FENN COLLEGE OF ENGINEERING
CLEVELAND STATE UNIVERSITY

(NASA-CR-164587) MINIMUM FUEL CONTROL OF A VEHICLE WITH A CONTINUOUSLY VARIABLE TRANSMISSION Final Technical Report, 1 Oct. 1978 - 30 Jun. 1980 (Cleveland State Univ.) 94 p HC A05/MF A01 CSCL 13F G3/85

N81-29035

Unclas 26940

Electrical Engineering

MINIMUM FUEL CONTROL OF A VEHICLE WITH A CONTINUOUSLY VARIABLE TRANSMISSION

Final Technical Report
NASA Grant No. NSG 3223

Grant Title: "Control of a Hybrid System with Flywheel Energy Storage and Continuously Variable Transmission"

Co-Principal Investigators:

James H. Burghart
John F. Donoghue

Grant Period: October 1, 1978-June 30, 1980

Department of Electrical Engineering
Cleveland State University
Cleveland, Ohio 44115

SUMMARY

The research described here was concerned with designing a control system for a vehicle with a heat engine and a continuously variable transmission. The objectives of control were to minimize fuel consumption and to achieve satisfactory dynamic response of vehicle variables as the vehicle was driven over a standard driving cycle. This is the first time that a control system design and evaluation has been attempted for this overall vehicle system. Even though the vehicle system was highly nonlinear, attention was restricted to linear control algorithms which could be easily understood and implemented on-board using a microcomputer. The effectiveness of these controllers in producing good dynamic behavior of the vehicle as well as minimum fuel consumption was demonstrated by simulation. Simulation results also revealed that the vehicle could exhibit unexpected dynamic behavior which must be taken into account in any control system design.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF TABLES

1.0 INTRODUCTION

An important current area of research involves the design and development of vehicles and vehicle components which will reduce the nation's unduly high dependence on imported petroleum. The Continuously Variable Transmission (CVT), whose ratio may take on any value within a certain range, is one component which may be used to ensure that the vehicle's engine is operated efficiently with respect to fuel consumption. The research described here was directed toward investigating the control problems associated with controlling the throttle and transmission ratio of a vehicle containing a CVT so that acceptable stable dynamic responses for all important vehicle variables are obtained while minimizing fuel consumption.

The vehicle under consideration is a 1,500 Kg conventional sedan with a six cylinder engine and a continuously variable transmission. The system is inherently nonlinear because the torque and speed relationships include the transmission ratio in a multiplicative form, the relationships among engine variables are nonlinear, and the vehicle aerodynamic drag and rolling resistance are functions of the square of the velocity. The presence of these nonlinearities significantly complicates the control system analysis and design. The work described here involves development of simplified models for the components of the system, the design of control systems to achieve the desired objectives and analysis of the closed-loop performance using digital simulation.

1.1 Previous Work

Several researchers have investigated ways to minimize energy consumption or maximize efficiency in a vehicle propulsion system [1,2,3,4,6,7,8,9]. Some of this work considered energy savings for the combination of a power plant and CVT [2,4,6] with little regard for overall control considerations, and some work focused on the design of control systems for this configuration when augmented by a flywheel [5,7,8]. In fact, there is little previous work which was directed toward the design of control systems for configurations which include a CVT but not a flywheel [4,8,9], and none of this work considered the complete vehicle system including drive train and road losses with evaluation of the control systems for a standard driving cycle.

1.2 Study Objectives

The objective of this research was to investigate control problems associated with the control of a vehicle containing a heat engine and a continuously variable transmission. The objective was to design control systems which would operate the vehicle so that (i) the heat engine is operated at the minimum energy consumption point for any particular demanded torque and speed, and (ii) the overall system is operated with minimum energy consumption for a standard driving cycle.

It is important to note that this investigation, for the first time, had the objective of considering the overall problem of controlling a vehicle with CVT while the vehicle is subject to a driving regime prescribed by a standard driving cycle; all of the major elements of vehicle dynamics were included so that the control of the entire vehicle system was examined.

An important aspect of the objectives for this work was a determination to take an approach which would lead to easily implementable controllers. Some considerations leading to this approach are:

1. It is not possible to derive control algorithms for this system without undue simplifications because of its high degree of system nonlinearity. It is also not desirable to derive such controls because, in general, they would be in a form which would be difficult to implement.

2. Any derivation and calculation of optimal controls would necessarily be tied to a particular driving cycle, and the controls would be open loop. The results desired here (although to be evaluated with respect to a particular driving cycle) were not to be dependent upon any particular cycle.

3. It was desirable to devise control algorithms for this system which would be in a relatively simple, feedback form so that they could be implemented easily in a small, inexpensive, onboard microcomputer.

As a result of taking this approach, it was decided to use a control approach which was directed toward designing linear control loops (or algorithms) which use signals obtainable from vehicle sensors. As a result of this decision, more specific control objectives were defined. These objectives were to determine the appropriate types of linear controllers for three control loops which control the engine speed, throttle, and CVT ratio, and to optimize the parameters of these controllers.

Section 2 deals with a description of the vehicle and development of its models while Section 3 describes the details of the approach used to design the control system. An overview of the simulation is given in Section 4, with simulation results discussed in Section 5. Conclusions and future work are discussed in Section 6.

2.0 VEHICLE DESCRIPTION

The vehicle used in this study is a conventional sedan with a six cylinder engine. However, the transmission employed is a continuously variable transmission (CVT). In the subsections below, we outline the methods used to model the vehicle and the drive train components, and we present the equations which describe the dynamic behavior of the overall system. A more detailed description of these models and a detailed derivation of the system state equations are given in Appendix A.

2.1 Vehicle Modeling

The propulsion system of the vehicle consists of an internal combustion or heat engine (HE) coupled to a continuously variable transmission; the CVT, in turn, delivers power to a differential which connects the drive shaft to the rear axle. The overall arrangement is shown in Figure 2.1. The method of modeling each component in the drive train is discussed in the paragraphs below.

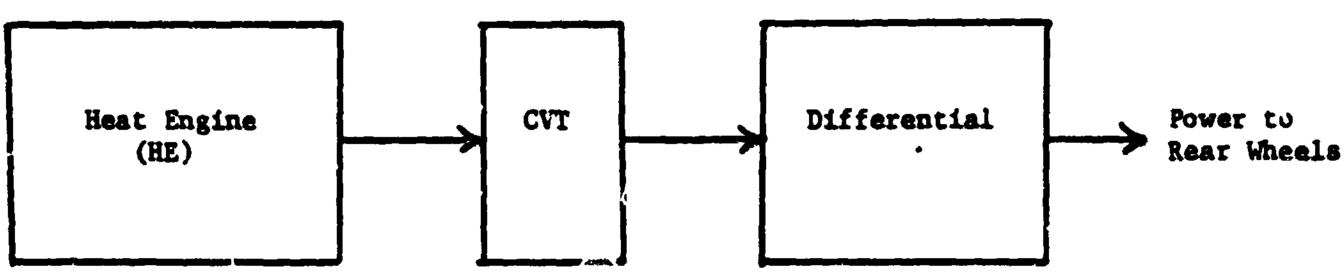

Figure 2.1 Drive Train Configuration

<u>Heat Engine (HE) Model</u>: The HE is modeled as a rotating inertia with an applied torque. The applied engine torque is the output of a first order low pass system whose input is a torque determined from an engine map. For a given engine speed and throttle setting, the engine map determines a steady state engine torque. This torque in turn is the input to the first order system whose output is the engine torque actually developed. The first order system (with a time constant of 1.0 second) approximates the dynamics of the throttle linkages and engine combustion characteristics (a step change in throttle does not produce a step change in developed engine torque). See Figure 2.2.

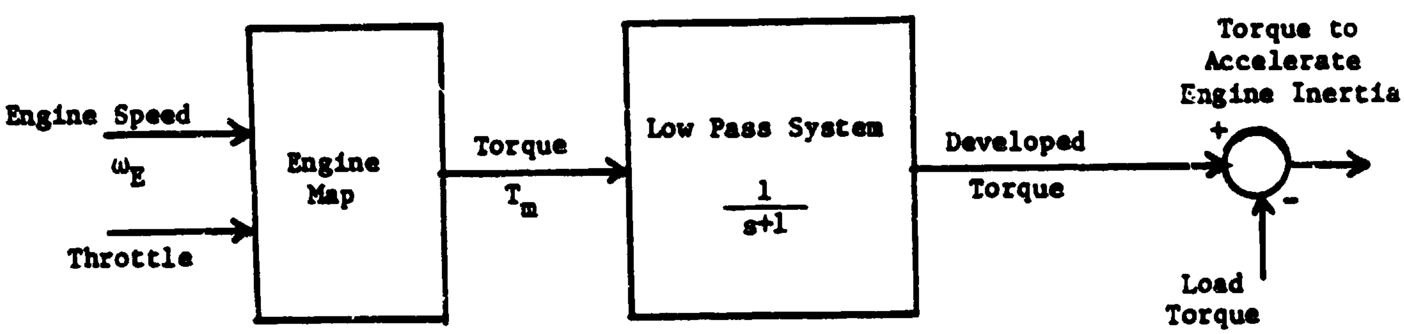

Figure 2.2 <u>Heat Engine Configuration</u>

<u>Continuously Variable Transmission</u>: The CVT is modeled as a ratio (which is defined to be the ratio of drive shaft speed to engine speed, and this is allowed to vary continuously from zero to infinity), an inertia and a fixed efficiency as

shown in Figure 2.3. The inertia is defined to be the equivalent CVT inertia as seen on the drive shaft side of the transmission. It was also assumed that there are second order dynamic effects associated with changing the ratio, i.e. if there is a step change in the variable which controlled the ratio, the actual ratio will respond with a second order response. This is also shown in Figure 2.3. Finally, we note that since the CVT ratio can be set to zero, no clutch is required between the HE and CVT.

Differential: The differential is modeled in exactly the same way as the CVT except that the ratio is fixed. The ratio is defined as the ratio of axle speed to drive shaft speed.

Vehicle Body: The model of the vehicle body and wheels takes into account the weight of the vehicle and payload, aerodynamic drag, rolling resistance and grade.

A detailed description of all the above models and how they are used to derive the differential equations which describe the overall vehicle dynamics is given in Appendix A. Appendix B contains the actual parameter values employed when simulating the system.

2.2 System Differential Equations

As described in Appendix A, there are four states which characterize the dynamic behavior of the overall vehicle system:

x_1 = Engine speed.

x_2 = Developed engine torque.

x_3 = CVT ratio.

x_4 = Rate of change of CVT ratio ($=\dot{x}_3$).

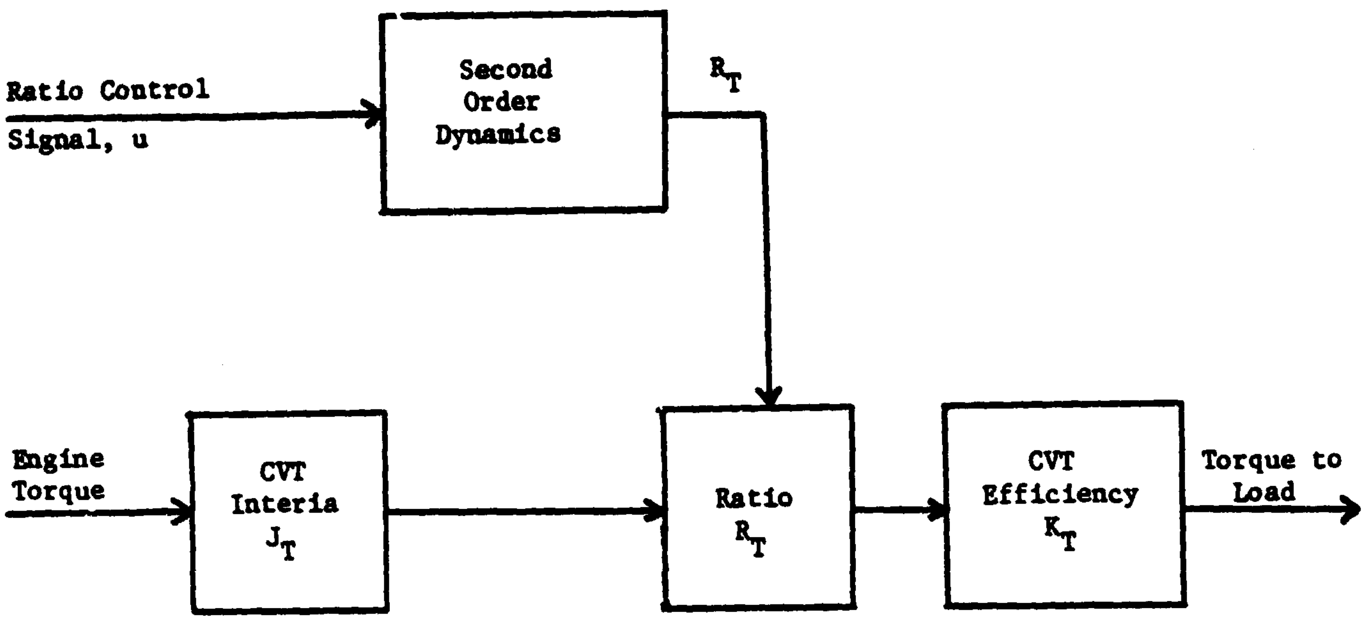

Figure 2.3 CVT Model

Since there is essentially no slippage between the engine and the rear wheels, the speed of the drive shaft and rear axle, as well as the vehicle velocity, can be computed from the engine speed, x_1. Also, since there is assumed to be a first order dynamic between a change in throttle setting and the corresponding change in developed engine torque, the latter, x_2, is also a state. Finally, since the CVT model includes a second order response of ratio to a change in the variable which controls the ratio, x_3 and x_4 are required states from the CVT model.

The four state equations which describe the vehicle dynamics are derived in Appendix A and are given as

$$\dot{x}_1 = [x_2 - M_T x_3(\phi_1 + \phi_2 + R_A R_W x_1 x_4)] / (J_E + M_T R_A R_W x_3^2) \qquad (2.2)$$

$$\dot{x}_2 = -[x_2 - \phi_3(x_1, u_1)] / \tau_L \qquad (2.3)$$

$$\dot{x}_3 = x_4 \qquad (2.4)$$

$$\dot{x}_4 = C_1 x_4 + C_2 x_3 + C_3 u_2 \qquad (2.5)$$

where

$$\phi_1(x_1, x_3) = R_A R_W [D(x_1, x_3) + R(x_1, x_3)] / M_T K_A K_T$$

$$\phi_2 = R_A (T_B + R_W W \sin \ ) / M_T K_A K_T$$

$$M_T = (J_T / R_A R_W) + (R_A J_A / K_T R_W) + (R_A R_W M_V / K_A K_T)$$

$$M_V = (W/g) + J_W R_W^2$$

and where u_1 is throttle setting, $\phi_3(x_1, u_1)$ is the steady state engine torque determined from the engine map (a function of throttle setting and engine speed), τ_L is the time constant of the throttle linkage/engine dynamic response, W is the vehicle weight, g is the acceleration of gravity, D and R are the aerodynamic drag and rolling resistance forces, respectively (both functions of x_1), β is the grade angle, R_W is the wheel radius, T_B is the braking torque, J_E, J_W, J_T and J_A are the rotating inertias of the engine, all four wheels, the CVT and the differential, respectively, K_A and K_T are the efficiencies of the differential and the CVT, respectively, R_A is the differential ratio, C_1, C_2 and C_3 are constants used to characterize the second order dynamic response of the ratio changing mechanism, and u_2 is the variable used to produce a change in the ratio (this variable will be generated by a ratio controller which is discussed in the next section). The vehicle velocity, v, can be computed from the above states by

$$v = R_A R_W x_1 x_3 \tag{2.6}$$

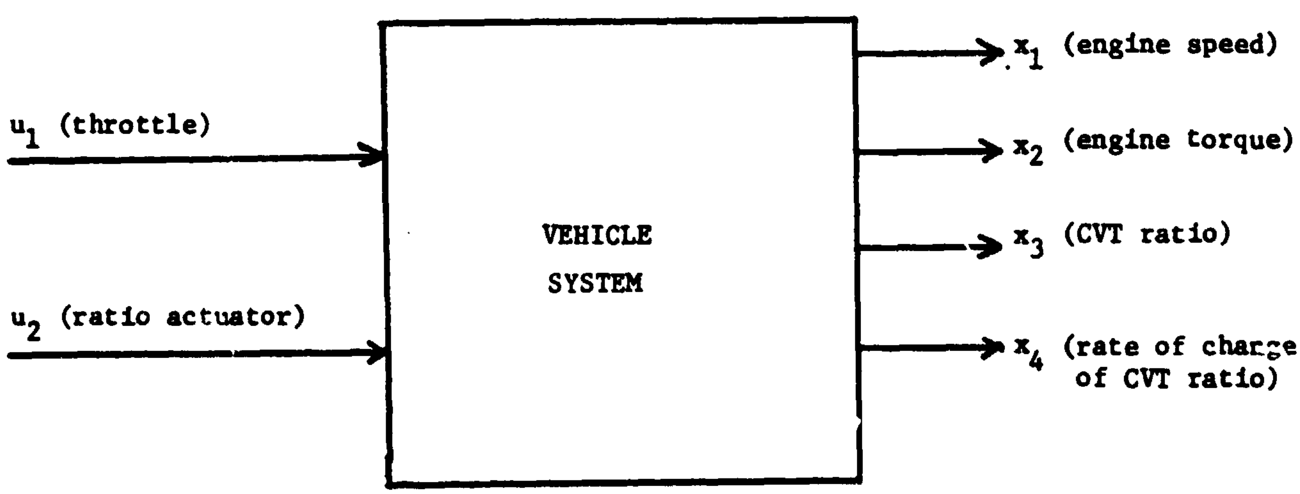

Figure 2.4 Block Diagram of Vehicle System

We note that two of the above state equations are highly nonlinear ((2.2) and (2.3)), thus complicating the control system design. A block diagram of the overall vehicle system showing the states and inputs is shown in Figure 2.4. The control inputs, u_1 and u_2, will be generated by the vehicle control system. Having described the characteristics of the system to be controlled, we now turn to a discussion of the design of the control system.

3.0 CONTROL SYSTEM DESIGN

The purpose of this section is to present the overall control philosophy for the vehicle system described in Section 2.0 and Appendix A, and to discuss the details of the control system designed to achieve the objectives which follow from this philosophy.

3.1 Control Objectives and Approach

The overall control objective is to manipulate throttle setting and CVT ratio changing input so that the vehicle achieves minimum fuel consumption while being driven over a standard driving cycle. An examination of a typical engine map shows that for each throttle setting there is an engine speed which achieves minimum fuel consumption for that throttle setting. In particular, a Minimum Fuel Curve can be developed from such a map which plots minimum fuel consumption engine speed against throttle setting. Such a curve (which is the one actually used in this study) is shown in Figure 3.1.

The existence of a minimum fuel consumption engine speed for each throttle setting suggests a specific control approach to achieve minimum fuel consumption: manipulate the throttle so that the vehicle follows the velocity requirements of the driving cycle, and, for each throttle setting, manipulate the CVT ratio so that the engine runs at a minimum fuel consumption speed for that throttle setting. The primary objective of

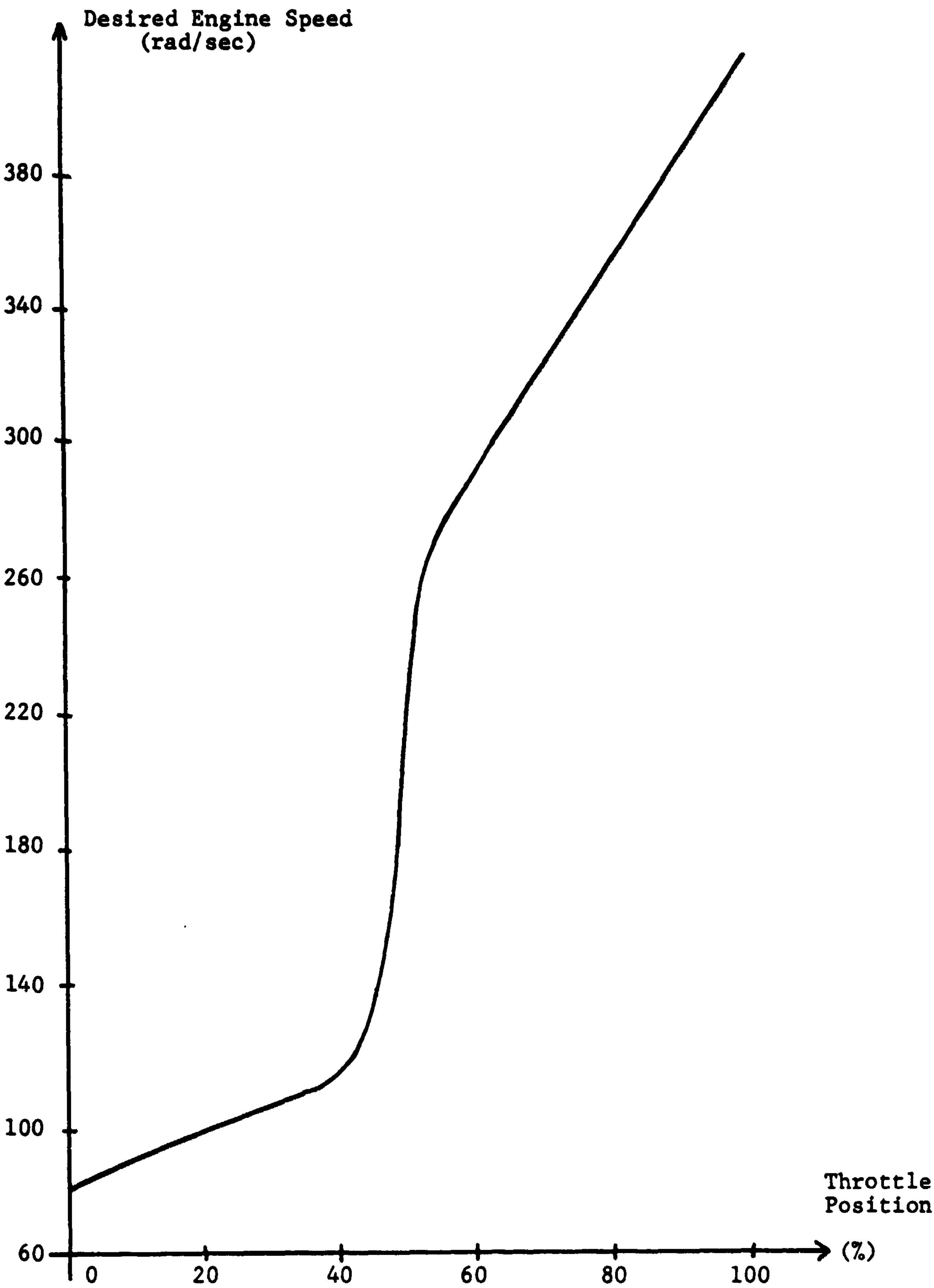

Figure 3.1 Minimum Fuel Curve

this study was to explore the feasibility of controlling the vehicle system in this manner. A block diagram of the control approach is shown in Figure 3.2.

3.2. Control System Design

In designing a control system to accomplish the task described above, an emphasis was placed on using simple, standard controllers which could be easily implemented. The use of linear control design methods for the design of these controllers was ruled out because of the highly nonlinear nature of the vehicle dynamics (see equations (2.2), (2.3), (2.6)). An additional nonlinearity is introduced by the minimum fuel curve which determines the engine speed set point for each throttle

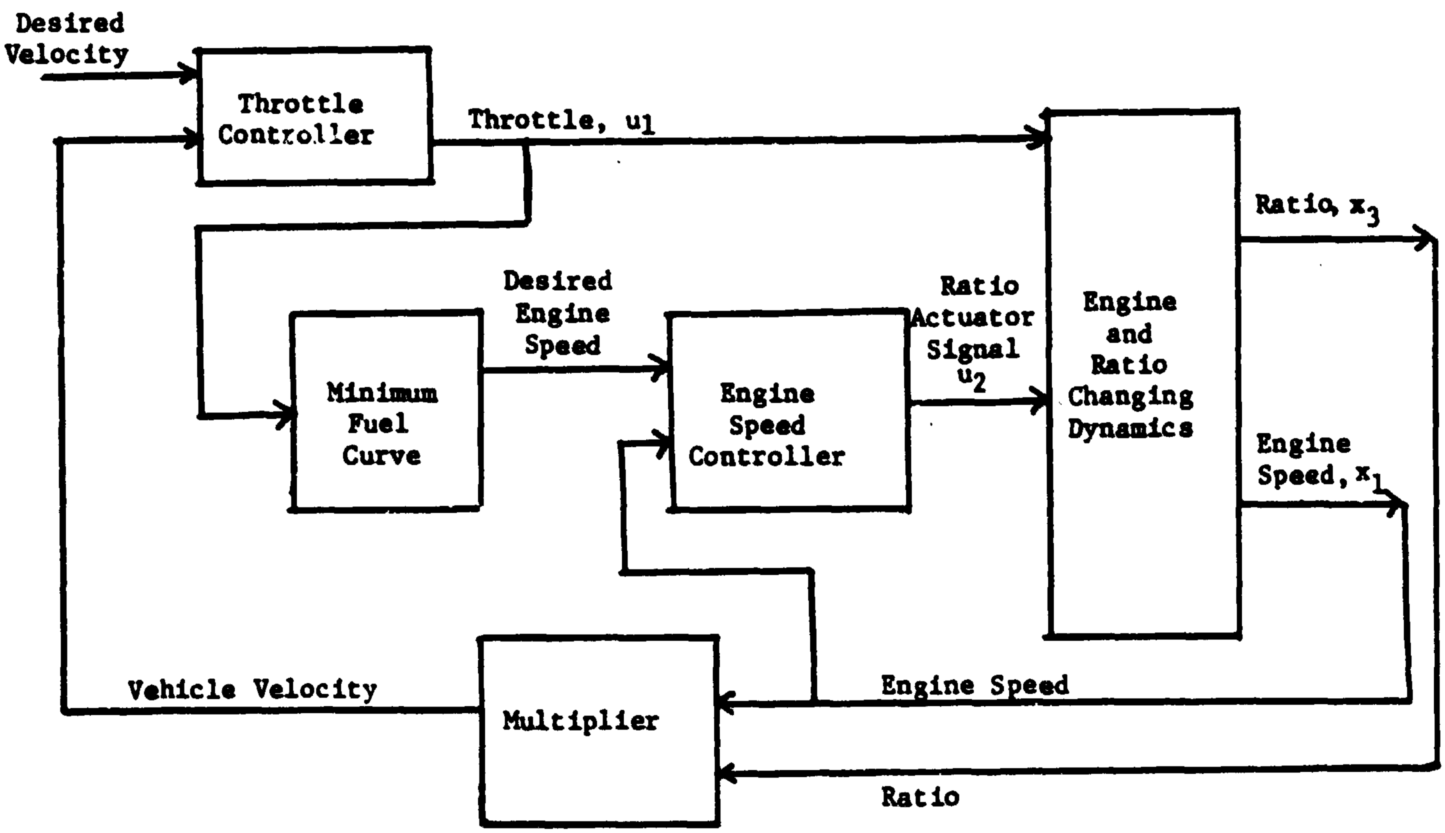

Figure 3.2 Vehicle Control Approach

setting (see Figures 3.1 and 3.2). This curve is a nonlinear function of throttle setting, thus introducing a significant nonlinearity in the vehicle velocity control loop.

An additional important factor in the design of the controllers is that they should use only easily measured system variables in a feedback configuration. For a given driving cycle, it is conceivable that using optimal control theory, open-loop trajectories for throttle setting and the ratio changing input could be generated to achieve minimum fuel consumption. However, they would not be practical to implement because of inaccuracies in the vehicle models used to generate them, and because in practice a vehicle would not be driven over the specific driving cycle they were designed for. Thus, some form of feedback control is required in an actual vehicle; hence the use of feedback control in this study.

The control system employes two primary control loops: one to control vehicle velocity by manipulating throttle, and the other to control engine speed by manipulating CVT ratio. This is shown in Figure 3.3. It is assumed that the operator of the vehicle generates a signal (by manipulating a pedal or some other device) which represents the desired vehicle velocity. This signal is the set point to the velocity control loop as shown in Figure 3.3. Based on the velocity error, the velocity controller generates a throttle setting and a braking torque to make the vehicle velocity equal the set point (idle throttle during braking and zero braking torque when throttle is above idle).

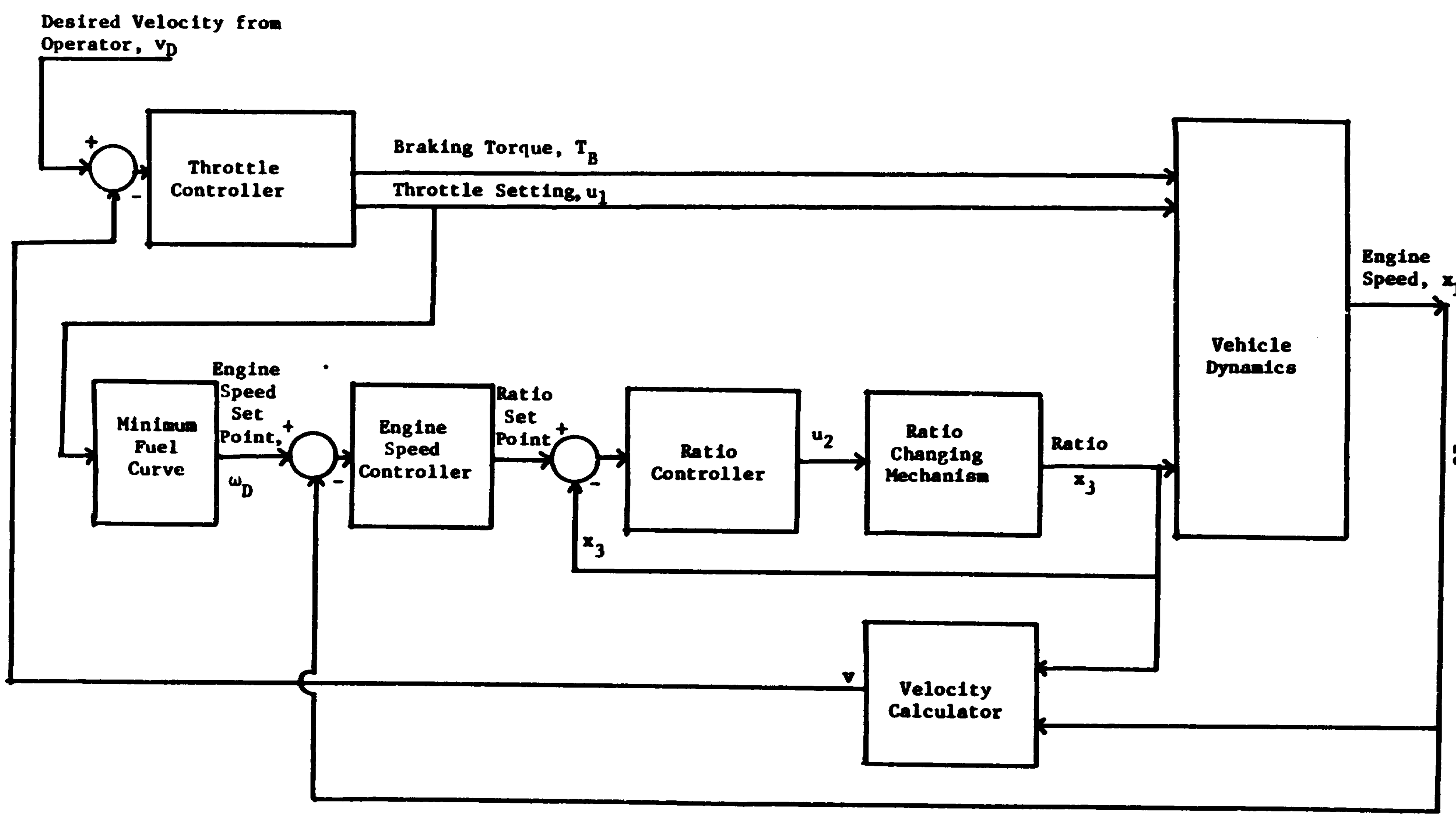

Figure 3.3 <u>Vehicle Control System - Initial Configuration</u>

For the given throttle setting, u_1, the minimum fuel curve generates an engine speed set point (for minimum fuel consumption) which is enforced by the engine speed controller manipulating the CVT ratio.

There are two possible ways of envisioning how the engine speed controller can change the CVT ratio. First, the output of the controller could be u_2, the variable which drives the ratio changing mechanism. Alternatively, the output of the engine speed controller could be a ratio set point which drives a ratio control loop; the ratio controller compares actual ratio to desired ratio and generates the input, u_2, to the ratio actuator (see Figure 3.3). This alternative allows the ratio controller to be selected to speed up, if necessary, a sluggish ratio changing mechanism. This second approach was the approach adopted in the present study.

The overall control system thus consists of three control loops: a velocity loop, an engine speed loop and a ratio loop. In addition, simulation runs showed that an improvement in control could be obtained by placing two low pass filters in the system; one to smooth out throttle changes and the other to smooth out changes in desired engine speed. The next three subsections describe the design of the controllers for each of the three control loops and the placement and design of the two low pass filters.

3.3 Vehicle Velocity Controller Design

The job of the vehicle velocity controller is to manipulate the throttle and brakes so that the vehicle velocity equals the velocity set point (driving cycle velocity). The input to the controller is the velocity error and, in keeping with the design objective of using simple, standard controllers, the controller is assumed to be a proportional plus integral (PI) controller. This is a natural choice because it represents how an actual operator would manipulate the throttle in response to a velocity error. In particular, an operator would depress the throttle in proportion to the velocity error; as the vehicle came up to speed and the error decreased, the operator would back off on the throttle, hence the need for the proportional term. An integral term is also needed to keep the throttle depressed at the value necessary to achieve zero steady state velocity error; without the integral term the throttle would approach idle position as the velocity error approached zero and a nonzero steady state velocity error would result.

Since the controller contains an integral term, and the actuator it is driving can limit (throttle has an upper limit of 100%), it is necessary to incorporate an antiwindup feature in the controller. The antiwindup feature essentially inhibits additional corrections to the throttle if the throttle is at its limit, and prevents the integrator in the PI controller from saturating or "winding up". The details of this feature are discussed in Appendix D.

Finally, later simulation runs revealed that rapid changes in the throttle at start-up (more rapid than an actual driver could produce) could cause instability in the vehicle velocity. It was found that smoothing of the throttle controller output reduces this possibility; hence a first order low pass filter was added at the output of the throttle controller.

A block diagram of the throttle controller is shown in Figure 3.4. The PI control algorithm, the antiwindup logic and the filter generate what might be considered a pedal position, P. A pedal position of K_1 (see blocks labeled throttle and brake limiter in Figure 3.4) corresponds to idle throttle. If the pedal is below K_1 the throttle limiter sets the throttle at idle position and the brake limiter generates a braking torque proportional to pedal position. For a pedal position between K_1 and 100 the braking torque is zero and the throttle equals the pedal position. For a pedal position above 100, the throttle is limited to 100%. The braking torque and throttle position are inputs to the vehicle dynamic simulation and the throttle setting u_1 is also an input to the minimum fuel curve (see Figure 3.3). This curve generates a set point for the engine speed control loop such that the engine runs at its minimum fuel consumption speed for the particular throttle setting.

Referring to Figure 3.4, the transfer function for the PI algorithm can be expressed as (ignoring the effects of the anti-windup logic).

$$\frac{M(s)}{V_e(s)} = \frac{K_{VP}s + K_{VI}}{s} \tag{3.1}$$

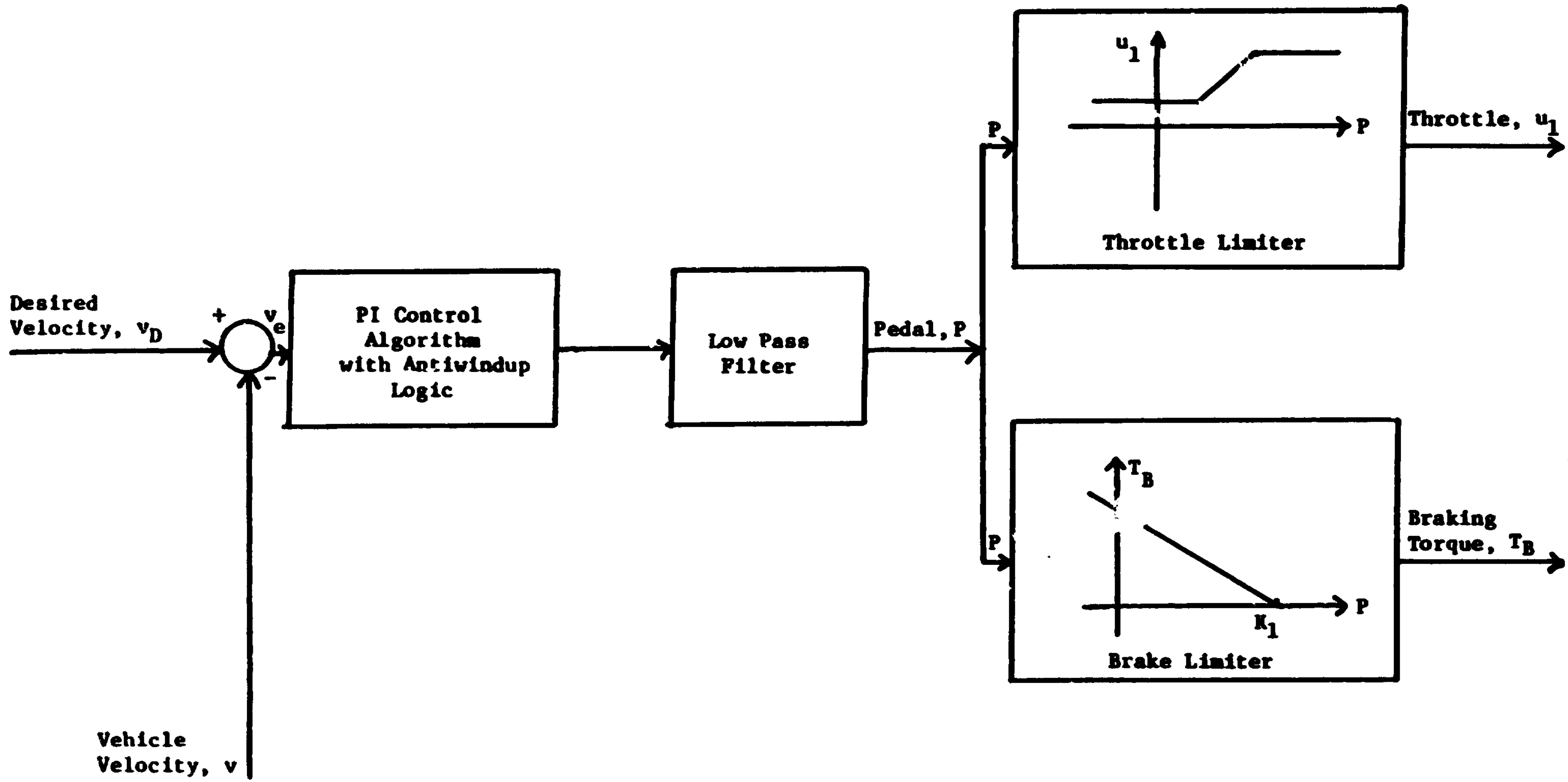

Figure 3.4 Velocity Controller

where K_{VP} and K_{VI} are the proportional and integral gains, respectively. If we define the controller state as (x_5 is used in conjunction with the engine speed controller)

$$x_6 = m - K_{VP}(v_D - v) \tag{3.2}$$

Then using (2.6) the controller state and output equations can be expressed as

$$\dot{x}_6 = K_{VI}(v_D - R_A R_W x_1 x_3) \tag{3.3}$$

and

$$m = x_6 + K_{VP}(v_D - R_A R_W x_1 x_3) \tag{3.4}$$

The transfer function of the throttle filter can be written as

$$\frac{P(s)}{M(s)} = \frac{1}{\tau_T s+1} \tag{3.5}$$

where τ_T is the filter time constant. If we define the filter state as

$$x_7 = P \tag{3.6}$$

then using (3.4) and (3.5) the filter state equation is

$$\dot{x}_7 = \frac{1}{\tau_T}(x_6 - x_7) + \frac{K_{VP}}{\tau_T}(v_D - R_A R_W x_1 x_3) \tag{3.7}$$

Equations (3.3), (3.6) and (3.7) describe the overall dynamic behavior of the throttle controller. It is important to note, however, that these equations only apply to the case where the actuator is not at a limit and the antiwindup logic has no effect on the controller output.

3.4 Engine Speed Controller Design

The job of the engine speed controller is to change the CVT ratio so that for a given throttle setting, the engine runs at its minimum fuel consumption speed. This is shown in Figure 3.5. The control algorithm was chosen to be a PI algorithm (the integral term is used to produce zero steady state error to a constant desired engine speed) with an antiwindup feature (the ratio has a lower limit of zero and the controller contains an integral term. See Section 3.2 and Appendix D). As discussed in Section 3.1, the output of the engine speed controller is a ratio set point R_D, which is implemented by the ratio control loop.

Simulation studies showed that the nonlinearity of the minimum fuel curve could produce instabilities in the control system. In particular, the curve (Figure 3.1) changes to a steep slope (large gain) for throttle settings above 45%; it was found that when the throttle made the transition from the low to high gain portions of the curve, the vehicle velocity could begin to oscillate. This instability was due, in part, to a rapid change in the engine speed set point. To slow down this rapid change, a low pass filter was placed immediately after the minimum fuel curve as shown in Figure 3.5. With the addition of this filter, the engine speed controller does not control to exactly the minimum fuel engine speed. However, it is unrealistic to expect the engine speed controller to follow rapid variations in desired engine speed; what is desirable is to track more gradual changes in optimum engine speed. Hence, the control loop controls to ω_A rather than ω_D.

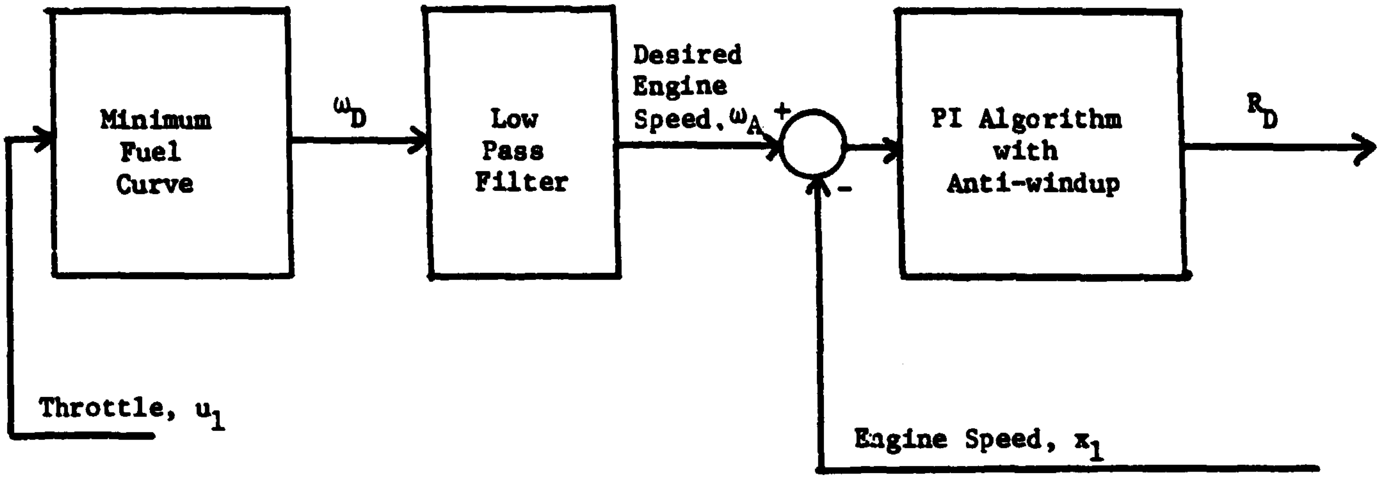

Figure 3.5 Engine Speed Controller

The transfer function for the PI algorithm can be expressed as (ignoring the effects of the antiwindup logic)

$$\frac{R_D(s)}{\omega_A(s) - x_1(s)} = \frac{K_{EP}s + K_{EI}}{s} \tag{3.8}$$

where K_{EP} and K_{EI} are the proportional and integral gains, respectively. If we define the controller state as

$$x_5 = R_D - K_{EP}(\omega_D - x_1) \tag{3.9}$$

then the controller state and output equations can be expressed as

$$\dot{x}_5 = K_{EI}(\omega_A - x_1) \tag{3.10}$$

and

$$R_D = x_5 + K_{EP}(\omega_A - x_1) \tag{3.11}$$

The actual output R_D is then modified by the antiwindup logic to produce the final desired ratio. The low pass filter transfer function is given as

$$\frac{\omega_A(s)}{\omega_D(s)} = \frac{1}{\tau_E s+1} \tag{3.12}$$

where τ_E is the filter time constant and ω_D is the minimum fuel engine speed corresponding to the current throttle setting. If we define the filter state as

$$x_8 = \omega_A \tag{3.13}$$

then from (3.12)

$$\dot{x}_8 = -\frac{1}{\tau_E} x_8 + \frac{1}{\tau_E} \omega_D(u_1) \tag{3.14}$$

where the dependence of ω_D on u_1 is shown explicity. Using (3.13) in (3.10) and (3.11) we have

$$\dot{x}_5 = K_{EI}(x_8 - x_1) \tag{3.15}$$

$$R_D = x_5 + K_{EP}(x_8 - x_1) \tag{3.16}$$

The state and output equations for the engine speed controller are thus given by (3.14), (3.15) and (3.16).

3.5 Ratio Control Loop

The ratio control loop is shown in Figure 3.6. In designing the ratio controller it is important to note that it is not necessary for the actual ratio, x_3, to equal the desired ratio,

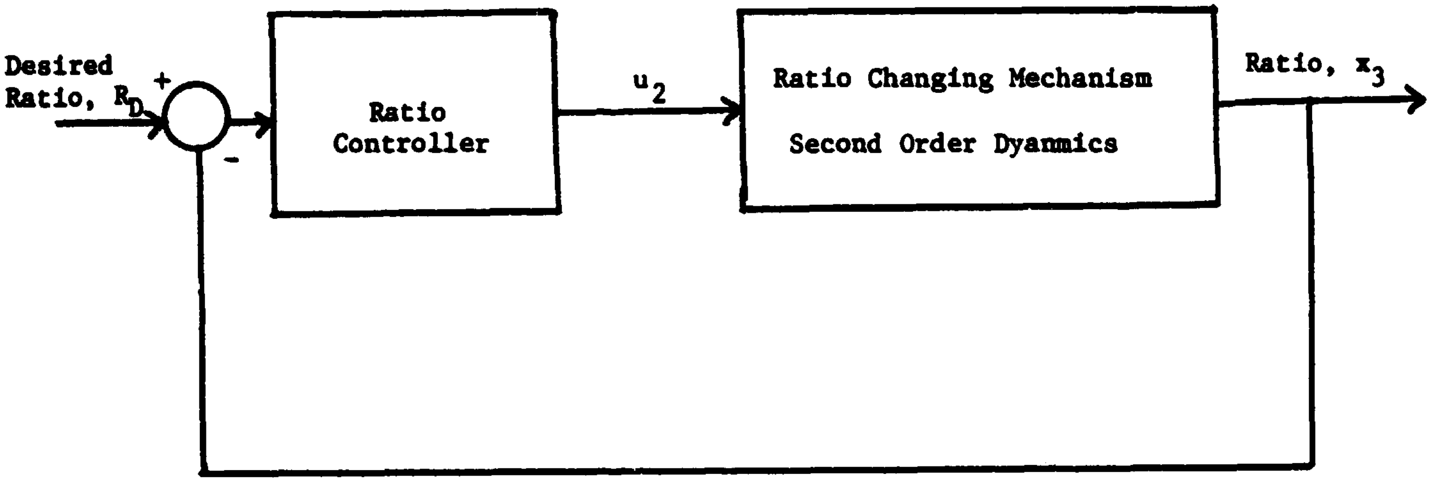

Figure 3.6 Ratio Control Loop

R_D, in steady state. The ratio is being changed so that the engine speed equals the engine speed set point; the actual, final value of ratio which achieves this is unimportant. Hence the ratio controller need not contain an integral term for zero steady state error. This controller is therefore chosen to be a proportional controller only. Thus, from Figure 3.6,

$$u_2 = K_{RP}(R_D - x_3) \tag{3.17}$$

where K_{RP} is the proportional gain of the ratio controller. Using (3.16) in (3.17) we have

$$u_2 = K_{RP}\ [x_5 + K_{EP}(x_8 - x_1) - x_3] \tag{3.18}$$

The second order dynamics relating x_3 and u_1 are given by (2.4) and (2.5).

The value of K_{RP} can be chosen to speed up the second order response of the ratio changing mechanism. In fact, this is the only reason for using a ratio control loop rather than having the engine speed controller generate u_2 and directly drive the ratio changing mechanism. It is important to note, however, that with a proportional ratio controller there is a second order system between the output of the engine speed controller and x_3, independent of whether the engine speed controller generates a set point for a ratio loop or generates the actuator input u_2 directly.

3.6 Summary of Closed-Loop State Equations

Since the closed loop-system state equations are spread throughout Section 2.0 and this section, it is useful to summarize them here for easy reference. First, the closed-loop system states are defined as

x_1 = engine speed.

x_2 = developed engine torque

x_3 = CVT ratio.

$x_4 = \dot{x}_3$

x_5, x_8 = states characterizing throttle controller.

x_6, x_7 = states characterizing engine speed controller.

The states x_5 through x_8 are not easily associated with physical variables because both controllers contain numerator dynamics.

The state equations are listed below using the equation number from previous sections.

$$\dot{x}_1 = \frac{x_2 - M_T x_3 (\phi_1(x_1, x_3) + \phi_2 + R_A R_W x_1 x_4)}{J_E + M_T R_A R_W x_3^2} \qquad (2.2)$$

$$\dot{x}_2 = - [x_2 - \phi_3(x_1, u_1)] / \tau_L \qquad (2.3)$$

$$\dot{x}_3 = x_4 \qquad (2.4)$$

$$\dot{x}_4 = C_2 x_3 + C_1 x_4 + C_3 u_2 \qquad (2.5)$$

$$\dot{x}_5 = K_{EI}(x_8 - x_1) \qquad (3.15)$$

$$\dot{x}_6 = K_{VI}(v_D - R_A R_W x_1 x_3) \qquad (3.3)$$

$$\dot{x}_7 = \frac{1}{\tau_T}(x_6 - x_7) + \frac{K_{VP}}{\tau_T}(v_D - R_A R_W x_1 x_3) \qquad (3.7)$$

$$\dot{x}_8 = \frac{1}{\tau_E} [-x_8 + \omega_D(u_1)] \qquad (3.14)$$

where v_D is the driving cycle velocity, $\omega_D(u_1)$ is the minimum fuel engine speed as a function of throttle setting (minimum fuel curve), u_1 is the throttle input, u_2 is the ratio actuator input, ϕ_3 is the steady state torque from the engine map, ϕ_1 reflects the effects of road load losses on engine speed and ϕ_2 reflects the effects of braking torque and grade on engine speed. The rest of the entries in the above equations are constants. Reference should be made to the appropriate prior section (determined by the equation number) for more details. The control inputs u_1 and u_2 are generated from the states by

$$u_1 = \begin{cases} 100, & x_7 \geq 100 \\ x_7, & K_1 \leq x_7 < 100 \\ K_1, & x_7 < K_1 \end{cases}$$

and

$$u_2 = K_{RP} \left[x_5 + K_{EP}(x_8 - x_1) - x_3\right] \qquad (3.18)$$

Here, K_1 represents idle throttle (see Figure 3.4).

Having described the control approach and the details of the control system, we conclude this section with a brief discussion of how the control system might be implemented in an actual vehicle.

3.7 Control System Implementation

The control system is envisioned as being implemented with an on-board microcomputer-based system. The control algorithms discussed in the previous subsections are easily programmed on such a system, and the measurements required for control are all easily generated from standard sensors.

A block diagram of the control system showing the operator, control computer and vehicle is given in Figure 3.7. To implement the control algorithms, the control computer requires measurements of vehicle velocity, engine speed and ratio. If the ratio is not measurable directly from the CVT, it can be calculated from engine speed and vehicle velocity (see Equation (2.6)). The control algorithms implemented by the computer are discrete time versions of equations (3.3), (3.7), (3.14), (3.15) and (3.18) (See Section 4.1 for the sampling time) which are given at the end of Section 3.6. These algorithms are easily implemented and do not require the use of component maps and interpolation.

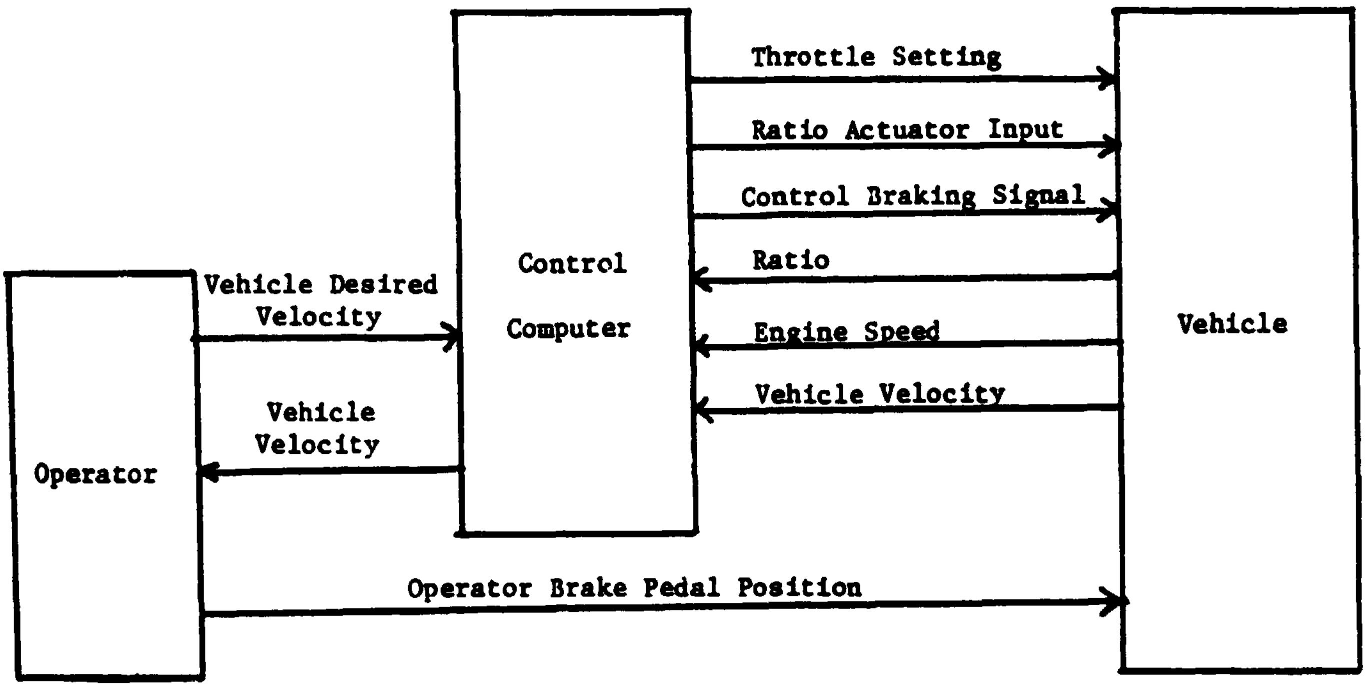

Figure 3.7 Computer Implemented Control System

It is assumed that the operator manipulates some sort of device (e.g. a pedal) to indicate desired velocity. Based on this demand, the control computer generates a throttle setting and a signal to change the ratio so that the desired velocity is achieved and the engine is running most efficiently. The computer also displays vehicle velocity to the operator. The brake pedal is assumed to be under the control of the operator. However, it is conceivable that the control computer could also generate braking actions to make the vehicle achieve a desired velocity. In fact, this approach was adopted in this study (see Figure 3.5). Having described the control system and the vehicle dynamics in this and the previous section, respectively, we now turn to a description of how the overall system was simulated in order to tune the controllers and study the dynamic behavior of the overall closed-loop system.

4.0 SYSTEM SIMULATION

The performance of the closed-loop system was evaluated using a digital computer simulation of the vehicle dynamics and the control algorithms. The vehicle was made to follow a specific velocity vs. time profile or driving cycle (SAE J227a, schedule D (modified)) as shown in Figure 4.1. The driving cycle velocity provided the command input to the throttle controller. In effect, the simulation solved the eight state equations describing the vehicle and controller dynamics for given initial conditions and the given driving cycle. The equations actually solved are given in Section 3.6; equations (2.2) through (2.5) describe the vehicle dynamics, while equations (3.3), (3.7), (3.14), (3.15) and (3.18) describe the controller dynamics and outputs. The simulation results are trajectories of vehicle and controller states and outputs (as the vehicle traverses the driving cycle) as well as vehicle energy consumption.

In the actual simulation, two different time scales were used for integrating the vehicle and controller state equations. The controller state equations were approximated with discrete time equivalents with a basic sampling period or time step size of 0.1 seconds. Hence the control inputs to the vehicle dynamics were updated every 0.1 seconds and held constant between updates. This was felt to be quite acceptable because it simulated what an on-board computer would actually do; the computer would generate piecewise constant controls updated

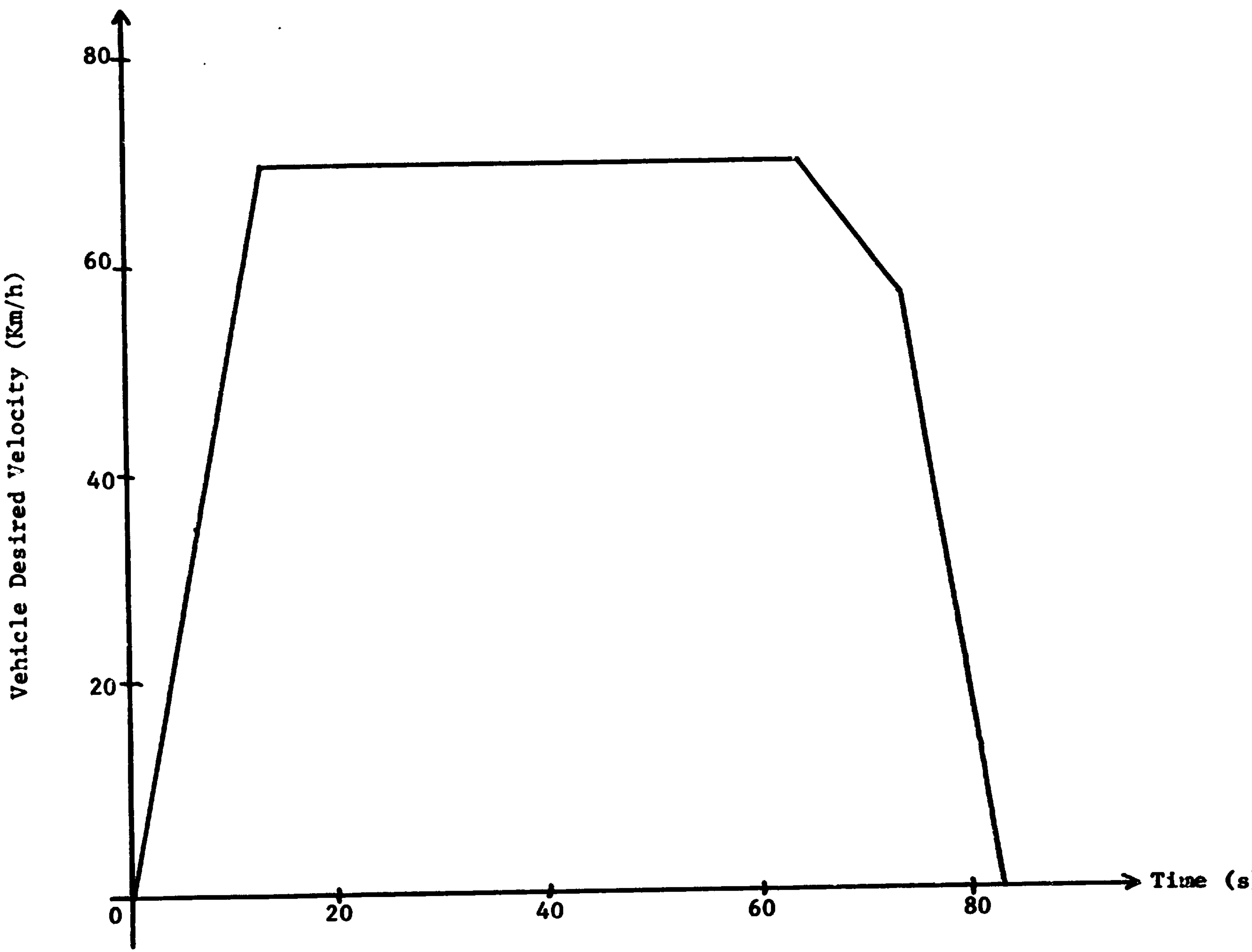

Figure 4.1 Driving Cycle SAE J227a, Schedule D (modified)

only at sampling instances. On the other hand, the vehicle state equations were integrated using a variable step size Runge-Kutta integration scheme which produced an average step size which was orders of magnitude smaller than 0.1 second. This, of course, makes sense since the vehicle is a continuous system and not piecewise constant as are the controller outputs.

The simulation program was organized into several Fortran subroutines whose primary purposes are listed in Table 4.1. Subroutines PHI3, PHI4 and PHI6 use steady-state engine characteristics, in the form of table or "maps", and interpolate as necessary to yield the desired results. A detailed listing of the simulation program is given in Appendix C.

Subroutine	Purpose
MAIN	Inputs data, coordinates simulation of system for driving cycle, computes control inputs, outputs results.
NIN	Numerically integrates state equations using 4th order Runge-Kutta method.
RHS	Computes right hand side of state equations for NIN.
PHI1	Computes drag and rolling resistance.
PHI2	Computes braking torque and effect of grade angle on vehicle.
PHI3	Computes steady state engine torque for given throttle and engine speed.
PHI4	Computes most fuel-efficient steady state engine speed for given throttle position.
PHI5	Limits the CVT ratio within pre-defined bounds.
PHI6	Computes steady state fuel rate for given engine speed and throttle.
VSP	Calculates driving cycle velocity for any time.

Table 4.1 Simulation Program Subroutines

5.0 SIMULATION RESULTS

The overall objective of the simulation was to study the feasibility of the control approach described in the preceeding sections. In particular, the simulation revealed the degree to which the controllers described in Section 3.0 can control the propulsion system, the ease or difficulty of tuning the controllers, and the nature of the resulting dynamic behavior of the vehicle. We consider these results in the subsections below.

5.1 Controller Tuning

The controllers were tuned so that the vehicle satisfactorily followed the specific driving cycle shown in Figure 4.1. The driving cycle consisted of an acceleration phase followed by a cruise phase and, finally, a braking phase. Two different (but constant) decelerations were assumed for the braking phase. The driving cycle served as a velocity command input to the propulsion control system.

Three controllers required tuning: the vehicle velocity or throttle controller, the engine speed controller and the ratio controller. In addition, the time constants of the two smoothing filters (one after the minimum fuel curve and one after the velocity PI controller) had to be selected. Specifically, it was necessary to select values for the proportional and integral gains, K_{VP} and K_{VI}, respectively, of the throttle controller, the proportional and integral gains, K_{EP} and K_{EI}, respectively, of the engine speed controller, the proportional gain, K_{RP}, of the ratio controller, and the two filter time constants, τ_E and τ_T.

Since the ratio control loop (see Figure 3.6) consists of the proportional control of a known, second order system, the gain, K_{RP}, was determined analytically; the particular value was chosen so that the ratio loop responded as a critically damped second order system to a step change in the ratio set point. Furthermore, it was found from repeated runs of the simulation that a time constant of 1.0 seconds for both filters provided adequate smoothing.

Generally speaking, an iterative process was used to select the gains for the throttle and engine speed PI controllers. The throttle controller was first tuned to obtain as close a match as possible between driving cycle velocity and vehicle actual velocity ("close" was measured by a combination of velocity mean square error and visual examination of the velocity profile). Then the engine speed controller was adjusted to obtain as close a match as possible between actual and desired engine speed ("close" measured by mean square error). Each time any of the four gains was changed, a resimulation of the complete driving cycle was required to evaluate the effects of the gain change. The process was repeated until no significant overall improvement in either the velocity error or engine speed error was obtained; although time consuming, the procedure did converge to a set of gain values which gave reasonably good dynamic response of the vehicle. The specific values for the controller gains are given in Appendix B. An example of responses which are representative of the best responses are given in Figures 5.1 through 5.4. These are discussed in detail in the next subsection.

5.2 Closed Loop Dynamic Response of the Vehicle

The dynamic responses of interest for the closed loop system are those of vehicle velocity, CVT ratio, engine speed and throttle setting.

Figures 5.1 through 5.4 show examples of the best responses obtained. In interpreting these curves, it should be remembered that for a given vehicle velocity, an increase in ratio is required to produce a decrease in engine speed. The basic response of the vehicle is summarized as follows: Initally the vehicle is at rest with an idle throttle, an idle engine speed and a zero ratio. As the driving cycle starts, the velocity error causes the throttle to increase. This in turn causes both the engine speed and its set point (via the minimum fuel curve) to increase. The engine speed is actually above the set point, so the ratio increases to bring this speed down. At about 45% throttle there is a rapid increase in engine speed set point (due to the nonlinear minimum fuel curve) which causes the ratio to drop to bring up the engine speed. This type of dynam ~ behavior continues (with the ratio slowly increasing again as the engine speed overshoots the set point) until the cruise part of the driving cycle is reached. At this point the

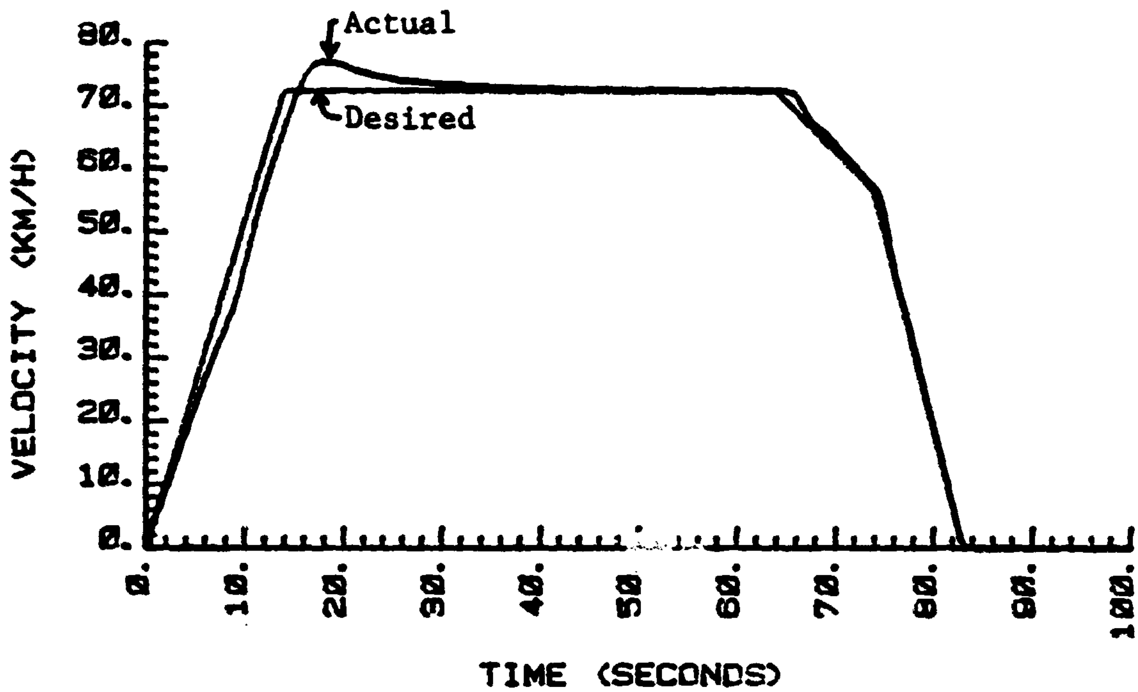

Figure 5.1 Vehicle Velocity

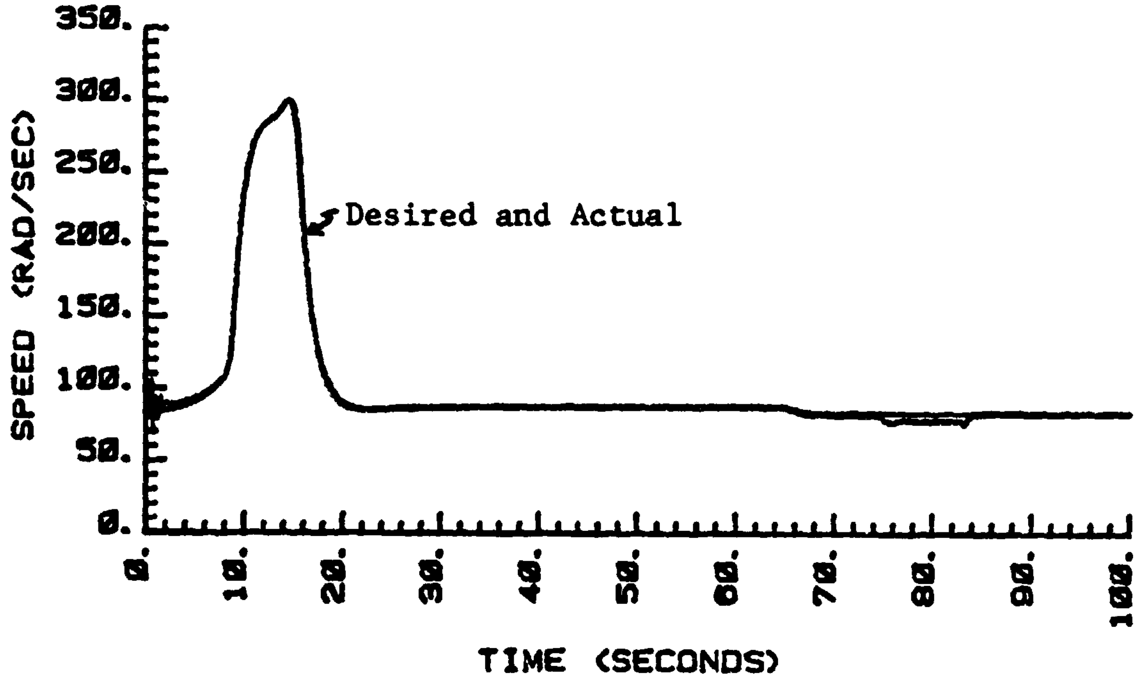

Figure 5.2 Engine Speed

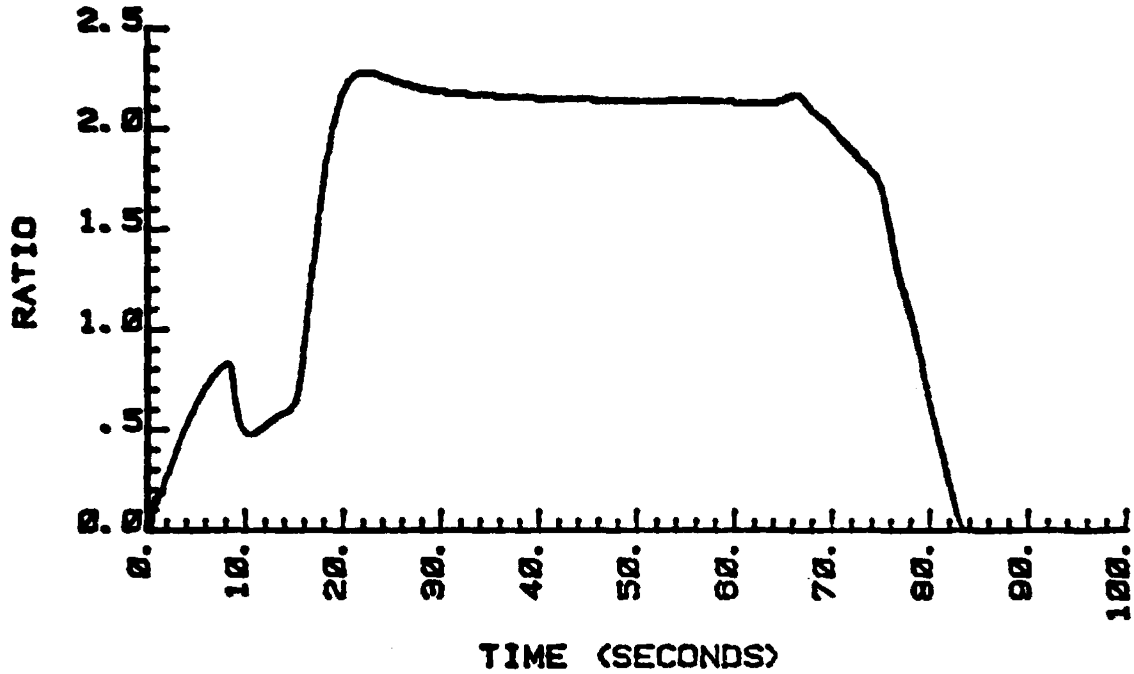

Figure 5.3 <u>CVT Ratio</u>

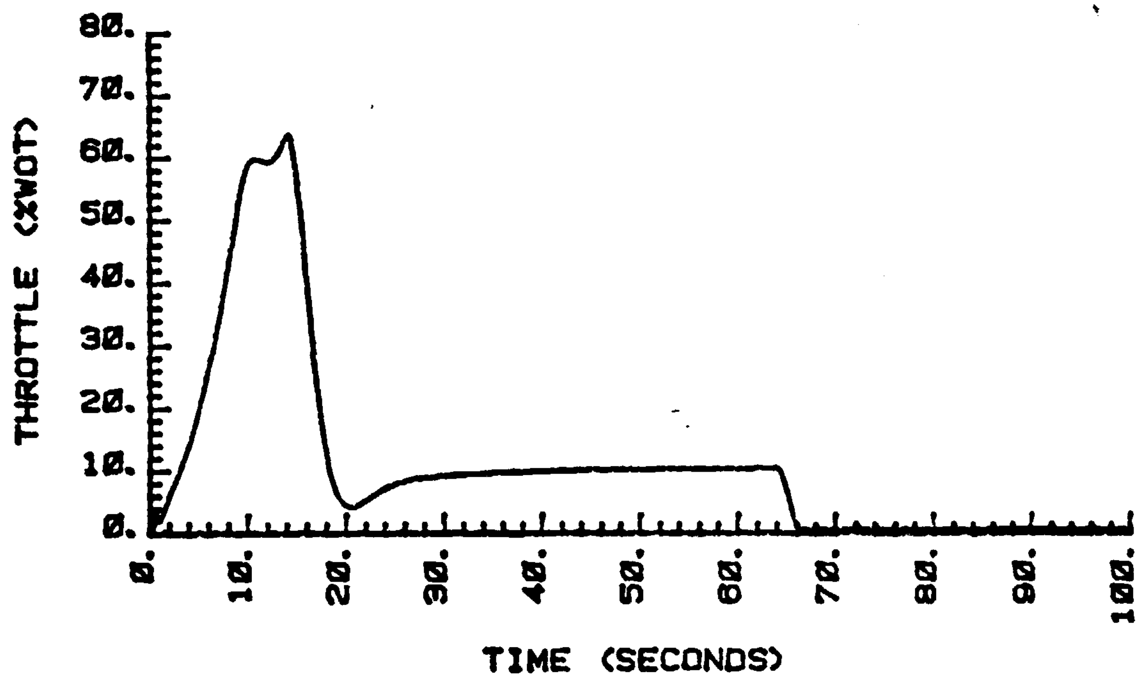

Figure 5.4 <u>Throttle</u>

velocity error rapidly decreases causing the throttle to drop off sharply. This, in turn, causes a rapid drop in desired engine speed via the minimum fuel curve to bring the engine speed down the ratio therefore increases to about 2.3. When the cruise period is complete, it is necessary to apply various amounts of braking torque to make the vehicle follow the driving cycle from cruise down to a stopped condition. During this time, the throttle is constant at idle position and the engine speed makes small excursions below the idle speed desired. Finally, the vehicle comes to a stop as a result of sufficient braking torque and the ratio approaching zero.

The responses demonstrate that the control system is capable of making the vehicle follow the driving cycle velocity profile while at the same time keeping the engine at the minimum fuel consumption engine speed; Figure 5.2 shows that there is very little error between desired and actual engine speed. Although the vehicle's actual velocity is quite close to the driving cycle velocity, there is some error during the acceleration phase and at the start of the cruise phase. To a large extent, this is due to the classical tradeoff in control system design between tracking error and transient response. If the velocity controller gain is increased so that the vehicle tracks the velocity ramp in the acceleration phase more closely, the transient response will be made more oscillatory and there will be a larger overshoot when the cruise phase is reached. In addition, as discussed later, increasing this gain to reduce the tracking error can also introduce instabilities when the vehicle first begins to accelerate.

In tuning the controllers to produce the above responses a number of observations were made concerning the sensitivity of the responses to controller gain values, and instabilities and unexpected dynamic behavior of the closed loop system. These observations are given below.

With regard to controller tuning, it was found that the responses of the vehicle variables were rather sensitive to choices for the controller gain values. This was due in large part to the nonlinearities of the vehicle and, particularly, the nonlinear minimum fuel curve. Examination of Figure 3.1 shows that the minimum fuel engine speed is a highly nonlinear function of throttle setting; examination of Figure 3.3 reveals that this nonlinearity acts as a nonlinear gain in the loop which is controlling engine speed by manipulating the ratio. The presence of a nonlinear gain makes it particularly difficult to come up with one set of controller gains which will give good transient response over a wide range of vehicle operation. If the controller gains are selected to give good transient response for throttle settings below 45% (the low gain region on the curve), then during acceleration when the throttle moves into the region above 45% (high gain region) the loop gain goes up markedly and the response will tend to be oscillatory. In fact, complete instability of all four variables during the acceleration phase of the drive cycle was often observed in the process of tuning the controllers. The final method used to deal with this nonlinear gain was a worst case approach: choose controller gains so that good performance was achieved in the high gain region of the minimum fuel curve; the closed-loop system would then be a little sluggish in the low gain region of the curve, but it would not produce oscillations in ratio, throttle and vehicle velocity.

Two other approaches to deal with this nonlinear gain which were considered but ultimately discarded are outlined below. First, with reference to Figure 3.3, a nonlinear gain was introduced at the output of the throttle controller; the output of this nonlinear gain was the throttle setting, u_1. The new nonlinearity was chosen to exactly compensate for the minimum fuel curve nonlinearity; in effect the gain in the engine speed control loop was made to be linear and the nonlinearity of the minimum fuel curve was shifted to the vehicle velocity control loop. It was felt that the performance of this latter loop might be less sensitive to the presence of a nonlinear gain. However, this did not turn out to be the case and the approach was abandoned. The second approach was similar to the first: the gains of the velocity PI controller were made to depend on the throttle position. In effect, the gains were decreased if the system was operating on the high gain portion of the minimum fuel curve. This approach exhibited instabilities as the system was passing from one gain region to the next, and so was also abandoned.

As mentioned above, an approach which gives satisfactory results is to tune the controller for the worst case where the system is operating on the steep part of the minimum fuel curve. Alternatively, consideration should be given to using an approximate, linear minimum fuel curve. It is not practical to control the engine speed to exactly the minimum fuel consumption engine speed for each throttle setting; the throttle is continually changing and there is always lag in the engine speed control loop. It would appear that the use of an approximate, linear minimum fuel curve will lead to both good fuel consumption characteristics and good dynamic response of the vehicle.

We now discuss the actual dynamic responses observed for the vehicle velocity, engine speed, ratio and throttle setting; instabilities and unusual dynamic responses observed during the simulation runs are discussed. We begin this discussion by first noting that the control loop on vehicle velocity and the one on engine speed are highly interacting. Throttle is used to control velocity, while ratio is used to control engine speed. This can be viewed as a two input/two output system as shown in Figure 5.5. The strong interaction between the loops comes about because a change in throttle (with ratio constant) will cause a change in both vehicle velocity and engine speed; similarly, a change in ratio (with throttle constant) will produce a change in both velocity and engine speed. Furthermore, the throttle setting (control input for the top loop Figure 5.5) determines the set point for the bottom loop. Hence a change in throttle forces a change in ratio. This strong and unusual type of interaction (the set point of one loop determined from the control input to another loop) makes this system very difficult to control.

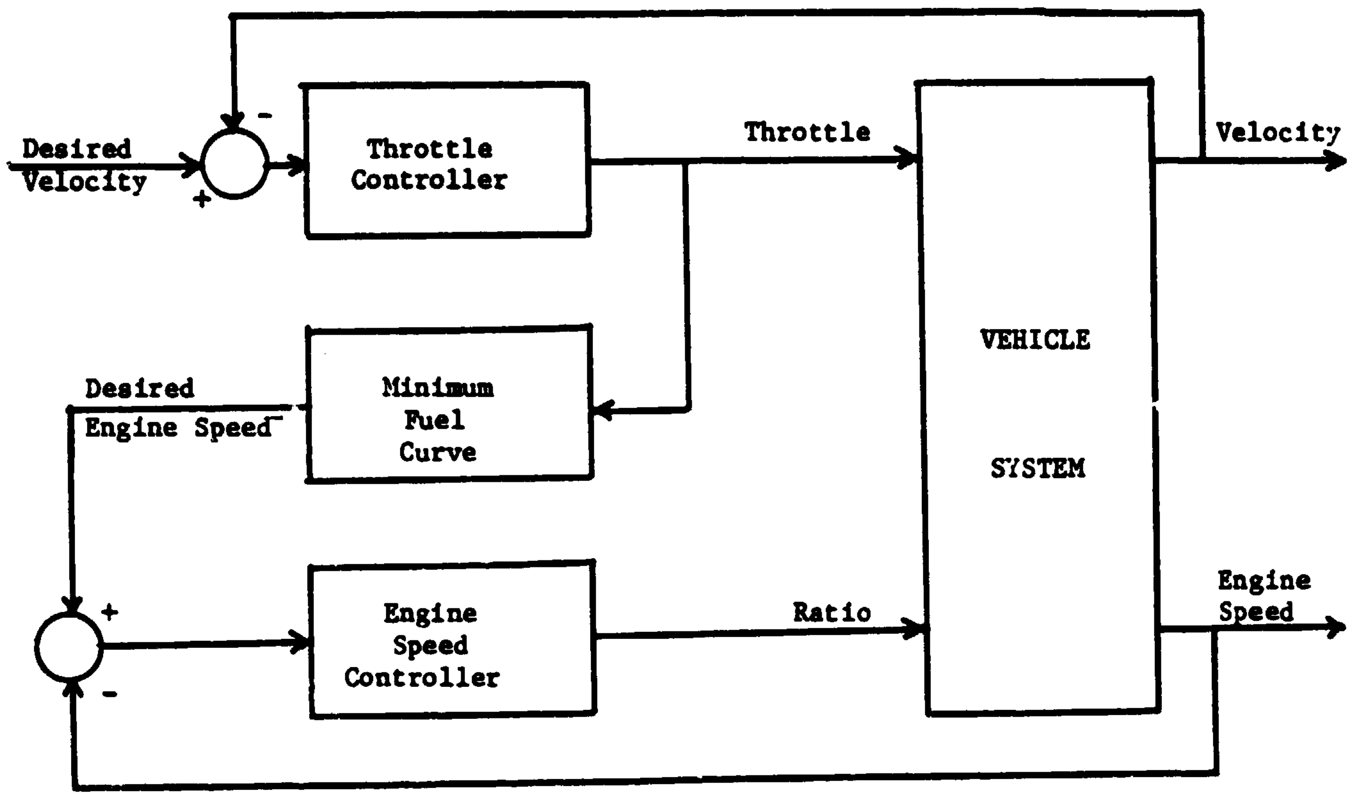

Figure 5.5 Two Input/Output Model of Control System

A typical instability in all four variables observed during the simulation runs was at start-up when the vehicle first begins to traverse the driving cycle (see Figure 5.5). The sequence of events is as follows. The desired velocity increases above zero, causing the throttle setting to increase above idle throttle. The vehicle will not move, however, until the ratio comes off zero. This will begin to occur in this instance because as the throttle setting increases, the desired engine speed also increases; this produces an engine speed error which causes the ratio to increase from zero. The instability begins to show up if either or both of the proportional gains of the controllers are too large. In this case, as the desired velocity initially increases there is a large throttle increase for the essentially unloaded engine (ratio is still close to zero). This causes the engine speed to increase rapidly and become much larger than the desired minimum fuel engine speed set point corresponding to the throttle setting. This large engine speed error, in turn, causes a large increase in the ratio in an attempt to bring the engine speed down. The increased ratio and large throttle setting cause the vehicle to accelerate rapidly, resulting in a vehicle velocity which is much greater than the desired velocity corresponding to the driving cycle. Also, the engine speed has now dropped below the set point due to the load placed on the engine. At this point the sequence of events is reversed: to counteract the large velocity error the throttle is set back to almost idle; the desired engine speed corresponding to this lower throttle is still above the actual engine speed, hence the ratio drops to almost zero. This causes the vehicle to come to a standstill, causing the throttle in turn to once again increase by a large amount. The cycle is then repeated.

The gains which cause the above instability are not unreasonable. For example, the instability was initially observed while attempting to tune the controllers by increasing the proportional gains so that the vehicle would more closely follow the ramp of desired velocity during the acceleration phase of the driving cycle. The start-up instability described above was the determining factor in how large the proportional gains could be made; keeping the gains low to avoid the instability produced a noticeable velocity error during the acceleration phase of the driving cycle (see Figure 5.1). This error could not be reduced by increasing the integral gain of the velocity controller because such an increase would produce too much velocity overshoot when the cruise phase of the driving cycle was reached (again, see Figure 5.1). The integral gain finally selected represented a tradeoff between velocity error during the acceleration phase and velocity overshoot when the cruise phase is reached.

A second, and quite dramatic, instability occurred during either the acceleration or cruise phase of the driving cycle. As with the start-up instability described above, this instability is a consequence of the pronounced interactions in the system, i.e. throttle changes and ratio changes each have a strong effect on both vehicle velocity and engine speed. If the controllers are not tuned properly, this interaction can cause the vehicle to actually speed up as that throttle is decreased during either the acceleration or cruise phase of the driving cycle. Responses of the system variables showing this instability are given in Figures 5.6 through 5.8. It must be remembered in examining these responses that they represent the case of improperly tuned controllers; they are included only because of the interacting instability they reveal.

Between about 15 seconds and 30 seconds the throttle is decreasing dramatically, yet the vehicle velocity continues to increase. Similarly, from 40 seconds to about 50 seconds the throttle increases while the vehicle velocity decreases. This phenomenon is due to the strong effect that both ratio and throttle have on vehicle velocity; the behavior from 15 to 30 seconds is explained as follows (the behavior from 40 to 50 seconds is explained by just the opposite reasoning). As the throttle is decreased to slow down the vehicle, the desired engine speed also decreases; this causes the ratio to increase in an attempt to bring the engine speed down. However, increasing the ratio also speeds up the vehicle, resulting in a further decrease in the throttle. If the ratio change has a greater effect on the vehicle velocity than the throttle change (which can occur in certain operating regions) then the above sequence of events will cause the vehicle to speed up as the throttle is decreased.

This type of behavior would, of course, be unacceptable to a human driver; less throttle means the vehicle must, within a short time, slow down, not speed up. In effect, what is occurring above is that the vehicle is being driven by the ratio, i.e. the ratio rather than the throttle is having the dominant

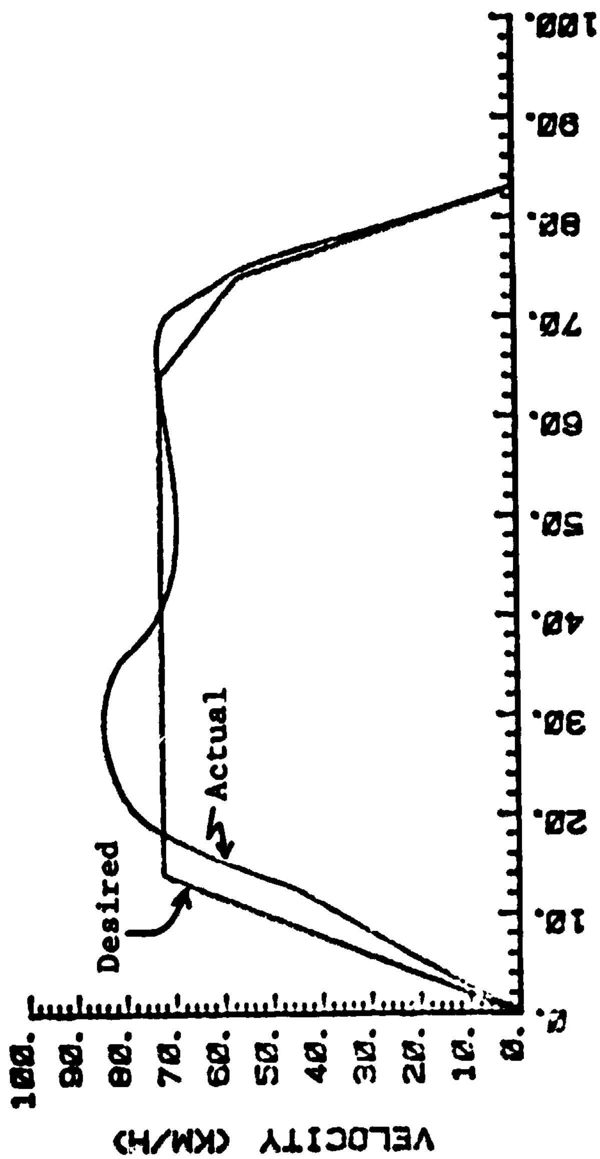

Figure 5.6 Vehicle Velocity for Improperly Tuned Controllers

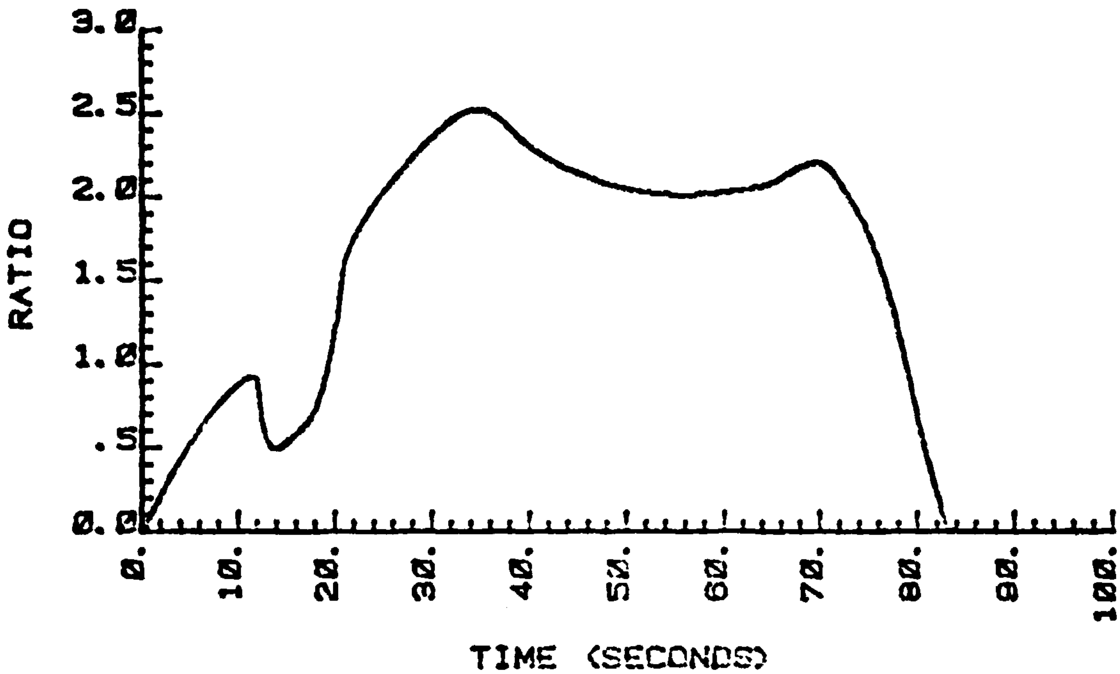

Figure 5.7 Ratio for Improperly Tuned Controllers

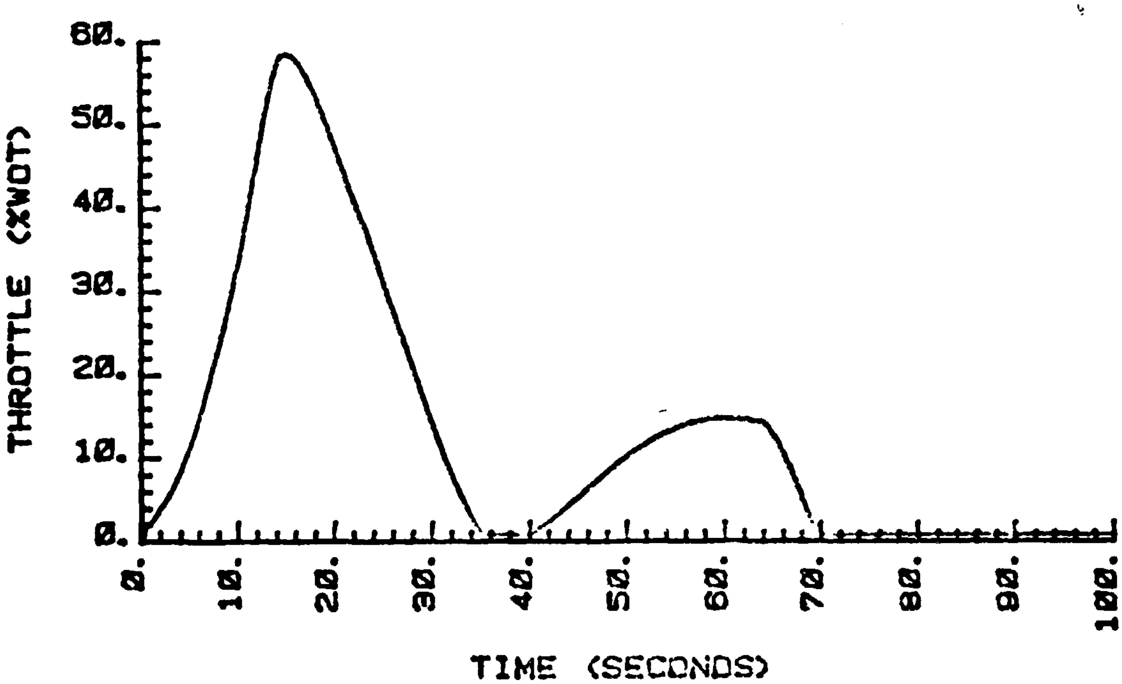

Figure 5.8 Throttle for Improperly Tuned Controllers

effect on vehicle velocity and it is being manipulated in a way to maintain the vehicle velocity at the driving cycle set point velocity. The problem is avoided by proper tuning of both controllers.

5.3 Conclusions

The experience gained from numerous simulations and from working with the controllers has led to a number of conclusions regarding the feasibility of the control philosophy and approach described in the preceeding pages of this report. To reiterate, the control philosophy is to minimize fuel consumption by manipulating throttle to achieve vehicle desired velocity and by manipulating ratio so that, for each throttle setting, the engine is operated at its minimum fuel consumption point. The simulation results demonstrate that it is feasible to control a vehicle in such a manner, and it is possible to do it with relatively simple controllers (PI controllers with anti-windup logic) which use easily measured system variables (CVT ratio, engine speed and vehicle velocity). Furthermore, there would be no difficulty in implementing the control system with an on-board microcomputer.

The simulation results also demonstrate that the vehicle nonlinearities and the minimum fuel curve nonlinearity can cause instabilities in vehicle operation both at start up and during the acceleration and cruise phases of the driving cycle. These instabilities can be avoided by proper tuning of the controllers. To simplify the job of tuning the controllers, and to minimize the possibility of producing an instability, consideration should be given to replacing the nonlinear minimum fuel curve with a straight line approximation, i.e. minimum fuel engine speed should be approximated by a linear function of throttle setting.

6.0 CONCLUSIONS AND FUTURE WORK

This project has demonstrated, for the first time, that it is possible to achieve satisfactory, stable control of all significant dynamic variables of a heat engine-CVT vehicle propulsion system by designing linear control loops for control of engine speed, throttle position and CVT ratio. The system simulation has illustrated the difficulty of controlling this nonlinear system which exhibits strong interaction between the input variables (CVT ratio and throttle) and output variables (engine speed and vehicle speed).

The feasibility of controlling the propulsion system so that the vehicle followed the modified SAE driving cycle was demonstrated by the simulation. Satisfactory dynamic responses of the variables were accompanied by a low average deviation of the engine speed from its desired, low fuel consumption value.

The control algorithms developed were in a simple, linear form which could easily be implemented on-board a vehicle using a microcomputer.

Future work could be concerned with adding a flywheel for energy storage to the system as investigated to date, and then to investigate the resultant control problems. Before such an investigation could be started, it would be necessary to define decision rules or strategies for using two sources of on-board energy, the engine and the flywheel. Thus, a prime candidate for near term future work is an investigation of power flow strategies for a vehicle with two (or more) on-board power sources. This should be followed by investigations into implementing these strategies using control systems.

7.0 REFERENCES

1. Dunn, H.S. and Wojciechowski, P.H. "Energy Storage and Conversion Efficiency in a Hydraulic/Gas-Turbine Hybrid," ASME Paper No. 74-GT-107, March 1974.

2. Orshansky, E., Huntley, P. and Weseloh, W. "Automobile Fuel Economy with Hydromechanical Transmission by Simulation Studies," SAE Paper No. 740308, February, 1974.

3. Prabhaker, R., Citron, S.J. and Goodson, R.E., "Optimization of Automobile Engine Fuel Economy and Emissions," Journal of Dynamic Systems, Measurement and Control, ASME Transactions, Series G, Vol. 99, No. 2, June 1977.

4. Reid, K.N. and Woods, R.L., "Fluid Control of a Hydrostatic Transmission in a Vehicle Propulsion System," Journal of Dynamic Systems, Measurement and Control, ASME Transactions, Series G, Vol. 95, No. 2, June 1973.

5. Sweet, L.M. and Anhalt, D.A."Optimal Control of Flywheel Hybrid Transmissions," Journal of Dynamic Systems, Measurement and Control, ASME Transactions, Series G, Vol. 100, No. 1, March 1978.

6. Christenson, B.C., Frank, A.A. and Beachley, N.H., "The Fuel-Saving Potential of Cars with Continuously Variable Transmissions and an Optimal Control Algorithm," ASME Paper No. 75-WA/Aut-20, August, 1975.

7. Chang, M.C., "Computer Simulation of an Advanced Hybrid Electric-Powered Vehicle," SAE Paper No. 780217, March, 1978.

8. Mechanical Technology Inc. and Bendix Corporation, "Feasibility Analysis of the Transmission for a Flywheel/ Heat Engine Hybrid Propulsion System," NTIS Report No. PB-212-097, Springfield, Virginia, November, 1971.

9. Beachley, N.H. and Frank, A.A., "Increased Fuel Economy in Transportation Systems by Use of Energy Management-Second Year's Program," Department of Transportation Report No. DOT-TST-76-57, Washington, D.C., December, 1975.

10. Burghart, J.H. and Donoghue, J.F., "Control of a Vehicle with a Continuously Variable Transmission," Proceedings of the 23rd Midwest Symposium on Circuits and Systems, August, 1980.

APPENDIX A

VEHICLE MODEL

Section 2.0 of the report presents an overview of the methods used to model the vehicle and the resulting state equations which describe the vehicle dynamics. This Appendix is devoted to a detailed presentation of this material. In developing the vehicle model, the following variables will be used (Appendix B contains a complete list of all variables and parameters used in the vehicle model):

D = vehicle drag force.
J_{EQ} = equivalent load inertia as seen by the CVT.
R = vehicle rolling resistance force.
R_T = CVT ratio.
T_A = torque required at rear axle.
T_B = braking torque.
T_E = actual torque developed by the engine.
T_L = equivalent load torque as seen by the CVT.
T_m = steady state engine torque.
T_1 = load torque seen by the engine.
u_1 = throttle setting.
u_2 = ratio actuator input.
v = vehicle velocity
v_w = wind velocity
x_1 = vehicle first state ($= \omega_E$).
x_2 = vehicle second state ($= T_E$).
x_3 = vehicle third state ($= R_T$).
x_4 = vehicle fourth state ($= \dot{R}_T$).

β = grade angle.

ω_E = engine speed.

ω_D = drive shaft speed.

The drive train of the vehicle is shown in Figure A1 and consists of a heat engine (HE), a continuously variable transmission (CVT), and a differential to transmit power from the drive shaft to the rear axle and wheels. Before deriving the equations which govern the overall dynamics of the vehicle, we first present the methods used to model each of the individual components.

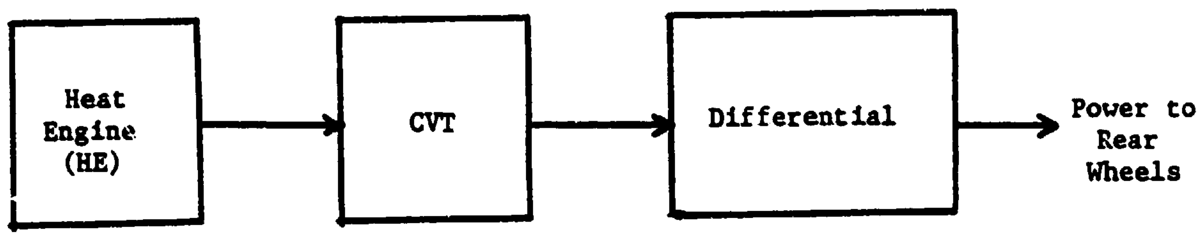

Figure A1 <u>Drive Train Configurations</u>

<u>Heat Engine</u>: The heat engine model is shown in Figure A2 and consists of a steady state engine map (which determines a steady state engine torque, T_m, for a given throttle setting, u_1, and engine speed, ω_E), a first order lag which models the throttle linkage and engine combustion dynamics (see Section 2.1 under heat engine model), and a rotating inertia.

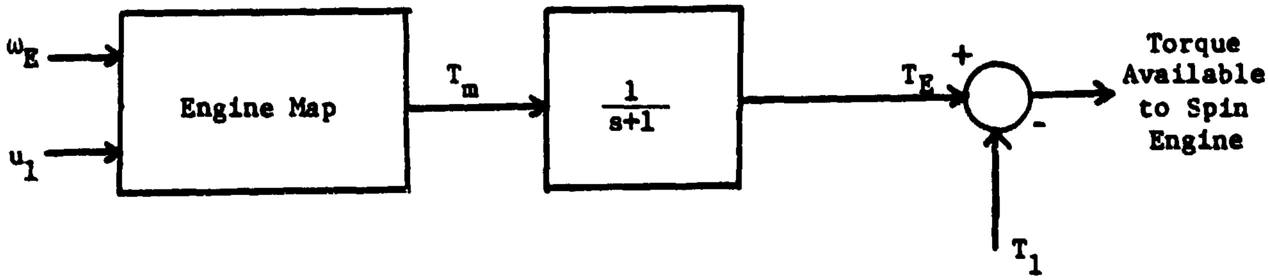

Figure A2 Engine Model

Continuously Variable Transmission: The CVT is modeled as shown in Figure A3 and consists of an inertia J_T, an efficiency K_T and a ratio, R_T, which is allowed to vary continuously from zero to infinity. The inertia is defined to be the equivalent CVT inertia as seen on the drive shaft side of the CVT. It is also assumed that there are second order dynamic effects associated with changing the ratio, i.e. if there is a step change in the variable, u_2, which controls the ratio, then the actual ratio will respond with a second order response.

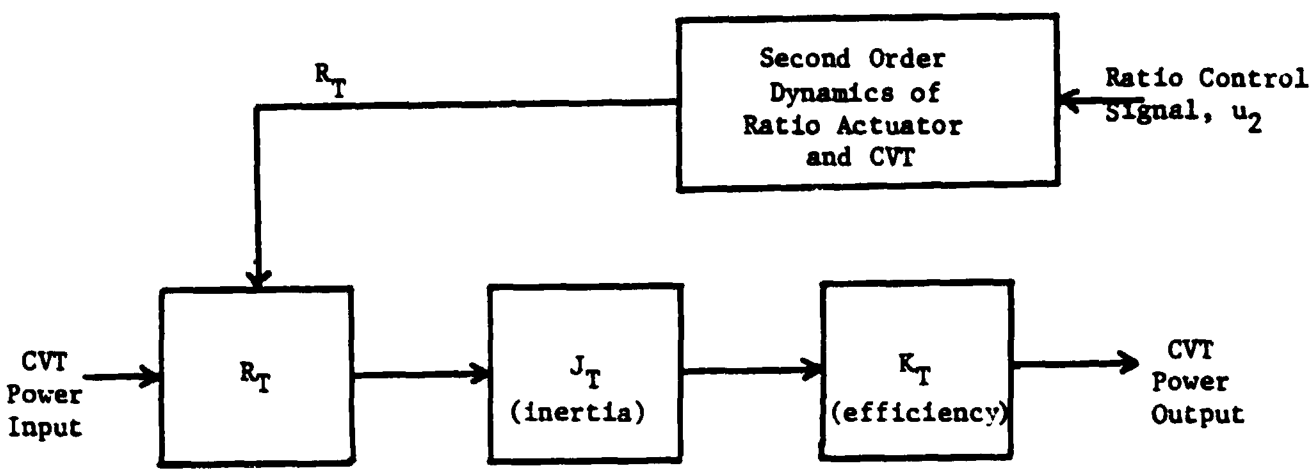

Figure A3 CVT Model

Differential: The differential is modeled in exactly the same way as the CVT, except that the ratio is fixed; hence there is a differential inertia, J_A (defined to be the inertia of the differential as seen on the axle side of the differential), an efficiency, K_A, and a ratio, R_A (defined to be the ratio of axle speed to drive shaft speed).

Vehicle Body: The vehicle body is assumed to consist of a mass to be accelerated (up or down a grade, as necessary) which is subject to aerodynamic drag and rolling resistance forces and braking torque.

In deriving the equations which describe the dynamic behavior of the drive train and vehicle body, we combine all of the inertias downstream of the CVT with the CVT inertia to form an equivalent load inertia; we also develop an expression for an equivalent CVT load torque which includes the combined effects of the aerodynamic drag and rolling resistance forces, the braking torque and the force necessary to accelerate the vehicle mass. The overall model is shown in Figure A4. In addition, as mentioned earler, the CVT ratio, R_T, has a second order dynamic response to a change in the ratio actuator input, u_2, as shown in Figure A3. Finally, the engine model includes a first order lag between the engine map and the developed engine torque, T_E, to simulate the dynamics of the throttle linkage and engine combustion. This is shown in Figure A2.

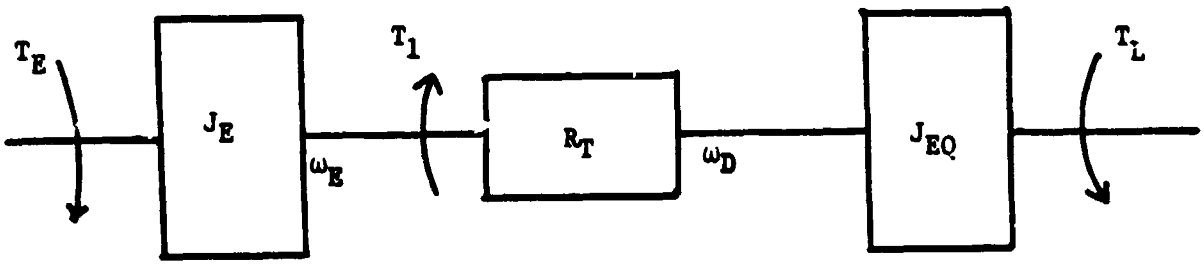

Figure A4 <u>Overall Model of Drive Train</u>

From Figure A4 we can write

$$J_E\dot{\omega}_E = T_E - T_1 \tag{A1}$$

$$J_{EQ}\dot{\omega}_D = (T_1/R_T) - T_L \tag{A2}$$

and

$$\omega_D = R_T\omega_E \tag{A3}$$

where J_E is the engine inertia. Differentiating (A3) and noting that the ratio is time varying, we get

$$\dot{\omega}_D = \dot{R}_{T\ E} + R_T\dot{\omega}_E \tag{A4}$$

Using (A4) in (A2) and solving for T_1 yields

$$T_1 = R_TJ_{EQ}(\dot{R}_T\omega_E + R_T\dot{\omega}_E) + R_T T_L \tag{A5}$$

Then using (A5) in (A1) we can write

$$J_E\dot{\omega}_E = T_E - R_TJ_{EQ}(\dot{R}_T\omega_E + R_T\dot{\omega}_E) - R_T T_L$$

or

$$(J_E + R_T^2\, J_{EQ})\, \dot{\omega}_E = -R_T \dot{R}_T\, J_{EQ}\omega_E + T_E - R_T T_L \qquad \text{(A6)}$$

It is now necessary to develop expressions for J_{EQ} and T_L to substitute into (A6). First consider J_{EQ}. This inertia consists of the sum of the CVT inertia, J_T, plus the differential inertia reflected through the differential ratio and taking into account the CVT efficiency (see Figure A3; note that J_T is upstream of K_T and that the equivalent inertia J_{EQ} is being computed at the position of J_T), and the combined axle and wheel inertias, J_W, reflected through the differential ratio and taking into account the efficiencies of both the differential and the CVT. Following standard procedures for reflecting inertias, we can write

$$J_{EQ} = R_T + (R_A^2/K_T)J_A + (R_A^2/K_T)J_A + (R_A^2/K_A K_T)\, J_W \qquad \text{(A7)}$$

Next, we derive an expression for the equivalent load torque, T_L, as seen by the CVT. The forces acting on the vehicle are given as

$$\text{Drag Force} = D = \tfrac{1}{2}(\rho/g)C_D A(v + v_W)^2\, \mathrm{sgn}(v + v_W) \qquad \text{(A8)}$$

$$\text{Rolling Resistance Force} = R = \mu W(1+1.4\times 10^{-3}v + 1.2\times 10^{-5}v^2) \qquad \text{(A9)}$$

$$\text{Acceleration Force} = (W/g)\dot{v} \qquad \text{(A10)}$$

$$\text{Grade Force} = W\sin\beta \qquad \text{(A11)}$$

Multiplying the sum of these forces by the wheel radius, and adding the braking torque to the result, we obtain the torque,

T_A, required at the axle necessary to provide the vehicle with acceleration $\dot{v}$:

$$T_A = R_W\,[D + R + W\sin\beta + (W/g)\dot{v}] + T_B \tag{A12}$$

Reflecting this torque through the differential efficiency and ratio, and through the CVT efficiency, we obtain

$$T_L = T_A(R_A/K_A K_T) \tag{A13}$$

Finally, we note that

$$v = R_A R_W R_T \omega_E \tag{A14}$$

and

$$\dot{v} = R_A R_W(\dot{R}_T \omega_E + R_T \dot{\omega}_E) \tag{A15}$$

We are now in a position to derive the first (of four) state equations which govern the vehicle dynamics. Substituting (A14) and (A15) into (A12), and that result into (A13), we can express T_L as a function of ω_E and $\dot{\omega}_E$ rather than v and $\dot{v}$. Then substituting that result for T_L along with (A7) into (A6), and using the notation $x_1 = \omega_E$, $x_2 = T_E$, $x_3 = R_T$ and $x_4 = \dot{R}_T$, we get

$$\dot{x}_1 = [x_2 - M_T x_3\,(\phi_1 + \phi_2 + R_A R_W x_1 x_4)]/(J_E + M_T R_A R_W x_3{}^2) \tag{A16}$$

where

$$\phi_1 = R_A R_W(D+R)/M_T K_A K_T \tag{A17}$$

$$\phi_2 = R_A(T_B + R_W W\sin\beta)/M_T K_A K_T \tag{A18}$$

$$M_T = (J_T/R_A R_W) + (R_A J_A/K_T R_W) + (R_A R_W M_V/K_A K_T) \tag{A19}$$

$$M_V = (W/g) + (J_W/R_W^2) \tag{A20}$$

The above state equation describes the response of engine speed, x_1 to changes in developed engine torque, x_2, CVT ratio, x_3 and x_4, grade angle, β, braking torque, T_B, and wind velocity, v_W. Given engine speed, vehicle speed can be determined from (A14).

The second state equation describes the response of developed engine torque to changes in engine speed and throttle setting (Characterizes throttle linkage and engine combustion dynamics, as well as engine steady-state torque-speed characteristics). Referring to Figure A5 we can write

$$\dot{x}_2 = -[x_2 - \phi_3(x_1, u_1)]/\tau_L \tag{A21}$$

where u_1 is throttle setting and $\phi_3(x_1, u_1)$ is the torque from the steady state engine map.

The third and fourth state equations describe the second order dynamic response of the ratio, $x_3 = R_T$, to changes in the ratio actuator input, u_2. In particular,

$$\dot{x}_3 = x_4 \tag{A22}$$

$$\dot{x}_4 = C_1 x_4 + C_2 x_3 + C_3 u_2 \tag{A23}$$

where C_1, C_2 and C_3 are constants which determine the actuator gain and second order dynamics.

Equations (A16), (A21), (A22) and (A23) are the vehicle state equations used in the study.

APPENDIX B: PARAMETER VALUES

Parameter	Description	Value
A	frontal area of vehicle	$2.00\,m^2$
C_D	vehicle drag coefficient	0.45
C_1	constant describing actuator dynamics	$-12.0\ s^{-1}$
C_2	constant describing actuator dynamics	$-12.0\ (s^2)^{-1}$
C_3	ratio actuator gain constant	$20.0\ (v\text{-}s^2)^{-1}$
g	gravity constant	$9.807\,m/s^2$
J_A	inertia of differential (as seen on axle side of differential)	$0.02\,m^2\text{-}kg$
J_E	engine inertia	$0.113\,m^2\text{-}kg$
J_T	CVT (drive shaft side of CVT)	$0.60\,m^2\text{-}kg$
J_W	combined inertia of all four wheels	$7.052\,m^2\text{-}kg$
K_A	efficiency of differential	0.96
K_{EI}	integral gain, engine speed controller	$0.09\ rad^{-1}$
K_{EP}	proportional gain, engine speed controller	0.018 s/rad
K_{RP}	ratio proportional controller gain	2.6 v
K_T	efficiency of transmission	0.9
K_{VI}	integral gain, velocity controller	$1.8\ m^{-1}$
K_{VP}	proportional gain, velocity controller	20.4 s/m
R_A	differential ratio (ratio of axle speed to drive shaft speed)	0.362
R_W	vehicle wheel radius	0.295 m
W	vehicle total weight	15,124 n
β	grade angle	0.0
μ	coefficient of rolling friction	0.0154 n/kg
ρ	air weight density	$12.02\ n/m^3$
τ_E	engine speed filter time constant	1.0 s
τ_T	throttle filter time constant	1.0 s
τ_L	Time constant: linkage dynamics	0.5 s

APPENDIX C: LISTING OF SIMULATION PROGRAM

```
FTN4,L
      PROGRAM HCVT(3,100)
C     *****************************************************************
C     *****************************************************************
C     *******  HEAT ENGINE/CVT SYSTEM SIMULATION  *********************
C     *****************************************************************
C     *****************************************************************
C
C
C     *****************************************************************
C     **************   SYSTEM VARIABLES   *****************************
C     *****************************************************************
C     A=VEH. FRONTAL AREA,(M**2).         BETA(I)=ROAD GRADE OVER DRIVING
C     BP=XMISSION DAMPING FACTOR.                 CYCLE,(DEGREES).
C     CD=VEH. DRAG COEFF.                 D=VEH. DRAG FORCE,(N).
C     DCP=DRIVING CYCLE PERIOD,(S).       DELT=SIMUL. TIME STEP SIZE,(S).
C     ENGMAP(I,J)=ENG. TORQUE WHEN        ENGSP(I)=POSSIBLE ENGINE SPEEDS
C       ENG. SPEED IS ENGSP(I) AND                 ON ENG. MAP,(R/S).
C       PEDAL IS PEDPCT(J),(N-M).         ER=XMISSION RATIO ERROR.
C     EW=ENG. SPEED ERROR,(R/S).          FUAV=AVER. FUEL CONSUMPTION
C     FUEL=FUEL CONSUMP. RATE FOR               RATE,(LBS/HR).
C          GIVEN ENG. SPEED AND           FURATE(I,J)=ARRAY OF VALUES OF
C          PEDAL,(LBS/HR).                   FUEL CONSUMP. RATES FOR GIVEN
C     G=GRAVITY CONST.,(M/S**2).             ENG. SPEED AND PEDAL,(LBS/HR)
C     INPRIN=NO. OF SIMUL. STEPS          JA=AXLE INERTIA,(KG-M**2).
C            BETWEEN RESULTS PRINTS       JE=ENG. INERTIA,(KG-M**2).
C     JPR=COUNTER USED TO DETERMINE       JT=XMISSION INERTIA,(KG-M**2).
C         IF PRINT/OUT SHOULD OCCUR       JWF,JWR=FRONT AND REAR AXLE
C     KA=EFF. OF REAR AXLE DIFF'TL.               INERTIAS,(KG-M**2).
C     KCEI,KCEP=INTEGRAL AND PROP.        KCR=RATIO CONTROLLER GAIN.
C        GAINS,ENG. SPD. CONTROLLER       KG=XMISSION RATIO ACTUATOR GAIN.
C     KP=SPR. CONST. OF XMISSION          KPDI,KPDP=INT. AND PROP. GAINS,
C        RATIO MOVABLE PULLEY.                     PEDAL CONTROLLER.
C     KR=XMISSION PULLEY RATIO/DIS-       KT=XMISSION EFF.
C        PLACEMENT GAIN.                  L=CURRENT PT. IN DRIVING CYCLE.
C     LL=L+1                              MET=METRIC PRINT FLAG, 1=PRINT
C     MP=MASS OF XMISSION PULLEY.             METRIC, 0=PRINT ENGLISH.
C     MT=COMPOSITE EXPRESSION USED        MU=COEFF. OF ROLLING FRIC.,N/KG.
C        IN VDOT EQN.,(KG-M).             MV=EQUIV. VEH. MASS,(KG).
C     MWF,MWR=COMBINED MASS OF            N=NO. OF STEPS IN DRIVING CYCLE.
C        FRONT AND REAR WHEELS,(KG)       NIN=NAME OF SUBR. WHICH
      NL=N+1                                  INTEGRATES STATE EQNS.
C     NM1=N-1                             NPR=COUNTER TO DETERMINE IF OUT-
C     NPRMAX=NO. OF OUTPUT LINES              PUT PAGE HEADER IS PRINTED.
C           PRINTED PER PAGE .            NV=NO. OF VAR'S INTERG'ED BY NIN
C     OMAGAE=ENG. SPEED,(RAD/SEC).        OMEGSP=ENG. SPD. SET PT.,(R/S).
C     OMPR=VALUE OF ENG. SPD. SET         OPTSP(I)=BEST ENG. SPD. SET PT.
C          PT. USED FOR PRINTING.            WHEN PDL. HAS VALUE PEDPCT(I)
C     PEDL=VALUE OF PDL USED IN RHS       PEDPCT(I)=PDL. VALUES CORRES. TO
C     PEDPOS(I)=VALUE OF PDL. CORR.           I, USED IN ENGINE MAP.
C       TO I, USED IN MIN FUEL CURV       PEDPRP=CONTRIB. TO PDL FROM PROP
C     PHI1=FCN WHICH COMPUTES                    PART OF PDL PI CONTROLLER
C          EFFECTS OF DRAG AND            PHI2=FCN WHICH COMPUTES EFFECTS
C          ROLLING RESISTANCE.                 OF BRK. TORQUE AND GRADE.
C     PHI3=COMPUTES ENG TORQUE FOR        PHI4=COMPUTES ENG SPD SET PT.FOR
```

```
C             GIVEN ENG SPD AND PEDAL.             MIN FUEL CONSUMPTION.
C        PHI5=LIMITS RATIO.
C        PHI6=COMPUTES FUEL CONSUMP.        PL=THROTTLE SETTING.
C        R=ROLLING RESISTANCE,(N).          RA=REAR AXLE RATIO.
C        RHO=AIR WT. DENSITY,(KG/M**3)      RHS=COMPUTES RIGHT HAND SIDE OF
C        RL=REAL VALUE OF L.                    STATE EQNS.
C        RLIM=LIMITED VALUE OF RATIO.       RNM1=REAL VALUE OF NM1.
C        RSP=RATIO SET PT.                  RSPMAX=MAX ALLOW VALUE OF RSP.
C        RSPMIN=MIN ALLOW VAL. OF RSP.      RSPROP=CONTRIB TO RSP FROM PROP
C        RTHAT=UNLIMITED VAL. OF RATIO             PART OF RSP PI CONTROLLER
C        RTMAX=MAX VALUE OF RATIO.          RTMIN=MIN VALUE OF RATIO.
C        RW=WHEEL RADIUS,(M).               SIMPER=SIMULATION LENGTH,(SECS).
C        T=CURRENT TIME,(SECS).             TAUL=TIME CONST OF ENG. TORQ,(S)
C        TBF,TBR=BRAKING TORQUES,FRONT      TBMAX=CONST.RELATING PEDL. POSN.
C                AND REAR WHEELS,(N-M)           AND BRAKING TORQUE.
C        TBRP=VAL OF TBR USED TO PRNT.
C        TIMARR(I)=VAL OF TIME USED IN      TIME=CURRENT TIME,(SECS).
C           IN COMP. DRIVING CYCLE VEL      TM=ENG. TORQUE FROM ENG. MAP.
C        U=RATIO CONTROLLER OUTPUT.         VDC(I)=DR CYCLE VEL PROFILE,M/S.
C        VROR=ERROR BETWEEN DRIVING         VERR=DIFF BET. AVER. OF PRESENT
C             CYCLE AND VEH. VEL'S,M/S          AND NEXT SAMPLE OF DR. CYC.
C        VRRMS=RMS VEL ERROR OVER               VEL. AND VEH. VEL.,(M/S).
C              DRIVING CYCLE.               VSP=COMPUTES PRESENT VALUE OF
C        VSPR=VSP.                              DRIVING CYCLE VEL.,(M/S).
C        VSQ=SUM OF SQ'S OF VEL ERRORS      VW(I)=WIND VEL OVER DR CYCLE,M/S
C        VWIND=PRESENT VAL OF WIND VEL      W=WT. OF VEH. AND WHEELS,(N).
C        X1=ENG.SPEED,RAD/SEC.              X1P=VAL OF X1 USED FOR PRINTING.
C        X2=DEVELOPED ENG TORQ.,N-M.        X2P=VAL OF X2 USED FOR PRINTING.
C        X3=XMISSION RATIO(UNLIMITED)       X4=DERIVATIVE OF X3
C        X5=STATE OF INTEGRATOR IN          X6=STATE OF INTEGRATOR IN PEDAL
C           ENG SPEED CONTROLLER.               CONTROLLER.
C        Y(I)=PRESENT VALUES OF STATES
C             USED BY NIN.
C        ***********************************************************************
C        ***********************************************************************
C
C
C        ***********************************************************************
C        ***************   ADDITIONAL SYSTEM VARIABLES   ************************
C        ***********************************************************************
C        IWRTE=CHANGES DATA OUTPUT, =0      NRUN=NO. OF RUNS OF SIMULATION
C              GIVES FULL OUTPUT, =1            PER PROGRAM EXECUTION.
C              GIVES REDUCED OUTPUT         VVEL=VEHICLE VELOCITY.
C        VVELP=VALUE OF VVEL USED FOR       PD=PEDAL POSITION.
C           PRINTING.                       XALP=FILTER CONST. FOR ENGINE
C        WALP=FILTER CONST. FOR ENG.            SPEED CONTROL LOOP.
C             SPEED SET POINT.              PALP=FILTER CONST. FOR PEDAL
C                                               VARIATIONS.
C        ***********************************************************************
C        ***********************************************************************
C
C
C        ***********************************************************************
C        DECLARE COMMONS, DIMENSIONS, REALS, EXTERNALS.
         COMMON LL,PEDL,VW(501),VDC(2001),TBF,TBR,BETA(501),C(91),JE,TAUL,K
```

```
     1CR,BP,KP,KR,MP,MT,MV,RA,RW,KA,KT,RHO,G,CD,A,W,MU,KG,KPDI,TIMARR(20
     101),DCP,TBMAX,KCEI,KCEP,KPDP,RSP,RNM1
      COMMON/CC/PIDLE,VVEL
      COMMON/DD/PD
      REAL JE,KA,KCR,KP,KR,KT,MP,MU,MT,MV,KG,KPDI,KCEI,KCEP,KPDP
      COMMON/AA/I
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
      DIMENSION Y(15)
      REAL JA,JT,JWF,JWR,MWF,MWR
C     EXTERNAL USED TO DENOTE SUBROUTINE NAME SPECIFIED IN A SUBROUTINE
      EXTERNAL RHS
C     *****************************************************************
C
C     FORMAT STATEMENTS.
  400 FORMAT(8F10.4)
  401 FORMAT(F10.4,5I5)
  405 FORMAT(16F5.1)
  410 FORMAT(10F8.1)
  415 FORMAT(10E8.2)
  500 FORMAT(1H1,36HSIMULATION OF HEAT ENGINE/CVT SYSTEM)
  501 FORMAT(1H0,3X,2HA=,F8.4,2X,3HBP=,F8.4,2X,3HCD=,F8.4,3X,2HG=,F8.4,2
     1X,3HJA=,F8.4,2X,3HJE=,F8.4,2X,3HJT=,F8.4,2X,6HPIDLE=,F8.4)
  502 FORMAT(1H0,1X,4HJWF=,F8.4,1X,4HJWR=,F8.4,1X,5HKCEI=,F8.4,1X,5HKCEP
     1=,F8.4,1X,4HKCR=,F8.4,2X,5HKPDI=,F8.4,2X,5HKPDP=,F8.4)
  503 FORMAT(1H0,2X,3HKR=,F8.4,2X,3HMP=,F8.4,2X,3HRA=,F8.4,2X,3HRW=,F8.4
     1,3X,2HW=,F12.4,2X,4HDCP=,F8.4,2X,3HKG=,F8.4,2X,3HKP=,F8.4)
  504 FORMAT(1H0,6X,3HMU=,F8.4,5X,4HRHO=,F8.4,3X,5HTAUL=,F8.6,3X,6HTBMAX
     1=F8.4)
  505 FORMAT(1H0,16HWIND VELOCITY = ,F8.4,5X,14HGRADE ANGLE = ,F6.2)
  506 FORMAT(1H0,16HEFFICIENCIES ARE,10X,24HREAR AXLE/DIFFERENTIAL- ,F7.
     14,5X,14HTRANSMISSION- ,F7.4)
  507 FORMAT(1H0,27HRATIO SET POINT LIMITS ARE-,2X,2F6.3)
  510 FORMAT(1H0,23HSIMULATION INTERVAL IS ,F7.2,8H SECONDS)
  511 FORMAT(1H0,17HTHE INTERVAL HAS ,I6,9H SEGMENTS)
  520 FORMAT(1H0,33HINITIAL CONDITIONS FOR SIMULATION)
  521 FORMAT(1H ,1X,7HOMEGAE=,F8.4,5H, TE=,F8.4,5H,  R=,F6.2,7H, RDOT=,F
     16.2,9H, SP INT=,F6.2,8H, PEDAL=,F7.3)
  530 FORMAT(1H1,4HTIME,2X,6HOMEGAE,3X,6HOMEGSP,3X,2HEW,6X,2HTE,6X,2HTM,
     16X,2HTH,6X,2HTB,6X,1HV,5X,3HVSP,1X,7HV ERROR,2X,4HPEDL,5X,1HU,4X,2
     1HRT,4X,3HRSP,5X,2HER,2X,4HFUEL,3X,2HX5,2X,2HX6)
  531 FORMAT(1H0,1X,5HWALP=,F8.4,10X,5HXALP=,F8.4)
  532 FORMAT(1H ,F5.1,1X,6(F7.1,1X),F6.1,1X,3(F6.2,1X),F6.2,1X,F6.2,1X,2
     1(F5.2,1X),F6.3,1X,F4.1,1X,F4.2,1X,F4.1)
  533 FORMAT(1H0,20HRMS VELOCITY ERROR =,F8.3,5X,26HAVERAGE FUEL CONSUMP
     1TION =,F8.3,5X,24HRMS ENGINE SPEED ERROR =,F8.3)
  540 FORMAT(1H ,F12.8)
  541 FORMAT(1H1,4HTIME,2X,6HOMEGAE,1X,6HOMEGSP,2X,2HEW,5X,2HTE,4X,2HTH,
     14X,2HTB,5X,1HV,4X,3HVSP,2X,4HVERR,2X,4HPEDL,3X,2HRT,2X,3HRSP)
  542 FORMAT(1H ,F5.1,3(1X,F6.1),3(1X,F5.1),4(1X,F5.2),2(1X,F4.2))
C     *****************************************************************
C
C     READ SYSTEM INPUTS.
C     READING IN PARAMETER VALUES
      READ(5,400) A,BP,CD,G,JA,JE,JT,PIDLE
```

```
      READ(5,400) KR,MP,RA,RW,W,DCP,KG,KP
      READ(5,400) MU,RHO,TAUL,TBMAX
      READ(5,400) KA,KT,RTMIN,RTMAY
      READ(5,401) DELT,N,INPRIN,MET,IWRTE,NRUN
      READ(5,400) VWIND,BETA(1)
      DO 5 I=2,501
 5    BETA(I)=BETA(1)
       VW(1)=VWIND
C     READ INITIAL STATES.
      READ(5,415) X3,X4
C     SAVE INITIAL STATES FOR SECOND RUN OF SIMULATION.
      X3I=X3
      X4I=X4
C     INITIALIZE VEHICLE VELOCITY.
      VVEL=0.0
C     READ SPEEDS ON ENGINE MAP.
      READ(5,400) (ENGSP(I),I=1,8)
      READ(5,400) (ENGSP(I),I=9,16)
      READ(5,400) (ENGSP(I),I=17,19)
C     CONVERT ENGINE SPEEDS TO RAD/SECS.
      DO 11 I=1,19
 11   ENGSP(I)=ENGSP(I)*0.10472
C     READ PEDAL PERCENTS ON ENGINE MAP.
      READ(5,405) (PEDPCT(I),I=1,10)
C     READ ENGINE MAP TORQUES CORRESPONDING TO ENGSP(I) AND PEDPCT(J).
      DO 12 I=1,19
 12   READ(5,410) (ENGMAP(I,J),J=1,10)
C     CONVERT ENGINE TORQUE TO N-M.
      DO 13 I=1,19
      DO 13 J=1,10
 13   ENGMAP(I,J)=ENGMAP(I,J)*1.35575
C     READ PEDAL PERCENTS ON MIN FUEL CURVE.
      READ(5,405) (PEDPOS(I),I=1,16)
      READ(5,405) (PEDPOS(I),I=17,21)
C     READ ENGINE SPEED VALUES ON MIN FUEL CURVE.
      READ(5,400) (OPTSP(I),I=1,8)
      READ(5,400) (OPTSP(I),I=9,16)
      READ(5,400) (OPTSP(I),I=17,21)
C     CONVERT SPEED VALUES TO RAD/SEC.
      DO 131 I=1,21
 131  OPTSP(I)=OPTSP(I)*0.10472
C     READ IN FUEL CONSUMPTION RATES.
      DO 14 I=1,19
 14   READ(5,410) (FURATE(I,J),J=1,10)
C     ******************************************************************
C
C     SET UP SAE DRIVING CYCLE.
      NM1=N-1
      RNM1=NM1
      DO 20 I=1,N
      J=I-1
      RJ=J
C     COMPUTE TIME AT EACH STEP IN SIMULATION.
      TIMARR(I)=RJ*DCP/RNM1
      T=TIMARR(I)
```

```
C     FIND DRIVING CYCLE VEL. AT EACH STEP IN SIMUL. AND STORE IN VDC(I)
  141 IF(T.GT.14.0) GO TO 15
      VDC(I)=1.43679*T
      GO TO 20
   15 IF(T.GT.64.0) GO TO 16
      VDC(I)=20.115
      GO TO 20
   16 IF(T.GT.74.0) GO TO 17
      VDC(I)=20.115-0.447*(T-64.0)
      GO TO 20
   17 IF(T.GT.83.0) GO TO 18
      VDC(I)=15.645-1.738*(T-74.0)
      GO TO 20
   18 IF(T.GT.108.0) GO TO 19
      VDC(I)=0.0
      GO TO 20
   19 T=T-108.0
      GO TO 141
   20 CONTINUE
C     ********************************************************************
C
C     OVERALL LOOP TO RUN SIMULATION TWICE.
      DO 60 JRUN=1,NRUN
C     SET STATES EQUAL TO INITIAL STATES FOR RUN.
      X3=X3I
      X4=X4I
      VVEL=0.0
C     READ IN CONTROLLER GAINS FOR THIS RUN.
      READ(5,400) JWF,JWR,KCEI,KCEP,KCR,KPDI,KPDP
      READ(5,400) WALP,XALP,PALP
C     PRINTING INPUT PARAMETERS
      WRITE(6,500)
      WRITE(6,501) A,BP,CD,G,JA,JE,JT,PIDLE
      WRITE(6,503) KR,MP,RA,RW,W,DCP,KG,KP
      WRITE(6,504) MU,RHO,TAUL,TBMAX
      WRITE(6,506) KA,KT
      WRITE(6,507) RTMIN,RTMAX
      WRITE(6,505) VWIND,BETA(1)
      L=N-1
      RL=L
C     PRINT TIME INTERVAL OF SIMULATION.
      SIMPER=RL*DELT
      WRITE(6,510) SIMPER
      WRITE(6,511) L
      WRITE(6,502) JWF,JWR,KCEI,KCEP,KCR,KPDI,KPDP
      WRITE(6,531) WALP,XALP
C     COMPUTE INITIAL VALUE OF ENGINE SPEED.
      X1=PHI4(PIDLE)
      WOLD=X1
      WNEW=X1
      X1LD=X1
      X1NW=X1
C     COMPUTE INITIAL VALUE OF DEVELOPED ENGINE TORQUE.
      X2=PHI3(PIDLE,X1)
C     COMPUTE INITIAL VALUE OF RATIO SET POINT.
```

```
      XS=0.0
      TC=0.0
C     COMPUTE INITIAL DRAG AND ROLLING RESISTANCE.
      D=0.5*(RHO/G)*CD*A*((VVEL+VW(1))**2)
      IF((VW(1)+VVEL).LT.0.0) D=-D
      R=MU*W*(1.0+1.4E-3*VVEL+1.2E-5*VVEL*VVEL)
C     INITIALIZE PEDAL CONTROLLER INTEGRATOR.
      X6=PIDLE
      WRITE(6,520)
      WRITE(6,521) X1,X2,X3,X4,X5,X6
C     INITIAL SIMULATION TIME.
      T=0.0
C     COMPUTE PEDAL CONTROLLER OUTPUT.
      VSPR=VSP(T)
      VERR=VSPR-VVEL
      PL=X6+KPDP*VERR
      POLD=PL
      PNEW=PL
      PD=PL
      IF(PL.LT.PIDLE) PL=PIDLE
C     COMPUTE THROTTLE POSITION (OUTPUT OF GAIN EQUALIZER).
      ZIDLE=PIDLE+0.001
      GO TO 24
      IF(PL.GT.63.29) GO TO 21
      IF(PL.GT.12.43) GO TO 22
      IF(PL.GT.ZIDLE) GO TO 29
      GO TO 24
   21 PL=1.087*PL-8.7
      GO TO 24
   22 PL=0.295*PL+41.33
      GO TO 24
   29 PL=PL
   24 CONTINUE
C     LIMIT THROTTLE POSITION.
      IF(PL.GT.100.0) PL=100.0
C     COMPUTE INITIAL ENGINE SPEED SET POINT.
      OMEGSP=WNEW
C     COMPUTE INITIAL RATIO SET POINT.
      RSP=X5
      RSPOLD=RSP
C     COMPUTE RATIO SET POINT LIMITS.
      RSPMAX=RTMAX*((KP+KCR*KR*KG)/(KCR*KR*KG))
      RSPMIN=RTMIN*((KP+KCR*KR*KG)/(KCR*KR*KG))
C     COMPUTE INITIAL OUTPUT OF RATIO CONTROLLER, ENGINE SPEED SET POINT
C           USED FOR PRINTING, ENGINE SPEED AND RATIO ERRORS, INTERNAL
C           ENGINE TORQUE, FUEL CONSUMPTION.
      U=KCR*(RSP-PHI5(X3))
      OMPR=PHI4(PL)
      EW=OMPR-X1
      ER=RSP-X3
      TM=PHI3(PL,X1)
```

```
      FUEL=PHI6(X1,PL)
      MV=W/G+(JWR+JWF)/(RW*RW)
      MT=JT/(RA*RW)+(RA*JA)/(KT*RW)+(RA*RW*MV)/(KA*KT)
C     COMPUTE INITIAL BRAKING TORQUE.
      TBR=0.0
      TBF=0.0
      IF(PD.LT.PIDLE) TBR=-(PD-PIDLE)*TBMAX
C     COMPUTE NO. OF ITERATIONS IN COMPUTER SIMULATION.
      NL=N-1
C     SET UP VECTOR OF CURRENT VALUES OF STATES FOR NUMERICAL INTEGR.
      Y(1)=X1
      Y(2)=X2
      Y(3)=X3
      Y(4)=X4
C     INITIAL OLD STATES OF CONTROLLER INTEGRATORS.
      X5OLD=X5
      X6OLD=X6
C     DETERMINE NO. OF OUTPUTS TO BE PRINTED (FULL OR REDUCED) AND
C     PRINT APPROPRIATE HEADING.
      IF(IWRTE.EQ.0) GO TO 32
      WRITE(6,541)
      GO TO 34
   32 WRITE(6,530)
   34 CONTINUE
C     SET UP NO. OF LINES OF OUTPUT PRINTED PER PAGE.
      JPR=0
      NPRMAX=60*INPRIN
      NPR=0
C     COMPUTE VALUES OF VARIABLES USED FOR PRINTING.
      TIME=0.0
      X1P=X1
      X2P=X2
      TBRP=TBR
      VVELP=VVEL
C     PRINT FIRST LINE OF OUTPUT.  IF MET=1,RESULTS ARE IN METRIC.
      WRITE(6,333)
  333 FORMAT(4H****)
      IF(MET.EQ.0) X1P=X1*9.5493
      IF(MET.EQ.0) OMPR=OMPR*9.5493
      IF(MET.EQ.0) EW=EW*9.5493
      IF(MET.EQ.0) X2P=X2*0.73746
      IF(MET.EQ.0) TM=TM*0.73746
      IF(MET.EQ.0) VVELP=VVEL*2.23714
      IF(MET.EQ.0) TBRP=TBR*0.73746
      IF(MET.EQ.0) VSPR=VSPR*2.23714
      IF(MET.EQ.0) VERR=VERR*2.23714
      IF(IWRTE.EQ.0) GO TO 36
      WRITE(6,542) TIME,X1P,OMPR,EW,X2P,PL,TBRP,VVELP,VSPR,VERR,PD,X3,RS
     1P
      GO TO 37
   36 WRITE(6,532) TIME,X1P,OMPR,EW,X2P,TM,PL,TBRP,VVELP,VSPR,VERR,PD,U,
     1X3,RSP,ER,FUEL,X5,X6
   37 CONTINUE
C     INITIALIZE VELOCITY SUM OF SQUARES AND AVER. FUEL CONSUMPTION.
      VSQ=0.0
```

```
      FUAV=0.0
      ENGER=0.0
C     ******************************************************************
C
C     BEGIN SIMULATION.
      DO 50 L=1,NL
      NPR=NPR+1
      JPR=JPR+1
      LL=L+1
      NV=4
      PEDL=PL
C     PERFORM NUMERICAL INTEGRATION OF STATE EQNS. OVER CURRENT TIME INT
      CALL NIN(RHS,NV,T,DELT,Y)
C     INCREMENT TIME.
      RL=L
      TIME=RL*DELT
C     UPDATE STATES WITH RESULTS FROM NUMERICAL INTEGRATION.
      X1=Y(1)
      X2=Y(2)
      IF(Y(3).GT.RTMAX) Y(3)=RTMAX
      IF(Y(3).LT.RTMIN) Y(3)=RTMIN
      X3=Y(3)
C     COMPUTE NEW VEHICLE VELOCITY.
      RLIM=PHIS(X3)
      VVEL=RA*RW*RLIM*X1
C     CALCULATE CURRENT VALUES OF SYSTEM VARIABLES.
      VSPR=VSP(T)
      OMEGSP=WNEW
      U=KCR*(RSP-RLIM)
      OMPR=WNEW
      EW=WNEW-X1
      ER=RSP-RLIM
      TM=PHI3(PL,X1)
      D=0.5*(RHO/G)*CD*A*((VVEL+VW(1))**2)
      IF((VW(1)+VVEL).LT.0.0) D=-D
      R=MU*W*(1.0+1.4E-3*VVEL+1.2E-5*VVEL*VVEL)
      FUEL=PHI6(X1,PL)
      TBR=0.0
      IF(PD.LT.PIDLE) TBR=-(PD-PIDLE)*TBMAX
C     IS IT TIME TO PRINT AN OUTPUT?
      IF(JPR.LT.INPRIN) GO TO 40
C     IF YES, COMPUTE PRINT VALUES, CONVERT UNITS IF REQUESTED, AND PRNT
      X1P=X1
      X2P=X2
      TBRP=TBR
      VVELP=VVEL
      VROR=VSPR-VVEL
      IF(MET.EQ.0) X1P=X1*9.5493
      IF(MET.EQ.0) OMPR=OMPR*9.5493
      IF(MET.EQ.0) EW=EW*9.5493
      IF(MET.EQ.0) X2P=X2*0.73746
      IF(MET.EQ.0) TM=TM*0.73746
      IF(MET.EQ.0) VVELP=VVEL*2.23714
      IF(MET.EQ.0) TBRP=TBR*0.73746
      IF(MET.EQ.0) VSPR=VSPR*2.23714
```

```
      IF(MET.EQ.0) VROR=VROR*2.23714
      IF(IWRTE.EQ.0) GO TO 38
      WRITE(6,542) TIME,X1P,OMPR,EW,X2P,PL,TBRP,VVELP,VSPR,VROR,PD,RLIM,
     1RSP
      GO TO 39
   38 WRITE(6,532) TIME,X1P,OMPR,EW,X2P,TM,PL,TBRP,VVELP,VSPR,VROR,PD,U,
     1RLIM,RSP,ER,FUEL,X5,X6
   39 CONTINUE
      JPR=0
   40 CONTINUE
C     PRINT PAGE HEADING IF APPROPRIATE.
      IF(IWRTE.EQ.0) GO TO 41
      IF(NPR.EQ.NPRMAX) WRITE(6,541)
      GO TO 42
   41 IF(NPR.EQ.NPRMAX) WRITE(6,530)
   42 CONTINUE
      IF(NPR.EQ.NPRMAX) NPR=0
C     ****************************************************************
C     COMPUTE OUTPUT OF PEDAL CONTROLLER.
C     VEL. ERROR IS BASED ON AN AVER OF PRESENT AND FUTURE VEL SET PTS.
      VERR=((VSP(T+DELT)+VSP(T))/2.0)-VVEL
C     COMPUTE INTEGRAL, PROP. AND TOTAL OUTPUT OF PI CONTROLLER.
      X6NEW=X6OLD+KPDI*VERR*DELT
      PEDPRP=KPDP*VERR
      PL=X6NEW+PEDPRP
C     ADJUST OUTPUT WITH ANTI-WIND/UP LOGIC.
      IF(PL.LE.100.0) GO TO 25
      IF(PEDPRP.GE.100.0) GO TO 23
      X6NEW=100.0-PEDPRP
      GO TO 25
   23 X6NEW=X6OLD
      PEDPRP=100.0-X6NEW
   25 CONTINUE
      PL=X6NEW+PEDPRP
C     FILTER PEDAL VARIATIONS.
      PLNEW=PALP*PLOLD+(1.-PALP)*PL
      PL=PLNEW
      PLOLD=PLNEW
      PD=PL
      IF(PL.LT.PIDLE) PL=PIDLE
      X6OLD=X6NEW
      X6=X6NEW
C     COMPUTE THROTTLE POSITION USING GAIN EQUALIZER.
      GO TO 28
      IF(PL.GT.63.29) GO TO 26
      IF(PL.GT.12.43) GO TO 27
      IF(PL.GT.ZIDLE) GO TO 31
      GO TO 28
   26 PL=1.087*PL-8.7
      GO TO 28
   27 PL=0.295*PL+41.33
```

```
      GO TO 28
   31 PL=PL
   28 CONTINUE
C     ******************************************************************
C     COMPUTE OUTPUT OF ENGINE SPEED CONTROLLER.
C     SAVE CURRENT RATIO SET POINT.
      RSPOLD=RSP
C     COMPUTE INTEGRAL, PROP. AND TOTAL OUTPUT OF PI CONTROLLER.
C     USE NEG. OF ERROR SINCE RATIO AND ERROR VARY INVERSELY.
C     FILTER ENGINE SPEED SET POINT AND ENGINE SPEED.
      WNEW=WALP*WOLD+(1.-WALP)*PHI4(PL)
      WOLD=WNEW
      X1NW=XALP*X1LD+(1.-XALP)*X1
      X1LD=X1NW
      X5NEW=X5OLD-KCEI*(WNEW-X1NW)*DELT
      RSPROP=-KCEP*(WNEW-X1NW)
      RSP=X5NEW+RSPROP
C     ADJUST OUTPUT WITH ANTI-WIND/UP LOGIC.
      IF(RSP.GE.RSPMAX) GO TO 30
      IF(RSP.GT.RSPMIN) GO TO 35
      DIFF=X5NEW-X5OLD
      IF(DIFF.GT.0.0) GO TO 35
      X5NEW=X5OLD
      RSP=RSPMIN
      GO TO 55
   30 IF(RSPROP.GE.RSPMAX) GO TO 33
      X5NEW=RSPMAX-RSPROP
      GO TO 35
   33 X5NEW=X5OLD
   35 CONTINUE
      RSP=X5NEW+RSPROP
      GO TO 53
C     INHIBIT RATIO SET POINT CHANGE IF IT MAKES
C          VEHICLE VELOCITY ERROR WORSE.
   55 IF(VERR) 51,52,52
   51 IF(RSP-RSPOLD) 53,54,54
   52 IF(RSP-RSPOLD) 54,53,53
   54 RSP=RSPOLD
      X5NEW=X5OLD
   53 CONTINUE
      X5OLD=X5NEW
      X5=X5NEW
C     ******************************************************************
C     COMPUTE SUM OF SQUARES OF VEL AND ENGINE SPEED ERRORS, AND
C          TOTAL FUEL CONSUMPTION.
      VSQ=VSQ+VERR*VERR*DELT
      ENGER=ENGER+EW*EW*DELT
   50 FUAV=FUAV+FUEL*DELT
C     COMPUTE VEL RMS ERROR, AND AVERAGE FUEL CONSUMPTION RATE.
      VRRMS=SQRT(VSQ/SIMPER)
      FUAV=FUAV/SIMPER
      ENGER=SQRT(ENGER/SIMPER)
   60 WRITE(6,533) VRRMS,FUAV,ENGER
      STOP
      END
```

```
      SUBROUTINE NIN(GR,N,X,DX,Y)
C     ************************************************************************
C     NUMERICAL INTEGRATION SUBROUTINE.  THIS SUBROUTINE INTEGRATES THE
C     STATE EQUATIONS USING A VARIABLE STEP SIZE RUNGE-KUTTA ALGORITHM.
C     GR=NAME OF SUBROUTINE WHICH COMPUTES CURRENT DERIVATIVES OF STATES
C     N=NO. OF EQNS. TO BE INTEGRATED.
C     X=CURRENT TIME.
C     DX=TIME STEP SIZE.
C     Y=VALUES OF STATES AT END OF INTEGRATION INTERVAL.
C     ************************************************************************
      DIMENSION S(15),YP(15),Y1(15),E(15),Z(15),XK(15,3),YMAX(15),Y0(15)
     1,P0(15),Y(15),P(15)
      DXMAX=DX
      DXMIN=1.0E-4
      ERRMIN=0.000005
      XF=X+DX
      IF(DXMAX.EQ.DXMIN) GO TO 300
      X0=X
      IF(X)10,10,30
   10 HT=XF
      DO 20 I=1,N
   20 YMAX(I)=ABS(Y(I))
   30 H=HT
   40 X=X0
      H=X0+AMIN1(H,DXMAX,DX)-X0
      DO 50 I=1,N
   50 Y0(I)=Y(I)
      CALL GR(Y,P,X,X0)
      DO 60 I=1,N
      P0(I)=P(I)
   60 Y(I)=Y0(I)+0.5*H*P0(I)
      X=X0+0.5*H
      CALL GR(Y,P,X,X0)
      DO 80 I=1,N
      S(I)=2.0*P(I)+P0(I)
   80 Y(I)=Y0(I)+0.5*H*P(I)
      CALL GR(Y,P,X,X0)
      DO 90 I=1,N
      S(I)=2.0*P(I)+S(I)
   90 Y(I)=Y0(I)+H*P(I)
      X=X0+H
      CALL GR(Y,P,X,X0)
      DO 100 I=1,N
  100 YP(I)=Y0(I)+H*(P(I)+S(I))/6.0
  110 X=X0+0.25*H
      DO 120 I=1,N
  120 Y(I)=Y0(I)+0.25*H*P0(I)
      CALL GR(Y,P,X,X0)
      DO 130 I=1,N
      S(I)=2.0*P(I)+P0(I)
  130 Y(I)=Y0(I)+0.25*H*P(I)
      CALL GR(Y,P,X,X0)
      DO 140 I=1,N
      S(I)=2.0*P(I)+S(I)
  140 Y(I)=Y0(I)+0.5*H*P(I)
```

```
      X=X0+0.5*H
      CALL GR(Y,P,X,X0)
      DO 150 I=1,N
      Y1(I)=Y0(I)+0.5*H*(P(I)+S(I))/6.0
150   Y(I)=Y1(I)
      CALL GR(Y,P,X,X0)
      DO 160 I=1,N
      S(I)=P(I)
160   Y(I)=Y1(I)+0.25*H*P(I)
      X=X0+0.75*H
      CALL GR(Y,P,X,X0)
      DO 170 I=1,N
      S(I)=2.0*P(I)+S(I)
170   Y(I)=Y1(I)+0.25*H*P(I)
      CALL GR(Y,P,X,X0)
      DO 180 I=1,N
      S(I)=2.0*P(I)+S(I)
180   Y(I)=Y1(I)+0.5*H*P(I)
      X=X0+H
      CALL GR(Y,P,X,X0)
      R=0.0
      DO 190 I=1,N
      Y(I)=Y1(I)+0.5*H*(P(I)+S(I))/6.0
      E(I)=(Y(I)-YP(I))/15.0
      Z(I)=AMAX1(YMAX(I),ABS(Y(I)))
      IF(Z(I).EQ.0.0) GO TO 190
      R=AMAX1(R,ABS(E(I))/Z(I))
      IF(ERRMIN-R) 185,190,190
185   IF(H-DXMIN) 190,190,270
190   CONTINUE
      DO 200 I=1,N
      Y(I)=Y(I)+E(I)
200   YMAX(I)=Z(I)
205   X0=X0+H
      IF(XF-X0) 210,210,220
210   RETURN
220   IF(R-ERRMIN) 230,230,240
230   H=H+H
240   IF(XF-X0-H) 250,250,40
250   HT=H
      H=XF-X0
      GO TO 40
270   H=AMAX1(0.5*H,DXMIN)
      DO 280 I=1,N
280   YP(I)=Y1(I)
      GO TO 110
300   KF=IFIX(DX/DXMAX)
      H=DX/FLOAT(KF)
      K=0
305   K=K+1
      DO 310 I=1,N
310   Y0(I)=Y(I)
      CALL GR(Y,P,X,X0)
      X=X+0.5*H
      DO 330 J=1,2
```

```
      DO 320 I=1,N
      XK(I,J)=H*P(I)
  320 Y(I)=Y0(I)+0.5*XK(I,J)
  330 CALL GR(Y,P,X,X0)
      X=X+0.5*H
      DO 340 I=1,N
      XK(1,3)=H*P(I)
  340 Y(I)=Y0(I)+XK(I,3)
      CALL GR(Y,P,X,X0)
      DO 350 I=1,N
  350 Y(I)=Y0(I)+(XK(I,1)+2.*(XK(I,2)+XK(I,3))+H*P(I))/6.0
      IF(K.LT.KF) GO TO 305
      X=XF
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 01725 COMMON = 00000

```
      SUBROUTINE RHS(X,XDOT,T,T0)
C     ******************************************************************
C     THIS SUBROUTINE EVALUATES THE RIGHT HAND SIDE OF THE STATE EQNS.
C     IT IS CALLED FROM NIN.
C     X=CURRENT VALUES OF STATES.
C     XDOT=CURRENT VALUES OF DERIVATIVES OF STATES (SUBROUTINE OUTPUT).
C     T=CURRENT TIME.
C     TO= APPARENTLY NOT USED IN SUBROUTINE.
C     ******************************************************************
      COMMON LL,PEDL,VW(501),VDC(2001),TBF,TBR,BETA(501),C(91),JE,TAUL,K
     1CR,BP,KP,KR,MP,MT,MV,RA,RW,KA,KT,RHO,G,CD,A,W,MU,KG,KPDI,TIMARR(20
     101),DCP,TBMAX,KCEI,KCEP,KPDP,RSP,RNM1
      COMMON/CC/PIDLE,VVEL
      COMMON/DD/PD
      REAL JE,KA,KCR,KP,KR,KT,MP,MU,MT,MV,KG,KPDI,KCEI,KCEP,KPDP
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
      DIMENSION X(15),XDOT(15)
      RLIM=PHI5(X(3))
      VVEL=RA*RW*RLIM*X(1)
      PDAL=PEDL
C     LIMIT THROTTLE POSITION.
      IF(PDAL.GT.100.0) PDAL=100.0
      IF(PDAL.LT.PIDLE) PDAL=PIDLE
C     COMPUTE BRAKING TORQUE.
      TBR=0.0
      IF(PD.LT.PIDLE) TBR=-(PD-PIDLE)*TBMAX
      TBF=TBR
      XDOT(1)=(X(2)-MT*RLIM*(PHI1(VVEL,VW(1))+PHI2(TBR,TBF,BETA(1))+(RA
     1*RW*X(4)*X(1)))))/(JE+MT*RLIM*RLIM*RA*RW)
      XDOT(2)=-(X(2)-PHI3(PDAL,X(1)))/TAUL
      XDOT(3)=X(4)
      IF((X(3).EQ.RTMAX).AND.(XDOT(3).GT.0.0)) XDOT(3)=0.0
      IF((X(3).EQ.RTMIN).AND.(XDOT(3).LT.0.0)) XDOT(3)=0.0
      XDOT(4)=-(BP*X(4)+KP*RLIM-KR*KG*KCR*(RSP-RLIM))/MP
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00368 COMMON = 10253

```
      REAL FUNCTION PHI1(VEL,VELW)
C     ***********************************************************************
C     THIS FUNCTION HAS A VALUE WHICH CONTAINS THE EFFECTS OF DRAG AND
C     ROLLING RESISTANCE ON THE VEHICLE.  IT APPEARS IN THE X3DOT EQN.
C     AND IS CALLED BY RHS.
C     VEL = VEHICLE VELOCITY.
C     VELW=WIND VELOCITY.
C     ***********************************************************************
      COMMON LL,PEDL,VW(501),VDC(2001),TBF,TBR,BETA(501),C(91),JE,TAUL,K
     1CR,BP,KP,KR,MP,MT,MV,RA,RW,KA,KT,RHO,G,CD,A,W,MU,KG,KPDI,TIMARR(20
     101),DCP,TBMAX,KCEI,KCEP,KPDP,RSP,RNM1
      COMMON/CC/PIDLE,VVEL
      REAL JE,KA,KCR,KP,KR,KT,MP,MU,MT,MV,KG,KPDI,KCEI,KCEP,KPDP
      COMMON/AA/I
C     COMPUTE DRAG FORCE.
      DEE=0.5*(RHO/G)*CD*A*((VEL+VELW)**2)
      IF((VEL+VELW).LT.0.0) DEE=-DEE
C     COMPUTE ROLLING RESISTANCE.
      RR=MU*W*(1.0+(1.4E-3)*VEL+(1.2E-5)*VEL*VEL)
      PHI1=(RA*RW*(DEE+RR))/(MT*KA*KT)
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00124 COMMON = 10253

```
      REAL FUNCTION PH12(RBT,FBT,ANGLE)
C     ********************************************************************
C     THIS FUNCTION HAS A VALUE WHICH CONTAINS THE EFFECTS OF BRAKING
C     TORQUE AND GRADE ON THE VEHICLE.  IT APPEARS IN THE X3DOT EQN.,
C     AND IS CALLED BY RHS.
C     RBT=REAR BRAKING TORQUE (BOTH WHEELS).
C     FBT=FRONT BRAKING TORQUE (BOTH WHEELS).
C     ANGLE=GRADE ANGLE.
C     ********************************************************************
      COMMON LL,PEDL,VW(501),VDC(2001),TBF,TBR,BETA(501),C(91),JE,TAUL,K
     1CR,BP,KP,KR,MP,MT,MV,RA,RW,KA,KT,RHO,G,CD,A,W,MU,KG,KPDI,TIMARR(20
     101),DCP,TBMAX,KCEI,KCEP,KPDP,RSP,RNM1
      COMMON/CC/PIDLE,VVEL
      REAL JE,KA,KCR,KP,KR,KT,MP,MU,MT,MV,KG,KPDI,KCEI,KCEP,KPDP
      COMMON/AA/I
      PHI2=RA*(RBT+FBT+RW*W*SIN(ANGLE))/(MT*KA*KT)
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00057 COMMON = 10253

```
      REAL FUNCTION PHI3(PEDAL,ESPEED)
C     ******************************************************************
C     THIS FUNCTION EQUALS THE STEADY STATE ENGINE TORQUE DEVELOPED FOR
C     A GIVEN ENGINE SPEED AND PEDAL POSITION.  IT IS CALLED BY RHS AND
C     NIN.
C     PEDAL = PEDAL POSITION.
C     ESPEED = ENGINE SPEED.
C     ******************************************************************
      COMMON/CC/PIDLE,VVEL
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
  200 FORMAT(1H0,26HPHI3 ARGUMENT OUT OF RANGE,5X,2F10.4)
C     SAVE VALUE OF PEDAL
      PLL=PEDAL
C     LIMIT THE PEDAL.
      IF(PEDAL.GT.100.0) PEDAL=100.0
      IF(PEDAL.LT.PIDLE) PEDAL=PIDLE
C     FIND THE INDICES OF THE VALUES IN THE ENG. SPEED ARRAY WHICH
C     BRACKET THE ACTUAL ENGINE SPEED.
      IMIN=1
      DO 5 I=1,19
      IF(ESPEED.GE.ENGSP(I)) IMIN=I
      IF(ESPEED.LE.ENGSP(I)) GO TO 7
    5 CONTINUE
    7 IMAX=IMIN+1
      IF(IMAX.GE.19) IMAX=19
      IF(IMAX.EQ.IMIN) IMIN=IMIN-1
C     FIND THE INDICES OF THE VALUES IN THE PEDAL PERCENT ARRAY WHICH
C     BRACKET THE ACTUAL PEDAL POSITION.
      JMIN=1
      DO 15 J=1,10
      IF(PEDAL.GE.PEDPCT(J)) JMIN=J
      IF(PEDAL.LE.PEDPCT(J)) GO TO 17
   15 CONTINUE
   17 JMAX=JMIN+1
      IF(JMAX.GE.10) JMAX=10
      IF(JMAX.EQ.JMIN) JMIN=JMIN-1
C     INTERPOLATE TO FIND THE TORQUES CORRESPONDING TO THE BRACKETING
C     VALUES OF PEDAL AND THE ACTUAL ENGINE SPEED.
   22 TLOWP=((ESPEED-ENGSP(IMIN))*ENGMAP(IMAX,JMIN)+(ENGSP(IMAX)-ESPEED)
     1*ENGMAP(IMIN,JMIN))/(ENGSP(IMAX)-ENGSP(IMIN))
      THIGHP=((ESPEED-ENGSP(IMIN))*ENGMAP(IMAX,JMAX)+(ENGSP(IMAX)-ESPEED
     1)*ENGMAP(IMIN,JMAX))/(ENGSP(IMAX)-ENGSP(IMIN))
C     INTERPOLATE BETWEEN THE ABOVE TORQUES TO FIND THE TORQUE
C     CORRESPONDING TO THE ACTUAL PEDAL POSITION.
      PHI3=((PEDAL-PEDPCT(JMIN))*THIGHP+(PEDPCT(JMAX)-PEDAL)*TLOWP)/(PED
     1PCT(JMAX)-PEDPCT(JMIN))
C     RESTORE VALUE OF PEDAL
      PEDAL=PLL
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

```
      REAL FUNCTION PHI4(PEDAL)
C     ******************************************************************
C     THIS FUNCTION COMPUTES THE ENGINE SPEED SET POINT FOR MIN FUEL
C     CONSUMPTION FOR A GIVEN PEDAL POSITION. IT IS CALLED BY MAIN.
C     PEDAL = PEDAL POSITION.
C     ******************************************************************
      COMMON/CC/PIDLE,VVEL
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
C     LIMIT THE PEDAL.
      PEDLL=PEDAL
      IF(PEDAL.LT.PIDLE) PEDLL=PIDLE
C     FIND THE INDICES OF THE VALUES IN THE PEDAL PERCENT ARRAY WHICH
C     BRACKET THE ACTUAL PEDAL PERCENT.
      IMIN=1
      DO 5 I=1,21
      IF(PEDLL.GE.PEDPOS(I)) IMIN=I
      IF(PEDLL.LE.PEDPOS(I)) GO TO 7
    5 CONTINUE
    7 IMAX=IMIN+1
      IF(IMAX.GE.21) IMAX=21
      IF(IMAX.EQ.IMIN) IMIN=IMIN-1
C     INTERPOLATE TO FIND THE ENGINE SPEED SET PT. CORRESPONDING TO THE
C     ACTUAL PEDAL.
   12 PHI4=((PEDLL-PEDPOS(IMIN))*OPTSP(IMAX)+(PEDPOS(IMAX)-PEDLL)*OPTSP(
     1IMIN))/(PEDPOS(IMAX)-PEDPOS(IMIN))
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS **   PROGRAM = 00158   COMMON = 00000

```
      REAL FUNCTION PHIS(RHAT)
C     ******************************************************************
C     THIS FUNCTION LIMITS THE TRANSMISSION RATIO.  IT IS CALLED BY
C     MAIN AND RHS.
C     RHAT=UNLIMITED RATIO VALUE.
C     ******************************************************************
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
      PHIS=RHAT
      IF(RTMIN.GT.RHAT) PHIS=RTMIN
      IF(PHIS.GT.RTMAX) PHIS=RTMAX
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00033 COMMON = 00000

```
      REAL FUNCTION PHI6(ESPEED,PEDAL)
C     ******************************************************************
C     THIS FUNCTION EQUALS THE FUEL CONSUMPTION RATE FOR A GIVEN
C     ENGINE SPEED AND PEDAL POSITION.  IT IS CALLED BY MAIN.
C     ESPEED=ENGINE SPEED.              PEDAL=PEDAL POSITION.
C     ******************************************************************
      COMMON/CC/PIDLE,VVEL
      COMMON/BB/ENGMAP(19,10),ENGSP(19),PEDPCT(10),FURATE(19,10),PEDPOS(
     121),OPTSP(21),RTMIN,RTMAX
      PEDLL=PEDAL
C     LIMIT PEDAL.
      IF(PEDAL.LT.PIDLE) PEDLL=PIDLE
C     FIND THE INDICES OF THE VALUES IN THE ENGINE SPEED ARRAY WHICH
C     BRACKET THE ACTUAL ENGINE SPEED.
      IMIN=1
      DO 5 I=1,19
      IF(ESPEED.GE.ENGSP(I)) IMIN=I
      IF(ESPEED.LE.ENGSP(I)) GO TO 7
    5 CONTINUE
    7 IMAX=IMIN+1
      IF(IMAX.GE.19) IMAX=19
      IF(IMAX.EQ.IMIN) IMIN=IMIN-1
C     FIND THE INDICES OF THE VALUES IN THE PEDAL PERCENT ARRAY WHICH
C     BRACKET THE ACTUAL PEDAL PERCENT.
      JMIN=1
      DO 15 J=1,10
      IF(PEDLL.GE.PEDPCT(J)) JMIN=J
      IF(PEDLL.LE.PEDPCT(J)) GO TO 17
   15 CONTINUE
   17 JMAX=JMIN+1
      IF(JMAX.GE.10) JMAX=10
      IF(JMAX.EQ.JMIN) JMIN=JMIN-1
C     INTERPOLATE TO FIND THE FUEL RATE CORRESPONDING TO THE BRACKETING
C     VALUES OF ENGINE SPEED AND THE ACTUAL ENGINE SPEED.
   22 FLOWP=((ESPEED-ENGSP(IMIN))*FURATE(IMAX,JMIN)+(ENGSP(IMAX)-ESPEED)
     1*FURATE(IMIN,JMIN))/(ENGSP(IMAX)-ENGSP(IMIN))
      FHIGHP=((ESPEED-ENGSP(IMIN))*FURATE(IMAX,JMAX)+(ENGSP(IMAX)-ESPEED
     1)*FURATE(IMIN,JMAX))/(ENGSP(IMAX)-ENGSP(IMIN))
C     INTERPOLATE TO FIND THE FUEL RATE CORRESPONDING TO THE ACTUAL
C     PEDAL POSITION.
      PHI6=((PEDLL-PEDPCT(JMIN))*FHIGHP+(PEDPCT(JMAX)-PEDLL)*FLOWP)/(PED
     1PCT(JMAX)-PEDPCT(JMIN))
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00345 COMMON = 00000

```
      REAL FUNCTION VSP(TIME)
C     ***********************************************************************
C     THIS FUNCTION EQUALS THE DRIVING CYCLE VELOCITY CORRESPONDING TO
C     A GIVEN TIME.  IT IS CALLED BY MAIN.
C     TIME=CURRENT TIME.                    VDC(I)=ARRAY OF DRIVING CYCLE
C                                                  VELOCITIES.
C     ***********************************************************************
      COMMON LL,PEDL,VW(501),VDC(2001),TBF,TBR,BETA(501),C(91),JE,TAUL,K
     1CR,BP,KP,KR,MP,MT,MV,RA,RW,KA,KT,RHO,G,CD,A,W,MU,KG,KPDI,TIMARR(20
     101),DCP,TBMAX,KCEI,KCEP,KPDP,RSP,RNM1
      COMMON/CC/PIDLE,VVEL
      REAL JE,KA,KCR,KP,KR,KT,MP,MU,MT,MV,KG,KPDI,KCEI,KCEP,KPDP
C     FIND THE INDEX OF THE VALUE IN THE TIMARR ARRAY WHICH CORRESPONDS
C     TO CURRENT TIME.
      I=IFIX(TIME*RNM1/DCP)+1
      J=I+1
      TMIN=TIMARR(I)
      TMAX=TIMARR(J)
C     INTERPOLATE TO FIND VELOCITY CORRESPONDING TO CURRENT TIME.
      VSP=((TIME-TMIN)*VDC(J)+(TMAX-TIME)*VDC(I))/(TMAX-TMIN)
      RETURN
      END
```

FTN4 COMPILER: HP92060-16092 REV. 2026 (800423)

** NO WARNINGS ** NO ERRORS ** PROGRAM = 00098 COMMON = 10253

APPENDIX D: ANTI WINDUP ALGORITHM

Whenever a controller contains an integral term and the actuator it is driving can limit (in one or both directions), there is the possibility that the integrator will "wind-up". In particular, if the actuator is driven to a limit and the error is not zero, the integrator will continue to increase its output even though no further reduction in the error is possible. If this continues for long enough, the integrator saturates or "winds-up". If the error then changes sign (e.g. the set point is changed) the actuator will not immediately respond to decrease the error because the saturated integrator is keeping it at its limit. This type of windup can cause instability and precautions must be taken to prevent it.

A common approach, and the one adopted here, is to prevent the integral term from further integration if the actuator is at a limit and the error is such that it would tend to windup the integrator. Since the controllers used in this work also contain proportional terms, additional logic is added to account for the interaction of these two parts of the control algorithm. Finally, because there are slight differences between the throttle and engine speed controller logics, a separate flow chart for each is included.

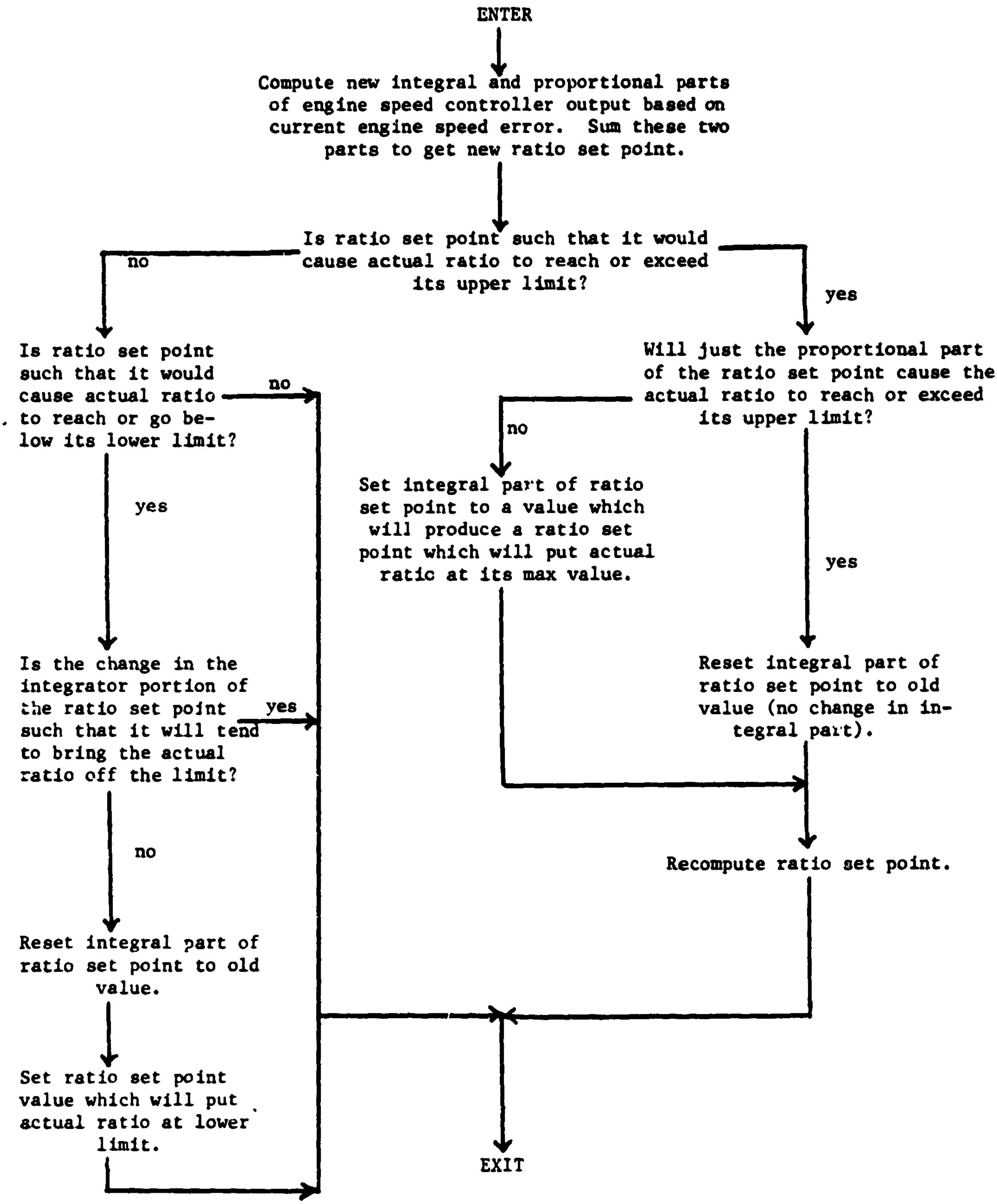

Figure D1 <u>Engine Speed Controller Anti-Windup Logic</u>

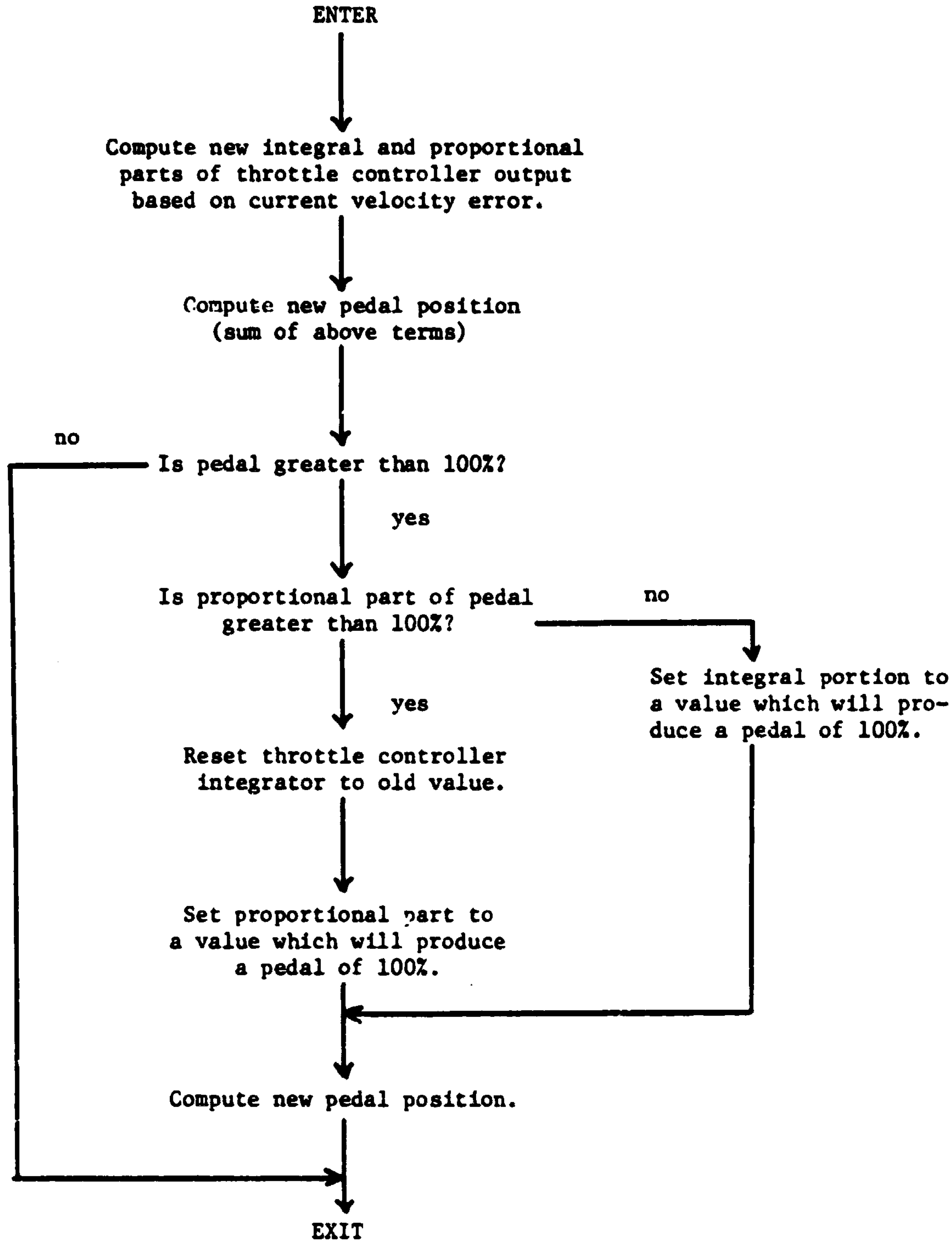

Figure D2 Throttle Controller Anti-Windup Logic