Development and Analysis of a Vertical Dynamic Railcar Model

by

Gregory Dale Buckner

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

APPROVED:

Dr. Robert H. Fries, Chairman

Dr. Charles E. Knight

Dr. Harry H. Robertshaw

November, 1987

Blacksburg, Virginia

Development and Analysis of a Vertical Dynamic Railcar Model

by

Gregory Dale Buckner

Dr. Robert H. Fries, Chairman

Mechanical Engineering

(ABSTRACT)

Severe railcar responses can result from crosslevel and vertical rail inputs. At low speeds rail joint excitation can coincide with the roll natural frequency of a vehicle. At high speeds, dynamic effects can cause high wheel loads and harsh ride for sensitive cargos. Computer simulation of these and other vertical dynamic effects cans assist in design selections of vehicle components and diagnosis of troublesome vehicle responses.

Many dynamic models available today lack the complexity to analyze accurately some of the important dynamic effects. In this report a 28-degree-of-freedom railcar model has been developed to analyze the vertical dynamic responses of railcars subjected to random and deterministic track inputs. This model features carbody vertical bending and torsional modes, multiple component trucks and suspensions, and rail irregularity inputs at each of the eight wheels.

Simulation results for a 100-ton vehicle operating on harmonic track inputs compare favorably with the AAR Flexible Carbody Model. Other simulations on random track evaluate the influence of auxiliary viscous stabilizers and increased payloads on railcar responses. These simulations demonstrate the effectiveness of the computer simulation as a design and analysis tool.

Acknowledgements

I would like to thank Dr. Robert H. Fries for his guidance and patience as my research advisor. Dr. Fries provided the time and insight that made this research possible. I would also like to thank Dr. Charles E. Knight and Dr. Harry H. Robertshaw for serving on my graduate committee, and Dr. Charles F. Reinholz for serving on my examining committee.

Several of my fellow classmates provided invaluable assistance during my studies at Virginia Tech. Among these, I would like to thank Brian West and Ed Moas for teaching me everything I know about the computer facilities at Tech.

I would like to thank my parents for all the support (motivational and financial) they provided during these months of poverty and despair. To Janice Judy and her family, I extend my most sincere thanks for making my first year in Virginia one of the best of my life.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Railcar response to crosslevel inputs can cause severe carbody rocking at low speeds when the rail input frequencies coincide with the railcar's natural frequencies. This phenomenon, called rock and roll, typically occurs in fully loaded cars operating at speeds in the range of 10 to 20 kilometers per hour. It can result in wheel lift, suspension spring bottoming, and wheel/rail contact forces that exceed 2.5 times their nominal static value. Recent trends toward the use of heavier carloads and cars with higher centers of gravity have exacerbated the problem. Railcar response to vertical inputs can cause lading damage, high wheel/rail contact forces, and extensive wear in suspension elements.

The analysis of railcar vertical and roll responses is not new; various mathematical models have been developed over the years to study these responses. The complexity of these dynamic models varies drastically. Reduced-complexity models developed by Sankar, et al. (1987), Ahlbeck (1977), and Platin, et al. (1976) consider only half-carbody dynamic responses. More complete models, such as the full-carbody models of White (1986) and Willis, et al. (1976), neglect carbody flexibility.

The Association of American Railroads (AAR) Flexible Carbody Model, developed by Tse (1974) and Hussain, et al. (1980), represents the carbody as two separate masses joined by bending and torsional springs. Many of these models, including the AAR Flexible Carbody Model, use simplified harmonic rail inputs that do not adequately simulate actual operating conditions. Many models do not incorporate individual truck components, such as secondary springs and friction dampers.

The ultimate goals of vertical and roll dynamic models are to realistically simulate railcar responses to vertical and crosslevel inputs and to evaluate the effectiveness of suspension designs and damping mechanisms in reducing these responses. To this end, a dynamic model should possess enough freedoms and features to adequately represent the dynamic system; these features should include full-carbody flexibility, wheel lift capabilities, multiple-component trucks, and the capability to introduce random track inputs.

The objective of this work was to develop a dynamic railcar model with continuous carbody flexibility and detailed truck component representation. The model has 28 degrees of freedom. Additionally, eight rail inputs provide excitation at each wheel. Inputs are vertical alignment and crosslevel displacements, either discrete or random. Partially linearized equations of motion help reduce run costs and still preserve accuracy of the simulation. The model has been evaluated over a variety of run conditions and compared to the response results of the AAR model. Additionally, the effects of auxiliary suspension damping and increased carloads have been investigated.

Chapter 2

Model Description

This chapter introduces the 28-degree-of-freedom vertical dynamic model. The model is shown in Figure 1 on page 4. Descriptions of the model's features and limitations are presented in the next three sections.

2.1 General Description

The carbody of Figure 1 is represented by a continuous beam with multiple degrees of freedom. These carbody freedoms include vertical and lateral translations, pitch and roll rotations, and first-mode vertical bending and torsional displacements. The non-rigid body modes provide a realistic and continuous representation of a carbody's freedoms, in contrast to the simpler half-car model of Sankar et al. (1987), and the double-carbody model of Hussain et al. (1980).

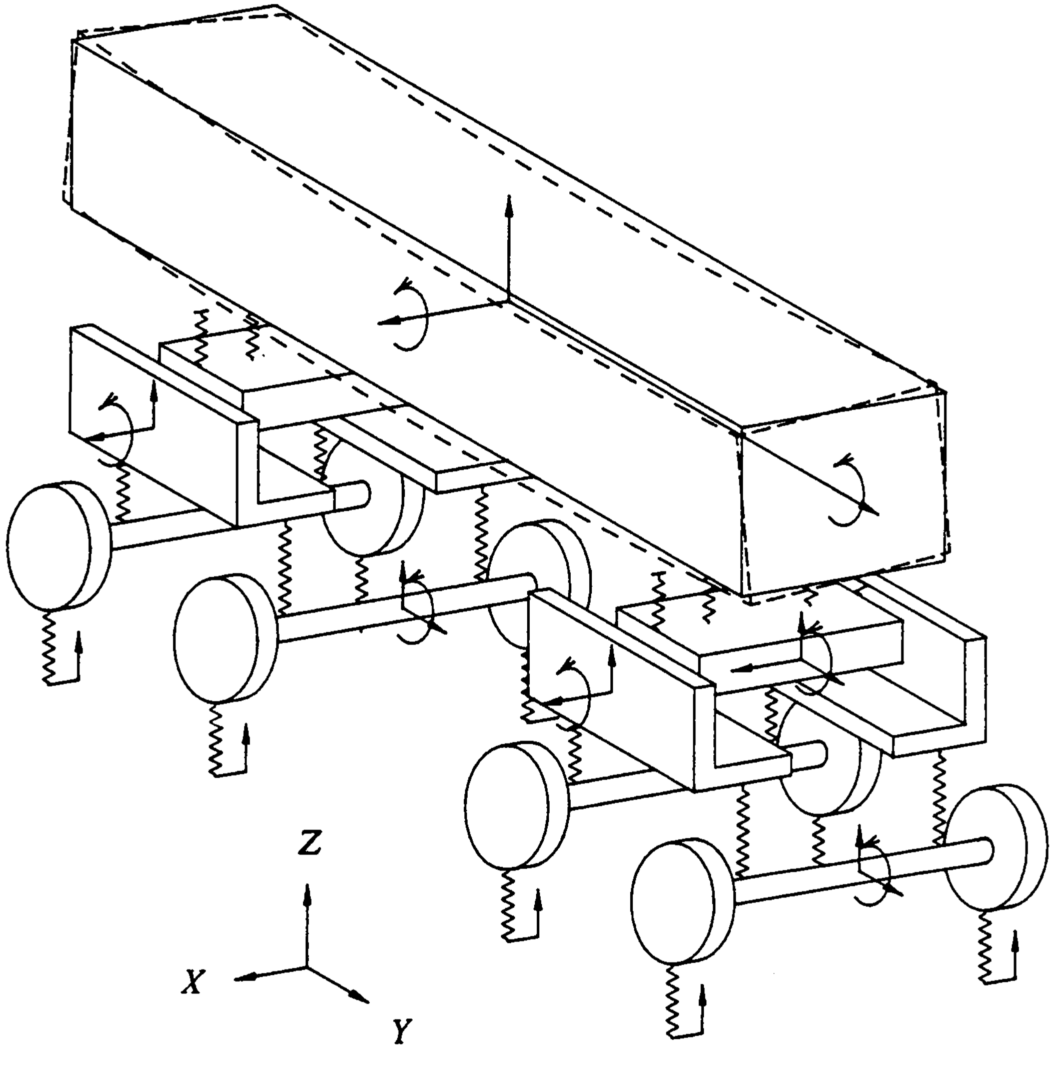

Figure 1. The 28-Degree-of-freedom Railcar Model

The carbody is supported by a centerplate and two sidebearers at each bolster. The front and rear bolsters each possess vertical and lateral translational freedoms, as well as roll rotational freedoms. Secondary suspensions join each bolster to a right and left sideframe. Each sideframe is free to translate vertically and rotate (pitch) about the lateral $\underset{\sim}{X}$ axis. The sideframes are supported by four wheelsets, which translate vertically and rotate (roll) about the longitudinal $\underset{\sim}{Y}$ axes.

A complete list of all active degrees of freedom is compiled below in Table 1.

Table 1. Model Degrees of Freedom

Component	**Freedom**	**Total**
Carbody	Vertical	1
	Lateral	1
	Pitch	1
	Roll	1
	First Bending	1
	First Torsion	1
Bolsters	Vertical	2
	Lateral	2
	Roll	2
Sideframes	Vertical	4
	Pitch	4
Axles	Vertical	4
	Roll	4
Total Degrees		**28**

Corresponding to each degree of freedom is a pair of state variables. The variable names are descriptive of the corresponding vehicle motion. For example, Z_C represents the vertical (Z) translation of the carbody, and $\dot{Z}_C$ represents its time derivative. The 56 state variables are presented in Table 2. The tables include array designations that are used in the computer program described in Chapter 4.

2.2 Features

Throughout the model, interconnections between masses are represented by spring stiffnesses. At some locations, these stiffnesses have non-linear characteristics that permit separation of two bodies in contact. The centerplate and sidebearer stiffnesses, for example, are represented by clearance springs which allow the carbody to completely separate from its bolster supports. This feature is a realistic representation that allows the carbody to assume one of several different configurations which might occur during severe roll or derailment conditions. These configurations are illustrated in Figure 2 on page 9.

Clearance springs also permit separation at each wheel/rail contact point. This feature realistically represents a more common phenomenon, wheel lift, which can occur at even moderate operating conditions (Hussain, 1980).

The primary damping mode is dry friction damping, or Coulomb damping, that exists in the vertical and lateral suspension groups. This damping is a constant-magnitude force that opposes relative

Table 2. The State Variables

Y Array Number	Symbol	Description
Y(1)	Z_C	Carbody Vertical Translation
Y(2)	$\dot{Z}_C$	Carbody Vertical Velocity
Y(3)	X_C	Carbody Lateral Translation
Y(4)	$\dot{X}_C$	Carbody Lateral Velocity
Y(5)	Θ_{CP}	Carbody Pitch Rotation
Y(6)	$\dot{\Theta}_{CP}$	Carbody Pitch Velocity
Y(7)	Θ_{CR}	Carbody Roll Rotation
Y(8)	$\dot{\Theta}_{CR}$	Carbody Roll Velocity
Y(9)	Δ	Carbody Bending Deflection
Y(10)	$\dot{\Delta}$	Carbody Bending Velocity
Y(11)	Ψ	Carbody Torsional Deflection
Y(12)	$\dot{\Psi}$	Carbody Torsional Velocity
Y(13)	Z_{B1}	Lead Bolster Vertical Translation
Y(14)	$\dot{Z}_{B1}$	Lead Bolster Vertical Velocity
Y(15)	Z_{B2}	Trailing Bolster Vertical Translation
Y(16)	$\dot{Z}_{B2}$	Trailing Bolster Vertical Velocity
Y(17)	X_{B1}	Lead Bolster Lateral Translation
Y(18)	$\dot{X}_{B1}$	Lead Bolster Lateral Velocity
Y(19)	X_{B2}	Trailing Bolster Lateral Translation
Y(20)	$\dot{X}_{B2}$	Trailing Bolster Lateral Velocity
Y(21)	B_1	Lead Bolster Roll Rotation
Y(22)	$\dot{B}_1$	Lead Bolster Roll Velocity
Y(23)	B_2	Trailing Bolster Roll Rotation
Y(24)	$\dot{B}_2$	Trailing Bolster Roll Velocity
Y(25)	Z_{SR1}	Lead Right Sideframe Vertical Translation
Y(26)	$\dot{Z}_{SR1}$	Lead Right Sideframe Vertical Velocity
Y(27)	Z_{SL1}	Lead Left Sideframe Vertical Translation
Y(28)	$\dot{Z}_{SL1}$	Lead Left Sideframe Vertical Velocity

Table 2. The State Variables (Continued)

Y Array Number	Symbol	Description
Y(29)	Z_{SR2}	Trailing Right Sideframe Vertical Translation
Y(30)	$\dot{Z}_{SR2}$	Trailing Right Sideframe Vertical Velocity
Y(31)	Z_{SL2}	Trailing Left Sideframe Vertical Translation
Y(32)	$\dot{Z}_{SL2}$	Trailing Left Sideframe Vertical Velocity
Y(33)	Σ_{SR1}	Lead Right Sideframe Pitch Rotation
Y(34)	$\dot{\Sigma}_{SR1}$	Lead Right Sideframe Pitch Velocity
Y(35)	Σ_{SL1}	Lead Left Sideframe Pitch Rotation
Y(36)	$\dot{\Sigma}_{SL1}$	Lead Left Sideframe Pitch Velocity
Y(37)	Σ_{SR2}	Trailing Right Sideframe Pitch Rotation
Y(38)	$\dot{\Sigma}_{SR2}$	Trailing Right Sideframe Pitch Velocity
Y(39)	Σ_{SL2}	Trailing Left Sideframe Pitch Rotation
Y(40)	$\dot{\Sigma}_{SL2}$	Trailing Left Sideframe Pitch Velocity
Y(41)	Z_{WF1}	Lead Front Wheelset Vertical Translation
Y(42)	$\dot{Z}_{WF1}$	Lead Front Wheelset Vertical Velocity
Y(43)	Z_{WR1}	Lead Rear Wheelset Vertical Translation
Y(44)	$\dot{Z}_{WR1}$	Lead Rear Wheelset Vertical Velocity
Y(45)	Z_{WF2}	Trailing Front Wheelset Vertical Translation
Y(46)	$\dot{Z}_{WF2}$	Trailing Front Wheelset Vertical Velocity
Y(47)	Z_{WR2}	Trailing Rear Wheelset Vertical Translation
Y(48)	$\dot{Z}_{WR2}$	Trailing Rear Wheelset Vertical Velocity
Y(49)	A_{F1}	Lead Front Wheelset Roll Rotation
Y(50)	$\dot{A}_{F1}$	Lead Front Wheelset Roll Velocity
Y(51)	A_{R1}	Lead Rear Wheelset Roll Rotation
Y(52)	$\dot{A}_{R1}$	Lead Rear Wheelset Roll Velocity
Y(53)	A_{F2}	Trailing Front Wheelset Roll Rotation
Y(54)	$\dot{A}_{F2}$	Trailing Front Wheelset Roll Velocity
Y(55)	A_{R2}	Trailing Rear Wheelset Roll Rotation
Y(56)	$\dot{A}_{R2}$	Trailing Rear Wheelset Roll Velocity

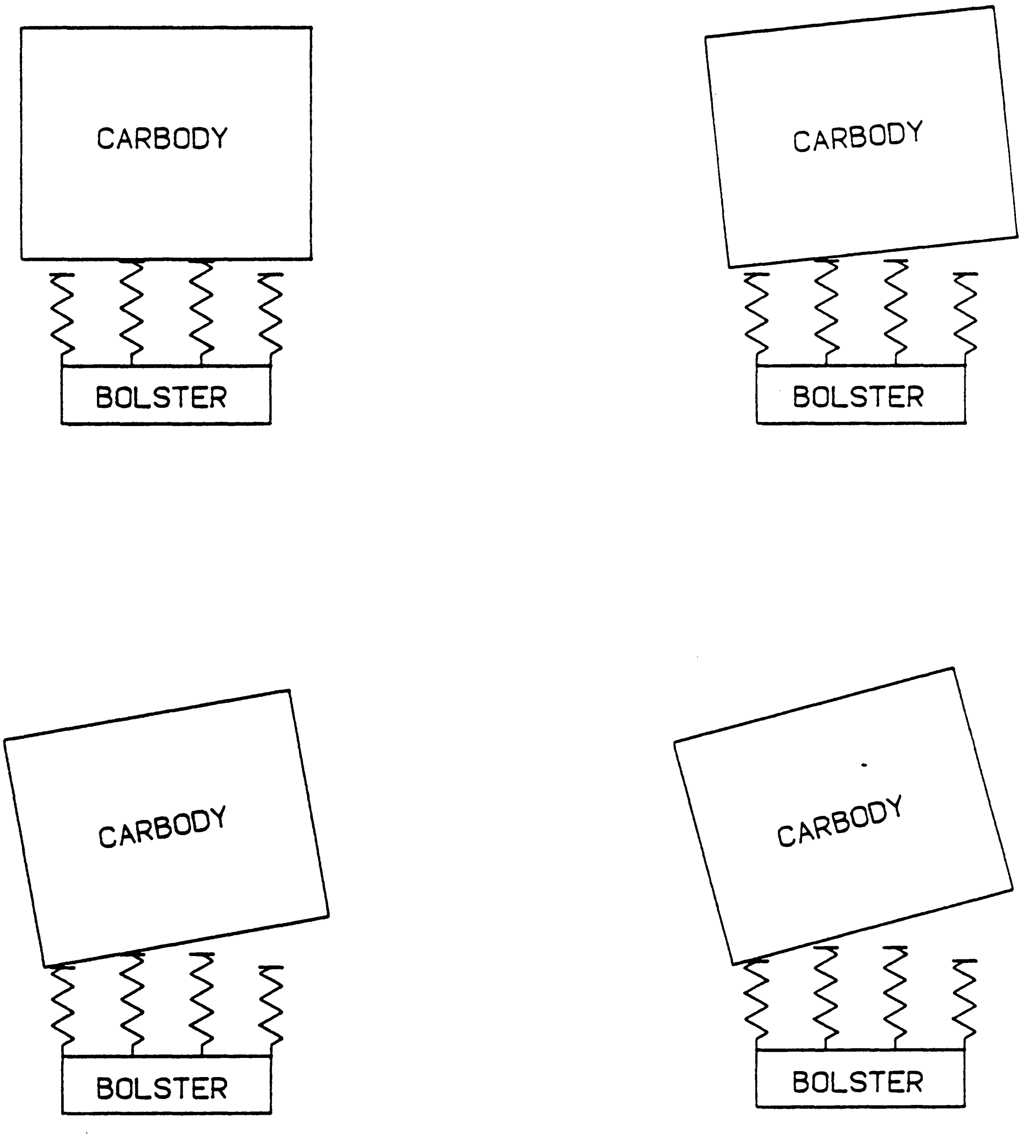

Figure 2. The Possible Carbody Roll Configurations

motions between the bolsters and sideframes. These forces result from contact pressure between the friction shoes and bolsters. Viscous damping can be added to the vertical suspension groups in the form of Stucki stabilizers. These dampers are compression-only devices that drastically reduce system responses at critical speeds (Stucki, 1985). Properties of these bi-linear devices are provided in Chapter 3.

Track inputs for the dynamic model are applied at each of the eight wheel/rail contact points. These inputs are random vertical and crosslevel irregularities generated in four separate subroutines. The random rail input subroutines, developed by Fries (1987), provide realistic excitation to the dynamic system, in contrast to the harmonic inputs frequently used (Hussain 1980). In addition to the random excitation, other discrete forms can be generated within the input routines.

2.3 Assumptions and Limitations

Inherent in any dynamic model are assumptions and limitations that simplify the analysis. For this dynamic model, several such restrictions exist. These are listed and briefly described below:

- Small Angles - roll and pitch angles throughout the model remain small enough that the following trigonometric approximations are valid:

$$\sin(\alpha) \cong \alpha \qquad \cos(\alpha) \cong 1 \qquad [2.3.1]$$

- Continuous Carbody - the carbody is modeled as a homogenous beam with appropriate inertial and physical properties.

- Tangent Tracks - this model simulates runs on tangent tracks only.
- Constant Velocity - vehicle velocity does not change during analysis.

Chapter 3

Derivation of Equations of Motion

3.1 Introduction

This chapter presents the derivation of motion equations for the twenty-eight degree-of-freedom railcar model. Section 3.2 introduces Lagrange's Equations, from which the equations will be derived. Section 3.3 details the formulation of kinetic energy expressions, which are utilized in Lagrange's Equations. Similarly, Section 3.4 documents the complete potential energy expressions. Section 3.5 presents the Rayleigh dissipation function, which includes all linear damping features of the system. The generalized input forces are included in Section 3.6. These forces account for the Coulomb damping forces of the suspension. The complete equations of motion are listed in Section 3.7.

3.2 Lagrange's Equations

The equations of motion for the twenty-eight degree-of-freedom model shown in Figure 1 on page 4 will be derived using a Lagrangian method. This method requires evaluation of the kinetic and potential energy expressions, the Rayleigh dissipation function, and the generalized forces. Once these expressions have been obtained, the equations of motion follow from Lagrange's Equations:

$$\frac{d}{dt}\left[\frac{\partial L}{\partial \dot{q}_i}\right]-\left[\frac{\partial L}{\partial q_i}\right]+\left[\frac{\partial D}{\partial \dot{q}_i}\right]=[F_i] \qquad [3.2.1]$$

where:

$L = T - V$

$T =$ kinetic energy expression

$V =$ potential energy expression

$D =$ Raleigh dissipation function

$F_i =$ generalized input force

$q_i = i^{th}$ generalized coordinate

$\dot{q}_i =$ first time derivative of i^{th} coordinate

3.3 *Kinetic Energy Expressions*

The derivation of a total kinetic energy expression begins with the carbody, which is significantly more complex than other model components. The carbody is modeled with six degrees of freedom. The kinetic energy expression for the carbody is complicated by the considerations of bending deflection in the vertical plane and torsional rotation in the lateral plane. To simplify the formulation of a total energy expression, the carbody is modeled as a homogenous beam.

3.3.1 Carbody Kinetic Energy Expression

According to the notation of Figure 3 on page 15 and Figure 4 on page 16, the carbody kinetic energy expression will be formulated using the vector $\underset{\sim}{R}$, which locates the mass center of each differential beam element. The carbody kinetic energy is derived from the following relation:

$$T_{CAR} = \frac{\xi}{2} \times \int_{\left(\frac{-W_{CAR}}{2}\right)}^{\left(\frac{W_{CAR}}{2}\right)} \int_{\left(\frac{-L_{CAR}}{2}\right)}^{\left(\frac{L_{CAR}}{2}\right)} \dot{\underset{\sim}{R}} \cdot \dot{\underset{\sim}{R}} \, dY \, dX \qquad [3.3.1]$$

In this equation, ξ represents the mass per unit area of the carbody. From the figures, the $\underset{\sim}{R}$ expression is found to be:

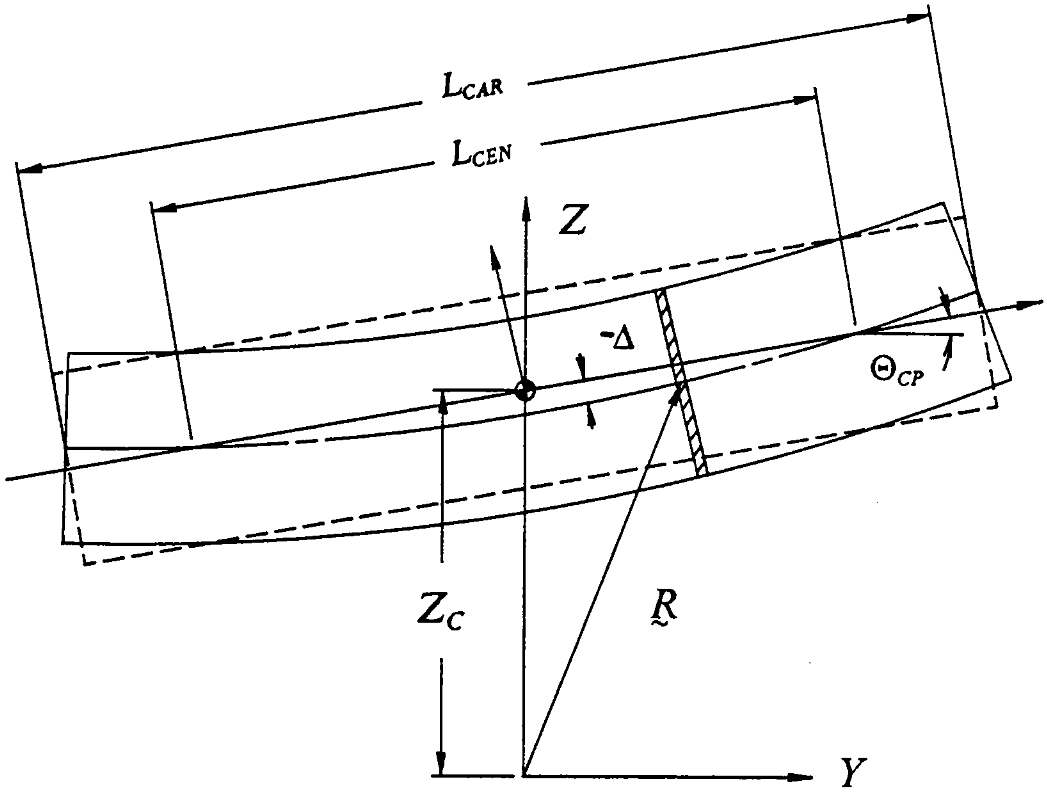

Figure 3. The Carbody Sideview

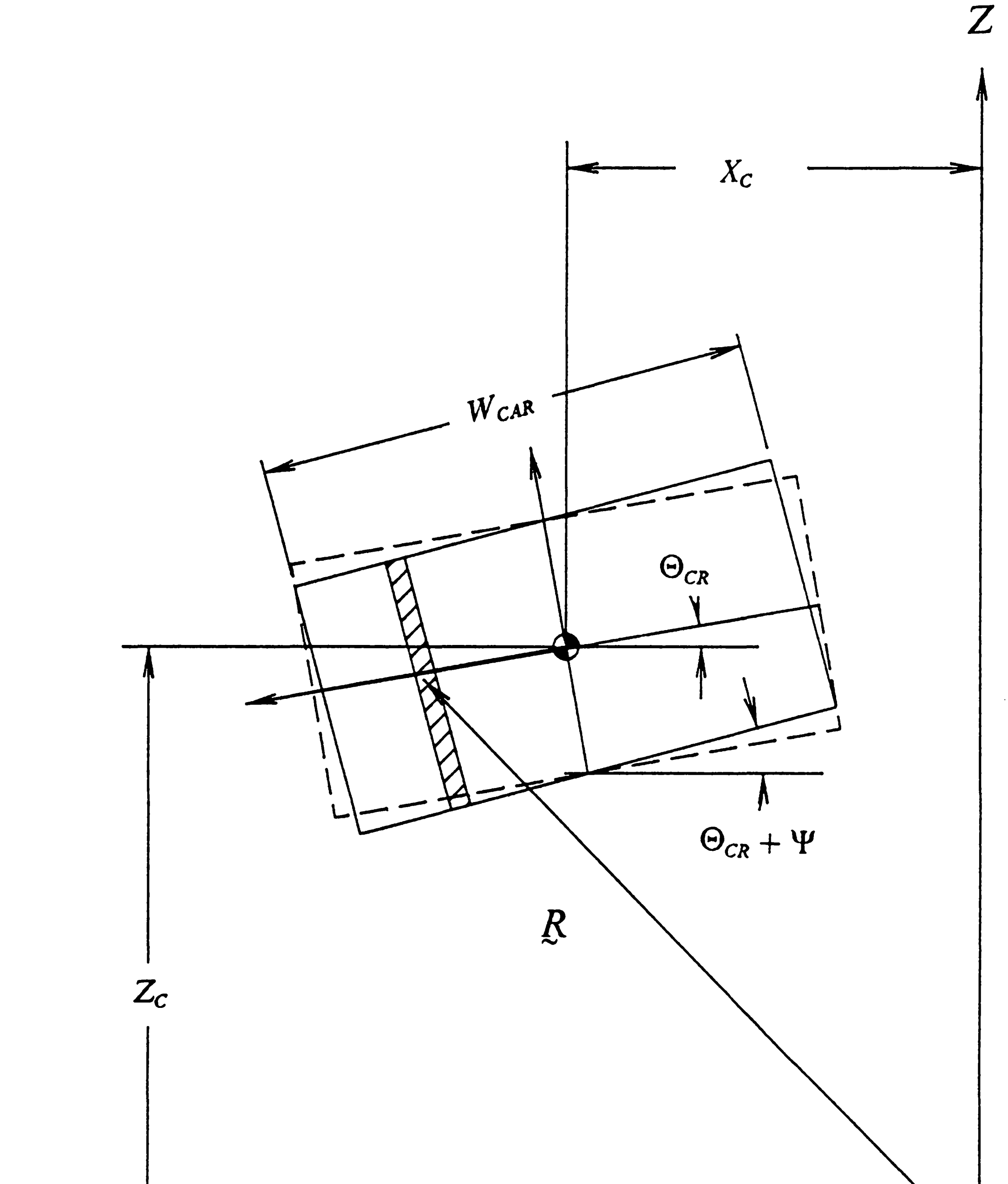

Figure 4. The Carbody Frontview

$$\underset{\sim}{R} = [X_C + (X\cos(\Theta_{CR} + \Psi))]\hat{i} + [(Y\cos\Theta_{CP}) - (\Delta\sin\Theta_{CP})]\hat{j} + [Z_C + (Y\sin\Theta_{CP}) + (\Delta\cos\Theta_{CP}) - (X\sin(\Theta_{CR} + \Psi))]\hat{k} \qquad [3.3.2]$$

Taking the time derivative of this expression, and noting that $\dot{X}$ and $\dot{Y} = 0$ (because the carbody is inextensible) yields:

$$\begin{aligned}\dot{\underset{\sim}{R}} = {} & [\dot{X}_C - X\sin(\Theta_{CR} + \Psi)(\dot{\Theta}_{CR} + \dot{\Psi})]\hat{i} \\ & + [(-Y\sin\Theta_{CP}\dot{\Theta}_{CP}) - (\dot{\Delta}\sin\Theta_{CP}) - (\Delta\cos\Theta_{CP}\dot{\Theta}_{CP})]\hat{j} \\ & + [\dot{Z}_C + (Y\cos\Theta_{CP}\dot{\Theta}_{CP}) + (\dot{\Delta}\cos\Theta_{CP}) - (\Delta\sin\Theta_{CP}\dot{\Theta}_{CP}) - X\cos(\Theta_{CR} + \Psi)(\dot{\Theta}_{CR} + \dot{\Psi})]\hat{k}\end{aligned} \qquad [3.3.3]$$

If the small angle assumption ($\cos\alpha \cong 1, \sin\alpha \cong \alpha$) is applied, this equation reduces to:

$$\begin{aligned}\dot{\underset{\sim}{R}} = {} & [\dot{X}_C - X(\Theta_{CR} + \Psi)(\dot{\Theta}_{CR} + \dot{\Psi})]\hat{i} \\ & + [(-Y\Theta_{CP}\dot{\Theta}_{CP}) - (\dot{\Delta}\Theta_{CP}) - (\Delta\dot{\Theta}_{CP})]\hat{j} \\ & + [\dot{Z}_C + (Y\dot{\Theta}_{CP}) + \dot{\Delta} - (\Delta\Theta_{CP}\dot{\Theta}_{CP}) - (X(\dot{\Theta}_{CR} + \dot{\Psi}))]\hat{k}\end{aligned} \qquad [3.3.4]$$

Next, taking the dot product of $\dot{\underset{\sim}{R}}$ with itself yields:

$$\begin{aligned}\dot{\underset{\sim}{R}} \bullet \dot{\underset{\sim}{R}} = {} & [\dot{X}_C^2 - (2X\dot{X}_C(\Theta_{CR} + \Psi)(\dot{\Theta}_{CR} + \dot{\Psi})) + (X^2(\Theta_{CR} + \Psi)^2(\dot{\Theta}_{CR} + \dot{\Psi})^2) \\ & + (Y^2\Theta_{CP}^2\dot{\Theta}_{CP}^2) + (\dot{\Delta}^2\Theta_{CP}^2) + (\Delta^2\dot{\Theta}_{CP}^2) + (2Y\dot{\Delta}\Theta_{CP}^2\dot{\Theta}_{CP}) + \dot{Z}_C^2 \\ & + (Y^2\dot{\Theta}_{CP}^2) + \dot{\Delta}^2 + (\Delta^2\Theta_{CP}^2\dot{\Theta}_{CP}^2) + (X^2(\dot{\Theta}_{CR} + \dot{\Psi})^2) + (2Y\dot{Z}_C\dot{\Theta}_{CP}) \\ & + (2\dot{Z}_C\dot{\Delta}) - (2\dot{Z}_C\Delta\Theta_{CP}\dot{\Theta}_{CP}) - (2\dot{Z}_C X(\dot{\Theta}_{CR} + \dot{\Psi})) + (2Y\dot{\Delta}\dot{\Theta}_{CP}) \\ & - (2Y\dot{\Theta}_{CP}X(\dot{\Theta}_{CR} + \dot{\Psi})) - (2\dot{\Delta}X(\dot{\Theta}_{CR} + \dot{\Psi})) + (2\Delta\Theta_{CP}\dot{\Theta}_{CP}X(\dot{\Theta}_{CR} + \dot{\Psi}))]\end{aligned} \qquad [3.3.5]$$

Now that an expression for the $\dot{\underset{\sim}{R}} \bullet \dot{\underset{\sim}{R}}$ product has been obtained, the vertical carbody kinetic energy expression of Equation 3.3.1 can be solved. Evaluation of this integral requires functional expressions for the deflection $\Delta(Y, t)$ and the torsion $\Psi(Y, t)$. A Ritz approximation for an n^{th} mode shape deflection expression (Lovejoy, 1987) has the form:

$$\Delta(Y, t) = \sum_{i=1}^{n} \phi_i(Y) q_i(t) \tag{3.3.6}$$

In this equation, $\phi_i(Y)$ is the i^{th} mode shape function for the beam and $q_i(t)$ is the i^{th} time-dependent portion. Thomson (1981) provides a shape function that can be normalized and reduced to first-mode form. For the simply-supported end conditions:

$$\phi_1(Y) = \cos(\frac{\pi Y}{L_{CEN}}) \tag{3.3.7}$$

Where L_{CEN} is the distance between centerplates shown in Figure 3 on page 15. For this model, L_{CEN} will be αL_{CAR}, where α is a fraction of L_{CAR}. Making this substitution yields the first-mode deflection expression:

$$\Delta(Y,t) = \cos(\frac{\pi Y}{\alpha L_{CAR}}) q(t) \tag{3.3.8}$$

A similar functional approximation can be developed for the torsional variable $\Psi(Y, t)$:

$$\Psi(Y, t) = \sum_{i=1}^{n} \zeta_i(Y) r_i(t) \tag{3.3.9}$$

$\zeta_i(Y)$ represents the i^{th} torsional mode shape along the longitudinal axis of the carbody, and $r_i(t)$ is the i^{th} time-dependent portion. Cook and Young (1985) provide a linear first-mode function of the form:

$$\zeta_1(Y) = \frac{Y}{\alpha L_{CAR}} \qquad [3.3.10]$$

Hence, a complete first-mode representation for the torsional carbody rotation takes the form:

$$\Psi(Y, t) = \frac{Y r(t)}{\alpha L_{CAR}} \qquad [3.3.11]$$

The carbody kinetic energy expression can now be evaluated:

$$T_{CAR} = \frac{\xi}{2} \times [I_1 + I_2 + I_3 + I_4 + I_5 + I_6 + I_7 + I_8 + I_9 + I_{10} + I_{11} + I_{12} + I_{13} + I_{14} + I_{15} + I_{16} + I_{17} + I_{18} + I_{19} + I_{20}] \qquad [3.3.12]$$

where:

$$I_1 = \iint \dot{X}_C^2 \, dY \, dX = \left[W_{CAR} L_{CAR} \dot{X}_C^2 \right]$$

$$I_2 = \iint -2X\dot{X}_C(\Theta_{CR} + \frac{Yr}{\alpha L_{CAR}})(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})\, dY\, dX = 0$$

$$I_3 = \iint X^2(\Theta_{CR} + \frac{Yr}{\alpha L_{CAR}})^2(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})^2\, dY\, dX = \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR}^2 \dot{\Theta}_{CR}^2}{12}\right]$$
$$+ \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR}^2 \dot{r}^2}{144\alpha^2}\right] + \left[\frac{W_{CAR}^3 L_{CAR} \dot{\Theta}_{CR}^2 r^2}{144\alpha^2}\right] + \left[\frac{W_{CAR}^3 L_{CAR} r^2 \dot{r}^2}{960\alpha^4}\right] + \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR} \dot{\Theta}_{CR} r\dot{r}}{36\alpha^2}\right]$$

$$I_4 = \iint Y^2 \Theta_{CP}^2 \dot{\Theta}_{CP}^2\, dY\, dX = \left[\frac{W_{CAR} L_{CAR}^3 \Theta_{CP}^2 \dot{\Theta}_{CP}^2}{12}\right]$$

$$I_5 = \iint \dot{q}^2 \cos^2(\frac{\pi Y}{\alpha L_{CAR}}) \Theta_{CP}^2\, dY\, dX = \left[.4118 W_{CAR} L_{CAR} \dot{q}^2 \Theta_{CP}^2\right]$$

$$I_6 = \iint q^2 \cos^2(\frac{\pi Y}{\alpha L_{CAR}}) \dot{\Theta}_{CP}^2\, dY\, dX = \left[.4118 W_{CAR} L_{CAR} q^2 \dot{\Theta}_{CP}^2\right]$$

$$I_7 = \iint 2Y\dot{q} \cos(\frac{\pi Y}{\alpha L_{CAR}}) \Theta_{CP}^2 \dot{\Theta}_{CP}\, dY\, dX = 0$$

$$I_8 = \iint \dot{Z}_C^2\, dY\, dX = \left[W_{CAR} L_{CAR} \dot{Z}_C^2\right]$$

$$I_9 = \iint Y^2 \dot{\Theta}_{CP}^2 \, dY \, dX = \left[\frac{W_{CAR} L_{CAR}^3 \dot{\Theta}_{CP}^2}{12} \right]$$

$$I_{10} = \iint \dot{q}^2 \cos^2(\frac{\pi Y}{\alpha L_{CAR}}) \, dY \, dX = \left[.4118 W_{CAR} L_{CAR} \dot{q}^2 \right]$$

$$I_{11} = \iint q^2 \cos^2(\frac{\pi Y}{\alpha L_{CAR}}) \Theta_{CP}^2 \dot{\Theta}_{CP}^2 \, dY \, dX = \left[.4118 W_{CAR} L_{CAR} q^2 \Theta_{CP}^2 \dot{\Theta}_{CP}^2 \right]$$

$$I_{12} = \iint X^2 (\dot{\Theta}_{CR} + \frac{Y \dot{r}}{\alpha L_{CAR}})^2 \, dY \, dX = \left[\frac{W_{CAR}^3 L_{CAR} \dot{\Theta}_{CR}^2}{12} \right] + \left[\frac{W_{CAR}^3 L_{CAR} \dot{r}^2}{144 \alpha^2} \right]$$

$$I_{13} = \iint 2 Y \dot{Z}_C \dot{\Theta}_{CP} \, dY \, dX = 0$$

$$I_{14} = \iint 2 \dot{Z}_C \dot{q} \cos(\frac{\pi Y}{\alpha L_{CAR}}) \, dY \, dX = \left[.9529 W_{CAR} L_{CAR} \dot{Z}_C \dot{q} \right]$$

$$I_{15} = \iint -2 \dot{Z}_C q \cos(\frac{\pi Y}{\alpha L_{CAR}}) \Theta_{CP} \dot{\Theta}_{CP} \, dY \, dX = \left[-.9529 W_{CAR} L_{CAR} \dot{Z}_C q \Theta_{CP} \dot{\Theta}_{CP} \right]$$

$$I_{16} = \iint -2\dot{Z}_C X(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})\,dY\,dX = 0$$

$$I_{17} = \iint 2Y\dot{q}\cos(\frac{\pi Y}{\alpha L_{CAR}})\dot{\Theta}_{CP}\,dY\,dX = 0$$

$$I_{18} = \iint -2Y\dot{\Theta}_{CP}X(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})\,dY\,dX = 0$$

$$I_{19} = \iint -2\dot{q}\cos(\frac{\pi Y}{\alpha L_{CAR}})X(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})\,dY\,dX = 0$$

$$I_{20} = \iint 2q\cos(\frac{\pi Y}{\alpha L_{CAR}})\Theta_{CP}\dot{\Theta}_{CP}X(\dot{\Theta}_{CR} + \frac{Y\dot{r}}{\alpha L_{CAR}})\,dY\,dX = 0$$

Finally, the vertical and torsional carbody kinetic energy expression for the twenty-eight degree-of-freedom model is complete:

$$
\begin{aligned}
T_{CAR} = \frac{\xi}{2} \times \{ & \left[W_{CAR}L_{CAR}\dot{X}_C^2\right] + \left[\frac{W_{CAR}^3 L_{CAR}\Theta_{CR}^2\dot{\Theta}_{CR}^2}{12}\right] + \left[\frac{W_{CAR}^3 L_{CAR}\Theta_{CR}^2\dot{r}^2}{144\alpha^2}\right] \\
& + \left[\frac{W_{CAR}^3 L_{CAR}\dot{\Theta}_{CR}^2 r^2}{144\alpha^2}\right] + \left[\frac{W_{CAR}^3 L_{CAR} r^2\dot{r}^2}{960\alpha^4}\right] + \left[\frac{W_{CAR}^3 L_{CAR}\Theta_{CR}\dot{\Theta}_{CR} r\dot{r}}{36\alpha^2}\right] \\
& + \left[\frac{W_{CAR}L_{CAR}^3\Theta_{CP}^2\dot{\Theta}_{CP}^2}{12}\right] + \left[.4118 W_{CAR}L_{CAR}\dot{q}^2\Theta_{CP}^2\right] + \left[.4118 W_{CAR}L_{CAR}q^2\dot{\Theta}_{CP}^2\right] \\
& + \left[W_{CAR}L_{CAR}\dot{Z}_C^2\right] + \left[\frac{W_{CAR}L_{CAR}^3\dot{\Theta}_{CP}^2}{12}\right] + \left[.4118 W_{CAR}L_{CAR}\dot{q}^2\right] \\
& + \left[.4118 W_{CAR}L_{CAR}q^2\Theta_{CP}^2\dot{\Theta}_{CP}^2\right] + \left[\frac{W_{CAR}^3 L_{CAR}\dot{\Theta}_{CR}^2}{12}\right] + \left[\frac{W_{CAR}^3 L_{CAR}\dot{r}^2}{144\alpha^2}\right] \\
& + \left[.9529 W_{CAR}L_{CAR}\dot{Z}_C\dot{q}\right] + \left[-.9529 W_{CAR}L_{CAR}\dot{Z}_C q\Theta_{CP}\dot{\Theta}_{CP}\right]\}
\end{aligned}
\qquad [3.3.13]
$$

3.3.2 Other Components of Kinetic Energy

The kinetic energy expressions for the remaining components are much simpler than the preceding carbody kinetic energy expression. This occurs because the remaining components each possess only two or three rigid-body modes of freedom, and simpler energy expressions result.

The bolsters each possess three rigid-body modes of motion: vertical translation, lateral translation, and roll rotation. Each bolster makes the following contribution to the total kinetic energy expression:

$$
T_{BOLSTER} = \frac{1}{2}M_B\dot{Z}_B^2 + \frac{1}{2}M_B\dot{X}_B^2 + \frac{1}{2}I_B\dot{\mathrm{B}}^2 \qquad [3.3.14]
$$

The sideframes each possess two rigid-body modes of motion: vertical translation and pitch rotation. Each sideframe makes the following contribution to the total kinetic energy expression:

$$T_{SIDEFRAME}=\frac{1}{2}M_S\dot{Z}_S^2+\frac{1}{2}I_S\dot{\Sigma}^2 \qquad [3.3.15]$$

The only remaining contributions to the kinetic energy expression come from each of the four wheelsets, which possess rigid-body freedoms of vertical translation and roll rotation. Each wheelset contributes the following to the total kinetic energy expression:

$$T_{WHEELSET}=\frac{1}{2}M_W\dot{Z}_W^2+\frac{1}{2}I_W\dot{\mathrm{A}}^2 \qquad [3.3.16]$$

3.3.3 Total Kinetic Energy Expression

The total kinetic energy expression is obtained by summing the contributions of Equations 3.3.13, 3.3.14, 3.3.15, and 3.3.16. The kinetic energy expressions obtained in Section 3.3.2 apply to more than one respective components, and are therefore used more than once.

$$
\begin{aligned}
T = \frac{\xi}{2} \times \{ & \left[W_{CAR} L_{CAR} \dot{X}_C^2 \right] + \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR}^2 \dot{\Theta}_{CR}^2}{12} \right] + \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR}^2 \dot{r}^2}{144\alpha^2} \right] \\
& + \left[\frac{W_{CAR}^3 L_{CAR} \dot{\Theta}_{CR}^2 r^2}{144\alpha^2} \right] + \left[\frac{W_{CAR}^3 L_{CAR} r^2 \dot{r}^2}{960\alpha^4} \right] + \left[\frac{W_{CAR}^3 L_{CAR} \Theta_{CR} \dot{\Theta}_{CR} r \dot{r}}{36\alpha^2} \right] \\
& + \left[\frac{W_{CAR} L_{CAR}^3 \Theta_{CP}^2 \dot{\Theta}_{CP}^2}{12} \right] + \left[.4118 W_{CAR} L_{CAR} \dot{q}^2 \Theta_{CP}^2 \right] + \left[.4118 W_{CAR} L_{CAR} q^2 \dot{\Theta}_{CP}^2 \right] \\
& + \left[W_{CAR} L_{CAR} \dot{Z}_C^2 \right] + \left[\frac{W_{CAR} L_{CAR}^3 \dot{\Theta}_{CP}^2}{12} \right] + \left[.4118 W_{CAR} L_{CAR} \dot{q}^2 \right] \\
& + \left[.4118 W_{CAR} L_{CAR} q^2 \Theta_{CP}^2 \dot{\Theta}_{CP}^2 \right] + \left[\frac{W_{CAR}^3 L_{CAR} \dot{\Theta}_{CR}^2}{12} \right] + \left[\frac{W_{CAR}^3 L_{CAR} \dot{r}^2}{144\alpha^2} \right] \\
& + \left[.9529 W_{CAR} L_{CAR} \dot{Z}_C \dot{q} \right] + \left[-.9529 W_{CAR} L_{CAR} \dot{Z}_C q \Theta_{CP} \dot{\Theta}_{CP} \right] \} \\
& + \frac{1}{2} M_B \left[\dot{Z}_{B1}^2 + \dot{Z}_{B2}^2 \right] + \frac{1}{2} M_B \left[\dot{X}_{B1}^2 + \dot{X}_{B2}^2 \right] + \frac{1}{2} I_B \left[\dot{\mathrm{B}}_1^2 + \dot{\mathrm{B}}_2^2 \right] \\
& + \frac{1}{2} M_S \left[\dot{Z}_{SR1}^2 + \dot{Z}_{SR2}^2 + \dot{Z}_{SL1}^2 + \dot{Z}_{SL2}^2 \right] + \frac{1}{2} I_S \left[\dot{\Sigma}_{R1}^2 + \dot{\Sigma}_{R2}^2 + \dot{\Sigma}_{L1}^2 + \dot{\Sigma}_{L2}^2 \right] \\
& + \frac{1}{2} M_W \left[\dot{Z}_{WF1}^2 + \dot{Z}_{WF2}^2 + \dot{Z}_{WR1}^2 + \dot{Z}_{WR2}^2 \right] + \frac{1}{2} I_W \left[\dot{\mathrm{A}}_{F1}^2 + \dot{\mathrm{A}}_{F2}^2 + \dot{\mathrm{A}}_{R1}^2 + \dot{\mathrm{A}}_{R2}^2 \right]
\end{aligned}
\qquad [3.3.17]
$$

3.4 Potential Energy Expressions

The total potential energy expression has contributions from several sources; strain energy effects in the carbody, spring potential energies, and gravitational energies are all important considerations. For convenience, each of these potential energies is presented separately. The analysis begins with the carbody.

3.4.1 Carbody Strain Energy Expressions

Because the carbody is free to deflect in the vertical plane and twist in the lateral plane, strain energy effects are an important consideration in the development of potential energy expressions. Cook and Young (1985) formulate these strain energies based on the fundamentals of mechanics. For the strain energy of deflection:

$$V_{BENDING} = \frac{EI}{2} \times \int_{\left(\frac{-L_{CAR}}{2}\right)}^{\left(\frac{L_{CAR}}{2}\right)} \left[\frac{\partial^2 \Delta(Y,t)}{\partial Y^2} \right]^2 dy \qquad [3.4.1]$$

From the Ritz deflection expression provided in Equation 3.3.8, this energy is:

$$V_{BENDING} = \left[\frac{.2060 E I q^2 \pi^4}{\alpha^4 L_{CAR}^3} \right] \qquad [3.4.2]$$

Similarly, for the torsion:

$$V_{TORSION} = \frac{G J_R}{2} \times \int_{\left(\frac{-L_{CAR}}{2}\right)}^{\left(\frac{L_{CAR}}{2}\right)} \left[\frac{\partial \Psi(Y,t)}{\partial Y} \right]^2 dy \qquad [3.4.3]$$

The torsional expression of Equation 3.3.10 will be used to evaluate this strain energy, but first the equivalent polar moment J_R must be determined. Although the carbody has been modeled as a homogenous beam, it is desirable for the beam to possess the same equivalent polar moment as the actual carbody. From Cook and Young (1985), J_R depends only on the geometry of the cross-section under consideration. For this reason, the cross-sectional properties of Figure 5 on page 28 will be used to evaluate J_R for the model.

J_R is calculated by:

$$J_R = \sum_{i=1}^{z} \frac{b_i t_i^3}{3} \qquad [3.4.4]$$

where:

b_i = the length of the i^{th} flange

t_i = the thickness of the i^{th} flange

z = the number of flanges in the cross-section

From the geometry of Figure 5 on page 28, and using typical carbody dimensions and bending stiffness, these calculations yield $J_R \cong .024\ m^4$. This value is also computed using an expression from Roark et al. (1975) with nearly the same result. Finally, the torsional strain energy is determined:

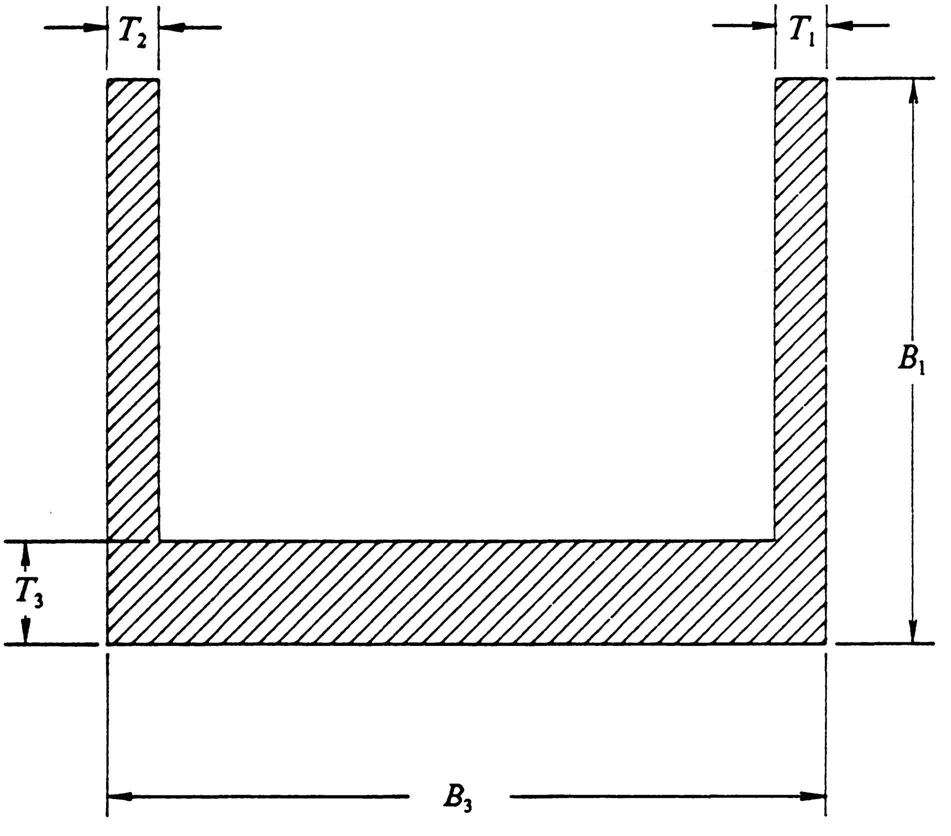

Figure 5. The Carbody Cross-Section

$$V_{TORSION} = \left[\frac{G\, J_R r^2}{2\alpha^2 L_{CAR}} \right] \qquad [3.4.5]$$

Hence, the complete carbody strain energy expression is determined:

$$V_{CARBODY} = \left[\frac{.2060 E I q^2 \pi^4}{\alpha^4 L_{CAR}^3} \right] + \left[\frac{G\, J_R r^2}{2\alpha^2 L_{CAR}} \right] \qquad [3.4.6]$$

3.4.2 Spring Potential Energy Expressions

Stiffnesses throughout the model are simulated by various types of springs, which are either linear or non-linear. Clearance springs are non-linear "compression-only" springs that will not prevent separation of two bodies.

Regardless of the type, all spring potential energies are calculated by the formula:

$$V_{SPRING} = \frac{1}{2}\left[k_s \Delta_s^2 \right] \qquad [3.4.7]$$

where:

$k_s =$ the springs stiffness

$\Delta_s =$ the relative displacement across the spring

3.4.2.1 Linear Spring Potential Energy Expressions

Linear springs are implemented into the suspension stiffness, both laterally and vertically, as seen in the model spring detail of Figure 6 on page 31 and the suspension detail of Figure 7 on page 32. The determination of linear spring potential energies is relatively simple, as it requires only a calculation of the relative displacements Δ_r, which can be positive, indicating compression, or negative, indicating separation. Referring to Figure 7 on page 32, the vertical suspension potential corresponding to the linear spring K_{SV} follows:

$$V_{KSV} = \frac{1}{2}\left[K_{SV}\Delta^2_{SV}\right] \qquad [3.4.8]$$

where:

$$\Delta_{SV} = Z_S - (Z_B \pm (\frac{L_B}{2} \times \mathrm{B})) \qquad [3.4.9]$$

The plus and minus of the preceding equation indicates that the calculation of Δ_{SV} depends on the location of the suspension spring K_{SV}. Positive refers to springs on the right side and negative to springs on the left side, because positive roll rotations of the bolster produce negative vertical translations on the left side. A similar convention is used in the following calculations.

The lateral potential energies for the suspension springs K_{SL} are calculated by:

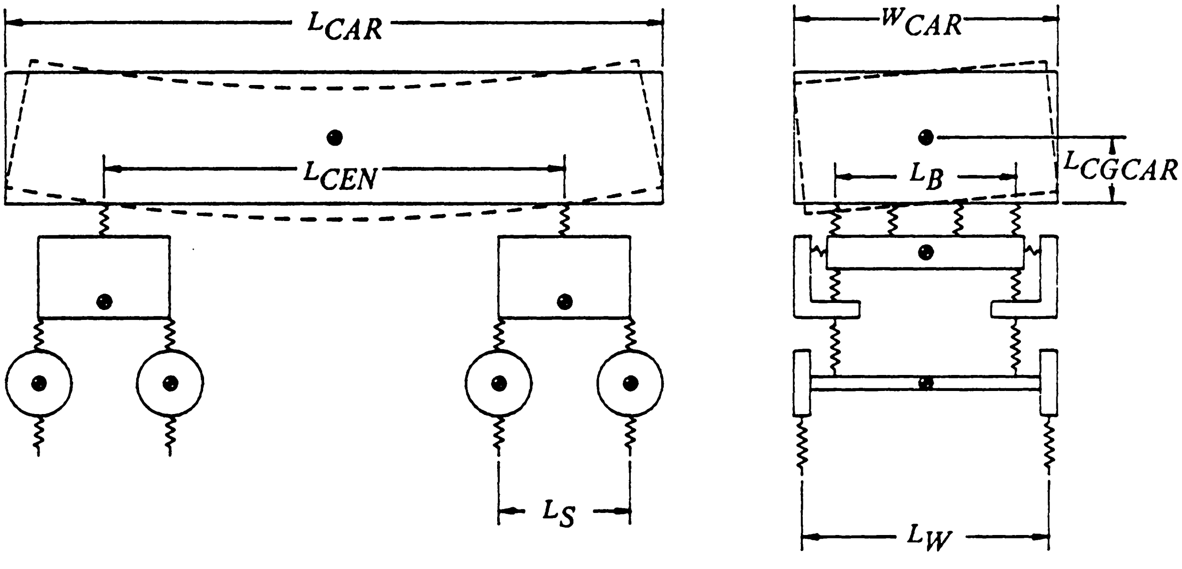

Figure 6. The Model Spring Groups

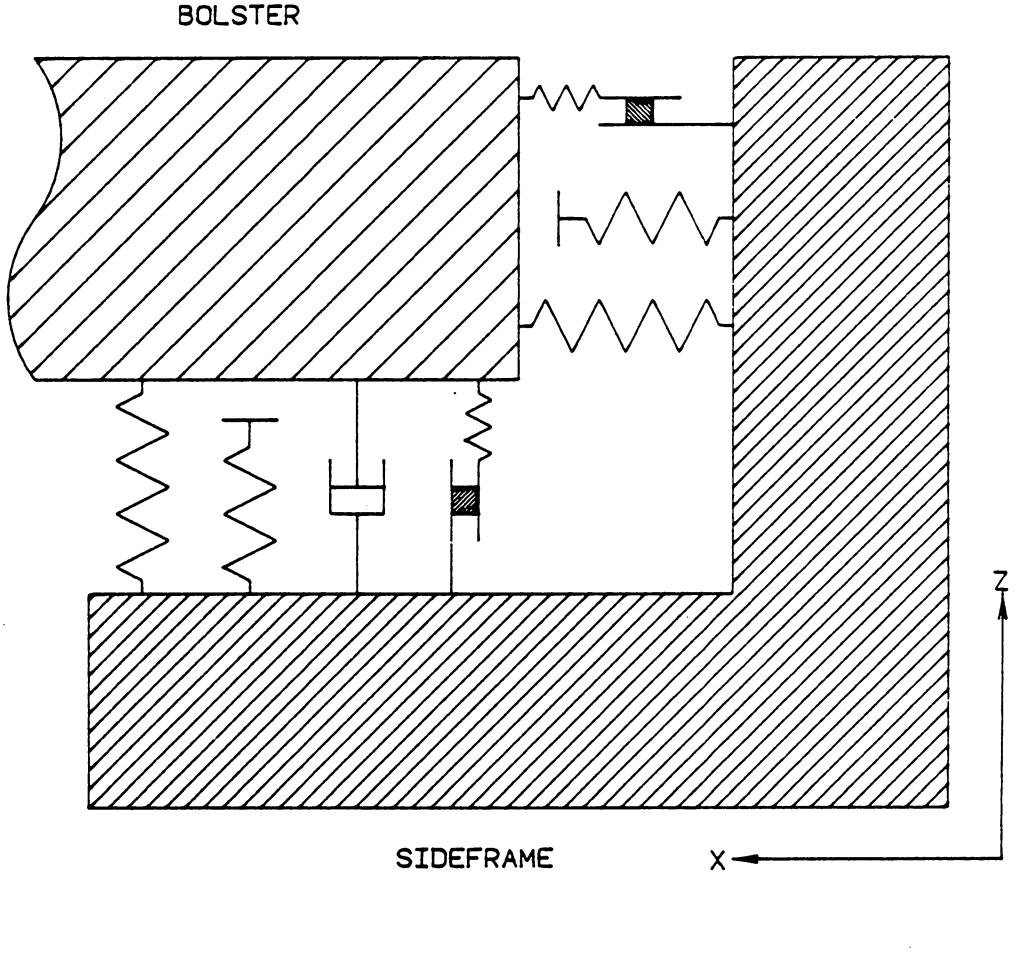

Figure 7. The Suspension Groups in Detail

$$V_{KSL} = \frac{1}{2}\left[K_{SL}\Delta_{SL}^2\right] \qquad [3.4.10]$$

where:

$$\Delta_{SL} = 0 - X_B \qquad [3.4.11]$$

(The sideframes do not possess lateral freedoms.)

Linear springs also occur at the sideframe/wheelset interfaces throughout the model. According to the spring group detail Figure 6 on page 31, the potential energies across these vertical springs, K_{SW}, are calculated by:

$$V_{KSW} = \frac{1}{2}\left[K_{SW}\Delta_{SW}^2\right] \qquad [3.4.12]$$

where:

$$\Delta_{SW} = (Z_W \pm (\frac{L_B}{2} \times \mathrm{A})) - (Z_S \pm (\frac{L_S}{2} \times \Sigma)) \qquad [3.4.13]$$

3.4.2.2 Clearance Spring Potential Energy Expressions

Clearance springs are an important feature of the railcar model. These springs are used frequently throughout the model to realistically simulate the separation and lift effects that exist in a railcar. By definition, clearance springs only permit the transmission of compressive forces. Thus, before the potential energy across a clearance spring can be calculated, the displacement across the spring must be evaluated to determine if contact exists. For contact to exist, the relative displacement across the spring Δ, must be greater than or equal to the clearance distance associated with the spring, C_S.

Again referring to the notation of Figure 7 on page 32, notice that clearance springs have been implemented into the suspension spring groups (both vertical and lateral) to represent spring bottoming. Clearance springs also model the centerplate stiffnesses (both vertical and lateral) to allow separation of the carbody from the bolsters (which occurs during derailment). The sidebearing springs of Figure 6 on page 31 are springs with significant clearance. They contact only during severe rolling action. The rail stiffness has been modeled as a clearance spring that allows wheel lift to occur.

The vertical centerplate stiffnesses are modeled by clearance springs of zero clearance. Their associated potential energies are calculated as follows:

$$V_{CEN\,V} = \frac{1}{2}\left[K_{CENV}\Delta_{CENV}^2\right] \qquad [3.4.14]$$

where:

$$\Delta_{CENV} = (Z_B \pm (\frac{D_{CEN}}{2} \times \text{B})) - (Z_C \pm (\frac{L_{CEN}}{2} \times \Theta_{CP}) \pm (\frac{D_{CEN}}{2} \times (\Theta_{CR} \pm \frac{\Psi}{2})))$$ [3.4.15]

The centerplate potential energy expression above is valid only if the relative displacement Δ_{CENV} is positive. Otherwise, there is no contact between the centerplate and the bolster, and the potential is zero.

The calculation of Δ_{CENV} assumes that vertical contributions from the pitch and roll rotations are directly additive. Technically, angular rotations are not vectors and cannot be summed in a vector sense. However, because the small angle assumption has been applied and will be verified in the output, the error associated with this assumption is negligible.

The lateral centerplate potential energies are similarly calculated:

$$V_{CEN\,L} = \frac{1}{2}\left[K_{CENL}\Delta^2_{CENL}\right]$$ [3.4.16]

where:

$$\Delta_{CENL} = (X_B + (L_{CGB} \times \text{B})) - (X_C - (L_{CGCAR} \times (\Theta_{CR} \pm \frac{\Psi}{2})))$$ [3.4.17]

The lateral potential energies associated with the centerplate stiffnesses are valid only if contact between the centerplate and bolster exists as determined from the Δ_{CENV} calculation. If contact does not exist, the lateral potential is also zero.

The sidebearing potential energies of Figure 6 on page 31 are calculated in an identical manner:

$$V_{SIDE} = \frac{1}{2}\left[K_{SIDE}\Delta^2_{SIDE}\right] \qquad [3.4.18]$$

where:

$$\Delta_{SIDE} = (Z_B \pm (\frac{L_{CB}}{2} \times \text{B})) - (Z_C \pm (\frac{L_{CAR}}{2} \times \Theta_{CP}) \pm (\frac{L_{CB}}{2} \times (\Theta_{CR} \pm \frac{\Psi}{2}))) \qquad [3.4.19]$$

The sidebearing spring potential energies exist only if Δ_{SIDE} is greater than the sidebearing clearance C_{SIDE} .

The clearance potential energies in the suspension are easily calculated. First, for the vertical suspension clearance springs K_{SCV}:

$$V_{SCV} = \frac{1}{2}\left[K_{SCV}\Delta^2_{SCV}\right] \qquad [3.4.20]$$

In this case, $\Delta_{SCV} = \Delta_{SV}$ calculated in Equation 3.4.9. Here, Δ_{SV} must exceed the vertical suspension clearance C_{SV}, which represents bottoming of the vertical suspension springs.

Similarly, the lateral suspension clearance potential energies are determined:

$$V_{SCL} = \frac{1}{2}\left[K_{SCL}\Delta_{SCL}^2\right] \qquad [3.4.21]$$

Again, $\Delta_{SCL} = \Delta_{SL}$ calculated in Equation 3.4.13. Δ_{SV} must exceed the lateral suspension clearance C_{SL} for the potential energy to exist.

Clearance springs are used on each of the eight rail inputs to simulate the rail/roadbed stiffness at these points. Hence, the wheel and rail are free to separate once the interface contact force becomes zero. For this reason, the K_{RAIL} clearance springs have zero clearance. The potential energies associated with these springs are calculated:

$$V_{RAIL} = \frac{1}{2}\left[K_{RAIL}\Delta_{RAIL}^2\right] \qquad [3.4.22]$$

where:

$$\Delta_{RAIL} = Z_{RAIL} - (Z_W \pm (\frac{L_W}{2} \times A)) \qquad [3.4.23]$$

3.4.3 Gravitational Potential Energy Expressions

Those components of the railcar model that translate in the vertical direction are directly influenced by the effects of gravity. The calculation of gravitational potential energy expressions follows from:

$$V_{GRAVITY} = m\, g\, \Delta_Z \qquad [3.4.26]$$

where:

m = mass of component

g = gravitational constant $\equiv 9.81 \frac{m}{s^2}$

Δ_Z = relative vertical displacement of component

Because the calculation of these potential energies is straightforward, the gravitational energy expressions associated with each model component will simply be listed. From Equation 3.4.26:

$$V_{G\,CARBODY} = M_C\, g\, Z_C \qquad [3.4.27]$$

$$V_{G\,BOLSTER} = M_B\, g\, Z_B \qquad [3.4.28]$$

$$V_{G\,SIDEFRAME} = M_S\, g\, Z_S \qquad [3.4.29]$$

$$V_{G\,WHEELSET} = M_W\, g\, Z_W \qquad [3.4.30]$$

3.4.4 Total Potential Energy Expression

The potential energy expressions of Equations 3.4.1 - 3.4.30 consist of carbody strain energies, spring potential energies, and gravitational energies throughout the model. Each of these potential energies contributes directly to the total potential energy equation. Each form of potential energy may occur in several different locations throughout the model. For example, the sideframe/wheelset spring energy computed in Equation 3.4.8 occurs in eight different locations of the model, as seen in Figure 1 on page 4. The total potential energy expression follows:

$$
\begin{aligned}
V = & \left[\frac{.2060 E I q^2 \pi^4}{\alpha^4 L_{CAR}^3}\right] + \left[\frac{G J_R r^2}{2\alpha^2 L_{CAR}}\right] + \frac{1}{2}\left[K_{SV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1)))^2\right] + \frac{1}{2}\left[K_{SV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1)))^2\right] \\
& + \frac{1}{2}\left[K_{SV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2)))^2\right] + \frac{1}{2}\left[K_{SV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2)))^2\right] \\
& + \frac{1}{2}\left[K_{SL}(-X_{B1})^2\right] + \frac{1}{2}\left[K_{SL}(-X_{B1})^2\right] + \frac{1}{2}\left[K_{SL}(-X_{B2})^2\right] + \frac{1}{2}\left[K_{SL}(-X_{B2})^2\right] \\
& + \frac{1}{2}\left[K_{SW}((Z_{WF1} + (\frac{L_B}{2} \times A_{F1})) - (Z_{SR1} + (\frac{L_S}{2} \times \Sigma_{R1})))^2\right] + \frac{1}{2}\left[K_{SW}((Z_{WF1} - (\frac{L_B}{2} \times A_{F1})) - (Z_{SL1} + (\frac{L_S}{2} \times \Sigma_{L1})))^2\right] \\
& + \frac{1}{2}\left[K_{SW}((Z_{WR1} + (\frac{L_B}{2} \times A_{R1})) - (Z_{SR1} - (\frac{L_S}{2} \times \Sigma_{R1})))^2\right] + \frac{1}{2}\left[K_{SW}((Z_{WR1} - (\frac{L_B}{2} \times A_{R1})) - (Z_{SL1} - (\frac{L_S}{2} \times \Sigma_{L1})))^2\right] \\
& + \frac{1}{2}\left[K_{SW}((Z_{WF2} + (\frac{L_B}{2} \times A_{F2})) - (Z_{SR2} + (\frac{L_S}{2} \times \Sigma_{R2})))^2\right] + \frac{1}{2}\left[K_{SW}((Z_{WF2} - (\frac{L_B}{2} \times A_{F2})) - (Z_{SL2} + (\frac{L_S}{2} \times \Sigma_{L2})))^2\right] \\
& + \frac{1}{2}\left[K_{SW}((Z_{WR2} + (\frac{L_B}{2} \times A_{R2})) - (Z_{SR2} - (\frac{L_S}{2} \times \Sigma_{R2})))^2\right] + \frac{1}{2}\left[K_{SW}((Z_{WR2} - (\frac{L_B}{2} \times A_{R2})) - (Z_{SL2} - (\frac{L_S}{2} \times \Sigma_{L2})))^2\right] \\
& + \frac{1}{2}\left[K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1)) - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1)) - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2)) - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2)) - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \qquad [3.4.31] \\
& + \frac{1}{2}\left[K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1)) - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_{CB}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1)) - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_{CB}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2)) - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_{CB}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2)) - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_{CB}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))^2\right] \\
& + \frac{1}{2}\left[K_{SCV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1)))^2\right] + \frac{1}{2}\left[K_{SCV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1)))^2\right] \\
& + \frac{1}{2}\left[K_{SCV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2)))^2\right] + \frac{1}{2}\left[K_{SCV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2)))^2\right] \\
& + \frac{1}{2}\left[K_{SCL}(-X_{B1})^2\right] + \frac{1}{2}\left[K_{SCL}(-X_{B1})^2\right] + \frac{1}{2}\left[K_{SCL}(-X_{B2})^2\right] + \frac{1}{2}\left[K_{SCL}(-X_{B2})^2\right] \\
& + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,FR1} - (Z_{WF1} + (\frac{L_W}{2} \times A_{F1})))^2\right] + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,FL1} - (Z_{WF1} - (\frac{L_W}{2} \times A_{F1})))^2\right] \\
& + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,RR1} - (Z_{WR1} + (\frac{L_W}{2} \times A_{R1})))^2\right] + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,RL1} - (Z_{WR1} - (\frac{L_W}{2} \times A_{R1})))^2\right] \\
& + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,FR2} - (Z_{WF2} + (\frac{L_W}{2} \times A_{F2})))^2\right] + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,FL2} - (Z_{WF2} - (\frac{L_W}{2} \times A_{F2})))^2\right] \\
& + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,RR2} - (Z_{WR2} + (\frac{L_W}{2} \times A_{R2})))^2\right] + \frac{1}{2}\left[K_{RAIL}(Z_{RAIL\,RL2} - (Z_{WR2} - (\frac{L_W}{2} \times A_{R2})))^2\right] \\
& + M_C g Z_C + M_B g (Z_{B1} + Z_{B2}) + M_S g (Z_{SR1} + Z_{SL1} + Z_{SR2} + Z_{SL2}) + M_W g (Z_{WF1} + Z_{WR1} + Z_{WF2} + Z_{WR2})
\end{aligned}
$$

3.5 Raleigh Dissipation Function

The Raleigh dissipation function is an expression that accounts for all linear damping effects in the model. Damping exists in various forms throughout the model, including carbody damping, viscous damping in the suspension, and dry friction damping, or Coulomb damping, in the suspension. Because Coulomb damping is a non-linear generalized force, it is considered in Section 3.6. For convenience, each of the linear damping sources will be discussed separately.

Regardless of the type, all linear damping effects are accounted for in Lagrange's equations by the Rayleigh dissipation function, whose form is similar to the spring potential energy expression of Equation 3.4.4:

$$D = \frac{1}{2}\left[C_d \dot{\Delta}_d^2\right] \qquad [3.5.1]$$

where:

C_d = the damping coefficient

$\dot{\Delta}_d$ = the relative velocity across the damping element

3.5.1 Carbody Damping

Because the carbody is modeled as a beam with deflection and torsional freedoms, it is necessary to implement forms of carbody damping that will inhibit steady-state oscillations.

3.5.1.1 Carbody Deflection Damping

Carbody deflection damping is represented by C_Δ. The magnitude of the damping coefficient, C_Δ, will be determined for an arbitrary damping ratio of $\zeta_c = 0.1$. From Thomson (1981), this damping coefficient is calculated by:

$$C = 2\zeta_c\sqrt{KM} \qquad [3.5.2]$$

where:

M = the mass expression from the equations of motion

K = the stiffness expression from the equations of motion

In the railcar equations of motion, M and K are the coefficients of the $\ddot{q}$ and q variables, respectively. From these expressions, the deflection damping coefficient is directly calculated from Equation 3.5.2. For typical values of carbody mass, length, and bending stiffness, the deflection damping coefficient is approximately:

$$C_\Delta \cong 125{,}000 \frac{N \times s}{m} \qquad [3.5.3]$$

3.5.1.2 Carbody Torsional Damping

Carbody torsional damping is represented by C_Ψ. As before, M and K from Equation 3.5.2 represent the coefficients of the $\ddot{r}$ and r variables, respectively. For typical carbody parameters, the torsional damping coefficient is found to be:

$$C_\Psi \cong 300{,}000 \frac{N \times s}{\surd} \qquad [3.5.4]$$

3.5.1.3 Total Carbody Damping

Equations 3.5.3 and 3.5.4 provide the total carbody damping expression:

$$D_{CAR} = \frac{1}{2}\left[C_\Delta \dot{q}^2\right] + \frac{1}{2}\left[C_\Psi \dot{r}^2\right] \qquad [3.5.5]$$

3.5.2 Auxiliary Viscous Damping

Viscous damping is a linear form that may exist in the suspension. Some railcar vehicles are equipped with viscous dampers, called Stucki stabilizers, for enhanced damping characteristics.

These dampers are "compression-only" devices; they provide damping only when the relative velocities between the bolsters and sideframes indicate approach. Stucki stabilizers have bi-linear damping characteristics, as indicated in Figure 8 on page 45, Stucki (1985).

From the figure, the viscous damping coefficient C_{SV} depends on the relative velocity across the device. Once this coefficient is determined, the Rayleigh dissipation expression for these viscous dampers is calculated by:

$$D_{SV} = \frac{1}{2}\left[C_{SV}\dot{\Delta}_{SV}^2\right] \qquad [3.5.6]$$

where:

$$\dot{\Delta}_{SV} = \dot{Z}_S - (\dot{Z}_B \pm (\frac{L_B}{2} \times \dot{\mathrm{B}})) \qquad [3.5.7]$$

3.5.3 Track Damping

Linear damping in the track and subgrade is considered at each input. The coefficient C_{RAIL} accounts for all track damping contributions. The Rayleigh dissipation expressions at each rail input is calculated by:

$$D_{RAIL} = \frac{1}{2}\left[C_{RAIL}\dot{\Delta}_{RAIL}^2\right] \qquad [3.5.7]$$

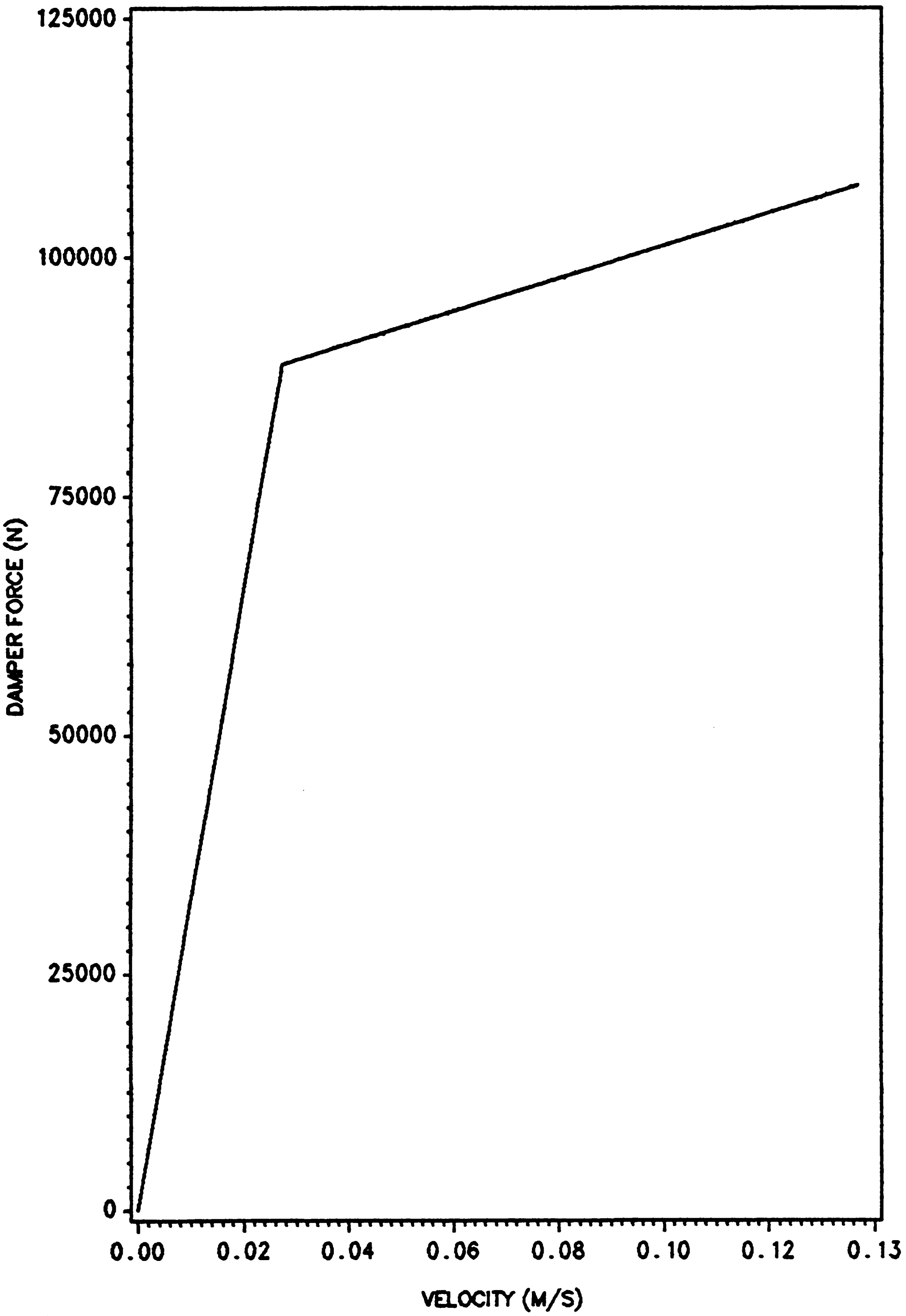

Figure 8. The Stucki Stabilizer Characteristics

where:

$$\dot{\Delta}_{RAIL} = \dot{Z}_{RAIL} - (\dot{Z}_W \pm (\frac{L_W}{2} \times \dot{A})) \quad [3.5.8]$$

3.5.4 Total Rayleigh Dissipation Expression

Damping contributions from the carbody and viscous suspension dampers have been considered in the preceding sections. A total formulation that accounts for all sources of linear damping in the model will now be presented. This expression is the total Rayleigh dissipation function:

$$\begin{aligned}
D = &\frac{1}{2}\left[C_\Delta \dot{q}^2\right] + \frac{1}{2}\left[C_\Psi \dot{r}^2\right] \\
&+ \frac{1}{2}\left[C_{SV}(\dot{Z}_{SR1} - (\dot{Z}_{B1} + (\frac{L_B}{2} \times \dot{B}_1)))^2\right] + \frac{1}{2}\left[C_{SV}(\dot{Z}_{SL1} - (\dot{Z}_{B1} - (\frac{L_B}{2} \times \dot{B}_1)))^2\right] \\
&+ \frac{1}{2}\left[C_{SV}(\dot{Z}_{SR2} - (\dot{Z}_{B2} + (\frac{L_B}{2} \times \dot{B}_2)))^2\right] + \frac{1}{2}\left[C_{SV}(\dot{Z}_{SL2} - (\dot{Z}_{B2} - (\frac{L_B}{2} \times \dot{B}_2)))^2\right] \\
&+ \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RFR1} - (\dot{Z}_{WF1} + (\frac{L_W}{2} \times \dot{A}_{F1})))^2\right] + \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RFL1} - (\dot{Z}_{WF1} - (\frac{L_W}{2} \times \dot{A}_{F1})))^2\right] \\
&+ \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RRR1} - (\dot{Z}_{WR1} + (\frac{L_W}{2} \times \dot{A}_{R1})))^2\right] + \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RRL1} - (\dot{Z}_{WR1} - (\frac{L_W}{2} \times \dot{A}_{R1})))^2\right] \\
&+ \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RFR2} - (\dot{Z}_{WF2} + (\frac{L_W}{2} \times \dot{A}_{F2})))^2\right] + \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RFL2} - (\dot{Z}_{WF2} - (\frac{L_W}{2} \times \dot{A}_{F2})))^2\right] \\
&+ \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RRR2} - (\dot{Z}_{WR2} + (\frac{L_W}{2} \times \dot{A}_{R2})))^2\right] + \frac{1}{2}\left[C_{RAIL}(\dot{Z}_{RRL2} - (\dot{Z}_{WR2} - (\frac{L_W}{2} \times \dot{A}_{R2})))^2\right]
\end{aligned} \quad [3.5.9]$$

3.6 Generalized Forces

Coulomb damping forces are non-linear generalized forces that occur at sliding contact points in the suspension. Specifically, these primary damping sources are a result of sliding contact between the sideframes and the friction shoes of the bolsters.

Although several friction models adequately represent Coulomb damping, the slider model adopted by Heller et al. (1977) has been selected for its numerical stability during solution. The typical Coulomb friction element consists of a linear spring, K_F in parallel with a friction element F_C, as shown in Figure 9 on page 48. The friction spring K_F allows calculation of the Coulomb friction force when it is below the breakaway point, F_C .

The general algorithm for determining the relative separation across the friction element Z_{SB} and the friction force in the element F_F at the i^{th} time step of a numerical solution follows:

1. Calculate the relative displacement between the bolster and sideframe:

$$D_i = Z_{S_i} - Z_{B_i} \qquad [3.6.1]$$

2. Calculate a temporary force across the element using the previous value of friction slip $Z_{SB_{i-1}}$:

$$F_{TEMP} = K_F \times (D_i + Z_{SB_{i-1}}) \qquad [3.6.2]$$

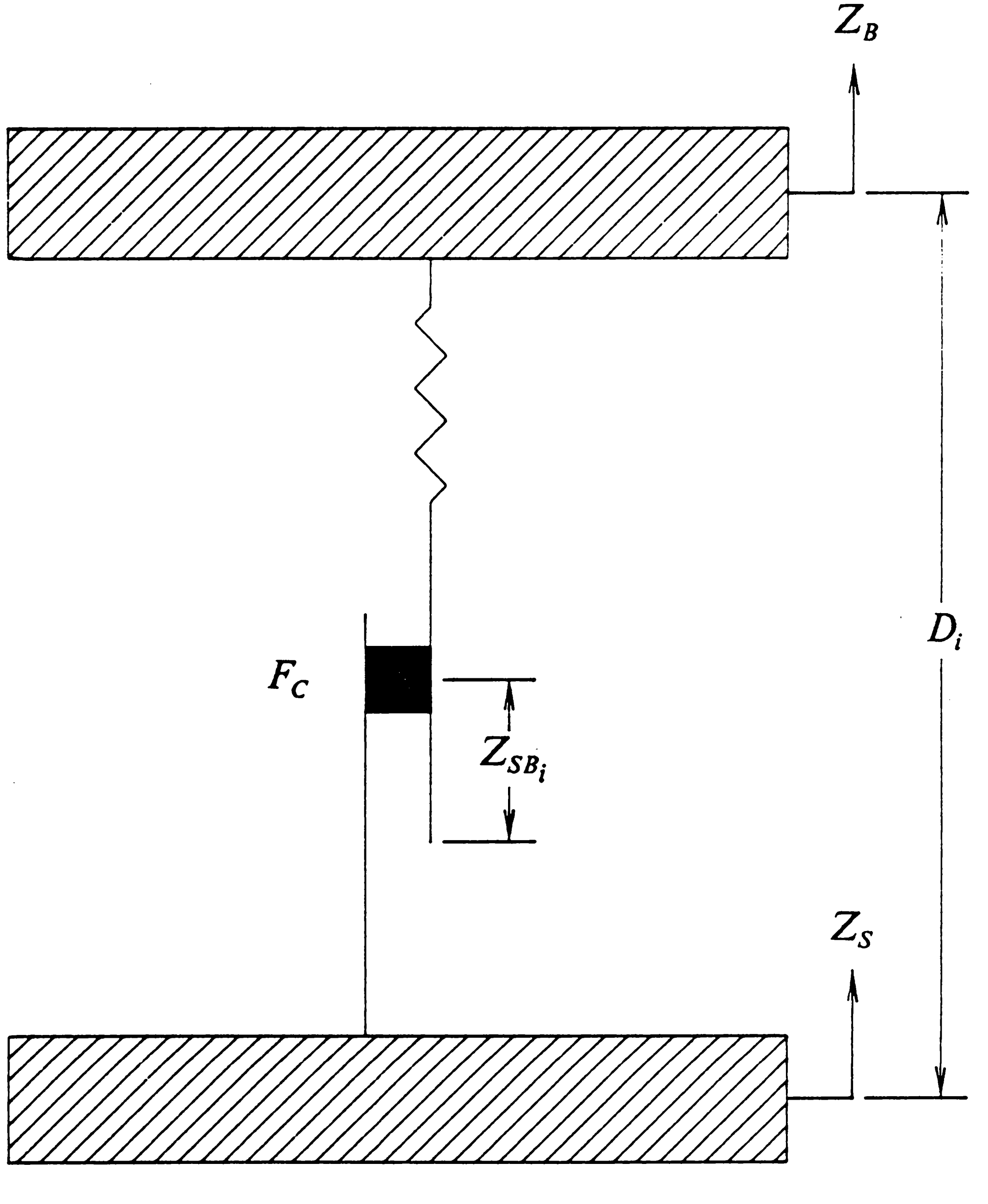

Figure 9. The Coulomb Friction Element

3. If the magnitude of this temporary force F_{TEMP} is less than the breakaway force F_C, then:

$$Z_{SB_i} = Z_{SB_{i-1}} \qquad [3.6.3]$$

4. If the magnitude of this temporary force F_{TEMP} is greater than (or equal to) the breakaway force F_C, then:

$$Z_{SB_i} = \left[\frac{sign(\dot{D}_i)F_C}{K_F}\right] - D_i \qquad [3.6.4]$$

5. Using this new value of Z_{SB_i}, the damping force across the Coulomb friction element is directly determined:

$$F_F = K_F(D_i + Z_{SB_i}) \qquad [3.6.5]$$

Similar friction forces are calculated for each Coulomb friction element in the model, whether vertical or lateral. A total of eight such forces must be calculated, 4 vertical and 4 lateral. These friction forces comprise the total generalized force expression:

$$\begin{aligned} F = & \left[K_{FV} \times (Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1)) + Z_{SBR1})\right] \\ & + \left[K_{FV} \times (Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1)) + Z_{SBL1})\right] \\ & + \left[K_{FV} \times (Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2)) + Z_{SBR2})\right] \\ & + \left[K_{FV} \times (Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2)) + Z_{SBL2})\right] \\ & + [K_{FL} \times (-X_{B1} + X_{SBR1})] + [K_{FL} \times (-X_{B1} + X_{SBL1})] \\ & + [K_{FL} \times (-X_{B2} + X_{SBR2})] + [K_{FL} \times (-X_{B2} + X_{SBL2})] \end{aligned} \qquad [3.6.6]$$

3.7 *Equations of Motion*

The kinetic energy, potential energy, dissipation energy, and generalized input forces in the model are required to formulate the equations of motion. This formulation utilizes Lagrange's Equations, which were introduced in Section 3.1. From Equation 3.1.1, the Lagrangian expression L is the difference in kinetic and potential energies: $L = T - V$, where T and V follow from Equations 3.3.17 and 3.4.31, respectively. The Rayleigh dissipation function, D, is presented in Equation 3.5.8. The Coulomb friction generalized input forces are presented in Equation 3.6.6. Using Lagrange's equations, the motion equations are derived starting with q_1, the carbody vertical translation coordinate Z_C.

Evaluating Lagrange's Equations with respect to each generalized coordinate q_i produces literally hundreds of high-order terms in the equations of motion. If small displacements are assumed, these high-order terms have negligible influence on the system's response. For this reason, only the first-order terms are retained in the equations of motion that follow. All product terms of second-order or higher are neglected in the final equations of motion.

NOTE The potential energy expression derived in Section 3.4 includes contributions from the sideframe/wheelset springs of Figure 1 on page 4. In subsequent modifications to the model, these springs have been replaced by kinematic constraints to eliminate high-frequency components. This replacement eliminates spring potential energy contributions from the sideframe/wheelset springs and changes the equations of motion presented below. These changes are reflected in the program listing of Appendix 1.

The equations that result from application of Lagrange's equations are presented in Equations 3.7.1 - 3.7.28.

$$
\begin{aligned}
\ddot{Z}_C = \frac{1}{M_C} \times \; & [\{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - M_C g]
\end{aligned}
\qquad [3.7.1]
$$

$$
\begin{aligned}
\ddot{X}_C = \frac{1}{M_C} \times [& \left[K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2})))) \right] \\
& + \left[K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2})))) \right] \\
& + \left[K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2})))) \right] \\
& + \left[K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2})))) \right]]
\end{aligned}
\qquad [3.7.2]
$$

$$
\begin{aligned}
\ddot{\Theta}_{CP} = \frac{L_{CEN}}{2I_{CP}} \times\ & [\{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& - \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}]
\end{aligned}
\qquad [3.7.3]
$$

$$\ddot{\Theta}_{CR} = \frac{L_B}{2I_{CR}} \times \begin{matrix}[\{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1)) \\ - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}\end{matrix}$$

$$+ \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2))$$
$$- (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}$$
$$- \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$- \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2))$$
$$- (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}$$
$$+ \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$+ \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2))$$
$$- (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \qquad [3.7.4]$$
$$- \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$- \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2))$$
$$- (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}]$$
$$- \frac{L_{CGCAR}}{I_{CR}} \times \begin{matrix}[\{K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}\end{matrix}$$
$$- \{K_{CENL}((X_{B1} + (L_{CGB} \times B_1))$$
$$- (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$- \{K_{CENL}((X_{B2} + (L_{CGB} \times B_2))$$
$$- (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}$$
$$- \{K_{CENL}((X_{B2} + (L_{CGB} \times B_2))$$
$$- (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}]$$

$$\ddot{q} = \frac{1}{.4118 M_C}[-.4765 M_C \ddot{Z}_C + \left[\frac{.4120 E I \pi^4}{\alpha^4 L_{CEN}^3}\right] q - 125000.D0\, \dot{q}] \qquad [3.7.5]$$

$$
\begin{aligned}
\ddot{r} = {} & \frac{12\alpha^2}{I_{Cr}} \times [\left[\frac{-GJ_R}{\alpha^2 L_{CEN}}\right] r - 300000\dot{r} \\
& + \{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times \mathrm{B}_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& - \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times \mathrm{B}_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times \mathrm{B}_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times \mathrm{B}_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times \mathrm{B}_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times \mathrm{B}_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times \mathrm{B}_1)) \\
& - (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times \mathrm{B}_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}] \\
& + \frac{L_{CGCAR}}{I_{CR}} \times \begin{matrix} [\{-K_{CENL}((X_{B1} + (L_{CGB} \times \mathrm{B}_1)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \end{matrix} \\
& - \{K_{CENL}((X_{B1} + (L_{CGB} \times \mathrm{B}_1)) \\
& - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
& + \{K_{CENL}((X_{B2} + (L_{CGB} \times \mathrm{B}_2)) \\
& - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENL}((X_{B2} + (L_{CGB} \times \mathrm{B}_2)) \\
& - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}]
\end{aligned}
\qquad [3.7.6]
$$

$$
\begin{aligned}
\ddot{Z}_{B1} = \frac{1}{M_B} \times [&\left[K_{SV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1))) \right] \\
&+ \left[K_{SCV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1))) \right] \\
&+ \left[C_S(\dot{Z}_{SR1} - (\dot{Z}_{B1} + (\frac{L_B}{2} \times \dot{B}_1))) \right] \\
&+ \left[K_{FV}((Z_{B1} + (\frac{L_B}{2} \times B_1)) + Z_{SBR1}) \right] \\
&+ \left[K_{SV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1))) \right] \\
&+ \left[K_{SCV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1))) \right] \\
&+ \left[C_S(\dot{Z}_{SL1} - (\dot{Z}_{B1} - (\frac{L_B}{2} \times \dot{B}_1))) \right] \\
&+ \left[K_{FV}((Z_{B1} - (\frac{L_B}{2} \times B_1)) + Z_{SBL1}) \right] \\
&- \{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1)) \\
&- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
&- \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1)) \\
&- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
&- \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1)) \\
&- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
&- \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1)) \\
&- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \\
&- M_B g]
\end{aligned}
\qquad [3.7.7]
$$

$$
\begin{aligned}
\ddot{Z}_{B2} = \frac{1}{M_B} \times [& \left[K_{SV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[K_{SCV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[C_S(\dot{Z}_{SR2} - (\dot{Z}_{B2} + (\frac{L_B}{2} \times \dot{B}_2))) \right] \\
& + \left[K_{FV}((Z_{B2} + (\frac{L_B}{2} \times B_2)) + Z_{SBR2}) \right] \\
& + \left[K_{SV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[K_{SCV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[C_S(\dot{Z}_{SL2} - (\dot{Z}_{B2} - (\frac{L_B}{2} \times \dot{B}_2))) \right] \\
& + \left[K_{FV}((Z_{B2} - (\frac{L_B}{2} \times B_2)) + Z_{SBL2}) \right] \\
& - \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - M_B g]
\end{aligned}
\qquad [3.7.8]
$$

$$
\begin{aligned}
\ddot{X}_{B1} = \frac{1}{M_B} \times [& \left[-K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2})))) \right] \\
& - \left[K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2})))) \right] \\
& + [K_{SL}(-X_{B1})] + [K_{SL}(-X_{B1})] \\
& + [K_{SCL}(-X_{B1})] + [K_{SCL}(-X_{B1})]] \\
& + [K_{FL}(-X_{B1} + X_{SBR1})] + [K_{FL}(-X_{B1} + X_{SBL1})]
\end{aligned}
\qquad [3.7.9]
$$

$$\ddot{X}_{B2} = \frac{1}{M_B} \times [\left[-K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2})))) \right]$$
$$- \left[K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2})))) \right] \quad [3.7.10]$$
$$+ [K_{SL}(-X_{B2})] + [K_{SL}(-X_{B2})]$$
$$+ [K_{SCL}(-X_{B2})] + [K_{SCL}(-X_{B2})]]$$
$$+ [K_{FL}(-X_{B2} + X_{SBR2})] + [K_{FL}(-X_{B2} + X_{SBL2})]$$

$$\ddot{B}_1 = \frac{L_B}{2I_B} \times [\left[K_{SV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1))) \right]$$
$$+ \left[K_{SCV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1))) \right]$$
$$+ \left[C_S(\dot{Z}_{SR1} - (\dot{Z}_{B1} + (\frac{L_B}{2} \times \dot{B}_1))) \right]$$
$$+ \left[K_{FV}((Z_{B1} + (\frac{L_B}{2} \times B_1)) + Z_{SBR1}) \right]$$
$$- \left[K_{SV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1))) \right]$$
$$- \left[K_{SCV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1))) \right]$$
$$- \left[C_S(\dot{Z}_{SL1} - (\dot{Z}_{B1} - (\frac{L_B}{2} \times \dot{B}_1))) \right]$$
$$- \left[K_{FV}((Z_{B1} - (\frac{L_B}{2} \times B_1)) + Z_{SBL1}) \right]$$
$$- \{K_{SIDE}((Z_{B1} + (\frac{L_{CB}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\} \quad [3.7.11]$$
$$+ \{K_{SIDE}((Z_{B1} - (\frac{L_{CB}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$- \{K_{CENV}((Z_{B1} + (\frac{D_{CEN}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}$$
$$+ \{K_{CENV}((Z_{B1} - (\frac{D_{CEN}}{2} \times B_1))$$
$$- (Z_C + (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}]$$
$$- \frac{L_{CGB}}{I_B} \times \begin{matrix} [\{K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}] \end{matrix}$$
$$- \frac{L_{CGB}}{I_B} \times \begin{matrix} [\{K_{CENL}((X_{B1} + (L_{CGB} \times B_1)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} + \frac{\Psi}{2}))))\}] \end{matrix}$$

$$
\begin{aligned}
\ddot{B}_2 = \frac{L_B}{2I_B} \times [& \left[K_{SV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[K_{SCV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2))) \right] \\
& + \left[C_S(\dot{Z}_{SR2} - (\dot{Z}_{B2} + (\frac{L_B}{2} \times \dot{B}_2))) \right] \\
& + \left[K_{FV}((Z_{B2} + (\frac{L_B}{2} \times B_2)) + Z_{SBR2}) \right] \\
& - \left[K_{SV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& - \left[K_{SCV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& - \left[C_S(\dot{Z}_{SL2} - (\dot{Z}_{B2} - (\frac{L_B}{2} \times \dot{B}_2))) \right] \\
& - \left[K_{FV}((Z_{B2} - (\frac{L_B}{2} \times B_2)) + Z_{SBL2}) \right] \\
& - \{K_{SIDE}((Z_{B2} + (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{SIDE}((Z_{B2} - (\frac{L_{CB}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{L_B}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& - \{K_{CENV}((Z_{B2} + (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) + (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\} \\
& + \{K_{CENV}((Z_{B2} - (\frac{D_{CEN}}{2} \times B_2)) \\
& - (Z_C - (\frac{L_{CEN}}{2} \times \Theta_{CP}) - (\frac{D_{CEN}}{2} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}] \\
& - \frac{L_{CGB}}{I_B} \times \begin{matrix} [\{K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}] \end{matrix} \\
& - \frac{L_{CGB}}{I_B} \times \begin{matrix} [\{K_{CENL}((X_{B2} + (L_{CGB} \times B_2)) \\ - (X_C - (L_{CGCAR} \times (\Theta_{CR} - \frac{\Psi}{2}))))\}] \end{matrix}
\end{aligned}
\quad [3.7.12]
$$

$$\ddot{Z}_{SR1} = \frac{1}{M_S} \times [\left[K_{SW}((Z_{WF1} + (\frac{L_B}{2} \times A_{F1})) - (Z_{SR1} + (\frac{L_S}{2} \times \Sigma_{R1})))\right]$$
$$+ \left[K_{SW}((Z_{WR1} + (\frac{L_B}{2} \times A_{R1})) - (Z_{SR1} - (\frac{L_S}{2} \times \Sigma_{R1})))\right]$$
$$- \left[K_{SV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1)))\right]$$
$$- \left[K_{SCV}(Z_{SR1} - (Z_{B1} + (\frac{L_B}{2} \times B_1)))\right] \quad [3.7.13]$$
$$- \left[C_S(\dot{Z}_{SR1} - (\dot{Z}_{B1} + (\frac{L_B}{2} \times \dot{B}_1)))\right]$$
$$- \left[K_{FV}((Z_{B1} + (\frac{L_B}{2} \times B_1)) + Z_{SBR1})\right]$$
$$- M_S g]$$

$$\ddot{Z}_{SL1} = \frac{1}{M_S} \times [\left[K_{SW}((Z_{WF1} - (\frac{L_B}{2} \times A_{F1})) - (Z_{SL1} + (\frac{L_S}{2} \times \Sigma_{L1})))\right]$$
$$+ \left[K_{SW}((Z_{WR1} - (\frac{L_B}{2} \times A_{R1})) - (Z_{SL1} - (\frac{L_S}{2} \times \Sigma_{L1})))\right]$$
$$- \left[K_{SV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1)))\right]$$
$$- \left[K_{SCV}(Z_{SL1} - (Z_{B1} - (\frac{L_B}{2} \times B_1)))\right] \quad [3.7.14]$$
$$- \left[C_S(\dot{Z}_{SL1} - (\dot{Z}_{B1} - (\frac{L_B}{2} \times \dot{B}_1)))\right]$$
$$- \left[K_{FV}((Z_{B1} - (\frac{L_B}{2} \times B_1)) + Z_{SBL1})\right]$$
$$- M_S g]$$

$$\ddot{Z}_{SR2} = \frac{1}{M_S} \times [\left[K_{SW}((Z_{WF2} + (\frac{L_B}{2} \times A_{F2})) - (Z_{SR2} + (\frac{L_S}{2} \times \Sigma_{R2})))\right]$$
$$+ \left[K_{SW}((Z_{WR2} + (\frac{L_B}{2} \times A_{R2})) - (Z_{SR2} - (\frac{L_S}{2} \times \Sigma_{R2})))\right]$$
$$- \left[K_{SV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2)))\right]$$
$$- \left[K_{SCV}(Z_{SR2} - (Z_{B2} + (\frac{L_B}{2} \times B_2)))\right] \quad [3.7.15]$$
$$- \left[C_S(\dot{Z}_{SR2} - (\dot{Z}_{B2} + (\frac{L_B}{2} \times \dot{B}_2)))\right]$$
$$- \left[K_{FV}((Z_{B2} + (\frac{L_B}{2} \times B_1)) + Z_{SBR2})\right]$$
$$- M_S g]$$

$$
\begin{aligned}
\ddot{Z}_{SL2} = \frac{1}{M_S} \times [& \left[K_{SW}((Z_{WF2} - (\frac{L_B}{2} \times A_{F2})) - (Z_{SL2} + (\frac{L_S}{2} \times \Sigma_{L2}))) \right] \\
& + \left[K_{SW}((Z_{WR2} - (\frac{L_B}{2} \times A_{R2})) - (Z_{SL2} - (\frac{L_S}{2} \times \Sigma_{L2}))) \right] \\
& - \left[K_{SV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& - \left[K_{SCV}(Z_{SL2} - (Z_{B2} - (\frac{L_B}{2} \times B_2))) \right] \\
& - \left[C_S(\dot{Z}_{SL2} - (\dot{Z}_{B2} - (\frac{L_B}{2} \times \dot{B}_2))) \right] \\
& - \left[K_{FV}((Z_{B2} - (\frac{L_B}{2} \times B_1)) + Z_{SBL2}) \right] \\
& - M_S g]
\end{aligned}
\qquad [3.7.16]
$$

$$
\begin{aligned}
\ddot{\Sigma}_{R1} = \frac{L_S}{2I_S} \times [& \left[K_{SW}((Z_{WF1} + (\frac{L_B}{2} \times A_{F1})) - (Z_{SR1} + (\frac{L_S}{2} \times \Sigma_{R1}))) \right] \\
& + \left[K_{SW}((Z_{WR1} + (\frac{L_B}{2} \times A_{R1})) - (Z_{SR1} - (\frac{L_S}{2} \times \Sigma_{R1}))) \right]]
\end{aligned}
\qquad [3.7.17]
$$

$$
\begin{aligned}
\ddot{\Sigma}_{L1} = \frac{L_S}{2I_S} \times [& \left[K_{SW}((Z_{WF1} - (\frac{L_B}{2} \times A_{F1})) - (Z_{SL1} + (\frac{L_S}{2} \times \Sigma_{L1}))) \right] \\
& + \left[K_{SW}((Z_{WR1} - (\frac{L_B}{2} \times A_{R1})) - (Z_{SL1} - (\frac{L_S}{2} \times \Sigma_{L1}))) \right]]
\end{aligned}
\qquad [3.7.18]
$$

$$
\begin{aligned}
\ddot{\Sigma}_{R2} = \frac{L_S}{2I_S} \times [& \left[K_{SW}((Z_{WF2} + (\frac{L_B}{2} \times A_{F2})) - (Z_{SR2} + (\frac{L_S}{2} \times \Sigma_{R2}))) \right] \\
& + \left[K_{SW}((Z_{WR2} + (\frac{L_B}{2} \times A_{R2})) - (Z_{SR2} - (\frac{L_S}{2} \times \Sigma_{R2}))) \right]]
\end{aligned}
\qquad [3.7.19]
$$

$$
\begin{aligned}
\ddot{\Sigma}_{L2} = \frac{L_S}{2I_S} \times [& \left[K_{SW}((Z_{WF2} - (\frac{L_B}{2} \times A_{F2})) - (Z_{SL2} + (\frac{L_S}{2} \times \Sigma_{L2}))) \right] \\
& + \left[K_{SW}((Z_{WR2} - (\frac{L_B}{2} \times A_{R2})) - (Z_{SL2} - (\frac{L_S}{2} \times \Sigma_{L2}))) \right]]
\end{aligned}
\qquad [3.7.20]
$$

$$\ddot{Z}_{WF1} = \frac{1}{M_W} \times [\left[K_{RAIL}(Z_{RAIL\,FR1} - (Z_{WF1} + (\frac{L_W}{2} \times A_{F1})))\right] + \left[K_{RAIL}(Z_{RAIL\,FL1} - (Z_{WF1} - (\frac{L_W}{2} \times A_{F1})))\right] - \left[K_{SW}((Z_{WF1} + (\frac{L_B}{2} \times A_{F1})) - (Z_{SR1} + (\frac{L_S}{2} \times \Sigma_{R1})))\right] - \left[K_{SW}((Z_{WF1} - (\frac{L_B}{2} \times A_{F1})) - (Z_{SL1} + (\frac{L_S}{2} \times \Sigma_{L1})))\right] - M_W g] \quad [3.7.21]$$

$$\ddot{Z}_{WR1} = \frac{1}{M_W} \times [\left[K_{RAIL}(Z_{RAIL\,RR1} - (Z_{WR1} + (\frac{L_W}{2} \times A_{R1})))\right] + \left[K_{RAIL}(Z_{RAIL\,RL1} - (Z_{WR1} - (\frac{L_W}{2} \times A_{R1})))\right] - \left[K_{SW}((Z_{WR1} + (\frac{L_B}{2} \times A_{R1})) - (Z_{SR1} - (\frac{L_S}{2} \times \Sigma_{R1})))\right] - \left[K_{SW}((Z_{WR1} - (\frac{L_B}{2} \times A_{R1})) - (Z_{SL1} - (\frac{L_S}{2} \times \Sigma_{L1})))\right] - M_W g] \quad [3.7.22]$$

$$\ddot{Z}_{WF2} = \frac{1}{M_W} \times [\left[K_{RAIL}(Z_{RAIL\,FR2} - (Z_{WF2} + (\frac{L_W}{2} \times A_{F2})))\right] + \left[K_{RAIL}(Z_{RAIL\,FL2} - (Z_{WF2} - (\frac{L_W}{2} \times A_{F2})))\right] - \left[K_{SW}((Z_{WF2} + (\frac{L_B}{2} \times A_{F2})) - (Z_{SR2} + (\frac{L_S}{2} \times \Sigma_{R2})))\right] - \left[K_{SW}((Z_{WF2} - (\frac{L_B}{2} \times A_{F2})) - (Z_{SL2} + (\frac{L_S}{2} \times \Sigma_{L2})))\right] - M_W g] \quad [3.7.23]$$

$$\ddot{Z}_{WR2} = \frac{1}{M_W} \times [\left[K_{RAIL}(Z_{RAIL\,RR2} - (Z_{WR2} + (\frac{L_W}{2} \times A_{R2})))\right] + \left[K_{RAIL}(Z_{RAIL\,RL2} - (Z_{WR2} - (\frac{L_W}{2} \times A_{R2})))\right] - \left[K_{SW}((Z_{WR2} + (\frac{L_B}{2} \times A_{R2})) - (Z_{SR2} - (\frac{L_S}{2} \times \Sigma_{R2})))\right] - \left[K_{SW}((Z_{WR2} - (\frac{L_B}{2} \times A_{R2})) - (Z_{SL2} - (\frac{L_S}{2} \times \Sigma_{L2})))\right] - M_W g] \quad [3.7.24]$$

$$\ddot{A}_{F1} = \frac{L_W}{2I_W} \times [\left[K_{RAIL}(Z_{RAIL\,FR1} - (Z_{WF1} + (\frac{L_W}{2} \times A_{F1})))\right] - \left[K_{RAIL}(Z_{RAIL\,FL1} - (Z_{WF1} - (\frac{L_W}{2} \times A_{F1})))\right] - \left[K_{SW}((Z_{WF1} + (\frac{L_B}{2} \times A_{F1})) - (Z_{SR1} + (\frac{L_S}{2} \times \Sigma_{R1})))\right] + \left[K_{SW}((Z_{WF1} - (\frac{L_B}{2} \times A_{F1})) - (Z_{SL1} + (\frac{L_S}{2} \times \Sigma_{L1})))\right]] \quad [3.7.25]$$

$$\ddot{A}_{R1} = \frac{L_W}{2I_W} \times [\left[K_{RAIL}(Z_{RAIL\,RR1} - (Z_{WR1} + (\frac{L_W}{2} \times A_{R1})))\right] - \left[K_{RAIL}(Z_{RAIL\,RL1} - (Z_{WR1} - (\frac{L_W}{2} \times A_{R1})))\right] - \left[K_{SW}((Z_{WR1} + (\frac{L_B}{2} \times A_{R1})) - (Z_{SR1} - (\frac{L_S}{2} \times \Sigma_{R1})))\right] + \left[K_{SW}((Z_{WR1} - (\frac{L_B}{2} \times A_{R1})) - (Z_{SL1} - (\frac{L_S}{2} \times \Sigma_{L1})))\right]] \quad [3.7.26]$$

$$\ddot{A}_{F2} = \frac{L_W}{2I_W} \times [\left[K_{RAIL}(Z_{RAIL\,FR2} - (Z_{WF2} + (\frac{L_W}{2} \times A_{F2})))\right] - \left[K_{RAIL}(Z_{RAIL\,FL2} - (Z_{WF2} - (\frac{L_W}{2} \times A_{F2})))\right] - \left[K_{SW}((Z_{WF2} + (\frac{L_B}{2} \times A_{F2})) - (Z_{SR2} + (\frac{L_S}{2} \times \Sigma_{R2})))\right] + \left[K_{SW}((Z_{WF2} - (\frac{L_B}{2} \times A_{F2})) - (Z_{SL2} + (\frac{L_S}{2} \times \Sigma_{L2})))\right]] \quad [3.7.27]$$

$$\ddot{A}_{R2} = \frac{L_W}{2I_W} \times [\left[K_{RAIL}(Z_{RAIL\,RR2} - (Z_{WR2} + (\frac{L_W}{2} \times A_{R2})))\right] - \left[K_{RAIL}(Z_{RAIL\,RL2} - (Z_{WR2} - (\frac{L_W}{2} \times A_{R2})))\right] - \left[K_{SW}((Z_{WR2} + (\frac{L_B}{2} \times A_{R2})) - (Z_{SR2} - (\frac{L_S}{2} \times \Sigma_{R2})))\right] + \left[K_{SW}((Z_{WR2} - (\frac{L_B}{2} \times A_{R2})) - (Z_{SL2} - (\frac{L_S}{2} \times \Sigma_{L2})))\right]] \quad [3.7.28]$$

Chapter 4

The Computer Program

This chapter outlines the important features of PROGRAM TERM, a FORTRAN77 computer program that solves the 28 equations of motion presented in Chapter 3.

PROGRAM TERM consists of several different subroutines that facilitate the computer simulation. PROGRAM MAIN is the driver program; it initializes all variables, establishes the time step and interval of solution, and calls other subroutines during the solution. SUBROUTINE RK4SYS implements a fourth-order Runge-Kutta equation solver. SUBROUTINE FUNCT provides the equations of motion for the system, and SUBROUTINES RAILF1, RAILR1, RAILF2, AND RAILR2 provide the random rail inputs to the system. For convenience, each of these sections will be separately discussed. A complete program listing is provided in Appendix 1.

4.1 PROGRAM MAIN

PROGRAM MAIN is the only interactive portion of the program. In this driver program, the user encodes most program variables, including railcar physical property data and initial conditions for the vehicle responses. In addition, the user can interactively specify output intervals, operating speed, solution time interval, and integration time step.

The physical property values are briefly described in the program documentation. These parameters refer to the railcar dimensions, inertias, and stiffnesses illustrated in Figure 1 on page 4 and Figure 6 on page 31. Typical parameters for 70-ton and 100-ton vehicles have been compiled from several different sources, including Tse (1974), Platin, et al. (1976), Cooperrider, et al. (1981), and White (1985). A complete listing of these physical property values for 70-ton railcars is presented in Table 3 on page 65. A similar compilation for 100-ton vehicles is presented in Table 4 on page 67.

During program execution, the vehicle speed VEL is entered interactively in (m/s). Other inputs which are frequently varied for each analysis, such as integration time step H (sec), total run time XT (sec), and output factor NUMOUT are interactive inputs for convenience. The output factor NUMOUT allows the user to specify how often the generated results will be printed into output files. An output factor of 1 specifies that results will be printed every time step; an output factor of 100 creates results every 100 time steps, etc.

Once all input variables have been specified, the system's initial conditions are specified. These initial conditions, stored in array Y0(1) - Y0(54), refer to a zero-datum position for each specific

Table 3. Vehicle Parameters for a Typical 70-ton Railcar

Program Symbol	Description	Value	Units
MC	Carbody Mass (With Coal Load)	77,200	kg
MB	Bolster Mass	527	kg
MS	Sideframe Mass	350	kg
MW	Wheelset Mass	1120	kg
LCAR	Carbody Length	14.9	m
WCAR	Carbody Width	3.50	m
LCB	Lateral Distance Between Sidebearer Springs	1.27	m
LB	Bolster Length	2.01	m
LS	Sideframe Length	1.73	m
LW	Wheelset Length	1.43	m
LCEN	Longitudinal Distance Between Centerplates	12.0	m
DCEN	Diameter of Centerplates	.356	m
LCGCAR	Height of Carbody Center of Gravity (Loaded)	1.75	m
LCGB	Height of Bolster Center of Gravity	.12	m
ICP	Carbody Pitch Moment of Inertia	1.43×10^6	$kg \times m^2$
ICR	Carbody Roll Moment of Inertia	78800	$kg \times m^2$
IB	Bolster Roll Moment of Inertia	243	$kg \times m^2$
IS	Sideframe Pitch Moment of Inertia	113	$kg \times m^2$
IW	Wheelset Roll Moment of Inertia	610	$kg \times m^2$
EI	Carbody Bending Stiffness	3.00×10^8	$N \times m^2$
JGR	Carbody Torsional Stiffness	2.00×10^8	$\frac{N \times m^2}{rad}$
FRICTL	Lateral Suspension Coulomb Damping Coefficient	9000	N
FRICTV	Vertical Suspension Coulomb Damping Coefficient	9000	N

Table 3. 70-ton Vehicle Parameters (Continued)

Program Symbol	Description	Current Value	Units
KCB	Vertical Sidebearing Spring Stiffness	6.27×10^8	N/m
KCENL	Lateral Centerplate Spring Stiffness	3.71×10^8	N/m
KCENV	Vertical Centerplate Spring Stiffness	3.75×10^7	N/m
KSUSL	Lateral Suspension Linear Spring Stiffness	1.63×10^6	N/m
KSUSV	Vertical Suspension Linear Spring Stiffness	3.56×10^6	N/m
KSUSCL	Lateral Suspension Clearance Spring Stiffness	1.46×10^7	N/m
KSUSCV	Vertical Suspension Clearance Spring Stiffness	5.62×10^7	N/m
KSUSFL	Lateral Suspension Friction Spring Stiffness	1.0×10^9	N/m
KSUSFV	Vertical Suspension Friction Spring Stiffness	1.0×10^9	N/m
KRAIL	Vertical Rail/Bed Spring Stiffness	1.75×10^7	N/m
CRAIL	Vertical Rail/Bed Damping Coefficient	17,500	$\frac{N \times sec}{m}$
CSUSV1	Region 1 Stucki Viscous Damping Coefficient	3.5×10^6	$\frac{N \times sec}{m}$
CSUSV2	Region 2 Stucki Viscous Damping Coefficient	146,000	$\frac{N \times sec}{m}$
C0	Stucki Viscous Damping Intercept	89,000	N
CCB	Vertical Clearance Between Sidebearing Springs	0.007	m
CSLC	Lateral Suspension Clearance (Compressive)	0.005	m
CSLT	Lateral Suspension Clearance (Tensile)	0.005	m
CSVC	Vertical Suspension Clearance (Compressive)	0.05	m
CSVT	Vertical Suspension Clearance (Tensile)	0.05	m

Table 4. Vehicle Parameters for a Typical 100-ton Railcar

Program Symbol	Description	Value	Units
MC	Carbody Mass (With Coal Load)	92,900	kg
MB	Bolster Mass	619	kg
MS	Sideframe Mass	447	kg
MW	Wheelset Mass	1430	kg
LCAR	Carbody Length	14.8	m
WCAR	Carbody Width	3.50	m
LCB	Lateral Distance Between Sidebearer Springs	1.27	m
LB	Bolster Length	2.01	m
LS	Sideframe Length	1.78	m
LW	Wheelset Length	1.43	m
LCEN	Longitudinal Distance Between Centerplates	11.9	m
DCEN	Diameter of Centerplates	.356	m
LCGCAR	Height of Carbody Center of Gravity (Loaded)	1.75	m
LCGB	Height of Bolster Center of Gravity	.12	m
ICP	Carbody Pitch Moment of Inertia	1.69×10^6	$kg \times m^2$
ICR	Carbody Roll Moment of Inertia	94900	$kg \times m^2$
IB	Bolster Roll Moment of Inertia	249	$kg \times m^2$
IS	Sideframe Pitch Moment of Inertia	158	$kg \times m^2$
IW	Wheelset Roll Moment of Inertia	850	$kg \times m^2$
EI	Carbody Bending Stiffness	3.00×10^8	$N \times m^2$
JGR	Carbody Torsional Stiffness	2.00×10^8	$\frac{N \times m^2}{rad}$
FRICTL	Lateral Suspension Coulomb Damping Coefficient	22000	N
FRICTV	Vertical Suspension Coulomb Damping Coefficient	22000	N

Table 4. 100-ton Vehicle Parameters (Continued)

Program Symbol	Description	Current Value	Units
KCB	Vertical Sidebearing Spring Stiffness	6.27×10^8	N/m
KCENL	Lateral Centerplate Spring Stiffness	3.71×10^8	N/m
KCENV	Vertical Centerplate Spring Stiffness	3.75×10^7	N/m
KSUSL	Lateral Suspension Linear Spring Stiffness	1.96×10^6	N/m
KSUSV	Vertical Suspension Linear Spring Stiffness	3.93×10^6	N/m
KSUSCL	Lateral Suspension Clearance Spring Stiffness	1.46×10^7	N/m
KSUSCV	Vertical Suspension Clearance Spring Stiffness	5.62×10^7	N/m
KSUSFL	Lateral Suspension Friction Spring Stiffness	1.0×10^9	N/m
KSUSFV	Vertical Suspension Friction Spring Stiffness	1.0×10^9	N/m
KRAIL	Vertical Rail/Bed Spring Stiffness	1.75×10^7	N/m
CRAIL	Vertical Rail/Bed Damping Coefficient	17,500	$\frac{N \times sec}{m}$
CSUSV1	Region 1 Stucki Viscous Damping Coefficient	3.5×10^6	$\frac{N \times sec}{m}$
CSUSV2	Region 2 Stucki Viscous Damping Coefficient	146,000	$\frac{N \times sec}{m}$
C0	Stucki Viscous Damping Intercept	89,000	N
CCB	Vertical Clearance Between Sidebearing Springs	0.007	m
CSLC	Lateral Suspension Clearance (Compressive)	0.005	m
CSLT	Lateral Suspension Clearance (Tensile)	0.005	m
CSVC	Vertical Suspension Clearance (Compressive)	0.05	m
CSVT	Vertical Suspension Clearance (Tensile)	0.05	m

coordinate. Most coordinates are initialized to zero, with exception of the vertical displacement coordinates. These vertical coordinates are initialized to their static equilibrium positions, which have been calculated and implemented into the program.

Once initial conditions have been evaluated, PROGRAM MAIN calls SUBROUTINE RK4SYS to solve the system equations for one time step. Once these equations have been solved, the system values Y(1) - Y(54) are returned to PROGRAM MAIN for printing, if necessary. PROGRAM MAIN continually indexes the time step and calls SUBROUTINE RK4SYS for solutions until the termination time is reached. At each time step, the relative values for each system variable are evaluated to establish maximums over the total time interval. These maximums are stored in the array YMAX(1) - YMAX(54), and are printed in output file YMAX at the conclusion of the run. It should be noted that these maximums are relative to the initial conditions specified for each coordinate; they represent the maximum absolute displacements from given initial conditions.

Progress during the run is continually monitored and appears on the CRT screen in ten percent increments. Once the run has completed, a PROGRAM FINISHED report notifies the user.

4.2 SUBROUTINE RK4SYS

SUBROUTINE RK4SYS implements a fourth-order Runge-Kutta system solver for the equations of motion. This integration scheme can be applied to any system of first-order equations. The equations of motion derived in Chapter 3 are easily converted to a system of 54 first-order equations using a procedure outlined by Atkinson, (1985). These 54 equations are then solved at each time step, t_{i+1} , for a given integration step H using the fourth-order Runge-Kutta algorithm:

$$Y_{i+1} = Y_i + \frac{H}{6}[v_1 + 2v_2 + 2v_3 + v_4] \qquad [4.3.1]$$

where:

$$v_1 = f(t_i, Y_i) \qquad [4.3.2]$$

$$v_2 = f(t_i + \frac{H}{2}, Y_i + \frac{H}{2}v_1) \qquad [4.3.3]$$

$$v_3 = f(t_i + \frac{H}{2}, Y_i + \frac{H}{2}v_2) \qquad [4.3.4]$$

$$v_4 = f(t_i + H, Y_i + Hv_3) \qquad [4.3.5]$$

The functional evaluations of Equation 4.3.1, $f(t, Y)$ are the first derivatives of the state variables. They are different for each system variable Y, and are determined from the first-order equations of motion for the system. These functions are evaluated in SUBROUTINE FUNCT, which is called 4 times every time step for each of the 54 first-order equations. Once each of the 54 system variables has been solved at the current time step, t_i, SUBROUTINE RK4SYS returns these values to PROGRAM MAIN.

4.3 SUBROUTINE FUNCT

SUBROUTINE FUNCT performs the function evaluations required by the Runge-Kutta algorithm of Equation 4.3.1. Because the 54 equations of motion are evaluated four times each time step, functional computations are the critical consumer of CPU time. For this reason, increasing the efficiency of function evaluations is very beneficial to reduce overall run costs.

To increase the efficiency of function evaluations, dynamic forces are computed separately and then applied wherever applicable in the equations of motion. This eliminates duplication of effort in calculating common forces. A single force, such as the front right vertical sidebearer force, can appear in several different equations of motion. To evaluate such a force every time it appears in a different equation would be quite costly.

From the Runge-Kutta algorithm of Equation 3.4.1, notice that the four function evaluations occur at different intervals within the time step. The first function evaluation occurs at the beginning of the time step. The second and third evaluations occur midway through the time step, and the fourth evaluation occurs at the end of the time step. For this reason, the rail input subroutines are called on the second and fourth function evaluations of each time step. The rail inputs at the second function computation are stored and used again at the third computation. Similarly, inputs at the fourth evaluation are stored and used again at the beginning of the next time step.

4.4 SUBROUTINES RAILF1, RAILR1, RAILF2, AND RAILR2

SUBROUTINE RAILF1 provides the random rail inputs for the front wheelset of the first truck. The subroutine uses the current time value t_i in addition to several other track and operating conditions to evaluate the left and right rail inputs. These inputs include the displacements, velocities, and accelerations of the left and right rail, although only the displacements are used in the computer simulation. Complete documentation for the random rail inputs routine is provided by Fries, (1987). Similarly, SUBROUTINES RAILR1, RAILF2, and RAILR2 provide the inputs for the first rear wheelset, the second front wheelset, and the second rear wheelset, respectively. These routines are called at times corresponding to the transport lag from the front wheelset to the respective wheelset.

4.5 Run Guidelines

PROGRAM TERM has been successfully tested for a variety of physical property values and operating conditions. It should be noted, however, that some considerations are necessary to insure valid simulations. Factors such as vehicle speed, initial conditions, track class, maximum cross-level, and physical properties have significant effects on the computer simulations. The most critical variable of any analysis, however, is the integration time step.

Judicial selection of an optimal time step depends on a variety of factors. Choosing excessively large time steps introduces the possibility of inaccurate analyses and numerical instability, while smaller time steps can drastically increase CPU time consumption. Although there are no set rules for selecting an optimal time step, some general guidelines will be mentioned. Most runs require a nominal time step of .0001 seconds for accurate results at moderate operating conditions. "Moderate operating conditions" refers to vehicle speeds less than 8 m/s, cross-level inputs less than .025 m, standard rail lengths, and typical physical property data (see Table 3 on page 65, and Table 4 on page 67). Simulations at higher vehicle speeds or more extreme operating conditions should employ reduced time steps.

The user should note that the rail inputs are dependent on selection of time step and vehicle speed; different values for these parameters will yield different input histories.

Chapter 5

Simulation Results

This chapter presents results from several different dynamic simulations using the TERM vertical dynamic model. Section 5.1 begins the analysis cases with a direct comparison of the TERM model to the AAR Flexible Carbody Model (Hussain, 1980). Section 5.2 evaluates the effects of auxiliary damping on vehicle response, and Section 5.3 examines the effects of increased coal loads on vehicle response.

5.1 Model Comparison

The AAR Flexible Carbody Vehicle Model, originally developed by Tse et al. (1974), has been extensively tested and is currently utilized in the railroad industry. Although many features of the

AAR model are similar to the TERM model, several fundamental differences exist. The 20 degree-of-freedom AAR model, illustrated in Figure 10 on page 77, consists of two half-carbodies connected by bending and torsional springs. Another obvious difference is AAR's lumped-mass representation of trucks. The 20 degrees of freedom are presented below in Table 5.

Table 5. The AAR Model: Degrees of freedom

Component	Freedom	Total
Front Carbody	Vertical Lateral Pitch Roll	1 1 1 1
Rear Carbody	Vertical Lateral Pitch Roll	1 1 1 1
Bolsters	Vertical Lateral Roll	2 2 2
Trucks	Vertical Lateral Roll	2 2 2
Total Degrees		**20**

In this section, a comparison between the TERM model and the AAR Flexible Carbody model is presented. Because the AAR model has no provision for runs on random track, a harmonic rail input is used for the comparison. Specifically, a rectified sine wave track input from the AAR program is employed. The left and right rail inputs are staggered by a half cycle to excite roll responses. A plot of these rail inputs is included in Figure 11 on page 78.

For the comparison run, the physical properties of a typical 100-ton railcar are used. These parameters are found in Table 4 on page 67, with the notable exception that auxiliary dampers are not included. A compilation of run conditions is provided below in Table 6. Selected results from the comparison run are presented graphically in Figure 11 on page 78, and Figure 12 on page 79. A listing of maximum displacements for various coordinates is presented in Table 7 on page 80.

Table 6. The Comparison Run Conditions.

Parameter	Value	Units
Vehicle Velocity	6.7056	$\frac{m}{sec}$
Track Period	23.7744	m
Track Amplitude	0.0127	m
Integration Time step	0.0001	sec
Run Duration	17.0000	sec

Figure 11 compares the carbody roll and vertical translation responses for each model. Although they compare favorably, a cautious interpretation of the plots is essential. The TERM vertical carbody response shows the **total** vertical carbody displacement at the midspan, including bending deflection. The AAR vertical carbody plot refers only to the **front** carbody displacement. Similarly, the TERM carbody roll response includes torsional contributions from the entire carbody, while the AAR roll response pertains only to the front carbody.

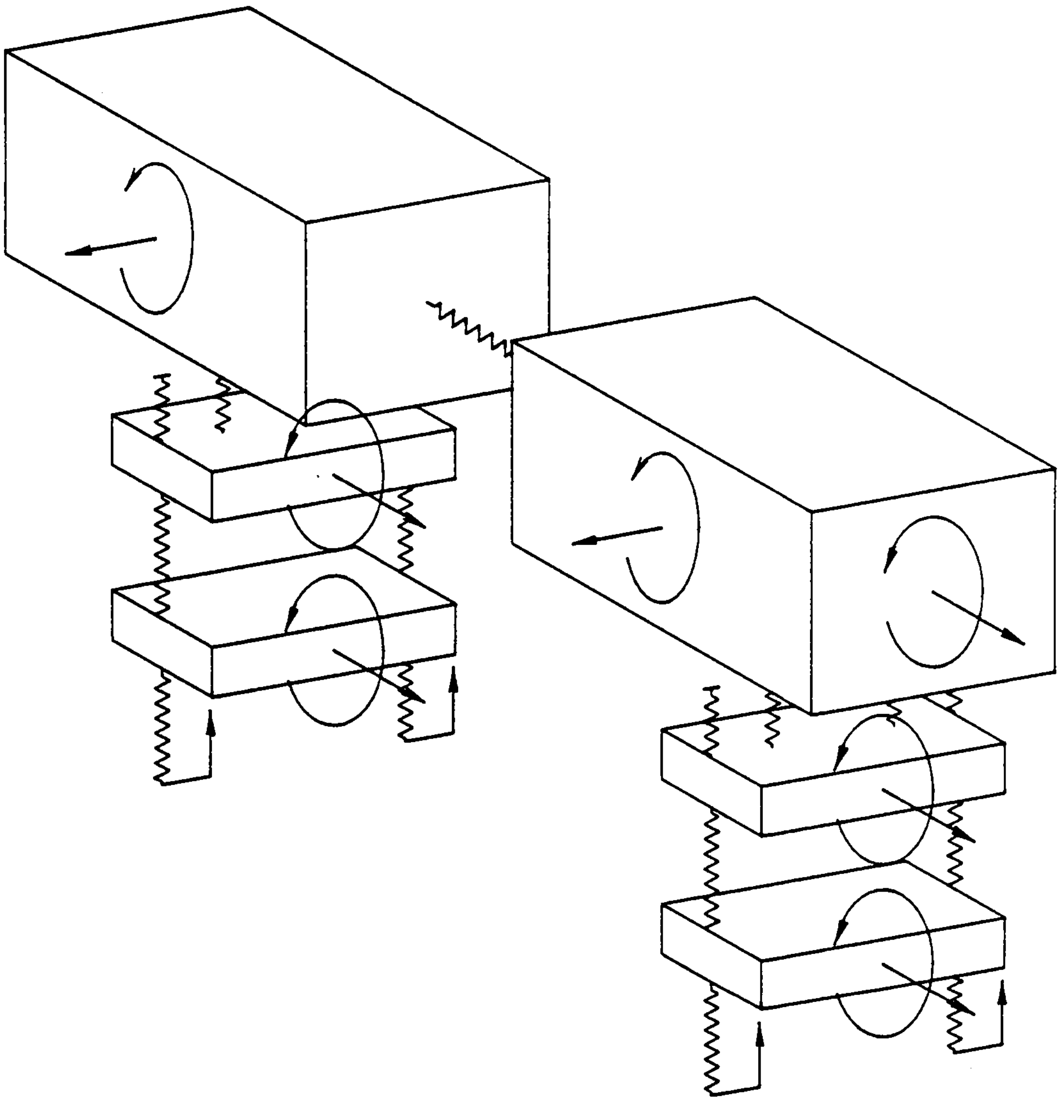

Figure 10. The 20 Degree-of-freedom AAR Railcar Model

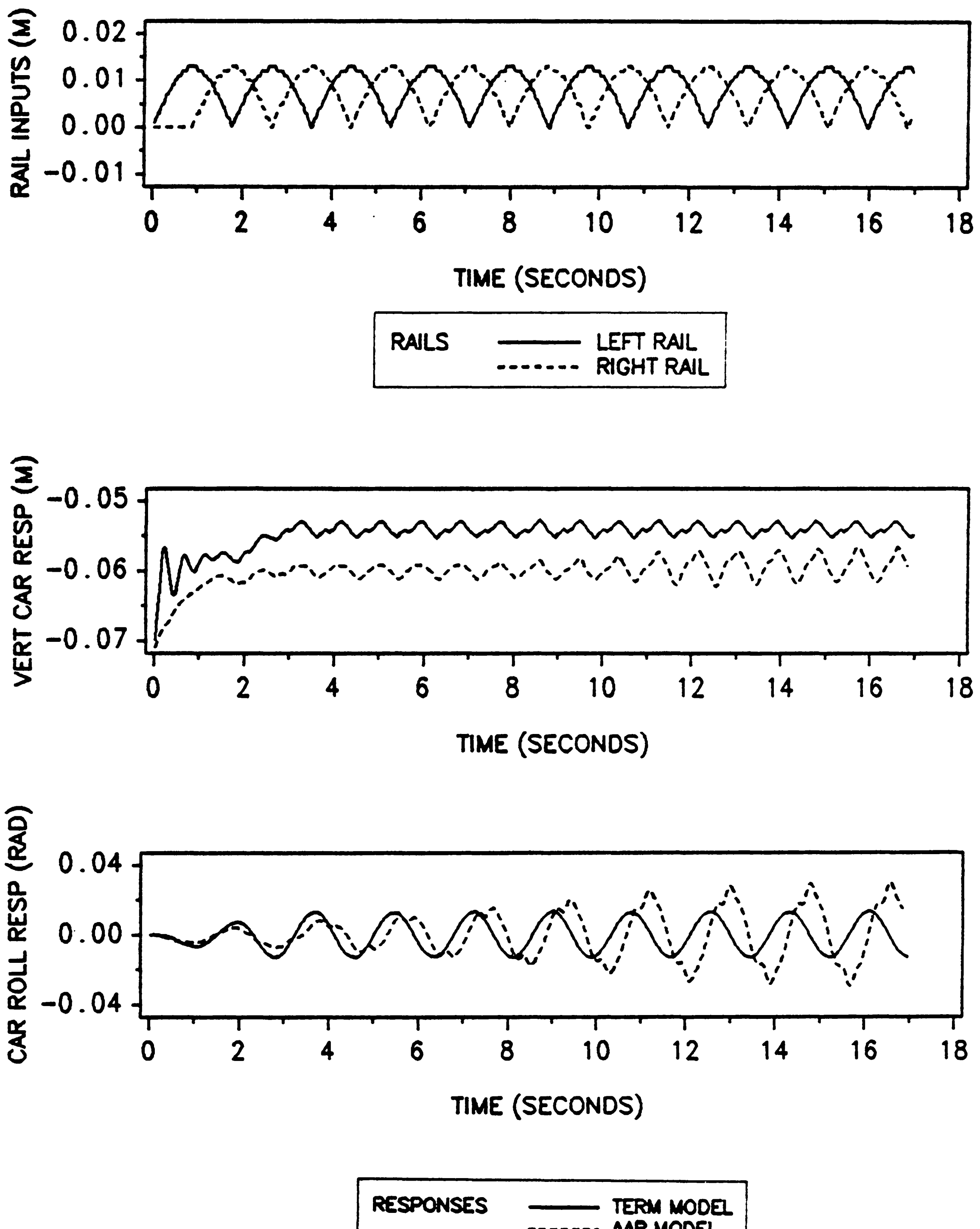

Figure 11. The Carbody Response Comparisons

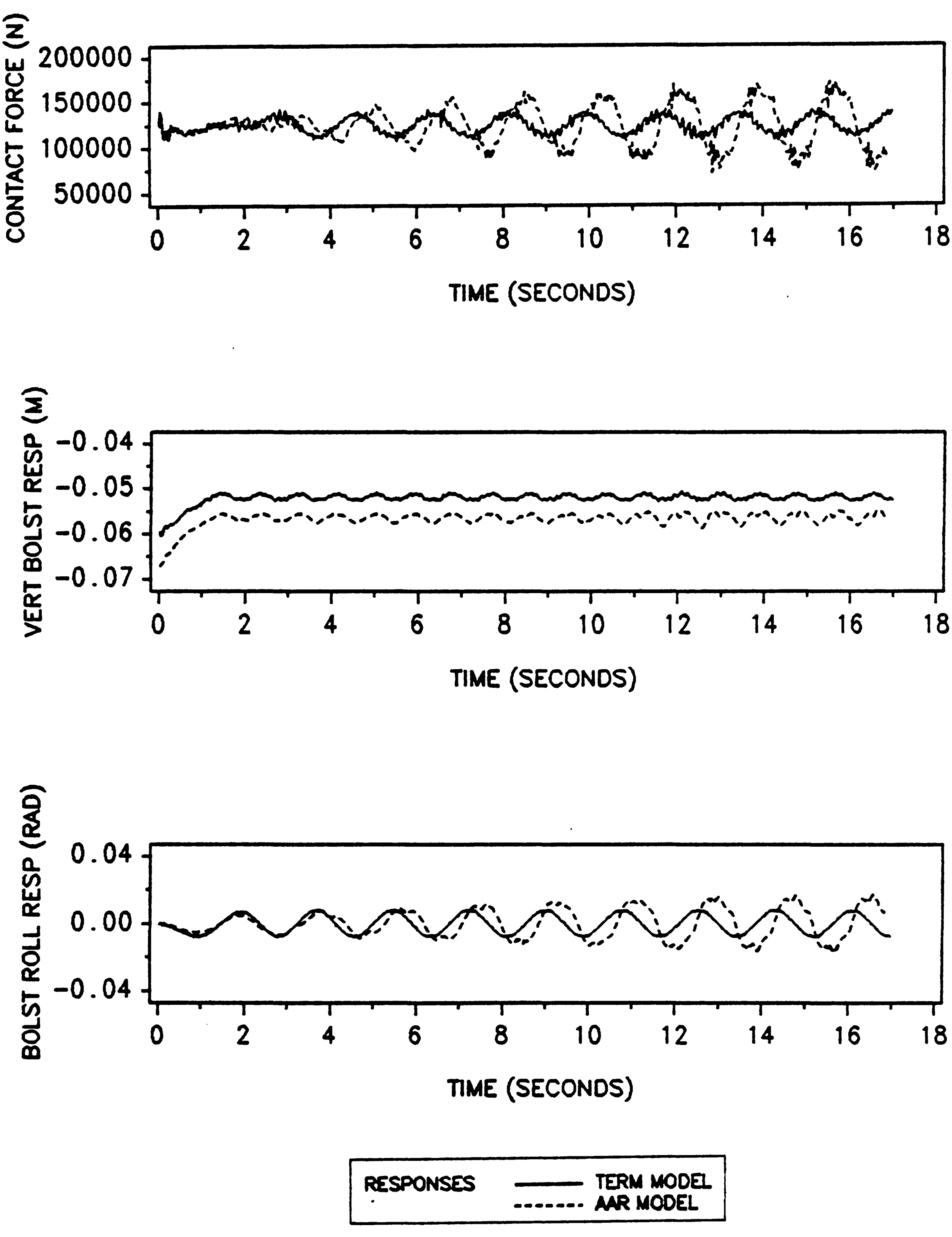

Figure 12. The Front Bolster Responses and Contact Force Comparisons

Table 7. Roll Comparison Results

Coordinate	Symbol	Maximum AAR Value	Maximum TERM Value	Units
Carbody Vertical Translation	Z_C	-0.0631	-0.0551	m
Carbody Lateral Translation	X_C	-0.0711	-0.0252	m
Carbody Roll Rotation	Θ_{CR}	-0.0298	0.0178	rad
Front Bolster Vertical Translation	Z_{B1}	-0.0601	-0.05371	m
Rear Bolster Vertical Translation	Z_{B2}	-0.0674	-0.05534	m
Front Bolster Roll Rotation	B_1	-0.0175	-0.0170	rad
Rear Bolster Roll Rotation	B_2	-0.0175	-0.0170	rad
Front Wheelset Vertical Translation	Z_{W1}	-0.0057	-0.0042	m
Rear Wheelset Vertical Translation	Z_{W2}	-0.0058	-0.0042	m
Front Wheelset Roll Rotation	A_1	0.0079	0.0097	rad
Rear Wheelset Roll Rotation	A_2	0.0082	-0.0098	rad

Regardless of modelling discrepancies, the response plots of Figure 11 highlight similar response trends for each model. The vertical carbody responses show similar frequency and amplitude content with a slight vertical offset. The initial oscillatory behavior of the TERM vertical carbody response may be due to differences in truck damping between the two models. The carbody roll responses are slightly out of phase and differ in amplitude. These inconsistencies can be justified, however, by observing the different inertial properties of the AAR half-carbody and the TERM continuous carbody.

Figure 12 shows the front bolster comparison responses for each model. In this case, the response plots refer to dynamically similar components in the TERM and AAR models. For this reason, the bolster responses compare quite favorably. The noticable offset in the vertical bolster responses might indicate a stiffer suspension in the TERM model.

Figure 12 also compares contact forces at the front right truck for each model. Differences in these contact force responses are easily explained. The TERM contact force is one of eight input forces, while the AAR contact force is one of only four. For this reason, the AAR contact force is halved for comparison. This accounts for the slight phase shift; the AAR contact force occurs at the center of the front truck, and the TERM contact force occurs at the first wheelset of the front truck.

5.2 *Auxiliary Damping Effects*

As described in Chapter 2, the TERM dynamic model can incorporate linear (or bi-linear) viscous dampers in the vertical suspension. Viscous dampers of this type are manufactured by the A. Stucki Company. They are reportedly quite effective in reducing the effects of resonant rocking at low speeds and severe bouncing at higher speeds (Stucki 1985).

To establish the effectiveness of Stucki dampers in the vertical suspension, the TERM model was tested at various speeds using the 100-ton vehicle parameters with and without auxiliary dampers. The primary goal of each simulation was to evaluate the maximum carbody roll during a five second run over class 4 random track inputs. The results of these runs are presented graphically in Figure 13 on page 83 and Figure 14 on page 84. Figure 13 shows that the carbody roll response is largely unaffected by auxiliary damping at very low speeds. As speeds approach the resonant condition of approximately 9 m/s, however, their influence is drastic. Similarly, the vertical carbody responses of Figure 14 illustrate the increasing influence of auxilary damping near the resonant speed. Both models experience amplified roll and displacement responses near 7.5 m/s, but only the model with auxiliary dampers recovers above this critical speed.

Although the random rail inputs are velocity-dependent, the differences in input histories at different speeds is relatively small. Figure 15 on page 85 and Figure 16 on page 87 show that the statistical trends of the inputs are preserved over the velocity range 5 - 10 m/s.

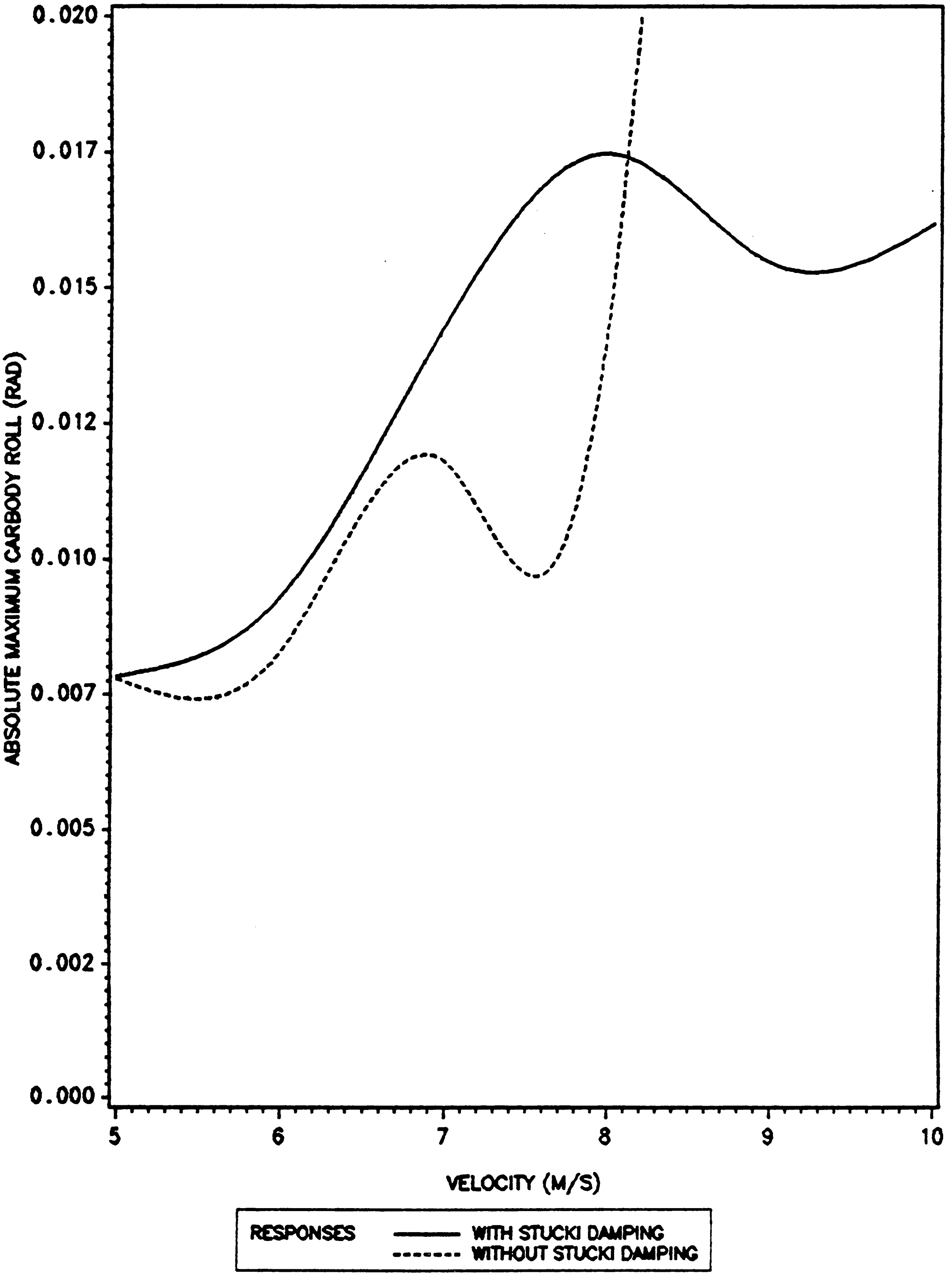

Figure 13. Maximum Carbody Roll Responses

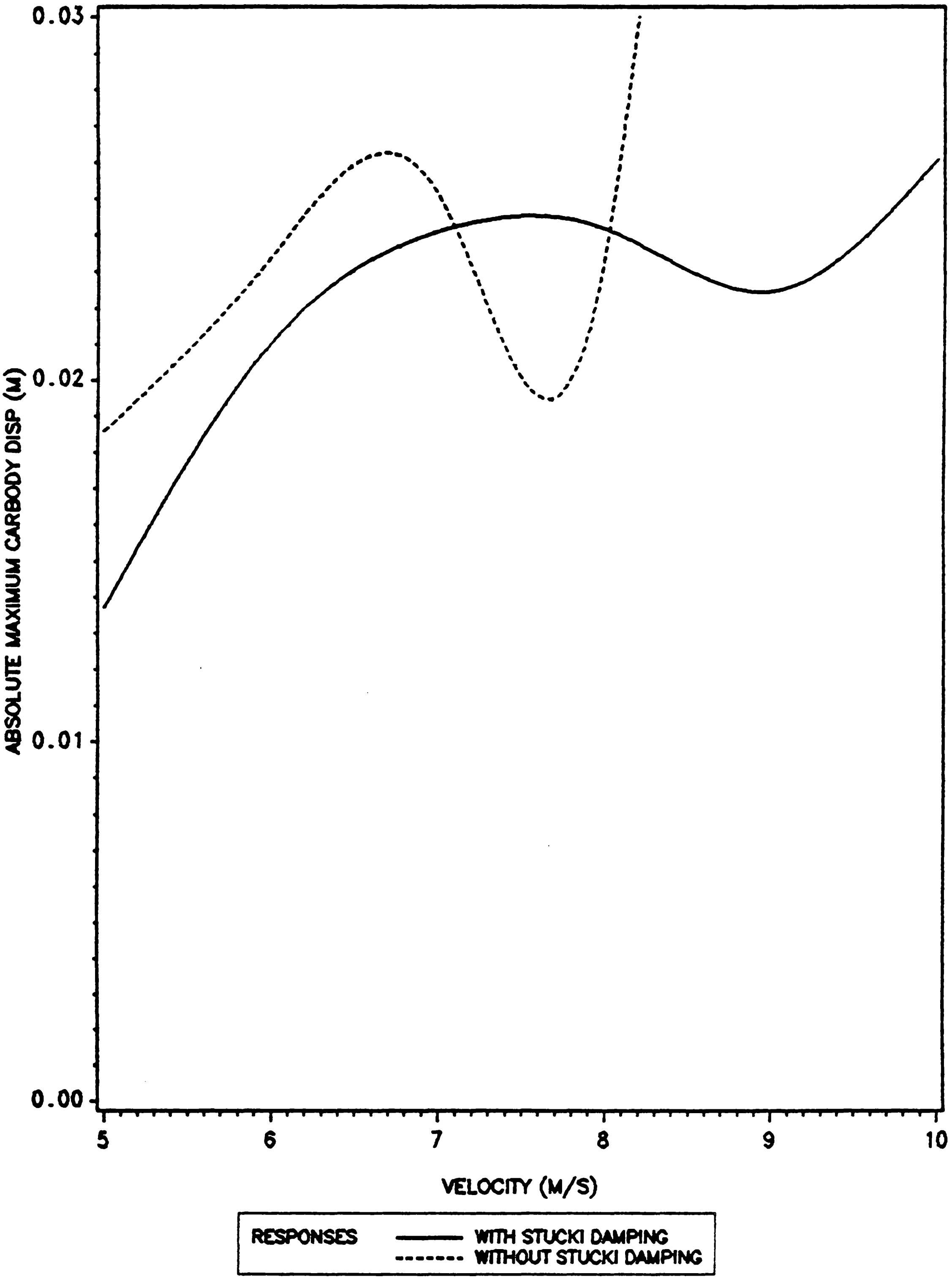

Figure 14. Maximum Carbody Translation Responses

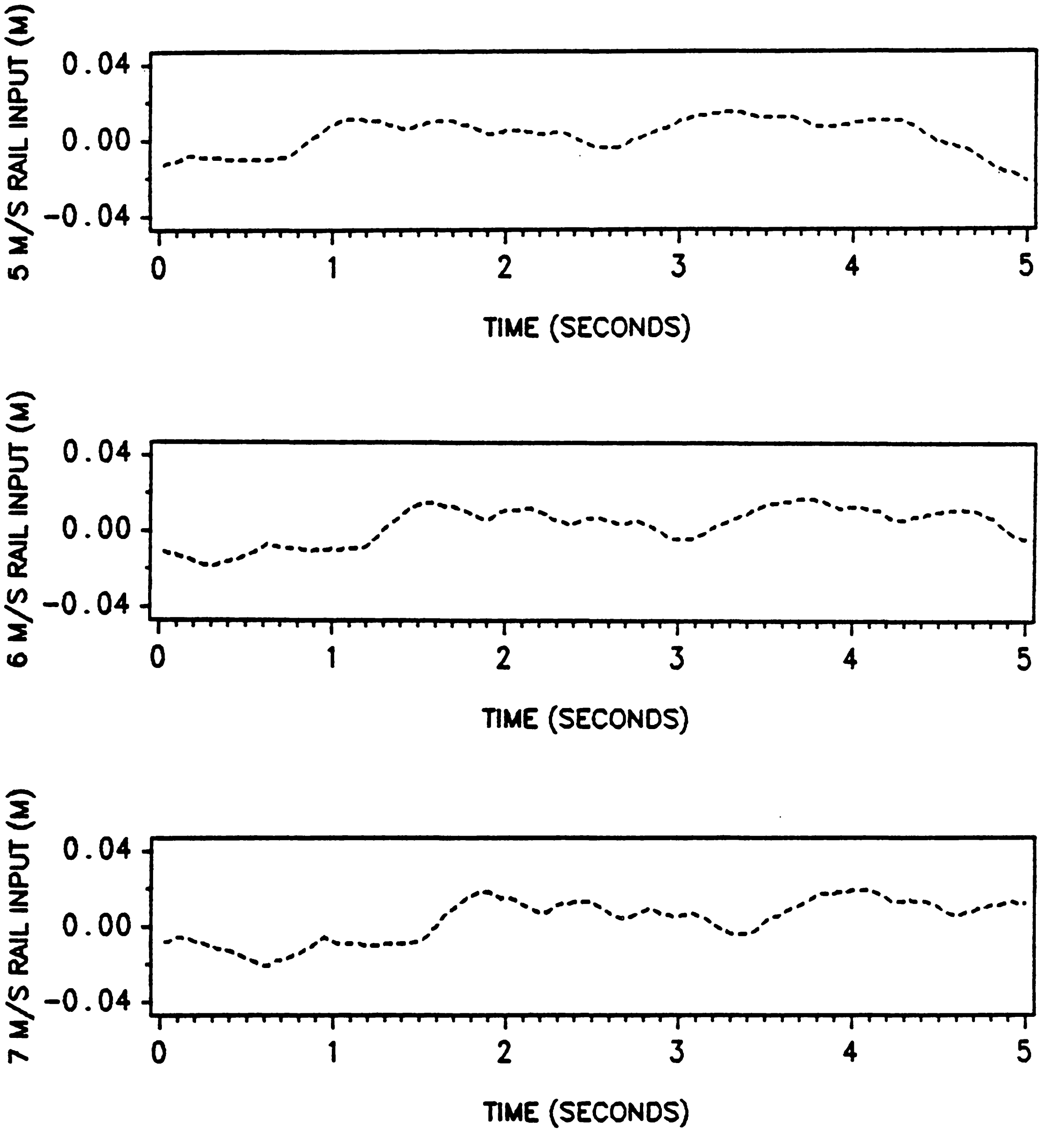

Figure 15. The Rail Input Plots at Various Speeds

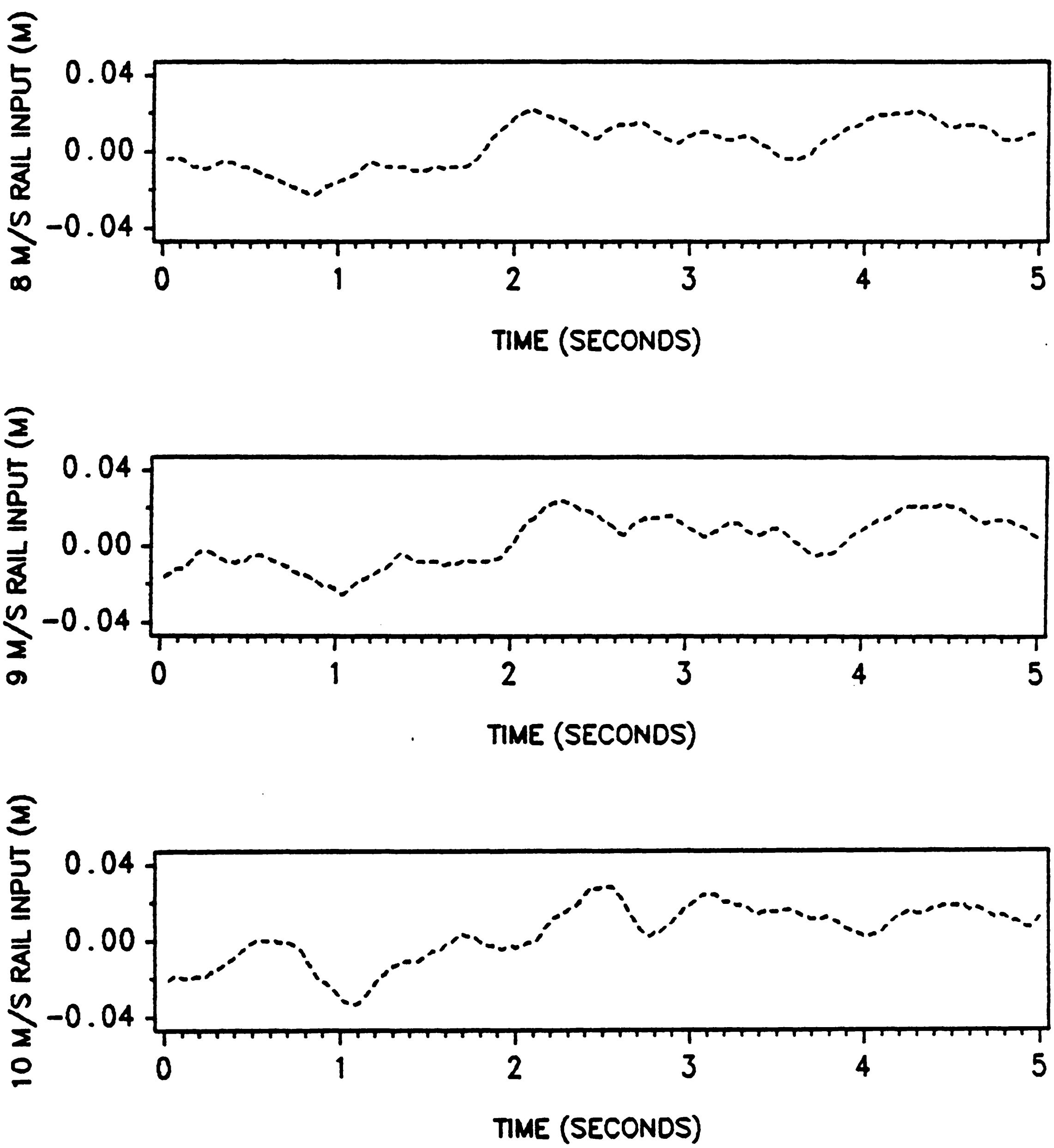

Figure 16. The Rail Input Plots at Various Speeds

As vehicle speeds exceed the resonant speed, it becomes necessary to reduce integration time step to preserve accuracy of the analysis. This reduction does not drastically alter the statistical properties of the random track inputs.

5.3 *Increased Carload Effects*

As stated in the introduction, a major goal in the development of the TERM railcar model is to provide a tool useful in evaluating the dynamic effects of heavy carloads. In this section, the differences in carbody roll responses and contact force magnitudes are examined for 70-ton and 100-ton vehicles.

The procedure for analysis is straightforward; first the carbody responses and wheel/rail contact forces are evaluated for a 70-ton vehicle, then similar analyses are performed for 100-ton vehicle. In each case, the appropriate physical parameters are taken from the tables of Chapter 4. Auxiliary dampers, discussed in the preceding section, are implemented for these analysis cases.

Figure 17 on page 90 illustrates the effects of increased payload on maximum carbody roll at various speeds. As expected, the 100-ton car experiences more severe roll responses near the critical speed than does the 70-ton car. At low vehicle speeds, the maximum carbody roll responses are nearly identical. As speeds approach 7.5 m/s, however, the effects of heavier loading are quite apparent. Once the simulation velocities exceed 8 m/s, both models experience a reduction in roll response as they depart from the critical speed.

The vertical translation responses of Figure 18 on page 91 show less drastic results. The absolute midspan deflection is consistently greater at all speeds for the 100-ton carload. In both cases, the vertical translations are not drastically altered at the critical speed.

A comparison of wheel/rail contact forces for the 70-ton and 100-ton vehicles reveals significant increases for the heavier loading. Figure 19 compares contact force responses for each vehicle operating at 7 m/s. The average static contact forces are obviously higher for the 100-ton load. Variations from the static level are far more prevalent for the 100-ton loading throughout the simulation. Instantaneous contact forces exceed 1.7 times their nominal static value for the 100-ton vehicle. For the 70-ton vehicle, contact force peaks rarely exceed 1.25 times their static value.

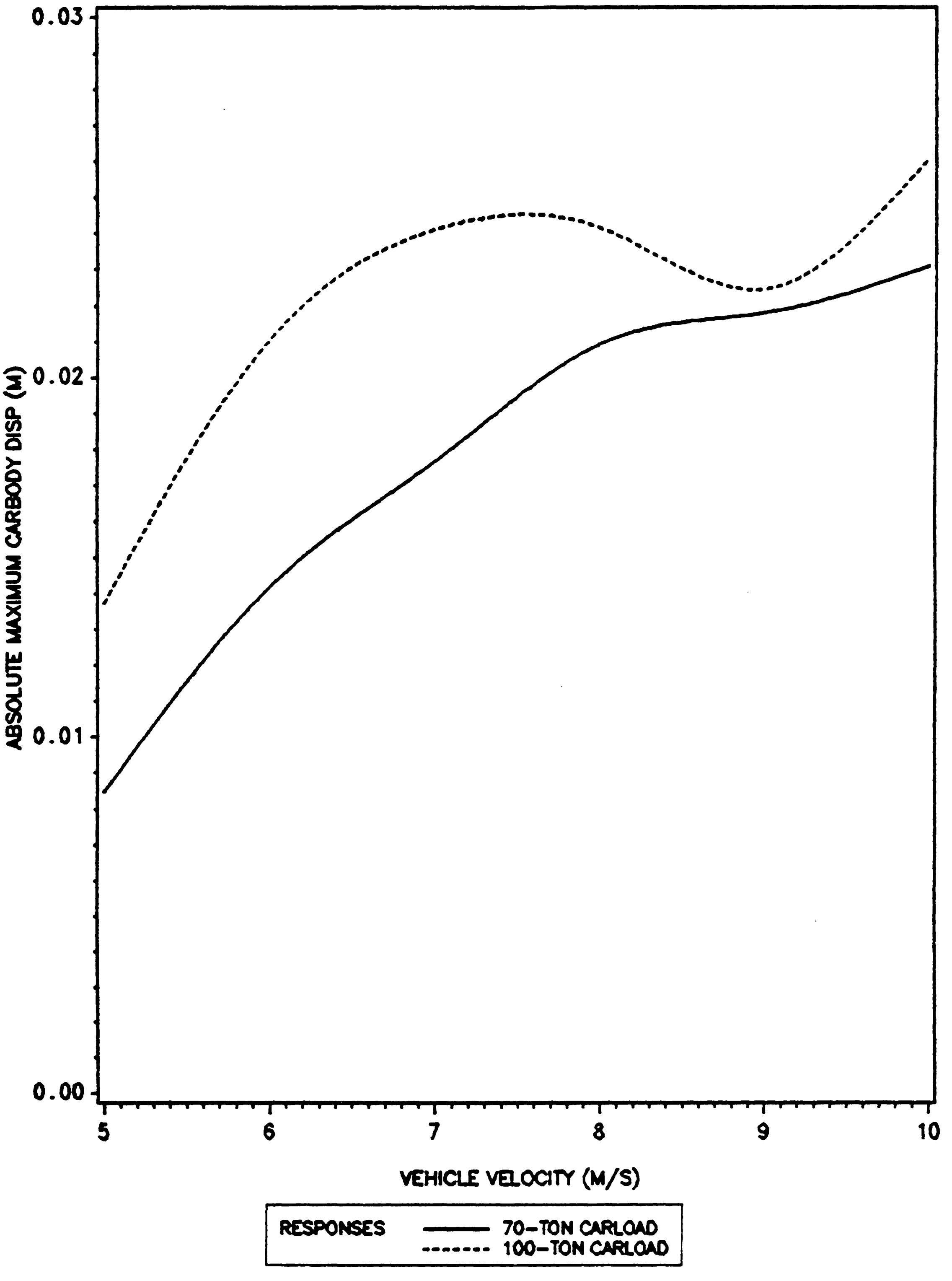

Figure 17. Maximum Carbody Roll Responses

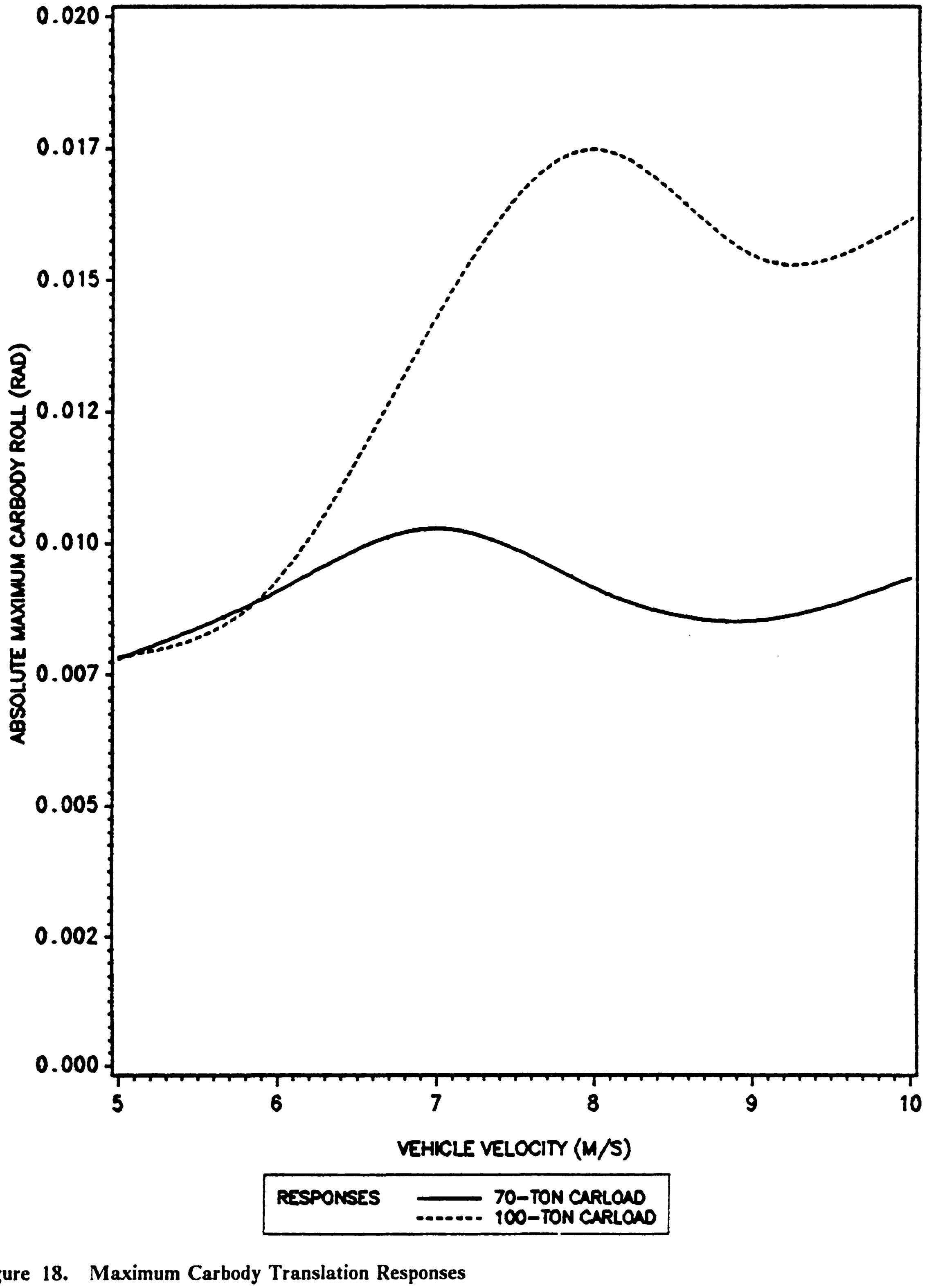

Figure 18. Maximum Carbody Translation Responses

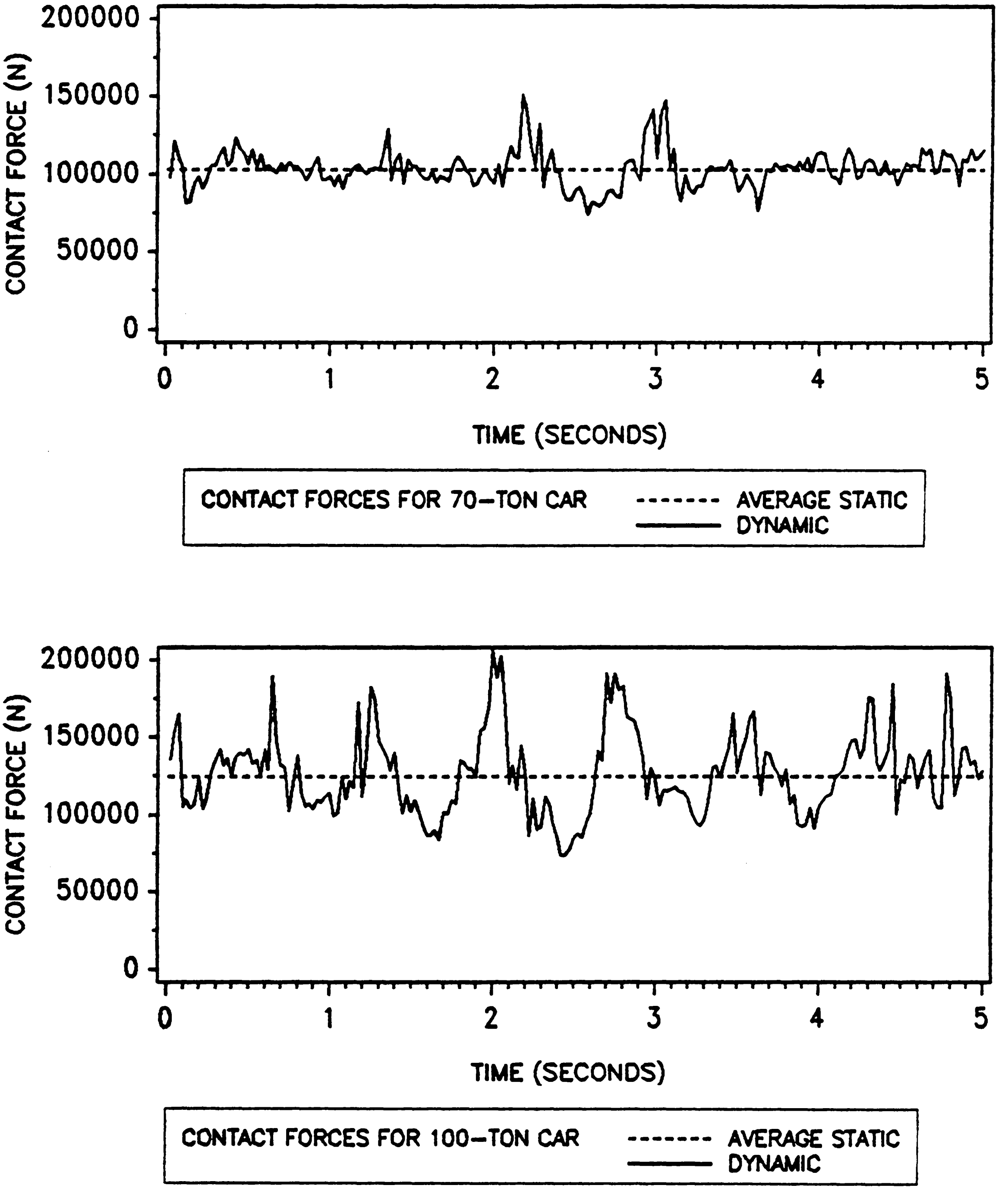

Figure 19. Wheel/Rail Contact Force Responses

Chapter 6

Summary, Conclusions, and Recommendations

6.1 Summary

A 28-degree-of-freedom vertical dynamic railcar model has been developed to evaluate railcar responses to vertical and crosslevel track inputs. The carbody features first-mode vertical bending deflection and first-mode torsional rotation freedoms. In addition, the model utilizes multiple-component trucks, and it is equipped for simulations on random and discrete vertical inputs.

Responses of the model have been compared to those of the AAR Flexible Carbody Model under similar operating conditions on harmonic track inputs. The results of this comparison are consistent for the two models, particularly when modelling dissimilarities are considered. The effects of nonlinear auxiliary viscous suspension damping have been evaluated for simulations on random inputs representative of class 4 track. The results of these simulations indicate that the dampers are very effective in reducing carbody roll and vertical responses, particularly near the critical vehicle speed. Carbody response comparisons for 70-ton and 100-ton vehicles on class 4 track show excessive roll responses for the heavier vehicle near the critical speed. A comparison of wheel/rail contact forces for the two models reveals dynamic forces that exceed 1.7 times their nominal static value for the 100-ton vehicle. For the 70-ton vehicle, dynamic contact forces did not exceed 1.3 times their static value.

6.2 Conclusions

The 28-degree-of-freedom dynamic model is effective in simulating the responses of a railcar to vertical and crosslevel inputs. Its capability to simulate operation on discrete and random track is a valuable feature of the model. Responses to harmonic rail inputs compared favorably to those of the AAR Flexible Carbody Model.

Results on random vertical inputs illustrate the effectiveness of auxiliary suspension stabilizers at low speeds. The model predicted more severe responses for 100-ton vehicles than for 70-ton vehicles under similar operating conditions, a conclusion supported by physical evidence in the field.

6.3 Recommendations

The 28-degree-of-freedom model has been tested extensively under various operating conditions. The results of these simulations indicate that the model will be an effective tool in evaluating railcar responses at other operating conditions. The model can successfully evaluate the effectiveness of alternative suspension and stabilizer designs.

Simulations that did not include auxiliary viscous damping occasionally encountered stability problems near critical speeds, as indicated by the results of Section 5.2. For this reason, the existing damping values should be increased for more realistic simulations when auxiliary damping is not employed.

References

Ahlbeck, D. R., "Preliminary Evaluation of Rail Vehicles by Computer Simulation", ***High Speed Ground Transportation Journal***, Fall 1977.

Atkinson, K., ***Elementary Numerical Analysis***, John Wiley and Sons, Inc., 1985.

Cook, R. D., and Young, W., C., ***Advanced Mechanics of Materials***, Macmillan Publishing Company, 1985.

Cooperrider, N. K., Law, E. H., and Fries, R. H., "Freight Car Dynamics: Field Test Results and Comparison With Theory", ***DOT-OS-40018***, June 1981.

Fries, R. H., and Coffey, B. M., "A State-Space Approach to the Synthesis of Random Vertical and Crosslevel Rail Irregularities", ***1987 ASME Winter Annual Meeting***, 1987.

Heller, R., Tuten, J. M., Kadala, P. S., and Law, E. H., "Analog and Digital Computer Simulation of Coulomb Friction", ***FRA-OR&D-77/08***, August 30, 1977.

Hussain, S. M. A., Singh, S. P., and Garg, V. K., "Evaluation of Non-linear Flexible Carbody Vehicle (Freight Car) Model", ***Association of American Railroads Report No. R-435***, August 1980.

Lovejoy, V. D., "Dynamics and Control of a Planar Truss Actuator", M.S. Thesis, ***Virginia Polytechnic Institute***, April, 1987.

Platin, B. E., Beaman, J. J., Hedrick, J. K., and Wormley, D. N., "Computational Methods to Predict Railcar Response to Track Cross-Level Variations", *FRA-OR&D-76-293*, September, 1976.

Roark, R. C., and Young, W. C., ***Formulas for Stress and Strain, Fifth Edition***, McGraw-Hill, Inc., 1975.

Sankar, T. S., and Samaha, M., "Stochastic Analysis of Railroad Freight Car Rocking Problem", ***Mechanical Systems and Signal Processing***, 1987.

Thomson, W. T., ***Theory of Vibrations With Applications, Second Edition***, Prentice-Hall, Inc., 1981.

Tse, Y. T., "Method of Analysis for the Dynamic Behavior of a Flexible Railroad Freight Car", M.S. Thesis, ***Illinois Institute of Technology***, December, 1974.

White, R. C., "Car Rocking Simulations of Two 140-Ton Railroad Flatcars", ***Acorn Associates Publication***, 1987.

Willis, T., and Shum, K., "A Nonlinear Mathematical Model of the Dynamics of a Railroad Freight Car/Freight Element", ***American Society of Mechanical Engineers Paper No. 76-DE-42***, April 1976.

Appendix A

Program Listing

```
***** THIS PROGRAM CALCULATES THE VERTICAL DISPLACEMENTS   *****
***** AND VELOCITIES OF A 28-DEGREE-OF-FREEDOM RAILCAR     *****
***** MODEL. PROGRAM CALLS SUBROUTINES RAILF1, RAILR1,     *****
***** RAILF2, AND RAILR2 (SUPSCRIPTS DENOTE FRONT OR REAR, *****
***** AND 1 REFERS TO FRONT TRUCK, 2 REFERS TO REAR TRUCK) *****
***** TO CALCULATE THE RAIL DISPLACEMENT AND VELOCITY      *****
***** INPUTS, AND IT CALLS SUBROUTINE RK4SYS TO IMPLEMENT  *****
***** A FOURTH-ORDER RUNGE-KUTTA SYSTEM SOLVER FOR THE     *****
***** SOLUTION OF THE DIFFERENTIAL EQUATIONS OF MOTION.    *****
*****                                                      *****

      IMPLICIT REAL*8(A - H, O - Z)
      REAL*8 KCB, KCENL, KCENV, KRAIL
      REAL*8 KSUS, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      REAL*8 Y(56), Y0(56), YMAX(56), TMAX(56), COEFF(15)
      REAL*8 LCAR, LCB, LCEN, LCGB, LCGCAR, LRATIO
      REAL*8 IW, IB, ICP, ICR, IS, LW, LB, LS, MW, MB, MC, MS
      INTEGER*4 PERCEN
      COMMON/SPRING/ KCB, KCENL, KCENV, KRAIL
      COMMON/SUSPEN/ KSUSL, KSUSV, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      COMMON/DAMPER/ CRAIL, CSUSV1, CSUSV2, C0
      COMMON/COULOM/ FRICTL, FRICTV
      COMMON/XLENT1/ LW, LB, LCAR, LCEN, LS, DCEN, WCAR
      COMMON/XLENT2/ LCB, LCGB, LCGCAR
      COMMON/WEIGHT/ MW, MB, MC, MS
      COMMON/XMOMNT/ IW, IB, ICR, ICP, IS
      COMMON/PROPER/ EI, GJR, LRATIO
      COMMON/CLEAR/ CCB, CSLC, CSLT, CSVC, CSVT
      COMMON/SEEDS/ SEEDF1, SEEDR1, SEEDF2, SEEDR2
      COMMON/TIME/ DT, VEL, X0
```

```
      COMMON/LAGNUM/ NUM1, NUM2, NUM3
      COMMON/TRUK1D/ ZRFR1, ZRFL1, ZRRR1, ZRRL1
      COMMON/TRUK2D/ ZRFR2, ZRFL2, ZRRR2, ZRRL2
      COMMON/TRUK1V/ ZRFR1D, ZRFL1D, ZRRR1D, ZRRL1D
      COMMON/TRUK2V/ ZRFR2D, ZRFL2D, ZRRR2D, ZRRL2D
      COMMON/COEFFS/ COEFF, STIFF, TWIST
      COMMON/REMEM1/ ZDDOT, QDDOT
      COMMON/REMEM2/ XR1OLD, XL1OLD, XR2OLD, XL2OLD
      COMMON/REMEM3/ ZR1OLD, ZL1OLD, ZR2OLD, ZL2OLD
      COMMON/PRINT1/ FWRFR1, FWRFL1, FWRRR1, FWRRL1
      COMMON/PRINT2/ FCBR1, FCBL1, FCBR2, FCBL2
      COMMON/PRINT3/ QFSR1L, QFSL1L, QFSR2L, QFSL2L
      COMMON/PRINT4/ QFSR1V, QFSL1V, QFSR2V, QFSL2V

************************************************************************
***** THE SYSTEM VARIABLES
************************************************************************

***** Y(1)  THE CARBODY'S VERTICAL POSITION
***** Y(2)  THE CARBODY'S VERTICAL VELOCITY
***** Y(3)  THE CARBODY'S LATERAL POSITION
***** Y(4)  THE CARBODY'S LATERAL VELOCITY
***** Y(5)  THE CARBODY'S ANGULAR PITCH POSITION
***** Y(6)  THE CARBODY'S ANGULAR PITCH VELOCITY
***** Y(7)  THE CARBODY'S ANGULAR ROLL POSITION
***** Y(8)  THE CARBODY'S ANGULAR ROLL VELOCITY
***** Y(9)  THE CARBODY'S MIDSPAN VERTICAL BENDING DEFLECTION
***** Y(10)  THE CARBODY'S MIDSPAN VERTICAL BENDING VELOCITY
***** Y(11)  THE CARBODY'S END TORSIONAL DISPLACEMENT
***** Y(12)  THE CARBODY'S END TORSIONAL VELOCITY
***** Y(13)  THE FIRST BOLSTER'S VERTICAL POSITION
***** Y(14)  THE FIRST BOLSTER'S VERTICAL VELOCITY
***** Y(15)  THE SECOND BOLSTER'S VERTICAL POSITION
***** Y(16)  THE SECOND BOLSTER'S VERTICAL VELOCITY
***** Y(17)  THE FIRST BOLSTER'S LATERAL POSITION
***** Y(18)  THE FIRST BOLSTER'S LATERAL VELOCITY
***** Y(19)  THE SECOND BOLSTER'S LATERAL POSITION
***** Y(20)  THE SECOND BOLSTER'S LATERAL VELOCITY
***** Y(21)  THE FIRST BOLSTER'S ANGULAR ROLL POSITION
***** Y(22)  THE FIRST BOLSTER'S ANGULAR ROLL VELOCITY
***** Y(23)  THE SECOND BOLSTER'S ANGULAR ROLL POSITION
***** Y(24)  THE SECOND BOLSTER'S ANGULAR ROLL VELOCITY
***** Y(25)  THE FIRST RIGHT SIDEFRAME'S VERTICAL POSITION
***** Y(26)  THE FIRST RIGHT SIDEFRAME'S VERTICAL VELOCITY
***** Y(27)  THE FIRST LEFT SIDEFRAME'S VERTICAL POSITION
***** Y(28)  THE FIRST LEFT SIDEFRAME'S VERTICAL VELOCITY
***** Y(29)  THE SECOND RIGHT SIDEFRAME'S VERTICAL POSITION
***** Y(30)  THE SECOND RIGHT SIDEFRAME'S VERTICAL VELOCITY
***** Y(31)  THE SECOND LEFT SIDEFRAME'S VERTICAL POSITION
***** Y(32)  THE SECOND LEFT SIDEFRAME'S VERTICAL VELOCITY
***** Y(33)  THE FIRST RIGHT SIDEFRAME'S ANGULAR PITCH POSITION
***** Y(34)  THE FIRST RIGHT SIDEFRAME'S ANGULAR PITCH VELOCITY
***** Y(35)  THE FIRST LEFT SIDEFRAME'S ANGULAR PITCH POSITION
***** Y(36)  THE FIRST LEFT SIDEFRAME'S ANGULAR PITCH VELOCITY
***** Y(37)  THE SECOND RIGHT SIDEFRAME'S ANGULAR PITCH POSITION
***** Y(38)  THE SECOND RIGHT SIDEFRAME'S ANGULAR PITCH VELOCITY
```

```
***** Y(39)  THE SECOND LEFT SIDEFRAME'S ANGULAR PITCH POSITION
***** Y(40)  THE SECOND LEFT SIDEFRAME'S ANGULAR PITCH VELOCITY
***** Y(41)  THE FIRST FRONT WHEELSET'S VERTICAL POSITION
***** Y(42)  THE FIRST FRONT WHEELSET'S VERTICAL VELOCITY
***** Y(43)  THE FIRST REAR WHEELSET'S VERTICAL POSITION
***** Y(44)  THE FIRST REAR WHEELSET'S VERTICAL VELOCITY
***** Y(45)  THE SECOND FRONT WHEELSET'S VERTICAL POSITION
***** Y(46)  THE SECOND FRONT WHEELSET'S VERTICAL VELOCITY
***** Y(47)  THE SECOND REAR WHEELSET'S VERTICAL POSITION
***** Y(48)  THE SECOND REAR WHEELSET'S VERTICAL VELOCITY
***** Y(49)  THE FIRST FRONT WHEELSET'S ANGULAR ROLL POSITION
***** Y(50)  THE FIRST FRONT WHEELSET'S ANGULAR ROLL VELOCITY
***** Y(51)  THE FIRST REAR WHEELSET'S ANGULAR ROLL POSITION
***** Y(52)  THE FIRST REAR WHEELSET'S ANGULAR ROLL VELOCITY
***** Y(53)  THE SECOND FRONT WHEELSET'S ANGULAR ROLL POSITION
***** Y(54)  THE SECOND FRONT WHEELSET'S ANGULAR ROLL VELOCITY
***** Y(55)  THE SECOND REAR WHEELSET'S ANGULAR ROLL POSITION
***** Y(56)  THE SECOND REAR WHEELSET'S ANGULAR ROLL VELOCITY

*******************************************************************
***** THE OUTPUT FILES
*******************************************************************

***** ZCAR IS THE TOTAL CARBODY VERTICAL DISPL. AND VEL. OUTPUT FILE
*    OPEN(1,FILE='ZCAR')

***** ROLCAR IS THE FRONT END CARBODY ROLL DISP. AND VEL. OUTPUT FILE
*    OPEN(2,FILE='ROLCAR')

***** ZBOL IS THE FRONT BOLSTER VERTICAL DISPL. AND VEL. OUTPUT FILE
*    OPEN(3,FILE='ZBOL')

***** ROLBOL IS THE FRONT BOLSTER ROLL DISP. AND VEL. OUTPUT FILE
*    OPEN(4,FILE='ROLBOL')

***** ZRAILR IS THE RIGHT RAIL DISPLACEMENTS INPUT FILE
*    OPEN(7,FILE='ZRAILR')

***** ZRAILL IS THE LEFT RAIL DISPLACEMENTS INPUT FILE
*    OPEN(8,FILE='ZRAILL')

***** FORCE IS THE CONTACT FORCE AND SIDEBEARER FORCE OUTPUT FILE
*    OPEN(9,FILE='FORCE')

***** YMAX IS THE ABSOLUTE MAXIMUM DISPL. AND VELOCITY OUTPUT FILE
*    OPEN(10,FILE='YMAX')

***** ZC IS THE BEAM VERTICAL DISPLACEMENT OUTPUT FILE
*    OPEN(11,FILE='ZC')

***** ZCDOT IS THE BEAM VERTICAL VELOCITY OUTPUT FILE
*    OPEN(12,FILE='ZCDOT')

***** XC IS THE BEAM LATERAL DISPLACEMENT OUTPUT FILE
*    OPEN(13,FILE='XC')
```

```
***** XCDOT IS THE BEAM LATERAL VELOCITY OUTPUT FILE
*     OPEN(14,FILE='XCDOT')

***** THETP IS THE BEAM ANGULAR PITCH OUTPUT FILE
*     OPEN(15,FILE='THETP')

***** THETPD IS THE BEAM ANGULAR PITCH VELOCITY OUTPUT FILE
*     OPEN(16,FILE='THETPD')

***** THETR IS THE BEAM ANGULAR ROLL OUTPUT FILE
*     OPEN(17,FILE='THETR')

***** THETRD IS THE BEAM ANGULAR ROLL VELOCITY OUTPUT FILE
*     OPEN(18,FILE='THETRD')

***** DELTA IS THE BEAM DEFLECTION OUTPUT FILE
*     OPEN(19,FILE='DELTA')

***** DELTAD IS THE BEAM VELOCITY OUTPUT FILE
*     OPEN(20,FILE='DELTAD')

***** PHI IS THE TORSIONAL DISPLACEMENT OUTPUT FILE
*     OPEN(21,FILE='PHI')

***** PHID IS THE TORSIONAL VELOCITY OUTPUT FILE
*     OPEN(22,FILE='PHID')

***** ZB1 IS THE FIRST BOLSTER VERTICAL DISPLACEMENT OUTPUT FILE
*     OPEN(23,FILE='ZB1')

***** ZB1D IS THE FIRST BOLSTER VERTICAL VELOCITY OUTPUT FILE
*     OPEN(24,FILE='ZB1D')

***** ZB2 IS THE SECOND BOLSTER VERTICAL DISPLACEMENT OUTPUT FILE
*     OPEN(25,FILE='ZB2')

***** ZB2D IS THE SECOND BOLSTER VERTICAL VELOCITY OUTPUT FILE
*     OPEN(26,FILE='ZB2D')

***** XB1 IS THE FIRST BOLSTER LATERAL DISPLACEMENT OUTPUT FILE
*     OPEN(27,FILE='XB1')

***** XB1D IS THE FIRST BOLSTER LATERAL VELOCITY OUTPUT FILE
*     OPEN(28,FILE='XB1D')

***** XB2 IS THE SECOND BOLSTER LATERAL DISPLACEMENT OUTPUT FILE
*     OPEN(29,FILE='XB2')

***** XB2D IS THE SECOND BOLSTER LATERAL VELOCITY OUTPUT FILE
*     OPEN(30,FILE='XB2D')

***** BETA1 IS THE FIRST BOLSTER ANGULAR ROLL OUTPUT FILE
*     OPEN(31,FILE='BETA1')

***** BETA1D IS THE FIRST BOLSTER ANGULAR ROLL VELOCITY OUTPUT FILE
*     OPEN(32,FILE='BETA1D')
```

```
***** BETA2 IS THE SECOND BOLSTER ANGULAR ROLL OUTPUT FILE
*     OPEN(33,FILE='BETA2')

***** BETA2D IS THE SECOND BOLSTER ANGULAR ROLL VELOCITY OUTPUT FILE
*     OPEN(34,FILE='BETA2D')

***** ZSR1 IS THE FIRST RIGHT SIDEFRAME VERT. DISPLACEMENT OUTPUT FILE
*     OPEN(35,FILE='ZSR1')

***** ZSR1D IS THE FIRST RIGHT SIDEFRAME VERT. VELOCITY OUTPUT FILE
*     OPEN(36,FILE='ZSR1D')

***** ZSL1 IS THE FIRST LEFT SIDEFRAME VERT. DISPLACEMENT OUTPUT FILE
*     OPEN(37,FILE='ZSL1')

***** ZSL1D IS THE FIRST LEFT SIDEFRAME VERT. VELOCITY OUTPUT FILE
*     OPEN(38,FILE='ZSL1D')

***** ZSR2 IS THE SECOND RIGHT SIDEFRAME VERT. DISPLACEMENT OUTPUT FILE
*     OPEN(39,FILE='ZSR2')

***** ZSR2D IS THE SECOND RIGHT SIDEFRAME VERT. VELOCITY OUTPUT FILE
*     OPEN(40,FILE='ZSR2D')

***** ZSL2 IS THE SECOND LEFT SIDEFRAME VERT. DISPLACEMENT OUTPUT FILE
*     OPEN(41,FILE='ZSL2')

***** ZSL2D IS THE SECOND LEFT SIDEFRAME VERT. VELOCITY OUTPUT FILE
*     OPEN(42,FILE='ZSL2D')

***** SIGR1 IS THE FIRST RIGHT SIDEFRAME PITCH OUTPUT FILE
*     OPEN(43,FILE='SIGR1')

***** SIGR1D IS THE FIRST RIGHT SIDEFRAME PITCH VELOCITY OUTPUT FILE
*     OPEN(44,FILE='SIGR1D')

***** SIGL1 IS THE FIRST LEFT SIDEFRAME PITCH OUTPUT FILE
*     OPEN(45,FILE='SIGL1')

***** SIGL1D IS THE FIRST LEFT SIDEFRAME PITCH VELOCITY OUTPUT FILE
*     OPEN(46,FILE='SIGL1D')

***** SIGR2 IS THE SECOND RIGHT SIDEFRAME PITCH OUTPUT FILE
*     OPEN(47,FILE='SIGR2')

***** SIGR2D IS THE SECOND RIGHT SIDEFRAME PITCH VELOCITY OUTPUT FILE
*     OPEN(48,FILE='SIGR2D')

***** SIGL2 IS THE SECOND LEFT SIDEFRAME PITCH OUTPUT FILE
*     OPEN(49,FILE='SIGL2')

***** SIGL2D IS THE SECOND LEFT SIDEFRAME PITCH VELOCITY OUTPUT FILE
*     OPEN(50,FILE='SIGL2D')
```

```
***** ZAF1 IS THE FIRST FRONT WHEELSET VERTICAL DISPLACEMENT OUTPUT FILE
*     OPEN(51,FILE='ZAF1')

***** ZAF1D IS THE FIRST FRONT WHEELSET VERTICAL VELOCITY OUTPUT FILE
*     OPEN(52,FILE='ZAF1D')

***** ZAR1 IS THE FIRST REAR WHEELSET VERTICAL DISPLACEMENT OUTPUT FILE
*     OPEN(53,FILE='ZAR1')

***** ZAR1D IS THE FIRST REAR WHEELSET VERTICAL VELOCITY OUTPUT FILE
*     OPEN(54,FILE='ZAR1D')

***** ZAF2 IS THE SECOND FRONT WHEELSET VERTICAL DISPLACEMENT FILE
*     OPEN(55,FILE='ZAF2')

***** ZAF2D IS THE SECOND FRONT WHEELSET VERTICAL VELOCITY OUTPUT FILE
*     OPEN(56,FILE='ZAF2D')

***** ZAR2 IS THE SECOND REAR WHEELSET VERTICAL DISPLACEMENT OUTPUT FILE
*     OPEN(57,FILE='ZAR2')

***** ZAR2D IS THE SECOND REAR WHEELSET VERTICAL VELOCITY OUTPUT FILE
*     OPEN(58,FILE='ZAR2D')

***** ALFF1 IS THE FIRST FRONT WHEELSET ANGULAR ROLL OUTPUT FILE
*     OPEN(59,FILE='ALFF1')

***** ALFF1D IS THE FIRST FRONT WHEELSET ANGULAR ROLL VELOCITY FILE
*     OPEN(60,FILE='ALFF1D')

***** ALFR1 IS THE FIRST REAR WHEELSET ANGULAR ROLL OUTPUT FILE
*     OPEN(61,FILE='ALFR1')

***** ALFR1D IS THE FIRST REAR WHEELSET ANGULAR ROLL VELOCITY FILE
*     OPEN(62,FILE='ALFR1D')

***** ALFF2 IS THE SECOND FRONT WHEELSET ANGULAR ROLL OUTPUT FILE
*     OPEN(63,FILE='ALFF2')

***** ALFF2D IS THE SECOND FRONT WHEELSET ANGULAR ROLL VELOCITY FILE
*     OPEN(64,FILE='ALFF2D')

***** ALFR2 IS THE SECOND REAR WHEELSET ANGULAR ROLL OUTPUT FILE
*     OPEN(65,FILE='ALFR2')

***** ALFR2D IS THE SECOND REAR WHEELSET ANGULAR ROLL VELOCITY FILE
*     OPEN(66,FILE='ALFR2D')

********************************************************************
***** THE SYSTEM PROPERTIES
********************************************************************

***** MC *TOTAL CARBODY MASS IS 92,915 KG
***** MB *MASS OF BOLSTERS IS 619 KG EACH
***** MS *MASS OF SIDEFRAMES IS 447 KG EACH
***** MW *MASS OF WHEELSETS IS 1427 KG EACH
```

```
***** LCAR *LENGTH OF CARBODY IS 14.75 M
***** WCAR *WIDTH OF CARBODY IS 3.50 M
***** LB *LENGTH OF BOLSTER (BETWEEN SUSPENSION GROUPS) IS 2.01 M
***** LS *LENGTH OF SIDEFRAME (BETWEEN SUSPENSION GROUPS) IS 1.78 M
***** LW *LENGTH OF WHEELSETS IS 1.43 M
***** LCEN *LONGITUDINAL CARBODY DISTANCE BETWEEN CENTERPLATES IS 11.89
***** DCEN *DIAMETER OF CENTERPLATES IS .356 M
***** LCB *LATERAL DISTANCE BETWEEN SIDEBEARERS IS 1.27 M
***** LCGCAR *VERTICAL DISTANCE FROM CENTERPLATE TO CARBODY CG IS 1.75 M
***** LCGB *VERTICAL DISTANCE FROM CENTERPLATE TO BOLSTER CG IS .12 M
***** ICP *PITCH MOMENT OF INERTIA FOR CARBODY IS 1.69 E6 KG*M**2
***** ICR *ROLL MOMENT OF INERTIA FOR CARBODY IS 94850 KG*M**2
***** IB *ROLL MOMENT OF INERTIA FOR BOLSTERS IS 249 KG*M**2
***** IS *PITCH MOMENT OF INERTIA FOR SIDEFRAMES IS 158 KG*M**2
***** IW *ROLL MOMENT OF INERTIA FOR WHEELSETS IS 850 KG*M**2
***** EI *CARBODY BENDING STIFFNESS IS 3 *10**8 N*M**2/RAD
***** GJR *CARBODY TORSIONAL STIFFNESS IS 2 * 10**8
***** CCB *CLEARANCE FOR SIDEBEARING SPRINGS IS .01 M
***** CSLC *CLEARANCE OF SUS. LATERAL COMPRESSIVE IS .005 M
***** CSLT *CLEARANCE OF SUS. LATERAL TENSILE IS .005 M
***** CSVC *CLEARANCE OF SUS. VERTICAL COMPRESSIVE IS .05 M
***** CSVT *CLEARANCE OF SUS. VERTICAL TENSILE IS .05 M
***** KCB *SIDEBEARING STIFFNESS IS 6.27 * 10**8 NEWTONS/METER
***** KCENL *LATERAL CENTERPLATE STIFFNESS IS 3.71 * 10**8 NEWTONS/METER
***** KCENV *VERTICAL CENTERPLATE STIFFNESS IS 3.75 * 10**7 NEWTONS/METE
***** KSUSL *LATERAL SUSPENSION STIFFNESS IS 1.96 * 10**6 NEWTONS/METER
***** KSUSV *VERTICAL SUSPENSION STIFFNESS IS 3.93 * 10**6 NEWTONS/METER
***** KSUSCL *LATERAL CLEARANCE STIFFNESS IS 1.46 * 10**7 NEWTONS/METER
***** KSUSCV *VERTICAL CLEARANCE STIFFNESS IS 5.62 * 10**7 NEWTONS/METER
***** KSUSFL *LATERAL FRICTION STIFFNESS IS 1 * 10**9 NEWTONS/METER
***** KSUSFV *VERTICAL FRICTION STIFFNESS IS 1 * 10**9 NEWTONS/METER
***** KRAIL *RAIL STIFFNESS IS 1.75 * 10**7 NEWTONS/METER
***** CRAIL *RAIL DAMPING COEFF. IS 22500 NEWTONS*SEC/METER
***** CSUSV1 *REGION 1 STUCKI COEFF. IS 3.5 * 10**6 NEWTONS*SEC/METER
***** CSUSV2 *REGION 2 STUCKI COEFF. IS 146000 NEWTONS*SEC/METER
***** C0 *STUCKI SLOPE INTERCEPT IS 89000 NEWTONS
***** FRICTL *THE LATERAL COULOMB FRICTION COEFF. IS 17,500 NEWTONS
***** FRICTV *THE VERTICAL COULOMB FRICTION COEFF. IS 17,500 NEWTONS

      MC = 92915.D0
      MB = 619.D0
      MS = 447.D0
      MW = 1427.D0
      LCAR = 14.75D0
      WCAR = 3.50D0
      LB = 2.01D0
      LS = 1.78D0
      LW = 1.43D0
      LCEN = 11.89D0
      LRATIO = LCEN / LCAR
      DCEN = .356D0
      LCB = 1.27D0
      LCGCAR = 1.75D0
      LCGB = .12D0
      ICP = (MC * LCAR**2) / 12.D0
      ICR = (MC * WCAR**2) / 12.D0
```

```
      IB = 249.D0
      IS = 158.D0
      IW = 850.D0
      EI = 3.00D08
      GJR = 2.00D08
      CCB = .0065D0
      CSLC = .005D0
      CSLT = .005D0
      CSVC = .05D0
      CSVT = .05D0
      KCB = 6.27D08
      KCENL = 3.71D08
      KCENV = 3.75D07
      KSUSL = 1.96D06
      KSUSV = 3.93D06
      KSUSCL = 1.46D07
      KSUSCV = 5.62D07
      KSUSFL = 1.D09
      KSUSFV = 1.D09
      KRAIL = 1.75D07
      CRAIL = 22500.D0
      CSUSV1 = 3.50D06
      CSUSV2 = 146000.D0
      C0 = 89000.D0
      FRICTL = 55000.D0
      FRICTV = 55000.D0

*********************************************************************
***** STARTUP INFORMATION
*********************************************************************

***** ALL SEED VALUES FOR RAIL INPUT PROGRAMS ARE SET TO .08786786

      SEEDF1 = .08786786D0
      SEEDR1 = SEEDF1
      SEEDF2 = SEEDF1
      SEEDR2 = SEEDF1

***** INPUT THE VEHICLE SPEED IN METERS/SECOND

      WRITE(*, *) 'ENTER VEHICLE SPEED (M/S)'
      READ(5, *) VEL

***** INPUT THE TIME INCREMENT STEP IN SECONDS

      WRITE(*, *) 'ENTER INCREMENT SIZE'
      READ(5, *) H

***** INPUT THE RUN DURATION TIME IN SECONDS

      WRITE(*, *) 'ENTER RUN TIME'
      READ(5, *) XT

***** INPUT THE PRINTING FACTOR

      WRITE(*, *) 'ENTER OUTPUT FACTOR'
```

```
      READ(5, *) NFACT
      NUMOUT = NFACT

      DT = H / 2.D0
      X0 = 0.D0
      XNEW = X0
      XTNEW = XNEW + H

***** COMPUTE THE TIME LAGS BETWEEN THE FOUR RAIL INPUTS

      TLAG1 = (LS + LCEN) / VEL
      TLAG2 = LCEN / VEL
      TLAG3 = LS / VEL
      NUM1 = TLAG1 / DT
      NUM2 = TLAG2 / DT
      NUM3 = TLAG3 / DT

***** THE FOLLOWING LINES ARE USED TO NOTIFY USER OF PROGRESS DURING RUN

      TNUM = (XT - X0) / H
      TENPCT = TNUM / 10.D0
      PCTNUM = TENPCT
      PERCEN = 10

**********************************************************************
***** THE INITIAL CONDITIONS FOR THE SYSTEM:
**********************************************************************

      Y0(1) = -.061D0
      Y0(2) = 0.D0
      Y0(3) = 0.D0
      Y0(4) = 0.D0
      Y0(5) = 0.D0
      Y0(6) = 0.D0
      Y0(7) = 0.D0
      Y0(8) = 0.D0
      Y0(9) = -.01D0
      Y0(10) = 0.D0
      Y0(11) = 0.D0
      Y0(12) = 0.D0
      Y0(13) = -.060D0
      Y0(14) = 0.D0
      Y0(15) = -.060D0
      Y0(16) = 0.D0
      Y0(17) = 0.D0
      Y0(18) = 0.D0
      Y0(19) = 0.D0
      Y0(20) = 0.D0
      Y0(21) = 0.D0
      Y0(22) = 0.D0
      Y0(23) = 0.D0
      Y0(24) = 0.D0
      Y0(25) = -.011D0
      Y0(26) = 0.D0
      Y0(27) = -.011D0
      Y0(28) = 0.D0
```

```
Y0(29) = -.011D0
Y0(30) = 0.D0
Y0(31) = -.011D0
Y0(32) = 0.D0
Y0(33) = 0.D0
Y0(34) = 0.D0
Y0(35) = 0.D0
Y0(36) = 0.D0
Y0(37) = 0.D0
Y0(38) = 0.D0
Y0(39) = 0.D0
Y0(40) = 0.D0
Y0(41) = -.008D0
Y0(42) = 0.D0
Y0(43) = -.008D0
Y0(44) = 0.D0
Y0(45) = -.008D0
Y0(46) = 0.D0
Y0(47) = -.008D0
Y0(48) = 0.D0
Y0(49) = 0.D0
Y0(50) = 0.D0
Y0(51) = 0.D0
Y0(52) = 0.D0
Y0(53) = 0.D0
Y0(54) = 0.D0
Y0(55) = 0.D0
Y0(56) = 0.D0

Y(1) = Y0(1)
Y(2) = Y0(2)
Y(3) = Y0(3)
Y(4) = Y0(4)
Y(5) = Y0(5)
Y(6) = Y0(6)
Y(7) = Y0(7)
Y(8) = Y0(8)
Y(9) = Y0(9)
Y(10) = Y0(10)
Y(11) = Y0(11)
Y(12) = Y0(12)
Y(13) = Y0(13)
Y(14) = Y0(14)
Y(15) = Y0(15)
Y(16) = Y0(16)
Y(17) = Y0(17)
Y(18) = Y0(18)
Y(19) = Y0(19)
Y(20) = Y0(20)
Y(21) = Y0(21)
Y(22) = Y0(22)
Y(23) = Y0(23)
Y(24) = Y0(24)
Y(25) = Y0(25)
Y(26) = Y0(26)
Y(27) = Y0(27)
```

```
      Y(28) = Y0(28)
      Y(29) = Y0(29)
      Y(30) = Y0(30)
      Y(31) = Y0(31)
      Y(32) = Y0(32)
      Y(33) = Y0(33)
      Y(34) = Y0(34)
      Y(35) = Y0(35)
      Y(36) = Y0(36)
      Y(37) = Y0(37)
      Y(38) = Y0(38)
      Y(39) = Y0(39)
      Y(40) = Y0(40)
      Y(41) = Y0(41)
      Y(42) = Y0(42)
      Y(43) = Y0(43)
      Y(44) = Y0(44)
      Y(45) = Y0(45)
      Y(46) = Y0(46)
      Y(47) = Y0(47)
      Y(48) = Y0(48)
      Y(49) = Y0(49)
      Y(50) = Y0(50)
      Y(51) = Y0(51)
      Y(52) = Y0(52)
      Y(53) = Y0(53)
      Y(54) = Y0(54)
      Y(55) = Y0(55)
      Y(56) = Y0(56)

C     WRITE(11, *) X0, Y(1)
C     WRITE(12, *) X0, Y(2)
C     WRITE(13, *) X0, Y(3)
C     WRITE(14, *) X0, Y(4)
C     WRITE(15, *) X0, Y(5)
C     WRITE(16, *) X0, Y(6)
C     WRITE(17, *) X0, Y(7)
C     WRITE(18, *) X0, Y(8)
C     WRITE(19, *) X0, Y(9)
C     WRITE(20, *) X0, Y(10)
C     WRITE(21, *) X0, Y(11)
C     WRITE(22, *) X0, Y(12)
C     WRITE(23, *) X0, Y(13)
C     WRITE(24, *) X0, Y(14)
C     WRITE(25, *) X0, Y(15)
C     WRITE(26, *) X0, Y(16)
C     WRITE(27, *) X0, Y(17)
C     WRITE(28, *) X0, Y(18)
C     WRITE(29, *) X0, Y(19)
C     WRITE(30, *) X0, Y(20)
C     WRITE(31, *) X0, Y(21)
C     WRITE(32, *) X0, Y(22)
C     WRITE(33, *) X0, Y(23)
C     WRITE(34, *) X0, Y(24)
C     WRITE(35, *) X0, Y(25)
C     WRITE(36, *) X0, Y(26)
```

```
C     WRITE(37, *) X0, Y(27)
C     WRITE(38, *) X0, Y(28)
C     WRITE(39, *) X0, Y(29)
C     WRITE(40, *) X0, Y(30)
C     WRITE(41, *) X0, Y(31)
C     WRITE(42, *) X0, Y(32)
C     WRITE(43, *) X0, Y(33)
C     WRITE(44, *) X0, Y(34)
C     WRITE(45, *) X0, Y(35)
C     WRITE(46, *) X0, Y(36)
C     WRITE(47, *) X0, Y(37)
C     WRITE(48, *) X0, Y(38)
C     WRITE(49, *) X0, Y(39)
C     WRITE(50, *) X0, Y(40)
C     WRITE(51, *) X0, Y(41)
C     WRITE(52, *) X0, Y(42)
C     WRITE(53, *) X0, Y(43)
C     WRITE(54, *) X0, Y(44)
C     WRITE(55, *) X0, Y(45)
C     WRITE(56, *) X0, Y(46)
C     WRITE(57, *) X0, Y(47)
C     WRITE(58, *) X0, Y(48)
C     WRITE(59, *) X0, Y(49)
C     WRITE(60, *) X0, Y(50)
C     WRITE(61, *) X0, Y(51)
C     WRITE(62, *) X0, Y(52)
C     WRITE(63, *) X0, Y(53)
C     WRITE(64, *) X0, Y(54)
C     WRITE(65, *) X0, Y(55)
C     WRITE(66, *) X0, Y(56)

      NCOUNT = 0

10    IF(XTNEW .LE. XT) THEN
        CALL RK4SYS(XNEW, Y, 56, XTNEW, H)

***** THE FOLLOWING BLOCK WILL RETAIN THE MAXIMUM SYSTEM
***** VALUES AND TIMES OF OCCURENCE FOR FUTURE PRINTING

        IF(NCOUNT .EQ. 0) THEN
          DO 200 I = 1, 56
            TMAX(I) = 0.D0
            YMAX(I) = 0.D0
200       CONTINUE
        ENDIF

        DO 300 I = 1, 56
          IF(DABS((Y(I) - Y0(I))) .GE. DABS(YMAX(I))) THEN
            TMAX(I) = XTNEW
            YMAX(I) = Y(I)- Y0(I)
          ENDIF
300     CONTINUE

***** THE PROGRAM CONTINUES

        NCOUNT = NCOUNT + 1
```

```
      IF(NCOUNT .EQ. NUMOUT) THEN

**** Z IS THE TOTAL MIDSPAN DEFLECTION
**** ZDOT IS THE TOTAL MIDSPAN VELOCITY
**** R IS THE TOTAL FRONT END ROLL ROTATION
**** RDOT IS THE TOTAL FRONT END ROLL VELOCITY

        Z = Y(1) + Y(9)
        ZDOT = Y(2) + Y(10)
        R = Y(7) + Y(11)
        RDOT = Y(8) + Y(12)

        WRITE(1, 101) XTNEW, Z, ZDOT, ZRFR1
        WRITE(2, 101) XTNEW, R, RDOT, ZRFR1
        WRITE(3, 101) XTNEW, Y(13), Y(14), ZRFR1
        WRITE(4, 101) XTNEW, Y(21), Y(22), ZRFR1
        WRITE(7, 102) XTNEW, ZRFR1, ZRRR1, ZRFR2, ZRRR2
        WRITE(8, 102) XTNEW, ZRFL1, ZRRL1, ZRFL2, ZRRL2
        WRITE(9, 103) XTNEW, FWRFR1, FCBR1

C       WRITE(11, *) XTNEW, Y(1)
C       WRITE(12, *) XTNEW, Y(2)
C       WRITE(13, *) XTNEW, Y(3)
C       WRITE(14, *) XTNEW, Y(4)
C       WRITE(15, *) XTNEW, Y(5)
C       WRITE(16, *) XTNEW, Y(6)
C       WRITE(17, *) XTNEW, Y(7)
C       WRITE(18, *) XTNEW, Y(8)
C       WRITE(19, *) XTNEW, Y(9)
C       WRITE(20, *) XTNEW, Y(10)
C       WRITE(21, *) XTNEW, Y(11)
C       WRITE(22, *) XTNEW, Y(12)
C       WRITE(23, *) XTNEW, Y(13)
C       WRITE(24, *) XTNEW, Y(14)
C       WRITE(25, *) XTNEW, Y(15)
C       WRITE(26, *) XTNEW, Y(16)
C       WRITE(27, *) XTNEW, Y(17)
C       WRITE(28, *) XTNEW, Y(18)
C       WRITE(29, *) XTNEW, Y(19)
C       WRITE(30, *) XTNEW, Y(20)
C       WRITE(31, *) XTNEW, Y(21)
C       WRITE(32, *) XTNEW, Y(22)
C       WRITE(33, *) XTNEW, Y(23)
C       WRITE(34, *) XTNEW, Y(24)
C       WRITE(35, *) XTNEW, Y(25)
C       WRITE(36, *) XTNEW, Y(26)
C       WRITE(37, *) XTNEW, Y(27)
C       WRITE(38, *) XTNEW, Y(28)
C       WRITE(39, *) XTNEW, Y(29)
C       WRITE(40, *) XTNEW, Y(30)
C       WRITE(41, *) XTNEW, Y(31)
C       WRITE(42, *) XTNEW, Y(32)
C       WRITE(43, *) XTNEW, Y(33)
C       WRITE(44, *) XTNEW, Y(34)
C       WRITE(45, *) XTNEW, Y(35)
C       WRITE(46, *) XTNEW, Y(36)
```

```
C        WRITE(47, *) XTNEW, Y(37)
C        WRITE(48, *) XTNEW, Y(38)
C        WRITE(49, *) XTNEW, Y(39)
C        WRITE(50, *) XTNEW, Y(40)
C        WRITE(51, *) XTNEW, Y(41)
C        WRITE(52, *) XTNEW, Y(42)
C        WRITE(53, *) XTNEW, Y(43)
C        WRITE(54, *) XTNEW, Y(44)
C        WRITE(55, *) XTNEW, Y(45)
C        WRITE(56, *) XTNEW, Y(46)
C        WRITE(57, *) XTNEW, Y(47)
C        WRITE(58, *) XTNEW, Y(48)
C        WRITE(59, *) XTNEW, Y(49)
C        WRITE(60, *) XTNEW, Y(50)
C        WRITE(61, *) XTNEW, Y(51)
C        WRITE(62, *) XTNEW, Y(52)
C        WRITE(63, *) XTNEW, Y(53)
C        WRITE(64, *) XTNEW, Y(54)
C        WRITE(65, *) XTNEW, Y(55)
C        WRITE(66, *) XTNEW, Y(56)

           NUMOUT = NUMOUT + NFACT
         ENDIF

         IF(NCOUNT .GE. PCTNUM) THEN

           WRITE(*, *) 'PERCENT OF RUN COMPLETE', PERCEN

           PERCEN = PERCEN + 10
           PCTNUM = PCTNUM + TENPCT
         ENDIF

         XNEW = XTNEW
         XTNEW = XNEW + H
         GOTO 10
       ELSE
         WRITE(*, *) 'PROGRAM FINISHED!'

         DO 400 I = 1, 56
           WRITE(10, 104) I, YMAX(I), TMAX(I)
400      CONTINUE

         STOP
       ENDIF

101   FORMAT(' ', F7.3, 3X, F14.6, 3X, F14.6, 3X, F14.6)
102   FORMAT(' ',F7.4, 1X, F9.3, 1X, F9.3, 1X, F9.3, 1X, F9.3)
103   FORMAT(' ',F7.4, 1X, F14.6, 1X, F14.6)
104   FORMAT(' ','YMAX(',I2,') = ', F8.5, ' AT TIME = ', F8.5)

       END


       SUBROUTINE RK4SYS(X, Y, N, XT, H)
       IMPLICIT REAL*8(A - H, O - Z)
       REAL*8 Y(N), V1(56), V2(56), V3(56), V4(56), YTEMP(56)
```

```
      REAL*8 KCB, KCENL, KCENV, KRAIL
      REAL*8 KSUS, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      REAL*8 COEFF(15)
      REAL*8 LCAR, LCB, LCEN, LCGB, LCGCAR, LRATIO
      REAL*8 IW, IB, ICP, ICR, IS, LW, LB, LS, MW, MB, MC, MS
      COMMON/SPRING/ KCB, KCENL, KCENV, KRAIL
      COMMON/SUSPEN/ KSUSL, KSUSV, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      COMMON/DAMPER/ CRAIL, CSUSV1, CSUSV2, C0
      COMMON/COULOM/ FRICTL, FRICTV
      COMMON/XLENT1/ LW, LB, LCAR, LCEN, LS, DCEN, WCAR
      COMMON/XLENT2/ LCB, LCGB, LCGCAR
      COMMON/WEIGHT/ MW, MB, MC, MS
      COMMON/XMOMNT/ IW, IB, ICR, ICP, IS
      COMMON/PROPER/ EI, GJR, LRATIO
      COMMON/CLEAR/ CCB, CSLC, CSLT, CSVC, CSVT
      COMMON/SEEDS/ SEEDF1, SEEDR1, SEEDF2, SEEDR2
      COMMON/TIME/ DT, VEL, X0
      COMMON/LAGNUM/ NUM1, NUM2, NUM3
      COMMON/TRUK1D/ ZRFR1, ZRFL1, ZRRR1, ZRRL1
      COMMON/TRUK2D/ ZRFR2, ZRFL2, ZRRR2, ZRRL2
      COMMON/TRUK1V/ ZRFR1D, ZRFL1D, ZRRR1D, ZRRL1D
      COMMON/TRUK2V/ ZRFR2D, ZRFL2D, ZRRR2D, ZRRL2D
      COMMON/COEFFS/ COEFF, STIFF, TWIST
      COMMON/REMEM1/ ZDDOT, QDDOT
      COMMON/REMEM2/ XR1OLD, XL1OLD, XR2OLD, XL2OLD
      COMMON/REMEM3/ ZR1OLD, ZL1OLD, ZR2OLD, ZL2OLD
      COMMON/PRINT1/ FWRFR1, FWRFL1, FWRRR1, FWRRL1
      COMMON/PRINT2/ FCBR1, FCBL1, FCBR2, FCBL2
      COMMON/PRINT3/ QFSR1L, QFSL1L, QFSR2L, QFSL2L
      COMMON/PRINT4/ QFSR1V, QFSL1V, QFSR2V, QFSL2V


10     IF(X .GE. XT) RETURN
      IF(X + H .GT. XT) H = XT - X
      NCALL = 1
      CALL FUNCT(X, Y, V1, N, NCALL)

      DO 100 I = 1, N
100      YTEMP(I) = Y(I) + (H * V1(I)) / 2.D0
      XTEMP = X + (H / 2.D0)
      NCALL = 2
      CALL FUNCT(XTEMP, YTEMP, V2, N, NCALL)

      DO 200 I = 1, N
200      YTEMP(I) = Y(I) + (H * V2(I)) / 2.D0
      NCALL = 3
      CALL FUNCT(XTEMP, YTEMP, V3, N, NCALL)

      DO 300 I = 1, N
300      YTEMP(I) = Y(I) + (H * V3(I))
      X = X + H
      NCALL = 4
      CALL FUNCT(X, YTEMP, V4, N, NCALL)

      DO 400 I = 1, N
        Y(I) = Y(I) + H * (V1(I) + 2.D0 * (V2(I) + V3(I))
```

```
     2        + V4(I)) / 6.D0
400   CONTINUE

      GOTO 10
      END


      SUBROUTINE FUNCT(X, Y, V, N, NCALL)
      IMPLICIT REAL*8(A - H, O - Z)
      REAL*8 V(56), Y(56)
      REAL*8 KCB, KCENL, KCENV, KRAIL
      REAL*8 KSUS, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      REAL*8 COEFF(15)
      REAL*8 LCAR, LCB, LCEN, LCGB, LCGCAR, LRATIO
      REAL*8 IW, IB, ICP, ICR, IS, LW, LB, LS, MW, MB, MC, MS
      COMMON/SPRING/ KCB, KCENL, KCENV, KRAIL
      COMMON/SUSPEN/ KSUSL, KSUSV, KSUSCL, KSUSCV, KSUSFL, KSUSFV
      COMMON/DAMPER/ CRAIL, CSUSV1, CSUSV2, C0
      COMMON/COULOM/ FRICTL, FRICTV
      COMMON/XLENT1/ LW, LB, LCAR, LCEN, LS, DCEN, WCAR
      COMMON/XLENT2/ LCB, LCGB, LCGCAR
      COMMON/WEIGHT/ MW, MB, MC, MS
      COMMON/XMOMNT/ IW, IB, ICR, ICP, IS
      COMMON/PROPER/ EI, GJR, LRATIO
      COMMON/CLEAR/ CCB, CSLC, CSLT, CSVC, CSVT
      COMMON/SEEDS/ SEEDF1, SEEDR1, SEEDF2, SEEDR2
      COMMON/TIME/ DT, VEL, X0
      COMMON/LAGNUM/ NUM1, NUM2, NUM3
      COMMON/TRUK1D/ ZRFR1, ZRFL1, ZRRR1, ZRRL1
      COMMON/TRUK2D/ ZRFR2, ZRFL2, ZRRR2, ZRRL2
      COMMON/TRUK1V/ ZRFR1D, ZRFL1D, ZRRR1D, ZRRL1D
      COMMON/TRUK2V/ ZRFR2D, ZRFL2D, ZRRR2D, ZRRL2D
      COMMON/COEFFS/ COEFF, STIFF, TWIST
      COMMON/REMEM1/ ZDDOT, QDDOT
      COMMON/REMEM2/ XR1OLD, XL1OLD, XR2OLD, XL2OLD
      COMMON/REMEM3/ ZR1OLD, ZL1OLD, ZR2OLD, ZL2OLD
      COMMON/PRINT1/ FWRFR1, FWRFL1, FWRRR1, FWRRL1
      COMMON/PRINT2/ FCBR1, FCBL1, FCBR2, FCBL2
      COMMON/PRINT3/ QFSR1L, QFSL1L, QFSR2L, QFSL2L
      COMMON/PRINT4/ QFSR1V, QFSL1V, QFSR2V, QFSL2V

*************************************************************************
***** THE FOLLOWING LOOP IS EXECUTED ONLY ONCE TO INITIALIZE
***** VALUES FOR THE EIGHT RAIL INPUTS IF X = X0.
***** THIS LOOP ALSO INITIALIZES VALUES OF COEFFICIENTS TO
***** BE USED IN THE EQUATIONS THAT FOLLOW:
*************************************************************************

      IF(X .EQ. X0) THEN
         DO 100 I = 1, NUM1
            CALL RAILF1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF1,
     2           ZRFL1, ZRFR1, ZFL1DD, ZFR1DD, ZRFL1D, ZRFR1D)
100      CONTINUE
         DO 200 I = 1, NUM2
            CALL RAILR1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR1,
```

```
   2        ZRRL1, ZRRR1, ZRL1DD, ZRR1DD, ZRRL1D, ZRRR1D)
200     CONTINUE
      DO 300 I = 1, NUM3
        CALL RAILF2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF2,
   2        ZRFL2, ZRFR2, ZFL2DD, ZFR2DD, ZRFL2D, ZRFR2D)
300     CONTINUE
      CALL RAILR2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR2,
   2        ZRRL2, ZRRR2, ZRL2DD, ZRR2DD, ZRRL2D, ZRRR2D)

      GRAV = 9.8055D0
      PI = DACOS(-1.D0)

      COEFF(1) = 1.D0 / MC
      COEFF(2) = .4765D0 * MC
      COEFF(3) = 1.D0 / ICP
      COEFF(4) = 1.D0 / ICR
      COEFF(5) = 1.D0 / MB
      COEFF(6) = 1.D0 / IB
      COEFF(7) = 1.D0 / MS
      COEFF(8) = 1.D0 / IS
      COEFF(9) = 1.D0 / MW
      COEFF(10) = 1.D0 / IW
      COEFF(11) = (12.D0 * LRATIO**2) / ICR
      STIFF =  (.4120D0 * EI * PI**4) / (LRATIO**4 * LCAR**3)
      TWIST =  (GJR) / (LRATIO**2 * LCAR)

      ZDDOT = 0.D0
      QDDOT = 0.D0
      XR1OLD = 0.D0
      XL1OLD = 0.D0
      XR2OLD = 0.D0
      XL2OLD = 0.D0
      ZR1OLD = 0.D0
      ZL1OLD = 0.D0
      ZR2OLD = 0.D0
      ZL2OLD = 0.D0

      WRITE(*, *) 'BEGINNING EXECUTION'

    ENDIF

*********************************************************************

    V(1) = Y(2)
    V(3) = Y(4)
    V(5) = Y(6)
    V(7) = Y(8)
    V(9) = Y(10)
    V(11) = Y(12)
    V(13) = Y(14)
    V(15) = Y(16)
    V(17) = Y(18)
    V(19) = Y(20)
    V(21) = Y(22)
    V(23) = Y(24)
    V(25) = Y(26)
```

```
      V(27) = Y(28)
      V(29) = Y(30)
      V(31) = Y(32)
      V(33) = Y(34)
      V(35) = Y(36)
      V(37) = Y(38)
      V(39) = Y(40)
      V(41) = Y(42)
      V(43) = Y(44)
      V(45) = Y(46)
      V(47) = Y(48)
      V(49) = Y(50)
      V(51) = Y(52)
      V(53) = Y(54)
      V(55) = Y(56)

      IF(NCALL .EQ. 1) THEN

      GOTO 400

      ENDIF

      IF(NCALL .EQ. 2) THEN

        CALL RAILF1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF1,
     2          ZRFL1, ZRFR1, ZFL1DD, ZFR1DD, ZRFL1D, ZRFR1D)
        CALL RAILR1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR1,
     2          ZRRL1, ZRRR1, ZRL1DD, ZRR1DD, ZRRL1D, ZRRR1D)
        CALL RAILF2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF2,
     2          ZRFL2, ZRFR2, ZFL2DD, ZFR2DD, ZRFL2D, ZRFR2D)
        CALL RAILR2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR2,
     2          ZRRL2, ZRRR2, ZRL2DD, ZRR2DD, ZRRL2D, ZRRR2D)

      GOTO 400

      ENDIF

      IF(NCALL .EQ. 3) THEN

      GOTO 400

      ENDIF

      IF(NCALL .EQ. 4) THEN

        CALL RAILF1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF1,
     2          ZRFL1, ZRFR1, ZFL1DD, ZFR1DD, ZRFL1D, ZRFR1D)
        CALL RAILR1(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR1,
     2          ZRRL1, ZRRR1, ZRL1DD, ZRR1DD, ZRRL1D, ZRRR1D)
        CALL RAILF2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDF2,
     2          ZRFL2, ZRFR2, ZFL2DD, ZFR2DD, ZRFL2D, ZRFR2D)
        CALL RAILR2(VEL, 4, DT, .77D0, 0.D0, 11.89D0, SEEDR2,
     2          ZRRL2, ZRRR2, ZRL2DD, ZRR2DD, ZRRL2D, ZRRR2D)

      GOTO 400
```

```
      ENDIF
******************************************************************
******************************************************************
*****                        <--|  /\   /\   /\   /\   /\    |\
***** FORCE CALCULATIONS        |_/  \ /  \ /  \ /  \ /  \____|\
*****                                 \/   \/   \/   \/       |\
******************************************************************
******************************************************************


***** FOR FUTURE USE IN THE FORCE CALCULATIONS, THE COMBINED ROLL
***** AND TORSIONAL ROTATATIONS OF THE CARBODY WILL BE COMBINED
***** INTO FRONT AND REAR ROTATIONS ROLLF AND ROLLR

400   ROLLF = Y(7) + (Y(11) / 2.D0)
      ROLLR = Y(7) - (Y(11) / 2.D0)



******************************************************************
***** FIRST, THE VERTICAL CENTERPLATE FORCES ARE CALCULATED,
***** USING A LOGIC CHECK TO DETERMINE WHETHER THESE SPRINGS
***** ARE IN CONTACT.
******************************************************************

***** CALCULATE THE RELATIVE VERTICAL DISPLACEMENTS BETWEEN THE CARBODY
***** AND BOLSTERS AT THE CENTERPLATE EDGES:

      DCER1V = ((Y(13) + ((DCEN / 2.D0) * Y(21)))
     2     - (Y(1) + ((LCEN / 2.D0) * Y(5)) + ((DCEN / 2.D0) * ROLLF)))

      DCEL1V = ((Y(13) - ((DCEN / 2.D0) * Y(21)))
     2     - (Y(1) + ((LCEN / 2.D0) * Y(5)) - ((DCEN / 2.D0) * ROLLF)))

      DCER2V = ((Y(15) + ((DCEN / 2.D0) * Y(23)))
     2     - (Y(1) - ((LCEN / 2.D0) * Y(5)) + ((DCEN / 2.D0) * ROLLR)))

      DCEL2V = ((Y(15) - ((DCEN / 2.D0) * Y(23)))
     2     - (Y(1) - ((LCEN / 2.D0) * Y(5)) - ((DCEN / 2.D0) * ROLLR)))

***** DETERMINE IF THESE RELATIVE DISPLACEMENTS INDICATE SEPARATION.
***** IF NO SEPARATION OCCURS, CALCULATE THE CENTERPLATE FORCES:

         IF(DCER1V .GT. 0.D0) THEN
            FCER1V = KCENV * DCER1V
         ELSE
            FCER1V = 0.D0
         ENDIF

         IF(DCEL1V .GT. 0.D0) THEN
            FCEL1V = KCENV * DCEL1V
         ELSE
            FCEL1V = 0.D0
         ENDIF
```

```
      IF(DCER2V .GT. 0.D0) THEN
         FCER2V = KCENV * DCER2V
      ELSE
         FCER2V = 0.D0
      ENDIF

      IF(DCEL2V .GT. 0.D0) THEN
         FCEL2V = KCENV * DCEL2V
      ELSE
         FCEL2V = 0.D0
      ENDIF

*********************************************************************
***** NEXT, THE LATERAL CENTERPLATE FORCES ARE CALCULATED,
***** DEPENDING ON WHETHER THE VERTICAL CENTERPLATE CONTACT
***** FORCES CALCULATED ABOVE EXIST. (IF CENTERPLATE CONTACT EXISTS)
*********************************************************************

***** CALCULATE THE RELATIVE LATERAL DISPLACEMENTS BETWEEN THE CARBODY
***** AND BOLSTERS AT THE CENTERPLATE EDGES:

     DCER1L = (Y(17) + (LCGB * Y(21))) - (Y(3) - (LCGCAR * ROLLF))
     DCEL1L = (Y(17) + (LCGB * Y(21))) - (Y(3) - (LCGCAR * ROLLF))
     DCER2L = (Y(19) + (LCGB * Y(23))) - (Y(3) - (LCGCAR * ROLLR))
     DCEL2L = (Y(19) + (LCGB * Y(23))) - (Y(3) - (LCGCAR * ROLLR))

     IF(FCER1V .GT. 0.D0) THEN
        FCER1L = KCENL * DCER1L
     ELSE
        FCER1L = 0.D0
     ENDIF

     IF(FCEL1V .GT. 0.D0) THEN
        FCEL1L = KCENL * DCEL1L
     ELSE
        FCEL1L = 0.D0
     ENDIF

     IF(FCER2V .GT. 0.D0) THEN
        FCER2L = KCENL * DCER2L
     ELSE
        FCER2L = 0.D0
     ENDIF

     IF(FCEL2V .GT. 0.D0) THEN
        FCEL2L = KCENL * DCEL2L
     ELSE
        FCEL2L = 0.D0
     ENDIF

*********************************************************************
***** NEXT, THE FORCES IN THE CARBODY/BOLSTER CLEARANCE SPRINGS
***** (SIDEBEARING SPRINGS) ARE CALCULATED, USING A LOGIC CHECK
***** TO DETERMINE WHETHER THE CLEARANCE SPRINGS ARE IN CONTACT
*********************************************************************
```

```
***** CALCULATE THE RELATIVE VERTICAL DISPLACEMENTS BETWEEN THE CARBODY
***** AND BOLSTERS AT THE SIDEBEARING SPRING LOCATIONS:

      DCBR1 = ((Y(13) + ((LCB / 2.D0) * Y(21)))
     2      - (Y(1) + ((LCEN / 2.D0) * Y(5)) + ((LCB / 2.D0) * ROLLF)))

      DCBL1 = ((Y(13) - ((LCB / 2.D0) * Y(21)))
     2      - (Y(1) + ((LCEN / 2.D0) * Y(5)) - ((LCB / 2.D0) * ROLLF)))

      DCBR2 = ((Y(15) + ((LCB / 2.D0) * Y(23)))
     2      - (Y(1) - ((LCEN / 2.D0) * Y(5)) + ((LCB / 2.D0) * ROLLR)))

      DCBL2 = ((Y(15) - ((LCB / 2.D0) * Y(23)))
     2      - (Y(1) - ((LCEN / 2.D0) * Y(5)) - ((LCB / 2.D0) * ROLLR)))

***** DETERMINE IF THESE RELATIVE DISPLACEMENTS INDICATE CONTACT BETWEEN
***** THE CARBODY AND SIDEBEARING SPRINGS. IF CONTACT OCCURS,
***** CALCULATE THE FORCES:

        IF(DCBR1 .GT. CCB) THEN
           FCBR1 = KCB * DCBR1
        ELSE
           FCBR1 = 0.D0
        ENDIF

        IF(DCBL1 .GT. CCB) THEN
           FCBL1 = KCB * DCBL1
        ELSE
           FCBL1 = 0.D0
        ENDIF

        IF(DCBR2 .GT. CCB) THEN
           FCBR2 = KCB * DCBR2
        ELSE
           FCBR2 = 0.D0
        ENDIF

        IF(DCBL2 .GT. CCB) THEN
           FCBL2 = KCB * DCBL2
        ELSE
           FCBL2 = 0.D0
        ENDIF

*********************************************************************
*****
***** LATERAL SUSPENSION GROUPS
*****
*********************************************************************

*********************************************************************
***** NEXT, THE <LATERAL> BOLSTER/SIDEFRAME FORCES (LATERAL SUSPENSION
***** FORCES) ARE CALCULATED, INCLUDING THE LINEAR SPRING FORCES
***** (DENOTED BY PLAIN 'F' PRESCRIPTS), THE NON-LINEAR CLEARANCE SPRING
***** FORCES (DENOTED BY 'CF' PRESCRIPTS), AND FINALLY THE
```

```
***** COULOMB FRICTION FORCES. (DENOTED BY 'QF' PRESCRIPTS)
*********************************************************************

***** CALCULATE THE RELATIVE LATERAL DISPLACEMENTS BETWEEN BOLSTERS
***** AND SIDEFRAMES:

      DBSR1L = -Y(17)
      DBSL1L = -Y(17)
      DBSR2L = -Y(19)
      DBSL2L = -Y(19)

***** CALCULATE THE RELATIVE LATERAL VELOCITIES BETWEEN BOLSTERS
***** AND SIDEFRAMES:

      VBSR1L = -Y(18)
      VBSL1L = -Y(18)
      VBSR2L = -Y(20)
      VBSL2L = -Y(20)

***** CALCULATE THE LATERAL SUSPENSION LINEAR SPRING FORCES:

      FSR1L = KSUSL * DBSR1L
      FSL1L = KSUSL * DBSL1L
      FSR2L = KSUSL * DBSR2L
      FSL2L = KSUSL * DBSL2L

***** DETERMINE IF THE LATERAL RELATIVE DISPLACEMENTS INDICATE CONTACT
***** AT THE NON-LINEAR SUSPENSION CLEARANCE SPRINGS.  IF CONTACT
***** OCCURS, CALCULATE THE LATERAL SUSPENSION CLEARANCE FORCES:

         IF(DBSR1L .GT. CSLT .OR. DBSR1L .LT. -CSLC) THEN
            CFSR1L = KSUSCL * DBSR1L
         ELSE
            CFSR1L = 0.D0
         ENDIF

         IF(DBSL1L .GT. CSLT .OR. DBSL1L .LT. -CSLC) THEN
            CFSL1L = KSUSCL * DBSL1L
         ELSE
            CFSL1L = 0.D0
         ENDIF

         IF(DBSR2L .GT. CSLT .OR. DBSR2L .LT. -CSLC) THEN
            CFSR2L = KSUSCL * DBSR2L
         ELSE
            CFSR2L = 0.D0
         ENDIF

         IF(DBSL2L .GT. CSLT .OR. DBSL2L .LT. -CSLC) THEN
            CFSL2L = KSUSCL * DBSL2L
         ELSE
            CFSL2L = 0.D0
         ENDIF

***** CALCULATE THE LATERAL FRICTION SPRING FORCES, REQUIRING A
***** MID-SEPARATION DISPLACEMENT VALUE, X.  VALUES OF X FROM
```

```
***** THE PREVIOUS STEP ARE DENOTED AS 'XOLD' VALUES, WHICH ARE
***** USED TO DETERMINE THE VALUES AT THE CURRENT STEP, 'XNEW'.

***** FIRST, CALCULATE TEMPORARY FRICTION SPRING FORCES USING
***** 'XOLD' VALUES:

      FTEMR1 = KSUSFL * (DBSR1L + XR1OLD)
      FTEML1 = KSUSFL * (DBSL1L + XL1OLD)
      FTEMR2 = KSUSFL * (DBSR2L + XR2OLD)
      FTEML2 = KSUSFL * (DBSL2L + XL2OLD)

      IF(VBSR1L .GT. 0.D0) THEN
        SINR1L = 1.D0
      ELSE
        SINR1L = -1.D0
      ENDIF

      IF(VBSL1L .GT. 0.D0) THEN
        SINL1L = 1.D0
      ELSE
        SINL1L = -1.D0
      ENDIF

      IF(VBSR2L .GT. 0.D0) THEN
        SINR2L = 1.D0
      ELSE
        SINR2L = -1.D0
      ENDIF

      IF(VBSL2L .GT. 0.D0) THEN
        SINL2L = 1.D0
      ELSE
        SINL2L = -1.D0
      ENDIF

         IF(DABS(FTEMR1) .LT. FRICTL) THEN
           XR1NEW = XR1OLD
         ELSE
           XR1NEW = ((SINR1L * FRICTL) / KSUSFL) - DBSR1L
         ENDIF

***** NOW THAT A VALUE FOR 'XNEW' IS KNOWN, THE LATERAL COULOMB
***** DAMPING FORCE, DENOTED BY PRESCRIPT 'QF', IS CALCULATED:

      QFSR1L = KSUSFL * (DBSR1L + XR1NEW)
      XR1OLD = XR1NEW

         IF(DABS(FTEML1) .LT. FRICTL) THEN
           XL1NEW = XL1OLD
         ELSE
           XL1NEW = ((SINL1L * FRICTL) / KSUSFL) - DBSL1L
         ENDIF

      QFSL1L = KSUSFL * (DBSL1L + XL1NEW)
      XL1OLD = XL1NEW
```

```
      IF(DABS(FTEMR2) .LT. FRICTL) THEN
         XR2NEW = XR2OLD
      ELSE
         XR2NEW = ((SINR2L * FRICTL) / KSUSFL) - DBSR2L
      ENDIF

   QFSR2L = KSUSFL * (DBSR2L + XR2NEW)
   XR2OLD = XR2NEW

      IF(DABS(FTEML2) .LT. FRICTL) THEN
         XL2NEW = XL2OLD
      ELSE
         XL2NEW = ((SINL2L * FRICTL) / KSUSFL) - DBSL2L
      ENDIF

   QFSL2L = KSUSFL * (DBSL2L + XL2NEW)
   XL2OLD = XL2NEW

****************************************************************
*****
***** VERTICAL SUSPENSION GROUPS
*****
****************************************************************

****************************************************************
***** NEXT, THE <VERTICAL> BOLSTER/SIDEFRAME FORCES (VERTICAL SUSPENSION
***** FORCES) ARE CALCULATED, INCLUDING THE LINEAR SPRING FORCES
***** (DENOTED BY PLAIN 'F' PRESCRIPTS), THE VISCOUS DAMPING FORCES
***** (DENOTED BY 'DF' PRESCRIPTS), THE NON-LINEAR CLEARANCE SPRING
***** FORCES (DENOTED BY 'CF' PRESCRIPTS), AND FINALLY THE
***** COULOMB FRICTION FORCES. (DENOTED BY 'QF' PRESCRIPTS)
****************************************************************

***** CALCULATE THE RELATIVE VERTICAL DISPLACEMENTS BETWEEN BOLSTERS
***** AND SIDEFRAMES:

   DBSR1V = Y(25) - (Y(13) + ((LB / 2.D0) * Y(21)))
   DBSL1V = Y(27) - (Y(13) - ((LB / 2.D0) * Y(21)))
   DBSR2V = Y(29) - (Y(15) + ((LB / 2.D0) * Y(23)))
   DBSL2V = Y(31) - (Y(15) - ((LB / 2.D0) * Y(23)))

***** CALCULATE THE RELATIVE VERTICAL VELOCITIES BETWEEN BOLSTERS
***** AND SIDEFRAMES:

   VBSR1V = Y(26) - (Y(14) + ((LB / 2.D0) * Y(22)))
   VBSL1V = Y(28) - (Y(14) - ((LB / 2.D0) * Y(22)))
   VBSR2V = Y(30) - (Y(16) + ((LB / 2.D0) * Y(24)))
   VBSL2V = Y(32) - (Y(16) - ((LB / 2.D0) * Y(24)))

***** CALCULATE THE VERTICAL SUSPENSION LINEAR SPRING FORCES:

   FSR1V = KSUSV * DBSR1V
   FSL1V = KSUSV * DBSL1V
   FSR2V = KSUSV * DBSR2V
   FSL2V = KSUSV * DBSL2V
```

```
***** CALCULATE THE VERTICAL SUSPENSION VISCOUS DAMPING FORCES:

      IF(VBSR1V .GT. 0.D0) THEN
       IF(VBSR1V .LT. .0254D0) THEN
        DFSR1V = CSUSV1 * VBSR1V
       ELSE
        DFSR1V = CSUSV2 * VBSR1V + C0
       ENDIF
      ELSE
        DFSR1V = 0.D0
      ENDIF

      IF(VBSL1V .GT. 0.D0) THEN
       IF(VBSL1V .LT. .0254D0) THEN
        DFSL1V = CSUSV1 * VBSL1V
       ELSE
        DFSL1V = CSUSV2 * VBSL1V + C0
       ENDIF
      ELSE
        DFSL1V = 0.D0
      ENDIF

      IF(VBSR2V .GT. 0.D0) THEN
       IF(VBSR2V .LT. .0254D0) THEN
        DFSR2V = CSUSV1 * VBSR2V
       ELSE
        DFSR2V = CSUSV2 * VBSR2V + C0
       ENDIF
      ELSE
        DFSR2V = 0.D0
      ENDIF

      IF(VBSL2V .GT. 0.D0) THEN
       IF(VBSL2V .LT. .0254D0) THEN
        DFSL2V = CSUSV1 * VBSL2V
       ELSE
        DFSL2V = CSUSV2 * VBSL2V + C0
       ENDIF
      ELSE
        DFSL2V = 0.D0
      ENDIF

***** DETERMINE IF THE VERTICAL RELATIVE DISPLACEMENTS INDICATE CONTACT
***** AT THE NON-LINEAR SUSPENSION CLEARANCE SPRINGS.  IF CONTACT
***** OCCURS, CALCULATE THE VERTICAL SUSPENSION CLEARANCE FORCES:

        IF(DBSR2V .GT. CSVT .OR. DBSR2V .LT. -CSVC) THEN
           CFSR1V = KSUSCV * DBSR1V
        ELSE
           CFSR1V = 0.D0
        ENDIF

        IF(DBSL1V .GT. CSVT .OR. DBSL1V .LT. -CSVC) THEN
           CFSL1V = KSUSCV * DBSL1V
        ELSE
           CFSL1V = 0.D0
```

```
      ENDIF

      IF(DBSR2V .GT. CSVT .OR. DBSR2V .LT. -CSVC) THEN
         CFSR2V = KSUSCV * DBSR2V
      ELSE
         CFSR2V = 0.D0
      ENDIF

      IF(DBSL2V .GT. CSVT .OR. DBSL2V .LT. -CSVC) THEN
         CFSL2V = KSUSCV * DBSL2V
      ELSE
         CFSL2V = 0.D0
      ENDIF

***** CALCULATE THE VERTICAL FRICTION SPRING FORCES, REQUIRING A
***** MID-SEPARATION DISPLACEMENT VALUE, Z.  VALUES OF Z FROM
***** THE PREVIOUS STEP ARE DENOTED AS 'ZOLD' VALUES, WHICH ARE
***** USED TO DETERMINE THE VALUES AT THE CURRENT STEP, 'ZNEW'.

***** FIRST, CALCULATE TEMPORARY FRICTION SPRING FORCES USING
***** 'ZOLD' VALUES:

    FTEMR1 = KSUSFV * (DBSR1V + ZR1OLD)
    FTEML1 = KSUSFV * (DBSL1V + ZL1OLD)
    FTEMR2 = KSUSFV * (DBSR2V + ZR2OLD)
    FTEML2 = KSUSFV * (DBSL2V + ZL2OLD)

    IF(VBSR1V .GT. 0.D0) THEN
       SINR1V = 1.D0
    ELSE
       SINR1V = -1.D0
    ENDIF

    IF(VBSL1V .GT. 0.D0) THEN
       SINL1V = 1.D0
    ELSE
       SINL1V = -1.D0
    ENDIF

    IF(VBSR2V .GT. 0.D0) THEN
       SINR2V = 1.D0
    ELSE
       SINR2V = -1.D0
    ENDIF

    IF(VBSL2V .GT. 0.D0) THEN
       SINL2V = 1.D0
    ELSE
       SINL2V = -1.D0
    ENDIF

       IF(DABS(FTEMR1) .LT. FRICTV) THEN
          ZR1NEW = ZR1OLD
       ELSE
          ZR1NEW = ((SINR1V * FRICTV) / KSUSFV) - DBSR1V
       ENDIF
```

```
***** NOW THAT A VALUE FOR 'ZNEW' IS KNOWN, THE VERTICAL COULOMB
***** DAMPING FORCE, DENOTED BY PRESCRIPT 'QF', IS CALCULATED:

      QFSR1V = KSUSFV * (DBSR1V + ZR1NEW)
      ZR1OLD = ZR1NEW

         IF(DABS(FTEML1) .LT. FRICTV) THEN
           ZL1NEW = ZL1OLD
         ELSE
           ZL1NEW = ((SINL1V * FRICTV) / KSUSFV) - DBSL1V
         ENDIF

      QFSL1V = KSUSFV * (DBSL1V + ZL1NEW)

      ZL1OLD = ZL1NEW

         IF(DABS(FTEMR2) .LT. FRICTV) THEN
           ZR2NEW = ZR2OLD
         ELSE
           ZR2NEW = ((SINR2V * FRICTV) / KSUSFV) - DBSR2V
         ENDIF

      QFSR2V = KSUSFV * (DBSR2V + ZR2NEW)
      ZR2OLD = ZR2NEW

         IF(DABS(FTEML2) .LT. FRICTV) THEN
           ZL2NEW = ZL2OLD
         ELSE
           ZL2NEW = ((SINL2V * FRICTV) / KSUSFV) - DBSL2V
         ENDIF

      QFSL2V = KSUSFV * (DBSL2V + ZL2NEW)
      ZL2OLD = ZL2NEW

*******************************************************************
***** FINALLY, THE WHEELSET/RAIL CONTACT FORCES ARE CALCULATED
*******************************************************************

***** CALCULATE THE RELATIVE VERTICAL DISPLACEMENTS BETWEEN THE WHEELSET
***** AND RAILS:

      DWRFR1 = (ZRFR1 - (Y(41) + ((LW / 2.D0) * Y(49))))
      DWRFL1 = (ZRFL1 - (Y(41) - ((LW / 2.D0) * Y(49))))
      DWRRR1 = (ZRRR1 - (Y(43) + ((LW / 2.D0) * Y(51))))
      DWRRL1 = (ZRRL1 - (Y(43) - ((LW / 2.D0) * Y(51))))
      DWRFR2 = (ZRFR2 - (Y(45) + ((LW / 2.D0) * Y(53))))
      DWRFL2 = (ZRFL2 - (Y(45) - ((LW / 2.D0) * Y(53))))
      DWRRR2 = (ZRRR2 - (Y(47) + ((LW / 2.D0) * Y(55))))
      DWRRL2 = (ZRRL2 - (Y(47) - ((LW / 2.D0) * Y(55))))

***** CALCULATE THE RELATIVE VERTICAL VELOCITIES BETWEEN THE WHEELSET
***** AND RAILS:

      VWRFR1 = (ZRFR1D - (Y(42) + ((LW / 2.D0) * Y(50))))
      VWRFL1 = (ZRFL1D - (Y(42) - ((LW / 2.D0) * Y(50))))
```

```
      VWRRR1 = (ZRRR1D - (Y(44) + ((LW / 2.D0) * Y(52))))
      VWRRL1 = (ZRRL1D - (Y(44) - ((LW / 2.D0) * Y(52))))
      VWRFR2 = (ZRFR2D - (Y(46) + ((LW / 2.D0) * Y(54))))
      VWRFL2 = (ZRFL2D - (Y(46) - ((LW / 2.D0) * Y(54))))
      VWRRR2 = (ZRRR2D - (Y(48) + ((LW / 2.D0) * Y(56))))
      VWRRL2 = (ZRRL2D - (Y(48) - ((LW / 2.D0) * Y(56))))

***** DETERMINE IF THESE RELATIVE DISPLACEMENTS INDICATE SEPARATION.
***** IF NO SEPARATION OCCURS, CALCULATE THE WHEEL/RAIL CONTACT
***** FORCES:

      IF(DWRFR1 .GT. 0.D0) THEN
        FWRFR1 = KRAIL * DWRFR1 + CRAIL * VWRFR1
      ELSE
        FWRFR1 = 0.D0
      ENDIF

      IF(DWRFL1 .GT. 0.D0) THEN
        FWRFL1 = KRAIL * DWRFL1 + CRAIL * VWRFL1
      ELSE
        FWRFL1 = 0.D0
      ENDIF

      IF(DWRRR1 .GT. 0.D0) THEN
        FWRRR1 = KRAIL * DWRRR1 + CRAIL * VWRRR1
      ELSE
        FWRRR1 = 0.D0
      ENDIF

      IF(DWRRL1 .GT. 0.D0) THEN
        FWRRL1 = KRAIL * DWRRL1 + CRAIL * VWRRL1
      ELSE
        FWRRL1 = 0.D0
      ENDIF

      IF(DWRFR2 .GT. 0.D0) THEN
        FWRFR2 = KRAIL * DWRFR2 + CRAIL * VWRFR2
      ELSE
        FWRFR2 = 0.D0
      ENDIF

      IF(DWRFL2 .GT. 0.D0) THEN
        FWRFL2 = KRAIL * DWRFL2 + CRAIL * VWRFL2
      ELSE
        FWRFL2 = 0.D0
      ENDIF

      IF(DWRRR2 .GT. 0.D0) THEN
        FWRRR2 = KRAIL * DWRRR2 + CRAIL * VWRRR2
      ELSE
        FWRRR2 = 0.D0
      ENDIF

      IF(DWRRL2 .GT. 0.D0) THEN
        FWRRL2 = KRAIL * DWRRL2 + CRAIL * VWRRL2
      ELSE
```

```
      FWRRL2 = 0.D0
    ENDIF

************************************************************************
************************************************************************
*****
***** NOW, THE FIRST-ORDER SYSTEM IS EVALUATED
*****
************************************************************************
************************************************************************

    V(2) = COEFF(1) * ((-COEFF(2) * QDDOT)
   2    + (FCBR1 + FCBL1 + FCBR2 + FCBL2)
   3    + (FCER1V + FCEL1V + FCER2V + FCEL2V)
   4    - (MC * GRAV))

    V(4) = COEFF(1) * (FCER1L + FCEL1L + FCER2L + FCEL2L)

    V(6) = COEFF(3) * (LCEN / 2.D0) * ((FCBR1 + FCBL1)
   2    - (FCBR2 + FCBL2) + (FCER1V + FCEL1V)
   3    - (FCER2V + FCEL2V))

    V(8) = COEFF(4) *
   2     ((LCB / 2.D0) * ((FCBR1 + FCBR2) - (FCBL1 + FCBL2))
   3     + (DCEN / 2.D0) * ((FCER1V + FCER2V) - (FCEL1V + FCEL2V))
   4     + LCGCAR * (-FCER1L - FCEL1L - FCER2L - FCEL2L))

    V(10) = (COEFF(1) / .4118D0) * ((-COEFF(2) * ZDDOT)
   2    - (STIFF * Y(9)) - (125000.D0 * Y(10)))

    V(12) = COEFF(11) * (-(TWIST * Y(11)) - (250000.D0 * Y(12))
   2     + (LCB / 2.D0) * ((FCBR1 - FCBR2) + (-FCBL1 + FCBL2))
   3     + (DCEN / 2.D0) * ((FCER1V - FCER2V) + (-FCEL1V + FCEL2V))
   4     + LCGCAR * (-FCER1L - FCEL1L + FCER2L + FCEL2L))

    V(14) = COEFF(5) * (-(FCBR1 + FCBL1) - (FCER1V + FCEL1V)
   2     + (FSR1V + FSL1V) + (CFSR1V + CFSL1V)
   3     + (DFSR1V + DFSL1V) + (QFSR1V + QFSL1V)
   4     - (MB * GRAV))

    V(16) = COEFF(5) * (-(FCBR2 + FCBL2) - (FCER2V + FCEL2V)
   2     + (FSR2V + FSL2V) + (CFSR2V + CFSL2V)
   2     + (DFSR2V + DFSL2V) + (QFSR2V + QFSL2V)
   4     - (MB * GRAV))

    V(18) = COEFF(5) * (-(FCER1L + FCEL2L)
   2     + (FSR1L + FSL1L) + (CFSR1L + CFSL1L)
   3     + (QFSR1L + QFSL1L))

    V(20) = COEFF(5) * (-(FCER2L + FCEL2L)
   2     + (FSR2L + FSL2L) + (CFSR2L + CFSL2L)
   3     + (QFSR2L + QFSL2L))

    V(22) = COEFF(6) * (LCB / 2.D0) * (FCBL1 - FCBR1)
   2     + COEFF(6) * (LB / 2.D0) * ((FSR1V - FSL1V)
```

```
3     + (CFSR1V - CFSL1V) + (DFSR1V - DFSL1V) + (QFSR1V - QFSL1V))
4     + COEFF(6) * (DCEN / 2.D0) * (FCEL1V - FCER1V)
5     + COEFF(6) * LCGB * (-FCEL1L - FCER1L)

 V(24) = COEFF(6) * (LCB / 2.D0) * (FCBL2 - FCBR2)
2     + COEFF(6) * (LB / 2.D0) * ((FSR2V - FSL2V)
3     + (CFSR2V - CFSL2V) + (DFSR2V - DFSL2V) + (QFSR2V - QFSL2V))
4     + COEFF(6) * (DCEN / 2.D0) * (FCEL2V - FCER2V)
5     + COEFF(6) * LCGB * (-FCEL2L - FCER2L)

 V(42) = COEFF(9) * ((FWRFR1 + FWRFL1) - (MW * GRAV)
2     - ((FSR1V + CFSR1V + DFSR1V + QFSR1V + MS * GRAV) / 2.D0)
3     - ((FSL1V + CFSL1V + DFSL1V + QFSL1V + MS * GRAV) / 2.D0))

 V(44) = COEFF(9) * ((FWRRR1 + FWRRL1) - (MW * GRAV)
2     - ((FSR1V + CFSR1V + DFSR1V + QFSR1V + MS * GRAV) / 2.D0)
3     - ((FSL1V + CFSL1V + DFSL1V + QFSL1V + MS * GRAV) / 2.D0))

 V(46) = COEFF(9) * ((FWRFR2 + FWRFL2) - (MW * GRAV)
2     - ((FSR2V + CFSR2V + DFSR2V + QFSR2V + MS * GRAV) / 2.D0)
3     - ((FSL2V + CFSL2V + DFSL2V + QFSL2V + MS * GRAV) / 2.D0))

 V(48) = COEFF(9) * ((FWRRR2 + FWRRL2) - (MW * GRAV)
2     - ((FSR2V + CFSR2V + DFSR2V + QFSR2V + MS * GRAV) / 2.D0)
3     - ((FSL2V + CFSL2V + DFSL2V + QFSL2V + MS * GRAV) / 2.D0))

 V(50) = COEFF(10) * (LW / 2.D0) * (FWRFR1 - FWRFL1)
2     + COEFF(10) * (LB / 2.D0)
3     * (((FSL1V + CFSL1V + DFSL1V + QFSL1V + MS * GRAV) / 2.D0)
4     - ((FSR1V + CFSR1V + DFSR1V + QFSR1V + MS * GRAV) / 2.D0))

 V(52) = COEFF(10) * (LW / 2.D0) * (FWRRR1 - FWRRL1)
2     + COEFF(10) * (LB / 2.D0)
3     * (((FSL1V + CFSL1V + DFSL1V + QFSL1V + MS * GRAV) / 2.D0)
4     - ((FSR1V + CFSR1V + DFSR1V + QFSR1V + MS * GRAV) / 2.D0))

 V(54) = COEFF(10) * (LW / 2.D0) * (FWRFR2 - FWRFL2)
2     + COEFF(10) * (LB / 2.D0)
3     * (((FSL2V + CFSL2V + DFSL2V + QFSL2V + MS * GRAV) / 2.D0)
4     - ((FSR2V + CFSR2V + DFSR2V + QFSR2V + MS * GRAV) / 2.D0))

 V(56) = COEFF(10) * (LW / 2.D0) * (FWRRR2 - FWRRL2)
2     + COEFF(10) * (LB / 2.D0)
3     * (((FSL2V + CFSL2V + DFSL2V + QFSL2V + MS * GRAV) / 2.D0)
4     - ((FSR2V + CFSR2V + DFSR2V + QFSR2V + MS * GRAV) / 2.D0))

 V(26) = ((V(42) + ((LB / 2.D0) * V(50)))
2     + (V(44) + ((LB / 2.D0) * V(52)))) / 2.D0

 V(28) = ((V(42) - ((LB / 2.D0) * V(50)))
2     + (V(44) - ((LB / 2.D0) * V(52)))) / 2.D0

 V(30) = ((V(46) + ((LB / 2.D0) * V(54)))
2     + (V(48) + ((LB / 2.D0) * V(56)))) / 2.D0

 V(32) = ((V(46) - ((LB / 2.D0) * V(54)))
```

```
     2     + (V(48) - ((LB / 2.D0) * V(56)))) / 2.D0

      V(34) = ((V(42) + ((LB / 2.D0) * V(50)))
     2     - (V(44) + ((LB / 2.D0) * V(52)))) / LS

      V(36) = ((V(42) - ((LB / 2.D0) * V(50)))
     2     - (V(44) - ((LB / 2.D0) * V(52)))) / LS

      V(38) = ((V(46) + ((LB / 2.D0) * V(54)))
     2     - (V(48) + ((LB / 2.D0) * V(56)))) / LS

      V(40) = ((V(46) - ((LB / 2.D0) * V(54)))
     2     - (V(48) - ((LB / 2.D0) * V(56)))) / LS

      ZDDOT = V(2)
      QDDOT = V(10)

C     WRITE(*, *) 'V1 = ', V(1)
C     WRITE(*, *) 'V2 = ', V(2)
C     WRITE(*, *) 'V3 = ', V(3)
C     WRITE(*, *) 'V4 = ', V(4)
C     WRITE(*, *) 'V5 = ', V(5)
C     WRITE(*, *) 'V6 = ', V(6)
C     WRITE(*, *) 'V7 = ', V(7)
C     WRITE(*, *) 'V8 = ', V(8)
C     WRITE(*, *) 'V9 = ', V(9)
C     WRITE(*, *) 'V10 = ', V(10)
C     WRITE(*, *) 'V11 = ', V(11)
C     WRITE(*, *) 'V12 = ', V(12)
C     WRITE(*, *) 'V13 = ', V(13)
C     WRITE(*, *) 'V14 = ', V(14)
C     WRITE(*, *) 'V15 = ', V(15)
C     WRITE(*, *) 'V16 = ', V(16)
C     WRITE(*, *) 'V17 = ', V(17)
C     WRITE(*, *) 'V18 = ', V(18)
C     WRITE(*, *) 'V19 = ', V(19)
C     WRITE(*, *) 'V20 = ', V(20)
C     WRITE(*, *) 'V21 = ', V(21)
C     WRITE(*, *) 'V22 = ', V(22)
C     WRITE(*, *) 'V23 = ', V(23)
C     WRITE(*, *) 'V24 = ', V(24)
C     WRITE(*, *) 'V25 = ', V(25)
C     WRITE(*, *) 'V26 = ', V(26)
C     WRITE(*, *) 'V27 = ', V(27)
C     WRITE(*, *) 'V28 = ', V(28)
C     WRITE(*, *) 'V29 = ', V(29)
C     WRITE(*, *) 'V30 = ', V(30)
C     WRITE(*, *) 'V31 = ', V(31)
C     WRITE(*, *) 'V32 = ', V(32)
C     WRITE(*, *) 'V33 = ', V(33)
C     WRITE(*, *) 'V34 = ', V(34)
C     WRITE(*, *) 'V35 = ', V(35)
C     WRITE(*, *) 'V36 = ', V(36)
C     WRITE(*, *) 'V37 = ', V(37)
C     WRITE(*, *) 'V38 = ', V(38)
C     WRITE(*, *) 'V39 = ', V(39)
```

```
C     WRITE(*, *) 'V40 = ', V(40)
C     WRITE(*, *) 'V41 = ', V(41)
C     WRITE(*, *) 'V42 = ', V(42)
C     WRITE(*, *) 'V43 = ', V(43)
C     WRITE(*, *) 'V44 = ', V(44)
C     WRITE(*, *) 'V45 = ', V(45)
C     WRITE(*, *) 'V46 = ', V(46)
C     WRITE(*, *) 'V47 = ', V(47)
C     WRITE(*, *) 'V48 = ', V(48)
C     WRITE(*, *) 'V49 = ', V(49)
C     WRITE(*, *) 'V50 = ', V(50)
C     WRITE(*, *) 'V51 = ', V(51)
C     WRITE(*, *) 'V52 = ', V(52)
C     WRITE(*, *) 'V53 = ', V(53)
C     WRITE(*, *) 'V54 = ', V(54)

      RETURN
      END

      SUBROUTINE RAILF1(V,ITRACK,DT,A,AMP,WLENG,SEED,YL,YR,YLDD,YRDD,
     2             YLD,YRD)
      IMPLICIT REAL*8(A - H, O - Z)
C
C        This subroutine computes the random vertical alignment of
C        left and right rails. The subroutine implements equations
C        2-4 and 2-5 in "Guideway-Suspension Tradeoffs in Rail Vehicle
C        Systems," by R. C. White, et al., Arizona State University
C        Report ERC-R-78035, January, 1978.
C
C        The equations are implemented as a shape filter using the state
C        transition matrix and approximating the inputs to be constant
C        between time samples. The inputs are Gausian random variables
C        generated by subroutines GAUSS and RAN. These subroutines
C        generate random variables with a mean of zero and a standard
C        standard deviation equal to one.
C
C        Written by R. H. Fries
C        November 1986
C
C        The shape filter equations are formulated as
C
C          XDOT = A*X + B*U
C
C        and the shape filter is implemented as
C
C          X(T+DT) = P*X(T) + Q*U(T)
C
C        where
C
C          P = the state transition matrix
C          Q = the forcing matrix
C
C        Input variables:
C
C        V      = vehicle speed (m/s)
```

```
C       ITRACK = track class:  4, 5, or 6 can be selected.
C       DT     = sample interval (sec)  Note:  Must remain constant.
C       A      = semi gauge (m)
C       AMP    = amplitude of sine wave to be added to the crosslevel
C                signal (m)
C       WLENG  = wavelength of the sine wave (m)
C       SEED   = REAL*8 seed of the random number generator in the
C                range (1.D-9, .999999999D0)
C
C       Output variables:
C
C       YL     = left rail alignment (m)
C       YL     = X1 - A*(X3+YDISCR)
C       YLD    = THE FIRST DERIVATIVE OF YL (M/S)
C       YLDD   = the second derivative of YL (m/s**2)
C
C       YR     = right rail alignment (m)
C       YR     = X1 + A*(X3+YDISCR)
C       YRD    = THE FIRST DERIVATIVE OF YR (M/S)
C       YRDD   = the second derivative of YR (M/S**2)
C
C       YLRMS  = RMS value of YL (m)
C       YRRMS  = RMS value of YR (m)
C
C       where
C
C       X1     = random vertical alignment (m)
C       X3     = random crosslevel alignment (rad)
C       YDISCR = the amplitude of the discrete crosslevel angle (rad)
C
C
      DIMENSION RN(2)
      DATA NCALL/-1/
C
C       The subroutine initialization block is entered only on the
C       first subroutine call.
C
      IF (NCALL .EQ. -1) THEN
        TWOPI = 2. * ACOS(-1.)
        BOMEGC = 0.8246
        BOMEGR = 0.02061
        IF (ITRACK .EQ. 4) THEN
          BOMEGS = 1.13
          AV = 2.39E-5
        ELSEIF (ITRACK .EQ. 5) THEN
          BOMEGS = 0.821
          AV = 9.35E-6
        ELSEIF (ITRACK .EQ. 6) THEN
          BOMEGS = 0.438
          AV = 1.5E-6
        ELSE
          WRITE(6,100) ITRACK
 100      FORMAT('ITRACK = ',I2,' NOT ALLOWED.')
          STOP
        ENDIF
        C1 = SQRT(TWOPI*AV*V)/SQRT(DT*2.)
```

```
      C2 = SQRT(TWOPI*AV*V)/(A*SQRT(DT*2.))
      OMEGC = V*BOMEGC
      OMEGR = V*BOMEGR
      OMEGS = V*BOMEGS
      IF (WLENG .NE. 0.) THEN
       OMEGD = V*TWOPI/WLENG
      ELSE
       OMEGD = 0.
      ENDIF
C
C      Compute elements of the state transition matrix
C
      P11 = EXP(-OMEGR*DT)
      P12 = (EXP(-OMEGR*DT) - EXP(-OMEGC*DT))/(OMEGC - OMEGR)
      P22 = EXP(-OMEGC*DT)
      P33 = EXP(-OMEGS*DT)
      P34 = (EXP(-OMEGS*DT) - EXP(-OMEGR*DT))/(OMEGR - OMEGS)
      P35 = -OMEGC*EXP(-OMEGS*DT)/((OMEGR-OMEGS)*(OMEGC-OMEGS))
     1     -OMEGC*EXP(-OMEGR*DT)/((OMEGS-OMEGR)*(OMEGC-OMEGR))
     2     -OMEGC*EXP(-OMEGC*DT)/((OMEGS-OMEGC)*(OMEGR-OMEGC))
      P44 = EXP(-OMEGR*DT)
      P45 = (EXP(-OMEGR*DT)-EXP(-OMEGC*DT))*OMEGC/(OMEGR-OMEGC)
      P55 = EXP(-OMEGC*DT)
C
C      Compute elements of the forcing matrix Q.
C      Q = E*B
C
      E11 = (1.-EXP(-OMEGR*DT))/OMEGR
      E12 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1     +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E22 = (1.-EXP(-OMEGC*DT))/OMEGC
      E34 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1     +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E35 = -OMEGC*(1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS)
     1       *(OMEGC-OMEGS))
     2     -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR)
     3       *(OMEGC-OMEGR))
     4     -(1.-EXP(-OMEGC*DT))/((OMEGS-OMEGC)*(OMEGR-OMEGC))
      E44 = E11
      E45 = -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGC-OMEGR))
     1     -(1.-EXP(-OMEGC*DT))/(OMEGR-OMEGC)
      E55 = E22
C
      Q11 = E12*C1*OMEGC
      Q21 = E22*C1*OMEGC
      Q32 = (E34+E35)*C2*OMEGC
      Q42 = (E44+E45)*C2*OMEGC
      Q52 = E55*C2*OMEGC
C
      NCALL = 0
C
C        Initialize the state variables to zero
C
      X1 = 0.
      X2 = 0.
      X3 = 0.
```

```
      X4 = 0.
      X5 = 0.
C
C      Initialize the sum squared variables to zero
C
      YLSQ = 0.
      YRSQ = 0.
    ENDIF
      NCALL = NCALL + 1
      CALL GAUSS(SEED,2,RN)
      X1 = P11*X1 + P12*X2 + Q11*RN(1)
      X2 = P22*X2 + Q21*RN(1)
      X3 = P33*X3 + P34*X4 + P35*X5 + Q32*RN(2)
      X4 = P44*X4 + P45*X5 + Q42*RN(2)
      X5 = P55*X5 + Q52*RN(2)
C
C      GET FIRST DERIVATIVES OF VERTICAL AND ALIGNMENT STATE
C         VARIABLES.
C
      X1D = -OMEGR*X1 + X2
      X2D = -OMEGC*X2 + C1*OMEGC*RN(1)
      X3D = -OMEGS*X3 + X4
C
C      Get second derivatives of vertical and alignment state
C         variables.
C
      X1DD = OMEGR**2*X1 - (OMEGR+OMEGC)*X2 + C1*OMEGC*RN(1)
      X3DD = OMEGS**2*X3 - (OMEGS+OMEGR)*X4 - OMEGC*X5
     1      + C2*OMEGC*RN(2)
C
C      Compute the discrete sine wave magnitude as a crosslevel
C         angle
C
      YDISCR = AMP*SIN(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSD   = AMP*OMEGD*COS(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSDD  = -OMEGD**2*YDISCR
C
C      Compute the rail displacements
C
      YL    = X1   - A*(X3   + YDISCR)
      YLD   = X1D  - A*(X3D  + YDSD)
      YLDD  = X1DD - A*(X3DD + YDSDD)
      YR    = X1   + A*(X3   + YDISCR)
      YRD   = X1D  + A*(X3D  + YDSD)
      YRDD  = X1DD + A*(X3DD + YDSDD)
      YLSQ  = YLSQ + YL*YL
      YRSQ  = YRSQ + YR*YR
      YLRMS = SQRT(YLSQ/FLOAT(NCALL))
      YRRMS = SQRT(YRSQ/FLOAT(NCALL))
    RETURN
    END


    SUBROUTINE RAILR1(V,ITRACK,DT,A,AMP,WLENG,SEED,YL,YR,YLDD,YRDD,
   2            YLD,YRD)
```

```
      IMPLICIT REAL*8(A - H, O - Z)
C
C        This subroutine computes the random vertical alignment of
C        left and right rails. The subroutine implements equations
C        2-4 and 2-5 in "Guideway-Suspension Tradeoffs in Rail Vehicle
C        Systems," by R. C. White, et al., Arizona State University
C        Report ERC-R-78035, January, 1978.
C
C        The equations are implemented as a shape filter using the state
C        transition matrix and approximating the inputs to be constant
C        between time samples. The inputs are Gausian random variables
C        generated by subroutines GAUSS and RAN. These subroutines
C        generate random variables with a mean of zero and a standard
C        standard deviation equal to one.
C
C        Written by R. H. Fries
C        November 1986
C
C        The shape filter equations are formulated as
C
C          XDOT = A*X + B*U
C
C        and the shape filter is implemented as
C
C          X(T+DT) = P*X(T) + Q*U(T)
C
C        where
C
C          P = the state transition matrix
C          Q = the forcing matrix
C
C        Input variables:
C
C        V      = vehicle speed (m/s)
C        ITRACK = track class: 4, 5, or 6 can be selected.
C        DT     = sample interval (sec) Note: Must remain constant.
C        A      = semi gauge (m)
C        AMP    = amplitude of sine wave to be added to the crosslevel
C                 signal (m)
C        WLENG  = wavelength of the sine wave (m)
C        SEED   = REAL*8 seed of the random number generator in the
C                 range (1.D-9, .999999999D0)
C
C        Output variables:
C
C        YL     = left rail alignment (m)
C        YL     = X1 - A*(X3+YDISCR)
C        YLD    = THE FIRST DERIVATIVE OF YL (M/S)
C        YLDD   = the second derivative of YL (m/s**2)
C
C        YR     = right rail alignment (m)
C        YR     = X1 + A*(X3+YDISCR)
C        YRD    = THE FIRST DERIVATIVE OF YR (M/S)
C        YRDD   = the second derivative of YR (M/S**2)
C
C        YLRMS  = RMS value of YL (m)
```

```
C       YRRMS  = RMS value of YR (m)
C
C       where
C
C       X1     = random vertical alignment (m)
C       X3     = random crosslevel alignment (rad)
C       YDISCR = the amplitude of the discrete crosslevel angle (rad)
C
C
      DIMENSION RN(2)
      DATA NCALL/-1/
C
C       The subroutine initialization block is entered only on the
C       first subroutine call.
C
      IF (NCALL .EQ. -1) THEN
        TWOPI = 2. * ACOS(-1.)
        BOMEGC = 0.8246
        BOMEGR = 0.02061
        IF (ITRACK .EQ. 4) THEN
          BOMEGS = 1.13
          AV = 2.39E-5
        ELSEIF (ITRACK .EQ. 5) THEN
          BOMEGS = 0.821
          AV = 9.35E-6
        ELSEIF (ITRACK .EQ. 6) THEN
          BOMEGS = 0.438
          AV = 1.5E-6
        ELSE
          WRITE(6,100) ITRACK
  100      FORMAT('ITRACK = ',I2,' NOT ALLOWED.')
          STOP
        ENDIF
        C1 = SQRT(TWOPI*AV*V)/SQRT(DT*2.)
        C2 = SQRT(TWOPI*AV*V)/(A*SQRT(DT*2.))
        OMEGC = V*BOMEGC
        OMEGR = V*BOMEGR
        OMEGS = V*BOMEGS
        IF (WLENG .NE. 0.) THEN
         OMEGD = V*TWOPI/WLENG
        ELSE
         OMEGD = 0.
        ENDIF
C
C       Compute elements of the state transition matrix
C
        P11 = EXP(-OMEGR*DT)
        P12 = (EXP(-OMEGR*DT) - EXP(-OMEGC*DT))/(OMEGC - OMEGR)
        P22 = EXP(-OMEGC*DT)
        P33 = EXP(-OMEGS*DT)
        P34 = (EXP(-OMEGS*DT) - EXP(-OMEGR*DT))/(OMEGR - OMEGS)
        P35 = -OMEGC*EXP(-OMEGS*DT)/((OMEGR-OMEGS)*(OMEGC-OMEGS))
     1        -OMEGC*EXP(-OMEGR*DT)/((OMEGS-OMEGR)*(OMEGC-OMEGR))
     2        -OMEGC*EXP(-OMEGC*DT)/((OMEGS-OMEGC)*(OMEGR-OMEGC))
        P44 = EXP(-OMEGR*DT)
        P45 = (EXP(-OMEGR*DT)-EXP(-OMEGC*DT))*OMEGC/(OMEGR-OMEGC)
```

```
      P55 = EXP(-OMEGC*DT)
C
C      Compute elements of the forcing matrix Q.
C      Q = E*B
C
      E11 = (1.-EXP(-OMEGR*DT))/OMEGR
      E12 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1      +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E22 = (1.-EXP(-OMEGC*DT))/OMEGC
      E34 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1      +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E35 = -OMEGC*(1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS)
     1        *(OMEGC-OMEGS))
     2      -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR)
     3        *(OMEGC-OMEGR))
     4      -(1.-EXP(-OMEGC*DT))/((OMEGS-OMEGC)*(OMEGR-OMEGC))
      E44 = E11
      E45 = -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGC-OMEGR))
     1      -(1.-EXP(-OMEGC*DT))/(OMEGR-OMEGC)
      E55 = E22
C
      Q11 = E12*C1*OMEGC
      Q21 = E22*C1*OMEGC
      Q32 = (E34+E35)*C2*OMEGC
      Q42 = (E44+E45)*C2*OMEGC
      Q52 = E55*C2*OMEGC
C
      NCALL = 0
C
C         Initialize the state variables to zero
C
      X1 = 0.
      X2 = 0.
      X3 = 0.
      X4 = 0.
      X5 = 0.
C
C      Initialize the sum squared variables to zero
C
      YLSQ = 0.
      YRSQ = 0.
    ENDIF
      NCALL = NCALL + 1
      CALL GAUSS(SEED,2,RN)
      X1 = P11*X1 + P12*X2 + Q11*RN(1)
      X2 = P22*X2 + Q21*RN(1)
      X3 = P33*X3 + P34*X4 + P35*X5 + Q32*RN(2)
      X4 = P44*X4 + P45*X5 + Q42*RN(2)
      X5 = P55*X5 + Q52*RN(2)
C
C      GET FIRST DERIVATIVES OF VERTICAL AND ALIGNMENT STATE
C         VARIABLES.
C
      X1D = -OMEGR*X1 + X2
      X2D = -OMEGC*X2 + C1*OMEGC*RN(1)
```

```
      X3D = -OMEGS*X3 + X4
C
C      Get second derivatives of vertical and alignment state
C        variables.
C
      X1DD = OMEGR**2*X1 - (OMEGR+OMEGC)*X2 + C1*OMEGC*RN(1)
      X3DD = OMEGS**2*X3 - (OMEGS+OMEGR)*X4 - OMEGC*X5
     1        + C2*OMEGC*RN(2)
C
C      Compute the discrete sine wave magnitude as a crosslevel
C        angle
C
      YDISCR = AMP*SIN(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSD   = AMP*OMEGD*COS(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSDD  = -OMEGD**2*YDISCR
C
C      Compute the rail displacements
C
      YL    = X1   - A*(X3   + YDISCR)
      YLD   = X1D  - A*(X3D  + YDSD)
      YLDD  = X1DD - A*(X3DD + YDSDD)
      YR    = X1   + A*(X3   + YDISCR)
      YRD   = X1D  + A*(X3D  + YDSD)
      YRDD  = X1DD + A*(X3DD + YDSDD)
      YLSQ  = YLSQ + YL*YL
      YRSQ  = YRSQ + YR*YR
      YLRMS = SQRT(YLSQ/FLOAT(NCALL))
      YRRMS = SQRT(YRSQ/FLOAT(NCALL))
      RETURN
      END

      SUBROUTINE RAILF2(V,ITRACK,DT,A,AMP,WLENG,SEED,YL,YR,YLDD,YRDD,
     2            YLD,YRD)
      IMPLICIT REAL*8(A - H, O - Z)
C
C      This subroutine computes the random vertical alignment of
C      left and right rails. The subroutine implements equations
C      2-4 and 2-5 in "Guideway-Suspension Tradeoffs in Rail Vehicle
C      Systems," by R. C. White, et al., Arizona State University
C      Report ERC-R-78035, January, 1978.
C
C      The equations are implemented as a shape filter using the state
C      transition matrix and approximating the inputs to be constant
C      between time samples. The inputs are Gausian random variables
C      generated by subroutines GAUSS and RAN. These subroutines
C      generate random variables with a mean of zero and a standard
C      standard deviation equal to one.
C
C      Written by R. H. Fries
C      November 1986
C
C      The shape filter equations are formulated as
C
C        XDOT = A*X + B*U
C
C      and the shape filter is implemented as
```

```
C
C        X(T+DT) = P*X(T) + Q*U(T)
C
C      where
C
C        P = the state transition matrix
C        Q = the forcing matrix
C
C      Input variables:
C
C      V      = vehicle speed (m/s)
C      ITRACK = track class: 4, 5, or 6 can be selected.
C      DT     = sample interval (sec)  Note: Must remain constant.
C      A      = semi gauge (m)
C      AMP    = amplitude of sine wave to be added to the crosslevel
C               signal (m)
C      WLENG  = wavelength of the sine wave (m)
C      SEED   = REAL*8 seed of the random number generator in the
C               range (1.D-9, .999999999D0)
C
C      Output variables:
C
C      YL     = left rail alignment (m)
C      YL     = X1 - A*(X3+YDISCR)
C      YLD    = THE FIRST DERIVATIVE OF YL (M/S)
C      YLDD   = the second derivative of YL (m/s**2)
C
C      YR     = right rail alignment (m)
C      YR     = X1 + A*(X3+YDISCR)
C      YRD    = THE FIRST DERIVATIVE OF YR (M/S)
C      YRDD   = the second derivative of YR (M/S**2)
C
C      YLRMS  = RMS value of YL (m)
C      YRRMS  = RMS value of YR (m)
C
C      where
C
C      X1     = random vertical alignment (m)
C      X3     = random crosslevel alignment (rad)
C      YDISCR = the amplitude of the discrete crosslevel angle (rad)
C
C
      DIMENSION RN(2)
      DATA NCALL/-1/
C
C      The subroutine initialization block is entered only on the
C      first subroutine call.
C
      IF (NCALL .EQ. -1) THEN
        TWOPI = 2. * ACOS(-1.)
        BOMEGC = 0.8246
        BOMEGR = 0.02061
        IF (ITRACK .EQ. 4) THEN
          BOMEGS = 1.13
          AV = 2.39E-5
        ELSEIF (ITRACK .EQ. 5) THEN
```

```
          BOMEGS = 0.821
          AV = 9.35E-6
        ELSEIF (ITRACK .EQ. 6) THEN
          BOMEGS = 0.438
          AV = 1.5E-6
        ELSE
          WRITE(6,100) ITRACK
 100       FORMAT('ITRACK = ',I2,' NOT ALLOWED.')
          STOP
        ENDIF
        C1 = SQRT(TWOPI*AV*V)/SQRT(DT*2.)
        C2 = SQRT(TWOPI*AV*V)/(A*SQRT(DT*2.))
        OMEGC = V*BOMEGC
        OMEGR = V*BOMEGR
        OMEGS = V*BOMEGS
        IF (WLENG .NE. 0.) THEN
         OMEGD = V*TWOPI/WLENG
        ELSE
         OMEGD = 0.
        ENDIF
C
C        Compute elements of the state transition matrix
C
        P11 = EXP(-OMEGR*DT)
        P12 = (EXP(-OMEGR*DT) - EXP(-OMEGC*DT))/(OMEGC - OMEGR)
        P22 = EXP(-OMEGC*DT)
        P33 = EXP(-OMEGS*DT)
        P34 = (EXP(-OMEGS*DT) - EXP(-OMEGR*DT))/(OMEGR - OMEGS)
        P35 = -OMEGC*EXP(-OMEGS*DT)/((OMEGR-OMEGS)*(OMEGC-OMEGS))
     1      -OMEGC*EXP(-OMEGR*DT)/((OMEGS-OMEGR)*(OMEGC-OMEGR))
     2      -OMEGC*EXP(-OMEGC*DT)/((OMEGS-OMEGC)*(OMEGR-OMEGC))
        P44 = EXP(-OMEGR*DT)
        P45 = (EXP(-OMEGR*DT)-EXP(-OMEGC*DT))*OMEGC/(OMEGR-OMEGC)
        P55 = EXP(-OMEGC*DT)
C
C        Compute elements of the forcing matrix Q.
C        Q = E*B
C
        E11 = (1.-EXP(-OMEGR*DT))/OMEGR
        E12 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1      +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
        E22 = (1.-EXP(-OMEGC*DT))/OMEGC
        E34 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1      +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
        E35 = -OMEGC*(1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS)
     1        *(OMEGC-OMEGS))
     2      -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR)
     3        *(OMEGC-OMEGR))
     4      -(1.-EXP(-OMEGC*DT))/((OMEGS-OMEGC)*(OMEGR-OMEGC))
        E44 = E11
        E45 = -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGC-OMEGR))
     1      -(1.-EXP(-OMEGC*DT))/(OMEGR-OMEGC)
        E55 = E22
C
        Q11 = E12*C1*OMEGC
        Q21 = E22*C1*OMEGC
```

```
      Q32 = (E34+E35)*C2*OMEGC
      Q42 = (E44+E45)*C2*OMEGC
      Q52 = E55*C2*OMEGC
C
      NCALL = 0
C
C        Initialize the state variables to zero
C
      X1 = 0.
      X2 = 0.
      X3 = 0.
      X4 = 0.
      X5 = 0.
C
C       Initialize the sum squared variables to zero
C
      YLSQ = 0.
      YRSQ = 0.
    ENDIF
      NCALL = NCALL + 1
      CALL GAUSS(SEED,2,RN)
      X1 = P11*X1 + P12*X2 + Q11*RN(1)
      X2 = P22*X2 + Q21*RN(1)
      X3 = P33*X3 + P34*X4 + P35*X5 + Q32*RN(2)
      X4 = P44*X4 + P45*X5 + Q42*RN(2)
      X5 = P55*X5 + Q52*RN(2)
C
C       GET FIRST DERIVATIVES OF VERTICAL AND ALIGNMENT STATE
C          VARIABLES.
C
      X1D = -OMEGR*X1 + X2
      X2D = -OMEGC*X2 + C1*OMEGC*RN(1)
      X3D = -OMEGS*X3 + X4
C
C       Get second derivatives of vertical and alignment state
C         variables.
C
      X1DD = OMEGR**2*X1 - (OMEGR+OMEGC)*X2 + C1*OMEGC*RN(1)
      X3DD = OMEGS**2*X3 - (OMEGS+OMEGR)*X4 - OMEGC*X5
   1       + C2*OMEGC*RN(2)
C
C       Compute the discrete sine wave magnitude as a crosslevel
C         angle
C
      YDISCR = AMP*SIN(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSD   = AMP*OMEGD*COS(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSDD  = -OMEGD**2*YDISCR
C
C       Compute the rail displacements
C
      YL    = X1   - A*(X3   + YDISCR)
      YLD   = X1D  - A*(X3D  + YDSD)
      YLDD  = X1DD - A*(X3DD + YDSDD)
      YR    = X1   + A*(X3   + YDISCR)
      YRD   = X1D  + A*(X3D  + YDSD)
      YRDD  = X1DD + A*(X3DD + YDSDD)
```

```
      YLSQ  = YLSQ + YL*YL
      YRSQ  = YRSQ + YR*YR
      YLRMS = SQRT(YLSQ/FLOAT(NCALL))
      YRRMS = SQRT(YRSQ/FLOAT(NCALL))
    RETURN
    END

    SUBROUTINE RAILR2(V,ITRACK,DT,A,AMP,WLENG,SEED,YL,YR,YLDD,YRDD,
   2              YLD,YRD)
    IMPLICIT REAL*8(A - H, O - Z)
C
C     This subroutine computes the random vertical alignment of
C     left and right rails. The subroutine implements equations
C     2-4 and 2-5 in "Guideway-Suspension Tradeoffs in Rail Vehicle
C     Systems," by R. C. White, et al., Arizona State University
C     Report ERC-R-78035, January, 1978.
C
C     The equations are implemented as a shape filter using the state
C     transition matrix and approximating the inputs to be constant
C     between time samples. The inputs are Gausian random variables
C     generated by subroutines GAUSS and RAN. These subroutines
C     generate random variables with a mean of zero and a standard
C     standard deviation equal to one.
C
C     Written by R. H. Fries
C     November 1986
C
C     The shape filter equations are formulated as
C
C       XDOT = A*X + B*U
C
C     and the shape filter is implemented as
C
C       X(T+DT) = P*X(T) + Q*U(T)
C
C     where
C
C       P = the state transition matrix
C       Q = the forcing matrix
C
C     Input variables:
C
C     V      = vehicle speed (m/s)
C     ITRACK = track class: 4, 5, or 6 can be selected.
C     DT     = sample interval (sec) Note: Must remain constant.
C     A      = semi gauge (m)
C     AMP    = amplitude of sine wave to be added to the crosslevel
C              signal (m)
C     WLENG  = wavelength of the sine wave (m)
C     SEED   = REAL*8 seed of the random number generator in the
C              range (1.D-9, .999999999D0)
C
C     Output variables:
C
C     YL     = left rail alignment (m)
```

```
C       YL     = X1 - A*(X3+YDISCR)
C       YLD    = THE FIRST DERIVATIVE OF YL (M/S)
C       YLDD   = the second derivative of YL (m/s**2)
C
C       YR     = right rail alignment (m)
C       YR     = X1 + A*(X3+YDISCR)
C       YRD    = THE FIRST DERIVATIVE OF YR (M/S)
C       YRDD   = the second derivative of YR (M/S**2)
C
C       YLRMS  = RMS value of YL (m)
C       YRRMS  = RMS value of YR (m)
C
C       where
C
C       X1     = random vertical alignment (m)
C       X3     = random crosslevel alignment (rad)
C       YDISCR = the amplitude of the discrete crosslevel angle (rad)
C
C
      DIMENSION RN(2)
      DATA NCALL/-1/
C
C       The subroutine initialization block is entered only on the
C       first subroutine call.
C
      IF (NCALL .EQ. -1) THEN
        TWOPI = 2. * ACOS(-1.)
        BOMEGC = 0.8246
        BOMEGR = 0.02061
        IF (ITRACK .EQ. 4) THEN
          BOMEGS = 1.13
          AV = 2.39E-5
        ELSEIF (ITRACK .EQ. 5) THEN
          BOMEGS = 0.821
          AV = 9.35E-6
        ELSEIF (ITRACK .EQ. 6) THEN
          BOMEGS = 0.438
          AV = 1.5E-6
        ELSE
          WRITE(6,100) ITRACK
 100      FORMAT('ITRACK = ',I2,' NOT ALLOWED.')
          STOP
        ENDIF
        C1 = SQRT(TWOPI*AV*V)/SQRT(DT*2.)
        C2 = SQRT(TWOPI*AV*V)/(A*SQRT(DT*2.))
        OMEGC = V*BOMEGC
        OMEGR = V*BOMEGR
        OMEGS = V*BOMEGS
        IF (WLENG .NE. 0.) THEN
         OMEGD = V*TWOPI/WLENG
        ELSE
         OMEGD = 0.
        ENDIF
C
C       Compute elements of the state transition matrix
C
```

```
      P11 = EXP(-OMEGR*DT)
      P12 = (EXP(-OMEGR*DT) - EXP(-OMEGC*DT))/(OMEGC - OMEGR)
      P22 = EXP(-OMEGC*DT)
      P33 = EXP(-OMEGS*DT)
      P34 = (EXP(-OMEGS*DT) - EXP(-OMEGR*DT))/(OMEGR - OMEGS)
      P35 = -OMEGC*EXP(-OMEGS*DT)/((OMEGR-OMEGS)*(OMEGC-OMEGS))
     1      -OMEGC*EXP(-OMEGR*DT)/((OMEGS-OMEGR)*(OMEGC-OMEGR))
     2      -OMEGC*EXP(-OMEGC*DT)/((OMEGS-OMEGC)*(OMEGR-OMEGC))
      P44 = EXP(-OMEGR*DT)
      P45 = (EXP(-OMEGR*DT)-EXP(-OMEGC*DT))*OMEGC/(OMEGR-OMEGC)
      P55 = EXP(-OMEGC*DT)
C
C      Compute elements of the forcing matrix Q.
C      Q = E*B
C
      E11 = (1.-EXP(-OMEGR*DT))/OMEGR
      E12 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1     +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E22 = (1.-EXP(-OMEGC*DT))/OMEGC
      E34 = (1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS))
     1     +(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR))
      E35 = -OMEGC*(1.-EXP(-OMEGS*DT))/(OMEGS*(OMEGR-OMEGS)
     1        *(OMEGC-OMEGS))
     2      -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGS-OMEGR)
     3        *(OMEGC-OMEGR))
     4      -(1.-EXP(-OMEGC*DT))/((OMEGS-OMEGC)*(OMEGR-OMEGC))
      E44 = E11
      E45 = -OMEGC*(1.-EXP(-OMEGR*DT))/(OMEGR*(OMEGC-OMEGR))
     1      -(1.-EXP(-OMEGC*DT))/(OMEGR-OMEGC)
      E55 = E22
C
      Q11 = E12*C1*OMEGC
      Q21 = E22*C1*OMEGC
      Q32 = (E34+E35)*C2*OMEGC
      Q42 = (E44+E45)*C2*OMEGC
      Q52 = E55*C2*OMEGC
C
      NCALL = 0
C
C         Initialize the state variables to zero
C
      X1 = 0.
      X2 = 0.
      X3 = 0.
      X4 = 0.
      X5 = 0.
C
C      Initialize the sum squared variables to zero
C
      YLSQ = 0.
      YRSQ = 0.
    ENDIF
      NCALL = NCALL + 1
      CALL GAUSS(SEED,2,RN)
      X1 = P11*X1 + P12*X2 + Q11*RN(1)
      X2 = P22*X2 + Q21*RN(1)
```

```
      X3 = P33*X3 + P34*X4 + P35*X5 + Q32*RN(2)
      X4 = P44*X4 + P45*X5 + Q42*RN(2)
      X5 = P55*X5 + Q52*RN(2)
C
C      GET FIRST DERIVATIVES OF VERTICAL AND ALIGNMENT STATE
C         VARIABLES.
C
      X1D = -OMEGR*X1 + X2
      X2D = -OMEGC*X2 + C1*OMEGC*RN(1)
      X3D = -OMEGS*X3 + X4
C
C      Get second derivatives of vertical and alignment state
C        variables.
C
      X1DD = OMEGR**2*X1 - (OMEGR+OMEGC)*X2 + C1*OMEGC*RN(1)
      X3DD = OMEGS**2*X3 - (OMEGS+OMEGR)*X4 - OMEGC*X5
     1        + C2*OMEGC*RN(2)
C
C      Compute the discrete sine wave magnitude as a crosslevel
C        angle
C
      YDISCR = AMP*SIN(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSD   = AMP*OMEGD*COS(OMEGD*DT*(FLOAT(NCALL-1)))/A
      YDSDD  = -OMEGD**2*YDISCR
C
C      Compute the rail displacements
C
      YL    = X1   - A*(X3   + YDISCR)
      YLD   = X1D  - A*(X3D  + YDSD)
      YLDD  = X1DD - A*(X3DD + YDSDD)
      YR    = X1   + A*(X3   + YDISCR)
      YRD   = X1D  + A*(X3D  + YDSD)
      YRDD  = X1DD + A*(X3DD + YDSDD)
      YLSQ  = YLSQ + YL*YL
      YRSQ  = YRSQ + YR*YR
      YLRMS = SQRT(YLSQ/FLOAT(NCALL))
      YRRMS = SQRT(YRSQ/FLOAT(NCALL))
    RETURN
    END


    SUBROUTINE GAUSS(SEED,NPTS,V)
    IMPLICIT REAL*8(A - H, O - Z)
C
C      Subroutine GAUSS transforms a random sequence with uniform
C      probability density function into a random sequence with a
C      Gaussian probability function.  The standard deviation is
C      S and the mean is AM.  The random sequence is V, and it is
C      NPTS is length.
C
C      GAUSS is a modified version of a subroutine from the IBM
C      Scientific Subroutine Package.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C    REAL*8 SEED
```

```
      DIMENSION V(NPTS)
C
C        Set the mean and standard deviation
C
      DATA AM/0./S/1./
C
      DO 20 J=1,NPTS
        A=0.
        DO 10 I=1,12
          CALL RAN(Y,SEED)
          A=A+Y
   10   CONTINUE
        V(J)=(A-6.)*S+AM
   20 CONTINUE
      RETURN
      END


      SUBROUTINE RAN(Y,SEED)
      IMPLICIT REAL*8(A - H, O - Z)
C
C        Subroutine RAN generates a random sequence with a uniform
C        probability density function.
C
C      REAL*8 SEED,PI,XX
      DATA IT/0/
      IF (IT.EQ.0) PI=DACOS(-1.D0)
      XX=(PI+SEED)**5
      SEED=XX-DINT(XX)
      Y=SEED
      RETURN
      END
```

The vita has been removed from
the scanned document