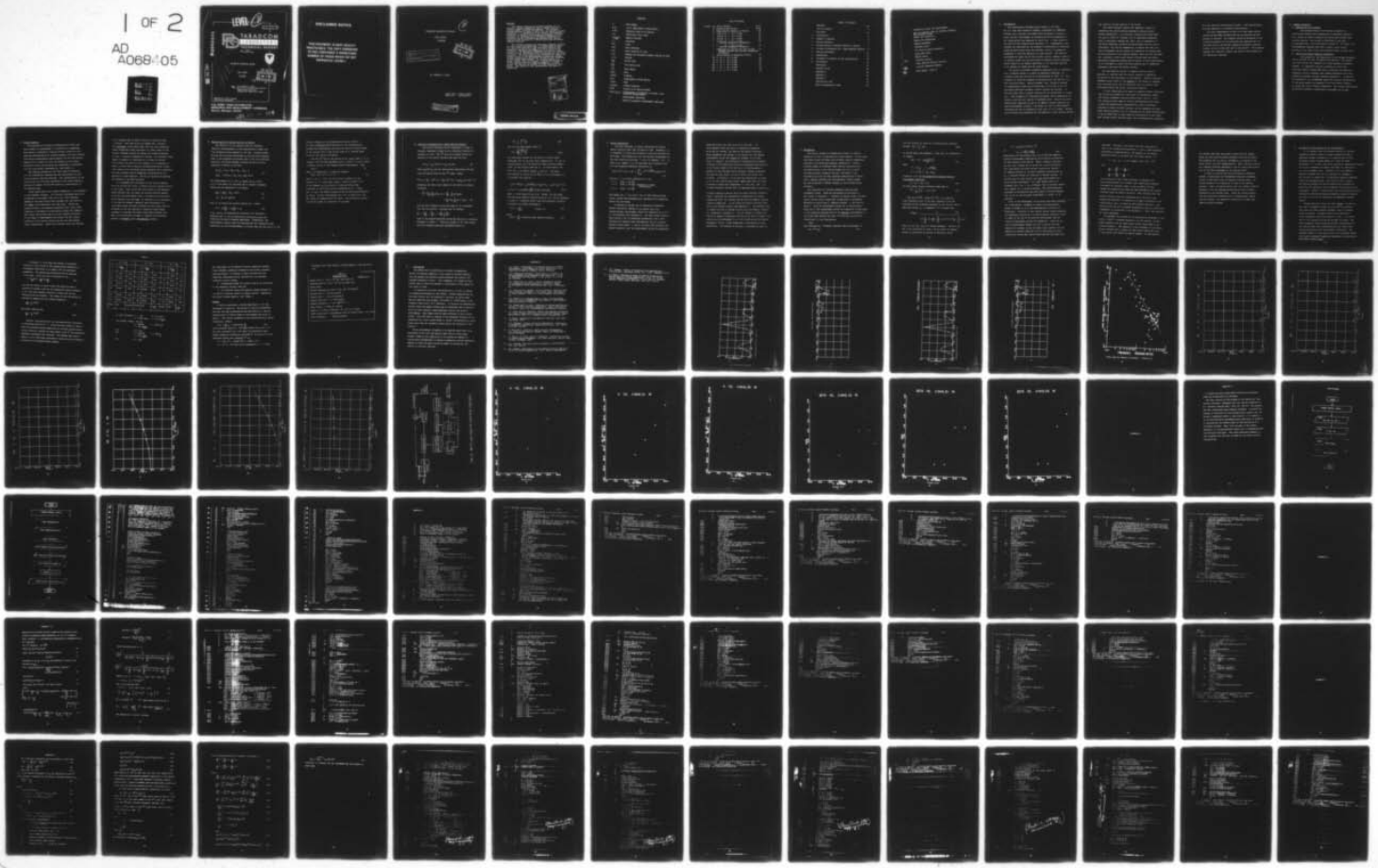
AD-A068 405 WAYNE STATE UNIV DETROIT MICH DEPT OF MECHANICAL ENG--ETC F/G 13/6
AUTOMOTIVE SUSPENSION CONTROL,(U)
OCT 78 H K SACHS

UNCLASSIFIED

DAAK30-77-C-0066
TARADCOM-TR-12377
NL

1 OF 2
AD A068405



LEVEL II

AD A068405

DDC FILE COPY

R and D

TARADCOM

LABORATORY

TECHNICAL REPORT

NO. 12377

AUTOMOTIVE SUSPENSION CONTROL

FINAL REPORT

Oct 1978

DDC
RECEIVED
MAY 8 1979
C

by Dr. Herbert K. Sachs

Wayne State University

CONTRACT NO. DAAK-30-77-C-0066

Approved for public release;
distribution unlimited.

U.S. ARMY TANK-AUTOMOTIVE
RESEARCH AND DEVELOPMENT COMMAND
Warren, Michigan 48090

79 05 07 096

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

(2)

AUTOMOTIVE SUSPENSION CONTROLS

FINAL REPORT

Oct 1978

DDC
RECEIVED
MAY 8 1979
C

The citation of commercial products in this report does not constitute an official endorsement or approval of such products.

DR. HERBERT K. SACHS

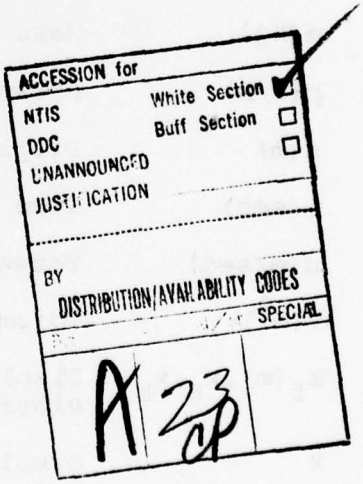
AMCMS CODE: 663621.4240011
PROJ. NO. : 1L663621A424

This document has been approved for public release and sale; its distribution is unlimited.

Abstract

This report contains the software package for an adaptive, optimal suspension control system relative to terrain random vibration disturbances. The proposed problem solution is shown to fall into two separate program categories: a) recognition of the terrain and parameter selection, by means of an on-board minicomputer or microprocessor, b) optimization of suspension parameters for arbitrary terrain configurations obtained from terrain statistics and executed on a centrally-located stationary computer facility.

The interface between the stationary computer facility and the on-board microprocessor is accomplished by means of a data bank prepared at the stationary facility and permanently stored in the memory of the on-board microprocessor. The suspension parameters are set by a servo-control unit on the vehicle which is activated by the microprocessor. The servo-control unit regulates the supply and release of air in the hydro-pneumatic suspension system, thereby increasing or decreasing the spring rate according to the optimal requirements. In a similar manner the damper orifice size is increased or diminished depending on the required effective damping parameter. If need arises, the vehicle can operate at fixed suspension parameters. The results of the investigation are shown in graph form.



PRECEDING PAGE BLANK

NOTATION

a	Wave mumber
A(m)	P.S.D. amplitude of road profile
$A(m^2)$	Effective area of air spring
a,b	Subscripts, axle, body
$c(\frac{N\text{-}sec}{m})$	Damping constant
C_i	Constants
F(N)	Force
H(m)	Axle clearance
h(t)	Input function of time
h,h*	Complex and conjugate complex funcion of time
i,n,m	Indices
$k(\frac{N}{m})$	Spring rate
$K(\frac{N}{m})$	Tire spring rate
ℓ(m)	Wave length
m(kg)	Mass
p(N/m)	Pressure
s(m)	Displacement across spring
t(sec)	Time
u(m/sec)	Forward velocity
$V(m^3)$	Volume of air spring system
$x_f(m), x_a, x_b$	Displacement in direction of travel, also vibration displacements
X	Displacement amplitude
y	Relative vibration displacement amplitude

List of Figures

Figure	1a	Actual Terrain	26-27
	1b	Reconstructed Terrain	28-29
	1c	Power Spectral Density of Terrain	30
	2	$(X\omega^2/g)$ vs u (H = 0.15m)	31
	3	$(X\omega^2/g)$ vs u (H = 0.21m)	32
	4	$(X\omega^2/g)$ vs u (H = 0.24m)	33
	5	$(X\omega^2/g)$ vs u (H = 0.21m) (Constant Suspension Parameters)	34
	6	Y vs u (H = 0.21m) (Constant Suspension Parameter)	35
	7	Comparison of $(X\omega^2/g)$ vs u (H = 0.21m) between constant and variable suspension parameters	36
	8	Block diagram of Suspension Control	37
	9a	K vs u (H = 0.15m)	38
	9b	K vs u (H = 0.21m)	39
	9c	K vs u (H = 0.24m)	40
	10a	ζ vs u (H = 0.15m)	41
	10b	ζ vs u (H = 0.21m)	42
	10c	ζ vs u (H = 0.24m)	43

Table of Contents

	Abstract	iii
	List of Figures	v
	Notations	vi
1-	Introduction	1
2-	Summary of Results	4
3-	Terrain Analysis	5
4-	Reconstruction of Terrain Profile on Vehicle	7
5-	Process of Averagins h(t), Power Spectral Density	9
6-	Design Constraints	11
7-	Methodology	13
8-	Discussion of Results of the Investigation	18
9-	Conclusions	23
	References	24
	Figures 1a-10b	26
	Appendix A	44
	Appendix B	60
	Appendix C	74
	Distribution List	94
	Report Documentation Page	98

α	Probability factor for displacement
γ	Ratio of specific heat at constant pressure and constant volume
δ	Symbol for variation.
Δ	Symbol for determinant
ζ	Damping parameter
η	Frequency ratio
μ	K/k ratio of spring constants
μ_1	m_a/m_b mass ratio
ρ	Weight factor
τ (sec)	Vibration period
ϕ	Power spectral density function
$\omega_i \frac{Rad}{Sec}$	Circular frequency Rad/Sec
$\Omega \frac{Rad}{Sec}$	Wave number, (also a)

1) Introduction

The investigation discussed herein began in 1975 when the U.S. Army Tank Automotive Command, predecessor to TARADCOM, entered into a contract with Wayne State University to pursue a study entitled "Development of Computerized Vehicle Suspensions". The investigation was completed in March 1976. A final report of the work was then submitted and approved, [1]. Subsequently a new proposal was submitted for the purpose of establishing a functional algorithm which would allow the automatic control of suspension parameters on vehicles operating in rough terrain. The vehicles considered are heavy and of the military variety; however, there is no inherent reason why the principles of adaptive control discussed herein should not be equally applicable to all vehicles required to be operable at speed over off-road terrain.

The notion of automatic suspension appeared first in connection with a complete design in a paper by Federspiel-Labrosse, [2]. A similar study was carried out by Westinghouse in 1965, [3]. The necessity of improving syspension characteristics was soon put on a more scientific basis. Works by Bender, Paul, Fenoglio, Karnopp (at Massachusetts Institute of Technology at the time [4,5,6,7]) considered feed-back automatic control systems for vehicles. It was next shown that a transfer function can be synthesized (n. Wiener [8]). Also Thompson [9] considered the optimal active suspension for bounce, pitch and roll control of passenger cars. None of the above publications suggested the use of an adaptive control employing on-board minicomputer circuits. The reason for this is that computer technology was not as advanced at that time, as it is today. Neither size nor price form obstacles for the adoption of such control methods

now, and will be much less so in the future.

The terms "optimal" control and "adaptive" control in connection with active vehicle suspension designs require further explanation. In a strictly literary sense there does not exist a "best" or "optimal" set of suspension parameters that is realizable, because the "best" type of suspension is one which completely isolates the sprung mass from any form of road shock. Any and all suspensions, no matter how soft, are force transducers and the softer the suspension the greater must be the excursions of the unsprung masses, which are not limitless. But, obviously, within given design criteria there are better performing suspension systems and the object of this investigation is to find means to select the best possible set of suspension parameters that meet the design criteria.

The term adaptive control is commonly used by control engineers to indicate that the control process is adapted to the source of the disturbance or perturbation. Another alternative feedback control where in the response of the system is compared with a desired output and the controller acts to minimize that difference between the actual and desired response.

In this investigation we employ an adaptive control algorithm that allows pre-sampling of the source of disturbance, namely the terrain roughness, from statistical data, [10,11]. From the infinitely wide range of terrain configurations that exist a large and substantially representative class of terrains, referred to herein as model terrains, can be assembled and their power spectral density (p.s.d.) functions can be placed side-by-side. It can be shown that in most cases the distribution of the roughness follows similar patterns which can be mathematically expressed

by error functions (discovered by Gauss). Such distributions are also referred to as Gaussian distributions.

For this investigation a total of 13 such model terrain statistics were made available and one was selected for the purpose of analysis. The computer algorithm dedicated to the on-board microprocessor was simulated and combined with the off-board terrain and optimal suspension parameter selection process, so as to close the look of the process. The following chapters will explain in detail the logic of the program and its operational features.

2) Summary of Results

a) Reconstruction of terrain.

The reconstruction of the terrain by means of a fast Fourier analysis and regeneration of measurable acceleration values at vehicle unsprung and sprung mass yielded very accurate results. The terrain as described in the form of tabulated data, [10] is plotted in Figure 19. Of course, the accelerometers employed must have a nearly linear output relative to the input within the frequency domain of their use.

b) Central Stationary Computer Simulations

The program for the computation of the Optimal parameter matrix has been written, documented and executed. The results have been plotted and compared with fixed suspension parameter systems. At comparable vertical acceleration levels the realizable speeds (speeds made good) of systems with adaptive suspension control exceeded those speeds, obtainable with fixed suspension parameter systems (passive suspension). If vehicles are operated at the recommended speeds the vertical accelerations of the sprung masses are expected to be sufficiently reduced to extend the life of vehicle components. But without statistically significant evidence no quantitative statement can be made.

3) Terrain Analysis

The signifiance of terrain configurations is that once their pattern is established it is possible to reconstruct them without a substantial loss of accuracy relative to the response characteristics of vehicles passing over such terrains. From terrain measurements (approximately 120 for each terrain sampled over a length of circa 120 meters) one can obtain equivalent Fourier series which are considered as representative for each terrain configuration. (see Figure 1).

The computer program for the "Fast Fourier Transform" [12] was successfully executed and the coefficients of each sine and cosine term were obtained. The reconstruction of the terrain from the 128 terms considered in the series showed an accuracy of between 97% and 99% relative to distribution of the measured data.

Since random data do not repeat themselves, it is necessary to take assembly averages. All terrains considered herein are considered to be ergodic, that is to say, that different samples of the same terrain when averaged over the length of the sample yield essentially the same value. Taking the squared values of the measured elevations from a mean level (squaring is necessary to avoid getting simply the algebraic mean value) and dividing the sum of all squared elevations by the number of measurements one obtains the mean square elevation of the terrain profile and the root mean square value, respectively. Hence for an ergodic input the averages

of all samples must be nearly alike for a specific type of terrain. Note that we do not assume that terrains are stationary, which would imply that the road elevations repeat themselves within multiples of a base length of the terrain, because terrains do change over longer distances. In fact, it would be necessary to collect an infinitely large number of samples to characterize all types of terrain. From a practical point of view, however, it is possible to select a finite set of terrain configurations which a particular vehicle will encounter during operation and to determine the most suitable form of suspension characteristics for operation in such terrains. The set of terrains so selected are referred to as model terrains.

It is assumed that the wheels of the vehicle passing over the terrain will stay in contact with the terrain profile at all times. Also, since the total length of the measured sample extends over 100 times the average distance between value sets the mean square value of elevation for each wheel may be assumed to be the same. In contrast to this assumption if one were to analyze a specific wave disturbance of a fixed wave length the wheel elevation of each wheel would be different at any one time and it therefore would become necessary to consider the phase relationship between each wheel (the distinction is made to indicate the difference between a stochastic and a deterministic input),

4) Reconstruction of Terrain Profile on Vehicle

The vehicle is to be equipped with low frequency sensitive accelerometers on the superstructure (body) and high frequency accelerometers on the axle (unsprung mass). With m_b the pro-rated sprung mass, m_a the pro-rated unsprung mass, k the suspension stiffness rate, K the tire stiffness rate and c the shock absorber constant we can write the equations of motion in the form

$$\begin{aligned} m_b \ddot{x}_b + k x_b + c\dot{x}_b - kx_a - c\dot{x}_a &= 0 \\ m_a\ddot{x}_a + (K+k)x_a + c\dot{x}_a - kx_b - c\dot{x}_b &= Kh(t) \end{aligned} \qquad (1)$$

The displacements in (1) are x_b (body) and x_a (axle). h(t) is the wheel lift and drop due to terrain roughness. Adding the two equations (1) we obtain

$$m_b\ddot{x}_b + m_a\ddot{x}_a + Kx_a = Kh(t) \qquad (2)$$

$$x_a = \int_\tau \left(\int_\tau \ddot{x}_a dt\right)dt \qquad (3)$$

From (2) we obtain the terrain profile h(t), namely

$$h(t) = \ddot{x}_b\left(\frac{m_b}{K}\right) + \ddot{x}_a\left(\frac{m_a}{K}\right) + x_a \qquad (2a)$$

If $\ddot{x}_b$ and $\ddot{x}_a$ are measured and converted into electronic impulses then (3) can be obtained by integrating twice and inverting the sign on summing amplifiers. Alternately, the accelerometer outputs could be digitized and the integration done numerically on the microprocessor. In either case the the result of (2a)

yields a reading of the reconstructed terrain profile. In order to demonstrate the accuracy of the reconstruction process the investigators converted the spatial Fourier series of the terrain into a time series based on a constant speed motion of the vehicle as follows.

Let the n^{th} term of the series be sin (nax_f) where n is an integer, a is the wave parameter ($a=2\pi/\ell$) where ℓ is wave length in meters) and x_f is the forward motion displacement. Then

$$x_f = u.t \tag{4}$$

where u is speed and t is time the argument

$$nax_f = n(2\pi\ u/\ell)\cdot t = n\cdot\omega\cdot t \tag{5}$$

where in (5) $\omega = 2\pi u/\ell$ is the circular frequency of the harmonic of "ℓ" meter wave length. Referring back to (1) we can compute $\ddot{x}_a$, $\ddot{x}_b$ relative to each and every term contained in the Fourier series representing h(t). These are, in fact, the signals sensed by the on-board accelerometers. $\ddot{x}_b$ and $\ddot{x}_a$ is then reintroduced into (2a) to obtain h(t) and the output is compared with the input. The correlation proved to be on the order of a fraction of a percent.

5) Process of Averaging h(t), Power Spectral Density.

The terrain elevation may be expressed in terms of discrete (measured) values or as a continous function of distance or time. The n^{th} term of a Fourier series expansion of the terrain function may have the form

$$h(t)_n = h_{nc} \cos n\omega t + h_{ns} \sin n\omega t \qquad (6)$$

where h_{nc} and h_{ns} are the coefficients associated with the sine and cosine terms of the n^{th} order. Hence

$$h^2(t)_n = h_{nc}^2 \cos^2 n\omega t + h_{ns}^2 \sin^2 n\omega t + h_{nc}h_{ns} \sin 2n\omega t \qquad (7)$$

Averaging the above with respect to one period of length τ we obtain

$$\bar{h}_n^2 = \frac{h_{nc}^2}{\tau} \int_\tau \cos^2 n\omega t \, dt + \frac{h_{ns}^2}{\tau} \int_\tau \sin^2 n\omega t dt + \frac{h_{nc}}{\tau} h_{ns} \int_\tau \sin 2 n\omega t dt \qquad (8)$$

Now the last integral on the left side of (8) integrated over the full period τ vanishes and (8) becomes

$$\bar{h}_n^2 = \left(\frac{h_{nc}^2}{2} + \frac{h_{ns}^2}{2}\right) = \left(\frac{h_{nc}^2 + h_{ns}^2}{2}\right) \qquad (9)$$

Each of the squared amplitude values $\bar{h}_n^2$ belongs to a discrete frequency of order n . Then, the average of all such values over the frequency spectrum considered herein is

$$\bar{h}^2 = \frac{1}{n} \sum_{1}^{120} \bar{h}_n^2 \qquad (10)$$

and its root mean square value is

$$\bar{h}_n = \sqrt{\frac{1}{n} \sum_{1}^{120} \bar{h}_n^2} \qquad (10a)$$

For each model terrain we can obtain its mean square amplitude (10) or its root (10a) respectively. If (10) is plotted versus $(n\omega)^2$ we obtain the power spectral density curve of the terrain (see Fig.1). In this Figure 1 we plot the $_{10}\log\ (\bar{h}^2/a)$ versus $_{10}\log\ (a)$. The dense distribution of points can be seen to lie in proximity of a straight line given by

$$_{10}\log\ (\bar{h}_1^2/a_1) - {}_{10}\log(\bar{h}_n^2/a_n) = ({}_{10}\log\ a_1 - {}_{10}\log\ a_n) f \qquad (11)$$

$$\text{or } \log f = ({}_{10}\log[\bar{h}_1^2\ a_n / \bar{h}_n^2\ a_1]) / {}_{10}\log(a_1/a_n) \qquad (11a)$$

where f is the slope of the line. Bender [13] has shown that the power spectral density function (11) can be expressed in terms of frequency rather than wave parameter yielding a straight line image expressed by

$$\phi(n\omega) = -\frac{\bar{h}u}{(n\omega)^2} \qquad \text{(Figure 1c)} \qquad (12)$$

where

$$\phi(n\omega) = \frac{h_n^2}{2\omega} = \text{discrete power spectral density.} \qquad (13)$$

6) Design Constraints

The axle clearance is usually determined by several design criteria other than softness of ride. Therefore, it is necessary to avoid the incidence of the axle striking the axle stops. The probability that the maximal displacement of the axle relative to the body Y will lie between $\pm \alpha\bar{y} = \pm H$ where α is a number, H is the available axle clearance and $\bar{y}$ is the expected suspension deflection, is

$$\text{Prob } [-\alpha\bar{y} \leq y(t) \leq \alpha\bar{y}] = \int_{-\alpha\bar{y}}^{\alpha\bar{y}} e^{(-(y/\bar{y})^2/2)} dy/\bar{y}(2\pi)^{1/2} \tag{14}$$

and for $\alpha = 1,2,3$ this probability is [14]

$\alpha = 1$ Prob. = 68.3%
$\alpha = 2$ Prob. = 95.4% — Gaussian or normal distribution
$\alpha = 3$ Prob. = 99.7%

This means for $\alpha = 3$, in only 3 out of 1000 working cycles would Y exceed the allowable axle clearance H and therefore strike the axle stops.

In seeking an optimal set of suspension parameters, design constraints imposed on the system limit the average maximal displacement across the suspension $\bar{y}$ to $1/\alpha = 1/3$ of available axle clearance H. For each terrain profile there exists a set of suspension parameters k and c that minimizes the maximal body accelerations $|\ddot{x}_b|$ at a certain average forward speed u and at the same time satisfies the design constraint that the displacement across the suspension

should be within the limit H 99.7% of the time. At other speeds,u,there are sets of suspension parameters which are not optimal, yet satisfy the design constraint. These parameters are optimal to the extent that the maximal displacement across the suspension elements is utilized. Since, in general, the acceleration transmissibility depends inversely on the deflection across the suspension and directly on the forward speed of the vehicle,it is advantageous to exhaust all the available axle clearance (without frequently striking the axle stops), so as to obtain the sofest ride possible. In Figures 2,3,4 we show the acceleration in units of "g" of the vehicle body versus "speed made good" for three differently chosen axle clearances H = .15m,.21m, .24m. All of these represent optimal data, as explained above, relative to variable suspension spring rates and shock absorber constants. Figure 5 portrays the acceleration transmissibility versus "speed made good" for constant spring rate and damping constant (passive suspension). In all cases the results obtained refer only to one terrain profile (see Appendix 1). The formulas used to compute the set of optimal spring rates and damping constants are presented in Appendix 2. For the purpose of showing the effects of speed on suspension deflection we show in Figure 6 the relative axle displacement $\bar{y}$ vs. u for fixed spring and damping parameters. Figure 7 shows comparison of acceleration transmissibility. The meaning of the data is discussed on page 18.

7) Methodology

The control process is schematically shown in Figure 8 (Figure 22 of [1]) in the form of a block diagram. At the right hand lower corner are shown the off-board logistic operations executed on any stationary computer installation. Given a discrete number of different model terrain statistics and the descriptions of the vehicle masses and tire or track stiffness optimal parameter matrices (see Table 1) are obtained which give the best possible combinations of spring rates, damping constants and corresponding forward speeds. The data are placed in memory storage on the on-board microprocessor.

The computation of optimal parameters requires many programming steps. These are documented and attached to this report in Appendix 2. Speaking in general terms, the mean square terrain profile amplitude is amplified or attenuated depending on spring rate k, damping constant c, and forward speed u. Only two displacement values are of significance. These are a) the displacement across the suspension as mentioned in the foregoing paragraph and b) the absolute displacement of the sprung mass because the peak accelerations are directly proportional to that displacement.

Thus:

peak acceleration = frequency squared times displacement or

$$|a_c| = \omega^2 \cdot |X_b| \qquad (14)$$

and (14) written in units of g (gravitational constant)

becomes $|\bar{a}_c| = \frac{\omega^2}{g} |X_b|$ (14a)

It can be shown (see appendix 2) that $|X_b|$ is a function of $|Y|$ namely

$$|X_b| = |Y| \frac{(1+4\zeta^2\eta^2)^{1/2}}{\eta^2} \quad (15)$$

where

$$\eta^2 = \omega^2/\omega_n^2 \cdot \zeta = c/2\sqrt{km_b} ;$$

$$\omega^2 = 4\pi^2 u^2/\ell^2; \ \omega_n^2 = k/m_b \ .$$

Instead of Y we use <u>the permissible average relative displacement amplitude</u>

$$\bar{y} = H/\alpha = H/3. (\alpha=3) \quad (16)$$

The mean square terrain elevation amplitude is:

$$\bar{h}^2 = \frac{1}{2m} \sum_{1}^{m} h_n^2 \ ; \ (m = 120). \quad (17)$$

The ratio $\bar{y}^2/\bar{h}^2$, where $\bar{y}^2 = H^2/9$, is a specific number depending only on the axle clearance H and the measured terrain roughness expressed by the Fourier series amplitude values h_n. In appendix 2 it is shown that:

$$(\bar{y}^2/\bar{h}^2) = \frac{\eta^4}{[\eta^4 \frac{\omega_b^2}{\Omega_a^2} - \eta^2[1+\frac{k}{K} + \frac{\omega_b^2}{\Omega_a^2}]+1]^2 + \zeta^2\eta^2[\eta^2(\frac{k}{K} + \frac{\omega_b^2}{\Omega_a^2})-1]^2} \quad (18)$$

where in (18) $\Omega_a^2 = K/m_a$ is a known parameter. Solving (18) for η^2 as a function of k and ζ we can infer the forward speeds u_i pertaining to the set of solutions, since

$$\omega^2 = 4\pi^2 u_i^2/\ell^2 = (k_i/m_b) \cdot \eta_i^2$$

or

$$u_i = \eta_i \ell \sqrt{k_i}/(2\pi\sqrt{m_b}) \qquad (19)$$

Substitution of the solution into (15) yields the absolute expected displacement amplitude $\bar{x}_b$ which should be minimized for best possible ride quality. Again, assuming ζ constant one can find a unique set of values for u and k satisfying this requirement and after further iteration by varying ζ (or damping constant c) the best parameter configuration u_o, k_o, c_o emerges whereby the subscript o denotes optimum. It follows that for each terrain only <u>one</u> forward speed will yield optimal performance. For other speeds there is a parameter set k (u), c (u), different from the optimal values, that renders (15) a relative minimum. The above described procedure of optimization has been coded in Fortran language and, as far as the principal investigator is aware, it is orginial.

Prior to the development of the above described procedure the investigators attempted to employ the optimization method of Bender et al of the Massachusetts Institute of Technology. According to that method optimal system responses are obtained by minimizing a so-called penalty function P which is equal to the sum of the absolute displacement and the relative displacement across the suspension. Since the relative displacements depend not only on spring rate and damping but foremost on the available axle clearance one can develop a so-called trade-off curve of acceleration transmissibility versus mean terrain amplitude and available axle

clearance. Obviously, the greater the axle clearance the less is the trasmitted acceleration, or for a fixed acceptable acceleration transmissibility, the greater is the vehicle speed with which it can move over the rough terrain. If

$$P = \rho\bar{x}_b + \alpha\bar{y} \ , \ (\rho \text{ is a number, } 10^{-4}<\rho<10^{4}) \tag{20}$$

then

$$\frac{dP}{dk} = \rho \frac{d\bar{x}_b}{dk} + \alpha \frac{d\bar{y}}{dk} = o \tag{21}$$

$$\frac{dP}{dc} = c \frac{d\bar{x}_b}{dc} + \alpha \frac{d\bar{y}}{dc} = o$$

yield values of k and c that minimize, or maximize (20). The differentiation (21) yields a very complicated algebraic polynomial of the eight power in the variables and even though the equations are coded for data processing the results obtained cannot be readily verified. They indicated that the penalty function could be represented by a relatively flat surface and therefore is insensitive to small variations in the coefficients k and c on which $\bar{x}_b$ and $\bar{y}$ depend implicity. The results yield one set of optimal parameters k_o and c_o, practically, independent of variations in the parameter c. Thus, this approach was later discarded.

To implement the results of the optimization developed in this study, a Table 1, prepared especially for the vehicle, is then placed together with the mean square terrain amplitudes in the on-board computer. The computer is also programed to reconstruct terrain profiles and to obtain the mean square amplitude value of the terrain over which the vehicle passes. It then selects

the nearest amplitude value model terrain and the nearest speed for which pre-calculated parameter sets are available. The parameter set (k_o and c_o) information is relayed to the servo control unit for implementation. Then, the servo control unit changes inflation pressures of the air spring and orifice size of the shock absorbers to the desired levels.

In order to execute the control function a comparison between instantaneous and recommended suspension parameter values is made based upon the measured forward speed. The optimal speed will be displaced in sight of the vehicle operator. Then, the servo-control unit will change inflation pressure of the air bellows springs and orifice of shock absorbers to the desired level. Since the terrain profile will be continually scanned within a predetermined length (say 100 meters), the operation constitutes a closed loop adaptive control process.

8) Discussion of the Results of the Investigation

It is apparent that the economic benefits of optimal suspension control reside in either extended periods of operation between overhauls, or in reduced time to cover distances (higher average speed) or in both of these advantages. It is also clear that performance improvements require an initial investment (no matter how small in relation to the total equipment cost). Whether or not the expected benefits outweigh the cost is a question of judgement. But the gain in speed over a conventional passive suspension system can be demonstrated. Let terrain profile, sprung and unsprung masses, tire spring rate and axle clearance be equal for two vehicles, one with passive and one with active suspension. Figure 7 clearly shows that the acceleration transmissibility through the passive suspension system is greater than that of the active suspension at comparable forward speeds.

Transmissibility ratios are never unique. By this we mean that bi-quadratic equations can yield two distinct real roots for each circular frequency value or forward speed considered. Frequency spectra have maxima at a finite frequency value and minimum at o and ∞. On either side of the frequency giving the maximum lie higher and lower frequency values yielding the same transmissibility. For reasons other than transmissibility one cannot make springs arbitrarily soft and dashpots ineffective. The parameter search is then restricted to the frequency domain that yields reasonable suspension parameters at practically realizable forward speeds.

In figures, 9, 10 we show the effects of parameter variation k and ζ, which is the dimensionless parameter of the damping coefficient c, on speed u for the optimized suspension. The spring force variations may be realized in two ways. Since the rate is defined as [15]

$$k = \frac{d}{ds}(pA) = \frac{dp}{ds} A + p \frac{dA}{ds} \tag{22}$$

one has the choice to alter either the effective area A relative to stroke s and let the pressure be nearly constant or to vary the pressure relative to stroke and let the effective area be constant. The former of the two choices is realized by shaping the air cushion requiring

$$\frac{dA}{ds} = \frac{k}{p} \ (m^2/m) \tag{23}$$

The latter implies that

$$\frac{dp}{ds} = \frac{k}{A} \ (N/m^3) \tag{24}$$

However, the spring rates are non-monotonially increasing or decreasing function of u giving the data listed in Table I. From the recorded forward speed and the recorded, reconstructed, terrain function yielding the mean square value $\bar{h}^2$ the appropriate suspension parameters are selected. The amended data matrix (Table 1) will yield body acceleration values which are relatively best for the recorded forward speeds.

Table I

H = .15m			H = .21m			H = .24m		
u	$k=\frac{dp_A}{ds}$	c	u	$k=\frac{dp_A}{ds}$	c	u	$k=\frac{dp_A}{ds}$	c
8	1000	1132	8	2000	267	8	2000	537
12	3000	980	12	4000	755	12	5000	422
16	6000	462	16	8000	537	16	9000	566
20	9000	1132	20	12000	654	20	14000	706
24	14000	706	24	17000	778	24	11000	3129
28	18000	800	28	24000	924	28	28000	998
m/sec	N/m	N sec/m	m/sec	N/m	$N\ \frac{sec}{m}$	m/sec	N/m	$\frac{N\ sec}{m}$

At axle clearance $H = .15m$ $u = 16$ m/sec,
$k = 3000$ (N/m), $c = 1132\ \frac{N\ sec}{m}$
are optimal values. So are
$u = 8m/sec$, $k = 2000$ N/m
$c = 267\ \frac{N\text{-}sec}{m}$
for $H = .21m$
and $u = 12m/sec$, $k = 5000\ \frac{N}{m}$,
$c = 422\ \frac{N\ sec}{m}$
for $H = .24m$.

The improvement can be deduced from the comparison between fixed (assumed) suspension parameters and optimal parameter response, Figure 7. In closing it shall be mentioned that remarkable advantages can be realized with the proposed suspension control system.

1) Recommended speeds for optimal control can be posted at the dashboard (digital read-out)

2) For whatever reason the operator deems necessary he can switch to passive control or manual control, employing the control scheme based on (24), Table I.

Example

A vehicle negotiates a terrain described in [10] and displayed in Figure 1a. The terrain is Fourier transformed and axle and body accelerations are monitored for a vehicle forward speed of 21m/sec based on the assumed data given in Table 2. The terrain roughness is now reconstructed by direct integration, namely,

$$h(t) = \frac{ma}{k} x_a + (x_a dt)dt + x_b \frac{m_b}{k} \qquad (2a)$$

giving terrain figure 1b. The mean square value (10), (17) are then introduced into (18) where the permissible mean square suspension deflection amplitude $\bar{y}^2$ is given 1/9 of the available square axle clearance H^2 e.g.

$H = .21m$, $H^2 = .0442m^2$ $H^2/0 = .00049 = \bar{y}^2$;

$\bar{y} = .07$ in. For each set of parameters k, ζ, η^2 that

satisfies (18) there exists a forward speed u, that satisfies (19).

Table 2

Technical Data

Tandem Axle

Sprung Mass M: 4.087×10^3 kg per wheel set

Unsprung Mass m: 3.043×10^2 kg per wheel set

Sprung Mass

Natural frequency 60 cpm = 1 Hz; $\omega_n^2 = 39.48/\text{sec}^2$

Spring rate $k = 161{,}354$ kg/sec^2

Spring rate $K = 1{,}201{,}376$ kg/sec^2

Damping Constant c = 17,976 kg/sec

Damping Parameter $\xi = c/2\,(kM)^{1/2} = .7$

Wave Length L = 100m (m = meter)

Speed u = 25 m/sec (maximum), 20, 15, 10, 5 m/sec

Double amplitude of fundamental wave of length 100m: .1 m (10cm)

Axle clearance: (Static-Source)

9) Conclusions

Any person who is called upon to review a prospective design for possible adoption in the course of product planning, must ask whether the benefits gained outweigh the costs of the proposed suspension control. This judgement will depend on many factors most of which are unknown to the author of this report at this point in time.

In connection with such considerations it is fair to assess the expected performance of the system. Current expectations are that best results will be obtained in terrains for which power spectral desities vary between $.1m^2$/rad/m to $.0001m^2$/rad/m in the frequency range from 2 to 6 rad/meter. In terrains of substantially greater roughness manual control may be preferable and in terrains which are much smoother fixed suspension control may be more advantageous. Both modes could be made available in this control system. Also note that no changes in the parameter settings of the suspension will occur unless speed or terrain configurations or both change more than set threshold levels during the operation of the vehicle.

Actual perfromance increases to be expected would have to be determined at the time that design trade offs are being made. However, based on the simplicity of the system as compared to others being contemplated, an adaptive suspension control mechanism appears to offer the most attractive means of optimizing ride control in military vehicles.

REFERENCES

1. H.K. Sachs, "Development of Computerized Active Vehicle Suspensions", Final Report No. 12126, March 1976, U.S. Army Tank Automotive Command, Warren, Michigan.

2. G.M. Federspiel-Labrosse, "Contribution a l'Etude et au Perfectionement de la Suspension des Vehicules," J. de la Suspension des Vehicules", J. de la SIA - F.I.S.T.A. pp. 427-436.

3. W.O. Osbon and L.R. Allen, "Active Suspension Systems for Automotive Military Vehicles", Scientific Paper 65-IDI-HYDRA-PI 1965, Westinghouse Res. Lab., Pittsburgh, PA.

4. I.L. Paul and E.K. Bender, "Active Vibration Isolation and Active Vehicle Suspension", PB.173648, Report DSR-76109-1, -2 (Bibliography), 1966, U.S. Clearinghouse, Springfield, VA.

5. E.K. Bender, D.C. Karnopp and I.L. Paul, "On the Optimization of Vehicle Suspensions Using Random Process Theory", ASME Publication 67-Tran-12.

6. E.K. Bender and I.L. Paul, "Analysis of Optimum and Preview Control of Active Vehicle Suspensions", Report DSR-76109-6, PB 176137, 1967, U.S. Clearinghouse, Springfield, Virginia.

7. I.L. Paul and B.F. Fenoglio, "Design and Computer Simulation of a Near Optimum Active Vibration Isolation System", Report DSR-76109-8, 1968, U.S. Document Center, Springfield, VA.

8. N. Wiener, Cybernetics; 2nd addition, New York, MIT Press and Wiley, 1961.

9. A.G. Thompson, "Design of Active Suspensions", Institution of Mechanical Engineering Automotive Division, Vol. 185/36, 1971, pp. 553-563.

10. S. Heal and C. Cicillini, "Micro Terrain Measurements", U.S. Army Tank-Automotive Command, Report No.45866L-RRC-9. Warren, Michigan.

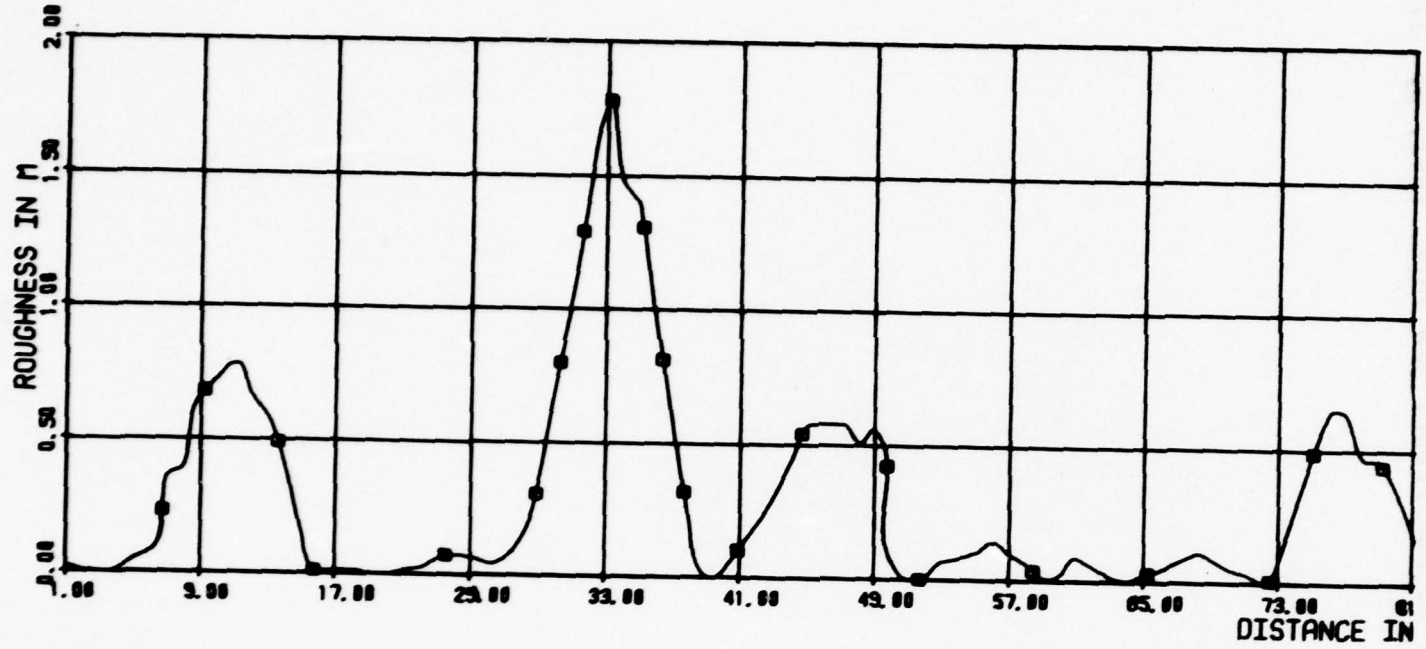
11. F. Kozin, L.S. Cote, and S.L. Bogdanoff, "Statistical Studies of Stable Ground Roughness", U.S. Army Tank-Auto. Center, Rept. No.8391, November, 1963

12. E.O. Brigham, "The Fast Fourier Transform", Prentice Hall, 1977, pp. 148-163.

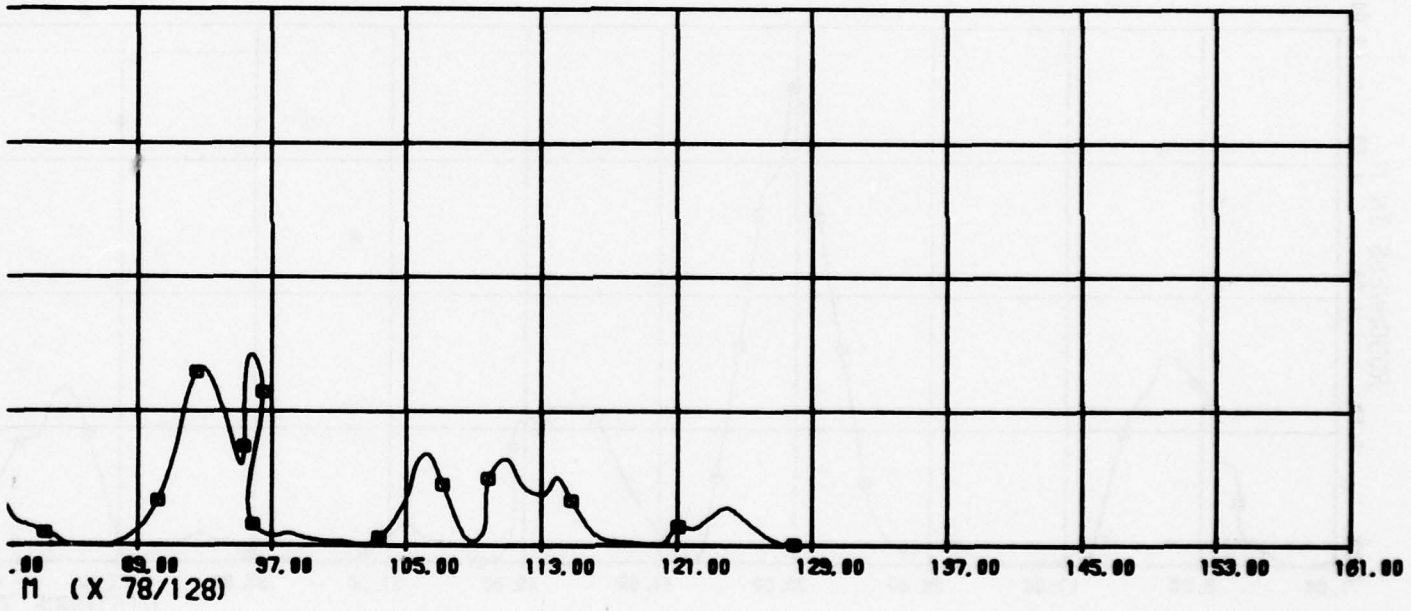
13. E.K. Bender, "Optimization of the Random Vibration Characteristics of Vehicle Suspensions", Ph.D. Thesis, Masachusetts.

14. W.T. Thomson, "Theory of Vibrations with Applications, Prentice Hall, Englewood Cliffs, N.J., 1972, Chapters 3,5,10.

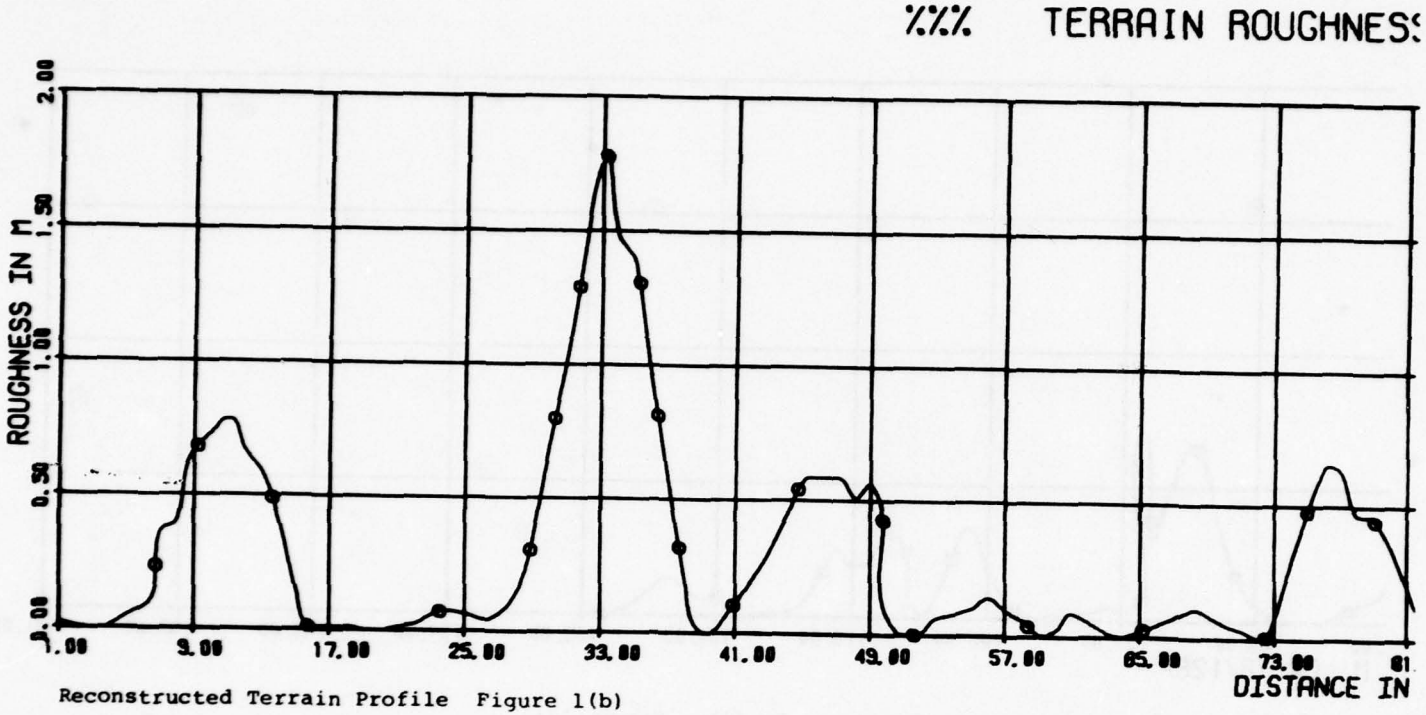
15. H.K.Sachs, "Theoretical Study on Elastic Properties of Bellows Type Air Spring Suspensions", Proc. of 7th Mid-western Mechanical Conference, G.E. Lay and L.E. Malvern, Editors, Plenum Press, New York, 1961, pp. 112-127.



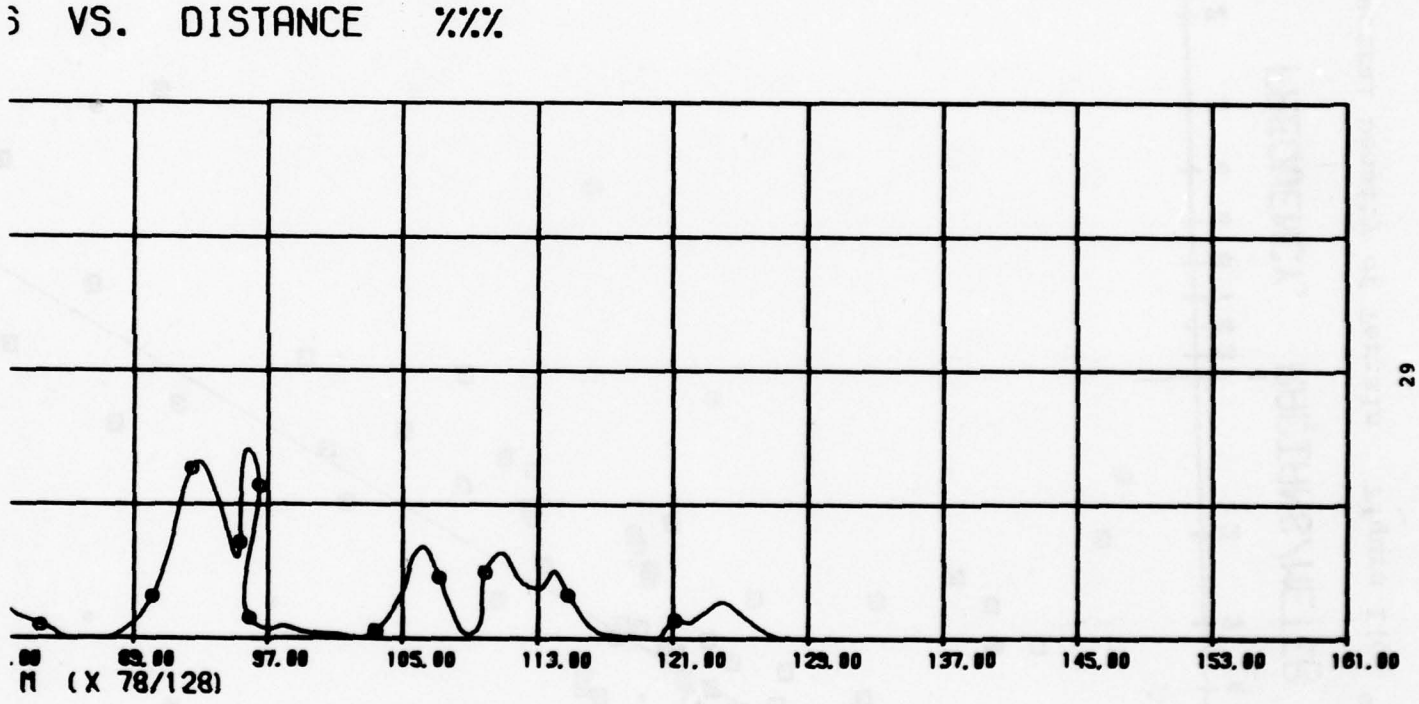
Actual Terrain Profile Figure 1(a)



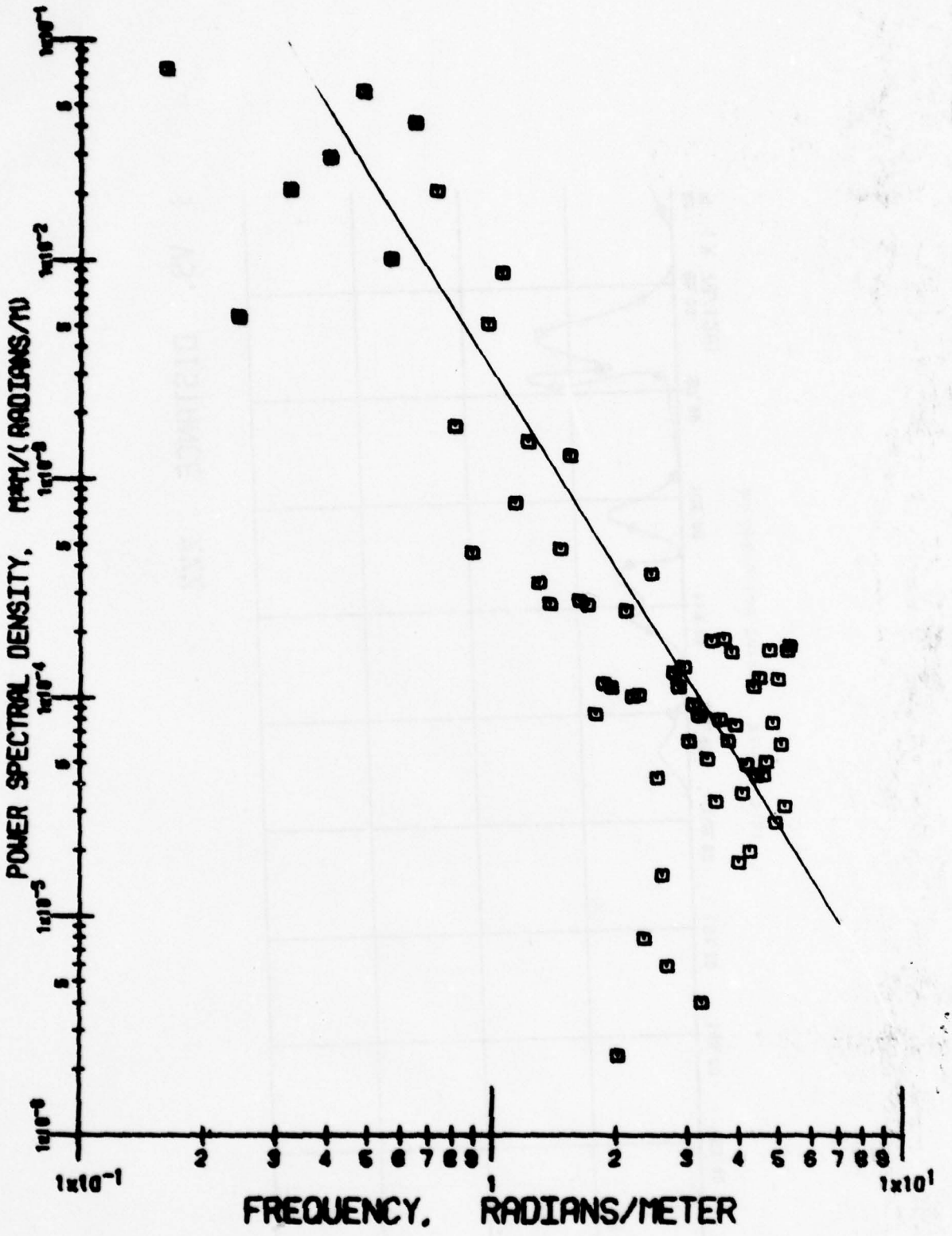
Actual Terrain Profile Figure 1(a)



Reconstructed Terrain Profile Figure 1(b)



Recontructed Terrain Figure 1(b)



Power Spectral Density of Terrain Figure 1(c)

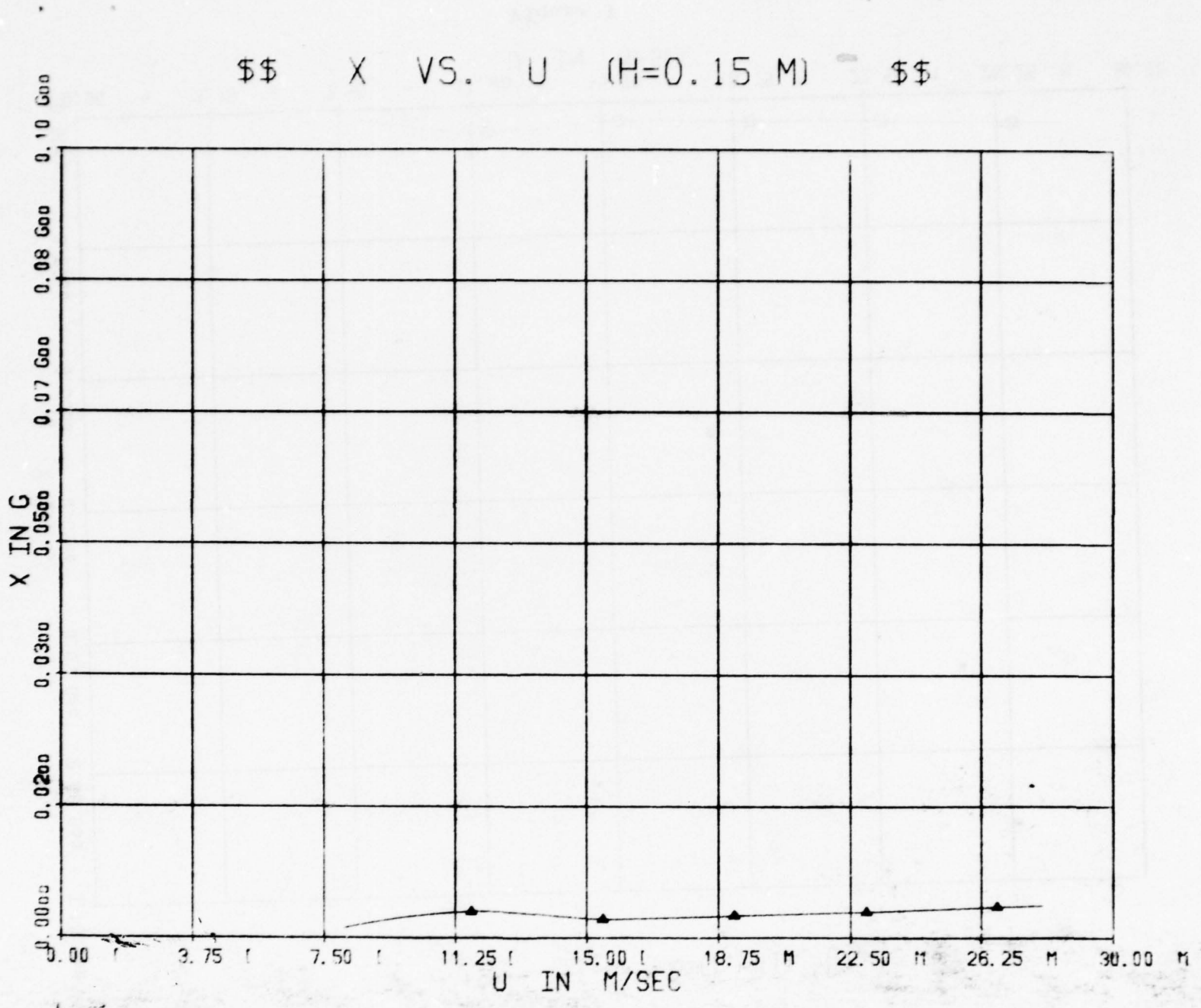

$$ X VS. U (H=0.15 M) $$
X IN G
0.0000
0.0200
0.0300
0.0500
0.07 G00
0.08 G00
0.10 G00
0.00
3.75
7.50
11.25
15.00
18.75 M
22.50 M
26.25 M
30.00 M
U IN M/SEC

Figure 2

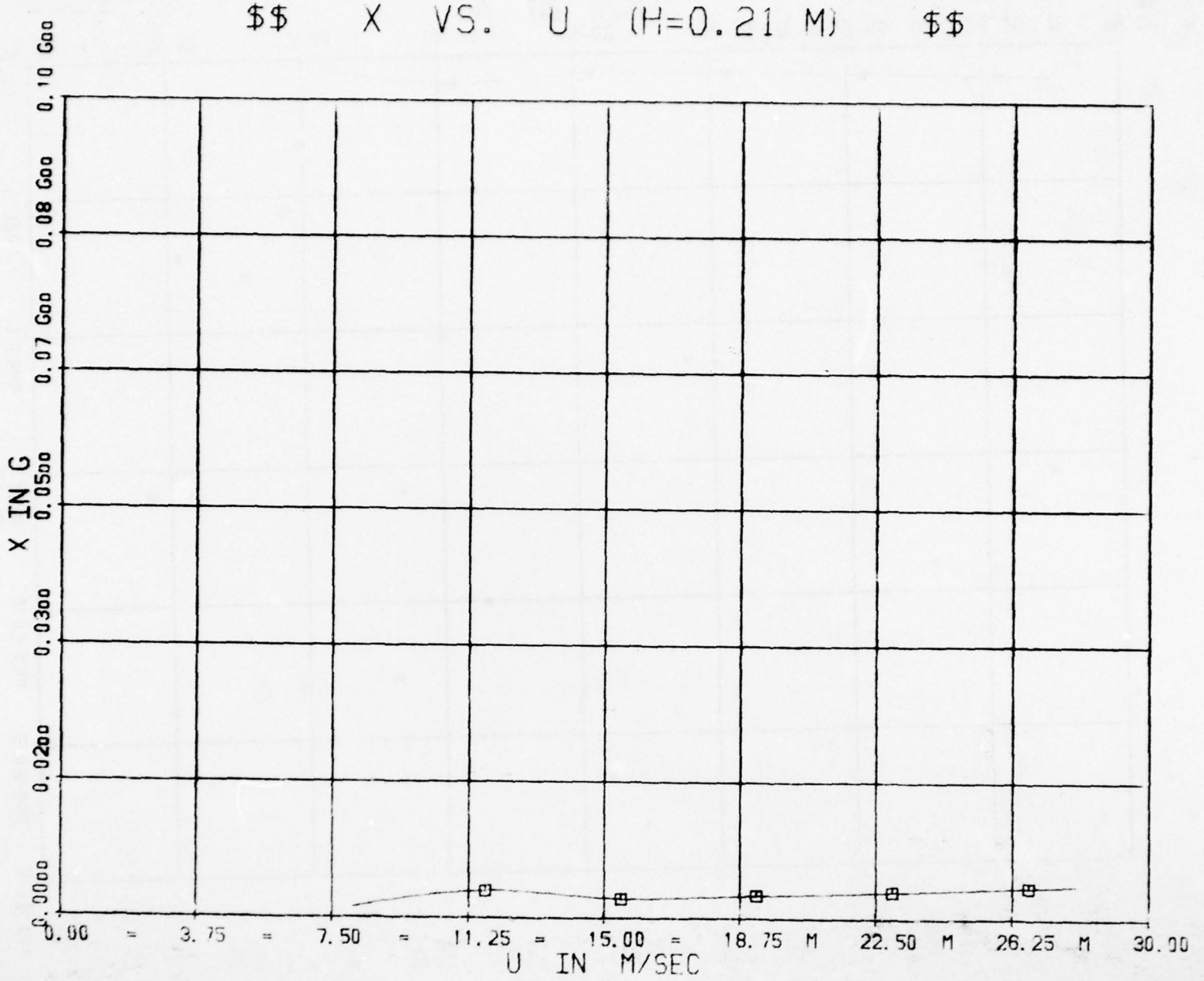

$$ X VS. U (H=0.21 M) $$
X IN G
0.0000
0.0200
0.0300
0.0500
0.0700
0.0800
0.1000
0.00 = 3.75 = 7.50 = 11.25 = 15.00 = 18.75 M 22.50 M 26.25 M 30.00
U IN M/SEC

Figure 3

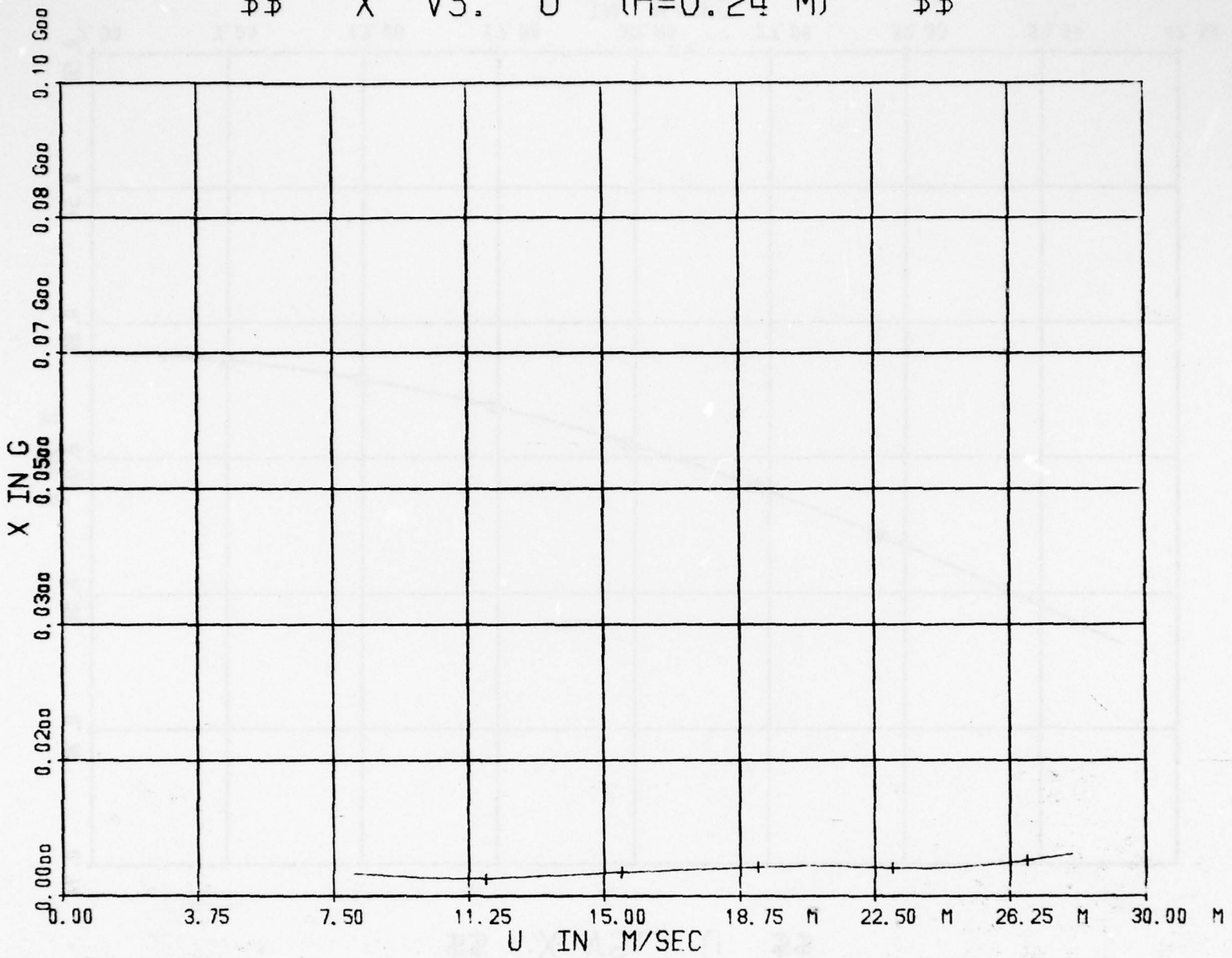

$$ X VS. U (H=0.24 M) $$
X IN G
0.0000
0.0200
0.0300
0.0500
0.0700
0.0800
0.1000
0.00
3.75
7.50
11.25
15.00
18.75 M
22.50 M
26.25 M
30.00 M
U IN M/SEC

Figure 4

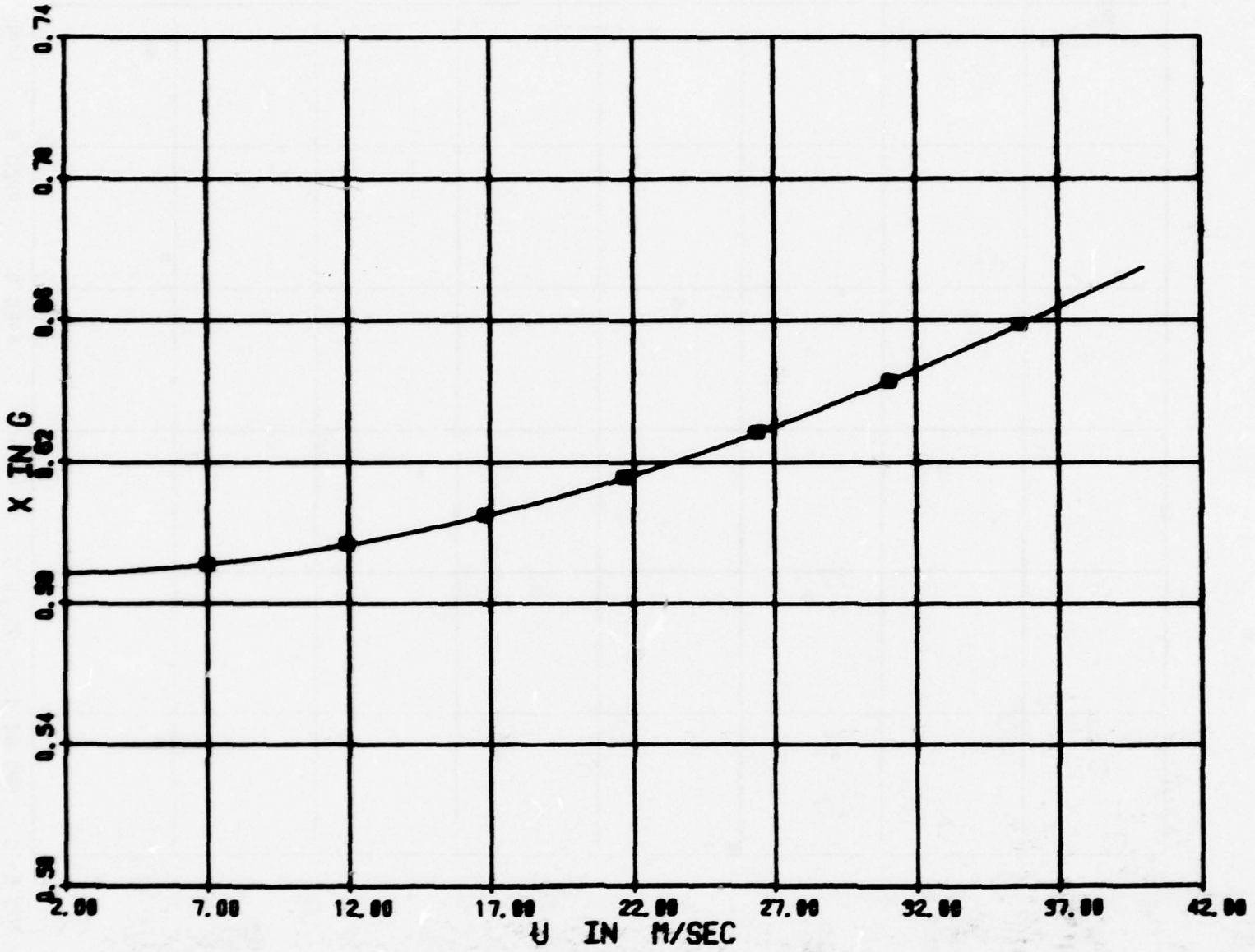

$$ X VS. U $$
0.74
0.70
0.66
0.62
0.58
0.54
0.50
X IN G
2.00
7.00
12.00
17.00
22.00
27.00
32.00
37.00
42.00
U IN M/SEC

Figure 5

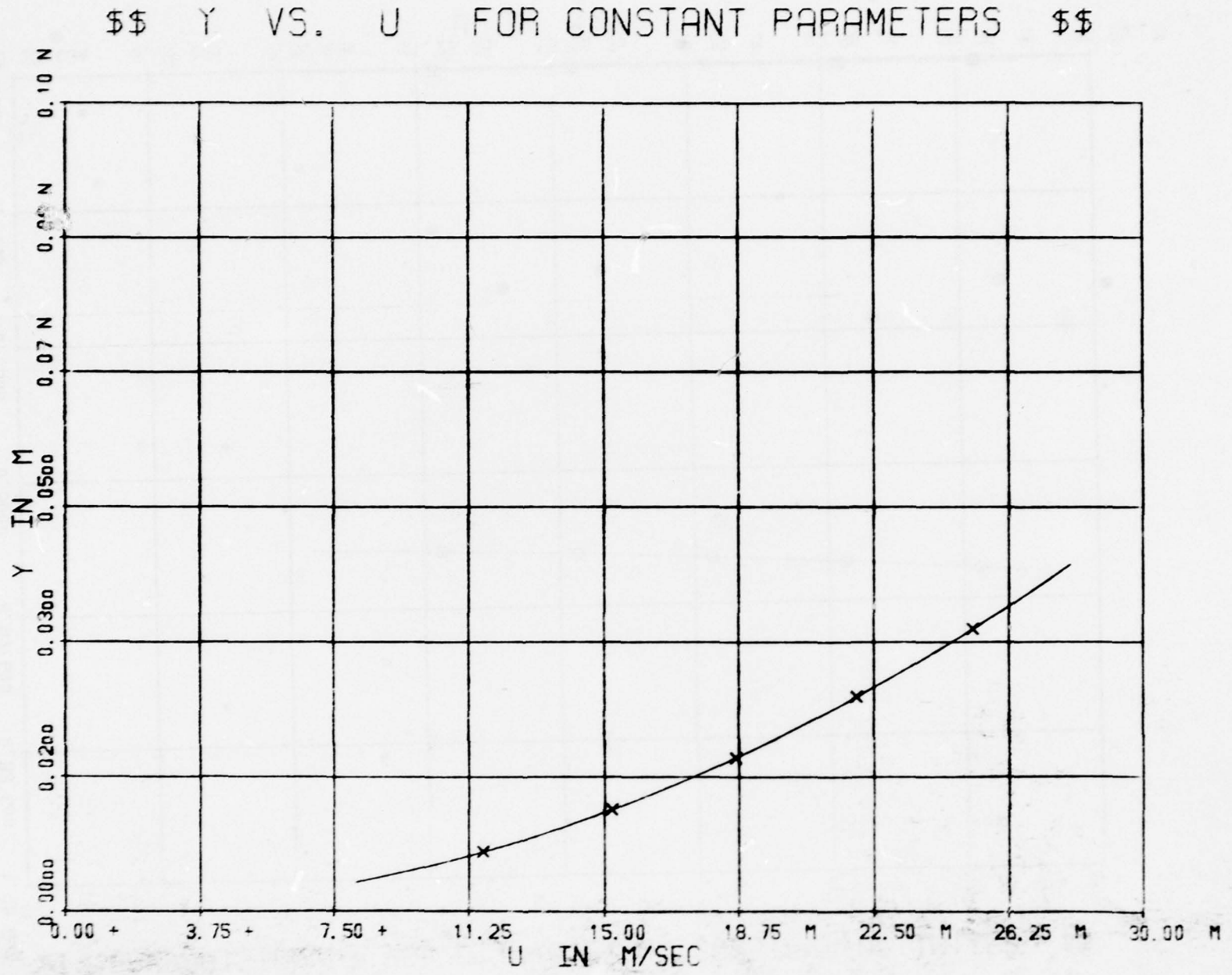

$$ Y VS. U FOR CONSTANT PARAMETERS $$
Y IN M
0.0000
0.0200
0.0300
0.0500
0.07 N
0.10 N
0.00 +
3.75 +
7.50 +
11.25
15.00
18.75 M
22.50 M
26.25 M
30.00 M
U IN M/SEC

Figure 6

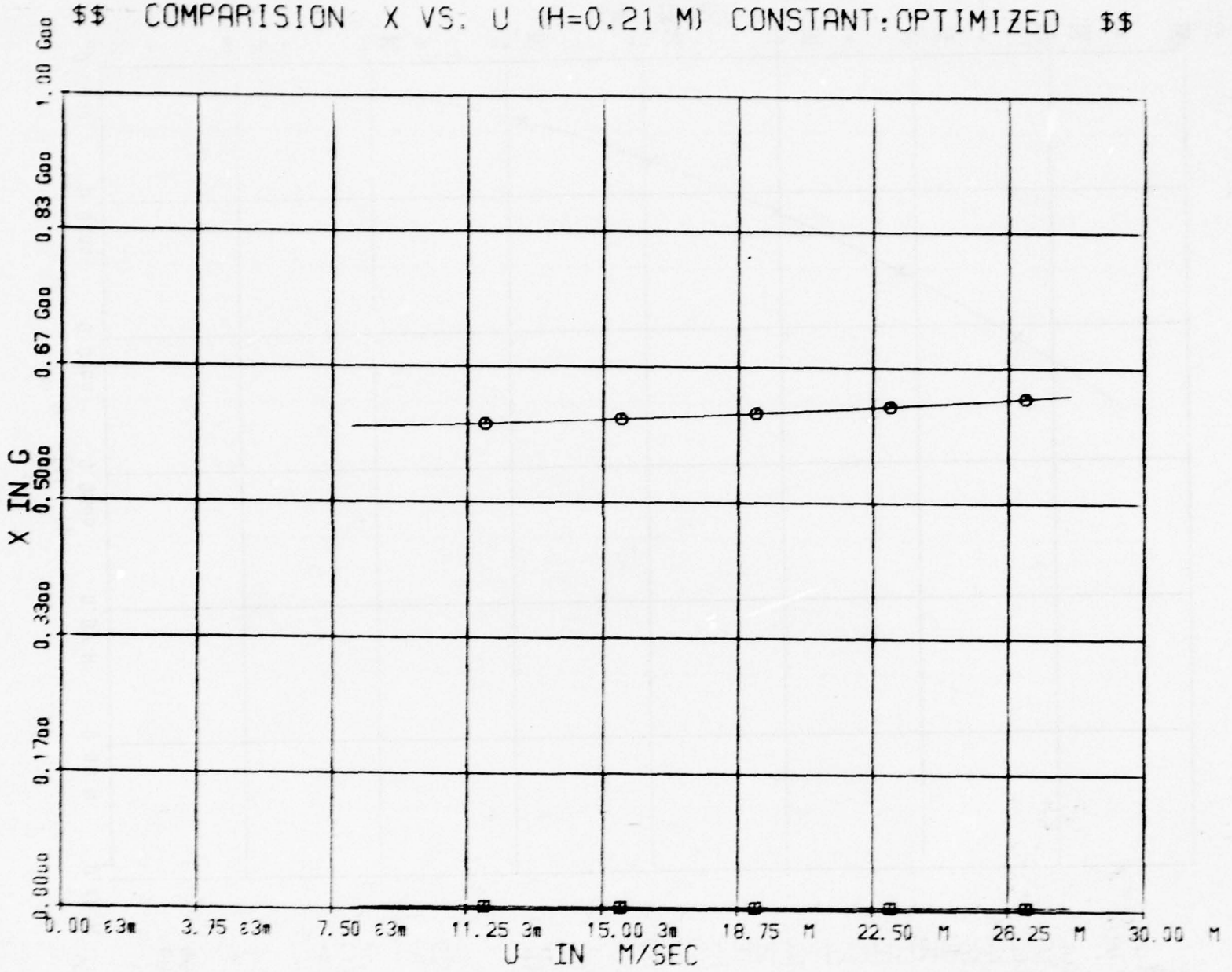

$$ COMPARISION X VS. U (H=0.21 M) CONSTANT:OPTIMIZED $$
X IN G
1.00 Goo
0.83 Goo
0.67 Goo
0.5000
0.3300
0.1700
0.0000
0.00 ε3m
3.75 ε3m
7.50 ε3m
11.25 3m
15.00 3m
18.75 M
22.50 M
26.25 M
30.00 M
U IN M/SEC

Figure 7

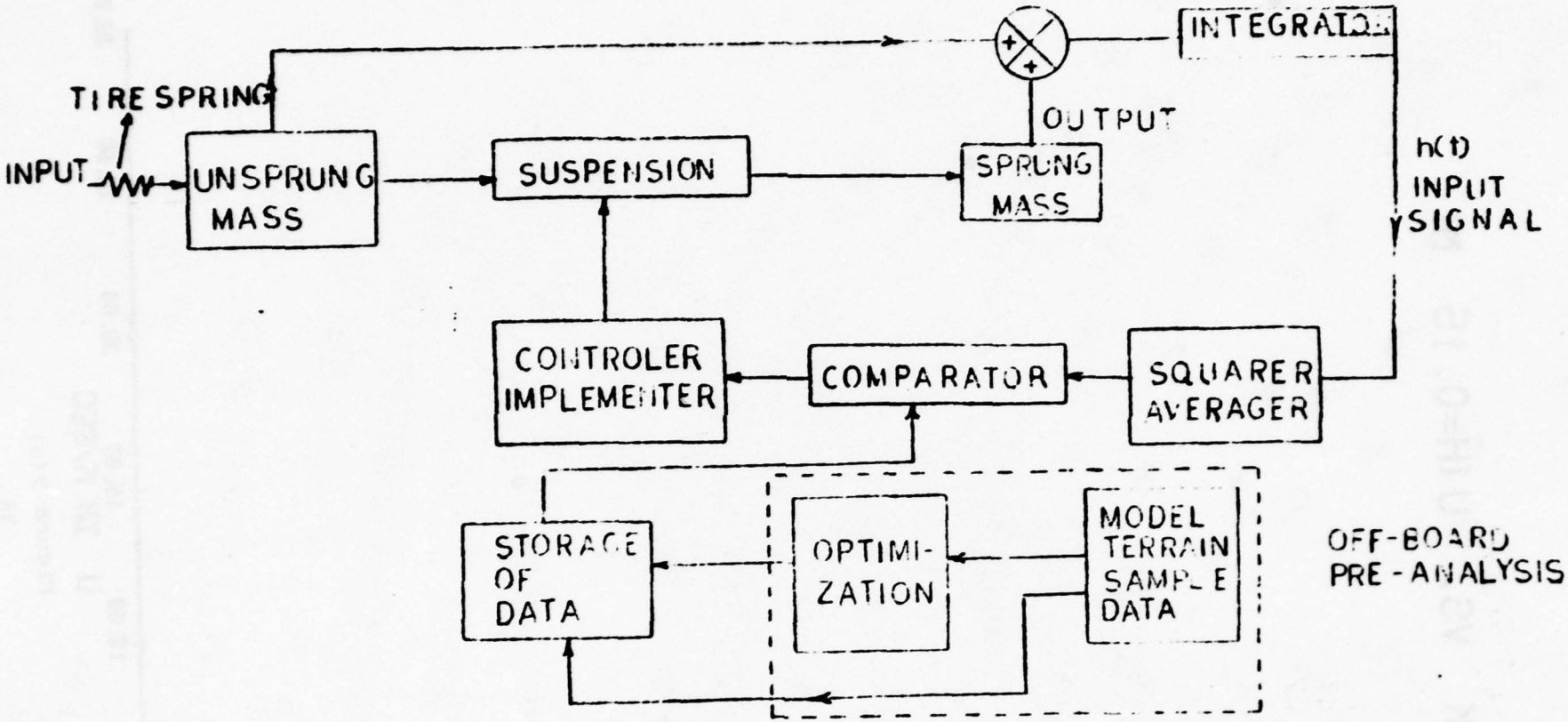


FIG. 8 BLOCK DIAGRAM OF ADAPTIVE CONTROL

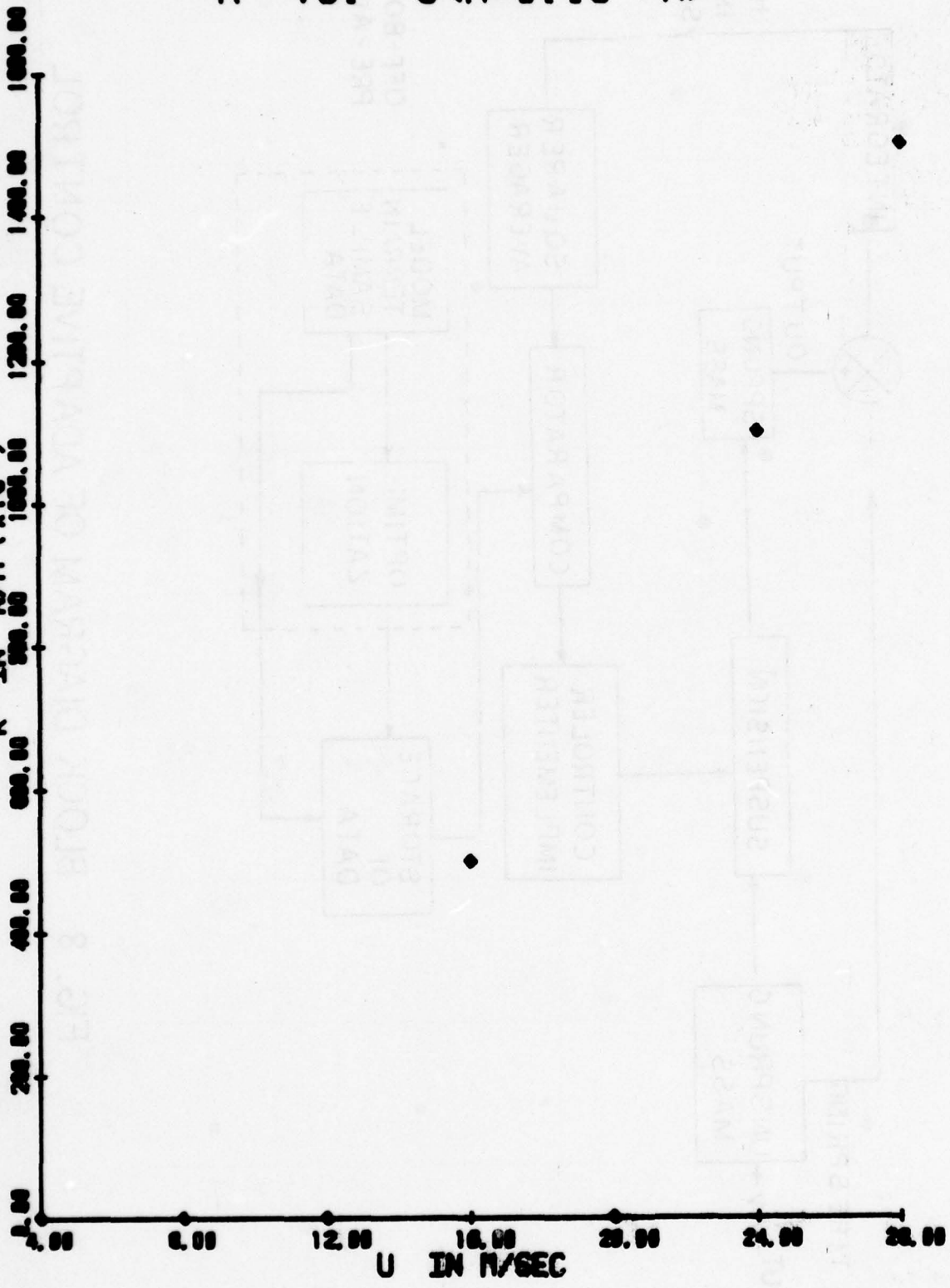

K VS. U (H=0.15 M)
K IN N/M (x10⁻¹)
U IN M/SEC
4.00
8.00
12.00
16.00
20.00
24.00
28.00

Figure 9(a)

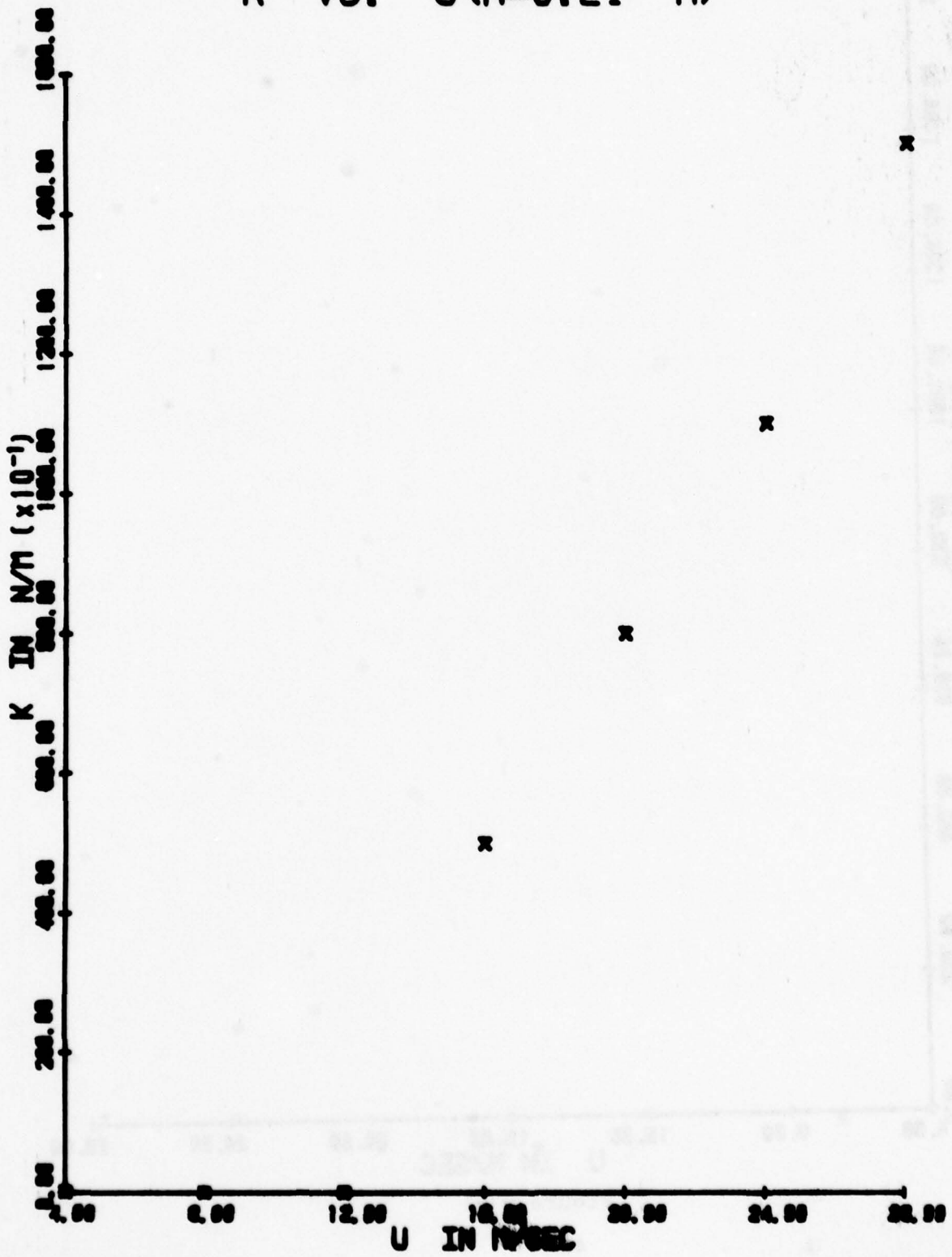


Figure 9(b)

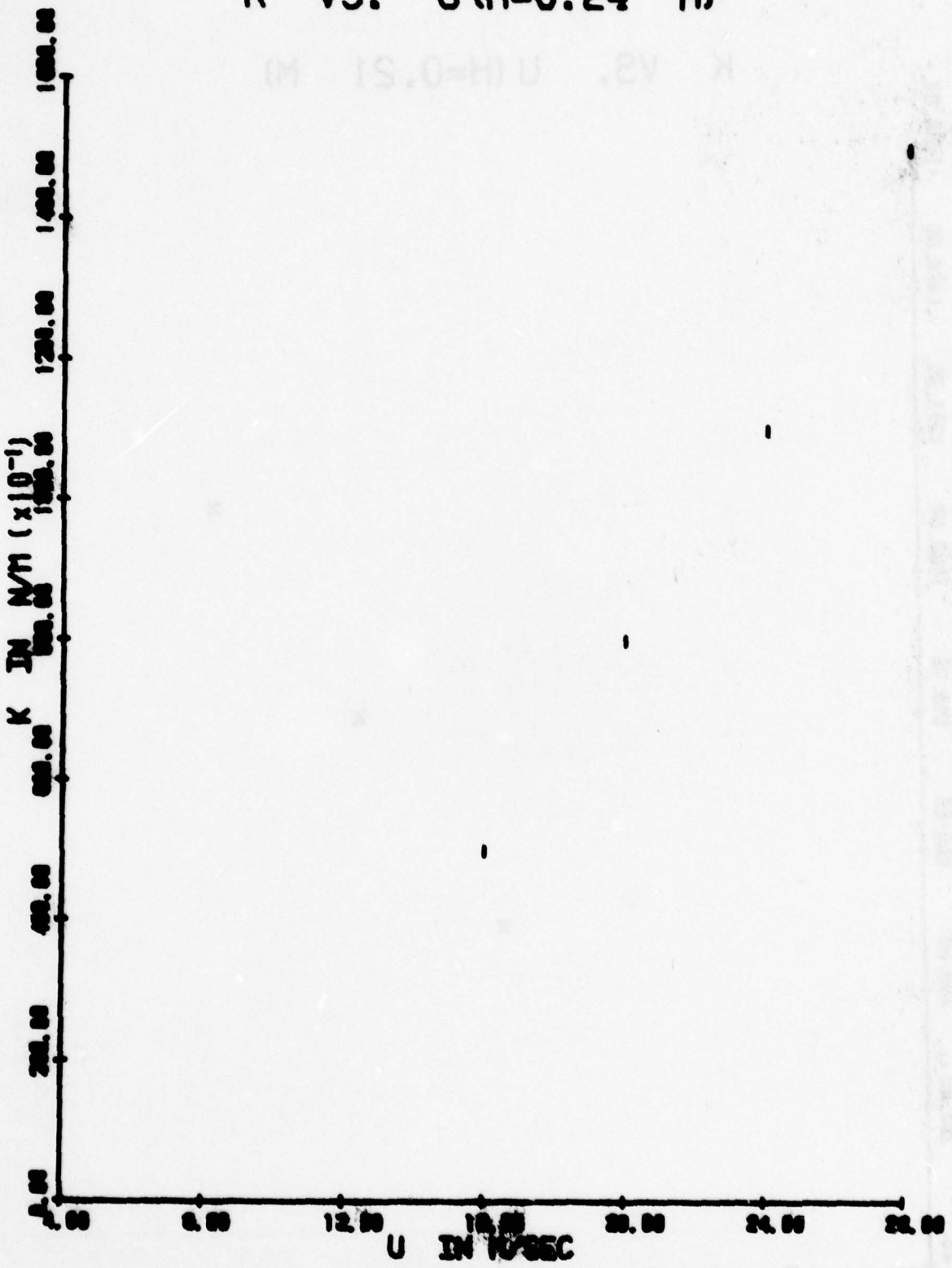


Figure 9(c)

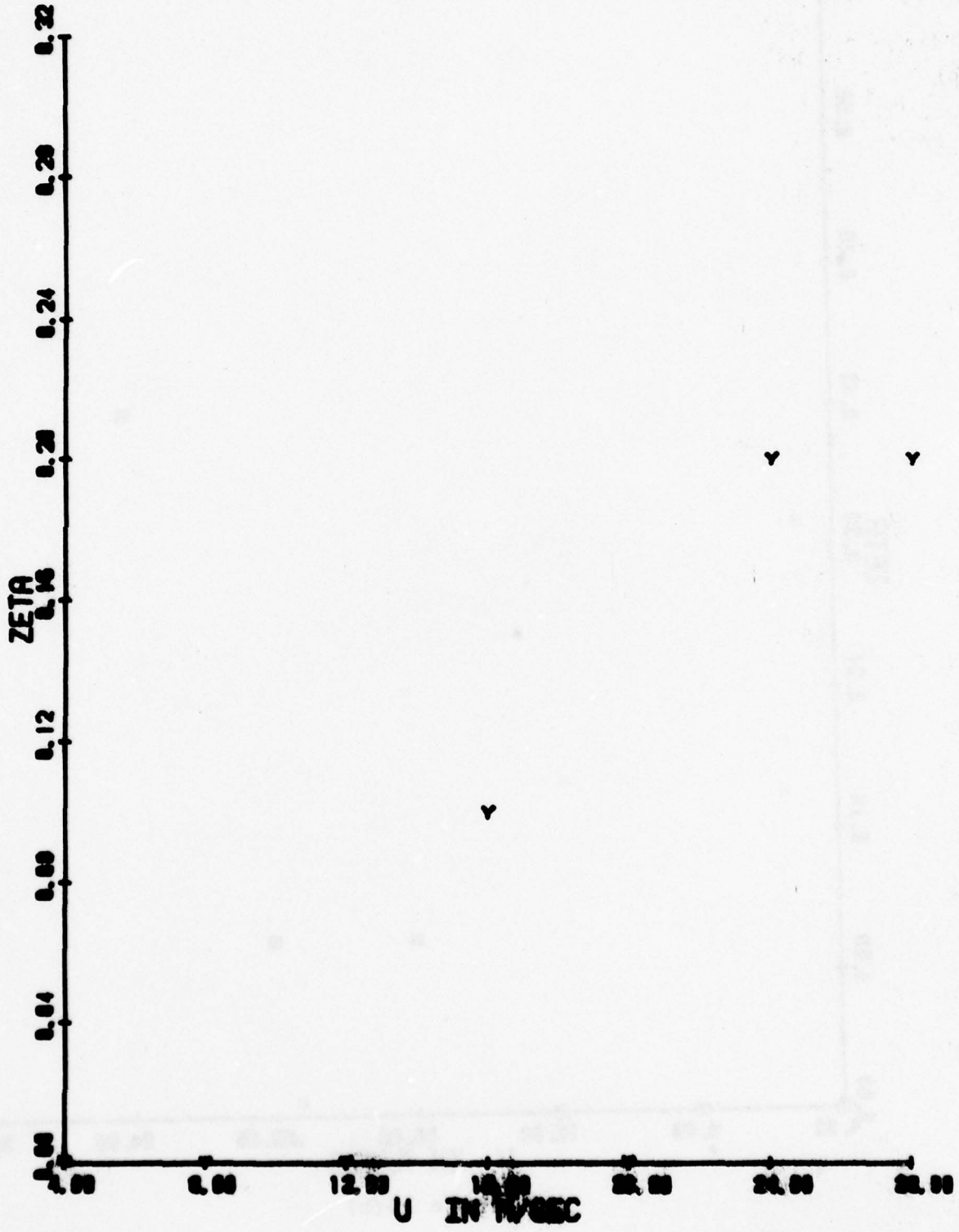

ZETA VS. U(H=0.15 M)
ZETA
0.32
0.20
0.24
0.28
0.16
0.12
0.08
0.04
0.00
4.00
8.00
12.00
16.00
20.00
24.00
28.00
U IN M/SEC

Figure 10(a)

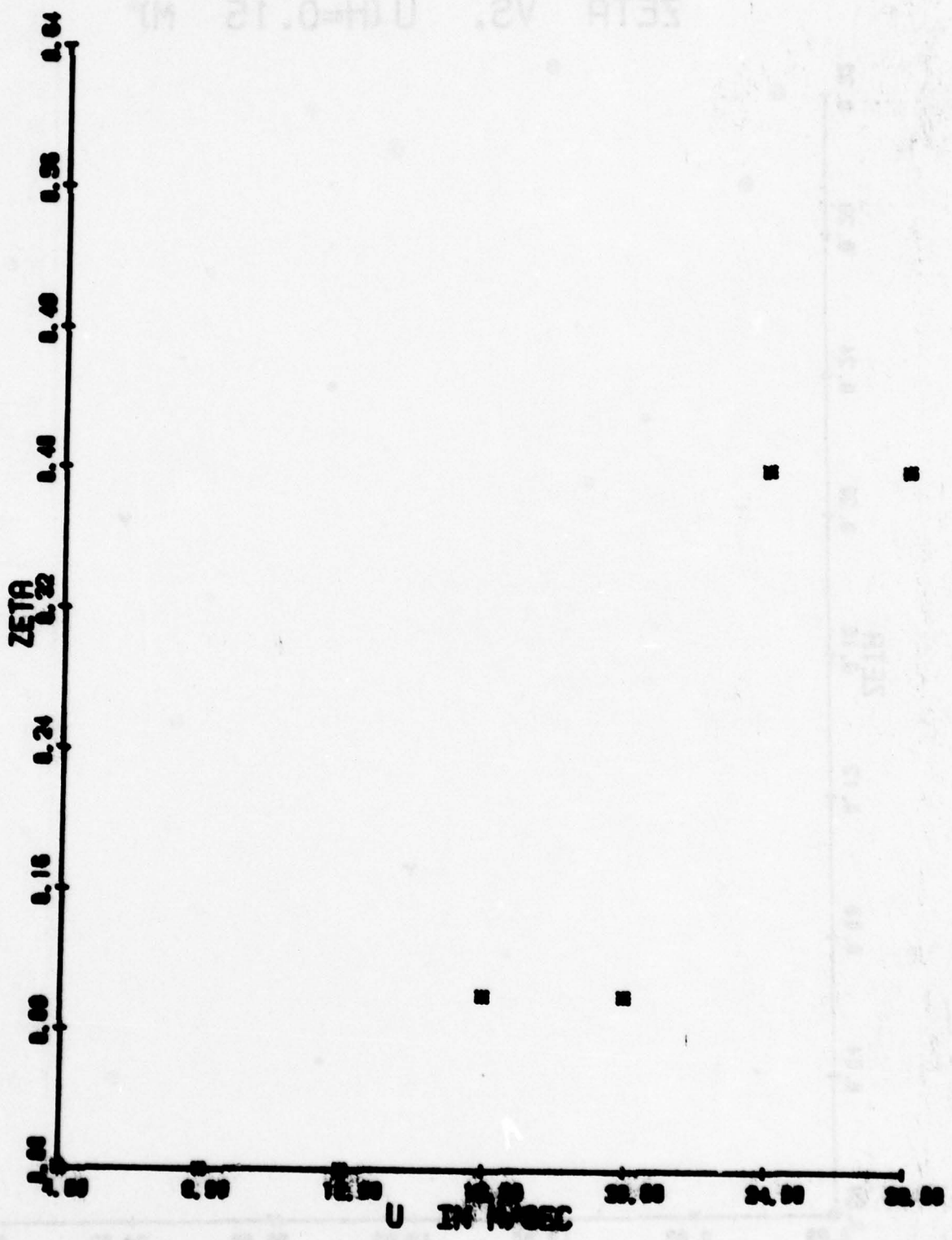


Figure 10(b)

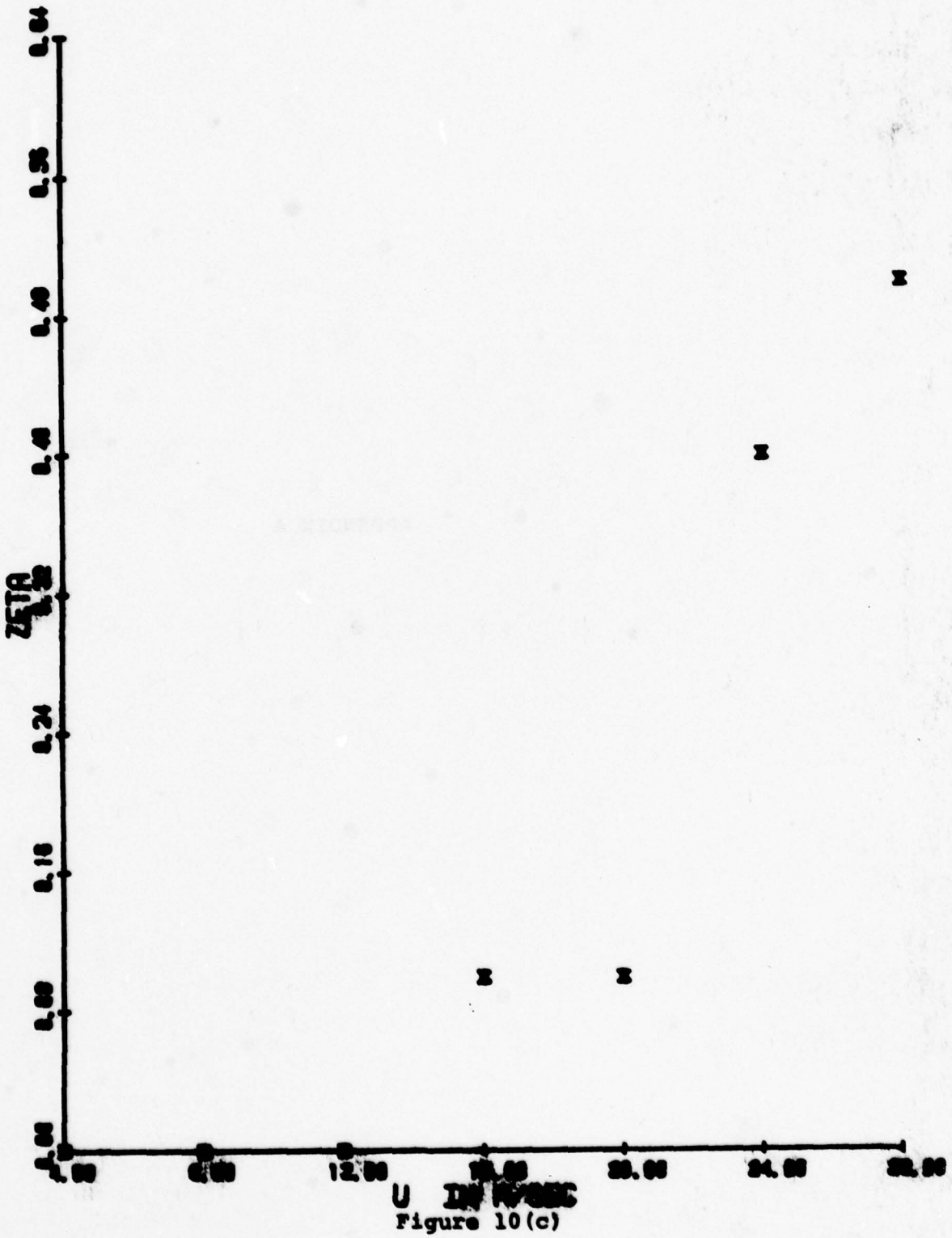


Figure 10(c)

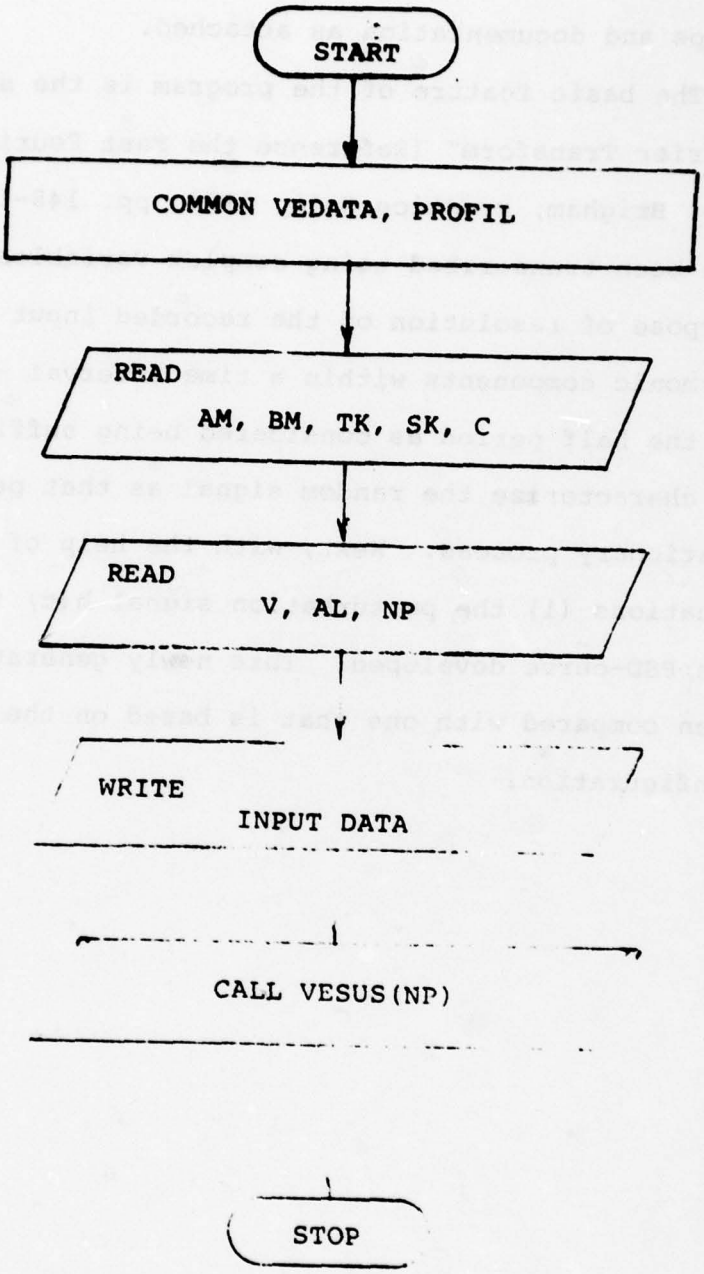
APPENDIX A

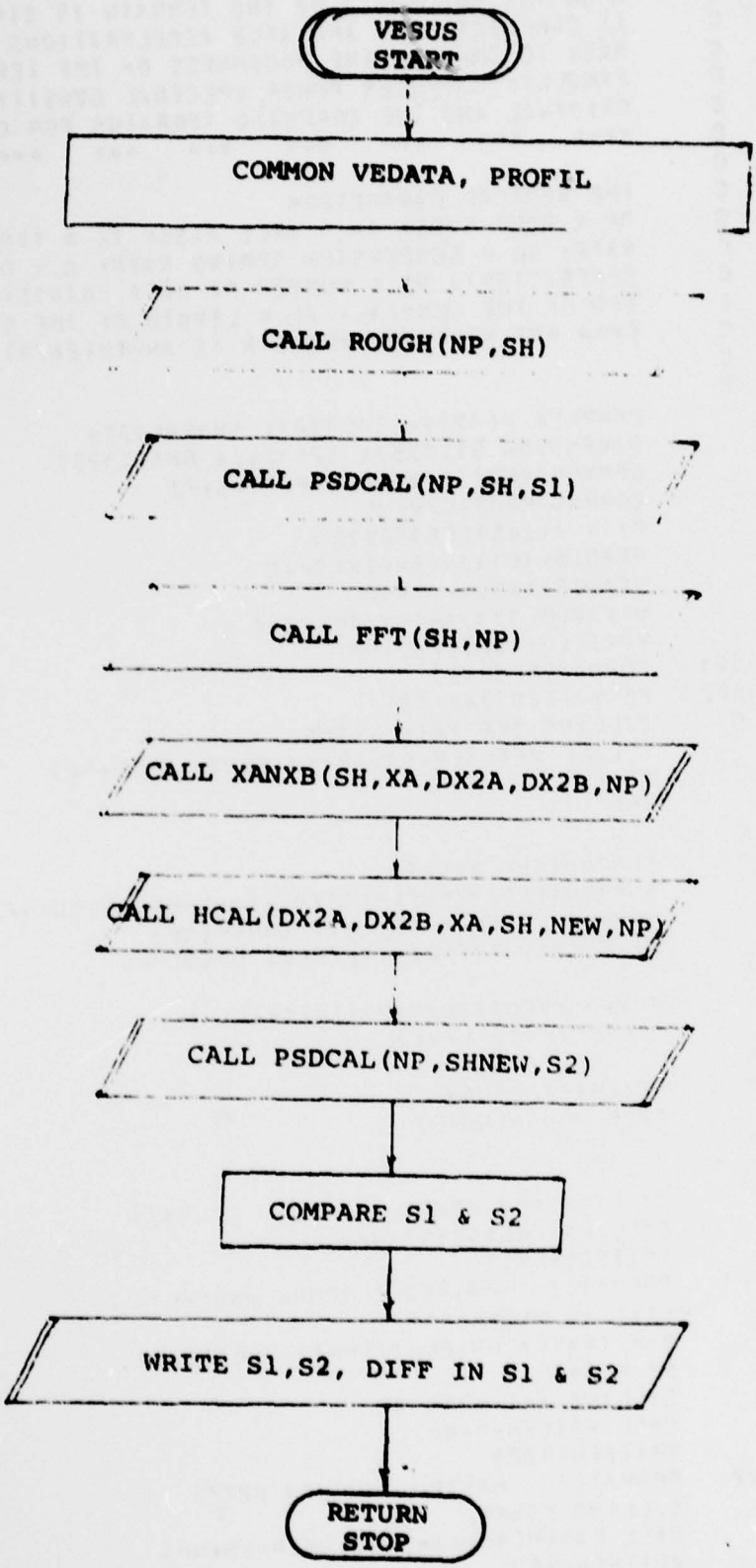
Appendix A

A program has been established containing the program steps and documentation as attached.

The basic feature of the program is the subroutine "Fast Fourier Transform" (Reference the Fast Fourier Transform by E.O. Brigham, Prentice Hall, 1974, pp. 148-171). The program has been transcribed using complex variables. It serves the purpose of resolution of the recorded input signals into its harmonic components within a time interval $-T<t<T$ wherein T is the half period as considered being sufficient in duration to characterize the random signal as that pertaining to a stationary process. Next, with the help of the system equations (1) the perturbation signal h(t) is regenerated and the PSD-curve developed. This newly generated ensemble is then compared with one that is based on the actual terrain configuration.

FLOW DIAGRAM




VESUS START
COMMON VEDATA, PROFIL
CALL ROUGH(NP,SH)
CALL PSDCAL(NP,SH,S1)
CALL FFT(SH,NP)
CALL XANXB(SH,XA,DX2A,DX2B,NP)
CALL HCAL(DX2A,DX2B,XA,SH,NEW,NP)
CALL PSDCAL(NP,SHNEW,S2)
COMPARE S1 & S2
WRITE S1,S2, DIFF IN S1 & S2
RETURN STOP

```
      C     THIS PROGRAM COMPUTES BODY AND AXLE DISPLACEMENTS
      C     WHEN THE ROUGHNESS OF THE TERRAIN IS GIVEN. THEN
      C     IT COMPUTES BODY AND AXLE ACCELERATIONS AND GOES
      C     BACK TO COMPUTE THE ROUGHNESS OF THE TERRAIN. IT
      C     FINALLY, COMPUTES POWER SPECTRAL DENSITIES OF THE
      C     ORIGINAL AND THE COMPUTED TERRAINS FOR COMPARISION
      C     ****   ***   ***   ***   ***   ***   ***   ***   *
      C
      C     THE VEHICAL PARAMETERS
      C     BM = BODY MASS, AM = AXLE MASS, TK = TIRE SPRING
      C     RATE, SK = SUSPENSION SPRING RATE, C = DAMPING
      C     COEFFICIENT, NP = NUMBER OF DATA POINTS, V = SP
      C     EED OF THE VEHICAL, AL = LENGTH OF THE SPAN,
      C     (FOR FFT NP=2**N, WHERE N IS AN INTEGER)
      C
      C
            COMPLEX H(128), SH(128), SHNEW(128)
            DIMENSION S1(128), S2(128), DPSD(128)
            COMMON/VEDATA/AM,BM,TK,SK,C,PI
            COMMON/PROFIL/V,AL
            PI = 3.1459265358979 3
            READ(5,101)AM,BM,TK,SK,C
            READ(5,102)V,AL,NP
            WRITE(6,101)AM,BM,TK,SK,C
            WRITE(6,102)V,AL,NP
      101   FORMAT(5F10.6)
      102   FORMAT(2F10.6,I3)
      C     CALLING THE MAIN VESUS
              CALL VESUS(H,S1,S2,SH,SHNEW,DPSD,NP)
            STOP
            END
      C
      C     SUBROUTINE VESUS
            SUBROUTINE VESUS(H,S1,S2,SH,SHNEW,DPSD,NP)
            COMPLEX H(NP), SH(NP), SHNEW(NP)
            DIMENSION S1(NP), S2(NP), DPSD(NP)
      C
            COMMON/VEDATA/AM,BM,TK,SK,C,PI
            COMMON/PROFIL/V,AL
      C
      C      CALLING ROUGH
            CALL ROUGH(NP,H)
      C
      C
      C     CALLING FOR POWER SPECTRAL DENSITY
            CALL PSDCAL(SH,S1,NP)
            WRITE(6,881)
      881   FORMAT('   PASSED THROUGH PSDCAL')
      C     PRINT IF NECESSARY
            N = IFIX(ALOG(FLOAT(NP))/ALOG(2.))
            NP = 2**N
      C     CALLING FFT PROGRAM
            CALL FFT(SH,N,NP)
            WRITE(6,882)
      882   FORMAT('   PASSED THROUGH FFT')
      C     CALLING XANXB
            CALL XANXB(SH,XA,DX2A,DX2B,XB,NP)
            WRITE(6,883)
```

```
      >     59      803   FORMAT('   PASSED THROUGH XANXB')
      >     60      C     PRINT IF NECESSARY
      >     61      C     CALLING HCAL
      >     62            CALL HCAL(DX2A,DX2B,XA,SHNEW,NP)
      >     63      C     CALLING PSDCAL SHNEW
      >     64            CALL PSDCAL(SHNEW,S2,NP)
      >     65      C     PRINT IF NECESSARY
      >     66            DO 1 J=1,NP
      >     67      1     DPSD(J) = S1(J) - S2(J)
      >     68            WRITE(6,201) (S1(I),S2(I),DPSD(I),I=1,NP)
      >     69      201   FORMAT(' ',8E15.6)
      >     70            RETURN
      >     71            END
      >     72      C
      >     73      C     SUBROUTINE ROUGH STARTS
      >     74            SUBROUTINE ROUGH(N,H)
      >     75            DIMENSION H1(1024)
      >     76            COMPLEX H(N)
      >     77            REAL*4 FMT(1)/'*'/
      >     78            READ(5,FMT)(H1(I),I=1,N)
      >     79            DO 1 I=1,N
      >     80            H1(I) = H1(I)*.3048
      >     81      1     H(I) = CMPLX(H1(I),0.0)
      >     82            WRITE(6,2) (H(I),I=1,12)
      >     83      2     FORMAT(6E15.6)
      >     84            RETURN
      >     85            END
      >     86      C
      >     87      C     SUBROUTINE PSDCAL
      >     88            SUBROUTINE PSDCAL(H,S,NP)
      >     89            COMPLEX H(NP)
      >     90            DIMENSION S(NP)
      >     91            COMMON/VEHICL/BM,RM,IX,SX,C,PI
      >     92            COMMON/PROFIL/V,AL
      >     93            OMEGA = 2.*PI*V/AL
      >     94            FIA = OMEGA*SQRT(RM/SX)
      >     95            DO 1 I=1,NP
      >     96      1     S(I) = H(I)*CONJG(H(I))/(2.*FIA)
      >     97            RETURN
      >     98            END
      >     99      C
      >    100      C     SUBROUTINE FFT
      >    101            SUBROUTINE FFT(A,N,NP)
      >    102            COMPLEX A(NP),U,W,T,CMPLX
      >    103      C     DIVIDING ALL ELEMENTS BY NP
      >    104            DO 1 I=1,NP
      >    105      1     A(I) = A(I)/NP
      >    106      C     REORDERING THE SEQUENCE
      >    107            NBUF = NP/2
      >    108            NBM1 = NP - 1
      >    109            DO 4 L = 1,NBM1
      >    110            IF(I.GE.J)GO TO 2
      >    111            T = A(J)
      >    112            A(J) = A(L)
      >    113            A(L) = T
      >    114      2     K=NBUF
      >    115      3     IF(K.GE.J)GO TO 4
      >    116            J = J - K
      >    117            K = K/2
      >    118            GO TO 3
      >    119      4     J = J+K
```

```
      C     COMPUTATION OF FFT
            PI = 3.1459265358979
            DO 6 M=1,N
            U = (1.0,0.0)
            ME = 2**M
            K = ME/2
            W = CMPLX(COS(PI/K), -SIN(PI/K))
            DO 6 J=1,K
            DO 5 L=J,NR,ME
            LPK = L+K
            T = A(LPK)*U
            A(LPK) = A(L) - T
    5       A(L) = A(L)+T
    6       U = U*W
            RETURN
            END
      C
      C      SUBROUTINE XANXB
            SUBROUTINE XANXB(H,XA,DX2A,DX2B,XB,NP)
            COMPLEX H(NP),XA(NP),XB(NP),DX2A(NP),DX2B(NP)
            COMPLEX A(2,2),CMPLX
            COMMON/VEDATA/AM,BM,TK,SK,C,PI
            COMMON/PROFIL/V,AL
      C
            AMU = TK/SK
            AMU1 = AM/BM
            OMEGA = 2.*PI*V/AL
            SRI = SQRT(BM/SM)
            ETA = OMEGA*SRI
            BETA = C/(2.*SRI)
            DO 1 I=1,NP
            AIM = 2.*BETA*ETA*I
            ARE = (ETA*I)**2
            A(1,1) = CMPLX(1.-ARE,AIM)
            A(1,2) = CMPLX(-1.,-AIM)
            A(2,2) = CMPLX(1.+AMU-AMU1*ARE,AIM)
            DELTA = A(1,1)*A(2,2) - A(1,2)*A(1,2)
            XA(I) = A(1,1)*AMU*H(I)/DELTA
            XB(I) = -A(1,2)*AMU*H(I)/DELTA
            ETA2 =ETA*ETA
            ETA4 = ETA2*ETA2
            DX2A(I) = ETA4*XA(I)
            DX2B(I) = ETA4*XB(I)
    1       CONTINUE
            RETURN
            END
      C
      C      SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
            SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
            COMPLEX H(NP),XA(NP),DX2A(NP),DX2B(NP)
            COMMON/VEDATA/AM,BM,TK,SK,C,PI
            C1 = AM/TK
            C2 = BM/TK
            DO 1 I = 1,NP
    1       H(I) = XA(I) + C1*DX2A(I) + C2*DX2B(I)
            RETURN
            END
END OF FILE
```

Appendix 1

```
C
C       THE VEHICLE PARAMETERS
C       BM = BODY MASS, AM = AXLE MASS, TK = TIRE SPRING
C       RATE, SK = SUSPENSION SPRING RATE, C = DAMPING
C       COEFFICIENT, NP = NUMBER OF DATA POINTS, V = SP
C       EED OF THE VEHICAL, AL = LENGTH OF THE SPAN,
C       (FOR FFT NP=2**N, WHERE N IS AN INTEGER)
C
0001            COMPLEX H(128), SH(128), SHNEW(128)
0002            DIMENSION XA(128), XB(128), DX2A(128), DX2B(128)
0003            DIMENSION HH(126), F(128), PHASEF(128), S1(128)
0004            DIMENSION S2(128), DPSD(128), Y(4), DERY(4)
0005            INTEGER RUNGE
0006            COMMON/FFF/SH,NP
0007            COMMON/XAB/XA,XB
0008            COMMON/VEDATA/AM,BM,TK,SK,C,PI
0009            COMMON/PROFIL/V,AL
0010            PI = 3.1459265358979
C         READING THE DATA RELATED TO VEHICLE PARAMETERS
0011            READ(5,203)AM,BM,TK,SK,C
C         READING THE LENGTH OF THE SPAN AL, AND THE VELOCITY V
0012            READ(5,204)V,AL,NP
0013      203   FORMAT(5F15.4)
0014      204   FORMAT(2F15.0,I3)
0015            WRITE(6,101)AM,BM,TK,SK,C
0016            WRITE(6,102)V,AL,NP
0017      101   FORMAT(1H1//5X,'$$$  VEHICLE SUSPENSION DESIGN  $$$'/
                5X,'*****************************'//
                5X,' INPUT DATA FOR THE SYSTEM :'/
                5X,'  UNSPRUNG MASS             = ',F10.3,'  KG'/
                5X,'  SPRUNG MASS               = ',F10.3,'  KG'/
                5X,'  TIRE SPRING RATE          = ',F10.3,'  N/M'/
                5X,'  SUSPENSION SPRING RATE    = ',F10.3,'  N/M'/
                5X,'DAMPING COEFFICIENT         = ',F10.3,' '/
                5X,'  ----------------------------------'///)
0018      102   FORMAT(5X,'  INPUT DATA:'/
                5X,'  SPEED OF THE VEHICLE = ',F4.2,'  M/SEC'/
                5X,'  TERRAIN WAVE LENGTH: L = ',F7.2,' M '/
                5X,'  NUMBER OF OBSERVATION POINTS = ',I5//
                5X,'  ----------------------------------'///)
C         SUBROUTINE ROUGH TO HANDLE THE TERRAIN DATA READING
C         IN FREE FORMAT
0019            CALL ROUGH(NP,SH)
C         SUBROUTINE PSDCAL CALCULATES THE POWER SPECTRAL DENSITY
0020            CALL PSDCAL(SH,S1,NP)
0021            N = 7
C         A FAST FOURIER TRANSFORM ALGORITHM, WHERE THE TOTAL NO
```

```
C       OF POINTS NP IS ALWAYS AN INTEGRAL POWER N OF 2
0322          CALL FFT(SH,N,NP)
C       SUBROUTINE FUN COMPUTES THE TERRAIN ROUGHNESS VALUE FROM
C       THE COMPLEX FOURIER SERIES
0323          CALL FUN(F)
C       PHASE SHIFT NEEDED, BECAUSE THE VALUES OBTAINED FROM
C       THE COMPLEX FOURIER SERIES ARE ADVANCED IN PHASE BY ONE
C       INCREMENT (BETWEEN TWO OBSERATION POINTS)
0324          DO 14 I=1,128
0325    14    PHASEF(I) = F(I)
0326          DO 15 I=2,128
0327    15    F(I) = PHASEF(I-1)
0328          F(1) = PHASEF(128)
C       4TH ORDER RUNGE-KUTTA INEGRATION STARTS HERE
0329          T =0.0
0330          TMAX = AL/V
0331          DELT = TMAX/128.0
0332          NDIM = 4
0333          DO 2 I=1,NDIM
0334    2     Y(I) = 0.0
C       RUNGE IS A FUNCTION SUBPROGRAM FOR INTEGRATION
0335    3     K = RUNGE(4,Y,DERY,T,DELT)
0336          IF(K.NE.1)GO TO 5
0337          DERY(1) = Y(3)
0338          DERY(2) = Y(4)
0339          CCC = C*(Y(4) - Y(3)) + SK*(Y(2) - Y(1))
C       THIS PROCEDURE TO OBTAIN AN INTEGER TO BE USED
C       AS THE SUBSCRIPT OF THE DISCRETE DATA ARRAY WAS NECESSARY
C       BECAUSE OF HARDWARE LIMITATIONS OF THE COMPUTER
0340          X = T/DELT
0341          II = IFIX(X)
0342          REST = X-II
0343          IF(REST.GT.0.75)II=II+1
0344          FF = F(II)
0345          XNEW = FLOAT(II)
0346          IF(ABS(X-XNEW).LT.0.1)GO TO 7
0347          FF = (F(II+1)+F(II))/2.0
0348          GO TO 8
0349    7     XA(II) = Y(1)
0350          XB(II) = Y(2)
0351          DXDA(II) = (1./AM)*(CCC-TK*(Y(1)-FF))
0352          DXDB(II) = (1.0/BM)*(-CCC)
0353    8     DERY(3)=(1.0/AM)*(CCC-TK*(Y(1)-FF))
0354          DERY(4) = (1.0/BM)*(-CCC)
0355    5     CONTINUE
              IF (T.LT.TMAX)GO TO 3
C       SUBROUTINE HCAL COMPUTES THE TERRAIN ROUGHNESS ENCO
C       BODY AND AXLE ACCLERATIONS , AND THE AXLE  DISPLACEMENT
C       ARE KNOWN
0357          CALL HCAL(DX2A,DX2B,XA,HH,NP)
0358          DO 10 I=1,NP
0359    10    SHNEW(I) = CMPLX(HH(I),0.0)
C       SUB OUTINE PSDCAL IS CALLED TO COMPUTE THE POWER
C       DENSITY OF  THE CALCULATED TERRAIN ROUGHNESS DATA
0360          CALL PSDCAL(SHNEW,S2,NP)
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)              MAIN              01-31-78

0061                  WRITE(6,201)
0062                  DO 1 J=1,NP
0063                  DPSD(J) = (S1(J) - S2(J))*100.0/S1(J)
0064            1     WRITE(6,202)S1(J),S2(J),DPSD(J)
0065          201     FORMAT(1H1///25X,'POWER SPECTRAL DENSITY COMPUTATIONS'///
                     @16X,'TERRAIN DATA',10X,'COMPUTED DATA',10X,'% DIFFERENCE'//
                     @)
0066          202     FORMAT(5X,3E22.5/)
0067                  STOP
0068                  END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = MAIN     , LINECNT =       57
 *STATISTICS*     SOURCE STATEMENTS =       68,PROGRAM SIZE =      8652
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)          MAIN          02-06-73

            C
            C     ***********SUBROUTINE FUN STARTS HERE***********
            C      THIS SUBROUTINE IS USED TO COMPUTE THE COMPLEX
            C      SERIES SUM
0001                SUBROUTINE FUN(F)
0002                DIMENSION F(128)
0003                COMPLEX A(128),AF(128),CMPLX
0004                 COMMON/FFF/A,NP
0005                COMMON/VEDATA/AM,BM,TK,SK,C,PI
0006                COMMON/PROFIL/V,AL
0007                 DO 5 I=1,NP
0008                F(I) = 0.0
0009                ANP = FLOAT(NP)
0010                DO 5 J=1,NP
0011                NI = I*J
0012                JJ = NI/NP
0013                ANI = FLOAT(NI)
            C        A LITTLE TRICK TO AVOID HAVING A LARGE ARGUMENT
            C        FOR THE SINE AND COSINE FUNCTIONS
0014                TH = 2.0*PI*((ANI/ANP)-JJ)
0015                SS = SIN(TH)
0016                CC = COS(TH)
0017                AF(I) = AF(I) + A(J)*CMPLX(CC,SS)
0018          5      CONTINUE
0019                ARF=REAL(AF(I))
0020                AIF = AIMAG(AF(I))
            C        SIGN OF THE RESULTANT FUNCTION VALUE DEPENDS ON THE
            C        SIGN OF THE IMAGINARY PART
            C        SAME FOR THE FIRST HALF
            C        OPPOSITE FOR THE SECOND HALF
0021                RATI =ARF/AIF
0022                IF(I.LE.64)GO TO 2
0023                AIF = -AIF
0024          2     F(I) = AIF*SQRT(1.0+RATI*RATI)
0025            4  CONTINUE
0026                RETURN
0027                END
*OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*   NAME = FUN      , LINECNT =      57
*STATISTICS*     SOURCE STATEMENTS =       27,PROGRAM SIZE =     1900
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
                C
                C       ************SUBROUTINE ROUGH STARTS HERE***********
                C       THIS IS A SUBROUTINE WRITTEN FOR FREE FORMAT READING
                C       OF THE TERRAIN DATA WHICH USUALLY BE GIVEN IN FEET, BUT
                C       IF THE DATA IS GIVEN IN METERS THEN SEE BELOW FOR A NECESSARY
                C       CHANGE
0001                    SUBROUTINE ROUGH(N,H)
0002                   DIMENSION H1(1024)
0003                   COMPLEX H(N)
0004                   REAL*4 FMT(1)/'*'/
0005                   READ(5,FMT)(H1(I),I=1,N)
0006                   S=0.0
0007                   DO 2 I=1,N
0008            2      S=S+H1(I)
0009                   HAV = S/FLOAT(N)
0010                   DO 1 I=1,N
0011                   H1(I) = H1(I)-HAV
                C       FOR DATA IN METERS FOLLOWING ONE CARD CONTAINING THE
                C       MULTIPLYING FACTOR SHOULD BE REMOVED
0012                   H1(I) = H1(I)*0.3048
0013            1      H(I) = CMPLX(H1(I),0.0)
                C      WRITE(6,101) (H(I),I=1,N)
0014            101    FORMAT(1H1///'  TERRAIN ROUGHNESS INPUT :'/
                      @2X,'*********************************'///
                      @2X,'HEIGHTS W.R.T. A REFRENCE'///
                      @(2X,4F12.5/))
0015                   RETURN
0016                   END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = ROUGH    , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =       16,PROGRAM SIZE =     4562
 *STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)          MAIN          02-06-78
C
C      **********SUBROUTINE PSDCAL STARTS HERE**********
C      THE FORMULA USED HERE CAN BE FOUND IN ANY TEXT ON
C      ON RANDOM VIBRETIONS
0001          SUBROUTINE PSDCAL(H,S,NP)
0002          COMPLEX H(NP)
0003          DIMENSION S(NP)
0004          COMMON/VEDATA/AM,RM,TK,SK,C,PI
0005          COMMON/PROFIL/V,AL
0006          OMEGA = 2.*PI*V/AL
0007          ETA = OMEGA*SQRT(RM/SK)
0008          DO 1 I=1,NP
0009        1 S(I) = H(I)*CONJG(H(I))/(2.*ETA)
0010          RETURN
0011          END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = PSDCAL   , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =       11,PROGRAM SIZE =      648
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
C
C       ********SUBROUTINE FFT STARTS HERE**********
0001          SUBROUTINE FFT(A,N,NB)
0002          COMPLEX A(NB),U,W,T,CMPLX
C         DIVIDING ALL ELEMENTS BY NB
0003          DO 1 J=1,NB
0004        1 A(J) = A(J)/NB
C         REORDERING THE SEQUENCE
0005          NBD2 = NB/2
0006          NBM1 = NB - 1
0007           J = 1
0008          DO 4 L = 1,NBM1
0009          IF(L.GE.J)GO TO 2
0010          T = A(J)
0011          A(J) = A(L)
0012          A(L) = T
0013        2 K=NBD2
C         WRITE(6,101)(A(I),I=1,6)
0014        101 FORMAT(3F15.6)
0015        3 IF(K.GE.J)GO TO 4
0016          J = J - K
0017          K = K/2
0018          GO TO 3
0019        4 J = J+K
C         COMPUTATION OF FFT
0020          PI = 3.14592653589793
0021          DO 6 M=1,N
0022          U = (1.0,0.0)
0023          ME = 2**M
0024          K = ME/2
0025          W = CMPLX(COS(PI/K), -SIN(PI/K))
0026          DO 6 J=1,K
0027          DO 5 L=J,NB,ME
0028          LPK = L+K
0029          T = A(LPK)*U
0030          A(LPK) = A(L) - T
C         WRITE(6,102)A(LPK)
0031        102 FORMAT(2F13.5)
0032        5 A(L) = A(L)+T
0033        6 U = U*W
0034          RETURN
0035          END
* OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
* OPTIONS IN EFFECT*   NAME = FFT      , LINECNT =      57
* STATISTICS*     SOURCE STATEMENTS =       35,PROGRAM SIZE =     130
* STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)          MAIN          02-04-73

      C
      C          **********SUBROUTINE HCAL STARTS HERE**********
      C        IF WE ADD THE TWO EQUATIONS OF MOTION FOR BODY AND AXLE
      C        THEN WE GET REQUIRED RELATION TO OBTAIN TERRAIN ROUGHNESS
0001                SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
0002                COMPLEX H(NP),XA(NP),DX2A(NP),DX2B(NP)
0003                COMMON/VEDATA/AM,BM,TK,SK,C,PI
0004                C1 = AM/TK
0005                C2 = BM/TK
0006                DO 1 I = 1,NP
0007           1    H(I) = XA(I) + C1*DX2A(I) + C2*DX2B(I)
0008                RETURN
0009                END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = HCAL      , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =          9,PROGRAM SIZE =      572
 *STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)          MAIN          02-06-78

            C       ********** SUBPROGRAM FUNCTION RUNGE STARTS HERE*******
            C       FUNCTION SUBPROGRAM FOR 4TH ORDER RUNGA-KUTTA METHOD
 0001               FUNCTION RUNGE(N,Y,F,X,H)
 0002               INTEGER RUNGE
 0003               DIMENSION PHI(50),SAVEY(50),Y(N),F(N)
 0004               DATA M/0/
 0005               M = M + 1
 0006               GO TO (1,2,3,4,5),M
 0007          1    RUNGE=1
 0008               RETURN
 0009          2    DO 22 J=1,N
 0010               SAVEY(J) = Y(J)
 0011               PHI(J) = F(J)
 0012          22   Y(J) = SAVEY(J) + .5*H*F(J)
 0013               X = X + .5*H
 0014               RUNGE =  1
 0015               RETURN
 0016          3    DO 33 J=1,N
 0017               PHI(J) = PHI(J) + 2.*F(J)
 0018          33   Y(J) = SAVEY(J)+.5*H*F(J)
 0019               RUNGE=1
 0020               RETURN
 0021          4    DO 44 J=1,N
 0022               PHI(J) = PHI(J)+2.*F(J)
 0023          44   Y(J) = SAVEY(J) + H*F(J)
 0024               X = X + 0.5*H
 0025               RUNGE = 1
 0026               RETURN
 0027          5    DO 55 J=1,N
 0028          55   Y(J) = SAVEY(J)+(PHI(J)+F(J))*H/6.
 0029               M = 0
 0030               RUNGE = 0
 0031               RETURN
 0032               END
  OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
  OPTIONS IN EFFECT*   NAME = RUNGE    , LINECNT =      57
  STATISTICS*     SOURCE STATEMENTS =      32,PROGRAM SIZE =     1...
  STATISTICS*  NO DIAGNOSTICS GENERATED

** NO ERRORS FLAGGED IN THE ABOVE COMPILATIONS.
```

APPENDIX B

Appendix B

Equations for optimal control based on the condition that relative suspension mean deflection is 1/3 of allowable axle clearance H and absolute acceleration transmissibility be a minimum:

for n^{th} harmonic $(j=\sqrt{-1})$

$$m_b\ddot{x}_b+c(\dot{x}_b-\dot{x}_a)+k(x_b-x_a)=0 \qquad a)$$

$$m_a\ddot{x}_a-c(\dot{x}_b-\dot{x}_a)-k(x_b-x_a)+Kx_a=Kh_n \exp(j\omega nt) \qquad b)$$

$$x_b-x_a=y \qquad c)$$

Multiply a) by m_a, b) by m_b and subtract b) from a) and divide by m_am_b ;

$$\ddot{y}+c\left((m_a+m_b)/m_am_b\right)\dot{y}+k\left(\frac{(m_a+m_b)}{m_am_b}\right)y+y(K/m_a)-x_b(K/m_b)= h_n(K/m_a)\exp(ju\omega t) \qquad d)$$

and from a)

$$\ddot{x}_b+\dot{y}(c/m_b)+y(k/m_b)=0 \qquad e)$$

Let $x_b=x_b \exp j(n\omega t+\phi)$, $y=Y \exp j(n\omega t+\psi)$ f)

Then

$$\begin{bmatrix} (-\omega^2n^2+\frac{k+K}{m_a}+\frac{k}{m_b}+c((m_a+m_b)/m_am_b)j\omega n), & -K/m_a \\ (\frac{c}{m_b}j\omega n+\frac{k}{m_b}) & \omega^2n^2 \end{bmatrix} \begin{Bmatrix} Y\exp j\phi \\ X\exp j\psi \end{Bmatrix} = \begin{Bmatrix} Kh_n/m_a \\ 0 \end{Bmatrix} \qquad g)$$

Determinant $\bar{D}$:

$$\omega^4n^4-\omega^2n^2\left(\frac{k+K}{m_a}+\frac{k}{m_b}+c\frac{m+m_b}{m_am_b}j\omega n\right)+\frac{K}{m_a}\left(\frac{c}{m_a}j\omega n+\frac{k}{m_b}\right)=\bar{D} \qquad h)$$

$$\exp(j\phi)Y = \frac{-kh_n\omega^2 n^2}{\bar{D}m_a} \qquad i)$$

$$\exp(j\psi)X = \frac{Kh_n(cj\omega n/m_a + k/m_b)}{\bar{D}\, m_a} \qquad j)$$

After Rationalizing i), j)

$$\left(\frac{Y_n}{h_n}\right)^2 = \frac{\eta_n^4}{\left[\eta_n^4\,\omega_b^2/\Omega_a^2 - \eta_n^2(1+\frac{k}{K}+\frac{\omega_b^2}{\Omega_a^2})+1\right]^2 + 4\zeta^2\eta_n^2\left[\eta_n^2(\frac{k}{K}+\frac{\omega_b^2}{\Omega_a^2})-1\right]^2} \qquad k)$$

$$\left(\frac{X_n}{h_n}\right)^2 = \frac{1+4\zeta^2\eta_n^2}{\left[\eta_n^4\frac{\omega_b^2}{\Omega_a^2} - \eta_n^2(1+\frac{k}{K}+\frac{\omega_b^2}{\Omega_a^2})+1\right]^2 + 4\zeta^2\eta_n^2\left[\eta_n^2(\frac{k}{K}+\frac{\omega_b^2}{\Omega_a^2})-1\right]^2} \qquad l)$$

Where in k), l) $\eta = \omega n/\omega_b$, $k/m_b = \omega_b^2$, $K/m_a = \Omega_a^2$,

$\zeta = c/c_c$, $c_c = 2(k\, m_b)^{1/2}$;

Let it be required that

$\alpha\, \bar{y} \leq H$ ot $\bar{y} = H/\alpha = H/3$ $(\alpha=3)$ m)

$$\bar{y}^2 = \frac{H^2}{9} = \frac{1}{2m}\sum_1^m h_n^2\,(X/h_n)^2 = \frac{1}{2m}\sum_1^m Y^2 \qquad n)$$

$\bar{X}^2 = (1+4\zeta^2\bar{y}^2)\,\bar{\eta}^4$ $\bar{\eta}^2$ = mean square value of η_n^2 o)

$\bar{\eta}^2 = \bar{\omega}^2/\omega_b^2 = \frac{4\pi\bar{u}^2}{\ell^2\omega_b^2}$ $\bar{u}^2$ = mean square value of speed u p)

The program for n) and o) follows.

```
      C     THE VEHICLE PARAMETERS
      C     AM = BODY MASS, AM = AXLE MASS, TK = TIRE SPRING
      C     RATE, SK = SUSPENSION SPRING RATE, C = DAMPING
      C     COEFFICIENT, NP = NUMBER OF DATA POINTS, V = SP
      C
      C     (FOR FFT N=2**N, WHERE N IS AN INTEGER)
      C
      C
0001          IMPLICIT REAL*8(A-H,O-Z)
0002          COMPLEX*16 H(128), SH(128), SHNEW(128)
0003          INTEGER RUNS
      C
0004          DIMENSION ROOT(2), WA(15)
0005          DIMENSION X(128),A(128),X0(128),DX2A(128),DX2B(128)
0006          DIMENSION Y(4),DERY(4),PRMT(5),AUX(16,4)
0007            DIMENSION SM(128)
0008          DIMENSION SS(128),SSM(128)
0009          DIMENSION F(128),PHASEF(128)
0010          COMMON/
0011          COMMON/X
0012          COMPLEX*16 SSH(129),DIFH(128)
0013          DIMENSION S1(128), S2(129), DPSD(128)
0014          COMMON/VEDATA/AM,BM,TK,SK,C,PI
0015          COMMON/PROFILE/V,AL
0016          COMMON/ROUGH/RUN
0017          INTEGER RUNS
0018          PI = 3.14592653589793
0019          READ(5,203)AM,BM,TK,SK,C
0020          READ(5,
0021    203   FORMAT(5
0022    204   FORMAT(
      C     WRITE(6,
      C     WRITE(6,
0023    101   FORMAT(1
     @5X,'*********************************'//
     @5X,' INPUT DATA FOR THE SYSTEM :'/
     @5X,'  UNSPRUNG MASS              = ',F10.2,'  KG'/
     @5X,'  SPRUNG MASS                = ',F10.2,'  KG'/
     @5X,'  TIRE SPRING RATE           = ',F10.2,'  N/M'/
     @5X,'  SUSPENSION SPRING RATE     = ',F10.2,'  N/M'/
     @5X,'DAMPING COEFFICIENT          = ',F10.2,'  '/
     @5X,'  ---------------------------------'///)
0024    102   FORMAT(5X,'  INPUT DATA:'/
     @5X,'  SPEED OF THE VEHICLE = ',F6.2,'  M/SEC'/
     @5X,'  TERRAIN WAVE LENGTH: L = ',F7.2,' M '/
     @5X,'  NUMBER OF OBSERVATION POINTS = ',I5//
     @5X,'  ---------------------------------'///)
0025          U = 20.0
      C
0026          CALL ROUGH(NP,SH)
0027          DO 12 I=1,128
0028    12    SSH(I) = SH(I)
      C     DO 500 IU=1,10
0029          N = 7
0030          CALL FFT(SH,N,NP)
```

```
      C
0031          CALL OPTIM(AM,BM,TK,PI,AL,U,SH,NP)
0032          CALL PSDCAL(SH,SI,NP)
0033          CALL FUN(F)
0034          DO 14 I=1,128
0035    14    PHASEF(I) = F(I)
0036          DO 15 I=2,128
0037    15     F(I) = PHASEF(I-1)
0038          F(1) = -PHASEF(128)
0039          T =0.0
      C
      C
0040          TMAX = AL/V
0041          DELT = TMAX/128.0
0042          NDIM = 4
      C
      C
0043          DO 2 I=1,NDIM
0044    2     Y(I) = 0.0
0045    3     K = RUNGE(4,Y,DERY,T,DELT)
0046          IF(K.NE.1)GO TO 5
0047          DERY(1) = Y(3)
0048          DERY(2) = Y(4)
0049          CCC = C*(Y(4) - Y(3)) + SK*(Y(2) - Y(1))
0050          X = T/DELT
0051          II = IFIX(SNGL(X))
0052          RESD = X-II
0053          IF(RESD.GT.0.75)II=II+1
0054          FF = F(II)
0055          XNEW =II
0056          IF(DABS(X-XNEW).LT.0.1)GO TO 7
0057          FF = (F(II+1)+F(II))/2.0
0058          GO TO 6
0059    7     XA(II) = Y(1)
0060          XB(II) = Y(2)
0061          DX2A(II) = (1./AM)*(CCC-TK*(Y(1)-FF))
0062          DX2B(II) = (1.0/BM)*(-CCC)
0063    6     DERY(3)=(1.0/AM)*(CCC-TK*(Y(1)-FF))
0064          DERY(4) = (1.0/BM)*(-CCC)
      C
      C
0065    5     CONTINUE
0066          IF (T.LT.TMAX)GO TO 3
      C
      C     CONSTRAINT EQUATION FOR OPTIMIZATION
      C
      C
0067    301      FORMAT('ROOTS ARE', 2E17.7)
      C
0068          CALL HCAL(DX2A,DX2B,XA,HH,NP)
0069          DO 18 I=1,128
0070    18    ORIGH(I) = REAL(SSH(I))
0071          DO 10 I=1,NP
0072    10    SHNEW(I) = DCMPLX(HH(I),0.002)
0073          CALL FFT(SHNEW,N,NP)
```

```
IBM 360 TERMINAL SYSTEM FORTRAN G(41336)                MAIN

0074                CALL PSDCAL(SHNEW,S2,NP)
0075                WRITE(6,201)
0076                DO 1 J=1,NP
0077                DPSD(J)=(ORIGH(J)-HH(J))*100.0/ORIGH(J)
0078          1     WRITE(6,202)ORIGH(J),HH(J),DPSD(J)
0079          201   FORMAT(1H1///,'   COMPARISION BETWEEN ORIGINAL & COMPUTED DAT
                   @5X,'TERRAIN DATA',10X,'COMPUTED DATA',10X,'% DIFFERENCE'//
                   @)
0080          202   FORMAT(5X,3E22.5/)
0081                WRITE(6,206)
0082          206   FORMAT('1H1,'      LOGRITHMIC OUTPUT'//
                   @5X,'FREQUENCY',8X,'PSD',14X,'LOG(FRQ)',9X,'LOG(PSD)')
0083                OMEGA = 2.0*PI*V/AL
0084                ETA = OMEGA*DSQRT(BM/SK)
                    C     TAKING LOGRITHMS OF PSD AND FREQUENCY VALUES
0085                DO 16 I=1,NP
0086                PSD=S1(I)*ETA*AL/(2.0*PI)
0087                FRQ = I*2.0*PI/AL
0088                PSDL=DLOG10(PSD)
0089                FRQL = DLOG10(FRQ)
0090                IF((I.EQ.1).OR.(I.GT.65))GO TO 16
0091                WRITE(6,210)FRQ,PSD,FRQL,PSDL
0092          16    CONTINUE
0093          210      FORMAT(5X,4E17.6)
              C
0094          99     CONTINUE
0095                STOP
0096                END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = MAIN     , LINECNT =       57
 *STATISTICS*     SOURCE STATEMENTS =        96,PROGRAM SIZE =     22542
 *STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
C
C     SUBROUTINE OPTIM STARTS HERE
C
      SUBROUTINE OPTIM(AM,BM,TK,PI,AL,U,SH,NP)
      IMPLICIT REAL *8(A-H,O-Z)
C
      DIMENSION ROOT(9), V(9)
      DIMENSION  COE(9), RR(9), RC(9), POL(9)
      COMPLEX *16   CONSH, SH(128)
C
C     WRITE(6,411)U
411   FORMAT(' U =',D17.6)
C     WRITE(6,401)(SH(I),I=125,128)
      SIGMAH = 0.0
C     WRITE(6,402)NP
402   FORMAT('   NP =',I5)
      DO 10 N=1,NP
10     SIGMAH = SIGMAH + CDABS(SH(N))
C
C     WRITE(6,407)SIGMAH
407   FORMAT('SIGMAH =',D17.6)
C
      DO 40 IH = 15, 40, 3
      XSOMIN = 1.0015
      H = IH*0.01
      ANP = NP
      CC = 9.0*SIGMAH/(ANP*H*H)
      WRITE(6,408)CC
408   FORMAT('CC =',D17.6)
      DO 30 IZ = 1,10
      Z = 0.1*IZ
      WRITE(6,421)
421   FORMAT(' ENTERED FIRST DO LOOP')
      DO 30 IK= 1,10
      SK = 1000.0*IK
      OMB = DSQRT(SK/BM)
      OMA = DSQRT(TK/AM)
      OMR = OMB/OMA
      WRITE(6,422)
422   FORMAT('  ENTERED THE SECOND LOOP')
      SKR = SK/TK
      C11 = SKR + OMR*OMR
      C12 = C11 + 1.0
      Z2 = Z*Z
      COE(1) = CC
      COE(2) = 0.0
      COE(3) = 4.*Z2 - 2.*C12
      COE(4) = 0.0
      COE(5) = C12*C12 + 2.*OMR*OMR - CC - 8.*Z2*C11*C11
      COE(6) = 0.0
      COE(7) = 4.*Z2*C11*C11 - 2.*OMR*OMR*C12
      COE(8) = 0.0
      COE(9) = OMR**4
C
C
```

```
                403   FORMAT('COE    ',D17.6)
                C     CALLING A PROGRAM FROM SSP
                C
0044                  CALL DPRBM(COE,9,RR,RC,POL,IR,IER)
                C
                C
0045            404   FORMAT('RRE RC',D17.6)
                C     WRITE(6,405)IR
0046            405   FORMAT('IR =',I5)
0047                  IF(IR.EQ.0)GO TO 30
0048                  I = 0
0049                  J=1
0050            19     IF(RC(J).EQ.0.0D1)GO TO 20
0051                  IF(J.EQ.IR)GO TO 25
0052                  J = J+1
0053                  GO TO 19
0054            20     IF(RR(J).GE.0.0D1)GO TO 22
0055                  IF(J.EQ.IR)GO TO 25
0056                  J = J +1
0057                  GO TO 19
0058            22    I = I + 1
0059                  ROOT(I) = RR(J)
0060                  J = J + 1
0061                  GO TO 19
0062            25     CONTINUE
0063                  IF(I.EQ.0)GO TO 30
                C      WRITE(6,201)((J,ROOT(J)),J=1,I)
0064            201   FORMAT(' ROOT(',I1,') = ',D17.6)
0065                  DO 27 J = 1,I
0066            27     V(J) = 0.5*ROOT(J)*AL*OMB/PI
0067                  DO 29 J1=1,I
0068                  IF(DABS(U-V(J1)).GT.0.2)GO TO 29
0069                  DO 28 J = 1, I
0070                  ROO2 = ROOT(J)*ROOT(J)
0071                  ROO4 = ROO2*ROO2
0072                  XSO = (1.+4.*Z2*ROO2)/(ROO4*CC)
0073                  IF(XSO.GE.XSOMIN)GO TO 26
0074                  XSOMIN = XSO
0075                  SKOPT = SK
0076                  ZOPT = Z
0077            28     CONTINUE
0078                  WRITE(6,202)((J,V(J)),J=1,I)
0079            202   FORMAT('    V(',I1,') =',D17.6)
0080                  WRITE(6,203)SK,Z,H
0081            203   FORMAT('  SK =',D17.6,'  Z =',D17.6,'  H =',F6.2)
0082            29      CONTINUE
0083            30     CONTINUE
0084                  WRITE(6,418)SKOPT,ZOPT
0085            418   FORMAT('SKOPT=',D17.6,'   ZOPT=',D17.6)
0086            40      CONTINUE
0087                  RETURN
0088                  END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = OPTIM    , LINECNT =       57
 *STATISTICS*     SOURCE STATEMENTS =       88,PROGRAM SIZE =     2948
```

```
C      ***********FUN
       SUBROUTINE FUN(F)
       DIMENSION F(128)
       COMPLEX A(128),AF(128),CMPLX
       COMMON/FFF/A,NP
       COMMON/VEDATA/AM,BM,TK,SK,C,FI
       COMMON/PROFILE/L
       DO 6 I=1,NP
       F(I) = 0.0
       ANP = FLOAT(NP)
       DO 5 J=1,NP
       NJ = I*J
       JJ = MOD(NJ,NP)
       ANI = FLOAT(NJ)
       TH = 2.00PI*((ANI/ANP)*JJ)
       SS = SIN(TH)
       CC = COS(TH)
       AF(I) = AF(I) + A(J)*CMPLX(CC,SS)
5      CONTINUE
       SRE=REAL(AF(I))
       AIE = AIMAG(AF(I))
       RATI =SRE/AIE
       IF(I.LE.64)GO TO 2
       AIE = -AIE
2      F(I) = AIE*SQRT(1.0+RATI*RATI)
       CONTINUE
       RETURN
       END
*OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*   NAME = FUN      , LINECNT =      57
*STATISTICS*     SOURCE STATEMENTS =      27,PROGRAM SIZE =     1004
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
                                                        MAIN
C
C       SUBROUTINE ROUGH STARTS
0001          SUBROUTINE ROUGH(N,H)
0002          DIMENSION H1(1024)
0003          COMPLEX H(N)
0004          REAL*4 FMT(1)/'(F)'/
0005          READ(5,FMT)(H1(I),I=1,N)
0006          S=0.0
0007          DO 2 I=1,N
0008        2 S=S+H1(I)
0009          HAV = S/FLOAT(N)
0010          DO 1 I=1,N
0011          H1(I) = H1(I)-HAV
0012          H1(I) = H1(I)*0.3048
0013        1 H(I) = CMPLX(H1(I),0.0)
C       WRITE(6,101) (H(I),I=1,N)
0014      101 FORMAT('1H1///' TERRAIN ROUGHNESS INPUT :'//
     2X,'**********************************'///
     2X,'HEIGHTS W.R.T. A REFRENCE'///
     2(2X,4E12.5/))
0015          RETURN
0016          END
 *OPTIONS IN EFFECT*   ID,EPCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = ROUGH    , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =       16,PROGRAM SIZE =     4552
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
ICHIGAN TERMINAL SYSTEM FORTRAN G(4.8.336)                    MAIN
C
C       SUBROUTINE PSDCAL
0001            SUBROUTINE PSDCAL(H,S,NP)
0002            COMPLEX H(NP)
0003            DIMENSION S(NP)
0004            COMMON/VEDATA/AM,BM,TK,SK,C,PI
0005            COMMON/PROFIL/V,AL
0006            OMEGA = 2.*PI*V/AL
0007            ETA = OMEGA*SQRT(BM/SK)
0008            DO 1 I=1,NP
0009          1 S(I) = H(I)*CONJG(H(I))/(2.*ETA)
0010            RETURN
0011            END
 *OPTIONS IN EFFECT*   ID,ECDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = PSDCAL   , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =       11,PROGRAM SIZE =      548
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(41336)                    MAIN

                C
                C       SUBROUTINE FFT
0001                    SUBROUTINE FFT(A,N,NB)
0002                    COMPLEX A(NB),U,W,T,CMPLX
                C       DIVIDING ALL ELEMENTS BY NB
0003                    DO 1 J=1,NB
0004            1       A(J) = A(J)/NB
                C       REORDERING THE SEQUENCE
0005                    NBD2 = NB/2
0006                    NBM1 = NB - 1
0007                    J = 1
0008                    DO 4 L = 1,NBM1
0009                    IF(L.GE.J)GO TO 2
0010                    T = A(J)
0011                    A(J) = A(L)
0012                    A(L) = T
0013            2       K=NBD2
                C       WRITE(6,101)(A(I),I=1,6)
0014            101     FORMAT(3F15.6)
0015            3       IF(K.GE.J)GO TO 4
0016                    J = J - K
0017                    K = K/2
0018                    GO TO 3
0019            4       J = J+K
                C       COMPUTATION OF FFT
0020                    PI = 3.14592653589793
0021                    DO 6 M=1,N
0022                    LE = (1.E0,0.0)
0023                    ME = 2**M
0024                    K = ME/2
0025                    W = CMPLX(COS(PI/K), -SIN(PI/K))
0026                    DO 5 J=1,K
0027                    DO 5 L=J,NB,ME
0028                    LPK = L+K
0029                    T = A(LPK)*U
0030                    A(LPK) = A(L) - T
                C       WRITE(6,102)A(LPK)
                102     FORMAT(2F13.5)
                5       A(L) = A(L)+T
                6       U = U*W
0031                    RETURN
0032                    END
* OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
* OPTIONS IN EFFECT*   NAME = FFT       , LINECNT =      57
* STATISTICS*          SOURCE STATEMENTS =      35,PROGRAM SIZE =     1304
* STATISTICS*          NO DIAGNOSTICS GENERATED
```

```
      C
      C     SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
0001        SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
0002        COMPLEX H(NP),XA(NP),DX2A(NP),DX2B(NP)
0003        COMMON/VEDATA/AM,RM,TK,SK,C,PI
0004        C1 = AM/TK
0005        C2 = RM/TK
0006        DO 1 I = 1,NP
0007      1 H(I) = XA(I) + C1*DX2A(I) + C2*DX2B(I)
0008        RETURN
0009        END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = HCAL      , LINECNT =      57
 *STATISTICS*  SOURCE STATEMENTS =      9,PROGRAM SIZE =     672
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
MICROSOFT TERMINAL SYSTEM FORTRAN G(41336)                    MAIN

                C       FUNCTION SUBPROGRAM FOR 4TH ORDER RUNGE-KUTTA METHOD
                        FUNCTION RUNGE(N,Y,F,X,H)
                        INTEGER RUNGE
                        DIMENSION PHI(50),SAVEY(50),Y(N),F(N)
                        DATA M/0/
                        M = M + 1
                        GO TO (1,2,3,4,5),M
              1         RUNGE=1
                        RETURN
              2         DO 22 J=1,N
                        SAVEY(J) = Y(J)
                        PHI(J) = F(J)
             22         Y(J) = SAVEY(J) + .5*H*F(J)
                        X = X + .5*H
                        RUNGE = 1
                        RETURN
              3         DO 33 J=1,N
                        PHI(J) = PHI(J) + 2.*F(J)
             33         Y(J) = SAVEY(J)+.5*H*F(J)
                        RUNGE=1
                        RETURN
              4         DO 44 J=1,N
                        PHI(J) = PHI(J)+2.*F(J)
             44         Y(J) = SAVEY(J) + H*F(J)
                        X = X + 0.5*H
                        RUNGE = 1
                        RETURN
              5         DO 55 J=1,N
             55         Y(J) = SAVEY(J)+(PHI(J)+F(J))*H/6.
                        M = 0
                        RUNGE = 0
                        RETURN
                        END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = RUNGE    , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =      32,PROGRAM SIZE =     1414
 *STATISTICS*  NO DIAGNOSTICS GENERATED)

NO DIAGNOSTICS FLAGGED IN THE ABOVE COMPILATIONS.
```

APPENDIX C

Appendix C

The algebraic formulation of the problem is given next.

$$\rho \bar{x}_b = \rho \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 \; |M_{xb}^n|^2\right)^{1/2} \tag{3}$$

$$\alpha \bar{y} = \alpha \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 \; |M_y^n|^2\right)^{1/2} \tag{4}$$

$$\text{where } \bar{h}_n^2 = h_n . h_n^*/2 \tag{5}$$

h_n^* is the complex conjugate of h_n the amplitude of the n^{th} order harmonic component of the terrain elevation spectrum h_n .

$$|M_{x_b}^n|^2 = \frac{N_1}{D_1} \tag{6}$$

where

$$N_1 = K^2 (\omega^2 c^2 n^2 + k^2) \tag{7}$$

$$D_1 = [k(1-(n\omega/\omega_o)^2 (m_a n^2 \omega^2 - (k+K)) + k^2]^2 + \tag{8}$$
$$(\omega cn)^2 [m_a (n\omega)^2 - K + k (n\omega/\omega_o)^2]^2$$

$$|M_y^n|^2 = \frac{N_2}{D_2} \tag{9}$$

where

$$N_2 = 1 \tag{10}$$

$$D_2 = [(\omega_o/n\omega)^2 - 1 + (n\omega)^2 m_a (1-\omega_o/n\omega)^2)/K - (k/K)]^2 +$$
$$(\omega cn)^2 [(\omega_o/n\omega)^2 - (m_a \omega_o^2 + k)/K]^2/k^2 \tag{11}$$

We use the following nomenclature:

: circular frequency of sprung mass when standing (constant) rad/sec

/L: impressed circular frequency rad/sec

: unsprung mass spring rate; N/m

k: sprung mass spring rate; N/m

c: damping constant of shock absorber (average) Nsec/m

M_a: axle mass per wheel set; kg

n: integer 1,2,3,..... order of vibration

$$B_{x2}=n^4\omega^4(c/\omega_o)^2 \tag{24}$$

$$A_{y1}=[(\omega_o/n\omega)^2-1+(n\omega)^2.(1-(\omega_o/n\omega)^2)m_a/K-(k/K)] \tag{25}$$

$$A_{y2}=(\omega_o/n\omega)^2-(m_a\omega_o^2+k)/K \tag{26}$$

$$B_{y1}=1/K \tag{27}$$

$$B_{y2}=(\omega cn/k)^2(A_{y2}/k+1/K) \tag{28}$$

Substitution of (14) to (28) into (12) and (13), respectively, will yield two simultaneous algebraic equations in the unknown parameters k and c. Note that because of automatic height control with ω_o equal a constant the pro-rated mass m_b does not enter into the equations explicitly but is contained in ω_o.

For the terrain identification (subroutine) we write:

$$\phi^n_{xx} = \bar{h}_n^2/\Omega_n = \bar{h}_n^2\, u/\omega_n=\phi_t.u \tag{29}$$

where in (29) ϕ^n_{xx} is the n^{th} order power spectral density value (m^3/rad), Ω_n in the wave number of the n^{th} order wave (Rad/m), ω_n is the n^{th}order circular frequency (rad/sec) and $\phi^n_t=\bar{h}_n^2/\omega_n$ (m^2sec/rad) is the n^{th} order power spectral density value relative to time. So

$$\phi^n_{xx} = u\phi^n_t \tag{30}$$

also

$$\phi = Au^2 \text{ (approximately)} \tag{31}$$

and

$$\phi_t = \bar{h}_n^2/\omega \tag{32}$$

such that

$$\bar{h}_n^2 = Au/\omega = A/\Omega = \phi_{xx}.\Omega \tag{33}$$

The average A- value A_{avg} is then

Partial differentiating (1) makes P a minimum, if

$$\frac{\partial P}{\partial k} = \rho \frac{\partial \bar{\bar{x}}_b}{\partial k} + \alpha \frac{\partial \bar{y}}{\partial k} = 0 \tag{12}$$

$$\frac{\partial P}{\partial c} = \rho \frac{\partial \bar{\bar{x}}_b}{\partial c} + \alpha \frac{\partial \bar{y}}{\partial c} = 0 \tag{13}$$

where

$$\frac{\partial \bar{\bar{x}}_b}{\partial k} = \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 |M_{x_b}^n|^2\right)^{-1/2} \cdot \sum_{n=1}^{n=\infty} \bar{h}_n^2 n^4 \frac{\partial |M_{xb}^n|^2}{\partial k} \tag{14}$$

$$\frac{\partial \bar{y}}{\partial k} = \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 |M_{xb}^n|^2\right)^{-1/2} \cdot \sum_{n=1}^{n=\infty} \bar{h}_n^2 \frac{\partial |M_y^n|^2}{\partial k} \tag{15}$$

$$\frac{\partial \bar{\bar{x}}_b}{\partial c} = \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 |M_{xb}^n|^2\right)^{-1/2} \cdot \sum_{n=1}^{n=\infty} \bar{h}_n^2 \frac{\partial |M_{xb}^n|^2}{\partial c} \tag{16}$$

$$\frac{\partial \bar{y}}{\partial c} = \left(\sum_{n=1}^{n=\infty} \bar{h}_n^2 |M_y|^2\right)^{-1/2} \cdot \sum_{n=1}^{n=\infty} \bar{h}_n^2 \frac{\partial |M_y^n|^2}{\partial c} \tag{17}$$

$$\frac{\partial |M_{xb}^n|^2}{\partial k} = 2[D_1 kK - N(A_{x1} B_{x2})]/D_1^2 \tag{18}$$

$$\frac{\partial |M_{xb}^n|^2}{\partial c} = 2\, cn^2\omega^2 \cdot [D_1 K^2 - N_1 A_{x2}^2]/D_1^2 \tag{19}$$

$$\frac{\partial |M_y^n|^2}{\partial k} = -2[A_{y1} B_{y1} + A_{y2} B_{y2}]/D_2^2 \tag{20}$$

$$\frac{\partial |M_y^n|^2}{\partial c} = 2c\omega^2 n^2 A_{y2}^2/k^2 \tag{21}$$

and

$$A_{x1} = k(1-(n\omega/\omega_o)^2)(m_a n^2\omega^2 - (k+K)) + k^2 \tag{22}$$

$$A_{x2} = (m_a n^2\omega^2 - K + k(n\omega/\omega_o)^2)$$

$$B_{x1} = (1-(n\omega/\omega_o)^2)(K+m_a n^2\omega^2)(K+m_a n^2\omega^2) + 2k(n\omega/\omega_o)^2 \tag{23}$$

$$A_{avg} = \left(\sum_{n=1}^{n=100} \phi_{xx}\, \Omega_n^2\right)/100 \qquad (34)$$

Equations (1) through (34) are processed and the program is shown next.

(11b)

```
C
C     AM = BODY MASS, AM = AXLE MASS, TK = TIRE SPRING
C     RATE, SK = SUSPENSION SPRING RATE, C = DAMPING
C     COEFFICIENT, NP = NUMBER OF DATA POINTS, V = SP
C     EED OF THE VEHICAL, AL = LENGTH OF THE SPAN,
C     (FOR FFT NP=2**N, WHERE N IS AN INTEGER)
C
C
0001        IMPLICIT REAL *8(A-H,O-Z)
0002        COMPLEX *16 H(128), SP(128), SHNEW(128)
0003        INTEGER RUNGE
0004         DIMENSION ROOT(2), WA(15)
0005        DIMENSION CCCCC(2,3)
0006        DIMENSION XA(128),A(128),XP(128),DX2A(128),DX2B(128)
0007        DIMENSION Y(4),DERY(4),PRM(5),AUX(16,4)
0008          DIMENSION HH(128)
            DIMENSION ORIGH(128)
0010        DIMENSION F(128),PHASEF(128)
0011        COMMON/FFF/SH,NP
0012        COMMON/XAB/XA,XB
0013        COMPLEX *16 SSH(128),DIFH(128)
            DIMENSION S1(128), S2(128), OPSD(128)
            COMMON/VEDATA/AM,BM,TK,SK,C,PI
            COMMON/PROFIL/V,AL
            COMMON/PLHH/PLH
            INTEGER RUNGE
            PI = 3.14159265358979D0
              A(5,203)AM,BM,TK,SK,C
            READ(5,204)V,AL,HCLEAR,NP
            FORMAT(5F10.6)
            FORMAT(3F10.6,I3)
              WRITE(6,101)AM,BM,TK,SK,C
              WRITE(6,102)V,AL,NP
            FORMAT(1H1//5X,'***  VEHICLE SUSPENSION DESIGN  ***'/
           ,' ********************************************'/
           ,' INPUT DATA FOR THE SYSTEM :'/
           ,'  SPRUNG MASS                = ',F10.2,' '/
           ,'  UNSPRUNG MASS              = ',F10.2,' '/
           ,'  TIRE SPRING RATE           = ',F10.2,' '/
           ,'  SUSPENSION SPRING RATE     = ',F10.2,' '/
           ,'  DAMPING COEFFICIENT        = ',F10.2,' '/
           ,' ---------------------------------------'/
                 ,'INPUT DATA:'/
           ,'  SPEED OF THE VEHICLE = ',F10.2,' M/SEC'/
           ,'  HORIZONTAL SPAN LENGTH: L = ',F7.2,' '/
           ,'  NUMBER OF OBSERVATION POINTS = ',I5/
           ,' ---------------------------------------'/)
            CALL FFT(NP,SH)
            DO 12 I=1,128
12          SSH(I) = SH(I)
C
            U = 4.0
            N = 7
            CALL FFT(SP,N,MP)
C
```

(Please print — better copy unavailable G.S.)

```
C
C
30    FORMAT('SM IN MAIN',2F17.5)
      READ(5,215)(((CCCCC(J,I),J=1,2),I=1,3)
215   FORMAT(2F17.4)
      RCH = 10.0
C
C     WRITE(6,310)ROH
310   FORMAT('ROH = ',E20.7)
C
C
      ITMAX = 100
      EPS = 1.0D-4
      CALL ROOTS1(RCOT,CCCCC)
      WRITE(6,301)RCOT
C
      SK = ROOT(1)
      C = RCOT(2)
      CALL PSDCAL(SK,S1,NP)
      CALL FUN(F)
      DO 14 I=1,128
14    PHASEF(I) = F(I)
      DO 15 I=2,128
15    F(I) = PHASEF(I-1)
      F(1) = -PHASEF(128)
      T =0.0
C
C
      TMAX = AL/V
      DELT = TMAX/128.0
      NDIM = 4
C
C
      DO 3 I=1,NDIM
      Y(I) = 0.0
3     K = KUTGE(4,Y,DERY,T,DELT)
      IF(K.NE.1)GO TO 5
      DERY(1) = Y(3)
      DERY(2) = Y(4)
      DERY(3) = SK*(Y(4) - Y(5)) + SK*(Y(2) - Y(1))
      T = T+DELT
      II = IFIX(SNGL(X))
      FF = X-II
      IF(ABS(FF).GT.0.75)II=II+1
      FF = F(II)
      XNEW =II
      IF(ABS(X-XNEW).LT.0.1)GO TO 7
      FF = (F(II+1)+F(II))/2.0
      GO TO 6
7     Y(1) = Y(1)
      Y(2) = Y(2)
      DX2A(II) = (1./AM)*(CCC-TK*(Y(1)-FF))
      DX2B(II) = (1.0/BM)*(-CCC)
6     DERY(3)=(1.0/AM)*(CCC-TK*(Y(1)-FF))
      DERY(4) = (1.0/BM)*(-CCC)
C
```

(Please Print - letter copy unavailable) Q.S.

```
                  WRITE (6,311)Y,DELY
          311     FORMAT(2 17.6)
          5       CONTINUE
                  IF (T.LT.TMAX)GO TO 3
                C
                C     CONSTRAINT EQUATION FOR OPTIMIZATION
                C
                C
                      YBAR = 0.0
                C     WRITE(6,305)YBAR
          305   FORMAT('YBAR',E20.7)
                      DO 20 J1=1,128
                C     WRITE(6,305)YBAR
          20          YBAR = YBAR + (XB(J1)  - XA(J1))**2
                      YBAR = DSQRT(YBAR/128.0)
                      ALPHA = 3.0
                      HCALC = ALPHA*YBAR
                      WRITE(6,335)HCLEAR,HCALC
          335   FORMAT('HHHH =',2E17.5)
                      IF(HCLEAR.GE.(ALPHA*YBAR))GO TO 22
                      YBAR = 0.)
                      GO TO 23
          22          WRITE(6,301)RGHT
          301         FORMAT('ROOTS ARE', 2E17.7)

                      CALL FFT(SSH,XA,XB,NN,NP)
                      DO 12 I=1,128
          12          SSH(I) = REAL(SSH(I))
                      DO 13 I=1,NP
          13          SHNEW(I) = DCMPLX(HH(I),0.0D0)
                      CALL FFT(SHNEW,N,NP)
                      CALL PSDCAL(SHNEW,S2,NP)
                      WRITE (6,202)
                      DO 3 J=1,NP
                      PSD(J)=(PSDF(J)-HH(J))*100.0/PSDF(J)
                      WRITE(6,203)I(J),HH(J),PSD(J)
                      FORMAT('1///,'   COMPARISON BETWEEN
                     *' DATA',10X,'COMPUTED DATA',10X,'
                      FORMAT(I4,E22.7/)
                     *,'
                      WRITE(6,1)
                    1 FORMAT('1',      LOGARITHMIC OUTPUT
                     *,'FREQUENCY',2X,'PSD',14X,'LOG(FREQ)',2X,'LOG
                      WL = 2.0*PI*V/AL
                      BETA=BA*DSQRT(BR/SR)
                      WRITE(6,*)' CONDITIONS OF PSD AND FREQUENCY VALUES
                      DO 4 I=1,NP
                      PSDL=DLOG10(PSD(I)*BETA*WL/(2.0*PI))
                      FRQL = I*2.0*PI/AL
                      PSDL=DLOG10(PSD)
                      FRQL = DLOG10(FRQ)
                      IF((I.EQ.1).OR.(I.GT.65))GO TO 16
                      WRITE(6,210)FRQ,PSD,FRQL,PSDL
          16          CONTINUE
```

(Please Print better Copy unavailable) G. S.

```
      21      FORMAT(5X,4L17.6)
      9       CONTINUE
              STOP
              END
*OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*   NAME = MAIN      , LINECNT =      57
*STATISTICS*     SOURCE STATEMENTS =      121,PROGRAM SIZE =    23170
*STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
C     SUBROUTINE SOLVE A SYSTEM OF NONLINEAR EQUATIONS
C     USING A METHOD WHICH DOES NOT NEED THE DERIVATIVE CALCULA
      SUBROUTINE ROOTS1(X,C)
      IMPLICIT REAL *8(A-H,O-Z)
      DIMENSION B(3,3),C(2,3),X(2),SE(2),E(3),LV(3),VV(3)
      DIMENSION B1(3,3)
      EMIN = 1.0E-7
      N = 2
      NP1 = N + 1
C
110   FORMAT(2E17.7)
C
      DO 3 I = 1,NP1
      B(1,I) = 1.0
      DO 2 J = 1,N
2     X(J) = C(J,I)
      CALL ERR(X,SE)
      DO 11 J=2,NP1
11     B(J,I) = SE(J - 1)
3     CONTINUE
C
51    DO 31 I =1,NP1
      DO 21 J =1,NP1
21    B1(I,J) = B(I,J)
      CALL MINV(B1,NP1,D,LV,VV)
      IF(D.EQ.0.0)GO TO 22
C
      DO 4 J=1,N
      X(J) = 0.0
      DO 4 I = 1,NP1
4     X(J) = X(J) + C(J,I)*B1(I,1)
      CALL ERR(X,SE)
      EE = 0.0
      DO 5 I=1,N
5     EE = EE+SE(I)*SE(I)
      EE = DSQRT(EE)
C
      IF(EE.LE.EMIN)GO TO 21
      WRITE(6,203)EE,X
203   FORMAT(2X,3E17.6)
      L = 1
      DO 7 I = 1,NP1
      E(I) = 0.0
      DO 6 J = 2,NP1
6     E(I) = E(I) + B(J,I)*B(J,I)
      E(I) = DSQRT(E(I))
      IF(I.EQ.1)GO TO 7
      IF(E(I).GT.E(I-1))L=I
7     CONTINUE
      DO 8 I=1,N
      B(I+1,L) = SE(I)
8     C(I,L) = X(I)
      GO TO 51
21      WRITE(6,201) X
201   FORMAT(2E17.6)
```

(Please Print better
Copy unavailable
G. S.)

```
                      SE
                22    WRITE (
                702   FORMAT(' MATRIX B IS SINGULAR')
                23    CONTINUE
                      RETURN
                      END
*OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*   NAME = ROOTS1    , LINECNT =      57
*STATISTICS*     SOURCE STATEMENTS =      54,PROGRAM SIZE =    1796
*STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
C
C     ***********SUBROUTINE ERR
      SUBROUTINE ERR(SUS,EQN)
      IMPLICIT REAL *8(A-H,O-Z)
      COMMON/EEE/SH,NP
      COMMON/VEDATA/AM, BM, TK, SKNCT, CNCT, PI
      COMMON/PROFIL/U,AL
      DIMENSION EQN(2)
      COMMON/ROHH/ROH
      DIMENSION SUS(2), H(128)
      COMPLEX *16  SH(128), SHSH
C
      C = 34600.0
      SK = 500.0
      DO 222 ISK=1,400
      SK = ISK*500.0
C     WRITE(6,101)SK
101   FORMAT('SK =',2D17.6)
C
      ALPH = 3.0
C     WRITE(6,402)AM,BM,TK,PI
402   FORMAT('AM,BM,TK,PI ',4D15.6)
      DO 12 I=1,128
      SHSH = SH(I)
      SHSH = SHSH*DCONJG(SHSH)/2.0
      PHH = DSQRT(PHH)
      H(I) = PHH
C
C     WRITE(6,411)SUS
411   FORMAT('SK & C ',2D17.6)
C     SK = (SUS(1))
      IF(SK.LT.1.0D2)SK = 100000.0
C     C = (SUS(2))
      IF(C.LT.1).D2)C = 1000.0
      AL = DSQRT(SK/BM)
         .IMPIAL/75.0
C
      DO 1 IN
      XX(I) = 0.0
      YY(I) = 0.0
      XX(I) = 0.0
      YY(I) = 0.0
      XX(I) = 0.0
      YY(I) = 0.0
      XX(I) = 0.0
      YY(I) = 0.0
C
C
      DO 1 N = 1,NP
      AN = N(N)
      PPI = DFLOAT(N*N)**AM
```

(Please Print +- better copy if available. T.S)

```
C
410   FORMAT('  PR2 =',D17.7)
      AX1 = SK*(1.-PR2)*(AM*PR1-(SK+TK))+SK*SK
      AX2 = AM*PR1 - TK + SK*PR2
      D1 = AX1*AX1+C*C*PR1*AX2*AX2
C
      D21 = RPR2 -1. + (PR1*AM/TK)*(1.-RPR2)-(SK/TK)
      D22 = RPR2 - (AM*WO*WO + SK)/TK
C
      D2 = D21*D21 + (PR1*C*C/(SK*SK))*D22*D22
      BX1 = (1.-PR2)*(TK+AM*PR1)+2.*SK*PR2
      BX2 = PR1*PR2*C*C
C
C     WRITE(6,403)PR2
403   FORMAT('PR2 =',E17.6)
C     WRITE(6,404)D1,D2
404   FORMAT('D1D2 =',2E17.6)
      RPR2 = 1./PR2
      AY1 = RPR2 -1. +(PR1*AM/TK)*(1.-RPR2) -(SK/TK)
      AY2 = RPR2 - (AM*WO*WO+SK)/TK
C
      BY1 = 1./TK
      BY2 = (PR1*C*C/(SK*SK))*((AY2/SK)+BY1)

      ANUM1 = TK*TK*(PR1*C*C+SK*SK)

      HN2 = HN*HN
      FPR1 = HN2/(D1*D1)
      FPR2 = HN2/(D2*D2)
      RCHN1 = RCHN1 - FPR1*(D1*SK*TK-ANUM1*(AX1*BX1+AX2*BX2))
      ALFA1 = ALFA1+FPR2*(AY1*BY1+AY2*BY2)
      RCHD = RCHD + HN2*ANUM1/D1
      ALFD = ALFD + HN2/D2

      RCHN2 = RCHN2+ FPR1*(D1*TK*TK-ANUM1*AX2*AX1)
      ALFA2 = ALFA2+ FPR2*AY2*AY2
      WRITE(6,409)RCHN1,RCHN2,ALFA1,ALFA2
409   FORMAT('RCHN1 & 2 ',2D17.6/'  ALFA1 &2',2D17.6)
      DMXK=DMXK+2.*(D1*SK*TK-ANUM1*(AX1*BX1+AX2*BX2))/(D1*D1)
      DMYK=DMYK-2.*(AY1*BY1+AY2*BY2)/(D2*D2)
      DMXC=DMXC+(2.*C*PR1/(D1*D1))*(D1*TK*TK-ANUM1*AX2*AX2)
      DMYC=DMYC+2.*C*PR1*AY2*AY2/(SK*SK)
     1 CONTINUE
      WRITE(6,412)C,SK,DMXK,DMYK,DMXC,DMYC
412   FORMAT('CSKDMXKDMYKDMXCDMYC',6D18.6)
C
222   CONTINUE
C
C     COEFFICIENT OF EQUATIONS 28 AND 29
C
      DD1 = DSQRT(RCHD)
      DD2 = DSQRT(ALFD)
C
      WRITE(6,405)DD1,DD2
```

Please run it - with copy - unanimable

```
                  409   FORMAT('DOLDD2 ',2E17.6)
                        WRITE(6,408)ALFA1,ALFA2
                  408   FORMAT(' ALFA1 ALFA2 ',2D17.6)
                        CR1 = RCHN1/DD1
                        CR2 = RCHN2/DD1
                        CA1 = ALFA1/DD2
                        CA2 = (C*W*W/(SK*SK))*ALFA2/DD2
                        WRITE(6,407)CR1,CR2,CA1,CA2
                  407   FORMAT('  CR1 CR2 ',2D17.6/'   CA1 CA2 ',2D17.6)
                        C
                        RATIL = ALPH/ROH
                        WRITE(6,204)RATIL
                  204   FORMAT('ROH=',D17.6)
                        RATIO1=CR1/CA1
                        RATIO2=CR2/CA2
                        WRITE(6,202)RATIO1,RATIO2
                  202   FORMAT('IN DQN',2E18.6)
                        C
                  203   FORMAT('ROH =',E17.6)
                  21    EQN(1) = RCH*CR1 + ALPH*CA1
                  22    EQN(2) = ROH*CR2 + ALPH*CA2
                        RETURN
                        END
OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
OPTIONS IN EFFECT*   NAME = ERR      , LINECNT =      57
STATISTICS*     SOURCE STATEMENTS =       97,PROGRAM SIZE =     4004
STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
MICHIGAN TERMINAL SYSTEM FORTRAN G(4127G)                    MAY                    26-07-

C
C      ***********FUN
0001          SUBROUTINE FUN(F)
0002          DIMENSION F(128)
0003          COMPLEX A(128),AF(128),CMPLX
0004           COMMON/FFF/A,NP
0005          COMMON/VEDA/AM,BN,TK,SK,C,PI
0006          COMMON/PROFIL/V,AL
0007           DO 4 I=1,NP
0008          F(I) = 0.0
0009          ANP = FLOAT(NP)
0010          DO 5 J=1,NP
0011          NI = I*J
0012          JJ = NI/NP
0013          ANI = FLOAT(NI)
0014          TH = 2.0*PI*((ANI/ANP)-JJ)
0015          SS = SIN(TH)
0016          CC = COS(TH)
0017          AF(I) = AF(I) + A(J)*CMPLX(CC,SS)
0018        5  CONTINUE
0019          APF=REAL(AF(I))
0020          AIF = AIMAG(AF(I))
0021          RATI =APF/AIF
0022          IF(I.LE.64)GO TO 2
0023          AIF = -AIF
0024        2  F(I) = AIF*SQRT(1.0+RATI*RATI)
0025        4  CONTINUE
0026          RETURN
0027          END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = FUN      , LINECNT =      57
 *STATISTICS*    SOURCE STATEMENTS =      27,PROGRAM SIZE =     1904
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

AD-A068 405 WAYNE STATE UNIV DETROIT MICH DEPT OF MECHANICAL ENG--ETC F/G 13/6
AUTOMOTIVE SUSPENSION CONTROL.(U)
OCT 78 H K SACHS

UNCLASSIFIED DAAK30-77-C-0066
TARADCOM-TR-12377 NL

2 OF 2
AD A068 405

END
DATE FILMED
6-79
DDC

```
C
C       SUBROUTINE ROUGH STARTS
0001          SUBROUTINE ROUGH(N,H)
0002          DIMENSION H1(1024)
0003          COMPLEX H(N)
0004          REAL*4 FMT(1)/'*'/
0005          READ(5,FMT)(H1(I),I=1,N)
0006          S=0.0
0007          DO 2 I=1,N
0008        2 S=S+H1(I)
0009          HAV = S/FLOAT(N)
0010          DO 1 I=1,N
0011          H1(I) = H1(I)-HAV
0012          H1(I) = H1(I)*0.3043
0013        1 H(I) = CMPLX(H1(I),0.0)
C           WRITE(6,101) (H(I),I=1,N)
0014      101 FORMAT('1H1////' TERRAIN ROUGHNESS INPUT :'/
           2?X,'**********'**********'**********'///
           32X,'HEIGHTS W.R.T. A REFRENCE'///
           3(2X,4F12.5/))
0015          RETURN
0016          END
```

OPTIONS IN EFFECT ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
OPTIONS IN EFFECT NAME = ROUGH , LINECNT = 57
STATISTICS SOURCE STATEMENTS = 16,PROGRAM SIZE = 4862
STATISTICS NO DIAGNOSTICS GENERATED

```
TOMSAN TERMINAL SYSTEM FORTRAN G(61896)                    MAIN

      C
      C     SUBROUTINE PSDCAL
      C     SUBROUTINE PSDCAL(H,S,NP)
0001        COMPLEX H(NP)
0002        DIMENSION S(NP)
0003        COMMON/VEDATA/AM,BM,TK,SK,C,PI
0004        COMMON/PROFIL/V,AL
0005        OMEGA = 2.*PI*V/AL
0006        ETA = OMEGA*SQRT(BM/SK)
0007        DO 1 I=1,NP
0008       1S(I) = H(I)*CONJG(H(I))/(2.*ETA)
0009        RETURN
0010        END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = PSDCAL   , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =           11,PROGRAM SIZE =      548
 *STATISTICS*   NO DIAGNOSTICS GENERATED
```

```
TERMINAL SYSTEM FORTRAN V(41336)

      C
      C     SUBROUTINE FFT
0001        SUBROUTINE FFT(A,N,NB)
0002        COMPLEX A(N8),U,W,T,CMPLX
      C     DIVIDING ALL ELEMENTS BY NB
0003        DO 1 J=1,NB
0004      1 A(J) = A(J)/NB
      C     REORDERING THE SEQUENCE
0005        NBD2 = NB/2
0006        NBM1 = NB - 1
0007        J = 1
0008        DO 4 L = 1,NBM1
0009        IF(L.GE.J)GO TO 2
0010        T = A(J)
0011        A(J) = A(L)
0012        A(L) = T
0013      2 K=NBD2
      C     WRITE(6,101)(A(I),I=1,6)
0014      101 FORMAT(3F15.6)
0015      3 IF(K.GE.J)GO TO 4
0016        J = J - K
0017        K = K/2
0018        GO TO 3
0019      4 J = J+K
      C     COMPUTATION OF FFT
0020        PI = 3.14592653589793
0021        DO 6 M=1,N
0022        U = (1.0,0.0)
0023        ME = 2**M
0024        K = ME/2
0025        W = CMPLX(COS(PI/K), -SIN(PI/K))
0026        DO 6 J=1,K
0027        DO 5 L=J,NB,ME
0028        LPK = L+K
0029        T = A(LPK)*U
0030        A(LPK) = A(L) - T
      C     WRITE(6,102)A(LPK)
      102 FORMAT(2F13.5)
0031      5 A(L) = A(L)+T
0032      6 U = U*W
0033        RETURN
0034        END

OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
OPTIONS IN EFFECT*   NAME = FFT     , LINECNT =      57
STATISTICS     SOURCE STATEMENTS =      35,PROGRAM SIZE =     190
STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
C
C       SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
0001          SUBROUTINE HCAL(DX2A,DX2B,XA,H,NP)
0002          COMPLEX H(NP),XA(NP),DX2A(NP),DX2B(NP)
0003          COMMON/VEDATA/AM,BM,TK,SK,C,PI
0004          C1 = AM/TK
0005          C2 = BM/TK
0006          DO 1 I = 1,NP
0007        1 H(I) = XA(I) + C1*DX2A(I) + C2*DX2B(I)
0008          RETURN
0009          END
 *OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
 *OPTIONS IN EFFECT*   NAME = HCAL      , LINECNT =      57
 *STATISTICS*     SOURCE STATEMENTS =          9,PROGRAM SIZE =      572
 *STATISTICS*  NO DIAGNOSTICS GENERATED
```

```
MICROSOFT TERMINAL SYSTEM FORTRAN G[41336]          MAIN
      C     FUNCTION SUBPROGRAM FOR 4TH ORDER RUNGE-KUTTA METHOD
0001        FUNCTION RUNGE(N,Y,F,K,H)
0002        INTEGER RUNGE
0003        DIMENSION PHI(50),SAVEY(50),Y(N),F(N)
0004        DATA M/0/
0005        M = M + 1
0006        GO TO (1,2,3,4,5),M
0007  1     RUNGE=1
0008        RETURN
0009  2     DO 22 J=1,N
0010        SAVEY(J) = Y(J)
0011        PHI(J) = F(J)
0012  22    Y(J) = SAVEY(J) + .5*H*F(J)
0013        K = K + .5*H
0014        RUNGE = 1
0015        RETURN
0016  3     DO 33 J=1,N
0017        PHI(J) = PHI(J) + 2.*F(J)
0018  33    Y(J) = SAVEY(J)+.5*H*F(J)
0019        RUNGE=1
0020        RETURN
0021  4     DO 44 J=1,N
0022        PHI(J) = PHI(J)+2.*F(J)
0023  44    Y(J) = SAVEY(J) + H*F(J)
0024        K = K + 0.5*H
0025        RUNGE = 1
0026        RETURN
0027  5     DO 55 J=1,N
0028  55    Y(J) = SAVEY(J)+(PHI(J)+F(J))*H/6.
0029        M = 0
0030        RUNGE = 0
0031        RETURN
0032        END
*OPTIONS IN EFFECT*   ID,EBCDIC,SOURCE,NOLIST,NODECK,LOAD,NOMAP
*OPTIONS IN EFFECT*   NAME = RUNGE    , LINECNT =      57
*STATISTICS*     SOURCE STATEMENTS =       32,PROGRAM SIZE =     1416
*STATISTICS*  NO DIAGNOSTICS GENERATED)

NO STATEMENTS FLAGGED IN THE ABOVE COMPILATIONS.
```

DISTRIBUTION LIST

	No. of Copies
Commander	
U.S. Army-Automotive Research & Development Command	
ATTN: Chief Scientist, DRDTA-NS	1
Dir., TARAD Lab., DRDTA-R	1
Systems Development Div., DRDTA-RE	1
Foreign Intelligence Office, DRDTA-NF	1
Technical Library, DRDTA-UL	2
Track & Susp. Func., DRDTA-RCKT	10
Warren, MI 48090	
U.S. Army Tank-Automotive Material Readiness Command	
ATTN: Commanding General, DRSTA-CG	1
Project Manager Fighting Vehicle Systems DRCPM-FVS	1
Project Manager, M-60 Tank Development, DRCPM-M60TD	1
Project Manager, XM1 Tank System, DRCPM-GCM	1
Federal Republic of Germany Liaison Office, FRG-LNO	1
Directorate for Weapons Systems Management, DRSTA-W	1
U.S. Marine Corps Liaison Office, USMC-LNO	1
Canadian Forces Liaison Office, CDLS-(D)	1
Warren, MI 48090	
Commanding Officer	1
U.S. Army Foreign Science & Technical Center	
Munitions Building, AMSTA-CR-ME	
Washington, D.C. 20315	
Commander	
U.S. Army Material Development & Readiness Command	2
ATTN: DRCRD-DM-G	
5001 Eisenhower Avenue	
Alexandria, VA 22333	
Commander	5
Defense Documentation Center	
Cameron Station	
Alexandria, VA 22314	
Harry Diamond Laboratories	1
ATTN: Technical Reports Group	
Washington, D.C. 21438	

	No. of Copies
U.S. Naval Civil Engineer Research & Engineering Laboratory Construction Batallion Center Port Hueneme, CA 93041	1
Commanding General U.S. Army Test and Evaluation Command Aberdeen Proving Ground, MD 21005	2
Commander National Defense Headquarters ATTN: Scientific Advisor Ottawa, Ontario Canada KIA OK2	1
Chief Office of Naval Research Washington, D.C. 20390	1
Superintendent U.S. Military Academy West Point, N.Y. 10996	3
Superintendent U.S. Naval Academy Annapolis, MD 21402	1
U.S. Army Deputy Chief of Staff for Operations Washington, DC 20310	2
National Tillage Machinery Laboratory U.S. Department of Agriculture Auburn, AL 36830	1
Commander Naval Ship Systems Command PMS 384-2; DWA Washington, DC 20360	1
Commanding General U.S. Army Engineer Research and Development Center Fort Belvoir, VA 22060	1
President U.S. Army Armor Board Fort Knox, KY 40121	1

	No. of Copies
President U.S. Army Artillery Board Fort Sill, OK 73503	1
President U.S. Army Infantry Board Fort Benning, GA 31905	1
President U.S. Army Airbourne Electronic and Special Warfare Board Fort Bragg, NC 26307	1
President U.S. Army Maintenance Board Fort Knox, KY 40121	1
Commanding General U.S. Marine Corps Landing Forces Development Center Quantico, VA 22134	2
Commanding General Headquarters USARAL ATTN: ARAOD Seattle, Washington 98749	
Office, Chief of Research & Development Department of the Army Washington, DC 20310	2
Commandant of the Marines U.S. Marine Corps ATTN: AO-4H Washington, DC 20380	1
Commanding Officer U.S. Army General Equipment Test Activity ATTN: Transportation Logistics Test Directorate Fort Lee, VA 23801	1
Canadian Army Staff 2450 Massachusetts Avenue Washington, DC 20008	4

	No. of Copies
Director U.S. Army Engineer Waterways Experiment Station Corps of Engineers P.O. Box 631 Vicksburg, MS 39181	3
United States Naval Industrial College of the Armed Forces ATTN: Vice Deputy Commandant Washington, DC 20315	2

UNCLAS (18) TARADCOM

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
(19) 1. REPORT NUMBER TR-12377	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) AUTOMOTIVE SUSPENSION CONTROL.	5. TYPE OF REPORT & PERIOD COVERED (9) Final Rept.
	6. PERFORMING ORG. REPORT NUMBER None
7. AUTHOR(s) (10) Dr. Herbert K. Sachs	8. CONTRACT OR GRANT NUMBER(s) (15) DAAK30-77-C-0066
9. PERFORMING ORGANIZATION NAME AND ADDRESS Wayne State University Department of Mechanical Engineering & Applied Mechanics Detroit, MI 48202 use 391091	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (16) Proj. No. 1L663621A424 AMCMS 663621.4240011
11. CONTROLLING OFFICE NAME AND ADDRESS US Army Tank-Automotive Research & Development Command ATTN: DRDTA-RCKT Warren, MI 48090	12. REPORT DATE (11) Oct 1978
	13. NUMBER OF PAGES 99
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) (12) 105 p.	15. SECURITY CLASS. (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

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

18. SUPPLEMENTARY NOTES

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

Computer, Suspension, Simulation, Terrain, Model, Ride, Vehicles, Control, Spring, Damper, Ride Control, Adaptive Control, Random Vibrations, Power Spectral Density of Terrains, Airsprings, Hydropneumatic Springs

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

This report contains the software package for an adaptive optimal suspension control system relative to terrain random vibration disturbances. The proposed problem solution is shown to fall into two separate program categories: a) recognition of the terrain and parameter selection, by means of an on-board minicomputer or microprocessor, b) optimization of suspension parameters for arbitrary terrain configurations obtained from terrain statistics and executed on a centrally located stationary computer facility.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLAS 391091

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

The interface between the stationary computer facility and the on-board microprocessor is accomplished by means of a data bank prepared at the stationary facility and permanently stored in the memory of the on-board microprocessor. The suspension parameters are set by a servo-control unit on the vehicle which is activated by the microprocessor. The servo-control unit regulates the supply and release of air in the hydropneumatic suspension system, thereby increasing or decreasing the spring rate according to the optimal requirements. In a similar manner the damper orifice size is increased or diminished depending on the required effective damping parameter. If need arises, the vehicle can operate at fixed suspension parameters. The results of the investigation are shown in graph form.