF,NDTHR                            CNST1040
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,      CNST1050
     1                RHF2MF,RF2MF1,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     CNST1060
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    CNST1070
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    CNST1080
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        CNST1090
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     CNST1100
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,                          CNST1110
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,                          CNST1120
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,                    CNST1130
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,                CNST1140
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,                CNST1150
     5               YCMP(6),ZCMP(6),PHICM(6)                           CNST1160
      NPAGE(1) = 1                                                      CNST1170
      NPAGE(2) = 1                                                      CNST1180
      NPAGE(3) = 1                                                      CNST1190
      NPAGE(11) = 1                                                     CNST1200
      NPAGE(12) = 1                                                     CNST1210
      NPAGE(13) = 1                                                     CNST1220
      NPAGE(17) = 1                                                     CNST1230
      PI = 3.14159265300                                                CNST1240
      TWOPI =  2.0*PI                                                   CNST1250
      PIO2  =  0.5 * PI                                                 CNST1260
      PIO4  =  0.25* PI                                                 CNST1270
      TWOPIR = 1.0/TWOPI                                                CNST1280
      PIO15R = 15.0/PI                                                  CNST1290
      RAD =   .01745329250O                                             CNST1300
      DO 7 I=1,4                                                        CNST1310
      A12(I) = A1(I)/A2(I)                                              CNST1320
      A23(I) = A2(I)*A3(I)/A1(I)                                        CNST1330
      A234(I) = A2(I)*A3(I)/A4(I)                                       CNST1340
      OMT2M1(I) = OMEGT(I)*A1(I)*A2(I)*(OMEGT(I)-1.0)                   CNST1350
      OMT2A2(I) = (OMEGT(I)*A2(I)*A3(I)*(A4(I)-OMEGT(I)*A2(I)))         CNST1360
     1          /(A4(I)*(OMT2M1(I)-A0(I)))                              CNST1370
    7 CONTINUE                                                          CNST1380
      TRO2 = 0.5*TR                                                     CNST1390
      TFO2 = 0.5*TF                                                     CNST1400
      AMUF = A*XMUF                                                     CNST1410
      BMUR = B*XMUR                                                     CNST1420
      XMUFO2 = 0.5*XMUF                                                 CNST1430
      AXMFO2 = A*XMUFO2                                                 CNST1440
      XMTFO4 = XMUFO2*TFO2                                              CNST1450
      TM4 = 0.25*XMUF*TF                                                CNST1460
      GMSTMP = 0.5*XMS*G/(A+B)                                          CNST1470
      AO2APB = A*GMSTMP                                                 CNST1480
      BO2APB = B*GMSTMP                                                 CNST1490
      GAM1 = AMUF-BMUR                                                  CNST1500
      SUMM = XMS+XMUF+XMUR                                              CNST1510
```

```
      DEL1 = DEL10                                          CNST1520
      DEL1D = DEL10D                                        CNST1530
      DEL3 = DEL30                                          CNST1540
      DEL3D = DEL30D                                        CNST1550
      IF(ISUS.EQ.1) GO TO 10                                CNST1560
      ZRO = ZR+RHO                                          CNST1570
      TSO2 = 0.5*TS                                         CNST1580
      RHO2 = RHO*RHO                                        CNST1590
      RHOMUR = RHO*XMUR                                     CNST1600
      RHMR2 = RHO*RHOMUR                                    CNST1610
      RTR = RR/TS                                           CNST1620
      BROMUR = RHOMUR*B                                     CNST1630
      RHMR2I = RHMR2+XIR                                    CNST1640
      PHIR = PHIRO*RAD                                      CNST1650
      PHIRD = PHIROD*RAD                                    CNST1660
   10 IF(ISUS.NE.0) GO TO 20                                CNST1670
      ZPR = ZF+RHO                                          CNST1680
      RRTS = RR*TS                                          CNST1690
      TIZ = XMUF*(A*A+TFO2*TFO2)+BMUR                       CNST1700
      XIZR = XIZ+XIR                                        CNST1710
   20 IF(ISUS.EQ.2) GO TO 30                                CNST1720
      RFTF = RF/(TF*TF)                                     CNST1730
      DEL2 = DEL20                                          CNST1740
      DEL2D = DEL20D                                        CNST1750
   30 IF(ISUS.NE.2) GO TO 40                                CNST1760
      ZFO = ZF+RHOF                                         CNST1770
      TSFO2 = 0.5*TSF                                       CNST1780
      RHOF2 = RHOF*RHOF                                     CNST1790
      RHFMUF = RHOF*XMUF                                    CNST1800
      RHF2MF = RHOF*RHFMUF                                  CNST1810
      RF2MFI = RHF2MF+XIF                                   CNST1820
      RTF = RF/TSF                                          CNST1830
      PHIF = PHIFO*RAD                                      CNST1840
      PHIFD = PHIFOD*RAD                                    CNST1850
   40 IF(ISUS.NE.1) GO TO 50                                CNST1860
      RRTR = RR/(TR*TR)                                     CNST1870
      XMURO2 = 0.5*XMUR                                     CNST1880
      BXMRO2 = B*XMURO2                                     CNST1890
      XMTRO4 = XMURO2*TRO2                                  CNST1900
      DEL4 = DEL40                                          CNST1910
      DEL4D = DEL40D                                        CNST1920
   50 CONTINUE                                              CNST1930
      U = UO                                                CNST1940
      V = VO                                                CNST1950
      W = WO                                                CNST1960
      P = PO*RAD                                            CNST1970
      Q = QO*RAD                                            CNST1980
      R = RO*RAD                                            CNST1990
      THETTP = 0.0                                          CNST2000
      PHITP = 0.0                                           CNST2010
      PSITP = 0.0                                           CNST2020
```

```
      THETN = THETAO*RAD                                              CNST2030
      PHIN = PHIO*RAD                                                 CNST2040
      PSIN = PSIO*RAD                                                 CNST2050
C     XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTXCNST2060
C           XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING CNST2070
C           THAT IS  THETN OR PHIN NOW .NE. 0.0                       CNST2080
C           USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST     CNST2090
      IF( THETN.NE.0.0 .OR.  PHIN.NE. 0.0) XINDN = 10.0               CNST2100
      THETTL = THETN                                                  CNST2110
      PHITL  = PHIN                                                   CNST2120
      PSITL  = PSIN                                                   CNST2130
      XCP = XCOP                                                      CNST2140
      YCP = YCOP                                                      CNST2150
      ZCP = ZCOP                                                      CNST2160
      PHIC1R = PHIC1*RAD                                              CNST2170
      PHIC2R = PHIC2*RAD                                              CNST2180
      PHIC3R = PHIC3*RAD                                              CNST2190
      PHIC4R = PHIC4*RAD                                              CNST2200
      PHIC5R = PHIC5*RAD                                              CNST2210
      PHIC6R = PHIC6*RAD                                              CNST2220
      TANPC2 = TAN(PHIC2R)                                            CNST2230
      TANPC1 = TAN(PHIC1R)                                            CNST2240
      TANPC3 = TAN(PHIC3R)                                            CNST2250
      TANPC4 = TAN(PHIC4R)                                            CNST2260
      TANPC5 = TAN(PHIC5R)                                            CNST2270
      TANPC6 = TAN(PHIC6R)                                            CNST2280
      NCB = NCRBSL-1                                                  CNST2290
      GO TO (72,73,74,75,76),NCB                                      CNST2300
   72 PHICLR = PHIC2R                                                 CNST2310
      YCLP = YC2P                                                     CNST2320
      ZCLP = ZC2P                                                     CNST2330
      TANPCL = TANPC2                                                 CNST2340
      YC3P = 1.0E+6                                                   CNST2350
      ZC3P = ZC2P+SIGN(1.0,ZC2P)                                      CNST2360
      GO TO 71                                                        CNST2370
   73 PHICLR = PHIC3R                                                 CNST2380
      YCLP = YC3P                                                     CNST2390
      ZCLP = ZC3P                                                     CNST2400
      TANPCL = TANPC3                                                 CNST2410
      YC4P = 1.0E+6                                                   CNST2420
      ZC4P = ZC3P+SIGN(1.0,ZC3P)                                      CNST2430
      GO TO 71                                                        CNST2440
   74 PHICLR = PHIC4R                                                 CNST2450
      YCLP = YC4P                                                     CNST2460
      ZCLP = ZC4P                                                     CNST2470
      TANPCL = TANPC4                                                 CNST2480
      YC5P = 1.0E+6                                                   CNST2490
      ZC5P = ZC4P+SIGN(1.0,ZC4P)                                      CNST2500
      GO TO 71                                                        CNST2510
   75 PHICLR = PHIC5R                                                 CNST2520
      YCLP = YC5P                                                     CNST2530
```

```
      ZCLP = ZC5P                                                       CNST2540
      TANPCL = TANPC5                                                   CNST2550
      YC6P = 1.0E+6                                                     CNST2560
      ZC6P = ZC5P+SIGN(1.0,ZC5P)                                        CNST2570
      GO TO 71                                                          CNST2580
   76 PHICLR = PHIC6R                                                   CNST2590
      YCLP = YC6P                                                       CNST2600
      ZCLP = ZC6P                                                       CNST2610
      TANPCL = TANPC6                                                   CNST2620
   71 CONTINUE                                                          CNST2630
      PSIFI  = PSIFIO*RAD                                               CNST2640
      PSIFID = PSIFDO*RAD                                               CNST2650
      DO 9 I=1,5                                                        CNST2660
      DO 9 J=1,4                                                        CNST2670
    9 PSBDRY(J,I) = PSBDRO(J,I) * RAD                                   CNST2680
C      NOTE,FIDJ(1)=0 FOR REAR WHEEL DRIVE,  FIDJ(2)=0 FOR FRONT DRIVE  CNST2690
C             FOR WHEEL ROTATION EQUATIONS  FRONT J=1 , REAR J=2        CNST2700
      DO  12  J=1,2                                                     CNST2710
      TBRAKB   =  0.25 * FIDJ(J)* ARBR(J) * ARBR(J)                     CNST2720
      TBRAKA = FIWJ(J) + TBRAKB                                         CNST2730
      TBRAKD = 1.0 /(TBRAKA*TBRAKA - TBRAKB*TBRAKB)                     CNST2740
      ARFAC1(J) = 6.0 * ARBR(J) *(TBRAKA-TBRAKB) * TBRAKD               CNST2750
      RPSFA(J) = TBRAKA * TBRAKD                                        CNST2760
      RPSFB(J) = TBRAKB * TBRAKD                                        CNST2770
      ARFAC2(J) = 6.0 * ARBR(J) / TBRAKA                                CNST2780
      RPSFC(J) = 1.0 / TBRAKA                                           CNST2790
      RPSFD(J) = TBRAKB / TBRAKA                                        CNST2800
       TLAMB(J) = RPSFD(J)                                              CNST2810
      ARFAC3(J) = (RPSFA(J) - RPSFB(J)) * 12.0                          CNST2820
      RPSFE(J) = 12.0 * RPSFC(J)                                        CNST2830
   12 CONTINUE                                                          CNST2840
      DO 13 I=1,4                                                       CNST2850
      J=1                                                               CNST2860
      IF(I.GE.3) J=2                                                    CNST2870
      ARBRI(I) = ARBR(J)                                                CNST2880
   13 CONTINUE                                                          CNST2890
C         RWDRIV = 1.0 FOR REAR WHEEL DRIVE, =0.0 FOR FRONT WHEEL DRIVE CNST2900
      RWDRIV = 1.0                                                      CNST2910
      IF(FIDJ(2) .EQ.0.0) RWDRIV = 0.0                                  CNST2920
C     JDEND SIGNIFIES 'DRIVE' END OF VEHICLE, =1 FOR FRONT, =2 FOR REAR CNST2930
      JDEND = 1                                                         CNST2940
       IF(FIDJ(1).EQ.0.0) JDEND = 2                                     CNST2950
      DO 23 I=1,4                                                       CNST2960
   23 TAU(I) = TAUO(I)                                                  CNST2970
  130 DO 132 J=1,NTTS                                                   CNST2980
  132 TT(J) = FLOAT(J-1)*DTT+BTT                                        CNST2990
      DO 134 J=1,NTLF                                                   CNST3000
  134 TTAU(J) = FLOAT(J-1)*DTLF+BTLF                                    CNST3010
      DRPM = (ERPM-BRPM)/FLOAT(NRPM-1)                                  CNST3020
      DO 136 J=1,NRPM                                                   CNST3030
  136 TRPME(J) = FLOAT(J-1)*DRPM+BRPM                                   CNST3040
```

```
RETURN                                                    CNST3050
END                                                       CNST3060
```

```
      SUBROUTINE CRBIMP(I)                                              CRBI0010
C            HVOSM-VD2 VERSION                                          CRBI0020
C            REVISED OCTOBER 1975   CALSPAN CORPORATION                 CRBI0030
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          CRBI0040
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,   CRBI0050
     2            PSIFIO,PSIFDO                                         CRBI0060
      DIMENSION YCIP(2)                                                 CRBI0070
      EQUIVALENCE (YCIP(1),YC1P)                                        CRBI0080
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             CRBI0090
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))CRBI0100
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),CRBI0110
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        CRBI0120
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       CRBI0130
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         CRBI0140
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           CRBI0150
     6           (PSIFID,VAR(22))                                       CRBI0160
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      CRBI0170
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8)))CRBI0180
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       CRBI0190
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     CRBI0200
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    CRBI0210
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          CRBI0220
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      CRBI0230
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         CRBI0240
     1           (DER(10),DPHIFD)                                       CRBI0250
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       CRBI0260
     1           (DER(14),DDEL4D)                                       CRBI0270
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,CRBI0280
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   CRBI0290
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),CRBI0300
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     CRBI0310
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  CRBI0320
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  CRBI0330
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     CRBI0340
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      CRBI0350
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),CRBI0360
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)CRBI0370
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      CRBI0380
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     CRBI0390
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   CRBI0400
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  CRBI0410
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       CRBI0420
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   CRBI0430
     1           (PSII(1),PSI1)                                         CRBI0440
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,CRBI0450
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          CRBI0460
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  CRBI0470
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    CRBI0480
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    CRBI0490
```

```
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1              A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2              A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,
     1              ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,
     5              YCMP(6),ZCMP(6),PHICM(6)
      DIMENSION FJPP(35)
      DO 20 N=1,35
   20 FJPP(N) = FJP(N,I)
    1 SNPSI = SIN(PSII(I))
      CSPSI = COS(PSII(I))
      SNPHI = SIN(PHII(I))
      CSPHI = COS(PHII(I))
      SFRX(I) = 0.0
      SFRY(I) = 0.0
      SFRZ(I) = 0.0
      TTAJ21 = CSPHI * SNPSI
      TTAJ31 = SNPHI * SNPSI
      AJMTX(1,2) = -SNPSI
      AJMTX(2,2) = CSPHI * CSPSI
      AJMTX(3,2) = SNPHI * CSPSI
      XJ = -26.0*RAD
    2 DO 11  J=1,53
      THTJ = 4.0*XJ
      STJ = SIN(THTJ)
      CTJ = COS(THTJ)
      AJMTX(1,1) = CTJ*CSPSI
      AJMTX(2,1) = TTAJ21*CTJ + SNPHI*STJ
      AJMTX(3,1) = TTAJ31*CTJ - CSPHI*STJ
      AJMTX(1,3) = CSPHI*STJ
```

```
      AJMTX(2,3) = TTAJ21*STJ - SNPHI*CTJ
      AJMTX(3,3) = TTAJ31*STJ + CSPHI*CTJ
C       AJMTX ANGLE SEQUENCE IS PHI,PSI,THJ
    3 DO 8  K=1,3
    4 DO 7  L=1,3
      BMTX(K,L) = 0.0
    5 DO 6  M=1,3
      BMTX(K,L) = BMTX(K,L)+AMTX(K,M)*AJMTX(M,L)
    6 CONTINUE
    7 CONTINUE
    8 CONTINUE
      HJ = -ZP(I)/BMTX(3,3)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 800
      YJP = YP(I)+BMTX(2,3)*HJ
      IF(YJP.LT.YC1P) GO TO 203
  800 HJ = (-ZP(I)+(YP(I)-YC1P)*TANPC1)/(BMTX(3,3)-BMTX(2,3)*TANPC1)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 805
      YJP = YP(I)+BMTX(2,3)*HJ
      ZJP = ZP(I)+BMTX(3,3)*HJ
      IF(YJP.GE.YC1P.AND.YJP.LE.YC2P.AND.(ABS(ZJP).LE.ABS(ZC2P)).AND.
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC2P))) GO TO 204
  805 HJ = (ZC2P-ZP(I)+(YP(I)-YC2P)*TANPC2)/(BMTX(3,3)-BMTX(2,3)*
     1      TANPC2)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 810
      YJP = YP(I)+BMTX(2,3)*HJ
      ZJP = ZP(I)+BMTX(3,3)*HJ
      IF(YJP.GT.YC2P.AND.YJP.LE.YC3P.AND.(ABS(ZJP).LE.ABS(ZC3P)).AND.
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC3P))) GO TO 204
  810 IF(NCRBSL.EQ.2) GO TO 10
      HJ = (ZC3P-ZP(I)+(YP(I)-YC3P)*TANPC3)/(BMTX(3,3)-BMTX(2,3)*TANPC3)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 815
      YJP = YP(I)+BMTX(2,3)*HJ
      ZJP = ZP(I)+BMTX(3,3)*HJ
      IF(YJP.GT.YC3P.AND.YJP.LE.YC4P.AND.(ABS(ZJP).LE.ABS(ZC4P)).AND.
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC4P))) GO TO 204
  815 IF(NCRBSL.EQ.3) GO TO 10
      HJ= (ZC4P-ZP(I)+(YP(I)-YC4P)*TANPC4)/(BMTX(3,3)-BMTX(2,3)*TANPC4)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 820
      YJP = YP(I)+BMTX(2,3)*HJ
      ZJP = ZP(I)+BMTX(3,3)*HJ
      IF(YJP.GT.YC4P.AND.YJP.LE.YC5P.AND.(ABS(ZJP).LE.ABS(ZC5P)).AND.
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC5P))) GO TO 204
  820 IF(NCRBSL.EQ.4) GO TO 10
      HJ= (ZC5P-ZP(I)+(YP(I)-YC5P)*TANPC5)/(BMTX(3,3)-BMTX(2,3)*TANPC5)
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 825
      YJP = YP(I)+BMTX(2,3)*HJ
      ZJP = ZP(I)+BMTX(3,3)*HJ
      IF(YJP.GT.YC5P.AND.YJP.LE.YC6P.AND.(ABS(ZJP).LE.ABS(ZC6P)).AND.
     1  (SIGN(1.0,ZJP).EQ.SIGN(1.0,ZC6P))) GO TO 204
  825 IF(NCRBSL.EQ.5) GO TO 10
      HJ= (ZC6P-ZP(I)+(YP(I)-YC6P)*TANPC6)/(BMTX(3,3)-BMTX(2,3)*TANPC6)
```

```
      IF(HJ.LT.0.0.OR.HJ.GE.RW(I)) GO TO 10                              CRBI1520
      YJP = YP(I)+BMTX(2,3)*HJ                                           CRBI1530
      IF(YJP.LT.YC6P) GO TO 10                                           CRBI1540
  203 ZJP = ZP(I)+BMTX(3,3)*HJ                                           CRBI1550
  204 XJP = XP(I)+BMTX(1,3)*HJ                                           CRBI1560
      CAJ = (XP(I)-XJP)/HJ                                               CRBI1570
      CBJ = (YP(I)-YJP)/HJ                                               CRBI1580
      CGJ = (ZP(I)-ZJP)/HJ                                               CRBI1590
      CALL INTRPL(FJPP,RWHJB,RWHJE,DRWHJ,RW(I)-HJ,FJ)                    CRBI1600
      SFRX(I) = SFRX(I)+FJ*CAJ                                           CRBI1610
      SFRY(I) = SFRY(I)+FJ*CBJ                                           CRBI1620
      SFRZ(I) = SFRZ(I)+FJ*CGJ                                           CRBI1630
   10 XJ = XJ+RAD                                                        CRBI1640
   11 CONTINUE                                                           CRBI1650
      FR(I) = SQRT(SFRX(I)**2+SFRY(I)**2+SFRZ(I)**2)                     CRBI1660
      IF(FR(I).NE.0.0)GO TO 110                                          CRBI1670
      CAR(I) = 0.0                                                       CRBI1680
      CBR(I) = 0.0                                                       CRBI1690
      CGR(I) = 0.0                                                       CRBI1700
      HI(I) = RW(I)                                                      CRBI1710
      RETURN                                                             CRBI1720
  110 CAR(I) = -SFRX(I)/FR(I)                                            CRBI1730
      CBR(I) = -SFRY(I)/FR(I)                                            CRBI1740
      CGR(I) = -SFRZ(I)/FR(I)                                            CRBI1750
      HI(I) = RW(I)-FR(I)/AKT(I)                                         CRBI1760
      IF(HI(I).GT.RW(I)-SIGT(I)) GO TO 111                               CRBI1770
      HI(I) = RW(I)-(FR(I)/AKT(I)+SIGT(I)*(XLAMT(I)-1.0))/XLAMT(I)       CRBI1780
  111 TYGP = YP(I)+HI(I)*CBR(I)                                          CRBI1790
      PHGI(I) = 0.0                                                      CRBI1800
      IF(TYGP.LE.YC1P)GO TO 12                                           CRBI1810
      IF(TYGP.GT.YC1P.AND.TYGP.LE.YC2P) GO TO 900                        CRBI1820
      GO TO 905                                                          CRBI1830
  900 PHGI(I) = PHIC1R                                                   CRBI1840
      GO TO 12                                                           CRBI1850
  905 IF(NCRBSL.EQ.2) GO TO 970                                          CRBI1860
      IF(TYGP.GT.YC2P.AND.TYGP.LE.YC3P) GO TO 910                        CRBI1870
      GO TO 915                                                          CRBI1880
  910 PHGI(I) = PHIC2R                                                   CRBI1890
      GO TO 12                                                           CRBI1900
  915 IF(NCRBSL.EQ.3) GO TO 970                                          CRBI1910
      IF(TYGP.GT.YC3P.AND.TYGP.LE.YC4P) GO TO 920                        CRBI1920
      GO TO 925                                                          CRBI1930
  920 PHGI(I) = PHIC3R                                                   CRBI1940
      GO TO 12                                                           CRBI1950
  925 IF(NCRBSL.EQ.4) GO TO 970                                          CRBI1960
      IF(TYGP.GT.YC4P.AND.TYGP.LE.YC5P) GO TO 930                        CRBI1970
      GO TO 935                                                          CRBI1980
  930 PHGI(I) = PHIC4R                                                   CRBI1990
      GO TO 12                                                           CRBI2000
  935 IF(NCRBSL.EQ.5) GO TO 970                                          CRBI2010
      IF(TYGP.GT.YC5P.AND.TYGP.LE.YC6P) GO TO 940                        CRBI2020
```

```
      GO TO 970
  940 PHGI(I) = PHIC5R
      GO TO 12
  970 PHGI(I) = PHICLR
   12 TCI = CAR(I)*CBYW(I)-CBR(I)*CAYW(I)
      TAI = CBR(I)*CGYW(I)-CGR(I)*CBYW(I)
      TBI = CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
      CPG(I) = COS(PHGI(I))
      SPG(I) = SIN(PHGI(I))
      TERM3 = TBI*SPG(I)
      TERM4 = TCI*CPG(I)
      DN1 = TAI * (TERM3 - TERM4)
      DN2 = -TBI*TERM4 - (TAI**2 + TCI**2)*SPG(I)
      DN3 =  (TAI**2 + TBI**2)*CPG(I)  + TCI*TERM3
      TERM5 = SQRT(DN1**2 + DN2**2 + DN3**2)
      SPG(I) = (-DN2/TERM5)
      PHGI(I) = ARSIN(SPG(I))
      THGI(I) = ATAN (DN1/DN3)
      CPG(I) = COS(PHGI(I))
      TERM6 = SQRT(DN1**2 + DN3**2)
      CTG(I) = DN3/TERM6
      STG(I) = DN1/TERM6
C     STORE XGPP(I),YGPP(I) AS WELL AS ZGPP(I) IN CRBIMP FOR PLOTTING
      XGPP(I) = XP(I) + HI(I) * CAR(I)
      YGPP(I) = TYGP
      ZGPP(I) = ZP(I)+HI(I)*CGR(I)
      RETURN
      END
```

```
      SUBROUTINE CTQB
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,
     1              NBTYP,ARFAC5(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,
     2              RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,
     3              DELTAE,PIO15R,COMEN5,TQE,RPME
C
C  THIS SUBROUTINE COMPUTES BRAKE TORQUES AT EACH WHEEL
C
       P1 = PONE
      P2 = PTWO
      PP(1) = PC
      PP(2) = PC
   10 IF((PC.GT.P1).AND.(PC.LE.P2)) PP(2)= P1 + AK1 * (PC - P1)
      IF( PC.GT.P2) PP(2)= P1 + AK1 * (P2-P1) + AK2 * (PC - P2)
C
C  JFR   FRONT WHEELS JFR=1; REAR WHEELS JFR=2
C
      DO 45 I=1,4
      JFR = 1
      IF(I.GT.2) JFR = 2
      NBTYP = IBTYP(JFR)
      PJ = PP(JFR)
      PJZ = PZERO(JFR)
      TQB(I) = 0.0
      IF((PJ-PJZ).LT.0.0) GO TO 45
      TEMP = TAU(I)
      DO 17 L=2,NTLF
       LL = L-1
   15  IF( TEMP.EQ.TTAU(LL)) GO TO 20
      IF((TEMP.GT.TTAU(LL)).AND.(TEMP.LE.TTAU(L))) GO TO 22
   17 CONTINUE
   18 PRINT 19
   19 FORMAT(1H0,3X,'LAST VALUE IN TABLE USED FOR FADE COEFF.(FLF)')
   20 FLF = TLF(LL)
      GO TO 23
C
C  PAR1 - FUNCTION SUBROUTINE TO DO LAGRANGIAN INTERPOLATION
C
   22 FLF = PAR1(NTLF,LL,TEMP,TTAU,TLF)
   23 GO TO (24,30,33,44),NBTYP
   24 GGLF = GN(2,JFR) * GN(3,JFR) * FLF
      OMGGLF = 1.0 - GGLF
```

```
      GX =  (GN(4,JFR)*(1.0+GGLF )+ GN(5,JFR)*OMGGLF)/ OMGGLF**2
      TQB(1) = (1.0/12.0)*(PJ-PJZ)*GN(1,JFR)*GGLF*GX
      GO TO 45
   30 GGLF = GN(2,JFR) * GN(3,JFR) * FLF
      TQB(1) = ((PJ-PJZ)/6.0)*GN(1,JFR)*GN(4,JFR)*(GGLF/(1.0-GGLF))
      GO TO 45
   33 GGLF = GN(8,JFR) * GN(12,JFR) * FLF
      OMGGL = (1.0 -GN(2,JFR) * GN(12,JFR) *FLF) * GN(9,JFR) *GN(10,JFR)
   35 TTTA=(GN(4,JFR)*(PJ-PJZ))/12.0
      TTTB= GN(3,JFR)*FLF
      TTTSZ = TTTB/SQRT(1.0 + TTTB**2)
      TTTSZG = TTTSZ * GN(7,JFR)
      TTTB = 1.0 + ((GN(6,JFR)+TTTSZG)/(GN(1,JFR)-GN(6,JFR)-TTTSZG))
   40 TTTC=TTTSZG +(GN(11,JFR)*GGLF)/OMGGL
      TTTD=GN(1,JFR)*GGLF/OMGGL
      TQB(I) =  TTTA * (TTTB*TTIC - TTTD)
      GO TO 45
   44 TQB(I) =((PJ-PJZ)/6.0)*GN(3,JFR)*GN(4,JFR)*GN(13,JFR) * FLF
   45 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CTQD
C              HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
C
C  THIS SUBROUTINE COMPUTES DRIVE LINE TORQUE AT THE PROPELLER SHAFT AT
C  VEHICLE END J
C
C  J = 1 OR 2 DEPENDING ON WHETHER FRONT OR REAR WHEEL DRIVE
C  TWOT  -  TABLE OF ENGINE TORQUE WIDE OPEN THROTTLE
C  TCT   -  TABLE OF ENGINE TORQUE CLOSED THROTTLE
C  TRPME -  TABLE OF ENGINE REVOLUTIONS
C  NTTS  -  NO. VALUES LISTED IN TABLE OF THROTTLE SETTINGS, MASTER CYL.
C           PRESSURE AND TRANSMISSION RATIO.
C  TTS   -  TABLE OF THROTTLE SETTINGS
C  TTR   -  TABLE OF TRANSMISSION RATIOS
C  TPC   -  TABLE OF MASTER CYLINDER PRESSURES
C  TT    -  TABLE OF TIME IN SECS. CORRES.TO TABULAR VALUES OF PC,TS,TR
C  TSEC  -  TIME IN SECS.
C
C  FIND TTTS,TTTR,PC FROM TABLES AT TIME TSEC
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1            (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1            (DER(14),DDEL4D)
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),
     A               XXFRCP(6),YMUMAT(6,6,4),XMXPMT(6,6,4),
     B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),
     C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,
```

```
     4               BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)          CTQD0500
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,CTQD0510
     1              NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP, CTQD0520
     2              RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,    CTQD0530
     3              DELTAE,PIO15R,COMEN5,TQE,RPME                       CTQD0540
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                     CTQD0550
      DIMENSION RPSI(4),DRPSI(4)                                        CTQD0560
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(I),DRPSI(1))                   CTQD0570
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE, CTQD0580
     1              IDRVER,IBUG                                         CTQD0590
C       IPATHT - STOP FOR DRIVER MODEL                                  CTQD0600
C       IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL                 CTQD0610
C       ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE            CTQD0620
C       ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES      CTQD0630
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,  CTQD0640
     1             TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,  CTQD0650
     2             TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,   CTQD0660
     3             VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,                 CTQD0670
     4             TESTT(5),DESSI(5),DISTI(5),PSIFHO,XIMPOR(9),         CTQD0680
     5             BFP1,BFP2,DRIEND                                     CTQD0690
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,     CTQD0700
     1             TESTR1,TESTR2,THESKD,FBRK,APB,DSCES,                 CTQD0710
     2             TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,           CTQD0720
     3             APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSO2,QAY, CTQD0730
     4             AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,    CTQD0740
     5             TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,   CTQD0750
     6             PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),   CTQD0760
     7             PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),   CTQD0770
     8             DEND                                                 CTQD0780
      DIMENSION VGRU(3),VGRD(3),GEAR(4)                                 CTQD0790
      EQUIVALENCE (VGRU(1),VGR12),(VGRD(1),VGR43),(GEAR(1),GEAR1)       CTQD0800
       TSEC = T                                                         CTQD0810
C     JDEND SIGNIFIES 'DRIVE' END OF VEHICLE, =1 FOR FRONT, =2 FOR REAR CTQD0820
      IF(JDEND.EQ.1) L = 1                                              CTQD0830
      IF(JDEND.EQ.2) L = 3                                              CTQD0840
       TQD(1) = 0.                                                      CTQD0850
       TQD(2) = 0.                                                      CTQD0860
      IF(IDRVER.NE.0) GO TO 50                                          CTQD0870
      IF(NTT1+NTT2+NTT3.EQ.0) GO TO 41                                  CTQD0880
       DO 10 I = 2,NTTS                                                 CTQD0890
      IA=I - 1                                                          CTQD0900
      RATIO = 0.0                                                       CTQD0910
      PC=TPC(IA)                                                        CTQD0920
      TTTS = TTS(IA)                                                    CTQD0930
      TTTR = TTR(IA)                                                    CTQD0940
      IF((TSEC.GT.TT(I-1)).AND.(TSEC.LT.TT(I))) GO TO 14                CTQD0950
      IF(TSEC.GT.TT(NTTS)) GO TO 11                                     CTQD0960
       IF(TSEC.EQ.TT( IA)) GO TO 21                                     CTQD0970
      IF(TSEC.EQ.TT(I)) GO TO 20                                        CTQD0980
   10 CONTINUE                                                          CTQD0990
   11 TTTR = TTR(NTTS)                                                  CTQD1000
```

```
      PRINT 12,TSEC,TTTR
   12 FORMAT(1H0,3X,'TSEC NOT WITHIN RANGE OF TABULAR VALUES TSEC =',
     1 E15.8,'SET TRANSMISSION RATIO,TTTR=',E15.8)
              NTTS IS THE LARGEST INTEGER IN  10.*(T1 + DT)  + 1.
           TO AVOID OUT-OF-RANGE MESSAGES, USE
C             NTTS IS THE LARGEST INTEGER IN  10.*(T1 + DT)  + 2.
       GO TO 20
   14 RATIO = (TSEC-TT(IA))/(TT(IA+1)-TT(IA))
      TTTR = TTR(IA)+RATIO*(TTR(IA+1)-TTR(IA))
C  WHEN RATIO IS .GT. 0.5 SET TTR TO NEXT VALUE IN TABLE
      IF(RATIO.GT.0.5) TTTR = TTR(IA+1)
C
C  PARI - FUNCTION SUBROUTINE TO DO LAGRANGIAN INTERPOLATION
C
      PC = PARI(NTTS,IA,TSEC,TT,TPC)
      IF(PC.LT.0.0) PC = 0.0
      TTTS = PARI(NTTS,IA,TSEC,TT,TTS)
      IF(TTTS.LT.0.0) TTTS = 0.0
       GO TO 21
   20 TTTR = TTR(IA+1)
      TTTS = TTS(IA+1)
      PC = TPC(IA+1)
      GO TO 21
C DRIVER CONTROLS CONVERTED HERE
   50 TTTS = APD/APDMAX
      IF(TTTS.LT.0.0) TTTS = 0.0
      PC = BFP1*FBRK+BFP2*FBRK**2
      IF(PC.LT.0.0) PC = 0.0
      TTTR = GEAR(IGEAR)
      RPME = PIO15R*ARBR(JDEND)*TTTR*(RPSI(L)+RPSI(L+1))
      IF(TQE.LT.0.0) GO TO 55
      IF(RPME.LT.VGRU(IGEAR)) GO TO 57
      IGEAR = IGEAR+1
      TTTR = GEAR(IGEAR)
      GO TO 21
   55 IF(IGEAR.EQ.1) GO TO 57
      IF(RPME.GT.VGRD(5-IGEAR)) GO TO 57
      IGEAR = IGEAR-1
      TTTR = GEAR(IGEAR)
   21 RPME = PIO15R*ARBR(JDEND)*TTTR*(RPSI(L)+RPSI(L+1))
   57 CONTINUE
      IF(RPME.EQ.0.0) GO TO 41
      IF((RPME.GT.0.).AND.(RPME.LT.500.0)) RPME = 500.0
      DO 30 I=2,NRPM
      IA = I - 1
       TEST = ABS(RPME-TRPME(IA))
       IF(TEST.GT.0.05) GO TO 27
   24 TQWOT = TWOT(IA)
      TQCT  = TCT(IA)
      GO TO 37
   27 IF((RPME.GT.TRPME( IA)).AND.(RPME.LT.TRPME(I))) GO TO 34
```

```
30 CONTINUE
   IA = NRPM
    GO TO 24
34 RATIO = (RPME-TRPME(IA))/(TRPME(IA+1)-TRPME(IA))
   TQWOT = TWOT(IA)+RATIO*(TWOT(IA+1)-TWOT(IA))
   TQCT = TCT(IA)+RATIO*(TCT(IA+1)-TCT(IA))
37 TQE = TQCT + TTTS * (TQWOT - TQCT)
   IF(RPME.GT.500.0) GO TO 40
   IF(PC .GT. 10.0) TTTR = 0.0
40 TQD(JDEND) = TQE * TTTR
41 RETURN
   END
```

```
      SUBROUTINE DAUX
C          HVOSM-VD2 VERSION
C          REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1            (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1            (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),        DAUX0500
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),        DAUX0510
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),      DAUX0520
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                     DAUX0530
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                         DAUX0540
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),     DAUX0550
     1           (PSII(1),PSI1)                                           DAUX0560
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,DAUX0570
     1          GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,             DAUX0580
     2           TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,     DAUX0590
     3           BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTF04,       DAUX0600
     4           XIZR,RTR,RHMP2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,       DAUX0610
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPDAUX0620
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,       DAUX0630
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,       DAUX0640
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,     DAUX0650
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ      DAUX0660
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,    DAUX0670
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,     DAUX0680
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,DAUX0690
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2DAUX0700
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,DAUX0710
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL              DAUX0720
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                   DAUX0730
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)         DAUX0740
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                       DAUX0750
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,      DAUX0760
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),      DAUX0770
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)          DAUX0780
      LOGICAL LCB1,LCB2                                                   DAUX0790
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                  DAUX0800
     1             XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,          DAUX0810
     2             SLOPE1,SLOPE2,XTRA(300)                                DAUX0820
      DIMENSION UI(4),VI(4),WI(4)                                         DAUX0830
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                        DAUX0840
      DIMENSION APITCH(4)                                                 DAUX0850
      EQUIVALENCE (APITCH(1),APTCH1)                                      DAUX0860
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),     DAUX0870
     A              XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),                DAUX0880
     B              XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),            DAUX0890
     C              XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4               DAUX0900
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)         DAUX0910
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)         DAUX0920
      COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),      DAUX0930
     1             TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,DAUX0940
     2      DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),      DAUX0950
     3             SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),    DAUX0960
     4             EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),        DAUX0970
     5             FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),     DAUX0980
     6             SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),        DAUX0990
     7             FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),        DAUX1000
```

```
     8          ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),
     9          RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD
      COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP
      LOGICAL  IUVS,IUVB,IRPS
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)
      DIMENSION RPSI(4),DRPSI(4)
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF
      DIMENSION DISP(4),VEL(4)
C
      IF(ISTOP.NE.0) RETURN
      CALL TMCNST
      IS1 = ISUS+1
      D12D22 = DEL1*DEL1 + DEL2*DEL2
      GO TO (10,11,12),IS1
   10 XIXP = XMUF*(ZF*(ZF+D1PD2)+.5*D12D22) + XMUR*ZRD3*ZRD3R
      XIYP = XIXP+RHOMUR*ZRD3R
      XIZP = TIZ+TIZ2
      XIXZP = AMUF*ZFD12 - BMUR*ZRD3
      XIYZP = TM4*D1MD2-RHOMUR*PHIR*ZRD3R
      GAM2 = XMUF*ZFD12+XMUR*ZRD3R
      GAM3 = GAM2-RHOMUR
      GAM4 = XIYZP+RHMR2*PHIR
      GAM5 = TIZ-XMUF*TFO2*TF-TIZ2
      GAM6 = XMUF*DD1P2+2.0*TG61
      GAM7 = XMUF*(ZF*DD1P2+DEL1*DEL1D+DEL2*DEL2D)+2.0*ZRD3*TG61
      GAM8 = 2.0*(TM4*DD1M2-RPR*TG61)
      GAM9 = AMUF*DD1P2 - 2.0*B*TG61
      GO TO 3
   11 XIXP = XMUFO2*(ZFD1*ZFD1+ZFD2*ZFD2) + XMURO2*(ZRD3*ZRD3+ZRD4*ZRD4)
      XIYP = XIXP
      XIZP = XMUF*(A*A+TFO2*TFO2) +XMUR*(B*B+TRO2*TRO2)
      XIXZP = AXMFO2*(ZFD1+ZFD2) -  BXMRO2*(ZRD3+ZRD4)
      XIYZP = XMTFO4*D1MD2 + XMTRO4*D3MD4
      GAM2 = XMUF*ZFD12 + XMUR*ZRD34
      GAM5 = XMUF*(A*A-TFO2*TFO2) + XMUR*(B*B-TRO2*TRO2)
      GAM6 = XMUF*DD1P2 + XMUR*DD3P4
      GAM7 = XMUF*(ZFD1*DEL1D+ZFD2*DEL2D) + XMUR*(ZRD3*DEL3D+ZRD4*DEL4D)
      GAM8 = XMUF*TFO2*DD1M2 + XMUR*TRO2*DD3M4
      GAM9 = AMUF*DD1P2 - BMUR*DD3P4
      GO TO 3
   12 XIXP = XMUF*ZFD1*ZFD1 + RHFMUF*ZFD1 + XMUR*ZRD3*ZRD3 + RHOMUR*ZRD3
      XIYP = XIXP + RHFMUF*ZFD1RF + RHOMUR*ZRD3R
```

```
      XIZP = XMUF*(A*A+RFPF*RFPF) + XMUR*(B*B+RPR*RPR)                  DAUX1520
      XIXZP = AMUF*ZFD1 - BMUR*ZRD3                                     DAUX1530
      XIYZP = -XMUF*RFPF*ZFD1RF - XMUR*RPR*ZRD3R                        DAUX1540
      GAM2 = XMUF*ZFD1RF + XMUR*ZRD3R                                   DAUX1550
      GAM3 = GAM2 - RHFMUF - RHOMUR                                     DAUX1560
      GAM4 = XIYZP + RHF2MF*PHIF + RHMR2*PHIR                           DAUX1570
      GAM5 = XMUF*(A*A-RFPF*RFPF) + XMUR*(B*B-RPR*RPR)                  DAUX1580
      GAM6 = 2.0*WFMF + 2.0*TG61                                        DAUX1590
      GAM7 = 2.0*ZFD1*WFMF + 2.0*ZRD3*TG61                              DAUX1600
      GAM8 = -2.0*RFPF*WFMF - 2.0*RPR*TG61                              DAUX1610
      GAM9 = 2.0*A*WFMF - 2.0*B*TG61                                    DAUX1620
    3 CALL VPOS                                                         DAUX1630
      CALL VGORNT                                                       DAUX1640
      CALL CTQD                                                         DAUX1650
      CALL CTQB                                                         DAUX1660
      CALL TIRFR                                                        DAUX1670
      IF(ISTOP.NE.0) RETURN                                             DAUX1680
      IF(ISUS.EQ.2) GO TO 20                                            DAUX1690
      DISP(1) = DEL1                                                    DAUX1700
      DISP(2) = DEL2                                                    DAUX1710
      VEL(1) = DEL1D                                                    DAUX1720
      VEL(2) = DEL2D                                                    DAUX1730
      GO TO 21                                                          DAUX1740
   20 DISP(1) = DEL1+TSFO2*PHIF                                         DAUX1750
      DISP(2) = DEL1-TSFO2*PHIF                                         DAUX1760
      VEL(1) = DEL1D+TSFO2*PHIFD                                        DAUX1770
      VEL(2) = DEL1D-TSFO2*PHIFD                                        DAUX1780
      GO TO 22                                                          DAUX1790
   21 IF(ISUS.NE.1) GO TO 22                                            DAUX1800
      DISP(3) = DEL3                                                    DAUX1810
      DISP(4) = DEL4                                                    DAUX1820
      VEL(3) = DEL3D                                                    DAUX1830
      VEL(4) = DEL4D                                                    DAUX1840
      GO TO 23                                                          DAUX1850
   22 DISP(3) = DEL3+TSO2*PHIR                                          DAUX1860
      DISP(4) = DEL3-TSO2*PHIR                                          DAUX1870
      VEL(3) = DEL3D+TSO2*PHIRD                                         DAUX1880
      VEL(4) = DEL3D-TSO2*PHIRD                                         DAUX1890
   23 CALL SUSFRC(DISP,VEL)                                             DAUX1900
      CALL UMOMNT(ISUS)                                                 DAUX1910
      GO TO (30,31,32),IS1                                              DAUX1920
   30 CALL MATRIX                                                       DAUX1930
      GO TO 34                                                          DAUX1940
   31 CALL MTRXIR                                                       DAUX1950
      GO TO 34                                                          DAUX1960
   32 CALL MTRXSF                                                       DAUX1970
   34 CALL SIMSOL(DMATX,10,10)                                          DAUX1980
      DU = DMATX(1,11)                                                  DAUX1990
      DV = DMATX(2,11)                                                  DAUX2000
      DW = DMATX(3,11)                                                  DAUX2010
      DP = DMATX(4,11)                                                  DAUX2020
```

```
      DQ = DMATX(5,11)
      DR = DMATX(6,11)
      DXCP = AMTX(1,1)*U + AMTX(1,2)*V + AMTX(1,3)*W
      DYCP = AMTX(2,1)*U + AMTX(2,2)*V + AMTX(2,3)*W
      DZCP = AMTX(3,1)*U + AMTX(3,2)*V + AMTX(3,3)*W
      DTHTTP = Q*CPHTP - R*SPHTP
      DPHITP = P + (Q*SPHTP + R*CPHTP)*TANTP
      DPSITP = (Q*SPHTP + R*CPHTP)*SECTP
      IF(ISUS.EQ.2) GO TO 40
      DDEL1D = DMATX(7,11)
      DDEL2D = DMATX(8,11)
      DDEL1 = DEL1D
      DDEL2 = DEL2D
      GO TO 41
   40 DDEL1D = DMATX(7,11)
      DPHIFD = DMATX(8,11)
      DDEL1 = DEL1D
      DPHIF = PHIFD
      GO TO 43
   41 IF(ISUS.NE.1) GO TO 43
      DDEL3D = DMATX(9,11)
      DDEL4D = DMATX(10,11)
      DDEL3 = DEL3D
      DDEL4 = DEL4D
      GO TO 44
   43 DDEL3D = DMATX(9,11)
      DPHIRD = DMATX(10,11)
      DDEL3 = DEL3D
      DPHIR = PHIRD
   44 CONTINUE
      IF(IHIT.EQ.0.AND.INDCRB.GE.0) RETURN
      DPSIFI = PSIFID
      T1PSI = 0.0
      T2PSI = 0.0
      IF(ABS(PSIFID).GT.EPSPS) T1PSI = SIGN(CPSP,PSIFID)
      IF(SIGN(1.,PSIFID) .NE. SIGN(1.,PSIFI)) GO TO 7
      ABSPSF = ABS(PSIFI)
      IF(ABSPSF .GT. OMGPS) T2PSI=SIGN((AKPS*(ABSPSF-OMGPS)),PSIFI)
    7 DDPSFI = (FYU(1)*(HCAH1-XPS*COS(PSIIP(1))*CTXG(1))+
     1          FYU(2)*(HCAH2-XPS*COS(PSIIP(2))*CTXG(2))-
     2       FXU(1)*(HCBH1+PHI1*HCGH1)-FXU(2)*(HCBH2+PHI2*HCGH2)-
     3        T1PSI-T2PSI+FZU(1)*HCAH1*PHI1+FZU(2)*HCAH2*PHI2)/XIPS
     4 +((FIWJF*COS(PSIFI))/XIPS) *(RPSI(1)*(P+PHI1D)+RPSI(2)*(P+PHI2D))
C     IN STEER DEGREE OF FREEDOM,GYROSCOPIC PRECESSION OF SPINNING WHEELS
C         SEE CONTINUATION CARD 4 JUST ABOVE
      RETURN
      END
```

```
      SUBROUTINE DAUXR(NTRA)                                            DAXR0010
C              HVOSM-VD2 VERSION                                        DAXR0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               DAXR0030
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,DAXR0040
     1              PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  DAXR0050
     2              CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),DAXR0060
     3              STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    DAXR0070
     4              XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), DAXR0080
     5              YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), DAXR0090
     6              CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    DAXR0100
     7              CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     DAXR0110
     8              SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),DAXR0120
     9              FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)DAXR0130
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      DAXR0140
     1              BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),    DAXR0150
     2              FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),  DAXR0160
     3              F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                 DAXR0170
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       DAXR0180
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   DAXR0190
     1           (PSII(1),PSI1)                                         DAXR0200
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     DAXR0210
     1              DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,     DAXR0220
     2              PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),     DAXR0230
     3              SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)         DAXR0240
      LOGICAL LCB1,LCB2                                                 DAXR0250
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   DAXR0260
     1              A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),       DAXR0270
     2              A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)            DAXR0280
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),   DAXR0290
     A              XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),               DAXR0300
     B              XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),           DAXR0310
     C              XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4              DAXR0320
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)       DAXR0330
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)       DAXR0340
      COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),    DAXR0350
     1              TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,DAXR0360
     2        DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),   DAXR0370
     3              SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),  DAXR0380
     4              EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),      DAXR0390
     5              FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),   DAXR0400
     6              SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),      DAXR0410
     7              FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),      DAXR0420
     8          ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),DAXR0430
     9          RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD   DAXR0440
      COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP DAXR0450
      LOGICAL  IUVS,IUVB,IRPS                                           DAXR0460
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB, DAXR0470
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),      DAXR0480
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)DAXR0490
```

```
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,             DAXR0500
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)                DAXR0510
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,DAXR0520
     1              NETYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,  DAXR0530
     2              RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4).VECS,     DAXR0540
     3              DELTAE,PIO15R,COMEN5,TQE,RPME                        DAXR0550
      COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                      DAXR0560
      DIMENSION RPSI(4),DRPSI(4)                                         DAXR0570
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                    DAXR0580
      DIMENSION XMUX(4)                                                  DAXR0590
C                                                                        DAXR0600
C                                                                        DAXR0610
C         NOTE FRCP REQUIRES FS WHICH IS DETERMINED LATER. THE INITIALIZDAXR0620
C            CALL SHOULD CORRECT THIS BY RECOMPUTING FS.                 DAXR0630
      DO 100 I=1,4                                                       DAXR0640
      IF (FR(I))  41,42,41                                               DAXR0650
   41 FRTEST(I) =  (FR(I) - FS(I) * SPHICI(I))                           DAXR0660
      IF(FRTEST(I)) 42,42,45                                             DAXR0670
   42 FC(I)= 0.0                                                         DAXR0680
      FS(I)= 0.0                                                         DAXR0690
      FRCP(I)= 0.0                                                       DAXR0700
      SLIP(I)    =  0.                                                   DAXR0710
      SLIPT(I) = 0.                                                      DAXR0720
      GO TO 100                                                          DAXR0730
   45 FRCP(I) = FRTEST(I) /CPHICI(I)                                     DAXR0740
       VECS = SQRT(UG(I)**2 + VG(I)**2)                                  DAXR0750
      ABSFRP = ABS(FRCP(I))                                              DAXR0760
      XMRATO = XMUGI(I)/XMUM(I)                                          DAXR0770
      CALL INTPR(XMUMAT,I,XXFRCP,XXUGMU,NXFRCP,NXUGMU,ABSFRP ,VECS,      DAXR0780
     1           XNMY,XT1,6,6)                                           DAXR0790
      CALL INTPR(XMXPMT,I,XXFRCP,XXUGMU,NXFRCP,NXUGMU,ABSFRP ,UGW(I),    DAXR0800
     1           XNMXP,XT1,6,6)                                          DAXR0810
      CALL INTPR(XMXSMT,I,XXFRCP,XXUGMU,NXFRCP,NXUGMU,ABSFRP ,UGW(I),    DAXR0820
     1           XNMXS,XT1,6,6)                                          DAXR0830
      CALL INTPR(SLIPMT,I,XXFRCP,XXUGMU,NXFRCP,NXUGMU,ABSFRP ,UGW(I),    DAXR0840
     1           SLIPP,XT1,6,6)                                          DAXR0850
      XMUI(I) = XNMY*XMRATO                                              DAXR0860
      XMUXP(I) = XNMXP*XMRATO                                            DAXR0870
      XMUXS(I) = XNMXS*XMRATO                                            DAXR0880
      FRCPM  =   FRCP(I) * XMUI(I)                                       DAXR0890
      FRCPMU(I) = FRCPM                                                  DAXR0900
      FC(I)= 0.0                                                         DAXR0910
      IF (IUVS(I)) GO TO 61                                              DAXR0920
      HRTERM  = ABS( HI(I) * RPSI(I))                                    DAXR0930
      IF (HRTERM - 0.5) 49,60,60                                         DAXR0940
   49 SLIP(I) = 0.0                                                      DAXR0950
      GO TO 63                                                           DAXR0960
   60 SLIP(I) = -1.0 * SIGN(1.0,UGW(I))* SIGN(1.0,RPSI(I))               DAXR0970
      GO TO 63                                                           DAXR0980
   61 SLIP(I) = 1.0  - RPSI(I) * SLPFAC(1)                               DAXR0990
   63 SLIPT(I) = ABS (SLIP(I))                                           DAXR1000
```

```
      IF( SLIPT(I) .GT. 1.0) SLIPT(I) = 1.0                            DAXR1010
      IF(SLIPT(I).EQ.1.0) SLIP(I) = SIGN(1.0,SLIP(I))                  DAXR1020
      CCT = CT(I)/FRCP(I)                                              DAXR1030
      IF(SLIPT(I).GE.SLIPP) GO TO 73                                   DAXR1040
      KK = 0                                                           DAXR1050
      RMU = 0.8                                                        DAXR1060
   70 SC2 = (RMU-1.0)*XMUXP(I)/(SLIPP-RMU*XMUXP(I)/CCT)**2             DAXR1070
      SC1 = -2.0*SC2*SLIPP                                             DAXR1080
      SC0 = XMUXP(I)+SC2*SLIPP**2                                      DAXR1090
      IF(SC0.GT.0.1*XMUXP(I)) GOTO 71                                  DAXR1100
      KK = KK+1                                                        DAXR1110
      IF(KK.GT.5) GO TO 999                                            DAXR1120
      RMU = RMU+0.5*(1.0-RMU)                                          DAXR1130
      GO TO 70                                                         DAXR1140
   71 SLIP1 = RMU*XMUXP(I)/CCT                                         DAXR1150
      IF(SLIPT(I).GT.SLIP1) GO TO 72                                   DAXR1160
      XMUX(I) = CCT*SLIPT(I)*SIGN(1.0,SLIP(I))                         DAXR1170
      GO TO 74                                                         DAXR1180
   72 XMUXT = SC0+SC1*SLIPT(I)+SC2*SLIPT(I)**2                         DAXR1190
      XMUX(I) = XMUXT*SIGN(1.0,SLIP(I))                                DAXR1200
      GO TO 74                                                         DAXR1210
   73 SC5 = (XMUXP(I)-XMUXS(I))/(SLIPP-1.0)**2                         DAXR1220
      SC4 = -2.0*SC5                                                   DAXR1230
      SC3 = XMUXS(I)+SC5                                               DAXR1240
      XMUXT = SC3+SC4*SLIPT(I)+SC5*SLIPT(I)**2                         DAXR1250
      XMUX(I) = XMUXT*SIGN(1.0,SLIP(I))                                DAXR1260
   74 CONTINUE                                                         DAXR1270
      RHOS(I) = XMUX(I)/XMUI(I)                                        DAXR1280
      RHOMAX = XMUXP(I)/XMUI(I)                                        DAXR1290
   75 EPSS(I) = 1.0/RHOMAX**2                                          DAXR1300
      IF(SLIPT(I).GT.SLIPP) EPSS(I) = 1.0/RHOS(I)**2                   DAXR1310
   77 FC(I) = -RHOS(I) * FRCPM      * SIGN(1.0,UGW(I))                 DAXR1320
      JFR = 1                                                          DAXR1330
      IF(I.GT.2) JFR = 2                                               DAXR1340
      IF(PP(JFR)-PZERO(JFR)) 81,81,80                                  DAXR1350
   80 FACTOR= (-FRCPM/SQRT(EPSS(I)+TERMP(I)))*SIGN(1.0,UGW(I))*SIGN(1.0,DAXR1360
     X   RHOS(I))                                                      DAXR1370
      IF (ABS(FACTOR) .LT. ABS(FC(I)))  FC(I)=FACTOR                   DAXR1380
C                                                                      DAXR1390
   81 FS(I) = 0.0                                                      DAXR1400
      BETBR(I) = 0.0                                                   DAXR1410
      EPSSFC = EPSS(I) * FC(I)                                         DAXR1420
      IF((EPSSFC*FC(I)).LT.(FRCPM*FRCPM-1.0)) GO TO 86                 DAXR1430
C              BETBR(I) SET TO CAUSE SKID INDICATION ON OUTPUT.        DAXR1440
      BETBR(I) = 3.1                                                   DAXR1450
      GO TO 100                                                        DAXR1460
C          LOGICAL VARIABLE IUVB SET UP IN SUBROUTINE TIRFR FOR SIDE FORDAXR1470
C        IF EITHER ABS(UG(I)).GT.0.5 OR ABS(VG(I)).GT.0.5, IUVB(I) IS TRDAXR1480
   86 IF( .NOT. IUVB(I)) GO TO 100                                     DAXR1490
   87 FS(I) = SQRT(FRCPM*FRCPM  - EPSSFC*FC(I))                        DAXR1500
C       CORRECT FS FOR THE SIGN OF BETBR LATER.                        DAXR1510
```

```
      IF(FRCP(I).GT.OMEGT(I)*A2(I)) GO TO 88
      BETP(I) = TERMB(I)*A234(I)*FRCP(I)*(A4(I)-FRCP(I)) /
     A         (A1(I)*FRCP(I)*(FRCP(I)-A2(I)) -A0(I)*A2(I))
      BETBR(I) = (TERM(I)+BETP(I)-PSITEM(I))*(A12(I)*FRCP(I)
     A         *(FRCP(I)-A2(I)) -A0(I)) / FS(I)
      GO TO 89
   88 BETP(I) = TERMB(I)*OMT2A2(I)
      BETBR(I) = (TERM(I)+BETP(I)-PSITEM(I))*
     A         (OMT2M1(I)-A0(I)) / FS(I)
   89 IF(ABS(BETBR(I)).LT.3.0)GO TO 90
      FS(I) = SIGN(FS(I),BETBR(I))
      GO TO 100
   90 FS(I) =FS(I)*(BETBR(I)-BETBR(I)*ABS(BETBR(I))/3. +BETBR(I)**3/27.)
  100 CONTINUE
C          AT 105 SET UP DIFFERENTIAL EQUATIONS FOR PINTIR
C
C  SUBROUTINE ADJTQB ADJUSTS BRAKE TORQUES AT EACH WHEEL
      IF(NTRA.EQ.2) CALL ADJTQB
      DO 107 I=1,4
      RRM(I) = 0.0
      IF(ABS(RPSI(I)).GT.ZETAB)RRM(I) =-RRMC(I)*FR(I)*SIGN(1.0,RPSI(I))
  107 CONTINUE
  105 J=0
      DO 110  I = 1,3, 2
      J= J + 1
      ARFTT = 6.0 * ARBR(J) * TQD(J)
      DRPSI(I)=RPSFA(J)*(12.*TQB(I)+ARFTT)-FC(I)*HI(I)*RPSFA(J)
     1  -RPSFB(J)*(12.0*TQB(I+1)+ARFTT)+FC(I+1)*HI(I+1)*RPSFB(J)
     2      +RPSFA(J)*RRM(I)-RPSFB(J)*RRM(I+1)
      DRPSI(I+1)=RPSFA(J)*(12.0*TQB(I+1)+ARFTT)-FC(I+1)*HI(I+1)*RPSFA(J)
     1         -RPSFB(J)*(12.0*TQB(I)+ARFTT) + FC(I) * HI(I) * RPSFB(J)
     2      +RPSFA(J)*RRM(I+1)-RPSFB(J)*RRM(I)
  110 CONTINUE
      RETURN
  999 WRITE(6,1000) I,CCT,XMUXP(I),KK,RMU,SC0,SC1,SC2
 1000 FORMAT(1H ,48HERROR IN CALCULATION OF XMUX, SUBROUTINE DAUXR   /
     A  5X,3HI =,I4,2X,5HCCT =,F8.3,2X,7HXMUXP =,F8.3,2X,4HKK =,I4,
     B  2X,5HRMU =,F8.4 / 5X,5HSC0 =,E12.6,2X,5HSC1 =,E12.6,
     C 2X,5HSC3 =,E12.6  )
      STOP
      END
```

```
      SUBROUTINE DRIVER(PSIFF,PSIFFD,JJ)
C           HVOSM-VD2 VERSION
C                REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1R1(2),
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1            (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIX2P,XIYZP,D1PD2,D1MD2,
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
```

```
     9              ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1              ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2              TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3              HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4             ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
C       COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,
     1                   COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,
     2                   STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE,
     1                 IDRVER,IBUG
C        IPATHT - STOP FOR DRIVER MODEL
C        IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL
C        ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE
C        ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,
     1              TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,
     2              TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,
     3              VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,
     4              TESTT(5),DESSI(5),DISTI(5),PSIFHO,XIMPOR(9),
     5              BFP1,BFP2,DRIEND
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,
     1              TESTR1,TESTR2,THESKD,FBRK,APB,DSDES,
     2              TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,
     3              APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSO2,QAY,
     4              AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,
     5              TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,
     6              PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),
     7              PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),
     8              DEND
C
      JJ = 1
      UT = SQRT(U*U+V*V+W*W)
C         CALCULATE AT INITIALIZATION AND ON IDRIVE = 1. IF NOT TIME
C         FOR DRIVER SAMPLE (IDRIVE.NE.1) BRANCH TO 50 AND CONTINUE
C         FILTER SUMMATION FOR PSIF. ALSO DONOT CHAGE APD,FBRK.
C         BRAKING, DRIVING TORQUE COMPUTED IN CTQB AND CTQD GIVEN
C         FBRK,APD.
      IF(IBUG.GT.1. OR.IBUG*IDRIVE.NE.0) WRITE(6,3001) T,IDRIVE
 3001 FORMAT(3H T=,F9.5,8H IDRIVE=,I2 )
      IF(IDRIVE.EQ.0) GO TO 50
  211 IDRIVE = 0
      THESKD = 0.0
      IF(U.NE.0.0.OR.V.NE.0.0) THESKD = ATAN2(V,U)
      IF(ISKIDP.EQ.2) GO TO 228
      IF(ABS(THESKD).GT.TESTR1) GO TO 225
      ISKIDP = 0
```

```
      GO TO 215
  225 ISKIDP = 1
      APD = 0.0
      FBRK = 0.0
      IF(IBUG.NE.0) WRITE(6,3005) T,THESKD,APD,FBRK,PSISKD,ISKIDP
 3005 FORMAT(3H T=,F9.5,' THESKD=',F10.4,' APD=',F8.2,' FBRK=',F8.2,
     X        ' PSISKD=',F10.4,' ISKIDP=',I4,' STMT 225' )
      IF(ABS(THESKD).LT.TESTR ) GO TO 300
      ISKIDP = 2
  228 PSISKD = FKSKID*(THESKD-PSIFF)
      IF(IBUG.NE.0) WRITE(6,3006) T,THESKD,APD,FBRK,PSISKD,ISKIDP
 3006 FORMAT(3H T=,F9.5,' THESKD=',F10.4,' APD=',F8.2,' FBRK=',F8.2,
     A        ' PSISKD='F10.4,' ISKIDP=',I4,' STMT 228' )
      GO TO 49
C         PSISKD IS AN ENTRY INTO PUSH-DOWN TABLE, BYPASS PATH FOLLOWING
C         PSIFF IS OUTPUT FROM PREVIOUS SAMPLE TIME FOR PATH FOLLOWING
C
C         ISKIDP NOT ZERO INDICATES PSISKD HAS BEEN USED INSTEAD OF
C                PATH FOLLOWER
C         VEHICLE ORIENTATION NOT TO CHANGE MORE THAN 90 DEG. IN A SKID
C
  215 CALL DRIVP
      IF(IBUG.NE.0)WRITE(6,3003) T,APD,FBRK
 3003 FORMAT(3H T=,F9.5,' APD=',F8.2,' FBRK=',F8.2, ' FROM DRIVP' )
  300 CONTINUE
      UPP = AMTX(1,1)*U+AMTX(1,2)*V+AMTX(1,3)*W
      VPP = AMTX(2,1)*U+AMTX(2,2)*V+AMTX(2,3)*W
      ST = DS/UT
      STSO2 = ST*ST/2.0
      QAY = UT*UT*PSIM/(APB*(1.0+FKD*UT**2))
      AXP = AMTX(1,2)*QAY
      AYP = AMTX(2,2)*QAY
      ET = 0.0
C         NEN IS NO. OF SAMPLE POINTS ALONG PROJESTED PATH (MAX.OF 7)
      DO 70 I=1,NEN
      DI = FLOAT(I)
      XVP = XCP+ST*DI*UPP+STSO2*DI**2*AXP
      YVP = YCP+ST*DI*VPP+STSO2*DI**2*AYP
      SLOPX = UPP+ST*DI*AXP
      SLOPY = VPP+ST*DI*AYP
      SLOPER = -(SLOPY/SLOPX)
C
C         PSIJ IS ANGLE OF ROTATION TO ACHIEVE NEW AXIS, Y", ALONG
C            THE ERROR LINE BY A CLOCKWISE ROTATION
C         THE PURPOSE IS TO HAVE THE SIGN OF THE ERROR IN THIS AXIS
C            SYSTEM INDICATE THE DIRECTION OF STEERING
C         PSIJJ FOR THE ERROR LINE, WHICH IS ANGLE OF PREDICTED PATH
C            + PI/2 .   BUT ROTATE BY (-PI+PSIJJ) WHICH IS ANGLE OF
C            PREDICTED PATH - PI/2   SINCE STEERING IS NEGATIVE
C            COUNTERCLOCKWISE BUT PSIJJ IS POSITIVE CLOCKWISE BY
C            MATHEMATICAL CONVENTION .
```

```
C
      PSIJ = ATAN2(SLOPX,SLOPY) - PIU2
C
      XINT = XVP-SLOPER*YVP
C        SOLVE FOR X,Y COORD. ON CLOSEST VALID DESIRED PATH EQN.
C        IF NO SOLUTION, IPATHT = 1
      DO 60 J=1,5
      Y =  (S(J,1)-XINT)/(SLOPER-S(J,2))
      IF(Y.GT.YTRANS(J).AND.Y.LE.YTRANS(J+1)) GO TO 61
   60 CONTINUE
      IPATHT = 1
      WRITE(6,3010) T,IPATHT
 3010 FORMAT(3H T=,F9.5,' IPATHT=',I4,' NO SOLN. FOR ERROR' )
      RETURN
   61 X = S(J,1)+S(J,2)*Y
      TERMX = X-XVP
      TERMY = Y-YVP
      AE = SQRT(TERMX**2+TERMY**2)
      TEMPOR = SIN(PSIJ)*TERMX+COS(PSIJ)*TERMY
C        EI IS THE ITH ERROR, EWT IS THE WEIGHTED ITH ERROR
C        ET HOLDS ACCUMULATION OF WEIGHTED ERRORS
      EI = SIGN(AE,TEMPOR)
      EWT = EI*WEIGHT(I)
      AREI(I) = EI
      ARCAPE(I) = EWT
   70 ET = ET + EWT
      IF(IBUG.NE.0)WRITE(6,98)T,AREI,ARCAPE
   98 FORMAT(3H T=,F9.5,' EI=',7F10.3, /' EWT=',7F10.3 )
      DPSISF = FKP*ET
      IF(ABS(PSIFF).GE.OMGPS.AND.PSIFF*DPSISF.GT.0.0) GO TO 50
C        NPD = NO. OF DELTA-PSI'S IN PUSH-DOWN TABLE-PPD(I).
C        EACH DELTA-PSI HAS AN ASSOCIATED TIME-TPD(I)
      NPD = NPD+1
      PPD(NPD) = DPSISF
      TPD(NPD) = T
C        PSIM IS UNFILTERED PSIFF
      PSIM = PSIM + DPSISF
      GO TO 50
   49 IF(ABS(PSIFF).GE.OMGPS.AND.PSIFF*PSISKD.GT.0.0)GO TO 50
      NPD = NPD+1
      PPD(NPD) = PSISKD
      TPD(NPD) = T
      PSIM = PSIM + PSISKD
   50 KCOUNT = 0
      PSIFF = PSIFFH
      PSIFFD = 0.0
      IF(NPD.EQ.0) GO TO 75
      DO 74 J=1,NPD
      TITE = T-TPD(J)-TAUF
      IF(TITE.LT.0.0) GO TO 74
      IF(TITE.GE.1.0) GO TO 73
```

```
   72 DPSILF = PPD(J)*(1.0-TMT*EXP(-TITE/TIL))
      PSIFF = PSIFF + DPSILF
      PSIFFD = PSIFFD+(PPD(J)-DPSILF)/TIL
      GO TO 74
   73 DPSILF = PPD(J)
      KCOUNT = KCOUNT+1
      PSIFF = PSIFF + DPSILF
   74 CONTINUE
      IF(KCOUNT.EQ.0) GO TO 75
      DO 16 J=1,KCOUNT
   16 PSIFFH = PSIFFH+PPD(J)
      NPD = NPD-KCOUNT
      DO 17 J=1,NPD
      PPD(J) = PPD(J+KCOUNT)
      TPD(J) = TPD(J+KCOUNT)
   17 CONTINUE
      KCOUNT = 0
   75 CONTINUE
C
      PSIJD = PSIJ/RAD
      IF(IBUG.GT.1.OR.IBUG*IDRIVE.NE.0) WRITE(6,97) T,PSIFF,
     X     PSIFFH,DPSILF,DPSISF,PSIJD,
     X               ISKIDP,NPD,(PPD(K),K=1,NPD),(TPD(K),K=1,NPD)
   97 FORMAT(3H T=,F9.5,' PSIFF=',F10.5,' PSIFFH=',F10.5,' DPSILF=',
     A       F10.5,' DPSISF=',F10.5,' PSIJD=',F10.5,,' ISKIDP=',I2,
     B       ' NPD=',I2  / ' PPD,TPD=', 10F10.5/' ',10F10.5 )
C
      APSI = ABS(PSIFF)
      IF(APSI.LE.OMGPS) GO TO 79
      PSIFF = SIGN(OMGPS,PSIFF)
   79 APSIM = ABS(PSIM)
      IF(APSIM.GT.OMGPS) PSIM=SIGN(OMGPS,PSIM)
      IF(ABS(PSIFFH).GT.OMGPS) PSIFFH=SIGN(OMGPS,PSIFFH)
   89 RETURN
      END
```

```
      SUBROUTINE DRIVID
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6          HED(36),DADE(3),X1R,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1           NPAGE(20)
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE,
     1             IDRVER,IBUG
C        IPATHT - STOP FOR DRIVER MODEL
C        IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL
C        ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE
C        ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,
     1           TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,
     2           TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,
     3           VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,
     4           TESTT(5),DESSI(5),DIST1(5),PSIFHO,XIMPOR(9),
     5           BFP1,BFP2,DRIEND
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,
     1           TESTR1,TESTR2,THESKD,FBRK,APB,DSDES,
     2           TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,
     3           APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSQ2,QAY,
     4           AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,
     5           TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,
     6           PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),
     7           PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),
     8           DEND
      WRITE(6,1001) EN,EMDT,DS,TAUF,TIL,TL
 1001 FORMAT(1H0,
     A  9X,52HNO.OF SAMPLE POINTS ALONG PROJECTED PATH     EN       = ,
     B  F8.2  /
     C 10X,52HTIME BETWEEN DRIVER SAMPLES                  EMDT     = ,
     D  F6.2,2X,3HSEC    /
     E 10X,52HINCREMENTAL DISTANCE ALONG PROJECTED PATH    DS       = ,
     F  F8.2,2X,6HINCHES  /
     G 10X,52HSTEERING FILTER TIME DELAY                   TAUF     = ,
     H  F8.2,2X,3HSEC         /
     I 10X,52HSTEERING FILTER TIME LAG                     TIL      = ,
```

```
    J  F8.2,2X,3HSEC      /                                          DRVD0500
    K 10X,52HSTEERING FILTER TIME LEAD                 TL     = ,    DRVD0510
    L  F8.2,2X,3HSEC      )                                          DRVD0520
     WRITE(6,1002) TESTS1,TESTS2,TESTB,TESTR1,TESTR2,FKP             DRVD0530
1002 FORMAT(1H ,                                                     DRVD0540
    A  9X,52HSPEED RESPONSE THRESHOLD                  TESTS1 = ,    DRVD0550
    B  F8.2,2X,6HIN/SEC  /                                           DRVD0560
    C 10X,52HSPEED RESPONSE INDIFFERENCE LEVEL         TESTS2 = ,    DRVD0570
    D  F8.2,2X,6HIN/SEC  /                                           DRVD0580
    E 10X,52HBRAKE PEDAL APPLICATION INDIFFERENCE LEVEL  TESTB  = ,  DRVD0590
    F  F8.2,2X,9HIN/SEC**2  /                                        DRVD0600
    G 10X,52HSKID CONTROL THRESHOLD                    TESTR1 = ,    DRVD0610
    H  F8.2,2X,7HRADIANS    /                                        DRVD0620
    I 10X,52HSKID CONTROL INDIFFERENCE LEVEL           TESTR2 = ,    DRVD0630
    J  F8.2,2X,7HRADIANS    /                                        DRVD0640
    K 10X,52HCONTROL GAIN FOR FRONT WHEEL STEER ANGLE   FKP    = ,   DRVD0650
    L  F8.2,2X,6HRAD/IN      )                                       DRVD0660
     WRITE(6,1003) FKSKID,FKS1,FKS2,FKD                              DRVD0670
1003 FORMAT(1H ,                                                     DRVD0680
    A  9X,52HSKID CONTROL GAIN                         FKSKID = ,    DRVD0690
    B  F8.2,2X,7HRAD/RAD    /                                        DRVD0700
    C 10X,52HSPEED RESPONSE BRAKE PEDAL FORCE GAIN     FKS1   = ,    DRVD0710
    D  F8.2,2X,12HLB/IN/SEC**2    /                                  DRVD0720
    E 10X,52HSPEED RESPONSE ACCELERATOR PEDAL GAIN     FKS2   = ,    DRVD0730
    F  F8.2,2X,12HIN/IN/SEC**2    /                                  DRVD0740
    G 10X,52HHANDLING QUALITY CONSTANT                 FKD    = ,    DRVD0750
    H  F8.2,2X,11H(SEC/IN)**2     )                                  DRVD0760
     WRITE(6,1004) APDMAX,BFP1,BFP2                                  DRVD0770
1004 FORMAT(1H ,                                                     DRVD0780
    A  9X,52HMAXIMUM ACCELERATOR PEDAL DEFLECTION      APDMAX = ,    DRVD0790
    B  F8.2,2X,6HINCHES      /                                       DRVD0800
    C 10X,52HBRAKE SYSTEM PRESSURE VS. BRAKE PEDAL     BPF1   = ,    DRVD0810
    D  F8.2,2X,6HPSI/LB      /                                       DRVD0820
    E 10X,52H    FORCE COEFFICIENTS                    BPF2   = ,    DRVD0830
    F  F8.2,2X,9HPSI/LB**2  /     )                                  DRVD0840
     WRITE(6,1005)                                                   DRVD0850
1005 FORMAT(1H0,                                                     DRVD0860
    A 1X,49H          DESIRED PATH DATA                    ,         DRVD0870
    B    30H       SPEED CHANGE DATA        ,                        DRVD0880
    C    32H         IMPORTANCE       ERROR    ,    /                DRVD0890
    D 1X,49H   Y' TRANSITION     X' INTERCEPT    SLOPE      ,        DRVD0900
    E    30H TIME    DESIRED     NULL    ,                           DRVD0910
    F    32H          WEIGHTING    WEIGHTING ,    /                  DRVD0920
    G 1X,49H          POINTS                               ,         DRVD0930
    H    30H            SPEED    DISTANCE  ,                         DRVD0940
    I    32H           FUNCTION      FUNCTION      )                 DRVD0950
     WRITE(6,1008)                                                   DRVD0960
1008 FORMAT(1H ,                                                     DRVD0970
    A 1X,49H        YTRANS           S(I,1)      S(I,2)     ,        DRVD0980
    B    30H TESTT    DESSI       DISTI    ,   /                     DRVD0990
    C 1X,49H          IN              IN                   ,         DRVD1000
```

```
    D     30H   SEC     IN/SEC          IN      ,  /  )
      WRITE(6,1006) (YTRANS(I),S(I,1),S(I,2),TESTT(I),DESSI(I),DISTI(I),
    A               XIMPOR(I),WEIGHT(I),I=1,5)
 1006 FORMAT(8X,F8.2,4X,F10.2,4X,F8.2,8X,F8.2,2X,F8.2,2X,F8.2,10X,F8.2,
    A        6X,F8.2 )
      WRITE(6,1007) (XIMPOR(I),WEIGHT(I),I=6,7)
 1007 FORMAT(88X,F8.2,6X,F8.2 )
      WRITE(6,1009) GEAR1,VGR12,VGR43,GEAR2,VGR23,VGR32,
    1               GEAR3,VGR34,VGR21,GEAR4
 1009 FORMAT(1H0,
    A  9X,50HTRANSMISSION      ENGINE UPSHIFT    ENGINE DOWNSHIFT     /
    B 10X,50H GEAR RATIO        SPEED - RPME       SPEED - RPME      //
    C 10X,5H1ST =,F7.3,5X,7HVGR12 =,F7.1,4X,7HVGR43 =,F7.1 /
    D 10X,5H2ND =,F7.3,5X,7HVGR23 =,F7.1,4X,7HVGR32 =,F7.1 /
    E 10X,5H3RD =,F7.3,5X,7HVGR34 =,F7.1,4X,7HVGR21 =,F7.1 /
    F 10X,5H4TH =,F7.3    )
      RETURN
      END
```

```
      SUBROUTINE DRIVP
C            HVOSM-VD2 VERSION
C            REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE,
     1              IDRVER,IBUG
C       IPATHT - STOP FOR DRIVER MODEL
C       IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL
C       ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE
C       ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKD0,FKP0,FKS10,FKS20,FKSKD0,
     1             TESTB0,TSTS10,TSTS20,TSTR10,TSTR20,OMEGA0,TAUF,TTL,
     2             TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,
     3             VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,
     4             TESTT(5),DESSI(5),DISTI(5),PSIFH0,XIMPOR(9),
     5             BFP1,BFP2,DRIEND
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTB,TESTS1,TESTS2,
     1             TESTR1,TESTR2,THESKD,FBRK,APB,DSDES,
     2             TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,
     3             APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSO2,GAY,
     4             AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,
     5             TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,
     6             PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),
     7             PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),
     8             DEND
C
C        ISMAIN NOT ZERO FOR SPEED MAINTENANCE, DO NOT UPDATE DIST
C
   10 IF(ISMAIN.NE.0) GO TO 15
      DIST = DIST-UT*EMDT
      IF(DIST.GT.0.0) GO TO 15
      DIST = DISTC
      ISMAIN = 1
C       NOTE: TESTT(1) MUST = T0
   15 IF(ITCHNG.EQ.0) GO TO 25
C         CHANGE DESIRED SPEED AND DISTANCE HERE
      ITTT = ITESTT-1
      IF(ITTT.GT.5) GO TO 25
      DESS = DESSI(ITTT)
      DISTC = DISTI(ITTT)
      DIST = DISTC
      ISMAIN = 0
      IF(IBUG.NE.0) WRITE(6,99) ITTT,DESS,DIST
   99 FORMAT(1H ,5X,6H ITTT=,I2,12H, DES.SPEED=,F10.3,7H, DIST=,F10.3)
   25 DELTV = DESS - UT
      FBRK = 0.0
      IF(DELTV.GE.0) GO TO 31
      IF(ABS(DELTV).GE.TESTS2) GO TO 32
C         NO CHANGE IN APD
   31 IF(DELTV.LT.TESTS1) GO TO 50
```

```
   32 DELTAX = (DELTV*UT)/DIST                                      DRVP0500
      IF(DELTAX.GT.0.0) GO TO 45                                    DRVP0510
      IF(ABS(DELTAX).LE.TESTB) GO TO 45                             DRVP0520
C        HERE DELTAX IS NEGATIVE, FKS1 IS INPUT AS POSITIVE,        DRVP0530
C        THEREFORE CHANGE SIGN TO GET POSITIVE PC IN CTQB           DRVP0540
      APD = 0.0                                                     DRVP0550
      FBRK =-FKS1*DELTAX                                            DRVP0560
      GO TO 50                                                      DRVP0570
C        AT STMT 45, ACCELERATION INCREASE, INCREASE ACC.PEDAL DEFL. DRVP0580
C        OR IF DELAX NEGATIVE BUT NOT BRAKING, DECREASE APD.        DRVP0590
   45 APD = APD+FKS2*DELTAX                                         DRVP0600
      IF(APD) 46,50,47                                              DRVP0610
   46 APD = 0.0                                                     DRVP0620
      GO TO 50                                                      DRVP0630
   47 IF(APD.GT.APDMAX) APD = APDMAX                                DRVP0640
   50 RETURN                                                        DRVP0650
      END                                                           DRVP0660
```

```
      SUBROUTINE DRVCNS
C           HVOSM-VD2 VERSION  -
C           REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1          CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2          PSIFIO,PSIFDO
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1          GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2          TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3          BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4          XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5          ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7          SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8          SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9          ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1          ZETA3D,SF21,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2          TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3          HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5          XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1          DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2          PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3          SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1          PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2          CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3          STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4          XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5          YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
```

```
     6          CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),           DRVC0500
     7          CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),            DRVC0510
     8          SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),DRVC0520
     9          FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CCXW(4)DRVC0530
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),          DRVC0540
     1          BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),           DRVC0550
     2          FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),         DRVC0560
     3          F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                        DRVC0570
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                           DRVC0580
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),      DRVC0590
     1          (PSII(1),PSI1)                                              DRVC0600
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),       DRVC0610
     A            XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),                    DRVC0620
     B            XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),                DRVC0630
     C            XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4                   DRVC0640
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)          DRVC0650
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)          DRVC0660
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAE,    DRVC0670
     1            CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),           DRVC0680
     2            TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)DRVC0690
     3          ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,               DRVC0700
     4            BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)                 DRVC0710
      COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,DRVC0720
     1            NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,       DRVC0730
     2            RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,          DRVC0740
     3            DELTAE,PIO15R,COMEN5,TQE,RPME                             DRVC0750
      COMMON /INTR/ NECR,TIMR,DTR,VARR(12),DERR(12)                         DRVC0760
      DIMENSION RPSI(4),DRPSI(4)                                            DRVC0770
      EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                      DRVC0780
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE,     DRVC0790
     1            IDRVER,IBUG                                               DRVC0800
C       IPATHT - STOP FOR DRIVER MODEL                                      DRVC0810
C       IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL                     DRVC0820
C       ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE                DRVC0830
C       ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES          DRVC0840
      COMMON/DRIVI/NEN,EMDT,ES,DS,APDMAX,FKDO,FKPO,FKS10,FKS20,FKSKDO,      DRVC0850
     1          TESTBO,TSTS10,TSTS20,TSTR10,TSTR20,OMEGAO,TAUF,TIL,         DRVC0860
     2          TL,S(5,2),NTRAN,YTRANS(6),GEAR1,GEAR2,GEAR3,GEAR4,          DRVC0870
     3          VGR12,VGR23,VGR34,VGR43,VGR32,VGR21,                        DRVC0880
     4          TESTT(5),DESSI(5),DISTI(5),PSIFHO,XIMPOR(9),                DRVC0890
     5          BFP1,BFP2,DRIEND                                            DRVC0900
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTE,TESTS1,TESTS2,         DRVC0910
     1          TESTR1,TESTR2,THESKD,FBRK,APB,DSCLS,                        DRVC0920
     2          TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,                  DRVC0930
     3          APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSQ2,QAY,        DRVC0940
     4          AXP,AYP,D1,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,           DRVC0950
     5          TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,          DRVC0960
     6          PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),          DRVC0970
     7          PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),          DRVC0980
     8          DEND                                                        DRVC0990
      DIMENSION VGRU(3),VGRD(3),GEAR(4)                                     DRVC1000
```

```
      EQUIVALENCE (VGRU(1),VGR12),(VGRD(1),VGR43),(GEAR(1),GEAR1)
C
C        TRKIN IS JUST LARGER THAN T ON FIRST RK STEP. (HALF INTERVAL)
      TRKIN = T0 + 0.51*DTCOMP
      TTPSIT = PSI0*RAD
      CONMPH = 3600./(12.*5280.)
      THESKD = 0.0
      FBRK = 0.0
      FKD = FKD0
      IF(OMGPS.EQ.0) OMGPS = 30.0*RAD
      PSIFFH = PSIFI0
      APB = A+B
      FKP = (2.0*APB*(1.0+FKD*2.76E5))/(EN*DS*DS)
C         2.78E5 IS SQUARE OF 30 MPH IN IN/SEC
      DO 61 I=1,NEN
      DI = FLOAT(I)
   61 WEIGHT(I) = XIMPOR(I)/(DI*DI)
      TMT = (TIL-TL)/TIL
C        TMT IS MULTIPIER FOR FITERING EQUATIN IN DRIVER
      FKS1 = FKS10
      FKS2 = FKS20
      FKSKID = FKSKD0
      TESTB = TESTB0
      TESTS1 = TSTS10
      TESTS2 = TSTS20
      TESTR1 = TSTR10
      TESTR2 = TSTR20
      IDRIVE = 1
      DELPTH = EMDT
      TPATH = T0 + DELPTH
C        IDRIVE, TPATH AND DELPTH USED TO DETERMINE DRIVER SAMPLE TIME
C        ASSUME SPEED MAINTENANCE MODE UNTIL SECOND CHANGE TIME
      DO 13 I=1,5
   13 TCTEST(I) = TESTT(I)
      ISMAIN = 1
      ITCHNG = 0
      ITESTT = 2
      DESS = DESSI(1)
      DISTC = DISTI(1)
      DIST = DISTC
C        GET INITIAL APD FOR EQUILIBRIUM
      L = 1
      IF(JDEND.EQ.2) L=3
      DO 10 I=1,4
   10 RPSI(I) = U0/HI(I)
      RPMOTR = PI015R*ARBR(JDEND)*(RPSI(L)+RPSI(L+1))
      DO 70 I=1,3
      IGEAR = I
      IF(RPMOTR*GEAR(I).LT.VGRU(I)) GO TO 71
   70 CONTINUE
      IGEAR = 4
```

```
71 TTTR = GEAR(IGEAR)
   RPME = RPMOTR*TTTR
   DF = -CONE*U0*ABS(U0)-CTWO*U0-CTHREE*SIGN(1.,U0)
   DO 11 I=1,4
11 DF = DF-PRMC(I)*FR(I)/HI(I)
   TQD(JDEND) = -DF*0.5*(HI(L)+HI(L+1))/(ARBR(JDEND)*12.0)
   TQE = TQD(JDEND)/TTTR
   DO 30 I=2,NRPM
   IA = I-1
   TEST = ABS(RPME-TRPME(IA))
   IF(TEST.LT.0.05) GO TO 24
   IF(TRPME(IA).LE.RPME.AND.RPME.LT.TRPME(I)) GO TO 34
30 CONTINUE
   IA = NRPM
24 TQWOT = TWOT(IA)
   TQCT = TCT(IA)
   GO TO 37
34 RATIO = (RPME-TRPME(IA))/(TRPME(IA+1)-TRPME(IA))
   TQWOT = TWOT(IA)+RATIO*(TWOT(IA+1)-TWOT(IA))
   TQCT  = TCT(IA) +RATIO*(TCT(IA+1)-TCT(IA))
37 TTTS = (TQE-TQCT)/(TQWOT-TQCT)
   APD = TTTS*APDMAX
   RETURN
   END
```

```
      SUBROUTINE GCP(I)                                                 GCP 0010
C             HVOSM-VD2 VERSION                                         GCP 0020
C             REVISED OCTOBER 1975   CALSPAN CORPORATION                GCP 0030
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,GCP 0040
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),  GCP 0050
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),GCP 0060
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),    GCP 0070
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), GCP 0080
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), GCP 0090
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),    GCP 0100
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),     GCP 0110
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),GCP 0120
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)GCP 0130
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      GCP 0140
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),    GCP 0150
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),  GCP 0160
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                 GCP 0170
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       GCP 0180
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   GCP 0190
     1             (PSII(1),PSI1)                                       GCP 0200
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,GCP 0210
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         GCP 0220
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, GCP 0230
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   GCP 0240
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   GCP 0250
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPGCP 0260
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    GCP 0270
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   GCP 0280
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, GCP 0290
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  GCP 0300
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  GCP 0310
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, GCP 0320
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,GCP 0330
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2GCP 0340
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,GCP 0350
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          GCP 0360
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 GCP 0370
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       GCP 0380
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   GCP 0390
     1                A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),    GCP 0400
     2                A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)         GCP 0410
    1  XLM1(I) = XP(I)*CAYW(I)+YP(I)*CBYW(I)+ZP(I)*CGYW(I)              GCP 0420
      XLM2(I) = XP(I)*CAGZ(I)+YP(I)*CBGZ(I)+ZPGI(I)*CGGZ(I)             GCP 0430
      XLM3(I) = D1(I)*XP(I)+D2(I)*YP(I)+D3(I)*ZP(I)                     GCP 0440
    2 CMTX(1,1) = CAYW(I)                                               GCP 0450
      CMTX(1,2) = CBYW(I)                                               GCP 0460
      CMTX(1,3) = CGYW(I)                                               GCP 0470
      CMTX(1,4) = XLM1(I)                                               GCP 0480
      CMTX(2,1) = CAGZ(I)                                               GCP 0490
```

```
      CMTX(2,2) = CBGZ(I)                                          GCP 0500
      CMTX(2,3) = CGGZ(I)                                          GCP 0510
      CMTX(2,4) = XLM2(I)                                          GCP 0520
      CMTX(3,1) = D1(I)                                            GCP 0530
      CMTX(3,2) = D2(I)                                            GCP 0540
      CMTX(3,3) = D3(I)                                            GCP 0550
      CMTX(3,4) = XLM3(I)                                          GCP 0560
      CALL SIMSOL(CMTX,3,3)                                        GCP 0570
    3 XGPP(I) = CMTX(1,4)                                          GCP 0580
      YGPP(I) = CMTX(2,4)                                          GCP 0590
      ZGPP(I) = CMTX(3,4)                                          GCP 0600
      TX = XGPP(I)-XP(I)                                           GCP 0610
      TY = YGPP(I)-YP(I)                                           GCP 0620
      TZ = ZGPP(I)-ZP(I)                                           GCP 0630
      DELTA(I) = SQRT(TX**2+TY**2+TZ**2)                           GCP 0640
      CAR(I) = TX/DELTA(I)                                         GCP 0650
      CBR(I) = TY/DELTA(I)                                         GCP 0660
      CGR(I) = TZ/DELTA(I)                                         GCP 0670
      HI(I) = AMIN1(DELTA(I),RW(I))                                GCP 0680
    4 FR(I) = 0.0                                                  GCP 0690
      IF(RW(I).EQ.HI(I)) RETURN                                    GCP 0700
      TRH = RW(I)-HI(I)                                            GCP 0710
      IF(TRH.GT.SIGT(I)) GO TO 5                                   GCP 0720
      FR(I) = AKT(I)*TRH                                           GCP 0730
      RETURN                                                       GCP 0740
    5 FR(I) = AKT(I)*(XLAMT(I)*(TRH-SIGT(I))+SIGT(I))              GCP 0750
      RETURN                                                       GCP 0760
      END                                                          GCP 0770
```

```
      SUBROUTINE IDOUT                                                  IDOT0010
C             HVOSM-VD2 VERSION                                         IDOT0020
C                 REVISED OCTOBER 1975  CALSPAN CORPORATION             IDOT0030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        IDOT0040
     1             NPAGE(20)                                            IDOT0050
      COMMON/INPT/PHI0,THETA0,PSI0,P0,Q0,R0,XCOP,YCOP,ZCOP,U0,V0,W0,    IDOT0060
     1            A,B,DEL10,DEL20,DEL30,PHIR0,DEL10D,DEL20D,DEL30D,     IDOT0070
     2            PHIR0D,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     IDOT0080
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,IDOT0090
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,T0, IDOT0100
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    IDOT0110
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  IDOT0120
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  IDOT0130
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   IDOT0140
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 IDOT0150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),IDOT0160
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDR0(4,5),UVWMIN,PQRMIN  IDOT0170
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          IDOT0180
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,  IDOT0190
     2             PSIFI0,PSIFD0                                        IDOT0200
      DIMENSION YCIP(2)                                                 IDOT0210
      EQUIVALENCE (YC1P(1),YC1P)                                        IDOT0220
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,P1,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,IDOT0230
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZR0,TR02,         IDOT0240
     2             TF02,TIZ,RH02,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,A02APB, IDOT0250
     3             B02APB,RFTF,TS02,RRTS,BROMUR,XMUFC2,AXMF02,XMTF04,   IDOT0260
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,X1XZP,X1YZP,D1PD2,D1MD2,   IDOT0270
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPIDOT0280
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    IDOT0290
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   IDOT0300
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, IDOT0310
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  IDOT0320
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  IDOT0330
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, IDOT0340
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,IDOT0350
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2IDOT0360
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,IDOT0370
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          IDOT0380
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 IDOT0390
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       IDOT0400
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,     IDOT0410
     1               OMEGRC,AKRE,AKREP,OMEGRE,END3                      IDOT0420
      COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,        IDOT0430
     1               DAPRB,DAPRE,DDAPR,NAPR                             IDOT0440
      DIMENSION APF(21),APR(21)                                         IDOT0450
      EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                 IDOT0460
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),   IDOT0470
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     IDOT0480
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          IDOT0490
```

```
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      COMMON/DRIVTT/TPATH,DELPTH,TCTEST(6),ITESTT,ITCHNG,IPATHT,IDRIVE,
     1              IDRVER,IBUG
C        IPATHT - STOP FOR DRIVER MODEL
C        IDRIVE - NOT ZERO, SAMPLE TIME FOR DRIVER MODEL
C        ITCHNG - NOT ZERO, CHANGE DESIRED SPEED AND DISTANCE
C        ITESTT,TCTEST(6) - INDEX AND INPUT TIMES FOR SPEED CHANGES
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),
     A               XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),
     B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),
     C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,
     2               PHIC3,PHIC4,PHIC5,PHIC6,NCRESL,
     3               TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,
     4               PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,
     5               YCMP(6),ZCMP(6),PHICM(6)
C
      DATA ZERO/0.0/
      DIMENSION TTARG(50) ,NTARG(10)
      DATA TTARG/50*0.0/ , NTARG/10*0/
      DIMENSION TXARG(21), TYARG(21)
      DATA TXARG/21*0.0/,TYARG/21*0.0/
      DATA CONI/4HCONS/, VARI/4HVARI/
      DIMENSION DINCH(2),DEG(2),DIPS(2),DPS(2),PS2PI(3),PS2I(3),
     1          DIPS2(3),PIPR(3),RAPRA(2),RADS(2),RPI(2),RPI2(3),
     2          RPI3(3),PPI(2),PPI3(2),PSPI(3),RAPS(2)
      DATA DINCH/4HINCH,4HES  /,  DEG/4HDEGR,4HEES /
      DATA   DPS/4HDEG/,4HSEC /, DIPS/4HIN/S,4HEC  /
      DATA RAPRA/4HRAD/,4HRAD /, RADS/4HRADI,4HANS /
      DATA   RPI/4HRAD/,4HIN  /,   PPI/4HLB/I,4HN   /
      DATA  PPI3/4HLB/I,4HN**3/, RAPS/4HRAD/,4HSEC /
      DATA PS2PI/4HLB-S,4HEC**,4H2/IN/, PS2I/4HLB-S,4HEC**,4H2-IN/
      DATA DIPS2/4HIN/S,4HEC**,4H2   /, PIPR/4HLB-I,4HN/RA,4HD   /
      DATA  RPI2/4HRAD/,4HIN**,4H2   /, RPI3/4HRAD/,4HIN**,4H3   /
      DATA  PSPI/4HLB-S,4HEC/I,4HN   /
      DATA SEC/4HSEC /
      DIMENSION PDI(2)
      DATA PDI/4HLB-I,4HN   /,PD/4HLB  /
      DIMENSION  TD1(2),TD2(2)
      DATA UD2/4HDEL2/,UPF/4HPHIF/,UD4/4HDEL4/,UPR/4HPHIR/
      DATA UDE/4HO   =/,UVE/4HOD =/
      DIMENSION TNU2(2),TNU3(3)
      DATA TNU2/4HNOT ,4HUSED/, TNU3/4HNOT ,4HUSED,4H    /
      DIMENSION TD3(2),T3D1(3),T3D2(3)
C
   11 WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
```

```
   1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),            IDOT1010
   2     (GHED(I),I=1,10),(SHED(I),I=1,10)                               IDOT1020
1000 FORMAT(1H1,9X,18A4,30X,2A4 / 5X,3(10A4) / 5X,2(10A4) / )           IDOT1030
     WRITE(6,1001) TO,SEC,T1,SEC,DTCOMP,SEC,MODE,DTPRNT,SEC              IDOT1040
1001 FORMAT(1H0,24X,39HP R O G R A M   C O N T R O L   D A T A  /        IDOT1050
   1 10X,38HSTART TIME                          TO      =,F10.4,2X,A4 /  IDOT1060
   2 10X,38HEND TIME                            T1      =,F10.4,2X,A4 /  IDOT1070
   3 10X,38HINTEGRATION INCREMENT               DTCOMP =,F10.4,2X,A4 /   IDOT1080
   4 62X,30H(0=VARIABLE STEP ADAMS-MOULTON    /                          IDOT1090
   5 10X,38HINTEGRATION MODE                    MODE    =,I5,            IDOT1100
   6  8X,16H-)1= RUNGA-KUTTA     /                                       IDOT1110
   7 62X,28H(2= FIXED STEP ADAMS-MOULTON   /                             IDOT1120
   8 10X,38HPRINT INTERVAL                      DTPRNT =,F10.4,2X,A4 )   IDOT1130
     WRITE(6,1002) ISUS,INDCRB,DELTC,SEC                                 IDOT1140
1002 FORMAT(1H ,                                                         IDOT1150
   1 61X,50H(0= INDEPENDENT FRONT SUSPENSION, SOLID REAR AXLE    /       IDOT1160
   2 10X,38HSUSPENSION OPTION                   ISUS    =,I5,            IDOT1170
   3  8X,42H-)1= INDEPENDENT FRONT AND REAR SUSPENSION    /              IDOT1180
   4 62X,42H(2= SOLID FRONT AND REAR AXLES                /              IDOT1190
   5 62X,42H(0= NO CURB, NO STEER DEGREE OF FREEDOM       /              IDOT1200
   6 10X,38HCURB/STEER OPTION                   INDCRB =,I5,             IDOT1210
   7  8X,10H-)1= CURB    /                                               IDOT1220
   8 62X,42H(-1=STEER DEGREE OF FREEDOM, NO CURB          /              IDOT1230
   9 10X,38HCURB INTEGRATION INCR.              DELTC   =,F10.5,2X,A4  ) IDOT1240
     WRITE(6,1003) COMEN4                                                IDOT1250
1003 FORMAT(1H ,                                                         IDOT1260
   A 9X,33HWHEEL SPIN EQUATION FACTOR     COMEN4 =,F10.5    )            IDOT1270
     IF(MODE.EQ.0) WRITE(6,1008) EBAR,EM,AAA,HMAX,HMIN,BET               IDOT1280
1008 FORMAT(1H0,9X,34HARGUMENTS FOR MODE 0 INTEGRATION :  /              IDOT1290
   A        8X,6(2X,F12.3)  )                                            IDOT1300
     WRITE(6,1004) XCOP,DINCH,UO,DIPS ,YCOP,DINCH,VO,DIPS,               IDOT1310
   A                ZCOP,DINCH,WO,DIPS                                   IDOT1320
1004 FORMAT(1H0,//,52X,38HI N I T I A L   C O N D I T I O N S   //       IDOT1330
   1 40X, 8HXCOP    =,F8.2,3X,2A4,39X,6HUO    =,F8.2,3X,2A4      /       IDOT1340
   2 10X,38HSPRUNG MASS C.G. POSITION      YCOP    =,F8.2,3X,2A4,        IDOT1350
   3  7X,38HSPRUNG MASS LINEAR VELOCITY     VO    =,F8.2,3X,2A4  /       IDOT1360
   4 40X, 8HZCOP    =,F8.2,3X,2A4,39X,6HWO    =,F8.2,3X,2A4        )     IDOT1370
     WRITE(6,1005) PHIO,DEG,PO,DPS,THETAO,DEG,QO,DPS,                    IDOT1380
   1                PSIO,DEG,RO,DPS                                      IDOT1390
1005 FORMAT(1H ,                                                         IDOT1400
   1 39X, 8HPHIO    =,F8.2,3X,2A4,39X,6HPO    =,F8.2,3X,2A4       /      IDOT1410
   2 10X,38HSPRUNG MASS ORIENTATION        THETAO =,F8.2,3X,2A4 ,        IDOT1420
   3  7X,38HSPRUNG MASS ANGULAR VELOCITY    QO    =,F8.2,3X,2A4 /        IDOT1430
   4 40X, 8HPSIO    = F8.2,3X,2A4,39X,6HRO    =,F8.2,3X,2A4       )      IDOT1440
     IF(ISUS.EQ.2) GO TO 101                                             IDOT1450
     UMP1 = UD2                                                          IDOT1460
     TD1(1) = DINCH(1)                                                   IDOT1470
     TD1(2) = DINCH(2)                                                   IDOT1480
     TD2(1) = DIPS(1)                                                    IDOT1490
     TD2(2) = DIPS(2)                                                    IDOT1500
     UMP = DEL20                                                         IDOT1510
```

```
      UMV = DEL20D
      GO TO 102
  101 UMP1 = UPF
      TD1(1) = DEG(1)
      TD1(2) = DEG(2)
      TD2(1) = DPS(1)
      TD2(2) = DPS(2)
      UMP = PHIFO
      UMV = PHIFOD
  102 WRITE(6,1006) DEL10,DINCH,DEL10D,DIPS,UMP1,UDE,UMP,TD1,UMP1,UVE,
     1             UMV,TD2
 1006 FORMAT(1H0,39X,8HDEL10  =,F8.2,3X,2A4,37X,8HDEL10D =,F8.2,3X,2A4/
     1 10X,30HUNSPRUNG MASS POSITIONS        ,2A4,F8.2,3X,2A4,
     2  7X,30HUNSPRUNG MASS VELOCITIES       ,2A4,F8.2,3X,2A4    )
      IF(ISUS.EQ.1) GO TO 103
      UMP1 = UPR
      TD1(1) = DEG(1)
      TD1(2) = DEG(2)
      TD2(1) = DPS(1)
      TD2(2) = DPS(2)
      UMP = PHIRO
      UMV = PHIROD
      GO TO 104
  103 UMP1 = UD4
      TD1(1) = DINCH(1)
      TD1(2) = DINCH(2)
      TD2(1) = DIPS(1)
      TD2(2) = DIPS(2)
      UMP = DEL40
      UMV = DEL40D
  104 WRITE(6,1007) DEL30,DINCH,DEL30D,DIPS,UMP1,UDE,UMP,TD1,UMP1,UVE,
     1             UMV,TD2,PSIF10,DEG,PSIFD0,DPS
 1007 FORMAT(1H ,39X,8HDEL30  =,F8.2,3X,2A4,37X,8HDEL30D =,F8.2,3X,2A4/
     1 40X,2A4,F8.2,3X,2A4,37X,2A4,F8.2,3X,2A4  /
     2 10X,38HSTEER ANGLE                     PSIFI0 =,F8.2,3X,2A4,
     3  7X,38HSTEER VELOCITY                  PSIFD0 =,F8.2,3X,2A4    )
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)
      WRITE(6,2001) XMS,PS2P1,     A,DINCH,
     1             XMUF,PS2PI,    B,DINCH,
     2             XMUR,PS2PI,    ZF,DINCH
 2001 FORMAT(1H0,
     1  9X,37HSPRUNG MASS                     XMS   =,F10.3,1X,3A4,
     2  5X,32HFRONT WHEEL X LOCATION    A    =,    F10.3,1X,2A4  /
     3 10X,37HFRONT UNSPRUNG MASS             XMUF  =,F10.3,1X,3A4,
     4  5X,32HREAR WHEEL X LOCATION     B    =,    F10.3,1X,2A4  /
     5 10X,37HREAR UNSPRUNG MASS              XMUR  =,F10.3,1X,3A4,
     6  5X,32HFRONT WHEEL Z LOCATION    ZF   =,    F10.3,1X,2A4  )
      TD1(1) = TNU2(1)
      TD1(2) = TNU2(2)
```

```
      IF(ISUS.EQ.2) GO TO 201
      GO TO 202
  201 TD1(1) = DINCH(1)
      TD1(2) = DINCH(2)
  202 CONTINUE
      WRITE(6,2002) XIX, PS2I,      ZR  ,DINCH,
     1              XIY, PS2I,      TF  ,DINCH,
     2              XIZ, PS2I,      TR  ,DINCH,
     3              XIXZ,PS2I,    RHOF,TD1
 2002 FORMAT(1H ,
     1  9X,37HX MOMENT OF INERTIA                XIX   =,F10.3,1X,3A4 ,
     2  5X,32HREAR WHEEL Z LOCATION       ZR    =,     F10.3,1X,2A4 /
     3 10X,37HY MOMENT OF INERTIA                XIY   =,F10.3,1X,3A4 ,
     4  5X,32HFRONT WHEEL TRACK           TF    =,     F10.3,1X,2A4 /
     5 10X,37HZ MOMENT OF INERTIA                XIZ   =,F10.3,1X,3A4 ,
     6  5X,32HREAR WHEEL TRACK            TR    =,     F10.3,1X,2A4 /
     7 10X,37HXZ PRODUCT OF INERTIA              XIXZ  =,F10.3,1X,3A4 ,
     8  5X,32HFRONT ROLL AXIS             RHOF =,      F10.3,1X,2A4 )
      DO 203 K=1,3
      T3D1(K) = TNU3(K)
  203 T3D2(K) = TNU3(K)
      DO 204 K=1,2
      TD1(K) = TNU2(K)
      TD2(K) = TNU2(K)
  204 TD3(K) = TNU2(K)
      IF(ISUS.EQ.1) GO TO 206
      DO 205 K=1,2
      T3D2(K) = PS2I(K)
      TD1(K) = DINCH(K)
  205 TD3(K) = DINCH(K)
      T3D2(3) = PS2I(3)
  206 IF(ISUS.NE.2) GO TO 208
      DO 207 K=1,2
      T3D1(K) = PS2I(K)
  207 TD2(K) = DINCH(K)
      T3D1(3) = PS2I(3)
  208 WRITE(6,2003) XIF, T3D1, RHO, TD1,
     1              XIR, T3D2, TSF, TD2,
     2              G   ,DIPS2,  TS,TD3
 2003 FORMAT(1H ,
     1  9X,37HFRONT AXLE MOMENT OF INERTIA  XIF   =,F10.3,1X,3A4 ,
     2  5X,32HREAR ROLL AXIS              RHO  =,     F10.3,1X,2A4 /
     3 10X,37HREAR AXLE MOMENT OF INERTIA   XIR   =,F10.3,1X,3A4 ,
     4  5X,32HFRONT SPRING TRACK          TSF  =,     F10.3,1X,2A4 /
     5 10X,37HGRAVITY                       G     =,F10.3,1X,3A4 ,
     6  5X,32HREAR SPRING TRACK           TS   =,     F10.3,1X,2A4 )
      DO 209 K=1,3
      T3D1(K) = TNU3(K)
      T3D2(K) = TNU3(K)
      IF(K.EQ.3) GO TO 209
      TD1(K) = TNU2(K)
```

```
      TD2(K) = TNU2(K)
      TD3(K) = TNU2(K)
  209 CONTINUE
      IF(ISUS.EQ.1) GO TO 211
      TD1(1) = RAPRA(1)
      TD1(2) = RAPRA(2)
      GO TO 213
  211 DO 212 K=1,3
      T3D1(K) = RPI2(K)
      T3D2(K) = RPI3(K)
      IF(K.EQ.3) GO TO 212
      TD2(K) = RADS(K)
      TD3(K) = RPI(K)
  212 CONTINUE
  213 WRITE(6,2004) X1,          DINCH,     RF,         PIPR,
     5               Y1,          DINCH,     RR,         PIPR,
     2               Z1,          DINCH,     AKRS,       TD1,
     3               X2,          DINCH,     AKDS,       TD2,
     4               Y2,          DINCH,     AKDS1,      TD3,
     5               Z2,          DINCH,     AKDS2,      T3D1,
     6                                       AKDS3,      T3D2
 2004 FORMAT(1H0,39X,7HX1     = ,F10.2,1X,2A4  ,
     1  9X,32HFRONT AUX ROLL STIFFNESS  RF    =,F10.2,1X,3A4 /
     2 10X,37HACCELEROMETER 1 POSITION        Y1     =,F10.2,1X,2A4 ,
     3  9X,32HREAR AUX ROLL STIFFNESS   RR    =,     F10.2,1X,3A4 /
     4 40X,7HZ1     =,F10.2,1X,2A4 ,
     5  9X,32HREAR ROLL-STEER COEF.     AKRS =,     F10.4,1X,2A4 /
     6 40X,7HX2     =,F10.2,1X,2A4 ,35X,6HAKDS =,F10.3,1X,2A4      /
     7 10X,37HACCELEROMETER 2 POSITION        Y2     =,F10.2,1X,2A4 ,
     8  9X,32HREAR DEFL-STEER COEFS.    AKDS1=,     F10.3,1X,2A4 /
     9 40X,7HZ2     =,F10.2,1X,2A4,35X,6HAKDS2=,F10.3,1X,3A4       /
     A101X,6HAKDS3=,F10.3,1X,3A4  )
      WRITE(6,2005) XIPS,PS2I,CPSP,PDI,EPSPS,RAPS,AKPS,PIPR,
     1              OMGPS,RADS,XPS,DINCH
 2005 FORMAT(1H0,15X,29HS T E E R I N G   S Y S T E M       /
     1 10X,31HMOMENT OF INERTIA        XIPS  =,F10.3,1X,3A4 /
     2 10X,31HCOULOMB FRICTION TORQUE CPSP  =,F10.3,1X,2A4 /
     3 10X,31HFRICTION LAG             EPSP  =,F10.3,1X,2A4 /
     4 10X,31HANGULAR STOP RATE        AKPS  =,F10.3,1X,3A4 /
     5 10X,31HANGULAR STOP POSITION    OMGPS =,F10.3,1X,2A4 /
     6 10X,31HPNEUMATIC TRAIL          XPS   =,F10.3,1X,2A4 )
      WRITE(6,2009) FIWJF,PS2I,FIWJR,PS2I,ARBRF,ARBRR,FIDJF,PS2I,
     1              FIDJR,PS2I
 2009 FORMAT(1H0,13X,28HD R I V E   L I N E   D A T A         /
     A 10X,33HFRONT WHEEL SPIN INERTIA  FIWJF =,F8.3,1X,3A4      /
     B 10X,33HREAR WHEEL SPIN INERTIA   FIWJR =,F8.3,1X,3A4      /
     C 10X,33HFRONT AXLE RATIO          ARBRF =F8.3,             /
     D 10X,33HREAR AXLE RATIO           ARBRR =F8.3,             /
     E 10X,33HFRONT DRIVELINE INERTIA   FIDJF =,F8.3,1X,3A4      /
     F 10X,33HREAD DRIVELINE INERTIA    FIDJR =,F8.3,1X,3A4      )
      WRITE(6,2006) AKF,       PPI,       AKR,        PPI,
```

```
     1               AKFC,     PPI,      AKRC,     PPI,                    IDOT3050
     2               AKFCP,    PPI3,     AKRCP,    PPI3                    IDOT3060
2006 FORMAT(1H0,36X,16HFRONT SUSPENSION,20X,15HREAR SUSPENSION  //         IDOT3070
     1 10X,41HSUSPENSION RATE                     AKF     =,F10.3,1X,2A4,  IDOT3080
     2  9X,8HAKR     =,F10.3,1X,2A4  /                                     IDOT3090
     3 10X,41HCOMPRESSION STOP COEFS.             AKFC    =,F10.3,1X,2A4,  IDOT3100
     4  9X,8HAKRC    =,F10.3,1X,2A4  /                                     IDOT3110
     5 43X,8HAKFCP   =,F10.3,1X,2A4,9X,8HAKRCP   =,F10.3,1X,2A4   )        IDOT3120
      WRITE(6,2007) AKFE,     PPI,      AKRE,     PPI,                     IDOT3130
     1               AKFEP,    PPI3,     AKREP,    PPI3,                   IDOT3140
     2               OMEGFC,   DINCH,    OMEGRC,   DINCH,                  IDOT3150
     3               OMEGFE,   DINCH,    OMEGRE,   DINCH                   IDOT3160
2007 FORMAT(1H ,                                                           IDOT3170
     1  9X,41HEXTENSION STOP COEFS.               AKFE    =,F10.3,1X,2A4,  IDOT3180
     2  9X, 8HAKRE    =,F10.3,1X,2A4  /                                    IDOT3190
     3 43X, 8HAKFEP   =,F10.3,1X,2A4,9X,8HAKREP  =,F10.3,1X,2A4  /         IDOT3200
     4 10X,41HCOMPRESSION STOP LOCATION           OMEGFC =,F10.3,1X,2A4,   IDOT3210
     5  9X, 8HOMEGRC =,F10.3,1X,2A4  /                                     IDOT3220
     6 10X,41HEXTENSION STOP LOCATION             OMEGFE =,F10.3,1X,2A4,   IDOT3230
     7  9X, 8HOMEGRE =,F10.3,1X,2A4  )                                     IDOT3240
      WRITE(6,2008) XLAMF,              XLAMR,                             IDOT3250
     1               CF,       PSPI,     CR,       PSPI,                   IDOT3260
     2               CFP,      PD,       CRP,      PD,                     IDOT3270
     3               EPSF,     DIPS,     EPSR,     DIPS                    IDOT3280
2008 FORMAT(1H ,                                                           IDOT3290
     1  9X,41HSTOP ENERGY DISSIPATION FACTOR      XLAMF   =,F10.3,         IDOT3300
     2 18X, 8HXLAMR  =,F10.3  /                                            IDOT3310
     3 10X,41HVISCOUS DAMPING COEF.               CF      =,F10.3,1X,3A4,  IDOT3320
     4  5X, 8HCR     =,F10.3,1X,3A4  /                                     IDOT3330
     5 10X,41HCOULOMB FRICTION                    CFP     =,F10.3,1X,1A4,  IDOT3340
     6 13X, 8HCRP    =,F10.3,1X,1A4  /                                     IDOT3350
     7 10X,41HFRICTION LAG                        EPSF    =,F10.3,1X,2A4,  IDOT3360
     8  9X, 8HEPSR   =,F10.3,1X,2A4  )                                     IDOT3370
      IF(ISUS.EQ.2.AND.IAPFR(1)+IAPFR(2).EQ.0) GO TO 404                   IDOT3380
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                       IDOT3390
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),             IDOT3400
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)                               IDOT3410
      IF(ISUS.EQ.2) GO TO 301                                              IDOT3420
      DO 306 K=1,2                                                         IDOT3430
      TD1(K) = DINCH(K)                                                    IDOT3440
 306  TD2(K) = DEG(K)                                                      IDOT3450
      IF(ISUS.EQ.1) GO TO 308                                              IDOT3460
      DO 307 K=1,2                                                         IDOT3470
      TD1(K) = TNU2(K)                                                     IDOT3480
 307  TD2(K) = TNU2(K)                                                     IDOT3490
 308  WRITE(6,3001) DINCH,DEG,TD1,TD2,DINCH,DINCH,TD1,TD1                  IDOT3500
3001 FORMAT(1H0,                                                           IDOT3510
     A 10X,18HFRONT WHEEL CAMBER, 8X,17HREAR WHEEL CAMBER,                 IDOT3520
     B  6X,23HFRONT HALF-TRACK CHANGE, 4X,22HREAR HALF-TRACK CHANGE  /     IDOT3530
     C 18X,2HVS,24X,2HVS,24X,2HVS,24X,2HVS  /                              IDOT3540
     D  9X,21HSUSPENSION DEFLECTION, 5X,21HSUSPENSION DEFLECTION,          IDOT3550
```

```
     E  5X,21HSUSPENSION DEFLECTION, 5X,21HSUSPENSION DEFLECTION     //
     F 12X,15HDELTAF      PHIC,11X,16HDELTAR      PHIRC ,
     G 10X,15HDELTAF      DTHF,11X,15HDELTAR      DTHR  /
     H 12X,2A4,2X,2A4,8X,2A4,2X,2A4,8X,2A4,2X,2A4,8X,2A4,2X,2A4       )
      Y = DELB
      DO 302 I=1,NDEL
      TTARG(I) = Y
      Y = Y+DDEL
  302 CONTINUE
      WRITE(6,3002) (TTARG(I),PHIC(I),TTARG(I),PHIRC(I),
     1               TTARG(I),DTHF(I),TTARG(I),DTHR(I),I=1,NDEL)
 3002 FORMAT(1H0,4(8X,F8.2,2X,F8.2)/(1X,4(8X,F8.2,2X,F8.2))        )
  301 CONTINUE
      IF( IAPFR(1) .EQ.0 .AND. IAPFR(2) .EQ.0) GO TO 400
      WRITE(6,4004)
 4004 FORMAT(1H0,8X,48HANTI-PITCH TABLES FOR CIRCUMFERENTIAL TIRE FORCE
     1     // 9X,11HFRONT WHEEL,5X,3HAPF,5X,10HREAR WHEEL,5X,3HAPR /
     2     9X,11HDEFL. - IN.,3X,8HLB/LB-FT,5X,10HDEFL.- IN.,
     3     3X,8HLB/LB-FT   / )
      FDEF = DAPFB
      RDEF = DAPRB
      MAP = NAPF
      IF(NAPF.NE.NAPR) MAP = MINO(NAPF,NAPR)
      IF(NAPF.EQ.0) GO TO 402
      IF(NAPR.EQ.0) GO TO 406
      DO 401 I=1,MAP
      WRITE(6,4005) FDEF,APF(I),RDEF,APR(I)
 4005 FORMAT(5X,4(5X,F8.4))
      FDEF = FDEF+DDAPF
  401 RDEF = RDEF+DDAPR
      IF(NAPF.EQ.NAPR) GO TO 404
      IF(NAPR.GT.NAPF) GO TO 402
  406 MAP1 = MAP+1
      DO 403 I=MAP1,NAPF
      WRITE(6,4006) FDEF,APF(I)
 4006 FORMAT(5X,2(5X,F8.4))
  403 FDEF = FDEF+DDAPF
      GO TO 404
  402 MAP1 = MAP+1
      DO 405 I=MAP1,NAPR
      WRITE(6,4007) RDEF,APR(I)
 4007 FORMAT(31X,2(5X,F8.4))
  405 RDEF = RDEF+DDAPR
      GO TO 404
  400 WRITE(6,4008)
 4008 FORMAT(21HONO ANTI-PITCH TABLES)
  404 CONTINUE
      IF(TINCR.EQ.0.0.AND.IDRVER.EQ.0) GO TO 408
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)
```

```
      NTPR = 0
      TTARG(1) = 0.0
      IF(TINCR.EQ.0.0) GO TO 304
      NTPR = (TE-TB)/TINCR + 1.5
      Y = TB
      DO 305 I=1,NTPR
      TTARG(I) = Y
      Y = Y+TINCR
  305 CONTINUE
      WRITE(6,3003)
 3003 FORMAT(1H0,//56X,21HDRIVER CONTROL TABLES  //
     1 4(32H       T       PSIF       TQF       TQR )       /
     2 4(32H     SEC       DEG     LB-FT    LB-FT)       /)
C        NTPR4 IS NUMBER OF LINES FOR TABLES IN FOUR GROUPS PER LINE
      NNADD = 0
      IF((MOD(NTPR,4)).NE.0) NNADD=1
      NTPR4 = NTPR/4 + NNADD
      NTPR43 = 3*NTPR4
      DO 303 J=1,NTPR4
      I1 = J
      I4 = MINO(NTPR ,I1+NTPR43)
      WRITE(6,3004)((TTARG(II),PSIF(II),TQF(II),TQR(II)),II=I1,I4,NTPR4)
 3004 FORMAT(1X,4(F8.3,F8.3,F8.1,F8.1)  )
  303 CONTINUE
  304 CONTINUE
      IF(IDRVER.NE.0) CALL DRIVID
  408 CONTINUE
C
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1      (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2      (GHED(I),I=1,10),(SHED(I),I=1,10)
      WRITE(6,4001)
 4001 FORMAT(1H0,60X,17HT I R E   D A T A   /
     A 54X,2HRF,10X,2HLF,10X,2HRR,10X,2HLR  )
      WRITE(6,4002) AKT,PPI,SIGT,DINCH,XLAMT,A0,A1,A2,A3,A4
 4002 FORMAT(1H0,
     A  9X,39HTIRE LINEAR SPRING RATE          AKT     =,4(F10.3,2X),2A4 /
     B 10X,39HDEFL. FOR INCREASED RATE         SIGT    =,4(F10.3,2X),2A4 /
     C 10X,39HSPRING RATE INCREASING FACTOR    XLAMT   =,4(F10.3,2X)     /
     D 41X, 8HA0      =,4(F10.3,2X)  /
     E 41X, 8HA1      =,4(F10.3,2X)  /
     F 10X,39HSIDE FORCE COEFFICIENTS          A2      =,4(F10.3,2X)     /
     G 41X, 8HA3      =,4(F10.3,2X)  /
     H 41X, 8HA4      =,4(F10.3,2X)   )
      WRITE(6,4003) OMEGT,RW,DINCH,XMUM,CT,PD,RRMC,AMU(1)
 4003 FORMAT(1H ,
     I 10X,39HTIRE OVERLOAD FACTOR             OMEGT   =,4(F10.3,2X)     /
     J 10X,39HTIRE UNDEFLECTED RADIUS          RW      =,4(F10.3,2X),2A4 /
     K 10X,39HTIRE MEASUREMENT FRICTION        XMUM    =,4(F10.3,2X)     /
     P 10X,39HCIRCUM. FORCE STIFFNESS          CT      =,4(F10.3,2X),A4 /
     L 10X,39HROLLING RESISTANCE MOMENT COEF RRMC    =,4(F10.3,2X),
```

```
    M   2X,8HLB-IN/LB    /                                             IDOT4580
    N 10X,34HNOMINAL GROUND FRICTION COEF AMU =,F10.4         /)       IDOT4590
     CALL IDOUTA(HED,DADE)                                             IDOT4600
     IF(INDCRB.NE.1) GO TO 702                                         IDOT4610
     WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                    IDOT4620
    1   (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),            IDOT4630
    2    (GHED(I),I=1,10),(SHED(I),I=1,10)                             IDOT4640
     WRITE(6,6010)                                                     IDOT4650
6010 FORMAT(1H0,22X,17HC U R B   D A T A    //                         IDOT4660
    A 10X,54HCURB SLOPE CHANGE      ELEVATION AT      CURB FACE ANGLE  / IDOT4670
    B 10X,34H LATERAL POSITION      SLOPE CHANGE     /                 IDOT4680
    C 18X,6HINCHES,11X,6HINCHES,11X,7HDEGREES      //  )               IDOT4690
     WRITE(6,6011) YC1P,                     PHIC1,                    IDOT4700
    A              YC2P,      ZC2P,          PHIC2,                    IDOT4710
    B              YC3P,      ZC3P,          PHIC3,                    IDOT4720
    C              YC4P,      ZC4P,          PHIC4,                    IDOT4730
    D              YC5P,      ZC5P,          PHIC5,                    IDOT4740
    E              YC6P,      ZC6P,          PHIC6,                    IDOT4750
    F              NCRBSL,    AMUC                                     IDOT4760
6011 FORMAT(1H ,                                                       IDOT4770
    A 11X,6HYC1P =,F9.2,23X,7HPHIC1 =,F9.2,  /                         IDOT4780
    B 12X,6HYC2P =,F9.2,3X,6HZC2P =,F9.2,5X,7HPHIC2 =,F9.2,    /       IDOT4790
    C 12X,6HYC3P =,F9.2,3X,6HZC3P =,F9.2,5X,7HPHIC3 =,F9.2,    /       IDOT4800
    D 12X,6HYC4P =,F9.2,3X,6HZC4P =,F9.2,5X,7HPHIC4 =,F9.2,    /       IDOT4810
    E 12X,6HYC5P =,F9.2,3X,6HZC5P =,F9.2,5X,7HPHIC5 =,F9.2,    /       IDOT4820
    F 12X,6HYC6P =,F9.2,3X,6HZC6P =,F9.2,5X,7HPHIC6 =,F9.2,    /       IDOT4830
    G 12X,8HNCRBSL =,I4   /                                            IDOT4840
    F 10X,43HCURB FRICTION COEFFICIENT FACTOR     AMUC  =,F8.3       ) IDOT4850
     WRITE(6,7001) RWHJB,RWHJE,DRWHJ                                   IDOT4860
7001 FORMAT(37H0WHEEL RADIUS-RADIAL SPRING FOR TABLE  /17H RWHJB(BEGIN)IDOT4870
    1   =,F8.3,7H INCHES / 17H RWHJE(END)      =,F8.3,5H   '' /,       IDOT4880
    2 17H DRWHJ(INCRE.)  =,F8.3,5H   '' )                              IDOT4890
     NFJP = 0                                                          IDOT4900
     IF(DRWHJ.EQ.0.0) GO TO 702                                        IDOT4910
     NFJP = (RWHJE-RWHJB)/DRWHJ + 1.2                                  IDOT4920
     IF(NFJP.LE.0) GO TO 702                                           IDOT4930
     Y = RWHJB                                                         IDOT4940
     DO 701 I=1,NFJP                                                   IDOT4950
     TTARG(I) = Y                                                      IDOT4960
     Y = Y + DRWHJ                                                     IDOT4970
 701 CONTINUE                                                          IDOT4980
     WRITE(6,7002)                                                     IDOT4990
7002 FORMAT(//1H ,3X,5HRW-HJ,6X,4HFJP.,6X,4HFJP.,6X,4HFJP.,6X,4HFJP. / IDOT5000
    A        5X,3HIN.,7X,4HLBS.,6X,4HLBS.,6X,4HLBS.,6X,4HLBS.    /     IDOT5010
    B        16X,2HRF,8X,2HLF,8X,2HRR,8X,2HLR    /  )                  IDOT5020
     DO 703 J=1,NFJP                                                   IDOT5030
     WRITE(6,7003) TTARG(J),(FJP(J,II),II=1,4)                         IDOT5040
7003 FORMAT(1H ,G9.3,4G10.3)                                           IDOT5050
 703 CONTINUE                                                          IDOT5060
 702 CONTINUE                                                          IDOT5070
     IF(NZTAB.EQ.0) GO TO 700                                          IDOT5080
```

```
      DO 601 I=1,50                                                     IDOT5090
  601 TTARG(I) = 0.0                                                    IDOT5100
      DO 602 I=1,10                                                     IDOT5110
  602 NTARG(I) = 0                                                      IDOT5120
      DO 603 I=1,NZTAB                                                  IDOT5130
      TTARG(I) =       XB(I)                                            IDOT5140
      TTARG(5 + I)= XE(I)                                               IDOT5150
      TTARG(10 + I) = XINCR(I)                                          IDOT5160
      TTARG(15 + I) = YB(I)                                             IDOT5170
      TTARG(20 + I) = YE(I)                                             IDOT5180
      TTARG(25 + I) = YINCR(I)                                          IDOT5190
      TTARG(30 + I) = AMUG(I)                                           IDOT5200
      NTARG(I)      = NBX(I)                                            IDOT5210
      NTARG(5 + I) = NBY(I)                                             IDOT5220
  603 CONTINUE                                                          IDOT5230
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                    IDOT5240
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),          IDOT5250
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)                            IDOT5260
      WRITE(6,6001)                                                     IDOT5270
 6001 FORMAT(//1H ,26X,25HTERRAIN TABLE ARGUMENTS        )              IDOT5280
      WRITE(6,6002)  (TTARG(I),I=1,5),                                  IDOT5290
     1               (TTARG(I),I=6,10),                                 IDOT5300
     2               (TTARG(I),I=11,14),ZERO,                           IDOT5310
     3               (TTARG(I),I=16,20),                                IDOT5320
     4               (TTARG(I),I=21,25),                                IDOT5330
     5               (TTARG(I),I=26,29),ZERO,                           IDOT5340
     6               (NTARG(I),I=1,5),                                  IDOT5350
     7               (NTARG(I),I=6,10),                                 IDOT5360
     8               (TTARG(I),I=31,35),                                IDOT5370
     9                NZTAB                                             IDOT5380
 6002 FORMAT(1H0,25X,11H XB(BEGIN)=,5F12.3,7H INCHES   /                IDOT5390
     A          26X,11H XE(END)  =,5F12.3,5H   ''     /                 IDOT5400
     B          26X,11H X(INCR)  =,5F12.3,5H   ''     /                 IDOT5410
     C          26X,11H YB(BEGIN)=,5F12.3,5H   ''     /                 IDOT5420
     D          26X,11H YE(END)  =,5F12.3,5H   ''     /                 IDOT5430
     E          26X,11H Y(INCR)  =,5F12.3,5H   ''     /                 IDOT5440
     F         25X,12HNO.X BOUNDS=,I8,4I12            /                 IDOT5450
     G         25X,12HNO.Y BOUNDS=,I8,4I12            /                 IDOT5460
     H          26X,11H AMUG     =,5F12.3             /                 IDOT5470
     I          25X,18HNO.TERRAIN TABLES=,I4          )                 IDOT5480
      IF(NZ5.EQ.0) GO TO 600                                            IDOT5490
      WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),                    IDOT5500
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),          IDOT5510
     2     (GHED(I),I=1,10),(SHED(I),I=1,10)                            IDOT5520
      NX5 = NX(NZTAB)                                                   IDOT5530
      NY5 = NY(NZTAB)                                                   IDOT5540
      WRITE(6,6004) NX5, (XXZGP5(I),I=1,NX5)                            IDOT5550
 6004 FORMAT(66H0 ARGUMENTS FOR TERRAIN TABLE WITH VARYING INCREMENTS (LIDOT5560
     1AST TABLE) /10H NO.OF X =, I3,2X,9H, X(ZGP)=, 12F9.3/24X,9F9.3)   IDOT5570
      WRITE(6,6003) NY5, (YYZGP5(I),I=1,NY5)                            IDOT5580
 6003 FORMAT(10H0NO.OF Y =, I3,2X,9H, Y(ZGP)=,12F9.3/24X, 9F9.3)        IDOT5590
```

```
C
  600 IF(NZTAB) 604,700,604
  604 WRITE(6,1000) (HED(I),I=1,18),DADE(1),DADE(2),
     1      (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2      (GHED(I),I=1,10),(SHED(I),I=1,10)
      LINES =3
      DO 614 I=1,NZTAB
      NNBX = NBX(I)
      NNBY = NBY(I)
      NNX = NX(I)
      NNY = NY(I)
      LINES = LINES + 9 + (NNY+1)*(NNX/7 + 2)
      IF(I.EQ.1) GO TO 606
      IF(LINES .LT.55) GO TO 606
      WRITE(6,1000) (HED(N),N=1,18),DADE(1),DADE(2),
     1      (VHED(N),N=1,10),(THED(N),N=1,10),(CHED(N),N=1,10),
     2      (GHED(N),N=1,10),(SHED(N),N=1,10)
      LINES =3
  606 WRITE(6,6005) I,AMUG(I),(XBDRY(J,I),J=1,NNBX)
 6005 FORMAT(19H0 TERRAIN TABLE NO. ,I3,  20X, 6H AMUG=, F13.5//
     X 1X,16H  X  BOUNDARIES=,4F13.5)
      WRITE(6,6006) (PSBDRO(J,I),J=1,NNBX)
 6006 FORMAT(1X,16H PSI BOUNDARIES=,4F13.5)
      WRITE(6,6007) (YBDRY(J,I),J=1,NNBY)
 6007 FORMAT(1X,16H  Y  BOUNDARIES=,2F13.5)
      IF( I.EQ.NZTAB .AND. NZ5.NE.0) GO TO 607
      ANAME = CON1
      Y= XB(I)
      YYY = XINCR(I)
      DO 605 J=1,NNX
      TXARG(J) = Y
      Y = Y + YYY
  605 CONTINUE
      Y = YB(I)
      YYY = YINCR(I)
      DO 609 J=1,NNY
      TYARG(J) = Y
      Y = Y + YYY
  609 CONTINUE
      GO TO 610
  607 ANAME = VARI
      DO 611 J=1,NNX
  611 TXARG(J) = XXZGP5(J)
      DO 612 J=1,NNY
  612 TYARG(J) = YYZGP5(J)
  610 WRITE(6,6008)ANAME,(TXARG(J),J=1,NNX)
 6008 FORMAT(1H0,A4,17H. INCREMENTS    X=,2X,7F13.5/26X,7F13.5/28X,7F13.5
     X )
      DO 613 II=1,NNY
      WRITE(6,6009) TYARG(II),(ZGP(JJ,II,I),JJ=1,NNX)
 6009 FORMAT(/2X,3H Y=,F13.5, 6X,7F13.5/26X,7F13.5/26X,7F13.5)
```

```
  613 CONTINUE                                                    IDOT6110
  614 CONTINUE                                                    IDOT6120
C                                                                 IDOT6130
  700 CONTINUE                                                    IDOT6140
C                                                                 IDOT6150
      WRITE(6,9007)                                               IDOT6160
 9007 FORMAT(1H1)                                                 IDOT6170
      RETURN                                                      IDOT6180
      END                                                         IDOT6190
```

```
      SUBROUTINE IDOUTA (HDD,DATE)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),
     1             NPAGE(20)
      COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),
     A                XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),
     B                XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),
     C                XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4
      EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)
      EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)
      DIMENSION HDD(36), DATE(3)
      DIMENSION NNN(6)
      DATA NNN/1,2,3,4,5,6/
      DATA JTO/6/
C
 1010 FORMAT(1H1,19X,18A4,20X,2A4 / 5X,3(10A4) / 5X,2(10A4) / )
      WRITE(JTO,1000)
 1000 FORMAT(1H0)
      WRITE(JTO,1040)
 1040 FORMAT( 1H0,
     A 61H SPEED   | LOAD   |  LATERAL FRICTION COEF.    | PEAK LONG. FRIC
     B 61HTION COEF. | SLIDING LONG. FRICTION     | SLIP AT PEAK LONG. F
     C  7HRICT. |    / 1X,
     D 61H IN/SEC  |  LBS   |           XMUY             |           XMUXP
     E 61H              |   COEF.     XMUXS             |           SLIPP
     F  7H        |    / 1X,
     G 61H                  |  RF      LF       RR      LR  |  RF      LF
     H 61HRR      LR  |  RF      LF       RR      LR  |  RF      LF      RR
     I  7H  LR   |    /   )
      DO 141 I1 = 1,NXUGMU
      DO 141 I2 = 1,NXFRCP
      WRITE(JTO,1044) XXUGMU(I1),XXFRCP(I2),(XMUMAT(I2,I1,I3),I3=1,4),
     A             (XMXPMT(I2,I1,I3),I3=1,4),(XMXSMT(I2,I1,I3),I3=1,4),
     B             (SLIPMT(I2,I1,I3),I3=1,4)
 1044 FORMAT(1H ,2(F7.2,1X),16(2X,F5.3)  )
  141 CONTINUE
C
      WRITE(JTO,1010) (HDD(I),I=1,18),DATE(1),DATE(2),
     1       (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),
     2       (CHED(I),I=1,10),(SHED(I),I=1,10)
      WRITE(JTO,1060) NTTS,TAUA,IBTYP(1), PZERO(1),IBTYP(2),PZERO(2),AK1
     1 ,CONE
 1060 FORMAT(1H0,35X,38H BRAKING SYSTEMS WITH COMPUTED TORQUES  ///
```

```
    140H NTTS,(MAX 101),NO.OF PC,TS,TR ENTRIES =,I10, 20X,                IDTA0500
    240H TAUA   AMBIENT TEMPERATURE, DEG.F      =,F10.1 /                 IDTA0510
    327H IBTYP(1)  FRONT BRAKE TYPE,12X,1H=,I10,20X,                      IDTA0520
    440H PO(1)  FRONT,BRAKE PUSH OUT PRES.,PSIG=,F10.2 /                  IDTA0530
    527H IBTYP(2)  REAR    ''     '' ,12X,1H=,I10,20X,                    IDTA0540
    640H PO(2)  REAR,  ''        ''     ''     '' =, F10.2 /              IDTA0550
    740H K1   SLOPE PR/PF (P1.LT.PF.LE.P2)       =, F10.4,20X,            IDTA0560
    840H C1     COEFFICIENT FOR RESISTING FORCE=, 1PE12.5)                IDTA0570
     WRITE(JTO,1062) AK2,CTWO,PONE,CTHREE,PTWO,ZETAB                      IDTA0580
 1062 FORMAT(28H K2       ''   ''    (PF.GT.P2),11X,1H=,F10.4, 20X,       IDTA0590
    140H C2           ''               ''        '' =, 1PE12.5  /         IDTA0600
    240H P1   BRAKE PRES. FOR PR/PF CHANGE,PSIG=,  E12.5,18X,             IDTA0610
    340H C3           ''               ''        '' =,  E12.5  /          IDTA0620
    440H P2      ''   ''        ''      ''       '' =,  E12.5,  10X,      IDTA0630
    559H ZETAB,THRESHOLD WHEEL SPEED TO LIMIT BRAKE TORQUE,RAD/SEC=,      IDTA0640
    6   OPF8.2 )                                                          IDTA0650
     WRITE(JTO,1000)                                                      IDTA0660
     WRITE(JTO,1064) BTT,ETT,DTT                                          IDTA0670
 1064 FORMAT(32H0 TABLES PC,TS,TR VARY WITH TIME     /                    IDTA0680
    1       17H  TT(J),J=1,NTTS,1X,F4.1,1X,2HTO,2X,F5.1,1X,               IDTA0690
    2      8HSECS. IN,1X,F4.2,1X,14HSEC INCREMENTS  )                     IDTA0700
     WRITE(JTO,1066)                                                      IDTA0710
 1066 FORMAT(// 3(10X,6H BRAKE,6X,22H THROTTLE TRANSMISSION)/3(9X,11H MAIDTA0720
    1STER CYL,     2X,18H SETTING     RATIO, 4X )/3(4X, 4H SEC,1X, 12H IDTA0730
    2PRES.(PSIG),23X)/  3(5X,3H TT, 4X,3H PC,10X, 3H TS, 8X,3H TR,5X ))IDTA0740
     NNADD = 0                                                            IDTA0750
     IF(MOD(NTTS,3).NE.0) NNADD =1                                        IDTA0760
     NT3 = NTTS/3 + NNADD                                                 IDTA0770
     NT33 = 2*NT3                                                         IDTA0780
     DO 165 J = 1,NT3                                                     IDTA0790
     I1 = J                                                               IDTA0800
     I3 = MIN0(NTTS,I1+NT33 )                                             IDTA0810
     WRITE(JTO,1068) (TT(II),TPC(II),TTS(II),TTR(II) ,II= I1,I3,NT3)      IDTA0820
 1068 FORMAT(3(1X,F7.2, F10.0, F12.3, F10.2,4X))                          IDTA0830
  165 CONTINUE                                                            IDTA0840
C                                                                         IDTA0850
     WRITE(JTO,1010) (HDD(I),I=1,18),DATE(1),DATE(2),                     IDTA0860
    1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),             IDTA0870
    2     (GHED(I),I=1,10),(SHED(I),I=1,10)                               IDTA0880
     WRITE(JTO,1000)                                                      IDTA0890
     WRITE(JTO,1070)                                                      IDTA0900
 1070 FORMAT( 50H0ENGINE RPM       ENGINE TORQUE      ENGINE TORQUE,55X,IDTA0910
    1 17H BRAKE PARAMETERS / 15X,19H WIDE OPEN THROTTLE,2X,16H CLOSED TIDTA0920
    2HROTTLE,53X,  6H FRONT,11X,5H REAR   / 21X,6H LB-FT,13X,6H LB-FT, IDTA0930
    357X,8H GN(I,1),10X, 8H GN(I,2) )                                     IDTA0940
     WRITE(JTO,1011)                                                      IDTA0950
 1011 FORMAT(1H )                                                         IDTA0960
     DO 170 I=1,NRPM                                                      IDTA0970
     WRITE(JTO,1072) TRPME(I), TWOT(I), TCT(I), I, GN(I,1), GN(I,2)       IDTA0980
 1072 FORMAT( 1X,F10.0,8X,F10.0,  9X, F10.0,46X, I3,2X,1PE15.5,2X,E15.5)IDTA0990
  170 CONTINUE                                                            IDTA1000
```

```
      NRPM1 = NRPM+1
      DO 174 I=NRPM1,16
      WRITE(JTO,1074) I, GN(I,1), GN(I,2)
 1074 FORMAT( 94X, I3,2X, 1PE15.5,2X,E15.5)
  174 CONTINUE
      WRITE(JTO,1000)
      WRITE(JTO,1000)
      WRITE(JTO,1080) (NNN(I),TAUO(I),I=1,4)
 1080 FORMAT( 34H0 INITIAL BRAKE TEMPERATURE, DEG.F  /  7H  TAUO(,I2,
     116H), RIGHT FRONT =, F8.2 / 7H  TAUO(,I2,16H), LEFT    ''   =,F8.2/
     2 7H  TAUO(, I2, 16H), RIGHT REAR  =,F8.2 / 7H  TAUO(,I2, 16H), LEF
     3T    ''   =, F8.2)
      WRITE(JTO,1000)
      WRITE(JTO,1082) BTLF,ETLF,DTLF
 1082 FORMAT(34H0 TABLE LF VARIES WITH TEMPERATURE    /
     1       17H TTAU(J),J=1,NTLF,1X,F4.1,10H(DEG.F) TO,1X,F6.1,
     2       10H(DEG.F) IN,1X,F4.1,1X,17HDEGREE INCREMENTS    )
      WRITE(JTO,1011)
      WRITE(JTO,1084)
 1084 FORMAT(1H0,48X,31H BRAKE LINING FADE COEFFICIENTS / 4(3X,5H TTAU,
     1 7X,3H LF,15X) / 4(3X,6H DEG.F,24X) )
      WRITE(JTO,1011)
      NNADD = 0
      IF(MOD(NTLF,4).NE.0) NNADD = 1
      NT4 = NTLF/4 + NNADD
      NT34 = 3*NT4
      DO 186 J=1,NT4
      I1 = J
      I2 = MINO(NTLF,I1+NT34)
      WRITE(JTO,1086) (TTAU(II),TLF(II),II=I1,I2,NT4)
 1086 FORMAT(  4(1X,  F7.1, F12.4, 13X))
  186 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INITEQ
C               HVOSM-VD2 VERSION
C               REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1          GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2           TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3           BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4           XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4          ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5           XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7           CTXG(4;,UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
```

```
     1            (PSI1(1),PSI1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
      DATA RPD/.01745329/
      RHF = 0.0
      RHR = 0.0
      IF(ISUS.NE.1) RHR = RHO
      IF(ISUS.EQ.2) RHF = RHOF
      CTHO = COS(THETAO*RPD)
      STHO = SIN(THETAO*RPD)
      SIR = XMS*A*G*CTHO/(A+B)
      SIF = XMS*G*CTHO-SIR
      DTF = (SIF/CTHO+XMUF*G)*0.5/AKT(1)
      DTR = (SIR/CTHO+XMUR*G)*0.5/AKT(3)
      SD1 = 0.5*(B*XMS*G/(A+B)-SIF)/AKF
      SD3 = 0.5*(A*XMS*G/(A+B)-SIR)/AKR
      HCG = -ZCOP
      ZF = (HCG+A*STHO-RW(1)+DTF)/CTHO-RHF-SD1
      ZR = (HCG-B*STHO-RW(3)+DTR)/CTHO-RHR-SD3
      FR(1) = AKT(1)*DTF
      FR(2) = FR(1)
      FR(3) = AKT(3)*DTR
      FR(4) = FR(3)
      HI(1) = RW(1)-DTF
      HI(2) = HI(1)
      HI(3) = RW(3)-DTR
      HI(4) = HI(3)
      RETURN
      END
```

```
      SUBROUTINE INPUT
C              HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION CARDIM(20),ICARD(300),DUM(18)
      DATA NBLKS/6/
      WRITE(6,1000)
 1000 FORMAT(1H1)
C          SET INPUT CARD COUNTER
      NC = 0
C        REWIND UNIT 2
      REWIND 2
C         READ A CARD
    1 READ(5,5000,END=999) (CARDIM(K),K=1,18),NSEQ,NCARD
 5000 FORMAT(18A4,2I4)
C        OUTPUT CARD IMAGE
      WRITE(2,2001) (CARDIM(K),K=1,18),NSEQ,NCARD
      WRITE(6,6000) (CARDIM(K),K=1,18),NSEQ,NCARD
 6000 FORMAT(1H ,18A4,2I4)
      IF(NCARD.GE.9999) GO TO 20
      IF(NCARD.LE.0) GO TO 90
      IF(NSEQ.GT.0) GO TO 1
      NC = NC+1
      ICARD(NC) = NCARD
      GO TO 1
   20 REWIND 2
C        TEST FOR AT LEAST ONE CARD OTHER THAN 9999
      IF(NC.LE.0) GO TO 91
C        SET COUNTER TO PROCESS ALL BLOCK NUMBERED CARDS
      IC = 1
C       DETERMINE CARD FORMAT AND TRANSFER TO PROPER CARD BLOCK
C        SUBROUTINE TO STORE DATA
   21 NBLK = ICARD(IC)/100
      NBCRD = ICARD(IC)-NBLK*100
C        FORMAT TEST
      IF(NBCRD.EQ.0) GO TO 22
C        NUMERIC INPUT
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD
 2000 FORMAT(9F8.0,2I4)
      GO TO 23
   22 CONTINUE
C        ALPHANUMERIC INPUT
      READ(2,2001) (DUM(K),K=1,18),NSEQ,NCARD
 2001 FORMAT(18A4,2I4)
C        BRANCH TO PROPER SUBROUTINE TO STORE INPUT
   23 IF(NBLK .LE.0) GO TO 92
      IF(NBLK.GT.NBLKS) GO TO 93
      GO TO(100,200,300,400,500,600),NBLK
C  PRINT ERROR MESSAGE HERE ?
  100 NERR = 0
```

```
      CALL BLK01(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0500
C        TEST FOR ERROR                                                 INPT0510
      IF(NERR.EQ.0) GO TO 30                                            INPT0520
      GO TO 94                                                          INPT0530
  200 CALL BLK02(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0540
      IF(NERR.EQ.0) GO TO 30                                            INPT0550
      GO TO 94                                                          INPT0560
  300 NERR = 0                                                          INPT0570
      CALL BLK03(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0580
      IF(NERR.EQ.0) GO TO 30                                            INPT0590
      GO TO 94                                                          INPT0600
  400 NERR = 0                                                          INPT0610
      CALL BLK04(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0620
      IF(NERR.EQ.0) GO TO 30                                            INPT0630
      GO TO 94                                                          INPT0640
  500 NERR = 0                                                          INPT0650
      CALL BLK05(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0660
      IF(NERR.EQ.0) GO TO 30                                            INPT0670
      GO TO 94                                                          INPT0680
  600 NERR = 0                                                          INPT0690
      CALL BLK06(NBLK,NBCRD,NSEQ,NCARD,DUM,NERR)                        INPT0700
      IF(NERR.EQ.0) GO TO 30                                            INPT0710
      GO TO 94                                                          INPT0720
   30 CONTINUE                                                          INPT0730
C        TEST IF ALL CARDS ARE READ                                     INPT0740
      IC = IC+1                                                         INPT0750
      IF(IC.GT.NC) GO TO 40                                             INPT0760
C        GET NEXT CARD FROM UNIT 2                                      INPT0770
      GO TO 21                                                          INPT0780
   40 CONTINUE                                                          INPT0790
C        SEARCH FOR END OF DATA                                         INPT0800
      READ(2,2001) (DUM(K),K=1,18),NSEQ,NCARD                           INPT0810
      IF(NCARD.NE.9999) GO TO 95                                        INPT0820
      GO TO 50                                                          INPT0830
  999 WRITE(6,6001) NCARD                                               INPT0840
 6001 FORMAT(56H UNEXPECTED END OF FILE ENCOUNTERED IN STMT NO. 1 OF SUBINPT0850
     1      34HROUTINE INPUT.  LAST CARD READ WAS   ,I4  )              INPT0860
      GO TO 49                                                          INPT0870
   90 WRITE(6,6002)                                                     INPT0880
 6002 FORMAT(56H A CARD NUMBERED LESS THAN OF EQUAL TO ZERO WAS ENCOUNTEINPT0890
     1      50HRED IN SUBROUTINE INPUT.  CARD IMAGE PRINTED ABOVE  )    INPT0900
      GO TO 49                                                          INPT0910
   91 WRITE(6,6003)                                                     INPT0920
 6003 FORMAT(33H THE NUMBER OF CARDS READ IS ZERO    )                  INPT0930
      GO TO 49                                                          INPT0940
   92 WRITE(6,6004)                                                     INPT0950
 6004 FORMAT(56H A BLOCK NUMBER OF LESS THAN OF EQUAL TO ZERO HAS BEEN OINPT0960
     1      7HBTAINED   )                                               INPT0970
      GO TO 49                                                          INPT0980
   93 WRITE(6,6005)                                                     INPT0990
 6005 FORMAT(56H A BLOCK NUMBER LARGER THAN THE ALLOWED NUMBER HAS BEEN INPT1000
```

```
    1         8HOBTAINED   )                                             INPT1010
      GO TO 49                                                           INPT1020
   94 WRITE(6,6006) NBLK,NBCRD,NSEQ,NCARD,NERR                           INPT1030
 6006 FORMAT(56H AN ERROR HAS OCCURRED IN STORING INPUT VALUES IN ONE OFINPT1040
    1        23H THE BLKXX SUBROUTINES.     /                            INPT1050
    2        39H THE CALLING ARGUMENTS FROM INPUT ARE :    /             INPT1060
    3         7H NBLK =,I4,2X,7HNBCRD =,I4,2X,6HNSEQ =,I4,2X,7HNCARD =,  INPT1070
    4         I4,2X,6HNERR =,I4     )                                    INPT1080
      GO TO 49                                                           INPT1090
   95 WRITE(6,6007)                                                      INPT1100
 6007 FORMAT(56H AN EXPECTED 9999 CARD HAS NOT BEEN ENCOUNTERED AFTER STINPT1110
    1        20H MT NO. 40 IN INPUT.      )                              INPT1120
   49 STOP                                                               INPT1130
   50 RETURN                                                             INPT1140
      END                                                                INPT1150
```

```
      SUBROUTINE INTPR (F,IW,XMA,ALP,NM,NA,XM,AX,ANS,ERR,ND1,ND2)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION FX(3),XMA(ND1),ALP(1),F(ND1,ND2,1)
C         NOTE VARIABLE DIMENSIONS
      REAL INTERF
       INTERF(X,X1,X2,Y1,Y2) = Y2 - (X2-X)*(Y2-Y1)/(X2-X1)
      ERR=0.0
       NA1=NA
    1  XX = AX
        X = XM
       NM1 = NM
       DO 8 I=2,NM1
       IF(XMA(I) - X) 8,6,6
    6  I1=I
       GO TO 10
    8  CONTINUE
       I1=NM1
   10 IF( XMA(I1-1)-XMA(I1))21,20,21
   20 ERR=1.0
      GO TO 23
   21 DO 14  I=2,NA1
       IF(ALP(I)-XX) 14,12,12
   12  J1=I
       GO TO 16
   14  CONTINUE
       J1=NA1
   16  JJ=J1-2
      IF( ALP(J1-1) -ALP(J1))22,20,22
   22 DO  19  L=1,2
       J=JJ+ L
   19  FX(L) = INTERF(X,XMA(I1-1),XMA(I1),F(I1-1,J,IW),F(I1,J,IW))
       FX(3) = INTERF(XX,ALP(J1-1),ALP(J1),FX(1),FX(2))
       ANS = FX(3)
   23 RETURN
      END
```

```
      SUBROUTINE INTRPL(TABLE,XMIN,XMAX,DX,X,Y)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
C     QUADRATIC INTERPOLATION SUBROUTINE INTRPL,ADDITIONAL ENTRY INTRPC
      DIMENSION TABLE(1)
      ENTRY INTRPC(TABLE,XMIN,XMAX,DX,X,Y,SLOPE)
    1 XLK = AMIN1(X,XMAX)
      XLK = AMAX1(XLK,XMIN)
      N1 = (XLK-XMIN)/DX+1.2
      N2 = N1+1
      NT = (XMAX-XMIN)/DX+1.2
      NO = N1-1
    2 IF(NO.GT.0) GO TO 3
      NO = N1
      N1 = N2
      N2 = N1+1
    3 IF(N2.LE.NT) GO TO 4
      N2 = N1
      N1 = NO
      NO = N1-1
    4 XXX = FLOAT(NO)*DX+XMIN
      DX2 = DX**2
      A = (TABLE(N2)-2.0*TABLE(N1)+TABLE(NO))/(2.0*DX2)
      B = (TABLE(N1)-TABLE(NO))/DX-A*(2.0*XXX-DX)
      C = TABLE(N1)-(A*XXX**2+B*XXX)
      Y = (A*XLK+B)*XLK+C
      SLOPE = 2.0 * A * XLK + B
      RETURN
      END
```

```
      SUBROUTINE INTRP5(INDX)                                           INT50010
C               HVOSM-VD2 VERSION                                       INT50020
C               REVISED OCTOBER 1975   CALSPAN CORPORATION              INT50030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    INT50040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      INT50050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      INT50060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF, INT50070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  INT50080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     INT50090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   INT50100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   INT50110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    INT50120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  INT50130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),INT50140
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  INT50150
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,INT50160
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   INT50170
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),INT50180
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     INT50190
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  INT50200
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  INT50210
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     INT50220
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      INT50230
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),INT50240
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)INT50250
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      INT50260
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     INT50270
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   INT50280
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  INT50290
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       INT50300
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   INT50310
     1            (PSII(1),PSI1)                                        INT50320
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,INT50330
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          INT50340
     2            TFO2,TI2,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  INT50350
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    INT50360
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    INT50370
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPINT50380
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     INT50390
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    INT50400
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  INT50410
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   INT50420
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  INT50430
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  INT50440
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,INT50450
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2INT50460
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,INT50470
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           INT50480
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 INT50490
```

```
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
C
C     NWHEEL = INDX
C     IXBDRY = 0
C     IYBDRY = 0
C     XLCEPT=0.0
C     XRCEPT=0.0
C     I5 = 0
C
      ERR = 0.0
   10 XXX = XP(INDX)
      YYY = YP(INDX)
      IT = 0
      DO 11 I=1,NZTAB
      IF( XB(I).EQ.XE(I) .OR. YB(I).EQ.YE(I)) GO TO 11
      IF(XXX.GE.XB(I).AND.XXX.LE.XE(I).AND.YYY.GE.YB(I).AND.YYY.LE.YE(I)
     X ) IT =I
   11 CONTINUE
      IF(IT.NE.0) GO TO 15
   13 ZPGI(INDX)= 0.0
      THGI(INDX)= 0.0
      PHGI(INDX)= 0.0
      XMUGI(INDX) = AMU(INDX)
      RETURN
C                ITV = 1 IDENTIFIES THE VARIABLE INCREMENT TABLE HERE.
   15 ITV = 0
      IF( IT.EQ. NZTAB .AND. NZ5.NE.0) ITV = 1
      XMUGI(INDX) = AMUG(IT)*AMU(INDX)
      XBT = XB(IT)
      XET = XE(IT)
      YBT = YB(IT)
      NXT = NX(IT)
      NYT = NY(IT)
      NBXT= NBX(IT)
      NBYT= NBY(IT)
      IF(ITV.GE.1) GO TO 20
C       TABLES WITH CONSTANT INCREMENT
      XINCRT  =  XINCR(IT)
      YINCRT  =  YINCR(IT)
      IX =(XXX-XBT)/XINCRT + 1.0
      IY =(YYY-YBT)/YINCRT + 1.0
      XX1 = XBT + FLOAT(IX-1)*XINCRT
      XX2 = XX1 + XINCRT
```

```
      YY1 = YBT + FLOAT(IY-1)*YINCRT
      YY3 = YY1 + YINCRT
      GO TO 40
C       IX IS LOW INDEX FOR X , IY IS LOW INDEX FOR Y
C       FLOAT(IX-1) IS   COUNT   OF INCREMENTS
C       VARIABLE INCREMENT TABLE (ARGUMENTS GIVEN, XXZPG5(21),YYZPG5(21)
   20 DO 22 I=2,NXT
      IF (XXZGP5(I) - XXX) 22,21,21
   21 IXX = I
      GO TO 25
   22 CONTINUE
      IXX = NXT
   25 IX = IXX-1
      IF (XXZGP5(IX) -XXZGP5(IXX)) 27,26,27
   26 ERR = 1.0
      GO TO 13
   27 DO 29 I=2,NYT
      IF (YYZGP5(I) - YYY) 29,28,28
   28 IYY= I
      GO TO 30
   29 CONTINUE
      IYY= NYT
   30 IY = IYY - 1
      IF(YYZGP5(IY)    - YYZGP5(IYY))35,26,35
   35 XX1 = XXZGP5(IX)
      XX2 = XXZGP5(IXX)
      YY1 = YYZGP5(IY)
      YY3 = YYZGP5(IYY)
      XINCRT = XX2 - XX1
      YINCRT = YY3 - YYI
   40 XX3 = XX1
      XX4 = XX2
C       SEARCH FOR Y BOUNDARIES IN THIS MESH.  Y BOUNDARIES HAVE CONSTAN
      IF (NBYT .EQ. 0) GO TO 54
      JJ = 0
      DO 41 I= 1,NBYT
      IF(YY1.GE.YBDRY(I,IT).OR. YBDRY(I,IT).GT.YY3) GO TO 41
      JJ = I
C
C     IYBDRY = I
C
      GO TO 42
   41 CONTINUE
   42 IF(JJ.EQ.0) GO TO 54
      IF(YYY.GE.YBDRY(JJ,IT))GO TO 50
      YY3 = YY1
      IF(ITV.GE.1) GO TO 44
   43 YY1 = YY3 - YINCRT
      IY = IY -1
      GO TO 54
   44 YY1 = YYZGP5(IY-1)
```

```
      IY = IY-1
      YINCRT = YY3 - YY1
      GO TO 54
   50 YY1 = YY3
      IF(ITV.GE.1) GO TO 52
   51 YY3 = YY1 + YINCRT
      IY = IY + 1
      GO TO 54
   52 YY3 = YYZGP5(IY +2)
      IY = IY + 1
      YINCRT =  YY3 - YY1
   54 YY2 = YY1
      YY4 = YY3
C        SEARCH FOR SLANTED BOUNDARIES
      IF (NBXT .EQ. 0) GO TO 61
      II = 0
      DO 60 I=1,NBXT
      XBDRT = XBDRY(I,IT)
C                 PI AND 2.*PI ARE SINGULARITIES FOR COTAN
      IF( AMOD(PSBDRY(I,IT) , PI)  .EQ. 0.0) GO TO 60
      CTNPSB = COTAN(PSBDRY(I,IT))
      XLCEPT =  XBDRT + (YY1-YBT)*CTNPSB
      XRCEPT =  XBDRT + (YY3-YBT)*CTNPSB
      II= I
      IF( XX1.LE.XLCEPT .AND. XLCEPT.LE.XX2) GO TO 80
      IF(XLCEPT.LE.XX1 .AND. XRCEPT.GT.XX3) GO TO 80
      IF(XLCEPT.GE.XX2 .AND. XRCEPT.LT.XX4) GO TO 80
   60 CONTINUE
C        NO SLANT BOUNDARY IN THIS MESH
   61 XXMXX1 = XXX-XX1
      YYMYY1 = YYY-YY1
      ZPG1 = ZGP(IX ,IY ,IT)
      ZPG2 = ZGP(IX+1 ,IY ,IT)
      ZPG3 = ZGP(IX ,IY+1 ,IT)
      ZPG4 = ZGP(IX +1,IY+1 ,IT)
      ZZ1 =  ZPG1 + XXMXX1* (ZPG2-ZPG1)/XINCRT
      ZZ2 =  ZPG3 + XXMXX1* (ZPG4-ZPG3)/XINCRT
      ZPGI(INDX) = ZZ1 + YYMYY1*(ZZ2-ZZ1)/YINCRT
      THG1 = ATAN2 ((ZPG1-ZPG2),XINCRT)
      THG3 = ATAN2 ((ZPG3-ZPG4),XINCRT)
      THGI(INDX) =  THG1 + YYMYY1 *(THG3- THG1)/YINCRT
      IF(YYMYY1) 62,65,63
   62 ZPH1 = ZZ1
      ZYINCR =    -YYMYY1
      GO TO 67
   63 IF( YY3- YYY) 65,64,65
   64 PHGI(INDX) = ATAN2( (ZZ2 - ZZ1)/YINCRT, COS(THG1))
C     NOTE THG1, AS ROLL REFERENCE IS TO POINT 1 HERE
      GO TO 68
   65 ZPH1 = ZZ2
      ZYINCR =      YY3 - YYY
```

```
   67 PHGI(INDX) = ATAN2( (ZPH1-ZPGI(INDX))/ZYINCR, COS(THGI(INDX)))
   68 RETURN
C  68 ZPGI10 = ZPGI(INDX)
C     THGI10 = THGI(INDX)/RAD
C     PHGI10 = PHGI(INDX)/RAD
C3000 RETURN
C        SLANT BOUNDARY IN THIS MESH
   80 ZXINCR = XINCRT
C
C     IXBDRY = II
C
      IF( XXX .GT.(XBDRT + (YYY - YBT)* CTNPSB) ) GO TO 140
C
C       WHEEL HAS NOT CROSSED THE SLANT BOUNDARY, STEP BACK ON X ,PERHAP
C     INDEX FOR HIGH GRID X IS IX+1, (XX2 AT IX+1,IY),(XX4 AT IX+1,IY+1)
C        COUNT OF CONSTANT INCREMENTS FOR XX2 IS IX
      NXW = IX
      IF(ITV.GE.1) GO TO 93
   83 XX2W = XX2 + XINCRT
      DO 85 I=1,NXW
      XX2W = XX2W - XINCRT
      IF( XX2W .GE. XLCEPT) GO TO 85
      IX2W= IX +2 - I
      GO TO 90
   85 CONTINUE
      IX2W = 2
      XX2W = XBT+ XINCRT
   90 XX1 = XX2W - XINCRT
      XX4W = XX4 + XINCRT
      DO 92  I=1,NXW
      XX4W = XX4W- XINCRT
      IF(XX4W .GE. XRCEPT) GO TO 92
      IX4W = IX  +2 - I
      GO TO  100
   92 CONTINUE
      IX4W = 2
      XX4W = XBT+ XINCRT
      GO TO 100
   93 NXW5 = IX
      NXWW = IX +2
      DO 95  I= 1,NXW5
      IX2W = NXWW - I
      IF( XXZGP5(IX2W) .LT. XLCEPT) GO TO 96
   95 CONTINUE
      IX2W = 2
   96 XX2W = XXZGP5(IX2W)
      XX1    = XXZGP5(IX2W-1)
      XINCRT =  XX2W - XX1
      DO 97  I= 1,NXW5
      IX4W = NXWW - I
      IF(XXZGP5(IX4W) .LT. XRCEPT) GO TO 98
```

```
 97 CONTINUE
    IX4W =2
 98 XX4W = XXZGP5(IX4W)
100 IX1W = IX2W -1
    IX3W = IX4W -1
    IF(IX1W  -   IX3W) 104,103,104
103 IX = IX1W
    GO TO 61
104 ZPG1 = ZGP(IX1W, IY, IT)
    ZPG2 = ZGP(IX2W, IY, IT)
    ZPG3 = ZGP(IX3W, IY+1, IT)
    ZPG4 = ZGP(IX4W, IY+1, IT)
    IF(IX2W - IX3W) 106,107,110
106 ZPH1  =  ZGP(IX3W-1 ,IY+1, IT)
C         ZPH1 IS POINT FIVE HERE
    GO TO 108
107 ZPH1    = ZPG3
108 ZPH2    = ZPG2
    ZTH1    = ZPG3
    ZTH2    = ZPG4
    IF(ITV.GE.1)ZXINCR = XXZGP5(IX4W) - XXZGP5(IX3W)
    GO TO 115
110 IF(IX1W - IX4W) 115,112,111
111 I5 =MAXO(IX1W-1 ,1)
C        ZPH2  IS POINT FIVE HERE
    ZPH2     = ZGP(I5, IY,IT)
    GO TO 113
112 ZPH2     = ZPG1
113 ZPH1     = ZPG4
    ZTH1     = ZPG1
    ZTH2     = ZPG2
    IF(ITV.GE.1)ZXINCR = XXZGP5(IX2W) - XXZGP5(IX1W)
115 ZZZ1  =  ZPG2
    XXMXX1 = XXX - XX2W
    YYMYY1 = YYY - YY2
    GO TO 180
C         WHEEL HAS CROSSED SLANT BOUNDARY. STEP AHEAD ON X, PERHAPS.
140 NXW = NXT -1
    KXW = IX
    IF(ITV.GE.1) GO TO 153
143 XX1W = XX1- XINCRT
    DO 145 I = KXW,NXW
    XX1W = XX1W + XINCRT
    IF(XX1W .LT. XLCEPT) GO TO 145
    IX1W = 1
    GO TO 150
145 CONTINUE
    XX1W = XET- XINCRT
    IX1W = NXW
150 XX1 = XX1W
    XX3W = XX3 - XINCRT
```

```
      DO 152 I= KXW ,NXW                                                INT53050
      XX3W = XX3W + XINCRT                                              INT53060
      IF(XX3W .LT. XRCEPT) GO TO 152                                    INT53070
      IX3W = I                                                          INT53080
      GO TO 160                                                         INT53090
  152 CONTINUE                                                          INT53100
      IX3W = NXW                                                        INT53110
      XX3W = XET- XINCRT                                                INT53120
      GO TO 160                                                         INT53130
  153 DO 155  I = KXW, NXW                                              INT53140
      IF(XXZGP5(I) .LT. XLCEPT) GO TO 155                               INT53150
      IX1W = I                                                          INT53160
      GO TO 156                                                         INT53170
  155 CONTINUE                                                          INT53180
      IX1W = NXW                                                        INT53190
  156 XX1W = XXZGP5(IX1W)                                               INT53200
      XX1  = XX1W                                                       INT53210
      XINCRT =  XXZGP5(IX1W + 1) - XX1                                  INT53220
      DO 157 I= KXW ,NXW                                                INT53230
      IF(XXZGP5(I) .LT. XRCEPT) GO TO 157                               INT53240
      IX3W = I                                                          INT53250
      GO TO 158                                                         INT53260
  157 CONTINUE                                                          INT53270
      IX3W = NXW                                                        INT53280
  158 XX3W = XXZGP5(IX3W)                                               INT53290
  160 IX2W = IX1W + 1                                                   INT53300
      IX4W = IX3W + 1                                                   INT53310
      IF(IX1W - IX3W) 164,163,164                                       INT53320
  163 IX = IX1W                                                         INT53330
      GO TO 61                                                          INT53340
  164 ZPG1 = ZGP(IX1W,IY,IT)                                            INT53350
      ZPG2 = ZGP(IX2W,IY,IT)                                            INT53360
      ZPG3 = ZGP(IX3W,IY+1,IT)                                          INT53370
      ZPG4 = ZGP(IX4W,IY+1,IT)                                          INT53380
      IF(IX2W - IX3W) 166,167,170                                       INT53390
  166 ZPH2 = ZGP( IX2W+1,IY,IT)                                         INT53400
C        ZPH2 IS POINT FIVE HERE                                        INT53410
      GO TO 168                                                         INT53420
  167 ZPH2 = ZPG2                                                       INT53430
  168 ZPH1 = ZPG3                                                       INT53440
      ZTH1 = ZPG1                                                       INT53450
      ZTH2 = ZPG2                                                       INT53460
      IF(ITV.GE.1)ZXINCR = XXZGP5(IX2W) -XXZGP5(IX1W)                   INT53470
      GO TO 175                                                         INT53480
  170 IF(IX1W - IX4W) 175,172,171                                       INT53490
  171 I5 = MIN0(IX4W+1,NXT)                                             INT53500
C        ZPH1 IS POINT FIVE HERE                                        INT53510
      ZPH1 = ZGP(I5,IY+1,IT)                                            INT53520
      GO TO 173                                                         INT53530
  172 ZPH1 = ZPG4                                                       INT53540
  173 ZPH2 = ZPG1                                                       INT53550
```

```
      ZTH1 = ZPG3
      ZTH2 = ZPG4
      IF(ITV.GE.1)ZXINCR= XXZGP5(IX4W) - XXZGP5(IX3W)
  175 ZZZ1 = ZPG1
      XXMXX1 = XXX - XX1
      YYMYY1 = YYY - YY1
  180 ZTH12 =  ZTH1-ZTH2
      TTANTH =  ZTH12/ZXINCR
      THGI(INDX) = ATAN2( ZTH12    , ZXINCR)
      TCOSTH = COS(THGI(INDX))
      PFAC = (ZPH1 - ZPH2)/YINCRT
      PHGI(INDX) = ATAN2( PFAC, TCOSTH)
      IF(TCOSTH) 186,185,186
  185 TTANPH = 0.0
      GO TO 187
  186 TTANPH =  PFAC/TCOSTH
  187 ZPGI(INDX) = ZZZ1 + YYMYY1*TCOSTH*TTANPH - XXMXX1* TTANTH
      RETURN
      END
```

```
      SUBROUTINE MATRIX                                                 MTRX0010
C           HVOSM-VD2 VERSION                                           MTRX0020
C           REVISED OCTOBER 1975   CALSPAN CORPORATION                  MTRX0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    MTRX0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      MTRX0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      MTRX0060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,MTRX0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  MTRX0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     MTRX0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   MTRX0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   MTRX0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    MTRX0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  MTRX0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),MTRX0140
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  MTRX0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             MTRX0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))MTRX0170
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),MTRX0180
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        MTRX0190
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       MTRX0200
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         MTRX0210
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           MTRX0220
     6           (PSIFID,VAR(22))                                       MTRX0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      MTRX0240
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))MTRX0250
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       MTRX0260
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     MTRX0270
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    MTRX0280
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          MTRX0290
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      MTRX0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         MTRX0310
     1           (DER(10),DPHIFD)                                       MTRX0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       MTRX0330
     1           (DER(14),DDEL4D)                                       MTRX0340
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,MTRX0350
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   MTRX0360
     2            CGYW(4),ZPGI(4),THG1(4),PHG1(4),CPG(4),SPG(4),CTG(4),MTRX0370
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     MTRX0380
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XCPP(4),  MTRX0390
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  MTRX0400
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     MTRX0410
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      MTRX0420
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),MTRX0430
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)MTRX0440
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      MTRX0450
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     MTRX0460
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   MTRX0470
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  MTRX0480
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       MTRX0490
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),    MTRX0500
     1            (PSII(1),PSI1)                                         MTRX0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTRX0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          MTRX0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  MTRX0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    MTRX0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    MTRX0560
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTRX0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     MTRX0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    MTRX0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  MTRX0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   MTRX0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   MTRX0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  MTRX0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTRX0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTRX0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTRX0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           MTRX0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  MTRX0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        MTRX0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                 MTRX0700
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,        MTRX0710
     2              SLOPE1,SLOPE2,XTRA(300)                              MTRX0720
      DIMENSION UI(4),VI(4),WI(4)                                        MTRX0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       MTRX0740
      DIMENSION APITCH(4)                                                MTRX0750
      EQUIVALENCE (APITCH(1),APTCH1)                                     MTRX0760
    1 CALL CLEAR (DMATX,DMATX(10,11))                                    MTRX0770
      DMATX(1,1) = SUMM                                                  MTRX0780
      DMATX(1,5) = GAM2                                                  MTRX0790
      DMATX(1,6) = RHOMUR*PHIR                                           MTRX0800
    2 DMATX(2,2) = SUMM                                                  MTRX0810
       DMATX(2,4) = -GAM2                                                MTRX0820
      DMATX(2,6) = GAM1                                                  MTRX0830
      DMATX(2,10) = -RHOMUR                                              MTRX0840
    3 DMATX(3,3) = XMS                                                   MTRX0850
    4 DMATX(4,2) = -GAM3                                                 MTRX0860
      DMATX(4,4) = XIX+XIXP                                              MTRX0870
      DMATX(4,6) = -XIXZ-XIXZP                                           MTRX0880
      DMATX(4,10) = RHOMUR*ZRD3                                          MTRX0890
    5 DMATX(5,1) = GAM2                                                  MTRX0900
      DMATX(5,5) = XIY+XIYP                                              MTRX0910
      DMATX(5,6) = -XIYZP                                                MTRX0920
    6 DMATX(6,1) = DMATX(1,6)                                            MTRX0930
      DMATX(6,2) = GAM1                                                  MTRX0940
      DMATX(6,4) = DMATX(4,6)+BROMUR                                     MTRX0950
      DMATX(6,5) = -XIYZP                                                MTRX0960
      DMATX(6,6) = XIZR+XIZP                                             MTRX0970
      DMATX(6,10) = BROMUR                                               MTRX0980
    7 DMATX(7,3) = XMUFO2                                                MTRX0990
      DMATX(7,4) = XMTFO4                                                MTRX1000
```

```
      DMATX(7,5) = -AXMF02
      DMATX(7,7) = XMUF02
    8 DMATX(8,3) = XMUF02
      DMATX(8,4) = -XMTF04
      DMATX(8,5) = -AXMF02
      DMATX(8,8) = XMUF02
    9 DMATX(9,3) = XMUR
      DMATX(9,4) = -DMATX(1,6)
      DMATX(9,5) = BMUR
      DMATX(9,9) = XMUR
      DMATX(9,10) = DMATX(9,4)
   10 DMATX(10,2) = -RHOMUR
      DMATX(10,3) = DMATX(9,4)
      DMATX(10,4) = XIR+ZRD3R*RHOMUR
      DMATX(10,5) = -BROMUR*PHIR
      DMATX(10,6) = BROMUR
      DMATX(10,9) = DMATX(9,4)
      DMATX(10,10) = RHMR2I
   11 GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
   12 DMATX(1,11) = SUMM*(VR-WQ-GSTH)-GAM2*PR+RHOMUR*PHIR*PQ+GAM1*(Q2+
     1              R2)-GAM6*Q-2.0*RHOMUR*RPHRD+SFXS+SFXU
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)+GAM6*P-GAM1*PQ-GAM2*QR-RHOMUR*
     1              PHIR*(P2+R2+PHIRD2)+SFYS+SFYU
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = GAM3*(UR-WP-GCTSP)+(XIXZ+XIXZP)*PQ-GAM7*P+(XIY-XIZ+
     1              XIXP)*QR-GAM4*(P2+R2)+RHOMUR*PHIR*ZRD3*PHIRD2+SNPS+
     2              SNPU
      DMATX(5,11) = XIXZ*(R2-P2)+(XIZ-XIX-XIYP)*PR+GAM2*(VR-WQ-GSTH)-
     1              (GAM7+2.*RHO*TG61)*Q+(XIXZP-BROMUR)*(Q2+R2)-
     2              XIYZP*PQ-2.0*XMUR*ZRD3R*RHO*RPHRD+SNTS+SNTU
   13 DMATX(6,11) = (XIX-XIY-GAM5)*PQ-(XIXZ+XIXZP-BROMUR)*QR+GAM8*Q+
     1              XIYZP*PR+GAM9*P+RHOMUR*PHIR*(VR-WQ-2.0*RHO*RPHRD-B*
     2              (Q2-P2-PHIRD2)-GSTH)+GAM1*(WP-UR+GCTSP)+SNPSS+SNPSU
      DMATX(7,11) = XMUF02*(UQ-VP-A*PR-TF02*QR+(ZF+DEL1)*(P2+Q2)+GCTCP)+
     1              FZU(1)+SI(1)
      DMATX(8,11) = XMUF02*(UQ-VP-A*PR+TF02*QR+(ZF+DEL2)*(P2+Q2)+GCTCP)+
     1              FZU(2)+SI(2)
      DMATX(9,11) = XMUR*(UQ-VP+RHO*PHIRD2+2.0*P*RHO*PHIRD+B*PR+RHO*PHIR
     1              *QR+ZRD3R*(P2+Q2)+GCTCP)+FZU(3)+FZU(4)+SI(3)+SI(4)
   14 DMATX(10,11) = RHOMUR*(UR-WP-2.0*P*(DEL3D-RHO*PHIR*PHIRD)-B*PQ+
     1               RHO*PHIR*(P2+R2)+ZRD3R*QR-GCTH*SIN(PHIT+PHIR))+
     2         PHIR*RHOMUR*(VP-UQ-2.0*P*RHO*PHIRD-B*PR-RHO*PHIR*QR-
     3               ZRD3R*(P2+Q2))-XIR*PHIR*(R2-Q2)-XIR*QR+
     4               SNPR
      RETURN
      END
```

```
      SUBROUTINE MTRXIR                                                 MTXI0010
C              HVOSM-VD2 VERSION                                        MTXI0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               MTXI0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    MTXI0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      MTXI0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      MTXI0060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,MTXI0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  MTXI0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     MTXI0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   MTXI0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   MTXI0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    MTXI0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  MTXI0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),MTXI0140
     1         XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  MTXI0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             MTXI0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))MTXI0170
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),MTXI0180
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        MTXI0190
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       MTXI0200
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         MTXI0210
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           MTXI0220
     6           (PSIFID,VAR(22))                                       MTXI0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      MTXI0240
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))MTXI0250
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       MTXI0260
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     MTXI0270
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    MTXI0280
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          MTXI0290
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      MTXI0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         MTXI0310
     1           (DER(10),DPHIFD)                                       MTXI0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       MTXI0330
     1           (DER(14),DDEL4D)                                       MTXI0340
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,MTXI0350
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   MTXI0360
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),MTXI0370
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     MTXI0380
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  MTXI0390
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  MTXI0400
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     MTXI0410
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      MTXI0420
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),MTXI0430
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)MTXI0440
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      MTXI0450
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),     MTXI0460
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),   MTXI0470
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                  MTXI0480
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       MTXI0490
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   MTXI0500
     1            (PSII(1),PSI1)                                        MTXI0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTXI0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         MTXI0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, MTXI0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   MTXI0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   MTXI0560
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTXI0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    MTXI0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   MTXI0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, MTXI0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  MTXI0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  MTXI0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, MTXI0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTXI0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTXI0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTXI0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          MTXI0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 MTXI0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       MTXI0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                MTXI0700
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       MTXI0710
     2              SLOPE1,SLOPE2,XTRA(300)                             MTXI0720
      DIMENSION UI(4),VI(4),WI(4)                                       MTXI0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                      MTXI0740
      DIMENSION APITCH(4)                                               MTXI0750
      EQUIVALENCE (APITCH(1),APTCH1)                                    MTXI0760
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,      MTXI0770
     1                RHF2MF,RF2MF1,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     MTXI0780
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    MTXI0790
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    MTXI0800
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        MTXI0810
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     MTXI0820
C                                                                       MTXI0830
      CALL CLEAR(DMATX(1,1),DMATX(10,11))                               MTXI0840
      DMATX(1,1) = SUMM                                                 MTXI0850
      DMATX(1,5) = GAM2                                                 MTXI0860
      DMATX(2,2) = SUMM                                                 MTXI0870
      DMATX(2,4) = -GAM2                                                MTXI0880
      DMATX(2,6) = GAM1                                                 MTXI0890
      DMATX(3,3) = XMS                                                  MTXI0900
      DMATX(4,2) = -GAM2                                                MTXI0910
      DMATX(4,4) = XIX+XIXP                                             MTXI0920
      DMATX(4,6) = -XIXZ-XIXZP                                          MTXI0930
      DMATX(5,1) = GAM2                                                 MTXI0940
      DMATX(5,5) = XIY+XIYP                                             MTXI0950
      DMATX(5,6) = -XIYZP                                               MTXI0960
      DMATX(6,2) = GAM1                                                 MTXI0970
      DMATX(6,4) = -XIXZ-XIXZP                                          MTXI0980
      DMATX(6,5) = -XIYZP                                               MTXI0990
      DMATX(6,6) = XIZ+XIZP                                             MTXI1000
```

```
      DMATX(7,3) = XMUFO2
      DMATX(7,4) = XMTFO4
      DMATX(7,5) = -AXMFO2
      DMATX(7,7) = XMUFO2
      DMATX(8,3) = XMUFO2
      DMATX(8,4) = -XMTFO4
      DMATX(8,5) = -AXMFO2
      DMATX(8,8) = XMUFO2
      DMATX(9,3) = XMURO2
      DMATX(9,4) = XMTRO4
      DMATX(9,5) = BXMRO2
      DMATX(9,9) = XMURO2
      DMATX(10,3) = XMURO2
      DMATX(10,4) = -XMTRO4
      DMATX(10,5) = BXMRO2
      DMATX(10,10) = XMURO2
      GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
      DMATX(1,11) = SUMM*(VR-WQ-GSTH)+GAM1*(Q2+R2)-GAM2*PR-GAM6*Q
     1           +SFXU+SFXS
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)-GAM1*PQ-GAM2*QR+GAM6*P
     1           +SFYU+SFYS
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = -GAM2*(WP-UR+GCTSP)+(XIXZ+XIXZP)*PQ-XIYZP*(P2+R2)
     1           +(XIY-XIZ+XIXP)*QR-GAM7*P+SNPU+SNPS
      DMATX(5,11) = GAM2*(VR-WQ-GSTH)-(XIX-XIZ+XIYP)*PR-GAM7*Q
     1           +XIXZP*(Q2+R2)-XIYZP*PQ+XIXZ*(R2-P2)+SNTU+SNTS
      DMATX(6,11) = GAM1*(WP-UR+GCTSP)+(XIX-XIY-GAM5)*PQ-(XIXZ+XIXZP)
     1           *QR+GAM8*Q+XIYZP*PR+GAM9*P+SNPSU+SNPSS
      DMATX(7,11) = XMUFO2*(UQ-VP+GCTCP-A*PR+ZFD1*(P2+R2)
     1           -TFO2*QR)+FZU(1)+SI(1)
      DMATX(8,11) = XMUFO2*(UQ-VP+GCTCP-A*PR+ZFD2*(P2+R2)
     1           +TFO2*QR)+FZU(2)+SI(2)
      DMATX(9,11) = XMURO2*(UQ-VP+GCTCP+B*PR+ZRD3*(P2+R2)
     1           -TRO2*QR)+FZU(3)+SI(3)
      DMATX(10,11) = XMURO2*(UQ-VP+GCTCP+B*PR+ZRD4*(P2+R2)
     1           +TRO2*QR)+FZU(4)+SI(4)
      RETURN
      END
```

```
      SUBROUTINE MTRXSF
C           HVOSM-VD2 VERSION
C           REVISED OCTOBER 1975   CALSPAN CORPORATION
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),X1R,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1            (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1            (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
```

```
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   MTXS0500
     1            (PSII(1),PSI1)                                         MTXS0510
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,MTXS0520
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          MTXS0530
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, MTXS0540
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   MTXS0550
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   MTXS0560
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPMTXS0570
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    MTXS0580
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   MTXS0590
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, MTXS0600
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  MTXS0610
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  MTXS0620
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, MTXS0630
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,MTXS0640
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2MTXS0650
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,MTXS0660
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           MTXS0670
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  MTXS0680
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       MTXS0690
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                MTXS0700
     1              XIYP,SPH1C,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       MTXS0710
     2              SLOPE1,SLOPE2,XTRA(300)                              MTXS0720
      DIMENSION UI(4),VI(4),WI(4)                                        MTXS0730
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       MTXS0740
      DIMENSION APITCH(4)                                                MTXS0750
      EQUIVALENCE (APITCH(1),APTCH1)                                     MTXS0760
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        MTXS0770
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),MTXS0780
     2               NCAMF,NCAMR,NDTHF,NDTHR                             MTXS0790
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       MTXS0800
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,      MTXS0810
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,     MTXS0820
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,     MTXS0830
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,         MTXS0840
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                      MTXS0850
C                                                                        MTXS0860
      CALL CLEAR(DMATX(1,1),DMATX(10,11))                                MTXS0870
      DMATX(1,1) = SUMM                                                  MTXS0880
      DMATX(1,5) = GAM2                                                  MTXS0890
      DMATX(1,6) = RHOMUR*PHIR+RHFMUF*PHIF                               MTXS0900
      DMATX(2,2) = SUMM                                                  MTXS0910
      DMATX(2,4) = -GAM2                                                 MTXS0920
      DMATX(2,6) = GAM1                                                  MTXS0930
      DMATX(2,8) = -RHFMUF                                               MTXS0940
      DMATX(2,10)= -RHOMUR                                               MTXS0950
      DMATX(3,3) = XMS                                                   MTXS0960
      DMATX(4,2) = -GAM3                                                 MTXS0970
      DMATX(4,4) = XIX+XIXP                                              MTXS0980
      DMATX(4,6) = -XIXZ-XIXZP                                           MTXS0990
      DMATX(4,8) = RHFMUF*ZFD1                                           MTXS1000
```

```
      DMATX(4,10)= RHOMUR*ZRD3
      DMATX(5,1) = GAM2
      DMATX(5,5) = XIY+XIYP
      DMATX(5,6) = -XIYZP
      DMATX(6,1) = RHFMUF*PHIF+RHOMUR*PHIR
      DMATX(6,2) = GAM1
      DMATX(6,4) = -XIXZ-XIXZP-RHFMUF*A+RHOMUR*B
      DMATX(6,5) = -XIYZP
      DMATX(6,6) = XIZ+XIZP+XIR+XIF
      DMATX(6,8) = -RHFMUF*A
      DMATX(6,10)= BROMUR
      DMATX(7,3) = XMUF
      DMATX(7,4) = -RHFMUF*PHIF
      DMATX(7,5) = -AMUF
      DMATX(7,7) = XMUF
      DMATX(7,8) = -RHFMUF*PHIF
      DMATX(8,2) = -RHFMUF
      DMATX(8,3) = -RHFMUF*PHIF
      DMATX(8,4) = XIF+RHFMUF*ZFD1RF
      DMATX(8,5) = AMUF*RFPF
      DMATX(8,6) = -RHFMUF*A
      DMATX(8,7) = -RHFMUF*PHIF
      DMATX(8,8) = RF2MFI
      DMATX(9,3) = XMUR
      DMATX(9,4) = -RHOMUR*PHIR
      DMATX(9,5) = BMUR
      DMATX(9,9) = XMUR
      DMATX(9,10)= -RHOMUR*PHIR
      DMATX(10,2) = -RHOMUR
      DMATX(10,3) = -RHOMUR*PHIR
      DMATX(10,4) = XIR+RHOMUR*ZRD3R
      DMATX(10,5) = -BMUR*RPR
      DMATX(10,6) = BROMUR
      DMATX(10,9) = -RHOMUR*PHIR
      DMATX(10,10)= RHMR2I
      GCTSP = G*AMTX(3,2)
      GCTCP = G*AMTX(3,3)
      DMATX(1,11) = SUMM*(VR-WQ-GSTH)-GAM2*PR+(RHOMUR*PHIR+RHFMUF*PHIF)
     1            *PQ+GAM1*(Q2+R2)-GAM6*Q-2.0*(RHOMUR*RPHRD+RHFMUF*
     2             RPHFD)+SFXU+SFXS
      DMATX(2,11) = SUMM*(WP-UR+GCTSP)+GAM6*P-GAM1*PQ-GAM2*QR
     1            -RHOMUR*PHIR*(P2+R2+PHIRD2)-RHFMUF*PHIF*(P2+R2+
     2             PHIFD2)+SFYU+SFYS
      DMATX(3,11) = XMS*(UQ-VP+GCTCP)-SFZ1+SFZS
      DMATX(4,11) = GAM3*(UR-WP-GCTSP)+(XIXZ+XIXZP)*PQ-GAM7*P
     1            +(XIY-XIZ+XIXP)*QR-GAM4*(P2+R2)+RHOMUR*PHIR*ZRD3*
     2             PHIRD2+RHFMUF*PHIF*ZFD1*PHIFD2+SNPS+SNPU
      DMATX(5,11) = GAM2*(VR-WQ-GSTH)+XIXZ*(R2-P2)+(XIZ-XIX-XIYP)*PR
     1            -GAM7*Q-2.0*Q*(RHOF*WFMF+RHO*TG61)+(XIXZP-BROMUR
     2            +RHOF*AMUF)*(Q2+R2)-XIYZP*PQ-2.0*RHOMUR*ZRD3R*RPHRD
     3            -2.0*RHFMUF*ZFD1RF*RPHFD+SNTU+SNTS
```

```
 DMATX(6,11) = GAM1*(WP-UR+GCTSP)+(XIX-XIY-GAM5)*PQ                    MTXS1520
1              -(XIXZ+XIXZP-BROMUR+AMUF*RHOF)*QR+GAM8*Q+XIYZP*PR       MTXS1530
2              +GAM9*P+XMUR*RPR*(VR-WQ-2.0*RHO*RPHRD-B*(Q2-P2          MTXS1540
3              -PHIRD2)-GSTH)+XMUF*RFPF*(VR-WQ-2.0*RHOF*RPHFD          MTXS1550
4              +A*(Q2-P2-PHIFD2)-GSTH)+SNPSS+SNPSU                     MTXS1560
 DMATX(7,11) = XMUF*(UQ-VP+RHOF*PHIFD2+2.0*P*RHOF*PHIFD-A*PR           MTXS1570
1              +RFPF*QR+ZFD1RF*(P2+Q2)+GCTCP)                          MTXS1580
2              +FZU(1)+FZU(2)+SI(1)+SI(2)                              MTXS1590
 DMATX(8,11) = RHFMUF*(UR-WP-2.0*P*DEL1D+2.0*P*RFPF*PHIFD+A*PQ         MTXS1600
1              +RFPF*(P2+R2)+ZFD1RF*QR-GCTH*SIN(PHIT+PHIF))            MTXS1610
2              +RHFMUF*PHIF*(VP-UQ-2.0*P*RHOF*PHIFD+A*PR               MTXS1620
3              -ZFD1RF*(P2+Q2))-XIF*PHIF*(R2-Q2)-XIF*QR+SNPF           MTXS1630
 DMATX(9,11) = XMUR*(UQ-VP+RHO*PHIRD2+2.0*P*RHO*PHIRD+B*PR             MTXS1640
1              +RPR*QR+ZRD3R*(P2+Q2)+GCTCP)                            MTXS1650
2              +FZU(3)+FZU(4)+SI(3)+SI(4)                              MTXS1660
 DMATX(10,11)= RHOMUR*(UR-WP-2.0*P*DEL3D+2.0*P*RPR*PHIRD-B*PQ          MTXS1670
1              +RPR*(P2+R2)+ZRD3R*QR-GCTH*SIN(PHIT+PHIR))              MTXS1680
2              +RHOMUR*PHIR*(VP-UQ-2.0*P*RHO*PHIRD-B*PR                MTXS1690
3              -ZRD3R*(P2+Q2))-XIR*PHIR*(R2-Q2)-XIR*QR+SNPR            MTXS1700
 RETURN                                                                MTXS1710
 END                                                                   MTXS1720
```

```
      SUBROUTINE OUTPUT(IND)                                            OUTP0010
C              HVOSM-VD2 VERSION                                        OUTP0020
C              REVISED OCTOBER 1975  CALSPAN CORPORATION                OUTP0030
      COMMON/HEAD/ VHED(20),CHED(20),GHED(20),SHED(20),THED(20),        OUTP0040
     1              NPAGE(20)                                           OUTP0050
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    OUTP0060
     1             A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,    OUTP0070
     2             PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                    OUTP0080
     3             XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,OUTP0090
     4             RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, OUTP0100
     5             T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,   OUTP0110
     6             HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB, OUTP0120
     7             DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR, OUTP0130
     8             NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),  OUTP0140
     9             NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                OUTP0150
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),OUTP0160
     1           XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN OUTP0170
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             OUTP0180
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))OUTP0190
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),OUTP0200
     2             (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),      OUTP0210
     3             (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),     OUTP0220
     4             (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),       OUTP0230
     5             (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),         OUTP0240
     6             (PSIFID,VAR(22))                                     OUTP0250
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      OUTP0260
     1             (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))OUTP0270
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       OUTP0280
     3             (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),   OUTP0290
     4             (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),  OUTP0300
     5             (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),        OUTP0310
     6             (DPSIFI,DER(21)),(DDPSFI,DER(22))                    OUTP0320
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         OUTP0330
     1             (DER(10),DPHIFD)                                     OUTP0340
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       OUTP0350
     1             (DER(14),DDEL4D)                                     OUTP0360
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,OUTP0370
     1              PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4), OUTP0380
     2              CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),OUTP0390
     3              STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),   OUTP0400
     4              XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), OUTP0410
     5              YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), OUTP0420
     6              CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),   OUTP0430
     7              CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),    OUTP0440
     8              SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),OUTP0450
     9              FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)OUTP0460
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      OUTP0470
     1              BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),   OUTP0480
     2              FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2), OUTP0490
```

```
3              F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                       OUTP0500
 DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                              OUTP0510
 EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),          OUTP0520
1              (PSII(1),PSI1).                                            OUTP0530
 COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,OUTP0540
1             GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,              OUTP0550
2              TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,       OUTP0560
3              BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,         OUTP0570
4              XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,         OUTP0580
5              ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPOUTP0590
6             ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,          OUTP0600
7              SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,         OUTP0610
8              SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,       OUTP0620
9              ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ        OUTP0630
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,         OUTP0640
1              ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,       OUTP0650
2              TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,OUTP0660
3              HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2OUTP0670
4             ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,OUTP0680
5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL                OUTP0690
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                        OUTP0700
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)              OUTP0710
 COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                            OUTP0720
1                DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,         OUTP0730
2                PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),         OUTP0740
3                SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)             OUTP0750
 LOGICAL LCB1,LCB2                                                        OUTP0760
 COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                            OUTP0770
 DIMENSION RPSI(4),DRPSI(4)                                               OUTP0780
 EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                          OUTP0790
 COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),          OUTP0800
A                XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),                   OUTP0810
B                XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),               OUTP0820
C                XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4                  OUTP0830
 EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)              OUTP0840
 EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)              OUTP0850
 COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),           OUTP0860
1               TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,OUTP0870
2        DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),         OUTP0880
3               SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),       OUTP0890
4               EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),           OUTP0900
5               FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),        OUTP0910
6               SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),           OUTP0920
7               FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),           OUTP0930
8           ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),OUTP0940
9           RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD         OUTP0950
 COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP OUTP0960
 LOGICAL  IUVS,IUVB,IRPS                                                  OUTP0970
 COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,OUTP0980
1               NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,       OUTP0990
2               RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,          OUTP1000
```

```
     3              DELTAE,PIO15R,COMEN5,TQE,RPME                         OUTP1010
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                  OUTP1020
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,         OUTP1030
     2              SLOPE1,SLOPE2,XTRA(300)                               OUTP1040
      DIMENSION UI(4),VI(4),WI(4)                                         OUTP1050
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                        OUTP1060
      DIMENSION APITCH(4)                                                 OUTP1070
      EQUIVALENCE (APITCH(1),APTCH1)                                      OUTP1080
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,         OUTP1090
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),OUTP1100
     2               NCAMF,NCAMR,NDTHF,NDTHR                              OUTP1110
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,        OUTP1120
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,       OUTP1130
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,      OUTP1140
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,      OUTP1150
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,          OUTP1160
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                       OUTP1170
      COMMON/DRIVE/EN,FKD,FKP,FKS1,FKS2,FKSKID,TESTE,TESTS1,TESTS2,       OUTP1180
     1             TESTR1,TESTR2,THESKD,FBRK,APB,DSDES,                   OUTP1190
     2             TRKIN,TMT,DESS,DIST,DISTC,CONMPH,UT,UTMPH,             OUTP1200
     3             APD,DELTAX,DELTV,TJ,TTEM,TTPSIT,PSISKD,ST,STSO2,QAY,   OUTP1210
     4             AXP,AYP,DI,UP,XVP,YVP,SLOPE,SLOPER,PSIJ,XINT,X,Y,      OUTP1220
     5             TERMX,TERMY,TEMPOR,AE,EI,EWT,AREI(7),ARCAPE(7),ET,     OUTP1230
     6             PSIFFH,TITE,DPSISF,DPSILF,PSIM,APSI,APSIM,TPD(10),     OUTP1240
     7             PPD(10),NPD,KCOUNT,ISKIDP,ISMAIN,IGEAR,WEIGHT(10),     OUTP1250
     8             DEND                                                   OUTP1260
      DIMENSION ASTR(4),SLPANG(4)                                         OUTP1270
      DIMENSION OTQD(4),ORPS(4),OSLIP(4)                                  OUTP1280
      DATA STAR,BLNK/1H*,1H /                                             OUTP1290
      DATA LPP/50/                                                        OUTP1300
      DATA TTTTTT/-9999.0/                                                OUTP1310
      IF(IND.NE.0) GO TO 400                                              OUTP1320
      LINES = 0                                                           OUTP1330
      RETURN                                                              OUTP1340
  400 LINES = LINES+1                                                     OUTP1350
      IF(MOD(LINES,LPP).NE.1) GO TO 500                                   OUTP1360
      XPAGE = 0.01*(LINES+LPP-1)/LPP                                      OUTP1370
      NT = 10                                                             OUTP1380
      DO 410 J=1,20                                                       OUTP1390
      IF(NPAGE(J).EQ.0) GO TO 410                                         OUTP1400
      NT = NT+1                                                           OUTP1410
      PAGE = NT+XPAGE                                                     OUTP1420
      WRITE(NT,1000) (HED(I),I=1,18),DADE(1),DADE(2),                     OUTP1430
     1     (VHED(I),I=1,10),(THED(I),I=1,10),(CHED(I),I=1,10),            OUTP1440
     2      (GHED(I),I=1,10),(SHED(I),I=1,10),PAGE                        OUTP1450
 1000 FORMAT(1H1,19X,18A4,20X,2A4 / 5X,3(10A4) / 5X,2(10A4),              OUTP1460
     1       22X,4HPAGE,1X,F8.2  /  )                                     OUTP1470
      GO TO(111,112,113,114,115,116,117,118,119,120,121,122,123,124,      OUTP1480
     *      125,126,127,128,129,130),J                                    OUTP1490
C  POSSIBLE ERROR MESSAGE                                                 OUTP1500
      GO TO 410                                                           OUTP1510
```

```
 111 WRITE(NT,1100)                                                       OUTP1520
1100 FORMAT(1H0,48X,23HS P R U N G    M A S S  /                          OUTP1530
    A62H  TIME   |          POSITION (FEET)           |     VELOCITY ( OUTP1540
    B62HFT/SEC)            |            ACCELERATION (G-UNITS)          OUTP1550
    C 6H     |   /                                                        OUTP1560
    D62H   SEC   |     XC'    |     YC'    |     ZC'    | FORWARD  |  LA OUTP1570
    E62HTERAL  | VERTICAL |    LONG.  |     LAT.  |     VERT.  |  RESU OUTP1580
    F 6HLT.  |   /)                                                       OUTP1590
     GO TO 410                                                            OUTP1600
 112 IF(ISUS.EQ.1) GO TO 1121                                             OUTP1610
     WRITE(NT,1200)                                                       OUTP1620
1200 FORMAT(                                                              OUTP1630
    A62H0        |                           S P R U N G   M A S S   OUTP1640
    B62H                   |  SIDESLIP |    COURSE |FRONT STEER| REAR  OUTP1650
    C 6HSTEER|  /                                                         OUTP1660
    D62H  TIME   |   ANGULAR VELOCITIES (DEG/SEC)     |      ORIENTATIO OUTP1670
    E62HN (DEGREES)        |    ANGLE  |    ANGLE  |    ANGLE  |    AN OUTP1680
    F 6HGLE  |  /                                                         OUTP1690
    G62H   SEC   |      P     |      Q     |      R     |     ROLL   |   P OUTP1700
    H62HITCH   |     YAW   |      DEG   |     DEG   |     DEG   |     D OUTP1710
    I 6HEG    |  /)                                                       OUTP1720
     GO TO 410                                                            OUTP1730
1121 WRITE(NT,1201)                                                       OUTP1740
1201 FORMAT(                                                              OUTP1750
    A62H0        |                           S P R U N G   M A S S   OUTP1760
    B62H                   |  SIDESLIP |    COURSE |FR.STEER|RR STEER| OUTP1770
    C 9HLR STEER|  /                                                      OUTP1780
    D62H  TIME   |   ANGULAR VELOCITIES (DEG/SEC)     |      ORIENTATIO OUTP1790
    E62HN (DEGREES)        |    ANGLE  |    ANGLE  | ANGLE |  ANGLE | OUTP1800
    F 9H  ANGLE |  /                                                      OUTP1810
    G62H   SEC   |      P     |      Q     |      R     |     ROLL   |   P OUTP1820
    H62HITCH   |     YAW   |      DEG   |     DEG   |   DEG  |   DEG  | OUTP1830
    I 9H   DEG  |  /   )                                                  OUTP1840
     GO TO 410                                                            OUTP1850
 113 WRITE(NT,1300)                                                       OUTP1860
1300 FORMAT(                                                              OUTP1870
    A62H0 TIME   |       WHEEL RIDE DISPLACEMENTS (INCHES)         |      OUTP1880
    B44H      WHEEL RIDE VELOCITIES (IN/SEC)          |   /               OUTP1890
    C62H   SEC   |       RF    |      LF    |       RR    |       LR    |  OUTP1900
    D44H  RF   |       LF   |       RR   |       LR   |   /)              OUTP1910
     GO TO 410                                                            OUTP1920
 114 GO TO(1140,1141,1142),IS1                                            OUTP1930
1140 WRITE(NT,1400)                                                       OUTP1940
1400 FORMAT(55H0          |                SPRUNG MASS             |  WHEEL OUTP1950
    A62HRIDE ACCEL  |     REAR ROLL CENTER RIDE         |      REAR AXLE A OUTP1960
    B15HNGULAR          |  /                                              OUTP1970
    C62H  TIME    | ANGULAR ACCELERATIONS (DEG/SEC**2)|    (IN/SEC**2)  OUTP1980
    D62H        | DEFL    | VELOCITY |ACCELERATION|  DEFL   | VELOCITY |  A OUTP1990
    E 9HCCEL    |  /                                                      OUTP2000
    F62H   SEC   |      DP/DT |      DQ/DT |      DR/DT |       RF   |      OUTP2010
    G62H LF   | INCHES |    IN/SEC | IN/SEC**2  |    DEG    |  DEG/SEC |DEG OUTP2020
```

```
   H 8H/SEC**2|   /)
    GO TO 410
1141 WRITE(NT,1401)
1401 FORMAT(1HO,
   A62H          |          SPRUNG MASS             |       WHEEL RIDE ACCE
   B14HL              | /
   C62H   TIME    | ANGULAR ACCEL. (DEG/SEC**2) |            (IN/SEC**2)
   D15H                | /
   E62H     SEC   |    DP/DT |    DQ/DT |    DR/DT |       RF |      LF |
   F15H  RR |      LR | / )
    GO TO 410
1142 WRITE(NT,1402)
1402 FORMAT(
   A62HO          |          SPRUNG MASS            |     FRONT ROLL CENTER
   B62H      |      REAR ROLL CENTER      | FR. AXLE ANGULAR|REAR AXLE A
   C 7HNGULAR|   /
   D62H   TIME    | ANGULAR ACCEL. (DEG/SEC**2) |  DEFL   |VELOCITY|  A
   E62HCCEL |  DEFL  |VELOCITY|  ACCEL |VELOCITY|  ACCEL |VELOCITY|
   F 7HACCEL |   /
   G62H     SEC   |    DP/DT |    DQ/DT |    DR/DT | INCHES | IN/SEC | IN
   H62H/SEC2| INCHES | IN/SEC | IN/SEC2| DEG/SEC |DEG/SEC2| DEG/SEC|DE
   I 7HG/SEC2|   /    )
    GO TO 410
 115 WRITE(NT,1500)
1500 FORMAT(1HO,8X,48H| STEER FRIC| STEER STOP|    STEER   |     STEER
   A 1H|  /
   B59H  TIME    |   TORQUE   |   TORQUE   |      VEL     |    ACCEL   |  /
   C59H   SEC    |    LB-IN   |    LB-IN   |   DEG/SEC   | DEG/SEC**2|  /)
    GO TO 410
 116 GO TO(1160,1161,1162),IS1
1160 WRITE(NT,1600)
1600 FORMAT(
   A62HO TIME    |    STEER ANGLE IN GROUND PLANE (DEG)            |   C
   B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) |  CAMBER ANGLE (DE
   C 6HG)    |   /
   D62H    SEC   |       RF    |       LF    |       RR    |       LR    |
   E62H  RF    |       LF    |       RR    |       LR    |       RF    |
   F 6HLF    |   /)
    GO TO 410
1161 WRITE(NT,1601)
1601 FORMAT(
   A62HO TIME    |    STEER ANGLE IN GROUND PLANE (DEG)            |   C
   B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) |    CAMBER ANGLE (
   C 9HDEG)    |   /
   D62H    SEC   |       RF    |       LF    |       RR    |       LR    |
   E62H  RF    |       LF    |       RR    |       LR    |   RF |   LF |   R
   F 9HR |  LR |   /  )
    GO TO 410
1162 WRITE(NT,1602)
1602 FORMAT(
   A62HO TIME    |    STEER ANGLE IN GROUND PLANE (DEG)            |   C
```

```
    B62HAMBER ANGLE RELATIVE TO GROUND PLANE (DEG) | AXLE ROLL ANGLE (  OUTP2540
    C 6HDEG) |    /                                                     OUTP2550
    D62H   SEC    |       RF    |.      LF    |       RR    |      LR    |   OUTP2560
    E62H  RF    |      LF    |      RR    |      LR    | FRONT-PHIF| REAR- OUTP2570
    F 6HPHIR |    / )                                                   OUTP2580
     GO TO 410                                                          OUTP2590
 117 WRITE(NT,1700)                                                     OUTP2600
1700 FORMAT(                                                            OUTP2610
    A62H0         |     LONGITUDINAL WHEEL CENTER VELOCITY         |    OUTP2620
    B44H    LATERAL CONTACT POINT VELOCITY          |   /               OUTP2630
    C62H  TIME   |       PARALLEL TO GROUND PLANE (FT/SEC)         |    OUTP2640
    D44H   PARALLEL TO GROUND PLANE (FT/SEC)        |   /               OUTP2650
    E62H   SEC    |       RF    |       LF    |       RR    |      LR    |  OUTP2660
    F44H  RF    |      LF    |      RR    |      LR    |   /)           OUTP2670
     GO TO 410                                                          OUTP2680
 118 WRITE(NT,1800)                                                     OUTP2690
1800 FORMAT(                                                            OUTP2700
    A58H0 TIME    |   TIRE CONTACT POINT ELEVATION (INCHES)        | /  OUTP2710
    B58H   SEC    |       RF    |       LF    |       RR    |      LR    | /) OUTP2720
     GO TO 410                                                          OUTP2730
 119 WRITE(NT,1900)                                                     OUTP2740
1900 FORMAT(                                                            OUTP2750
    A62H0 TIME   |          TOTAL SUSPENSION FORCE (LBS)             |  OUTP2760
    B44H     SUSPENSION ANTI-PITCH FORCE (LBS)      |    /              OUTP2770
    C62H   SEC    |       RF    |       LF    |       RR    |      LR    |  OUTP2780
    D44H  RF    |      LF    |      RR    |      LR    |    /)          OUTP2790
     GO TO 410                                                          OUTP2800
 120 WRITE(NT,2000)                                                     OUTP2810
2000 FORMAT(                                                            OUTP2820
    A62H0 TIME   |        SUSPENSION DAMPING FORCE (LBS)              | OUTP2830
    B44H       SUSPENSION SPRING FORCE (LBS)        |      /            OUTP2840
    C62H   SEC    |       RF    |       LF    |       RR    |      LR    |  OUTP2850
    D44H  RF    |      LF    |      RR    |      LR    |    /)          OUTP2860
     GO TO 410                                                          OUTP2870
 121 WRITE(NT,2100)                                                     OUTP2880
2100 FORMAT(                                                            OUTP2890
    A62H0 TIME   |          RADIAL TIRE FORCES (LBS)                  | OUTP2900
    B44H       ROLLING RADIUS (INCHES)              |   /               OUTP2910
    C62H   SEC    |       RF    |       LF    |       RR    |      LR    |  OUTP2920
    D44H  RF    |      LF    |      RR    |      LR    |   /)           OUTP2930
     GO TO 410                                                          OUTP2940
 122 WRITE(NT,2200)                                                     OUTP2950
2200 FORMAT(                                                            OUTP2960
    A62H0 TIME   |          TIRE NORMAL FORCE (LBS)             |       OUTP2970
    B62H TIRE SIDE FORCE (LBS)              |         SLIP ANGLE (DEG)  OUTP2980
    C 6H     |    /                                                     OUTP2990
    D62H  SEC    |      RF  |      LF   |      RR  |      LR  |      RF OUTP3000
    E62H  |      LF  |      RR  |      LR  |   RF   |  ·LF  |   RR  |   OUTP3010
    F 6H LR  |    /)                                                    OUTP3020
     GO TO 410                                                          OUTP3030
 123 WRITE(NT,2300)                                                     OUTP3040
```

```
2300 FORMAT(1H0,105X,25H|  ENGINE    |  ENGINE    |    /              OUTP3050
    A62H  TIME    |          TIRE TRACTIVE FORCE (LBS)             |   OUTP3060
    B62H            DRIVING TORQUE (LB-FT)            |   SPEED  |  TO OUTP3070
    C 6HRQUE |    /                                                    OUTP3080
    D62H   SEC    |      RF    |      LF    |      RR    |     LR    |  OUTP3090
    E62H  RF    |      LF    |      RR    |      LR    |    RPM    |  F OUTP3100
    F 6H1-LB |    /     )                                              OUTP3110
     GO TO 410                                                         OUTP3120
 124 WRITE(NT,2400)                                                    OUTP3130
2400 FORMAT(                                                           OUTP3140
    A62HO TIME    |   Z'-VERTICAL TIRE FORCE (LBS)          |  X'-HORIZO OUTP3150
    B62HNTAL TIRE FORCE (LBS)        |   Y'-HORIZONTAL TIRE FORCE (LBS)  OUTP3160
    C 6H      |    /                                                   OUTP3170
    D62H   SEC    |     RF    |     LF    |     RR    |     LR    |     RF    |  OUTP3180
    E62H  LF    |     RR    |     LR    |     RF    |     LF    |     RR    |  OUTP3190
    F 6HLR     |    /)                                                 OUTP3200
     GO TO 410                                                         OUTP3210
 125 WRITE(NT,2500)                                                    OUTP3220
2500 FORMAT(                                                           OUTP3230
    A62HO TIME    |        TERRAIN ELEVATION (IN)           |   TERRAIN S OUTP3240
    B62HLOPE-CAMBER (PHIG) (DEG)    |   TERRAIN SLOPE-PITCH (THETAG) (D  OUTP3250
    C 6HEG) |    /                                                     OUTP3260
    D62H   SEC    |     RF    |     LF    |     RR    |     LR    |     RF    |  OUTP3270
    E62H  LF    |     RR    |    LR     |     RF    |     LF    |     RR    |  OUTP3280
    F 6HLR     |    /)                                                 OUTP3290
     GO TO 410                                                         OUTP3300
 126 WRITE(NT,2600)                                                    OUTP3310
2600 FORMAT(                                                           OUTP3320
    A62HO TIME    | SPRUNG MASS ACCELERATION LOCATION 1(G-UNITS)   | SPR OUTP3330
    B44HUNG MASS ACCELERATION LOCATION 2 (G-UNITS) |      /             OUTP3340
    C62H   SEC    |   LONG.    |    LAT.    |   VERT.    |  RESULT. |   L OUTP3350
    D44HONG.   |    LAT.    |   VERT.    |  RESULT. |      /)           OUTP3360
     GO TO 410                                                         OUTP3370
 127 WRITE(NT,2700)                                                    OUTP3380
2700 FORMAT(                                                           OUTP3390
    A62HO TIME     |          WHEEL SLIP (PERCENT)            |        OUTP3400
    B62H FRICTION RATIO                |         WHEEL ROTATION (REV/SEC)   OUTP3410
    C 6H      |    /                                                   OUTP3420
    D62H   SEC    |     RF    |     LF    |     RR    |     LR    |     RF    |  OUTP3430
    E62H  LF    |     RR    |     LR    |     RF    |     LF    |     RR    |  OUTP3440
    F 6HLR     |    /     )                                            OUTP3450
     GO TO 410                                                         OUTP3460
 128 WRITE(NT,2800)                                                    OUTP3470
2800 FORMAT(                                                           OUTP3480
    A62HO         |   HYDRAULIC PRES.     |                   BRAKE TORQU OUTP3490
    B62HES                   |              BRAKE TEMPERATURE          OUTP3500
    C 6H      |    /                                                   OUTP3510
    D62H  TIME    |        (PSIG)        |                     (LB-FT)  OUTP3520
    E62H                     |               (DEG-F)                   OUTP3530
    F 6H      |    /                                                   OUTP3540
    G62H   SEC    |   FRONT    |    REAR    |      RF    |      LF    |  OUTP3550
```

```
    H62H  RR   |       LR    |       RF    |       LF    |       RR    |
    I 6HLR    |    /     )
     GO TO 410
 129 WRITE(NT,2900)
2900 FORMAT(
    A62HO         |                                       DISSIPATED ENERGY
    B44H(FT-LB)                                          |    /
    C62H  TIME    |                    BRAKES                         |
    D44H                     TIRES                       |    /
    E62H   SEC    |      RF    |       LR    |       RR    |       LR    |
    F44H  RF   |       LF    |       RR    |       LR    |    /     )
     GO TO 410
 130 WRITE(NT,3000)
3000 FORMAT(
    A62HO TIME    | DELTA PSIF| ERROR ETJ |   DESIRED |   ACC PED | BRA
    B15HKE PED |       |   /
    C62H   SEC    |      DEG   |      IN.  |   ACCEL   |     DEFL   |
    D15HFORCE  | GEAR |   /
    E62H          |            |           | IN/SEC**2|      IN.   |
    F15H  LBS  |       |   /   )
 410 CONTINUE
 500 NT = 10
     DO 600 J=1,20
     IF(NPAGE(J).EQ.0) GO TO 600
     GO TO (11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,
    *     28,29,30),J
  11 NT = NT+1
     ACLON = (DU-VR+WQ)/G
     ACLAT = (DV+UR-WP)/G
     ACVER = (DW+VP-UQ)/G
     ULON = U/12.
     VLAT = V/12.
     WVER = W/12.
     ACRES = SQRT(ACLON**2+ACLAT**2+ACVER**2)
     OXCP = XCP/12.
     OYCP = YCP/12.
     OZCP = ZCP/12.
     WRITE(NT,5000) T,OXCP,OYCP,OZCP,ULON,VLAT,WVER,ACLON,ACLAT,ACVER,
    *               ACRES
5000 FORMAT(' ',F7.4,10(2X,F10.2))
     GO TO 600
  12 NT = NT+1
     ONU = 0.0
     IF(DYCP.EQ.0.0.AND.DXCP.EQ.0.0) GO TO 212
     ONU = ATAN2(DYCP,DXCP)/RAD
 212 ROLL = P/RAD
     PITCH = Q/RAD
     YAW = R/RAD
     PHIO = PHIT/RAD
     THTAO = THETT/RAD
     PSIO = PSIT/RAD
```

```
      OBETA = ONU-PSIO
      PSIFO = PSI1/RAD
      IF(ISUS.EQ.1) GO TO 213
      OPSIR = PSI3/RAD
      WRITE(NT,5000) T,ROLL,PITCH,YAW,PHIO,THTAO,PSIO,OBETA,ONU,PSIFO,
     *               OPSIR
      GO TO 600
  213 OPSI3 = PSI3/RAD
      OPSI4 = PSI4/RAD
      WRITE(NT,5004) T,ROLL,PITCH,YAW,PHIO,THTAO,PSIO,OBETA,ONU,PSIFO,
     *               OPSI3,OPSI4
      GO TO 600
   13 NT = NT+1
      GO TO(131,132,133),IS1
  131 OETA3 = DEL3+TRO2*PHIR
      OETA3 = DEL3+TRO2*PHIR
      OETA4 = DEL3-TRO2*PHIR
      OETA3D = DEL3D+TRO2*PHIRD
      OETA4D = DEL3D-TRO2*PHIRD
      WRITE(NT,5000) T,DEL1,DEL2,OETA3,OETA4,DEL1D,DEL2D,OETA3D,OETA4D
      GO TO 600
  132 WRITE(NT,5000) T,DEL1,DEL2,DEL3,DEL4,DEL1D,DEL2D,DEL3D,DEL4D
      GO TO 600
 5004 FORMAT(1H ,F7.4,8(2X,F10.2),3(2X,F7.2) )
  133 OETA1 = DEL1+TFO2*PHIF
      OETA2 = DEL1-TFO2*PHIF
      OETA3 = DEL3+TRO2*PHIR
      OETA4 = DEL3-TRO2*PHIR
      OETA1D = DEL1D+TFO2*PHIFD
      OETA2D = DEL1D-TFO2*PHIFD
      OETA3D = DEL3D+TRO2*PHIRD
      OETA4D = DEL3D-TRO2*PHIRD
      WRITE(NT,5000) T,OETA1,OETA2,OETA3,OETA4,OETA1D,OETA2D,OETA3D,
     *               OETA4D
      GO TO 600
   14 NT = NT+1
      ODP = DP/RAD
      ODQ = DQ/RAD
      ODR = DR/RAD
      IF(ISUS.EQ.1) GO TO 141
      DPHDTO = PHIRD/RAD
      OPHDD = DPHIRD/RAD
      IF(ISUS.EQ.2) GO TO 142
      PHIRO = PHIR/RAD
      WRITE(NT,5001) T,ODP,ODQ,ODR,DDEL1D,DDEL2D,DEL3,DEL3D,DDEL3D,
     *               PHIRO,DPHDTO,OPHDD
 5001 FORMAT(' ',F7.4,3(2X,F10.2),2(2X,F9.1),2X,F7.2,2X,F9.1,2X,
     *       F9.1,2X,F7.2,2X,F9.1,2X,F9.1 )
      GO TO 600
  141 WRITE(NT,5005) T,ODP,ODQ,ODR,DDEL1D,DDEL2D,DDEL3D,DDEL4D
 5005 FORMAT(1H ,F7.3,3(2X,F8.2),10(2X,F7.1) )
```

```
      GO TO 600                                                          OUTP4580
  142 DPFDTO = PHIFD/RAD                                                 OUTP4590
      OPFDD = DPHIFD/RAD                                                 OUTP4600
      WRITE(NT,5005) T,ODP,ODQ,ODR,DEL1,DEL1D,DDEL1D,DEL3,DEL3D,DDEL3D,  OUTP4610
     *               DPFDTO,OPFDD,DPHDTO,OPHDD                           OUTP4620
      GO TO 600                                                          OUTP4630
   15 NT = NT+1                                                          OUTP4640
      ODPSF1 = DPSIFI/RAD                                                OUTP4650
      ODDPSF = DDPSFI/RAD                                                OUTP4660
      WRITE(NT,5000) T,T1PSI,T2PSI,ODPSFI,ODDPSF                         OUTP4670
      GO TO 600                                                          OUTP4680
   16 NT = NT+1                                                          OUTP4690
      PHRF = PHICI(1)/RAD                                                OUTP4700
      PHLF = PHICI(2)/RAD                                                OUTP4710
      PHRR = PHICI(3)/RAD                                                OUTP4720
      PHLR = PHICI(4)/RAD                                                OUTP4730
      PSRF = PSIIP(1)/RAD                                                OUTP4740
      PSLF = PSIIP(2)/RAD                                                OUTP4750
      PSRR = PSIIP(3)/RAD                                                OUTP4760
      PSLR = PSIIP(4)/RAD                                                OUTP4770
      IF(ISUS.EQ.2) GO TO 162                                            OUTP4780
      PHI10 = PHI1/RAD                                                   OUTP4790
      PHI20 = PHI2/RAD                                                   OUTP4800
      IF(ISUS.EQ.1) GO TO 161                                            OUTP4810
      WRITE(NT,5000) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHI10,    OUTP4820
     *               PHI20                                               OUTP4830
      GO TO 600                                                          OUTP4840
  161 PHI30 = PHI3/RAD                                                   OUTP4850
      PHI40 = PHI4/RAD                                                   OUTP4860
      WRITE(NT,5006) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHI10,    OUTP4870
     *               PHI20,PHI30,PHI40                                   OUTP4880
      GO TO 600                                                          OUTP4890
  162 PHIFO = PHIF/RAD                                                   OUTP4900
      PHIRO = PHIR/RAD                                                   OUTP4910
      WRITE(NT,5000) T,PSRF,PSLF,PSRR,PSLR,PHRF,PHLF,PHRR,PHLR,PHIFO,    OUTP4920
     *               PHIRO                                               OUTP4930
 5006 FORMAT(1H ,F7.4,8(2X,F10.2),4(1X,F6.2) )                           OUTP4940
      GO TO 600                                                          OUTP4950
   17 NT = NT+1                                                          OUTP4960
      VLNRF = UG(1)/12.                                                  OUTP4970
      VLNLF = UG(2)/12.                                                  OUTP4980
      VLNRR = UG(3)/12.                                                  OUTP4990
      VLNLR = UG(4)/12.                                                  OUTP5000
      VLTRF = VG(1)/12.                                                  OUTP5010
      VLTLF = VG(2)/12.                                                  OUTP5020
      VLTRR = VG(3)/12.                                                  OUTP5030
      VLTLR = VG(4)/12.                                                  OUTP5040
      WRITE(NT,5000) T,VLNRF,VLNLF,VLNRR,VLNLR,VLTRF,VLTLF,VLTRR,VLTLR   OUTP5050
      GO TO 600                                                          OUTP5060
   18 NT = NT+1                                                          OUTP5070
      WRITE(NT,5000) T,(ZGPP(I),I=1,4)                                   OUTP5080
```

```
      GO TO 600
   19 NT = NT+1
      WRITE(NT,5000) T,(SI(I),I=1,4),(APITCH(I),I=1,4)
      GO TO 600
   20 NT = NT+1
      IF(ISUS.EQ.2) GO TO 201
      OD1 = -CF*DEL1D
      OD2 = -CF*DEL2D
      GO TO 202
  201 OD1 = -CF*(DEL1D+TSF02*PHIFD)
      OD2 = -CF*(DEL1D-TSF02*PHIFD)
  202 IF(ISUS.EQ.1) GO TO 203
      OD3 = -CR*(DEL3D+TS02*PHIRD)
      OD4 = -CR*(DEL3D-TS02*PHIRD)
      GO TO 204
  203 OD3 = -CR*DEL3D
      OD4 = -CR*DEL4D
  204 CONTINUE
      OSP1 = -F2FI(1)
      OSP2 = -F2FI(2)
      OSP3 = -F2RI(1)
      OSP4 = -F2RI(2)
      WRITE(NT,5000) T,OD1,OD2,OD3,OD4,OSP1,OSP2,OSP3,OSP4
      GO TO 600
   21 NT = NT+1
      WRITE(NT,5000) T,(FR(I),I=1,4),(HI(I),I=1,4)
      GO TO 600
   22 NT = NT+1
      DO 220 I=1,4
      ASTR(I) = BLNK
      IF(ABS(BETBR(I)).GT.3.0) ASTR(I)=STAR
      SLPANG(I) = (TERM(I)-PSITEM(I))/RAD
  220 CONTINUE
      WRITE(NT,5003) T,(FRCP(I),I=1,4),(FS(I),ASTR(I),I=1,4),
     *                (SLPANG(I),I=1,4)
 5003 FORMAT(1H ,F7.4,1X,4(1X,F10.2),4(1X,F9.2,A1),4(1X,F7.2)    )
      GO TO 600
   23 NT = NT+1
      OTQD(1) = 0.5*TQD(1)*ARBR(1)
      OTQD(2) = OTQD(1)
      OTQD(3) = 0.5*TQD(2)*ARBR(2)
      OTQD(4) = OTQD(3)
      WRITE(NT,5000) T,(FC(I),I=1,4),(OTQD(I),I=1,4),RPME,TQE
      GO TO 600
   24 NT = NT+1
      FR10 = AMTX(3,1)*FXU(1)+AMTX(3,2)*FYU(1)+AMTX(3,3)*FZU(1)
      FR20 = AMTX(3,1)*FXU(2)+AMTX(3,2)*FYU(2)+AMTX(3,3)*FZU(2)
      FR30 = AMTX(3,1)*FXU(3)+AMTX(3,2)*FYU(3)+AMTX(3,3)*FZU(3)
      FR40 = AMTX(3,1)*FXU(4)+AMTX(3,2)*FYU(4)+AMTX(3,3)*FZU(4)
      FXPU1 = AMTX(1,1)*FXU(1)+AMTX(1,2)*FYU(1)+AMTX(1,3)*FZU(1)
      FXPU2 = AMTX(1,1)*FXU(2)+AMTX(1,2)*FYU(2)+AMTX(1,3)*FZU(2)
```

```
      FXPU3 = AMTX(I,1)*FXU(3)+AMTX(1,2)*FYU(3)+AMTX(1,3)*FZU(3)          OUTP5600
      FXPU4 = AMTX(1,1)*FXU(4)+AMTX(1,2)*FYU(4)+AMTX(1,3)*FZU(4)          OUTP5610
      FYPU1 = AMTX(2,1)*FXU(1)+AMTX(2,2)*FYU(1)+AMTX(2,3)*FZU(1)          OUTP5620
      FYPU2 = AMTX(2,1)*FXU(2)+AMTX(2,2)*FYU(2)+AMTX(2,3)*FZU(2)          OUTP5630
      FYPU3 = AMTX(2,I)*FXU(3)+AMTX(2,2)*FYU(3)+AMTX(2,3)*FZU(3)          OUTP5640
      FYPU4 = AMTX(2,1)*FXU(4)+AMTX(2,2)*FYU(4)+AMTX(2,3)*FZU(4)          OUTP5650
      WRITE(NT,5002) T,FR10,FR20,FR30,FR40,FXPU1,FXPU2,FXPU3,FXPU4,       OUTP5660
     *               FYPU1,FYPU2,FYPU3,FYPU4                              OUTP5670
 5002 FORMAT(' ',F7.4,I2(2X,F8.1) )                                       OUTP5680
      GO TO 600                                                           OUTP5690
   25 NT = NT+1                                                           OUTP5700
      PHG10 = PHGI(1)/RAD                                                 OUTP5710
      PHG20 = PHGI(2)/RAD                                                 OUTP5720
      PHG30 = PHGI(3)/RAD                                                 OUTP5730
      PHG40 = PHGI(4)/RAD                                                 OUTP5740
      THG10 = THGI(I)/RAD                                                 OUTP5750
      THG20 = THGI(2)/RAD                                                 OUTP5760
      THG30 = THGI(3)/RAD                                                 OUTP5770
      THG40 = THGI(4)/RAD                                                 OUTP5780
      WRITE(NT,5002) T,(ZPGI(I),I=1,4),PHG10,PHG20,PHG30,PHG40,THGIO,     OUTP5790
     *               THG20,THG30,THG40                                    OUTP5800
      GO TO 600                                                           OUTP5810
   26 NT = NT+1                                                           OUTP5820
      AX1 = (DU-VR+WQ-XI*(Q2+R2)+Y1*(PQ-DR)+Z1*(PR+DQ))/G                 OUTP5830
      AX2 = (DU-VR+WQ-X2*(Q2+R2)+Y2*(PQ-DR)+Z2*(PR+DQ))/G                 OUTP5840
      AY1 = (DV+UR-WP+XI*(PQ+DR)-Y1*(P2+R2)+ZI*(QR-DP))/G                 OUTP5850
      AY2 = (DV+UR-WP+X2*(PQ+DR)-Y2*(P2+R2)+Z2*(QR-DP))/G                 OUTP5860
      AZ1 = (DW+VP-UQ+X1*(PR-DQ)+Y1*(QR+DP)-Z1*(P2+Q2))/G                 OUTP5870
      AZ2 = (DW+VP-UQ+X2*(PR-DQ)+Y2*(QR+DP)-Z2*(P2+Q2))/G                 OUTP5880
      A1R = SQRT(AX1**2+AY1**2+AZ1**2)                                    OUTP5890
      A2R = SQRT(AX2**2+AY2**2+AZ2**2)                                    OUTP5900
      WRITE(NT,5000) T,AX1,AYI,AZ1,A1R,AX2,AY2,AZ2,A2R                    OUTP5910
      GO TO 600                                                           OUTP5920
   27 NT = NT+1                                                           OUTP5930
      DO 27I I=1,4                                                        OUTP5940
      ORPS(I) = RPSI(I)*TWOPIR                                            OUTP5950
  271 OSLIP(I) = SLIPAV(I)*100.0                                          OUTP5960
      WRITE(NT,5007) T,(OSLIP(I),I=I,4),(RHOSAV(I),I=1,4),                OUTP5970
     1               (ORPS(I),I=1,4)                                      OUTP5980
 5007 FORMAT(IH ,F7.4,8(2X,F8.3),4(2X,F8.2) )                             OUTP5990
      GO TO 600                                                           OUTP6000
   28 NT = NT+1                                                           OUTP6010
      WRITE(NT,5000) T,PP(1),PP(2),(TQB(I),I=I,4),(TAU(I),I=1,4)          OUTP6020
      GO TO 600                                                           OUTP6030
   29 NT = NT+1                                                           OUTP6040
      WRITE(NT,5000) T,(RPSSM(I),I=1,4),(FCSLSM(I),I=1,4)                 OUTP6050
      GO TO 600                                                           OUTP6060
   30 NT = NT+1                                                           OUTP6070
      DPSSO = DPSISF/RAD                                                  OUTP6080
      WRITE(NT,5008) T,DPSSO,ET,DELTAX,APD,FBRK,IGEAR                     OUTP6090
 5008 FORMAT(1H ,F7.3,5(2X,F10.2),I4  )                                   OUTP6100
```

```
600 CONTINUE                                                        OUTP6110
    RETURN                                                          OUTP6120
    ENTRY THPLOT(IPLT)                                              OUTP6130
    GO TO(901,902,903),IPLT                                         OUTP6140
901 WRITE(3) HED,VHED,THED,CHED,GHED,SHED,DADE                      OUTP6150
    RETURN                                                          OUTP6160
902 WRITE(3) T,ULON,VLAT,ACLON,ACLAT,ACVER,ACRES,ROLL,PITCH,        OUTP6170
   1       YAW,PHIO,THTAO,PSIO,AX1,AY1,AZ1,A1R,AX2,AY2,AZ2,A2R      OUTP6180
    RETURN                                                          OUTP6190
903 WRITE(3) (TTTTTT,I=1,21)                                        OUTP6200
    RETURN                                                          OUTP6210
    END                                                             OUTP6220
```

```
      FUNCTION PARI(NN,IA,TSEC,X,Y)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION X(51), Y(51)
      PARI = 0.0
      N = NN
      XX = TSEC
      J = IA
      JA = J+1
      JB = J+2
    5 IF(JB.LE.N) GO TO 10
      J = N
      JA = N-1
      JB = N-2
   10 XA = X(J)
      XB = X(JA)
      XC = X(JB)
      YA = Y(J)
      YB = Y(JA)
   15 YC = Y(JB)
      D1 = (YB-YA)/(XB-XA)
      D2 = (YC-YB)/(XC-XB)
      D3 = (D2-D1)/(XC-XA)
      PARI = YA + (XX-XA) * (D1 + D3 * (XX-XB))
      RETURN
      END
```

```
      SUBROUTINE PINT1(IN,MODE,N,/X/,/HH/,YY,YYP,A)                     00048220
C***********************************************************************00048230
C*                                                                     *00048240
C*     SUBROUTINE PINT1                                                *00048250
C*                                                                     *00048260
C*     PURPOSE                                                         *00048270
C*       TO SOLVE A SYSTEM OF N REAL ORDINARY DIFFERENTIAL EQUATIONS OF *00048280
C*       THE FIRST ORDER                                               *00048290
C*                                                                     *00048300
C*     USAGE                                                           *00048310
C*       CALL PINT1(IN,MODE,N,X,HH,YY,YYP,A)                           *00048320
C*                                                                     *00048330
C*     DESCRIPTION OF PARAMETERS                                       *00048340
C*       N      NUMBER OF EQUATIONS                                    *00048350
C*       IN     INDICATOR FOR INITIALIZATION OF INTEGRATION STEP , IF   *00048360
C*              IN = 1 THE ROUTINE INITIALIZES                         *00048370
C*              IN = 2 THE ROUTINE INTEGRATES ONE STEP                 *00048380
C*       MODE   THE OPTION WORD(=0,1,OR 2) FOR USING ONE OF THE THREE MOD*00048390
C*              ES OF INTEGRATION.   IF MODE EQUALS                    *00048400
C*                 0 - THE ADAMS-MOULTON VARIABLE STEP-SIZE IS USED,    *00048410
C*                 1 - THE RUNGE-KUTTA FIXED STEP-SIZE IS USED,        *00048420
C*                 2 - THE ADAMS FIXED STEP-SIZE IS USED               *00048430
C*       A      IS AN ARRAY OF DIMENSION SIX CONTAINING THE PARAMETERS  *00048440
C*                 FOR THE VARIABLE MODE                               *00048450
C*       X      THE SOURCE VARIABLE                                    *00048460
C*       HH     THE INCREMENT IN SOURCE VARIABLE OR THE STEP SIZE      *00048470
C*       YY     THE TARGET VARIABLES UPDATED BY THIS ROUTINE           *00048480
C*       YYP    THE ARRAY OF FIRST DERIVATIVES OF THE TARGET VARIABLES  *00048490
C*               COMPUTED  IN  THE SUBROUTINE DAUX                     *00048500
 1000 CONTINUE                                                          00048510
C*    METHOD                                                           *00048520
C*       THE ROUTINE USES THE E.K.BLUM MODIFICATION OF THE RUNGE-KUTTA  *00048530
C*       FOURTH-ORDER METHOD,THE FOURTH ORDER ( FIXED AND VARIABLE )    *00048540
C*       ADAMS-MOULTON PREDICTOR -CORRECTOR METHOD.                    *00048550
C*                                                                     *00048560
C*     REMARKS                                                         *00048570
C*     BEFORE EXECUTING THE FIRST PINT1 CALL, THE USER MUST INITIALIZE  *00048580
C*     X,HH, AND EACH OF THE TARGET VARIABLE.                          *00048590
C*                                                                     *00048600
C*       THE SECOND ENTRY POINT ( IN=2 ) MAY BE USED ANY NUMBER OF TIMES*00048610
C*     AFTER THE FIRST PINT1 CALL (IN=1) TO INTEGRATE ONE STEP-SIZE.    *00048620
C*                                                                     *00048630
```

```
C*    SUBROUTINES REQUIRED                                              *
C*       ( ERRMSG ) NOT USED, SEE CARD SERIAL NUMBER05302840            *
C*      THE USER MUST PROVIDE A SUBROUTINE NAMED 'DAUX' WHICH EVALUATES*
C*      THE N DERIVATIVES OF THE SYSTEM OF N FIRST  ORDER DIFFERENIAL   *
C*      EQUATION                                                        *
C*                                                                      *
C*    AUTHOR                                                            *
C*       SQUARE PARTEE                                                  *
C*       AUGUST 1966                                                    *
C*       CORNELL AERONAUTICAL LAB.                                      *
C*                                                                      *
C***********************************************************************
      DIMENSION YY(1),YYP(1),A(1)
      DIMENSION Y(30),YNO(30),YN1(30),YN2(30),YN3(30),YPNO(30),YN(30),
     *          YPN(30),YPN1(30),YPN2(30),YPN3(30),P(30),Q(30)
      DOUBLE PRECISION H,DY,Y,YNO,YN,YN1,YN2,YN3,YPNO,YPN,YPN1,YPN2,
     *                  YPN3,DABS,P,Q
      EQUIVALENCE (YPNO(1),P(1))
      EQUIVALENCE (YNO(1),Q(1))
C
C     MODE=0     VARIABLE ADAMS MOULTON METHOD
C     MODE= 1     FIXED RUNGE-KUTTA
C     MODE= 2     FIXED ADAMS MOULTON METHOD
C
    1 INN = IN
      GO TO (2,50),INN
    2 NMODE = MODE + 1
      NDO = 1
      NGO = 1
      NSS = 1
    3 GO TO (8,50,5),NMODE
    5 NGO = 2
      GO TO 50
C
C     SET UP VARIABLE MODE PARAMETERS
C
    8 NGO = 3
      EMAX = ABS(A(1))
      IF (EMAX.EQ.0.0) EMAX = .1E-03
      EMIN = EMAX * .01
      IF (A(2).NE.0.0) EMIN = EMAX/ABS(A(2))
      AA = ABS(A(3))
      IF (AA.EQ.0.0) AA = 1.0
      HMAX = ABS(A(4))
      IF(HMAX.EQ.0.0) HMAX = 10.E+03
      HMIN = ABS(A(5))
      IF (HMIN.EQ.0.0)  HMIN = .1E-06
      BETA = ABS(A(6))
      IF (BETA.GE.1.0 .OR. BETA.LE.0.0) BETA = .5
      NMSG = 0
C
   50 GO  TO ( 100, 111, 200, 300    ) , NDO
C
C         FIXED RUNGE - KUTTA      INITIALIZATION
  100 DO 102  I=1,N
```

```
      Q(I) = 0.0
      Y(I) = YY(I)
  102 CONTINUE
      NSTEP = 0
  103 CALL  DAUX
  106 XOO = X
      H = HH
      IF (NGO.EQ.1) GO TO 110
  108 XN3 = X
      DO 109  I=1,N
      YPN3(I) = YYP(I)
      YN3(I) = YY(I)
  109 CONTINUE
  110 NDO = 2
      IF (   NSS .EQ. 1 )   RETURN
C
C                 ONE  POINT  INTEGRATE
C
  111 XOO = X
      H = HH
      DO 112  I=1,N
      DY = YYP(I)
      P(I) = H*DY
      Y(I) = Y(I)+.5DO*P(I)
      Q(I) = P(I)
        YY(I)  = Y(I)
  112 CONTINUE
      X = XOO +  .5 * HH
      CALL  DAUX
  113 DO 115  I=1,N
      DY = YYP(I)
      P(I) = H*DY
      Y(I) = Y(I)+.5DO*P(I)-.5DO*Q(I)
      Q(I) = Q(I)/6.DO
        YY(I)  = Y(I)
  115 CONTINUE
  116 CALL  DAUX
  117 DO 120  I=1,N
      DY = YYP(I)
      P(I) = H*DY-.5DO*P(I)
      Y(I) = Y(I)+P(I)
      Q(I) = Q(I)-P(I)
        YY(I)  = Y(I)
  120 CONTINUE
      X = XOO  +  HH
      CALL  DAUX
  121 DO 125  I=1,N
      DY = YYP(I)
      P(I) = H*DY+2.0*P(I)
      Y(I) = Y(I) + Q(I)+P(I)/6.0DO
        YY(I)  = Y(I)
  125 CONTINUE
      CALL  DAUX
C
C            END OF FIXED STEP RUNGE - KUTTA
```

```
C
C
C
  130 IF  ( NGO   .EQ. 1   )     RETURN
  135 NSTEP = NSTEP + 1
      GO TO (136,140,145), NSTEP
C
C     SET UP THREE POINTS FOR ADAMS MOULTON'S
C
  136 XN2 = X
      DO 138  I=1,N
      YPN2(I)  =  YYP(I)
  138 YN2(I) = Y(I)
      RETURN
  140 XN1 = X
      DO 142  I=1,N
      YPN1(I)  =  YYP(I)
      YN1(I) = Y(I)
  142 CONTINUE
      RETURN
  145 XN = X
      DO 146  I=1,N
      YN(I) = Y(I)
      YPN (I)  =  YYP(I)
  146 CONTINUE
      NSTEP = 0
      NFIRST = 1
      NCRE = 0
      NDO = NGO + 1
      RETURN
C
C
C      FIXED ADAMS MOULTON PREDICTOR METHOD
C
  200 XOO = X
      H = HH
      X = XOO  +  HH
      DO 220  I=1,N
      Y(I) = YN(I)+H*(55.D0*YPN(I)-59.D0*YPN1(I)+37.D0*YPN2(I)-9.0D0*
     4   YPN3(I)) / 24.D0
        YY(I)  = Y(I)
  220 CONTINUE
      CALL  DAUX
      DO 225  I=1,N
      DY = YYP(I)
      Y(I) = YN(I)+H*(9.D0*DY   +19.D0*YPN(I)-5.D0*YPN1(I)+YPN2(I))
     5    / 24.D0
        YY(I)  = Y(I)
  225 CONTINUE
      CALL  DAUX
      DO 250  I=1,N
C     SAVE VALUES
      YPN3(I) = YPN2(I)
      YPN2(I) = YPN1(I)
      YPN1(I) = YPN(I)
```

```
      YPN(I)    =   YYP(I)
      YN3(I) = YN2(I)
      YN2(I) = YN1(I)
      YN1(I) = YN(I)
      YN(I) = Y(I)
  250 CONTINUE
  251 XN3 = XN2
      XN2 = XN1
      XN1 = XN
      XN = X
      RETURN
C
C     VARIABLE ADAMS MOULTON METHOD
C
C
  300 XOO = X
      H = HH
      X = XOO   +  HH
      DO 364  I=1,N
       Y(I)= YN(I)+H*(55.DO*YPN(I)-59.DO*YPN1(I)+37.DO*YPN2(I)-9.DO*
     6        YPN3(I)) / 24.DO
        YY(I)  = Y(I)
      P(I) = Y(I)
  364 CONTINUE
      CALL  DAUX
      DO 365  I=1,N
      DY = YYP(I)
      Y(I) = YN(I)+H*(9.DO*DY   +19.DO*YPN(I)-5.DO*YPN1(I)+YPN2(I))
     7          / 24.DO
        YY(I)  = Y(I)
  365 CONTINUE
      CALL  DAUX
C
C     END VARIABLE ADAM MOULTON
C
      ERROR = 0.0
      DO 370  I=1,N
      PRED = SNGL(P(I))
C
C     SAVE VALUES
  366 YPNO(I) = YPN3(I)
      YPN3(I) = YPN2(I)
      YPN2(I) = YPN1(I)
      YPN1(I) = YPN(I)
      YPN(I)  = YYP(I)
      YNO(I) = YN3(I)
      YN3(I) = YN2(I)
      YN2(I) = YN1(I)
      YN1(I) = YN(I)
      YN(I) = Y(I)
      DD = AMAX1(ABS(SNGL(Y(I))),AA)
      DERR = ABS(PRED-SNGL(Y(I)))/(14.0*DD)
      ERROR = AMAX1(ERROR,DERR)
  370 CONTINUE
  375 XNO = XN3
```

```
      XN3 = XN2
      XN2 = XN1
      XN1 = XN
      XN = X
C     ERROR TESTS      ADAMS MOULTON
C
  305 IF (ERROR.GT.EMAX) GO TO 315
      NFIRST = 2
      IF (ERROR.LT.EMIN) GO TO 330
  306 NFIRST = 2
      NCRE = 0
      RETURN
C
C     REDUCE STEP SIZE
C
  315 NSS=2
      NCRE = 0
  316 HH = HH*BETA
      IF( ABS(HH) .GT.  HMIN   )  GO  TO  319
      HH = SIGN( HMIN, HH  )
      IF (NMSG.NE.0) GO TO 306
C
C     CALL ERRMSG(10  ,39H  MINIMUM STEP SIZE  IN  PINT1
      WRITE(6,317)
  317 FORMAT(28H0 MINIMUM STEP SIZE IN PINT1)
C
      NMSG = 1
      A(1) =-A(1)
C
  319 GO TO (     320,325), NFIRST
C     ERROR FIRST VARIABLE POINT
  320 X = XN0
      DO 321  I=1,N
      YY(I) = YN0(I)
  321 CONTINUE
      GO TO   100
C     ERROR DURING VARIABLE MODE
  325 X = XN1
      DO 327  I=1,N
      YY(I) = YN1(I)
  327 CONTINUE
      GO TO   100
C
C     INCREASE STEP SIZE HERE
C
  330 NSS = 1
      NCRE = NCRE + 1
      IF (NCRE.LE.2) RETURN
C     NOW INCREASE
  335 NCRE = 0
      HH = SIGN(AMIN1(ABS(HH/BETA),HMAX),HH)
      GO TO 106
      END
```

```
      SUBROUTINE PLOTTP(IPLT)                                           PLOT0010
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    PLOT0020
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,       PLOT0030
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                       PLOT0040
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,PLOT0050
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  PLOT0060
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,      PLOT0070
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,    PLOT0080
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,    PLOT0090
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),     PLOT0100
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                    PLOT0110
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),PLOT0120
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN  PLOT0130
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             PLOT0140
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))PLOT0150
     1       ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),PLOT0160
     2          (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),         PLOT0170
     3          (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),        PLOT0180
     4          (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),          PLOT0190
     5          (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),            PLOT0200
     6          (PSIFID,VAR(22))                                        PLOT0210
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      PLOT0220
     1          (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))PLOT0230
     2       ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),          PLOT0240
     3          (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),      PLOT0250
     4          (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),     PLOT0260
     5          (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),           PLOT0270
     6          (DPSIFI,DER(21)),(DDPSFI,DER(22))                       PLOT0280
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         PLOT0290
     1          (DER(10),DPHIFD)                                        PLOT0300
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       PLOT0310
     1          (DER(14),DDEL4D)                                        PLOT0320
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,PLOT0330
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),    PLOT0340
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),PLOT0350
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),      PLOT0360
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  PLOT0370
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  PLOT0380
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),      PLOT0390
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),       PLOT0400
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),PLOT0410
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)PLOT0420
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      PLOT0430
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),      PLOT0440
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),    PLOT0450
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                   PLOT0460
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       PLOT0470
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   PLOT0480
     1          (PSII(1),PSI1)                                          PLOT0490
```

```
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,PLOT0500
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,         PLOT0510
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, PLOT0520
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   PLOT0530
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   PLOT0540
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPPLOT0550
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,    PLOT0560
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   PLOT0570
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, PLOT0580
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  PLOT0590
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  PLOT0600
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, PLOT0610
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,PLOT0620
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2PLOT0630
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,PLOT0640
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          PLOT0650
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                 PLOT0660
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)       PLOT0670
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                     PLOT0680
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   PLOT0690
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   PLOT0700
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       PLOT0710
      LOGICAL LCB1,LCB2                                                 PLOT0720
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),AO(4),A1(4),A2(4),A3(4),   PLOT0730
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),     PLOT0740
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)          PLOT0750
      DIMENSION ICONTW(4)                                               PLOT0760
      DATA TTTTTT/-9999.0/                                              PLOT0770
    1 GO TO (2,3,4),IPLT                                                PLOT0780
    2 WRITE(1)HED,DADE,A,B,TS,ZR,RHO,ZF,RW(1),TF,TR                     PLOT0790
      RETURN                                                            PLOT0800
    3 DO 6 J=1,4                                                        PLOT0810
      IF(FRCP(J).GT.0.01) GO TO 5                                       PLOT0820
      ICONTW(J) = 0                                                     PLOT0830
      GO TO 6                                                           PLOT0840
5     ICONTW(J) = 1                                                     PLOT0850
      IF(ABS(BETBR(J)).GT.3.0) ICONTW(J)= -1                            PLOT0860
6     CONTINUE                                                          PLOT0870
      WRITE(1) T,XCP,YCP,ZCP,PHIT,THETT,PSIT,DEL1,DEL2,DEL3,PHIR,PSI1,  PLOT0880
     1 PHI1,PHI2,(XGPP(J),YGPP(J),ZGPP(J),J=1,4),ICONTW                 PLOT0890
      RETURN                                                            PLOT0900
    4 WRITE(1) (TTTTTT,I=1,30)                                          PLOT0910
      RETURN                                                            PLOT0920
      END                                                               PLOT0930
```

```
      SUBROUTINE RUFFRC(I,ZGM)
C           HVOSM-RD2 VERSION
C           HVOSM-VD2 VERSION
C                REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1            CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRE,
     2            PSIFIO,PSIFDO
      DIMENSION YC1P(2)
      EQUIVALENCE (YC1P(1),YC1P)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2            CGYW(4),ZPGI(4),THG1(4),PHG1(4),CPG(4),SPG(4),CTG(4),
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9            FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1            BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2            FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3            F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1            (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIX2P,XIYZP,D1PD2,D1MD2,
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5            XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
```

```
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),A0(4),A1(4),A2(4),A3(4),      RUFF0500
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),        RUFF0510
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)             RUFF0520
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF                                  RUFF0530
      DIMENSION ZGM(2205)                                                   RUFF0540
      DIMENSION FJPP(35)                                                    RUFF0550
      DO 20 N=1,35                                                          RUFF0560
   20 FJPP(N) = FJP(N,I)                                                    RUFF0570
      SNPSI = SIN(PSII(I))                                                  RUFF0580
      CSPSI = COS(PSII(I))                                                  RUFF0590
      SNPHI = SIN(PHII(I))                                                  RUFF0600
      CSPHI = COS(PHII(I))                                                  RUFF0610
      SFRX(I) = 0.0                                                         RUFF0620
      SFRY(I) = 0.0                                                         RUFF0630
      SFRZ(I) = 0.0                                                         RUFF0640
      TTAJ21 = CSPHI*SNPSI                                                  RUFF0650
      TTAJ31 = SNPHI*SNPSI                                                  RUFF0660
      AJMTX(1,2) = -SNPSI                                                   RUFF0670
      AJMTX(2,2) = CSPHI*CSPSI                                              RUFF0680
      AJMTX(3,2) = SNPHI*CSPSI                                              RUFF0690
      INDF = 0                                                              RUFF0700
      INDL = 0                                                              RUFF0710
      MF = IFIX((XP(I)-RW(I))/DELG)                                         RUFF0720
      ML = MF+IFIX(2.0*RW(I)/DELG)                                          RUFF0730
      IF(MF.GE.1) GO TO 10                                                  RUFF0740
      MF = 1                                                                RUFF0750
      INDF = 1                                                              RUFF0760
   10 IF(ML.LE.NEND) GO TO 11                                               RUFF0770
      ML = NEND                                                             RUFF0780
      INDL = 1                                                              RUFF0790
   11 DO 100 J=1,21                                                         RUFF0800
      THTJ = (-44.0+4.0*J)*RAD                                              RUFF0810
      STJ = SIN(THTJ)                                                       RUFF0820
      CTJ = COS(THTJ)                                                       RUFF0830
      AJMTX(1,1) = CTJ*CSPSI                                                RUFF0840
      AJMTX(2,1) = TTAJ21*CTJ+SNPHI*STJ                                     RUFF0850
      AJMTX(3,1) = TTAJ31*CTJ-CSPHI*STJ                                     RUFF0860
      AJMTX(1,3) = CSPHI*STJ                                                RUFF0870
      AJMTX(2,3) = TTAJ21*STJ-SNPHI*CTJ                                     RUFF0880
      AJMTX(3,3) = TTAJ31*STJ+CSPHI*CTJ                                     RUFF0890
      DO 8 K=1,3                                                            RUFF0900
      DO 7 L=1,3                                                            RUFF0910
      BMTX(K,L) = 0.0                                                       RUFF0920
      DO 6 M=1,3                                                            RUFF0930
    6 BMTX(K,L) = BMTX(K,L)+AMTX(K,M)*AJMTX(M,L)                            RUFF0940
    7 CONTINUE                                                              RUFF0950
    8 CONTINUE                                                              RUFF0960
      IF(BMTX(3,3).EQ.0.0) GO TO 100                                        RUFF0970
      DO 50 M=MF,ML                                                         RUFF0980
      ZM1 = 0.0                                                             RUFF0990
      IF(M.LT.NEND) ZM1 = ZGM(M+1)                                          RUFF1000
```

```
      XM = DELG*(M-1)
      XM1 = XM+DELG
      TMP = (BMTX(1,3)*(ZM1-ZGM(M))/(BMTX(3,3)*DELG))
      IF(TMP.EQ.1) GO TO 50
      XD = XM-XP(I)
      XD1 = XM1-XP(I)
      IF(ABS(XD).LT.0.001.OR.ABS(XD1).LT.0.001) GO TO 49
      SM = (ZGM(M)-ZP(I))/XD
      SM1 = (ZM1-ZP(I))/XD1
      SGM = SIGN(1.0,SM)
      SGM1 = SIGN(1.0,SM1)
      IF(SGM.GT.0.0.AND.SGM1.LT.0.0) GO TO 50
      IF(SGM.EQ.SGM1.AND.SM1.GT.SM) GO TO 50
   49 TMP1 = 1.0/(1.0-TMP)
      XJP = TMP1*(XP(I)+BMTX(1,3)*(ZGM(M)-ZP(I)-XM*
     1     (ZM1-ZGM(M))/DELG)/BMTX(3,3))
      IF(XJP.GE.XM) GO TO 60
      IF(XJP.GE.0.0) GO TO 100
      GO TO 69
   60 IF(XJP.LE.XM1) GO TO 70
   50 CONTINUE
      GO TO 69
   70 IF(ABS(BMTX(1,3)).LT.0.0001) GO TO 71
      HJ = (XJP-XP(I))/BMTX(1,3)
      ZJP = ZP(I)+BMTX(3,3)*HJ
      GO TO 72
   71 ZJP = ZGM(M)+(XJP-XM)*(ZM1-ZGM(M))/DELG
      HJ = (ZJP-ZP(I))/BMTX(3,3)
      XJP = XP(I)
      GO TO 72
   69 ZJP = 0.0
      XJP = XP(I)-BMTX(1,3)*ZP(I)/BMTX(3,3)
      HJ = -ZP(I)/BMTX(3,3)
   72 YJP = YP(I)+BMTX(2,3)*HJ
      IF(HJ.LT.0.0.OR.HJ.GT.RW(I)) GO TO 100
      CAJ = (XP(I)-XJP)/HJ
      CBJ = (YP(I)-YJP)/HJ
      CGJ = (ZP(I)-ZJP)/HJ
      CALL INTRPL(FJPP,RWHJB,RWHJE,DRWHJ,RW(I)-HJ,FJ)
      SFRX(I) = SFRX(I)+FJ*CAJ
      SFRY(I) = SFRY(I)+FJ*CBJ
      SFRZ(I) = SFRZ(I)+FJ*CGJ
  100 CONTINUE
      FR(I) = SQRT(SFRX(I)**2+SFRY(I)**2+SFRZ(I)**2)
      IF(FR(I).NE.0.0) GO TO 110
      CAR(I) = 0.0
      CBR(I) = 0.0
      CGR(I) = 1.0
      HI(I) = RW(I)
      PHGI(I) = 0.0
      THGI(I) = 0.0
```

```
      SPG(I) = 0.0
      TXGP = XP(I)
      GO TO 112
  110 CAR(I) = -SFRX(I)/FR(I)
      CBR(I) = -SFRY(I)/FR(I)
      CGR(I) = -SFRZ(I)/FR(I)
      HI(I) = RW(I)-FR(I)/AKT(I)
      IF(HI(I).GT.RW(I)-SIGT(I)) GO TO 111
      HI(I) = RW(I)-(FR(I)/AKT(I)+SIGT(I)*(XLAMT(I)-1.0))/XLAMT(I)
  111 TXGP = XP(I)+HI(I)*CAR(I)
      ME = TXGP/DELG+1
      TPHGI = 0.0
      TTHGI = ATAN2((ZGM(ME)-ZGM(ME+1)),DELG)
      TAI = CBR(I)*CGYW(I)-CGR(I)*CBYW(I)
      TBI = CGR(I)*CAYW(I)-CAR(I)*CGYW(I)
      TCI = CAR(I)*CBYW(I)-CBR(I)*CAYW(I)
      STI = SIN(TTHGI)
      CTI = COS(TTHGI)
      DN1 = (TCI*TCI+TBI*TBI)*STI-TAI*TCI*CTI
      DN2 = -TBI*(TAI*STI+TCI*CTI)
      DN3 = (TAI*TAI+TBI*TBI)*CTI-TAI*TCI*STI
      TERM5 = SQRT(DN1*DN1+DN2*DN2+DN3*DN3)
      SPG(I) = -DN2/TERM5
      PHGI(I) = ARSIN(SPG(I))
      THGI(I) = ATAN(DN1/DN3)
  112 CPG(I) = COS(PHGI(I))
      CTG(I) = COS(THGI(I))
      STG(I) = SIN(THGI(I))
      XGPP(I) = TXGP
      YGPP(I) = YP(I)+HI(I)*CBR(I)
      ZGPP(I) = ZP(I)+HI(I)*CGR(I)
      RETURN
      END
```

```
      SUBROUTINE RUFRED(NEND,DELG,DGMAX,ZRTAB)
C              HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
C              HVOSM-RD2 VERSION
C          HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975  CALSPAN CORPORATION
      DIMENSION ZRTAB(2205)
      IF(NEND.GT.2200) GO TO 900
      READ(4,END=901) (ZRTAB(I),I=1,NEND)
      GO TO 12
  901 WRITE(6,9001)
 9001 FORMAT('      END OF FILE ENCOUNTERED IN READ OF ROUGHNESS '/
     1    ' DATA BEFORE NEND POINTS WERE READ.'  )
      NEND = I
   12 DGMAX = (NEND-1)*DELG
      RETURN
  900 WRITE(6,9000)
 9000 FORMAT('      NUMBER OF LAST ROUGHNESS DATA POINT IS GREATER '/
     1    ' THAN THE ALLOWED 2200.  PROGRAM TERMINATED.'  )
      STOP
      END
```

```
      SUBROUTINE SIMSOL (A,KK,LL)
C***********************************************************************
C*                                                                     *
C*    SUBROUTINE SIMSOL      (SINGLE PRECISION VERSION)                *
C*                                                                     *
C*    AUTHOR                                                           *
C*       DR.JOHN T. FLECK                                              *
C*       (REVISED BY F.E. BUTLER)                                      *
C*                                                                     *
C*    REFERENCE                                                        *
C*       ''SUBROUTINES TO SOLVE AN INDEPENDENT SET OF LINEAR           *
C*       SIMULTANEOUS EQUATIONS''  HS/FEB/PAW-84, 21 JULY 1965.        *
C*                                                                     *
C*    PURPOSE                                                          *
C*       TO SOLVE A SET OF SIMULTANEOUS LINEAR EQUATIONS, AX=B.        *
C*                                                                     *
C*    USAGE                                                            *
C*       CALL SIMSOL (A,N,ND1)                                         *
C*                                                                     *
C*    DESCRIPTION OF PARAMETERS                                        *
C*       A - IS A 2-DIMENSIONAL (ND1*ND2) MATRIX OF COEFFICIENTS.      *
C*       N - IS THE NUMBER OF EQUATIONS AND UNKNOWNS.                  *
C*       ND1 - IS THE FIRST DIMENSION OF A IN CALLING PROGRAM.         *
C*            ( ND1.GE.N  AND  ND2.GE.N+1 )                            *
C*                                                                     *
C*    CALLING PROGRAM SETUP                                            *
C*       A(I,J) FOR I,J=1,N                                            *
C*       A(I,N+1)=B(I) FOR I=1,N                                       *
C*       THE SOLUTION IS RETURNED IN COLUMN N+1 OF MATRIX A.           *
C*       MATRIX A IS DESTROYED BY THE SUBROUTINE.                      *
 1000 CONTINUE
C*    REMARKS                                                          *
C*       IF MATRIX A IS SINGULAR, AN ERROR MESSAGE IS PRINTED          *
C*       AND THE JOB IS TERMINATED.                                    *
C*                                                                     *
C*    METHOD                                                           *
C*       SOLUTION IS OBTAINED BY ELIMINATION USING LARGEST PIVOTAL     *
C*       DIVISOR OF EACH COLUMN. EACH STAGE OF ELIMINATION CONSISTS    *
C*       OF INTERCHANGING ROWS WHEN NECESSARY TO AVOID DIVISION BY     *
C*       ZERO OR SMALL NUMBERS.                                        *
C*       THE FORWARD SOLUTION TO OBTAIN VARIABLE N IS DONE IN N        *
C*       STAGES. THE BACK SOLUTION FOR THE OTHER VARIABLES IS          *
C*       CALCULATED BY SUCCESSIVE SUBSTITUTIONS. FINAL SOLUTION        *
C*       VALUES ARE DEVELOPED IN COLUMN N+1 OF MATRIX A, WITH          *
C*       VARIABLE 1 IN A(1,N+1), VARIABLE 2 IN A(2,N+1),.....,         *
C*       VARIABLE N IN A(N,N+1).                                       *
C*                                                                     *
C***********************************************************************
      REAL A(LL,1),B,BIG
      N  = KK
      N1 = N+1
      DO 50  L=1,N
      L1 = L+1
      BIG = 0.0
      DO 25  I=L,N
      IF (ABS(A(I,L)).LE.ABS(BIG)) GO TO 25
      K = I
      BIG = A(I,L)
```

```
   25 CONTINUE
      IF (BIG.NE.0.0) GO TO 30
      WRITE(6,32000)
32000  FORMAT(24H SIMSOL MATRIX SINGULAR.)
      STOP
   30 DO 40  J=L,N1
      IF (K.EQ.L) GO TO 40
      B      = A(K,J)
      A(K,J) = A(L,J)
      A(L,J) = B
   40 A(L,J) = A(L,J)/BIG
      IF (L.EQ.N) GO TO 50
      DO 48  I=L1,N
      IF (A(I,L).EQ.0.0) GO TO 48
      DO 45  J=L1,N1
   45 A(I,J) = A(I,J)-A(I,L)*A(L,J)
   48 CONTINUE
   50 CONTINUE
      IF (N.EQ.1) RETURN
      N2 = N-1
      DO 60  L=1,N2
      I  = N-L
      L1 = I+1
      DO 60  J=L1,N
   60 A(I,N1) = A(I,N1)-A(I,J)*A(J,N1)
      RETURN
      END
```

```
      SUBROUTINE SUSFRC(DISP,VEL)                                       SUSF0010
C           HVOSM-VD2 VERSION                                           SUSF0020
C           REVISED OCTOBER 1975   CALSPAN CORPORATION                  SUSF0030
C                                                                       SUSF0040
C  SUBROUTINE TO COMPUTE SUSPENSION FORCES ACTING BETWEEN SPRUNG        SUSF0050
C  AND UNSPRUNG MASSES                                                  SUSF0060
C                                                                       SUSF0070
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    SUSF0080
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      SUSF0090
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      SUSF0100
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,SUSF0110
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  SUSF0120
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     SUSF0130
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   SUSF0140
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   SUSF0150
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    SUSF0160
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  SUSF0170
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),SUSF0180
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN SUSF0190
      COMMON/INPT3/ AKFC,AKFCP,OMEGFC,AKFE,AKFEP,OMEGFE,AKRC,AKRCP,     SUSF0200
     1             OMEGRC,AKRE,AKREP,OMEGRE,END3                        SUSF0210
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             SUSF0220
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))SUSF0230
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),SUSF0240
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        SUSF0250
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       SUSF0260
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         SUSF0270
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           SUSF0280
     6           (PSIFID,VAR(22))                                       SUSF0290
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      SUSF0300
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))SUSF0310
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       SUSF0320
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     SUSF0330
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    SUSF0340
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          SUSF0350
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      SUSF0360
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         SUSF0370
     1           (DER(10),DPHIFD)                                       SUSF0380
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       SUSF0390
     1           (DER(14),DDEL4D)                                       SUSF0400
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,SUSF0410
     1            PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),   SUSF0420
     2            CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),SUSF0430
     3            STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),     SUSF0440
     4            XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),  SUSF0450
     5            YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),  SUSF0460
     6            CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),     SUSF0470
     7            CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),      SUSF0480
     8            SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),SUSF0490
```

```
9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)SUSF0500
 COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),         SUSF0510
1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),       SUSF0520
2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),     SUSF0530
3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                    SUSF0540
 DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                          SUSF0550
 EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),      SUSF0560
1            (PSII(1),PSI1)                                           SUSF0570
 COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,   SUSF0580
1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,            SUSF0590
2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,    SUSF0600
3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,      SUSF0610
4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,      SUSF0620
5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPSUSF0630
6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,       SUSF0640
7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,      SUSF0650
8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,    SUSF0660
9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ     SUSF0670
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,     SUSF0680
1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,    SUSF0690
2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,SUSF0700
3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2SUSF0710
4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,SUSF0720
5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             SUSF0730
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                    SUSF0740
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)          SUSF0750
 COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                   SUSF0760
1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,          SUSF0770
2              SLOPE1,SLOPE2,XTRA(300)                                SUSF0780
 DIMENSION UI(4),VI(4),WI(4)                                          SUSF0790
 EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                         SUSF0800
 DIMENSION APITCH(4)                                                  SUSF0810
 EQUIVALENCE (APITCH(1),APTCH1)                                       SUSF0820
 COMMON/APTABL/ APFR(21,2),IAPFR(2),DAPFB,DAPFE,DDAPF,NAPF,           SUSF0830
1               DAPRB,DAPRE,DDAPR,NAPR                                SUSF0840
 DIMENSION APF(21),APR(21)                                            SUSF0850
 EQUIVALENCE (APFR(1,1),APF(1)),(APFR(1,2),APR(1))                    SUSF0860
 EQUIVALENCE (IAPF,IAPFR(1)),(IAPR,IAPFR(2))                          SUSF0870
 COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,          SUSF0880
1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),SUSF0890
2               NCAMF,NCAMR,NDTHF,NDTHR                               SUSF0900
 COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,         SUSF0910
1                RHF2MF,RF2MF1,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,        SUSF0920
2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,       SUSF0930
3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,       SUSF0940
4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,           SUSF0950
5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                        SUSF0960
 DIMENSION DISP(4),VEL(4),F1I(4),F2I(4)                               SUSF0970
 EQUIVALENCE (F1I(1),F1FI(1)),(F2I(1),F2FI(1))                        SUSF0980
C                                                                     SUSF0990
 DO 500 I=1,4                                                         SUSF1000
```

```
      IF(I.GE.3) GO TO 200
      IF(EPSF.LE.0.0) GO TO 10
      IF(ABS(VEL(I)).GE.EPSF) GO TO 10
      F1I(I) = (CFP/EPSF)*VEL(I)
      GO TO 20
   10 F1I(I) = SIGN(CFP,VEL(I))
   20 XLM = 1.0
      TMP = DISP(I)*VEL(I)
      IF(DISP(I).GT.OMEGFE) GO TO 30
      IF(DISP(I).LT.OMEGFC) GO TO 40
      F2I(I) = AKF*DISP(I)
      GO TO 100
   30 IF(TMP.LT.0.0) XLM = XLAMF
      DISP1 = DISP(I)-OMEGFE
      F2I(I) = AKF*DISP(I)+XLM*(AKFE*DISP1+AKFEP*DISP1**3)
      GO TO 100
   40 IF(TMP.LT.0.0) XLM = XLAMF
      DISP1 = DISP(I)-OMEGFC
      F2I(I) = AKF*DISP(I)+XLM*(AKFC*DISP1+AKFCP*DISP1**3)
      GO TO 100
  100 IF(IAPF.EQ.0) GO TO 150
      APITCH(I) = 0.0
      IF(FC(I).EQ.0.0) GO TO 150
      TMP3 = COS(PHII(I))*COS(PSII(I))/12.0
      CALL INTRPL(APF,DAPFB,DAPFE,DDAPF,DISP(I),APC)
      APITCH(I) = -APC*FC(I)*HI(I)*TMP3
  150 ABAR = RFTF*D21
      IF(ISUS.EQ.2) GO TO 105
      IF(I.EQ.2) GO TO 102
      FJF(1) = -SLOPE1*(FYU(1)*HCGH1-FZU(1)*HCBH1) + FYU(1)*DTDD1
      GO TO 103
  102 ABAR = -ABAR
      FJF(2) = -SLOPE2*(FYU(2)*HCGH2-FZU(2)*HCBH2) - FYU(2)*DTDD2
      GO TO 103
  105 ABAR =-RTF*PHIF
      IF(I.EQ.2) ABAR = -ABAR
      FJF(I) = 0.0
  103 SI(I) = BO2APB-CF*VEL(I)-F1I(I)-F2I(I)+ABAR+FJF(I)+APITCH(I)
      GO TO 500
  200 IF(EPSR.LE.0.0) GO TO 210
      IF(ABS(VEL(I)).GE.EPSR) GO TO 210
      F1I(I) = (CRP/EPSR)*VEL(I)
      GO TO 220
  210 F1I(I) = SIGN(CRP,VEL(I))
  220 XLM = 1.0
      TMP = DISP(I)*VEL(I)
      IF(DISP(I).GT.OMEGRE) GO TO 230
      IF(DISP(I).LT.OMEGRC) GO TO 240
      F2I(I) = AKR*DISP(I)
      GO TO 300
  230 IF(TMP.LT.0.0) XLM = XLAMR
```

```
      DISP1 = DISP(I)-OMEGRE                                             SUSF1520
      F2I(I) = AKR*DISP(I)+XLM*(AKRE*DISP1+AKREP*DISP1**3)               SUSF1530
      GO TO 300                                                          SUSF1540
  240 IF(TMP.LT.0.0) XLM = XLAMR                                         SUSF1550
      DISP1 = DISP(I)-OMEGRC                                             SUSF1560
      F2I(I) = AKR*DISP(I)+XLM*(AKRC*DISP1+AKRCP*DISP1**3)               SUSF1570
  300 IF(IAPR.EQ.0) GO TO 350                                            SUSF1580
      APITCH(I) = 0.0                                                    SUSF1590
      IF(FC(I).EQ.0.0) GO TO 350                                         SUSF1600
      TMP3 = COS(PHII(I))*COS(PSII(I))/12.0                              SUSF1610
      CALL INTRPL(APR,DAPRB,DAPRE,DDAPR,DISP(I),APC)                     SUSF1620
      APITCH(I) =  APC*FC(I)*HI(I)*TMP3                                  SUSF1630
  350 ABAR = RRTR*D43                                                    SUSF1640
      IF(ISUS.NE.1) GO TO 305                                            SUSF1650
      IF(I.EQ.4) GO TO 302                                               SUSF1660
      FJF(3) = -SLOPE3*(FYU(3)*HCGH3-FZU(3)*HCBH3) + FYU(3)*DTDD3        SUSF1670
      GO TO 303                                                          SUSF1680
  302 ABAR = - ABAR                                                      SUSF1690
      FJF(4) = -SLOPE4*(FYU(4)*HCGH4-FZU(4)*HCBH4) - FYU(4)*DTDD4        SUSF1700
      GO TO 303                                                          SUSF1710
  305 ABAR = -RTR*PHIR                                                   SUSF1720
      IF(I.EQ.4) ABAR = -ABAR                                            SUSF1730
      FJF(I) = 0.0                                                       SUSF1740
  303 SI(I) = AO2APB-CR*VEL(I)-F1I(I)-F2I(I)+ABAR+APITCH(I)+FJF(I)       SUSF1750
  500 CONTINUE                                                           SUSF1760
      SFZ1 = SI(1)+SI(2)+SI(3)+SI(4)                                     SUSF1770
      RETURN                                                             SUSF1780
      END                                                                SUSF1790
```

```
      SUBROUTINE TEREAD(I,NNBX,NNBY,NNX,NNY,NZ5T,NERR)                  TERE0010
C               HVOSM-VD2 VERSION                                       TERE0020
C               REVISED OCTOBER 1975   CALSPAN CORPORATION              TERE0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    TERE0040
     1            A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,     TERE0050
     2            PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                     TERE0060
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,TERE0070
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, TERE0080
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,    TERE0090
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,  TERE0100
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,  TERE0110
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),   TERE0120
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                 TERE0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),TERE0140
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN TERE0150
      DIMENSION DUM(18)                                                 TERE0160
      LSEQ = 0                                                          TERE0170
      IF(NNBX.LE.0) GO TO 10                                            TERE0180
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD                            TERE0190
 2000 FORMAT(9F8.0,2I4)                                                 TERE0200
      IF(NSEQ.LT.LSEQ) GO TO 98                                         TERE0210
      LSEQ = NSEQ                                                       TERE0220
      IF(NNBX.GT.4) GOTO 98                                             TERE0230
      DO 11 K=1,NNBX                                                    TERE0240
   11 XBDRY(K,I) = DUM(K)                                               TERE0250
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD                            TERE0260
      IF(NSEQ.LT.LSEQ) GO TO 98                                         TERE0270
      LSEQ = NSEQ                                                       TERE0280
      DO 12 K=1,NNBX                                                    TERE0290
   12 PSBDRO(K,I) = DUM(K)                                              TERE0300
   10 IF(NNBY.LE.0) GO TO 20                                            TERE0310
      IF(NNBY.GT.2) GO TO 98                                            TERE0320
      READ(2,2000) (DUM(K),K=1,9),NSEQ,NCARD                            TERE0330
      IF(NSEQ.LT.LSEQ) GO TO 98                                         TERE0340
      LSEQ = NSEQ                                                       TERE0350
      DO 13 K=1,NNBY                                                    TERE0360
   13 YBDRY(K,I) = DUM(K)                                               TERE0370
   20 NYCDS = (NNY-1)/9+1                                               TERE0380
      DO 30 J=1,NNX                                                     TERE0390
      M = 0                                                             TERE0400
      DO 40 K=1,NYCDS                                                   TERE0410
      READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD                            TERE0420
      IF(NSEQ.LT.LSEQ) GO TO 98                                         TERE0430
      LSEQ = NSEQ                                                       TERE0440
      DO 50 N=1,9                                                       TERE0450
      M = M+1                                                           TERE0460
      ZGP(J,M,I) = DUM(N)                                               TERE0470
      IF(M.GE.NNY) GO TO 30                                             TERE0480
   50 CONTINUE                                                          TERE0490
```

```
40 CONTINUE
30 CONTINUE
   IF(NZ5T.EQ.0) GO TO 99
   M = 0
   DO 60 K=1,NYCDS
   READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD
   IF(NSEQ.LT.LSEQ) GO TO 98
   LSEQ = NSEQ
   DO 61 N=1,9
   M = M+1
   YYZGP5(M) = DUM(N)
   IF(M.GE.NNY) GO TO 70
61 CONTINUE
60 CONTINUE
70 NXCDS = (NNX-1)/9 + 1
   M = 0
   DO 71 K=1,NXCDS
   READ(2,2000) (DUM(N),N=1,9),NSEQ,NCARD
   IF(NSEQ.LT.LSEQ) GO TO 98
   LSEQ = NSEQ
   DO 72 N=1,9
   M = M+1
   XXZGP5(M) = DUM(N)
   IF(M.GE.NNX) GO TO 99
72 CONTINUE
71 CONTINUE
98 NERR = 1
99 RETURN
   END
```

```
      SUBROUTINE TIRFR                                                  TIRF0010
C            HVOSM-VD2 VERSION                                          TIRF0020
C            REVISED OCTOBER 1975   CALSPAN CORPORATION                 TIRF0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    TIRF0040
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,       TIRF0050
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                       TIRF0060
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,TIRF0070
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  TIRF0080
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,      TIRF0090
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,    TIRF0100
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,    TIRF0110
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),     TIRF0120
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                   TIRF0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),TIRF0140
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN TIRF0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             TIRF0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))TIRF0170
     1        ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),TIRF0180
     2         (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),          TIRF0190
     3         (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),         TIRF0200
     4         (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),           TIRF0210
     5         (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),             TIRF0220
     6         (PSIFID,VAR(22))                                         TIRF0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      TIRF0240
     1         (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))TIRF0250
     2        ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),         TIRF0260
     3         (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),       TIRF0270
     4         (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),      TIRF0280
     5         (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),            TIRF0290
     6         (DPSIFI,DER(21)),(DDPSFI,DER(22))                        TIRF0300
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,TIRF0310
     1          PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),     TIRF0320
     2          CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),TIRF0330
     3          STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),       TIRF0340
     4          XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),    TIRF0350
     5          YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),    TIRF0360
     6          CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),       TIRF0370
     7          CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),        TIRF0380
     8          SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),TIRF0390
     9          FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)TIRF0400
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),      TIRF0410
     1          BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),       TIRF0420
     2          FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),     TIRF0430
     3          F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                    TIRF0440
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                       TIRF0450
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),   TIRF0460
     1         (PSII(1),PSI1)                                           TIRF0470
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,TIRF0480
     1         GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,            TIRF0490
```

```
2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  TIRF0500
3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    TIRF0510
4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    TIRF0520
5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPTIRF0530
6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     TIRF0540
7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    TIRF0550
8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  TIRF0560
9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   TIRF0570
 COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   TIRF0580
1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  TIRF0590
2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,TIRF0600
3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2TIRF0610
4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,TIRF0620
5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           TIRF0630
 DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  TIRF0640
 EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        TIRF0650
 COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                      TIRF0660
1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,    TIRF0670
2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),    TIRF0680
3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)        TIRF0690
 LOGICAL LCB1,LCB2                                                  TIRF0700
 COMMON /INPT4/ FIDJ(2),FIWJ(2),ARBR(2),NXUGMU,NXFRCP,XXUGMU(6),    TIRF0710
A               XXFRCP(6),XMUMAT(6,6,4),XMXPMT(6,6,4),              TIRF0720
B               XMXSMT(6,6,4),SLIPMT(6,6,4),CT(4),XMUM(4),          TIRF0730
C               XMUXP(4),XMUXS(4),RRMC(4),RRM(4),COMEN4             TIRF0740
 EQUIVALENCE (FIDJ(1),FIDJF),(FIDJ(2),FIDJR),(FIWJ(1),FIWJF)        TIRF0750
 EQUIVALENCE (FIWJ(2),FIWJR),(ARBR(1),ARBRF),(ARBR(2),ARBRR)        TIRF0760
 COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),     TIRF0770
1              TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,TIRF0780
2        DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),   TIRF0790
3              SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),  TIRF0800
4              EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),      TIRF0810
5              FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),   TIRF0820
6              SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),      TIRF0830
7              FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),      TIRF0840
8           ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),TIRF0850
9           RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD   TIRF0860
 COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP  TIRF0870
 LOGICAL  IUVS,IUVB,IRPS                                            TIRF0880
 COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,  TIRF0890
1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),      TIRF0900
2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)TIRF0910
3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,         TIRF0920
4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)            TIRF0930
 COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,TIRF0940
1              NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP,  TIRF0950
2              RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,     TIRF0960
3              DELTAE,PIO15R,COMEN5,TQE,RPME                        TIRF0970
 COMMON /INTR/ NEQR,TIMR,DTR,VARR(12),DERR(12)                      TIRF0980
 DIMENSION RPSI(4),DRPSI(4)                                         TIRF0990
 EQUIVALENCE(VARR(1),RPSI(1)),(DERR(1),DRPSI(1))                    TIRF1000
```

```
C
C      VARIABLES NO LONGER USED INCLUDE TQF(50),TQR(50),TI(4),TIHI(4)
C
C
C
C     FOR ISTEP=1, TIME T IS A HALF-INTERVAL AHEAD OF CLOSING TIME FOR
C      'OUTER' INTEGRATION, SO TABLE VALUES ARE FOUND AT TIME T,
C      AND THE INTEGRATED VARIABLES ARE SET AT THIS NEW HALF INTERVAL,
C      USING THE DERIVATIVE VALUES COMPUTED AT CLOSE OF PREVIOUS
C      INTERVAL, SAID DERIVATIVES HAVING BEEN COMPUTED WITH THE FINAL
C      VARIABLE VALUES FOR THE INTEGRATION OF THE PREVIOUS INTERVAL.
C        FOR ISTEP=2, TIME T IS SAME AS FOR ISTEP=1, AND THE INTEGRATED
C       VARIABLES ARE REEVALUATED, USING THE DERIVATIVES FROM ISTEP=1.
C        FOR ISTEP=3, TIME T IS A FULL INTERVAL AHEAD,AND THE INTEGRATED
C      VARIABLES ARE REEVALUATED,USING THE DERIVATIVES FROM ISTEP=2.
C        FOR ISTEP=4, TIME T IS SAME AS FOR ISTEP=3,AND INTEGRATED
C      VARIABLES ARE UPDATED FINALLY BY THE AVERAGING.  NEW DERIVATIVES
C      USING THESE FINAL VALUES OF THE INTEGRATED VARIABLES ARE
C      COMPUTED, READY FOR NEXT INTERVAL.
C
      DO 15 I=1,4
      SPHICI(I) = CAYW(I)*CAGZ(I)+CBYW(I)*CBGZ(I)+CGYW(I)*CGGZ(I)
      PHICI(I)  = ARSIN(SPHICI(I))
   15 CONTINUE
      ISTEP = ISTEP + 1
      IF(ISTEP.EQ.2.OR.ISTEP.EQ.4) RETURN
C           RETAIN FC,FS, RPSI, DRPSI AS COMPUTED TO THE HALF-STEP ONCE,
C                 TO THE SECOND HALF-STEP ONCE.
      STEPD = 1.0
      IF (MODE -1)  17,16,17
   16 STEPD = 0.5
   17 DO 21 I=1,4
      FC(I)    = 0.0
      FS(I)    = 0.0
      FRCP(I) = 0.0
      FCXU(I) = 0.0
      FCYU(I) = 0.0
      FCZU(I) = 0.0
      FRXU(I) = 0.0
      FRYU(I) = 0.0
      FRZU(I) = 0.0
      FSXU(I) = 0.0
      FSYU(I) = 0.0
      FSZU(I) = 0.0
      FXU(I)  = 0.0
      FYU(I)  = 0.0
      FZU(I)  = 0.0
      SFCDTR(I) = 0.0
      SFSDTR(I) = 0.0
      SFRCPR(I) = 0.0
      SSLIP (I) = 0.0
```

```
      FCAV  (I) = 0.0
      FSAV  (I) = 0.0
      FRCPAV(I) = 0.0
      SLIPAV(I) = 0.0
      SRHOS(I) = 0.0
      RHOSAV(I) = 0.0
C       NOTE THAT RPSSM  AND FCSLSM ARE NOT CHANGED.
C            FORMERLY, TIRFRC(I) WAS CALLED FROM VGORNT SEPARATELY FOR
C          EACH WHEEL. THE SUM OF THE FORCES FOR ALL WHEELS WAS THEREFOR
C             NOT ZEROED IN TIRFRC BUT IN VPOS FOR EACH RUNGE-KUTTA STEP
C             SFXU,SFYU,SFZU
   21 CONTINUE
      SFXU = 0.0
      SFYU = 0.0
      SFZU = 0.0
      DO 23 I=1,4
      IF (FR(I).NE.0.0) GO TO 30
      IF(TQB(I).NE.0.0) GO TO 30
   23 CONTINUE
      RETURN
C           AT 30 RADIAL TIRE FORCE IS NON-ZERO FOR AT LEAST ONE WHEEL
C             OR TORQUE IS NON-ZERO FOR AT LEAST ONE WHEEL.
   30 DO 31 I=1,4
      CPHICI(I) = COS(PHICI(I))
      PSITEM(I) = PSIIP(I)* SIGN(1.0,UG(I))
C           UGW(I) IS VELOCITY IN WHEEL PLANE
      UGW(I) = UG(I) * COS(PSIIP(I))  + VG(I) * SIN(PSIIP(I))
      ABSUGW(I) = ABS(UGW(I))
C      IUVS FOR SLIP TEST,  IUVB FOR FS  (SIDE FORCE) TEST
      IUVS(I) = .FALSE.
      IF( ABSUGW(I)  .GE. 0.5) IUVS(I) = .TRUE.
      IUVB(I) = .FALSE.
      IF(ABS(UG(I)).GT.0.5  .OR. ABS(VG(I)).GT.0.5) IUVB(I) = .TRUE.
      TERM(I) = 0.0
      IF(.NOT.IUVB(I)) GO TO 301
      IF(UG(I).NE.0.0.OR.VG(I).NE.0.0) TERM(I)=ATAN2(VG(I),ABS(UG(I)))
  301  TERMPI = TAN(TERM(I)-PSITEM(I))
      TERMP(I)= TERMPI * TERMPI
      TERMB(I)= PHICI(I) - 0.6366*PHICI(I) * ABS(PHICI(I))
      FSXFAC(I) =  AMTX(I,I)*CAS(I)+AMTX(2,I)*CBS(I)+AMTX(3,1)*CGS(I)
      FSYFAC(I) =  AMTX(1,2)*CAS(I)+AMTX(2,2)*CBS(I)+AMTX(3,2)*CGS(I)
      FSZFAC(I) =  AMTX(1,3)*CAS(I)+AMTX(2,3)*CBS(I)+AMTX(3,3)*CGS(I)
      FRXFAC(I) =  AMTX(1,1)*CAGZ(I)+AMTX(2,1)*CBGZ(I)+AMTX(3,1)*CGGZ(I)
      FRYFAC(I) =  AMTX(1,2)*CAGZ(I)+AMTX(2,2)*CBGZ(I)+AMTX(3,2)*CGGZ(I)
      FRZFAC(I) =  AMTX(I,3)*CAGZ(I)+AMTX(2,3)*CBGZ(I)+AMTX(3,3)*CGGZ(I)
      FCXFAC(I) =  AMTX(1,1)*CAC(I) +AMTX(2,1)*CBC(I) +AMTX(3,1)*CGC(I)
      FCYFAC(I) =  AMTX(1,2)*CAC(I) +AMTX(2,2)*CBC(I) +AMTX(3,2)*CGC(I)
      FCZFAC(I) =  AMTX(1,3)*CAC(I) +AMTX(2,3)*CBC(I) +AMTX(3,3)*CGC(I)
   31 CONTINUE
      DO 33  I=1,4
      SLPFAC(I) = 0.0
```

```
      IF( ABSUGW(I)  .LE. 0.5) GO TO 33
      SLPFAC(I) = HI(I) / UGW(I)
   33 CONTINUE
      IF(IRPS) GO TO 35
      IRPS = .TRUE.
      DO 34 I=1,4
      RPSI(I) = 0.0
      IF (SLPFAC(I) .EQ. 0.0) GO TO 34
      RPSI(I) = 1.0/SLPFAC(I)
   34 CONTINUE
C          GET DRPSI USING RPSI COMPUTED ABOVE.
       NTRA = 1
      CALL DAUXR(NTRA)
   35 DTSTEP = DT *STEPD
      DTR = DTSTEP
      DTTEST = 1.0E20
      DO 38 I=1,4
      IF (DRPSI(I) .EQ.0.0 .OR. RPSI(I) .EQ.0.0) GO TO 38
   36 DTINT = ABS (RPSI(I)/DRPSI(I))
      IF(DTINT - DTTEST) 37,38,38
   37 DTTEST = DTINT
   38 CONTINUE
C           DTTEST MUST NOT BE ZERO.
      IDTCNT = 1
C
C
   50 IF (DTR  - COMEN4*DTTEST) 52,52,51
   51 DTR = 0.5 * DTR
      IDTCNT = 2*IDTCNT
      IF (IDTCNT .GE.32) GO TO 52
      GO TO 50
   52 IF(DTR.GT.0.0) GO TO 55
      ISTOP = 1
      GO TO 64
   55  NTRA = 2
      CALL DAUXR(NTRA)
      DO 58 I=1,4
      RPSI(I) = RPSI(I) + DRPSI(I)*DTR
C
C  COMPUTE CHANGE IN BRAKE TEMPERATURE AND ADD TO PREVIOUS VALUE FOR
C  BRAKE TEMPERATURE
C            NOTE, TQB(I)*RPSI(I) IS NECESSARILY NEGATIVE,THE SIGNS HAV
C           BEEN SET OPPOSITE. SEE SUBROUTINE ADJTQB,CALLED BY DAUXR.
C
       JFR = 1
       IF(I.GT.2) JFR = 2
      UG36 = 0.0
      ABSUG = ABS(UG(I))
      IF(ABSUG.GT.0.0) UG36 = ABSUG**(0.36)
      DELTAE =(-(TQB(I)*RPSI(I))/777.8  -GN(14,JFR)*UG36 *
     1          ( TAU(I) - TAUA)) * DTR
```

```
      TAU(I)    = TAU(I)    + (DELTAE/(GN(15,JFR)*GN(16,JFR)))
   58 CONTINUE
      IDTCNT = IDTCNT - 1
C     IF(ISTOP) 64,59,64
   59 DO 70 I=1,4
      SFCDTR(I) = SFCDTR(I) +  FC(I) *DTR
      SFSDTR(I) = SFSDTR(I) +  FS(I) *DTR
      SFRCPR(I) = SFRCPR(I) + FRCP(I) * DTR
      SSLIP(I) = SSLIP(I) + SLIP(I) * DTR
      SRHOS(I) = SRHOS(I) + RHOS(I) * DTR
C     RPSSM AND FCSLSM FOR PRINT ONLY.
      RPSSM(I) = RPSSM(I) - (TOB(I) * RPSI(I)) * DTR
      IF(SLPFAC(I).EQ.0.0) GO TO 70
       XTEM = ABS(FC(I)*HI(I)/(12.0*SLPFAC(I)))
      FCSLSM(I) = FCSLSM(I) + (SLIP(I) * XTEM) * DTR
   70 CONTINUE
      IF(IDTCNT) 62,62,55
C
   62 DO 75 I = 1,4
      FCAV  (I) = SFCDTR(I) / DTSTEP
      FC(I) = FCAV  (I)
      RHOSAV(I) = SRHOS(I)/DTSTEP
      FSAV  (I) = SFSDTR(I) / DTSTEP
      FS(I) = FSAV  (I)
      FRCPAV(I) = SFRCPR(I) / DTSTEP
      FRCP(I) = FRCPAV(I)
C                     SLIPAV NOT YET USED.
      SLIPAV(I) = SSLIP(I) / DTSTEP
      FSXU(I)  = FS(I) * FSXFAC(I)
      FSYU(I)  = FS(I) * FSYFAC(I)
      FSZU(I)  = FS(I) * FSZFAC(I)
      FRXU(I)  =-FRCP(I) * FRXFAC(I)
      FRYU(I)  =-FRCP(I) * FRYFAC(I)
      FRZU(I)  =-FRCP(I) * FRZFAC(I)
      FCXU(I)  = FC(I) * FCXFAC(I)
      FCYU(I)  = FC(I) * FCYFAC(I)
      FCZU(I)  = FC(I) * FCZFAC(I)
      FXU(I) = FRXU(I) + FCXU(I) +FSXU(I)
      SFXU   = SFXU + FXU(I)
      FYU(I) = FRYU(I) + FCYU(I) +FSYU(I)
      SFYU   = SFYU + FYU(I)
      FZU(I) = FRZU(I) + FCZU(I) +FSZU(I)
      SFZU   = SFZU + FZU(I)
   75 CONTINUE
C
   64 RETURN
C       ISTOP.NE.0 CAUSES PRINTING OF OUTPUT UP TO CURRENT RUNGE-KUTTA
C         INTERVAL, MESSAGE, AND TERMINATION OF THIS RUN AT END OF THIS
C         INTERVAL IN THE MAIN PROGRAM.
C
C
```

```
C  63 IF (ISTOP) 64,65,64                                                 TIRF3050
C              AT 64 TEMPORARY ERROR STOP.                                TIRF3060
C  64 CALL OUTPUT(2)                                                      TIRF3070
C     CALL PLOTTP(3)                                                      TIRF3080
C     CALL ABDUMP                                                         TIRF3090
C     SUBR ABDUMP CAUSES 'ABNORMAL END' AND DUMP ON OUR OPERATING SYSTEMTIRF3100
C  65 RETURN                                                              TIRF3110
      END                                                                 TIRF3120
```

```
      SUBROUTINE TMCNST                                                 TMCN0010
C              HVOSM-VD2 VERSION                                        TMCN0020
C              REVISED OCTOBER 1975   CALSPAN CORPORATION               TMCN0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    TMCN0040
     1           A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,      TMCN0050
     2           PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                      TMCN0060
     3           XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,TMCN0070
     4           RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,  TMCN0080
     5           T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,     TMCN0090
     6           HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,   TMCN0100
     7           DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,   TMCN0110
     8           NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),    TMCN0120
     9           NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)                  TMCN0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),TMCN0140
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN TMCN0150
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             TMCN0160
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))TMCN0170
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),TMCN0180
     2           (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),        TMCN0190
     3           (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),       TMCN0200
     4           (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),         TMCN0210
     5           (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),           TMCN0220
     6           (PSIFID,VAR(22))                                       TMCN0230
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      TMCN0240
     1           (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))TMCN0250
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),       TMCN0260
     3           (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),     TMCN0270
     4           (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),    TMCN0280
     5           (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),          TMCN0290
     6           (DPSIFI,DER(21)),(DDPSFI,DER(22))                      TMCN0300
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         TMCN0310
     1           (DER(10),DPHIFD)                                       TMCN0320
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       TMCN0330
     1           (DER(14),DDEL4D)                                       TMCN0340
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,TMCN0350
     1           GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          TMCN0360
     2            TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,  TMCN0370
     3            BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,    TMCN0380
     4            XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,    TMCN0390
     5            ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPTMCN0400
     6           ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     TMCN0410
     7            SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,    TMCN0420
     8            SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,  TMCN0430
     9            ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ   TMCN0440
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,  TMCN0450
     1            ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,  TMCN0460
     2            TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,TMCN0470
     3            HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2TMCN0480
     4           ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,TMCN0490
```

```
     5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL             TMCN0500
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                     TMCN0510
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)           TMCN0520
C      COMMON/EINDEX/ FOR EULER ANGLE INDEXING,MAIN,CNSTNT,DAUX,TMCNST      TMCN0530
      COMMON/EINDEX/ TWOPI,PIO2,PIO4,XINDN,XINDL,THETTL,PHITL,PSITL,        TMCN0540
     1               COSTHN,SINTHN,COSPSN,SINPSN,COSPHN,SINPHN,CTHETP,      TMCN0550
     2               STHETP,CPSTP,SPSTP,BNMTX(3,3), CNMTX(3,3),ENDEIN       TMCN0560
      COMMON /COMP4/FIDAR(2),FIDIW(2),FIDWR2(2),SPHICI(4),CPHICI(4),        TMCN0570
     1              TIHI(4),ARBRI(4), PSITEM(4),SLPFAC(4),DTSTEP,DTTEST,TMCN0580
     2       DTINT,TWOPIR,FRTEST(4),XMUI(4),FRCPMU(4),HRTERM,SLIP(4),       TMCN0590
     3              SLIPT(4),RHOS(4),EPSS(4),TERMP(4),TERMB(4),TERM(4),     TMCN0600
     4              EPSSFC,FSXFAC(4),FSYFAC(4),FSZFAC(4),FRXFAC(4),         TMCN0610
     5              FRYFAC(4),FRZFAC(4),FCXFAC(4),FCYFAC(4),FCZFAC(4),      TMCN0620
     6              SFCDTR(4),SFSDTR(4),SFRCPR(4),SSLIP(4),FCAV(4),         TMCN0630
     7              FSAV(4),FRCPAV(4),SLIPAV(4),RPSSM(4),FCSLSM(4),         TMCN0640
     8          ARTQ6(4),TQFAC(4),ARFAC1(2),ARFAC2(2),RPSFA(2),RPSFB(2),TMCN0650
     9          RPSFC(2),RPSFD(2),HRPSFA(4),HRPSFB(4),HRPSFC(4),STEPD       TMCN0660
      COMMON /COMP4/  XBRAK(16),IUVS(4),IUVB(4),IRPS,IDTCNT,ISTEP,ISTOP     TMCN0670
      LOGICAL  IUVS,IUVB,IRPS                                               TMCN0680
      COMMON/INPT5/ IBTYP(2),AK1,AK2,PONE,PTWO,PZERO(2),GN(16,2),ZETAB,     TMCN0690
     1              CONE,CTWO,CTHREE,TAUA,TAUO(4),TLF(51),TTAU(51),         TMCN0700
     2              TRPME(12),TWOT(12),TCT(12),TT(101),TPC(101),TTR(101)TMCN0710
     3             ,TTS(101),BTLF,ETLF,DTLF,NTLF,BRPM,ERPM,NRPM,            TMCN0720
     4              BTT,ETT,DTT,NTT1,NTT2,NTT3,NTTS,XINPT5(9)               TMCN0730
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,           TMCN0740
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),TMCN0750
     2               NCAMF,NCAMR,NDTHF,NDTHR                                TMCN0760
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,          TMCN0770
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,         TMCN0780
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,        TMCN0790
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,        TMCN0800
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,            TMCN0810
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF                         TMCN0820
      DIMENSION ANAME(3)                                                    TMCN0830
      DATA ANAME(1)/4HPSIT/,ANAME(2)/4HTHET/,ANAME(3)/4HPHIT/               TMCN0840
C       * * * * FOR TEMPORARY ERROR STOP, USE THE VARIABLE ASTOP AS SHOWTMCN0850
C     ASTOP IS SOME LARGE NUMBER TO BE COMPARED TO THE ANGLES IN RADIANSTMCN0860
      DATA ASTOP/3000./                                                     TMCN0870
      IF(PHITP.GE.ASTOP .OR. THETTP.GE.ASTOP) GO TO 60                      TMCN0880
C         * * * * ** * * *                                                  TMCN0890
C          THETTL,PHITL,PSITL ARE VALUES OF THETT,PHIT,PSIT FROM PREVIOUTMCN0900
C           TIME INTERVAL, USED TO TEST NEW ANGLES     IN SUBROUTINE TMCTMCN0910
C      XINDL IS PREVIOUS VALUE OF XINDN. XINDL INITIALLY ZERO GETS BNMTXTMCN0920
C            XINDN.NE.0.0 FOR THETAO OR PHIO .NE.0.0, OR AFTER INDEXING     TMCN0930
C            THAT IS  THETN OR PHIN NOW .NE. 0.0                            TMCN0940
C            USED IN MAIN PROGRAM AND IN SUBROUTINES CNSTNT,TMCNST          TMCN0950
      UQ = U*Q                                                              TMCN0960
      WP = W*P                                                              TMCN0970
      UR = U*R                                                              TMCN0980
      QR = Q*R                                                              TMCN0990
      VP = V*P                                                              TMCN1000
```

```
      PR = P*R
      P2 = P*P
      Q2 = Q*Q
      R2 = R*R
      VR = V*R
      WQ = W*Q
      PQ = P*Q
      ZFD1 = ZF+DEL1
      ZRD3 = ZR+DEL3
      IF(ISUS.NE.1) GO TO 100
      D3PD4 = DEL3+DEL4
      D3MD4 = DEL3-DEL4
      D43 = -D3MD4
      DD3P4 = DEL3D+DEL4D
      DD3M4 = DEL3D-DEL4D
      ZRD34 = ZR+0.5*D3PD4
      ZRD4 = ZR+DEL4
      GO TO 200
  100 IF(ISUS.NE.2) GO TO 200
      PHIF2 = PHIF*PHIF
      PHIFD2 = PHIFD*PHIFD
      ZFD1RF = ZFD1+RHOF
      RPF2M = RHF2MF*PHIF2
      RFPF = RHOF*PHIF
      WFMF = XMUF*(DEL1D-RFPF*PHIFD)
      PHFP = PHIF-PHITP
      RPHFD = R*PHIFD
      TPF = 0.5*TF*PHIF
      GO TO 300
  200 IF(ISUS.EQ.2) GO TO 300
      ZFD2 = ZF+DEL2
      D1PD2 = DEL1+DEL2
      D1MD2 = DEL1-DEL2
      DD1P2 = DEL1D+DEL2D
      DD1M2 = DEL1D-DEL2D
      D21 = -D1MD2
      ZFD12 = ZF+0.5*D1PD2
  300 IF(ISUS.EQ.1) GO TO 400
      PHIR2 = PHIR*PHIR
      PHIRD2 = PHIRD*PHIRD
      ZRD3R = ZRD3+RHO
      ZFD3R = ZF+DEL3+RHO
      RPR = RHO*PHIR
      TIZ2 = RHMR2*PHIR2
      TG61 = XMUR*(DEL3D-RPR*PHIRD)
      PHRP = PHIR-PHITP
      TPR = 0.5*TR*PHIR
      RPHRD = R*PHIRD
  400 CONTINUE
    2 SPHTP = SIN(PHITP)
      CPHTP = COS(PHITP)
```

```
      TANTP = TAN(THETTP)                                               TMCN1520
      CTHETP = COS(THETTP)                                              TMCN1530
      SECTP = 1.0/CTHETP                                                TMCN1540
      IF(XINDN)    7, 5, 7                                              TMCN1550
    5 THETT  = THETTP                                                   TMCN1560
      PHIT   = PHITP                                                    TMCN1570
      PSIT   = PSITP + PSIN                                             TMCN1580
      SINPS = SIN(PSIT)                                                 TMCN1590
      COSPS = COS(PSIT)                                                 TMCN1600
      GO TO 70                                                          TMCN1610
    7 IF(XINDN - XINDL)  9,11,9                                         TMCN1620
C          COMPUTE BNMTX  ONCE AFTER EACH INDEXING ON THETMX            TMCN1630
    9 XINDL = XINDN                                                     TMCN1640
C          IF THETAO OR PHIO .NE.0.0 COMPUTE BNMTX  ONCE AT T=TO        TMCN1650
      COSTHN = COS(THETN)                                               TMCN1660
      SINTHN = SIN(THETN)                                               TMCN1670
      COSPHN = COS(PHIN)                                                TMCN1680
      SINPHN = SIN(PHIN)                                                TMCN1690
      COSPSN = COS(PSIN)                                                TMCN1700
      SINPSN = SIN(PSIN)                                                TMCN1710
      BNMTX (1,1)= COSTHN *COSPSN                                       TMCN1720
      BNMTX (2,1) = COSTHN*SINPSN                                       TMCN1730
      BNMTX (3,1) = -SINTHN                                             TMCN1740
      BNMTX (1,2) = -COSPHN*SINPSN + SINPHN*SINTHN*COSPSN               TMCN1750
      BNMTX (2,2) =  COSPHN*COSPSN + SINPHN*SINTHN*SINPSN               TMCN1760
      BNMTX (3,2) =  COSTHN*SINPHN                                      TMCN1770
      BNMTX (1,3) =  SINPHN*SINPSN + COSPHN*SINTHN*COSPSN               TMCN1780
      BNMTX (2,3) = -COSPSN*SINPHN + COSPHN*SINTHN*SINPSN               TMCN1790
      BNMTX (3,3) =  COSTHN*COSPHN                                      TMCN1800
   11 STHETP = SIN(THETTP)                                              TMCN1810
      SPSTP  = SIN(PSITP)                                               TMCN1820
      CPSTP  = COS(PSITP)                                               TMCN1830
      CNMTX (1,1) = CTHETP*CPSTP                                        TMCN1840
      CNMTX (2,1) = CTHETP*SPSTP                                        TMCN1850
      CNMTX (3,1) = -STHETP                                             TMCN1860
      TMP1    =   SPHTP * STHETP                                        TMCN1870
      TMP2    =   CPHTP * STHETP                                        TMCN1880
      CNMTX (1,2) =  -CPHTP*SPSTP  + TMP1*CPSTP                         TMCN1890
      CNMTX (2,2) =   CPHTP*CPSTP  + TMP1*SPSTP                         TMCN1900
      CNMTX (3,2) =   CTHETP*SPHTP                                      TMCN1910
      CNMTX (1,3) =  SPHTP*SPSTP   + TMP2*CPSTP                         TMCN1920
      CNMTX (2,3) = -CPSTP*SPHTP   + TMP2*SPSTP                         TMCN1930
      CNMTX (3,3) =  CTHETP*CPHTP                                       TMCN1940
C         COMPUTE CNMTX EACH R-K STEP IF XINDN.NE.0.0                   TMCN1950
C     ITRY, INDICATOR TO ALLOW ONE ADDITIONAL REVOLUTION FOR TRIAL ANGLETMCN1960
      ITRY = 0                                                          TMCN1970
C         IANG = 1  FOR PSIT, =2 FOR THETT,  =3 FOR PHIT  DETERMINATION TMCN1980
      IANG = 1                                                          TMCN1990
      ANGL = PSITL                                                      TMCN2000
      TMP3 =  BNMTX(2,1)*CNMTX(1,1) + BNMTX(2,2)*CNMTX(2,1)  +          TMCN2010
     X        BNMTX(2,3)*CNMTX(3,1)                                     TMCN2020
```

```
      TMP4 =         BNMTX(1,1)*CNMTX(1,1) + BNMTX(1,2)*CNMTX(2,1) +      TMCN2030
     X               BNMTX(1,3)*CNMTX(3,1)                                TMCN2040
C     NOTE, TANA AND ANGA=ATAN(TANA) NOT USED WHEN DENOMINATOR TANA ZEROTMCN2050
      IF(TMP4)  18,14,18                                                  TMCN2060
   14 IF(TMP3) 15,16,17                                                   TMCN2070
   15 ANGA = - PIO2                                                       TMCN2080
      GO TO 21                                                            TMCN2090
   16 ISTOP = 4                                                           TMCN2100
      GO TO 64                                                            TMCN2110
C                                                                         TMCN2120
   17 ANGA =   PIO2                                                       TMCN2130
      GO TO 21                                                            TMCN2140
   18 TANA = TMP3/TMP4                                                    TMCN2150
C                                                                         TMCN2160
   20 ANGA = ATAN(TANA)                                                   TMCN2170
   21 NREV = ANGL/TWOPI  + SIGN(0.1 ,ANGL)                                TMCN2180
      FNREV = FLOAT(NREV)  * TWOPI                                        TMCN2190
   22 ANGTRY =  ANGA + FNREV                                              TMCN2200
      DIFFA  =  ANGTRY                                                    TMCN2210
      DIFFL =  DIFFA - ANGL                                               TMCN2220
      IF(ABS(DIFFL) - PIO4) 40,40,25                                      TMCN2230
   25 DIFFA = ANGTRY + PI                                                 TMCN2240
      DIFFL = DIFFA - ANGL                                                TMCN2250
      IF(ABS(DIFFL) - PIO4) 40,40,27                                      TMCN2260
   27 DIFFA = ANGTRY - PI                                                 TMCN2270
      DIFFL = DIFFA - ANGL                                                TMCN2280
      IF(ABS(DIFFL) - PIO4) 40,40,29                                      TMCN2290
   29 IF(ANGTRY) 30,30,31                                                 TMCN2300
   30 TWOPIA = TWOPI                                                      TMCN2310
      GO TO 32                                                            TMCN2320
   31 TWOPIA = - TWOPI                                                    TMCN2330
   32 DIFFA  =  ANGTRY + TWOPIA                                           TMCN2340
      DIFFL = DIFFA - ANGL                                                TMCN2350
      IF(ABS(DIFFL) - PIO4) 40,40,33                                      TMCN2360
   33 IF (ITRY) 36,34,36                                                  TMCN2370
   34 FNREV = FNREV + SIGN(TWOPI,ANGL)                                    TMCN2380
      ITRY = 1                                                            TMCN2390
C        ONCE ONLY, INCREASE FNREV BY ONE REVOLUTION AND TRY AGAIN        TMCN2400
      GO TO 22                                                            TMCN2410
   36 ISTOP = 5                                                           TMCN2420
      WRITE(6,1005) T,ANAME(IANG),ANGL, DIFFA ,ANGA,ANGTRY                TMCN2430
 1005 FORMAT( 7H0 TIME=,F8.3,5X,A4,11H  PREVIOUS=,1PE13.5,6H, NEW=,E13.5TMCN2440
     X,12H, AS ARCTAN=, E13.5, 16H,  CORR.FOR REV=,E13.5 ,8H   STOP5)     TMCN2450
      GO TO 64                                                            TMCN2460
C                                                                         TMCN2470
   40 ITRY = 0                                                            TMCN2480
      IF(IANG-2) 41,50,59                                                 TMCN2490
   41 IANG = 2                                                            TMCN2500
      PSIT = DIFFA                                                        TMCN2510
      SINPS  = SIN(PSIT)                                                  TMCN2520
      COSPS = COS(PSIT)                                                   TMCN2530
```

```
      ANGL = THETTL
      TMP5  =        -(BNMTX(3,1)*CNMTX(1,1) + BNMTX(3,2)*CNMTX(2,1) +
     X                 BNMTX(3,3)*CNMTX(3,1) )
      IF( ABS(SINPS) - 0.7) 42,42,43
   42 TMPP4 = TMP4/COSPS
      IF (TMPP4) 49,44,49
   43 TMPP4 = TMP3/SINPS
      IF (TMPP4) 49,44,49
   44 IF( TMP5) 45,46,47
   45 ANGA = - PIO2
      GO TO 21
   46 ISTOP = 6
      GO TO 64
   47 ANGA = PIO2
      GO TO 21
   49 TANA = TMP5/TMPP4
      GO TO 20
   50 IANG = 3
      THETT  = DIFFA
      ANGL  = PHITL
      TMP6  =  BNMTX(3,1)*CNMTX(1,2) +  BNMTX(3,2)*CNMTX(2,2)  +
     X         BNMTX(3,3)*CNMTX(3,2)
      TMP7  =  BNMTX(3,1)*CNMTX(1,3) +  BNMTX(3,2)*CNMTX(2,3)  +
     X         BNMTX(3,3)*CNMTX(3,3)
      IF(TMP7)  55,51,55
   51 IF(TMP6)  52,53,54
   52 ANGA = - PIO2
      GO TO 21
   53 ISTOP = 7
      GO TO 64
   54 ANGA =   PIO2
      GO TO 21
   55 TANA =  TMP6/TMP7
      GO TO 20
   59 PHIT = DIFFA
C       AT ST 70 HAVE NEW PSIT,THETT,PHIT
   70 CONTINUE
C        * * * * * * * * * *
      IF(THETT.GE.ASTOP .OR. PSIT  .GE.ASTOP) GO TO 60
      IF(PHIT .GE.ASTOP) GO TO 60
C  70 COSTH = COS(THETT)
C        * * * * * * * * * *
      COSTH = COS(THETT)
      SINTH = SIN(THETT)
C       COSPS,SINPS COMPUTED ABOVE EITHER AFTER ST 5 OR AFTER ST 41
      COSPH = COS(PHIT)
      SINPH = SIN(PHIT)
    3 CONTINUE
      GCTH = G*COSTH
      GSTH = G*SINTH
      SFXS = 0.0
```

```
      ABSU = ABS(U)                                                      TMCN3050
      IF(ABSU.GE.1.0)SFXS=-CONE*U*ABSU-CTWO*U-CTHREE*SIGN(1.,U)          TMCN3060
      RETURN                                                             TMCN3070
C       * * * * * * * * *                                                TMCN3080
C      ISTOP.NE.0 CAUSES PRINTING OF OUTPUT UP TO CURRENT RUNGE-KUTTA    TMCN3090
C        INTERVAL, MESSAGE, AND TERMINATION OF THIS RUN AT END OF THIS   TMCN3100
C        INTERVAL IN THE MAIN PROGRAM.                                   TMCN3110
C                                                                        TMCN3120
C                                                                        TMCN3130
   60 ISTOP = 30                                                         TMCN3140
C               AT 64 TEMPORARY ERROR STOP.                              TMCN3150
   64 CALL OUTPUT(2)                                                     TMCN3160
      CALL PLOTTP(3)                                                     TMCN3170
      WRITE(6,1006) T, ISTOP                                             TMCN3180
 1006 FORMAT(7H0 TIME=,F8.3,5X, 7H ISTOP=,I3,21H IN SUBROUTINE TMCNST)   TMCN3190
C     CALL ABDUMP                                                        TMCN3200
C     SUBR ABDUMP CAUSES 'ABNORMAL END' AND DUMP ON OUR OPERATING SYSTEMTMCN3210
      STOP                                                               TMCN3220
C           IF STOP IS CODED AS HERE, DOES NOT RETURN TO MAIN PROGRAM.   TMCN3230
C        * * * * * * * * *                                               TMCN3240
C  64 WRITE(6,1006) T, ISTOP                                             TMCN3250
C     RETURN                                                             TMCN3260
      END                                                                TMCN3270
```

```
      SUBROUTINE TREAD(NCARD,NCRDS,NT,NDIM,ARRAY,NERR)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION ARRAY(2),DUM(9)
      IF(NT.GT.NDIM) GO TO 90
      K = 0
      LSEQ = 0
      DO10 I=1,NCRDS
      READ(2,2000) (DUM(N),N=1,9),NSEQ,LCARD
 2000 FORMAT(9F8.0,2I4)
      IF(NCARD.NE.LCARD) GO TO 90
      IF(NSEQ.LE.LSEQ) GO TO 90
      LSEQ = NSEQ
      DO 20 N=1,9
      K = K+1
      ARRAY(K) = DUM(N)
      IF(K.GE.NT) GO TO 91
   20 CONTINUE
   10 CONTINUE
   91 RETURN
   90 NERR = 1
      RETURN
      END
```

```
      SUBROUTINE T2READ(NCARD,ND1,NI,NJ,ARRAY,NERR)
C              HVOSM-RD2 VERSION
C              HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION ARRAY(ND1,NJ),DUM(9)
      LSEQ = 0
      NICRDS = (NI-1)/9 + 1
      DO 30 J=1,NJ
      K = 0
      DO 20 I=1,NICRDS
      READ(2,2000) (DUM(N),N=1,9),NSEQ,LCARD
 2000 FORMAT(9F8.0,2I4)
      IF(NCARD.NE.LCARD) GO TO 90
      IF(NSEQ.LE.LSEQ) GO TO 90
      LSEQ = NSEQ
      DO 10 N=1,9
      K = K+1
      ARRAY(K,J) = DUM(N)
      IF(K.GE.NI) GO TO 30
   10 CONTINUE
   20 CONTINUE
   30 CONTINUE
      RETURN
   90 NERR = 1
      RETURN
      END
```

```
      SUBROUTINE UMOMNT(IS)
C              HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975   CALSPAN CORPORATION
C   SUBROUTINE TO COMPUTE THE MOMENTS ACTING ON THE SPRUNG AND
C   UNSPRUNG MASSES RESULTING FROM TIRE FORCES AND SUSPENSIN FORCES.
C
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,
     1          A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,
     2          PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3          XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4          RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO,
     5          T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6          HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7          DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8          NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9          NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1        XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1           PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2           CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3           STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4           XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5           YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6           CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7           CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8           SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9           FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1           BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2           FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3           F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1          (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1          GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2           TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3           BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4           XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5           ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6          ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7           SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8           SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9           ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1           ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2           TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3           HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2
```

```
    4              ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,UMOM0500
    5               XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL           UMOM0510
     DIMENSION HCAH(4),HCBH(4),HCGH(4)                                    UMOM0520
     EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)          UMOM0530
     COMMON/COMP5/ TAU(4),TQD(2),TQB(4),PP(2),TLAMB(2),PC,RWDRIV,JDEND,UMOM0540
    1               NBTYP,ARFAC3(2),RPSFE(2),RHOSMX(3),SLIPMX(3),SLIPP, UMOM0550
    2               RHOMAX,RHOSAV(4),SRHOS(4),UGW(4),ABSUGW(4),VECS,     UMOM0560
    3               DELTAE,PIO15R,COMEN5,TQE,RPME                        UMOM0570
     COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,          UMOM0580
    1                AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),UMOM0590
    2                NCAMF,NCAMR,NDTHF,NDTHR                              UMOM0600
     COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHCF2,RHFMUF,         UMOM0610
    1                 RHF2MF,RF2MF1,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,      UMOM0620
    2                 DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,     UMOM0630
    3                 PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,     UMOM0640
    4                 PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,         UMOM0650
    5                 DTDD2,DTDD3,DTDD4,FJF(4),SNPF                      UMOM0660
C                                                                         UMOM0670
     IS1 = IS+1                                                           UMOM0680
     GO TO (10,20,30),IS1                                                 UMOM0690
C                                                                         UMOM0700
C   MOMENTS FOR SUSPENSION OPTION 0 , INDEPENDENT FRONT, SOLID AXLE REAR UMOM0710
C                                                                         UMOM0720
  10 TERM1 = ZFD1+HCGH1                                                   UMOM0730
     TERM2 = ZFD2+HCGH2                                                   UMOM0740
C                                                                         UMOM0750
C   ROLL MOMENT                                                           UMOM0760
C                                                                         UMOM0770
     SNPU = -FYU(1)*TERM1 - FYU(2)*TERM2 - (FYU(3)+FYU(4))*ZRD3           UMOM0780
    1       +SI(2)*(TFO2+DTHF2) - SI(1)*(TFO2+DTHF1)                      UMOM0790
    2       +(SI(4)-SI(3))*TSO2 + 12.0*RWDRIV*TQD(2)                      UMOM0800
C                                                                         UMOM0810
C   PITCH MOMENT                                                          UMOM0820
C                                                                         UMOM0830
     SNTU =  (SI(1)+SI(2))*A - (SI(3)+SI(4))*B                            UMOM0840
    1       +FXU(1)*TERM1 + FXU(2)*TERM2                                  UMOM0850
    2       +FXU(3)*(ZRD3R+TPR+HCGH3) + FXU(4)*(ZRD3R-TPR+HCGH4)          UMOM0860
C                                                                         UMOM0870
C   YAW MOMENT                                                            UMOM0880
C                                                                         UMOM0890
     SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)                          UMOM0900
    1       -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)                          UMOM0910
    2       -FXU(1)*(TFO2+DTHF1+HCBH1) + FXU(2)*(TFO2+DTHF2-HCBH2)        UMOM0920
    3       -FXU(3)*(TRO2-RPR+HCBH3) + FXU(4)*(TRO2+RPR-HCBH4)            UMOM0930
C                                                                         UMOM0940
C   REAR AXLE ROLL MOMENT                                                 UMOM0950
C                                                                         UMOM0960
     SNPR =  FZU(3)*(TRO2-RPR+HCBH3) - FZU(4)*(TRO2+RPR-HCBH4)            UMOM0970
    1       -FYU(3)*(RHO+TPR+HCGH3)  - FYU(4)*(RHO-TPR+HCGH4)             UMOM0980
    2       +(SI(3)-SI(4))*TSO2 - 12.0*RWDRIV*TQD(2)                      UMOM0990
     RETURN                                                               UMOM1000
```

```
C
C  MOMENTS FOR SUSPENSION OPTION 1, INDEPENDENT FRONT AND REAR
C
   20 TERM1 = ZFD1+HCGH1
      TERM2 = ZFD2+HCGH2
      TERM3 = ZRD3+HCGH3
      TERM4 = ZRD4+HCGH4
C
C  ROLL MOMENT
C
      SNPU =  SI(2)*(TFO2+DTHF2) - SI(1)*(TFO2+DTHF1)
     1       +SI(4)*(TRO2+DTHR4) - SI(3)*(TRO2+DTHR3)
     2       -FYU(1)*TERM1 - FYU(2)*TERM2 - FYU(3)*TERM3 - FYU(4)*TERM4
C
C  PITCH MOMENT
C
      SNTU = (SI(1)+SI(2))*A - (SI(3)+SI(4))*B
     1       +FXU(1)*TERM1 + FXU(2)*TERM2 + FXU(3)*TERM3 + FXU(4)*TERM4
C
C  YAW MOMENT
C
      SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)
     1       -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)
     2       -FXU(1)*(TFO2+DTHF1+HCBH1) + FXU(2)*(TFO2+DTHF2-HCBH2)
     3       -FXU(3)*(TRO2+DTHR3+HCBH3) + FXU(4)*(TRO2+DTHR4-HCBH4)
      RETURN
C
C  MOMENTS FOR SUSPENSION OPTION 2, SOLID FRONT AND REAR AXLES
C
      FWDRIV = 1.0-RWDRIV
C
C  ROLL MOMENT
C
   30 SNPU = -(FYU(1)+FYU(2))*ZFD1 - (FYU(3)+FYU(4))*ZRD3
     1       +(SI(2)-SI(1))*TSFO2  + (SI(4)-SI(3))*TSO2
     2       +12.0*RWDRIV*TQD(2) + 12.0*FWDRIV*TQD(1)
C
C  PITCH MOMENT
C
      SNTU =  (SI(1)+SI(2))*A - (SI(3)+SI(4))*B
     1       +FXU(1)*(ZFD1RF+TPF+HCGH1) + FXU(2)*(ZFD1RF-TPF+HCGH2)
     2       +FXU(3)*(ZRD3R+TPR+HCGH3)  + FXU(4)*(ZRD3R-TPR+HCGH4)
C
C  YAW MOMENT
C
      SNPSU = FYU(1)*(A+HCAH1) + FYU(2)*(A+HCAH2)
     1       -FYU(3)*(B-HCAH3) - FYU(4)*(B-HCAH4)
     2       -FXU(1)*(TFO2-RFPF+HCBH1) + FXU(2)*(TFO2+RFPF-HCBH2)
     3       -FXU(3)*(TRO2-RPR+HCBH3)  + FXU(4)*(TRO2+RPR-HCBH4)
C
C  FRONT AXLE ROLL MOMENT
```

```
C
      SNPF =  FZU(1)*(TFO2-RFPF+HCBH1) - FZU(2)*(TFO2+RFPF-HCBH2)
     1        -FYU(1)*(RHOF+TPF+HCGH1)  - FYU(2)*(RHOF-TPF+HCGH2)
     2        +(SI(1)-SI(2))*TSFO2 - 12.0*FWDRIV*TQD(1)
C
C  REAR AXLE ROLL MOMENT
C
      SNPR =  FZU(3)*(TRO2-RPR+HCBH3) - FZU(4)*(TRO2+RPR-HCBH4)
     1        -FYU(3)*(RHO+TPR+HCGH3)  - FYU(4)*(RHO-TPR+HCGH4)
     2        +(SI(3)-SI(4))*TSO2 - 12.0*RWDRIV*TQD(2)
      RETURN
      END
```

```
      SUBROUTINE VGORNT                                                 VGOR0010
C              HVOSM-VD2 VERSION                                        VGOR0020
C              REVISED OCTOBER 1975  CALSPAN CORPORATION                VGOR0030
      COMMON/INPT/PHIO,THETAO,PSIO,PO,QO,RO,XCOP,YCOP,ZCOP,UO,VO,WO,    VGOR0040
     1              A,B,DEL10,DEL20,DEL30,PHIRO,DEL10D,DEL20D,DEL30D,   VGOR0050
     2              PHIROD,TF,TR,ZF,ZR,RHO,AKRS,XMUR,                   VGOR0060
     3              XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,VGOR0070
     4              RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,TO, VGOR0080
     5              T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,  VGOR0090
     6              HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB, VGOR0100
     7              DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR, VGOR0110
     8              NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5), VGOR0120
     9              NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)               VGOR0130
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),VGOR0140
     1            XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN VGOR0150
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,          VGOR0160
     1              CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB, VGOR0170
     2              PSIFIO,PSIFDO                                       VGOR0180
      DIMENSION YCIP(2)                                                 VGOR0190
      EQUIVALENCE (YCIP(1),YC1P)                                        VGOR0200
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)                             VGOR0210
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))VGOR0220
     1           ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),VGOR0230
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),       VGOR0240
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),      VGOR0250
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),        VGOR0260
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),          VGOR0270
     6            (PSIFID,VAR(22))                                      VGOR0280
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),      VGOR0290
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))VGOR0300
     2           ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),      VGOR0310
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),    VGOR0320
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),   VGOR0330
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),         VGOR0340
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))                     VGOR0350
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),         VGOR0360
     1            (DER(10),DPHIFD)                                      VGOR0370
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),       VGOR0380
     1            (DER(14),DDEL4D)                                      VGOR0390
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,VGOR0400
     1              PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4), VGOR0410
     2              CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),VGOR0420
     3              STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),   VGOR0430
     4              XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4), VGOR0440
     5              YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4), VGOR0450
     6              CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),   VGOR0460
     7              CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),    VGOR0470
     8              SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),VGOR0480
     9              FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)VGOR0490
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),
     1             BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),
     2             FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),
     3             F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),
     1            (PSII(1),PSI1)
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,
     2             TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB,
     3             BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,
     4             XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,
     5             ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRP
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,
     7             SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,
     8             SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1,
     9             ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,
     1             ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1,
     2             TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,
     3             HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PU,PHIR2
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,
     5             XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL
      DIMENSION HCAH(4),HCBH(4),HCGH(4)
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,
     1               DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,
     2               PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),
     3               SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)
      LOGICAL LCB1,LCB2
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,
     1              XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,
     2              SLOPE1,SLOPE2,XTRA(300)
      DIMENSION UI(4),VI(4),WI(4)
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)
      DIMENSION APITCH(4)
      EQUIVALENCE (APITCH(1),APTCH1)
      COMMON /TIRIN/ AKT(4),SIGT(4),XLAMT(4),AO(4),A1(4),A2(4),A3(4),
     1               A4(4),OMEGT(4),AMU(4),RW(4),FJP(35,4),A234(4),
     2               A12(4),OMT2A2(4),OMT2M1(4),A23(4),ITIR(4)
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,
     1               AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),
     2               NCAMF,NCAMR,NDTHF,NDTHR
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,
     1                RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,
     2                DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,
     3                PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,
     4                PHI3D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,
     5                DTDD2,DTDD3,DTDD4,FJF(4),SNPF
      COMMON/NEWCRB/ YC3P,YC4P,YC5P,YC6P,YCLP,
     1               ZC3P,ZC4P,ZC5P,ZC6P,ZCLP,
```

```
     2              PHIC3,PHIC4,PHIC5,PHIC6,NCRBSL,
     3              TANPC3,TANPC4,TANPC5,TANPC6,TANPCL,
     4              PHIC3R,PHIC4R,PHIC5R,PHIC6R,PHICLR,
     5              YCMP(6),ZCMP(6),PHICM(6)
      COMMON /RUFNES/ DELG,DGMAX,NEND,IRUF
    1 DO 17  I=1,4
      XCPHI = COS(PHII(I))
      XSPHI = SIN(PHII(I))
      XCPSI = COS(PSII(I))
      XSPSI = SIN(PSII(I))
      TMP4 =   XCPHI * XCPSI
      TMP3 =   XSPHI * XCPSI
    2 CAYW(I) =-AMTX(1,1)*XSPSI+ AMTX(1,2)*TMP4 + AMTX(1,3)*TMP3
      CBYW(I) =-AMTX(2,1)*XSPSI+ AMTX(2,2)*TMP4 + AMTX(2,3)*TMP3
      CGYW(I) =-AMTX(3,1)*XSPSI+ AMTX(3,2)*TMP4 + AMTX(3,3)*TMP3
      IF(INDCRB.LE.0) GO TO 3
      LCB1(I) = RW(I).GT.YC1P-YP(I)
      LCB2(I) = RW(I).LE.YP(I)-YCLP
      IF(ICBHIT.EQ.0) GO TO 3
      PHGI(I) = 0.0
      THGI(I) = 0.0
      ZPGI(I) = 0.0
      SPG(I) = 0.0
      CPG(I) = 1.0
      STG(I) = 0.0
      CTG(I) = 1.0
      IF(.NOT.LCB2(I)) GO TO 4
      ZPGI(I) = ZCLP+(YP(I)-YCLP)*TANPCL
      PHGI(I) = PHICLR
      SPG(I) = SIN(PHGI(I))
      CPG(I) = COS(PHGI(I))
      GO TO 30
C         INTRP5 LOOKS UP THGI, PHGI, ZPGI, AND XMUGI FOR EACH WHEEL.
    3 IF(IRUF.EQ.0) GO TO 31
      IF(XP(I)+RW(I).LT.0.0.OR.XP(I)-RW(I).GT.DGMAX) GO TO 31
      CALL RUFFRC(I,ZGP)
      XMUGI(I) = AMU(I)
      GO TO 33
   31 CALL INTRP5(I)
   32 CPG(I) = COS(PHGI(I))
      SPG(I) = SIN(PHGI(I))
      CTG(I) = COS(THGI(I))
      STG(I) = SIN(THGI(I))
   30 CAGZ(I) = CPG(I)*STG(I)
      CBGZ(I) = -SPG(I)
      CGGZ(I) = CTG(I)*CPG(I)
      P1 = CBYW(I)*CGGZ(I)
      P7 = CBGZ(I)*CGYW(I)
      P3 = CGYW(I)*CAGZ(I)
      P4 = CGGZ(I)*CAYW(I)
      P5 = CAYW(I)*CBGZ(I)
```

```
      P6 = CAGZ(I)*CBYW(I)
      D1(I) = P1-P7
      D2(I) = P3-P4
      D3(I) = P5-P6
      CALL GCP(I)
C         XMUGI(I) IS SET IN INTRP5
C         IF ICBHIT.NE.0 AND LCB1(I) AND LCB2(I) BOTH  FALSE,  XMUGI(I)
C         NOT SET IN THIS INTERVAL.  RETAINS LAST VALUE, SHOULD BE FOR
C         FLAT TERRAIN. (RADIAL SPRING TIRE MODE IN CRBIMP REQUIRES
C         FLAT TERRAIN PREVIOUS TO CURB HIT)
      GO TO 5
    4 IF(.NOT.LCB1(I))GO TO 30
      CALL CRBIMP(I)
      XMUGI(I) = AMUC*AMU(I)
   33 CAGZ(I) = CPG(I)*STG(I)
      CBGZ(I) = -SPG(I)
      CGGZ(I) = CTG(I)*CPG(I)
      P1 = CBYW(I)*CGGZ(I)
      P7 = CBGZ(I)*CGYW(I)
      P3 = CGYW(I)*CAGZ(I)
      P4 = CGGZ(I)*CAYW(I)
      P5 = CAYW(I)*CBGZ(I)
      P6 = CAGZ(I)*CBYW(I)
      D1(I) = P1-P7
      D2(I) = P3-P4
      D3(I) = P5-P6
    5 CAH(I) = AMTX(1,1)*CAR(I)+AMTX(2,1)*CBR(I)+AMTX(3,1)*CGR(I)
      CBH(I) = AMTX(1,2)*CAR(I)+AMTX(2,2)*CBR(I)+AMTX(3,2)*CGR(I)
      CGH(I) = AMTX(1,3)*CAR(I)+AMTX(2,3)*CBR(I)+AMTX(3,3)*CGR(I)
      HCAH(I) = HI(I)*CAH(I)
      HCBH(I) = HI(I)*CBH(I)
      HCGH(I) = HI(I)*CGH(I)
   17 CONTINUE
C
      IF(ISUS.NE.0) GO TO 90
      V1 = V+A*R-ZFD1*P-HCGH1*(P+PHI1D)+DTDD1*DEL1D
      V2 = V+A*R-ZFD2*P-HCGH2*(P+PHI2D)-DTDD2*DEL2D
      V3 = V-B*R-ZRD3*P-(RHO+TPR+HCGH3)*(P+PHIRD)
      V4 = V-B*R-ZRD3*P-(RHO-TPR+HCGH4)*(P+PHIRD)
      W1 = W-A*Q+(TFO2+DTHF1)*P+DEL1D+HCBH1*(P+PHI1D)
      W2 = W-A*Q-(TFO2+DTHF2)*P+DEL2D+HCBH2*(P+PHI2D)
      W3 = W+B*Q+DEL3D-(RPR-TRO2-HCBH3)*(P+PHIRD)
      W4 = W+B*Q+DEL3D-(RPR+TRO2-HCBH4)*(P+PHIRD)
      GO TO 95
   90 IF(ISUS.EQ.2) GO TO 91
      V1 = V+A*R-ZFD1*P-HCGH1*(P+PHI1D)+DTDD1*DEL1D
      V2 = V+A*R-ZFD2*P-HCGH2*(P+PHI2D)-DTDD2*DEL2D
      V3 = V-B*R-ZRD3*P-HCGH3*(P+PHI3D)+DTDD3*DEL3D
      V4 = V-B*R-ZRD4*P-HCGH4*(P+PHI4D)-DTDD4*DEL4D
      W1 = W-A*Q+(TFO2+DTHF1)*P+DEL1D+HCBH1*(P+PHI1D)
      W2 = W-A*Q-(TFO2+DTHF2)*P+DEL2D+HCBH2*(P+PHI2D)
```

```
      W3 = W+B*Q+(TRO2+DTHR3)*P+DEL3D+HCBH3*(P+PHI3D)
      W4 = W+B*Q-(TRO2+DTHR4)*P+DEL4D+HCBH4*(P+PHI4D)
      GO TO 95
   91 V1 = V+A*R-ZFD1*P-(RHOF+TPF+HCGH1)*(P+PHIFD)
      V2 = V+A*R-ZFD1*P-(RHOF-TPF+HCGH2)*(P+PHIFD)
      V3 = V-B*R-ZRD3*P-(RHO+TPR+HCGH3)*(P+PHIRD)
      V4 = V-B*R-ZRD3*P-(RHO-TPR+HCGH4)*(P+PHIRD)
      W1 = W-A*Q+DEL1D-(RFPF-TFO2-HCBH1)*(P+PHIFD)
      W2 = W-A*Q+DEL1D-(RFPF+TFO2-HCBH2)*(P+PHIFD)
      W3 = W+B*Q+DEL3D-(RPR-TRO2-HCBH3)*(P+PHIRD)
      W4 = W+B*Q+DEL3D-(RPR+TRO2-HCBH4)*(P+PHIRD)
C
   95 DO 170 I=1,4
   10 AX(I) = CBY*CGGZ(I)-CGY*CBGZ(I)
      BX(I) = CGY*CAGZ(I)-CAY*CGGZ(I)
      CX(I) = CAY*CBGZ(I)-CBY*CAGZ(I)
      DISTX = SQRT(AX(I)**2+BX(I)**2+CX(I)**2)
      CTXG(I) = (CAX*AX(I)+CBX*BX(I)+CGX*CX(I))/DISTX
      CTXG(I) = SIGN(AMIN1(ABS(CTXG(I)),1.0),CTXG(I))
      STXG(I) = SIGN(SQRT(1.0-CTXG(I)**2),CGX*DISTX-CX(I))
      UG(I) = UI(I)*CTXG(I)-WI(I)*STXG(I)
   11 AY(I) = CGX*CBGZ(I)-CBX*CGGZ(I)
      BY(I) = CAX*CGGZ(I)-CGX*CAGZ(I)
      CY(I) = CBX*CAGZ(I)-CAX*CBGZ(I)
      DISTY = SQRT(AY(I)**2+BY(I)**2+CY(I)**2)
   12 CPYG(I) = (CAY*AY(I)+CBY*BY(I)+CGY*CY(I))/DISTY
      CPYG(I) = SIGN(AMIN1(ABS(CPYG(I)),1.0),CPYG(I))
      SPYG(I) = SIGN(SQRT(1.0-CPYG(I)**2),CGY*DISTY-CY(I))
      VG(I) = VI(I)*CPYG(I)-WI(I)*SPYG(I)
      DISTD = SQRT(D1(I)**2+D2(I)**2+D3(I)**2)
   13 CAZW = - AMTX(1,2)*XSPHI + AMTX(1,3)*XCPHI
      CBZW = - AMTX(2,2)*XSPHI + AMTX(2,3)*XCPHI
      CGZW = - AMTX(3,2)*XSPHI + AMTX(3,3)*XCPHI
      PSIIP(I) = PSII(I)*(CAGZ(I)*CAZW+CBGZ(I)*CBZW+CGGZ(I)*CGZW)
   14 CAC(I) = D1(I)/DISTD
      CBC(I) = D2(I)/DISTD
      CGC(I) = D3(I)/DISTD
   15 AS(I) = CGC(I)*CBGZ(I)-CBC(I)*CGGZ(I)
      BS(I) = CAC(I)*CGGZ(I)-CGC(I)*CAGZ(I)
      CS(I) = CBC(I)*CAGZ(I)-CAC(I)*CBGZ(I)
      DISTS = SQRT(AS(I)**2+BS(I)**2+CS(I)**2)
      CAS(I) = AS(I)/DISTS
      CBS(I) = BS(I)/DISTS
      CGS(I) = CS(I)/DISTS
C
C  16 CALL TIRFRC(I)
C
  170 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VPOS
C           HVOSM-VD2 VERSION
C              REVISED OCTOBER 1975  CALSPAN CORPORATION
      COMMON/INPT/PHI0,THETA0,PSI0,P0,Q0,R0,XCOP,YCOP,ZCOP,U0,V0,W0,
     1            A,B,DEL10,DEL20,DEL30,PHIR0,DEL10D,DEL20D,DEL30D,
     2            PHIR0D,TF,TR,ZF,ZR,RHO,AKRS,XMUR,
     3            XMS,XMUF,XIX,XIY,XIZ,XIXZ,CF,AKF,XLAMF,OMEGF,CFP,EPSF,
     4            RF,CR,AKR,XLAMR,OMEGR,CRP,EPSR,RR,TS,THMAX,DTCOMP,T0,
     5            T1,DTCMP1,DTPRNT,MODE,EBAR,EM,AAA,HMAX,HMIN,BET,G,
     6            HED(36),DADE(3),XIR,X1,Y1,Z1,X2,Y2,Z2,PHIC(50),DELB,
     7            DELE,DDEL,NDEL,PSIF(50),TQF(50),TQR(50),TB,TE,TINCR,
     8            NZTAB,NZ5,XBDRY(4,5),PSBDRY(4,5),YBDRY(2,5),NBX(5),
     9            NBY(5),NTBL1,NTBL2,NTBL3,ZGP(21,21,5)
      COMMON/INPT/XB(5),XE(5),XINCR(5),NX(5),YB(5),YE(5),YINCR(5),NY(5),
     1          XXZGP5(21),YYZGP5(21),AMUG(5),PSBDRO(4,5),UVWMIN,PQRMIN
      COMMON/INPT1/YC1P,YC2P,ZC2P,DELTC,PHIC1,PHIC2,AMUC,XIPS,
     1             CPSP,OMGPS,AKPS,EPSPS,XPS,RWHJB,RWHJE,DRWHJ,INDCRB,
     2             PSIFI0,PSIFD0
      DIMENSION YCIP(2)
      EQUIVALENCE (YCIP(1),YC1P)
      COMMON /INTG/NEQ,T,DT,VAR(50),DER(50)
      EQUIVALENCE (U,VAR(1)),(V,VAR(2)),(W,VAR(3)),(P,VAR(4)),(Q,VAR(5))
     1          ,(R,VAR(6)),(DEL1,VAR(7)),(DEL1D,VAR(8)),(DEL2,VAR(9)),
     2            (DEL2D,VAR(10)),(DEL3,VAR(11)),(DEL3D,VAR(12)),
     3            (PHIR,VAR(13)),(PHIRD,VAR(14)),(THETTP,VAR(15)),
     4            (PHITP,VAR(16)),(PSITP,VAR(17)),(XCP,VAR(18)),
     5            (YCP,VAR(19)),(ZCP,VAR(20)),(PSIFI,VAR(21)),
     6            (PSIFID,VAR(22))
      EQUIVALENCE (DU,DER(1)),(DV,DER(2)),(DW,DER(3)),(DP,DER(4)),
     1            (DQ,DER(5)),(DR,DER(6)),(DDEL1,DER(7)),(DDEL1D,DER(8))
     2          ,(DDEL2,DER(9)),(DDEL2D,DER(10)),(DDEL3,DER(11)),
     3            (DDEL3D,DER(12)),(DPHIR,DER(13)),(DPHIRD,DER(14)),
     4            (DTHTTP,DER(15)),(DPHITP,DER(16)),(DPSITP,DER(17)),
     5            (DXCP,DER(18)),(DYCP,DER(19)),(DZCP,DER(20)),
     6            (DPSIFI,DER(21)),(DDPSFI,DER(22))
      EQUIVALENCE (VAR(9),PHIF),(VAR(10),PHIFD),(DER(9),DPHIF),
     1            (DER(10),DPHIFD)
      EQUIVALENCE (VAR(13),DEL4),(VAR(14),DEL4D),(DER(13),DDEL4),
     1            (DER(14),DDEL4D)
      COMMON /DIMV/X1P,X2P,X3P,X4P,Y1P,Y2P,Y3P,Y4P,Z1P,Z2P,Z3P,Z4P,PHI1,
     1             PHI2,PHI3,PHI4,PSI1,PSI2,PSI3,PSI4,CAYW(4),CBYW(4),
     2             CGYW(4),ZPGI(4),THGI(4),PHGI(4),CPG(4),SPG(4),CTG(4),
     3             STG(4),CAGZ(4),CBGZ(4),CGGZ(4),D1(4),D2(4),D3(4),
     4             XLM1(4),XLM2(4),XLM3(4),AMTX(3,3),CMTX(3,4),XGPP(4),
     5             YGPP(4),ZGPP(4),DMATX(10,11),DELTA(4),CAR(4),CBR(4),
     6             CGR(4),FR(4),HI(4),FC(4),TI(4),AX(4),BX(4),CX(4),
     7             CTXG(4),UG(4),STXG(4),AY(4),BY(4),CY(4),CPYG(4),
     8             SPYG(4),VG(4),PSIIP(4),PHICI(4),CAC(4),CBC(4),CGC(4),
     9             FCXU(4),FCYU(4),FCZU(4),FS(4),CAXW(4),CBXW(4),CGXW(4)
```

```
      COMMON /DIMV/AS(4),BS(4),CS(4),CAS(4),CBS(4),CGS(4),BETP(4),       VPOS0500
     1              BETBR(4),FSXU(4),FSYU(4),FSZU(4),FRXU(4),FRYU(4),    VPOS0510
     2              FRZU(4),FXU(4),FYU(4),FZU(4),SI(4),F1FI(2),F1RI(2),  VPOS0520
     3              F2FI(2),F2RI(2),CAH(4),CBH(4),CGH(4)                 VPOS0530
      DIMENSION XP(4),YP(4),ZP(4),PHII(4),PSII(4)                        VPOS0540
      EQUIVALENCE (XP(1),X1P),(YP(1),Y1P),(ZP(1),Z1P),(PHII(1),PHI1),    VPOS0550
     1            (PSII(1),PSI1)                                         VPOS0560
      COMMON /COMP/SUMM,THETN,PHIN,PSIN,PI,RAD,GAM1,GAM2,GAM3,GAM4,GAM5,VPOS0570
     1            GAM6,GAM7,GAM8,GAM9,THETT,PHIT,PSIT,ZRO,TRO2,          VPOS0580
     2              TFO2,TIZ,RHO2,RHOMUR,AMUF,BMUR,ZPR,TM4,RHMR2,AO2APB, VPOS0590
     3              BO2APB,RFTF,TSO2,RRTS,BROMUR,XMUFO2,AXMFO2,XMTFO4,   VPOS0600
     4              XIZR,RTR,RHMR2I,XIXP,XIZP,XIXZP,XIYZP,D1PD2,D1MD2,   VPOS0610
     5              ZRD3,ZRD3R,ZFD3R,ZFD12,TIZ2,TG61,DD1P2,DD1M2,RPR,PHRPVPOS0620
     6            ,TANTP,SPHTP,CPHTP,SECTP,SFXS,SFYS,SFZS,SNPS,SNTS,     VPOS0630
     7              SNPSS,TPR,CAY,CBY,CGY,CAX,CBX,CGX,SFYU,SFXU,SFYUF,   VPOS0640
     8              SFYUR,SFZU,COSTH,SINTH,COSPS,SINPS,COSPH,SINPH,ANG1, VPOS0650
     9              ANG2,CPHI,SPHI,CPSI,SPSI,P1,P7,P3,P4,P5,P6,TX,TY,TZ  VPOS0660
      COMMON /COMP/TRH,DISTX,DISTY,DISTD,DISTS,D21,ZETA4,ZETA4D,ZETA3,   VPOS0670
     1              ZETA3D,SFZ1,SNPU,SNTU,HCGH1,HCGH2,HCGH3,HCGH4,TERM1, VPOS0680
     2              TERM2,SNPSU,SNPR,HCBH1,HCBH2,HCBH3,HCBH4,HCAH1,HCAH2,VPOS0690
     3              HCAH3,HCAH4,UQ,WP,UR,QR,VP,PR,P2,Q2,R2,VR,WQ,PQ,PHIR2VPOS0700
     4            ,PHIRD2,RPHRD,GCTH,GSTH,GCTSP,GCTCP,XXX,YYY,IX,IY,XX1,VPOS0710
     5              XX2,YY1,YY2,THG1,THG2,PHG1,PHG2,ZZ1,ZZ2,LLL          VPOS0720
      DIMENSION HCAH(4),HCBH(4),HCGH(4)                                  VPOS0730
      EQUIVALENCE (HCAH(1),HCAH1),(HCBH(1),HCBH1),(HCGH(1),HCGH1)        VPOS0740
      COMMON /COMPN/ FRSP(4),FRCP(4),ICBHIT,JCBHIT,                      VPOS0750
     1                DPSINT,TANPC1,TANPC2,PHIC1R,PHIC2R,AMUCMP,PHI1D,   VPOS0760
     2                PHI2D,LCB1(4),LCB2(4),IHIT,AJMTX(3,3),BMTX(3,3),   VPOS0770
     3                SFRX(4),SFRY(4),SFRZ(4),T1PSI,T2PSI,XMUGI(4)       VPOS0780
      LOGICAL LCB1,LCB2                                                  VPOS0790
      COMMON/ADTNL/ U1,U2,U3,U4,V1,V2,V3,V4,W1,W2,W3,W4,                 VPOS0800
     1               XIYP,SPHIC,CPHIC,APTCH1,APTCH2,APTCH3,APTCH4,       VPOS0810
     2               SLOPE1,SLOPE2,XTRA(300)                             VPOS0820
      DIMENSION UI(4),VI(4),WI(4)                                        VPOS0830
      EQUIVALENCE (UI(1),U1),(VI(1),V1),(WI(1),W1)                       VPOS0840
      DIMENSION APITCH(4)                                                VPOS0850
      EQUIVALENCE (APITCH(1),APTCH1)                                     VPOS0860
      COMMON /INSUS/ XIF,RHOF,TSF,PHIFO,PHIFOD,DEL40,DEL40D,ISUS,        VPOS0870
     1                AKDS,AKDS1,AKDS2,AKDS3,PHIRC(50),DTHF(50),DTHR(50),VPOS0880
     2                NCAMF,NCAMR,NDTHF,NDTHR                            VPOS0890
      COMMON /SUSCMP/ XMURO2,BXMRO2,XMTRO4,ZFO,TSFO2,RHOF2,RHFMUF,       VPOS0900
     1                 RHF2MF,RF2MFI,RTF,RRTR,D3PD4,D3MD4,D43,DD3P4,     VPOS0910
     2                 DD3M4,ZFD1RF,ZRD34,RFPF,RPF2M,WFMF,PHFP,PHIF2,    VPOS0920
     3                 PHIFD2,RPHFD,ZFD1,ZFD2,ZRD4,TPF,SLOPE3,SLOPE4,    VPOS0930
     4                 PH13D,PHI4D,DTHF1,DTHF2,DTHR3,DTHR4,DTDD1,        VPOS0940
     5                 DTDD2,DTDD3,DTDD4,FJF(4),SNPF                     VPOS0950
C                                                                        VPOS0960
      IS1 = ISUS+1                                                       VPOS0970
C                                                                        VPOS0980
C  LONGITUDINAL WHEEL CENTER VELOCITIES                                  VPOS0990
C                                                                        VPOS1000
```

```
      GO TO (10,11,12),IS1
C
C  SUSPENSION OPTION 0, INDEPENDENT FRONT AND SOLID AXLE REAR
C
   10 IF(NDTHF.EQ.0) GO TO 101
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL1,DTHF1,DTDD1)
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL2,DTHF2,DTDD2)
  101 U1 = U-(TFO2+DTHF1)*R+ZFD1*Q
      U2 = U+(TFO2+DTHF2)*R+ZFD2*Q
      U3 = U-(TRO2-RPR)*R+(ZRD3R+TPR)*Q
      U4 = U+(TRO2+RPR)*R+(ZRD3R-TPR)*Q
      GO TO 13
C
C  SUSPENSION OPTION 1, INDEPENDENT FRONT AND REAR
C
   11 IF(NDTHF.EQ.0) GO TO 111
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL1,DTHF1,DTDD1)
      CALL INTRPC(DTHF,DELB,DELE,DDEL,DEL2,DTHF2,DTDD2)
  111 IF(NDTHR.EQ.0) GO TO 112
      CALL INTRPC(DTHR,DELB,DELE,DDEL,DEL3,DTHR3,DTDD3)
      CALL INTRPC(DTHR,DELB,DELE,DDEL,DEL4,DTHR4,DTDD4)
  112 U1 = U-(TFO2+DTHF1)*R+ZFD1*Q
      U2 = U+(TFO2+DTHF2)*R+ZFD2*Q
      U3 = U-(TRO2+DTHR3)*R + ZRD3*Q
      U4 = U+(TRO2+DTHR4)*R + ZRD4*Q
      GO TO 13
C
C  SUSPENSION OPTION 2, SOLID FRONT AND REAR AXLES
C
   12 U1 = U-(TFO2-RFPF)*R+(ZFD1RF+TPF)*Q
      U2 = U+(TFO2+RFPF)*R+(ZFD1RF-TPF)*Q
      U3 = U-(TRO2-RPR)*R +(ZRD3R+TPR)*Q
      U4 = U+(TRO2+RPR)*R +(ZRD3R-TPR)*Q
   13 CONTINUE
C            FORMERLY, TIRFRC(I) WAS CALLED FROM VGORNT SEPARATELY FOR
C         EACH WHEEL. THE SUM OF THE FORCES FOR ALL WHEELS WAS THEREFOR
C            NOT ZEROED IN TIRFRC BUT IN VPOS FOR EACH RUNGE-KUTTA STEP
C            SFXU,SFYU,SFZU
C         SFYUF AND SFYUR NO LONGER USED
      SFYUF = 0.0
      SFYUR = 0.0
    2 AMTX(1,1) = COSTH*COSPS
      AMTX(2,1) = COSTH*SINPS
      AMTX(3,1) = -SINTH
      AMTX(1,2) = -COSPH*SINPS+SINPH*SINTH*COSPS
      AMTX(2,2) = COSPH*COSPS+SINPH*SINTH*SINPS
      AMTX(3,2) = COSTH*SINPH
      AMTX(1,3) = SINPH*SINPS+COSPH*SINTH*COSPS
      AMTX(2,3) = -COSPS*SINPH+COSPH*SINTH*SINPS
      AMTX(3,3) = COSTH*COSPH
      CAY = AMTX(1,2)
```

```
      CBY = AMTX(2,2)                                                   VPOS1520
      CGY = AMTX(3,2)                                                   VPOS1530
      CAX = AMTX(1,1)                                                   VPOS1540
      CBX = AMTX(2,1)                                                   VPOS1550
      CGX = AMTX(3,1)                                                   VPOS1560
C                                                                       VPOS1570
      IF(ISUS.EQ.2) GO TO 21                                            VPOS1580
      YTMP = TFO2+DTHF1                                                 VPOS1590
      ZTMP = ZFD1                                                       VPOS1600
      GO TO 31                                                          VPOS1610
   21 YTMP = TFO2-RFPF                                                  VPOS1620
      ZTMP = ZFO+DEL1+TPF                                               VPOS1630
   31 X1P = XCP+AMTX(1,1)*A+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP               VPOS1640
      Y1P = YCP+AMTX(2,1)*A+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP               VPOS1650
      Z1P = ZCP+AMTX(3,1)*A+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP               VPOS1660
      IF(ISUS.EQ.2) GO TO 22                                            VPOS1670
      YTMP = -TFO2-DTHF2                                                VPOS1680
      ZTMP = ZFD2                                                       VPOS1690
      GO TO 32                                                          VPOS1700
   22 YTMP = -TFO2-RFPF                                                 VPOS1710
      ZTMP = ZFO+DEL2-TPF                                               VPOS1720
   32 X2P = XCP+AMTX(1,1)*A+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP               VPOS1730
      Y2P = YCP+AMTX(2,1)*A+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP               VPOS1740
      Z2P = ZCP+AMTX(3,1)*A+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP               VPOS1750
      IF(ISUS.EQ.1) GO TO 23                                            VPOS1760
      YTMP = TRO2-RPR                                                   VPOS1770
      ZTMP = ZRO+DEL3+TPR                                               VPOS1780
      GO TO 33                                                          VPOS1790
   23 YTMP = TRO2+DTHR3                                                 VPOS1800
      ZTMP = ZRD3                                                       VPOS1810
   33 X3P = XCP-AMTX(1,1)*B+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP               VPOS1820
      Y3P = YCP-AMTX(2,1)*B+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP               VPOS1830
      Z3P = ZCP-AMTX(3,1)*B+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP               VPOS1840
      IF(ISUS.EQ.1) GO TO 24                                            VPOS1850
      YTMP = -TRO2-RPR                                                  VPOS1860
      ZTMP = ZRO+DEL3-TPR                                               VPOS1870
      GO TO 34                                                          VPOS1880
   24 YTMP = -TRO2-DTHR4                                                VPOS1890
      ZTMP = ZRD4                                                       VPOS1900
   34 X4P = XCP-AMTX(1,1)*B+AMTX(1,2)*YTMP+AMTX(1,3)*ZTMP               VPOS1910
      Y4P = YCP-AMTX(2,1)*B+AMTX(2,2)*YTMP+AMTX(2,3)*ZTMP               VPOS1920
      Z4P = ZCP-AMTX(3,1)*B+AMTX(3,2)*YTMP+AMTX(3,3)*ZTMP               VPOS1930
C                                                                       VPOS1940
C     QUADRATIC INTERPOLATION SUBROUTINE INTRPL,ADDITIONAL ENTRY INTRPC VPOS1950
C                                                                       VPOS1960
C                                                                       VPOS1970
      IF(ISUS.EQ.2) GO TO 50                                            VPOS1980
      CALL INTRPC(PHIC,DELB,DELE,DDEL,DEL1,PHI1,SLOPE1)                 VPOS1990
      PHI1 = PHI1*RAD                                                   VPOS2000
      SLOPE1 = SLOPE1*RAD                                               VPOS2010
      PHI1D = SLOPE1*DEL1D                                              VPOS2020
```

```
      CALL INTRPC(PHIC,DELB,DELE,DDEL,DEL2,PHI2,SLOPE2)                 VPOS2030
      PHI2 = -PHI2*RAD                                                  VPOS2040
      SLOPE2 = -SLOPE2*RAD                                              VPOS2050
      PHI2D = SLOPE2*DEL2D                                              VPOS2060
      GO TO 51                                                          VPOS2070
   50 PHI1 = PHIF                                                       VPOS2080
      PHI2 = PHIF                                                       VPOS2090
      PHI1D = PHIFD                                                     VPOS2100
      PHI2D = PHIFD                                                     VPOS2110
   51 IF(ISUS.EQ.1) GO TO 52                                            VPOS2120
      PHI3 = PHIR                                                       VPOS2130
      PHI4 = PHIR                                                       VPOS2140
      PHI3D = PHIRD                                                     VPOS2150
      PHI4D = PHIRD                                                     VPOS2160
      GO TO 53                                                          VPOS2170
   52 CALL INTRPC(PHIRC,DELB,DELE,DDEL,DEL3,PHI3,SLOPE3)                VPOS2180
      PHI3 = PHI3*RAD                                                   VPOS2190
      SLOPE3 = SLOPE3*RAD                                               VPOS2200
      PHI3D = SLOPE3*DEL3D                                              VPOS2210
      CALL INTRPC(PHIRC,DELB,DELE,DDEL,DEL4,PHI4,SLOPE4)                VPOS2220
      PHI4 = -PHI4*RAD                                                  VPOS2230
      SLOPE4 = -SLOPE4*RAD                                              VPOS2240
      PHI4D = SLOPE4*DEL4D                                              VPOS2250
   53 CONTINUE                                                          VPOS2260
C                                                                       VPOS2270
   40 IF(INDCRB.EQ.0) GO TO 5                                           VPOS2280
      IF(IHIT.EQ.1.OR.INDCRB.LT.0) GO TO 6                              VPOS2290
    5 CALL DRIVER(PSICON,PSISLP,J)                                      VPOS2300
      IF(J.NE.0) GO TO 5001                                             VPOS2310
      PSICON = 0.0                                                      VPOS2320
      PSISLP = 0.0                                                      VPOS2330
      IF(NTBL1.NE.0) CALL INTRPC(PSIF,TB,TE,TINCR,T,PSICON,PSISLP)      VPOS2340
      PSICON = PSICON*RAD                                               VPOS2350
      PSISLP=PSISLP*RAD                                                 VPOS2360
 5001 CONTINUE                                                          VPOS2370
C     FORMERLY PSIFP=PSI1,NO LONGER USED.FORMERLY PSIFID=(PSI1-PSIFP)/DTVPOS2380
      PSI1 = PSICON                                                     VPOS2390
      PSIFID = PSISLP                                                   VPOS2400
      PSIFI = PSI1                                                      VPOS2410
      GO TO 7                                                           VPOS2420
    6 PSI1 = PSIFI                                                      VPOS2430
    7 PSI2 = PSI1                                                       VPOS2440
C                                                                       VPOS2450
      IF(ISUS.EQ.1) GO TO 54                                            VPOS2460
      PSI3 = AKRS*PHIR                                                  VPOS2470
      PSI4 = PSI3                                                       VPOS2480
      RETURN                                                            VPOS2490
   54 PSI3 = AKDS+AKDS1*DEL3+AKDS2*DEL3**2+AKDS3*DEL3**3                VPOS2500
      PSI4 = -(AKDS+AKDS1*DEL4+AKDS2*DEL4**2+AKDS3*DEL4**3)             VPOS2510
      RETURN                                                            VPOS2520
      END                                                               VPOS2530
```

```
      SUBROUTINE WHEEL(/AKT/,/SIGT/,/XLAMT/,/RWHJB/,/RWHJE/,/DRWHJ/,
     1 /NFJP/,/RW/,FJP,/NO/)
C             HVOSM-VD2 VERSION
C             REVISED OCTOBER 1975   CALSPAN CORPORATION
      DIMENSION FJP(50)
    1 DA = 4.0*0.01745
      FJP(1) = 0.0
      N = NFJP
      IF (N.LE.NO) GO TO 3
      PRINT 2,N,NO
2       FORMAT ('ODIM. FOR FJP TOO SMALL,',I6,' NEEDED.', I6,' PROVIDED.
     1 ')
      STOP
3     CONTINUE
      NL = N-1
      DD = (RWHJE-RWHJB)/FLOAT(NL)
      DDK = DD*AKT
      K = 0
      D = 0.0
      DO 10  J=2,N
      FJP(J) = FJP(J-1)+DDK
      D = D+DD
      IF (K.NE.0) GO TO 10
      IF (D.LT.SIGT) GO TO 10
      X = DDK
      DDK = DDK*XLAMT
      FJP(J) = FJP(J)+(DDK-X)*(D-SIGT)/DD
      K = 1
   10 CONTINUE
   15 R = RW
      DO 19  J=2,N
      B = 1.0
      DDK = DD/R
      Z=DDK
  200 ANG = 0.0
      F = Z*B
  201 ANG = ANG+DA
      Y=1-Z
      X = COS(ANG)
      IF(X.LE.Y) GO TO 16
      F = F+2.0*(X-Y)*B
      GO TO 201
   16 B = FJP(J)/F
      FJP(J) = DDK*B
      IF (J.EQ.N) GO TO 1901
      I=J+1
      DO 18  L=I,N
      Z=Z+DDK
  300 ANG = 0.0
```

```
      F = Z*B
  301 ANG = ANG+DA
      Y=1-Z
      X = COS(ANG)
      IF(X.LE.Y) GO TO 18
      F = F+2.0*(X-Y)*B
      GO TO 301
   18 FJP(L) = FJP(L)-F
   19 R = R-DD
 1901 DD =0.0
      DO 20 J=2,N
      DD=DD+FJP(J)
20    FJP(J)=FJP(J-1)+DD
      RETURN
      END
```

4. SYSTEM REQUIREMENTS

The HVOSM is executed at Calspan Corporation on an IBM/370 Model 165 under the System 370 Operating System. The source programs are compiled with the IBM Fortran H compiler. Input requirements consist of a card reader (Fortran unit 5), a sequential data set (FORTRAN unit 2) for temporary storage of the card input, and a sequential data set containing road roughness data (FORTRAN unit 4). Output requirements consist of two sequential data sets (FORTRAN units 1 and 3) which store certain data for subsequent processing. Unit 1 is used by the HVOSM Vehicle Graphics Program. Unit 3 is currently not used but is intended to store data for subsequent time history plotting. In addition, FORTRAN units 6 and 11 through 30 are used for printed output.

The JCL required to execute the HVOSM using the loader from object and/or load modules is illustrated in Figure 4.1-1, where DSHVOSV2 is the HVOSM load module name.

The Calspan procedure LOADGO is shown in Figure 4.1-2. The HVOSM program version to be executed is stored as module name as a member of the partitioned data set LOADLIB, and core storage size (GCORE) varies as a function of program version and buffer size.

Approximate core storage requirements for each program version, including system routines but excluding I/O buffers, are indicated below.

HVOSM-RD2	133200 bytes
HVOSM-VD2	148300 bytes

```
//  EXEC LOADGO,GCORE=320K,GTIME='(1,00)'
//GO.SYSLIN DD DSN=LOADLIB(DSHVOSR2),DISP=SHR
//GO.FT01F001 DD UNIT=9TRACK,DSN=LCDS.ROLL,DISP=(NEW,CATLG),
// DCB=(RECFM=VBS,LRECL=200,BLKSIZE=8004),LABEL=(1,,,OUT,RETPD=100)
//GO.FT02F001 DD DSN=&&DSIN,UNIT=SYSDA,DISP=(NEW,DELETE),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=6400),SPACE=(TRK,(1,1),RLSE)
//GO.FT03F001 DD DUMMY
//GO.FT04F001 DD DUMMY
//GO.FT11F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT12F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT13F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT14F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT15F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT16F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT17F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT18F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT19F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT20F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT21F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT22F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT23F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT24F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT25F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT26F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT27F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT28F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT29F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.FT30F001 DD SYSOUT=A,SPACE=(TRK,(0,15),RLSE),
// DCB=(RECFM=VBA,BLKSIZE=6447,LRECL=137,BUFNO=2)
//GO.SYSIN DD *
```

HVOSM DATA DECK

.

.

.

Figure 4.1-1 HVOSM JOB CONTROL LANGUAGE

```
// EXEC LOADGO,GLINES=15000,GTIME='(,30)'
***  CAL PROCEDURES  ASMFG, ASMGG, ASMHG, FORTGG, FORTHG, GO, LOADGO    C(
***
XX   PROC LCOPY=1,LLINES=1000,LLIB=DUMMY,LPARM=,ENTRY=MAIN,
XX         LOUT=A,LFORMS=STD1,GOUT=A,GFORMS=STD1,
XX         GCARDS=2000,GOUT=A,GFORMS=STD1,GCOPY=1,GLINES=5000,GCORE=128K,
XX         GTIME='(,20)',GPARM=,DUMP=DEFAULT,
XX         PLOTTER=NONE,LONG=X,PAPER=SG
***
XXGO   EXEC  PGM=LOADER,PARM='SIZE=&GCORE,&LPARM,EP=&ENTRY/&GPARM',
IEF653I SUBSTITUTION JCL - PGM=LOADER,PARM='SIZE=128K,,EP=MAIN/',
XX    REGION=&GCORE,TIME=&GTIME,ACCT=&DUMP
IEF653I SUBSTITUTION JCL - REGION=128K,TIME=(,30),ACCT=DEFAULT
***
XXFT05F001 DD DDNAME=SYSIN
***
XXFT06F001 DD SYSOUT=(&GOUT,COPY&GCOPY,&GFORMS),SPACE=(4,&GLINES,RLSE),
IEF653I SUBSTITUTION JCL - SYSOUT=(A,COPY1,STD1),SPACE=(4,15000,RLSE),
XX               UNIT=SYSOUT,
XX               DCB=(BLKSIZE=6447,RECFM=VBA,LRECL=137,BUFNO=2)
***
XXFT07F001 DD SYSOUT=B,UNIT=SYSOUT,
XX               SPACE=(1,&GCARDS,RLSE),
IEF653I SUBSTITUTION JCL - SPACE=(1,2000,RLSE),
XX               DCB=(BLKSIZE=3120,RECFM=FB,LRECL=80,BUFNO=2)
***
XXPLOTLIB   DD  DSN=SYSTEM.GENSUBS,DISP=SHR
***
XXPLOTTER  DD  DDNAME=&PLOTTER
IEF653I SUBSTITUTION JCL - DDNAME=NONE
***
XXCALCOMP DD SYSOUT=(C,PLOTWRTR,&PAPER),UNIT=SYSDA,
IEF653I SUBSTITUTION JCL - SYSOUT=(C,PLOTWRTR,SG),UNIT=SYSDA,
XX             DCB=(LRECL=400,BLKSIZE=4000,BUFNO=2),
XX             SPACE=(TRK,(100,50),RLSE),FCB=&LONG.&PAPER
IEF653I SUBSTITUTION JCL - SPACE=(TRK,(100,50),RLSE),FCB=XSG
***
XXLDX    DD DSN=&&&PLOTTER,UNIT=SYSDA,DISP=(MOD,PASS),
IEF653I SUBSTITUTION JCL - DSN=&&NONE,UNIT=SYSDA,DISP=(MOD,PASS),
XX          SPACE=(CYL,(20,10)),FCB=&LONG
IEF653I SUBSTITUTION JCL - SPACE=(CYL,(20,10)),FCB=X
***
XXSYSLIB  DD DSN=&LLIB,DISP=SHR,DCB=(RECFM=U,BLKSIZE=13030)
IEF653I SUBSTITUTION JCL - DSN=DUMMY,DISP=SHR,DCB=(RECFM=U,BLKSIZE=13030)
XX          DD  DSN=SYSTEM.FORTSUBS,DISP=SHR
XX          DD  DSN=SYS1.FORTLIB,DISP=SHR
XX          DD  DSN=SYSTEM.GENSUBS,DISP=SHR
XX          DD  DSN=SYSTEM.IMSL,DISP=SHR
***
XXSYSLOUT DD SYSOUT=(&LOUT,COPY&LCOPY,&LFORMS),SPACE=(1,&LLINES,RLSE),
IEF653I SUBSTITUTION JCL - SYSOUT=(A,COPY1,STD1),SPACE=(1,1000,RLSE),
XX    UNIT=SYSDA,
XX        DCB=(BLKSIZE=3146,RECFM=FBSA,LRECL=121,BUFNO=2)
***
XXSYSUDUMP  DD  SYSOUT=A,UNIT=SYSOUT,SPACE=(TRK,(0,30),RLSE)
***
//GO.SYSLIN DD DSN=LOADLIB(DSRCMOD),DISP=SHR
//GO.SYSIN DD *
```

Figure 4.1-2   Calspan Procedure LOADGO

5. HVOSM PREPROCESSING PROGRAM

A listing os the HVOSM Preprocessing Program routines is provided in this section.

```
      REAL*4 LW
      READ(5,100) IVEH,IVER,IOUT,IRD,IFD,LW
  100 FORMAT(5I4,F10.0)
      IF(IVEH.NE.0) CALL VEHLIB(IVEH,IVER,IOUT)
      IF(LW.NE.0.0) CALL VEHCAL(LW)
      IF(IRD.EQ.0) GO TO 10
      READ(5,101) S1,S2,S3,Y1,Y2,R,B,DBS
  101 FORMAT(8F8.0)
      CALL RBDTCH(S1,S2,S3,Y1,Y2,R,B,DBS)
   10 IF(IFD.EQ.0) GO TO 99
      READ(5,102) S1,S2,S3,Y1,Y2,Y3,R,B1,B2,DBS
  102 FORMAT(10F8.0)
      CALL FBDTCH(S1,S2,S3,Y1,Y2,Y3,R,B1,B2,DBS)
   99 STOP
      END
```

```
      SUBROUTINE VEHCAL(LW)
      DIMENSION PS2PI(3),PS2I(3),PPI(2),PSPI(3)
      REAL*4LW,LW2,LW3
      DATA PS2PI/4HLB-S,4HEC**,4H2/IN/  ,
     1      PS2I/4HLB-S,4HEC**,4H2-IN/  ,
     2      PSPI/4HLB-S,4HEC/I,4HN   /  ,
     3       PPI/4HLB/I,4HN   /         ,
     4       DIN/4HIN  /                ,
     5       PLB/4HLB  /
      DATA G/386.4/, PI/3.1415927/
C********
C       .CALCULATION OF VEHICLE PARAMETERS FROM:
C     BASSO,G.L., "FUNCTIONAL DERIVATION OF VEHICLE PARAMETERS FOR
C     DYNAMIC STUDIES", NATIONAL RESEARCH COUNCIL OF CANADA, NATIONAL
C     AERONAUTICAL ESTABLISHMENT, REPORT NO. LTR-ST-747, SEP 1974
C*******
      LW2 = LW*LW
      LW3= LW2*LW
      WT = 2.451E-3*LW3
      WUT = 126.6+0.111*WT
      WUF = 0.385*WUT
      WUR = WUT-WUF
      WS = WT-WUT
      WFT = (62.727-0.0629*LW)*WT/100.
      WRT = WT-WFT
      WFS = WFT-WUF
      WRS = WRT-WUR
      A = WRS*LW/WS
      B = LW-A
      XMS = WS/G
      XMUF = WUF/G
      XMUR = WUR/G
      TF = 12.571+0.419*LW
      TR = 11.211+0.428*LW
      XIZ = XMS*26.352*WT**0.577
      XIX = XMS*4.752*WT**0.546
      XIYT = (3.1104*WT**1.82)/G
      XIYU = XMUF*(144.+A*A)+XMUR*(144.+B*B)
      XIY = XIYT-XIYU
      XIXZ = 0.0
      XIR = 0.12484*XMUR*TR*TR
      FN = 1.696-1.415E-4*WT
      SK = 4.0*FN*FN*PI*PI*XMS
      RK = 42.17+0.125E-2*WT
      AKF = RK*SK/200.
      AKR = 0.5*SK-AKF
      CF = 0.246*SQRT(AKF*WFS/(2.0*G))
      CR = 0.416*SQRT(AKR*WRS/(2.0*G))
      TS = 0.702*TR
```

```
      WRITE(6,100)  LW    ,DIN  ,WT   ,PLB  ,WS   ,PLB  ,WUF  ,PLB  ,WUR  ,PLB
      WRITE(6,101)  XMS         ,PS2PI      ,XMUF       ,PS2PI      ,
     1              XMUR        ,PS2PI      ,XIX        ,PS2I       ,
     2              XIY         ,PS2I       ,XIZ        ,PS2I       ,
     3              XIXZ        ,PS2I       ,XIR        ,PS2I
      WRITE(6,102)  A           ,DIN        ,B          ,DIN        ,
     1              TF          ,DIN        ,TR         ,DIN        ,
     2              TS          ,DIN        ,AKF        ,PPI        ,
     3              AKR         ,PPI        ,CF         ,PSPI       ,
     4              CR          ,PSPI
  100 FORMAT(1H1
     1  9X,36HWHEELBASE                           LW    =,F10.3,2X,A4  /
     2 10X,36HTOTAL VEHICLE WEIGHT                WT    =,F10.3,2X,A4  /
     3 10X,36HSPRUNG WEIGHT                       WS    =,F10.3,2X,A4  /
     4 10X,36HFRONT UNSPRUNG WEIGHT               WUF   =,F10.3,2X,A4  /
     5 10X,36HREAR UNSPRUNG WEIGHT                WUR   =,F10.3,2X,A4  )
  101 FORMAT(1H0,19X,22HHVOSM INPUT PARAMETERS,25X,10HCARD FIELD  //
     1 9X,31HSPRUNG MASS                     XMS  =,F10.3,2X,3A4,3X,7H201   1/
     2 9X,31HFRONT UNSPRUNG MASS             XMUF =,F10.3,2X,3A4,3X,7H201   2/
     3 9X,31HREAR UNSPRUNG MASS              XMUR =,F10.3,2X,3A4,3X,7H201   3/
     4 9X,31HROLL INERTIA                    XIX  =,F10.3,2X,3A4,3X,7H201   4/
     5 9X,31HPITCH INERTIA                   XIY  =,F10.3,2X,3A4,3X,7H201   5/
     6 9X,31HYAW INERTIA                     XIZ  =,F10.3,2X,3A4,3X,7H201   6/
     7 9X,31HROLL-YAW INERTIA PRODUCT        XIXZ =,F10.3,2X,3A4,3X,7H201   7/
     8 9X,31HREAR AXLE ROLL INERTIA          XIR  =,F10.3,2X,3A4,3X,7H201   8)
  102 FORMAT(1H ,
     1 8X,31HSPRUNG MASS CG. LOCATION        A    =,F10.3,2X,A4,11X,7H202   1/
     2 9X,31H                                B    =,F10.3,2X,A4,11X,7H202   2/
     3 9X,31HFRONT TRACK                     TF   =,F10.3,2X,A4,11X,7H202   3/
     4 9X,31HREAR TRACK                      TR   =,F10.3,2X,A4,11X,7H202   4/
     5 9X,31HREAR SPRING TRACK               TS   =,F10.3,2X,A4,11X,7H202   6/
     6 9X,31HFRONT SPRING RATE               AKF  =,F10.3,2X,2A4,7X,7H204   1/
     7 9X,31HREAR SPRING RATE                AKR  =,F10.3,2X,2A4,7X,7H205   1/
     8 9X,31HFRONT DAMPING COEF.             CF   =,F10.3,2X,3A4,3X,7H206   1/
     9 9X,31HREAR DAMPING COEF.              CR   =,F10.3,2X,3A4,3X,7H206   4)
      RETURN
      END
```

```
      SUBROUTINE VEHLIB(IVEH,IVER,IOUT)
      COMMON /VDATA/ VEH(10,50,10),ZCOND(5,10),NRDC(10),NVDC(10)
      REAL*8 VEH
      WRITE(6,1000)
 1000 FORMAT(1H1,
     1 10X,11H***CAUTION:    /
     2 10X,57H   VALUES OF ZF, ZR GIVEN IN FIELDS 7, 8 OF CARD 203 ARE /
     3 10X,57HBASED ON, AND ARE CONSISTENT WITH, LOADING CONDITIONS,   /
     4 10X,57HTIRE RADII AND TIRE RATES AS PER REFERENCE INDICATED IN  /
     5 10X,57HBLOCK DATA SUBROUTINE.  THE VALUES USED IN DETERMINING   /
     6 10X,57HZF, ZR ARE PRINTED WITH THE DATA SET BELOW.  IF CHANGES  /
     7 10X,57HARE MADE TO THE INITIAL CENTER OF GRAVITY ELEVATION, TIRE/
     8 10X,57HRADII OR RATES, INITIAL EQUILIBRIUM WILL NOT BE MAINTAIN-/
     9 10X,57HED IF ZF, ZR ARE INPUT.  BY DELETENG ZF, ZR FROM CARD 203/
     A 10X,57HINITIAL EQUILIBRIUM CAN BE ASSURED BY AN AUTOMATIC INT-  /
     B 10X,57HERNAL CALL TO SUBROUTINE INITEQ BASED ON THE SUPLIED     /
     C 10X,50HTIRE PROPERTIES AND CENTER OF GRAVITY LOCATION.      // )
      N = NRDC(IVEH)
      IF(IVER.EQ.2) N = NVDC(IVEH)
      WRITE(6,1001) ((VEH(I,J,IVEH),I=1,10),J=1,N)
 1001 FORMAT(5X,10A8)
      WRITE(6,1002) (ZCOND(I,IVEH),I=1,5)
 1002 FORMAT(1H0,
     1  9X,56H   VALUES OF ZF, ZR ON CARD 203 WERE COMPUTED BASED ON   /
     2 10X,26HTHE FOLLOWING PARAMETERS :    /
     3 10X,36HSPRUNG MASS CG HEIGHT ABOVE GROUND =,F10.3,2X,2HIN    /
     4 10X,20HTIRE RADII    FRONT =,F10.3,2X,2HIN   /
     5 10X,20H              REAR  =,F10.3,2X,2HIN   /
     6 10X,20HTIRE RATES    FRONT =,F10.3,2X,5HLB/IN   /
     7 10X,20H              REAR  =,F10.3,2X,5HLB/IN   )
      IF(IOUT.NE.0) WRITE(IOUT,1003) ((VEH(I,J,IVEH),I=1,10),J=1,N)
 1003 FORMAT(10A8)
      RETURN
      END
```

```
      BLOCK DATA
      COMMON /VDATA/ VEH(10,50,10),ZCOND(5,10),NRDC(10),NVDC(10)
      DIMENSION V11(10,20),V12(10,20),V13(10,20),V14(10,20),V15(10,20),
     1          V21(10,20),V22(10,20),V31(10,20),V32(10,20),
     2          V41(10,20),V42(10,20),V51(10,20),V52(10,20),
     3          V61(10,20),V62(10,20)
      REAL*8 VEH,V11,V12,V13,V14,V15,V21,V22,V31,V32,V41,V42,V51,V52,
     1     V61,V62
      EQUIVALENCE (V11(1,1),VEH(1,1,1)), (V12(1,1),VEH(1,10,1)),
     1            (V13(1,1),VEH(1,19,1)),(V14(1,1),VEH(1,28,1)),
     2            (V15(1,1),VEH(1,37,1)),
     3            (V21(1,1),VEH(1,1,2)), (V22(1,1),VEH(1,10,2)),
     4            (V31(1,1),VEH(1,1,3)), (V32(1,1),VEH(1,10,3)),
     5            (V41(1,1),VEH(1,1,4)), (V42(1,1),VEH(1,10,4)),
     6            (V51(1,1),VEH(1,1,5)), (V52(1,1),VEH(1,10,5))
     7            (V61(1,1),VEH(1,1,6)), (V62(1,1),VEH(1,10,6))
      DATA NRDC/18,14,14,14,15,10,4*0/
      DATA NVDC/38,15,15,15,16,10,4*0/
      DATA ZCOND/23.0   ,14.0   ,14.0   ,1098.  ,1098.  ,
     1           24.6   ,14.4   ,14.4   ,1210.  ,1680.  ,
     2           24.03  ,13.2   ,13.2   ,1450.  ,1450.  ,
     3           19.48  ,12.8   ,12.8   ,1500.  ,1500.  ,
     4           23.17  ,12.6   ,12.6   ,760.   ,1060.  ,
     5           19.85  ,11.83  ,11.83  ,1240.  ,1240.  /
C
C  1963 FORD DATA (VEH. NO. 1) FROM REFERENCE 1
C
      DATA V11/
     1 8H    1963, 8H FORD GA, 8HLAXY FOU, 8HR - DOOR, 8H SEDAN  ,
     2 8H        , 8H        , 8H        , 8H        , 8H     200,
     3 8H10.818  , 8H0.608   , 8H0.945   , 8H6000.   , 8H35477.  ,
     4 8H35800.  , 8H-192.   , 8H435.6   , 8H        , 8H     201,
     5 8H54.63   , 8H64.62   , 8H61.2    , 8H60.5    , 8H-2.0    ,
     6 8H46.52   , 8H        , 8H        , 8H        , 8H     202,
     7 8H        , 8H        , 8H        , 8H        , 8H        ,
     8 8H        , 8H10.138  , 8H12.038  , 8H        , 8H     203,
     9 8H131.0   , 8H300.    , 8H600.    , 8H300.    , 8H600.    ,
     A 8H.05     , 8H-3.0    , 8H5.0     , 8H        , 8H     204,
     B 8H194.0   , 8H300.    , 8H600.    , 8H300.    , 8H600.    ,
     C 8H.05     , 8H-4.0    , 8H4.5     , 8H        , 8H     205,
     D 8H1.3     , 8H58.0    , 8H0.001   , 8H1.75    , 8H97.0    ,
     E 8H0.001   , 8H        , 8H        , 8H        , 8H     206,
     F 8H266000. , 8H59244.  , 8H0.059   , 8H        , 8H        ,
     G 8H        , 8H        , 8H        , 8H        , 8H     207,
     H 8H492.0   , 8H600.    , 8H0.4     , 8H5000.   , 8H0.075   ,
     I 8H1.5     , 8H        , 8H        , 8H        , 8H     208/
      DATA V12/
     1 8H-5.0    , 8H5.0     , 8H1.0     , 8H        , 8H        ,
     2 8H        , 8H        , 8H        , 8H        , 8H     209,
```

```
3 8H-5.7     , 8H-3.9     , 8H-2.45    , 8H-1.3     , 8H-0.4     ,     00000500
4 8H0.3      , 8H0.6      , 8H0.65     , 8H0.3      , 8H    1 209,     00000510
5 8H-0.4     , 8H-1.3     , 8H         , 8H         , 8H         ,     00000520
6 8H         , 8H         , 8H         , 8H         , 8H    2 209,     00000530
7 8H-5.0     , 8H5.0      , 8H0.5      , 8H         , 8H         ,     00000540
8 8H         , 8H         , 8H         , 8H         , 8H      210,     00000550
9 8H.1079    , 8H.1053    , 8H.1030    , 8H.1011    , 8H.0994    ,     00000560
A 8H.0981    , 8H.0971    , 8H.0964    , 8H.0959    , 8H    1 210,     00000570
B 8H.0958    , 8H.0960    , 8H.0965    , 8H.0973    , 8H.0984    ,     00000580
C 8H.0998    , 8H.1015    , 8H.1035    , 8H.1058    , 8H    2 210,     00000590
D 8H.1085    , 8H.1114    , 8H.1147    , 8H         , 8H         ,     00000600
E 8H         , 8H         , 8H         , 8H         , 8H    3 210,     00000610
F 8H-5.0     , 8H5.0      , 8H5.0      , 8H         , 8H         ,     00000620
G 8H         , 8H         , 8H         , 8H         , 8H      211,     00000630
H 8H0.092    , 8H0.092    , 8H0.092    , 8H         , 8H         ,     00000640
I 8H         , 8H         , 8H         , 8H         , 8H    1 211/     00000650
 DATA V13/                                                             00000660
1 8H0.0      , 8H12.2     , 8H6.5      , 8H13.6     , 8H1.0      ,     00000670
2 8H3.0      , 8H         , 8H         , 8H         , 8H      212,     00000680
3 8H1.0      , 8H1.0      , 8H1000.    , 8H1000.    , 8H110.     ,     00000690
4 8H192.     , 8H0.1      , 8H         , 8H         , 8H      213,     00000700
5 8H3.0      , 8H3.0      , 8H         , 8H         , 8H         ,     00000710
6 8H         , 8H         , 8H         , 8H         , 8H      214,     00000720
7 8H7.62     , 8H1.4      , 8H0.48     , 8H0.942    , 8H0.0      ,     00000730
8 8H3.12     , 8H6.21     , 8H6.43     , 8H4.62     , 8H    1 214,     00000740
9 8H1.0      , 8H9.25     , 8H0.384    , 8H0.0      , 8H10.0     ,     00000750
A 8H  10.E10, 8H  10.E10, 8H         , 8H         , 8H    2 214,     00000760
B 8H7.62     , 8H1.4      , 8H0.476    , 8H0.691    , 8H0.0      ,     00000770
C 8H3.12     , 8H6.21     , 8H6.43     , 8H4.62     , 8H    3 214,     00000780
D 8H1.0      , 8H9.25     , 8H0.381    , 8H0.0      , 8H10.0     ,     00000790
E 8H  10.E10, 8H  10.E10, 8H         , 8H         , 8H    4 214,     00000800
F 8H500.     , 8H4900.    , 8H400.     , 8H         , 8H         ,     00000810
G 8H         , 8H         , 8H         , 8H         , 8H      215,     00000820
H 8H500.     , 8H563.     , 8H594.     , 8H618.     , 8H630.     ,     00000830
I 8H621.     , 8H600.     , 8H561.     , 8H516.     , 8H    1 215/     00000840
 DATA V14/                                                             00000850
1 8H480.     , 8H438.     , 8H420.     , 8H         , 8H         ,     00000860
2 8H         , 8H         , 8H         , 8H         , 8H    2 215,     00000870
3 8H0.0      , 8H-120.    , 8H-144.    , 8H-165.    , 8H-180.    ,     00000880
4 8H-192.    , 8H-204.    , 8H-216.    , 8H-231.    , 8H    3 215,     00000890
5 8H-249.    , 8H-267.    , 8H-288.    , 8H         , 8H         ,     00000900
6 8H         , 8H         , 8H         , 8H         , 8H    4 215,     00000910
7 8H0.0      , 8H1000.    , 8H20.0     , 8H         , 8H         ,     00000920
8 8H         , 8H         , 8H         , 8H         , 8H      216,     00000930
9 8H0.960    , 8H0.974    , 8H0.985    , 8H0.996    , 8H1.0      ,     00000940
A 8H1.030    , 8H1.010    , 8H1.0      , 8H0.995    , 8H    1 216,     00000950
B 8H0.982    , 8H0.972    , 8H0.952    , 8H0.930    , 8H0.907    ,     00000960
C 8H0.859    , 8H0.814    , 8H0.770    , 8H0.727    , 8H    2 216,     00000970
D 8H0.687    , 8H0.645    , 8H0.609    , 8H0.586    , 8H0.561    ,     00000980
E 8H0.536    , 8H0.515    , 8H0.550    , 8H0.488    , 8H    3 216,     00000990
F 8H0.475    , 8H0.465    , 8H0.454    , 8H0.444    , 8H0.441    ,     00001000
```

```
     G 8H0.438   , 8H0.435   , 8H0.432   , 8H0.429   , 8H   4 216,
     H 8H0.425   , 8H0.422   , 8H0.419   , 8H0.416   , 8H0.414   ,
     I 8H0.410   , 8H0.407   , 8H0.404   , 8H0.401   , 8H   5 216/
      DATA V15/
     1 8H0.398   , 8H0.395   , 8H0.391   , 8H0.388   , 8H0.385   ,
     2 8H0.382   , 8H        , 8H        , 8H        , 8H   6 216,
     3 8H9.6136-5, 8H.02853  , 8H60.336  , 8H        , 8H        ,
     4 8H        , 8H        , 8H        , 8H        , 8H     217/
C
C  1971 DODGE DATA (VEH.NO. 2) FROM REFERENCE 2
C
      DATA V21/
     1 8H    1971, 8H DODGE C, 8HORONET  , 8HCURB LOA, 8HDING    ,
     2 8H        , 8H        , 8H        , 8H        , 8H     200,
     3 8H7.57    , 8H0.51    , 8H0.82    , 8H312.    , 8H22800.  ,
     4 8H22600.  , 8H530.    , 8H550.    , 8H        , 8H     201,
     5 8H47.7    , 8H70.3    , 8H59.8    , 8H61.8    , 8H        ,
     6 8H47.0    , 8H        , 8H        , 8H        , 8H     202,
     7 8H        , 8H        , 8H        , 8H        , 8H        ,
     8 8H        , 8H11.5    , 8H11.5    , 8H        , 8H     203,
     9 8H105.    , 8H84.0    , 8H0.0     , 8H483.0   , 8H0.0     ,
     A 8H0.5     , 8H-2.4    , 8H2.1     , 8H        , 8H     204,
     B 8H120.    , 8H204.    , 8H0.0     , 8H744.    , 8H0.0     ,
     C 8H0.5     , 8H-4.4    , 8H3.6     , 8H        , 8H     205,
     D 8H6.85    , 8H40.0    , 8H.01     , 8H5.48    , 8H38.0    ,
     E 8H.01     , 8H        , 8H        , 8H        , 8H     206,
     F 8H40400.  , 8H-5100.  , 8H0.02    , 8H        , 8H        ,
     G 8H        , 8H        , 8H        , 8H        , 8H     207,
     H 8H-3.0    , 8H3.0     , 8H1.0     , 8H        , 8H        ,
     I 8H        , 8H        , 8H        , 8H        , 8H     209/
      DATA V22/
     1 8H-.52    , 8H.04     , 8H.27     , 8H.31     , 8H.08     ,
     2 8H-.54    , 8H-.95    , 8H        , 8H        , 8H   1 209,
     3 8H-3.0    , 8H3.0     , 8H1.0     , 8H        , 8H        ,
     4 8H        , 8H        , 8H        , 8H        , 8H     210,
     5 8H.04     , 8H.07     , 8H.10     , 8H.13     , 8H.16     ,
     6 8H.19     , 8H.22     , 8H        , 8H        , 8H   1 210,
     7 8H-3.0    , 8H3.0     , 8H1.0     , 8H        , 8H        ,
     8 8H        , 8H        , 8H        , 8H        , 8H     211,
     9 8H.60     , 8H.45     , 8H.30     , 8H.15     , 8H0.0     ,
     A 8H-.15    , 8H-.30    , 8H        , 8H        , 8H   1 211,
     B 8H0.0     , 8H9.4     , 8H0.7     , 8H9.4     , 8H1.0     ,
     C 8H2.71    , 8H        , 8H        , 8H        , 8H     212/
C
C 1971 CHEVROLET DATA (VEH NO. 3) FROM REFERENCE 2
C
      DATA V31/
     1 8H    1971, 8H CHEVROL, 8HET BROOK, 8HWOOD STA, 8HTION WAG,
     2 8HON   CUR, 8HB LOADIN, 8HG       , 8H        , 8H     200,
     3 8H10.3    , 8H0.63    , 8H1.03    , 8H5920.   , 8H41400.  ,
     4 8H41700.  , 8H1790.   , 8H750.    , 8H        , 8H     201,
```

```
     5 8H64.3    , 8H60.7    , 8H63.5    , 8H63.5    , 8H          ,
     6 8H45.3    , 8H        , 8H        , 8H        , 8H      202,
     7 8H        , 8H        , 8H        , 8H        , 8H          ,
     8 8H        , 8H11.1    , 8H10.8    , 8H        , 8H      203,
     9 8H141.    , 8H141.    , 8H0.0     , 8H451.2   , 8H0.0       ,
     A 8H0.5     , 8H-3.5    , 8H2.3     , 8H        , 8H      204,
     B 8H210.    , 8H189.    , 8H0.0     , 8H861.    , 8H0.0       ,
     C 8H0.5     , 8H-3.6    , 8H2.3     , 8H        , 8H      205,
     D 8H5.07    , 8H43.0    , 8H.01     , 8H4.41    , 8H73.0      ,
     E 8H.01     , 8H        , 8H        , 8H        , 8H      206,
     F 8H408000. , 8H-62000. , 8H0.033   , 8H        , 8H          ,
     G 8H        , 8H        , 8H        , 8H        , 8H      207,
     H 8H-4.0    , 8H3.0     , 8H1.0     , 8H        , 8H          ,
     I 8H        , 8H        , 8H        , 8H        , 8H      209/
      DATA V32/
     1 8H.28     , 8H.61     , 8H.79     , 8H.79     , 8H.5        ,
     2 8H0.0     , 8H-.83    , 8H-1.68   , 8H        , 8H    1 209,
     3 8H-4.0    , 8H3.0     , 8H1.0     , 8H        , 8H          ,
     4 8H        , 8H        , 8H        , 8H        , 8H      210,
     5 8H.25     , 8H.22     , 8H.19     , 8H.16     , 8H.13       ,
     6 8H.10     , 8H.07     , 8H.04     , 8H        , 8H    1 210,
     7 8H-4.0    , 8H4.0     , 8H4.0     , 8H        , 8H          ,
     8 8H        , 8H        , 8H        , 8H        , 8H      211,
     9 8H.09     , 8H.15     , 8H.21     , 8H        , 8H          ,
     A 8H        , 8H        , 8H        , 8H        , 8H    1 211,
     B 8H0.0     , 8H14.8    , 8H.07     , 8H14.8    , 8H1.0       ,
     C 8H3.08    , 8H        , 8H        , 8H        , 8H      212/
C
C    1971 PONTIAC DATA (VEH. NO. 4) FROM REFERENCE 2
C
      DATA V41/
     1 8H    1971, 8H PONTIAC, 8H TRANS-A, 8HM  CURB , 8HLOADING ,
     2 8H        , 8H        , 8H        , 8H        , 8H      200,
     3 8H8.0     , 8H0.53    , 8H0.82    , 8H2760.   , 8H18500.    ,
     4 8H18900.  , 8H230.    , 8H530.    , 8H        , 8H      201,
     5 8H40.     , 8H68.     , 8H61.9    , 8H60.4    , 8H          ,
     6 8H45.5    , 8H        , 8H        , 8H        , 8H      202,
     7 8H        , 8H        , 8H        , 8H        , 8H          ,
     8 8H        , 8H7.4     , 8H7.1     , 8H        , 8H      203,
     9 8H99.     , 8H99.     , 8H0.0     , 8H247.5   , 8H0.0       ,
     A 8H0.5     , 8H-2.0    , 8H2.5     , 8H        , 8H      204,
     B 8H147.    , 8H147.    , 8H0.0     , 8H588.    , 8H0.0       ,
     C 8H0.5     , 8H-3.3    , 8H3.7     , 8H        , 8H      205,
     D 8H7.28    , 8H3.5     , 8H.01     , 8H2.1     , 8H55.0      ,
     E 8H.01     , 8H        , 8H        , 8H        , 8H      206,
     F 8H356000. , 8H630000. , 8H-.008   , 8H        , 8H          ,
     G 8H        , 8H        , 8H        , 8H        , 8H      207,
     H 8H-2.0    , 8H4.0     , 8H1.0     , 8H        , 8H          ,
     I 8H        , 8H        , 8H        , 8H        , 8H      209/
      DATA V42/
     1 8H.54     , 8H.75     , 8H.75     , 8H.54     , 8H.10       ,
```

```
     2 8H-.47     , 8H-.73     , 8H         , 8H         , 8H    1 209,
     3 8H-2.0     , 8H4.0      , 8H1.0      , 8H         , 8H         ,
     4 8H         , 8H         , 8H         , 8H         , 8H      210,
     5 8H.03      , 8H.06      , 8H.09      , 8H.12      , 8H.15      ,
     6 8H.18      , 8H.21      , 8H         , 8H         , 8H    1 210,
     7 8H-2.0     , 8H4.0      , 8H1.0      , 8H         , 8H         ,
     8 8H         , 8H         , 8H         , 8H         , 8H      211,
     9 8H.12      , 8H.135     , 8H.15      , 8H.165     , 8H.18      ,
     A 8H.195     , 8H.21      , 8H         , 8H         , 8H    1 211,
     B 8H0.0      , 8H8.0      , 8H0.7      , 8H8.0      , 8H1.0      ,
     C 8H3.42     , 8H         , 8H         , 8H         , 8H      212/
C
C 1971 VOLKSWAGEN DATA (VEH NO. 5) FROM REFERENCE 2
C
      DATA V51/
     1 8H   1971  , 8HVOLKSWAG, 8HEN SUPER, 8H BEETLE , 8H         ,
     2 8H         , 8H         , 8H         , 8H         , 8H      200,
     3 8H4.23     , 8H0.36     , 8H0.57     , 8H1300.    , 8H8900.    ,
     4 8H7900.    , 8H0.0      , 8H         , 8H         , 8H      201,
     5 8H57.1     , 8H38.7     , 8H53.8     , 8H51.5     , 8H         ,
     6 8H         , 8H         , 8H         , 8H         , 8H      202,
     7 8H         , 8H         , 8H         , 8H         , 8H         ,
     8 8H         , 8H11.1     , 8H10.9     , 8H         , 8H      203,
     9 8H65.7     , 8H98.55    , 8H0.0      , 8H460.     , 8H0.0      ,
     A 8H0.5      , 8H-1.8     , 8H3.4      , 8H         , 8H      204,
     B 8H115.     , 8H69.0     , 8H0.0      , 8H333.5    , 8H0.0      ,
     C 8H0.5      , 8H-1.85    , 8H3.35     , 8H         , 8H      205,
     D 8H5.53     , 8H35.0     , 8H.01      , 8H4.27     , 8H40.0     ,
     E 8H.01      , 8H         , 8H         , 8H         , 8H      206,
     F 8H93000.   , 8H28300.   , 8H         , 8H0.0      , 8H.03025   ,
     G 8H-1.56E-2, 8H-6.48E-4, 8H         , 8H         , 8H      207,
     H 8H-2.0     , 8H3.0      , 8H1.0      , 8H         , 8H         ,
     I 8H         , 8H         , 8H         , 8H         , 8H      209/
      DATA V52/
     1 8H.04      , 8H.5       , 8H1.0      , 8H1.59     , 8H2.21     ,
     2 8H3.31     , 8H         , 8H         , 8H         , 8H    1 209,
     3 8H-2.74    , 8H-2.04    , 8H-1.3     , 8H-0.58    , 8H0.14     ,
     4 8H0.86     , 8H         , 8H         , 8H         , 8H    2 209,
     5 8H-3.0     , 8H3.0      , 8H3.0      , 8H         , 8H         ,
     6 8H         , 8H         , 8H         , 8H         , 8H      210,
     7 8H.09      , 8H0.0      , 8H-0.9     , 8H         , 8H         ,
     8 8H         , 8H         , 8H         , 8H         , 8H    1 210,
     9 8H-3.0     , 8H3.0      , 8H3.0      , 8H         , 8H         ,
     A 8H         , 8H         , 8H         , 8H         , 8H      211,
     B 8H.20      , 8H.29      , 8H.38      , 8H         , 8H         ,
     C 8H         , 8H         , 8H         , 8H         , 8H    1 211,
     D 8H0.0      , 8H7.35     , 8H0.3      , 8H7.35     , 8H1.0      ,
     E 8H4.13     , 8H         , 8H         , 8H         , 8H      212/
C
C 1971 VEGA DATA (VEH. NO. 6) FROM REFERENCE 3
C
```

```
      DATA V61/
     1 8H     1971, 8H VEGA 23, 8H00 SPORT, 8H COUPE   , 8H2-PASSEN,
     2 8HGER LOAD, 8H        , 8H        , 8H        , 8H     200,
     3 8H5.831   , 8H0.424   , 8H0.575   , 8H2000.   , 8H12000.  ,
     4 8H15600.  , 8H-100.   , 8H250.    , 8H        , 8H     201,
     5 8H43.87   , 8H53.13   , 8H55.1    , 8H54.1    , 8H1.31    ,
     6 8H38.0    , 8H        , 8H        , 8H        , 8H     202,
     7 8H        , 8H        , 8H        , 8H        , 8H        ,
     8 8H        , 8H8.58    , 8H7.21    , 8H        , 8H     203,
     9 8H96.0    , 8H        , 8H        , 8H        , 8H        ,
     A 8H        , 8H-2.2    , 8H3.84    , 8H        , 8H     204,
     B 8H121.0   , 8H        , 8H        , 8H        , 8H        ,
     C 8H        , 8H-2.2    , 8H4.85    , 8H        , 8H     205,
     D 8H2.0     , 8H37.0    , 8H0.01    , 8H2.0     , 8H58.0    ,
     E 8H0.01    , 8H        , 8H        , 8H        , 8H     206,
     F 8H0.0     , 8H11690.  , 8H-0.01   , 8H        , 8H        ,
     G 8H        , 8H        , 8H        , 8H        , 8H     207,
     H 8H-4.0    , 8H4.0     , 8H1.0     , 8H        , 8H        ,
     H 8H        , 8H        , 8H        , 8H        , 8H     209/
      DATA V62/
     1 8H-4.75   , 8H-3.08   , 8H-1.75   , 8H-0.73   , 8H0.0     ,
     2 8H0.48    , 8H0.65    , 8H0.78    , 8H0.83    , 8H   1 209/
C********
C          REFERENCES
C  1  MCHENRY,R.R. AND DELEYS,N.J., "VEHICLE DYNAMICS IN SINGLE
C     VEHICLE ACCIDENTS - VALIDATION AND EXTENSIONS OF A COMPUTER
C     PROGRAM", CALSPAN REPORT NO. VJ-2251-V-3, DEC 1968
C
C  2  SCHURING,D.J.,KUNKEL,D.T.,MASSING,D.E.,ROLAND,R.D.,"THE INFLUENCE
C     OF TIRE PROPERTIES ON PASSENGER VEHICLE HANDLING - VOLUME III -
C     APPENDICIES A-E", CALSPAN REPORT NO. ZM-5350-K-3, JUNE 1974
C
C  3  DELEYS,N.J. AND SEGAL,D.J., "VEHICLE REDIRECTION EFFECTIVENESS
C     OF MEDIAN BERMS AND CURBS", CALSPAN REPORT NO. HF-5095-V-2,
C     MAY 1973
C
C********
      END
```

```
      SUBROUTINE FBDTCH(S1,S2,S3,Y1,Y2,Y3,R,B1,B2,DBS)
C******
C        PROGRAM TO COMPUTE AND PUNCH HVOSM V-4 TERRAIN TABLES FOR ROUND
C        FLAT DITCH BOTTOM CROSS SECTIONS
C******
C        INPUTS REQUIRED:
C          S1 = SHOULDER SLOPE
C          S2 = SIDE SLOPE
C          S3 = BACK SLOPE
C          Y1 = LATERAL POSITION OF SHOULDER BREAK (INCHES)
C          Y2 = LATERAL POSITION OF INTERSECTION OF SIDE SLOPE AND FLAT
C              DITCH BOTTOM (INCHES)
C          Y3 = LATERAL POSITION OF INTERSECTION OF FLAT DITCH BOTTOM
C               AND BACK SLOPE (INCHES)
C          R = TANGENT POINT OF SHOULDER ROUNDING FROM SHOULDER BREAK
C              MEASURED ALONG SHOULDER AND SIDE SLOPES (INCHES)
C          B1 = TANGENT POINT OF SIDE SLOPE-DITCH BOTTOM ROUNDING FROM
C               INTERSECTION MEASURED ALONG SLOPE AND BOTTOM (INCHES)
C          B2 = TANGENT POINT OF DITCH BOTTOM-BACK SLOPE ROUNDING FROM
C               INTERSECTION MEASURED ALONG BOTTOM AND SLOPE (INCHES)
C          DBS= LATERAL RUN-OUT DISTANCE OF THE BACK SLOPE (INCHES)
C******
C        OUTPUT:
C          PRINTED AND PUNCHED TERRAIN TABLES INCLUDING HVOSM CARD 14
C          (USER MUST SUPPLY TABLE FRICTION COEFFICIENTS ON THIS CARD).
C          TABLE 1 = SHOULDER
C          TABLE 2 = SHOULDER-SIDE SLOPE ROUNDING
C          TABLE 3 = SIDE SLOPE
C          TABLE 4 = SIDE SLOPE-BOTTOM ROUNDING
C          TABLE 5 = DITCH BOTTOM, BACK SLOPE ROUNDING AND BACK SLOPE
C******
C         NOTE:
C
C         BEGINNING, END AND INCREMENT IN THE X' DIRECTION ARE THE SAME
C         FOR ALL FIVE TERRAIN TABLES AND ARE FIXED AT VALUES OF
C         XB=-500, XE=9500, AND DX=5000.
C
C******
C
C        METHOD:
C          THE THREE ROUNDINGS ARE COMPUTED AS CIRCULAR ARCS BETWEEN
C          TANGENT POINTS AS DETERMINED BY R,B1,AND B2.
C          THE EDGE OF PAVEMENT IS ASSUMED TO LIE ALONG THE X' AXIS
C          AT 0.0 ELEVATION.
C          IF THE ELEVATION DROP AT 10 FT. FROM THE EDGE OF PAVEMENT
C          DUE TO SHOULDER SLOPE AND ROUNDING EXCEEDS 10" THE SIDE SLOPE
C          IS MOVED LATERALLY TO MEET THIS CONSTRAINT.
C******
C        COMPUTATIONAL CONSTRAINTS:
```

```
C           THE THREE INPUT SLOPES MUST BE NON-ZERO.
C           THE BACK SLOPE IS ENTERED AS A NEGATIVE QUANTITY.
C           THE DITCH BOTTOM IS ASSUMED HORIZONTAL.
C           0<Y1<Y2<Y3
C           INPUTS MUST BE COMPATIBLE SUCH THAT THE LATERAL POSITION OF T
C           END OF THE SHOULDER ROUNDING IS LESS THAN THE LATERAL POSITIO
C           OF THE BEGINNING OF THE SIDE SLOPE-BOTTOM ROUNDING. IF THIS
C           CONDITION IS NOT MET AN ERROR MESSAGE IS OUTPUT.
C           Y3-Y2>B1+B2
C******
      DIMENSION X(10),Y(30),Z(30)
      IND = 0
    1 Z1 = Y1*S1
      Z2 = Z1+S2*(Y2-Y1)
      Z3 = Z2
      Y4 = Y1-R/SQRT(S1**2+1.)
      Z4 = Z1-S1*(Y1-Y4)
      Y5 = Y1+R/SQRT(S2**2+1.)
      Z5 = Z1+S2*(Y5-Y1)
      YC1 = S1*S2*(Z5-Z4+Y5/S2-Y4/S1)/(S1-S2)
      ZC1 = -YC1/S1+Z4+Y4/S1
      R1 = SQRT((Y4-YC1)**2+(Z4-ZC1)**2)
      WRITE(6,9000) YC1,ZC1,R1
 9000 FORMAT('1',10X,'YC1=',F10.2,10X,'ZC1=',F10.2,'R1=',F10.2)
      IF(IND.EQ.1) GO TO 10
      YY = 120.0
      IF  ((YY.GT.Y5).OR.(S1.GT.0.08333).OR.(YY.LT.Y4))  GO TO 10
      Z1T = ZC1-SQRT(R1**2-(YY-YC1)**2)
      IF(Z1T.LE.10.) GO TO 10
      ZZ = S1*YY
      DEL = 10.-ZZ
      Z1D = ZZ+DEL
      A = S1**2+1.
      D = -2.*(YY+S1**2*YC1-S1*ZC1+Z1D*S1)
      C = (S1*YC1)**2-2.*S1*ZC1*YC1+ZC1**2+2.*Z1D*S1*YC1-2.*Z1D*ZC1
      C = C+ZZ**2+2.*ZZ*DEL+DEL**2+YY**2-R1**2
      YT = (-D-SQRT(D**2-4.*A*C))/(2.*A)
      ZC1 = S1*YT+ZC1-S1*YC1
      YD = YT-YC1
      YC1 = YT
      Y1 = Y1+YD
      Y2 = Y2+YD
      Y3 = Y3+YD
      IND = 1
      WRITE(6,9001) YC1,ZC1,Y1,Y2,Y3
 9001 FORMAT('0','PROFILE MODS. YC1=',F10.2,' ZC1=',F10.2,' Y1=',F10.2,
     1       ' Y2=',F10.2,' Y3=',F10.2 )
      GO TO 1
   10 Y7 = Y2+B1
      Z7 = Z2
      Y6 = Y2-B1/SQRT(S2**2+1.)
```

```
      IF(Y6.GE.Y5) GO TO 569
      WRITE(6,7000)  Y5,Y6
 7000 FORMAT('0      INPUT INCOMPATABILITY  Y6<Y5  STOP' /
     1      10X,'Y5 =',F12.3,10X,'Y6 =',F12.3)
      STOP
  569 Z6 = Z2-S2*(Y2-Y6)
      YC2 = Y7
      R2 = (Y7-Y6)*SQRT(S2**2+1.)/S2
      ZC2 = Z7-R2
      Y8 = Y3-B2
      Z8 = Z3
      Y9 = Y3+B2/SQRT(S3**2+1.)
      Z9 = Z3+S3*(Y9-Y3)
      YC3 = Y8
      R3 = -(Y9-Y8)*SQRT(S3**2+1.)/S3
      ZC3 = Z3-R3
  200 WRITE(6,2001) S1,S2,S3,Y1,Y2,Y3,R,B1,B2,DBS
 2001 FORMAT(' S1=',F8.6,' S2=',F8.6,' S3=',F8.5,' Y1=',F8.2,' Y2=',
     1       F8.2,' Y3=',F8.2,' R=',F8.1,' B1=',F8.2,' B2=',F8.2,'
     2       DBS=',F8.1)
      NC = 500
      DO 100 M=6,7
      WRITE(M,9010) NC
 9010 FORMAT(76X,I4)
  100 CONTINUE
  101 IZ = 1
      ZI = 0.0
      XB = -500.
      XE = 9500.
      DX = 5000.
      YB = 0.
      YE = Y4
      XNB = 0.
      YNB = 0.
      DY = .5*Y4
      N = 3
      Z(1) = 0.
      Z(2) = .5*Z4
      Z(3) = Z4
      NCRD = 1
      DO 1011 I=4,9
 1011 Z(I) = 0.0
      GO TO 900
  102 IZ = 2
      ZI = 0.0
      YB = Y4
      YE = Y5
      N = (YE-YB)/6.+1
      IF(N.GT.21) N=21
      DY = (YE-YB)/(N-1)
      DO 110 I=1,N
```

```
      Y(I) = YB+(I-1)*DY
  110 Z(I) = ZC1-SQRT(R1**2-(Y(I)-YC1)**2)
      NCRD = N/9 + 1
      IF(MOD(N,9).EQ.0) NCRD = NCRD-1
      NF = 9*NCRD
      IF(NF.EQ.N) GO TO 900
      NP1 = N+1
      DO 1101 I=NP1,NF
 1101 Z(I) = 0.0
      GO TO 900
  103 IZ = 3
      ZI = 0.0
      YB = Y5
      YE = Y6
      N = 3
      DY = (YE-YB)/2.
      Z(1) = Z5
      Z(2) = Z5+.5*(Z6-Z5)
      Z(3) = Z6
      NCRD = 1
      DO 1031 I=4,9
 1031 Z(I) = 0.0
      GO TO 900
  104 IZ = 4
      ZI = 0.0
      YB = Y6
      YE = Y7
      N = (YE-YB)/6.+1
      IF(N.GT.21) N = 21
      DY = (YE-YB)/(N-1)
      DO 111 I=1,N
      Y(I) = YB+(I-1)*DY
  111 Z(I) = ZC2+SQRT(R2**2-(Y(I)-YC2)**2)
      NCRD = N/9 + 1
      IF(MOD(N,9).EQ.0) NCRD = NCRD-1
      NF = 9*NCRD
      IF(NF.EQ.N) GO TO 900
      NP1 = N+1
      DO 1041 I=NP1,NF
 1041 Z(I) = 0.0
      GO TO 900
  105 IZ = 5
      ZI = 1.0
      N = 21
      YB = Y7
      DY = (Y9-Y8)/18.
      Z(1) = Z8
      Y(1) = YB
      DO 742 J=1,19
      JJ = J+1
      Y(JJ) = Y8+(J-1)*DY
```

```
 742 Z(JJ) = ZC3+SQRT(R3**2-(Y(JJ)-YC3)**2)
     Y(21) = Y(20)+DBS
     Z(21) = Z(20)+S3*(Y(21)-Y(20))
     YE = Y(21)
     X(1) = XB
     X(2) = XB+DX
     X(3) = XE
     DO 1052 I=4,9
1052 X(I) = 0.0
     DX = 3.
     DY = 21.
     NCRD = 3
     DO 1051 I=22,27
     Z(I) = 0.0
1051 Y(I) = 0.0
 900 CONTINUE
     ICRD = 500+IZ
     DO 301 M=6,7
     WRITE(M,1000) XB,XE,DX,YB,YE,DY,XNB,YNB,ZI,ICRD
1000 FORMAT(9F8.2,4X,I4)
 301 CONTINUE
     ITAB = 0
     DO 901 K=1,3
     DO 902 J=1,NCRD
     ITAB = ITAB+1
     IB = J*9-8
     IE = IB+8
     DO 302 M=6,7
     WRITE(M,1001) (Z(I),I=IB,IE),ITAB,ICRD
 302 CONTINUE
1001 FORMAT(9F8.2,2I4)
 902 CONTINUE
 901 CONTINUE
     GO TO(102,103,104,105,950),IZ
 950 CONTINUE
     DO 303 M=6,7
     ITB = ITAB
     DO 310 J=1,3
     IB = J*9-8
     IE = IB+8
     ITB = ITB+1
     WRITE(M,1001) (Y(I),I=IB,IE),ITB,ICRD
 310 CONTINUE
     ITB = ITB+1
     WRITE(M,1001) (X(I),I=1,9),ITB,ICRD
     NC = 506
     WRITE(M,9010) NC
 303 CONTINUE
     RETURN
     END
```

```
      SUBROUTINE RBDTCH(S1,S2,S3,Y1,Y2,R,B,DBS)
C******
C        DITCH BOTTOM CROSS SECTIONS
C******
C        INPUTS REQUIRED:
C          S1=SHOULDER SLOPE
C          S2=SIDE SLOPE
C          S3=BACK SLOPE
C          Y1=LATERAL POSITION OF SHOULDER BREAK (INCHES)
C          Y2=LATERAL POSITION OF INTERSECTION OF SIDE AND BACK SLOPES (
C          R=TANGENT POINT OF SHOULDER ROUNDING FROM SHOULDER BREAK
C            MEASURED ALONG SHOULDER AND SIDE SLOPES (INCHES)
C          B=DITCH WIDTH, MEASURED HORIZONTALLY BETWEEN TANGENT POINTS (
C          DBS= LATERAL RUN-OUT DISTANCE OF THE BACK SLOPE (INCHES)
C******
C        OUTPUT:
C          PRINTED AND PUNCHED TERRAIN TABLES INCLUDING HVOSM CARD 14
C          USER MUST SUPPLY TABLE FRICTION COEFFICIENTS ON THIS CARD
C          TABLE 1 = SHOULDER
C          TABLE 2 = SHOULDER-SIDE SLOPE ROUNDING
C          TABLE 3 = SIDE SLOPE
C          TABLE 4 = DITCH BOTTOM ROUNDING
C          TABLE 5 = BACK SLOPE
C******
C         NOTE:
C
C         BEGINNING, END AND INCREMENT IN THE X' DIRECTION ARE THE SAME
C         FOR ALL FIVE TERRAIN TABLES AND ARE FIXED AT VALUES OF
C         XB=-500, XE=9500, AND DX=5000.
C
C******
C
C        METHOD:
C          SHOULDER-SIDE SLOPE ROUNDING IS COMPUTED AS A CIRCULAR ARC BE
C          TANGENT POINTS AS DETERMINED BY R.
C          DITCH BOTTOM IS COMPUTED AS TWO CIRCULAR ARCS AS DETERMINED
C          BY B AND GEOMETRIC SLOPE AND ELEVATION CONSTRAINTS.
C          THE EDGE OF PAVEMENT IS ASSUMED TO LIE ALONG THE X' AXIS
C          AT 0.0 ELEVATION.
C          IF THE ELEVATION DROP AT 10' FROM THE EDGE OF PAVEMENT DUE TO
C          SHOULDER SLOPE AND ROUNDING EXCEEDS 10" THE SIDE SLOPE IS MOV
C          LATERALLY TO MEET THIS CONSTRAINT
C******
C        COMPUTATIONAL CONSTRAINTS:
C          THE THREE INPUT SLOPES MUST BE ENTERED AS NON-ZERO
C          THE BACK SLOPE IS ENTERED AS A NEGATIVE QUANTITY
C          0<Y1<Y2
C          INPUTS MUST BE COMPATIBLE SUCH THAT THE LATERAL POSITION
C          OF THE END OF THE SHOULDER ROUNDING IS LESS THAN THE LATERAL
```

```
C          POSITION OF THE BEGINNING OF THE BOTTOM ROUNDING. IF THIS
C          CONDITION IS NOT MET AN ERROR MESSAGE IS OUTPUT.
C******
      DIMENSION X(10),Y(30),Z(30)
      IND = 0
    1 Z1 = Y1*S1
      Z2 = Z1+S2*(Y2-Y1)
      Y3 = Y1-R/SQRT(S1**2+1.)
      Z3 = Z1-R*S1/SQRT(S1**2+1.)
      Y4 = Y1+R/SQRT(S2**2+1.)
      Z4 = Z1+R*S2/SQRT(S2**2+1.)
      YC1 = S1*S2*(Z4-Z3+Y4/S2-Y3/S1)/(S1-S2)
      ZC1 = -YC1/S1+Z3+Y3/S1
      R1 = SQRT((Y3-YC1)**2+(Z3-ZC1)**2)
      WRITE(6,9000)YC1,ZC1,R1
 9000 FORMAT('1',10X,'YC1=',F10.2,10X,'ZC1=',F10.2,10X,'R1=',F10.2)
      IF(IND.EQ.1) GO TO 10
      YY = 120.
      IF  ((YY.GT.Y4).OR.(S1.GT.0.08333).OR.(YY.LT.Y3))  GO TO 10
      Z1T = ZC1-SQRT(R1*R1-(YY-YC1)**2)
      IF(Z1T.LE.10.) GO TO 10
      ZZ = S1*YY
      DEL = 10.-ZZ
      Z1D = ZZ+DEL
      A = S1**2+1.
      D = -2.*(YY+S1**2*YC1-S1*ZC1+Z1D*S1)
      C = (S1*YC1)**2-2.*S1*ZC1*YC1+ZC1**2+2.*Z1D*S1*YC1-2.*Z1D*ZC1
      C = C+ZZ**2+2.*ZZ*DEL+DEL**2+YY**2-R1**2
      YT = (-D-SQRT(D*D-4.*A*C))/(2.*A)
      ZC1 = S1*YT+ZC1-S1*YC1
      YD = YT-YC1
      YC1 = YT
      Y1 = Y1+YD
      Y2 = Y2+YD
      IND = 1
      WRITE(6,9001)YC1,ZC1,Y1,Y2
 9001 FORMAT('0','PROFILE MODS. YC1=',F10.2,10X,'ZC1=',F10.2,10X,
     1        'Y1=',F10.2,10X,'Y2=',F10.2)
      GO TO 1
   10 R2 = .5*B*SQRT(1.+S2**2)/S2
      R3 =-.5*B*SQRT(1.+S3**2)/S3
      A1 = R3/S3-SQRT(R3**2-(.5*B)**2)/S3
      A2 = -R2/S2+SQRT(R2**2-(.5*B)**2)/S2
      Z11 = S2*S3*(B+Z2/S3-Z1/S2+Y1-Y2+A1+A2)/(S2-S3)
      Z9 = Z11-R2
      Z10 = Z11-R3
      Z7 = Z9+SQRT(R2**2-(.5*B)**2)
      Z8 = Z10+SQRT(R3**2-(.5*B)**2)
      Y7 = Y1+(Z7-Z1)/S2
      IF(Y7.GT.Y4) GO TO 569
      WRITE(6,7000)
```

```
7000 FORMAT('0   INPUT INCOMPATIBILITY  Y7<Y4  STOP' )
     STOP
 569 Y8 = Y2+(Z8-Z2)/S3
     Y11 = Y7+B/2.
     YC2 = Y11
     ZC2 = Z9
     YC3 = Y11
     ZC3 = Z10
 200 WRITE(6,2001) S1,S2,S3,Y1,Y2,R,B,DBS
2001 FORMAT(' S1=',F8.6,' S2=',F8.6,' S3=',F8.5,' Y1=',F8.2,' Y2=',
    1       F8.2,' R=',F8.1,' B=',F8.2,' DBS=',F8.1)
     NC = 500
     DO 100 M=6,7
     WRITE(M,9010) NC
9010 FORMAT(76X,I4)
 100 CONTINUE
 101 IZ = 1
     ZI = 0.0
     XB = -500.
     XE = 9500.
     DX = 5000.
     YB = 0.
     YE = Y3
     XNB = 0.
     YNB = 0.
     DY = .5*Y3
     N = 3
     Z(1) = 0.
     Z(2) = .5*Z3
     Z(3) = Z3
     NCRD = 1
     DO 1011 I=4,9
1011 Z(I) = 0.0
     GO TO 900
 102 IZ = 2
     ZI = 0.0
     YB = Y3
     YE = Y4
     N = (YE-YB)/6.+1
     IF(N.GT.21) N=21
     DY = (YE-YB)/(N-1)
     DO 110 I=1,N
     Y(I) = YB+(I-1)*DY
 110 Z(I) = ZC1-SQRT(R1**2-(Y(I)-YC1)**2)
     NCRD = N/9 + 1
     IF(MOD(N,9).EQ.0) NCRD = NCRD-1
     NF = 9*NCRD
     IF(NF.EQ.N) GO TO 900
     NP1 = N+1
     DO 1101 I=NP1,NF
1101 Z(I) = 0.0
```

```
      GO TO 900
  103 IZ = 3
      ZI = 0.0
      YB = Y4
      YE = Y7
      N = 3
      DY = (YE-YB)/2.
      Z(1) = Z4
      Z(2) = Z4+.5*(Z7-Z4)
      Z(3) = Z7
      NCRD = 1
      DO 1031 I=4,9
 1031 Z(I) = 0.0
      GO TO 900
  104 IZ = 4
      ZI = 0.0
      YB = Y7
      YE = Y8
      N = (YE-YB)/6.+1
      IF(N.GT.21) N=21
      DY = (YE-YB)/(N-1)
      DO 111 I=1,N
      Y(I) = YB+(I-1)*DY
      IF(Y(I).GE.Y11) GO TO 112
      Z(I) = ZC2+SQRT(R2**2-(Y(I)-YC2)**2)
      GO TO 111
  112 Z(I) = ZC3+SQRT(R3**2-(Y(I)-YC3)**2)
  111 CONTINUE
      NCRD = N/9 + 1
      IF(MOD(N,9).EQ.0) NCRD = NCRD-1
      NF = 9*NCRD
      IF(NF.EQ.N) GO TO 900
      NP1 = N+1
      DO 1041 I=NP1,NF
 1041 Z(I) = 0.0
      GO TO 900
  105 IZ = 5
      ZI = 1.0
      YB = Y8
      YE = Y8+DBS
      DY = (YE-YB)/2.
      N = 3
      Z(1) = Z8
      Z(2) = Z8+S3*DY
      Z(3) = Z8+2.*S3*DY
      X(1) = XB
      X(2) = XB+DX
      X(3) = XE
      Y(1) = YB
      Y(2) = YB+DY
      Y(3) = YE
```

```
      NCRD = 1
      DO 1052 I=4,9
      Z(I) = 0.0
      Y(I) = 0.0
 1052 X(I) = 0.0
      DX = 3.
      DY = 3.
  900 CONTINUE
      ICRD = 500+IZ
      DO 301 M=6,7
  301 WRITE(M,1000) XB,XE,DX,YB,YE,DY,XNB,YNB,ZI,ICRD
 1000 FORMAT(9F8.2,4X,I4)
      ITAB = 0
      DO 901 K=1,3
      DO 902 J=1,NCRD
      ITAB = ITAB+1
      IB = J*9-8
      IE = IB+8
      DO 302 M=6,7
  302 WRITE(M,1001) (Z(I),I=IB,IE),ITAB,ICRD
 1001 FORMAT(9F8.2,2I4)
  902 CONTINUE
  901 CONTINUE
      GO TO(102,103,104,105,950),IZ
  950 CONTINUE
      ITB1 = ITAB+1
      ITB2 = ITB1+1
      NC = 506
      DO 303 M=6,7
      WRITE(M,1001) (Y(I),I=1,9),ITB1,ICRD
      WRITE(M,1001) (X(I),I=1,9),ITB2,ICRD
  303 WRITE(M,9010) NC
      RETURN
      END
```

6. REFERENCES

1. McHenry, R. R. and DeLeys, N. J., "Vehicle Dynamics in Single Vehicle Accidents - Validation and Extensions of a Computer Simulation," Calspan Report No. VJ-2251-V-3, December 1968.

2. "Automobile Accidents Related to Railroad Grade Crossings - A Study of the Effects of Topography and a Computer Graphics Display of Traffic Flow," Calspan Report No. VJ-2251-V-3, March 1969.

3. McHenry, R. R. and DeLeys, N. J., "Automobile Dynamics - A Computer Simulation of Three-Dimensional Motions for Use in Studies of Braking Systems and the Driving Task," Calspan Report No. VJ-2251-V-7, August 1970.

4. Kroll, C. V. and Roland, R. D., "A Preview-Predictor Model of Driver Behavior in Emergency Situations," Calspan Report No. VJ-2251-V-6, October 1970.

5. Young, R. D., et al, "Simulation of Vehicle Impact with the Texas Concrete Median Barrier - Volume 1: Test Comparisons and Parameter Study," Texas Transportation Institute Report No. 140-5, June 1972.

6. Weaver, G. D., et al, "Effect of Curb Geometry and Location on Vehicle Behavior," NCHRP Project 20-7, Texas Transportation Report No. RF845, October 1972.

FHWA
R&D